Table 10 (p. 6 of 13)

```
C=DECK       BEHX1
      SUBROUTINE BEHX1
     1 (IFILE,NPRINX,IMODX,IFAST,ILOADX,PHRASE)
C
C    PURPOSE: OBTAIN general buckling load
C
C    YOU MUST WRITE CODE THAT, USING
C    THE VARIABLES IN THE LABELLED
C    COMMON BLOCKS AS INPUT, ULTIMATELY
C    YIELDS THE RESPONSE VARIABLE FOR
C    THE ith LOAD CASE, ILOADX:
C
C      GENBUK(ILOADX)
C
C    AS OUTPUT. THE ith CASE REFERS
C    TO ith ENVIRONMENT (e.g. load com-
C    bination).
C
C    DEFINITIONS OF INPUT DATA:
C     IMODX  = DESIGN CONTROL INTEGER:
C      IMODX = 0 MEANS BASELINE DESIGN
C      IMODX = 1 MEANS PERTURBED DESIGN
C      IFAST = 0 MEANS FEW  SHORTCUTS FOR PERTURBED DESIGNS
C      IFAST = 1 MEANS MORE SHORTCUTS FOR PERTURBED DESIGNS
C     IFILE = FILE FOR OUTPUT LIST:
C     NPRINX= OUTPUT CONTROL INTEGER:
C      NPRINX=0 MEANS SMALLEST AMOUNT
C      NPRINX=1 MEANS MEDIUM AMOUNT
C      NPRINX=2 MEANS LOTS OF OUTPUT
C
C      ILOADX = ith LOADING COMBINATION
C      PHRASE = general buckling load
C
C    OUTPUT:
C
C      GENBUK(ILOADX)
C
       CHARACTER*80 PHRASE
C   INSERT ADDITIONAL COMMON BLOCKS:
      COMMON/FV01/WLAND,TLAND,ECLAND,TWLAND,HWLAND,TFLAND,WFLAND,TSKIN
      REAL WLAND,TLAND,ECLAND,TWLAND,HWLAND,TFLAND,WFLAND,TSKIN
      COMMON/FV09/RADCYL,LENCYL,BSTR,TSTR,HSTR,BRNG,TRNG,HRNG,EMOD
      REAL RADCYL,LENCYL,BSTR,TSTR,HSTR,BRNG,TRNG,HRNG,EMOD
      COMMON/FV20/PX(20)
      REAL PX
      COMMON/FV26/GENBUK(20),GENBUKA(20),GENBUKF(20)
      REAL GENBUK,GENBUKA,GENBUKF
      COMMON/FV29/PANBUK(20),PANBUKA(20),PANBUKF(20)
      REAL PANBUK,PANBUKA,PANBUKF
      COMMON/FV32/STRESS(20),STRESSA(20),STRESSF(20)
      REAL STRESS,STRESSA,STRESSF
      COMMON/IV01/KLAND,NLAND,MLOWG,MHIGHG,MLOWP,MHIGHP
      INTEGER KLAND,NLAND,MLOWG,MHIGHG,MLOWP,MHIGHP
      COMMON/FV18/NU,DENS,WEIGHT
      REAL NU,DENS,WEIGHT
      COMMON/FV21/PY(20),PX0(20),PY0(20)
      REAL PY,PX0,PY0
C
C
C  INSERT SUBROUTINE STATEMENTS HERE.
C
C
      COMMON/INSTAB/INDIC
      COMMON/EIGB4M/EIGCOM(200),EIGNEG(200),EIGCRN
      COMMON/WVEB4M/NWVCOM(200),NWVNEG(200),IWAVEB,NWVCRN
      COMMON/EIGBUK/EIGCRT
      COMMON/NWVBUK/NWVCRT
      COMMON/BUCKN/N0BX,NMINBX,NMAXBX,INCRBX
      COMMON/BUCKN0/N0B,NMAXB
      COMMON/RBEGX/RBEG
      COMMON/NXPART/NXLAND,NXWEB,NXFLNG,NXSKIN,NXSMER
      COMMON/PRMOUT/IFILE3,IFILE4,IFILE8,IFILE9,IFIL11
      COMMON/EIGALL/EIG0,EIG1,EIG2,EIG3,EIG4
      COMMON/WAVALL/NWAV0,NWAV1,NWAV2,NWAV3,NWAV4
      COMMON/NUMPAR/IPARX,IVARX,IALLOW,ICONSX,NDECX,NLINKX,NESCAP,ITYPEX
      common/caseblock/CASE
      CHARACTER*28 CASE
      CHARACTER*35 CASA
```

GENOPT created

GENOPT user insertions.

Table 10 (p. 7 of 10)

```
      REAL LENMOD
      REAL NXLAND,NXWEB,NXFLNG,NOTLND,NXSKIN,NXSMER
C
      PI = 3.1415927
      FLAND = FLOAT(NLAND)
      NOTLND = 2.*PI*RADCYL - FLAND*WLAND
      DENOM = NOTLND + NOTLND*HSTR*TSTR/(TSKIN*BSTR)
     1       +2.*FLAND*HWLAND*TWLAND/TSKIN
     1       +2.*FLAND*WFLAND*TFLAND/TSKIN
     1          +FLAND*TLAND/TSKIN
      NXSKIN = PX(ILOADX)/DENOM
      NXLAND = NXSKIN*TLAND/TSKIN
      NXWEB  = NXSKIN*TWLAND/TSKIN
      NXFLNG = NXSKIN*TFLAND/TSKIN
      NXSMER = NXSKIN*(1. + HSTR*TSTR/(TSKIN*BSTR))
C
      INDIC = 4
      LENMOD = 1.0
      RAVE = 100.*LENCYL*LENMOD/PI
      RBEG = RAVE -0.707*RADCYL
      N0B = MLOWG*100
      NMAXB = MHIGHG*100
C
C  Get model without any weld lands:
      IF (IMODX.EQ.0) THEN
C        CALL BOSDEC(0,24,ILOADX,INDIC)
C
         IF (ITYPEX.EQ.2) THEN
C          Get CASE.BEHX0 file for input for BIGBOSOR4...
C          CASE.BEHX0 is an input file for BIGBOSOR4 for behavior no. 0:
C          general buckling load without any weld lands
           I=INDEX(CASE,' ')
           IF(I.NE.0) THEN
              CASA=CASE(:I-1)//'.BEHX0'
           ELSE
              CASA=CASE//'.BEHX0'
           ENDIF
           OPEN(UNIT=61,FILE=CASA,STATUS='UNKNOWN')
           CALL BOSDEC(0,61,ILOADX,INDIC)
           CLOSE(UNIT=61)
           WRITE(IFILE,'(/,/,A,A,/,A)')
     1    ' BIGBOSOR4 input file for:',
     1    ' general buckling load, no weld lands',
     1      CASA
         ENDIF
      ENDIF
C
      CALL BOSDEC(1,24,ILOADX,INDIC)
C
      IF (ITYPEX.EQ.2) THEN
C       Get CASE.BEHX1 file for input for BIGBOSOR4...
C       CASE.BEHX1 is an input file for BIGBOSOR4 for behavior no. 1:
C       general buckling load
        I=INDEX(CASE,' ')
        IF(I.NE.0) THEN
           CASA=CASE(:I-1)//'.BEHX1'
        ELSE
           CASA=CASE//'.BEHX1'
        ENDIF
        OPEN(UNIT=61,FILE=CASA,STATUS='UNKNOWN')
        CALL BOSDEC(1,61,ILOADX,INDIC)
        CLOSE(UNIT=61)
        WRITE(IFILE,'(/,/,A,A,/,A)')
     1  ' BIGBOSOR4 input file for:',
     1  ' general buckling load',
     1    CASA
      ENDIF
C
      CALL B4READ
      IF (IMODX.EQ.0) THEN
         N0BX = N0B
         NMINBX = N0B
         NMAXBX = NMAXB
         INCRBX = 100
      ELSE
         NOBX = NWAV1
         NMINBX = NWAV1
         NMAXBX = NWAV1
```

GENOPT user insertion

Table 10 (p. 8 of 13)

```
         INCRBX = 100
       ENDIF
       REWIND IFILE9
       CALL STOCM1(IFILE9)
       CALL STOCM2(IFILE9)
       CALL B4MAIN
       CALL GASP(DUM1,DUM2,-2,DUM3)
       IF (IMODX.EQ.0) THEN
          EIG1 = EIGCRT
          NWAV1= NWVCRT
       ENDIF
C
       WRITE(IFILE,'(/,A)')
      1 ' GENERAL BUCKLING LOAD FACTORS AND MODES (BEHX1)'
       DO 10 I = 1,IWAVEB
          WRITE(IFILE,'(A,1P,E12.4,A,I4,A)')
      1   '      ',EIGCOM(I),'(',NWVCOM(I),')'
   10 CONTINUE
       WRITE(IFILE,'(A,1P,E12.4)')
      1' Critical buckling load factor, GENBUK=',EIGCRT
       WRITE(IFILE,'(A,I5)')
      1' Critical number of circumferential waves, NWVCRT=',NWVCRT
       GENBUK(ILOADX) = EIGCRT
C
       RETURN
       END
C
C
C
C
C=DECK       BEHX2
       SUBROUTINE BEHX2
      1 (IFILE,NPRINX,IMODX,IFAST,ILOADX,PHRASE)
C
C    PURPOSE: OBTAIN "panel" buckling
C
C    YOU MUST WRITE CODE THAT, USING
C    THE VARIABLES IN THE LABELLED
C    COMMON BLOCKS AS INPUT, ULTIMATELY
C    YIELDS THE RESPONSE VARIABLE FOR
C    THE ith LOAD CASE, ILOADX:
C
C      PANBUK(ILOADX)
C
C    AS OUTPUT. THE ith CASE REFERS
C    TO ith ENVIRONMENT (e.g. load com-
C    bination).
C
C    DEFINITIONS OF INPUT DATA:
C     IMODX  = DESIGN CONTROL INTEGER:
C      IMODX = 0 MEANS BASELINE DESIGN
C      IMODX = 1 MEANS PERTURBED DESIGN
C      IFAST = 0 MEANS FEW  SHORTCUTS FOR PERTURBED DESIGNS
C      IFAST = 1 MEANS MORE SHORTCUTS FOR PERTURBED DESIGNS
C     IFILE = FILE FOR OUTPUT LIST:
C     NPRINX= OUTPUT CONTROL INTEGER:
C      NPRINX=0 MEANS SMALLEST AMOUNT
C      NPRINX=1 MEANS MEDIUM AMOUNT
C      NPRINX=2 MEANS LOTS OF OUTPUT
C
C      ILOADX = ith LOADING COMBINATION
C      PHRASE = "panel" buckling
C
C    OUTPUT:
C
C      PANBUK(ILOADX)
C
       CHARACTER*80 PHRASE
C   INSERT ADDITIONAL COMMON BLOCKS:
       COMMON/FV01/WLAND,TLAND,ECLAND,TWLAND,HWLAND,TFLAND,WFLAND,TSKIN
       REAL WLAND,TLAND,ECLAND,TWLAND,HWLAND,TFLAND,WFLAND,TSKIN
       COMMON/FV09/RADCYL,LENCYL,BSTR,TSTR,HSTR,BRNG,TRNG,HRNG,EMOD
       REAL RADCYL,LENCYL,BSTR,TSTR,HSTR,BRNG,TRNG,HRNG,EMOD
       COMMON/FV20/PX(20)
       REAL PX
       COMMON/FV26/GENBUK(20),GENBUKA(20),GENBUKF(20)
       REAL GENBUK,GENBUKA,GENBUKF
       COMMON/FV29/PANBUK(20),PANBUKA(20),PANBUKF(20)
```

GENOPT user-insertion

GENOPT created

(inter-ring buckling)

GENOPT created

Table 10 (p. 9 of 13)

```
      REAL PANBUK,PANBUKA,PANBUKF
      COMMON/FV32/STRESS(20),STRESSA(20),STRESSF(20)
      REAL STRESS,STRESSA,STRESSF
      COMMON/IV01/KLAND,NLAND,MLOWG,MHIGHG,MLOWP,MHIGHP
      INTEGER KLAND,NLAND,MLOWG,MHIGHG,MLOWP,MHIGHP
      COMMON/FV18/NU,DENS,WEIGHT
      REAL NU,DENS,WEIGHT
      COMMON/FV21/PY(20),PX0(20),PY0(20)
      REAL PY,PX0,PY0
C
C
C  INSERT SUBROUTINE STATEMENTS HERE.
C
C
      COMMON/INSTAB/INDIC
      COMMON/EIGB4M/EIGCOM(200),EIGNEG(200),EIGCRN
      COMMON/WVEB4M/NWVCOM(200),NWVNEG(200),IWAVEB,NWVCRN
      COMMON/EIGBUK/EIGCRT
      COMMON/NWVBUK/NWVCRT
      COMMON/BUCKN/N0BX,NMINBX,NMAXBX,INCRBX
      COMMON/BUCKN0/N0B,NMAXB
      COMMON/RBEGX/RBEG
      COMMON/PRMOUT/IFILE3,IFILE4,IFILE8,IFILE9,IFIL11
      COMMON/EIGALL/EIG0,EIG1,EIG2,EIG3,EIG4
      COMMON/WAVALL/NWAV0,NWAV1,NWAV2,NWAV3,NWAV4
      COMMON/NUMPAR/IPARX,IVARX,IALLOW,ICONSX,NDECX,NLINKX,NESCAP,ITYPEX
      common/caseblock/CASE
      CHARACTER*28 CASE
      CHARACTER*35 CASA
      REAL NXLAND,NXWEB,NXFLNG,NOTLND,NXSKIN,NXSMER
C
      PI = 3.1415927
      FLAND = FLOAT(NLAND)
      NOTLND = 2.*PI*RADCYL - FLAND*WLAND
      DENOM = NOTLND + NOTLND*HSTR*TSTR/(TSKIN*BSTR)
     1       +2.*FLAND*HWLAND*TWLAND/TSKIN
     1       +2.*FLAND*WFLAND*TFLAND/TSKIN
     1          +FLAND*TLAND/TSKIN
      NXSKIN = PX(ILOADX)/DENOM
      NXLAND = NXSKIN*TLAND/TSKIN
      NXWEB  = NXSKIN*TWLAND/TSKIN
      NXFLNG = NXSKIN*TFLAND/TSKIN
      NXSMER = NXSKIN*(1. + HSTR*TSTR/(TSKIN*BSTR))
C
      INDIC = 4
      RAVE = 100.*BRNG/PI
      RBEG = RAVE -0.707*RADCYL
      N0B = MLOWP*100
      NMAXB = MHIGHP*100
C
      CALL BOSDEC(2,24,ILOADX,INDIC)
C
      IF (ITYPEX.EQ.2) THEN
C      Get CASE.BEHX2 file for input for BIGBOSOR4...
C      CASE.BEHX2 is an input file for BIGBOSOR4 for behavior no. 2:
C      "panel" buckling load
         I=INDEX(CASE,' ')
         IF(I.NE.0) THEN
            CASA=CASE(:I-1)//'.BEHX2'
         ELSE
            CASA=CASE//'.BEHX2'
         ENDIF
         OPEN(UNIT=61,FILE=CASA,STATUS='UNKNOWN')
         CALL BOSDEC(2,61,ILOADX,INDIC)
         CLOSE(UNIT=61)
         WRITE(IFILE,'(/,/,A,A,/,A)')
     1 ' BIGBOSOR4 input file for:',
     1 ' "panel" buckling load',
     1   CASA
      ENDIF
C
      CALL B4READ
      IF (IMODX.EQ.0) THEN
         N0BX = N0B
         NMINBX = N0B
         NMAXBX = NMAXB
         INCRBX = 100
      ELSE
```

GENOPT created

GENOPT user insertion

Table 10 (p. 10 of 13)

```
         NOBX = NWAV2
         NMINBX = NWAV2
         NMAXBX = NWAV2
         INCRBX = 100
      ENDIF
      REWIND IFILE9
      CALL STOCM1(IFILE9)
      CALL STOCM2(IFILE9)
      CALL B4MAIN
      CALL GASP(DUM1,DUM2,-2,DUM3)
      IF (IMODX.EQ.0) THEN
         EIG2 = EIGCRT
         NWAV2= NWVCRT
      ENDIF
C
      WRITE(IFILE,'(/,A)')
     1 ' "PANEL" BUCKLING LOAD FACTORS AND MODES (BEHX2)'
      DO 10 I = 1,IWAVEB
         WRITE(IFILE,'(A,1P,E12.4,A,I4,A)')
     1  '      ',EIGCOM(I),'(',NWVCOM(I),')'
   10 CONTINUE
      WRITE(IFILE,'(A,1P,E12.4)')
     1' Critical buckling load factor, PANBUK=',EIGCRT
      WRITE(IFILE,'(A,I5)')
     1' Critical number of circumferential waves, NWVCRT=',NWVCRT
      PANBUK(ILOADX) = EIGCRT
C
      RETURN
      END
C
C
C
C
C=DECK       BEHX3
      SUBROUTINE BEHX3
     1 (IFILE,NPRINX,IMODX,IFAST,ILOADX,PHRASE)
C
C    PURPOSE: OBTAIN weld land effective stress
C
C    YOU MUST WRITE CODE THAT, USING
C    THE VARIABLES IN THE LABELLED
C    COMMON BLOCKS AS INPUT, ULTIMATELY
C    YIELDS THE RESPONSE VARIABLE FOR
C    THE ith LOAD CASE, ILOADX:
C
C      STRESS(ILOADX)
C
C    AS OUTPUT. THE ith CASE REFERS
C    TO ith ENVIRONMENT (e.g. load com-
C    bination).
C
C    DEFINITIONS OF INPUT DATA:
C     IMODX  = DESIGN CONTROL INTEGER:
C      IMODX = 0 MEANS BASELINE DESIGN
C      IMODX = 1 MEANS PERTURBED DESIGN
C      IFAST = 0 MEANS FEW  SHORTCUTS FOR PERTURBED DESIGNS
C      IFAST = 1 MEANS MORE SHORTCUTS FOR PERTURBED DESIGNS
C     IFILE = FILE FOR OUTPUT LIST:
C     NPRINX= OUTPUT CONTROL INTEGER:
C      NPRINX=0 MEANS SMALLEST AMOUNT
C      NPRINX=1 MEANS MEDIUM AMOUNT
C      NPRINX=2 MEANS LOTS OF OUTPUT
C
C      ILOADX = ith LOADING COMBINATION
C      PHRASE = weld land effective stress
C
C    OUTPUT:
C
C      STRESS(ILOADX)
C
       CHARACTER*80 PHRASE
C   INSERT ADDITIONAL COMMON BLOCKS:
      COMMON/FV01/WLAND,TLAND,ECLAND,TWLAND,HWLAND,TFLAND,WFLAND,TSKIN
      REAL WLAND,TLAND,ECLAND,TWLAND,HWLAND,TFLAND,WFLAND,TSKIN
      COMMON/FV09/RADCYL,LENCYL,BSTR,TSTR,HSTR,BRNG,TRNG,HRNG,EMOD
      REAL RADCYL,LENCYL,BSTR,TSTR,HSTR,BRNG,TRNG,HRNG,EMOD
      COMMON/FV20/PX(20)
      REAL PX
```

