

Mathcad と Excel VBA を使った

初歩の F E M

上出拓郎 著

まえがき - VBA事始

荷重 $P = 200000$ (N) を面積 $A = 100$ (cm²) で割って応力 または (N/cm²)を求めるもっとも簡単な例から出発しよう。

	A	B
1	$\sigma = P/A$	
2	荷重(N)	200000
3	面積(cm2)	100
4	応力 σ 、 τ (N/cm2)	
5		

Excelを立ち上げてインプットとアウトプットのシートを作る。シートに下のように書きこめば、まず、適当なファイル名「たとえばtest.」などつけて保存する。

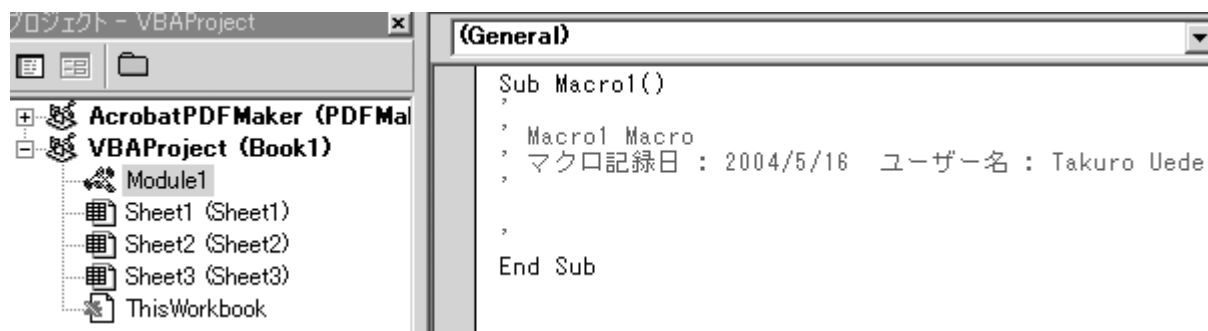
セルB2とセルB3にそれぞれ荷重と面積の値をインプットして、B4に答えをアウトプットしようという訳である。さて、ここでは通常の表計算の方法はとらない。いきなり、マウスで「ツール」をクリックして、「マクロ」、「新しいマクロの記録」とクリックする。



ついでダイアログボックスが出たら「OK」をクリックする。

そして、ふたたび「ツール」「マクロ」「記録終了」とクリックする。

さらに、「マクロ」「マクロ」とクリックして、出てきたダイアログボックスの編集をクリックする。すると次のような、マクロの記録があらわれる。



Sub Macro1() から End Sub までの行にいわゆるプログラムを書き込む。

```
Sub Macro1()
    Macro1 Macro
    マクロ記録日 : 2004/5/16 ユーザー名 : Takuro Uede

    Dim P, A, s
    P = Cells(2, 2)
    A = Cells(3, 2)

    s = P / A

    Cells(4, 2) = s

End Sub
```

これで次のようなプログラムはできあがりである。

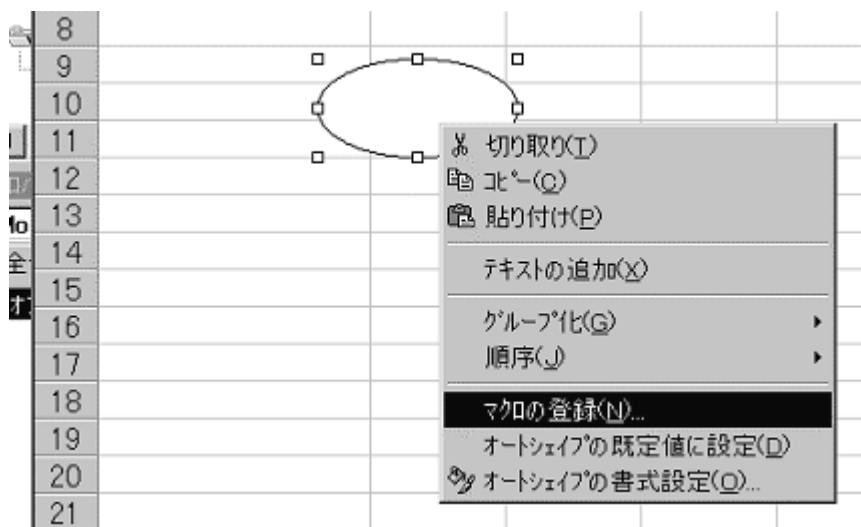
使用する変数 P,A,s を宣言する。

Pに 2 行目 2 列目のセルの中身を入れる。

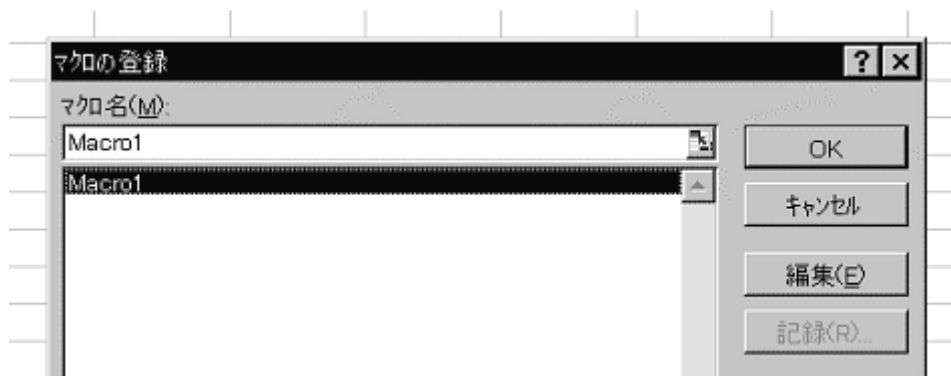
Aに 3 行目 2 列目の中身を入れる。

$s=P/A$ の計算をして 4 行目 2 列目のセルに s の答えを出す。

下のステータスバーのExcelのところをクリックしてExcelのシートにもどり、描画によってたとえば楕円を描き、楕円をさして右クリックすると



「マクロの登録」をクリックして下のダイアログボックスがでてくるので、



図のように下の大きな空間のMacro1をマウスでポイントして青色に変化させ、OKをクリックすると、マクロボタンのできあがりである。

	A	B	C
1	$\sigma = P/A$		
2	荷重(N)	200000	
3	面積(cm ²)	100	
4	応力 σ 、 τ (N/cm ²)	2000	
5			
6			
7		1	
8			
9			
10			
11			
12			

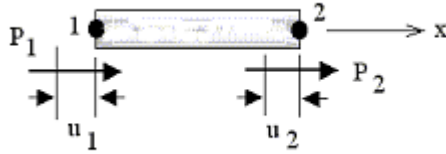
楕円の真中をクリックするとセル B4 に答えがあらわれる。

これで、Excel VBA によって、計算を行う方法がわかった。

第1章 FEM 初歩 トラス

1D の問題

要素の剛性マトリクス Element stiffness matrix



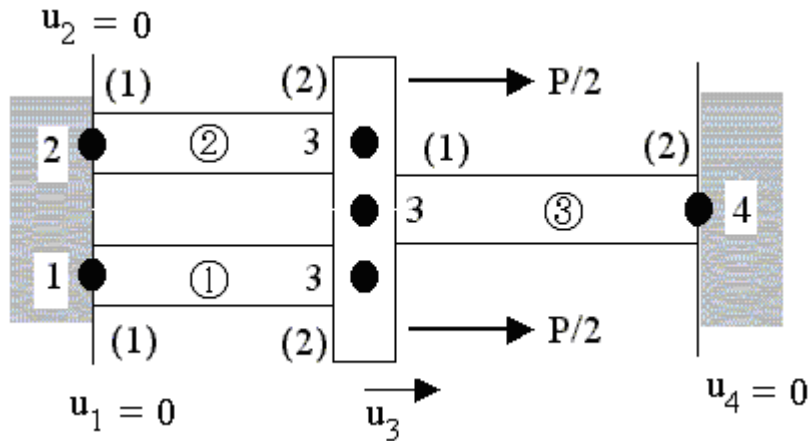
軸方向に荷重をうける部材の端点1と端点2において、図のように P1 の力と P2 の力が働き、端点1は u_1 、端点2は u_2 だけ x 方向に変位するものとする。部材の縦弾性係数を E、断面積を A、長さを L とする。

変位 u_1 、 u_2 と力 P_1 、 P_2 との間にはフックの法則によって次の関係がある。

$$\begin{pmatrix} P_1 \\ P_2 \end{pmatrix} = \frac{EA}{L} \begin{pmatrix} u_1 - u_2 \\ -u_1 + u_2 \end{pmatrix} = \frac{EA}{L} \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$$

上の変位に掛けるマトリクス $K_{el} = \frac{EA}{L} \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$ を要素の剛性マトリクスとよぶ。

全体の剛性マトリクス Global stiffness matrix



最もシンプルな例で説明をすすめよう。各要素の縦弾性係数、断面積、長さを簡単にするため下記のように仮定しよう。

$$\begin{pmatrix} E_{ie} \\ W_{ie} \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad L = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix} \quad .$$

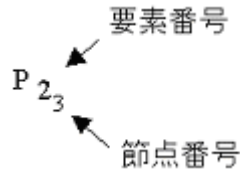
要素 (1)、(2)、(3) にたいして、下の要素マトリクスが作られる。

$$Kel(1) = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \quad Kel(2) = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \quad Kel(3) = \begin{pmatrix} 0.5 & -0.5 \\ -0.5 & 0.5 \end{pmatrix} \quad .$$

また、節点の各変位と各節点に働く力とは下の関係がある。

$$\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_3 \end{pmatrix} = \begin{pmatrix} P1_1 \\ P1_3 \end{pmatrix} \quad \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_3 \end{pmatrix} = \begin{pmatrix} P2_1 \\ P2_3 \end{pmatrix} \quad \begin{pmatrix} 0.5 & -0.5 \\ -0.5 & 0.5 \end{pmatrix} \cdot \begin{pmatrix} u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} P3_3 \\ P3_4 \end{pmatrix}$$

なお、各サブスクリプトは下のとおりである。



各節点に働く力と各節点の変位の関係式を求めるにあたり、各要素の端点 (1) (2) に働く力と変位の全体の剛性方程式に寄与する大きさを求めると下のようになる。

$$\begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} P1_1 \\ 0 \\ P1_3 \\ 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} 0 \\ P2_2 \\ P2_3 \\ 0 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & -0.5 \\ 0 & 0 & -0.5 & 0.5 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ P3_3 \\ P3_4 \end{pmatrix}$$

上の方程式における要素 、 、 の要素剛性マトリクスは以下のようにあらわすことができる。

$$K1 = \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad K2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad K3 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & -0.5 \\ 0 & 0 & -0.5 & 0.5 \end{pmatrix}$$

下のようにこれらを加え合わせたものを全体剛性マトリクスと呼ぶ。

$$KG = K1 + K2 + K3 \quad .$$

剛性マトリクスの現在の例にたいする値は下のようになる。

$$KG = \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & -1 & 0 \\ -1 & -1 & 2.5 & -0.5 \\ 0 & 0 & -0.5 & 0.5 \end{pmatrix} \quad .$$

上の値を使用して下のよう全体剛性方程式を作成する。

$$\begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & -1 & 0 \\ -1 & -1 & 2.5 & -0.5 \\ 0 & 0 & -0.5 & 0.5 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} P1_1 \\ P2_2 \\ P1_3 + P2_3 + P3_3 \\ P3_4 \end{pmatrix} = \begin{pmatrix} (?) \\ ? \\ 1 \\ (?) \end{pmatrix} \quad .$$

全体の剛性方程式において右辺の3行目の $P1_3 + P2_3 + P3_3$ は $P = 1$ という荷重がかかっていることが判明しているので数値が1と決定できるが、1行目、2行目、4行目の数値は不明である。左辺における変位のうち節点1, 2, 4は静止しているため、それぞれの接点に対応する変位 u_1 , u_2 , u_4 は0である。このように、変位が最初から既知数であることが条件づけられていることを拘束条件という。

また、節点3で荷重が既知数であることを荷重条件という。

両者の条件を合わせて境界条件という。

境界条件による全体剛性マトリクスと荷重ベクトルの修正

全体の剛性方程式、すなわち連立方程式において、左辺の u_1 , u_2 , u_3 , u_4 のうち、3つの u_1 , u_2 , u_4 は拘束条件から0という答えがすでに出ているので、

$$\begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & -1 & 0 \\ -1 & -1 & 2.5 & -0.5 \\ 0 & 0 & -0.5 & 0.5 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} P1_1 \\ P2_2 \\ P1_3 + P2_3 + P3_3 \\ P3_4 \end{pmatrix} = \begin{pmatrix} (?) \\ ? \\ 1 \\ (?) \end{pmatrix}$$

から、全体剛性マトリクスと荷重ベクトルを修正し、答のうちの3つの u_1 , u_2 , u_4 が0なるように誘導してやらなければならない。この方法は後でのべるが、それは

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 2.5 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

のような修正された剛性方程式となり、とりあえず変位を求めてしまおう。

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 2.5 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}^{-1} \cdot \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0.4 \\ 0 \end{pmatrix} \quad .$$

求められた変位から力の値が判明する。

$$\begin{pmatrix} P1_1 \\ P1_3 \end{pmatrix} = Kel(1) \cdot \begin{pmatrix} u_1 \\ u_3 \end{pmatrix} = \begin{pmatrix} -0.4 \\ 0.4 \end{pmatrix} \quad \begin{pmatrix} P2_2 \\ P2_3 \end{pmatrix} = Kel(2) \cdot \begin{pmatrix} u_2 \\ u_3 \end{pmatrix} = \begin{pmatrix} -0.4 \\ 0.4 \end{pmatrix}$$

$$\begin{pmatrix} P3_3 \\ P3_4 \end{pmatrix} = Kel(3) \cdot \begin{pmatrix} u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} 0.2 \\ -0.2 \end{pmatrix}$$

連立方程式に既知の根を含める方法

$$\begin{pmatrix} a11 & a12 & a13 \\ a21 & a22 & a23 \\ a31 & a32 & a33 \end{pmatrix} \cdot \begin{pmatrix} u1 \\ u2 \\ u3 \end{pmatrix} = \begin{pmatrix} b1 \\ b2 \\ b3 \end{pmatrix}$$

上の連立方程式の係数マトリクスと未知数ベクトルの積を分解すると下となる。

$$\begin{pmatrix} a11 \cdot u1 + a12 \cdot u2 + a13 \cdot u3 \\ a21 \cdot u1 + a22 \cdot u2 + a23 \cdot u3 \\ a31 \cdot u1 + a32 \cdot u2 + a33 \cdot u3 \end{pmatrix} = \begin{pmatrix} b1 \\ b2 \\ b3 \end{pmatrix}$$

ここで、 $u_1 = B$ のように未知数のひとつ $u1$ がすでにわかっているものとする。 そのときは

$$\begin{pmatrix} u1 \\ a21 \cdot u1 + a22 \cdot u2 + a23 \cdot u3 \\ a31 \cdot u1 + a32 \cdot u2 + a33 \cdot u3 \end{pmatrix} = \begin{pmatrix} B \\ b2 \\ b3 \end{pmatrix}$$

で、かきなおせば

$$\begin{pmatrix} u1 \\ a22 \cdot u2 + a23 \cdot u3 \\ a32 \cdot u2 + a33 \cdot u3 \end{pmatrix} = \begin{pmatrix} B \\ b2 - a21 \cdot B \\ b3 - a31 \cdot B \end{pmatrix} \quad \text{でさらにかきなおすと、}$$

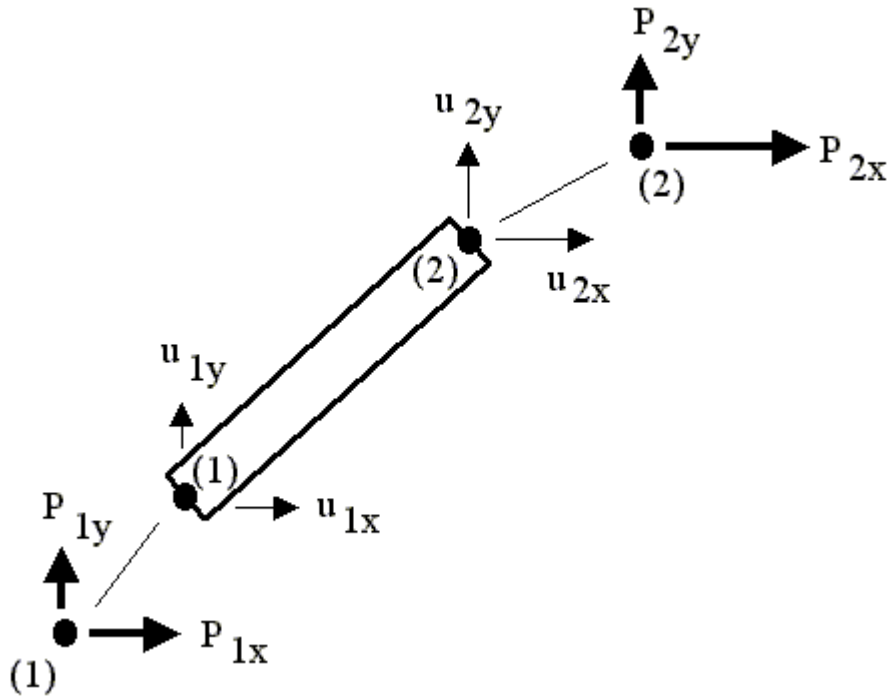
$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & a22 & a23 \\ 0 & a32 & a33 \end{pmatrix} \cdot \begin{pmatrix} u1 \\ u2 \\ u3 \end{pmatrix} = \begin{pmatrix} B \\ b2 - a21 \cdot B \\ b3 - a31 \cdot B \end{pmatrix}$$

すなわち、係数マトリクスの既知数の接点番号の行と列の対角の数値を 1 に変更し、他の数値を 0 にすれば、境界条件 $u1=0$ を入れた連立方程式となる。 これで 2 つの未知数 $u2$ 、 $u3$ を解くことができる。

この方法を応用して、いずれかの答の判明している場合の $u1$ 、 $u2$ 、 $u3$ を求める連立方程式の修正が簡単に行える。

2D の問題

要素の剛性マトリクス Element stiffness matrix



図で軸方向に荷重をうける部材を考える。部材の端点 [1] と端点 [2] において、 P_1 の力 (x 方向成分 P_{1x} 、 y 方向成分 P_{1y}) と P_2 (x 方向成分 P_{2x} 、 y 方向成分 P_{2y}) の力が働くものとする。端点 [1] は u_1 (x 方向成分 u_{1x} 、 y 方向成分 u_{1y}) 変位するものとする。端点 [2] は u_2 (x 方向成分 u_{2x} 、 y 方向成分 u_{2y}) だけ変位するものとする。部材の縦弾性係数を E 、断面積を A 、長さを L とする。

フックの法則を適用しよう。変位 u と力 P_1 、 P_2 の間には次の関係がなりたつ。ただし、 cs 、 sn は部材の傾きの \cos 、 \sin をあらわす。

$$\begin{pmatrix} P_{1x} \\ P_{1y} \\ P_{2x} \\ P_{2y} \end{pmatrix} = \frac{AE}{L} \begin{pmatrix} cs^2 & cs \cdot sn & 0 & 0 \\ cs \cdot sn & sn^2 & 0 & 0 \\ 0 & 0 & cs^2 & cs \cdot sn \\ 0 & 0 & cs \cdot sn & sn^2 \end{pmatrix} \cdot \begin{pmatrix} u_{1x} - u_{2x} \\ u_{1y} - u_{2y} \\ -u_{1x} + u_{2x} \\ -u_{1y} + u_{2y} \end{pmatrix} \quad .$$

上の式を変形すると、

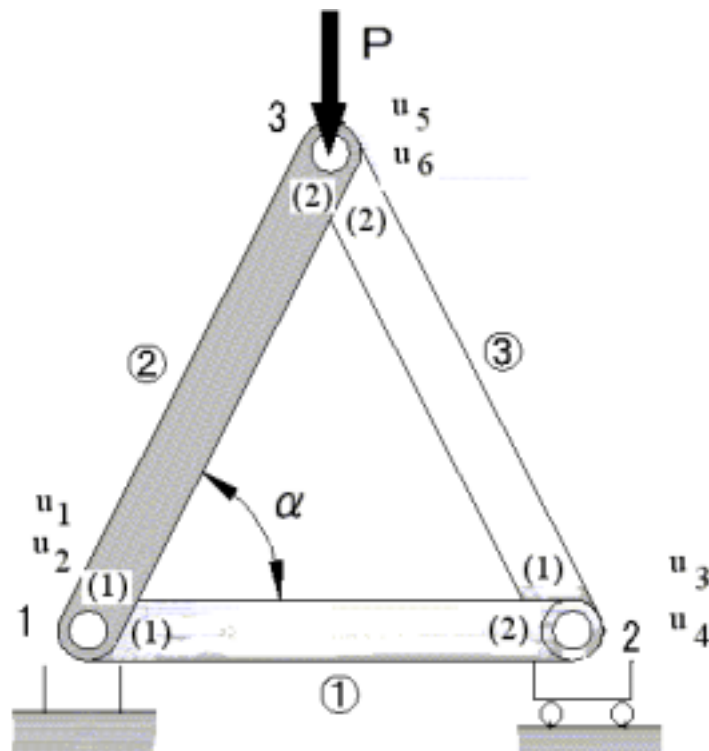
$$\begin{pmatrix} P_{1x} \\ P_{1y} \\ P_{2x} \\ P_{2y} \end{pmatrix} = \frac{AE}{L} \begin{pmatrix} cs^2 & cs \cdot sn & -cs^2 & -cs \cdot sn \\ cs \cdot sn & sn^2 & -cs \cdot sn & -sn^2 \\ -cs^2 & -cs \cdot sn & cs^2 & cs \cdot sn \\ -cs \cdot sn & -sn^2 & cs \cdot sn & sn^2 \end{pmatrix} \cdot \begin{pmatrix} u_{1x} \\ u_{1y} \\ u_{2x} \\ u_{2y} \end{pmatrix} \quad \text{となる。}$$

$$Kel(ie) = \frac{A_{ie} E_{ie}}{L_{ie}} \begin{pmatrix} cs_{ie}^2 & cs_{ie} \cdot sn_{ie} & -cs_{ie}^2 & -cs_{ie} \cdot sn_{ie} \\ cs_{ie} \cdot sn_{ie} & sn_{ie}^2 & -cs_{ie} \cdot sn_{ie} & -sn_{ie}^2 \\ -cs_{ie}^2 & -cs_{ie} \cdot sn_{ie} & cs_{ie}^2 & cs_{ie} \cdot sn_{ie} \\ -cs_{ie} \cdot sn_{ie} & -sn_{ie}^2 & cs_{ie} \cdot sn_{ie} & sn_{ie}^2 \end{pmatrix}$$

を要素の剛性マトリクスとよぶ。

全体の剛性マトリクス

次に示す最もシンプルなトラスを例にとって説明をすすめよう。



図において、 $\alpha = 60^\circ$ として、 $P = 100000$ の荷重がかかるものとしよう。ここでは、計算の経緯がわかりやすいように、すべての部材の断面積、縦弾性係数、長さは簡単化して1としよう。

$$\begin{bmatrix} A \\ L \\ E \end{bmatrix} := \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \quad \begin{bmatrix} cs_1 \\ cs_2 \\ cs_3 \end{bmatrix} := \begin{bmatrix} 1 \\ \frac{1}{2} \\ -\frac{1}{2} \end{bmatrix} \quad \begin{bmatrix} sn_1 \\ sn_2 \\ sn_3 \end{bmatrix} := \begin{bmatrix} 0 \\ \frac{\sqrt{3}}{2} \\ \frac{\sqrt{3}}{2} \end{bmatrix}$$

要素 (1) (2) (3) にたいして、3つの要素マトリクスができる。

$$\text{Kel}(1) = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{Kel}(2) = \begin{bmatrix} 0.25 & 0.433 & -0.25 & -0.433 \\ 0.433 & 0.75 & -0.433 & -0.75 \\ -0.25 & -0.433 & 0.25 & 0.433 \\ -0.433 & -0.75 & 0.433 & 0.75 \end{bmatrix} \quad \text{Kel}(3) = \begin{bmatrix} 0.25 & -0.433 & -0.25 & 0.433 \\ -0.433 & 0.75 & 0.433 & -0.75 \\ -0.25 & 0.433 & 0.25 & -0.433 \\ 0.433 & -0.75 & -0.433 & 0.75 \end{bmatrix}$$

要素 (1) (2) (3) にたいして、節点の各変位と各節点に働く力との間には次のように 3 つの関係式がなりたつ。

$$\begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} P_{1_1} \\ P_{1_2} \\ P_{1_3} \\ P_{1_4} \end{bmatrix} \quad \begin{bmatrix} 0.25 & 0.433 & -0.25 & -0.433 \\ 0.433 & 0.75 & -0.433 & -0.75 \\ -0.25 & -0.433 & 0.25 & 0.433 \\ -0.433 & -0.75 & 0.433 & 0.75 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_5 \\ u_6 \end{bmatrix} = \begin{bmatrix} P_{2_1} \\ P_{2_2} \\ P_{2_5} \\ P_{2_6} \end{bmatrix}$$

$$\begin{bmatrix} 0.25 & -0.433 & -0.25 & 0.433 \\ -0.433 & 0.75 & 0.433 & -0.75 \\ -0.25 & 0.433 & 0.25 & -0.433 \\ 0.433 & -0.75 & -0.433 & 0.75 \end{bmatrix} \begin{bmatrix} u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix} = \begin{bmatrix} P_{3_3} \\ P_{3_4} \\ P_{3_5} \\ P_{3_6} \end{bmatrix}$$

要素番号
P_{3₅}
荷重番号

各節点には、荷重、変位とも x、y 方向の 2 個の値があるから、それぞれ x 方向 y 方向の番号を振り当てる。 節点 (1) には 1 , 2、節点 (2) には 3 , 4、節点 (3) には 5 , 6 というふうに。 節点 (1) から (3) の変位 u₁、u₂、u₃、--- u₆ をすべて縦にならべて、要素 (1)、(2) (3) の端点 [1] [2] に働く力との関係をまとめて見よう。

次に示す 3 個の関係は、要素 (1)、(2)、(3) の端点 [1]、[2] に働く力が、いかに全体の剛性方程式に寄与するかをあらわしている。

$$\text{K1} = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix} = \begin{bmatrix} P_{1_1} \\ P_{1_2} \\ P_{1_3} \\ P_{1_4} \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{aligned}
 K2 &= \begin{bmatrix} 0.25 & 0.433 & 0 & 0 & -0.25 & -0.433 \\ 0.433 & 0.75 & 0 & 0 & -0.433 & -0.75 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -0.25 & -0.433 & 0 & 0 & 0.25 & 0.433 \\ -0.433 & -0.75 & 0 & 0 & 0.433 & 0.75 \end{bmatrix} & \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix} & \begin{bmatrix} P_{2_1} \\ P_{2_2} \\ 0 \\ 0 \\ P_{2_5} \\ P_{2_6} \end{bmatrix} \\
 K3 &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.25 & -0.433 & -0.25 & 0.433 \\ 0 & 0 & -0.433 & 0.75 & 0.433 & -0.75 \\ 0 & 0 & -0.25 & 0.433 & 0.25 & -0.433 \\ 0 & 0 & 0.433 & -0.75 & -0.433 & 0.75 \end{bmatrix} & \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix} & \begin{bmatrix} 0 \\ 0 \\ P_{3_3} \\ P_{3_4} \\ P_{3_5} \\ P_{3_6} \end{bmatrix}
 \end{aligned}$$

要素 (1) (2) (3) の要素剛性マトリクスは上の関係を使って、全体の剛性マトリクスの中に埋め込まれる。 次の式に示すように、これらを加え合わせたものを全体剛性マトリクスと呼ぶ。

$$KG := K1 + K2 + K3$$

実際の数値をいれたものは次のようになる。

$$KG = \begin{bmatrix} 1.25 & 0.433 & -1 & 0 & -0.25 & -0.433 \\ 0.433 & 0.75 & 0 & 0 & -0.433 & -0.75 \\ -1 & 0 & 1.25 & -0.433 & -0.25 & 0.433 \\ 0 & 0 & -0.433 & 0.75 & 0.433 & -0.75 \\ -0.25 & -0.433 & -0.25 & 0.433 & 0.5 & 0 \\ -0.433 & -0.75 & 0.433 & -0.75 & 0 & 1.5 \end{bmatrix} \cdot$$

これを使って剛性方程式を作成する

$$\begin{bmatrix} 1.25 & 0.433 & -1 & 0 & -0.25 & -0.433 \\ 0.433 & 0.75 & 0 & 0 & -0.433 & -0.75 \\ -1 & 0 & 1.25 & -0.433 & -0.25 & 0.433 \\ 0 & 0 & -0.433 & 0.75 & 0.433 & -0.75 \\ -0.25 & -0.433 & -0.25 & 0.433 & 0.5 & 0 \\ -0.433 & -0.75 & 0.433 & -0.75 & 0 & 1.5 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix} = \begin{bmatrix} P_{1_1} + P_{2_1} \\ P_{1_2} + P_{2_2} \\ P_{1_3} + P_{3_3} \\ P_{1_4} + P_{3_4} \\ P_{2_5} + P_{3_5} \\ P_{2_6} + P_{3_6} \end{bmatrix}$$

境界条件による全体剛性マトリクスと荷重ベクトルの修正

荷重番号 1 , 2 では、支持点からどのような反力を受けるかわからないので荷重は未知、荷重番号 3 では、支持点が x 方向にスライドするのでは反力 0、すなわち既知で数値は 0、荷重番号 4

では反力が不明なので未知、荷重番号5では x 方向荷重 0 のため既知で 0、荷重番号 6 では下向きに 100000 となる。 したがって下のごとくなる。

$$\begin{bmatrix} 1.25 & 0.433 & -1 & 0 & -0.25 & -0.433 \\ 0.433 & 0.75 & 0 & 0 & -0.433 & -0.75 \\ -1 & 0 & 1.25 & -0.433 & -0.25 & 0.433 \\ 0 & 0 & -0.433 & 0.75 & 0.433 & -0.75 \\ -0.25 & -0.433 & -0.25 & 0.433 & 0.5 & 0 \\ -0.433 & -0.75 & 0.433 & -0.75 & 0 & 1.5 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix} = \begin{bmatrix} ? \\ ? \\ 0 \\ ? \\ 0 \\ -1 \cdot 10^5 \end{bmatrix}$$

境界条件 $\begin{pmatrix} u_1 \\ u_2 \\ u_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$ によって、剛性マトリクスの修正がなされ、右辺の?のところが埋めら

れることになり、

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1.25 & 0 & -0.25 & 0.433 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -0.25 & 0 & 0.5 & 0 \\ 0 & 0 & 0.433 & 0 & 0 & 1.5 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \cdot 10^5 \end{bmatrix}$$

これを解くことによって、変位はすべて求められ

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1.25 & 0 & -0.25 & 0.433 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -0.25 & 0 & 0.5 & 0 \\ 0 & 0 & 0.433 & 0 & 0 & 1.5 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \cdot 10^5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0.028 \\ 0 \\ 0.014 \\ -0.073 \end{bmatrix}$$

ポストプロセッシング

上に求められた変位から、各要素（部材）の軸方向の伸びをもとめ、

$$\begin{bmatrix} \text{elong}_1 \\ \text{elong}_2 \\ \text{elong}_3 \end{bmatrix} := \begin{bmatrix} \text{cs}_1 \cdot (u_3 - u_1) + \text{sn}_1 \cdot (u_4 - u_2) \\ \text{cs}_2 \cdot (u_5 - u_1) + \text{sn}_2 \cdot (u_6 - u_2) \\ \text{cs}_3 \cdot (u_5 - u_3) + \text{sn}_3 \cdot (u_6 - u_4) \end{bmatrix}$$

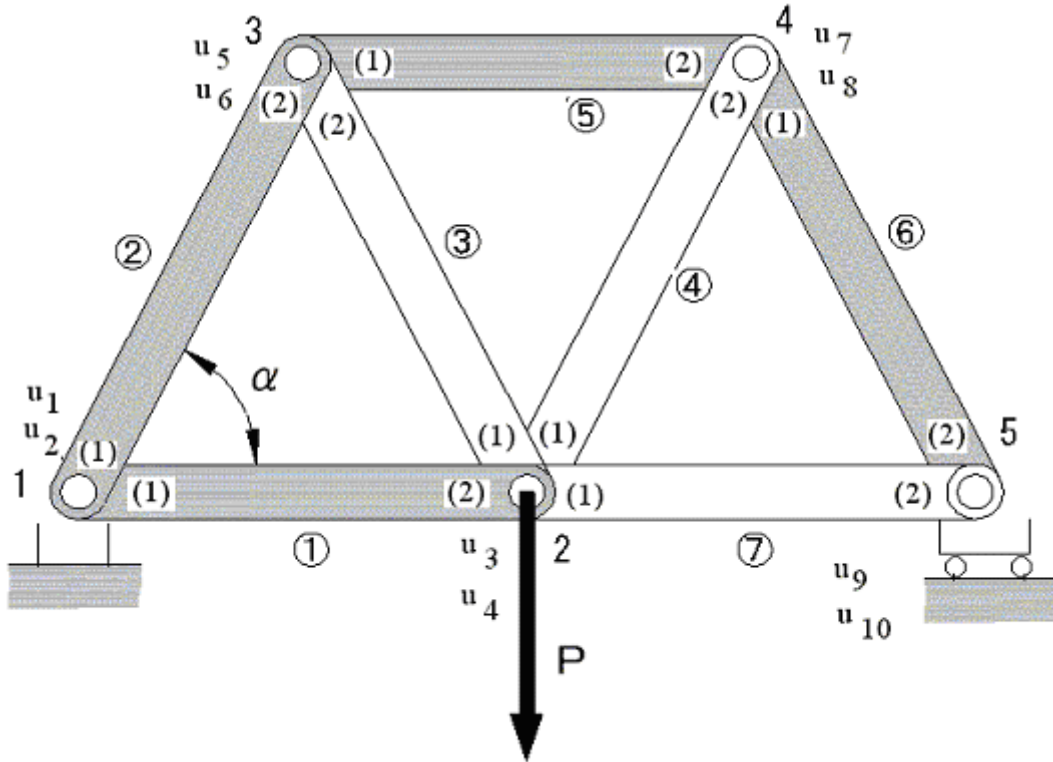
断面積、縦弾性係数、長さの補正を行うと

$$P_{ie} := \frac{A_{ie} \cdot E}{L_{ie}} \cdot \text{elong}_{ie}$$

次のように各要素の軸方向の力の値が判明する。

$$\begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix} = \begin{bmatrix} 2.887 \cdot 10^4 \\ -5.774 \cdot 10^4 \\ -5.774 \cdot 10^4 \end{bmatrix} \quad \blacksquare$$

2DトラスのFEM定式化



今までにおこなってきた行列の操作を自動的に行わせることを考えよう。 図において、 $\alpha = 60^\circ$ として、部材(1) (2) (3) (4) (5) (6)とも $L=1$, $A=1$, $E=1$ として、 $P=1$ の力がかかるとき、接点2の変位を求めるための計算の一般化を試みよう。まず、要素の断面積 A 、長さ L 、縦弾性係数 E を与えよう。

$$A_{ie} := 1 \quad L_{ie} := 1 \quad E := 1$$

要素 (element) の数、1 から要素の数までの範囲を示す数値 ie 、が与えられる。

$$n_{el} := 7 \quad ie := 1..n_{el}$$

次に、節点 (node) の数

$$n_{nodes} := 5$$

各要素の傾きをあらわす、 \cos 、 \sin は端点 [1] から端点 [2] に向かう傾きをとる。

$$\begin{bmatrix} cs_1 \\ cs_2 \\ cs_3 \\ cs_4 \end{bmatrix} := \begin{bmatrix} 1 \\ \frac{1}{2} \\ -\frac{1}{2} \\ \frac{1}{2} \end{bmatrix} \quad \begin{bmatrix} cs_5 \\ cs_6 \\ cs_7 \end{bmatrix} := \begin{bmatrix} 1 \\ \frac{1}{2} \\ 1 \end{bmatrix} \quad \begin{bmatrix} sn_1 \\ sn_2 \\ sn_3 \\ sn_4 \end{bmatrix} := \begin{bmatrix} 0 \\ \frac{\sqrt{3}}{2} \\ \frac{\sqrt{3}}{2} \\ \frac{\sqrt{3}}{2} \end{bmatrix} \quad \begin{bmatrix} sn_5 \\ sn_6 \\ sn_7 \end{bmatrix} := \begin{bmatrix} 0 \\ -\frac{\sqrt{3}}{2} \\ 0 \end{bmatrix}$$

各要素の端点 [1] と端点 [2] がどの節点番号に一致するか、要素番号順にあらわしたトポロジーテーブル (topology table) を作成する。

$$\text{Top} := \begin{bmatrix} 1 & 2 \\ 1 & 3 \\ 2 & 3 \\ 2 & 4 \\ 3 & 4 \\ 4 & 5 \\ 2 & 5 \end{bmatrix}$$

topology table の 1 列目は端点 [1] にたいする節点の番号、 2 列目は端点 [2] にたいする接点の番号に対応する。

$$\begin{bmatrix} \text{nn_pe} \\ \text{ndof_pn} \end{bmatrix} := \begin{bmatrix} 2 \\ 2 \end{bmatrix}$$

で各要素における端点の数、端点ごとの自由度 (この場合は 2 D なので x、 y 方向の 2) があたえられる。

$$\text{n_nodes} := 5 \quad \text{ndof_pe} := \text{nn_pe} \cdot \text{ndof_pn}$$

で接点の数、要素ごとの自由度 (要素 1 個に接点 2 個、接点 1 個に自由度 2 なのでこの場合は 4) を与える。

要素の剛性マトリクスの作成

次の式で要素の剛性マトリクスが計算される。

$$\text{ie} := 1..n_el$$

$$\text{Kel}_{\text{ie}} := \frac{\text{A}_{\text{ie}} \cdot \text{E}}{\text{L}_{\text{ie}}} \cdot \begin{bmatrix} (\text{cs}_{\text{ie}})^2 & \text{cs}_{\text{ie}} \cdot \text{sn}_{\text{ie}} & -(\text{cs}_{\text{ie}})^2 & -\text{cs}_{\text{ie}} \cdot \text{sn}_{\text{ie}} \\ \text{cs}_{\text{ie}} \cdot \text{sn}_{\text{ie}} & (\text{sn}_{\text{ie}})^2 & -\text{cs}_{\text{ie}} \cdot \text{sn}_{\text{ie}} & -(\text{sn}_{\text{ie}})^2 \\ -(\text{cs}_{\text{ie}})^2 & -\text{cs}_{\text{ie}} \cdot \text{sn}_{\text{ie}} & (\text{cs}_{\text{ie}})^2 & \text{cs}_{\text{ie}} \cdot \text{sn}_{\text{ie}} \\ -\text{cs}_{\text{ie}} \cdot \text{sn}_{\text{ie}} & -(\text{sn}_{\text{ie}})^2 & \text{cs}_{\text{ie}} \cdot \text{sn}_{\text{ie}} & (\text{sn}_{\text{ie}})^2 \end{bmatrix}$$

全体の剛性マトリクスの作成

演算のレンジを指定し、要素の剛性マトリクスの内容を節点ごとにまとめ、全体の剛性マトリクス K を求める。

$$\text{n} := \text{n_nodes} \cdot \text{ndof_pn}$$

$$\text{K}_{\text{n},\text{n}} := 0 \quad \text{ie} := 1..n_el$$

$$\text{p} := 1..\text{ndof_pn} \quad \text{q} := 1..\text{ndof_pn}$$

$$i := 1..n_{pe} \quad j := 1..n_{pe}$$

$$K_{\left[\left(\text{Top}_{ie,i-1} \right)^{2+p}, \left(\text{Top}_{ie,j-1} \right)^{2+q} \right]} := K_{\left[\left(\text{Top}_{ie,i-1} \right)^{2+p}, \left(\text{Top}_{ie,j-1} \right)^{2+q} \right]} \dots + \left(\text{Kel}_{ie} \right)_{(i-1)^{2+p},(j-1)^{2+q}}$$

境界条件のインプット

$$\text{Restrains} := \begin{bmatrix} 1 & 0 \\ 2 & 0 \\ 10 & 0 \end{bmatrix} \quad \text{Loads} := (4 \ -1)$$

拘束条件すなわち、変位 u_1 、 u_2 、 u_{10} が既知数 0 で、荷重は荷重番号 4 に（接点 2 の y 方向）- 1 であると指定する。

$$n_{res} := \text{rows}(\text{Restrains})$$

によって、拘束件数を求める。

境界条件による全体剛性マトリクスの修正

$$\begin{aligned} id &:= 1..n_{res} \\ indp &:= 1..2 \cdot n_{nodes} \quad indj := 1..2 \cdot n_{nodes} \\ K_{mod} &:= K \\ K_{mod}_{\text{Restrains}_{id,1}, indj} &:= 0 \\ K_{mod}_{indj, \text{Restrains}_{id,1}} &:= 0 \\ K_{mod}_{\left(\text{Restrains}_{id,1} \right), \text{Restrains}_{id,1}} &:= 1 \end{aligned}$$

荷重ベクトルの修正

$$\begin{aligned} nl &:= \text{rows}(\text{Loads}) \\ il &:= 1..nl \\ f_{mod}_{indp} &:= 0 \\ f_{mod}_{\left(\text{Loads}_{il,1} \right)} &:= f_{mod}_{\left(\text{Loads}_{il,1} \right)} + \text{Loads}_{il,2} \\ f_{mod}_{indp} &:= f_{mod}_{indp} - K_{indp, \text{Restrains}_{id,1}} \cdot \text{Restrains}_{id,2} \\ f_{mod}_{\left(\text{Restrains}_{id,1} \right)} &:= \text{Restrains}_{id,2} \quad | \quad K \end{aligned}$$

修正された全体の剛性マトリクスと荷重ベクトルはこの例では次のようになる。

$$K_{\text{mod}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2.5 & 0 & -0.25 & 0.433 & -0.25 & -0.433 & -1 & 0 \\ 0 & 0 & 0 & 1.5 & 0.433 & -0.75 & -0.433 & -0.75 & 0 & 0 \\ 0 & 0 & -0.25 & 0.433 & 1.5 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0.433 & -0.75 & 0 & 1.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & -0.25 & -0.433 & -1 & 0 & 1.5 & 0 & -0.25 & 0 \\ 0 & 0 & -0.433 & -0.75 & 0 & 0 & 0 & 1.5 & 0.433 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & -0.25 & 0.433 & 1.25 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{f}_{\text{mod}} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

変位の計算

$$\mathbf{u} := K_{\text{mod}}^{-1} \cdot \mathbf{f}_{\text{mod}}$$

によって変位が次のように求められる。

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0.289 \\ -1.833 \\ 0.577 \end{bmatrix} \quad \begin{bmatrix} u_6 \\ u_7 \\ u_8 \\ u_9 \\ u_{10} \end{bmatrix} = \begin{bmatrix} -1 \\ 0 \\ -1 \\ 0.577 \\ 0 \end{bmatrix} \quad \mathbf{fsol} := K \cdot \mathbf{u} \quad \begin{bmatrix} fsol_1 \\ fsol_2 \\ fsol_3 \\ fsol_4 \\ fsol_5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0.5 \\ 0 \\ -1 \\ 0 \end{bmatrix} \quad \begin{bmatrix} fsol_6 \\ fsol_7 \\ fsol_8 \\ fsol_9 \\ fsol_{10} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0.5 \end{bmatrix}$$

左側のベクトルが変位で、右側は検算によって荷重を求めてみたところである。

ポストプロセッシング (Post processing)

変位ベクトルを元の要素の端点 [1] [2] の変位と要素に発生する軸方向の力の関係によって、もどしてやると各部材の軸力が求まる。

$$p_{ie} := \text{Top}_{ie,1} \quad r_{ie} := \text{Top}_{ie,2}$$

$$i := 1..n \text{ nodes} \quad uu_i := u_{2i-1} \quad vv_i := u_{2i}$$

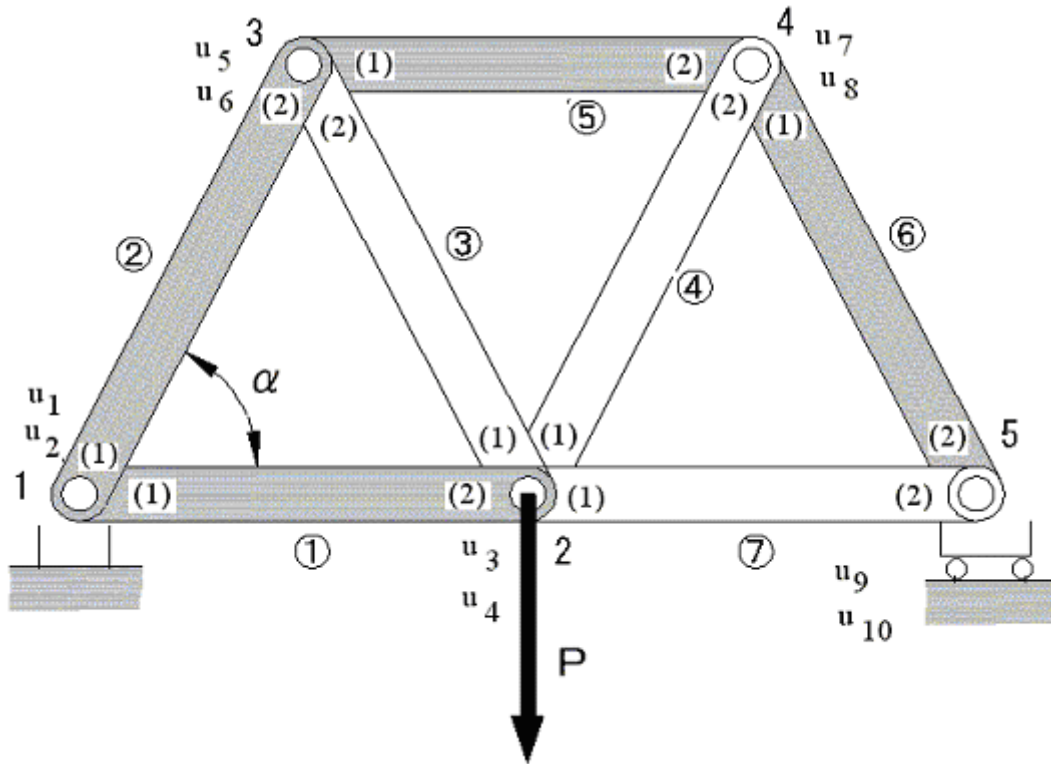
$$\text{def}_{ie} := \cos_{ie} \cdot [uu(r_{ie}) - uu(p_{ie})] + \sin_{ie} \cdot [vv(r_{ie}) - vv(p_{ie})]$$

$$P_{ie} := \frac{A_{ie} \cdot E}{L_{ie}} \cdot \text{def}_{ie} \quad \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \end{bmatrix} = \begin{bmatrix} 0.289 \\ -0.577 \\ 0.577 \\ 0.577 \end{bmatrix} \quad \begin{bmatrix} P_5 \\ P_6 \\ P_7 \end{bmatrix} = \begin{bmatrix} -0.577 \\ -0.577 \\ 0.289 \end{bmatrix}$$

では次に、これを表計算ソフト Excel VBA を使ってプログラムを作ろう。

次のようなトラスのたわみと各部材にかかる荷重を FEM でもとめてみよう。 荷重番号を節点 1 には (1 , 2) 節点 2 に (3 , 4) 節点 3 に (5 , 6) 節点 4 に (7 , 8)

節点5に(9, 10)とわりあてる。 各部材の長さを 200 cm、断面積を 10 cm² とし、荷重が荷重番号4(節点2のy方向)に -10000 N かかるものとして、各節点の変位、各部材にかかる力をもとめてみよう。



