Project 053 – Validation of Low Exposure Noise Modeling by Open-Source Data Management and Visualization Systems Integrated with AEDT

Stanford University

Project Lead Investigator

Juan J. Alonso
Vance D. and Arlene C. Coffman Professor
Department of Aeronautics & Astronautics
Stanford University
Stanford, CA 94305
Phone: (650) 723-9954
E-mail: jjalonso@stanford.edu

University Participants

Stanford University

- PI: Prof. Juan J. Alonso
- FAA Award Number: 13-C-AJFE-SU-022
- Period of Performance: February 5, 2020 to February 4, 2021
- Tasks:
    1. Aviation Environmental Design Tool (AEDT) integration with Metroplex Overflight Noise Analysis (MONA) software
    2. Validation and verification of AEDT noise predictions in 55–65 db Day-Night Average Sound Level (DNL) areas.
    3. Software architecture infrastructure suggestions for AEDT.

Project Funding Level

Year 1 of ASCENT Project 53 has been allocated FAA funds in the amount of $169,903. Cost sharing in excess of this amount has been identified from various sources. Mr. Thomas Rindfleisch and Mr. Donald Jackson are contributing all of their time, uncompensated, to the project. In addition, contractor costs for the development of the MONA project website, the cost of undergraduate student support and summer interns, and some equipment purchases (and installation costs) are also being used to generate cost share for this project. During the first nine months of this project, a total of more than $640,000 of cost share has already been accounted for.

Investigation Team

The investigation team is made up of the faculty, graduate and undergraduate students, and collaborators listed below with their respective areas of expertise / areas of contribution:

1. Juan J. Alonso (PI, Stanford Aeronautics & Astronautics): overall responsibility for the project and its technical and administrative elements.
2. Nick Bowman (Graduate Student, Stanford Computer Science): MONA project cloud infrastructure, cloud-based execution of AEDT analyses, Apache Kafka-based data collection.
3. Brynne Hurst (Graduate Student, Stanford Computer Science): flight trajectory database analysis and synthesis.
4. Donald Jackson (Collaborator, independent consultant): overall MONA project infrastructure (servers, databases, hardware / software monitoring), GIS, web-based visualization deployment, technical guidance.
5. Priscilla Lui (Co-term Student, Stanford Computer Science): real-time Sound-Level Monitoring (SLM) software, metrics, Raspberry Pi connectivity.

6. Vikas Munukutla (Graduate Student, Stanford Computer Science): automation of AEDT analyses via generation/query of input/output databases on the cloud.
7. Chetanya Rastogi (Graduate Student, Stanford Computer Science): overall database infrastructure improvements, noise monitoring / filtering software.
8. Thomas Rindfleisch (Collaborator, Stanford University Emeritus): noise monitoring and filtering, aircraft trajectory collection / processing, visualization.
9. Aditeya Shukla (Undergraduate Student, Stanford Aeronautics & Astronautics): artificial intelligence/machine learning (AI/ML) classification of aircraft trajectories, real-time Sound-Level Monitoring (SLM) software
10. Kadin Hendricks (Undergraduate Student, Aeronautics & Astronautics): AEDT input/output database structure.

Project Overview

The MONA project (Metroplex Overflight Noise Analysis) was started to provide real-time and objective data, analyses, and reports to key stakeholders and policy makers to mitigate the noise impacts of the deployment of new NextGen procedures. This system (a) collects and archives air traffic data using a network of antennae and receivers, (b) analyzes noise impacts using a variety of metrics, (c) visualizes resulting large-scale datasets, and (d) uses a network of sound-level monitors to enhance the quality of noise predictions. The focus of this ASCENT project is to improve upon the noise predictions of MONA through tighter integration with the FAA's Aviation Environmental Design Tool (AEDT). In particular, our work is focused on the following three tasks: (1) integrate and automate AEDT's noise analysis capabilities, (2) validate and verify (V&V) AEDT's noise predictions in 55–65 db Day-Night Average Sound Level (DNL) areas, and (3) propose software engineering/architectural choices for future AEDT development to enhance usability in multiple workflows including API formulation, visualization interfaces, resilient data acquisition and storage, and cloud computing.

