

Multimedia Signal Processing 1st Module and Fundamentals of Multimedia Signal Processing

date: February 3rd, 2023

Ex.1 (Pt.12)

A digital filter $h[n]$ has the following DFT:

$$H[k] = \{3, 2, 1, 0, 1, 0, 1, 2\}.$$

It is used to filter a continuous signal $\bar{x}(t) = \cos(2\pi f_1 t) + 2 \sin(2\pi f_2 t)$, where $f_1 = 5\text{kHz}$ and $f_2 = 15\text{kHz}$, sampled at 40kHz .

[7 pts] Define the steps to filter the signal working with in the frequency domain, in particular: what is the sampled signal, how it can be transformed in the frequency domain, how to apply the filter in the frequency domain, how the output can be converted back in the time domain).

[5 pts] What will be the output signal $y(n)$?

Ex.2 (Pt.8)

Compute the time response of the system described by the following transfer function:

$$H(z) = \frac{(1 - z^{-1})^2}{1 - 0.32z^{-1} + 0.8z^{-2}}$$

[4 pts] Provide the first 4 outputs when the input signal is the unit step.

[4 pts] Provide the first 4 outputs when the input signal is $x(n) = \delta(n) + 1$.

CONTINUES ON THE BACK

Ex.3 (Pt.12)

1) [2 pt] You are given a signal $x(t)$, sampled every $T_s = 62.5$ microseconds, with the following DFT:



Define the signal $x(t)$ which has duration of 50 milliseconds.

- 2) [6.5 pt] We want to change the sampling rate of the system to $F_{s_new} = 12\text{KHz}$.
- Implement the correct chain of operations to avoid aliasing artifacts. If needed, use a filter with 129 coefficients. Define the final obtained signal as y .
 - Define a signal z which is the result of the sampling rate change applied to x but contains aliasing artifacts.
 - Plot the absolute value of the DFTs of y and z as a function of the normalized frequency axis (keep F_s as normalization frequency).
 - Which are the positions of the peaks of $Y(f)$? To which components of the signal x are they due?
 - $Z(f)$ contains multiple peaks. Comments on the positions of all the peaks (you can stop at the Nyquist frequency).
- 3) [3.5 pt] We want to remove the aliasing artifacts from the signal z such that it resembles the signal y .
- Starting from z , can we totally remove the artifacts or there will remain any unremovable aliasing components?
 - Design an FIR band-pass filter to remove the artifacts from z (those we can remove). You choose the parameters to design the filter.
 - As an alternative, design an IIR notch-filter to remove the same artifacts from z .

Solutions

Ex.1

The sampled signal will be:

$$\begin{aligned}x[n] &= \cos\left(2\pi \frac{5}{40} n\right) + 2 \sin\left(2\pi \frac{15}{40} n\right) = \cos\left(\frac{\pi}{4} n\right) + 2 \sin\left(\frac{3}{4} \pi n\right) = \\ &= \frac{1}{2} \left(e^{-j\frac{\pi}{4}n} + e^{j\frac{\pi}{4}n} \right) + j \left(e^{j\frac{3\pi}{4}n} - e^{-j\frac{3\pi}{4}n} \right)\end{aligned}$$

It is periodic with a period of 8 samples and the dft of a period starting, e.g. from $n = 0$ will be:

$$X[k = 0..7] = 8 \cdot \left\{ 0, \frac{1}{2}, 0, -j, 0, j, 0, \frac{1}{2} \right\}$$

