Appendix E

Code

E.1 Subroutines for the isotropic 8β element

```
C=============================================================SHL8
C
C     ELEMENT STIFFNESS MATRIX FOR AN ASSUMED STRESS
C     FOUR NODE MEMBRANE ELEMENT WITH DRILLING DEGREES OF
C     FREEDOM - 8 BETA PARAMETERS
C
C     (K+P)a = f - DISPLACEMENT FORMULATION
C     (K+h)a = f - MIXED FORMULATION
C
C-----------------------------------------------------------------
      SUBROUTINE SHL8(XYZ,ID,MAXN,NSHL8)

      IMPLICIT REAL*8 (A-H,O-Z)
      PARAMETER (NP=8)

      DIMENSION XYZ(4,MAXN),ID(6,MAXN),S(24,24)
      DIMENSION RS(2,16),GWS(2,16),B(3,14),BY(1,14)
      DIMENSION T1(3,14),T4(14,14),T1Y(1,14),T2Y(14,14)
      DIMENSION BTAY(3,4),BTAY2(3,4)
      DIMENSION P(3,NP),P1(3,NP),T2(NP,14),T3(NP,14)
      DIMENSION P2(NP,NP),XDU(NP,NP),CINV(3,3)
      DIMENSION NDOF(6),PPURE(3,NP)
      DIMENSION PTRANSO(3,3),PTRANS(3,3),QTRANS(3,3)
      DIMENSION IPVT(NP),WORK(NP),TEST1(NP,NP)
      DIMENSION NEIG(24),SEIG(24,24)
      DIMENSION BPL(3,12),BS(2,12),T1S(2,12)
      DIMENSION T1PL(6,12),T2PL(12,12)
      DIMENSION DB(3,3),DS(2,2)

      COMMON /CONSTR/ ICONSTR(9)
```

```
      COMMON /VECT   / V(4,3),XYL(3,5)
      COMMON /GENR   / NAM,INE,INI,NIDENT
      COMMON /SIDE   / ALPHA(4),XLENG(4)
      COMMON /TEMP   / IDE,II,JJ,KK,LL,HH,EE,UU
      COMMON /WEIGHI/ WGHTO
      COMMON /IOLIST/ NTM,NTR,NIN,NOT,NSP,NFL,NT7,NT8

      DATA ZERO /0.D0/, ONE /1.D0/, TWO /2.D0/, FOUR /4.D0/
      DATA NEIG /1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,
     &           19,20,21,22,23,24/
      DATA NDOF /1,2,3,4,5,6/
      DATA FF   /1.2D0/

      IAXIS=1

      ND=24
      NUMN=4
      NTOT=0
      NTELE=2            !  DEFAULT - DISPLACEMENT FORMULATION
      INT=9              !  DEFAULT - FULL INTEGRATION
      INTTAY=4
      INT2=4
      NWARP=0            !  DEFAULT - WARP CORRECTION NOT ACTIVE
      NPRINT=0
      NSTAT=1
      NEIGEN=0
      NLOCK=1            !  DEFAULT - LOCKING CORRECTION ACTIVE
      WGHTO=0.01
      GG=-1.             !  DEFAULT - GG=EE/(TWO+TWO*UU)
      IMIXED=5           !  DEFAULT - NT-FORMULATION
      ALPHACORR=ONE
      IPSTATE=2          !  DEFAULT - P-MATRIX CLASSIFICATION

C-----LOOP THROUGH ALL ELEMENTS------------------------------------
      DO 400 MM=1,NSHL8

C-----READ NODE NUMBERS AND MATERIAL PROPERTIES--------------------
C-------AND ELEMENT PARAMETERS-------------------------------------
      NAM=0
      NIDENT=0
      CALL CFREE
      CALL CFREEPT
      CALL CFREEI(' ',IDE,1)
      CALL CFREEI('N',II,4)
      CALL CFREER('H',HH,1)
      CALL CFREER('E',EE,1)
      CALL CFREER('U',UU,1)
      CALL CFREEI('G',NAM,4)
```

```
      CALL CFREEI('I',INT,1)
      CALL CFREEI('J',INTTAY,1)
      CALL CFREEI('K',INT2,1)
      CALL CFREER('X',GG,1)
      CALL CFREEI('S',NSTAT,1)
      CALL CFREEI('T',NTELE,1)
      CALL CFREEI('B',IPSTATE,1)
      CALL CFREEI('L',NEIGEN,1)
      CALL CFREEI('Z',NLOCK,1)
      CALL CFREER('D',WGHTO,1)
      CALL CFREEI('W',NWARP,1)
      CALL CFREEI('A',IMIXED,1)
      CALL CFREER('Y',ALPHACORR,1)
      CALL CFREEI('F',ICONSTR,9)       !  READ DIFFERENT P-MATRIX

      IF (GG.LT.ZERO) GG=EE/(TWO+TWO*UU)
      IF (GG.EQ.ZERO) NTELE=3
      IF (NAM.EQ.0) NIDENT=0
      NCHCK=NAM

C-----PRINT ELEMENT DATA------------------------------------------
 10   IF (NPRINT.NE.0) GO TO 15
