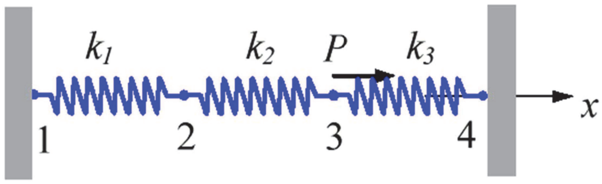
December 13, 2020

Linear FEM Assignment (1 EC)

Exercise 1 (1 pts.)

For the spring system shown below with $k_2 = 2k_1 = 2k_3 = 2k$ and given spring properties of $E_k = 1GPa$ (Young's modulus), $L_k = 1m$ and $A_k = 1cm^2$ (cross section), note that k = EA/L , and boundary conditions of $P = 1N$ and $u_1 = u_4 = 0$, find:



1. The global stiffness matrix
2. Displacements of nodes 2 and 3
3. The reaction forces at node 1 and 4
4. The force in the spring 2

Exercise 2 (2 pts.)

Consider a linear elastic bar of length $L = 1m$ with a cross section $A = 1cm^2$ and a given Young's modulus $E = 1GPa$. The bar is rigidly supported in the left end and in the right acts a concentrated force $F = 1N$.

1. Solve this problem analytically.
2. Solve this problem numerically by making use of the finite element method. Consider that the bar is made out of three elements. And construct the global stiffness matrix.
3. Create a MatLab script for the given problem. Note that your script must be independent of number of elements.
4. Compare your analytical solution with the FEM result.
5. Discuss the effect of number of elements on the accuracy of FEM result.

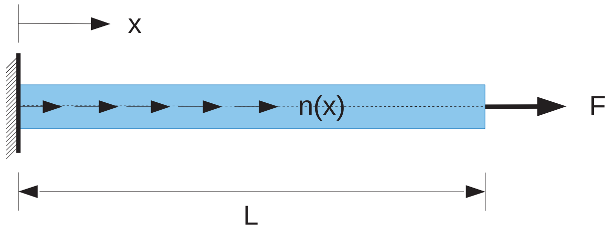


Figure 1: Note that line load is equal to zero ($n(x) = 0$).

Exercise 3 (5+2 (optional) pts.)

Consider a 2D truss as shown in Fig.2. The horizontal and vertical truss elements have an initial length $L_0 = 1m$ and the diagonal elements have an initial length of $\sqrt{2}L_0$. All members have $E = 1GPa$, $A = 1cm^2$ and ρ = $1000kg/m^3$. Constrain all the bottom nodes in vertical direction only (free along X) and constrain all the left side nodes in horizontal direction (free along Y),

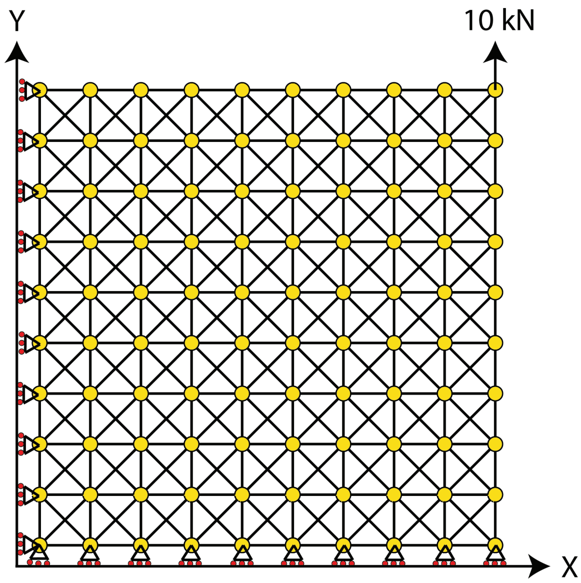


Figure 2: 2D truss model system with a concenterated force

1. Apply a load 10 kN to the top right node +Y direction and use a finite element method to compute the displacement of this node. Compute the lowest 6 eigenpairs

(eigenvalues and eigenvectors (also called mode shapes) of the stiffness matrix, check Matlab 'eig' function for this). Plot the mode shapes and discuss your plots.

2. Considering the system as a solid structure, compute the Young's Modulus (E_{struc}), Shear Modulus (G_{struc}) and Poisson ratio (ν_{struc}) of the bulk structure using the following formula's ,

   (a) Apply a 1 kN vertical load (+Y direction) on all the top nodes of the structure (Fig.3) measure the elongation (ΔL).

