

EE 313 Linear Signals & Systems (Fall 2018)

Solution Set for Homework #5 on Time and Frequency Responses of FIR Filters

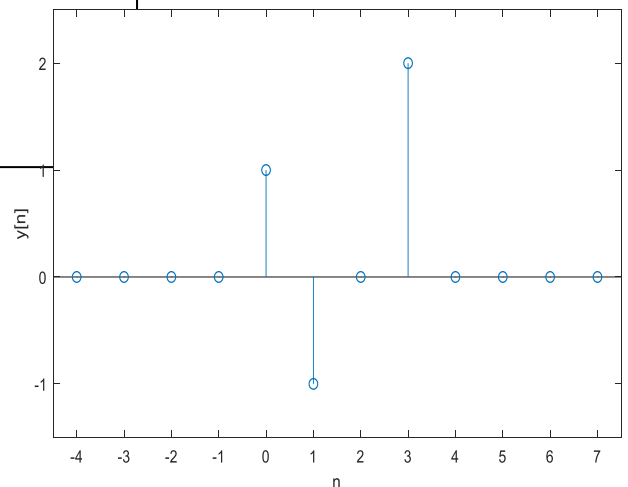
By: Houshang Salimian & Prof. Brian L. Evans

1. Prologue: Company X has developed a digital hardware (or software) module to sell to other companies to include in their designs. At the same time, Company X would also like to protect the intellectual property contained in the module. Company X decides to develop a linear time-invariant (LTI) model for the input-output relationship for their module to help other companies include the module in their designs. Companies can then use the LTI module in simulations of the overall system that includes the module from Company X.

Input $x[n]$	Output $y[n]$
$\delta[n] - \delta[n-1]$	$\delta[n] - \delta[n-1] + 2\delta[n-3]$
$\cos(\pi n/2)$	$2\cos(\pi n/2 - \pi/4)$

Solution: (a) To plot $y[n] = \delta[n] - \delta[n-1] + 2\delta[n-3]$, we'll need to define $\delta[n]$ in MATLAB where n is a discrete-time index. One could use `rectpuls(n)` or `tripuls(n)`.

```
n = -4:7;
y = tripuls(n) - tripuls(n-1) + 2*tripuls(n-3);
stem(n, y);
xlim([-4.5 7.5])
ylim([-1.5 2.5])
xlabel('n')
ylabel('y[n]')
```



(b) If $x_1[n] = \delta[n] - \delta[n - 1]$ then

$$y_1[n] = \delta[n] - \delta[n - 1] + 2\delta[n - 3]$$



Using time-invariant property of the system,

$$x_2[n] = x_1[n - 1] = \delta[n - 1] - \delta[n - 2]$$

$$y_2[n] = y_1[n - 1]$$

$$y_2[n] = \delta[n - 1] - \delta[n - 2] + 2\delta[n - 4]$$

Therefore,

$$x[n] = 7x_1[n] + 7x_2[n] = 7x_1[n] + 7x_1[n - 1] = 7\delta[n] - 7\delta[n - 2]$$

Using the linear property of LTI systems.

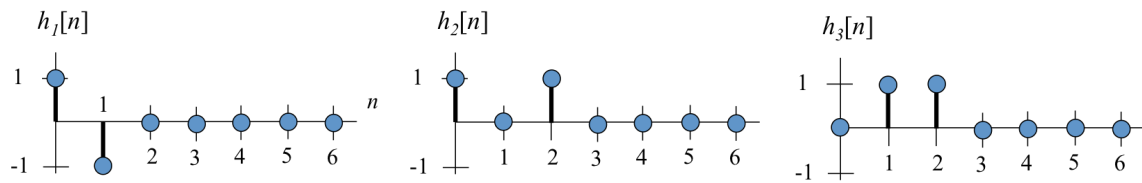
$$x[n] = 7x_1[n] + 7x_2[n] \rightarrow y[n] = 7y_1[n] + 7y_2[n]$$

$$y[n] = 7y_1[n] + 7y_2[n] = 7\delta[n] - 7\delta[n - 2] + 14\delta[n - 3] + 14\delta[n - 4]$$

2. Prologue: The question concerns associativity and commutativity properties of the convolution sum. This question introduces a cascade, which is used to build complex blocks from simpler ones. The first system is a first-order difference filter.

