Universitat Politècnica de Catalunya
Numerical Methods in Engineering
Computational Solid Mechanics and Dynamics

Direct Stiffness method

Assignment 1

Eduard Gómez
February 17, 2020

Contents

1 Introduction

1.1 Problem statement

Consider the truss problem defined in the figure 1. All geometric and material properties: L, α, E and A, as well as the applied forces P and H are to be kept as variables. This truss has 8 degrees of freedom, with six of them removable by the fixed displacement conditions at nodes 2, 3 and 4. This structure is statically indeterminate as long as $\alpha \neq 0$.

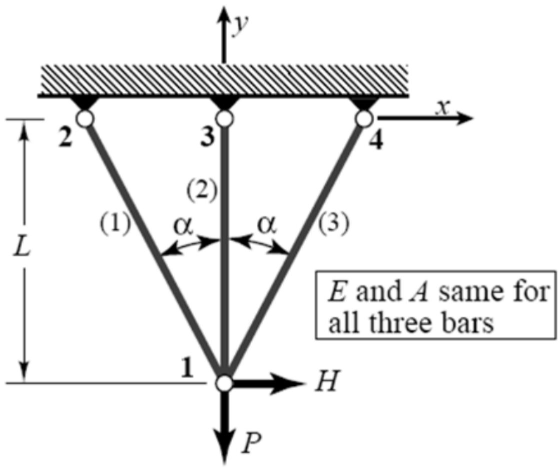

Figure 1: Truss structure. Geometry and mechanical features.

2 Resolution

2.1 Question 1

Statement

Show that the master stiffness equations are

$$\frac{EA}{L}\begin{bmatrix} 2cs^2 & 0 & -cs^2 & c^2s & 0 & 0 & -cs^2 & -c^2s \\ & 1+2c^3 & c^2s & -c^3 & 0 & -1 & -c^2s & -c^3 \\ & & cs^2 & -c^2s & 0 & 0 & 0 & 0 \\ & & & c^3 & 0 & 0 & 0 & 0 \\ & & & & 0 & 0 & 0 & 0 \\ & & & & & 1 & 0 & 0 \\ & & & & & & cs^2 & c^2s \\ \text{symm} & & & & & & & c^3 \end{bmatrix}\begin{bmatrix} u_{x1} \\ u_{y1} \\ u_{x2} \\ u_{y2} \\ u_{x3} \\ u_{y3} \\ u_{x4} \\ u_{y4} \end{bmatrix} = \begin{bmatrix} H \\ -P \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{1}$$

in which c = cosα and s = sinα. Explain from physics why the 5$^{\text{th}}$ row and column contain only zeros.

Solution

Let's start by computing the local stiffness matrices. The general formula for an arbitrary bar is:

$$K^e = \frac{EA}{L}\begin{bmatrix} c^2 & sc & -c^2 & -sc \\ & s^2 & -sc & -s^2 \\ & & c^2 & sc \\ \text{symm} & & & s^2 \end{bmatrix} \tag{2}$$

with all cosines and sines of θ, the counterclockwise angle of the bar with respect to the horizontal. Consider that the angles are for each element, in degrees:

$$\boldsymbol{\theta} = \begin{bmatrix} 90+\alpha, & 90, & 90-\alpha \end{bmatrix}$$

These imply the following trigonometric identities:

$$\cos\boldsymbol{\theta} = \begin{bmatrix} -\sin\alpha & 0 & \sin\alpha \end{bmatrix}$$
$$\sin\boldsymbol{\theta} = \begin{bmatrix} \cos\alpha & 1 & \cos\alpha \end{bmatrix}$$

