Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
ICAT-2021-04_jansson		

4. Title and Subtitle	5. Report Date
Development of a Fast Method to Analyze Patterns in Airport Noise	September 2021
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
Jansson, Madeleine; Hansman, R. John	

9. Performing Organization Name and Address	10. Work Unit No. (TRAIS)
Center of Excellence for: Alternative Jet Fuels and Environment Massachusetts Institute of Technology Cambridge, MA 02139 USA	11. Contract or Grant No. 13-C-AJFE-MIT-060 & 075

12. Sponsoring Agency Name and Address	13. Type of Report and Period Covered
Federal Aviation Administration Office of Environment and Energy	Technical Report
	14. Sponsoring Agency Code

15. Supplementary Notes

https://hdl.handle.net/1721.1/133212

16. Abstract

Traditional airport noise modeling is limited in its ability to analyze large quantities of flight tracks due to high computation time. As a result, yearly noise reports are often limited to modeling flights from a single “representative day,” which lacks detail arising from the natural dispersion of flight tracks and variety in airport operations occurring throughout an entire year of operations.

A framework for processing actual flight data and applying an existing, fast noise approximation is presented. Tens of thousands of flights can be analyzed in a matter of hours, allowing for a data-comprehensive approach to calculating noise metrics. Method results are cross-validated against the Aviation Environmental Design Tool (AEDT) on a single-event basis and an existing aggregate result on a multi-event basis.

Results for a variety of metrics are presented based on data sourced from Boston Logan International Airport in 2016. Day-Night-Level (DNL) is calculated on a yearly, daily, and hourly basis, highlighting the variability in noise patterns depending on evolving airport runway configuration. N60 is calculated as a supplemental metric on a daily basis.

17. Key Words	18. Distribution Statement
ASCENT	

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified		

MIT
International Center for
Air Transportation

DEVELOPMENT OF A FAST METHOD TO ANALYZE PATTERNS IN AIRPORT NOISE

Madeleine Jansson and R. John Hansman

This report is based on the Master's Thesis of Madeleine Jansson submitted to the Department of Aeronautics and Astronautics in partial fulfillment of the requirements for the degree of Master of Science at the Massachusetts Institute of Technology.

Report No. ICAT-2021-04
September 2021

MIT International Center for Air Transportation (ICAT)
Department of Aeronautics & Astronautics
Massachusetts Institute of Technology
Cambridge, MA 02139 USA

[Page Intentionally Left Blank]

Development of a Fast Method to Analyze Patterns in Airport Noise

by

Madeleine Jansson

Submitted to the Department of Aerospace Engineering
on August 19, 2021 in Partial Fulfillment of the
Requirements for the Degree of Master of Science in
Aerospace Engineering

ABSTRACT

Traditional airport noise modeling is limited in its ability to analyze large quantities of flight tracks due to high computation time. As a result, yearly noise reports are often limited to modeling flights from a single "representative day," which lacks detail arising from the natural dispersion of flight tracks and variety in airport operations occurring throughout an entire year of operations.

A framework for processing actual flight data and applying an existing, fast noise approximation is presented. Tens of thousands of flights can be analyzed in a matter of hours, allowing for a data-comprehensive approach to calculating noise metrics. Method results are cross-validated against the Aviation Environmental Design Tool (AEDT) on a single-event basis and an existing aggregate result on a multi-event basis.

Results for a variety of metrics are presented based on data sourced from Boston Logan International Airport in 2016. Day-Night-Level (DNL) is calculated on a yearly, daily, and hourly basis, highlighting the variability in noise patterns depending on evolving airport runway configuration. N_{60} is calculated as a supplemental metric on a daily basis.

Thesis Supervisor: R. John Hansman
Title: T. Wilson Professor of Aeronautics and Astronautics

Contents

Table of Figures

Acknowledgements

This research was funded by the U.S. Federal Aviation Administration Office of Environment and Energy through ASCENT, the FAA Center of Excellence for Alternative Jet Fuels and the Environment, projects 72 and 23 through FAA Award Number 13-C-AJFE-MIT under the supervision of Sean Doyle (72), Donald Scata (72), Chris Dorbian and Joseph DiPardo (23). Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the FAA.

1 – Overview of Aviation Noise

1.1 Motivation

Aviation noise modeling is an important environmental impact tool allowing airport officials and researchers to identify the geographic distribution of noise exposure, understand the operational drivers of an airport's noise footprint, and communicate with the surrounding public. When performed on an individual flight basis, noise modeling requires inputting a flight trajectory and aircraft technical details, as depicted in Figure 1. The noise results from multiple flights within a given time period can then be combined into aggregate exposure impacts.

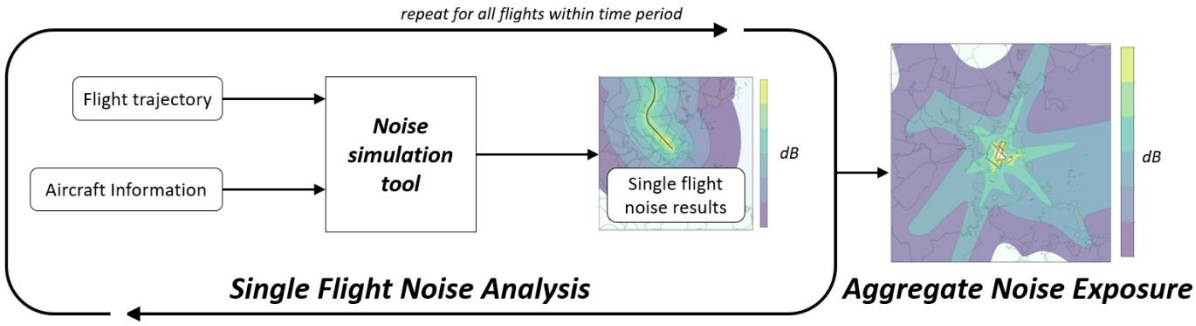


Figure 1. *Block diagram depicting the standard noise analysis process for single and multi-events.*

Because existing flight simulation tools are computationally (and therefore time) expensive, multi-event analyses for large periods of time often rely on significant assumptions about typical flight paths, schedule, and fleet mix. As a result, these aggregate models are limited in their ability to fully represent operational variation and natural dispersion in flight paths.

This thesis addresses these limitations in noise modeling. In presenting a joint data-processing and noise-approximation approach, actual flight data is analyzed at Boston Logan International Airport on a variety of timescales with no assumptions made about average airport operations.

1.2 Aviation Noise Metrics

Figure 2 illustrates the noise experienced by an observer of a single flight. The peak sound pressure level is referred to as L_{max} (or $L_{A,max}$ when pressures are weighted by their frequencies to favor those which drive human annoyance). The average sound pressure level for a given flight is referred to as L_{eq} ("equivalent level"). Sound pressure is also often integrated over time within 10 decibels of its maximum value—this value is known as sound exposure level (SEL).

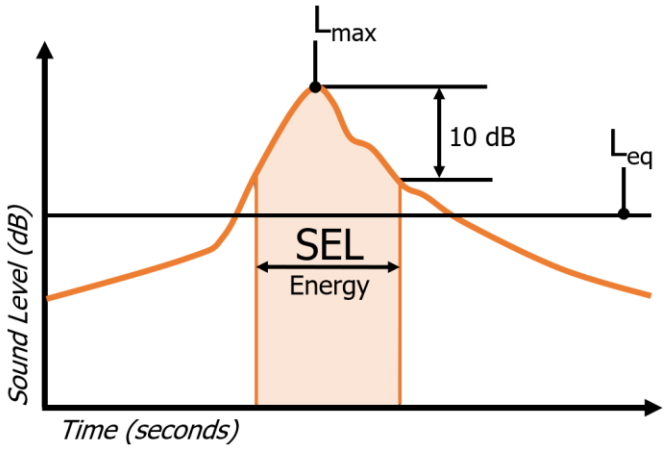


Figure 2. *Graphical depiction of three common noise metrics (L_{max}, L_{eq}, and SEL) for an observation of a single flight. Adapted from [1].*

Aggregate event metrics account for noise from *multiple* flights over a given time period. For example, the N_X metric determines the number of overflights at a particular location that exceed a specified $L_{A,max}$ level (*X* dB) on a given day. One N_X metric variety is $N_{60\ day,\ 50\ night}$ (abbreviated N_{60}), which [2] correlated with noise complaints on a peak air traffic day.

The most common aggregate metric is annual average Day-Night Level (DNL) which represents a typical day of flight noise at a given airport, penalizing nighttime flights between 10pm and 7am with an additional 10 dB. DNL estimates are built up from the SEL results of many modeled flights according to Equation 1. The summation of these SEL values (essentially the total integrated noise exposure) is divided by the total time duration modeled.

$$DNL = 10\log\frac{1}{\Delta t}\left(\sum_{day\ flights}^{\Delta t} 10^{\frac{SEL}{10}} + \sum_{night\ flights}^{\Delta t} 10^{\frac{SEL+10}{10}}\right)$$

Equation 1. *The formula for Day-Night Level is based off of the SEL metric for individual flights. DNL serves as an "average" for aviation noise occurring over a given period of time, Δt.*

True calculation of annual average DNL would require simulating noise for *every* flight within a given year and dividing by the total number of seconds in a year. Comprehensively calculating DNL using every flight is, however, a costly time investment because major airports such as Boston Logan support hundreds of thousands of aircraft operations annually. Based on discussion with an industry subject matter expert, DNL is commonly approximated by making assumptions about a "representative day" of flight operations such that only noise from the representative day is modeled and divided by the number of seconds in a single day.

1.3 "Representative Day" Noise Modeling Assumptions

"Representative day" analysis first requires determining the average number of daily flight operations given the yearly total—this number must then be further decomposed into daytime and nighttime operations. Operations must also be divided amongst runways and this distribution either split evenly throughout 24 hours or unevenly between daytime and nighttime. Flight tracks for each runway are selected based on recorded data: "backbone" tracks represent the centerlines of common procedures while "subtracks" diverge from the backbone to mimic real-life dispersion. Finally, flights are split between a number of aircraft that are characteristic of the yearly fleet mix, each with their own vertical flight profile (altitude and power along the flight path).

In short, DNL may be estimated by summing the partial contributions of combinations of runway, flight procedure, time of day, and aircraft type. In the simplified example shown below

in Equation 2, the representative day includes a (potentially fractional) number $n_{A320s,Dep,Rwy9}$ of A320 departures off of runway 9 and $n_{B737s,Arr,Rwy22}$ of B737 arrivals onto runway 22.

$$DNL_{rep\ day} = 10\log\frac{1}{86{,}400\ \mathrm{s}}\left(n_{A320s,Dep,Rwy9}\cdot 10^{\frac{SEL_{A320s,Dep,Rwy\ 9}}{10}} + n_{B737s,Arr,Rwy22}\cdot 10^{\frac{SEL_{B737s,Arr,Rwy\ 22}}{10}} + \cdots\right)$$

Equation 2. *A modified formula for DNL illustrating how the metric can be built for a representative day using the partial contributions of different flight operations.*

There are different ways of estimating and combining operational distributions for the same year at the same airport, leading to variability in DNL results. Furthermore, "annual average" metrics in-and-of-themselves are unable to reflect natural variation in noise due to seasonal, daily, and even hourly changes in traffic and weather patterns.

1.4 Fast Flight Modeling Approach

This thesis presents an approach for faster noise analysis, enabling the use of actual flight data rather than assuming a representative day. This approach can quickly generate noise results on a single event ($L_{A,max}$ and SEL) or multi event (DNL and N_X) basis. Using such a tool, noise can be calculated on hourly, daily, monthly, or yearly timeframes, revealing insights into the variability and patterns of aviation noise. A block diagram of the framework is shown below in Figure 3.

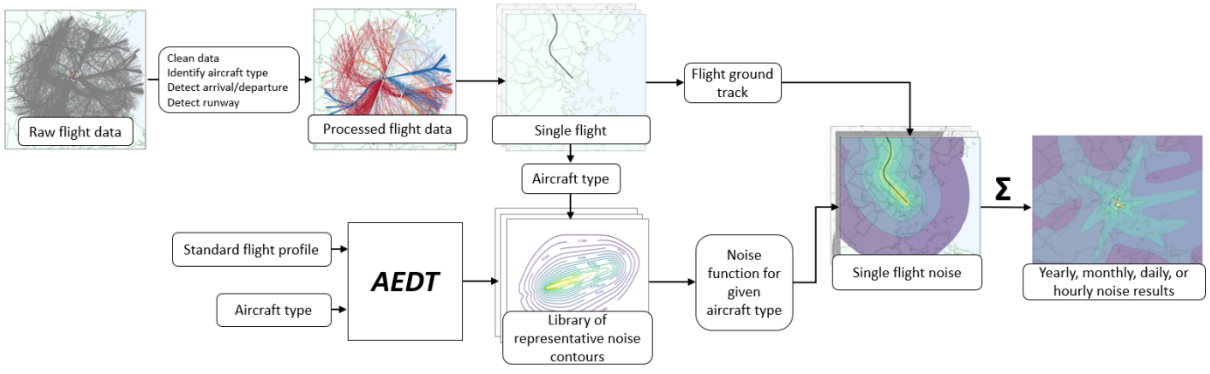


Figure 3. *Block diagram of fast noise modelling framework.*

The first step of this process is to clean and pre-process a source of raw flight data, described in *2 – Flight Data Processing*. The next step is to apply a computationally efficient noise approximation method on each flight. This method involves pre-calculating noise results using the Aviation Environmental Design Tool (AEDT) noise simulator. Noise results are calculated for different families (or "bins") of aircraft type using default vertical flight profiles (altitude and power settings) and assembled into a library of contours. The contours can then be "molded" around each flight track in the dataset. This molding process is detailed in *3 – Noise Approximation Method*. Finally, flight-level noise results are aggregated according to the dataset schedule to compute multi-event metrics. *4 – Noise Results* presents yearly, daily, hourly, and sub-hourly noise results for Boston Logan International Airport (KBOS). In addition to providing accurate estimates of noise based on actual flight ground tracks, the noise results highlight the effect of runway configuration on the variation of noise surrounding an airport.

2 – Flight Data Processing

2.1 Flight Data Sources

The flight data used for this analysis comes from the Airport Surface Detection Equipment, Model X (ASDE-X) system. The noise modeling framework can be applied more generally to any source of aviation data, such as Automatic Dependent Surveillance-Broadcast (ADS-B) or the Enhanced Traffic Management System (ETMS). Each data source varies not just in its data format, but also in temporal frequency, geographic extent of observations, and the presence of coverage outages. For the methodology used within this thesis, any flight data source is acceptable so long as it contains geographic coordinates, time, and aircraft type for each flight.

2.2 Data Cleanup

Flight data must first be pre-processed such that individual operations are grouped into unique data structures. The steps required to accomplish this depend on the data source. Raw ASDE-X data, for example, comes in the form of disjointed observations and requires extensive preprocessing, as described in *Appendix A – ASDE-X System*. After pre-processing, data needs to be cleaned to remove sensor noise and format it to interface with the library of noise contours. The cleanup steps are illustrated in the Figure 4 flow diagram and then discussed in the subsections to follow.

