ภาคผนวก ข

โปรแกรมคอมพิวเตอร์

ข.1 ตัวแปรที่สำคัญในโปรแกรมคอมพิวเตอร์

EF( )	=	เวกเตอร์ของแรงที่กระทำต่อชิ้นส่วนย่อย
EI	=	ผลคูณของโมดูลัสความยืดหยุ่นกับโมเมนต์ความเฉื่อย
EK(,)	=	สติฟเนสเมตริกซ์ของชิ้นส่วนย่อย
H	=	ความยาวของชิ้นส่วนย่อยตามส่วนโค้ง
IB	=	เงื่อนไขขอบเขตของจุดรองรับของชิ้นส่วนย่อย
ITMAX	=	จำนวนครั้งของการกระทำซ้ำมากที่สุดที่กำหนด
MBAND	=	จำนวนแถวของแบนต์เมตริกซ์
NDF	=	จำนวนดีกรีอิสระต่อจุดขั้ว
NEL	=	ลำดับที่ของชิ้นส่วนย่อย
NELEM	=	จำนวนชิ้นส่วนย่อย
NODE	=	ลำดับที่ของจุดขั้ว
NODI( )	=	ลำดับที่ของจุดขั้วด้านซ้ายของชิ้นส่วนย่อย
NODJ( )	=	ลำดับที่ของจุดขั้วด้านขวาของชิ้นส่วนย่อย
NUMDF	=	จำนวนดีกรีอิสระของระบบรวม
NUMNP	=	จำนวนจุดขั้วของระบบรวม
Q( )	=	ดีกรีอิสระของชิ้นส่วนย่อย
QO( )	=	ดีกรีอิสระของระบบรวม
SH( )	=	แรงเฉือนที่เกิดขึ้น
SK(,)	=	สติฟเนสเมตริกซ์แบบไม่เป็นเชิงเส้นของระบบรวม
SL	=	ระยะช่วงพาด
ST	=	ความยาวส่วนโค้ง
TF( )	=	เวกเตอร์ของแรงที่กระทำต่อท่อ
TH	=	แรงดึงตามแนวราบที่ปลายบน
TOLER	=	ค่าความคลาดเคลื่อนที่กำหนด
WT( )	=	ค่าสัมประสิทธิ์ถ่วงน้ำหนักสำหรับจุดในการอินทิเกรตของ Gauss
XH	=	ระยะระหว่างจุดรองรับในแนวราบ
XT( )	=	ค่าของตำแหน่งของจุดในการอินทิเกรทของGauss
YH	=	ระยะระหว่างจุดรองรับในแนวดิ่ง

ข.2 โปรแกรมคอมพิวเตอร์

```
C       Program "LexcableR.for"
C       Finite element computer program for static equilibrium analysis of
C    cables with axial deformations using Lagrangian coordinates and
C    cubic polynomials shape functions
C    Revised 16 October 2014
        IMPLICIT REAL*8(A-H,O-Z)
    COMMON/PASS1/FTOP,W,C,ST,H,CK,YH
    COMMON/BLKN/NEL
    COMMON/SIN2/NDFD,NBW,NUMNPD
    COMMON/COM2/NDF,NUMNP,NELEM
    COMMON/CINT/XT(4),WT(4)
    DIMENSION SK(200,4),EK(4,4),KON(200),QO(200),Q(4),KK(4),NODI(100)
    DIMENSION NODJ(100),TF(200),EF(4),P(4),R(4),S(4),IB(6),DQO(200)
    DIMENSION EPN(50),EPN1(50),EPN2(50),TENODE(50),SX(200)
        DIMENSION DYSND(200),DYSEL(200)
        CHARACTER*20 OUT
 901    FORMAT(A)
    OPEN(UNIT=5,FILE='LexcableR_input',STATUS='UNKNOWN')
        WRITE(*,902)
 902 FORMAT(' OUTPUT FILE NAME : ',\)
    READ(*,901) OUT
        OPEN(6,FILE=OUT,STATUS='NEW')
        READ(5,*) LSTEP
    READ(5,*) NELEM,ITMAX,TOLER,YH,ST,FTOP,W,DO,DI,E
        WRITE(6,2001)
2001    FORMAT(10X,'OUTPUT OF EXTENSIBLE CABLES PROGRAM "LexcableR.for"'/)
    WRITE(6,2002) NELEM,ITMAX,TOLER,YH,ST,FTOP,W,DO,DI,E
2002 FORMAT(10X,'NUMBER OF ELEMENTS          ',I5,/
   *    10X,'NUMBER OF ITERATIONS        ',I5,/
```

```
*      10X,'TOLERANCE                ',F12.6,/
*      10X,'SUPPORT ELEVATION        ',F12.6,/
*      10X,'TOTAL ARC LENGTH         ',F12.6,5X,'M'/
*      10X,'TENSION                  ',F12.6,5X,'N'/
*      10X,'DISTRIBUTED LOAD         ',F12.6,5X,'N/M'/
*      10X,'OUTSIDE DIAMETER         ',F12.6,5X,'M'/
*      10X,'INSIDE DIAMETER          ',F12.6,5X,'M'/
*      10X,'ELASTIC MODULUS          ',E12.6,5X,'MPa'/)
      PI=3.1415926
      GRAV=9.81
      AO=PI*DO**2/4.
