Electrical and Computer Engineering Department

Digital Signal Processing

ENCS4310

Second Semester 2022/2023

Assignment 1

Prepared by: Omar Masalmah

ID: 1200060

Instructor's Name: Dr. Qadri Mayyala

Section: 2

Date: 12-05-2023

Consider the following continuous time signal:

$$x(t) = \cos(2\pi 2t) + 0.5\cos(2\pi 50t) + 0.25\cos(2\pi 80t)$$

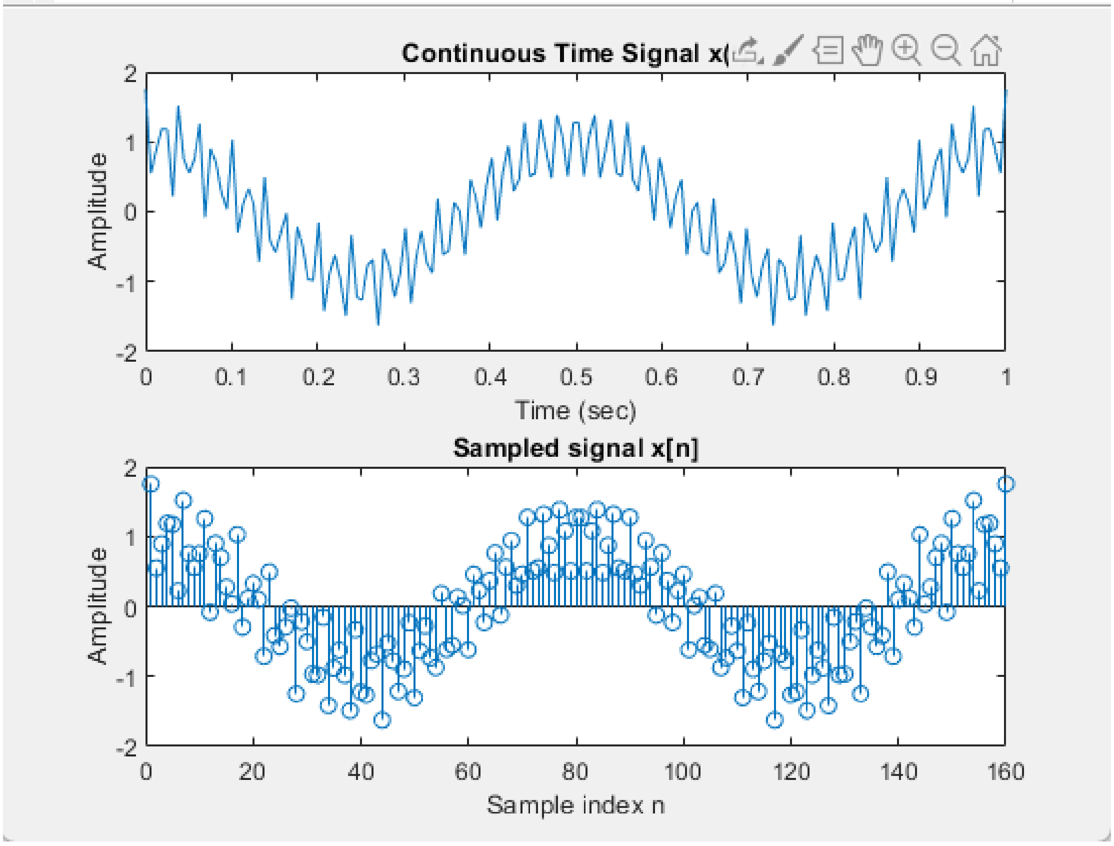
Let Fs = 160 samples/sec.

a) Plot x[n] for 1 sec (i.e., 160 samples)

```
% Define the continuous time signal
t = linspace(0, 1, 160); % time vector from 0 to 1 sec with 160 samples
x = cos(2*pi*2*t) + 0.5*cos(2*pi*50*t) + 0.25*cos(2*pi*80*t); % signal definition

% Plot the signal
subplot(2,1,1);
plot(t, x);
xlabel('Time (sec)');
ylabel('Amplitude');
title('Continuous Time Signal x(t)');

% Stem the sampled signal xn
subplot(2,1,2);
stem(x);
title('Sampled signal x[n]');
xlabel('Sample index n');
ylabel('Amplitude');
```