      CALL TOP
      WRITE (NOT,2003) IMIXED,INT
      WRITE (NOT,2000)
 15   WRITE (NOT,2001) IDE,II,JJ,KK,LL,HH,EE,GG,UU
      NPRINT=NPRINT+1
      IF(NPRINT.GT.50) NPRINT=0

C-----SKIP FOR IDENTICAL ELEMENTS---------------------------------
      IF (NIDENT.LT.0.OR.NIDENT.GT.1) NIDENT=0
      IF (NIDENT.EQ.1.AND.NCHCK.GT.NAM) GOTO 295

C-----LOCAL COORDINATES, SIDE LENGTHS & ANGLES--------------------
      CALL LOCAL(XYZ,MAXN,IAXIS)

      DO 19 I=1,4
      CALL ANGLEN(I,ALPHA(I),XLENG(I))
 19   CONTINUE

C-----------------------------------------------------------------
C
C  ELEMENT K MATRIX - MIXED AND DISPL. FORMULATION
C
C-----------------------------------------------------------------

C-----CALCULATE COMPLIANCE MATRIX---------------------------------
      CALL MATLWINV(NSTAT,EE,UU,CINV)
```

```
C-----INITIALIZE MATRICES------------------------------------------
      DO 20 I=1,24
      DO 20 J=1,24
 20   S(I,J)=ZERO

      DO 22 J=1,14
      B(1,J)=ZERO
      B(2,J)=ZERO
      B(3,J)=ZERO
      T1Y(1,J)=ZERO
      BY(1,J)=ZERO
 22   CONTINUE

      DO 23 I=1,NP
      DO 23 J=1,NP
 23   P2(I,J)=ZERO

      DO 24 I=1,14
      DO 24 J=1,14
      T2Y(I,J)=ZERO
 24   CONTINUE

      DO 25 I=1,NP
      DO 25 J=1,14
 25   T2(I,J)=ZERO

      DO 26 J=1,4
      BTAY(1,J)=ZERO
      BTAY(2,J)=ZERO
      BTAY(3,J)=ZERO
 26   CONTINUE

      DO 27 I=1,3
      DO 27 J=1,NP
 27   P(I,J)=ZERO

      CALL FORMTRANSC (PTRANSO)
      CALL MEMXB3 (BY,DETJ0)

C-----CALCULATE ELEMENT AREA---------------------------------------
      CALL ELAREA(AREA)

C-----CALCULATE MEMBRANE LOCKING CORRECTION------------------------
      IF (NLOCK.EQ.1) THEN
      CALL INTPTS(INTTAY,RS,GWS)
      DO 35 I=1,INTTAY
      CALL MEMLOK(I,RS,BTAY2,DETJ)
```

```
      DO 35 K=1,3
      DO 35 L=1,4
      BTAY(K,L)=BTAY(K,L)+BTAY2(K,L)*GWS(1,I)*GWS(2,I)*DETJ/AREA
 35   CONTINUE
      ENDIF

C-----DEFINE INTEGRATION POINTS-----------------------------------
      CALL INTPTS(INT,RS,GWS)

C-----LOOP THROUGH THE INTEGRATION POINTS-------------------------
      DO 80 I=1,INT
      CALL MEMXBTJ  (I,RS,B,DETJ,BTAY,NLOCK,AJM11,AJM12,AJM21,AJM22)
      IF (IPSTATE.EQ.2) THEN
        CALL HQ8P2 (I,RS,PPURE)                  !  DETERMINE P
      ELSEIF (IPSTATE.EQ.1) THEN
        CALL HQ8P1 (I,RS,PPURE)                  !  DETERMINE P
      ELSE
        CALL HQ8PA (I,RS,PPURE)                  !  DETERMINE P
      ENDIF

      CALL FIXP (P,PTRANS0,PTRANS,QTRANS,AJM11,AJM12,AJM21,AJM22,
     &           DETJ,DETJ0,PPURE,IMIXED,3,NP)  !  TRANSFORM P

C-----EVALUATE INTEGRAL-------------------------------------------
C.....C^-1 x P ...................................................
      DO 50 K=1,3
      DO 50 L=1,NP
      C1=CINV(K,1)*P(1,L)+CINV(K,2)*P(2,L)+CINV(K,3)*P(3,L)
      P1(K,L)=C1*DETJ*HH*GWS(1,I)*GWS(2,I)
 50   CONTINUE

C.....P^t x C^-1 x P..............................................
      DO 54 K=1,NP
      DO 54 L=1,NP
      C1=P(1,K)*P1(1,L)+P(2,K)*P1(2,L)+P(3,K)*P1(3,L)
      P2(K,L)=P2(K,L)+C1
 54   CONTINUE

C.....P^t x B.....................................................
      DO 60 K=1,NP
      DO 60 L=1,12
      C1=P(1,K)*B (1,L)+P(2,K)*B (2,L)+P(3,K)*B (3,L)
      T2(K,L)=T2(K,L)+C1*DETJ*HH*GWS(1,I)*GWS(2,I)
 60   CONTINUE

C-----END OF INTEGRATION LOOP-------------------------------------

 80   CONTINUE
```

```
