C0E 472: DIGITAL SIGNAL PROCESSING

Lab 08: Frequency Response: Bandpass and Nulling Filters

INDEX NUMBER: 3050420

NAME: HARDY HAWA TUNTEIYA

COURSE: BSc. Biomedical Engineering

Year 4

INTRODUCTION

The goal of this lab is to study the response of FIR filters to inputs such as complex exponentials and sinusoids.

This lab teaches on the bandpass filters and nulling filters. Bandpass filters can be used to detect and extract information from sinusoidal signals, e.g., tones in a touch-tone telephone dialer. Nulling filters can be used to remove sinusoidal interference, e.g., jamming signals in a radar.

Functions Involved

The functions used in this lab are firfilt(), or conv()

LAB EXERCISE

1 .Nulling Filters for Rejection

(a) Designing Two FIR Nulling Filters

We want to create filters that eliminate signals with frequencies of 0.44π and 0.7π. The general form of a nulling filter is:

$$y[n] = x[n] - 2cos(\omega_n)x[n-1] + x[n-2]$$

When $\omega_n = 0.44\pi$; the coefficients are:

b_0 = 1;   b_1 = -2cos(0.44 π) ;   b_2 = 1;

When $\omega_n = 0.7\pi$; the coefficients are:

b_0 = 1;   b_1 = -2cos(0.7 π) ;   b_2 = 1;

(b) Generate an Input Signal (x[n]=5cos(0.3πn)+22cos(0.44πn−π/3)+22cos(0.7πn−π/4))

```
n = 0:149;
x = 5*cos(0.3*pi*n) + 22*cos(0.44*pi*n - pi/3) + 22*cos(0.7*pi*n - pi/4);
```

(c) Apply the two nulling filters in cascade to the input signal.

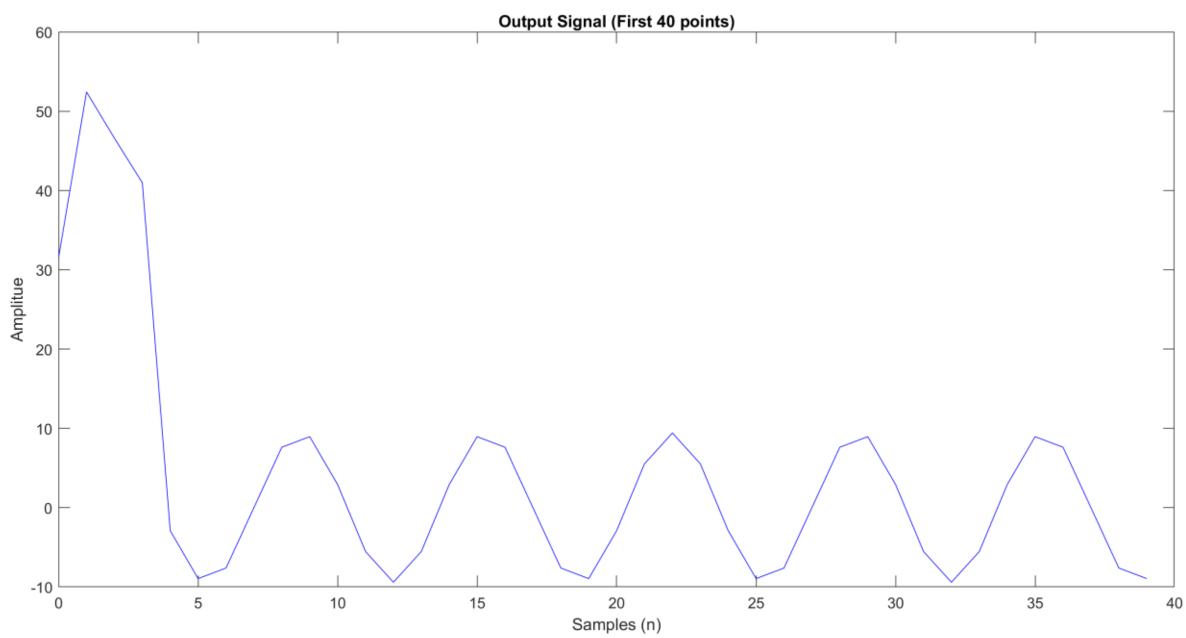
```
%c. Apply the two nulling filters in cascade
b1 = [1, -2*cos(0.44*pi), 1]; % Coefficients for 0.44π nulling filter
b2 = [1, -2*cos(0.7*pi), 1]; % Coefficients for 0.7π nulling filter
y1 = firfilt(b1, x);
```

```
y2 = firfilt(b2, y1);
```

