Linear Elastic Finite Element Coding And Analysis

ME 434/613 : FINITE ELEMENT AND BOUNDARY ELEMENT METHODS

by

Group 1

140100097	**P. V. Sai Sushant**	**Credit**
140100099	**T. Sai Goutam**	**Credit**
140100101	**P. Vachan Deep**	**Credit**
14D170011	**Palash Gajjar**	**Credit**
14D170017	**Amey Mankar**	**Credit**

under the guidance of

Prof. Parag Tandaiya

Department of Mechanical Engineering
Indian Institute of Technology, Bombay
Mumbai 400 076

Contents

Part 1

Results

1.1 Uni–axial patch test (plane stress)

Schematic of the test case is shown in Figure 1.1. A uniform stress $\sigma = 1$ is applied on the edge 1–2–3. $E = 100$ and $\nu = 0.3$.

The theoretical values of stresses are $\sigma_y = 1$, $\sigma_x = 0$ and $\tau_{xy} = 0$. The theoretical strains are $\epsilon_y = \sigma_y/E$, $\epsilon_x = -\nu\epsilon_y$ and $\gamma_{xy} = \tau_{xy}/G$[1]. As is clear from Table 1.1, the values for stress and strain obtained by simulation indeed match well with theoretical values.

The theoretical displacement field for this problem is $u = x\epsilon_x = -(\nu/E)x$ and $v = y\epsilon_y = y/E$. The nodal displacements obtained from the simulation are shown in Table 1.2.

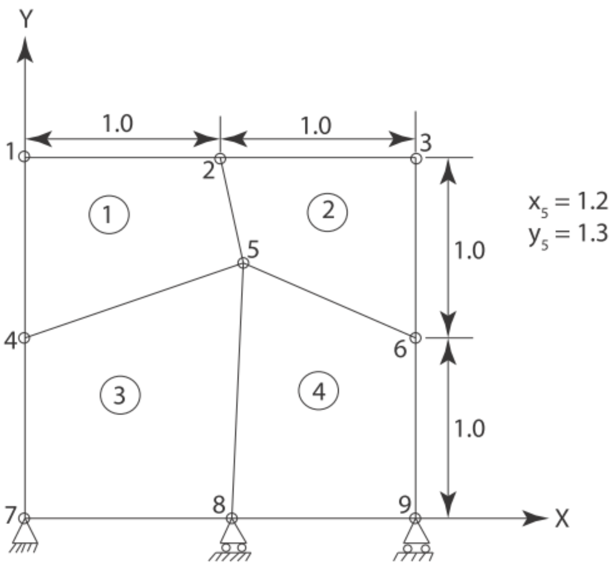


Figure 1.1: Mesh for problem (a)

[1] $G = \frac{E}{2(1+\nu)}$

Element #	σ_x	σ_y	ϵ_x	ϵ_y
1	-0.7866E-16	0.1000E+01	-0.3000E-02	0.1000E-01
2	-0.1880E-15	0.1000E+01	-0.3000E-02	0.1000E-01
3	0.5370E-15	0.1000E+01	-0.3000E-02	0.1000E-01
4	0.2511E-15	0.1000E+01	-0.3000E-02	0.1000E-01

Table 1.1: Stresses and strains at output gauss points of the elements for problem (a). Theoretical values are $\sigma_y = 1$, $\sigma_x = 0$, $\epsilon_y = 0.01$ and $\epsilon_x = -0.003$.

Node #	u	v
1	0.0000E+00	0.2000E-01
2	-0.3000E-02	0.2000E-01
3	-0.6000E-02	0.2000E-01
4	0.0000E+00	0.1000E-01
5	-0.3600E-02	0.1300E-01
6	-0.6000E-02	0.1000E-01
7	0.0000E+00	0.0000E+00
8	-0.3000E-02	0.0000E+00
9	-0.6000E-02	0.0000E+00

Table 1.2: Nodal displacements for problem (a). Theoretical displacement field is $u = -0.003x$ and $v = 0.01y$.

1.2 Shear patch test (plane stress)

Schematic of the test case is shown in Figure 1.2. A constant shear stress of $\tau_{xy} = 1.0$ is applied on the edge 1–2–3. $E = 100^7$ and $\nu = 0.3$.

The theoretical values of stresses are $\sigma_y = 0$, $\sigma_x = 0$ and $\tau_{xy} = 1.0$. The theoretical strains are $\epsilon_y = 0$, $\epsilon_x = 0$ and $\gamma_{xy} = \tau_{xy}/G = 0.26E-06$. As is clear from Table 1.3, the values for stress and strain obtained by simulation indeed match well with theoretical values.

The nodal displacements obtained from the simulation are shown in Table 1.4.

Element #	σ_x	σ_y	τ_{xy}	ϵ_x	ϵ_y	γ_{xy}
1	-0.16E-15	0.26E-17	0.10E+01	-0.16E-22	0.50E-23	0.26E-06
2	-0.27E-15	-0.42E-15	0.10E+01	-0.14E-22	-0.34E-22	0.26E-06
3	-0.19E-15	-0.96E-16	0.10E+01	-0.17E-22	-0.38E-23	0.26E-06
4	-0.41E-15	0.12E-15	0.10E+01	-0.45E-22	0.24E-22	0.26E-06

Table 1.3: Stresses and strains at output gauss points of the elements for problem (b). The theoretical values of stresses are $\sigma_y = 0$, $\sigma_x = 0$ and $\tau_{xy} = 1.0$. The theoretical strains are $\epsilon_y = 0$, $\epsilon_x = 0$ and $\gamma_{xy} = \tau_{xy}/G = 0.26E-06$.

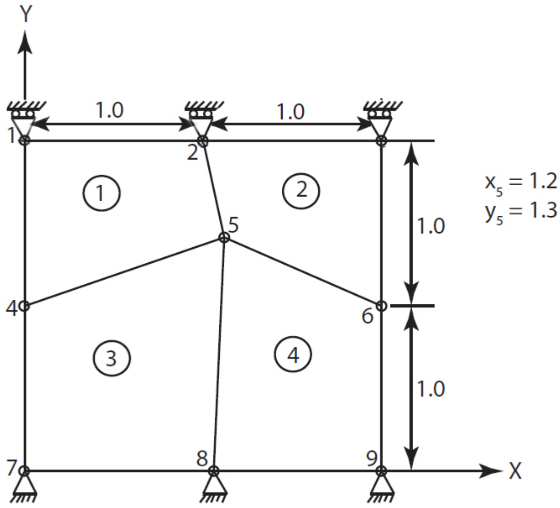


Figure 1.2: Mesh for problem (b)

Node #	u	v
1	0.5200E-06	0.0000E+00
2	0.5200E-06	0.0000E+00
3	0.5200E-06	0.0000E+00
4	0.2600E-06	0.0000E+00
5	0.3380E-06	0.0000E+00
6	0.2600E-06	0.0000E+00
7	0.0000E+00	0.0000E+00
8	0.0000E+00	0.0000E+00
9	0.0000E+00	0.0000E+00

Table 1.4: Nodal displacements for problem (b).

1.3 Uni–axial patch test (axi–symmetric)

The schematic for this test is as shown in Figure 1.1. Again, a uniform stress $\sigma = 1$ is applied on the edge 1–2–3. $E = 100$ and $\nu = 0.3$.

The theoretical values of stresses are $\sigma_r = 0$, $\sigma_z = 1$, $\sigma_\theta = 0$, $\tau_{rz} = 0$. The corresponding values of strains are $\epsilon_z = \sigma_z/E$, $\epsilon_r = \epsilon_\theta = -\nu\epsilon_z$ and $\gamma_{rz} = \tau_{rz}/G$. Stresses and strains obtained from simulation are shown in Tables 1.5 and 1.6 respectively. Clearly, there is good agreement between numerical and analytical values.

The theoretical displacement field for this problem is $u = r\epsilon_\theta = -(\nu/E)r$ and $v = z\epsilon_z = z/E$. The nodal displacements obtained from simulation are shown in Table 1.7.

1.4 Shear patch test (Plane strain)

Schematic of the test case is shown in Figure 1.7. A constant shear stress of $\tau_{xy} = 1$ is applied on the edge 1–2–3. $E = 100^7$ and $\nu = 0.3$.

Element #	σ_r	σ_z	σ_θ	τ_{rz}
1	-0.1004E-07	0.1000E+01	-0.8011E-08	0.1094E-07
2	0.1886E-08	0.1000E+01	0.1326E-08	-0.2563E-08
3	-0.1871E-08	0.1000E+01	0.1056E-08	0.2888E-08
4	-0.1004E-08	0.1000E+01	0.4487E-09	0.7219E-10

Table 1.5: Stress at output gauss points for problem (c). Theoretical values are $\sigma_r = \sigma_\theta = \tau_{rz} = 0$ and $\sigma_z = 1$.

Element #	ϵ_r	ϵ_z	ϵ_θ	γ_{rz}
1	-0.3000E-02	0.1000E-01	-0.3000E-02	0.2844E-09
2	-0.3000E-02	0.1000E-01	-0.3000E-02	-0.6663E-10
3	-0.3000E-02	0.1000E-01	-0.3000E-02	0.7509E-10
4	-0.3000E-02	0.1000E-01	-0.3000E-02	0.1877E-11

Table 1.6: Strain at output gauss points for problem (c). Theoretical values are $\epsilon_r = \epsilon_\theta = -0.003$, $\epsilon_z = 0.01$ and $\gamma_{rz} = 0$.

The theoretical values of stresses are $\sigma_y = 0$, $\sigma_x = 0$ and $\tau_{xy} = 1.0$. The theoretical strains are $\epsilon_y = 0$, $\epsilon_x = 0$ and $\gamma_{xy} = \tau_{xy}/G = 0.26E - 06$. As is clear from Table 1.8, the values for stress and strain obtained by simulation indeed match well with theoretical values.

The nodal displacements obtained from the simulation are shown in Table 1.9.

