Introduction to Materials Modelling: FEM

Sheet 1 — MT20

FEM Matlab Class

First ensure you have Matlab installed, then download and open `FEM-studentCopy.m` from Canvas `https://canvas.ox.ac.uk/courses/74566/pages/day5-fe-modelling`). This is a complete 2D elastic finite element code. During this class you will familiarise yourself with some aspects of the code and make some modifications to improve your understanding of how it works. First of all try running it! It solves the plane strain equilibrium problem of an end loaded cantilever beam. Before you make any changes, save a copy of the code as you might want to compare to the original version! (*hint*: You can run the entire code by pressing F5, an individual section by pressing command+F5 (on Mac) or you can highlight a few lines and evaluate the selection (shift+F7 on Mac). Try adding some break points by clicking in the margin of the code and stepping through line by line if you want to see in more detail what it's doing).

1. (a) A nodal force of -0.1 is applied at the top right corner of the beam at node `mx+1` located at (dx, dy). This is a traction (*Neumann*) boundary condition. What are the nodal displacements in the x and y direction of this node ?

   (b) Modify the boundary conditions so that a vertical displacement (*Dirichlet*) boundary condition is applied instead. First identify the line of code where the force is applied, comment it out and rerun the code to check that the undeformed state is the solution.

   (c) Apply the vertical displacement you found in part (a) to node `mx+1` and update the vector containing the fixed degrees of freedom to include not only the left edge, which is fixed in x and y, but also the y displacement of node `mx+1`.

   (d) The applied displacement vector should now have only one non-zero component, this will result in several non-zero forces which can be computed using: `f = f - K*u`; Now solve the system of equations to find the unknown nodal displacements.

   (e) Check your solution by finding the reaction forces, eg `rf = K*u;`. What is the vertical reaction force at the top right corner node? (*hint* you should find the same applied force as in part (a).) *Optional* Confirm that the deformed mesh and stress plots also agree.

   (f) *Optional* Confirm that the reaction forces in the x and y directions both sum to zero.

2. In this question you will modify the code to simulate uniaxial tension
   (a) First define the set of nodes on the right edge
   (b) Apply a horizontal displacement at every node in `Sright` to generate a 1% strain in ε_{xx}.
   (c) Update the fixed degrees of freedom to include the applied displacements on the right edge
   (d) Plot the deformed mesh with a scale factor of 10 and 100, does it look correct? Plot the stres components by changing index 1 in `Z = mean(stress(1,:))*ones(2);` to be 1,2 or 3 to plot σ_{xx}, σ_{yy} or σ_{xy} respectively. You'll see that the stress state is not uniaxial due to the yy and xy stresses near the fixed end. This is beacuase the boundary conditions are not quite right there. How can you fix them?
   (e) Instead of fixing every node on the left edge in the y direction, fix only 1 node in y (you will still need to fix them all in x). Confirm that the stress and strain you obtain agree with Hooke's law eg $\sigma_{xx} = (\lambda + 2\mu)\varepsilon_{xx} + \lambda\varepsilon_{yy} =$ `D(1,1)*strain(1) + D(1,2)*strain(2)` should be equal to `stress(1)` and $\sigma_{yy} = \sigma_{xy} = 0$ eg `stress(2)` $=$ `stress(3)` $= 0$. You should observe that there is no shear strain but there is a negative $\varepsilon_{yy} = -\nu/(1-\nu)\varepsilon_{xx}$ strain due to Poison contraction eg `strain(2,:)` should equal `-nu/(1-nu) * strain(1,:)` and ε_{xx} = strain(1,:) should be 0.01. You can check this by plotting the fields and by looking at values for any element.

3. *Optional*
   (a) Write down the Jacobian matrix $\boldsymbol{J}$ for an element. Does it vary either within an element or between different elements?
   (b) How does the determinant of $\boldsymbol{J}$ for an element relate to the area of the element?
   (c) Change the number of elements so that $\boldsymbol{J}$ is the identity matrix and check that the shape function derivatives are the same for both the isoparametric element and the real element eg $\partial N^a/\partial s_i = \partial N^a/\partial x_i$.

Bigger Hints!

1c) `u((mx+1)*2) = -0.1780;` and `fixedDofs = [2*Sleft-1;2*Sleft; 2*(mx+1)];`
2a) `Sright = find(x(:,1) == dx);`
2b) `u(2*Sright-1) = 0.01*dx;`
2c) `fixedDofs = [2*Sleft-1;2*Sleft; 2*Sright-1];`