Machine Learning Approach for Aircraft Performance Model Parameter Estimation for Trajectory Prediction Applications

Aida Sharif Rohani
Universities Space Research Association
NASA Ames Research Center
Moffett Field, CA, USA
aida.sharifrohani@nasa.gov

Tejas G. Puranik
Universities Space Research Association
NASA Ames Research Center
Moffett Field, CA, USA
tejas.puranik@nasa.gov

Krishna M. Kalyanam
Aviation Systems Division
NASA Ames Research Center
Moffett Field, CA, USA
krishna.m.kalyanam@nasa.gov

***Abstract*—The accurate prediction of aircraft trajectory by ground-based decision support tools (DST) is a major concern in air traffic management (ATM). Aircraft trajectory prediction tools rely on a simplified point-mass aircraft performance model (APM) to make their predictions. Even though the performance coefficients and weight of an aircraft are a vital part of the APM's predictions and accuracy, these coefficients are proprietary in nature and therefore, unavailable to DSTs. The current ATM research focuses on improving the estimate of some APM parameters by freezing all other coefficients. This simplified approach introduces unwanted sources of bias and negatively impacts the accuracy of the performance model. In this paper, we apply machine learning (ML) models for the prediction of three key APM parameters simultaneously (two drag coefficients and the initial aircraft weight). To accomplish this, we employ an ordinary differential equation (ODE) fitting approach to generate optimized APM parameter labels customized to each individual flight record. Subsequently, we train ML models to capture the relationship between the historical data and the optimized APM parameters. Two different ML model solutions are applied and the APM coefficients are predicted for unseen flights. The results indicate that the ML models are able to capture APM parameters relationships with flight-related features with good accuracy.**

***Index Terms*—trajectory prediction, machine learning, aircraft performance model, air traffic management, drag coefficients**

Notation

h	Altitude of aircraft above mean sea level
C_L	Lift coefficient
C_{D0}	Parasite drag coefficient
κ	Induced drag coefficient
δ_{cl}	Thrust setting coefficient for climb
δ_{des}	Thrust setting coefficient for descent
m_0	Starting (take off) mass of aircraft
$m(t)$	Mass of the aircraft at time t
$m_f(t)$	Total mass of fuel consumed up to time t
V_t	True airspeed
V_w	Wind speed
T_{max}	Maximum climb thrust
AF	Acceleration Factor
g	Acceleration due to gravity
S	Wing reference area
ρ	Density of air
f	Fuel flow rate
ψ_w	Wind direction
ψ_a	Aircraft heading

I. Introduction

Air traffic trajectory prediction plays a crucial role in effective air traffic management. By accurately forecasting the trajectories of aircraft, air traffic controllers can optimize airspace utilization, enhance safety, and improve overall operational efficiency. Although trajectory prediction for conventional aircraft is well-established in terms of physics, it relies on accurate knowledge of aircraft performance model (APM) parameters (such as drag coefficients) and operational procedures specific to the flight being predicted. Over the years, advancements in data availability and computing power have contributed to improved trajectory prediction accuracy.

However, there exist obstacles to accessing the proprietary aerodynamic data (such as drag coefficients or the aircraft mass) held by aircraft manufacturers, due to its commercial and competitive nature. This vital information, crucial for trajectory prediction, is often subject to strict confidentiality. When accurate aerodynamic data is unavailable, the Base of Aircraft Data (BADA), developed by Eurocontrol, serves as a valuable resource [1]. It provides a comprehensive collection of performance characteristics for various aircraft types making it the default APM resource for current ATM studies. However, there are limitations associated with using BADA for trajectory prediction purposes. These limitations include the unavailability of precise speed intent, aircraft mass, and unknown thrust setting of the engines (which can vary depending on the flight phase). Additionally, the generalized nature of the available APM parameters in BADA may overlook specific nuances and variations that exist among different actual trajectories [2].

Inaccurate trajectory predictions can arise from various factors, including imprecise input parameters, misinterpretation of pilot intent, inaccurate APM parameters, noisy radar data, and flawed modeling assumptions. There have been numerous studies that try to quantify the sensitivity of trajectory prediction models with respect to various assumptions [3], [4]. Early validation work had already identified the importance of accurate aircraft weight information for trajectory prediction [5]. As a result, researchers have been actively investigating different approaches to estimate the unknown parameters of the APM using available data. Some of the early research in this domain relied on flight manual data for improving aero-propulsive models such as the work by Gong and Chan [6]. Many of these studies have primarily focused on optimizing a single APM parameter, often the initial aircraft weight, while making assumptions for the remaining APM parameters. Schultz et al. [7], [8] proposed a method for weight estimation during the climb phase by dynamically adjusting the weight based on thrust and drag information from BADA. Thipphavong [9] also developed a top-of-climb matching method using different aircraft weight parameters and selecting the one that best fit the data. Climb trajectory modeling has been investigated for separation assurance automation [10]. Dalmau et al. [11] focused on estimating fuel consumption in the descent phase rather than determining the initial aircraft mass based on radar tracks. Alligier et al. [12] proposed a modification to the BADA thrust model to derive an equivalent weight and thrust setting profile specific to the climb phase of flights. This modified model enables the prediction of the aircraft's energy rate. In a related study, Alligier et al. [13] focused on learning the thrust law and mass parameters using historical flight data. They utilized the BADA drag polar model and employed a data-driven approach to estimate the thrust law and mass values. In a subsequent study, Alligier et al. [14] employed machine learning (ML) methods to estimate the aircraft mass, leading to enhanced aircraft climb prediction compared to the baseline method that relies on the reference mass from BADA. They assumed the max thrust setting for the climb phase. Sun et al. [15] introduced a stochastic total energy model formulation for estimating aircraft drag parameters. This formulation is highly versatile and incorporates the thrust setting and mass estimation within the learning process. The authors incorporated maximum thrust information from BADA.

