First US-Europe Air Transportation Research and Development Symposium (ATRDS2025)

Shadow Evaluation of Real-Time Machine Learning Services in the Houston Airspace

William J. Coupe
NASA Ames Research Center
Moffett Field, CA, USA
william.j.coupe@nasa.gov

Alexandre Amblard, Sarah Youlton
Crown Consulting Inc.
Moffett Field, CA, USA
alexandre.amblard@nasa.gov,
sarah.a.youlton@nasa.gov

Matthew Kistler
Mosaic ATM
Leesburg, VA, USA
mkistler@mosaicatm.com

***Abstract*—NASA is conducting a series of field evaluations between 2022 through 2030 to develop and demonstrate, in an operational environment, new technologies supporting efficient airspace operations. As part of the evaluations, NASA deployed the Machine Learning Airport Surface Model to enable pre-departure Trajectory Option Set rerouting in the North Texas airspace. After a successful field evaluation in North Texas, the Houston airspace was selected as a new location to validate scalability of the approach and benefits. This paper provides shadow mode validation results of the Machine Learning Airport Surface Model running in the Houston airspace. Shadow mode consists of the system running passively in real-time while generating predictions for departures and arrivals, but without users acting on system recommendations. The shadow evaluation is an important step towards validation to ensure behavior of the system and machine learning algorithms running in real-time matches results generated in offline training.**

***Keywords*—Machine Learning, Real-Time Services, Shadow Evaluation, Validation**

I. Introduction

For efficient airspace operations, the National Aeronautics and Space Administration (NASA) develops technologies through the Airspace Operations and Safety Program with work being executed by the Air Traffic Management - eXploration (ATM-X) Project. Within the ATM-X Project, the Digital Information Platform (DIP) Sub-project has established partnerships with the Federal Aviation Administration (FAA), the National Air Traffic Controllers Association (NATCA), five major US airlines and one regional airline (American Airlines, Delta Air Lines, jetBlue Airways, Southwest Airlines, United Airlines, and Envoy Airlines) to conduct a series of operational evaluations [1]. The field evaluations bring together a cloud based ecosystem of digital services, test vehicles, and partnerships with major US flight operators to develop and demonstrate, in an operational environment, concepts and technologies supporting efficient airspace operations.

The activities will consist of four operational evaluations between 2022 and 2030. A core concept of the evaluation series is that each capability does not live in isolation, rather the technologies are designed to build on and extend capabilities developed in previous evaluations. This enables the introduction of incremental capabilities that can be tested and evaluated in a crawl, walk, run approach.

The first evaluation is focused on single flight pre-departure Trajectory Option Set (TOS) rerouting. To enable this, NASA developed the Collaborative Digital Departure Reroute (CDDR) system that alerts flight operators to pre-departure reroute opportunities and enables electronic coordination between flight operators and Air Traffic Control (ATC) via a NASA user interface [2]. The capability was initially developed as part of NASA's Airspace Technology Demonstration 2 (ATD-2) Sub-project and deployed to the North Texas airspace including: Dallas/Fort Worth International Airport (KDFW) and Dallas Love Field Airport (KDAL) Air Traffic Control Towers, Dallas-Fort Worth TRACON (D10), Fort Worth Air Route Traffic Control Center (ZFW), American Airlines Integrated Operations Center, Southwest Airlines Network Operations Center, and Envoy Airlines Headquarters [3], [4].

The CDDR system developed under ATD-2 demonstrated efficiency benefits [4], however, was designed as a monolithic decision support tool that had challenges with scaling across the National Airspace System (NAS). To address these challenges, NASA's DIP Sub-project put the CDDR system through digital transformation resulting in a scalable system leveraging modern Machine Learning (ML) techniques, designed with a service oriented architecture, and deployed in a cloud environment [5]. The DIP system was initially validated in the North Texas airspace in field evaluations between 2022 through 2024 [1].

After validating the DIP CDDR system in North Texas, NASA worked with FAA and NATCA to identify a new location in the NAS to validate scalability of the approach and architecture. NASA investigated the top 10 busiest Terminal Radar Approach CONtrol (TRACON) facilities across the NAS and Houston TRACON was selected in coordination with stakeholders as the location for a 2025 field evaluation of the DIP CDDR system.

After selection of the Houston airspace, NASA conducted a series of observations in Houston between September 2024 through January 2025 including: George Bush Intercontinental Airport (KIAH), William P Hobby Airport (KHOU), Houston TRACON (I90), and Houston Center (ZHU). After researching the airspace, a set of initial capabilities were deployed to Houston and NASA worked with ATC and flight

operator stakeholders to refine the capabilities over time.

This paper provides shadow mode validation results of the ML Airport Surface Model in the Houston airspace. The ML Airport Surface Model is the underlying CDDR predictive engine powering the pre-departure TOS rerouting evaluation. Shadow mode consists of the system running passively in real-time while generating predictions for departures and arrivals, but without users acting on the reroute recommendations. The shadow evaluation is an important step towards validation to ensure behavior of the system and ML algorithms running in real-time matches results generated in the offline training. After validation, the system will be used for a field evaluation in Houston to capture and report out efficiency benefits.

Section II provides background about previous NASA field demos leading to the ATD-2 CDDR system and Section III discusses the digital transformation resulting in the DIP ML Airport Surface Model. Section IV provides details about the Houston airspace and Section V provides validation results for the ML predictive services. Section VI details directions for future work and Section VII provides concluding remarks.

II. Background on NASA Field Evaluations

NASA has a long history of developing and field testing new technologies in the NAS to help manage airspace operations. This work started in the early 1990's with the Traffic Management Advisor (TMA) [6] and the Center TRACON Automation System (CTAS) [7]. The CTAS/TMA tools developed by NASA were evaluated at the Fort Worth Air Route Traffic Control Center [8] and later tech transferred to the FAA and became Time Based Flow Management (TBFM) [9]. TBFM is a core Decision Support Tool (DST) for time-based management in the en route and terminal environments.

Building on work done for arrivals, departures and surface operations were incorporated using Trajectory Based Operations (TBO) concepts by NASA, the FAA, and industry to improve the flow of traffic into and out of the nation's busiest airports. NASA developed technologies for specific phases of flight were integrated [10] across surface [11]–[13] and airspace domains [14] and deployed as the Integrated Arrival Departure and Surface (IADS) traffic management system [4], [15] in 2017 to Charlotte Douglas International Airport as part of NASA's ATD-2 Sub-project. The IADS system was developed in alignment with FAA's Surface Collaborative Decision Making (S-CDM) Concept of Operations [16] and refined over time [17]. This technology was transferred to the FAA and lessons learned incorporated into the surface management solution known as Terminal Flight Data Manager (TFDM) [18].

The IADS systems generated predictions including but not limited to airport configuration, runway assignment, unimpeded taxi times, and arrival ON times [17]. The airport surface model predictions were used as input to the IADS Terminal Scheduler which applied all known constraints across each airport surface and the terminal boundary to generate predictions for the Estimated Take Off Time (ETOT) for each departure flight [19]. To generate the airport surface model predictions, the IADS system relied upon detailed adaptation developed for each individual airport and the terminal airspace.

Adaptation for each airport requires creating a detailed structure of a link-node network defining the gate locations, ramp and taxiway structure, and runway locations. The adaptation goes beyond defining the physical structure of the airport and also requires detailed knowledge from ATC encoded in decision trees including departure fix to runway mappings and other local knowledge that might be applied to the airport or airspace. Creation of the adaptation is often a manual process that requires significant time and effort to both build and maintain.

