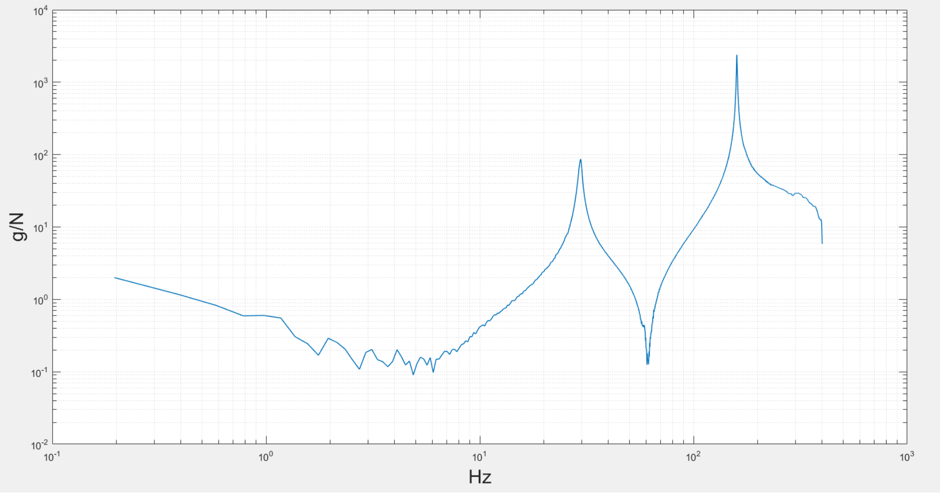
POLITECNICO DI TORINO

MSc Automotive Engineering

A.Y. 2023/24

Car Body Design

Project 3



Carlo Vittorio Colucci 329703
Riccardo Bressani 323665
Pierpaolo Placida 323197
Alessio Covetti 329876

Introduction

Modal analysis is a powerful technique exploited not only in construction engineering, but also in mechanics field, where the ability to investigate systems response to a vibrating load is fundamental.

Taking into consideration a cantilever beam subjected to a vertical excitation, this project aims to compare experimental results with a numerical model performed in Matlab®. Furthermore, recreating such analytic system enables to adjust model parameters, obtaining a more precise approximation of the real case study. In the end, the FRF (Frequency Response Function) is retrieved, allowing to better visualize the resonance phenomenon, as well as the effect of damping and Young's Modulus on its frequencies and amplitudes.

Method and results

Production of the FEM model of the beam

For the experiment, the cantilever beam assembly is organized with an accelerometer and a piezoelectric hammer. After setting geometrical and material properties on a Matlab® script, one of the key parameters to define in the analytical model is the number of elements to discretize the beam.

Length $[m]$	0,28
Mass $[kg]$	0,068
Young's Modulus $[N/m^2]$	70000e-6
Moment of inertia $[mm^4]$	6,75e-11
Number of elements	28
Frequency range of interest $[Hz]$	0 - 400

Table 1: Beam model characteristics

Following a procedure defined by the FEM (Finite Element Method) theory, the mass and stiffness matrices are created. Particular attention is paid to the accelerometer position: its mass is added to the mass matrix in the corresponding node. Due to the presence of a fixed constraint on the extremity (thus not having any degree of freedom), first node values are removed from both matrices.

Solution of the eigenproblem

Once the mass and stiffness matrices are computed it is possible to solve the undamped eigenvalue problem. The result of the homogeneous associated equation defines the resonance frequencies and the modal shapes of the analysed system. The complete equation taken into account is:

$$[M] \cdot \{\ddot{q}\} + [C] \cdot \{\dot{q}\} + [K] \cdot \{q\} = \{F\} \tag{1}$$

$$\{q\} = \{q_0\} \cdot e^{j\omega \cdot t}$$

Considering:

- Undamped system $[C] = 0$
- Homogeneous associated problem $\{F\} = 0$

Rearranging the equations, the following eigenvalue problem is solved:

$$\big[-[M] \cdot \omega^2 + [K]\big] \cdot \{q_0\} = \{0\}$$

There are two possible solutions for the problem, the trivial $\{q_o\} = 0$ and a more complex one that is evaluated by studying the determinant of the matrix:

$$det\big[-[M] \cdot \omega^2 + [K]\big] = 0$$

The solution of the eigenproblem provides the eigenvectors and the eigenvalues of the matrix $[M]^{-1} \cdot [K]$.
The obtained eigenvalues ω_i are the resonance frequencies of the system, meanwhile the eigenvectors correspond to the modal shapes of the beam. Since each node of the discretized beam is characterised by two degrees of freedom, the possible modal shapes of the system are 56. The case study limits the range of frequencies from 0 to 400 Hz, thus only the

first two modal shapes are taken into account. Indeed, the resonance frequency of the third modal shape does not belong to the considered frequency range.

