ME 494/5594 – Robotics Systems Identification

HW #5: Due Wednesday, October 5th, 2022

For this assignment you will once again be using the pendulum simulation; however, this time around there will be **NO** motor controller. You will be estimating the equation of motion of the pendulum through exciting the system (with the motor) with good input design.

The pendulum function has been updated to allow simulation with and without an active controller.

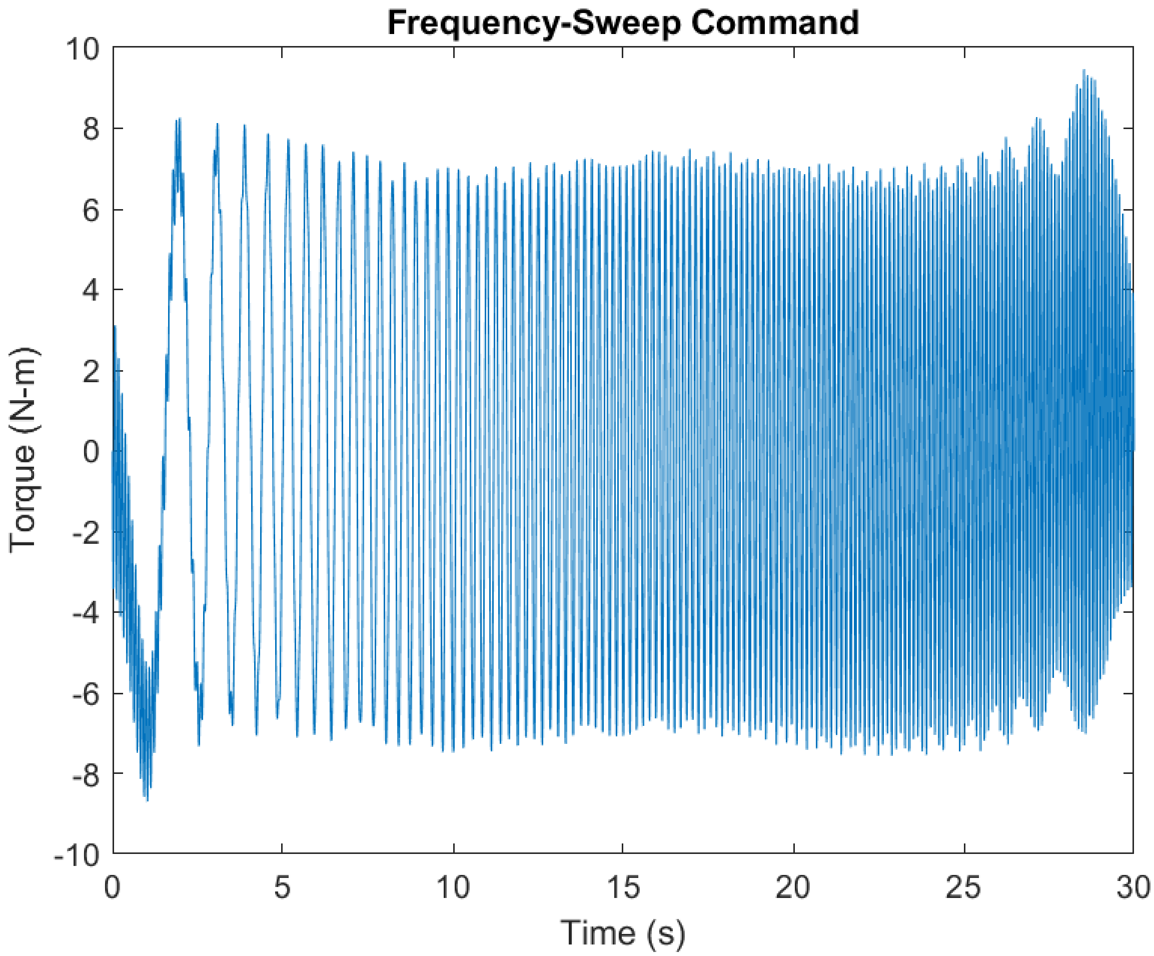
```
[y,yd, ydd, u_cmd, t] = pend(ydes, dt, tfinal,control_on)
```

If control_on = 1, then the PID controller is active and ydes is the desired angle. If control_on = 0, then ydes is the motor command (u or u_cmd) directly. For this assignment control_on should be set to 0.

