



Proceeding Paper

A System for Measurement and Analysis of Aircraft Noise Impacts [†]

Donald C. Jackson ^{*}, Thomas C. Rindfleisch  and Juan J. Alonso 

Department of Aeronautics & Astronautics, Stanford University, Stanford, CA 94305, USA;
tcr@stanford.edu (T.C.R.); jjalonso@stanford.edu (J.J.A.)

* Correspondence: dcj@aircraft-noise.org

† Presented at the 9th OpenSky Symposium, Brussels, Belgium, 18–19 November 2021.

Abstract: The Metroplex Overflight Noise Analysis (MONA) project seeks to measure, analyze, and archive the ground noise generated by aircraft overflights and to provide accurate and actionable data for a variety of different purposes. On the one hand, experimental datasets collected and processed by the MONA system can serve as an openly-available database for validation and verification (V&V) of improved noise prediction methods. On the other, study conclusions derived from both the experimental and computational data can serve to inform technical discussions and options involving aircraft noise, aircraft routes, and the potential impacts of the FAA's NextGen procedure changes on overflown communities at varying distances from the airport. Given the complex interdependencies between the noise levels perceived on the ground and the air-traffic patterns that generate the aircraft noise, a secondary goal of the MONA project is to share, through compelling visualizations, key results with broad communities of stakeholders to help generate a common understanding and reach better decisions more quickly. In this paper, we focus on the description of the MONA system architecture, its design, and its current set of capabilities. Subsequent publications will focus on the results we are obtaining through the use of the MONA system.

Keywords: aircraft noise; ADS-B; data collection



Citation: Jackson, D.C.; Rindfleisch, T.C.; Alonso, J.J. A System for Measurement and Analysis of Aircraft Noise Impacts. *Eng. Proc.* **2021**, *13*, 6. <https://doi.org/10.3390/engproc2021013006>

Academic Editor: Junzi Sun

Published: 29 December 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Overview

As a consequence of community complaints resulting from the changes in air-traffic patterns over the San Francisco Bay Area metroplex during the past 5 years, it became increasingly clear to the authors that there is a dearth of high-quality aircraft noise data (from measurements and/or predictions), particularly in areas away from the airport boundary that have not traditionally been the main focus of noise complaints. In addition, through a number of community interactions, we also became aware of the difficulties involved in relating potential flight route changes to noise impacts on the ground. Such lack of actionable data and of methods to effectively communicate with broad-and-often-non-technical communities led us to the development of the aircraft noise system we describe in this paper.

The Metroplex Overflight Noise Analysis (MONA) project set out to achieve the following objectives:

- To measure and analyze ground noise data generated by aircraft overflights in complex metroplex situations;
- To create, curate, and archive experimental datasets that can serve as an openly-available database for validation and verification (V&V) of improved noise prediction methods;
- To fully automate noise analyses based on the Aviation Environmental Design Tool (AEDT) without the need for user intervention; and
- To share, through compelling and interactive visualizations, key analysis results with broad communities of stakeholders.

Our hope is that experimental and computational studies conducted using the MONA system can serve to inform decisions involving aircraft noise, aircraft routes, and the potential impacts of the FAA's NextGen [1] procedure changes on overflowed communities at varying distances from the airport. Moreover, we also hope that the open-source nature of the design, software, and hardware of the MONA system can be easily replicated in many metroplexes around the world with relatively-low cost, to provide a source of high-quality data to inform conversations and future steps. In the near future, we expect to investigate parts of the OpenSky system that could be integrated into MONA.

A secondary goal of the MONA project is to share, through analyses and visualizations, key results with broad communities of stakeholders who generally lack access to this kind of data. In this paper, we focus on the description of the MONA system architecture, its design, and its current set of capabilities. Subsequent publications will focus on the results we are obtaining through the use of the MONA system.

2. Measurement and Collection of Data

In order to quantify and analyze the noise impact of aircraft overflights, we need to know both the trajectories of aircraft flights and the resulting ground noise. To that end, we collect the following types of data:

- Aircraft Positions (via ADS-B) and Speed over Ground;
- Sound Levels;
- Flight and Aircraft Metadata;
- Air Traffic Routes and Procedures;
- Wind and Weather Conditions.

To date, MONA ground stations have been deployed primarily at the houses of interested and motivated members of the public, with costs borne by those (or other) volunteers (including the authors). In the following sections, we describe the key elements of the MONA ground station and measurement and collection system.

