

5.6 弾性モデル

弾性部材における3次元部材の剛性行列 $[K_L]$ は、以下のように表される。これについては、理論マニュアルを参照されたい。

$$[K_L] = \begin{bmatrix} \frac{EA}{\ell} & 0 & 0 & 0 & 0 & 0 & -\frac{EA}{\ell} & 0 & 0 & 0 & 0 & 0 \\ & \frac{12EI_z}{\ell^3} & 0 & 0 & 0 & \frac{6EI_z}{\ell^2} & 0 & -\frac{12EI_z}{\ell^3} & 0 & 0 & 0 & \frac{6EI_z}{\ell^2} \\ & & \frac{12EI_y}{\ell^3} & 0 & -\frac{6EI_y}{\ell^2} & 0 & 0 & 0 & -\frac{12EI_y}{\ell^3} & 0 & -\frac{6EI_y}{\ell^2} & 0 \\ & & & \frac{GI_x}{\ell} & 0 & 0 & 0 & 0 & 0 & -\frac{GI_x}{\ell} & 0 & 0 \\ & & & & \frac{4EI_y}{\ell} & 0 & 0 & 0 & \frac{6EI_y}{\ell^2} & 0 & \frac{2EI_y}{\ell} & 0 \\ & & & & & \frac{4EI_z}{\ell} & 0 & -\frac{6EI_z}{\ell^2} & 0 & 0 & 0 & \frac{2EI_z}{\ell} \\ & & & & & & \frac{EA}{\ell} & 0 & 0 & 0 & 0 & 0 \\ & & & & & & & \frac{12EI_z}{\ell^3} & 0 & 0 & 0 & -\frac{6EI_z}{\ell^2} \\ & & & & & & & & \frac{12EI_y}{\ell^3} & 0 & \frac{6EI_y}{\ell^2} & 0 \\ & & & & & & & & & \frac{GI_x}{\ell} & 0 & 0 \\ & & & & & & & & & & \frac{4EI_y}{\ell} & 0 \\ & & & & & & & & & & & \frac{4EI_z}{\ell} \end{bmatrix}$$

.....(5.9)

この弾性剛性を計算するためのサブルーチンコールは、

```

C                                     Model_No.1 通常の有限要素弾性モデル
call Cal_lin_stiff_M1(Member(i),Element(ie),ak_linear(1,1,i))
  if(Member(i).i_rigid_length.ne.0..or.Member(i).j_rigid_length.ne.0.)
*   call Deal_Rigid_element(ak_linear(1,1,i),
*       Member(i).i_rigid_length,Member(i).j_rigid_length)

```

である。また、後のサブルーチンコールは、この部材が両端もしくはどちらかに剛域を有する場合に対応し、そこでは剛性を座標変換する。上記2つに関連するサブルーチンを以下に示す。

```

C
C   SUBROUTINE /Cal_lin_stiff_M1
C

```

```

C      線形剛性行列の計算(ok)
C
      subroutine Cal_lin_stiff_M1(Member,Element,ak_linear)
      implicit real*8(A-H,O-Z)
      include "submain.h"
      record / member_s      / Member
      record / element_s     / Element
      dimension ak_linear(12,12)

C
C      Model_No.1 = 1          ! 通常の有限要素弾塑性モデル
C
C      要素数
C      structure / element_s/
C      integer element_type    ! 要素タイプ
C      integer n_element      ! 部材モデルの数
C      real*8 E                ! ヤング係数
C      real*8 G                ! せん断係数
C      real*8 A                ! 断面積
C      real*8 RIx              ! ねじり剛性
C      real*8 RIy              ! y 軸断面二次モーメント
C      real*8 RIz              ! z 軸断面二次モーメント
C      real*8 ASy              ! 各部材のY 軸回りのせん断変形用等価断面積
C      real*8 ASz              ! 各部材のZ 軸回りのせん断変形用等価断面積
C      real*8 AM               ! 単位長さ当たりの質量
C      integer nm_damp         ! 部材減衰の有無
C      end structure
C      record /element_s/ Element
C      ALLOCATABLE ::Element(:)
C      ALLOCATE (Element(n_element))
C
C      部材数
C      structure / member_s/
C      integer nm_element      ! 要素番号
C      integer element_type    ! 要素タイプ
C      integer n_element_type  ! 要素タイプ別番号
C      integer nm_dll_element  ! DLL を用いた要素か ( 0 ; システム内要素、 1 : DLL 要素 )
C      integer nm_point(2)     ! 節点番号
C      integer irest(12)       ! 部材両端の自由度番号表
C      integer ijp(2)          ! 両端節点への結合状況 (0:剛結合 1:ピン結合)
C      integer nm_analysis     ! 部材解析種別
C      integer nm_group        ! 部材グループ
C      integer nm_local_coord(2) ! 局所座標系の有無とその回転行列の番号
C      integer nm_damp         ! 部材減衰の有無とその減衰行列の番号
C      real*8 alength          ! 長さ
C      real*8 rot_x            ! 部材主軸の回転角度 ( 度 )
C      real*8 force(12)        ! 部材両端の部材端力
C      end structure
C      record / member_s / Member
C      ak_linear      real*8 線形剛性行列
C
      do i=1,12
      do j=1,12
      ak_linear(j,i) = 0.0
      end do

