Homework 5

Louis Lin

January 24, 2022

University of California, San Diego

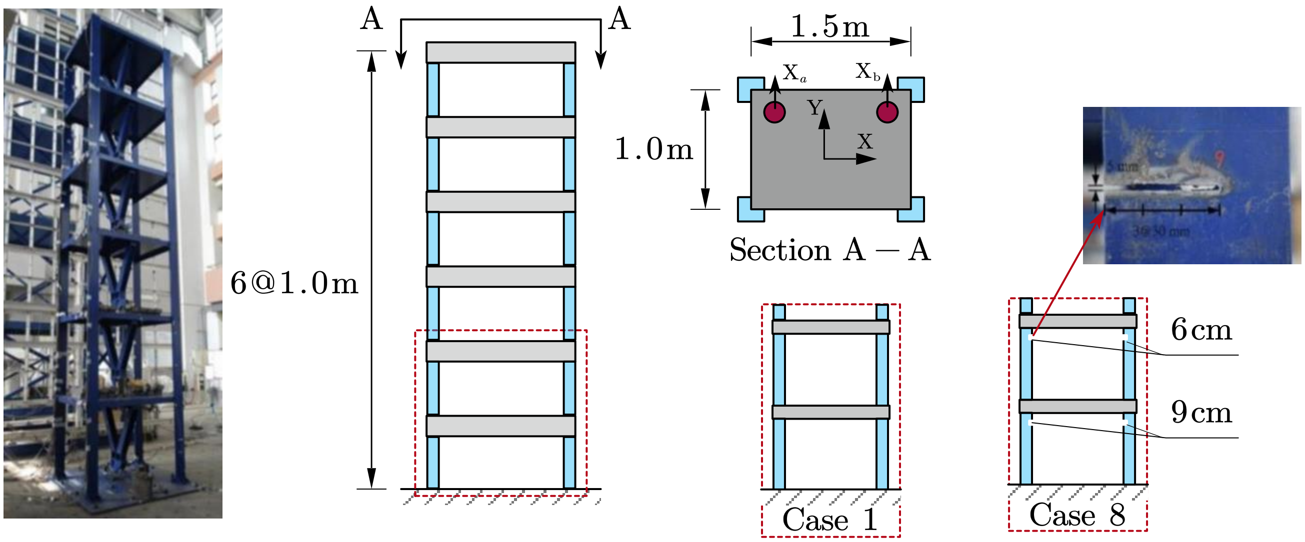
Prof. Chin-Hsiung Loh

SE 267A Signal Processing and Spectral Analysis

Problem 5

The current homework uses the enhanced frequency decomposition method to identify the modal characteristics of a 6-story steel structure using only output white noise excitation data shown on the next page.

In this study, two configuration of the building was explored, case 1 and case 8. Case 1 was the virgin state of the building while Case 8 had notches on building on first floor and second floor to simulate damage to the building