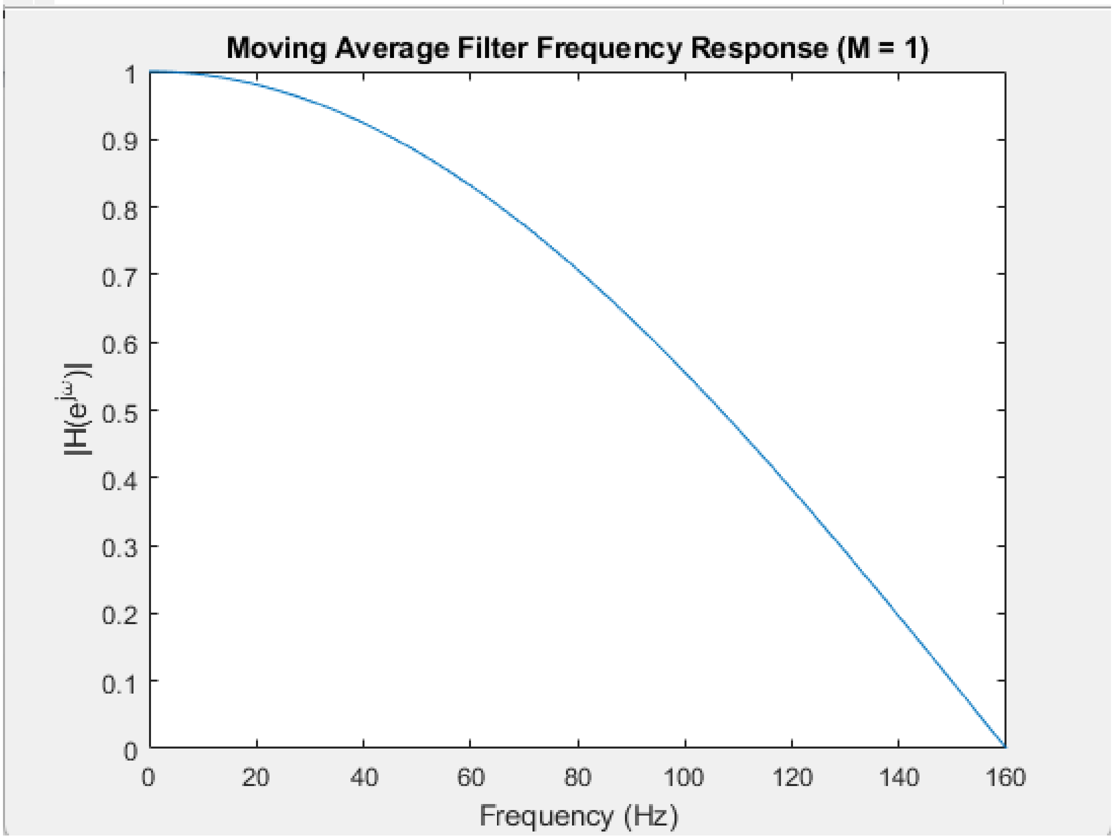


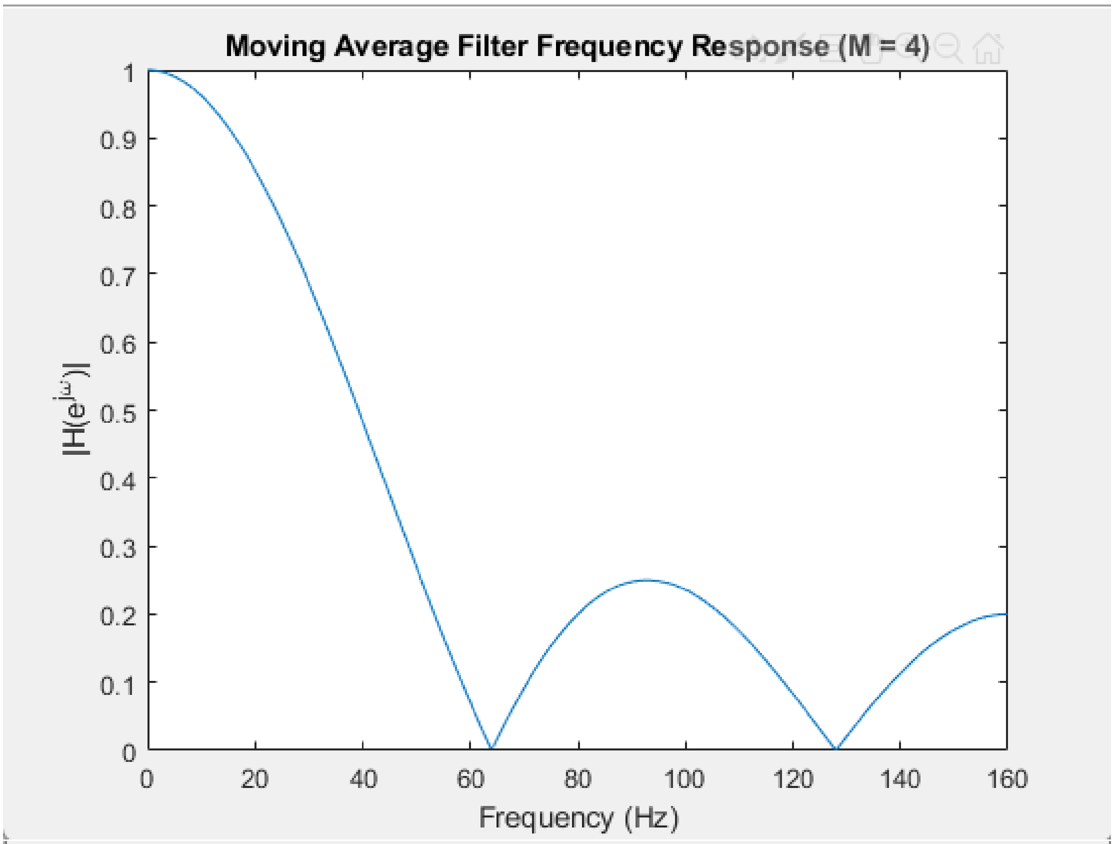
b) Consider the moving average filter $y[n]=\dfrac{1}{M+1}\sum_{k=0}^{M}x[n-k]$ (M: window size)

