Contents

```
% In the name of Allah the beneficent the merciful
% Code by NVE Team, Sharif University, Tehran, Iran
% Date : 1395/12/24
```

Cleaning

```
clc; clear; close all
```

Optimization

Constraints

```
lb = [1e4;1e4;1e4;1e4;1e4;1e4;1e4;1e4;1e4;0;0;0;0;0;0;0;0;0;-.2;-.2;-.2;-.2;-.2;-.2;-
.2;-.2;-.2;50];
ub =
[1e6;1e6;1e6;1e6;1e6;1e6;1e6;1e6;1e6;pi/2;pi/2;pi/2;pi/2;pi/2;pi/2;pi/2;pi/2;pi/2;-
.1;-.1;.2;.2;.2;.2;.2;.2;.2;70];

x0 = lb*10.5;
x0(1:9) = 5e4;
x0(end)=50;
f=fmincon(@myfun,x0,[],[],[],[],lb,ub);
```

```
Solver stopped prematurely.

fmincon stopped because it exceeded the function evaluation limit,
options.MaxFunEvals = 3000 (the default value).
```

Variables

```
global m I T_Amp T_Freq T_phase
T_Amp = 100;     % N/m
T_Freq = 50;     % Rad/s
T_phase = 0;     % Radian
```

Mass Matrix

```
m = 150; % mass of the engine or powertrain
I = [5.82 -0.82 0.19 ; -0.82 3.41 -0.21 ; 0.19 -0.21 5.50];     % Inertia matrix
M = [m*eye(3) zeros(3) ; zeros(3) I];
```

Stiffness Matrix

Mount Positions

```
r_1 = f(19:21);
r_2 = f(22:24);
r_3 = f(25:27);
B_1 = [0 -r_1(3) r_1(2) ; r_1(3) 0 -r_1(1) ; -r_1(2) r_1(1) 0];
B_2 = [0 -r_2(3) r_2(2) ; r_2(3) 0 -r_2(1) ; -r_2(2) r_2(1) 0];
B_3 = [0 -r_3(3) r_3(2) ; r_3(3) 0 -r_3(1) ; -r_3(2) r_3(1) 0];

% Mount Inclinations
o_1 = f(10:12);
o_2 = f(13:15);
o_3 = f(16:18);

% Mount Rotation Matrices
A_1 = [cos(o_1(3))*cos(o_1(2)) -
sin(o_1(3))*cos(o_1(1))+cos(o_1(3))*sin(o_1(2))*sin(o_1(1))
sin(o_1(3))*sin(o_1(1))+cos(o_1(3))*sin(o_1(2))*cos(o_1(1));
    sin(o_1(3))*cos(o_1(2))
cos(o_1(3))*cos(o_1(1))+sin(o_1(3))*sin(o_1(2))*sin(o_1(1)) -
cos(o_1(3))*sin(o_1(1))+sin(o_1(3))*sin(o_1(2))*cos(o_1(1));
    -sin(o_1(2)) cos(o_1(2))*sin(o_1(1)) cos(o_1(2))*cos(o_1(1))];
A_2 = [cos(o_2(3))*cos(o_2(2)) -
sin(o_2(3))*cos(o_2(1))+cos(o_2(3))*sin(o_2(2))*sin(o_2(1))
sin(o_2(3))*sin(o_2(1))+cos(o_2(3))*sin(o_2(2))*cos(o_2(1));
    sin(o_2(3))*cos(o_2(2))
cos(o_2(3))*cos(o_2(1))+sin(o_2(3))*sin(o_2(2))*sin(o_2(1)) -
cos(o_2(3))*sin(o_2(1))+sin(o_2(3))*sin(o_2(2))*cos(o_2(1));
    -sin(o_2(2)) cos(o_2(2))*sin(o_2(1)) cos(o_2(2))*cos(o_2(1))];
A_3 = [cos(o_3(3))*cos(o_3(2)) -
sin(o_3(3))*cos(o_3(1))+cos(o_3(3))*sin(o_3(2))*sin(o_3(1))
sin(o_3(3))*sin(o_3(1))+cos(o_3(3))*sin(o_3(2))*cos(o_3(1));
    sin(o_3(3))*cos(o_3(2))
cos(o_3(3))*cos(o_3(1))+sin(o_3(3))*sin(o_3(2))*sin(o_3(1)) -
cos(o_3(3))*sin(o_3(1))+sin(o_3(3))*sin(o_3(2))*cos(o_3(1));
    -sin(o_3(2)) cos(o_3(2))*sin(o_3(1)) cos(o_3(2))*cos(o_3(1))];

% Mount Stiffness
k_l_1 = diag([f(1) f(4) f(7)]);
```

```
k_l_2 = diag([f(2) f(5) f(8)]);
k_l_3 = diag([f(3) f(6) f(9)]);
k_1 = A_1*k_l_1*A_1';
k_2 = A_2*k_l_2*A_2';
k_3 = A_3*k_l_3*A_3';

% Finally! The Stiffness Matrix
K = [k_1 k_1*B_1' ; (k_1*B_1')' B_1*k_1*B_1'] + [k_2 k_2*B_2' ; (k_2*B_2')'
B_2*k_2*B_2'] + [k_3 k_3*B_3' ; (k_3*B_3')' B_3*k_3*B_3'];
```

