ENG 257-(EQC 1993/157)

Development of a Reinforced Concrete Plastic Hinge Model and Appendices (2 volumes)

Ms Kim Tania Douglas, Dept of Civil & Resource Engineering, University of Auckland

UniServices

Auckland UniServices Limited

ENG 257

Our Ref: 4742

EQC Project 157

DEVELOPMENT OF A REINFORCED CONCRETE PLASTIC HINGE MODEL

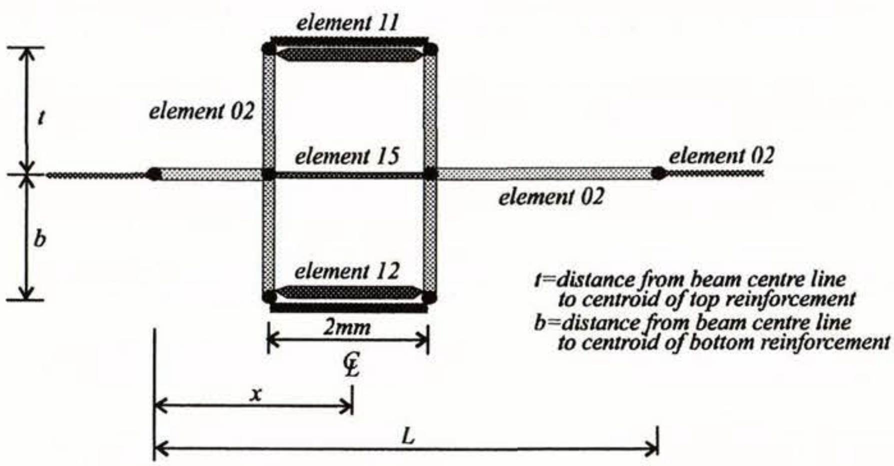
APPENDICES

APPENDICES

Appendix A

Hinge Elements - General Notes

Elements 11, 12 and 15, joined by rigid 02 elements positioned as shown in Fig. A.1, form the basis of the hinge sub-structure. The vertical rigid 02 elements are assigned stiffness values of approximately $1000EI_{eff}$, while the stiffness of the horizontal 02 rigid elements is set at approximately $2EI_{eff}$. When inputting the respective element information the elements must be entered in the following sequence: element 11 - steel elements, element 12 - concrete elements, element 15 - shear element, element 02 - beam-column elements.



For Reversing Plastic Hinges L=*depth of beam* $x=(f_n+f_p)/4$	*For Negative Moment Uni-directional Plastic Hinges* L=*depth of beam* $x=f_n/2$	*For Positive Moment Uni-directional Plastic Hinges* L=*two times depth of beam* x= *depth of beam*

Fig. A.1 Positioning of Hinge Sub-Structure Elements

Although it appears as though there is a reasonable amount of repetition from one hinge element group's input information to the next this has been done simply to avoid transferring and storing large amounts of data, and thus enhance computational efficiency.

A1 Steel Element

DRAIN-ANAL USER GUIDE
INPUT DATA SECTION
STEEL ELEMENT (TYPE 11)

C2.11(a) Control Information
One line

Columns	Notes	Variable	Data
1-5(I)			No. of stiffness types (max.40). See Section C2.11(b).

C2.11(b). Stiffness Types
One line for each stiffness type

Columns	Notes	Variable	Data
1-5(I)			Stiffness type number, in sequence beginning with 1.
6-15(R)			Young's modulus.
16-25(R)			Section area.
26-35(R)			Yield stress.
36-45(R)			Strain at the initiation of strain hardening.
46-55(R)			Ultimate strain.
56-65(R)			Ultimate stress.
66-75(R)			Hinge percentage inelastic strength softening factor

C2.11(c). Specified 'Event' Strains
One line for each stiffness type

Columns	Notes	Variable	Data
1-5(I)			Stiffness type number, in sequence beginning with 1.
6-12(R)			First event strain (> strain at initiation of strain hardening).
13-19(R)			Second event strain..
20-26(R)			Third event strain.
:	:	:	: : :
55-61(R)			Eighth event strain.
62-68(R)			Ninth
69-75(R)			Tenth event strain (< ultimate strain)

C2.11(d). Element Generation Commands

One line for each generation command. The first element can be assigned any number. Subsequent elements must be defined in numerical sequence. Lines for the first and last elements must be included.

Columns	Notes	Variable	Data
1-5(I)			Element number, or number of first element in a sequentially numbered series of elements to be generated by this command.
6-10(I)			Node number at element end, i
11-15(I)			Node number at element end, j
16-20(I)			Node number increment for element generation. Default=1
21-25(I)			Stiffness type number.
26-30(I)			Hinge number (0: if not in a hinge).
31-35(I)			Hinge position(0: if not in a hinge 1: if in top half of beam 2: if in bottom half of beam).

One line of hinge information placed between the generation command lines of the associated top and bottom steel elements. Omit if elements not part a hinge sub-structure.

Columns	Notes	Variable	Data
1-10(R)			Shear reinforcement area.
11-20(R)			Shear reinforcement yield stress.
21-30(R)			Stirrup spacing.
31-40(R)			Distance between top and bottom steel centroids.
41-50(R)			Positive moment hinge moment to shear ratio.
51-60(R)			Negative moment hinge moment to shear ratio.
61-70(R)			Length of rigid end zone penetration ($\geq$0.0: reversing & -ve mom. uni-directional hinges = -f : +ve mom. uni-directional hinges)
71-80(R)			Factor to allow for rigid end zone penetration ($\geq$1.0: reversing & -ve mom. uni-directional hinges = 2.0: +ve mom. uni-directional hinges)

A2 Concrete Element

DRAIN-ANAL USER GUIDE
INPUT DATA SECTION
CONCRETE ELEMENT (TYPE 12)

C2.11(a) Control Information
One line

Columns	Notes	Variable	Data
1-5(I)			No. of stiffness types(max.40). See Section C2.12(b).

C2.11(b). Stiffness Types
One line for each stiffness type

Columns	Notes	Variable	Data
1-5(I)			Stiffness type number, in sequence beginning with 1.
6-15(R)			Concrete modulus.
16-25(R)			Gross area of the cracked section, A_g.
26-35(R)			Area (usually taken as 1/2 cracked section area).
36-45(R)			Tensile yield stress.
46-55(R)			Cylinder strength, f'_c.
56-65(R)			Contact stress effects code (0:ignore, 1:include).
66-75(R)			Factor multiplied by $A_g f'_c$ to give the contact stress at crack closure (usually taken as 0.05)

C2.11(c). Element Generation Commands
One line for each generation command. The first element can be assigned any number. Subsequent elements must be defined in numerical sequence. Lines for the first and last elements must be included.

Columns	Notes	Variable	Data
1-5(I)			Element number, or number of first element in a sequentially numbered series of elements to be generated by this command.
6-10(I)			Node number at element end, i
11-15(I)			Node number at element end, j
16-20(I)			Node number increment for element generation. Default=1
21-25(I)			Stiffness type number.
26-30(I)			Hinge number (0: if not in a hinge).
31-35(I)			Hinge position(0:not in a hinge, 1: in top half of beam 2: in bottom half of beam).

A3 Shear Element

DRAIN-ANAL USER GUIDE
INPUT DATA SECTION
SHEAR ELEMENT (TYPE 15)

C2.11(a) Control Information
One line

Columns	Notes	Variable	Data
1-5(I)			No. of stiffness types (max.40). See Section C2.15(b).

C2.11(b). Stiffness Types - Flexure
One line for each stiffness type

Columns	Notes	Variable	Data
1-5(I)			Stiffness type number, in sequence beginning with 1.
6-15(R)			Top steel - Young's modulus.
16-25(R)			Cross sectional area.
26-35(R)			Yield stress.
36-45(R)			Distance from bottom of beam to steel centroid
46-55(R)			Bottom steel - Young's modulus.
56-65(R)			Cross sectional area.
66-75(R)			Yield stress.
76-85(R)			Distance from top of beam to steel centroid.

C2.11(c). Stiffness Types - Shear
One line for each stiffness type

Columns	Notes	Variable	Data
1-5(I)			Stiffness type number, in sequence beginning with 1.
6-15(R)			Shear reinforcement area.
16-25(R)			Stirrup leg length.
26-35(R)			Stirrup spacing.
36-45(R)			Factor to allow for axial load effects.
46-55(R)			Distance between top and bottom steel centroids.

C2.15(d). Shear Reinforcement Properties

One line for each stiffness type

Columns	Notes	Variable	Data
1-5(I)			Stiffness type number, in sequence beginning with 1.
6-15(R)			Young's modulus.
16-25(R)			Yield stress.
26-35(R)			Strain at the initiation of strain hardening.
36-45(R)			Ultimate strain.
46-55(R)			Ultimate stress.

C2.11(e). Element Generation Commands

One line for each generation command. The first element can be assigned any number. Subsequent elements must be defined in numerical sequence. Lines for the first and last elements must be included.

Columns	Notes	Variable	Data
1-5(I)			Element number, or number of first element in a sequentially numbered series of elements to be generated by this command.
6-10(I)			Node number at element end, i
11-15(I)			Node number at element end, j
16-20(I)			Node number increment for element generation. Default=1
21-25(I)			Stiffness type number.
26-30(I)			Hinge number (0: if not in a hinge).