Therefore the three local stiffness matrices are, in terms of α:

$$K^1 = \frac{EA}{L/c}\begin{bmatrix} s^2 & -sc & -s^2 & sc \\ & c^2 & sc & -c^2 \\ & & s^2 & -sc \\ \text{symm} & & & c^2 \end{bmatrix} \tag{3}$$

$$K^2 = \frac{EA}{L}\begin{bmatrix} 0 & 0 & 0 & 0 \\ & 1 & 0 & -1 \\ & & 0 & 0 \\ \text{symm} & & & 1 \end{bmatrix} \tag{4}$$

$$K^3 = \frac{EA}{L/c}\begin{bmatrix} s^2 & sc & -s^2 & -sc \\ & c^2 & -sc & -c^2 \\ & & s^2 & sc \\ \text{symm} & & & c^2 \end{bmatrix} \tag{5}$$

The assembled matrix is:

$$K = \frac{EA}{L}\begin{bmatrix} K^1_{11}+K^2_{11}+K^3_{11} & K^1_{12}+K^2_{12}+K^3_{12} & K^1_{13} & K^1_{14} & K^2_{13} & K^2_{14} & K^3_{13} & K^3_{14} \\ & K^1_{22}+K^2_{22}+K^3_{22} & K^1_{23} & K^1_{24} & K^2_{23} & K^2_{24} & K^3_{23} & K^3_{24} \\ & & K^1_{33} & K^1_{34} & 0 & 0 & 0 & 0 \\ & & & K^1_{44}0 & 0 & 0 & 0 & 0 \\ & & & & K^2_{33} & K^2_{34} & 0 & 0 \\ & & & & & K^2_{44} & 0 & 0 \\ & & & & & & K^3_{33} & K^3_{34} \\ \text{symm} & & & & & & & K^3_{44} \end{bmatrix} \tag{6}$$

Substituting the values it becomes:

$$K = \begin{bmatrix} cs+0+cs & -sc^2+0+sc^2 & -cs^2 & sc^2 & 0 & 1 & -cs^2 & -c^2s \\ & c^3+1+c^3 & sc^2 & -c^3 & 0 & -1 & -c^2s & -c^3 \\ & & cs^2 & -sc^2 & 0 & 0 & 0 & 0 \\ & & & c^3 & 0 & 0 & 0 & 0 \\ & & & & 0 & 0 & 0 & 0 \\ & & & & & 1 & 0 & 0 \\ & & & & & & cs^2 & c^2s \\ \text{symm} & & & & & & & c^3 \end{bmatrix} \tag{7}$$

Operating:

$$K = \frac{EA}{L}\begin{bmatrix} 2cs & 0 & -cs^2 & sc^2 & 0 & 1 & -cs^2 & -c^2s \\ & 1+2c^3 & sc^2 & -c^3 & 0 & -1 & -c^2s & -c^3 \\ & & cs^2 & -sc^2 & 0 & 0 & 0 & 0 \\ & & & c^3 & 0 & 0 & 0 & 0 \\ & & & & 0 & 0 & 0 & 0 \\ & & & & & 1 & 0 & 0 \\ & & & & & & cs^2 & c^2s \\ \text{symm} & & & & & & & c^3 \end{bmatrix} \tag{8}$$

The force vector is easy to obtain since there are only two external loads, both on node 1:

$$F = \begin{bmatrix} H & -P & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{9}$$

Discussion

It is evident that the stiffness matrices in equations 1 and 8 are the same. The vector in equation 9 is also identical to the one in expression 1.

Let's now think about column and row 5. First we must understand the meaning of rows and columns. Row 5 excluding column five (i.e K_{5i} for $i = [1,4] \cup [6,8]$) describes how the 5^{th} degree of freedom depends on all other degrees of freedom:

$$u_5 = \left(F_5 - \sum_{\substack{i=1 \\ i\neq 5}}^{8} K_{5i} u_i \right) / K_{55} \tag{10}$$

Column 5 excluding row 5 (i.e K_{i5} for $i = [1,4] \cup [6,8]$) states the reciprocal; how all other degrees of freedom depend on DoF number 5:

$$u_i = \left(F_i - K_{i5}u_5 - \sum_{\substack{j=1 \\ j\neq i,5}}^{8} K_{ij} u_j \right) / K_{ii} \qquad i = [1,4] \cup [6,8] \tag{11}$$

Finally, the intersection between the two (K_{55}) is the stiffness against internal and external forces acting on point 5. In equation 10, the summation term equals the internal forces and F_5 is the external load.

