3. THE RADIAL TENT

(The JAVA version of this chapter by Keith MacBain is highly recommended: http://www-ec.njit.edu/civil/gateway.html)

The radial tent is a special case of the cable net, see Figure 3.1, in which the shape of the base is specified along with the vertical force components in the vertical cables and the force components at one point in each of the circumferential cables. The remaining radial coordinates are then computed iteratively to satisfy equilibrium. The *taper* of the tent is controlled by the forces in the circumferential cables.

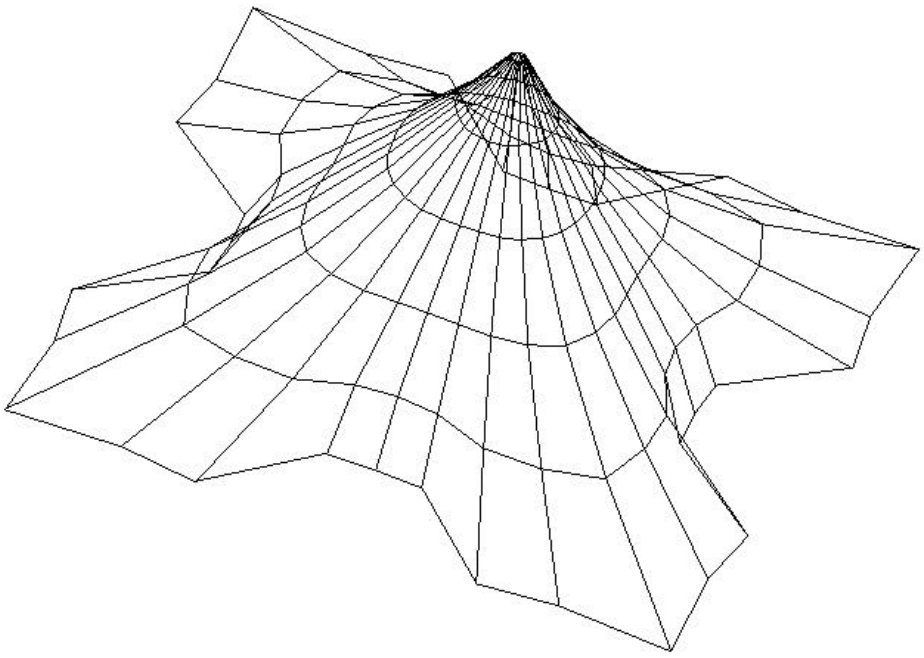
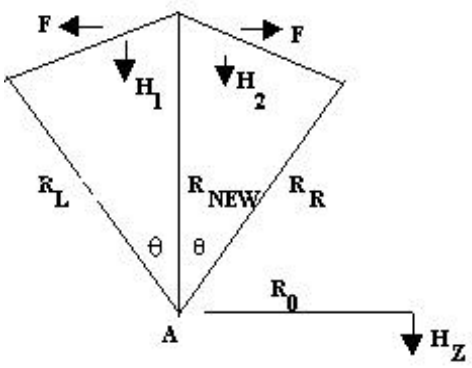
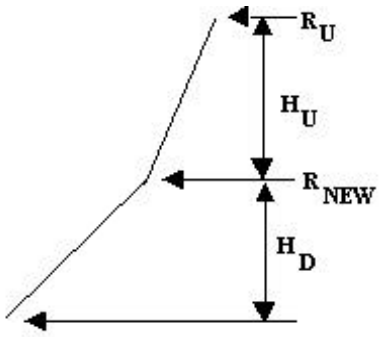


Figure 3.1

Consider a typical node in the radial tent. Horizontal and vertical sections through this node are shown in Figures 3.2 and 3.3.



Horizontal Section
Figure 3.2



Vertical Section
Figure 3.3

Sum moments about point A

$$F\ R_{NEW} = R_0\ H_Z$$

F => H_1 ,H_2 from the slopes

$F / H_1 = R_L \sin\Theta / (R_{NEW} - R_L \cos\Theta)$

$$H_1 = \frac{R_0 H_Z}{R_{NEW}} \frac{R_{NEW} - R_L \cos q}{R_L \sin q}$$

$$H_2 = \frac{R_0 H_Z}{R_{NEW}} \frac{R_{NEW} - R_R \cos q}{R_R \sin q}$$

$$H_3 = \frac{V(R_{NEW} - R_U)}{H_U}$$

$$H_4 = \frac{V(R_D - R_{NEW})}{H_D}$$

Vertical force components must sum to zero

$$H_1 + H_2 + H_3 + H_4 = 0$$

This gives a quadratic for R_{NEW} that is used in the computer program.

