MATH5004 LAB 10
2D Finite Element Method

Consider the thin plate subjected to a uniformly distributed load as shown in Fig. 1(a) below.

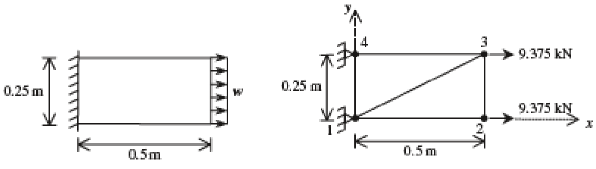

Fig. 1 Thin plate and Discretization of Thin Plate Using Two Linear Triangles

The plate is discretised using two linear triangular elements as shown in Fig. 1(b).
Given E = 210GPa, ν = 0.3, t = 0.025m, and $w = 3000\ kN/m^2$, a case of plane stress is assumed.
Determine:

1. the global stiffness matrix for the structure.
2. the horizontal and vertical displacements at nodes 2 and 3.
3. the reactions at nodes 1 and 4.
4. the stresses in each element.
5. the principal stresses and principal angle for each element.

Step 1 – Discretising the Domain:

The domain is subdivided into two elements and four nodes as shown in Fig. 1(b). (More elements must be used in order to obtain reliable results.) The total force due to the distributed load is divided equally between nodes 2 and 3. The units used in the MATLAB calculations are kN and meter. Table 1 shows the element connectivity for this example.

Table 1. Topology of domain mesh

Element number	Nodes		
	First node	Second node	Third node
1	1	3	4
2	1	2	3

Step 2 – Writing the Element Stiffness Matrices:

The two element stiffness matrices $K^{(1)}$ and $K^{(2)}$ are obtained by the MATLAB function *LinearTriangleElementStiffness*. Each matrix has size 6 × 6.

```
>> E=210e6

    E =

        210000000

>>NU=0.3

   NU =

        0.3000

>>t=0.025

    t =

         0.0250
```

```
>>k1=LinearTriangleElementStiffness(E,NU,t,0,0,0.5,0.25,0,0.25,1)
k1 =

   1.0e+06 *

    2.0192         0         0   -1.0096   -2.0192    1.0096
         0    5.7692   -0.8654         0    0.8654   -5.7692
         0   -0.8654    1.4423         0   -1.4423    0.8654
   -1.0096         0         0    0.5048    1.0096   -0.5048
   -2.0192    0.8654   -1.4423    1.0096    3.4615   -1.8750
    1.0096   -5.7692    0.8654   -0.5048   -1.8750    6.2740

>>k2=LinearTriangleElementStiffness(E,NU,t,0,0,0.5,0,0.5,0.25,1)
k2 =

   1.0e+06 *

    1.4423         0   -1.4423    0.8654         0   -0.8654
         0    0.5048    1.0096   -0.5048   -1.0096         0
   -1.4423    1.0096    3.4615   -1.8750   -2.0192    0.8654
    0.8654   -0.5048   -1.8750    6.2740    1.0096   -5.7692
         0   -1.0096   -2.0192    1.0096    2.0192         0
   -0.8654         0    0.8654   -5.7692         0    5.7692
```

Step 3 – Assembling the Global Stiffness Matrix:

Since the structure has four nodes, the size of the global stiffness matrix is 8×8. Therefore to obtain K we first set up a zero matrix of size 8×8 then make two calls to the MATLAB function *LinearTriangleAssemble* since we have two elements in the structure. Each call to the function will assemble one element. The following are the MATLAB commands:

```
>> K=zeros(8,8)
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     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
>> K=LinearTriangleAssemble(K,k1,1,3,4)
K =

   1.0e+06 *

    2.0192         0         0         0         0   -1.0096   -2.0192    1.0096
         0    5.7692         0         0   -0.8654         0    0.8654   -5.7692
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
         0   -0.8654         0         0    1.4423         0   -1.4423    0.8654
   -1.0096         0         0         0         0    0.5048    1.0096   -0.5048
   -2.0192    0.8654         0         0   -1.4423    1.0096    3.4615   -1.8750
    1.0096   -5.7692         0         0    0.8654   -0.5048   -1.8750    6.2740
>> K=LinearTriangleAssemble(K,k2,1,2,3)
K =

   1.0e+06 *

    3.4615         0   -1.4423    0.8654         0   -1.8750   -2.0192    1.0096
         0    6.2740    1.0096   -0.5048   -1.8750         0    0.8654   -5.7692
   -1.4423    1.0096    3.4615   -1.8750   -2.0192    0.8654         0         0
    0.8654   -0.5048   -1.8750    6.2740    1.0096   -5.7692         0         0
         0   -1.8750   -2.0192    1.0096    3.4615         0   -1.4423    0.8654
   -1.8750         0    0.8654   -5.7692         0    6.2740    1.0096   -0.5048
   -2.0192    0.8654         0         0   -1.4423    1.0096    3.4615   -1.8750
    1.0096   -5.7692         0         0    0.8654   -0.5048   -1.8750    6.2740
```

