

```

1 %% POST PROCESSOR FOR HW 1 (SE 201B: NONLINEAR STRUCTURAL ANALYSIS)
2
3 % Run this file in the same folder as run.tcl
4
5 % Run this file after running the code run.tcl in OpenSees. The text
6 % files generated in the process will be loaded and used for plotting.
7
8 close all
9 clc
10
11 %% INPUT
12 matDef = 'uniaxialMaterial Steel02 1 29.332482664456787 30000 0.02 3. 0.5 0.15 0. 1. 0. 1. 0.';
13 L = 60;
14 A = 10.216981781666002;
15 localOpenSeesPath = "C:\Users\Louis Lin\Workspace\Academic\UCSD\SE 201B\OpenSees\bin\OpenSees.exe"; % full path to OpenSees
executable
16 analysisType = 'Transient'; % Choose between 'Static' & 'Transient'
17 algorithm = 'Newton'; % Choose between 'Newton', 'ModifiedNewton', and 'ModifiedNewton -initial'
18 %% PLOT
19 switch analysisType
20 case 'Transient'
21     U_TH = load(['Results/disp_' analysisType '_' algorithm '.txt']);
22     U = U_TH(:,2);
23     t = U_TH(:,1);
24     R_TH = load(['Results/res_' analysisType '_' algorithm '.txt']);
25     R = -R_TH(:,2);
26     V_TH = load(['Results/vel_' analysisType '_' algorithm '.txt']);
27     V = V_TH(:,2);
28     A_TH = load(['Results/acc_' analysisType '_' algorithm '.txt']);
29     Ac_total = A_TH(:,2);
30
31     % Time Histories
32     figure();
33     subplot(3,1,1);
34     plot(t,U,'b','LineWidth',2);
35     indexmax = find(max(abs(U)) == abs(U));
36     Umax = U(indexmax);
37     tmax = t(indexmax);
38     hold on
39     plot(tmax,Umax,'ro','MarkerFaceColor','r')
40     text(tmax,Umax,sprintf(' u_{rel, max} = %1.3f in',abs(Umax)), 'VerticalAlignment','middle','HorizontalAlignment','left','FontSize',12,'FontWeight','Bold');
41     ylabel('u_{relative}(t)');
42     set(gca,'XTick',[]);
43     set(gca,'YLim',[-abs(Umax)-4,abs(Umax)+4])
44     set(gca,'XLim',[0 t(end)])
45     title('Time Histories');
46     set(gca,'FontSize',18,'FontWeight','Bold')
47
48     subplot(3,1,2);
49     plot(t,V,'b','LineWidth',2);
50     indexmax = find(max(abs(V)) == abs(V));
51     Vmax = V(indexmax);
52     tmax = t(indexmax);
53     hold on
54     plot(tmax,Vmax,'ro','MarkerFaceColor','r')
55     text(tmax,Vmax,sprintf(' v_{rel, max} = %1.3f in/s',abs(Vmax)), 'VerticalAlignment','middle','HorizontalAlignment','left','FontSize',12,'FontWeight','Bold');

```

```

12,'FontWeight','Bold');
56 ylabel('v_{relative}(t)');
57 set(gca,'XTick',[]);
58 set(gca,'YLim',[-abs(Vmax)-10,abs(Vmax)+10])
59 set(gca,'XLim',[0 t(end)])
60 set(gca,'FontSize',18,'FontWeight','Bold')
61
62 subplot(3,1,3);
63 plot(t,Ac_total,'b','LineWidth',2);
64 indexmax = find(max(abs(Ac_total)) == abs(Ac_total));
65 Amax = Ac_total(indexmax);
66 tmax = t(indexmax);
67 hold on
68 plot(tmax,Amax,'ro','MarkerFaceColor','r')
69 text(tmax,Amax,sprintf(' a_{abs, max} = %1.3f in/s^2',abs(Amax)), 'VerticalAlignment','middle','HorizontalAlignment','left','FontSize',
12,'FontWeight','Bold');
70 ylabel('a_{absolute}(t)');
71 xlabel('Time [sec]')
72 set(gca,'YLim',[-abs(Amax)-50,abs(Amax)+50])
73 set(gca,'XLim',[0 t(end)])
74 set(gca,'FontSize',18,'FontWeight','Bold')
75 set(gcf, 'Position', [1200,0,1300,700])
76 print("..\matlab\P2\submittal\figures\"+string(21)+" Opensees A-V-U",'-dsvg','-PMicrosoft Print to PDF','-r600','-painters');
77
78 % Resistance - Displacement Curve
79 figure(); hold on;
80 plot(U,R,'b','LineWidth',2);
81 matHyst = get_materialHysteresis(matDef, U./L, 10, localOpenSeesPath);
82 plot(matHyst(:,1).*L, matHyst(:,2).*A,'k--','LineWidth',1.5)
83 xlabel('U_{rel} [in]')
84 ylabel('R [kips]')
85 set(gca,'FontSize',18,'FontWeight','Bold'); grid on;
86 legend('Linearized PO Curve b/w equilibrium pts','True F-d curve','location','best','FontSize',14)
87 title("Opensees " + analysisType + " Analysis " + algorithm + " Method");
88 set(gcf, 'Position', [1200,0,1300,700])
89 print("..\matlab\P2\submittal\figures\"+string(21)+" Opensees " + analysisType, '-dsvg', '-PMicrosoft Print to PDF', '-r600', '-painters');
90
91 case 'Static'
92 U = load(['Results/disp_' analysisType '_' algorithm '.txt']);
93 R = -load(['Results/res_' analysisType '_' algorithm '.txt']);
94
95 % Resistance - Displacement Curve
96 figure();
97 plot([0;U],[0;R],'b','LineWidth',2);
98 hold on
99 plot([0;U],[0;R],'ro','LineWidth',2);
100 matHyst = get_materialHysteresis(matDef, U./L, 100, localOpenSeesPath);
101 plot(matHyst(:,1).*L, matHyst(:,2).*A,'k--','LineWidth',1.5)
102 xlabel('U [in]')
103 ylabel('R [kips]')
104 legend('Linearized PO Curve b/w equilibrium pts','Equilibrium Pts','True F-d curve','location','best','FontSize',14)
105 set(gca,'FontSize',18,'FontWeight','Bold'); grid on;
106 title("Opensees " + analysisType + " Analysis " + algorithm + " Method");
107 set(gcf, 'Position', [1200,0,1300,700])
108 print("..\matlab\P2\submittal\figures\"+string(21)+" Opensees " + analysisType, '-dsvg', '-PMicrosoft Print to PDF', '-r600', '-painters');
109 end

```