Solving the radial tent is described in the batch program RADLAY.BAT

Radlay.bat

```
radtent radtent.dat fort.60 fort.150
copy fort.150 fort.50
lplot1 fort.50 fort.150
copy fort.150 fort.50
layoutpl fort.50 fort.150
```

The principal step here is RADTENT.FOR which solves for the radial coordinates as described above. Since the cases considered are very symmetric, radtent works with 1/8th of the structure. LPLOT1.FOR then maps 1/8th to 1/4; LAYOUTPL.FOR then maps 1/4 into the full structure. COPY.FOR simply copies files.

Radtent.for

```
C     RADIAL TENT INPUT
      DIMENSION HT(20), H0(20), ZZ(20,20)
      READ(50,1) NRINGS, NSEGS,V,RAD
      READ(50,2) (HT(I),I=1,NRINGS)
      READ(50,2) (H0(I),I=1,NRINGS)
      write(60,1) NRINGS, NSEGS,V,RAD
      write(60,2) (HT(I),I=1,NRINGS)
      write(60,2) (H0(I),I=1,NRINGS)
      THETA=3.14159/(4.*FLOAT(NSEGS))
      NSG=NSEGS+1
    1 FORMAT(2I5,2F10.3)
    2 FORMAT(6F10.3)
      DO 10 I=1,NSG
      TH=FLOAT(I-1)*THETA
      ZZ(NRINGS,I)=RAD
   10 ZZ(1,I)=50./COS(TH)
      write(*,*) 'Do you want to change the square base?
     1 (yes,no=1,0)'
      read(*,*) iyes
      if(iyes.ne.1) go to 20
C      write(*,*) (zz(1,i),i=1,nsg)
      read(50,*)(zz(1,i),i=1,nsg)
   20 CALL TENT(NRINGS,NSEGS,V,HT,H0,ZZ)
      STOP
      END
C
      SUBROUTINE TENT(NRINGS,NSEGS,V,HT,H0,ZZ)
C     RADIAL TENT PROGRAM
      DIMENSION HT(1),H0(1),ZZ(20,20)
      DIMENSION NP(100),MI(100),R(300),FOR(300)
      DIMENSION IFIX(110),ISYM(100),JT(100,100)
C
C     HT=RING HEIGHT
C     HO=REFERENCE RADIAL FORCE
C     V=VERTICAL FORCE COMPONENT
C     ZZ(RING LEVEL,SEGMENT)=RADIAL COORDINATES
C
      THETA=3.14159/(4.*FLOAT(NSEGS))
      CTN=COS(THETA)/SIN(THETA)
      SI=SIN(THETA)
```

```
    NRING2=NRINGS-2
    NR1=NRINGS-1
    NSEG1=NSEGS+1
    NSG=NSEGS+1
    DO 10 I=2,NR1
    DO 10 J=1,NSEG1
 10 ZZ(I,J)=ZZ(1,J)*float(nrings-i)/float(NRINGS)
    WRITE(60,9)((ZZ(I,J),J=1,NSG),I=1,NRINGS)
    NIT=10
    DO 100 ITER=1,NIT
    DO 100 I1=1,NRING2
    I=NRINGS-I1
    HZ=H0(I)
    R0=ZZ(I,1)
    DO 1 J=1,nsg
    ZU=HT(I+1)
    ZI=HT(I)
    ZD=HT(I-1)
    HU=-(ZI-ZU)
    HD=ZI-ZD
    RU=ZZ(I+1,J)
    RD=ZZ(I-1,J)
    IF(J.EQ.1) RL=ZZ(I,J+1)
    IF(J.EQ.NSG) RR=ZZ(I,J-1)
    IF(J.NE.1) RL=ZZ(I,J-1)
    IF(J.NE.NSG) RR=ZZ(I,J+1)
    RI=ZZ(I,J)
    A=V*(1./HD+1./HU)
    B=R0*HZ*(1./RR+1./RL)/SI-V*(RD/HD+RU/HU)
    C=-2.*R0*HZ*CTN
    write(60,9) a,b,c
    RNEW=(-B+SQRT(B*B-4.*A*C))/(2.*A)
    RATIO=RNEW/ZZ(I,J)
    ZZ(I,J)=RNEW
  1 WRITE(60,9) RATIO,RNEW
  9 FORMAT(5E20.8)
100 CONTINUE
    WRITE(60,9)((ZZ(I,J),J=1,NSG),I=1,NRINGS)
    NN=0
    NB=0
    DO 200 I=1,NRINGS
    DO 200 J=1,NSG
    ANG=FLOAT(J-1)*THETA
    NN=NN+1
    ISYM(nn)=0
    R(3*NN-2)=ZZ(I,J)*SIN(ANG)
    R(3*NN-1)=-ZZ(I,J)*COS(ANG)+50.
    R(3*NN)  =HT(I)
    if(nn.eq.1) isym(nn)=3
    if(nn.ne.1.and .j.eq.1) isym(nn)=2
    if(nn.ne.1.and.j.eq.nsg) isym(nn)=4
    if(isym(nn).eq.0.and .i.eq.1) isym(nn)=1
    IF(NN.EQ.1) GO TO 200
    IF(J.EQ.1) GO TO 201
    NB=NB+1
    NP(NB)=NN
    MI(NB)=NN-1
```

```
  201 IF(I.EQ.1) GO TO 200
      NB=NB+1
      NP(NB)=NN
      MI(NB)=NN-NSG
  200 CONTINUE
      WRITE(150,150)NB,NN
  150 FORMAT(2I5)
      WRITE(150,151) (NP(I),MI(I),FOR(I),I=1,NB)
  151 FORMAT(2I5,E20.8)
      WRITE(150,152)(R(3*I-2),R(3*I-1),R(3*I), IFIX(I),ISYM(I),
     1 (JT(I,J),J=1,8),I=1,NN)
  152 format(3f10.3,2i2,8i5)
c      call splot(np,mi,nn,nb,r,for,0)
c      call splot(np,mi,nn,nb,r,for,1)
c      call splot(np,mi,nn,nb,r,for,2)
      RETURN
      END
```