The system as developed under ATD-2 was designed as a monolithic DST, leveraging physics-based models with adaptation, which created a bottleneck to scalability [5]. To align with the FAA's vision for an Info-Centric NAS and to address scalability challenges, the DIP Sub-project applied a digital transformation to the IADS system. The digital transformation led to the ML Airport Surface Model leveraging modern ML techniques in place of legacy physics-based models, developed with a service-oriented architecture, and deployed in a cloud environment.

III. Machine Learning Airport Surface Model

Starting with the IADS monolithic decision support tool, key capabilities were broken out as individual services. For predictive services, ML was applied to replace the physics-based services that relied on adaptation. Outputs of each individual service are made available through well defined Application Programming Interface (API) deployed on NASA's Digital Information Platform (DIP) [20]. Each service can be consumed and used as a building block for downstream service developers. By making the services available through API, others can benefit from the ML Airport Surface Model accelerating the development cycle for new capabilities that require these foundational data elements.

Figure 1 shows a detailed view of the ML Airport Surface Model architecture. The ML Airport Surface Model is deployed as a service-oriented architecture in which each logical service is distinct with well-defined inputs and outputs. The ML Airport Surface Model starts at the bottom of the figure from a foundation of raw data feeds including FAA System Wide Information Management (SWIM) data feeds and other available airline or airport data feeds. The raw data feeds contain valuable data, but can provide inconsistent information on the same flight that is difficult to reconcile in a real-time environment.

To address this challenge, NASA developed logic that could resolve data processing and mediation complexities. Much of this work is embodied in the Fuser service [21]. Both the Fuser and the ML Airport Surface Model are intended to supplement existing and planned FAA capabilities such as the SWIM data feeds. The Fuser framework mediates between the disparate sources of data, pulling in the right data, at the right time. The Fuser leverages heuristics and analysis on which data source is best to use for a specific need and provides access to the information in a well-defined, common data model.

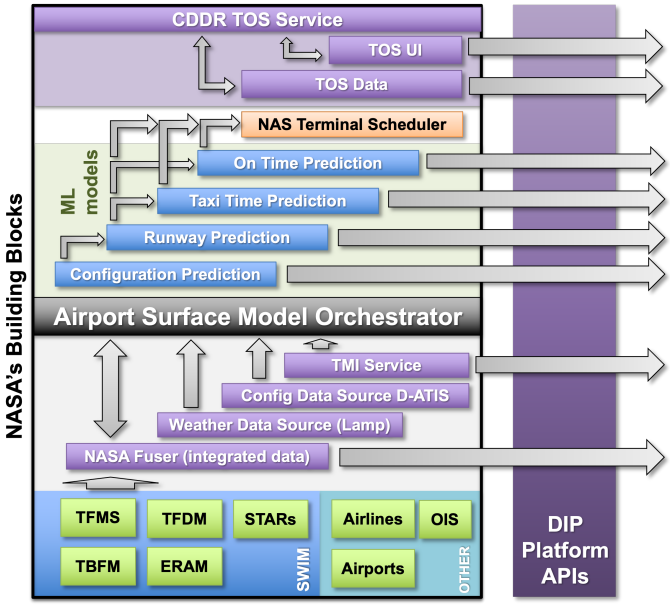

Figure 1: Predictive engine service-oriented architecture. TFMS: Traffic Flow Management System, TFDM: Terminal Flight Data Manager, STARS: Standard Terminal Automation Replacement System, TBFM: Time-Based Flow Management, ERAM: En Route Automation Modernization, SWIM: System Wide Information Management, OIS: Operational Information System, D-ATIS: Digital-Automatic Terminal Information Service

The Fuser data is used as input to the airport surface model orchestrator. The orchestrator also pulls in weather data, current airport configuration data in the form of Digital Automatic Terminal Information Service (D-ATIS), and restriction data from NASA's Traffic Management Initiative (TMI) service. The TMI service combines restriction data from FAA SWIM data feeds with local restrictions only available on the FAA Operational Information System (OIS) page. The restriction data is parsed to identify individual restrictions correlated and assigned at the flight level by the TMI service prior to being passed as input to the orchestrator.

The orchestrator is responsible for collecting the inputs required by each ML prediction service and for calling the ML services in the proper order. Even though each service is distinct with well defined inputs and outputs, there are dependencies between the different ML prediction services that need to be accounted for. Figure 1 shows the dependencies between the services as the output of the airport configuration service is used as input to the runway service. Similarly, the output of the runway service is used as input to the taxi time service and the arrival ON time service. Outputs of the runway, taxi-time, and arrival ON time services are used as input to the NAS Terminal Scheduler.

ML techniques have been applied to aviation problems for many years [22] to develop prediction models. However, the real challenge isn't building an ML model; the challenge is building an integrated ML system and to continuously operate it in production [23]. Without the proper approach, ML

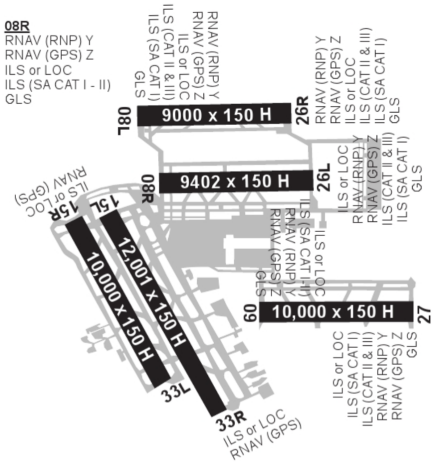

Figure 2: KIAH runways

applications can easily incur massive ongoing maintenance costs at the system level [24]. To address this challenge, in recent years there has been focused work on Machine Learning Operations (MLOps) to develop infrastructures and platforms for end-to-end life-cycle management of ML [23], [25], [26].

For deployment of the ML Airport Surface Model, we adopt MLOps best practices across the real-time system and the offline training infrastructure [5]. The adoption of MLOps best practices helps reduce risk in deployment by ensuring both the models and the pipelines feeding the models are consistent between the offline training infrastructure and the real-time deployment. MLOps best practices also allow for automation, reproducibility, monitoring, and continuous integration of ML into production software.

IV. Houston Airspace

Houston TRACON contains two major airports, George Bush Intercontinental Airport (KIAH) and William P Hobby Airport (KHOU), and six additional small regional airports. These airports share 15 departure fixes and 11 arrival fixes along the terminal boundary.

KIAH airport contains five runways, see Figure 2. Typically, departures are assigned to the diagonal runways (15L, 15R, 33L, 33R). During time periods of high demand, ATC will offload additional departure demand to one of the three parallel runways (8L, 8R, 9, 26L, 26R, 27). Arrivals will typically be assigned to one of the three parallel runways.

ATC will communicate to stakeholders of the system which runways are available for departures and arrivals through D-ATIS. D-ATIS provides text messages to aircraft, airlines, and other users outside the standard reception range of conventional ATIS via landline and data link communications to the cockpit [27]. D-ATIS includes weather information, runway serviceability and any other information considered necessary to maintain a safe ATC environment at an airport.

D-ATIS runway information is converted to a text string representing available runways. For example, when departures are able to use runways 33L and 33R D-ATIS will be consumed and translated into a string in the form D_33L_33R.

This information from D-ATIS is critical to ensure runway predictions are aligned with the actual operations.