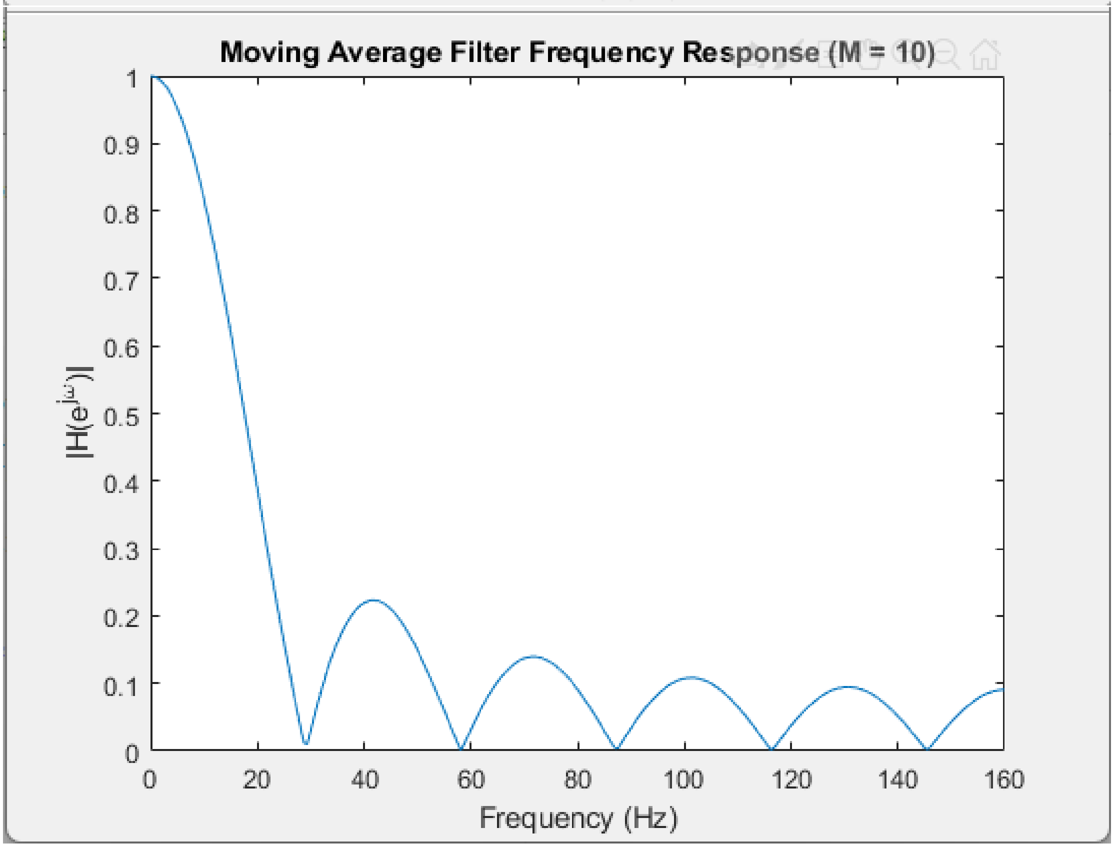
Plot the filter frequency response $|H(e^{j\omega})|$ for different values of M (M=0, M=4, M=10), give your conclusions.

```
% Define the continuous time signal
t = linspace(0, 1, 160); % time vector from 0 to 1 sec with 160 samples
x = cos(2*pi*2*t) + 0.5*cos(2*pi*50*t) + 0.25*cos(2*pi*80*t); % signal definition

% Define the filter
M = [1 4 10]; % different window sizes
for i = 1:length(M)
    h = ones(1, M(i) + 1)/(M(i) + 1); % moving average filter
    H = fft(h, 512); % compute the frequency response
    f = linspace(0, 1, 512/2 + 1)*160; % frequency vector in Hz
    figure;
    plot(f, abs(H(1:512/2 + 1))); % plot the magnitude spectrum
    xlabel('Frequency (Hz)');
    ylabel('|H(e^{j\omega})|');
    title(['Moving Average Filter Frequency Response (M = ' num2str(M(i)) ')']);
end
```



