Assignment 3

Pavani Chowdary
2019112005

Question 1

Load ‘guitar.wav’ using the MIRToolbox. Design a high-pass filter such that we still hear the melody but as virtual pitch after filtering.

Loading the track ‘guitar.wav’ using MIRToolbox -

```
y = miraudio('guitar.wav')
%[y, fs] = audioread('guitar.wav');
```

Extracting pitch -

```
pitch = mirpitch(y,'Frame')
%f0 = pitch(y,fs)
```

Upon observing the pitch information, we can see that the frequency threshold would be

Designing a high-pass filter using the cut-off frequency obtained -

```
fc = min(f0)/12; % Cutoff frequency
[b, a] = highpass(y, fc, fs);

% Apply filter to audio data
y1 = filter(b, a, y);
sound(y1, fs);
```

Question 2

Feed-forward Echo effect: Design a filter that adds reverberation to the file 'guitar.wav'. In other words, add a delayed and attenuated version of the audio file to itself. Set the gain a to 0.5 and delay to 250ms

$$y(n) = x(n) + ax(n{-}D)$$

To add reverberation to the audio clip, we can add a delayed and attenuated version of itself to the audio clip, by using delay

```
a = 0.5;
d = 0.25;
sample = round(fs*d);
y1 = [zeros(sample, 1); y(1:end-sample, 1)]*a;

%adding the delayed and attenuated signal to itself
y2 = y+y1;
y2 = y2/max(abs(y2));

sound(y2, fs);
```

We can observe the echo effect in the resultant signal obtained as the filter output.

Question 3

Feedback: Design a filter that adds echo/reverberation to the file 'guitar.wav'. In other words, add a delayed and attenuated version of the audio file to itself. Use an attenuation value/gain a = 0.5. Vary delay (D) from 50ms to 500ms in steps of 50ms. At which point is the reverb noticeable? At that timepoint of noticeable reverb, change gain from 0.1 to 1 in steps of 0.1 and comment on the results.

$$y(n) = ay(n{-}D) + x(n)$$

To design a filter that adds reverberation to the given track, we can add a delayed and attenuated version of itself to the signal. Here, gain is given as a = 0.5, and delayed is to be varied from 50ms to 500ms, in steps of 50ms.

```
a = 0.5;
d = 0.05:0.05:0.5;
l = length(d)
y1 = zeros(length(y),1);
for i = 1:l
    sample = round(d(i)*fs);
    y1 = [zeros(sample, 1); y1(1:end-sample, 1)]*a;
    y2 = y + y1;
    sound(y2, fs);
    pause(1);
end
```

After listening to the resultant signals, it can be observed that reverb is noticed around 0.2s. Now varying the gain from 0,1 to 1 in steps of 0.1,

```
gain = 0.1:0.1:1;
d = 0.2;
```

Question 4

Design a resonant filter with the following specifications:
• Resonant frequency = 440Hz
• BW = 50Hz

$$y(n) = A_0x(n) + a_1y(n-1) - a_2y(n-2)$$
$$where, A_0 = (1+R^2)sin(\theta); a_1 = 2Rcos(\theta); a_2 = R2; R = 1–(BW/2)$$

Matlab script to design a resonant filter with the given specifications-

```
f0 = 440;
BW = 50;
%fs = 44100Hz
R = 1 - (BW/2);
theta = 2*pi*f0/fs;

A0=(1+R^2)*sin(theta);

a1=2*R*cos(theta);
a2=R^2;

%filter coefficients
b = [A0 0 -A0];
a = [1 -a1 -a2];

y1 = filter(b, a, y1);
```

Question 5

Create a 2-second long complex wave that is a sum of a two sine waves of frequencies 400 Hz and 1000Hz (fs = 44100hz). Design a notch filter to attenuate the 1000hz wave.

```
fs = 44100;
f0 = 1000;
bw = 50;

t = 0:1/fs:2-1/fs;
f1 = 400;
f2 = 1000;
y = sin(2*pi*f1*t) + 0.5*sin(2*pi*f2*t);

w0 = f0/(fs/2);
[b, a] = iirnotch(f0/(fs/2), bw);

y1 = filter(b, a, y);
sound(y1,fs);

plot(t, y);
plot(t, y1);
```

The filtered signal has only one frequency component (400Hz). The filter is a stable filter.