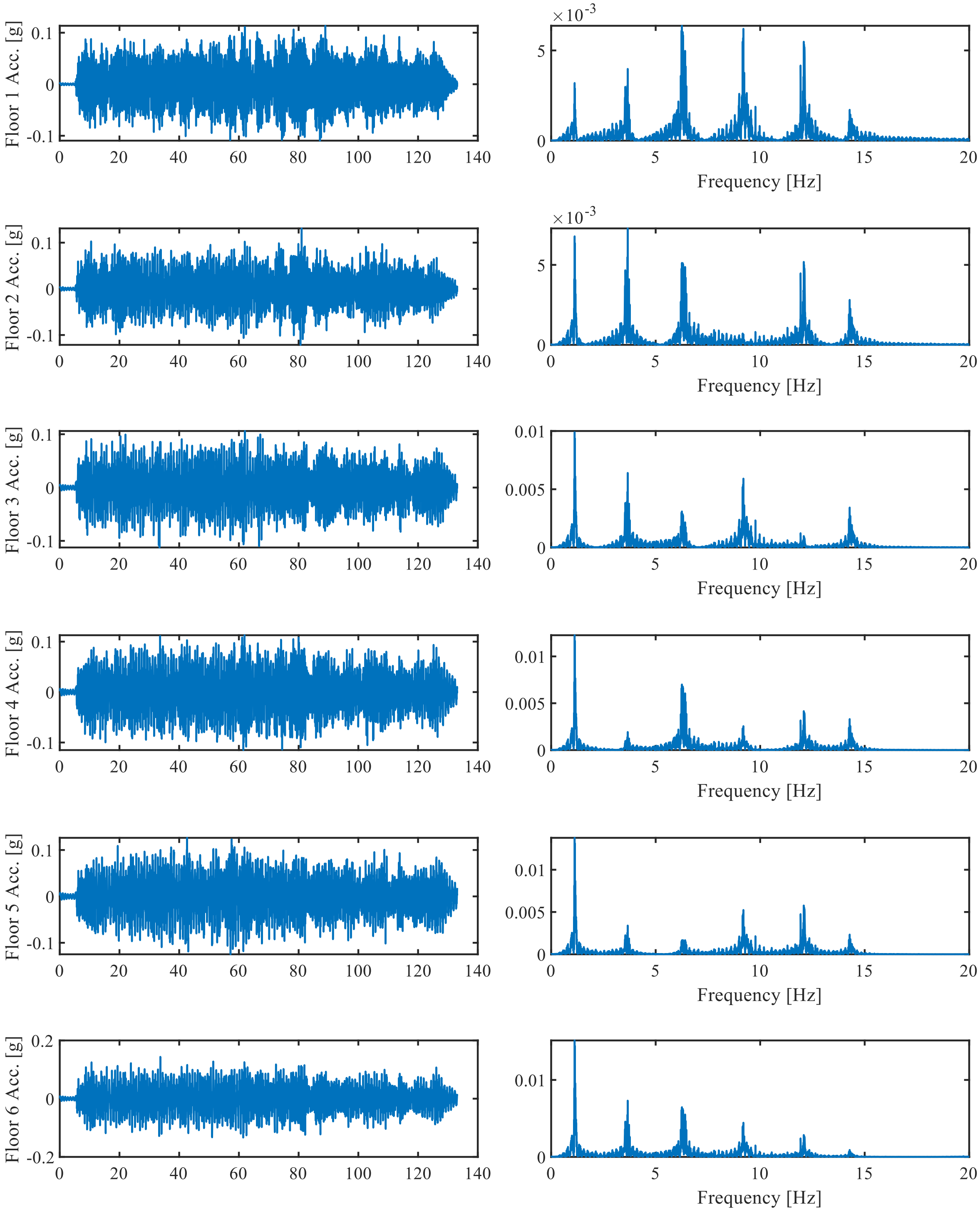
(NTS)

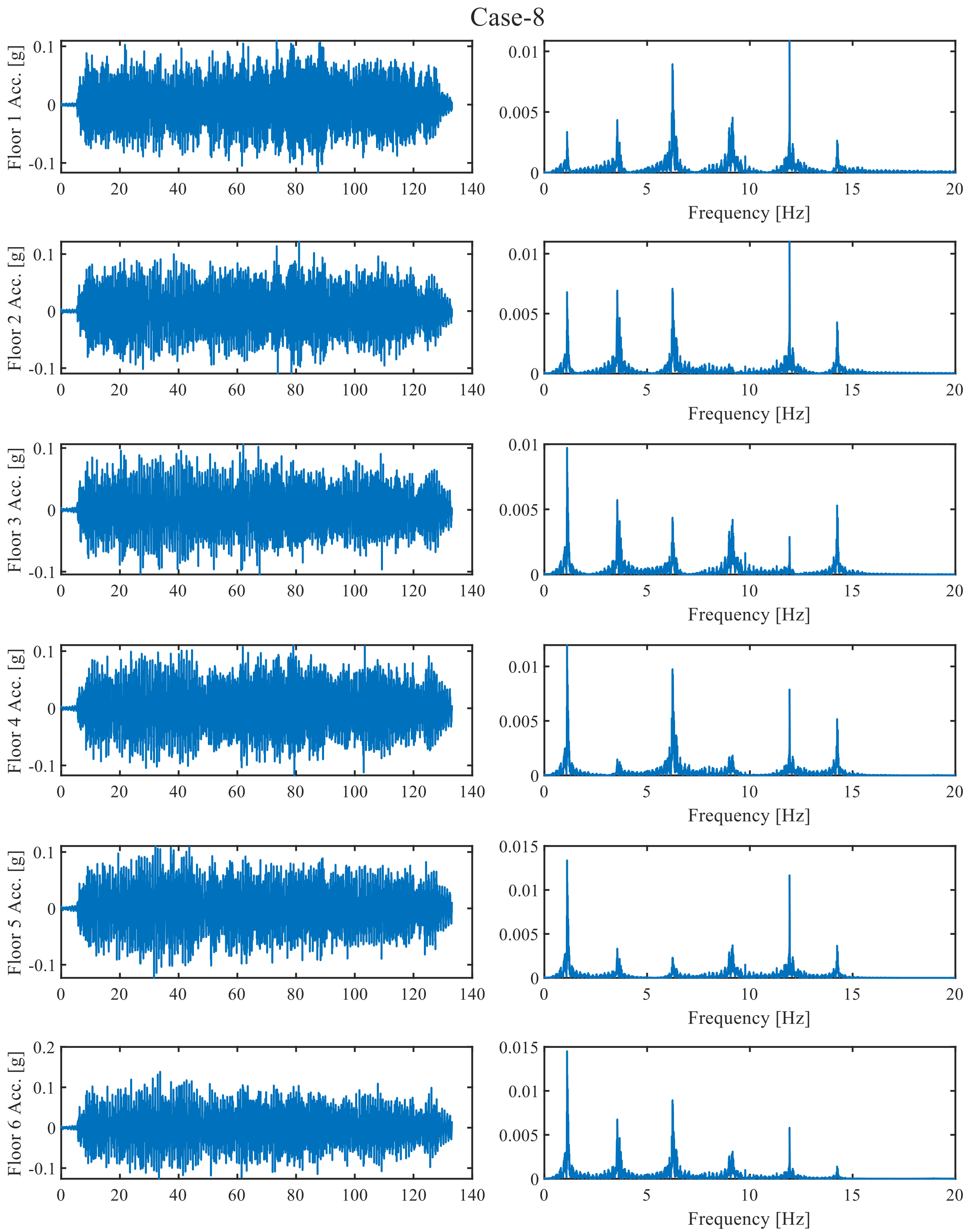
The enhanced frequency decomposition method is based on using the output correlation functions of the structure under white noise excitation to estimate the impulse response function of the structure. For this test, 2, x-direction data was given per floor (sensor locations are shown in the figure). Using the impulse response functions, the singular value decomposition algorithm is applied to a matrix of the auto and cross correlation functions in order to obtain a psuedo-eigenvalue and psuedo-eigenvector of the system for a range of frequencies. Using the peaks in the largest singular value number plotted against the frequencies of the correlation spectrums, the most prominent mode shapes for these peak modal frequencies can be found.

