

EMA 605, Course Project Report

Program Description:

The program has been written in *python*. The class is named *project* which conduct two dimensional FE analysis of a thick-walled cylindrical pressure vessel.

The class variables are (can be identified inside the class as *self.variable_name*):

1. **E** - Young's Modulus
2. **v** – Poisson Ratio
3. **N** – Number of element sides along each of the radial and polar boaders
4. **a** – inner radius of the modeled vessel
5. **b** – outer radius of the modeled vessel
6. **P** – uniform pressure along the inner surface
7. **n_nodes** - number of nodes from FE meshing, which is $(N+1)^2$
8. **xy** - list containing x,y – coordinates of the nodes in the order of indices given in input
9. **n_el** – number of elements from FE meshing, which is $2*N^2$
10. **element**- matrix with nodes for each element.
11. **Load** – nodal load vector
12. **k_global** - global stiffness matrix. Initialized as a zero matrix with $2*n_nodes$ rows and columns.
13. **E_matrix** - Constitutive matrix is initialized by plain stress condition.
14. **displacement** -Displacement vector
15. **displ_scaler** – magnitude of the displacement, for checking displacement error
16. **xy_final** - node coordinates after displacement
17. **strain_vector** – strain vector calculated from element B matrix and displacement
18. **stress_vector** – stress vector calculate from E and strain_vector
19. **von** – von Mises stress calculated from stress vector
20. **cond** – condition number for stiffness matrix

Class member functions (explains the program steps and calculation in detail):

1. **Mesh(self)**: Generates mesh by constructing nodes and building connectivity for each CST element. Formulas are adopted from handout.
2. **LoadVector(self)**: Generates work equivalent load vector for thick walled cylindrical pressure vessel. The pressure is uniform inside:
 - a. Iterate over each element in the first row. (Element facing inside of the tube)
 - i) Pressure acts normal to the inner face.
 - ii) Its shared equally by the two nodes (node 1 and 3) of each triangle element. {found out by integrating the shape functions for two nodes}
 - iii) X and Y component of the forces is added to the right indices in the load vector column.
3. **B(self,el)**: Takes input every triangle element node and generates Strain Matrix for a Constant Strain Triangle element, assuming first node lies at origin. This is also done using explicit formula. The formula uses element nodal- coordinates.
4. **Global_Stiffness(self)**: Generates Global Stiffness Matrix for the plane structure-
 - a. Iterate over each element, say element *i*
 - i. Get **B** matrix and **J** for element *i*
 - ii. Compute **local- K** matrix using the relation $0.5 * |J| B^T E B$. This is a 6x6 matrix.
 - iii. Form a **Temp_global_K** matrix with $2 * n_nodes$ rows and columns and assign **local- K** to the right global node.
 - iv. Add **Temp_global_K** to the **k_global**
5. **Dirichlet_BC(self)**: Module to apply nodal displacement boundary condition to the system. Elements in the first column have their y direction motion constrained and nodes in the last column their x-direction motion.
 - a. Iterate over nodes in the first column and last column say *i* th.
 - b. Make *i* th row and column of the stiffness matrix zero
 - c. Make the *i* th diagonal element 1
 - d. Also make the *i* th load vector element 0.
6. **Strain_stress(self)**: Finds stress and strain in each element. Strain is found by matrix multiplication of **strain matrix B** and **displacement** vector. Stress is found by matrix multiplication of **E** and **strain**. Strain and stress from each element are assembled to have the global vector. Calculate **von-mises** stress for plotting
7. **Global_displacement(self)**: Use linear solver to solve linear system for displacement, meanwhile, the magnitude of displacement is also computed.