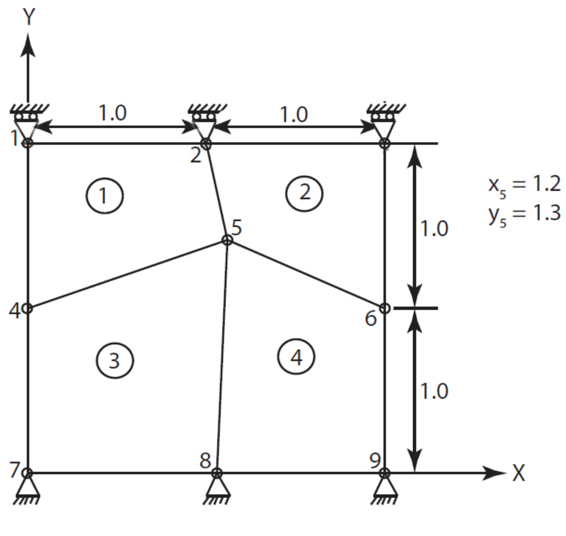


Figure 1.3: Mesh for problem (d)

Node #	u	v
1	0.0000E+00	0.2000E-01
2	-0.3000E-02	0.2000E-01
3	-0.6000E-02	0.2000E-01
4	0.0000E+00	0.1000E-01
5	-0.3600E-02	0.1300E-01
6	-0.6000E-02	0.1000E-01
7	0.0000E+00	0.0000E+00
8	-0.3000E-02	0.0000E+00
9	-0.6000E-02	0.0000E+00

Table 1.7: Nodal displacements for problem (c). Note that these are identical to those given in Table 1.2. Theoretical displacement field is $u = -0.003r$ and $v = 0.01z$.

Element #	σ_x	σ_y	τ_{xy}	ϵ_x	ϵ_y	γ_{xy}
1	-0.76E-15	-0.31E-15	0.10E+01	-0.58E-22	0.20E-23	0.26E-06
2	0.60E-15	0.42E-15	0.10E+01	0.38E-22	0.15E-22	0.26E-06
3	-0.18E-16	-0.24E-16	0.10E+01	-0.67E-24	-0.15E-23	0.26E-06
4	-0.50E-15	-0.33E-15	0.10E+01	-0.32E-22	-0.10E-22	0.26E-06

Table 1.8: Stresses and strains at output gauss points of the elements for problem (d). The theoretical values of stresses are $\sigma_y = 0$, $\sigma_x = 0$ and $\tau_{xy} = 1.0$. The theoretical strains are $\epsilon_y = 0$, $\epsilon_x = 0$ and $\gamma_{xy} = \tau_{xy}/G = 0.26E - 06$.

1.5 Thick cylinder subjected to internal pressure

Analytical solution for this problem is given by

$$\sigma_r = \frac{a^2 p}{b^2 - a^2}\left(1 - \frac{b^2}{r^2}\right)$$
$$\sigma_\theta = \frac{a^2 p}{b^2 - a^2}\left(1 + \frac{b^2}{r^2}\right),$$

Where, a is the inner radius, b is the outer radius of the hollow cylinder and p is the internal pressure. For this case, with $a = 1$, $b = 2$ and $p = 1$, we get

$$\sigma_r = \frac{1}{3}\left(1 - \frac{4}{r^2}\right)$$
$$\sigma_\theta = \frac{1}{3}\left(1 + \frac{4}{r^2}\right).$$

1.5.1 Treatment as plane strain case

The schematic of this case is given in Figure 1.4. Table 1.10 compares numerical and analytical values of σ_r and σ_θ for all the elements. It can be seen that the numerical values don't match very well with analytical ones but, at the same time they are not drastically off. This discrepancy can be explained by the fact that since 4 noded rectangles are being considered, the curves in the elements as depicted

Node #	u	v
1	0.5200E-06	0.0000E+00
2	0.5200E-06	0.0000E+00
3	0.5200E-06	0.0000E+00
4	0.2600E-06	0.0000E+00
5	0.3380E-06	0.0000E+00
6	0.2600E-06	0.0000E+00
7	0.0000E+00	0.0000E+00
8	0.0000E+00	0.0000E+00
9	0.0000E+00	0.0000E+00

Table 1.9: Nodal displacements for problem (d).

in Figure 1.4 are not captured correctly. Consequently, the domain itself is not modelled correctly with the inner and outer arcs being treated as 3 piece–wise linear segments. Hence, the non–agreement between numerical and analytical values is not entirely unexpected. Analytical vs Numerical stresses

Element #	σ_r numerical	σ_θ numerical	σ_r analytical	σ_θ analytical
1	-7.257646E-01	1.430465E+00	-7.950518E-01	1.461718E+00
2	-3.703436E-01	1.069144E+00	-4.224014E-01	1.089068E+00
3	-1.662729E-01	8.627029E-01	-2.079381E-01	8.746048E-01
4	-3.838692E-02	7.337069E-01	-7.332268E-02	7.399893E-01
5	-7.257000E-01	1.430300E+00	-7.957709E-01	1.462438E+00
6	-3.703000E-01	1.069100E+00	-4.226027E-01	1.089269E+00
7	-1.663000E-01	8.627000E-01	-2.077483E-01	8.744150E-01
8	-3.840000E-02	7.338000E-01	-7.293267E-02	7.395993E-01
9	-7.257646E-01	1.430465E+00	-7.950518E-01	1.461718E+00
10	-3.703436E-01	1.069144E+00	-4.224014E-01	1.089068E+00
11	-1.662729E-01	8.627029E-01	-2.079381E-01	8.746048E-01
12	-3.838692E-02	7.337069E-01	-7.332268E-02	7.399893E-01

Table 1.10: Comparison of stress values for problem (e) with geometry treated as plane strain case.

is plotted in Fig 1.5.

1.5.2 Treatment as axi-symmetric case

The schematic of this case is given in Figure 1.6. Table 1.10 compares numerical and analytical values of σ_r and σ_θ for all the elements. As opposed to previous case, in this case, the agreement between numerical and analytical values is much better. However, since the analytical stresses vary as $1/r^2$, just 4 elements cannot possibly capture the variation very well. Analytical vs Numerical stresses is plotted in Fig 1.7.

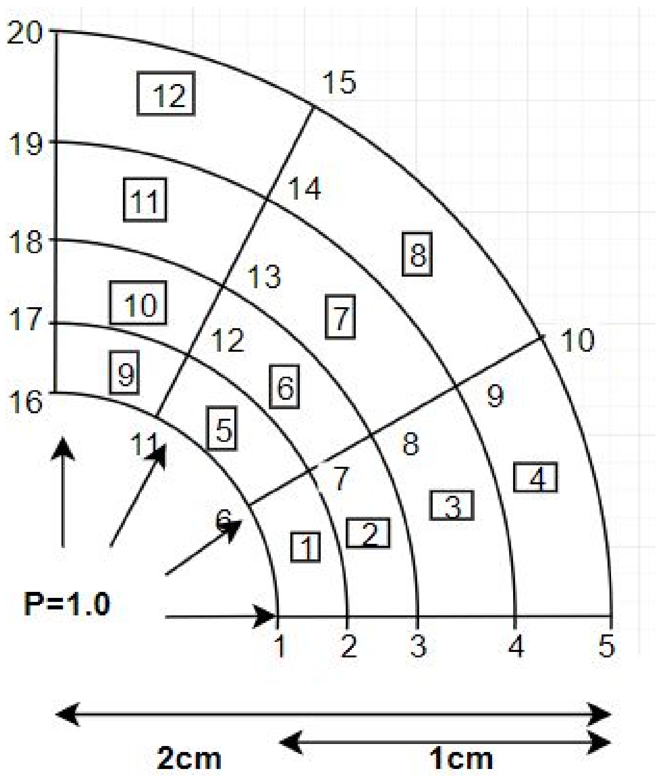


Figure 1.4: Mesh and element numbering for problem (e) with geometry treated as plane strain case.

Element #	σ_r numerical	σ_θ numerical	σ_r analytical	σ_θ analytical
1	-7.175000E-01	1.390000E+00	-7.073171E-01	1.373984E+00
2	-3.700000E-01	1.037000E+00	-3.661202E-01	1.032787E+00
3	-1.705000E-01	8.354000E-01	-1.686275E-01	8.352941E-01
4	-4.546000E-02	7.094000E-01	-4.424779E-02	7.109145E-01

Table 1.11: Comparision of stress values for problem (e) with geometry treated as axi–symmetric case.

1.6 Plate with hole

In this problem, a rectangular plate with a hole is subjected to tension as shown in figure 1.8. Two meshes have been generated for the analysis of this problem as shown in the figures 1.9 and 1.10. Theoretical solution for this case is given by,

$$\sigma_\theta = \frac{\sigma_o}{2}\left[1 + \frac{a^2}{r^2} - \left(1 + \frac{3a^4}{r^4}\right)cos2\theta\right]$$

where a is the radius of the hole. To obtain σ_{xx} along the y-axis, we substitute $\theta = \frac{\pi}{2}$, and for σ_{yy} along the x-axis, we substitute $\theta = 0$. Stress concentration factor K is given by,

$$K = \frac{\sigma_{max}}{\sigma_{avg}}$$

Here $\sigma_{avg} = \sigma_o * 5/(5-1) = 1.25$.
Stress concentration factor for the coarse mesh is 1.82 and for the fine mesh is 1.64.
Comparison of numerical and analytical results are shown in figures 1.11 through 1.14.

It is observed that the results of the coarse mesh are very different from the theoretical results, this can be attributed to the inability of the coarse mesh to model the hole region adequately. Results of the fine mesh sit well with the theoretical values. The fine mesh only fails to capture the rise and dip of the σyy values as shown in 1.14.