Appendix B

B1 Steel Element Response Subroutine

```
c ************************************************************************
      SUBROUTINE RESP11(kresis,ksave,kgem,kstep,ndof,kst,kenr,ener,
     1          ened,enso,tim,beto,relas,rdamp,rinit,ddise,
     2          dise,vele,igrc)
c ************************************************************************
c STEEL TRUSS ELEMENT
c   Element response. Update element state, form static and
c   damping resisting forces, perform energy calculations, update
c   damage measures, and put element results in /THELM/ for saving
c   or printing.
c -----------------------------------------------------------------
c DOUBLE PRECISION / LARGE
      include 'double.h'
c -----------------------------------------------------------------
c  CALLED FROM : respxx (once for each sub-step).
c -----------------------------------------------------------------
c  INPUT
c       kresis      = indicator for calculating resisting forces
c                     ( 1: static only, 2: static and damping).
c       ksave       = indicator for saving element results
c                     (1: y, 0: n).
c       kgem        = second order analysis code (> 0: y, 0: n).
c       ndof        = no. of element DOF.
c       kenr        = energy calculation indicator
c                     (2: static + dynamic, 1: static, 0: none).
c       tim         = time for current step.
c       beto        = initial stiffness damping factor.
c       ddise(ndof) = element nodal incremental displacement vector.
c       ddis(ndof)  = element total nodal displacement vector.
c       vele(ndof)  = element nodal velocity vector.
c  OUTPUT
c       ener        = change of element energy.
c       ened        = change of element damping energy.
c       enso        = change of element second-order energy.
c       relas(ndof) = element static resisting force vector.
c       rdamp(ndof) = element damping resisting force vector.
c       rinit(ndof) = element initial resisting force vector.
c  MODIFY
c       aint        = initial force.
c       area        = cross sectional area of the member.
c       b           = ratio of inelastic to elastic steel modulus.
c       cosa        = cosine of element inclination angle (from end I to end j, anti-clockwise).
c       dsf         = factored change in force increment.
c       dsp         = factored change in force increment.
c       eal         = EA/L component.
c       eale        = EA/L elastic component.
```

```
c       ealep       =  element stiffness between events.
c       ealep       =  slope of the line between events.
c       ealepr      =  inelastic stiffness at load reversal.
c       ealh        =  current EA/L stiffness used for hinge calculations.
c       ealx        =  previous EA/L component
c       Eb          =  slope of the strain hardening asymptote for cyclic curves with strains less than 6%.
c       eps_*       =  Four user input strain levels, where; * = 1,2,3,4.
c       epsampa     =  strain amplitude of the previous cyclic ascending half cycle.
c       epsampd     =  strain amplitude of the previous cyclic descending half cycle.
c       epsasc      =  the sum of the strain amplitudes for all ascending curves after the yield strain is
c                      first exceeded.
c       epsb        =  6% strain.
c       epsdes      =  the sum of the strain amplitudes for all desending curves after the yield strain is
c                      first exceeded.
c       epsevt      =  strain at the next event.
c       epsh        =  strain at the starting point of the strain hardening.
c       epsmn       =  strain at the starting point of the tensile loading curve.
c       epsmx       =  strain at the starting point of the compressive loading curve.
c       epso        =  train at the intersection of the two asymptotes.
c       epsr        =  strain at load reversal.
c       epsu        =  strain at the ultimate stress.
c       epsy        =  yield strain of the steel.
c       Es          =  elastic modulus of the steel.
c       f0_06       =  stress at 6% strain on the monotonic curve.
c       fevt        =  stress at the next event.
c       fl          =  element length.
c       fmn         =  stress at the starting point of the tensile loading curve.
c       fmx         =  stress at the starting point of the compressive loading curve.
c       fo          =  stress at the intersection of the two asymptotes.
c       fr          =  stress at load reversal.
c       fu          =  ultimate stress of the steel.
c       fy          =  yield stress of the steel.
c       ihing       =  hinge number, 0:not in a hinge.
c       ihpos       =  code indicating position in plastic hinge; 0:not included, 1:top half of beam,
c                      2:bottom half of beam.
c       imem        =  element no.
c       isenn       =  step no. for senn.
c       isenp       =  step no. for senp.
c       ivenn       =  step no. for venn.
c       ivenp       =  step no. for venp.
c       kbuck       =  buckling code for element (1 : buckles in compression, 0: yields in compression).
c       kdum        =  dummy
c       kelastic    =  code indicating whether in a post yield elastic or inelastic cycle, 0:inelastic,
c                      1:elastic.
c       kody        =  current yield code.
c       kodyr       =  yield code at load reversal.
c       kodyx       =  previous yield code.
c       kst         =  stiffness formation code ( 1: y, 0: n).
c       kstep       =   step no. in this segment
c       lm(4)       =  location matrix.
c       monot       =  logical variable to determine whether the hysteresis c curve is monotonic.
c       monotelas   =  logical variable to determine whether the hysteresis curve has reversed
c                      inelastically.
c       nodj        =  node no. for element end j.
```

```
c      pyc          =  compressive yield force.
c      pycsh        =  compressive force at the initiation of strain hardening.
c      pyt          =  tensile yield force.
c      pytsh        =  tensile force at the initiation of strain hardening.
c      R            =  a parameter which defines the curvature of the unloading or loading stress-strain
c                      curves.
c      rcb          =  a user input percentage softening in the inelastic cyclic reyielding stress level, due
c                      to kinking of reinforcement within the hinge region.
c      revld        =  laod reversal indicator, if revld<0 the load has reversed.
c      rlp          =  current element hinge length.
c      rlpc         =  hinge length used y concrete elements.
c      rlpevt       =  hinge length at the next event.
c      rlpevtold    =  previous hinge length.
c      rlpv         =  hinge length used by shear element.
c      senn         =  envelop value for negative force.
c      senp         =  envelop value for positive force.
c      sep          =  current steel force.
c      sepevtb      =  past event force.
c      sepevtf      =  next event force.
c      sepevtold    =  force at the last event.
c      sepevtr      =  next new event force on the inelastic yielding curve.
c      sepr         =  maximum or minimum force at load reversal from inelastic yield.
c      sina         =  sine of element inclination angle.
c      st           =  total force for element at the end of the step.
c      stdp         =  damping force for previous step.
c      stp          =  total force for previous step.
c      venn         =  envelop value for negative deformation.
c      venp         =  envelop value for positive deformation.
c      vtot         =  total element deformation.
c      vtotevtb     =  past event deformation.
c      vtotevtf     =  next event deformation.
c      vtotevtold   =  deformation at the last event.
c      vtotevtr     =  next new event deformation on the inelastic curve.
c      vtotmn       =  deformation at the starting point of the tensile loading curve.
c      vtotmx       =  deformation at the starting point of the compressive loading curve.
c      vtotold      =  total deformation at the beginning of the current step.

c ---------------------------------------------------------------------
c  LABELLED COMMONS
      include 'infel11.h'
c hinge addition - KTD 15/12/93
      include 'hinge.h'
      common/tapes/inp,iou
      common /work/ fac,factor,facacc,dv,delta,delti,w(1994)
c      delta        =  P-delta/L
c      delti        =  P-delta (increment)/L
c      dsel         =  change in force for elastic (E) component.
c      dsep         =  change in forec for elasto-plastic (E-P)
c                      component.
c      dv           =  deformation increment.
c      dvp          =  plastic deformation increment.
c      fac,factor,facac= event factor used to trace out the
c                      non-linear path.
c ---------------------------------------------------------------------
```

```
c  ARGUMENT DECLARATIONS
    dimension relas(ndof),rdamp(ndof),rinit(ndof),
   1         ddise(ndof),dise(ndof),vele(ndof)
c ------------------------------------------------------------------

c ------------------RESET ENVELOPE STEP VALUES FOR NEW ANALYSIS SEGMENT
    if(kstep.eq.-1)then
     isenn=0
     isenp=0
     ivenn=0
     ivenp=0
     kstep=1
    end if

c hinge addition - KTD 22/12/93
c --------------------------------------READ CURRENT HINGE INFORMATION

    if(ihing.ne.0) then
     call hstart(ihing,igrc)
     if(ihpos.eq.1) then
      fcr=fn+rigid
      rlp=rlpn
     elseif(ihpos.eq.2) then
      fcr=fp+rigid
      rlp=rlpp
     endif
    endif

c --------------------CALCULATE EXTENSION AND UPDATE TOTAL DEFORMATION
    vtotold=vtot
    dv=cosa*(ddise(3)-ddise(1))+sina*(ddise(4)-ddise(2))
    vtot=vtot+dv
    revld=dv*dvold

c -------------------------------------CHECK IF THE CURVE IS MONOTONIC
    if(((sep.ge.pyt).or.(sep.le.pyc)).and.(revld.lt.0.d0)) monot=0

c ---------------------------------TRACE STRESS STRAIN HYSTERESIS LOOP
c ---describes the stress strain characteristics of steel under cyclic
c  loading
c ------------------------------------------------------------------
    facac=0.d0

    do while(facac.lt.0.999999d0)
     factor=1.d0 - facac

c --------------------------------------MONOTONIC STRESS STRAIN CURVE
c---monotonic loading
     if(monot.eq.1) then
      if(kody.eq.0) then
c---get factor for status change
       pyy=0.d0
       dsep=eale*dv
       ealh=eale
```

```
        if(dv.gt.0.d0) then
         fac=(pyt-sep)/dsep
         if(fac.lt.factor) then
c---positive yield
          factor=fac
          pyy=pyt
         endif
        elseif(dv.lt.0.d0) then
         fac=(pyc-sep)/dsep
         if(fac.lt.factor) then
c---negative yield
          factor=fac
          pyy=pyc
         endif
        endif
c---update and calculate (elastic) energy
        if(pyy.eq.0.d0) then
c---remains elastic
         dsp=factor*dsep
         sep=sep+dsp
         if(kenr.gt.0) ener=ener+((sep-dsp*0.5d0)-aint)*factor*dv
        else
c---new yield
         if(kenr.gt.0) ener=ener+(((sep+pyy)*0.5d0)-aint)*factor*dv
         sep=pyy
         if(dv.gt.0.d0) then
          pytsh=sep+0.001*Es*area*(epsh-epsy)
          if(ihing.ne.0) then
           rlpevtold=rlp
           rlpevt=rigidf*(fcr+((pytsh-pyt)*d_dd/dabs(seps)))
           if(rigidf.eq.2.d0) then
            if((rlpevt.lt.rlpevtold).or.
     1                   (rlpevt.gt.4.d0*d_dd)) then
             rlpevt=rlpevtold
            endif
           elseif(rigidf.ne.2.d0) then
            if((rlpevt.lt.rlpevtold).or.
     1                   (rlpevt.gt.2.d0*d_dd)) then
             rlpevt=rlpevtold
            endif
           endif
           vtottsh=0.5*(epsy+epsh)*rlpevt
          else
           vtottsh=rlp*epsh
          endif
          vtotevt=vtottsh
          sepevt=pytsh
          ealep=(sepevt-pyt)/(vtotevt-(vtot-(1.d0-factor)*dv))
         elseif(dv.lt.0.d0) then
          pycsh=sep-0.001*Es*area*(epsh-epsy)
          if(ihing.ne.0) then
           vtotcsh=-0.5*epsh*rlp
          else
           vtotcsh=-rlp*epsh
```

```
            endif
            vtotevt=vtotcsh
            sepevt=pycsh
            ealep=(sepevt-pyt)/(vtotevt-(vtot-(1.d0-factor)*dv))
          endif
          kody=1
        endif

c---yielded or buckled
       elseif(kody.eq.1) then
        pyy=0.d0
        if((kody.ne.kodyx).and.((kodyx.ne.4).or.(kodyx.ne.6))) then
          epsb=0.06d0
          rn0_06=epsb-epsh
          q0_06=epsu-epsh
          rm0_06=((((fu/fy)*(30*q0_06+1)**2)-60*q0_06-1)/
     1          (15*q0_06**2))
          psh0_06=((rm0_06*rn0_06+2)/(60*rn0_06+2))+((rn0_06*
     1          (60-rm0_06))/(2*(30*q0_06+1)**2))
          if(epsb.le.epsy) then
            f0_06=Es*epsb
          elseif((epsb.gt.epsy).and.(epsb.le.epsh)) then
            f0_06=fy
          elseif((epsb.gt.epsh).and.(epsb.le.epsu)) then
            f0_06=psh0_06*fy
          endif
          Eb=(f0_06-fy)/(epsb-epsy)
          b=Eb/Es
        endif
c ---calculate the strain-hardening modulus
        dsep=ealep*dv
        ealh=ealep
c---get factor for status change
        if(dv.gt.0.d0) then
          fac=(pytsh-sep)/dsep
          if(fac.lt.factor) then
c---positive strain hardening
            factor=fac
            pyy=pytsh
          endif
        elseif(dv.lt.0.d0) then
          fac=(pycsh-sep)/dsep
          if(fac.lt.factor) then
c---negative strain hardening
            factor=fac
            pyy=pycsh
          endif
        endif
c---update and calculate energy
        if(pyy.eq.0.d0) then
c---not strain hardening
          dsp=factor*dsep
          sep=sep+dsp
          if(kenr.gt.0) ener=ener+((sep-dsp*0.5d0)-aint)*factor*dv
```

```
         else
c---started strain hardening
          if(kenr.gt.0) ener=ener+(((sep+pyy)*0.5d0)-aint)*factor*dv
          vtotevtold=vtotevt
          sepevtold=sepevt
          sep=pyy
          kody=2
         endif

        elseif((kody.eq.2).or.(kody.ge.8)) then
         pyy=0.d0
c---continues to strain harden
         if((kody.ne.kodyx).and.((kodyx.ne.4).and.(kodyx.ne.6))) then
c---calculate new event force and stiffness
          if(kody.ne.17) then
           if(kody.eq.2) then
            epsev=eps(1)
           elseif(kody.ge.8) then
            epsev=eps((kody-6))
           endif
           if(epsev.eq.1.0d8) then
            epsev=epsu
            fev=fu
            kody=17
           else
            rn_evt=epsev-epsh
            q_evt=epsu-epsh
            rm_evt=((((fu/fy)*(30*q_evt+1)**2)-60*q_evt-1)/
     1                          (15*q_evt**2))
            psh_evt=((rm_evt*rn_evt+2)/(60*rn_evt+2))+((rn_evt*
     1                  (60-rm_evt))/(2*(30*q_evt+1)**2))
            fev=psh_evt*fy
           endif
          elseif(kody.eq.17) then
           epsev=epsu
           fev=fu
          endif
          if(dv.lt.0.d0) then
           epsevt=-epsev
           fevt=-fev
          elseif(dv.gt.0.d0) then
           epsevt=epsev
           fevt=fev
          endif
          sepevt=fevt*area
          if(ihing.ne.0) then
           if(dv.gt.0.d0) then
            rlpevtold=rlpevt
            rlpevt=rigidf*(fcr+((sepevt-pyt)*d_dd/dabs(seps)))
            if(rigidf.eq.2.d0) then
             if((rlpevt.lt.rlpevtold).or.
     1                    (rlpevt.gt.4.d0*d_dd)) then
              rlpevt=rlpevtold
             endif
```

```
            elseif(rigidf.ne.2.d0) then
             if((rlpevt.lt.rlpevtold).or.
     1                       (rlpevt.gt.2.d0*d_dd)) then
               rlpevt=rlpevtold
              endif
            endif
            vtotevt=0.5*(epsevt+epsy)*rlpevt
           elseif(dv.lt.0.d0) then
            vtotevt=0.5*epsevt*rlp
           endif
          else
           vtotevt=epsevt*rlp
          endif
          ealep=(sepevt-sepevtold)/(vtotevt-vtotevtold)
         endif
         dsep=ealep*dv
         ealh=ealep
         fac=(sepevt-sep)/dsep
         if(fac.lt.factor) then
c---reached event strain level
          factor=fac
          pyy=sepevt
          if(kody.eq.2) then
           kody=8
          else
           kody=kody+1
          endif
         endif
c---update and calculate energy
         if(pyy.eq.0.d0) then
          dsp=factor*dsep
          sep=sep+dsp
          if(kenr.gt.0) ener=ener+((sep-dsp*0.5d0)-aint)*factor*dv
         else
          if(kenr.gt.0) ener=ener+(((sep+pyy)*0.5d0)-aint)*factor*dv
          vtotevtold=vtotevt
          sepevtold=sepevt
          sep=pyy
          if(kody.eq.18) then
           write(iou,15)imem,sep,vtot
           write(*,15)imem,sep,vtot
  15        format(/' *** Exceeded ultimate steel stress value,'
     1          'member = ',i5,' sep = ',f10.4,' vtot = ',f10.4)
           stop
          endif
         endif
        endif

c ------------------------------------------------CYCLIC STRAIN CURVES
       elseif(monot.eq.0) then
c ---yielded and now cycling
        pyy=0.d0
c ---cyclic loading curves
        if(revld.lt.0.d0) then
```

```
          if(ihing.ne.0) rlp=rlpv
          if((kody.eq.3).or.(kody.eq.4)) then
           vtotdum=vtotevtb
           sepdum=sepevtb
           vtotevtb=vtotevtf
           sepevtb=sepevtf
           vtotevtf=vtotdum
           sepevtf=sepdum
           if(kody.eq.3) then
            kody=4
            kelastic=1
           elseif(kody.eq.4) then
            kody=3
            kelastic=0
           endif
          elseif(kody.eq.5) then
           vtotevtb=vtotold
           sepevtb=sep
           vtotevtf=vtotold+(vtotr-vtotevtold)
           sepevtf=sep+(sepr-sepevtold)
           ealep=(sepevtf-sepevtb)/(vtotevtf-vtotevtb)
           kody=4
           kelastic=1
          elseif(kody.eq.6) then
           vtotevtb=vtotold
           sepevtb=sep
           vtotevtf=vtotold-(vtotr-vtotevtold)
           sepevtf=sep-(sepr-sepevtold)
           ealep=(sepevtf-sepevtb)/(vtotevtf-vtotevtb)
           kody=3
           kelastic=0
          else
           if(dv.gt.0.d0) then
            if(ihing.ne.0) then
             epsmn=2.d0*vtotold/rlp
            else
             epsmn=vtotold/rlp
            endif
            fmn=sep/area
            vtotmn=vtotold
            epsampd=dabs(epsmn-epsmx)
            if(kelastic.eq.1) then
             epsdes=epsdes+dabs(epsmn-epsr)
            else
             epsdes=epsdes+epsampd
            endif
            epsr=epsmn
            fr=fmn
            for=fy
            epsor=epsr+(for-fr)/(Es)
            if(epsor.gt.epsy) epsor=epsy
            epsrev=epsampd/epsy
           elseif(dv.lt.0.d0) then
c---load reversal from tensile yield - update hinge length
```

```
            if(ihing.ne.0) then
             rlpevt=rigidf*(fcr+((sep-pyt)*d_dd/dabs(seps)))
             if(rigidf.eq.2.d0) then
              if((rlpevt.lt.rlp).or.(rlpevt.gt.4.d0*d_dd)) then
               rlpevt=rlp
              else
               rlp=rlpevt
              endif
             elseif(rigidf.ne.2.d0) then
              if((rlpevt.lt.rlp).or.(rlpevt.gt.2.d0*d_dd)) then
               rlpevt=rlp
              else
               rlp=rlpevt
              endif
             endif
             rlpevtold=rlpevt
             epsmx=2.d0*vtotold/rlpevt-epsy
c because the initial hinge length is purely an estimate it is possible that
c at low forces and high axial loads the hinge length is sufficiently large
c enough so that the maximum post elastic strain is less than the yield strain
c -this is not possible - thus the addition of the following statement
             if(epsmx.lt.epsy) then
              if(kody.eq.1) then
               epsmx=epsy+((sep/area)-fy)/(0.001*Es)
              elseif(kody.gt.1) then
               epsmx=epsh+((sep/area)-(pytsh/area))/Eb
              endif
             endif
            else
             rlpevt=rlp
             epsmx=vtotold/rlpevt
            endif
            rlpc=rlpevt
c the hinge softening reduction factor is ignored when calculating the maximum
c stress level as the softening only occurs when the element is yielding, and
c should not affect the unloading curves, therefore the stress-strain
c relationship employed for modelling the Bauschinger effect should be used
c with the unaltered values.
            fmx=sep/area
            vtotmx=vtotold
            epsampa=dabs(epsmx-epsmn)
            if(kelastic.eq.1) then
             epsasc=epsasc+(epsmx-epsr)
            else
             epsasc=epsasc+epsampa
            endif
            epsr=epsmx
            fr=fmx
            for=-fy
            epsor=epsr+(for-fr)/(Es)
            if(epsor.lt.(-epsy)) epsor=-epsy
            epsrev=epsampa/epsy
           endif
           epssht=(epsasc-0.925*epsdes)/epsy
```

```
        X=(log(epsrev))**2.0*dabs(log(dabs(epssht+20.0))-2.0)
        Y=0.4+3.6*X**(-0.9)
        R=Y*log(epsrev)
        epso=(for-fr+Es*epsr-Eb*epsor)/(Es-Eb)
        fo=Es*epso+fr-Es*epsr
        if(dv.gt.0.d0) fevt=fmn+1.35*fy
        if(dv.lt.0.d0) fevt=fmx-1.35*fy
        vtotevtold=vtotold
        sepevtold=sep
        vtotevtr=vtotevt
        sepevtr=sepevt
        ealepr=ealep
        kodyr=kody
        vtotr=vtotold
        sepr=sep
        sepevt=fevt*area
        epsevt=(fevt-fr)/(Es)+epsr
        kody=3
        kelastic=0
       endif

      elseif(revld.gt.0.d0) then
       if((kody.ne.kodyx).and.((kodyx.ne.4).and.(kodyx.ne.6))) then
        if(kody.eq.5) then
         if(sep.eq.sepevtold) then
          if(dv.gt.0.d0) fevt=fmn+2.d0*fy
          if(dv.lt.0.d0) fevt=fmx-2.d0*fy
          sepevt=(1.d0-rcb)*fevt*area
          epsevt=(fevt-fr)/Es+epsr
         endif
        elseif(kody.ge.7) then
c---calculate new event force and stiffness
         if(kelastic.eq.1) then
          if(dv.gt.0.d0) then
           epsr=epsmn
           fr=fmn
           for=fy
           epsor=epsr+(for-fr)/(Es)
           if(epsor.gt.epsy) epsor=epsy
           epsrev=epsampd/epsy
          elseif(dv.lt.0.d0) then
           epsr=epsmx
           fr=fmx
           for=-fy
           epsor=epsr+(for-fr)/(Es)
           if(epsor.lt.(-epsy)) epsor=-epsy
           epsrev=epsampa/epsy
          endif
          epssht=(epsasc-0.925*epsdes)/epsy
          X=(log(epsrev))**2.0*dabs(log(dabs(epssht+20.0))-2.0)
          Y=0.4+3.6*X**(-0.9)
          R=Y*log(epsrev)
          epso=(for-fr+Es*epsr-Eb*epsor)/(Es-Eb)
          fo=Es*epso+fr-Es*epsr
```

```
            endif
            if(kody.ne.17) then
             epsev=eps((kody-6))
             if(epsev.eq.1.0d8) then
              if(dv.lt.0.d0) then
               epsevt=epsmx-epsu
               fevt=fmx-fy-fu
              elseif(dv.gt.0.d0) then
               epsevt=epsmn+epsu
               fevt=fmn+fy+fu
              endif
              kody=17
             else
              rn_evt=epsev-epsh
              q_evt=epsu-epsh
              rm_evt=((((fu/fy)*(30*q_evt+1)**2)-60*q_evt-1)/
     1                             (15*q_evt**2))
              psh_evt=((rm_evt*rn_evt+2)/(60*rn_evt+2))+((rn_evt*
     1                  (60-rm_evt))/(2*(30*q_evt+1)**2))
              fev=psh_evt*fy
              if(dv.lt.0.d0) then
               epsevt=((fmx-fy-fev)-fevt)/(0.5*(Es+Eb))+epsevt
               fevt=fmx-fy-fev
              elseif(dv.gt.0.d0) then
               epsevt=((fmn+fy+fev)-fevt)/(0.5*(Es+Eb))+epsevt
               fevt=fmn+fy+fev
              endif
             endif
            elseif(kody.eq.17) then
             if(dv.lt.0.d0) then
              epsevt=epsmx-epsu
              fevt=fmx-fy-fu
             elseif(dv.gt.0.d0) then
              epsevt=epsmn+epsu
              fevt=fmn+fy+fu
             endif
            endif
            sepevt=(1.d0-rcb)*fevt*area
           endif
          endif
         endif

c ---Calculate the strain at the next event
         if((kody.ne.kodyx).and.((kodyx.ne.4).and.(kodyx.ne.6))) then
          if((kody.eq.3).or.((kody.eq.5).and.(sep.eq.sepevtold)).
     1            or.((kody.ge.7).and.(kody.lt.17))) then
c ---Use Newton-Raphson Method
           iter=0
           maxiter=15
           etol=1.d-3
           epsevto=-9.d8
           do while((dabs((epsevt-epsevto)/epsevt).gt.etol).and.
     1                              (iter.le.maxiter))
            iter=iter+1
```

```
        epss=(epsevt-epsr)/(epso-epsr)
        if(epss.lt.0.d0) then
         iter=maxiter+3
        else
         fss=b*epss+(1-b)*epss/(1+epss**R)**(1/R)
         u=fss*(fo-fr)+fr-fevt
         dfss=b+(1-b)*(1-epss**R/(1+epss**R))/
     1                      (1+epss**R)**(1/R)
         derivu=dfss*(fo-fr)/(epso-epsr)
         if(derivu.lt.etol) then
          iter=maxiter+2
         else
          epsevto=epsevt
          epsevt=epsevt-u/derivu
         endif
        endif
       enddo
       if(iter.eq.(maxiter+1)) then
        write(iou,30)maxiter
        write(*,30)maxiter
30       format(/' *** Couldn"t find next event strain ',
     1                  ' within ',i3,' iterations, RESP11')
        stop
       elseif(iter.eq.(maxiter+2)) then
        write(iou,35)
        write(*,35)
35       format(/' *** Divide by zero error ',
     1               ' in Newton-Raphson iterations, RESP11')
        stop
       elseif(iter.eq.(maxiter+3)) then
        write(iou,40)
        write(*,40)
40       format(/' *** Invalid exponentiation error',
     1               ' in Newton-Raphson iterations, RESP11')
        stop
       endif
      endif
      if((kody.eq.3).or.(kody.eq.5)) then
       if(ihing.ne.0) then
        vtotevt=0.5*epsevt*rlp
       else
        vtotevt=epsevt*rlp
       endif
       vtotevtb=vtotold
       sepevtb=sep
       vtotevtf=vtotevt
       sepevtf=sepevt
       ealep=(sepevtf-sepevtb)/(vtotevtf-vtotevtb)
      elseif(kody.ge.7) then
       if(ihing.ne.0) then
        if(dv.gt.0.d0) then
         rlpevtold=rlpevt
         rlpevt=rigidf*(fcr+((sepevt-pyt)*d_dd/dabs(seps)))
         if(rigidf.eq.2.d0) then
```

```
            if((rlpevt.lt.rlpevtold).or.
     1                           (rlpevt.gt.4.d0*d_dd)) then
              rlpevt=rlpevtold
            endif
          elseif(rigidf.ne.2.d0) then
            if((rlpevt.lt.rlpevtold).or.
     1                           (rlpevt.gt.2.d0*d_dd)) then
              rlpevt=rlpevtold
            endif
          endif
          vtotevt=0.5*(epsevt+epsy)*rlpevt
        elseif(dv.lt.0.d0) then
          vtotevt=0.5*epsevt*rlpevt
        endif
      else
        vtotevt=epsevt*rlp
      endif
      ealep=(sepevt-sepevtold)/(vtotevt-vtotevtold)
    endif
  endif

  if(kody.eq.3) then
    pyy=0.d0
    dsep=ealep*dv
    ealh=ealep
    fac=(sepevtf-sep)/dsep
    if(fac.lt.factor) then
      factor=fac
      vtotold=vtotevtf
      pyy=sepevtf
      if(vtotevtb.eq.vtotr) then
        vtotevtold=vtotevtf
        sepevtold=sepevtf
      else
c---completion of elastic inner loop
        ealep=(sepevt-pyy)/(vtotevt-vtotold)
      endif
      kody=5
      if(revld.lt.0.d0) revld=1.d0
    endif

  elseif(kody.eq.4) then
    pyy=0.d0
    dsep=ealep*dv
    ealh=ealep
    fac=(sepevtf-sep)/dsep
    if(fac.lt.factor) then
      factor=fac
      vtotold=vtotevtf
      pyy=sepevtf
      if(pyy.ne.sepr) then
        vtotevtb=vtotevtf
        sepevtb=sepevtf
        vtotevtf=vtotr
```

```
      sepevtf=sepr
      ealep=(sepevtf-sepevtb)/(vtotevtf-vtotevtb)
      kody=6
    elseif(pyy.eq.sepr) then
      kody=kodyr
      vtotevtold=vtotevtf
      sepevtold=sepevtf
      vtotevt=vtotevtr
      sepevt=sepevtr
      ealep=ealepr
      if(monotelas.eq.1) monot=1
    endif
    if(revld.lt.0.d0) revld=1.d0
  endif

elseif(kody.eq.5) then
  pyy=0.d0
  dsep=ealep*dv
  ealh=ealep
  fac=(sepevt-sep)/dsep
  if(fac.lt.factor) then
    factor=fac
    vtotold=vtotevt
    pyy=sepevt
    vtotevtold=vtotevt
    sepevtold=sepevt
    kody=7
    monotelas=0
  endif

elseif(kody.eq.6) then
  pyy=0.d0
  dsep=ealep*dv
  ealh=ealep
  fac=(sepevtf-sep)/dsep
  if(fac.lt.factor) then
    factor=fac
    vtotold=vtotr
    pyy=sepr
    kody=kodyr
    vtotevtold=vtotevtf
    sepevtold=sepevtf
    vtotevt=vtotevtr
    sepevt=sepevtr
    ealep=ealepr
    if(monotelas.eq.1) monot=1
  endif

elseif(kody.ge.7) then
  pyy=0.d0
  dsep=ealep*dv
  ealh=ealep
  fac=(sepevt-sep)/dsep
  if(fac.lt.factor) then
```

```
        factor=fac
        vtotold=vtotevt
        pyy=sepevt
        vtotevtold=vtotevt
        sepevtold=sepevt
        kody=kody+1
       endif
      endif

c ---update and calculate energy if an event has not occurred
      if(pyy.eq.0.d0) then
        dsp=factor*dsep
        sep=sep+dsp
        if(kenr.gt.0) ener=ener+((sep-dsp*0.5d0)-aint)*factor*dv
c ---update and calculate energy if an event has occurred
      elseif(pyy.ne.0.d0) then
        if(kenr.gt.0) ener=ener+(((sep+pyy)*0.5d0)-aint)*factor*dv
        sep=pyy
        if(kody.eq.18) then
          write(iou,50)imem,sep,vtot
          write(*,50)imem,sep,vtot
 50       format(/' *** Exceeded ultimate steel stress value,'
  1             'member = ',i5,' sep = ',f10.4,' vtot = ',f10.4)
          stop
        endif
      endif
     endif

c ------------------------------------------CHECK COMPLETION OF CYCLE
      facac=facac+factor
     enddo

     if(ealh.lt.0.d0) then
      write(iou,55)imem,kody
      write(*,55)imem,kody
 55  format(/' *** Negative stiffness,'
  1              ' member = ',i5,2x,'yield code = ',i5)
      stop
     endif

c ------------------------------------------UPDATE TOTAL STATIC FORCE
c ---save current force for second order energy
     stt=st
     st=sep

c hinge addition - KTD 22/1/93
c ----------------------------ASSIGN AND SAVE UPDATED HINGE INFORMATION
     if(ihing.ne.0) then
c ---update element reactions and new event values
      if(ihpos.eq.1) then
       vtott=vtot
       seprt=st
       if((kody.eq.0).and.((vtotold.eq.0.d0).or.(revld.lt.0.d0)))then
        if(dv.gt.0.d0) then
```

```
      seprtevt=pyt
      vtotrtevt=pyt/ealh
     elseif(dv.lt.0.d0) then
      seprtevt=pyc
      vtotrtevt=pyc/ealh
     endif
    elseif(kody.ne.kodyx) then
     if((kody.eq.4).or.(kody.eq.6)) then
      seprtevt=sepevtf
      vtotrtevt=vtotevtf
     else
      seprtevt=sepevt
      vtotrtevt=vtotevt
     endif
     ealrt=ealh
     rlpn=rlpevtold
    endif
   elseif(ihpos.eq.2) then
    vtotb=vtot
    seprb=st
    if((kody.eq.0).and.((vtotold.eq.0.d0).or.(revld.lt.0.d0)))then
     if(dv.gt.0.d0) then
      seprbevt=pyt
      vtotrbevt=pyt/ealh
     elseif(dv.lt.0.d0) then
      seprbevt=pyc
      vtotrbevt=pyc/ealh
     endif
    elseif(kody.ne.kodyx) then
     if((kody.eq.4).or.(kody.eq.6)) then
      seprbevt=sepevtf
      vtotrbevt=vtotevtf
     else
      seprbevt=sepevt
      vtotrbevt=vtotevt
     endif
     ealrb=ealh
     rlpp=rlpevtold
    endif
   endif
c ---update hinge length for shear and concrete element if relevant
   if((sep.gt.pyt).and.(dv.gt.0.d0)) then
    rlpvold=rlpv
    rlpv=rigidf*(fcr+((sep-pyt)*d_dd/dabs(seps)))
    if(rigidf.eq.2.d0) then
     if((rlpv.lt.rlpvold).or.(rlpv.gt.4.d0*d_dd)) then
      rlpv=rlpvold
     endif
    elseif(rigidf.ne.2.d0) then
     if((rlpv.lt.rlpvold).or.(rlpv.gt.2.d0*d_dd)) then
      rlpv=rlpvold
     endif
    endif
   elseif((kody.eq.3).and.(dv.lt.0.d0)) then
```

```
      if((kody.ne.kodyx).and.(((kodyx.le.2).or.(kodyx.ge.7)).and.
    1                              (kody.ne.17))) then
        if(ihpos.eq.1) then
          rlpnc=rlpc
        elseif(ihpos.eq.2) then
          rlppc=rlpc
        endif
      endif
     endif
     call hfinish(ihing,igrc)
    endif
    dvold=dv

c ----------------------------------E-COMPONENT AND SECOND ORDER ENERGY
c---second order energy (based on static force only)
    if(kgem.gt.0)then
     delta=(sina*(dise(3)-dise(1))+cosa*(dise(2)-dise(4)))/rlp
     delti=(sina*(ddise(3)-ddise(1))+cosa*(ddise(2)-ddise(4)))/rlp
     enso=enso-((st+stt)*0.5)*(delta-delti*0.5)*delti*rlp
    endif

c extra information
c calculate current strain and stress
    if((ihing.ne.0).and.(kody.ne.0)) then
     if(((kody.eq.1).or.(kody.eq.2).or.(kody.ge.7)).and.
    1                              (dv.gt.0.d0)) then
       epscur=2.d0*vtot/rlpv-epsy
     else
       epscur=2.d0*vtot/rlpv
     endif
    elseif((ihing.eq.0).or.(kody.eq.0)) then
     epscur=vtot/rlp
    endif
    fcur=st/area

c ----------------------------------------------STATIC RESISTING FORCES
    call elas11(relas,rinit,dise,ndof,kgem)

c --------------------------------------------DAMPING FORCES AND ENERGY
    if(kresis.eq.2) call damp11(rdamp,vele,kenr,ndof,beto,ened,dv)

c -----------------------------------------------------ENVELOPE VALUES
    call upen11(kstep)

c --------------------------------------------------------SAVE RESULTS
    if(ksave.eq.1) call save11(epscur,fcur)

c -----------------------------------SET INDICATOR FOR STIFFNESS CHANGE
    if (kodyx.ne.kody) kst=1

c ------------------------------------------------------------------
    RETURN
    END
```

