Mason Nixon
ELEC 5410
Dr. Stanley Reeves
September 13, 2010

Project #2: Difference Equations

1. Let y[n] – 0.9y[n - 1] + 0.7y[n - 2] = x[n] + 0.5x[n - 1] + 0.3x[n - 2]. This difference equation can be implemented using the *filter* command. Find the vectors a and b that represent the difference equation above for the *filter* command.

By using the help file for filter in MATLAB and comparing the standard form of the difference equation [a(1)*y(n) = b(1)*x(n) + b(2)*x(n-1) + ... + b(nb+1)*x(n-nb) - a(2)*y(n-1) - ... - a(na+1)*y(n-na)], the vectors a and b are:

```
a=[1,-0.9,0.7];
b=[1,0.5,0.3];
```

2. Calculate h[n] analytically for the difference equation above. Your answer should be a functional expression. Hint: You may find the *residuez* function useful. Note that the initial expression may come out with complex components. However, it is possible to simplify it into an expression that is real.

```
[r p k] = residuez(b,a);
```

MATLAB Output:

```
-----------
     r =  0.2857 - 0.8101i
        0.2857 + 0.8101i
     p =  0.4500 + 0.7053i
        0.4500 - 0.7053i
     k =  0.4286
-----------
```

$$y[n] – 0.9y[n-1] + 0.7y[n-2] = x[n] + 0.5x[n-1] + 0.3x[n-2]$$

$$Y\left(e^{-jw}\right) - 0.9e^{-jw}Y\left(e^{-jw}\right) + 0.7e^{-j2w}Y\left(e^{-jw}\right) = X\left(e^{-jw}\right) + 0.5e^{-jw}X\left(e^{-jw}\right) + 0.3e^{-j2w}X\left(e^{-jw}\right)$$

$$H\left(e^{-jw}\right) = \frac{Y\left(e^{-jw}\right)}{X(e^{-jw})} = \frac{\left[1 + 0.5\left(e^{-jw}\right) + 0.3\left(e^{-j2w}\right)\right]}{[1 - 0.9(e^{-jw}) + 0.7(e^{-j2w})]}$$

$$H\left(e^{-jw}\right) = K + \frac{r_1}{1 - p_1 e^{-jw}} + \frac{r_2}{1 - p_2 e^{-jw}}$$

From MATLAB’s residuez:

$$H\left(e^{-jw}\right) = 0.4286 + \frac{0.2857 - j0.8101}{1 - (0.45 + j0.7053)e^{-jw}} + \frac{0.2857 + j0.8101}{1 - (0.45 - j0.7053)e^{-jw}}$$

$$\begin{aligned} h[n] = 0.4286\ \delta[n] &+ (0.2857 - j0.8101)(0.45 + j0.7053)^n \mu[n] \\ &+ (0.2857\ + j0.8101)(0.45 - j0.7053)^n\ \mu[n] \end{aligned}$$

$$h[n] = 0.4286\,\delta[n] + [(0.2857 - j0.8101)(0.45 + j0.7053)^{n} \\ + (0.2857\ + j0.8101)(0.45 - j0.7053)^{n}]\,\mu[n]$$

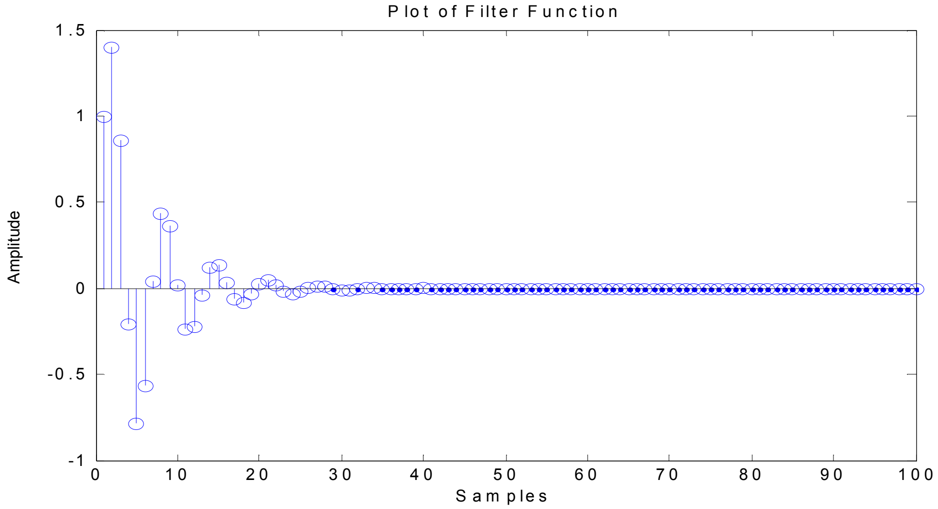
$$h[n] = 0.4286\,\delta[n] + [0.7186e^{-j(70.6-57.5n)} + 0.7186e^{j(70.6-57.5n)}]\,\mu[n]$$

