**信号与系统——MATLAB综合实验**

音乐合成实验报告

学号：2019011008

无92 刘雪枫

2021年9月5日

目录

# 实验目的

1. 了解基本的乐理知识；
2. 深入理解离散量与连续量的关系，理解离散傅里叶变换在信号处理中的应用；
3. 学习MATLAB工具，在合成音乐的过程中学会运用MATLAB进行频谱分析与信号处理；
4. 在动手实验的过程中提高实践能力以及解决问题的能力，体会信号处理的乐趣。

# 实验平台

本次实验使用Windows平台上的MATLAB R2021a (64-bit) 进行实验。

# 实验原理

## 乐音

人类听觉可以感受到的声音大体上可划分为噪音、语音、乐音，等等。其中，乐音的基本特征可以用基波频率（简称“基频”）、谐波频率分量和包络波形来描述。乐音的声调由基频决定，而不同乐器发出的乐音的音色则由高次谐波分量决定。此外，包络也是描述乐音特性的重要因素。本实验将通过模拟乐器发出的谐波分量和包络来合成电子音乐。

## 时频分析

振动随时间的变化关系称作振动的时域波形；组成振动的各单频正弦波的强度与频率的关系称作振动的频域波形。时域波形与频域波形的关系由傅里叶变换式给出：

以上是针对连续信号的情况。计算机不可能存储连续的信号，对此可以将连续信号离散化。即每隔一段固定的时间*T*s对连续信号*f*(*t*) 进行抽样，得到离散序列*x*(*n*)。每秒钟的采样次数： 称为“采样率”。如果信号是周期信号，则只需对整数倍个周期进行采样。设采样总时间为*T*­1，则基频一定为 的整数倍。设采样点数为*N*，由此得到离散傅里叶变换：

由此便可得到离散周期信号的频谱，用于近似连续周期信号的频谱。通过对频谱进行分析和设置，我们就能够得到离散时域信号，进而合成音乐。

# 实验内容

## 简单的合成音乐

### 简单合成《东方红》

先用单频正弦波来演奏《东方红》，只需要按《东方红》曲谱逐个音符进行拼接即可。将F大调的各个音的频率储存在矩阵tunes中，将曲谱储存在矩阵song中，其中第一列代表音调，第二列代表持续时间（时间为1代表四分音符）。拼接单频正弦波的关键代码如下：

for i = 1 : 1 : len

f = tunes(song(i, 1));

time\_len = song(i, 2) \* beat\_len;

t = linspace(0, time\_len - 1 / Fs, Fs \* time\_len)';

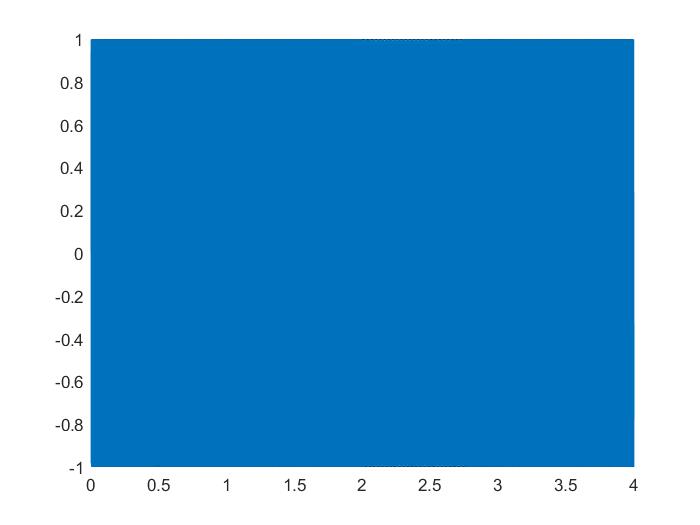
tmp\_res = sin(2 \* pi \* f \* t);

res = [res; tmp\_res];

end

由此得到的向量res即为音乐向量。

音乐向量的波形如下：



可以看到，音乐的幅度处处相等。播放音乐，发现在相邻两个音符之间存在“啪”的一声，非常影响音乐质量。究其原因，可能是相邻两个音符之间的相位不连续造成音符连接处存在高频分量导致的。

