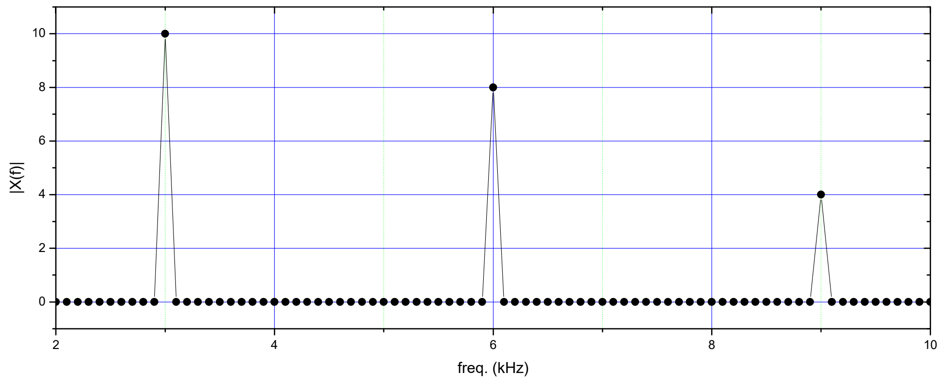
Multimedia Signal Processing 1st Module

10 /2/2016

Ex.1 (Pt.13)

A continuous real signal has the following spectrum:



We sample the signal at 24 kHz and we want to build a filter $H(z)$ to completely remove the carrier at 6kHz and keep unaltered the component at 3kHz.

a) Define the z-transform of the filter and draw its zero-pole plot.
b) Find the phases and amplitudes of the two output frequencies at 3kHz and 9 kHz.
c) What should be done in order to resample the signal to 18 ksamples/s?

Hint: Since the signal is real it will also have a counterpart at negative frequencies.

Ex.2 (Pt.9)

A random white noise is filtered with a second order pure IIR filter (no zeros outside from the origin). The output signal presents the following autocorrelation values: $r_0 = 1, r_1 = 0.5, r_2 = 0.2$

Find the coefficients of the filter and the power of the input noise.

Provide its z transform

Ex.3 (Pt. 11 – MATLAB code)

```
1) Generate five cosine tones with  the following parameters:
       Amplitude    Frequency [Hz]  Phase [deg]
  x1   1.0          200             0
  x2   0.75         400             0
  x3   0.5          600             90
  x4   0.25         800             90
  x5   0.125        1000            -90
All five signals have a duration of 1 second and a sampling rate of 44.1kHz
2) Generate the signal x6 as the sum of the five signal generated in point 1
3) Apply a Hanning window to select the first 512 samples of the signal x6(n).
   Provide the commands to plot the windowed signal
4) Compute the DFT of the signal obtained in step 3 using matrix multiplication
5) Compute the DFT of the signal obtained in step 3 using the MATLAB function fft
6) Compute the difference of the results in steps 4 and 5. Compute the maximum absolute error of the
   result in step 4 with reference to the result in step 5.
...Comment on what you expect from this result.
```