The expected benefits of this project mirror the tasks mentioned above, including (a) ability to automate complex noise analyses in metroplexes so they are available in near-real time after the preceding 24-hr period, (b) a better understanding of the accuracy of AEDT's current noise models in low noise (55–65 db DNL) areas and the reasons for the discrepancies (if any) in existing predictions, and (c) guidance to software developers on flexible architectures and APIs for AEDT so that the tool is more versatile and generally applicable. AEDT predictions are built around the policy context of an average annual day. All the V&V results produced and shared by the MONA team will be focused on a cumulative daily basis for which flight track data is directly collected.

Background and Previous Accomplishments

The MONA project (Metroplex Overflight Noise Analysis) started approximately 2.5 years ago with the main objective of providing real-time and objective data, analyses, and reports to key stakeholders and policy makers to help in mitigating the noise impacts of the deployment of new NextGen procedures. Since then, we have put together a preliminary open-source system that (a) collects, archives, and makes available air traffic data using a series of networked antennae and receivers 24/7, (b) analyzes noise impacts using a variety of metrics (based on both a MONA-developed noise prediction tool and, albeit manually, the noise predictions tools within AEDT), (c) visualizes resulting large-scale datasets in a simple, user-friendly fashion using both a bespoke website and Uber's kepler.gl and deck.gl large-scale data visualization toolboxes, and (d) is beginning to use a small network of low-cost sound-level monitors scattered across the Bay Area to enhance noise predictions so they describe exactly the actual noise levels experienced.

The longer-term objectives of the MONA project are to (a) ensure the validation and verification of all noise predictions provided (by AEDT or other tools) in both areas near the airport and in other areas further away from the airport, typically 55–65 db DNL, (b) achieve full automation of complex noise analyses in metroplexes in the United States, including AEDT-based noise predictions, (c) make all results web-accessible for in-depth interpretations of historical and proposed changes, (d) eventually study potential alternative traffic patterns in complex metroplexes to mitigate aviation environmental impacts, and (e) export the proven/validated MONA technology to other metroplexes via open-source software/hardware.

When ASCENT Project 53 started (Feb 2020), the MONA software had achieved several significant objectives which positioned the team to achieve the work described in the project proposal. MONA has:

1. Deployed a network of ADS-B / MLAT antennae and had completed initial versions of the software necessary to merge the data streams from all of these antennae including de-duplication of sightings, identification of aircraft

equipment and routes flown, physical interpolation of data missing from the joint observations, and archiving (in appropriate json-based formats) of the information collected for successive analysis.

2. Begun the process of integration with FAA's AEDT (v2d, at the time) software that enables manual prediction of noise footprints at arbitrary receptor locations. Initial efforts to incorporate NASA's Aircraft Noise Prediction Program (ANOPP2) tools predictions had also been pursued and put aside until full AEDT integration is completed.
3. Begun to incorporate measurements from networked sound-level monitors into the system and to develop preliminary versions of non-aircraft-noise filtering techniques (of the raw noise data) based on digital filtering, aircraft position information, and automated identification of background noise levels. Significant further efforts to improve the filtering techniques by using measurements and correlations from multiple sound-level monitors are also planned.
4. Over the previous year, interfaced the above-described MONA preliminary software modules with the kepler.gl open-source visualization framework, developed by Uber, to be able to visualize and animate aircraft positions and paths, noise predictions, various routes and procedures, etc., to better communicate the results of our work (see Figure 1). A preliminary version of the MONA website, which helps visualize and makes accessible the same information but for a less experienced user, had also been started (see Figure 2 for an image from an early prototype).