本问题的完整代码位于文件hw\_1\_2\_1\_1.m中，得到的音乐文件为hw\_1\_2\_1\_1.wav。

### 加入包络

为了解决相位不连续的问题，并模拟真实乐器的强度变化，需要给正弦波加上包络。为了让音符接触处的幅度为零，并且模拟声音先增强后逐渐减弱的过程，引入形如下图的包络函数：

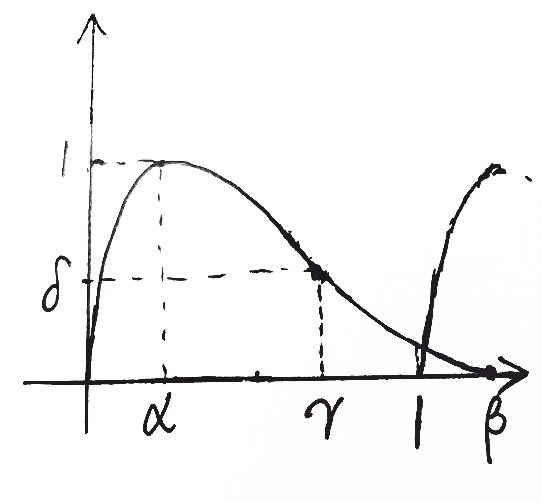
上升

过冲

保持

下降

为了拟合该包络函数，我们计划采用三段抛物线进行拟合：



上图中，0~*α*、*α*~*γ*、*γ*~*β*三段分别为三段抛物线，从左向右一次记作*C*1、*C*2、*C*3。抛物线的*C*1顶点位于 (*α*, 1) 处，且经过点 (0, 0)，因此可以写出它的方程：

同理，*C*2、*C*3的顶点分别为 (*α*, 1) 和 (*β*, 0)，因此写出它们的方程：

由于*C*2过点 (*γ*, *δ*)，因此可以得到：

由于包络函数应当尽量光滑，因此令在各段抛物线连接处满足函数值连续、函数值的一阶导数连续。因此*C*2和*C*3的连接处满足方程：

注意方程是过定的，因此*α*、*β*、*γ*、*δ*四个参量不独立。采用*α*、*β*、*γ*作为自由参量得到：

合适选取参量*α*、*β*、*γ*即可得到包络函数。取：

得到包络函数，封装为MATLAB函数envelope，写在函数文件envelope.m中。由于在拼接音符时会用到重叠部分的长度，因此函数也将*β*参量一并返回。函数代码如下：

function [y, beta\_param] = envelope(x)

alpha\_param = 0.075;

gamma\_param = 0.75;

beta\_param = 1.1;

delta\_param = 1 + (gamma\_param-alpha\_param)./(alpha\_param-beta\_param);

A = -1./(alpha\_param.^2);

B = (delta\_param-1)./(gamma\_param-alpha\_param).^2;

C = (gamma\_param-alpha\_param)./(gamma\_param-beta\_param) .\* B;

y = ...

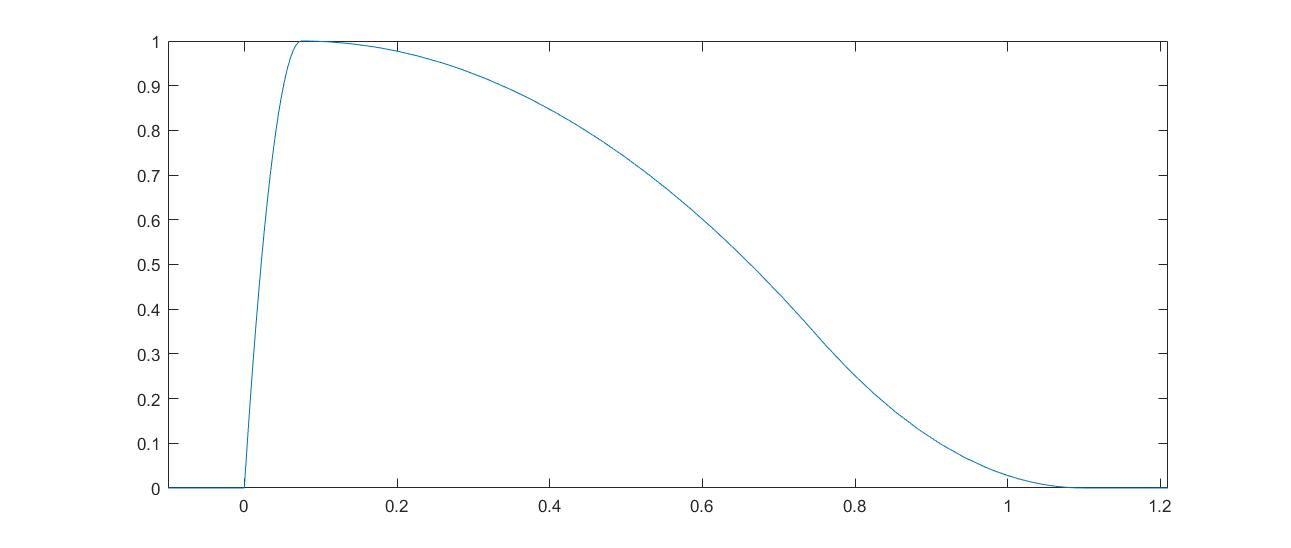
(x >= 0 & x < alpha\_param) .\* (A .\* (x-alpha\_param).^2 + 1) + ...