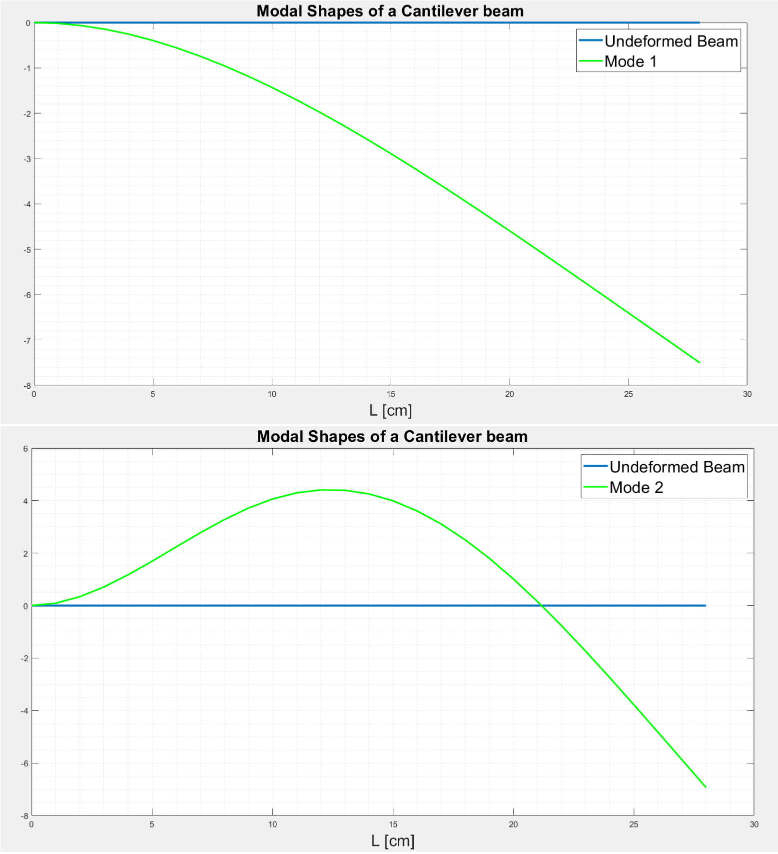


Figure 1: Modal Shapes at 30 and 160,4 Hz respectively

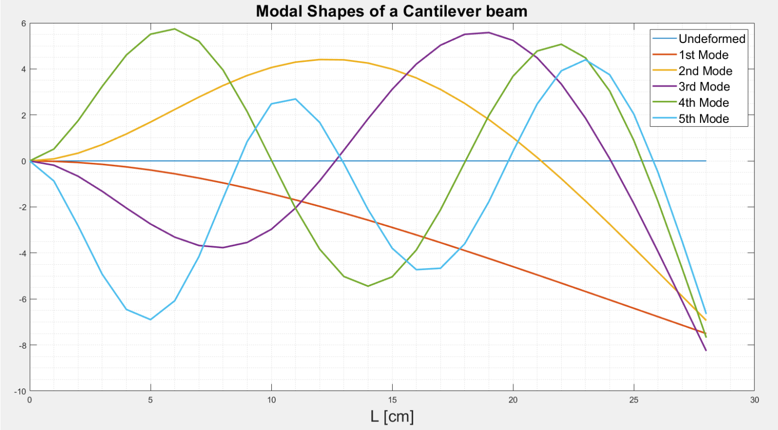


Figure 2: First five Modal Shapes of the beam

Modal analysis for the definition of the modal damping matrix

Once having found the latter quantities, the next step is to evaluate the damping matrix. Since it is not possible to retrieve it directly, the modal damping matrix is firstly obtained by applying the modal expansion theorem. A coordinate transformation is performed, switching from the canonical space to the space of the modal shapes:

$$\{q\} = [U] \cdot \{\xi\}$$

where $[U]$ is the modal base matrix and $\{\xi\}$ is the vector of the solutions in such space. Pre-multiplying equation (1) by $[U]^T$ it is possible to write:

$$[U]^T \cdot [M] \cdot [U] \cdot \{\ddot{\xi}\} + [U]^T \cdot [C] \cdot [U] \cdot \{\dot{\xi}\} + [U]^T \cdot [K] \cdot [U] \cdot \{\xi\} = [U]^T \cdot \{F\}$$