Radtent.dat

```
    6    4 10.      1.
0.        10.       20.       30.       40.       50.
12.       12.       12.       12.       12.       12.
35. 36. 55. 59. 70.7
```

Lplot1.for

```
      include 'fgraph.fi'
C     GENERAL MEMBRANE LAYOUT PLOT PROGRAM
c     Mapping 1/8 to 1/4
      DIMENSION FORH(400),X(400),Y(400),Z(400),JT(400,4),IFIX(400)
     1 ,NP(400),MI(400),ISYM(400),R(1000),New(1000)
      READ(50,1)NB,NN
      WRITE(6,1)NB,NN
    1 FORMAT(2I5)
      READ(50,222)(NP(I),MI(I),FORH(I),I=1,NB)
  222 FORMAT(2I5,E20.8)
      WRITE(6,2)(FORH(I),NP(I),MI(I),I=1,NB)
    2 FORMAT(f10.3,2I5)
      READ(50,3)(X(I),Y(I),Z(I),IFIX(I),ISYM(I),(JT(I,J),J=1,4),I=1,
     1 NN)
      WRITE(6,3)(X(I),Y(I),Z(I),IFIX(I),ISYM(I),(JT(I,J),J=1,4),I=1,
     1 NN)
    3 FORMAT(3f10.3,2I2,4I5)
c     ISYM = 4 implies a node on the diagonal
C      WRITE(*,*) 'ENTER THE Y OFFSET'
C      READ(*,*)YSHIFT
C      WRITE(*,*)YSHIFT
      YSHIFT=50.
c
c     THIS IS A SPECIAL CASE
      NB0=NB
      NN0=NN
      DO 31 I=1,NN0
      IF(ISYM(I).EQ.4) GO TO 31
      NN=NN+1
      new(i)=nn
```

```
      X(NN)=-y(I)+YSHIFT
      Y(NN)=-x(I)+YSHIFT
      Z(NN)=Z(I)
   31 CONTINUE
      DO 61 I=1,NB0
      IP=NP(I)
      IM=MI(I)
      if(isym(ip).eq.4.and.isym(im).eq.4)go to 61
   62 NB=NB+1
      NP(NB)=IP
      if(isym(ip).ne.4)np(nb)=new(ip)
      MI(NB)=IM
      if(isym(im).ne.4)mi(nb)=new(im)
   61 CONTINUE
      DO 71 I=1,NN
      R(3*I-2)=X(I)
      y(i)=y(i)-YSHIFT
      R(3*I-1)=Y(I)
      R(3*I)  =Z(I)
      Isym(I)=0
      IF(X(I).EQ.0.) Isym(I)=2
      IF(Y(I).EQ.0.) ISYM(I)=1
      IF(X(I).EQ.0..AND.Y(I).EQ.0.) ISYM(I)=3
   71 WRITE(6,72)I, R(3*I-2),R(3*I-1),R(3*I)
      WRITE(150,1)NB,NN
      WRITE(150,222)(NP(I),MI(I),forh(i),I=1,NB)
      WRITE(150,3)(X(I),Y(I),Z(I),IFIX(I),ISYM(I),(JT(I,J),J=1,4),I=1,
     1 NN)
   72 FORMAT(I5,3E20.8)
      CALL SPLOT(NP,MI,NN,NB,R,forh,0)
      STOP
      END
c
      SUBROUTINE PLOT(NB, NN, X, Y, NP, MI,for,iwrite)
      INCLUDE 'FGRAPH.FD'
      DIMENSION  NP(1), MI(1), X(1), Y(1),for(1)
      INTEGER*2 DUMMY,xk,yk,xm,ym,lx,ly
      RECORD /XYCOORD/ XY
c     record /rccoord/ curpos
      character*6  text
      character*15 text1
      CHARACTER*64 FONTPATH
      CHARACTER*20 LIST
      FONTPATH='\f32\lib\courb.fon'