C-----DETERMINE INVERSE OF H -------------------------------------
      CALL XEQY(XDU,P2,NP,NP)
      CALL DGEFA(P2,NP,NP,IPVT,INFO)
      JOB=01
      CALL DGEDI(P2,NP,NP,IPVT,DETX,WORK,JOB)
      CALL TESTINVERSE(XDU,P2,NP,NP,TEST1)

C.....H^-1 * G...................................................
      DO 90 K=1,NP
      DO 90 L=1,12
      C1=P2(K,1)*T2(1,L)+P2(K,2)*T2(2,L)+P2(K,3)*T2(3,L)
     &  +P2(K,4)*T2(4,L)+P2(K,5)*T2(5,L)+P2(K,6)*T2(6,L)
     &  +P2(K,7)*T2(7,L)+P2(K,8)*T2(8,L)
      T3(K,L)=C1
 90   CONTINUE

C.....K = G^t * H^-1 * G.........................................
      DO 95 K=1,12
      DO 95 L=1,12
      C1=T2(1,K)*T3(1,L)+T2(2,K)*T3(2,L)+T2(3,K)*T3(3,L)
     &  +T2(4,K)*T3(4,L)+T2(5,K)*T3(5,L)+T2(6,K)*T3(6,L)
     &  +T2(7,K)*T3(7,L)+T2(8,K)*T3(8,L)
      T4(K,L)=C1
 95   CONTINUE

C-----------------------------------------------------------------
C
C  ELEMENT P MATRIX - DISPLACEMENT FORMULATION
C
C-----------------------------------------------------------------

      IF (NTELE.EQ.3) GOTO 280
      IF (NTELE.EQ.1) GOTO 200

C-----EVALUATE INTEGRAL AND SUM K AND P MATRICES------------------
      DO 140 L=1,12
      T1Y(1,L)=BY(1,L)*DETJO*HH*FOUR
 140  CONTINUE

      DO 141 K=1,12
      DO 141 L=1,12
      T2Y(K,L)=BY(1,K)*T1Y(1,L)
 141  CONTINUE

      DO 150 K=1,12
      DO 150 L=1,12
      T4(K,L)=T4(K,L)+t2y(k,l)*GG
```

```
 150  CONTINUE
      GOTO 280

C-----------------------------------------------------------------
C
C  ELEMENT h MATRIX - MIXED FORMULATION
C
C-----------------------------------------------------------------

C-----DEFINE INTEGRATION POINTS-----------------------------------
 200  CALL INTPTS(INT2,RS,GWS)

C-----LOOP THROUGH THE INTEGRATION POINTS-------------------------
      DO 250 I=1,INT2
      CALL MEMXB2 (I,RS,BY,DETJ)

C-----EVALUATE INTEGRAL-------------------------------------------
      DO 250 L=1,12
      T1Y(1,L)=T1Y(1,L)+BY(1,L)*DETJ*HH*GWS(1,I)*GWS(2,I)
 250  CONTINUE

C.....h * hT......................................................
      DO 260 I=1,12
      DO 260 J=1,12
      T2Y(I,J)=T1Y(1,I)*T1Y(1,J)
 260  CONTINUE

C.....K + h.......................................................

      VOLEL=HH*AREA
      DO 270 I=1,12
      DO 270 J=1,12
      T4(I,J)=T4(I,J)+T2Y(I,J)*GG/VOLEL
 270  CONTINUE

C-----SUM TO FORM MEMBRANE STIFFNESS------------------------------
 280  DO 290 K=1,4
      DO 290 L=1,4
      S(6*K-5,6*L-5)=T4(3*K-2,3*L-2)
      S(6*K-4,6*L-4)=T4(3*K-1,3*L-1)
      S(6*K-5,6*L-4)=T4(3*K-2,3*L-1)
      S(6*K-4,6*L-5)=T4(3*K-1,3*L-2)

C.....DRILLING ROTATIONS..........................................
      S(6*K-0,6*L-0)=T4(3*K-0,3*L-0)
      S(6*K-5,6*L-0)=T4(3*K-2,3*L-0)
      S(6*K-4,6*L-0)=T4(3*K-1,3*L-0)
      S(6*K-0,6*L-5)=T4(3*K-0,3*L-2)
```

```
      S(6*K-0,6*L-4)=T4(3*K-0,3*L-1)
 290  CONTINUE

C-----------------------------------------------------------------
C
C   ELEMENT K MATRIX (BENDING STIFFNESS)
C
C-----------------------------------------------------------------

      INT3=4

C-----DEFINE INTEGRATION POINTS-----------------------------------
      CALL INTPTS(INT3,RS,GWS)

C-----CALCULATE MATERIAL MATRIX-----------------------------------
      CALL MATPL(EE,UU,HH,FF,DB,DS)

C-----INITIALIZE MATRICES-----------------------------------------
      DO 530 I=1,3
      DO 530 J=1,12
 530  BPL(I,J)=ZERO

      DO 531 I=1,2