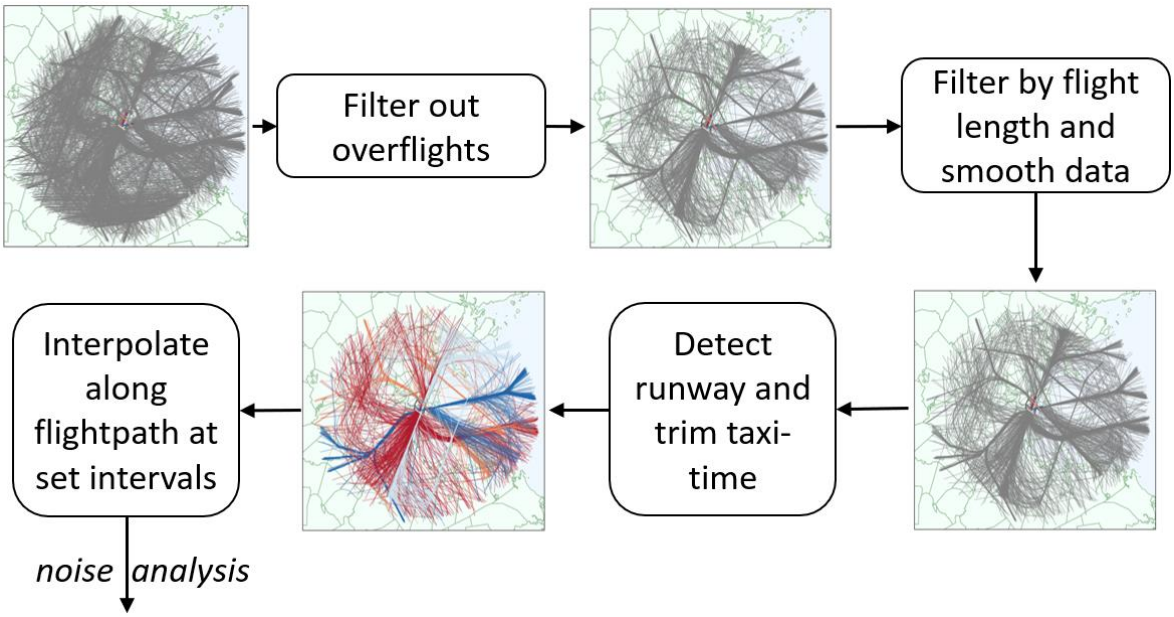


Figure 4. *Diagram summarizing the data cleanup processes.*

2.2.1 Airport Information

As part of the data cleanup and subsequent noise analysis processes, geographic information about an airport is used to identify arrival/departure status and classify each operation's runway. This information includes: an airport origin (latitude, longitude); coordinates approximating the airport planform (a polygon roughly surrounding the runways, taxiways, and aprons); a list of runways, their corresponding endpoint coordinates, and their widths; the airport altitude; and the UTC time zone. The key geographic parameters of an airport are visualized as in Figure 5 below.

Within the code-base supporting this thesis, object-oriented programming was used to create data structures for airports, runways, and flights. For example, the *Runway* class stores runway endpoints and contains methods to calculate length and generate geometric polygons used for runway detection across flights (to be discussed further in a subsequent section).

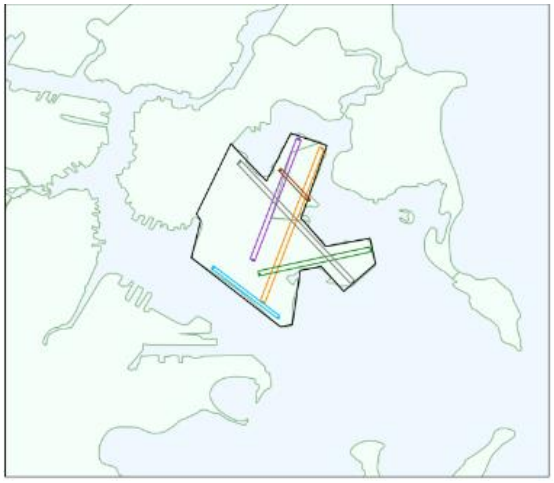


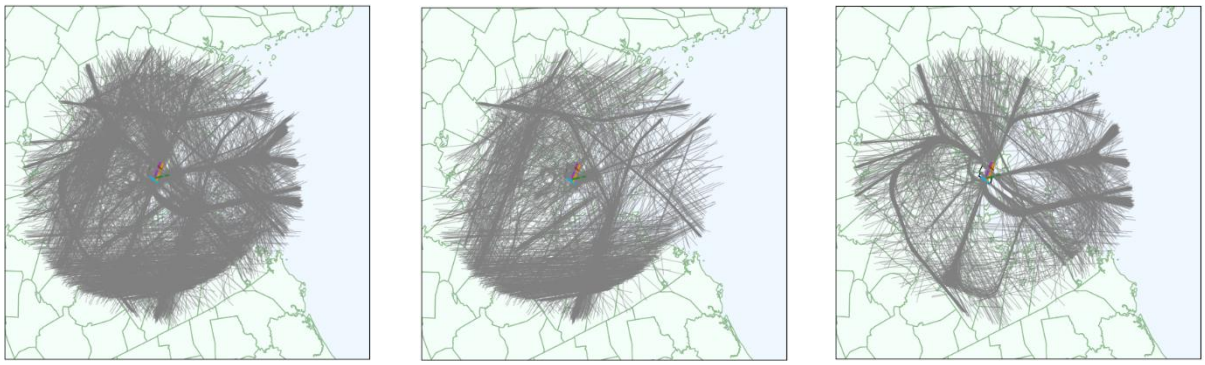
Figure 5. *Geographic parameters of Boston Logan airport. The airport polygon is shown in black and runway polygons and endpoints are color-coded.*

2.2.2 Flight Status Detection and Overflight Filter

First, flights are classified as arrivals, departures, or overflights based on the first and last position (latitude, longitude) recorded by the ASDE-X system. If the flight starts farther from the airport origin than it ends, it is classified as an arrival (and departures are classified vice versa).

Next, overflights (airplanes that fly above the airport but do not land) and non-flights (airplanes and ground vehicles that do not take off from the airport) must be filtered out.[1] Overflights and non-flights are identified based on minimum altitude, maximum altitude, and whether or not a given trajectory is within the airport polygon for at least than 10 observations (arrivals and departures meet this final criterion easily due to slow taxi-time). Overflights and non-flights that are not caught by these filters are likely caught during the subsequent runway detection phase or interpolation phases. The overflight/non-flight filter results are depicted graphically in Figure 6.

[1] Overflights are filtered out because they typically occur at high altitudes and are irrelevant to noise exposure.



(a) ***(b)*** ***(c)***

Figure 6. (a) *Flights prior to applying overflight/non-flight filter;* ***(b)*** *flights* removed *by the filter; and* ***(c)*** *arrivals and departures remaining after applying the filter.*

2.2.3 Cumulative Length Filter

In order to remove fragments of incomplete flights, flights are filtered by their cumulative length traveled. Latitude and longitude are converted into *x,y* coordinates in nautical miles from the airport origin. Cumulative "along-track" flight length was calculated based on these *x,y* coordinates and all flights below a threshold were removed from the dataset. Figure 7 illustrates an example of flights removed by the filter. Removed flights typically fall within a few categories: snippets of real flights that have been shortened due to a loss of ASDE-X coverage; snippets of overflights that are only within radar range for a brief period overhead the airport (due to their high altitude), or snippets of movement around the airport taxiways and apron.

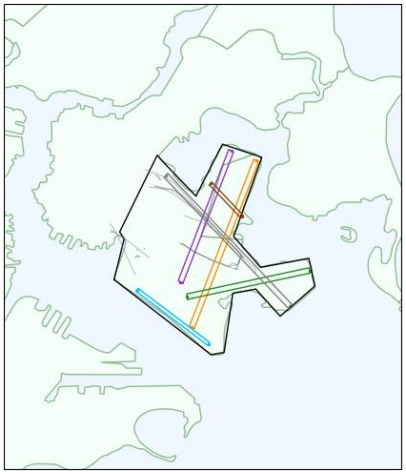
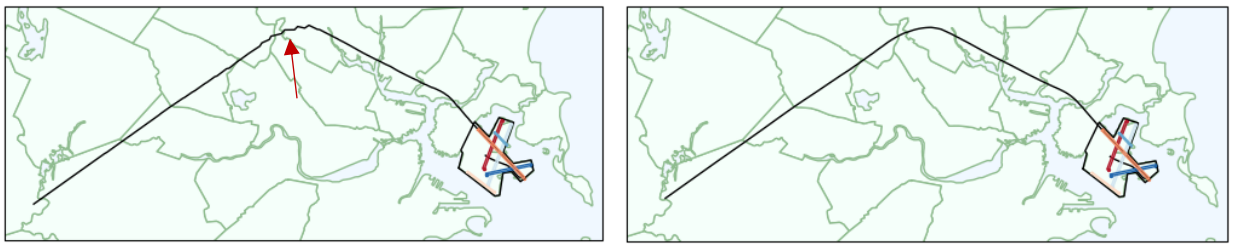


Figure 7. *Flights removed by the cumulative length filter when applied to a sample dataset.*

2.2.4 Flight Smoothing Filter

A centered moving average with a window of 9 observation points is then applied to the geographic coordinates of each flight to filter out sensor noise. This noise is typically most evident during turns, an example of which is shown in Figure 8.



(a) ***(b)***

Figure 8. *An example flight before* ***(a)*** *and after* ***(b)*** *applying a moving average to filter out sensor noise.*

2.2.5 Runway Detection and Taxi-Time Trimming

In order to align flight tracks with the pre-calculated noise contours, flights must start or end at a defined point on the runway. This means that the departure or arrival runway must be identified and all observations on the taxiways and apron removed.

Runway detection is performed according to a modified version of the algorithm presented in [3]. A runway is considered a viable match if **a)** a flight travels down it for a specified percentage of the total runway length and **b)** there are a minimum number of observation points on said runway. The time-wise first viable match is selected. An example of runway detection results is shown in Figure 9.

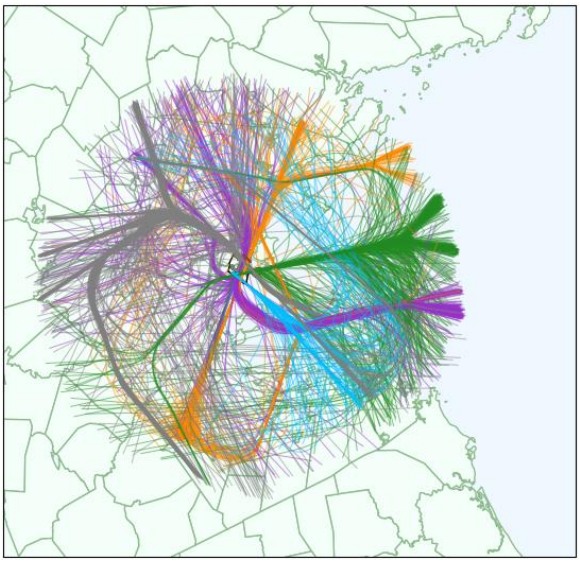
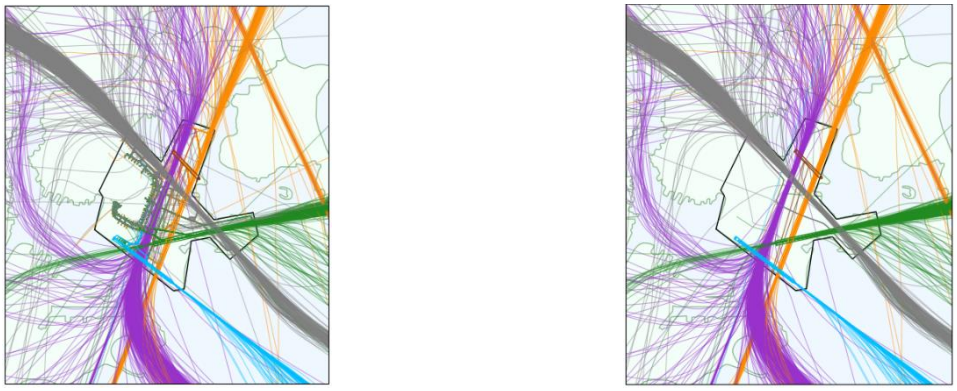


***Figure 9**. A sample of flights from KBOS are color coded by their detected runways. Flights in purple are for runway 04L/22R; orange for runway 04R/22L; green for runway 09/27; gray for runway 15R/33L; and light blue for 14/32.*

Following runway detection, all flights are stripped of their taxi and runway observations, which are replaced by a single common runway endpoint. In this manner, all flights on a runway are specified to travel the entire runway. A zoomed in example before and after this trimming and replacement process is shown in Figure 10.



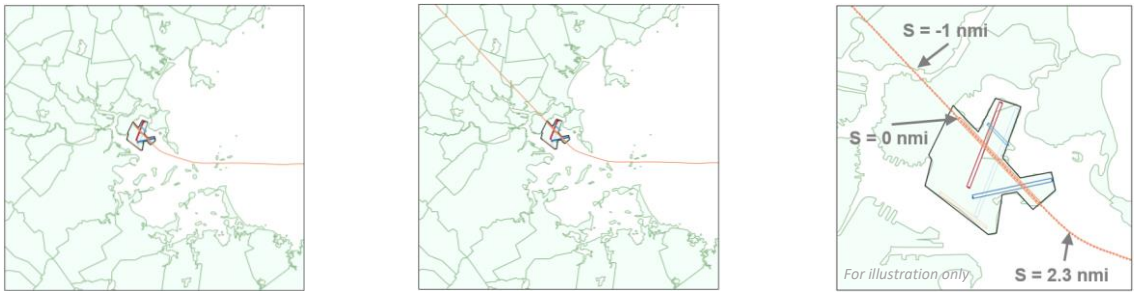
(a) ***(b)***

***Figure 10**. A zoomed in view of flights **(a)** before and **(b)** after taxi-way trimming. There is a noticeable difference in activity on the taxiways and apron between the figures. Cleaning large datasets is an art, not a science, and not all taxiway noise can be removed.*

2.2.6 Flight Interpolation

The final step in data pre-processing is to interpolate flight ground tracks (latitude, longitude) to set intervals of cumulative track length. These intervals are used to line up flight ground tracks

with the noise levels at the corresponding stage of vertical flight profile (altitude and power). Flights are interpolated in both "directions." For example, an arrival ground track is interpolated backwards *prior to arrival* and also *after arrival* in a straight line extending outwards from the runway. An example of an interpolated departure is shown in Figure 11: notice how the final flight track extends in the direction of Runway 33L, even though the departure takes off of Runway 15R.



(a) ***(b)*** ***(c)***

Figure 11. *A departure off of runway 15R at KBOS* ***(a)*** *before and* ***(b)*** *after interpolation along the flight path;* ***(c)*** *depicts the same flight zoomed in around the airport with three illustrative cumulative track distance points (*s *values).*

3 – Noise Approximation Method

3.1 Aviation Environmental Design Tool (AEDT)

The existing industry-standard tool for aggregate noise analysis is the FAA's Aviation Environmental Design Tool (AEDT), for which a high-level summary is shown in Figure 12. AEDT bases its noise calculations off of experimentally-derived noise-power-distance (NPD) tables, applying corrections for noise attenuation and directivity, weather, flight speed and path geometry, and optionally for terrain and bank angle.

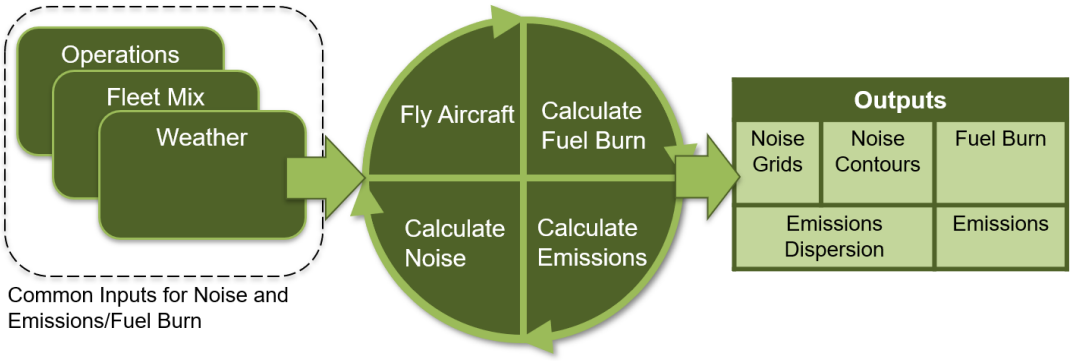


Figure 12. *High-level block diagram showing the capabilities of AEDT. Adapted from [4].*

To run AEDT on an individual flight, a user must input the aircraft type, geographic coordinates (latitude and longitude of the flight), and select an arrival or departure flight profile (altitude and power along the flight path). An example AEDT flight profile is shown in Figure 13.