2.1. Sensor Controller

The measurement of sound levels and reception of ADS-B transmissions requires a distributed network of sensors mounted outdoors throughout the geographic region of interest, and a means to access/retrieve these data. For the MONA project, we have implemented a series of sensor controllers, incorporating a single-board computer (SBC), a Global Positioning System (GPS) receiver (to provide a highly-accurate time pulse, as well as 3D location), with both network connectivity and power via Power over Ethernet (PoE). These components are integrated within a waterproof/weatherproof enclosure, to support long-term outdoor deployments. The sensor controller runs a Network Time Protocol (NTP) daemon, configured to utilize the pulse per second (PPS) output of the integrated GPS receiver, to provide a Stratum-1 time base, to minimize time differences between our distributed network of sensors. We have developed software to collect each sensor output and transmit/publish them in real-time to a centralized aggregator hosted in a data-center via the Internet. A single sensor controller is capable of simultaneously supporting both ADS-B reception and sound-level monitoring. Because of the long-term field deployment of the sensors, autonomous operation and secure remote access are essential. Remote access is accomplished using reverse-SSH tunnels established by the sensor controller to another server, and whenever possible maintenance of the sensor host is performed via Ansible configuration management scripts, through these SSH tunnels. Figure 1 below shows our typical MONA ground station installation and a view of the sensor controller components inside their weather-proof enclosure.



Figure 1. Sensor controller rooftop deployment with ADS-B and SLM (left), Sensor controller components (right).

2.2. ADS-B Receiver

The primary ADS-B receivers in the MONA network are based on the PiAware/dump1090-fa software by FlightAware [2] using a standard RTL-SDR dongle (inside the sensor controller enclosure in Figure 1 (right)), connected to an ADS-B antenna affixed to the same enclosure. Every second, the JSON output of dump1090-fa is captured by a software daemon, and the ADS-B messages within are minimally processed and then transmitted/published to our centralized aggregator, implemented as an Apache Kafka [3] cluster. The collector daemon also publishes receiver metadata (including GPS location, and sensor controller status) to the same aggregator. At sites that prefer the commercial, fully-integrated Radarcape ADS-B product [4], we developed a variant of the collector daemon to capture and transmit the messages and metadata from this receiver.

2.3. Sound-Level Monitor (SLM)

We use Convergence Instruments (CI) SLMs [5], connected via USB to the sensor controller to measure noise. Another software daemon captures the outputs of the SLM, and transmits/publishes them in real-time to a centralized server, again using Apache Kafka. Recent models of the CI SLM optionally support a USB-Audio feature, providing access to the sampled audio waveform, which we selectively save/transmit in order to capture both noise metrics and audio recordings of aircraft overflights. The SLM collector daemon publishes SLM metadata (including SLM configuration, GPS location, and sensor controller status) to the central server.

2.4. Flight and Aircraft Metadata

Aircraft ADS-B positions alone do not provide a complete description of the flight. Important/valuable missing metadata includes:

- Airport(s) of arrival and departure;
- Assigned runways;
- Air Traffic Control (ATC) assigned routes and procedures;
- Airframe, engine, and ownership.

Arrival and departure airport information can often be obtained via external API access (e.g., [6]), or can be inferred by comparison of the first or last known ADS-B position with airport/runway locations. ATC assigned procedures, routes, and runways can be inferred by comparison of the aircraft's trajectory to the locations (and sequence) of way-points and runways. An area of ongoing development is the integration/incorporation of the FAA System Wide Information Management (SWIM) [7] data feeds, in order to fuse [8] this rich source of metadata with ADS-B aircraft positions. SWIM messages are ingested

into Kafka topics, providing reliable reception of these real-time data feeds. Airframe, engine, and ownership information are obtained by joining the aircraft's ICAO24 unique identifier (included in the ADS-B message) with aircraft registration datasets, including the FAA Aircraft Registry [9] (for US aircraft), and OpenSky's Aircraft Metadata Database [10] (for others).

2.5. Air Traffic Routes and Procedures

The FAA Coded Instrument Flight Procedures (CIFP) [11] is one definitive source for this information that we download, parse, and archive monthly. The CIFP provides airport, runway, and waypoint locations that we use for geo-spatial processing and queries. Flight procedures are converted to a directed-graph representation, and are then processed using both standard and custom graph algorithms.

2.6. Wind and Weather

ADS-B messages usually provide the aircraft's ground speed, but not its airspeed, which is an important factor for the prediction of the resulting noise. We obtain wind speed and direction measurements from the National Oceanic and Atmospheric Administration (NOAA) High-Resolution Rapid Refresh [12] dataset, and in conjunction with the ADS-B provided ground speed and heading information, we estimate the net airspeed [13], which is used to define the specific aircraft trajectory for noise prediction.