The represented quantities define the modal mass and stiffness matrices:

$$[U]^T \cdot [M] \cdot [U] = [I]$$

$$[U]^T \cdot [K] \cdot [U] = [\Omega]$$

$$[U]^T \cdot [C] \cdot [U] = [C]_{modal}$$

where:

$$[\Omega] = diag[\omega_i^2]$$

and the modal damping matrix is defined as:

$$[C]_{modal} = diag[C_{i,modal}] = \zeta_{modal} \cdot diag[C_{i,critical}]$$

The critical damping $C_{i,critical}$ is obtained from the modal mass and stiffness matrices previously evaluated:

$$C_{i,critical} = 2 \cdot \sqrt{K_{i,modal} \cdot M_{i,modal}}$$

It is now possible to retrieve the damping matrix:

$$[C] = ([U]^T)^{-1} \cdot [C]_{modal} \cdot [U]^{-1}$$

Finally, in order to plot the Bode diagram of the system, it is necessary to retrieve the latter in state space representation. To obtain this formulation, it is necessary to make a first assumption regarding the damping coefficient ζ_{modal}. Thus, the A,B,C,D matrices are computed:

$$\begin{cases} \dot{x} &= A \cdot x + B \cdot u \\ y &= C \cdot x + D \cdot u \end{cases}$$

Representing respectively:

- Dynamic matrix $[A]$;
- Input matrix $[B]$;
- Output matrix $[C]$;
- Direct input/output matrix $[D]$.

Tuning of the model parameters through a comparison with experimental results

Once the state space is computed, the analytical results can be compared with the experimental ones obtained in the laboratory.

The following representation is obtained with a damping coefficient $\zeta_{modal} = 0{,}2$ and a Young's Modulus $E = 70000 \cdot 10^6 N/m^2$:

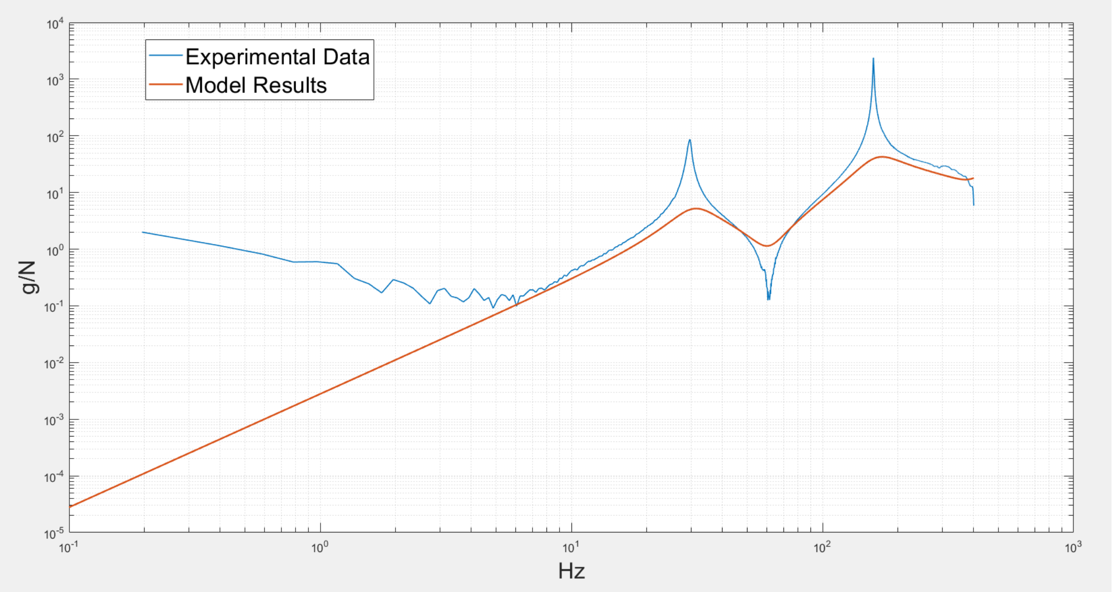


Figure 3: Bode diagram comparing experimental and analytical curves.