Solution: (a) For any system, $h[n] = y[n]$ when $x[n] = \delta[n]$.

$$\text{So, } h_1[n] = \delta[n] - \delta[n - 1], h_2[n] = \delta[n] + \delta[n - 2], h_3[n] = \delta[n - 1] + \delta[n - 2]$$



(b) $h[n] = h_1[n] * h_2[n] * h_3[n]$

First calculate $h_1[n] * h_2[n]$. We'll first compute the answer graphically. We can flip $h_2[n]$ and slide it across $h_1[n]$. We get 1, -1, 1, -1 for $n = 0, 1, 2,$ and 3 , which has a length of 4 samples (length of $h_1[n]$ + length of $h_2[n]$ - 1). We can then flip and slide $h_3[n]$ across the result for $h_1[n] * h_2[n]$ to obtain 0, 1, 0, 0, 0, -1 for $n = 0, 1, 2, 3, 4,$ and 5 , which has a length of 5 samples.

Or, we could write out everything using mathematical formulas:

$$\begin{aligned}
 h_1[n] * h_2[n] &= \sum h_1[k]h_2[n - k] = \sum (\delta[k] - \delta[k - 1])(\delta[n - k] + \delta[n - k - 2]) \\
 &= \sum \delta[k](\delta[n - k] + \delta[n - k - 2]) - \sum \delta[k - 1](\delta[n - k] + \delta[n - k - 2]) \\
 &= \delta[n] + \delta[n - 2] - \delta[n - 1] - \delta[n - 3] \\
 h[n] &= (h_1[n] * h_2[n]) * h_3[n] \\
 &= \sum (\delta[k] - \delta[k - 1] + \delta[k - 2] - \delta[k - 3])(\delta[n - k - 1] + \delta[n - k - 2]) \\
 &= \sum (\delta[k])(\delta[n - k - 1] + \delta[n - k - 2]) \\
 &\quad - \sum (\delta[k - 1])(\delta[n - k - 1] + \delta[n - k - 2]) \\
 &\quad + \sum (\delta[k - 2])(\delta[n - k - 1] + \delta[n - k - 2]) \\
 &\quad - \sum (\delta[k - 3])(\delta[n - k - 1] + \delta[n - k - 2]) \\
 &= \delta[n - 1] + \delta[n - 2] - \delta[n - 2] - \delta[n - 3] + \delta[n - 3] + \delta[n - 4] - \delta[n - 4] \\
 &\quad - \delta[n - 5] \\
 &= \delta[n - 1] - \delta[n - 5]
 \end{aligned}$$

(c) $h[n] = \delta[n - 1] - \delta[n - 5]$ which gives

$$y[n] = h[n] * x[n] = \sum_{k=1}^5 h[k]x[n - k] = x[n - 1] - x[n - 5]$$

Epilogue: Given what we know about the convolution sum properties, do you think that re-ordering the filter blocks would change the output? Would your answer change if we fed back the final output and added it to the input to the first block in the cascade?

3. Prologue: This problem revisits lowpass filters that you have had seen on several occasions, including homework assignments, tune-ups, and lecture. This problem provides more detail into the discrete-time Fourier transform, and asks you to calculate it analytically.

Solution: (a) $y[n] = 2x[n] - 3x[n - 1] + 2x[n - 2]$

$$h[n] = 2\delta[n] - 3\delta[n - 1] + 2\delta[n - 2]$$

The discrete-time Fourier transform of the impulse response gives the frequency response of the filter:

$$\begin{aligned}
 H(e^{j\hat{\omega}}) &= \sum_{n=0}^M h[n]e^{-j\hat{\omega}n} = \sum_{n=0}^M (2\delta[n] - 3\delta[n-1] + 2\delta[n-2])e^{-j\hat{\omega}n} \\
 &= 2 \sum_{n=0}^M \delta[n]e^{-j\hat{\omega}n} - 3 \sum_{n=0}^M \delta[n-1]e^{-j\hat{\omega}n} + 2 \sum_{n=0}^M \delta[n-2]e^{-j\hat{\omega}n} \\
 &= 2 - 3e^{-j\hat{\omega}} + 2e^{-j2\hat{\omega}}
 \end{aligned}$$

$$H(e^{j\hat{\omega}}) = 2 - 3e^{-j\hat{\omega}} + 2e^{-j2\hat{\omega}} = e^{-j\hat{\omega}}(2e^{j\hat{\omega}} - 3 + 2e^{-j\hat{\omega}}) = (4 \cos \hat{\omega} - 3)e^{-j\hat{\omega}}$$