	A	B	C	D	E	F	G	H	I	J	K	L	
1	TRUSS FEM												
2	上出拓郎 1989/12/31 作成												
3	接点数 = 5	要素数 = 7										E = 20600000	縦弾性係数
4	拘束点数 3	荷重点数 1										計算ボタン	
6	要素番号	断面積	長さ	cos(a)	sin(a)	Topology		拘束番号	拘束値	荷重番号	荷重		
7	1	10	200	1.000	0.000	1	2	1	0	4	-100000		
8	2	10	200	0.500	0.866	1	3	2	0				
9	3	10	200	-0.500	0.866	2	3	10	0				
10	4	10	200	0.500	0.866	2	4						
11	5	10	200	1.000	0.000	3	4						
12	6	10	200	0.500	-0.866	4	5						
13	7	10	200	1.000	0.000	2	5						
14	8												
15	入力												
16	要素番号	力	接点番号	変位		検算							
17	1	28867.51	1	u1	0								
18	2	-57735	2	u2	0								
19	3	57735.03	3	u3	0.02803								
20	4	57735.03	4	u4	-0.178								
21	5	-57735	5	u5	0.05605								
22	6	-57735	6	u6	-0.0971								
23	7	28867.51	7	u7	-2E-17								
24	8		9	u8	-0.0971								
25				u9	0.05605								
26				u10	0								

図に示すように、入力の部分と出力の部分割り振る。そして マクロ VBA をつかってプログラムを作り、表の上中央の楕円形のボタンを押せば出力として、各部材の受ける力と変位番号にたいする変位が求められるものとする。

```
Sub Macro1()  
' Macro1 Macro TRUSS FEM トラスの F E M  
Dim nnode, ne, nload, nrest, ie, i, j, ip, ij, p, r  
Dim top(10, 2), restraints(10, 2), loads(10, 2)  
Dim E, cs(10), sn(10), area(10), L(10)  
Dim Kel(10, 4, 4), KG(20, 20)  
Dim Kmod(20, 20), f(10), fmod(10)  
Dim pp, a(20, 20), cc(20), u(20), s, Force(10), elong(10), uu(10, 2)  
  
'-インプットを表から読み込む-----  
E = Cells(3, 9)  
nnode = Cells(3, 2): ne = Cells(3, 4): nrest = Cells(4, 2): nload = Cells(4, 4)  
  
'-データを表から読み込む-----  
For i = 1 To ne  
cs(i) = Cells(i + 6, 4): sn(i) = Cells(i + 6, 5)  
area(i) = Cells(i + 6, 2): L(i) = Cells(i + 6, 3)  
  
For j = 1 To 2  
top(i, j) = Cells(i + 6, j + 5)  
Next  
Next  
  
For i = 1 To nrest  
For j = 1 To 2  
restraints(i, j) = Cells(i + 6, j + 8)  
Next  
Next  
  
For i = 1 To nload  
For j = 1 To 2  
loads(i, j) = Cells(i + 6, j + 10)  
Next  
Next  
  
'-要素の剛性マトリクス Kel を作るサブルーチン呼び出す----  
For ie = 1 To ne  
elementstifmat Kel(), ie, cs(), sn(), area(), E, L()  
Next  
  
'-全体の剛性マトリクス KG を作るサブルーチン呼び出す----  
globstifmat nnode, ne, top(), Kel(), KG(), ie  
  
'-境界条件により KG を修正するサブルーチン呼び出す----  
modifyK KG(), restraints(), Kmod(), nnode, nrest  
  
'-境界条件により荷重ベクトルを修正するサブルーチン呼び出す----  
forcevecmod nnode, nrest, nload, loads(), f(), fmod(), restraints(), KG()  
  
'-連立方程式を解くサブルーチン呼び出す----  
For i = 1 To nnode * 2  
cc(i) = fmod(i)  
  
For j = 1 To nnode * 2  
a(i, j) = Kmod(i, j)
```

Next
Next

matsol a(), cc(), u(), nnode

'-ポストプロセスのサブルーチンを呼び出す----
postprocess cs(), sn(), top(), u(), uu(), elong(), ne, nnode

For ie = 1 To ne
Force(ie) = area(ie) * E / L(ie) * elong(ie)
Cells(ie + 16, 2) = Force(ie)
Next

For i = 1 To 2 * nnode
Cells(i + 16, 5) = u(i)
Next

End Sub

Sub elementstifmat(Kel(), ie, cs(), sn(), area(), E, L())
Kel(ie, 1, 1) = cs(ie) ^ 2: Kel(ie, 1, 2) = cs(ie) * sn(ie)
Kel(ie, 2, 1) = Kel(ie, 1, 2): Kel(ie, 2, 2) = sn(ie) ^ 2
For i = 1 To 2
For j = 1 To 2
Kel(ie, i, j + 2) = -Kel(ie, i, j)
Kel(ie, i + 2, j) = -Kel(ie, i, j)
Kel(ie, i + 2, j + 2) = Kel(ie, i, j)
Next
Next
For i = 1 To 4
For j = 1 To 4
Kel(ie, i, j) = area(ie) * E / L(ie) * Kel(ie, i, j)
Next
Next
End Sub

Sub globstifmat(nnode, ne, top(), Kel(), KG(), ie)
For i = 1 To 2 * nnode
For j = 1 To 2 * nnode
KG(i, j) = 0
Next
Next
For ie = 1 To ne
For p = 1 To 2
For j = 1 To 2
ip = top(ie, p)
ij = top(ie, j)
KG(2 * ip - 1, 2 * ij - 1) = KG(2 * ip - 1, 2 * ij - 1) + Kel(ie, 2 * p - 1, 2 * j - 1)
KG(2 * ip, 2 * ij - 1) = KG(2 * ip, 2 * ij - 1) + Kel(ie, 2 * p, 2 * j - 1)
KG(2 * ip - 1, 2 * ij) = KG(2 * ip - 1, 2 * ij) + Kel(ie, 2 * p - 1, 2 * j)
KG(2 * ip, 2 * ij) = KG(2 * ip, 2 * ij) + Kel(ie, 2 * p, 2 * j)
Next
Next
Next
End Sub

Sub modifyK(KG(), restraints(), Kmod(), nnode, nrest)
For ip = 1 To 2 * nnode
For ij = 1 To 2 * nnode
Kmod(ip, ij) = KG(ip, ij)
Next
Next
For ij = 1 To 2 * nnode
For i = 1 To nrest

```

Kmod(restraints(i, 1), ij) = 0
Kmod(ij, restraints(i, 1)) = 0
Kmod(restraints(i, 1), restraints(i, 1)) = 1
Next
Next
End Sub

```

```

Sub forcevecmom(nnode, nrest, nload, loads(), f(), fmod(), restraints(), KG())
For i = 1 To nnode * 2
f(i) = 0
fmod(i) = 0
Next
For i = 1 To nload
fmod(loads(i, 1)) = loads(i, 2)
Next
For i = 1 To nnode * 2
For p = 1 To nres
fmod(i) = fmod(i) - KG(i, restraints(p, 1)) * restraints(p, 2)
Next
Next
For i = 1 To nrest
fmod(i) = restraints(i, 2)
Next
End Sub

```

```

Sub matsol(a(), cc(), u(), nnode)
For j = 1 To nnode * 2 - 1
For i = j + 1 To nnode * 2
pp = a(i, j) / a(j, j)
cc(i) = cc(i) - cc(j) * pp
For ind = j To nnode * 2
a(i, ind) = a(i, ind) - a(j, ind) * pp
Next
Next
Next
u(nnode * 2) = cc(nnode * 2) / a(nnode * 2, nnode * 2)
For j = nnode * 2 - 1 To 1 Step -1
s = 0
For i = j + 1 To nnode * 2
s = s + a(j, i) * u(i)
Next
u(j) = (cc(j) - s) / a(j, j)
Next
End Sub

```

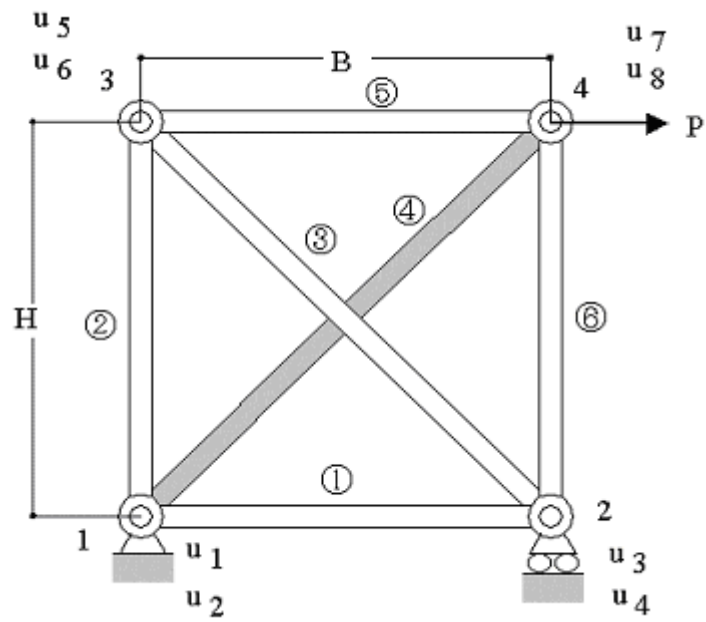
```

Sub postprocess(cs(), sn(), top(), u(), uu(), elong(), ne, nnode)
For ie = 1 To ne
elong(ie) = 0
Next
For i = 1 To nnode
uu(i, 1) = u(2 * i - 1): uu(i, 2) = u(2 * i): ' x-displacement,y-displacement
Next
For ie = 1 To ne
p = top(ie, 1): r = top(ie, 2)
elong(ie) = cs(ie) * (uu(r, 1) - uu(p, 1)) + sn(ie) * (uu(r, 2) - uu(p, 2))
Next
End Sub

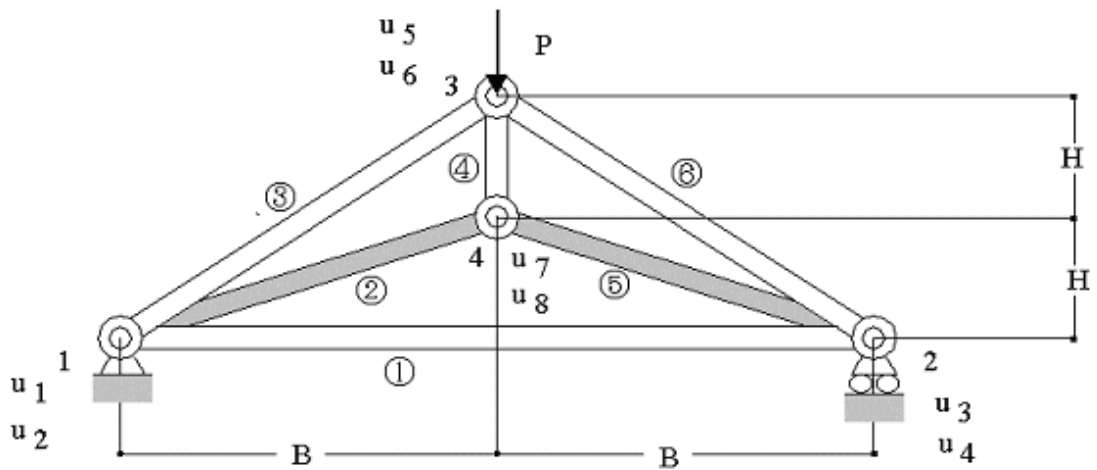
```

練習問題

1 . 図において、 $H = 2000 \text{ mm}$ 、 $B = 2000 \text{ mm}$ 、 $P = 100000 \text{ N}$ とする。また、 $E = 206000 \text{ N/mm}^2$ 、 $A = 1000 \text{ mm}^2$ とする。各節点の変位、各部材に発生する荷重を求めよ。



2. 図において、 $H = 400 \text{ mm}$ 、 $B = 1200 \text{ mm}$ 、 $P = 100000 \text{ N}$ とする。また、 $E = 206000 \text{ N/mm}^2$ 、 $A = 1000 \text{ mm}^2$ とする。各節点の変位、各部材に発生する荷重を求めよ。



(ヒント) 1.

第 2 章 曲げモーメントを受ける部材

2.1. 連続ばりの問題

(1) 要素の剛性方程式

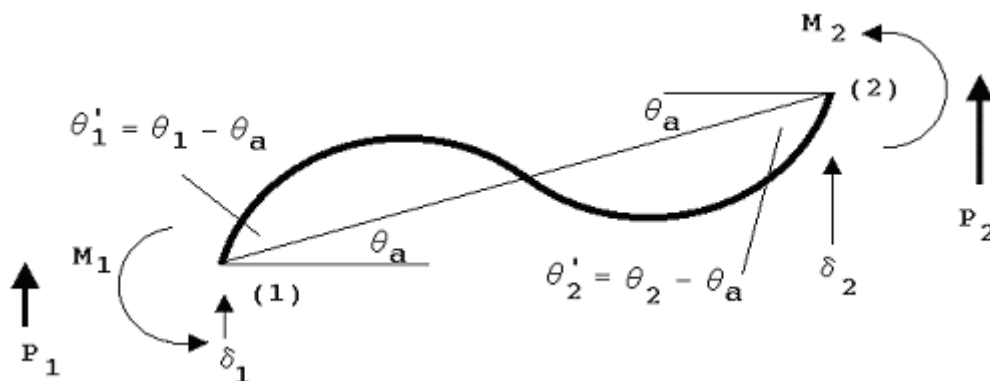


図において長さLのはりが左端点で M_1 の曲げモーメントを受け、右端点で M_2 の曲げモーメントを受けているものとする。そして、左端点、右端点での傾き角度をそれぞれ θ_1 、 θ_2 とする。このとき、曲げモーメントと傾き角度とはつぎのような関係式が成り立つ。

$$\theta_1 = \frac{M_1}{3 \cdot EI} \cdot L + \frac{M_2}{6 \cdot EI} \cdot L \quad \theta_2 = \frac{M_1}{6 \cdot EI} \cdot L + \frac{M_2}{3 \cdot EI} \cdot L$$

$$M_1 = \frac{4 \cdot EI}{L} \cdot \theta_1 + \frac{2 \cdot EI}{L} \cdot \theta_2 \quad M_2 = \frac{2 \cdot EI}{L} \cdot \theta_1 + \frac{4 \cdot EI}{L} \cdot \theta_2$$

次に、下の図のように、はりが端点に垂直方向の荷重 P_1 、 P_2 を受けて、端点(1)と(2)がそれぞれ、垂直方向に δ_1 、 δ_2 変位したものとする。



すると、次の関係式がなりたつ。

$$\theta_a = \frac{\delta_2 - \delta_1}{L} \quad \theta'_1 = \theta_1 - \theta_a \quad \theta'_2 = \theta_2 - \theta_a$$

$$M_1 = \begin{bmatrix} \frac{6 \cdot EI}{L^2} & \frac{4 \cdot EI}{L} & -\frac{6 \cdot EI}{L^2} & \frac{2 \cdot EI}{L} \end{bmatrix} \cdot \begin{bmatrix} \delta_1 \\ \theta_1 \\ \delta_2 \\ \theta_2 \end{bmatrix}$$

$$M_2 = \begin{bmatrix} \frac{6 \cdot E \cdot I}{L^2} & \frac{2 \cdot E \cdot I}{L} & -\frac{6 \cdot E \cdot I}{L^2} & \frac{4 \cdot E \cdot I}{L} \end{bmatrix} \cdot \begin{bmatrix} \delta_1 \\ \theta_1 \\ \delta_2 \\ \theta_2 \end{bmatrix}$$

荷重 P_1 、 P_2 と M_1 、 M_2 には次の関係式がなりたつ。

$$P_1 = \frac{M_1 + M_2}{L} \quad P_2 = \frac{-(M_1 + M_2)}{L}$$

これから次のような要素の剛性方程式ができる。

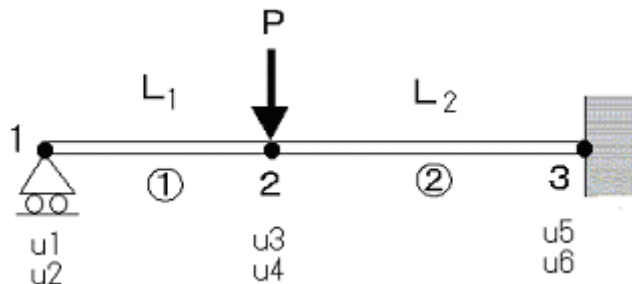
$$\begin{bmatrix} P_1 \\ M_1 \\ P_2 \\ M_2 \end{bmatrix} = \begin{bmatrix} \frac{12 \cdot E \cdot I}{L^3} & \frac{6 \cdot E \cdot I}{L^2} & -\frac{12 \cdot E \cdot I}{L^3} & \frac{6 \cdot E \cdot I}{L^2} \\ \frac{6 \cdot E \cdot I}{L^2} & \frac{4 \cdot E \cdot I}{L} & -\frac{6 \cdot E \cdot I}{L^2} & \frac{2 \cdot E \cdot I}{L} \\ -\frac{12 \cdot E \cdot I}{L^3} & -\frac{6 \cdot E \cdot I}{L^2} & \frac{12 \cdot E \cdot I}{L^3} & -\frac{6 \cdot E \cdot I}{L^2} \\ \frac{6 \cdot E \cdot I}{L^2} & \frac{2 \cdot E \cdot I}{L} & -\frac{6 \cdot E \cdot I}{L^2} & \frac{4 \cdot E \cdot I}{L} \end{bmatrix} \cdot \begin{bmatrix} \delta_1 \\ \theta_1 \\ \delta_2 \\ \theta_2 \end{bmatrix}$$

実際の計算で使い易いよう、接点（要素の端点）の変位と回転は u であらわし、サブスクリプトが奇数のものは変位、偶数のものは回転とする。接点（要素の端点）の力とモーメントは P で表し、サブスクリプトが奇数のものは力、偶数のものはモーメントとする。

$$\begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \end{bmatrix} = \frac{2 \cdot E \cdot I_{ie}}{(L_{ie})^3} \cdot \begin{bmatrix} 6 & 3 \cdot L_{ie} & -6 & 3 \cdot L_{ie} \\ 3 \cdot L_{ie} & 2 \cdot (L_{ie})^2 & -3 \cdot L_{ie} & (L_{ie})^2 \\ -6 & -3 \cdot L_{ie} & 6 & -3 \cdot L_{ie} \\ 3 \cdot L_{ie} & (L_{ie})^2 & -3 \cdot L_{ie} & 2 \cdot (L_{ie})^2 \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}$$

(2) 連続ばりの例

(1) で要素の剛性方程式を求めることができたので、トラスの場合で学んだ方法とおなじ要領で定式化する。次の例をとって進める。



図においてはりが左端点で単純支持、右端点で固定されているものとする。接点 2 に P の荷重を受ける場合の接点 2 の変位と回転、接点 1 の反力と接点 3 にかかる反力およびモーメントを求

める。 なお、数式のチェックが楽なように、L、I、E は1とする。

データのインプット

$$\begin{bmatrix} E \\ A \\ I \\ L \\ n_{el} \end{bmatrix} := \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 2 \end{bmatrix} \quad \begin{matrix} ie := 1..n_{el} & A_{ie} := A & I_{ie} := I & E_{ie} := E & L_{ie} := L \\ n_{node} = 3 \end{matrix}$$

$$Top := \begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix}$$

要素の数、1から要素の数までの範囲を示す数値 ie、要素の断面積 A、長さ L、縦弾性係数 E を与る。さらに、Top で各要素の端点(1)、(2) に対する全体の接点の対応を右辺のマトリクス の行の順番に与る。

要素の剛性マトリクスの作成

$$Kel_{ie} := \frac{2 \cdot E_{ie} \cdot I_{ie}}{(L_{ie})^3} \begin{bmatrix} 6 & 3 \cdot L_{ie} & -6 & 3 \cdot L_{ie} \\ 3 \cdot L_{ie} & 2 \cdot (L_{ie})^2 & -3 \cdot L_{ie} & (L_{ie})^2 \\ -6 & -3 \cdot L_{ie} & 6 & -3 \cdot L_{ie} \\ 3 \cdot L_{ie} & (L_{ie})^2 & -3 \cdot L_{ie} & 2 \cdot (L_{ie})^2 \end{bmatrix}$$

$$Kel_1 = \begin{bmatrix} 12 & 6 & -12 & 6 \\ 6 & 4 & -6 & 2 \\ -12 & -6 & 12 & -6 \\ 6 & 2 & -6 & 4 \end{bmatrix} \quad Kel_2 = \begin{bmatrix} 12 & 6 & -12 & 6 \\ 6 & 4 & -6 & 2 \\ -12 & -6 & 12 & -6 \\ 6 & 2 & -6 & 4 \end{bmatrix}$$

全体の剛性マトリクスの作成

$$n := 2 \cdot n_{nodes} \quad K_{n,n} := 0 \quad ie := 1..n_{el}$$

$$p := 1..2 \quad q := 1..2 \quad i := 1..2 \quad j := 1..2$$

$$K_{[(Top_{ie,i}-1) \cdot 2+p], (Top_{ie,j}-1) \cdot 2+q]} := K_{[(Top_{ie,i}-1) \cdot 2+p], (Top_{ie,j}-1) \cdot 2+q]} + (Kel_{ie})_{(i-1) \cdot 2+p, (j-1) \cdot 2+q}$$

境界条件のインプット

$$Restrains := \begin{bmatrix} 1 & 0 \\ 5 & 0 \\ 6 & 0 \end{bmatrix} \quad Loads := (3 \ -1)$$

$$n_{res} := rows(Restrains)$$

変位 u_1 、 u_5 、 u_6 が既知数 0 で、荷重は荷重番号 3（接点 2 の y 方向）-1 と指定する。また、拘束件数を求める。

境界条件による全体剛性マトリクスと荷重ベクトルの修正

$$\text{indp} := 1..2 \cdot n_{\text{nodes}} \quad \text{indj} := 1..2 \cdot n_{\text{nodes}} \quad \text{id} := 1..n_{\text{res}} \quad \text{nl} := \text{rows}(\text{Loads}) \quad \text{il} := 1..nl$$

$$\begin{aligned} \text{Kmod} &:= \text{K} \\ \text{Kmod}_{\text{Restrains}_{\text{id},1}, \text{indj}} &:= 0 \\ \text{Kmod}_{\text{indj}, \text{Restrains}_{\text{id},1}} &:= 0 \\ \text{Kmod}_{(\text{Restrains}_{\text{id},1}), \text{Restrains}_{\text{id},1}} &:= 1 \\ \text{fmod}_{\text{indp}} &:= 0 \\ \text{fmod}_{(\text{Loads}_{\text{il},1})} &:= \text{fmod}_{(\text{Loads}_{\text{il},1})} + \text{Loads}_{\text{il},2} \\ \text{fmod}_{\text{indp}} &:= \text{fmod}_{\text{indp}} - \text{K}_{\text{indp}, \text{Restrains}_{\text{id},1}} \cdot \text{Restrains}_{\text{id},2} \\ \text{fmod}_{(\text{Restrains}_{\text{id},1})} &:= \text{Restrains}_{\text{id},2} \end{aligned}$$

全体剛性マトリクスが修正されて Kmod 、荷重ベクトルが fmod となる。変位ベクトル u は次のように求める。

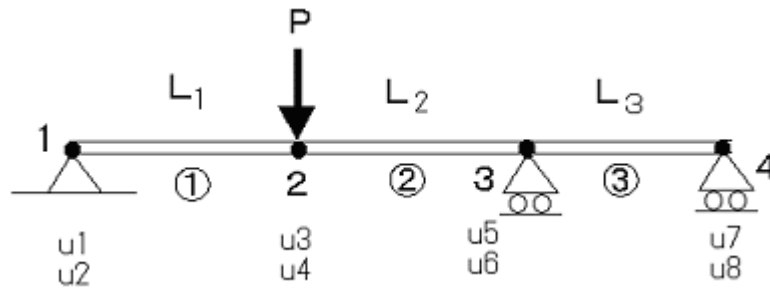
$$u := \text{Kmod}^{-1} \cdot \text{fmod} \quad u = \begin{bmatrix} 0 \\ -0.125 \\ -0.073 \\ 0.031 \\ 0 \\ 0 \end{bmatrix}$$

また、得られた変位ベクトル u を、各要素の端点 (1)、(2) に割り当てて、それぞれの要素の剛性マトリクスをかけると、その要素の端点にたいしてその要素が受け持つ力とモーメントが求められる。

$$p_{\text{ie}} := \text{Top}_{\text{ie},1} \quad r_{\text{ie}} := \text{Top}_{\text{ie},2} \quad i := 1..n_{\text{nodes}} \quad uu_i := u_{2i-1} \quad vv_i := u_{2i}$$

$$\text{PP}_{\text{ie}} := \text{Kel}_{\text{ie}} \cdot \begin{bmatrix} uu(p_{\text{ie}}) \\ vv(p_{\text{ie}}) \\ uu(r_{\text{ie}}) \\ vv(r_{\text{ie}}) \end{bmatrix} \quad \text{PP}_1 = \begin{bmatrix} 0.313 \\ 0 \\ -0.313 \\ 0.313 \end{bmatrix} \quad \text{PP}_2 = \begin{bmatrix} -0.688 \\ -0.313 \\ 0.688 \\ -0.375 \end{bmatrix}$$

(3) 連続ばりの例題 2



図においてはりが左端点、中端点、右端点で単純支持でx方向には拘束を受けいないものとする。
 $L_1 = 0.5$ 、 $L_2 = 0.5$ 、 $L_3 = 1$ とし、節点2に $P = 1$ の荷重を受ける場合の各節点回転、反力、モーメントを求める。なお、数式のチェックが楽なように、 I 、 E は 1 とする。

$$\begin{bmatrix} E \\ A \\ I \\ n_{el} \end{bmatrix} := \begin{bmatrix} 1 \\ 1 \\ 1 \\ 3 \end{bmatrix} \quad \text{Top} := \begin{bmatrix} 1 & 2 \\ 2 & 3 \\ 3 & 4 \end{bmatrix} \quad \begin{bmatrix} L_1 \\ L_2 \\ L_3 \end{bmatrix} := \begin{bmatrix} 0.5 \\ 0.5 \\ 1 \end{bmatrix}$$

$$ie := 1..n_{el} \quad A_{ie} := A \quad I_{ie} := I \quad E_{ie} := E$$

$$n_{nodes} := 4$$

要素の剛性マトリクスの作成は前例と同じ。

全体の剛性マトリクスの作成も定式化されているので前例と同じ。自動的に計算が進む。要素の剛性マトリクスの内容が接点ごとにまとめられて、全体の剛性マトリクス K も同じように求める。境界条件のインプットは次のように行う。

$$\text{Restrains} := \begin{bmatrix} 1 & 0 \\ 5 & 0 \\ 7 & 0 \end{bmatrix} \quad \text{Loads} := (3 - 1)$$

境界条件による全体剛性マトリクスと荷重ベクトルの修正がなされ、全体剛性マトリクスが修正されて $K_{I\text{mod}}$ 、荷重マトリクスが f_{mod} となる。変位ベクトル u が求められる。

$$u = \begin{bmatrix} 0 \\ -0.047 \\ -0.015 \\ 3.906 \cdot 10^{-3} \\ 0 \\ 0.031 \\ 0 \\ -0.016 \end{bmatrix}$$

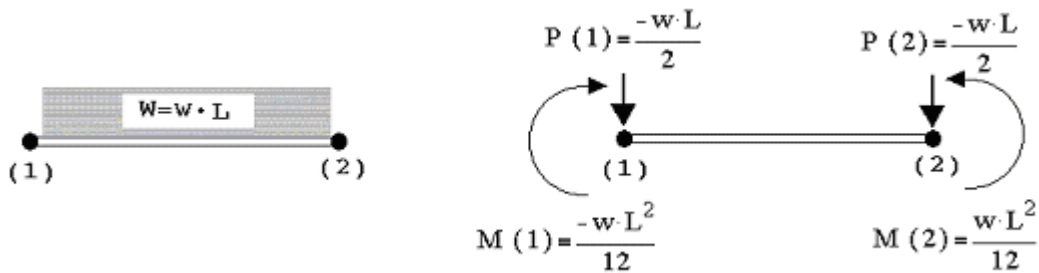
また、得られた変位 u のベクトルを、各要素の端点 (1)、(2) に割り当てて、それぞれの要素の剛性マトリクスをかけると、要素のそれぞれの端点にたいしてその要素が受け持つ力とモーメントが次のように求められる。

$$p_{ie} := \text{Top}_{ie,1} \quad r_{ie} := \text{Top}_{ie,2} \quad i := 1..n \text{ nodes}$$

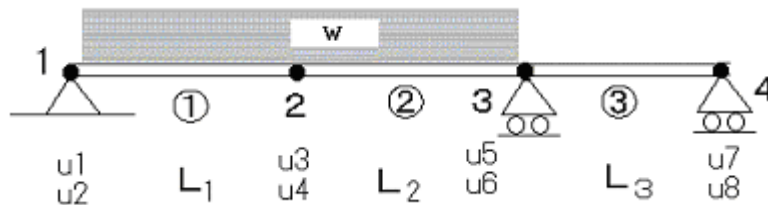
$$uu_i := u_{2i-1} \quad vv_i := u_{2i}$$

$$PP_{ie} := K_{el_{ie}} \cdot \begin{bmatrix} uu(p_{ie}) \\ vv(p_{ie}) \\ uu(r_{ie}) \\ vv(r_{ie}) \end{bmatrix} \quad PP_1 = \begin{bmatrix} 0.406 \\ 0 \\ -0.406 \\ 0.203 \end{bmatrix} \quad PP_2 = \begin{bmatrix} -0.594 \\ -0.203 \\ 0.594 \\ -0.094 \end{bmatrix} \quad PP_3 = \begin{bmatrix} 0.094 \\ 0.094 \\ -0.094 \\ 0 \end{bmatrix}$$

(4) 等分布荷重の場合



図右のように部材が左端点、右端点の間に下向きに等分布荷重を受ける場合には、その部材内部両端点に図左のような相当荷重と相当モーメントが加わっているとして境界条件を作る。それでは例題にはいる。



図における連続ばりの節点1と3の間に等分布荷重がかかっている。節点1と3の間の中点における変位を知りたい場合には、節点2をもうけ、定石にしたがって、まずデータのインプットを行う。

$$\begin{bmatrix} E \\ A \\ I \\ n_{el} \end{bmatrix} := \begin{bmatrix} 1 \\ 1 \\ 1 \\ 3 \end{bmatrix} \quad \text{Top} := \begin{bmatrix} 1 & 2 \\ 2 & 3 \\ 3 & 4 \end{bmatrix} \quad \begin{bmatrix} L_1 \\ L_2 \\ L_3 \end{bmatrix} := \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} \quad n_{nodes} := 4$$

要素の剛性マトリクスと全体の剛性マトリクスの作成も定式化されているので自動的に計算が進む。要素の剛性マトリクスの内容が接点ごとにまとめられて、全体の剛性マトリクス \$K\$ が求められる。境界条件のインプットで、等分布荷重は節点の集中荷重とモーメントとしてインプットされる。

$$\text{Restrains} := \begin{bmatrix} 1 & 0 \\ 5 & 0 \\ 7 & 0 \end{bmatrix} \quad w := 1 \quad \text{Loads} := \begin{bmatrix} 1 & \frac{-w \cdot L_1}{2} \\ 2 & \frac{-w \cdot (L_1)^2}{12} \\ 3 & \frac{-w \cdot L_1}{2} + \frac{-w \cdot L_2}{2} \\ 4 & \frac{w \cdot (L_1)^2}{12} + \frac{-w \cdot (L_2)^2}{12} \\ 5 & \frac{-w \cdot L_2}{2} \\ 6 & \frac{w \cdot (L_2)^2}{12} \end{bmatrix}$$

境界条件による全体剛性マトリクスと荷重ベクトルの修正が自動的になされ、全体剛性マトリクスが修正される。

変位ベクトル u が求められる。

$$u := K_{\text{mod}}^{-1} \cdot f_{\text{mod}} \quad u = \begin{bmatrix} 0 \\ -0.25 \\ -0.146 \\ 0.021 \\ 0 \\ 0.167 \\ 0 \\ -0.083 \end{bmatrix}$$

また、得られた変位 u のベクトルを、各要素の端点 (1)、(2) に割り当てて、それぞれの要素の剛性マトリクスをかけると、要素のそれぞれの端点にたいしてその要素が等分布荷重に相当する内部的な相当荷重ベクトルと、本来受け持つべき荷重ベクトルの和としてのベクトルが求められる。

$$p_{ie} := \text{Top}_{ie,1} \quad r_{ie} := \text{Top}_{ie,2} \quad i := 1..n \text{ nodes}$$

$$uu_i := u_{2i-1} \quad vv_i := u_{2i}$$

$$PP_{ie} := K_{el_{ie}} \cdot \begin{bmatrix} uu(p_{ie}) \\ vv(p_{ie}) \\ uu(r_{ie}) \\ vv(r_{ie}) \end{bmatrix} \quad PP_1 = \begin{bmatrix} 0.375 \\ -0.083 \\ -0.375 \\ 0.458 \end{bmatrix} \quad PP_2 = \begin{bmatrix} -0.625 \\ -0.458 \\ 0.625 \\ -0.167 \end{bmatrix} \quad PP_3 = \begin{bmatrix} 0.125 \\ 0.25 \\ -0.125 \\ 0 \end{bmatrix}$$

それぞれの要素の受け持つべき荷重ベクトルは等分布荷重による内部の相当荷重ベクトルを次の

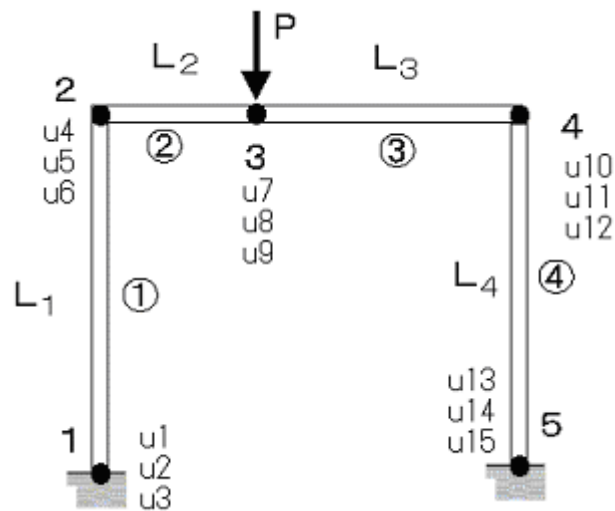
ように差し引いてもとめることができる。

$$\text{intforce}_1 := \begin{bmatrix} \frac{-w \cdot L_1}{2} \\ \frac{-w \cdot (L_1)^2}{12} \\ \frac{-w \cdot L_1}{2} \\ \frac{w \cdot (L_1)^2}{12} \end{bmatrix} \quad \text{intforce}_2 := \begin{bmatrix} \frac{-w \cdot L_2}{2} \\ \frac{-w \cdot (L_2)^2}{12} \\ \frac{-w \cdot L_2}{2} \\ \frac{w \cdot (L_2)^2}{12} \end{bmatrix} \quad i := 1..2$$

$$\text{PPP}_i := \text{PP}_i - \text{intforce}_i \quad \text{PPP}_1 = \begin{bmatrix} 0.875 \\ 0 \\ 0.125 \\ 0.375 \end{bmatrix} \quad \text{PPP}_2 = \begin{bmatrix} -0.125 \\ -0.375 \\ 1.125 \\ -0.25 \end{bmatrix}$$

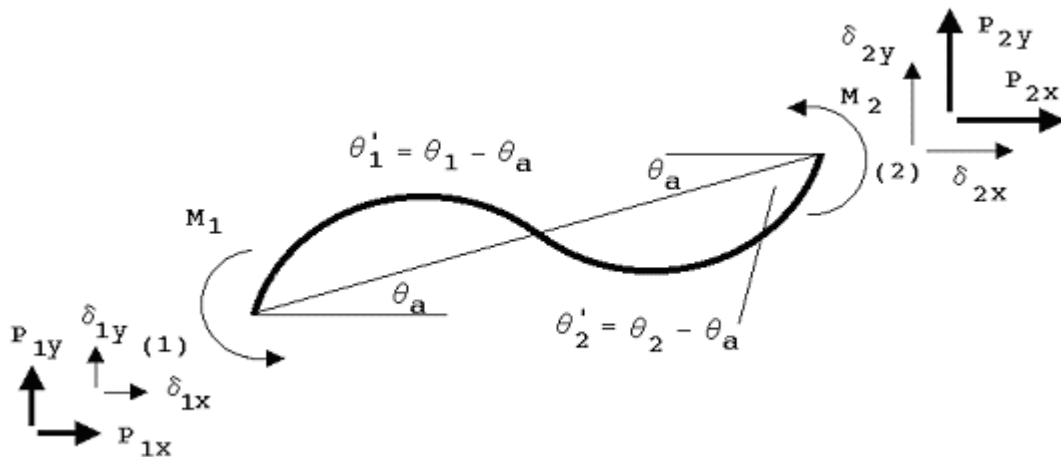
PPP1、PPP2 が要素 1、2 の端点において、それぞれの受け持つ荷重ベクトルをあらわす。

2.2 フレーム(ラーメン)



上の図には、よく、簡単なカーポートなどで見かけるが、**堅型と横型の部材(要素)で端点で互いに固定されるように組み合わせて構造物が示されている。** これは**フレーム(ラーメン)構造**の一例である。これらの構造物を解析する方法は連続ばりの考え方を一歩進めることで組立てることができる。

(1) 要素の剛性方程式



図に示すように、はりが端点に水平・垂直方向の荷重 P_{1x} 、 P_{1y} 、 P_{2x} 、 P_{2y} とモーメント M_1 、 M_2 を受けて、端点(1)と(2)がそれぞれ、水平・垂直方向に $1x$ 、 $1y$ 、 $2x$ 、 $2y$ 変位し、また、回転角がそれぞれ 1 、 2 となったとする。力・モーメントと変位・角度の関係は連続ばりの要素剛性方程式に軸力による力と変位の関係をもぐりこませることによって、次のようになる。

$$\begin{bmatrix} P_{1x} \\ P_{1y} \\ M_1 \\ P_{2x} \\ P_{2y} \\ M_2 \end{bmatrix} = \begin{bmatrix} A \cdot E & 0 & 0 & -A \cdot E & 0 & 0 \\ 0 & \frac{12 \cdot E \cdot I}{L^3} & \frac{6 \cdot E \cdot I}{L^2} & 0 & \frac{-12 \cdot E \cdot I}{L^3} & \frac{6 \cdot E \cdot I}{L^2} \\ 0 & \frac{6 \cdot E \cdot I}{L^2} & \frac{4 \cdot E \cdot I}{L} & 0 & \frac{-6 \cdot E \cdot I}{L^2} & \frac{2 \cdot E \cdot I}{L} \\ -A \cdot E & 0 & 0 & A \cdot E & 0 & 0 \\ 0 & \frac{-12 \cdot E \cdot I}{L^3} & \frac{-6 \cdot E \cdot I}{L^2} & 0 & \frac{12 \cdot E \cdot I}{L^3} & \frac{-6 \cdot E \cdot I}{L^2} \\ 0 & \frac{6 \cdot E \cdot I}{L^2} & \frac{2 \cdot E \cdot I}{L} & 0 & \frac{-6 \cdot E \cdot I}{L^2} & \frac{4 \cdot E \cdot I}{L} \end{bmatrix} \cdot \begin{bmatrix} \delta_{1x} \\ \delta_{1y} \\ \theta_1 \\ \delta_{2x} \\ \delta_{2y} \\ \theta_2 \end{bmatrix}$$

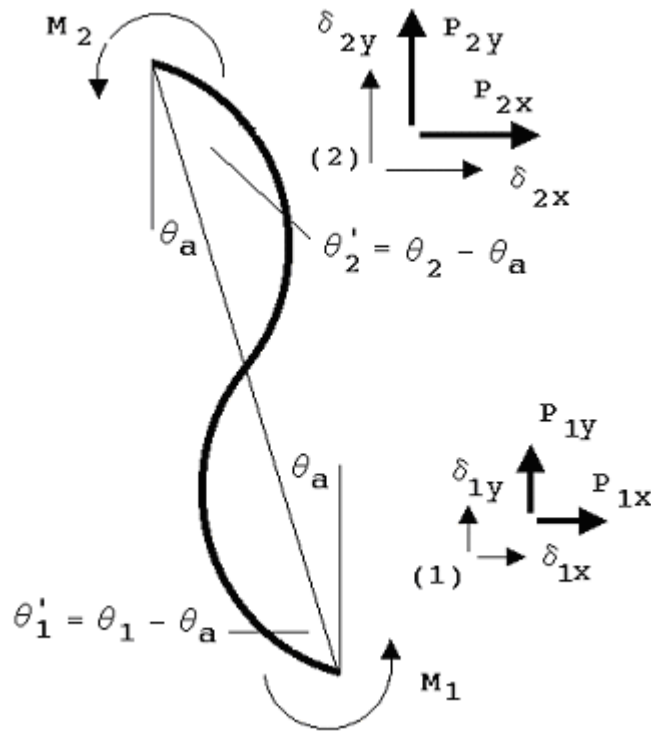
実際の計算で使い易いように次のようにまとめる。接点（要素の端点）の変位と回転は u であらわし、サブスクリプト 1, 2 番目ではじまるもの（ $2n-1$ と $2n$ ）は変位、3 番目のもの（ $2n+1$ ）は回転とする。接点（要素の端点）の力とモーメントは P 表し、サブスクリプトが 1, 2 番目ではじまるもの（ $2n-1$ と $2n$ ）は力、3 番目のもの（ $2n+1$ ）はモーメントとする。

$$\mu_{ie} := \frac{A_{ie} \cdot (L_{ie})^2}{2 \cdot I_{ie}}$$

$$\begin{bmatrix} P1 \\ P2 \\ P3 \\ P4 \\ P5 \\ P6 \end{bmatrix} = \frac{2 \cdot I_{ie} \cdot E_{ie}}{(L_{ie})^3} \cdot \begin{bmatrix} \mu_{ie} & 0 & 0 & -\mu_{ie} & 0 & 0 \\ 0 & 6 & 3 \cdot L_{ie} & 0 & -6 & 3 \cdot L_{ie} \\ 0 & 3 \cdot L_{ie} & 2 \cdot (L_{ie})^2 & 0 & -3 \cdot L_{ie} & (L_{ie})^2 \\ -\mu_{ie} & 0 & 0 & \mu_{ie} & 0 & 0 \\ 0 & -6 & -3 \cdot L_{ie} & 0 & 6 & -3 \cdot L_{ie} \\ 0 & 3 \cdot L_{ie} & (L_{ie})^2 & 0 & -3 \cdot L_{ie} & 2 \cdot (L_{ie})^2 \end{bmatrix} \cdot \begin{bmatrix} u1 \\ u2 \\ u3 \\ u4 \\ u5 \\ u6 \end{bmatrix}$$

上は水平の部材の要素剛性方程式である。端点 (1) は左、端点 (2) は右という方向にたいする約束に注意してほしい。

つぎに、フレームには下の図に示すように、垂直に立ち上がる部材がある。



これについても、水平の場合と同じ考察をくわえると、座標軸の順番、方向を注意して要素剛性方程式が得られる。

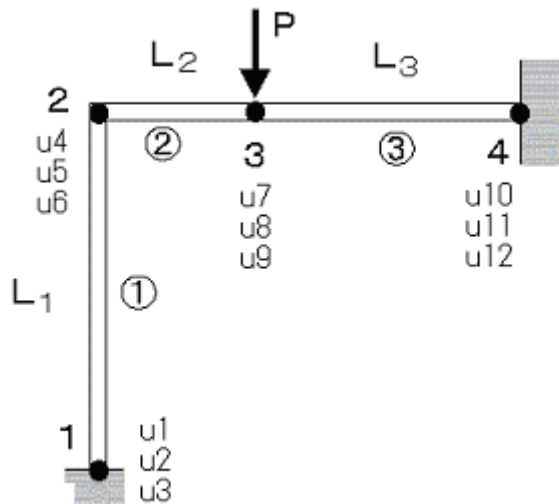
$$\mu_{ie} := \frac{A_{ie} \cdot (L_{ie})^2}{2 \cdot I_{ie}}$$

$$\begin{bmatrix} P1 \\ P2 \\ P3 \\ P4 \\ P5 \\ P6 \end{bmatrix} = \frac{2 \cdot I_{ie} \cdot E_{ie}}{(L_{ie})^3} \cdot \begin{bmatrix} 6 & 0 & -3 \cdot L_{ie} & -6 & 0 & -3 \cdot L_{ie} \\ 0 & \mu_{ie} & 0 & 0 & -\mu_{ie} & 0 \\ -3 \cdot L_{ie} & 0 & 2 \cdot (L_{ie})^2 & 3 \cdot L_{ie} & 0 & (L_{ie})^2 \\ -6 & 0 & 3 \cdot L_{ie} & 6 & 0 & 3 \cdot L_{ie} \\ 0 & -\mu_{ie} & 0 & 0 & \mu_{ie} & 0 \\ -3 \cdot L_{ie} & 0 & (L_{ie})^2 & 3 \cdot L_{ie} & 0 & 2 \cdot (L_{ie})^2 \end{bmatrix} \begin{bmatrix} u1 \\ u2 \\ u3 \\ u4 \\ u5 \\ u6 \end{bmatrix}$$

上は垂直に立ち上がる部材の剛性方程式である。 端点 (1) は下、端点 (2)は上という方向にたいする条件に注意してほしい。

剛性方程式が水平部材と垂直部材に作成されたので、それらの要素剛性マトリクスを計算して全体に組立てるとフレーム（ラーメン）の強度計算に必要な諸元が計算される。

(2) フレーム（ラーメン）の例



(1) で要素の剛性方程式を求めることができたので、連続ばりとおなじ要領で定式化する。
上の例をとって進める。

図において柱は下端点で基礎に節点1で固定されており、上端点ではりの要素と節点2で固定されている。はりの要素ととは本来一体物であるが、荷重の働く点を節点3とするとこの点の変位、回転、荷重、モーメントが求まり都合がよい。節点を任意にとればその変位、回転、荷重、モーメントが求まる。要素の右端点は節点4で右の基礎に固定されている。接点3にPの荷重を受ける場合の各節点の変位と回転、力およびモーメントを求める。なお、数値が見やすいように、最初はL、I、E、Aは1とする。

データのインプット

$$\begin{bmatrix} E \\ A \\ I \\ n_{el} \end{bmatrix} := \begin{bmatrix} 1 \\ 1 \\ 1 \\ 3 \end{bmatrix} \quad \begin{bmatrix} L_1 \\ L_2 \\ L_3 \end{bmatrix} := \begin{bmatrix} 2 \\ 1 \\ 1 \end{bmatrix} \quad ie := 1..n_{el} \quad A_{ie} := A \quad I_{ie} := I \quad E_{ie} := E$$

$$n_{nodes} := 4$$

要素の数、1から要素の数までの範囲を示す数値ie、要素の断面積A、長さL、縦弾性係数Eが与えられる。さらに、Topで各要素の端点(1)、(2)に対する全体の接点の対応が右辺のマトリクスの行の順番に与えられる。また部材が垂直部材か水平部材であるかをしめすcos値とsin値も与えられる。

$$Top := \begin{bmatrix} 1 & 2 \\ 2 & 3 \\ 3 & 4 \end{bmatrix} \quad \begin{bmatrix} cs_1 \\ cs_2 \\ cs_3 \end{bmatrix} := \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \quad \begin{bmatrix} sn_1 \\ sn_2 \\ sn_3 \end{bmatrix} := \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