From all these studies, a strong link between aircraft mass and other APM parameters is evident. The coupling between various variables in the aircraft model can result in different estimation scenarios when using historical data to estimate APM parameters. These variations are influenced by the choice of parameters being estimated and other assumptions such as holding several parameters constant for estimating the variable of interest. Therefore, our primary focus in this study is to enhance the estimation of crucial aircraft performance model parameters simultaneously, specifically two drag coefficients and the takeoff weight. These parameters play a vital role in ensuring the accuracy of predictions made by ground-based Decision Support Tools (DSTs).

The subsequent sections of this paper are structured as follows: Section II outlines the dataset utilized in this study and explains the preprocessing steps. Section III provides a technical overview of the APM estimation process by an ordinary differential equation (ODE) fitting approach. Section IV introduces the ML solutions, as well as a description of the features and labels used. In Section V, the results are presented and discussed. In Section VI, the simulation results for a single flight are shown and finally, Section VII concludes the paper.

II. DATA PREPROCESSING

The Sherlock data warehouse at NASA Ames[1] serves as the source for the raw track data (or Integrated Flight Format (IFF) dataset), event data (EV), and meteorological data forecasts (rapid refresh (RAP) dataset) used in this study. The IFF dataset contains essential features such as flight ID, airline, aircraft type, latitude, longitude, altitude, ground speed, and UTC timestamp. The RAP data provided meteorological information for North America, including variables like wind speed, temperature, air pressure, and geopotential height. The datasets covered a duration of 340 days in the year 2019. We filtered the flights from three widely used aircraft types: B738, B737, and A320 landings at four prominent destination airports in the United States: LAX, DEN, MSP, and DFW. The origin airports of these flights contained flights from all over the United States. Engine information for each flight was obtained from the FAA. Additionally, new terms that will be used in ODE fitting process were derived from the existing data (e.g., flight phase, air density, acceleration factor (reciprocal of the energy share factor), and etc.). Following the merging of the IFF, RAP, EV, and engine datasets, the data underwent a series of preprocessing steps, which included:

- Removal of trajectories with a significant number of missing values (less than 5% of the data). Flights that lacked data for crucial segments essential for the ODE fitting process, such as the climb or descent phases, were identified and excluded.
- Filling the missing rows for categorical features. For some columns such as engine type or airport names, we implemented a backfilling technique based on the constant values observed for each flight. Since these values remained consistent throughout the flight, we used the constant values to fill in the missing rows.
- Smoothing using spline interpolation technique. The raw track data contained one position report every 6 to 12 seconds, resulting in noisy trajectory data, affecting features such as speed and altitude. To address this issue, a vital step was the application of spline interpolation techniques (by Pandas rolling window function) to smooth the trajectories.
- Feature scaling to ensure consistent scale across all features. The "MinMaxScaler" from the sci-kit-learn library transforms the features by scaling them to a specified range, typically between 0 and 1. This normalization is important when dealing with features that have different units or varying magnitudes
- Extraction of month, day, and time information from the time stamps. The time extraction allows us to capture the

[1] https://sherlock.opendata.arc.nasa.gov/sherlock_open/

temporal aspects of the flights and incorporate them as features in our ML models. The time and month of the year can be indicative of various factors such as air traffic patterns, weather conditions, and seasonal variations.
- Conversion of categorical features (aircraft type, airline, engine, month, etc.) to numerical values using "OneHotEncoder" from the sci-kit-learn library. This transformation allowed us to represent these categorical variables in a numerical format, enabling their incorporation into the ML models for analysis and prediction.

The next step was to convert the flight-related data from a timeseries format to a structured dataset where each flight corresponds to a single example. By identifying the flight phases (climb, cruise, and descent) in the flight timeseries, it became possible to extract meaningful information pertaining to each phase. As part of this process, the average values of the key parameters in the ODE fitting process, such as Rate of Climb (ROC), True Airspeed (TAS), Calibrated Airspeed (CAS), and Mach number (Mach), were calculated for each flight phase. The computed average values for these variables were then included as new features in the dataset. This transformation facilitated the creation of a dataset where each flight is represented by a single row, consolidating the relevant information for that particular flight. However, it is important to note that this conversion from a timeseries to a single-example dataset does entail a loss of some valuable information that results from removing all the temporal features. Nonetheless, this transformation significantly reduces the input feature dimension, making the dataset more suitable for efficient APM parameter estimation especially when a complete inflight trajectory is not available and accessible to DST prior to a flight.

III. Technical Approach

Physics-based models are crucial for predicting flight trajectories as they offer a comprehensive understanding of aircraft dynamics. These models utilize fundamental principles of physics to accurately describe the aircraft's motion and behavior. A simplified mathematical representation, known as the point mass model, is commonly employed in trajectory prediction applications in air traffic management.