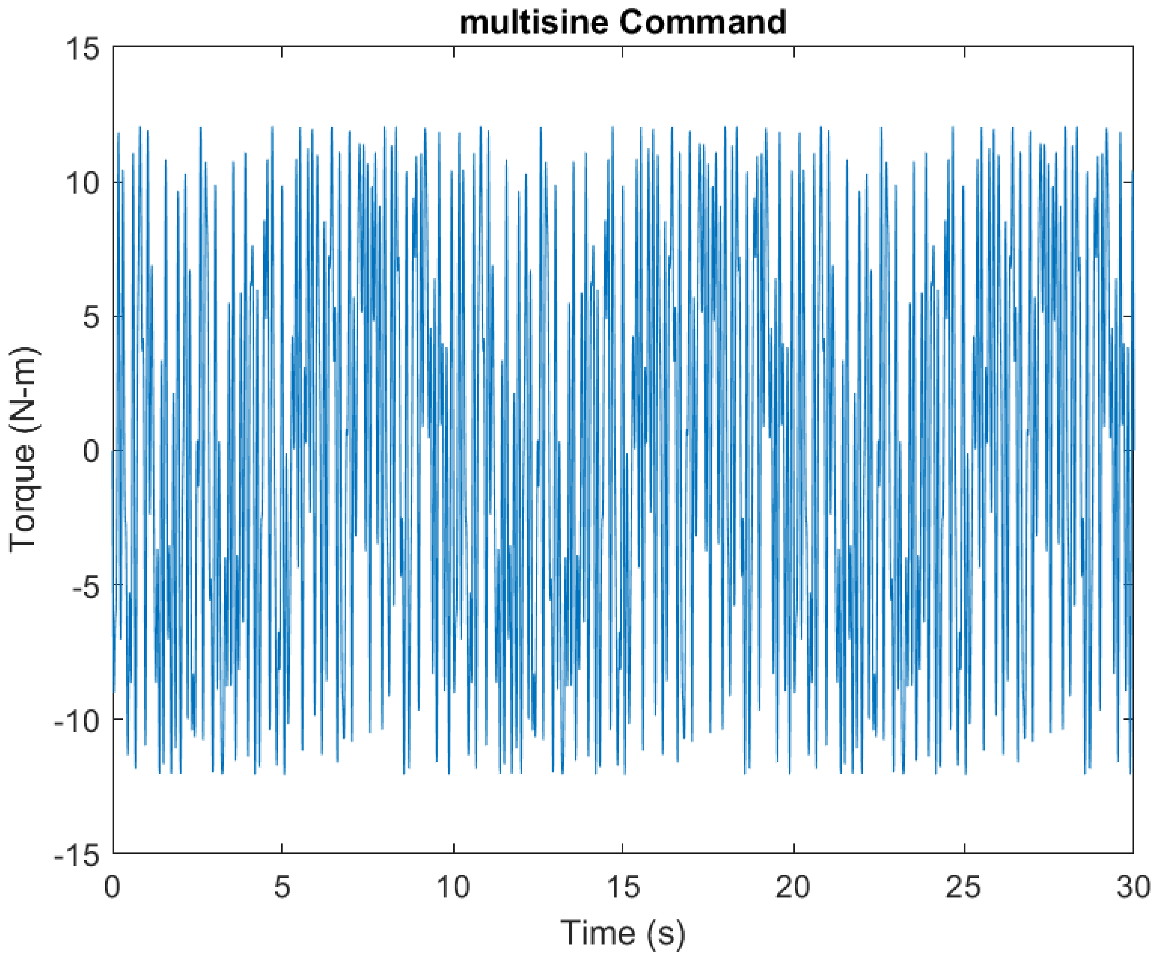
1. **Input Design:** Create a multisine **and** frequency sweep that use the same frequency range and amplitude (input time may be different). You must use at least 10 Hz as the upper frequency limit. Amplitudes of +/-10 N-m work pretty well to excite the system (about the magnitude of when the PID controller was active).

   Show two plots of your input designs vs. time.

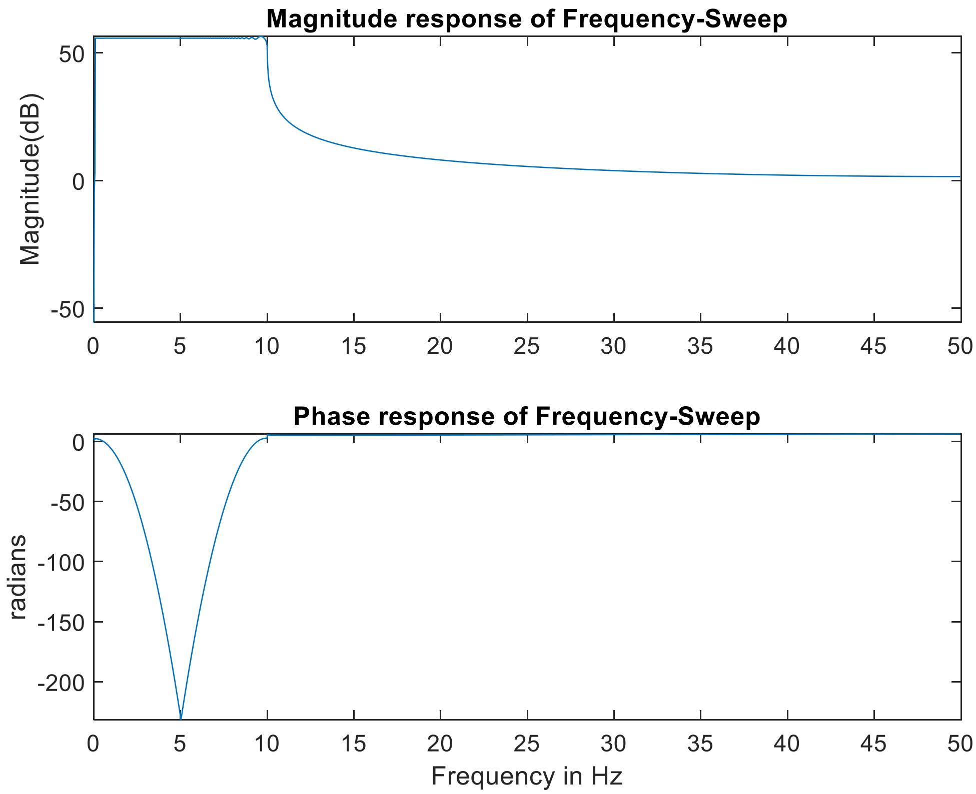
   Perform a FFT of both input designs and show the frequency response of both signals on a single plot. How similar are the inputs from a frequency domain perspective.