要素の剛性マトリクスの作成

$$\begin{aligned}
\mathbf{Kel}_{ie} := & \frac{2 \cdot I_{ie} \cdot E_{ie}}{(L_{ie})^3} \cdot \begin{bmatrix} \mu_{ie} & 0 & 0 & -\mu_{ie} & 0 & 0 \\ 0 & 6 & 3 \cdot L_{ie} & 0 & -6 & 3 \cdot L_{ie} \\ 0 & 3 \cdot L_{ie} & 2 \cdot (L_{ie})^2 & 0 & -3 \cdot L_{ie} & (L_{ie})^2 \\ -\mu_{ie} & 0 & 0 & \mu_{ie} & 0 & 0 \\ 0 & -6 & -3 \cdot L_{ie} & 0 & 6 & -3 \cdot L_{ie} \\ 0 & 3 \cdot L_{ie} & (L_{ie})^2 & 0 & -3 \cdot L_{ie} & 2 \cdot (L_{ie})^2 \end{bmatrix} \cdot (\cos_{ie} = 1) \dots \\
& + \frac{2 \cdot I_{ie} \cdot E_{ie}}{(L_{ie})^3} \cdot \begin{bmatrix} 6 & 0 & -3 \cdot L_{ie} & -6 & 0 & -3 \cdot L_{ie} \\ 0 & \mu_{ie} & 0 & 0 & -\mu_{ie} & 0 \\ -3 \cdot L_{ie} & 0 & 2 \cdot (L_{ie})^2 & 3 \cdot L_{ie} & 0 & (L_{ie})^2 \\ -6 & 0 & 3 \cdot L_{ie} & 6 & 0 & 3 \cdot L_{ie} \\ 0 & -\mu_{ie} & 0 & 0 & \mu_{ie} & 0 \\ -3 \cdot L_{ie} & 0 & (L_{ie})^2 & 3 \cdot L_{ie} & 0 & 2 \cdot (L_{ie})^2 \end{bmatrix} \cdot (\cos_{ie} = 0)
\end{aligned}$$

要素の剛性マトリクスの式が上のように入られる。cos 値が1か0かによって、水平・垂直の条件を式に含める。実際の数値はしたようになる。

$$\mathbf{Kel}_1 = \begin{bmatrix} 1.5 & 0 & -1.5 & -1.5 & 0 & -1.5 \\ 0 & 0.5 & 0 & 0 & -0.5 & 0 \\ -1.5 & 0 & 2 & 1.5 & 0 & 1 \\ -1.5 & 0 & 1.5 & 1.5 & 0 & 1.5 \\ 0 & -0.5 & 0 & 0 & 0.5 & 0 \\ -1.5 & 0 & 1 & 1.5 & 0 & 2 \end{bmatrix} \quad \mathbf{Kel}_2 = \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 12 & 6 & 0 & -12 & 6 \\ 0 & 6 & 4 & 0 & -6 & 2 \\ -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & -12 & -6 & 0 & 12 & -6 \\ 0 & 6 & 2 & 0 & -6 & 4 \end{bmatrix} \quad \mathbf{Kel}_3 = \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 12 & 6 & 0 & -12 & 6 \\ 0 & 6 & 4 & 0 & -6 & 2 \\ -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & -12 & -6 & 0 & 12 & -6 \\ 0 & 6 & 2 & 0 & -6 & 4 \end{bmatrix}$$

全体の剛性マトリクスの作成

$$\begin{aligned}
n & := 3 \cdot n_{nodes} & \mathbf{K}_{n,n} & := 0 & ie & := 1.. n_{el} & i & := 1.. 2 & j & := 1.. 2 \\
p & := 1.. 3 & q & := 1.. 3
\end{aligned}$$

$$\mathbf{K}[(\text{Top}_{ie,i}-1) \cdot 3+p], (\text{Top}_{ie,j}-1) \cdot 3+q := \mathbf{K}[(\text{Top}_{ie,i}-1) \cdot 3+p], (\text{Top}_{ie,j}-1) \cdot 3+q \dots + (\mathbf{Kel}_{ie})_{(i-1) \cdot 3+p, (j-1) \cdot 3+q}$$

演算のレンジが指定され、要素の剛性マトリクスの内容が節点ごとにまとめられて、全体の剛性マトリクス \mathbf{K} が求められる。今回は各節点には3個の自由度があるので、p と q は1から3にわたってまとめられている。

境界条件と荷重条件のインプット

$$\text{Restrains} := \begin{bmatrix} 1 & 0 \\ 2 & 0 \\ 3 & 0 \\ 10 & 0 \\ 11 & 0 \\ 12 & 0 \end{bmatrix} \quad n_{\text{res}} := \text{rows}(\text{Restrains}) \quad \text{Loads} := (8 - 1)$$

拘束条件すなわち、変位 u_1 、 u_2 、 u_3 、 u_{10} 、 u_{11} 、 u_{12} が既知数 0 で、荷重は荷重番号 8 に（接点 3 の y 方向）- 1 であると指定される。 また、拘束件数が求められる。

境界条件による全体剛性マトリクスと荷重ベクトルの修正

$$p := 1..nn_{pe} \quad nn_{pe} = 2$$

$$indp := 1..3 \cdot n_{nodes} \quad indj := 1..3 \cdot n_{nodes} \quad n_{res} = 6 \quad id := 1..n_{res}$$

$$K_{mod} := K$$

$$K_{mod}_{Restrains_{id,1}, indj} := 0$$

$$K_{mod}_{indj, Restrains_{id,1}} := 0$$

$$K_{mod}_{(Restrains_{id,1}), Restrains_{id,1}} := 1$$

$$f_{mod}_{indp} := 0$$

$$f_{mod}_{(Loads_{il,1})} := f_{mod}_{(Loads_{il,1})} + Loads_{il,2}$$

$$f_{mod}_{indp} := f_{mod}_{indp} - K_{indp, Restrains_{id,1}} \cdot Restrains_{id,2}$$

$$f_{mod}_{(Restrains_{id,1})} := Restrains_{id,2}$$

演算のレンジが指定され、全体剛性マトリクスが修正されて K_{mod} となる。 荷重マトリクスが修正されて f_{mod} となる。

変位ベクトル u は次のように求められる。

$$u := K_{\text{mod}}^{-1} \cdot f_{\text{mod}} \quad u = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -0.054 \\ -0.304 \\ 0.071 \\ -0.027 \\ -0.176 \\ 0.21 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

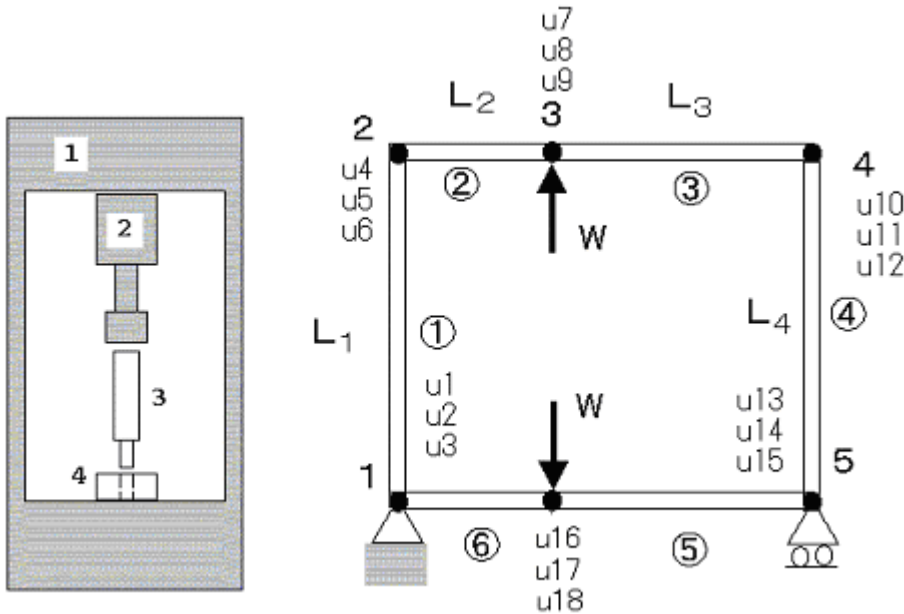
得られた変位 u のベクトルを、各要素の端点 (1)、(2) に割り当てて、それぞれの要素の剛性マトリクスをかけると、その要素の端点にたいしてその要素が受け持つ x と y 方向の力とモーメントが求められる。

$$p_{ie} := \text{Top}_{ie,1} \quad r_{ie} := \text{Top}_{ie,2} \quad i := 1..n_{\text{nodes}}$$

$$uu_i := u_{3i-2} \quad vv_i := u_{3i-1} \quad qq_i := u_{3i}$$

$$PP_{ie} := Ke_{ie} \cdot \begin{bmatrix} uu(p_{ie}) \\ vv(p_{ie}) \\ qq(p_{ie}) \\ uu(r_{ie}) \\ vv(r_{ie}) \\ qq(r_{ie}) \end{bmatrix} \quad PP_1 = \begin{bmatrix} -0.027 \\ 0.152 \\ -8.929 \cdot 10^{-3} \\ 0.027 \\ -0.152 \\ 0.062 \end{bmatrix} \quad PP_2 = \begin{bmatrix} -0.027 \\ 0.152 \\ -0.062 \\ 0.027 \\ -0.152 \\ 0.214 \end{bmatrix} \quad PP_3 = \begin{bmatrix} -0.027 \\ -0.848 \\ -0.214 \\ 0.027 \\ 0.848 \\ -0.634 \end{bmatrix}$$

(3) 構造物の例 (簡易プレスのフレーム)



図は一種の簡易プレスを示している。油圧ジャッキ2をつかって、軸3をホイール4の穴に押し込む場合のフレームとして、型鋼を額縁状に組み合わせフレーム1として油圧ジャッキの上向きの反力を上部のビーム（クラウンという）で、ホイールにかかる下向きの反力を下部のビーム（ベッドという）で受ける。クラウンとベッドを結びつける両脇に部材をアブライトまたはコラムという。このフレーム構造物に早速 FEM を使っていくらかの節点での曲げ応力を計算してみよう。

鋼の縦弾性係数、コラムに使用する型鋼を I 100 x 50 x 5 x 7 とし、クラウンとベッドに使用する型鋼を I 250 x 250 x 9 x 14 とすると断面積 (cm²) と断面二次モーメント (cm⁴) は鋼材のカタログの表から求められる。Top で各部材の端部の節点番号を与える。また、cs と sn の行列で各要素の傾きの cos と sin が (すなわち垂直部材か水平部材かを示す指標) が与えられる。また、ex1 と ex2 は型鋼の高さの半分 (cm) を示しこれはのちに曲げ応力算定のときに使用される。

$$\begin{bmatrix} E \\ A1 \\ A2 \\ I1 \\ I2 \\ n_{el} \\ n_{nodes} \end{bmatrix} := \begin{bmatrix} 2.06 \cdot 10^7 \\ 11.85 \\ 91.43 \\ 187 \\ 10700 \\ 6 \\ 6 \end{bmatrix}$$

$$\text{Top} := \begin{bmatrix} 1 & 2 \\ 2 & 3 \\ 3 & 4 \\ 5 & 4 \\ 6 & 5 \\ 1 & 6 \end{bmatrix}$$

$$\begin{bmatrix} cs_1 & sn_1 \\ cs_2 & sn_2 \\ cs_3 & sn_3 \\ cs_4 & sn_4 \\ cs_5 & sn_5 \\ cs_6 & sn_6 \end{bmatrix} := \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 0 \end{bmatrix}$$

$$ie := 1..n_{el}$$

$$E_{ie} := E \quad ex1 := 5 \quad ex2 := 12.5$$

各部材の断面積、断面二次モーメント、長さを与える。

$$\begin{bmatrix} A_1 & I_1 & L_1 \\ A_4 & I_4 & L_4 \end{bmatrix} := \begin{bmatrix} A1 & I1 & 80 \\ A1 & I1 & 80 \end{bmatrix} \quad \begin{bmatrix} A_2 & I_2 & L_2 \\ A_3 & I_3 & L_3 \\ A_5 & I_5 & L_5 \\ A_6 & I_6 & L_6 \end{bmatrix} := \begin{bmatrix} A2 & I2 & 40 \\ A2 & I2 & 40 \\ A2 & I2 & 40 \\ A2 & I2 & 40 \end{bmatrix}$$

プログラムにしたがって、与えられたデータに従って要素マトリクスが作られ、さらに全体マトリクスが組立てられる。次に、拘束条件すなわち、変位 u_1 、 u_2 、 u_4 が既知数 0 で、荷重は荷重番号 8 (接点 3 の y 方向) 100000 N、荷重番号 17 に (接点 6 の y 方向) -100000 N と指定される。また、拘束件数も求められる。

$$W := 100000$$

$$\text{Restrains} := \begin{bmatrix} 1 & 0 \\ 2 & 0 \\ 14 & 0 \end{bmatrix} \quad \text{Loads} := \begin{bmatrix} 8 & W \\ 17 & -W \end{bmatrix}$$

$$n_{\text{res}} := \text{rows}(\text{Restrains})$$

プログラムは境界条件による全体剛性マトリクスと荷重ベクトルの修正を自動的に行い、全体剛性マトリクスが修正されて K_{mod} となる。荷重マトリクスが修正されて f_{mod} となる。 K_{mod} の逆行列を f_{mod} に掛けると変位ベクトル u が求められる。変位ベクトルは見やすくするため、行列の形であらわし、第一列目に節点番号、2, 3, 4 列目にそれぞれ、x 方向の変位、y 方向の変位、および回転角 (ラジアン) を示す。

$$\text{ind} := 1..n_{\text{nodes}} \quad u_{x_{\text{ind}}} := u_{3 \cdot \text{ind} - 2} \quad u_{y_{\text{ind}}} := u_{3 \cdot \text{ind} - 1} \quad u_{\theta_{\text{ind}}} := u_{3 \cdot \text{ind}}$$

$$u_{\text{all}_{\text{ind},1}} := \text{ind} \quad u_{\text{all}_{\text{ind},2}} := u_{x_{\text{ind}}} \quad u_{\text{all}_{\text{ind},3}} := u_{y_{\text{ind}}} \quad u_{\text{all}_{\text{ind},4}} := u_{\theta_{\text{ind}}}$$

$$u_{\text{all}} = \begin{bmatrix} 1 & 0 & 0 & -1.784 \cdot 10^{-4} \\ 2 & 0 & 0.016 & 1.784 \cdot 10^{-4} \\ 3 & 0 & 0.021 & 0 \\ 4 & 0 & 0.016 & -1.784 \cdot 10^{-4} \\ 5 & 0 & 0 & 1.784 \cdot 10^{-4} \\ 6 & 0 & -4.777 \cdot 10^{-3} & 0 \end{bmatrix}$$

得られた変位 u のベクトルを、各要素の端点 (1)、(2) に割り当てて、それぞれの要素の剛性マトリクスをかけると、その要素の端点にたいしてその要素が受け持つ x と y 方向の力とモーメントが求められる。荷重ベクトルは見やすくするため、行列の形であらわし、第一列目に要素番号 1 の端点 1 と 2 の x 方向の力、y 方向の力、および曲げモーメントを示す。第 2, 3, 4, 5, 6 列目はそれぞれ要素番号 2, 3, 4, 5, 6 に対応する。

$$p_{\text{ie}} := \text{Top}_{\text{ie},1} \quad r_{\text{ie}} := \text{Top}_{\text{ie},2} \quad i := 1..n_{\text{nodes}} \quad uu_i := u_{3i-2} \quad vv_i := u_{3i-1} \quad qq_i := u_{3i}$$

$$PP_{ie} := KeI_{ie} \cdot \begin{bmatrix} uu(p_{ie}) \\ vv(p_{ie}) \\ qq(p_{ie}) \\ uu(r_{ie}) \\ vv(r_{ie}) \\ qq(r_{ie}) \end{bmatrix} \quad P^{<ie>} := PP_{ie} \quad i := 1..6 \quad j := 1..n_{el} \quad P_{i,j} := P_{i,j} \cdot (|P_{i,j}| > 0.01)$$

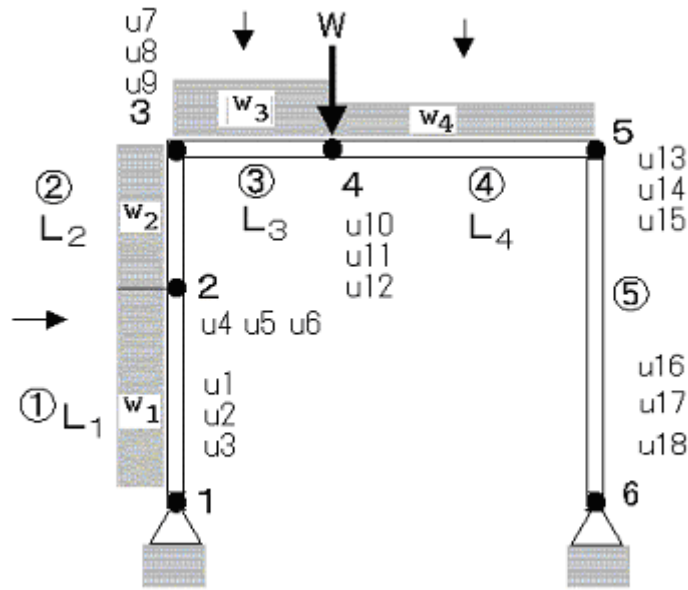
$$P = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ -5 \cdot 10^4 & -5 \cdot 10^4 & 5 \cdot 10^4 & -5 \cdot 10^4 & -5 \cdot 10^4 & 5 \cdot 10^4 \\ -1.718 \cdot 10^4 & -1.718 \cdot 10^4 & 1.983 \cdot 10^6 & 1.718 \cdot 10^4 & -1.983 \cdot 10^6 & 1.718 \cdot 10^4 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 5 \cdot 10^4 & 5 \cdot 10^4 & -5 \cdot 10^4 & 5 \cdot 10^4 & 5 \cdot 10^4 & -5 \cdot 10^4 \\ 1.718 \cdot 10^4 & -1.983 \cdot 10^6 & 1.718 \cdot 10^4 & -1.718 \cdot 10^4 & -1.718 \cdot 10^4 & 1.983 \cdot 10^6 \end{bmatrix}$$

クラウンの荷重のかかる部分(変位番号 9)とベッドの荷重のかかる部分(変位番号 18)の曲げ応力はしたのように求められる。(単位は N/cm²) なお、アブライトの引張り応力はWを断面積の2倍で割って(左右対称だから)あわせて求められている。

$$\sigma_9 := \frac{(PP_2)_6}{I_2} \cdot ex_2 \quad \sigma_9 = -2.316 \cdot 10^3 \quad \frac{W}{A_1 \cdot 2} = 4.219 \cdot 10^3$$

$$\sigma_{18} := \frac{(PP_6)_6}{I_2} \cdot ex_2 \quad \sigma_{18} = 2.316 \cdot 10^3$$

(4) 等分布荷重を含む場合



上図のように部材が等分布荷重を受ける場合には、連続ばりにおいて、その部材内部両端点に相当荷重と相当モーメントが加わっているとして境界条件を作った。 フレームにおいても同じ考え方をを用いるが、荷重ベクトル、変位ベクトルの行が増す分と垂直部材が増えた分を勘案して内力ベクトルを作成する。 この例ではある部材に等分布荷重がかかり、またある節点には外力が働くより一般的にした問題を考えよう。 鋼の縦弾性係数、すべての部材は型鋼 I 100 x 50 x 7 で構成されるものとする。

$$\begin{bmatrix} E \\ A1 \\ I1 \\ n_{el} \\ n_{nodes} \end{bmatrix} := \begin{bmatrix} 2.06 \cdot 10^7 \\ 11.85 \\ 187 \\ 5 \\ 6 \end{bmatrix} \quad \text{Top} := \begin{bmatrix} 1 & 2 \\ 2 & 3 \\ 3 & 4 \\ 4 & 5 \\ 6 & 5 \end{bmatrix} \quad \begin{bmatrix} cs_1 & sn_1 \\ cs_2 & sn_2 \\ cs_3 & sn_3 \\ cs_4 & sn_4 \\ cs_5 & sn_5 \end{bmatrix} := \begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$ie := 1..n_{el} \quad E_{ie} := E$$

各部材の断面積、断面二次モーメント、長さが与えられる。

$$\begin{bmatrix} A_1 & I_1 & L_1 \\ A_2 & I_2 & L_2 \\ A_3 & I_3 & L_3 \\ A_4 & I_4 & L_4 \\ A_5 & I_5 & L_5 \end{bmatrix} := \begin{bmatrix} A1 & I1 & 200 \\ A1 & I1 & 200 \\ A1 & I1 & 200 \\ A1 & I1 & 200 \\ A1 & I1 & 400 \end{bmatrix}$$

上に与えられたデータに従って要素剛性マトリクスが作られ、さらに全体剛性マトリクスが自動的に組立てられる。 次に、拘束条件すなわち、変位 u1、u2、u16、u17 が既知数 0 である

と指定る。 また、拘束件数も求める。

$$\text{Restrains} := \begin{bmatrix} 1 & 0 \\ 2 & 0 \\ 16 & 0 \\ 17 & 0 \end{bmatrix} \quad n_{\text{res}} := \text{rows}(\text{Restrains})$$

次に等分布荷重による部材端相当内部荷重ベクトルを作成する。ここでは一般化してすべての部材に等分布荷重がかかるように式を作成し、あらためて要素の等分布荷重の w 値が限りなく 0 にちかいものとしている。また、部材が水平か垂直かの条件を式にふくめて要素の内力ベクトル intforce を作成している。

$$\begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \end{bmatrix} := \begin{bmatrix} 3 \\ 3 \\ 5 \\ 5 \\ 10^{-20} \end{bmatrix} \quad \text{intforce}_{ie} := \frac{w_{ie} \cdot L_{ie}}{12} \cdot \begin{bmatrix} 0 \\ -6 \\ -L_{ie} \\ 0 \\ -6 \\ L_{ie} \end{bmatrix} \cdot (\text{cs}_{ie} = 1) + \frac{w_{ie} \cdot L_{ie}}{12} \cdot \begin{bmatrix} -6 \\ 0 \\ -L_{ie} \\ -6 \\ 0 \\ L_{ie} \end{bmatrix} \cdot (\text{cs}_{ie} = 0)$$

要素の内力ベクトルを全体の内力ベクトルへと組立てる。まず、要素の内力ベクトルの上 3 行と下 3 行をそれぞれの端点の対応するグローバル変位番号をしめす intf というベクトルを作成する。

$$p_{ie} := \text{Top}_{ie,1} \quad r_{ie} := \text{Top}_{ie,2} \quad \text{intf}_{ie} := \begin{bmatrix} 3 \cdot p_{ie} - 2 \\ 3 \cdot p_{ie} - 1 \\ 3 \cdot p_{ie} \\ 3 \cdot r_{ie} - 2 \\ 3 \cdot r_{ie} - 1 \\ 3 \cdot r_{ie} \end{bmatrix}$$

さらに、ベクトル intf に 1 列を追加して、第 1 列で変位番号のベクトルを、第 2 列で対応する内力ベクトルを示す行列とする。

$$\text{inf}_{ie} := \text{augment}(\text{intf}_{ie}, \text{intforce}_{ie})$$

異なる要素の内力ベクトルで、変位番号を同じくするものは加算させる必要があるので、 INF という行列を作り、全体の内力ベクトルを組立てて、さらにそれらを整列（ソート）させて、第 1 列で変位番号のベクトルを、第 2 列で対応する内力ベクトルの合計を表わす。

$$i := 1..6 \quad \text{INF}_{3 \cdot n_{\text{nodes}}, 2} := 0 \quad \text{INF}_{(\text{inf}_{ie})_{i,1}, 1} := (\text{inf}_{ie})_{i,1} \quad \text{INF}_{(\text{inf}_{ie})_{i,1}, 2} := \text{INF}_{(\text{inf}_{ie})_{i,1}, 2} + (\text{inf}_{ie})_{i,2} \\ \text{INF} := \text{csort}(\text{INF}, 1)$$

次に、節点にかかる荷重番号と外力とが行列 extforce で与えられる。

$$W := 2000 \quad \text{extforce} := \begin{bmatrix} 11 & -W \\ 10 & 0 \\ 12 & 0 \end{bmatrix} \quad n_{\text{extf}} := \text{rows}(\text{extforce})$$

$$i := 1.. 3 \cdot n_{\text{nodes}} \quad \text{iext} := 1.. n_{\text{extf}} \quad \text{Loads} := \text{INF}$$

$$\text{Loads}_{\text{extforce}_{\text{iext},1},2} := (\text{extforce}_{\text{iext},2}) + \text{INF}_{(\text{extforce}_{\text{iext},1}),2}$$

内力の行列 INF と外力の行列 extforce を組み合わせることで、力の行列 Loads が作成される。

$$i := 1.. 3 \cdot n_{\text{nodes}} \quad \text{iext} := 1.. n_{\text{extf}} \quad \text{Loads} := \text{INF}$$

$$\text{Loads}_{\text{extforce}_{\text{iext},1},2} := (\text{extforce}_{\text{iext},2}) + \text{INF}_{(\text{extforce}_{\text{iext},1}),2}$$

境界条件を構成する行列 Restraints と Loads を使って、プログラムによる全体剛性マトリクスと荷重ベクトルの修正を自動的に行い、全体剛性マトリクスが修正されて Kmod となる。荷重マトリクスが修正されて fmod となる。Kmod の逆行列を fmod に掛けると変位ベクトル u が求められる。変位ベクトルは見やすくするため、行列の形であらわし、第一列目に節点番号、2, 3, 4 列目にそれぞれ、x 方向の変位、y 方向の変位、および回転角 (ラジアン) を示す。

$$\text{ind} := 1.. n_{\text{nodes}} \quad \text{ux}_{\text{ind}} := u_{3 \cdot \text{ind} - 2} \quad \text{uy}_{\text{ind}} := u_{3 \cdot \text{ind} - 1} \quad \text{u}\theta_{\text{ind}} := u_{3 \cdot \text{ind}}$$

$$\text{uall}_{\text{ind},1} := \text{ind} \quad \text{uall}_{\text{ind},2} := \text{ux}_{\text{ind}} \quad \text{uall}_{\text{ind},3} := \text{uy}_{\text{ind}} \quad \text{uall}_{\text{ind},4} := \text{u}\theta_{\text{ind}} \quad i := 1.. n_{\text{nodes}}$$

$$\text{uall} = \begin{bmatrix} 1 & 0 & 0 & 0.011 \\ 2 & -2.062 & -2.13 \cdot 10^{-3} & 7.68 \cdot 10^{-3} \\ 3 & -2.702 & -4.26 \cdot 10^{-3} & -1.691 \cdot 10^{-3} \\ 4 & -2.702 & -0.64 & -1.033 \cdot 10^{-3} \\ 5 & -2.702 & -2.294 \cdot 10^{-3} & 5.854 \cdot 10^{-3} \\ 6 & 0 & 0 & 7.204 \cdot 10^{-3} \end{bmatrix}$$

得られた変位 u のベクトルを、各要素の端点 (1)、(2) に割り当てて、それぞれの要素の剛性マトリクスをかけると、その要素の端点にたいしてその要素が受け持つ x と y 方向の力とモーメントが求められる。荷重ベクトルは見やすくするため、行列の形であらわし、第一列目に要素番号 1 の端点 1 と 2 の x 方向の力、y 方向の力、および曲げモーメントを示す。第 2, 3, 4, 5, 6 列目はそれぞれ要素番号 2, 3, 4, 5, 6 に対応する。

$$p_{\text{ie}} := \text{Top}_{\text{ie},1} \quad r_{\text{ie}} := \text{Top}_{\text{ie},2}$$

$$\text{uu}_i := u_{3 \cdot i - 2} \quad \text{vv}_i := u_{3 \cdot i - 1} \quad \text{qq}_i := u_{3 \cdot i}$$

$$PP_{ie} := KeI_{ie} \cdot \begin{bmatrix} uu(p_{ie}) \\ vv(p_{ie}) \\ qq(p_{ie}) \\ uu(r_{ie}) \\ vv(r_{ie}) \\ qq(r_{ie}) \end{bmatrix} \quad P^{<ie>} := PP_{ie} - \text{intforce}_{ie}$$

$$P = \begin{bmatrix} 1.135 \cdot 10^3 & 535.004 & -64.996 & -64.996 & 64.996 \\ 2.6 \cdot 10^3 & 2.6 \cdot 10^3 & 2.6 \cdot 10^3 & -400 & 1.4 \cdot 10^3 \\ 5.093 \cdot 10^{-11} & 1.67 \cdot 10^5 & 2.14 \cdot 10^5 & -2.06 \cdot 10^5 & -6.148 \cdot 10^{-11} \\ -535.004 & 64.996 & 64.996 & 64.996 & -64.996 \\ -2.6 \cdot 10^3 & -2.6 \cdot 10^3 & -1.6 \cdot 10^3 & 1.4 \cdot 10^3 & -1.4 \cdot 10^3 \\ -1.67 \cdot 10^5 & -2.14 \cdot 10^5 & 2.06 \cdot 10^5 & 2.6 \cdot 10^4 & -2.6 \cdot 10^4 \end{bmatrix}$$

水平部材の右端点、垂直部材の上端点の曲げ応力はしたのように求められる。(単位は N/cm²)

$$\sigma_{ie} := \frac{(P^{<ie>})_6}{I_{ie}} \cdot ex \quad \sigma = \begin{bmatrix} -4.465 \cdot 10^3 \\ -5.722 \cdot 10^3 \\ 5.508 \cdot 10^3 \\ 695.146 \\ -695.146 \end{bmatrix}$$

2.3 Excel VBA によるフレーム(ラーメン)のプログラム

これまで、フレームの計算を数学ソフト Mathcad で一般化した、同じように Excel VBA を使ってプログラムを作ろう。まず、入力のための表をつくろう。図の上中央の楕円形のボタンを押せば出力として、各部材の端部 (1)、(2) の受ける力と変位変位が求まるものとする。

要素番号	A (cm ²)	I (cm ⁴)	L (cm)	cos(a)	sin(a)	Topology	w (N/cm)	拘束番号	拘束値	荷重番号	外力(N)
1	11.85	187.00	200.00	0.00	1.00	1 2	3.00	1	0	11	-2000
2	11.85	187.00	200.00	0.00	1.00	2 3	3.00	2	0		
3	11.85	187.00	200.00	1.00	0.00	3 4	5.00	16	0		
4	11.85	187.00	200.00	1.00	0.00	4 5	5.00	17	0		
5	11.85	187.00	400.00	0.00	1.00	6 5	0.00				

次に出力の表をしたの図のように入力の表の右側に作成する。要素の数が増えてもよいように、表の下側に余裕をもたすために入力・出力の表を幅広のものとした。

要素番号	端点にかかる力						変位					
	端点1			端点2			端点1			端点2		
	Px	Py	M	Px	Py	M	u1x	u1y	u1θ	u2x	u2y	u2θ
N	N	Ncm	N	N	Ncm	cm	cm	rad	cm	cm	rad	
1	1135	2600	0	-535	-2600	-167001	0.000	0.000	0.011	-2.062	-0.002	0.008
2	535	2600	167001	65	-2600	-214002	-2.062	-0.002	0.008	-2.702	-0.004	-0.002
3	-65	2600	214002	65	-1600	205998	-2.702	-0.004	-0.002	-2.702	-0.640	-0.001
4	-65	-400	-205998	65	1400	25998	-2.702	-0.640	-0.001	-2.702	-0.002	0.006
5	65	1400	0	-65	-1400	-25998	0.000	0.000	0.007	-2.702	-0.002	0.006

Sub Macro1()

'Macro1 Macro FRAME FEM by Takuro Uede 00/01/16

Dim nnode, ne, nrest, nextforce, nload, ie, i, j, id, jd, p, q, r

Dim top(100, 2), restraints(100, 2), we(100), intforce(100, 6), extforce(100, 2)

Dim intf(100, 6), inf(100, 6, 2), inloads(100, 2), loads(100, 2)

Dim E, cs(100), sn(100), area(100), AI(100), L(100)

Dim dKel, Kel(100, 6, 6), KG(300, 300)

Dim Kmod(300, 300), f(300), fmod(300)

Dim a(300, 300), cc(300), u(300), s, Force(100), elong(100), uu(100, 3)

Dim ue(100, 6), pp(100, 6)

'インプット (1) -----

E = Cells(3, 10)

nnode = Cells(3, 2): ne = Cells(3, 5): nrest = Cells(4, 2): nextforce = Cells(4, 5)

'データインプット (2) -----

```
For i = 1 To ne
cs(i) = Cells(i + 6, 5): sn(i) = Cells(i + 6, 6)
area(i) = Cells(i + 6, 2): AI(i) = Cells(i + 6, 3): L(i) = Cells(i + 6, 4)
```

```
For j = 1 To 2
top(i, j) = Cells(i + 6, j + 6)
Next
```

```
we(i) = Cells(i + 6, 9)
```

```
Next
```

```
For i = 1 To nrest
For j = 1 To 2
restraints(i, j) = Cells(i + 6, j + 9)
Next
Next
```

```
For i = 1 To nextforce
For j = 1 To 2
extforce(i, j) = Cells(i + 6, j + 11)
Next
Next
```

```
'要素剛性マトリクス Kel 作成サブルーチン呼び出し -----
```

```
For ie = 1 To ne
If cs(ie) = 1 Then
elementstifmat1 Kel(), ie, area(), AI(), E, L()
End If
If cs(ie) = 0 Then
elementstifmat2 Kel(), ie, area(), AI(), E, L()
End If
Next
```

```
'全体剛性マトリクス K G 組立てサブルーチン呼び出し-----
```

```
globstifmat nnode, ne, top(), Kel(), KG(), ie
```

```
'内力ベクトルの作成サブルーチン呼び出し-----
```

```
internalforce ne, nnode, L(), we(), cs(), top(), intforce(), intf(), inf(), inloads()
```

```
'内力と外力を組み合わせる荷重マトリクス作成サブルーチン呼び出し--
```

```
makeloads nnode, nextforce, inloads(), extforce(), loads()
```

```
'拘束条件による K G の修正サブルーチン呼び出し-----
```

```
modifyK KG(), restraints(), Kmod(), nnode, nrest
```

```
'拘束条件を加味した荷重ベクトル作成サブルーチン呼び出し-----
```

```
forcevecmod nnode, nrest, nload, loads(), f(), fmod(), restraints(), KG()
```

```
'連立一次方程式解法サブルーチンの呼び出し-----
```

```
For i = 1 To nnode * 3
cc(i) = fmod(i)
For j = 1 To nnode * 3
a(i, j) = Kmod(i, j)
Next
Next
```

```
matsol a(), cc(), u(), nnode
```

```
'最終処理サブルーチンの呼び出しと表へのアウトプット-----
```

```
postprocess top(), Kel(), u(), uu(), ue(), pp(), intforce(), ne, nnode
For ie = 1 To ne
For i = 1 To 6
Cells(ie + 6, i + 15) = pp(ie, i)
Cells(ie + 6, i + 21) = ue(ie, i)
Next
Next
```

```
Next
End Sub
```

```
Sub elementstifmat1(Kel(), ie, area(), AI(), E, L())
For i = 1 To 6
For j = 1 To 6
Kel(ie, i, j) = 0
Next
Next
mu = area(ie) * L(ie) ^ 2 / 2 / AI(ie)
Kel(ie, 1, 1) = mu: Kel(ie, 1, 4) = -mu
Kel(ie, 2, 2) = 6: Kel(ie, 2, 3) = 3 * L(ie): Kel(ie, 2, 5) = -6
Kel(ie, 2, 6) = 3 * L(ie)
Kel(ie, 3, 2) = 3 * L(ie): Kel(ie, 3, 3) = 2 * L(ie) ^ 2: Kel(ie, 3, 5) = -3 * L(ie)
Kel(ie, 3, 6) = L(ie) ^ 2
Kel(ie, 4, 1) = -mu: Kel(ie, 4, 4) = mu
Kel(ie, 5, 2) = -6: Kel(ie, 5, 3) = -3 * L(ie)
Kel(ie, 5, 5) = 6: Kel(ie, 5, 6) = -3 * L(ie)
Kel(ie, 6, 2) = 3 * L(ie): Kel(ie, 6, 3) = L(ie) ^ 2
Kel(ie, 6, 5) = -3 * L(ie): Kel(ie, 6, 6) = 2 * L(ie) ^ 2
For i = 1 To 6
For j = 1 To 6
Kel(ie, i, j) = 2 * AI(ie) * E / L(ie) ^ 3 * Kel(ie, i, j)
Next
Next
End Sub
```

```
Sub elementstifmat2(Kel(), ie, area(), AI(), E, L())
For i = 1 To 6
For j = 1 To 6
Kel(ie, i, j) = 0
Next
Next
mu = area(ie) * L(ie) ^ 2 / 2 / AI(ie)
Kel(ie, 1, 1) = 6: Kel(ie, 1, 3) = -3 * L(ie): Kel(ie, 1, 4) = -6
Kel(ie, 1, 6) = -3 * L(ie)
Kel(ie, 2, 2) = mu: Kel(ie, 2, 5) = -mu
Kel(ie, 3, 1) = -3 * L(ie): Kel(ie, 3, 3) = 2 * L(ie) ^ 2: Kel(ie, 3, 4) = 3 * L(ie)
Kel(ie, 3, 6) = L(ie) ^ 2
Kel(ie, 4, 1) = -6: Kel(ie, 4, 3) = 3 * L(ie)
Kel(ie, 4, 4) = 6: Kel(ie, 4, 6) = 3 * L(ie)
Kel(ie, 5, 2) = -mu: Kel(ie, 5, 5) = mu
Kel(ie, 6, 1) = -3 * L(ie): Kel(ie, 6, 3) = L(ie) ^ 2
Kel(ie, 6, 4) = 3 * L(ie): Kel(ie, 6, 6) = 2 * L(ie) ^ 2
For i = 1 To 6
For j = 1 To 6
Kel(ie, i, j) = 2 * AI(ie) * E / L(ie) ^ 3 * Kel(ie, i, j)
Next
Next
End Sub
```

```
Sub globstifmat(nnode, ne, top(), Kel(), KG(), ie)

For i = 1 To 3 * nnode
For j = 1 To 3 * nnode
KG(i, j) = 0
Next
Next
For ie = 1 To ne
For i = 1 To 2
For j = 1 To 2
For p = 1 To 3
For q = 1 To 3
id = top(ie, i): 'id of the node at the end (1) or (2) of an element
jd = top(ie, j): 'id of the node at the end (1) or (2) of an element
dKel = Kel(ie, 3 * i - 3 + p, 3 * j - 3 + q)
KG(3 * id - 3 + p, 3 * jd - 3 + q) = KG(3 * id - 3 + p, 3 * jd - 3 + q) + dKel

```

```

Next
Next
Next
Next
Next
End Sub

```

```

Sub internalforce(ne, nnode, L(), we(), cs(), top(), intforce(), intf(), inf(), inloads())
For ie = 1 To ne
intforce(ie, 1) = 0: intforce(ie, 2) = -6: intforce(ie, 3) = -L(ie)
intforce(ie, 4) = 0: intforce(ie, 5) = -6: intforce(ie, 6) = L(ie)
If cs(ie) = 0 Then
intforce(ie, 1) = -6: intforce(ie, 2) = 0
intforce(ie, 4) = -6: intforce(ie, 5) = 0
End If
For i = 1 To 6
intforce(ie, i) = intforce(ie, i) * we(ie) * L(ie) / 12
Next
p = top(ie, 1): r = top(ie, 2)
intf(ie, 1) = 3 * p - 2: intf(ie, 2) = 3 * p - 1: intf(ie, 3) = 3 * p
intf(ie, 4) = 3 * r - 2: intf(ie, 5) = 3 * r - 1: intf(ie, 6) = 3 * r
For i = 1 To 6
inf(ie, i, 1) = intf(ie, i): inf(ie, i, 2) = intforce(ie, i)
Next
Next
For i = 1 To 3 * nnode
For j = 1 To 2
inloads(i, j) = 0
Next
Next
For ie = 1 To ne
For i = 1 To 6
inloads(inf(ie, i, 1), 1) = inf(ie, i, 1)
inloads(inf(ie, i, 1), 2) = inloads(inf(ie, i, 1), 2) + inf(ie, i, 2)
Next
Next
End Sub

```

```

Sub makeloads(nnode, nextforce, inloads(), extforce(), loads())
For i = 1 To 3 * nnode
loads(i, 1) = inloads(i, 1): loads(i, 2) = inloads(i, 2)
Next
For i = 1 To nextforce
loads(extforce(i, 1), 2) = extforce(i, 2) + inloads(extforce(i, 1), 2)
Next
End Sub

```

```

Sub modifyK(KG(), restraints(), Kmod(), nnode, nrest)
For ip = 1 To 3 * nnode
For ij = 1 To 3 * nnode
Kmod(ip, ij) = KG(ip, ij)
Next
Next
For ij = 1 To 3 * nnode
For i = 1 To nrest
Kmod(restraints(i, 1), ij) = 0
Kmod(ij, restraints(i, 1)) = 0
Kmod(restraints(i, 1), restraints(i, 1)) = 1
Next
Next
End Sub

```

```

Sub forcevecmod(nnode, nrest, nload, loads(), f(), fmod(), restraints(), KG())
For i = 1 To nnode * 3
f(i) = 0
fmod(i) = 0
Next

```

```

For i = 1 To nnode * 3
fmod(i) = loads(i, 2)
Next
For i = 1 To nnode * 3
For p = 1 To nrest
fmod(i) = fmod(i) - KG(i, restraints(p, 1)) * restraints(p, 2)
Next
Next
For i = 1 To nrest
fmod(restraints(i, 1)) = restraints(i, 2)
Next
End Sub

```

```

Sub matsol(a(), cc(), u(), nnode)
For j = 1 To nnode * 3 - 1
For i = j + 1 To nnode * 3
pp = a(i, j) / a(j, j)
cc(i) = cc(i) - cc(j) * pp
For ind = j To nnode * 3
a(i, ind) = a(i, ind) - a(j, ind) * pp
Next
Next
Next
u(nnode * 3) = cc(nnode * 3) / a(nnode * 3, nnode * 3)
For j = nnode * 3 - 1 To 1 Step -1
s = 0
For i = j + 1 To nnode * 3
s = s + a(j, i) * u(i)
Next
u(j) = (cc(j) - s) / a(j, j)
Next
End Sub

```

```

Sub postprocess(top(), Kel(), u(), uu(), ue(), pp(), intforce(), ne, nnode)
For ie = 1 To ne
For i = 1 To 6
pp(ie, i) = 0
Next
Next
For i = 1 To nnode
uu(i, 1) = u(3 * i - 2): uu(i, 2) = u(3 * i - 1): uu(i, 3) = u(3 * i)
Next
For ie = 1 To ne
p = top(ie, 1): r = top(ie, 2)
For i = 1 To 3
ue(ie, i) = uu(p, i): ue(ie, i + 3) = uu(r, i)
Next
For i = 1 To 6
For j = 1 To 6
pp(ie, i) = pp(ie, i) + Kel(ie, i, j) * ue(ie, j)
Next
Next
For i = 1 To 6
pp(ie, i) = pp(ie, i) - intforce(ie, i)
Next
Next
End Sub

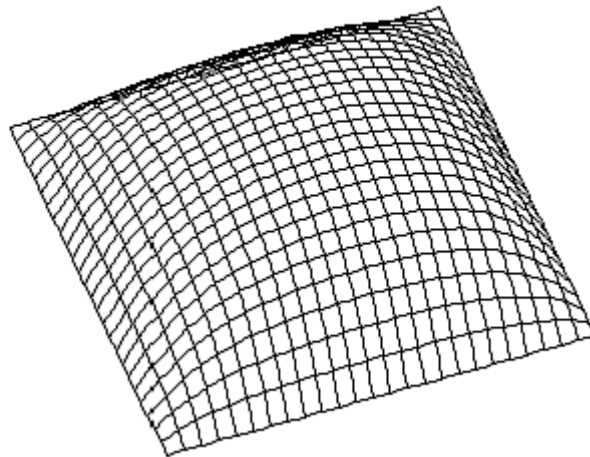
```

第3章 FEM初歩 ポアソンの方程式

枠にシャボン膜をはりつけやさしく圧力をかけた場合を想像しよう。これはポアソンの方程式

$$-\left(\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2}\right) = f$$

によって表現することができる。シャボン膜がつくる表面の形状を数値計算を用いて近似的に求めて見よう。なんの得があるのかだつて？

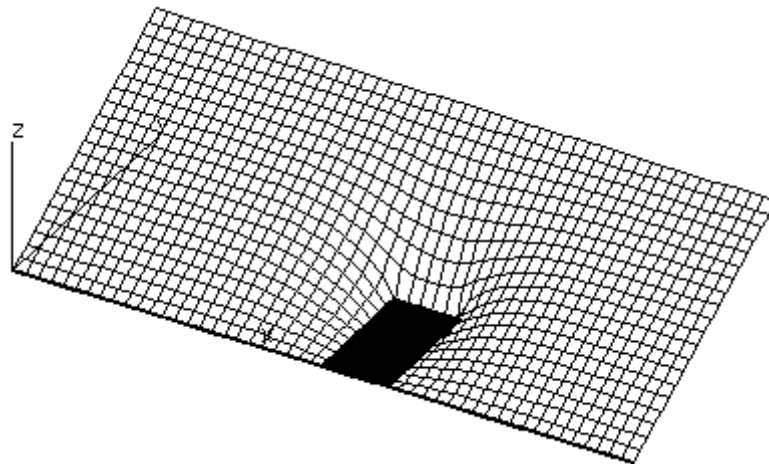


機械屋には材料力学のねじりの問題、電気屋には誘電体中の静電場解析を行うにはこれは極めて重要な課題であるからである。

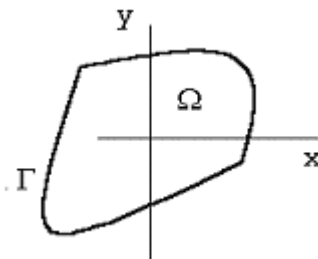
またこのポアソン方程式の特殊なケースとして、右辺を0とおいた

$$\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = 0$$

はラプラスの方程式と呼ばれ、流体力学、土木工学、電磁気学、伝熱学、気象学に極めて重要な役割を果たす関数となるからである。



では、数値計算の課題はなにか？ それは、あたえられた枠でシャボン膜の形状をもとめよという問題に帰せられる。



薄膜アナロジーによって、 Ω は次の微分方程式と境界条件を満たす。なお境界内の領域、 Ω は断面の境界を表す。 Γ は Ω という周辺領域からシャボン膜のように、あるいはドームのように、膨れ上がる膜のその高さをあらわすととらえよう。

$$-\left(\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2}\right) = f \quad \text{in } \Omega$$

$$\psi = 0 \quad \text{on } \Gamma$$

FEM では領域を小さな区画に分けて、その区画の3隅あるいは4隅の角の部分の座標に相応する ψ の高さを近似的に求めようとするのである。ひとたび網目状の x - y 座標の区画の点にたいする ψ の大きさが求めれば、区画内の点にたいしては区画の4隅または3隅の ψ の値から近似式（一次または2次）を使っ