```

```

end do
rl=Member.alength -
* Member.i_rigid_length - Member.j_rigid_length
young = Element.E
ra    = Element.A
riz   = Element.RIz
riy   = Element.RIy
rix   = Element.RIx
seng  = Element.G
IP    = Member.ijp(1)      ! i 端ピン接合か（現在使用不可）
JP    = Member.ijp(2)      ! j 端ピン接合か（現在使用不可）
asy   = Element.ASy
asz   = Element.ASz

C
C   -   立体構造部材の部材座標系での部材剛性マトリックスを作成する。
C   -
C   -   RA       : 各部材の断面積。
C   -   RIX      : 各部材の X 軸回りの極 2 次モーメント。
C   -   RIY      : 各部材の Y 軸回りの断面 2 次モーメント。
C   -   RIZ      : 各部材の Z 軸回りの断面 2 次モーメント。
C   -   YOUNG    : 各部材のヤング係数（ヤング率）。
C   -   SENG     : 各部材のせん断剛性係数。
C   -   ASY      : 各部材の Y 軸回りのせん断変形用等価断面積。
C   -   ASZ      : 各部材の Z 軸回りのせん断変形用等価断面積。
C   -   JCON     : 各部材の部材端拘束表。
C   -   RL       : 各部材の部材長。
C   -   ak_linear : 部材剛性マトリックス。
C
ak_linear(1,1)= YOUNG*RA/RL
ak_linear(1,7)=-ak_linear(1,1)
ak_linear(7,7)=ak_linear(1,1)
IF(IP.EQ.0.AND.JP.EQ.0) GOTO 40          ! 両端ピン接合処理へ
RL2=RL*RL
RL3=RL*RL*RL
IF(ASY.NE.0.0.OR.ASZ.NE.0.0) GOTO 20     ! せん断変形を考慮する処理へ
ZERL2= 6.*YOUNG*RIZ/RL2
ZERL3= 2.*ZERL2/RL
ak_linear(2,2) = ZERL3
ak_linear(2,6) = ZERL2
ak_linear(2,8) =-ZERL3
ak_linear(2,12)= ZERL2
YERL2= 6.*YOUNG*RIY/RL2
YERL3= 2.*YERL2/RL
ak_linear(3,3) = YERL3
ak_linear(3,5) =-YERL2
ak_linear(3,9) =-YERL3
ak_linear(3,11)=-YERL2
ak_linear(4,4) = SENG*RIX/RL
ak_linear(4,10)=-ak_linear(4,4)
ak_linear(5,5) = 4.0*YOUNG*RIY/RL
ak_linear(5,9) = YERL2
ak_linear(5,11)= ak_linear(5,5)*0.5
ak_linear(6,6) = 4.0*YOUNG*RIZ/RL
ak_linear(6,8) =-ZERL2

```

```

      ak_linear(6,12)= ak_linear(6,6)*0.5
      ak_linear(8,8)=  ZERL3
      ak_linear(8,12)=-ZERL2
      ak_linear(9,9)=  YERL3
      ak_linear(9,11)= YERL2
      ak_linear(10,10)= ak_linear(4,4)
      ak_linear(11,11)= ak_linear(5,5)
      ak_linear(12,12)= ak_linear(6,6)
      DO  I=1,11
      DO  J=i+1,12
      ak_linear(J,I) =ak_linear(I,J)
      enddo
      enddo
      GOTO 30
C
C  --- せん断変形を考慮  -----
C
20 CONTINUE
  FY=0.0
  FZ=0.0
  IF(ASY.NE.0.0) FY=12.*YOUNG*RIZ/(SENG*ASY*RL2)
  IF(ASZ.NE.0.0) FZ=12.*YOUNG*RIY/(SENG*ASZ*RL2)
  ZERL3=12.*YOUNG*RIZ/(RL3*(1.+FY))
  ZERL2= 6.*YOUNG*RIZ/(RL2*(1.+FY))
  ak_linear(2,2) = ZERL3
  ak_linear(2,6) = ZERL2
  ak_linear(2,8) =-ZERL3
  ak_linear(2,12)= ZERL2
  YERL3=12.*YOUNG*RIY/(RL3*(1.+FZ))
  YERL2= 6.*YOUNG*RIY/(RL2*(1.+FZ))
  ak_linear(3,3) = YERL3
  ak_linear(3,5) =-YERL2
  ak_linear(3,9) =-YERL3
  ak_linear(3,11)=-YERL2
  ak_linear(4,4) = SENG*RIX/RL
  ak_linear(4,10)=-ak_linear(4,4)
  ak_linear(5,5) = (4.0+FZ)*YOUNG*RIY/(RL*(1.+FZ))
  ak_linear(5,9) = YERL2
  ak_linear(5,11)= (2.0-FZ)*YOUNG*RIY/(RL*(1.+FZ))
  ak_linear(6,6) = (4.0+FY)*YOUNG*RIZ/(RL*(1.+FY))
  ak_linear(6,8) =-ZERL2
  ak_linear(6,12)= (2.0-FY)*YOUNG*RIZ/(RL*(1.+FY))
  ak_linear(8,8)=  ZERL3
  ak_linear(8,12)=-ZERL2
  ak_linear(9,9)=  YERL3
  ak_linear(9,11)= YERL2
  ak_linear(10,10)= ak_linear(4,4)
  ak_linear(11,11)= ak_linear(5,5)
  ak_linear(12,12)= ak_linear(6,6)
  DO  I=1,11
  DO  J=i+1,12
  ak_linear(J,I) =ak_linear(I,J)
  enddo
  enddo