Let's go back to physics now. Node 3 is only connected to the rest of the geometry via the second element, hence any effect of displacing point 3 must propagate through this element. Since element 2 is perfectly vertical, it offers no impediment to horizontal motion of node 3, thus it transfers no internal horizontal forces neither from the rest of the domain to point 3 nor vice versa. By the same rule, it cannot dump any external horizontal load on 3 onto the rest of the structure, hence being unable to resist it. Then the conclusion is:

1. Since other displacements cannot affect u_{x3}, all row 5 (except at the 5^{th} column) is zero.
2. Since displacement u_{x3} cannot affect the other displacements, all column 5 (except at the 5^{th} row) is zero.
3. Since point 3 cannot resist any horizontal external load, and internal horizontal forces are zero, u_{x3} must be unbounded for any $F_5 \neq 0$. Therefore K_{55} must be zero.

2.2 Question 2

Statement

Apply the boundary conditions and show the 2-equation modified stiffness system.

Solution

Since all prescribed displacements are zero, reducing the system entails removing all rows and columns except 1 and 2. The system becomes:

$$\frac{EA}{L}\begin{bmatrix} 2cs^2 & 0 \\ 0 & 1+2c^3 \end{bmatrix}\begin{bmatrix} u_{x1} \\ u_{y1} \end{bmatrix} = \begin{bmatrix} H \\ -P \end{bmatrix} \tag{12}$$

2.3 Question 3

Statement

Solve for the displacements u_{x1} and u_{y1}. Check that the solution makes physical sense for the limit cases $\alpha \to 0$ and $\alpha \to 90$. Why does u_{x1} “blow up” if $H \neq 0$ and $\alpha \to 0$?

Solution

Solving the equation system in expression 12 is trivial:

$$u_{x1} = \frac{L}{EA}\frac{H}{2cs^2} \qquad u_{y1} = -\frac{L}{EA}\frac{P}{1+2c^3} \tag{13}$$

Discussion

The limit case $\alpha \to 0$ is equivalent to having three overlapping vertical bars from node one to superposed notes 2,3 and 4. Mathematically the result is:

$$\lim_{\alpha\to 0^+} u_{x1} = +\infty \qquad \lim_{\alpha\to 0^+} u_{y1} = -\frac{LP}{3EA} \tag{14}$$

The physical explanation of this result is that the system is equivalent to a pendulum with a bar thrice as thick as that of the non-limit case. This pendulum resists vertical motion but cannot resist horizontal displacement of node 1 since all bars are strictly vertical. Furthermore, u_{x1} can only remain bounded for $H = 0$, in which case it becomes undetermined yet finite; that is, any value is acceptable.

On the other hand, the limit case $\alpha \to \pi/2$ returns the following result:

$$\lim_{\alpha\to \pi/2^-} u_{x1} = +\infty \qquad \lim_{\alpha\to \pi/2^-} u_{y1} = -\frac{LP}{EA} \tag{15}$$

The physical explanation of this result is that the system is equivalent to a single vertical bar. This is because the other two bars become infinitely long both to the right and to the left. Any displacement u_{x1} becomes negligible in front of the infinite length of these bars. That is:

$$\alpha \to \pi/2 \quad \Rightarrow \quad \varepsilon_i = \frac{u_{x1}}{L/\cos\alpha} \to 0 \quad \Rightarrow \quad \sigma_i = E\varepsilon_i \to 0 \qquad \text{for } i = 1,3 \tag{16}$$

Since there is no strain, there is no stress, and therefore no push-back against horizontal loads on point 1. The system is therefore equivalent to a pendulum were the bar is the same as the vertical bar in the non-limit case. The consequences are the same as in the previous limit case albeit with different values.