て の値を求めることができる。

結論をいえば u を u であらわすと、上の問題はトラスで解いてきたのと同様、次のような方法で近似的に解くことができる。

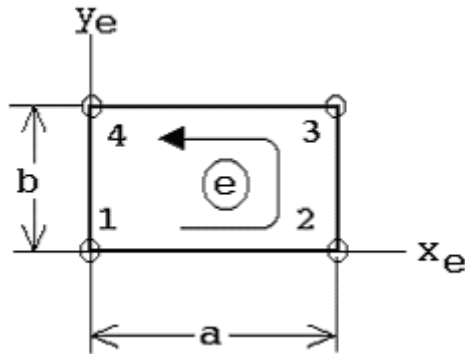
まず、区画(要素)をおおう薄膜の剛性に相当する要素剛性マトリクスを $[K_e]$ 、圧力による膨らまし力のベクトルを $[f_e]$ 、外力のベクトルを $[Q_e]$ とすると、区画の隅の各点の高さ(ポテンシャル)を並べたベクトル $[u_e]$ との間には次の剛性方程式が成り立つ。

$$[K_e] \cdot [u_e] = [f_e] + [Q_e]$$

要素に対する $[K_e]$ 、 $[f_e]$ 、 $[Q_e]$ を求めた後は、全体の剛性方程式を組立て、境界条件によって、全体の剛性マトリクスと全体の力ベクトルを修正したのち、一気に u を求める。これは今までやってきた手順とまったく同じである

長方形一次要素の剛性方程式

要素を x_e 、 y_e 方向に平行な辺をもつ長方形とし、その要素の4隅を4個の節点のローカル番号を左下隅からスタートして、反時計回りに1, 2, 3, 4としよう。4個の節点における u の値に節点のローカル番号を付けて u_1, u_2, u_3, u_4 としよう。



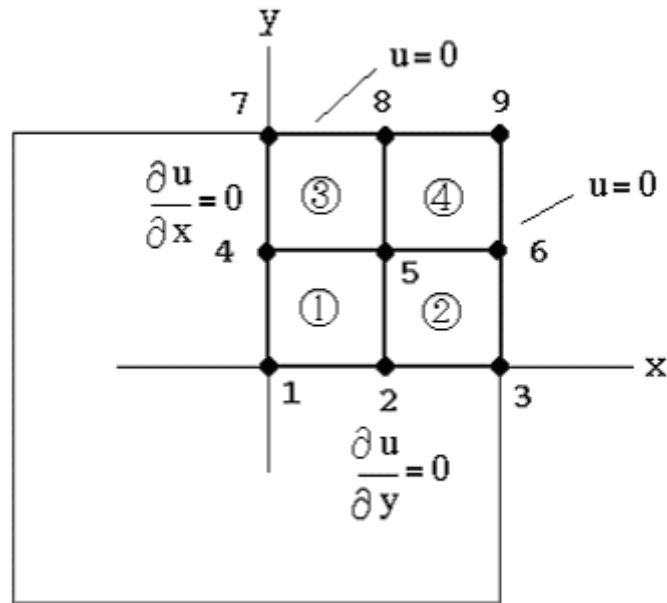
このとき要素の剛性マトリクスは結論からいえば

$$K_{el_{ie}} := \frac{b_{ie}}{6 \cdot a_{ie}} \begin{bmatrix} 2 & -2 & -1 & 1 \\ -2 & 2 & 1 & -1 \\ -1 & 1 & 2 & -2 \\ 1 & -1 & -2 & 2 \end{bmatrix} + \frac{a_{ie}}{6 \cdot b_{ie}} \begin{bmatrix} 2 & 1 & -1 & -2 \\ 1 & 2 & -2 & -1 \\ -1 & -2 & 2 & 1 \\ -2 & -1 & 1 & 2 \end{bmatrix}$$

で与えられる。また内力(すなわち薄膜にたいする内部圧力 ff による力)にたいするベクトルは結論からいえば、次のように与えられる。

$$f_{e_{ie}} := \frac{ff}{4} \cdot a_{ie} \cdot b_{ie} \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

では早速例題に入ろう。



図は正方形の境界を有する場合の例で、性質の対称性から第1象元の1/4の部分を検討しよう。節点3の座標は(1,0)、節点7の座標は(0,1)とする。まず、データの入力をはじめよう。

$$\begin{bmatrix} n_{el} \\ n_{nodes} \\ n_{res} \end{bmatrix} := \begin{bmatrix} 4 \\ 9 \\ 5 \end{bmatrix} \quad \text{Top} := \begin{bmatrix} 1 & 2 & 5 & 4 \\ 2 & 3 & 6 & 5 \\ 4 & 5 & 8 & 7 \\ 5 & 6 & 9 & 8 \end{bmatrix}$$

要素の数、節点 (node) の数、拘束点 (restraint) の数を与える。

Topology table で1行目は、要素(1)の左下から反時計回りに、順に4隅の節点番号が1, 2, 5, 4であることを示す。2行目、3行目、4行目は同じく要素(2)、(3)、(4)の左下から反時計回りに順に4隅の節点番号を示す。4つの要素はすべて同じ寸法の長方形なので、要素の剛性マトリクスは4個とも次のようになる。

$$a_{ie} := 0.5 \quad b_{ie} := 0.5 \quad \text{Kel}_{ie} := \frac{b_{ie}}{6 \cdot a_{ie}} \begin{bmatrix} 2 & -2 & -1 & 1 \\ -2 & 2 & 1 & -1 \\ -1 & 1 & 2 & -2 \\ 1 & -1 & -2 & 2 \end{bmatrix} + \frac{a_{ie}}{6 \cdot b_{ie}} \begin{bmatrix} 2 & 1 & -1 & -2 \\ 1 & 2 & -2 & -1 \\ -1 & -2 & 2 & 1 \\ -2 & -1 & 1 & 2 \end{bmatrix}$$

要素 (1) の要素剛性マトリクスの実際の数値は次のようになるが、他の 3 個のマトリクスもまったく同一である。

$$K_{el_1} = \begin{bmatrix} 0.667 & -0.167 & -0.333 & -0.167 \\ -0.167 & 0.667 & -0.167 & -0.333 \\ -0.333 & -0.167 & 0.667 & -0.167 \\ -0.167 & -0.333 & -0.167 & 0.667 \end{bmatrix}$$

さて、4 個の要素の剛性マトリクスの全体剛性マトリクスへの組立てに入ろう。幸いなことに、ねじりの問題では、節点ひとつに u の値がひとつのみ対応するので、全体が見通しやすい。要素 (1)、(2)、(3)、(4) のマトリクスの構成数値の全体マトリクスへの入り先を次の表に示す。列につけられた () の中の番号 u につけられたサブスクリプトが、全体マトリクスの中への落ち着き先の列と行の番地に一致する。同じ番地へ組み込まれる複数の値はすべて加え合わせられる。

$$\begin{array}{cc} \begin{array}{c} (1\ 2\ 5\ 4) \\ \begin{bmatrix} \text{xxx} & \text{xxx} & \text{xxx} & \text{xxx} \\ \text{xxx} & \text{xxx} & \text{xxx} & \text{xxx} \\ \text{xxx} & \text{xxx} & \text{xxx} & \text{xxx} \\ \text{xxx} & \text{xxx} & \text{xxx} & \text{xxx} \end{bmatrix} \end{array} & \begin{array}{c} \begin{bmatrix} u_1 \\ u_2 \\ u_5 \\ u_4 \end{bmatrix} \end{array} \\ \begin{array}{c} (2\ 3\ 6\ 5) \\ \begin{bmatrix} \text{xxx} & \text{xxx} & \text{xxx} & \text{xxx} \\ \text{xxx} & \text{xxx} & \text{xxx} & \text{xxx} \\ \text{xxx} & \text{xxx} & \text{xxx} & \text{xxx} \\ \text{xxx} & \text{xxx} & \text{xxx} & \text{xxx} \end{bmatrix} \end{array} & \begin{array}{c} \begin{bmatrix} u_2 \\ u_3 \\ u_6 \\ u_5 \end{bmatrix} \end{array} \\ \begin{array}{c} (4\ 5\ 8\ 7) \\ \begin{bmatrix} \text{xxx} & \text{xxx} & \text{xxx} & \text{xxx} \\ \text{xxx} & \text{xxx} & \text{xxx} & \text{xxx} \\ \text{xxx} & \text{xxx} & \text{xxx} & \text{xxx} \\ \text{xxx} & \text{xxx} & \text{xxx} & \text{xxx} \end{bmatrix} \end{array} & \begin{array}{c} \begin{bmatrix} u_4 \\ u_5 \\ u_8 \\ u_7 \end{bmatrix} \end{array} \\ \begin{array}{c} (5\ 6\ 9\ 8) \\ \begin{bmatrix} \text{xxx} & \text{xxx} & \text{xxx} & \text{xxx} \\ \text{xxx} & \text{xxx} & \text{xxx} & \text{xxx} \\ \text{xxx} & \text{xxx} & \text{xxx} & \text{xxx} \\ \text{xxx} & \text{xxx} & \text{xxx} & \text{xxx} \end{bmatrix} \end{array} & \begin{array}{c} \begin{bmatrix} u_5 \\ u_6 \\ u_9 \\ u_8 \end{bmatrix} \end{array} \end{array}$$

この作業は次の数式で行われる。

$$n := n_{\text{nodes}} \quad K_{n,n} := 0 \quad ie := 1..n_{\text{el}} \quad i := 1..4 \quad j := 1..4$$

$$K_{(Top_{ie,i}), (Top_{ie,j})} := K_{(Top_{ie,i}), Top_{ie,j}} + (K_{el_{ie}})_{i,j}$$

その結果、全体の剛性マトリクスはつぎのようになる。

$$K = \begin{bmatrix} 0.667 & -0.167 & 0 & -0.167 & -0.333 & 0 & 0 & 0 & 0 \\ -0.167 & 1.333 & -0.167 & -0.333 & -0.333 & -0.333 & 0 & 0 & 0 \\ 0 & -0.167 & 0.667 & 0 & -0.333 & -0.167 & 0 & 0 & 0 \\ -0.167 & -0.333 & 0 & 1.333 & -0.333 & 0 & -0.167 & -0.333 & 0 \\ -0.333 & -0.333 & -0.333 & -0.333 & 2.667 & -0.333 & -0.333 & -0.333 & -0.333 \\ 0 & -0.333 & -0.167 & 0 & -0.333 & 1.333 & 0 & -0.333 & -0.167 \\ 0 & 0 & 0 & -0.167 & -0.333 & 0 & 0.667 & -0.167 & 0 \\ 0 & 0 & 0 & -0.333 & -0.333 & -0.333 & -0.167 & 1.333 & -0.167 \\ 0 & 0 & 0 & 0 & -0.333 & -0.167 & 0 & -0.167 & 0.667 \end{bmatrix}$$

節点 3 9、節点 9 7 においては $u = 0$ という拘束条件が次のように与えられる。

$$\text{Restrains} := \begin{bmatrix} 3 & 0 \\ 6 & 0 \\ 9 & 0 \\ 8 & 0 \\ 7 & 0 \end{bmatrix}$$

内力による、ベクトルが下のように与えられる。

$$ff := 2 \quad fe_{ie} := \frac{ff}{4} \cdot a_{ie} \cdot b_{ie} \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

外力に相当するベクトル $\{Q_e\}$ は節点 3 9、節点 9 7 線での外力 Q に関しては不明(?) だが、拘束条件によって、これらに対応する節点の右辺での力ベクトル相当成分は、連立方程式修正時に強制的に u の解 0 にされる。また、 x 軸に沿う境界の部分では $\frac{\partial u}{\partial y} = 0$ で、 y 軸に沿う境界の部分では $\frac{\partial u}{\partial x} = 0$ の境界条件があり、これは理由は後まわしにすると、これらの辺にある節点の $\{Q_e\}$ が 0 とすることができる。また、境界内の節点 5 では周囲の要素に相殺されて $\{Q_e\}$ が 0 となる。結局、ここでは、 $\{Q_e\}$ の項目は無視して u を求める方程式に進むことができる。

これらにしたがって、全体の剛性マトリクスと内力を境界条件で修正する段階にはいった。 まず、Kmod はおなじみの

$$\text{indp} := 1..n_{\text{nodes}} \quad \text{indj} := 1..n_{\text{nodes}} \quad \text{id} := 1..n_{\text{res}}$$

$$\text{Kmod} := \text{K}$$

$$\text{Kmod}_{\text{Restrains}_{\text{id},1}, \text{indj}} := 0$$

$$\text{Kmod}_{\text{indj}, \text{Restrains}_{\text{id},1}} := 0$$

$$\text{Kmod}_{(\text{Restrains}_{\text{id},1}), \text{Restrains}_{\text{id},1}} := 1$$

fmodは

$$f_{\text{indp}} := 0$$

$$f_{(\text{Top}_{\text{ie},p})} := f_{(\text{Top}_{\text{ie},p})} + (f_{\text{ie}})_p$$

$$\text{fmod} := f$$

$$\text{fmod}_{\text{indp}} := \text{fmod}_{\text{indp}} - \text{K}_{\text{indp}, \text{Restrains}_{\text{id},1}} \cdot \text{Restrains}_{\text{id},2}$$

$$\text{fmod}_{(\text{Restrains}_{\text{id},1})} := \text{Restrains}_{\text{id},2}$$

これで一気に u を求めることができる。

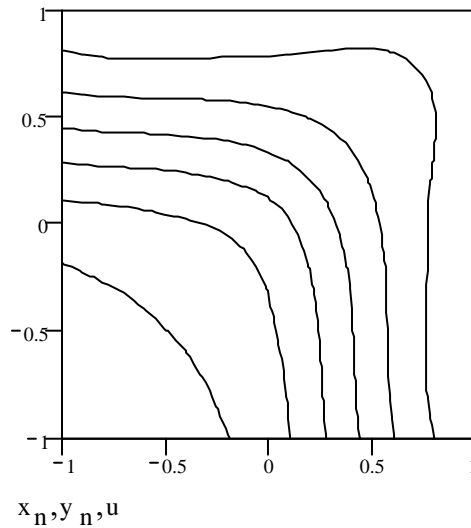
$$\text{fmod} = \begin{bmatrix} 0.125 \\ 0.25 \\ 0 \\ 0.25 \\ 0.5 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad u := \text{Kmod}^{-1} \cdot \text{fmod} \quad u = \begin{bmatrix} 0.621 \\ 0.482 \\ 0 \\ 0.482 \\ 0.386 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

等高線をソフトの力で描いて見よう。 まず、節点の座標の情報。

$$\text{Nodes} := \begin{bmatrix} 1 & 0 & 0 \\ 2 & 0.5 & 0 \\ 3 & 1 & 0 \\ 4 & 0 & 0.5 \\ 5 & 0.5 & 0.5 \\ 6 & 1 & 0.5 \\ 7 & 0 & 1 \\ 8 & 0.5 & 1 \\ 9 & 1 & 1 \end{bmatrix}$$

$x_n := \text{Nodes} \langle 1 \rangle$ $y_n := \text{Nodes} \langle 2 \rangle$
 $i := 1..n \text{ nodes}$

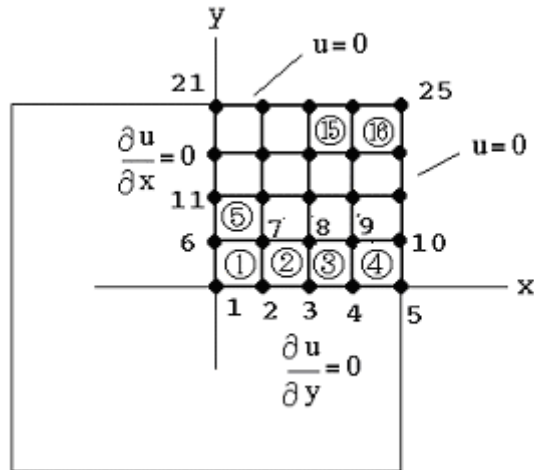
そして等高線



シャボン膜の膨らみの1/4の部分の等高線。メッシュがあらいで、精度的に若干の問題はあるとしても、よくできているではないか。等高線のコミあっているところが、せん断応力の大きいところである。まずはこれでもって、ポアソン方程式の有限要素法のもっとも簡単な導入例としよう。

長方形一次要素メッシュを増やしたモデル

前の節で最も簡単な有限要素法の数式ができあがったので、精度を上げるため、前の例の要素の数を 16 に、節点の数を 25 と大きくとって検討して見よう。入力のデータを変更するだけで、計算は数式が自動的にやってくれる。



図は正方形の境界を有する場合の前の例とおなじで、第 1 象限 1/4 の部分のメッシュ数を増やただけである。 節点 5 の座標は (1,0)、節点 21 の座標は (0,1) である。

まず、要素の数、節点の数、拘束点の数、トポロジーテーブル、拘束条件のデータを入力しよう。

$$\begin{bmatrix} n_{el} \\ n_{nodes} \\ n_{res} \end{bmatrix} := \begin{bmatrix} 16 \\ 25 \\ 9 \end{bmatrix}$$

$$ie := 1..16$$

$$a_{ie} := 0.25 \quad b_{ie} := 0.25 \quad ff := 2$$

$$\text{Top} = \begin{bmatrix} 1 & 2 & 7 & 6 \\ 2 & 3 & 8 & 7 \\ 3 & 4 & 9 & 8 \\ 4 & 5 & 10 & 9 \\ 6 & 7 & 12 & 11 \\ 7 & 8 & 13 & 12 \\ 8 & 9 & 14 & 13 \\ 9 & 10 & 15 & 14 \\ 11 & 12 & 17 & 16 \\ 12 & 13 & 18 & 17 \\ 13 & 14 & 19 & 18 \\ 14 & 15 & 20 & 19 \\ 16 & 17 & 22 & 21 \\ 17 & 18 & 23 & 22 \\ 18 & 19 & 24 & 23 \\ 19 & 20 & 25 & 24 \end{bmatrix} \quad \text{Restrains} := \begin{bmatrix} 5 & 0 \\ 10 & 0 \\ 15 & 0 \\ 20 & 0 \\ 25 & 0 \\ 24 & 0 \\ 23 & 0 \\ 22 & 0 \\ 21 & 0 \end{bmatrix}$$

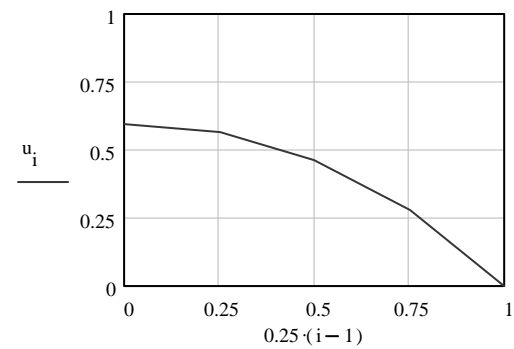
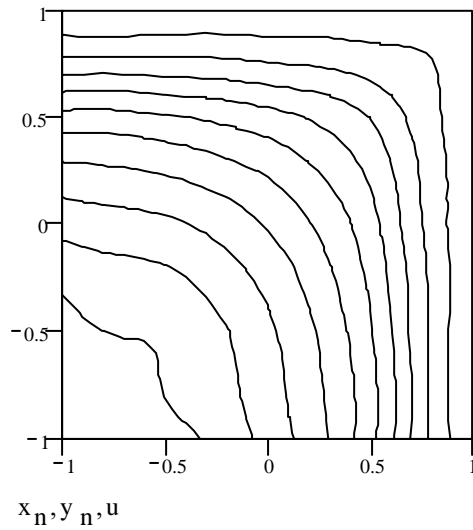
これらのデータによって要素剛性マトリクスが作られ、全体剛性マトリクスの組立て、境界条件による全体剛性マトリクスと内力ベクトルの修正が行われ、一気に u が求められる。（次頁）

また、応力関数の等高線も求められる。（次頁）

等高線図を見ると、 x 軸と y 軸の方向に等高線が混み合っている。従って、せん断応力は等高線の接線の方向とおなじ方向なので、 x 軸の線上で垂直方向のものと、 y 軸の線上で水平方向らしいと観察できる。

シャボンだまを想像してみても感覚的にもゆるせる。これは長方形断面なので、対称性があるから、 x 軸にそって u の変化をグラフにして見よう。（次頁）

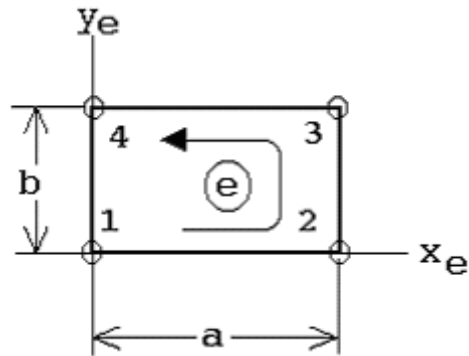
fmod =	0.031	u =	0.597
	0.063		0.565
	0.063		0.464
	0.063		0.283
	0		0
	0.063		0.565
	0.125		0.535
	0.125		0.441
	0.125		0.27
	0		0
	0.063		0.464
	0.125		0.441
	0.125		0.368
	0.125		0.229
	0		0
	0.063		0.283
	0.125		0.27
	0.125		0.229
	0.125		0.15
	0		0
	0		0
	0		0
	0		0
	0		0
	0		0
	0		0



もしこの例がねじりの問題であるなら、せん断応力は図のグラフの勾配に比例する。x 軸にそう (1,0) のところの勾配が最も大きいので、この点で最大のせん断応力となる。

形状関数

もう一度長方形一次要素へ戻って見よう。



4 隅の節点でローカル番号 1 , 2 , 3 , 4 に対して、これまでの計算で関数の値が u という値で求められた。たとえば、要素 (4) にたいしては、全体から見た節点の番号は 4 , 5 , 10 , 9 なので

$$\psi_1 = u_4, \quad \psi_2 = u_5, \quad \psi_3 = u_{10}, \quad \psi_4 = u_9$$

となり、4 隅の関数の数値が与えられた。

長方形一次要素の内部における任意の点の座標にたいする、 ψ の値はローカルの $x_e - y_e$ 座標の原点からの位置を (x, y) であらわすと、下の近似式であたえられる。

$$\psi = N_1 \cdot \psi_1 + N_2 \cdot \psi_2 + N_3 \cdot \psi_3 + N_4 \cdot \psi_4$$

ただし、

$$N_1(x, y) := \left(1 - \frac{x}{a}\right) \cdot \left(1 - \frac{y}{b}\right) \quad N_2(x, y) := \frac{x}{a} \cdot \left(1 - \frac{y}{b}\right)$$

$$N_3(x, y) := \frac{x \cdot y}{a \cdot b} \quad N_4(x, y) := \left(1 - \frac{x}{a}\right) \cdot \frac{y}{b}$$

の近似に使う上の関数 N_1, N_2, N_3, N_4 を形状関数と名づける。

Excel VBA による長方形一次有限要素モデル

まず、入力と出力のためのシートを作ろう。 節点数、要素数、拘束点数の入力の右に、楕円形の計算ボタンを配し、節点番号にたいする、x、y座標、要素の4隅にある節点の番号をしるすトポロジーテーブル、拘束番号(拘束値のある節点番号)と拘束値の入力場所を配置する。 出力は節点番号とuの値、検算のためのfmodを配置する。 読者も自分に便利な独自のシートを作りたい。

1	ポアソン微分方程式の有限方程式による解法 長方形一次要素モデル												
2	節点数=	9	要素数=	4	ff=	2						2000/1/26	上出
3	拘束点数	5											
4													
5	節点番号	x	y	Topology				拘束番号	拘束値	節点番号	uの値	fmod	
6	1	0	0	1	1	2	5	4	3	0	1	0.62142857	0.125
7	2	0.5	0	2	2	3	6	5	6	0	2	0.48214286	0.25
8	3	1	0	3	4	5	8	7	9	0	3	0	0
9	4	0	0.5	4	5	6	9	8	8	0	4	0.48214286	0.25
10	5	0.5	0.5	5					7	0	5	0.38571429	0.5
11	6	1	0.5	6							6	0	0
12	7	0	1	7							7	0	0
13	8	0.5	1	8							8	0	0
14	9	1	1	9							9	0	0
15	10			10							10		
16	11			11							11		
17	12			12							12		
18	13			13							13		
19	14			14							14		
20	15			15							15		
21													

Sub Macro1()

'ポアソン方程式 FEM 長方形一次要素モデル

```
Dim nnode, ne, nrest, i, j, ind, ie, p, ir
Dim x(100), y(100), top(100, 5), nodes(400, 3), restraints(100, 2), k1, k2
Dim x1, x2, x3, x4, y1, y2, y3, y4, aa, bb, Ae, KS(4, 4), fe(100, 4), f(400)
Dim K(100, 4, 4), Kel(100, 4, 4), KG(400, 400), indp, indj
Dim Kmod(400, 400), fmod(400), pp, a(400, 400), c(400), u(400), s
```

'インプット (1) -----

```
nnode = Cells(2, 2): ne = Cells(2, 4): nrest = Cells(3, 2): ff = Cells(2, 7)
```

'インプット (2) -----

```
For ind = 1 To nnode
x(ind) = Cells(ind + 5, 2): y(ind) = Cells(ind + 5, 3)
Next
For p = 1 To 4
```

```

For ie = 1 To ne
top(ie, p) = Cells(ie + 5, p + 4)
Next
Next
For j = 1 To 2
For i = 1 To nrest
restraints(i, j) = Cells(i + 5, j + 8)
Next
Next

```

'要素剛性マトリクスと内力ベクトル作成-----

```

For i = 1 To nnode
f(i) = 0
Next
For ie = 1 To ne
x1 = x(top(ie, 1)): x2 = x(top(ie, 2)): x3 = x(top(ie, 3)): x4 = x(top(ie, 4))
y1 = y(top(ie, 1)): y2 = y(top(ie, 2)): y3 = y(top(ie, 3)): y4 = y(top(ie, 4))

stiffnessmatrix KS(), x1, x2, x3, x4, y1, y2, y3, y4, ff, fe(), ie

```

```

For p = 1 To 4
For j = 1 To 4
Kel(ie, p, j) = KS(p, j)
Next
Next
For p = 1 To 4
f(top(ie, p)) = f(top(ie, p)) + fe(ie, p)
Next
Next

```

'全体剛性マトリクスの組立てサブルーチン呼び出し-----

```

globmat nnode, top(), Kel(), KG()

```

'境界条件を加味した剛性マトリクスと内力ベクトルの修正サブルーチン呼び出し-----

```

modification nnode, nrest, KG(), f(), fmod(), Kmod(), restraints()

```

'連立一次方程式解法サブルーチンの呼び出し-----

```

For j = 1 To nnode
c(j) = fmod(j)
For i = 1 To nnode
a(i, j) = Kmod(i, j)
Next
Next
matsol nnode, a(), c(), u()

```

'アウトプット-----

```

For i = 1 To nnode

Cells(i + 5, 14) = u(i)
Cells(i + 5, 15) = fmod(i)
Next
End Sub

```

' (3) サブルーチン

```

Sub stiffnessmatrix(KS(), x1, x2, x3, x4, y1, y2, y3, y4, ff, fe(), ie)
aa = Abs(x1 - x2): bb = Abs(y4 - y1)

```

$k1 = bb / 6 / aa$; $k2 = aa / 6 / bb$

$KS(1, 1) = 2 * k1 + 2 * k2$; $KS(1, 2) = -2 * k1 + 1 * k2$
 $KS(1, 3) = -1 * k1 - 1 * k2$; $KS(1, 4) = 1 * k1 - 2 * k2$
 $KS(2, 1) = -2 * k1 + 1 * k2$; $KS(2, 2) = 2 * k1 + 2 * k2$
 $KS(2, 3) = 1 * k1 - 2 * k2$; $KS(2, 4) = -1 * k1 - 1 * k2$

$KS(3, 1) = -1 * k1 - 1 * k2$; $KS(3, 2) = 1 * k1 - 2 * k2$
 $KS(3, 3) = 2 * k1 + 2 * k2$; $KS(3, 4) = -2 * k1 + 1 * k2$
 $KS(4, 1) = 1 * k1 - 2 * k2$; $KS(4, 2) = -1 * k1 - 1 * k2$
 $KS(4, 3) = -2 * k1 + 1 * k2$; $KS(4, 4) = 2 * k1 + 2 * k2$

```
For p = 1 To 4
fe(ie, p) = ff / 4 * aa * bb
Next
End Sub
```

```
Sub globmat(nnode, top(), Kel(), KG())
For i = 1 To nnode
For j = 1 To nnode
KG(i, j) = 0
Next
Next
For ie = 1 To 4
For p = 1 To 4
For j = 1 To 4
indp = top(ie, p)
indj = top(ie, j)
```

```
KG(indp, indj) = KG(indp, indj) + Kel(ie, p, j)
Next
Next
Next
End Sub
```

```
Sub modification(nnode, nrest, KG(), f(), fmod(), Kmod(), restraints())
For i = 1 To nnode
fmod(i) = f(i)
For j = 1 To nnode
Kmod(i, j) = KG(i, j)
Next
Next
```

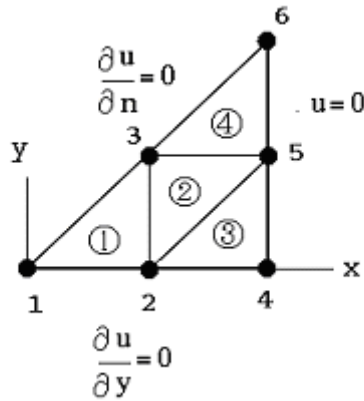
```
For indp = 1 To nnode
For indj = 1 To nnode
For ir = 1 To nrest
If indp = restraints(ir, 1) Then
Kmod(indp, indj) = 0
Kmod(indj, indp) = 0
Kmod(indp, indp) = 1
End If
Next
Next
Next
For i = 1 To nnode: '-----modification of the force vector
For j = 1 To nnode
fmod(j) = fmod(j) - KG(j, restraints(i, 1)) * f(restraints(i, 2))
```

```
Next
Next
For ir = 1 To nrest
For indp = 1 To nnode
If indp = restraints(ir, 1) Then
fmod(indp) = restraints(ir, 2)
End If
Next
Next
End Sub
```

```
Sub matsol(nnode, a(), c(), u())
For j = 1 To nnode
For i = j + 1 To nnode
pp = a(i, j) / a(j, j)
c(i) = c(i) - c(j) * pp

For ind = j To nnode - 1
a(i, ind) = a(i, ind) - a(j, ind) * pp
Next
Next
Next
u(nnode) = c(nnode) / a(nnode, nnode)
For j = nnode - 1 To 1 Step -1
s = 0
For i = j + 1 To nnode - 1
s = s + a(j, i) * u(i)
Next
u(j) = (c(j) - s) / a(j, j)
Next
End Sub
```

三角形一次要素によるモデル

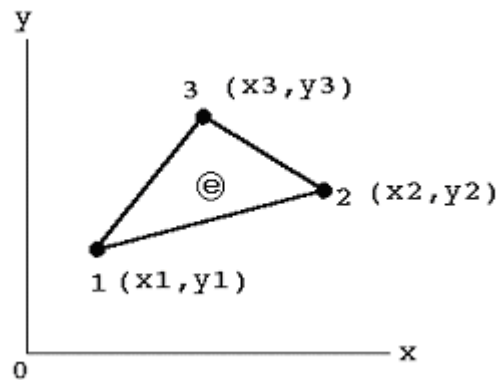


図は前の例とおなじ問題を三角形一次要素で検討する例である。正方形の境界においてはx軸とy軸にたいして対称であり、なおかつ直線 $y = x$ に対しても対象なので、三角形の要素を使うと、前例のさらに半分の $1/8$ の部分に境界を分割することができる。

とりあえず、Nodes なるマトリクスを導入し、節点番号、x座標、y座標の値を入力し、Topology table で要素の3隅の節点番号を指定することにしよう。なお、Topology table で要素の3隅の節点の番号はどの要素でも反時計回りに数えていることに注意されたい。

$$\begin{bmatrix} n_{el} \\ n_{nodes} \\ n_{res} \end{bmatrix} := \begin{bmatrix} 4 \\ 6 \\ 3 \end{bmatrix} \quad \text{Nodes} := \begin{bmatrix} 1 & 0 & 0 \\ 2 & 0.5 & 0 \\ 3 & 0.5 & 0.5 \\ 4 & 1 & 0 \\ 5 & 1 & 0.5 \\ 6 & 1 & 1 \end{bmatrix} \quad \text{Top} := \begin{bmatrix} 1 & 2 & 3 \\ 5 & 3 & 2 \\ 2 & 4 & 5 \\ 3 & 5 & 6 \end{bmatrix}$$

さて、要素剛性マトリクスをつくるため、任意の3角一次要素を考え、その3つの頂点を節点とし反時計周りにローカル番号を1、2、3とつけて、その座標をそれぞれ (x_1, y_1) 、 (x_2, y_2) 、 (x_3, y_3) とする。(次頁の図参照)



各要素にたいして、ローカル節点番号 1 , 2 , 3 の座標は先に与えられたデータから、次のようにまとめられる。

$$x_n := \text{Nodes} \langle n \rangle \quad y_n := \text{Nodes} \langle n \rangle \quad ie := 1..n_{el}$$

$$\begin{bmatrix} x_{1_{ie}} & y_{1_{ie}} \\ x_{2_{ie}} & y_{2_{ie}} \\ x_{3_{ie}} & y_{3_{ie}} \end{bmatrix} := \begin{bmatrix} x_n(\text{Top}_{ie,1}) & y_n(\text{Top}_{ie,1}) \\ x_n(\text{Top}_{ie,2}) & y_n(\text{Top}_{ie,2}) \\ x_n(\text{Top}_{ie,3}) & y_n(\text{Top}_{ie,3}) \end{bmatrix}$$

各要素にたいして、次の値 a、b、c を計算しよう。

$$\begin{bmatrix} a_{1_{ie}} & b_{1_{ie}} & c_{1_{ie}} \\ a_{2_{ie}} & b_{2_{ie}} & c_{2_{ie}} \\ a_{3_{ie}} & b_{3_{ie}} & c_{3_{ie}} \end{bmatrix} := \begin{bmatrix} x_{2_{ie}} \cdot y_{3_{ie}} - x_{3_{ie}} \cdot y_{2_{ie}} & y_{2_{ie}} - y_{3_{ie}} & x_{3_{ie}} - x_{2_{ie}} \\ x_{3_{ie}} \cdot y_{1_{ie}} - x_{1_{ie}} \cdot y_{3_{ie}} & y_{3_{ie}} - y_{1_{ie}} & x_{1_{ie}} - x_{3_{ie}} \\ x_{1_{ie}} \cdot y_{2_{ie}} - x_{2_{ie}} \cdot y_{1_{ie}} & y_{1_{ie}} - y_{2_{ie}} & x_{2_{ie}} - x_{1_{ie}} \end{bmatrix} \quad Ae_{ie} := \frac{a_{1_{ie}} + a_{2_{ie}} + a_{3_{ie}}}{2}$$

要素の剛性マトリクスは次のように与えられる。

$$K_{el_{ie}} := \begin{bmatrix} (b_{1_{ie}})^2 + (c_{1_{ie}})^2 & b_{1_{ie}} \cdot b_{2_{ie}} + c_{1_{ie}} \cdot c_{2_{ie}} & b_{1_{ie}} \cdot b_{3_{ie}} + c_{1_{ie}} \cdot c_{3_{ie}} \\ b_{1_{ie}} \cdot b_{2_{ie}} + c_{1_{ie}} \cdot c_{2_{ie}} & (b_{2_{ie}})^2 + (c_{2_{ie}})^2 & b_{3_{ie}} \cdot b_{2_{ie}} + c_{3_{ie}} \cdot c_{2_{ie}} \\ b_{1_{ie}} \cdot b_{3_{ie}} + c_{1_{ie}} \cdot c_{3_{ie}} & b_{3_{ie}} \cdot b_{2_{ie}} + c_{3_{ie}} \cdot c_{2_{ie}} & (b_{3_{ie}})^2 + (c_{3_{ie}})^2 \end{bmatrix} \cdot \frac{1}{4 \cdot Ae_{ie}}$$

全体の剛性マトリクスは四角一次要素の場合と同じような操作で、次のようになる。ただし、三角一次要素では節点の数が3つなっているので、i、j は1

から 3 までのレンジ変数となることに注意されたい。

$$n := n_{\text{nodes}} \quad K_{n,n} := 0 \quad ie := 1..n_{\text{el}} \quad i := 1..3 \quad j := 1..3$$

$$K_{(\text{Top}_{ie,i}), (\text{Top}_{ie,j})} := K_{(\text{Top}_{ie,i}), \text{Top}_{ie,j}} + (K_{el_{ie}})_{i,j}$$

数値的には

$$K = \begin{bmatrix} 0.5 & -0.5 & 0 & 0 & 0 & 0 \\ -0.5 & 2 & -1 & -0.5 & 0 & 0 \\ 0 & -1 & 2 & 0 & -1 & 0 \\ 0 & -0.5 & 0 & 1 & -0.5 & 0 \\ 0 & 0 & -1 & -0.5 & 2 & -0.5 \\ 0 & 0 & 0 & 0 & -0.5 & 0.5 \end{bmatrix}$$

拘束条件と内力のベクトルが次のようにあたえられる。

$$\text{Restrains} := \begin{bmatrix} 4 & 0 \\ 5 & 0 \\ 6 & 0 \end{bmatrix} \quad ff := 2 \quad fe_{ie} := \frac{1}{3} \cdot ff \cdot Ae_{ie} \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

拘束条件と内力ベクトルによる、全体剛性マトリクスの修正は四角のときとまったく同じ式によって行われる。

$$\text{indp} := 1..n_{\text{nodes}} \quad \text{indj} := 1..n_{\text{nodes}} \quad \text{id} := 1..n_{\text{res}}$$

$$K_{\text{mod}} := K$$

$$K_{\text{mod}}_{\text{Restrains}_{id,1}, \text{indj}} := 0$$

$$K_{\text{mod}}_{\text{indj}, \text{Restrains}_{id,1}} := 0$$

$$K_{\text{mod}}_{(\text{Restrains}_{id,1}), \text{Restrains}_{id,1}} := 1$$

数値的には、

$$K_{\text{mod}} = \begin{bmatrix} 0.5 & -0.5 & 0 & 0 & 0 & 0 \\ -0.5 & 2 & -1 & 0 & 0 & 0 \\ 0 & -1 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

荷重マトリクスの修正も四角のときとまったく同じ式によって行われる。

$$f_{\text{indp}} := 0$$

$$f_{(\text{Top}_{ie,p})} := f_{(\text{Top}_{ie,p})} + (f_{ie})_p$$

$$f_{\text{mod}} := f$$

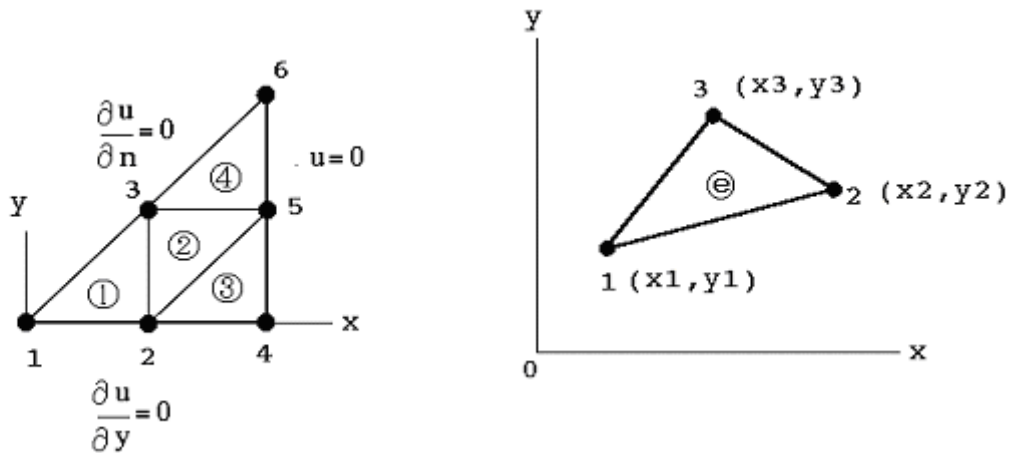
$$f_{\text{mod}_{\text{indp}}} := f_{\text{mod}_{\text{indp}}} - K_{\text{indp,Restrains}_{id,1}} \cdot \text{Restrains}_{id,2}$$

$$f_{\text{mod}}(\text{Restrains}_{id,1}) := \text{Restrains}_{id,2}$$

したがって、関数の値 u はこれも今までとまったく同じ計算で次のようにもとめられる。

$$f_{\text{mod}} = \begin{bmatrix} 0.083 \\ 0.25 \\ 0.25 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad u := K_{\text{mod}}^{-1} \cdot f_{\text{mod}} \quad u = \begin{bmatrix} 0.625 \\ 0.458 \\ 0.354 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

形状関数



両図を振り返って見よう。要素(3)を一例としてとると、3隅の節点でローカル番号1, 2, 3にたいして全体から見た節点の番号は2, 4, 5なので

$$1 = u_2, \quad 2 = u_4, \quad 3 = u_5$$

となり、3隅の関数の数値が与えられる。三角形一次要素の内部における任意の点の座標にたいする、 ψ の値は全体のx-y座標の原点からの位置を(x, y)であらわすと、下の近似式であたえられる。

$$\psi = N_1 \cdot \psi_1 + N_2 \cdot \psi_2 + N_3 \cdot \psi_3$$

ただし、a, b, c を次の式より求めて、

$$\begin{bmatrix} a_{1,ie} & b_{1,ie} & c_{1,ie} \\ a_{2,ie} & b_{2,ie} & c_{2,ie} \\ a_{3,ie} & b_{3,ie} & c_{3,ie} \end{bmatrix} := \begin{bmatrix} x_{2,ie} \cdot y_{3,ie} - x_{3,ie} \cdot y_{2,ie} & y_{2,ie} - y_{3,ie} & x_{3,ie} - x_{2,ie} \\ x_{3,ie} \cdot y_{1,ie} - x_{1,ie} \cdot y_{3,ie} & y_{3,ie} - y_{1,ie} & x_{1,ie} - x_{3,ie} \\ x_{1,ie} \cdot y_{2,ie} - x_{2,ie} \cdot y_{1,ie} & y_{1,ie} - y_{2,ie} & x_{2,ie} - x_{1,ie} \end{bmatrix} \quad Ae_{ie} := \frac{a_{1,ie} + a_{2,ie} + a_{3,ie}}{2}$$

ついで、N1, N2, N3 がもとめられる。

$$N_1(x,y) := \frac{1}{2 \cdot Ae_3} \cdot (a_{1_3} + b_{1_3} \cdot x + c_{1_3} \cdot y)$$

$$N_2(x,y) := \frac{1}{2 \cdot Ae_3} \cdot (a_{2_3} + b_{2_3} \cdot x + c_{2_3} \cdot y)$$

$$N_3(x,y) := \frac{1}{2 \cdot Ae_3} \cdot (a_{3_3} + b_{3_3} \cdot x + c_{3_3} \cdot y)$$

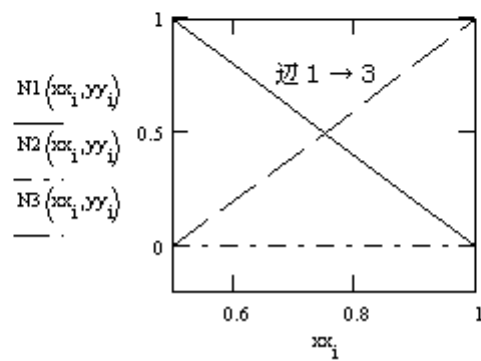
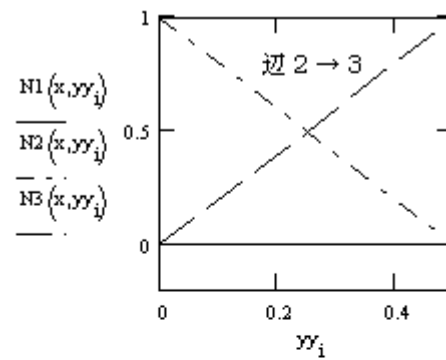
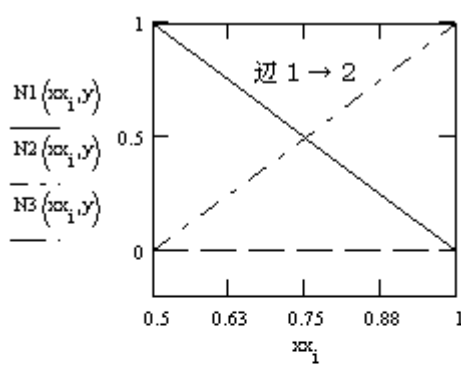
の近似に使う関数 N1, N2, N3 を形状関数と名づける。

要素(3)を一例として、3隅の節点のローカル番号1, 2, 3の座標にたいして形状関数をもとめてみると、つぎのように各節点ではその番号に対応する値は1で、対応しない番号の値は0である。

すなわち、ローカル番号1, 2, 3の節点に対する、応力関数 ψ の値は近似式によって 1, 2, 3に一致する。

$$\begin{aligned} \begin{bmatrix} x \\ y \end{bmatrix} &:= \begin{bmatrix} 0.5 \\ 0 \end{bmatrix} & \begin{bmatrix} N1(x,y) \\ N2(x,y) \\ N3(x,y) \end{bmatrix} &= \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} & \begin{bmatrix} x \\ y \end{bmatrix} &:= \begin{bmatrix} 1 \\ 0 \end{bmatrix} & \begin{bmatrix} N1(x,y) \\ N2(x,y) \\ N3(x,y) \end{bmatrix} &= \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \\ & & \begin{bmatrix} x \\ y \end{bmatrix} &:= \begin{bmatrix} 1 \\ 0.5 \end{bmatrix} & \begin{bmatrix} N1(x,y) \\ N2(x,y) \\ N3(x,y) \end{bmatrix} &= \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \end{aligned}$$

また、辺 1 → 2、辺 2 → 3、辺 1 → 3 に沿って各形状関数の値をそれぞれ、グラフにすると下のようになる。



FEM の原理 - 要素の剛性方程式

ここでは、FEM (有限要素法) がどのような原理を使っているのか考えよう。
次の微分方程式において e は断面のひとつの要素の境界内の領域、 e はその要素の境界としよう。

$$-\left(\frac{\partial^2 \Psi}{\partial x^2} + \frac{\partial^2 \Psi}{\partial y^2}\right) = f$$

上の微分方程式は次のような方法で近似的に解かれる。まず、の微分方程式の両辺に形状関数 N を乗じて e の領域にわたって面積分を行う。

$$\int_{\Omega_e} N \cdot \left[-\left(\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2}\right) \right] \cdot dA = \int_{\Omega_e} N \cdot f \cdot dA$$

これは部分積分の公式とグリーンの定理によって、次のように線積分の項の混ざった方程式に書き換えられる。ここで、 n を境界線の法線とする。

$$\int_{\Omega_e} \left(\frac{\partial N}{\partial x} \cdot \frac{\partial \psi}{\partial x} + \frac{\partial N}{\partial y} \cdot \frac{\partial \psi}{\partial y} \right) \cdot dA - \oint_{\Gamma_e} N \cdot \frac{\partial \psi}{\partial n} \cdot ds = \int_{\Omega_e} N \cdot f \cdot dA$$

かきなおして、

$$\int_{\Omega_e} \left(\frac{\partial N}{\partial x} \cdot \frac{\partial \psi}{\partial x} + \frac{\partial N}{\partial y} \cdot \frac{\partial \psi}{\partial y} \right) \cdot dA = fe + Qe$$

ただし、

$$fe = \int_{\Omega_e} N \cdot f \cdot dA$$

$$Qe = \oint_{\Gamma_e} N \cdot \frac{\partial \psi}{\partial n} \cdot ds$$

ここで、近似式 $\psi = N1 \cdot \psi_1 + N2 \cdot \psi_2 + N3 \cdot \psi_3$ を思いだし、また方程式内の N を $N1$ として、

$$\int_{\Omega_e} \left(\frac{\partial N}{\partial x} \cdot \frac{\partial \psi}{\partial x} + \frac{\partial N}{\partial y} \cdot \frac{\partial \psi}{\partial y} \right) \cdot dA = fe + Qe$$