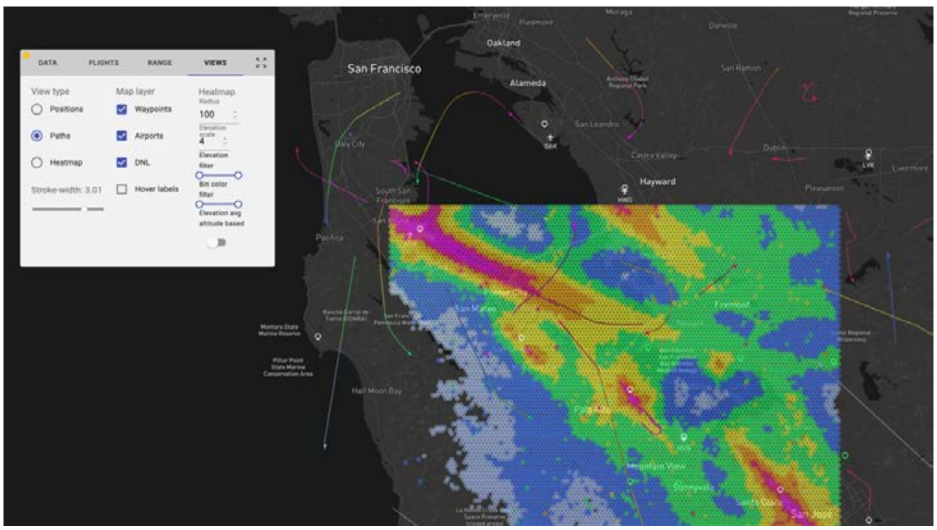

Figure 1. MONA visualization (using kepler.gl and deck.gl) of traffic patterns in the Bay Area.

Figure 2. Current MONA web-access prototype for real-time aircraft location and 24-hr DNL contours.

Task 1 – AEDT Integration with MONA software

Stanford University

Given that the vision of MONA is to make 24/7 aircraft noise information available through a simple visualization interface, predictions of aircraft noise are a fundamental component of the entire effort. We eventually intend to validate these noise predictions with data collected from strategically located SLMs whose raw sound data has been appropriately filtered to eliminate non-aircraft noise sources. This Task, however, focuses on the necessary integration steps to make AEDT noise analyses available through the MONA interface. This Task will be complete when noise analyses in an entire metroplex (the Bay Area in our initial test case) for an entire 24-hr period can be performed, visualized, analyzed, and archived without user intervention. The automation of the entire workflow requires the implementation and automation of a number of key steps:

1. Starting from a set of MONA-acquired and pre-processed flight paths and associated aircraft equipment for the previous 24-hr period, the necessary AEDT inputs are generated in an SQL study database that can be later consumed by AEDT. The input database must contain the actual location of the aircraft as a function of time, the specific aircraft equipment, and other auxiliary parameters needed for the analysis.
2. The setup of a noise analysis in AEDT, including all the necessary metric descriptions, receptor locations, annualizations, and additional input data must be automatically generated and included in the input study database as permitted by AEDT v3.
3. Automated execution of arbitrary analyses can then be pursued so that AEDT can be run through a batch process without user intervention. For this batch process, we have developed a cloud-based solution that automatically fires up a cloud instance, sets up the necessary communication structures, runs the AEDT study, and returns the study results to the computer executing the study.
4. A module for the extraction of the computed metrics and their spatial distributions, for arbitrary metric computations, is being developed. The interaction with the output of AEDT analyses is intended to happen directly through the SQL output study database that contains and stores all the necessary information.

During the period covered by this report, we have completed a preliminary version of all four subtasks (some of which are described in more detail below) and are continuing the process of both automating every step, scaling our capabilities to handle larger and larger studies, and using automated AEDT analyses to support Task 2. In the process, we have created our own cloud-based AEDT execution environment (which we have named `raedt`, for *remote* AEDT) that works on Google Cloud Platform instances of arbitrary size (number of processors, memory, etc.)