2.4 Question 4

Statement

Recover the axial forces in the three members. Why do F[1] and F[3] "blow up" if $H \neq 0$ and $\alpha \to 0$?

Solution

Starting from the constitutive equation for 1D stress we can recover the force-displacement relationship:

$$\sigma = E\varepsilon \tag{17}$$
$$F/A = Eu_A/l \tag{18}$$
$$F = \frac{EA}{l}u_A \tag{19}$$

Where u_A is the displacement projected onto the axial direction and l is the length of the element. The results are:

$$F^1 = \frac{EA}{L/c}(u_{1x}s + u_{y1}c) = \frac{H}{2s} - \frac{c^2P}{1+2c^3} \tag{20}$$
$$F^2 = \frac{EA}{L}u_{y1} = -\frac{P}{1+2c^3} \tag{21}$$
$$F^3 = \frac{EA}{L/c}(-u_{1x}s + u_{y1}c) = -\frac{H}{2s} - \frac{c^2P}{1+2c^3} \tag{22}$$

Discussion

F^1 and F^3 blow up for $\alpha \to 0$ because the horizontal projection of the axial force diminishes as α becomes smaller. Since H remains constant, the axial force must increase to compensate. At the limit, the projection is zero and therefore the axial force is unbounded. Of course for $H = 0$ the horizontal projections of F^1 and F^3 are zero regardless of the angle.

2.5 Question 5

Statement

Dr. Who proposes "improving" the result for the example truss of the 1st lesson by putting one extra node, 4 at the midpoint of member (3) 1-3, so that it is subdivided in two different members: (3) 1-4 and (4) 3-4. His "reasoning" is that more is better. Try Dr. Who's suggestion by hand computations and verify that the solution "blows up" because the modified master stiffness is singular. Explain physically.

Solution

Let's first obtain the elemental stiffness matrices:

$$K^1 = 10 \begin{bmatrix} 1 & 0 & -1 & 0 \\ & 0 & 0 & 0 \\ & & 1 & 0 \\ \text{symm} & & & 0 \end{bmatrix} K^2 = 10 \begin{bmatrix} 0 & 0 & 0 & 0 \\ & 1 & 0 & -1 \\ & & 0 & 0 \\ \text{symm} & & & 1 \end{bmatrix}$$

$$K^3 = K^4 = 7.0711 \begin{bmatrix} 1 & 1 & -1 & -1 \\ & 1 & -1 & -1 \\ & & 1 & 1 \\ \text{symm} & & & 1 \end{bmatrix}$$

Assembling the global stiffness matrix yields:

$$\begin{bmatrix} 17.071 & 7.0711 & -10 & 0 & 0 & 0 & -7.0711 & -7.0711 \\ 7.0711 & 7.0711 & 0 & 0 & 0 & 0 & -7.0711 & -7.0711 \\ -10 & 0 & 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10 & 0 & -10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 7.0711 & 7.0711 & -7.0711 & -7.0711 \\ 0 & 0 & 0 & -10 & 7.0711 & 17.071 & -7.0711 & -7.0711 \\ -7.0711 & -7.0711 & 0 & 0 & -7.0711 & -7.0711 & 14.142 & 14.142 \\ -7.0711 & -7.0711 & 0 & 0 & -7.0711 & -7.0711 & 14.142 & 14.142 \end{bmatrix} \tag{23}$$

After reducing it, it becomes:

$$\begin{bmatrix} 10 & 0 & 0 & 0 & 0 \\ 0 & 7.0711 & 7.0711 & -7.0711 & -7.0711 \\ 0 & 7.0711 & 17.071 & -7.0711 & -7.0711 \\ 0 & -7.0711 & -7.0711 & 14.142 & 14.142 \\ 0 & -7.0711 & -7.0711 & 14.142 & 14.142 \end{bmatrix} \begin{bmatrix} u_2 \\ u_3 \\ v_3 \\ u_4 \\ v_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \\ 1 \\ 0 \\ 0 \end{bmatrix} \tag{24}$$

This matrix is clearly singular since the last two columns are identical.