      LIST="t'courb'"//'h10w10b'
      DUMMY = SETVIDEOMODE( $VRES16COLOR)
      DUMMY=REGISTERFONTS(FONTPATH)
      DUMMY=SETFONT(LIST)
      AMAXX=639-20
      AMAYY=479-20
      XMIN=X(1)
      XMAX=X(1)
      YMIN=Y(1)
      YMAX=Y(1)
      DO 2 I=1,NN
      XI=X(I)
      YI=Y(I)
```

```
    IF(XMIN.GT.XI) XMIN=XI
    IF(XMAX.LT.XI) XMAX=XI
    IF(YMIN.GT.YI) YMIN=YI
 2  IF(YMAX.LT.YI) YMAX=YI
    SCALE = AMAX1((XMAX-XMIN)/AMAXX,(YMAX-YMIN)/AMAYY)
    XSHIFT = (XMAX+XMIN)/2.0 - 639/2*SCALE
    YSHIFT = (YMAX+YMIN)/2.0 - 479/2*SCALE
    DO 3 I=1,NB
    K=NP(I)
    M=MI(I)
    XK=(X(K)-XSHIFT)/SCALE
    YK=(Y(K)-YSHIFT)/SCALE
    XM=(X(M)-XSHIFT)/SCALE
    YM=(Y(M)-YSHIFT)/SCALE
    YK = 479-YK
    YM = 479-YM
    LX=((XK+XM)/2)
    LY=((YK+YM)/2)
    CALL MOVETO  ( XK, YK,  XY)
    DUMMY = LINETO  ( XM, YM)
    if(iwrite.ne.2) go to 998
    call  moveto(lx,ly,xy)
    write(text, '(i3)') i
    call outgtext (text)
998 if(iwrite.eq.0.or.iwrite.eq.2) go to 3
    call moveto(lx,ly,xy)
    write(text1,'(f7.0)') for(i)
    call outgtext (text1)
 3  CONTINUE
    if(iwrite.ne.2) go to 996
    do 997 i=1,nn
    lx=(x(i)-xshift)/scale
    yk=(y(i)-yshift)/scale
    ly=(479-yk)
    call  moveto(lx,ly,xy)
    write(text, '(i3)') i
    call outgtext (text)
997 continue
996 continue
    RETURN
    END

    SUBROUTINE SPLOT ( NP,NM,NN,NB,R,for,iwrite)
    INCLUDE 'FGRAPH.FD'
    DIMENSION  NP(1),NM(1),RXY(1000),ROT(3,3),for(1)
    DIMENSION ANGL(3),NT(3),A(3,3),R1(3,3,3)
    INTEGER*2 DUMMY
    DIMENSION R(1),X(200),Y(200),RZ(1000)
    WRITE(*,1)
  1 FORMAT(' YOU ARE ABOUT TO ENTER A GRAPHICS '
   1 'DISPLAY MODE'/' THE KEYBOARD COMMANDS ARE'//
   1 '     +1...POSITIVE ROTATION ABOUT X AXIS'/
   1 '     -1...NEGATIVE ROTATION ABOUT X AXIS'/
   1 '     +2...POSITIVE ROTATION ABOUT Y AXIS'/
   1 '     -2...NEGATIVE ROTATION ABOUT Y AXIS'/
   1 '     +3...POSITIVE ROTATION ABOUT Z AXIS'/
   1 '     -3...NEGATIVE ROTATION ABOUT Z AXIS'/
```

```
   1 '      0...EXIT')
     READ(*,*)
     DO 616 I=1,3
     DO 617 J=1,3
     DO 617 K=1,3
617  R1(I,J,K)=0.
616  R1(I,I,I)=1.
     THX=0.
     THY=00.
     THZ=00.
     DTH=10.
70   PI=3.14159
     DO 604 I=1,3
     DO 603 J=1,3
603  ROT(J,I)=0.
604  ROT(I,I)=1.
     ANGL(1)=THX
     ANGL(2)=THY
     ANGL(3)=THZ