を書き換えると、

$$\begin{aligned} & \left(\int_{\Omega_e} \left(\frac{\partial N_1}{\partial x} \cdot \frac{\partial N_1}{\partial x} + \frac{\partial N_1}{\partial y} \cdot \frac{\partial N_1}{\partial y} \right) \cdot dA \right) \cdot \psi_1 + \left(\int_{\Omega_e} \left(\frac{\partial N_1}{\partial x} \cdot \frac{\partial N_2}{\partial x} + \frac{\partial N_1}{\partial y} \cdot \frac{\partial N_2}{\partial y} \right) \cdot dA \right) \cdot \psi_2 \\ & + \left(\int_{\Omega_e} \left(\frac{\partial N_1}{\partial x} \cdot \frac{\partial N_3}{\partial x} + \frac{\partial N_1}{\partial y} \cdot \frac{\partial N_3}{\partial y} \right) \cdot dA \right) \cdot \psi_3 = \int_{\Omega_e} N_1 \cdot f \cdot dA + \oint_{\Gamma_e} N_1 \cdot \frac{\partial \psi}{\partial n} ds \end{aligned}$$

同じように、N を N2 とすると、

$$\begin{aligned} & \left(\int_{\Omega_e} \left(\frac{\partial N_2}{\partial x} \cdot \frac{\partial N_1}{\partial x} + \frac{\partial N_2}{\partial y} \cdot \frac{\partial N_1}{\partial y} \right) \cdot dA \right) \cdot \psi_1 + \left(\int_{\Omega_e} \left(\frac{\partial N_2}{\partial x} \cdot \frac{\partial N_2}{\partial x} + \frac{\partial N_2}{\partial y} \cdot \frac{\partial N_2}{\partial y} \right) \cdot dA \right) \cdot \psi_2 \\ & + \left(\int_{\Omega_e} \left(\frac{\partial N_2}{\partial x} \cdot \frac{\partial N_3}{\partial x} + \frac{\partial N_2}{\partial y} \cdot \frac{\partial N_3}{\partial y} \right) \cdot dA \right) \cdot \psi_3 = \int_{\Omega_e} N_2 \cdot f \cdot dA + \oint_{\Gamma_e} N_2 \cdot \frac{\partial \psi}{\partial n} ds \end{aligned}$$

同じように、N を N3 とすると、

$$\begin{aligned} & \left(\int_{\Omega_e} \left(\frac{\partial N_3}{\partial x} \cdot \frac{\partial N_1}{\partial x} + \frac{\partial N_3}{\partial y} \cdot \frac{\partial N_1}{\partial y} \right) \cdot dA \right) \cdot \psi_1 + \left(\int_{\Omega_e} \left(\frac{\partial N_3}{\partial x} \cdot \frac{\partial N_2}{\partial x} + \frac{\partial N_3}{\partial y} \cdot \frac{\partial N_2}{\partial y} \right) \cdot dA \right) \cdot \psi_2 \\ & + \left(\int_{\Omega_e} \left(\frac{\partial N_3}{\partial x} \cdot \frac{\partial N_3}{\partial x} + \frac{\partial N_3}{\partial y} \cdot \frac{\partial N_3}{\partial y} \right) \cdot dA \right) \cdot \psi_3 = \int_{\Omega_e} N_3 \cdot f \cdot dA + \oint_{\Gamma_e} N_3 \cdot \frac{\partial \psi}{\partial n} ds \end{aligned}$$

以上をまとめると、

$$Ke11 \cdot \psi_1 + Ke12 \cdot \psi_2 + Ke13 \cdot \psi_3 = Fe1 + Qe1$$

$$Ke21 \cdot \psi_1 + Ke22 \cdot \psi_2 + Ke23 \cdot \psi_3 = Fe2 + Qe2$$

$$Ke31 \cdot \psi_1 + Ke32 \cdot \psi_2 + Ke33 \cdot \psi_3 = Fe3 + Qe3$$

のように、 1、 2、 3 を未知数とする、要素の剛性方程式ができる。

要素の剛性マトリクス、内力ベクトル、外力ベクトル

今取り扱っている計算では、ひとつの節点にひとつの値（ポテンシャル）が対応するので、要素には節点の数だけの行と列の組み合わせの剛性マトリクスがつくられる。三角一次要素では節点を3個もつので、 $3 \times 3 = 9$ のマトリクスとなり、四角一次要素では $4 \times 4 = 16$ となる。したがって、 p 個の節点をもつ要素の剛性マトリクスの i 行、 j 列の成分は $i=1\dots p, j=1\dots p$ とすると

$$Ke_{ij} = \int_{\Omega_e} \left(\frac{\partial N_i}{\partial x} \cdot \frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y} \cdot \frac{\partial N_j}{\partial y} \right) \cdot dA$$

となる。また、右辺の第1項の内力ベクトル $\{ fe \}$ の i 行目の成分は、

$$fe_i = \int_{\Omega_e} N_i \cdot f \cdot dA$$

となり、今までに取り扱っている薄膜アナロジーにおいては f はシャボンだまを膨らますための、圧力に相当し、定数となるので、この成分の値は三角要素では $f/3$ に、四角要素では $f/4$ に要素の境界内の面積を乗じたものに等しい。

外力に相当するベクトル $\{ Qe \}$ の i 行目の成分は、

$$qe_i = \oint_{\Gamma_e} N_i \cdot \frac{\partial \psi}{\partial n} \cdot ds$$

となるが、今までの例では、計算が行われる領域（要素の集まったもの）の境界のある部分では、節点に拘束条件による値が割り当てられ、したがって右辺の境界にある節点の外力の値 qe_i は既知数としての（あるいは u ）によって置きかえられる（一応無視される）。また、境界ののこりの部分では $\frac{\partial \psi}{\partial n} = 0$ なので、右辺の qe_i は0となる。また境界内の節点では互いに節点と境界を共有する要素にわたっての線積分で、打ち消されて0となってしまう。すなわち巧妙に、この項がおもてに出ないように仕組みられているのである。

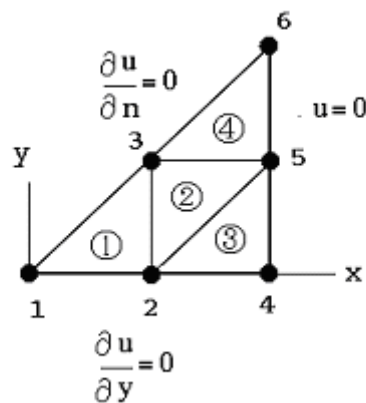
結論として、形状関数から求められた成分で作られる要素剛性マトリクスと、内力のベクトルを使うことによって、（あるいは u ）を求める連立方程式が形成されるのである。

EXCEL VBA による三角形一次要素モデル

長方形の柱のねじりの問題を近似的に解析するために、応力関数 (x, y) がある。応力関数 (x, y) は下のようなポアソン微分方程式を満足する。

$$-\left(\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2}\right) = 2 \quad \text{in}$$

$$\psi = 0 \quad \text{on}$$



正方形の境界を有する場合の例で、性質の対称性から $1/8$ の部分を検討しよう。節点 6 の座標は $(1, 1)$ とする。

長方形一次要素モデル、入力と出力のためのシートを利用する。要素の 3 隅にある節点の番号をしるすトポロジーテーブルの列数がひとつ減少する。

読者も自分に便利な独自のシートを作られたい。

ファイル(F) 編集(E) 表示(V) 挿入(I) 書式(O) ツール(T) データ(D) ウィンドウ(W) ヘルプ(H) Acrobat

G22 =

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	ポアソン微分方程式の有限方程式による解法 三角形一次要素モデル														
2	節点数 =	6	要素数 =	4	ff =	2								2000/1/26	上出
3	拘束点数	3													
4															
5	節点番号	x	y	Topology			拘束番号	拘束値	節点番号	u の値	fmod				
6	1	0	0	1	1	2	3	4	0	1	0.625	0.083			
7	2	0.5	0	2	5	3	2	5	0	2	0.45833333	0.25			
8	3	0.5	0.5	3	2	4	5	6	0	3	0.35416667	0.25			
9	4	1	0	4	3	5	6			4	0	0			
10	5	1	0.5	5						5	0	0			
11	6	1	1	6						6	0	0			
12	7			7						7					
13	8			8						8					
14	9			9						9					
15	10			10						10					
16	11			11						11					
17	12			12						12					
18	13			13						13					
19	14			14						14					
20	15			15						15					

長方形一次要素プログラムと本質的に同じであるが、要素剛性マトリクス（3行3列）と内力ベクトル（3行）が異なる。また三角形要素では頂点が3個なので、レンジ数が1 - 3となる。（長方形では1 - 4）。長方形モデル用プログラムのマイナーチェンジで作成することができる。

Sub Macro1()

'ポアソン方程式 FEM 三角形一次要素モデル

Dim nnode, ne, nrest, i, j, ind, ie, p, ir

Dim x(100), y(100), top(100, 5), nodes(400, 3), restraints(100, 2), k1, k2

Dim x1, x2, x3, y1, y2, y3, aa, bb, Ae, KS(4, 4), fe(100, 4), f(400)

Dim K(100, 4, 4), Kel(100, 4, 4), KG(400, 400), indp, indj

Dim Kmod(400, 400), fmod(400), pp, a(400, 400), c(400), u(400), s

'インプット (1) -----

nnode = Cells(2, 2): ne = Cells(2, 4): nrest = Cells(3, 2): ff = Cells(2, 7)

'インプット (2) -----

For ind = 1 To nnode

x(ind) = Cells(ind + 5, 2): y(ind) = Cells(ind + 5, 3)

Next

For p = 1 To 3: '4==> for triangle

For ie = 1 To ne

top(ie, p) = Cells(ie + 5, p + 4)

Next

Next

For j = 1 To 2

```

For i = 1 To nrest
restraints(i, j) = Cells(i + 5, j + 8)
Next
Next

```

'要素剛性マトリクスと内力ベクトル作成-----

```

For i = 1 To nnode
f(i) = 0
Next
For ie = 1 To ne
x1 = x(top(ie, 1)): x2 = x(top(ie, 2)): x3 = x(top(ie, 3))
y1 = y(top(ie, 1)): y2 = y(top(ie, 2)): y3 = y(top(ie, 3))

```

stiffnessmatrix KS(), x1, x2, x3, y1, y2, y3, ff, fe(), ie

```

For p = 1 To 3
For j = 1 To 3
Kel(ie, p, j) = KS(p, j)
Next
Next
For p = 1 To 3
f(top(ie, p)) = f(top(ie, p)) + fe(ie, p)
Next
Next

```

'全体剛性マトリクスの組立てサブルーチン呼び出し-----

globmat ne, nnode, top(), Kel(), KG()

'境界条件を加味した剛性マトリクスと内力ベクトルの修正サブルーチン呼び出し----

mofification nnode, nrest, KG(), f(), fmod(), Kmod(), restraints()

'連立一次方程式解法サブルーチンの呼び出し-----

```

For j = 1 To nnode
c(j) = fmod(j)
For i = 1 To nnode
a(i, j) = Kmod(i, j)
Next
Next
matsol nnode, a(), c(), u()

```

'アウトプット----

```

For i = 1 To nnode

Cells(i + 5, 14) = u(i)
Cells(i + 5, 15) = fmod(i)
Next
End Sub

```

' (3) サブルーチン

Sub stiffnessmatrix(KS(), x1, x2, x3, y1, y2, y3, ff, fe(), ie)

```

a1 = x2 * y3 - x3 * y2: b1 = y2 - y3: c1 = -(x2 - x3)
a2 = x3 * y1 - x1 * y3: b2 = y3 - y1: c2 = -(x3 - x1)
a3 = x1 * y2 - x2 * y1: b3 = y1 - y2: c3 = -(x1 - x2)
Ae = (a1 + a2 + a3) / 2

```

$KS(1, 1) = b1^2 + c1^2$: $KS(1, 2) = b1 * b2 + c1 * c2$: $KS(1, 3) = b1 * b3 + c1 * c3$
 $KS(2, 1) = b2 * b1 + c2 * c1$: $KS(2, 2) = b2^2 + c2^2$: $KS(2, 3) = b2 * b3 + c2 * c3$
 $KS(3, 1) = b3 * b1 + c3 * c1$: $KS(3, 2) = b3 * b2 + c3 * c2$: $KS(3, 3) = b3^2 + c3^2$

```

For i = 1 To 3
For j = 1 To 3
KS(i, j) = KS(i, j) / 4 / Ae
Next
Next

```

```

For p = 1 To 3
fe(ie, p) = ff / 3 * 1 * Ae
Next

```

End Sub

Sub globmat(ne, nnode, top(), Kel(), KG())

```

For i = 1 To nnode
For j = 1 To nnode
KG(i, j) = 0
Next
Next

```

```

For ie = 1 To ne
For p = 1 To 3
For j = 1 To 3
indp = top(ie, p)
indj = top(ie, j)

```

$KG(indp, indj) = KG(indp, indj) + Kel(ie, p, j)$

```

Next
Next
Next
End Sub

```

Sub modification(nnode, nrest, KG(), f(), fmod(), Kmod(), restraints())

```

For i = 1 To nnode
fmod(i) = f(i)
For j = 1 To nnode
Kmod(i, j) = KG(i, j)
Next
Next

```

```

For indp = 1 To nnode
For indj = 1 To nnode
For ir = 1 To nrest
If indp = restraints(ir, 1) Then
Kmod(indp, indj) = 0
Kmod(indj, indp) = 0
Kmod(indp, indp) = 1
End If
Next
Next
Next

```

```

For i = 1 To nnode: '-----modification of the force vector

```

```

For j = 1 To nnode
fmod(j) = fmod(j) - KG(j, restraints(i, 1)) * f(restraints(i, 2))
Next
Next
For ir = 1 To nrest
For indp = 1 To nnode
If indp = restraints(ir, 1) Then
fmod(indp) = restraints(ir, 2)
End If
Next
Next
End Sub

```

```

Sub matsol(nnode, a(), c(), u())
For j = 1 To nnode
For i = j + 1 To nnode
pp = a(i, j) / a(j, j)
c(i) = c(i) - c(j) * pp

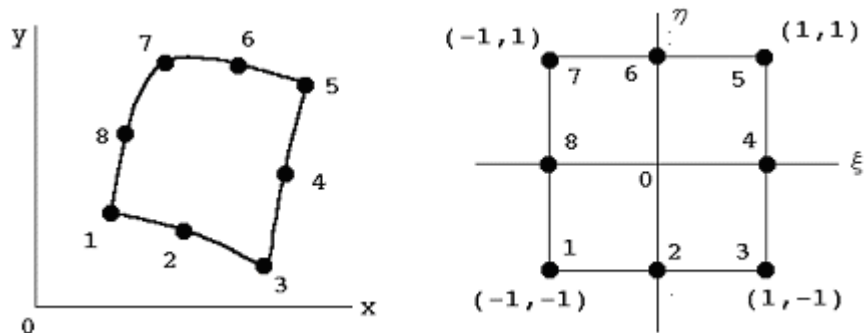
For ind = j To nnode - 1
a(i, ind) = a(i, ind) - a(j, ind) * pp
Next
Next
Next
u(nnode) = c(nnode) / a(nnode, nnode)
For j = nnode - 1 To 1 Step -1
s = 0
For i = j + 1 To nnode - 1
s = s + a(j, i) * u(i)
Next
u(j) = (c(j) - s) / a(j, j)
Next
End Sub

```

写像をもちいた四角形2次要素モデル

写像、マスター要素、Gaussの近似積分

四角形2次要素モデルは先に学んだ四角形一次要素モデルよりも少数の分割要素数で精度の高い解が得られる。そして、写像を併用すると、不規則な形状の断面をモデル化するのに極めて都合がよい。ここでは、四角形2次要素の形状関数として、1要素に8点を配置するセレンデピテイ関数を取り上げる。



左図 にしめすように、 $x-y$ 座標系に、ある一つの要素に反時計回りに節点 1 (x_1, y_1)、2 (x_2, y_2)、3 (x_3, y_3)、...8 (x_8, y_8) を配置する。この要素を $x-y$ 座標系に $(-1, -1)$ と $(1, 1)$ の間の区域を占める正方形に写像して、右図のように節点を 1 ($-1, -1$)、2 ($0, -1$)、...8 ($-1, 0$) と定めてこれをマスター要素とする。

要素剛性マトリクス、内力ベクトルを作成する際には面積分を行わなければならないが、左図の要素での積分は領域の境界の形が素直でなく難しいので、右図のマスター要素へ移して正方形の領域で積分の近似をやってやろうというのである。

理論は省いてテクニックを順に見てゆこう。

$N_1(\xi, \zeta)$ 、 $N_2(\xi, \zeta)$ 、... $N_8(\xi, \zeta)$ を 8 個の (ξ, ζ) の形状関数としよう。

そして、右図の要素の中の任意の位置の座標 (ξ, ζ) は、左図の $x-y$ 座標系での要素の中の座標 (x, y) に次の式で対応させる。

$$x = N1(\xi, \eta) \cdot x_1 + N2(\xi, \eta) \cdot x_2 + \dots + N8(\xi, \eta) \cdot x_8$$

$$y = N1(\xi, \eta) \cdot y_1 + N2(\xi, \eta) \cdot y_2 + \dots + N8(\xi, \eta) \cdot y_8$$

この (x, y) と (ξ, η) の関係の上で

$$Ke_{ij} = \int_{\Omega_e} \left(\frac{\partial N_i}{\partial x} \cdot \frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y} \cdot \frac{\partial N_j}{\partial y} \right) \cdot dA$$

$$fe_i = \int_{\Omega_e} N_i \cdot f \cdot dA$$

の面積分を (ξ, η) の座標系で近似計算する。その手順は $\frac{\partial N}{\partial x}$ 、 $\frac{\partial N}{\partial y}$ を

(ξ, η) の座標系でもとめるために、とりあえず、

$$\begin{pmatrix} \frac{\partial N_i}{\partial \xi} \\ \frac{\partial N_i}{\partial \eta} \end{pmatrix} = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{pmatrix} \cdot \begin{pmatrix} \frac{\partial N_i}{\partial x} \\ \frac{\partial N_i}{\partial y} \end{pmatrix}$$

$$J = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{pmatrix}$$

で J (ヤコビアンという) もとめ、この J の逆行列を計算することで、

$$\begin{pmatrix} \frac{\partial N_i}{\partial x} \\ \frac{\partial N_i}{\partial y} \end{pmatrix} = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{pmatrix}^{-1} \cdot \begin{pmatrix} \frac{\partial N_i}{\partial \xi} \\ \frac{\partial N_i}{\partial \eta} \end{pmatrix}$$

$$J^{-1} = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{pmatrix}^{-1}$$

と $\frac{\partial N}{\partial x}$ 、 $\frac{\partial N}{\partial y}$ を求める。

要素剛性マトリクスの成分をつくるための、 x - y 座標での面積分、内力ベクトルの成分をつくるための面積分は、次の方法で ξ - η 座標での正方形にたいする面積分で求めることができる。

$$\iint f(x, y) dx dy = \iint f[g(\xi, \eta), h(\xi, \eta)] \cdot |J(\xi, \eta)| \cdot d\xi d\eta$$

この式の意味は次のとおりである。左図の領域上での面積分とは、まな板の上にあの領域の形に囲いを作り、その中にメリケン粉を入れて、領域内の各点 (x, y) に応ずる高さが関数の値となるように粉を盛る。さて囲いの中の粉の体積はいくらかというのが上の左辺の意味である。ところが左図の領域の境界の形は素直でないので、その粉を右図の正方形の囲いの中に移し変えて、わかりやすくして計算しようというのが右辺の意味である。

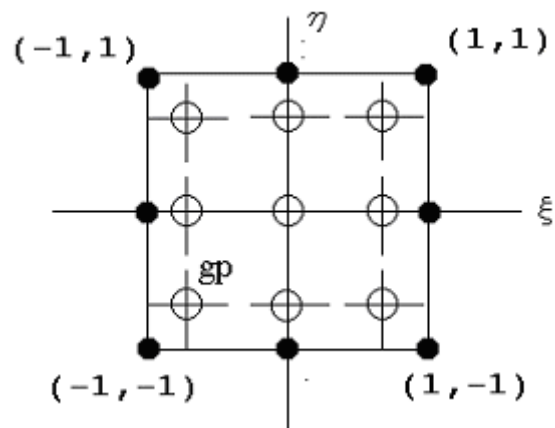
ここで、移し変え（写像）のために、 $x = g(\xi, \eta)$ 、 $y = h(\xi, \eta)$ という式が使われる。この移し変えによって領域の形が変わるので、当然同じ分量の粉はその深さがかわる。その比率が $|J(\xi, \eta)|$ というわけである。ここで、 $|J(\xi, \eta)|$ は単に下の式を記号簡単にしたに過ぎない。

$$|J(\xi, \eta)| = \begin{vmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{vmatrix} = \frac{\partial x}{\partial \xi} \cdot \frac{\partial y}{\partial \eta} - \frac{\partial y}{\partial \xi} \cdot \frac{\partial x}{\partial \eta}$$

次式

$$\iint f(x, y) dx dy = \iint f[g(\xi, \eta), h(\xi, \eta)] \cdot |J(\xi, \eta)| \cdot d\xi d\eta$$

の右辺の積分は次の Gauss の近似積分というものによって実行する。



図の白丸に示された 9 個の点を設ける。その座標は下の gp の値が (,) に使われるがどの点がどの値を使用しているかは図から明白であろう。

$$gp \equiv \begin{bmatrix} -.774596669241483 \\ 0 \\ .774596669241483 \end{bmatrix}$$

これらの点を Gauss ポイントと名づける。

$$w \equiv \begin{bmatrix} .555555555555556 \\ .888888888888889 \\ .555555555555556 \end{bmatrix}$$

上にしめされた値 w は重み係数と名づけられ、これらの値は gp の値に対応する。

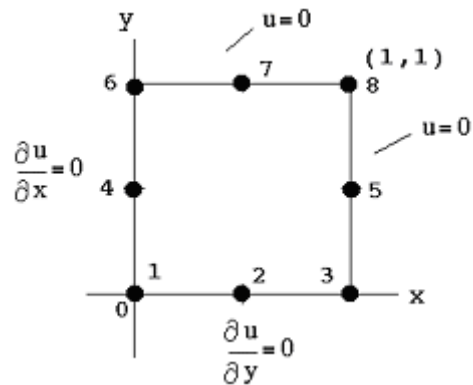
Gauss の近似積分はこれら 9 個の Gauss ポイントにおける、関数の値に重み係数をかけたものの 9 個の値を合計することで済まそうというのである。式で表せば、次のとおりである。

$$\iint G(\xi, \eta) \cdot d\xi d\eta = \sum_{i=1}^3 \sum_{j=1}^3 w_i w_j G(gp_i, gp_j)$$

これで準備が出来上がった。最も簡単な例を使ってプログラムを作ろう。

四角2次要素モデルのプログラム化

最も簡単な1要素8節点のモデルのプログラム化を行おう。このプログラムは入力さえ多くすれば多くの要素・節点をも扱うことができるが、見やすいように単純なモデルでスタートする。8個の節点にグローバルな節点番号1, 2, ..., 8をつける。



まず、形状関数を定義する。これは2次のセレンデピテイ関数と呼ばれる。

$$N(\xi, \eta) := \frac{1}{4} \begin{bmatrix} -(-1+\xi) \cdot (-1+\eta) \cdot (\xi+\eta+1) \\ 2 \cdot (-1+\xi^2) \cdot (-1+\eta) \\ (1+\xi) \cdot (-1+\eta) \cdot (-\xi+\eta+1) \\ -2 \cdot (1+\xi) \cdot (-1+\eta^2) \\ (1+\xi) \cdot (1+\eta) \cdot (\xi+\eta-1) \\ -2 \cdot (-1+\xi^2) \cdot (1+\eta) \\ -(-1+\xi) \cdot (1+\eta) \cdot (-\xi+\eta-1) \\ 2 \cdot (-1+\xi) \cdot (-1+\eta^2) \end{bmatrix}$$

つぎに $\frac{\partial N}{\partial \xi}$ 、 $\frac{\partial N}{\partial \eta}$ を上の式から求め、関数の形で定義しておこう。

$$dNd\xi(\xi, \eta) := \frac{1}{4} \cdot \begin{bmatrix} 2 \cdot \xi - 2 \cdot \eta \cdot \xi - \eta^2 + \eta \\ 4 \cdot \xi \cdot (-1 + \eta) \\ 2 \cdot \xi - 2 \cdot \eta \cdot \xi + \eta^2 - \eta \\ 2 - 2 \cdot \eta^2 \\ 2 \cdot \xi + 2 \cdot \eta \cdot \xi + \eta^2 + \eta \\ -4 \cdot \xi \cdot (1 + \eta) \\ 2 \cdot \xi + 2 \cdot \eta \cdot \xi - \eta^2 - \eta \\ -2 + 2 \cdot \eta^2 \end{bmatrix}$$

$$dNd\eta(\xi, \eta) := \frac{1}{4} \cdot \begin{bmatrix} 2 \cdot \eta - \xi^2 - 2 \cdot \eta \cdot \xi + \xi \\ -2 + 2 \cdot \xi^2 \\ 2 \cdot \eta - \xi^2 + 2 \cdot \eta \cdot \xi - \xi \\ -4 \cdot (1 + \xi) \cdot \eta \\ 2 \cdot \eta + \xi^2 + 2 \cdot \eta \cdot \xi + \xi \\ 2 - 2 \cdot \xi^2 \\ 2 \cdot \eta + \xi^2 - 2 \cdot \eta \cdot \xi - \xi \\ 4 \cdot (-1 + \xi) \cdot \eta \end{bmatrix}$$

これで、これらの関数は後の演算の時に自動的に呼び出されることになる。

さて、入力を始めよう。

$$\text{Nodes} := \begin{bmatrix} 1 & 0 & 0 \\ 2 & 0.5 & 0 \\ 3 & 1 & 0 \\ 4 & 0 & 0.5 \\ 5 & 1 & 0.5 \\ 6 & 0 & 1 \\ 7 & 0.5 & 1 \\ 8 & 1 & 1 \end{bmatrix}$$

$$\text{Top} := (1 \ 2 \ 3 \ 5 \ 8 \ 7 \ 6 \ 4)$$

$$n_{\text{nodes}} := \text{rows}(\text{Nodes}) \quad n_{\text{el}} := \text{rows}(\text{Top}) \quad ie := 1..n_{\text{el}}$$

ie 番目の要素のローカル節点番号 1, 2, ..., 8 に対する、x - y 座標を求めよう。

$$x_{n_{j,ie}} := \text{Nodes}(\text{Top}_{ie,j}, 2) \quad y_{n_{j,ie}} := \text{Nodes}(\text{Top}_{ie,j}, 3)$$

x と y は次の式によって 変換されるが、式中の x_1, x_2, \dots, x_8 と y_1, y_2, \dots, y_8 は与えられた座標の数値であるから、これらは定数である。

$$x = N1(\xi, \eta) \cdot x_1 + N2(\xi, \eta) \cdot x_2 + \dots + N8(\xi, \eta) \cdot x_8$$

$$y = N1(\xi, \eta) \cdot y_1 + N2(\xi, \eta) \cdot y_2 + \dots + N8(\xi, \eta) \cdot y_8$$

J (ヤコビアン) を求めるための $\frac{\partial x}{\partial \xi}$ 、 $\frac{\partial x}{\partial \eta}$ 、 $\frac{\partial y}{\partial \xi}$ 、 $\frac{\partial y}{\partial \eta}$ は上の式で N1, N2,...N8 を、 で偏微分すればえられる。 結局それらは

$$dx d\xi(\xi, \eta, ie) := dN d\xi(\xi, \eta) \cdot x_n^{<ie>}$$

$$dy d\xi(\xi, \eta, ie) := dN d\xi(\xi, \eta) \cdot y_n^{<ie>}$$

$$dx d\eta(\xi, \eta, ie) := dN d\eta(\xi, \eta) \cdot x_n^{<ie>}$$

$$dy d\eta(\xi, \eta, ie) := dN d\eta(\xi, \eta) \cdot y_n^{<ie>}$$

J (ヤコビアン) は約束にしたがって、

$$J(\xi, \eta, ie) := \begin{bmatrix} dx d\xi(\xi, \eta, ie) & dy d\xi(\xi, \eta, ie) \\ dx d\eta(\xi, \eta, ie) & dy d\eta(\xi, \eta, ie) \end{bmatrix}$$

J (ヤコビアン) の逆行列の成分を使って $\frac{\partial N}{\partial x}$ 、 $\frac{\partial N}{\partial y}$ を計算する。

$$DN dx(\xi, \eta, ie) := (J(\xi, \eta, ie)^{-1})_{1,1} \cdot dN d\xi(\xi, \eta) + (J(\xi, \eta, ie)^{-1})_{1,2} \cdot dN d\eta(\xi, \eta)$$

$$DN dy(\xi, \eta, ie) := (J(\xi, \eta, ie)^{-1})_{2,1} \cdot dN d\xi(\xi, \eta) + (J(\xi, \eta, ie)^{-1})_{2,2} \cdot dN d\eta(\xi, \eta)$$

これで、要素剛性マトリクスの成分を求めるための面積分の中身がきまった。

$$KM(\xi, \eta, ie) := (DN dx(\xi, \eta, ie) \cdot DN dx(\xi, \eta, ie)^T + DN dy(\xi, \eta, ie) \cdot DN dy(\xi, \eta, ie)^T) \cdot |J(\xi, \eta, ie)|$$

同様に内力ベクトルを計算するための面積分の中身も決まった。

$$ff := 2$$

$$Q(\xi, \eta, ie) := ff \cdot N(\xi, \eta) \cdot |J(\xi, \eta, ie)|$$

いよいよ Gauss の近似積分に取り掛かりよう。 まずは Gauss ポイント gp の座標の数値と重み係数 w を与える。

$$gp \equiv \begin{bmatrix} -.774596669241483 \\ 0 \\ .774596669241483 \end{bmatrix} \quad w \equiv \begin{bmatrix} .555555555555556 \\ .888888888888889 \\ .555555555555556 \end{bmatrix}$$

次に、要素剛性マトリクスの成分、

$$n_{gp} := 3 \quad i := 1..n_{gp} \quad j := 1..n_{gp}$$

$$Kel_{ie} := \left(\sum_i \sum_j w_i \cdot w_j \cdot KM(gp_i, gp_j, ie) \right)$$

そして、内力マトリクス、

$$F_{ie} := \left(\sum_i \sum_j w_i \cdot w_j \cdot Q(gp_i, gp_j, ie) \right)$$

ここまでくれば、拘束条件を入れて、

$$\text{Restrains} := \begin{bmatrix} 3 & 0 \\ 5 & 0 \\ 8 & 0 \\ 7 & 0 \\ 6 & 0 \end{bmatrix}$$

全体の荷重ベクトルを組立てる。

$$ii := 1..n_{nodes} \quad jj := 1..ndof_{pe} \quad \text{Loads}_{ii,1} := ii \quad \text{Loads}_{ii,2} := 0$$

$$ind := 1..n_{nodes}$$

$$\text{Loads}_{ind,2} := \sum_{p=1}^{nn_{pe}} \sum_{ie=1}^{n_{el}} (\text{ind} = \text{Top}_{ie,p}) \cdot (F_{ie})_p$$

全体の剛性マトリクスをおなじみの式で組立てる。

$$n := n_{nodes} \quad K_{n,n} := 0$$

$$i := 1..8 \quad j := 1..8$$

$$K_{(Top_{ie,i}), (Top_{ie,j})} := K_{(Top_{ie,i}), Top_{ie,j}} + (Kel_{ie})_{i,j}$$

これらにしたがって、全体の剛性マトリクスと内力を境界条件で修正する段階にはいった。まず、Kmod はおなじみの

$$n_{res} := \text{rows}(\text{Restrains}) \quad id := 1..n_{res}$$

$$indp := 1..n_{nodes} \quad indj := 1..n_{nodes}$$

$$Kmod := K$$

$$Kmod_{\text{Restrains}_{id,1}, indj} := 0$$

$$Kmod_{indj, \text{Restrains}_{id,1}} := 0$$

$$Kmod_{(\text{Restrains}_{id,1}), \text{Restrains}_{id,1}} := 1$$

fmod は

$$nl := \text{rows}(\text{Loads}) \quad il := 1..nl$$

$$fmod_{indp} := 0$$

$$fmod_{(\text{Loads}_{il,1})} := fmod_{(\text{Loads}_{il,1})} + \text{Loads}_{il,2}$$

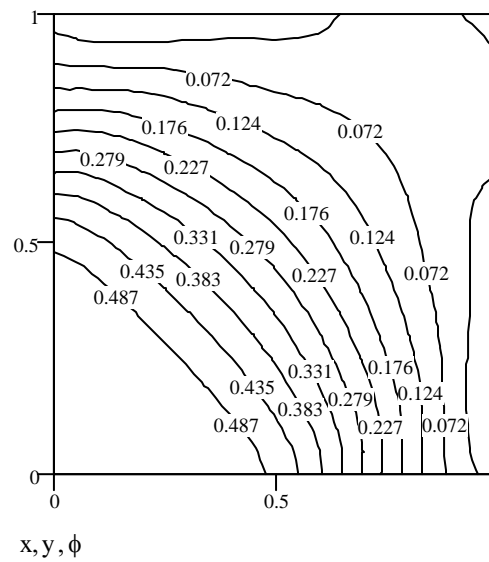
$$fmod_{indp} := fmod_{indp} - K_{indp, \text{Restrains}_{id,1}} \cdot \text{Restrains}_{id,2}$$

$$fmod_{(\text{Restrains}_{id,1})} := \text{Restrains}_{id,2}$$

これで一気に u を求めることができる。

$$\text{fmod} = \begin{bmatrix} -0.167 \\ 0.667 \\ 0 \\ 0.667 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad u := \text{Kmod}^{-1} \cdot \text{fmod} \quad u = \begin{bmatrix} 0.539 \\ 0.48 \\ 0 \\ 0.48 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

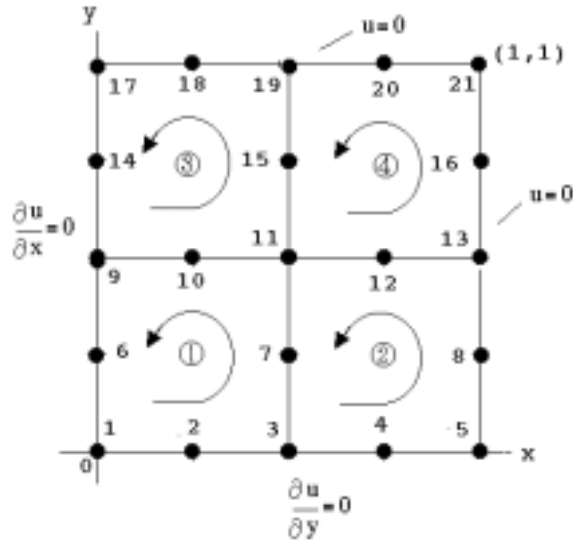
そして等高線



これで、で写像を使った四角 2 次要素モデルが出来上がった。これらの筋書きをアレンジして Excel VBA のプログラムを作成して見よう。

Excel VBA による四角2次有限要素モデル

4要素 21節点、正方形柱のねじりの問題で、やはり、1/4 の部分を取り上げる。
要素番号と節点番号は次のとおりである。



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	ポアソン微分方程式の有限方程式による解法 四角2次要素モデル																		
2	節点数 = 21 要素数 = 4 n = 2																		
3	拘束点数 9																		
4																			
5	節点番号	x	y	Topology	拘束番号	拘束値	節点番号	u の値	fmod										
6	1	0	0	1 1 2 3 7 11 10 9 6	5	0	1	0.5881893	-0.04										
7	2	0.25	0	2 3 4 5 8 13 12 11 7	8	0	2	0.5580762	0.167										
8	3	0.5	0	3 9 10 11 15 19 18 17 14	13	0	3	0.4583349	-0.08										
9	4	0.75	0	4 11 12 13 16 21 20 19 15	16	0	4	0.2792466	0.167										
10	5	1	0	5 6 7 8 9 10 11 12 13	21	0	5	0	0										
11	6	0	0.25	6 14 15 16 17 18 19 20	20	0	6	0.5580762	0.167										
12	7	0.5	0.25	7 12 13 14 15 16 17 18	19	0	7	0.4355035	0.333										
13	8	1	0.25	8 13 14 15 16 17 18 19	18	0	8	0	0										
14	9	0	0.5	9 14 15 16 17 18 19 20	17	0	9	0.4583349	-0.08										
15	10	0.25	0.5	10 15 16 17 18 19 20 21			10	0.4355035	0.333										
16	11	0.5	0.5	11 16 17 18 19 20 21			11	0.3584754	-0.17										
17	12	0.75	0.5	12 17 18 19 20 21			12	0.22885	0.333										
18	13	1	0.5	13 18 19 20 21			13	0	0										
19	14	0	0.75	14 19 20 21			14	0.2792466	0.167										
20	15	0.5	0.75	15 20 21			15	0.22885	0.333										
21	16	1	0.75	16 21			16	0	0										
22	17	0	1				17	0	0										
23	18	0.25	1				18	0	0										
24	19	0.5	1				19	0	0										
25	20	0.75	1				20	0	0										
26	21	1	1				21	0	0										
27																			

要素剛性マトリクスと内力ベクトルはメインプログラムで作成する。 各要素において、その節点に対応する反時計まわりのローカル番号 1、2、3...8 の x-y 座標をまず求める。 Gauss ポイントの 9 個の座標と重み係数を与えて、そ

の座標にたいする形状関数 N の値、 $\frac{\partial N}{\partial \xi}$ 、 $\frac{\partial N}{\partial \eta}$ を計算して、ヤコビアン J を

求めその逆行列から $\frac{\partial N}{\partial x}$ 、 $\frac{\partial N}{\partial y}$ を計算する。

形状関数とヤコビアンに関するプログラムはサブルーチンとする。そして、Gauss の近似積分を実行して要素剛性方程式と内力ベクトルを求める。その後の手順は今までと殆ど同じである。

サンプルプログラム

Sub Macro1()

'ポアソン方程式 FEM 四角2次要素モデル

Dim nnode, ne, nrest, i, j, ind, ie, p, ir

Dim x(100), y(100), top(100, 8), nodes(400, 3), restraints(100, 2)

Dim fe(100, 8), f(400), fmod(400), Q(100, 8)

Dim Kmod(400, 400), pp, a(400, 400), c(400), u(400), s

Dim gpx(9), gpy(9), ggpx(9), ggpy(9), ww(9), w(3)

Dim ex, ey, N(8), DNx(8), DNy(8), xx(8), yy(8), xxx(9, 8), yyy(9, 8)

Dim DNex(8), DNey(8), K(100, 8, 8), Kel(100, 8, 8), KG(400, 400), indp, indj

Dim dexx(8), dexy(8), deyy(8), deyx(8), dxex(8), dxey(8), dyex(8), dyey(8)

Dim ja11, ja12, ja21, ja22, detj, invj11, invj12, invj21, invj22

'インプット (1) -----

nnode = Cells(2, 2): ne = Cells(2, 4): nrest = Cells(3, 2): ff = Cells(2, 7)

'インプット (2) -----

For ind = 1 To nnode

x(ind) = Cells(ind + 5, 2): y(ind) = Cells(ind + 5, 3)

Next

For p = 1 To 8

For ie = 1 To ne

top(ie, p) = Cells(ie + 5, p + 4)

Next

Next

For j = 1 To 2

For i = 1 To nrest

restraints(i, j) = Cells(i + 5, j + 12)

Next

Next

'要素剛性マトリクスと内力ベクトル作成-----

For ie = 1 To ne

For p = 1 To 8

For j = 1 To 8

```

Kel(ie, p, j) = 0
Next
fe(ie, p) = 0
Next
Next

gauss gpx(), gpy(), w(), ww()

For ie = 1 To ne
For i = 1 To 9
ex = gpx(i)
ey = gpy(i)
For p = 1 To 8
xx(p) = x(top(ie, p))
yy(p) = y(top(ie, p))
shapefunct ex, ey, N(), DNx(), DNey()
Next
jacobian N(), DNx(), DNy(), DNex(), DNey(), xx(), yy(), ja11, ja12, ja21, ja22, detj, invj11, invj12,
invj21, invj22

For p = 1 To 8
For j = 1 To 8
K(ie, p, j) = (DNx(p) * DNx(j) + DNy(p) * DNy(j)) * detj
Kel(ie, p, j) = Kel(ie, p, j) + ww(i) * K(ie, p, j)
Next
Q(ie, p) = ff * N(p) * detj
fe(ie, p) = fe(ie, p) + ww(i) * Q(ie, p)

Next

Next
Next

'全体剛性マトリクスの組立てサブルーチン呼び出し-----
globmat ne, nnode, top(), Kel(), KG()

'全体ない力ベクトルの組立てサブルーチン呼び出し---
globforce ne, top(), fe(), f()

'境界条件を加味した剛性マトリクスと内力ベクトルの修正サブルーチン呼び出し----
modification nnode, nrest, KG(), f(), fmod(), Kmod(), restraints()

'連立一次方程式解法サブルーチンの呼び出し-----
For j = 1 To nnode
c(j) = fmod(j)
For i = 1 To nnode
a(i, j) = Kmod(i, j)
Next
Next
matsol nnode, a(), c(), u()

'アウトプット----
For i = 1 To nnode

```

Cells(i + 5, 18) = u(i)
Cells(i + 5, 19) = fmod(i)
Next

End Sub

Sub gauss(gpx(), gpy(), w(), ww())
gpx(1) = -0.7746: gpx(2) = -0.7746: gpx(3) = -0.7746: gpx(4) = 0: gpx(5) = 0
gpy(1) = -0.7746: gpy(2) = 0: gpy(3) = 0.7746: gpy(4) = -0.7746: gpy(5) = 0

gpx(6) = 0: gpx(7) = 0.7746: gpx(8) = 0.7746: gpx(9) = 0.7746
gpy(6) = 0.7746: gpy(7) = -0.7746: gpy(8) = 0: gpy(9) = 0.7746

w(1) = 5 / 9: w(2) = 8 / 9: w(3) = 5 / 9
ww(1) = w(1) * w(1): ww(2) = w(1) * w(2): ww(3) = w(1) * w(3)
ww(4) = w(2) * w(1): ww(5) = w(2) * w(2): ww(6) = w(2) * w(3)
ww(7) = w(3) * w(1): ww(8) = w(3) * w(2): ww(9) = w(3) * w(3)

End Sub

Sub shapefunct(ex, ey, N(), DNex(), DNey())

N(1) = -0.25 * (-1 + ex) * (-1 + ey) * (ex + ey + 1)
N(2) = 0.25 * 2 * (-1 + ex ^ 2) * (-1 + ey)
N(3) = 0.25 * (1 + ex) * (-1 + ey) * (-ex + ey + 1)
N(4) = -0.25 * 2 * (1 + ex) * (-1 + ey ^ 2)
N(5) = 0.25 * (1 + ex) * (1 + ey) * (ex + ey - 1)
N(6) = -0.25 * 2 * (-1 + ex ^ 2) * (1 + ey)
N(7) = -0.25 * (-1 + ex) * (1 + ey) * (-ex + ey - 1)
N(8) = 0.25 * 2 * (-1 + ex) * (-1 + ey ^ 2)
DNex(1) = 0.25 * (2 * ex - 2 * ey * ex - ey ^ 2 + ey)
DNex(2) = 0.25 * 4 * ex * (-1 + ey)
DNex(3) = 0.25 * (2 * ex - 2 * ey * ex + ey ^ 2 - ey)
DNex(4) = 0.25 * (2 - 2 * ey ^ 2)
DNex(5) = 0.25 * (2 * ex + 2 * ey * ex + ey ^ 2 + ey)
DNex(6) = 0.25 * (-4 * ex * (1 + ey))
DNex(7) = 0.25 * (2 * ex + 2 * ey * ex - ey ^ 2 - ey)
DNex(8) = 0.25 * (-2 + 2 * ey ^ 2)
DNey(1) = 0.25 * (2 * ey - ex ^ 2 - 2 * ey * ex + ex)
DNey(2) = 0.25 * (-2 + 2 * ex ^ 2)
DNey(3) = 0.25 * (2 * ey - ex ^ 2 + 2 * ey * ex - ex)
DNey(4) = 0.25 * (-4 * (1 + ex) * ey)
DNey(5) = 0.25 * (2 * ey + ex ^ 2 + 2 * ey * ex + ex)
DNey(6) = 0.25 * (2 - 2 * ex ^ 2)
DNey(7) = 0.25 * (2 * ey + ex ^ 2 - 2 * ey * ex - ex)
DNey(8) = 0.25 * 4 * (-1 + ex) * ey

End Sub

Sub jacobian(N(), DNx(), DNy(), DNex(), DNey(), xx(), yy(), ja11, ja12, ja21, ja22, detj, invj11,
invj12, invj21, invj22)

For p = 1 To 8
ja11 = 0: ja12 = 0: ja21 = 0: ja22 = 0

Next

For p = 1 To 8

ja11 = ja11 + DNex(p) * xx(p): ja12 = ja12 + DNex(p) * yy(p)

ja21 = ja21 + DNey(p) * xx(p): ja22 = ja22 + DNey(p) * yy(p)

Next

detj = ja11 * ja22 - ja12 * ja21

invj11 = ja22 / detj: invj12 = -ja12 / detj: invj21 = -ja21 / detj: invj22 = ja11 / detj

For p = 1 To 8

DNx(p) = invj11 * DNex(p) + invj12 * DNey(p)

DNy(p) = invj21 * DNex(p) + invj22 * DNey(p)

Next

End Sub

Sub globmat(ne, nnode, top(), Kel(), KG())

For i = 1 To nnode

For j = 1 To nnode

KG(i, j) = 0

Next

Next

For ie = 1 To ne

For p = 1 To 8

For j = 1 To 8

indp = top(ie, p)

indj = top(ie, j)

KG(indp, indj) = KG(indp, indj) + Kel(ie, p, j)

Next

Next

Next

End Sub

Sub globforce(ne, top(), fe(), f())

For ie = 1 To ne

For p = 1 To 8

f(top(ie, p)) = f(top(ie, p)) + fe(ie, p)

Next

Next

End Sub

Sub mofication(nnode, nrest, KG(), f(), fmod(), Kmod(), restraints())

For i = 1 To nnode

fmod(i) = f(i)

For j = 1 To nnode

Kmod(i, j) = KG(i, j)

Next

Next

For indp = 1 To nnode

For indj = 1 To nnode

For ir = 1 To nrest

If indp = restraints(ir, 1) Then

```

Kmod(indp, indj) = 0
Kmod(indj, indp) = 0
Kmod(indp, indp) = 1
End If
Next
Next
Next
For i = 1 To nnode: '-----modification of the force vector
For j = 1 To nnode
fmod(j) = fmod(j) - KG(j, restraints(i, 1)) * restraints(i, 2)
Next
Next
For ir = 1 To nrest
For indp = 1 To nnode
If indp = restraints(ir, 1) Then
fmod(indp) = restraints(ir, 2)
End If
Next
Next
End Sub

```

```

Sub matsol(nnode, a(), c(), u())
For j = 1 To nnode
For i = j + 1 To nnode
pp = a(i, j) / a(j, j)
c(i) = c(i) - c(j) * pp

For ind = j To nnode - 1
a(i, ind) = a(i, ind) - a(j, ind) * pp
Next
Next
Next
u(nnode) = c(nnode) / a(nnode, nnode)
For j = nnode - 1 To 1 Step -1
s = 0
For i = j + 1 To nnode - 1
s = s + a(j, i) * u(i)
Next
u(j) = (c(j) - s) / a(j, j)
Next
End Sub

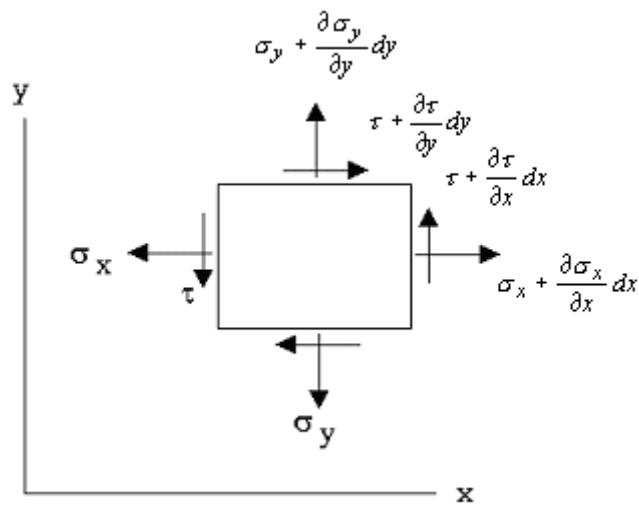
```