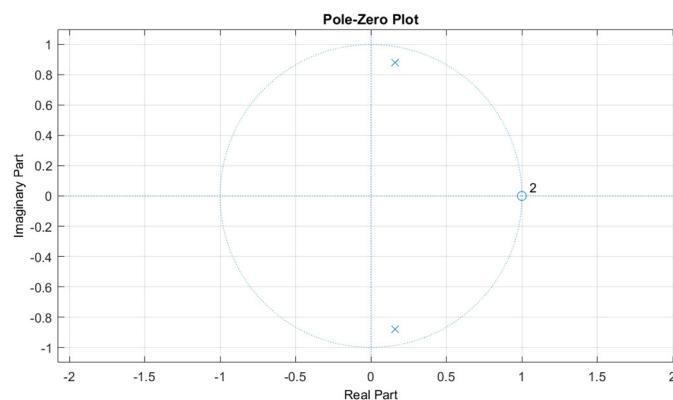
Evaluating the dft from a different starting point will just introduce a phase term:

$$x[n-l] \rightarrow X[k] \cdot e^{-j\frac{\pi}{4}kl}$$

The overall effect of the filter $H[k]$, obtained multiplying elementwise $H[k]$ with $X[k]$, will then be doubling the amplitude of the components at $\pm \frac{\pi}{4}$ and removing the components at $\pm \frac{3\pi}{4}$.

$$\text{Then } y[n] = 2 \cos\left(\frac{\pi}{4} n\right)$$

Ex.2

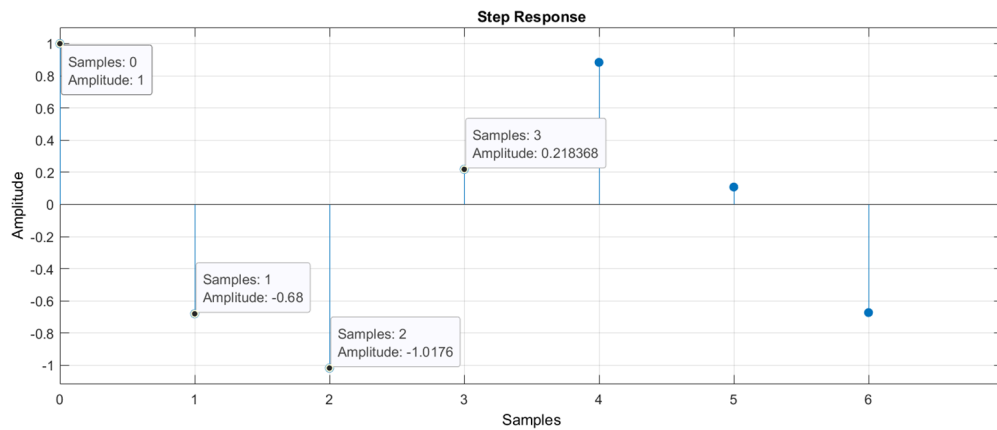


The filter can be written in time domain as:

$$y[n] = 0.32y[n-1] - 0.8y[n-2] + x[n] - 2x[n-1] + x[n-2].$$

The step response is:

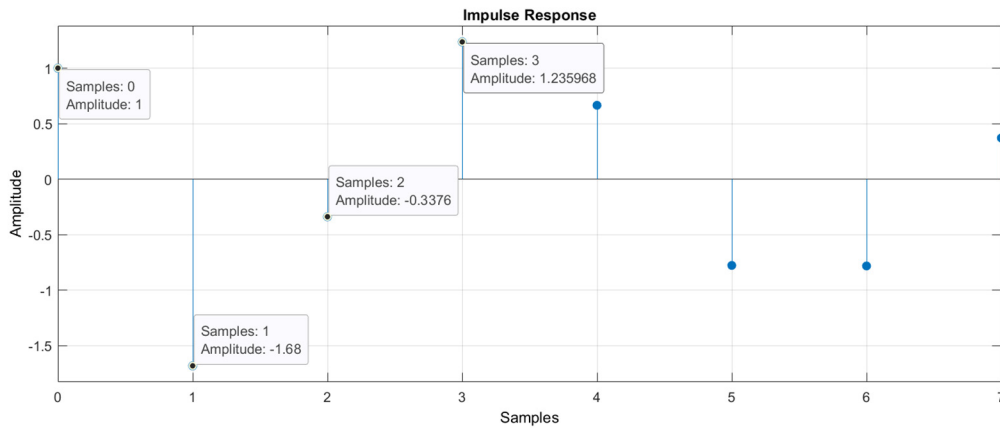
n	x	Y
0	1	1
1	1	$0.32+1-2=-0.68$
2	1	$0.32 \cdot (-0.68)-0.8+1-2+1=-1.0176$
3	1	$0.32 \cdot (-1.0176)-0.8 \cdot (-0.68)+1-2+1=0.2184$



In the second case, when $x(n) = \delta(n) + 1$, we can remove the constant "1" since, due to the presence of the zero (double zero) in the origin, the continuous component will be completely removed, so, the signal can be considered just as an impulse.