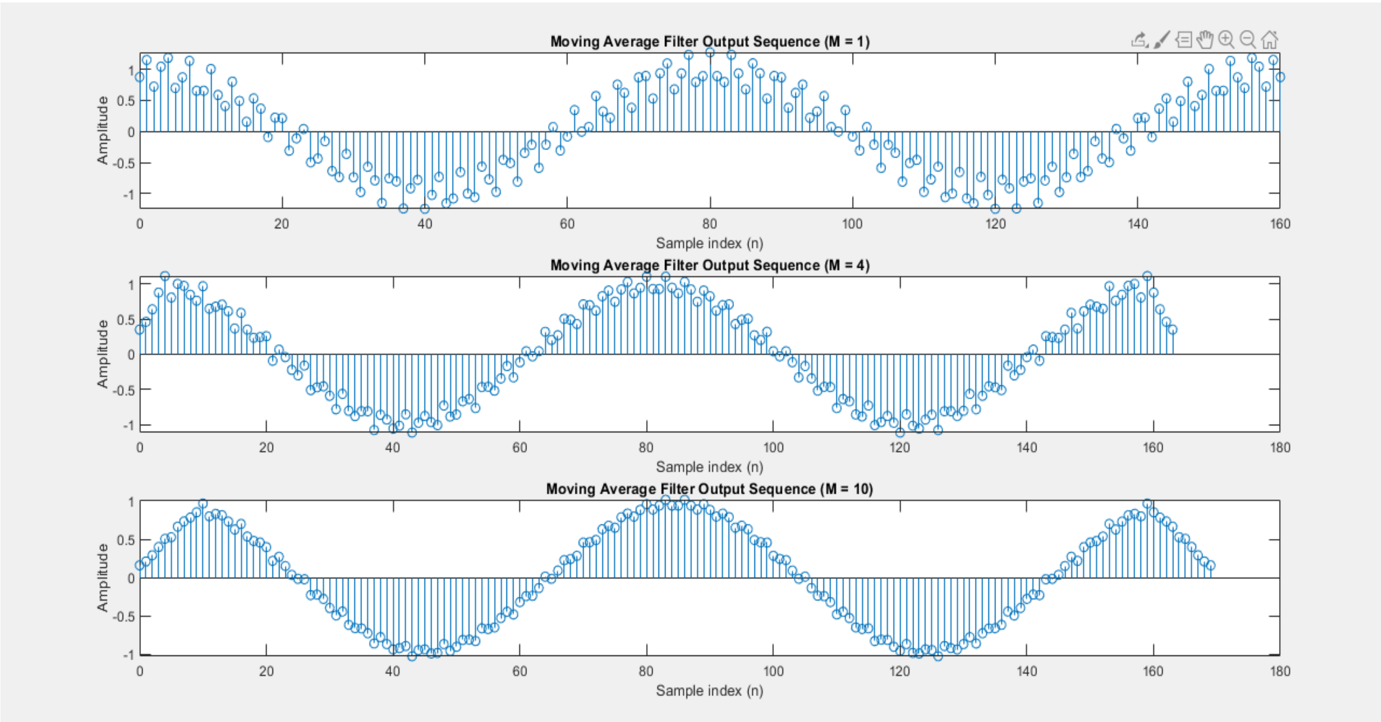

Moving Average Filter Frequency Response (M = 4)
|H(e^{jω})|
Frequency (Hz)
0
20
40
60
80
100
120
140
160
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


Moving Average Filter Frequency Response (M = 10)
|H(e^{jω})|
Frequency (Hz)
0
20
40
60
80
100
120
140
160
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

c) Plot the response (output sequence y[n]) for the different window size.

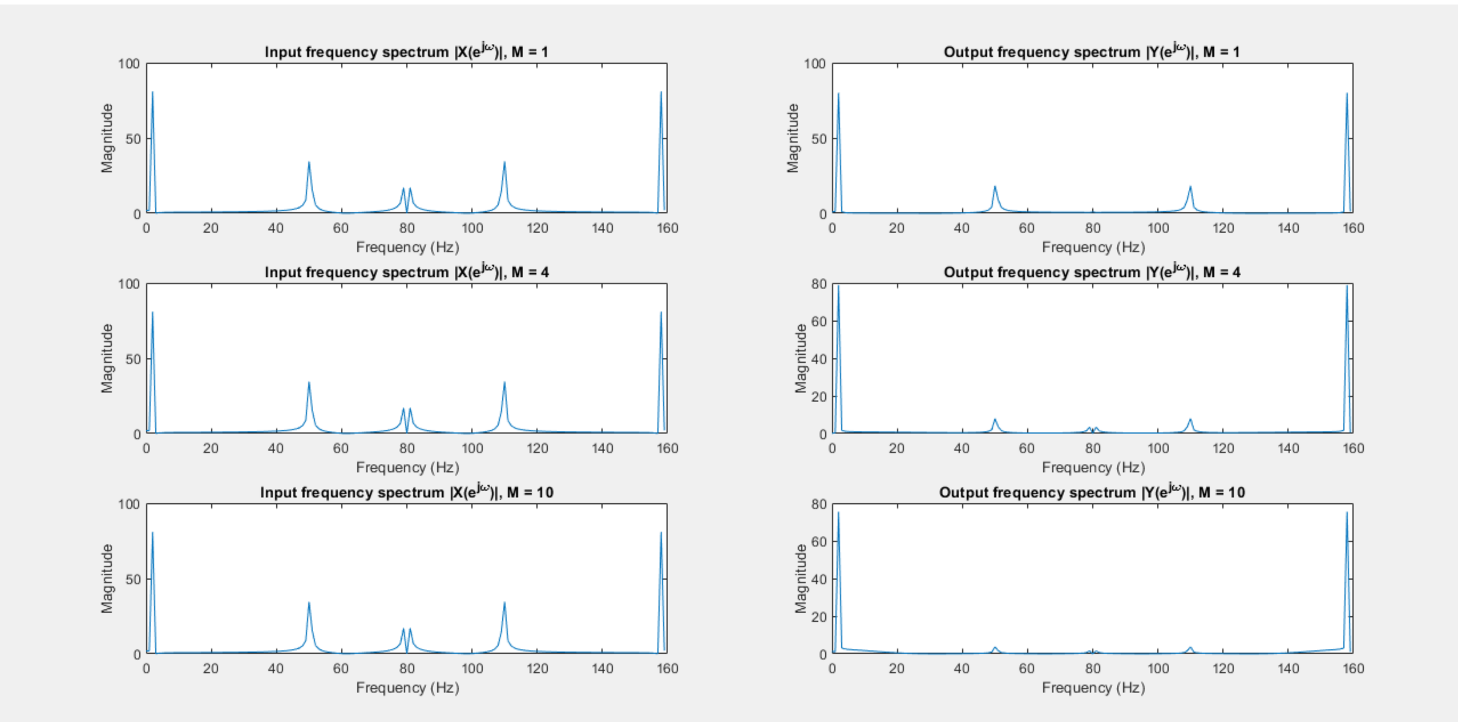
```
% Define the continuous time signal
t = linspace(0, 1, 160); % time vector from 0 to 1 sec with 160 samples
x = cos(2*pi*2*t) + 0.5*cos(2*pi*50*t) + 0.25*cos(2*pi*80*t); % signal definition

% Define the filter and filter the signal
M = [1 4 10]; % different window sizes
for i = 1:length(M)
    h = ones(1, M(i) + 1)/(M(i) + 1); % moving average filter
    y = conv(x, h); % filter the signal
    n = 0:length(y) - 1; % discrete time vector
    subplot(length(M), 1, i);
    stem(n, y); % plot the output sequence
    xlabel('Sample index (n)');
    ylabel('Amplitude');
    title(['Moving Average Filter Output Sequence (M = ' num2str(M(i)) ')']);
end
```