```

```

C
C   --- ピン結合処理 -----
C
30 CONTINUE
   IF(IP.EQ.1.AND.JP.EQ.1) GOTO 31 ! 両端剛接合の場合処理終了
   KP=1
   IF(JP.EQ.0) KP=2
   CALL SPPIN(KP,ak_linear)      ! 部材の片方がピン接合の場合の処理
31 CONTINUE
   RETURN

C
C   --- トラス部材の剛性 -----
C
40 CONTINUE
   ak_linear(7,7) = ak_linear(1,1)
   ak_linear(7,1) = -ak_linear(1,1)
   RETURN
END

C
C   SUBROUTINE /SPPIN
C
C   一端ピン部材剛性行列変換
C
SUBROUTINE SPPIN(KP,ak_linear)
  IMPLICIT REAL*8(A-H,O-Z)
  DIMENSION ak_linear(12,12),MM(2,2),S(12,2),A(2)
  DATA MM/5,6,11,12/

C
C   - ak_linear      : 部材剛性マトリックス。
C
   ak_linear(4,4) =0.0
   ak_linear(4,10) =0.0
   ak_linear(10,4) =0.0
   ak_linear(10,10)=0.0
   DO 10 MX=1,2
     M=MM(MX,KP)
     A(MX)=0.0
     IF(ak_linear(M,M).EQ.0.0) GOTO 10
     A(MX)=1.0/ak_linear(M,M)
     DO 11 I=1,12
       S(I,MX)=ak_linear(I,M)
11 CONTINUE
10 CONTINUE
     DO 20 MX=1,2
       M=MM(MX,KP)
       DO 20 I=1,12
         ak_linear(I,M)=0.0
         ak_linear(M,I)=0.0
20 CONTINUE
     DO 30 MX=1,2
       M=MM(MX,KP)
       IF(A(MX).EQ.0.0) GOTO 30
       AA=A(MX)
       S(M,MX)=0.0

```

```

DO 31 I=1,12
DO 31 J=1,12
ak_linear(I,J)=ak_linear(I,J)-AA*S(I,MX)*S(J,MX)
31 CONTINUE
30 CONTINUE
return
end

C
C      SUBROUTINE /Deal_Rigid_element
C
C      剛域のための剛性行列座標変換(ok)
C
C
C      部材に剛域を有する場合は、このルーチンを使用する。
C      剛域を含む部材の座標系は、
C
C      =====
C
C      |----- Lg -----|
C      |----- L -----|
C
C      部材端部の変位(U,V, )及び節点力(Px,Py,Mz)と
C      剛域内部点(u,v, )の変位及び節点力(px,py,mz)の関係は、
C      幾何学的条件と力の釣合より
C
C      U = u          Px = px
C      V = v - Lg      Py = py
C      =              Mz = mz + Lg Py
C
C      (1)
C
C      あるいは、
C
C      u = U          px = Px
C      v = V + Lg      py = Py
C      =              mz = Mz - LgPy
C
C      (2)
C
C      ここで、Lg は剛域長さである。上式を行列で表すと、
C
C      | 1 0 0 |      | 1 0 0 |
C      Ru = | 0 1 Lg |      Rf = | 0 1 0 |
C      | 0 0 1 |      | 0 -Lg 1 |
C
C      (3)
C
C      u = Ru U      p = Rf P
C
C      (4)
C
C      u = (u,v, )      U = (U,V, )
C      p = (px,py,mz)      P = (Px,Py,Mz)
C
C      (5)
C
C      で表される。次に、剛域内部端の変位と節点力を用いた力の釣合は、
C
C      p = K u
C
C      (6)
C
C      で表されるものとする。式(4)を用いて、式(6)を変換する。
C
C      RfP = K Ru U
C
C      (7)

```

C さらに、行列 Rf の逆行列を両辺の左より掛けると、上式は、以下ようになる。

C

$$C \quad P = Rfb \ K \ Ru \ U \quad (8)$$

C

C 行列 Rf の逆行列 Rfb は、

C

$$C \quad Rfb = \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & Lg & 1 \end{vmatrix} = Ru \quad (9)$$

