AFIT GRADUATE SCHOOL OF ENGINEERING AND MANAGEMENT
QUALIFYING EXAM QUESTION

Mr. Tim Coon

(Part 1: Covering topics from Vibration Damping and Control)

Oct 2014

Format: Take Home (open text, open notes, & Matlab)

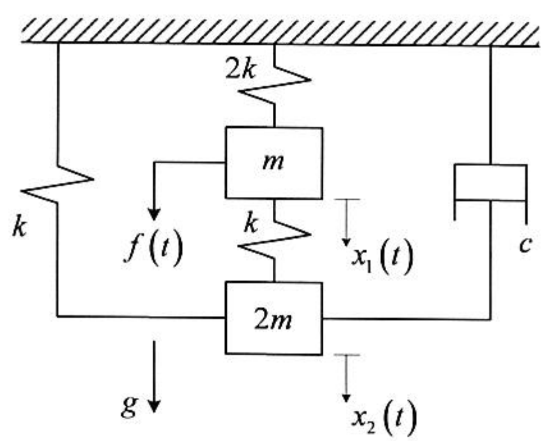
Duration: 48 hours max

Exam will be graded pass/fail. Be sure to answer all questions.

(Independent work only—do not discuss questions or solutions with anyone)

Analysis of Experimental Data

Experimental data for a spring-mass system shown below was measured in the lab. The data consisted of sensors located on mass 1 and mass 2, with a random input applied at mass 1 as shown. The lab technician carefully measured the two masses, found to 1 kg and 2 kg respectively, but lost his notes on the other values of c and k. The technician was sure that four identical springs were used, and so the designated values of (k, k, and 2k) are correct, to an unknown constant. (note: 2 springs were connected in parallel between mass 1 and the ceiling, resulting in 2k) It was unknown as to whether the measurements (x_1 and x_2) were displacements, velocity, or acceleration. In fact, the two measurements may be two different types (i.e., one displacement and one acceleration).



Question: Using the data provided (via CD), determine the correct values of c and k, answering all the questions below.

Note: The data was sampled at 5 samples per second. The first column represents the input f(t), and the second and third represent the measurements at mass 1 and 2 respectively. The recorded data consists of 10,000 data samples, but the initial condition (when the data recording was initiated) is unknown. As part of your conclusions, determine which types of measurements were made (displacements, velocity, or acceleration) and justify your claim. Include in your

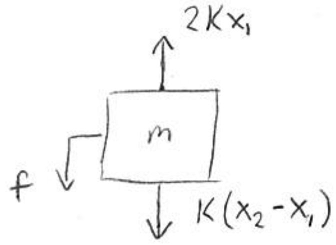
△ =

results, the computed FRFs, and the coherence functions. Also include the measured mode shapes. Explain how you processed the data, and how you derived your values of k and c.

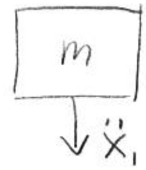
Be sure to answer the following questions (providing justification for each response):

1. What is the significance of the unknown initial condition? How did this affect the way you processed the data?
2. What is the value of k?
3. What is the value of c?
4. What type of measurement is x_1 ? (displacements, velocity, or acceleration)
5. What type of measurement is x_2 ? (displacements, velocity, or acceleration)
6. Was 5 samples per second adequate to measure the dynamics of this system or should a slower or faster sample rate be used? (explain in detail).
7. Using your determined values of k and c, compare the measured mode shapes (obtained from the FRFs) to the theoretical model.