With this first assumption, it is possible to notice that the resonance frequencies of the two systems are not the same. Furthermore, the peak values in correspondence of the excitation frequencies are different. Indeed, the system is overdamped and is characterised by an higher stiffness with respect to the laboratory model.

The system must be tuned to achieve a proper comparison of the results. The two main parameters that affect the stiffness and the damping of the analytical model are the Young's Modulus and the

damping coefficient respectively.
With respect to the first case, new parameters are chosen:

- Young's Module $E = 65000 \cdot 10^6 N/m^2$

  which will affect resonance frequencies.
- Damping coefficient $\zeta = 0.015$

  influencing peak values.

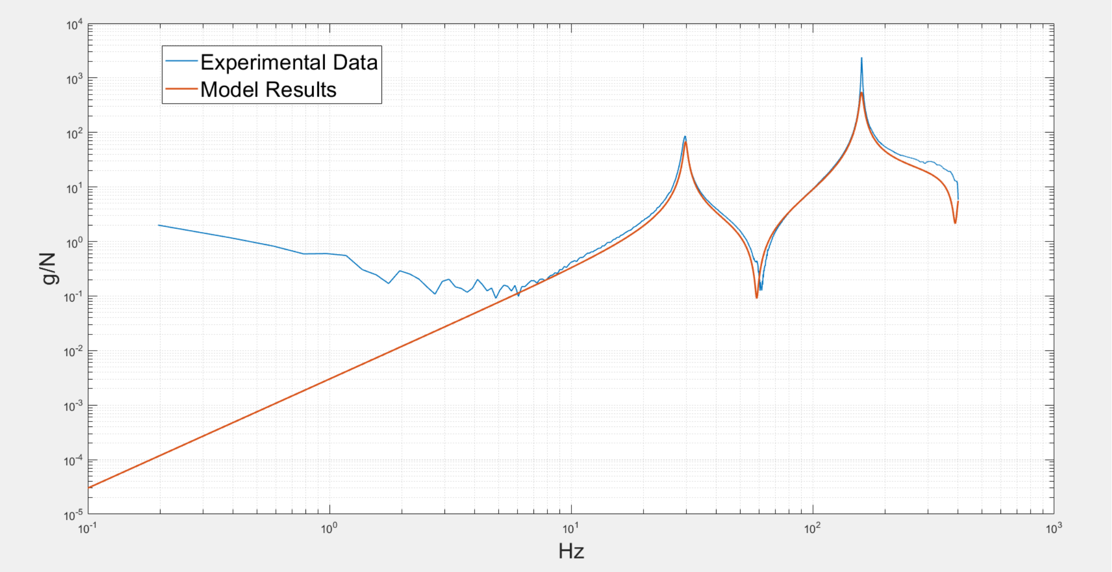


Figure 4: Bode diagram with tuned parameters.

It is possible to observe how the latter values provide a good approximation of the experimental results.

Conclusions

The result of the project is a complete construction of an analytical model for the so mentioned cantilever beam.
The comparison with experimental results allows to validate the numerical architecture performed in Matlab®, by means of iterative parameters tuning.
In the end, the advantage of having such numerical model is to predict what could be the system response when loaded in a different manner. Indeed, equivalent analytical architectures are particularly useful for those systems that could be physically expensive to represent in reality.