3. Data Collection, Archival Storage, Access, and Management

Real-time ADS-B messages, SLM measurements, and SWIM data feeds are received and stored by a distributed-event streaming platform implemented using Apache Kafka. These streams are processed and subsequently ingested into a Postgres [14] database, augmented with both PostGIS [15] (supporting geo-spatial queries) and TimescaleDB [16] (supporting very large time-series tables). ADS-B messages from multiple receivers are de-duplicated, segmented into flights, and relevant metadata is added. Trajectories are included in each flight record/row, encoded as PostGIS 4D LineStrings, which enables arbitrary spatio-temporal queries supporting statistical analyses on vast numbers of flights over arbitrary time periods. In our work, it is a frequent requirement to know the point, distance, and/or time of closest approach (PCA, DCA, TCA) of an aircraft trajectory to a location of interest (LOI) such as the position of an SLM. PostGIS queries can dynamically compute, filter, and return these values for any stored flight trajectory and LOI combination. Visualization of a full day of flights including each aircraft position over the entire San Francisco Bay Area requires low-latency access to thousands of flight records, containing millions of positions, and the ability to rapidly sequence/step/animate through a range of these dates is an important tool to understand flight traffic changes over time. Local access to this data is key for reasonable performance, and an optimized binary representation of a trajectory is included in the database to minimize visualization processing delays. MONA applications and services can use/access the real-time data streams and/or the historic data in the database, as needed.

4. Data Processing and Analysis

The raw data acquired by the MONA system must be processed before it can be used as input to our analyses and to compute statistics of the information collected. This section describes some of the data analyses we have automated in MONA. Once aircraft trajectories and measured noise have been captured, stored, and made available for future use, we process, analyze, quantify, compare, categorize, and summarize the noise impacts.

4.1. Attribution of Sound Levels to Aircraft Overflights

The sound-level measurements obtained both from MONA SLMs and other providers (e.g., [17]) include sampled sounds generated from every source, but only the noise resulting from aircraft overflights is relevant to our research. A number of techniques for

attributing sound peaks to aircraft are described in the literature. Particularly relevant examples include threshold-and-duration, directional and/or arrayed microphones, and spectral identification/categorization. In our experience, an effective methodology is what we refer to as “Determined Aircraft Position/Location for Aircraft Noise Extraction” (DA-PLANE). This algorithm involves computing the time and distance of an aircraft’s closest approach to the SLM location (from ADS-B trajectory data), which gives the estimated time of arrival of the aircraft sound peak maximum at the SLM. Then, we use time-series filtering and analysis to locate peaks above the background in the sound profile that may have been caused by aircraft overflights. These are then time matched with the closest approach data to isolate and identify the sound peaks resulting from aircraft. The net profile amplitudes of isolated peaks above the background are then analyzed to extract desired noise metrics for each identified overflight event. Other implementations of this technique include those of Harding and Ferrier [18] and Giladi [19]. In the MONA system, the $maximum(LAeq_{1s})$ value and SEL metric are extracted and stored in the database, with relations to both the flight (aircraft) and SLM (measurement) location.

4.2. Metric Computation

With aircraft trajectories encoded as geo-spatial datatypes and measured noise metrics attributed to specific aircraft flights and precise locations, we compute standard noise metrics such as Number-Above-Noise-Level, DNL/CNEL, Time-Above-Noise-Level, Background Level, and so forth. Non-noise metrics such as overflight counts per day (within a distance/range) are also computed.

4.3. Geographic Aggregation of Metrics

It is desirable to analyze both aircraft overflight counts and noise metrics over aggregated geographic areas so as to estimate the annoyance, health, and other effects on affected populations. This aggregation processes can take place over either Civic Regions or Grids, which are described below.

4.3.1. Civic Regions

City, County, and US Congressional District boundaries are published in common GIS formats, which we use in conjunction with PostGIS and client-side GIS software libraries to aggregate and report metrics by civic region.

4.3.2. Grids

Aggregation over a regular grid is a powerful technique to represent metrics with uniform granularity over a geographic area. We use the H3 Hexagonal hierarchical geospatial indexing system [20] in a variety of resolutions (sizes) as a standard, regular, geometric grid defined for the entire Earth. H3 hexagon grids are fused with other geographic data sources, for example, we have computed population count estimates for H3 hexagons by apportioning US Census block population data by the block boundary’s percentage containment within the hexagon. The resulting population count of each hexagon enables weighting of metrics aggregated per hexagon by the number of people impacted. (see Figure 2).