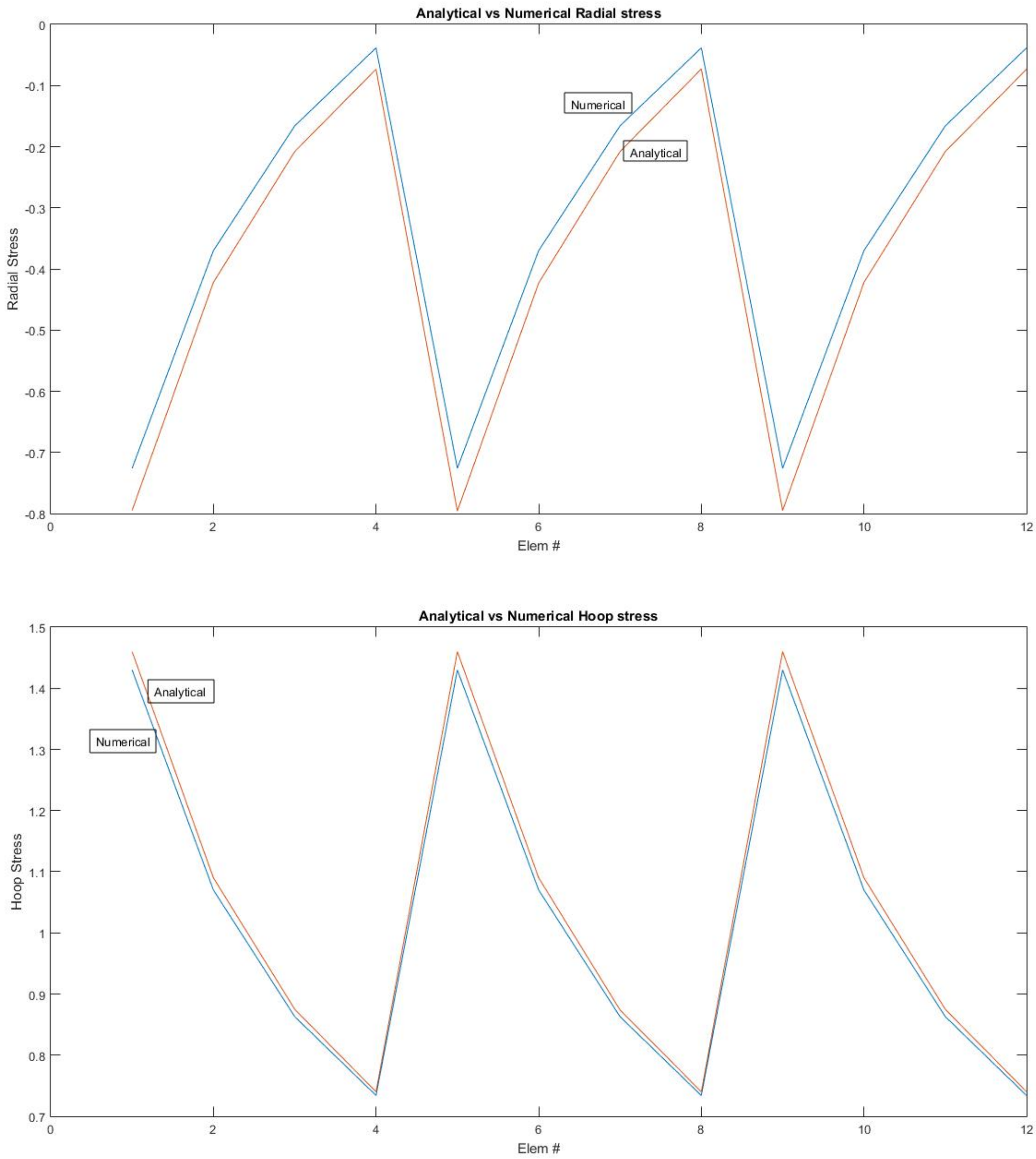


Figure 1.5: Analytical vs Numerical stresses for problem (e) treated as plane strain case

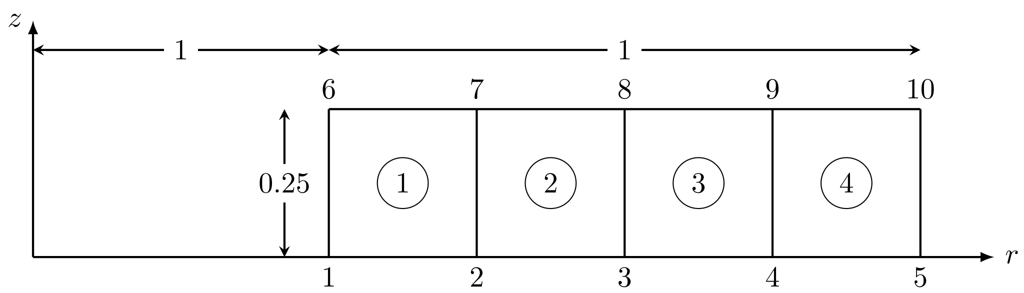
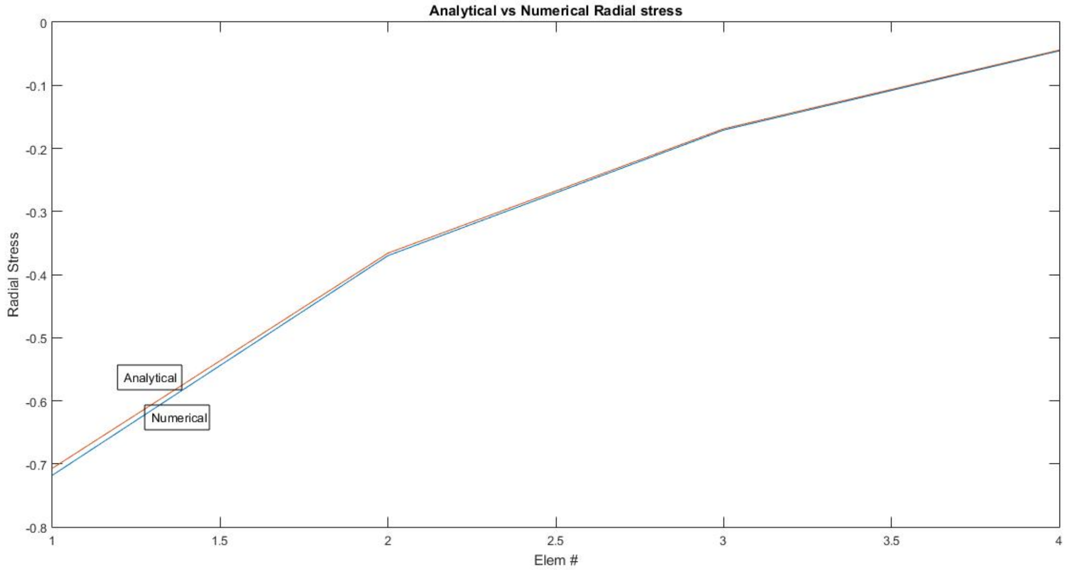


Figure 1.6: Mesh and element numbering for problem (e) with geometry treated as axi–symmetric case.



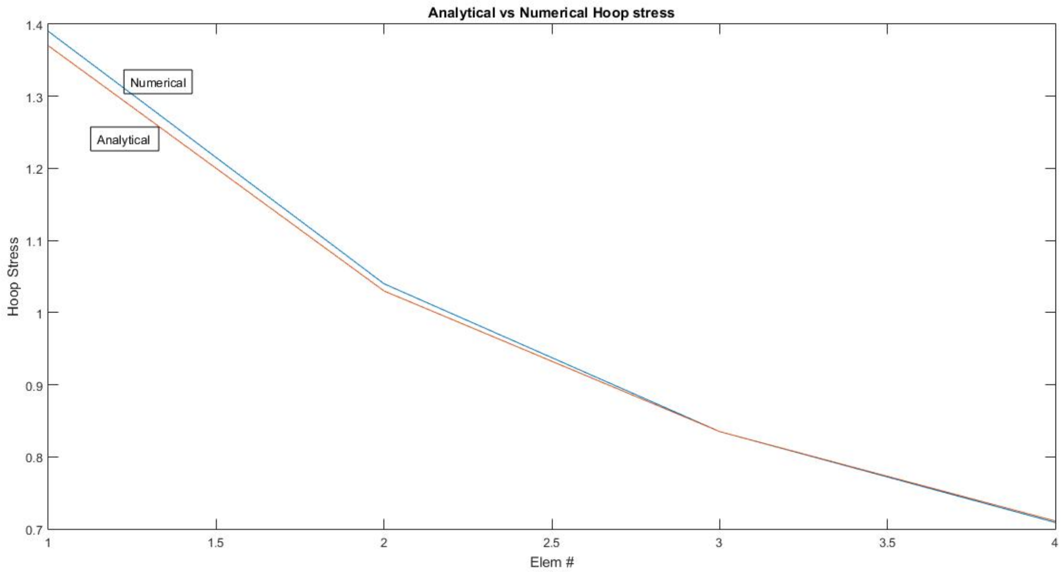


Figure 1.7: Analytical vs Numerical stresses for problem (e) treated as axisymmetric case

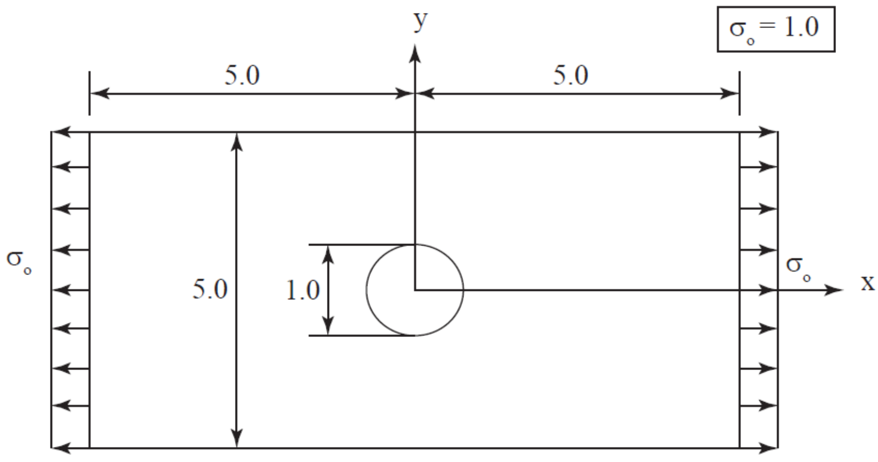


Figure 1.8: Schematic for problem (f)

