Sonification of Motion of Unmanned Aerial Vehicles

by

Israa M. Ali

**A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Engineering
(Robotics Engineering)
in the University of Michigan-Dearborn
2025**

Master's Thesis Committee:

**Associate Professor Alireza Mohammadi, Chair
Professor Ella Atkins (Virginia Tech.)
Dr. Michael Putty**

Israa M. Ali

aliim@umich.edu

ORCID iD: 0009-0001-6177-2595

© Israa M. Ali 2025

Acknowledgements

The completion of this dissertation is possible due to the support of my advisor, Dr. Alireza Mohammadi, thank you for your continued guidance. I am very grateful you were with me on this research journey from the beginning. A special thank you to Dr. Ella Atkins, who played a critical role in my decision to farther pursue engineering into Robotics and being on the committee. Along with Dr. Michael Putty, thank you all for your time and support.

I would also like to acknowledge my family and friends for their endless support and encouragement throughout my academic career. Thank you for all your love and bearing behind the scenes. Lastly, thank you to my employer, General Motors, for contributing to this degree as well as my fellow team for their support throughout the process.

Table of Contents

List of Tables

List of Figures

List of Acronyms

CSV comma-separated values

HDI Human Drone Interaction

HRI Human Robot Interaction

IRB Institutional Review Board

MIDI Musical Instrument Digital Interface

NASA National Aeronautics and Space Administration

NED North-East-Down

UAV Unmanned Aerial Vehicle

Abstract

The development of Unmanned Aerial Vehicles (UAVs) can be enhanced through the use of sonification, an emerging field within Human-Robot Interaction (HRI). This dissertation introduces UAVSonification, a computational algorithm that maps simulation data to musical notes using the Musical Instrument Digital Interface (MIDI). By integrating UAVSonification with Formation Flight Simulation in Simulink, this study explores the sonification of UAV trajectories under environmental conditions. The function transforms simulation data into auditory signals, allowing users to discern key dynamics through sound. Specifically, the data series for multiple UAVs is mapped to piano notes via MIDI on MATLAB, providing auditory insights into UAV trajectories, environmental conditions, and control errors. A well-controlled flight path and stable heading controller produce harmonious sounds, while disruptions and deviations result in dissonance. UAVSonification offers a unique auditory approach to understanding UAV behavior in relation to control dynamics and environmental conditions. The sonification of UAVs has the potential to aid in the planning and analysis of UAV trajectories and controllers, as well as in creative endeavors. The effectiveness of the proposed method is shown through MATLAB numerical simulations.

Chapter 1
Introduction

Sound has a unique ability to transport listeners to different times and places, a tune can evoke vivid memories or emotional responses, like the sound of cicadas conjuring up the stillness of a summer evening. Hearing, one of the most fundamental human senses, is uniquely capable of influencing mood, memory, and perception in ways that the other senses may not [1]. Despite its potential to evoke powerful reactions and communicate complex information intuitively, sound remains underutilized in engineering applications, often overshadowed by visual or tactile interfaces.

Recent advancements have significantly propelled the field of sonification, the art and science of converting data into sound, into a new era characterized by broadened applications and increased appreciation. As a potent analytical tool, sonification is particularly valuable for interpreting complex datasets, including those with temporal, spatial, or multidimensional characteristics that are difficult to visualize or comprehend quickly [2]. By translating these data attributes into audible forms, sonification enables listeners to discern patterns and relationships through an auditory medium. This method not only enhances the accessibility of data but also unveils insights that might otherwise remain obscured. Moreover, sonification's ability to integrate with other data representation techniques enriches the overall interpretive process, opening novel avenues for both academic research and practical application in various fields.

Institutions such as NASA have already embraced sonification to interpret and share insights into phenomena like black hole behavior [3]. Their technique included assigning musical notes to various data elements such as brightness and position of celestial objects. The black holes sonification spammed across multi-wavelength images with each wavelength mapped to a different range of audible tones, radio waves to the lowest tones, optical data to medium tones, and X-rays to the highest tones. By transforming data collected from astronomical observations into sound, scientists can gain a fresh perspective and potentially reveal patterns or anomalies in ways that visual representations alone may not. The resulting auditory experiences also engage the public, making complex scientific concepts more accessible and compelling.

In robotics, sonification has also gained traction, particularly within the realm of human-robot interaction (HRI). By adding auditory layers to robotic feedback, engineers can enhance communication and emotional resonance between robots and humans. Sonification in HRI can offer intuitive cues for users creating more immersive interactions that facilitate faster response times, reduce cognitive load, and allow non-experts to interpret robotic actions with ease. Beyond HRI, sonification also plays a role in industrial robotics [4]. Sonification aids in the efficiency of robotic systems and deepens the emotional engagement and accessibility of these systems, bridging the gap between technical data and human intuition [5]. Below in Figure 1 is a summary of the categorization of sound in HRI recreated from Designing Sound for Social Robots: Candidate Design Principles [6].

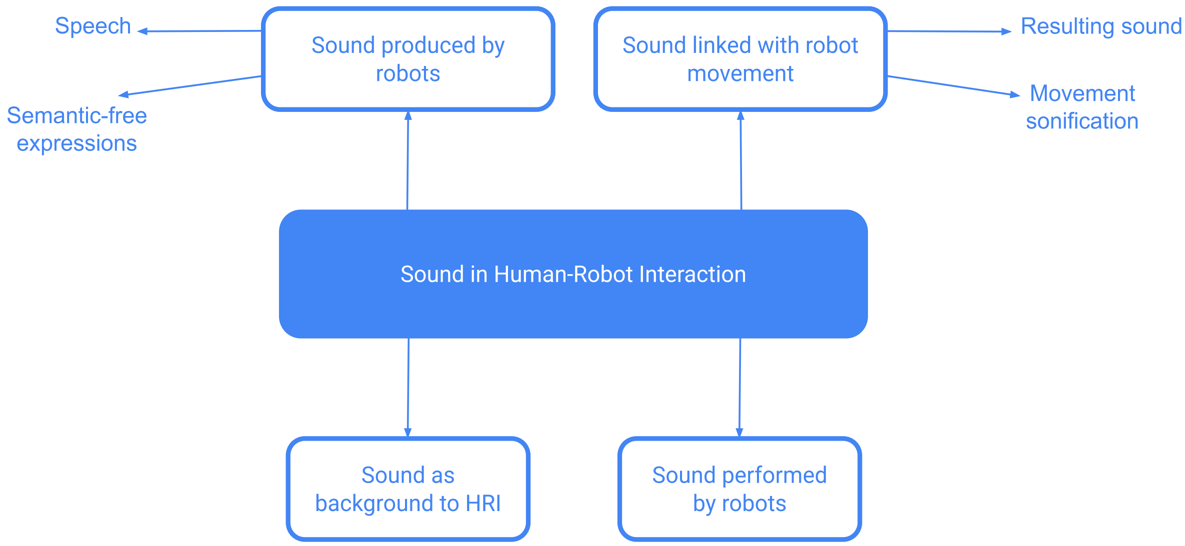

Figure 1: A high-level categorization of sound in HRI

As sonification techniques continue to advance, the possibilities for sound to revolutionize data interpretation, enhance interactivity, and foster a more intuitive connection with technology seem boundless. In fields ranging from environmental monitoring to medicine, sonification is gradually being recognized for its unique capacity to simplify complex information, make invisible processes perceptible, and transform the way we understand and interact with the world through sound [7].

As unmanned aerial vehicles (UAVs) have transitioned from primarily military applications to a broad spectrum of civilian and commercial uses, including search and rescue, infrastructure inspection, and delivery services, the demand for enhanced human-drone interaction (HDI) has grown substantially [8]. Effective HDI is crucial for ensuring that operators and observers can manage, interpret, and respond to UAV behavior accurately and efficiently, especially as these devices operate in increasingly complex and dynamic environments. Traditional methods of

UAV monitoring, which often rely heavily on visual data, can be limited in situations where screens and displays become crowded, visibility is restricted, or real-time visual feedback is delayed. In this context, sonification offers a compelling solution to augment HDI by providing an additional auditory layer of information.

Sonification enables real-time auditory feedback that can reveal critical information about UAV trajectory, stability, and environmental influences, such as wind and turbulence, by translating data patterns into sound. This auditory layer has the potential to enhance the understanding of UAV behavior by conveying nuanced changes, deviations, and events that might be easily overlooked in visual data streams. For instance, subtle shifts in a UAV's trajectory due to wind gusts can be challenging to detect visually in real time but may become immediately apparent through changes in pitch, tempo, or harmony in a sonified output. Sonification is particularly valuable in scenarios where users may have limited visual access to UAVs, such as during nighttime operations, search and rescue missions in dense terrain, or urban environments with multiple UAVs in flight. Figure 2 below depicts a UAV flying under wind conditions with its resulting data. Visual data can be complex for multiple data series, while audio could be simpler to decipher.

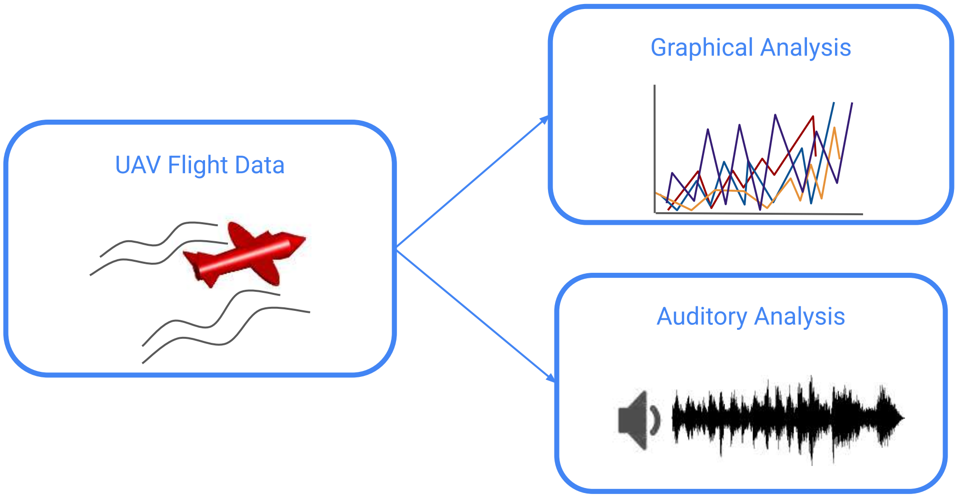

Figure 2: Depiction of data results of a UAV in Wind

This research introduces UAVSonification, a novel MATLAB-based function specifically designed to map UAV simulation data to musical notes using the Musical Instrument Digital Interface (MIDI) protocol. UAVSonification is integrated with the Formation Flight Simulation module in Simulink, providing a robust platform for sonifying UAV trajectories under a wide range of simulated conditions. These conditions include environmental variables such as wind and turbulence, which critically affect UAV performance.

