

CHAPTER 11

FREQUENCY RESPONSE: MODAL STATE SPACE FORM

11.1 Introduction

In Chapter 10 we constructed the modal form of the state equations for the overall frequency response as well as for the individual mode contributions. This short chapter of MATLAB code will carry out both overall and individual mode frequency response calculations. The code will also allow us to plot the different forms of frequency responses covered in Chapter 3.

11.2 Modal State Space Setup, `tdofss_modal_xfer_modes.m` Listing

After executing the “`tdofss_eig.m`” code to provide eigenvalues and eigenvectors, we enter a section of code that yields similar results to those resulting from an ANSYS simulation. In the ANSYS case, we would have access to the eigenvalues and mass normalized eigenvectors, similar to the “`xn`” and “`w1, w2 and w3`” from `tdofss_eig.m`.

Since we can add proportional damping to our modal model, the code prompts for a value for zeta.

Knowing zeta and the eigenvalues, the system matrix can be setup as shown in (10.35), as 2x2 blocks along the diagonal. The three 2x2 submatrices of the system matrix are defined for individual mode contribution calculations.

The next step is to define a 6x3 input matrix, 6 states and three possible inputs representing forces applied to only mass 1, only mass 2 or only mass 3. We start out by defining three separate 3x1 force vectors, one for each mass, F_1 , F_2 and F_3 . Each of these vectors is transformed from physical to principal coordinates by premultiplying by x_n transpose. The three 3x1 vectors are padded with zeros resulting in three 6x1 vectors, which are then inserted as columns in the 6x3 input matrix “`b`.”

The output matrix, “`c`,” is defined in one step as shown in (10.38) by incorporating the appropriate elements of “`xn`.” However, only displacement states are output, giving a 3x6 matrix.

The direct transmission matrix is set to zero.

```

%      tdofss_modal_xfer_modes.m      state-space modal form transfer function analysis
%      of tdof model, proportional damping, modal contribution plotting

      clf;

      clear all;

%      run tdofss_eig.m to provide eigenvalues and eigenvectors

      tdofss_eig;

%      note, this is the point where we would start if we had eigenvalue results from ANSYS,
%      using the eigenvalues and eigenvectors to define state space equations in
%      principal coordinates

%      define damping ratio to be used for proportional damping in the state space equation
%      in principal coordinates

      zeta = input('input zeta, 0.02 = 2% of critical damping (default) ... ');

      if (isempty(zeta))
      zeta = 0.02;
      else
      end

%      setup 6x6 state-space system matrix for all three modes in principal
%      coordinates, a_ss

      a_ss = [0      1      0      0      0      0
              0      0      0      0      0      0
              0      0      0      1      0      0
              0      0      -w2^2  -2*zeta*w2  0      0
              0      0      0      0      0      1
              0      0      0      0      -w3^2  -2*zeta*w3];

%      setup three 2x2 state-space matrices, one for each individual mode

      a1_ss = a_ss(1:2,1:2);

      a2_ss = a_ss(3:4,3:4);

      a3_ss = a_ss(5:6,5:6);

%      transform the 3x1 force vectors in physical coordinates to principal coordinates and
%      then insert the principal forces in the appropriate rows in the state-space
%      6x1 input matrix, padding with zeros as appropriate

%      define three force vectors in physical coordinates, where each is for
%      a force applied to a single mass

      F1 = [1 0 0];

      F2 = [0 1 0];

      F3 = [0 0 1];

```

```

% calculate the three force vectors in principal coordinates by pre-multiplying
% by the transpose of the normalized modal matrix

Fp1 = xn'*F1;

Fp2 = xn'*F2;

Fp3 = xn'*F3;

% expand the force vectors in principal coordinates from 3x1 to 6x1, padding with zeros

b1 = [0 Fp1(1) 0 Fp1(2) 0 Fp1(3)]; % principal force applied at mass 1

b2 = [0 Fp2(1) 0 Fp2(2) 0 Fp2(3)]; % principal force applied at mass 2

b3 = [0 Fp3(1) 0 Fp3(2) 0 Fp3(3)]; % principal force applied at mass 3

b = [b1 b2 b3];

% the output matrix c is setup in one step, to allow the "bode" command to
% output the desired physical coordinates directly without having to go
% through any intermediate steps.

% setup the output matrix for displacement transfer functions, each row
% represents the position outputs of mass 1, mass 2 and mass 3
% velocities not included, so c is only 3x6 instead of 6x6

c = [xn(1,1) 0 xn(1,2) 0 xn(1,3) 0
     xn(2,1) 0 xn(2,2) 0 xn(2,3) 0
     xn(3,1) 0 xn(3,2) 0 xn(3,3) 0];

% define direct transmission matrix d

d = zeros(3,3);