      AI=PI*DI**2/4.
      AS=(AO-AI)*1000000.
      EA=E*AS
      NDF=2
      MBAND=4
      NUMNP=NELEM+1
      NUMDF=NUMNP*NDF
      H=ST/FLOAT(NELEM)
      DS=H/2.
      DO 100 J=1,NELEM
      N1=2*(J-1)
      KON(N1+1)=J
      KON(N1+2)=J+1
      NODI(J)=J
      NODJ(J)=J+1
  100 CONTINUE
      DO 25 I=1,2
      L1=(NDF+1)*(I-1)+1
      IB(L1)=NELEM*(I-1)+1
      IB(L1+1)=0
```

```
      IB(L1+2)=1
   25 CONTINUE
C
      CALL SETINT
C
            DO 55 I=1,NUMDF
   55       QO(I)=0.
      DO 54 I=1,NUMNP
            EPN(I)=0.
            EPN1(I)=0.
            EPN2(I)=0.
      TENODE(I)=0.
   54       CONTINUE
C
            IF(LSTEP.EQ.1) THEN
            GOTO 57
            END IF
C
   56       READ(5,*,END=3000) YH,ST,FTOP,W,DO,DI,E
C
   57 DO 50 ITER=1,ITMAX
      DO 10 I=1,NUMDF
      TF(I)=0.
      DO 10 J=1,MBAND
   10 SK(I,J)=0.
            IF(ITER.EQ.1) THEN
            CK=0.
            DYS=0.
            ELSE
            CK=YH/SL
            END IF
```

```
      DO 1000 NEL=1,NELEM
      CALL ELEQ(KON,QO,Q)
      DO 20 I=1,4
      EF(I)=0.
      DO 20 J=1,4
   20 EK(I,J)=0.
      DO 30 L=1,4
      CALL SHAPE1(XT(L),P,R,S,DS,Q,YA,DYA,DYAM1,DYS,DYSM1,EPN,EPS)
      DO 40 I=1,4
      EF(I)=EF(I)-FTOP*WT(L)*R(I)*DYS/DYSM1**0.5*DS
     *          -W*WT(L)*P(I)*DS
      DO 40 J=1,4
      EK(I,J)=EK(I,J)+FTOP*WT(L)*R(I)*R(J)*(1+EPS)**2/DYSM1**1.5*DS
   40 CONTINUE
   30 CONTINUE
C
            DYSEL(NEL)=DYS
C
      KK(2)=2*NODI(NEL)
      KK(4)=2*NODJ(NEL)
      KK(1)=KK(2)-1
      KK(3)=KK(4)-1
      DO 400 I=1,4
      II=KK(I)
      TF(II)=TF(II)-EF(I)
      DO 400 J=1,4
      IF(KK(J).LT.II) GOTO 400
      JJ=KK(J)-II+1
      SK(II,JJ)=SK(II,JJ)+EK(I,J)
  400 CONTINUE
 1000 CONTINUE
```

```
C
C  INSERT BOUNDARY CONDITIONS
C
      CALL BOUND(SK,TF,IB,NUMDF,MBAND)
C
C  SOLVE SIMULTENEOUS EQUATIONS
C
      CALL BANSOL(SK,TF,NUMDF,MBAND,200,4)
C
C  INCREMENTAL PROCESS
C
      SUM=0.