The core functionality of UAVSonification lies in its ability to convert essential data points, such as UAV position, wind conditions, and controller errors, into a comprehensive array of auditory cues. This conversion facilitates an intuitive understanding of complex dynamics that are often challenging to decipher through visual data alone. By employing auditory signals to represent variations in flight dynamics, UAVSonification offers an innovative feedback mechanism that significantly enhances the development and fine-tuning of UAV trajectories and control systems.

Moreover, this auditory feedback channel allows developers and researchers to quickly and effectively identify patterns and anomalies in UAV behavior, thereby accelerating the iterative process of controller design. The integration of sound not only simplifies the interpretation of intricate flight data but also enriches the simulation experience, making it more accessible and engaging for users. As a result, UAVSonification stands out as a valuable tool in the ongoing development of more sophisticated and reliable UAV systems, promoting enhanced situational awareness and control precision in UAV operations.

The primary objective of UAVSonification is to transform complex flight data into sound in a way that highlights distinctions between smooth, stable trajectories and erratic, potentially problematic ones. When the UAV maintains a stable flight path, the sonified output might reflect a steady, harmonious melody. In contrast, abrupt deviations, or corrections due to environmental disturbances or controller error could manifest as dissonant notes, changes in tempo, or other variations in the sound profile, alerting users to potentially significant changes. This auditory feedback mechanism can allow users to assess UAV performance, respond to anomalies, and maintain better situational awareness without relying solely on visual monitoring tools.

By offering an innovative approach to UAV data interpretation, UAVSonification opens new avenues for HDI and UAV performance analysis, leveraging the human brain's ability to quickly interpret changes in sound.

Chapter 2
Related Work

Research in UAV-specific sonification, particularly focused on error tracking and trajectory monitoring, remains limited. While the field of HRI has seen some exploration of sonification applications for robots, little has been applied directly to UAVs. This gap represents a significant opportunity for advancement in UAV sonification, where auditory cues could enhance users' real-time understanding of UAV behavior, trajectory deviations, and system errors. To achieve effective sonification in this context, a thoughtful approach is essential, one that aligns the sonification strategy with the specific data characteristics and the intended insights to be conveyed.

The literature review begins by examining foundational theories, tools, and techniques that form the bedrock of sonification systems, especially those applicable to UAV trajectory and error sonification. Walker and Nees' *Theory of Sonification* is particularly relevant here, as it emphasizes the importance of mapping data dimensions, such as time, position, or velocity, to acoustic dimensions, such as pitch, volume, and rhythm, in ways that match the goals and perceptual needs of the end-user [9]. Their work suggests that sonification is not simply about auditory feedback for status updates or alarms but can offer a holistic auditory representation of complex data, enabling a more nuanced exploration of temporal patterns and spatial changes. This research underpins the dissertation's aim to develop a sonification approach that allows for real-time, holistic monitoring of UAV dynamics, leveraging auditory cues to provide an intuitive grasp of UAV trajectory and error patterns. Sonification is beneficial for analyzing this type of data since the human auditory system is good at detecting temporal changes and patterns. There are additional advantages over visual data, it creates an opportunity for visually impaired people to access scientific data [10].

Building on this theoretical foundation, Latupeirissa and Bresin explore the role of sonification in enhancing HRI by giving robots a "communicative ability" through sound. In their work with PepperOSC, a system integrating audio tools like Pure Data, MaxMSP, and SuperCollider, they demonstrated that real-time auditory feedback could make robots' movements more intuitively understandable, thus fostering more engaging interactions with humans [11]. The study revealed a preference for natural and ambient sounds, which may feel more intuitive and less intrusive to users than artificial or alarm-based sounds. This insight will guide the selection of sound types in the UAV sonification system, aiming to choose notes and soundscapes that enhance the interpretability of trajectory and error data without causing user fatigue.

In addition, Zahray and Savery's research on sonifying robot gestures to improve awareness of robot states and the enjoyment of interaction presents valuable techniques for mapping sound to motion [12]. By sonifying different movements and gestures of the robot Shimon through variations in pitch, timbre, and synthetic motor sounds, they examined how auditory cues could convey meaningful information about movement direction and intensity. This approach to auditory mapping enabled users to intuitively associate specific sounds with the robot's movements, although unpleasant sounds were found to detract from the user experience. Their findings inform this dissertation's approach to UAV sonification by reinforcing the importance of clear, pleasant, and contextually relevant sounds to effectively communicate UAV trajectory changes and error states. In another study, Donmez and Cummings explored sonification for UAV supervisory control. Their work, *Auditory Decision Aiding in Supervisory Control of Multiple Unmanned Aerial Vehicles*, evaluated how sonification could assist military personnel in monitoring UAVs by providing auditory cues for trajectory deviations. Results showed a 19% improvement in reaction time when sonification was applied, highlighting the effectiveness of audio cues in fast-paced environments [13]. However, the abundance of audio cues became overwhelming when supervising multiple UAVs simultaneously, underscoring the need for a more streamlined approach to sonification. Simplified and familiar sounds can improve listener comprehension and reduce cognitive load. This principle is integrated into this dissertation to ensure that UAV trajectory and error sonification remain clear, non-intrusive, and intuitively understood by operators [14].

Together, these studies provide a robust foundation for the design of a UAV sonification system that is functional and user centric. By integrating principles from both theoretical and applied research, the dissertation will aim to create a sonification framework that transforms multilayered UAV data into intuitive auditory cues, enhancing users' development and understanding of UAV performance. Table 1 below summarizes the literature review conducted.

Table 1: Summary of Literature Review

Publication	Summary	Takeaway
Theory of Sonification [9]	Provides a robust theoretical framework	How to align sonification with data exploration
The Sound of Science [10]	Benefits and challenges of data sonification as an alternative to traditional data visualization	There is an opportunity for UAV development to more inclusive

PepperOSC: enabling interactive sonification of a robot's expressive movement [11]	Tool for robot movement sonification, enhancing HRI through creative sound applications	Natural and ambient sounds are more intuitive for humans
Robot Gesture Sonification to Enhance Awareness of Robot Status and Enjoyment of Interaction [12]	Improving HRI sonification by conveying the robot's status and intentions without relying on visual cues	The significance of using clear, pleasant, and contextually appropriate sounds to effectively convey UAV trajectory changes and error states
Auditory Decision Aiding in Supervisory Control of Multiple Unmanned Aerial Vehicles [13]	The study explores how auditory aids improve decision-making and situational awareness in managing multiple UAVs	The effectiveness of audio cues in fast-paced environments
Toward Improving User Experience and Shared Task Performance with Mobile Robots through Parameterized Nonverbal State Sonification [14]	Introduces a parameterized sound model for enhancing HRI by improving user understanding, engagement, and task performance	Using simple and familiar sounds enhances understanding and minimizes cognitive load

Chapter 3
Background

UAV Simulation

To sonify UAVs' motion, a prototype on Simulink is used. The MathWorks fixed-wing UAV block provides a high-fidelity model for simulating fixed-wing UAVs [15]. It includes nonlinear equations of motion, customizable aerodynamics, propulsion, and environmental effects, enabling realistic analysis of flight dynamics and control system design. The equations of motion for the fixed-wing UAV block are nonlinear and represent the 6-degrees-of-freedom (6-DOF) dynamics of a rigid body in flight [16]:

$$X = m\,[\dot{u} + qw - rv]$$
$$Y = m\,[\dot{v} + ru - pw]$$
$$Z = m\,[\dot{w} + pv - qu]$$

Here X, Y, and Z are the externally applied forces. The inertial velocity components are u, v, and w. The mass of the rigid body is m. The angular rates are p, q, and r.

$$L = I_{xx}\dot{p} + \left(I_{zz} - I_{yy}\right)rq + I_{xz}(\dot{r} + pz)$$
$$M = I_{zz}\dot{q} + (I_{xx} - I_{zz})rp + I_{xz}(r^2 - p^2)$$
$$N = I_{zz}\dot{r} + \left(I_{yy} - I_{xx}\right)qp + I_{zx}(\dot{p} - qr)$$

Here L, M, and N are the inertial moment acting on the UAV. I is the inertia tensor. The UAV block is used to simulate multiple UAVs along with the Guidance Model block [17]. The Guidance Model block provides a high-level framework for simulating UAV trajectory tracking. It includes pre-configured guidance logic for position, velocity, and altitude control, allowing the integration of waypoints and flight paths into autonomous navigation systems. This model simplifies the design and testing of guidance algorithms by enabling customizable inputs for UAV dynamics and environmental conditions. The inputs and outputs for a fixed-wing UAV is summarized below in Table 2.

Table 2: Inputs and outputs for the Guidance Model block

Inputs	**Outputs**
Height – Altitude [m]	North, East, Height – position [m]
Airspeed – speed [$\frac{m}{s}$]	Airspeed – speed [$\frac{m}{s}$]
RollAngle – Roll angle [rads]	HeadingAngle – angle between ground velocity and north [rads]
Environment – wind and gravity [$\frac{m}{s^2}$]	FlightPathAngle – angle between ground velocity and north-east [rads]
	RollAngle – roll angle [rads]
	RollAngleRate – angular velocity [rads]

In this block, there is a UAV controller configuration tab. Here, the proportional (P) and derivative (D) gains can be tuned for the fixed-winged UAV to control height, flight path angle, roll, air speed, and min/max flight path angle. The model outputs UAV states and uses the built-in derivative function to compute time derived of the state. It is assumed that there is zero sideslip and autopilot controls the airspeed, altitude, and roll angle. These existing MathWorks blocks and functions on Simulink are useful for prototyping UAVs on a trajectory for this dissertation to collect data on the motion throughout.