Subtask Name: Remote and Automatable AEDT Execution in the Cloud

Objective
The main problem being addressed in this subtask is that AEDT can only run on the Windows operating system, which often requires the acquisition of specialized, standalone hardware in order to be able to run AEDT studies. In addition, studies can only be constructed via a graphical user interface (GUI), which is not easily automatable or executable in a scriptable, programmatic manner, making it difficult to efficiently run a large breadth and variety of surveys. The objective in this subtask is to move AEDT execution to the cloud by taking advantage of Windows Instances running on the Google Cloud Platform that are pre-configured with AEDT and all supporting software. Our goal was to create a system in which these instances can be remotely accessed from any platform/operating system in a programmatic manner. This would allow members of the team to write scripts to programmatically create and run AEDT studies in a scalable manner (discussed in a later subtask). To accomplish this goal, we proposed and developed a system that is based on templated creation of Windows-based virtual machines in the cloud and is controlled by a custom command-line software package uniquely tailored to carry out the necessary tasks and administrative actions necessary to run remote AEDT studies.

Research Approach
There were three main components of our approach to achieve the goals set forward in this Task. First, we had to select a suitable cloud platform on which to build our cloud AEDT infrastructure. Of the three main cloud providers (Google Cloud Platform (GCP), Amazon Web Services, and Microsoft Azure), we found that GCP offered the best combination of cost effectiveness, well-documented command-line and language-specific APIs, and performance flexibility. We note, however, that the same capability can be replicated in any of the three major cloud providers with the necessary customization changes. Next, we had to manually construct a disk image (based on Windows Server) that would act as the template from which all AEDT-capable virtual machines (VMs) would be instantiated from. At the moment, this process involves manual installation of the necessary software packages on a Windows Server 2017 machine (including SQL Server), but we envision that this process can be automated in future versions of the project. Finally, we focused on developing a command-line software package implemented in a mix of Python and shell scripts that leveraged Google's GCP APIs to give users the ability to carry out important remote tasks in a programmatic manner. The outcome has been the possibility to create AEDT instances, open two-way communication channels between the computer / program running the analysis and the cloud instance executing it, run the AEDT analysis, and retrieve the results of the study. This software is currently being deployed and tested and will be available as open-source software to anyone who wants to benefit from its automation capabilities.

Milestones
- Set up GCP infrastructure, with Windows Server "base install image" containing all necessary software, standardizing the environment for all team members.
- Defined library of command line tools (named **`raedt`**) to enable creation, control, and destruction of AEDT VMs in an easily scriptable format.
- Enabled seamless SSH connection, remote port tunneling for SQL Server access, connection via RDP for necessary GUI-based actions, and remote SSH command execution to support automated AEDT tools.

Overall, we have achieved cost-effective execution of AEDT studies (on the order of $0.05–$0.10 per hour for minimum required hardware specifications) and have set ourselves up in Project 53 for the scalable execution of much larger studies, repeatedly as needed, while allowing ourselves to tailor the cloud instance hardware resources that are needed for each individual study. This capability has been integrated with the capabilities described in the next subtask to automate our AEDT analyses, as we had suggested in our original proposal.

Subtask Name: Automatable AEDT Study Creation/Execution and Results Extraction

Objective(s)
One of the main goals of MONA / Ascent 53 (see Task 2) is to compare MONA SLM sensor output against AEDT predictions to see how similar the two measurements are and to gain a more nuanced understanding of the accuracy of AEDT predictions in 55–65 dB DNL areas. However, to run AEDT studies, one has to manually enter every flight using the AEDT GUI, and given that there are many thousands of flights per day in the San Francisco International Airport (SFO) region, this task is difficult to do through the GUI. This task would be both feasible and routine if, instead, the studies are automatically generated and executed by a script that can access the flights for the day. Moreover, if this script can run completely on the cloud using

the infrastructure described in the “Remote and Automatable AEDT Execution in the Cloud” subtask, then anyone on any laptop or remote server can run AEDT studies programmatically on their machine regardless of whether their machine is a Windows machine and independently of the resources needed (computer memory, number of processors, etc.) to carry out the AEDT analysis in reasonable time. The purpose of this subtask is to reverse-engineer the logic of AEDT study creation, execution, and results extraction as it would happen in the AEDT GUI to skip the overhead that comes with using the GUI for large, multi-flight studies, using a Windows VM and our internal command line tool `raedt` for cost-effective execution, ease of access, and flexibility.