      DO 70 I=1,NUMDF
      TEMP=QO(I)
      DQO(I)=TF(I)
      QO(I)=QO(I)+DQO(I)
      DIFF=ABS(TEMP-QO(I))
      SUM=SUM+DIFF
   70 CONTINUE
C
C  COMPUTE UNSTRAINED AND STRAINED SPAN LENGTH
C       SG=unstrained span element length
C            SL=unstrained span length
C            SG0=strained span element length
C            SL0=strained span length
C
      SL=0.
           SLO=0.
      DO 1010 NEL=1,NELEM
      CALL ELEQ(KON,QO,Q)
      SG=0.
```

```
          SGO=0.
      DO 60 L=1,4
      CALL SHAPE1(XT(L),P,R,S,DS,Q,YA,DYA,DYAM1,DYS,DYSM1,EPN,EPS)
          SG=SG+WT(L)*(1.-DYS**2)**0.5*DS/(1.+EPS)
      SGO=SGO+WT(L)*DYSM1**0.5*DS
   60     CONTINUE
      SL=SL+SG
          SLO=SLO+SGO
          ND=NEL+1
          SX(1)=0.
          SX(ND)=SL
          IF(NEL.EQ.1)THEN
          SII=0.D0
      CALL SHAPE1(SII,P,R,S,DS,Q,YA,DYA,DYAM1,DYS,DYSM1,EPN,EPS)
C     DYSND(1)=DYS
C           UPDATE DYS
          DYSND(1)=CK*(DYSM1)**0.5-QO(2)
C           UPDATE DYS  NO2
          DYSND(1)=CK*(1.-DYSND(1)**2)**0.5-QO(2)
C           UPDATE DYS  NO3
          DYSND(1)=CK*(1.-DYSND(1)**2)**0.5-QO(2)
C           UPDATE DYS  NO4
          DYSND(1)=CK*(1.-DYSND(1)**2)**0.5-QO(2)
C
          SII=1.D0
      CALL SHAPE1(SII,P,R,S,DS,Q,YA,DYA,DYAM1,DYS,DYSM1,EPN,EPS)
          DYSND(2)=DYS
          DYSND(2)=CK*(1.-DYSND(2)**2)**0.5-Q(4)
          ELSE
          SII=1.D0
      CALL SHAPE1(SII,P,R,S,DS,Q,YA,DYA,DYAM1,DYS,DYSM1,EPN,EPS)
```

```
      DYSND(ND)=DYS
            END IF
 1010 CONTINUE
C
C  COMPUTE AXIAL STRAIN USING NEWTION'S METHOD
C
      SUM1=0.
