Project 054 AEDT Evaluation and Development Support

Georgia Institute of Technology

Project Lead Investigator

Principal Investigator:
Professor Dimitri N. Mavris
Director, Aerospace Systems Design Laboratory
School of Aerospace Engineering
Georgia Institute of Technology
Phone: (404) 894-1557
Fax: (404) 894-6596
Email: dimitri.mavris@ae.gatech.edu

Co-Principal Investigator:
Dr. Michelle Kirby
Chief, Civil Aviation Division
Aerospace Systems Design Laboratory
School of Aerospace Engineering
Georgia Institute of Technology
Phone: (404) 385-2780
Fax: (404) 894-6596
Email: michelle.kirby@ae.gatech.edu

University Participants

Georgia Institute of Technology

- PIs: Dr. Dimitri Mavris, Dr. Michelle Kirby
- FAA Award Number: 13-C-AJFE-GIT-054
- Period of Performance: February 5, 2020 to February 4, 2021
- Tasks:
    1. Improved Departure Modeling (divided into four areas: High Altitude Airport Study; Refinement of Thrust and Weight Assumptions; Estimation of Thrust Using ANP Equations; and Comparison of NADP Profiles to Real-world Operations.
    2. Arrival Profile Modeling.
    3. Full Flight Modeling.
    4. System Testing and Evaluation of AEDT.

Project Funding Level

The project is funded at the following levels: GT ($700,000). In terms of cost-share details, GT has agreed to a total of $700,000 in matching funds. This total includes salaries for the project director, research engineers, and graduate research assistants, as well as computing, financial, and administrative support, including meeting arrangements. GT has also agreed to provide tuition remission for the students, paid for by state funds.

Investigation Team

Prof. Dimitri Mavris, Principal Investigator, Georgia Institute of Technology
Dr. Michelle Kirby, Co-Investigator, Georgia Institute of Technology
Dr. Yongchang Li, Research Faculty, Georgia Institute of Technology
Dr. Tejas Puranik, Research Faculty, Georgia Institute of Technology
Dr. Mohammed Hassan, Research Faculty, Georgia Institute of Technology
Dr. Holger Pfaender, Research Faculty, Georgia Institute of Technology

Mr. David Anvid, Research Faculty, Georgia Institute of Technology
Ameya Behere, Graduate Student, Georgia Institute of Technology
Eleni Sotiropoulos-Georgiopoulos, Graduate Student, Georgia Institute of Technology
Ayaka Miyamoto, Graduate Student, Georgia Institute of Technology
Rukmini Roy, Graduate Student, Georgia Institute of Technology
Jirat Bhanpato, Graduate Student, Georgia Institute of Technology
Hyungu Choi, Graduate Student, Georgia Institute of Technology
Bogdan Dorca, Graduate Student, Georgia Institute of Technology
Zhenyu Gao, Graduate Student, Georgia Institute of Technology
Santusht Sairam, Graduate Student, Georgia Institute of Technology

Project Overview

This project is providing data and methods to continue to improve the aircraft weight, takeoff thrust, and departure and arrival procedure modeling capabilities within the FAA's Aviation Environmental Design Tool (AEDT). Some of the modeling assumptions in AEDT are considered overly conservative and could be improved using industry and airport flight operational data. This funding would continue to support the implementation of these methods and data into AEDT4. To facilitate this, the Georgia Tech team will utilize real-world data flight and noise monitoring data to improve departure, full flight, and arrival modeling. In addition, this research will provide the FAA Environment and Energy office with evaluations and assessments of AEDT's future service pack releases.

Task 1 – Improved Departure Modeling: High Altitude Airport Study

Georgia Institute of Technology

Objectives

The conclusion of ASCENT Project 45 provided recommendations for noise abatement departure procedures (NADP) to be modeled in future versions of AEDT. While these procedures had been validated by comparison to real-world data at U.S. airports, it was not evident whether the recommended procedures would adequately represent operations at high altitude edge case airports. Project 54 aims to validate the recommended NADP profiles at very high-altitude airports

Research Approach

Introduction

Very high-altitude airports present an edge case in terms of aircraft performance due to low ambient atmospheric temperatures, densities, and pressures. Such airports often require special considerations for operations such as limitations on max takeoff weight, use of modified flap, and reduced thrust settings. Hence, it is not evident whether the newly recommended NADP profiles would correctly represent real-world operations out of such airports. This task aims to verify whether the recommendations hold true, and what steps (if any) need to be taken to address any potential shortcomings.

Methodology

The validation dataset for this Task is Flight Operations Quality Assurance (FOQA) obtained from Georgia Tech's airline partner. Subsequent considerations of airport and aircraft selection was based on the availability of associated data in the FOQA dataset. Based on an analysis of the metadata of all FOQA flights, the following high-altitude airports were selected for modeling in this Task:

Table 1. High altitude airports considered in this work

Airport Code	Location	Elevation, Mean Sea Level (MSL)
UIO/SEQM	Quito, Ecuador	7830 ft
MEX/MMMX	Mexico City, Mexico	7316 ft
JNB/JAOR	Johannesburg, South Africa	5558 ft
DEN/KDEN	Denver, Colorado	5433 ft

In this report, results are presented for 737-700 aircraft operations at MMMX airport. FOQA data contains information about the aircraft's trajectory, thrust, weight, and numerous other characteristics. This information was compared against performance reports obtained from AEDT. AEDT simulations were set up to fly two NADP profile sets, NADP1_1 and NADP2_11. Each profile set contained the baseline profile definition along with alternate weight and reduced thrust versions. Detailed descriptions on the profile definitions are available in the 2019 ASCENT 45 Annual Report. Additionally, the default STANDARD profile set in AEDT was also modeled to serve as the baseline.

Results and Discussion

This section shows some of the preliminary results obtained in this research area. Figure 1, Figure 2, and Figure 3 show the comparison of trajectories, ground speed, and net corrected thrust, respectively, from AEDT profiles against the FOQA. These comparisons provide a visual comparison and a first assessment of the efficacy of NADP profiles at representing high altitude airport departure operations. Note that all FOQA flights are shown as solid lines in black color in all three figures.

The trajectory plot shows that the reduced thrust versions of profiles are a better representation of real-world flights than full thrust flights. Full thrust profile trajectories tend to be higher than the FOQA trajectories, due to the use of higher thrust levels. In the groundspeed plot, a wide variation is observed in the FOQA flights. The speed at liftoff for AEDT profiles tends to be lower, and the terminal speed at 10,000 ft above field elevation (AFE_ altitude tends to be higher. A possible reason for this is the implication of wind speed in the calculation of groundspeed. AEDT assumes a constant headwind applied for all departure operations, whereas FOQA flights reflect actual wind conditions on the day of the flight, which leads to higher variability. In future analyses, true or calibrated airspeed will be used instead, which should eliminate this source of difference. Finally, the net corrected thrust comparisons also show better agreement with the reduced thrust profile versions at lower altitudes. Real-world thrust tends to be higher than AEDT in the latter half of the departure operation.

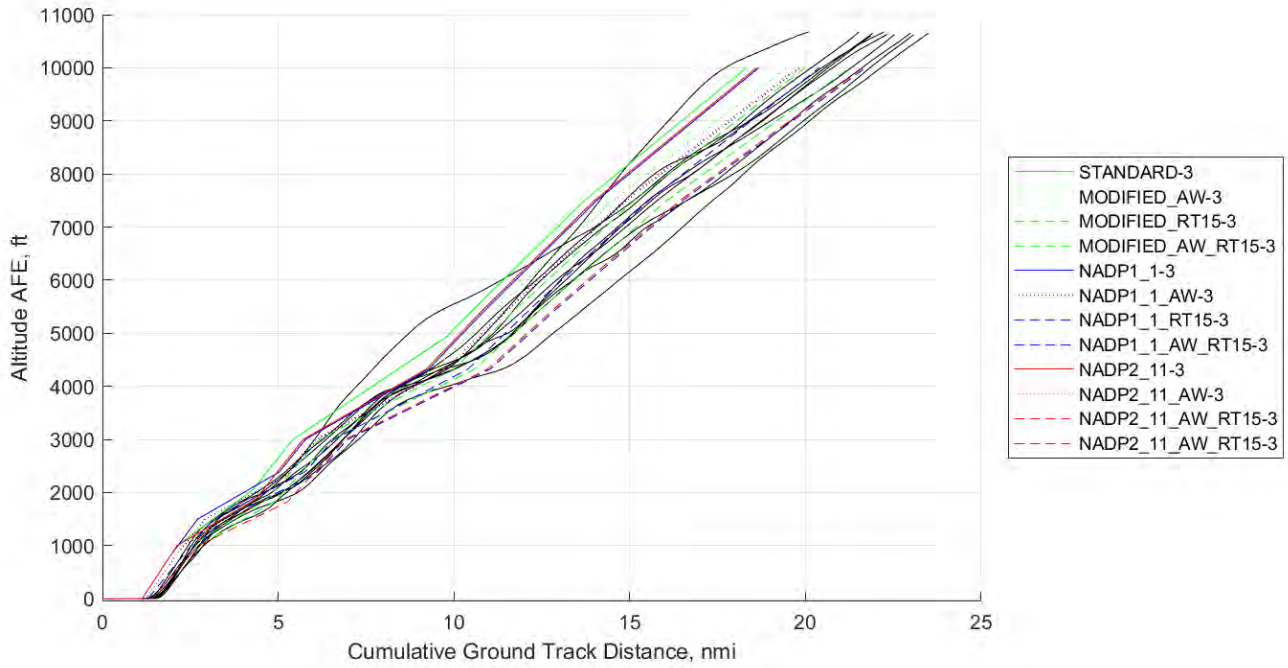


Figure 1. Flight trajectories from AEDT profile sets compared with FOQA flight data.

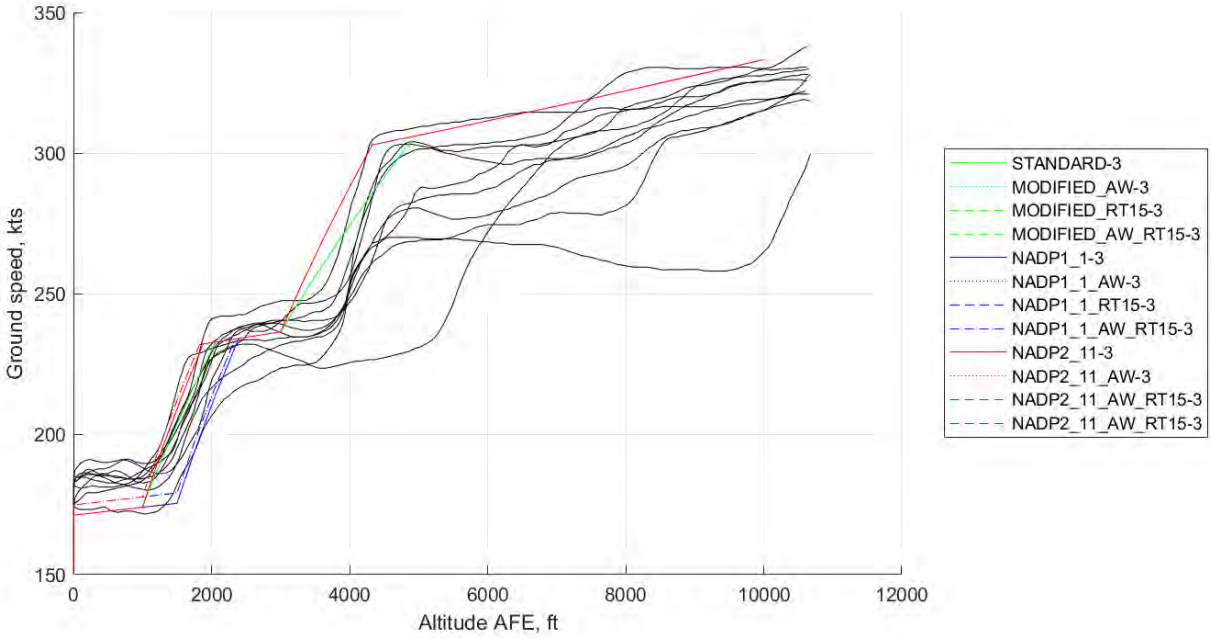


Figure 2. Groundspeed variation from AEDT profile sets compared with FOQA flight data.

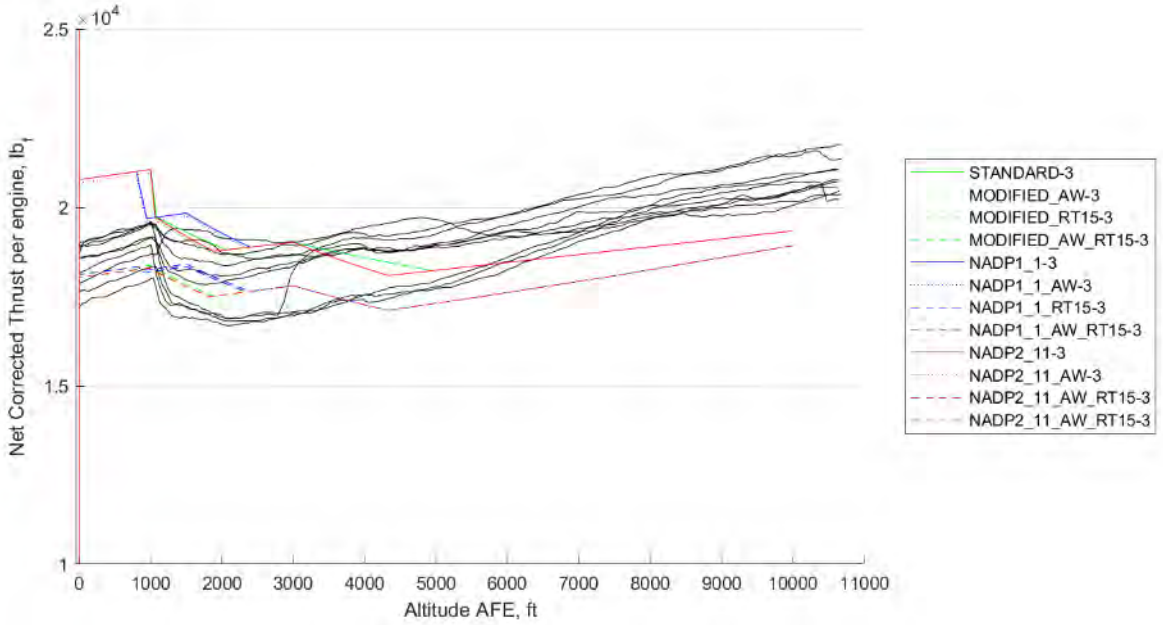


Figure 3. Thrust variation from AEDT profile sets compared with FOQA flight data.

Overall, initial comparative analysis shows that the reduced thrust variants of AEDT profile sets seem to reasonably represent real-world operations at high altitude airports. Additional analyses at other airports and with other aircraft types will be conducted.

Task 1 – Improved Departure Modeling: Refinement of Thrust and Weight Assumptions

Georgia Institute of Technology

Objectives

The objective of this Task is to use the new FOQA data to analyze previous takeoff thrust and weight assumptions and compare the results to previous years' weight model and thrust reduction models.

Research Approach

Weight Assumptions

Introduction

In this step, various linear, multilinear, and quadratic regression models have been used to determine relations between aircraft takeoff weights and Great Circle Distance (GCD), airport elevation, and runway length for each airframe of the FOQA data.

Methodology

First, it was necessary to clean and gather the takeoff weight, takeoff runway length, runway elevation, and GCD for each flight. According to the results from the ASCENT Project 35 team, it has been decided to neglect the presence of "tankered fuel". Additionally, all flights with a GCD lower than 50 nmi were not considered since they correspond to repositioning or test flights. Finally, recording errors such as null or blank values were removed.

For each airframe, the following regression models were applied first to all data and later to the data averaged for each city-pair:

- TOW=A0*GCD
- TOW=A1*GCD+A2*RXYlength+A3*RWYelev+A4
- TOW=A5*GCD²+A6*GCD+A7

Results and Discussion

Some of the plots obtained are presented in Figure 4. It is impossible to plot the results of multilinear regressions in 2D so only linear (in black) and quadratic (in red) regressions were plotted. On the non-averaged data, clusters of points can be observed. They correspond to data for the same city-pair, since those clusters are not present on the averaged plots.

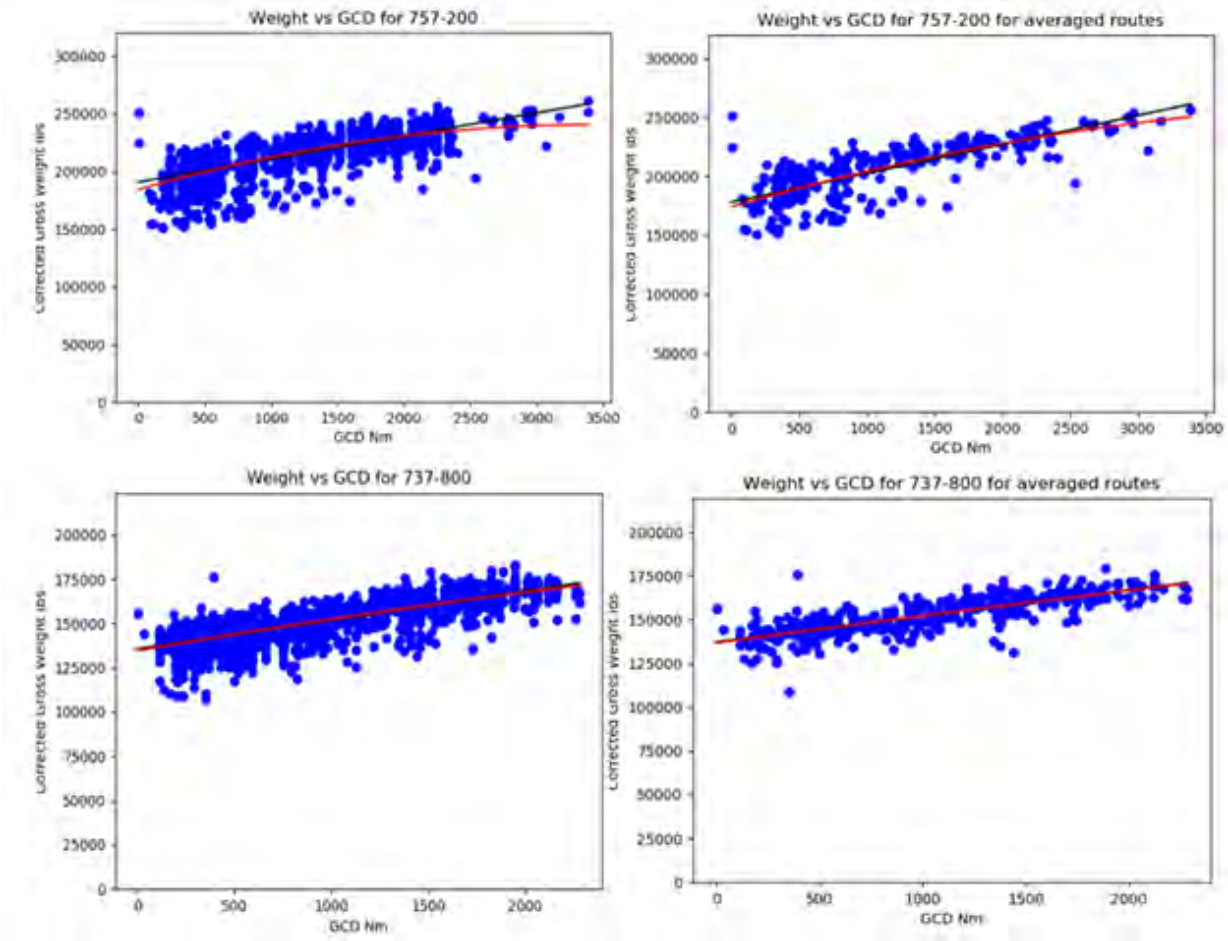


Figure 4. Weight versus GCD for 757-200 and 737-800.

The R^2 and regression coefficients for each airframe and model are presented in the following table. The model that provides the best R^2 for each airframe is highlighted in green. For those airframes that present low R^2 values, two causes have been identified. First, some airframes have a small set of data points especially once averaged by city-pair. In addition, for short-range flights, there is more variability in the flight path since the departure and arrival procedures account for a larger share of the flight duration. The GCD does not take this variability into account.

Table 2. Regression results for all airframes

Airframe	Number of data points	Averaged by city-pair ?	Model	R^2	Reg. Coeff. Weight	Reg. Coeff. elevation	Reg. Coeff. length	Reg. Coeff. quadratic Weight	Intercept
	1992	No	Linear	0.300	12.289				98828.255
	1992	No	Quadratic	0.302	16.959			-0.004	97827.283
717-200	304	Yes	Linear	0.320	11.088				99084.854
	304	Yes	Quadratic	0.355	26.178			-0.013	95738.690
	1992	No	Multi-	0.303	12.440	0.169	0.052		98027.002
	304	Yes	Multi-	0.321	11.061	-0.128	0.084		98367.382
	114	No	Linear	0.502	10.685				123545.482
	114	No	Quadratic	0.507	15.221			-0.003	122361.263
737-700	36	Yes	Linear	0.582	11.718				122579.044
	36	Yes	Quadratic	0.589	17.727			-0.004	120919.474
	114	No	Multi-	0.513	11.683	-0.394	0.098		122601.269
	36	Yes	Multi-	0.605	12.247	-0.295	0.500		117732.292
	1926	No	Linear	0.617	16.373				135604.861
	1926	No	Quadratic	0.618	18.993			-0.001	134652.489
737-800	389	Yes	Linear	0.627	14.864				137201.305
	389	Yes	Quadratic	0.627	16.027			-0.001	136742.694
	1926	No	Multi-	0.617	16.332	-0.017	0.162		133917.782
	389	Yes	Multi-	0.629	14.701	-0.317	0.189		135679.033
	4844	No	Linear	0.624	16.807				143685.984
	4844	No	Quadratic	0.631	25.170			-0.004	140155.187
737-900	388	Yes	Linear	0.647	15.365				143919.400
	388	Yes	Quadratic	0.647	16.234			0.000	143583.782
	4844	No	Multi-	0.629	16.316	-0.695	0.290		141795.918
	388	Yes	Multi-	0.658	15.058	-0.972	0.100		144072.997
	1917	No	Linear	0.551	20.026				190667.481
	1917	No	Quadratic	0.565	32.645			-0.005	184338.703
757-200	279	Yes	Linear	0.594	24.507				178122.597
	279	Yes	Quadratic	0.600	33.219			-0.003	174091.182
	1917	No	Multi-	0.556	20.087	0.868	0.503		184783.691
	279	Yes	Multi-	0.606	24.229	0.338	1.503		162850.011
	478	No	Linear	0.496	21.857				208133.502
	478	No	Quadratic	0.513	40.764			-0.007	196485.663
757-300	65	Yes	Linear	0.743	22.041				205428.409
	65	Yes	Quadratic	0.754	33.167			-0.004	199346.141
	478	No	Multi-	0.523	22.644	4.016	-0.524		209435.976
	65	Yes	Multi-	0.759	22.782	2.674	-0.644		209642.511
	414	No	Linear	0.946	36.796				447798.275
	414	No	Quadratic	0.947	45.895			-0.001	435869.424
777-200LR	47	Yes	Linear	0.831	41.091				431553.113
	47	Yes	Quadratic	0.835	52.819			-0.002	419400.774
	414	No	Multi-	0.947	38.713	-1.133	-2.231		470286.019
	47	Yes	Multi-	0.833	42.183	-2.223	-1.422		447037.943

Airframe	Number of data Points	Averaged by city-pair ?	Model	R^2	Reg. Coeff. Weight	Reg. Coeff. elevation	Reg. Coeff. length	Reg. Coeff. quadratic Weight	Intercept
	466	No	Linear	0.821	39.582				408355.484
	466	No	Quadratic	0.823	47.728			-0.001	396925.470
777-200ER	56	Yes	Linear	0.798	42.299				399177.803
	56	Yes	Quadratic	0.806	59.856			-0.003	381719.896
	466	No	Multi-Linear	0.824	36.967	-14.003	1.453		404666.877
	56	Yes	Multi-Linear	0.798	42.138	-0.821	0.101		398848.892
	737	No	Linear	0.453	14.016				126774.658
	737	No	Quadratic	0.465	22.795			-0.005	123897.936
A319-100	234	Yes	Linear	0.486	13.214				127165.200
	234	Yes	Quadratic	0.507	22.902			-0.006	124012.509
	737	No	Multi-Linear	0.458	13.574	-0.324	0.067		126872.635
	234	Yes	Multi-Linear	0.489	13.124	0.047	0.227		124873.896
	957	No	Linear	0.476	18.216				133517.487
	957	No	Quadratic	0.482	26.469			-0.005	130459.344
A320-200	257	Yes	Linear	0.543	17.657				133157.523
	257	Yes	Quadratic	0.544	20.887			-0.002	132082.854
	957	No	Multi-Linear	0.488	17.509	-0.826	0.281		132111.273
	257	Yes	Multi-Linear	0.576	16.966	-1.398	0.496		130056.908
	325	No	Linear	0.584	17.724				157855.198
	325	No	Quadratic	0.584	17.002			0.000	158128.635
A321-200	92	Yes	Linear	0.600	16.376				158043.767
	92	Yes	Quadratic	0.604	9.394			0.004	160688.378
	325	No	Multi-Linear	0.584	17.666	-0.194	0.033		157747.549
	92	Yes	Multi-Linear	0.611	16.004	-0.846	0.471		154241.818
	491	No	Linear	0.794	31.086				341722.638
	491	No	Quadratic	0.812	49.715			-0.003	313121.055
A330-200	50	Yes	Linear	0.773	33.933				332352.394
	50	Yes	Quadratic	0.781	46.419			-0.002	320536.934
	491	No	Multi-Linear	0.796	30.539	-1.227	1.563		324727.594
	50	Yes	Multi-Linear	0.794	33.226	6.560	5.220		269574.853
	825	No	Linear	0.730	33.263				365027.300
	825	No	Quadratic	0.736	47.597			-0.003	349744.170
A330-300	98	Yes	Linear	0.641	34.518				353201.106
	98	Yes	Quadratic	0.642	28.616			0.001	357686.442
	825	No	Multi-Linear	0.733	32.734	0.386	1.818		345171.378
	98	Yes	Multi-Linear	0.662	32.314	-3.856	6.163		289707.955
	1860	No	Linear	0.466	20.927				132849.562
	1860	No	Quadratic	0.468	26.751			-0.004	131076.817
MD-90	227	Yes	Linear	0.518	19.644				133280.083
	227	Yes	Quadratic	0.518	22.243			-0.002	132593.858
	1860	No	Multi-Linear	0.479	21.240	-1.181	0.573		128177.782
	227	Yes	Multi-Linear	0.542	20.001	-1.239	0.569		128638.373

Table 3. Comparison with ASCENT 35 results for B757-200 and B737-800

Airframe			Number of flights	R^2	Reg. Coeff. Weight	Reg. Coeff. elevation	Reg. Coeff. length	Intercept
757-200	Ascent 035 Report	All flights	45343	0.662				
		Average per city-pair	376	0.828	19.188	0	0.364	175571.496
	FOQA Data	All flights	1917	0.556	20.087	0.868	0.503	184783.691
		Average per city-pair	279	0.606	24.229	0.338	1.503	162850.011
Difference (average by city-pair model)				0.222	-5.041	-0.338	-1.139	12721.485
% Difference (average by city-pair model)				-36.6%	20.8%	100.0%	75.8%	-7.8%
737-800	Ascent 035 Report	All flights	33933	0.569				
		Average per city-pair	467	0.812	14.88	-0.094	0.625	128007.473
	FOQA Data	All flights	1926	0.617	16.332	-0.017	0.162	133917.782
		Average per city-pair	389	0.629	14.701	-0.317	0.189	135679.033
Difference (average by city-pair model)				0.183	0.179	0.223	0.436	-7671.56
% Difference (average by city-pair model)				-29.1%	-1.2%	70.3%	-230.7%	5.7%

Compared to previous years' results, the FOQA data provides a significantly lower number of data points. The order of magnitude for each regression coefficient stay the same.