      DO 531 J=1,12
 531  BS(I,J)=ZERO

      DO 535 I=1,12
      DO 535 J=1,12
      T2PL(I,J)=ZERO
 535  CONTINUE

C-----CALC. JACOBIAN MATRIX AT SAMP. POINTS-----------------------
      CALL JASAPM

C-----LOOP THROUGH THE INTEGRATION POINTS-------------------------
      DO 590 I=1,INT3
      CALL CPT1B (I,RS,BPL,BS,DETJ)

C-----EVALUATE INTEGRAL (MATRIX MULTIPLICATION)-------------------
      DO 550 K=1,3
      DO 550 L=1,12
      C1=DB(K,1)*BPL(1,L)+DB(K,2)*BPL(2,L)+DB(K,3)*BPL(3,L)
      T1PL(K,L)=C1*GWS(1,I)*GWS(2,I)*DETJ
 550  CONTINUE

      DO 555 K=1,2
      DO 555 L=1,12
      C1=DS(K,1)*BS(1,L)+DS(K,2)*BS(2,L)
```

```
      T1S(K,L)=C1*GWS(1,I)*GWS(2,I)*DETJ
 555  CONTINUE

      DO 560 K=1,12
      DO 560 L=1,12
      C1=BPL(1,K)*T1PL(1,L)+BPL(2,K)*T1PL(2,L)+BPL(3,K)*T1PL(3,L)
      C2=BS (1,K)*T1S (1,L)+BS (2,K)*T1S (2,L)
      T2PL(K,L)=T2PL(K,L)+C1+C2
 560  CONTINUE

 590  CONTINUE

C-----END INTEGRATION LOOP-----------------------------------------

C-----SUM TO FORM ELEMENT STIFFNESS MATRIX-------------------------
      DO 580 K=1,4
      DO 580 L=1,4
      S(6*K-3,6*L-3)=T2PL(3*K-2,3*L-2)
      S(6*K-2,6*L-2)=T2PL(3*K-1,3*L-1)
      S(6*K-1,6*L-1)=T2PL(3*K-0,3*L-0)
      S(6*K-3,6*L-2)=T2PL(3*K-2,3*L-1)
      S(6*K-2,6*L-3)=T2PL(3*K-1,3*L-2)
      S(6*K-3,6*L-1)=T2PL(3*K-2,3*L-0)
      S(6*K-1,6*L-3)=T2PL(3*K-0,3*L-2)
      S(6*K-2,6*L-1)=T2PL(3*K-1,3*L-0)
      S(6*K-1,6*L-2)=T2PL(3*K-0,3*L-1)
 580  CONTINUE

C-----CONVERT FROM FLAT TO WARPED----------------------------------
      IF (NWARP.NE.0) CALL TWISTM(S)

C-----ROTATE STIFNESS TO GLOBAL SYSTEM-----------------------------
      CALL ROTATE(S)

C-----IDENTIFY DOF WHICH HAVE STIFFNESS----------------------------
 295  DO 320 K=1,6
      IF (S(K,K).NE.ZERO) ID(K,II)=1
      IF (S(K+6,K+6).NE.ZERO) ID(K,JJ)=1
      IF (S(K+12,K+12).NE.ZERO) ID(K,KK)=1
      IF (S(K+18,K+18).NE.ZERO) ID(K,LL)=1
 320  CONTINUE

C-----SAVE ELEMENT ARRAYS------------------------------------------
      WRITE (NT7) IDE,NUMN,II,JJ,KK,LL,NDOF,ND
      WRITE (NT8) S
      WRITE (NSP) IDE,II,JJ,KK,LL,XYL,T3,IMIXED,PTRANSO,DETJO,
     &            BTAY,NLOCK,ALPHACORR,NWARP
```

```
C-----SAVE STIFFNESS MATRIX TO SOLVE EIGENVALUES------------------
      IF (NEIGEN.NE.0) THEN
      NPRINT=0
      NEX=24
      DO 350 I=1,NEX
      DO 350 J=1,NEX
 350  SEIG(I,J)=S(NEIG(I),NEIG(J))
      CALL CALJAC(SEIG,NEX,NEIGEN)
      ENDIF

C-----CHECK FOR ELEMENT GENERATION--------------------------------
      IF (NAM.EQ.0) GOTO 400
      NAM=NAM-1
      IDE=IDE+INE
      II=II+INI
      JJ=JJ+INI
      KK=KK+INI
      LL=LL+INI
      NTOT=NTOT+1
      GOTO 10
 400  IF ((MM+NTOT).EQ.NSHL8) RETURN

C-----------------------------------------------------------------
 2000 FORMAT (/,' EL. #    I    J    K    L         H            E'
     *          ,'             G         U ')
 2001 FORMAT (1I6,4I5,1F9.3,2E12.3,1F9.3)
 2002 FORMAT (5E10.3)
 2003 FORMAT (/,' ELEMENT TYPE: SHL8, Version: ',1I1,/,
     *           ' INTEGRATION SCHEME: ',1I2,'-POINT')
      END
C-----------------------------------------------------------------

C==========================================================MEMXBTJ
C
C     SUBROUTINE EVALUATES THE OPERATOR MATRIX
C     [B-MATRIX] FOR THE ELEMENT K MATRIX OF A
C     FOUR NODE 8 DOF MEMBRANE ELEMENT WITH IN-
C     PLANE DRILLING ROTATIONS.
C