FBD   MAD

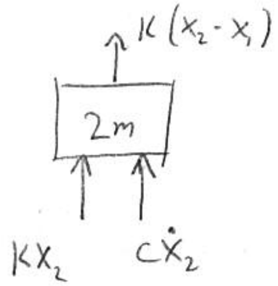


=

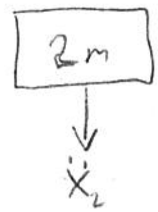


$$\Rightarrow \quad m\ddot{x}_1 = f + K(x_2 - x_1) - 2Kx_1$$

$$m\ddot{x}_1 + 3Kx_1 - Kx_2 = f$$



=



$$\Rightarrow \quad 2m\ddot{x}_2 = -K(x_2 - x_1) - Kx_2 - c\dot{x}_2$$

$$= Kx_1 - 2Kx_2 - c\dot{x}_2$$

$$2m\ddot{x}_2 + c\dot{x}_2 - Kx_1 + 2Kx_2 = 0$$

1st ORDER

$$z_1 = x_1 \qquad \dot{z}_1 = z_2$$

$$z_2 = \dot{x}_1 \qquad \dot{z}_2 = \frac{1}{m}\left[f - 3Kz_1 + Kz_3\right]$$

$$z_3 = x_2 \qquad \dot{z}_3 = z_4$$

$$z_4 = \dot{x}_2 \qquad \dot{z}_4 = \frac{1}{2m}\left[-cz_4 + Kz_1 - 2Kz_3\right]$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{3K}{m} & 0 & \frac{K}{m} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{K}{2m} & 0 & -\frac{K}{m} & -\frac{c}{2m} \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ 1/m \\ 0 \\ 0 \end{bmatrix}$$

2nd ORDER

$$\begin{bmatrix} m & 0 \\ 0 & 2m \end{bmatrix} \ddot{\underline{x}} + \begin{bmatrix} 0 & 0 \\ 0 & c \end{bmatrix} \dot{\underline{x}} + \begin{bmatrix} 3K & -K \\ -K & 2K \end{bmatrix} \underline{x} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} f$$

$$M\ddot{\underline{x}} + C\dot{\underline{x}} + K\underline{x} = Ff$$

Tim Coon

MECH719 Qual Exam Question Responses

1. What is the significance of the unknown initial condition? How did this affect the way you processed data?

   The unknown initial condition is significant to producing a time-domain simulation. All of the questions asked are answered using frequency-domain analysis, thus, the unknown initial condition did not affect the processing of data.

2. What is the value of k? $k = 10$ N/m

   The state-space model of the system was derived and the FRF was simulated using a guessed value for k ($c = 0$). The value of k was then adjusted manually until the resonant peaks of the measured data and the simulation lined up at the same frequencies. The resonant frequencies are close to the natural frequencies and the natural frequency is solely dependent on k because the mass is fixed and known.

3. What is the value of c? $c = 0.6$ N-s/m

   Using the state-space model and $k = 10$ N/m, the FRF was simulated and compared to the measured data. The value of c was adjusted until the slopes and heights of the peaks were similar. It is possible to use a log-decrement or half-power method to find the damping ratio from the measured data FRF and then calculate the damping coefficient.

4. What type of measurement is x_1?

   Measurement, x_1, is the velocity of mass one. This is evident by comparison of the simulated FRF for velocity output and the measured data FRF. The figure shows simulated position, velocity, and acceleration transfer functions for complete comparison.

5. What type of measurement is x_2?

   Measurement, x_2, is the position of mass two. This is evident by comparison of the simulated FRF for velocity output and the measured data FRF. Notice that the phase of the measured data jumps 360 degrees at 0.5 Hz, but this is equivalent.

6. Was five samples per second adequate to measure the dynamics of this system or should a slower or faster sample rate be used?

   Five samples per second was an adequate rate. The nyquist frequency is 2.5 Hz and the second mode occurs around 0.9 Hz, so there is no aliasing effect. A slightly slower sample rate could be used if necessary. The duration of the sampling allowed for the separation into two sets for averaging to improve appearance of the FRFs. A running average could also be implemented and might reduce the need for a longer sampling time.

7. Using your determined values of k and c, compare the measured mode shapes (obtained from the FRFs) to the theoretical mode shapes?

   Mode shapes of simulated second-order model are the eigenvectors. These are plotted and show the first mode shape with both masses displaced in the same direction and the second mode shape with the masses displaced in opposite directions.