Research Approach

There are three main steps to this subtask. We first run an AEDT "create-study" script that creates a copy of the empty study database and populates the copy according to the various flights, airports, and desired noise metrics in the study. This step is the largest and most significant aspect of the subtask and involves researching the smallest details of every necessary value needed to be inserted into flight-related tables, airport-related tables, and metric-related tables. It also involves investigating how to replicate the simplest possible study—the empty study—in order to maintain all views, stored procedures, etc. for the new study. Next, the “remotely execute the study” step on the VM, which populates the results-related tables in the study’s database, is completed. This step simply issues a command called RunStudy.exe over SSH to the VM so that the VM can run the study on the AEDT databases that were just populated by the create-study script. Lastly, we run an “extract results” script which extracts the noise metric results produced after executing the study and stores the result in JSON files on the user’s machine. This step involves setting the right permissions in AEDT configuration files so that XML extraction is possible, running a SQL query to extract the XML results from the EVENT_RESULTS table of the study, and then converting the XML to JSON using Python libraries.

Milestones

- Connected to the flights `postgres` databases and AEDT SQL Server databases.
- Created a backup of the empty study database and used the diff between an ASIF-generated study and empty study to fill in the appropriate flight, airport, and metric tables for a one-flight, multi-receptor study in which the flight is landing at SFO.
- Extracted results after running the one-flight study, converted the XML result to JSON, and saved the JSON file.
- Used Windows VM cloud execution to create an automation flow for AEDT studies which consists of starting the VM, starting SSH tunnel to VM, running create study script, running RunStudy.exe remotely on the VM using SSH, and running results extraction script.

Task 2 – Validation and Verification of AEDT Noise Predictions in 55-65 db DNL Areas

Stanford University

The noise prediction modules inside of AEDT, based on noise power distance (NPD) relationships and certification data, were developed and calibrated mainly for areas of objectionable noise close to the airports (> 65 db DNL), at a constant velocity (160 knots), and for a particular aircraft high-lift system configuration. Even including efforts such as those in ASCENT Project 43 (which re-evaluated the noise power distance curves using ANOPP analyses and the ability to change the aircraft configuration during arrival/departure procedures), there is some evidence that the accuracy of AEDT’s predictions in areas of lower noise (between 55–65 db DNL) is lacking and must be improved. For these reasons, in this Task we proposed to undertake a thorough evaluation of the accuracy of AEDT’s predictions when measured against microphone readings from a small network of SLMs that the MONA project has acquired, tested, and sited. In order to accomplish the objectives of this Task, we will pursue the following steps:

1. Data acquisition and archiving for noise measurements at 3–4 locations. We will begin with the acquisition of the raw noise data (Leq, Lmin, Lmax) at a small number of locations. Even though we have already been collecting this data at one particular location, as part of this effort we will re-evaluate the most useful locations for the existing SLMs, place the SLMs at the desired locations, and begin collecting and archiving data for both the validation study and future use. Data will be collected 24/7, with particular attention paid to events that fall under the noise-level category that is the focus of this Task.
2. As a pre-processing step to the V&V portion of this work, we will focus our activities on the continued development of a series of non-aircraft noise removal algorithms that combine filtering techniques, automatic identification of

multiple aircraft peaks, automatic detection of background and peak noise levels, and real-time information regarding the position, velocity, and heading of the aircraft so we can maintain high levels of accuracy. In order to further improve the quality of our aircraft-only noise data, we also intend to use the noise measured at multiple, non-collocated but nearby SLMs and use the correlations between these measurements and aircraft locations to filter out false positives.

3. A validation campaign will then be pursued covering variations of the most important variables that can be observed in the 55–65 db DNL areas of the Bay Area metroplex: aircraft mass/weight, aircraft type, aircraft altitude at the time of the noise event, aircraft slant distance, and aircraft speed. No attempt to correct predictions for atmospheric conditions will be made, although we expect to pursue validation efforts over a number of different days and conditions that will span the range of the typically observed weather conditions.
4. The results of the validation study will be carefully documented in an archival publication. If possible, discrepancies between the measured noise levels and the AEDT predictions will be attributed to the main source causing the discrepancy.