      DO 500 NODE=1,NUMNP
      ND=2*NODE-1
      CALL SHAPE1(XT(L),P,R,S,DS,Q,YA,DYA,DYAM1,DYS,DYSM1,EPN,EPS)
C
      FTOPL=FTOP/COS(ASIN(DYSND(NUMNP)))
      EAEP=EA*EPN(NODE)
            FEPN=EAEP-FTOPL+W*(YH-(CK*SX(NODE)-QO(ND)))/(1.+EPN(NODE))
            DFEPN=EA-W*(YH-(CK*SX(NODE)-QO(ND)))/(1.+EPN(NODE))**2
            EPN2(NODE)=EPN1(NODE)-FEPN/DFEPN
            DIFF1=ABS(EPN2(NODE)-EPN1(NODE))
            SUM1=SUM1+DIFF1
            EPN1(NODE)=EPN2(NODE)
            EPN(NODE)=EPN2(NODE)
C           TENODE(NODE)=EA*EPN(NODE)
C           TENODE(NODE)=FTOPL-W*(YH-(TAN*SX(NODE)-QO(ND)))/(1.+EPN(NODE))
            TENODE(NODE)=FTOPL-W*(YH-(CK*SX(NODE)-QO(ND)))/(1.+EPN(NODE))
  500       CONTINUE

C  TEST FOR CONVERGENCE
C
            IF(ABS(SUM).LE.TOLER) GOTO 90
C     IF(ABS(SUM).LE.TOLER .AND. ABS(SUM1).LE.TOLER) GOTO 90
      WRITE(6,*)
      WRITE(6,107) ITER
```

```
  107 FORMAT(/10X,'NUMBER OF ITERATIONS        ',I5)
C
C  PRINT OUT THE RESULTS
C
      CALL RESULT(QO,SL,SLO,SX,EPN,TENODE,FTOPL,DYSND)
   50 CONTINUE
      WRITE(*,108)
  108 FORMAT(10X,'NO CONVERGENCE')
   90 CONTINUE
C
      GOTO 56
 3000 STOP
      END


      SUBROUTINE SETINT
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/CINT/XI(4),W(4)
      XI(1)=-0.861136311594053D0
      XI(2)=-0.339981043584856D0
      XI(3)=-XI(2)
      XI(4)=-XI(1)
      W(1)=0.347854845137453D0
      W(2)=0.652145154862546D0
      W(3)=W(2)
      W(4)=W(1)
      RETURN
      END


      SUBROUTINE ELEQ(KON,QO,Q)
```

```
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION KON(200),QO(200),Q(4)
      COMMON/PASS1/FTOP,W,C,ST,H,CK,YH
      COMMON/BLKN/NEL
      L=2*(NEL-1)
      N1=KON(L+1)
      N2=KON(L+2)
      K1=2*(N1-1)
      K2=2*(N2-1)
      DO 2 I=1,2
      J1=K1+I
      J2=K2+I
      Q(I)=QO(J1)
    2 Q(I+2)=QO(J2)
      RETURN
      END


      SUBROUTINE SHAPE1(SI,P,R,S,DS,Q,YA,DYA,DYAM1,DYS,DYSM1,EPN,EPS)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/PASS1/FTOP,W,C,ST,H,CK,YH
      COMMON/BLKN/NEL
      DIMENSION P(4),R(4),S(4),Q(4)
      DIMENSION PE(2),QE(2),EPN(50)
      XI=(1.+SI)*DS
      P(1)=1.-3.*XI**2/H**2+2.*XI**3/H**3
      P(2)=XI*(1.-2.*XI/H+XI**2/H**2)
      P(3)=3.*XI**2/H**2-2.*XI**3/H**3
      P(4)=XI*(-XI/H+XI**2/H**2)
      R(1)=(-6.*XI/H+6.*XI**2/H**2)/H
      R(2)=1.-4.*XI/H+3.*XI**2/H**2
```

```
R(3)=(6.*XI/H-6.*XI**2/H**2)/H
R(4)=-2.*XI/H+3.*XI**2/H**2
S(1)=-6./H**2+12.*XI/H**3
S(2)=(-4.+6.*XI/H)/H
S(3)=6./H**2-12.*XI/H**3
S(4)=(-2.+6.*XI/H)/H
QE(1)=EPN(NEL)
QE(2)=EPN(NEL+1)
PE(1)=1.-XI/H
PE(2)=XI/H
EPS=PE(1)*QE(1)+PE(2)*QE(2)
YA=P(1)*Q(1)+P(2)*Q(2)+P(3)*Q(3)+P(4)*Q(4)
DYA=R(1)*Q(1)+R(2)*Q(2)+R(3)*Q(3)+R(4)*Q(4)
DYAM1=(1.+EPS)**2-DYA**2
      DYS1=CK*SQRT(1.-DYS**2)-DYA
      DYSM=1.-DYS1**2.