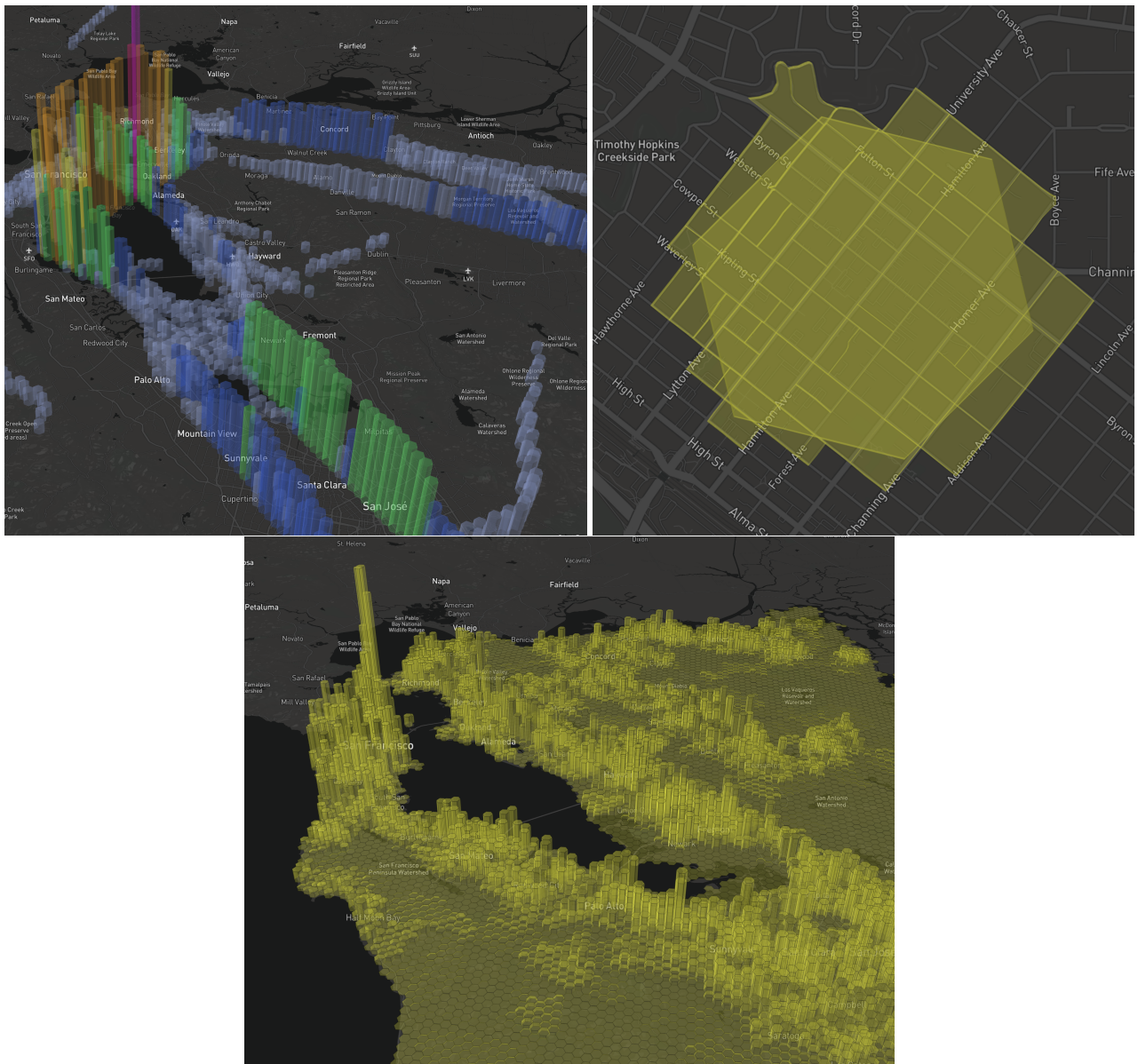


Figure 2. 24 h overhead count by hexagon (top left), apportioning census-block population to hexagon (top right), and population per hexagon of SF Bay Area (bottom).

4.4. Aircraft Noise Prediction

It is infeasible to deploy SLMs in sufficient number and geographic density to obtain measured noise throughout an entire metroplex (cost and logistics being two significant constraints). However, it is feasible to capture all flight traffic via ADS-B by deploying a relatively small number of receivers over the metroplex. Ideally we could use the collected trajectory data to predict the noise generated by each and every aircraft on a fine-grained receptor grid to estimate noise metrics across the entire region of interest.

The FAA’s Aviation Environmental Design Tool (AEDT) [21] is the required software application for assessment of U.S. regulatory actions related to aircraft noise and emissions. Our (aspirational) goal is to run AEDT predictions for every aircraft flight across the SF Bay Area each day, and aggregate the resulting predicted noise metrics to provide quantified noise impacts across the entire metroplex, as a function of location and time.

To these ends, we are working to:

- Automate AEDT study creation, execution, and metric result extraction;
- Accurately model AEDT flight trajectories using ADS-B data; and
- Evaluate and compare AEDT's noise predictions to measured noise levels, in a manner similar to Giladi and Menachi [22].

More detailed descriptions of these individual tasks follow.

