MMI 503/MMI 603 - Audio Signal Processing 2

Project 1 Report

Author: Nicholas Tong

Date: 2/2/2023

Project Description and Problem Statement:

The project aims to develop a set of audio analysis functions that can provide insights into the loudness and spectral content of audio signals. The problem we are trying to solve is to provide a solution to effectively analyze audio signals and extract valuable information from them. The proposed solution is a MATLAB code that generates the audio signals, analyzes them, and plots the results, providing a user-friendly and efficient method for analyzing audio signals.

Methodology:

The analysis system consists of two main components: audio signals generation and audio analysis. The methodology for this project involves developing a set of audio analysis functions that can provide insights into the loudness and spectral content of audio signals. The functions will be implemented using MATLAB and will include the following capabilities:

1. Signal Generation: The audio signals are generated using MATLAB code, which includes a white noise signal, a sine tone @ 1000 Hz, and an exponential sine sweep from 100-20k Hz.
2. Signal Analysis: The code will include functions for analyzing audio signals such as computing the root mean square (RMS) level and spectral content of the signal. The audio signals will be analyzed using two functions, the "rms_loudness" function, and the "spectral_analyzer" function
3. Loudness Analysis: The code will include functions for measuring loudness, such as computing and returning the root-mean-squared loudness in decibels (dB).
4. Spectral Content Analysis: The code will include functions for analyzing the spectral content of audio signals, such as computing and returning the frequency bin values of the buffer.
5. Data Storage and Plotting: The loudness and spectral content will be stored in pre-allocated arrays which will be then used to plot the RMS-loudness and frequency content over time.

Block Diagram:

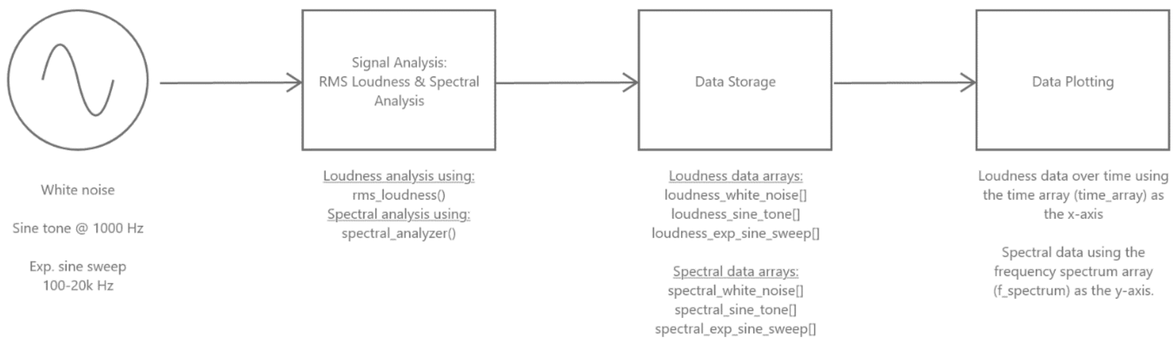

Figure 1

Plots:

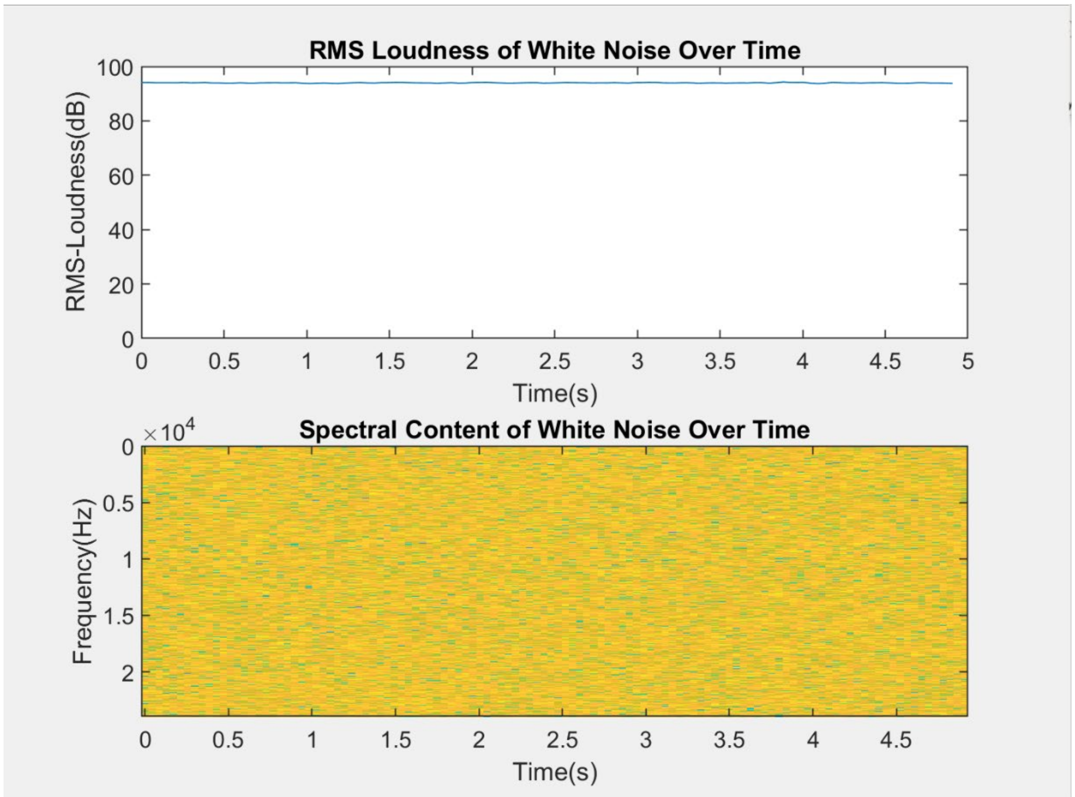

Figure 2. Note: RMS-Loudness y-value normalized with ylim([0 100])