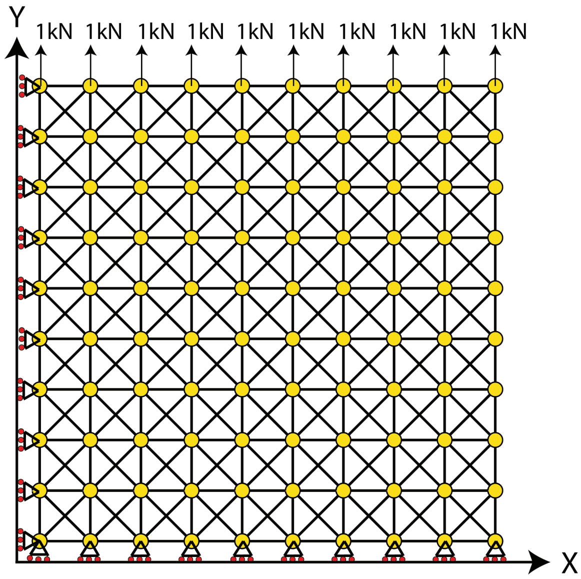


Figure 3: 2D truss model system to compute Young's modulus

$$E_{struc} = \frac{\text{Longitudinal Stress}}{\text{Longitudinal Strain}} = \frac{\text{TotalLoad}/(dW)}{\Delta L/L}$$

Where, $TotalLoad$ is sum of all the loads on the top nodes, d is the cross-sectional diameter of the truss members (Compute using $A = \pi d^2/4$) and ΔL is the average displacement of the top row.

   (b) Also measure the average change in width (ΔW) of the top row and compute Poisson ratio as,

$$\nu = \frac{\text{Change in transverse length}}{\text{Change in longitudinal length}} = \frac{\Delta W}{\Delta L}$$

Note that positive Poisson's ratio corresponds to contraction
in a transversal direction when the specimen is stretched longitudinally.

(c) To measure shear modulus first remove all the constraints on side nodes and fix *all* the bottom nodes in Y-direction, also fix just *one* corner(bottom left) node in both X and Y directions (Fig.4). Now, apply a shear load (+X direction) of 1 kN on all the top nodes and compute G_{struc} as follows,

$$G_{struc} = \frac{\text{Shear Stress}}{\text{Shear Strain}} = \frac{\text{Total Load}/(dW)}{\Delta W/L}$$

here, ΔW is the average displacement of the top row nodes.

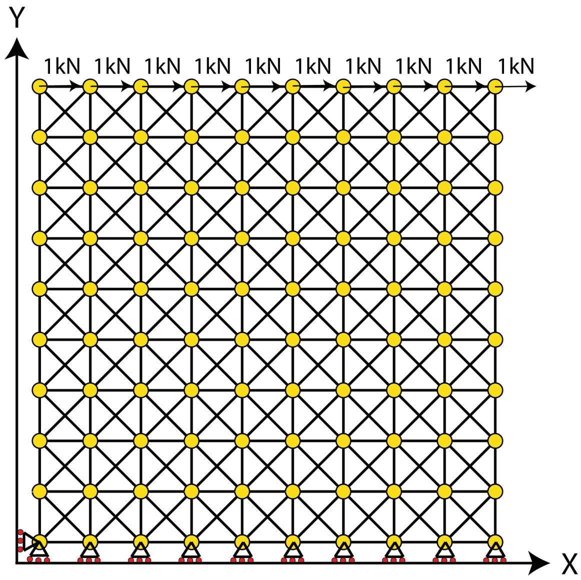


Figure 4: 2D truss model system to compute shear modulus

3. Optional: **For extra points only (2 pts.)**
Create an anisotropic structure by removing all horizontal elements.

(a) Apply a 1 kN vertical load (+Y direction) on all the top nodes of the structure (Fig.5(a)). Measure the vertical Young's modulus E_v.

(b) Apply a 1 kN horizontal load (+X direction) on all the right nodes of the structure (Fig.5(b)). Measure the horizontal Young's modulus E_h.

(c) Apply a 1 kN horizontal load (+X direction) on all the top nodes of the structure (Fig.5(c)). Measure the horizontal Shear modulus G_h.

(d) Apply a 1 kN vertical load (+Y direction) on all the right nodes of the structure (Fig.5(d)). Measure the vertical Shear modulus G_v.