   The measured data came as one velocity and one position measurement. Velocity is integrated in the frequency domain by dividing $j\omega_i$ at each frequency and the position FRF is readily obtained from the measured velocities. Mode shapes are determined using the magnitude of the FRF at the resonant frequencies modified by the sign of the cosine of the phase at the respective frequencies. Mode shape 1 shows the masses moving in-phase while mode shape 2 shows them out-of-phase, as expected from the theoretical model. The relative sizes of the eigenvector elements are similar (see below) and the marginal deviation is attributed to imperfect data collection.

$$V1_{calc} = \begin{matrix} 1.0 \\ 2.2 \end{matrix} \qquad V2_{calc} = \begin{matrix} 1.0 \\ -0.2 \end{matrix}$$

$$V1_{meas} = \begin{matrix} 1.0 \\ 1.9 \end{matrix} \qquad V2_{meas} = \begin{matrix} 1.0 \\ -0.2 \end{matrix}$$

```
% Tim Coon 12 January 2015
% Qualifying Exam Question #1
% Covering Material from MECH 719, Vibration Damping and Control
% Advisor: Dr. Richard Cobb
clear; close all; clc;
scrsz = get(0,'ScreenSize');
scrwidth = scrsz(3);
scrheight = scrsz(4);

%% load the experimental data
load('MECH719_QualDATA.mat');

% split the data into multiple samples. there are 10000 data points
sample_interval = 2^11;         % use a power of two so FFT doesn't pad with↙
zeros
num_samples = floor(length(data)/sample_interval);
for sample = 1:num_samples
    Sstart = sample_interval*(sample-1)+1;
    Send = sample_interval*sample;
    F(:,sample) = data(Sstart:Send,1);
    X(:,:,sample) = data(Sstart:Send,2:3);
    p1(:,sample) = cumtrapz(X(:,1,sample));
end

% test parameters
Fs = 5;             % (1/s) sampling frequency
dt = 1/Fs;          % (s) sample time
N = length(F);      % (-) number of samples
T = N*dt;           % (s) time duration of sample
t = (0:N-1)*dt;     % (s) time vector
% frequency vector
fnyq = 1/(2*dt);                % (Hz) nyquist frequency
df = 1/T;                       % (Hz) frequency interval
f = 0:df:fnyq/2;

%% process the Frequency Response Function and Coherence Function
% calculate the auto-PSD of the input
for sample = 1:num_samples
    % calculate the auto-PSD of the input
    Sai(:,sample) = FFT_PSD(F(:,sample),F(:,sample),T);
end
Sff = mean(Sai,2);

for out = 1:2
    for sample = 1:num_samples
        x = X(:,out,sample);
        % calculate the auto-PSD of the output
        Sao(:,sample) = FFT_PSD(x,x,T);
```