8. **Analytical_displacement(self)**: Generates analytical solution for displacement, only useful for error analysis.
9. **Analytical_Strain_Energy(self)**: Generates analytical solution for strain energy, only useful for error analysis.
10. **Analytical_Normal_Stress(self)**: Generates analytical solution for normal stress, only useful for error analysis.
11. **Error(self)**: Compare the absolute error and relative error based on the calculations from **Analytical_displacement(self)**, **Analytical_Strain_Energy(self)** and **Analytical_Normal_Stress(self)**, only useful if **error_analysis** is switched on.
12. **Plot_lines(self)**: This module is to plot all the results.
 - a. Plot the elements before deformation in **black**.
 - b. Plot elements after deformation in **cyan**.
 - c. Plot color map of error in normal stress.
13. The **main** function – (*python* supports in class testing)
 - a. Assign Values to the input variables
 - b. Forms an object of the class – **testcase**. All the calculations are done in the following sequence.
 - I) **self.Mesh()**
 - II) **self.LoadVector()**
 - III) **self.Global_Stiffness()**
 - IV) **self.Dirichlet_BC()**
 - V) **self.Global_displacement()**
 - VI) **self.Strain_stress()**
 - VII) **self.Error()** – if Initialized
 - VIII) **Plot_Lines()**
 - c. If error analysis is initialized:
 - i) Run program from 2x2 to 12x12 grid resolution
 - ii) Plot following graphs for different grid resolution
 - 1) Max Von-mises error
 - 2) Max displacement error and relative error
 - 3) Strain energy error and relative error
 - 4) Maximum Normal Stress error and relative error
 - 5) Average normal error
 - 6) Condition Number for stiffness matrix
 - 7) Residual

Results:

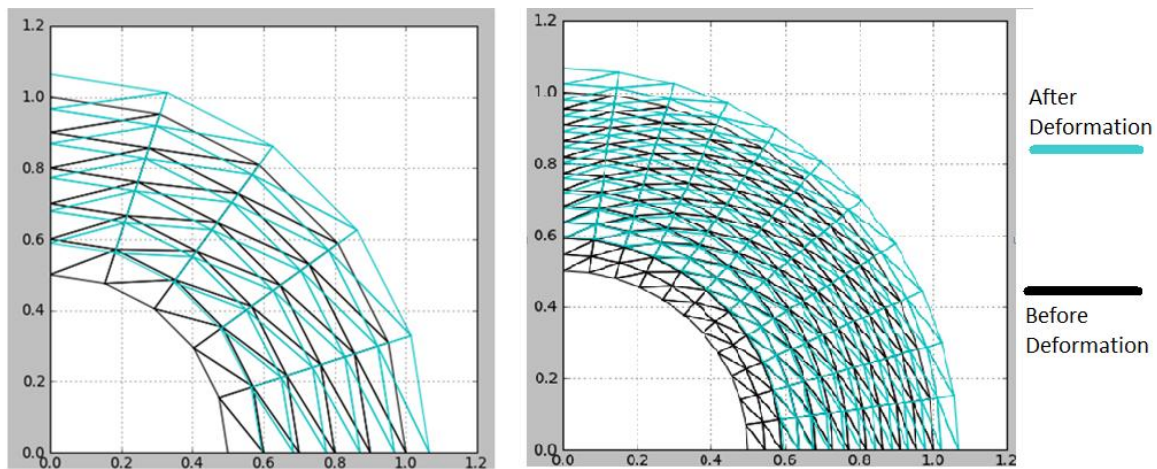
The experiment setup (with standard units):

Variable	Value
E	10
ν	0.25
a	0.5
b	1
P	1

Program output for $N=5$ and $N=11$:

1. Displacement:

As discussed in the handout, the problem was solved on a quarter cross-section. Black mesh indicates the original position and the cyan one shows the deformation of part due to pressure.



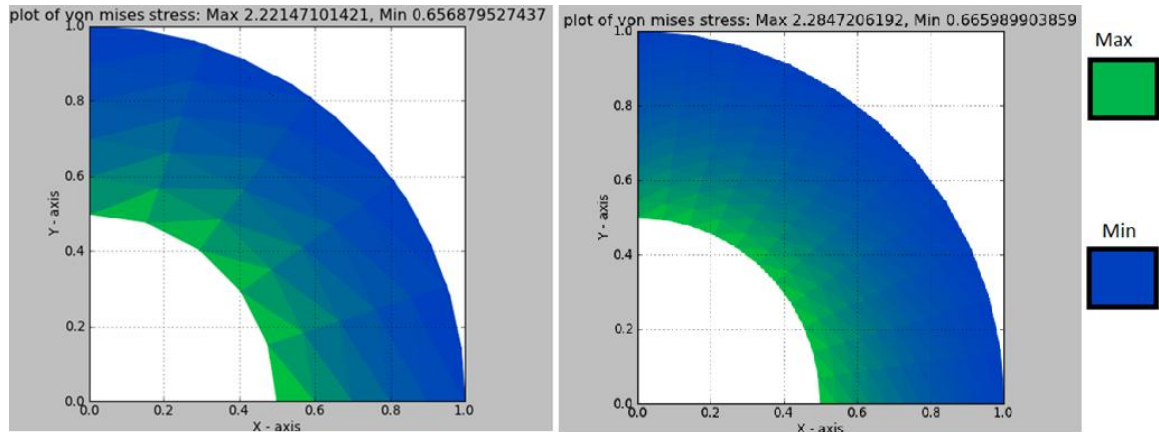
Plot showing Mesh before and after deformation (for $N = 5$ and $N = 11$)