Thrust assumptions

Introduction
In this step, takeoff thrust data were used to determine reduced thrust distribution for each airframe.

Methodology
First, the data were cleaned by removing data recording errors, missing elements, and flight with a GCD lower than 20 nmi since they correspond to maintenance or repositioning flights. To limit computation time, it has been decided to only consider thrust values during takeoff roll when the speed of the aircraft is between 80 kts and 110 kts. The maximum calculated net corrected average thrust value, which is the thrust corrected for pressure difference, in this segment has been used. For each airframe, a histogram of takeoff thrusts has been plotted. The goal is then to obtain the reduced thrust percentage by dividing each thrust value by the maximum takeoff thrust of each airframe. However, this requires some tuning that has not been accomplished yet.

Results and Discussion
Histograms appear in the following Figures 5 and 6. It is important to check the total number of flights for each airframe before looking at thrust histograms. City-pairs and date and time can also create visible patterns on the thrust distributions.

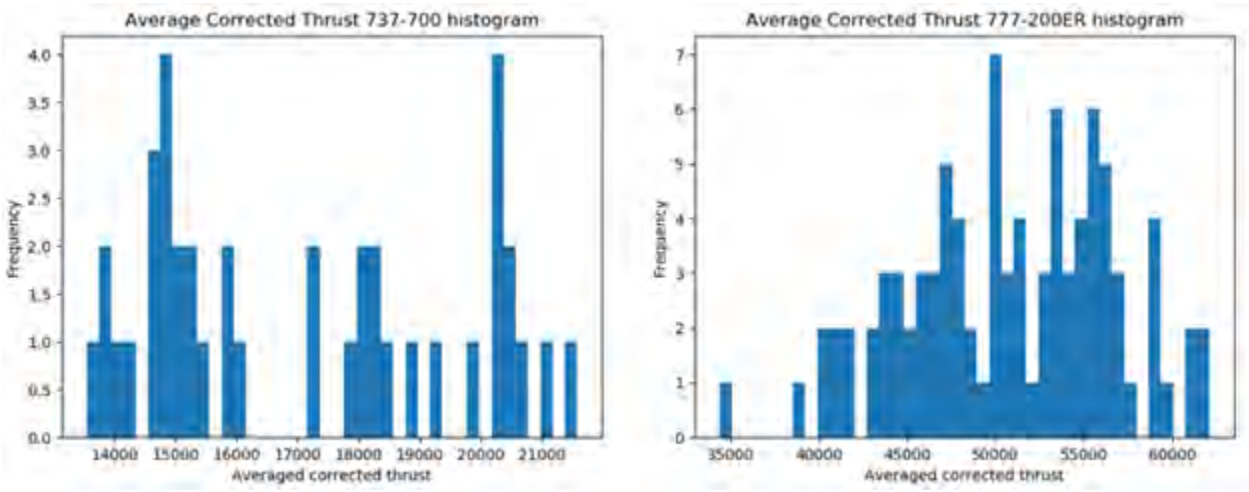


Figure 5. Thrust distributions for B737-700 and B777-200ER.

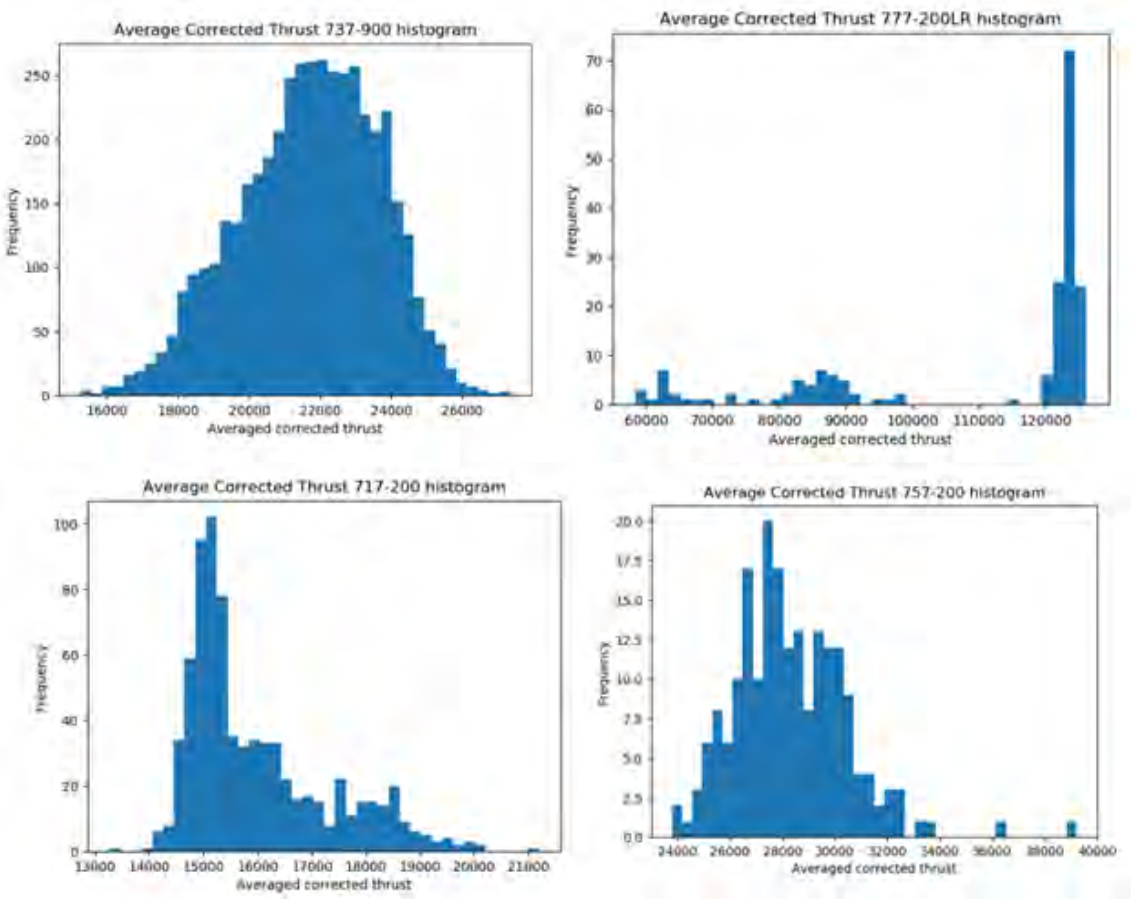


Figure 6. Thrust distributions for B737-900, B777-200LR, B717-200 and B757-200.

For the 777-200LR, we can observe a thrust distribution that is coherent with a long-range aircraft. Indeed, these aircraft almost always takeoff with the maximum fuel possible, which means maximum weight and thus maximum thrust. In the case of the 717-200, we observe the opposite trend. Since this aircraft is used for short-range flights, the maximum quantity of fuel is not required, therefore the maximum thrust at takeoff is not necessary. The 757-200 is another interesting case to study. Indeed, it is overpowered since it uses derated 767 engines. As a result, it always takes off at a reduced thrust. It also provides a comparison with previous years' results since this airframe was studied by ASCENT Project 35 in 2016, shown in Figure 7.

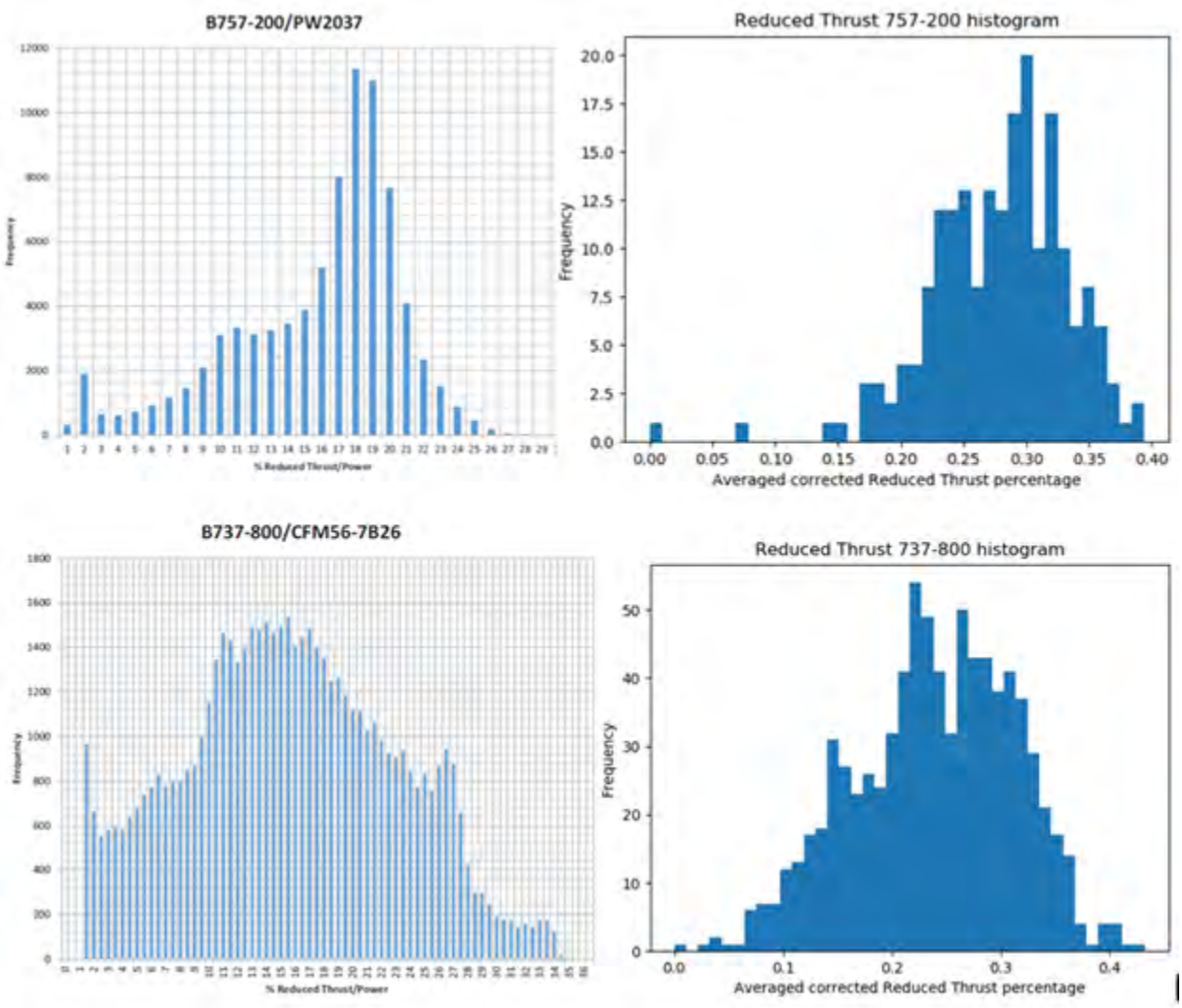


Figure 7. Comparison between ASCENT 35 and FOQA data thrust reduction results.

A comparison with previous year's results validates the shape of our thrust reduction distribution. Therefore, despite the small number of FOQA data points available, the observations are still consistent.

Task 1 – Improved Departure Modeling: Estimation of Thrust Using ANP Equations

Georgia Institute of Technology

Objectives

This research aims to develop regressions from FOQA data to obtain coefficients sets for the aircraft noise and performance (ANP) thrust equation. With these sets of coefficients, thrust can be estimated for flights based on radar/ADS-B tracking sources. This effort is directed at enabling estimation of thrust for threaded track or radar flights for which detailed information might not be available to model a fixed-point profile in AEDT.

Research Approach

Introduction

The ANP thrust equations are used by AEDT as part of its performance model. The equation assumes that the net corrected thrust depends on a constant term, a linear dependency on airspeed and temperature, and a quadratic dependency on altitude. The equation is:

$$\frac{F_n}{\delta} = E + Fv + G_A h + G_B h^2 + HT$$

Where

$\frac{F_n}{\delta}$ is the net corrected thrust per engine (lbf)
v is the equivalent/calibrated airspeed (kt)
h is the pressure altitude MSL (ft)
T is the temperature at the aircraft (˚C)
E, F, G_A, G_B, H are regression coefficients that depend on power state and temperature state

Typically, the regression coefficients are available in a database and can be used to compute the thrust value at various points in the aircraft trajectory. Here, thrust data is available in the FOQA data, and hence the regression coefficients can be estimated using a simple linear regression technique.

Methodology

The overall methodology for this task follows these steps:

1. Group FOQA data by aircraft type.
2. Isolate departure and arrival sections of each flight. Note that ANP performance models are only applicable in the terminal area (below 10,000 ft altitude), therefore the regression should only be applied to the dataset which conforms to these limits.
3. Identify power mode of the aircraft using normalized thrust lever angle (NTLA) variable in the FOQA data.
4. Perform linear regression on the data grouped by power mode to obtain the set of regression coefficients.
5. Use coefficients for thrust prediction.

Results and Discussion

Some preliminary results have been obtained for step 3 of the methodology. Identification of the power mode of the aircraft requires grouping data by the NTLA variable. At present, AEDT contains six power modes for departure for most jet aircraft. These are outlined in Table 4.

Table 4. Description of ANP Thrust Codes and corresponding ranges in FOQA

ANP Thrust Code	Description	Range for NTLA
T	Full takeoff thrust	97.5% and above
F	5% reduced takeoff thrust	92.5% to 97.5%
X	10% reduced takeoff thrust	87.5% to 92.5%
Z	15% reduced takeoff thrust	82.5% to 87.5%
C	Full climb thrust	82.5% and below
D	10% reduced climb thrust	Not considered

One final data processing step is to remove the transient periods where the thrust is not stabilized. This was done by performing an analysis on the NTLA values, using finite differences to calculate derivatives. The time derivative was calculated using three finite difference formulas, the forward, central, and backward difference, respectively.

$$\frac{NTLA_{i+1} - NTLA_i}{t_{i+1} - t_i}, \frac{NTLA_{i+1} - NTLA_{i-1}}{t_{i+1} - t_{i-1}}, \frac{NTLA_i - NTLA_{i-1}}{t_i - t_{i-1}}$$

By applying a suitable threshold to these computed derivatives, unsteady regions of thrust can be identified and eliminated. Figure 8 shows the outcome of this process and its effect on the sampled trajectory altitude and airspeed. Once transient thrust data is removed, the remaining dataset can easily be grouped based on the ranges specified in Table 4.

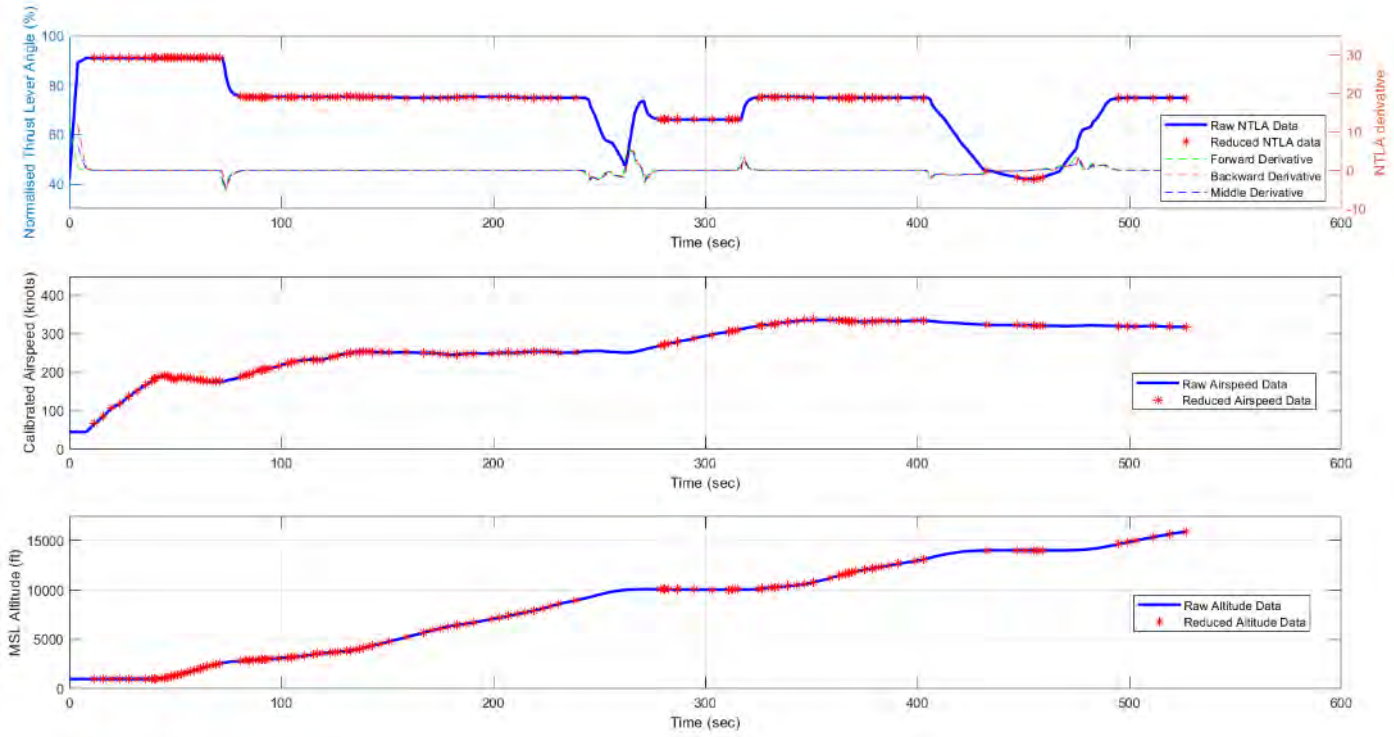


Figure 8. Filtering of FOQA data with finite difference time derivatives of NTLA.

This work is currently in progress and the next steps focus on the computation of the thrust coefficients using regression models.

Task 1 – Improved Departure Modeling: Comparison of NADP Profiles to Real-world Operations

Georgia Institute of Technology

Objective

Prior research in ASCENT Project 45 provided recommendations for NADPs to be modeled in future versions of AEDT. Comparisons were made between NADP profiles within the NADP library to determine the differences between each profile. As a result, the six NADP-1 and thirteen NADP-2 profiles were reduced to two profiles that are most representative of the variability among each group of NADPs. This task aims to investigate similarity between the recommended NADPs and real-world departure operations.

Research Approach

Previous research efforts made in ASCENT Project 45 computed comparisons between the six NADP-1 and thirteen NADP-2 profiles defined in the NADP library. These profiles were modeled in AEDT for the B737-800, A320-211, and A330-301 at stage lengths 1, 3, and 5. NADP-1 profile 1 and NADP-2 profile 11 were found to be most representative, based on their ability to capture variability across all NADP profiles within their respective groups. However, comparisons between the two recommended NADP profiles and real-world operations are needed to ensure that the recommended NADP profiles are representative of real-world operations. For the purposes of this task, comparisons are made for the three airframes with NADP profiles already modeled in AEDT only. Table 5 outlines the 18 combinations of NADP settings to be used for comparison.

Table 5. NADP profiles to be compared.

NADP Type	Profile Number	Stage Length	Airframe
NADP-1	1	1, 3, 5	B737-800, A320-211, A330-301
NADP-2	11		

To accomplish this task, the trajectories to be compared are processed to align data points and remove anomalous flights. Next, the trajectories are grouped by airframe and stage length to enable comparisons between similar operating procedures across the two datasets. Comparison metrics are then computed between real and NADP trajectories for each grouping, enabling their similarities to be investigated. Figure 9 provides an overview of this approach.

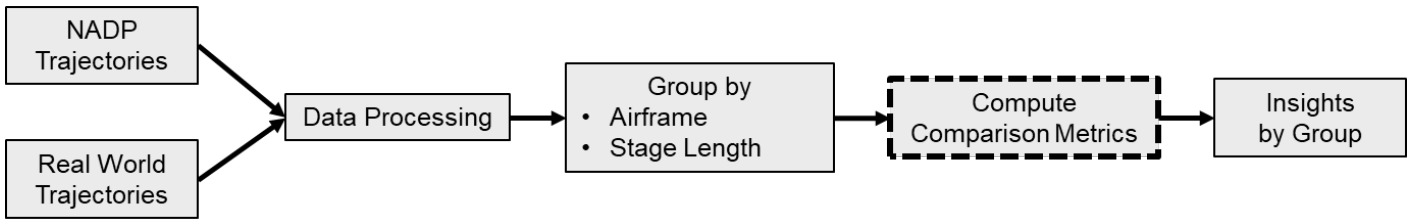


Figure 9. Summary of overall approach.

Real-world Dataset

The real-world data to be utilized was obtained from publicly available OpenSky ADS-B data, which contains radar-tracked aircraft trajectories. For the purposes of this task, an OpenSky ADS-B dataset with one full year of departure operations from all airlines operating the B737-800, A320-211, and A330-301 at SFO for the year 2019 was selected for comparison. This real-world dataset contains a total of 8,517 flights operated by 23 airlines.

Performance-based parameters such as altitude and ground speed are readily available in this dataset. Due to the nature of radar-track data, important performance metrics which define operating conditions such as thrust and weight for each flight are missing. Flight great-circle distance computed from origin-destination pair was identified as a proxy for weight. Each flight within this dataset contains trajectory information sampled up to 25 nautical miles of cumulative ground track distance.

To accurately make comparisons, different operating conditions must be grouped and analyzed separately. The grouping parameters identified are airframe and flight great-circle distance. Because AEDT estimates weights by bins of stage lengths, flight distance was converted into stage lengths as defined by Table 6. The distribution of stage lengths by airframe is illustrated in Figure 10. Notably, only the B737-800 contains flights operated at stage length 5 and the A330 contains much smaller number of flights operated at only stage lengths 4, 6, and 7. By grouping the trajectories, direct comparisons to NADP profiles modeled for different airframes at varying stage lengths can be made. In future iterations, differences in operating procedures between airlines can also be investigated by further grouping the dataset by operator.

Table 6. Stage length bin definition

Stage Length	Flight Great-Circle Distance [nmi]
1	0–500
2	500–1,000
3	1000–1,500
4	1500–2500
5	2500–3500
6	3500–4500
7	4500–5500
8	5500–6500
9	6500–11,000
M	Maximum range at maximum takeoff weight

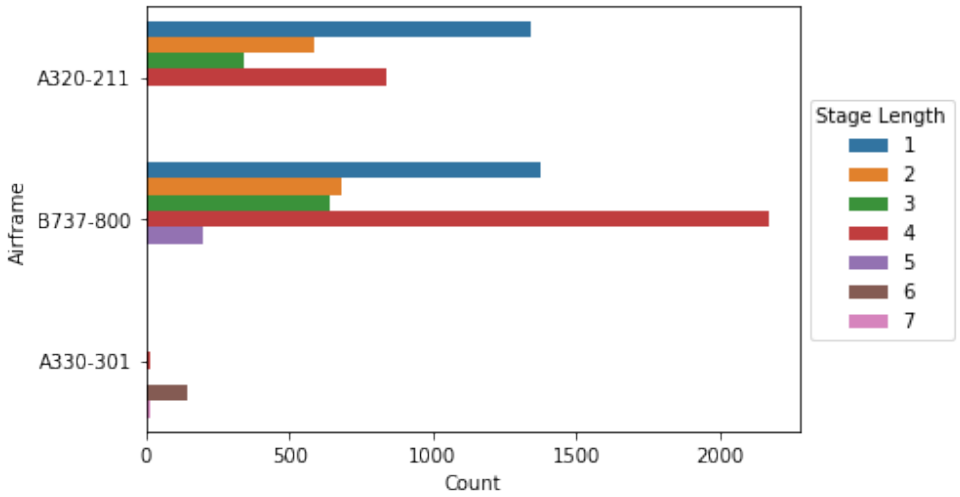


Figure 10. Number of flights by airframe and stage length in the real-world dataset.

Data Processing

The real-world dataset must be preprocessed to remove anomalous flights and adjusted such that pairwise comparison can be made against NADPs. Anomalous trajectories can be identified by examining the final altitude and the takeoff distance based on expected behaviors. Specifically, flights that do not cross 10,000 ft above ground level (AGL) at cumulative ground track distance of 25 nmi or have takeoff ground-roll segment greater than the maximum runway length of two nmi at SFO are removed from the comparison. Using this procedure, 2.88% of the flights are identified as anomalous, as shown in Figure 11.

Additionally, several flights contain marginally negative altitude data points during ground roll which can safely be shifted to 0 ft AGL. The remaining variation in the ground-roll segment can then be used to capture the effect of weight on takeoff distance. Finally, real-world trajectories exhibit climb acceleration beyond 10,000 ft AGL which is not modeled in the NADP profiles as illustrated in Figure 12. To avoid making comparisons beyond the departure segment, the trajectory of each flight is capped at 10,000 ft.

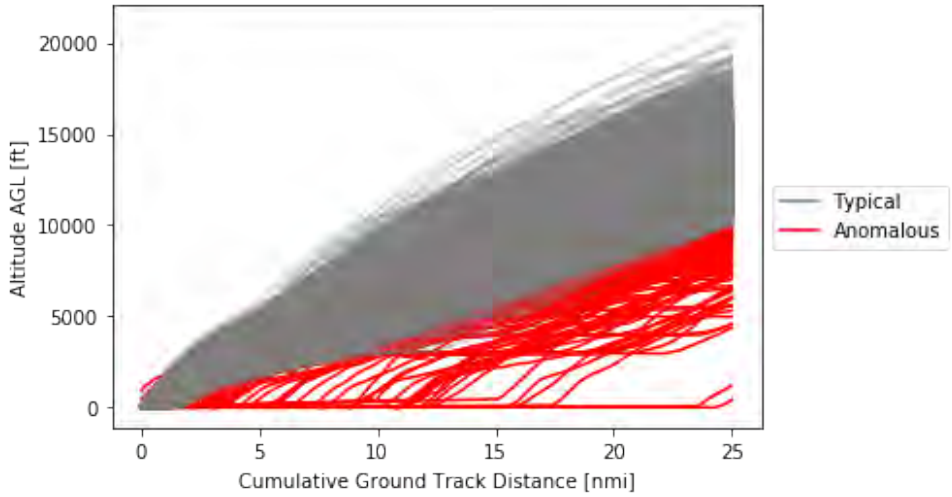


Figure 11. Real-world flights classified as anomalous and removed from comparison.

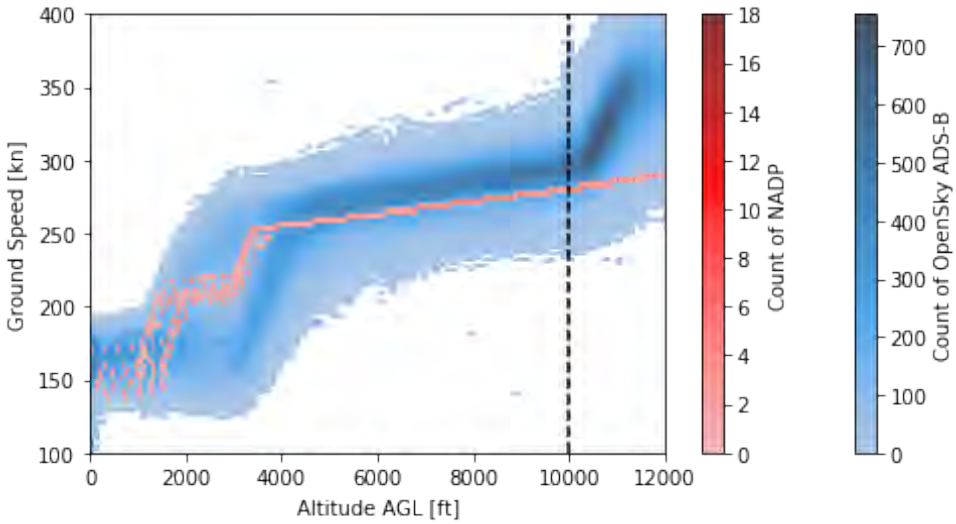


Figure 12. Distribution of ground speed as a function of altitude above ground level (AGL).

Comparison Metrics

In this task, altitude and ground speed profiles are utilized to compute comparison metrics as these parameters are already available across both the real-world and NADP datasets. In future iterations of this task, noise- and emissions-based metrics can be utilized as additional metrics once the real-world profiles have been sufficiently reduced to a feasible number of representative operating procedures and modeled in AEDT. This may be accomplished using clustering algorithms on existing metrics.