Musical Instrument Digital Interface

The Musical Instrument Digital Interface (MIDI) is a universally recognized technical standard that facilitates communication and synchronization between electronic musical instruments, computers, and various other devices [18]. Developed in the early 1980s, MIDI was a transformative innovation in the music industry, allowing seamless interoperability among equipment that was previously restricted by proprietary protocols. Unlike recorded audio, which encapsulates the sound of music, MIDI transmits symbolic information about musical elements, such as notes, timing, pitch, velocity, and control changes. This abstraction enables the receiving device to interpret and generate music using its own sound library or synthesis engine, offering unparalleled flexibility in music production and performance. Figure 3 below shows a simple flow of how MIDI is created.

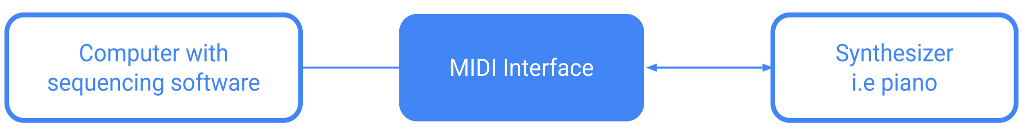

Figure 3: MIDI Flow Simplified

The MIDI protocol's architecture is inherently compact and efficient, supporting up to sixteen independent channels over a single connection. Each channel can control a distinct instrument or sound, making it possible for a single controller to manage a complex arrangement of devices.

This versatility has cemented MIDI's role as a cornerstone of modern music technology, facilitating applications ranging from live performances and studio recording to algorithmic composition and interactive installations. Beyond its traditional musical applications, MIDI has expanded into other fields, including gaming, interactive art, and robotics, where precise timing and control are essential.

In MATLAB, MIDI functionality is enabled through the Audio Toolbox, providing a robust framework for creating, sending, and receiving MIDI messages [19]. This integration supports real-time interaction with MIDI devices and control surfaces, making MATLAB a powerful tool for audio signal processing, algorithm development, and experimental music research. For instance, users can send MIDI messages to external synthesizers, receive input from MIDI controllers, or programmatically manipulate MIDI data to generate complex musical structures.

Key features of MATLAB's MIDI capabilities include interfacing with connected MIDI devices through functions such as mididevice and midimsg, enabling both input and output operations. MATLAB also supports MIDI control surfaces, allowing for dynamic parameter adjustments during audio processing workflows. Advanced users can implement real-time MIDI callbacks and synchronization to create interactive and responsive systems. Furthermore, MATLAB's ability to interpret and generate MIDI messages opens possibilities for innovative applications, such as sonifying scientific data, automating composition processes, and designing custom synthesizers.

By leveraging the MIDI protocol within MATLAB, users can bridge the gap between computational audio research and practical music production. This integration exemplifies the adaptability of MIDI as a medium for both creative and technical exploration, underlining its enduring significance in digital music technology.

The use of MIDI in robotics is seen in creative ways. In Suzuki and Hashimoto's *Robotic interface for embodied interaction via dance and musical performance,* MIDI is used to control the sound generation and synchronization of the robotic system with the human performers. Through MIDI signals, the robot communicates with external musical devices and software, allowing it to produce sounds, trigger musical events, and adapt its movements in real-time to the ongoing performance. This integration of MIDI ensures seamless coordination between the robot's reactive actions and the musical elements, creating a cohesive and dynamic interaction during performances [20]. Besides musical robots, there is little MIDI integration with development projects, such as the one in this dissertation.

Chapter 4
Methodology

UAV Simulation

This dissertation leverages the Formation Flight Simulation in Simulink, developed by MathWorks, as a foundational model for simulating coordinated flight missions involving multiple UAVs following designated waypoints [21]. A simplified flow of the simulation is shown below in Figure 4.

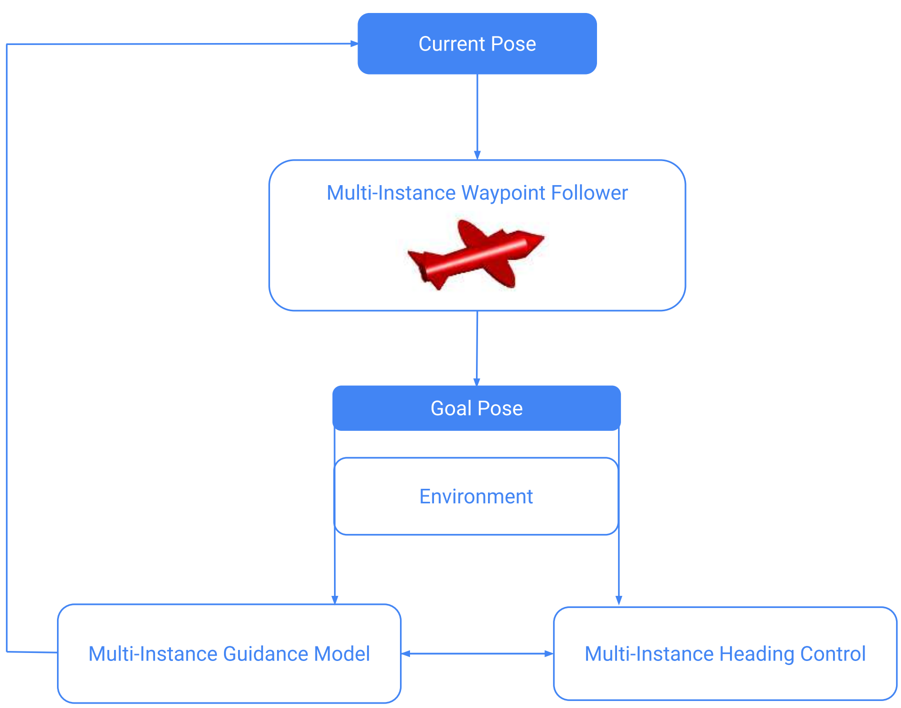

Figure 4: Simplified Flow of Formation Flight Simulation on Simulink

In this model, waypoints are specified in a north-east-down (NED) reference frame in meters, allowing for precise spatial navigation. The UAVs are fixed-winged models. The core components of the formationFlightSimulation.slx model include: a multi-instance waypoint follower for trajectory tracking, a multi-instance heading controller for directional control, a multi-instance guidance model for path management, and a UAV animation block for real-time visualization. Figure 5 and Figure 6 show the multi-instance guidance model and the environmental block, respectively.

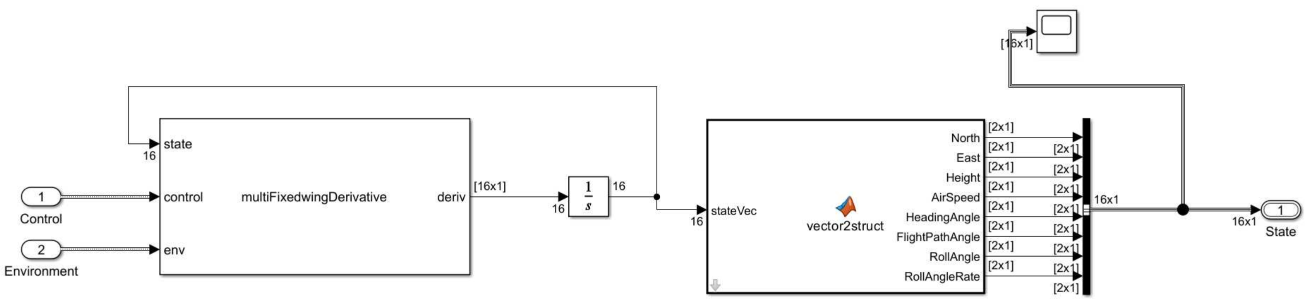

Figure 5: Multi-Instance Guidance Model

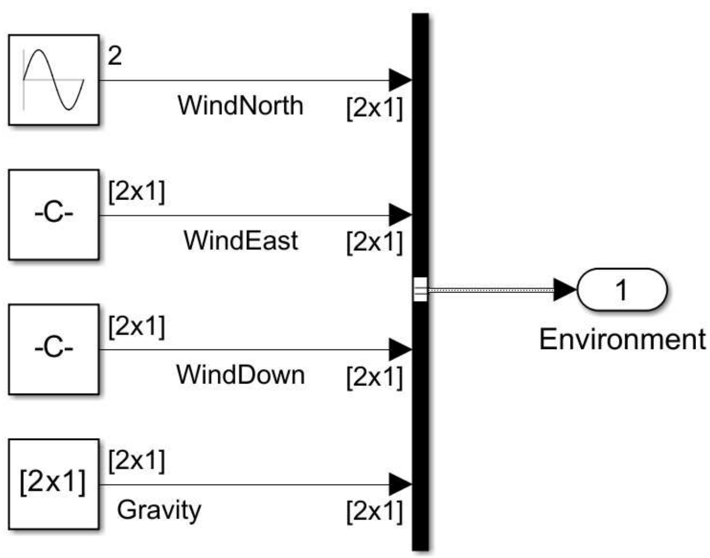

Figure 6: Environmental Block

For this dissertation, the simulation is tailored to study two UAVs. Key modifications were made to the model’s blocks to suit this scope, optimizing the simulation for detailed analysis of paired UAV behavior. The environment block was customized to incorporate wind oscillations, providing a more dynamic and realistic simulation environment that closely mirrors the challenges UAVs might face in real-world conditions. Additionally, a heading controller error calculation was integrated into the Simulink model to facilitate real-time monitoring of directional deviations, further enhancing the accuracy and depth of the simulation for research purposes. These enhancements enable a comprehensive study of UAV trajectory stability and heading controller performance in varying environmental conditions.