The displacement due to normal pressure from inside is radial, as expected. There is circular symmetry in the displacement values. For finer mesh ($N=11$), the approximation on geometry is better, where the shape is more near to circular and result displacement field is smoother.

2. Von- Mises Stress:

The color map below shows the plot of von-mises stress for all the elements. The stress field is constant inside each element, but with sharp discontinuity at the element edge.

The von- mises stress is highest at the inner surface and gradually decreases to minima at the outer surface.



Color map showing Von-Mises stress distribution (for N = 5 and N =11)

Stress discontinuity is more prominent for coarse mesh. Von-mises stress distribution is smoother for N=11 compared to N=5.

Error Analysis:

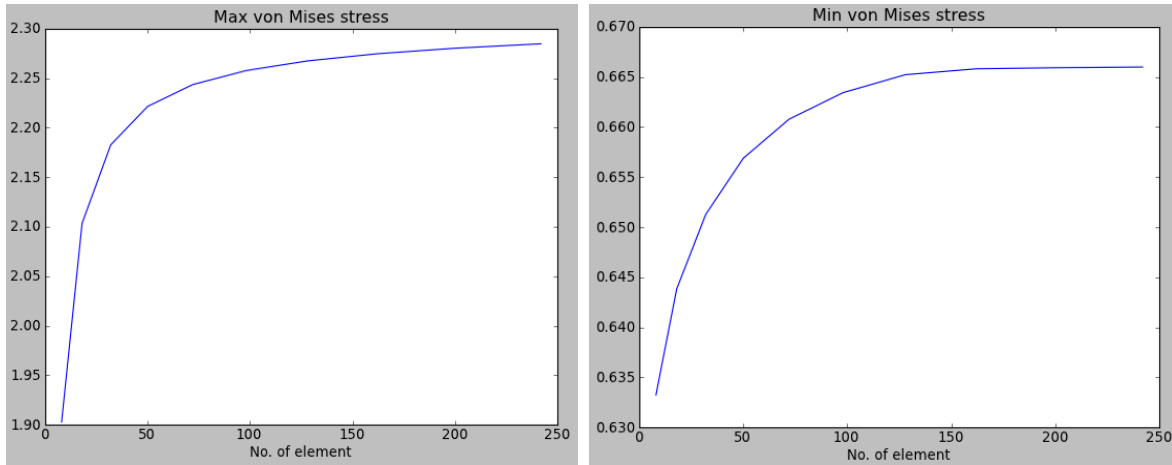
Errors involved in the simulation includes: discretization error, round off error and manipulation error. We could refine the mesh to decrease the discretization error, after using double precision float point number, there is not too much we can do to round off error and manipulation error. However, these errors need to be careful exam to make sure the correctness of the simulation result.

With the same program setup, varying N from 2 to 11, the number of elements used in the FE simulation varies from 8 to 242. With the increasing of the number of elements, the FE simulation shows the convergence properties.

1. von Mises stress

From the von Mises stress graph, the max von Mises stress converge to 2.2847206192 and min von Mises stress converge to 0.665989903859. The von Mises stress is calculated by equation:

$$\sigma_e = \frac{1}{\sqrt{2}} ((\sigma_x - \sigma_y)^2 + \sigma_x^2 + \sigma_y^2 + 6\tau^2)^{\frac{1}{2}}$$