     NT(1)=1
       NT(2)=2
     NT(3)=3
     I=0
302  I=I+1
     IF(ANGL(I))606,605,606
606  L=NT(I)
     GO TO 612
618  DO 607 J=1,3
     DO 607 JA=1,3
     A(J,JA)=0.
     DO 607 JB=1,3
607  A(J,JA)=A(J,JA)+R1(L,J,JB)*ROT(JB,JA)
     DO 608 K=1,3
     DO 608 J=1,3
608  ROT(K,J)=A(K,J)
605  IF(I-3) 302,303,303
303  DO 805 I=1,NN
     RZ(I)=0.
     DO 806 K=1,3
806  RZ(I)=RZ(I)+ROT(3,K)*R(3*I-3+K)
     DO 805 J=1,2
     RXY(2*I-2+J)=0.
     DO 805 K=1,3
805  RXY(2*I-2+J)=RXY(2*I-2+J)+ROT(J,K)*R(3*I-3+K)
     GO TO 59
612  ANG=ANGL(I)*PI/180.
     IF(L-2)613,614,615
613  R1(1,2,2)=COS(ANG)
     R1(1,2,3)=SIN(ANG)
     R1(1,3,3)=R1(1,2,2)
     R1(1,3,2)=-R1(1,2,3)
     GO TO 618
614  R1(2,1,1)=COS(ANG)
     R1(2,1,3)=-SIN(ANG)
     R1(2,3,1)=-R1(2,1,3)
     R1(2,3,3)=R1(2,1,1)
     GO TO 618
```

```
  615 R1(3,1,1)=COS(ANG)
      R1(3,1,2)=SIN(ANG)
      R1(3,2,1)=-R1(3,1,2)
      R1(3,2,2)=R1(3,1,1)
      GO TO 618
   59 DO 24 I=1,NN
      X(I)=RXY(2*I-1)
   24 Y(I)=RXY(2*I)
      CALL PLOT(NB,NN,X,Y,NP,NM,for,iwrite)
      READ(*,*) IVAL
      IF(IVAL.EQ.+1) GO TO 2000
      IF(IVAL.EQ.-1) GO TO 3000
      IF(IVAL.EQ. 2) GO TO 4000
      IF(IVAL.EQ.-2) GO TO 5000
      IF(IVAL.EQ. 3) GO TO 6000
      IF(IVAL.EQ.-3) GO TO 7000
      IF(IVAL.EQ. 0) GO TO 8000
 2000 THX=THX+DTH
      GO TO 70
 3000 THX=THX-DTH
      GO TO 70
 4000 THY=THY+DTH
      GO TO 70
 5000 THY=THY-DTH
      GO TO 70
 6000 THZ=THZ+DTH
      GO TO 70
 7000 THZ=THZ-DTH
      GO TO 70
 8000 CALL UNREGISTERFONTS()
      DUMMY = SETVIDEOMODE( $DEFAULTMODE )
      RETURN
      END

                              Layoutpl.for
      include 'fgraph.fi'
C     GENERAL MEMBRANE LAYOUT PLOT PROGRAM
c     For 1/4 to full layout
      DIMENSION FORH(400),X(400),Y(400),Z(400),JT(400,4),IFIX(400)
     1 ,NP(400),MI(400),ISYM(400),R(1000),NSTART(1000)
      READ(50,1)NB,NN
      WRITE(6,1)NB,NN
    1 FORMAT(2I5)
      READ(50,222)(NP(I),MI(I),forh(i),I=1,NB)
  222 format(2i5,e20.8)
      WRITE(6,2)(FORH(I),NP(I),MI(I),I=1,NB)
    2 FORMAT(F10.2,2I5)
      READ(50,3)(X(I),Y(I),Z(I),IFIX(I),ISYM(I),(JT(I,J),J=1,4),I=1,
     1 NN)
      WRITE(6,3)(X(I),Y(I),Z(I),IFIX(I),ISYM(I),(JT(I,J),J=1,4),I=1,
     1 NN)
    3 FORMAT(3F10.3,2I2,4I5)
c     Reflect node point about coord axes