Step 4 – Applying the Boundary Conditions:

The matrix [K]{U} ={F} for this structure is obtained as follows using the global stiffness matrix obtained in the previous step:

$$10^6 \begin{bmatrix} 3.46 & 0 & -1.44 & 0.87 & 0 & 1.88 & -2.02 & 1.01 \\ 0 & 6.27 & 1.01 & -0.50 & -1.88 & 0 & 0.87 & -5.77 \\ -1.44 & 1.01 & 3.46 & -1.88 & -2.02 & 0.87 & 0 & 0 \\ 0.87 & -0.50 & -1.88 & 6.27 & 1.01 & -5.77 & 0 & 0 \\ 0 & -1.88 & -2.02 & 1.01 & 3.46 & 0 & -1.44 & 0.87 \\ -1.88 & 0 & 0.87 & -5.77 & 0 & 6.27 & 1.01 & -0.50 \\ -2.02 & 0.87 & 0 & 0 & -1.44 & 1.01 & 3.46 & -1.88 \\ 1.01 & -5.77 & 0 & 0 & 0.87 & -0.50 & -1.88 & 6.27 \end{bmatrix} \begin{bmatrix} U_{1x} \\ U_{1y} \\ U_{2x} \\ U_{2y} \\ U_{3x} \\ U_{3y} \\ U_{4x} \\ U_{4y} \end{bmatrix} = \begin{bmatrix} F_{1x} \\ F_{1y} \\ F_{2x} \\ F_{2y} \\ F_{3x} \\ F_{3y} \\ F_{4x} \\ F_{4y} \end{bmatrix} \qquad (*)$$

The boundary conditions for this problem are given as:

$$U_{1x} = U_{1y} = U_{4x} = U_{4x} = 0$$

$$F_{2x} = 9.375, F_{2y} = 0, F_{3x} = 9.375, F_{3y} = 0$$

Inserting the above conditions into $(*)$ we obtain:

$$10^6 \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3.46 & -1.88 & -2.02 & 0.87 & 0 & 0 \\ 0 & 0 & -1.88 & 6.27 & 1.01 & -5.77 & 0 & 0 \\ 0 & 0 & -2.02 & 1.01 & 3.46 & 0 & 0 & 0 \\ 0 & 0 & 0.87 & -5.77 & 0 & 6.27 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} U_{1x} \\ U_{1y} \\ U_{2x} \\ U_{2y} \\ U_{3x} \\ U_{3y} \\ U_{4x} \\ U_{4y} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 9.375 \\ 0 \\ 9.375 \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad (*)$$

Step 5 – Solving the Equations:

Solving the system of equations in (11.11) will be performed by partitioning (manually) and Gaussian elimination (with MATLAB). First we partition (11.11) by extracting the submatrix in rows 3 to 6 and columns 3 to 6. Therefore we obtain:

$$\begin{bmatrix} 3.46 & -1.88 & -2.02 & 0.87 \\ -1.88 & 6.27 & 1.01 & -5.77 \\ -2.02 & 1.01 & 3.46 & 0 \\ 0.87 & -5.77 & 0 & 6.27 \end{bmatrix} \begin{bmatrix} U_{2x} \\ U_{2y} \\ U_{3x} \\ U_{3y} \end{bmatrix} = \begin{bmatrix} 9.375 \\ 0 \\ 9.375 \\ 0 \end{bmatrix}$$