The heading angle control gain is adjusted to explore the effects of different controller strengths on UAV trajectory and stability. Two scenarios are defined to illustrate this contrast: in Scenario A, the heading angle control gain is set to 0.97, reflecting a stronger control response, while in Scenario B, the gain is set to 0.30, representing a weaker control influence. After running each simulation for 30 seconds, the output data is saved to a CSV file for post-simulation analysis and sonification.

This data includes various simulation metrics, but the focus of this dissertation is on three key variables: UAV pose, wind effects, and heading controller error. These data points provide insights into the UAVs' movement dynamics, stability under different control strengths, and the system’s responsiveness to external disturbances. The modified simulation model also includes visual aids that display the UAV trajectories and heading controller error, offering an immediate visual context to complement the data. Figure 7 and Figure 8 below are the resulting UAVs animation and heading controller error, respectively, that can be visualized throughout the simulation run.

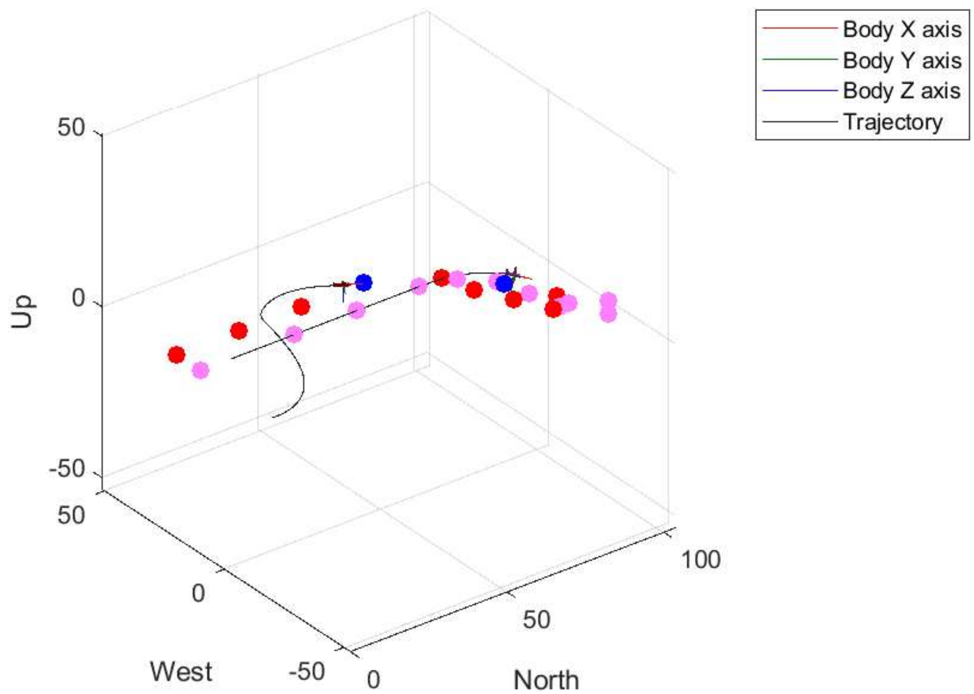

Figure 7: UAV Trajectory from UAV Animation Block

In Figure 7 above, it is a snippet of the two UAVs midflight. The red and pink dots are the waypoints for each UAV that the waypoint follower is going towards. The blue indicates the next waypoint the UAV is going to reach.

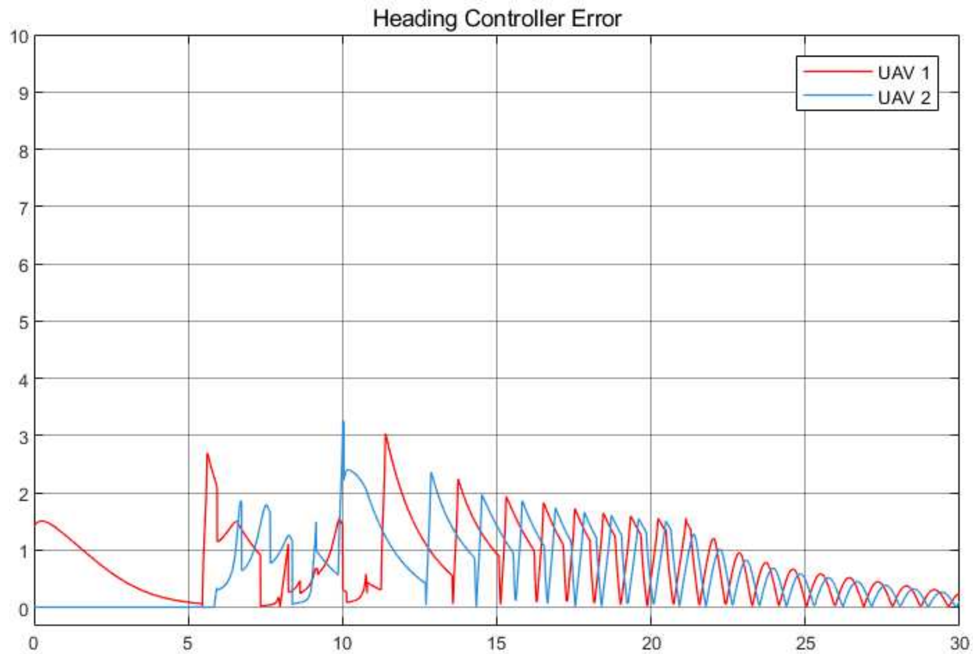

Figure 8: Heading Controller Error from Simulation

This setup allows for a comprehensive analysis of control strength effects on UAV performance, which can be further enhanced through sonification to reveal subtle changes and patterns in the UAVs' behavior.

The CSV file generated from the simulation contains data on each UAV's pose, wind conditions, and heading controller error. The pose data reflects the UAV's position in meters along the x, y, and z axes, as well as the heading angle χ in radians, which represents the current course direction. This position information translates to the north, east, altitude, and heading angle in the simulation's north-east-down (NED) reference frame. Wind data is derived from the environmental block, which incorporates oscillating wind conditions to simulate realistic, non-ideal flight paths influenced by environmental factors. Heading controller error is captured as the angular difference between each UAV’s current heading angle and its desired heading, providing insight into the UAV’s directional accuracy and stability.

Sonification

The CSV file collected from the simulation, serves as the input for the UAVSonification process, a MATLAB function designed to map UAV simulation data to sound using MIDI format. UAVSonification translates the pose, wind, and heading controller error data into musical notes, using existing MATLAB-compatible MIDI functions [22]. The core of this sonification process is a MIDI matrix, structured as an Nx8 matrix where each row represents a distinct note event. The matrix columns include the track number, channel number, note number, velocity, start time in seconds, end time in seconds, and placeholders for MIDI-specific messages (note_on and note_off). Each UAV is treated as a separate voice in the MIDI matrix, with each data series (pose, wind, heading controller error) assigned to a unique channel.

In this setup, the Track number identifies the sequence for each UAV's data. Channel number distinguishes the different data series for each UAV. Note number corresponds to the pitch, mapped based on the values of the UAV’s pose, wind, and heading controller error data. Velocity represents the note's intensity, linked to the data’s magnitude. Start and end times specify each note’s duration, based on the time steps of the simulation.

The last two columns, which typically store note_on and note_off message numbers, are not utilized in this dissertation. Table 3 below summarizes the components of the MIDI matrix, defining each parameter’s role in creating a sonified representation of the UAV data.

Table 3: Summary of MIDI Matrix for UAV Simulation Data

Matrix Component	Defined in UAVSonification
Track Number	Current data Series
Channel Number	Pose = 1 Wind = 2 Heading Controller Error = 3
Note Number	Assigned pitch from piano notes
Velocity	Intensity of the note
Start Time	Note start time
End Time	Note end time

To start, all notes are set to a uniform duration of 0.5 seconds. The north and east positions of both UAVs are extracted to determine their poses, and a threshold of 6.0 is used as the limit for acceptable pose change between time steps. Since pose changes are typically subtle, even with a less aggressive controller, the difference between the current and previous pose is scaled by a factor of ten to enhance the effect. A loop then evaluates the pose data, playing notes based on whether the change is within or beyond the set threshold. For pose changes within the threshold, harmonious piano notes, such as C2, D2, E2, G2, A2, and B2, are played, creating a calm, melodic soundscape [23]. When pose deviations exceed the threshold, notes are mapped to a more dissonant range from A4 to D#5. This dissonance, created by playing sharp and flat notes closely together, highlights significant pose changes [24]. Dissonant notes are categorized as unpleasant audio to the human ear while more harmonious notes are categorized as pleasant audio. Below in Figure 9 is a flow process of the UAV Sonification algorithm.