4.4.1. AEDT Automation

Current AEDT implementation and workflows are focused primarily on desktop PC use, via its graphical user interface. This usage model does not support the automated processing of thousands of flights per day, over many years. To implement an automation facility, we leveraged AEDT's use of, and reliance on, a Microsoft SQL Server [23] database. Using AEDT's database schema documentation in conjunction with a database table "diff" tool we developed, we gained an understanding of how to create and populate the tables necessary to describe a complete AEDT study. We then developed a software library to facilitate scripted study creation over a network connection to the SQL Server database used by AEDT. AEDT provides a command-line utility, `RunStudy.exe`, to initiate the execution of a specified study that we can invoke over the network. An AEDT study's computed metrics are written into the SQL Server database, so we can also access and extract these results over the network. We created a virtual machine (VM) disk image including AEDT and all supporting packages pre-installed, which can be instantiated and run at any scale, on a commercial cloud provider. We then developed a study-executor application that takes a study-description as input, orchestrates and connects to an AEDT VM, creates an AEDT study using the trajectory of the flight's database ID (provided in the study description) including altitude and speed controls to match the observed trajectory, executes the study, and extracts/stores the metric results in our database. Next, we developed a study-creation application that generates any number of study-descriptions (based on SQL queries specifying any desired database column criteria), and submits each resulting study description onto a job queue. Finally the study executor was enhanced to query the job queue for a study-description to process. As a result, we are able to run any number of AEDT studies in parallel, limited only by the number of AEDT VMs and study executors we create. Both the job and extracted-metrics queues are implemented using the Apache Kafka cluster.

4.4.2. AEDT Trajectory Modeling from ADS-B Data

AEDT is typically used to model flights from a number of specified ground track positions (without altitude). AEDT combines the specified ground track with flight performance models from the EUROCONTROL Aircraft Noise and Performance (ANP) Database [24] and Base of Aircraft Data (BADA) [25] to simulate the aircraft's trajectory for its predictions. This computed, simulated trajectory may differ from the trajectory reported by ADS-B. AEDT provides additional functionality to add altitude and airspeed control codes to the ground track (Section 3.9.1 "Track Control Flights" in [21]), which we use in order to more closely model the reported trajectory. ADS-B trajectory processing steps we use include:

- Smoothing and filtering to remove anomalies that AEDT would reject (e.g., increases in altitude during a descent);
- Line Simplification [26,27];
- Estimation of the aircraft's airspeed, using ADS-B provided ground speed and heading in combination with wind speed and direction data obtained from NOAA.

4.4.3. Comparison of AEDT Noise Prediction to Measured Noise

Our AEDT studies specify the LAMAX and SEL noise metrics, per flight, at each receptor (SLM) location. These metric values are stored in our database, with relations to the flight and location, just as we do with the measured noise peaks attributed to aircraft (described previously). With both predicted and measured noise, comparisons are made,

and which can be filtered, grouped, analyzed, and reported by any available metadata field (e.g., aircraft model) At present, we are actively using this system to predict and compare large numbers (thousands) of flights each day over several years. We anticipate publishing the results of these comparisons, when we have carefully reviewed and validated an amount of data that we deem to be statistically significant.

5. Visualization

Flight traffic information and noise metric contours are visualized using a web-based application that provides 3D geospatial views, based on the `deck.gl` [28] library (see Figure 3): historic flight-track visualizations obtain archived flight trajectories from the database, and real-time flight visualizations obtain streaming trajectories from Kafka. Real-time access to both SLM and ADS-B data has provided unanticipated benefits: we recently developed another web application that displays both SLM metrics overlaid with aircraft times of closest approach (see Figure 4). Viewing this real time visualization while simultaneously hearing passing aircraft is a powerful tool to generate ideas and form hypotheses about how and when overflights generate annoyance.

