EE473 Final Project Report

Umut Eren Usturalı,Ekrem Fatih Yılmazer

January 2020

1 Introduction

Our aim in this project is to synthesize instrument sounds. A synthesizer is an electronic musical instrument that generates audio signals. Specifically, our goal is to create the timbre of a guitar and a flute.

The timbre of a musical instrument can be considered to be the sum of several harmonic signals modulated by envelopes. This approach to analyzing and synthesizing sound is called additive synthesis. The envelopes here have a few important features. The envelope consists of mainly four different parts. These parts are usually called attack, decay, sustain and release. Attack refers the initial portion of the note during which amplitude of harmonics increase rapidly. In the decay part the amplitudes slightly decrease and then remain roughly constant for the sustain portion. The release is the final part in which the amplitude goes to zero and the note ends.

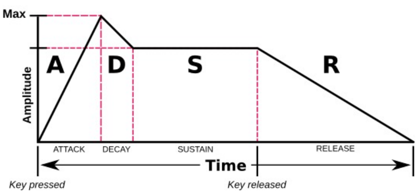

Figure 1: ADSR (attack-decay-sustain-release) envelope

Mathematically, this can be represented as

$$x(t) = \sum_{n=1}^{N} a_n(t) \cos(n\omega t) \tag{1}$$

where $x(t)$ is the sound signal for a note, N is the number of harmonics, and $a_n(t)$ is the envelope of the n-th harmonic. [1] In order to synthesize the sound of an instrument, $a_n(t)$ for this instrument needs to be determined. This can be done from a recording of the instrument [2] in a few different ways. In this project, we use two of these methods. These are described below.

2 Chunk Method

The first method is called the chunk method, in which the audio signal is divided into smaller chunks. The discrete Fourier transform of each of these chunks is computed via FFT. Under the assumption that $a_n(t)$ do not change too rapidly within single chunk, the peaks in these DFT correspond to the value of $a_n(t)$ within the chunk. In order to construct $a_n(t)$ for the entire duration of the audio signal, we then use linear interpolation. Sound generation is accomplished by generating pure sinusoidals for each harmonic and modulating these signals by the corresponding envelopes computed from the original recording.

In Figure 2, the envelopes computed for a guitar note and for a flute note are shown. In terms of quality, the flute sound is convincing but the synthesized guitar sound is not satisfactory.

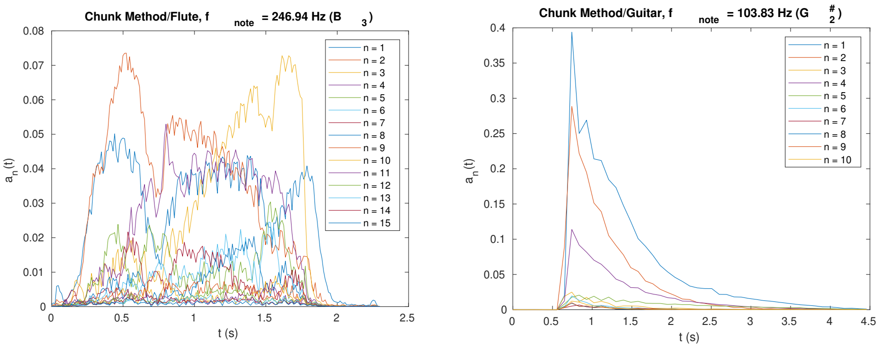

Figure 2: Chunk method computed envelopes

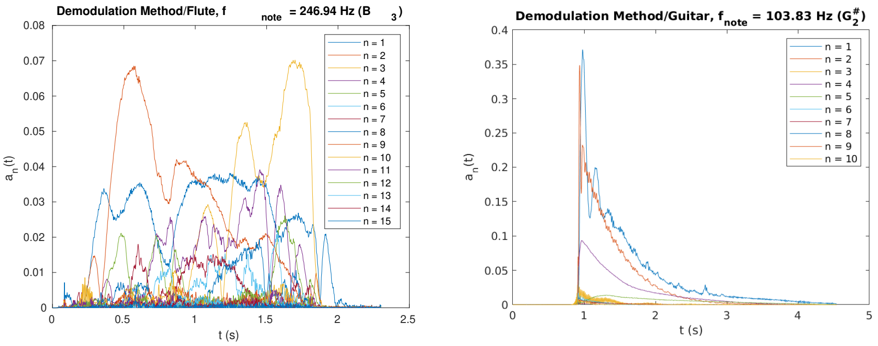

Figure 3: Demodulation method computed envelopes

3 Demodulation method

The second method relies on the observation that $a_n(t)$ do not have large bandwidths. The action of modulation by a sinusoidal is equivalent to a shift in the Fourier domain, which can be reversed by multiplying with the same sinusoidal again and using a lowpass filter. Mathematically,

$$\mathrm{LP}\left[x(t)cos(n\omega t)\right] = \mathrm{LP}\left[\sum_{r=1}^{N} a_r(t)\cos\left(n\omega t\right)\cos\left(r\omega t\right)\right] = a_n(t). \tag{2}$$

Parks-McClellan algorithm is utilized to create a discrete-time lowpass filter for this purpose. Since the frequency spacings between the harmonics and thus the critical frequency of the lowpass filter is very small compared to the sampling frequency, downsampling is needed for better filter performance. The ratio of downsampling depends on the note frequency and the number of harmonics that are needed.

In Figure 3, the envelopes obtained via this method are shown. The quality of the flute sound is comparable to the chunk-method result, and the quality of the guitar sound is greatly improved. Thus it is possible to conclude that this method is the better choice overall.

References

[1] James W Beauchamp. *Analysis, synthesis, and perception of musical sounds*. Springer, 2012, pp. 1–89.

[2] Lawrence Fritts. *University of Iowa Electronic Music Studios*. 2020. URL: http://theremin.music.uiowa.edu/MIS.html.