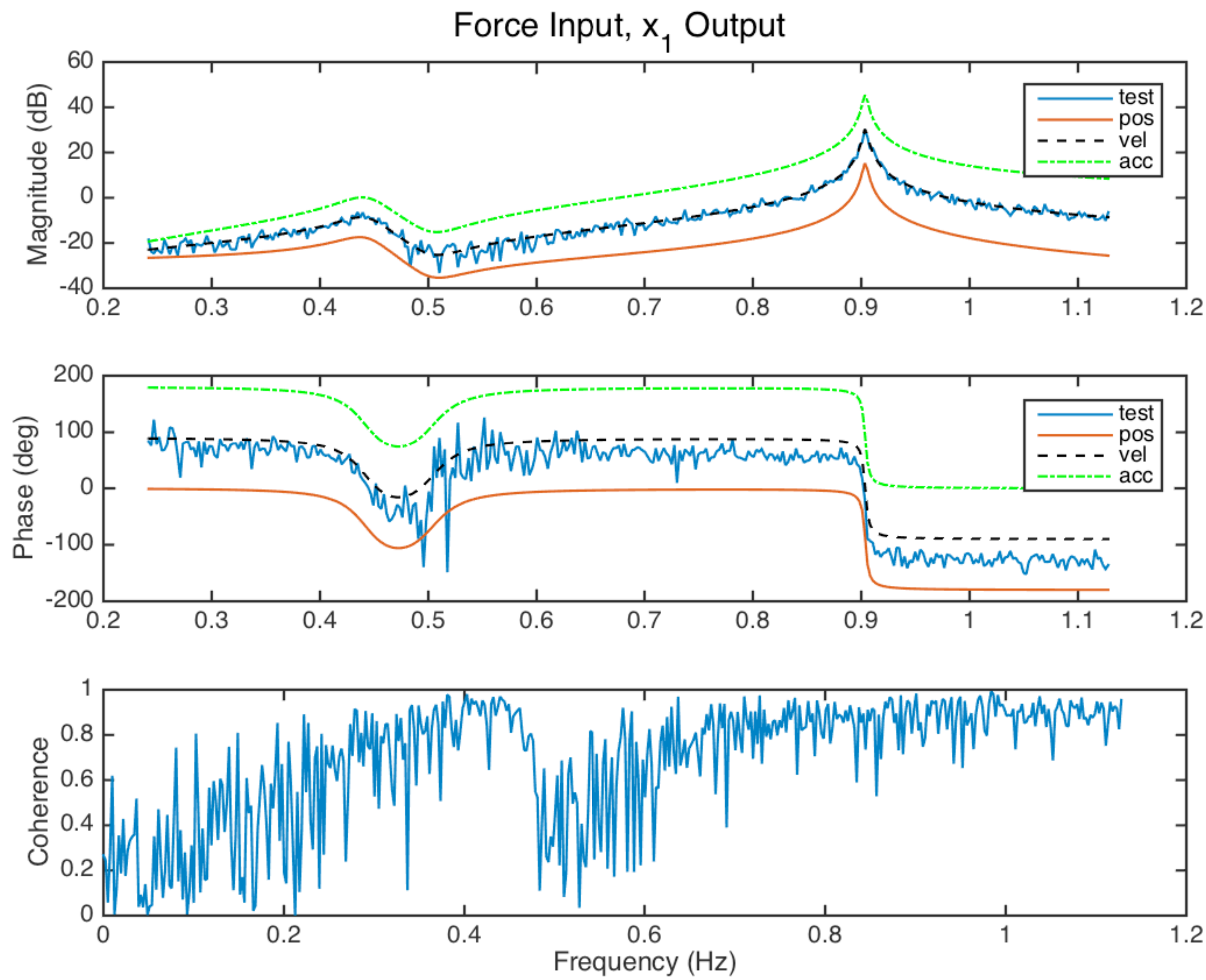
```
        % calculate the cross-PSD in/out
        Scr(:,sample) = FFT_PSD(F(:,sample),x,T);
    end
    % find average PSDs for accuracy
    Sxx(:,out) = mean(Sao,2);
    % calculate the cross-PSDs of in/outs
    Sfx(:,out) = mean(Scr,2);
    % calculate the FRF data
    H(:,out) = Sfx(:,out)./Sff;
    % calculate the Coherence
    C(:,out) = abs(Sfx(:,out)).^2./(Sff.*Sxx(:,out));
end
Phase = rad2deg(angle(H));
H_mag = 20*log10(abs(H));       % dB

%% Plot Simulated and measured FRFs
% system parameter values
m = 1;      % (kg)
c = 0.6;      % (N-s/m)
k = 10;      % (N/m)

% first-order state-space
A = [   0   1   0    0;
     -3*k/m 0  k/m   0;
        0   0   0    1;
      k/2*m 0 -k/m -c/2*m];
B = [0 1/m 0 0]';
C1 = eye(4);
D = zeros(4,1);

% calculate transfer functions
s = tf('s');
TF1 = C1*inv(s*eye(4)-A)*B;
F_p1 = TF1(1);      F_p2 = TF1(3);      % position
F_v1 = TF1(2);      F_v2 = TF1(4);      % velocity
F_a1 = TF1(2)*s;    F_a2 = TF1(4)*s;    % acceleration
w = f*(2*pi);                           % (rad/sec) frequency vector

% simulations
[magp(:,1),phasep(:,1)] = bode(F_p1,w);
[magv(:,1),phasev(:,1)] = bode(F_v1,w);
[maga(:,1),phasea(:,1)] = bode(F_a1,w);
[magp(:,2),phasep(:,2)] = bode(F_p2,w);
[magv(:,2),phasev(:,2)] = bode(F_v2,w);
[maga(:,2),phasea(:,2)] = bode(F_a2,w);
magp = 20*log10(magp); magv = 20*log10(magv); maga = 20*log10(maga); % (dB)