B2 Concrete Element Response Subroutine

```
c ************************************************************************
      SUBROUTINE RESP12(kresis,ksave,kgem,kstep,ndof,kst,kenr,ener,
     1            ened,enso,tim,beto,relas,rdamp,rinit,ddise,
     2            dise,vele,igrc)
c ************************************************************************
c CONCRETE TRUSS ELEMENT
c   Element response. Update element state, form static and
c   damping resisting forces, perform energy calculations, update
c   damage measures, and put element results in /THELM/ for saving
c   or printing.
c ----------------------------------------------------------------
c DOUBLE PRECISION / LARGE
      include 'double.h'
c ----------------------------------------------------------------
c  CALLED FROM : respxx (once for each sub-step).
c ----------------------------------------------------------------
c  INPUT
c      kresis       =  indicator for calculating resisting forces
c                      ( 1: static only, 2: static and damping).
c      ksave        =  indicator for saving element results
c                      (1: y, 0: n).
c      kgem         =   second order analysis code (> 0: y, 0: n).
c      ndof         =  no. of element DOF.
c      kenr         =  energy calculation indicator
c                      (2: static + dynamic, 1: static, 0: none).
c      tim          =  time for current step.
c      beto         =  initial stiffness damping factor.
c      ddise(ndof)  =  element nodal incremental displacement vector.
c      ddis(ndof)   =  element total nodal displacement vector.
c      vele(ndof)   =  element nodal velocity vector.
c  OUTPUT
c      ener         =  change of element elasto-plastic energy.
c      ened         =  change of element damping energy.
c      enso         =  change of element second-order energy.
c      relas(ndof)  =  element static resisting force vector.
c      rdamp(ndof)  =  element damping resisting force vector.
c      rinit(ndof)  =  element initial resisting force vector.
c  MODIFY
c      aint         =  initial force.
c      area         =  cross-sectional area of the concrete.
c      cosa         =  cosine of element inclination angle (from end i to end j, anti-clockwise).
c      dvold        =  previous deformation increment.
c      eal          =  current EA/L for the element.
c      ealc         =  stiffness EA/L while cracks are opening and closing.
c      ealcs1       =  initial contact stress stiffness.
c      ealcs2       =  final contact stress stiffness.
c      ealcd        =  pseudo degrading concrete modulus.
c      eale         =  elastic EA/L.
```

```
c    ealh      =  current EA/L stiffness used for hinge calculations.
c    ealsh     =  pseudo strain hardening concrete modulus.
c    ealx      =  previous EA/L.
c    Ec        =  elastic concrete modulus.
c    eff       =  factor allowing for the changing hinge length.
c    fc        =  concrete cyclinder compressive strength.
c    fco       =  the contact stress value at zero strain.
c    fl        =  element length.
c    fy        =  yield stress.
c    ihing     =  hinge number, 0:not in a hinge.
c    ihpos     =  code indicating position in plastic hinge; 0:not included, 1:top half of beam,
c                 2:bottom half of beam.
c    imem      =  element no.
c    isenn     =  step no. for senn.
c    isenp     =  step no. for senp.
c    ivenn     =  step no. for venn.
c    ivenp     =  step no. for venp.
c    kcont     =  contact stress efects code: 0=ignore contact stress effects, 1=include contact stress
c                 effects
c    kdum      =  dummy
c    kody      =  current yield code.
c    kodyold   =  'contact stress' yield code before load reversal.
c    kodyx     =  previous yield code.
c    kst       =  stiffness formation code ( 1: y, 0: n).
c    kstep     =  step no. in this segment
c    lm(4)     =  location matrix.
c    nodj      =  node no. for element end j.
c    pyt       =  tensile yield force.
c    rlp       =  current element length.
c    rlpo      =  original element length.
c    senn      =  envelop value for negative force.
c    senp      =  envelop value for positive force.
c    sep       =  current force.
c    sepb      =  'contact stress' force at load reversal.
c    sepc      =  concrete compressive strength force.
c    sepco     =  the contact force value at zero deformation.
c    sepcs     =  force at the start of contact stress effects.
c    sepcsf    =  force at finishing point of contact stress.
c    sepevt    =  next event force.
c    sepmn     =  minimum force at load reversal.
c    sepyint   =  compressive yield envelope intercept force, initially set to 3/4f'c.
c    sina      =  sine of element inclination angle.
c    st        =  total force for element.
c    stdp      =  damping force for previous step.
c    stp       =  total force for previous step.
c    vbuck     =  total buckling deformation.
c    venn      =  envelop value for negative deformation.
c    venp      =  envelop value for positive deformation.
c    vpacn     =  accumulated negative plastic deformation.
c    vtot      =  total element deformation.
c    vtotc     =  compressive deformation at f'c.
c    vtotcs    =  strain at the starting point of contact stress.
c    vtotcsf   =  strain at the finishing point of contact stress.
c    vtotevt   =  next event deformation.
```

```
c      vtotmn        =  minimum strain value before loading reversal.
c      vtotmx        =  maximum strain value before loading reversal.
c      vtotos        =  strain at zero stress.
c      vtotult       =  ultimate tensile deformation,(set to equal ultimate steel deformation).
c      vtotyc        =  compressive deformation at 3/4f'c.
c      vtotyint      =  compressive yield envelope deformation intercept, set to vtotyc initially.
c      vtotyt        =  positive yield strain.
c -----------------------------------------------------------------
c  LABELLED COMMONS
c   include 'infel12.h'
c   hinge addition - KTD 20/12/93
c   include 'hinge.h'
c   common/tapes/inp,iou
c   common /work/ dsep,fac,factor,facacc,dv,delta,delti,pyy,dum,dvp,
c   1w(1990)
c      delta         =  P-delta/L
c      delti         =  P-delta (increment)/L
c      dsep          =  change in force.
c      dv            =  deformation increment.
c      dvp           =  plastic deformation increment.
c      fac,factor,facac= event factor used to trace out the
c                       non-linear path.
c -----------------------------------------------------------------
c  ARGUMENT DECLARATIONS
      dimension relas(ndof),rdamp(ndof),rinit(ndof),
     1          ddise(ndof),dise(ndof),vele(ndof)
c -----------------------------------------------------------------

c -----------------RESET ENVELOPE STEP VALUES FOR NEW ANALYSIS SEGMENT
      if(kstep.eq.-1)then
       isenn=0
       isenp=0
       ivenn=0
       ivenp=0
       kstep=1
      end if

c hinge addition - KTD 15/12/93
c ------------------------------------READ CURRENT HINGE INFORMATION
      if(ihing.ne.0) then
       call hstart(ihing,igrc)
       if(ihpos.eq.1) then
         rlp=rlpnc
         eff=rlpo/rlpnc
       elseif(ihpos.eq.2) then
         rlp=rlppc
         eff=rlpo/rlppc
       endif
      endif

c --------------------------------CALCULATE EXTENSION AND INCREMENTS
      vtotold=vtot
      dv=cosa*(ddise(3)-ddise(1))+sina*(ddise(4)-ddise(2))
      vtot=vtot+dv
```

```
      revld=dv*dvold

c -----------------------------------UPDATE TOTAL DEFORMATION AND STRAIN

      if(vtot.le.vtotc) then
        write(iou,5)
        write(*,5)
    5   format(/' *** Exceeded compressive yield strain value')
        if(vtot.le.(3.d0*vtotc)) then
          write(iou,10)
          write(*,10)
   10     format(/' *** Exceeded ultimate compressive strain value')
        endif
      endif

c -------------------------------TRACE NON-LINEAR PATH FOR E-P COMPONENT
      facac=0.d0

   20 continue
        factor=1.d0 - facac

c---not yielded
        if(kody.eq.0) then
c---get factor for status change
          pyy=0.d0
          dsep=eff*eale*dv
          ealh=eff*eale
          if(dv.lt.0.d0) then
            if((kodyold.eq.5).or.(kodyold.eq.6)) then
              fac=(sepb-sep)/dsep
              if(fac.lt.factor) then
c---contact stress reyield
                factor=fac
                pyy=sepb
                if(kodyold.eq.5) then
                  sepevt=sepco
                  vtotevt=vtotos
                elseif(kodyold.eq.6) then
                  sepevt=sepcsf
                  vtotevt=vtotcsf
                endif
                kody=kodyold
              endif
            else
              fac=(sepyint-sep)/dsep
              if(fac.lt.factor) then
c---negative yield
                factor=fac
                pyy=sepyint
                if(vtotyint.gt.vtotc) then
                  sepevt=sepc
                  vtotevt=vtotc
                  ealcsh=(sepevt-sepyint)/(vtotevt-vtotyint)
                  kody=1
```

```
             elseif(vtotyint.ge.3.d0*vtotc) then
              sepevt=0.2*sepc
              vtotevt=3.d0*vtotc
              ealcsd=(sepevt-sepyint)/(vtotevt-vtotyint)
              kody=2
             endif
            endif
           endif
          elseif(dv.gt.0.d0) then
           fac=(pyt-sep)/dsep
           if(fac.lt.factor) then
c---positive yield
            factor=fac
            vbuck=0.d0
            pyy=pyt
            if(kodyold.eq.0) then
             vtotyt=vtotold+factor*dv
             vtotos=vtotyt-pyt/(eff*eale)
            elseif((kodyold.eq.5).or.(kodyold.eq.6)) then
             kodyold=0
            endif
            vtotevt=1.d20
            sepevt=pyt+ealc*(vtotevt-vtotyt)
            kody=3
           endif
          endif
c---update and calculate (elastic) energy
          if(pyy.ne.0.d0)then
c---new yield
           if(kenr.gt.0) ener=ener+((sep+pyy)*0.5d0-aint)*factor*dv
           sep=pyy
          else
c---remains elastic
           dsp=factor*dsep
           sep=sep+dsp
           if(kenr.gt.0) ener=ener+(sep-dsp*0.5d0-aint)*factor*dv
          endif

c---yielded in compression ( >3/4fc')
        elseif(kody.eq.1) then
          pyy=0.d0
          dsep=ealcsh*dv
          ealh=ealcsh
          if(dv.lt.0.d0)then
c---continuing to yield
           fac=(sepc-sep)/dsep
           if(fac.lt.factor) then
c---reached concrete compressive strength
            factor=fac
            pyy=sepc
            sepevt=0.2*sepc
            vtotevt=3.d0*vtotc
            ealcsd=(sepevt-sepc)/(vtotevt-vtotc)
            kody=2
```

```
        endif
        dvp=factor*dv
        vpacn=vpacn+dvp
       elseif(dv.gt.0.d0) then
        factor=0.d0
        kody=0
       endif
c---update and calculate energy
       if(pyy.ne.0.d0)then
        if(kenr.gt.0) ener=ener+((sep+pyy)*0.5d0-aint)*factor*dv
        sep=pyy
       else
c---hasn't reached compressive strength
        dsp=factor*dsep
        sep=sep+dsp
        if(kenr.gt.0) ener=ener+(sep-dsp*0.5d0-aint)*factor*dv
       endif

c---yielded in compression ( >fc')
      elseif(kody.eq.2) then
       dsep=ealcsd*dv
       ealh=ealcsd
       if(dv.lt.0.d0)then
c---continuing to yield
        fac=((0.2*sepc)-sep)/dsep
        if(fac.lt.factor) then
c---reached concrete compressive strength
         factor=fac
         pyy=0.2*sepc
         write(iou,15)
         write(*,15)
  15      format(/' *** Exceeded ultimate compressive strain value')
        endif
        dvp=factor*dv
        dvp=factor*dv
        vpacn=vpacn+dvp
        dsp=factor*dsep
        sep=sep+dsp
        if(kenr.gt.0) ener=ener+(sep-dsp*0.5d0-aint)*factor*dv
       elseif(dv.gt.0.d0) then
        factor=0.d0
        kody=0
       endif

c---cracked in tension
      elseif(kody.eq.3) then
       dsep=ealc*dv
       ealh=ealc
       if(dv.gt.0.d0) then
c---continuing to crack
        vbuck=vbuck+factor*dv
        dsp=factor*dsep
        sep=sep+dsp
        if(kenr.gt.0) ener=ener+(sep-dsp*0.5d0-aint)*factor*dv
```

```
        elseif(dv.lt.0.d0) then
c---load reversal
         sepmx=sep
         vtotmx=vtotold
         factor=0.d0
         if(kcont.ne.0) then
          vtotcs=0.6*(vtotmx-vtotyt)+vtotyt
          sepcs=sepmx-ealc*(vtotmx-vtotcs)
          sepevt=sepcs
          vtotevt=vtotcs
         else
          sepevt=pyt
          vtotevt=vtotmx-vbuck
         endif
         kody=4
        endif

c---cracks closing
       elseif(kody.eq.4) then
        pyy=0.d0
        dsep=ealc*dv
        ealh=ealc
        if(dv.lt.0.d0) then
         if(kcont.ne.0) then
          fac=(sepcs-sep)/dsep
          if(fac.lt.factor) then
           factor=fac
           if(rlp.ne.rlpo) then
            vtotc=vtotyc+0.5*(epsc-epsyc)*rlp
            vtotyint=(sepc*(1.d0-vtotc/(4.d0*(vtotc-vtotyc))))/
     1                (eff*eale-sepc/(4.d0*(vtotc-vtotyc)))
            if(vtotyint.le.vtotc) then
             vtotyint=7.d0*sepc/(5.d0*eff*eale+(2.d0*sepc)/vtotc)
            endif
            sepyint=eff*eale*vtotyint
           endif
           sepcsf=0.75*sepyint
           vtotcsf=vtotyint-(sepyint-sepcsf)/(eff*eale)
           ealcs2=(sepcsf-sepco)/(vtotcsf-vtotos)
           vtotcrit=vtotos-sepco/ealcs2
           if(vtotcrit.ge.vtotcs) then
            ealcs2=((sepcsf-sepcs)/(vtotcsf-vtotcs))
            sepevt=sepcsf
            vtotevt=vtotcsf
            kody=6
           elseif(vtotcrit.lt.vtotcs) then
            ealcs1=(sepco-sepcs)/(vtotos-vtotcs)
            sepevt=sepco
            vtotevt=vtotos
            kody=5
           endif
           pyy=sepcs
          endif
         else
```

```
          fac=(pyt-sep)/dsep
          if(fac.lt.factor) then
            factor=fac
            if(rlp.ne.rlpo) then
              vtotc=vtotyc+0.5*(epsc-epsyc)*rlp
              vtotyint=(sepc*(1.d0-vtotc/(4.d0*(vtotc-vtotyc))))/
     1               (eff*eale-sepc/(4.d0*(vtotc-vtotyc)))
              if(vtotyint.le.vtotc) then
                vtotyint=7.d0*sepc/(5.d0*eff*eale+(2.d0*sepc)/vtotc)
              endif
              sepyint=eff*eale*vtotyint
            endif
            sepevt=sepyint
            vtotevt=vtotyint
            kody=0
            pyy=pyt
          endif
        endif
      elseif(dv.gt.0.d0) then
        factor=0.d0
        vtotevt=1.d20
        sepevt=pyt+ealc*(vtotevt-vtotyt)
        kody=3
      endif
      vbuck=vbuck+factor*dv
c---update and calculate energy
      if(pyy.ne.0.d0)then
        if(kenr.gt.0) ener=ener+((sep+pyy)*0.5d0-aint)*factor*dv
        sep=pyy
      else
        dsp=factor*dsep
        sep=sep+dsp
        if(kenr.gt.0) ener=ener+(sep-dsp*0.5d0-aint)*factor*dv
      endif

c---contact stress effects :first stage
    elseif(kody.eq.5) then
      pyy=0.d0
      dsep=ealcs1*dv
      ealh=ealcs1
      if(dv.lt.0.d0) then
        fac=(sepco-sep)/dsep
        if(fac.lt.factor) then
          factor=fac
          pyy=sepco
          sepevt=sepcsf
          vtotevt=vtotcsf
          kody=6
        endif
      elseif(dv.gt.0.d0) then
        sepb=sep
        factor=0.d0
        kodyold=kody
        kody=0
```

```
      endif
c---update and calculate energy
      if(pyy.ne.0.d0) then
       if(kenr.gt.0) ener=ener+((sep+pyy)*0.5d0-aint)*factor*dv
       sep=pyy
      else
       dsp=factor*dsep
       sep=sep+dsp
       if(kenr.gt.0) ener=ener+(sep-dsp*0.5d0-aint)*factor*dv
      endif

c---contact stress effects: second stage
     elseif(kody.eq.6) then
      pyy=0.d0
      dsep=ealcs2*dv
      ealh=ealcs2
      if(dv.lt.0.d0) then
       fac=(sepcsf-sep)/dsep
       if(fac.lt.factor) then
        factor=fac
        pyy=sepcsf
        sepevt=sepyint
        vtotevt=vtotyint
        kodyold=0
       endif
      elseif(dv.gt.0.d0) then
       sepb=sep
       factor=0.d0
       kodyold=kody
       kody=0
      endif
c---update and calculate energy
      if(pyy.ne.0.d0) then
       if(kenr.gt.0) ener=ener+((sep+pyy)*0.5d0-aint)*factor*dv
       sep=pyy
       kody=0
      else
       dsp=factor*dsep
       sep=sep+dsp
       if(kenr.gt.0) ener=ener+(sep-dsp*0.5d0-aint)*factor*dv
      endif
     endif

c ------------------------------------------CHECK COMPLETION OF CYCLE
     facac=facac+factor
    if(facac.lt.0.999999d0) go to 20

    if(ealh.lt.0.d0) then
     write(iou,25)imem,kody
     write(*,25)imem,kody
  25 format(/' *** Negative stiffness,'
   1              ' member = ',i5,2x,'yield code = ',i5)
     stop
    endif
```

```
c --------------------------------------------UPDATE TOTAL STATIC FORCE
c --save current force for second order energy
      stt=st
      st=sep

c---------------------------------------------------------------------
      if(ihing.ne.0) then
        if(ihpos.eq.1) then
          ealct=ealh
          sepct=st
          if(((kody.eq.0).and.(vtotold.eq.0.d0)).or.((kody.eq.0).and.
     1                              (revld.lt.0.d0))) then
            if(dv.gt.0.d0) then
              sepctevt=pyt
              vtotctevt=vtotyt
            elseif(dv.lt.0.d0) then
              sepctevt=sepyint
              vtotctevt=vtotyc
            endif
          elseif((kody.ne.kodyx).or.(revld.lt.0.d0)) then
            sepctevt=sepevt
            vtotctevt=vtotevt
          endif
        elseif(ihpos.eq.2) then
          ealcb=ealh
          sepcb=st
          if(((kody.eq.0).and.(vtotold.eq.0.d0)).or.((kody.eq.0).and.
     1                              (revld.lt.0.d0))) then
            if(dv.gt.0.d0) then
              sepcbevt=pyt
              vtotcbevt=vtotyt
            elseif(dv.lt.0.d0) then
              sepcbevt=sepyint
              vtotcbevt=vtotyc
            endif
          elseif((kody.ne.kodyx).or.(revld.lt.0.d0)) then
            sepcbevt=sepevt
            vtotcbevt=vtotevt
          endif
        endif
        call hfinish(ihing,igrc)
      endif
      dvold=dv

c ----------------------------------E-COMPONENT AND SECOND ORDER ENERGY
      if(kenr.gt.0) then
c---second order energy (based on static force only)
        if(kgem.gt.0)then
          delta=(sina*(dise(3)-dise(1))+cosa*(dise(2)-dise(4)))/rlp
          delti=(sina*(ddise(3)-ddise(1))+cosa*(ddise(2)-ddise(4)))/rlp
          enso=enso-((st+stt)*0.5)*(delta-delti*0.5)*delti*rlp
        end if
      end if
```

```
c ---------------------------------------------STATIC RESISTING FORCES
   call elas12(relas,rinit,dise,ndof,kgem)

c -------------------------------------------DAMPING FORCES AND ENERGY
   if(kresis.eq.2) call damp12(rdamp,vele,kenr,ndof,beto,ened,dv)

c ----------------------------------------------------ENVELOPE VALUES
   call upen12(kstep)

c -----------------------------------------------------SAVE RESULTS
   if(ksave.eq.1) call save12

c ----------------------------------SET INDICATOR FOR STIFFNESS CHANGE
   if (kodyx.ne.kody) kst=1

c ------------------------------------------------------------------
   RETURN
   END
```