(d) Plot the First 40 Points of The Output Signal

```
% Plot first 40 points
figure;
plot(n(1:40), y2(1:40), 'b');

title('Output Signal (First 40 points)');
xlabel('n');
ylabel('y[n]');
```

OUTPUT



(e) Mathematical Formula for Output Signal

The original signal was $x[n]=5\cos(0.3\pi n)+22\cos(0.44\pi n-\pi/3)+22\cos(0.7\pi n-\pi/4)$

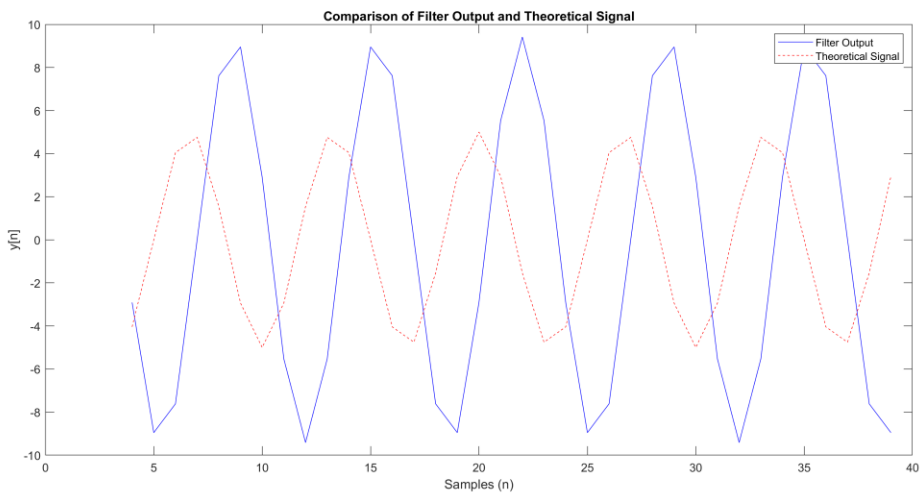
The nulling filter removed the 2nd and 3rd components from the original signal since the contain 0.7π and 0.44π. Thus

Output Signal, $y[n] \approx 5\cos(0.3\pi n)$

Plot this theoretical signal alongside the filtered output.

```
%e. Plot theoretical Signal and Compare with Output signal
y_theoretical = 5*cos(0.3*pi*n);
figure;
plot(n(5:40), y2(5:40), 'b', n(5:40), y_theoretical(5:40), 'r--');
title('Comparison of Filter Output and Theoretical Signal');
xlabel('Samples (n)');
ylabel('y[n]');
legend('Filter Output', 'Theoretical Signal');
```

OUTPUT



(f) Explain why the output signal is different for the first few points. How many "start-up" points are found. How is this number related to the lengths of the filters designed in part

When you cascade two length-3 FIR filters, the initial few points (4 points) of the output signal are influenced by the start-up transients because the filters are still accumulating enough past data points to function correctly. After these initial 4 points, the filters have enough data, and the output stabilizes to accurately reflect the filtered signal.

The number of start-up points is related to the lengths of the filters. Since each filter has 3 coefficients, the total start-up length is to(3-1)+(3-1)=4 points.

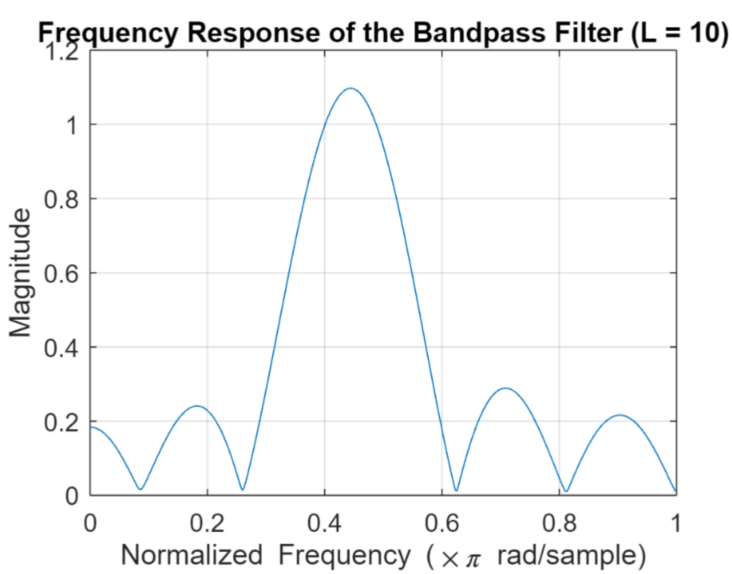
3.2 Simple Bandpass Filter Design

The formula for creating a simple band_pass: $h[n] = \frac{2}{L}\cos(\hat{\omega}_c n), \qquad 0 \le n < L$