GENOPT user information

GENOPT created

Table 10 (p. 11 of 13)

```
      COMMON/FV26/GENBUK(20),GENBUKA(20),GENBUKF(20)
      REAL GENBUK,GENBUKA,GENBUKF
      COMMON/FV29/PANBUK(20),PANBUKA(20),PANBUKF(20)
      REAL PANBUK,PANBUKA,PANBUKF
      COMMON/FV32/STRESS(20),STRESSA(20),STRESSF(20)
      REAL STRESS,STRESSA,STRESSF
      COMMON/IV01/KLAND,NLAND,MLOWG,MHIGHG,MLOWP,MHIGHP
      INTEGER KLAND,NLAND,MLOWG,MHIGHG,MLOWP,MHIGHP
      COMMON/FV18/NU,DENS,WEIGHT
      REAL NU,DENS,WEIGHT
      COMMON/FV21/PY(20),PX0(20),PY0(20)
      REAL PY,PX0,PY0
C
C
C  INSERT SUBROUTINE STATEMENTS HERE.
C
      REAL NXLAND,NXWEB,NXFLNG,NOTLND,NXSKIN,NXSMER
C
      PI = 3.1415927
      FLAND = FLOAT(NLAND)
      NOTLND = 2.*PI*RADCYL - FLAND*WLAND
      DENOM = NOTLND + NOTLND*HSTR*TSTR/(TSKIN*BSTR)
     1        +2.*FLAND*HWLAND*TWLAND/TSKIN
     1        +2.*FLAND*WFLAND*TFLAND/TSKIN
     1           +FLAND*TLAND/TSKIN
      NXSKIN = PX(ILOADX)/DENOM
      NXLAND = NXSKIN*TLAND/TSKIN
      NXWEB  = NXSKIN*TWLAND/TSKIN
      NXFLNG = NXSKIN*TFLAND/TSKIN
      NXSMER = NXSKIN*(1. + HSTR*TSTR/(TSKIN*BSTR))
C
      STRLND = NXLAND/TLAND
      STRWEB = NXWEB/TWLAND
      STRFLG = NXFLNG/TFLAND
C
      STRESS(ILOADX) = MAX(STRLND,STRWEB)
      STRESS(ILOADX) = MAX(STRESS(ILOADX),STRFLG)
C
      RETURN
      END
C
C
C
C
C=DECK      USRCON
      SUBROUTINE USRCON(INUMTT,IMODX,CONMAX,ICONSX,IPOINC,CONSTX,
     1  WORDCX,WORDMX,PCWORD,CPLOTX,ICARX,IFILEX)
C    PURPOSE: GENERATE USER-WRITTEN
C    INEQUALITY CONSTRAINT CONDITION
C    USING ANY COMBINATION OF PROGRAM
C    VARIABLES.
C    YOU MUST WRITE CODE THAT, USING
C    THE VARIABLES IN THE LABELLED
C    COMMON BLOCKS AS INPUT, ULTIMATELY
C    YIELDS A CONSTRAINT CONDITION,
C    CALLED "CONX" IN THIS ROUTINE.
      DIMENSION WORDCX(*),WORDMX(*),IPOINC(*),CONSTX(*)
      DIMENSION PCWORD(*),CPLOTX(*)
      CHARACTER*80 WORDCX,WORDMX,PCWORD
C  INSERT ADDITIONAL COMMON BLOCKS:
      COMMON/FV01/WLAND,TLAND,ECLAND,TWLAND,HWLAND,TFLAND,WFLAND,TSKIN
      REAL WLAND,TLAND,ECLAND,TWLAND,HWLAND,TFLAND,WFLAND,TSKIN
      COMMON/FV09/RADCYL,LENCYL,BSTR,TSTR,HSTR,BRNG,TRNG,HRNG,EMOD
      REAL RADCYL,LENCYL,BSTR,TSTR,HSTR,BRNG,TRNG,HRNG,EMOD
      COMMON/FV20/PX(20)
      REAL PX
      COMMON/FV26/GENBUK(20),GENBUKA(20),GENBUKF(20)
      REAL GENBUK,GENBUKA,GENBUKF
      COMMON/FV29/PANBUK(20),PANBUKA(20),PANBUKF(20)
      REAL PANBUK,PANBUKA,PANBUKF
      COMMON/FV32/STRESS(20),STRESSA(20),STRESSF(20)
      REAL STRESS,STRESSA,STRESSF
      COMMON/IV01/KLAND,NLAND,MLOWG,MHIGHG,MLOWP,MHIGHP
      INTEGER KLAND,NLAND,MLOWG,MHIGHG,MLOWP,MHIGHP
      COMMON/FV18/NU,DENS,WEIGHT
      REAL NU,DENS,WEIGHT
      COMMON/FV21/PY(20),PX0(20),PY0(20)
      REAL PY,PX0,PY0
```

GENOPT created

GENOPT user insertion

GENOPT created

Table 10 (p. 12 of 13)

```
C
      CONX = 0.0
C
C  INSERT USER-WRITTEN STATEMENTS
C  HERE. THE CONSTRAINT CONDITION
C  THAT YOU CALCULATE IS CALLED "CONX"
C
      IF (CONX.EQ.0.0) RETURN
      IF (CONX.LT.0.0) THEN
         WRITE(IFILEX,*)' CONX MUST BE GREATER THAN ZERO.'
         CALL EXIT
      ENDIF
C
C  DO NOT CHANGE THE FOLLOWING STATEMENTS, EXCEPT WORDC
C
      ICARX = ICARX + 1
      INUMTT = INUMTT + 1
      WORDCX(ICARX) = ' USER: PROVIDE THIS.'
      CPLOTX(ICARX) = CONX - 1.
      CALL BLANKX(WORDCX(ICARX),IENDP)
      PCWORD(ICARX)  = WORDCX(ICARX)(1:IENDP)//' -1'
      IF (IMODX.EQ.0.AND.CONX.GT.CONMAX) GO TO 200
      IF (IMODX.EQ.1.AND.IPOINC(INUMTT).EQ.0) GO TO 200
      ICONSX = ICONSX + 1
      IF (IMODX.EQ.0) IPOINC(INUMTT) = 1
      CONSTX(ICONSX) = CONX
      WORDMX(ICONSX)  = WORDCX(ICARX)(1:IENDP)//' -1'
  200 CONTINUE
C  END OF USRCON
C
C
      RETURN
      END
C
C
C
C=DECK       USRLNK
      SUBROUTINE USRLNK(VARI,I,VARIAB)
C Purpose: generate user-written
C linking conditions using any
C combination of decision variables.
C You must write conde that, using
C the variables in the subroutine
C argument VARIAB as input, ultimately
C yield a value for the linked variable
C VARI.
C
C VARI is the Ith entry of the array
C VARIAB.  You have decided that this
C is to be a linked variable with user
C defined linking.  It is linked to
C the decision variables in the array
C VARIAB.
C An example will provide the simplest
C explanation of this:
C Let"s say that the 5th decision
C variable candidate (I=5) is linked
C to the decision variable candidates
C 2 and 7. (You used DECIDE to select
C these as decision variables.
C In this case VARI is equal to
C VARIAB(I).  You then write your
C linking equation in the form
C VARI=f(VARIAB(2),VARIAB(7)).
C Use the index I in an IF statement if
C you have more than one user-defined
C linked variable.
C
C
      REAL VARI,VARIAB(50)
      INTEGER I
C
C  INSERT USER-WRITTEN DECLARATION
C  STATEMENTS HERE.
C
C  INSERT USER-WRITTEN
C  STATEMENTS HERE.
C
```

Created by GENOPT

Table 10 (p. 13 of 13)

```
C
C  END OF USRLNK
      RETURN
      END
C=DECK       OBJECT
      SUBROUTINE OBJECT(IFILE,NPRINX,IMODX,OBJGEN,PHRASE)
C   PURPOSE:weight of the weld land+"extra" edge stringers
C
C   YOU MUST WRITE CODE THAT, USING
C   THE VARIABLES IN THE LABELLED
C   COMMON BLOCKS AS INPUT, ULTIMATELY
C   YIELDS THE OBJECTIVE FUNCTION
C         WEIGHT
C   AS OUTPUT. MAKE SURE TO INCLUDE AT
C   THE END OF THE SUBROUTINE, THE
C   STATEMENT: OBJGEN = WEIGHT
C
C
C   DEFINITIONS OF INPUT DATA:
C    IMODX  = DESIGN CONTROL INTEGER:
C     IMODX = 0 MEANS BASELINE DESIGN
C     IMODX = 1 MEANS PERTURBED DESIGN
C     IFAST = 0 MEANS FEW  SHORTCUTS FOR PERTURBED DESIGNS
C     IFAST = 1 MEANS MORE SHORTCUTS FOR PERTURBED DESIGNS
C    IFILE = FILE FOR OUTPUT LIST:
C    NPRINX= OUTPUT CONTROL INTEGER:
C     NPRINX=0 MEANS SMALLEST AMOUNT
C     NPRINX=1 MEANS MEDIUM AMOUNT
C     NPRINX=2 MEANS LOTS OF OUTPUT
C
C   DEFINITION OF PHRASE:
C     PHRASE = weight of the weld land+"extra" edge stringers
C
       CHARACTER*80 PHRASE
C  INSERT ADDITIONAL COMMON BLOCKS:
      COMMON/FV01/WLAND,TLAND,ECLAND,TWLAND,HWLAND,TFLAND,WFLAND,TSKIN
      REAL WLAND,TLAND,ECLAND,TWLAND,HWLAND,TFLAND,WFLAND,TSKIN
      COMMON/FV09/RADCYL,LENCYL,BSTR,TSTR,HSTR,BRNG,TRNG,HRNG,EMOD
      REAL RADCYL,LENCYL,BSTR,TSTR,HSTR,BRNG,TRNG,HRNG,EMOD
      COMMON/FV20/PX(20)
      REAL PX
      COMMON/FV26/GENBUK(20),GENBUKA(20),GENBUKF(20)
      REAL GENBUK,GENBUKA,GENBUKF
      COMMON/FV29/PANBUK(20),PANBUKA(20),PANBUKF(20)
      REAL PANBUK,PANBUKA,PANBUKF
      COMMON/FV32/STRESS(20),STRESSA(20),STRESSF(20)
      REAL STRESS,STRESSA,STRESSF
      COMMON/IV01/KLAND,NLAND,MLOWG,MHIGHG,MLOWP,MHIGHP
      INTEGER KLAND,NLAND,MLOWG,MHIGHG,MLOWP,MHIGHP
      COMMON/FV18/NU,DENS,WEIGHT
      REAL NU,DENS,WEIGHT
      COMMON/FV21/PY(20),PX0(20),PY0(20)
      REAL PY,PX0,PY0
C
C
C  INSERT SUBROUTINE STATEMENTS HERE.
C
      WEIGHT=(WLAND*TLAND+TWLAND*TWLAND+TFLAND*WFLAND)*DENS*386.4
C
      OBJGEN =WEIGHT
C
C
      RETURN
      END
C
C
C
```

created by GENOPT

created by GENOPT

created by GENOPT

only this line was "inserted by the GENOPT user" in Sub. OBJECT.

This is the weight per axial length of the weld land and ~~its~~ one of its edge stringers.

Table 11 (4 pages)

```
Results from the command:
diff behavior.weldland behvior.new > behavior.diff
The following is the behavior.diff file.
===================================================
453,535d452 (added to the GENOPT-created version of SUBROUTINE BEHX1)
<       COMMON/INSTAB/INDIC
<       COMMON/EIGB4M/EIGCOM(200),EIGNEG(200),EIGCRN
<       COMMON/WVEB4M/NWVCOM(200),NWVNEG(200),IWAVEB,NWVCRN
<       COMMON/EIGBUK/EIGCRT
<       COMMON/NWVBUK/NWVCRT
<       COMMON/BUCKN/N0BX,NMINBX,NMAXBX,INCRBX
<       COMMON/BUCKN0/N0B,NMAXB
<       COMMON/RBEGX/RBEG
<       COMMON/NXPART/NXLAND,NXWEB,NXFLNG,NXSKIN,NXSMER
<       COMMON/PRMOUT/IFILE3,IFILE4,IFILE8,IFILE9,IFIL11
<       COMMON/EIGALL/EIG0,EIG1,EIG2,EIG3,EIG4
<       COMMON/WAVALL/NWAV0,NWAV1,NWAV2,NWAV3,NWAV4
<       COMMON/NUMPAR/IPARX,IVARX,IALLOW,ICONSX,NDECX,NLINKX,NESCAP,ITYPEX
<       common/caseblock/CASE
<       CHARACTER*28 CASE
<       CHARACTER*35 CASA
<       REAL LENMOD
<       REAL NXLAND,NXWEB,NXFLNG,NOTLND,NXSKIN,NXSMER
< C
<       PI = 3.1415927
<       FLAND = FLOAT(NLAND)
<       NOTLND = 2.*PI*RADCYL - FLAND*WLAND
<       DENOM = NOTLND + NOTLND*HSTR*TSTR/(TSKIN*BSTR)
<      1        +2.*FLAND*HWLAND*TWLAND/TSKIN
<      1        +2.*FLAND*WFLAND*TFLAND/TSKIN
<      1           +FLAND*TLAND/TSKIN
<       NXSKIN = PX(ILOADX)/DENOM
<       NXLAND = NXSKIN*TLAND/TSKIN
<       NXWEB  = NXSKIN*TWLAND/TSKIN
<       NXFLNG = NXSKIN*TFLAND/TSKIN
<       NXSMER = NXSKIN*(1. + HSTR*TSTR/(TSKIN*BSTR))
< C
<       INDIC = 4
<       LENMOD = 1.0
<       RAVE = 100.*LENCYL*LENMOD/PI
<       RBEG = RAVE -0.707*RADCYL
<       N0B = MLOWG*100
<       NMAXB = MHIGHG*100
< C
< C  Get model without any weld lands:
<       IF (IMODX.EQ.0) THEN
< C        CALL BOSDEC(0,24,ILOADX,INDIC)
< C
<          IF (ITYPEX.EQ.2) THEN
< C          Get CASE.BEHX0 file for input for BIGBOSOR4...
< C          CASE.BEHX0 is an input file for BIGBOSOR4 for behavior no. 0:
< C          general buckling load without any weld lands
<            I=INDEX(CASE,' ')
<            IF(I.NE.0) THEN
<               CASA=CASE(:I-1)//'.BEHX0'
<            ELSE
<               CASA=CASE//'.BEHX0'
<            ENDIF
<            OPEN(UNIT=61,FILE=CASA,STATUS='UNKNOWN')
<            CALL BOSDEC(0,61,ILOADX,INDIC)
<            CLOSE(UNIT=61)
<            WRITE(IFILE,'(/,/,A,A,/,A)')
<      1    ' BIGBOSOR4 input file for:',
<      1    ' general buckling load, no weld lands',
<      1     CASA
<          ENDIF
<       ENDIF
< C
<       CALL BOSDEC(1,24,ILOADX,INDIC)
< C
<       IF (ITYPEX.EQ.2) THEN
< C      Get CASE.BEHX1 file for input for BIGBOSOR4...
< C      CASE.BEHX1 is an input file for BIGBOSOR4 for behavior no. 1:
< C      general buckling load
<          I=INDEX(CASE,' ')
<          IF(I.NE.0) THEN
```

created automatically by GENOPT

general buckling

"Fleshed out" by the GENOPT user

creates an input file for BIGBOSOR4: wcold.BEHX0

```
<             CASA=CASE(:I-1)//'.BEHX1'
<          ELSE
<             CASA=CASE//'.BEHX1'
<          ENDIF
<          OPEN(UNIT=61,FILE=CASA,STATUS='UNKNOWN')
<          CALL BOSDEC(1,61,ILOADX,INDIC)
<          CLOSE(UNIT=61)
<          WRITE(IFILE,'(/,/,A,A,/,A)')
<        1 ' BIGBOSOR4 input file for:',
<        1 ' general buckling load',
<        1   CASA
<         ENDIF
537,569d453
<         CALL B4READ
<         IF (IMODX.EQ.0) THEN
<            N0BX = N0B
<            NMINBX = N0B
<            NMAXBX = NMAXB
<            INCRBX = 100
<         ELSE
<            NOBX = NWAV1
<            NMINBX = NWAV1
<            NMAXBX = NWAV1
<            INCRBX = 100
<         ENDIF
<         REWIND IFILE9
<         CALL STOCM1(IFILE9)
<         CALL STOCM2(IFILE9)
<         CALL B4MAIN
<         CALL GASP(DUM1,DUM2,-2,DUM3)
<         IF (IMODX.EQ.0) THEN
<            EIG1 = EIGCRT
<            NWAV1= NWVCRT
<         ENDIF
< C
<         WRITE(IFILE,'(/,A)')
<        1 ' GENERAL BUCKLING LOAD FACTORS AND MODES (BEHX1)'
<         DO 10 I = 1,IWAVEB
<            WRITE(IFILE,'(A,1P,E12.4,A,I4,A)')
<        1   '       ',EIGCOM(I),'(',NWVCOM(I),')'
<     10 CONTINUE
<         WRITE(IFILE,'(A,1P,E12.4)')
<        1' Critical buckling load factor, GENBUK=',EIGCRT
<         WRITE(IFILE,'(A,I5)')
<        1' Critical number of circumferential waves, NWVCRT=',NWVCRT
<         GENBUK(ILOADX) = EIGCRT
---
639,716d522 (added to the GENOPT-created version of SUBROUTINE BEHX2)
<         COMMON/INSTAB/INDIC
<         COMMON/EIGB4M/EIGCOM(200),EIGNEG(200),EIGCRN
<         COMMON/WVEB4M/NWVCOM(200),NWVNEG(200),IWAVEB,NWVCRN
<         COMMON/EIGBUK/EIGCRT
<         COMMON/NWVBUK/NWVCRT
<         COMMON/BUCKN/N0BX,NMINBX,NMAXBX,INCRBX
<         COMMON/BUCKN0/N0B,NMAXB
<         COMMON/RBEGX/RBEG
<         COMMON/PRMOUT/IFILE3,IFILE4,IFILE8,IFILE9,IFIL11
<         COMMON/EIGALL/EIG0,EIG1,EIG2,EIG3,EIG4
<         COMMON/WAVALL/NWAV0,NWAV1,NWAV2,NWAV3,NWAV4
<         COMMON/NUMPAR/IPARX,IVARX,IALLOW,ICONSX,NDECX,NLINKX,NESCAP,ITYPEX
<         common/caseblock/CASE
<         CHARACTER*28 CASE
<         CHARACTER*35 CASA
<         REAL NXLAND,NXWEB,NXFLNG,NOTLND,NXSKIN,NXSMER
< C
<         PI = 3.1415927
<         FLAND = FLOAT(NLAND)
<         NOTLND = 2.*PI*RADCYL - FLAND*WLAND
<         DENOM = NOTLND + NOTLND*HSTR*TSTR/(TSKIN*BSTR)
<        1        +2.*FLAND*HWLAND*TWLAND/TSKIN
<        1        +2.*FLAND*WFLAND*TFLAND/TSKIN
<        1           +FLAND*TLAND/TSKIN
<         NXSKIN = PX(ILOADX)/DENOM
<         NXLAND = NXSKIN*TLAND/TSKIN
<         NXWEB  = NXSKIN*TWLAND/TSKIN
<         NXFLNG = NXSKIN*TFLAND/TSKIN
<         NXSMER = NXSKIN*(1. + HSTR*TSTR/(TSKIN*BSTR))
```

Table 11 (p. 2 of 4)

create an input file for BIGBOSOR4 (wcold.BEHX1)

Execute BIGBOSOR4 preprocess

Execute BIGBOSOR4 main process

re-initialize GASP

inter-ring buckling

Table 11 (p. 3 of 4)

```
< C
<       INDIC = 4
<       RAVE = 100.*BRNG/PI
<       RBEG = RAVE -0.707*RADCYL
<       N0B = MLOWP*100
<       NMAXB = MHIGHP*100
< C
<       CALL BOSDEC(2,24,ILOADX,INDIC)
< C
<       IF (ITYPEX.EQ.2) THEN
< C      Get CASE.BEHX2 file for input for BIGBOSOR4...
< C      CASE.BEHX2 is an input file for BIGBOSOR4 for behavior no. 2:
< C      "panel" buckling load
<          I=INDEX(CASE,' ')
<          IF(I.NE.0) THEN
<             CASA=CASE(:I-1)//'.BEHX2'
<          ELSE
<             CASA=CASE//'.BEHX2'
<          ENDIF
<          OPEN(UNIT=61,FILE=CASA,STATUS='UNKNOWN')
<          CALL BOSDEC(2,61,ILOADX,INDIC)
<          CLOSE(UNIT=61)
<          WRITE(IFILE,'(/,/,A,A,/,A)')
<      1 ' BIGBOSOR4 input file for:',
<      1 ' "panel" buckling load',
<      1   CASA
<       ENDIF
< C
<       CALL B4READ
<       IF (IMODX.EQ.0) THEN
<          N0BX = N0B
<          NMINBX = N0B
<          NMAXBX = NMAXB
<          INCRBX = 100
<       ELSE
<          N0BX = NWAV2
<          NMINBX = NWAV2
<          NMAXBX = NWAV2
<          INCRBX = 100
<       ENDIF
<       REWIND IFILE9
<       CALL STOCM1(IFILE9)
<       CALL STOCM2(IFILE9)
<       CALL B4MAIN
<       CALL GASP(DUM1,DUM2,-2,DUM3)
<       IF (IMODX.EQ.0) THEN
<          EIG2 = EIGCRT
<          NWAV2= NWVCRT
<       ENDIF
718,728d523
<       WRITE(IFILE,'(/,A)')
<      1 ' "PANEL" BUCKLING LOAD FACTORS AND MODES (BEHX2)'
<       DO 10 I = 1,IWAVEB
<          WRITE(IFILE,'(A,1P,E12.4,A,I4,A)')
<      1  '       ',EIGCOM(I),'(',NWVCOM(I),')'
<    10 CONTINUE
<       WRITE(IFILE,'(A,1P,E12.4)')
<      1' Critical buckling load factor, PANBUK=',EIGCRT
<       WRITE(IFILE,'(A,I5)')
<      1' Critical number of circumferential waves, NWVCRT=',NWVCRT
<       PANBUK(ILOADX) = EIGCRT


797d591 (added to the GENOPT-created version of SUBROUTINE BEHX3)
<       REAL NXLAND,NXWEB,NXFLNG,NOTLND,NXSKIN,NXSMER
799,814d592
<       PI = 3.1415927
<       FLAND = FLOAT(NLAND)
<       NOTLND = 2.*PI*RADCYL - FLAND*WLAND
<       DENOM = NOTLND + NOTLND*HSTR*TSTR/(TSKIN*BSTR)
<      1       +2.*FLAND*HWLAND*TWLAND/TSKIN
<      1       +2.*FLAND*WFLAND*TFLAND/TSKIN
<      1          +FLAND*TLAND/TSKIN
<       NXSKIN = PX(ILOADX)/DENOM
<       NXLAND = NXSKIN*TLAND/TSKIN
<       NXWEB  = NXSKIN*TWLAND/TSKIN
<       NXFLNG = NXSKIN*TFLAND/TSKIN
<       NXSMER = NXSKIN*(1. + HSTR*TSTR/(TSKIN*BSTR))
```