第4章 平面弾性問題

4.1 平面弾性における基礎式

2つの平行な平面にはさまれた厚さ h の弾性体 を考えよう。 厚さ h が の寸法に比べて非常に小さい場合は平面応力問題という。 厚さ h が の寸法に比べて非常に大きい場合は平面ひずみ問題という。

(1) 基本の関係式



力の釣合いの関係式：

f_x と f_y を x 、 y 方向に働く体積力とすると

$$\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau}{\partial y} + f_x = 0$$

$$\frac{\partial \tau}{\partial x} + \frac{\partial \sigma_y}{\partial y} + f_y = 0$$

ひずみと変位との関係式：

$$\varepsilon_x = \frac{\partial u}{\partial x} \quad , \quad \varepsilon_y = \frac{\partial v}{\partial y} \quad , \quad 2\gamma = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \quad .$$

応力とひずみとの関係式：

$$\begin{pmatrix} \sigma_x \\ \sigma_y \\ \tau \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} & 0 \\ c_{12} & c_{22} & 0 \\ 0 & 0 & c_{66} \end{pmatrix} \cdot \begin{pmatrix} \varepsilon_x \\ \varepsilon_y \\ 2\gamma \end{pmatrix}$$

ただし、材料を x 、 y 、 z 方向に等方性とし、 E を縦弾性係数、 ν をポアソン比とすると、

平面応力問題では、

$$c_{11} = \frac{E}{1-\nu^2} \quad c_{22} = \frac{E}{1-\nu^2}$$

$$c_{12} = \nu \cdot c_{11} = \nu \cdot c_{22}$$

$$c_{66} = G \quad G = \frac{E}{2(1+\nu)}$$

平面ひずみ問題では、

$$c_{11} = \frac{E \cdot (1-\nu)}{1-\nu-2 \cdot \nu^2} \quad c_{22} = c_{11}$$

$$c_{12} = \frac{\nu \cdot E}{1-\nu-2 \cdot \nu^2}$$

(2) 基本となる微分方程式

上記の関係式から次の微分方程式が導かれる。

$$-\frac{\partial}{\partial x} \left(c_{11} \frac{\partial u}{\partial x} + c_{12} \frac{\partial v}{\partial y} \right) - \frac{\partial}{\partial y} \left(c_{66} \cdot \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right) = f_x$$

$$-\frac{\partial}{\partial x} \left(c_{66} \cdot \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right) - \frac{\partial}{\partial y} \left(c_{11} \frac{\partial u}{\partial x} + c_{12} \frac{\partial v}{\partial y} \right) = f_y$$

上の式が有限要素法（近似的解法）で解かれる微分方程式である。また、

境界に対して発生する応力成分を変位 u , v で表現すると、

$$t_x = \left(c11 \frac{\partial u}{\partial x} + c12 \frac{\partial v}{\partial y} \right) \cdot n_x + \left(c66 \cdot \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right) \cdot n_y$$

$$t_y = \left(c66 \cdot \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right) \cdot n_x + \left(c11 \frac{\partial u}{\partial x} + c12 \frac{\partial v}{\partial y} \right) \cdot n_y$$

となる。ここで、 t_x 、 t_y は境界に発生する応力の成分で、 n_x 、 n_y は境界に対する法線の成分である。

(3) 微分方程式を近似的に解く準備

微分方程式の両辺に重み関数 $w1$ を掛けて要素の境界内で面積分し、下の方程式の両辺に重み関数 $w2$ を掛けてこれも要素の境界内で面積分し、部分積分とグリーンの定理を使って変形すると次の式がえられる。

$$h_e \cdot \int_{\Omega_e} \left(\frac{\partial w1}{\partial x} \left(c11 \frac{\partial u}{\partial x} + c12 \frac{\partial v}{\partial y} \right) + \frac{\partial w1}{\partial y} \left(c66 \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right) - w1 \cdot f_x \right) dA$$

$$- h_e \cdot \int_{\Gamma_e} w1 \cdot \left(\left(c11 \frac{\partial u}{\partial x} + c12 \frac{\partial v}{\partial y} \right) \cdot n_x + c66 \cdot \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \cdot n_y \right) ds = 0$$

$$h_e \cdot \int_{\Omega_e} \left(\frac{\partial w2}{\partial x} \left(c11 \frac{\partial u}{\partial x} + c12 \frac{\partial v}{\partial y} \right) + \frac{\partial w2}{\partial y} \left(c66 \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right) - w2 \cdot f_x \right) dA$$

$$- h_e \cdot \int_{\Gamma_e} w2 \cdot \left(\left(c11 \frac{\partial u}{\partial x} + c12 \frac{\partial v}{\partial y} \right) \cdot n_x + c66 \cdot \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \cdot n_y \right) ds = 0$$

ここで、 h_e は厚さである。左辺の第2項は下のようにも表すことができる。

$$h_e \cdot \oint_{\Gamma_e} w1 \cdot t_x ds \quad h_e \cdot \oint_{\Gamma_e} w2 \cdot t_y ds$$

(4) 剛性マトリクス、力のベクトル、剛性方程式

剛性マトリクスは4個のマトリクスが複合されたものとなり、 h を厚さとして、

$$K11_{ij} = \int_{\Omega_e} h \cdot \left(c11 \frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} + c66 \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} \right) dA$$

$$K12_{ij} = K21_{ji} = \int_{\Omega_e} h \cdot \left(c11 \frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial y} + c66 \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial x} \right) dA$$

$$K22_{ij} = \int_{\Omega_e} h \cdot \left(c66 \frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} + c22 \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} \right) dA$$

力のベクトルは、

$$F1_i = \int_{\Omega_e} h \cdot N_i \cdot f_x dA + \int_{\Gamma_e} h \cdot N_i \cdot t_x ds$$

$$F2_i = \int_{\Omega_e} h \cdot N_i \cdot f_y dA + \int_{\Gamma_e} h \cdot N_i \cdot t_y ds$$

剛性方程式は、

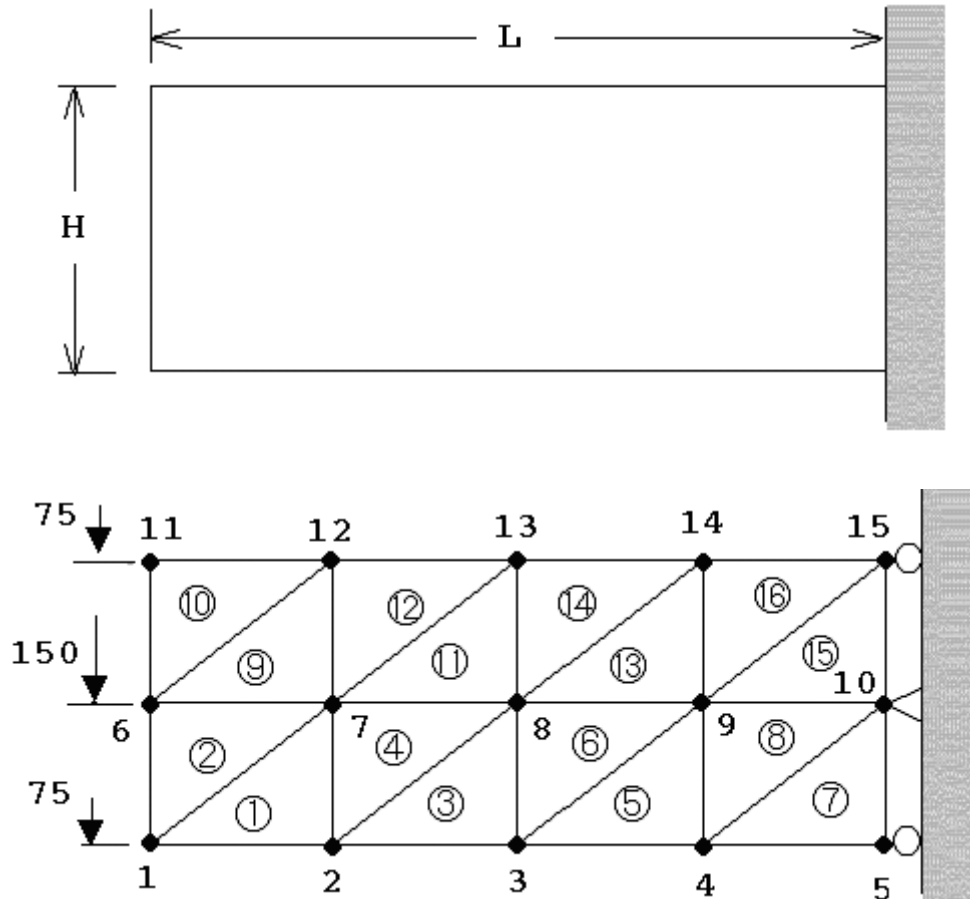
$$\begin{pmatrix} K11 & K12 \\ K21 & K22 \end{pmatrix} \cdot \begin{pmatrix} \{u\} \\ \{v\} \end{pmatrix} = \begin{pmatrix} \{F1\} \\ \{F2\} \end{pmatrix}$$

となる。

4.2. 平面応力問題－三角形1次要素モデル

(1) 要素剛性マトリクス

三角形1次要素モデルの形状関数 N_1 、 N_2 、 N_3 はねじりの場合とまったく同じである。早速例題に入ろう。



図にしめすような板状のはりがあり、その厚さ $t = 2.5 \text{ cm}$ 、長さ $L = 25 \text{ cm}$ 、高さ $H = 5 \text{ cm}$ として、先端に $75 + 150 + 75 = 300 \text{ N}$ の荷重がかかるものとする。板状のはりを図のように16個の要素に分割し、15個の節点を決める。各要素の3隅の節点は反時計まわりに、かつ直角の頂点が2番目になるようかぞえる。そのかぞえ方はトポロジ－テーブルから合点していただきたい。

ではまず材料の特性値の入力から開始しよう。

$$\begin{bmatrix} E \\ t \\ v \end{bmatrix} := \begin{bmatrix} 2.06 \cdot 10^7 \\ 2.5 \\ .25 \end{bmatrix} \quad \begin{bmatrix} c11 & c12 & c01 \\ c21 & c22 & c02 \\ c03 & c04 & c66 \end{bmatrix} := \frac{E}{1 - \nu^2} \cdot \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1 - \nu}{2} \end{bmatrix}$$

節点の座標、トポロジーテーブル、要素数、節点数、拘束点数などを入力する。

$$\begin{bmatrix} n_{el} \\ n_{nodes} \\ n_{res} \end{bmatrix} = \begin{bmatrix} 16 \\ 15 \\ 4 \end{bmatrix}$$

$$ie := 1..n_{el}$$

	1	2	3
1	1	0	0
2	2	6.25	0
3	3	12.5	0
4	4	18.75	0
5	5	25	0
6	6	0	2.5
7	7	6.25	2.5
8	8	12.5	2.5
9	9	18.75	2.5
10	10	25	2.5
11	11	0	5
12	12	6.25	5
13	13	12.5	5
14	14	18.75	5
15	15	25	5

Nodes =

	1	2	3
1	1	2	7
2	7	6	1
3	2	3	8
4	8	7	2
5	3	4	9
6	9	8	3
7	4	5	10
8	10	9	4
9	6	7	12
10	12	11	6
11	7	8	13
12	13	12	7
13	8	9	14
14	14	13	8
15	9	10	15
16	15	14	9

Top =

$$x_n := \text{Nodes}^{<2>}$$

$$y_n := \text{Nodes}^{<3>}$$

ねじりの問題で三角形要素についてまとめたおなじように、各要素にたいするローカル節点番号 1、2、3 にたいする座標をつぎのようにもとめ、

$$\begin{bmatrix} x_{1_{ie}} & y_{1_{ie}} \\ x_{2_{ie}} & y_{2_{ie}} \\ x_{3_{ie}} & y_{3_{ie}} \end{bmatrix} := \begin{bmatrix} x_{n(\text{Top}_{ie,1})} & y_{n(\text{Top}_{ie,1})} \\ x_{n(\text{Top}_{ie,2})} & y_{n(\text{Top}_{ie,2})} \\ x_{n(\text{Top}_{ie,3})} & y_{n(\text{Top}_{ie,3})} \end{bmatrix}$$

形状関数の係数をもとめるため、まず a, b, c を算出するところもねじり問題に同じ。

$$\begin{bmatrix} a_{1_{ie}} & b_{1_{ie}} & c_{1_{ie}} \\ a_{2_{ie}} & b_{2_{ie}} & c_{2_{ie}} \\ a_{3_{ie}} & b_{3_{ie}} & c_{3_{ie}} \end{bmatrix} := \begin{bmatrix} x_{2_{ie}} \cdot y_{3_{ie}} - x_{3_{ie}} \cdot y_{2_{ie}} & y_{2_{ie}} - y_{3_{ie}} & x_{3_{ie}} - x_{2_{ie}} \\ x_{3_{ie}} \cdot y_{1_{ie}} - x_{1_{ie}} \cdot y_{3_{ie}} & y_{3_{ie}} - y_{1_{ie}} & x_{1_{ie}} - x_{3_{ie}} \\ x_{1_{ie}} \cdot y_{2_{ie}} - x_{2_{ie}} \cdot y_{1_{ie}} & y_{1_{ie}} - y_{2_{ie}} & x_{2_{ie}} - x_{1_{ie}} \end{bmatrix} \quad A e_{ie} := \frac{a_{1_{ie}} + a_{2_{ie}} + a_{3_{ie}}}{2}$$

要素の剛性マトリクスをつくるに先だって、前節の剛性方程式

$$\begin{pmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{pmatrix} \cdot \begin{pmatrix} \{u\} \\ \{v\} \end{pmatrix} = \begin{pmatrix} \{F1\} \\ \{F2\} \end{pmatrix}$$

では各ローカル節点の u と v とが $\{u\}$ と $\{v\}$ のグループに分かれていて、ねじりの場合のように $\{u\}$ だけしかなかった場合に比べて、少し扱いに面倒そうなので、これらの成分 ($u_1, u_2, u_3, v_1, v_2, v_3$) を並べ替えて ($u_1, v_1, u_2, v_2, u_3, v_3$) というようにしよう。

そして、さらに u は奇数の $1, 3, 5, \dots$ のブスクリプトをとるものとし、

v は偶数の $2, 4, 6, \dots$ のサブスクリプトをとるものとする。

すなわち、トラスと同じように ($u_1, v_1, u_2, v_2, u_3, v_3$) \rightarrow ($u_1, u_2, u_3, u_4, u_5, u_6$) に置き換える。

そのために、3行3列のマトリクス $K_{11}, K_{12}, K_{21}, K_{22}$ を成分0を加えて6行に書き替えて、もとの成分が適当な場所にくるように配列する。配列のしかたを、とくに0が並んでいる様子を、ねじりの場合とくらべて、とくと観察していただきたい。ねじりの場合に比べて微分方程式の形がすこし複雑になった分、マトリクスも増えている。

まず K11は、

$$K_{el11_{ie}} := \frac{t \cdot c11}{4 \cdot Ae_{ie}} \cdot \begin{bmatrix} (b_{1_{ie}})^2 & 0 & b_{1_{ie}} \cdot b_{2_{ie}} & 0 & b_{1_{ie}} \cdot b_{3_{ie}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ b_{1_{ie}} \cdot b_{2_{ie}} & 0 & (b_{2_{ie}})^2 & 0 & b_{3_{ie}} \cdot b_{2_{ie}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ b_{1_{ie}} \cdot b_{3_{ie}} & 0 & b_{3_{ie}} \cdot b_{2_{ie}} & 0 & (b_{3_{ie}})^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} + \frac{t \cdot c66}{4 \cdot Ae_{ie}} \cdot \begin{bmatrix} (c_{1_{ie}})^2 & 0 & c_{1_{ie}} \cdot c_{2_{ie}} & 0 & c_{1_{ie}} \cdot c_{3_{ie}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ c_{1_{ie}} \cdot c_{2_{ie}} & 0 & (c_{2_{ie}})^2 & 0 & c_{3_{ie}} \cdot c_{2_{ie}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ c_{1_{ie}} \cdot c_{3_{ie}} & 0 & c_{3_{ie}} \cdot c_{2_{ie}} & 0 & (c_{3_{ie}})^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

K22は、

$$K_{el22_{ie}} := \frac{t \cdot c66}{4 \cdot Ae_{ie}} \cdot \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & (b_{1_{ie}})^2 & 0 & b_{1_{ie}} \cdot b_{2_{ie}} & 0 & b_{1_{ie}} \cdot b_{3_{ie}} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & b_{1_{ie}} \cdot b_{2_{ie}} & 0 & (b_{2_{ie}})^2 & 0 & b_{3_{ie}} \cdot b_{2_{ie}} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & b_{1_{ie}} \cdot b_{3_{ie}} & 0 & b_{3_{ie}} \cdot b_{2_{ie}} & 0 & (b_{3_{ie}})^2 \end{bmatrix} + \frac{t \cdot c22}{4 \cdot Ae_{ie}} \cdot \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & (c_{1_{ie}})^2 & 0 & c_{1_{ie}} \cdot c_{2_{ie}} & 0 & c_{1_{ie}} \cdot c_{3_{ie}} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & c_{1_{ie}} \cdot c_{2_{ie}} & 0 & (c_{2_{ie}})^2 & 0 & c_{3_{ie}} \cdot c_{2_{ie}} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & c_{1_{ie}} \cdot c_{3_{ie}} & 0 & c_{3_{ie}} \cdot c_{2_{ie}} & 0 & (c_{3_{ie}})^2 \end{bmatrix}$$

K12は、

$$K_{el12_{ie}} := \frac{t \cdot c12}{4 \cdot Ae_{ie}} \cdot \begin{bmatrix} 0 & b_{1_{ie}} \cdot c_{1_{ie}} & 0 & b_{1_{ie}} \cdot c_{2_{ie}} & 0 & b_{1_{ie}} \cdot c_{3_{ie}} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & b_{2_{ie}} \cdot c_{1_{ie}} & 0 & b_{2_{ie}} \cdot c_{2_{ie}} & 0 & b_{2_{ie}} \cdot c_{3_{ie}} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & b_{3_{ie}} \cdot c_{1_{ie}} & 0 & b_{3_{ie}} \cdot c_{2_{ie}} & 0 & b_{3_{ie}} \cdot c_{3_{ie}} \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} + \frac{t \cdot c66}{4 \cdot Ae_{ie}} \cdot \begin{bmatrix} 0 & b_{1_{ie}} \cdot c_{1_{ie}} & 0 & b_{2_{ie}} \cdot c_{1_{ie}} & 0 & b_{3_{ie}} \cdot c_{1_{ie}} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & b_{1_{ie}} \cdot c_{2_{ie}} & 0 & b_{2_{ie}} \cdot c_{2_{ie}} & 0 & b_{3_{ie}} \cdot c_{2_{ie}} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & b_{1_{ie}} \cdot c_{3_{ie}} & 0 & b_{2_{ie}} \cdot c_{3_{ie}} & 0 & b_{3_{ie}} \cdot c_{3_{ie}} \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

K21は、

$$K_{el21_{ie}} := \frac{t \cdot c66}{4 \cdot A e_{ie}} \cdot \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ b_{1_{ie}} \cdot c_{1_{ie}} & 0 & b_{1_{ie}} \cdot c_{2_{ie}} & 0 & b_{1_{ie}} \cdot c_{3_{ie}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ b_{2_{ie}} \cdot c_{1_{ie}} & 0 & b_{2_{ie}} \cdot c_{2_{ie}} & 0 & b_{2_{ie}} \cdot c_{3_{ie}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ b_{3_{ie}} \cdot c_{1_{ie}} & 0 & b_{3_{ie}} \cdot c_{2_{ie}} & 0 & b_{3_{ie}} \cdot c_{3_{ie}} & 0 \end{bmatrix} + \frac{t \cdot c12}{4 \cdot A e_{ie}} \cdot \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ b_{1_{ie}} \cdot c_{1_{ie}} & 0 & b_{2_{ie}} \cdot c_{1_{ie}} & 0 & b_{3_{ie}} \cdot c_{1_{ie}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ b_{1_{ie}} \cdot c_{2_{ie}} & 0 & b_{2_{ie}} \cdot c_{2_{ie}} & 0 & b_{3_{ie}} \cdot c_{2_{ie}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ b_{1_{ie}} \cdot c_{3_{ie}} & 0 & b_{2_{ie}} \cdot c_{3_{ie}} & 0 & b_{3_{ie}} \cdot c_{3_{ie}} & 0 \end{bmatrix}$$

結局、要素剛性マトリクス Kel は、

$$K_{el_{ie}} := K_{el11_{ie}} + K_{el12_{ie}} + K_{el21_{ie}} + K_{el22_{ie}}$$

(2) 全体剛性マトリクス、境界条件作成から変位の算出まで

要素剛性マトリクス Kel が出来上がれば、後の手順はトラスの場合とほとんど同じである。ねじりの場合は一節点につき一個の変数 u、トラスの場合は一節点につき2個の u、フレームの場合は一節点につき3個の変数 u であった。それぞれに対して、全体剛性マトリクスの組み立ての式は若干異なっているが、平面応力の場合はトラスの場合と同じである。

$$n := n_{nodes} \cdot 2 \quad K_{n,n} := 0$$

$$ie := 1..n_{el} \quad i := 1..3 \quad j := 1..3 \quad p := 1..2 \quad q := 1..2$$

$$K[(Top_{ie,i}-1) \cdot 2+p], (Top_{ie,j}-1) \cdot 2+q := K[(Top_{ie,i}-1) \cdot 2+p], (Top_{ie,j}-1) \cdot 2+q \cdots + (K_{el_{ie}})_{(i-1) \cdot 2+p, (j-1) \cdot 2+q}$$

拘束条件と外力ベクトルが次のように与えられる。

$$\text{Restrains} := \begin{bmatrix} 9 & 0 \\ 19 & 0 \\ 20 & 0 \\ 29 & 0 \end{bmatrix} \quad \text{Loads} := \begin{bmatrix} 2 & -75 \\ 12 & -150 \\ 22 & -75 \end{bmatrix}$$

拘束条件と外力ベクトルによる、全体剛性マトリクスの修正と荷重ベクトルの修正は次のように行われる。

$$p := 1..nn_{pe} \quad nn_{pe} = 3 \quad id := 1..n_{res}$$

$$indp := 1..2 \cdot n_{nodes} \quad indj := 1..2 \cdot n_{nodes}$$

$$K_{mod} := K$$

$$K_{mod}_{Restrains_{id,1}, indj} := 0$$

$$K_{mod}_{indj, Restrains_{id,1}} := 0$$

$$K_{mod}_{(Restrains_{id,1}), Restrains_{id,1}} := 1$$

$$nl := rows(Loads) \quad il := 1..nl$$

$$f_{mod}_{indp} := 0$$

$$f_{mod}_{(Loads_{il,1})} := f_{mod}_{(Loads_{il,1})} + Loads_{il,2}$$

$$f_{mod}_{indp} := f_{mod}_{indp} - K_{indp, Restrains_{id,1}} \cdot Restrains_{id,2}$$

$$f_{mod}_{(Restrains_{id,1})} := Restrains_{id,2}$$

これらによって、以下のように u が求められる。 u は全部で節点数掛ける2で30個あるが以下では最初の14個だけが示されている。

$$u := K_{\text{mod}}^{-1} \cdot f_{\text{mod}}$$

u =

	1
1	$1.308 \cdot 10^{-4}$
2	$-9.379 \cdot 10^{-4}$
3	$1.234 \cdot 10^{-4}$
4	$-6.032 \cdot 10^{-4}$
5	$1.001 \cdot 10^{-4}$
6	$-3.085 \cdot 10^{-4}$
7	$6.044 \cdot 10^{-5}$
8	$-9.407 \cdot 10^{-5}$
9	0
10	$-1.103 \cdot 10^{-5}$
11	$-6.045 \cdot 10^{-8}$
12	$-9.386 \cdot 10^{-4}$
13	$6.327 \cdot 10^{-7}$
14	$-6.034 \cdot 10^{-4}$

(3) ポストプロセッシング (Postprocessing) - ひずみを求める

計算によって各節点の変位が求められた。奇数のサブスクリプトが x 方向の変位 U をあらわし、偶数のサブスクリプトが y 方向の変位 V を表すものとする、各要素の頂点 1, 2, 3 番目にある節点の変位 U 、 V は次のように求められる。

$$j := 1..3 \quad U_{ie,j} := u_{2 \cdot \text{Top}_{ie,j} - 1} \quad V_{ie,j} := u_{2 \cdot \text{Top}_{ie,j}}$$

さて、三角形一次要素の形状関数は ie 番目の要素にたいして次のようにあらわされる。

$$N1(ie, x, y) := \frac{1}{2 \cdot Ae_{ie}} \cdot (a_{1ie} + b_{1ie} \cdot x + c_{1ie} \cdot y)$$

$$N2(ie, x, y) := \frac{1}{2 \cdot Ae_{ie}} \cdot (a_{2ie} + b_{2ie} \cdot x + c_{2ie} \cdot y)$$

$$N3(ie, x, y) := \frac{1}{2 \cdot Ae_{ie}} \cdot (a_{3ie} + b_{3ie} \cdot x + c_{3ie} \cdot y)$$

要素内の任意の点における変位 U と V は、3 個の頂点の変位 ($U1, V1, U2, V2, U3, V3$) とその要素にたいする形状関数 $N1, N2, N3$ との近似的な関係式によって与えられる。

すなわち、

$$U = N1 \cdot U1 + N2 \cdot U2 + N3 \cdot U3 \quad \text{であり、} \quad V = N1 \cdot V1 + N2 \cdot V2 + N3 \cdot V3 \quad \text{である。}$$

また、ひずみは $\varepsilon_x = \frac{\partial u}{\partial x}$ 、 $\varepsilon_y = \frac{\partial v}{\partial y}$ 、 $2\gamma = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}$ であった。

U の x による偏微分は、近似的な関係式をみると ($U1, V1, U2, V2, U3, V3$) は定数なので、

$$\frac{\partial U}{\partial x} = \frac{\partial N1}{\partial x} U1 + \frac{\partial N2}{\partial x} U2 + \frac{\partial N3}{\partial x} U3 \quad \text{となる。}$$

V、y に関しても同じことがいえるので、形状関数の x あるいは y の偏微分と (U1 , U2 , U3) または (V1 , V2 , V3) の乗数を加えることによって、要素内の点の U または V の偏微分をえることができる。

$\frac{\partial^*}{\partial^*}$ を記号 d*d* であらわすことにすると、ひずみは次の一連の手順によって求められる。 まずは、形状関数の偏微分、

$$dN1dx_{ie} := \frac{1}{(2 \cdot Ae_{ie})} \cdot b_{1ie} \quad dN1dy_{ie} := \frac{1}{(2 \cdot Ae_{ie})} \cdot c_{1ie}$$

$$dN2dx_{ie} := \frac{1}{(2 \cdot Ae_{ie})} \cdot b_{2ie} \quad dN2dy_{ie} := \frac{1}{(2 \cdot Ae_{ie})} \cdot c_{2ie}$$

$$dN3dx_{ie} := \frac{1}{(2 \cdot Ae_{ie})} \cdot b_{3ie} \quad dN3dy_{ie} := \frac{1}{(2 \cdot Ae_{ie})} \cdot c_{3ie}$$

次に U と V の偏微分を求める。

$$dUdx_{ie} := dN1dx_{ie} \cdot U_{ie,1} + dN2dx_{ie} \cdot U_{ie,2} + dN3dx_{ie} \cdot U_{ie,3}$$

$$dUdy_{ie} := dN1dy_{ie} \cdot U_{ie,1} + dN2dy_{ie} \cdot U_{ie,2} + dN3dy_{ie} \cdot U_{ie,3}$$

$$dVdx_{ie} := dN1dx_{ie} \cdot V_{ie,1} + dN2dx_{ie} \cdot V_{ie,2} + dN3dx_{ie} \cdot V_{ie,3}$$

$$dVdy_{ie} := dN1dy_{ie} \cdot V_{ie,1} + dN2dy_{ie} \cdot V_{ie,2} + dN3dy_{ie} \cdot V_{ie,3}$$

ひずみを求める。 上の各偏微分には変数 x 、 y を含んでいないので、要素内部ではひずみは位置の座標にかかわらず一定である。 すなわち一個の要素に一組の x、 y、 γ が存在する。

$$\begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma \end{bmatrix} := \begin{bmatrix} dUdx \\ dVdy \\ \frac{1}{2} \cdot (dUdy + dVdx) \end{bmatrix}$$

数値を表示するために、各行が要素番号、 x 、 y 、 から成るマトリクスを作成する。

$$\text{num}_{ie} := ie$$

$$\text{strain}^{<1>} := \text{num} \cdot 10^{-6} \quad \text{strain}^{<2>} := \varepsilon_x \quad \text{strain}^{<3>} := \varepsilon_y \quad \text{strain}^{<4>} := \gamma$$

$$\text{strain} \cdot 10^6 = \begin{bmatrix} 1 & -1.183 & -0.106 & 2.232 \\ 2 & 0.111 & -0.26 & 0.645 \\ 3 & -3.725 & 0.193 & 3.825 \\ 4 & 0.115 & -0.106 & -0.916 \\ 5 & -6.342 & 0.67 & 5.521 \\ 6 & 0.145 & 0.193 & -2.493 \\ 7 & -9.671 & 4.411 & 6.643 \\ 8 & -0.362 & 0.67 & -4.245 \\ 9 & 0.111 & -0.747 & 2.287 \\ 10 & 1.386 & -0.585 & 0.661 \\ 11 & 0.115 & -1.062 & 3.871 \\ 12 & 3.925 & -0.747 & -0.955 \\ 13 & 0.145 & -1.371 & 5.366 \\ 14 & 6.444 & -1.062 & -2.569 \\ 15 & -0.362 & 1.28 & 7.392 \\ 16 & 9.148 & -1.371 & -3.965 \end{bmatrix} \cdot$$

(4) 応力を求める

ひずみと応力には次の関係式が成立する。

$$\begin{bmatrix} \sigma_{x_{ie}} \\ \sigma_{y_{ie}} \\ \tau_{ie} \end{bmatrix} := \begin{bmatrix} c11 & c12 & 0 \\ c12 & c22 & 0 \\ 0 & 0 & c66 \end{bmatrix} \cdot \begin{bmatrix} \varepsilon_{x_{ie}} \\ \varepsilon_{y_{ie}} \\ \gamma_{ie} \end{bmatrix}$$

数値を表示するために、各行が要素番号、 x、 y、 から成るマトリクスを作成する。 応力の単位は N/cm² である。

$$\text{stress}^{<1>} := \text{num} \quad \text{stress}^{<2>} := \sigma_x \quad \text{stress}^{<3>} := \sigma_y \quad \text{stress}^{<4>} := \tau$$

$$\text{stress} = \begin{bmatrix} 1 & -26.579 & -8.818 & 18.39 \\ 2 & 1.007 & -5.112 & 5.316 \\ 3 & -80.796 & -16.222 & 31.519 \\ 4 & 1.954 & -1.685 & -7.547 \\ 5 & -135.682 & -20.124 & 45.496 \\ 6 & 4.256 & 5.041 & -20.542 \\ 7 & -188.277 & 43.794 & 54.742 \\ 8 & -4.275 & 12.727 & -34.977 \\ 9 & -1.666 & -15.801 & 18.846 \\ 10 & 27.238 & -5.242 & 5.448 \\ 11 & -3.3 & -22.704 & 31.893 \\ 12 & 82.142 & 5.151 & -7.866 \\ 13 & -4.336 & -29.328 & 44.216 \\ 14 & 135.762 & 12.061 & -21.169 \\ 15 & -0.922 & 26.139 & 60.908 \\ 16 & 193.474 & 20.125 & -32.673 \end{bmatrix}$$

(5)主応力 1、 2 と 最大せん断応力

主応力 1、 2 と最大せん断応力が次の式で定義される。

$$\begin{bmatrix} \sigma_{1_{ie}} \\ \sigma_{2_{ie}} \end{bmatrix} := \begin{bmatrix} \frac{\sigma_{x_{ie}} + \sigma_{y_{ie}}}{2} + \sqrt{\left(\frac{\sigma_{x_{ie}} - \sigma_{y_{ie}}}{2}\right)^2 + (\tau_{ie})^2} \\ \frac{\sigma_{x_{ie}} + \sigma_{y_{ie}}}{2} - \sqrt{\left(\frac{\sigma_{x_{ie}} - \sigma_{y_{ie}}}{2}\right)^2 + (\tau_{ie})^2} \end{bmatrix} \quad \tau_{\max} := \frac{\sigma_1 - \sigma_2}{2}$$

また、主応力の働く面の x 軸となす角度 は次の式であらわされる。

$$\phi_{ie} := \frac{1}{2} \cdot \text{atan} \left(\frac{2 \cdot \tau_{ie}}{\sigma_{x_{ie}} - \sigma_{y_{ie}}} \right)$$

数値を表示するために、各行が要素番号、 1、 2、 max、 から成るマトリクスを作成する。 応力の単位は N/cm² である。

$$\text{princstress}^{<2>} := \sigma_1 \quad \text{princstress}^{<3>} := \sigma_2 \quad \text{princstress}^{<4>} := \tau_{\max} \quad \text{princstress}^{<1>} := \text{num}$$

$$\text{princstress}^{<5>} := \phi \cdot \frac{180}{\pi}$$

princstress =	1	2.723	-38.121	-8.88	-32.112
	2	4.081	-8.186	3.059	30.039
	3	-3.388	-93.63	-32.287	-22.155
	4	7.898	-7.628	1.82	-38.222
	5	-4.363	-151.444	-57.779	-19.109
	6	25.194	-15.897	-0.392	44.453
	7	56.059	-200.542	-116.035	-12.628
	8	40.221	-31.769	-8.501	38.17
	9	11.395	-28.861	7.068	34.721
	10	28.128	-6.131	16.24	9.272
	11	20.334	-46.339	9.702	36.54
	12	82.937	4.355	38.496	-5.774
	13	29.115	-62.779	12.496	37.11
	14	139.285	8.539	61.85	-9.447
	15	75.001	-49.784	-13.53	-38.738
	16	199.428	14.171	86.675	-10.327

Sub Macro1()

'平面応力問題 F E M 三角形一次要素モデル

Dim nnode, ne, nrest, i, j, ind, ie, p, ir, rows

Dim x(100), y(100), top(100, 5), nodes(400, 3), restraints(100, 2), k1, k2

Dim x1, x2, x3, y1, y2, y3, aa, bb, Ae, KS(4, 4), Loads(100, 2), f(400)

Dim c11, c12, c22, c66, K11(6, 6), K12(6, 6), K21(6, 6), K22(6, 6)

Dim Kel(100, 6, 6), KG(400, 400), indp, indj

Dim Kmod(400, 400), fmod(400), pp, a(400, 400), c(400), u(400), s

Dim ux(100), vy(100), ux1, ux2, ux3, vy1, vy2, vy3, eex(100), eey(100), eet(100)

Dim sx(100), sy(100), st(100), s1(100), s2(100), stmax(100), pf(100)

'インプット (1) -----

nnode = Cells(2, 2): ne = Cells(2, 4): t = Cells(2, 6): nu = Cells(2, 9)

nrest = Cells(3, 2): nload = Cells(3, 4): E = Cells(4, 2)

'インプット (2) -----

For ind = 1 To nnode

x(ind) = Cells(ind + 6, 2): y(ind) = Cells(ind + 6, 3)

Next

For p = 1 To 3: ' for triangle

For ie = 1 To ne

top(ie, p) = Cells(ie + 6, p + 4)

Next

Next

For j = 1 To 2

For i = 1 To nrest

restraints(i, j) = Cells(i + 6, j + 8)

Next

For i = 1 To nload

Loads(i, j) = Cells(i + 6, j + 10)

Next

Next

'要素剛性マトリクス作成準備-----

c11 = E / (1 - nu ^ 2): c22 = c11: c12 = E * nu / (1 - nu ^ 2)

c66 = E / 2 / (1 + nu)

For ie = 1 To ne

x1 = x(top(ie, 1)): x2 = x(top(ie, 2)): x3 = x(top(ie, 3))

y1 = y(top(ie, 1)): y2 = y(top(ie, 2)): y3 = y(top(ie, 3))

'要素剛性マトリクスサブルーチン呼び出しと作成----

elematrix x1, x2, x3, y1, y2, y3, K11(), K12(), K21(), K22(), c11, c12, c22, c66

For p = 1 To 6

For j = 1 To 6

Kel(ie, p, j) = (K11(p, j) + K12(p, j) + K21(p, j) + K22(p, j)) * t

Next

Next

Next

'全体剛性マトリクスの組立てサブルーチン呼び出し-----

globmat ne, nnode, top(), Kel(), KG()

'外力ベクトルを荷重ベクトルに組み込むサブルーチン呼び出し---
extforce nnode, nload, Loads(), f()

'境界条件を加味した剛性マトリクス修正と荷重ベクトル作成のサブルーチン呼び出し----
modification nnode, nrest, KG(), f(), fmod(), Kmod(), restraints()

'連立一次方程式解法サブルーチンの呼び出し-----

```
rows = 2 * nnode
For j = 1 To rows
c(j) = fmod(j)
For i = 1 To rows
a(i, j) = Kmod(i, j)
Next
Next
matsol rows, a(), c(), u()
```

'u()の値を x、 y 方向の成分に分類----

```
For i = 1 To nnode
ux(i) = u(2 * i - 1): vy(i) = u(2 * i)
Next
```

```
postprocess ne, x(), y(), ux(), vy(), eex(), eey(), eet(), top()
stress ne, c11, c22, c12, c66, eex(), eey(), eet(), sx(), sy(), st()
princstress ne, sx(), sy(), st(), s1(), s2(), stmax(), pf()
```

'アウトプット----

```
For i = 1 To nnode
```

```
Cells(i + 6, 15) = ux(i)
Cells(i + 6, 16) = vy(i)
Next
```

```
For ie = 1 To ne
```

```
Cells(ie + 6, 18) = eex(ie) * 10 ^ 6: Cells(ie + 6, 19) = eey(ie) * 10 ^ 6
Cells(ie + 6, 20) = eet(ie) * 10 ^ 6
```

```
Cells(ie + 6, 22) = sx(ie): Cells(ie + 6, 23) = sy(ie)
Cells(ie + 6, 24) = st(ie)
```

```
Cells(ie + 6, 26) = s1(ie): Cells(ie + 6, 27) = s2(ie)
Cells(ie + 6, 28) = stmax(ie): Cells(ie + 6, 29) = pf(ie) * 180 / 3.1415
```

```
Next
```

```
End Sub
```

' (3) サブルーチン

```
Sub elematrix(x1, x2, x3, y1, y2, y3, K11(), K12(), K21(), K22(), c11, c12, c22, c66)
```

```
a1 = x2 * y3 - x3 * y2: b1 = y2 - y3: c1 = -(x2 - x3)
a2 = x3 * y1 - x1 * y3: b2 = y3 - y1: c2 = -(x3 - x1)
a3 = x1 * y2 - x2 * y1: b3 = y1 - y2: c3 = -(x1 - x2)
Ae = (a1 + a2 + a3) / 2
```

```

For i = 1 To 6
For j = 1 To 6
K11(i, j) = 0: K12(i, j) = 0: K21(i, j) = 0: K22(i, j) = 0
Next
Next

```

```

K11(1, 1) = c11 * b1 * b1 + c66 * c1 * c1
K11(1, 3) = c11 * b1 * b2 + c66 * c1 * c2
K11(1, 5) = c11 * b1 * b3 + c66 * c1 * c3
K11(3, 1) = c11 * b2 * b1 + c66 * c2 * c1
K11(3, 3) = c11 * b2 * b2 + c66 * c2 * c2
K11(3, 5) = c11 * b2 * b3 + c66 * c2 * c3
K11(5, 1) = c11 * b3 * b1 + c66 * c3 * c1
K11(5, 3) = c11 * b3 * b2 + c66 * c3 * c2
K11(5, 5) = c11 * b3 * b3 + c66 * c3 * c3

```

```

K22(2, 2) = c66 * b1 * b1 + c22 * c1 * c1
K22(2, 4) = c66 * b1 * b2 + c22 * c1 * c2
K22(2, 6) = c66 * b1 * b3 + c22 * c1 * c3
K22(4, 2) = c66 * b2 * b1 + c22 * c2 * c1
K22(4, 4) = c66 * b2 * b2 + c22 * c2 * c2
K22(4, 6) = c66 * b2 * b3 + c22 * c2 * c3
K22(6, 2) = c66 * b3 * b1 + c22 * c3 * c1
K22(6, 4) = c66 * b3 * b2 + c22 * c3 * c2
K22(6, 6) = c66 * b3 * b3 + c22 * c3 * c3

```

```

K12(1, 2) = c12 * b1 * c1 + c66 * c1 * b1
K12(1, 4) = c12 * b1 * c2 + c66 * c1 * b2
K12(1, 6) = c12 * b1 * c3 + c66 * c1 * b3
K12(3, 2) = c12 * b2 * c1 + c66 * c2 * b1
K12(3, 4) = c12 * b2 * c2 + c66 * c2 * b2
K12(3, 6) = c12 * b2 * c3 + c66 * c2 * b3
K12(5, 2) = c12 * b3 * c1 + c66 * c3 * b1
K12(5, 4) = c12 * b3 * c2 + c66 * c3 * b2
K12(5, 6) = c12 * b3 * c3 + c66 * c3 * b3

```

```

K21(2, 1) = c66 * b1 * c1 + c12 * c1 * b1
K21(2, 3) = c66 * b1 * c2 + c12 * c1 * b2
K21(2, 5) = c66 * b1 * c3 + c12 * c1 * b3
K21(4, 1) = c66 * b2 * c1 + c12 * c2 * b1
K21(4, 3) = c66 * b2 * c2 + c12 * c2 * b2
K21(4, 5) = c66 * b2 * c3 + c12 * c2 * b3
K21(6, 1) = c66 * b3 * c1 + c12 * c3 * b1
K21(6, 3) = c66 * b3 * c2 + c12 * c3 * b2
K21(6, 5) = c66 * b3 * c3 + c12 * c3 * b3

```

```

For i = 1 To 6
For j = 1 To 6
K11(i, j) = K11(i, j) / Ae / 4: K12(i, j) = K12(i, j) / Ae / 4
K21(i, j) = K21(i, j) / Ae / 4: K22(i, j) = K22(i, j) / Ae / 4
Next
Next

```

End Sub

Sub globmat(ne, nnode, top(), Kel(), KG())

```

For i = 1 To 2 * nnode
For j = 1 To 2 * nnode
KG(i, j) = 0
Next
Next

For ie = 1 To ne

For p = 1 To 3
For j = 1 To 3

ip = top(ie, p)
ij = top(ie, j)

KG(2 * ip - 1, 2 * ij - 1) = KG(2 * ip - 1, 2 * ij - 1) + Kel(ie, 2 * p - 1, 2 * j - 1)
KG(2 * ip, 2 * ij - 1) = KG(2 * ip, 2 * ij - 1) + Kel(ie, 2 * p, 2 * j - 1)
KG(2 * ip - 1, 2 * ij) = KG(2 * ip - 1, 2 * ij) + Kel(ie, 2 * p - 1, 2 * j)
KG(2 * ip, 2 * ij) = KG(2 * ip, 2 * ij) + Kel(ie, 2 * p, 2 * j)

Next
Next
Next

End Sub
Sub extforce(nnode, nload, Loads(), f())
For i = 1 To 2 * nnode
f(i) = 0
Next

For i = 1 To nload
f(Loads(i, 1)) = Loads(i, 2)
Next
End Sub

Sub modification(nnode, nrest, KG(), f(), fmod(), Kmod(), restraints())
For i = 1 To 2 * nnode
fmod(i) = f(i)
For j = 1 To 2 * nnode
Kmod(i, j) = KG(i, j)
Next
Next

For i = 1 To 2 * nnode
For ir = 1 To nrest
Kmod(restraints(ir, 1), i) = 0
Kmod(i, restraints(ir, 1)) = 0
Kmod(restraints(ir, 1), restraints(ir, 1)) = 1
Next
Next

For i = 1 To nrest: '-----modification of the force vector
For j = 1 To 2 * nnode
fmod(j) = fmod(j) - KG(j, restraints(i, 1)) * restraints(i, 2)
Next
Next
For ir = 1 To nrest
fmod(restraints(ir, 1)) = restraints(ir, 2)
Next

```

End Sub

Sub matsol(rows, a(), c(), u())

For j = 1 To rows - 1

For i = j + 1 To rows

pp = a(i, j) / a(j, j)

c(i) = c(i) - c(j) * pp

For ind = j To rows

a(i, ind) = a(i, ind) - a(j, ind) * pp

Next

Next

Next

u(rows) = c(rows) / a(rows, rows)

For j = rows - 1 To 1 Step -1

s = 0

For i = j + 1 To rows

s = s + a(j, i) * u(i)

Next

u(j) = (c(j) - s) / a(j, j)

Next

End Sub

Sub postprocess(ne, x(), y(), ux(), vy(), eex(), eey(), eet(), top())

For ie = 1 To ne

x1 = x(top(ie, 1)): x2 = x(top(ie, 2)): x3 = x(top(ie, 3))

y1 = y(top(ie, 1)): y2 = y(top(ie, 2)): y3 = y(top(ie, 3))

a1 = x2 * y3 - x3 * y2: b1 = y2 - y3: c1 = -(x2 - x3)

a2 = x3 * y1 - x1 * y3: b2 = y3 - y1: c2 = -(x3 - x1)

a3 = x1 * y2 - x2 * y1: b3 = y1 - y2: c3 = -(x1 - x2)

Ae = (a1 + a2 + a3) / 2

ux1 = ux(top(ie, 1)): ux2 = ux(top(ie, 2)): ux3 = ux(top(ie, 3))

vy1 = vy(top(ie, 1)): vy2 = vy(top(ie, 2)): vy3 = vy(top(ie, 3))

eex(ie) = (b1 * ux1 + b2 * ux2 + b3 * ux3) / 2 / Ae

eey(ie) = (c1 * vy1 + c2 * vy2 + c3 * vy3) / 2 / Ae

eet(ie) = c1 * ux1 + c2 * ux2 + c3 * ux3 + b1 * vy1 + b2 * vy2 + b3 * vy3

eet(ie) = 0.5 * eet(ie) / 2 / Ae

Next

End Sub

Sub stress(ne, c11, c22, c12, c66, eex(), eey(), eet(), sx(), sy(), st())

For ie = 1 To ne

sx(ie) = c11 * eex(ie) + c12 * eey(ie)

sy(ie) = c12 * eex(ie) + c22 * eey(ie)

st(ie) = c66 * eet(ie)

Next

End Sub

Sub princstress(ne, sx(), sy(), st(), s1(), s2(), stmax(), pf())

For ie = 1 To ne

stmax(ie) = 0.5 * (sx(ie) - sy(ie))

s1(ie) = 0.5 * (sx(ie) + sy(ie)) + Sqr(stmax(ie) ^ 2 + st(ie) ^ 2)