Discussion

The results for point 2 are the same as before, so there was no gain in accuracy. The rest of displacements cannot be calculated.
Moving on to a physical interpretation, the reason why u_{x4} blows up is because it is only connected to two bars, 4 and 5, and these two are co-linear, so it cannot resist horizontal transverse loads. Appendix A.1 prooves that a node connected only to co-linear elements will always cause a singular stiffness matrix.

One cannot gain further accuracy by adding nodes since this problem is linear. Two-noded elements are sufficient to fully capture the exact solution. Furthermore, since nodes act like hinges, adding more of them not only wastes computational power, it also changes the problem and fails to accurately portray the geometry. It is therefore counter-producing both from modelling and numerical points of view.

3 Extra work

3.1 Programmed solution

In order to solve problems for an arbitrary geometry, I wrote some code in Matlab using symbolic variables. The results are correct however Matlab does a poor job at simplifying trigonometric expressions. The code can be found in the appendix. The output looks like the following:

```
>> CSMD_HW1

u =

[ (H*L)/(2*A*E*sin(a)^2*(1 - sin(a)^2)^(1/2)), -(L*P)/(A*E*(2*(1 - sin(a)^2)^(3/2) + 1))]
[                                            0,                                         0]
[                                            0,                                         0]
[                                            0,                                         0]

```

I also created a non-symbolic variable version. Using it on Question 5 gets:

```
>> CSMD_HW1_not_sym
Warning: Matrix is singular to working precision.
> In CSMD_HW1_not_sym (line 97)


u =

     0     0
   NaN     0
   NaN   NaN
   NaN   NaN
```

A Attachments

A.1 Proof that co-linear elements yield singular stiffness matrices

My aim is to prove that the existence of a node connected only to two co-linear elements, without restricted displacements, will cause an singular stiffness matrix. Let's start by considering two elements in between co-linear nodes A, O, and B. We'll call node O the critical node.

$$\boldsymbol{A} = -l_1 \begin{bmatrix} \cos\alpha \\ \sin\alpha \end{bmatrix} \qquad \boldsymbol{O} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad \boldsymbol{B} = l_2 \begin{bmatrix} \cos\alpha \\ \sin\alpha \end{bmatrix} \tag{25}$$

Where l_1 and l_2 are the lengths of elements $\hat{AO}$ and $\hat{OB}$ respectively. Their stiffness matrices will look like:

$$K^1 = k_1 \begin{bmatrix} c^2 & sc & -c^2 & -sc \\ & s^2 & -sc & -s^2 \\ & & c^2 & sc \\ \text{symm} & & & s^2 \end{bmatrix} \qquad K^2 = k_2 \begin{bmatrix} c^2 & sc & -c^2 & -sc \\ & s^2 & -sc & -s^2 \\ & & c^2 & sc \\ \text{symm} & & & s^2 \end{bmatrix} \tag{26}$$

where $k_i = (EA/L)_i$. The global stiffness matrix will be:

$$\left[\begin{array}{cc|cc|cc} k_1c^2 & k_1sc & -k_1c^2 & -k_1sc & 0 & 0 \\ & k_1s^2 & -k_1sc & -k_1s^2 & 0 & 0 \\ \hline & & (k_1+k_2)c^2 & (k_1+k_2)sc & -k_2c^2 & -k_2sc \\ & & & (k_1+k_2)s^2 & -k_2sc & -k_2s^2 \\ \hline & & & & k_2c^2 & k_2sc \\ \text{symm} & & & & & k_2s^2 \end{array}\right] \tag{27}$$

Since we only want to study the effects of this geometry on the critical node, we'll focus on the central 2×2 minor:

$$M_{22} = (k_1 + k_2) \begin{bmatrix} c^2 & sc \\ sc & s^2 \end{bmatrix} \tag{28}$$