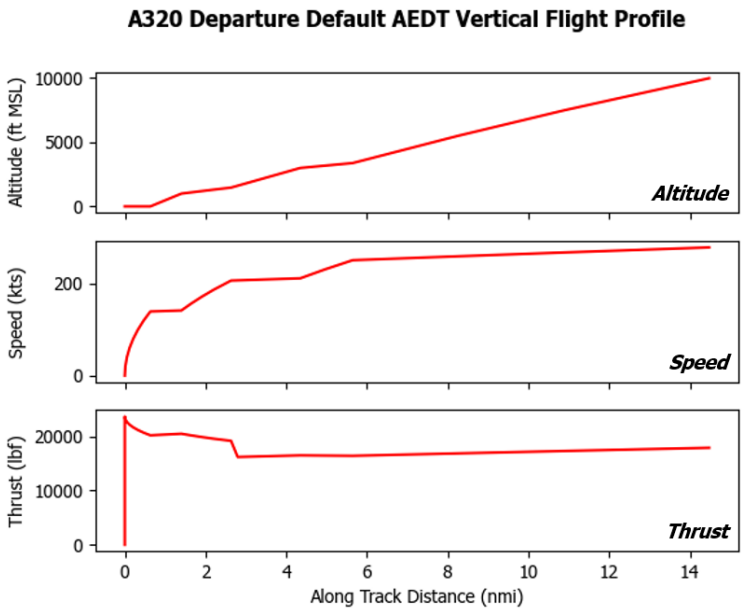


Figure 13. *AEDT default vertical flight profile for an A320 departure with an intermediate stage length.*

The noise approximation method used in this thesis is not specific to AEDT. Any noise simulator is acceptable so long as it can output geometric noise contours along straight in-and-out departure and arrival tracks for a variety of aircraft types.

As previously discussed in *1 – Overview of Aviation Noise*, discussion with an industry professional revealed that existing simulator runtimes do not easily support analyses with large numbers of flights (such as all the flights in a year). For this reason, airport yearly DNL is often calculated by simulating a representative day composed of averaged arrival and departure counts, fleet mix, runway use, flight tracks, flight profiles, and schedule. As a result, these DNL estimates do not fully capture the natural dispersion present in real flight tracks.

3.2 Fast Noise Approximation Method

Because the goal of this effort was to quickly analyze large quantities of actual flights, a faster alternative to traditional AEDT analysis was imperative. The implemented solution was the “half-width” approach presented by [5], so-named for the underlying noise functions derived from *half*

the width of symmetrical noise contours. Within this approach, AEDT is used to pre-calculate noise contours for combinations of aircraft type, vertical flight profile (altitude and power), and arrival/departure status. Flight paths are specified as "straight in-and-out" arrivals and departures extending from a runway reference point (the noise results are therefore symmetric about the ground track). For each combination of aircraft type and arrival/departure status, the contour width for a given noise level is known as a function of the along-flight-track-distance (*s*), as shown in Figure 14. Contour levels can then be geometrically molded to ground tracks, as in Figure 15.

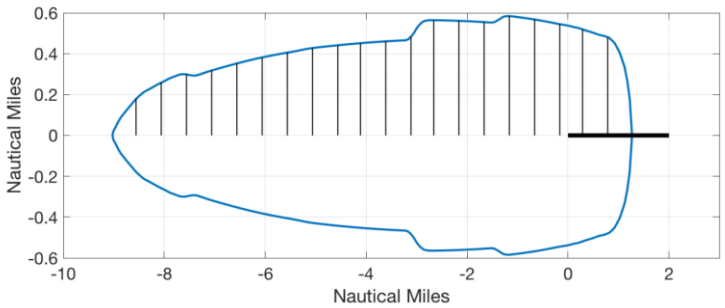


Figure 14. *A Boeing 737-800 arrival contour at the 60dB L_{MAX} level, from [5].*

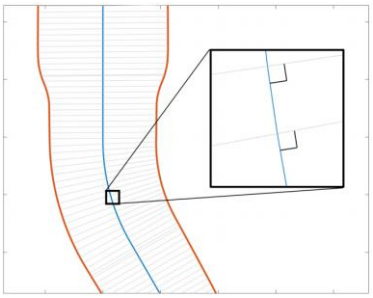


Figure 15 *. Flight track (in blue) with a noise contour (in red) molded around it, from [5].*

Because contours applied by the half-width method are derived from simulating straight in-and-out flights, the method is not able to capture influences on noise like path geometry and bank angle, such as what might be seen when turning from the downwind to upwind legs on an arrival. Additionally, SEL metric contours see inaccuracies due to their dependence on exposure time (unlike $L_{A,max}$ contours), as noted by [5]. When applied to thousands of flights in aggregate, however, the method's shortcomings are outweighed by its broad-strokes accuracy and speed.

3.3 Overview of "Half-Width" Method Application

An overview of the "half-width" method application for this thesis is presented in Figure 16. The following subsections will describe the method's steps in greater detail: first, *Generation of Half-Width Method "Function Library"* will describe how aircraft are "binned" according to their vehicle model and how contours for each bin are generated using AEDT to ultimately build a library of noise results. Then, *Implementation of Noise Approximation Method* will detail the computational implementation of the noise approximation method. Finally, *Validation of Method Results* will show how the method compares to AEDT-simulated results for a few representative flights.

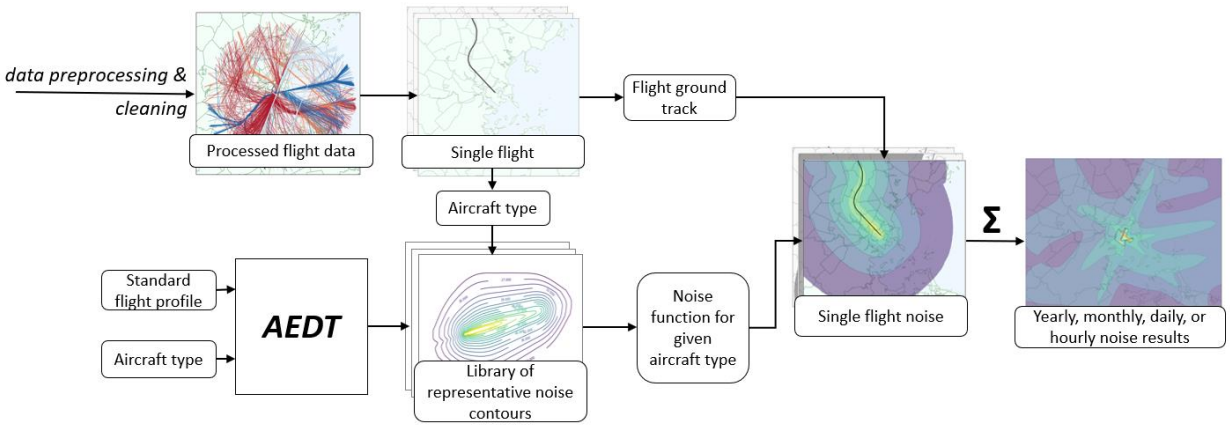


Figure 16. *Block diagram for the overall noise approximation process.*

3.3.1 Generation of Half-Width Method "Function Library"

The noise analysis tool presented in this thesis uses half-width functions for nine representative bins of aircraft type, presented in Table 1, with aircraft sorted according to research by Brenner [2]. Every flight analyzed is grouped into one of these bins based on the vehicle model recorded by the ASDE-X system. Half-width functions exist for all bins in both arrival and departure varieties.

Category	Acronym	Representative Aircraft Type
Twin Aisle Jet	TA	B773
Airbus A320 Family	A320	A320
Boeing 757 Family	B757	B752
Boeing 737 Family	B737	B738
Older Jet	OJ	MD88
Large Regional Jet	LRJ	E170
Small Regional Jet, Business Jet, and Turboprop	SRJ	E145
Piston Engine	PNJ	C402
Unknown	UNK	*Excluded from analysis*

***Table 1**. Adapted from [2]. The addition of piston engine aircraft is made for this thesis.*

The half-width functions were derived from AEDT simulations using atmospheric conditions for Boston Logan airport and default vertical profiles for intermediate stage-length flights. *Appendix B – Flight Profiles Used for Approximation Contours* shows the arrival and departure altitude profiles used by AEDT for each aircraft bin compared against a sample of ASDE-X data recorded at KBOS. Note that the altitude profiles typically match well with ASDE-X data although there is future opportunity to use median ASDE-X profiles for more specific combinations of aircraft type, arrival/departure status, runway, procedure, etc.

SEL simulations were run on an 80 nmi square grid centered at the airport and discretized every 0.025 nmi. Half-width functions were typically generated at 1 dB increments, with the lowest noise level limited by the grid size and the highest noise level limited by the grid discretization.[2] The shape of the contours was approximated such that functions for a given noise level were always a 1-to-1, as shown in Figure 17, which is helpful for some algorithmic implementations of the method.

[2] $L_{A,max}$ simulations were run on a 60 nmi square grid discretized every 0.1 nmi. Functions were generated at typically 2 dB_A increments. The lowest noise level was between 30 and 35 dB_A for all contour bins and the upper level limited by grid discretization.

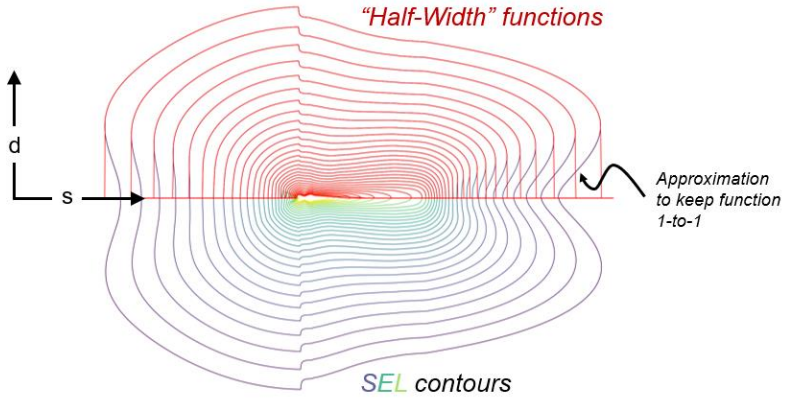


Figure 17. *Half-width functions for a flight shown in red over the raw SEL contours. "s" denotes the distance along the flight path and "d" denotes the perpendicular distance from the flight path.*

The contour data was then rearranged into noise functions for a given along-track distance "s" with respect to perpendicular distance from the track "d." This can be visualized as taking a cross section through all the contour levels at a given along-track distance, as seen in Figure 18.

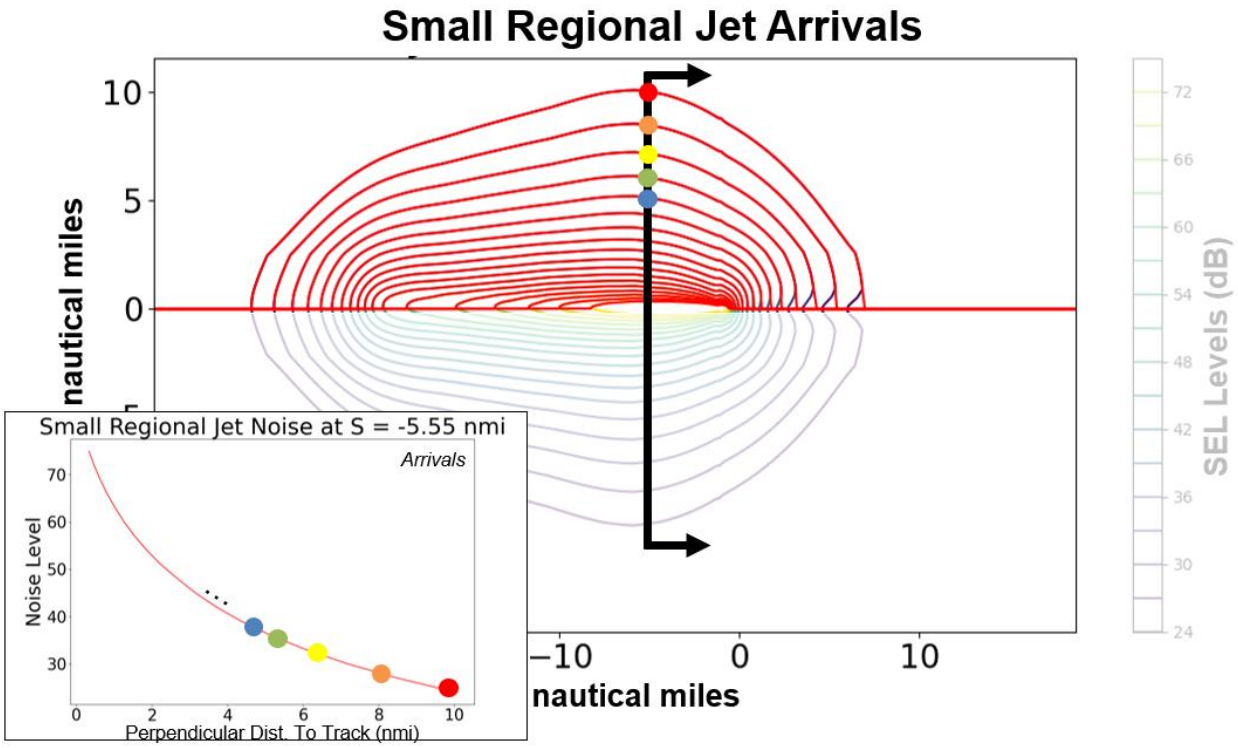


Figure 18. *Process for generating interpolation functions from AEDT results. (Note: plot for illustration purposes only)*

3.3.2 Implementation of Noise Approximation Method

The key insight behind this thesis' application of the "half-width" method is that knowing the perpendicular distance from an observation point to a flight ground track allows us to interpolate the noise level at that observation point, as shown below in Figure 19.

The codebase underlying this thesis calculates noise results on an observation grid specified around the airport. The example shown in Equation 3 specifies a grid extending 10 nmi from the airport in all four cardinal directions, with observation points spaced every 0.25 nmi.

```
gridX, gridY = np.meshgrid(np.arange(-10,10,0.25),np.arange(-10,10,0.25))
```

Equation 3. *Code to create a grid of observation points around the airport of interest.*

For a given flight ground track, the noise at each observation point is interpolated from the function associated with the perpendicular along-track point. For example, in Figure 19 below, the observation point at Hull, Massachusetts is associated with the noise function for an along-track point of *s = Y nmi*.