$$\text{Note: } e^{-ja} + e^{ja} = 2\cos(a),$$

$$\boxed{h[n] = 0.4286\,\delta[n] + [1.43\cos(70.6 - 57.5n)]\mu[n]}$$

3. Create an impulse (not a pulse!) of length 100. Recall that systems described by linear constant-coefficient difference equations are LSI systems (assuming initial rest conditions). Characterize the LSI system above by finding the first 100 points of the impulse response using filter, and plot the result with stem.

```
--------------------------------
vector=[1,zeros(1,99)];
a=[1,-0.9,0.7];
b=[1,0.5,0.3];
y=filter(b,a,vector);
stem(y)
xlabel('Samples')
ylabel('Amplitude')
title('Plot of Filter Function')
---------------------------------
```



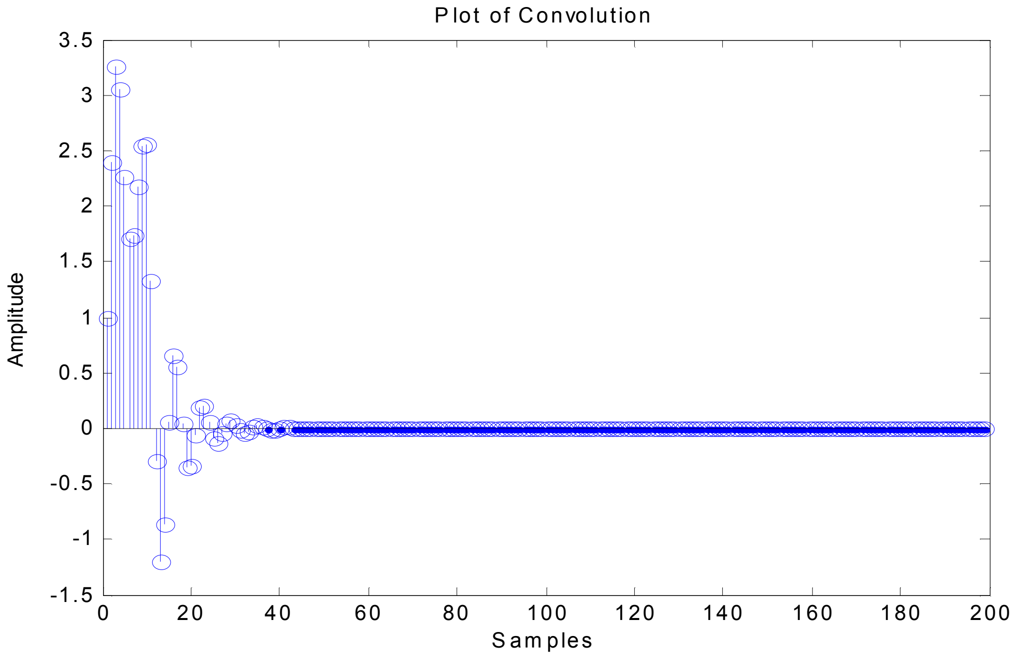
4. Examine two ways of implementing an LSI system:

(a) Create a pulse of width 10 and zeropadded to a total length of 100.

```
pulse  = [ones(1,10),zeros(1,(100-10))];
```