↑ stress in weld land & "extra" weld land stringers

```
< C
<       STRLND = NXLAND/TLAND
<       STRWEB = NXWEB/TWLAND
<       STRFLG = NXFLNG/TFLAND
816,817d593
<       STRESS(ILOADX) = MAX(STRLND,STRWEB)
<       STRESS(ILOADX) = MAX(STRESS(ILOADX),STRFLG)

994d769 (added to the GENOPT-created version of SUBROUTINE OBJECT)
<       WEIGHT=(WLAND*TLAND+TWLAND*TWLAND+TFLAND*WFLAND)*DENS*386.4
```

Table 11 (p. 4-14)
stress from membrane compression

objective

Table 12 (6 pages)

```
C=DECK       BOSDEC
C
C  PURPOSE IS TO SET UP BIGBOSOR4 INPUT FILE FOR "weldland"
C
      SUBROUTINE BOSDEC(INDX,IFIL14,ILOADX,INDIC)
C  Insert labelled common blocks: weldland.COM
      COMMON/FV01/WLAND,TLAND,ECLAND,TWLAND,HWLAND,TFLAND,WFLAND,TSKIN
      REAL WLAND,TLAND,ECLAND,TWLAND,HWLAND,TFLAND,WFLAND,TSKIN
      COMMON/FV09/RADCYL,LENCYL,BSTR,TSTR,HSTR,BRNG,TRNG,HRNG,EMOD
      REAL RADCYL,LENCYL,BSTR,TSTR,HSTR,BRNG,TRNG,HRNG,EMOD
      COMMON/FV20/PX(20)
      REAL PX
      COMMON/FV26/GENBUK(20),GENBUKA(20),GENBUKF(20)
      REAL GENBUK,GENBUKA,GENBUKF
      COMMON/FV29/PANBUK(20),PANBUKA(20),PANBUKF(20)
      REAL PANBUK,PANBUKA,PANBUKF
      COMMON/FV32/STRESS(20),STRESSA(20),STRESSF(20)
      REAL STRESS,STRESSA,STRESSF
      COMMON/IV01/KLAND,NLAND,MLOWG,MHIGHG,MLOWP,MHIGHP
      INTEGER KLAND,NLAND,MLOWG,MHIGHG,MLOWP,MHIGHP
      COMMON/FV18/NU,DENS,WEIGHT
      REAL NU,DENS,WEIGHT
      COMMON/FV21/PY(20),PX0(20),PY0(20)
      REAL PY,PX0,PY0
C  end of weldland.COM
      COMMON/BUCKN0/N0B,NMAXB
      COMMON/RBEGX/RBEG
      COMMON/NXPART/NXLAND,NXWEB,NXFLNG,NXSKIN,NXSMER
      DIMENSION ISMEAR(50),NX(50),R1(50),R2(50),Z1(50),Z2(50)
      DIMENSION THICK(50),NSHAPE(50),RC(50),ZC(50),IFLANG(50)
      DIMENSION DR(50),DZ(50),NMESH(50),JLAND(50)
      REAL NXLAND,NXWEB,NXFLNG,NXSKIN,NXSMER,NX
      COMMON/PRMOUT/IFILE3,IFILE4,IFILE8,IFILE9,IFIL11
      REWIND IFIL14
C
      WRITE(IFILE4,3)
    3 FORMAT(//'  ***************  BOSDEC  ******************'/
     1'  The purpose of BOSDEC is to set up an input file, NAME.ALL,'/
     1'  for a cylindrical shell. NAME is your name for'/
     1'  the case. The file NAME.ALL is a BOSOR4 input "deck" used'/
     1'  by SUBROUTINE B4READ.'/
     1'  ************************************************'/)
C
      CALL MOVER(0,0,ISMEAR,1,50)
      CALL MOVER(0,0,NSHAPE,1,50)
      CALL MOVER(0,0,IFLANG,1,50)
      CALL MOVER(0,0,NMESH,1,50)
      CALL MOVER(0,0,JLAND,1,50)
      CALL MOVER(0.,0,R1,1,50)
      CALL MOVER(0.,0,Z1,1,50)
      CALL MOVER(0.,0,R2,1,50)
      CALL MOVER(0.,0,Z2,1,50)
      CALL MOVER(0.,0,RC,1,50)
      CALL MOVER(0.,0,ZC,1,50)
      CALL MOVER(0.,0,DR,1,50)
      CALL MOVER(0.,0,DZ,1,50)
      CALL MOVER(0.,0,NX,1,50)
      CALL MOVER(0.,0,THICK,1,50)
C
      IF (INDX.EQ.0) THEN
         PI = 3.1415927
         R1(1) = RBEG
         Z1(1) = 0.
         R2(1) = RBEG
         Z2(1) = 2.0*RADCYL
         RC(1) = RBEG
         ZC(1) = RADCYL
         NX(1) = PX(ILOADX)/(2.0*PI*RADCYL)
         NSHAPE(1) = 2
         THICK(1) = TSKIN
         NMESH(1) = 91
         NSEGS = 1
         WRITE(IFIL14,'(A)')
     1 ' general buckling, no weld land (INDIC=4)'
         WRITE(IFIL14,'(A,I3,A)')
     1 ' 4, 1, 0, 0, ',NSEGS,' $ INDIC,NPRT,ISTRESS,IPRE,NSEG'
         I = 1
         WRITE(IFIL14,'(A,4I6)')' H   $ Segment number ',I,I,I,I
```

BIGBOSOR4 input file with no weld lands

Table 12 (p. 246)

```
      WRITE(IFIL14,'(I4,A,I4,A)') NMESH(I),', 3, ',NSHAPE(I),
     1 ' $ NMESH,NTYPEH,NSHAPE'
      WRITE(IFIL14,'(1P,2E14.6,A)') R1(I),Z1(I),' $ R1,Z1'
      WRITE(IFIL14,'(1P,2E14.6,A)') R2(I),Z2(I),' $ R2,Z2'
      WRITE(IFIL14,'(1P,2E14.6,A)') RC(I),ZC(I),
     1  ', -1.  $ RC,ZC,SUR'
      WRITE(IFIL14, '(A,1P,E14.6,A)')' 0, 3, ',THICK(I)/2.,
     1'  $ IMP,NTYPEZ,ZVAL'
      WRITE(IFIL14,'(A)')' N  $ do not print r(s), etc.'
      WRITE(IFIL14,'(A)')
     1 ' 0, 0., 2, 0, 2  $ NRINGS,K,NSTRES,NRLOAD,NTYPE'
      WRITE(IFIL14,'(1P,2E14.6,A)') Z1(I),Z2(I),' $ Z1,Z2'
      WRITE(IFIL14,'(A)')' 0., 0.  $ FN10(1), FN10(2)'
      WRITE(IFIL14,'(1P,2E14.6,A)') NX(I),NX(I),' $ FN20(1),FN20(2)'
      WRITE(IFIL14,'(A)')' N  $ do not print prestresses'
      WRITE(IFIL14, '(A,1P,3E14.6,A)')' 2, ',EMOD,NU,DENS,
     1 ' $ NWALL,EMOD,NU,DENS'
      WRITE(IFIL14,'(A)')' 0.  $ ALPHA'
      NRS = 1
      WRITE(IFIL14,'(I3,A)') NRS,', 0  $ NRS,NSUR'
      WRITE(IFIL14,'(A)')' N  $ do not print ref.surf.location'
      WRITE(IFIL14,'(A)')' Y  $ are there stringers'
      N1 = 2.*PI*R1(I)/BRNG
      WRITE(IFIL14,'(I8,A)') N1,', 0  $ N1,K1'
      WRITE(IFIL14, '(1P,3E14.6,A)') EMOD,NU,DENS,' $ EMOD,NU,DENS'
      WRITE(IFIL14,'(A)')' Y  $ is stringer cross section constant'
      WRITE(IFIL14,'(A)')' Y  $ is stringer cross section rectangu'
      WRITE(IFIL14,'(1P,2E14.6,A)') TRNG,HRNG,' $ TRNG,HRNG'
      WRITE(IFIL14,'(A)')' Y  $ are there rings'
      WRITE(IFIL14,'(A)')' 0  $ K2'
      WRITE(IFIL14, '(1P,3E14.6,A)') EMOD,NU,DENS,' $ EMOD,NU,DENS'
      WRITE(IFIL14,'(A)')' Y  $ is ring cross section constant'
      WRITE(IFIL14,'(A)')' Y  $ is ring cross section rectangular'
      WRITE(IFIL14,'(1P,E14.6,A)') BSTR,'  $ BSTR'
      WRITE(IFIL14,'(1P,2E14.6,A)') TSTR,HSTR,' $ TSTR,HSTR'
      WRITE(IFIL14,'(A)')' N  $ do not print C(i,j)'
      WRITE(IFIL14,'(A)')' N  $ do not print distributed loads'
      WRITE(IFIL14,'(A)')' H  $ GLOBAL DATA BEGINS...'
      WRITE(IFIL14,'(A)')' 0  $ NLAST'
      WRITE(IFIL14,'(A)')' N  $ any expanded plots?'
      NMINB = N0B
      INCRB = 100
      NVEC  = 1
      WRITE(IFIL14,'(5I6,A)') N0B,NMINB,NMAXB,INCRB,NVEC,
     1 '  $ N0B,NMINB,NMAXB,INCRB,NVEC'
      WRITE(IFIL14,'(A)')' H  $ CONSTRAINT CONDITIONS FOLLOW....'
      WRITE(IFIL14,'(I6,A)') NSEGS,'  $ how many segments?'
      WRITE(IFIL14,'(A,4I6)')
     1 ' H  $ CONSTRAINT CONDITIONS FOR SEGMENT ',I,I,I,I
      WRITE(IFIL14,'(A,A)')' 0, 2, 1, 0, 0, 1, 1',
     1 ' $ poles,ground,etc.'
      WRITE(IFIL14,'(A)')' 0., 0.  $ D1,D2'
      WRITE(IFIL14,'(A)')' Y  $ same for preb. & buck.'
      WRITE(IFIL14,'(A,A)')' 91, 0, 0, 1, 1',
     1 ' $ INODE,etc.'
      WRITE(IFIL14,'(A)')' 0., 0.  $ D1,D2'
      WRITE(IFIL14,'(A)')' Y  $ same for preb. & buck.'
      WRITE(IFIL14,'(A)')' N  $ joined to lower segs?'
      WRITE(IFIL14,'(A)')' N  $ rigid body motion possible?'
      WRITE(IFIL14,'(A)')' Y  $ do you want output for seg.?'
      WRITE(IFIL14,'(A)')' N  $ descrete ring forces?'
      RETURN
    ENDIF
C
    PI = 3.1415927
    R = RADCYL
    RLAND = RADCYL - ECLAND
    FLAND = FLOAT(NLAND)
    DPHI = 2.*PI/FLAND
    PHI0 = -DPHI
    NLAND2 = NLAND/2 + 1
    IADD = 2
    IF (KLAND.EQ.1) IADD = 4
    IF (KLAND.EQ.2) IADD = 6
C
    DO 100 ILAND = 1,NLAND2
C
    ISEG = (ILAND - 1)*IADD + 1
```

BIGBOSORY input file for shell with no weld bands (not used for optimization)

Table 12 (p. 3-16)

```
      PHI0 = PHI0 + DPHI
      PHIM = PHI0 -0.5*WLAND/RLAND
      PHIP = PHI0 +0.5*WLAND/RLAND
      SP   = SIN(PHIP)
      CP   = COS(PHIP)
      SM   = SIN(PHIM)
      CM   = COS(PHIM)
      IF (ABS(PHI0).LT.0.00001) THEN
         PHI0 = 0.
         SM = 0.
         CM = 1.
      ENDIF
      DIFF = ABS(PI - PHI0)
      IF (DIFF.LT.0.00001) THEN
         PHI0 = PI
         SP = 0.
         CP = -1.
      ENDIF
      RPSKIN  = RBEG + R*SP
      ZPSKIN  = R*(1. - CP)
      RMSKIN  = RBEG + R*SM
      ZMSKIN  = R*(1. - CM)
      RPLAND  = RBEG + RLAND*SP
      ZPLAND  = ECLAND + RLAND*(1. - CP)
      RMLAND  = RBEG + RLAND*SM
      ZMLAND  = ECLAND + RLAND*(1. - CM)
      DMR = -ECLAND*SM
      DMZ =  ECLAND*CM
      DPR =  ECLAND*SP
      DPZ = -ECLAND*CP
C
      NSHAPE(ISEG) = 2
      DR(ISEG) = DMR
      DZ(ISEG) = DMZ
      R1(ISEG) = RMLAND
      Z1(ISEG) = ZMLAND
      R2(ISEG) = RPLAND
      Z2(ISEG) = ZPLAND
      RC(ISEG) = RBEG
      ZC(ISEG) = R
      NX(ISEG) = NXLAND
      THICK(ISEG) = TLAND
      JLAND(ISEG) = 1
      NSEGS = ISEG
C
      IF (PHI0.LT.0.9999*PI) THEN
         ISEGP = ISEG + 1
         IF (KLAND.EQ.1) ISEGP = ISEG + 2
         IF (KLAND.EQ.2) ISEGP = ISEG + 3
         NSHAPE(ISEGP) = 2
         DR(ISEGP) = DPR
         DZ(ISEGP) = DPZ
         R1(ISEGP) = RPSKIN
         Z1(ISEGP) = ZPSKIN
         PHIM = PHI0  + DPHI -0.5*WLAND/RLAND
         IF (PHIM.GT.PI) PHIM = PI
         SMM  = SIN(PHIM)
         CMM  = COS(PHIM)
         RMSKIN  = RBEG + R*SMM
         ZMSKIN  = R*(1. - CMM)
         R2(ISEGP) = RMSKIN
         Z2(ISEGP) = ZMSKIN
         RC(ISEGP) = RBEG
         ZC(ISEGP) = R
         NX(ISEGP) = NXSMER
         ISMEAR(ISEGP) = 1
         THICK(ISEGP) = TSKIN
         NSEGS = ISEGP
      ENDIF
C
      IF (KLAND.GT.0) THEN
         RPWEB1 = RBEG + (R - 0.5*TSKIN)*SP
         ZPWEB1 = R -(R -0.5*TSKIN)*CP
         RMWEB1 = RBEG + (R - 0.5*TSKIN)*SM
         ZMWEB1 = R -(R -0.5*TSKIN)*CM
         RPWEB2 = RBEG + (R - 0.5*TSKIN -HWLAND)*SP
         ZPWEB2 = R -(R -0.5*TSKIN -HWLAND)*CP
         RMWEB2 = RBEG + (R - 0.5*TSKIN -HWLAND)*SM
```

GET end points of shell segments, loading, center of curvature, radial & axial discontinuities between shell segments

stringer web

Table 12 (p. 4 of 6)

```
            ZMWEB2 = R -(R -0.5*TSKIN -HWLAND)*CM
            DMR = -0.5*TSKIN*SM
            DMZ =  0.5*TSKIN*CM
            DPR =  (ECLAND -0.5*TSKIN)*SP
            DPZ = -(ECLAND -0.5*TSKIN)*CP
            IF (ISEG.GT.1) THEN
               IF (KLAND.EQ.1) ISEGM = ISEG - 1
               IF (KLAND.EQ.2) ISEGM = ISEG - 2
               R1(ISEGM) = RMWEB1
               Z1(ISEGM) = ZMWEB1
               R2(ISEGM) = RMWEB2
               Z2(ISEGM) = ZMWEB2
               NSHAPE(ISEGM) = 1
               DR(ISEGM) = DMR
               DZ(ISEGM) = DMZ
               NX(ISEGM) = NXWEB
               THICK(ISEGM) = TWLAND
            ENDIF
C
            IF (PHI0.LT.0.9999*PI) THEN
               ISEGP = ISEG + 1
               R1(ISEGP) = RPWEB1
               Z1(ISEGP) = ZPWEB1
               R2(ISEGP) = RPWEB2
               Z2(ISEGP) = ZPWEB2
               NSHAPE(ISEGP) = 1
               DR(ISEGP) = DPR
               DZ(ISEGP) = DPZ
               NX(ISEGP) = NXWEB
               THICK(ISEGP) = TWLAND
            ENDIF
C
            IF (KLAND.GT.1) THEN
               RPFLG2 = RPWEB2 +0.5*WFLAND*CP
               ZPFLG2 = ZPWEB2 +0.5*WFLAND*SP
               RPFLG1 = RPWEB2 -0.5*WFLAND*CP
               ZPFLG1 = ZPWEB2 -0.5*WFLAND*SP
               RMFLG2 = RMWEB2 +0.5*WFLAND*CM
               ZMFLG2 = ZMWEB2 +0.5*WFLAND*SM
               RMFLG1 = RMWEB2 -0.5*WFLAND*CM
               ZMFLG1 = ZMWEB2 -0.5*WFLAND*SM
C
               IF (ISEG.GT.1) THEN
                  ISEG1M = ISEG - 1
                  R1(ISEG1M) = RMFLG1
                  Z1(ISEG1M) = ZMFLG1
                  R2(ISEG1M) = RMFLG2
                  Z2(ISEG1M) = ZMFLG2
                  NSHAPE(ISEG1M) = 1
                  NX(ISEG1M) = NXFLNG
                  THICK(ISEG1M) = TFLAND
                  IFLANG(ISEG1M) = 1
               ENDIF
               IF (PHI0.LT.0.9999*PI) THEN
                  ISEG2P = ISEG + 2
                  R1(ISEG2P) = RPFLG1
                  Z1(ISEG2P) = ZPFLG1
                  R2(ISEG2P) = RPFLG2
                  Z2(ISEG2P) = ZPFLG2
                  NSHAPE(ISEG2P) = 1
                  NX(ISEG2P) = NXFLNG
                  THICK(ISEG2P) = TFLAND
                  IFLANG(ISEG2P) = 1
               ENDIF
            ENDIF
         ENDIF
C
  100 CONTINUE
C
         WRITE(IFILE4,'(/,A)')
        1' NSHAPE,ISMEAR, R1,       Z1,       R2,       Z2,       RC,       ZC'
         DO 200 I = 1,NSEGS
            WRITE(IFILE4,'(2I5,1P,6E14.6)')
        1 NSHAPE(I),ISMEAR(I),R1(I),Z1(I),R2(I),Z2(I),RC(I),ZC(I)
  200 CONTINUE
C
C234567890123456789012345678901234567890123456789012345678901234567890123456789012
         WRITE(IFILE4,'(/,A)')
```