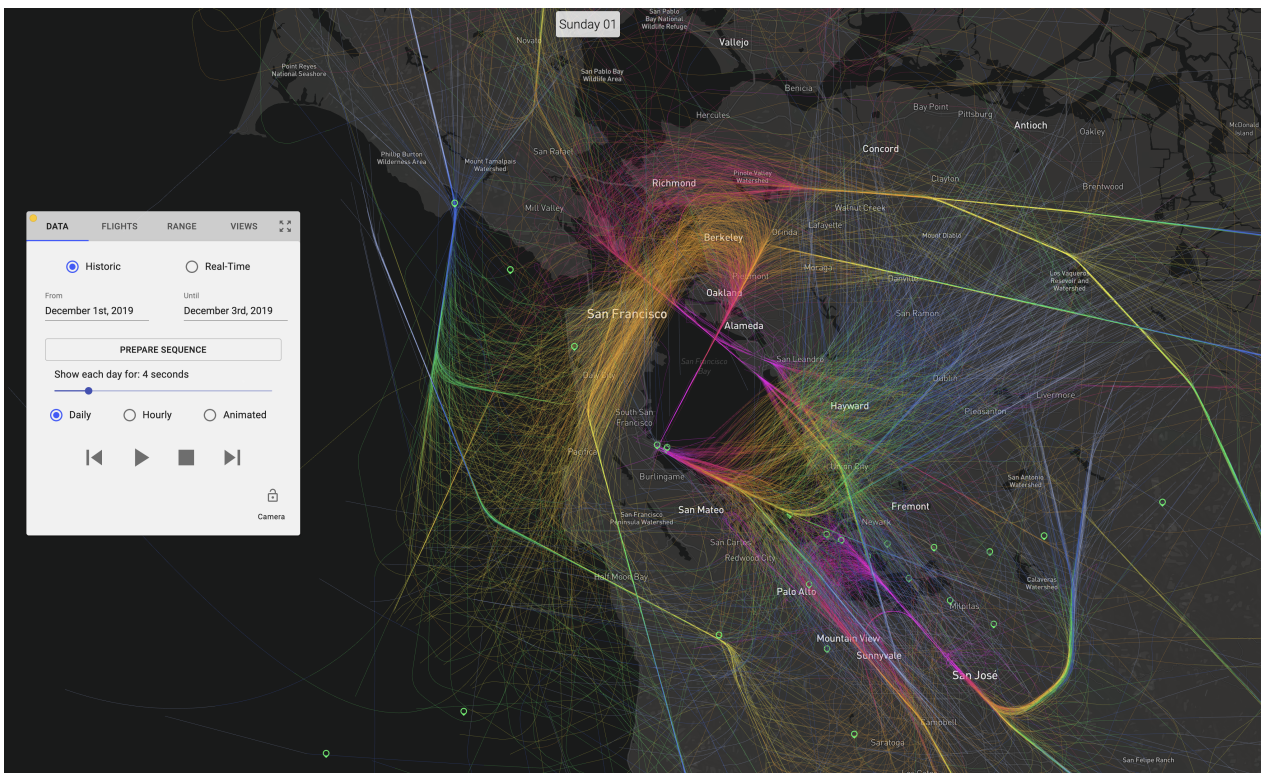


Figure 3. Visual representation of historic traffic patterns over San Francisco Bay Area metroplex.

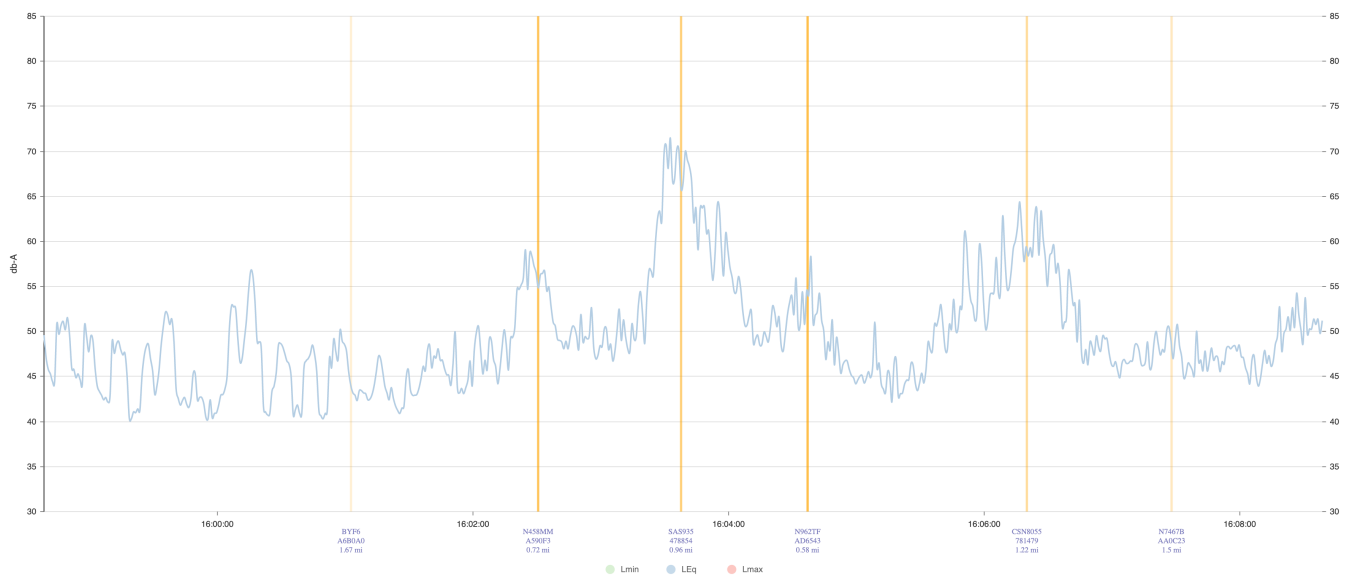


Figure 4. Real-time visualization of noise measurements with aircraft times of closest approach indicated with vertical lines.

6. Comparison to OpenSky Network

The MONA ADS-B collection subsystem provides similar functionality to that of the OpenSky Network, at a much smaller scale, and focused only on our specific needs. Key benefits (not provided by other systems) include:

- Real-time access to ADS-B messages received by all MONA receivers;
- ADS-B messages from all receivers are aggregated, de-duplicated, segmented into “flights”, and stored in a local Postgres database, enabling:
 - High performance access to flight and trajectory data;
 - Trajectories are represented as PostGIS 4D LineStrings, facilitating powerful geo-temporal queries;
 - Inclusion of specialized trajectory representations optimized for high-performance visualization, especially animation.