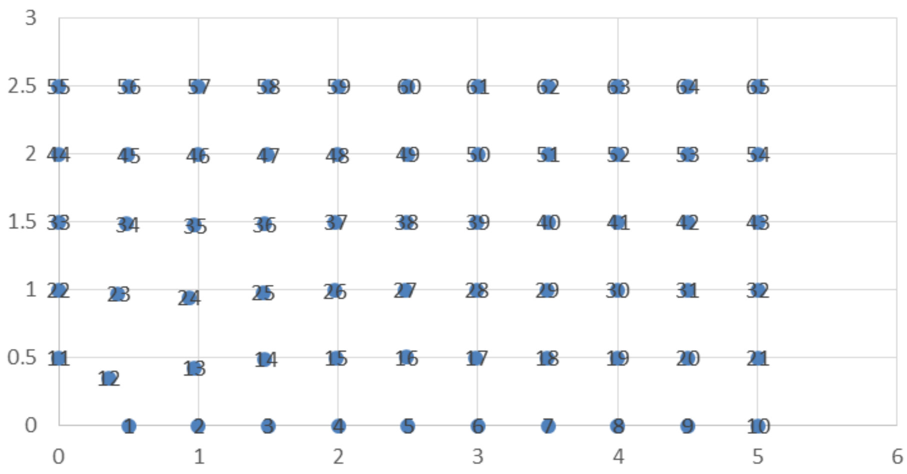


Figure 1.9: Coarse grid with numbered nodes for problem (f)

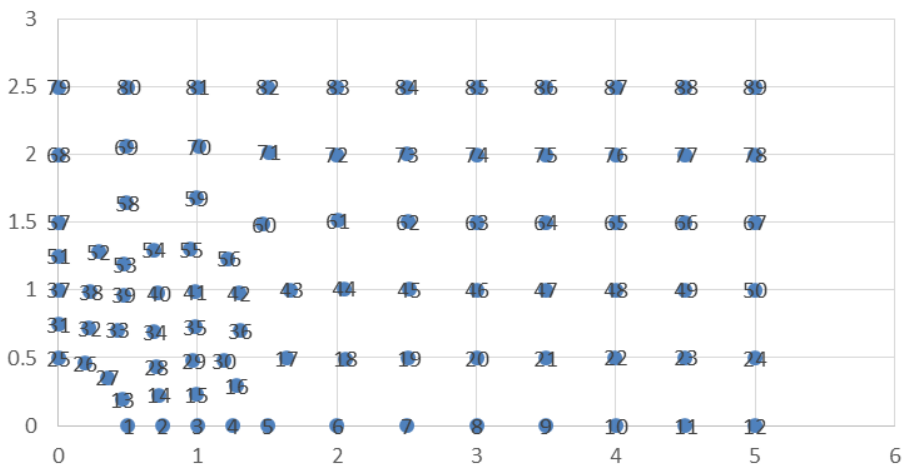


Figure 1.10: Fine grid with numbered nodes for problem (f)