C-----------------------------------------------------------------
      SUBROUTINE MEMXBTJ(I,RS,B,DETJ,BTAY,NLOCK,
     &                   AJM11,AJM12,AJM21,AJM22)

      IMPLICIT REAL*8 (A-H,O-Z)

      DIMENSION RS(2,16),QR(9),QS(9),B(3,14),BTAY(3,4)

      COMMON /VECT  / V(4,3),XYL(3,5)
```

```
      COMMON /TEMP   / IDE,II,JJ,KK,LL,HH,EE,UU
      COMMON /SIDE   / ALPHA(4),XLENG(4)
      COMMON /IOLIST/ NTM,NTR,NIN,NOT,NSP,NFL,NT7,NT8

      DATA  ZERO /0.D0/, ONE /1.D0/, TWO /2.D0/,
     &      FOUR /4.D0/, EIGHT /8.D0/

      RCOORD=RS(1,I)
      SCOORD=RS(2,I)

      XYL11=XYL(1,1)
      XYL21=XYL(2,1)
      XYL12=XYL(1,2)
      XYL22=XYL(2,2)
      XYL13=XYL(1,3)
      XYL23=XYL(2,3)
      XYL14=XYL(1,4)
      XYL24=XYL(2,4)

C-----CALCULATE SHAPE FCN. DERIVITIVES AT GIVEN POINT--------------
C.....CALCULATE LINEAR FUNCTIONS..................................
      QR1=-(ONE-SCOORD)/FOUR
      QR2= (ONE-SCOORD)/FOUR
      QR3= (ONE+SCOORD)/FOUR
      QR4=-(ONE+SCOORD)/FOUR
      QR(1)=QR1
      QR(2)=QR2
      QR(3)=QR3
      QR(4)=QR4

C.....CALCULATE QUADRATIC FUNCTIONS...............................
      QR(5)=-RCOORD*(ONE-SCOORD)
      QR(6)= (ONE-SCOORD)*(ONE+SCOORD)/TWO
      QR(7)=-RCOORD*(ONE+SCOORD)
      QR(8)=-(ONE-SCOORD)*(ONE+SCOORD)/TWO
      QR(9)=-TWO*RCOORD*(ONE-SCOORD**TWO)

C.....CALCULATE LINEAR FUNCTIONS..................................
      QS1=-(ONE-RCOORD)/FOUR
      QS2=-(ONE+RCOORD)/FOUR
      QS3= (ONE+RCOORD)/FOUR
      QS4= (ONE-RCOORD)/FOUR
      QS(1)=QS1
      QS(2)=QS2
      QS(3)=QS3
      QS(4)=QS4

C.....CALCULATE QUADRATIC FUNCTIONS...............................
```

```
      QS(5)=-(ONE+RCOORD)*(ONE-RCOORD)/TWO
      QS(6)=-SCOORD*(ONE+RCOORD)
      QS(7)= (ONE+RCOORD)*(ONE-RCOORD)/TWO
      QS(8)=-SCOORD*(ONE-RCOORD)
      QS(9)=-TWO*SCOORD*(ONE-RCOORD**TWO)

C-----CALCULATE JACOBIAN MATRIX-----------------------------------
      AJM11=QR1*XYL11+QR2*XYL12 + QR3*XYL13+QR4*XYL14
      AJM12=QR1*XYL21+QR2*XYL22 + QR3*XYL23+QR4*XYL24
      AJM21=QS1*XYL11+QS2*XYL12 + QS3*XYL13+QS4*XYL14
      AJM22=QS1*XYL21+QS2*XYL22 + QS3*XYL23+QS4*XYL24

C-----INVERT JACOBIAN MATRIX--------------------------------------
      DETJ=AJM11*AJM22-AJM12*AJM21
      IF (DETJ.LE.0.) WRITE(NOT,2000)DETJ
      IF (DETJ.LE.0.) WRITE(NTM,2000)DETJ
      AJI11= AJM22/DETJ
      AJI22= AJM11/DETJ
      AJI12=-AJM12/DETJ
      AJI21=-AJM21/DETJ

C-----CALCULATE SIDE LENGTHS AND ANGLES---------------------------
      DO 40 IC=1,4
      MC=IC+4
      TEMP=ONE/IC
      LC=MC-1+4*INT(TEMP)
      KC=MOD(MC,4)+1
      JC=LC-4
      XLENIJ=XLENG(JC)
      XLENIK=XLENG(IC)
      ALPHIJ=ALPHA(JC)
      ALPHIK=ALPHA(IC)

C-----EVALUATE B-MATRIX-------------------------------------------
      B(1,3*IC-2)=QR(IC)*AJI11+QS(IC)*AJI12
      B(3,3*IC-1)=B(1,3*IC-2)
      B(2,3*IC-1)=QR(IC)*AJI21+QS(IC)*AJI22
      B(3,3*IC-2)=B(2,3*IC-1)
      B(1,3*IC)=XLENIJ*DCOS(ALPHIJ)/EIGHT*
     &          (QR(LC)*AJI11+QS(LC)*AJI12)
     &         -XLENIK*DCOS(ALPHIK)/EIGHT*
     &          (QR(MC)*AJI11+QS(MC)*AJI12)
      B(2,3*IC)=XLENIJ*DSIN(ALPHIJ)/EIGHT*
     &          (QS(LC)*AJI22+QR(LC)*AJI21)
     &         -XLENIK*DSIN(ALPHIK)/EIGHT*
     &          (QS(MC)*AJI22+QR(MC)*AJI21)
      B(3,3*IC)=XLENIJ*DCOS(ALPHIJ)/EIGHT*
     &          (QS(LC)*AJI22+QR(LC)*AJI21)
```

```
     &          -XLENIK*DCOS(ALPHIK)/EIGHT*