The natural frequencies of the structure during case 1 and case 8 are summarized below. This is also corroborated by using the fast Fourier Transform of the excitation per floor. As the signal travels through the building, the white noise is filtered through the dynamics of the system and produces sharper peaks around the building's natural frequency which are shown.

	Mode 1 [Hz]	Mode 2 [Hz]	Mode 3 [Hz]	Mode 4 [Hz]	Mode 5 [Hz]	Mode 6 [Hz]
Case 1	1.12	3.66	6.25	9.20	12.13	14.31
Case 8	1.12	3.54	6.25	9.18	11.94	14.26

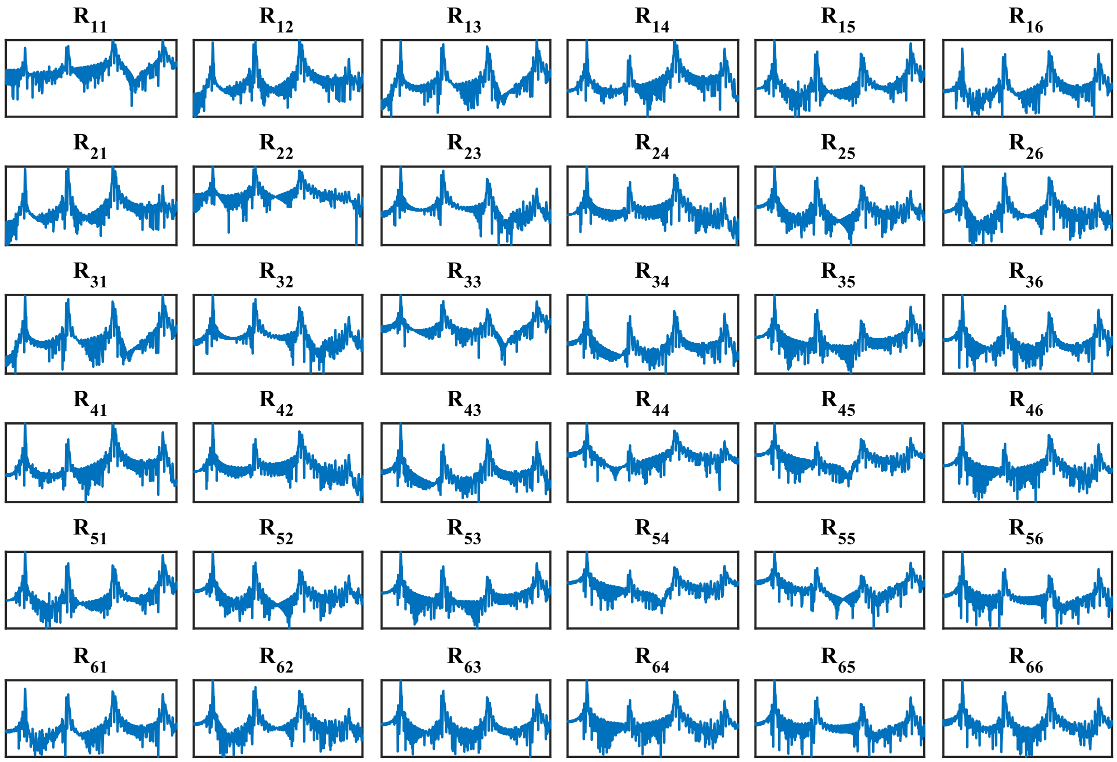
Case-1


Floor 1 Acc. [g]
Floor 2 Acc. [g]
Floor 3 Acc. [g]
Floor 4 Acc. [g]
Floor 5 Acc. [g]
Floor 6 Acc. [g]
Frequency [Hz]
×10^-3

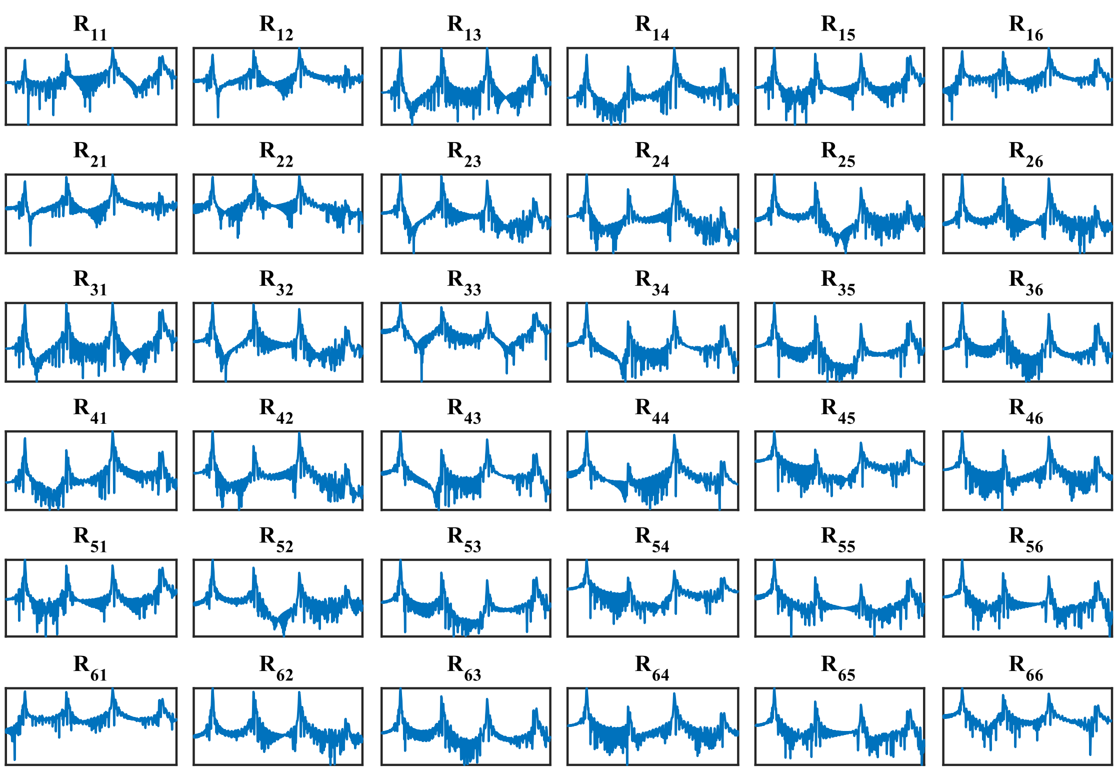


The cross-correlation spectrums are shown on the next page as a result of converting the cross and auto correlation functions through the combination of quadra and co-spectrum components.