The system response will then be:

n	x	Y
0	1	1
1	0	$0.32-2=-1.68$
2	0	$0.32 \cdot (-1.68)-0.8+1=-0.3376$
3	0	$0.32 \cdot (-0.3376)-0.8 \cdot (-1.68)=1.236$



Ex.3 (MATLAB CODE)

```
close all
clearvars
clc
```

```
%% 1. [2 pt]
```

```
% You are given a signal x(t), sampled every Ts = 62.5 microseconds,
% with the following DFT (see the exam text).
% Define the signal x(t) which has duration of 50 milliseconds.
```

```
f0 = 1/8;
f1 = 1/4;
Fs = 1/62.5e-6;
duration = 50e-3;
```

```
time = 0:1/Fs:duration - 1/Fs;
```

```
x = cos(2*pi*f0*Fs*time) + cos(2*pi*f1*Fs*time);
```

```
%% 2. [6.5 pt]
```

```
% We want to change the sampling rate of the system to Fs_new = 12KHz.
% Implement the correct chain of operations to avoid aliasing artifacts.
% If needed, use a filter with 129 coefficients.
% Define the final obtained signal as y.
% Define a signal z which is the result of the sampling rate change
applied
% to x but contains aliasing artifacts.
% Plot the absolute value of the DFTs of y and z as a function of the
% normalized frequency axis (keep Fs as normalization frequency).
% Which are the positions of the peaks of Y(f)?
% To which components of the signal x are they due?
% Z(f) contains multiple peaks. Comments on the positions of all
% the peaks (you can stop at the Nyquist frequency)
```

```
% Fs = 1/62.5e-6 = 2e6/125 = 2^4 * 10^3 = 16KHz
% Fs_new = 12 KHz
```

```

% --> Fs_new/Fs = 12/16 --> L = 3, M = 4
L = 3;
M = 4;
Fs_new = Fs * L/M;

% upsampling
x_upsampled = zeros(1, length(x) * L);
x_upsampled(1:L:end) = x;

% filtering
cutoff = min([1/(2*L), 1/(2*M)]);
cutoff_filter = 2*cutoff;
h_multirate = L*fir1(128, cutoff_filter);
x_f = filter(h_multirate, 1, x_upsampled);

% decimation (absence of aliasing artifacts)
y = x_f(1:M:end);
% downsampling (presence of aliasing artifacts)
z = x_upsampled(1:M:end);

% Plot the absolute value of the DFTs of y and z as a function of the
% normalized frequency axis (keep Fs as normalization frequency).
N = length(y);
Y = fft(y);
Z = fft(z);
freq_axis = 0:1/N:1 -1/N;

figure;
plot(freq_axis, abs(Y));
grid;
title('|DFT| of the signal y(n)');
xlabel('f [norm]');

% Which are the positions of the peaks of Y(f)?
% y(n) is the result of the correct chain of operations.
% After the upsampling by L = 3, we find peaks (until Nyquist) in:
% related to f0: 1/24, 1/3 - 1/24 = 7/24, 1/3 + 1/24 = 9/24
% related to f1: 1/12, 1/3 - 1/12 = 1/4, 1/3 + 1/12 = 5/12
% After the filter, only the frequencies before the cut-off are kept
% cut-off = 1/8 --> we keep only 1/24 (related to f0), 1/12 (related to
f1)
% After the downsampling, everything is expanded by M = 4, thus we find:
% related to f0: 1/6
% related to f1: 1/3
% After the Nyquist, everything is symmetric

figure;
plot(freq_axis, abs(Z));
grid;
title('|DFT| of the signal z(n)');
xlabel('f [norm]');

% Z(f) contains multiple peaks. Comments on the positions of all the peaks
% (you can stop at the Nyquist frequency).
% z(n) is the result of upsampling-downsampling (without the filtering).
% After the upsampling by L = 3, we find peaks (until Nyquist) in:
% related to f0: 1/24, 1/3 - 1/24 = 7/24, 1/3 + 1/24 = 9/24
% related to f1: 1/12, 1/3 - 1/12 = 1/4, 1/3 + 1/12 = 5/12