      FDYS=DYS1-CK*SQRT(DYSM)+DYA
      DFDYS=1.+CK*DYS1/SQRT(DYSM)
      DYS=DYS1-FDYS/DFDYS
      DYSM1=1.-DYS**2
RETURN
END


SUBROUTINE BOUND(SK,TF,IB,NUMDF,MBAND)
IMPLICIT REAL*8(A-H,O-Z)
COMMON/COM2/NDF,NUMNP,NELEM
DIMENSION SK(200,4),TF(200),IB(6)
DO 100 L=1,2
L1=(NDF+1)*(L-1)+1
NO=IB(L1)
```

```
      K1=NDF*(NO-1)
      DO 90 I=1,NDF
      L2=L1+I
      IF(IB(L2)) 90,10,90
   10 KR=K1+I
      DO 50 J=2,MBAND
      KV=KR+J-1
      IF(NUMDF-KV) 30,20,20
   20 SK(KR,J)=0.
   30 KV=KR-J+1
      IF(KV) 50,50,40
   40 SK(KV,J)=0.
   50 CONTINUE
      SK(KR,1)=1.
      TF(KR)=0.
   90 CONTINUE
  100 CONTINUE
      RETURN
      END


      SUBROUTINE BANSOL(S,R,NSIZE,MBAND,NDIM,MDIM)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION S(NDIM,MDIM),R(NDIM)
  700 DO 790 N=1,NSIZE
      DO 780 L=2,MBAND
      IF(S(N,L).EQ.0.) GOTO 780
      I=N+L-1
      C=S(N,L)/S(N,1)
      J=0
      DO 750 K=L,MBAND
```

```
   J=J+1
750 S(I,J)=S(I,J)-C*S(N,K)
   S(N,L)=C
780 CONTINUE
790 CONTINUE
800 DO 830 N=1,NSIZE
   DO 820 L=2,MBAND
   IF(S(N,L).EQ.0.) GOTO 820
   I=N+L-1
   R(I)=R(I)-S(N,L)*R(N)
820 CONTINUE
830 R(N)=R(N)/S(N,1)
   DO 860 M=2,NSIZE
   N=NSIZE+1-M
   DO 850 L=2,MBAND
   IF(S(N,L).EQ.0.) GOTO 850
   K=N+L-1
   R(N)=R(N)-S(N,L)*R(K)
850 CONTINUE
860 CONTINUE
   RETURN
   END


   SUBROUTINE RESULT(QO,SL,SLO,SX,EPN,TNODE,FTOPL,DYSND)
   IMPLICIT REAL*8(A-H,O-Z)
   COMMON/PASS1/FTOP,W,C,ST,H,CK,YH
   COMMON/COM2/NDF,NUMNP,NELEM
   COMMON/BLKN/NEL
   DIMENSION X(200),QO(200),EPN(50),TNODE(50),SX(200),DYSND(200)
   WRITE(6,110) FTOPL,FTOP,YH,SL,SLO
```

```
110 FORMAT(10X,'TOTAL TENSION(FTOPL)        ',F16.4,5X,'N'/
  *         10X,'HORIZONTAL TENSION(FTOP)     ',F16.4,5X,'N'/
         *       10X,'SUPPORT ELEVATION           ',F16.4,5X,'M'/
  *       10X,'UNSTRAINED CABLE SPAN LENGTH ',F16.4,5X,'M'/
  *         10X,'STRAINED CABLE SPAN LENGTH   ',F16.4,5X,'M'/)
         CK=YH/SL
         SK=YH/SQRT(SL**2+YH**2)
   WRITE(6,111)
   DO 80 L=1,NUMNP
   NL=NUMNP-L
   NL1=NL+1
   DTS=NL*H
   K1=NDF*NL+1
   K2=K1+1
   X(K1)=CK*SX(NL1)-QO(K1)
   SLOPE=ASIN(SK)-ASIN(QO(K2))
         X(K2)=SIN(SLOPE)
 80 WRITE(6,112) NL1,DTS,SX(NL1),(X(J),J=K1,K2),DYSND(NL1),EPN(NL1),
  *TNODE(NL1),TNODE(NL1)*COS(ASIN(DYSND(NL1)))
111 FORMAT(10X,'NODE',5X,'ARC-LENGTH',5X,'COOR. X',3X,'DISPL Y',8X,
  *'DY/DS',7X,'DYSND',7X,'STRAIN',7X,'TENSION(N)',8X,'TH'/)
112 FORMAT(7X,I5,5X,6F12.6,5X,F12.6,5X,F12.6)
   RETURN
   END
```