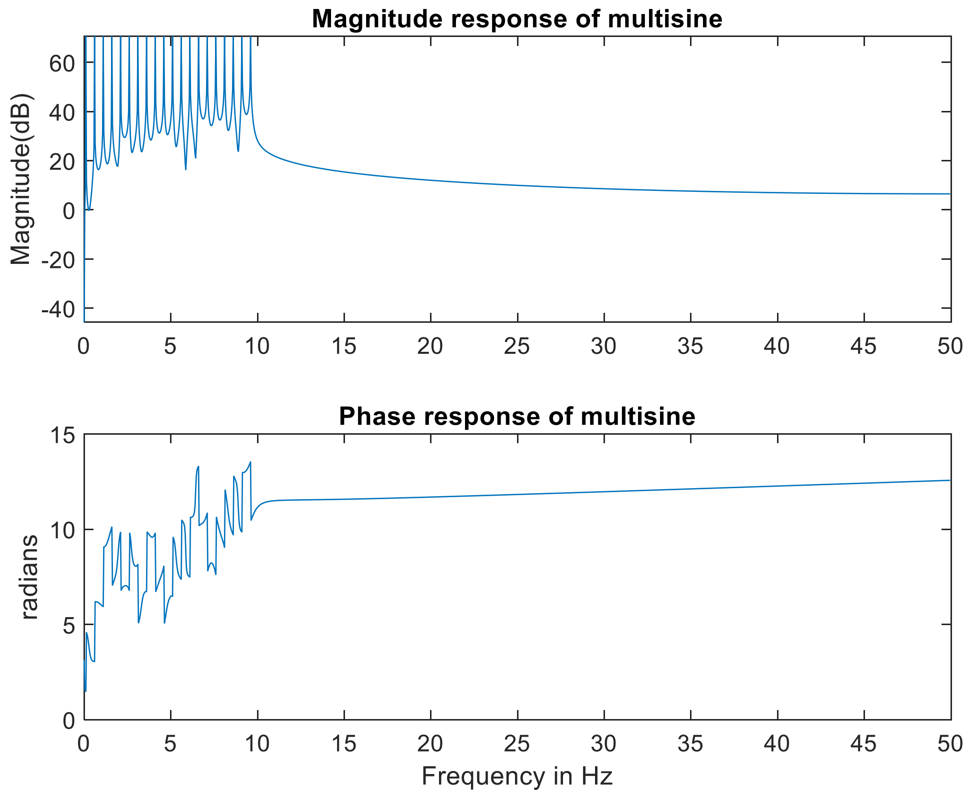
Frequency-Sweep Command
Torque (N-m)
Time (s)
10
8
6
4
2
0
-2
-4
-6
-8
-10
0
5
10
15
20
25
30