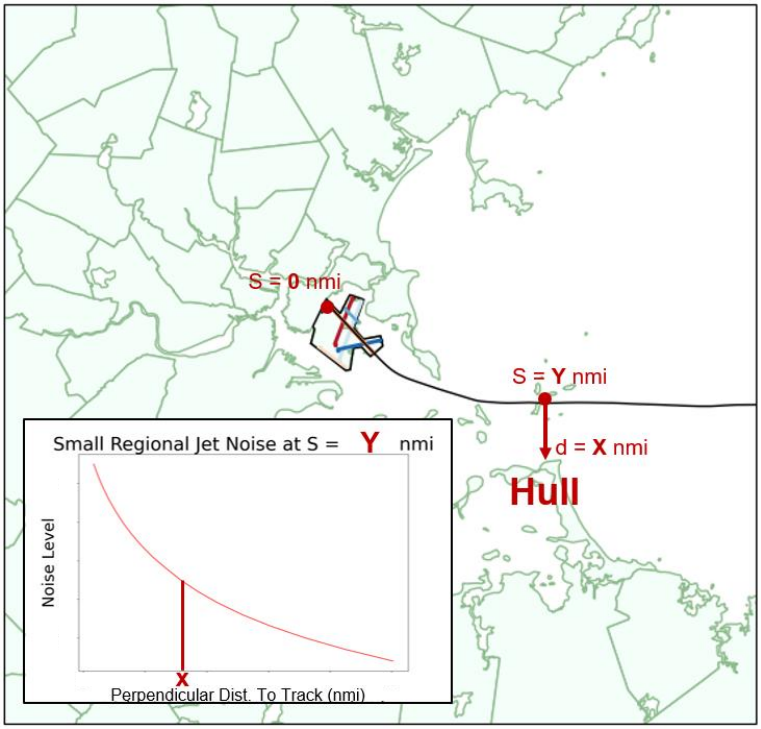


Figure 19. *Visual example of calculating noise at an observation point on Hull, MA.*

The process of finding the perpendicular along-track points for all observation points on the grid can be simplified under the assumption that the perpendicular points are also the along-track points with the minimum distance to the observation points.

```
dists = scpd.cdist(observation_points,along_track_points,'euclidean')
min_inds = dists.argmin(axis=1)
min_dists = np.take_along_axis(dists, min_inds[:,None], axis=1).flatten()
```

Equation 4. *The* cdist *function from the* SciPy Spatial Distance (scpd) *package [6] can be used to quickly calculate the distances between all along-track points and all observation points. Then, the minimum distances for each observation point can be determined using* argmin *and* take_along_axis *from the* NumPy *package [7]. The variable* min_inds *points to the corresponding along-track points.*

The combination of observation locations and their corresponding along-track points and distances is enough to interpolate from the AEDT-acquired noise functions (Recall, there is a noise function with respect to distance for every combination of aircraft type, arrival/departure status, and along-track point).

3.4 Validation of Method Results

The noise approximation method was validated by comparing representative arrival and departure tracks for two aircraft types with different noise signatures. An arrival with a tight downwind-upwind leg is presented below in Figure 20 for a typically louder aircraft, such as an MD-88, and in Figure 21 for a typically quieter aircraft, such as an E145. The most significant difference is a "wrinkling" in the contours between the downwind and upwind legs of the flight. Additionally, approximated noise is underestimated early in the arrival around the downwind leg, in the North-East of both figures below.

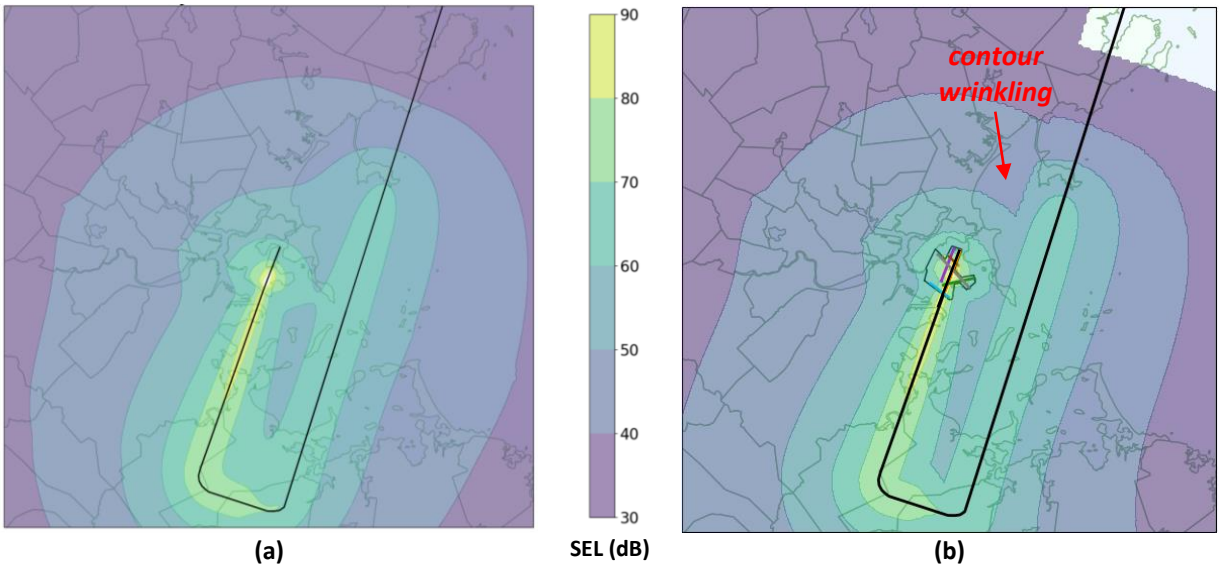


Figure 20. *Arrival flight comparison for an Older Jet (OJ), modeled by an MD-88, between* ***(a)*** *AEDT and* ***(b)*** *the noise approximation method used in this thesis.*

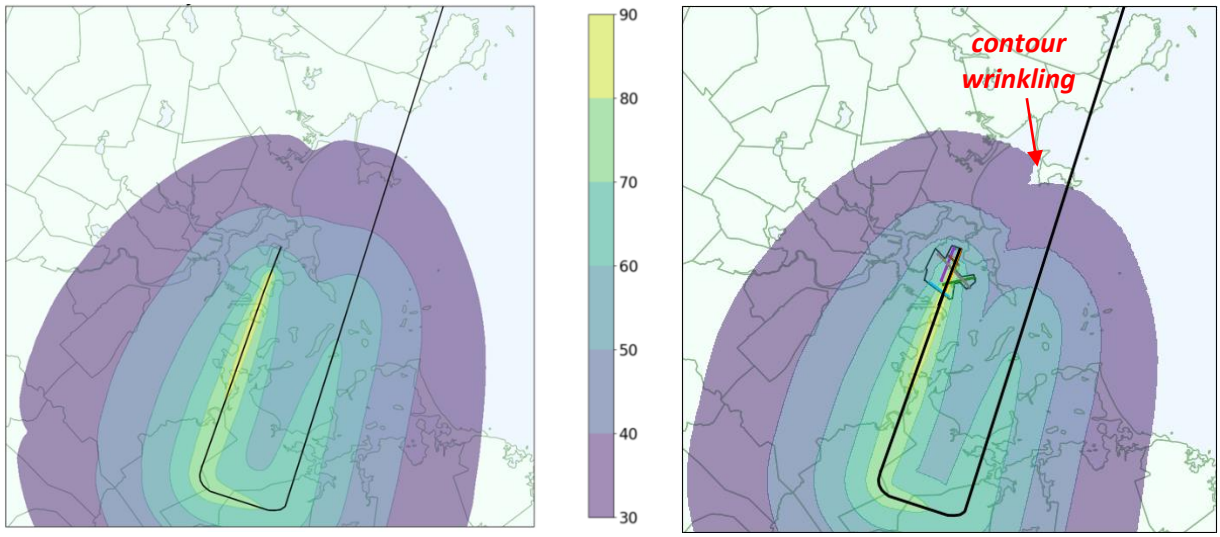


Figure 21. *Arrival flight comparison for a Small Regional Jet (SRJ), modeled by an E145, between* ***(a)*** *AEDT and* ***(b)*** *the noise approximation method used in this thesis.*

AEDT-approximation comparisons were also made for a representative departure track and the same aircraft types. Figure 22 shows the comparison made for a modeled versus approximated MD-88 and Figure 23 shows the comparison made for an E145. The most significant difference is once again a "wrinkling" of the contours on the inside of turns.

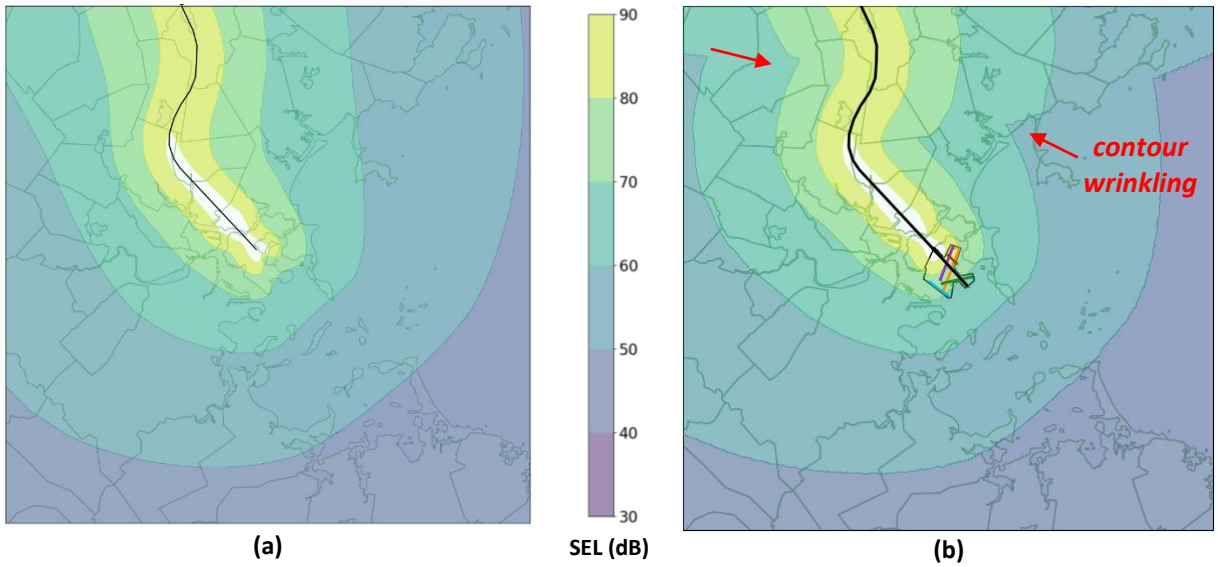


Figure 22. *Departure flight comparison for an Older Jet (OJ), modeled by an MD-88, between* ***(a)*** *AEDT and* ***(b)*** *the noise approximation method used in this thesis.*

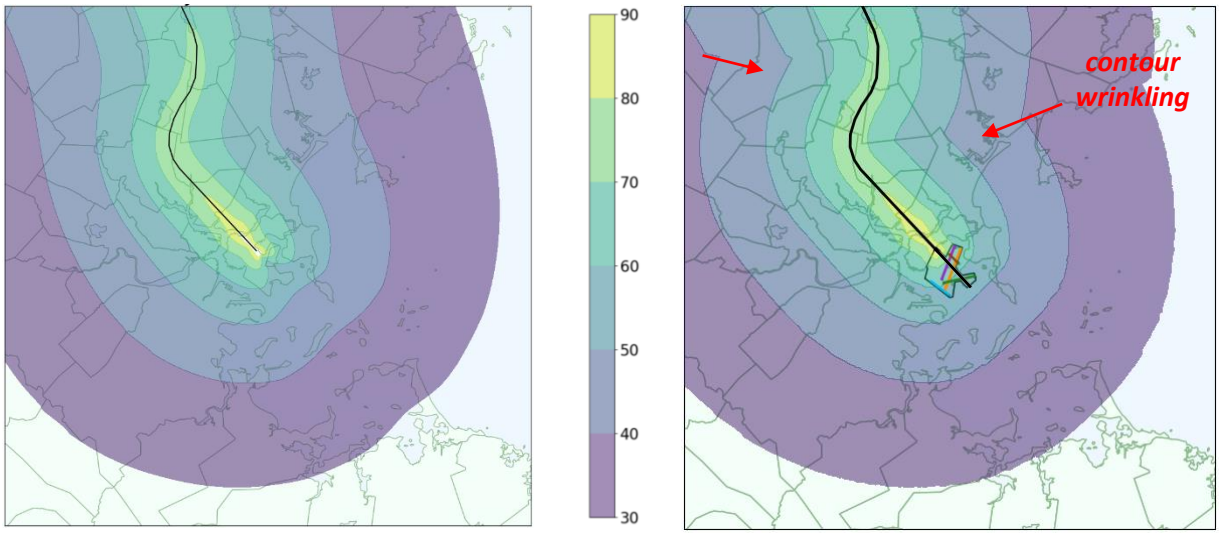


Figure 23. *Departure flight comparison for a Small Regional Jet (SRJ), modeled by an E145, between* ***(a)*** *AEDT and* ***(b)*** *the noise approximation method used in this thesis.*

Generally speaking, differences between the approximation method and AEDT results fall within two categories: A) modeling differences and B) approximation implementation differences. Modeling differences include path geometry and bank angle assumptions. Because the approximated contours are derived from straight-in-out flights, they do not account for bank angle or varying noise-exposure time on the inside or outside of a turn, as noted by [5]. Approximation implementation also has an effect on modeled noise—the results presented in this thesis are generated by linking observation points to their perpendicular "along-flight-track" points by assuming these points are also the closest "along-flight-track" points. This assumption is less accurate along the inside of a turn.[3]

Ultimately, the differences in modeled noise using the "half-width" method are small and local. When the "half-width" method is applied to hundreds or thousands of flights, effects at this scale are assumed insignificant compared to the broader benefit of accurately modeling the

[3] Note that the method accuracy results presented [5] are not necessarily produced with the same method implementation.

natural dispersion of flight tracks and operations beyond the capabilities of using a "representative day."

4 – Noise Results

4.1 Airport (KBOS)

This thesis presents noise results from Boston Logan International Airport (KBOS). KBOS is an interesting airport for studying noise because flights arrive and depart in a variety of directions due to the airport's location along the Boston Harbor and Atlantic Ocean, which subjects it to highly variable wind conditions. KBOS has a total of six runways operating in four "primary configurations": North-East, South-East, South-West, and North-West, as summarized in Table 2 below [8]. An airport diagram graphically featuring the six runways and primary configurations is shown in Figure 24.

KBOS Primary Configuration	Departure Runways	Arrival Runways	Operational Percentage
North-East	9, 4L, 4R	4L, 4R	18%
South-East	15R, 14, 9	15R, 15L	17%
South-West	22L, 22R	22L, 22R, 27	28%
North-West	33L, 27	33L, 32, 27	37%

***Table 2**. KBOS configuration name, runway usage, and operational percentage; adapted from [8].*

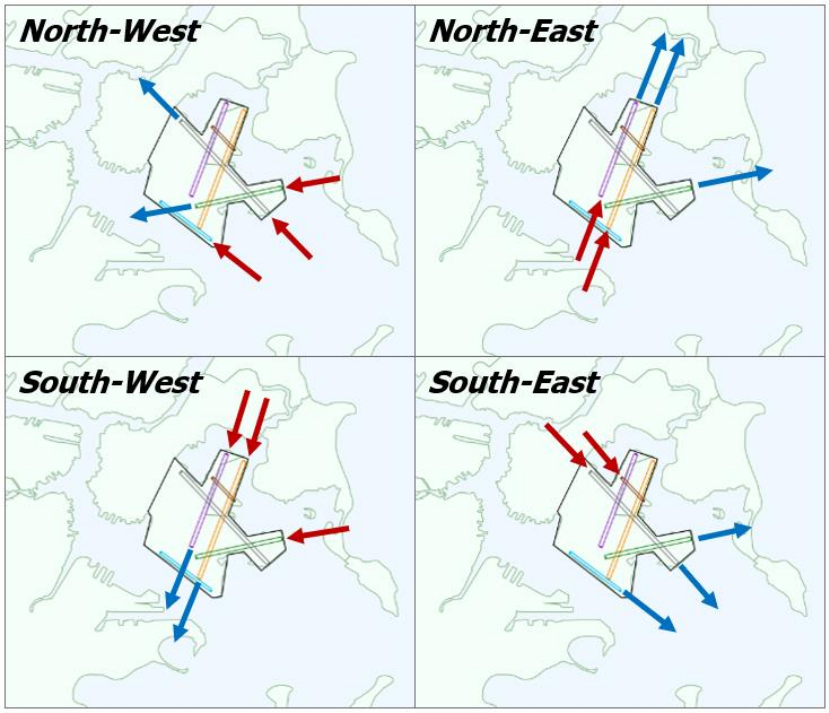


Figure 24. *KBOS primary runway configurations diagram; adapted from [8].*

4.2 DNL Estimation from Large Dataset

This framework was used to produce an estimate for DNL using over 200,000 flights recorded at KBOS in 2016. The timeframe of data available for the author's use is pictured in Figure 25**.** The quantity of recorded flights is so large that it can be assumed to produce a DNL result resembling that of an annual average.