To facilitate computation of profile differences, profiles from both datasets are resampled using linear interpolation to align each point at a fixed cumulative ground track distance increment of 0.33 nmi. This enables computation of pairwise absolute differences at each data point along the profiles. Figure 13 visualizes this comparison process and Figure 14 provides examples for B737-800 comparison.

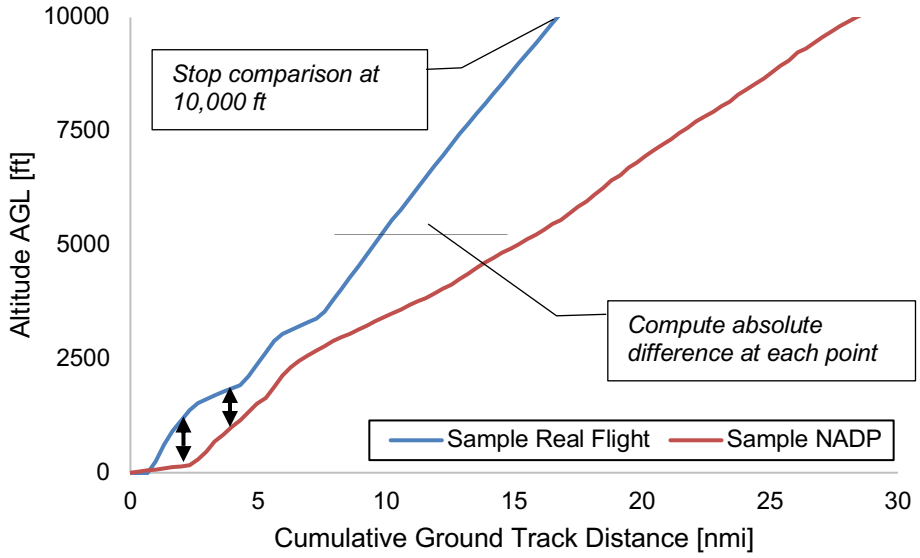


Figure 13. Trajectory comparison method visualization.

Because comparison is capped at 10,000 ft AGL, the number of data points per profiles being compared varies depending on the vertical speed. A flight that is rapidly climbing will reach 10,000 ft AGL at a shorter cumulative ground track distance and hence contain a smaller number of data points. For the current datasets, this yields a median of 49 data points per comparison.

To mitigate this, a root-mean-squared (RMS) based methodology is employed as an overall measure of discrepancy between any two profiles being compared, where x represents the comparison metric of interest and n is the minimum number of data points between either profile.

$$x_{rms} = \sqrt{\frac{\sum_{i=0}^{n}(x_{Real,i} - x_{NADP,i})^2}{n}}$$

The resulting RMS values indicate how separated the profiles are from each other on average, with higher values implying larger separation. Two matrices of altitude and speed RMS difference between real profiles and NADPs are then generated for each airframe. Then, the resulting RMS matrices are joined with meta data from the real-world trajectories to aggregate comparisons by stage length and by airline.

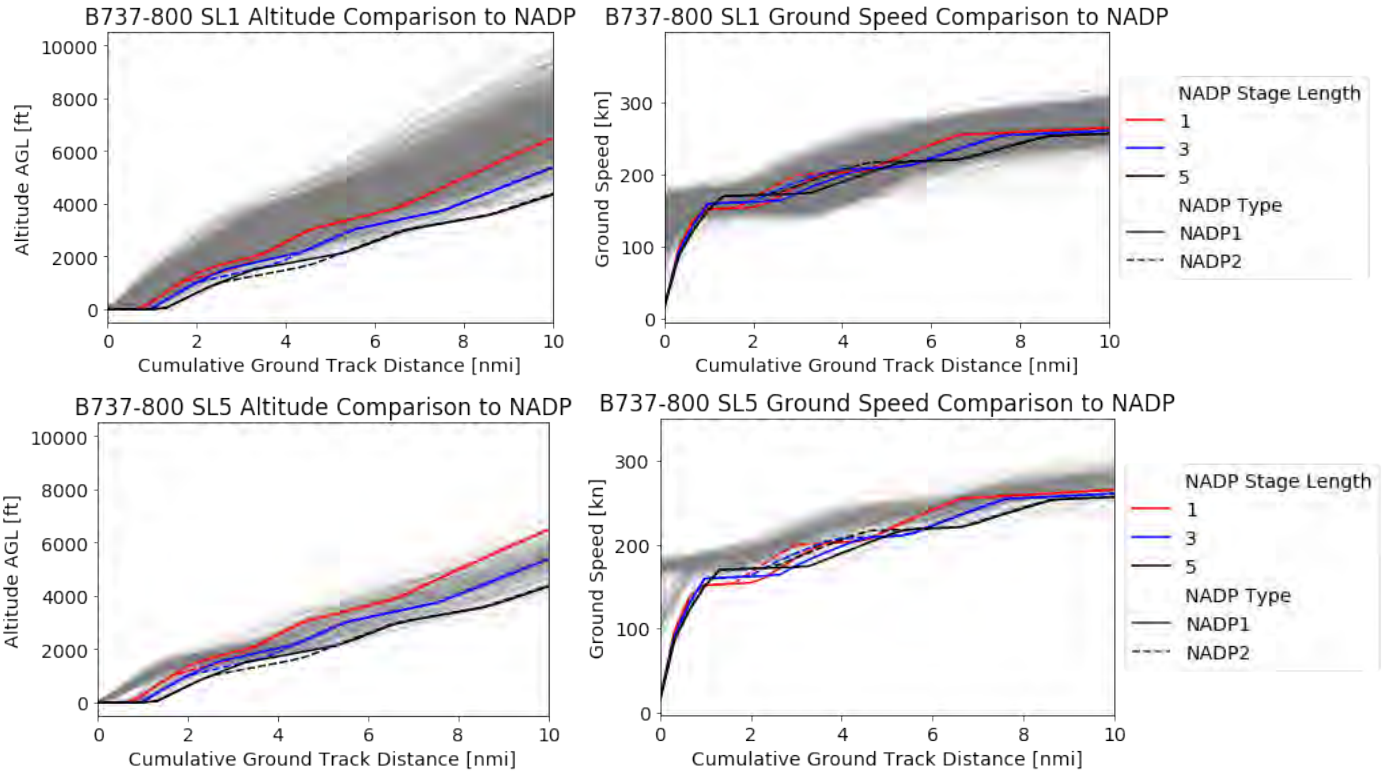


Figure 14. Sample comparison between NADP and real B737-800 flights

Results and Discussion

The resulting RMS differences between each real flight and each NADP profile are aggregated by comparison grouping. The grouped results are then visualized using boxplots. Figure 15 and Figure 16 illustrates the aggregated results where each box represents a distribution of RMS for all airframes (compared to NADP modeled using the same airframe) at a given stage length. Results separated by airframe are presented in Table 7 and Table 8. Such visualization enables the median and the interquartile range (IQR) of RMS distributions by group to be compared. The points plotted outside of the boxplot whiskers indicate outliers. Any RMS below $Q1 - 1.5IQR$ or above $Q3 + 1.5IQR$ are classified as outliers.

Figure 15 illustrates that varying NADP stage length noticeably influences altitude RMS. This makes intuitive sense since higher stage lengths imply higher weight and hence affect the altitude profile. When NADP is modeled at stage length 1, the resulting RMS distributions indicate that the profile is most similar to real flights flown at stage length 2 as indicated by the medians. At NADP stage lengths 3 and 5, real flights flown at stage length 5 and become the most similar, respectively. In all three NADP stage length cases, the distributions of RMS show an increasing shift when compared to other real flight stage lengths with increasing stage length difference. This trend is reflected across both NADP-1 profile 1 and NADP-2 profile 11. This result implies that NADP altitude profiles are generally more similar to real profiles flown at higher stage lengths. However, this does not imply that NADP altitude profiles are lower or higher than real profiles since RMS computation involves squaring the differences.

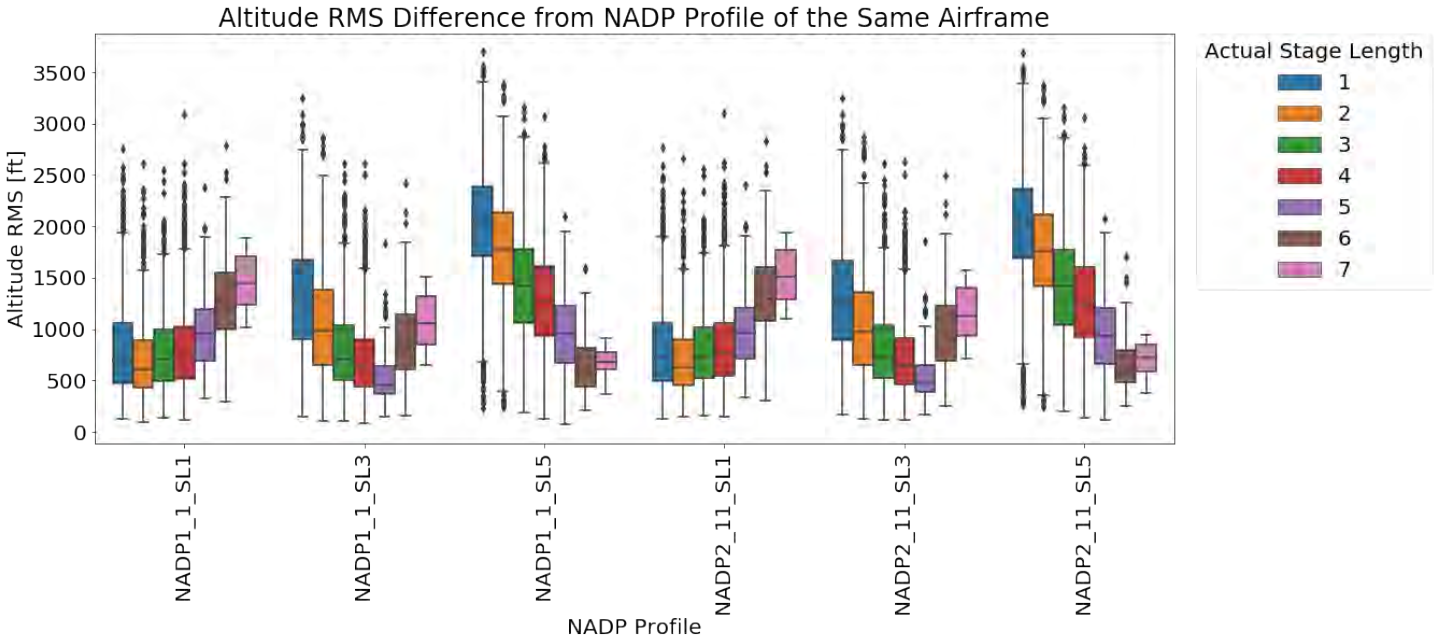


Figure 15. Altitude RMS difference from NADP profiles computed per airframe and grouped by stage length.

Figure 16 illustrates that varying NADP stage length result in marginal variation in ground speed RMS. This indicates that the results found for altitude RMS can be attributed to mostly differences in vertical speed utilized at different stage lengths. Furthermore, real flights flown at stage lengths 1–4 appear to be equally similar to NADP in all cases while stage lengths 5–7 deviate from this trend. This could be due to the lack of real flight data flown at stage lengths 5–7. Stage length 5 only contains flights flown by one airline using the B737-800 and stage lengths 6–7 only contain flights flown by A330-301, which has a much smaller number of flight records as shown in Figure 10.

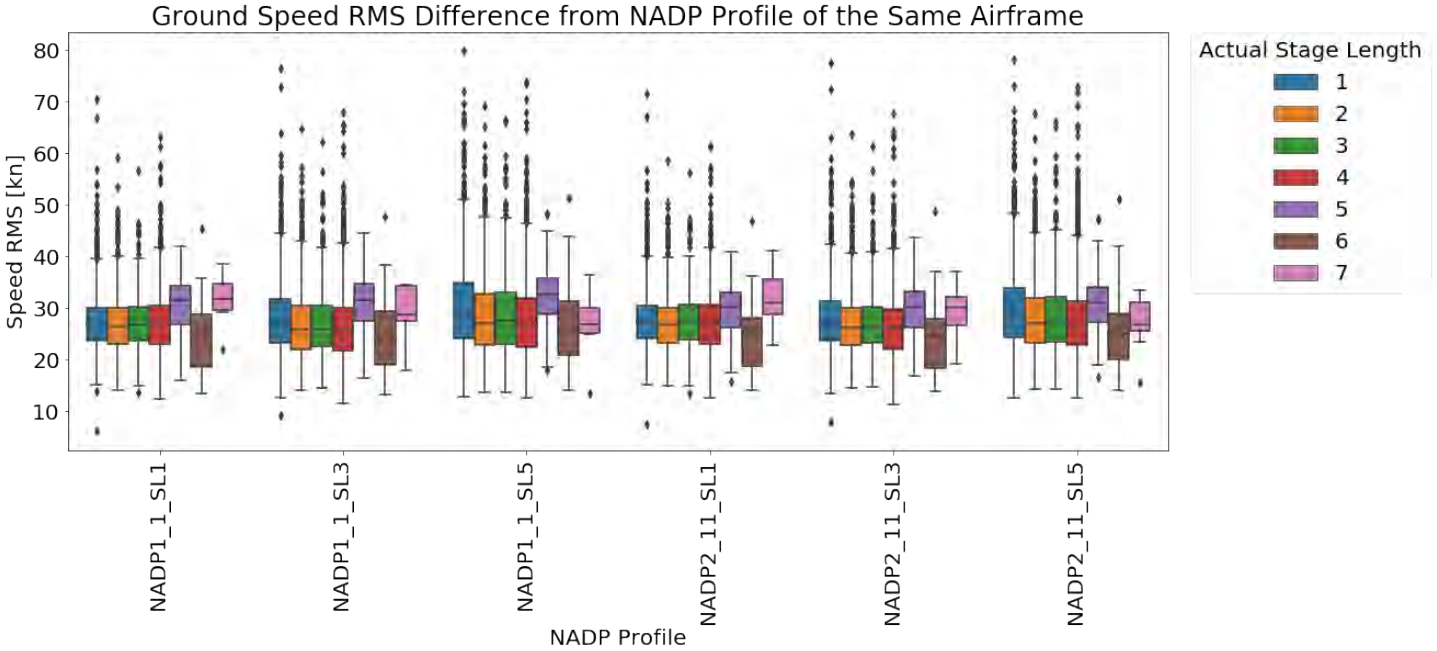


Figure 16. Ground speed RMS difference from NADP profiles computed per airframe and grouped by stage length.

Table 7. Summary of median RMS differences for NADP-1 profile 1

		Median Altitude RMS [ft]			Median Ground Speed RMS [kn]		
NADP Profile	Actual Stage Length	A320-211	B737-800	A330-301	A320-211	B737-800	A330-301
NADP1_1_SL1	1	673.11	738.71		26.97	26.54	
	2	650.4	567.08		26.88	26.03	
	3	737.63	687.56		27.98	26.25	
	4	836.3	710.79	726.74	27.69	26.22	27.73
	5		955.25			31.5	
	6			1221.04			24.84
	7			1447.89			31.64
NADP1_1_SL3	1	1197.28	1368.38		27.19	26.58	
	2	972.85	998.47		25.96	25.7	
	3	701.52	713.78		26.16	25.86	
	4	659.08	612.47	1137.23	25.94	25.67	26.66
	5		464.28			31.43	
	6			830.45			24.23
	7			1060.67			28.82
NADP1_1_SL5	1	2011.37	2102.08		29.57	28.27	
	2	1794.11	1780.05		27.76	26.76	
	3	1448.09	1416.11		27.63	27.29	
	4	1220.7	1277.74	1910.05	26.58	26.73	26.71
	5		949.27			32.51	
	6			589.94			25.76
	7			687.11			26.87

Table 8. Summary of median RMS differences for NADP-2 profile 11

		Median Altitude RMS [ft]			Median Ground Speed RMS [kn]		
NADP Profile	Actual Stage Length	A320-211	B737-800	A330-301	A320-211	B737-800	A330-301
NADP2_11_SL1	1	695.28	745.53		27.6	26.88	
	2	685.32	578.21		27.77	26.12	
	3	780.55	706.21		28.91	26.39	
	4	896	728.52	747.36	28.52	26.01	29.65
	5		966.67			30.26	
	6			1298.96			25.04
	7			1511.74			31.04
NADP2_11_SL3	1	1164.64	1363		27.48	26.78	
	2	961.15	1000.38		26.56	25.82	
	3	725.97	722.39		27.65	25.78	
	4	694.28	627.93	1066.71	26.94	25.5	28.7
	5		482.17			30.16	
	6			917.22			24.67
	7			1130.19			30.06
NADP2_11_SL5	1	1975.8	2091.61		28.58	28.1	
	2	1763	1766.61		27.28	26.73	
	3	1423.33	1405.8		27.91	26.83	
	4	1188.94	1265.61	1836.9	26.74	26.7	27.46
	5		929.36			31.08	
	6			590.7			24.95
	7			728.38			26.85

Task 2 – Arrival Profile Modeling

Georgia Institute of Technology

Objective

The objective of Task 2 was to find and develop arrival procedures for AEDT that better capture existing operations as observed in airlines' flight data. The 2020-2021 objective specifically has been to determine what additional arrival profiles must be included in the AEDT models to represent real-life arrival profiles based on 2019 flight operations of multiple airlines. Previous research in this area has shown that a systemic statistical method must be developed to define the continuous descent approach and level-off for all aircrafts. In particular, the level-off trends remained unclear from the previous year's work. Hence it was recommended to incorporate threaded track data to confirm these findings and explore these uncertainties further.

Introduction

AEDT currently models arrival profiles using specified fixed-point trajectories or manufacture-provided procedures. Task 2 compares data from real flights to the models in AEDT to make recommendations on how to improve AEDT models such that they capture real flight operations.

Research Approach

To accomplish the objective outlined above, the goal was to examine prior years' research and arrival profile recommendations; study prior algorithms for level-off detection, level-off length calculation, and other parameters; conduct similar efforts with arrival profiles from other data sources available (threaded track, ADS-B); develop a modified algorithm for applying to threaded track/ADSB data; use statistical analysis to confirm the original recommendations and refine as needed; group airports into similar arrival procedure behaviours using machine learning or data analysis techniques based on threaded track data; obtain diversity of operational conditions, statistical trends, etc for clusters of data; and finally demonstrate implementation in AEDT.

Hence, the researchers for this task first enumerated and documented a summary of the recommendations for arrival modelling from the previous year, with a focus on understanding: What was done previously? How was it done? What needs improving? And most importantly, how can it be improved upon? In documenting the previous year's recommendations, the researchers were able to scope the level-off altitudes that were high priority in determining level-off trends as seen in Table 9 below:

Table 9. Summary of existing recommendations for AEDT arrival modeling

Aircraft	CDA	LO 2000	LO 3000	LO 4000
B717-200	X	1	2	3
B737-700	X	1	2	3
B737-800	1	2	X	3
B737-900	1	2	X	3
B757-200	X	1	2	3
B757-300	1	2	X	3
B777-200ER	1	2	X	3
B777-200LR	1	2	X	3
MD-90	X	1	2	3
A319-100	1	2	X	3
A320-200	1	2	X	3
A321-200	1	2	X	3
A330-200	1	2	X	3
A330-300	1	2	X	3

Here, X corresponds to "Already in AEDT", 1 is high priority, 2 is medium priority, and 3 is low priority.

From this, the task breakdown as prescribed was:

1. Expand one-airline analysis from previous year to new multi-airline flight data.
   a. Identify popular altitudes and tolerances.
   b. Define continuous descent approach and level-off based on aircraft.
   c. Identify operational differences based on airline.
2. Continue analysis of level-off length histogram.
   a. Evaluate the current assumption that manufacturer supplied level-off distance value is the best option for AEDT setting.
   b. Conduct detailed analysis on airport subsets to look for level-off distance trends.
3. Edit/improve existing Python code for level-off detection and grouping to enable analysis of 2019 data.
   a. Define statistics-based analysis methods

Methodology

Despite having the intention of using threaded track data for this effort, a subset of 2019 OpenSky ADS-B data was used due to delays in receiving the threaded track dataset. The OpenSky dataset includes data from 415 flights operated by 32 airframes and 44 airlines which arrived at San Francisco International Airport (SFO). Statistical trends of approach profiles were identified in the OpenSky data and compared to existing arrival models in AEDT to locate gaps or inaccuracies in AEDT. The aircraft types observed in this dataset are:

'A21N' 'A319' 'A320' 'A321' 'A332' 'A343' 'A346' 'A359' 'A35K' 'A388' 'B38M' 'B39M' 'B737' 'B738' 'B739' 'B744' 'B748' 'B752' 'B753' 'B763' 'B764' 'B772' 'B77W' 'B788' 'B789' 'CL30' 'CRJ2' 'CRJ7' 'CRJ9' 'E75L' 'E75S' 'GLEX'

The dataset is in the form of a .csv table and is analyzed using Python. The code is built such that it may be implemented to larger datasets such as the threaded track data as they become available. The code differentiates between separate flights using the threaded track primary key associated to the flight. For each flight, potential level-offs are identified by finding consecutive data points which fall within some vertical speed tolerance. These flat portions of flights are compared against some ground track distance tolerance to determine if these points represent a level-off. All potential level-offs identified within 1500 feet above touchdown and within 5 miles of touchdown are disregarded as they fall in the category of final approach. Thus, the task in refining level-off detection for analysis is determining the appropriate vertical speed tolerance and ground track distance tolerance.

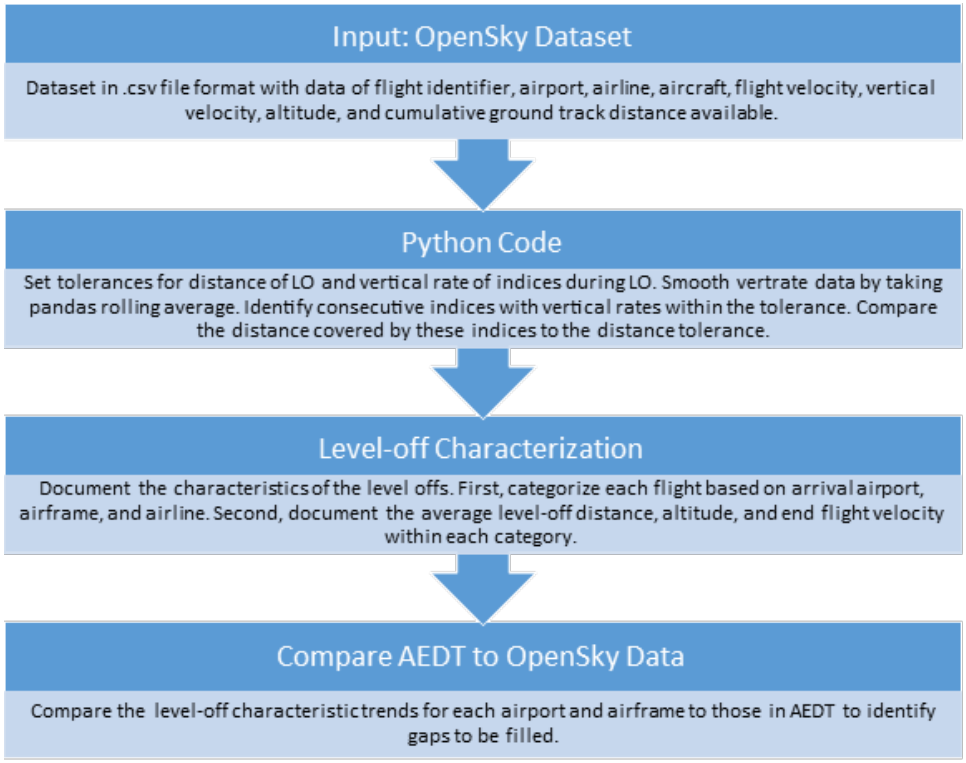


Figure 17. Flowchart of Task 2.

A systemic, statistical method for the determination of these key tolerances is being worked on. This was begun by conducting a design of experiments and writing a design of experiments (DoE) code in Python. The following DoE was created where the distance speed tolerances were one and two nmi corresponding to 0 and 1, respectively, and vertical speed tolerance levels of 200 to 500 ft/min corresponding to 0–3, respectively, as seen in Table 10:

Table 10. Level-off tolerance testing design of experiments

Experiment	Distance Tolerance (nmi)	Vertical Speed Tolerance (ft/min)
1	0	0
2	1	0
3	0	1
4	1	1
5	0	2
6	1	2
7	0	3
8	1	3

Using this to guide the cases or experiments that were run with various settings on the code, it was found that larger vertical speed tolerances did not affect the level-off detection and instead shorter level-off distances were not being detected by the code even if they seemed to be visually present in certain plots. This is best exemplified with the following plot in Figure 18. Examining distance tolerances was deemed high priority as seen with the level-off detected at 6000 ft by the code, and the potential level-off visually detected at 3000 ft which remains undetected by the Python code.

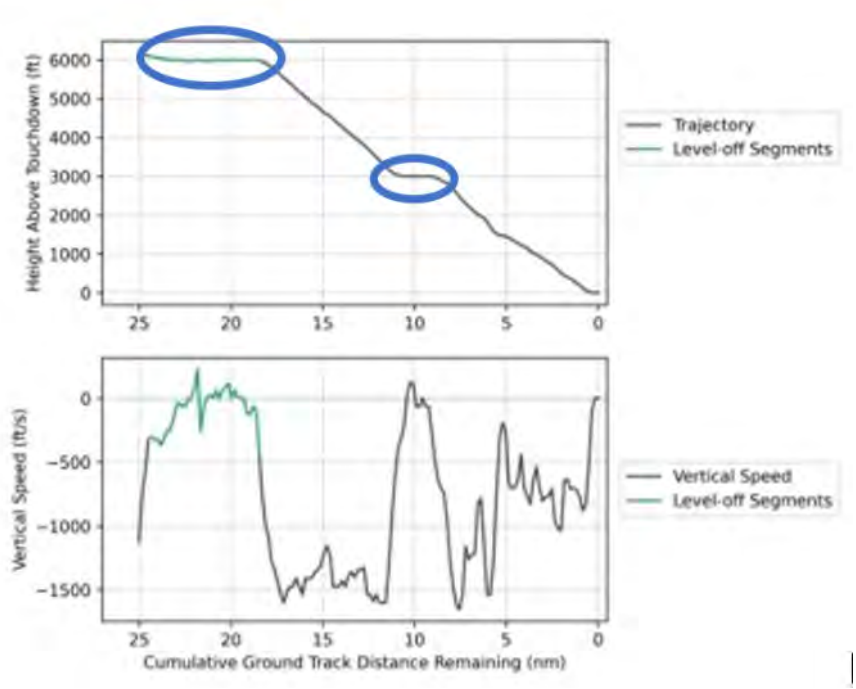


Figure 18. Distance tolerance two nmi, vertical speed tolerance 500 ft/min, Flight 110.

Hence lowering the level-off distance tolerances below two nmi would be required to help capture more flights with vertical speed tolerances in an appropriate range. A small sample set of code was run to understand the distribution of "shorter" level-offs across 10 flights, 50 flights, and soon the whole dataset to further examine these ambiguous level-offs.

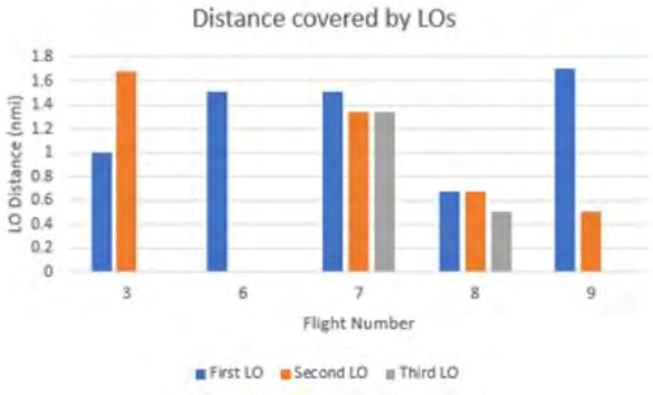


Figure 19. Level-off distances for first 10 flights.