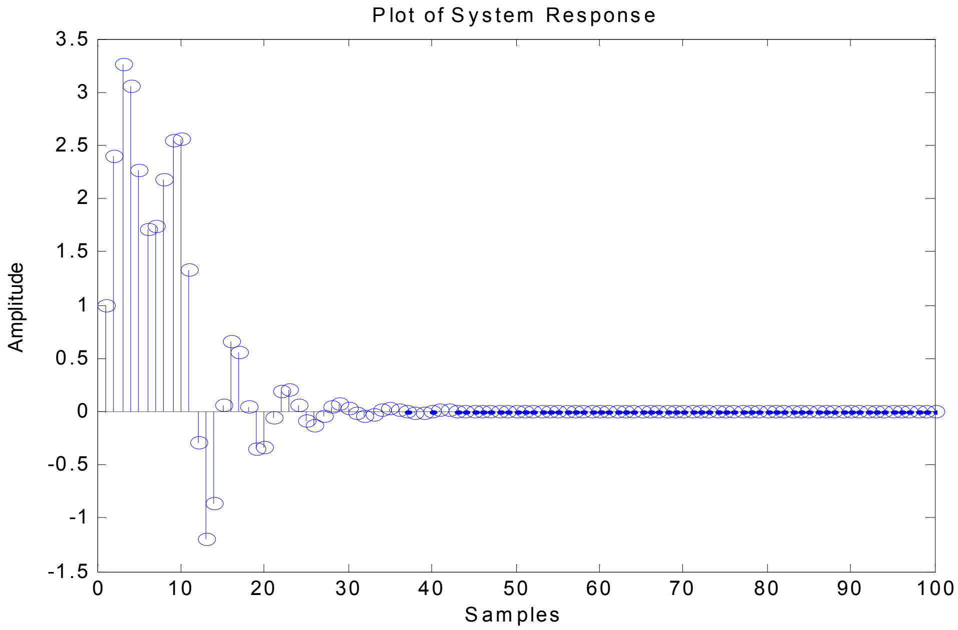
(b) Find the response of the system to this input pulse using *conv* by convolving with the impulse response.

```
----------------
   y=filter(b,a,vector);
   resp = conv(y,pulse);
   stem(resp)
   xlabel('Samples')
   ylabel('Amplitude')
   title('Plot of Convolution')
----------------
```



(c) Find the response of the system to this input pulse using *filter* by filtering with the difference equation.

```
----------------
y=filter(b,a,pulse);
stem(y)
----------------
```



(d) Explain any differences you observe between these two results.

The convolution plot is generated by convolving the 10 pulses and 90 zeros with the impulse response that was created using MATLAB's *conv* function. Since the convolution superimposes one plot onto another, the resultant plot extends for 199 data points and the last 99 converge to zero.

The system response plot generated using the *filter* function with inputs a, b, and the pulse vector which implements the difference equation that describes the system. Using this method there are only the 10 non-zero inputs into the system and so the additional zeros are not present and the resultant system response plot contains 100 data points which are identical to the first 100 points on the convolution plot.

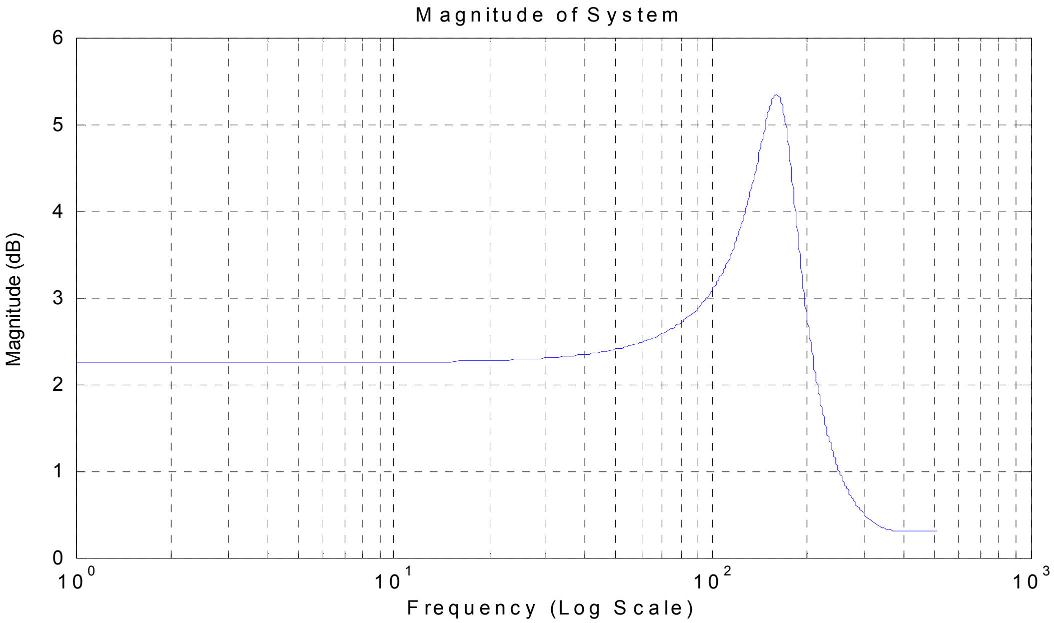
5. Examine the frequency response:

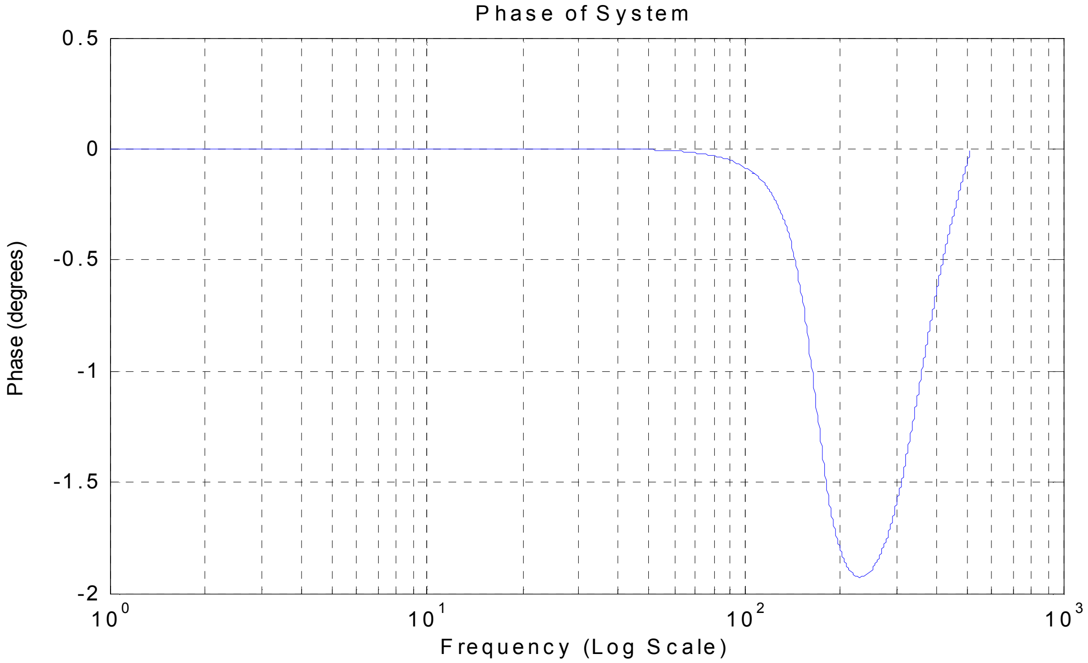
(a) Find an expression for the frequency response of the system described by the difference equation given.

$$H\left(e^{-jw}\right) = \frac{Y\left(e^{-jw}\right)}{X\left(e^{-jw}\right)} = \frac{\left[1 + 0.5\left(e^{-jw}\right) + 0.3\left(e^{-j2w}\right)\right]}{\left[1 - 0.9\left(e^{-jw}\right) + 0.7\left(e^{-j2w}\right)\right]}$$

(b) Use the command *freqz* to plot the magnitude and phase response of the system.

```
----------------------
[h,w] = freqz(b,a);
H=abs(h);
W=angle(h);
figure;
semilogx(H);
figure;
semilogx(W);
----------------------
```





(c) Is this system more of a highpass, a lowpass, or a bandpass filter? Explain your answer.
Since the magnitude plot indicates that only low frequencies are allowed to pass and other frequencies are attenuated, the system seems to be a lowpass filter with a very high ripple and sharp roll off.