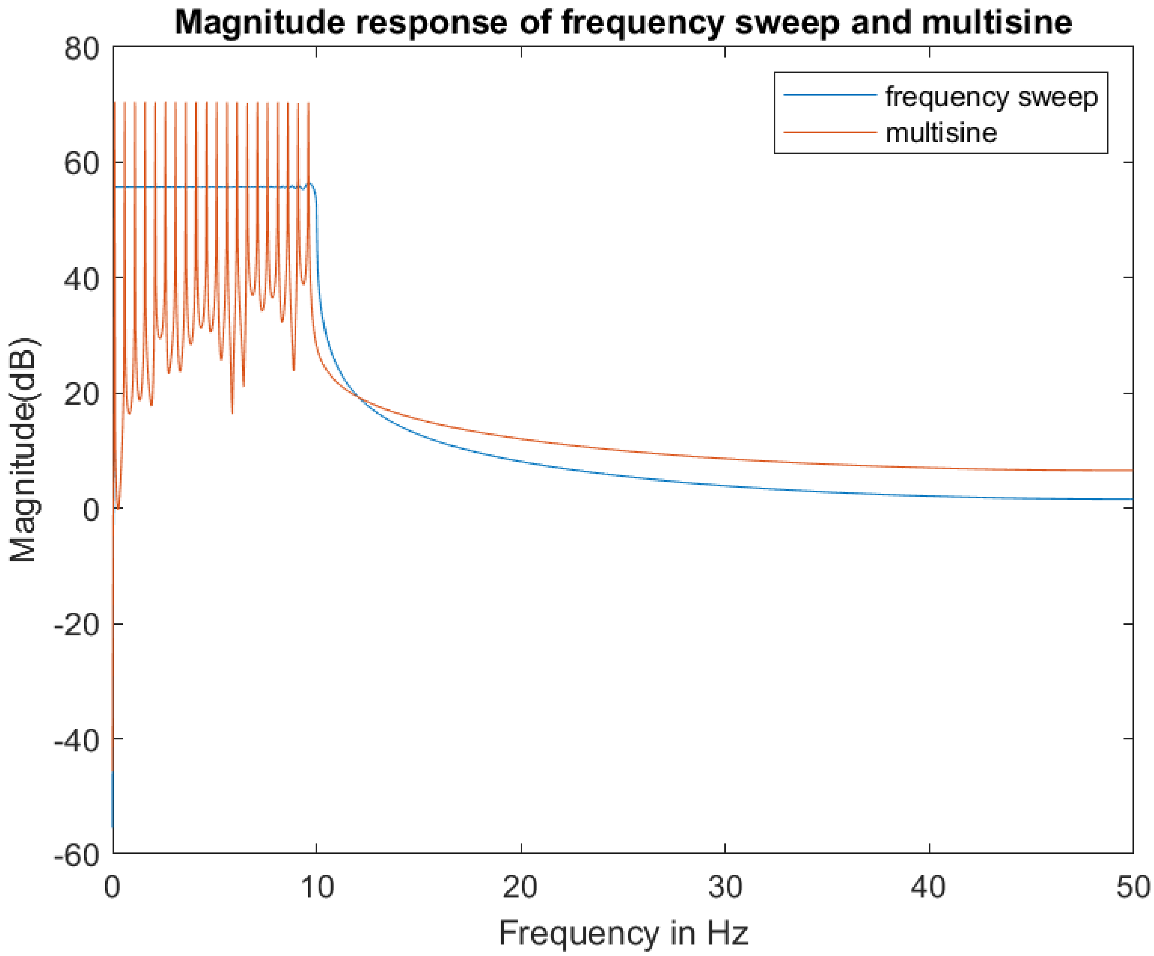


multisine Command
Torque (N-m)
Time (s)
15
10
5
0
-5
-10
-15
0
5
10
15
20
25
30



Magnitude response of Frequency-Sweep
Magnitude(dB)
50
0
-50
0
5
10
15
20
25
30
35
40
45
50
Phase response of Frequency-Sweep
radians
0
-50
-100
-150
-200
0
5
10
15
20
25
30
35
40
45
50
Frequency in Hz


Magnitude response of multisine
Magnitude(dB)
60
40
20
0
-20
-40
0
5
10
15
20
25
30
35
40
45
50
Phase response of multisine
radians
15
10
5
0
Frequency in Hz



Both input signals have a maximum frequency range of 10 Hz and exponentially taper to 0 dB. The magnitude of the peaks of the multisine signal are significantly higher than the magnitude of the frequency sweep over the 0 to 10 hz frequency range.

2. **OLS:** For both input datafiles, estimate a model using OLS of the pendulum. You may need to look up the equation of motion for a pendulum to get ideas for the model structure (Hint: there should be a trig function in your regressor matrix) Use the SAME model structure for both input files. Use the angular acceleration as your dependent variable (z). Show your model structure (with estimated coefficients). Show a plot of the model output and the actual vs. time. Add the confidence bounds onto the model output. Also, make sure to show your Coefficient of Determination (quality of your fit) for both input datafiles.

   Model structure for the frequency sweep is:

$$Y_{hat} = -.0005 - 60.0453\sin(y) + .6763\dot{y} + 1.6269u_{cmd}$$

Model structure for the multisine is:

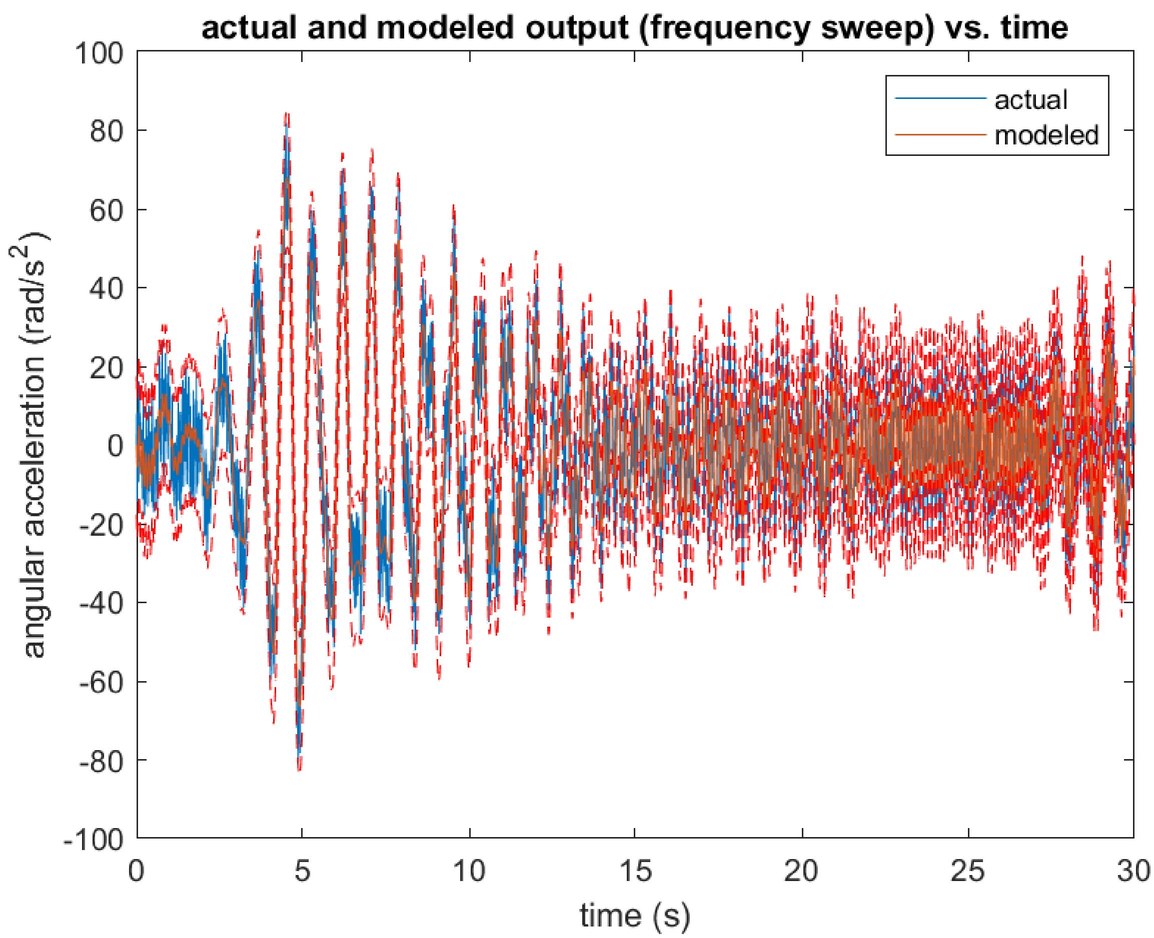
$$Y_{hat} = -.1385 - 60.0565\sin(y) + .3560\dot{y} + 2.4271u_{cmd}$$