When examining the first 10 flights, the researchers noted the clustering of level-off distances in the range 0.3–1 nmi and 1–2 nmi. Hence these groupings were used when examining the dataset of the first 50 flights and taking the two clusters of level-off distances and finding the mean, median, and mode between these two ranges. The findings for the two clusterings are captured in the tables below:

Table 11. Central tendencies for level-off distances for first 50 flights (0.3– 1 nmi)

Mean (0.3-1)	0.751324
Median (0.3-1)	**0.503356**
Mode (0.3-1)	0.671141

Table 12. Central tendencies for level-off distances for first 50 flights (1-2 nmi)

Mean (1-2)	1.339576
Median (1-2)	**1.342282**
Mode (1-2)	1.677853

However, when analysing the whole dataset, the airframes need to be accounted for and compared to AEDT. Hence the profile points were extracted from AEDT for each airframe, altitude, and distance with the intention of modifying the Python code such that it would have the ability to categorize level-offs by airframe and central tendencies.

Hence, in parallel with the statistical analysis of level-off tolerance determination, developments were made in the Python code to document and calculate characteristics of the level-offs such that they may be analyzed based on arrival airport, aircraft, and airline. In addition, the horizontal flight speed reached at the end of the level-off and the altitude at which the level-off occurs is documented. The three categorizations between airport, aircraft, and airline are made in anticipation of differences in arrival behavior for differences in each category. Upon grouping the flights by its category, the level-off characteristics of those flights are analyzed to identify prominent level-off altitudes, distances, and flight speeds. These level-off characteristic trends are then compared to the airframe-specific arrival models in AEDT.

Results and Discussion

Currently available results include a design of experiments approach to explore level-off distance tolerances and a baseline Python code to identify level-offs and plot these results.

The Python code is currently under development and has yet to yield notable results. Thus, next steps include the completion of the code such that level-off characterization can be performed for the entire dataset. The output of the code will include a calculation of the mean, median, and mode of the characteristics for each category. These statistical characteristics will be used in addition to a graphing of these metrics to identify significant trends that may be used to generalize the arrival profile for use in AEDT.

However, it should be noted that the OpenSky dataset which is currently being used only includes arrivals at SFO. Characterization of arrival profiles at multiple airports will be an important step in yielding results which can be translated into comparison with, and implementation into, AEDT's arrival modelling.

Task 3 – Full Flight Modeling

Georgia Institute of Technology

Objectives

The objective of this Task is to improve the full flight modeling within AEDT. At present, to conduct a full flight in AEDT, a sensor path must be defined for each aircraft type and each origin-destination city-pair and is based on utilizing radar track data, which is being replaced with the new threaded track approach developed by The MITRE Corporation. Even with the threaded track data, this can be a daunting task for the user to set up an AEDT study. This task investigates the accuracy of the threaded track data compared to a truth model, FOQA, data, where all states of the aircraft flight are known, including thrust, weight, and fuel flow. Statistical analysis will be performed to determine how accurate the threaded tracks are and provide guidance to the FAA on improvement areas.

Research Approach

This Task's initial focus is to investigate the accuracy of thread track data and analyze the average behavior of FOQA flight data. Thus, the research approach will be described in two parts.

Investigate the accuracy of the thread track data compare to FOQA data

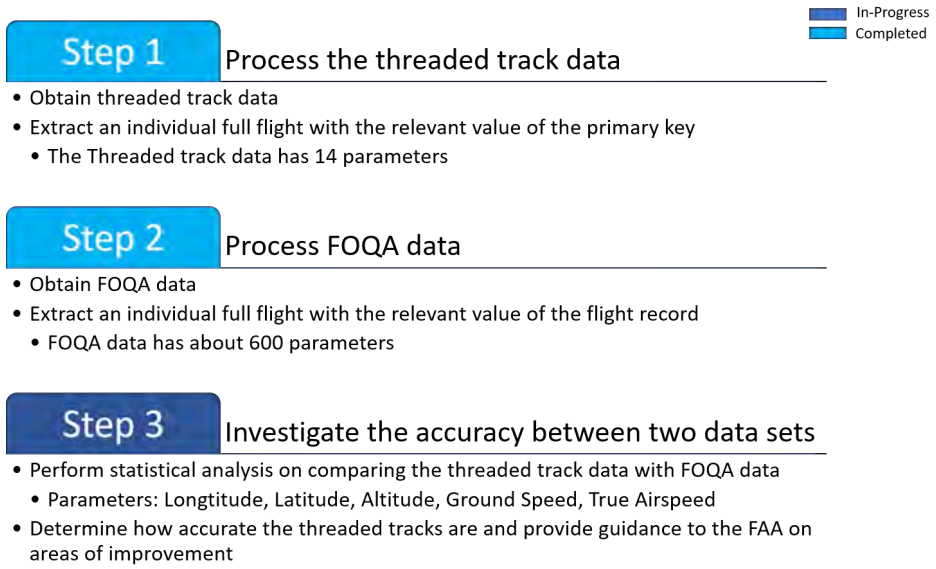


Figure 20. Research approach for investigating the accuracy of the threaded track data.

This section aims to study how accurate the threaded track data is compared to the true model FOQA data. The threaded track data has only 14 parameters and low-fidelity data compared with FOQA data. MITRE's new threaded track approach is used for establishing a sensor path to conduct the full flight modeling in AEDT. Therefore, it is necessary to analyze how accurate the threaded track data is because the full fight modeling performance in AEDT varies. Therefore, it is also necessary to analyze how accurate the threaded track data is because the full fight modeling performance in AEDT depends on the data's accuracy. The research method is represented in Figure 20. Step 3 of the method is currently in progress and is dependent on receiving the threaded track data.

Study the average behavior within FOQA data

This section's main objective is to analyze the average behavior of FOQA data and compare it to what AEDT has for full flight modeling. FOQA data includes about 21,000 flights from 14 different airframes. The first step is to find out if a specific flight pattern exists within each city pair. Instead of analyzing about 1001 city-pairs in the FOQA data, the top five city pairs were selected and investigated based on the highest flight operations. Scripts have been created to automate data extraction from the raw FOQA file so that only flight data corresponding to the relevant city pair can be extracted. Next, a trajectory was generated using each city pair's extracted data to analyze a specific pattern through the density-based spatial clustering of applications with noise (DBSCAN) algorithm. DBSCAN is a popular data clustering algorithm in machine learning and groups together points that are close to each other based on distance measurement. Two parameters of the DBSCAN algorithm are required to adjust the optimal clustering results. The first parameter is eps, which is the maximum distance between two samples for one to be considered in the other's neighborhood. The second one, min_samples, is the number of points in a neighborhood considered a core point. Longitude and latitude were used for creating the trajectory and clustering. The detailed procedure is shown in Figure 21.

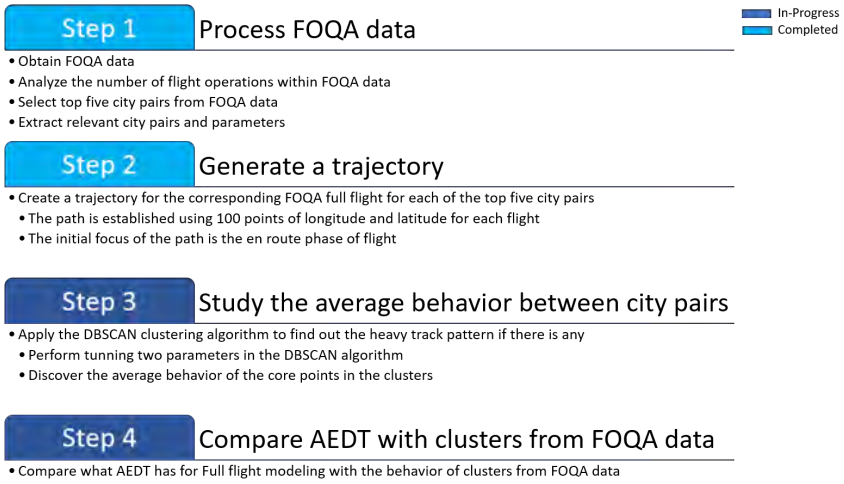


Figure 21. Research approach for analyzing the average behavior of FOQA data

Results and Discussion

This section presents preliminary results for investigating the average behavior for top city pars. The results of the top five city pairs using Python scripts are presented in Table 13. The city pair from ATL to LAS has the highest number of flight operations with 417 flights, while the city pair from SEA to LAS has the lowest number of flight operations with 315 flights in the top five city pairs data. The city pair from SLC to LAS has the shortest length in terms of the great-circle distance.

Table 13. Top five city pairs from FOQA data

Origin	Destination	# of flights	Origin	Destination	# of flights	Total # of flight	GC distance (nm)
ATL	LAS	417	LAS	ATL	30	447	1514.488
MSP	LAS	378	LAS	MSP	24	402	1127.305
ATL	SLC	330	SLC	ATL	32	362	1378.838
SLC	LAS	255	LAS	SLC	104	359	319.622
SEA	LAS	315	LAS	SEA	18	333	753.300

** ATL: Hartsfield-Jackson Atlanta International Airport
MSP: Minneapolis-Saint Paul International Airport
SLC: Salt Lake City International Airport
LAS: McCarran International Airport
SEA: Seattle-Tacoma International Airport

Figure 22 represents the clustering results with the DBSCAN algorithm performed to city pairs from ATL to LAS. Even though the total number of flight operations for city pairs from ATL to LAS is the same for both plots, the DBSCAN algorithm shows different results depending on the parameters' values. The first plot on the left shows results with five eps and three min_samples, and the second plot on the right shows results with four eps and 15 min_samples. The number of outliers for the first plot is 39, which is less than the second plot. However, the second plot shows a better clustering result displaying a good illustration of the meaningful track trajectory. The contents of parameters and outliers are given in Table 14.

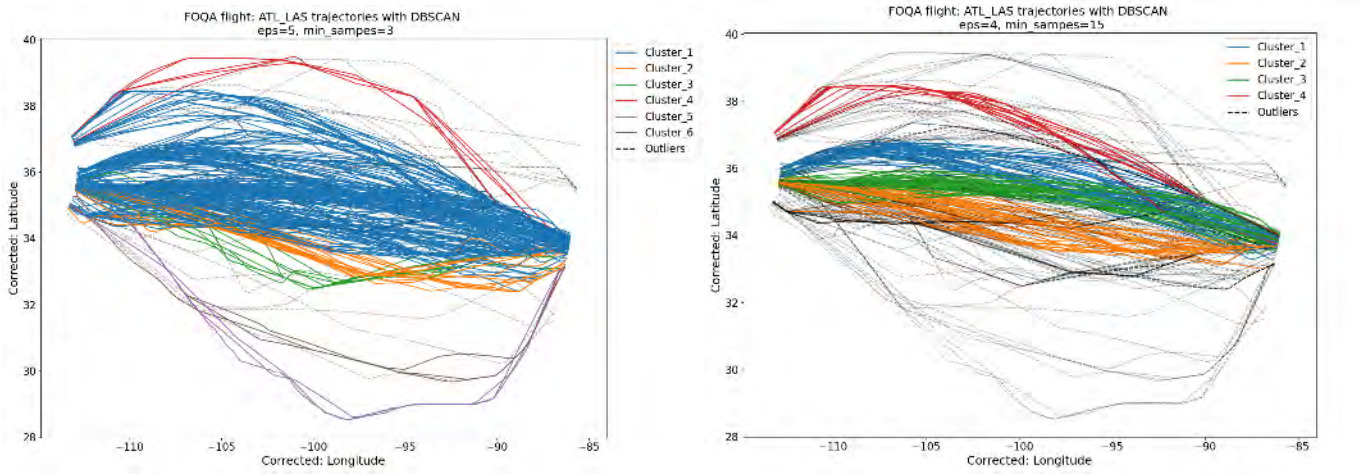


Figure 22. Clustering results of ATL to LAS trajectories with DBSCAN.

Table 14. Parameters and outliers in DBSCAN algorithm

City Pair	eps	Min_samples	Number of clusters	Number of outliers
ATL to LAS	5	3	6	39
ATL to LAS	4	15	4	164

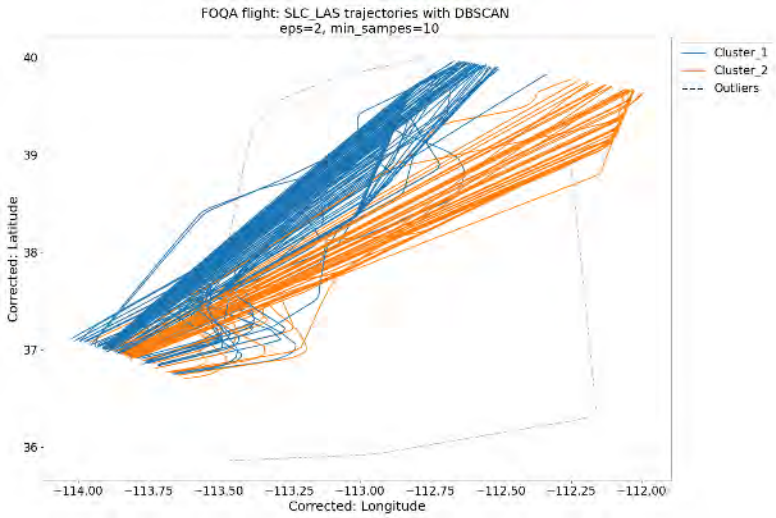


Figure 23. Clustering results of SLC to LAS trajectories with DBSCAN.

The clustering results of city pairs from LAS to SLC is illustrated in Figure 23. Most of the full flight trajectory from LAS to SLC is shown in two distinct patterns. Compared to the clustering results of city pairs from LAS to ATL, two patterns were well established. There are only two outliers and only 0.7% of the total flight. The city pair's clustering results from SLC to LAS are well detected compared to the city pair from ATL to LAS because the flight distance is relatively short. Moreover, there is less uncertainty, such as weather and traffic congestion.

Task 4 – System Testing and Evaluation of AEDT

Georgia Institute of Technology

Objective

To provide the best possible environmental impacts modeling capabilities in AEDT, the FAA AEE continues to develop AEDT by improving existing modeling methods and data and adding new functionalities. The AEDT development team has been exercising an agile development process, where minor updates are released in new Sprint versions every three weeks, and major updates and/or new functionalities are incorporated as new versions of AEDT. The FAA AEE seeks an independent effort in system testing to evaluate the accuracy, functionality, and capabilities of AEDT and support the future development process. Thus, the objective of this Task is to provide FAA with high-quality systematic testing and evaluation of AEDT 3 and its future releases to evaluate AEDT's capability and identify gaps in the tools' functionality and areas for further development.

Research Approach

Under this area, the GT research team has been coordinating with FAA on the upcoming AEDT features and conducting necessary testing and evaluation efforts of the newly incorporated capability. For each of the AEDT releases, depending on the type of updates, we identify the key features and functionalities to first conduct capability demonstration to make sure the implemented features are working properly. Then we define the scope and test cases of the system testing and evaluation effort based on the key changes to the AEDT version from the previous releases. Due to the dynamic nature of the AEDT development process, we remain flexible in the choice of the testing and evaluation approach and the work scope. We always use the best available methods and data to ensure accuracy and functionalities of future AEDT versions. When it is applied, uncertainty quantification analysis is conducted to understand the sensitivities of output response to variation in input variables and quantify major contributors to output uncertainties.

TGO/CIR Profile Development

This task focuses on updating the method of creating touch-and-go (TGO) and circuit (CIR) profiles from approach and departure procedures and using the updated method to develop TGO/CIR profiles for all 146 ANP fixed wing civil aircraft with procedure profiles. These newly developed TGO/CIR profiles are to be migrated into the FLEET database. The following goals are to be achieved:

1. Evaluate existing TGO and CIR profiles in the FLEET database to see how they are constructed.
2. Determine if the associated arrival/departure STANDARD stage 1 procedural profiles contain steps that are missing from the existing TGO and CIR profiles.
3. Create a method (program or script) to generate TGO and CIR profiles which utilize all available steps appropriately.
4. Develop TGO and CIR profiles for all civil aircraft with procedural arrival and departure profiles.

A sample C# code used for generating the original profiles was initially provided for this task. This code works by copying the procedure steps from the DEP/ARR (departure/arrival) profiles and re-arranging them in order to create TGO/CIR profiles compliant with the integrated noise model (INM) manual rules.

For a circuit profile, the code first copies the following procedure steps from the **DEPARTURE STAGE LENGTH 1** (steps shown in *italic* text were missing from the initial version of the code and were newly added as part of this feature):

1. TakeoffAirplaneProcedureStep
2. ClimbAirplaneProcedureStep
3. AccelerateAirplaneProcedureStep
4. *PercentAccelerateAirplaneProcedureStep*

This is done until an altitude of 1500 ft is reached for non-piston type large aircraft and 900 ft for piston type small aircraft. The code then creates Level steps and Fit-To-Track Distance steps at either 1500 ft or 900 ft altitude for a distance of 500 ft.

Finally, from the **ARRIVAL** procedure, at 900/1500 ft and downwards, the code copies (steps shown in *italic* text were missing from the initial version of the code and were newly added as part of this feature):

1. DescendAirplaneProcedureStep
2. *IdleThrustDescendAirplaneProcedureStep*
3. *DeceleratingThrustDescendAirplaneProcedureStep*
4. *IdleThrustLevelAirplaneProcedureStep*
5. *DeceleratingThrustLevelAirplaneProcedureStep*
6. LevelAirplaneProcedureStep
7. LandAirplaneProcedureStep
8. LandingDecelerateAirplaneProcedureStep

For a TGO profile, the code first creates a Level step at either 1500 ft or 900 ft altitude (depending on aircraft size and engine type) for a distance of 500 ft. It then copies the following procedure steps from the **ARRIVAL** procedure (steps shown in italic text were missing from the initial version of the code and were newly added as part of this feature):

1. DescendAirplaneProcedureStep
2. *IdleThrustDescendAirplaneProcedureStep*
3. *DeceleratingThrustDescendAirplaneProcedureStep*
4. *IdleThrustLevelAirplaneProcedureStep*
5. *DeceleratingThrustLevelAirplaneProcedureStep*
6. LevelAirplaneProcedureStep
7. LandAirplaneProcedureStep

From **DEPARTURE STAGE LENGTH 1**, the code then copies (steps shown in *italic* text were missing from the initial version of the code and were newly added as part of this feature):

1. TakeoffAirplaneProcedureStep
2. ClimbAirplaneProcedureStep
3. AccelerateAirplaneProcedureStep
4. *PercentAccelerateAirplaneProcedureStep*

As it can be seen in the earlier section, many new procedure steps that did not exist in the previous INM version of the code had to be defined inside the code. New arrival steps were inserted inside the code (*IdleThrustDescendAirplaneProcedureStep, DeceleratingThrustDescendAirplaneProcedureStep, IdleThrustLevelAirplaneProcedureStep, DeceleratingThrustLevelAirplaneProcedureStep)* that did not exist in the INM version of the code. Similarly, new departure steps were inserted inside the code (*PercentAccelerateAirplaneProcedureStep*) that did not exist in the INM code version.

Similar rules for the missing arrival steps were made based on the ones currently implemented for "DescendAirplaneProcedureStep". Additionally, similar rules for the missing departure steps were created for the existing "AccelerateAirplaneProcedureStep".

Besides the newly added step types, multiple problems were identified during the initial testing of the profiles that were being generated by the original INM code. New rules had to be added (code changes) in order to deal with the problems summarized below:

- **Problem 1:** Missing "DESCEND" steps that lead to invalid profiles that fail to run.
  - Solution: If there are no "Descend" steps, add one between the "Level" and "Land" steps.
- **Problem 2:** Missing "ACCELERATE" steps that lead to invalid profiles with 0 ft/min calibrated airspeed during the levelling step.
  - Solution: If there are no "ACCELERATE" steps, add one between the "Climb" and "Level" steps at 900/1500 ft.
- **Problem 3:** Non-zero/Too large climb rates that lead to the aircraft overclimbing.
  - Solution: Always climb to 1500 ft and add one acceleration step with 0 ft/min climb rate (ignore departure step order).

To deal with Problem 1, the code has been modified by first checking the entire aircraft TGO/CIR profile after it has been generated. If there are no "Descend" steps, one would be added between the "Level" and "Land" steps at 900/1500 ft with 3/5 degrees climb angle and same flap id/calibrated airspeed as the level step from earlier (complies with INM manual).

To deal with Problem 2, the code has been modified by first checking the entire aircraft TGO/CIR profile after it has been generated. If there are no "ACCELERATE" steps, one would be added between the "Climb" and "Level" steps at 900/1500 ft with the same calibrated airspeed as the first accelerate step in the departure profile and a 0 ft/min climb rate (complies with INM manual).

To deal with Problem 3, the code has been changed in order to do the following. Any aircraft now always climbs to 1500 ft and then one acceleration step with 0 ft/min climb rate (ignore departure step order) is added. This ensures that the aircraft will not overclimb.

Finally, it has been discovered that, for a few aircraft, there are multiple acceleration steps before the 1500 feet threshold that do not lead to the aircraft overclimbing. The only way to fix this is to do it manually via SQL scripts; it cannot be done via the code, hence not all the TGO/CIR profiles generated via the code are correct. Furthermore, minor issues such as incorrect flap IDs and missing thrust levels have been fixed. The original code also did not have any output files before, so this was a newly added feature. These are necessary for importing the new TGO/CIR profiles inside the FLEET database and for testing/validation purposes.

After running the code and making the necessary manual modifications using SQL scripts, a total of 146 aircraft have received new TGO/CIR profiles. The ALLPROFILES code was used to generate a study with operations for all the TGO/CIR profiles available. These have been validated by comparing the performance, noise, and emission results to the old TGO/CIR profiles (if available) and by checking that they are all compliant to the INM technical manual rules.

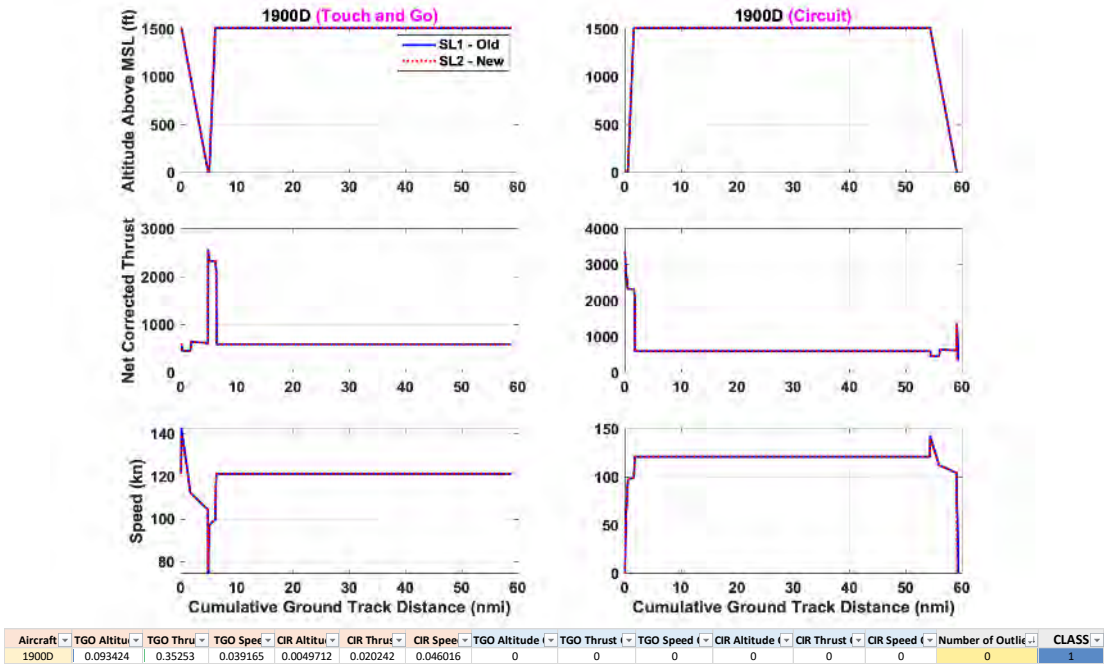
Performance, emissions, and noise reports were generated for each individual TGO/CIR aircraft operation from the ALLPROFILES study. In order to give a short summary of how the validation procedure was conducted, Table 15 shows how the new and the old TGO/CIR profiles compare in terms of altitude, thrust, and speed. Aircraft profiles are categorized into three classes. If the differences in thrust, altitude, and speed are very small, aircraft profiles are placed in Class 1. Small differences are the usual case because the old TGO/CIR profiles have not been updated while the departure/arrival profiles have. If there are noticeable differences in at most two metrics, aircraft are placed in Class 2. Finally, if there are differences in three or more metrics, the aircraft are placed in Class 3.

Similar validation exercises were conducted for emissions and noise results. In most cases, if an aircraft belongs to a certain class in the performance analysis, it will most likely belong to the same class in the emissions/noise analysis hence they are correlated. The new TGO/CIR profiles are correct in all cases, however, since they have been manually checked and modified to comply with the rules stated in the INM manual.

Table 15. TGO/CIR new profile validation – performance analysis

Aircraft	TGO Altitu	TGO Thru	TGO Spee	CIR Altitu	CIR Thrus	CIR Spee	TGO Altitude (	TGO Thrust (	TGO Speed (	CIR Altitude (	CIR Thrust (	CIR Speed (	Number of Outlie	CLASS
757RR	1.2588	6.4518	23.7764	1.6705	6.1069	23.6409	1	1	1	1	1	1	6	3
GASEPV	7.7663	18.8621	28.6158	3.0623	17.7301	28.7757	1	1	1	1	1	1	6	3
7478	2.8058	6.4451	0.61077	2.8098	5.9656	1.9477	1	1	1	1	1	1	6	3
7878R	3.1345	4.4027	2.4121	3.0163	3.6005	2.1409	1	1	1	1	1	1	6	3
757PW	1.4162	3.8269	23.3915	1.876	3.9109	23.2478	1	1	1	1	1	1	6	3
CNA208	0.76438	13.5054	10.1363	0.76505	13.4839	10.1313	1	1	1	1	1	1	6	3
737300	0.35585	12.6295	11.0537	1.3005	12.6422	10.9606	0	1	1	1	1	1	5	3
GASEPF	3.865	1.4302	1.8325	3.0907	2.7288	1.8013	1	0	1	1	1	1	5	3
737400	0.17499	8.6473	10.5778	0.31587	8.6443	10.7225	0	1	1	0	1	1	4	3
7773ER	1.4004	1.2003	9.9733	1.4014	1.6407	10.8757	1	0	1	1	0	1	4	3
A7D	0.53831	26.532	36.5818	0.54065	25.0798	36.3189	0	1	1	0	1	1	4	3
BEC58P	7.4548	2.0912	1.8095	3.0837	2.6292	1.3411	1	0	1	1	0	1	4	3
CNA441	0.15572	18.3239	14.2388	0.15548	18.0942	14.1991	0	1	1	0	1	1	4	3
COMSEP	4.1603	1.58	2.4458	3.0871	2.4255	2.0209	1	0	1	1	0	1	4	3
DO228	0.26025	14.6117	17.2204	0.26292	14.5334	17.2079	0	1	1	0	1	1	4	3
DO328	0.10922	25.0217	7.7012	0.11378	24.6641	7.6842	0	1	1	0	1	1	4	3
F4C	0.42517	8.4394	13.6394	0.3308	8.1031	13.5413	0	1	1	0	1	1	4	3
PA42	0.3981	8.4198	23.1615	0.40376	8.2643	23.1044	0	1	1	0	1	1	4	3
EMB170	16.3683	1.9768	0.49652	16.8534	3.6723	0.65458	1	0	0	1	1	1	4	3
EMB175	16.6533	1.9787	0.50487	17.1477	3.7079	0.65679	1	0	0	1	1	1	4	3
CONCRD	1.4136	1.1503	0.52184	1.1318	1.1155	0.56848	1	0	0	1	0	1	3	3
GIIB	0.61549	28.9136	0.41479	0.61563	28.4992	0.41351	0	1	0	1	1	0	3	3
737700	1.2638	4.9429	0.48432	0.003227	0.89591	0.18952	1	1	0	0	0	0	2	2
747400	0.14553	1.052	0.82844	0.13867	1.0054	0.81918	0	0	1	0	0	1	2	2
CNA182	0.42672	24.3567	0.27829	0.42773	24.3432	0.27767	0	1	0	0	1	0	2	2
EMB190	8.0991	1.3891	0.26882	8.2999	2.4714	0.28696	1	0	0	1	0	0	2	2
EMB195	7.5999	1.3287	0.2575	7.7819	2.3565	0.26537	1	0	0	1	0	0	2	2
GII	0.46618	22.1061	0.26734	0.46644	21.799	0.26649	0	1	0	0	1	0	2	2
GIV	0.25405	32.0278	0.11198	0.25372	31.7593	0.11157	0	1	0	0	1	0	2	2
GV	0.42477	32.3935	0.25677	0.42508	31.8307	0.25616	0	1	0	0	1	0	2	2
MU3001	0.31299	10.2873	0.26152	0.23728	10.1786	0.29214	0	1	0	0	1	0	2	2
717200	0.033272	17.7628	0.32346	0.014823	0.31536	0.024225	0	1	0	0	0	0	1	2
737500	0.039278	1.0246	9.3699	0.019643	0.15805	0.034347	0	0	1	0	0	0	1	2
CNA172	0.60949	1.2432	0.38697	0.60877	1.2356	0.38603	0	0	0	1	0	0	1	2
DC3	0.1038	0.55899	0.50377	0.11346	0.55041	0.53487	0	0	0	0	0	1	1	2
1900D	0.093424	0.35253	0.039165	0.0049712	0.020242	0.046016	0	0	0	0	0	0	0	1
707320	0.0067908	0.0024614	0.00064674	0.0063304	0.0014562	0.00095019	0	0	0	0	0	0	0	1
707QN	0.0067908	0.0024614	0.00064674	0.0063304	0.0014562	0.00095019	0	0	0	0	0	0	0	1
720B	0.050674	0.26129	0.031031	0.0052844	0.0021059	0.030858	0	0	0	0	0	0	0	1
727100	0.013473	0.001616	0.036057	0.01374	0.0024865	0.035835	0	0	0	0	0	0	0	1
727D15	0.0029818	0.0030262	0.01748	0.0032144	0.0025467	0.017376	0	0	0	0	0	0	0	1
727D17	0.0072623	0.0047558	0.022315	0.0068498	0.0058001	0.022251	0	0	0	0	0	0	0	1
727EM1	0.013473	0.001616	0.036057	0.01374	0.0024865	0.035835	0	0	0	0	0	0	0	1
727EM2	0.0029818	0.0030262	0.01748	0.0032144	0.0025467	0.017376	0	0	0	0	0	0	0	1

The validation analysis was conducted for each of 146 aircraft individually by comparing the performance, emissions, and noise results between existing and updated TGO/CIR, and then grouped to the three classes based on the comparison differences. An example of a Class 1 aircraft (1900D) is shown below. There are no noticeable discrepancies to be found here.