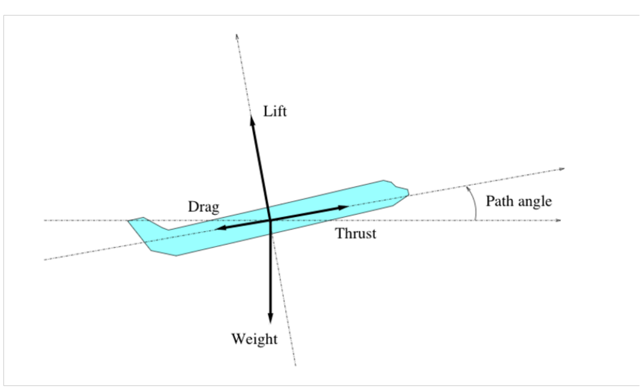

Fig. 1: Forces acting on an airborne aircraft.

The point mass model simplifies the aircraft's dynamics by treating it as a single point mass located at its center of gravity, disregarding complex aerodynamic and structural details. This model establishes a relationship between the inertial acceleration of the aircraft and the forces acting on its center of mass, such as thrust, drag, and gravity (Figure 1). Using the point mass model, the aircraft's trajectory can be predicted by solving a set of differential equations that consider the forces acting on the aircraft and the initial conditions such as mass, velocity, and position.

In conventional trajectory prediction applications that utilize the point-mass model, certain parameters related to thrust, drag, and weight are assumed to be readily available, often obtained from BADA tables. However, parameters for the remaining components of the model need to be estimated using the available data. By incorporating flight data, operational procedures, and other relevant data sources, the model can then predict trajectories. Nevertheless, the accuracy of these predictions can be compromised by the introduction of errors resulting from inaccurate parameter values. To address this issue, an enhanced approach is proposed in this study, aiming to improve the accuracy of trajectory prediction tools by learning previously unknown APM parameters. Parameters such as initial aircraft mass, thrust setting, and drag coefficients are estimated using historical flight data. This novel approach simultaneously estimates all coefficients.

The total energy model (TEM) establishes a relationship between the work exerted by the forces on an aircraft and the resulting change in the system's total mechanical energy. Equation (1) represents the TEM:

$$\frac{(T-D)V_t}{mg} = \dot{h} + \frac{V_t}{g}\left[\dot{V}_t + \frac{d}{dt}\big(V_w \cos(\psi_a - \psi_w)\big)\right] \quad (1)$$

In this equation, T refers to the thrust generated by the aircraft engines, D is the drag force acting on the aircraft, m is the aircraft's mass, V_t is the true airspeed of the aircraft, V_w is the wind speed, $\dot{h}$ is the vertical speed, g is the acceleration due to gravity, ψ_a and ψ_w are the aircraft heading and wind direction respectively. We assume a clean drag configuration, considering incompressible airflow and low Mach speed flight. We also assume that the effect of wind-related components and flight resistance arising from flaps or landing gear are negligible. Using standard thrust and clean drag models, T and D can be written as:

$$T = \delta T_{max} \text{ and } D = 0.5\rho V_t^2 S \underbrace{(C_{D0} + \kappa C_L^2)}_{C_D}$$

Here T_{max} is the maximum climb thrust, δ is the thrust setting which allows scaling the thrust force up or down, rho is the air density, S is the wing reference area, C_{D0}, and κ are the parasite and induced drag coefficients. By substituting T and D, we obtain the parameterized version of the TEM, as shown in equation (2).

$$\dot{h} = \frac{\delta}{(m_0 - m_f)}\Big[\frac{T_{max}V_t}{g.AF}\Big] - \frac{C_{D0}}{(m_0 - m_f)}\Big[\frac{\rho V_t^2 S}{2g.AF}\Big] - \kappa(m_0 - m_f)\Big[\frac{2g.cos^2(\gamma)}{S\rho V_t.AF}\Big] \tag{2}$$

$$\dot{m} = f \tag{3}$$

Here, AF is the acceleration factor, γ is the flight path angle, m_f is the cumulative weight of fuel consumed up to time t, and f is the fuel flow rate. The functional form of T_{max} and f depends on the phase of flight, altitude, mode, and other factors. The quantities highlighted in blue in equation (2) i.e., namely parasite drag coefficient (C_{D0}), induced drag coefficient (κ), starting mass of the aircraft (m_0), and thrust settings for climb and descent (δ_{des},δ_{cl}), are estimated by the ODE optimization process, and those in black are available using the historical flight data. A set of upper and lower bounds is provided to the optimizer, which includes constraints such as the minimum and maximum aircraft weight. These limits ensure that the estimated APM parameters remain within the acceptable range of operational conditions and constraints. Equations (2) and (3) together form an ODE system that, when combined with the corresponding initial and boundary conditions, enables the calculation of the aircraft's motion within each flight segment. Our approach for computing the APM parameters involves fitting the ODEs described by equations (2) and (3) to reconstruct the altitude profile of a historical flight record using our model with minimal error. By separating the known quantities, such as true airspeed, from the unknown parameters, we iteratively calculate the optimal model parameters that minimize the mean square of altitude error. For more detailed information about the ODE process and its specific parameters, the reader is referred to our previous papers on this topic [16], [17].

The data obtained from the ODE fitting process comprises pairs of historical flights and their corresponding optimized APM parameters. Since thrust setting values are not a parameter used in ground-based trajectory prediction tools and are mainly dictated by the pilot during the flight, we excluded them from the set of ML labels. Each historical flight is accompanied by its trajectory, along with additional data such as aircraft type, engine type, airline, city pair, and total flight distance. These pieces of information serve as features in the ML training process. The ML model's targets, or labels, are the APM parameters obtained from the ODE fitting process.