Solutions

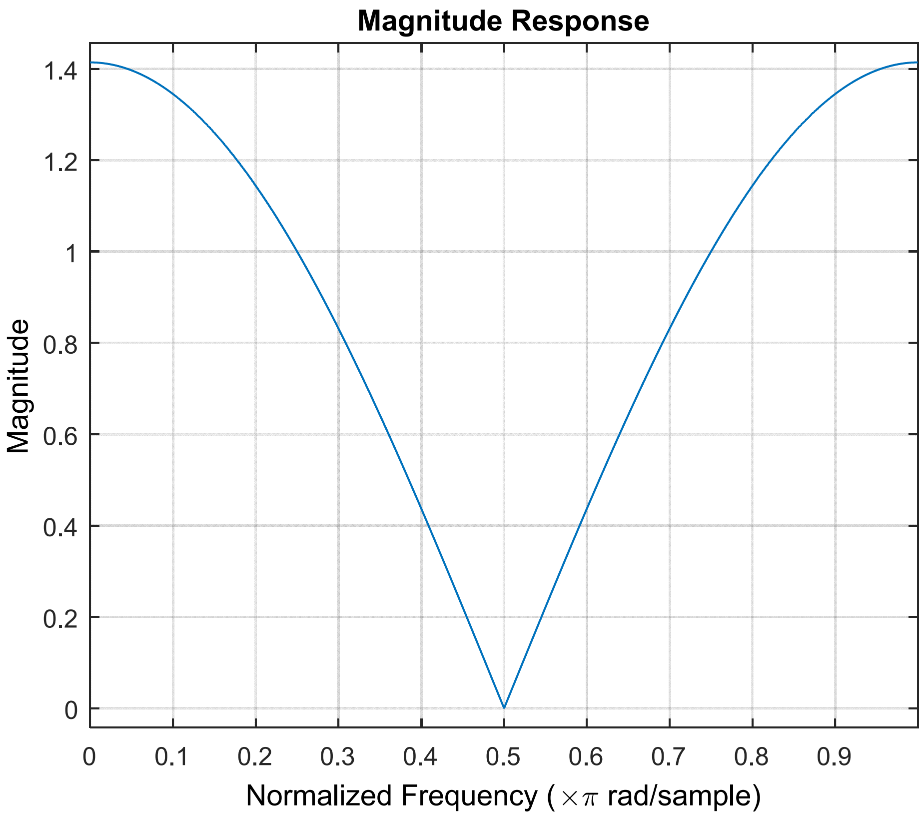
Ex.1

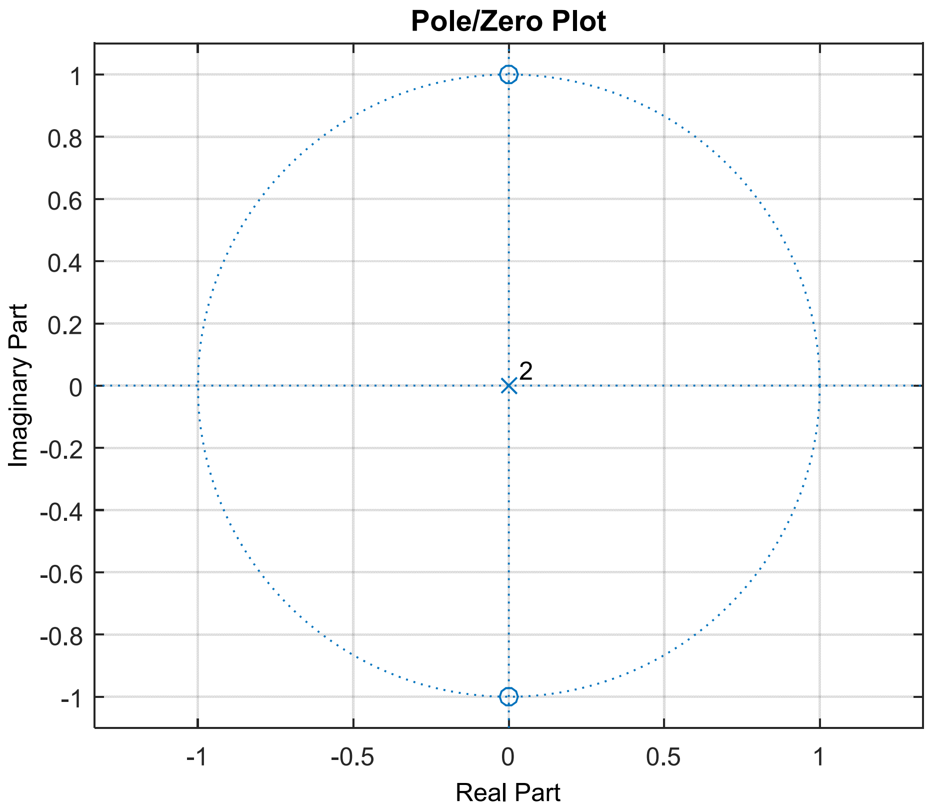
In order to remove the component at 6kHz ($6kHz\frac{2\pi}{24kHz}=\frac{\pi}{2}rad/s$ in normalized frequencies) we can apply two zeros at $\pm\frac{\pi}{2}$: $H(z)=A\left(1-jz^{-1}\right)\left(1+jz^{-1}\right)=A\left(1+z^{-2}\right)$

The filter magnitude in 3kHz ($3kHz\frac{2\pi}{24kHz}=\frac{\pi}{4}rad/s$) is

$$\left|H\left(z=e^{j\pi/4}\right)\right|=A\left|\left(1+e^{-j\pi/2}\right)\right|=A\left|e^{-j\pi/4}\left(e^{+j\pi/4}+e^{-j\pi/4}\right)\right|=A\cdot 2\cdot\cos\left(\frac{\pi}{4}\right)=A\sqrt{2}$$