(a) Generate a Bandpass Filter

```
L = 10; % The filter length
wc = 0.44 * pi; % Center Frequency

% Generate the bandpass filter coefficients using the formula
h = (2 / L) * cos(wc * (0:(L-1)));
% Frequency response
[H, W] = freqz(h, 1, 512);

% Plot the frequency response
figure;
plot(W/pi, abs(H));
title('Frequency Response of the Bandpass Filter (L = 10)');
xlabel('Normalized Frequency (\times\pi rad/sample)');
ylabel('Magnitude');
grid on;

% Measure the gain at specific frequencies
freqs = [0.3, 0.44, 0.7] * pi;
gains = abs(freqz(h, 1, freqs));
disp('Gains at specified frequencies:');
disp(['Frequency 0.3\pi: ', num2str(gains(1))]);
disp(['Frequency 0.44\pi: ', num2str(gains(2))]);
disp(['Frequency 0.7\pi: ', num2str(gains(3))]);
```

OUTPUT



Gains at specified frequencies:

Frequency 0.3\pi: 0.28362

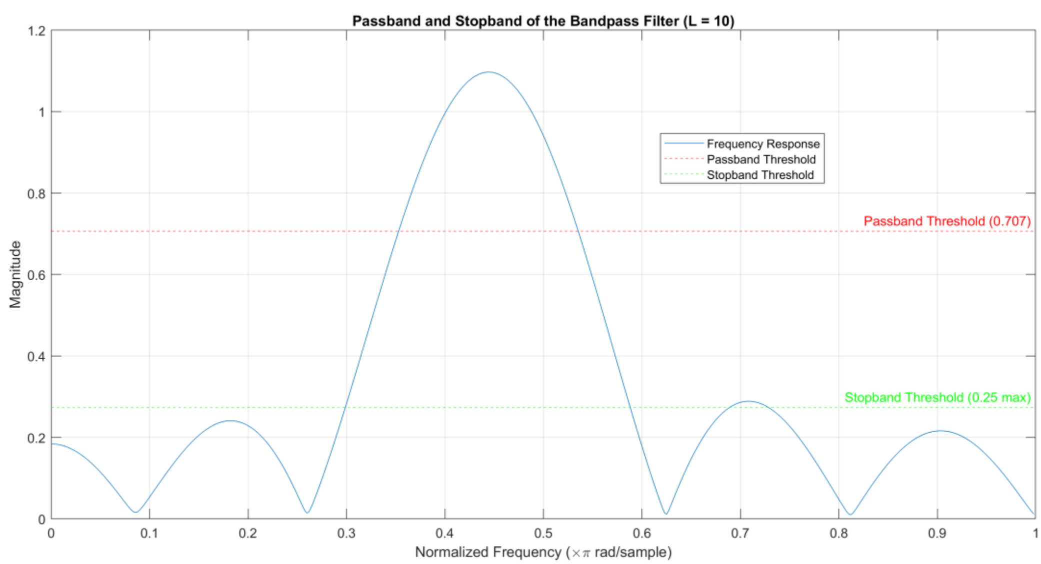
Frequency 0.44\pi: 1.0961

Frequency 0.7\pi: 0.28614

(b) Determine the passband width (where the response is greater than 0.707 of the maximum) for L = 10, 20, and 40.

```
% Passband and Stopband determination
passband_threshold = 1/sqrt(2);
stopband_threshold = 0.25 * max(abs(H));

% Plot with thresholds
figure;
plot(W/pi, abs(H));
hold on;
yline(passband_threshold, 'r--', 'Passband Threshold (0.707)');
yline(stopband_threshold, 'g--', 'Stopband Threshold (0.25 max)');
title('Passband and Stopband of the Bandpass Filter (L = 10)');
xlabel('Normalized Frequency (\times\pi rad/sample)');
ylabel('Magnitude');
legend('Frequency Response', 'Passband Threshold', 'Stopband Threshold');
grid on;
```

OUTPUT



(c) Selectivity of L = 10 Bandpass Filter

The L = 10 bandpass filter has a peak at 0.44π and attenuates other frequencies. The passband is centered around 0.44π, and the filter reduces components at 0.3π and 0.7π significantly.