C

C であるため、部材両端で表す剛性行列は、以下のように表される。

C

$$C \quad K = \begin{vmatrix} & T \\ Ru & K \end{vmatrix} Ru$$

C

C

```

subroutine Deal_Rigid_element(ak,rigid_i,rigid_j)
implicit real*8(A-H,O-Z)
dimension ak(12,12),akk(12,12),r1(6,6),r2(6,6)
data r1/1.,0.,0.,0.,0.,0.,
*      0.,1.,0.,0.,0.,0.,
*      0.,0.,1.,0.,0.,0.,
*      0.,0.,0.,1.,0.,0.,
*      0.,0.,0.,0.,1.,0.,
*      0.,0.,0.,0.,0.,1./
data r2/1.,0.,0.,0.,0.,0.,
*      0.,1.,0.,0.,0.,0.,
*      0.,0.,1.,0.,0.,0.,
*      0.,0.,0.,1.,0.,0.,
*      0.,0.,0.,0.,1.,0.,
*      0.,0.,0.,0.,0.,1./
if(rigid_i.eq.0.and.rigid_j.eq.0.) return
if(rigid_i.ne.0.) then
r1(2,6)=rigid_i
r1(3,5)=-rigid_i
else
r1(2,6)=0.
r1(3,5)=0.
endif
if(rigid_j.ne.0.) then
r2(2,6)=-rigid_j
r2(3,5)=rigid_j
else
r2(2,6)=0.
r2(3,5)=0.
endif
do i=1,12
do j=1,12
akk(i,j)=ak(i,j)
enddo
enddo
call Rotate_K6(akk,r1,r2,ak)
return
end

```

```

C
C      SUBROUTINE /Deal_Rigid_element_v
C
C      剛域のための変位座標変換(部材端から剛域内部点)
C
C      u = U
C      v = V + Lg  z
C      w = W - Lg  y
C
      subroutine Deal_Rigid_element_v(vv,rigid_i,rigid_j)
      implicit real*8(A-H,O-Z)
      dimension vv(12)
      if(rigid_i.eq.0..and.rigid_j.eq.0.) return
      if(rigid_i.ne.0.) then
        vv(2)=vv(2)+rigid_i*vv(6)
        vv(3)=vv(3)-rigid_i*vv(5)
      endif
      if(rigid_j.ne.0.) then
        vv(8)=vv(8)-rigid_j*vv(12)
        vv(9)=vv(9)+rigid_j*vv(11)
      endif
      return
      end
C
C      SUBROUTINE /Deal_Rigid_element_f
C
C      剛域のための応力座標変換(剛域内部点から部材端)
C
C      Px = px
C      Py = py
C      My = my - Lg Pz
C      Mz = mz + Lg Py
C
      subroutine Deal_Rigid_element_f(ff,rigid_i,rigid_j)
      implicit real*8(A-H,O-Z)
      dimension ff(12)
      if(rigid_i.eq.0..and.rigid_j.eq.0.) return
      if(rigid_i.ne.0.) then
        ff(5)=ff(5)-rigid_i*ff(3)
        ff(6)=ff(6)+rigid_i*ff(2)
      endif
      if(rigid_j.ne.0.) then
        ff(11)=ff(11)+rigid_j*ff(9)
        ff(12)=ff(12)-rigid_j*ff(8)
      endif
      return
      end

```

最初のサブルーチン Cal_lin_stiff_M1()は、式(5.5)で表される線形の剛性行列 ak_linear を作成する。まず、剛性行列をゼロクリアする。部材長さ r1 は、部材の実際の長さから両端の剛域長さを引いて設定する。後は、式(5.9)にしたがって剛性行列を計算する。このサブルーチ

ンは、両端ピン接合、どちらか片方ピン接合、部材のせん断変形を考慮することが可能であるが、現在では使用不可となっている。

次からのサブルーチンは、部材に剛域を有する場合に関連するもので、サブルーチン Deal_Rigid_element() は、剛性行列 ak を座標変換する。まず、両端に剛域がない場合は、そのままこのサブルーチンより戻る。次に、 i 端と j 端の剛域長さをを用いて座標変換行列を作成する。この座標変換行列を用いて、サブルーチン Rotate_K6() より、変換操作を行う。

サブルーチン Deal_Rigid_element_v() と Deal_Rigid_element_f() は、変位や応力ベクトルを、各々、部材端部から剛域内部点へ、逆に、剛域内部点から部材端部へと変換するプログラムである。変換式は理論マニュアルあるいはプログラム内のコメントを参照されたい。

