Problem Set 4

Muthu Ram Kumar Avichi (1010188967)

Question 2

Matlab Code:

```
% ---------------------------------------------------------------------------
% Matlab code for Problem Set 4 - Question 2
% Normalised density, Derivates of compliance
% ---------------------------------------------------------------------------
% clear data space
clear;
% close figure windows
close all;
% element geometry
x_e = [0 0.01 0.01 0]; y_e = [0 0 0.01 0.01]; % nodal locations (m)
area = (0.01*0.01); % area of element (m^2)
% initialize a few variables
Ke_f = zeros(8); % total stiffness matrix
K_star = zeros(8); % current term of stiffness matrix
f = zeros(8,1); % nodal force vector
d = zeros(8,1); % nodal displacement vector
E = 72e9; % Young's modulus (Pa)
nu = 0.3; % Poisson's ratio
% matrix of elastic constants
Dstar = (1/(1-nu^2)) * [1 nu 0; nu 1 0; 0 0 (1-nu)/2]; % (Pa)
% set up Gauss quadrature
xi = zeros(2);
eta = zeros(2);
xi(1) = - 1 / sqrt(3); xi(2) = 1 / sqrt(3); % Gauss points for x-direction
eta(1) = - 1 / sqrt(3); eta(2) = 1 / sqrt(3); % Gauss points for y-direction
w(1) = 1; w(2) = 1; % weights for Gauss quadrature
ax = 0; bx = 0.01;   % Limits for the x values
ay = 0; by = 0.01;   % Limits for the y values
J_det = ((bx - ax)/2)*((by - ay)/2); % determinant of Jacobian of
transformation to (xi, eta)
x = [gauss(ax, bx, xi(1)) gauss(ax, bx, xi(2))]; y = [gauss(ay, by, eta(1))
gauss(ay, by, eta(2))]; % physical locations of Gauss points (m)
for i = 1:2
   for j = 1:2
       % value of H at current Gauss point
       H = (1 / area) * [(y(j) - y_e(4)), 0, -(y(j) - y_e(4)), 0 (y(j) -
y_e(1)), 0 , -(y(j) - y_e(1)), 0;
       0, (x(i) - x_e(2)), 0, -(x(i) - x_e(1)), 0, (x(i) - x_e(1)), 0, -(x(i)
- x_e(2));
       (x(i) - x_e(2)), (y(j) - y_e(4)), -(x(i) - x_e(1)), -(y(j) - y_e(4)),
(x(i) - x_e(1)), (y(j) - y_e(1)),-(x(i) - x_e(2)), -(y(j) - y_e(1))];
       % contribution to stiffness matrix from current Gauss point
       K_star = E * w(i) * w(j) * J_det * H' * Dstar * H;
```

```
        % current total stiffness matrix
        Ke_f = Ke_f + K_star;
    end
end
disp("The stiffness matrix value is:"); disp(Ke_f);
% -------------------------------------------------------------------------
% Derivatives of Compliance
nrho = [0.45 0.7 0.35 0.6]; % normalised density
p = 3; % penalisation factor
d1 = 0.01*[0.01 0 0.015 0.003 0.015 0.006 0.005 0.004]';
d2 = 0.01*[0.015 0.003 0.02 0.008 0.03 0.009 0.015 0.006]';
d3 = 0.01*[0.005 0.004 0.015 0.006 0.025 0.005 0.015 0.004]';
d4 = 0.01*[0.015 0.006 0.03 0.009 0.035 0.006 0.025 0.005]';
% Derivatives of Compliance
ddc1 = -p * (nrho(1)^(p-1)) * d1' * Ke_f * d1;
ddc2 = -p * (nrho(2)^(p-1)) * d2' * Ke_f * d2;
ddc3 = -p * (nrho(3)^(p-1)) * d3' * Ke_f * d3;
ddc4 = -p * (nrho(4)^(p-1)) * d4' * Ke_f * d4;
disp("The values of derivatives are:"); disp(ddc1); disp(ddc2); disp(ddc3);
disp(ddc4);
% -------------------------------------------------------------------------
% Function definitions
% Function to compute the Gauss quadrature.
function result = gauss(a, b, c)
   % This function calculates the average of the square of a and b,
   % and the product of c and d.
   % Calculate the square of a and b, then find their average
   result = (a*(1 - c)/2) + (b*(1 + c)/2);
end
```

Output:

```
The stiffness matrix value is:
   1.0e+10 *

    3.5604    1.2857   -2.1758   -0.0989   -1.7802   -1.2857    0.3956    0.0989
    1.2857    3.5604    0.0989    0.3956   -1.2857   -1.7802   -0.0989   -2.1758
   -2.1758    0.0989    3.5604   -1.2857    0.3956   -0.0989   -1.7802    1.2857
   -0.0989    0.3956   -1.2857    3.5604    0.0989   -2.1758    1.2857   -1.7802
   -1.7802   -1.2857    0.3956    0.0989    3.5604    1.2857   -2.1758   -0.0989
   -1.2857   -1.7802   -0.0989   -2.1758    1.2857    3.5604    0.0989    0.3956
    0.3956   -0.0989   -1.7802    1.2857   -2.1758    0.0989    3.5604   -1.2857
    0.0989   -2.1758    1.2857   -1.7802   -0.0989    0.3956   -1.2857    3.5604

The values of derivatives are:
 -419.0148

  -1.8150e+03

 -417.6900

  -1.5390e+03
```

Explanation:

The code blocks given in the lecture slides are followed systematically to implement the derivative calculations. Steps for the computation,

- Gauss points are computed from Gauss quadrature.
- Compute stiffness matrix from the equation, iterating over four gauss points.
- Compute K star from the lecture equations using D star and E1.
- Write out u and v displacements at the nodes in the order mentioned.
- Compute the derivatives of compliance and output the results.