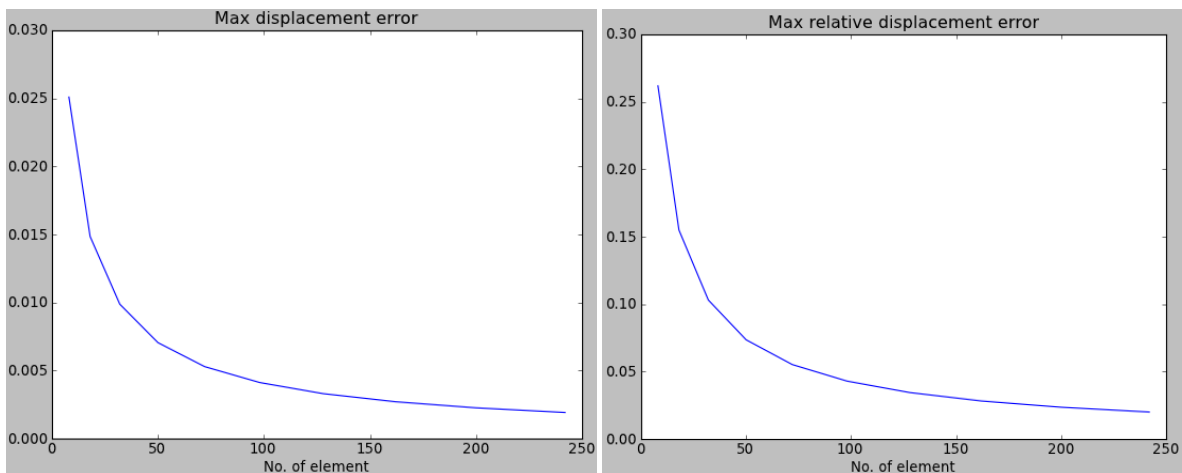
Plot for Max and Min von Mises stress

2. Displacement

Displacement error is calculated by comparing the magnitude of the displacement from FE solution to the analytical solution:

$$u_r = \frac{P_a(1-\nu^2)a^2}{E(b^2-a^2)} \left(\frac{r}{1+\nu} + \frac{b^2}{(1-\nu)r} \right)$$

From the figures below, it shows that both the max error and relative error converge to 0 as the resolution of the mesh increase, the relative error will be smaller than 5% if $N=7$, that is 98 elements of the analysis. We can also observe that the errors are positive, which is corresponding to the fact that FE solution is incline to be stiffer than the analytical solution, where the displacement is smaller compare to the analytical one.

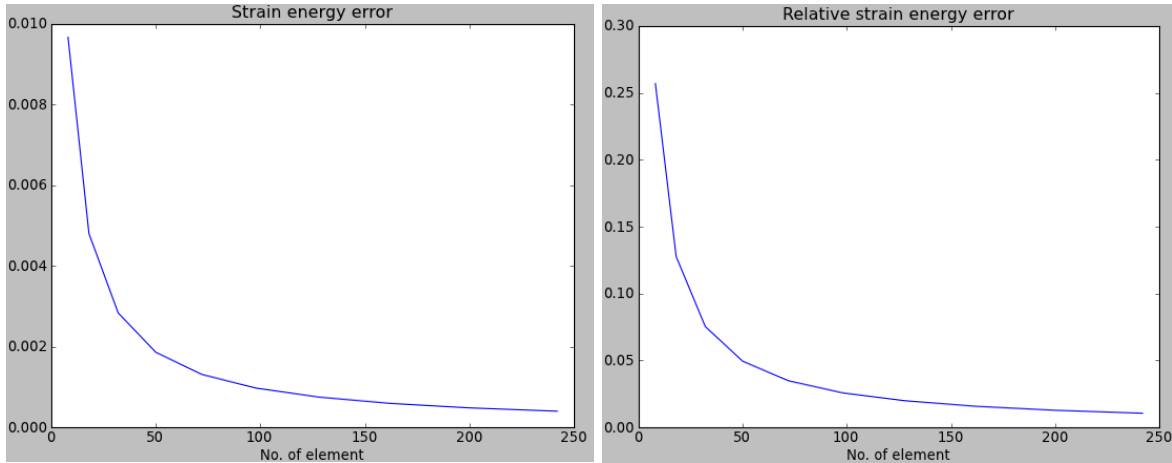


Plot for max error and max relative error of displacement

3. Strain energy

The strain energy error is calculated by comparing the FE solution to the analytical solution:

$$U = \frac{\pi P_a^2 a^2}{4(b^2 - a^2)E} \left[(1 - \nu)a^2 + (1 + \nu)b^2 \right]$$