```

11.3 Frequency Response Calculation

We will begin this section by defining the vector of frequencies to be used for the frequency response plot. Then we will define a state space model, using the matrices defined in the section above.

Because we are using a 6x3 input matrix and a 3x6 output matrix, we have access to nine frequency response plots, the displacement for all three degrees of freedom for three different force application points. To plot the four distinct frequency responses, the appropriate indices are used to define magnitude and phase.

```

% Define a vector of frequencies to use, radians/sec. The logspace command uses
% the log10 value as limits, i.e. -1 is 10^-1 = 0.1 rad/sec, and 1 is
% 10^1 = 10 rad/sec. The 200 defines 200 frequency points.

```

```

w = logspace(-1,1,200);

% define four state-space systems using the "ss" command
% sys is for all modes for all 3 forcing functions
% sys1 is for mode 1 for all 3 forcing functions
% sys2 is for mode 2 for all 3 forcing functions
% sys3 is for mode 3 for all 3 forcing functions

sys = ss(a_ss,b,c,d);

sys1 = ss(a1_ss,b(1:2,:),c(:,1:2),d);

sys2 = ss(a2_ss,b(3:4,:),c(:,3:4),d);

sys3 = ss(a3_ss,b(5:6,:),c(:,5:6),d);

% use the bode command with left hand magnitude and phase vector arguments
% to provide values for further analysis/plotting

[mag,phs] = bode(sys,w);

[mag1,phs1] = bode(sys1,w);

[mag2,phs2] = bode(sys2,w);

[mag3,phs3] = bode(sys3,w);

% pick out the specific magnitudes and phases for four distinct responses

z11mag = mag(1,1,:);

z21mag = mag(2,1,:);

z31mag = mag(3,1,:);

z22mag = mag(2,2,:);

z11magdb = 20*log10(z11mag);

z21magdb = 20*log10(z21mag);

z31magdb = 20*log10(z31mag);

z22magdb = 20*log10(z22mag);

z11phs = phs(1,1,:);

z21phs = phs(2,1,:);

z31phs = phs(3,1,:);

z22phs = phs(2,2,:);

% pick out the three individual mode contributions to z11

```

```

z111mag = mag1(1,1,:);
z112mag = mag2(1,1,:);
z113mag = mag3(1,1,:);
z111magdb = 20*log10(z111mag);
z112magdb = 20*log10(z112mag);
z113magdb = 20*log10(z113mag);
z111phs = phs1(1,1,:);
z112phs = phs2(1,1,:);
z113phs = phs3(1,1,:);

```

11.4 Frequency Response Plotting

```

%      truncate peaks for plotting of expanded linear scale
z11plotmag = z11mag;
z111plotmag = z111mag;
z112plotmag = z112mag;
z113plotmag = z113mag;
for cnt = 1:length(z11mag)
    if z11plotmag(cnt) >= 3.0
        z11plotmag(cnt) = 3.0;
    end
    if z111plotmag(cnt) >= 3.0
        z111plotmag(cnt) = 3.0;
    end
    if z112plotmag(cnt) >= 3.0
        z112plotmag(cnt) = 3.0;
    end
    if z113plotmag(cnt) >= 3.0