web of weld land — 5th frame edge

outstanding flange of Tee-shaped weld → land edge stiffener

Table 12 (p. 5 of 6)

```
     1' NSHAPE,ISMEAR, DR,     DZ,     NX,     THICK     IFLANG   JLAND'
      DO 300 I = 1,NSEGS
         WRITE(IFILE4,'(2I5,1P,4E14.6,2I5)')
     1 NSHAPE(I),ISMEAR(I),DR(I),DZ(I),NX(I),THICK(I),
     1 IFLANG(I),JLAND(I)
  300 CONTINUE
C
      IF (INDX.EQ.1) WRITE(IFIL14,'(A)')
     1 ' general buckling (INDIC=4)'
      IF (INDX.EQ.2) WRITE(IFIL14,'(A)')
     1 ' inter-ring ("panel") buckling (INDIC=4)'
      WRITE(IFIL14,'(A,I3,A)')
     1' 4, 1, 0, 0, ',NSEGS,' $ INDIC,NPRT,ISTRESS,IPRE,NSEG'
C
      DO 500 I = 1,NSEGS
C
      WRITE(IFIL14,'(A,4I6)')' H   $ Segment number ',I,I,I,I
      IF (JLAND(I).EQ.1) NMESH(I) = 21
      IF (ISMEAR(I).EQ.1.AND.JLAND(I).EQ.0) NMESH(I) = 91
      IF (ISMEAR(I).EQ.0.AND.JLAND(I).EQ.0) NMESH(I) = 11
      WRITE(IFIL14,'(I4,A,I4,A)') NMESH(I),', 3, ',NSHAPE(I),
     1' $ NMESH,NTYPEH,NSHAPE'
      WRITE(IFIL14,'(1P,2E14.6,A)') R1(I),Z1(I),' $ R1,Z1'
      WRITE(IFIL14,'(1P,2E14.6,A)') R2(I),Z2(I),' $ R2,Z2'
      IF (NSHAPE(I).EQ.2)
     1WRITE(IFIL14,'(1P,2E14.6,A)') RC(I),ZC(I),
     1', -1.  $ RC,ZC,SUR'
      WRITE(IFIL14, '(A,1P,E14.6,A)')' 0, 3, ',THICK(I)/2.,
     1'  $ IMP,NTYPEZ,ZVAL'
      WRITE(IFIL14,'(A)')' N  $ do not print r(s), etc.'
      WRITE(IFIL14,'(A)')
     1 ' 0, 0., 2, 0, 2  $ NRINGS,K,NSTRES,NRLOAD,NTYPE'
      WRITE(IFIL14,'(1P,2E14.6,A)') Z1(I),Z2(I),' $ Z1,Z2'
      WRITE(IFIL14,'(A)')' 0., 0.  $ FN10(1), FN10(2)'
      WRITE(IFIL14,'(1P,2E14.6,A)') NX(I),NX(I),' $ FN20(1),FN20(2)'
      WRITE(IFIL14,'(A)')' N  $ do not print prestresses'
      WRITE(IFIL14, '(A,1P,3E14.6,A)')' 2, ',EMOD,NU,DENS,
     1' $ NWALL,EMOD,NU,DENS'
      WRITE(IFIL14,'(A)')' 0.  $ ALPHA'
      NRS = 0
      IF (ISMEAR(I).EQ.1) NRS = 1
      WRITE(IFIL14,'(I3,A)') NRS,', 0  $ NRS,NSUR'
      IF (ISMEAR(I).EQ.0) THEN
        WRITE(IFIL14,'(A)')' N  $ do not print ref.surf.location'
        WRITE(IFIL14,'(A)')' N  $ do not print C(i,j)'
        WRITE(IFIL14,'(A)')' N  $ do not print distributed loads'
      ELSE
        WRITE(IFIL14,'(A)')' N  $ do not print ref.surf.location'
        IF (INDX.EQ.2) WRITE(IFIL14,'(A)')' N  $ are there stringers'
C
        IF (INDX.EQ.1) THEN
          WRITE(IFIL14,'(A)')' Y  $ are there stringers'
          N1 = 2.*PI*R1(I)/BRNG
          WRITE(IFIL14,'(I8,A)') N1,', 0  $ N1,K1'
          WRITE(IFIL14, '(1P,3E14.6,A)') EMOD,NU,DENS,' $ EMOD,NU,DENS'
          WRITE(IFIL14,'(A)')' Y  $ is stringer cross section constant'
          WRITE(IFIL14,'(A)')' Y  $ is stringer cross section rectangu'
          WRITE(IFIL14,'(1P,2E14.6,A)') TRNG,HRNG,' $ TRNG,HRNG'
        ENDIF
C
C234567890123456789012345678901234567890123456789012345678901234567890123456789012
        WRITE(IFIL14,'(A)')' Y  $ are there rings'
        WRITE(IFIL14,'(A)')' 0  $ K2'
        WRITE(IFIL14, '(1P,3E14.6,A)') EMOD,NU,DENS,' $ EMOD,NU,DENS'
        WRITE(IFIL14,'(A)')' Y  $ is ring cross section constant'
        WRITE(IFIL14,'(A)')' Y  $ is ring cross section rectangular'
        WRITE(IFIL14,'(1P,E14.6,A)') BSTR,'  $ BSTR'
        WRITE(IFIL14,'(1P,2E14.6,A)') TSTR,HSTR,' $ TSTR,HSTR'
        WRITE(IFIL14,'(A)')' N  $ do not print C(i,j)'
        WRITE(IFIL14,'(A)')' N  $ do not print distributed loads'
      ENDIF
C
  500 CONTINUE
C
      WRITE(IFIL14,'(A)')' H  $ GLOBAL DATA BEGINS...'
      WRITE(IFIL14,'(A)')' 0  $ NLAST'
      WRITE(IFIL14,'(A)')' N  $ any expanded plots?'
      NMINB = N0B
```

create BIGBOSOR4 input file for the cylindrical shell with NLAND weld lands

Table 12 (p. 6-16)

```
      INCRB = 100
      NVEC  = 1
      WRITE(IFIL14,'(5I6,A)') N0B,NMINB,NMAXB,INCRB,NVEC,
     1 '  $ N0B,NMINB,NMAXB,INCRB,NVEC'
      WRITE(IFIL14,'(A)')' H  $ CONSTRAINT CONDITIONS FOLLOW....'
      WRITE(IFIL14,'(I6,A)') NSEGS,'  $ how many segments?'
C
      DO 800 I = 1,NSEGS
C
      WRITE(IFIL14,'(A,4I6)')
     1' H  $ CONSTRAINT CONDITIONS FOR SEGMENT ',I,I,I,I
      IF (I.EQ.1) THEN
        WRITE(IFIL14,'(A,A)')' 0, 1, 1, 0, 0, 1, 1',
     1' $ poles,ground,etc.'
        WRITE(IFIL14,'(1P,2E14.6,A)') DR(I),DZ(I),'  $ D1,D2'
        WRITE(IFIL14,'(A)')' Y  $ same for preb. & buck.'
        WRITE(IFIL14,'(A)')' N  $ joined to lower segs?'
      ENDIF
C
      IF (I.GT.1.AND.I.LT.NSEGS) THEN
        WRITE(IFIL14,'(A)')' 0, 0  $ poles,ground'
        WRITE(IFIL14,'(A)')' Y  $ joined to lower segs?'
        INODE = 1
        IF (IFLANG(I).EQ.1) INODE = NMESH(I)/2 + 1
        JSEG = I - 1
        IF (ISMEAR(I).EQ.1.OR.JLAND(I).EQ.1)
     1  JSEG = I - 1 - KLAND
        JNODE = NMESH(JSEG)
        WRITE(IFIL14,'(A,3I5,A)')' 1, ',INODE,JSEG,JNODE,
     1' $ prev.seg.,INODE,JSEG,JNODE'
        WRITE(IFIL14,'(A)')
     1' 1, 1, 1, 1  $ IUSTAR,IVSTAR,IWSTAR,ICHI'
        WRITE(IFIL14,'(1P,2E14.6,A)') DR(I),DZ(I),'  $ D1,D2'
        WRITE(IFIL14,'(A)')' Y  $ same for preb. & buck.'
      ENDIF
C
      IF (I.EQ.NSEGS) THEN
        WRITE(IFIL14,'(A,I5,A,A)')
     1' 0, 1, ',NMESH(I),' 0, 0, 1, 1',
     1' $ poles,ground,etc'
        WRITE(IFIL14,'(1P,2E14.6,A)') DR(I),DZ(I),'  $ D1,D2'
        WRITE(IFIL14,'(A)')' Y  $ same for preb. & buck.'
        WRITE(IFIL14,'(A)')' Y  $ joined to lower segs?'
        INODE = 1
        JSEG = I - 1
        IF (ISMEAR(I).EQ.1.OR.JLAND(I).EQ.1)
     1  JSEG = I - 1 - KLAND
        JNODE = NMESH(JSEG)
        WRITE(IFIL14,'(A,3I5,A)')' 1, ',INODE,JSEG,JNODE,
     1' $ prev.seg.,INODE,JSEG,JNODE'
        WRITE(IFIL14,'(A)')
     1' 1, 1, 1, 1  $ IUSTAR,IVSTAR,IWSTAR,ICHI'
        WRITE(IFIL14,'(1P,2E14.6,A)') DR(I),DZ(I),'  $ D1,D2'
        WRITE(IFIL14,'(A)')' Y  $ same for preb. & buck.'
      ENDIF
C
  800 CONTINUE
C
      WRITE(IFIL14,'(A)')' N  $ rigid body motion possible?'
C
      DO 900 I = 1,NSEGS
        WRITE(IFIL14,'(A)')' Y  $ do you want output for seg.?'
  900 CONTINUE
      WRITE(IFIL14,'(A)')' N  $ descrete ring forces?'
C
      RETURN
      END
```

create BIGBOSOR4 input file for cylindrical shell with NLAND weld lands

Table 13 wcold.BEG

```
n               $ Do you want a tutorial session and tutorial output?
 4.000000       $ width of the weld land: WLAND
0.3250000       $ thickness of the weld land: TLAND
0.1334045       $ eccentricity of the weld land: ECLAND
        2       $ index for weld land edge stringer (0 or 1 or 2): KLAND
0.2000000       $ web thickness of the weld land edge stringer: TWLAND
 1.500000       $ height of the web of the weld land edge stringer: HWLAND
0.2000000       $ thickness of the outstanding flange of the stringer: TFLAND
 1.500000       $ width of the outstanding flange of the stringer: WFLAND
        3       $ number of weld lands in 360 degrees: NLAND
0.0581910       $ thickness of the cylindrical shell skin: TSKIN
       48       $ radius of the cylindrical shell: RADCYL
    68.75       $ length of the cylindrical shell: LENCYL
 2.277300       $ stringer spacing: BSTR
0.0864620       $ stringer thickness: TSTR
 0.989770       $ stringer height: HSTR
 13.97400       $ ring spacing: BRNG
 0.185760       $ ring thickness: TRNG
 0.971730       $ ring height: HRNG
0.1100000E+08   $ Young's modulus: EMOD
0.3000000       $ Poisson ratio: NU
0.2500000E-03   $ mass density (aluminum = 0.00025 in English units): DENS
         1      $ low end of the M-range for general buckling: MLOWG
        10      $ High end of the M-range for general buckling: MHIGHG
         1      $ low end of the M-range for "panel" buckling: MLOWP
        10      $ high end of the M-range for "panel" buckling: MHIGHP
         1      $ Number NCASES  of load cases (environments): NCASES
-669234.63      $ total axial load (2*pi*r*Nx): PX( 1)
         0      $ uniform normal pressure: PY( 1)
         0      $ axial load in Load Set B: PX0( 1)
         0      $ pressure in Load Set B: PY0( 1)
 1.000000       $ allowable for general buckling: GENBUKA( 1)
 2.000000       $ factor of safety for general buckling: GENBUKF( 1)
 1.000000       $ allowable for "panel" buckling: PANBUKA( 1)
 1.500000       $ factor of safety for "panel" buckling: PANBUKF( 1)
 70000.00       $ maximum allowable stress: STRESSA( 1)
 1.000000       $ factor of safety for effective stress: STRESSF( 1)
```

Tee shaped →

note →

Previously optimized by PANDA2. See Appendix 1.

input for BEGIN

Table 14 wscold.DEC

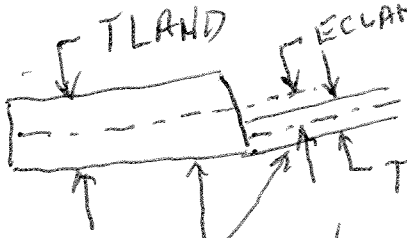
```
n                 $ Do you want a tutorial session and tutorial output?
WLAND → (1)       $ Choose a decision variable (1,2,3,...)
      4           $ Lower bound of variable no.( 1)
      6           $ Upper bound of variable no.( 1)
y                 $ Any more decision variables (Y or N) ?
TLAND → (2)       $ Choose a decision variable (1,2,3,...)
  0.1000000       $ Lower bound of variable no.( 2)
  0.5000000       $ Upper bound of variable no.( 2)
y                 $ Any more decision variables (Y or N) ?
TWLAND → (4)      $ Choose a decision variable (1,2,3,...)
  0.2000000E-01   $ Lower bound of variable no.( 4)
  0.3000000       $ Upper bound of variable no.( 4)
y                 $ Any more decision variables (Y or N) ?
HWLAND → (5)      $ Choose a decision variable (1,2,3,...)
  0.2000000       $ Lower bound of variable no.( 5)
   2.000000       $ Upper bound of variable no.( 5)
y                 $ Any more decision variables (Y or N) ?
TFLAND → (6)      $ Choose a decision variable (1,2,3,...)
  0.2000000E-01   $ Lower bound of variable no.( 6)
  0.3000000       $ Upper bound of variable no.( 6)
y                 $ Any more decision variables (Y or N) ?
WFLAND → (7)      $ Choose a decision variable (1,2,3,...)
  0.2000000       $ Lower bound of variable no.( 7)
   2.000000       $ Upper bound of variable no.( 7)
y                 $ Any more decision variables (Y or N) ?
TSKIN → (8)       $ Choose a decision variable (1,2,3,...)
  0.5818000E-01   $ Lower bound of variable no.( 8)
  0.5820000E-01   $ Upper bound of variable no.( 8)
n                 $ Any more decision variables (Y or N) ?
y                 $ Any linked variables (Y or N) ?
ECLAND → (3)      $ Choose a linked variable (1,2,3,...)
TLAND → (2)       $ Choose type of linking (1=polynomial; 2=user-defined)
  0.5000000       $ To which variable is this variable linked?
          1       $ Assign a value to the linking coefficient, C(j)
y                 $ To what power is the decision variable raised?
TSKIN → (8)       $ Any other decision variables in the linking expression?
 -0.5000000       $ To which variable is this variable linked?
          1       $ Assign a value to the linking coefficient, C(j)
n                 $ To what power is the decision variable raised?
n                 $ Any other decision variables in the linking expression?
n                 $ Any constant C0 in the linking expression?
n                 $ Any more linked variables (Y or N) ?
y                 $ Any inequality relations among variables? (type H)
y                 $ Any escape variables (Y or N) ?
                  $ Want to have escape variables chosen by default?
```

Note the tight upper & lower bounds. TSKIN has to be a decision variable because it is involved in a linking expression, but we don't want it to change from it's previously optimized value of 0.05819".

input for DECIDE

Linking expression:

$$ECLAND = TLAND - \frac{TSKIN}{2}$$



Outer surfaces of weld land & skin are flush

Table 15 wscold.OPT

```
n          $ Do you want a tutorial session and tutorial output?
   0       $ Choose an analysis you DON'T want (1, 2,..), IBEHAV
   0       $ NPRINT= output index (0=GOOD, 1=ok, 2=debug, 3=too much)
   1       $ Choose type of analysis (1=opt., 2=fixed, 3=sensit.) ITYPE
   5       $ How many design iterations in this run (3 to 25)?
n          $ Take "shortcuts" for perturbed designs (Y or N)?
   2       $ Choose 1 or 2 or 3 or 4 or 5 for IDESIGN
   1       $ Choose 1 or 2 or 3 or 4 or 5 for move limits, IMOVE
y          $ Do you want default (RATIO=10) for initial move limit jump?
y          $ Do you want the default perturbation (dx/x = 0.05)?
n          $ Do you want to have dx/x modified by GENOPT?
n          $ Do you want to reset total iterations to zero (Type H)?
```

input for MAINSETUP/OPTIMIZE

preferred choice

preferred choice

always use 5. I do.

Table 16 wcold.CPL

```
n       $ Do you want a tutorial session and tutorial output?
n       $ Any design variables to be plotted v. iterations (Y or N)?
n       $ Any design margins to be plotted v. iterations (Y or N)?
n       $ Do you want to get more plots before your next "SUPEROPT"?
```

Input for CHOOSEPLOT
(used after completion of the SUPEROPT computer run)

The optimum design is the best "ALMOST FEASIBLE" or "FEASIBLE" design.
wcold.5.ps = output from chooseplot/diplot

□ weight of the weld land+"extra" edge stringers: WEIGHT (16)

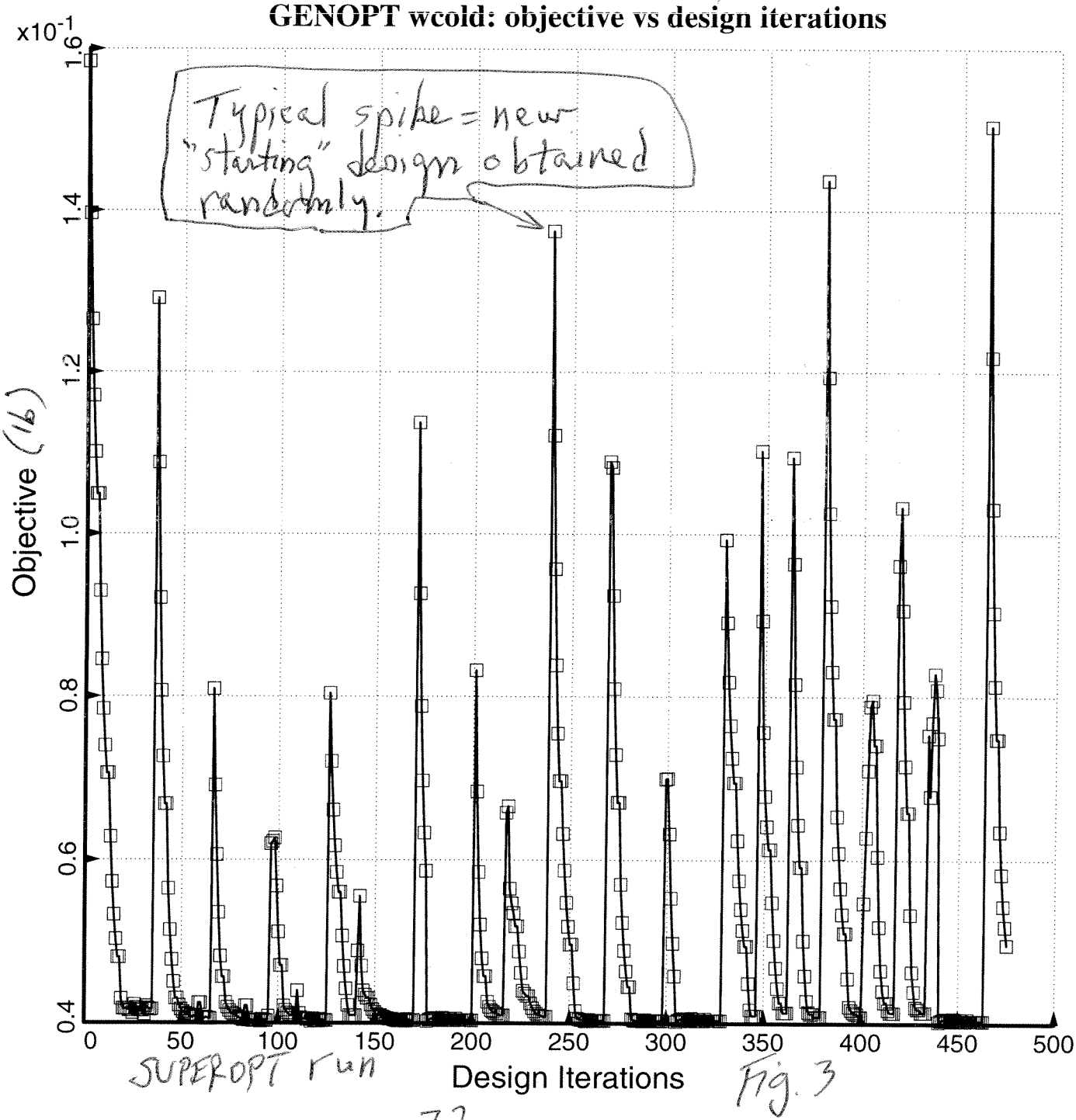


Fig. 3

(3 pages) Table 17 wcold.OPM (output from OPTIMIZE)

```
      n          $ Do you want a tutorial session and tutorial output?
         0       $ Choose an analysis you DON'T want (1, 2,..), IBEHAV
         0       $ NPRINT= output index (0=GOOD, 1=ok, 2=debug, 3=too much)
         2       $ Choose type of analysis (1=opt., 2=fixed, 3=sensit.) ITYPE
         5       $ How many design iterations in this run (3 to 25)?
      n          $ Take "shortcuts" for perturbed designs (Y or N)?
         2       $ Choose 1 or 2 or 3 or 4 or 5 for IDESIGN
         1       $ Choose 1 or 2 or 3 or 4 or 5 for move limits, IMOVE
      y          $ Do you want default (RATIO=10) for initial move limit jump?
      y          $ Do you want the default perturbation (dx/x = 0.05)?
      n          $ Do you want to have dx/x modified by GENOPT?
      n          $ Do you want to reset total iterations to zero (Type H)?

 ************ END OF THE  wcold.OPT  FILE ***********
 ************ MARCH, 2008 VERSION OF GENOPT ****************
 ******** BEGINNING OF THE  wcold.OPM  FILE *********


 ********************** MAIN PROCESSOR **********************
 The purpose of the mainprocessor, OPTIMIZE, is to perform,
 in a batch mode, the work specified by MAINSETUP for the case
 called wcold.  Results are stored in the file  wcold.OPM.
 Please inspect wcold.OPM before doing more design iterations.
 ************************************************************
```