Appendix

```
%%
% Car body design and aerodynamics
% a.a. 2023-2024
% Exercise 3 - FEM modeling and modal analysis of a cantilever beam
%

%%
format short e
clc
clear
close all
%% INITIAL VALUES
L               = 280e-3;                                                    %
[m]       Length of the beam
b               = 30e-3;                                                     %
[m]       Width of the beam
h               = 3e-3;                                                      %
[m]       Thickness of the beam
ro              = 2700;                                                      %
[kg/m^3]  Density  of the aluminum
massa_trave     = ro*b*h*L;                                                  %
[kg]      Mass of the beam
E=70000e6;                                                                 % [N/
m^2]   Young modulus of the aluminum
%E               = 65000e6;  % tuned
Poiss           = 0.33;                                                      %
[-]       Poisson coefficient
G               = E/(2*(1+Poiss));                                           % [N/
m^2]   Shear coefficient
n_el            = 28;                                                        %
[-]       Number of elements
l               = L/n_el;                                                    %
[m]       Length of a single element

m_acc           = 18e-3;                                                     %
[kg]      Mass of the accelerometer
L_acc           = 100e-3;                                                    %
[m]       Distance from the clamping point

L_hammer        = 100e-3;                                                    %
[-]       Distance hammering point from the clamping end

n_n             = n_el+1;                                                    %
[-]       Number of nodes
n_hammer        = L_hammer/l+1;                                              %
[-]       Node at which the hammer is applied
n_acc           = L_acc/l+1;                                                 %
[-]       Node where the accelerometer is applied

A_matrix        = b*h;                                                       %
[m^2]     Beam section area
```

```
I_y               = b*h^3/12;                                              %
[m^4]     Moment of inertia of the section area

chi               = (12+11*Poiss)/10/(1+Poiss);                             %
[-]       Shear factor of a rectangular section

omega             = 0.1:0.01:400;                                          %
[Hz]      Frequency range of interest
omega             = omega';

%% Mass and stiffness matrix creation
% Behaviour in the plane orthogonal to plane of the beam node and reference node
PHI1              = 12*E*I_y*chi/G/A_matrix/l^2;                              %
Coefficient which takes into account the shear deformation of the section in the
plane orthogonal to plane of the beam node and reference node
m1                = 156+294*PHI1+140*PHI1^2;
m2                = 22+38.5*PHI1+17.5*PHI1^2;
m3                = 54+126*PHI1+70*PHI1^2;
m4                = 13+31.5*PHI1+17.5*PHI1^2;
m5                = 4+7*PHI1+3.5*PHI1^2;
m6                = 3+7*PHI1+3.5*PHI1^2;
m7                = 36;
m8                = 3-15*PHI1;
m9                = 4+5*PHI1+10*PHI1^2;
m10               = 1+5*PHI1-5*PHI1^2;

m_F1_1            = ro*A_matrix*l/(420*(1+PHI1)^2)*[ m1         l*m2
m3           -l*m4
                                                l*m2     l^2*m5      l*m4
-l^2*m6
                                                m3      l*m4         m1           -l*m2
                                                -l*m4   -l^2*m6      -l*m2
l^2*m5];

m_F1_2            = ro*I_y/(30*l*(1+PHI1)^2)*[m7         l*m8         -m7
l*m8
                                                l*m8    l^2*m9       -l*m8
-l^2*m10
                                                -m7     -l*m8        m7           -l*m8
                                                l*m8    -l^2*m10     -l*m8
l^2*m9];

m_F               = m_F1_1+m_F1_2;

k_F               = E*I_y/(l^3*(1+PHI1))*[12        6*l                -12
6*l
                                        6*l     (4+PHI1)*l^2    -6*l         (2-
PHI1)*l^2