本節では、部材モデルの幾何学的非線形剛性について解説する。幾何学的非線形性を考慮した接線剛性 $[K_T]$ は、次式で評価される。

$$[K_T] = [K_L] + [K_G] + [K_N] \quad \dots\dots(5.10)$$

ここで、線形剛性行列 $[K_L]$ は前節で示した。次に幾何剛性行列 $[K_G]$ は、次式で示される。ここで、 \bar{N} は部材の軸力を示す。

$$[K_G] = \bar{N} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & \frac{6}{5\ell} & 0 & 0 & 0 & \frac{1}{10} & 0 & -\frac{6}{5\ell} & 0 & 0 & 0 & \frac{1}{10} \\ & & \frac{6}{5\ell} & 0 & -\frac{1}{10} & 0 & 0 & 0 & -\frac{6}{5\ell} & 0 & -\frac{1}{10} & 0 \\ & & & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & \frac{2\ell}{15} & 0 & 0 & 0 & \frac{1}{10} & 0 & -\frac{\ell}{30} & 0 \\ & & & & & \frac{2\ell}{15} & 0 & -\frac{1}{10} & 0 & 0 & 0 & -\frac{\ell}{30} \\ & & & & & & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & & & & \frac{6}{5\ell} & 0 & 0 & 0 & -\frac{1}{10} \\ & & & & & & & & \frac{6}{5\ell} & 0 & \frac{1}{10} & 0 \\ & & & & & & & & & 0 & 0 & 0 \\ & & & & & & & & & & \frac{2\ell}{15} & 0 \\ & & & & & & & & & & & \frac{2\ell}{15} \end{bmatrix} \quad \dots\dots(5.11)$$

sym

5.7 幾何学的非線形モデル

5.7.1 幾何学的非線形剛性行列

また、大変位剛性行列 $[K_N]$ は、

$$[K_N] = \frac{EA}{\ell} \begin{bmatrix} 0 & -\frac{\bar{v}_1 - \bar{v}_2}{\ell} & -\frac{\bar{w}_1 - \bar{w}_2}{\ell} & 0 & 0 & 0 & 0 & \frac{\bar{v}_1 - \bar{v}_2}{\ell} & \frac{\bar{w}_1 - \bar{w}_2}{\ell} & 0 & 0 & 0 \\ \left(\frac{\bar{v}_1 - \bar{v}_2}{\ell}\right)^2 & \frac{\bar{v}_1 - \bar{v}_2}{\ell} \cdot \frac{\bar{w}_1 - \bar{w}_2}{\ell} & 0 & 0 & 0 & \frac{\bar{v}_1 - \bar{v}_2}{\ell} & -\left(\frac{\bar{v}_1 - \bar{v}_2}{\ell}\right)^2 & -\frac{\bar{v}_1 - \bar{v}_2}{\ell} \cdot \frac{\bar{w}_1 - \bar{w}_2}{\ell} & 0 & 0 & 0 \\ & \left(\frac{\bar{w}_1 - \bar{w}_2}{\ell}\right)^2 & 0 & 0 & 0 & \frac{\bar{w}_1 - \bar{w}_2}{\ell} & -\frac{\bar{v}_1 - \bar{v}_2}{\ell} \cdot \frac{\bar{w}_1 - \bar{w}_2}{\ell} & -\left(\frac{\bar{w}_1 - \bar{w}_2}{\ell}\right)^2 & 0 & 0 & 0 \\ & & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & & 0 & -\frac{\bar{v}_1 - \bar{v}_2}{\ell} & -\frac{\bar{w}_1 - \bar{w}_2}{\ell} & 0 & 0 & 0 \\ & & & & & & \left(\frac{\bar{v}_1 - \bar{v}_2}{\ell}\right)^2 & \frac{\bar{v}_1 - \bar{v}_2}{\ell} \cdot \frac{\bar{w}_1 - \bar{w}_2}{\ell} & 0 & 0 & 0 \\ & & & & & & & \left(\frac{\bar{w}_1 - \bar{w}_2}{\ell}\right)^2 & 0 & 0 & 0 \\ & & & & & & & & 0 & 0 & 0 \\ & & & & & & & & & 0 & 0 \\ & & & & & & & & & & 0 \end{bmatrix}$$

sym

.....(5.12)

で表される。

幾何剛性行列 $[K_G]$ と大変位剛性行列 $[K_N]$ を作成するサブルーチンをコールするコードは、例えば、Cal_nonlin_stiff_M1()で以下のように行われる。

```
C
C      SUBROUTINE /Cal_nonlin_stiff_M1
C
C      弾性立体フレーム部材の接線剛性行列の計算(ok)
C
      subroutine Cal_nonlin_stiff_M1(Member ,Element ,
*      an,ak )
      implicit real*8(A-H,O-Z)
      include "submain.h"
      record / member_s    / Member
      record / element_s    / Element
      dimension ak(12,12)
      .
      .
C
      call Cal_lin_stiff_M1(Member,Element,ak) ! 線形剛性を計算する
      call Cal_geomet_stiff(an,Member,ak)    ! 幾何学的非線形剛性を作成
      EA=Element.A*Element.E
```

```

call Create_Kn(ak,Member.an_vv(1),Member.an_ww(1),
*           EA,Member.alength )      ! 大変位剛性剛列の作成
.
.
return
end