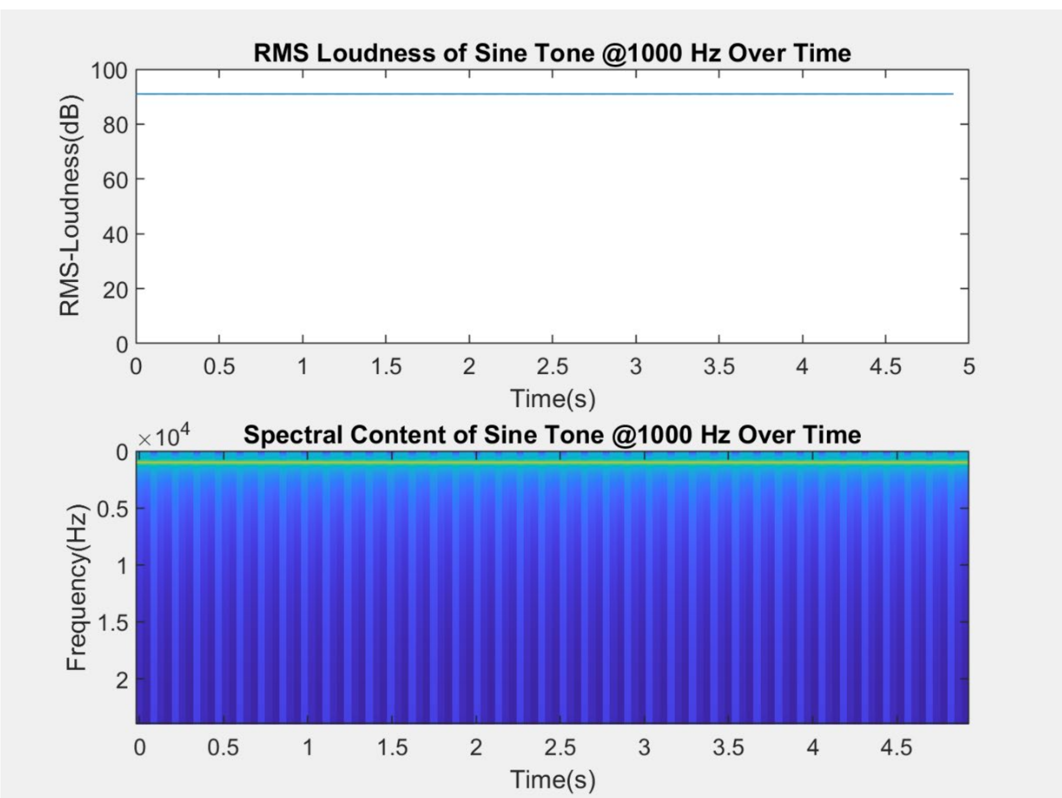

Figure 3. Note: RMS-Loudness y-value normalized with ylim([0 100])

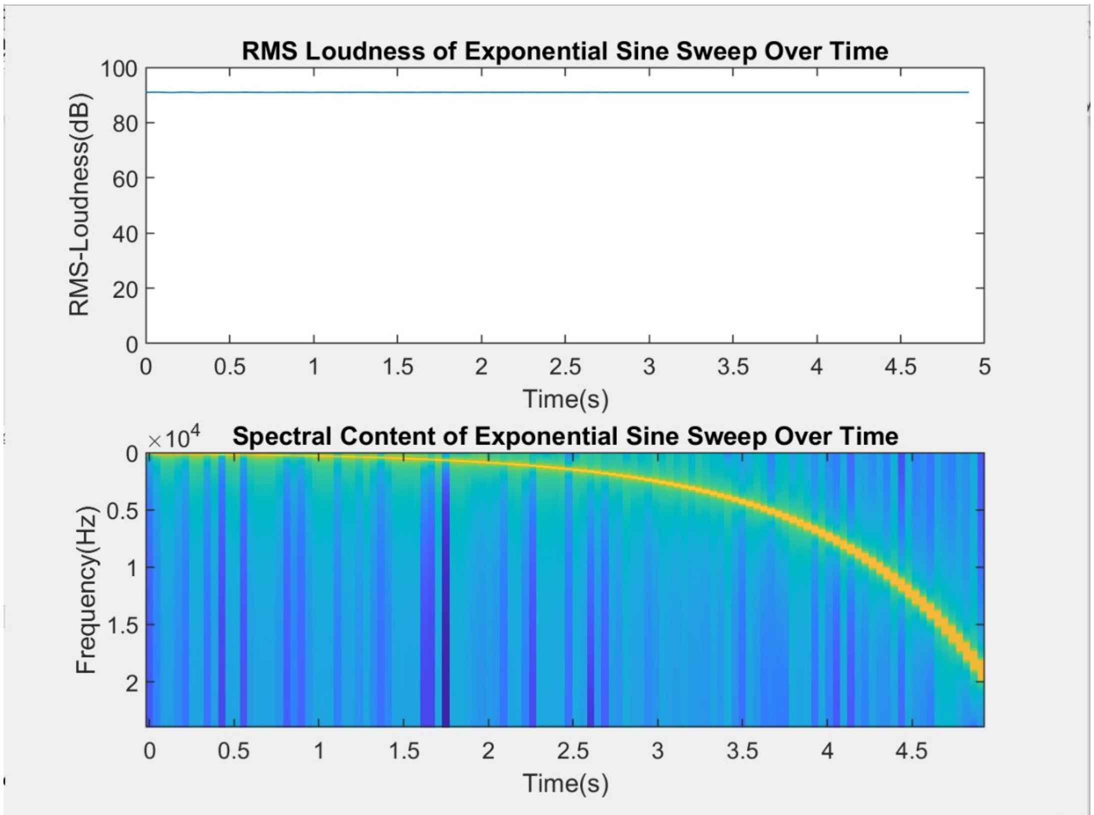

Figure 4. Note: RMS-Loudness y-value normalized with ylim([0 100])