Aircraft	TGO Altitu	TGO Thru	TGO Spee	CIR Altitu	CIR Thrus	CIR Spee	TGO Altitude (	TGO Thrust (	TGO Speed (	CIR Altitude (	CIR Thrust (	CIR Speed (	Number of Outlie	CLASS
1900D	0.093424	0.35253	0.039165	0.0049712	0.020242	0.046016	0	0	0	0	0	0	0	1

Figure 24. Class 1 aircraft—performance analysis.

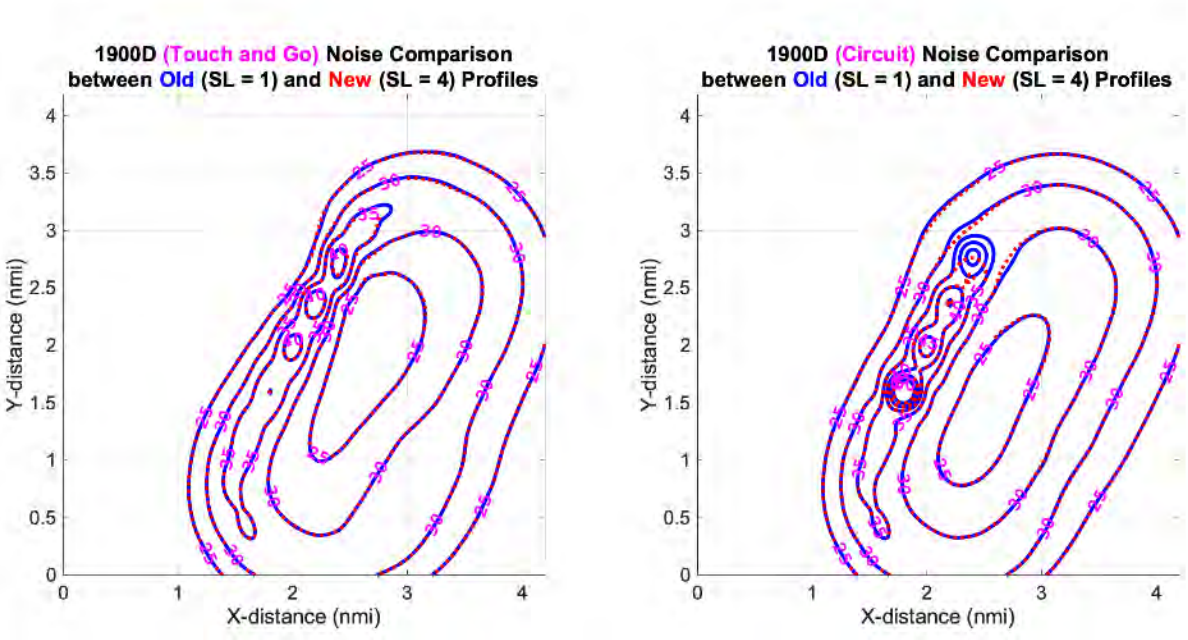
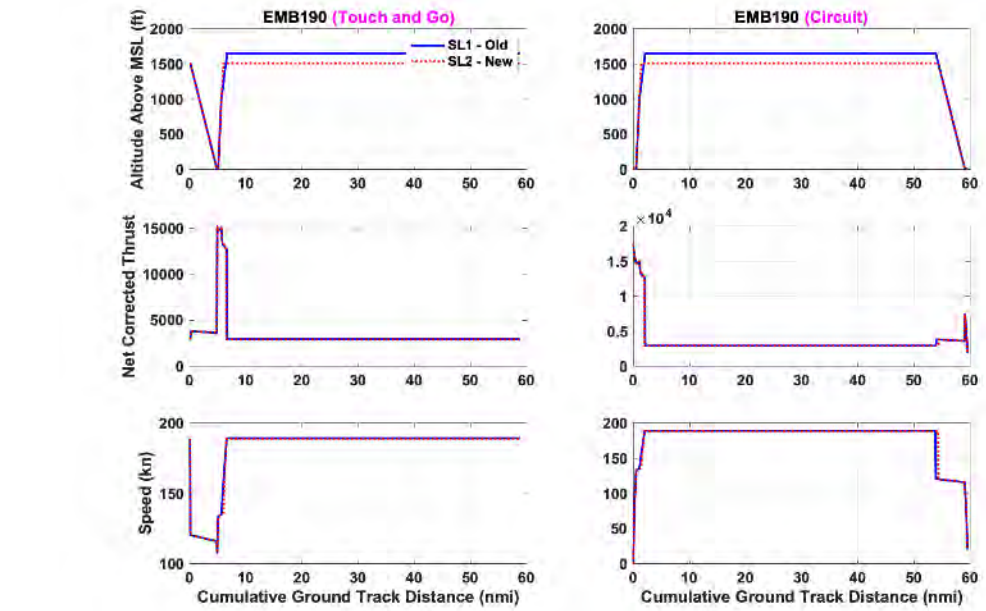


Figure 25. Class 1 aircraft—noise analysis.

An example of a Class 2 aircraft (EMB190) is shown below. As it can be seen, there are only two discrepancies here, namely the altitude for both the TGO and CIR profiles. This is due to the EMB190 overclimbing for the old TGO/CIR profiles. Discrepancies in terms of noise are thus present and expected because of the performance differences.



Aircraft	TGO Altitu	TGO Thru	TGO Spee	CIR Altitu	CIR Thrus	CIR Spee	TGO Altitude	TGO Thrust	TGO Speed	CIR Altitude	CIR Thrust	CIR Speed	Number of Outlie	CLASS
EMB190	8.0991	1.3891	0.26882	8.2999	2.4714	0.28696	1	0	0	1	0	0	2	2

Figure 26. Class 2 aircraft—performance analysis.

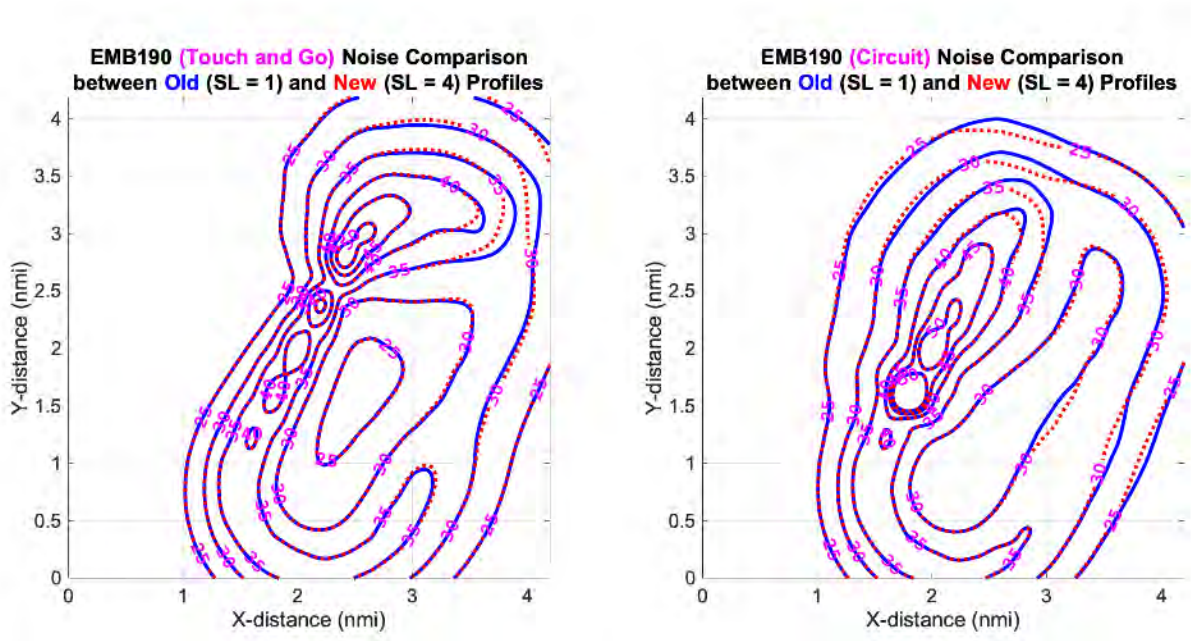
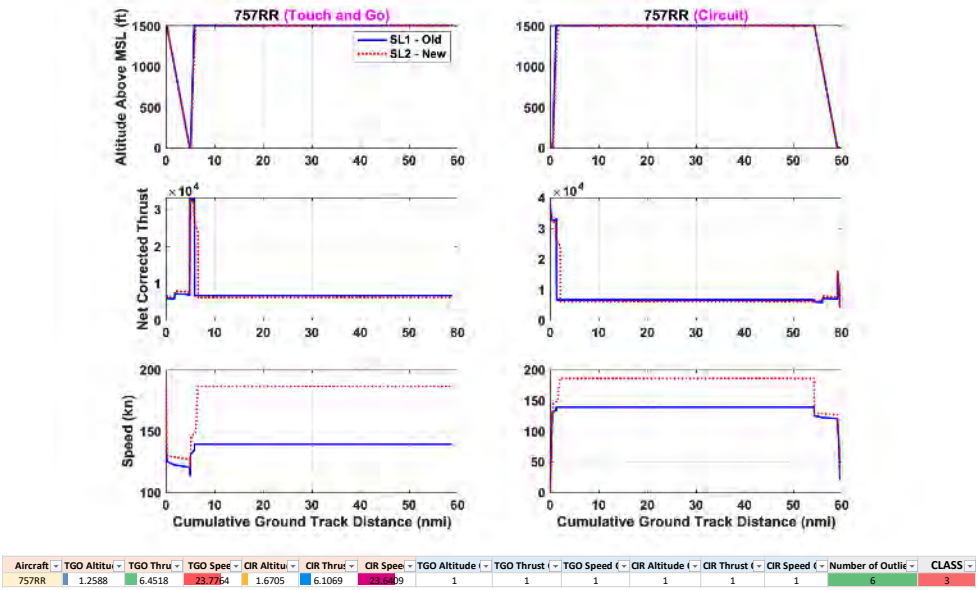


Figure 27. Class 2 aircraft—noise analysis.

An example of a Class 3 aircraft (757RR) is shown below. As can be seen, there are a total of four discrepancies, namely the thrust and speed for both the TGO and CIR profiles. This is mainly due to the old TGO/CIR profiles not having the updated

parameter values corresponding to the current version of the approach/departure profiles. Discrepancies in terms of noise are thus present and expected because of the performance differences.



Aircraft	TGO Altitu	TGO Thru	TGO Spee	CIR Altitu	CIR Thrus	CIR Spee	TGO Altitude	TGO Thrust	TGO Speed	CIR Altitude	CIR Thrust	CIR Speed	Number of Outlie	CLASS
757RR	1.2588	6.4518	23.7764	1.6705	6.1069	23.6409	1	1	1	1	1	1	6	3

Figure 28. Class 3 aircraft—performance analysis.

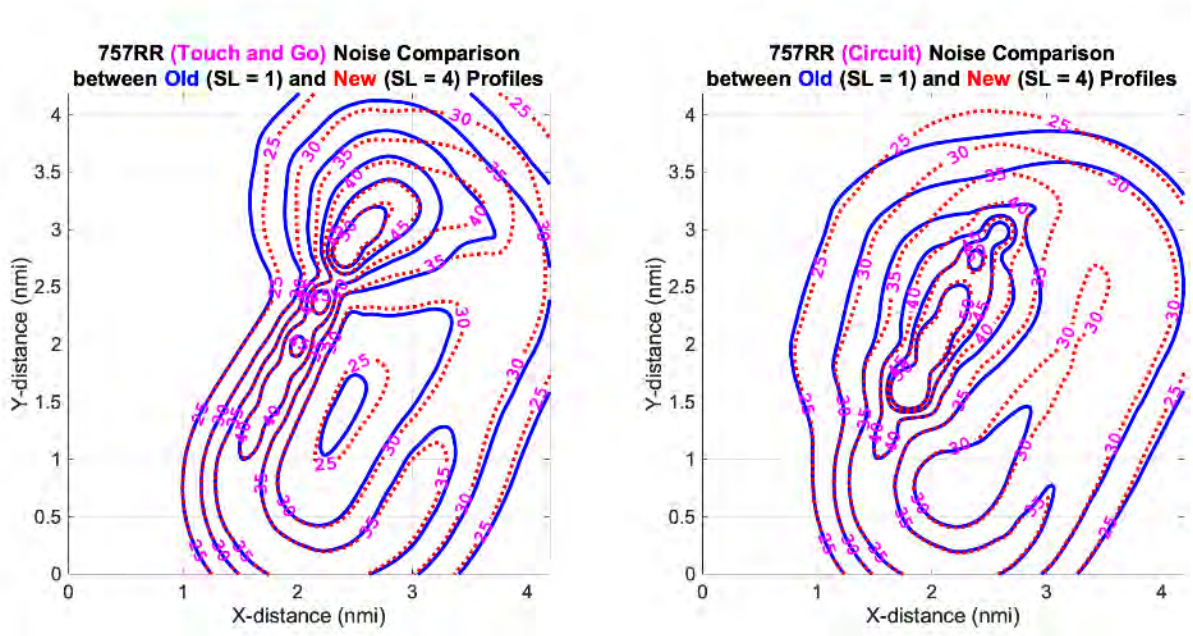


Figure 29. Class 3 aircraft—noise analysis.

Finally, it can be concluded that the noise class categorization is highly correlated with the performance class categorization, as it can be seen in the confusion matrix below in Table 16 which shows how many aircraft belong to which

performance/noise class combination. Many aircraft that are in Class 1 of the performance result (good agreement) and in Class 2 of the noise result (minor differences) have modified or new profiles as the final profile.

Table 16. TGO/CIR profiles performance/noise confusion matrix

Performance\Noise Classes	Noise = 1	Noise = 2	Noise = 3
Performance = 1	95	17	1
Performance = 2	6	5	2
Performance = 3	5	8	9

Investigation on EUROCONTROL recommended ANP implementations

Investigation of BADA4 arrival fuel consumption inconsistencies

The inconsistency identified is that for some aircraft type, such as the A300-622 shown in Figure 30, there is a reduction in the BADA4 fuel consumption levels near five nmi from the touchdown (the grey curve on the left plot), which does not correlate with the thrust variation (the grey curve on the right plot, which indicates thrust is increasing in that region).

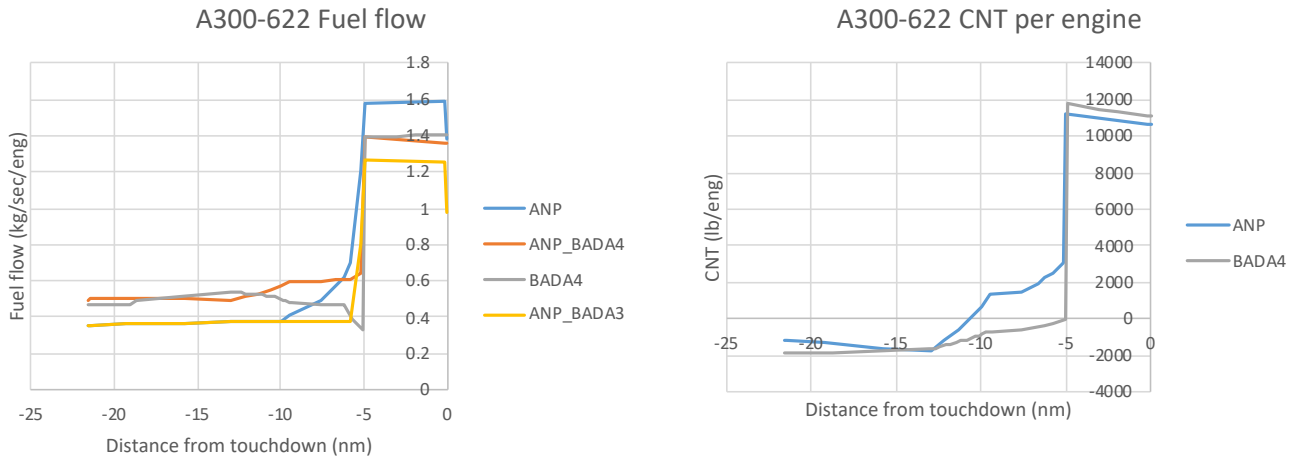


Figure 30. Example of arrival fuel consumption inconsistency.

Investigation of this problem starts from the STANDARD procedural approach profile of A300-622, which is shown below in Figure 31. It can be seen from Figure 31 that before touchdown, the landing approach procedure consists of two major sections: steps 1–6 belong to the idle descent section, while steps 7–8 belong to the non-idle section. After a matching processing between AEDT performance report and the approach profile, it can be identified that the BADA4 fuel consumption reduction region in the left plot of Figure 30 exactly matches the step 6 in the approach profile. As shown in Figure 32, the start and end of the reduction region correspond exactly to the two altitudes at the beginning and end of step 6. This leads to a crucial conclusion that the fuel consumption reduction region belongs to idle descent.

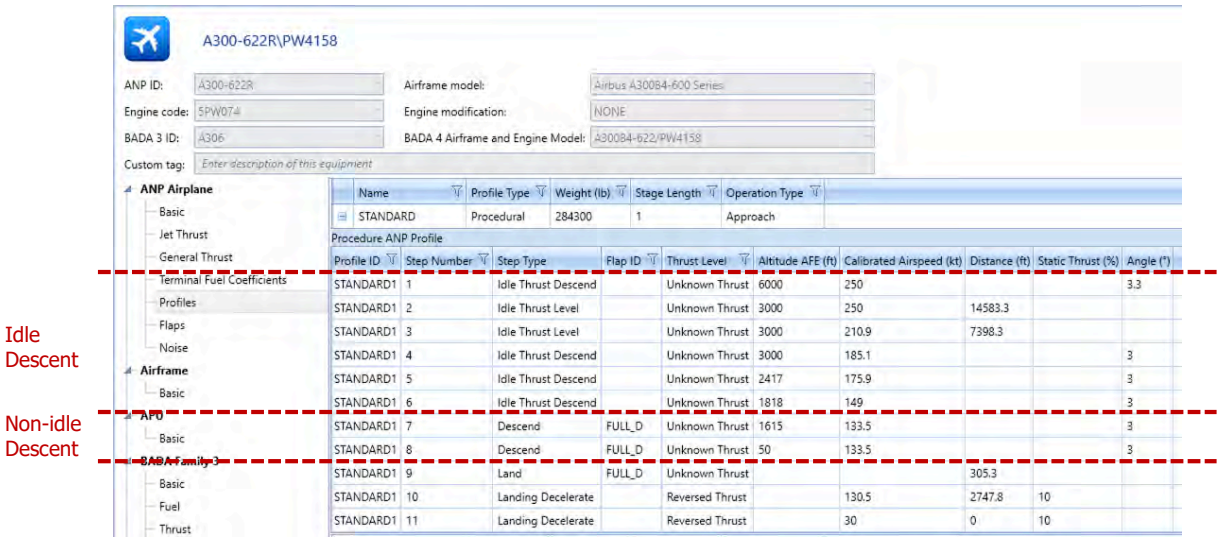


Figure 31. A300-622 STANDARD approach profile.

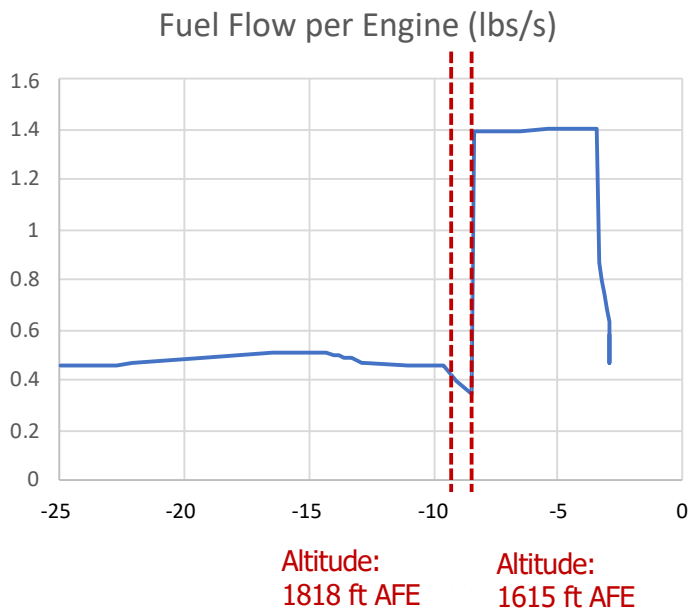


Figure 32. BADA4 fuel consumption for A300-622.

In the BADA4 fuel consumption equations, idle descent and non-idle descent are governed by two completely different equations.

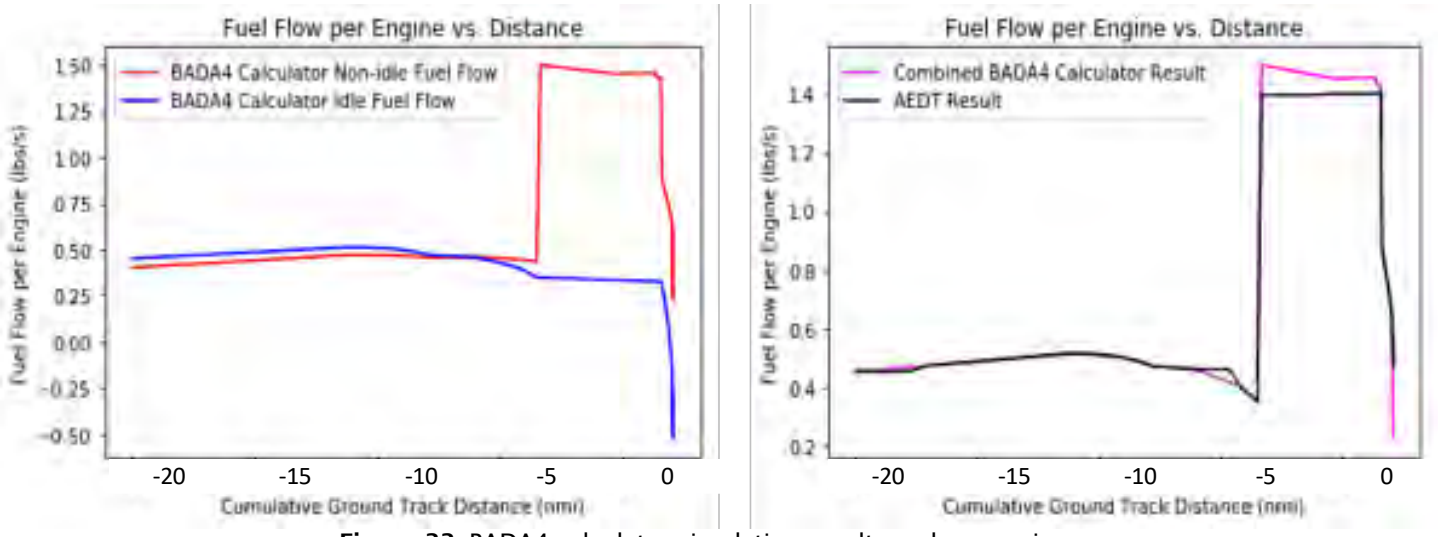


Figure 33. BADA4 calculator simulation results and comparisons.

After identifying the cutoff line between idle and non-idle fuel consumption regions, a simulation code called "BADA4 calculator is utilized to further verify the result. In the BADA4 calculator, the same BADA4 performance equations and coefficients are used to simulate an aircraft's performance for departure or approach operations. By implementing the BADA4 equations, it serves as a way to conduct model validation and verification for AEDT's result. Figure 33 shows the simulation results from the BADA4 calculator for A300-622 STANDARD approach and a comparison to AEDT. The left plot of Figure 33 contains both idle fuel flow and non-idle fuel flow results from BADA4 calculator throughout the entire approach process. The magenta curve in the right plot of Figure 33 combines the idle fuel flow and non-idle fuel flow results such that it matches the approach profile in Figure 31. In this combined BADA4 calculator result, the idle fuel flow result is used before the around five nmi location on the x-axis; the non-idle fuel flow result is used after this point. It can be observed that right before the transition point, since idle fuel calculation is used, the reduction in fuel consumption in fact matches the BADA4 fuel flow calculation correctly. In the right plot of Figure 33, fuel consumption result from the AEDT performance report is also plotted as a comparison. There is a very good agreement between AEDT and BADA4 calculation. Therefore, the conclusion from this investigation is that the reduction in the BADA4 fuel consumption correctly reflects the real BADA4 fuel flow relationships.

A follow-up investigation here is that, in Figure 30, why does the ANP_BADA4 fuel flow result not have such a reduction since it also used BADA4 idle fuel flow methods? For this question, we also use the BADA4 calculator to gain insights on the possible causes. The ANP_BADA4 fuel flow result consists of ANP thrust and BADA4 fuel flow calculation, that is, the BADA4 fuel consumption coefficient uses the ANP thrust coefficient C_T in the calculation. Note that only the BADA4 non-idle fuel consumption coefficient is affected by C_T. The left plot of Figure 34 shows the non-idle and idle fuel flow variations during the arrival operation of A300-622, with a dashed line indicating the transition between idle and non-idle phases in the arrival profile. A possible reason here is that, in the ANP_BADA4 result, transition from idle fuel flow to non-idle fuel flow happened earlier than the time point in the approach profile. The right part of Figure 34 shows two different combined idle and non-idle fuel flow results from BADA4 calculator, and one corresponds to the normal transition and the other corresponds to early transition. A reduction in the fuel flow is observed when the transition happens at the correct place near 5 nautical miles before touchdown. On the other hand, if the transition happens earlier or no transition from idle to non-idle happens during the arrival operation (only non-idle fuel flow), there is no such reduction. It is worth mentioning that the ANP_BADA4 fuel flow result in Figure 30 has a very similar shape to the non-idle fuel flow curve in Figure 34.