The determinant of the minor, also known as the co-factor is:

$$C_{22} = det(M_{22}) = (k_1 + k_2)^2 (c^2 s^2 - s^2 c^2) = 0 \tag{29}$$

The determinant of the stiffness matrix can be rewritten as the determinants of the minors:

$$\det(K) = \det \begin{bmatrix} C_{11} & C_{12} & 0 \\ C_{21} & C_{22} & C_{23} \\ 0 & C_{23} & C_{33} \end{bmatrix} = \det \begin{bmatrix} C_{11} & C_{12} & 0 \\ C_{21} & 0 & C_{23} \\ 0 & C_{23} & C_{33} \end{bmatrix} = 0 \tag{30}$$

Hence it is proven that a node connected only to two colinear elements of arbitrary E, A, length and orientation will yield a singular system.

This holds true for any larger geometry since the critical rows and columns will only have more non-empty minors if there is an element connected to the critical node, which goes against the premise of the proof.

A.2 Classwork (10 Feb 2020)

This is the solution to the classwork proposed on February 10th, 2020. Unfortunately I don't have a copy of the problem statement. All values are shown in SI unless stated otherwise.

To start we must compute the local stiffness matrices:

$$K^1 = K^5 = 2 \times 10^4 \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix} \qquad K^2 = 2 \times 10^4 \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \end{bmatrix}$$

$$K^3 = 7,071 \begin{bmatrix} 1 & -1 & -1 & 1 \\ -1 & 1 & 1 & -1 \\ -1 & 1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \qquad K^4 = 7,071 \begin{bmatrix} 1 & 1 & -1 & -1 \\ 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 \\ -1 & -1 & 1 & 1 \end{bmatrix}$$

Perfoming the assembly yields:

$$K = 10^7 \begin{bmatrix} 0.7071 & 0.7071 & 0 & 0 & -0.7071 & -0.7071 & 0 & 0 \\ 0.7071 & 2.7071 & 0 & -2 & -0.7071 & -0.7071 & 0 & 0 \\ 0 & 0 & 2.7071 & -0.7071 & -2 & 0 & -0.7071 & 0.7071 \\ 0 & -2 & -0.7071 & 2.7071 & 0 & 0 & 0.7071 & -0.7071 \\ -0.7071 & -0.7071 & -2 & 0 & 2.7071 & 0.7071 & 0 & 0 \\ -0.7071 & -0.7071 & 0 & 0 & 0.7071 & 2.7071 & 0 & -2 \\ 0 & 0 & -0.7071 & 0.7071 & 0 & 0 & 0.7071 & -0.7071 \\ 0 & 0 & 0.7071 & -0.7071 & 0 & -2 & -0.7071 & 2.7071 \end{bmatrix} \tag{31}$$

Removing rows and columns with prescribed displacements (1, 2, 7, 8) and applying the loads returns:

$$10^7 \begin{bmatrix} 2.7071 & -0.7071 & -2 & 0 \\ -0.7071 & 2.7071 & 0 & 0 \\ -2 & 0 & 2.7071 & 0.7071 \\ 0 & 0 & 0.7071 & 2.7071 \end{bmatrix} \begin{bmatrix} u_{x2} \\ u_{y2} \\ u_{x3} \\ u_{y3} \end{bmatrix} = \begin{bmatrix} 80,000 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{32}$$

Solving we obtain the result:

$$\begin{bmatrix} u_{x2} \\ u_{y2} \\ u_{x3} \\ u_{y3} \end{bmatrix} = \begin{bmatrix} 8.541 \\ 2.231 \\ 6.772 \\ -1.769 \end{bmatrix} \times 10^{-3} \text{ m} \tag{33}$$

Moving on to internal forces, they can be calculated as:

$$x^e = (\boldsymbol{u}_2^e - \boldsymbol{u}_1^e) \cdot \boldsymbol{v} - l$$

$$F_{\text{int}}^e = \left(\frac{EA}{l}\right)^e x^e$$

where for each element e; x is the elongation in the axial direction, $\boldsymbol{u}_i$ is the displacement of node i, $\boldsymbol{v}$ is the unit vector from node 1 to 2 and l is the original length. F_{int} is the internal force. The result is:

$$\begin{bmatrix} F_{\text{int}}^1 \\ F_{\text{int}}^2 \\ F_{\text{int}}^3 \\ F_{\text{int}}^4 \\ F_{\text{int}}^5 \end{bmatrix} = \begin{bmatrix} 44.62 \\ -35.37 \\ -63.10 \\ 50.03 \\ -35.37 \end{bmatrix} \times 10^{3} \text{ N} \tag{34}$$

where positive forces indicate tension, and negative compression.

