MATH5004 LAB 8

2D FEM: The plane truss element

The plane truss element has modulus of elasticity E, cross-sectional area A, and length L. The plane trust element has four degrees of freedom. Let $C = \cos\theta$ and $S = \sin\theta$ and the element stiffness matrix is given by

$$k^e = \frac{EA}{L}\begin{bmatrix} C^2 & CS & -C^2 & -CS \\ CS & S^2 & -CS & -S^2 \\ -C^2 & -CS & C^2 & CS \\ -CS & -S^2 & CS & S^2 \end{bmatrix},$$

The force for each element is

$$f^e = \frac{EA}{L}\begin{bmatrix} -C & -S & C & S \end{bmatrix}\mathbf{u}^e,$$

where $\mathbf{u}^e = [u_{1x},\ u_{1y},\ u_{2x},\ u_{2y}]^T$ is element displacement vector. If n is number of nodes in the area A, the global stiffness matrix K has dimension $2n \times 2n$. The element stress is obtained by dividing the element force by the cross sectional area A. Assemble all element matrices and element force vectors, we obtain a global system of equations

$$\mathbf{KU} = \mathbf{F}.$$

If there is an inclined support at one of the nodes of the truss then the global stiffness matrix needs to be modified using the following equation:

$$\mathbf{K}_{new} = \mathbf{T}\,\mathbf{K}_{new}\mathbf{T}^T,$$

where $\mathbf{T}$ is a $2n\times 2n$ transformation matrix that is obtained by making a call to the MATLAB function *PlaneTrussInclinedSupport*.

Example. Consider the plane truss with modulus of elasticity $E = 210GPa$ and cross-sectional area $A = 1 \times 10^{-4}\ m^2$.

Find:

1. the global stiffness matrix for the structure.
2. the horizontal displacement at node 2.
3. the horizontal and vertical displacements at node 3.
4. the reactions at nodes 1 and 2.
5. the stress in each element.

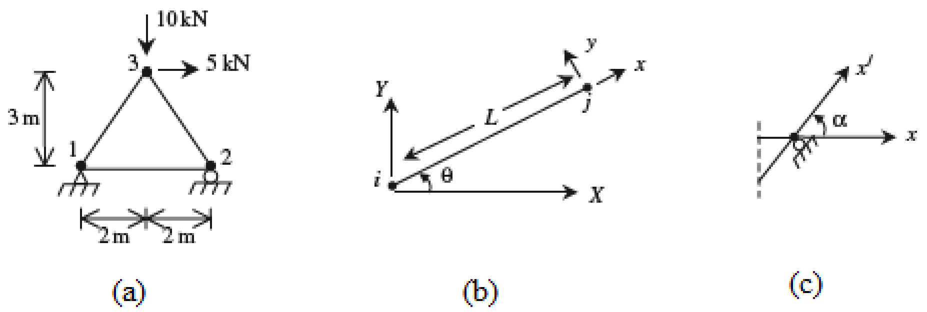


Fig 1. Plane Truss with Three Elements (a), and two nodal truss element (b) with inclined support (c).

MATLAB Functions

The following MATLAB functions are used for this problem:

- *PlaneTrussElementLength.m*
- *PlaneTrussElementStiffness.m*,
- *PlaneTrussAssemble.m*
- *PlaneTrussElementForce.m*
- *PlaneTrussElementStress.m*
- *PlaneTrussInclinedSupport.m*

```
function y = PlaneTrussElementLength(x1,y1,x2,y2)
%PlaneTrussElementLength This function returns the length of the
% plane truss element whose first node has
% coordinates (x1, y1) and second node has
% coordinates (x2, y2).
y = sqrt((x2-x1)*(x2-x1) + (y2-y1)*(y2-y1));
```

```
function y = PlaneTrussAssemble(K,k,i,j)
%PlaneTrussAssemble This function assembles the element stiffness
% matrix k of the plane truss element with nodes
% i and j into the global stiffness matrix K.
% This function returns the global stiffness
% matrix K after the element stiffness matrix
% k is assembled.
K(2*i-1,2*i-1) = K(2*i-1,2*i-1) + k(1,1) ;
K(2*i-1,2*i) = K(2*i-1,2*i) + k(1,2) ;
K(2*i-1,2*j-1) = K(2*i-1,2*j-1) + k(1,3) ;
K(2*i-1,2*j) = K(2*i-1,2*j) + k(1,4) ;
K(2*i,2*i-1) = K(2*i,2*i-1) + k(2,1) ;
K(2*i,2*i) = K(2*i,2*i) + k(2,2) ;
K(2*i,2*j-1) = K(2*i,2*j-1) + k(2,3) ;
K(2*i,2*j) = K(2*i,2*j) + k(2,4) ;
K(2*j-1,2*i-1) = K(2*j-1,2*i-1) + k(3,1) ;
K(2*j-1,2*i) = K(2*j-1,2*i) + k(3,2) ;
K(2*j-1,2*j-1) = K(2*j-1,2*j-1) + k(3,3) ;
K(2*j-1,2*j) = K(2*j-1,2*j) + k(3,4) ;
K(2*j,2*i-1) = K(2*j,2*i-1) + k(4,1) ;
K(2*j,2*i) = K(2*j,2*i) + k(4,2) ;
K(2*j,2*j-1) = K(2*j,2*j-1) + k(4,3) ;
K(2*j,2*j) = K(2*j,2*j) + k(4,4) ;
y = K;
```

```
function y = PlaneTrussElementStiffness(E,A,L, theta)
%PlaneTrussElementStiffness This function returns the element
% stiffness matrix for a plane truss
% element with modulus of elasticity E,
% cross-sectional area A, length L, and
% angle theta (in degrees).
% The size of the element stiffness
% matrix is 4 x 4.
x = theta*pi/180;
C = cos(x);
S = sin(x);
y = E*A/L*[C*C C*S -C*C -C*S ; C*S S*S -C*S -S*S ;
           -C*C -C*S C*C C*S ; -C*S -S*S C*S S*S] ;
```

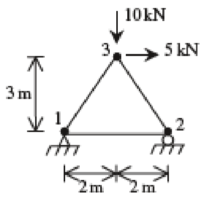
```
function y = PlaneTrussElementForce(E,A,L,theta,u)
%PlaneTrussElementForce This function returns the element force
% given the modulus of elasticity E, the
% cross-sectional area A, the length L,
% the angle theta (in degrees), and the
% element nodal displacement vector u.
x = theta* pi/180;
C = cos(x);
S = sin(x);
y = E*A/L*[-C -S C S]* u;
```

```
function y = PlaneTrussElementStress(E,L,theta,u)
%PlaneTrussElementStress This function returns the element stress
% given the modulus of elasticity E, the length L, the angle theta (in
% degrees), and the element nodal displacement vector u.
x = theta * pi/180;
C = cos(x);
S = sin(x);
y = E/L*[-C -S C S]* u;
```

```
function y = PlaneTrussInclinedSupport(T,i,alpha)
%PlaneTrussInclinedSupport This function calculates the tranformation
% matrix T of the inclined support at node i with angle of
% inclination alpha (in degrees).
x = alpha*pi/180;
T(2*i-1,2*i-1) = cos(x) ;
T(2*i-1,2*i) = sin(x) ;
T(2*i,2*i-1) = -sin(x) ;
T(2*i,2*i) = cos(x) ;
y = T;
```