The equation is in the form of a real-valued amplitude function $4 \cos \hat{\omega} - 3$ times a phase function $e^{-j\hat{\omega}}$. Since the factor $(4 \cos \hat{\omega} - 3)$ is not positive for all $\hat{\omega}$, it follows that:

$$4 \cos \hat{\omega} - 3 \geq 0 \rightarrow H(e^{j\hat{\omega}}) = (4 \cos \hat{\omega} - 3)e^{-j\hat{\omega}}$$

$$4 \cos \hat{\omega} - 3 < 0 \rightarrow H(e^{j\hat{\omega}}) = (3 - 4 \cos \hat{\omega})e^{-j(\hat{\omega}+\pi)}$$

$$4 \cos \hat{\omega} - 3 = 0 \Rightarrow \hat{\omega} = \cos^{-1}\left(\frac{3}{4}\right) = \pm 0.23\pi$$

For $-\pi < \hat{\omega} < \pi$, amplitude and angle of $H(e^{j\hat{\omega}})$ can be written as:

$$|H(e^{j\hat{\omega}})| = |4 \cos \hat{\omega} - 3| = \begin{cases} 4 \cos \hat{\omega} - 3, & -0.23\pi \leq \hat{\omega} \leq 0.23\pi \\ 3 - 4 \cos \hat{\omega}, & -\pi < \hat{\omega} < -0.23\pi \text{ or } 0.23\pi < \hat{\omega} < \pi \end{cases}$$

$$\angle H(e^{j\hat{\omega}}) = \begin{cases} -\hat{\omega}, & -0.23\pi \leq \hat{\omega} \leq 0.23\pi \\ -\hat{\omega} - \pi, & -\pi < \hat{\omega} < -0.23\pi \text{ or } 0.23\pi < \hat{\omega} < \pi \end{cases}$$

(b) The period of $\cos \hat{\omega}$ is equal to $2\pi \rightarrow$ the period of $H(e^{j\hat{\omega}})$ is equal to 2π

(c) The MATLAB code for this part is below. The magnitude is plotted in linear units.

