MultimediaSignal Processing 1st Module

MATLAB part [12 pts]

June 29th, 2021

Text:

1. [5 pt] You are given a filter H(z) = (1 - a^50*z^(-50)) / (1 – a*z^(-1)).
    - Determine the values of "a" among the set [0.9*exp(-pi*j), -2j*exp(-3/2*pi*j), 0.5*exp(-pi/4*j)] such that the filter is real. (hint: once selected the acceptable value (or values), take its/their real part with the function "real" otherwise Matlab considers it/them as complex variable).
    - Is the filter FIR or IIR? Does it depend on the chosen value(s) of "a" or not? (hint: focus on the numerator terms. z^(-50) in time corresponds to …?)
    - Find the causal filter coefficients for all the selected values of "a" in the first 100 samples.
    - Plot the impulse response of the filter h(n) for all the selected values of "a". Use the function "stem".
    - Select the value of "a" which returns a filter with decreasing coefficients.
    - Plot the amplitude of the frequency response of the selected filter as a function of the frequency axis starting from 0. Use the function "freqz", using the same number of samples of h(n) and using the flag "whole" to see the entire frequency spectrum.
2. [1.5 pt] Define a sinusoidal signal s as the sum of two cosinusoids. One is centered around the peak frequency of the selected filter, the other one has a period equal to 100 times that of the first sinusoid. The sinusoids have the same amplitude = 1. The signal s is defined over 600 samples.
3. [1 pt] Filter the signal s with the selected filter.
    - Plot the amplitude of the DFT of the signal s and of the filtered signal (better use "stem").
    - How does the amplitude of DFT(s_f) change with respect to DFT(s)?
4. [4.5 pt] Apply a window to the signal s. Try two different windows: the rectangular window (use the function "rectwin") and the blackman window (use the function "blackman"). In particular, choose the length Nw of the window among the values [250, 300, 350, 400] such that the windowed signal maintains a number of samples which is an integer multiple of the period of s.
    - Plot, in the same figure, the signal s, and the two windowed signals obtained with the two windows.
    - Plot the amplitude of the DFTs of the three signals as a function of frequency axis starting from 0. Use the function "stem".
    - Are there any differences between the DFT of s and DFT of s windowed with rectwin? If yes, why? Are there any differences between the DFT of s and DFT of s windowed with blackman? If yes, why? (hint: to compare the three DFTs, it is better if you normalize each DFT amplitude by the number of samples of DFT).
    - Justify the behaviour of the DFT of the signal windowed with blackman. (hint: in time domain, we are windowing the signal. In frequency domain, this corresponds to...? Perform the frequency domain computations and verify your answer with a plot, plotting the DFT of the windowed signal with blackman and your frequency domain result).