optimum design after 1 superopt

```
 STRUCTURAL ANALYSIS FOR DESIGN ITERATION NO.   0:
0
  STRUCTURAL ANALYSIS WITH UNPERTURBED DECISION VARIABLES
 VAR. DEC. ESCAPE LINK. LINKED  LINKING    LOWER      CURRENT     UPPER       DEFINITION
  NO. VAR.  VAR.  VAR.    TO   CONSTANT    BOUND      VALUE      BOUND
   1   Y     N     N      0    0.00E+00  4.00E+00  4.0000E+00  6.00E+00  width of the weld land: »
WLAND
   2   Y     N     N      0    0.00E+00  1.00E-01  1.0000E-01  5.00E-01  thickness of the weld la»
nd: TLAND
   3   N     N     Y      2    5.00E-01  0.00E+00  2.0910E-02  0.00E+00  eccentricity of the weld»
 land: ECLAND
   4   Y     N     N      0    0.00E+00  2.00E-02  6.2756E-02  3.00E-01  web thickness of the wel»
d land edge stringer: TWLAND
   5   Y     N     N      0    0.00E+00  2.00E-01  1.6156E+00  2.00E+00  height of the web of the»
 weld land edge stringer: HWLAND
   6   Y     N     N      0    0.00E+00  2.00E-02  2.0600E-02  3.00E-01  thickness of the outstan»
ding flange of the stringer: TFLA
   7   Y     N     N      0    0.00E+00  2.00E-01  6.0845E-01  2.00E+00  width of the outstanding»
 flange of the stringer: WFLAND
   8   Y     Y     N      0    0.00E+00  5.82E-02  5.8180E-02  5.82E-02  thickness of the cylindr»
ical shell skin: TSKIN

 BIGBOSOR4 input file for: general buckling load, no weld lands
wcold.BEHX0

 BIGBOSOR4 input file for: general buckling load
wcold.BEHX1

 GENERAL BUCKLING LOAD FACTORS AND MODES (BEHX1)
        2.2037E+00( 100)
        1.9667E+00( 200)
        2.2288E+00( 300)
        2.7064E+00( 400)
        3.3894E+00( 500)
        3.1335E+00( 600)
        2.6331E+00( 700)
        2.3567E+00( 800)
        2.2276E+00( 900)
        2.1950E+00(1000)
 Critical buckling load factor, GENBUK=  1.9667E+00
 Critical number of circumferential waves, NWVCRT=  200

 BIGBOSOR4 input file for: "panel" buckling load
wcold.BEHX2

 "PANEL" BUCKLING LOAD FACTORS AND MODES (BEHX2)
        2.0487E+00( 100)
```

← critical value. (200) means 2 axial half waves over the length of the cylindrical shell. See Appendix 2

← inter-ring buckling

Table 17 (p. 2 of 3)

```
        1.4957E+00( 200)
        1.5398E+00( 300)
        1.4726E+00( 400)
        1.5132E+00( 500)
        1.6456E+00( 600)
        1.8492E+00( 700)
        2.1124E+00( 800)
        2.4276E+00( 900)
        2.7812E+00(1000)
Critical buckling load factor, PANBUK=  1.4726E+00
Critical number of circumferential waves, NWVCRT=  400

***** RESULTS FOR LOAD SET NO.  1  ******
PARAMETERS WHICH DESCRIBE BEHAVIOR (e.g. stress, buckling load)

BEH.   CURRENT
NO.      VALUE            DEFINITION
 1    1.967E+00    general buckling load: GENBUK(1 )
 2    1.473E+00    "panel" buckling: PANBUK(1 )
 3    2.331E+04    weld land effective stress: STRESS(1 )

******* NOTE ******* NOTE ******* NOTE ****** NOTE ******
The phrase, "NOT APPLY", for MARGIN VALUE means that that
particular margin value is exactly zero.
*** END NOTE *** END NOTE *** END NOTE *** END NOTE *****


***** RESULTS FOR LOAD SET NO.  1  ******
MARGINS CORRESPONDING TO CURRENT DESIGN (F.S.= FACTOR OF SAFETY)

MARGIN CURRENT
NO.      VALUE            DEFINITION
 1   -1.663E-02  (GENBUK(1 )/GENBUKA(1 )) / GENBUKF(1 )-1; F.S.=  2.00
 2   -1.829E-02  (PANBUK(1 )/PANBUKA(1 )) / PANBUKF(1 )-1; F.S.=  1.50
 3    2.003E+00  (STRESSA(1 )/STRESS(1 )) / STRESSF(1 )-1; F.S.=  1.00
************************************************
******************** DESIGN OBJECTIVE ********************
*************                                  ***********
   CURRENT VALUE OF THE OBJECTIVE FUNCTION:

VAR.  CURRENT
NO.     VALUE            DEFINITION
 1   4.023E-02  weight of the weld land+"extra" edge stringer: WEIGHT
*************                                  ***********
******************** DESIGN OBJECTIVE ********************
************************************************************
***********  ALL 1 LOAD CASES PROCESSED ***********
************************************************


PARAMETERS WHICH ARE ALWAYS FIXED.  NONE CAN BE DECISION VARIAB.

VAR.   CURRENT
NO.      VALUE            DEFINITION
 1    4.800E+01  radius of the cylindrical shell: RADCYL
 2    6.875E+01  length of the cylindrical shell: LENCYL
 3    2.277E+00  stringer spacing: BSTR
 4    8.646E-02  stringer thickness: TSTR
 5    9.898E-01  stringer height: HSTR
 6    1.397E+01  ring spacing: BRNG
 7    1.858E-01  ring thickness: TRNG
 8    9.717E-01  ring height: HRNG
 9    1.100E+07  Young's modulus: EMOD
10    3.000E-01  Poisson ratio: NU
11    2.500E-04  mass density (aluminum = 0.00025 in English units): DENS
PARAMETERS WHICH ARE ENVIRONMENTAL FACTORS (e.g. loads, temps.)

VAR.   CURRENT
NO.      VALUE            DEFINITION
 1    6.692E+05  total axial load (2*pi*r*Nx): PX(1 )
 2    0.000E+00  uniform normal pressure: PY(1 )
 3    0.000E+00  axial load in Load Set B: PX0(1 )
 4    0.000E+00  pressure in Load Set B: PY0(1 )
PARAMETERS WHICH ARE CLASSIFIED AS ALLOWABLES (e.g. max. stress)

VAR.   CURRENT
```

← critical value. (400) means 4 axial half waves between rings. See Appendix 2

Factors of safety are > 1.0 because the shell is perfect in this BIGBOSOR4 model.

one "extra" edge stringer. (The definition of WEIGHT should be changed. "stringers" should be replaced by "stringer")

Table 17 (p. 3 of 3)

```
NO.      VALUE            DEFINITION
 1     1.000E+00  allowable for general buckling: GENBUKA(1 )
 2     1.000E+00  allowable for "panel" buckling: PANBUKA(1 )
 3     7.000E+04  maximum allowable effective stress: STRESSA(1 )
PARAMETERS WHICH ARE FACTORS OF SAFETY

VAR.    CURRENT
NO.      VALUE            DEFINITION
 1     2.000E+00  factor of safety for general buckling: GENBUKF(1 )
 2     1.500E+00  factor of safety for "panel" buckling: PANBUKF(1 )
 3     1.000E+00  factor of safety for effective stress: STRESSF(1 )

   0 INEQUALITY CONSTRAINTS WHICH MUST BE SATISFIED


DESCRIPTION OF FILES USED AND GENERATED IN THIS RUN:

wcold.NAM = This file contains only the name of the case.
wcold.OPM = Output data. Please list this file and inspect
            carefully before proceeding.
wcold.OPP = Output file containing evolution of design and
            margins since the beginning of optimization cycles.
wcold.CBL = Labelled common blocks for analysis.
            (This is an unformatted sequential file.)
wcold.OPT = This file contains the input data for MAINSETUP
             as well as OPTIMIZE. The batch command OPTIMIZE
             can be given over and over again without having
             to return to MAINSETUP because wcold.OPT exists.
URPROMPT.DAT= Prompt file for interactive input.

For further information about files used and generated
during operation of GENOPT, give the command HELPG FILES.

Menu of commands: CHOOSEPLOT, OPTIMIZE, MAINSETUP, CHANGE,
                  DECIDE, SUPEROPT

IN ORDER TO AVOID FALSE CONVERGENCE OF THE DESIGN, BE SURE TO
RUN  "OPTIMIZE"  MANY TIMES DURING AN OPTIMIZATION AND/OR USE
THE "GLOBAL" OPTIMIZING SCRIPT, "SUPEROPT".
********************* END OF wcold.OPM FILE *****************
```

Table 18 wcold. CHG

```
    n          $ Do you want a tutorial session and tutorial output?
    y          $ Do you want to change any values in Parameter Set No. 1?
       1       $ Number of parameter to change (1, 2, 3, . .)
 4.000000      $ New value of the parameter
    y          $ Want to change any other parameters in this set?
       2       $ Number of parameter to change (1, 2, 3, . .)
0.1000000      $ New value of the parameter
    y          $ Want to change any other parameters in this set?
       3       $ Number of parameter to change (1, 2, 3, . .)
0.2091000E-01  $ New value of the parameter
    y          $ Want to change any other parameters in this set?
       4       $ Number of parameter to change (1, 2, 3, . .)
0.6275600E-01  $ New value of the parameter
    y          $ Want to change any other parameters in this set?
       5       $ Number of parameter to change (1, 2, 3, . .)
 1.615600      $ New value of the parameter
    y          $ Want to change any other parameters in this set?
       6       $ Number of parameter to change (1, 2, 3, . .)
0.2060000E-01  $ New value of the parameter
    y          $ Want to change any other parameters in this set?
       7       $ Number of parameter to change (1, 2, 3, . .)
0.6084500      $ New value of the parameter
    y          $ Want to change any other parameters in this set?
       8       $ Number of parameter to change (1, 2, 3, . .)
0.5818000E-01  $ New value of the parameter
    n          $ Want to change any other parameters in this set?
    n          $ Do you want to change values of any "fixed" parameters?
    n          $ Do you want to change any loads?
    n          $ Do you want to change values of allowables?
    n          $ Do you want to change any factors of safety?
```

input for CHANGE

Note: "CHANGE" is used as a device to save the optimum design.

The user should always do this!

Table 19 wscold. BEHXO

```
general buckling, no weld land (INDIC=4)
4, 1, 0, 0,   1 $ INDIC,NPRT,ISTRESS,IPRE,NSEG
H   $ Segment number      1     1     1     1
 91, 3,    2 $ NMESH,NTYPEH,NSHAPE
 2.154444E+03  0.000000E+00 $ R1,Z1
 2.154444E+03  9.600000E+01 $ R2,Z2
 2.154444E+03  4.800000E+01, -1.  $ RC,ZC,SUR
0, 3,   2.909000E-02  $ IMP,NTYPEZ,ZVAL
N  $ do not print r(s), etc.
0, 0., 2, 0, 2  $ NRINGS,K,NSTRES,NRLOAD,NTYPE
 0.000000E+00  9.600000E+01 $ Z1,Z2
0., 0.  $ FN10(1), FN10(2)
-2.219000E+03 -2.219000E+03 $ FN20(1),FN20(2)
N  $ do not print prestresses
2,   1.100000E+07  3.000000E-01  2.500000E-04 $ NWALL,EMOD,NU,DENS
0.  $ ALPHA
 1, 0  $ NRS,NSUR
N  $ do not print ref.surf.location
Y  $ are there stringers
    968, 0  $ N1,K1
 1.100000E+07  3.000000E-01  2.500000E-04 $ EMOD,NU,DENS
Y  $ is stringer cross section constant
Y  $ is stringer cross section rectangu
 1.857600E-01  9.717300E-01 $ TRNG,HRNG
Y  $ are there rings
0  $ K2
 1.100000E+07  3.000000E-01  2.500000E-04 $ EMOD,NU,DENS
Y  $ is ring cross section constant
Y  $ is ring cross section rectangular
 2.277300E+00  $ BSTR
 8.646200E-02  9.897700E-01 $ TSTR,HSTR
N  $ do not print C(i,j)
N  $ do not print distributed loads
H  $ GLOBAL DATA BEGINS...
0  $ NLAST
N  $ any expanded plots?
  100   100  1000   100     1  $ N0B,NMINB,NMAXB,INCRB,NVEC
H  $ CONSTRAINT CONDITIONS FOLLOW....
    1  $ how many segments?
H  $ CONSTRAINT CONDITIONS FOR SEGMENT      1     1     1     1
0, 2, 1, 0, 0, 1, 1 $ poles,ground,etc.
0., 0.  $ D1,D2
Y  $ same for preb. & buck.
91, 0, 0, 1, 1 $ INODE,etc.
0., 0.  $ D1,D2
Y  $ same for preb. & buck.
N  $ joined to lower segs?
N  $ rigid body motion possible?
Y  $ do you want output for seg.?
N  $ descrete ring forces?
```

Created by BOSDEC with INDX=0
(See Subroutine BEHX1 on
pp. 6-8 of Table 10)

Valid input for BIGBOSOR4: model
with no weld lands.

Table 20 wcold, ALL (after "cleanup") (2 pages)

```
general buckling, no weld land (INDIC=4)
        4       $ INDIC = analysis type indicator
        1       $ NPRT = output options (1=minimum, 2=medium, 3=maximum)
        0       $ ISTRES= output control (0=resultants, 1=sigma, 2=epsilon)
        0       $ IPRE  = indicator for prebuckling stress calculation (0 or 1)
        1       $ NSEG  = number of shell segments (less than 195)
     H          $
     H          $ SEGMENT NUMBER    1    1    1    1    1    1    1    1
     H          $ NODAL POINT DISTRIBUTION FOLLOWS...
       91       $ NMESH = number of node points (5 = min.; 98 = max.)( 1)
        3       $ NTYPEH= control integer (1 or 3) for nodal point spacing
     H          $ REFERENCE SURFACE GEOMETRY FOLLOWS...
        2       $ NSHAPE= indicator (1,2 or 4) for geometry of meridian
  2154.444      $ R1      = radius at beginning of segment (see p. 66)
  0.000000      $ Z1      = global axial coordinate at beginning of segment
  2154.444      $ R2      = radius at end of segment
  96.00000      $ Z2      = global axial coordinate at end of segment
  2154.444      $ RC      = radius from axis of rev. to center of curvature
  48.00000      $ ZC      = axial coordinate of center of curvature
 -1.000000      $ SROT=indicator for direction of increasing arc (-1. or +1.)
     H          $ IMPERFECTION SHAPE FOLLOWS...
        0       $ IMP   = indicator for imperfection (0=none, 1=some)
     H          $ REFERENCE SURFACE LOCATION RELATIVE TO WALL
        3       $ NTYPEZ= control (1 or 3) for reference surface location
 0.2909000E-01  $ ZVAL  = distance from leftmost surf. to reference surf.
     N          $ Do you want to print out r(s), r'(s), etc. for this segment?
     H          $ DISCRETE RING INPUT FOLLOWS...
        0       $ NRINGS= number (max=20) of discrete rings in this segment
  0.000000      $ K=elastic foundation modulus (e.g. lb/in**3)in this seg.
     H          $ PREBUCKLING RESULTANTS INPUT FOLLOWS...
        2       $ NSTRES = number of meridional stations for Nx, Ny callouts
        0       $ NRLOAD = number of preloaded discrete rings in entire shell
        2       $ NTYPE = control for meaning of loading callout (2=z, 3=r)
  0.000000      $ Z(I)  = axial coordinate of Ith loading callout, z( 1)
  96.00000      $ Z(I)  = axial coordinate of Ith loading callout, z( 2)
  0.000000      $ FN10  = meridional prestress at Ith callout, FN10( 1)
  0.000000      $ FN10  = meridional prestress at Ith callout, FN10( 2)
 -2219.000      $ FN20  = circumferential prestress at Ith callout, FN20( 1)
 -2219.000      $ FN20  = circumferential prestress at Ith callout, FN20( 2)
     N          $ Do you want to print out prestresses at meridional stations?
     H          $ SHELL WALL CONSTRUCTION FOLLOWS...
        2       $ NWALL=index (1, 2, 4, 5, 6, 7, 8, 9, 10) for wall construction
 0.1100000E+08  $ E     = Young's modulus for skin
 0.3000000      $ U     = Poisson's ratio for skin
 0.2500000E-03  $ SM =mass density of skin (e.g. alum.=.00025 lb-sec**2/in**4)
  0.000000      $ ALPHA = coefficient of thermal expansion
        1       $ NRS   = control (0 or 1) for addition of smeared stiffeners
        0       $ NSUR    = control for thickness input (0 or 1 or -1)
     N          $ Do you want to print out ref. surf. location and thickness?
     H          $ SMEARED STIFFENER INPUT FOLLOWS...
     Y          $ Are there stringers or isogrid (please answer Y or N)?
      968       $ N1    = number of stringers in 360 degrees
        0       $ K1 =control (0 or 1) for internal or external stringers
 0.1100000E+08  $ E1    = stringer modulus
 0.3000000      $ U1    = stringer Poisson ratio
 0.2500000E-03  $ STIFMD= stringer mass density
     Y          $ Is the stringer cross section constant in this segment?
     Y          $ Is the stringer cross section rectangular (Y or N)?
 0.1857600      $ T1    = thickness of stringer (constant)
 0.9717301      $ H1    = height of stringer (constant)
     Y          $ Are there rings (please answer Y or N)?
        0       $ K2  =control (0 or 1) for internal or external rings
 0.1100000E+08  $ E2    = ring modulus
 0.3000000      $ U2    = ring Poisson ratio
 0.2500000E-03  $ RGMD= ring mass density
     Y          $ Is the ring cross section constant in this segment?
     Y          $ Is the ring cross section rectangular (Y or N)?
  2.277300      $ D2    = arc length between adjacent rings (constant)
 0.8646200E-01  $ T2    = thickness of ring (constant)
 0.9897701      $ H2    = height of ring (constant)
     N          $ Do you want to print out the C(i,j) at meridional stations?
     N          $ Do you want to print out distributed loads along meridian?
     H          $
     H          $ GLOBAL DATA BEGINS...
        0       $ NLAST = plot options  (-1=none, 0=geometry, 1=u,v,w)
     N          $ Are there any regions for which you want expanded plots?
      100       $ N0B   = starting number of circ. waves (buckling analysis)
      100       $ NMINB = minimum number of circ. waves  (buckling analysis)
```

Table 20 (p. 2-92)

```
    1000        $ NMAXB = maximum number of circ. waves  (buckling analysis)
     100        $ INCRB = increment in number of circ. waves (buckling)
       1        $ NVEC  = number of eigenvalues for each wave number
       H        $ CONSTRAINT CONDITIONS FOLLOW....
       1        $ How many segments in the structure?
       H        $
       H        $ CONSTRAINT CONDITIONS FOR SEGMENT NO.    1    1    1    1
       H        $ POLES INPUT FOLLOWS...
       0        $ Number of poles (places where r=0) in SEGMENT( 1)
       H        $ INPUT FOR CONSTRAINTS TO GROUND FOLLOWS...
       2        $ At how many stations is this segment constrained to ground?
       1        $ INODE = nodal point number of constraint to ground, INODE( 1)
       0        $ IUSTAR=axial displacement constraint (0 or 1 or 2)
       0        $ IVSTAR=circumferential displacement(0=free,1=0,2=imposed)
       1        $ IWSTAR=radial displacement(0=free,1=constrained,2=imposed)
       1        $ ICHI=meridional rotation (0=free,1=constrained,2=imposed)
0.000000        $ D1     = radial component of offset of ground support
0.000000        $ D2     = axial component of offset of ground support
       Y        $ Is this constraint the same for both prebuckling and buckling?
      91        $ INODE = nodal point number of constraint to ground, INODE( 2)
       0        $ IUSTAR=axial displacement constraint (0 or 1 or 2)
       0        $ IVSTAR=circumferential displacement(0=free,1=0,2=imposed)
       1        $ IWSTAR=radial displacement(0=free,1=constrained,2=imposed)
       1        $ ICHI=meridional rotation (0=free,1=constrained,2=imposed)
0.000000        $ D1     = radial component of offset of ground support
0.000000        $ D2     = axial component of offset of ground support
       Y        $ Is this constraint the same for both prebuckling and buckling?
       H        $ JUNCTION CONDITION INPUT FOLLOWS...
       N        $ Is this segment joined to any lower-numbered segments?
       H        $ RIGID BODY CONSTRAINT INPUT FOLLOWS...
       N        $ Given existing constraints, are rigid body modes possible?
       H        $ "GLOBAL3" QUESTIONS (AT END OF CASE)...
       Y        $ Do you want to list output for segment( 1)
       N        $ Do you want to list forces in the discrete rings, if any?
```

properly annotated version of wcold.BEHXO, which is automatically generated from wcold.BEHXO by "cleanup".