B3 Shear Element Response Subroutines

```
c ***********************************************************************
      SUBROUTINE RESP15(kresis,ksave,kgem,kstep,ndof,kst,kenr,ener,
     1              ened,enso,tim,beto,relas,rdamp,rinit,combe,
     2              ddise,dise,vele,igrc)
c ***********************************************************************
c SHEAR ELEMENT
c   Element response. Update element state, form static and
c   damping resisting forces, perform energy calculations, update
c   damage measures, and put element results in /THELM/ for saving
c   or printing.
c ------------------------------------------------------------------
c DOUBLE PRECISION / LARGE
      include 'double.h'
c ------------------------------------------------------------------
c  CALLED FROM : respxx (once for each sub-step).
c ------------------------------------------------------------------
c  INPUT
c        beto        =  initial stiffness damping factor.
c        ddise(ndof) =  element nodal incremental displacement vector.
c        ddis(ndof)  =  element total nodal displacement vector.
c        kresis      =  indicator for calculating resisting forces
c                       ( 1: static only, 2: static and damping).
c        ksave       =  indicator for saving element results
c                       (1: y, 0: n).
c        kgem        =  second order analysis code (> 0: y, 0: n).
c        kenr        =  energy calculation indicator
c                       (2: static + dynamic, 1: static, 0: none).
c        ndof        =  no. of element DOF.
c        tim         =  time for current step.
c        vele(ndof)  =  element nodal velocity vector.
c  OUTPUT
c        ener        =  change of element elasto-plastic energy.
c        ened        =  change of element damping energy.
c        enso        =  change of element second-order energy.
c        relas(ndof) =  element static resisting force vector.
c        rdamp(ndof) =  element damping resisting force vector.
c        rinit(ndof) =  element initial resisting force vector.
c  MODIFY
c        Asb         =  area of bottom longitudinal reinforcement.
c        Ast         =  area of top longitudinal reinforcement.
c        Av          =  area of shear reinforcement
c        bot         =  distance from the beam centre line to the centroid of thelongitudinal steel in the
c                       bottom half of the beam.
c        cosa        =  cosine of element inclination angle (from end ito end j, anti-clockwise).
c        delta       =  P-delta/L
c        delti       =  P-delta (increment)/L
c        dsep        =  change in shear force
c        dtheta      =  rotation increment.
c        dv          =  shear deformation increment.
```

```
c     dvin()     = change in inelastic shear deformation in the negative hinge since last inelastic load
c                  reversal.
c     dvip()     = change in inelastic shear deformation in the positive hinge since last inelastic load
c                  reversal.
c     dvold      = previous deformation increment.
c     eal        = EA/L.
c     ealv(0:16) = different EA/L values for variou stages of the hysteresis loop.
c     ealx       = previous EA/L.
c     epsvh      = stirrup strain at initiation of strain hardening.
c     epsvu      = stirrup ultimate strain.
c     epsvy      = stirrup yield strain.
c     Evb        = inelastic stirrup stress-strain modulus (to 6% strain).
c     Evs        = Young's Modulus for the stirrups.
c     fac,factor,facac= event factor used to trace out thenon-linear path.
c     fl         = length of element.
c     fn         = longitudinal projection of the negative moment plastic hinge diagonal cracks.
c     fp         = longitudinal projection of the positive moment plastic hinge diagonal cracks.
c     fpconst    = factor allowing for the affect of the axial load on the diagonal concrete compression
c                  force.
c     fvu        = stirrup ultimate stress.
c     fvy        = stirrup yield stress.
c     fyb        = yield stress of bottom longitudinal reinforcement.
c     fyt        = yield stress of top longitudinal reinforcement.
c     ihing      = hinge number, 0:not in a hinge.
c     imem       = element no.
c     isenn      = step no. for senn.
c     isenp      = step no. for senp.
c     ivenn      = step no. for venn.
c     ivenp      = step no. for venp.
c     kelas      = code indicating whether the current hysteresis loop is a 0:inelastic or 1:elastic
c                  reversal.
c     kody       = current yield code.
c     kodyx      = previous yield code.
c     kst        = stiffness formation code ( 1: y, 0: n).
c     kstep      = step no. in this segment
c     levent     = code to indicate if a longitudinal steel or concrete hinge element event has occurred
c                  in the current step; 1=yes 0=no.
c     lm(6)      = location matrix.
c     nodj       = node no. for element end j.
c     psicrtn    = critical curvature, above which yielding in the first stirrup of the negative truss
c                  occurs.
c     psicrtp    = critical curvature, above which yielding in the first stirrup of the positive truss
c                  occurs.
c     psimnn     = minimum curvature in the negative hinge.
c     psimnp     = minimum curvature in the positive hinge.
c     rlp        = current hinge length.
c     rlpevt     = plastic hinge length at the next longitudinal hinge element event.
c     rlpnv      = current length of the negative moment plastic hinge.
c     rlppv      = current length of the positive moment plastic hinge.
c     rM         = current maximum hinge moment.
c     rMh        = moment occurring at the hinge sub-structure,
c     rMevt      = maximum moment at the next longitudinal hinge element event.
c     rMVc       = current moment to shear ratio for the associated plastic hinge.
c     rMycn      = negative yield moment.
```

```
c     rMycp      =  positive yield moment.
c     s          =  stirrup spacing.
c     selng      =  shear deformation due to elongation of the hinge reinforcement.
c     senn       =  envelop value for negative force.
c     senp       =  envelop value for positive force.
c     sep        =  current shear force.
c     sep1       =  shear force at load reversal from inelastic stiffness
c     sepcold    =  concrete diagonal compression force at beginning of step.
c     sepcs      =  shear force at the intersection of the reloading shearslop stiffness slope and the post
                    yield elastic loading line.
c     sepevt     =  shear force at the next yield event.
c     sepint     =  shear force at the intersection of the contact stress
c     sepmne     =  minimum contact stress shear force in the previous cycle.
c     sepmxe     =  maximum contact stress shear force in the previous cycle.
c     sepslop    =  shear force at zero shear deformation.
c     sepycn     =  positive 'yield' shear force.
c     sepycp     =  negative 'yield' shear force.
c     sina       =  sine of element inclination angle.
c     sl         =  stirrup leg length.
c     st         =  shear force at current load step.
c     stdp       =  damping force for previous step.
c     stp        =  total force for previous step.
c     stt        =  shear force at previous load step.
c     talpha     =  average of talphan and talphap.
c     talphan    =  tan(angle of the negative moment hinge diagonal compressive force).
c     talphap    =  tan(angle of the positive moment hinge diagonal compressive force).
c     theta      =  current hinge rotation.
c     thetaevt   =  total rotation at the next longitudinal hinge element event.
c     thetar     =  angle at which there is a shear force sign reversal from an inelastic deformation.
c     thetard    =  dummy angle at which there is a shear force sign reversal.
c     thetarn    =  angle at which there is a shear force sign reversal from a yielding positive moment
c                   hinge.
c     thetarp    =  angle at which there is a shear force sign reversal from a yielding negative moment
                    hinge.
c     top        =  distance from the beam centre line to the centroid of the longitudinal steel in the
c                   top half of the beam.
c     vcevt      =  extension of the hinge compression zone at the next longitudinal hinge element
c                   event.
c     ven(0)     =  total elastic shear deformation in the negative hinge.
c     venn       =  envelop value for negative deformation.
c     venp       =  envelop value for positive deformation.
c     vep(0)     =  total elastic shear deformation in the positive hinge.
c     vin(0)     =  total inelastic shear deformation in the negative hinge.
c     vint       =  shear deformation at the intersection of the contact stress line and the post yield
c                   elastic loading line.
c     vip(0)     =  total inealstic shear deformation in the positive hinge.
c     vss        =  shear deformation at zero shear force.
c     vtot       =  total shear deformation.
c     vtotevt    =  shear deformation at the next yield event.
c     vy         =  'yield' shear deformation.
c -------------------------------------------------------------------
c  LABELLED COMMONS
      include 'infel15.h'
      include 'hinge.h'
```

```
      include 'eqns.h'
      include 'event.h'
      common/tapes/inp,iou
      common/work/fac,factor,facacc,dv,delta,delti,w(1994)

c -----------------------------------------------------------------
c  ARGUMENT DECLARATIONS
      dimension relas(ndof),rdamp(ndof),rinit(ndof),combe(neqq),
     1          ddise(ndof),dise(ndof),vele(ndof)
c -----------------------------------------------------------------

c -----------------RESET ENVELOPE STEP VALUES FOR NEW ANALYSIS SEGMENT
      if(kstep.eq.-1)then
       isenn=0
       isenp=0
       ivenn=0
       ivenp=0
       kstep=1
       sepcold=0.d0
      end if

c ------------------------------------READ CURRENT HINGE INFORMATION
      call hstart(ihing,igrc)

c ----------------------------------------------DEFLECTION INCREMENT
      dv=cosa*(ddise(4)-ddise(2))-sina*(ddise(3)-ddise(1))
      dtheta=((vtotb-vtott)/d_dd)-theta
      dsepc=ealv(kody)*dv

c ------------------------------------------UPDATE TOTAL DEFORMATION
      revld=dv*dvold
      vtot=vtot+dv
      dvold=dv
      rMold=rM
      if(rMold.eq.0.d0) then
       if(dtheta.gt.0.d0) then
        rMVc=rMVp
       elseif(dtheta.lt.0.d0) then
        rMVc=rMVn
       endif
       rMVcn=rMVc
      elseif(rMold.ne.0.d0) then
       rMVcn=dabs(rMold/sep)
      endif

c ----------------------UPDATE TOTAL ROTATION AND MOMENT ACROSS HINGE
      thetaold=theta
      theta=theta+dtheta
c calculate the position of the hinge, the moment occurring at the hinge
c sub-structure, then the maximum moment
      top=(db-dt+d_dd)/2.d0
      bot=(dt-db+d_dd)/2.d0
      rMh=(seprb+sepcb)*bot-(seprt+sepct)*top
      rMVpos=rMVcn-((fn/2.d0+fp/2.d0)/2.d0)
```

```
      rM=rMh+(rMh/rMVpos)*(fn/2.d0+fp/2.d0)/2.d0
      if((dabs((rM-rMh)/rMh).gt.1.d0).and.((dabs(rM).lt.
     1   dabs(0.75d0*rMycn)).or.(dabs(rM).lt.dabs(0.75d0*rMycp)))) then
        rMVpos=0.d0
        rM=rMh
      else
        rMVc=rMVcn
      endif

c ------------------------------TRACE NON-LINEAR PATH FOR E-P COMPONENT
      facac=0.d0

   10 continue
        factor=1.d0 - facac

c---not yielded
c---elastic tensile loading
        if(kody.eq.0) then
c---get factor for status change
          pyy=0.d0
          dsep=ealv(kody)*dv
          if(dsep.gt.0.d0) then
            if(sepycp.eq.0.d0) then
              fac=1.d20
            elseif(sepycp.ne.0.d0) then
              fac=(sepycp-sep)/dsep
            endif
            if(fac.lt.factor) then
c---tension yield
              pyy=sepycp
              if(ealv(0).eq.ealv(1)) then
                sepycn=-pyy
              else
                sepycn=-fn/fp*pyy
              endif
              kody=6
c---determine the next hinge element event
c   the axial load level is assumed to remain constant up until the next event
              call nhevnt(rlpevt,rMVc,rMevt,rMVpos,
     1                    sep,sepevt,thetaevt,vcevt)
c---calculate the shear deformation level at the next hinge element event
              krev=0
              call stpstr(igrc,krev,rMevt,rlpevt,thetaevt,
     1                    vcevt,vtotevt)
c Because the moment to shear ratio is assumed to remain at the current
c level, at the next event, where in reality it changes, it is possible
c that the shear force at the next event will be less than the current
c shear force. Also because of the changing hinge length the elastic shear
c stiffness is only an estimate consequently, it is possible that the shear
c deformation at the next event will be less than the current shear
c deformation
c -these situations are computationally impossible, hence the addition of
c these statements
              if(dabs(sepevt).lt.dabs(sep)) then
```

```
          write(iou,20)sep
 20         format(' ***ERROR - sep gt sepevt - sep='f10.0)
          sepevt=pyy*(rMVc/(rMVc-0.002))
         endif
         if(dabs(vtotevt).lt.dabs(vtot)) then
          write(iou,25)sep
 25         format(' ***ERROR - vtot gt vtotevt - sep='f10.0)
          vtotevt=vtot*(rMVc/(rMVc-0.002))
         endif
         ealv(6)=(sepevt-pyy)/(vtotevt-(vtot-(1.d0-factor)*dv))
        endif
       elseif(dsep.lt.0.d0) then
        fac=(-0.5d-8-sep)/dsep
        if(fac.lt.factor) then
c---elastic compression loading
         factor=fac
         pyy=-0.5d-8
         kody=1
        endif
       endif
c---update and calculate (elastic) energy
       if(pyy.ne.0.d0)then
        if(kenr.gt.0) ener=ener+((sep+pyy)*0.5d0-aint)*factor*dv
        sep=pyy
       else
        dsp=factor*dsep
        sep=sep+dsp
        if(kenr.gt.0) ener=ener+(sep-dsp*0.5d0-aint)*factor*dv
       endif

c---not yielded
c---elastic compression loading
      elseif(kody.eq.1) then
c---get factor for status change
       pyy=0.d0
       dsep=ealv(kody)*dv
       if(dsep.lt.0.d0) then
        if(sepycn.eq.0.d0) then
         fac=1.d20
        elseif(sepycn.ne.0.d0) then
         fac=(sepycn-sep)/dsep
        endif
        if(fac.lt.factor) then
c---compression yield
         pyy=sepycn
         if(ealv(0).eq.ealv(1)) then
          sepycp=dabs(pyy)
         else
          sepycp=fp/fn*dabs(pyy)
         endif
         kody=6
c---determine the next hinge element event
         call nhevnt(rlpevt,rMVc,rMevt,rMVpos,
  1                        sep,sepevt,thetaevt,vcevt)
```

```
c---calculate the shear deformation level at the next hinge element event
          krev=0
          call stpstr(igrc,krev,rMevt,rlpevt,thetaevt,
     1                                      vcevt,vtotevt)
          if(dabs(sepevt).lt.dabs(sep)) then
           write(iou,30)sep
 30         format(' ***ERROR - sep gt sepevt - sep='f10.0)
           sepevt=pyy*(rMVc/(rMVc-0.002))
          endif
          if(dabs(vtotevt).lt.dabs(vtot)) then
           write(iou,35)sep
 35         format(' ***ERROR - vtot gt vtotevt - sep='f10.0)
           vtotevt=vtot*(rMVc/(rMVc-0.002))
          endif
          ealv(6)=(sepevt-pyy)/(vtotevt-(vtot-(1.d0-factor)*dv))
         endif
        elseif(dsep.gt.0.d0) then
         fac=(0.5d-8-sep)/dsep
         if(fac.lt.factor) then
c---elastic tension loading
          factor=fac
          pyy=0.5d-8
          kody=0
         endif
        endif
c---update and calculate (elastic) energy
        if(pyy.ne.0.d0)then
         if(kenr.gt.0) ener=ener+((sep+pyy)*0.5d0-aint)*factor*dv
         sep=pyy
        else
         dsp=factor*dsep
         sep=sep+dsp
         if(kenr.gt.0) ener=ener+(sep-dsp*0.5d0-aint)*factor*dv
        endif

c---inelastic loading
       elseif(kody.ge.6) then
        pyy=0.d0
        dsep=ealv(kody)*dv
        if(sep*dsep.gt.0.d0) then
         if(dsep.gt.0.d0) then
          if(levent.eq.1) then
           pyy=sep+factor*dsep
           kody=kody+1
c---determine the next hinge element event
           call nhevnt(rlpevt,rMVc,rMevt,rMVpos,
     1                              sep,sepevt,thetaevt,vcevt)
c---calculate the shear deformation level at the next hinge element event
           krev=0
           call stpstr(igrc,krev,rMevt,rlpevt,thetaevt,
     1                                      vcevt,vtotevt)
           if(dabs(sepevt).lt.dabs(sep)) then
            write(iou,40)sep
 40          format(' ***ERROR - sep gt sepevt - sep='f10.0)
```

```
            sepevt=pyy*(rMVc/(rMVc-0.002))
           endif
c  in some situations when the stiffness of the compression zone elements
c  is less than that of the tension zone elements, the length of the
c  compression zone will reduce at a faster rate than the tension zone
c  causing the length of the beam centre-line to reduce as the shear element
c  moves along its yielding path therefore the total shear deformation
c  can actually reduce in some situations.  Also if the beam is undergoing
c  unsymmetrical loading the elastic extensions that occur in each half
c  cycle will be different but the theory uses the elastic extension that
c  occurred in the previous half cycle to calculate the elastic reloading
c  stiffness, this can lead to much greater shear deformations than the
c  theory suggests.
c  Computationally these situations are not possible hence the
c  addition of this statement.
           if(dabs(vtotevt).lt.dabs(vtot)) then
            write(iou,50)vtot
 50          format(' ***ERROR - vtot gt vtotevt - vtot='f10.0)
            vtotevt=vtot*(rMVc/(rMVc-0.002))
           endif
           ealv(kody)=(sepevt-pyy)/(vtotevt-(vtot-((1.d0-factor)
  1                                            *dv)))
          endif
         elseif(dsep.lt.0.d0) then
          if(levent.eq.1) then
           pyy=sep+factor*dsep
           kody=kody+1
c---determine the next hinge element event
           call nhevnt(rlpevt,rMVc,rMevt,rMVpos,
  1                         sep,sepevt,thetaevt,vcevt)
c---calculate the shear deformation level at the next hinge element event
           krev=0
           call stpstr(igrc,krev,rMevt,rlpevt,thetaevt,
  1                                      vcevt,vtotevt)
           if(dabs(sepevt).lt.dabs(sep)) then
            write(iou,60)sep
 60          format(' ***ERROR - sep gt sepevt - sep='f10.0)
            sepevt=pyy*(rMVc/(rMVc-0.002))
           endif
           if(dabs(vtotevt).lt.dabs(vtot)) then
            write(iou,70)sep
 70          format(' ***ERROR - vtot gt spvtot - sep='f10.0)
            vtotevt=vtot*(rMVc/(rMVc-0.002))
           endif
           ealv(kody)=(sepevt-pyy)/(vtotevt-(vtot-((1.d0-factor)
  1                                            *dv)))
          endif
         endif
        elseif(sep*dsep.lt.0.d0) then
         factor=0.d0
         krev=1
         if(sep.gt.0.5d-8) then
          if(rM.gt.0.d0) then
           rlpold=rigidf*(rigid+fp+((seprb-pyp)*d_dd/dabs(sep)))
```

```
            if(rigidf.eq.2.d0) then
              if((rlpold.lt.rlppv).or.(rlpold.gt.4.d0*d_dd)) then
                rlpold=rlppv
              endif
            elseif(rigidf.ne.2.d0) then
              if((rlpold.lt.rlppv).or.(rlpold.gt.2.d0*d_dd)) then
                rlpold=rlppv
              endif
            endif
          elseif(rM.lt.0.d0) then
            rlpold=rigidf*(rigid+fn+((seprt-pyn)*d_dd/dabs(seps)))
            if(rigidf.eq.2.d0) then
              if((rlpold.lt.rlpnv).or.(rlpold.gt.4.d0*d_dd)) then
                rlpold=rlpnv
              endif
            elseif(rigidf.ne.2.d0) then
              if((rlpold.lt.rlpnv).or.(rlpold.gt.2.d0*d_dd)) then
                rlpold=rlpnv
              endif
            endif
          endif
        elseif(sep.lt.-0.5d-8) then
          if(rM.gt.0.d0) then
            rlpold=rigidf*(rigid+fp+((seprb-pyp)*d_dd/dabs(seps)))
            if(rigidf.eq.2.d0) then
              if((rlpold.lt.rlppv).or.(rlpold.gt.4.d0*d_dd)) then
                rlpold=rlppv
              endif
            elseif(rigidf.ne.2.d0) then
              if((rlpold.lt.rlppv).or.(rlpold.gt.2.d0*d_dd)) then
                rlpold=rlppv
              endif
            endif
          elseif(rM.lt.0.d0) then
            rlpold=rigidf*(rigid+fn+((seprt-pyn)*d_dd/dabs(seps)))
            if(rigidf.eq.2.d0) then
              if((rlpold.lt.rlpnv).or.(rlpold.gt.4.d0*d_dd)) then
                rlpold=rlpnv
              endif
            elseif(rigidf.ne.2.d0) then
              if((rlpold.lt.rlpnv).or.(rlpold.gt.2.d0*d_dd)) then
                rlpold=rlpnv
              endif
            endif
          endif
        endif
        vcevt=vtotb+(vtott-vtotb)*bot/d_dd
c---calculate the stirrup extensions
        call stpstr(igrc,krev,rMold,rlpold,thetaold,vcevt,vtotdum)
        sep1=sep
        kody1=kody
        if(dtheta.gt.0.d0) ealv(2)=dabs(sep)/ven(0)
        if(dtheta.lt.0.d0) ealv(2)=dabs(sep)/vep(0)
        kelas=0
```

```
            revld=1.d0
            kody=2
          endif
c---update and calculate energy
          if(pyy.ne.0.d0)then
c---new yield
            if(kenr.gt.0) ener=ener+((sep+pyy)*0.5d0-aint)*factor*dv
            sep=pyy
          else
c---remains elastic
            dsp=factor*dsep
            sep=sep+dsp
            if(kenr.gt.0) ener=ener+(sep-dsp*0.5-aint)*factor*dv
          endif

c---unloading
        elseif(kody.eq.2) then
          pyy=0.d0
          if(revld.lt.0.d0) then
            if(kelas.eq.0) then
              kelas=1
            elseif(kelas.eq.1) then
              kelas=0
            endif
            revld=1.d0
          endif
          dsep=ealv(2)*dv
          if(sep*dsep.lt.0.d0) then
            if(dsep.lt.0.d0) then
              fac=(-0.5d-8-sep)/dsep
              if(fac.lt.factor) then
c---negative shear slop
                factor=fac
                pyy=-0.5d-8
                vss=vtot-(1.d0-factor)*dv
                if(kelas.eq.0) then
c---load reversal occurred from positive 'yielding' load
                  if(dtheta.gt.0.d0) then
                    ealv(3)=-sepslop/(-(vin(0)+selng))
                    vint=vss-(vin(0)+selng)-0.4*(vip(0)+selng)
                  elseif(dtheta.lt.0.d0) then
                    ealv(3)=-sepslop/(-(vip(0)+selng))
                    vint=vss-(vip(0)+selng)-0.4*(vin(0)+selng)
                  endif
                elseif(kelas.eq.1) then
c---load reversal occurred from positive elastic load
                  ealv(3)=(-sepslop-pyy)/(0.d0-vss)
                  if(dtheta.gt.0.d0) then
                    vint=-0.4*(vip(0)+selng)
                  elseif(dtheta.lt.0.d0) then
                    vint=-0.4*(vin(0)+selng)
                  endif
                endif
                sepint=pyy+ealv(3)*(vint-vss)
```

```
         if(dtheta.gt.0.d0) then
           ealv(4)=(sepycn-sepint)/(-(vip(0)+selng+vep(0))-vint)
           if((ealv(3).gt.ealv(4)).or.(ealv(3).lt.0.d0)) then
             ealv(4)=(sepycn-pyy)/(-(vip(0)+selng+vep(0))-vss)
             kody=4
           else
             kody=3
           endif
           thetarp=theta-(1.d0-factor)*dtheta
         elseif(dtheta.lt.0.d0) then
           ealv(4)=(sepycn-sepint)/(-(vin(0)+selng+ven(0))-vint)
           if((ealv(3).gt.ealv(4)).or.(ealv(3).lt.0.d0)) then
             ealv(4)=(sepycn-pyy)/(-(vin(0)+selng+ven(0))-vss)
             kody=4
           else
             kody=3
           endif
           thetarn=theta-(1.d0-factor)*dtheta
         endif
       endif
     elseif(dsep.gt.0.d0) then
       fac=(0.5d-8-sep)/dsep
       if(fac.lt.factor) then
c---positive shear slop
         factor=fac
         pyy=0.5d-8
         vss=vtot-(1.d0-factor)*dv
         if(kelas.eq.0) then
c---load reversal occurred from negative 'yielding' load
           if(dtheta.gt.0.d0) then
             ealv(3)=sepslop/(vin(0)+selng)
             vint=vss+(vin(0)+selng)+0.4*(vip(0)+selng)
           elseif(dtheta.lt.0.d0) then
             ealv(3)=sepslop/(vip(0)+selng)
             vint=vss+(vip(0)+selng)+0.4*(vin(0)+selng)
           endif
         elseif(kelas.eq.1) then
c---load reversal occurred from negative elastic load
           ealv(3)=(sepslop-pyy)/(0.d0-vss)
           if(dtheta.gt.0.d0) then
             vint=0.4*(vip(0)+selng)
           elseif(dtheta.lt.0.d0) then
             vint=0.4*(vin(0)+selng)
           endif
         endif
         sepint=pyy+ealv(3)*(vint-vss)
         if(dtheta.gt.0.d0) then
           ealv(4)=(sepycp-sepint)/((vip(0)+selng+vep(0))-vint)
           if((ealv(3).gt.ealv(4)).or.(ealv(3).lt.0.d0)) then
             ealv(4)=(sepycp-pyy)/((vip(0)+selng+vep(0))-vss)
             kody=4
           else
             kody=3
           endif
```

```
                thetarp=theta-(1.d0-factor)*dtheta
              elseif(dtheta.lt.0.d0) then
                ealv(4)=(sepycp-sepint)/((vin(0)+selng+ven(0))-vint)
                if((ealv(3).gt.ealv(4)).or.(ealv(3).lt.0.d0)) then
                  ealv(4)=(sepycp-pyy)/((vin(0)+selng+ven(0))-vss)
                  kody=4
                else
                  kody=3
                endif
                thetarn=theta-(1.d0-factor)*dtheta
              endif
            endif
          endif
        elseif(sep*dsep.gt.0.d0) then
          fac=(sep1-sep)/dsep
          if(fac.lt.factor) then
            factor=fac
            pyy=sep1
            kody=kody1
          endif
        endif
c---update and calculate energy
        if(pyy.ne.0.d0)then
c---stiffness change
          if(kenr.gt.0) ener=ener+((sep+pyy)*0.5d0-aint)*factor*dv
          sep=pyy
        else
c---stiffness remains constant
          dsp=factor*dsep
          sep=sep+dsp
          if(kenr.gt.0) ener=ener+(sep-dsp*0.5-aint)*factor*dv
        endif

      elseif(kody.eq.3) then
c---contact stress effects
        pyy=0.d0
        dsep=ealv(3)*dv
        if(revld.gt.0.d0) then
          if(sep*sepint.lt.0.d0) then
c---included to enable combe to be updated - is not a true event
            if(dsep.gt.0.d0) then
              fac=(0.5d-8-sep)/dsep
              if(fac.lt.factor) then
                factor=fac
                pyy=0.5d-8
                kst=1
              endif
            elseif(dsep.lt.0.d0) then
              fac=(-0.5d-8-sep)/dsep
              if(fac.lt.factor) then
                factor=fac
                pyy=-0.5d-8
                kst=1
              endif
```

```
         endif
        elseif(sep*sepint.gt.0.d0) then
         fac=(sepint-sep)/dsep
         if(fac.lt.factor) then
c---reloading
          factor=fac
          pyy=sepint
          vint=vtot-(1.d0-factor)*dv
          if(dsep.gt.0.d0) then
           if(dtheta.gt.0.d0) then
            ealv(4)=(sepycp-sepint)/((vip(0)+selng+vep(0))-vint)
           elseif(dtheta.lt.0.d0) then
            ealv(4)=(sepycp-sepint)/((vin(0)+selng+ven(0))-vint)
           endif
          elseif(dsep.lt.0.d0) then
           if(dtheta.gt.0.d0) then
            ealv(4)=(sepycn-sepint)/
     1                       (-(vip(0)+selng+vep(0))-vint)
           elseif(dtheta.lt.0.d0) then
            ealv(4)=(sepycn-sepint)/
     1                       (-(vin(0)+selng+ven(0))-vint)
           endif
          endif
          kody=4
         endif
        endif
       elseif(revld.lt.0.d0) then
        factor=0.d0
        if(kelas.eq.0) then
         kelas=1
        elseif(kelas.eq.1) then
         kelas=0
        endif
        revld=1.d0
        ealv(5)=ealv(2)
        x=2.d0*sepslop/(ealv(5)-ealv(3))
        if(dsep.gt.0.d0) then
c---unloading from negative shear slop portion
         sepmne=sep
         sepcs=sepmne+ealv(5)*x
        elseif(dsep.lt.0.d0) then
c---unloading from positive shear slop portion
         sepmxe=sep
         sepcs=sepmxe-ealv(5)*x
        endif
        kody=5
       endif
c---update and calculate energy
       if(pyy.ne.0.d0)then
c---stiffness change
        if(kenr.gt.0) ener=ener+((sep+pyy)*0.5d0-aint)*factor*dv
        sep=pyy
       else
c---stiffness remains constant
```

```
         dsp=factor*dsep
         sep=sep+dsp
         if(kenr.gt.0) ener=ener+(sep-dsp*0.5-aint)*factor*dv
        endif

c---post yield elastic loading
      elseif(kody.eq.4) then
c---get factor for status change
        pyy=0.d0
        dsep=ealv(4)*dv
        if(revld.gt.0.d0) then
         if(sep.gt.0.d0) then
          fac=(sepycp-sep)/dsep
         elseif(sep.lt.0.d0) then
          fac=(sepycn-sep)/dsep
         endif
         if(fac.lt.factor) then
c---yield
          factor=fac
          pyy=sep+factor*dsep
c---determine the next hinge element event
          call nhevnt(rlpevt,rMVc,rMevt,rMVpos,
     1                       sep,sepevt,thetaevt,vcevt)
          if(dtheta.gt.0.d0) thetar=thetarp
          if(dtheta.lt.0.d0) thetar=thetarn
c---calculate the shear deformation level at the next hinge element event
          krev=0
          call stpstr(igrc,krev,rMevt,rlpevt,thetaevt,
     1                           vcevt,vtotevt)
          if(dabs(sepevt).lt.dabs(sep)) then
           write(iou,120)sep
 120        format(' ***ERROR - sep gt sepevt - sep='f10.0)
           sepevt=pyy*(rMVc/(rMVc-0.002))
          endif
          if(dabs(vtotevt).lt.dabs(vtot)) then
           write(iou,130)sep
 130        format(' ***ERROR - vtot gt spvtot - sep='f10.0)
           vtotevt=vtot*(rMVc/(rMVc-0.002))
          endif
          kody=6
          ealv(kody)=(sepevt-pyy)/(vtotevt-(vtot-((1.d0-factor)
     1                                      *dv)))
         endif
        elseif(revld.lt.0.d0) then
         factor=0.d0
         if(kelas.eq.0) then
          kelas=1
         elseif(kelas.eq.1) then
          kelas=0
         endif
         revld=1.d0
         sep1=sep
         kody1=kody
         if(dsep.gt.0.d0) then
```

```
          if(dtheta.gt.0.d0) ealv(2)=dabs(sepycn)/ven(0)
          if(dtheta.lt.0.d0) ealv(2)=dabs(sepycn)/vep(0)
        elseif(dsep.lt.0.d0) then
          if(dtheta.gt.0.d0) ealv(2)=dabs(sepycp)/ven(0)
          if(dtheta.lt.0.d0) ealv(2)=dabs(sepycp)/vep(0)
        endif
        kody=2
      endif
c---update and calculate (elastic) energy
      if(pyy.ne.0.d0)then
c---stiffness change
        if(kenr.gt.0) ener=ener+((sep+pyy)*0.5d0-aint)*factor*dv
        sep=pyy
      else
c---stiffness remains constant
        dsp=factor*dsep
        sep=sep+dsp
        if(kenr.gt.0) ener=ener+(sep-dsp*0.5-aint)*factor*dv
      endif

c---post yield elastic contact stress loading
    elseif(kody.eq.5) then
c---get factor for status change
      pyy=0.d0
      if(revld.lt.0.d0) then
        if(kelas.eq.0) then
          kelas=1
        elseif(kelas.eq.1) then
          kelas=0
        endif
        revld=1.d0
      endif
      dsep=ealv(5)*dv
      if(sepcs.lt.sep) then
        if(dsep.gt.0.d0) then
          if(sep*sepmxe.lt.0.d0) then
c---included to enable combe to be updated - is not a true event
            fac=(0.5d-8-sep)/dsep
            if(fac.lt.factor) then
              factor=fac
              pyy=0.5d-8
              kst=1
            endif
          elseif(sep*sepmxe.gt.0.d0) then
            fac=(sepmxe-sep)/dsep
            if(fac.lt.factor) then
c---positive shear slop
              factor=fac
              pyy=sepmxe
              kody=3
            endif
          endif
        elseif(dsep.lt.0.d0) then
          if(sep*sepcs.lt.0.d0) then
```

```
c---included to enable combe to be updated - is not a true event
           fac=(-0.5d-8-sep)/dsep
           if(fac.lt.factor) then
            factor=fac
            pyy=-0.5d-8
            kst=1
           endif
          elseif(sep*sepcs.gt.0.d0) then
           fac=(sepcs-sep)/dsep
           if(fac.lt.factor) then
c---negative shear slop
            factor=fac
            pyy=sepcs
            vcs=vtot-(1.d0-factor)*dv
            if(dtheta.gt.0.d0) vint=-0.4*(vip(0)+selng)
            if(dtheta.lt.0.d0) vint=-0.4*(vin(0)+selng)
            if(vcs.gt.vint) then
             sepint=pyy+(vint-vcs)*ealv(3)
             kody=3
            elseif(vcs.le.vint) then
             if(dtheta.gt.0.d0) then
              ealv(4)=(sepycn-pyy)/(-(vip(0)+selng+vep(0))-vcs)
             elseif(dtheta.lt.0.d0) then
              ealv(4)=(sepycn-pyy)/(-(vin(0)+selng+ven(0))-vcs)
             endif
             kody=4
            endif
           endif
          endif
         endif
        elseif(sepcs.gt.sep) then
         if(dsep.lt.0.d0) then
          if(sep*sepmne.lt.0.d0) then
c---included to enable combe to be updated - is not a true event
           fac=(-0.5d-8-sep)/dsep
           if(fac.lt.factor) then
            factor=fac
            pyy=-0.5d-8
            kst=1
           endif
          elseif(sep*sepmne.gt.0.d0) then
           fac=(sepmne-sep)/dsep
           if(fac.lt.factor) then
c---negative shear slop
            factor=fac
            pyy=sepmne
            kody=3
           endif
          endif
         elseif(dsep.gt.0.d0) then
          if(sep*sepcs.lt.0.d0) then
c---included to enable combe to be updated - is not a true event
           fac=(0.5d-8-sep)/dsep
           if(fac.lt.factor) then
```

```
            factor=fac
            pyy=0.5d-8
            kst=1
          endif
        elseif(sep*sepcs.gt.0.d0) then
          fac=(sepcs-sep)/dsep
          if(fac.lt.factor) then
c---positive shear slop
            factor=fac
            pyy=sepcs
            vcs=vtot-(1.d0-factor)*dv
            if(dtheta.gt.0.d0) vint=0.4*(vip(0)+selng)
            if(dtheta.lt.0.d0) vint=0.4*(vin(0)+selng)
            if(vcs.lt.vint) then
              sepint=pyy+(vint-vcs)*ealv(3)
              kody=3
            elseif(vcs.ge.vint) then
              if(dtheta.gt.0.d0) then
                ealv(4)=(sepycp-pyy)/((vip(0)+selng+vep(0))-vcs)
              elseif(dtheta.lt.0.d0) then
                ealv(4)=(sepycp-pyy)/((vin(0)+selng+ven(0))-vcs)
              endif
              kody=4
            endif
          endif
        endif
      endif
    endif
c---update and calculate (elastic) energy
    if(pyy.ne.0.d0)then
c---stiffness change
      if(kenr.gt.0) ener=ener+((sep+pyy)*0.5d0-aint)*factor*dv
      sep=pyy
    else
c---stiffness remains constant
      dsp=factor*dsep
      sep=sep+dsp
      if(kenr.gt.0) ener=ener+(sep-dsp*0.5-aint)*factor*dv
    endif
  endif

c -------------------------------------------CHECK COMPLETION OF CYCLE
   facac=facac+factor
   if (facac.lt.0.999999d0) go to 10

   if(ealv(kody).lt.0.d0) then
    write(iou,135)imem,kody
    write(*,135)imem,kody
 135  format(/' *** Negative stiffness,'
   1              ' member = ',i5,2x,'yield code = ',i5)
    stop
   endif

c combe(neqq) brought through the subroutine argument declaration
```

```
c ---make allowance for plastic hinge concrete diagonal compressive strength
      if((sep*dsep).gt.0.d0) then
       if(sep.ge.0.5d-8) then
        dspc=dsepc*talpha*fpconst
       elseif(sep.le.-0.5d-8) then
        dspc=-dsepc*talpha*fpconst
       endif
      elseif((sep*dsep).lt.0.d0) then
       if(sep.ge.0.5d-8) then
        dspc=dsepc*talpha*fpconst
       elseif(sep.le.-0.5d-8) then
        dspc=-dsepc*talpha*fpconst
       endif
      endif
      if(lm(1).ne.neqq) then
       combe(lm(1))=combe(lm(1))+cosa*sepcold-cosa*dspc/facc
      endif
      if(lm(2).ne.neqq) then
       combe(lm(2))=combe(lm(2))+sina*sepcold-sina*dspc/facc
      endif
      if(lm(3).ne.neqq) then
       combe(lm(3))=combe(lm(3))-cosa*sepcold+cosa*dspc/facc
      endif
      if(lm(4).ne.neqq) then
       combe(lm(4))=combe(lm(4))-sina*sepcold+sina*dspc/facc
      endif
      sepcold=dspc/facc
      levent=0
      pconst=(seprt+sepct)+(seprb+sepcb)
     1                  -(sep/dabs(sep))*talpha*fpconst*sep
      seps=sep

c -------------------------------------------UPDATE TOTAL STATIC FORCE
c --save current force for second order energy
      stt=st
      st=sep

c --------------------------------E-COMPONENT AND SECOND ORDER ENERGY
c---second order energy (based on static force only)
      if((kenr.gt.0).and.(kgem.gt.0)) then
       delta=(cosa*(dise(1)-dise(3))+sina*(dise(2)-dise(4)))/rlp
       delti=(cosa*(ddise(1)-ddise(3))+sina*(ddise(2)-ddise(4)))/rlp
       enso=enso-((st+stt)*0.5)*(delta-delti*0.5)*delti*rlp
      endif

c -------------------------------------------STATIC RESISTING FORCES
      call elas15(relas,rinit,dise,ndof,kgem)

c -------------------------------------------DAMPING FORCES AND ENERGY
      if(kresis.eq.2) call damp15(rdamp,vele,kenr,ndof,beto,ened,dv)

c --------------------------------------------------ENVELOPE VALUES
      call upen15(kstep)
```

```
c ----------------------------------------------------SAVE RESULTS
    if(ksave.eq.1) call save15(rM)

c ---------------------------------SET INDICATOR FOR STIFFNESS CHANGE
    if (kodyx.ne.kody) kst=1

c ----------------------------ASSIGN AND SAVE UPDATED HINGE INFORMATION
    call hfinish(ihing,igrc)

c -----------------------------------------------------------------
    RETURN
    END


c ************************************************************************
    SUBROUTINE NHEVNT(fpconst,rlpevt,rMevt,rMVpos,
   1                  sep,sepevt,talpha,thetaevt,vcevt)
c ************************************************************************
c SHEAR ELEMENT
c   Calculates the next longitudinal element event within a designated
c   plastic hinge zone, then determines the associated hinge moment,
c   shear force, plastic hinge length, compression zone extension
c   and rotation.
c -----------------------------------------------------------------
c DOUBLE PRECISION / LARGE
    include 'double.h'
c -----------------------------------------------------------------
c  CALLED FROM : resp15
c -----------------------------------------------------------------
c  INPUT
c      ealcb       =  stiffness of  the bottom concrete hinge element.
c      ealct       =  stiffness of  the top concrete hinge element.
c      ealrb       =  stiffness of  the bottom steel hinge element.
c      ealrt       =  stiffness of the top steel hinge element.
c      sepcb       =  force in the bottom concrete hinge element.
c      sepct       =  force in the top concrete hinge element.
c      seprb       =  force in the bottom steel hinge element.
c      seprt       =  force in the top steel hinge element.
c      sepcbevt    =  force in the bottom concrete hinge element at its next event.
c      sepctevt    =  force in the top concrete hinge element at its next event.
c      seprbevt    =  force in the bottom steel hinge element at its next event.
c      seprtevt    =  force in the top steel hinge element at its next event.
c      vtotb       =  extension of the bottom hinge elements.
c      vtott       =  extensionof the top hinge elements.
c      vtotcbevt   =  extension of the bottom concrete hinge element at its next event.
c      vtotctevt   =  extension of the top concrete hinge element at its next event.
c      vtotrbevt   =  extension of the bottom steel hinge element at its next event.
c      vtotrtevt   =  extension of the top steel hinge element at its next event.
c -----------------------------------------------------------------
c  OUTPUT
c      sepbv       =  total force in the longitudinal elements in the bottom
c                     half of the beam.
c      sepcbv      =  force in the bottom concrete element at the first
c                     respective hinge element event.
```

```
c      sepctv      =  force in the top concrete element at the first
c                     respective hinge element event.
c      seprbv      =  force in the bottom steel element at the first
c                     respective hinge element event.
c      seprtv      =  force in the top steel element at the first
c                     respective hinge element event.
c      septv       =  total force in the longitudinal elements in the top
c                     half of the beam.
c      sepcbv      =  force in the bottom concrete element at the first
c                     respective hinge element event.
c      sepctv      =  force in the top concrete element at the first
c                     respective hinge element event.
c      vtbv        =  extension of the bottom elements at the first
c                     respective hinge element event.
c      vttv        =  extension of the top elements at the first
c                     respective hinge element event.
c -------------------------------------------------------------------
c  LABELLED COMMONS
      common/tapes/inp,iou
      include 'hinge.h'
c -------------------------------------------------------------------
c  ARGUMENT DECLARATIONS
c -------------------------------------------------------------------

c ----determine the next longitudinal hinge element event
      top=(db-dt+d_dd)/2.d0
      bot=(dt-db+d_dd)/2.d0
      dvrt=vtotrtevt-vtott
      dvct=vtotctevt-vtott
      if(dabs(dvrt).le.dabs(dvct)) then
       dvt=dvrt
      elseif(dabs(dvrt).gt.dabs(dvct)) then
       dvt=dvct
      endif
      dvrb=vtotrbevt-vtotb
      dvcb=vtotcbevt-vtotb
      if(dabs(dvrb).le.dabs(dvcb)) then
       dvb=dvrb
      elseif(dabs(dvrb).gt.dabs(dvcb)) then
       dvb=dvcb
      endif
      dsept=dvt*(ealrt+ealct)
      dsepb=dvb*(ealrb+ealcb)
c---calculate the respective element forces at the next hinge event
      if(dabs(dsept).le.dabs(dsepb)) then
       if(dvt.eq.dvrt) then
        seprtv=seprtevt
        sepctv=sepct+ealct*dvt
       elseif(dvt.eq.dvct) then
        sepctv=sepctevt
        seprtv=seprt+ealrt*dvt
       endif
       septv=seprtv+sepctv
c because the magnitudes of the axial load and longitudinal component of
```

```
c the concrete diagonal compressive force at the next event are unknown,
c they are assumed to remain at their current levels when determining the
c forces in the members at the next hinge event (remembering P+Fd=T+C)
        sepbv=-septv+((seprb+sepcb)+(seprt+sepct))
        seprbv=sepbv*(seprb/(seprb+sepcb))
        sepcbv=sepbv*(sepcb/(seprb+sepcb))
        dvb=(seprbv-seprb)/ealrb
      elseif(dabs(dsepb).lt.dabs(dsept)) then
        if(dvb.eq.dvrb) then
          seprbv=seprbevt
          sepcbv=sepcb+ealcb*dvb
        elseif(dvb.eq.dvcb) then
          sepcbv=sepcbevt
          seprbv=seprb+ealrb*dvb
        endif
        sepbv=seprbv+sepcbv
        septv=-sepbv+((seprb+sepcb)+(seprt+sepct))
        seprtv=septv*(seprt/(seprt+sepct))
        sepctv=septv*(sepct/(seprt+sepct))
        dvt=(seprtv-seprt)/ealrt
      endif
c---calculate the hinge extensions at the next element event
      vtbv=vtotb+dvb
      vttv=vtott+dvt
c---calculate the maximum moment, shear force, plastic hinge length,
c  compression zone extension and rotation at the next longitudinal hinge
c  element event
      rMhevt=rMVc/(rMVc-0.002)*(-septv*top+sepbv*bot)
      if(rMVpos.eq.0.d0) then
        sepevt=(sep/dabs(sep))*dabs(rMhevt/rMVc)
        rMevt=rMhevt
      elseif(rMVpos.ne.0.d0) then
        sepevt=(sep/dabs(sep))*dabs(rMhevt/rMVpos)
        rMevt=rMhevt+(rMhevt/rMVpos)*(fn/2.d0+fp/2.d0)/2.d0
      endif
      if(septv.gt.0.d0) then
        rlpevt=rigidf*(rigid+fn+((seprtv-pyp)*d_dd/dabs(sepevt)))
        if(rigidf.eq.2.d0) then
          if((rlpevt.lt.rlpnv).or.(rlpevt.gt.4.d0*d_dd)) then
            rlpevt=rlpnv
          endif
        elseif(rigidf.ne.2.d0) then
          if((rlpevt.lt.rlpnv).or.(rlpevt.gt.2.d0*d_dd)) then
            rlpevt=rlpnv
          endif
        endif
        vcevt=vtbv+(vttv-vtbv)*bot/d_dd
        thetaevt=(vtbv-vttv)/d_dd
      elseif(septv.lt.0.d0) then
        rlpevt=rigidf*(rigid+fp+((seprbv-pyp)*d_dd/dabs(sepevt)))
        if(rigidf.eq.2.d0) then
          if((rlpevt.lt.rlppv).or.(rlpevt.gt.4.d0*d_dd)) then
            rlpevt=rlppv
          endif
```

```
      elseif(rigidf.ne.2.d0) then
        if((rlpevt.lt.rlppv).or.(rlpevt.gt.2.d0*d_dd)) then
          rlpevt=rlppv
        endif
      endif
      vcevt=vtbv+(vttv-vtbv)*bot/d_dd
      thetaevt=(vtbv-vttv)/d_dd
    endif
c -----------------------------------------------------------------
    RETURN
    END


c ************************************************************************
    SUBROUTINE STPSTR(igrc,krev,rMevt,rlpevt,thetaevt,vcevt,spvtot)
c ************************************************************************
c SHEAR ELEMENT
c   Calculates the strain in each of the stirrups within a designated
c   plastic hinge zone.
c -----------------------------------------------------------------
c DOUBLE PRECISION / LARGE
    include 'double.h'
c -----------------------------------------------------------------
c  CALLED FROM : resp15
c -----------------------------------------------------------------
c  INPUT
c      krev        =  code indicating what the loading situation is:
c                     0: looking for the next yielding event
c                     1: load reversing
c      rMevt       =  moment acting across the hinge at the event being considered.
c      vcevt       =  extension of the hinge compression steel.
c  OUTPUT
c      psimx       =  maximum curvature in the hinge.
c      pvtot(k)    =  shear deformation in 'truss' stirrup 'k'.
c      spvtot      =  total shear deformation for a moment level of rMevt
c      ven(0)      =  total elastic shear deformation in the negative hinge.
c      vep(0)      =  total elastic shear deformation in the positive hinge.
c      vin(0)      =  total inelastic shear deformation in the negative hinge.
c      vip(0)      =  total inealstic shear deformation in the positive hinge.
c      psicrtn     =  critical curvature, above which yielding in the first
c                     stirrup of the negative truss occurs.
c      psicrtp     =  critical curvature, above which yielding in the first
c                     stirrup of the positive truss occurs.
c -----------------------------------------------------------------
c  LABELLED COMMONS
    common/tapes/inp,iou
    include 'infel15.h'
    include 'hinge.h'
c -----------------------------------------------------------------
c  ARGUMENT DECLARATIONS
    dimension pvtot(20)
c -----------------------------------------------------------------

if(rMevt.gt.0.d0) then
```

```
    psimn=psimnp
    f=fp
   elseif(rMevt.lt.0.d0) then
    psimn=psimnn
    f=fn
   endif
   rtheta=(thetaevt-thetar)
   psimx=2.d0*rtheta/rlpevt-psimn
c In positive moment uni-directional plastic hinges there is no rigid end zone
c yield penetration and the entire hinge length is used when determining the
c shear deformation. In reversing and negative moment plastic hinges
c only the length of the hinge zone which extends out into the beam is used
c to determine the shear deformation that arises from the rotation within
c plastic hinge zones. The rotation resulting from yield penetration into
c the rigid end block does not cause shear deformation. However the extension
c of the rigid end block (which usually results in a large vertical crack at
c the face of the rigid end block) that arises from the yield penetration is
c used when determining the shear deformation due to elongation.
   if(rigidf.eq.2.d0) then
    rlph=rlpevt
   else
    rlph=(rlpevt/rigidf)-rigid
   endif
   vip(0)=0.d0
   vep(0)=0.d0
   vin(0)=0.d0
   ven(0)=0.d0
   spvtot=0.d0
   k=1
   do while((pvtot(k).ne.0.d0).and.(k.le.20))
    pvtot(k)=0.d0
    k=k+1
   enddo
   do k=1,int(rlph/f)+1
    x=k*f
    if(x.le.rlph) then
     psix=psimx-(x/rlph)*(psimx-psimn)
     psix_f=psimx-((x-f)/rlph)*(psimx-psimn)
     pvtot(k)=0.5d0*(psix+psix_f)*f*(f/2.d0)
    elseif((x.gt.rlph).and.((x-f).lt.rlph)) then
     psix_f=psimx-((x-f)/rlph)*(psimx-psimn)
     pvtot(k)=0.5d0*(psix_f+psimn)*(rlph-(x-f))*(f/2.d0)
    elseif((x-f).ge.rlph) then
     pvtot(k)=0.d0
    endif
    pvtot(k)=dabs(pvtot(k))

    if(krev.eq.0) then
     if(rMevt.gt.0.d0) then
      vis=vip(k)
     elseif(rMevt.lt.0.d0) then
      vis=vin(k)
     endif
     spvtot=spvtot+pvtot(k)+vis
```

```
  elseif(krev.eq.1) then
    if(rMevt.gt.0.d0) then
      call stpstrel(pvtot(k),vep(k))
      dvips=pvtot(k)-vep(k)
      if(dvips.gt.0.d0) then
        if(kelas.eq.1) then
          vip(k)=vip(k)-dvip(k)+dvips
        elseif(kelas.eq.0) then
          vip(k)=vip(k)+dvips
        endif
        dvip(k)=dvips
      endif
      vip(0)=vip(0)+vip(k)
      vep(0)=vep(0)+vep(k)
    elseif(rMevt.lt.0.d0) then
      call stpstrel(pvtot(k),ven(k))
      dvins=pvtot(k)-ven(k)
      if(dvins.gt.0.d0) then
        if(kelas.eq.1) then
          vin(k)=vin(k)-dvin(k)+dvins
        elseif(kelas.eq.0) then
          vin(k)=vin(k)+dvins
        endif
        dvin(k)=dvins
      endif
      vin(0)=vin(0)+vin(k)
      ven(0)=ven(0)+ven(k)
    endif
  endif
enddo

if(krev.eq.0) then
  if(rMevt.gt.0.d0) then
    selng=vcevt*fp/d_dd
    if(selng.lt.0.d0) selng=0.d0
    spvtot=spvtot+selng
  elseif(rMevt.lt.0.d0) then
    selng=vcevt*fn/d_dd
    if(selng.lt.0.d0) selng=0.d0
    spvtot=spvtot+selng
  endif
elseif(krev.eq.1) then
  if((rMevt.gt.0.d0).and.(rlpnv.ne.0.d0)) then
    if(rigidf.eq.2.d0) then
      rlpph=rlppv
    else
      rlpph=(rlppv/rigidf-rigid)/fn
    endif
    do m=1,int(rlpph)+1
      indexn=int((m*fn)/fp)
      if(indexn.eq.0) then
        pvtot(m)=(m*fn/fp)*pvtot(1)
        ven(m)=(m*fn/fp)*vep(1)
      else
```

```
          pvtot(m)=pvtot(indexn)-((pvtot(indexn)
     1               -pvtot(indexn+1))*((m*fn-indexn*fp)/fp))
          ven(m)=vep(indexn)-((vep(indexn)
     1               -vep(indexn+1))*((m*fn-indexn*fp)/fp))
        endif
        dvins=pvtot(m)-ven(m)
        if(dvins.gt.0.d0) then
          if(kelas.eq.1) then
            vin(m)=vin(m)-dvin(m)+dvins
          elseif(kelas.eq.0) then
            vin(m)=vin(m)+dvins
          endif
          dvin(m)=dvins
        endif
        vin(0)=vin(0)+vin(m)
        ven(0)=ven(0)+ven(m)
      enddo
    elseif((rMevt.lt.0.d0).and.(rlppv.ne.0.d0)) then
      if(rigidf.eq.2.d0) then
        rlpnh=rlpnv
      else
        rlpnh=(rlpnv/rigidf-rigid)/fp
      endif
      do m=1,int(rlpnh)+1
        indexp=int((m*fp)/fn)
        if(indexp.eq.0) then
          pvtot(m)=(m*fp/fn)*pvtot(1)
          vep(m)=(m*fp/fn)*ven(1)
        else
          pvtot(m)=pvtot(indexp)-((pvtot(indexp)
     1               -pvtot(indexp+1))*((m*fp-indexp*fn)/fn))
          vep(m)=ven(indexp)-((ven(indexp)
     1               -ven(indexp+1))*((m*fp-indexp*fn)/fn))
        endif
        dvips=pvtot(m)-vep(m)
        if(dvips.gt.0.d0) then
          if(kelas.eq.1) then
            vip(m)=vip(m)-dvip(m)+dvips
          elseif(kelas.eq.0) then
            vip(m)=vip(m)+dvips
          endif
          dvip(m)=dvips
        endif
        vip(0)=vip(0)+vip(m)
        vep(0)=vep(0)+vep(m)
      enddo
    endif
    if(rMevt.gt.0.d0) then
      selng=vcevt*fp/d_dd
      if(selng.lt.0.d0) selng=0.d0
      spvtot=vip(0)+vep(0)+selng
      selng=selng*dabs(vtot/spvtot)
    elseif(rMevt.lt.0.d0) then
      selng=vcevt*fn/d_dd
```

```
        if(selng.lt.0.d0) selng=0.d0
        spvtot=vin(0)+ven(0)+selng
        selng=selng*dabs(vtot/spvtot)
       endif

c -----------------------------------DISTRIBUTE PREDICTION ERROR EVENLY
      vep(0)=vep(0)*dabs(vtot/spvtot)
      vip(0)=vip(0)*dabs(vtot/spvtot)
      ven(0)=ven(0)*dabs(vtot/spvtot)
      vin(0)=vin(0)*dabs(vtot/spvtot)

c ---------------------------ASSIGN AND SAVE UPDATED HINGE INFORMATION
      call hfinish(ihing,igrc)
     endif
     spvtot=vtot/dabs(vtot)*spvtot

c -----------------------------------------------------------------
     RETURN
     END


c ***********************************************************************
     SUBROUTINE STPSTREL(dvtot,ve)
c ***********************************************************************
c SHEAR ELEMENT
c   Calculates the elastic recovery strain in each of the "truss"
c   stirrups within a designated plastic hinge zone.
c -----------------------------------------------------------------
c DOUBLE PRECISION / LARGE
     include 'double.h'
c -----------------------------------------------------------------
c  CALLED FROM : resp15
c -----------------------------------------------------------------
c  INPUT
c     dvtot       =  total shear deformation for a particular truss stirrup.
c     eps1        =  current stirrup strain.
c     epsamp      =  strain amplitude of the previous half cycle.
c     epsasc      =  total strain amplitude of all the ascending curves after
c                    the yield strain has been exceeded.
c     epsdes      =  total strain amplitude of all the descending curves
c                    after the yield strain has been exceeded.
c     epsevt      =  next 'event' strain.
c     epsevtold   =  previous 'event' strain.
c     epso        =  strain at the intersection between two asymptotes.
c     epsor       =  strain at the origin point of the second asymptote.
c     f1          =  current stirrup stress.
c     fevt        =  next 'event' stress.
c     fo          =  stress at the intersection between two asymptotes.
c     for         =  stress at the origin point of the second asymptote.
c  OUTPUT
c     ve          =  total elastic shear deformation for the truss stirrup.
c -----------------------------------------------------------------
c  LABELLED COMMONS
     common/tapes/inp,iou
```

```
      include 'infel15.h'
c ---------------------------------------------------------------------
c  ARGUMENT DECLARATIONS
c ---------------------------------------------------------------------

ve=0.d0
      eps1=dvtot/sl
      if(eps1.gt.epsvy) then
        if(eps1.le.epsvh) then
          f1=fvy
        elseif(eps1.gt.epsvh) then
          rn_1=eps1-epsvh
          q_1=epsvu-epsvh
          rm_1=((((fvu/fvy)*(30*q_1+1)**2)-60*q_1-1)/(15*q_1**2))
          psh_1=((rm_1*rn_1+2)/(60*rn_1+2))+((rn_1*
     1                    (60-rm_1))/(2*(30*q_1+1)**2))
          f1=psh_1*fvy
        endif
        epsamp=eps1
        epsasc=epsamp
        epsdes=0.d0
        b=Evb/Evs
        epsrev=epsamp/epsvy
        epssht=(epsasc-0.925*epsdes)/epsvy
        X=(log(epsrev))**2.0*dabs(log(dabs(epssht+20.0))-2.0)
        Y=0.4+3.6*X**(-0.9)
        R=Y*log(epsrev)
        for=-fvy
        epsor=eps1+(for-f1)/Evs
        if(epsor.lt.(epsvy)) epsor=epsvy
        epso=(for-f1+Evs*eps1-Evb*epsor)/(Evs-Evb)
        fo=Evs*epso+f1-Evs*eps1
        epsevtold=eps1
        fevt=-0.5d-8
        epsevt=(fevt-f1)/Evs+eps1

c ---Use Newton-Raphson Method
        iter=0
        maxiter=15
        tol=1.d-3
        epsevto=-9.d8
        do while((dabs((epsevt-epsevto)/epsevt).gt.tol).and.
     1                          (iter.le.maxiter))
          iter=iter+1
          epss=(epsevt-eps1)/(epso-eps1)
          if(epss.lt.0.d0) then
            iter=maxiter+3
          else
            fss=b*epss + (1-b)*epss/(1+epss**R)**(1/R)
            u=fss*(fo-f1) + f1 - fevt
            dfss=b + (1-b)*(1-epss**R/(1+epss**R))/(1+epss**R)**(1/R)
            derivu=dfss*(fo-f1)/(epso-eps1)
            if(derivu.lt.tol) then
              iter=maxiter+2
```

```
          else
            epsevto=epsevt
            epsevt=epsevt-u/derivu
          endif
        endif
      enddo
      if(iter.eq.(maxiter+1)) then
        write(iou,20)maxiter
        write(*,20)maxiter
 20     format(/' *** Couldn"t find next strain ',
  1                      ' within ',i3,' iterations, RESP15')
        stop
      elseif(iter.eq.(maxiter+2)) then
        write(iou,30)
        write(*,30)
 30     format(/' *** Divide by zero error ',
  1                    ' in Newton-Raphson iterations, RESP15')
        stop
      elseif(iter.eq.(maxiter+3)) then
        write(iou,40)
        write(*,40)
 40     format(/' *** Invalid exponentiation error',
  1                    ' in Newton-Raphson iterations, RESP15'/)
        stop
      endif
      ve=(epsevtold-epsevt)*sl
    elseif(eps1.le.epsvy) then
      ve=eps1*sl
    endif

c ------------------------------------------------------------------
    RETURN
    END
```