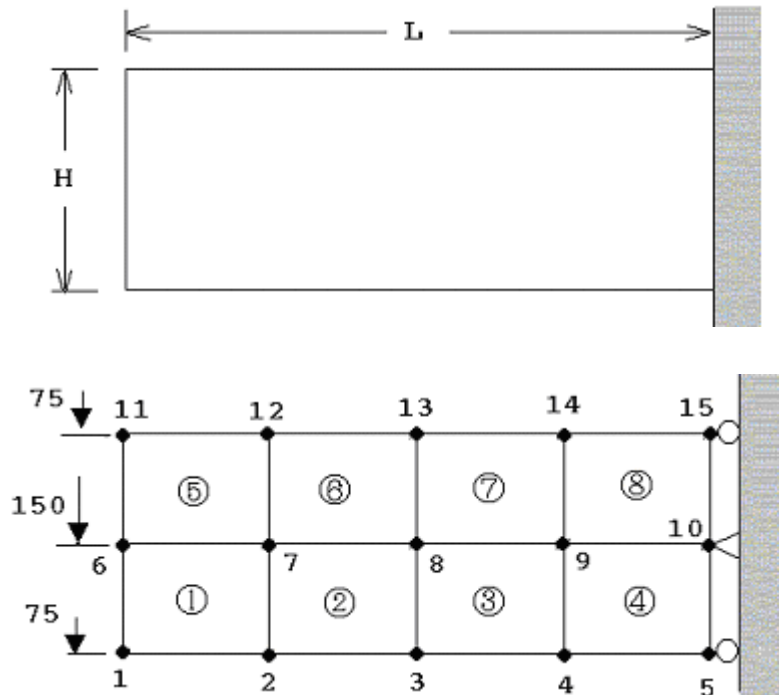
```
s2(ie) = 0.5 * (sx(ie) + sy(ie)) - Sqr(stmax(ie) ^ 2 + st(ie) ^ 2)
pf(ie) = 0.5 * Atn(2 * st(ie) / (sx(ie) - sy(ie)))
Next
```

```
End Sub
```

4.4 平面応力問題—長方形1次要素モデル

(1) 要素剛性マトリクス

長方形1次要素モデルの形状関数 N_1, N_2, N_3, N_4 はねじりの場合とまったく同じである。早速例題に入ろう。



図にしめすような板状のはりがあり、その厚さ $t = 2.5 \text{ cm}$ 、長さ $L = 25 \text{ cm}$ 、高さ $H = 5 \text{ cm}$ として、先端に $75 + 150 + 75 = 300 \text{ N}$ の荷重がかかるものとする。板状のはりを8個の要素に分割し、15個の節点を決める。各要素の4隅の節点は左下の節点から反時計まわりにかぞえる。

ではまず材料の特性値の入力から開始しよう。

$$\begin{bmatrix} E \\ t \\ \nu \end{bmatrix} := \begin{bmatrix} 2.06 \cdot 10^7 \\ 2.5 \\ .25 \end{bmatrix} \quad \begin{bmatrix} c11 & c12 & c01 \\ c21 & c22 & c02 \\ c03 & c04 & c66 \end{bmatrix} := \frac{E}{1 - \nu^2} \cdot \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1 - \nu}{2} \end{bmatrix}$$

節点の座標、トポロジーテーブル、要素数、節点数、拘束点数などを入力する。

$$\begin{bmatrix} n_{el} \\ n_{nodes} \\ n_{res} \end{bmatrix} = \begin{bmatrix} 8 \\ 15 \\ 4 \end{bmatrix} \blacksquare$$

$$ie := 1..n_{el}$$

$$x_{n_i} := Nodes_{i,2} \quad y_{n_i} := Nodes_{i,3}$$

$$Nodes = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 6.25 & 0 \\ 3 & 12.5 & 0 \\ 4 & 18.75 & 0 \\ 5 & 25 & 0 \\ 6 & 0 & 2.5 \\ 7 & 6.25 & 2.5 \\ 8 & 12.5 & 2.5 \\ 9 & 18.75 & 2.5 \\ 10 & 25 & 2.5 \\ 11 & 0 & 5 \\ 12 & 6.25 & 5 \\ 13 & 12.5 & 5 \\ 14 & 18.75 & 5 \\ 15 & 25 & 5 \end{bmatrix} \quad Top = \begin{bmatrix} 1 & 2 & 7 & 6 \\ 2 & 3 & 8 & 7 \\ 3 & 4 & 9 & 8 \\ 4 & 5 & 10 & 9 \\ 6 & 7 & 12 & 11 \\ 7 & 8 & 13 & 12 \\ 8 & 9 & 14 & 13 \\ 9 & 10 & 15 & 14 \end{bmatrix}$$

ねじりの問題で長方形要素についてまとめたとおなじように、各要素にたいするローカル節点番号 1, 2, 3, 4 にたいする座標をつぎのようにもとめ、

$$\begin{bmatrix} x_{1_{ie}} & y_{1_{ie}} \\ x_{2_{ie}} & y_{2_{ie}} \\ x_{3_{ie}} & y_{3_{ie}} \\ x_{4_{ie}} & y_{4_{ie}} \end{bmatrix} := \begin{bmatrix} x_{n(Top_{ie,1})} & y_{n(Top_{ie,1})} \\ x_{n(Top_{ie,2})} & y_{n(Top_{ie,2})} \\ x_{n(Top_{ie,3})} & y_{n(Top_{ie,3})} \\ x_{n(Top_{ie,4})} & y_{n(Top_{ie,4})} \end{bmatrix} \quad \begin{matrix} a_{ie} := |x_{1_{ie}} - x_{2_{ie}}| \\ b_{ie} := |y_{4_{ie}} - y_{1_{ie}}| \end{matrix}$$

形状関数の係数をもとめるため、まず a , b , c を算出するところもねじり問題に同じ。要素の剛性マトリクスをつくるに先だって、剛性方程式

$$\begin{pmatrix} K11 & K12 \\ K21 & K22 \end{pmatrix} \cdot \begin{pmatrix} \{u\} \\ \{v\} \end{pmatrix} = \begin{pmatrix} \{F1\} \\ \{F2\} \end{pmatrix}$$

では各ローカル節点の u と v とが $\{u\}$ と $\{v\}$ のグループに分かれていて、これらの成分 $(u_1, u_2, u_3, u_4, v_1, v_2, v_3, v_4)$ をを並べ替えて $(u_1, v_1, u_2, v_2, u_3, v_3, u_4, v_4)$ というようにしよう。そして、さらに u は奇数の $1, 3, 5, \dots$ のサブスクリプトをとるものとし、 v は偶数の $2, 4, 6, \dots$ のサブスクリプトをとるものとする。すなわち、トラスのときと同じように $(u_1, v_1, u_2, v_2, u_3, v_3, u_4, v_4)$ $(u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8)$ に置き換える。そのため 4 行 4 列のマトリクス $K11, K12, K21, K22$ を成分 0 を加えて 8 行に書き替えて、もとの成分が適当な場所にくるように配列する。配列のしかたを、とくに 0 が並んでいる様子をとくと観察していただきたい。

まず $K11$ は、

$$K_{el11}_{ie} := \frac{t \cdot c_{11} \cdot b_{ie}}{6 \cdot a_{ie}} \cdot \begin{bmatrix} 2 & 0 & -2 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -2 & 0 & 2 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 2 & 0 & -2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & -2 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} + \frac{t \cdot c_{66} \cdot a_{ie}}{6 \cdot b_{ie}} \cdot \begin{bmatrix} 2 & 0 & 1 & 0 & -1 & 0 & -2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 2 & 0 & -2 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & -2 & 0 & 2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -2 & 0 & -1 & 0 & 1 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$K22$ は、

$$K_{el22}_{ie} := \frac{t \cdot c_{66} \cdot b_{ie}}{6 \cdot a_{ie}} \cdot \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & -2 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -2 & 0 & 2 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 2 & 0 & -2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & -2 & 0 & 2 \end{bmatrix} + \frac{t \cdot c_{22} \cdot a_{ie}}{6 \cdot b_{ie}} \cdot \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 1 & 0 & -1 & 0 & -2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 2 & 0 & -2 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & -2 & 0 & 2 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -2 & 0 & -1 & 0 & 1 & 0 & 2 \end{bmatrix}$$

$K12$ は、

$$\text{Kel12}_{ie} := \frac{t \cdot c12}{4} \cdot \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & -1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & -1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} + \frac{t \cdot c66}{4} \cdot \begin{bmatrix} 0 & 1 & 0 & -1 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

K21 は、

$$\text{Kel21}_{ie} := \frac{t \cdot c66}{4} \cdot \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & -1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & -1 & 0 & -1 & 0 \end{bmatrix} + \frac{t \cdot c12}{4} \cdot \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 1 & 0 & -1 & 0 \end{bmatrix}$$

結局、要素剛性マトリクス Kel は、

$$\text{Kel}_{ie} := \text{Kel11}_{ie} + \text{Kel12}_{ie} + \text{Kel21}_{ie} + \text{Kel22}_{ie}$$

(2) 全体剛性マトリクス、境界条件作成から変位の算出まで

要素剛性マトリクス Kel が出来上がれば、後の手順はトラスの場合とほとんど同じである。ねじりの場合は一節点につき一個の変数 u 、トラスの場合は一節点につき 2 個の u 、フレームの場合は一節点につき 3 個の変数 u であった。それぞれに対して、全体剛性マトリクスの組み立ての式は若干異なっているが、平面応力の場合はトラスの場合と同じである。

$$\text{ndof}_{pn} = 2 \quad \text{ndof}_{pe} = 8$$

$$n := n_{\text{nodes}} \cdot \text{ndof}_{pn} \quad K_{n,n} := 0$$

$$ie := 1..n_{el} \quad i := 1..nn_{pe} \quad j := 1..nn_{pe} \quad p := 1..\text{ndof}_{pn} \quad q := 1..\text{ndof}_{pn}$$

$$K_{\left[\left(\text{Top}_{ie,i} - 1 \right) \cdot 2 + p, \left(\text{Top}_{ie,j} - 1 \right) \cdot 2 + q \right]} := K_{\left[\left(\text{Top}_{ie,i} - 1 \right) \cdot 2 + p, \left(\text{Top}_{ie,j} - 1 \right) \cdot 2 + q \right]} + \left(K_{el_{ie}} \right)_{(i-1) \cdot 2 + p, (j-1) \cdot 2 + q}$$

拘束条件と外力ベクトルが次のように与えられる。

$$\text{Restrains} := \begin{bmatrix} 9 & 0 \\ 19 & 0 \\ 20 & 0 \\ 29 & 0 \end{bmatrix} \quad \text{Loads} := \begin{bmatrix} 2 & -75 \\ 12 & -150 \\ 22 & -75 \end{bmatrix}$$

拘束条件と外力ベクトルによる、全体剛性マトリクスの修正と荷重ベクトルの修正は次のように行われる。

$$\text{indp} := 1..2 \cdot n_{\text{nodes}} \quad \text{indj} := 1..2 \cdot n_{\text{nodes}} \quad \text{id} := 1..n_{\text{res}}$$

$$K_{\text{mod}} := K$$

$$K_{\text{mod}}_{\text{Restrains}_{id,1}, \text{indj}} := 0$$

$$K_{\text{mod}}_{\text{indj}, \text{Restrains}_{id,1}} := 0$$

$$K_{\text{mod}}_{\left(\text{Restrains}_{id,1} \right), \text{Restrains}_{id,1}} := 1$$

$$nl := \text{rows}(\text{Loads}) \quad il := 1..nl$$

$$f_{\text{mod}}_{\text{indp}} := 0$$

$$f_{\text{mod}}_{\left(\text{Restrains}_{id,1} \right)} := \text{Restrains}_{id,2}$$

$$f_{\text{mod}}_{\left(\text{Loads}_{il,1} \right)} := f_{\text{mod}}_{\left(\text{Loads}_{il,1} \right)} + \text{Loads}_{il,2}$$

$$f_{\text{mod}}_{\text{indp}} := f_{\text{mod}}_{\text{indp}} - K_{\text{indp}, \text{Restrains}_{id,1}} \cdot \text{Restrains}_{id,2}$$

これらによって、以下のように u が求められる。 f_{mod} と u はそれぞれ全部で節点数 $\times 2 = 30$ 個あるが以下ではその中の 15 個だけが示されている。

$$u := K_{\text{mod}}^{-1} \cdot f_{\text{mod}}$$

$$u =$$

	1
1	$2.665 \cdot 10^{-4}$
2	$-1.826 \cdot 10^{-3}$
3	$2.497 \cdot 10^{-4}$
4	$-1.163 \cdot 10^{-3}$
5	$1.998 \cdot 10^{-4}$
6	$-5.838 \cdot 10^{-4}$
7	$1.167 \cdot 10^{-4}$
8	$-1.705 \cdot 10^{-4}$
9	0
10	$-8.02 \cdot 10^{-6}$
11	0
12	$-1.825 \cdot 10^{-3}$
13	0
14	$-1.161 \cdot 10^{-3}$
15	0

(3) ポストプロセッシング (Postprocessing) - ひずみを求める

計算によって各節点の変位が求められた。奇数のサフィックスが x 方向の変位 U をあらわし、偶数のサフィックスが y 方向の変位 V を表すものとする、各要素の頂点 1, 2, 3 番目にある節点の変位 U, V は次のように求められる。

$$j := 1..4 \quad U_{ie,j} := u_{2 \cdot \text{Top}_{ie,j} - 1} \quad V_{ie,j} := u_{2 \cdot \text{Top}_{ie,j}}$$

さて、長方形一次要素の形状関数は ie 番目の要素にたいして次のようにあらわされる。

$$N1(ie, x, y) := \left(1 - \frac{x}{a_{ie}}\right) \cdot \left(1 - \frac{y}{b_{ie}}\right)$$

$$N2(ie, x, y) := \frac{x}{a_{ie}} \cdot \left(1 - \frac{y}{b_{ie}}\right)$$

$$N3(ie, x, y) := \frac{x \cdot y}{a_{ie} \cdot b_{ie}}$$

$$N4(ie, x, y) := \left(1 - \frac{x}{a_{ie}}\right) \cdot \frac{y}{b_{ie}}$$

要素内の任意の点における変位 U と V は、4 個の頂点の変位 ($U1, V1, U2, V2, U3, V3, U4, V4$) とその要素にたいする形状関数 $N1, N2, N3, N4$ との近似的な関係式によって与えられる。すなわち、 $U = N1 \cdot U1 + N2 \cdot U2 + N3 \cdot U3 + N4 \cdot U4$ であり、 $V = N1 \cdot V1 + N2 \cdot V2 + N3 \cdot V3 + N4 \cdot V4$ である。

また、ひずみは $\varepsilon_x = \frac{\partial u}{\partial x}$ 、 $\varepsilon_y = \frac{\partial v}{\partial y}$ 、 $2\gamma = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}$ であった。 U の x

による偏微分は、近似的な関係式をみると ($U1, V1, U2, V2, U3, V3, U4, V4$) は定数なので、

$$\frac{\partial U}{\partial x} = \frac{\partial N1}{\partial x} U1 + \frac{\partial N2}{\partial x} U2 + \frac{\partial N3}{\partial x} U3 + \frac{\partial N4}{\partial x} U4$$

となる。V、y に関しても同じことがいえるので、形状関数の x あるいは y の偏微分と (U1, U2, U3, U4) または (V1, V2, V3, V4) の乗数を加えることによって、要素内の点の U または V の偏微分をえることができる。

$\frac{\partial^*}{\partial^*}$ を記号 d*d*であらわすことにすると、ひずみは次の一連の手順によって求められる。

まずは、形状関数の偏微分、

$$\begin{aligned} dN1x(ie, x, y) &:= \frac{-1}{a_{ie}} \cdot \left(1 - \frac{y}{b_{ie}}\right) & dN1y(ie, x, y) &:= \frac{-1}{b_{ie}} \cdot \left(1 - \frac{x}{a_{ie}}\right) \\ dN2x(ie, x, y) &:= \frac{1}{a_{ie}} \cdot \left(1 - \frac{y}{b_{ie}}\right) & dN2y(ie, x, y) &:= \frac{-x}{(a_{ie} \cdot b_{ie})} \\ dN3x(ie, x, y) &:= \frac{y}{(a_{ie} \cdot b_{ie})} & dN3y(ie, x, y) &:= \frac{x}{a_{ie} \cdot b_{ie}} \\ dN4x(ie, x, y) &:= \frac{-y}{(a_{ie} \cdot b_{ie})} & dN4y(ie, x, y) &:= \frac{1}{b_{ie}} \cdot \left(1 - \frac{x}{a_{ie}}\right) \end{aligned}$$

次に U と V の偏微分を求める。

$$\begin{aligned} dUdx(ie, x, y) &:= dN1x(ie, x, y) \cdot U_{ie,1} + dN2x(ie, x, y) \cdot U_{ie,2} + dN3x(ie, x, y) \cdot U_{ie,3} + dN4x(ie, x, y) \cdot U_{ie,4} \\ dUdy(ie, x, y) &:= dN1y(ie, x, y) \cdot U_{ie,1} + dN2y(ie, x, y) \cdot U_{ie,2} + dN3y(ie, x, y) \cdot U_{ie,3} + dN4y(ie, x, y) \cdot U_{ie,4} \\ dVdx(ie, x, y) &:= dN1x(ie, x, y) \cdot V_{ie,1} + dN2x(ie, x, y) \cdot V_{ie,2} + dN3x(ie, x, y) \cdot V_{ie,3} + dN4x(ie, x, y) \cdot V_{ie,4} \\ dVdy(ie, x, y) &:= dN1y(ie, x, y) \cdot V_{ie,1} + dN2y(ie, x, y) \cdot V_{ie,2} + dN3y(ie, x, y) \cdot V_{ie,3} + dN4y(ie, x, y) \cdot V_{ie,4} \end{aligned}$$

ひずみを求める。各偏微分には変数 x または y を含んでいるので、要素内部ではひずみは位置の座標におうじて変化する。各要素につきそれらの中央の一組の x、y、を求めることにしよう。

$$x_{c_{ie}} := \frac{x_{1_{ie}} + x_{2_{ie}}}{2} \quad y_{c_{ie}} := \frac{y_{1_{ie}} + y_{4_{ie}}}{2}$$

$$dUdx_{ie} := dUdx(ie, x_{c_{ie}}, y_{c_{ie}})$$

$$dUdy_{ie} := dUdy(ie, x_{c_{ie}}, y_{c_{ie}})$$

$$dVdx_{ie} := dVdx(ie, x_{c_{ie}}, y_{c_{ie}})$$

$$dVdy_{ie} := dVdy(ie, x_{c_{ie}}, y_{c_{ie}})$$

$$\begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma \end{bmatrix} := \begin{bmatrix} dUdx \\ dVdy \\ \frac{1}{2} \cdot (dUdy + dVdx) \end{bmatrix}$$

数値を表示するために、各行が要素番号、 x 、 y 、から成るマトリクスを作成する。

$$\text{num}_{ie} := ie \quad \text{strain}^{<1>} := \text{num} \cdot 10^{-6} \quad \text{strain}^{<2>} := \varepsilon_x \quad \text{strain}^{<3>} := \varepsilon_y \quad \text{strain}^{<4>} := \gamma$$

$$\text{strain} \cdot 10^6 = \begin{bmatrix} 1 & -1.339 & 0.406 & 1.456 \\ 2 & -3.991 & 1.761 & 11.434 \\ 3 & -6.648 & 2.58 & 34.695 \\ 4 & -9.339 & 6.584 & 71.496 \\ 5 & 4.018 & -0.406 & 1.376 \\ 6 & 11.974 & -1.761 & 11.28 \\ 7 & 19.943 & -2.58 & 34.599 \\ 8 & 28.016 & -6.584 & 71.226 \end{bmatrix} \cdot$$

(4) 応力を求める

ひずみと応力には次の関係式が成立する。

$$\begin{bmatrix} \sigma_{x_{ie}} \\ \sigma_{y_{ie}} \\ \tau_{ie} \end{bmatrix} := \begin{bmatrix} c11 & c12 & 0 \\ c12 & c22 & 0 \\ 0 & 0 & c66 \end{bmatrix} \cdot \begin{bmatrix} \varepsilon_{x_{ie}} \\ \varepsilon_{y_{ie}} \\ \gamma_{ie} \end{bmatrix}$$

数値を表示するために、各行が要素番号、 x、 y、 から成るマトリクスを作成する。 応力の単位は N/cm² である。

$$\text{stress}^{<2>} := \sigma_x \quad \text{stress}^{<3>} := \sigma_y \quad \text{stress}^{<4>} := \tau \quad \text{stress}^{<1>} := \text{num}$$

$$\text{stress} = \begin{bmatrix} 1 & -27.199 & 1.561 & 12 \\ 2 & -78.024 & 16.775 & 94.218 \\ 3 & -131.901 & 20.169 & 285.887 \\ 4 & -169.032 & 93.374 & 589.128 \\ 5 & 86.057 & 13.153 & 11.337 \\ 6 & 253.423 & 27.074 & 92.95 \\ 7 & 424.047 & 52.867 & 285.094 \\ 8 & 579.434 & 9.227 & 586.902 \end{bmatrix}$$

(5)主応力 1、 2 と 最大せん断応力

主応力 1、 2 と最大せん断応力が次の式で定義される。

$$\begin{bmatrix} \sigma_{1_{ie}} \\ \sigma_{2_{ie}} \end{bmatrix} := \begin{bmatrix} \frac{\sigma_{x_{ie}} + \sigma_{y_{ie}}}{2} + \sqrt{\left(\frac{\sigma_{x_{ie}} - \sigma_{y_{ie}}}{2}\right)^2 + (\tau_{ie})^2} \\ \frac{\sigma_{x_{ie}} + \sigma_{y_{ie}}}{2} - \sqrt{\left(\frac{\sigma_{x_{ie}} - \sigma_{y_{ie}}}{2}\right)^2 + (\tau_{ie})^2} \end{bmatrix}$$

$$\tau_{\max} := \frac{\sigma_1 - \sigma_2}{2}$$

また、大きいほうの主応力の働く面の法線と x 軸となす角度 ϕ は次の式であらわされる。

$$\phi_{ie} := \frac{1}{2} \cdot \text{atan} \left(\frac{2 \cdot \tau_{ie}}{\sigma_{x_{ie}} - \sigma_{y_{ie}}} \right)$$

数値を表示するために、各行が要素番号、 1、 2、 max、 から成るマトリクスを作成する。 応力の単位は N/cm² である。

princstress^{<2>} := σ_1 princstress^{<3>} := σ_2 princstress^{<4>} := τ_{\max} princstress^{<1>} := num

$$\text{princstress}^{\langle 5 \rangle} := \phi \cdot \frac{180}{\pi}$$

$$\text{princstress} = \begin{bmatrix} 1 & 5.91 & -31.548 & 18.729 & -19.922 \\ 2 & 74.845 & -136.094 & 105.469 & -31.647 \\ 3 & 239.959 & -351.691 & 295.825 & -37.553 \\ 4 & 565.732 & -641.39 & 603.561 & -38.722 \\ 5 & 87.779 & 11.431 & 38.174 & 8.639 \\ 6 & 286.7 & -6.203 & 146.452 & 19.698 \\ 7 & 578.636 & -101.722 & 340.179 & 28.468 \\ 8 & 946.817 & -358.156 & 652.486 & 32.045 \end{bmatrix}$$

Sub Macro1()

'平面応力問題 F E M 4角形一次要素モデル

Dim nnode, ne, nrest, i, j, ind, ie, p, ir, rows

Dim x(100), y(100), top(100, 5), nodes(400, 3), restraints(100, 2), k1, k2

Dim x1, x2, x3, x4, y1, y2, y3, y4, aa, bb, Ae, KS(4, 4), Loads(100, 2), f(400)

Dim c11, c12, c22, c66, K11(8, 8), K12(8, 8), K21(8, 8), K22(8, 8), ba, ab

Dim Kel(100, 8, 8), KG(400, 400), indp, indj

Dim Kmod(400, 400), fmod(400), pp, a(400, 400), c(400), u(400), s

Dim ux(100), vy(100), ux1, ux2, ux3, ux4, vy1, vy2, vy3, vy4

Dim eex(100), eey(100), eet(100)

Dim sx(100), sy(100), st(100), s1(100), s2(100), stmax(100), pf(100)

'インプット (1) -----

nnode = Cells(2, 2): ne = Cells(2, 4): t = Cells(2, 6): nu = Cells(2, 9)

nrest = Cells(3, 2): nload = Cells(3, 4): E = Cells(4, 2)

'インプット (2) -----

For ind = 1 To nnode

x(ind) = Cells(ind + 6, 2): y(ind) = Cells(ind + 6, 3)

Next

For p = 1 To 4: ' for rectangle

For ie = 1 To ne

top(ie, p) = Cells(ie + 6, p + 4)

Next

Next

For j = 1 To 2

For i = 1 To nrest

restraints(i, j) = Cells(i + 6, j + 8)

Next

For i = 1 To nload

Loads(i, j) = Cells(i + 6, j + 10)

Next

Next

'要素剛性マトリクス作成準備-----

c11 = E / (1 - nu ^ 2): c22 = c11: c12 = E * nu / (1 - nu ^ 2)

c66 = E / 2 / (1 + nu)

For ie = 1 To ne

x1 = x(top(ie, 1)): x2 = x(top(ie, 2)): x3 = x(top(ie, 3)): x4 = x(top(ie, 4))

y1 = y(top(ie, 1)): y2 = y(top(ie, 2)): y3 = y(top(ie, 3)): y4 = y(top(ie, 4))

aa = Abs(x2 - x1): bb = Abs(y4 - y1)

'要素剛性マトリクスサブルーチン呼び出しと作成----

elematrix aa, bb, K11(), K12(), K21(), K22(), c11, c12, c22, c66

For p = 1 To 8

For j = 1 To 8

Kel(ie, p, j) = (K11(p, j) + K12(p, j) + K21(p, j) + K22(p, j)) * t

Next

Next

Next

'全体剛性マトリクスの組立てサブルーチン呼び出し-----

globmat ne, nnode, top(), Kel(), KG()

'外力ベクトルを荷重ベクトルに組み込むサブルーチン呼び出し---
extforce nnode, nload, Loads(), f()

'境界条件を加味した剛性マトリクス修正と荷重ベクトル作成のサブルーチン呼び出し----
modification nnode, nrest, KG(), f(), fmod(), Kmod(), restraints()

'連立一次方程式解法サブルーチンの呼び出し-----

```
rows = 2 * nnode
For j = 1 To rows
c(j) = fmod(j)
For i = 1 To rows
a(i, j) = Kmod(i, j)
Next
Next
matsol rows, a(), c(), u()
```

'u()の値を x、y 方向の成分に分類----

```
For i = 1 To nnode
ux(i) = u(2 * i - 1): vy(i) = u(2 * i)
Next
```

```
postprocess ne, x(), y(), ux(), vy(), eex(), eey(), eet(), top()
stress ne, c11, c22, c12, c66, eex(), eey(), eet(), sx(), sy(), st()
princstress ne, sx(), sy(), st(), s1(), s2(), stmax(), pf()
```

'アウトプット----

```
For i = 1 To nnode
```

```
Cells(i + 6, 15) = ux(i)
Cells(i + 6, 16) = vy(i)
Next
```

```
For ie = 1 To ne
```

```
Cells(ie + 6, 18) = eex(ie) * 10 ^ 6: Cells(ie + 6, 19) = eey(ie) * 10 ^ 6
Cells(ie + 6, 20) = eet(ie) * 10 ^ 6
```

```
Cells(ie + 6, 22) = sx(ie): Cells(ie + 6, 23) = sy(ie)
Cells(ie + 6, 24) = st(ie)
```

```
Cells(ie + 6, 26) = s1(ie): Cells(ie + 6, 27) = s2(ie)
Cells(ie + 6, 28) = stmax(ie): Cells(ie + 6, 29) = pf(ie) * 180 / 3.1415
```

```
Next
End Sub
```

' (3) サブルーチン

```
Sub elematrix(aa, bb, K11(), K12(), K21(), K22(), c11, c12, c22, c66)
```

```
For i = 1 To 8
For j = 1 To 8
K11(i, j) = 0: K12(i, j) = 0: K21(i, j) = 0: K22(i, j) = 0
Next
Next
```

ba = c11 * bb / aa / 6: ab = c66 * aa / bb / 6

K11(1, 1) = 2 * ba + 2 * ab: K11(1, 3) = -2 * ba + 1 * ab
K11(1, 5) = -1 * ba - 1 * ab: K11(1, 7) = 1 * ba - 2 * ab
K11(3, 1) = -2 * ba + 1 * ab: K11(3, 3) = 2 * ba + 2 * ab
K11(3, 5) = 1 * ba - 2 * ab: K11(3, 7) = -1 * ba - 1 * ab
K11(5, 1) = -1 * ba - 1 * ab: K11(5, 3) = 1 * ba - 2 * ab
K11(5, 5) = 2 * ba + 2 * ab: K11(5, 7) = -2 * ba + 1 * ab
K11(7, 1) = 1 * ba - 2 * ab: K11(7, 3) = -1 * ba - 1 * ab
K11(7, 5) = -2 * ba + 1 * ab: K11(7, 7) = 2 * ba + 2 * ab

ba = c66 * bb / aa / 6: ab = c22 * aa / bb / 6

K22(2, 2) = 2 * ba + 2 * ab: K22(2, 4) = -2 * ba + 1 * ab
K22(2, 6) = -1 * ba - 1 * ab: K22(2, 8) = 1 * ba - 2 * ab
K22(4, 2) = -2 * ba + 1 * ab: K22(4, 4) = 2 * ba + 2 * ab
K22(4, 6) = 1 * ba - 2 * ab: K22(4, 8) = -1 * ba - 1 * ab
K22(6, 2) = -1 * ba - 1 * ab: K22(6, 4) = 1 * ba - 2 * ab
K22(6, 6) = 2 * ba + 2 * ab: K22(6, 8) = -2 * ba + 1 * ab
K22(8, 2) = 1 * ba - 2 * ab: K22(8, 4) = -1 * ba - 1 * ab
K22(8, 6) = -2 * ba + 1 * ab: K22(8, 8) = 2 * ba + 2 * ab

ba = c12 / 4: ab = c66 / 4

K12(1, 2) = 1 * ba + 1 * ab: K12(1, 4) = 1 * ba - 1 * ab
K12(1, 6) = -1 * ba - 1 * ab: K12(1, 8) = -1 * ba + 1 * ab
K12(3, 2) = -1 * ba + 1 * ab: K12(3, 4) = -1 * ba - 1 * ab
K12(3, 6) = 1 * ba - 1 * ab: K12(3, 8) = 1 * ba + 1 * ab
K12(5, 2) = -1 * ba - 1 * ab: K12(5, 4) = -1 * ba + 1 * ab
K12(5, 6) = 1 * ba + 1 * ab: K12(5, 8) = 1 * ba - 1 * ab
K12(7, 2) = 1 * ba - 1 * ab: K12(7, 4) = 1 * ba + 1 * ab
K12(7, 6) = -1 * ba + 1 * ab: K12(7, 8) = -1 * ba - 1 * ab

ba = c66 / 4: ab = c12 / 4

K21(2, 1) = 1 * ba + 1 * ab: K21(2, 3) = 1 * ba - 1 * ab
K21(2, 5) = -1 * ba - 1 * ab: K21(2, 7) = -1 * ba + 1 * ab
K21(4, 1) = -1 * ba + 1 * ab: K21(4, 3) = -1 * ba - 1 * ab
K21(4, 5) = 1 * ba - 1 * ab: K21(4, 7) = 1 * ba + 1 * ab
K21(6, 1) = -1 * ba - 1 * ab: K21(6, 3) = -1 * ba + 1 * ab
K21(6, 5) = 1 * ba + 1 * ab: K21(6, 7) = 1 * ba - 1 * ab
K21(8, 1) = 1 * ba - 1 * ab: K21(8, 3) = 1 * ba + 1 * ab
K21(8, 5) = -1 * ba + 1 * ab: K21(8, 7) = -1 * ba - 1 * ab

End Sub

Sub globmat(ne, nnode, top(), Kel(), KG())

For i = 1 To 2 * nnode

For j = 1 To 2 * nnode

KG(i, j) = 0

Next

Next

For ie = 1 To ne

For p = 1 To 4

For j = 1 To 4

```
ip = top(ie, p)
ij = top(ie, j)
```

```
KG(2 * ip - 1, 2 * ij - 1) = KG(2 * ip - 1, 2 * ij - 1) + Kel(ie, 2 * p - 1, 2 * j - 1)
KG(2 * ip, 2 * ij - 1) = KG(2 * ip, 2 * ij - 1) + Kel(ie, 2 * p, 2 * j - 1)
KG(2 * ip - 1, 2 * ij) = KG(2 * ip - 1, 2 * ij) + Kel(ie, 2 * p - 1, 2 * j)
KG(2 * ip, 2 * ij) = KG(2 * ip, 2 * ij) + Kel(ie, 2 * p, 2 * j)
```

```
Next
Next
Next
```

```
End Sub
Sub extforce(nnode, nload, Loads(), f())
For i = 1 To 2 * nnode
f(i) = 0
Next
```

```
For i = 1 To nload
f(Loads(i, 1)) = Loads(i, 2)
Next
End Sub
```

```
Sub modification(nnode, nrest, KG(), f(), fmod(), Kmod(), restraints())
For i = 1 To 2 * nnode
fmod(i) = f(i)
For j = 1 To 2 * nnode
Kmod(i, j) = KG(i, j)
Next
Next
```

```
For i = 1 To 2 * nnode
For ir = 1 To nrest
Kmod(restraints(ir, 1), i) = 0
Kmod(i, restraints(ir, 1)) = 0
Kmod(restraints(ir, 1), restraints(ir, 1)) = 1
Next
Next
For i = 1 To nrest: '-----modification of the force vector
For j = 1 To 2 * nnode
fmod(j) = fmod(j) - KG(j, restraints(i, 1)) * restraints(i, 2)
Next
Next
For ir = 1 To nrest
fmod(restraints(ir, 1)) = restraints(ir, 2)
Next
End Sub
```

```
Sub matsol(rows, a(), c(), u())
For j = 1 To rows - 1
For i = j + 1 To rows
pp = a(i, j) / a(j, j)
c(i) = c(i) - c(j) * pp
```

```
For ind = j To rows
a(i, ind) = a(i, ind) - a(j, ind) * pp
```

```

Next
Next
Next
u(rows) = c(rows) / a(rows, rows)
For j = rows - 1 To 1 Step -1
s = 0
For i = j + 1 To rows
s = s + a(j, i) * u(i)
Next
u(j) = (c(j) - s) / a(j, j)
Next
End Sub

```

```

Sub postprocess(ne, x(), y(), ux(), vy(), eex(), eey(), eet(), top())
For ie = 1 To ne
x1 = x(top(ie, 1)): x2 = x(top(ie, 2)): x3 = x(top(ie, 3)): x4 = x(top(ie, 4))
y1 = y(top(ie, 1)): y2 = y(top(ie, 2)): y3 = y(top(ie, 3)): y4 = y(top(ie, 4))
aa = Abs(x2 - x1): bb = Abs(y4 - y1)
xc = (x1 + x2) / 2: yc = (y1 + y4) / 2

```

```

dN1x = -(1 - yc / bb) / aa: dN1y = -(1 - xc / aa) / bb
dN2x = (1 - yc / bb) / aa: dN2y = -xc / aa / bb
dN3x = yc / aa / bb: dN3y = xc / aa / bb
dN4x = -yc / aa / bb: dN4y = (1 - xc / aa) / bb

```

```

ux1 = ux(top(ie, 1)): ux2 = ux(top(ie, 2)): ux3 = ux(top(ie, 3)): ux4 = ux(top(ie, 4))
vy1 = vy(top(ie, 1)): vy2 = vy(top(ie, 2)): vy3 = vy(top(ie, 3)): vy4 = vy(top(ie, 4))

```

```

eex(ie) = dN1x * ux1 + dN2x * ux2 + dN3x * ux3 + dN4x * ux4
eey(ie) = dN1y * vy1 + dN2y * vy2 + dN3y * vy3 + dN4y * vy4
eet(ie) = dN1y * ux1 + dN2y * ux2 + dN3y * ux3 + dN4y * ux4
eet(ie) = eet(ie) + dN1x * vy1 + dN2x * vy2 + dN3x * vy3 + dN4x * vy4
eet(ie) = 0.5 * eet(ie)
Next

```

```

End Sub

```

```

Sub stress(ne, c11, c22, c12, c66, eex(), eey(), eet(), sx(), sy(), st())

```

```

For ie = 1 To ne
sx(ie) = c11 * eex(ie) + c12 * eey(ie)
sy(ie) = c12 * eex(ie) + c22 * eey(ie)
st(ie) = c66 * eet(ie)
Next

```

```

End Sub

```

```

Sub princstress(ne, sx(), sy(), st(), s1(), s2(), stmax(), pf())
For ie = 1 To ne
stmax(ie) = 0.5 * (sx(ie) - sy(ie))
s1(ie) = 0.5 * (sx(ie) + sy(ie)) + Sqr(stmax(ie) ^ 2 + st(ie) ^ 2)
s2(ie) = 0.5 * (sx(ie) + sy(ie)) - Sqr(stmax(ie) ^ 2 + st(ie) ^ 2)
pf(ie) = 0.5 * Atn(2 * st(ie) / (sx(ie) - sy(ie)))
Next

```

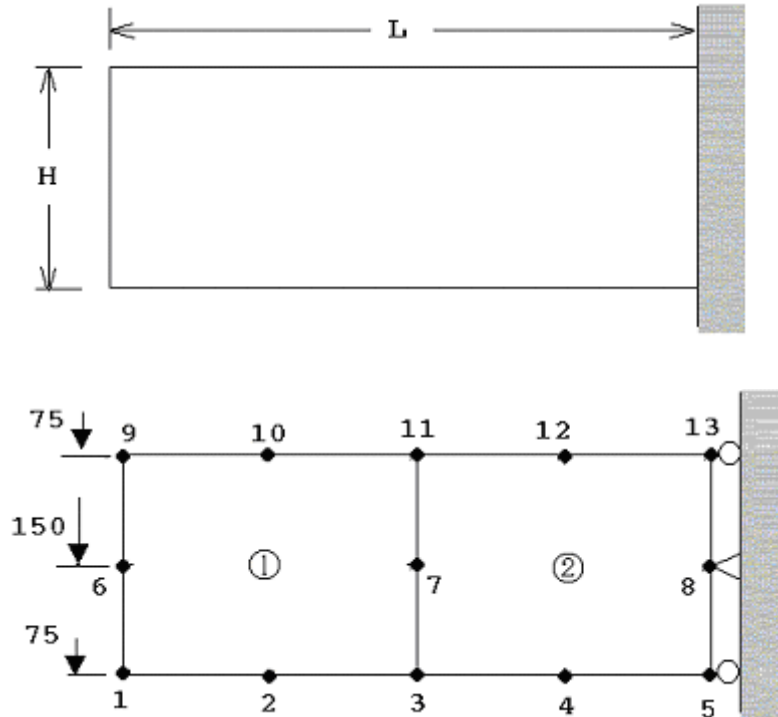
```

End Sub

```

4.6 平面弾性問題—写像を使った四角2次要素モデル

四角2次要素モデルは複雑な形状の問題を、一次要素モデルよりもはるかに少ない要素数で、よい精度で計算することができる。 三角形、長方形一次モデルで用いた例でスタートする。



まず、形状関数を定義する。 これは2次のセレンデピテイ関数と呼ばれる。

$$N(\xi, \eta) \equiv \frac{1}{4} \begin{bmatrix} -(-1+\xi) \cdot (-1+\eta) \cdot (\xi+\eta+1) \\ 2 \cdot (-1+\xi^2) \cdot (-1+\eta) \\ (1+\xi) \cdot (-1+\eta) \cdot (-\xi+\eta+1) \\ -2 \cdot (1+\xi) \cdot (-1+\eta^2) \\ (1+\xi) \cdot (1+\eta) \cdot (\xi+\eta-1) \\ -2 \cdot (-1+\xi^2) \cdot (1+\eta) \\ -(-1+\xi) \cdot (1+\eta) \cdot (-\xi+\eta-1) \\ 2 \cdot (-1+\xi) \cdot (-1+\eta^2) \end{bmatrix}$$

つぎに $\frac{\partial N}{\partial \xi}$ と $\frac{\partial N}{\partial \eta}$ を上から求め、関数の形で定義しておこう。 記号の都合で $\frac{\partial N}{\partial \xi}$

と $\frac{\partial N}{\partial \eta}$ は dNd 、 dNd と表すので了承されたい。

$$dNd\xi(\xi, \eta) := \frac{1}{4} \begin{bmatrix} 2 \cdot \xi - 2 \cdot \eta \cdot \xi - \eta^2 + \eta \\ 4 \cdot \xi \cdot (-1 + \eta) \\ 2 \cdot \xi - 2 \cdot \eta \cdot \xi + \eta^2 - \eta \\ 2 - 2 \cdot \eta^2 \\ 2 \cdot \xi + 2 \cdot \eta \cdot \xi + \eta^2 + \eta \\ -4 \cdot \xi \cdot (1 + \eta) \\ 2 \cdot \xi + 2 \cdot \eta \cdot \xi - \eta^2 - \eta \\ -2 + 2 \cdot \eta^2 \end{bmatrix} \quad dNd\eta(\xi, \eta) := \frac{1}{4} \begin{bmatrix} 2 \cdot \eta - \xi^2 - 2 \cdot \eta \cdot \xi + \xi \\ -2 + 2 \cdot \xi^2 \\ 2 \cdot \eta - \xi^2 + 2 \cdot \eta \cdot \xi - \xi \\ -4 \cdot (1 + \xi) \cdot \eta \\ 2 \cdot \eta + \xi^2 + 2 \cdot \eta \cdot \xi + \xi \\ 2 - 2 \cdot \xi^2 \\ 2 \cdot \eta + \xi^2 - 2 \cdot \eta \cdot \xi - \xi \\ 4 \cdot (-1 + \xi) \cdot \eta \end{bmatrix}$$

これで、これらの関数は後の演算の時に自動的に呼び出されることになる。

さて、入力を始めよう。

$$\begin{bmatrix} E \\ t \\ v \end{bmatrix} := \begin{bmatrix} 2.06 \cdot 10^7 \\ 2.5 \\ .25 \end{bmatrix} \quad \begin{bmatrix} c11 & c12 & c01 \\ c21 & c22 & c02 \\ c03 & c04 & c66 \end{bmatrix} := \frac{E}{1 - v^2} \begin{bmatrix} 1 & v & 0 \\ v & 1 & 0 \\ 0 & 0 & \frac{1 - v}{2} \end{bmatrix}$$

節点の座標、トポロジーテーブル、要素数、節点数、拘束点数などを入力する。

$$\text{Nodes} := \begin{bmatrix} 1 & 0 & 0 \\ 2 & 6.25 & 0 \\ 3 & 12.5 & 0 \\ 4 & 18.75 & 0 \\ 5 & 25 & 0 \\ 6 & 0 & 2.5 \\ 7 & 12.5 & 2.5 \\ 8 & 25 & 2.5 \\ 9 & 0 & 5 \\ 10 & 6.25 & 5 \\ 11 & 12.5 & 5 \\ 12 & 18.75 & 5 \\ 13 & 25 & 5 \end{bmatrix} \quad \text{Top} := \begin{bmatrix} 1 & 2 & 3 & 7 & 11 & 10 & 9 & 6 \\ 3 & 4 & 5 & 8 & 13 & 12 & 11 & 7 \end{bmatrix}$$

$$\begin{bmatrix} n_{el} \\ n_{nodes} \\ n_{res} \end{bmatrix} = \begin{bmatrix} 2 \\ 13 \\ 4 \end{bmatrix} \quad ie := 1..n_{el}$$

x と y はつぎの式によって に変換されるが、式中の x1、x2、--- x8 と y1、y2 --- y8 は与えられた座標の数値であるから、これらは定数である。

$$x = N1(\xi, \eta)x1 + N2(\xi, \eta)x2 + \dots + N8(\xi, \eta)x8$$

$$y = N1(\xi, \eta)y1 + N2(\xi, \eta)y2 + \dots + N8(\xi, \eta)y8$$

J (ヤコビアン) を求めるための $\frac{\partial x}{\partial \xi}$ 、 $\frac{\partial x}{\partial \eta}$ 、 $\frac{\partial y}{\partial \xi}$ 、 $\frac{\partial y}{\partial \eta}$ は上の式で N1, N2 --- N8

を、 で偏微分すればえられる。 結局それらは、

$$dx d\xi(\xi, \eta, ie) := dN d\xi(\xi, \eta) \cdot x_n^{<ie>}$$

$$dy d\xi(\xi, \eta, ie) := dN d\xi(\xi, \eta) \cdot y_n^{<ie>}$$

$$dx d\eta(\xi, \eta, ie) := dN d\eta(\xi, \eta) \cdot x_n^{<ie>}$$

$$dy d\eta(\xi, \eta, ie) := dN d\eta(\xi, \eta) \cdot y_n^{<ie>}$$

J (ヤコビアン) は約束にしたがって、

$$J(\xi, \eta, ie) := \begin{bmatrix} dx d\xi(\xi, \eta, ie) & dy d\xi(\xi, \eta, ie) \\ dx d\eta(\xi, \eta, ie) & dy d\eta(\xi, \eta, ie) \end{bmatrix}$$

J (ヤコビアン) の逆行列の成分をつかって $\frac{\partial N}{\partial \xi}$ と $\frac{\partial N}{\partial \eta}$ を計算する。 これらは、

$$DN dx(\xi, \eta, ie) := (J(\xi, \eta, ie)^{-1})_{1,1} \cdot dN d\xi(\xi, \eta) + (J(\xi, \eta, ie)^{-1})_{1,2} \cdot dN d\eta(\xi, \eta)$$

$$DN dy(\xi, \eta, ie) := (J(\xi, \eta, ie)^{-1})_{2,1} \cdot dN d\xi(\xi, \eta) + (J(\xi, \eta, ie)^{-1})_{2,2} \cdot dN d\eta(\xi, \eta)$$

これで要素剛性マトリクスの成分を求めるための面積分の中身がきまった。

$$\text{MakeK11}(\text{DNdx}, \text{DNdy}) := t \cdot (c11 \cdot \text{DNdx} \cdot \text{DNdx}^T + c66 \cdot \text{DNdy} \cdot \text{DNdy}^T)$$

$$\text{MakeK12}(\text{DNdx}, \text{DNdy}) := t \cdot (c12 \cdot \text{DNdx} \cdot \text{DNdy}^T + c66 \cdot \text{DNdy} \cdot \text{DNdx}^T)$$

$$\text{MakeK21}(\text{DNdx}, \text{DNdy}) := t \cdot (c21 \cdot \text{DNdy} \cdot \text{DNdx}^T + c66 \cdot \text{DNdx} \cdot \text{DNdy}^T)$$

$$\text{MakeK22}(\text{DNdx}, \text{DNdy}) := t \cdot (c66 \cdot \text{DNdx} \cdot \text{DNdx}^T + c22 \cdot \text{DNdy} \cdot \text{DNdy}^T)$$

$$\text{K11}(\xi, \eta, \text{ie}) := \text{MakeK11}(\text{DNdx}(\xi, \eta, \text{ie}), \text{DNdy}(\xi, \eta, \text{ie})) \cdot \text{J}(\xi, \eta, \text{ie})$$

$$\text{K12}(\xi, \eta, \text{ie}) := \text{MakeK12}(\text{DNdx}(\xi, \eta, \text{ie}), \text{DNdy}(\xi, \eta, \text{ie})) \cdot \text{J}(\xi, \eta, \text{ie})$$

$$\text{K21}(\xi, \eta, \text{ie}) := \text{MakeK21}(\text{DNdx}(\xi, \eta, \text{ie}), \text{DNdy}(\xi, \eta, \text{ie})) \cdot \text{J}(\xi, \eta, \text{ie})$$

$$\text{K22}(\xi, \eta, \text{ie}) := \text{MakeK22}(\text{DNdx}(\xi, \eta, \text{ie}), \text{DNdy}(\xi, \eta, \text{ie})) \cdot \text{J}(\xi, \eta, \text{ie})$$