Valid input file for BIGBOSOR4.
(no weld lands)

Table 21 wcold.OUT (abridged)

```
wcold.behx0.out
no weld lands   (General buckling, no weld lands)
 ***** EIGENVALUES AND MODE SHAPES *****
  EIGENVALUE(CIRC. WAVES)
 ========================================
    2.9177E+00( 100)
    2.2982E+00( 200) <--very close to critical value
    2.2942E+00( 300) <--critical value. 300 means 3 axial halfwaves. Fig 4
    2.6675E+00( 400)                                  (See Appendix 2)
    3.3583E+00( 500)
    4.3005E+00( 600)
    5.4743E+00( 700)
    6.8507E+00( 800)
    8.4329E+00( 900)
    1.0208E+01(1000)
 ========================================
```

Output from BIGBOSOR4 for the shell with no weld lands. (Corresponding to the input data listed in Tables 19 & 20).

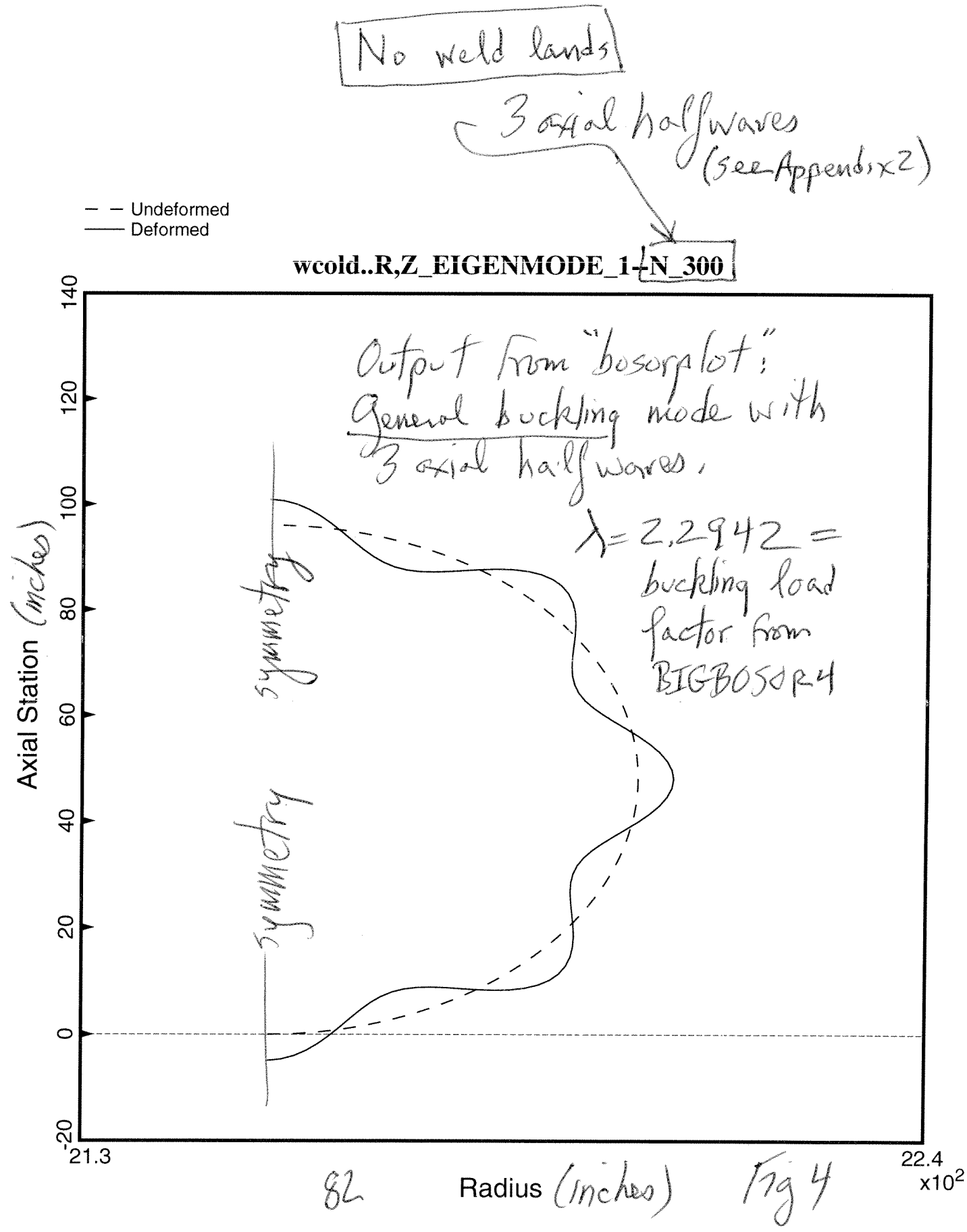

No weld lands
3 axial halfwaves (see Appendix 2)
Undeformed
Deformed
wcold..R,Z_EIGENMODE_1-N_300
Output from "bosorplot":
General buckling mode with 3 axial halfwaves.
λ = 2.2942 = buckling load factor from BIGBOSOR4
symmetry
symmetry
Axial Station (inches)
140
120
100
80
60
40
20
0
-20
21.3
22.4
x10^2
Radius (inches)

Fig 4

Table 22 wcold.OUT (abridged)

wcold.behx1.out (General buckling with weld lands)
with weld lands spaced at 120 degree interval
shell acreage rectangular stringers & rings are smeared out
***** EIGENVALUES AND MODE SHAPES *****
EIGENVALUE(CIRC. WAVES)

```
=====================================
   2.2037E+00( 100)
   1.9667E+00( 200) <--general buckling
   2.2288E+00( 300)
   2.7064E+00( 400)
   3.3895E+00( 500)
   3.1335E+00( 600)
   2.6328E+00( 700)
   2.3563E+00( 800)
   2.2269E+00( 900)
   2.1944E+00(1000) <--sidesway of "edge" stringers
   2.2208E+00(1100)
   2.3039E+00(1200)
   2.4178E+00(1300)
   2.5405E+00(1400)
   2.5524E+00(1500)
   2.4029E+00(1600)
   2.2762E+00(1700)
   2.1733E+00(1800)
   2.0907E+00(1900)
   2.0245E+00(2000)
   1.9704E+00(2100)
   1.9291E+00(2200)
   1.8990E+00(2300)
   1.8785E+00(2400)
   1.8667E+00(2500)
   1.8624E+00(2600) <--local buckling of weld land
   1.8649E+00(2700)
   1.8735E+00(2800)
   1.8877E+00(2900)
   1.9071E+00(3000)
=====================================
```

Fig. 5 (2 axial halfwaves)

Fig 6 (10 axial halfwaves)

Fig. 7 (26 axial halfwaves)

Output from BIGBOSOR4 for the shell with Tee-stiffened weld lands every 120 degrees.

General buckling.

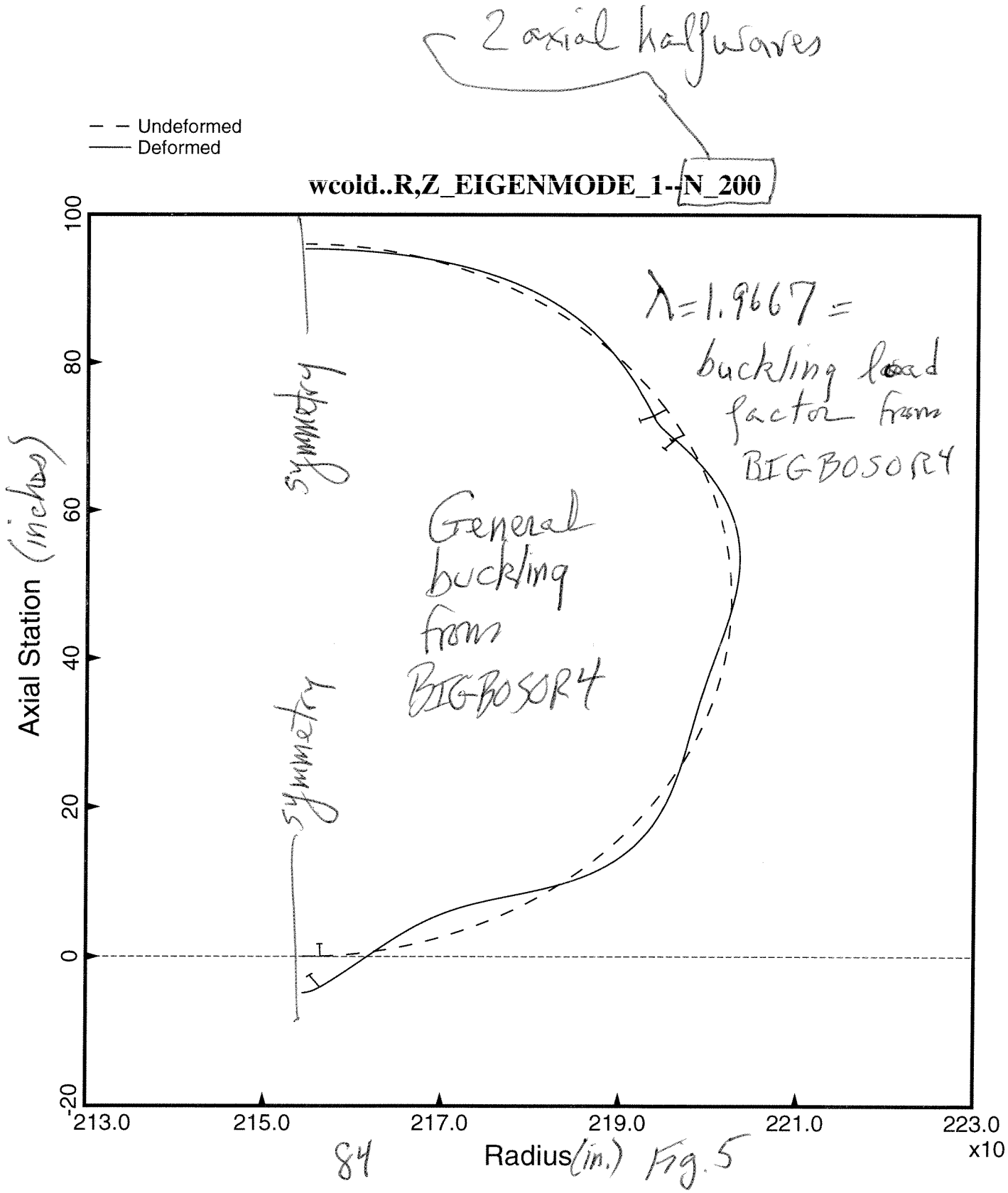

2 axial halfwaves
Undeformed
Deformed
wcold..R,Z_EIGENMODE_1--N_200
λ=1.9667 = buckling load factor from BIGBOSOR4
General buckling from BIGBOSOR4
symmetry
symmetry
Axial Station (inches)
100
80
60
40
20
0
-20
213.0
215.0
217.0
219.0
221.0
223.0
x10
Radius (in.)

Fig. 5

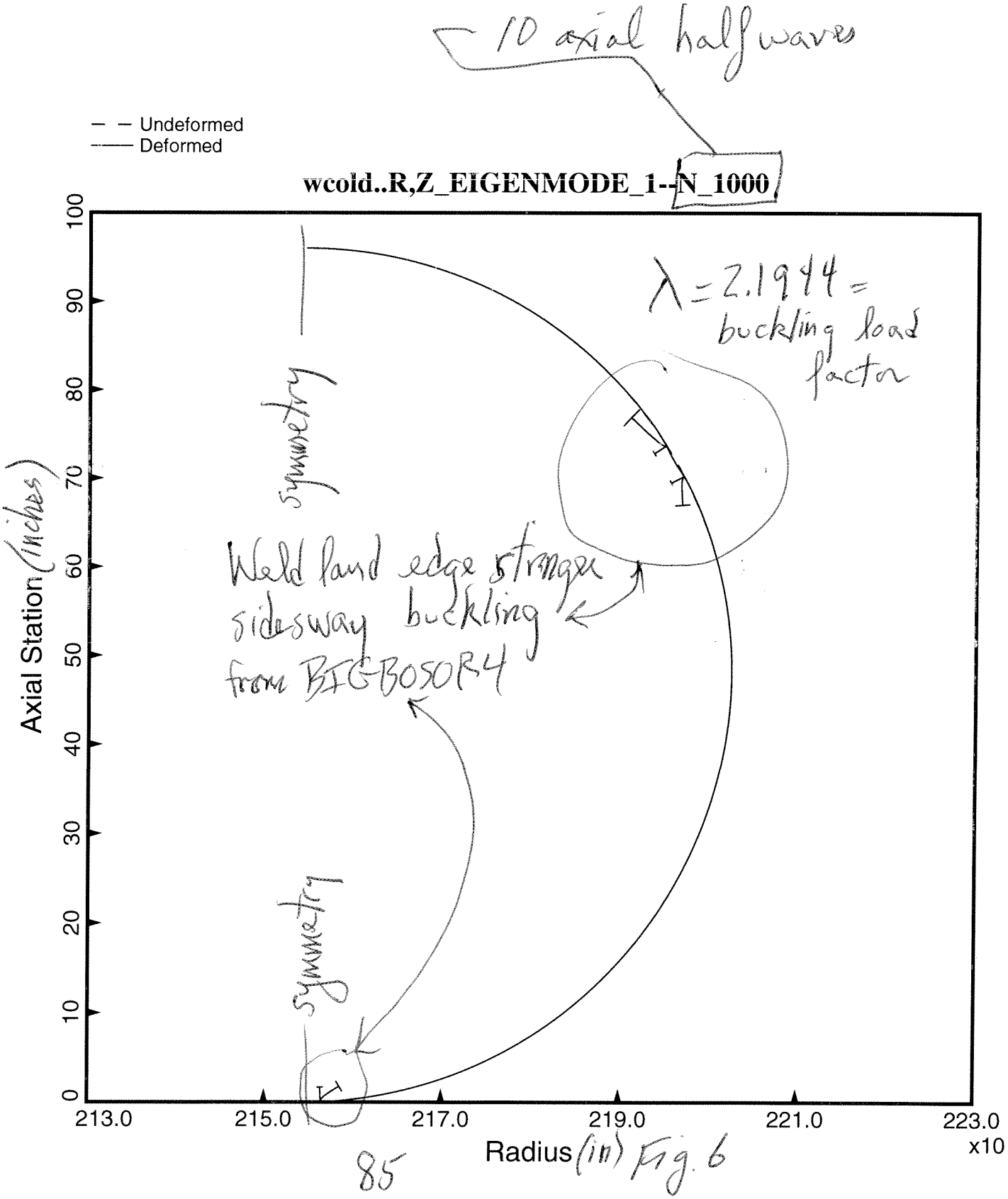

10 axial halfwaves
— — Undeformed
—— Deformed
wcold..R,Z_EIGENMODE_1--N_1000
λ = 2.1944 = buckling load factor
symmetry
Weld land edge stringer sidesway buckling from BIGBOSOR4
symmetry
Axial Station (inches)
0
10
20
30
40
50
60
70
80
90
100
213.0
215.0
217.0
219.0
221.0
223.0
x10
Radius (in)

Fig. 6

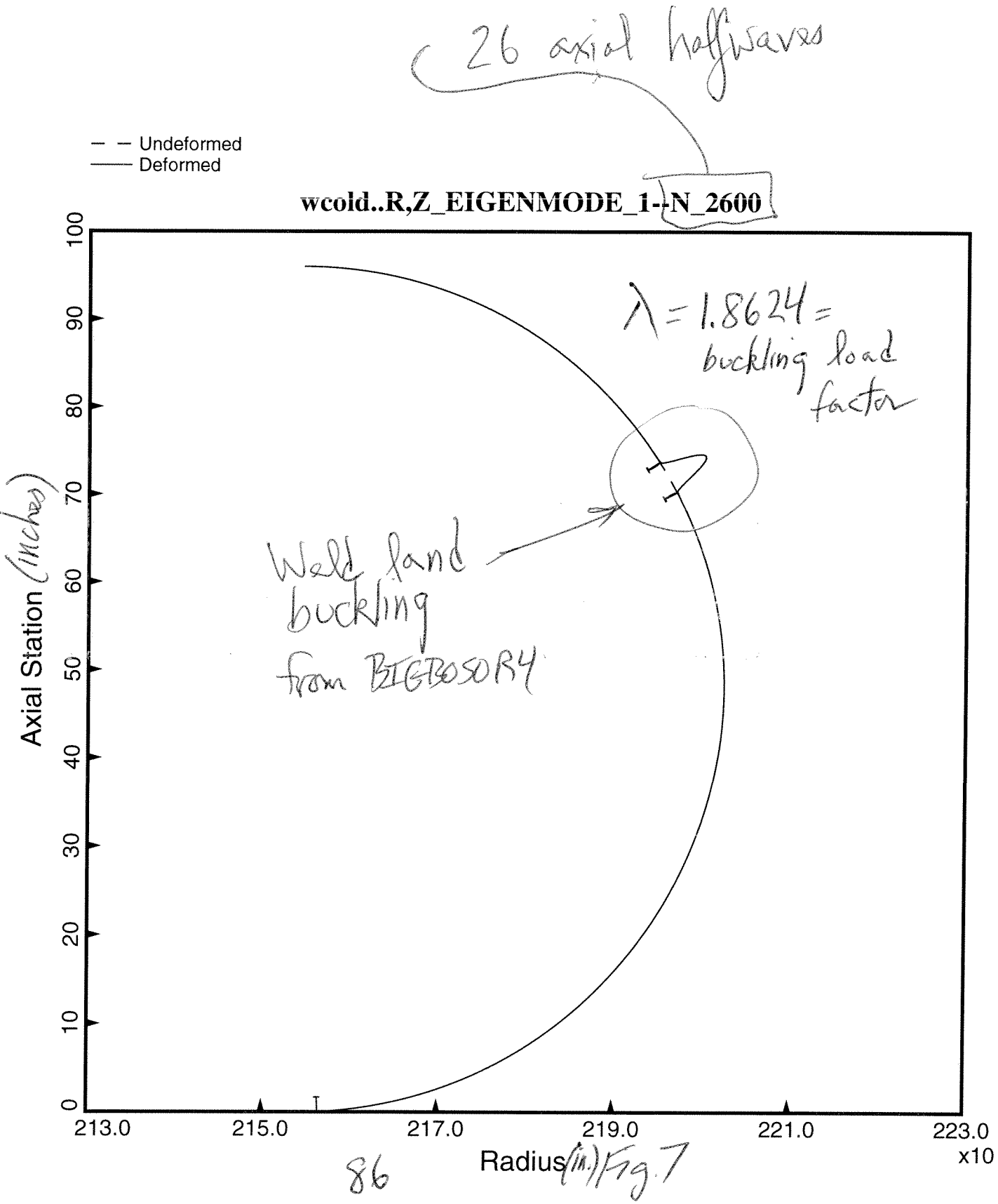

26 axial halfwaves
– – Undeformed
—— Deformed
wcold..R,Z_EIGENMODE_1--N_2600
λ = 1.8624 = buckling load factor
Weld land buckling from BIGBOSOR4
Axial Station (inches)
0
10
20
30
40
50
60
70
80
90
100
213.0
215.0
217.0
219.0
221.0
223.0
x10
Radius (in.)