c                     isym=1  points on the x axis
c                     isym=2  points on the y axis
c                     isym=3  center of symmetry
      NB0=NB
```

```
      NN0=NN
      DO 31 I=1,NN0
      NSTART(I)=NN+1
      IF(ISYM(I).EQ.1) GO TO 22
      IF(ISYM(I).EQ.3) GO TO 31
      NN=NN+1
      ifix(nn)=ifix(i)
      X(NN)=X(I)
      Y(NN)=-Y(I)
      Z(NN)=Z(I)
      IF(ISYM(I).EQ.2) GO TO 31
      NN=NN+1
      ifix(nn)=ifix(i)
      X(NN)=-X(I)
      Y(NN)=-Y(I)
      Z(NN)=Z(I)
   22 NN=NN+1
      ifix(nn)=ifix(i)
      X(NN)=-X(I)
      Y(NN)=Y(I)
      Z(NN)=Z(I)
   31 CONTINUE
      DO 61 I=1,NB0
      IP=NP(I)
      IM=MI(I)
      IPS=NSTART(IP)-1
      IMS=NSTART(IM)-1
      ISUM=ISYM(IP)+ISYM(IM)
      if(isum.eq.0) go to 62
      isngl=0
      if(isum.ge.4.or.isym(ip).eq.isym(im)) isngl=1
      IF(isum.ge.4.OR.ISYM(IP).EQ.ISYM(IM)) GO TO 63
   62 NB=NB+1
      IPS=IPS+1
      IMS=IMS+1
      NP(NB)=IPS
      MI(NB)=IMS
      FORH(NB)=FORH(I)
      IF(isym(ip).eq.1.or.isym(ip).eq.3) GO TO 65
      IF(isym(im).eq.1.or.isym(im).eq.3) GO TO 66
      GO TO 64
   65 NP(NB)=IP
      IPS=IPS-1
      GO TO 64
   66 MI(NB)=IM
      IMS=IMS-1
   64 NB=NB+1
      IPS=IPS+1
      IMS=IMS+1
      NP(NB)=IPS
      MI(NB)=IMS
      FORH(NB)=FORH(I)
      IF(isym(ip).eq.2) GO TO 165
      IF(isym(im).eq.2) GO TO 166
      if(isym(iP).eq.3) go to 465
      if(isym(im).eq.3) go to 466
      GO TO 63
```

```
  465 np(nb)=ip
      ips=ips-1
      go to 63
  466 mi(nb)=im
      ims=ims-1
      go to 63
  165 ips=ips-1
      NP(NB)=IPs
      GO TO 63
  166 ims=ims-1
      MI(NB)=IMs
  63  NB=NB+1
  100 IPS=IPS+1
      IMS=IMS+1
      NP(NB)=IPS
      MI(NB)=IMS
      FORH(NB)=FORH(I)
      if(isngl.eq.1.and.isym(ip).eq.3) go to 265
      if(isngl.eq.1.and.isym(im).eq.3) go to 266
      if(isngl.eq.1)go to 61
      IF(isym(ip).eq.2.or.isym(ip).eq.3) GO TO 265
      IF(isym(im).eq.2.or.isym(im).eq.3) GO TO 266
      IF(isym(ip).eq.1.or.isym(ip).eq.3) GO TO 365
      IF(isym(im).eq.1.or.isym(im).eq.3) GO TO 366
      GO TO61
  265 NP(NB)=IP
      GO TO 61
  266 MI(NB)=IM
      go to 61
  365 ips=ips-1
      np(nb)=ips
      go to 61
  366 ims=ims-1
      mi(nb)=ims
   61 CONTINUE
      DO 71 I=1,NN
      R(3*I-2)=X(I)
      R(3*I-1)=Y(I)
      R(3*I)  =Z(I)