V. Validation in Houston Airspace

Validation of the ML Airport Surface Model includes offline validation of the ML models and real-time performance monitoring. The goal of the validation is to ensure the real-time system metrics match the offline validation and that the system accuracy is high enough for operational use. Accuracy of the ML models is critical to ensure the pre-departure reroute recommendations provided to flight operators are accurate and the actual delay savings when a flight is rerouted matches the predicted delay savings [1], [3], [4].

For each ML model, we report the date range for training and testing and the associated features used. For the real-time system, the performance was evaluated on over five months of data ranging between 2024-07-01 through 2025-01-15. Each ML model running in the real-time system was developed and deployed using XGBoost [28].

A. Departure Runway Model Accuracy

The departure runway model assigns each departure to a runway that is within the set of active D-ATIS departure runways. Features of the model include: binary variable if the flight plan has been filed, departure fix name, D-ATIS airport configuration, wake vortex categorization, Airport Surface Detection Equipment Model X (ASDE-X) latitude, and ASDE-X longitude. Using these features, the model predicts the departure runway starting 3 hours before departure through the OFF event (actual take off time). The training and testing for the ML departure runway model was done with one year of data between 2022-07-01 through 2023-07-01.

Figure 3 shows the accuracy of the departure runway model. The top subplot shows the confusion matrix results from the offline validation on a test set. The horizontal axis represents the predicted runway and the vertical axis represents the true runway the flight used. The true runway is determined by processing the flown trajectory data and cross referencing the trajectory to the known positions of the runways. Each grid cell of the confusion matrix represents the percentage of overall demand that falls into that grid cell.

The bottom subplot of Figure 3 shows the accuracy of the ML departure runway model running in the real-time system colored in blue and labeled ML. The horizontal axis is the date and the vertical axis is the percentage of flights with a correct runway prediction. The runway prediction is sampled at the OUT event. We sample the prediction at the OUT (actual off block time) event because the decision to reroute flights for the CDDR service is often made by flight operators just prior to pushback.

In addition to the ML departure runway model, the real-time system also enables ATC to input a taxi plan which is a Decision Tree (DT) that defines the departure fix to runway mapping. When ATC enters a taxi plan, it defines the load balancing strategy for departures across the different runways. The taxi plan empowers ATC to have control over the runway assignments in contrast to the ML departure runway model which is trying to predict the runway assignments. The bottom

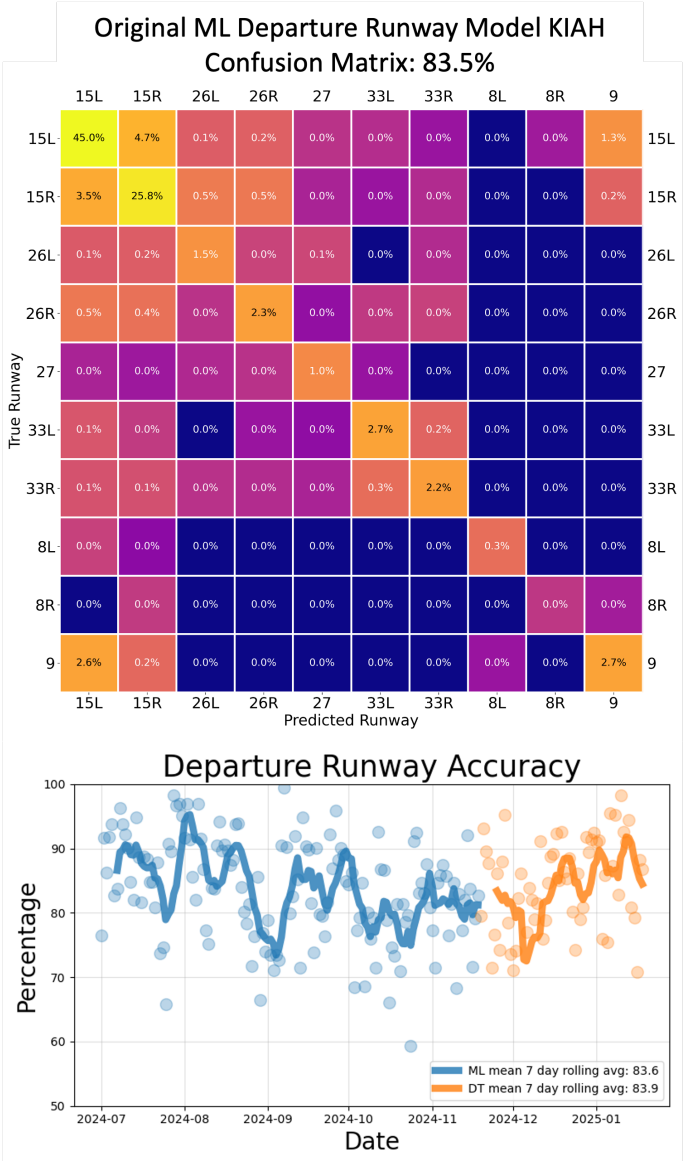

Figure 3: Departure runway accuracy

subplot of Figure 3 shows the accuracy of the departure runway assignment generated by the ATC taxi plan running in the real-time system colored in orange and labeled DT.

As can be seen in Figure 3, the ML departure runway model running in the real-time system had an overall accuracy of 83.6%, which matches the 83.5% accuracy expected from offline validation. It is also interesting to see that the ML accuracy in the real-time system is similar to the performance generated by ATC taxi plan which had overall accuracy of 83.9%. This is encouraging and indicates the predictive engine is capable of running with or without ATC inputs.

The results in the bottom subplot of Figure 3 compare the ML departure runway model to the ATC taxi plan with predictions made at the OUT event. One benefit of the ML models is that the predictions can improve as flights move towards the runway. This is in contrast to the ATC taxi plan assignments which are defined by a static decision tree and do not update unless ATC changes the taxi plan.