A.3 Matlab code

```
DISP = 1; LOAD = 2; %Keywords
%% Data entry
% Variable declararion
L = sym('L', 'positive');
E = sym('E', 'positive');
A = sym('A', 'positive');
a = sym('a', 'positive');
P = sym('P','real');
H = sym('H','real');

% Geometry
X = [0, -L;
     -L*sin(a)/cos(a), 0
    0,0;
    L*sin(a)/cos(a),0];

T = [1,2;
     1,3;
     1,4];

% Boundary conditions
BC = {[LOAD H], [LOAD -P];
      [DISP 0], [DISP 0];
      [DISP 0], [DISP 0];
      [DISP 0], [DISP 0]};

 %% Assembly

 n_elems = size(T,1);
 n_points = size(X,1);

 K = 0*sym('K',[n_points*2, n_points*2]);

 for el=1:n_elems
    X1 = X(T(el,1),:);
    X2 = X(T(el,2),:);
    len = sqrt((X1(1)-X2(1))^2 + (X1(2)-X2(2))^2);
    c = (X2(1) - X1(1))/len;
    s = (X2(2) - X1(2))/len;
    R = [c s 0 0; -s c 0 0; 0 0 c s; 0 0 -s c];
    K_local = E*A/len * R' * [1 0 -1 0; 0 0 0 0; -1 0 1 0; 0 0 0 0] * R;

    for i=1:4
        for j=1:4
            axis_i = mod(i+1,2)+1;
            axis_j = mod(j+1,2)+1;
```

```
            local_i = floor((i-1)/2)+1;
            local_j = floor((j-1)/2)+1;

            global_i = T(el,local_i);
            global_j = T(el,local_j);

            I = (global_i-1) * 2 + axis_i;
            J = (global_j-1) * 2 + axis_j;

            K(I,J) = K(I,J) + K_local(i,j);
        end
    end
 end

  %% Applying BC
 F = 0*sym('F',[n_points*2,1]);
 DOF_free = [];
 DOF_presc = [];
 u_presc = [];

for pt = n_points:-1:1
    for axis = [2,1]
        i = (pt-1)*2 + axis;

        if BC{pt,axis}(1) == LOAD  % Free displacements
            F(i) = BC{pt,axis}(2);
            DOF_free = [i, DOF_free];

        elseif BC{pt,axis}(1) == DISP  % Prescribed displacements
            DOF_presc = [i, DOF_presc];
            u_presc = [BC{pt,axis}(2), u_presc];
            F = F - K(:,i)*BC{pt,axis}(2);
            K(i,:) = [];
            K(:,i) = [];
            F(i) = [];
        end
    end
end

%% Solution

u_free = simplify(K\F);

U(DOF_free,1) = u_free;
U(DOF_presc,1) = u_presc;
u = reshape(U, [2, n_points])'

InternalForces = 0*sym('IF',[n_elems, 1]);
```

```
for el = 1:n_elems
    node1 = T(el,1);
    node2 = T(el,2);
    len = sqrt((X1(1)-X2(1))^2 + (X1(2)-X2(2))^2);
    v = (X(node2,:) - X(node1,:))/len; % Director vector
    X1_new = X(node1,:) + u(node1,:);
    X2_new = X(node2,:) + u(node2,:);
    x = (X2_new - X1_new)*v' - len; %Change in elongation projected onto axial direction
    InternalForces(el) = E*A / len * x;
end
```