We are interested in exploring ways to contribute the streaming feed of MONA aggregated ADS-B messages to OpenSky, and to receive streaming OpenSky ADS-B messages from non-MONA receivers (for regions in which we operate) to increase coverage and improve resiliency. For historic data, enabling automated reliable replication of OpenSky data for local database ingestion would facilitate instantiations of the MONA infrastructure in new geographic regions. This is ongoing work that we hope to report on in the future.

7. Example Use of System

Early in 2021, the FAA published the results of its multi-year research effort to quantify the impacts of aircraft noise exposure on communities around commercial service airports in the US, the Neighborhood Environmental Survey [29], and solicited public comment on a number of aspects of the report, including noise metrics. As members of the public, with knowledge, experience, and opinions on the topics and questions posed, we submitted our comments [30], which included analyses enabled by the MONA system, specifically the data and charts illustrating:

- Changes in aircraft traffic resulting from the Covid-19 pandemic;
- Corresponding changes in the number of aircraft noise complaints per day;
- Measured changes in both the DNL and Number-Above-Noise-Level metrics.

This, and other analyses, are enabled by the MONA system. Additional publications will focus on the results of these studies.

8. Conclusions

In this paper we described, at a high level, the infrastructure we have developed in the MONA system and that our work relies on. In short, a modern hardware/software infrastructure has been created that can be deployed anywhere in the world and that can provide real-time and historic assessments of the aircraft noise generated by flight patterns in complex metroplex environments. Together with the automation of AEDT studies, more detailed comparisons between measurements and predictions can be made, the sources of modeling discrepancies can be characterized, and even improved models of noise can be created. Much like the OpenSky Network, our approach relies on a well-proven, low-cost, distributed network of sensors and software for the processing, storage, and use of the information collected. Investigations regarding the suitability of utilizing OpenSky Network information within the MONA system are ongoing.

9. Acknowledgements

- The portion of our work comparing AEDT noise predictions to measured noise is partially funded by the Aviation Sustainability Center (ASCENT) as Project 053 [31]. This ASCENT project is focused on, and limited to, our AEDT comparison effort and does not fund, nor is an endorsement of, broader MONA efforts including the deployment of the MONA ground-station network or other MONA analyses, recommendations and proposals;
- Sean Doyle of the FAA's Office of Environment and Energy serves as the ASCENT-053 program manager, and provided valuable advice and assistance with our usage of AEDT;
- AEDT developers at the Volpe National Transportation Systems Center provided important assistance and support;
- Bruno Paillard, co-founder and chief engineer of Convergence Instruments has provided valuable help and advice with our sound monitoring efforts;
- Baruch Berger is the primary developer of the flight-track visualization web application.

In addition, a number of research assistants and Stanford University students have contributed and continue to contribute to the development of the MONA system, including:

- Nick Bowman led the implementation of our automated AEDT infrastructure, worked on AEDT/ADS-B trajectory modelling, and brought up our initial Kafka cluster;
- Vikas Munukutla developed the initial AEDT database study creation software;
- Aditeya Shukla and Priscilla Lui developed the SLM collector daemon;
- Aditeya Shukla developed the real-time SLM visualization application;
- Brian Munguía integrated NOAA weather data and worked on AEDT/ADS-B trajectory modelling.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. FAA. Next Generation Air Transportation System (NextGen). Available online: <https://www.faa.gov/nextgen/> (accessed on 7 November 2021).
2. FlightAware. PiAware. Available online: <http://flightaware.com/adsb/piaware/> (accessed on 7 November 2021).
3. Apache Kafka. Available online: <https://kafka.apache.org/> (accessed on 7 November 2021).
4. Jetvision. Radarcape. Available online: https://shop.jetvision.de/radarcape_en (accessed on 7 November 2021).
5. Convergence-Instruments. Sound Level Meters. Available online: <https://dev.convergenceinstruments.com/pc/sound-level-meter-data-loggers/> (accessed on 7 November 2021).
6. FlightAware. AeroAPI. Available online: <http://flightaware.com/commercial/aeroapi/> (accessed on 7 November 2021).