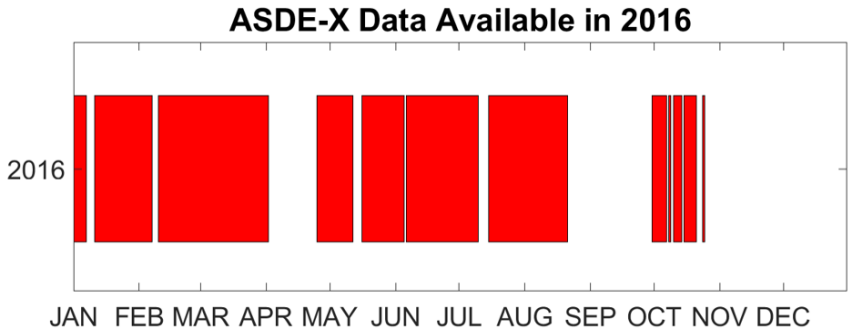


Figure 25. *Timeframe of ASDE-X data available and used for KBOS DNL calculation in 2016.*

Annual average DNL was calculated by approximating noise for all flights in this dataset, applying nighttime noise penalties as needed and summing noise at each geographic grid cell according to Equation 1, reproduced in a modified form below.

$$DNL = 10\log\frac{1}{\Delta t_{2016_{available}}}\left(\sum_{day\ flights}^{2016_{available}} 10^{\frac{SEL}{10}} + \sum_{night\ flights}^{2016_{available}} 10^{\frac{SEL+10}{10}}\right)$$

Instead of dividing by the number of seconds in a day, the DNL estimate was divided by the number of seconds spanning the dataset, accounting for coverage "outages". Coverage outages are common in the ASDE-X system and refer in this work to periods of time in which there is no flight data recorded. The source of these outages is unclear—they could be caused by a pause in system operation at the airport level or the system may have been operational but the data unsaved. Finally, it is possible that the lack of flight data is a real reflection of a lack of airport activity.

Outages were detected on an hour-by-hour basis. The total time duration simulated was discretized into a series of hour-long bins. If a flight occurred within a given hour, said hour was marked as "active." Figure 26 shows a histogram of "active" hours in the 2016 dataset at KBOS. The most noticeable dip in data occurs between *1am* and *5am*, with less than half of hours between *3am* and *4am* having flights recorded. It is impossible to know whether the lack of flights is real or perceived but because flight activity at KBOS is so low in the early morning regardless, "inactive" bins between the hours of *1am* and *5am* were overridden and counted towards the Δt time value used in DNL. Under this condition there were altogether approximately 201 days' worth of active time bins in this dataset. Δt in the DNL equation above was therefore set to the number of seconds in 201 days.

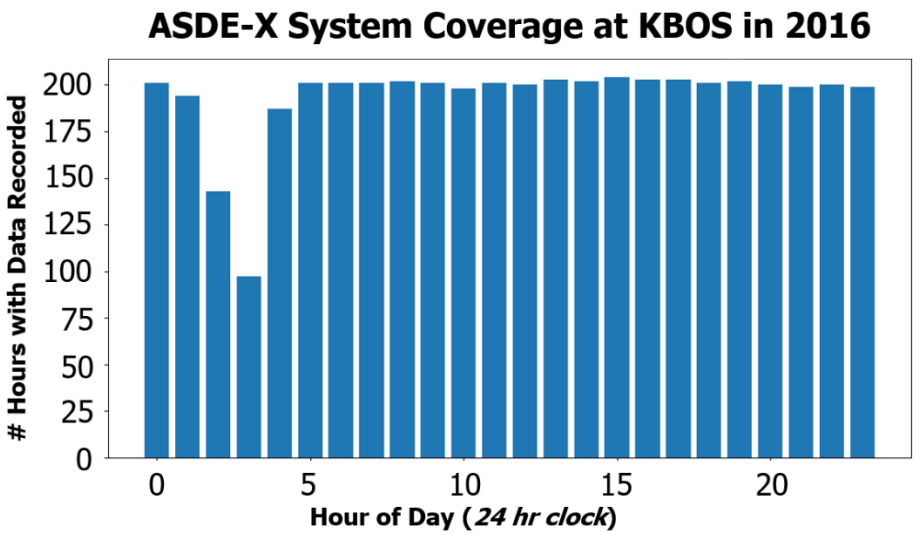


Figure 26. *Plot showing in which hours of the day flights were recorded at KBOS in 2016 (out of the available ASDE-X dataset). Dips in the bar plot show that flights are most often not recorded between the hours of* 1am *and* 5am*.*

Figure 27 shows the resulting estimate for DNL at KBOS. Approximately 210,500 flights were successfully analyzed on a 20 nmi by 20 nmi grid with 0.25 nmi spacing in a total noise computation time of 12.35 hours. As an aside, [9] states that 391,222 airport operations occurred at KBOS in the entirety of 2016. Weighting this value by 54.9% (201 days analyzed / 366 days in 2016) gives an estimated 214,851 flights occurring within the time-frame of data available (assuming an even distribution of flights throughout the year). This matches fairly well with the 210,500-number analyzed, suggesting that the data pre-processing and cleaning filters perform well at detecting real flights. The difference of approximately 4,000 flights could be due to the 2016 dataset not including the peak travel months of November and December.

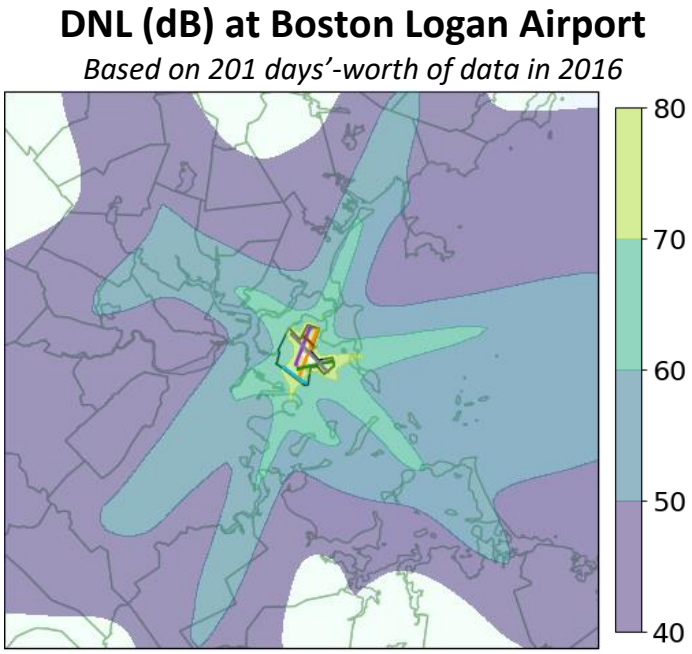


Figure 27. *DNL for KBOS using data from 2016 computed on a 20 nmi x 20 nmi grid with 0.25 nmi grid spacing.*

This DNL estimate shown in Figure 27 is then compared in Figure 28 against annual average DNL results produced for the same year by Harris, Miller, Miller & Hanson Inc. (HMMH), provided by the author of [10]. The similarities between these two maps suggests that the fast, "half-width" approach produces realistic noise results.

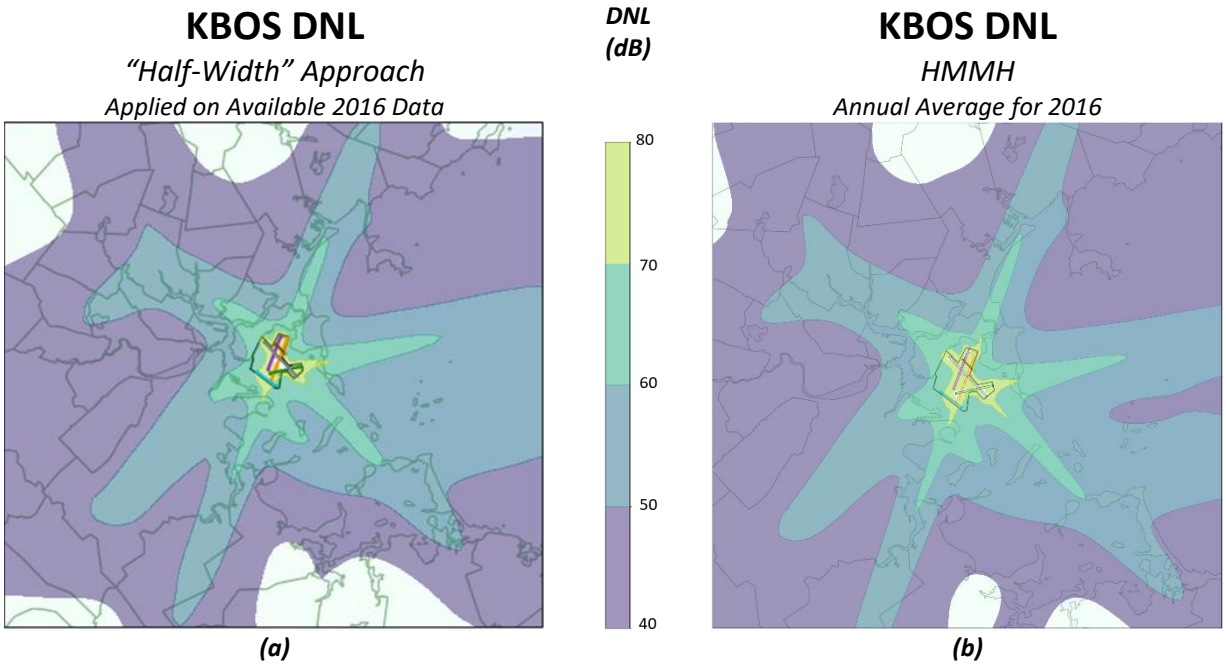


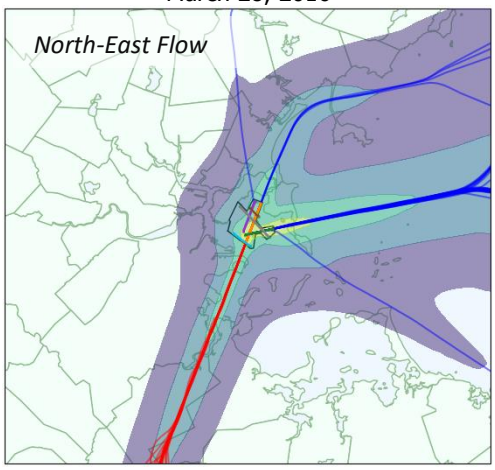
Figure 28. *DNL for KBOS as modeled by* ***(a)*** *the "half-width" method and* ***(b)*** *2016 annual average modeling conducted by HMMH.*

4.3 Hourly DNL

This method can also be used for shorter time frames to illustrate noise variability due to evolving runway configuration, airport traffic levels, and fleet mix. The results below show an hour-by-hour analysis conducted for March 28th, 2016. Within this analysis, "DNL" for each hour is calculated assuming Δt = 3600s and applying the standard 10dB penalty for every flight within hours between 10pm and 7am. Note that results before 7am are omitted due to low traffic as recorded by the ASDE-X system. The day begins with a primarily North-East traffic flow with flights arriving on runway 4R and departing from runways 4L, 4R, and 9. Departures from runway 15R in the hours to follow also suggest a hint of South-East flow. Between *5pm* and *6pm*, noise exposure is more uniformly widespread around the airport as the traffic flows shift from the North-East to North-West. The runway configuration continues to evolve throughout the evening into a primarily South-West flow after *9pm*.

Hourly DNL (7am)

March 28, 2016

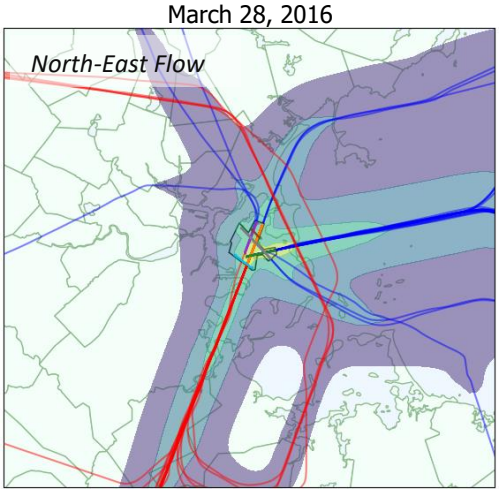


DNL (dB)

80
70
60
50
40

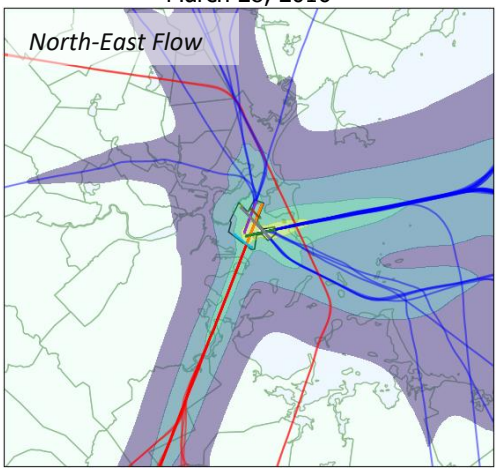
Hourly DNL (8am)

March 28, 2016



Hourly DNL (9am)

March 28, 2016

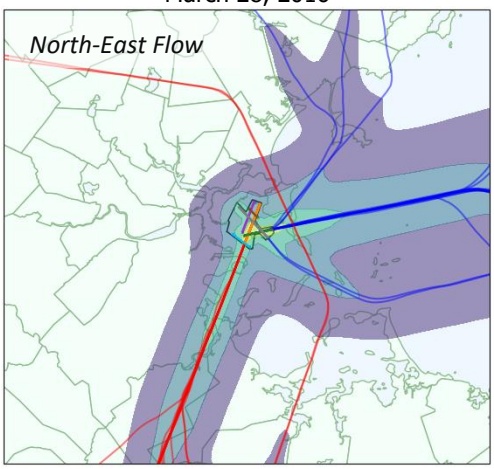


DNL (dB)

80
70
60
50
40

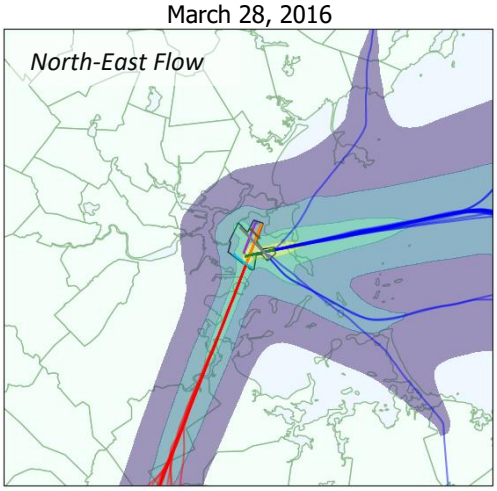
Hourly DNL (10am)

March 28, 2016



Hourly DNL (11am)

March 28, 2016

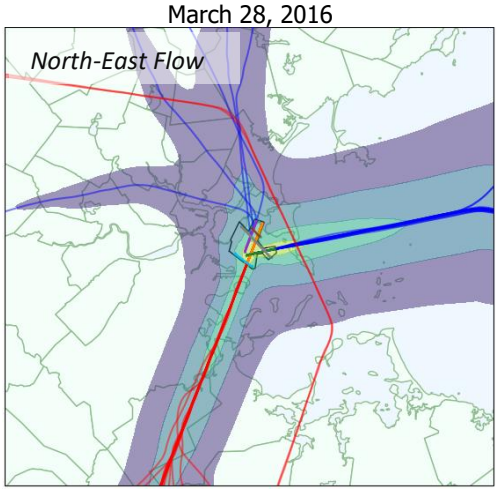


DNL (dB)