(x >= alpha\_param & x < gamma\_param) .\* (B .\* (x - alpha\_param).^2 + 1) + ...

(x >= gamma\_param & x < beta\_param) .\* C .\* (x - beta\_param).^2;

end

绘制出包络函数的图像如下：



将上一个实验中的单频正弦波与包络函数做乘积，然后将各个音符相互重叠地拼在一起（下一个音符在上一个音符未结束时便开始，开始位置位于包络函数自变量为1的位置）。用last\_n\_padding记录上一个音符的重叠部分的长度（采样点数），则关键代码为：

len = size(song);

len = len(1);

res = [];

[~, padding] = envelope(0);

last\_nPadding = 0;

for i = 1 : 1 : len

f = tunes(song(i, 1));

time\_len = song(i, 2) \* beat\_len;

nTime\_len = Fs \* time\_len \* padding;

t = linspace(0, time\_len \* padding - 1 / Fs, nTime\_len)';

tmp\_res = envelope(t/time\_len) .\* sin(2 \* pi \* f \* t);

if (last\_nPadding == 0)

res = [res; tmp\_res];

else

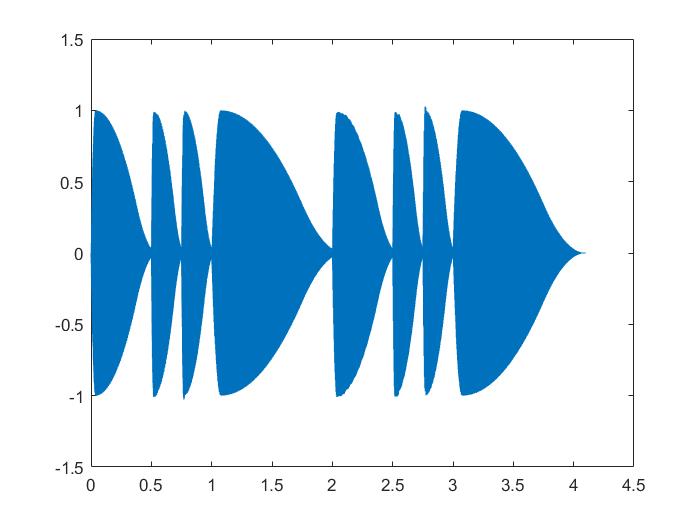
res = [res(1:end-last\_nPadding); res(end-last\_nPadding)+tmp\_res(1:last\_nPadding); tmp\_res(last\_nPadding+1:end)];

end

last\_nPadding = round(nTime\_len - Fs \* time\_len);

end

最终得到的声音波形如下：



播放音乐，可以听出原来两个音符之间“啪”的一生消失了，并且音乐有了跳跃感，给人的感觉得到了明显的改善。

本问题的完整代码位于文件hw\_1\_2\_1\_2.m内，得到的音乐文件为hw\_1\_2\_1\_2.wav。

### 音乐变调

将音乐升高八度的方法非常简单。由于两个相邻的八度之间频率相差一倍，因此只需要在播放时将采样率扩大一倍即可：

sound(res, Fs \* 2);