   71 WRITE(6,72)I, R(3*I-2),R(3*I-1),R(3*I)
      WRITE(150,1)NB,NN
      WRITE(150,222)(NP(I),MI(I),forh(i),I=1,NB)
      WRITE(150,3)(X(I),Y(I),Z(I),IFIX(I),ISYM(I),(JT(I,J),J=1,4),I=1,
     1 NN)
   72 FORMAT(I5,3E20.8)
      CALL SPLOT(NP,MI,NN,NB,R,forh,0)
      STOP
      END
      SUBROUTINE PLOT(NB, NN, X, Y, NP, MI,for,iwrite)
      INCLUDE 'FGRAPH.FD'
      DIMENSION  NP(1), MI(1), X(1), Y(1),for(1)
      INTEGER*2 DUMMY,xk,yk,xm,ym,lx,ly
      RECORD /XYCOORD/ XY
C     record /rccoord/ curpos
      character*6  text
      character*15 text1
      CHARACTER*64 FONTPATH
```

```
      CHARACTER*20 LIST
      FONTPATH='\f32\lib\courb.fon'
      LIST="t'courb'"//'h10w10b'
      DUMMY = SETVIDEOMODE( $VRES16COLOR)
      DUMMY=REGISTERFONTS(FONTPATH)
      DUMMY=SETFONT(LIST)
      AMAXX=639-20
      AMAYY=479-20
      XMIN=X(1)
      XMAX=X(1)
      YMIN=Y(1)
      YMAX=Y(1)
      DO 2 I=1,NN
      XI=X(I)
      YI=Y(I)
      IF(XMIN.GT.XI) XMIN=XI
      IF(XMAX.LT.XI) XMAX=XI
      IF(YMIN.GT.YI) YMIN=YI
 2    IF(YMAX.LT.YI) YMAX=YI
      SCALE = AMAX1((XMAX-XMIN)/AMAXX,(YMAX-YMIN)/AMAYY)
      XSHIFT = (XMAX+XMIN)/2.0 - 639/2*SCALE
      YSHIFT = (YMAX+YMIN)/2.0 - 479/2*SCALE
      DO 3 I=1,NB
      K=NP(I)
      M=MI(I)
      XK=(X(K)-XSHIFT)/SCALE
      YK=(Y(K)-YSHIFT)/SCALE
      XM=(X(M)-XSHIFT)/SCALE
      YM=(Y(M)-YSHIFT)/SCALE
      YK = 479-YK
      YM = 479-YM
      LX=((XK+XM)/2)
      LY=((YK+YM)/2)
      CALL MOVETO  ( XK, YK,  XY)
      DUMMY = LINETO  ( XM, YM)
      if(iwrite.ne.2) go to 998
      call  moveto(lx,ly,xy)
      write(text, '(i3)') i
      call outgtext (text)
998   if(iwrite.eq.0.or.iwrite.eq.2) go to 3
      call moveto(lx,ly,xy)
      write(text1,'(f7.0)') for(i)
      call outgtext (text1)
 3    CONTINUE
      if(iwrite.ne.2) go to 996
      do 997 i=1,nn
      lx=(x(i)-xshift)/scale
      yk=(y(i)-yshift)/scale
      ly=(479-yk)
      call  moveto(lx,ly,xy)
      write(text, '(i3)') i
      call outgtext (text)
997   continue
996   continue
      RETURN
      END
```

```
    SUBROUTINE SPLOT ( NP,NM,NN,NB,R,for,iwrite)
    INCLUDE 'FGRAPH.FD'
    DIMENSION  NP(1),NM(1),RXY(1000),ROT(3,3),for(1)
    DIMENSION ANGL(3),NT(3),A(3,3),R1(3,3,3)
    INTEGER*2 DUMMY
    DIMENSION R(1),X(400),Y(400),RZ(1000)