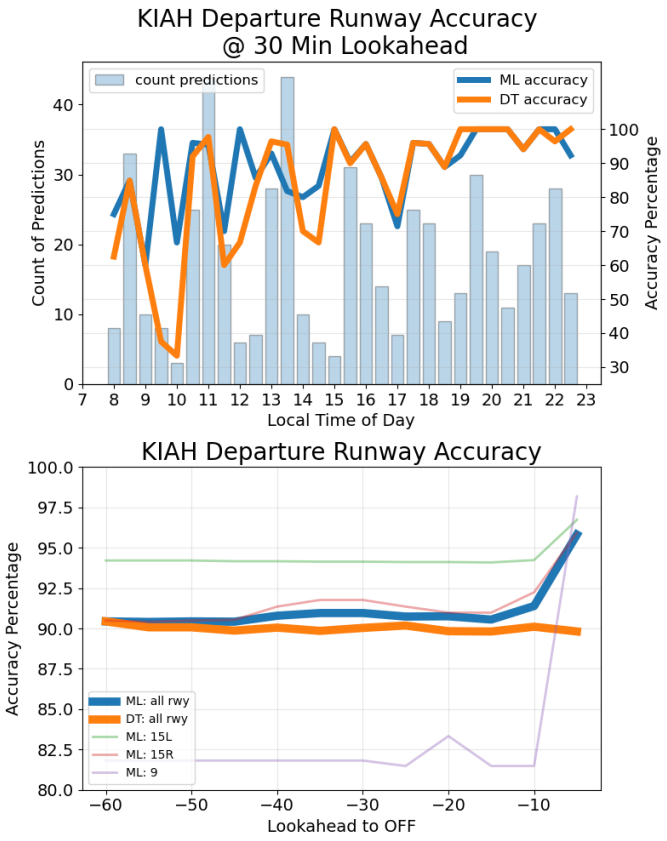

Figure 4: 2025-01-12 departure runway prediction accuracy

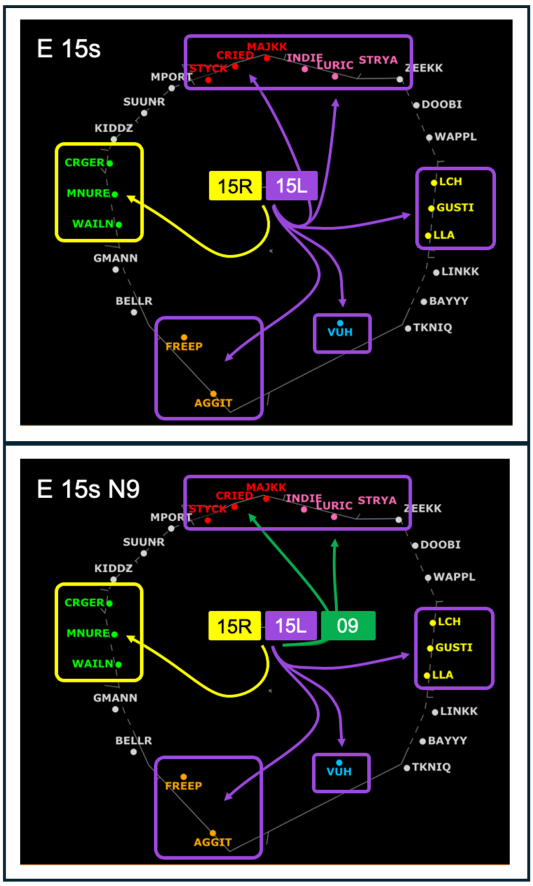

Figure 5: KIAH departure flows

Consider Figure 4 which compares the ML departure runway model results to the ATC taxi plan results for a single day on 2025-01-12. The top subplot shows the local time of day on the horizontal axis. The bar chart shows the count of departures in the given time bin and the line chart shows the departure runway accuracy for the ML model and the ATC taxi plan in blue and orange, respectively. The bottom subplot shows the accuracy of the ML model and the ATC taxi plan as a function of time before the OFF event.

As can be seen in the bottom subplot of Figure 4, the accuracy of the ATC taxi plan in orange shows constant performance 1 hour before departure through the OFF event. In contrast, the ML departure runway model shows constant performance up until the OUT event, and then the model accuracy improves after OUT and before OFF. The ML departure runway model improves after the OUT event because of the ASDE-X latitude and longitude features.

After pushback, as the flight taxies towards the runway, the ASDE-X surface surveillance information is provided to the ML model and the model learns that flights located at different locations on the surface are more likely to use specific runways. For example, when a physical queue builds at a particular runway, knowing the ASDE-X latitude and longitude for flights will enable the model to know a particular flight is in or near the physical queue, and thus very likely to use that runway.

B. Departure Runway Model Improvements

During time periods of high departure demand, ATC will often offload some of the departure demand to a third runway to reduce queue size and taxi times. An example of a load balancing strategy is shown in Figure 5. The top subplot of Figure 5 shows a scenario where departures are assigned to runways 15L and 15R and the associated departure fix to runway mappings. The bottom subplot of Figure 5 shows a scenario where departures are assigned to runways 15L, 15R, and 9 where North departures have been offloaded to the third runway 9. This dynamic use of the airspace helps ATC increase efficiency of the operations.

After deployment of the ML departure runway models and evaluating performance in the real-time system, we observed that one of the biggest challenges for the ML models is predicting the offloading of demand to the third runway. This can be seen in the bottom subplot of Figure 4 where the ML model performance is broken down by actual runway used. As shown in Figure 4, the ML model performance on runway 9 is much lower than runways 15L and 15R.

The ML model does not perform as well on runway 9 because the model struggles to understand when ATC is offloading demand to the third runway compared to time periods when ATC is utilizing runways 15L and 15R only. Ideally, ATC would communicate when demand is being offloaded to a third runway through the D-ATIS configuration.

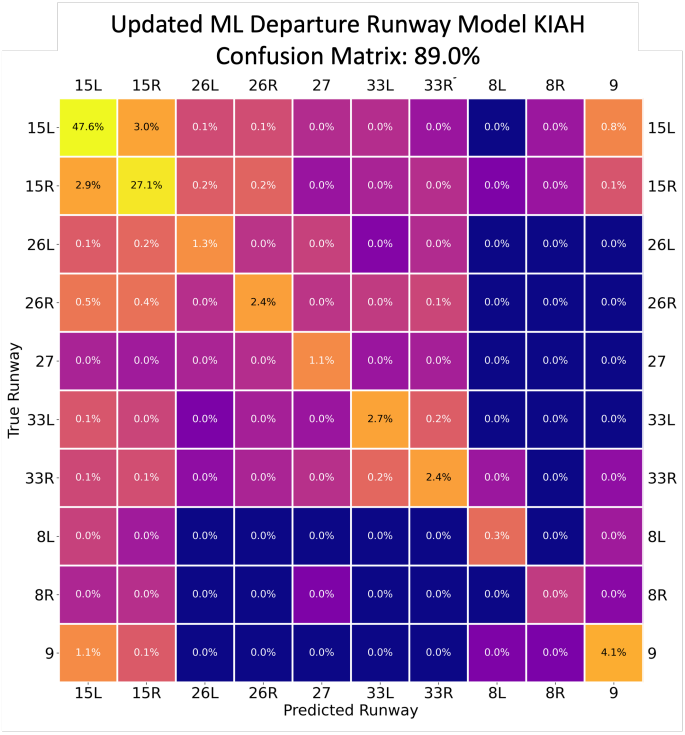

Figure 6: Offline updated ML departure runway accuracy

When ATC is using only the 15s then D-ATIS would read D_15L_15R, and when they want to offload to 9 it would read D_15L_15R_9. In practice, we find that ATC will set D-ATIS departures to D_15L_15R_9 and will determine when to offload based on their experience without communicating to stakeholders of the system.

When talking to ATC, we learned that the decision to offload demand is often based on peak surface traffic leading to high taxi times. In order to capture this information, we augmented the ML departure runway model to include local time of day. Local time of day captures time periods when demand peaks and ATC is likely to offload departures to a third runway. When using this feature, it is important to convert timestamps to local time of day, as opposed to UTC, since the scheduled demand and bank structure follows the local time.

Figure 6 shows the accuracy of the updated ML departure runway model designed to include the additional feature: local time of day. The horizontal axis represents the runway predicted by the model and the vertical axis represents the true runway the flight used. Overall, the updated ML model using local time of day increased the accuracy to 89.0% from the original 83.5%. The majority of improvement occurs on runways that are used predominately for offloading the demand.

For example, if we look at the bottom row of Figure 6 and compare to the bottom row of Figure 3, we can analyze the results for flights that actually used runway 9. For the original model shown in Figure 3, we see that overall 2.7% of flights used runway 9 and had correct predictions, however, 2.8% of flights used runway 9 and had incorrect predictions. For flights using runway 9 in the original ML model the accuracy was around 50%. Compare this to Figure 6 where we see that overall 4.1% of flights used runway 9 and had correct predictions, compared to 1.2% of flights using runway 9 with incorrect predictions. For flights using runway 9 in the updated ML model, the accuracy improved to around 77%.