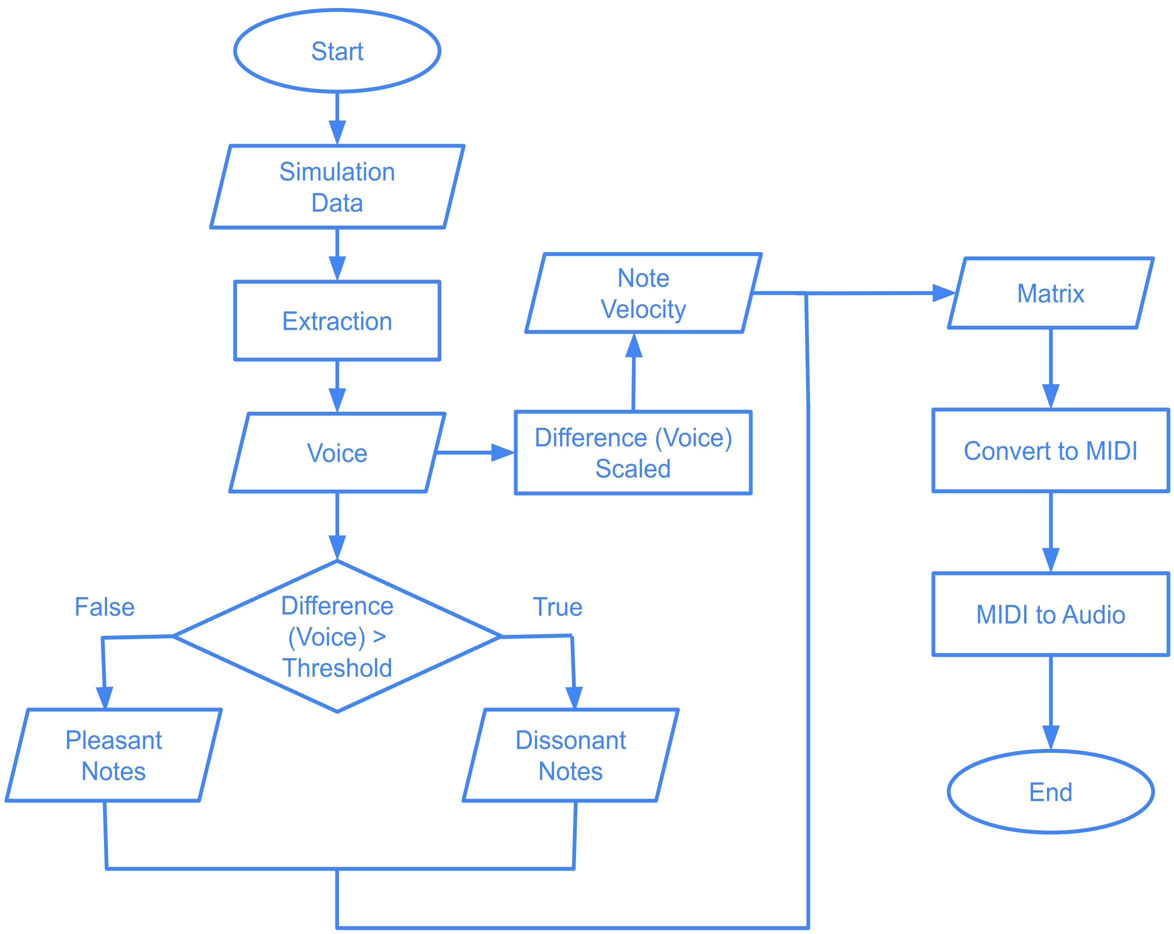

Figure 9: UAV Sonification Algorithm Flow

The intensity, or velocity, of the pose notes, is scaled between 25 and 70 on a MIDI velocity scale of 0 to 127. Since pose data is not the primary focus in this UAVSonification scenario, the velocity range remains moderate. As pose deviations grow, the note velocity also increases within this range to draw more attention to substantial changes.

The wind data is then processed and normalized to fit a MIDI piano note range from C4 to B4 and set to a low velocity of 10 to maintain it as a subtle background layer that enhances the realism of the UAV environment without dominating the soundscape.

Heading controller error data is then extracted from the CSV file, with an error threshold of 4.0 set to differentiate a well-tuned controller from one that may be struggling. Like the pose data, heading controller error within the threshold range is mapped to pleasant, consonant notes such as A2, B2, D3, E3, G3, A3, and C4, creating a smooth tonal layer that indicates stable control. However, when the heading controller error exceeds the threshold, it is mapped to a dissonant range from A4 to C7, with sharp notes between them to convey a sense of instability. The velocity of heading controller error notes is scaled from 25 to 102 on the MIDI scale to emphasize error magnitude, reaching higher intensities as the error increases.

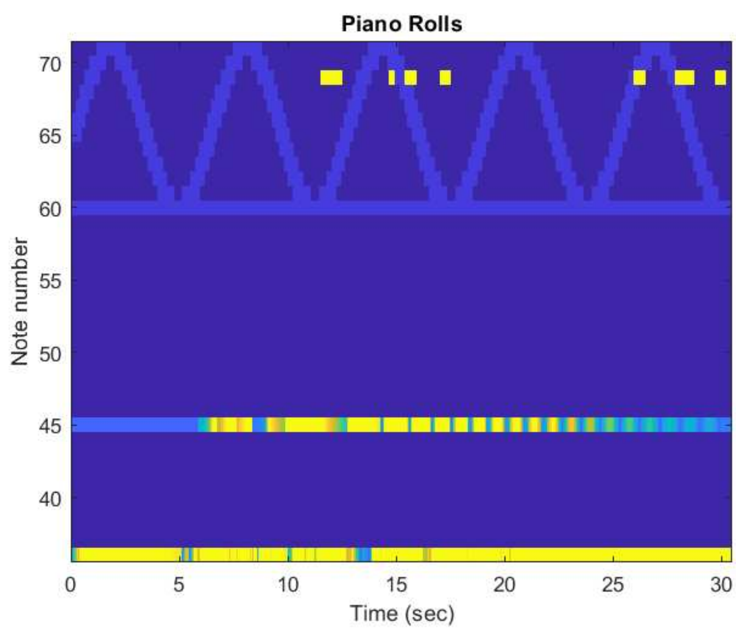

Figure 10: Piano Roll Graph

The MIDI matrix is then created, combining pose, wind, and heading controller error data for both UAVs. The preexisting matrix2midi function converts the MIDI matrix into a format compatible with MATLAB piano notes. With the midi2audio function, the MIDI data is transformed into audio, allowing the sound to be played back immediately or saved as a WAV file for later listening. For visual reference, a piano roll graph such as the one in Figure 10 above, is also displayed, providing a detailed view of each note's timing, pitch, and intensity across the sonified data.

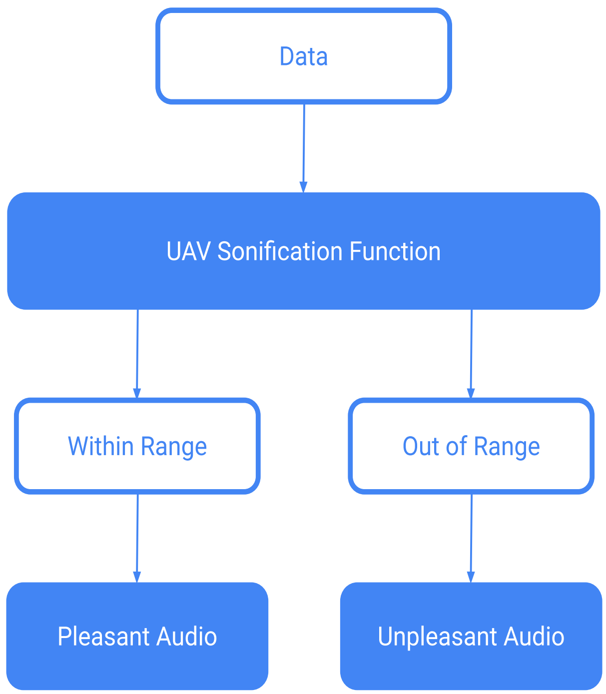

Figure 11: Summary of Sonification Mapping

Figure 11 shows the breakdown of the data summarizing the path of the input data points. Through the created UAV Sonification function, if data is within the selected range, it gets mapped to pleasant audio, if data is out of the selected range, it gets mapped to unpleasant audio.

Verification

To evaluate that the sonification effectively conveys the intended data, specifically, making the pose, wind, and heading controller error interpretable by humans, a study was conducted involving ten participants. This study received approval to be exempted by the Institutional Review Board (IRB). The participants selected were at random and had no prior experience of sonification to avoid biases. This is to assess the intuitiveness and accessibility of sonification for individuals without specialized knowledge in the field.

Chapter 5
Results

Iterations of the Formation Flight Simulation model were conducted to find an acceptable heading controller gain to indicate a decent trajectory with low error and its counterpart. These are labeled as Scenario A and Scenario B. Scenario A has a heading angle control gain of 0.97. Scenario B has a heading angle control gain of 0.30.

Below is the resulting animation trajectory and heading controller error for each scenario from the simulation.

In summary for Scenario A, the trajectories of the two UAVs shown in Figure 12 are able to follow the set waypoints with little error. After the set waypoints are reached, the UAVs continue to fly in a relatively stable manner. The resulting heading controller errors for the UAVs in this scenario is shown in Figure 13, the error decreases over time during the 30 seconds simulation with the 0.97 heading angle control gain.