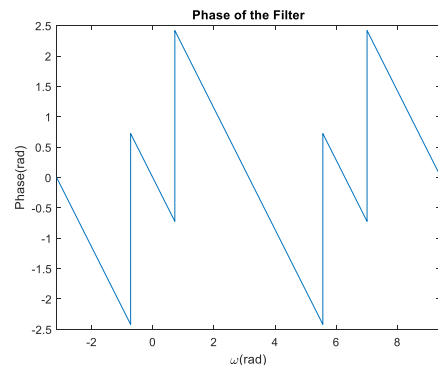
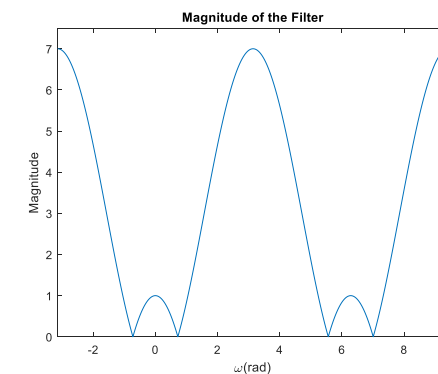
```

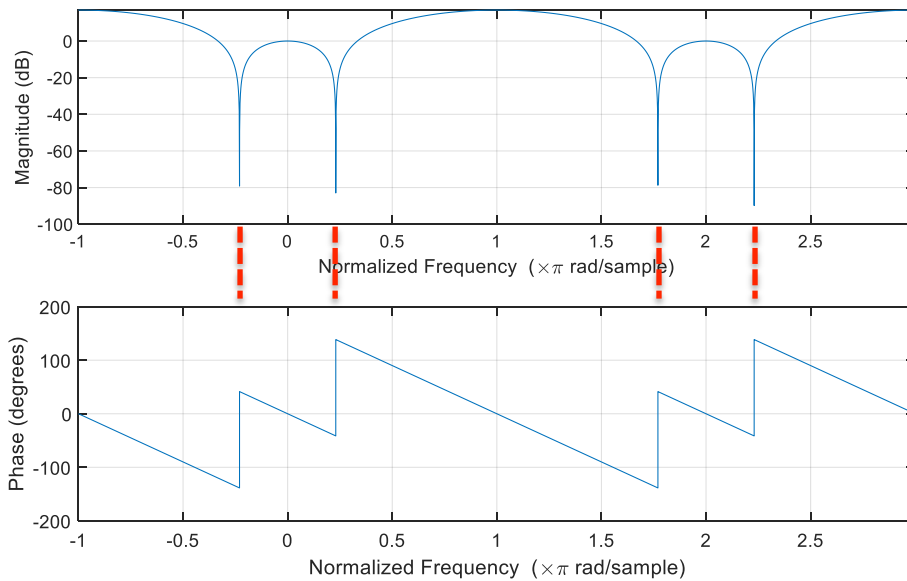
dw = 1/10000;
w = -pi : dw : 3*pi;
H = (4*cos(w)-3).*exp(-j*w);
A = abs(H);
phi = angle(H);
figure
plot(w,A)
xlim([-pi 3*pi])
ylim([0 7.5])
title("Magnitude of the Filter")
xlabel("\omega (rad)")
ylabel("Magnitude")
figure
plot(w,phi)
xlim([-pi 3*pi])
title("Phase of the Filter")
xlabel("\omega (rad)")
ylabel("Phase (rad)")
    
```

Following is the code that uses freqz to plot the magnitude and the phase of filter.

```

w = -pi:1/10000:3*pi;
h = [2, -3, 2];
freqz(h,1,w)
    
```





Magnitude response is highpass— the filter amplifies high frequency components by 20 dB more than low frequency components. f_{reqz} shows magnitude in decibels (dB): $A_{dB} = 20 \log_{10} |A|$.

Phase response is called linear. In the phase response, each discontinuity represents a change in value of 180 degrees or π radians. This corresponds to a sign change in the amplitude function since $\exp(j\pi) = -1$ and $\exp(-j\pi) = -1$. The frequencies at which the discontinuities occur, as highlighted by the **red dashed lines**, do not pass through the filter—the magnitude response is zero at those frequencies which is $20 \log_{10}(0) = -\infty$ in dB.

(d) Output is zero when $H(e^{j\hat{\omega}}) = 0$. When $4 \cos \hat{\omega} - 3 = 0$, $\hat{\omega} = 2k\pi \pm 0.23\pi$, where k is an integer.

(e)

$$\text{for } \hat{\omega} = \frac{\pi}{13} \rightarrow H\left(e^{j\frac{\pi}{13}}\right) = e^{-j\frac{\pi}{13}} \left(4 \cos\left(\frac{\pi}{13}\right) - 3\right) = 0.8838 e^{j\frac{\pi}{13}}$$

$$\text{for } \hat{\omega} = \frac{\pi}{13} \rightarrow A = 0.8838, \quad \phi = -\frac{\pi}{13}$$

$$x[n] = \sin\left(\frac{\pi n}{13}\right) = \cos\left(\frac{\pi n}{13} - \frac{\pi}{2}\right)$$

$$y[n] = 0.8838 \cos\left(\frac{\pi n}{13} - \frac{\pi}{2} - \frac{\pi}{13}\right) = 0.8838 \cos\left(\frac{\pi n}{13} - \frac{15\pi}{26}\right)$$