IV. Machine Learning Solutions

The ODE-fitting step is a complex process that is very time-consuming and is performed offline. Moreover, it can only be done on historical flights that have already landed and is, therefore, not useful for future flights. Thus, training an ML model to learn the relationship between the data features and ODE-generated labels is desired to make the APM parameters prediction faster and easier for future flight trajectories. After fitting the APM parameters to individual flight trajectories using historical data, we obtained the "labels" that serve as training data for a supervised ML model. Since the APM values that need to be predicted are numerical in nature, a regression ML model needs to be trained to accurately predict these values. In our case, the regression problem involves predicting values for C_{D0}, κ, and m_0. While many ML regression algorithms are designed to predict a single numerical value, certain algorithms, such as Linear Regression and Decision Trees, inherently support multi-output regression. In our APM parameter prediction, all three labels (CC_{D0}, κ, and m_0) were estimated simultaneously using ODE fitting. Therefore, all these labels are predicted together, as they are interdependent and influenced by both the input variables and each other. This type of problem is known as multiple-output regression. Three ML models including Linear Regression, Random Forest Regression (RFR), and XGBoost are selected to be trained and compared in terms of performance. The ML models capture the relationship between specific flight trajectory features and APM coefficients, enabling accurate trajectory prediction in future flights.

Our dataset is divided into two parts: 80% for training, and 20% for testing. To prevent overfitting, and fine-tune the ML models, a cross-validation technique was employed. A 5-fold cross-validation was applied, dividing the training data into five subsets. Through iterative training and evaluation on various combinations of training and validation sets, we optimized models' hyperparameters. Subsequently, the test set was employed to assess the final models' performance on previously unseen data. The trained ML models can provide the appropriate set of APM coefficients for any future flight, minimizing prediction errors and enhancing trajectory prediction accuracy. Figure 2 provides an overview of the APM estimation process, illustrating the progression from raw data to the ML testing results.

Our ML approach is divided into solutions: 1- using a subset of features that are known prior to the flight departure and do not change during flight (such as aircraft type, current temperature, destination airport, etc.) and 2- using a subset of in-flight features of the trajectory (such as average cruise altitude, Mach, and rate of climb, etc.) as well as all the pre-departure features from the first solution. The target variables are the APM parameters that were obtained by fitting them to a set of ODEs for each flight trajectory. The features in Solutions 1 and 2 are presented in Table I, accompanied by a brief description of each feature.

To evaluate the performance of our ML models, we employ two regression performance metrics: Mean Squared Error (MSE) and Mean Absolute Percentage Error (MAPE). These metrics serve as benchmarks to assess the accuracy and reliability of our models' predictions. MSE provides an overall measure of the model's performance by calculating the average squared difference between predicted and actual values. On the other hand, MAPE quantifies the percentage deviation between the predicted and actual values for each individual target variable.

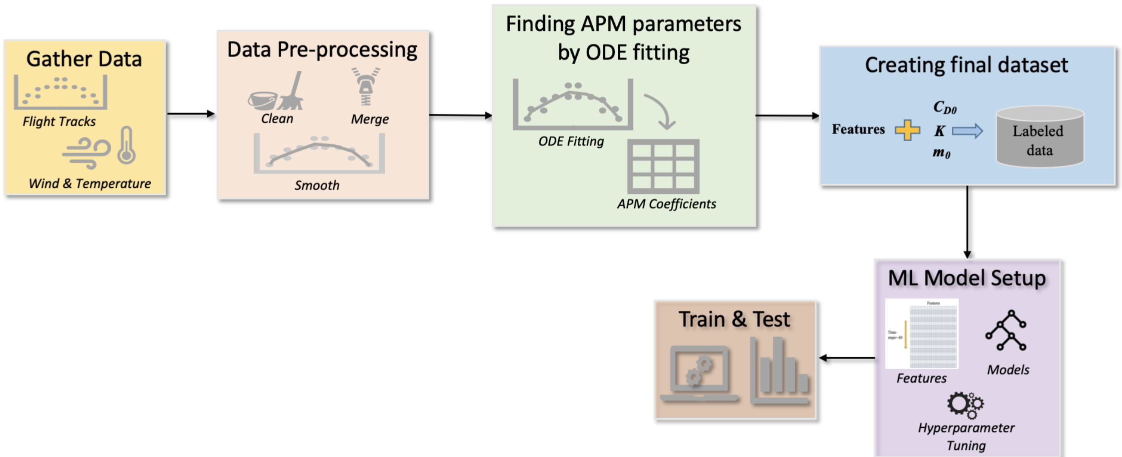

Fig. 2: The overview of the APM estimation process.

TABLE I: List of features in Solutions 1 & 2.