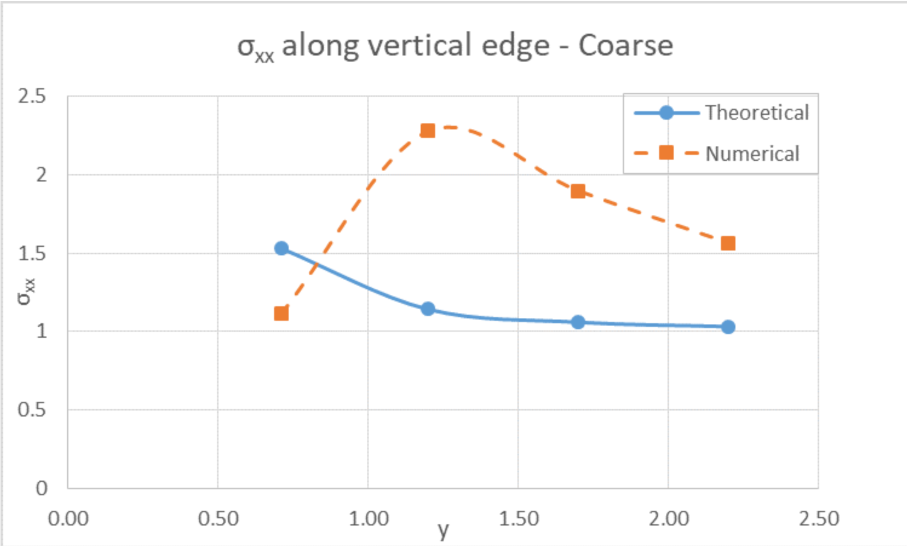


Figure 1.11: σ_{xx} - Coarse

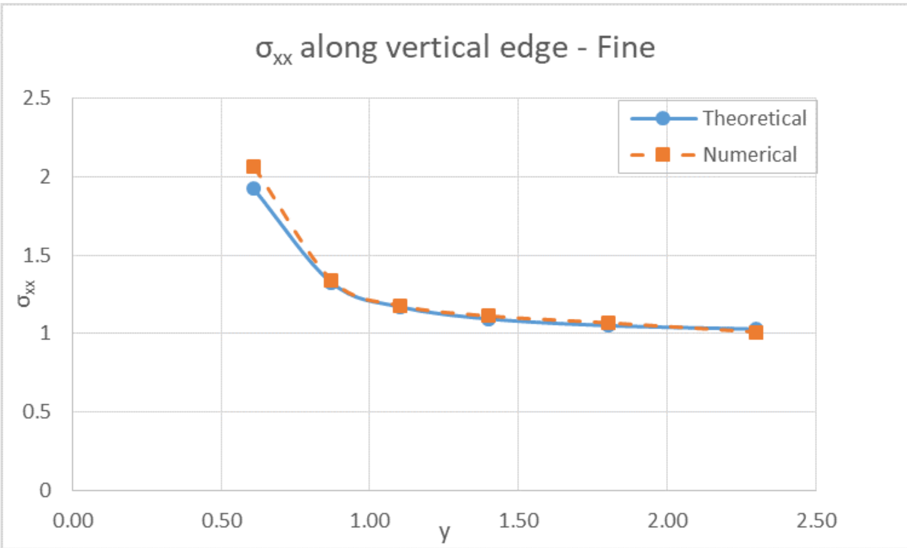


Figure 1.12: σ_{xx} - Fine

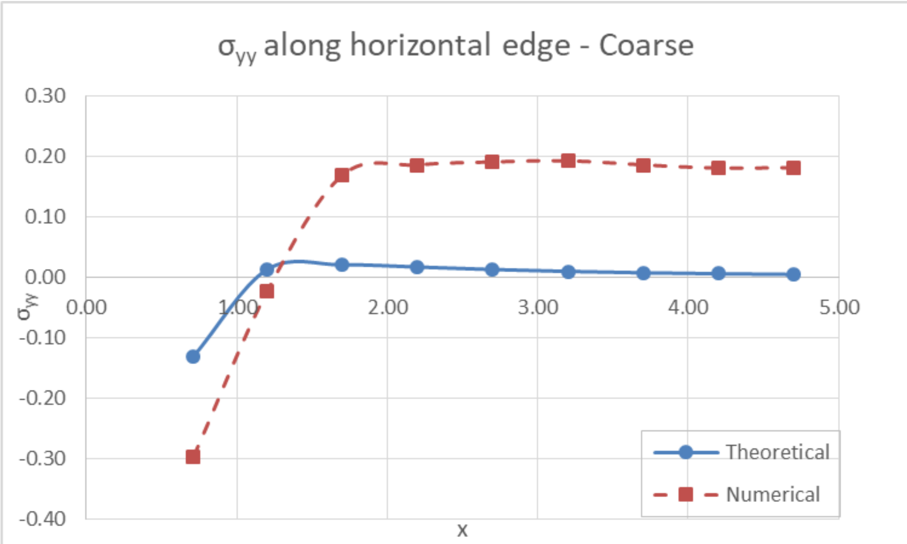


Figure 1.13: σ_{yy} - Coarse

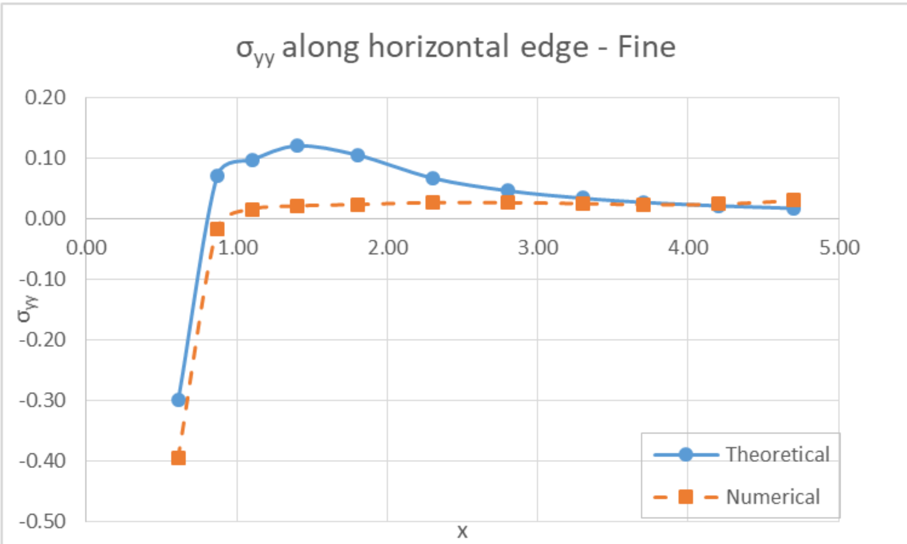


Figure 1.14: σ_{yy} - Fine

Part 2

Code and Input file listings

Listing 2.1: Code for Linear Elastic Element Routine ELMT01 ”mock.for”

```
C     -------------------------------------------------------------------------
C
C            APPENDIX 1: GENERAL STRUCTURE OF ELEMENT SUBROUTINE IN FEAP
C     -------------------------------------------------------------------------
C
C
      SUBROUTINE ELMT01(D,UL,XL,STRL,EPSL,QL,IX,TL,S,P,VEL,ACCEL,
     1 NDF,NDM,NST,NS,NQ,ISW)
C
C---- MOCK ELEMENT ROUTINE
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      CHARACTER*8 HEAD
      CHARACTER*1 O
      COMMON /CHEAD/ O,HEAD(20)
      COMMON /CDATA/ NUMNP,NUMEL,NUMMAT,NEN,NEQ,IPR
      COMMON /ELDATA/ DM,NELMT,MA,MCT,IEL,NEL
      DIMENSION D(*),UL(NDF,*),XL(NDM,*),IX(*),TL(*),S(NST,*),P(*)
      DIMENSION STRL(NS,*),EPSL(NS,*),QL(NQ,*),VEL(NDF,*),ACCEL(NDF,*)

c---- Variables required for ISW 1
      integer lineNo, maxLines, MODE
      character curLine*100, curStr*100

c---- Variables required for ISW 3
      integer k,row,col,L
      INTEGER NQP,NDM,NQUAD
      DIMENSION SG(16),TG(16),WG(16)
      DIMENSION B(4,NST),SHP(3,NEL),DMAT(4,4)
      DIMENSION PROD(NST,NST),TEMP(4,NST)

c---- Variables for ISW 4
      REAL XGAUS, YGAUS
      integer curNode

C     ...VARIABLES FOR ISW 6 (ADDITIONAL VARIABLES)...
      DIMENSION BTRANS(NST, 4), BTRSTR(NST)
      INTEGER I, J, CURR_NST

C     ...VARIABLES FOR ISW 9 (ADDITIONAL VARIABLES)...
      integer NSDM, T
      DIMENSION BUL(4), DMEP(4), ULARRAY(8)

      NSDM = 4
```

```
      NQUAD = 5
C     write(*,*)"In main program"
C     write(6,'("nquad = ",I5)')NQUAD

C
C---- DESCRIPTION OF THE PARAMETER LIST AND SOME OF COMMON BLOCK VARIABLES
C
C---- D(*)             = MATERIAL PROPERTY ARRAY
C---- UL(NDF,NEL)      = NODAL DISPLACEMENTS
C---- VEL(NDF,NEL)     = NODAL VELOCITIES
C---- ACCEL(NDF,NEL)   = NODAL ACCELERATIONS
C---- XL(NDM,NEL)      = NODAL COORDINATES
C---- TL(NEL)          = NODAL TEMPERATURES
C---- IX(NEL)          = CONNECTIVITY ARRAY
C---- STRL(NS,NQUAD)   = ELEMENT STRESSES AT INTEGRATION AND OUTPUT POINTS
C---- EPSL(NS,NQUAD)   = ELEMENT STRAINS AT INTEGRATION AND OUTPUT POINTS
C---- QL(NQ,NQUAD)     = INTERNAL VARIABLES AT INTEGRATION AND OUTPUT POINTS
C---- S(NST,NST)       = TANGENT STIFFNESS MATRIX (ISW = 3) OR
C----                    CONSISTENT MASS MATRIX (ISW = 5)
C---- P(NST)           = ELEMENT FORCE ARRAY (ISW = 6) OR
C----                    LUMPED MASS MATRIX (ISW = 5)
C---- HEAD(20)         = HEADER FOR PRINTOUTS
C---- THE FOLL VARIABLES HAVE BEEN DEFINED ELSEWHERE IN THE PROGRAM
C---- NDF              = NUMBER OF DEGREES OF FREEDOM PER NODE
C---- NDM              = NUMBER OF SPATIAL DIMENSIONS
C---- NST              = DIMENSION OF STIFFNESS ARRAY (NDF*NEN)
C---- NS               = DIMENSION OF STRESS OR STRAIN ARRAY (NSDM)
C---- NQ               = DIMENSION OF INTERNAL VARIABLE ARRAY (NQDM)
C---- NQUAD            = NUMBER OF INTEGRATION AND OUTPUT QUADRATURE POINTS
C---- NELMT            = ELEMENT NUMBER CORRESPONDING TO PRESENT ELEMENT
C---- NEL              = NUMBER OF NODES IN THE PRESENT ELEMENT
C---- NEN              = MAXIMUM NUMBER OF NODES PER ELEMENT IN THE MESH
C---- MA               = MATERIAL SET NUMBER TO WHICH THE PRESENT
C----                    ELEMENT BELONGS
C---- ISW              = SWITCH WHICH IS SET BY THE CALLING MACRO AND
C                        WHICH CONTROLS TO WHICH MODULE OF THE ELEMENT ROUTINE
C                        THE EXECUTION IS TO BE TRANSFERRED:
C                        ISW = 1 -> READ IN MATERIAL PROPERTIES
C                        ISW = 2 -> UNUSED ISW
C                        ISW = 3 -> COMPUTE ELEMENT STIFFNESS MATRIX
C                        ISW = 4 -> PRINTOUT ELEMENT VARIABLES (E.G. STRESSES)
C                        ISW = 5 -> COMPUTE MASS MATRIX
C                        ISW = 6 -> COMPUTE ELEMENT FORCE ARRAY
C                        ISW = 7 -> UNUSED ISW
C                        ISW = 8 -> UNUSED ISW
C                        ISW = 9 -> UPDATE ELEMENT VARIABLES
C
C---- OUTLINE OF THE SUBROUTINE
C
C---- GO TO CORRECT ARRAY PROCESSOR
      GO TO (1,2,3,4,5,6,7,8,9), ISW
C
1     CONTINUE
C
C---- INPUT MATERIAL PROPERTIES
C---- THIS MODULE IS ACCESSED WHEN MACRO 'MATE' IS EXECUTED
C
C---- TASKS TO BE PERFORMED IN THIS MODULE:
C
C---- 1) READ INTO ARRAY 'D' MATERIAL PROPERTIES FOR THE ELEMENT
C---- 2) PRINT OUT AN ECHO OF THE INPUT MATERIAL DATA
C
      maxLines = 100
      write(6,'("Entering ISW 1")')
```

```
c      open(UNIT=5,FILE="fpin.dat")
        read(5,'(A100)')curLine
        write(*,*)curLine
          curStr = curLine(1:10)
c         x body force
          read(curStr,*)D(1)
c         y body force
          curStr = curLine(11:20)
          read(curStr,*)D(2)
c         MODE
          curStr = curLine(21:30)
          read(curStr,*)D(3)
c         Mass density
          curStr = curLine(31:40)
          read(curStr,*)D(4)
c         Youngs modulus
          curStr = curLine(41:50)
          read(curStr,*)D(5)
c         Poissons ratio
          curStr = curLine(51:60)
          read(curStr,*)D(6)

      MODE = D(3)
      write(6,*)"Element Data read from file:"
      write(6,'("Element type",I4," | 1 ==> Plain strain | 2 ==>",
     +" Axi-symmetric | 3 ==> Plain stress")')MODE
      write(6,'("Mass density",F10.3)')D(4)
      write(6,'("Youngs modulus",F10.3)')D(5)
      write(6,'("Poissons ratio",F10.3)')D(6)
      write(6,*)"Leaving ISW 1"
      RETURN
C
2     CONTINUE
C---- UNUSED ISW
      RETURN
C
3     CONTINUE
C
C---- COMPUTE ELEMENT STIFFNESS MATRIX
C---- THIS MODULE IS ACCESSED WHEN MACROS 'TANG' AND 'UTAN' ARE EXECUTED
C
C---- TASKS TO BE PERFORMED IN THIS MODULE:
C
C---- 1) COMPUTE THE ELEMENT STIFFNESS MATRIX
C---- 2) RETURN IT IN ARRAY 'S(NST,NST)'
C
      write(6,'("Entering ISW 3")')
      MODE = D(3)
C     write(6,'("nquad = ",I5)')NQUAD
      if(NQUAD .eq. 5) then
        NQP = 2
      elseif(NQUAD .eq. 1) then
        NQP = 1
      endif
      CALL PGAUS2(NQP,LINT,SG,TG,WG)
C     write(6,'("Lint = ",I5)')LINT
      do row = 1,NST
          do col = 1,NST
            S(row,col) = 0.0
          enddo
      enddo
      do L=1,LINT
C       write(6,'("\tEntering ISW 3 loop")')
        CALL SHAP2D(SG(L),TG(L),XL,SHP,XSJ,NDM,NEL,IX,.TRUE.)
```

```
C       write(6,'("XSJ just after shap2d = ",E20.2)')XSJ
        do row=1,3
C         write(6,'("SHP(",I1,",1:4) just after shap2d",4E16.4)')row,
     +    SHP(row,1),SHP(row,2),SHP(row,3),SHP(row,4)
        enddo
        CALL BMAT(B,SHP,XL,XJC,NEL,NDM,NST,MODE)
        CALL ELAS(D,DMAT,MODE)
C       write(6,'("SG,TG,detJ,XJC,weight at current GP:",5E12.2)')
     +  SG(L),TG(L),XSJ,XJC,WG(L)
        do row=1,4
C         write(6,'("B(",I1,",1:8) in ISW3 = ",8E12.4)')row,B(row,1),
     +    B(row,2),B(row,3),B(row,4),B(row,5),B(row,6),B(row,7),B(row,8)
        enddo
        do row=1,4
C         write(6,'("DMAT(",I1,",:) in ISW3 = ",4F12.1)')row,
     +    DMAT(row,1),DMAT(row,2),DMAT(row,3),DMAT(row,4)
        enddo
c------ compute D*B and save it as TEMP
        do row = 1,4
          do col = 1,NST
            sum = 0.0
            do k = 1,4
              sum = sum + DMAT(row,k)*B(k,col)
            enddo
            TEMP(row,col) = sum
          enddo
        enddo
        do row=1,4
C         write(6,'("TEMP(",I1,",1:8) in ISW3 = ",8E20.4)')row,
C    +    TEMP(row,1),TEMP(row,2),TEMP(row,3),TEMP(row,4),TEMP(row,5),
C    +    TEMP(row,6),TEMP(row,7),TEMP(row,8)
        enddo
c------ compute BTRANS*TEMP and save it as PROD
        do row = 1,NST
          do col = 1,NST
          sum = 0
            do k = 1,4
              sum = sum + B(k,row)*TEMP(k,col)
            enddo
            PROD(row,col) = sum
          enddo
        enddo
        do row=1,8
C         write(6,'("PROD(",I1,",1:8) in ISW3 = ",8E20.4)')row,
C    +    PROD(row,1),PROD(row,2),PROD(row,3),PROD(row,4),PROD(row,5),
C    +    PROD(row,6),PROD(row,7),PROD(row,8)
        enddo
c------ add contribution of current gauss point to S
        do row = 1,NST
          do col = 1,NST
            S(row,col) = S(row,col) + XSJ*XJC*WG(L)*PROD(row,col)
          enddo
        enddo
      enddo
      do row=1,NST
C       write(6,'("S(",I1,",1:8) = ",8E20.6)')row,S(row,1),S(row,2),
     +  S(row,3),S(row,4),S(row,5),S(row,6),S(row,7),S(row,8)
      enddo
      write(6,*)"Leaving ISW 3"
      RETURN
4     CONTINUE
C
C---- OUTPUT ELEMENT VARIABLES SUCH AS STRESSES AND STRAINS
C---- THIS MODULE IS ACCESSED WHEN MACRO 'STRE' IS EXECUTED
```

```
C
C---- TASKS TO BE CARRIED OUT IN THIS MODULE:
C
C---- 1) PRINTOURT ELEMENT VARIABLES SUCH AS STRESSES, STRAINS AND
C----    INTERNAL VARIABLES AT OUTPUT QUADRATURE POINTS
C
C
      write(6,'("Entering ISW 4")')
      CALL SHAP2D(0.0,0.0,XL,SHP,XSJ,NDM,NEL,IX,.FALSE.)
      XGAUS = 0.0
      YGAUS = 0.0
      do curNode = 1,NEL
        XGAUS = XGAUS + SHP(3,curNode)*XL(1,curNode)
        YGAUS = YGAUS + SHP(3,curNode)*XL(2,curNode)
      enddo