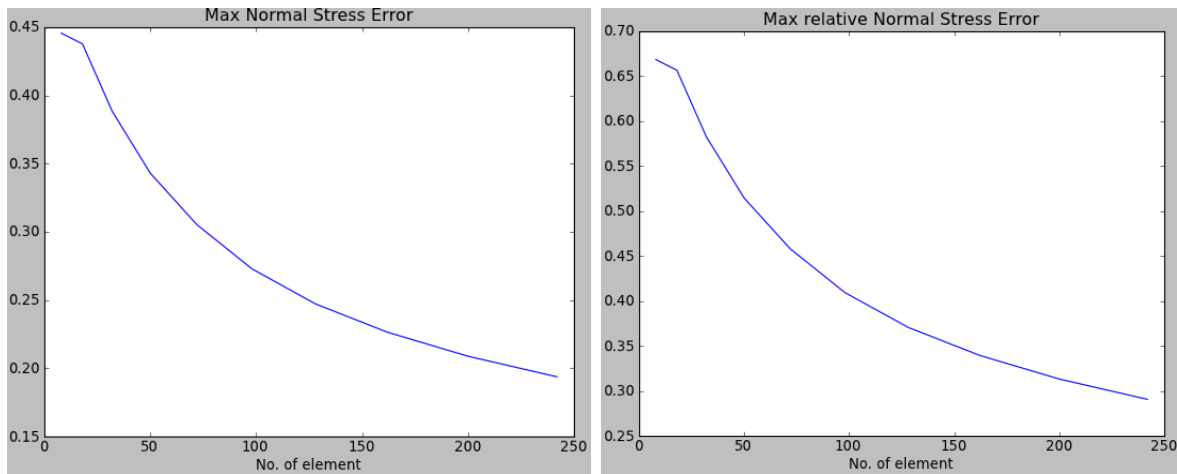
Plot for error and relative error of strain energy

The error and relative error is approaching 0 as N increase, the relative error is lower than 5% if N=5, that is 50 elements used for FE approximation. The positivity of strain energy error also shows that the strain energy from FE solution is small than the analytical solution, which shows that FE solution is stiffer than the analytical solution, which has less strain energy under same load.

4. Normal stress

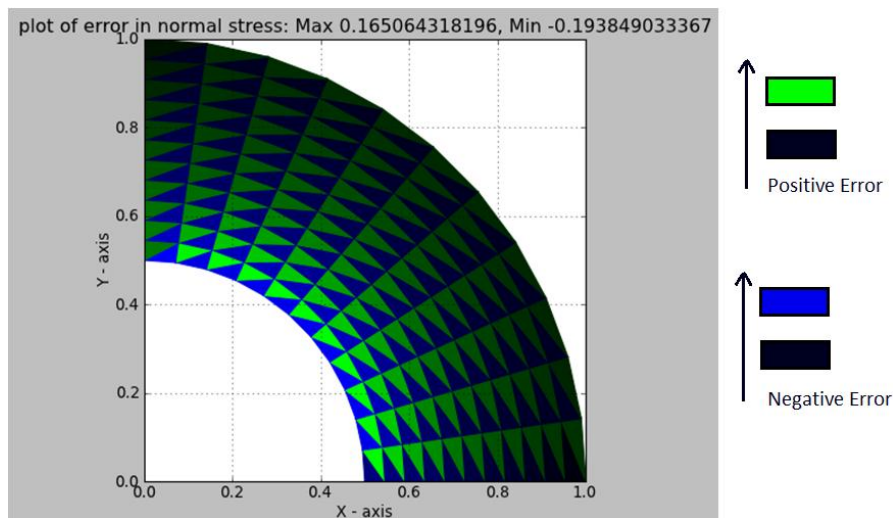
The error of Normal stress is acquired by comparing to the analytical solution for each element, where the analytical solution is calculated by equation:

$$\sigma_x + \sigma_y = \sigma_r + \sigma_\theta = \frac{2a^2}{b^2 - a^2} P_a$$



Max and max relative error of normal stress

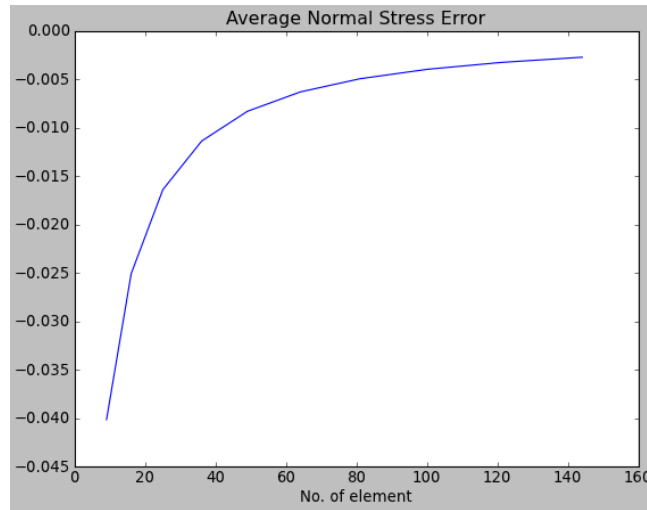
Though the max normal stress error and relative error in each element is decreasing with refinement of the mesh, the max normal stress error and relative error are still *quite high*. However, after careful inspection, the normal stress error forms a pattern, where each element with a positive normal stress error is companioned by an element with a negative counterpart. Shown as the figure below:



Color map showing deviation in Normal Stress for each element

From the plot of error in displacement and normal stress, we can observe that, displacement error reduces at a rate faster than normal stress. This is due to fact that stress will be a one-order lower polynomial than displacement.

With positive and negative error cancel out, the normal stress error over the whole part is converging to zero from a negative value.

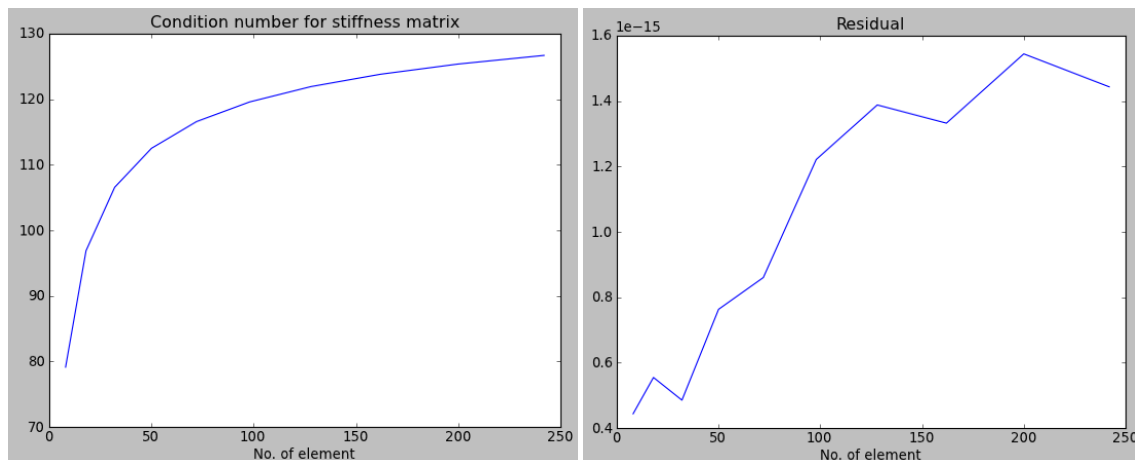


Plot for Average Normal Stress

This behavior is corresponding to the fact that the FE solution is stiffer than the analytical solution only on the overall case, where the average normal stresses are smaller than analytical solutions. But this is not necessarily true at every point inside FE solution field, where positive and negative error pattern corresponding to larger and smaller normal stress compare to the analytical solution, respectively).

5. Numerical error

We also print out the condition number of the stiffness matrix and residual from the solution of the linear system:



Plot of Condition Number and Residual

The condition number of stiffness matrix goes to a reasonable number as the dimension of the matrix increase. To make sure there is not huge numerical error, the residual has also been calculated.

The residual of the solution of the linear system remain to be very small, this gives us the confidence to have the convergence of our FE approximation.

Conclusion:

1. All key parameters checked here converge monotonically to the analytical value. So finite element does work, approximately with exact value in limiting case.
2. Except maximum error in stress values, all other parameters converge quite fast. Error reduces at a very fast rate in the beginning but there is an asymptotic behavior as FEM solution becomes closer to analytical solution.
3. We can conclude that after one point, further increase in mesh size will result in slightly improved results but on the cost of high computation time. So for achieving a good result with optimal expenditure of resources, the change in error with finer meshing can be a good parameter to check for stopping.