Finally, all the data used for validation purposes will be processed at both the aggregate level and the level of individual flight predictions so that data-driven methods for the improvement of the NPD curves used in AEDT (or those produced in ASCENT Project 43) can be pursued in the future. For example, if all of the recorded overflights of a particular aircraft type, which have variability in mass, atmospheric conditions, high-lift system configuration, etc. have a corresponding time history of the recorded sound pressure level at an observer location, a learning algorithm could be devised to correct the AEDT predictions as functions of altitude, airspeed, and distance to the observer. Such a data-driven methodology can lead to significantly improved predictions in AEDT.

Subtask Name: Validation & Verification of AEDT predictions in 55–65 db DNL areas

Objectives

The main objective of this Task is to evaluate the accuracy of AEDT noise predictions by directly comparing them with the metrics obtained from processing the raw data collected by the network of MONA SLMs. Another important aspect of this Task is to develop a well-documented library that abstracts the noise filtering and noise metric calculation aspects and can be open-sourced to assist future developments in this area.

Research Approach

The raw SLM data at multiple locations is currently captured and stored in a centralized database with associated timestamps which can be retrieved by running respective SQL queries. These data come from calibrated networked Convergence Instruments equipment that we have installed at various locations around the Bay Area and that have been tested with co-located sound measurement equipment loaned by SFO and found to agree with that equipment to within 0.1 dB. Such tests that we conducted lend credibility to the quality of the sound data recorded from our SLMs. The retrieved data is then processed through the noise filtering pipeline which was developed and implemented to removes any non-aircraft noise by following a multi-step process that identifying a noise matches the sound peaks with a corresponding flight and provides a summary statistic about the noise levels. The main steps of this process include:

- Retrieval of raw noise recording (including non-aircraft noise sources) from the MONA database.
- Calculation of a background noise profile to establish the (time-varying) identified background noise level.
- Development of a threshold profile, above the background level, below which all noise events are filtered.
- Signal processing of remaining signal to identify peak-like structures that might correspond to an aircraft event.
- Matching each potential *peaklet* to times and points of closest approach from ADS-B database. *Peaklets* whose time is within a specified time of a Time of Closest Approach (TCA) are retained and associated with a particular flight event. Peaks that do not have a corresponding potential aircraft are also filtered.

The result is a filtered signal that is guaranteed to correspond to a conservative estimate of the actual aircraft noise as the possibility exists that some aircraft noise events (such as some sensitive flights, etc) are recorded but not identified in our flight database collected simultaneously. Figure 3 below is an example of the interactive plots produced with this now fully automated process. This process can also be run during any subset of the day (a subset of the recorded data) to prepare actual recordings of flight events at multiple locations for comparisons with AEDT predictions. For example, flight recordings during the early hours of the day (between 1–4 AM) tend to have very low levels of background noise and, therefore, are prime candidates for the comparisons with AEDT predictions. We have now been collecting sound recordings at multiple locations for about a year and are beginning to prepare a database of test cases that will serve as the basis for our evaluation.

A Python client has also been developed to interact with the DB and provides APIs to process and perform required operations, including post-processing and plotting. For developing the Python client, the best software engineering practices were employed and provide configurable API access.

Milestones

- Defined and documented APIs for communicating with the centralized database by abstracting the SQL queries.
- Verified the pipeline for obtaining TCAs and measuring associated peaks using the SLM data from the database.
- Restructured original noise-filtering code and modularized it for future reusability.
- Provided an end-to-end integration for running the noise filtering pipeline directly on the database data.
- Begun to create a database of flights that will be used for the comparisons with the automated AEDT analyses described in an earlier subtask.