c---- Writing data to file: el#, x, y, 4 stress components, 4 strain components
      write(6,'(I4,",",E12.2,",",E12.2)')NELMT,XGAUS,YGAUS
      write(6,'(E12.2,",",E12.2,",",E12.2,",",E12.2)')STRL(1,NQUAD),
     +STRL(2,NQUAD),STRL(3,NQUAD),STRL(4,NQUAD)
      write(6,'(E12.2,",",E12.2,",",E12.2,",",E12.2)')EPSL(1,NQUAD),
     +EPSL(2,NQUAD),EPSL(3,NQUAD),EPSL(4,NQUAD)
      write(6,*)"Leaving ISW 4"
      RETURN
C
5     CONTINUE
C
C---- COMPUTE THE CONSISTENT AND LUMPED MASS MATRICES
C---- THIS MODULE IS ACCESSESD WHEN MACROS 'LMAS' AND 'CMAS' ARE EXECUTED
C
C---- TASKS TO BE CARRIED OUT IN THIS MODULE:
C
C---- 1) COMPUTE CONSISTENT AND LUMPED ELEMENT MASS MATRICES
C---- 2) RETURN THEM IN ARRAYS 'S(NST,NST)' AND 'P(NST)', RESPECTIVELY
C
      RETURN
C
6     CONTINUE

      write(6,'("Entering ISW 6")')
      MODE = D(3)
C     ...CHECKING THE QUADRATURE POINTS...
      IF(NQUAD .EQ. 5) THEN
        NQP = 2
      ELSEIF(NQUAD .EQ. 1) THEN
        NQP = 1
      ENDIF
C     ...GET QUADRATURE POINTS AND THEIR WEIGHTS...
      CALL PGAUS2(NQP,LINT,SG,TG,WG)
C     ...LOOP OVER INTEGRATION POINTS FROM 1 TO LINT...
      DO L=1, LINT
C     ...SHAPE FUNCTIONS AND THEIR X & Y DERIVATIVES AT CURRENT GAUSS POINT...
        CALL SHAP2D(SG(L),TG(L),XL,SHP,XSJ,NDM,NEL,IX,.TRUE.)
C     ...B MATRIX FORMATION...
        CALL BMAT(B,SHP,XL,XJC,NEL,NDM,NST,MODE)
C     ...ELEMENT FORCE ARRAY CONTRIBUTION DUE TO BODY FORCES CORRESPONDING TO GAUSS POINT L
    ...
        CURR_NST = 1
        DO I=1, NEL
                DO J=1, 2
                        P(CURR_NST) = P(CURR_NST) + XSJ*XJC*WG(L)*SHP(3, I)*D(J)
                        CURR_NST = CURR_NST + 1
                ENDDO
        ENDDO
C     ...COMPUTING BTRANS FROM B...
```

```
        DO I=1, 4
                DO J=1, NST
                BTRANS(J, I) = B(I, J)
                ENDDO
        ENDDO
C     ...SUBTRACTING INTERNAL FORCES DUE TO ELEMENT STRESSES AT GAUSS POINT L...
C     ...COMPUTING BTRANS*STRL(*,L)...
        DO J=1, NST
            SUM1 = 0
                DO I=1, 4
                        SUM1 = SUM1 + BTRANS(J, I)*STRL(I, L)
                ENDDO
                BTRSTR(J) = SUM1
        ENDDO
        DO J=1, NST
                P(J) = P(J) - XSJ*XJC*WG(L)*BTRSTR(J)
        ENDDO
      ENDDO
      do I=1,NST
        write(6,'("P(",I2,") = ",F12.2)')I,P(I)
      enddo
      write(6,*)"Leaving ISW 6"
      RETURN
C
C---- COMPUTE ELEMENT FORCE ARRAY
C---- THIS MODULE IS ACCESSED WHEN MACRO 'FORM' IS EXECUTED
C
C---- TASKS TO BE PERFORMED IN THIS MODULE:
C
C---- 1) COMPUTE NODAL FORCES ARISING FROM EXTERNALLY APPLIED DISTRIBUTED
C----    LOADS (GRAVITY LOADS, HYDROSTATIC PRESSURES, THERMAL LOADS, ETC.)
C---- 2) COMPUTE INTERNAL FORCES ARISING FROM (INITIAL) ELEMENT STRESS FIELD
C---- 3) SUBTRACT THE LATTER FROM THE FORMER. THIS IS THE ELEMENT FORCE
C----    VECTOR. IN NON-LINEAR PROBLEMS, THIS IS CALLED AS OUT-OF-BALANCE
C----    FORCES.
C---- 4) RETURN THE ELEMENT FORCES IN ARRAY 'P(NST)'
C
C---- REMARKS:
C
C---- 1) THE INTERNAL FORCES DUE TO ELEMENT STRESSES IS GIVEN BY
C----    INTEGRAL OF 'BTRANS*STRL' OVER THE VOLUME OF THE ELEMENT.
C        HERE 'BTRANS' DENOTES TRANSPOSE OF STRAIN-DISPLACEMENT
C        MATRIX 'B' AND 'STRL' DENOTES ELEMENT STRESS ARRAY.
C
      RETURN
C
7     CONTINUE
C---- UNUSED ISW
      RETURN
C
8     CONTINUE
C---- UNUSED ISW
      RETURN
C
9     CONTINUE
      write(6,'("Entering ISW 9")')
      MODE = D(3)
      LL = 0
      DO KKK = 1,2
        IF(KKK.EQ.1) THEN
                CALL PGAUS2(2,LINT,SG,TG,WG)
        ELSEIF(KKK.EQ.2) THEN
                CALL PGAUS2(1,LINT,SG,TG,WG)
        ENDIF
```

```
        DO COL=1,NEL
            ULARRAY(2*COL-1)=UL(1,COL)
            ULARRAY(2*COL)=UL(2,COL)
            write(6,'("ULARRAY() = ",2F12.2)')UL(1,COL),UL(2,COL)
        ENDDO

        DO L=1,LINT
                LL = LL + 1
            write(6,'("LL = ",I1)')LL
                CALL SHAP2D(SG(L),TG(L),XL,SHP,XSJ,NDM,NEL,IX,.TRUE.)
                CALL BMAT(B,SHP,XL,XJC,NEL,NDM,NST,MODE)
                CALL ELAS(D,DMAT,MODE)
C     ...COMPUTING THE PRODUCT B*UL AND STORING IN BUL...
                        DO I=1, NSDM
                          BUL(I)=0
                              DO J=1,2*NEL
                                BUL(I)=BUL(I)+B(I,J)*ULARRAY(J)
                              ENDDO
                        ENDDO
         write(6,'("BUL(:) = ",4F20.6)')BUL(1),BUL(2),BUL(3),BUL(4)
C     ...UPDATING EPSL(*, LL) VALUES...
                        DO I=1, NSDM
                                 EPSL(I, LL) = EPSL(I, LL) + BUL(I)
                        ENDDO
C         ...COMPUTING DMAT*EPSL(*, LL) AND STORING IN DMEP...
                        DO I=1, NSDM
                 DMEP(I) = 0
                                 DO J=1, NSDM
                                          DMEP(I) = DMEP(I) + DMAT(I, J)*EPSL(J, LL)
                                 ENDDO
                        ENDDO
C     ...COMPUTING STRESSES AND STORING IN STRL...
          DO I=1, NSDM
            STRL(I, LL) = STRL(I, LL) + DMEP(I)
          ENDDO
        ENDDO
      ENDDO
      write(6,*)"Leaving ISW 9"
      RETURN
C
C---- UPDATE ELEMENT VARIABLES AT INTEGRATION AND OUTPUT POINTS.
C---- THIS MODULE IS ACCESSED WHEN MACRO 'CEQS' IS EXECUTED.
C
C---- TASKS TO BE PERFORMED IN THIS MODULE:
C
C---- 1) UPDATE ELEMENT STRESSES, STRAINS AND INTERNAL VARIABLES
C---- 2) RETURN UPDATED QUANTITIES IN 'STRL', 'EPSL' AND 'QL', RESPECTIVELY.
      RETURN
C
C---- FORMAT STATEMENTS:
C
C---- extra return statement added as per sir's mail
      RETURN
      END

      SUBROUTINE BMAT(B,SHP,XL,XJC,NEL,NDM,NST,MODE)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      dimension B(4,NST),XL(NDM,NEL),SHP(3,NEL)
      integer row,col,curNode,NEL,NDM,NST
c---- Initialise B, XJC
c---- note: NST=NDM*NEN. if NEN>NEL, then coloumns with col>NDM*NEL in B remain 0
      do row = 1,4
        do col = 1,NST
```

```
          B(row,col) = 0.0
        enddo
      enddo
      XJC = 0.0
      if(MODE .eq. 1 .or. MODE .eq. 3) then
c------ 1: Plane strain, 3: Plane stress
        XJC = 1.0
        write(6,'("XJC = ",F4.2)')XJC
        do curNode = 1,NEL
          B(1,2*curNode-1) = 0.0
          B(1,2*curNode) = 0.0
          B(2,2*curNode-1) = SHP(1,curNode)
          B(2,2*curNode) = 0.0
          B(3,2*curNode-1) = 0.0
          B(3,2*curNode) = SHP(2,curNode)
          B(4,2*curNode-1) = SHP(2,curNode)
          B(4,2*curNode) = SHP(1,curNode)
        enddo
      elseif(MODE .eq. 2) then
c------ Axi-symmetric
        r = 0.0
        do curNode = 1,NEL
          r = r+SHP(3,curNode)*XL(1,curNode)
        enddo
        XJC = 2*3.141592654*r
        do curNode = 1,NEL
          B(1,2*curNode-1) = SHP(1,curNode)
          B(1,2*curNode) = 0.0
          B(2,2*curNode-1) = 0.0
          B(2,2*curNode) = SHP(2,curNode)
          B(3,2*curNode-1) = SHP(3,curNode)/r
          B(3,2*curNode) = 0.0
          B(4,2*curNode-1) = SHP(2,curNode)
          B(4,2*curNode) = SHP(1,curNode)
        enddo
      endif
      do row=1,4
        write(6,'("B(",I1,",1:8) = ",8E12.4)')row,B(row,1),B(row,2),
     +  B(row,3),B(row,4),B(row,5),B(row,6),B(row,7),B(row,8)
      enddo
      END

      SUBROUTINE ELAS(D,DMAT,MODE)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      dimension D(*), DMAT(4,4)
      real E, nu
      integer row,col,MODE
C---- Fill the upper triangle and then use symmetry
      E = D(5)
      nu = D(6)
      write(6,'("E = ",F12.2)')D(5)
      if(MODE .eq. 1) then
C------ Plane strain
        DMAT(1,1) = 0.0
        DMAT(1,2) = 0.0
        DMAT(1,3) = 0.0
        DMAT(1,4) = 0.0
        DMAT(2,2) = E*(1-nu)/((1+nu)*(1-2*nu))
        DMAT(2,3) = E*nu/((1+nu)*(1-2*nu))
        DMAT(2,4) = 0.0
        DMAT(3,3) = E*(1-nu)/((1+nu)*(1-2*nu))
        DMAT(3,4) = 0.0
        DMAT(4,4) = E/(2*(1+nu))
        do row=2,4
```

```
          do col=1,row-1
            DMAT(row,col) = DMAT(col,row)
          enddo
        enddo

      elseif(MODE .eq. 2) then
C------ Axi-symmetric
        DMAT(1,1) = E*(1-nu)/((1+nu)*(1-2*nu))
        DMAT(1,2) = E*nu/((1+nu)*(1-2*nu))
        DMAT(1,3) = DMAT(1,2)
        DMAT(1,4) = 0.0
        DMAT(2,2) = DMAT(1,1)
        DMAT(2,3) = DMAT(1,2)
        DMAT(2,4) = 0.0
        DMAT(3,3) = DMAT(1,1)
        DMAT(3,4) = 0.0
        DMAT(4,4) = E/(2*(1+nu))
        do row=2,4
          do col=1,row-1
            DMAT(row,col) = DMAT(col,row)
          enddo
        enddo

      elseif(MODE .eq. 3) then
C------ Plane stress
        DMAT(1,1) = 0.0
        DMAT(1,2) = 0.0
        DMAT(1,3) = 0.0
        DMAT(1,4) = 0.0
        DMAT(2,2) = E/(1-nu*nu)
        DMAT(2,3) = E*nu/(1-nu*nu)
        DMAT(2,4) = 0.0
        DMAT(3,3) = E/(1-nu*nu)
        DMAT(3,4) = 0.0
        DMAT(4,4) = E/(2*(1+nu))
        do row=2,4
          do col=1,row-1
            DMAT(row,col) = DMAT(col,row)
          enddo
        enddo

      else
C------ Invalid mode
        do row=1,4
          do col=1,4
            write(6,*)"Invalid mode"
            DMAT(row,col) = 0.0
          enddo
        enddo
      endif
      do row=1,4
        write(6,'("D(",I1,",:) = ",4F12.1)')row,DMAT(row,1),DMAT(row,2),
     +   DMAT(row,3),DMAT(row,4)
      enddo
      END
```