% overlay data Bode plots and calculated bode plots
```

```
L = min([length(f),length(H_mag)])-50;
start = 100;
pos = [0 0 scrwidth/2 scrheight; scrwidth/2 0 scrwidth/2 scrheight];
for fig = 1:2
    figure('Position',pos(fig,:))
    suptitle(titles1(fig))
    subplot(311)
    plot(f(start:L),H_mag(start:L,fig))
    hold on
    plot(f(start:L),magp(start:L,fig))
    plot(f(start:L),magv(start:L,fig),'k--')
    plot(f(start:L),maga(start:L,fig),'g-.')
    ylabel('Magnitude (dB)')
    hold off
    legend('test','pos','vel','acc')
    % phase plots
    subplot(312)
    plot(f(start:L),Phase(start:L,fig))
    hold on
    plot(f(start:L),phasep(start:L,fig))
    plot(f(start:L),phasev(start:L,fig),'k--')
    plot(f(start:L),phasea(start:L,fig),'g-.')
    ylabel('Phase (deg)')
    hold off
    legend('test','pos','vel','acc')
    % coherence plots
    subplot(313)
    plot(f(start:L),C(start:L,fig))
    xlabel('Frequency (Hz)'); ylabel('Coherence');
end

%% Compare measured mode shapes (from FRFs) to theoretical mode shapes
% to find the theoretical mode shapes, use the second-order system
M = [m 0; 0 2*m];
C = [0 0; 0 c];
K = [3*k -k; -k 2*k];
fm = [1; 0];
[V,D] = eig(K,M);
V1_ind = find(max(V(:,1)));
V2_ind = find(max(V(:,2)));
eVector1_calc = V(:,1)/V(V1_ind,1)
eVector2_calc = V(:,2)/V(V2_ind,2)


%% Output data for Ezera
FreqV = f(start:L);
frf = H(start:L,:);
save('MECH791_Qual_EZERA_DATA.mat','FreqV','frf')
```