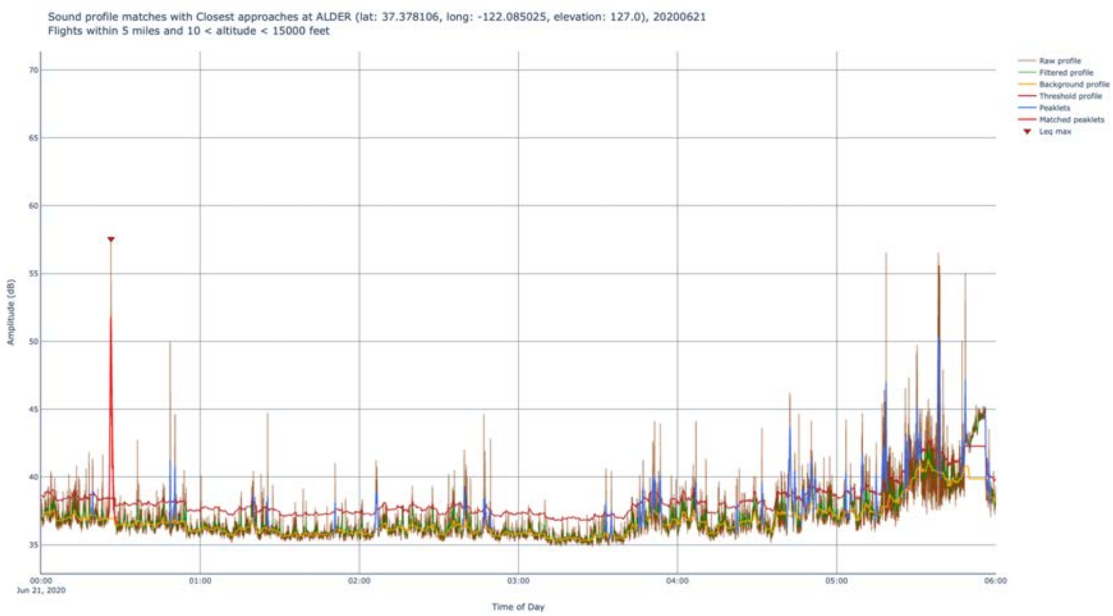

Figure 3. Graphical depiction of multi-step process used to filter non-aircraft data from SLM raw recordings. Example from 6-hour interval on June 21, 2020.

Subtask Name: Real-time SLM Sound Recording and Metric Calculation

Objective

There are interesting real-time applications (versus post-processing data every 24 hours or every other reasonable time period) we would like to achieve with our noise data. However, before completion of this Task, those applications were impossible because ingesting the noise data involved waiting 6–24 hours for a file to be published to the cloud by the manufacturers of the SLM. Another downside was that this file contained already-computed metrics that were computed onboard the SLM, so we could not access the raw data which might prove useful for future analysis. Thus, this research task aimed to bypass the cloud file by using a Raspberry Pi to directly process the audio recorded by the SLM in real-time and send it to our database. The final vision is to integrate this SLM code and the ADSB-collector code onto a Raspberry Pi, to create an all-in-one MONA unit.

Research Approach

In terms of learning to compute the metrics ourselves, we consulted Dr. Bruno Paillard, the senior designer at the company that sold us the SLMs. With his guidance, we were able to replicate results for Leq, Lmax, and Lmin to near-perfect accuracy.

As for the goal of real-time, we leveraged past experiences implementing real-time applications on microprocessors to architect the final design of the software running on the Raspberry Pi.

Milestones

- Designed and implemented an automated Python library that records audio from aSLM, computes metrics, and sends it to a server / database.
- Successfully wrote an algorithm to compute Lmin, Lmax and LEq on streaming audio and verified it with 99.99% accuracy.
- Added a resilient way to store computed metrics in case of a network failure and implemented exponential back off for sending attempts.
- Made the script configurable (custom log intervals, custom frame rates, audio formats, etc.).
- Used multiprocessing and multithreading to leverage the Raspberry Pi's quadcore and improve throughput.
- Supported raw collection of audio samples (saving chunks of audio for later use).

Task 3 – Software Architecture Infrastructure Suggestions for Future AEDT Development

Stanford University

The MONA project is working on a software infrastructure to setup hardware, collect data from various types of sensors, archive/store those data without interruptions in a fail-safe manner, and process the data (and store the results in appropriate databases) for future use that will be distributed in an open-source manner. Such use of multiple functional modules has resulted in the need to carefully think about the overall software architecture of the project, the standards used for the representation and communication of the data, the kinds of databases that can be used, the existing visualization standards that must be interfaced with, and the potential for cloud computing infrastructure to shorten the time to deployment of the MONA software in different metroplexes beyond the Bay Area.