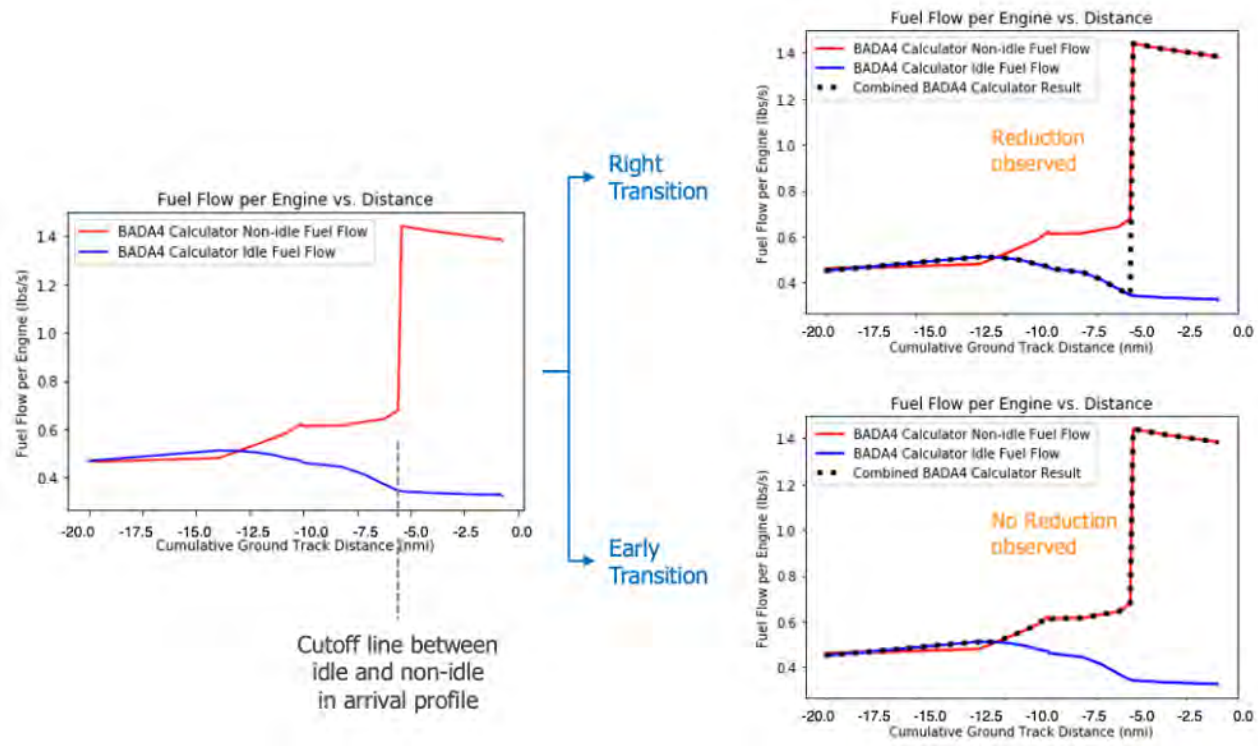


Figure 34. BADA4 calculator simulations for ANP_BADA4 result.

Investigation of arrival thrust bump near flap transition

It was found that for some aircraft such as the 737-700, the BADA4 idle thrust model shows a small, unexpected bump in the transition near 10 nmi from touchdown. In Figure 35, the grey curve is the BADA4 idle thrust variation during 737-700 approach operation. The thrust bump near 10 nmi was deemed as a possible anomaly in the simulation process. After conducting AEDT tests on many other aircraft, it was identified that this phenomenon happens not only with the 737-700, but also with a few other aircraft. An initial inspection into the arrival profile shows that the unexpected bump normally happens near a flap transition region for some aircraft. However, a deeper investigation into the thrust calculation process is required to further uncover the possible reasons behind this phenomenon.

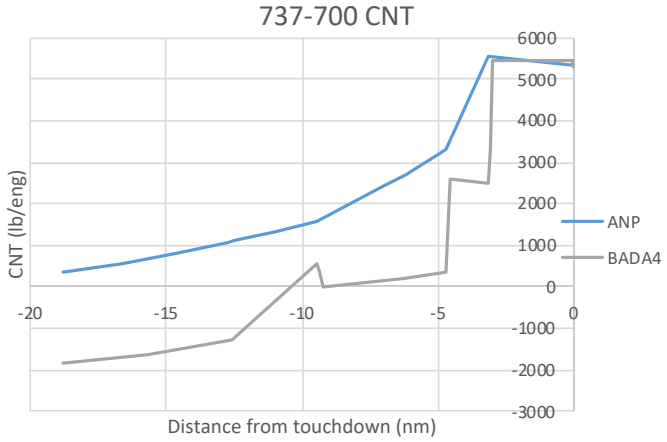


Figure 35. Unexpected bump of idle thrust in the flap transition region.

In BADA4, arrival thrust is calculated by the total energy model (TEM), given by

$$(T - D) \cdot V_{TAS} = mg_0 \frac{dh}{dt} + mV_{TAS} \frac{dV_{TAS}}{dt}$$

where T is thrust, D is drag, and V_{TAS} is the true airspeed. The right-hand side of the TEM equation has two components, namely the rates of change for potential energy and kinetic energy, respectively. In TEM, thrust is determined jointly by multiple factors, such as airspeed, rate of climb, acceleration, mass, and drag. Drag is a main influencer in the calculation of thrust, yet any other factor can contribute to the unexpected bump. In the following process, we use BADA4 calculator and AEDT intermediate performance outputs to investigate the problem.

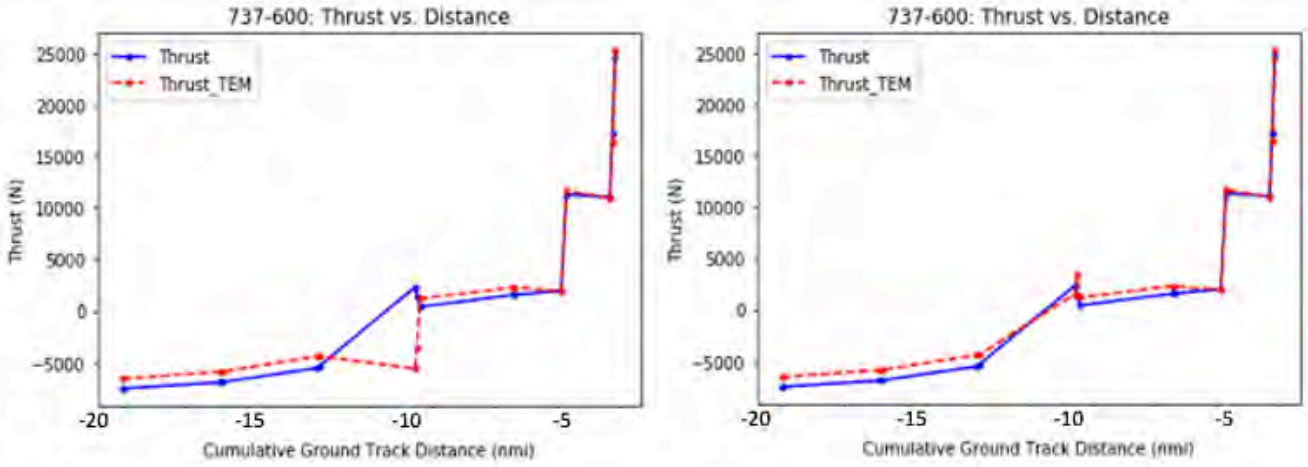


Figure 36. Arrival thrust comparison between AEDT and BADA4 calculator, with (left) and without (right) full TEM.

The left plot of Figure 36 shows a thrust comparison between AEDT and the BADA4 calculator for 737-600 arrival operation. The blue curve is from AEDT performance report; the red curve is the result calculated by the BADA4 calculator on the same operation with the full TEM implemented. One can observe from the left plot of Figure 36 that when the complete version of TEM is implemented, there is a disagreement between AEDT and the BADA4 calculator, especially in the "controversial region" around 10 nmi. Starting from the point around -13 nmi, the thrust result from the BADA4 calculator first decreases, then climbs sharply right before -10 nmi. In contrast, the AEDT result starts to climb right after -13 nmi and has a reduction in

the controversial region. A first assumption here is that the difference is caused by AEDT's incomplete implementation of TEM. With this assumption, an action was taken on the BADA4 calculator simulation, in which the items in TEM are omitted one by one to see their influences in thrust calculation. The right plot of Figure 36 shows the BADA4 calculator result when the entire change in kinetic energy term in the TEM is omitted. Although slight differences still exist between the BADA4 calculator and AEDT, the trend is much closer compared to the full implementation of TEM in the BADA4 calculator.

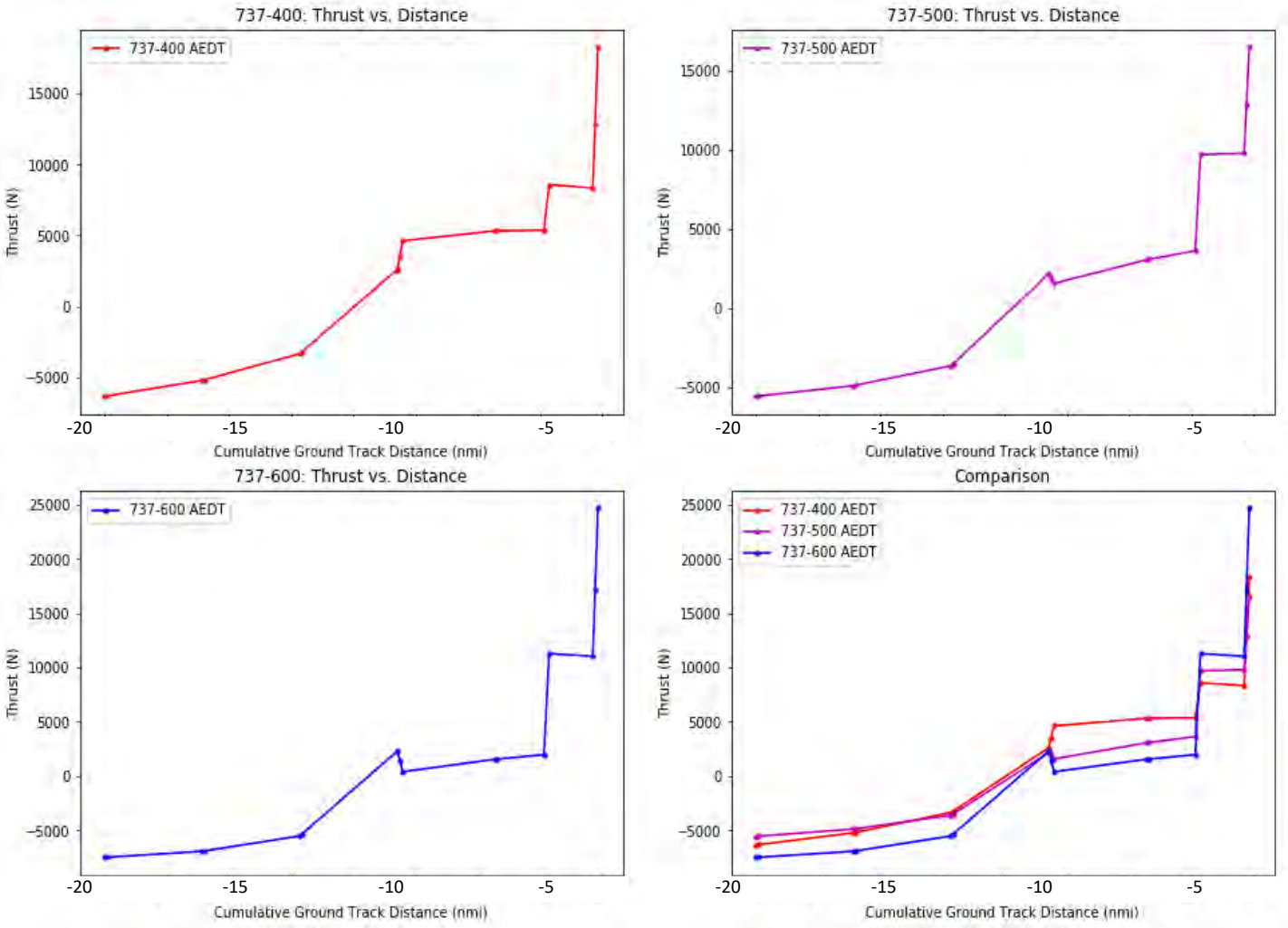


Figure 37. BADA4 thrust comparison between the 737 family.

After confirming that AEDT does not miss the kinetic energy term when calculating BADA4 idle thrust, a detailed comparison within the same aircraft family is utilized to further explore the role of the kinetic energy term and other factors in the unexpected bump. Figure 37 shows the BADA4 idle thrust comparison between three aircraft within the 737 family: 737-400, 747-500, and 737-600. The lower right plot of Figure 37 includes a direct comparison between the three aircraft. An interesting fact here is that, in the controversial region right before -10 nmi, three aircraft that have similar approach profiles show different characteristics. Compared to the 737-600, which has an obvious unexpected bump, the bump in the 737-500 becomes milder, as the reduction after the peak is smaller in magnitude. 737-400, however, does not have the bump at all as the idle thrust only increases in the controversial region. The three different patterns within the same aircraft family is worthy of a closer look into the intermediate results for these three aircraft.

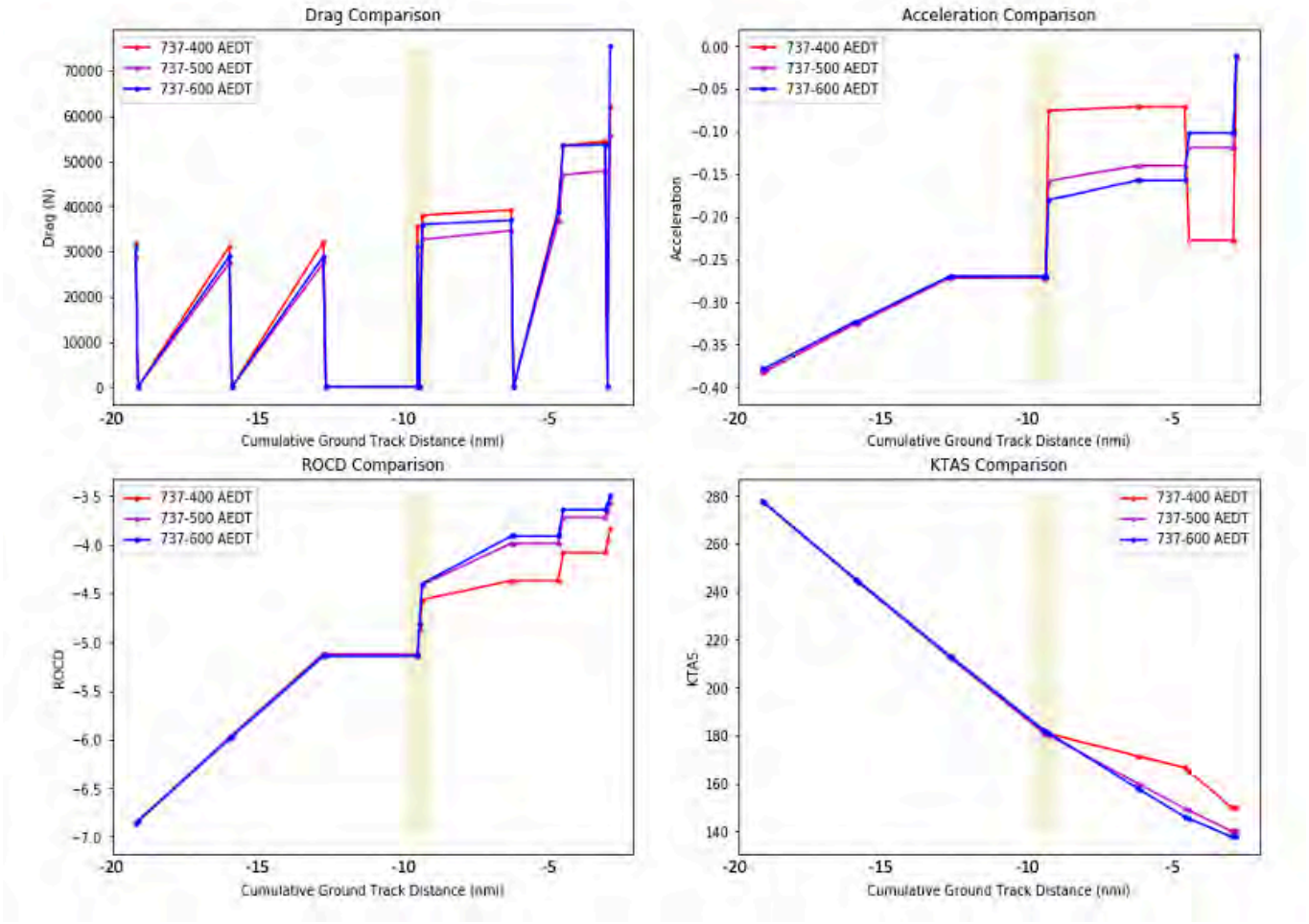


Figure 38. TEM components comparison between the 737 family.

Figure 38 shows the variations of four intermediate outputs for 737-400, 737-500, and 737-600: drag, acceleration, rate of climb (ROCD), and true airspeed (KTAS), which are all key terms in the TEM. In addition, the controversial region is also marked as a shaded region. It is observed that in the controversial region, all four intermediate outputs change trend to some extent. Among the four intermediate outputs, it is observed that acceleration is the most significant driving factor behind the thrust difference within the same 737 family. 737-400 has the smallest acceleration in absolute value, which leads to the only increasing thrust case among the three aircraft. On the other hand, 737-500 and 737-600 transit to more negative accelerations in the controversial region. In TEM, they result in a thrust reduction after the transition and cause a thrust bump. In conclusion, in the BADA4 idle thrust model, the unexpected thrust bump in the transition near 10 nmi from touchdown is mainly affected by the acceleration level of the aircraft.

Profile editor testing

System testing and evaluation of AEDT's user-defined profile editor feature is conducted with the intent to verify both the functionality of graphical user interface (GUI) elements and the validity of performance results. Initial analysis consists of tests examining AEDT's response to conventional and unorthodox usage of the profile editor GUI shown in Figure 39, whereby parameters within both copies of existing profiles and newly generated alternatives are varied in an attempt to evaluate the feature. Existing arrival and departure profiles face alterations to weight, flap ID, airspeed, thrust and angle parameters across a range of values—both practical and infeasible—to discern AEDT's response to GUI alterations. Additional profiles are created with random step type combinations to confirm AEDT's capacity to reject and/or display warnings when

detecting irreconcilable profile components. This battery of tests demonstrates the full functionality of the profile editor GUI and reveals only a single bug (where modifications to thrust level under the BADA-4 tab result in program crashes) which has been reported and resolved.

Demonstrating GUI functionality facilitates the next round of testing, focused on ensuring the presence of accurate performance disparities across different profiles. Delayed deceleration approach (DDA) (http://atmseminar.org/seminarContent/seminar12/papers/12th_ATM_RD_Seminar_paper_119.pdf) is chosen as the candidate for comparison based on its ease of implementation within the profile editor, its clearly discernible differences across performance, noise, and emissions during real-word tests, and its potential for implementation across a number of U.S. airports.

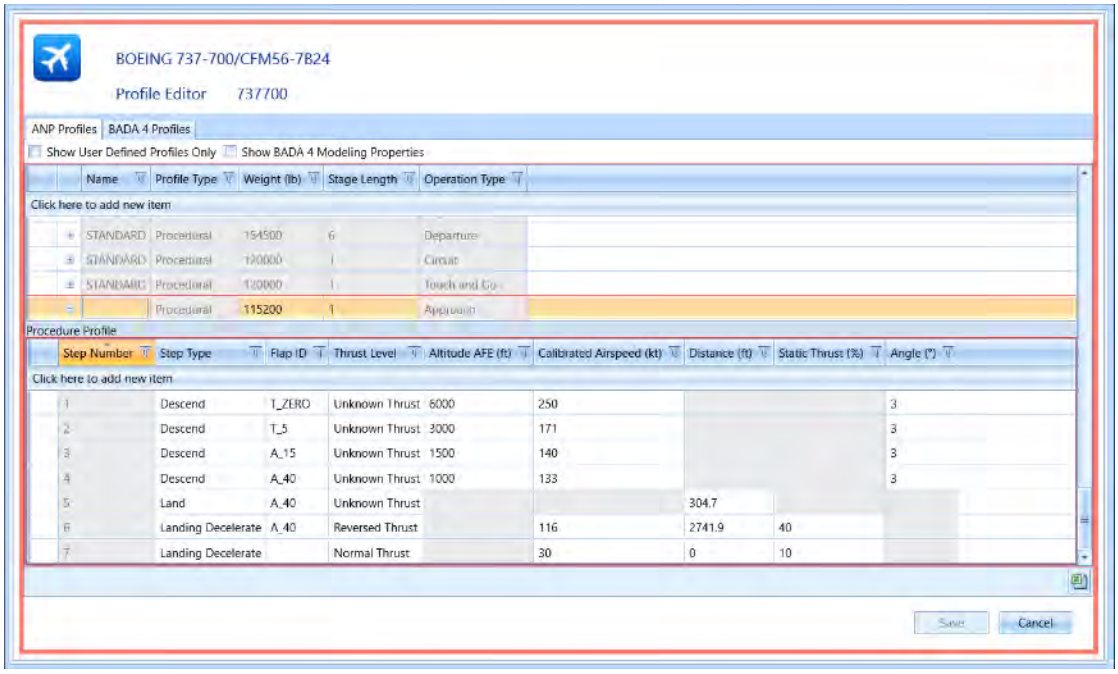


Figure 39. User-defined profile editor GUI.

Implementation of the DDA profile is preceded by the definition of a standard arrival procedure for comparison, shown in Table 17. Whereas the standard profile is quick to lower speed and deploy flaps, DDA profiles maintain their initial airspeed and delay flap deployment to reduce drag and engine power requirements. This results in a matching trajectory between both approaches and a linearly shifted (i.e. delayed) groundspeed profile in the latter case. Research thus far reveals varying degrees of correlation between speed and flap deployment on fuel savings and noise reductions across different aircraft but generally prioritizes the role of the latter in the B777 and A320 lines. The A320-211 is chosen as the test aircraft based on its prevalence in the industry alongside the KATL airport owing to its potential savings from DDA utilization.

Table 17. A320-211 standard arrival procedure

Step Number	Step Type	Flap ID	Thrust Level	Altitude—AFE (ft)	Airspeed (kts)	Distance (ft)	Static Thrust (%)	Angle (°)
1	Idle Thrust Descend	A_00	Idle Thrust	6000	248.93			3
2	Idle Thrust Level	A_00	Idle Approach	3000	249.5	5437		
3	Idle Thrust Level	A_01	Idle Approach	3000	187.18	3671		
4	Idle Thrust Level	A_05	Idle Approach	3000	174.66	5209		
5	Level	A_15	Unknown Thrust	3000	151.41	25000		
6	Descend	A_30	Unknown Thrust	3000	139.11			3
7	Land	A_30	Idle Approach			393.8		
8	Landing Decelerate		Reversed Thrust		139	3837.5	40	
9	Landing Decelerate		Normal Thrust		30	0	10	

Step 5 (highlighted in red) is absent in AEDT's default arrival profile for the A320 and manually added to create a flexible level step simulating the effects of traditional and delayed approach. The same profile feature is recast as Step 2 in Table 18, which models extended time at higher altitudes to mimic a delayed approach.

Table 18. A320-211 DDA procedure

Step Number	Step Type	Flap ID	Thrust Level	Altitude—AFE (ft)	Airspeed (kts)	Distance (ft)	Static Thrust (%)	Angle (°)
1	Idle Thrust Descend	A_00	Idle Thrust	6000	248.93			3
2	Level	A_00	Idle Approach	3000	249.5	25000		
3	Idle Thrust Level	A_00	Idle Approach	3000	249.5	5437		
4	Idle Thrust Level	A_01	Idle Approach	3000	187.18	3671		
5	Idle Thrust Level	A_05	Idle Approach	3000	174.66	5209		
6	Descend	A_30	Unknown Thrust	2817	139.11			3
7	Land	A_30	Idle Approach			393.8		
8	Landing Decelerate		Reversed Thrust		139	3837.5	40	
9	Landing Decelerate		Normal Thrust		30	0	10	

Figure 40 examines the noise contours of the standard and DDA A320 profiles and reveals a noticeable reduction in noise across all decibel (dB) ranges. The 5–10% difference in contour area present across all noise levels conforms to values demonstrated in literature and confirms the accuracy of the profile editor with regards to its impact on noise calculations.

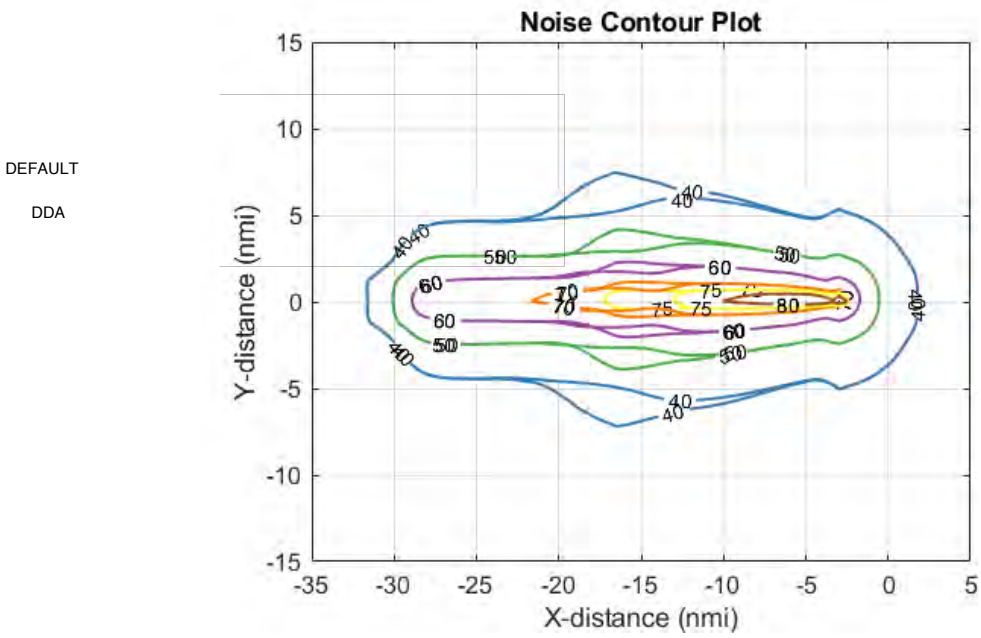


Figure 40. Noise contours for default and DDA profiles.

Figure 41 shows the general performance characteristics of default approach and DDA and reveals an identical trajectory and different ground speeds. These observations are once again validated by the literature, which aims to maintain a consistent trajectory and shift speed alterations to produce the noise reductions shown above.

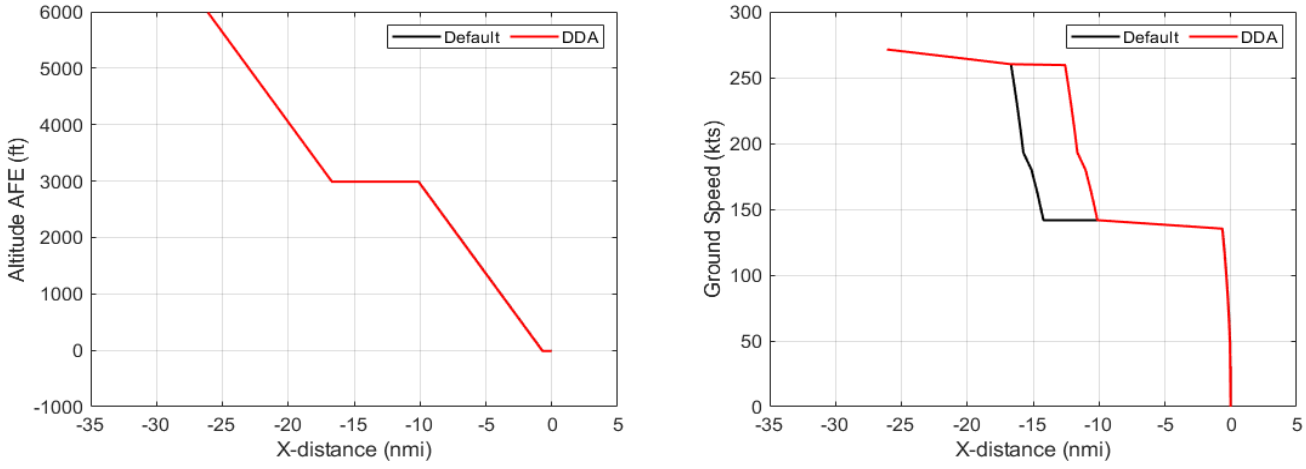


Figure 41. Default and DDA trajectory and speed.