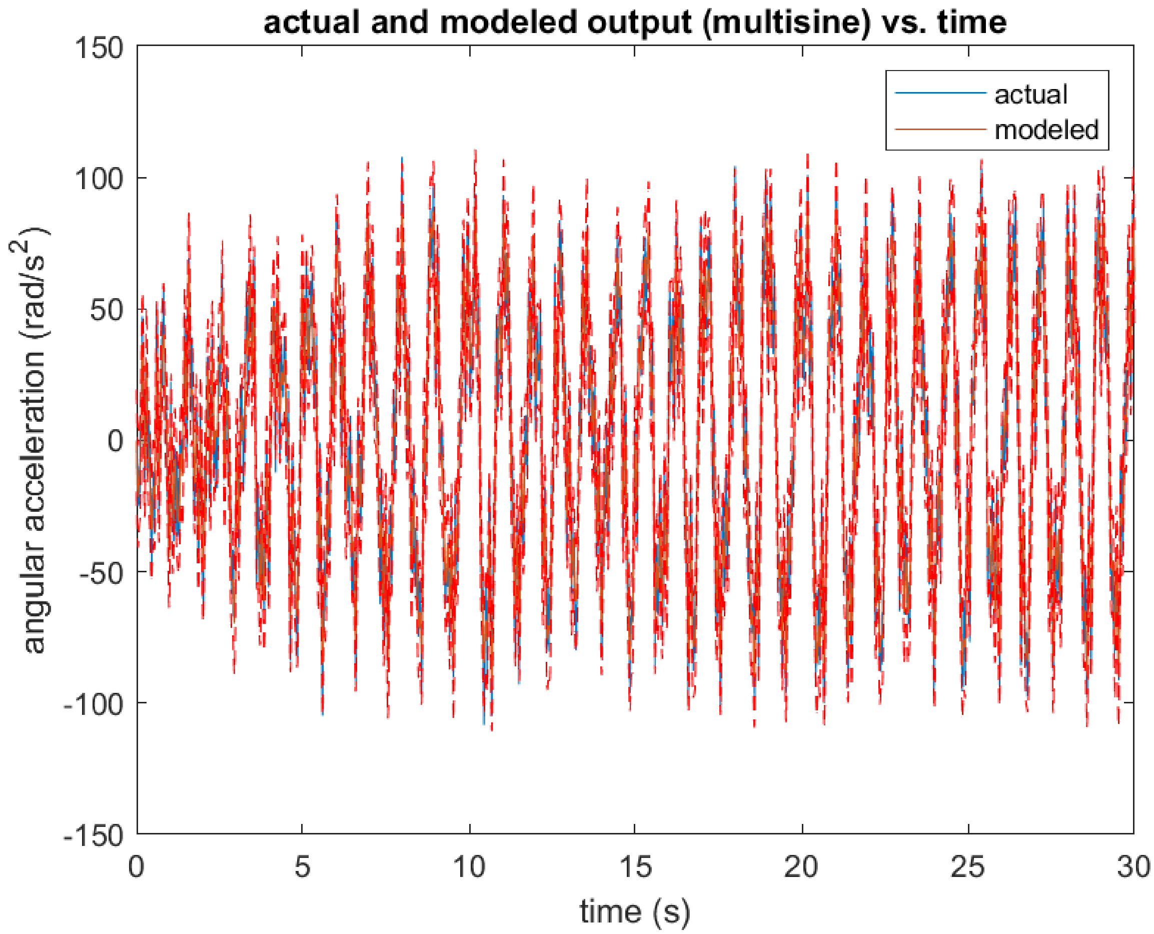
The frequency sweep R^2 value is:

$R^2 = 0.8188$

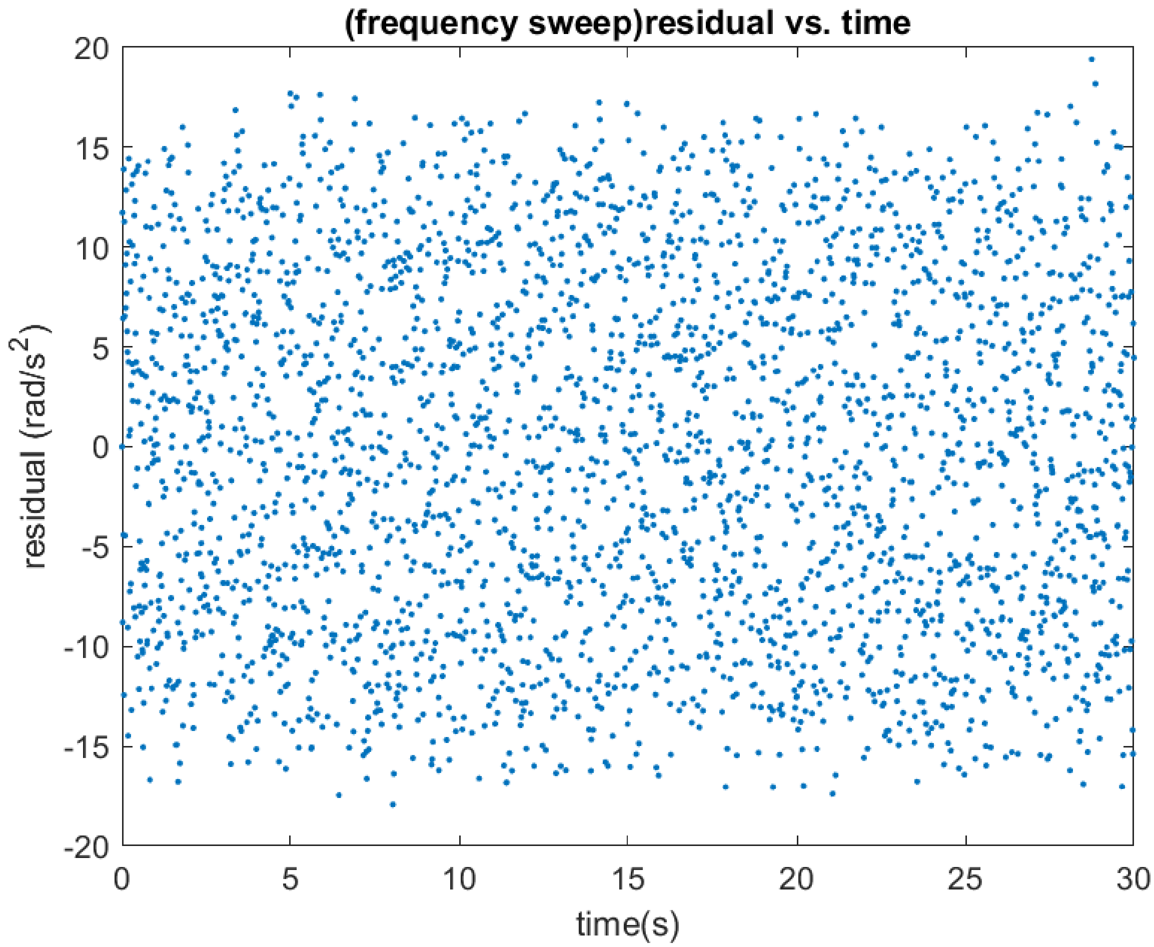
The multisine R^2 value is:

$R^2 = 0.$

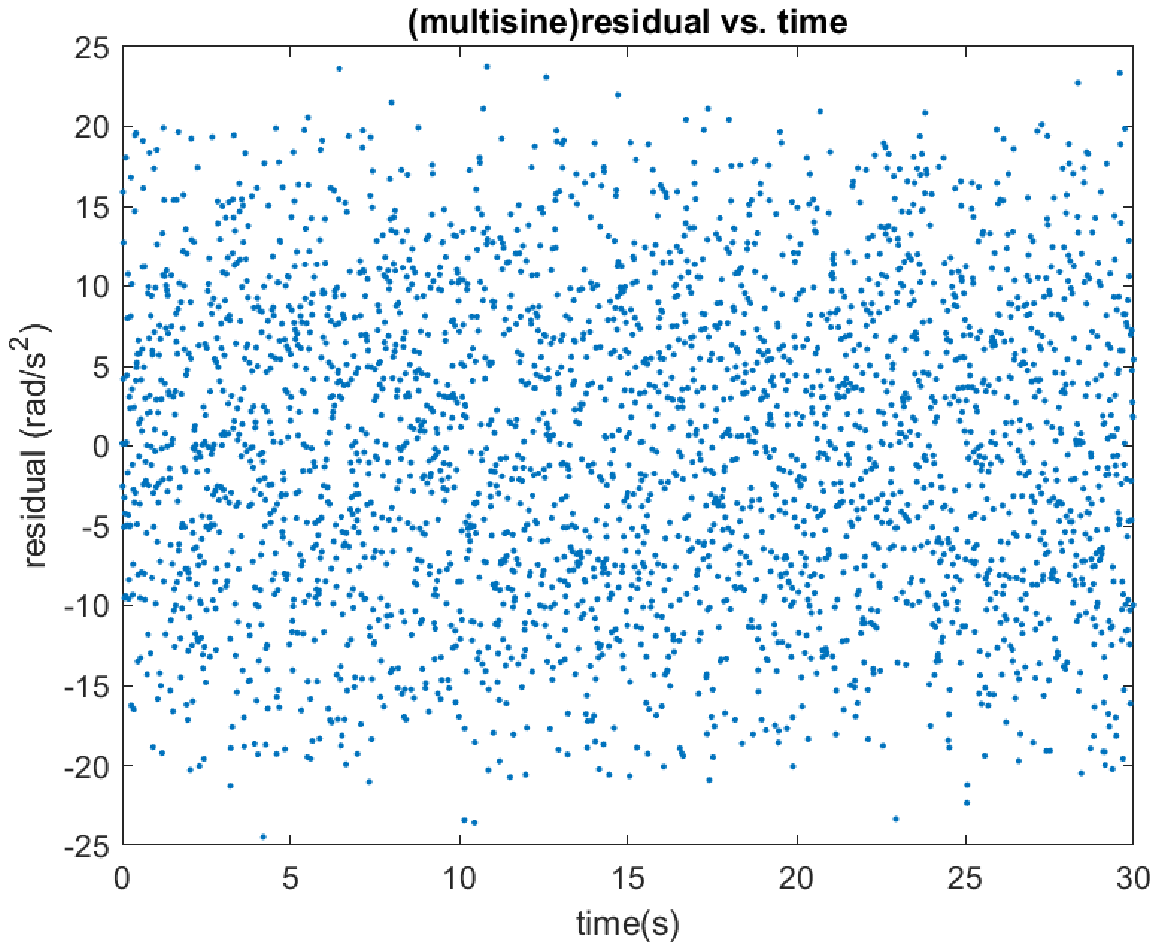




3. **Residuals:** Analyze the residuals (2 plots) as we have discussed in class for both of the input files. Comment on your results. If there are any issues, comment on potential ways to address them.