Damping Matrix

Mount Damping Coefficients

```
c_l_1 = diag([94.6 111.3 92.4])*10;
c_l_2 = diag([72.8 72.8 84.4])*10;
c_l_3 = diag([203.9 41.7 82.0])*10;
c_1 = A_1*c_l_1*A_1';
c_2 = A_2*c_l_2*A_2';
c_3 = A_3*c_l_3*A_3';

% Damping Matrix
C = [c_1 c_1*B_1' ; (c_1*B_1')' B_1*c_1*B_1'] + [c_2 c_2*B_2' ; (c_2*B_2')'
B_2*c_2*B_2'] + [c_3 c_3*B_3' ; (c_3*B_3')' B_3*c_3*B_3'];
```

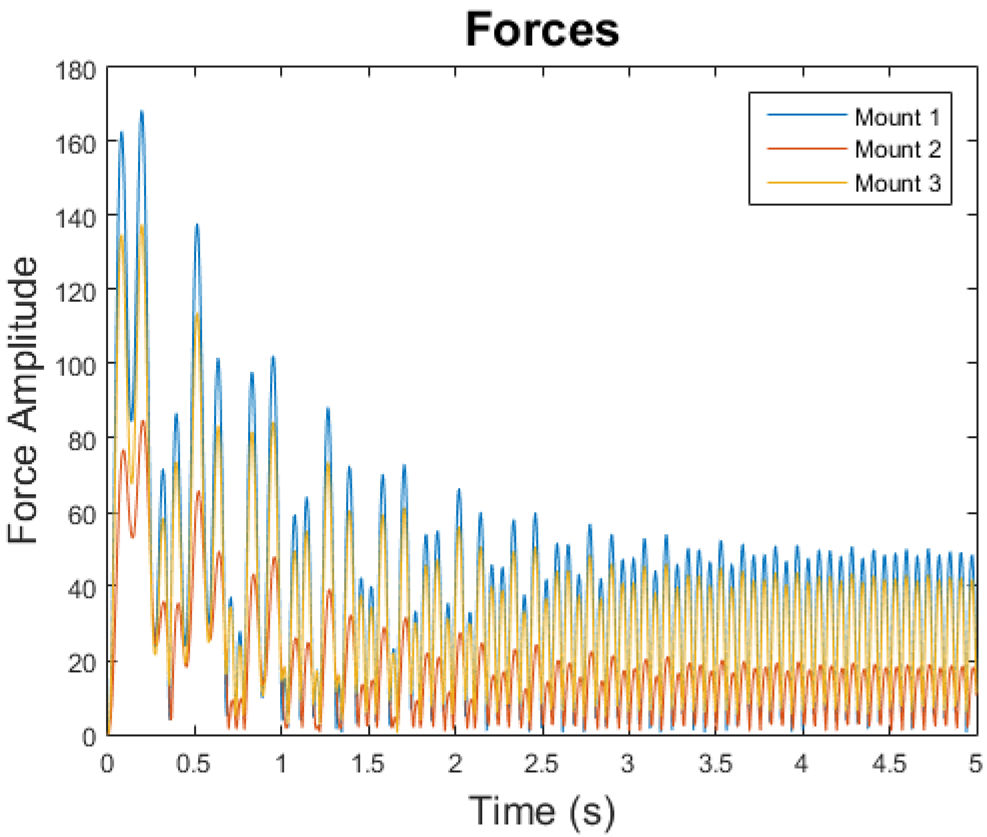
Execution

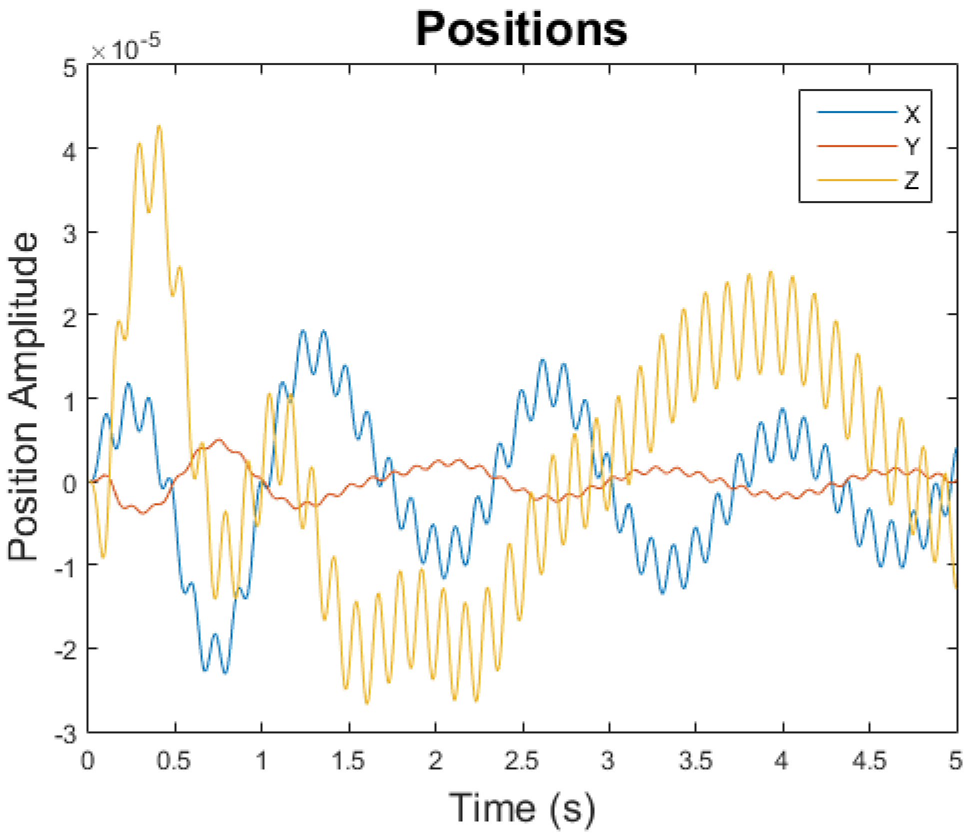
```
x0 = [0;0;0;0;0;0;0;0;0;0;0;0];     % The initial condition
[t,x] = ode45(@eng_mount, [0 5], x0, [], M, C, K);      % solving the ODE with the
duration of 5 seconds
P1 = cube([0 0 0],[0 0 0]);     % plotting the engine in its equilibrium position
```