Appendix C

C1 DRAIN-2DX Input File for Cantilever Beam 2A

```
b2atc    new   0          R.C.F Beam B Experiment
  12  12   0   2   0   0   4   1   0 /b1
   1    0.120      0.000
   2    0.000      0.000
   3    0.1080     0.194
   4    0.1080     0.000
   5    0.1080    -0.194
   6    0.1100     0.194
   7    0.1100     0.000
   8    0.1100    -0.194
   9    0.5000     0.000
  10    1.2500     0.000
  11    1.5000     0.000
  12    2.0000     0.000
   1   1   1   1   1   1          /b4  slab end
   2   0   1   0   2   1          /b4  slab/beam junction
  11   2   1   0  .0     element type 11 -- for steel bar
   1
   1 2.00000E8 0.1571E-2   306.E3  1.989E-2  0.199E0  459.0E3    0.10
   1 0.0225 0.0250 0.0300  0.035  0.040  0.050  0.060  0.080  0.100  0.150
   1   3   6   1   1   1   1
 0.1853E-3   307.E3     0.1    0.388     1.5      1.5    0.100      1.0
   2   5   8   1   1   1   2
  12   2   1   0  .0     element type 12 -- for concrete
   1
   1 2.88200E7  0.03031  0.015155   3.76E0   37.6E3   1    0.05
   1   3   6   1   1   1   1
   2   5   8   1   1   1   2
  15   1   1   0  .0     element type 15 -- for shear
   1
   1 2.00000E8 0.1571E-2   306.E3    0.444 2.00000E8 0.1571E-2   306.E3    0.444
   1 0.1853E-3    0.450    0.100     1.0   0.388
   1 2.00000E8    307.E3  0.01057   0.1772  421.3E3
   1   4   7   1   1   1
   2  10   1   0  .0     RC beams
   4   0   1
   1  2.882E7  .9999     0.1000 1.464E-4   4   4   2
   2  28.82E9  .9999     0.1000 2.083E-3   4   4   2
   3  2.882E7  .9999     0.1000 2.083E-3   4   4   2
   4  2.882E7  .0250     0.0500 1.042E-3   4   4   2 0.040      .2
   1   1  999999.  999999.
```

```
  1  2  0  1  0  1  1
  2  4  0  3  0  1  1
  3  4  0  2  0  1  1
  4  5  0  2  0  1  1
  6  7  0  2  0  1  1
  7  8  0  2  0  1  1
  7  9  0  3  0  1  1
  9 10  0  4  0  1  1
 10 11  0  4  0  1  1
 11 12  0  4  0  1  1
  0                       /d1
  0                           /SECTION E V1.
  0  0  0  0  0  0      /f1
  1  1  1  0  0  5      /g1
 10
 11
 10
  1  1  2
  2  1  2
  3  1
  0  2  0  0  0  0  0
DLL1   1    Equivalent dead load
  11  0.0       -3.20
DLL2   1    Applied vertical load
  11  0.0       100.000
stat   0     Dead Load
  1  0  0  -1  0
   0.100
    1.00            100
DLL1  1.000
stat   0       +2Di
  1  0  0  -1  0
          11      2   2.0E-4
          15.7E-3      20
DLL2  5.000
stop
```

C2 DRAIN-2DX Echo File for Cantilever Beam 2A

DRAIN-2DX

PROGRAM VERSION 00.2, JULY 1991

*** WARNING : TEST VERSION ONLY ***

A. PROBLEM INITIATION

A1. PROBLEM STATUS

Problem name	= b2atc
Problem status	= new
Data checking code	= 0 (execute)
Problem title	= R.C.F Beam B Experiment

B. NODE DEFINITION

B1. CONTROL INFORMATION

Total No. of Nodes	=	12
No. of Control Nodes	=	12
No. of Coord. Generation Commands	=	0
No. of Zero Displacement Commands	=	2
No. of Identical Displ. Commands	=	0
No. of Lumped Mass Commands	=	0
No. of Element Groups	=	4
Total No. of Hinges	=	1
Energy Calculation Code	=	No

B2. CONTROL NODE COORDINATES

Node No.	Offset Node	X-Coord	Y-Coord
1		-1.2000E-01	0.0000E+00
2		0.0000E+00	0.0000E+00
3		1.0800E-01	1.9400E-01

4	1.0800E-01	0.0000E+00
5	1.0800E-01	-1.9400E-01
6	1.1000E-01	1.9400E-01
7	1.1000E-01	0.0000E+00
8	1.1000E-01	-1.9400E-01
9	5.0000E-01	0.0000E+00
10	1.2500E+00	0.0000E+00
11	1.5000E+00	0.0000E+00
12	2.0000E+00	0.0000E+00

B3. COORDINATE GENERATION COMMANDS

None

COMPLETE NODE COORDINATES

Node	X-Coord	Y-Coord
1	-1.2000E-01	0.0000E+00
2	0.0000E+00	0.0000E+00
3	1.0800E-01	1.9400E-01
4	1.0800E-01	0.0000E+00
5	1.0800E-01	-1.9400E-01
6	1.1000E-01	1.9400E-01
7	1.1000E-01	0.0000E+00
8	1.1000E-01	-1.9400E-01
9	5.0000E-01	0.0000E+00
10	1.2500E+00	0.0000E+00
11	1.5000E+00	0.0000E+00
12	2.0000E+00	0.0000E+00

B4. ZERO DISPLACEMENT COMMANDS

First Node	X Code	Y Code	R Code	Last Node	Node Diff
1	1	1	1	1	1
2	0	1	0	2	1

B5. IDENTICAL DISPLACEMENT COMMANDS

None

STRUCTURE DEGREES OF FREEDOM

Node	X	Y	R
1	0	0	0
2	1	0	2

3	3	4	5
4	6	7	8
5	9	10	11
6	12	13	14
7	15	16	17
8	18	19	20
9	21	22	23
10	24	25	26
11	27	28	29
12	30	31	32

C. ELEMENT DEFINITION

C1. CONTROL INFORMATION, ELEMENT GROUP 1

Element type	= 11
No. of elements	= 2
Event code	= 1 (calculate)
Second order code	= 0 (ignore 2nd order effects)
Damping factor	= 0.0000E+00
Group title	= element type 11 -- for steel

C2.11 STEEL TRUSS ELEMENTS

C2.11(a). Control Information

No. of stiffness types = 1

C2.11(b). Stiffness and Yielding Properties

Type No.	Youngs Modulus	Section Area	Yield Stress	Strain at init. S.H.	Ultimate Strain	Ultimate Stress Buck.	Hinge Softening Fact.
1	2.0000E+08	1.5710E-03	3.0600E+05	1.9890E-02	1.9900E-01	4.5900E+05	1.0000E-01

C2.11(d). Specified Strain Levels

Type	Input Strain Values:									
No.	1	2	3	4	5	6	7	8	9	10
1	0.0225	0.0250	0.0300	0.0350	0.0400	0.0500	0.0600	0.0800	0.1000	0.1500

C2.11(c). Element Generation Commands

Elem No.	Node I	Node J	Node Diff	Stif Type	Hinge No.	Hinge Pos.	Shear Area	ShYld Stress	Stp Spacg	Lever Arm	PMoSh Ratio	NMoSh Ratio	Rigid Dist.	Rigid Fac
1	3	6	1	1	1	1	1.85E-04	3.07E+05	0.100	0.388	1.5	1.5	0.10	1.00
2	5	8	1	1	1	2								

C1. CONTROL INFORMATION, ELEMENT GROUP 2

Element type	= 12
No. of elements	= 2
Event code	= 1 (calculate)
Second order code	= 0 (ignore 2nd order effects)
Damping factor	= 0.0000E+00
Group title	= element type 12 -- for concrete