Research Approach

The FAA Office of Environment and Energy (AEE) has expressed interest in leveraging the ASCENT 53 experience (in software development-related aspects) and the expertise of professional software developers from Silicon Valley in the MONA team to provide suggestions / guidance to be used in deriving requirements for future AEDT development.

As part of this Task, our team is creating an architecture document that shows the organization of data, storage, processing, cloud-based analysis and automation, etc., so that future versions of AEDT can be more easily integrated into complex analysis workflows that we cannot do easily today. The integrity (and redundancy / resilience) of the data acquisition process is also of paramount importance to the success of environmental data collection and analysis efforts, and ASCENT 53 is using the latest tried-and-true distributed data collection approaches (which we will describe in a future report) and that have implications for interaction with AEDT. Our team has been pioneering visual representation approaches for the display of both the raw data and the processed data that could be more easily integrated with AEDT. All of these experiences from the ASCENT 53 team efforts, together with feedback from briefings (to Silicon Valley software developers and Stanford computer science faculty) that will provide additional suggestions regarding the typical workflows expected in AEDT v5, will be archived and provided as the output of this Task. Finally, we will create a report that will contain recommendations regarding (a) expected requirements for AEDT v5, (b) best software development practices / interfaces that might be followed, (c) organization of the overall code base, (d) APIs for external execution of AEDT's modules, and (e) suggestions for database and datafile formats and visualization interfaces. This Task is necessarily the last one for the first year of ASCENT 53 as it relies on the various experiences in Tasks 1 and 2 that are informing the outcome.

The output of these efforts will be communicated periodically with ASCENT's program managers, who will be invited to participate in the brainstorming sessions and in our internal deliberations.

Milestones

- Completed architecture of MONA system (database data storage, multi-server / multi-computer communication, cloud instances of AEDT, two-way communication with cloud instances, software module architecture, postGIS

system, U.S. Census data incorporation, Apache Kafka / Zookeeper server for resilient/redundant data acquisition, and visualization infrastructure).
- Identified major elements that interact with AEDT to better understand potential changes to the AEDT structure and the availability of APIs that would be helpful.

Major Accomplishments

- Created a completely new infrastructure for ASCENT 53 / MONA that can scale to the types of data collection and analysis expected of a complex metroplex such as that of the Bay Area.
- Developed a working version of the non-aircraft noise filtering process to compare sound recordings with AEDT predictions.
- Created a cloud-based automated approach to run AEDT studies so that they can be automated in the Bay Area metroplex.
- Collected necessary data to generate a database of individual flight data that will be used in our AEDT V&V study.
- Rewrote large portions of the software infrastructure to (a) make it more readily available via open-source, and (b) to provide input to Task 3, suggestions for AEDT improvements for better integration into MONA-like automated analyses and comparisons.

Publications

None thus far. An initial publication detailing the infrastructure and early results of the MONA system is planned for later this year.

Outreach Efforts

As part of our efforts to produce data of high quality that can be used for decision making, we have engaged with the Palo Alto City Council, with various citizen groups in the Bay Area, and like-minded individuals that may lend their help to the development of the ASCENT 53 / MONA infrastructure. Although we have been asked to participate in technical discussions in the existing local noise roundtables, these interactions have yet to take place and will be limited to discussions of technical elements of relevance to the community.

Awards

None

Student Involvement

A number of undergraduate and graduate students are / have been part of our team during this past nine months since the project started. Their names and areas of responsibility are listed at the beginning of this document. None of the students have yet graduated, although three of them will at the end (or before the end) of this academic year.

Plans for Next Period

We intend to complete all three Tasks in our Statement of Work as planned. In addition to the completion of all milestones, the release of appropriate parts of the ASCENT 53 / MONA project and the demonstration of various capabilities through participation in aircraft noise related meetings / conferences is also envisioned.