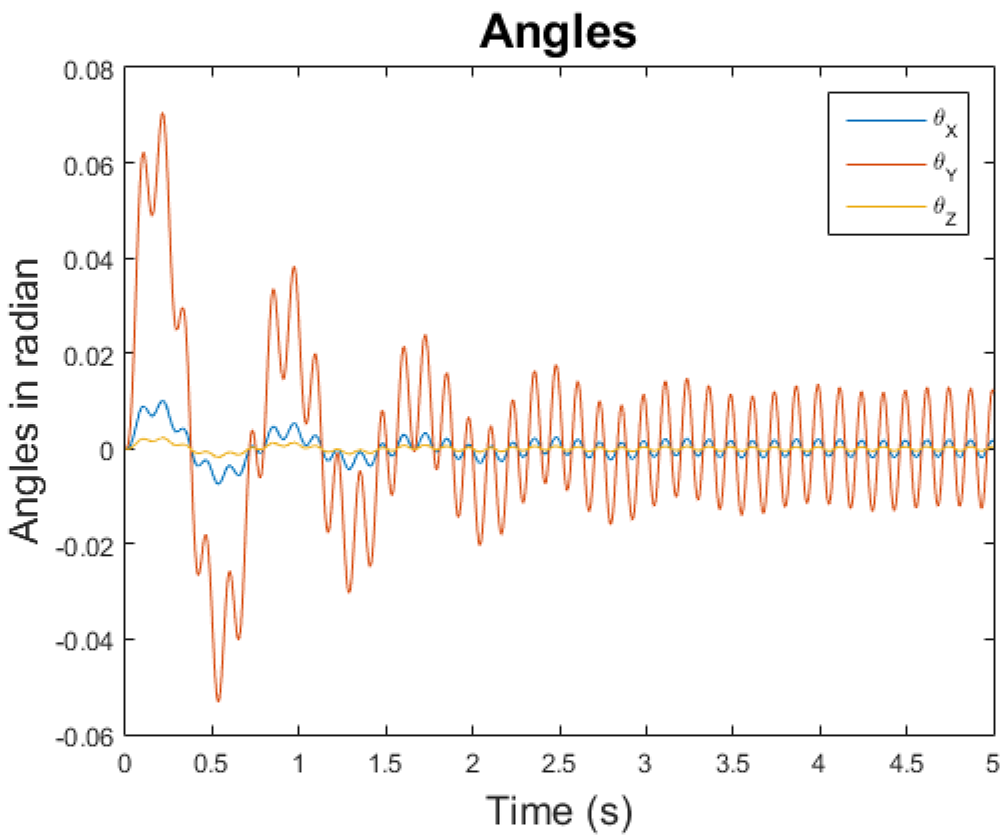
Time Results

```
F_1 = zeros(length(t),3);   F_2 = F_1;  F_3 = F_1;
F_1_n = t';  F_2_n = F_1_n;   F_3_n = F_1_n;

for i = 1:length(t)
    F_1(i,:) = (-k_1*[eye(3) B_1']*x(i,1:6)' - c_1*[eye(3) B_1']*x(i,7:12)')';
    F_1_n(i) = norm(F_1(i,:));
    F_2(i,:) = (-k_2*[eye(3) B_2']*x(i,1:6)' - c_2*[eye(3) B_2']*x(i,7:12)')';
    F_2_n(i) = norm(F_2(i,:));
    F_3(i,:) = (-k_3*[eye(3) B_3']*x(i,1:6)' - c_3*[eye(3) B_3']*x(i,7:12)')';
    F_3_n(i) = norm(F_3(i,:));
end

% Plots
```

```
figure;plot(t,F_1_n)
hold on
plot(t,F_2_n)
plot(t,F_3_n)
title('Forces','fontsize',18);
xlabel('Time (s)','fontsize',15);
ylabel('Force Amplitude','fontsize',15);
legend('Mount 1','Mount 2','Mount 3');

figure;plot(t,x(:,1:3))
title('Positions','fontsize',18);
xlabel('Time (s)','fontsize',15);
ylabel('Position Amplitude','fontsize',15);
legend('X','Y','Z');

figure;plot(t,x(:,4:6))
title('Angles','fontsize',18);
xlabel('Time (s)','fontsize',15);
ylabel('Angles in radian','fontsize',15);
legend('\theta_X','\theta_Y','\theta_Z');
```