d) Plot the input signal frequency spectrum |X(e^jw)| and the output frequency spectrum |Y(e^jw)| .

```
% Define the input signal x[n]
Fs = 160;
t = linspace(0, 1, Fs);
x = cos(2*pi*2*t) + 0.5*cos(2*pi*50*t) + 0.25*cos(2*pi*80*t);
% Define the window sizes
M = [1, 4, 10];
% Plot the frequency spectra for each window size
for i = 1:length(M)
    % Define the moving average filter impulse response
    h = ones(1, M(i)+1) / (M(i)+1);
    % Convolve x[n] with h[n]
    y = conv(x, h, 'same');
    % Calculate the DFT of x[n] and y[n]
    X = fft(x);
    Y = fft(y);
    % Calculate the frequency axis
    f = (0:length(X)-1) * Fs / length(X);
    % Plot the frequency spectra
    subplot(length(M), 2, 2*i-1);
    plot(f, abs(X));
    xlabel('Frequency (Hz)');
    ylabel('Magnitude');
    title(['Input frequency spectrum |X(e^{j\omega})|, M = ', num2str(M(i))]);
    subplot(length(M), 2, 2*i);
    plot(f, abs(Y));
    xlabel('Frequency (Hz)');
    ylabel('Magnitude');
    title(['Output frequency spectrum |Y(e^{j\omega})|, M = ', num2str(M(i))]);
end
```



e) Find the optimum window size (M) to obtain the first sinusoidal signal (cos(2π .2.t)).

```
% Define the input signal x[n]
Fs = 160;
t = linspace(0, 1, Fs);
x = cos(2*pi*2*t) + 0.5*cos(2*pi*50*t) + 0.25*cos(2*pi*80*t);

% Define the moving average filter impulse response
M = 15;
h = ones(1, M+1) / (M+1);

% Compute the frequency responses of X(e) and Y(e)
N = length(x);
X = fft(x);
Y = fft(conv(x, h, 'same'));
f = Fs*(0:(N/2))/N;

% Plot the magnitude spectra of X(e) and Y(e)
figure;
subplot(2,1,1);
plot(f, abs(X(1:N/2+1)));
xlabel('Frequency (Hz)');
ylabel('|X(e)|');
title('Input signal frequency spectrum');
subplot(2,1,2);
plot(f, abs(Y(1:N/2+1)));
xlabel('Frequency (Hz)');
ylabel('|Y(e)|');
title(['Output signal frequency spectrum, M = ', num2str(M)]);
```