Fig. 7

Table 23 wcold.OUT (abridged)
inter-ring buckling

```
wcold.behx2.out
buckling between adjacent rings
shell acreage rectangular stringers are smeared out
 ***** EIGENVALUES AND MODE SHAPES *****
  EIGENVALUE(CIRC. WAVES)
 ========================================
    2.0487E+00( 100)
    1.4958E+00( 200) <--close to critical value
    1.5398E+00( 300)
    1.4726E+00( 400) <--critical value
    1.5132E+00( 500)
    1.6456E+00( 600)
    1.8492E+00( 700)
    2.1124E+00( 800)
    2.4276E+00( 900)
    2.7812E+00(1000)
 ========================================
```

, inter-ring buckling

Output from BIGBOSOR4
for inter-ring buckling

Inter-ring buckling

2 axial halfwaves between rings

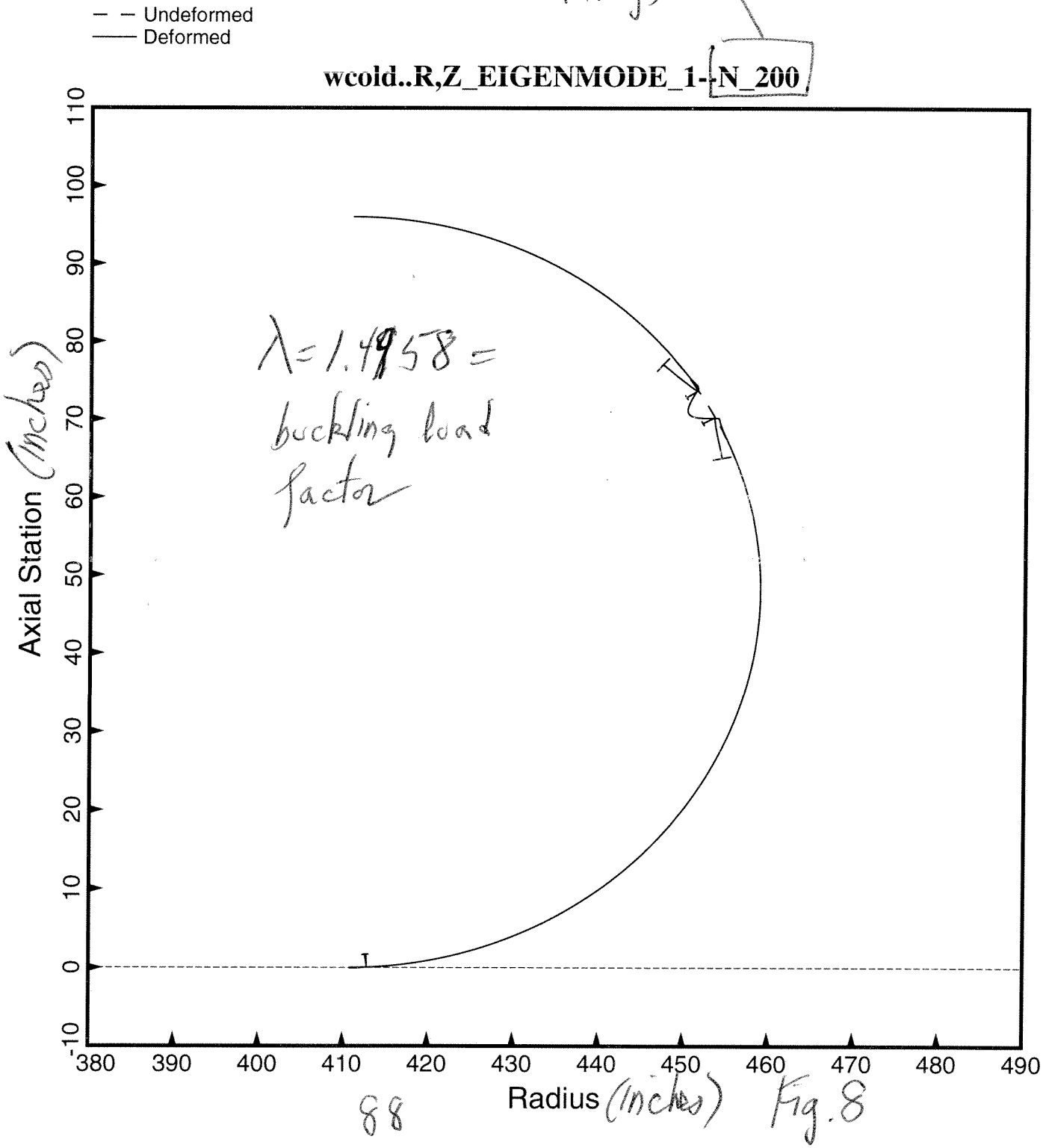


Fig. 8

Inter-ring buckling

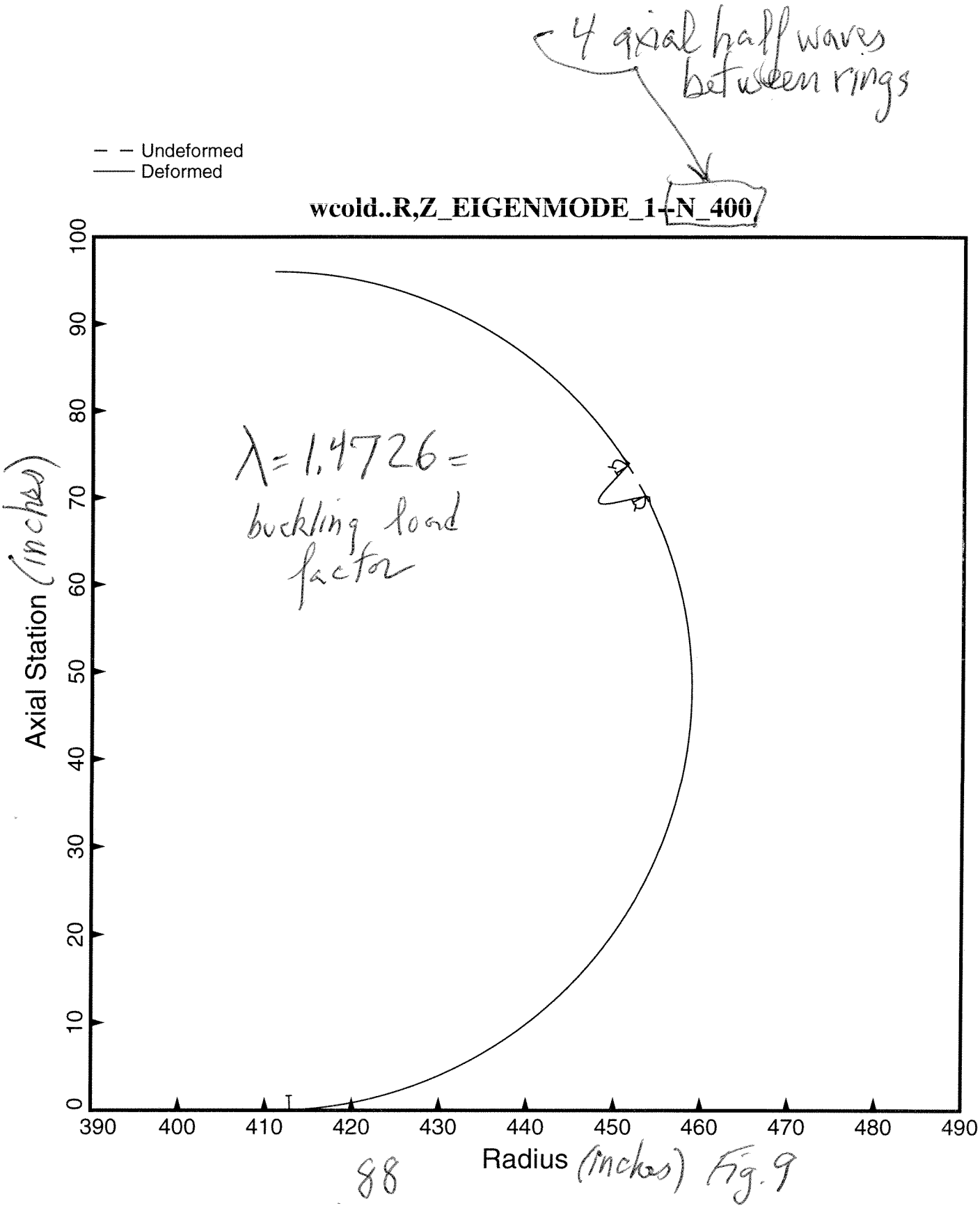
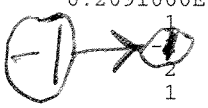


Fig. 9

Table 24 nasacoldbend.PAN

```
n             $ Do you want a tutorial session and tutorial output?
 75.398       $ Panel length in the plane of the screen, L2
   1          $ Enter control ILAND for weldland (0=none or 1=weldland)
   3          $ Number of BOSOR4-type segments in the weld land, KLAND
 4.000000     $ Width of the weld land, WLAND
0.1000000     $ Thickness of the weld land, TLAND
0.6275600E-01 $ Thickness of "extra" weld land edge stringer web, TWLAND
 1.615600     $ Height of "extra" weld land stringer web, HWLAND
0.2060000E-01 $ Thickness of outstanding flange of weld land stringer, TFLAND
0.6084500     $ Width of outstanding flange of weld land stringer, WFLAND
0.2091000E-01 $ Eccentricity of the weld land outer surface, ECLAND
   1          $ Enter control (1=sym; 2=s.s.) for boundary condition
  -1          $ Enter ILOCAL=0 or 1 or -1 or -2 (Type (H)elp), ILOCAL
   2          $ Number of halfwaves in the axial direction [see H(elp)],NWAVE
   1          $ How many eigenvalues (get at least 3) do you want?
```



input for "panel3"

ILOCAL = -1 for acreage stringers modeled as shell branches

optimized weld land variables from "wcold." (See Table 17, p.1 of Table 17)

"panel3" creates a valid input file for BIGBOSOR4. The shell acreage stringers are modeled as shell branches & the shell acreage rings are smeared out.

Table 25

nasacoldbend.OUT (abridged) (Stringers ~~are~~ in the shell acreage are little shell branches) (rings are smeared out)

```
 ***** EIGENVALUES AND MODE SHAPES *****
  EIGENVALUE(CIRC. WAVES)
 ========================================
    2.1877E+00( 100)
    1.9615E+00( 200) <---general buckling (Fig. 10)
    2.1203E+00( 300)
    2.5287E+00( 400)
    3.1218E+00( 500)
    3.0497E+00( 600)
    2.5650E+00( 700)
    2.2956E+00( 800)
    2.1691E+00( 900)
    2.1384E+00(1000) <---~~ring~~ sidesway (Fig. 11)
    2.1743E+00(1100)
    2.2574E+00(1200)
    2.3734E+00(1300)
    2.5049E+00(1400)
    2.4933E+00(1500)
    2.3483E+00(1600)
    2.2242E+00(1700)
    2.1232E+00(1800)
    2.0422E+00(1900)
    1.9782E+00(2000)
    1.9286E+00(2100)
    1.8915E+00(2200)
    1.8651E+00(2300)
    1.8481E+00(2400)
    1.8394E+00(2500)
    1.8381E+00(2600) <---weld land buckling (Fig. 12)
    1.8434E+00(2700)
    1.8547E+00(2800)
    1.8714E+00(2900)
    1.8931E+00(3000)
 ========================================
```

Output from BIGBOSOR4

Compare with Table 22

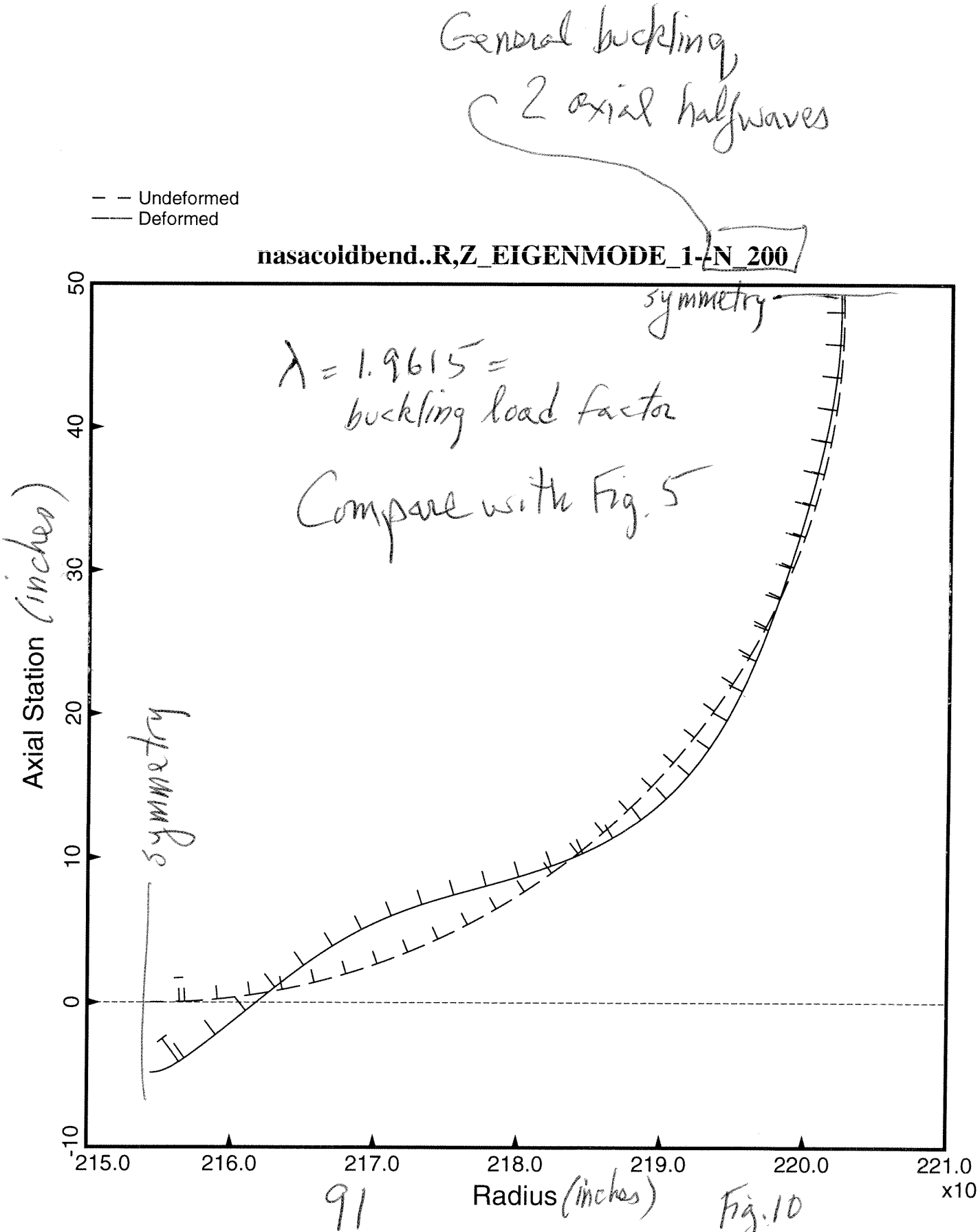

General buckling
2 axial halfwaves
– – Undeformed
—— Deformed
nasacoldbend..R,Z_EIGENMODE_1-EN_200
symmetry
λ = 1.9615 =
buckling load factor
Compare with Fig. 5
symmetry
Axial Station (inches)
50
40
30
20
10
0
-10
215.0
216.0
217.0
218.0
219.0
220.0
221.0
x10
Radius (inches)