Listing 2.2: Input file for problem (a)

```
FEAP   ****** PATCH TEST (Plane stress)
    9    4    1    2    2    4    4    1    5    0    0    0
COOR
    1    0    0.0000    2.0000
    2    0    1.0000    2.0000
    3    0    2.0000    2.0000
    4    0    0.0000    1.0000
    5    0    1.2000    1.3000
    6    0    2.0000    1.0000
    7    0    0.0000    0.0000
    8    0    1.0000    0.0000
    9    0    2.0000    0.0000

ELEM
    1    1    1    4    5    2    0
    2    1    2    5    6    3    0
    3    1    4    7    8    5    0
    4    1    5    8    9    6    0

MATE
    1    1
       0.0       0.0       3.0       0.0  100.0000    0.3000

BOUN
    1    0    0    0
    2    0    0    0
    3    0    0    0
    7    0    1    1
    8    0    0    1
    9    0    0    1

FORC
    1    0    1    1
    0.0000       0.5
    2    0    1    1
    0.0000       1.0
    3    0    1    1
    0.0000       0.5


END
MACR
PRIN
TANG
FORM
SOLV
CEQS
DISP
STRE
END
STOP
```

Listing 2.3: Input file for problem (b)

```
FEAP  *** FOUR ELEMENT PATCH ***
    9    4    1    2    2    4    4    1    5    0    0    0

COOR
    1    1       0.0       2.0
    3    0       2.0       2.0
    4    0       0.0       1.0
    5    0       1.2       1.3
    6    0       2.0       1.0
    7    1       0.0       0.0
    9    0       2.0       0.0


ELEM
    1    1    1    4    5    2    1
    3    1    4    7    8    5    1


BOUN
    1    0    0   -1
    2    0    0   -1
    3    0    0   -1
    7    0   -1   -1
    8    0   -1   -1
    9    0   -1   -1


FORC
    1    0    1    1
       0.5       0.0
    2    0    1    1
       1.0       0.0
    3    0    1    1
       0.5       0.0
    4    0    1    1
       0.0      -1.0
    6    0    1    1
       0.0       1.0


MATE
    1    1
       0.0       0.0       3.0       0.0  10000000    0.3000


END
MACR
PRIN
TANG
FORM
SOLV
CEQS
DISP
STRE
END
STOP
```

Listing 2.4: Input file for problem (c)

```
FEAP   ****** PATCH TEST (Axi symmetric)
    9    4    1    2    2    4    4    1    5    0    0    0
COOR
    1    0    0.0000    2.0000
    2    0    1.0000    2.0000
    3    0    2.0000    2.0000
    4    0    0.0000    1.0000
    5    0    1.2000    1.3000
    6    0    2.0000    1.0000
    7    0    0.0000    0.0000
    8    0    1.0000    0.0000
    9    0    2.0000    0.0000

ELEM
    1    1    1    4    5    2    0
    2    1    2    5    6    3    0
    3    1    4    7    8    5    0
    4    1    5    8    9    6    0

MATE
    1    1
       0.0       0.0       2.0       0.0  100.0000    0.3000

BOUN
    1    0    0    0
    2    0    0    0
    3    0    0    0
    7    0    1    1
    8    0    0    1
    9    0    0    1

FORC
    1    0    1    1
    0.0000 1.0471975
    2    0    1    1
    0.0000 6.2831853
    3    0    1    1
    0.0000 5.2359877


END
MACR
PRIN
TANG
FORM
SOLV
CEQS
DISP
STRE
END
STOP
```

Listing 2.5: Input file for problem (d)

```
FEAP  *** FOUR ELEMENT PATCH ***
    9    4    1    2    2    4    4    1    5    0    0    0

COOR
    1    1       0.0       2.0
    3    0       2.0       2.0
    4    0       0.0       1.0
    5    0       1.2       1.3
    6    0       2.0       1.0
    7    1       0.0       0.0
    9    0       2.0       0.0


ELEM
    1    1    1    4    5    2    1
    3    1    4    7    8    5    1


BOUN
    1    0    0   -1
    2    0    0   -1
    3    0    0   -1
    7    0   -1   -1
    8    0   -1   -1
    9    0   -1   -1


FORC
    1    0    1    1
       0.5       0.0
    2    0    1    1
       1.0       0.0
    3    0    1    1
       0.5       0.0
    4    0    1    1
       0.0      -1.0
    6    0    1    1
       0.0       1.0


MATE
    1    1
       0.0       0.0       1.0       0.0  10000000    0.3000


END
MACR
PRIN
TANG
FORM
SOLV
CEQS
DISP
STRE
END
STOP
```