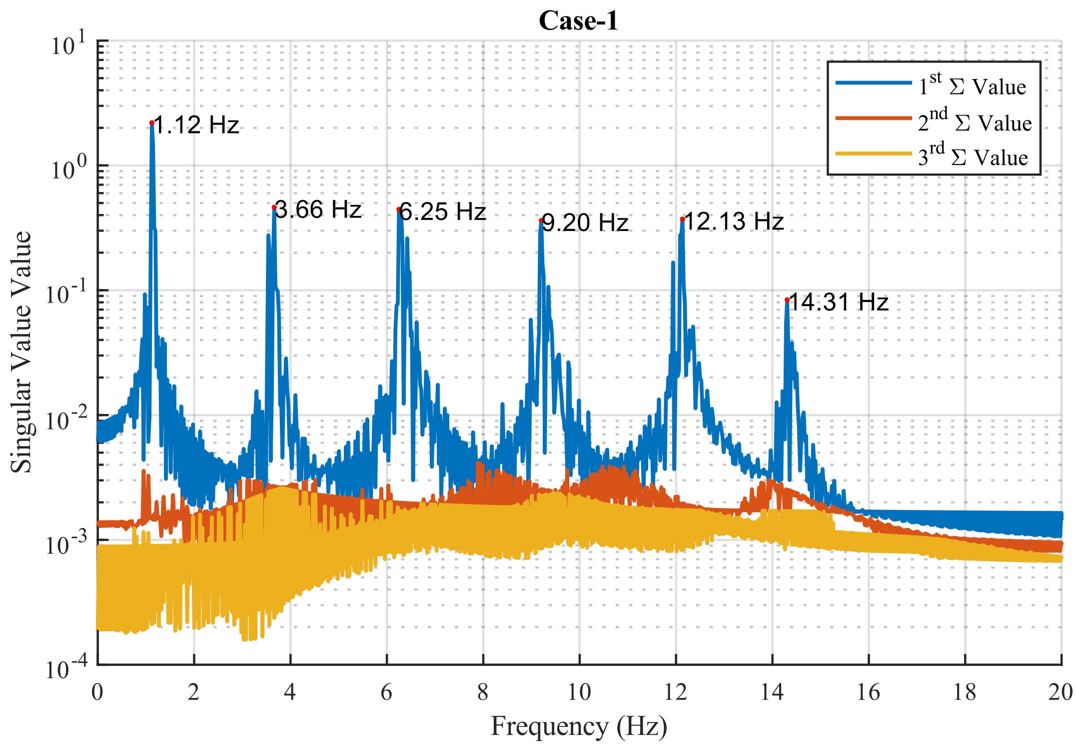
Case-1

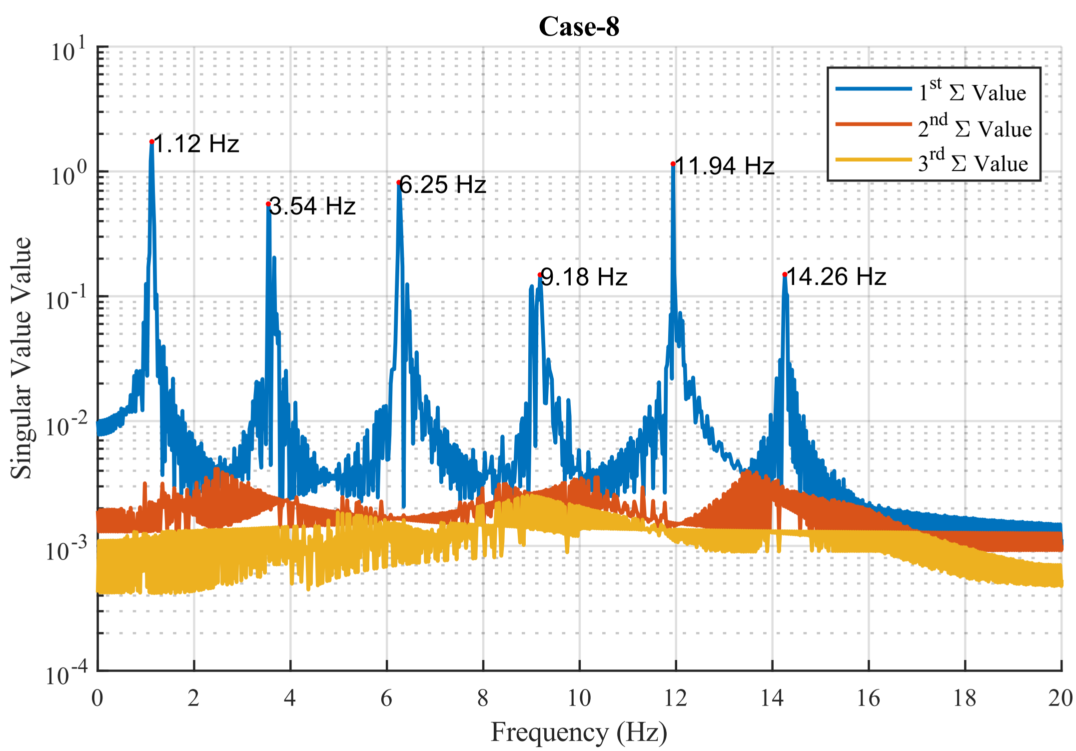

R_{11}
R_{12}
R_{13}
R_{14}
R_{15}
R_{16}
R_{21}
R_{22}
R_{23}
R_{24}
R_{25}
R_{26}
R_{31}
R_{32}
R_{33}
R_{34}
R_{35}
R_{36}
R_{41}
R_{42}
R_{43}
R_{44}
R_{45}
R_{46}
R_{51}
R_{52}
R_{53}
R_{54}
R_{55}
R_{56}
R_{61}
R_{62}
R_{63}
R_{64}
R_{65}
R_{66}

Case-8


R_{11}
R_{12}
R_{13}
R_{14}
R_{15}
R_{16}
R_{21}
R_{22}
R_{23}
R_{24}
R_{25}
R_{26}
R_{31}
R_{32}
R_{33}
R_{34}
R_{35}
R_{36}
R_{41}
R_{42}