Explanation Based on the Frequency Plot

On the graph, when the frequency is around ω = 0.3π, the gain at this frequency is low, indicating that the filter attenuates this frequency component.

When the frequency is around ω = 0.44π, the gain at this frequency is high, close to 1, indicating that the filter allows this to pass through fully, with minimal attenuation.

When the frequency is around ω = 0.7π, the gain at this frequency is low, indicating that the filter attenuates this frequency.

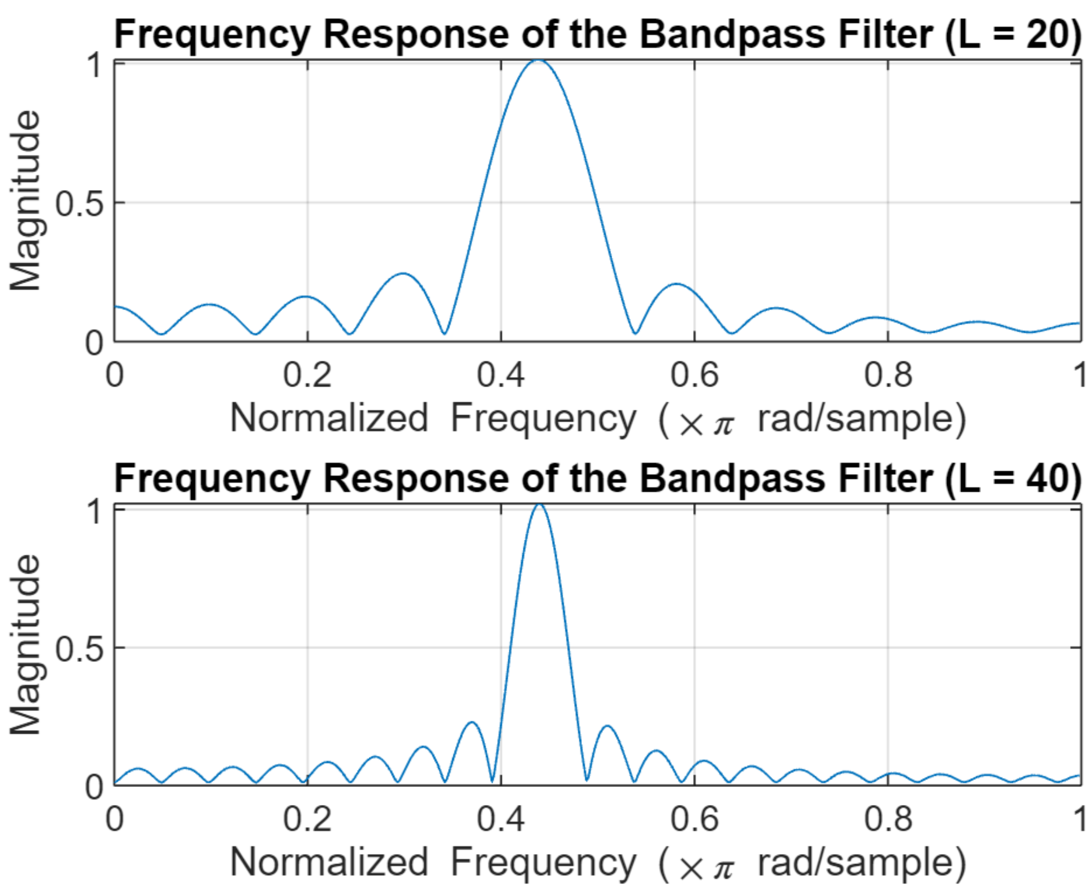
(d) Increase Filter Length for Better Reduction