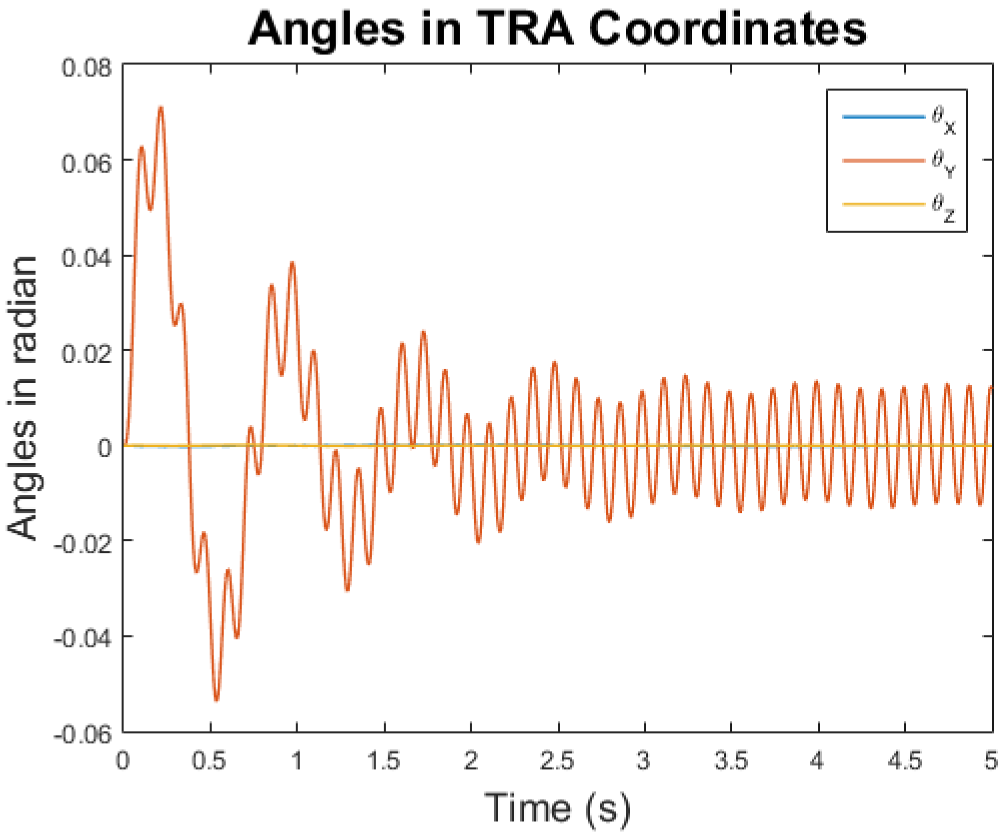

Forces
Force Amplitude
Time (s)
0
20
40
60
80
100
120
140
160
180
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
Mount 1
Mount 2
Mount 3


Positions
×10^-5
Position Amplitude
Time (s)
-3
-2
-1
0
1
2
3
4
5
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
X
Y
Z



Finding TRA

```
[q_TRA,R_TRA] = TRA_finder(I,[0;1;0]);
EE = zeros(length(x),3);
for i = 1:length(x)
EE(i,:) = (R_TRA*x(i,4:6)')';
end

% Plot
figure;plot(t,EE,'linewidth',1)
title('Angles in TRA Coordinates','fontsize',18);
xlabel('Time (s)','fontsize',15);
ylabel('Angles in radian','fontsize',15);
legend('\theta_X','\theta_Y','\theta_Z');

% End of code
```


Angles in TRA Coordinates
Angles in radian
Time (s)
0.08
0.06
0.04
0.02
0
-0.02
-0.04
-0.06
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
θ_X
θ_Y
θ_Z