Finally, Figure 42 displays centerline noise and compares fuel burn across each phase of arrival between both profiles. The graphs fall in line with earlier results, reiterate the overall reduction in noise, and confirm the reduction in DDA emissions.

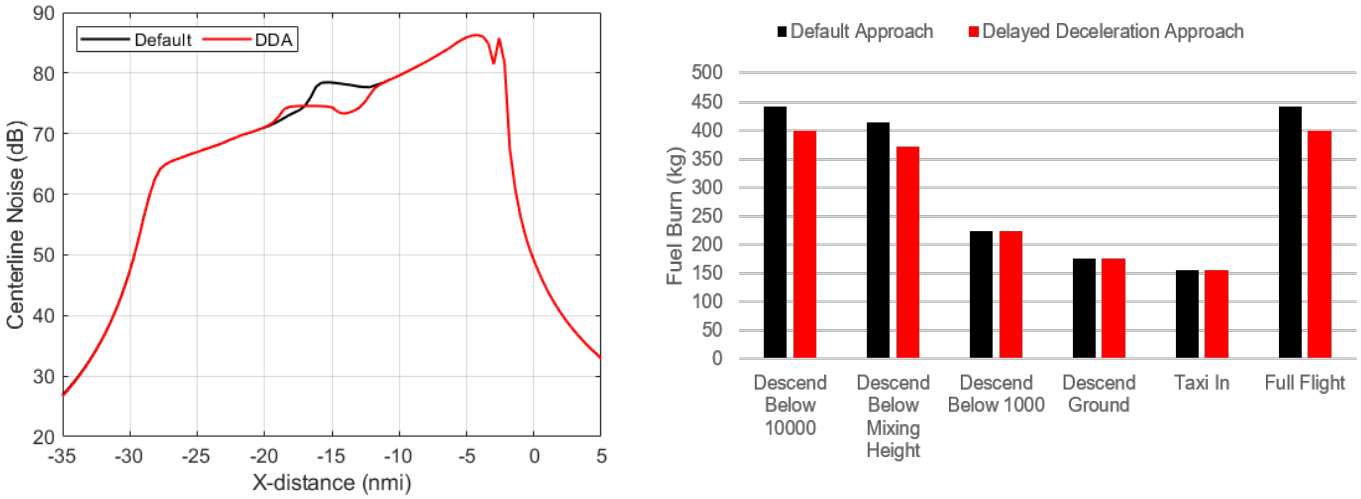


Figure 40. Default and DDA centerline noise and emissions.

The successful generation of a DDA profile in accordance with the literature validates the functionality of the profile editor with respect to the modeling of conventional profiles and moderate deviations from standard procedures. The final set of tests models large departures from standard profiles and incorporates impractical or infeasible profile elements to determine AEDT's response to unexpected inputs and potential fringe cases. The 737-800 and A320-211 aircraft are used to model variations in weight, step sequence, ground speed, and flap orientation across departures and arrivals at KATL.

Figure 43 and Figure 44 show the impact of weight variation on the 737-800's arrival and departure performance respectively. Attempts to input negative or zero weight values during profile creation cause errors as expected; however, AEDT is robust in handling all positive weight values and generates accurate performance results across a wide range of weights. Values between 50,000–1,000,000 lbs are input to examine AEDT's response and none cause errors despite their infeasibility.

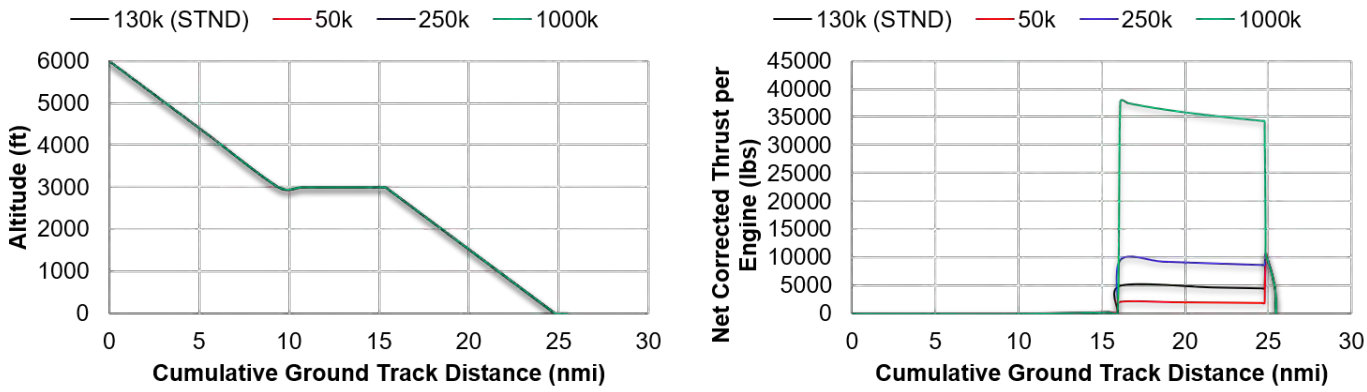


Figure 41. Comparing B737-800 arrival weights.

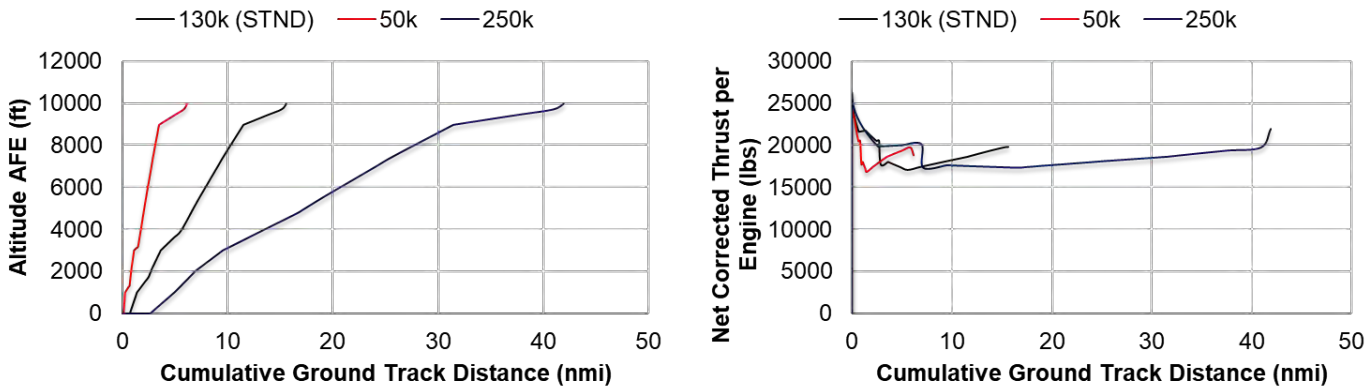


Figure 42. Comparing B737-800 departure weights.

Alterations in weight are followed by modifications to arrival and departure steps to introduce new elements. The standard arrival profile is changed to begin with landing steps (as opposed to the default idle-step descent) and departure is modified to exhibit decreases in altitude and begin with climb and acceleration steps (rather than the default takeoff steps). All of these changes yield errors upon attempting to run the created profiles, with the log files indicating that ANP procedures can only begin with level flight, descent, takeoff ground roll, or cruise climb and cannot experience altitude decreases during climb and increases during descent.

Table 19 and Figure 45 compare the standard A320-211 departure profile to its modified counterparts, where flap settings and ground speeds are modified beyond conventional boundaries. AEDT demonstrates robustness in both cases, managing to capture the slight differences owing to flap variation and model a reasonable interpretation of performance owing to a climb speed of 3000 kts without any errors.

The range of tests examining AEDT's response to variations in weight, step types, flap settings, and ground speeds indicates the profile editor's capacity to handle both theoretical and physical impossibilities. Inputting exceedingly high speeds and flap settings continue to generate fairly accurate results and attempts at altering fundamental parameters such as departure or arrival sequencing and using negative weight causes errors with clear explanations in the log file.

Table 19. Comparing A320-211 departure flap and ground speed settings

Step Number	Step Type	Flap ID	Thrust Level	Altitude—AFE (ft)	Airspeed (kts)	Climb Rate (ft/min)
1	Takeoff	1+F	Max Takeoff		0	
2	Climb	1+F	Max Takeoff	1000		
3	Accelerate	1+F	Max Takeoff		186.2	1150.5
4	Accelerate	1	Max Takeoff		208.1	1300.7
5	Climb	ZERO	Max Climb	3000		
6	Accelerate	ZERO/1+F	Max Climb		3000/250	1230.7
7	Climb	ZERO/1+F	Max Climb	5500		
8	Climb	ZERO	Max Climb	7500		
9	Climb	ZERO	Max Climb	10000		

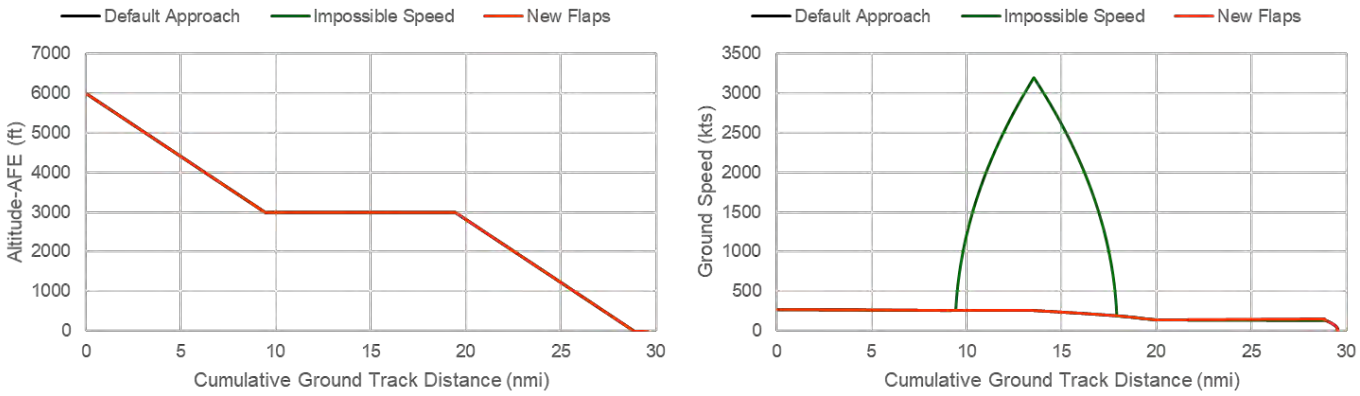


Figure 43. Comparing A320-211 departure flap and ground speed settings.

Considered in conjunction, the profile editor tests demonstrate GUI functionality and is capable of producing expected results for performance and noise metrics.

Emissions Dispersion Computational Efficiencies

Background
Previous investigations have shown that emissions metrics in AEDT are more computationally expensive to run relative to other metrics. Georgia Tech was tasked to investigate whether this feature persisted and whether previous recommendations were still valid.

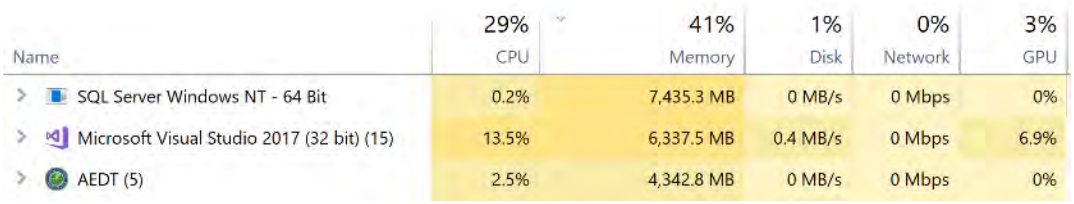
Approach
Two-step process:

1. Run study cases on a local machine
   STUDY_DULLES was used to test if the feature persisted for one emissions dispersion and one emissions metric. Run times were documented and the study was backed up to a server.
2. Run the study on a different machine that enables Visual Studio profiling tools
   The study was retrieved from the server and rerun on a different machine with Visual Studio 2017 installed. Visual Studio profiling tools were used to investigate performance issues pertaining to the study. Visual Studio diagnostic capabilities analyzed memory and CPU usage, among others.

Results
This analysis was done using the performance profiling tools in Visual Studio 2017 running the main ribbon GUI in debug mode from within Visual Studio on a workstation machine with a six core (12 threads) Xeon CPU with 48Gb of memory, and a 1 Tb NVme system drive.

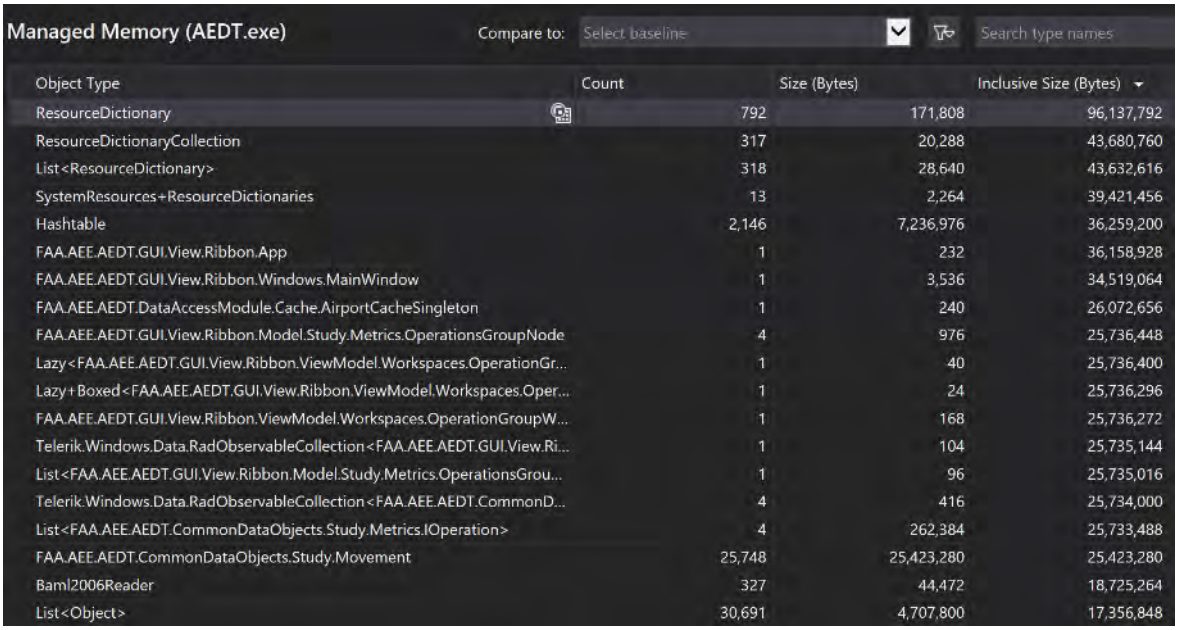
1. Emissions dispersion metric
   Figures 49 to Figure 51 show snapshots of the memory and CPU status timeline when running the emissions dispersion metric. The snapshots show memory usage and CPU performance while loading the study, processing operations, results extraction and generation, and finally results retrieval.
2. Emissions metric
   Figures 52 to Figure 57 show snapshots of the memory and CPU status timeline when running the emissions metric. The snapshots show memory usage and CPU performance while loading the study, processing operations, results extraction and generation, and finally results retrieval.

   Moreover, Figures 58 to Figure 61 show further investigation into event processing threads, which were carefully isolated to identify opportunities for computational run time reductions.



Name	29% CPU	41% Memory	1% Disk	0% Network	3% GPU
SQL Server Windows NT - 64 Bit	0.2%	7,435.3 MB	0 MB/s	0 Mbps	0%
Microsoft Visual Studio 2017 (32 bit) (15)	13.5%	6,337.5 MB	0.4 MB/s	0 Mbps	6.9%
AEDT (5)	2.5%	4,342.8 MB	0 MB/s	0 Mbps	0%

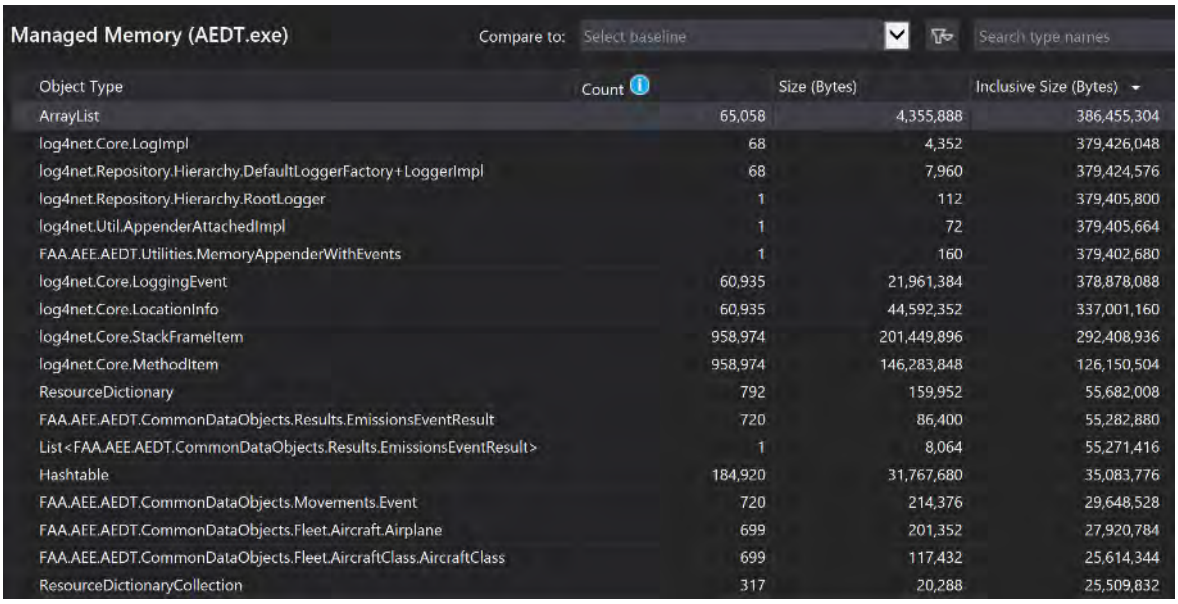
Figure 44. Task manager – Visual Studio CPU load.



Managed Memory (AEDT.exe) Compare to: Select baseline Search type names

Object Type	Count	Size (Bytes)	Inclusive Size (Bytes)
ResourceDictionary	792	171,808	96,137,792
ResourceDictionaryCollection	317	20,288	43,680,760
List<ResourceDictionary>	318	28,640	43,632,616
SystemResources+ResourceDictionaries	13	2,264	39,421,456
Hashtable	2,146	7,236,976	36,259,200
FAA.AEE.AEDT.GUI.View.Ribbon.App	1	232	36,158,928
FAA.AEE.AEDT.GUI.View.Ribbon.Windows.MainWindow	1	3,536	34,519,064
FAA.AEE.AEDT.DataAccessModule.Cache.AirportCacheSingleton	1	240	26,072,656
FAA.AEE.AEDT.GUI.View.Ribbon.Model.Study.Metrics.OperationsGroupNode	4	976	25,736,448
Lazy<FAA.AEE.AEDT.GUI.View.Ribbon.ViewModel.Workspaces.OperationGr...	1	40	25,736,400
Lazy+Boxed<FAA.AEE.AEDT.GUI.View.Ribbon.ViewModel.Workspaces.Oper...	1	24	25,736,296
FAA.AEE.AEDT.GUI.View.Ribbon.ViewModel.Workspaces.OperationGroupW...	1	168	25,736,272
Telerik.Windows.Data.RadObservableCollection<FAA.AEE.AEDT.GUI.View.Ri...	1	104	25,735,144
List<FAA.AEE.AEDT.GUI.View.Ribbon.Model.Study.Metrics.OperationsGrou...	1	96	25,735,016
Telerik.Windows.Data.RadObservableCollection<FAA.AEE.AEDT.CommonD...	4	416	25,734,000
List<FAA.AEE.AEDT.CommonDataObjects.Study.Metrics.IOperation>	4	262,384	25,733,488
FAA.AEE.AEDT.CommonDataObjects.Study.Movement	25,748	25,423,280	25,423,280
Baml2006Reader	327	44,472	18,725,264
List<Object>	30,691	4,707,800	17,356,848

Figure 45. Memory usage after loading study before run.



Managed Memory (AEDT.exe) Compare to: Select baseline Search type names

Object Type	Count	Size (Bytes)	Inclusive Size (Bytes)
ArrayList	65,058	4,355,888	386,455,304
log4net.Core.LogImpl	68	4,352	379,426,048
log4net.Repository.Hierarchy.DefaultLoggerFactory+LoggerImpl	68	7,960	379,424,576
log4net.Repository.Hierarchy.RootLogger	1	112	379,405,800
log4net.Util.AppenderAttachedImpl	1	72	379,405,664
FAA.AEE.AEDT.Utilities.MemoryAppenderWithEvents	1	160	379,402,680
log4net.Core.LoggingEvent	60,935	21,961,384	378,878,088
log4net.Core.LocationInfo	60,935	44,592,352	337,001,160
log4net.Core.StackFrameItem	958,974	201,449,896	292,408,936
log4net.Core.MethodItem	958,974	146,283,848	126,150,504
ResourceDictionary	792	159,952	55,682,008
FAA.AEE.AEDT.CommonDataObjects.Results.EmissionsEventResult	720	86,400	55,282,880
List<FAA.AEE.AEDT.CommonDataObjects.Results.EmissionsEventResult>	1	8,064	55,271,416
Hashtable	184,920	31,767,680	35,083,776
FAA.AEE.AEDT.CommonDataObjects.Movements.Event	720	214,376	29,648,528
FAA.AEE.AEDT.CommonDataObjects.Fleet.Aircraft.Airplane	699	201,352	27,920,784
FAA.AEE.AEDT.CommonDataObjects.Fleet.AircraftClass.AircraftClass	699	117,432	25,614,344
ResourceDictionaryCollection	317	20,288	25,509,832

Figure 46. Memory snapshot while processing operations.

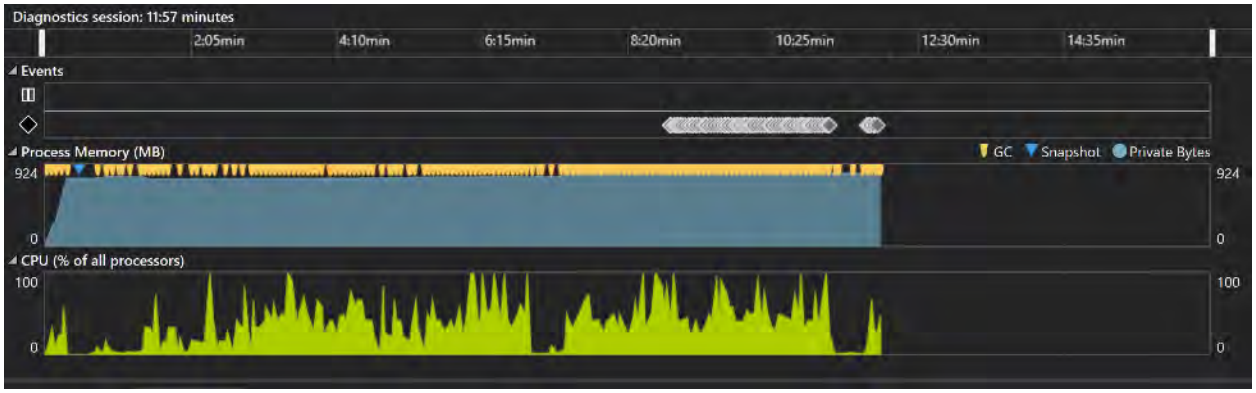


Figure 47. Processing operations.

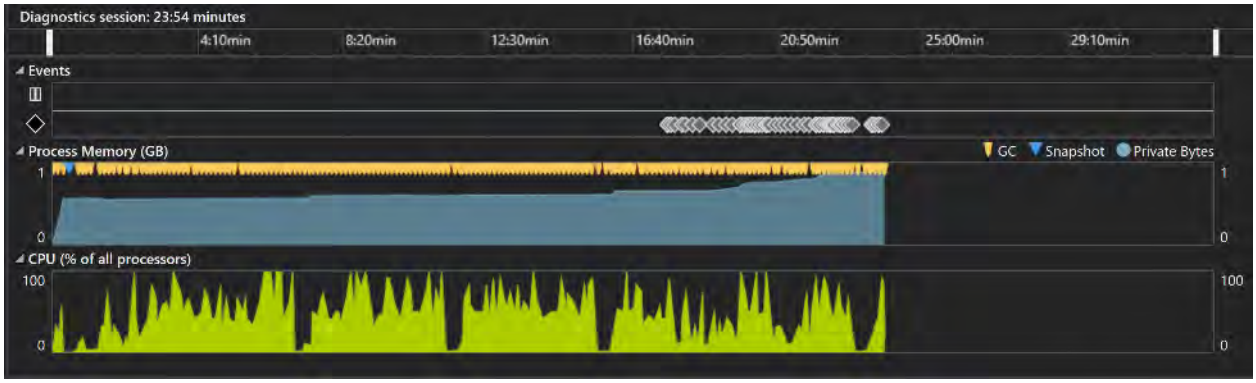


Figure 48. Start of result extraction after completing results generation, showing increase in memory use.

Figure 49. Retrieving event results.

Figure 50. Overall snapshot of performance profiling.

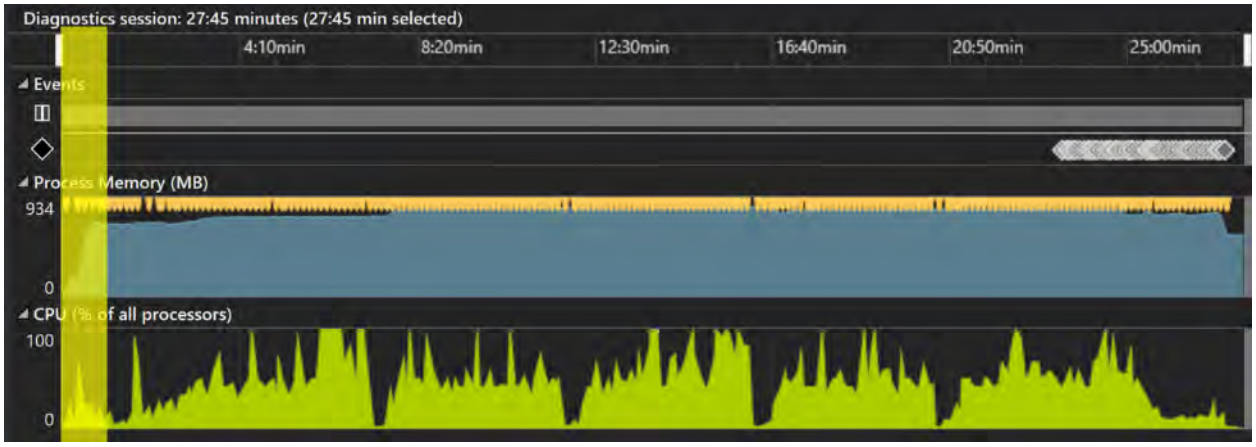


Figure 51. Application start and loading STUDY_DULLES (highlighted in yellow).