Fig. 10

General buckling model

10 axial halfwaves

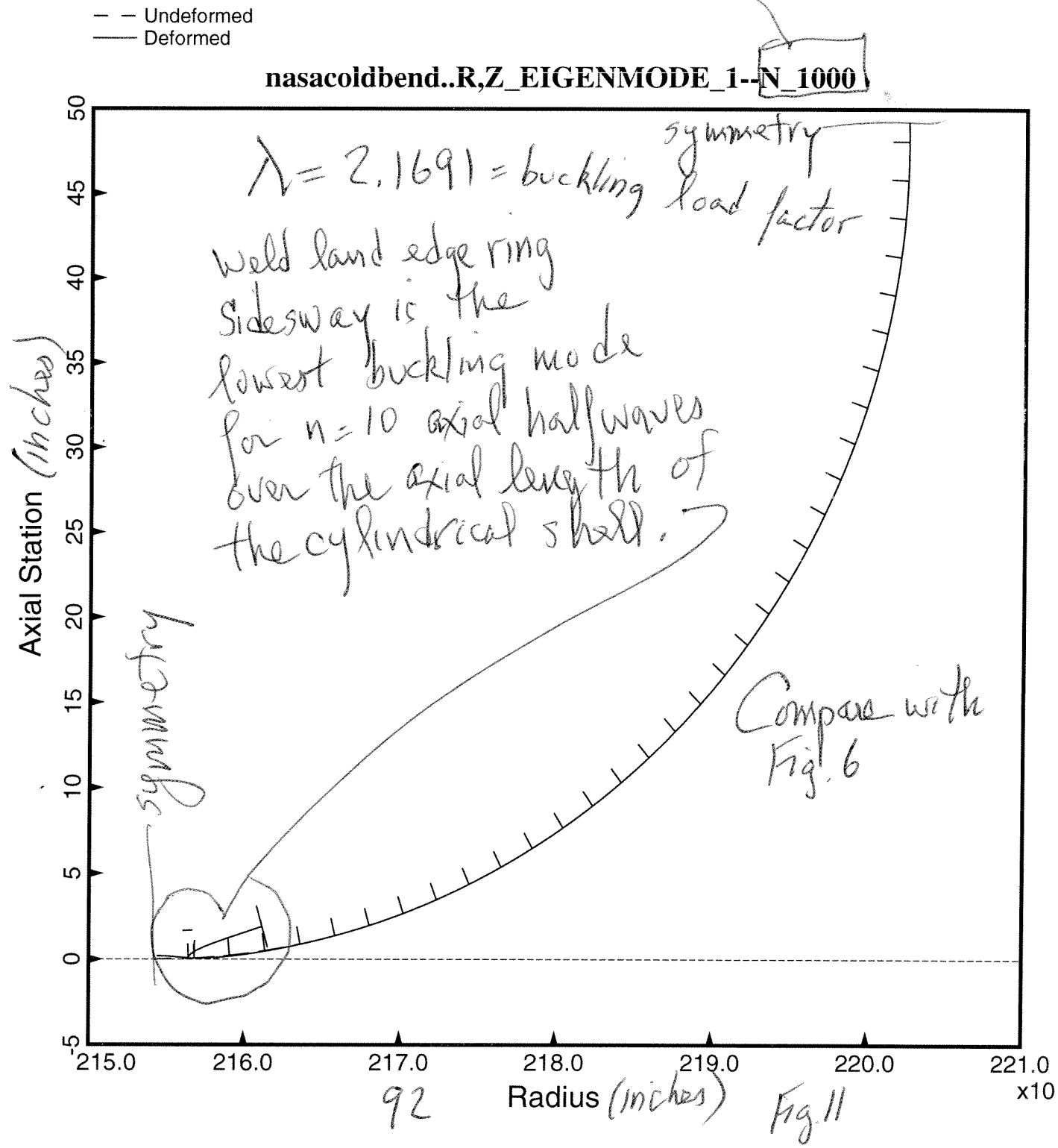


Fig. 11

26 axial half waves over the length of the cylindrical shell

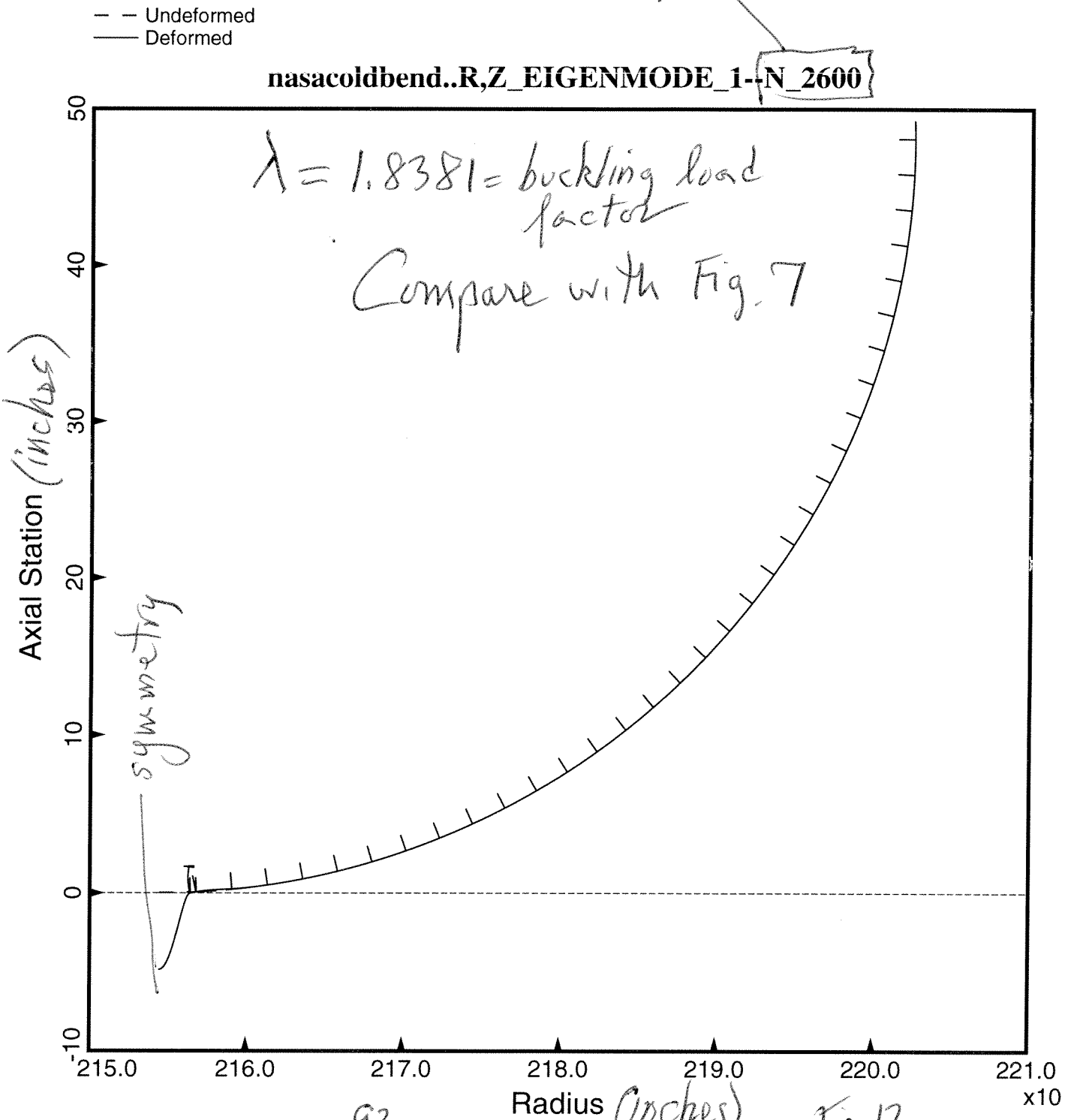


Fig. 12

Table 26 nasacoldbend.PAN

```
     n          $ Do you want a tutorial session and tutorial output?
      75.398    $ Panel length in the plane of the screen, L2
        1       $ Enter control ILAND for weldland (0=none or 1=weldland)
        3       $ Number of BOSOR4-type segments in the weld land, KLAND
  4.000000      $ Width of the weld land, WLAND
0.1000000       $ Thickness of the weld land, TLAND
0.6275600E-01 $ Thickness of "extra" weld land edge stringer web, TWLAND
 1.615600       $ Height of "extra" weld land stringer web, HWLAND
0.2060000E-01 $ Thickness of outstanding flange of weld land stringer, TFLAND
0.6084500       $ Width of outstanding flange of weld land stringer, WFLAND
0.2091000E-01 $ Eccentricity of the weld land outer surface, ECLAND
        1       $ Enter control (1=sym; 2=s.s.) for boundary condition
       -2       $ Enter ILOCAL=0 or 1 or -1 or -2 (Type (H)elp), ILOCAL
        2       $ Number of halfwaves in the axial direction [see H(elp)],NWAVE
        1       $ How many eigenvalues (get at least 3) do you want?
```

input for "panels"

ILOCAL = -2 means shell acreage stringers are smeared out as well as the rings.

Table 27 nasacoldbend.OUT

nasacoldbend.OUT (abridged)
shell acreage stringers and rings smeared out

```
 ***** EIGENVALUES AND MODE SHAPES *****
  EIGENVALUE(CIRC. WAVES)
 ========================================
    2.2798E+00( 100)
    2.0227E+00( 200) <--general buckling
    2.2314E+00( 300)
    2.7317E+00( 400)
    3.4067E+00( 500)
    3.0592E+00( 600)
    2.5684E+00( 700)
    2.2977E+00( 800)
    2.1708E+00( 900)
    2.1400E+00(1000) <--edge stringer sidesway
    2.1758E+00(1100)
    2.2589E+00(1200)
    2.3752E+00(1300)
    2.5077E+00(1400)
    2.4973E+00(1500)
    2.3518E+00(1600)
    2.2279E+00(1700)
    2.1271E+00(1800)
    2.0463E+00(1900)
    1.9824E+00(2000)
    1.9330E+00(2100)
    1.8959E+00(2200)
    1.8696E+00(2300)
    1.8527E+00(2400)
    1.8441E+00(2500)
    1.8428E+00(2600) <--weld land buckling
    1.8481E+00(2700)
    1.8594E+00(2800)
    1.8761E+00(2900)
    1.8978E+00(3000)
 ========================================
```

(Fig. 13)

(Fig. 14)

(Fig. 15)

Compare with Table 25.

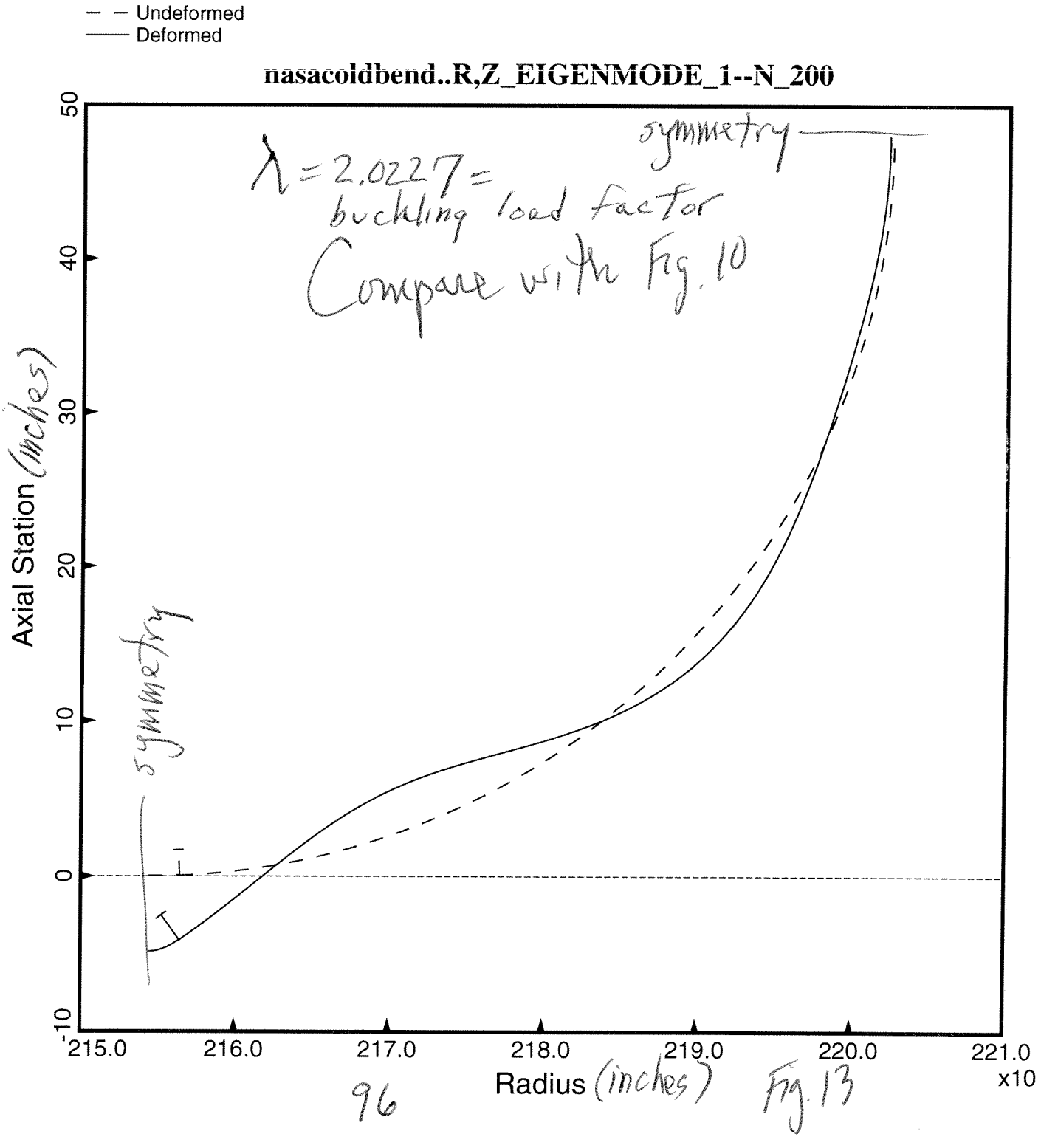

– – Undeformed
—— Deformed
nasacoldbend..R,Z_EIGENMODE_1--N_200
λ = 2.0227 = buckling load factor
Compare with Fig. 10
symmetry
symmetry
Axial Station (inches)
50
40
30
20
10
0
-10
215.0
216.0
217.0
218.0
219.0
220.0
221.0
x10
Radius (inches)

Fig. 13

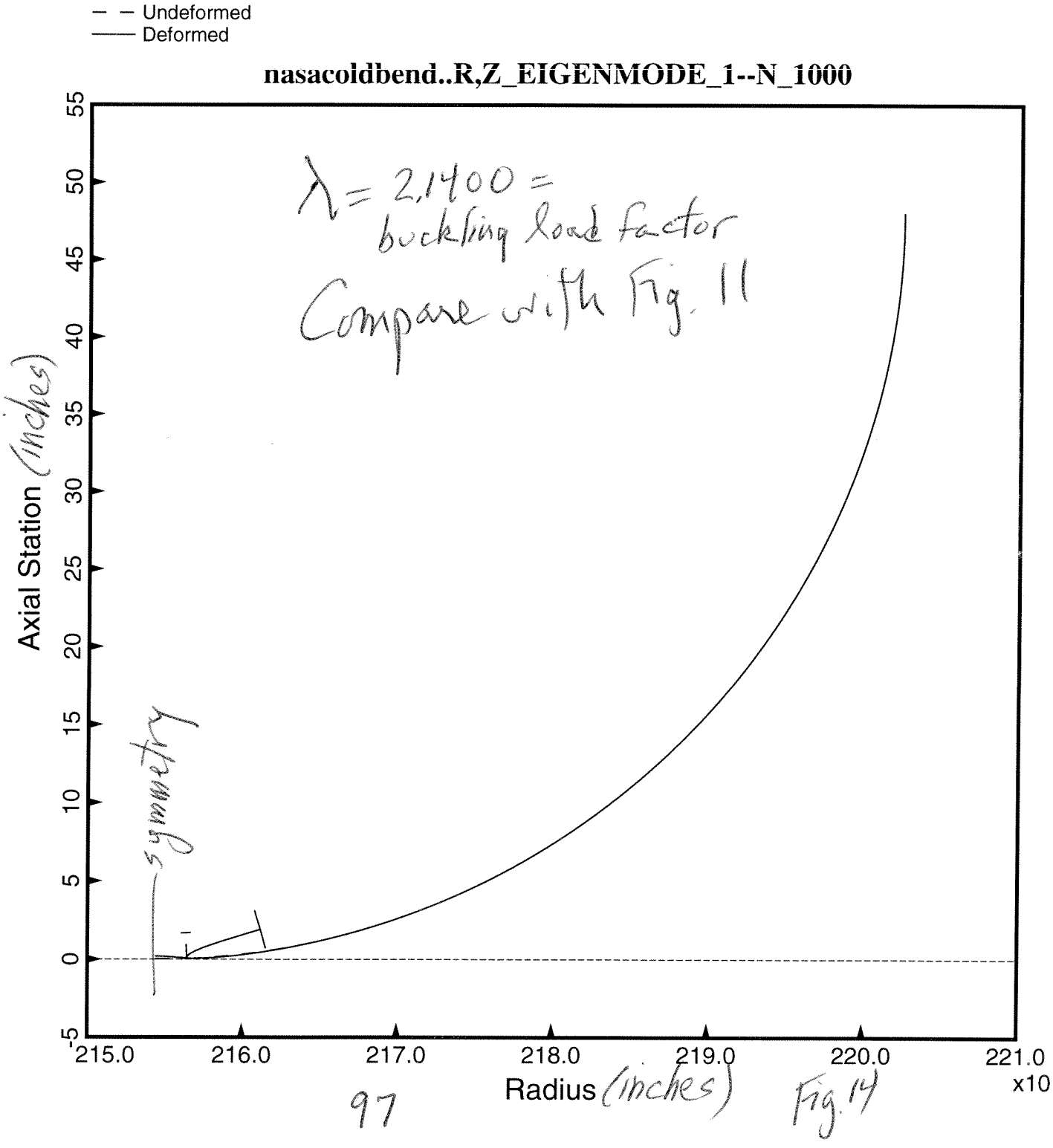

– – Undeformed
—— Deformed
nasacoldbend..R,Z_EIGENMODE_1--N_1000
λ = 2.1400 = buckling load factor
Compare with Fig. 11
Axial Station (inches)
55
50
45
40
35
30
25
20
15
10
5
0
-5
Symmetry
215.0
216.0
217.0
218.0
219.0
220.0
221.0
x10
Radius (inches)

Fig. 14

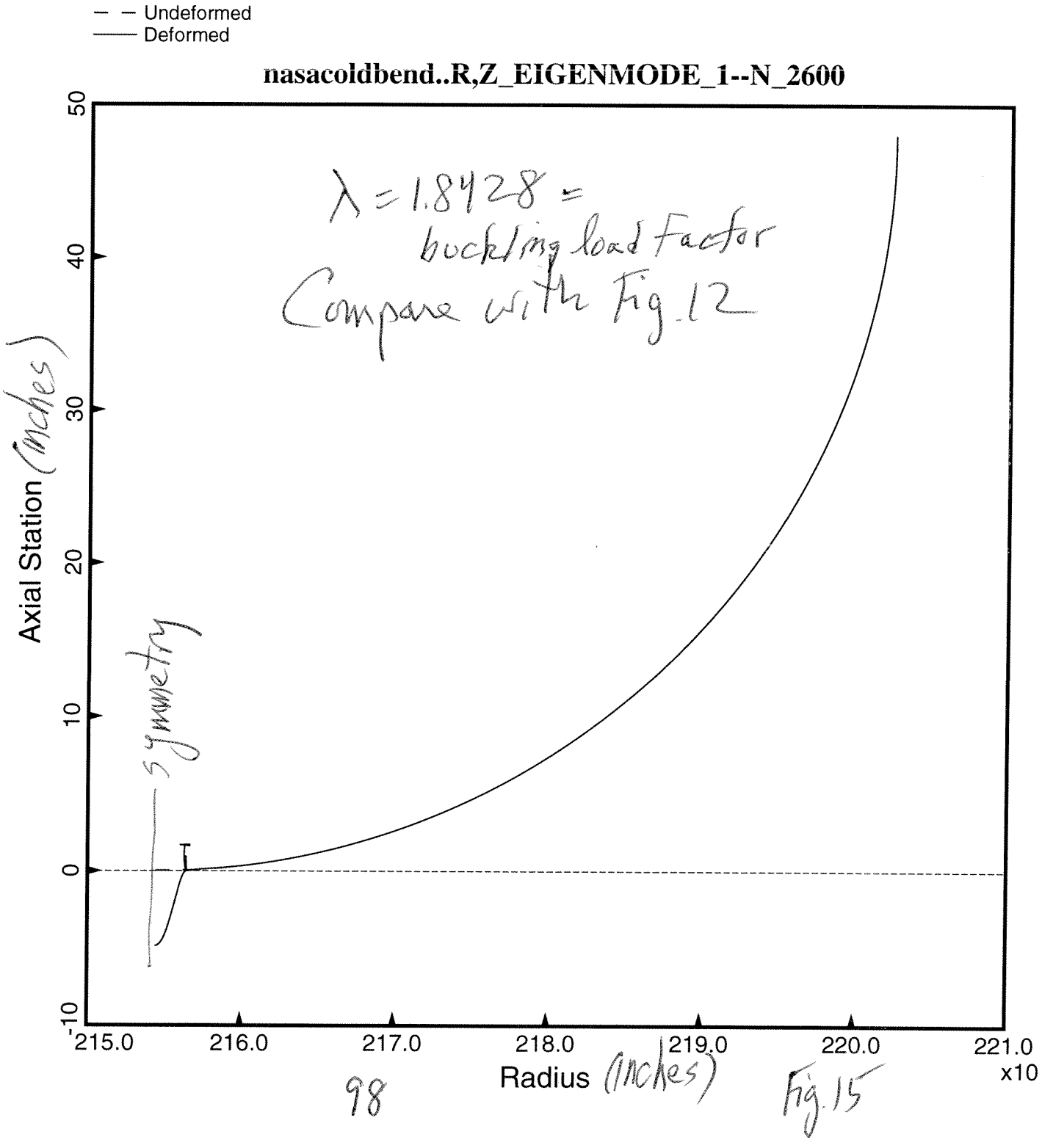

— — Undeformed
—— Deformed
nasacoldbend..R,Z_EIGENMODE_1--N_2600
λ = 1.8428 =
buckling load Factor
Compare with Fig. 12
Symmetry
Axial Station (inches)
50
40
30
20
10
0
-10
215.0
216.0
217.0
218.0
219.0
220.0
221.0
x10
Radius (inches)

Fig. 15

Table 28 nasacoldbend.PAN

```
     n          $ Do you want a tutorial session and tutorial output?
    75.398      $ Panel length in the plane of the screen, L2
       1        $ Enter control ILAND for weldland (0=none or 1=weldland)
       3        $ Number of BOSOR4-type segments in the weld land, KLAND
 4.000000       $ Width of the weld land, WLAND
0.1000000       $ Thickness of the weld land, TLAND
0.6275600E-01   $ Thickness of "extra" weld land edge stringer web, TWLAND
 1.615600       $ Height of "extra" weld land stringer web, HWLAND
0.2060000E-01   $ Thickness of outstanding flange of weld land stringer, TFLAND
0.6084500       $ Width of outstanding flange of weld land stringer, WFLAND
0.2091000E-01   $ Eccentricity of the weld land outer surface, ECLAND
       1        $ Enter control (1=sym; 2=s.s.) for boundary condition
       1        $ Enter ILOCAL=0 or 1 or -1 or -2 (Type (H)elp), ILOCAL
       2        $ Number of halfwaves in the axial direction [see H(elp)],NWAVE
       1        $ How many eigenvalues (get at least 3) do you want?
```

input for "panel3"

ILOCAL = 1 means inter-ring buckling.
shell acreage stringers are shell branches.

Table 29 nasacoldbend.OUT

nasacoldbend.OUT (abridged)
inter-ring buckling

```
 ***** EIGENVALUES AND MODE SHAPES *****
  EIGENVALUE(CIRC. WAVES)
 ========================================
    1.5919E+00( 100)
    1.4446E+00( 200) <--close to critical
    1.4536E+00( 300)
    1.3643E+00( 400)
    1.3515E+00( 500) <--critical value
    1.3735E+00( 600)
    1.4169E+00( 700)
    1.4769E+00( 800)
    1.5524E+00( 900)
 ========================================
```

(Fig. 16)
(Fig. 17)
(Fig. 18)

Inter-ring buckling