Solution:

```
close all
clearvars
clc

%% 1.

% [5 pt] You are given a filter H(z) = (1 - a^50*z^(-50)) / (1 - az^(-1)).
% Determine the values of "a" among the set [0.9*exp(-pi*j), -2j*exp(-
3/2*pi*j),
% 0.5*exp(-pi/4*j)] such that the filter is real.  (hint: once selected
the
% acceptable value (or values), take its/their real part with the function
"real"
% otherwise Matlab considers it/them as complex variable).
% Is the filter FIR or IIR? Does it depend on the chosen value(s) of "a"
or not?
% (hint: focus on the numerator terms. z^(-50) in time corresponds to
,Ä¶?)
% Find the causal filter coefficients for all the selected values of "a"
in
% the first 100 samples.
% Plot the impulse response of the filter h(n) for all the selected values
% of "a". Use the function "stem".
% Select the value of "a" which returns a filter with decreasing
coefficients.
% Plot the amplitude of the frequency response of the selected filter as a
% function of the frequency axis starting from 0. Use the function
"freqz",
% using the same number of samples of h(n) and using the flag "whole" to
see
% the entire frequency spectrum.

% Determine the values of a among the set [0.9*exp(-pi*j),
% -2j*exp(-3/2*pi*j), 0.5*exp(-pi/4*j)] such that the filter is real.
% The real values are the first and the second one
a1 = real(0.9*exp(-pi*1i));
a2 = real(-2*1i*exp(-3/2*pi*1i));
% We can define the two possible filters:
A1 = [1, -a1];
A2 = [1, -a2];
B1 = zeros(1, 51);
B1(1) = 1;
B1(51) = -a1^50;
B2 = zeros(1, 51);
B2(1) =1;
B2(51) = -a2^50;
```

```
% Is the filter FIR or IIR? Does it depend on the chosen value(s) of "a"
or
% not?
% The filter is FIR and this behaviour does not depend on the chosen value
% of "a".
% We can notice that the filter is FIR in two ways: (i) computing the
% Z^{-1} transform; (ii) evaluating the function filter for a certain
% number of samples and checking that after some time it goes to 0.
% the Z^{-1} transform can be easily computed by separating the terms at
numerator:
% H(z) = 1/ (1 - az^(-1)) - a^50*z^(-50)) / (1 - az^(-1)) -->
% h(n) = a^n*u(n) - a^50 * conv(delta(n-50), a^n*u(n)) -->
% h(n) = a^n *u(n) - a^50 * a^(n-50)*u(n-50) -->
% h(n) = a^n * u(n) - a^n * u(n-50) -->
% the second term deletes the first one for n >= 50. --> the filter is
FIR.

% Find the causal filter coefficients for all the selected values of "a"
in
% the first 100 samples.
% We can define a delta signal
delta = zeros(1, 100);
delta(1) = 1;
h1 = filter(B1, A1, delta);
h2 = filter(B2, A2, delta);

% Plot the impulse response of the filter h(n) for all the selected values
% of "a". Use the function "stem".
% We can notice that both the two filters are limited in time. For n >=
50,
% they are all zeros.
figure;
stem(h1);
figure;
stem(h2);

% Select the value of "a" which returns a filter with decreasing
coefficients.
% The chosen filter is h1.
A = A1;
B = B1;
h = h1;

% Plot the amplitude of the frequency response of the selected filter as a
% function of the frequency axis starting from 0. Use the function
"freqz",
% using the same number of samples of h(n) and using the flag "whole" to
see
% the entire frequency spectrum.
```

```
N = length(h);
H = freqz(B, A, N, 'whole');
figure;
plot(0:1/N:(1-1/N), abs(H));

%% 2.

% [1.5 pt] Define a sinusoidal signal s as the sum of two cosinusoids.
% One is centered around the peak frequency of the selected filter,
% the other one has a period equal to 100 times that of the first
sinusoid.
% The sinusoids have the same amplitude = 1.
% The signal s is defined over 600 samples.

% the peak frequency of the filter can be derived in two ways: (i)
% checking the maximum of the plotted DFT amplitude as a function of
frequency;
% (ii) checking the denominator of the filter = 1 + 0.9z^-1. --> this
% corresponds to a root in z = -0.9 --> frequency = 0.5.
f0 = 0.5;
% the frequency is the inverse of the period
f1 = f0/100;
% sample axis goes from 0 until 600 -1 samples
n = 0:600-1;
s0 = cos(2*pi*f0*n);
s1 = cos(2*pi*f1*n);
s = s0 + s1;

%% 3.

% [1 pt] Filter the signal s with the selected filter.
% Plot the amplitude of the DFT of the signal s and of the filtered signal
(better use "stem").
% How does the amplitude of DFT(s_f) change with respect to DFT(s)?


s_f = filter(B, A, s);

% DFT of s
S = fft(s);
% DFT of the filtered signal.
S_f = fft(s_f);

% frequency axis:
freq_axis = 0:1/length(s):1 - 1/length(s);

figure;
stem(freq_axis, abs(S));
leg{1} = 'Original signal';
```

```matlab
hold on;
stem(freq_axis, abs(S_f));
leg{2} = 'Filtered signal';
legend(leg);

% We can notice that, in correspondence of f0, the filtered signal