(frequency sweep)residual vs. time
residual (rad/s^2)
time(s)
20
15
10
5
0
-5
-10
-15
-20
0
5
10
15
20
25
30



The residual vs. time plot(s) show no pattern. This leads me to believe that model is adequately capturing the dynamics of the pendulum and that the residuals are primarily from the noise.

4. **Normalized Regressor:** Using your final model structure, normalize your regressor matrix and your dependent variable for both input datafiles. Show your normalized estimated parameters, what do you see in the relative importance of the parameters? Are there differences between the input types?

   Normalized Regressors (frequency sweep) =

   Sin(y): -0.8307
   $\dot{y}$: 0.0749
   u_cmd: 0.3928

   Normalized Regressors (multisine) =

   Sin(y): -0.9022
   $\dot{y}$: 0.0481
   u_cmd: 0.3647

After normalizing the regressors, the sin(y) regressor was shown to have the highest relative importance. The input command u was also important but still less than half of the sin(y). $\dot{y}$ is not very important, being less than 1% of the relative importance of the sin(y) regressor. Both sets have slightly different relative importance and follow the same trend.

Code:

#1

```
fmin = 0.1; % hz
fmax = 10.0;   % hz
dt = .01; % s
T = 30; % s
amp = 10; % N-m
Fs = 1/dt;
fu = [fmin:0.5: fmax]

% frequency sweep input/output %
[u,t,pf,f] = mksswp(amp,fmin,fmax,dt,T);

% multisine input/output %
[u2,t2,pf,f,M,ph] = mkmsswp(amp,fmin,fmax,dt,T, 1, fu);


% plotting %
figure(1)
plot(t, u)
title('Frequency-Sweep Command')
xlabel('Time (s)')
ylabel('Torque (N-m)')

figure(2)
plot(t2, u2)
title('multisine Command')
xlabel('Time (s)')
ylabel('Torque (N-m)')


% Frequency Sweep FFT %
NFFT = length(u);
Y = fft(u,NFFT);
F = ((0:1/NFFT:1-1/NFFT)*Fs).';
magnitudeY= abs(Y);        % Magnitude of the FFT
```

```
phaseY = unwrap(angle(Y));  % Phase of the FFT
dB_mag=mag2db(magnitudeY);
figure(3)
subplot(2,1,1);plot(F(1:NFFT/2),dB_mag(1:NFFT/2));title('Magnitude response
of Frequency-Sweep');
ylabel('Magnitude(dB)');
subplot(2,1,2);plot(F(1:NFFT/2),phaseY(1:NFFT/2));title('Phase response of
Frequency-Sweep');
xlabel('Frequency in Hz')
ylabel('radians');


% Frequency Sweep FFT %
NFFT2 = length(u2);
Y2 = fft(u2,NFFT2);
F2 = ((0:1/NFFT2:1-1/NFFT2)*Fs).';
magnitudeY2= abs(Y2);       % Magnitude of the FFT
phaseY2 = unwrap(angle(Y2));  % Phase of the FFT
dB_mag2=mag2db(magnitudeY2);
figure(4)
% hold on
subplot(2,1,1);plot(F2(1:NFFT2/2),dB_mag2(1:NFFT2/2));title('Magnitude
response of multisine');
ylabel('Magnitude(dB)');
subplot(2,1,2);plot(F2(1:NFFT2/2),phaseY2(1:NFFT2/2));title('Phase response
of multisine');
xlabel('Frequency in Hz')
ylabel('radians');

% combined Mag plot %
figure(5)
plot(F(1:NFFT/2),dB_mag(1:NFFT/2),F2(1:NFFT2/2),dB_mag2(1:NFFT2/2))
title('Magnitude response of frequency sweep and multisine');
ylabel('Magnitude(dB)');
xlabel('Frequency in Hz')
legend('frequency sweep', 'multisine')
```