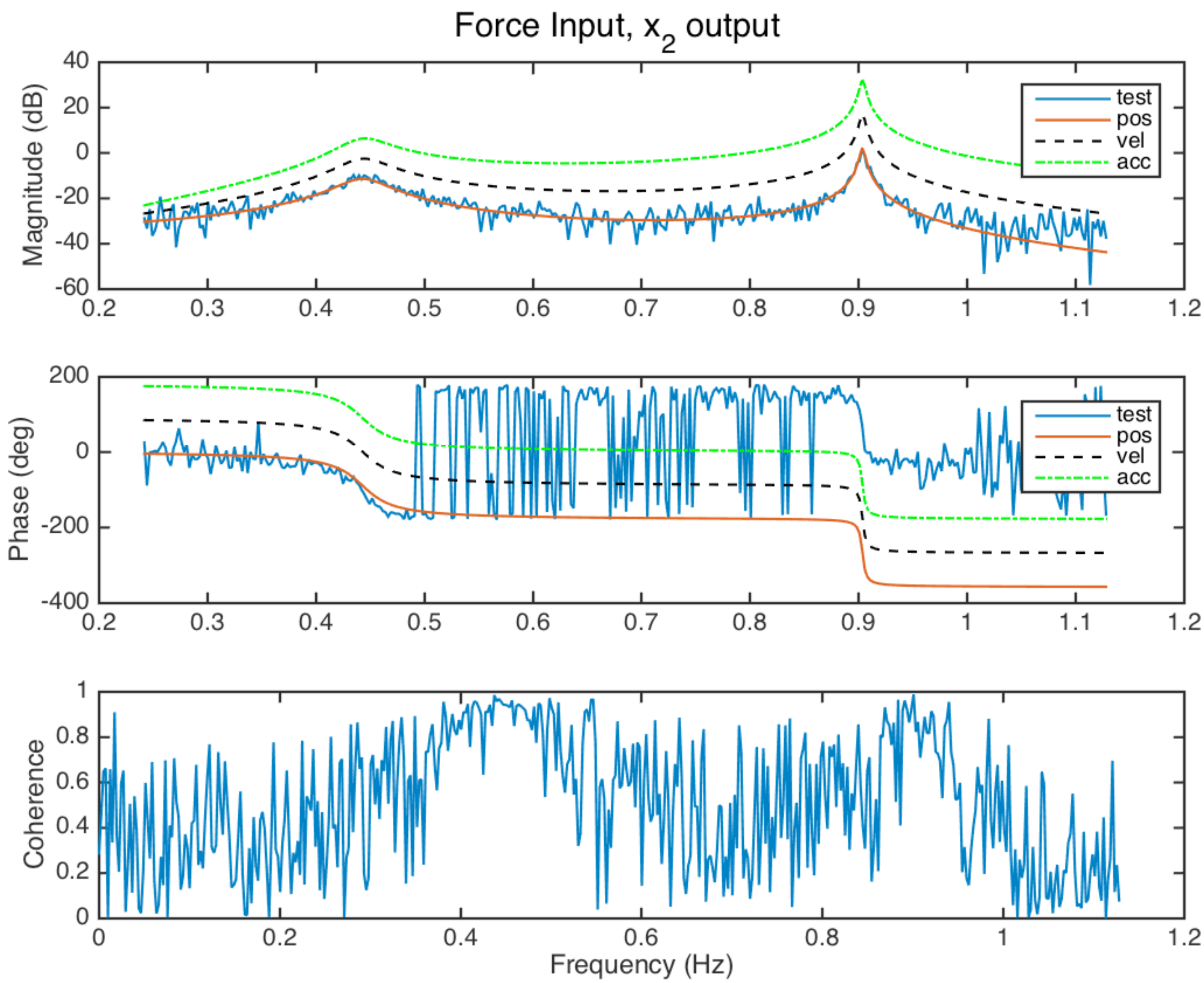
```
%% Plot mode shapes from measured data

H = H(start:L,:);
f = f(start:L);

Hp1 = H(:,1)./(1i*2*pi*f');     % integrate data in frequency domain
Hp1_mag = abs(Hp1);
Hp2_mag = abs(H(:,2));

figure()
plot(f,Hp1_mag,f,Hp2_mag)
title('FRFs')
legend('Mass #1 Pos','Mass #2 Pos');
ylabel('Absolute Magnitude (dB)'); xlabel('Frequency (Hz)');

% measured data from plots
w_r1 = 0.437;      % (Hz)
Hmag_p1r1 = 0.1708;    % (-) NOT dB
Hmag_p2r1 = 0.3286;
w_r2 = 0.9033;
Hmag_p1r2 = 5.818;
Hmag_p2r2 = 1.283;

% % extract values automatically (not so great)
% r1_ind = find(abs(f-0.4370) < df/2);
% r2_ind = find(abs(f-0.9033) < df/2);
% Hmag_p1r1 = Hp1_mag(r1_ind);
% Hmag_p2r1 = Hp2_mag(r1_ind);
% Hmag_p1r2 = Hp1_mag(r2_ind);
% Hmag_p2r2 = Hp2_mag(r2_ind);

% to determine the sign, use cosd(phaseangle)
% use phase angles from analytical FRF plots
S_p1r1 = sign(cosd(-55.61));
S_p2r1 = sign(cosd(-74.12));
S_p1r2 = sign(cosd(-82.1));
S_p2r2 = sign(cosd(-257.7));

Vm =[S_p1r1*Hmag_p1r1 S_p1r2*Hmag_p1r2;
     S_p2r1*Hmag_p2r1 S_p2r2*Hmag_p2r2];

Vm1_ind = find(max(Vm(:,1)));
Vm2_ind = find(max(Vm(:,2)));
eVector1_meas = Vm(:,1)/Vm(Vm1_ind,1)
eVector2_meas = Vm(:,2)/Vm(Vm2_ind,2)
```