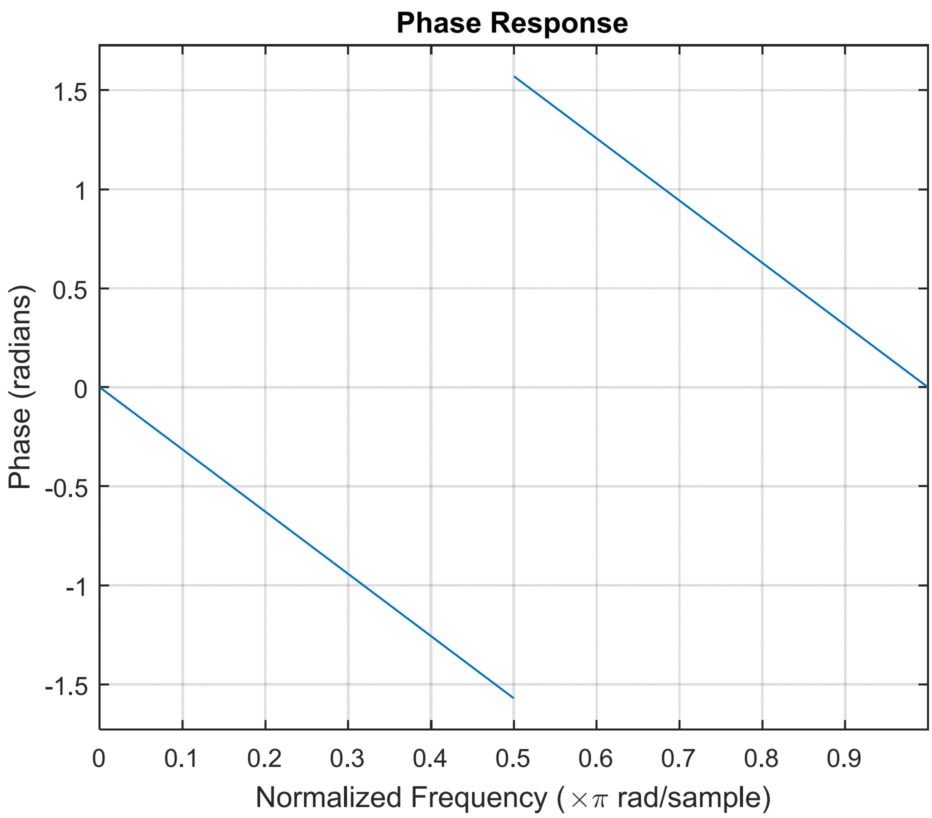
in order to get $\left|H\left(z=e^{j\pi/4}\right)\right|=1\rightarrow A\sqrt{2}=1\rightarrow A=\frac{\sqrt{2}}{2}$





The component at 9kHz will also keep its amplitude since $\left|H\left(z=e^{j\pi/4}\right)\right|=\left|H\left(z=e^{j3\pi/4}\right)\right|=1$

$$\measuredangle H\left(z=e^{j\omega}\right)=\measuredangle A\left(1+e^{-j2\omega}\right)=\measuredangle e^{-j\omega}\left(e^{j\omega}+e^{-j\omega}\right)=\measuredangle\left(e^{-j\omega}\cos(\omega)\right)=-\omega\cdot sign\left(\cos(\omega)\right)$$



In order to change the sample rate from 24kHz to 18kHz, since the greatest common divisor is 6kHz we have to upsample of an order of 3 and to downsample of an order of 4.

Ex.2

Apply the Yule-Walker formula:

$$\begin{bmatrix} r_0 & r_1 & r_2 \\ r_1 & r_0 & r_1 \\ r_2 & r_1 & r_0 \end{bmatrix}\begin{bmatrix} 1 \\ a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} \sigma^2{}_W \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1/2 & 1/5 \\ 1/2 & 1 & 1/2 \\ 1/5 & 1/2 & 1 \end{bmatrix}\begin{bmatrix} 1 \\ a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} \sigma^2{}_W \\ 0 \\ 0 \end{bmatrix}$$

$$a_0 = -\frac{8}{15}, \quad a_1 = \frac{1}{15}, \sigma^2{}_W = \frac{56}{15}$$

$$y[n] = a_0 y[n-1] + a_1 y[n-2] + w[n]$$

$$H(z) = \frac{1}{1 - a_0 z^{-1} - a_1 z^{-2}}$$

Ex.3

```
clear
clc
close all

% 1) Generate five cosine tones with  the following parameters:
%        Amplitude    Frequency [Hz]  Phase [deg]
%    x1  1.0          200             0
%    x2  0.75         400             0
%    x3  0.5          600             90
%    x4  0.25         800             90
%    x5  0.125        1000            -90
% All five signals have a duration of 1 second and a sampling rate of 44.1
% kHz
Fs = 44100;
t = (0:1/Fs:1)';
x1 = 1.0*cos(2*pi*200*t);
x2 = 0.75*cos(2*pi*400*t);
x3 = 0.5*cos(2*pi*600*t+pi/2);
x4 = 0.25*cos(2*pi*800*t+pi/2);
x5 = 0.125*cos(2*pi*1000*t-pi/2);

% 2) Generate the signal x6 as the sum of the five signal generated in
% point 1
x6 = x1+x2+x3+x4+x5;

% figure
% plot(t,x6)
% xlabel('Time [s]'), ylabel('Amplitude')
% title('Signal x_6(t)')

% 3) Apply a Hanning window to select the first 512 samples of the signal
% x6(t). Provide the commands to plot the windowd signal
N = 512;
win = hanning(N);
x6_win = x6(1:N) .* win;

figure
plot(t(1:N),x6_win)
```

```
ylabel('Time [s]'), ylabel('Amplitude')
title('Signal x6_{win}(t)')

% 4) Compute the DFT of the signal obtained in step 3 using matrix
% multiplication
W = exp(-1i*2*pi/N);
r = 0:N-1;
v = W.^r;
V = fliplr(vander(v));

X6_WIN = V*x6_win;

% 5) Compute the DFT of the signal obtained in step 3 using the MATLAB
% function fft
X6_WIN_FFT = fft(x6_win,N);

% 6) Compute the difference of the results in steps 4 and 5. Compute the maximum
absolute error of the result in step 4 with reference to the result in step 5.
%    Comment on what you expect from this result.
d = X6_WIN - X6_WIN_FFT;
abs_err = max(abs(d));
```