     &           (QS(MC)*AJI22+QR(MC)*AJI21)
     &          +XLENIJ*DSIN(ALPHIJ)/EIGHT*
     &           (QR(LC)*AJI11+QS(LC)*AJI12)
     &          -XLENIK*DSIN(ALPHIK)/EIGHT*
     &           (QR(MC)*AJI11+QS(MC)*AJI12)

C-----CORRECT SHAPE FUNCTIONS: MEMBRANE LOCKING--------------------
      IF (NLOCK.EQ.1) THEN
      B(1,3*IC)=B(1,3*IC)-BTAY(1,IC)
      B(2,3*IC)=B(2,3*IC)-BTAY(2,IC)
      B(3,3*IC)=B(3,3*IC)-BTAY(3,IC)
      ENDIF
  40  CONTINUE

C-----ADD BUBBLE---------------------------------------------------
      B(1,13)=QR(9)*AJI11+QS(9)*AJI12
      B(2,14)=QR(9)*AJI21+QS(9)*AJI22
      B(3,13)=B(2,14)
      B(3,14)=B(1,13)
      RETURN

C-----------------------------------------------------------------
 2000 FORMAT(' JACOBIAN=',E15.6,
     *        ' NODE NUMBERS NOT IN ORDER ')
      END
C-----------------------------------------------------------------

C===========================================================HQ8PA
C
C     SUBROUTINE EVALUATES THE P MATRIX OF A HYBRID FOUR NODE
C     MEMBRANE ELEMENT USING 8 BETA PARAMETERS.
C
C-----------------------------------------------------------------
      SUBROUTINE HQ8PA(I,RS,P)

      IMPLICIT REAL*8 (A-H,O-Z)
      PARAMETER (NPOSS=57)

      DIMENSION RS(2,16),P(3,8), SM(3,NPOSS)

      COMMON /IOLIST/ NTM,NTR,NIN,NOT,NSP,NFL,NT7,NT8
      COMMON /CONSTR/ ICONSTR(9)

      DATA  ZERO /0.D0/, ONE /1.D0/, TWO /2.D0/

C-----OBTAIN R & S COORDINATES------------------------------------
      R=RS(1,I)
```

```
      S=RS(2,I)

C-----INITIALIZE-----------------------------------------------
      DO 10 K=1,3
      DO 10 L=1,8
 10   P(K,L)=ZERO
      DO 20 K=1,3
      DO 20 L=1,NPOSS
 20   SM(K,L)=ZERO

C-----EVALUATE P-----------------------------------------------
      SM(1, 1)=ONE
      SM(2, 2)=ONE
      SM(3, 3)=ONE

      SM(1, 4)=R
      SM(1, 5)=S
      SM(2, 6)=R
      SM(2, 7)=S
      SM(3, 8)=R
      SM(3, 9)=S

      SM(1,10)=ONE
      SM(2,10)=ONE
      SM(1,11)=ONE
      SM(2,11)=-ONE

      SM(2,12)=-S
      SM(3,12)=R
      SM(1,13)=-R
      SM(3,13)=S

      SM(1,14)=R*R
      SM(2,15)=R*R
      SM(3,16)=R*R
      SM(1,17)=S*S
      SM(2,18)=S*S
      SM(3,19)=S*S

      SM(1,20)=R*S
      SM(2,21)=R*S
      SM(3,22)=R*S

      SM(1,23)=-R*R
      SM(2,24)=-R*R
      SM(3,25)=-R*R
      SM(1,26)=-S*S
      SM(2,27)=-S*S
```

```
SM(3,28)=-S*S

SM(1,29)=-R*S
SM(2,30)=-R*S
SM(3,31)=-R*S

SM(1,32)=S*S
SM(2,32)=-R*R

SM(1,33)=S
SM(2,33)=-R
SM(1,34)=S
SM(3,34)=-R
SM(1,35)=-R
SM(2,35)=S
SM(2,36)=S
SM(3,36)=-R
SM(1,37)=-R
SM(3,37)=S

SM(2,38)=-R
SM(3,38)=S

SM(1,39)=R
SM(2,39)=-S
SM(1,40)=R
SM(3,40)=-S
SM(1,41)=-S
SM(2,41)=R
SM(2,42)=R
SM(3,42)=-S

SM(1,43)=-S
SM(3,43)=R

SM(2,44)=-S
SM(3,44)=R

SM(1,45)=S*S
SM(3,45)=-R*R
SM(1,46)=-R*R
SM(2,46)=S*S
SM(2,47)=S*S
SM(3,47)=-R*R
SM(1,48)=-R*R
SM(3,48)=S*S
SM(2,49)=-R*R
SM(3,49)=S*S
```

```
      SM(1,50)=R*R
      SM(2,50)=-S*S
      SM(1,51)=R*R
      SM(3,51)=-S*S
      SM(1,52)=-S*S
      SM(2,52)=R*R
      SM(2,53)=R*R
      SM(3,53)=-S*S
      SM(1,54)=-S*S
      SM(3,54)=R*R
      SM(2,55)=-S*S
      SM(3,55)=R*R

      SM(1,56)=S*S
      SM(3,56)=-R*S
      SM(2,57)=-R*R
      SM(3,57)=R*S

      DO 50 K=1,8
      DO 40 L=1,3
      P(L,K)=SM(L,ICONSTR(K))
 40   CONTINUE
 50   CONTINUE
      RETURN

      END
C-----------------------------------------------------------------------