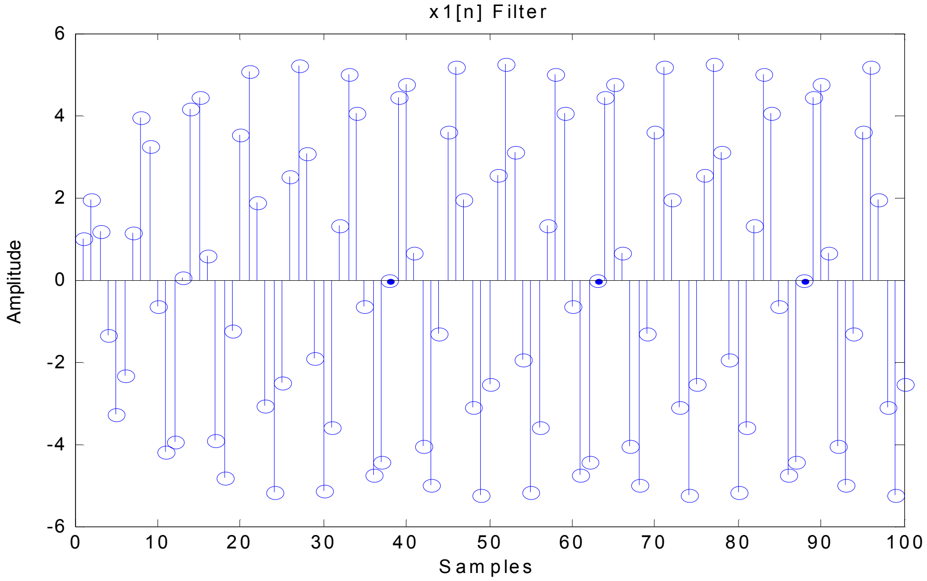
6. Examine the response to two sine waves:

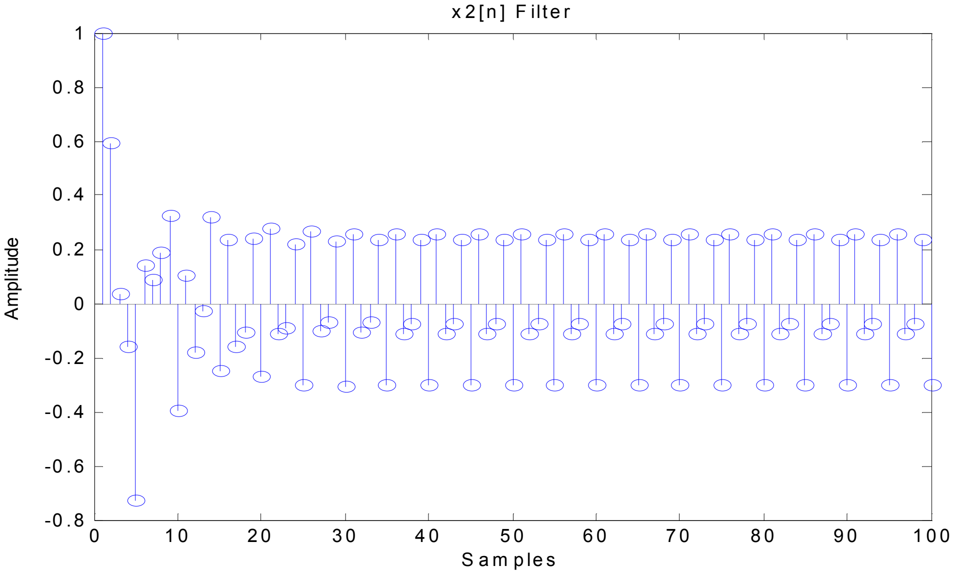
(a) Create two signals x1[n] = cos(0.32πn) and x2[n] = cos(0.8πn), both of length 100.

```
x1 = cos(0.32*pi*(0:99));
x2 = cos(0.8*pi*(0:99));
```

(b) Filter each signal separately with the filter defined in Step 1, and plot them.

```
----------------
y1 = filter(b,a,x1);
y2 = filter(b,a,x2);
stem(y1)
figure
stem(y2)
----------------
```