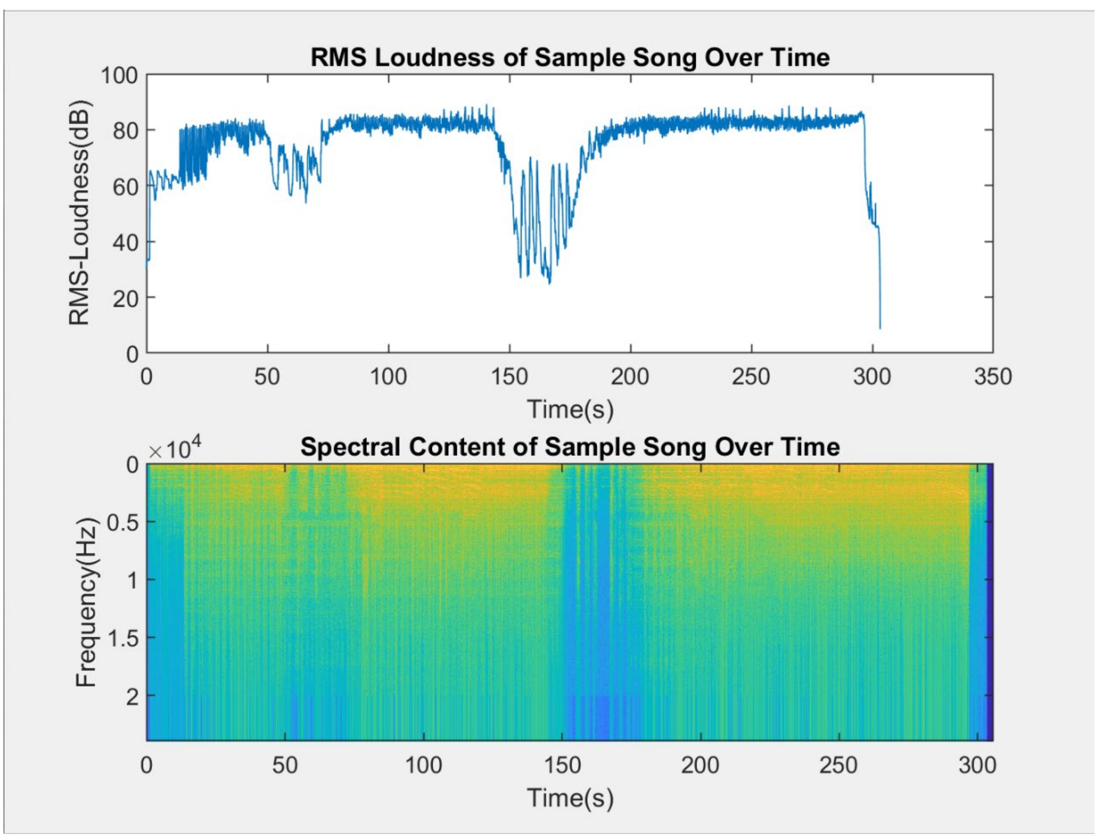

Figure 5. Note: characteristics like song sections and bass-heavy content can be visually seen

Results and Discussion:

The results of the audio analysis demonstrate the effectiveness of the functions in providing insights into the loudness and spectral content of audio signals. The root mean square (RMS) level and spectral content were computed for the generated audio signals, and the results were plotted over time. The loudness analysis showed the RMS levels of the generated signals remained relatively flat across time at approximately ~90 dB_{rms} for the sine tone and exponential sine sweep signals and approximately ~93 dB_{rms} for the white noise signal. The spectral content analysis demonstrated that the sine tone was predominantly composed of energy at 1000 Hz, with very little energy at other frequencies. The exponential sine sweep showed a gradual logarithmic increase in energy from 100 Hz to 20k Hz. The white noise signal showed an equal distribution of energy across all frequencies.

Analyses systems like these can be used to identify the different characteristics of audio signals and are useful in applications such as audio compression, restoration, and signal processing. The results of the analysis can also be used to visualize and identify different sections of an audio file, which can be useful for editing and signal-processing applications. The information extracted from this analysis can be used in various applications. For example, the loudness analysis can be used in ensuring that audio signals meet industry standards for loudness, while the spectral content function can be used to analyze the frequency distribution of sound energy in an audio signal and help design filters and equalization settings. Additionally, the code and functions developed in this project could be integrated into larger systems and used as a building block for more advanced and larger-scaled audio applications.