Increase L until the filter reduces frequencies at 0.3π and 0.7π by a factor of 10.

```
% Define longer filter lengths
L1 = 20;
L2 = 40;

% Generate filters using the cosine method
h1 = (2 / L1) * cos(wc * (0:(L1-1)));
h2 = (2 / L2) * cos(wc * (0:(L2-1)));


% Frequency responses
[H1, W1] = freqz(h1, 1, 512);
[H2, W2] = freqz(h2, 1, 512);


% Plot the frequency responses
figure;
subplot(2,1,1);
plot(W1/pi, abs(H1));
title('Frequency Response of the Bandpass Filter (L = 20)');
xlabel('Normalized Frequency (\times\pi rad/sample)');
ylabel('Magnitude');
grid on;


subplot(2,1,2);
```

```
plot(W2/pi, abs(H2));
title('Frequency Response of the Bandpass Filter (L = 40)');
xlabel('Normalized Frequency (\times\pi rad/sample)');
ylabel('Magnitude');
grid on;
```

OUTPUT



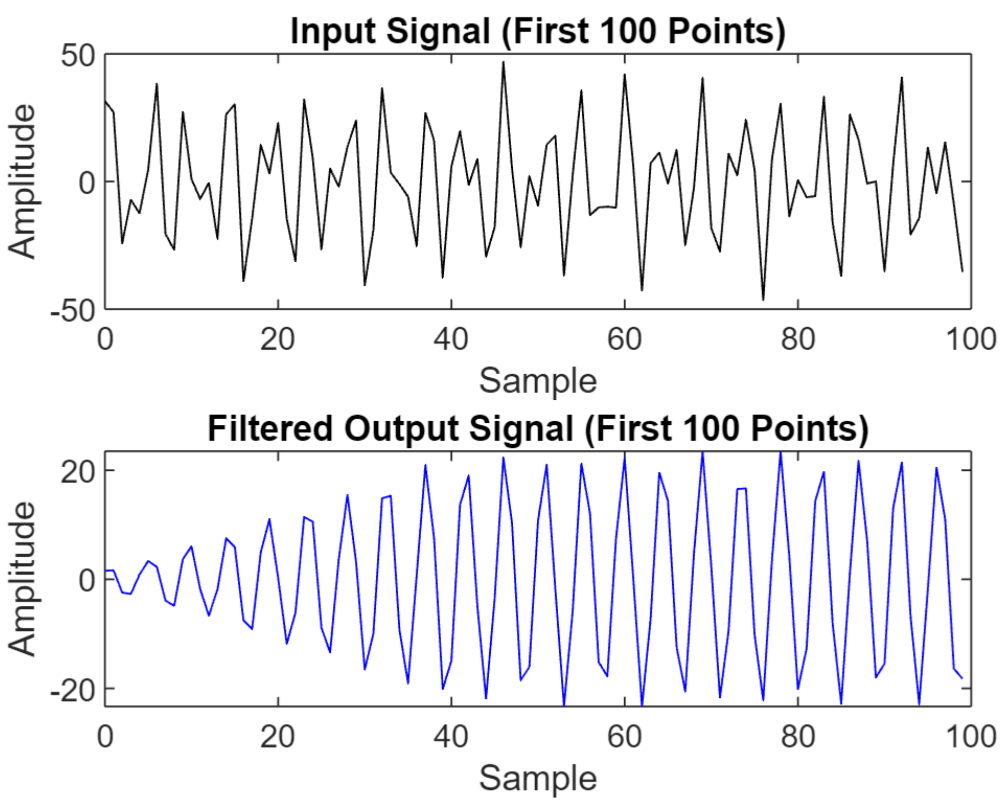
(e) Filter the Sum of 3 Sinusoids

Use the filter to process the input signal
$x[n]=5\cos(0.3\pi n)+22\cos(0.44\pi n-\pi/3)+22\cos(0.7\pi n-\pi/4)$

```
% Generate the input signal
n = 0:149;
```

```
x = 5*cos(0.3*pi*n) + 22*cos(0.44*pi*n - pi/3) + 22*cos(0.7*pi*n - pi/4);

% Filter the signal using L = 40 filter (assuming this is the best length)
y_filtered = firfilt(h2, x);

% Plot input and output signals (first 100 points)
figure;
subplot(2,1,1);
plot(n(1:100), x(1:100), 'k')

title('Input Signal (First 100 Points)');
xlabel('Sample');
ylabel('Amplitude');


subplot(2,1,2);
plot(n(1:100), y_filtered(1:100), 'b');

title('Filtered Output Signal (First 100 Points)');
xlabel('Sample');
ylabel('Amplitude');
```