```

```

                                z113plotmag(cnt) = 3.0;

                                end

                                end

                                end
% plot the four transfer functions separately, in a 2x2 subplot form

                                subplot(2,2,1)
                                semilogx(w,z11magdb(1:),'k-')
                                title('state space, z11, z33 db magnitude')
                                ylabel('magnitude, db')
                                axis([.1 10 -150 50])
                                grid

                                subplot(2,2,2)
                                semilogx(w,z21magdb(1:),'k-')
                                title('state space, z21, z12, z23, z32 db magnitude')
                                ylabel('magnitude, db')
                                axis([.1 10 -150 50])
                                grid

                                subplot(2,2,3)
                                semilogx(w,z31magdb(1:),'k-')
                                title('state space, z31, z13 db magnitude')
                                xlabel('frequency, rad/sec')
                                ylabel('magnitude, db')
                                axis([.1 10 -150 50])
                                grid

                                subplot(2,2,4)
                                semilogx(w,z22magdb(1:),'k-')
                                title('state space, z22 db magnitude')
                                xlabel('frequency, rad/sec')
                                ylabel('magnitude, db')
                                axis([.1 10 -150 50])
                                grid

                                disp('execution paused to display figure, "enter" to continue'); pause

                                subplot(2,2,1)
                                semilogx(w,z11phs(1:),'k-')
                                title('state space, z11, z33 phase')
                                ylabel('phase, deg')
                                grid

                                subplot(2,2,2)
                                semilogx(w,z21phs(1:),'k-')
                                title('state space, z21, z12, z23, z32 phase')
                                ylabel('phase, deg')
                                grid

                                subplot(2,2,3)
                                semilogx(w,z31phs(1:),'k-')
                                title('state space, z31, z13 phase')

```

```

xlabel('frequency, rad/sec')
ylabel('phase, deg')
grid

subplot(2,2,4)
semilogx(w,z22phs(1,:),'k-')
title('state space, z22 phase')
xlabel('frequency, rad/sec')
ylabel('phase, deg')
grid

disp('execution paused to display figure, "enter" to continue'); pause

% plot the overall plus individual mode contributions separately

subplot(2,2,1)
semilogx(w,z11magdb(1,:),'k-')
title('State-Space Modal, z11 db magnitude')
ylabel('magnitude, db')
axis([.1 10 -60 40])
grid

subplot(2,2,2)
semilogx(w,z111magdb(1,:),'k-')
title('State-Space Modal, z11 db magnitude of mode 1')
ylabel('magnitude, db')
axis([.1 10 -60 40])
grid

subplot(2,2,3)
semilogx(w,z112magdb(1,:),'k-')
title('State-Space Modal, z11 db magnitude of mode 2')
xlabel('frequency, rad/sec')
ylabel('magnitude, db')
axis([.1 10 -60 40])
grid

subplot(2,2,4)
semilogx(w,z113magdb(1,:),'k-')
title('State-Space Modal, z11 db magnitude of mode 3')
xlabel('frequency, rad/sec')
ylabel('magnitude, db')
axis([.1 10 -60 40])
grid

disp('execution paused to display figure, "enter" to continue'); pause

subplot(2,2,1)
semilogx(w,z11phs(1,:),'k-')
title('State-Space Modal, z11 phase')
ylabel('phase, deg')
grid

subplot(2,2,2)
semilogx(w,z111phs(1,:),'k-')

```

```

title('State-Space Modal, z11 phase of mode 1')
ylabel('phase, deg')
grid

subplot(2,2,3)
semilogx(w,z112phs(1:),'k-')
title('State-Space Modal, z11 phase of mode 2')
xlabel('frequency, rad/sec')
ylabel('phase, deg')
grid

subplot(2,2,4)
semilogx(w,z113phs(1:),'k-')
title('State-Space Modal, z11 phase of mode 3')
xlabel('frequency, rad/sec')
ylabel('phase, deg')
grid

disp('execution paused to display figure, "enter" to continue'); pause

subplot(1,1,1);

% plot the overlaid transfer function and individual mode contributions

loglog(w,z11mag(1:),'k+',w,z111mag(1:),'k-',w,z112mag(1:),'k-',w, ...
        z113mag(1:),'k-')
title('State-Space Modal Mode Contributions, z11 db magnitude')
xlabel('frequency, rad/sec')
ylabel('magnitude, db')
axis([.1 10 .001 100])
grid

disp('execution paused to display figure, "enter" to continue'); pause

semilogx(w,z11mag(1:),'k+',w,z111mag(1:),'k-',w,z112mag(1:), ...
        'k-',w,z113mag(1:),'k-')
title('State-Space Modal Mode Contributions, z11 linear magnitude')
xlabel('frequency, rad/sec')
ylabel('magnitude')
grid

disp('execution paused to display figure, "enter" to continue'); pause

semilogx(w,z11plotmag(1:),'k+',w,z111plotmag(1:),'k-', ...
        w,z112plotmag(1:),'k-',w,z113plotmag(1:),'k-')
title('State-Space Modal Mode Contributions, z11 linear magnitude')
xlabel('frequency, rad/sec')
ylabel('magnitude')
axis([.1 10 0 3]);
grid

disp('execution paused to display figure, "enter" to continue'); pause

semilogx(w,z11phs(1:),'k+',w,z111phs(1:),'k-',w,z112phs(1:),'k-', ...
        w,z113phs(1:),'k-')