C. Departure Estimated Take Off Time Accuracy

Outputs from the ML models are passed as inputs to the terminal scheduler which applies all known constraints at both the surface and terminal boundary, see Figure 1. The scheduler outputs the Estimated Take Off Time (ETOT) for each departure flight. For analysis of departure ETOT prediction accuracy, we restrict our attention to United Airlines major flights at KIAH, as these flights will participate in the upcoming field evaluation and provide Earliest Off-Block Time (EOBT) predictions.

ETOT prediction accuracy was analyzed for the time period 2024-07-01 through 2025-01-15. During this time period there were 87,138 total United Airlines major flights. To eliminate outliers from skewing the metrics, we calculate the ETOT error as the Actual Take Off Time minus the ETOT sampled at the OUT event. The Inter Quartile Range (IQR) of the error was calculated as the difference between the 25^{th} and 75^{th} quartile of the distribution of error. Outlier flights with error beyond the median ± 3.5 times the IQR were excluded from the analysis. Applying this filter for predictions at the OUT event resulted in 85,399 flights. The same filter was applied for predictions at spot crossing, resulting in a total of 82,753 flights for analysis of ETOT predictions.

Figure 7 illustrates the departure ETOT accuracy results. The top and bottom subplots of Figure 7 show the distribution of the ETOT error sampled at the OUT event and spot crossing event, respectively. The ETOT error distribution generated with the ML departure runway model are plotted in blue and results with the ATC taxi plan to assign departure runways are plotted in orange. To benchmark the ETOT accuracy results, the TFMS Estimated Time of Departure (ETD) error is plotted with a grey line.

The top subplot of Figure 7 illustrates that the standard deviation (STD) of the ETOT prediction error sampled at the OUT event using the ML departure runway model and ATC taxi plan were 7.3 minutes and 7.6 minutes, respectively. Both methods improved upon the TFMS ETD sampled at the OUT event which had a STD of error 8.8 minutes. The largest improvement can be seen along the right tail of the error distribution where the ML Airport Surface Model is more accurately predicting long taxi times for delayed flights. The improved accuracy for delayed flights is important when using the system to recommend pre-departure reroutes.

The bottom subplot of Figure 7 shows a larger improvement for ETOT predictions sampled at the spot crossing event. For ETOT predictions sampled at the spot crossing, the method using the ML departure runway model and ATC taxi plan had STD of error 3.8 minutes and 4.1 minutes, respectively. Both methods show a significant improvement to the TFMS ETD which had a STD of error 7.9 minutes. The majority of improvement compared to the TFMS ETD can be attributed

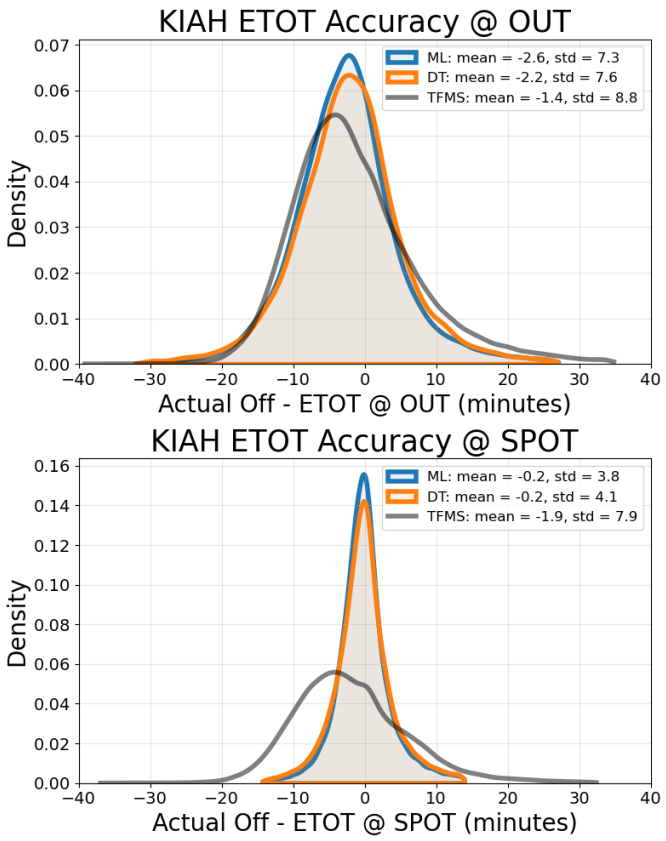

Figure 7: Estimated Take Off Time (ETOT) accuracy

to dynamic updates the ML Airport Surface Model applies at pushback and spot crossing.

The ETOT prediction for each flight is sampled from the last prediction prior to the event. Once the event happens, e.g. at the OUT event when the flight pushes back, the predictive engine updates to account for the detected activity. For example, prior to the OUT event the predictive engine will generate an Unimpeded Take Off Time (UTOT) that is equal to the Earliest Off Block Time (EOBT) plus the unimpeded taxi time.

After the flight pushes back, the predictive engine will update the UTOT to be equal to the Actual Off Block Time plus the unimpeded taxi time. A similar update occurs at the spot crossing, where the UTOT will update to be the Actual Spot Crossing Time plus the unimpeded Airport Movement Area taxi time. Since predictions are sampled prior to the event, the predictions represent what decision makers would rely upon when choosing to reroute flights and will contain the error associated with the EOBT or the ramp taxi time.

Comparing the top and bottom subplots of Figure 7 allows us to measure the benefit of updating the predictions based on the detected actual events. For example, if we look at the ETOT error STD for predictions made with the ML departure runway model, we see that just prior to the OUT event the STD is 7.3 minutes compared to the STD just prior to the spot crossing is 3.8 minutes. The main difference leading to the prediction improvement for an individual flight between the top and bottom subplot, is the UTOT update. In the top subplot, the UTOT is calculated using the EOBT as the starting point of the trajectory prior to pushback. In the bottom subplot, the UTOT is calculated with Actual Off Block Time (AOBT) after the pushback event as the starting point of the trajectory.

Prior to the digital transformation that resulted in the ML Airport Surface Model, the original IADS system continuously detected position of flights off the gate and updated 4-D trajectory predictions at 10 second intervals. The adaptation based approach leveraging physics-based 4-D trajectory predictions generated attractive results for KDFW and KDAL [3], [4], [17]. The challenge with the adaptation approach, however, is it took considerable time and effort to build and maintain for each individual airport and the terminal airspace. Previous work compared the results of the continuously updated 4-D trajectory predictions of the IADS system to the ML Airport Surface Model which only updates the trajectory at the OUT and spot crossing event [5]. Results showed the the ML Airport Surface Model could match performance while only applying two discrete trajectory updates.

D. Arrival Runway Model Accuracy

For analysis of arrival predictions, we started with all $119,870$ arrival flights to KIAH within the time range 2024-07-01 through 2025-01-15. To eliminate outliers, we calculated the error in the ON time predictions sampled at the arrival fix crossing event and the associated IQR. We filter flights with error beyond the median ± 3.5 times the IQR, resulting in $118,870$ flights for analysis. Input features for the ML arrival runway model include: arrival fix name, arrival runway prediction from TBFM, D-ATIS airport configuration, aircraft engine class, aircraft wake vortex class, last position latitude, last position longitude, and last position altitude.