The solution of the above system is obtained using MATLAB as follows. Note that the backslash operator “\” is used for Gaussian elimination.

```
>> k = K(3:6, 3:6)

k =

  1.0e+06 *

   3.4615  -1.8750  -2.0192   0.8654
  -1.8750   6.2740   1.0096  -5.7692
  -2.0192   1.0096   3.4615        0
   0.8654  -5.7692        0   6.2740

>> f = [9.375 ; 0 ; 9.375 ; 0]

f =

   9.3750
        0
   9.3750
        0
>> u=k\f

u =

  1.0e-05 *

   0.7111
   0.1115
   0.6531
   0.0045
```

It is now clear that the horizontal and vertical displacements at node 2 are 0.7111m and 0.1115 m, respectively, and the horizontal and vertical displacements at node 3 are 0.6531m and 0.0045 m, respectively. When a larger number of elements is used we expect to get the same result for the horizontal displacements at nodes 2 and 3.

Step 6 – Post-processing:

In this step, we obtain the reactions at nodes 1 and 4, and the stresses in each element. First, we set up the global nodal displacement vector U, then we calculate the global nodal force vector F.

```
>> U=[0 ; 0 ; u ; 0 ; 0]

U =

  1.0e-05 *

        0
        0
   0.7111
   0.1115
   0.6531
   0.0045
        0
        0

>> F=K*U

F =

  -9.3750
  -5.6295
   9.3750
   0.0000
   9.3750
  -0.0000
  -9.3750
   5.6295
```

Thus, the horizontal and vertical reactions at node 1 are forces of 9.375 kN (directed to the left) and 5.6295 kN (directed downwards). The horizontal and vertical reactions at node 4 are forces of 9.375 kN (directed to the left) and 5.6295 kN (directed upwards). Obviously force equilibrium is satisfied for this problem. Next we set up the element nodal displacement vectors u_1 and u_2 then we calculate the element stresses *sigma1* and *sigma2* by making calls to the MATLAB function *LinearTriangleElementStresses*.

```
>> u1=[U(1) ; U(2) ; U(5) ; U(6) ; U(7) ; U(8)]

u1 =

  1.0e-05 *

        0
        0
   0.6531
   0.0045
        0
        0

>> u2=[U(1) ; U(2) ; U(3) ; U(4) ; U(5) ; U(6)]

u2 =

  1.0e-05 *

        0
        0
   0.7111
   0.1115
   0.6531
   0.0045
```

```
>> sigma1=LinearTriangleElementStresses(E,NU,0,0,0.5,0.25,0,0.25,1,u1 )

sigma1 =

   1.0e+03 *

    3.0144
    0.9043
    0.0072

>> sigma2=LinearTriangleElementStresses(E,NU,0,0,0.5,0,0.5,0.25,1,u2)

sigma2 =

   1.0e+03 *

    2.9856
   -0.0036
   -0.0072
```

Thus it is clear that the stresses in element 1 are $\sigma_x = 3.0144MPa$ (tensile), $\sigma_y = 0.9043MPa$ (tensile), and $\tau_{xy} = 0.0072MPa$ (positive). The stresses in element 2 are $\sigma_x = 2.9856MPa$ (tensile), $\sigma_y = 0.0036MPa$ (compressive), and $\tau_{xy} = 0.0072MPa$ (negative). It is clear that the stresses in the x-direction approach closely the correct value of 3MPa (tensile). Next we calculate the principal stresses and principal angle for each element by making calls to the MATLAB function *LinearTriangleElementPStresses*.

```
>> s1=LinearTriangleElementPStresses(sigma1)

s1 =

   1.0e+03 *

    3.0144
    0.9043
    0.0002

>> s2=LinearTriangleElementPStresses(sigma2)

s2 =

   1.0e+03 *

    2.9856
   -0.0036
   -0.0001
```

Thus it is clear that the principal stresses in element 1 are $\sigma_1 = 3.0144MPa$ (tensile), $\sigma_2 = 0.9043MPa$ (tensile), while the principal angle $\theta_p = 0.2^o$. The principal stresses in element 2 are $\sigma_1 = 2.9856MPa$ (tensile), $\sigma_2 = 0.0036MPa$ (compressive), while the principal angle $\theta_p = -0.1^o$.

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