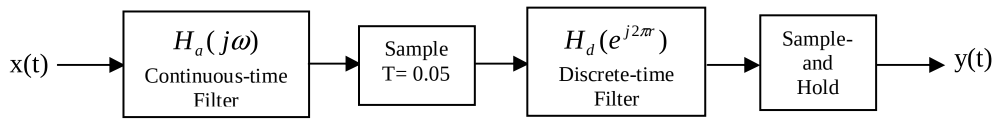
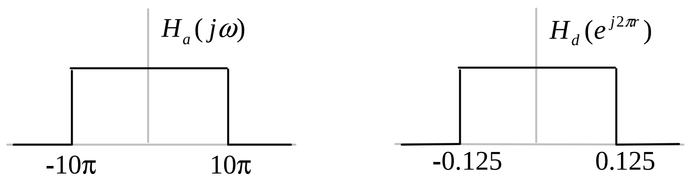
EE422G Homework #11 Solution

1. (2 points) A discrete-time system for processing continuous-time signals is shown below. Sketch the magnitude of the frequency response of an equivalent continuous-time system. Notice that the frequency response of the discrete-time filter $H_d(e^{j2\pi r})$ is written in terms of the normalized frequency r = ωT/(2π).





To find the equivalent impulse response h(t), consider the input δ(t), which corresponds to 1 in the frequency domain. After going through the first filter, the signal is band-limited to 10π. The sampling frequency is ω_s = 2π /0.05 = 40π, which is higher than the Nyquist rate of 20 π. Thus, no aliasing occurs and the spectrum repeats itself every 40 π. The cut-off frequency of the discrete filter is 0.125*ω_s = 5 π. Since it is a discrete filter, its spectrum also repeats every 40π. The resulting spectrum is a rectangular spectrum with central component between -5π and 5π, repeating every 40π. Finally the frequency spectrum of a sample and hold is a sinc function. Multiplying by our rectangular pulse results in a partial "sinc" function, non-zero only between ±40nπ + [-5π, 5π] for all integer n.

2. (2 points) A discrete-time system is defined by

$$y(nT) = x(nT) - a_1 y(nT - T) - a_2 y(nT - 2T)$$

Determine a_1 and a_2 such that the frequency response at ω=0 is 1.0 and the frequency response at ω=0.5ω_s is 0.1, where ω_s is the sampling frequency.

$$y(nT) = x(nT) - a_1 y(nT - T) - a_2 y(nT - 2T)$$

Transform the difference equation into Z-domain, we have

$$Y(z) = X(z) - a_1 z^{-1} Y(z) - a_2 z^{-2} Y(z)$$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2}}$$

We substitute $z=e^{j\omega T}$ to evaluate the DTFT

$$H(e^{j\omega T}) = \frac{1}{1 + a_1 e^{-j\omega T} + a_2 e^{-2j\omega T}}$$

At ω=0,

$$H(e^{j0T}) = \frac{1}{1 + a_1 + a_2} = 1.0 \Rightarrow a_1 + a_2 = 0$$

At ω=0.5ω_s = π/T

$$H(e^{j\pi T}) = \frac{1}{1 - a_1 + a_2} = 0.1 \Rightarrow -a_1 + a_2 = 9$$

Combining the two equations, we get a_1=-4.5, a_2=4.5.

3. (1 points) Show that the following filter has linear phase:

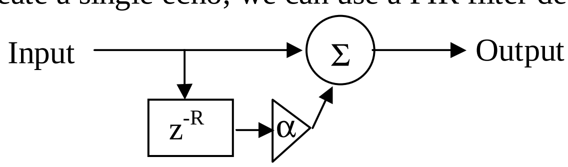
$$H(z) = a_0(z^0 + z^{-2N}) + a_1(z^{-1} + z^{-2N+1}) + \ldots + a_{N-1}(z^{-N+1} + z^{-N-1}) + a_N z^{-N}$$

Notice that this is a symmetric impulse response as we have discussed in class. We substitute $z=e^{j\omega T}$ to evaluate the DTFT:

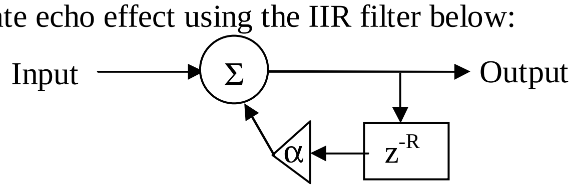
$$H(e^{j\omega T}) = a_0(e^0 + e^{-j2N\omega T}) + a_1(e^{-j\omega T} + e^{-j(2N-1)\omega T}) + \ldots + a_{N-1}(e^{-j(N-1)\omega T} + e^{-j(N+1)\omega T}) + a_N e^{-jN\omega T}$$
$$= e^{-jN\omega T}\left[a_0(e^{jN\omega T} + e^{-jN\omega T}) + a_1(e^{j(N-1)\omega T} + e^{-j(N-1)\omega T}) + \ldots + a_{N-1}(e^{j\omega T} + e^{-j\omega T}) + a_N\right]$$

Recall that the conjugate of $e^{j\theta}$ is $e^{-j\theta}$. Thus, the expression inside the square bracket is real. Hence, the overall phase of $H(e^{jwT})$ is $-N\omega T$, which is linear.

4. (5 points) Digital Audio Effect: In this exercise, we will use MATLAB to produce echo effect using both FIR and IIR filters. Download the audio file "`singing.wav`" from the course homework website. The sampling frequency of this recording is 44.1 kHz. To create a single echo, we can use a FIR filter described below:



a. Implement the above FIR filter, choosing an R such that it corresponds to a delay of 0.3 second and select an α such that the dynamic range of the output stays within -1 and 1.
b. We can also create echo effect using the IIR filter below:



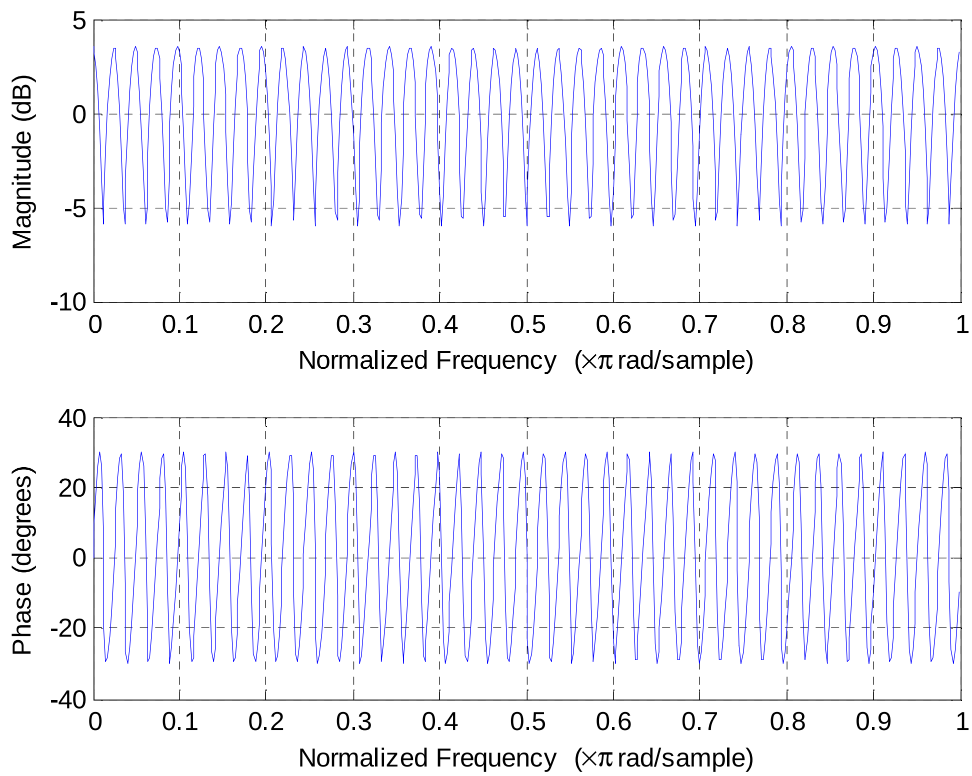
Implement this filter using the same R as in part a) and select an α such that filter is stable and the output dynamic range stays within -1 and 1. Explain the perceptual difference between the two filters.

For both part a) and b), turn in your MATLAB code as well as the magnitude and phase response of the filters.