Figure 8 shows the offline validation accuracy of the ML arrival runway model illustrated as a confusion matrix. The horizontal axis represents the runway predicted by the model and the vertical axis represents the true runway the flight used. The true runway is determined by processing the flown trajectory data and cross referencing the trajectory to the known positions of the runways. Each grid cell of the confusion matrix represents the percentage of overall demand that falls into that grid cell.

The overall accuracy of the ML arrival runway model in offline validation was 80.5%. As can be seen in Figure 8, the majority of arrivals use the three parallel runways with runway 9 being used the least. The limited use of runway 9 for arrivals allows ATC to use runway 9 to offload departure demand as discussed in Section V-B.

Figure 9 shows the results of the arrival runway predictions running in the real-time system. To baseline the system, we utilized arrival runway predictions from FAA's TBFM between 2024-07-01 through 2024-12-17 and with NASA's ML arrival runway model between 2024-12-18 through 2025-01-15. In the top subplot of Figure 9, the horizontal axis is the date and the vertical axis is the accuracy percentage. The accuracy for each individual day is shown with a dot and the seven day rolling average is shown with a solid line. The prediction accuracy for TBFM is illustrated in grey and the

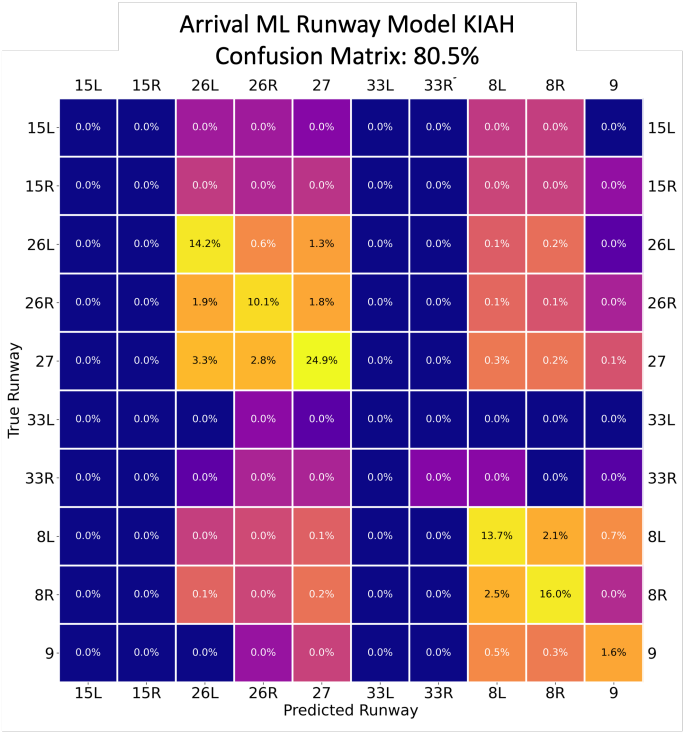

Figure 8: Offline arrival ML runway accuracy

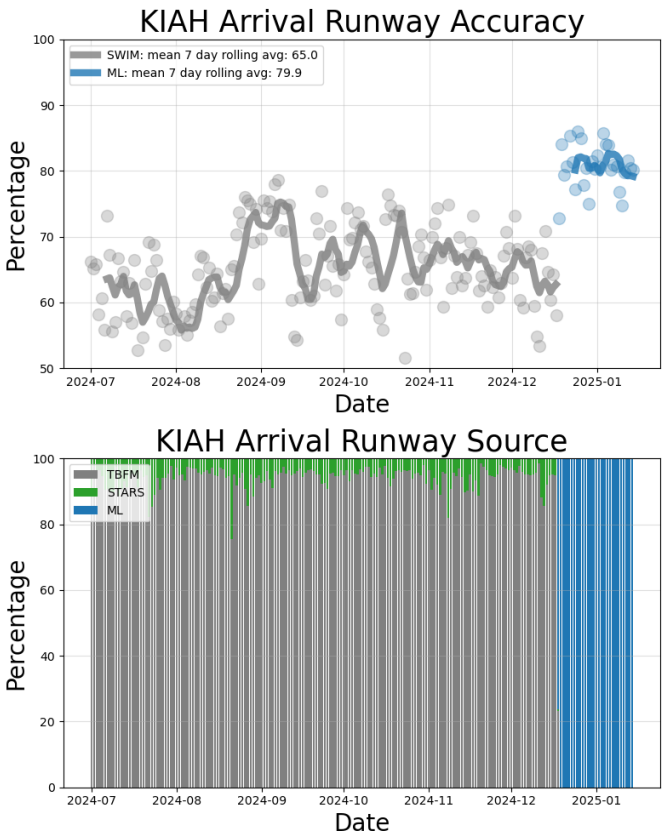

Figure 9: Real-time arrival runway accuracy

prediction accuracy for the ML arrival runway model is shown in blue.

The bottom subplot of Figure 9 shows the data source of the arrival runway prediction. If the TBFM system is predicting the runway, the data is labeled as TBFM and colored in grey. As the flight is handled by I90 TRACON, ATC has the ability to assign the flight to a runway and provide this information in the scratch pad. If the scratch pad entry is made, the arrival runway is no longer a prediction and the flight is assigned to the given runway. Flights with scratchpad entries are labeled as STARS and colored in green to illustrate what percentage of flights when crossing the arrival fix have a prediction compared to a runway assignment. Starting on 2024-12-18 the ML arrival runway model was used for all flights and labeled in the bottom subplot as ML.

As can be seen in Figure 9, the ML arrival runway model running in the real-time system had an overall accuracy of 79.9% which matched the 80.5% accuracy from the offline validation. The overall accuracy of TBFM combined with ATC scratch pad entries was 65%. The large improvement in arrival runway prediction accuracy comparing the ML approach to TBFM is consistent with prior results at KDFW [5].

E. Arrival Estimated On Accuracy

To evaluate the accuracy of the ML arrival Estimated ON (EON) model, we filter flights to the $118,870$ flights used in the analysis of Section V-D. Input features for the ML arrival EON model include: current timestamp, departure stand actual time, departure stand initial time, departure runway scheduled time, timestamp first tracked, arrival runway scheduled time, last position latitude, last position longitude, last position altitude, TBFM arrival runway, TBFM arrival runway ETA, TBFM stream class, TBFM arrival meter fix, ML model arrival runway, airline, and aircraft type.

Figure 10 shows the results of the ML arrival EON model compared to the TBFM Estimated Time of Arrival (ETA) used as a benchmark. In the top subplot, the horizontal axis represents the date and the vertical axis represents the STD of the error measured as the Actual ON minus the predicted ON sampled at the arrival fix crossing. The STD for each day is plotted with a small dot and the seven day rolling average is shown with a solid line. As can be seen, the average STD of error for the EON prediction was 2.2 minutes compared to the benchmark TBFM ETA which had 3.0 minute STD of error.

As shown in Figure 9, the real-time system had a large improvement in arrival runway prediction accuracy starting on 2024-12-18 when we transitioned from using TBFM arrival runways to the ML arrival runway prediction. When looking at the ML arrival EON model accuracy in Figure 10, we don't see an improvement in arrival ON prediction accuracy because the ML arrival EON model is using input features that include both the TBFM arrival runway prediction and the ML arrival runway prediction. Therefore, the ML arrival EON model was benefiting from the improved runway predictions throughout the entire time range between 2024-07-01 and 2025-01-15.