R_{43}
R_{44}
R_{45}
R_{46}
R_{51}
R_{52}
R_{53}
R_{54}
R_{55}
R_{56}
R_{61}
R_{62}
R_{63}
R_{64}
R_{65}
R_{66}

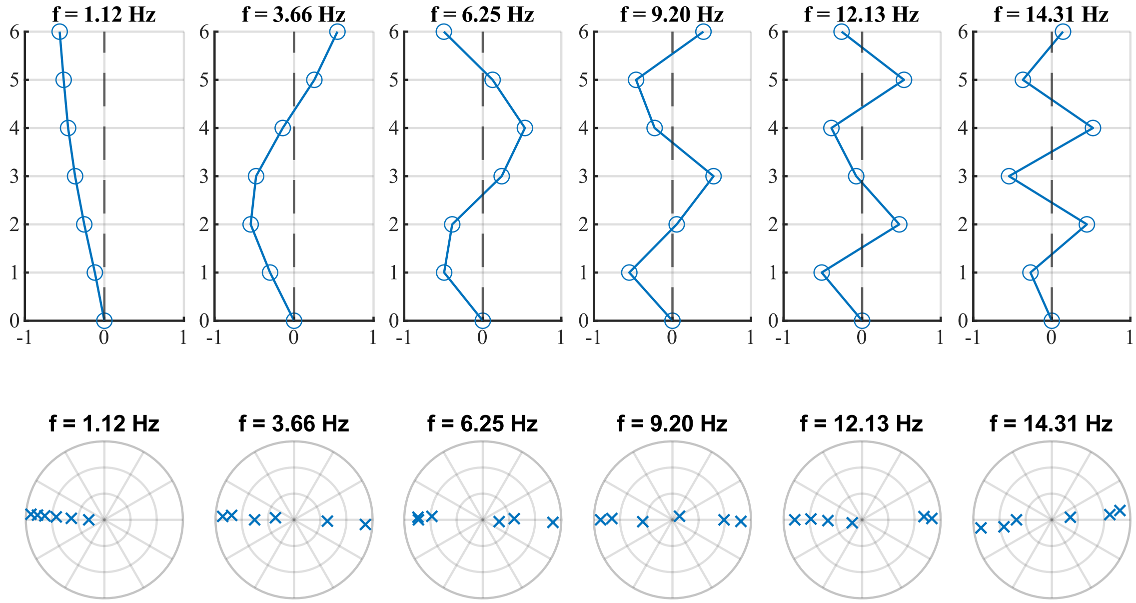
It is shown that the singular value of the first entry in the diagonal is 10^3 larger than the second or third number meaning, using only the first value to approximate the system is fine. Case 1 shows that modes 2-5 have similar singular values amplitudes whereas Case 8 shows some modes larger or smaller than their Case1 counterparts.



