Digital Signal Processing Mini Project

Cao Zhengyang
Department of Electronics and Electrical Engineering
12110623@mail.sustech.edu.cn

I. Introduction

The objective of this experiment was to transform digital music sheet into audio waveforms using MATLAB programming. The fundamental elements of music, such as pitch, rhythm, and timbre, were explored. By analyzing the characteristics of digital sheet music, the computation of musical note frequencies, generation of waveforms, and simulation of different musical instruments' sounds were achieved.

II. Understanding music sheet and code them

Digital sheet music utilizes numerical symbols (1 to 7) to represent the seven basic degrees of the musical scale. Correspondingly, these numbers are represented as C to B or Do to Ti. Rests are denoted by 0. The duration of musical notes is represented using hyphens, short lines, and dots in digital sheet music.

Using double date type to represent all the elements mention above:

- tone: 0 to 7
- scale A to G: 1 to 7
- octave: 1 for one octave higher, -1 for one octave lower, 0 remain same
- rising: 1 for higher, -1 for lower, 0 remain same
- rhythm: how many sec should a note last
- fs: sample frequency of a note

III. Envelope Decay Function and Adjustment of Harmonic Energy Ratios

A. Envelope Decay Function

In real musical instruments, vibrations naturally attenuate over time, creating a more organic sound. To capture this characteristic, an envelope decay function is applied to the waveform. The exponential decay function utilized in this experiment follows the code:

```
y = y.*(exp(-2*t/rhythm)-0.15*sin(t));
```

The result ends up like:

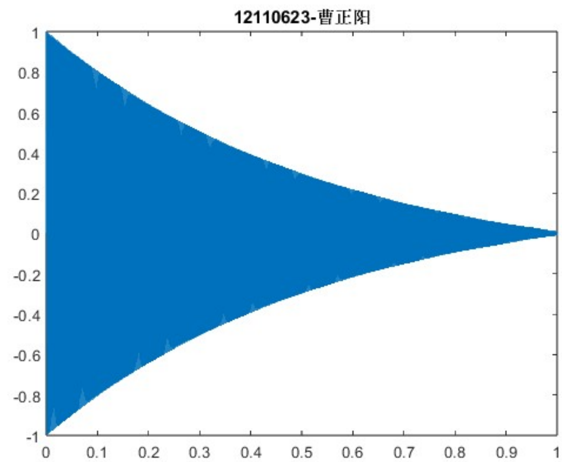

Fig. 1. Favourite Envelope Decay

The end of the note is barely zero, which will not effect next note.

B. Adjustment of Harmonic Energy Ratios

Different musical instruments produce varying ratios of harmonics. By adjusting the energy ratios of harmonics, the experiment aimed to simulate diverse instrument timbres. Experimentation with different harmonic energy ratios, along with the analysis of waveform and frequency spectrum changes, facilitated the selection of appropriate ratios to achieve the desired instrument timbres. To find the proper ratio, i referenced a Penn State University article on dynamics and the piano sound. [1] The harmonics component is not decaying very much. So, i use function like this:

```
waves = 0.8*sin(2*pi*f*t)+0.4*sin(2*pi*2*
    f*t)+ 0.2*sin(2*pi*3*f*t)+0.1*sin(2*pi
    *4*f*t)+0.05*sin(2*pi*5*f*t)+0.025*sin
    (2*pi*6*f*t);
```

The result ends up like:

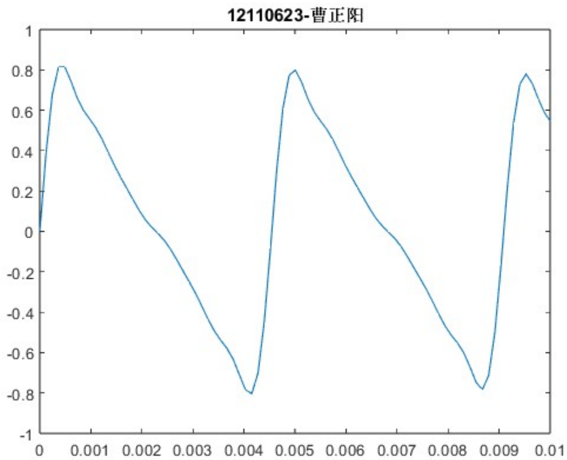

Fig. 2. Harmonic wave

IV. Results and Analysis

By turning sheet into code, we get the music.

```
clear;
%config parameter
scale = 3;
octave = 0;
rising = 0;
rhythm = 0.4;
delaying = 0;
fs = 44100;
waves = 0;
wave = [
    [0 scale octave rising rhythm 3 fs];
    [6 scale octave rising rhythm 1 fs];
    [7 scale octave rising rhythm 1 fs];
    [1 scale octave 1 rhythm -1 fs];
    [7 scale octave rising rhythm -1 fs];
    [1 scale octave 1 rhythm delaying fs
       ];
    [3 scale octave 1 rhythm delaying fs
       ];
    ...
```

The music is stored in skycity.wav

V. Conclusion

This experiment elucidates the significance of envelope decay in music signal processing. By implementing various decay functions, the synthesized waveforms acquire depth and authenticity, making the simulated music more compelling and natural to the listener. Understanding and manipulating envelope decay functions are essential skills in creating lifelike musical simulations.

References

[1] Russell, D.A. (1997) Hammer nonlinearity, dynamics and the piano sound, Hammer nonlinearity, dynamics and the Piano Sound. Available at: https://www.acs.psu.edu/drussell/Piano/Dynamics.html (Accessed: 07 November 2023).