Feature	Solution 1	Solution 2	Description
Aircraft	✓	✓	Aircraft type
Airline	✓	✓	Name of airline
Distance	✓	✓	Total distance
Engine	✓	✓	Engine type
Month	✓	✓	Month of the year
Weekday	✓	✓	Day of the week
Time	✓	✓	Time of the day as hour
DEST	✓	✓	Destination
T (ORG)	✓	✓	Temperature at origin
P (ORG)	✓	✓	Pressure at origin
T (DEST)	✓	✓	Temperature at destination
P (DEST)	✓	✓	Pressure at destination
Mach (CRU)	✓	✓	avg. Mach at Cruise
Alt (CRU)	✓	✓	avg. altitude at Cruise
ROC (CLB)	–	✓	avg. ROC at Climb
CAS (CLB)	–	✓	avg. CAS at Climb
TAS (CLB)	–	✓	avg. TAS at Climb
Mach (CLB)	–	✓	avg. Mach at Climb
ROC (DSC)	–	✓	avg. ROC at Climb
CAS (DSC)	–	✓	avg. CAS at Climb
TAS (DSC)	–	✓	avg. TAS at Climb
Mach (DSC)	–	✓	avg. Mach at Climb

V. Results and Discussion

This section presents the results of the evaluation for three ML models: Linear Regression, Random Forest Regression, and XGBoost. The performance of these models was assessed using the Mean Squared Error (MSE) for the combined target variables C_{D0}, κ, and initial mass (m_0). Additionally, the individual performance scores for each label were reported as the Mean Absolute Percentage Error (MAPE). The results in Table II for Solution 1 show that the Random Forest regression model achieved the lowest MSE for the combined target variables, indicating its superior overall performance compared to Linear Regression and XGBoost. When evaluating the MAPE for each label, the XGBoost model demonstrated higher accuracy in predicting κ, while Linear Regression outperformed the other models in predicting C_{D0} and m_0. Notably, Random Forest performed closely behind the best-performing model across all three labels.

TABLE II: Comparison of ML performance scores for different models in Solution 1.

ML model	MSE	MAPE		
		C_{D0}	κ	initial mass
Linear Regression	0.025	3.03%	8.37%	5.3%
Random Forest	0.024	3.19%	6.21%	5.77%
XGBoost	0.031	3.22%	6.04%	7.0%

The results in Table III for Solution 2 show that Random Forest Regression maintained its overall superior performance compared to the other two models, showing improvement compared to Solution 1. For the combined target variables, Random Forest achieved the lowest MSE, indicating better overall performance. When examining the individual MAPE scores, Random Forest outperformed the other two models in predicting κ. XGBoost slightly outperformed the other models in predicting C_{D0}, while Linear Regression performed the best in predicting initial mass (m_0). These findings highlight the consistent and improved performance of Random Forest in solution 2, maintaining its superiority in the combined target variables' MSE.

TABLE III: Comparison of ML performance scores for different models in Solution 2.

ML model	MSE	MAPE		
		C_{D0}	κ	initial mass
Linear regression	0.025	3.03%	6.93%	4.67%
Random forest	0.021	3.01%	6.07%	4.89%
XGBoost	0.026	2.95%	6.13%	5.61%

Figure 3 shows the predicted and true labels of the three aircraft types in our dataset. The predicted values are outputs of the Random Forest model in Solution 2 (the best-performing set of features and model). The boxplots for the predicted labels demonstrate the effectiveness of the ML model in

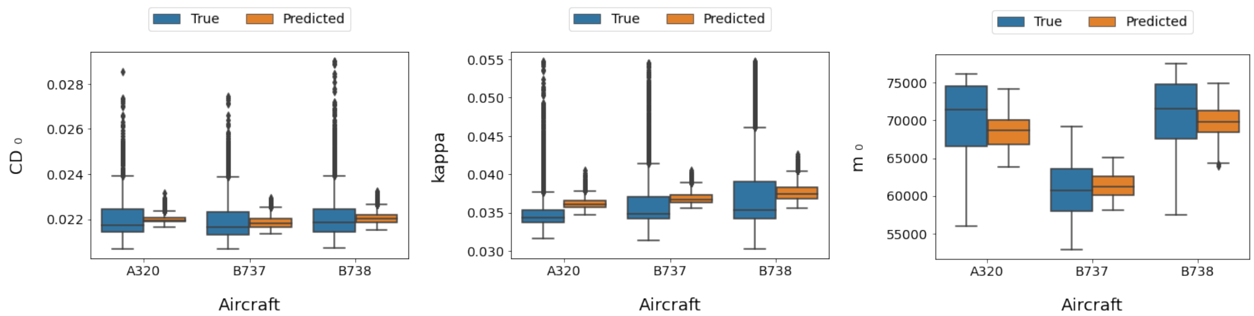

Fig. 3: Boxplots of the predicted and true labels by Random Forest in Solution 2.

accurately predicting the initial aircraft mass. The boxplots reveal a clear distinction between different aircraft types, indicating that the model performs well in capturing the variations specific to each type. The medians, box lengths, and whisker lengths vary significantly between the different types, implying distinct characteristics and behaviors specific to each type.

The C_{D0} boxplot shows that the median values of the predicted values are closely aligned with the true C_{D0} data. This suggests that the model's predictions capture the central tendency of the C_{D0} values accurately. When comparing C_{D0} across different aircraft types, the boxplot indicates that there is not a substantial amount of variability observed. The boxes in the boxplot are relatively narrow, indicating that the C_{D0} values for different aircraft types are clustered around similar ranges. However, the whiskers of the true values are larger compared to the predicted values. This indicates that the model might not fully capture the true variability present in the C_{D0} labels. The larger whiskers in the true boxplot suggest that there is more spread and variability in the actual C_{D0} values than what is represented by the model's predictions. This observation can be due to the nature of the modeling approach. Unlike the complex relationships and dynamics captured by the ODE process during label generation, the model is trained on simplified pre-flight and average in-flight values. As a result, the model may not fully capture the intricate variability inherent in the C_{D0} values. The larger whiskers in the truth data boxplots suggest a wider range of C_{D0} values that the model fails to capture accurately. The boxplots for the κ exhibit similar trends to C_{D0}. The predicted median values for κ align closely with the true data, indicating accurate predictions of the central tendency. However, when comparing κ across different aircraft types, more noticeable differences are evident compared to C_{D0} plots.