```

```
title('State-Space Modal Mode Contributions, z11 phase')
xlabel('frequency, rad/sec')
ylabel('phase, deg')
grid
```

11.5 Code Results – Frequency Response Plots, 2% of Critical Damping

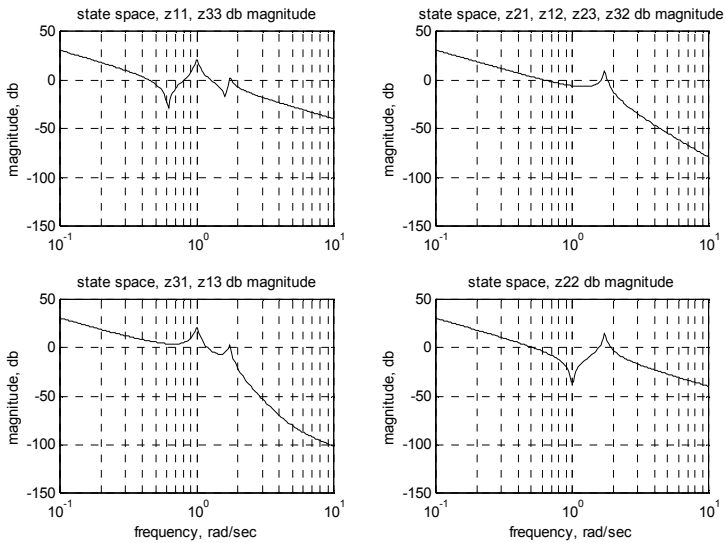


Figure 11.1: Magnitude output for four distinct frequency responses, proportional damping $\zeta = 2\%$.

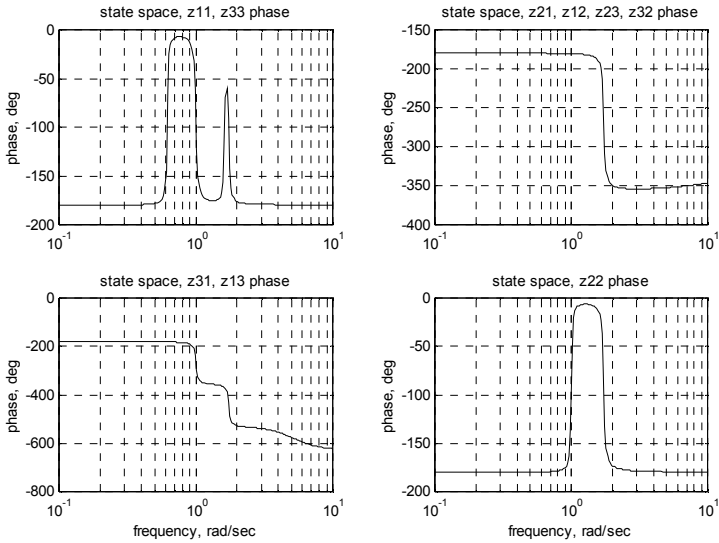


Figure 11.2: Phase output for four distinct frequency responses, proportional damping $\zeta = 2\%$.