C2.12 CONCRETE TRUSS ELEMENTS (Buckle in tension)

C2.12(a). Control Information

No. of stiffness types = 1

C2.12(b). Stiffness Types and Yielding Properties

Type No.	Concrete Modulus	Gross Sect Area	Section Area	Post Crack Stress	Cylinder Stress	Cont. S.C.	Contact Stress F.
1	2.8820E+07	3.0310E-02	1.5155E-02	3.7600E+00	3.7600E+04	1	5.0000E-02

C2.12(c). Element Generation Commands

Elem No.	Node I	Node J	Node Diff	Stif Type	Hinge No.	Hinge Pos.
1	3	6	1	1	1	1
2	5	8	1	1	1	2

C1. CONTROL INFORMATION, ELEMENT GROUP 3

Element type	= 15
No. of elements	= 1
Event code	= 1 (calculate)
Second order code	= 0 (ignore 2nd order effects)
Damping factor	= 0.0000E+00
Group title	= element type 15 -- for shear

C2.15 SHEAR ELEMENTS

C2.15(a). Control Information

No. of stiffness types = 1

C2.15(b). Stiffness Types - Flexure

Type	Top Steel				Bottom Steel			
No.	Youngs Mod.	Area	Yld Stress	d	Youngs Mod.	Area	Yld Stress	d
1	2.000E+08	1.571E-03	3.060E+05	4.440E-01	2.000E+08	1.571E-03	3.060E+05	4.440E-01

C2.15(c). Stiffness Types - Shear

Type No	Shear Area	Stirrup Leg	Stirrup Length	Axial Spacing	Dist Bet. Load Fact Steel Cen
1	1.8530E-04	4.5000E-01	1.0000E-01	1.0000E+00	3.8800E-01

C2.15(d). Stirrup Steel Properties

Type No.	Youngs Modulus	Yield Stress	Strain Hardening	Ultimate Strain	Ultimate Stress
1	2.0000E+08	3.0700E+05	1.0570E-02	1.7720E-01	4.2130E+05

C2.15(e). Element Generation Commands

Elem No.	Node I	Node J	Node Diff	Stif Type	Hinge No.
1	4	7	1	1	1

C1. CONTROL INFORMATION, ELEMENT GROUP 4

Element type	=	2
No. of elements	=	10
Event code	=	1 (calculate)
Second order code	=	0 (ignore 2nd order effects)
Damping factor	=	0.0000E+00
Group title	=	RC beams

C2.02 BEAM COLUMN ELEMENTS

C2.02(a). Control Information

No. of stiffness types	=	4
No. of eccentricity types	=	0
No. of yield surfaces	=	1

C2.02(b). Stiffness Types

Type No.	Youngs Modulus	Hardening Ratio	Section Area	Reference Inertia	Flex. II	Stif. JJ	Fac. IJ	Shear Area	Poisn Ratio
1	2.882E+07	9.999E-01	1.000E-01	1.464E-04	4.00	4.00	2.00	0.000E+00	0.00
2	2.882E+10	9.999E-01	1.000E-01	2.083E-03	4.00	4.00	2.00	0.000E+00	0.00
3	2.882E+07	9.999E-01	1.000E-01	2.083E-03	4.00	4.00	2.00	0.000E+00	0.00
4	2.882E+07	2.500E-02	5.000E-02	1.042E-03	4.00	4.00	2.00	4.000E-02	0.20

C2.02(d). Yield Interaction Surfaces

Surf No.	Shape Code	Yield Moments Positive	Yield Moments Negative	Yield Forces Compression	Yield Forces Tension	Coordinates of A Moment	Coordinates of A Force	Coordinates of B Moment	Coordinates of B Force
1	1	1.000E+06	1.000E+06						

C2.02(e). Element Generation Commands

Elem No.	Node I	Node J	Node Diff	Stif Type	Eccy Type	Yield Surfaces End I	Yield Surfaces End J
1	1	2	1	1	0	1	1
2	2	4	1	3	0	1	1
3	3	4	1	2	0	1	1
4	4	5	1	2	0	1	1
5	6	7	1	2	0	1	1
6	7	8	1	2	0	1	1
7	7	9	1	3	0	1	1
8	9	10	1	4	0	1	1
9	10	11	1	4	0	1	1
10	11	12	1	4	0	1	1

D. STRUCTURE SECTIONS

*** NO STRUCTURAL SECTIONS ***

E. RELATIVE DISPLACEMENT SETS

*** NO RELATIVE DISPLACEMENT SETS ***

F. RESULTS POST-PROCESSING SPECIFICATION

F1. CONTROL INFORMATION AS INPUT (-1:ALL)

No. of nodes for X disp. post-processing = 0
No. of nodes for Y disp. post-processing = 0
No. of nodes for R rot. post-processing = 0
No. of struc. sections for post-processing = 0
No. of rel. disp. sets for post-processing = 0
No. of elements for post-processing = 0

G. RESULTS PRINT SPECIFICATION

G1. CONTROL INFORMATION AS INPUT (-1:ALL)

No. of nodes for X disp. printout	=	1
No. of nodes for Y disp. printout	=	1
No. of nodes for R rot. printout	=	1
No. of struc. sections for printout	=	0
No. of rel. disp. sets for printout	=	0
No. of elements for printout	=	5

F. SUMMARY OF RESULTS POST-PROCESSING SPECIFICATION

F2. NODES FOR X-DISPLACEMENT POST-PROCESSING

10

Total = 1

F3. NODES FOR Y-DISPLACEMENT POST-PROCESSING

11

Total = 1

F4. NODES FOR R-ROTATION POST-PROCESSING

10

Total = 1

F7. ELEMENTS FOR POST-PROCESSING

Group 1 Elements : 1 2
Group 2 Elements : 1 2
Group 3 Elements : 1

Total = 5

G. SUMMARY OF RESULTS PRINT SPECIFICATION

G2. NODES FOR X-DISPLACEMENT PRINTOUT

10

Total = 1

G3. NODES FOR Y-DISPLACEMENT PRINTOUT

11

Total = 1

G4. NODES FOR R-ROTATION PRINTOUT

10

Total = 1

G7. ELEMENTS FOR PRINTOUT

Group 1 Elements : 1 2
Group 2 Elements : 1 2
Group 3 Elements : 1

Total = 5

H. LOAD DEFINITION

H1. CONTROL INFORMATION

No. of static element load patterns	=	0
No. of static nodal load patterns	=	2
No. of ground acceleration records	=	0
No. of acceleration response spectra	=	0
No. of initial velocity patterns	=	0
No. of ground displacement records	=	0
No. of dynamic force records	=	0

H3. STATIC NODAL LOAD PATTERNS

H3(a). Pattern No. 1

Pattern name	=	DLL1
Pattern title	=	Equivalent dead load
No. of commands	=	1

H3(b). Nodal Load Commands

First Node	X Load	Y Load	Moment Load	Last Node	Node Diff
11	0.0000E+00	-3.2000E+00	0.0000E+00	11	1

NODAL LOADS FOR THIS PATTERN

Node	X-Load	Y-Load	Moment
11	0.0000E+00	-3.2000E+00	0.0000E+00

H3(a). Pattern No. 2

Pattern name	=	DLL2
Pattern title	=	Applied vertical load
No. of commands	=	1

H3(b). Nodal Load Commands

First Node	X Load	Y Load	Moment Load	Last Node	Node Diff
11	0.0000E+00	1.0000E+02	0.0000E+00	11	1

NODAL LOADS FOR THIS PATTERN

Node	X-Load	Y-Load	Moment
11	0.0000E+00	1.0000E+02	0.0000E+00

STORAGE REQUIREMENTS

Excluding element data	=	3444
No. of element blocks	=	1
Total if only one element block	=	7297
Total to be entirely in memory	=	11600
Total available	=	65000

I. ANALYSIS SPECIFICATION

I1. CONTROL INFORMATION

Analysis type	=	stat
Analysis title	=	Dead Load
Second order code	=	0 (ignore)

I3. STATIC ANALYSIS

I3(a). Control Information

No. of static nodal load patterns	=	1
Load step interval for saving state	=	-2 (do not save)
Load step interval for post-proc.	=	-2 (no post-proprocessing)
Load step interval for printout	=	-1 (every event)
Load step interval for envelopes	=	-2 (no envelopes)

I3(b). Step Size Criteria

Load step size = 1.0000E-01

I3(c). Analysis Termination Criteria

Max. load factor	=	1.000E+00
Max. value of controlled disp.	=	1.000E+06
Max. no. of load steps	=	1000
Max. no. of events in any step	=	100
Max. no. of successive direction changes	=	3
Max. absolute nodal translation	=	1.000E+06
Max. absolute nodal rotation	=	1.000E+06

I3(d). Nodal Load Patterns

Name	Multiplier
DLL1	1.000E+00

BEGIN STATIC ANALYSIS, SEGMENT NO. 1

NODAL LOAD INCREMENTS FOR UNIT LOAD FACTOR

Node	X-Load	Y-Load	Moment
11	0.0000E+00	-3.2000E+00	0.0000E+00

ANALYSIS LOG

Step No.	Ev. No.	Ev. Type	Ev. Fact	Gp No	Ele No	Event Code	Load Factor	Force Unbalance	Moment Unbalance
1	1	1	0.0000	3	1	1	1.0016E-06	3.2289E-16	4.6307E-17

	2	1	0.0079	2	1	3	7.9244E-03	7.1917E-06	2.9823E-13
	3	2	0.0921				1.0000E-01	3.8932E-11	4.7517E-12
2	1	2	0.1000				2.0000E-01	2.9161E-11	2.4738E-12
3	1	2	0.1000				3.0000E-01	4.2061E-11	2.8009E-12
4	1	2	0.1000				4.0000E-01	2.6134E-11	2.0091E-12
5	1	2	0.1000				5.0000E-01	2.7004E-11	3.3165E-12
6	1	2	0.1000				6.0000E-01	3.9553E-11	3.6535E-12
7	1	2	0.1000				7.0000E-01	3.2932E-11	1.8037E-12
8	1	2	0.1000				8.0000E-01	2.7333E-11	3.8094E-12
9	1	2	0.1000				9.0000E-01	4.1127E-11	2.9630E-12
10	1	5	0.1000				1.0000E+00	3.7386E-11	3.9853E-12

TERMINATED : MAX. LOAD FACTOR

SUCCESSFUL COMPLETION OF ANALYSIS

I. ANALYSIS SPECIFICATION

I1. CONTROL INFORMATION

Analysis type = stat
Analysis title = +2Di
Second order code = 0 (ignore)

I3. STATIC ANALYSIS

I3(a). Control Information

No. of static nodal load patterns = 1
Load step interval for saving state = -2 (do not save)
Load step interval for post-proc. = -2 (no post-proprocessing)
Load step interval for printout = -1 (every event)
Load step interval for envelopes = -2 (no envelopes)

I3(b). Step Size Criteria

Load step size = 1.0000E+00

Node 1 for controlled displacement = 11
Node 2 for controlled displacement = 0
Displacement direction = Y
Max. displ. increment per load step = 2.0000E-04

I3(c). Analysis Termination Criteria

Max. load factor = 1.000E+00
Max. value of controlled disp. = 1.570E-02
Max. no. of load steps = 1000

Max. no. of events in any step = 20
Max. no. of successive direction changes = 3
Max. absolute nodal translation = 1.000E+06
Max. absolute nodal rotation = 1.000E+06

I3(d). Nodal Load Patterns

Name Multiplier

DLL2 5.000E+00

BEGIN STATIC ANALYSIS, SEGMENT NO. 2

NODAL LOAD INCREMENTS FOR UNIT LOAD FACTOR

Node	X-Load	Y-Load	Moment
11	0.0000E+00	5.0000E+02	0.0000E+00

ANALYSIS LOG

Step No.	Ev. No.	Ev. Type	Ev. Fact	Gp No	Ele No	Event Code	Load Factor	Force Unbalance	Moment Unbalance
1	1	1	0.0000	2	1	4	6.6147E-08	3.7385E-11	3.9844E-12
	2	1	0.0020	2	1	5	2.5398E-03	1.7101E-04	1.9814E-11
	3	1	0.0039	3	1	0	6.4000E-03	2.2097E-05	2.0552E-11
	4	1	0.0001	2	2	3	6.4507E-03	2.0142E-05	1.8938E-11
	5	3	0.0005				6.9250E-03	2.0401E-11	3.0737E-12
2	1	3	0.0066				1.3489E-02	2.1050E-10	5.3122E-11
3	1	3	0.0066				2.0052E-02	3.8003E-10	8.0162E-11
4	1	3	0.0066				2.6616E-02	2.3796E-10	3.8304E-11
5	1	3	0.0066				3.3179E-02	3.8236E-10	7.5140E-11
6	1	3	0.0066				3.9743E-02	2.4561E-10	4.8470E-11
7	1	3	0.0066				4.6306E-02	2.5534E-10	4.3702E-11
8	1	3	0.0066				5.2870E-02	2.9966E-10	6.4063E-11
9	1	3	0.0066				5.9434E-02	3.0275E-10	5.6048E-11
10	1	3	0.0066				6.5997E-02	3.0207E-10	5.5735E-11
11	1	3	0.0066				7.2561E-02	2.4075E-10	3.8128E-11
12	1	3	0.0066				7.9124E-02	2.3618E-10	5.6964E-11
13	1	3	0.0066				8.5688E-02	2.6245E-10	4.9951E-11
14	1	3	0.0066				9.2251E-02	3.5917E-10	7.6383E-11
15	1	3	0.0066				9.8815E-02	3.0528E-10	5.4953E-11
16	1	3	0.0066				1.0538E-01	2.4616E-10	5.7128E-11
17	1	3	0.0066				1.1194E-01	2.8368E-10	6.1277E-11
18	1	3	0.0066				1.1851E-01	3.5683E-10	2.0166E-11

19	1	3	0.0066				1.2507E-01	2.5161E-10	4.9383E-11
20	1	3	0.0066				1.3163E-01	3.0326E-10	7.2916E-11
21	1	3	0.0066				1.3820E-01	3.0569E-10	5.0704E-11
22	1	3	0.0066				1.4476E-01	2.5128E-10	4.0203E-11
23	1	3	0.0066				1.5132E-01	3.2526E-10	7.1310E-11
24	1	3	0.0066				1.5789E-01	3.3242E-10	6.9548E-11
25	1	3	0.0066				1.6445E-01	3.8301E-10	8.7041E-11
26	1	3	0.0066				1.7101E-01	3.1328E-10	5.9003E-11
27	1	3	0.0066				1.7758E-01	3.0582E-10	4.9511E-11
28	1	3	0.0066				1.8414E-01	3.6723E-10	7.8145E-11
29	1	3	0.0066				1.9070E-01	3.3293E-10	3.8369E-11
30	1	3	0.0066				1.9727E-01	2.4261E-10	5.6531E-11
31	1	3	0.0066				2.0383E-01	3.5101E-10	1.7209E-11
32	1	3	0.0066				2.1040E-01	2.6108E-10	4.5929E-11
33	1	3	0.0066				2.1696E-01	2.8905E-10	6.2073E-11
34	1	3	0.0066				2.2352E-01	2.9883E-10	5.2381E-11
35	1	3	0.0066				2.3009E-01	2.5545E-10	5.7582E-11
36	1	3	0.0066				2.3665E-01	3.5914E-10	5.5365E-11
37	1	3	0.0066				2.4321E-01	2.2780E-10	4.3485E-11
38	1	3	0.0066				2.4978E-01	3.3936E-10	5.1187E-11
39	1	1	0.0055	1	2	11	2.5524E-01	1.2666E-04	4.2832E-11
	2	3	0.0000				2.5525E-01	7.6056E-11	1.0796E-11
40	1	3	0.0001				2.5530E-01	3.8386E-10	5.9401E-11
41	1	3	0.0001				2.5535E-01	5.7361E-10	8.0028E-11
42	1	3	0.0001				2.5540E-01	7.5744E-10	9.9323E-11
43	1	3	0.0001				2.5545E-01	6.8439E-10	7.4663E-11
44	1	3	0.0001				2.5550E-01	4.2081E-10	6.1220E-11
45	1	3	0.0001				2.5555E-01	5.1210E-10	6.9263E-11
46	1	3	0.0001				2.5561E-01	3.1628E-10	9.7202E-11
47	1	3	0.0001				2.5566E-01	7.9217E-10	9.0128E-11
48	1	3	0.0001				2.5571E-01	6.1959E-10	8.4825E-11
49	1	3	0.0001				2.5576E-01	3.9512E-10	5.6354E-11
50	1	3	0.0001				2.5581E-01	4.4284E-10	8.1997E-11
51	1	3	0.0001				2.5586E-01	4.8942E-10	6.2270E-11
52	1	3	0.0001				2.5591E-01	4.5122E-10	6.8155E-11
53	1	3	0.0001				2.5596E-01	4.5043E-10	6.3991E-11
54	1	3	0.0001				2.5601E-01	1.0402E-09	1.3833E-10
55	1	3	0.0001				2.5606E-01	7.1753E-10	8.5516E-11
56	1	3	0.0001				2.5611E-01	6.1442E-10	8.0970E-11
57	1	3	0.0001				2.5616E-01	8.3264E-10	1.0060E-10
58	1	3	0.0001				2.5621E-01	4.0546E-10	6.2693E-11
59	1	3	0.0001				2.5626E-01	3.9381E-10	6.6620E-11
60	1	3	0.0001				2.5631E-01	5.6031E-10	7.3800E-11
61	1	3	0.0001				2.5636E-01	6.4762E-10	7.8911E-11
62	1	3	0.0001				2.5642E-01	5.4466E-10	8.2451E-11
63	1	3	0.0001				2.5647E-01	6.2204E-10	7.6700E-11
64	1	3	0.0001				2.5652E-01	4.3275E-10	6.9557E-11
65	1	3	0.0001				2.5657E-01	7.1080E-10	9.2456E-11
66	1	3	0.0001				2.5662E-01	5.8776E-10	6.6931E-11
67	1	3	0.0001				2.5667E-01	9.7538E-10	1.1090E-10
68	1	3	0.0001				2.5672E-01	8.4088E-10	1.0778E-10
69	1	3	0.0001				2.5677E-01	4.3106E-10	9.8282E-11
70	1	3	0.0001				2.5682E-01	4.8215E-10	7.9239E-11
71	1	3	0.0001				2.5687E-01	7.1812E-10	1.3398E-10

72	1	3	0.0001	2.5692E-01	5.6104E-10	7.3784E-11
73	1	3	0.0001	2.5697E-01	6.6092E-10	8.7266E-11
74	1	3	0.0001	2.5702E-01	1.0150E-09	1.0536E-10
75	1	3	0.0001	2.5707E-01	6.9245E-10	8.1911E-11
76	1	3	0.0001	2.5712E-01	6.1061E-10	8.2401E-11
77	1	3	0.0001	2.5718E-01	8.8573E-10	1.0956E-10
78	1	3	0.0001	2.5723E-01	8.3338E-10	9.5539E-11
79	1	3	0.0001	2.5728E-01	5.9936E-10	7.8521E-11
80	1	6	0.0001	2.5733E-01	3.5192E-10	6.8979E-11

TERMINATED : MAX. CONTROLLED DISPL.

SUCCESSFUL COMPLETION OF ANALYSIS

STOP. END OF ANALYSIS SESSION.

C3 DRAIN-2DX Output File for Cantilever Beam 2A

```
*---------------------------------------------------------*
RESULTS PRINTOUT : PROBLEM I.D. = b2atc
*---------------------------------------------------------*
 PROBLEM TITLE =  R.C.F Beam B Experiment
*---------------------------------------------------------*


*---------------------------------------------------------*
RESULTS FOR ANALYSIS SEGMENT   1 ANALYSIS TYPE = stat
*---------------------------------------------------------*
TITLE =     Dead Load
*----------------------------------- ----------------------*


HISTORY OF X DISPLACEMENTS

   Step   Time/         Node
    No. Load Factor      10

    1 1.0016E-06 6.9487E-25
      7.9244E-03 2.0795E-09
      1.0000E-01 2.5658E-07
    2 2.0000E-01 5.3299E-07
    3 3.0000E-01 8.0940E-07
    4 4.0000E-01 1.0858E-06
    5 5.0000E-01 1.3622E-06
    6 6.0000E-01 1.6386E-06
    7 7.0000E-01 1.9150E-06
    8 8.0000E-01 2.1914E-06
    9 9.0000E-01 2.4678E-06
   10 1.0000E+00 2.7442E-06


 HISTORY OF Y DISPLACEMENTS

   Step   Time/         Node
    No. Load Factor      11

    1 1.0016E-06 -1.7717E-10
      7.9244E-03 -1.4018E-06
      1.0000E-01 -1.9339E-05
    2 2.0000E-01 -3.8821E-05
    3 3.0000E-01 -5.8302E-05
    4 4.0000E-01 -7.7784E-05
    5 5.0000E-01 -9.7265E-05
    6 6.0000E-01 -1.1675E-04
    7 7.0000E-01 -1.3623E-04
    8 8.0000E-01 -1.5571E-04
```

9	9.0000E-01	-1.7519E-04
10	1.0000E+00	-1.9467E-04

HISTORY OF ROTATIONS

Step No.	Time/ Load Factor	Node 10
1	1.0016E-06	-1.3460E-10
	7.9244E-03	-1.0650E-06
	1.0000E-01	-1.4627E-05
2	2.0000E-01	-2.9356E-05
3	3.0000E-01	-4.4084E-05
4	4.0000E-01	-5.8813E-05
5	5.0000E-01	-7.3542E-05
6	6.0000E-01	-8.8271E-05
7	7.0000E-01	-1.0300E-04
8	8.0000E-01	-1.1773E-04
9	9.0000E-01	-1.3246E-04
10	1.0000E+00	-1.4719E-04

HISTORY FOR ELEMENT GROUP 1 ELEMENT NUMBER 1

STEEL TRUSS ELEMENT (TYPE 11). NODES = 3 6

Step No.	Time/ Load Fact.	Yield Code	Static Force	Total Extension	Hinge Length	Current Strain	Current Stress
1	1.002E-06	0	4.804E-06	3.342E-12	2.186E-01	1.529E-11	3.058E-03
1	7.924E-03	0	4.100E-02	2.852E-08	2.186E-01	1.305E-07	2.610E+01
1	1.000E-01	0	1.180E+00	8.206E-07	2.186E-01	3.754E-06	7.508E+02
2	2.000E-01	0	2.416E+00	1.681E-06	2.186E-01	7.690E-06	1.538E+03
3	3.000E-01	0	3.653E+00	2.541E-06	2.186E-01	1.163E-05	2.325E+03
4	4.000E-01	0	4.889E+00	3.401E-06	2.186E-01	1.556E-05	3.112E+03
5	5.000E-01	0	6.126E+00	4.262E-06	2.186E-01	1.950E-05	3.899E+03
6	6.000E-01	0	7.362E+00	5.122E-06	2.186E-01	2.343E-05	4.686E+03
7	7.000E-01	0	8.599E+00	5.982E-06	2.186E-01	2.737E-05	5.473E+03
8	8.000E-01	0	9.835E+00	6.842E-06	2.186E-01	3.130E-05	6.261E+03
9	9.000E-01	0	1.107E+01	7.703E-06	2.186E-01	3.524E-05	7.048E+03
10	1.000E+00	0	1.231E+01	8.563E-06	2.186E-01	3.917E-05	7.835E+03

HISTORY FOR ELEMENT GROUP 1 ELEMENT NUMBER 2

STEEL TRUSS ELEMENT (TYPE 11). NODES = 5 8

Step No.	Time/ Load Fact.	Yield Code	Static Force	Total Extension	Hinge Length	Current Strain	Current Stress
1	1.002E-06	0	-4.804E-06	-3.342E-12	2.186E-01	-1.529E-11	-3.058E-03
1	7.924E-03	0	-3.502E-02	-2.436E-08	2.186E-01	-1.115E-07	-2.229E+01
1	1.000E-01	0	-4.419E-01	-3.074E-07	2.186E-01	-1.407E-06	-2.813E+02

2	2.000E-01	0	-8.839E-01	-6.149E-07	2.186E-01	-2.813E-06	-5.626E+02
3	3.000E-01	0	-1.326E+00	-9.223E-07	2.186E-01	-4.220E-06	-8.439E+02
4	4.000E-01	0	-1.768E+00	-1.230E-06	2.186E-01	-5.626E-06	-1.125E+03
5	5.000E-01	0	-2.210E+00	-1.537E-06	2.186E-01	-7.033E-06	-1.407E+03
6	6.000E-01	0	-2.652E+00	-1.845E-06	2.186E-01	-8.439E-06	-1.688E+03
7	7.000E-01	0	-3.094E+00	-2.152E-06	2.186E-01	-9.846E-06	-1.969E+03
8	8.000E-01	0	-3.535E+00	-2.460E-06	2.186E-01	-1.125E-05	-2.250E+03
9	9.000E-01	0	-3.977E+00	-2.767E-06	2.186E-01	-1.266E-05	-2.532E+03
10	1.000E+00	0	-4.419E+00	-3.074E-06	2.186E-01	-1.407E-05	-2.813E+03

HISTORY FOR ELEMENT GROUP 2 ELEMENT NUMBER 1

CONCRETE TRUSS ELEMENT (TYPE 12). NODES = 3 6

Step No.	Time/ Load Fact.	Yield Code	Static Force	Total Extension
1	1.002E-06	0	6.678E-06	3.342E-12
1	7.924E-03	3	5.698E-02	2.852E-08
1	1.000E-01	3	5.698E-02	8.206E-07
2	2.000E-01	3	5.699E-02	1.681E-06
3	3.000E-01	3	5.699E-02	2.541E-06
4	4.000E-01	3	5.699E-02	3.401E-06
5	5.000E-01	3	5.699E-02	4.262E-06
6	6.000E-01	3	5.699E-02	5.122E-06
7	7.000E-01	3	5.699E-02	5.982E-06
8	8.000E-01	3	5.700E-02	6.842E-06
9	9.000E-01	3	5.700E-02	7.703E-06
10	1.000E+00	3	5.700E-02	8.563E-06

HISTORY FOR ELEMENT GROUP 2 ELEMENT NUMBER 2

CONCRETE TRUSS ELEMENT (TYPE 12). NODES = 5 8

Step No.	Time/ Load Fact.	Yield Code	Static Force	Total Extension
1	1.002E-06	0	-6.678E-06	-3.342E-12
1	7.924E-03	0	-4.868E-02	-2.436E-08
1	1.000E-01	0	-6.143E-01	-3.074E-07
2	2.000E-01	0	-1.229E+00	-6.149E-07
3	3.000E-01	0	-1.843E+00	-9.223E-07
4	4.000E-01	0	-2.457E+00	-1.230E-06
5	5.000E-01	0	-3.072E+00	-1.537E-06
6	6.000E-01	0	-3.686E+00	-1.845E-06
7	7.000E-01	0	-4.300E+00	-2.152E-06
8	8.000E-01	0	-4.915E+00	-2.460E-06
9	9.000E-01	0	-5.529E+00	-2.767E-06
10	1.000E+00	0	-6.143E+00	-3.074E-06