C=================================================================FIXP
C
C     A-MATRIX FOR AN ASSUMED STRESS FOUR NODE MEMBRANE ELEMENT
C     WITH DRILLING DEGREES OF FREEDOM - 8 BETA PARAMETERS
C
C-----------------------------------------------------------------------
      SUBROUTINE FIXP (P,PTRANS0,PTRANS,QTRANS,AJM11,AJM12,
     &                 AJM21,AJM22,DETJ,DETJ0,PPURE,IMIXED,I,J)

      IMPLICIT REAL*8 (A-H,O-Z)

      DIMENSION P(I,J), PTRANS0(3,3), PTRANS(3,3), QTRANS(3,3)
      DIMENSION PPURE(I,J)

      DATA  ZERO /0.D0/, ONE /1.D0/, TWO /2.D0/

      GOTO (10,20,30,40,50,60) IMIXED
      STOP 'ILLEGAL MIXED ELEMENT FORMULATION SPECIFIED'

C-----NC (OR I) NOT CONSTRAINED-----------------------------------
```

```
 10    DO 15 K=1,I
       DO 15 L=4,J
       P(K,L)=PPURE(K,L)
 15    CONTINUE
       GOTO 500

C-----EP EQUILIBRIUM CONSTRAINED----------------------------------
 20    DO 25 K=1,I
       DO 25 L=4,J
       C1=PTRANSO(K,1)*PPURE(1,L)+PTRANSO(K,2)*PPURE(2,L)+
      &   PTRANSO(K,3)*PPURE(3,L)
       P(K,L)=C1
 25    CONTINUE
       GOTO 500

C-----OC ORTHOGONAL CONSTRAINED-----------------------------------
 30    DO 35 K=1,I
       DO 35 L=4,J
       C1=PTRANSO(K,1)*PPURE(1,L)+PTRANSO(K,2)*PPURE(2,L)+
      &   PTRANSO(K,3)*PPURE(3,L)
       P(K,L)=C1/(DETJ)
 35    CONTINUE
       GOTO 500

C-----EC ELEMENT OPTIMALLY CONSTRAINED----------------------------
 40    STOP ' 4: NOT IMPLEMENTED'
       GOTO 500

C-----NT NORMALIZED TRANSFORMATION OF STRESS FIELD----------------
 50    DO 55 K=1,I
       DO 55 L=4,J
       CALL FORMTRANSJ  (PTRANS,AJM11,AJM12,AJM21,AJM22)
       C2=PTRANS (K,1)*PPURE(1,L)+PTRANS (K,2)*PPURE(2,L)+
      &   PTRANS (K,3)*PPURE(3,L)
       P(K,L)=C2/(DETJ**2)
 55    CONTINUE
       GOTO 500

C-----PH PHYSICAL COMPONENTS IN ISOPARAMETRIC SPACE---------------
 60    DO 65 K=1,I
       DO 65 L=4,J
       CALL FORMTRANSJQ (QTRANS,AJM11,AJM12,AJM21,AJM22,DETJ)
       C3=QTRANS (K,1)*PPURE(1,L)+QTRANS (K,2)*PPURE(2,L)+
      &   QTRANS (K,3)*PPURE(3,L)
       P(K,L)=C3
 65    CONTINUE
       GOTO 500
```

```
C-----ASSIGN LOWER ORDER STRESS-----------------------------------
 500  DO 550 K=1,3
      DO 550 L=1,3
      P(K,L) = PPURE(K,L)
 550  CONTINUE
      RETURN

      END
C-----------------------------------------------------------------

C=========================================================MATLWINV
C
C     INVERSE MATERIAL LAW FOR MEMBRANE ELEMENTS
C     COMPLIANCE MATRIX
C
C-----------------------------------------------------------------
      SUBROUTINE MATLWINV(NSTAT,EE,UU,C)

      IMPLICIT REAL*8 (A-H,O-Z)

      DIMENSION C(3,3)

      DATA ZERO /0.D0/, ONE /1.D0/, TWO /2.D0/, HALF /0.50D0/

C-----MEMBRANE PLANE STRESS---------------------------------------
      IF (NSTAT.EQ.2) GO TO 20
      ENTRY=ONE/EE
      C(1,1)=ENTRY
      C(2,2)=ENTRY
      C(1,2)=-UU/EE
      C(2,1)=-UU/EE
      C(3,3)=TWO*(ONE+UU)/EE
      GO TO 40

C-----MEMBRANE PLANE STRAIN---------------------------------------
 20   ENTRY=( (ONE+UU)*(ONE-TWO*UU) ) / ( TWO * EE * (UU-HALF) )
      C(1,1)=-ENTRY*(ONE-UU)
      C(2,2)=-ENTRY*(ONE-UU)
      C(1,2)= ENTRY*UU
      C(2,1)= ENTRY*UU
      C(3,3)=-ENTRY*TWO

 40   C(1,3)=ZERO
      C(2,3)=ZERO
      C(3,1)=ZERO
      C(3,2)=ZERO
      RETURN
```

```
      END
C-----------------------------------------------------------------
```