    WRITE(*,1)
  1 FORMAT(' YOU ARE ABOUT TO ENTER A GRAPHICS '
   1 'DISPLAY MODE'/' THE KEYBOARD COMMANDS ARE'//
   1 '     +1...POSITIVE ROTATION ABOUT X AXIS'/
   1 '     -1...NEGATIVE ROTATION ABOUT X AXIS'/
   1 '     +2...POSITIVE ROTATION ABOUT Y AXIS'/
   1 '     -2...NEGATIVE ROTATION ABOUT Y AXIS'/
   1 '     +3...POSITIVE ROTATION ABOUT Z AXIS'/
   1 '     -3...NEGATIVE ROTATION ABOUT Z AXIS'/
   1 '      0...EXIT')
    READ(*,*)
    DO 616 I=1,3
    DO 617 J=1,3
    DO 617 K=1,3
617 R1(I,J,K)=0.
616 R1(I,I,I)=1.
    THX=0.
    THY=00.
    THZ=00.
    DTH=10.
70  PI=3.14159
    DO 604 I=1,3
    DO 603 J=1,3
603 ROT(J,I)=0.
604 ROT(I,I)=1.
    ANGL(1)=THX
    ANGL(2)=THY
    ANGL(3)=THZ
    NT(1)=1
      NT(2)=2
    NT(3)=3
    I=0
302 I=I+1
    IF(ANGL(I))606,605,606
606 L=NT(I)
    GO TO 612
618 DO 607 J=1,3
    DO 607 JA=1,3
    A(J,JA)=0.
    DO 607 JB=1,3
607 A(J,JA)=A(J,JA)+R1(L,J,JB)*ROT(JB,JA)
    DO 608 K=1,3
    DO 608 J=1,3
608 ROT(K,J)=A(K,J)
605 IF(I-3) 302,303,303
303 DO 805 I=1,NN
    RZ(I)=0.
    DO 806 K=1,3
806 RZ(I)=RZ(I)+ROT(3,K)*R(3*I-3+K)
    DO 805 J=1,2
    RXY(2*I-2+J)=0.
```

```
      DO 805 K=1,3
  805 RXY(2*I-2+J)=RXY(2*I-2+J)+ROT(J,K)*R(3*I-3+K)
      GO TO 59
  612 ANG=ANGL(I)*PI/180.
      IF(L-2)613,614,615
  613 R1(1,2,2)=COS(ANG)
      R1(1,2,3)=SIN(ANG)
      R1(1,3,3)=R1(1,2,2)
      R1(1,3,2)=-R1(1,2,3)
      GO TO 618
  614 R1(2,1,1)=COS(ANG)
      R1(2,1,3)=-SIN(ANG)
      R1(2,3,1)=-R1(2,1,3)
      R1(2,3,3)=R1(2,1,1)
      GO TO 618
  615 R1(3,1,1)=COS(ANG)
      R1(3,1,2)=SIN(ANG)
      R1(3,2,1)=-R1(3,1,2)
      R1(3,2,2)=R1(3,1,1)
      GO TO 618
   59 DO 24 I=1,NN
      X(I)=RXY(2*I-1)
   24 Y(I)=RXY(2*I)
      CALL PLOT(NB,NN,X,Y,NP,NM,for,iwrite)
      READ(*,*) IVAL
      IF(IVAL.EQ.+1) GO TO 2000
      IF(IVAL.EQ.-1) GO TO 3000
      IF(IVAL.EQ. 2) GO TO 4000
      IF(IVAL.EQ.-2) GO TO 5000
      IF(IVAL.EQ. 3) GO TO 6000
      IF(IVAL.EQ.-3) GO TO 7000
      IF(IVAL.EQ. 0) GO TO 8000
 2000 THX=THX+DTH
      GO TO 70
 3000 THX=THX-DTH
      GO TO 70
 4000 THY=THY+DTH
      GO TO 70
 5000 THY=THY-DTH
      GO TO 70
 6000 THZ=THZ+DTH
      GO TO 70
 7000 THZ=THZ-DTH
      GO TO 70
 8000 CALL UNREGISTERFONTS()
      DUMMY = SETVIDEOMODE( $DEFAULTMODE )
      RETURN
      END
```