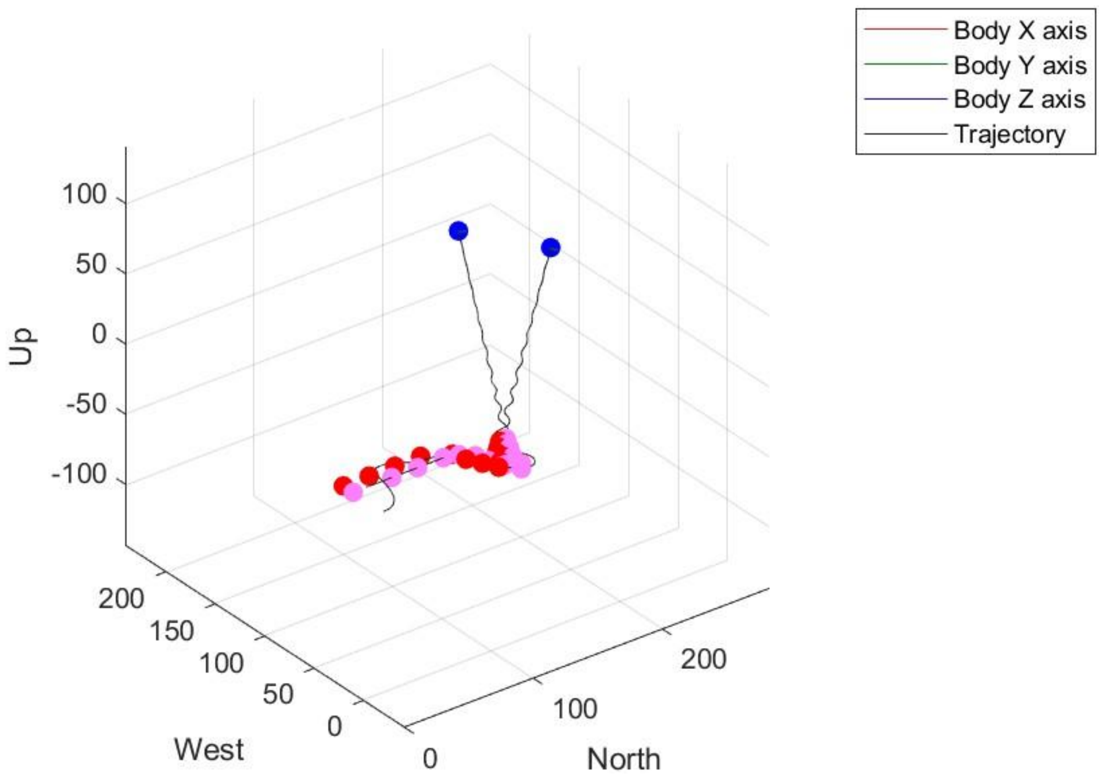

Figure 12: Resulting Trajectories of UAVs in Scenario A

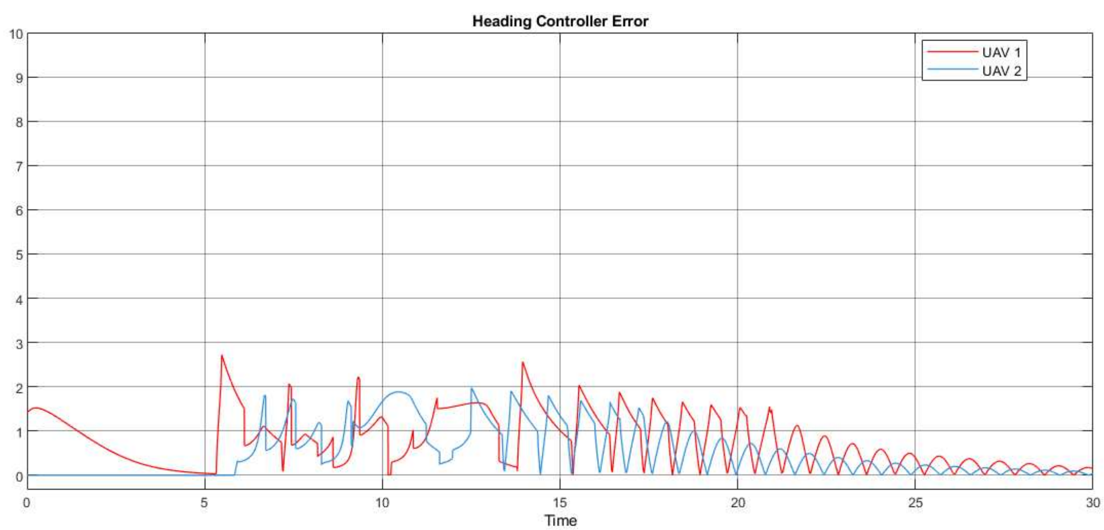

Figure 13: Resulting Heading Controller Errors for UAVs in Scenario A

In summary for Scenario B, the trajectories in Figure 14 show the UAVs' failures to reach most set waypoints, this is due to the heading angle control gain being less aggressive at 0.30. The resulting controller errors in Figure 15 show the growing error over the 30 second simulation run.

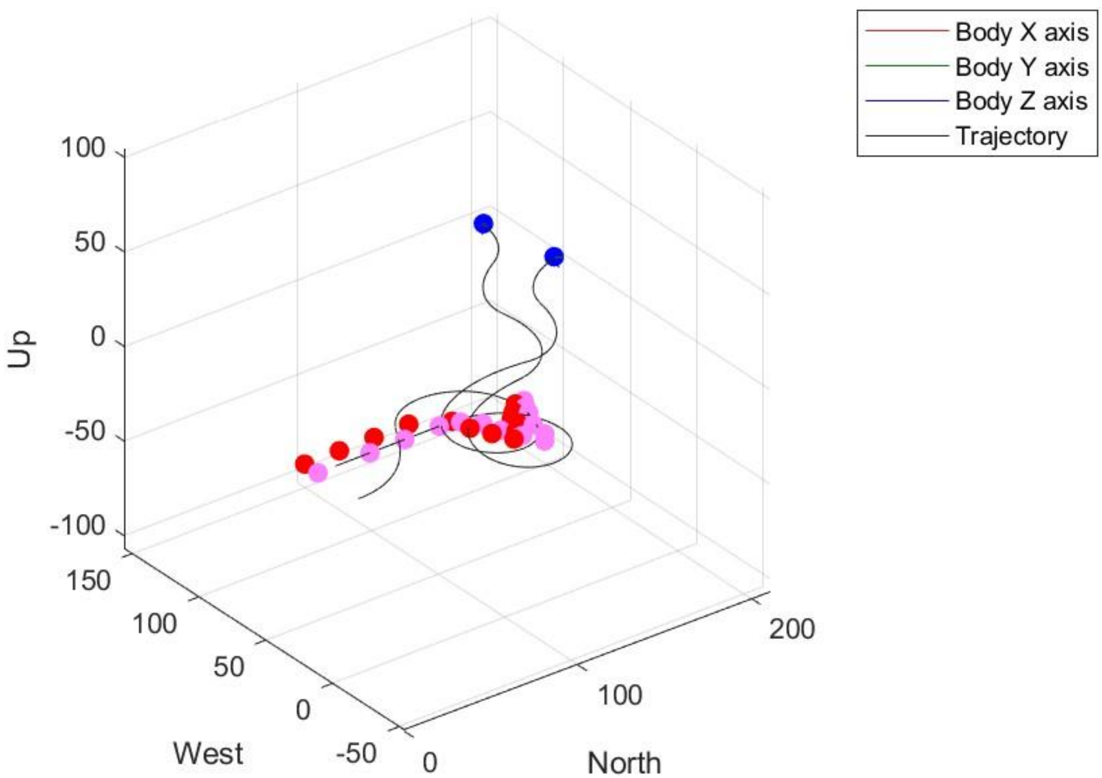

Figure 14: Resulting Trajectories of UAVs in Scenario B

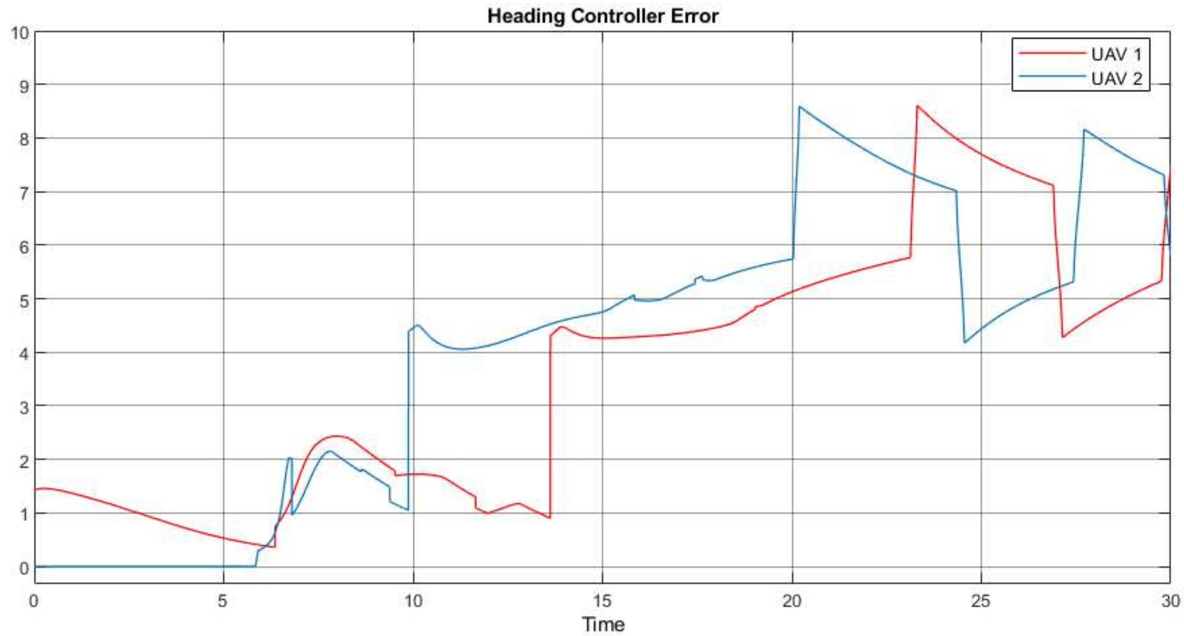

Figure 15: Resulting Heading Controller Errors for UAVs in Scenario B

CSV files for Scenario A and Scenario B are then saved and transferred to the UAVSonification.m for sonifying. The resulting piano rolls for the pose in Scenario A, to the left, and Scenario B, to the right, is shown below in Figure 16.

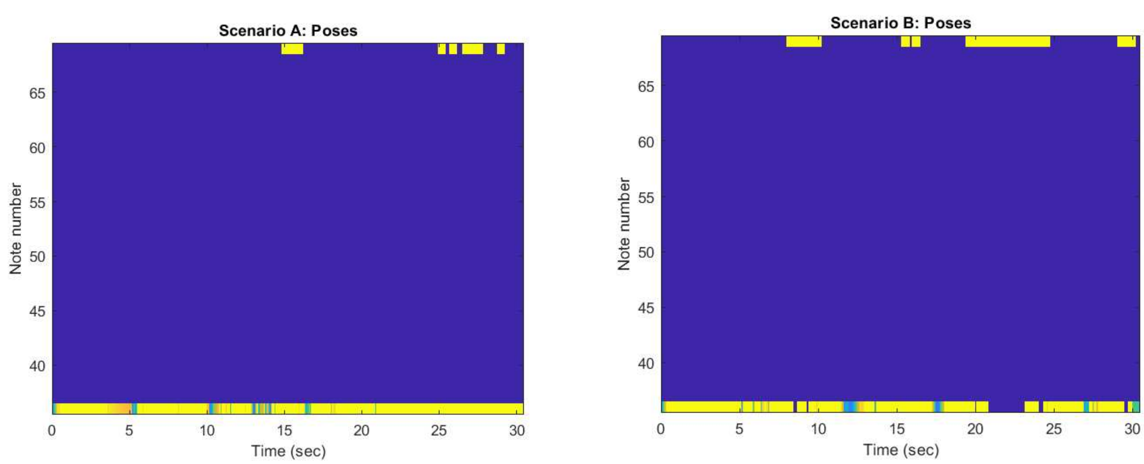

Figure 16: Resulting Piano Rolls for Pose in Scenario A and Scenario B

The resulting piano rolls for the controller errors for the scenarios is shown below in Figure 17. Scenario A is to the left, Scenario B is to the right.