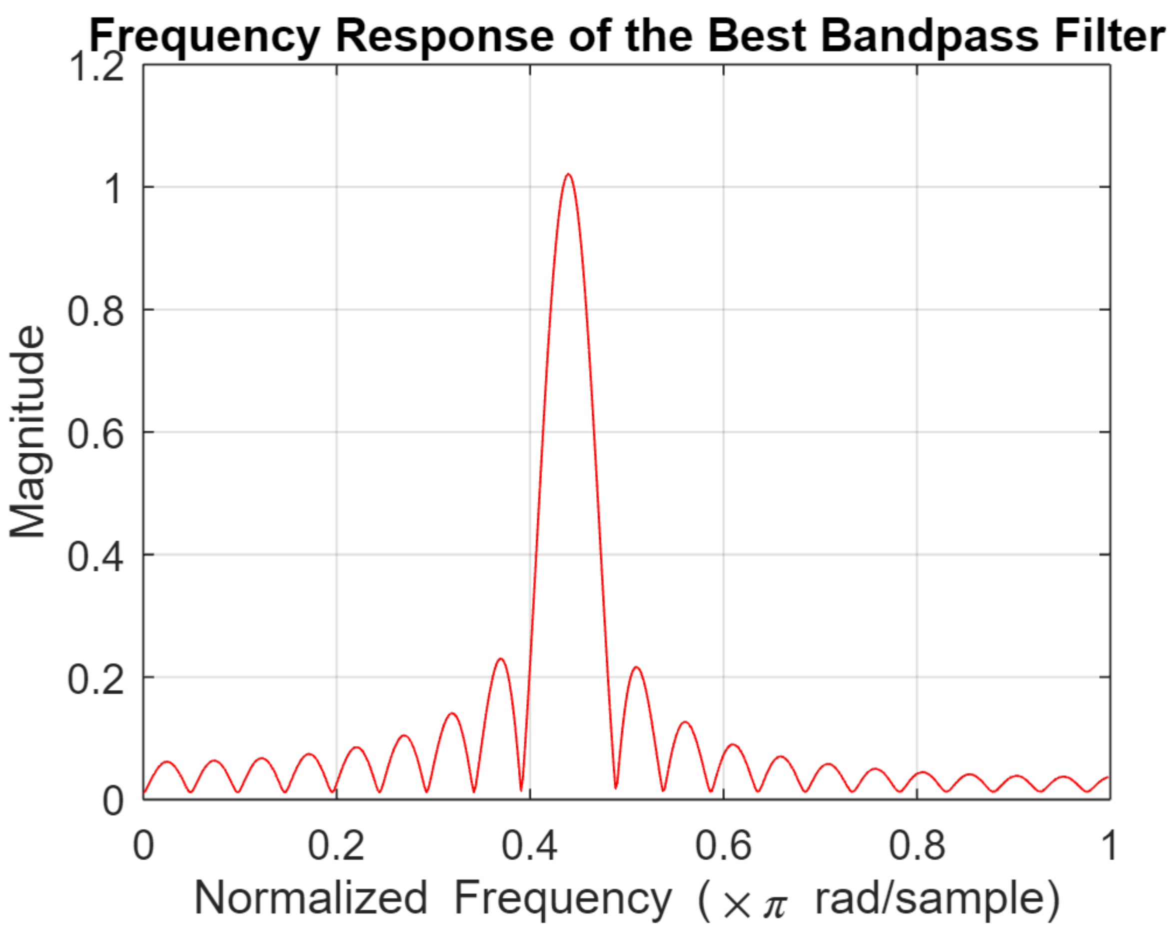
OUTPUT



(f) Plot Frequency Response and Explain

```
% Plot frequency response of the filter from part (d)
figure;
plot(W2/pi, abs(H2), 'r');
title('Frequency Response of the Best Bandpass Filter');
xlabel('Normalized Frequency (\times\pi rad/sample)');
ylabel('Magnitude');
grid on;
```

OUTPUT



Explanation

The frequency response H(ejω) of a filter shows how the filter affects frequency components of the input signal.

The magnitude |H(ejω)| states how much the filter amplifies or reduces each frequency. In our bandpass filter centered at ω=0.44π, this frequency passes through with little change, while frequencies at $\omega = 0.3\pi$ and $\omega=0.7\pi$ are greatly reduced. This shows the filter's ability to selectively pass certain frequencies and block others.