Based on the best-performing model, the Random Forest Regression, feature importance analysis was conducted for Solutions 1 and 2. This analysis aimed to identify the key factors that influence the predictions of the target variables, considering both preflight and in-flight information. Figure 4 displays the sorted feature importance in Solution 1, which primarily relies on preflight knowledge. Among the features, "Aircraft Type" emerged as the most significant, suggesting that the type of aircraft used has a strong impact on the predicted outcomes. The feature "Airline" has the second-highest importance. Other influential features include "Average Altitude during Cruise," "Temperature at Origin and Destination Airports," and "Average Mach during Cruise."

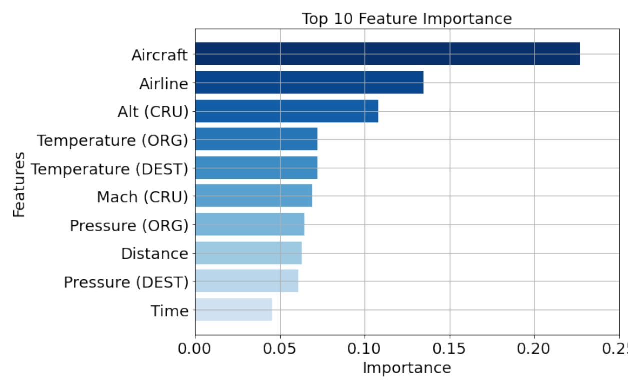

Fig. 4: Feature importance in Solution 1.

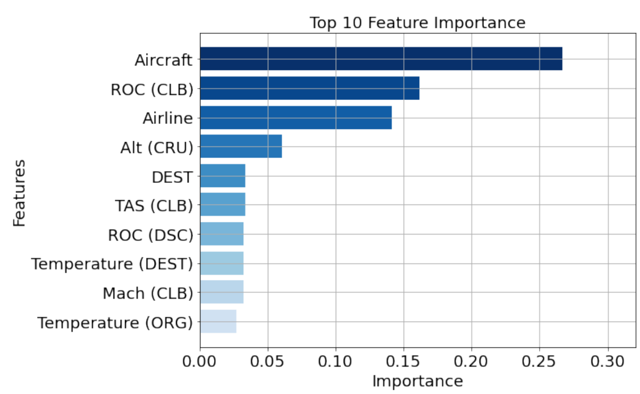

Fig. 5: Feature importance in Solution 2.

The feature importance for Solution 2, as illustrated in Figure 5, revealed the relative importance of various preflight and in-flight related features. Consistent with Solution 1, "Aircraft Type" is the most important feature. "Rate of Climb during Climb", "Rate of Climb during Descent", and "Airline Type" were also identified as important factors in predicting the target variables. Comparing Solution 2 to Solution 1, the introduction of ODE-related terms, such as Rate of Climb, TAS, and Mach during the Climb and Descent phases has replaced some of the important features from Solution 1 (such as temperature and pressure). The incorporation of ODE-related terms has provided valuable insights into the features, highlighting the relevance of in-flight dynamics in addition to preflight information. The inclusion of ODE terms has slightly improved the performance of all three ML models. This suggests that by leveraging preflight information, we can obtain estimations of the APM parameters that are sufficiently accurate.

The substantial influence of "Aircraft Type" on the ML predictions can be attributed to its direct relationship with the initial mass (m_0). By considering aircraft type as a prominent feature in the ML model, the inherent relationship between aircraft type and the initial mass is effectively captured, enabling precise predictions.

VI. Simulation Evaluation

Once the ML phase is completed, the ML-derived models are evaluated using NASA's Autonomy Development Toolkit (ADK) simulation software [18] (also known as AutoResolver [19], [20]). The ADK software encompasses various models such as airspace, airports, aircraft performance, wind, weather, and atmospheric conditions. Within the ADK, trajectory generators are available to simulate historical flights based on flight plans, aircraft performance, and atmospheric models. For more detailed information, readers are encouraged to refer to the appropriate resources. By simulating a specific historical flight, we can progressively predict the trajectory at predefined intervals, such as every 2 minutes. It is essential to note that the prediction process incorporates a 6-degree-of-freedom aircraft dynamics model, rendering it more intricate compared to the TEM employed in our ODE fitting approach. Predictions are made for a specified time horizon into the future, typically 1 hour.

To assess the performance of the ML-derived models, we conducted two sets of predictions for each flight. The first set utilized the baseline BADA APM parameters, while the second set employed the ML-derived APM parameters. With the exception of the derived coefficients (C_{D0}, κ, and m_0), all other model parameters and assumptions remained consistent across the two prediction sets. Figure 6 shows a flight from the A320 aircraft type during the climb phase. Two distinct predictions are generated: one utilizing the ML-modified APM (dashed green line), and the other employing the default BADA APM (dashed red line). The solid black line represents the actual historical flight that is simulated or played back in ADK. The dashed blue line indicates the current timestamp at which the prediction is made. The predictions are conducted at intervals of two minutes, and the trajectory is projected over a time span of 15 minutes into the future. Figure 6 presents snapshots of the predictions at four different time points during the climb phase. Closer proximity to the current timestamp corresponds to higher accuracy in the trajectory prediction. For this particular flight, employing the default BADA APM parameters results in a lower altitude compared to the actual flight. However, as the flight approaches the cruise altitude, the accuracy of the prediction improves. On the other hand, utilizing the ML-derived APM parameters enhances the overall accuracy of trajectory prediction. The ML model incorporates flight-specific information, allowing it to adapt the APM parameters to better align with the true trajectory. This improvement is a consequence of the ODE fitting method employed, where the model "learns" and adjusts the APM parameters accordingly. This simulation evaluation will be conducted on several flights to confirm the improvements afforded by the ML modeling process.