Step 1 *Discretizing the Domain*

Element Number	Node *i*	Node *j*
1	1	2
2	1	3
3	2	3



Step 2 *Writing the Element Stiffness Matrices*

The tool ***PlaneTrussElementStiffness.m*** are used to obtain three element stiffness matrices $k^{(1)}$, $k^{(2)}$, and $k^{(3)}$.

```
» E=210e6

E =

   210000000
```

```
» A=1e-4

A =

   1.0000e-004


» L1=4

L1 =

   4

» L2=PlaneTrussElementLength(0,0,2,3)

L2 =

   3.6056


» L3=PlaneTrussElementLength(0,0,-2,3)

L3 =

    3.6056

» k1=PlaneTrussElementStiffness(E,A,L1,0)

k1 =

      5250     0    -5250     0
         0     0        0     0
     -5250     0     5250     0
         0     0        0     0


» theta2=atan(3/2)*180/pi

theta2 =

   56.3099


» theta3=180-theta2

theta3 =

   123.6901
```

```
» k2=PlaneTrussElementStiffness(E,A,L2,theta2)

k2 =

  1.0e+003 *

     1.7921    2.6882   -1.7921   -2.6882
     2.6882    4.0322   -2.6882   -4.0322
    -1.7921   -2.6882    1.7921    2.6882
    -2.6882   -4.0322    2.6882    4.0322

» k3=PlaneTrussElementStiffness(E,A,L3,theta3)

k3 =

  1.0e+003 *

     1.7921   -2.6882   -1.7921    2.6882
    -2.6882    4.0322    2.6882   -4.0322
    -1.7921    2.6882    1.7921   -2.6882
     2.6882   -4.0322   -2.6882    4.0322
```

```
» K=PlaneTrussAssemble(K,k2,1,3)

K =

  1.0e+003 *

    7.0421    2.6882   -5.2500    0    -1.7921   -2.6882
    2.6882    4.0322         0    0    -2.6882  -4.0322
```

Step 3 *Assembling the Global Stiffness Matrix*

As the structure has three nodes, each node has two degree of freedoms (displacements in x and y direction), the size of the global stiffness matrix is 6 × 6. We firstly initialize a zero matrix K of size 6×6.

```
» K=zeros(6,6)

K =

     0     0     0     0     0     0
     0     0     0     0     0     0
     0     0     0     0     0     0
     0     0     0     0     0     0
     0     0     0     0     0     0
     0     0     0     0     0     0
```

We then make three calls to the MATLAB function *PlaneTrussAssemble.m*

```
» K=PlaneTrussAssemble(K,k1,1,2)

K =


    5250    0   -5250    0    0    0
       0    0       0    0    0    0
   -5250    0    5250    0    0    0
       0    0       0    0    0    0
       0    0       0    0    0    0
       0    0       0    0    0    0

» K=PlaneTrussAssemble(K,k2,1,3)

K =

  1.0e+003 *

    7.0421    2.6882   -5.2500    0   -1.7921   -2.6882
    2.6882    4.0322         0    0   -2.6882   -4.0322

   -5.2500         0    5.2500    0         0         0
         0         0         0    0         0         0
   -1.7921   -2.6882         0    0    1.7921    2.6882
   -2.6882   -4.0322         0    0    2.6882    4.0322

» K=PlaneTrussAssemble(K,k3,2,3)

K =

  1.0e+003 *


    7.0421    2.6882   -5.2500         0   -1.7921   -2.6882
    2.6882    4.0322         0         0   -2.6882   -4.0322
   -5.2500         0    7.0421   -2.6882   -1.7921    2.6882
         0         0   -2.6882    4.0322    2.6882   -4.0322
   -1.7921   -2.6882   -1.7921    2.6882    3.5842    0.0000
   -2.6882   -4.0322    2.6882   -4.0322    0.0000    8.0645
```

Step 4 *Applying the Boundary Conditions*

The boundary conditions are

$$U1x = U1y = U2y = 0, \qquad F2x = 0, \qquad F3x = 5, \; F3y = -10$$

Imposing BCs to the global stiffness matrix **K** obtained in the previous step:

$$\mathbf{KU} = \mathbf{F},$$

where

$$\mathbf{U} = \begin{bmatrix} U_{1x} \\ U_{1y} \\ U_{2x} \\ U_{2y} \\ U_{3x} \\ U_{3y} \end{bmatrix} \text{ and } \mathbf{F} = \begin{bmatrix} F_{1x} \\ F_{1y} \\ F_{2x} \\ F_{2y} \\ F_{3x} \\ F_{3y} \end{bmatrix}.$$