HISTORY FOR ELEMENT GROUP 3 ELEMENT NUMBER 1

SHEAR ELEMENT (TYPE 15). NODES = 4 7

Step No.	Time/ Load Fact.	Yield Code	Shear Force	Total Shear Deformation	Moment at end of Step	Rotation at end of Step
1	1.002E-06	1	-3.205E-06	-1.780E-11	-4.805E-06	-1.723E-11
1	7.924E-03	1	-2.536E-02	-1.409E-07	-3.802E-02	-1.363E-07
1	1.000E-01	1	-3.200E-01	-1.778E-06	-4.798E-01	-2.907E-06
2	2.000E-01	1	-6.400E-01	-3.555E-06	-9.595E-01	-5.917E-06
3	3.000E-01	1	-9.600E-01	-5.333E-06	-1.439E+00	-8.926E-06
4	4.000E-01	1	-1.280E+00	-7.110E-06	-1.919E+00	-1.194E-05
5	5.000E-01	1	-1.600E+00	-8.888E-06	-2.399E+00	-1.495E-05
6	6.000E-01	1	-1.920E+00	-1.067E-05	-2.879E+00	-1.795E-05
7	7.000E-01	1	-2.240E+00	-1.244E-05	-3.358E+00	-2.096E-05
8	8.000E-01	1	-2.560E+00	-1.422E-05	-3.838E+00	-2.397E-05
9	9.000E-01	1	-2.880E+00	-1.600E-05	-4.318E+00	-2.698E-05
10	1.000E+00	1	-3.200E+00	-1.778E-05	-4.798E+00	-2.999E-05

--

RESULTS FOR ANALYSIS SEGMENT 2 ANALYSIS TYPE = stat

--

TITLE = +2Di

--

HISTORY OF X DISPLACEMENTS

Step No.	Time/ Load Factor	Node 10
1	6.6147E-08	2.7442E-06
	2.5398E-03	1.6474E-06
	6.4000E-03	1.4692E-06
	6.4507E-03	1.4710E-06
	6.9250E-03	1.6738E-06
2	1.3489E-02	4.4796E-06
3	2.0052E-02	7.2854E-06
4	2.6616E-02	1.0091E-05
5	3.3179E-02	1.2897E-05
6	3.9743E-02	1.5703E-05
7	4.6306E-02	1.8509E-05
8	5.2870E-02	2.1315E-05
9	5.9434E-02	2.4120E-05
10	6.5997E-02	2.6926E-05
11	7.2561E-02	2.9732E-05
12	7.9124E-02	3.2538E-05
13	8.5688E-02	3.5344E-05
14	9.2251E-02	3.8150E-05
15	9.8815E-02	4.0955E-05
16	1.0538E-01	4.3761E-05
17	1.1194E-01	4.6567E-05

18	1.1851E-01	4.9373E-05
19	1.2507E-01	5.2179E-05
20	1.3163E-01	5.4984E-05
21	1.3820E-01	5.7790E-05
22	1.4476E-01	6.0596E-05
23	1.5132E-01	6.3402E-05
24	1.5789E-01	6.6208E-05
25	1.6445E-01	6.9014E-05
26	1.7101E-01	7.1819E-05
27	1.7758E-01	7.4625E-05
28	1.8414E-01	7.7431E-05
29	1.9070E-01	8.0237E-05
30	1.9727E-01	8.3043E-05
31	2.0383E-01	8.5849E-05
32	2.1040E-01	8.8654E-05
33	2.1696E-01	9.1460E-05
34	2.2352E-01	9.4266E-05
35	2.3009E-01	9.7072E-05
36	2.3665E-01	9.9878E-05
37	2.4321E-01	1.0268E-04
38	2.4978E-01	1.0549E-04
39	2.5524E-01	1.0783E-04
	2.5525E-01	1.1246E-04
40	2.5530E-01	1.4018E-04
41	2.5535E-01	1.6790E-04
42	2.5540E-01	1.9562E-04
43	2.5545E-01	2.2334E-04
44	2.5550E-01	2.5106E-04
45	2.5555E-01	2.7878E-04
46	2.5561E-01	3.0650E-04
47	2.5566E-01	3.3422E-04
48	2.5571E-01	3.6194E-04
49	2.5576E-01	3.8966E-04
50	2.5581E-01	4.1738E-04
51	2.5586E-01	4.4510E-04
52	2.5591E-01	4.7282E-04
53	2.5596E-01	5.0054E-04
54	2.5601E-01	5.2826E-04
55	2.5606E-01	5.5598E-04
56	2.5611E-01	5.8370E-04
57	2.5616E-01	6.1142E-04
58	2.5621E-01	6.3914E-04
59	2.5626E-01	6.6686E-04
60	2.5631E-01	6.9458E-04
61	2.5636E-01	7.2230E-04
62	2.5642E-01	7.5002E-04
63	2.5647E-01	7.7774E-04
64	2.5652E-01	8.0546E-04
65	2.5657E-01	8.3318E-04
66	2.5662E-01	8.6090E-04
67	2.5667E-01	8.8862E-04
68	2.5672E-01	9.1634E-04
69	2.5677E-01	9.4406E-04
70	2.5682E-01	9.7178E-04

71	2.5687E-01	9.9950E-04
72	2.5692E-01	1.0272E-03
73	2.5697E-01	1.0549E-03
74	2.5702E-01	1.0827E-03
75	2.5707E-01	1.1104E-03
76	2.5712E-01	1.1381E-03
77	2.5717E-01	1.1658E-03
78	2.5723E-01	1.1935E-03
79	2.5728E-01	1.2213E-03
80	2.5733E-01	1.2490E-03

HISTORY OF Y DISPLACEMENTS

Step No.	Time/ Load Factor	Node 11
1	6.6147E-08	-1.9467E-04
	2.5398E-03	-1.1736E-04
	6.4000E-03	-1.0525E-05
	6.4507E-03	-9.1233E-06
	6.9250E-03	5.3279E-06
2	1.3489E-02	2.0533E-04
3	2.0052E-02	4.0533E-04
4	2.6616E-02	6.0533E-04
5	3.3179E-02	8.0533E-04
6	3.9743E-02	1.0053E-03
7	4.6306E-02	1.2053E-03
8	5.2870E-02	1.4053E-03
9	5.9434E-02	1.6053E-03
10	6.5997E-02	1.8053E-03
11	7.2561E-02	2.0053E-03
12	7.9124E-02	2.2053E-03
13	8.5688E-02	2.4053E-03
14	9.2251E-02	2.6053E-03
15	9.8815E-02	2.8053E-03
16	1.0538E-01	3.0053E-03
17	1.1194E-01	3.2053E-03
18	1.1851E-01	3.4053E-03
19	1.2507E-01	3.6053E-03
20	1.3163E-01	3.8053E-03
21	1.3820E-01	4.0053E-03
22	1.4476E-01	4.2053E-03
23	1.5132E-01	4.4053E-03
24	1.5789E-01	4.6053E-03
25	1.6445E-01	4.8053E-03
26	1.7101E-01	5.0053E-03
27	1.7758E-01	5.2053E-03
28	1.8414E-01	5.4053E-03
29	1.9070E-01	5.6053E-03
30	1.9727E-01	5.8053E-03
31	2.0383E-01	6.0053E-03
32	2.1040E-01	6.2053E-03
33	2.1696E-01	6.4053E-03

34	2.2352E-01	6.6053E-03
35	2.3009E-01	6.8053E-03
36	2.3665E-01	7.0053E-03
37	2.4321E-01	7.2053E-03
38	2.4978E-01	7.4053E-03
39	2.5524E-01	7.5719E-03
	2.5525E-01	7.6053E-03
40	2.5530E-01	7.8053E-03
41	2.5535E-01	8.0053E-03
42	2.5540E-01	8.2053E-03
43	2.5545E-01	8.4053E-03
44	2.5550E-01	8.6053E-03
45	2.5555E-01	8.8053E-03
46	2.5561E-01	9.0053E-03
47	2.5566E-01	9.2053E-03
48	2.5571E-01	9.4053E-03
49	2.5576E-01	9.6053E-03
50	2.5581E-01	9.8053E-03
51	2.5586E-01	1.0005E-02
52	2.5591E-01	1.0205E-02
53	2.5596E-01	1.0405E-02
54	2.5601E-01	1.0605E-02
55	2.5606E-01	1.0805E-02
56	2.5611E-01	1.1005E-02
57	2.5616E-01	1.1205E-02
58	2.5621E-01	1.1405E-02
59	2.5626E-01	1.1605E-02
60	2.5631E-01	1.1805E-02
61	2.5636E-01	1.2005E-02
62	2.5642E-01	1.2205E-02
63	2.5647E-01	1.2405E-02
64	2.5652E-01	1.2605E-02
65	2.5657E-01	1.2805E-02
66	2.5662E-01	1.3005E-02
67	2.5667E-01	1.3205E-02
68	2.5672E-01	1.3405E-02
69	2.5677E-01	1.3605E-02
70	2.5682E-01	1.3805E-02
71	2.5687E-01	1.4005E-02
72	2.5692E-01	1.4205E-02
73	2.5697E-01	1.4405E-02
74	2.5702E-01	1.4605E-02
75	2.5707E-01	1.4805E-02
76	2.5712E-01	1.5005E-02
77	2.5717E-01	1.5205E-02
78	2.5723E-01	1.5405E-02
79	2.5728E-01	1.5605E-02
80	2.5733E-01	1.5805E-02

HISTORY OF ROTATIONS

Step No.	Time/ Load Factor	Node 10
1	6.6147E-08	-1.4719E-04
	2.5398E-03	-8.8736E-05
	6.4000E-03	-7.5722E-06
	6.4507E-03	-6.5070E-06
	6.9250E-03	4.4183E-06
2	1.3489E-02	1.5562E-04
3	2.0052E-02	3.0682E-04
4	2.6616E-02	4.5802E-04
5	3.3179E-02	6.0923E-04
6	3.9743E-02	7.6043E-04
7	4.6306E-02	9.1163E-04
8	5.2870E-02	1.0628E-03
9	5.9434E-02	1.2140E-03
10	6.5997E-02	1.3652E-03
11	7.2561E-02	1.5164E-03
12	7.9124E-02	1.6676E-03
13	8.5688E-02	1.8188E-03
14	9.2251E-02	1.9700E-03
15	9.8815E-02	2.1212E-03
16	1.0538E-01	2.2724E-03
17	1.1194E-01	2.4237E-03
18	1.1851E-01	2.5749E-03
19	1.2507E-01	2.7261E-03
20	1.3163E-01	2.8773E-03
21	1.3820E-01	3.0285E-03
22	1.4476E-01	3.1797E-03
23	1.5132E-01	3.3309E-03
24	1.5789E-01	3.4821E-03
25	1.6445E-01	3.6333E-03
26	1.7101E-01	3.7845E-03
27	1.7758E-01	3.9357E-03
28	1.8414E-01	4.0869E-03
29	1.9070E-01	4.2381E-03
30	1.9727E-01	4.3893E-03
31	2.0383E-01	4.5405E-03
32	2.1040E-01	4.6917E-03
33	2.1696E-01	4.8429E-03
34	2.2352E-01	4.9941E-03
35	2.3009E-01	5.1453E-03
36	2.3665E-01	5.2965E-03
37	2.4321E-01	5.4477E-03
38	2.4978E-01	5.5989E-03
39	2.5524E-01	5.7248E-03
	2.5525E-01	5.7489E-03
40	2.5530E-01	5.8928E-03
41	2.5535E-01	6.0368E-03
42	2.5540E-01	6.1807E-03
43	2.5545E-01	6.3246E-03
44	2.5550E-01	6.4686E-03
45	2.5555E-01	6.6125E-03

46	2.5561E-01	6.7565E-03
47	2.5566E-01	6.9004E-03
48	2.5571E-01	7.0444E-03
49	2.5576E-01	7.1883E-03
50	2.5581E-01	7.3322E-03
51	2.5586E-01	7.4762E-03
52	2.5591E-01	7.6201E-03
53	2.5596E-01	7.7641E-03
54	2.5601E-01	7.9080E-03
55	2.5606E-01	8.0519E-03
56	2.5611E-01	8.1959E-03
57	2.5616E-01	8.3398E-03
58	2.5621E-01	8.4838E-03
59	2.5626E-01	8.6277E-03
60	2.5631E-01	8.7716E-03
61	2.5636E-01	8.9156E-03
62	2.5642E-01	9.0595E-03
63	2.5647E-01	9.2035E-03
64	2.5652E-01	9.3474E-03
65	2.5657E-01	9.4914E-03
66	2.5662E-01	9.6353E-03
67	2.5667E-01	9.7792E-03
68	2.5672E-01	9.9232E-03
69	2.5677E-01	1.0067E-02
70	2.5682E-01	1.0211E-02
71	2.5687E-01	1.0355E-02
72	2.5692E-01	1.0499E-02
73	2.5697E-01	1.0643E-02
74	2.5702E-01	1.0787E-02
75	2.5707E-01	1.0931E-02
76	2.5712E-01	1.1075E-02
77	2.5717E-01	1.1219E-02
78	2.5723E-01	1.1363E-02
79	2.5728E-01	1.1507E-02
80	2.5733E-01	1.1650E-02

HISTORY FOR ELEMENT GROUP 1 ELEMENT NUMBER 1

STEEL TRUSS ELEMENT (TYPE 11). NODES = 3 6

Step No.	Time/ Load Fact.	Yield Code	Static Force	Total Extension	Hinge Length	Current Strain	Current Stress
1	6.615E-08	0	1.231E+01	8.563E-06	2.186E-01	3.917E-05	7.835E+03
1	2.540E-03	0	7.401E+00	5.149E-06	2.186E-01	2.356E-05	4.711E+03
1	6.400E-03	0	4.224E+00	2.938E-06	2.186E-01	1.344E-05	2.689E+03
1	6.451E-03	0	4.188E+00	2.914E-06	2.186E-01	1.333E-05	2.666E+03
1	6.925E-03	0	3.855E+00	2.682E-06	2.186E-01	1.227E-05	2.454E+03
2	1.349E-02	0	-7.607E-01	-5.292E-07	2.186E-01	-2.421E-06	-4.842E+02
3	2.005E-02	0	-5.376E+00	-3.740E-06	2.186E-01	-1.711E-05	-3.422E+03
4	2.662E-02	0	-9.991E+00	-6.951E-06	2.186E-01	-3.180E-05	-6.360E+03
5	3.318E-02	0	-1.461E+01	-1.016E-05	2.186E-01	-4.649E-05	-9.298E+03
6	3.974E-02	0	-1.922E+01	-1.337E-05	2.186E-01	-6.118E-05	-1.224E+04

7	4.631E-02	0	-2.384E+01	-1.658E-05	2.186E-01	-7.587E-05	-1.517E+04
8	5.287E-02	0	-2.845E+01	-1.979E-05	2.186E-01	-9.055E-05	-1.811E+04
9	5.943E-02	0	-3.307E+01	-2.300E-05	2.186E-01	-1.052E-04	-2.105E+04
10	6.600E-02	0	-3.768E+01	-2.622E-05	2.186E-01	-1.199E-04	-2.399E+04
11	7.256E-02	0	-4.230E+01	-2.943E-05	2.186E-01	-1.346E-04	-2.692E+04
12	7.912E-02	0	-4.691E+01	-3.264E-05	2.186E-01	-1.493E-04	-2.986E+04
13	8.569E-02	0	-5.153E+01	-3.585E-05	2.186E-01	-1.640E-04	-3.280E+04
14	9.225E-02	0	-5.614E+01	-3.906E-05	2.186E-01	-1.787E-04	-3.574E+04
15	9.881E-02	0	-6.076E+01	-4.227E-05	2.186E-01	-1.934E-04	-3.868E+04
16	1.054E-01	0	-6.537E+01	-4.548E-05	2.186E-01	-2.081E-04	-4.161E+04
17	1.119E-01	0	-6.999E+01	-4.869E-05	2.186E-01	-2.228E-04	-4.455E+04
18	1.185E-01	0	-7.460E+01	-5.190E-05	2.186E-01	-2.374E-04	-4.749E+04
19	1.251E-01	0	-7.922E+01	-5.511E-05	2.186E-01	-2.521E-04	-5.043E+04
20	1.316E-01	0	-8.384E+01	-5.832E-05	2.186E-01	-2.668E-04	-5.336E+04
21	1.382E-01	0	-8.845E+01	-6.153E-05	2.186E-01	-2.815E-04	-5.630E+04
22	1.448E-01	0	-9.307E+01	-6.475E-05	2.186E-01	-2.962E-04	-5.924E+04
23	1.513E-01	0	-9.768E+01	-6.796E-05	2.186E-01	-3.109E-04	-6.218E+04
24	1.579E-01	0	-1.023E+02	-7.117E-05	2.186E-01	-3.256E-04	-6.512E+04
25	1.645E-01	0	-1.069E+02	-7.438E-05	2.186E-01	-3.403E-04	-6.805E+04
26	1.710E-01	0	-1.115E+02	-7.759E-05	2.186E-01	-3.550E-04	-7.099E+04
27	1.776E-01	0	-1.161E+02	-8.080E-05	2.186E-01	-3.696E-04	-7.393E+04
28	1.841E-01	0	-1.208E+02	-8.401E-05	2.186E-01	-3.843E-04	-7.687E+04
29	1.907E-01	0	-1.254E+02	-8.722E-05	2.186E-01	-3.990E-04	-7.980E+04
30	1.973E-01	0	-1.300E+02	-9.043E-05	2.186E-01	-4.137E-04	-8.274E+04
31	2.038E-01	0	-1.346E+02	-9.364E-05	2.186E-01	-4.284E-04	-8.568E+04
32	2.104E-01	0	-1.392E+02	-9.685E-05	2.186E-01	-4.431E-04	-8.862E+04
33	2.170E-01	0	-1.438E+02	-1.001E-04	2.186E-01	-4.578E-04	-9.156E+04
34	2.235E-01	0	-1.484E+02	-1.033E-04	2.186E-01	-4.725E-04	-9.449E+04
35	2.301E-01	0	-1.531E+02	-1.065E-04	2.186E-01	-4.872E-04	-9.743E+04
36	2.366E-01	0	-1.577E+02	-1.097E-04	2.186E-01	-5.018E-04	-1.004E+05
37	2.432E-01	0	-1.623E+02	-1.129E-04	2.186E-01	-5.165E-04	-1.033E+05
38	2.498E-01	0	-1.669E+02	-1.161E-04	2.186E-01	-5.312E-04	-1.062E+05
39	2.552E-01	0	-1.708E+02	-1.188E-04	2.186E-01	-5.435E-04	-1.087E+05
39	2.553E-01	0	-1.708E+02	-1.188E-04	2.186E-01	-5.435E-04	-1.087E+05
40	2.553E-01	0	-1.708E+02	-1.188E-04	2.186E-01	-5.436E-04	-1.087E+05
41	2.554E-01	0	-1.708E+02	-1.188E-04	2.186E-01	-5.437E-04	-1.087E+05
42	2.554E-01	0	-1.709E+02	-1.189E-04	2.186E-01	-5.438E-04	-1.088E+05
43	2.555E-01	0	-1.709E+02	-1.189E-04	2.186E-01	-5.439E-04	-1.088E+05
44	2.555E-01	0	-1.709E+02	-1.189E-04	2.186E-01	-5.440E-04	-1.088E+05
45	2.556E-01	0	-1.710E+02	-1.189E-04	2.186E-01	-5.442E-04	-1.088E+05
46	2.556E-01	0	-1.710E+02	-1.190E-04	2.186E-01	-5.443E-04	-1.089E+05
47	2.557E-01	0	-1.710E+02	-1.190E-04	2.186E-01	-5.444E-04	-1.089E+05
48	2.557E-01	0	-1.711E+02	-1.190E-04	2.186E-01	-5.445E-04	-1.089E+05
49	2.558E-01	0	-1.711E+02	-1.190E-04	2.186E-01	-5.446E-04	-1.089E+05
50	2.558E-01	0	-1.712E+02	-1.191E-04	2.186E-01	-5.447E-04	-1.089E+05
51	2.559E-01	0	-1.712E+02	-1.191E-04	2.186E-01	-5.448E-04	-1.090E+05
52	2.559E-01	0	-1.712E+02	-1.191E-04	2.186E-01	-5.449E-04	-1.090E+05
53	2.560E-01	0	-1.713E+02	-1.191E-04	2.186E-01	-5.451E-04	-1.090E+05
54	2.560E-01	0	-1.713E+02	-1.192E-04	2.186E-01	-5.452E-04	-1.090E+05
55	2.561E-01	0	-1.713E+02	-1.192E-04	2.186E-01	-5.453E-04	-1.091E+05
56	2.561E-01	0	-1.714E+02	-1.192E-04	2.186E-01	-5.454E-04	-1.091E+05
57	2.562E-01	0	-1.714E+02	-1.192E-04	2.186E-01	-5.455E-04	-1.091E+05
58	2.562E-01	0	-1.714E+02	-1.193E-04	2.186E-01	-5.456E-04	-1.091E+05
59	2.563E-01	0	-1.715E+02	-1.193E-04	2.186E-01	-5.457E-04	-1.091E+05

60	2.563E-01	0	-1.715E+02	-1.193E-04	2.186E-01	-5.459E-04	-1.092E+05
61	2.564E-01	0	-1.715E+02	-1.193E-04	2.186E-01	-5.460E-04	-1.092E+05
62	2.564E-01	0	-1.716E+02	-1.194E-04	2.186E-01	-5.461E-04	-1.092E+05
63	2.565E-01	0	-1.716E+02	-1.194E-04	2.186E-01	-5.462E-04	-1.092E+05
64	2.565E-01	0	-1.716E+02	-1.194E-04	2.186E-01	-5.463E-04	-1.093E+05
65	2.566E-01	0	-1.717E+02	-1.194E-04	2.186E-01	-5.464E-04	-1.093E+05
66	2.566E-01	0	-1.717E+02	-1.195E-04	2.186E-01	-5.465E-04	-1.093E+05
67	2.567E-01	0	-1.718E+02	-1.195E-04	2.186E-01	-5.466E-04	-1.093E+05
68	2.567E-01	0	-1.718E+02	-1.195E-04	2.186E-01	-5.468E-04	-1.094E+05
69	2.568E-01	0	-1.718E+02	-1.195E-04	2.186E-01	-5.469E-04	-1.094E+05
70	2.568E-01	0	-1.719E+02	-1.196E-04	2.186E-01	-5.470E-04	-1.094E+05
71	2.569E-01	0	-1.719E+02	-1.196E-04	2.186E-01	-5.471E-04	-1.094E+05
72	2.569E-01	0	-1.719E+02	-1.196E-04	2.186E-01	-5.472E-04	-1.094E+05
73	2.570E-01	0	-1.720E+02	-1.196E-04	2.186E-01	-5.473E-04	-1.095E+05
74	2.570E-01	0	-1.720E+02	-1.197E-04	2.186E-01	-5.474E-04	-1.095E+05
75	2.571E-01	0	-1.720E+02	-1.197E-04	2.186E-01	-5.476E-04	-1.095E+05
76	2.571E-01	0	-1.721E+02	-1.197E-04	2.186E-01	-5.477E-04	-1.095E+05
77	2.572E-01	0	-1.721E+02	-1.197E-04	2.186E-01	-5.478E-04	-1.096E+05
78	2.572E-01	0	-1.721E+02	-1.198E-04	2.186E-01	-5.479E-04	-1.096E+05
79	2.573E-01	0	-1.722E+02	-1.198E-04	2.186E-01	-5.480E-04	-1.096E+05
80	2.573E-01	0	-1.722E+02	-1.198E-04	2.186E-01	-5.481E-04	-1.096E+05

HISTORY FOR ELEMENT GROUP 1 ELEMENT NUMBER 2

STEEL TRUSS ELEMENT (TYPE 11). NODES = 5 8

Step No.	Time/ Load Fact.	Yield Code	Static Force	Total Extension	Hinge Length	Current Strain	Current Stress
1	6.615E-08	0	-4.419E+00	-3.074E-06	2.186E-01	-1.407E-05	-2.813E+03
1	2.540E-03	0	-2.666E+00	-1.854E-06	2.186E-01	-8.484E-06	-1.697E+03
1	6.400E-03	0	3.414E-05	2.375E-11	2.186E-01	1.087E-10	2.173E-02
1	6.451E-03	0	4.100E-02	2.852E-08	2.186E-01	1.305E-07	2.610E+01
1	6.925E-03	0	9.573E-01	6.660E-07	2.186E-01	3.047E-06	6.094E+02
2	1.349E-02	0	1.364E+01	9.488E-06	2.186E-01	4.341E-05	8.681E+03
3	2.005E-02	0	2.632E+01	1.831E-05	2.186E-01	8.377E-05	1.675E+04
4	2.662E-02	0	3.900E+01	2.713E-05	2.186E-01	1.241E-04	2.483E+04
5	3.318E-02	0	5.168E+01	3.596E-05	2.186E-01	1.645E-04	3.290E+04
6	3.974E-02	0	6.436E+01	4.478E-05	2.186E-01	2.049E-04	4.097E+04
7	4.631E-02	0	7.705E+01	5.360E-05	2.186E-01	2.452E-04	4.904E+04
8	5.287E-02	0	8.973E+01	6.242E-05	2.186E-01	2.856E-04	5.711E+04
9	5.943E-02	0	1.024E+02	7.124E-05	2.186E-01	3.259E-04	6.519E+04
10	6.600E-02	0	1.151E+02	8.007E-05	2.186E-01	3.663E-04	7.326E+04
11	7.256E-02	0	1.278E+02	8.889E-05	2.186E-01	4.067E-04	8.133E+04
12	7.912E-02	0	1.405E+02	9.771E-05	2.186E-01	4.470E-04	8.940E+04
13	8.569E-02	0	1.531E+02	1.065E-04	2.186E-01	4.874E-04	9.747E+04
14	9.225E-02	0	1.658E+02	1.154E-04	2.186E-01	5.277E-04	1.055E+05
15	9.881E-02	0	1.785E+02	1.242E-04	2.186E-01	5.681E-04	1.136E+05
16	1.054E-01	0	1.912E+02	1.330E-04	2.186E-01	6.085E-04	1.217E+05
17	1.119E-01	0	2.039E+02	1.418E-04	2.186E-01	6.488E-04	1.298E+05
18	1.185E-01	0	2.165E+02	1.506E-04	2.186E-01	6.892E-04	1.378E+05
19	1.251E-01	0	2.292E+02	1.595E-04	2.186E-01	7.295E-04	1.459E+05
20	1.316E-01	0	2.419E+02	1.683E-04	2.186E-01	7.699E-04	1.540E+05