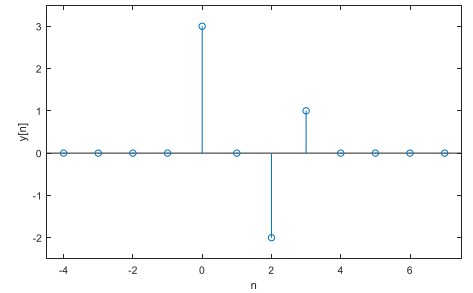
Second solution:

$$x[n] = \sin\left(\frac{\pi n}{13}\right) = \frac{1}{2j} \left(e^{j\frac{\pi n}{13}} - e^{-j\frac{\pi n}{13}} \right)$$

$$y[n] = \frac{1}{2j} \left(H(e^{j\frac{\pi}{13}}) e^{j\frac{\pi n}{13}} - H(e^{-j\frac{\pi}{13}}) e^{-j\frac{\pi n}{13}} \right) = 0.8838 \sin\left(\frac{\pi n}{13} - \frac{\pi}{13}\right) = 0.8838 \cos\left(\frac{\pi n}{13} - \frac{15\pi}{26}\right)$$

4. Solution: (a) tripulse(n) is used as $\delta[n]$ in the following code:

```
n = -4:7;
y = 3*tripuls(n) - 2*tripuls(n-2) + tripuls(n-3);
stem(n,y);
xlim([-4.5 7.5]);
ylim([-2.5 3.5]);
xlabel('n');
ylabel('y[n]');
```



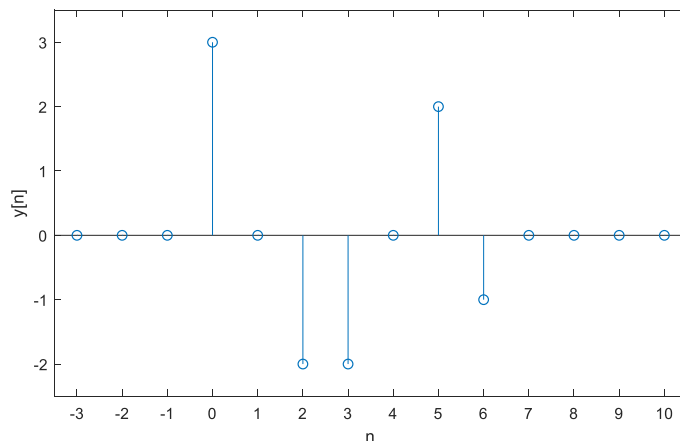
(b) Since the system is LTI, we can use the output for $x[n] = \delta[n]$, to derive the output for given input.

$$x_1[n] = \delta[n] \rightarrow y_1[n] = \delta[n] - \delta[n-3]$$

$$x[n] = 3\delta[n] - 2\delta[n-2] + \delta[n-3] = 3x_1[n] - 2x_1[n-2] + x_1[n-3]$$

$$y[n] = 3y_1[n] - 2y_1[n-2] + y_1[n-3] = 3(\delta[n] - \delta[n-3]) - 2(\delta[n-2] - \delta[n-5]) + (\delta[n-3] - \delta[n-6])$$

$$y[n] = 3\delta[n] - 2\delta[n-2] - 2\delta[n-3] + 2\delta[n-5] - \delta[n-6]$$



(c) When $x[n] = \cos\left(\frac{\pi n}{3} + \frac{\pi}{2}\right)$ the output is $y[n] = 2\cos\left(\frac{\pi n}{3} + \frac{\pi}{2}\right)$

Therefore, for $\hat{\omega} = \frac{\pi}{3}$, $H\left(e^{j\frac{\pi}{3}}\right) = 2$, and its phase is equal to zero,

$$x[n] = \cos\left(\frac{\pi(n-3)}{3}\right) \rightarrow y[n] = 2\cos\left(\frac{\pi(n-3)}{3}\right) = 2\cos\left(\frac{\pi n}{3} - \pi\right)$$

(d) for $x[n] = \delta[n] \rightarrow y[n] = h[n] = \delta[n] - \delta[n-3]$

$$h[n] = \delta[n] - \delta[n-3] \text{ which means } H\left(e^{j\hat{\omega}}\right) = 1 - e^{-j3\hat{\omega}} \text{ and hence } H\left(e^{j\frac{\pi}{2}}\right) = 1 - e^{-j\frac{3\pi}{2}} = 1 + j$$

The statement is false.