```

ここでは、幾何剛性行列を作成するサブルーチン Cal_geomet_stiff() について解説する。このサブルーチンは、軸力 an と部材長さ al より、幾何剛性行列 $[K_G]$ を作成する。このサブルーチンの内容を以下の示す。

```

C
C      SUBROUTINE /Cal_geomet_stiff
C
C      幾何剛性の計算(ok)
C
      subroutine Cal_geomet_stiff(an,Member,ak)
      implicit real*8(A-H,O-Z)
      include "submain.h"
      record / member_s      / Member
      dimension ak(12,12)

C
C      an          real*8   部材軸力
C      Element     structure
C      Member       structure
C      ak          real*8   線形剛性行列
C
      al=Member.alength -
* Member.i_rigid_length - Member.j_rigid_length
      s1=an*6./(5.*al)
      s2=an/10.
      s3=an*al*2./15.
      s4=an*al/30.
      ak(2,2)=ak(2,2) +s1
      ak(2,6)=ak(2,6) +s2
      ak(2,8)=ak(2,8) -s1
      ak(2,12)=ak(2,12)+s2
      ak(6,2)=ak(2,6)
      ak(8,2)=ak(2,8)
      ak(12,2)=ak(2,12)

      ak(3,3)=ak(3,3) +s1
      ak(3,5)=ak(3,5) -s2
      ak(3,9)=ak(3,9) -s1
      ak(3,11)=ak(3,11)-s2
      ak(5,3)=ak(3,5)
      ak(9,3)=ak(3,9)
      ak(11,3)=ak(3,11)

      ak(5,5)=ak(5,5) +s3
      ak(5,9)=ak(5,9) +s2
      ak(5,11)=ak(5,11)-s4

```

```

ak(9,5)=ak(5,9)
ak(11,5)=ak(5,11)

ak(6,6)=ak(6,6) +s3
ak(6,8)=ak(6,8) -s2
ak(6,12)=ak(6,12)-s4
ak(8,6)=ak(6,8)
ak(12,6)=ak(6,12)

ak(8,8)=ak(8,8) +s1
ak(8,12)=ak(8,12) -s2
ak(12,8)=ak(8,12)

ak(9,9)=ak(9,9) +s1
ak(9,11)=ak(9,11) +s2
ak(11,9)=ak(9,11)

ak(11,11)=ak(11,11)+s3
ak(12,12)=ak(12,12)+s3

return
end

```

次に、大変位剛性行列を作成するサブルーチン Create_Kn() について解説する。このサブルーチンは、部材の面外変位 vv、ww と部材長さ rl より、式(5.8)にしたがって大変位剛性行列 $[K_N]$ を作成する。このサブルーチンの内容を以下に示す。

```

C
C      SUBROUTINE /Create_Kn
C
C      非線形剛性の計算(ok)
C
      subroutine Create_Kn(akn,vv,ww,ea,rl)
      implicit real*8(A-H,O-Z)
C
      dimension akn(12,12),ak(3,3)
      do i=1,3
      do j=1,3
      ak(i,j)=0.
      enddo
      enddo
      rl2=rl*rl
      rl3=rl2*rl
      ak(1,2)=ea*vv/rl2
      ak(1,3)=ea*ww/rl2
      ak(2,2)=ea*vv*vv/rl3
      ak(2,3)=ea*vv*ww/rl3
      ak(3,3)=ea*ww*ww/rl3
      ak(2,1)=ak(1,2)

```

5.7.2 大変位剛性 行列

```

ak(3,1)=ak(1,3)
ak(3,2)=ak(2,3)
do i=1,3
do j=1,3
akk=ak(i,j)
akn(i,j) =akn(i,j)+akk
akn(i+6,j) =akn(i+6,j)-akk
akn(i,j+6) =akn(i,j+6)-akk
akn(i+6,j+6)=akn(i+6,j+6)+akk
enddo
enddo
return
end

```

5.8 せん断型モデル

本節では、せん断型弾塑性モデルの剛性行列を作成するサブプログラムについて解説する。ただし、この部材モデルに含まれている各種の履歴モデルについては次章で説明する。ここでは、3つのサブルーチンについて説明する。まず、このサブルーチンをコールするコードを以下に示す。

```

call Cal_lin_stiff_M2(Member(i),Element(ie),ak_linear(1,1,i))
call Cal_check_stiff_M2(Member(i),Element(ie),RO_work,vv,vpp )
call Cal_nonlin_stiff_M2(Member(i),Element(ie),ak )

```

この3つのサブルーチンは、上から線形剛性を求め、次に部材の履歴特性をチェックし、せん断ばねの接線剛性をセットする。最後に、せん断ばねの接線剛性を用いて接線剛性行列を作成する。これらのサブルーチンの内容を具体的に以下に示す。

```

C
C      SUBROUTINE /Cal_lin_stiff_M2
C
C      Model_No.2 3次元せん断弾塑性モデル
C
      subroutine Cal_lin_stiff_M2(Member,Element,ak_linear)
      implicit real*8(A-H,O-Z)
      include "submain.h"
      record / member_s2    / Member
      record / element_s2   / Element
      dimension ak_linear(12,12)
      data BI_MODEL_NUMBER /11/
      data RO_MODEL_NUMBER /12/
C
C      3次元せん断弾塑性モデル（モデルNo.2）
C
c      要素数（モデルNo.2 3次元せん断弾塑性モデル：トリリニア型）
c      element_s 構造体と同一