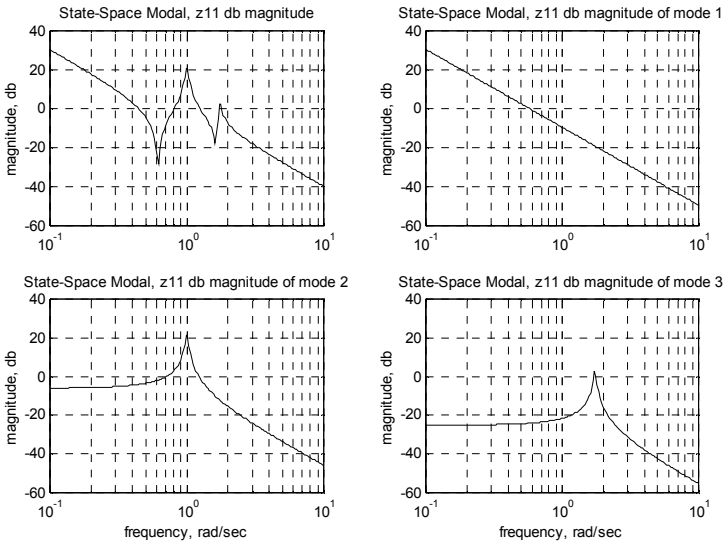


Figure 11.3: Magnitude output for z11 frequency response and individual mode contributions.

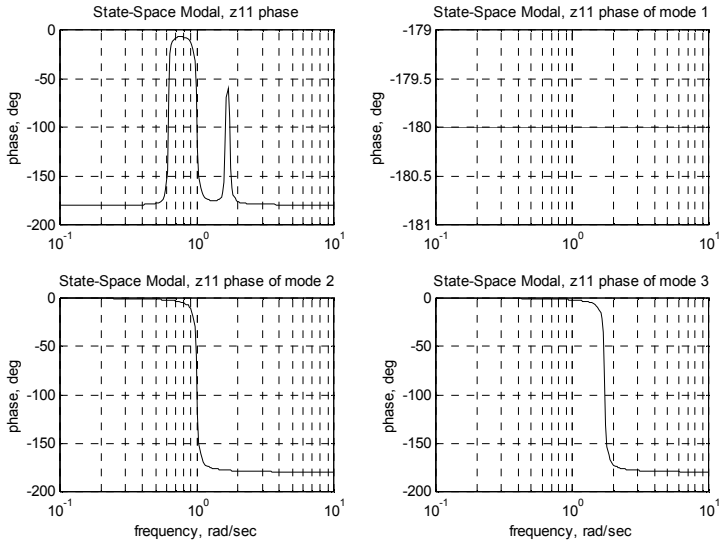


Figure 11.4: Phase output for z11 frequency response and individual mode contributions.

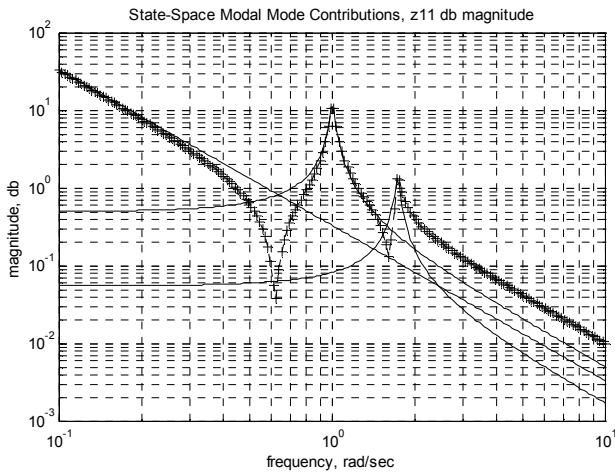


Figure 11.5: Overlaid magnitude output for z11 frequency response and individual mode contributions.

11.6 Forms of Frequency Response Plotting

This section of code is used to plot various forms of frequency responses for the z11 transfer function, as shown in Chapter 3, Section 3.6. All the plots