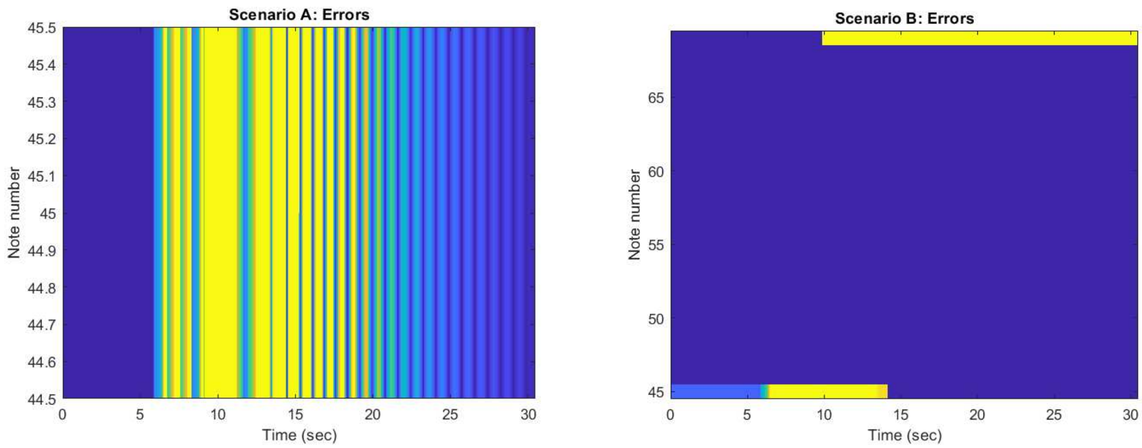

Figure 17: Resulting Piano Rolls for Controller Error in Scenario A and Scenario B

Lastly, the resulting piano rolls for the pose, controller errors, and wind is combined to make a sort of symphony of each Scenario. The sinusoidal wave in Figure 18 overlapped with the previous pose and errors can be observed. To the left is Scenario A, to the right is Scenario B.

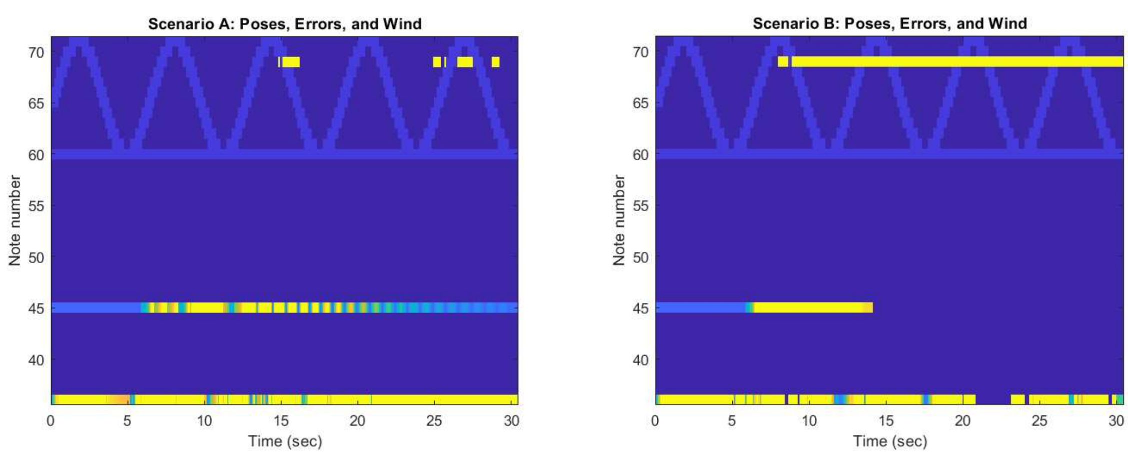

Figure 18: Resulting Piano Rolls in for full Scenario A and Scenario B

After collecting audios for these two scenarios, they were played to the ten participants in the study. The sonification successfully conveyed differences in pose, wind, and heading controller errors, as evidenced by participant feedback. They were all asked to listen to the two audios and select which one sounded more pleasant. Nine out of ten participants, despite having no prior experience with sonification, were able to distinguish between the two test scenarios. Scenario A, which represented a smoother trajectory with minimal heading controller error, was consistently perceived as more pleasant and harmonious. In contrast, Scenario B, characterized by greater heading controller error and more abrupt changes, was noted to sound more intense and dissonant, evoking discomfort.

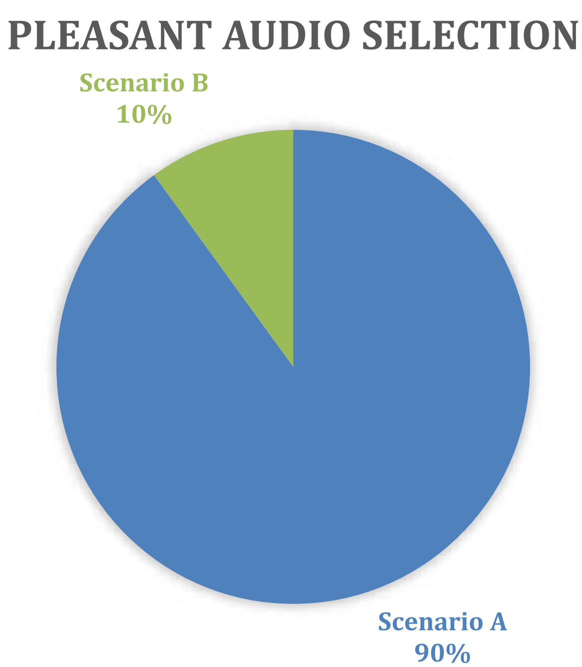

Figure 19: Participant Study Results

The results are summarized above in Figure 19. This feedback confirms that the sonification accurately reflects trajectory smoothness and controller error in a way that is simply interpretable.

Chapter 6
Discussion and Future Work

This dissertation developed a comprehensive function for sonifying UAV simulation data, introducing an innovative approach to the analysis and interpretation of UAV trajectories and controller performance. The core achievement lies in demonstrating how raw data can be translated into auditory patterns, ranging from harmonious to dissonant sounds, that intuitively convey critical information. By bridging the gap between engineering and auditory perception, this method offers a fresh perspective on UAV development, making it more inclusive and engaging for a broader audience.

One key advantage of sonification is its potential to enhance accessibility for individuals with visual impairments, providing them with an alternative medium to engage with UAV development. Beyond accessibility, the universal nature of sound transcends disciplinary boundaries, enabling individuals from diverse backgrounds to identify and interpret patterns, anomalies, or deviations in UAV behavior without requiring in-depth technical expertise. Participants in this study, for example, could discern when UAVs deviated from their intended trajectories simply by listening, underscoring the intuitive power of sonified data. The statement that "sound is universal" [25] is not merely theoretical, it is demonstrated here as a practical reality.

Despite the dissertation's promising results, the scope of the current implementation was constrained by time and existing tools. The simulation employed serves as proof of concept but leaves ample room for expansion. Future work should consider integrating more environmentally realistic scenarios, such as turbulence, complex wind patterns, or multi-terrain navigation, to better mimic real-world challenges. During the development phase, various platforms, including Gazebo and Unreal Engine, were evaluated for their suitability, though the final implementation focused on MATLAB for its robust computational and sonification capabilities. However, the current sonification is limited to piano notes, an area ripe for exploration. By leveraging advanced MIDI functionalities, future iterations could expand the auditory palette to include diverse instrument sounds, timbres, or even dynamic soundscapes tailored to specific UAV behaviors.

GitHub

Additionally, the dissertation shares its resources openly via a GitHub repository, providing access to the model used, functions, and documentation as shown in Figure 20. This open-source approach encourages collaboration and adaptation, enabling others to customize the framework for different types of simulations or datasets. The versatility of the system means it could accommodate varied inputs, from autonomous cars to marine robotics, illustrating its potential beyond UAVs.

UAV-Sonification

Sonification of multiple UAVs

Background

This project expands upon the MATLAB formation flight simulation (link: https://www.mathworks.com/help/uav/ug/simulating-multiple-UAV-simulink-system-object-blocks.html) to simulate and sonify the trajectories of two UAVs with wind conditions. Using UAVSonification, data from these simulations are transformed into sound, allowing users to hear variations that indicate key dynamics:

• Controller errors

• Wind oscillations

• Drastic changes in the UAVs' pose

A well-controlled trajectory and stable heading controller produce more harmonious sounds, while disruptions or deviations add dissonance.

Applications

• Planning and analyzing UAV trajectories (up to 100 UAVs) with Simulink's Waypoint Follower and Guidance Model.

• Auditory insight into flight path smoothness and controller effectiveness, along with wind oscillations.

UAVSonification transforms simulation data into sound, offering a unique, auditory approach to understanding UAV behavior in response to control dynamics and environmental conditions.

Figure 20: GitHub Project Page

Further avenues for exploration are particularly exciting. For instance, how would sonifying the behavior of UAV swarms sound? Would their collective trajectories create a cacophony, or could they be orchestrated into a symphony? The intersection of sonification and artistic expression opens intriguing possibilities for creative projects. Could UAV trajectories inspire musical compositions or immersive auditory installations? While this dissertation provides the foundation, it also highlights the limitless potential of UAVSonification to advance both technical and artistic domains.

This dissertation underscores the untapped potential of sonification in UAV development. By making complex data more intuitive and accessible, it lays the groundwork for innovations that could transform the way we design, test, and interact with UAV systems.