```

```

c
c      structure / element_s2/
c      integer element_type ! 要素タイプ(6)
c      integer n_element   ! 非線形要素番号
c      real*8  AK_1         ! 第一剛性
c      real*8  AK_2         ! 第二剛性
c      real*8  AK_3         ! 第三剛性
c      real*8  Q_1          ! 第一折れ点のせん断力
c      real*8  Q_2          ! 第二折れ点のせん断力
c      real*8  AKu          ! 軸方向バネ
c      real*8  dm1          ! ダミー
c      real*8  dm2          ! ダミー
c      real*8  dm3          ! ダミー
c      real*8  dm4          ! ダミー
c      integer nm_damp      ! 部材減衰の有無(1)
c      integer nm_type      ! 履歴モデルのタイプ
c      real*8  ANP          ! 軸方向耐力
c      real*8  AQPv         ! y 方向耐力
c      real*8  AQPw         ! z 方向耐力
c      real*8  dmm(3)       ! ダミー
c      end structure
c      record /element_s2/ Element
c      ALLOCATABLE ::Element(:)
c      ALLOCATE (Element(n_element))
c
C
C      3次元せん断弾塑性モデル用(モデルNo.2) member_s 構造体
C
c      ALLOCATABLE :: Member (:)
c      ALLOCATE (Member (n_member))
c      Element          structure
c      Member           structure
c      ak_linear        real*8   線形剛性行列
c
c      do i=1,12
c      do j=1,12
c      ak_linear(j,i) = 0.0
c      end do
c      end do
c      ak=Element.Aku
c      ak_linear(1,1)= ak
c      ak_linear(1,7)=-ak
c      ak_linear(7,7)= ak
c      ak_linear(7,1)=-ak
c      if (Element.nm_type.eq.BI_MODEL_NUMBER) then
c      ak=Element.AK_1*Member.alength
c      else if (Element.nm_type.eq.RO_MODEL_NUMBER) then
c      ak=Element.AK_1/Member.alength
c      else
c      ak=Element.AK_1
c      endif
c      ak_linear(2,2)= ak
c      ak_linear(2,8)=-ak
c      ak_linear(8,8)= ak

```

```

      ak_linear(8,2)=-ak

      ak_linear(3,3)= ak                      ! 5
      ak_linear(3,9)=-ak
      ak_linear(9,9)= ak
      ak_linear(9,3)=-ak

c
      ! Member.AKv_tan=Element.AK_1          ! 6
      Member.AKv_tan=ak                      ! Member.AKv_tan=Element.AK_1
      ! Member.AKw_tan=Element.AK_1
      Member.AKw_tan=ak                      ! Member.AKw_tan=Element.AK_1
      Member.istat_v=0                      ! 7
      Member.istat_w=0
      return
end

c
c      SUBROUTINE /Cal_check_stiff_M2
c
c      Model_No.2 3次元せん断弾塑性モデル
c
      subroutine Cal_check_stiff_M2(Member,Element,RO_work,vv,vpp)
      implicit real*8(A-H,O-Z)
      include "submain.h"
      include "submainx.h"
      record / member_s2 / Member
      record / element_s2 / Element
      record / RO_work_s / RO_work
      dimension vv(12),vpp(12),RO_work(*)

c
c      3次元せん断弾塑性モデル用 (モデル No.2) member_s 構造体
c
c
c      部材
c      structure / member_s2/
c      integer nm_element          ! 要素番号
c      integer element_type        ! 要素タイプ
c      integer n_element_type      ! 要素タイプ別番号
c      integer nm_so               ! 部材の層番号
c      integer nm_dll_element      ! DLL を用いた要素か ( 0 ; システム内要素、 1 : DLL 要素 )
c      integer nm_point(2)        ! 節点番号
c      integer irest(12)          ! 部材両端の自由度番号表
c      integer istat_v             ! y 方向:履歴特性の状態 (*)
c      integer istat_w            ! z 方向:履歴特性の状態 (*)
c      integer nm_analysis        ! 部材解析種別
c      integer nm_group           ! 部材グループ
c      integer nm_local_coord(2) ! 局所座標系の有無とその回転行列の番号
c      integer nm_damp            ! 部材減衰の有無とその減衰行列の番号
c      real*8 alength             ! 長さ
c      real*8 rot_x               ! 部材主軸の回転角度 ( 度 )
c      real*8 force(12)          ! 部材両端の部材端力 ( 釣合座標系 )
c      real*8 stress(6)          ! 部材中央の応力 ( 部材座標系 )
c      real*8 AKv_tan             ! 接線剛性 (*)
c      real*8 AKw_tan            ! 接線剛性 (*)
c      real*8 dm_v(5)             ! y 方向耐力:履歴特性で使用 (*)
c      real*8 dm_w(5)            ! z 方向耐力:履歴特性で使用 (*)
c      end structure