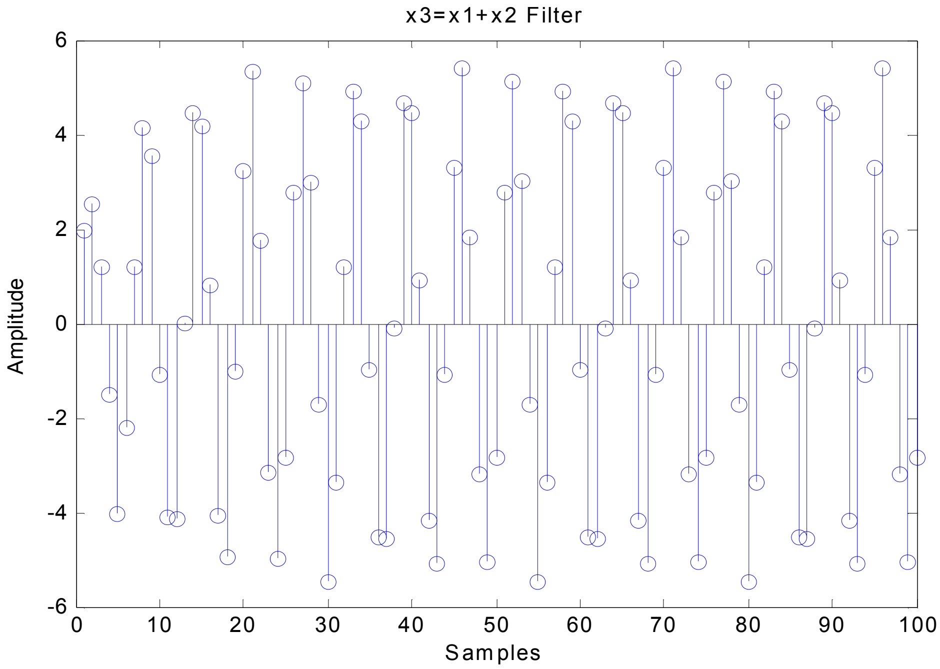


(c) Explain the outputs you observe in terms of the frequency magnitude response of the system.

The magnitude of $x_1[n]$'s response is greater than $x_2[n]$'s. Also a cosine pattern is present in the output.

(d) Now add the two signals together and filter the sum of the two. Explain the output in light of your previous observations.

```
----------------
x3 = x1 + x2;
y3 = filter(b,a,x3);
stem(y3)
----------------
```



The filter of the sum of the x_1 and x_2 signals is equal to the sum of their respective filters, i.e. each sample in the y_3 plot above is the sum of the samples in each of the y_1 and y_2 filters.

Supplement: M-file

```
% Mason Nixon
% Assignment 2 in DSP Fall 2010

a=[1,-0.9,0.7];
b=[1,0.5,0.3];

[r p k] = residuez(b,a);
vector=[1,zeros(1,99)];
y=filter(b,a,vector);
stem(y)
xlabel('Samples')
ylabel('Amplitude')
title('Plot of Filter Function')
%%
pulse  = [ones(1,10),zeros(1,(100-10))];
y=filter(b,a,vector);
resp = conv(y,pulse);
figure
stem(resp)
xlabel('Samples')
ylabel('Amplitude')
title('Plot of Convolution')

y=filter(b,a,pulse);
figure
stem(y)
xlabel('Samples')
ylabel('Amplitude')
title('Plot of System Response')
%%
a=[1,-0.9,0.7];
b=[1,0.5,0.3];
[h,w] = freqz(b,a);
H=abs(h);
W=angle(h);
figure;
semilogx(H);
figure;
semilogx(W);
%%
y1 = filter(b,a,x1);
y2 = filter(b,a,x2);
stem(y1)
title('x1[n] Filter');
ylabel('Amplitude')
xlabel('Samples')
figure
stem(y2)
title('x2[n] Filter');
ylabel('Amplitude')
xlabel('Samples')

x3 = x1 + x2;
y3 = filter(b,a,x3);
figure
stem(y3)
title('x3=x1+x2 Filter');
ylabel('Amplitude')
xlabel('Samples')
```