We need to implement the transfer function: $H(z) = 1 + \alpha z^{-r}$

```
>> [x,fs] = wavread('singing.wav');     % fs stores the sampling frequency
>> r = round(fs*0.3);                   % Corresponding to 0.3 second delay
>> b = zeros(1,r+1);
>> b(1) = 1;                            % Impulse response
>> alpha = 0.5;
>> b(r+1) = alpha;
>> y = filter(b,1,x);                   % FIR filtering
```

```
>> max(y)                                % Check dynamic range
ans =
   0.6763
>> min(y)
ans =
  -0.9398
>> wavplay(y,fs);                        % You should hear one echo
>> freqz(b,1)
```



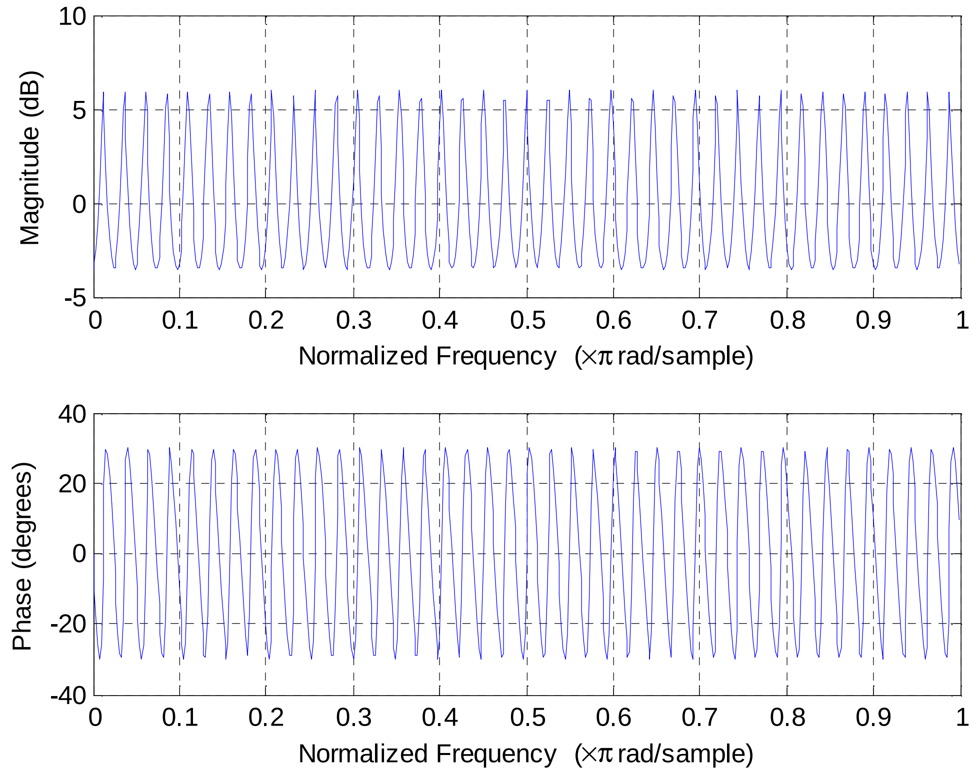
The oscillating spectrum is due to the zeros on the unit circle (convince yourself that it is the case). Due to the shape of the frequency spectrum, this type of filter is called a FIR comb filter.

Next we build a IIR filter: $H(z) = 1/(1+\alpha z^{-r})$
Using long division, it is easy to see that $H(z) = 1+\alpha z^{-r}+\alpha^2 z^{-2r}+\alpha^3 z^{-3r}+\ldots$
Thus, $Y(z) = H(z)X(z) = X(z)+\alpha z^{-r}X(z)+\alpha^2 z^{-2r}X(z)+\alpha^3 z^{-3r}X(z)+\ldots.$

Provided that $|\alpha|<1$, $Y(z)$ is a summation of infinite number of decaying echos!

```
>> y = filter(1,b,x);
>> wavplay(y,fs)
>> freqz(1,b)
```

This filter is also called a IIR comb filter, which is quite similar to that of the FIR.


10
5
0
-5
Magnitude (dB)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)
40
20
0
-20
-40
Phase (degrees)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)