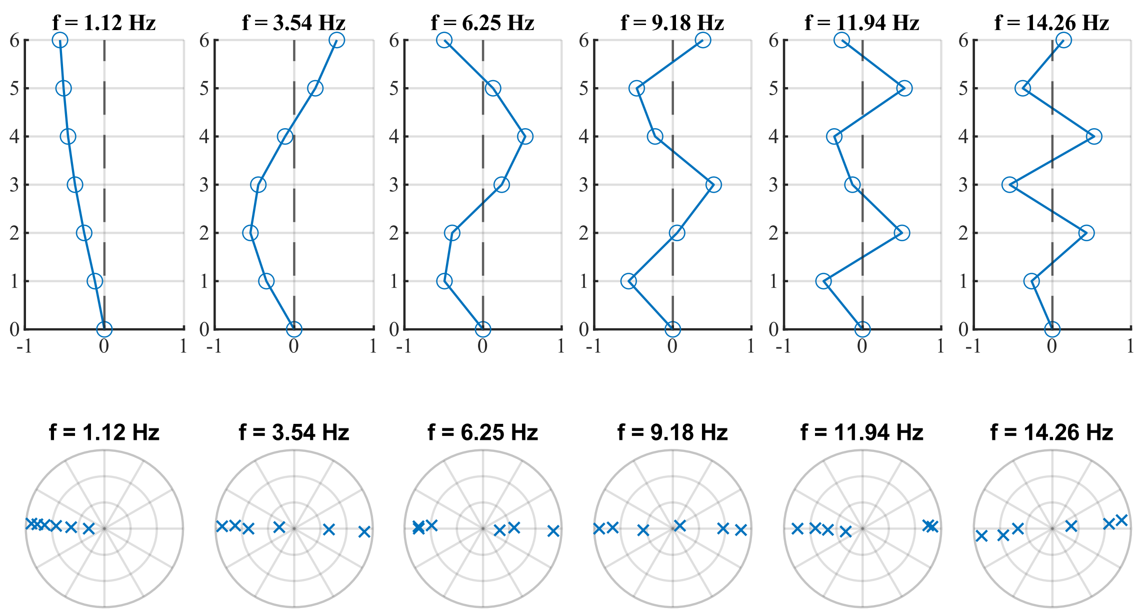


From the results, it is shown that the frequencies did not change much between case 1 and case 8 even with the cut in the first two story columns. The biggest change was in higher frequency modes but modes 1 and 3 did not see a change in their natural frequencies between the two cases. The mode shape also remained the same between the two cases. The polar plots are strongly colinear meaning this building can be considered classically damped.