7. FAA. System Wide Information Management (SWIM). Available online: https://www.faa.gov/air_traffic/technology/swim/ (accessed on 7 November 2021).
8. Wikipedia. Data Fusion. Available online: https://en.wikipedia.org/w/index.php?title=Data_fusion&oldid=1048017628 (accessed on 7 November 2021).
9. FAA. Aircraft Registry. Available online: https://www.faa.gov/licenses_certificates/aircraft_certification/aircraft_registry/ (accessed on 7 November 2021).
10. OpenSky. Aircraft Metadata Database. Available online: <https://opensky-network.org/datasets/metadata/> (accessed on 7 November 2021).
11. FAA. Coded Instrument Flight Procedures (CIFP). Available online: https://www.faa.gov/air_traffic/flight_info/aeronav/digital_products/cifp/ (accessed on 7 November 2021).
12. Benjamin, S.G.; Weygandt, S.S.; Brown, J.M.; Hu, M.; Alexander, C.R.; Smirnova, T.G.; Olson, J.B.; James, E.P.; Dowell, D.C.; Grell, G.A.; et al. A North American Hourly Assimilation and Model Forecast Cycle: The Rapid Refresh. *Mon. Weather. Rev.* **2016**, *144*, 1669–1694. [[CrossRef](#)]
13. Wynnyk, C. Wind Analysis in Aviation Applications. In Proceedings of the 2012 IEEE/AIAA 31st Digital Avionics Systems Conference (DASC), Williamsburg, VA, USA, 14–18 October 2012; p. 5C2-1. [[CrossRef](#)]
14. Postgres. Available online: <https://www.postgresql.org/> (accessed on 7 November 2021).
15. PostGIS. Available online: <https://postgis.net/> (accessed on 7 November 2021).
16. TimescaleDB. Available online: <https://www.timescale.com/> (accessed on 7 November 2021).
17. Sunnyvale NoiseLab. Available online: <https://syv.noiselab.casper.aero/> (accessed on 7 November 2021).
18. Harding, M.; Ferrier, D. Using Post Analysis of a Noise Sample Stream in place of Noise Monitor Based Thresholds in the Detection of Aircraft Noise. In *INTER-NOISE and NOISE-CON 2014 Congress and Conference Proceedings, Melbourne, Australia*; Institute of Noise Control Engineering: Reston, VA, USA, 2014; Volume 249, p. 5386.
19. Giladi, R. Real-time Identification of Aircraft Sound Events. *Transp. Res. Part Transp. Environ.* **2020**, *87*, 102527. [[CrossRef](#)] [[PubMed](#)]
20. H3: Hexagonal Hierarchical Geospatial Indexing System. Available online: <https://h3geo.org> (accessed on 7 November 2021).
21. Lee, C.; Thrasher, T.; Boeker, E.; Downs, R.; Gorshkov, S.; Hansen, A.; Hwang, S.; Koopmann, J.; Malwitz, A.; Noel, G.; et al. *Aviation Environmental Design Tool (AEDT) Technical Manual, Version 3d*; Report Number: DOT-VNTSC-FAA-21-06; Federal Aviation Administration: Washington, DC, USA, March 2021.
22. Giladi, R.; Menachi, E. Validating Aircraft Noise Models. *Proceedings* **2020**, *59*, 12. [[CrossRef](#)]
23. Microsoft. SQL Server. Available online: <https://www.microsoft.com/en-us/sql-server/sql-server-2019> (accessed on 7 November 2021).
24. EUROCONTROL. Aircraft Noise and Performance (ANP) Database. Available online: <https://www.aircraftnoisemodel.org/> (accessed on 7 November 2021).
25. EUROCONTROL. Base of Aircraft Data (BADA). Available online: <https://www.eurocontrol.int/model/bada> (accessed on 7 November 2021).
26. Douglas, D.H.; Peucker, T.K. Algorithms for the Reduction of the Number of Points Required to Represent a Line or its Caricature. *Cartogr. Int. J. Geogr. Inf. Geovis.* **1973**, *10*, 112–122. [[CrossRef](#)]
27. Ramer, U. An Iterative Procedure for the Polygonal Approximation of Plane Curves. *Comput. Graph. Image Process.* **1972**, *1*, 244–256. [[CrossRef](#)]
28. deck.gl: WebGL-powered Visualization Framework for Large-Scale Datasets. Available online: <https://deck.gl> (accessed on 7 November 2021).
29. FAA. Neighborhood Environmental Survey. Available online: https://www.faa.gov/regulations_policies/policy_guidance/noise/survey/ (accessed on 7 November 2021).
30. MONA-Project. Submitted Comments on FAA NES. Available online: https://downloads.regulations.gov/FAA-2021-0037-3754/attachment_1.pdf (accessed on 7 November 2021).
31. ASCENT-053. Validation of Low-Exposure Noise Modeling by Open-Source Data Management and Visualization Systems Integrated with AEDT. Available online: <https://ascent.aero/project/validation-of-low-exposure-noise-modeling-by-open-source-data-management-and-visualization-systems-integrated-with-aedt/> (accessed on 7 November 2021).

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle		5. Report Date	
		6. Performing Organization Code	
7. Author(s)		8. Performing Organization Report No.	
9. Performing Organization Name and Address		10. Work Unit No. (TRAIS)	
		11. Contract or Grant No.	
12. Sponsoring Agency Name and Address		13. Type of Report and Period Covered	
		14. Sponsoring Agency Code	
15. Supplementary Notes			
16. Abstract			
17. Key Words		18. Distribution Statement	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages	22. Price