except the Nyquist plot use user-defined damping and 200 frequency points. The Nyquist section recalculates the system matrix to use a damping zeta of 0.02 and 800 frequency points in order to plot in the designated format.

```
%      plot only z11 transfer function in different formats

      orient tall

%      log mag, log freq

      subplot(2,1,1)
      loglog(w,z11mag(1:),'k-')
      title('z11, z33 log mag versus log freq')
      ylabel('magnitude')
      grid

      subplot(2,1,2)
      semilogx(w,z11phs(1:),'k-')
      title('z11, z33 phase versus log freq')
      xlabel('frequency, rad/sec')
      ylabel('phase, deg')
      grid

      disp('execution paused to display figure, "enter" to continue'); pause

%      db mag, log freq

      subplot(2,1,1)
      semilogx(w,z11magdb(1:),'k-')
      title('z11, z33 db mag versus log freq')
      ylabel('magnitude, db')
      grid

      subplot(2,1,2)
      semilogx(w,z11phs(1:),'k-')
      title('z11, z33 phase versus log freq')
      xlabel('frequency, rad/sec')
      ylabel('phase, deg')
      grid

      disp('execution paused to display figure, "enter" to continue'); pause

%      db mag, lin freq

      subplot(2,1,1)
      plot(w,z11magdb(1:),'k-')
      title('z11, z33 db mag versus linear freq')
      ylabel('magnitude, db')
      grid

      subplot(2,1,2)
      plot(w,z11phs(1:),'k-')
      title('z11, z33 phase versus linear freq')
      xlabel('frequency, rad/sec')
```

```

ylabel('phase, deg')
grid

disp('execution paused to display figure, "enter" to continue'); pause

% lin mag, lin freq

subplot(2,1,1)
plot(w,z11mag(1:),'k-')
title('z11, z33 linear mag versus linear freq')
ylabel('magnitude')
grid

subplot(2,1,2)
plot(w,z11phs(1:),'k-')
title('z11, z33 phase versus linear freq')
xlabel('frequency, rad/sec')
ylabel('phase, deg')
grid

disp('execution paused to display figure, "enter" to continue'); pause

% linear real versus log freq, linear imag versus log freq

z11real = z11mag.*cos(z11phs*pi/180); % convert from mag/angle to real
z11realdB = 20*log10(z11real);

z11imag = z11mag.*sin(z11phs*pi/180); % convert from mag/angle to imag
z11imagdB = 20*log10(z11imag);

subplot(2,1,1)
semilogx(w,z11real(1:),'k-')
title('z11, z33 linear real mag versus log freq')
ylabel('real magnitude')
grid

subplot(2,1,2)
semilogx(w,z11imag(1:),'k-')
title('z11, z33 linear imaginary versus log freq')
xlabel('frequency, rad/sec')
ylabel('imaginary magnitude');
grid

disp('execution paused to display figure, "enter" to continue'); pause

% linear real versus linear freq, linear imag versus linear freq

subplot(2,1,1)
plot(w,z11real(1:),'k-')
title('z11, z33 linear real mag versus linear freq')
ylabel('real magnitude')
grid

```

```

subplot(2,1,2)
plot(w,z11imag(1,:),'k-')
title('z11, z33 linear imaginary versus linear freq')
xlabel('frequency, rad/sec')
ylabel('imaginary magnitude');
grid

disp('execution paused to display figure, "enter" to continue'); pause

% real versus imaginary (Nyquist), redo frequency response with 800 points for
% finer frequency resolution for Nyquist plot and use zeta = 0.02 to fit on plot

zeta = 0.02;

a_ss = [0      1      0      0      0      0
        0      0      0      0      0      0
        0      0      0      1      0      0
        0      0     -w2^2  -2*zeta*w2  0      0
        0      0      0      0      0      1
        0      0      0      0     -w3^2  -2*zeta*w3];

w = logspace(-1,1,800);

sys = ss(a_ss,b,c,d);

[mag,phs] = bode(sys,w);

z11mag = mag(1,1,:);

z11magdb = 20*log10(z11mag);

z11phs = phs(1,1,:);

z11real = z11mag.*cos(z11phs*pi/180);    % convert from mag/angle to real

z11imag = z11mag.*sin(z11phs*pi/180);  % convert from mag/angle to imag

subplot(1,1,1)

plot(z11real(1,:),z11imag(1,:),'k+');
title('z11, z33 real versus imaginary, "Nyquist"')
ylabel('imag')
axis('square')
axis([-15 15 -15 15])
grid

```

Problem

Note: This problem refers to the two dof system shown in [Figure P2.2](#).

P11.1 (MATLAB) Modify the `tdofss_modal_xfer_modes.m` code for the two dof system with $m_1 = m_2 = m = 1$, $k_1 = k_2 = k = 1$ and plot the frequency responses with and without the individual mode contributions overlaid.