Appendix A

UAV Sonification Code

```
%% Sonifying UAVs
% Following the DiffusionMusic model using MIDI from the simulation
% dataseries
% Sonification: Mapping notes to pose for uav1 and uav2, winds, and
% controller error and displaying piano notes

data = readmatrix('simulation_data.csv');
time = data(:, 1);
posData = data(:, 6:9);
wind = data(:, 10:11);
error = data(:, 12:13);

% Voice is data series
% # of columns
voices_pos = size(posData, 4);
voices_win = size(wind, 2);
voices_err = size(error, 2);

numDataPoints = length(time);
% Can modify here for different sounds
% velocity = 100; % Intensity of the note

velocityBase_p = 25; % Base intensity for pose notes
velocityRange_p = 70; % up to 127
noteDuration = 0.5;
MATRIX = [];
% MIDI uses a matrix of size Nx8
% Track #, Channel #, Note #, Velocity, Start Time (s), End Time (s),
% Message #
% of note_on, Message # of note_off

% 2 voices = 2 uavs

% Pose threshold for sound type
% North and East
pos_thrsh = 6;
```

```
for i = 1:voices_pos
    series = posData(:, i); % Current data series

    % How drastic is the change of pose between points, scaled to 10
    pos_diff = abs(diff(series)*10);

     % Create MIDI notes for each time step
     for j = 1:numDataPoints - 1
        M = zeros(1, 6);
        M(1) = i; % Track number
        M(2) = 1; % Channel number

        % If difference is within threshold, pleasant notes played
        if pos_diff(j) <= pos_thrsh
            % Map values below boundary to pleasant range
            % Pleasant intervals (C2, D2, E2, G2, A2, B2)
            pitches = [36, 38, 40, 43, 45, 47];
            M(3) = pitches(mod(round(rescale(series(j), 1, ...
                length(pitches))) - 1, length(pitches)) + 1);
        else
            % Out of threshold will sound "bad"; pose is drastically
            % changing
            % Dissonant intervals (A4, D#5)
            dissonantPitches = linspace(69,75);
            M(3) = dissonantPitches(mod(round(rescale(series(j), 1, ...
                length(dissonantPitches))) - 1, ...
                length(dissonantPitches)) + 1);
        end

        M(4) = min(velocityBase_p + round(velocityRange_p * ...
            pos_diff(j)), 127); % scaling velocity to change in pose
        M(5) = time(j); % Note start time
        M(6) = time(j) + noteDuration; % Note end time

        % Append the note to the MATRIX
        MATRIX = [MATRIX; M];
     end
end

wind_velocity = 10; % Velocity here is the intensity of the note, ...
% wind is lower
for i = 1:voices_win
    series = wind(:, i); % Current data series

     % Normalize data series to fit MIDI note range for Piano
```

```
    minNote = 60; % Lowest note (C4)
    maxNote = 71; % Highest note (B4)
    normalizedSeries = rescale(series, minNote, maxNote);
    % Create MIDI notes for each time step
    for j = 1:numDataPoints - 1
        M = zeros(1, 6);
        M(1) = i; % Track number
        M(2) = 2; % Channel number
        M(3) = round(normalizedSeries(j)); % Note number
        M(4) = wind_velocity; % Velocity
        M(5) = time(j); % Note start time
        M(6) = time(j) + noteDuration; % Note end time

        % Append the note to the MATRIX
        MATRIX = [MATRIX; M];
    end
end

velocityBase_e = 25; % Base intensity for controller error notes
velocityRange_e = 102; % up to 127
% MIDI track for error
% Error limit for "good" controller
err_limit = 4;
for i = 1:voices_err
    series = error(:, i);

    for j = 1:numDataPoints - 1
        M = zeros(1, 6);
        M(1) = i; % Track number
        M(2) = 3; % Channel number

        % Adjust pitch based on error limit
        if series(j) <= err_limit
            % Map values below boundary to pleasant range
            % Pleasant intervals (A2, B2, D3, E3, G3, A3, C4)
            pitches = [45,47,50,52,55,57,60];
            M(3) = pitches(mod(round(rescale(series(j), 1, ...
                length(pitches))) - 1, length(pitches)) + 1);
        else
            % Map values above boundary to dissonant range
            % Dissonant intervals (A4,C7)
            dissonantPitches = linspace(69,96);
            M(3) = dissonantPitches(mod(round(rescale(series(j), 1, ...
                length(dissonantPitches))) - 1, ...
                length(dissonantPitches)) + 1);
        end
```

```
        M(4) = min(velocityBase_e + round(velocityRange_e * ...
            series(j)), 127); % scaling velocity to change in error
        M(5) = time(j); % Note start time
        M(6) = time(j) + noteDuration; % Note end time

        % Append the note to the MATRIX
        MATRIX = [MATRIX; M];
    end
end

% Create the MIDI file
% This generates notes capable in MATLAB and into a MIDI file
midi_new = matrix2midi(MATRIX);
writemidi(midi_new, 'csvSonification.mid');
% Convert to audio
[y, Fs] = midi2audio(midi_new);
% Optional: Write audio to a WAV file
audiowrite('csvSonification.wav', y, Fs);

% Play the audio
soundsc(y, Fs);

% Display piano roll
Notes = midiInfo(midi_new);
[PR, t, nn] = piano_roll(Notes, 1);

figure;
imagesc(t, nn, PR);
axis xy;
xlabel('Time (sec)');
ylabel('Note number');
```

References

[1] A. Snaith, Sound and Literature, Cambridge: Cambridge University Press, 2020.

[2] K. Z. P. W. A. M. Amal Kacem, "Wave space sonification of the folding pathways of protein molecules modeled as hyper-redundant robotic mechanisms," *Multimedia Tools and Applications,* vol. 83, no. 2, pp. 1-21, 2024.

[3] NASA Science Editorial Team, "NASA," NASA, 14 June 2022. [Online]. Available: https://science.nasa.gov/science-research/astrophysics/data-sonifications/. [Accessed 22 January 2025].

[4] K. Z. A. M. Amal Kacem, "Transforming Motion Into Sound: A Novel Sonification Approach for Teams of Mobile Robots," in *2024 ASME/ISCIE International Symposium on Flexible Automation (ISFA2024)*, Seattle, 2024.

[5] I. L. R. B. I. T. Bastian Orthmann, "Sounding Robots: Design and Evaluation of Auditory Displays for Unintentional Human-robot Interaction," *ACM Transactions on Human-Robot Interaction,* vol. 12, no. 4, pp. 1-26, 2023.

[6] O. B. M. V. Frederic Anthony Robinson, "Designing Sound for Social Robots: Candidate Design Principles," *International Journal of Social Robotics,* vol. 14, pp. 1507-1525, 2022.

[7] R. A. Fitch, "Sonification as a Pathway to Knowledge," Oregon State University , Corvallis, 2024.

[8] M. A. Dante Tezza, "The State-of-the-Art of Human–Drone Interaction: A Survey," *IEEE,* vol. 7, pp. 167438 - 167454, 2019.

[9] B. W. Michael Nees, "Theory of Sonification," in *Principles of Sonification: An Introduction to Auditory Display*, Logos Publishing House, 2012, pp. 9-39.

[10] L. Bornmann, "The sound of science: Data sonification has emerged as possible alternative to data visualization," *EMBO Reports,* vol. 25, no. 9, pp. 3743-3747, 2024.

[11] R. B. Adrian B. Latupeirissa, "PepperOSC: enabling interactive sonification of a robot's expressive movement," *Journal on Multimodal User Interfaces,* vol. 17, pp. 231-239, 2023.

[12] R. S. L. S. G. W. Lisa Zahray, "Robot Gesture Sonification to Enhance Awareness of Robot Status and Enjoyment of Interaction," *IEEE,* Vols. 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN), pp. 978-985, 2020.

[13] M. C. H. D. G. Birsen Donmez, "Auditory Decision Aiding in Supervisory Control of Multiple Unmanned Aerial Vehicles," *Sage Journals,* vol. 51, no. 5, 2009.

[14] R. A. D. K. E. A. C. Liam Roy, "Toward Improving User Experience and Shared Task Performance with Mobile Robots through Parameterized Nonverbal State Sonification," in *Sound and Robotics*, New York, Chapman and Hall/CRC, 2023, p. 23.

[15] MathWorks, "fixedwing," The MathWorks Inc., 2024. [Online]. Available: https://www.mathworks.com/help/uav/ref/fixedwing.html. [Accessed 22 January 2025].

[16] V. Dobrokhodov, "Kinematics and Dynamics of Fixed-Wing UAVs," in *Handbook of Umanned Aerial Vehicles*, Springer Reference, 2014, pp. 243-277.

[17] MathWorks, "Guidance Model," The MathWorks Inc., 2024. [Online]. Available: https://www.mathworks.com/help/uav/ref/guidancemodel.html. [Accessed 22 January 2025].

[18] D. M. Huber, The MIDI Manual, New York, 2007.

[19] MathWorks, "Musical Instrument Digital Interface (MIDI)," The MathWorks Inc., 2024. [Online]. Available: https://www.mathworks.com/help/audio/musical-instrument-digital-interface-midi.html. [Accessed 22 January 2025].

[20] S. H. K. Suzuki, "Robotic interface for embodied interaction via dance and musical performance," *IEEE,* vol. 92, no. 4, pp. 656-671, 2004.

[21] MathWorks, "Simulate Multiple Fixed-Wing UAVs in Simulink Using System Objects," The MathWorks Inc., 2024. [Online]. Available: https://www.mathworks.com/help/uav/ug/simulating-multiple-UAV-simulink-system-object-blocks.html. [Accessed 22 January 2025].

[22] K. Schutte, "MATLAB and MIDI," [Online]. Available: https://kenschutte.com/midi/. [Accessed 22 January 2025].

[23] "Inspired Acoustics," ENTEL, [Online]. Available: https://inspiredacoustics.com/en/MIDI_note_numbers_and_center_frequencies. [Accessed 22 January 2025].

[24] B. J. B. J. Y. H. L. C. S. S. Eve Poudrier, "The Influence of Dissonance on Listeners' Perceived Emotions in Rhythmically Complex Musical Excerpts," *Auditory Perception & Cognition,* vol. 7, no. 4, pp. 291-318, 2024.

[25] J. Gottlieb, "Music Everywhere," *The Harvard Gazette,* 21 November 2019.