升高八度后的音乐文件为hw\_1\_2\_1\_3\_high.wav。同理将采样率缩小为原来的一半可以降低八度。降低八度后的音乐文件为hw\_1\_2\_1\_3\_low.wav。

如果要将音乐升高半个音阶，由于一个半个音阶的频率相差，因此可以用resample函数将采样率变为原来的，再按原采样率进行播放即可：

res\_resamp = resample(res, round(Fs ./ 2.^ (1/12)), Fs);

得到的音乐文件为hw\_1\_2\_1\_3.wav。

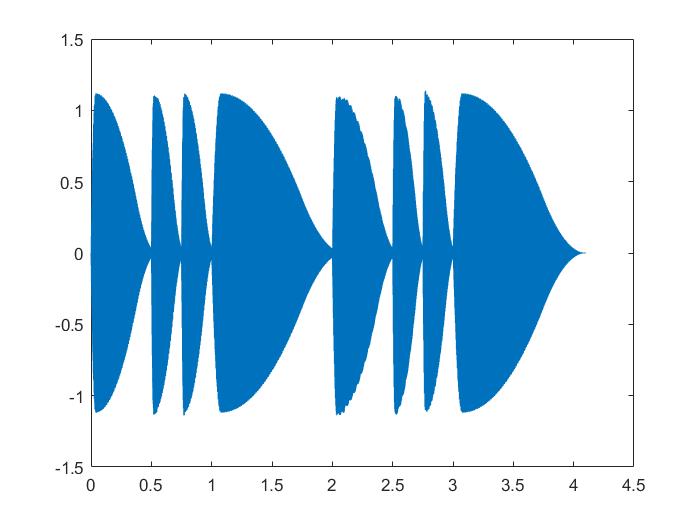
本问题的完整代码位于文件hw\_1\_2\_1\_3.m中。

### 增加谐波分量

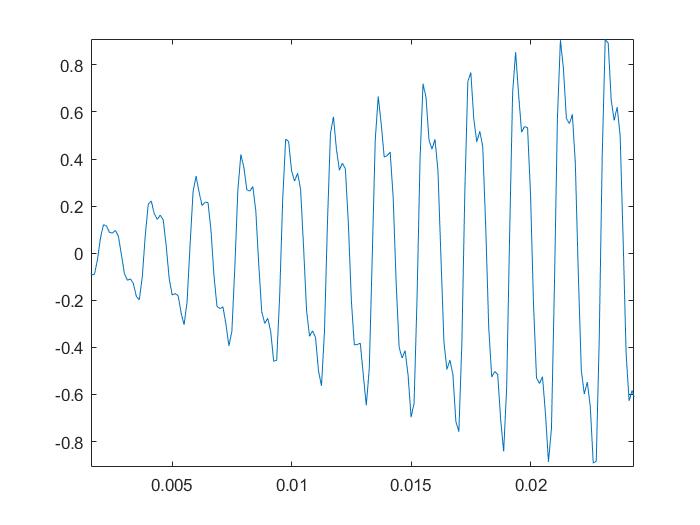
单频正弦波的声音难免让人感到有些刺耳，增加谐波分量可以改变音色，模拟出不同乐器弹奏的乐曲。增加谐波分量的方法很简单，只需要在生成单频正弦波的基础上加上一定强度的高频正弦波分量即可。设基波分量的振幅为1，取二次谐波为0.2、三次谐波为0.3，关键代码如下：

tmp\_res = envelope(t/time\_len) .\* (sin(2\*pi\*f \* t) + 0.2\*sin(2\*pi\*2\*f \* t) + 0.3\*sin(2\*pi\*3\*f \* t));

得到的声音波形：



由于绘制的时间范围远大于周期数，因此和之前的波形无法看出区别。故取前200个采样点进行绘制：



可以看到，声音的波形不再是单频正弦波。由于有高次谐波分量，因此波形相对于正弦波有一些变形。

播放音乐，可以听出音乐确实类似于风琴声。

本问题的完整代码参见hw\_1\_2\_1\_4.m，最终得到的音乐文件为hw\_1\_2\_1\_4.wav。

### 自选音乐合成