VII. Conclusion and Future steps

In this study, we applied ML models to predict key APM parameters of initial flight mass (m_0), C_{D0}, and κ. The key challenge in this setup was the absence of ground truth data for APM parameters. To address this challenge, we devised a novel ODE approach to fit the aerodynamic models and simultaneously estimate APM parameters for each flight in our dataset. Due to the substantial computational requirements of the ODE fitting approach, its real-time application for estimating the APM parameters in every individual flight becomes impractical for the DST. Additionally, the unavailability of the complete time-series data before the flight necessitated the development of an alternative solution utilizing preflight and simplified features to estimate the APM parameters. To achieve this, ML models were trained to estimate these parameters, exploring two distinct solutions: one utilizing preflight features and the other incorporating various inflight data features. The evaluation of the trained ML models demonstrated that the Random Forest model exhibited the best performance among the tested approaches.

The results from the feature importance showed that the "Aircraft type" has the highest importance in accurate label predictions in both solutions due to its strong correlation with the initial aircraft mass. It was also shown that adding a few extra inflight features such as average values of TAS, CAS, and Mach during the climb and descent phases slightly improved the ML models' performances in Solution 2. Despite the superior performance of Solution 2, which incorporated inflight data, Solution 1 holds promise as a viable means of estimating APM parameters solely utilizing preflight information. Another observation was the ML models' inability to capture the complex relationship between the features and labels for predicting C_{D0} and κ. The comparison between the predicted and true labels in Solution 2 revealed that the true data exhibited a broader range of variability across all aircraft types, while the predicted values displayed narrower ranges. This limitation can be attributed to the models' reliance on a limited set of pre-flight and average in-flight features, which may not fully capture the intricate dynamics and complexities associated with the entire time series.

Moving forward, we will broaden our dataset to encompass a wider range of aircraft types and destination airports, allowing for a more comprehensive analysis and evaluation of our

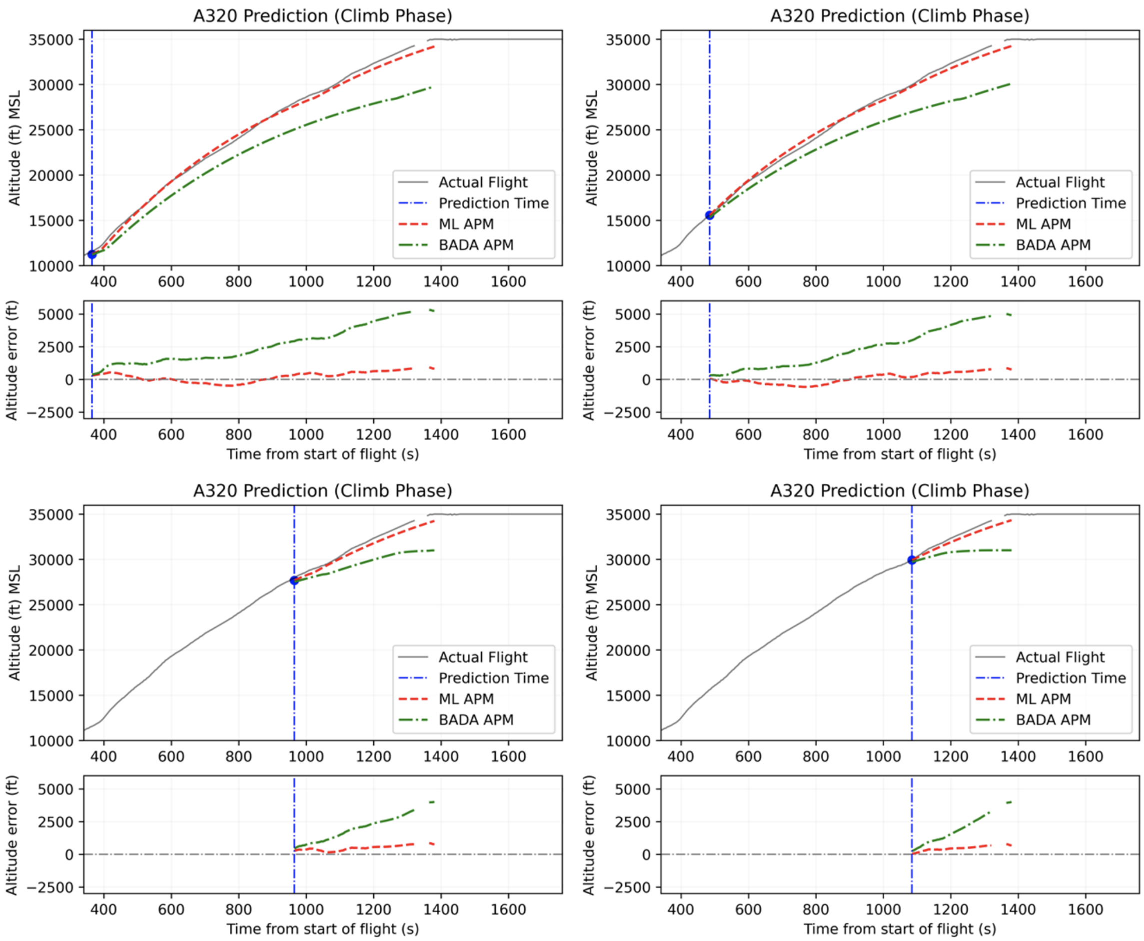

Fig. 6: Trajectory prediction using BADA and ML-derived APM for example flight.