The middle subplot of Figure 10 compares the distribution of the ML arrival EON error and the benchmark TBFM ETA error colored in blue and grey, respectively. The ML

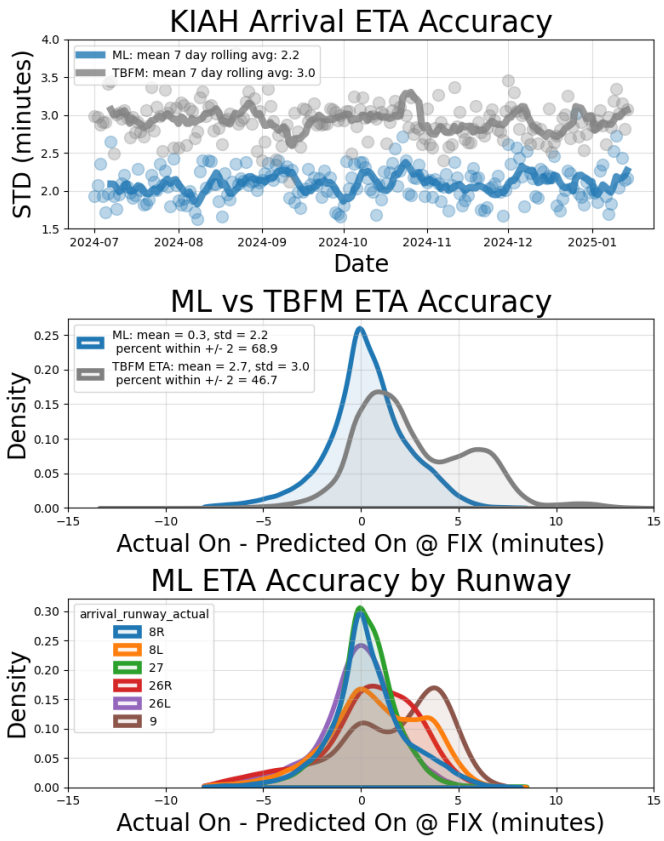

Figure 10: Real-time arrival ETA accuracy

arrival EON error had a mean value of 0.3 minutes with STD of error 2.2 minutes resulting in 68.9% of arrivals landing within ± 2 minutes of the prediction. The TBFM ETA error had a mean value of 2.7 minutes with STD of error 3.0 minutes resulting in 46.7% of arrivals landing within ± 2 minutes of the prediction. The improvement of the arrival EON predictions compared to TBFM benchmark is consistent with prior results at KDFW [5].

The bottom subplot of Figure 10 shows the distribution of the ML arrival EON error broken out by the actual runway the flight landed on. It is interesting to see the difference in the shape of the distribution for different runways. Some runways show a unimodal distribution of error whereas some runways show a heavy right tail with bimodal distribution.

Comparing the bottom subplot of Figure 10 with the confusion matrix shown in Figure 8, we see that the heavy right tails and bimodal distribution of error is associated with flights that land on runways with higher runway prediction error. This shows that one of the biggest contributions to error when predicting arrival ON times is associated with incorrectly predicting the runway the flight will use. This result increases the value of the runway prediction accuracy improvement shown in Figure 9 as the more accurate predictions from the ML arrival runway model can improve predictions of arrival ON.

VI. Future Work

Future work will use the validated ML Airport Surface model and DIP CDDR system in the Houston airspace to demonstrate benefits associated with pre-departure rerouting. An important outcome of this future work is to ensure the DIP CDDR system and operational benefits are scalable to other locations across the NAS beyond North Texas.

NASA will also investigate the use of existing FAA infrastructure including TFMS Pre-Departure Reroute (PDRR) and Route Amendment Dialogue (RAD) to facilitate the reroute request and flight plan amendment [29]. Combining NASA's ML Airport Surface Model with existing FAA infrastructure for the electronic coordination between flight operator and ATC can unlock benefits without requiring any deployment of NASA user interfaces into ATC facilities.

During the investigation of the Houston airspace, NASA also identified an opportunity to use the arrival predictions to increase the use of Established on Required Navigation Performance (EoR) procedures at KIAH airport. KIAH currently has EoR procedures, however, ATC has challenges clearing flights to fly the EoR because not all flights are equipped and capable (mixed equipage environment).

In coordination with Houston TRACON, NASA developed a tool that identifies equipped aircraft and leverages the arrival runway and estimated ON predictions to help TRACON controllers load balance the arrival demand and potentially increase the use of EoR. The EoR decision support tool will continue to mature and be evaluated throughout the Houston field evaluation.

VII. Conclusion

NASA's ML Airport Surface Model is the result of digital transformation from a legacy monolithic decision support tool to a scalable system developed with a service oriented architecture, leveraging modern ML techniques, and deployed in a cloud environment. After validating the ML Airport Surface Model in the North Texas airspace, NASA worked with FAA and NATCA to identify the Houston airspace to validate scalability of the approach and conduct a field evaluation in 2025 to demonstrate efficiency benefits resulting from pre-departure Trajectory Option Set rerouting.

This paper provides shadow mode validation results of the ML Airport Surface Model running in the Houston airspace between 2024-07-01 through 2025-01-15. Shadow mode consists of the system running passively in real-time while generating predictions for departures and arrivals, but without users acting on the system recommendations. The shadow evaluation is an important step towards validation to ensure behavior of the system and ML algorithms running in real-time matches results generated in offline training.

Validation results for departures showed that the ML departure runway model running in real-time matched the expected offline validation results. The biggest challenge with the ML departure runway model was shown to be time periods when ATC offloads demand to a third runway but does not communicate the strategy to stakeholders. To address this, an additional feature was included to the ML departure runway model to account for time of day and the accuracy was shown

to increase from 83% to 89%. The majority of improvement was shown to occur on runways that are used predominately for offloading the demand.

The Estimated Take Off Time (ETOT) prediction accuracy was shown to be similar when using either the ML departure runway model or the ATC taxi plan to generate runway assignments. The ETOT accuracy sampled at the pushback event was measured to have a standard deviation of error 7.3 minutes when using the ML models compared to 7.6 minutes when using the taxi plan. It was encouraging to see the runway and ETOT predictions when using ML can match performance of the ATC taxi plan as this indicates the system can be used with or without ATC input.

The ETOT accuracy was shown to outperform the benchmark TFMS Estimated Time of Departure (ETD) when sampled at both the OUT event and the spot crossing event. At the OUT event, the largest improvement was for flights experiencing large taxi times which is important when using the system to recommend pre-departure reroutes. At the spot crossing event, the majority of improvements with ETOT predictions is associated with the dynamic updates the ML Airport Surface Model applies to the unimpeded trajectory.

Validation results for arrivals showed that the ML arrival runway model running in real-time matched the expected offline validation results. The ML arrival runway model generated 79.9% accuracy which matched the offline validation 80.5% accuracy results. The real-time ML arrival runway model showed an improvement over the benchmark TBFM arrival runway predictions which generated 65% accuracy.

Similarly, the ML arrival estimated ON time showed an improvement over the benchmark TBFM ETA prediction accuracy resulting in mean error 0.3 minutes with 2.2 minute STD of error compared to mean error 2.7 minutes with 3.0 minute STD of error. Accuracy of the arrival runway predictions were shown to be a large driver of the accuracy for arrival ON time predictions.

Overall, shadow validation results for both the departures and arrivals was encouraging and an important step towards an operational evaluation. Future work will use the validated systems for both departures and arrivals to improve efficiency of the Houston airspace. Results from the Houston field evaluation will be reported and lessons learned will be shared.