                                        -12     -6*l            12           -6*l
                                        6*l     (2-PHI1)*l^2    -6*l
(4+PHI1)*l^2];

M                 = zeros(2*n_n);     % dof= 2* number of nodes
```

```
MM              = zeros(2*n_n);     % temporary matrix for the evaluation of
matrix M
K               = zeros(2*n_n);
KK              = zeros(2*n_n);

for t=1:1:n_el % this cycle adds a 4x4 matrix to M and K at each iteration
    MM  = zeros(2*n_n);
    KK  = zeros(2*n_n);
    qq  = 2*(t-1)+1;       % counter
    qq1 = qq+3;               % counter

    MM(qq:qq1,qq:qq1) = m_F;   % on the diagonal of the matrix M and k
    KK(qq:qq1,qq:qq1) = k_F;
    M=M+MM;
    K=K+KK;
end

% Addition of the accelerometer mass at the proper node
M(2*n_acc-1,2*n_acc-1) = M(2*n_acc-1,2*n_acc-1)+m_acc;
% M(21,21) is the DOF referred to the vertical motion, dependent on the vertical
mass of that node.

% constraints of the degrees of freedom
nf              = size(K,1);   % number of dof of the nodes
Mc              = M(3:nf,3:nf);                                             %
Constrained mass matrix
Kc              = K(3:nf,3:nf);                                             %
Constrained stiffness matrix
%the node on the constrain has 0 mass and o stiffness

%% MODE SHAPES

[eigenvector,eigenvalue]=eig(Kc,Mc);
new_mode_shapes=[];

i_autoval = 5;
eigenvalue_Hz=[];

for i_plot= 1:i_autoval

    eigenvalue_Hz=[eigenvalue_Hz (sqrt(eigenvalue(i_plot,i_plot)))/(2*pi)];

end

for i_plot= 1:(length(eigenvector))
    if i_plot==1
        new_mode_shapes = [new_mode_shapes; zeros(1,length(eigenvector))];
    end

    if rem(i_plot,2) == 1   % counter odd
        new_mode_shapes = [new_mode_shapes; eigenvector(i_plot,:)];
    end
end
```

```
mode_shapes_plot=2;
for i_plot= 1: mode_shapes_plot

    l_l_plot=linspace(0,L*100,(n_n));
    beam_plot = zeros(1,length(l_l_plot));

    figure;
    plot(l_l_plot,beam_plot, LineWidth=2.5); hold on;
    plot(l_l_plot,-(new_mode_shapes(:,i_plot)),'g',LineWidth=2); hold on;
    numero_modo=i_plot;
    ddd=sprintf('Mode %g ', numero_modo );

    legend("Undeformed Beam", ddd, fontsize=20 )
    xlabel("L [cm]",FontSize=20)
    title("Modal Shapes of a Cantilever beam", FontSize=20)
    grid minor
    hold off
end

l_l=linspace(0,L*100,(n_n));
figure;
beam = zeros(1,length(l_l));
plot(l_l,beam, LineWidth=1); hold on;
plot(l_l,-(new_mode_shapes(:,1)),LineWidth=2); hold on;
plot(l_l,-(new_mode_shapes(:,2)),LineWidth=2); hold on;
plot(l_l,-(new_mode_shapes(:,3)),LineWidth=2); hold on;
plot(l_l,-(new_mode_shapes(:,4)),LineWidth=2); hold on;
plot(l_l,-(new_mode_shapes(:,5)),LineWidth=2); hold on;

legend("Undeformed","1st Mode","2nd Mode","3rd Mode", "4th Mode","5th
Mode",FontSize=15)
xlabel("L [cm]",FontSize=20)
title("Modal Shapes of a Cantilever beam",FontSize=20)
grid minor
hold off

%% MODAL ANALYSIS

Mc_modal=(eigenvector'*Mc*eigenvector);
Kc_modal=(eigenvector'*Kc*eigenvector);

C_critical=[];
for i_plot= 1:length(Mc)
    Cc_value = 2*sqrt(Kc_modal(i_plot,i_plot)*Mc_modal(i_plot,i_plot));
    C_critical(i_plot,i_plot) = Cc_value;
end
%0.018
zeta = 0.2; % TO BE TUNED
C_modal = zeta*C_critical;
Cc  = ((eigenvector')^-1)*C_modal*(eigenvector)^-1;

%% State Space

M_inv=Mc^-1;
```

```
n_in=n_acc;
A = [-M_inv*Cc   -M_inv*Kc; eye(length(Kc))   zeros(length(Kc))];
B = [M_inv; zeros(length(Kc))];
C = A((n_in-1)*2-1,:);
D = B((n_in-1)*2-1,:);

omega_rad = omega*2*pi; %[rad/s]
sys=ss(A,B,C,D);
sys=sys((n_acc-1)*2-1);

[mag,phase] = bode(sys,omega_rad);
mag = squeeze(mag);

%% Comparison with Experimental data
experimental_data = load("Sq11_sn.mat");
y_values = experimental_data.FRF_Point2_Point1.y_values.values(1,:);

x_axis = 0;
for i_plot= 2:experimental_data.FRF_Point2_Point1.x_values.number_of_values
    new_value =
(x_axis(i_plot-1)+experimental_data.FRF_Point2_Point1.x_values.increment);
    x_axis = [x_axis new_value];
end

figure;
loglog(x_axis,abs(y_values),linewidth=1); hold on;
loglog(omega,mag, linewidth=1.5)
grid minor
xlabel(experimental_data.FRF_Point2_Point1.x_values.quantity.label,FontSize=20)
ylabel(experimental_data.FRF_Point2_Point1.y_values.quantity.label,FontSize=20)
legend("Experimental Data","Model Results",FontSize=20)
```