Listing 2.6: Input file for problem (e) with geometry treated as plane strain case

```
FEAP  *** THICK CYLINDER SUBJECTED TO INTERNAL PRESSURE - PLANE STRAIN
   20   12    1    2    2    4    4    1    5    0    0    0

COOR
    1    1       1.0       0.0
    5    0       2.0       0.0
    6    1     0.866       0.5
   10    0     1.732       1.0
   11    1       0.5     0.866
   15    0       1.0     1.732
   16    1       0.0       1.0
   20    0       0.0       2.0

ELEM
    1    1    1    2    7    6    1
    5    1    6    7   12   11    1
    9    1   11   12   17   16    1


BOUN
    1    1    0   -1
    5    0    0    1
   16    1   -1    0
   20    0    1    0


FORC
    1    0    1    1
 0.2617994       0.0
    6    0    1    1
 0.4534498 0.2617994
   11    0    1    1
 0.2617994 0.4534498
   16    0    1    1
       0.0 0.2617994



MATE
    1    1
       0.0       0.0       1.0       0.0  100.0000    0.3000


END
MACR
PRIN
TANG
FORM
SOLV
CEQS
DISP
STRE
END
STOP
```

Listing 2.7: Input file for problem (e) with geometry treated as axi–symmetric case

```
FEAP  *** THICK CYLINDER AS AXI-SYMMETRIC***
   10    4    1    2    2    4    4    1    5    0    0    0

COOR
    1    1       1.0       0.0
    5    0       2.0       0.0
    6    1       1.0      0.25
   10    0       2.0      0.25


ELEM
    1    1    1    2    7    6    1


BOUN
    1    1    0   -1
   10    1    0    1


FORC
    1    0    1    1
 0.7853982       0.0
    6    0    1    1
 0.7853982       0.0


MATE
    1    1
       0.0       0.0       2.0       0.0  100.0000    0.3000


END
MACR
PRIN
TANG
FORM
SOLV
CEQS
DISP
STRE
END
STOP
```

Listing 2.8: Input file for problem (f) - Coarse

```
FEAP  *** Plate with hole - Coarse***
   65   49    1    2    2    4    3    1    5    0    0    0

COOR
    1    0  0.500000  0.000000
    2    0  1.000000  0.000000
    3    0  1.500000  0.000000
    4    0  2.000000  0.000000
    5    0  2.500000  0.000000
    6    0  3.000000  0.000000
    7    0  3.500000  0.000000
    8    0  4.000000  0.000000
    9    0  4.500000  0.000000
   10    0  5.000000  0.000000
   11    0  0.000000  0.500000
   12    0  0.353553  0.353553
   13    0  0.969517  0.427911
   14    0  1.475639  0.492114
   15    0  1.982678  0.499991
   16    0  2.483895  0.504510
   17    0  2.986052  0.502848
   18    0  3.493025  0.501447
   19    0  3.996513  0.500724
   20    0  4.498257  0.500360
   21    0  5.000000  0.500000
   22    0  0.000000  1.000000
   23    0  0.425126  0.973113
   24    0  0.933994  0.943655
   25    0  1.461539  0.981559
   26    0  1.973347  0.998135
   27    0  2.477838  1.001766
   28    0  2.988937  1.001305
   29    0  3.495421  1.000951
   30    0  3.997824  1.000455
   31    0  4.499130  1.000181
   32    0  5.000000  1.000000
   33    0  0.000000  1.500000
   34    0  0.486309  1.487981
   35    0  0.972938  1.477409
   36    0  1.470738  1.486220
   37    0  1.979030  1.500054
   38    0  2.486340  1.499403
   39    0  2.993376  1.498603
   40    0  3.499842  1.500033
   41    0  3.999729  1.500059
   42    0  4.499567  1.500091
   43    0  5.000000  1.500000
   44    0  0.000000  2.000000
   45    0  0.497257  1.994391
   46    0  0.993585  1.992857
   47    0  1.493525  1.995563
   48    0  1.992216  1.997280
   49    0  2.494690  1.998794
   50    0  2.999981  1.999997
   51    0  3.499947  2.000012
   52    0  3.999892  2.000024
   53    0  4.499785  2.000046
   54    0  5.000000  2.000000
   55    0  0.000000  2.500000
   56    0  0.500000  2.500000
   57    0  1.000000  2.500000
   58    0  1.500000  2.500000
   59    0  2.000000  2.500000
```

```
   60    0  2.500000  2.500000
   61    0  3.000000  2.500000
   62    0  3.500000  2.500000
   63    0  4.000000  2.500000
   64    0  4.500000  2.500000
   65    0  5.000000  2.500000


ELEM
    1    1    1    2   13   12    1
   10    1   11   12   23   22    1
   20    1   22   23   34   33    1
   30    1   33   34   45   44    1
   40    1   44   45   56   55    1


MATE
    1    1
       0.0       0.0       3.0       0.0  10000000    0.3000


BOUN
    1    1    0   -1
   10    0    0    1
   11    0    1    0
   22    0    1    0
   33    0    1    0
   44    0    1    0
   55    0    1    0


FORC
   10    0    1    1
      0.25       0.0
   21    0    1    1
       0.5       0.0
   32    0    1    1
       0.5       0.0
   43    1    1    1
       0.5       0.0
   54    0    1    1
       0.5       0.0
   55    1    1    1
       0.5       0.0
   65    0    1    1
      0.25       0.0


END
MACR
PRIN
TANG
FORM
SOLV
CEQS
DISP
STRE
END
STOP
```

Listing 2.9: Input file for problem (f) - Fine

```
FEAP  *** Plate with hole - Fine***
   89   70    1    2    2    4    3    1    5    0    0    0

COOR
    1    0  0.500000  0.000000
    2    0  0.750000  0.000000
    3    0  1.000000  0.000000
    4    0  1.250000  0.000000
    5    0  1.500000  0.000000
    6    0  2.000000  0.000000
    7    0  2.500000  0.000000
    8    0  3.000000  0.000000
    9    0  3.500000  0.000000
   10    0  4.000000  0.000000
   11    0  4.500000  0.000000
   12    0  5.000000  0.000000
   13    0  0.461940  0.191342
   14    0  0.722026  0.223746
   15    0  0.989575  0.231515
   16    0  1.279869  0.292371
   17    0  1.635694  0.497291
   18    0  2.059770  0.492550
   19    0  2.513851  0.498609
   20    0  3.003334  0.499840
   21    0  3.500200  0.500121
   22    0  3.996513  0.500724
   23    0  4.498257  0.500360
   24    0  5.000000  0.500000
   25    0  0.000000  0.500000
   26    0  0.191342  0.461940
   27    0  0.353553  0.353553
   28    0  0.704391  0.430204
   29    0  0.968610  0.479798
   30    0  1.186898  0.476471
   31    0  0.000000  0.750000
   32    0  0.220612  0.721693
   33    0  0.423019  0.705018
   34    0  0.688058  0.690986
   35    0  0.977971  0.726355
   36    0  1.306249  0.699762
   37    0  0.000000  1.000000
   38    0  0.229863  0.990381
   39    0  0.467237  0.965217
   40    0  0.716462  0.980904
   41    0  0.979494  0.986566
   42    0  1.297777  0.977748
   43    0  1.667821  1.000929
   44    0  2.053472  1.010766
   45    0  2.519875  1.005809
   46    0  3.004963  1.002226
   47    0  3.495421  1.000951
   48    0  3.997824  1.000455
   49    0  4.499130  1.000181
   50    0  5.000000  1.000000
   51    0  0.000000  1.250000
   52    0  0.289588  1.283227
   53    0  0.472851  1.192277
   54    0  0.684752  1.298923
   55    0  0.948894  1.302008
   56    0  1.218297  1.231655
   57    0  0.000000  1.500000
   58    0  0.491749  1.643283
   59    0  0.992143  1.678981
```

```
   60    0  1.470738  1.486220
   61    0  2.008185  1.512968
   62    0  2.506467  1.505642
   63    0  3.002497  1.502289
   64    0  3.499842  1.500033
   65    0  3.999729  1.500059
   66    0  4.499567  1.500091
   67    0  5.000000  1.500000
   68    0  0.000000  2.000000
   69    0  0.486328  2.057780
   70    0  1.004874  2.060845
   71    0  1.509369  2.017573
   72    0  1.996852  1.999986
   73    0  2.501210  2.003678
   74    0  2.999981  1.999997
   75    0  3.499947  2.000012
   76    0  3.999892  2.000024
   77    0  4.499785  2.000046
   78    0  5.000000  2.000000
   79    0  0.000000  2.500000
   80    0  0.500000  2.500000
   81    0  1.000000  2.500000
   82    0  1.500000  2.500000
   83    0  2.000000  2.500000
   84    0  2.500000  2.500000
   85    0  3.000000  2.500000
   86    0  3.500000  2.500000
   87    0  4.000000  2.500000
   88    0  4.500000  2.500000
   89    0  5.000000  2.500000

ELEM
    1    1    1    2   14   13    1
   12    1   13   14   28   27    1
   15    1   16   17   36   30    0
   16    1   25   26   32   31    1
   21    1   31   32   38   37    1
   26    1   36   17   43   42    0
   27    1   17   18   44   43    1
   34    1   37   38   52   51    1
   39    1   42   43   60   56    0
   40    1   43   44   61   60    1
   47    1   51   52   58   57    0
   48    1   52   53   54   58    0
   49    1   54   55   59   58    1
   51    1   57   58   69   68    1
   61    1   68   69   80   79    1

MATE
    1    1
       0.0       0.0       3.0       0.0  10000000    0.3000

BOUN
    1    1    0   -1
   12    0    0    1
   25    0    1    0
   31    0    1    0
   37    0    1    0
   51    0    1    0
   57    0    1    0
   68    0    1    0
   79    0    1    0

FORC
```

```
   12    0    1    1
      0.25       0.0
   24    0    1    1
       0.5       0.0
   50    0    1    1
       0.5       0.0
   67    0    1    1
       0.5       0.0
   78    0    1    1
       0.5       0.0
   89    0    1    1
      0.25       0.0


END
MACR
PRIN
TANG
FORM
SOLV
CEQS
DISP
STRE
END
STOP
```