Case-1



Case-8



```
%% Import Data %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear; clc;
addpath("Case-8\")
addpath("Case-1\")

fs = 200;
nFloors = 6;
ii = 65:76;
caseName = "Case-1";

for fl = 1:nFloors
    x_a = readvars(caseName+"\WN_50gal _SPECIMEN1_"+ ii(2*fl-1)+ "_A"+fl+"a__.TXT");
    x_b = readvars(caseName+"\WN_50gal _SPECIMEN1_"+ ii(2*fl)+ "_A"+fl+"b__.TXT");
    signals.("FL"+fl) = 0.5*(x_a +x_b);
end

% Plot the data
figure(1); clf;
tiledlayout(nFloors,2,"TileSpacing","compact","Padding","compact")

for fl = 1:nFloors
    signal = signals.("FL"+fl);
    nexttile()
    plot(signal)
    ylabel("Floor " + fl + " Acc. [g]")

    nexttile()
    npt = length(signal);
    npt = 2^nextpow2(npt);
    y = fft(signal,npt);
    power = abs(y)* 2/npt;
    f = (0:npt-1)*fs/npt;
    id = (1:npt/2);
    plot(f(id), power(id));
    xlim([0,20]);
end

sgtitle(caseName,'fontname','times')
print_figure(1,".\figure",caseName+" Unfiltered Data",8,6.5);
%% Data Preprocessing %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clc;
Fs = 200; % Current sampling rate
% fs = 80; % Resampling rate
n = 4;
% Wn = [0.1,20]/(fs/2);
Wn = 20/(0.5*fs);
% ftype = 'bandpass';
ftype = 'low';
```

```
[b,a] = butter(n,Wn,ftype);

figure(2); clf;
tiledlayout(nFloors,2,"TileSpacing","compact","Padding","compact")
for fl = 1:nFloors
   signal = signals.("FL"+fl);

%     signal = resample(signal, fs, Fs); % Resample the data to be 80Hz
   signal = detrend(signal);
   signal = filtfilt(b,a,signal);

   nexttile()
   plot(signal)
   ylabel("Floor " + fl + " Acc. [g]")

   nexttile()
   npt = length(signal);
   npt = 2^nextpow2(npt);
   y = fft(signal,npt);
   power = abs(y)* 2/npt;
   f = (0:npt-1)*fs/npt;
   id = (1:npt/2);
   plot(f(id), power(id));

   xlim([0,20]);
end
sgtitle(caseName,'fontname','times')
print_figure(2,".\figure",caseName+" Fltered Data",8,6.5);
%% Cross Correlation Function Matrix %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clc;
maxlag = 2^13;
Ryy = zeros(nFloors,nFloors,maxlag);
Gyy = zeros(nFloors,nFloors,maxlag);

% Calculate the correlation between signals
for ii = 1:nFloors
   for jj = 1:nFloors
      R_ii_jj = xcorr(signals.("FL"+ii),signals.("FL"+jj),maxlag, "unbiased");
      R_ii_jj = flip(R_ii_jj(1:maxlag));
      R_ii_jj(maxlag/2:end) = 0;
      Ryy(ii,jj,:) = R_ii_jj;
   end
end

% Calculate the correlation spectrums
for ii = 1:nFloors
   for jj = 1:nFloors
      lxy = 0.5* (Ryy(ii,jj,:)+Ryy(jj,ii,:)); % Co-Correlation
```

```
        qxy = 0.5* (Ryy(ii,jj,:)-Ryy(jj,ii,:)); % Quadra-Correlation
        Lxy = real(fft(lxy)); %Co-Spectrum
        Qxy = imag(fft(qxy)); % Quadra-Spectrum
        Gyy(ii,jj,:) = Lxy-Qxy*1i; % Cross Spectrum
    end
end

npt = maxlag;
fxy = (0:npt-1)* fs /npt; %
id = (1:npt/2);

figure(3); clf;
tiledlayout(nFloors,nFloors,"TileSpacing","compact","Padding","compact")

for ii = 1:nFloors
    for jj = 1:nFloors
        nexttile()
        spectrum = abs(Gyy(ii,jj,id))*2/maxlag;
        spectrum = reshape(spectrum,[],1);
        semilogy(fxy(id) ,spectrum);
        title("R_{"+ii +"" + jj+"}");
        xlim([0,10])
        xticks([])
        yticks([])
    end
end

sgtitle(caseName,'fontname','times')
print_figure(3,".\figure",caseName+" Cross Correlation",4.5,6.5);
%%  Singular Value plots %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clc;
singularValues = zeros([nFloors,maxlag]);
modeshapes = zeros([nFloors,1,maxlag]);

for ff = 1:maxlag
    [U,S,V] = svd(Gyy(:,:,ff),'econ');
    singularValues(:,ff) = diag(S);
    modeshapes(:,:,ff) = U(:,1);
end

figure(4); clf; clc;
hold on;
fxy = (0:npt-1)* fs /npt; %
id = (1:npt/2);
p = plot(fxy(id), singularValues(1:3,id)',LineWidth=1.5);
xlim([0,20]);

nPks = 6;
```

```
[pk, loc] =
findpeaks(singularValues(1,id),NPeaks=nPks,MinPeakHeight=max(singularValues(1,:))*0.01,MinPeakDistan
ce=50);

plot(fxy(loc), pk,'r.')
for nn = 1:nPks
    text(fxy(loc(nn)), pk(nn),sprintf("%.2f Hz", fxy(loc(nn))))
end

set(gca,"yscale","log")
ylabel ('Singular Value Value');
xlabel ("Frequency (Hz)"); %
legend("1^{st} \Sigma Value", "2^{nd} \Sigma Value","3^{rd} \Sigma Value")
grid
xlim([0,20])
title(caseName)
print_figure(4,".\figure",caseName+" Singular Value",4,6.5);
%% Polar Plots %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure(5); clf; clc;
tiledlayout(2,nFloors,"TileSpacing","compact","Padding","compact","TileIndexing","columnmajor")

for ss = 1:6
    nexttile;
    hold on;
    shape = [0; real(modeshapes(:,:,loc(ss)))];
    shape = shape./norm(shape);
    plot(shape,0:6, "-o");
    title(sprintf("f = %.2f Hz", fxy(loc(ss))))
    xline(0,'--');
    grid;
%     pbaspect([1,1,1])
    xlim([-1 1]*1)

    nexttile;
    theta = angle(modeshapes(:,:,loc(ss)));
    rho = abs(modeshapes(:,:,loc(ss)));
    polarplot(theta, rho,'x');
    title(sprintf("f = %.2f Hz", fxy(loc(ss))))
    Ax = gca;
    Ax.RTickLabel = [];
    Ax.ThetaTickLabel = [];

end
sgtitle(caseName,'fontname','times')
print_figure(5,".\figure",caseName+" Mode Shape + Polar Plot",4,6.5);
```