We then obtain

$$10^3 \begin{bmatrix} 7.041 & 2.6882 & -5.2500 & 0 & -1.7921 & -2.6882 \\ 2.6882 & 4.0322 & 0 & 0 & -2.6882 & -4.0322 \\ -5.2500 & 0 & 7.0421 & -2.6882 & -1.7921 & 2.6882 \\ 0 & 0 & -2.6882 & 4.0322 & 2.6882 & -4.0322 \\ -1.7921 & -2.6882 & -1.7921 & 2.6882 & 3.5842 & 0 \\ -2.6882 & -4.0322 & 2.6882 & -4.0322 & 0 & 8.0645 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ U_{2x} \\ 0 \\ U_{3x} \\ U_{3y} \end{bmatrix} = \begin{bmatrix} F_{1x} \\ F_{1y} \\ 0 \\ F_{2y} \\ 5 \\ -10 \end{bmatrix}$$

Step 5 *Solving the Equations*

The solution of the following system

$$10^3 \begin{bmatrix} 7.0421 & -1.7921 & 2.6882 \\ -1.7921 & 3.5842 & 0 \\ 2.6882 & 0 & 8.0645 \end{bmatrix} \begin{bmatrix} U_{2x} \\ U_{3x} \\ U_{3y} \end{bmatrix} = \begin{bmatrix} 0 \\ 5 \\ -10 \end{bmatrix}$$

is obtained using MATLAB code as follows:

```
» k=[K(3,3) K(3,5:6) ; K(5:6,3) K(5:6,5:6)]

k =

   1.0e+003 *

      7.0421     -1.7921     2.6882
     -1.7921      3.5842     0.0000
      2.6882      0.0000     8.0645

» f=[0 ; 5 ; -10]

f =
        0
        5
      -10
```

```
» u=k\f

u =

    0.0011
    0.0020
   -0.0016
```

which means that the horizontal displacement at node 2 is 0.0011 m, and the horizontal and vertical displacements at node 3 are 0.0020m and −0.0016m, respectively.

Step 6 *Post-processing*

This step is to determine the reactions at nodes 1 and 2, and the stress in each element.

- set up the global nodal displacement vector **U,** then we calculate the global nodal force vector **F**.

```
» U=[0 ; 0 ; u(1) ; 0 ; u(2:3)]

U =

         0
         0
    0.0011
         0
    0.0020
   -0.0016

» F=K*U

F =

   -5.0000
    1.2500
   -0.0000
    8.7500
    5.0000
  -10.0000
```

The result F indicates that

- the horizontal and vertical reactions at node 1 are forces of 5 kN (directed to the left) and 1.25 kN (directed upwards).
- The vertical reaction at node 2 is a force of 8.75N (directed upwards).

- Set up the element nodal displacement vectors $\mathbf{u}^{(1)}$, $\mathbf{u}^{(2)}$, and $\mathbf{u}^{(3)}$

```
» u1=[U(1) ; U(2) ; U(3) ; U(4)]

u1 =

         0
         0
    0.0011
         0
```

```
» u2=[U(1) ; U(2) ; U(5) ; U(6)]

u2 =

          0
          0
     0.0020
    -0.0016

» u3=[U(3) ; U(4) ; U(5) ; U(6)]

u3 =

     0.0011
          0
     0.0020
    -0.0016
```

➢ Use the tool *PlaneTrussElementStress.m* to Calculate the element stresses *sigma1*, *sigma2*, and *sigma3.*

```
» sigma1=PlaneTrussElementStress(E,L1,0,u1)

sigma1 =

     5.8333e+004

» sigma2=PlaneTrussElementStress(E,L2,theta2,u2)

sigma2 =

    -1.5023e+004

» sigma3=PlaneTrussElementStress(E,L3,theta3,u3)

sigma3 =

    -1.0516e+005
```

We conclude that the stress in each element as follows:

- In element 1, the stress is 1 58.3333MPa (tensile),
- In element 2, the stress is 15.023MPa (compressive),
- In element 3, the stress is 105.16MPa (compressive).

-------------------------End of Document -------------------------