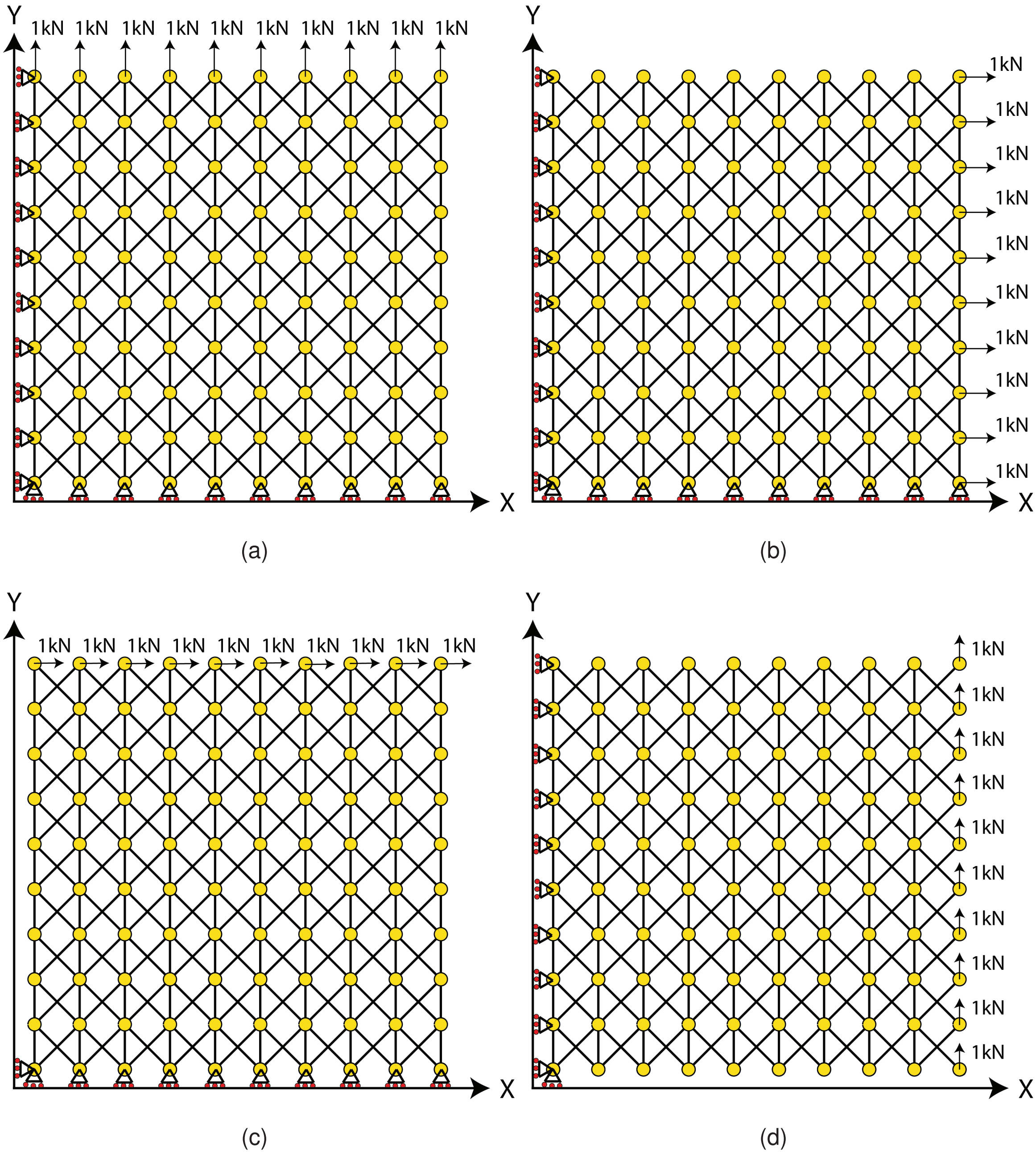


Figure 5: Anisotropic 2D truss structure under different loading conditions in order to compute elastic moduli: (a) E_v, (b) E_h, (c) G_h and (d) G_v.

Note that this exercise will be compared with its corresponding MD assignment which is given by Prof.Luding.

APiE Excercise – FEM for 2D Trusses (Hint)

This is a quick overview of the steps for solving the Exercise FEM for 2D Trusses. As usual, there are different ways to implement the code - some more and some less efficient. For extremely large networks (1000 x 1000 or more) efficient solution techniques also matter and involve using sparse matrices, preconditioners and iterative solvers such as conjugate gradient etc. (see script/lecture for more details).

General Instructions:

- A mass-spring network is given, which is similar to a network of bars in compression/tension only i.e. a truss. Each node in the truss can move in both x,y directions (except nodes that are fixed at the bottom).

- As shown in the script (and lecture) you can write the FEM stiffness matrix for each bar/spring in the network. Given case is simple since there are only two orientations (0, 90) in the network. We STRONGLY encourage you to write a -function- which returns an element stiffness matrix (and force vector) and not compute matrix for each member individually. Thus, in Matlab you can have a function as follows,

  function [Ke, Fe]= esf_2d(E,theta)

  —compute element stiffness matrix—
  —compute element force vector—

  return Ke, Fe;

  end You may want to code it for arbitrary orientation (θ) and stiffness E or keep matters simple for now.

- Call the above function from the main program and assemble the element matrices in a global (big) matrix and apply boundary conditions and solve.

Algorithm

Step 1: Create a global numbering for each node in the network, store node locations (x, y) in a matrix.
Step 2: Create a connectivity array for all the elements in the network for e.g,
Elem(1,1) = 1;
Elem(1,2) = 7; %Element no. 1 is connected to global node number 1 and 7%
etc. ...
Step 3: Initialize element stiffness matrix([Ke]) and element force vector ($\{Fe\}$) to zero.

Step 4: Initialize a global stiffness matrix ([K]) and global force vector ({F}). Determine the size based on total number of nodes $\times$ 2 .
Step 5: Loop over all the elements and compute element stiffness matrix ([Ke, Fe]=esf_2d(200,90) etc.) and add [Ke] and {Fe} to [K] and {F} based on conn ectivity array.
Step 6: Apply boundary conditions i.e. remove rows and columns corresponding to the nodes which are fixed, this will reduce the size of your global stiffness matrix.
Step 7: Solve, {U}= [K]\{F} .

If you have any question, please contact

i.ostanin@utwente.nl