```

```

c
c  ALLOCATABLE :: Member (:)
c  ALLOCATE (Member (n_member))
c  Element      structure
c  Member       structure
c  ak_linear     real*8   線形剛性行列
c
c  No_rireki=Element.nm_type                                ! 8
c  if(No_rireki/10.eq.0) then
c  goto(5,10,20,30,40,50,60),No_rireki+1
5  continue
c
c                                     規定モデル：武田モデル
c  call Takeda_TriLiner(Member,Element,vv,vpp)
c  goto 999
10 continue
c
c                                     トリリニア：Nomal
c  call Mesing_TriLiner(Member,Element,vv,vpp)
c  goto 999
20 continue
c
c                                     トリリニア：最大点指向型
c  call DirecMax_TriLiner(Member,Element,vv,vpp)
c  goto 999
30 continue
c
c                                     トリリニア：武田モデル
c  call Takeda_TriLiner(Member,Element,vv,vpp)
c  goto 999
40 continue
c
c                                     バイリニア：Nomal
c  if(Member.istat_v.eq.0.and.Member.istat_w.eq.0) then
c  Element.AK_2=Element.AK_1
c  Element.Q_2 =Element.Q_1
c  endif
c  call Mesing_TriLiner(Member,Element,vv,vpp)
c  goto 999
50 continue
c
c  goto 999
60 continue
c
c                                     履歴モデル
c  goto 999
c  elseif(No_rireki/10.eq.1) then
c  goto(101,102),No_rireki - 10
101 continue
c
c                                     修正バイリニアモデル（火力センター）
c  mro=Element.n_section(1)
c  write(222,'(A20,I5)') 'Modify Bi-Liner', mro
c  call Modify_Bi_Liner1(Member,Element,RO_work(mro),vv,vpp)
c  goto 999
102 continue
c
c                                     修正 R0 モデル（火力センター）
c  mro=Element.n_section(1)
c  write(222,'(A20,I5)') 'Modify R-0', mro
c  call Modify_R01(Member,Element,RO_work(mro),vv,vpp)

```



```

        goto 999
    endif
999 continue
    return
end

C
C      SUBROUTINE /Cal_nonlin_stiff_M2
C
C      Model_No.2 3次元せん断弾塑性モデル
C
      subroutine Cal_nonlin_stiff_M2(Member,Element,ak)
      implicit real*8(A-H,O-Z)
      include "submain.h"
      record / member_s2      / Member
      record / element_s2     / Element
      dimension ak(12,12)
C
c      ALLOCATABLE :: Member (:)
c      ALLOCATE (Member (n_member))
c      Element      structure
c      Member       structure
c      ak           real*8 非線形剛性行列
c
      do i=1,12
      do j=1,12
      ak(j,i) = 0.0
      end do
      end do
      akk=Element.Aku
      ak(1,1)= akk
      ak(1,7)=-akk
      ak(7,7)= akk
      ak(7,1)=-akk
      akk=Member.AKv_tan
      ak(2,2)= akk
      ak(2,8)=-akk
      ak(8,8)= akk
      ak(8,2)=-akk
      akk=Member.AKw_tan
      ak(3,3)= akk
      ak(3,9)=-akk
      ak(9,9)= akk
      ak(9,3)=-akk
      return
      end

```

プログラムの内容をコードに従って説明する。これらのプログラムは、せん断型弾塑性モデルであり、単純なプログラムである。

1. 線形の剛性行列 `ak_linear` をゼロクリアする。
2. 軸方向剛性を構造体より取得し、剛性行列にセットする。

3. 要素の履歴特性番号にしたがって、横方向せん断ばね定数を取得する。修正バイリニアモデルの剛性は、ばね定数に部材長さを掛けて求め、また、修正 R0 モデルでは、部材長さで割って求める。これについてはリファレンスマニュアルを参照されたい。通常のせん断型ではそのままのばね定数をせん断初期剛性としてセットする。
4. ばね定数を剛性行列の y 方向位置にセットする。
5. ばね定数を剛性行列の z 方向位置にセットする。
6. ばね定数を構造体に初期設定する。
7. ばね定数が弾性であるかどうかチェックするためのパラメータに、初期値として弾性を意味する 0 をセットする。
8. 要素の履歴特性番号（履歴モデルのタイプ）を表す構造体よりパラメータを取得し、その履歴特性番号にしたがって、処理を分割する。このサブルーチンについては前章で説明した。そちらを参照されたい。
9. 非線形の剛性行列 ak をゼロクリアする。
10. 軸方向剛性を構造体より取得し、剛性行列にセットする。
11. せん断ばねの y 方向せん断剛性を取得し、剛性行列にセットする。
12. せん断ばねの z 方向せん断剛性を取得し、剛性行列にセットする。