21	1.382E-01	0	2.546E+02	1.771E-04	2.186E-01	8.103E-04	1.621E+05
22	1.448E-01	0	2.673E+02	1.859E-04	2.186E-01	8.506E-04	1.701E+05
23	1.513E-01	0	2.799E+02	1.948E-04	2.186E-01	8.910E-04	1.782E+05
24	1.579E-01	0	2.926E+02	2.036E-04	2.186E-01	9.313E-04	1.863E+05
25	1.645E-01	0	3.053E+02	2.124E-04	2.186E-01	9.717E-04	1.943E+05
26	1.710E-01	0	3.180E+02	2.212E-04	2.186E-01	1.012E-03	2.024E+05
27	1.776E-01	0	3.307E+02	2.300E-04	2.186E-01	1.052E-03	2.105E+05
28	1.841E-01	0	3.434E+02	2.389E-04	2.186E-01	1.093E-03	2.186E+05
29	1.907E-01	0	3.560E+02	2.477E-04	2.186E-01	1.133E-03	2.266E+05
30	1.973E-01	0	3.687E+02	2.565E-04	2.186E-01	1.174E-03	2.347E+05
31	2.038E-01	0	3.814E+02	2.653E-04	2.186E-01	1.214E-03	2.428E+05
32	2.104E-01	0	3.941E+02	2.742E-04	2.186E-01	1.254E-03	2.508E+05
33	2.170E-01	0	4.068E+02	2.830E-04	2.186E-01	1.295E-03	2.589E+05
34	2.235E-01	0	4.194E+02	2.918E-04	2.186E-01	1.335E-03	2.670E+05
35	2.301E-01	0	4.321E+02	3.006E-04	2.186E-01	1.375E-03	2.751E+05
36	2.366E-01	0	4.448E+02	3.094E-04	2.186E-01	1.416E-03	2.831E+05
37	2.432E-01	0	4.575E+02	3.183E-04	2.186E-01	1.456E-03	2.912E+05
38	2.498E-01	0	4.702E+02	3.271E-04	2.186E-01	1.496E-03	2.993E+05
39	2.552E-01	1	4.807E+02	3.344E-04	3.186E-01	5.695E-04	3.060E+05
39	2.553E-01	1	4.807E+02	3.437E-04	3.186E-01	6.274E-04	3.060E+05
40	2.553E-01	1	4.808E+02	3.992E-04	3.189E-01	9.732E-04	3.061E+05
41	2.554E-01	1	4.809E+02	4.546E-04	3.192E-01	1.318E-03	3.061E+05
42	2.554E-01	1	4.810E+02	5.101E-04	3.196E-01	1.663E-03	3.062E+05
43	2.555E-01	1	4.811E+02	5.656E-04	3.199E-01	2.006E-03	3.063E+05
44	2.555E-01	1	4.812E+02	6.210E-04	3.202E-01	2.350E-03	3.063E+05
45	2.556E-01	1	4.813E+02	6.765E-04	3.205E-01	2.692E-03	3.064E+05
46	2.556E-01	1	4.814E+02	7.320E-04	3.208E-01	3.034E-03	3.064E+05
47	2.557E-01	1	4.815E+02	7.874E-04	3.211E-01	3.375E-03	3.065E+05
48	2.557E-01	1	4.816E+02	8.429E-04	3.214E-01	3.716E-03	3.066E+05
49	2.558E-01	1	4.817E+02	8.984E-04	3.217E-01	4.055E-03	3.066E+05
50	2.558E-01	1	4.818E+02	9.538E-04	3.220E-01	4.395E-03	3.067E+05
51	2.559E-01	1	4.819E+02	1.009E-03	3.223E-01	4.733E-03	3.068E+05
52	2.559E-01	1	4.820E+02	1.065E-03	3.226E-01	5.071E-03	3.068E+05
53	2.560E-01	1	4.821E+02	1.120E-03	3.229E-01	5.409E-03	3.069E+05
54	2.560E-01	1	4.822E+02	1.176E-03	3.232E-01	5.745E-03	3.069E+05
55	2.561E-01	1	4.823E+02	1.231E-03	3.235E-01	6.081E-03	3.070E+05
56	2.561E-01	1	4.824E+02	1.287E-03	3.238E-01	6.417E-03	3.071E+05
57	2.562E-01	1	4.825E+02	1.342E-03	3.241E-01	6.752E-03	3.071E+05
58	2.562E-01	1	4.826E+02	1.398E-03	3.244E-01	7.086E-03	3.072E+05
59	2.563E-01	1	4.827E+02	1.453E-03	3.247E-01	7.420E-03	3.073E+05
60	2.563E-01	1	4.828E+02	1.508E-03	3.250E-01	7.753E-03	3.073E+05
61	2.564E-01	1	4.829E+02	1.564E-03	3.253E-01	8.085E-03	3.074E+05
62	2.564E-01	1	4.830E+02	1.619E-03	3.256E-01	8.417E-03	3.074E+05
63	2.565E-01	1	4.831E+02	1.675E-03	3.259E-01	8.748E-03	3.075E+05
64	2.565E-01	1	4.832E+02	1.730E-03	3.262E-01	9.078E-03	3.076E+05
65	2.566E-01	1	4.833E+02	1.786E-03	3.265E-01	9.408E-03	3.076E+05
66	2.566E-01	1	4.834E+02	1.841E-03	3.268E-01	9.738E-03	3.077E+05
67	2.567E-01	1	4.835E+02	1.897E-03	3.271E-01	1.007E-02	3.078E+05
68	2.567E-01	1	4.836E+02	1.952E-03	3.274E-01	1.039E-02	3.078E+05
69	2.568E-01	1	4.837E+02	2.008E-03	3.277E-01	1.072E-02	3.079E+05
70	2.568E-01	1	4.838E+02	2.063E-03	3.280E-01	1.105E-02	3.079E+05
71	2.569E-01	1	4.839E+02	2.119E-03	3.283E-01	1.138E-02	3.080E+05
72	2.569E-01	1	4.840E+02	2.174E-03	3.286E-01	1.170E-02	3.081E+05
73	2.570E-01	1	4.841E+02	2.230E-03	3.289E-01	1.203E-02	3.081E+05

74	2.570E-01	1	4.842E+02	2.285E-03	3.292E-01	1.235E-02	3.082E+05
75	2.571E-01	1	4.843E+02	2.340E-03	3.295E-01	1.267E-02	3.082E+05
76	2.571E-01	1	4.844E+02	2.396E-03	3.298E-01	1.300E-02	3.083E+05
77	2.572E-01	1	4.845E+02	2.451E-03	3.301E-01	1.332E-02	3.084E+05
78	2.572E-01	1	4.846E+02	2.507E-03	3.304E-01	1.364E-02	3.084E+05
79	2.573E-01	1	4.847E+02	2.562E-03	3.307E-01	1.396E-02	3.085E+05
80	2.573E-01	1	4.847E+02	2.618E-03	3.310E-01	1.429E-02	3.086E+05

HISTORY FOR ELEMENT GROUP 2 ELEMENT NUMBER 1

CONCRETE TRUSS ELEMENT (TYPE 12). NODES = 3 6

Step No.	Time/ Load Fact.	Yield Code	Static Force	Total Extension
1	6.615E-08	4	5.700E-02	8.563E-06
1	2.540E-03	6	5.682E-02	5.149E-06
1	6.400E-03	6	-4.224E+00	2.938E-06
1	6.451E-03	6	-4.272E+00	2.914E-06
1	6.925E-03	6	-4.721E+00	2.682E-06
2	1.349E-02	6	-1.094E+01	-5.292E-07
3	2.005E-02	6	-1.716E+01	-3.740E-06
4	2.662E-02	6	-2.337E+01	-6.951E-06
5	3.318E-02	6	-2.959E+01	-1.016E-05
6	3.974E-02	6	-3.581E+01	-1.337E-05
7	4.631E-02	6	-4.202E+01	-1.658E-05
8	5.287E-02	6	-4.824E+01	-1.979E-05
9	5.943E-02	6	-5.446E+01	-2.300E-05
10	6.600E-02	6	-6.068E+01	-2.622E-05
11	7.256E-02	6	-6.689E+01	-2.943E-05
12	7.912E-02	6	-7.311E+01	-3.264E-05
13	8.569E-02	6	-7.933E+01	-3.585E-05
14	9.225E-02	6	-8.554E+01	-3.906E-05
15	9.881E-02	6	-9.176E+01	-4.227E-05
16	1.054E-01	6	-9.798E+01	-4.548E-05
17	1.119E-01	6	-1.042E+02	-4.869E-05
18	1.185E-01	6	-1.104E+02	-5.190E-05
19	1.251E-01	6	-1.166E+02	-5.511E-05
20	1.316E-01	6	-1.228E+02	-5.832E-05
21	1.382E-01	6	-1.291E+02	-6.153E-05
22	1.448E-01	6	-1.353E+02	-6.475E-05
23	1.513E-01	6	-1.415E+02	-6.796E-05
24	1.579E-01	6	-1.477E+02	-7.117E-05
25	1.645E-01	6	-1.539E+02	-7.438E-05
26	1.710E-01	6	-1.602E+02	-7.759E-05
27	1.776E-01	6	-1.664E+02	-8.080E-05
28	1.841E-01	6	-1.726E+02	-8.401E-05
29	1.907E-01	6	-1.788E+02	-8.722E-05
30	1.973E-01	6	-1.850E+02	-9.043E-05
31	2.038E-01	6	-1.912E+02	-9.364E-05
32	2.104E-01	6	-1.975E+02	-9.685E-05
33	2.170E-01	6	-2.037E+02	-1.001E-04
34	2.235E-01	6	-2.099E+02	-1.033E-04

35	2.301E-01	6	-2.161E+02	-1.065E-04
36	2.366E-01	6	-2.223E+02	-1.097E-04
37	2.432E-01	6	-2.285E+02	-1.129E-04
38	2.498E-01	6	-2.348E+02	-1.161E-04
39	2.552E-01	6	-2.399E+02	-1.188E-04
39	2.553E-01	6	-2.399E+02	-1.188E-04
40	2.553E-01	6	-2.400E+02	-1.188E-04
41	2.554E-01	6	-2.400E+02	-1.188E-04
42	2.554E-01	6	-2.401E+02	-1.189E-04
43	2.555E-01	6	-2.401E+02	-1.189E-04
44	2.555E-01	6	-2.402E+02	-1.189E-04
45	2.556E-01	6	-2.402E+02	-1.189E-04
46	2.556E-01	6	-2.403E+02	-1.190E-04
47	2.557E-01	6	-2.403E+02	-1.190E-04
48	2.557E-01	6	-2.404E+02	-1.190E-04
49	2.558E-01	6	-2.404E+02	-1.190E-04
50	2.558E-01	6	-2.405E+02	-1.191E-04
51	2.559E-01	6	-2.405E+02	-1.191E-04
52	2.559E-01	6	-2.406E+02	-1.191E-04
53	2.560E-01	6	-2.406E+02	-1.191E-04
54	2.560E-01	6	-2.407E+02	-1.192E-04
55	2.561E-01	6	-2.407E+02	-1.192E-04
56	2.561E-01	6	-2.408E+02	-1.192E-04
57	2.562E-01	6	-2.408E+02	-1.192E-04
58	2.562E-01	6	-2.409E+02	-1.193E-04
59	2.563E-01	6	-2.409E+02	-1.193E-04
60	2.563E-01	6	-2.410E+02	-1.193E-04
61	2.564E-01	6	-2.410E+02	-1.193E-04
62	2.564E-01	6	-2.410E+02	-1.194E-04
63	2.565E-01	6	-2.411E+02	-1.194E-04
64	2.565E-01	6	-2.411E+02	-1.194E-04
65	2.566E-01	6	-2.412E+02	-1.194E-04
66	2.566E-01	6	-2.412E+02	-1.195E-04
67	2.567E-01	6	-2.413E+02	-1.195E-04
68	2.567E-01	6	-2.413E+02	-1.195E-04
69	2.568E-01	6	-2.414E+02	-1.195E-04
70	2.568E-01	6	-2.414E+02	-1.196E-04
71	2.569E-01	6	-2.415E+02	-1.196E-04
72	2.569E-01	6	-2.415E+02	-1.196E-04
73	2.570E-01	6	-2.416E+02	-1.196E-04
74	2.570E-01	6	-2.416E+02	-1.197E-04
75	2.571E-01	6	-2.417E+02	-1.197E-04
76	2.571E-01	6	-2.417E+02	-1.197E-04
77	2.572E-01	6	-2.418E+02	-1.197E-04
78	2.572E-01	6	-2.418E+02	-1.198E-04
79	2.573E-01	6	-2.419E+02	-1.198E-04
80	2.573E-01	6	-2.419E+02	-1.198E-04

HISTORY FOR ELEMENT GROUP 2 ELEMENT NUMBER 2

CONCRETE TRUSS ELEMENT (TYPE 12). NODES = 5 8

Step No.	Time/ Load Fact.	Yield Code	Static Force	Total Extension
1	6.615E-08	0	-6.143E+00	-3.074E-06
1	2.540E-03	0	-3.705E+00	-1.854E-06
1	6.400E-03	0	4.746E-05	2.375E-11
1	6.451E-03	3	5.698E-02	2.852E-08
1	6.925E-03	3	5.698E-02	6.660E-07
2	1.349E-02	3	5.700E-02	9.488E-06
3	2.005E-02	3	5.702E-02	1.831E-05
4	2.662E-02	3	5.704E-02	2.713E-05
5	3.318E-02	3	5.705E-02	3.596E-05
6	3.974E-02	3	5.707E-02	4.478E-05
7	4.631E-02	3	5.709E-02	5.360E-05
8	5.287E-02	3	5.711E-02	6.242E-05
9	5.943E-02	3	5.713E-02	7.124E-05
10	6.600E-02	3	5.714E-02	8.007E-05
11	7.256E-02	3	5.716E-02	8.889E-05
12	7.912E-02	3	5.718E-02	9.771E-05
13	8.569E-02	3	5.720E-02	1.065E-04
14	9.225E-02	3	5.721E-02	1.154E-04
15	9.881E-02	3	5.723E-02	1.242E-04
16	1.054E-01	3	5.725E-02	1.330E-04
17	1.119E-01	3	5.727E-02	1.418E-04
18	1.185E-01	3	5.728E-02	1.506E-04
19	1.251E-01	3	5.730E-02	1.595E-04
20	1.316E-01	3	5.732E-02	1.683E-04
21	1.382E-01	3	5.734E-02	1.771E-04
22	1.448E-01	3	5.735E-02	1.859E-04
23	1.513E-01	3	5.737E-02	1.948E-04
24	1.579E-01	3	5.739E-02	2.036E-04
25	1.645E-01	3	5.741E-02	2.124E-04
26	1.710E-01	3	5.742E-02	2.212E-04
27	1.776E-01	3	5.744E-02	2.300E-04
28	1.841E-01	3	5.746E-02	2.389E-04
29	1.907E-01	3	5.748E-02	2.477E-04
30	1.973E-01	3	5.750E-02	2.565E-04
31	2.038E-01	3	5.751E-02	2.653E-04
32	2.104E-01	3	5.753E-02	2.742E-04
33	2.170E-01	3	5.755E-02	2.830E-04
34	2.235E-01	3	5.757E-02	2.918E-04
35	2.301E-01	3	5.758E-02	3.006E-04
36	2.366E-01	3	5.760E-02	3.094E-04
37	2.432E-01	3	5.762E-02	3.183E-04
38	2.498E-01	3	5.764E-02	3.271E-04
39	2.552E-01	3	5.765E-02	3.344E-04
39	2.553E-01	3	5.767E-02	3.437E-04
40	2.553E-01	3	5.778E-02	3.992E-04
41	2.554E-01	3	5.789E-02	4.546E-04
42	2.554E-01	3	5.800E-02	5.101E-04

43	2.555E-01	3	5.811E-02	5.656E-04
44	2.555E-01	3	5.822E-02	6.210E-04
45	2.556E-01	3	5.833E-02	6.765E-04
46	2.556E-01	3	5.845E-02	7.320E-04
47	2.557E-01	3	5.856E-02	7.874E-04
48	2.557E-01	3	5.867E-02	8.429E-04
49	2.558E-01	3	5.878E-02	8.984E-04
50	2.558E-01	3	5.889E-02	9.538E-04
51	2.559E-01	3	5.900E-02	1.009E-03
52	2.559E-01	3	5.911E-02	1.065E-03
53	2.560E-01	3	5.922E-02	1.120E-03
54	2.560E-01	3	5.933E-02	1.176E-03
55	2.561E-01	3	5.944E-02	1.231E-03
56	2.561E-01	3	5.955E-02	1.287E-03
57	2.562E-01	3	5.966E-02	1.342E-03
58	2.562E-01	3	5.978E-02	1.398E-03
59	2.563E-01	3	5.989E-02	1.453E-03
60	2.563E-01	3	6.000E-02	1.508E-03
61	2.564E-01	3	6.011E-02	1.564E-03
62	2.564E-01	3	6.022E-02	1.619E-03
63	2.565E-01	3	6.033E-02	1.675E-03
64	2.565E-01	3	6.044E-02	1.730E-03
65	2.566E-01	3	6.055E-02	1.786E-03
66	2.566E-01	3	6.066E-02	1.841E-03
67	2.567E-01	3	6.077E-02	1.897E-03
68	2.567E-01	3	6.088E-02	1.952E-03
69	2.568E-01	3	6.099E-02	2.008E-03
70	2.568E-01	3	6.111E-02	2.063E-03
71	2.569E-01	3	6.122E-02	2.119E-03
72	2.569E-01	3	6.133E-02	2.174E-03
73	2.570E-01	3	6.144E-02	2.230E-03
74	2.570E-01	3	6.155E-02	2.285E-03
75	2.571E-01	3	6.166E-02	2.340E-03
76	2.571E-01	3	6.177E-02	2.396E-03
77	2.572E-01	3	6.188E-02	2.451E-03
78	2.572E-01	3	6.199E-02	2.507E-03
79	2.573E-01	3	6.210E-02	2.562E-03
80	2.573E-01	3	6.221E-02	2.618E-03

HISTORY FOR ELEMENT GROUP 3 ELEMENT NUMBER 1

SHEAR ELEMENT (TYPE 15). NODES = 4 7

Step No.	Time/ Load Fact.	Yield Code	Shear Force	Total Shear Deformation	Moment at end of Step	Rotation at end of Step
1	6.615E-08	1	-3.200E+00	-1.778E-05	-4.798E+00	-2.999E-05
1	2.540E-03	1	-1.930E+00	-1.072E-05	-2.894E+00	-1.805E-05
1	6.400E-03	0	2.217E-05	1.232E-10	3.786E-05	-7.573E-06
1	6.451E-03	0	2.536E-02	1.409E-07	3.766E-02	-7.436E-06
1	6.925E-03	0	2.625E-01	1.458E-06	3.938E-01	-5.195E-06
2	1.349E-02	0	3.544E+00	1.969E-05	5.314E+00	2.582E-05
3	2.005E-02	0	6.826E+00	3.792E-05	1.023E+01	5.683E-05

4	2.662E-02	0	1.011E+01	5.615E-05	1.515E+01	8.784E-05
5	3.318E-02	0	1.339E+01	7.438E-05	2.007E+01	1.189E-04
6	3.974E-02	0	1.667E+01	9.261E-05	2.500E+01	1.499E-04
7	4.631E-02	0	1.995E+01	1.108E-04	2.992E+01	1.809E-04
8	5.287E-02	0	2.323E+01	1.291E-04	3.484E+01	2.119E-04
9	5.943E-02	0	2.652E+01	1.473E-04	3.976E+01	2.429E-04
10	6.600E-02	0	2.980E+01	1.655E-04	4.468E+01	2.739E-04
11	7.256E-02	0	3.308E+01	1.838E-04	4.960E+01	3.049E-04
12	7.912E-02	0	3.636E+01	2.020E-04	5.452E+01	3.360E-04
13	8.569E-02	0	3.964E+01	2.202E-04	5.944E+01	3.670E-04
14	9.225E-02	0	4.293E+01	2.385E-04	6.436E+01	3.980E-04
15	9.881E-02	0	4.621E+01	2.567E-04	6.928E+01	4.290E-04
16	1.054E-01	0	4.949E+01	2.749E-04	7.420E+01	4.600E-04
17	1.119E-01	0	5.277E+01	2.931E-04	7.912E+01	4.910E-04
18	1.185E-01	0	5.605E+01	3.114E-04	8.404E+01	5.220E-04
19	1.251E-01	0	5.933E+01	3.296E-04	8.896E+01	5.530E-04
20	1.316E-01	0	6.262E+01	3.478E-04	9.388E+01	5.841E-04
21	1.382E-01	0	6.590E+01	3.661E-04	9.880E+01	6.151E-04
22	1.448E-01	0	6.918E+01	3.843E-04	1.037E+02	6.461E-04
23	1.513E-01	0	7.246E+01	4.025E-04	1.086E+02	6.771E-04
24	1.579E-01	0	7.574E+01	4.208E-04	1.136E+02	7.081E-04
25	1.645E-01	0	7.903E+01	4.390E-04	1.185E+02	7.391E-04
26	1.710E-01	0	8.231E+01	4.572E-04	1.234E+02	7.701E-04
27	1.776E-01	0	8.559E+01	4.754E-04	1.283E+02	8.011E-04
28	1.841E-01	0	8.887E+01	4.937E-04	1.332E+02	8.322E-04
29	1.907E-01	0	9.215E+01	5.119E-04	1.382E+02	8.632E-04
30	1.973E-01	0	9.543E+01	5.301E-04	1.431E+02	8.942E-04
31	2.038E-01	0	9.872E+01	5.484E-04	1.480E+02	9.252E-04
32	2.104E-01	0	1.020E+02	5.666E-04	1.529E+02	9.562E-04
33	2.170E-01	0	1.053E+02	5.848E-04	1.578E+02	9.872E-04
34	2.235E-01	0	1.086E+02	6.031E-04	1.628E+02	1.018E-03
35	2.301E-01	0	1.118E+02	6.213E-04	1.677E+02	1.049E-03
36	2.366E-01	0	1.151E+02	6.395E-04	1.726E+02	1.080E-03
37	2.432E-01	0	1.184E+02	6.577E-04	1.775E+02	1.111E-03
38	2.498E-01	0	1.217E+02	6.760E-04	1.824E+02	1.142E-03
39	2.552E-01	0	1.244E+02	6.912E-04	1.865E+02	1.168E-03
39	2.553E-01	0	1.244E+02	6.912E-04	1.866E+02	1.192E-03
40	2.553E-01	0	1.245E+02	6.913E-04	1.866E+02	1.335E-03
41	2.554E-01	0	1.245E+02	6.915E-04	1.866E+02	1.478E-03
42	2.554E-01	0	1.245E+02	6.916E-04	1.867E+02	1.621E-03
43	2.555E-01	0	1.245E+02	6.917E-04	1.867E+02	1.764E-03
44	2.555E-01	0	1.246E+02	6.919E-04	1.867E+02	1.907E-03
45	2.556E-01	0	1.246E+02	6.920E-04	1.868E+02	2.050E-03
46	2.556E-01	0	1.246E+02	6.922E-04	1.868E+02	2.193E-03
47	2.557E-01	0	1.246E+02	6.923E-04	1.869E+02	2.336E-03
48	2.557E-01	0	1.247E+02	6.924E-04	1.869E+02	2.479E-03
49	2.558E-01	0	1.247E+02	6.926E-04	1.869E+02	2.622E-03
50	2.558E-01	0	1.247E+02	6.927E-04	1.870E+02	2.765E-03
51	2.559E-01	0	1.247E+02	6.929E-04	1.870E+02	2.908E-03
52	2.559E-01	0	1.248E+02	6.930E-04	1.870E+02	3.051E-03
53	2.560E-01	0	1.248E+02	6.932E-04	1.871E+02	3.194E-03
54	2.560E-01	0	1.248E+02	6.933E-04	1.871E+02	3.337E-03
55	2.561E-01	0	1.248E+02	6.934E-04	1.872E+02	3.480E-03
56	2.561E-01	0	1.249E+02	6.936E-04	1.872E+02	3.623E-03

57	2.562E-01	0	1.249E+02	6.937E-04	1.872E+02	3.766E-03
58	2.562E-01	0	1.249E+02	6.939E-04	1.873E+02	3.909E-03
59	2.563E-01	0	1.249E+02	6.940E-04	1.873E+02	4.052E-03
60	2.563E-01	0	1.250E+02	6.941E-04	1.873E+02	4.195E-03
61	2.564E-01	0	1.250E+02	6.943E-04	1.874E+02	4.338E-03
62	2.564E-01	0	1.250E+02	6.944E-04	1.874E+02	4.481E-03
63	2.565E-01	0	1.250E+02	6.946E-04	1.875E+02	4.624E-03
64	2.565E-01	0	1.251E+02	6.947E-04	1.875E+02	4.767E-03
65	2.566E-01	0	1.251E+02	6.948E-04	1.875E+02	4.910E-03
66	2.566E-01	0	1.251E+02	6.950E-04	1.876E+02	5.053E-03
67	2.567E-01	0	1.251E+02	6.951E-04	1.876E+02	5.196E-03
68	2.567E-01	0	1.252E+02	6.953E-04	1.877E+02	5.339E-03
69	2.568E-01	0	1.252E+02	6.954E-04	1.877E+02	5.482E-03
70	2.568E-01	0	1.252E+02	6.955E-04	1.877E+02	5.625E-03
71	2.569E-01	0	1.252E+02	6.957E-04	1.878E+02	5.768E-03
72	2.569E-01	0	1.253E+02	6.958E-04	1.878E+02	5.912E-03
73	2.570E-01	0	1.253E+02	6.960E-04	1.878E+02	6.055E-03
74	2.570E-01	0	1.253E+02	6.961E-04	1.879E+02	6.198E-03
75	2.571E-01	0	1.253E+02	6.962E-04	1.879E+02	6.341E-03
76	2.571E-01	0	1.254E+02	6.964E-04	1.880E+02	6.484E-03
77	2.572E-01	0	1.254E+02	6.965E-04	1.880E+02	6.627E-03
78	2.572E-01	0	1.254E+02	6.967E-04	1.880E+02	6.770E-03
79	2.573E-01	0	1.254E+02	6.968E-04	1.881E+02	6.913E-03
80	2.573E-01	0	1.255E+02	6.969E-04	1.881E+02	7.056E-03