80
70
60
50
40

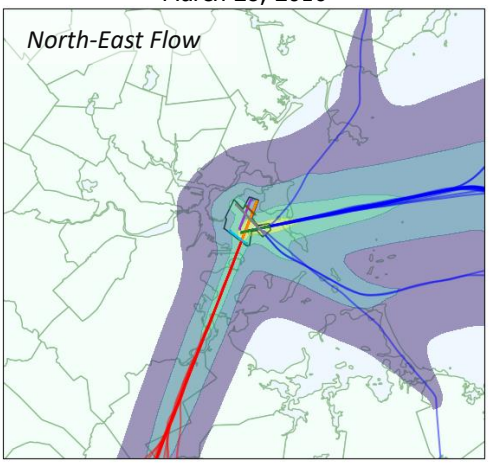
Hourly DNL (12pm)

March 28, 2016



Hourly DNL (1pm)

March 28, 2016

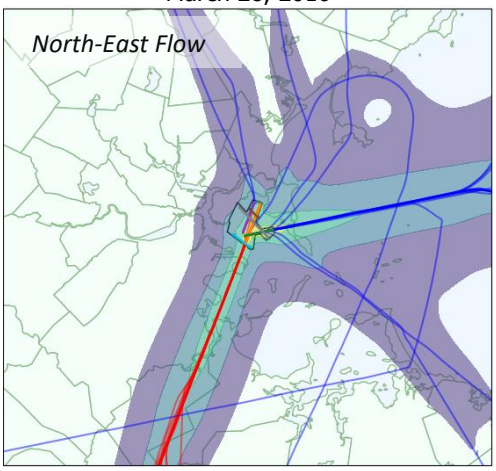


DNL (dB)

80
70
60
50
40

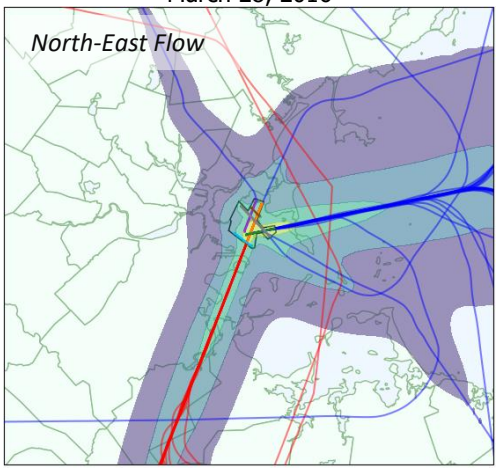
Hourly DNL (2pm)

March 28, 2016



Hourly DNL (3pm)

March 28, 2016

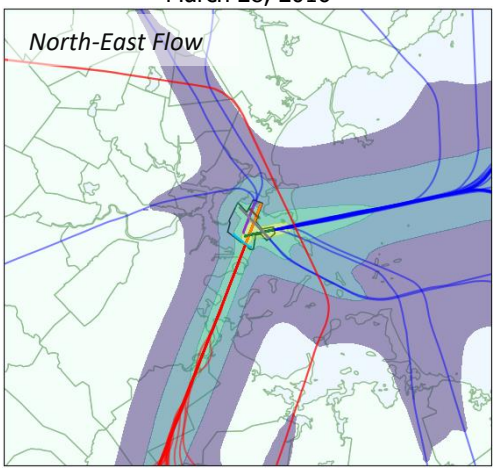


DNL (dB)

80
70
60
50
40

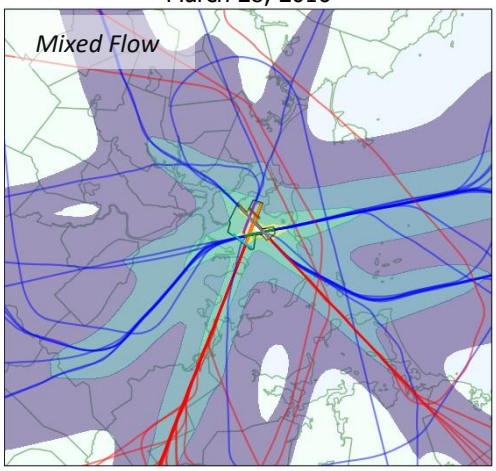
Hourly DNL (4pm)

March 28, 2016



Hourly DNL (5pm)

March 28, 2016

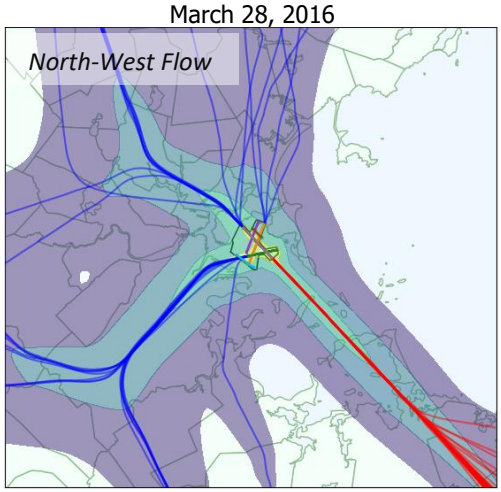


DNL (dB)

80
70
60
50
40

Hourly DNL (6pm)

March 28, 2016



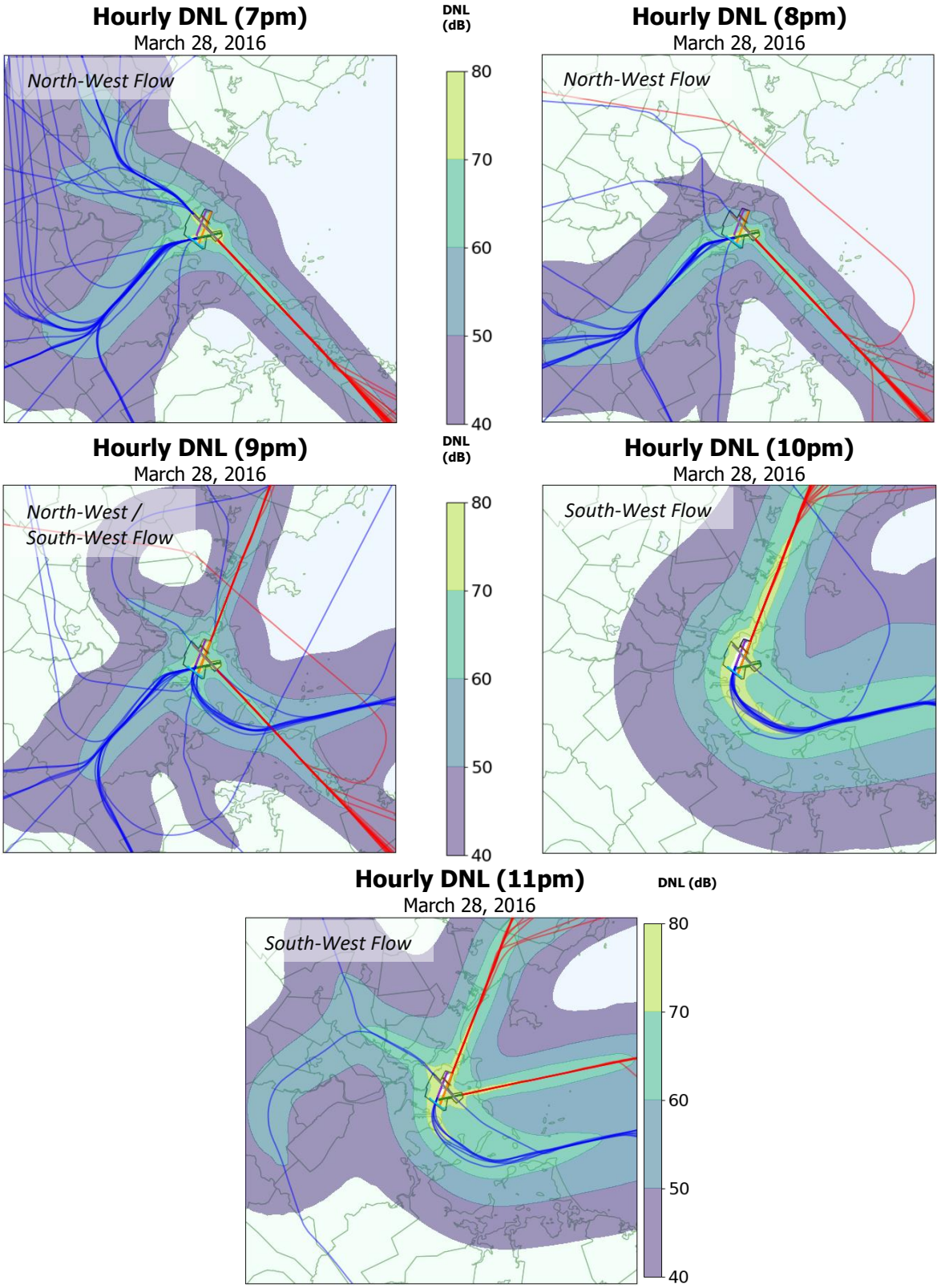
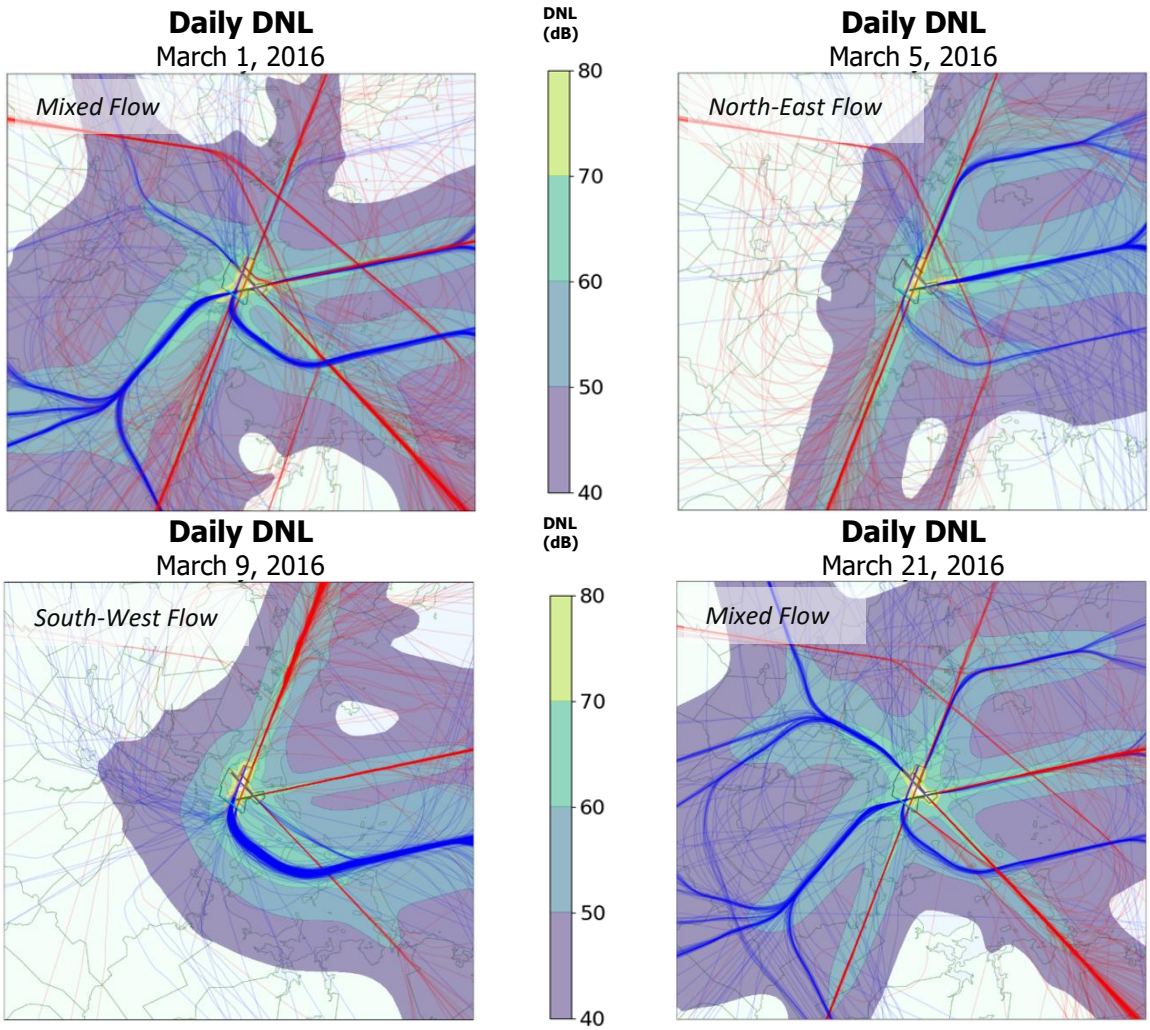


Figure 29. *Hourly DNL from 7am through 11pm on March 28, 2016 at KBOS. Arrival flight tracks are shown in red and departures are shown in blue.*

4.4 Daily DNL

DNL can also be calculated on a daily basis, as presented below in Figure 30. Here, a selection of days from March 2016 is presented, using each day's number of "active" hours as the respective Δt estimates. The variation in noise between the different days shows the effect that runway configuration has upon noise experienced by the airport-surrounding public. March 5th, 9th, and 29th are examples of days that operated in one dominating traffic flow: North-East, South-West, and North-West, respectively. In comparison, March 1st, 21st, and 28th all operated in a mix of traffic flows. The increased runway variety led to a more uniform spread of noise around the airport.



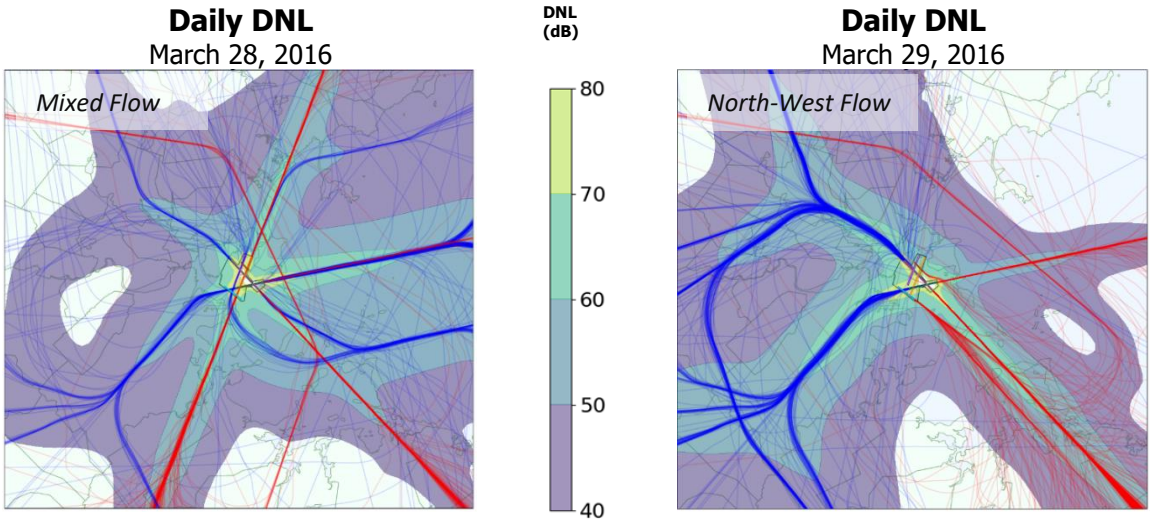


Figure 30. *A selection of daily DNL results from March, 2016 at KBOS. Arrival flight tracks are shown in red and departures are shown in blue.*

4.5 Daily N_{60}

The process is applied to flights using $L_{A,max}$ metric contours and the results aggregated into the $N_{60\ day,\ 50\ night}$ metric. Results are presented for March 21st and March 29th in Figure 31.