REFERENCES

[1] W. J. Coupe and S. Saxena, "Towards sustainable aviation with efficient airspace operations," in *34th Congress of the International Council of the Aeronautical Sciences (ICAS)*, 2024.

[2] E. Chevalley, G. L. Juro, D. Bakowski, I. Robeson, L. X. Chen, W. J. Coupe, Y. C. Jung, and R. A. Capps, "Nasa atd-2 trajectory option set prototype capability for rerouting departures in metroplex airspace," in *2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC)*. IEEE, 2020, pp. 1–10.

[3] W. J. Coupe, D. Bhadoria, Y. Jung, E. Chevalley, and G. Juro, "Shadow evaluation of the atd-2 phase 3 trajectory option set reroute capability in the north texas metroplex," in *Fourteenth USA/Europe Air Traffic Management Research and Development Seminar (ATM 2021)*.

[4] ——, "Atd-2 field evaluation of pre-departure trajectory option set reroutes in the north texas metroplex," in *2022 IEEE/AIAA 41st Digital Avionics Systems Conference (DASC)*. IEEE, 2022, pp. 1–10.

[5] W. J. Coupe, A. Amblard, S. Youlton, and M. Kistler, "Machine learning airport surface model," in *2023 IEEE/AIAA 42nd Digital Avionics Systems Conference (DASC)*. IEEE, 2023, pp. 1–10.

[6] W. Nedell, H. Erzberger, and F. Neuman, "The traffic management advisor," in *1990 American Control Conference*. IEEE, 1990, pp. 514–520.

[7] H. Erzberger, "Ctas: Computer intelligence for air traffic control in the terminal area," Tech. Rep., 1992.

[8] H. N. Swenson, D. Vincent, and L. Tobias, "Design and operational evaluation of the traffic management advisor at the ft. worth air route traffic control center," in *United States/Europe Air Traffic Management Research and Development Seminar*, 1997.

[9] "Time based flow management," https://www.faa.gov/air_traffic/publications/atpubs/foa_html/chap18_section_25.html, accessed: 2024-05-15.

[10] R. Coppenbarger, Y. Jung, T. Kozon, A. Farrahi, W. Malik, H. Lee, E. Chevalley, and M. Kistler, "Benefit opportunities for integrated surface and airspace departure scheduling: a study of operations at charlotte-douglas international airport," in *Digital Avionics Systems Conference (DASC)*, 2016.

[11] Y. Jung, W. Malik, L. Tobias, G. Gupta, T. Hoang, and M. Hayashi, "Performance evaluation of sarda: an individual aircraft-based advisory concept for surface management," *Air Traffic Control Quarterly*, vol. 22, no. 3, pp. 195–221, 2014.

[12] M. Hayashi, T. Hoang, Y. C. Jung, W. Malik, H. Lee, and V. L. Dulchinos, "Evaluation of pushback decision-support tool concept for charlotte douglas international airport ramp operations," in *11th USA/Europe Air Traffic Management Research and Development Seminar*.

[13] S. Lockwood, S. Atkins, and N. Dorighi, "Surface management systems simulations in nasa's future flight central," in *AIAA Modeling and Simulation Technologies Conference and Exhibit*, 2002, p. 4680.

[14] S. A. Engelland, R. Capps, K. B. Day, M. S. Kistler, F. Gaither, and G. Juro, "Precision departure release capability (pdrc) final report," 2013.

[15] Y. Jung, W. Coupe, A. Capps, S. Engelland, and S. Sharma, "Field evaluation of the baseline integrated arrival, departure, surface capabilities at charlotte douglas interntional airport," in *Thirteenth USA/Europe Air Traffic Management Research and Development Seminar (ATM2019)*, 2019.

[16] FAA Air Traffic Organization Surface Operations Office, "U.s. airport surface collaborative decision making (cdm) concept of operations (conops) in the near-term: application of the surface concept at united states airports," 2014.

[17] W. J. Coupe, Y. Jung, H. Lee, L. Chen, and I. J. Robeson, "Scheduling improvements following the phase 1 field evaluation of the atd-2 integrated arrival, departure, and surface concept," in *Thirteenth USA/Europe Air Traffic Management Research and Development Seminar (ATM2019)*, 2019.

[18] "Terminal flight data manager," https://www.faa.gov/air_traffic/technology/tfdm, accessed: 2024-05-15.

[19] W. J. Coupe, Y. Jung, L. Chen, and I. Robeson, "Atd-2 phase 3 scheduling in a metroplex environment incorporating trajectory option sets," in *2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC)*. IEEE, 2020, pp. 1–10.

[20] M. M. Gurram, P. Hegde, and S. Saxena, "Nasa's digital information platform (dip) to accelerate nas transformation," in *AIAA AVIATION 2023 Forum*, 2023, p. 3400.

[21] "Fuser high level software description," https://aviationsystems.arc.nasa.gov/atd2-industry-workshop/fuser/High-Level-Software-Description_84377871.html, accessed: 2024-05-15.

[22] B. Sridhar, "Applications of machine learning techniques to aviation operations: Promises and challenges," in *2020 International Conference on Artificial Intelligence and Data Analytics for Air Transportation (AIDA-AT)*. IEEE, 2020, pp. 1–12.

[23] "Mlops: Continuous delivery and automation pipelines in machine learning," https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning, accessed: 2023-01-22.

[24] D. Sculley, G. Holt, D. Golovin, E. Davydov, T. Phillips, D. Ebner, V. Chaudhary, and M. Young, "Machine learning: The high interest credit card of technical debt," 2014.

[25] Y. Zhou, Y. Yu, and B. Ding, "Towards mlops: A case study of ml pipeline platform," in *2020 International conference on artificial intelligence and computer engineering (ICAICE)*. IEEE, 2020, pp. 494–500.

[26] M. M. John, H. H. Olsson, and J. Bosch, "Towards mlops: A framework

and maturity model," in *2021 47th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. IEEE, 2021, pp. 1–8.
[27] "Faa: Pilot/controller glossay," https://www.faa.gov/air_traffic/publications/atpubs/pcg_html/glossary-d.html#$DIGITAL-AUTOMATIC%20TERMINAL%20INFORMATION%20SERVICE, accessed: 2024-05-15.
[28] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16, 2016.
[29] "Faa: Traffic management national, center, and terminal: Traffic management initiatives," https://www.faa.gov/air_traffic/publications/atpubs/foa_html/chap18_section_7.html, accessed: 2024-05-15.

AUTHOR BIOGRAPHIES

Dr. William Jeremy Coupe is an Aerospace Engineer at NASA Ames Research Center. He received his BS degree in Mathematics from the University of San Francisco and both MS degree in Applied Mathematics and Statistics and PhD degree in Computer Engineering from the University of California, Santa Cruz.

Dr. Alexandre Amblard is a Data Scientist for Crown Consulting Inc. He received his BS degree in Physics from the University of Caen-Normandy and both his MS and PhD degree in Physics from the University of Paris-Saclay.

Sarah Youlton is a Software Engineer at Crown Consulting Inc. She received both her BS degree in Physics and Mathematics and her MS degree in Applied Mathematics from Santa Clara University.

Matthew Kistler is a Principal Analyst at Mosaic ATM supporting NASA Ames Research Center. He received his BS degree in Aeronautical Engineering from California Polytechnic State University in San Luis Obispo, California.