#2-4

```
fmin = 0.1; % hz
fmax = 10.0;   % hz
dt = .01; % s
T = 30; % s
```

```
amp = 10; % N-m
Fs = 1/dt;
control_on = 0; % no conrol

% frequency sweep input/output %
[u,t,pf,f] = mksswp(amp,fmin,fmax,dt,T);
N = length(t);
[y,yd, ydd, u_cmd, t] = pend(u, dt, T, control_on);


% model from frequency sweep %
x = [ones(N,1), sin(y), yd, u_cmd];
T_hat = (x'*x)\x'*ydd
Y_hat = x*T_hat;

% frequency-sweep coefficient of determination %
disp('The model using frequency sweep data R squared value is:')
R_sqfw = (T_hat'*x'*ydd - N*mean(ydd)^2) / (ydd'*ydd - N*mean(ydd)^2)

% frequency sweep confidence interval %
v = (ydd - Y_hat);
s_sq = sum(v.^2)/(length(v) -length(T_hat));
T_hat_confid = 2 * sqrt(s_sq) * diag((x'*x)^-1);
disp('and the confidence intervals are (frequency sweep):')
T_hat_confid

% frequency sweep residual %
R = ydd - Y_hat;

% Normalized Regressor %

% for k=1:length(T_hat)
%     init = strcat('v',num2str(k));
%     init_ar = strcat('v',num2str(k));
%     sjj.(init) = 0;
%     qj.(init_ar) = [];
% end
% for k=1:length(T_hat)
%     for i = 1:length(t)
%         sjj.(strcat('v',num2str(k))) = sjj.(strcat('v',num2str(k))) + (x(i,k) -
mean(x(:,k)))^2;
%
```

```
%     end
% end
% for k = 1:length(T_hat)
%     qj.(strcat('v',num2str(k))) = (x(:,k) -
mean(x(:,k)))/sqrt(sjj.(strcat('v',num2str(k))));
% end
% xr2 = [];
% np = length(T_hat);
% np_temp = np -1;
% for k = 1:np_temp
%     xr2 = [xr2, qj.(strcat('v',num2str(k)))];
% end
% T_hatnorm2 = (xr'*xr)\xr'*qj.(strcat('v',num2str(length(T_hat))))

sjj_1 = 0;
sjj_2 = 0;
sjj_3 = 0;
sjj_4 = 0;
sjj_5 = 0;
qj_1 = [];
qj_2 = [];
qj_3 = [];
qj_4 = [];
qj_5 = [];
for i = 1:length(t)
  sjj_1 = sjj_1 + (x(i,1) - mean(x(:,1)))^2;
  sjj_2 = sjj_2 + (x(i,2) - mean(x(:,2)))^2;
  sjj_3 = sjj_3 + (x(i,3) - mean(x(:,3)))^2;
  sjj_4 = sjj_4 + (x(i,4) - mean(x(:,4)))^2;
  sjj_5 = sjj_5 + (ydd(i) - mean(ydd))^2;
end

qj_1 = (x(:,1) - mean(x(:,1)))/sqrt(sjj_1);
qj_2 = (x(:,2) - mean(x(:,2)))/sqrt(sjj_2);
qj_3 = (x(:,3) - mean(x(:,3)))/sqrt(sjj_3);
qj_4 = (x(:,4) - mean(x(:,4)))/sqrt(sjj_4);
qj_5 = (ydd - mean(ydd))/sqrt(sjj_5);

xr = [qj_2, qj_3, qj_4];
T_hatnorm = (xr'*xr)\xr'*qj_5 % #3
% Frequency Sweep Plot %
figure(1)
```

```
plot(t,ydd,t,Y_hat, t, Y_hat + 2 * sqrt(s_sq),'--r', t, Y_hat - 2 * sqrt(s_sq), '--r')
title('actual and modeled output (frequency sweep) vs. time')
xlabel('time (s)')
ylabel('angular acceleration (rad/s^2)')
legend('actual','modeled')

figure(2)
plot(t,R,'.')
title('(frequency sweep)residual vs. time')
xlabel('time(s)')
ylabel('residual (rad/s^2)')

% multisine input/output %
fu = [fmin:0.5: fmax];
[u2,t2,pf,f,M,ph] = mkmsswp(amp,fmin,fmax,dt,T,1,fu);
N2 = length(t2);
[y2,yd2, ydd2, u_cmd2, t2] = pend(u2, dt, T, control_on);




% model from multisine %
x2 = [ones(N2,1), sin(y2), yd2, u_cmd2];
T_hat2 = (x2'*x2)\x2'*ydd2
Y_hat2 = x2*T_hat2;

% multisine coefficient of determination %
disp('The model using multisine data R squared value is:')
R_sqms = (T_hat2'*x2'*ydd2 - N2*mean(ydd2)^2) / (ydd2'*ydd2 -
N2*mean(ydd2)^2)

% frequency sweep confidence interval %
v2 = (ydd2 - Y_hat2);
s_sq2 = sum(v2.^2)/(length(v2) -length(T_hat2));
T_hat_confid2 = 2 * sqrt(s_sq2) * diag((x2'*x2)^-1);
disp('and the confidence intervals are (multisine):')
T_hat_confid2

% multisine residual %
R2 = ydd2 - Y_hat2;

% multisine Normalized Regressor %
```

```
sjj2_1 = 0;
sjj2_2 = 0;
sjj2_3 = 0;
sjj2_4 = 0;
sjj2_5 = 0;
qj2_1 = [];
qj2_2 = [];
qj2_3 = [];
qj2_4 = [];
qj2_5 = [];
for i = 1:length(t2)
  sjj2_1 = sjj2_1 + (x2(i,1) - mean(x2(:,1)))^2;
  sjj2_2 = sjj2_2 + (x2(i,2) - mean(x2(:,2)))^2;
  sjj2_3 = sjj2_3 + (x2(i,3) - mean(x2(:,3)))^2;
  sjj2_4 = sjj2_4 + (x2(i,4) - mean(x2(:,4)))^2;
  sjj2_5 = sjj2_5 + (ydd2(i) - mean(ydd2))^2;
end

qj2_1 = (x2(:,1) - mean(x2(:,1)))/sqrt(sjj2_1);
qj2_2 = (x2(:,2) - mean(x2(:,2)))/sqrt(sjj2_2);
qj2_3 = (x2(:,3) - mean(x2(:,3)))/sqrt(sjj2_3);
qj2_4 = (x2(:,4) - mean(x2(:,4)))/sqrt(sjj2_4);
qj2_5 = (ydd2 - mean(ydd2))/sqrt(sjj2_5);

xr2 = [qj2_2, qj2_3, qj2_4];
T_hatnorm2 = (xr2'*xr2)\xr2'*qj2_5 % #3
% Frequency Sweep Plot %
figure(3)
plot(t2, ydd2, t2, Y_hat2, t2, Y_hat2 + 2 * sqrt(s_sq2),'--r', t2, Y_hat2 - 2 *
sqrt(s_sq2), '--r')
title('actual and modeled output (multisine) vs. time')
xlabel('time (s)')
ylabel('angular acceleration (rad/s^2)')
legend('actual','modeled')

figure(4)
plot(t2,R2,'.')
title('(multisine)residual vs. time')
xlabel('time(s)')
ylabel('residual (rad/s^2)')
```