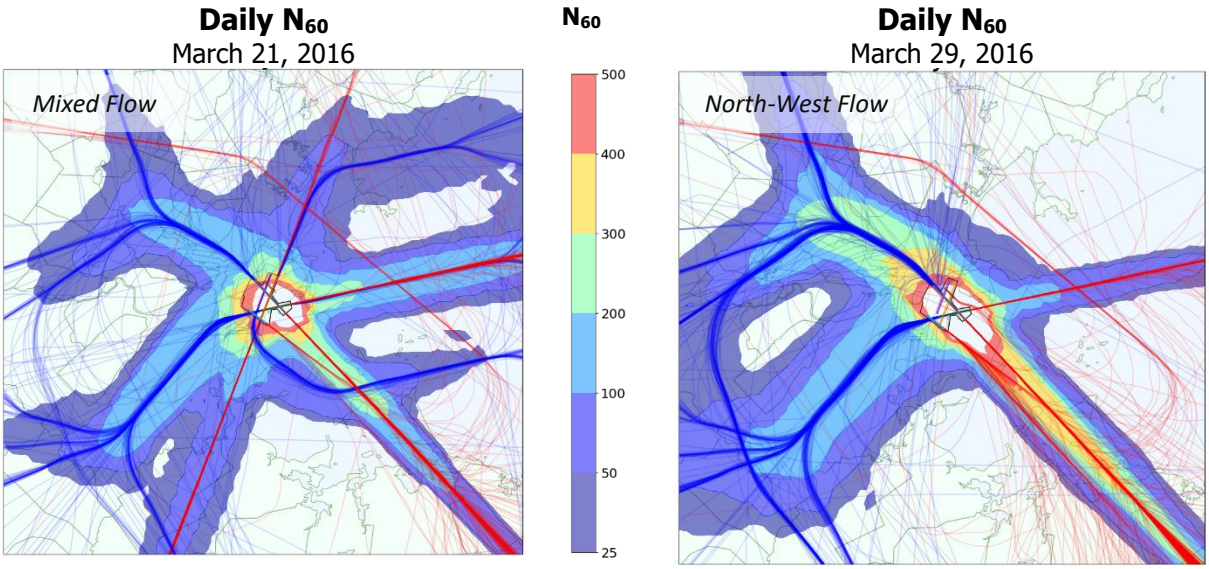


Figure 31. *A selection of daily N_{60} results from March, 2016 at KBOS. Arrival flight tracks are shown in red and departures are shown in blue.*

4.6 Location-Based Analysis

The same analysis can be applied to a specific location on the geographic grid to determine the frequency of noise events at a particular location. This type of analysis may be useful in

determining the effects of sound level and event frequency on human annoyance. An example was conducted between 7pm and 8pm on March 28th, 2016 at a target location surrounding KBOS, shown in Figure 32.

Figure 32. *The target location for noise analysis is marked as a red diamond.*

Figure 33 illustrates a timeline of noise events at this location above the 40 dB_A $L_{A,max}$ threshold (events below this threshold are marked temporally on the timeline). Peak noise results from two example flights within the hour are then shown in Figure 34.

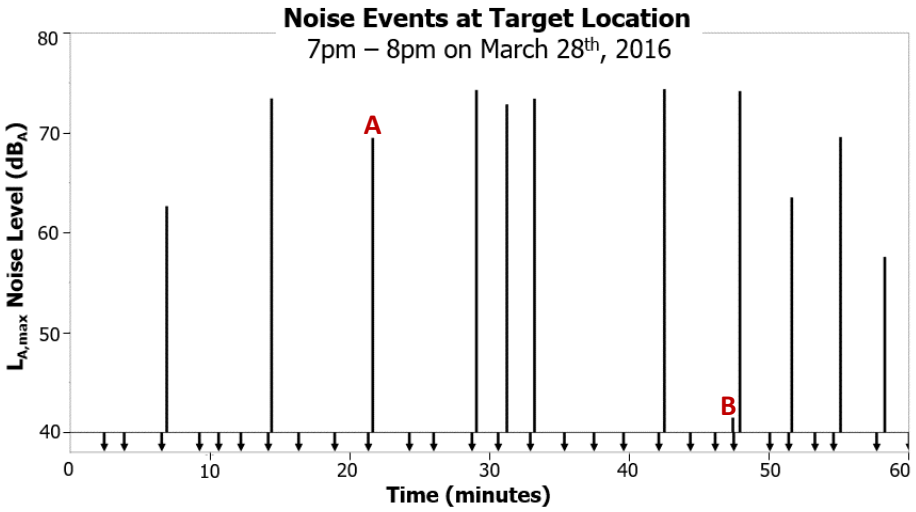


Figure 33. *Timeline of noise events (based on time of closest approach) at a target location between 7pm and 8 pm on March 28th, 2016 at KBOS. Events at or below 40 dB_A are marked temporally by a downwards arrow. Flights with a closest point of approach greater than 4 nautical miles from the target are omitted.*

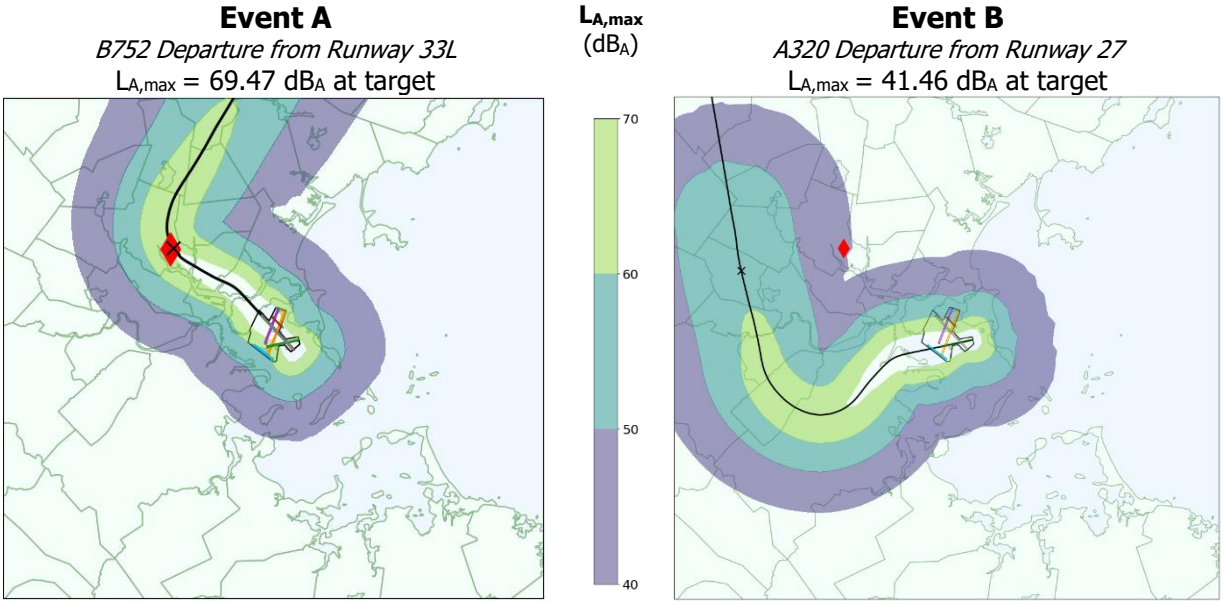


Figure 34. *Peak noise from two example events at a target location between 7pm and 8pm on March 28th, 2016. The target location is shown as a red diamond and the closest point of approach to the target location is shown as a black 'X'.*

5 – Conclusion and Future Work

This work presents a framework by which actual flight trajectories are used to estimate existing noise metrics as well as evaluate high-frequency changes in the geospatial distribution of noise surrounding an airport. This research addresses two key limitations of existing standard noise modeling: first, the computational speed of the implemented "half-width" method means that "representative" assumptions need not be made for aggregate metrics such as annual average DNL. Instead, the approach allows us to model and summate noise from every flight in a large dataset within a reasonable timeframe. Second, the same modeling approach allows for easy breakdown of noise into temporal segments. Viewing the progression of hourly or daily noise reveals insights into the evolution of airport activity and the resulting impact on surrounding communities.

The results presented in this thesis motivate several directions for future research. The library of noise contours could be expanded (with no significant runtime impacts) to include more granular aircraft type bins or procedure/runway specific results. *Appendix B – Flight Profiles Used for Approximation Contours* shows how AEDT-default profiles do not perfectly reflect real-world data at a given airport. In the future, ASDE-X data could be used to generate more tailored flight profiles and resulting noise functions for combinations of aircraft type and runway. These improvements would increase the overall accuracy of noise results.

Location-based analysis such as presented in Figure 33 could also be coupled with geographic noise complaint data to study noise annoyance factors. For example, the peak noise ($L_{A,max}$),

integrated noise (SEL), and frequency of overflights above a certain level in a given time period could all be quickly calculated and evaluated as potential complaint-triggering factors.

Appendix A – ASDE-X System

All sources of flight data, whether ASDE-X, ADS-B, ETMS, etc. must be cleaned of sensor noise and formatted to interface with the noise approximation code. Each data source, however, will require different pre-processing techniques to shape the data for each flight into unique data structures. This appendix describes pre-processing techniques specific to data coming from the ASDE-X system. ASDE-X, or Airport Surface Detection Equipment, Model X, is a system currently available at over 30 airports nationwide. ASDE-X operates by fusing information from radar, multilateration, and satellite-based sensors over an approximately 15-mile radius from a given airport. ASDE-X messages contain flight information such as a vehicle's position, speed, runway, and type.

ASDE-X Message Parsing

The first step in preprocessing ASDE-X data is to decode the individual messages from their XML format. As a practical matter, parsing messages results in an approximately 8x reduction in file-size,[4] which alleviates some of infrastructural challenges associated with ASDE-X data storage. Each ASDE-X message pertains to a specific airport and contains one or more '*positionReports*' that describe the state of a ground or air vehicle at a given time. An example ASDE-X message for Daniel K Inouye International Airport (PHNL) in Honolulu, Hawaii is shown below in Figure 35.

[4] July, 2016 data at KBOS was reduced from 19.4 GB to 2.47 GB. ASDE-X messages often come with a non-XML header, not shown in Figure 35, that further increases the starting file-size and must be stripped from the messages.

```
<airport>PHNL</airport>
<positionReport full="true">
        <time>2016-07-04T03:00:00.000Z</time>
        <track>3462</track>
        <flightId>
                <aircraftId>UNKN</aircraftId>
        </flightId>
        <flightInfo>
                <tgtType>unknown</tgtType>
                <runway>0</runway>
        </flightInfo>
        <position>
                <x>-1605</x>
                <y>1075</y>
                <latitude>21.33057757280767</latitude>
                <longitude>-157.9426560178399</longitude>
                <altitude>6</altitude>
        </position>
        …
```

Figure 35. *Sample XML message from the ASDE-X system, with tags shown in blue, attributes in green, and values in pink.*

Packages exist in several programming languages to process XML data, but the values of relevant vehicle states can also be extracted by string searching without this external overhead. Python was selected for this task due to its string search speed.

Each *positionReport* tag acts as an aircraft "observation" and is individually searched for the following relevant flight parameters: observation time, latitude, longitude, altitude, aircraft type, and callsign. After extracting these values (which may not all occur within a given *positionReport*) from the XML format, the observations must be classified as part of an existing flight on-file or a new flight. This classification is done primarily using the *track* element, which is a number between 0 and 4095 that is temporarily associated with a particular airport operation (such as an arriving or departing flight—in Figure 35 the track value is *3462*). All *positionReport* contain the *track* tag and it is because of this consistency that the *track* value was chosen as the primary identifying feature by which to group observations.

Because *track* values are reused, it is also necessary to detect when resets occurred. This was first done by tracking an XML keyword *tse* (track service end). The presence of an affirmative *tse*

value signals a close-out of the existing *track* data structure and initiates the opening of a new data structure. Relying on the *tse* keyword alone, however, is an unreliable method and may result in missing some *track* closeouts. Therefore, closeout logic was also supported by checking that the time between *track* observations is always less than 10 minutes and that the *acType* (aircraft type) and *aircraftId* (callsign) values remain consistent across observations. Ultimately, the information for each aircraft operation is stored in a Python dictionary containing the aircraft type and lists of corresponding observation times, longitudes, latitudes, and altitude values.

Appendix B – Flight Profiles Used for Approximation Contours

This section illustrates the arrival and departure altitude profiles for each aircraft type bin used to generate contour approximation functions in AEDT. The plots also illustrate a sample of corresponding altitude profiles from ASDE-X data at KBOS**.**

A320 Family Bin Profiles

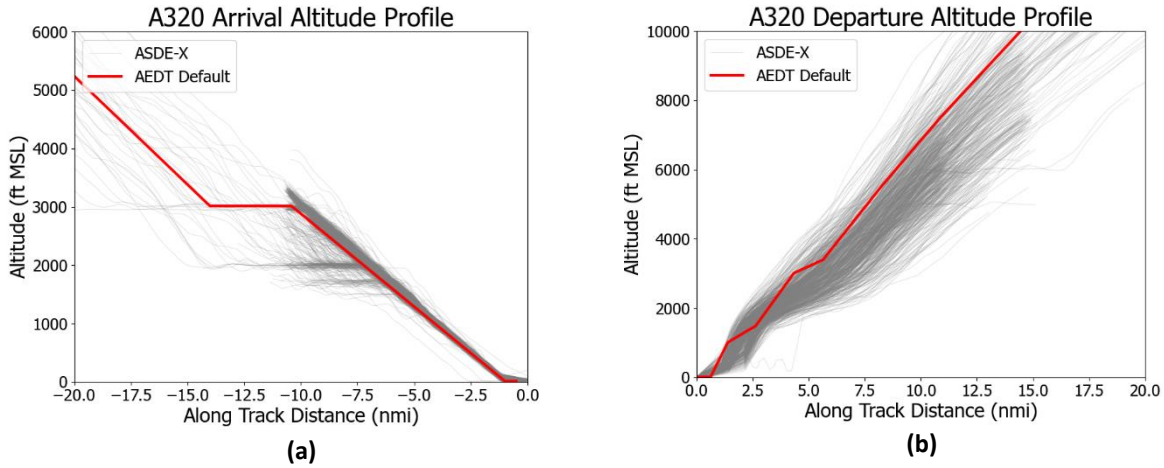


Figure 36. *Altitude profiles for an* ***(a)*** *arrival and* ***(b)*** *intermediate stage-length departure for aircraft within the A320 Family bin. The AEDT default profiles (modeled by an A320) are shown in red and ASDE-X altitude profiles (for all aircraft within the A320 Family bin) are shown in gray.*

B737 Family Bin Profiles

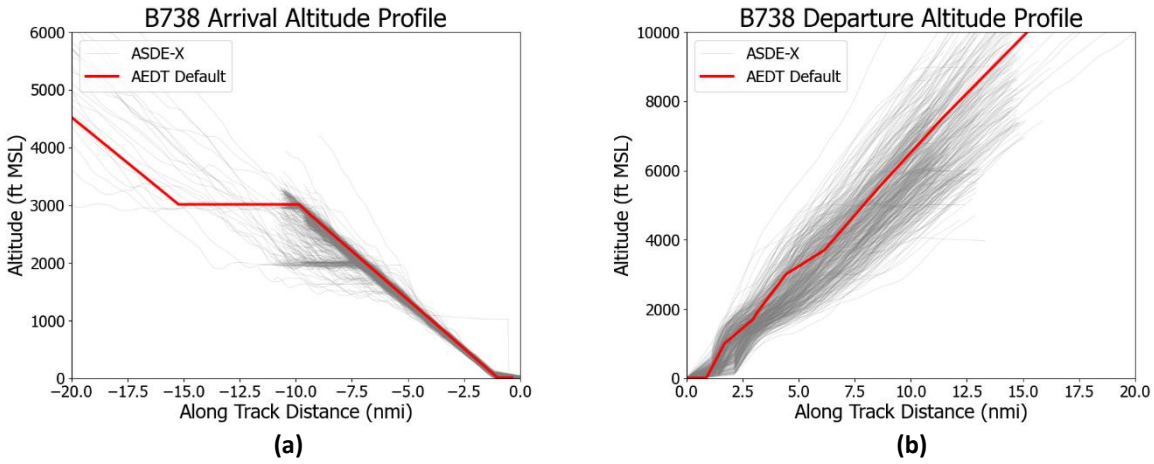


Figure 37. *Altitude profiles for an* ***(a)*** *arrival and* ***(b)*** *intermediate stage-length departure for aircraft within the B737 Family bin. The AEDT default profiles (modeled by a B738) are shown in red and ASDE-X altitude profiles (for all aircraft within the B737 Family bin) are shown in gray.*