Appendix F

List of definitions

Advanced elements

Finite elements with both drilling degrees of freedom and an assumed stress interpolation field.

Composite materials

Generally a composite material is a material with two or more constituents, combined by physical process on the macroscopic scale.

Dirichlet problem [63]

Boundary value problem:

$$\begin{aligned} \frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2} &= 0 && \in R \\ u(x_1, x_2) &= f(x_1, x_2) && \forall (x_1, x_2) \in C \end{aligned} \tag{F.1}$$

Finding a solution of Laplace's equation in a region R with given boundary values, is called a Dirichlet problem. It is known that, if the boundary curve C and the boundary value function are reasonably well behaved, then there exists a unique solution to the Dirichlet problem in (F.1).

Drilling degrees of freedom

In-plane rotational degrees of freedom of membrane elements.

Finite elements [64]

A geometrically complex domain of a problem is represented as a collection of geometrically simple subdomains, called finite elements.

Interpolation functions [64]

Over each finite element the approximation functions are derived using the basic idea that any continuous function can be represented by a linear combination of algebraic polynomials. The approximation functions are derived using concepts from interpolation theory, and are therefore called interpolation functions.

Low order elements

Finite elements with the least possible number of nodes per element. For example a quadrilateral element with only four nodes.

Orthotropic materials

Materials that posses different stiffnesses along three perpendicular axes.

Strain-displacement relations [65]

In small displacement theory the strain-displacement relations are given as follows:

$$\begin{aligned}
\epsilon_{11} &= \frac{\partial u_1}{\partial x_1} \\
\epsilon_{22} &= \frac{\partial u_2}{\partial x_2} \\
\epsilon_{33} &= \frac{\partial u_3}{\partial x_3} \\
\gamma_{12} &= \frac{\partial u_2}{\partial x_1} + \frac{\partial u_1}{\partial x_2} \\
\gamma_{13} &= \frac{\partial u_3}{\partial x_1} + \frac{\partial u_1}{\partial x_3} \\
\gamma_{23} &= \frac{\partial u_3}{\partial x_2} + \frac{\partial u_2}{\partial x_3}
\end{aligned} \tag{F.2}$$

Stress [65]

The state of internal force at a point of the body is defined by nine components of stress:

$$\begin{matrix}
\sigma_{11} & \tau_{12} & \tau_{13} \\
\tau_{21} & \sigma_{22} & \tau_{23} \\
\tau_{31} & \tau_{32} & \sigma_{33}
\end{matrix} \tag{F.3}$$

which should satisfy the equations of equilibrium:

$$\begin{aligned}
\frac{\partial \sigma_{11}}{\partial x_1} + \frac{\partial \tau_{12}}{\partial x_2} + \frac{\partial \tau_{13}}{\partial x_3} + U_1 &= 0 \\
\frac{\partial \tau_{21}}{\partial x_1} + \frac{\partial \sigma_{22}}{\partial x_2} + \frac{\partial \tau_{23}}{\partial x_3} + U_2 &= 0 \\
\frac{\partial \tau_{31}}{\partial x_1} + \frac{\partial \tau_{32}}{\partial x_2} + \frac{\partial \sigma_{33}}{\partial x_3} + U_3 &= 0
\end{aligned} \tag{F.4}$$

and

$$\begin{aligned}
\tau_{12} &= \tau_{21} \\
\tau_{13} &= \tau_{31} \\
\tau_{23} &= \tau_{32}
\end{aligned} \tag{F.5}$$

where U_1, U_2 and U_3 are components of the body forces per unit volume. By eliminating τ_{21}, τ_{31} and τ_{32} by the use of (F.5), then (F.4) becomes:

$$\begin{aligned}
\frac{\partial \sigma_{11}}{\partial x_1} + \frac{\partial \tau_{12}}{\partial x_2} + \frac{\partial \tau_{13}}{\partial x_3} + U_1 &= 0 \\
\frac{\partial \tau_{12}}{\partial x_1} + \frac{\partial \sigma_{22}}{\partial x_2} + \frac{\partial \tau_{23}}{\partial x_3} + U_2 &= 0 \\
\frac{\partial \tau_{13}}{\partial x_1} + \frac{\partial \tau_{23}}{\partial x_2} + \frac{\partial \sigma_{33}}{\partial x_3} + U_3 &= 0
\end{aligned} \tag{F.6}$$

Strain [65]

The state of strain at a point of the body is defined by six components of strain, namely:

$$\epsilon_{11},\ \epsilon_{22},\ \epsilon_{33},\ \gamma_{12},\ \gamma_{13} \text{ and } \gamma_{23}. \tag{F.7}$$

Sobolev spaces of functions [38]

Sobolev spaces of functions are defined as follows:

$$H^k = H^k(\Omega) = \{w | w \in L_2; w_{,x} \in L_2; ...; w_{,x...x} \in L_2\} \tag{F.8}$$

where

$$L_2 = L_2(\Omega) = \{w | \int_0^1 w^2 \, \mathrm{d}x < \infty\} \tag{F.9}$$

In words, the Sobolev space of degree k, denoted H^k, consists of functions that possess square-integrable generalized derivatives through order k. A square-integrable function is called an L_2-function, by virtue of (F.9). From (F.8), it can be seen that:

$$H^0 = L_2 \tag{F.10}$$

and that

$$H^{k+1} \subset H^k \tag{F.11}$$

Variational form [64]

In the variational solution of differential equations, the differential equation is put into an equivalent variational form, and then the approximate solution is assumed to be a combination, $\sum c_j \phi_j$, of given approximation functions ϕ_j. The parameters c_j are determined from the variational form. The variational methods suffer from the disadvantage that the approximation functions for problems with arbitrary domains are difficult to construct.