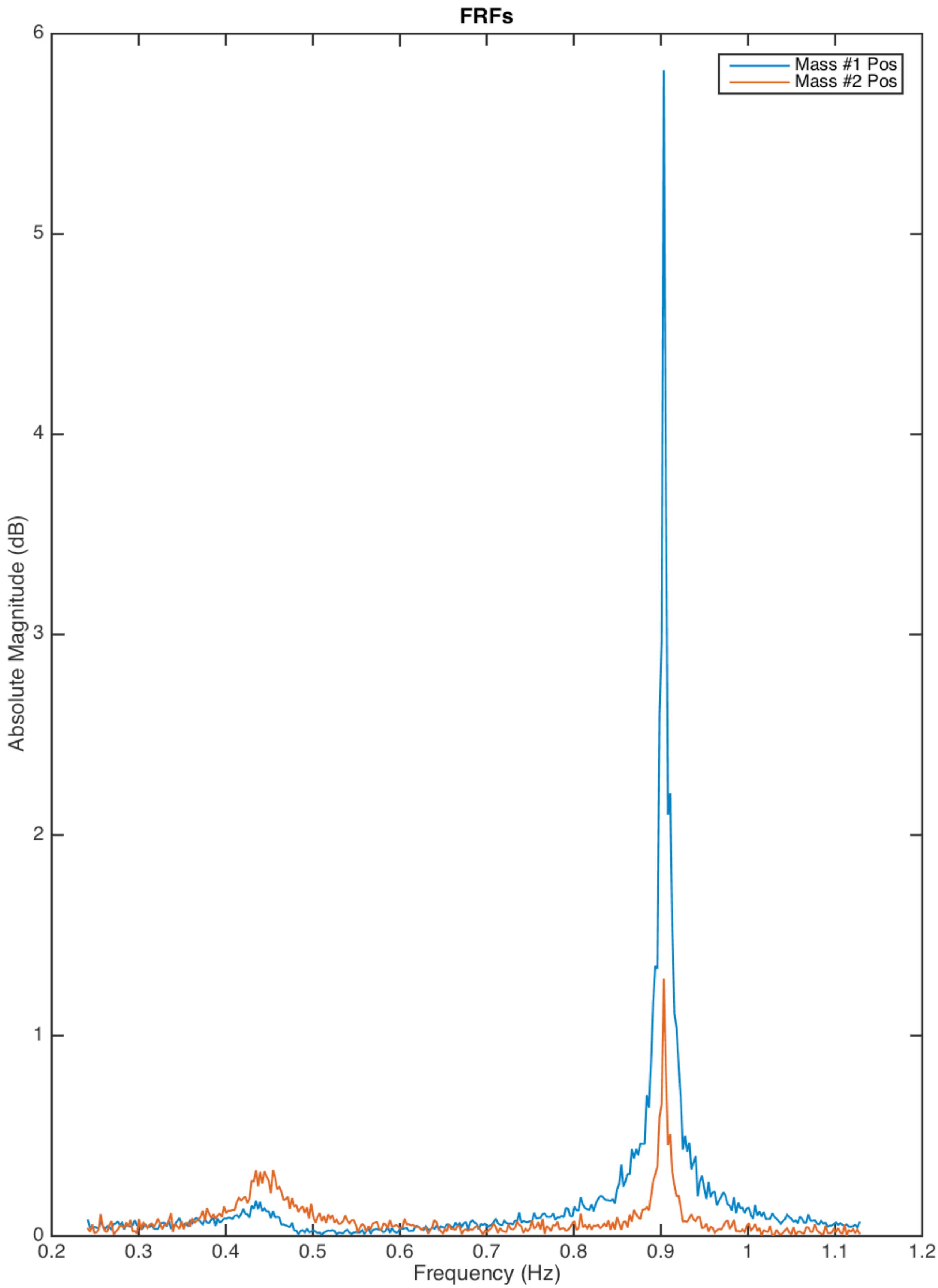
```
function [ S ] = FFT_PSD( x,y,T )
%FFT_PSD computes the PSD of a set of sampled data utilizing fft()
%   x = input
%   y = output


xdft = fft(x);
xdft = xdft(1:length(x)/2);           % "folding" causes mirror image
xdft(2:end-1) = 2*xdft(2:end-1);     % account for magnitude of "de-folding"

ydft = fft(y);
ydft = ydft(1:length(y)/2);
ydft(2:end-1) = 2*ydft(2:end-1);

S = (1/T)*conj(xdft).*ydft;

end
```

```
function [ T ] = titles1( t )
%TITLES this function will allow plot titles to be set in a loop
%   have to use cellstr to make a string an array entry

Tarray = cellstr([  'Force Input, x_1 Output'; ...
                    'Force Input, x_2 output';      ]);

T = Tarray(t);
end
```



Force Input, x_1 Output
Magnitude (dB)
test
pos
vel
acc
Phase (deg)
Coherence
Frequency (Hz)


Force Input, x_2 output
Magnitude (dB)
test
pos
vel
acc
Phase (deg)
test
pos
vel
acc
Coherence
Frequency (Hz)


FRFs
Mass #1 Pos
Mass #2 Pos
Absolute Magnitude (dB)
Frequency (Hz)
0
1
2
3
4
5
6
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1.1
1.2