B757 Family Bin Profiles

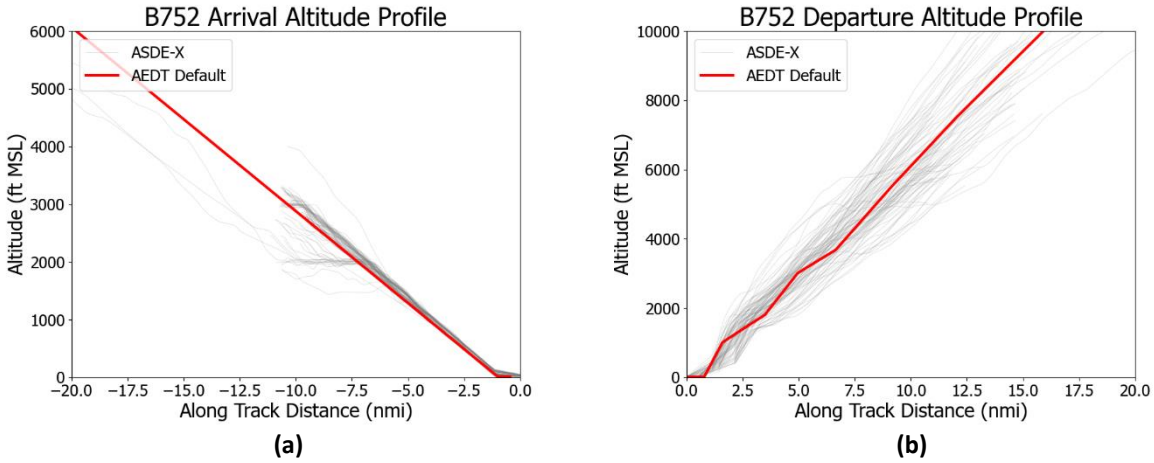


Figure 38. *Altitude profiles for an* ***(a)*** *arrival and* ***(b)*** *intermediate stage-length departure for aircraft within the B757 Family bin. The AEDT default profiles (modeled by a B752) are shown in red and ASDE-X altitude profiles (for all aircraft within the B757 Family bin) are shown in gray.*

Large Regional Jet Bin Profiles

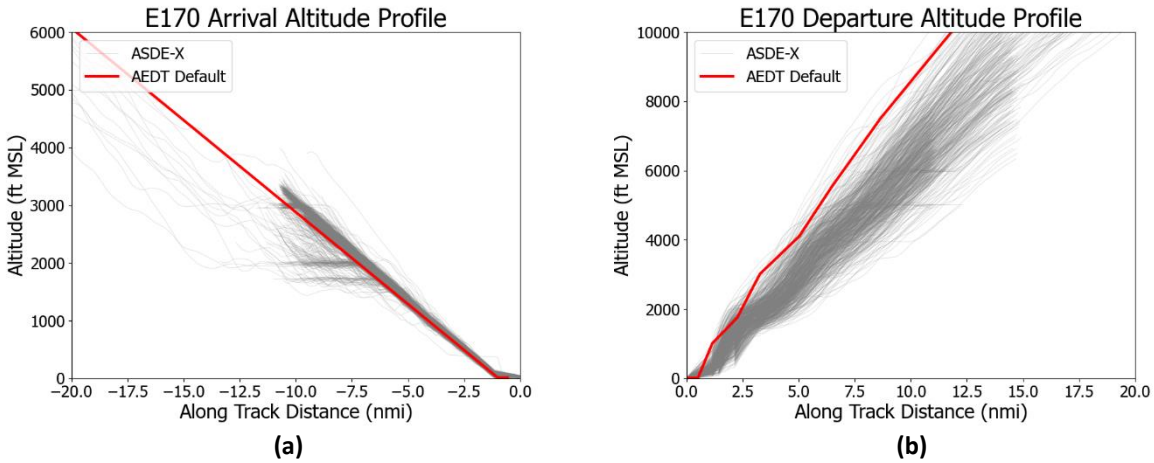


Figure 39. *Altitude profiles for an* ***(a)*** *arrival and* ***(b)*** *intermediate stage-length departure for aircraft within the Large Regional Jet bin. The AEDT default profiles (modeled by a E170) are shown in red and ASDE-X altitude profiles (for all aircraft within the Large Regional Jet bin) are shown in gray.*

Older Jet Bin Profiles

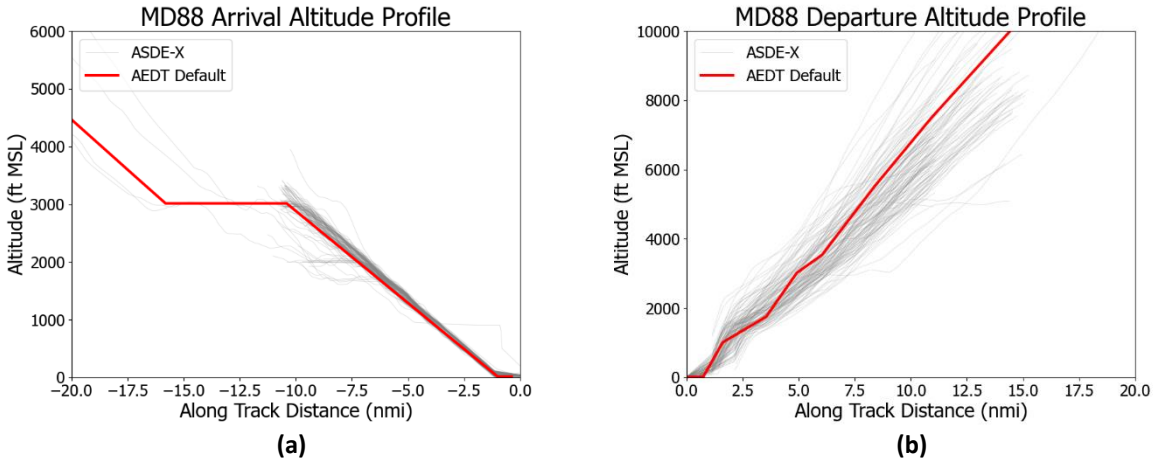


Figure 40. *Altitude profiles for an* ***(a)*** *arrival and* ***(b)*** *intermediate stage-length departure for aircraft within the Older Jet bin. The AEDT default profiles (modeled by an MD-88) are shown in red; ASDE-X altitude profiles (for all aircraft within the Older Jet bin) are modeled in gray.*

Piston Engine Bin Profiles

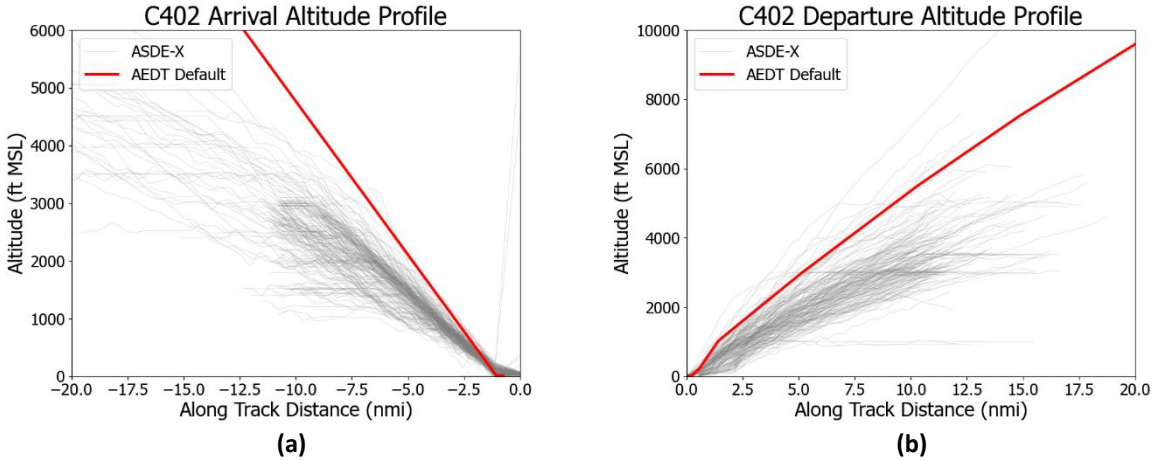


Figure 41. *Altitude profiles for an* ***(a)*** *arrival and* ***(b)*** *intermediate stage-length departure for aircraft within the Piston Engine bin. The AEDT default profiles (modeled by an C402) are shown in red; ASDE-X altitude profiles (for all aircraft within the Piston Engine bin) are shown in gray.*

Small Regional Jet Bin Profiles

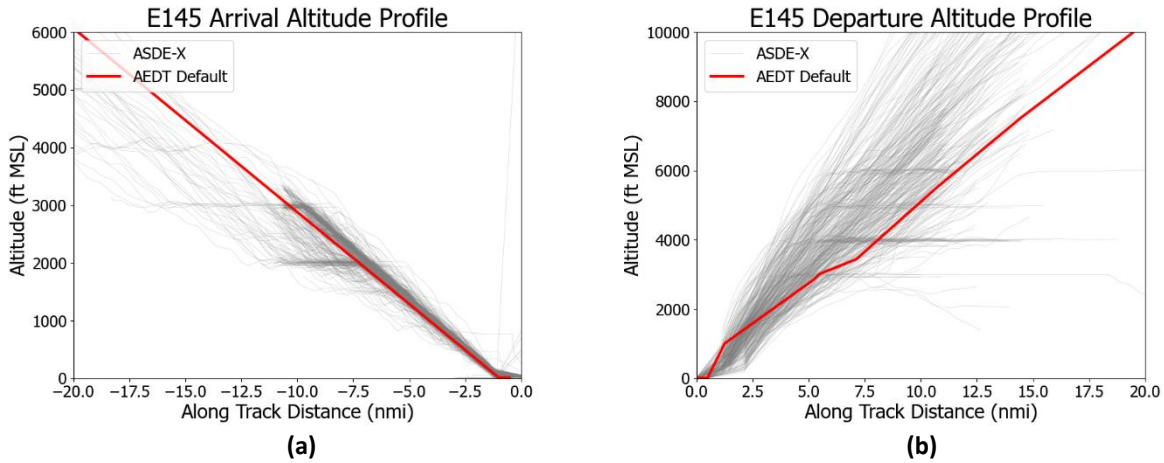


Figure 42. *Altitude profiles for an* ***(a)*** *arrival and* ***(b)*** *intermediate stage-length departure for aircraft within the Small Regional Jet bin. The AEDT default profiles (modeled by an E145) are shown in red; ASDE-X altitude profiles (for all aircraft within the Small Regional Jet bin) are shown in gray.*

Twin Aisle Bin Profiles

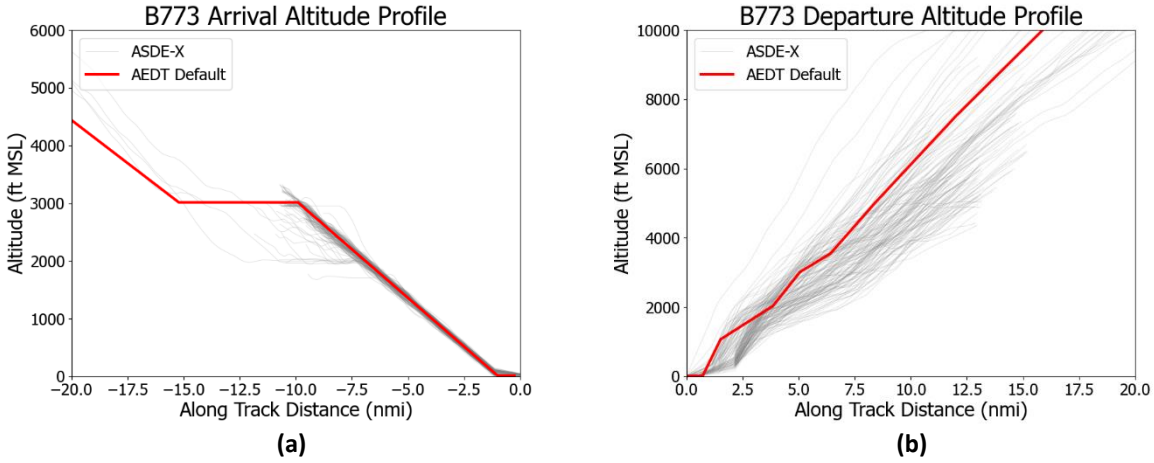


Figure 43. *Altitude profiles for an* ***(a)*** *arrival and* ***(b)*** *intermediate stage-length departure for aircraft within the Twin Aisle family bin. The AEDT default profiles (modeled by a B773) are shown in red; ASDE-X altitude profiles (for all aircraft within the Twin Aisle bin) are shown in gray.*

Bibliography

[1] Federal Aviation Administration, "Fundamentals of Noise and Sound," 13 July 2020. [Online]. Available: https://www.faa.gov/regulations_policies/policy_guidance/noise/basics/. [Accessed 1 August 2020].

[2] M. Brenner, "Comparison of Methods for Evaluating Impacts of Aviation Noise on Communities," Massachusetts Institute of Technology, 2017. Available: http://hdl.handle.net/1721.1/112413.

[3] G. Venturelli Cavalheiro, P. D. Vascik, M. Travnik, S. Salgueiro and J. Hansman, "Characterizing Effective Navigation Performance of Legacy Aircraft to Enable Reduced Procedural Separation in Support of Advanced Air Mobility," in *AIAA Machine Learning and Big Data Applications to Air Transportation I*, 2021. Available: https://doi.org/10.2514/6.2021-2389.

[4] Federal Aviation Administration, "AEDT & Legacy Tools Comparisons," 2016. Available: https://aedt.faa.gov/Documents/Comparison_AEDT_Legacy_Summary.pdf.

[5] L. Jensen and R. J. Hansman, "Data-Driven Flight Procedure Simulation and Noise Analysis in a Large-Scale Air Transportation System," 2018. Available: http://hdl.handle.net/1721.1/119288.

[6] P. Virtanen et al., "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python.," *Nature Methods,* vol. 17, pp. 261-272, 2020. Available: https://doi.org/10.1038/s41592-019-0686-2.

[7] C. R. Harris et al., "Array programming with NumPy," *Nature,* vol. 585, p. 357–362, 2020. Available: https://doi.org/10.1038/s41586-020-2649-2.

[8] Massport, "How Logan Operates," [Online]. Available: https://www.massport.com/logan-airport/about-logan/noise-abatement/how-logan-operates/. [Accessed 10 July 2021].

[9] Massport, Aviation General Management, "Boston Logan International Airport: Monthly Airport Traffic Summary - December 2016," Massachusetts Port Authority, Boston, MA, 2016. Available: https://www.massport.com/media/1737/dec-16-avstats-airport-traffic-summary.pdf.

[10] C. Bullock, "Aviation Effects on Local Business: Mapping Community Impact and Policy Strategies for Noise Remediation," Massachusetts Institute of Technology, 2020.