% is enhanced (the filter has a maximum in f0). In f1, the filtered signal
% is attenuated following the behaviour of the filter.

%% 4.

% [4.5 pt] Apply a window to the signal s.
% Try two different windows: the rectangular window (use the function
"rectwin")
% and the blackman window (use the function "blackman").
% In particular, choose the length Nw of the window among the values
% [250, 300, 350, 400] such that the windowed signal maintains a number of
% samples which is an integer multiple of the period of s.
% Plot, in the same figure, the signal s, and the two windowed signals
% obtained with the two windows.
% Plot the amplitude of the DFTs of the three signals as a function of
% frequency axis starting from 0. Use the function stem.
% Are there any differences between the DFT(s) and DFT(s windowed with
rectwin)?
% If yes, why? Are there any differences between the DFT(s) and DFT(s
windowed
% with blackman)? If yes, why? (hint: to compare the three DFTs, it is
better
% if you normalize each amplitude by the number of samples of the DFT).
% Justify the behaviour of the DFT of the signal windowed with blackman.
% (hint: in time domain, we are windowing the signal. In frequency domain,
% this corresponds to...? Perform the frequency domain computations and
% verify your answer with a plot, plotting the DFT of the windowed signal
with
% blackman and your frequency domain result).


% Cchoose the length Nw of the window among the values
% [250, 300, 350, 400] such that the windowed signal maintains a number of
% samples which is an integer multiple of the period of s.

% p0 = 1/f0 = 2 samples
% p1 = 1/f1 = 200 samples
% Period of s = least common multiple(p0, p1) = 200 samples.
% --> we select Nw = 400.
Nw = 400;

% Try two different windows:
% the rectangular window and the blackman window
```

```
w_r = rectwin(Nw);
w_b = blackman(Nw);

% Window the signal s. Since the windows are defined as column vectors, we
% have to transpose them.
s_w_r = s(1:Nw) .* w_r';
s_w_b = s(1:Nw) .* w_b';

% Plot, in the same figure, the signal s, and the two windowed signals
% obtained with the two windows.
figure;
plot(n, s);
leg{1} = 'Original signal';
hold on;
plot(0:Nw-1, s_w_r);
leg{2} = 'Windowed signal with rectwin';
plot(0:Nw-1, s_w_b);
leg{3} = 'Windowed signal with blackman';
legend(leg);

% Plot the amplitude of the DFTs of the three signals as a function of
% frequency axis starting from 0. Use the function stem.
% Are there any differences between the DFT(s) and DFT(s windowed with
rectwin)?
% If yes, why? Are there any differences between the DFT(s) and DFT(s
windowed
% with blackman)? If yes, why? (hint: to compare the three DFTs, it is
better
% if you normalize each amplitude by the number of samples of the DFT).

% DFTs of the windowed signals
S_w_r = fft(s_w_r);
S_w_b = fft(s_w_b);

% normalize each DFT by the number of samples
figure;
stem(0:1/length(s):1 - 1/length(s), abs(S)/length(S));
leg{1} = 'DFT of the original signal';
hold on;
stem(0:1/Nw:1 - 1/Nw, abs(S_w_r)/Nw, '--');
leg{2} = 'DFT of the windowed signal with rectwin';
stem(0:1/Nw:1 - 1/Nw, abs(S_w_b)/Nw, '-.');
leg{3} = 'DFT of the windowed signal with blackman';
legend(leg);

% as expected, the peaks of DFTs of s and s_w_r are the same. The
rectangular window
% cuts the signal in a number of samples multiple of the signal period -->
% the DFT peaks remain exactly the same as before.
```

```
% The DFT of s_w_b is different from that of s, and this is due to the
% window which has not an ideal rectangular behaviour.

% Justify the behaviour of the DFT of the signal windowed with blackman.
% (hint: in time domain, we are windowing the signal. In frequency domain,
% this corresponds to...? Perform the frequency domain computations and
% verify your answer with a plot, plotting the DFT of the windowed signal
with
% blackman and your frequency domain result).
% the signal s_w_b is the result of an element-wise product in time domain
% between s and w_b. In frequency domain, this corresponds to the cyclic
% convolution between DFT(s) and DFT(w_b) --> to verify:

% compute the DFTs of s (until Nw samples) and w_b.
Sw = fft(s(1:Nw));
W_b = fft(w_b');

% compute the cyclic convolution between the two terms, over Nw samples.
cc_s_w_b = cconv(Sw, W_b, Nw);

% Plot the result with the previously computed S_w_b, and check they are
% the same. (we need a further normalization to adjust the amplitudes)

figure,
stem(0:1/Nw:1 - 1/Nw, abs(cc_s_w_b)/(Nw));
leg = {};
leg{1} = 'Cyclic conv between the DFT of s and the DFT of w_b';
hold on;
stem(0:1/Nw:1 - 1/Nw, abs(S_w_b), '--');
leg{2} = 'DFT of the windowed signal with blackman';
legend(leg);
```