models' performance. Moreover, we will focus on replacing the ODE fitting process, which can take up to 10 minutes for predicting the APM parameters for each flight, with a deep-learning model that leverages the entire time series of flight data. This approach holds the potential to expedite the label generation stage and further enhance the accuracy and efficiency of estimating APM parameters. Finally, the ML-derived APM parameters will be further assessed and validated by comparing them against the reference BADA values using flight simulation software.

References

[1] A. Harada, T. Kozuka, A. Miyazawa, N. K. Wickramasinghe, and M. Brown, "Analysis of air traffic efficiency using dynamic programming trajectory optimization," in *29th Congress of the International Council of the Aeronautical Sciences*, 2014.

[2] G. Senzig D. A. Fleming and R. J. Iovinelli, "Fuel consumption modeling in support of atm environmental decision-making," in *Eighth USA/Europe Air Traffic Management Research and Development Seminar*, 2009.

[3] M. R. Jackson, Y. J. Zhao, and R. A. Slattery, "Sensitivity of trajectory prediction in air traffic management," *Journal of Guidance, Control, and Dynamics*, vol. 22, no. 2, pp. 219–228, 1999.

[4] S. Mondoloni and N. Rozen, "Aircraft trajectory prediction and synchronization for air traffic management applications," *Progress in aerospace sciences*, vol. 119, no. 100640, 2020.

[5] W. Chan, R. Bach, and J. Walton, "Improving and validating ctas performance models," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2000, p. 4476.

[6] C. Gong and W. Chan, "Using flight manual data to derive aero-propulsive models for predicting aircraft trajectories," in *AIAA's Aircraft Technology, Integration, and Operations (ATIO) 2002 Technical Forum*, 2002, p. 5844.

[7] C. Schultz, D. Thipphavong, and H. Erzberger, "Adaptive trajectory prediction algorithm for climbing flights," in *AIAA Guidance, Navigation, and Control Conference*, no. 2012-4931, 2012.

[8] D. P. Thipphavong, C. A. Schultz, A. G. Lee, and S. H. Chan, "Adaptive algorithm to improve trajectory prediction accuracy of climbing aircraft," *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 1, pp. 15–24, 2013.

[9] D. P. Thipphavong, "Top-of-climb matching method for reducing aircraft trajectory prediction errors," *Journal of aircraft*, vol. 53, no. 5, pp. 1211–1223, 2016.

[10] D. Thipphavong, "Analysis of climb trajectory modeling for separation assurance automation," in *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2008, p. 7407.

[11] R. Dalmau, X. Prats, A. Ramonjoan, and S. Soley, "Estimating fuel consumption from radar tracks: a validation exercise using FDR and radar tracks from descent trajectories," *CEAS Aeronautical Journal*, vol. 11, no. 2, pp. 355–365, 2020.

[12] R. Alligier, D. Gianazza, and N. Durand, "Energy rate prediction using

an equivalent thrust setting profile," in *ICRAT 2012, 5th International Conference on Research in Air Transportation*, 2012.

[13] R. Alligier, D. Gianazza, and N. Durand, "Learning the aircraft mass and thrust to improve the ground-based trajectory prediction of climbing flights," *Transportation Research Part C: Emerging Technologies*, vol. 36, pp. 45–60, 2013.

[14] R. Alligier, D. Gianazza, , and N. Durand, "Machine learning and mass estimation methods for ground-based aircraft climb prediction," *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS*, vol. 16, pp. 3138–3149, 2015.

[15] J. Sun, J. M. Hoekstra, and J. Ellerbroek, "Estimating aircraft drag polar using open flight surveillance data and a stochastic total energy model," *Transportation Research Part C: Emerging Technologies*, vol. 114, pp. 391–404, 2020.

[16] T. Puranik, A. Sharif Rohani, and K. Kalyanam, "An ode-fitting approach to estimate critical aircraft performance parameters for trajectory prediction," in *IEEE/AIAA 41st Digital Avionics Systems Conference (DASC)*, 2022.

[17] A. Fernandes, D. Wesely, T. G. Puranik, A. Sharif Rohani, K. M. Kalyanam, and D. Morin, "Prediction of critical aircraft performance model parameters from historical flight data," in *AIAA SCITECH 2023 Forum*, 2023, p. 2532.

[18] R. D. Windhorst, S. Zelinksi, T. A. Lauderdale, A. V. Sadovsky, Y.-C. Chu, J. Phillips, Y. Zheng, T. Nguyen, and J. D'Amore, "Initial validation of a simulation system for studying interoperability in future air traffic management systems," in *AIAA AVIATION 2021 FORUM*, 2021, p. 2352.

[19] H. Erzberger, "Automated conflict resolution for air traffic control," 2005.

[20] R. D. Windhorst, T. A. Lauderdale, R. A. Coppenbarger, and H. Erzberger, "Validation of an automated system for arrival traffic management," in *AIAA AVIATION 2022 Forum*, 2022, p. 3828.