Advanced Finite Element

ME EN 7540 ANSYS Example of Axially Loaded Bar Spring 2006

In this example, we will solve the basic bar problem discussed in *Handout 8* by using a program ANSYS 8.0. The command (batch) method will be used instead of the GUI method. The batch mode allows the use of ANSYS Parametric Design Language (APDL).

Element

Link1 (2-D spar or truss) is chosen as the element type (Fig. 1). Details of input and output specifications of this element can be found in *Element Library* part of the ANSYS document.

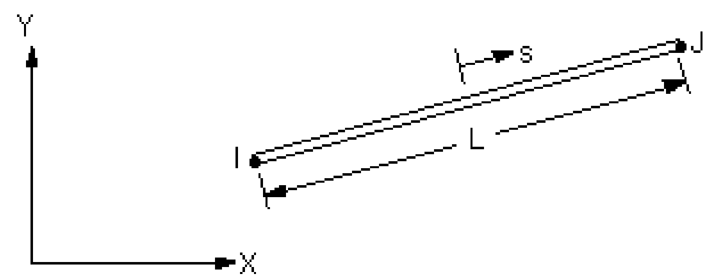

Fig. 1 Link1 (2-D spar).

The shape function used for this element is

$$u = \frac{1}{2}\left(u_i(1-s) + u_j(1+s)\right)$$

The spar element assumes a straight bar, axially loaded at its ends, of uniform properties from end to end. The length of the spar must be greater than zero, so nodes I and J must not be coincident. The spar must lie in an X-Y plane and must have an area greater than zero. The temperature is assumed to vary linearly along the length of the spar.

The displacement function implies a uniform stress in the spar. The initial strain is also used in calculating the stress stiffness matrix, if any, for the first cumulative iteration.

Input Commands

The command lines used for the analysis of the basic bar problem discussed in *Handout 8* are illustrated in Table 1.

Table 1 Input command (batch file)

```
/prep7                                  !preprocessor mode
L = 1                                   !initialize length L
emax = 3                                !number of elements
area = 1                                !cross-sectional area
C = 1                                   !body load constant q(x) = Cx
R = 1                                   !applied load R

et,1,1                                  !element type = Link1
r,1,area                                !real constant: area = 1
mp,ex,1,1                               !young modulus
mp,prxy,1,0.3                           !poisson's ratio

*do,n,1,emax+1                          !nodes creation loop
n,n,L/emax*(n-1),0
*enddo
                                        !elements creation loop
*do,n,1,emax
e,n,n+1
*enddo

d,1,all,0                               !displacement boundary condition

*dim,fi,array,emax,1                    !set the force fi array dimension
*dim,fj,array,emax,1                    !set the force fj array dimension

*do,e,1,emax                            !define load fi and fj for each element
uniform = (L/emax)*((e-1)*C*L/emax)
fi(e) = 1/3*(C*(L/emax)**2/2) + uniform/2
fj(e) = 2/3*(C*(L/emax)**2/2) + uniform/2
*end do

f,1,fx,fi(1)                            !apply forces to nodes
*do,n,2,emax
f,n,fx,fi(n)+fj(n-1)
*enddo
f,emax+1,fx,fj(emax)+R

fini

/solu                                   !solution mode
solve                                   !solve the problem
fini

/post1                                  !postprocessing mode
prnsol,ux                               !list nodal displacements
ETABLE,saxl,LS,1                        !create table containing element axial
pretab,saxl                             !list element axial stress

fini
```

Results

The result outputs from ANSYS are shown in Table 2 and Table 3, nodal displacement and element axial stress.

Table 2 Nodal displacement results

```
PRINT UX   NODAL SOLUTION PER NODE

  ***** POST1 NODAL DEGREE OF FREEDOM LISTING *****

  LOAD STEP=     1  SUBSTEP=     1
   TIME=    1.0000      LOAD CASE=   0

  THE FOLLOWING DEGREE OF FREEDOM RESULTS ARE IN GLOBAL COORDINATES

    NODE      UX
       1   0.0000
       2  0.49383
       3  0.95062
       4   1.3333

 MAXIMUM ABSOLUTE VALUES
 NODE          4
 VALUE    1.3333
```

Table 3 Element axial stress results

```
PRINT ELEMENT TABLE ITEMS PER ELEMENT

  ***** POST1 ELEMENT TABLE LISTING *****

    STAT    CURRENT
    ELEM    SAXL
       1   1.4815
       2   1.3704
       3   1.1481

 MINIMUM VALUES
 ELEM          3
 VALUE    1.1481

 MAXIMUM VALUES
 ELEM          1
 VALUE    1.4815
```