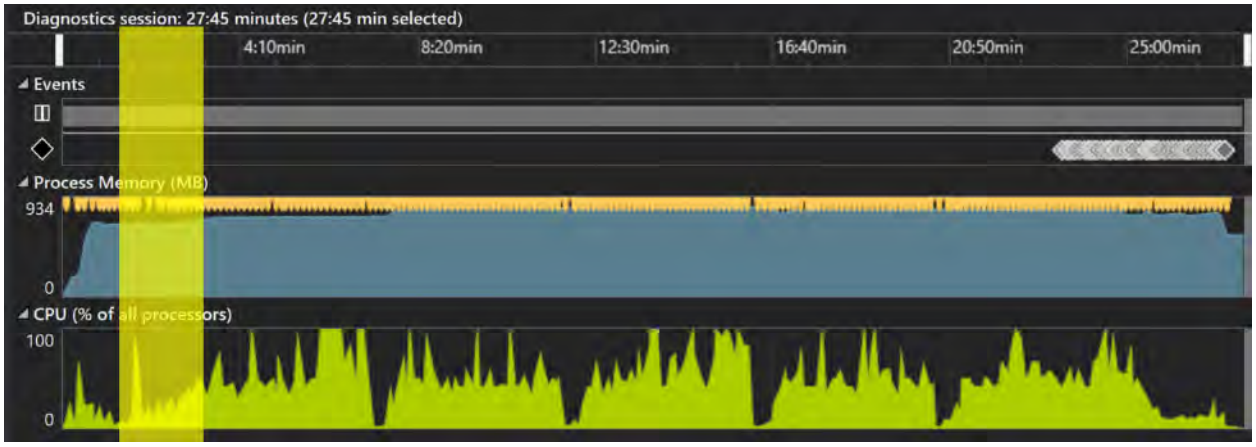


Figure 52. Run metric results and initial warmup (highlighted in yellow).

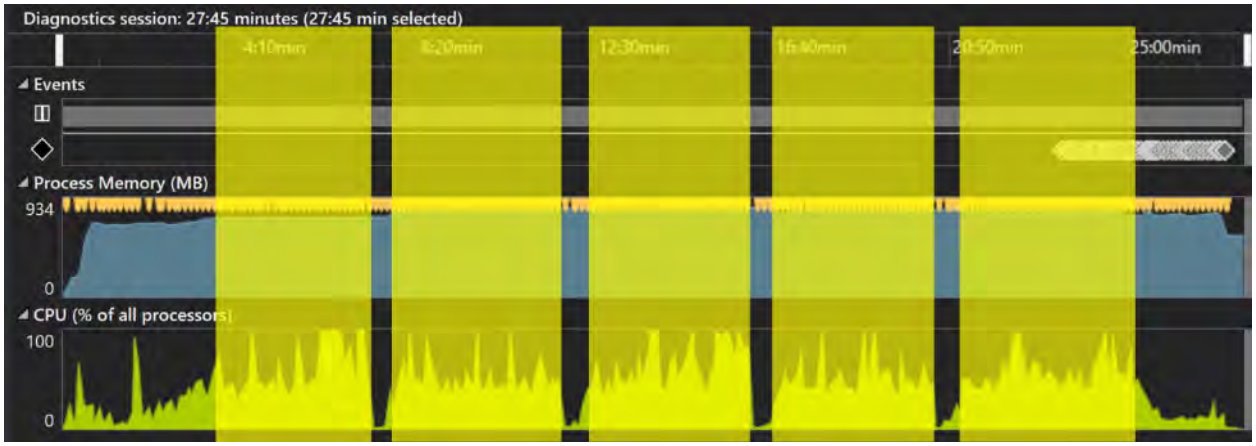


Figure 53. Processing operations (highlighted in yellow), appears to be a reasonable CPU load for hyper-threading.

Figure 54. ~20 second pauses (highlighted in yellow). Unclear why this happens given no obvious memory or SQL delay.

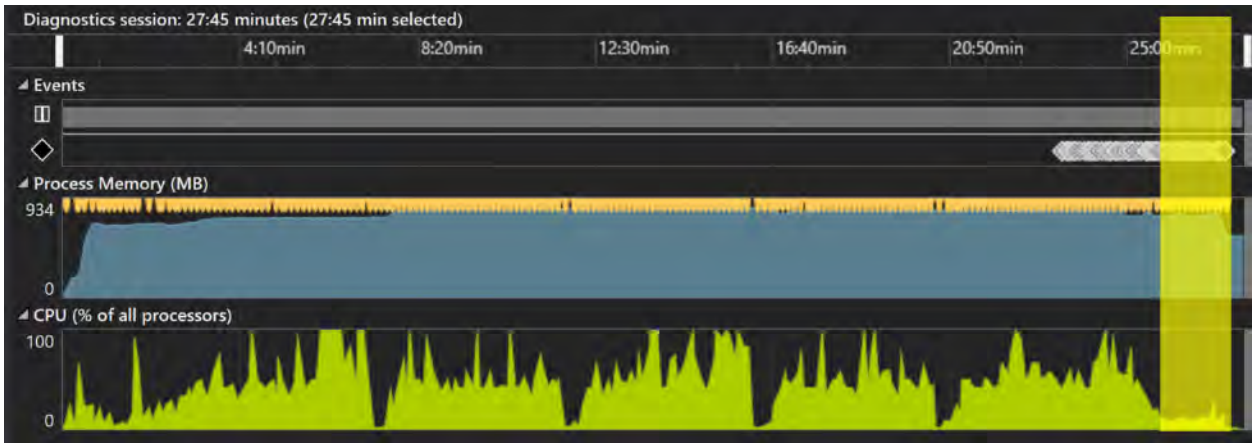


Figure 55. Reloading study and airport data and waiting for user input (highlighted in yellow).

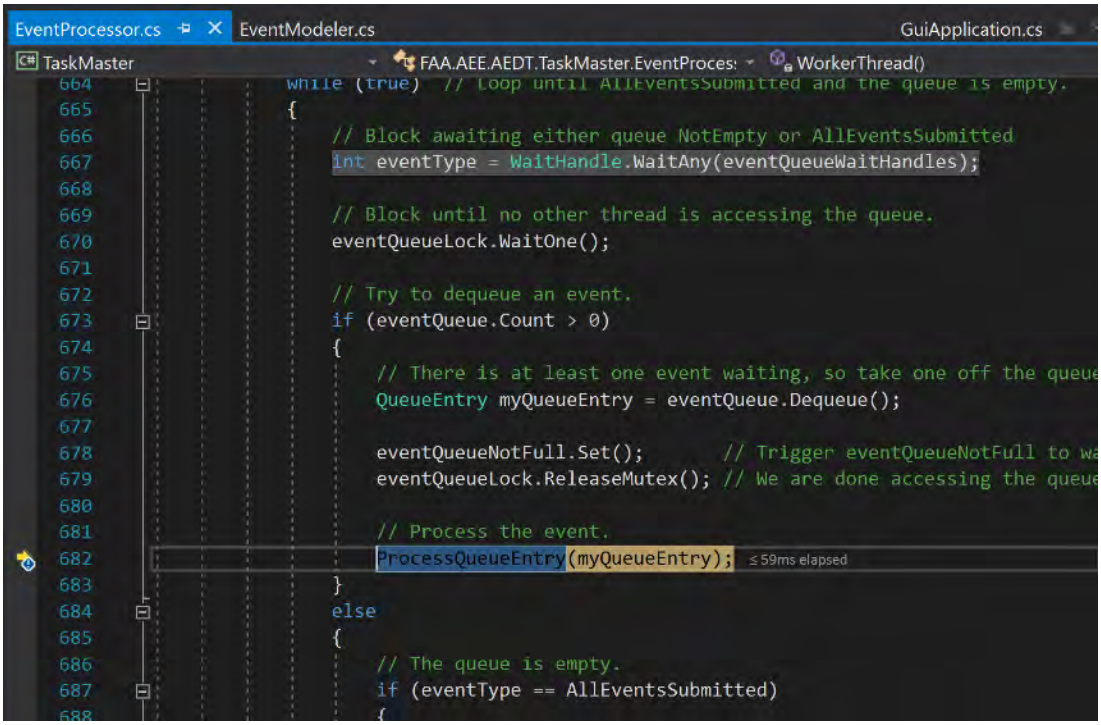


Figure 56. Event processing code for a single thread.

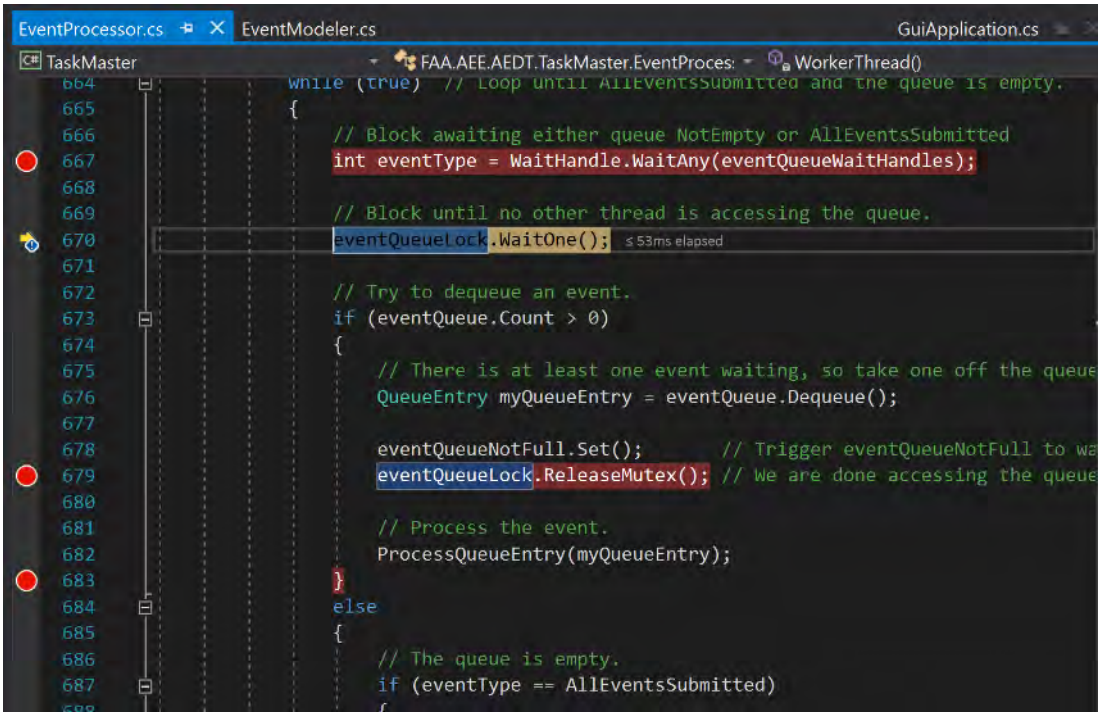


Figure 57. Time spent to block in line 667 was 53ms.

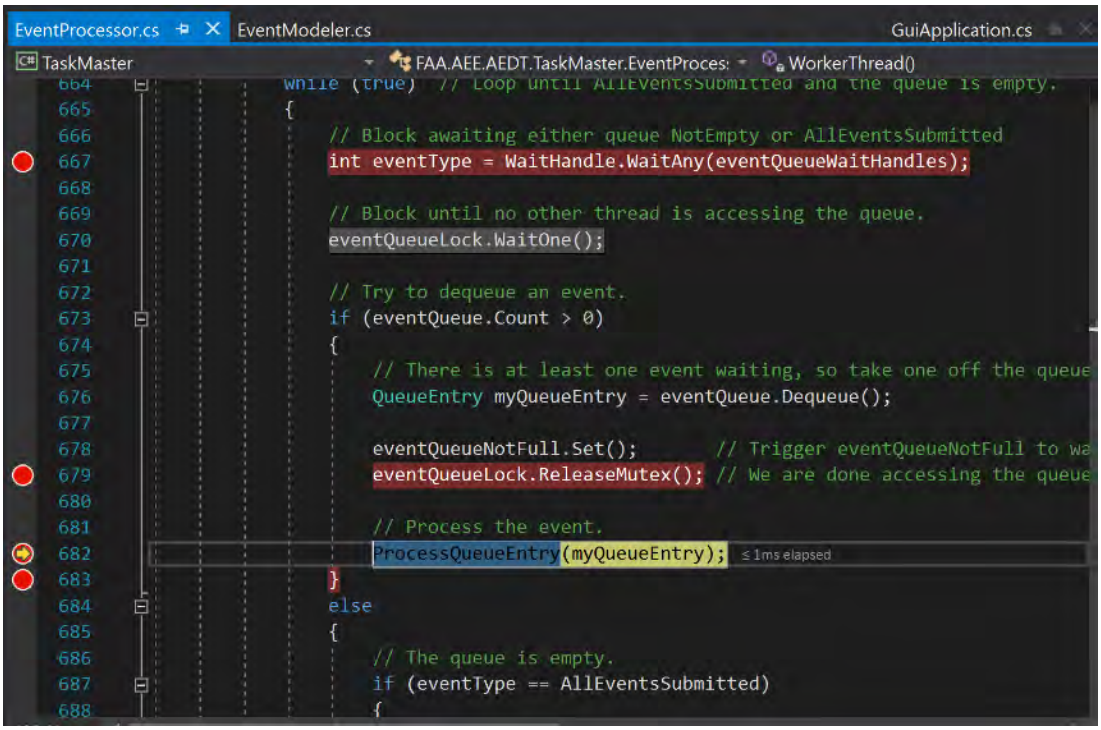


Figure 58. Time spent to get to ProcessQueueEntry from block was <1ms.

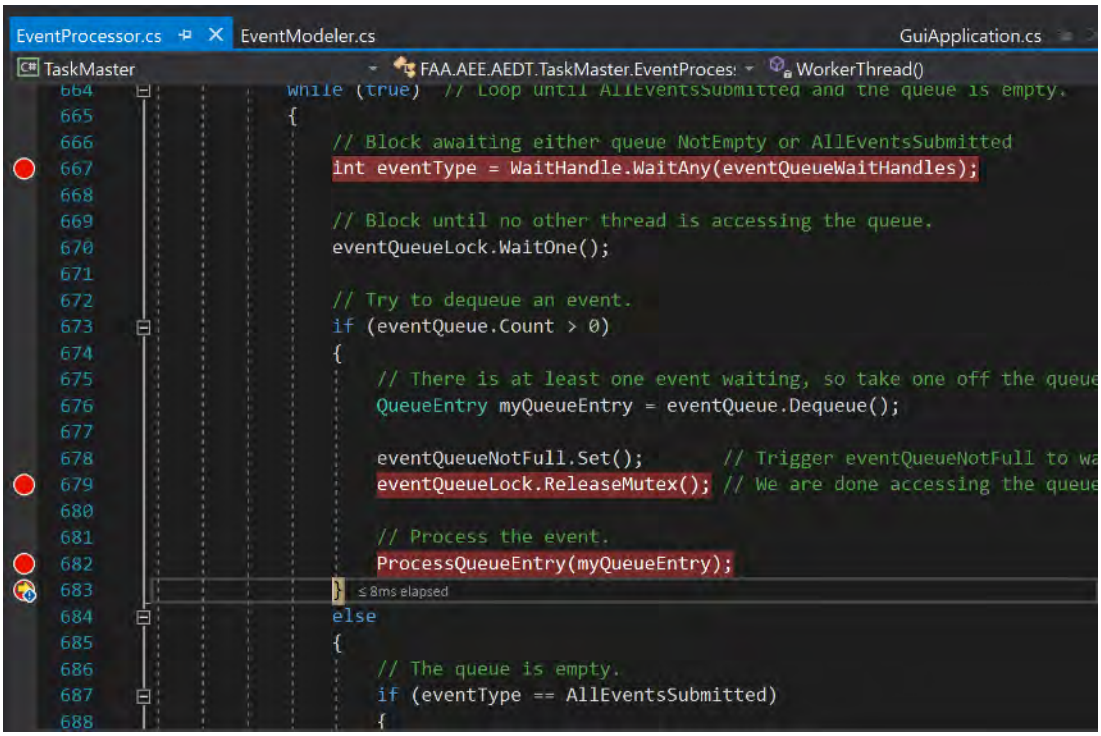


Figure 59. Time spent in ProcessQueueEntry was <8ms.

Recommendations

1. Emissions Dispersion

   Attempts have been made to check if query can be improved using the SQL database tuning advisor. As shown in Figure 62, the estimated performance improvement was about 9%, which is less than a millisecond. This shows that only small improvements could be gained. The bigger issue was the 3-4 milliseconds for connection resets, shown in Figure 63, and the resulting lag from doing so repeatedly.

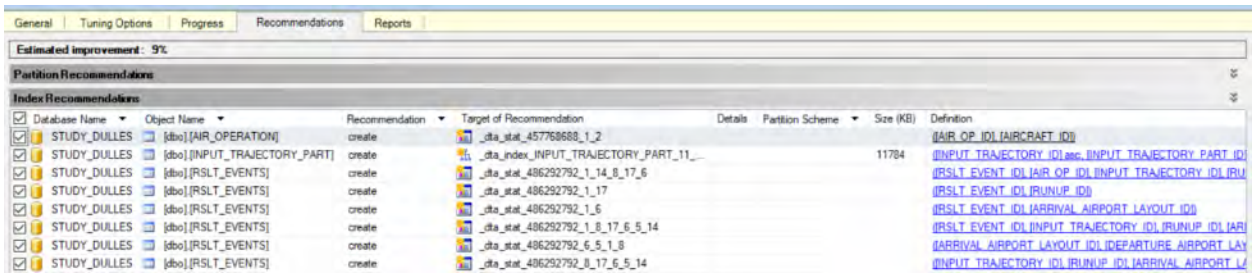
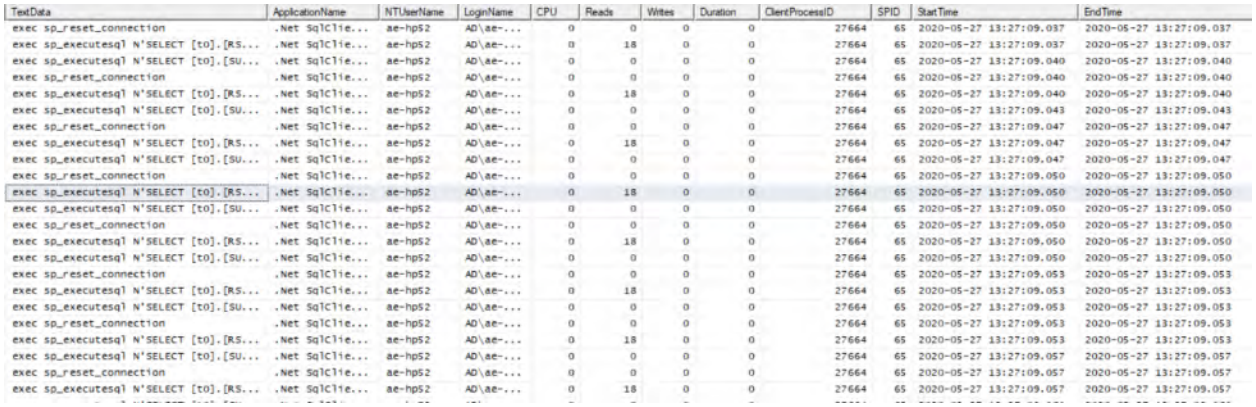


Figure 60. SQL database tuning advisor showing an estimated improvement of 9%.



TextData	ApplicationName	NTUserName	LoginName	CPU	Reads	Writes	Duration	ClientProcessID	SPID	StartTime	EndTime
exec sp_reset_connection	.Net SqlClie...	ae-hpS2	AD\ae-...	0	0	0	0	27664	65	2020-05-27 13:27:09.037	2020-05-27 13:27:09.037
exec sp_executesql N'SELECT [to].[RS...	.Net SqlClie...	ae-hpS2	AD\ae-...	0	18	0	0	27664	65	2020-05-27 13:27:09.037	2020-05-27 13:27:09.037
exec sp_executesql N'SELECT [to].[SU...	.Net SqlClie...	ae-hpS2	AD\ae-...	0	0	0	0	27664	65	2020-05-27 13:27:09.040	2020-05-27 13:27:09.040
exec sp_reset_connection	.Net SqlClie...	ae-hpS2	AD\ae-...	0	0	0	0	27664	65	2020-05-27 13:27:09.040	2020-05-27 13:27:09.040
exec sp_executesql N'SELECT [to].[RS...	.Net SqlClie...	ae-hpS2	AD\ae-...	0	18	0	0	27664	65	2020-05-27 13:27:09.040	2020-05-27 13:27:09.040
exec sp_executesql N'SELECT [to].[SU...	.Net SqlClie...	ae-hpS2	AD\ae-...	0	0	0	0	27664	65	2020-05-27 13:27:09.043	2020-05-27 13:27:09.043
exec sp_reset_connection	.Net SqlClie...	ae-hpS2	AD\ae-...	0	0	0	0	27664	65	2020-05-27 13:27:09.047	2020-05-27 13:27:09.047
exec sp_executesql N'SELECT [to].[RS...	.Net SqlClie...	ae-hpS2	AD\ae-...	0	18	0	0	27664	65	2020-05-27 13:27:09.047	2020-05-27 13:27:09.047
exec sp_executesql N'SELECT [to].[SU...	.Net SqlClie...	ae-hpS2	AD\ae-...	0	0	0	0	27664	65	2020-05-27 13:27:09.047	2020-05-27 13:27:09.047
exec sp_reset_connection	.Net SqlClie...	ae-hpS2	AD\ae-...	0	0	0	0	27664	65	2020-05-27 13:27:09.050	2020-05-27 13:27:09.050
exec sp_executesql N'SELECT [to].[RS...	.Net SqlClie...	ae-hpS2	AD\ae-...	0	18	0	0	27664	65	2020-05-27 13:27:09.050	2020-05-27 13:27:09.050
exec sp_executesql N'SELECT [to].[SU...	.Net SqlClie...	ae-hpS2	AD\ae-...	0	0	0	0	27664	65	2020-05-27 13:27:09.050	2020-05-27 13:27:09.050
exec sp_reset_connection	.Net SqlClie...	ae-hpS2	AD\ae-...	0	0	0	0	27664	65	2020-05-27 13:27:09.050	2020-05-27 13:27:09.050
exec sp_executesql N'SELECT [to].[RS...	.Net SqlClie...	ae-hpS2	AD\ae-...	0	18	0	0	27664	65	2020-05-27 13:27:09.050	2020-05-27 13:27:09.050
exec sp_executesql N'SELECT [to].[SU...	.Net SqlClie...	ae-hpS2	AD\ae-...	0	0	0	0	27664	65	2020-05-27 13:27:09.050	2020-05-27 13:27:09.050
exec sp_reset_connection	.Net SqlClie...	ae-hpS2	AD\ae-...	0	0	0	0	27664	65	2020-05-27 13:27:09.053	2020-05-27 13:27:09.053
exec sp_executesql N'SELECT [to].[RS...	.Net SqlClie...	ae-hpS2	AD\ae-...	0	18	0	0	27664	65	2020-05-27 13:27:09.053	2020-05-27 13:27:09.053
exec sp_executesql N'SELECT [to].[SU...	.Net SqlClie...	ae-hpS2	AD\ae-...	0	0	0	0	27664	65	2020-05-27 13:27:09.053	2020-05-27 13:27:09.053
exec sp_reset_connection	.Net SqlClie...	ae-hpS2	AD\ae-...	0	0	0	0	27664	65	2020-05-27 13:27:09.053	2020-05-27 13:27:09.053
exec sp_executesql N'SELECT [to].[RS...	.Net SqlClie...	ae-hpS2	AD\ae-...	0	18	0	0	27664	65	2020-05-27 13:27:09.053	2020-05-27 13:27:09.053
exec sp_executesql N'SELECT [to].[SU...	.Net SqlClie...	ae-hpS2	AD\ae-...	0	0	0	0	27664	65	2020-05-27 13:27:09.057	2020-05-27 13:27:09.057
exec sp_reset_connection	.Net SqlClie...	ae-hpS2	AD\ae-...	0	0	0	0	27664	65	2020-05-27 13:27:09.057	2020-05-27 13:27:09.057
exec sp_executesql N'SELECT [to].[RS...	.Net SqlClie...	ae-hpS2	AD\ae-...	0	18	0	0	27664	65	2020-05-27 13:27:09.057	2020-05-27 13:27:09.057

Figure 61. Connection resets.

Based on the previous documentation, it is not anticipated that any database tuning would result in much improvement. Additionally, the log shows that each of the queries gets run against the same ID what seems to be six or seven times (it is unclear why this is different). The issue needs to be looked into since it seems unnecessary to pull identical queries and results repeatedly this many times.

2. Emissions
   Overall, memory usage for the application was limited to ~1Gb and a maximum of ~4-5Gb for the SQL server. CPU usage for the application made a relatively good use of the cores and no large usage was observed for the SQL server. As for the Disk/IO, no obvious impact or memory swap appears to be required, and there appears to be no obvious easy improvements.

A deep dive into event processing queues supports the recommendation to either increase queue length in order to reduce synchronization time, or alternatively rethink the way the queuing and synchronization interact. It should be possible to achieve ~5x speed up.

Milestones

- N/A

Major Accomplishments

High Altitude Airport Study

- Identification of test cases to assess validity of NADP profiles at high altitude airports.

Refinement of Thrust and Weight Assumptions

- Regressions (weight versus GCD) and multilinear regressions (weight vs CGD, airport elevation, and runway length) for each airframe for FOQA data.
- Comparison with previous years' weight model results.
- Plotting and interpretation of the thrust distribution for each airframe conditioning of FOQA data to perform regressions for ANP thrust coefficients.

Comparison of NADP Profiles to Real-world Operations

- Implemented data cleaning process to extract and group real-world flight trajectories for comparison.
- Computed overall altitude and ground speed differences between the two recommended NADP profiles and 1-year departures data from SFO for B737-800, A320-211, and A330-301.

Arrival Profile Modeling

- Produced a baseline Python script to identify level-offs based on a vertical speed tolerance and distance tolerance.
- Developed a systemic statistical method to evaluate the effect of different vertical speed and distance tolerances using a design of experiments.
- Identified key characteristics of arrival profiles and level-offs and began the addition of these capabilities to the baseline Python script.

Full Flight Modeling

- Generated a baseline Python script to compare the accuracy of the thread track data with the true model, which is FOQA data.
- Developed a Python script to identify top city pairs from FOQA data and apply the DBSCAN clustering algorithm to find specific trajectory patterns if there is any.

System Testing and Evaluation of AEDT

- Creation of a tool capable of automatically generating TGO/CIR profiles for aircraft with procedural profiles in AEDT and validation of these newly created TGO/CIR profiles.
- Conducted in-depth investigations on several AEDT new features on ANP/BADA4 fuel consumption and thrust modeling.
- Performed evaluation and validation on profile editor.
- Investigated the computational efficiency associated with emissions and emissions dispersion modeling and made recommendations to improve the efficiency.

Publications

- Behere, A., Bhanpato, J., Puranik, T.G., Li, Y., Kirby, M., Mavris D.N., "Data-driven Approach to Environmental Impact Assessment of Real-World Operations", in AIAA SciTech Forum 2021

Outreach Efforts

Bi-weekly calls with the FAA, Volpe, and ATAC. Bi-annual ASCENT meetings. Attended AIAA Aviation conference to present conference paper publication.

Awards

None

Student Involvement

Ameya Behere, Eleni Sotiropoulos-Georgiopoulos, Ayaka Miyamoto, Rukmini Roy, Jirat Bhanpato, Hyungu Choi, Bogdan Dorca, Zhenyu Gao, Santusht Sairam, Graduate Research Assistants, Georgia Institute of Technology

Plans for Next Period

The primary focus for the next period will be:

- Evaluation of test cases for high altitude airport study.
- Comparison of NADP profiles to real-world operations: Expand comparison to other airports and implement comparison by airline to identify differences in operating procedures.
- For the refinement of thrust assumptions, comparisons with weights per stage length in the AEDT tool.
- Linear regression analysis of FOQA data to obtain ANP thrust coefficients.
- For arrival profile modeling, complete the existing development of the Python script with flight characterization capabilities such that this code may be used to identify arrival profile trends which can then be compared to existing AEDT arrival models.
- For full flight modeling, continue investigating the accuracy of the treaded track data with FOQA data and finding the average behavior for the top city pairs.
- Continue system testing and evaluation.