```

```

% After the downsampling, everything is expanded by  $M = 4$ , thus we find:
% related to  $f_0$ :  $1/24 \rightarrow 1/6$ ,  $7/24 \rightarrow 28/24 \pmod{1} = 1/6$ ,  $9/24 \rightarrow$ 
 $36/24 \pmod{1} = 1/2$ 
% related to  $f_1$ :  $1/12 \rightarrow 1/3$ ,  $1/4 \rightarrow 1 \pmod{1} = 0$ ,  $5/12 \rightarrow 20/12 \pmod{1}$ 
 $= 2/3$  (which creates a symmetric peak before the Nyquist at  $1/3$ ).
% Overall, we find peaks in these frequencies:  $0, 1/6, 1/3, 1/2$ .
% After the Nyquist, everything is symmetric

```

```

%% 3. [3.5 pt]

```

```

% We want to remove the aliasing artifacts from the signal  $z$  such that it
% resembles the signal  $y$ .

```

```

% Starting from  $z$ , can we totally remove the artifacts or there will
remain

```

```

% some unremovable aliasing components?

```

```

% Design an FIR band-pass filter to remove the artifacts from  $z$  (those we
% can remove). You choose the parameters to design the filter.

```

```

% As an alternative, design an IIR notch-filter to remove the same
artifacts from  $z$ .

```

```

% Actually, all the peaks of  $z$  contain aliasing artifacts.

```

```

% As a matter of fact, also the peaks in  $1/6$  and  $1/3$  (which are present in
%  $y$ ) contain aliasing due to the contributions in the upsampled signal in
%  $7/24$  (which, after downsampling, becomes  $1/6$ ) and in  $5/12$  (which, after
% downsampling, becomes  $1/3$ ).

```

```

% So, we cannot completely remove the artifacts.

```

```

% FIR band-pass filter:

```

```

% we want to remove the contributions in  $0$  and  $0.5$  and maintain the
% contributions in  $1/6$  and  $1/3$ .

```

```

% We can design a band-pass filter with two cutoff frequencies

```

```

% You can choose the frequencies, leaving some space for the transition
% band for example

```

```

transition_band = 1/12;

```

```

cut_off_left = 1/6 - 1/12;

```

```

cut_off_right = 1/3 + 1/12;

```

```

% create the FIR filter

```

```

h = fir1(64, [cut_off_left, cut_off_right]*2);

```

```

% to better analyze what happens to the signal  $z$  (not required)

```

```

w = filter(h, 1, z);

```

```

W = fft(w);

```

```

figure;

```

```

plot(freq_axis, abs(W));

```

```

grid;

```

```

title('|DFT| of the FIR-filtered signal  $z(n)$ ');

```

```

xlabel('f [norm]');

```

```

% Notch filter:

```

```

% We want to completely remove frequencies  $0$  and  $0.5$  and maintain the
% contributions in  $1/6$  and  $1/3$ .

```

```

% We can design a filter with zeros on the unit circle and poles with same
% phase of the zeros but absolute value  $0.9$ .

```

```

zeroes = [1, -1];

```

```

poles = [0.9, -0.9];

```

```

A = poly(poles);

```

```

B = poly(zeroes);

```

```
% to better analyze what happens to the signal z (not required)
w = filter(B, A, z);
W = fft(w);
figure;
plot(freq_axis, abs(W));
grid;
title('|DFT| of the Notch-filtered signal z(n)');
xlabel('f [norm]');
```