Multimedia Signal Processing 1st Module

Fundamentals of Multimedia Signal Processing

4/11/2017

Ex.1 (Pt.14)

An signal presents two spurious components represented by sinusoids at 500Hz and 1kHz that must be removed. The signal is sampled at 4kHz.
Propose a filter that is able to remove the two spurious components from the spectrum of the signal.

1. **[5 Pts.]** Provide the z transform of the filter and draw its zeros-poles plot.
2. **[3 Pts.]** Draw an approximate behaviour of the amplitude of the filter frequency response defining its exact gain at the zero frequency and at the Nyquist frequency
3. **[2 Pts.]** Provide the first 3 output samples of the impulse response of the filter.

Provide a further filter improvement that is able to alter as less as possible the original signal spectrum, (at least for its magnitude) but still removing the spurious frequencies.

4. **[4 Pts.]** Provide the z transform of this new filter.

Ex.2 (Pt.9)

We want to remove the continuous component (zero frequency) from the signal $x(n)$ working directly in the frequency domain using the DFT. We want to work just on small signal portions of 4 samples.

Propose a filter $H(k)$ **[3pts.]** in the frequency domain and apply it to the signal (provide also the W matrix) **[3pts.]** in order to get our goal; for the values of the signal portion assume 4 random numbers and calculate the result iDFT of the result $y(n)$ in the time domain **[3pts.]**. What are the assumptions on the signal?

Ex.3 (Pt. 11 – MATLAB code)

Given the signal x(t)= A_1 cos(2 pi f_1 t) + A_2 cos(2 pi f_2 t)

1. Create the signal x(n) as x(t) with t from 0 to 0.5 seconds sampled at Fs=8000 Hz. Use A_1=0.7, A_2=0.5, f_1=1800 Hz, f_2=3600 Hz.
2. Create the signal y(n) by re-sampling x(n) to 6000 Hz, without using the MATLAB functions for automatic re-sampling
3. Plot the magnitude of the 2048-point DFTs of the original and resampled signal (same plot, normalized frequency)

Solutions

Ex.1

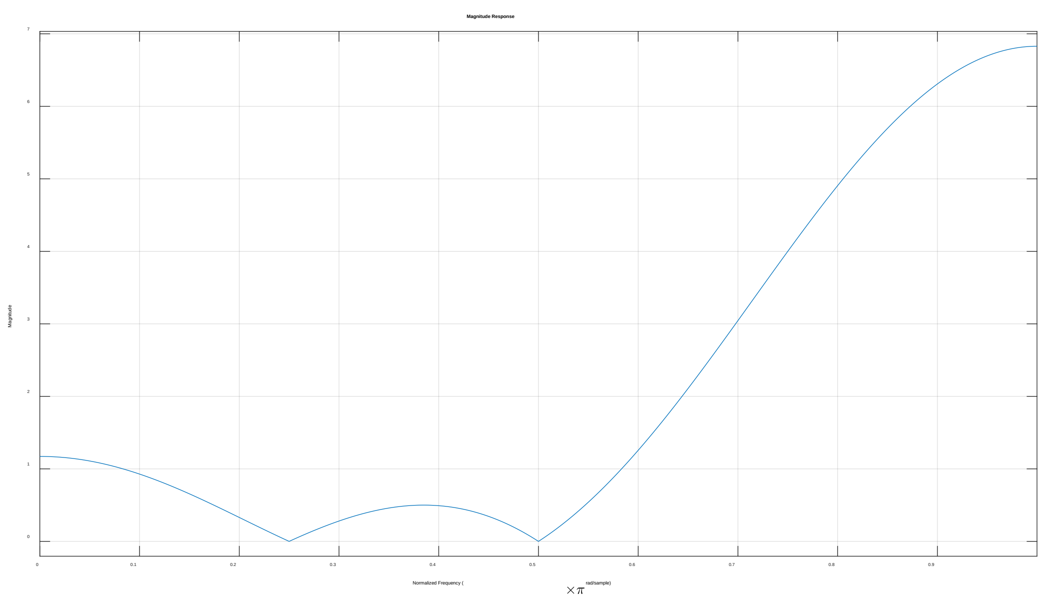
Since we are sampling the signal at 4kHz, in order to remove the two spurious sinusoids at 500Hz and 1000Hz we must completely attenuate the two normalized pulsations:

$$\omega_1 = \pm\frac{500Hz}{4000Hz}2\pi = \pm\frac{\pi}{4}$$

$$\omega_2 = \pm\frac{1000Hz}{4000Hz}2\pi = \pm\frac{\pi}{2}$$

So, to build the desired filter I have to place two zeros at those frequencies:

$$H_1(z) = \left(1 - 2\cos\left(\frac{\pi}{4}\right)z^{-1} + z^{-2}\right)\cdot\left(1 - 2\cos\left(\frac{\pi}{2}\right)z^{-1} + z^{-2}\right) =$$
$$= \left(1 - \sqrt{2}z^{-1} + z^{-2}\right)\cdot\left(1 + z^{-2}\right) = 1 - \sqrt{2}z^{-1} + 2z^{-2} - \sqrt{2}z^{-3} + z^{-4}$$



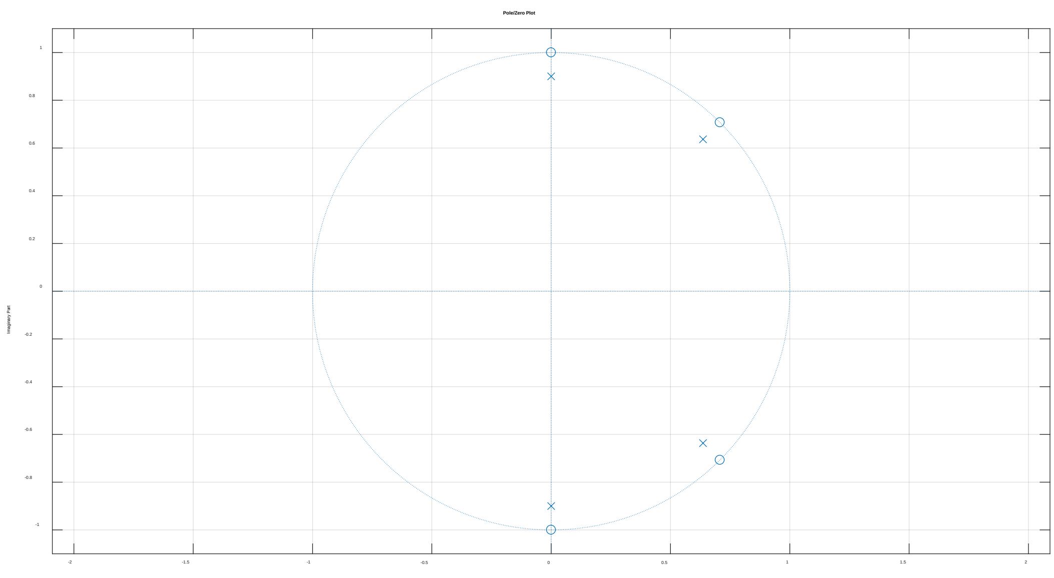
$$H_1(z=1) = 4 - 2\sqrt{2}$$
$$H_1(z=-1) = 4 + 2\sqrt{2}$$

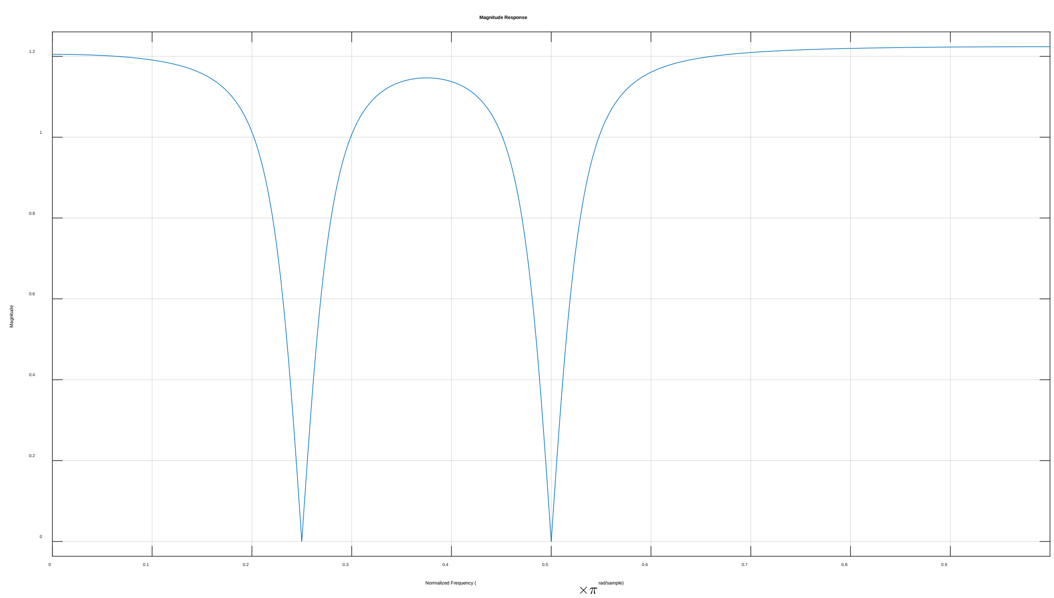
To reduce the alterations of such a filter on other frequencies of the spectrum of the original signal we can place two couples of conjugate poles close to the zeros:

$$H_2(z)=\frac{\left(1-2\cos\left(\frac{\pi}{4}\right)z^{-1}+z^{-2}\right)\cdot\left(1-2\cos\left(\frac{\pi}{2}\right)z^{-1}+z^{-2}\right)}{\left(1-2\rho\cos\left(\frac{\pi}{4}\right)z^{-1}+\rho^2z^{-2}\right)\cdot\left(1-2\rho\cos\left(\frac{\pi}{2}\right)z^{-1}+\rho^2z^{-2}\right)}=$$

$$=\frac{\left(1-\sqrt{2}z^{-1}+z^{-2}\right)\cdot\left(1+z^{-2}\right)}{\left(1-\sqrt{2}\rho z^{-1}+\rho^2z^{-2}\right)\cdot\left(1+\rho^2z^{-2}\right)}=\frac{1-\sqrt{2}z^{-1}+2z^{-2}-\sqrt{2}z^{-3}+z^{-4}}{1-\sqrt{2}\rho z^{-1}+2\rho^2z^{-2}-\sqrt{2}\rho^3z^{-3}+\rho^4z^{-4}}$$

Where a possible value for ρ could be, e.g. 0.9.





Ex.2

The filter will be $H(k)=\begin{bmatrix}0 & 1 & 1 & 1\end{bmatrix}$, a signal portion could be e.g. $x(n)=\begin{bmatrix}1 & 2 & -2 & 3\end{bmatrix}$.

$$\begin{bmatrix}1 & 1 & 1 & 1\end{bmatrix}$$

The W matrix will be:

So, $X[k]=[4 \quad 3+j \quad -6 \quad 3-j]$. Applying the filter the result will be $Y[k]=[0 \quad 3+j \quad -6 \quad 3-j]$ and in the time domain it will be: $y[n]=[0 \quad 1 \quad -3 \quad 2]$.

The signal is assumed periodic with a period of 4 samples.

Ex.3

```
clear all
close all
clc

% Given the signal x=A1*cos(f1*2*pi*t)+A2*cos(f2*2*pi*t);

% 1. Create the signal x with A1=0.7, f1= 1800Hz, A2=0.5,
f2=3600Hz
%    and t from 0 to 0.5 seconds sampled at Fs=8000Hz
Fs=8000;
t=0:1/Fs:0.5;
A1=0.7; f1=1800; A2=0.5; f2=3600;
x=A1*cos(2*pi*f1*t)+A2*cos(2*pi*f2*t);

% 2. Create the signal y by re-sampling x to 6000 Hz
%    (without using the MATLAB functions for automatic re-
sampling)

q=6000/Fs;
[L, M]=rat(q);

y_int=zeros(1,length(x)*L);
y_int(1:L:end)=x;
y_int=filter(fir1(63,1/L),1,y_int);
y_dec=filter(M*fir1(63,1/M),1,y_int);
y=y_dec(1:M:end);

% 3. Plot the magnitude of the 2048-point DFTs of the
original and the resampled
% signal (same plot, normalized frequency);
Nfft=2048;
X=fft(x, Nfft);
Y=fft(y, Nfft);
w_norm=linspace(0,2,Nfft);
figure;
plot(w_norm, abs(X)); hold on
```

```
plot(w_norm, abs(Y)); hold off
xlabel('normalized \omega');
ylabel('|X(k)|, |Y(k)|');
title('Magnitude of the DFT');
legend('Original Signal', 'Resampled signal');
```