いよいよ Gauss の近似積分に取り掛かりよう。まずは Gauss ポイント gp の座標の数値と重み係数 w を与える。

$$\text{gp} \equiv \begin{bmatrix} -.774596669241483 \\ 0 \\ .774596669241483 \end{bmatrix} \quad \text{w} \equiv \begin{bmatrix} \frac{5}{9} \\ \frac{8}{9} \\ \frac{5}{9} \end{bmatrix}$$

まずは、要素剛性マトリクスの成分、

$$\begin{aligned} i &:= 1..3 & j &:= 1..3 & \mathbf{W}_{i,j} &:= w_i \cdot w_j \\ \text{Kel11}_{\text{ie}} &:= \left(\sum_i \sum_j \mathbf{W}_{i,j} \cdot \text{K11}(\text{gp}_i, \text{gp}_j, \text{ie}) \right) \\ \text{Kel12}_{\text{ie}} &:= \left(\sum_i \sum_j \mathbf{W}_{i,j} \cdot \text{K12}(\text{gp}_i, \text{gp}_j, \text{ie}) \right) \\ \text{Kel21}_{\text{ie}} &:= \left(\sum_i \sum_j \mathbf{W}_{i,j} \cdot \text{K21}(\text{gp}_i, \text{gp}_j, \text{ie}) \right) \\ \text{Kel22}_{\text{ie}} &:= \left(\sum_i \sum_j \mathbf{W}_{i,j} \cdot \text{K22}(\text{gp}_i, \text{gp}_j, \text{ie}) \right) \end{aligned}$$

全体の剛性マトリクスを組立てる。

$$\begin{aligned}
ie &:= 1..n_{el} \quad n := n_{nodes} \cdot 2 \quad K_{n,n} := 0 \\
i &:= 1..8 \quad j := 1..8 \\
K[(Top_{ie,i} - 1) \cdot 2 + 1], (Top_{ie,j} - 1) \cdot 2 + 1 &:= K[(Top_{ie,i} - 1) \cdot 2 + 1], (Top_{ie,j} - 1) \cdot 2 + 1 + (Kel11_{ie})_{i,j} \\
K[(Top_{ie,i} - 1) \cdot 2 + 1], (Top_{ie,j} - 1) \cdot 2 + 2 &:= K[(Top_{ie,i} - 1) \cdot 2 + 1], (Top_{ie,j} - 1) \cdot 2 + 2 + (Kel12_{ie})_{i,j} \\
K[(Top_{ie,i} - 1) \cdot 2 + 2], (Top_{ie,j} - 1) \cdot 2 + 1 &:= K[(Top_{ie,i} - 1) \cdot 2 + 2], (Top_{ie,j} - 1) \cdot 2 + 1 + (Kel21_{ie})_{i,j} \\
K[(Top_{ie,i} - 1) \cdot 2 + 2], (Top_{ie,j} - 1) \cdot 2 + 2 &:= K[(Top_{ie,i} - 1) \cdot 2 + 2], (Top_{ie,j} - 1) \cdot 2 + 2 + (Kel22_{ie})_{i,j}
\end{aligned}$$

拘束条件と外力ベクトルが次のように与えられる。

$$\text{Restrains} := \begin{bmatrix} 9 & 0 \\ 15 & 0 \\ 16 & 0 \\ 25 & 0 \end{bmatrix} \quad \text{Loads} := \begin{bmatrix} 2 & -75 \\ 12 & -150 \\ 18 & -75 \end{bmatrix}$$

拘束条件と外力ベクトルによる、全体剛性マトリクスの修正と荷重ベクトルの修正は次のように行われる。

$$\begin{aligned}
p &:= 1..n_{pe} \quad n_{pe} = 8 \\
indp &:= 1..2 \cdot n_{nodes} \quad indj := 1..2 \cdot n_{nodes} \quad id := 1..n_{res} \\
K_{mod} &:= K \\
K_{mod}_{\text{Restrains}_{id,1}, indj} &:= 0 \\
K_{mod}_{indj, \text{Restrains}_{id,1}} &:= 0 \\
K_{mod}_{(\text{Restrains}_{id,1}), \text{Restrains}_{id,1}} &:= 1 \\
nl &:= \text{rows}(\text{Loads}) \quad il := 1..nl \\
f_{mod}_{indp} &:= 0 \\
f_{mod}_{(\text{Loads}_{il,1})} &:= f_{mod}_{(\text{Loads}_{il,1})} + \text{Loads}_{il,2} \\
f_{mod}_{indp} &:= f_{mod}_{indp} - K_{indp, \text{Restrains}_{id,1}} \cdot \text{Restrains}_{id,2} \\
f_{mod}_{(\text{Restrains}_{id,1})} &:= \text{Restrains}_{id,2}
\end{aligned}$$

これらによって、以下のように u が求められる。 u は全部で節点数 $x \cdot 2 = 26$ 個

あるが以下ではその中の 13 個だけが示されている。

$$u := K_{\text{mod}}^{-1} \cdot f_{\text{mod}}$$

u =

	1
1	$4.374 \cdot 10^{-4}$
2	$-2.931 \cdot 10^{-3}$
3	$4.008 \cdot 10^{-4}$
4	$-1.859 \cdot 10^{-3}$
5	$3.275 \cdot 10^{-4}$
6	$-9.244 \cdot 10^{-4}$
7	$1.824 \cdot 10^{-4}$
8	$-2.63 \cdot 10^{-4}$
9	0
10	$-1.098 \cdot 10^{-5}$
11	0
12	$-2.931 \cdot 10^{-3}$

これで、写像を使った平面弾性四角 2 次要素モデルが出来上がった。これらの筋書きをアレンジして Excel VBA のプログラムを作成して見よう。

ポストプロセッシング(Postprocessing) - ひずみを求める

計算によって各節点の変位が求められた。奇数のサブスクリプトが x 方向の変位 U をあらわし、偶数のサブスクリプトが y 方向の変位 V を表すものとする、各要素の頂点 1, 2, 3, 4, 5, 6, 7, 8 番目にある節点の変位 U、V は次のように求められる。

$$p := 1..8 \quad U_{p,ie} := u_{2 \cdot \text{Top}_{ie,p} - 1} \quad V_{p,ie} := u_{2 \cdot \text{Top}_{ie,p}}$$

さて、四角形 2 次要素の形状関数は写像によって 次の関数であらわされ、これはどの要素にも共通である。

$$N(\xi, \eta) \equiv \frac{1}{4} \begin{bmatrix} -(-1+\xi) \cdot (-1+\eta) \cdot (\xi+\eta+1) \\ 2 \cdot (-1+\xi^2) \cdot (-1+\eta) \\ (1+\xi) \cdot (-1+\eta) \cdot (-\xi+\eta+1) \\ -2 \cdot (1+\xi) \cdot (-1+\eta^2) \\ (1+\xi) \cdot (1+\eta) \cdot (\xi+\eta-1) \\ -2 \cdot (-1+\xi^2) \cdot (1+\eta) \\ -(-1+\xi) \cdot (1+\eta) \cdot (-\xi+\eta-1) \\ 2 \cdot (-1+\xi) \cdot (-1+\eta^2) \end{bmatrix}$$

要素内の任意の点における変位 U と V は、8 個の節点の変位 (U1, V1, U2, V2, U3, V3, --- U8, V8) とその要素にたいする形状関数 N1, N2, N3, --- N8 との近似的な関係式によって与えられる。すなわち、

$$U = N1 \cdot U1 + N2 \cdot U2 + \dots + N8 \cdot U8$$

$$V = N1 \cdot V1 + N2 \cdot V2 + \dots + N8 \cdot V8$$

また、ひずみは

$$\varepsilon_x = \frac{\partial u}{\partial x} \quad , \quad \varepsilon_y = \frac{\partial v}{\partial y} \quad , \quad 2\gamma = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}$$

U の x による偏微分は、近似的な関係式をみると (U1, V1, U2, V2, U3,

$V3, \dots, U8, V8$) は定数なので、

$$\frac{\partial U}{\partial x} = \frac{\partial N1}{\partial x}U1 + \frac{\partial N2}{\partial x}U2 + \frac{\partial N3}{\partial x}U3 \dots + \frac{\partial N8}{\partial x}U8$$

V, y についても同じことがいえるので、形状関数の x あるいは y の偏微分と ($U1, U2, U3, \dots, U8$) または ($V1, V2, V3, \dots, V8$) の乗数を加えることによって、要素内の点の U または V の偏微分をえることができる。ひずみは次の一連の手順によって求められる。

まずは、 ξ であらわされた形状関数の偏微分、

$$dNd\xi(\xi, \eta) := \frac{1}{4} \begin{bmatrix} 2 \cdot \xi - 2 \cdot \eta \cdot \xi - \eta^2 + \eta \\ 4 \cdot \xi \cdot (-1 + \eta) \\ 2 \cdot \xi - 2 \cdot \eta \cdot \xi + \eta^2 - \eta \\ 2 - 2 \cdot \eta^2 \\ 2 \cdot \xi + 2 \cdot \eta \cdot \xi + \eta^2 + \eta \\ -4 \cdot \xi \cdot (1 + \eta) \\ 2 \cdot \xi + 2 \cdot \eta \cdot \xi - \eta^2 - \eta \\ -2 + 2 \cdot \eta^2 \end{bmatrix} \quad dNd\eta(\xi, \eta) := \frac{1}{4} \begin{bmatrix} 2 \cdot \eta - \xi^2 - 2 \cdot \eta \cdot \xi + \xi \\ -2 + 2 \cdot \xi^2 \\ 2 \cdot \eta - \xi^2 + 2 \cdot \eta \cdot \xi - \xi \\ -4 \cdot (1 + \xi) \cdot \eta \\ 2 \cdot \eta + \xi^2 + 2 \cdot \eta \cdot \xi + \xi \\ 2 - 2 \cdot \xi^2 \\ 2 \cdot \eta + \xi^2 - 2 \cdot \eta \cdot \xi - \xi \\ 4 \cdot (-1 + \xi) \cdot \eta \end{bmatrix}$$

から、次の一連の操作によって、節点における $\frac{\partial^*}{\partial x}$ と $\frac{\partial^*}{\partial y}$ を導く。ひずみ、応力を求める場所を各要素のもつ節点 1, 2, 3, ..., 8 としよう。これは、 s_ξ, s_η の次の点 (s_ξ, s_η) に対応する。

$$s_\xi := \begin{bmatrix} -1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ -1 \\ -1 \end{bmatrix} \quad s_\eta := \begin{bmatrix} -1 \\ -1 \\ -1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

この点に対応する、ヤコビアンをつぎのように算出し、

$$dx d\xi(\xi, \eta, ie) := dNd\xi(\xi, \eta) \cdot x_n^{<ie>}$$

$$dy d\xi(\xi, \eta, ie) := dNd\xi(\xi, \eta) \cdot y_n^{<ie>}$$

$$dx d\eta(\xi, \eta, ie) := dNd\eta(\xi, \eta) \cdot x_n^{<ie>}$$

$$dy d\eta(\xi, \eta, ie) := dNd\eta(\xi, \eta) \cdot y_n^{<ie>}$$

$$J(\xi, \eta, ie) := \begin{bmatrix} dx d\xi(\xi, \eta, ie) & dy d\xi(\xi, \eta, ie) \\ dx d\eta(\xi, \eta, ie) & dy d\eta(\xi, \eta, ie) \end{bmatrix}$$

逆ヤコビアンによって、 x 、 y にたいする N の偏微分を求める。

$$DNdx(\xi, \eta, ie) := \left(J(\xi, \eta, ie)^{-1} \right)_{1,1} \cdot dNd\xi(\xi, \eta) + \left(J(\xi, \eta, ie)^{-1} \right)_{1,2} \cdot dNd\eta(\xi, \eta)$$

$$DNdy(\xi, \eta, ie) := \left(J(\xi, \eta, ie)^{-1} \right)_{2,1} \cdot dNd\xi(\xi, \eta) + \left(J(\xi, \eta, ie)^{-1} \right)_{2,2} \cdot dNd\eta(\xi, \eta)$$

次に U と V の偏微分を求める。

$$dUdx(\xi, \eta, ie) := DNdx(\xi, \eta, ie) \cdot U^{<ie>}$$

$$dUdy(\xi, \eta, ie) := DNdy(\xi, \eta, ie) \cdot U^{<ie>}$$

$$dVdx(\xi, \eta, ie) := DNdx(\xi, \eta, ie) \cdot V^{<ie>}$$

$$dVdy(\xi, \eta, ie) := DNdy(\xi, \eta, ie) \cdot V^{<ie>}$$

ここまでくれば、ひずみが求まる。 各要素の節点ごとに一組の x 、 y 、
を求めることにしよう。

$$\begin{bmatrix} \varepsilon_{x_{p,ie}} \\ \varepsilon_{y_{p,ie}} \\ \gamma_{p,ie} \end{bmatrix} := \begin{bmatrix} dUdx(s\xi_p, s\eta_p, ie) \\ dVdy(s\xi_p, s\eta_p, ie) \\ \frac{1}{2} \cdot \left[\left(dVdx(s\xi_p, s\eta_p, ie) \right) + dUdy(s\xi_p, s\eta_p, ie) \right] \end{bmatrix}$$

数値を表示する。 一列目が要素 1、2 列目が要素 2 に相当し、行がローカル
節点番号 1, 2, 3、--- 8 に相当して、 x 、 y 、 のマトリクスを表示す
る。

$$\varepsilon_x = \begin{bmatrix} -2.916 \cdot 10^{-6} & -2.023 \cdot 10^{-5} \\ -8.797 \cdot 10^{-6} & -2.62 \cdot 10^{-5} \\ -1.468 \cdot 10^{-5} & -3.217 \cdot 10^{-5} \\ 0 & 0 \\ 1.468 \cdot 10^{-5} & 3.217 \cdot 10^{-5} \\ 8.797 \cdot 10^{-6} & 2.62 \cdot 10^{-5} \\ 2.916 \cdot 10^{-6} & 2.023 \cdot 10^{-5} \\ 0 & 0 \end{bmatrix} \quad \varepsilon_y = \begin{bmatrix} 6.368 \cdot 10^{-7} & 4.203 \cdot 10^{-6} \\ 2.42 \cdot 10^{-6} & 6.493 \cdot 10^{-6} \\ 4.203 \cdot 10^{-6} & 8.783 \cdot 10^{-6} \\ 0 & 0 \\ -4.203 \cdot 10^{-6} & -8.783 \cdot 10^{-6} \\ -2.42 \cdot 10^{-6} & -6.493 \cdot 10^{-6} \\ -6.368 \cdot 10^{-7} & -4.203 \cdot 10^{-6} \\ 0 & 0 \end{bmatrix} \quad \gamma = \begin{bmatrix} 3.788 \cdot 10^{-6} & 3.79 \cdot 10^{-6} \\ 1.123 \cdot 10^{-7} & 6.049 \cdot 10^{-8} \\ 3.788 \cdot 10^{-6} & 3.79 \cdot 10^{-6} \\ 3.966 \cdot 10^{-6} & 4.019 \cdot 10^{-6} \\ 3.788 \cdot 10^{-6} & 3.79 \cdot 10^{-6} \\ 1.123 \cdot 10^{-7} & 6.049 \cdot 10^{-8} \\ 3.788 \cdot 10^{-6} & 3.79 \cdot 10^{-6} \\ 3.966 \cdot 10^{-6} & 4.019 \cdot 10^{-6} \end{bmatrix}$$

応力を求める

ひずみと応力には次の関係式が成立する。

$$\begin{bmatrix} \sigma_{x_{p,ie}} \\ \sigma_{y_{p,ie}} \\ \tau_{p,ie} \end{bmatrix} := \begin{bmatrix} c_{11} & c_{12} & 0 \\ c_{12} & c_{22} & 0 \\ 0 & 0 & c_{66} \end{bmatrix} \cdot \begin{bmatrix} \varepsilon_{x_{p,ie}} \\ \varepsilon_{y_{p,ie}} \\ \gamma_{p,ie} \end{bmatrix}$$

数値を表示するために、各列が各要素で、各行がローカル節点番号順にならぶ、

x、 y、 を行列で表示する。 応力の単位は N/cm² である。

$$\sigma_x = \begin{bmatrix} -60.573 & -421.462 \\ -180 & -540 \\ -299.427 & -658.538 \\ 2.697 \cdot 10^{-12} & 1.027 \cdot 10^{-12} \\ 299.427 & 658.538 \\ 180 & 540 \\ 60.573 & 421.462 \\ 2.297 \cdot 10^{-12} & 1.037 \cdot 10^{-13} \end{bmatrix} \quad \sigma_y = \begin{bmatrix} -2.025 & -18.787 \\ 4.848 & -1.248 \\ 11.721 & 16.292 \\ -1.994 \cdot 10^{-12} & -3.792 \cdot 10^{-13} \\ -11.721 & -16.292 \\ -4.848 & 1.248 \\ 2.025 & 18.787 \\ 1.733 \cdot 10^{-11} & -2.642 \cdot 10^{-12} \end{bmatrix} \quad \tau = \begin{bmatrix} 31.211 & 31.229 \\ 0.925 & 0.498 \\ 31.211 & 31.229 \\ 32.681 & 33.116 \\ 31.211 & 31.229 \\ 0.925 & 0.498 \\ 31.211 & 31.229 \\ 32.681 & 33.116 \end{bmatrix}$$

主応力 1、 2 と最大せん断応力

主応力 1、 2 と最大せん断応力が次の式で定義される。

$$\begin{bmatrix} \sigma_{1,p,ie} \\ \sigma_{2,p,ie} \\ \tau_{\max,p,ie} \end{bmatrix} := \begin{bmatrix} \frac{\sigma_{x,p,ie} + \sigma_{y,p,ie}}{2} + \sqrt{\left(\frac{\sigma_{x,p,ie} - \sigma_{y,p,ie}}{2}\right)^2 + (\tau_{p,ie})^2} \\ \frac{\sigma_{x,p,ie} + \sigma_{y,p,ie}}{2} - \sqrt{\left(\frac{\sigma_{x,p,ie} - \sigma_{y,p,ie}}{2}\right)^2 + (\tau_{p,ie})^2} \\ \sqrt{\left(\frac{\sigma_{x,p,ie} - \sigma_{y,p,ie}}{2}\right)^2 + (\tau_{p,ie})^2} \end{bmatrix}$$

各要素を列としローカル節点番号順に下る 1、2、max の行列を表示する。
 応力の単位は N/cm² である。

$$\sigma_1 = \begin{bmatrix} 11.492 & -16.38 \\ 4.852 & -1.247 \\ 14.821 & 17.734 \\ 32.681 & 33.116 \\ 302.527 & 659.98 \\ 180.005 & 540 \\ 74.091 & 423.869 \\ 32.681 & 33.116 \end{bmatrix} \quad \sigma_2 = \begin{bmatrix} -74.091 & -423.869 \\ -180.005 & -540 \\ -302.527 & -659.98 \\ -32.681 & -33.116 \\ -14.821 & -17.734 \\ -4.852 & 1.247 \\ -11.492 & 16.38 \\ -32.681 & -33.116 \end{bmatrix} \quad \tau_{\max} = \begin{bmatrix} 42.791 & 203.745 \\ 92.429 & 269.377 \\ 158.674 & 338.857 \\ 32.681 & 33.116 \\ 158.674 & 338.857 \\ 92.429 & 269.377 \\ 42.791 & 203.745 \\ 32.681 & 33.116 \end{bmatrix}$$

Sub Macro1()

'平面応力問題 F E M 四角形 2 次要素モデル

Dim nnode, ne, nrest, i, j, ind, ie, p, ir, rows
Dim x(100), y(100), top(100, 8), nodes(400, 3), restraints(100, 2), k1, k2
Dim Loads(100, 2), f(400)
Dim c11, c12, c22, c66, t
Dim K11(100, 8, 8), K12(100, 8, 8), K21(100, 8, 8), K22(100, 8, 8)
Dim KG(400, 400), indp, indj
Dim Kmod(400, 400), fmod(400), pp, a(400, 400), c(400), u(400), s
Dim gpx(9), gpy(9), ww(9), w(3)
Dim ex, ey, N(8), DNx(8), DNy(8), xx(8), yy(8), xxx(9, 8), yyy(9, 8)
Dim DNex(8), DNey(8)
Dim dexx(8), dexy(8), deyy(8), deyx(8), dxex(8), dxey(8), dyex(8), dyey(8)
Dim ja11, ja12, ja21, ja22, detj, invj11, invj12, invj21, invj22
Dim ux(100), vy(100), uu(8), vv(8)
Dim sex(8), sey(8), eex(100, 8), eey(100, 8), eet(100, 8)
Dim sx(100, 8), sy(100, 8), st(100, 8), s1(100, 8), s2(100, 8), stmax(100, 8)

'インプット (1) -----

nnode = Cells(2, 2): ne = Cells(2, 4): t = Cells(2, 6): nu = Cells(2, 13)
nrest = Cells(3, 2): nload = Cells(3, 4): E = Cells(4, 2)

'インプット (2) -----

For ind = 1 To nnode
x(ind) = Cells(ind + 6, 2): y(ind) = Cells(ind + 6, 3)
Next
For p = 1 To 8: ' for serendipity
For ie = 1 To ne
top(ie, p) = Cells(ie + 6, p + 4)
Next
Next
For j = 1 To 2
For i = 1 To nrest
restraints(i, j) = Cells(i + 6, j + 12)
Next
For i = 1 To nload
Loads(i, j) = Cells(i + 6, j + 14)
Next
Next

'要素剛性マトリクス作成準備-----

c11 = E / (1 - nu ^ 2): c22 = c11: c12 = E * nu / (1 - nu ^ 2)
c66 = E / 2 / (1 + nu)

'要素剛性マトリクス-----

For ie = 1 To ne

For p = 1 To 8
For j = 1 To 8
K11(ie, p, j) = 0: K12(ie, p, j) = 0: K21(ie, p, j) = 0: K22(ie, p, j) = 0
Next
Next
Next

$N(4) = -0.25 * 2 * (1 + ex) * (-1 + ey ^ 2)$
 $N(5) = 0.25 * (1 + ex) * (1 + ey) * (ex + ey - 1)$
 $N(6) = -0.25 * 2 * (-1 + ex ^ 2) * (1 + ey)$
 $N(7) = -0.25 * (-1 + ex) * (1 + ey) * (-ex + ey - 1)$
 $N(8) = 0.25 * 2 * (-1 + ex) * (-1 + ey ^ 2)$
 $DNex(1) = 0.25 * (2 * ex - 2 * ey * ex - ey ^ 2 + ey)$
 $DNex(2) = 0.25 * 4 * ex * (-1 + ey)$
 $DNex(3) = 0.25 * (2 * ex - 2 * ey * ex + ey ^ 2 - ey)$
 $DNex(4) = 0.25 * (2 - 2 * ey ^ 2)$
 $DNex(5) = 0.25 * (2 * ex + 2 * ey * ex + ey ^ 2 + ey)$
 $DNex(6) = 0.25 * (-4 * ex * (1 + ey))$
 $DNex(7) = 0.25 * (2 * ex + 2 * ey * ex - ey ^ 2 - ey)$
 $DNex(8) = 0.25 * (-2 + 2 * ey ^ 2)$
 $DNey(1) = 0.25 * (2 * ey - ex ^ 2 - 2 * ey * ex + ex)$
 $DNey(2) = 0.25 * (-2 + 2 * ex ^ 2)$
 $DNey(3) = 0.25 * (2 * ey - ex ^ 2 + 2 * ey * ex - ex)$
 $DNey(4) = 0.25 * (-4 * (1 + ex) * ey)$
 $DNey(5) = 0.25 * (2 * ey + ex ^ 2 + 2 * ey * ex + ex)$
 $DNey(6) = 0.25 * (2 - 2 * ex ^ 2)$
 $DNey(7) = 0.25 * (2 * ey + ex ^ 2 - 2 * ey * ex - ex)$
 $DNey(8) = 0.25 * 4 * (-1 + ex) * ey$

End Sub

Sub jacobian(DNx(), DNy(), DNex(), DNey(), xx(), yy(), detj)

For p = 1 To 8

ja11 = 0: ja12 = 0: ja21 = 0: ja22 = 0

Next

For p = 1 To 8

ja11 = ja11 + DNex(p) * xx(p): ja12 = ja12 + DNex(p) * yy(p)

ja21 = ja21 + DNey(p) * xx(p): ja22 = ja22 + DNey(p) * yy(p)

Next

detj = ja11 * ja22 - ja12 * ja21

invj11 = ja22 / detj: invj12 = -ja12 / detj: invj21 = -ja21 / detj: invj22 = ja11 / detj

For p = 1 To 8

DNx(p) = invj11 * DNex(p) + invj12 * DNey(p)

DNy(p) = invj21 * DNex(p) + invj22 * DNey(p)

Next

End Sub

Sub elemat(ie, i, c11, c12, c22, c66, DNx(), DNy(), ww(), detj, K11(), K12(), K21(), K22())

For p = 1 To 8

For j = 1 To 8

kk11 = (c11 * DNx(p) * DNx(j) + c66 * DNy(p) * DNy(j))

kk12 = (c12 * DNx(p) * DNy(j) + c66 * DNy(p) * DNx(j))

kk21 = (c12 * DNy(p) * DNx(j) + c66 * DNx(p) * DNy(j))

kk22 = (c66 * DNx(p) * DNx(j) + c22 * DNy(p) * DNy(j))

K11(ie, p, j) = K11(ie, p, j) + ww(i) * kk11 * detj

K12(ie, p, j) = K12(ie, p, j) + ww(i) * kk12 * detj

K21(ie, p, j) = K21(ie, p, j) + ww(i) * kk21 * detj

K22(ie, p, j) = K22(ie, p, j) + ww(i) * kk22 * detj

Next

Next

End Sub

Sub globstifmat(nnode, ne, top(), K11(), K12(), K21(), K22, KG())

For i = 1 To 2 * nnode

For j = 1 To 2 * nnode

KG(i, j) = 0

Next

Next

For ie = 1 To ne

For p = 1 To 8

For j = 1 To 8

ip = top(ie, p)

ij = top(ie, j)

$KG(2 * ip - 1, 2 * ij - 1) = KG(2 * ip - 1, 2 * ij - 1) + K11(ie, p, j)$

$KG(2 * ip - 1, 2 * ij) = KG(2 * ip - 1, 2 * ij) + K12(ie, p, j)$

$KG(2 * ip, 2 * ij - 1) = KG(2 * ip, 2 * ij - 1) + K21(ie, p, j)$

$KG(2 * ip, 2 * ij) = KG(2 * ip, 2 * ij) + K22(ie, p, j)$

Next

Next

Next

End Sub

Sub extforce(nnode, nload, Loads(), f())

For i = 1 To 2 * nnode

f(i) = 0

Next

For i = 1 To nload

f(Loads(i, 1)) = Loads(i, 2)

Next

End Sub

Sub modification(nnode, nrest, KG(), f(), fmod(), Kmod(), restraints())

For i = 1 To 2 * nnode

fmod(i) = f(i)

For j = 1 To 2 * nnode

Kmod(i, j) = KG(i, j)

Next

Next

For i = 1 To 2 * nnode

For ir = 1 To nrest

Kmod(restraints(ir, 1), i) = 0

Kmod(i, restraints(ir, 1)) = 0

```

Kmod(restraints(ir, 1), restraints(ir, 1)) = 1
Next
Next
For i = 1 To nrest: '-----modification of the force vector
For j = 1 To 2 * nnode
fmod(j) = fmod(j) - KG(j, restraints(i, 1)) * restraints(i, 2)
Next
Next
For ir = 1 To nrest
fmod(restraints(ir, 1)) = restraints(ir, 2)
Next
End Sub
Sub matsol(rows, a(), c(), u())
For j = 1 To rows - 1
For i = j + 1 To rows
pp = a(i, j) / a(j, j)
c(i) = c(i) - c(j) * pp

For ind = j To rows
a(i, ind) = a(i, ind) - a(j, ind) * pp
Next
Next
Next
u(rows) = c(rows) / a(rows, rows)
For j = rows - 1 To 1 Step -1
s = 0
For i = j + 1 To rows
s = s + a(j, i) * u(i)
Next
u(j) = (c(j) - s) / a(j, j)
Next
End Sub

Sub nodepoint(sex(), sey())
sex(1) = -1: sex(2) = 0: sex(3) = 1: sex(4) = 1
sex(5) = 1: sex(6) = 0: sex(7) = -1: sex(8) = -1
sey(1) = -1: sey(2) = -1: sey(3) = -1: sey(4) = 0
sey(5) = 1: sey(6) = 1: sey(7) = 1: sey(8) = 0
End Sub

Sub strain(ie, i, DNx(), DNy(), uu(), vv(), eex(), eey(), eet())

DUx = 0: DUy = 0: DVx = 0: DVy = 0

For p = 1 To 8
DUx = DUx + DNx(p) * uu(p): DUy = DUy + DNy(p) * uu(p)
DVx = DVx + DNx(p) * vv(p): DVy = DVy + DNy(p) * vv(p)
Next
eex(ie, i) = DUx: eey(ie, i) = DVy: eet(ie, i) = (DVx + DUy) / 2

End Sub

Sub stress(ie, c11, c22, c12, c66, eex(), eey(), eet(), sx(), sy(), st())
For p = 1 To 8
sx(ie, p) = c11 * eex(ie, p) + c12 * eey(ie, p)
sy(ie, p) = c12 * eex(ie, p) + c22 * eey(ie, p)
st(ie, p) = c66 * eet(ie, p)

```

```
Next  
End Sub
```

```
Sub princstress(ie, sx(), sy(), st(), s1(), s2(), stmax())  
For p = 1 To 8  
half = (sx(ie, p) - sy(ie, p)) / 2  
s1(ie, p) = 0.5 * (sx(ie, p) + sy(ie, p)) + Sqr(half ^ 2 + st(ie, p) ^ 2)  
s2(ie, p) = 0.5 * (sx(ie, p) + sy(ie, p)) - Sqr(half ^ 2 + st(ie, p) ^ 2)  
stmax(ie, p) = Sqr(half ^ 2 + st(ie, p) ^ 2)  
Next  
  
End Sub
```

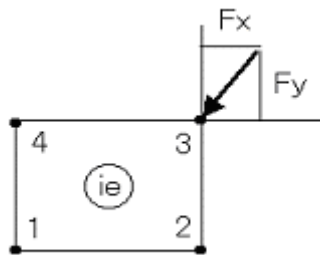
4.8 荷重データの入れ方

平面弾性問題の要素の剛性方程式において、右辺の荷重ベクトルはx、y方向について、それぞれ

$$F1_i = \int_{\Omega_e} tN_i \cdot f_x dA + \int_{\Gamma_e} tN_i \cdot t_x ds \quad F2_i = \int_{\Omega_e} tN_i \cdot f_y dA + \int_{\Gamma_e} tN_i \cdot t_y ds$$

であった。ここで、 t は要素の厚み、 f_x 、 f_y は要素の体積に働く内力（体積力）の x、y 方向の成分、 t_x 、 t_y は境界の線にそってに働く単位長さあたりの外力の x、y 方向の成分である。 N_i は形状関数である。これらの荷重ベクトルのローカル番号で i 番目の成分は、x 方向に対して $F1_i$ 、y 方向について $F2_i$ として計算される。

外力が節点に働く集中荷重の場合



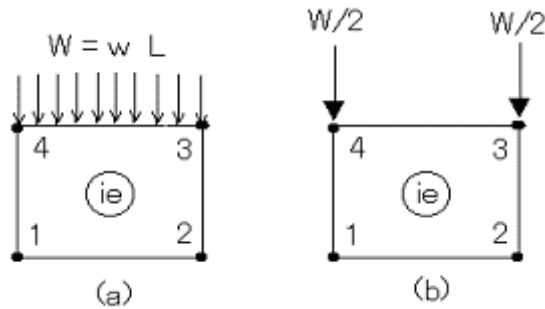
図は長方形 1 次要素の例を示しているが、これは三角形 1 次、四角形 2 次についても同様に考えればよい。要素 (ie) のローカル番号 3 に集中荷重が図のようにかかっているときは、まず荷重を x、y 方向の成分に分ける。そしてローカル番号 3 がグローバル番号 N であるとき、次のようにデータを与える。

荷重番号 $2n - 1$ に荷重 F_x

荷重番号 $2n$ に荷重 F_y

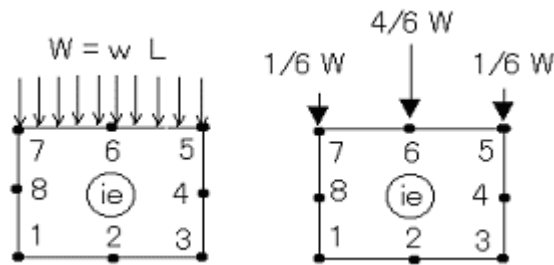
外力が境界に働く等分布荷重の場合

- 長方形 1 次または三角形 1 次要素の場合



要素 (ie) のローカル番号 3 と 4 の間に等分布荷重 $W = WL$ がかかるときは、ローカル番号 3、4 にそれぞれ $W/2$ の集中荷重を振り分けてデータとする。

- 四角形 2 次要素の場合



要素(ie)のローカル番号 5 と 7 の間に等分布荷重 $W = wL$ がかかるときは、ローカル番号 5、6、7 にそれぞれ $W/6$ 、 $4W/6$ 、 $W/6$ の集中荷重を振り分けてデータとする。この理由は次のとおりである。式 (a) の中の外力に対する項はローカル番号 5 , 6 , 7 に対して次のように書くことができる。

$$\int_0^L N_5 \cdot w dx \quad \int_0^L N_6 \cdot w dx \quad \int_0^L N_7 \cdot w dx$$

ここで L は節点 5 から 7 までの辺の長さである。 W は辺 5 7 にかかる単位長さあたりの外力で一定値である。ところで形状関数 N は x の関数で、辺 5 7 は $x = 1$ に相当するから上の積分を x を ξ に変換して $\xi = -1$ から $\xi = 1$

1 にたいして自然座標では2の長さにならなければならないこととする。このとき、

$\frac{\partial x}{\partial \xi} = \frac{L}{2}$ に着目し、定数 $W = wL$ をくくりだして、形状関数について積分する

ると、節点5、6、7にたいして、次の係数がえられる。

$$\frac{1}{2} \int_{-1}^1 N(\xi, \eta)_5 d\xi = \frac{1}{2} \int_{-1}^1 \frac{1}{4} \cdot (1 + \xi) \cdot (1 + \eta) \cdot (\xi + \eta - 1) d\xi = 0.167$$

$$\frac{1}{2} \int_{-1}^1 N(\xi, \eta)_6 d\xi = \frac{1}{2} \int_{-1}^1 \frac{1}{4} \cdot [-2 \cdot (-1 + \xi^2) \cdot (1 + \eta)] d\xi = 0.667$$

$$\frac{1}{2} \int_{-1}^1 N(\xi, \eta)_7 d\xi = \frac{1}{2} \int_{-1}^1 \frac{1}{4} \cdot (-(-1 + \xi) \cdot (1 + \eta) \cdot (-\xi + \eta - 1)) d\xi = 0.167$$

これが、先ほどの W の振り分けの係数 $1/6, 4/6, 1/6$ の根拠である。

体積力の場合

体積力は自重などのように、要素に比重を考慮せねばならない場合の内力である。通常の弾性問題では外力がはるかに大きいので無視できる。

- 長方形1次または三角形1次要素の場合

三角1次要素にたいしては三角形の面積を Ae とし、厚みを t とし、比重をとすると、ローカル節点1, 2, 3に等分に $\rho \cdot t \cdot \frac{Ae}{3}$ の集中荷重を配分して入力データとする。長方形1次要素では、長方形の面積を Ae とすると、ローカル節点1, 2, 3, 4に等分に $\rho \cdot t \cdot \frac{Ae}{4}$ の集中荷重を配分して入力データとする。

- 四角形2次要素の場合

式 (a) のそれぞれの第 1 項が体積力となる。 これらを、次のように書きかえる。

$$\rho \cdot t \cdot \left(\int N_i dA \right) = \rho \cdot t \cdot \int_{-1}^1 \int_{-1}^1 N_i \cdot |J| d\xi d\eta$$

上の面積分は 9 点のガウスポイントにたいする重み係数を与えてつぎのように加算で近似する。

$$\begin{aligned} i &:= 1..3 & j &:= 1..3 \\ Q(\xi, \eta, ie) &:= \rho \cdot t \cdot N(\xi, \eta) \cdot |J(\xi, \eta, ie)| \\ F_{ie} &:= \left(\sum_i \sum_j w_i \cdot w_j \cdot Q(gp_i, gp_j, ie) \right) \end{aligned}$$

荷重番号と荷重

各要素にたいして、グローバル節点番号 N にたいして、

荷重番号 $2n - 1$ に荷重 F_x

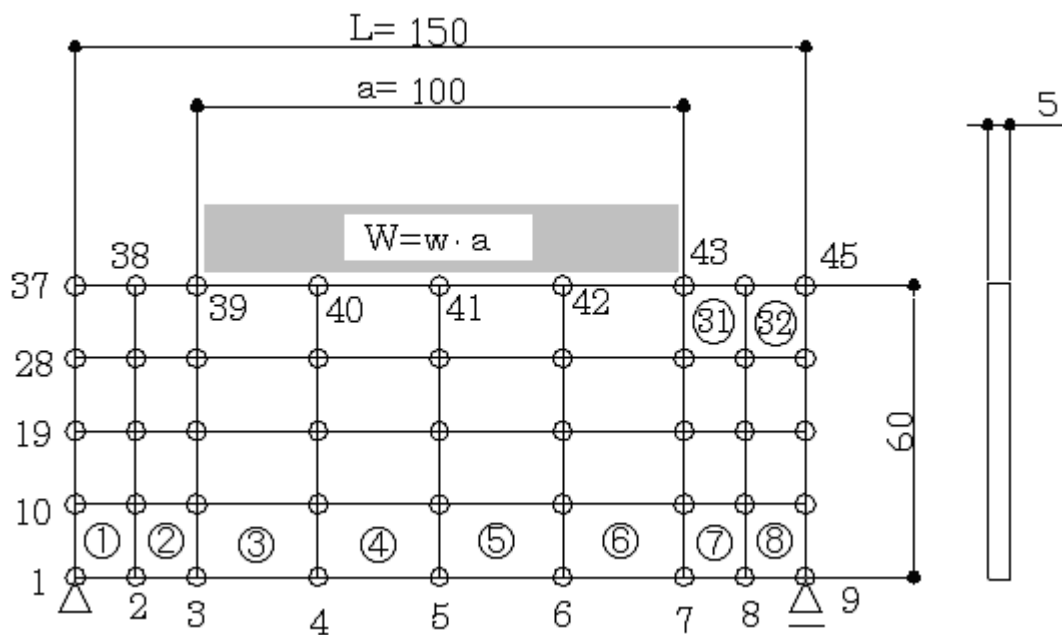
荷重番号 $2n$ に荷重 F_y

を与え、複数の隣あう要素が節点を共有する場合は、その荷重番号の荷重はすべて合計してデータとする。

練習問題

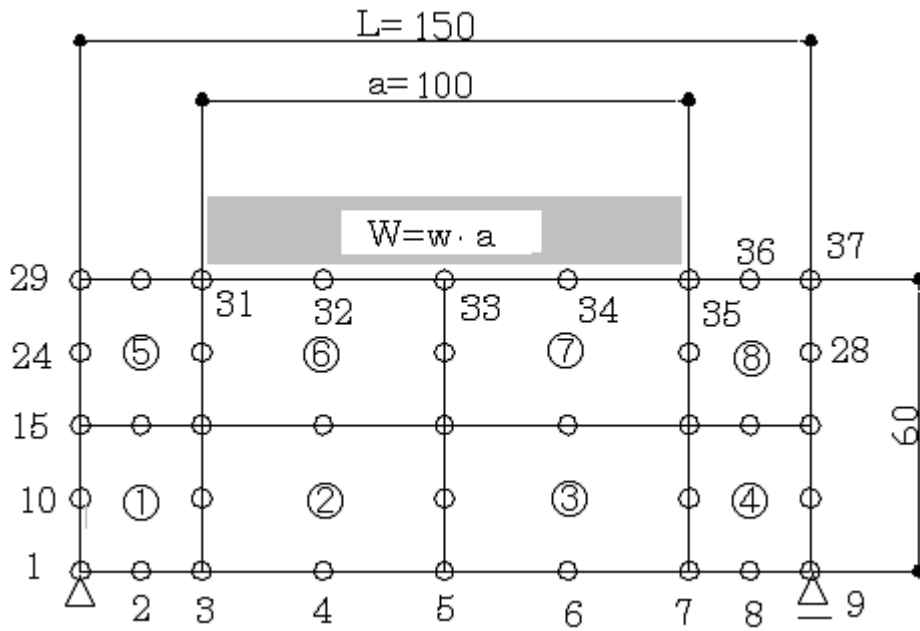
以下に示す練習問題はすべて平面応力問題である。プログラムの係数を若干変更することで、平面ひずみ問題に作り変えることもできるので、読者は以下の問題に追加して、独自の応用問題をつくり計算練習することを推奨する。また材料は鋼とし、すべて $E = 2.06 \times 10^7 \text{ N/cm}^2$ とする。

1.



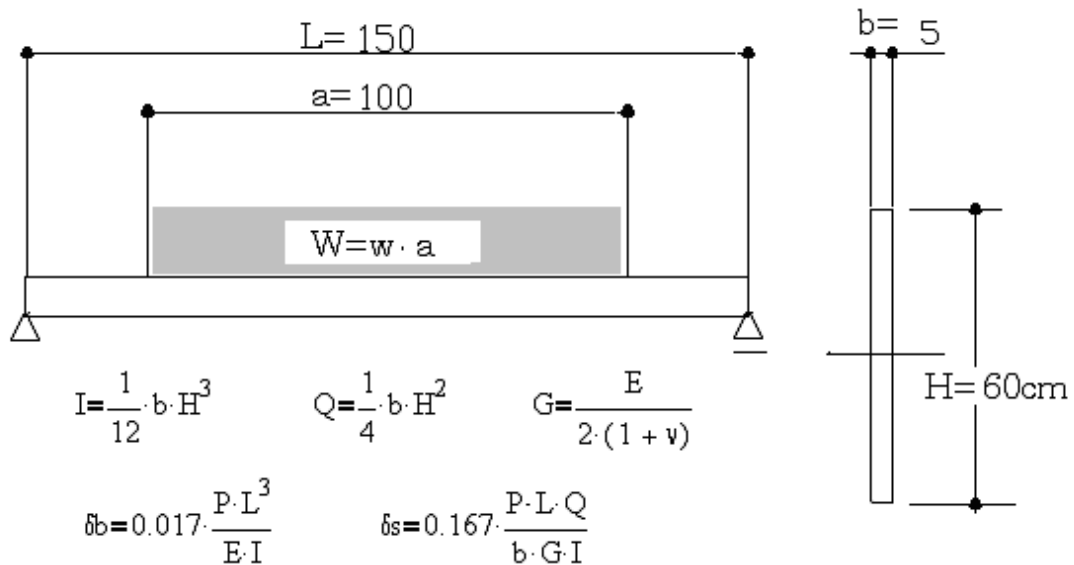
図はプレスのベッドを構成する2枚の鋼板のうちの1枚を示す。 $L = 150 \text{ cm}$ 、 $H = 60 \text{ cm}$ 、板厚 $t = 5 \text{ cm}$ で節点1で x 、 y 方向に拘束され ($u_1 = 0, u_2 = 0$)、節点9で y 方向に拘束されている。 ($u_{18} = 0$) この鋼板は上面の中央 100 cm の長さにあわたり、金型から $W = 200000 \text{ N}$ の等分布荷重を受ける。(単位長さあたりの荷重: $w = W/a = 2000 \text{ N/cm}$)。 上面中央の節点41における、 y 方向のたわみを、長方形1次要素の有限要素モデルによって求めよ。

2.



上図を四角2次有限要素モデルによって解け。

3.



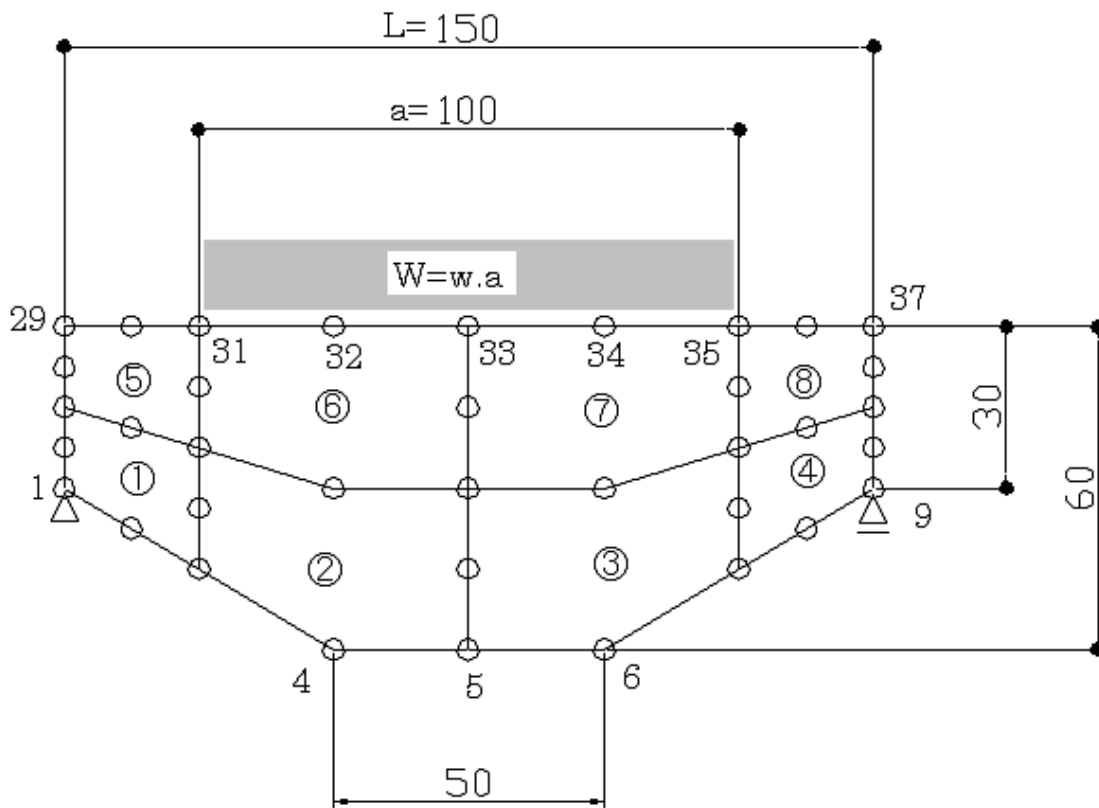
有限要素法をつかわなかった以前は、機械プレスのベッドのたわみにたいして、次のようなクライテリア(設計条件)があたえられていた。ビームの支持点間距離をLとす

ると、 $a = \frac{2}{3}L$ の間に等分布荷重をかけたものとして、中央のたわみを計算する。 L

にくらべて、比較的 H の大きなビームに図のような分布荷重をかけるときは、せん断たわみ δ_s と曲げによるたわみ δ_b を加えたものを、中央のたわみ δ とする。 δ は長さ L の $1/10000$ 近辺以内とする。 問題1の分布荷重条件、鋼板の寸法を利用し、上図に記載されている計算式を使用すると $\delta_s = 0.006323 \text{ cm}$ 、 $\delta_b = 0.006189 \text{ cm}$ 、 $\delta = 0.013$ となり、 $L/10000 = 0.015$ と比較するとこの鋼板の寸法は合格である。

このたわみと問題1、問題2で得られたたわみと比較せよ。

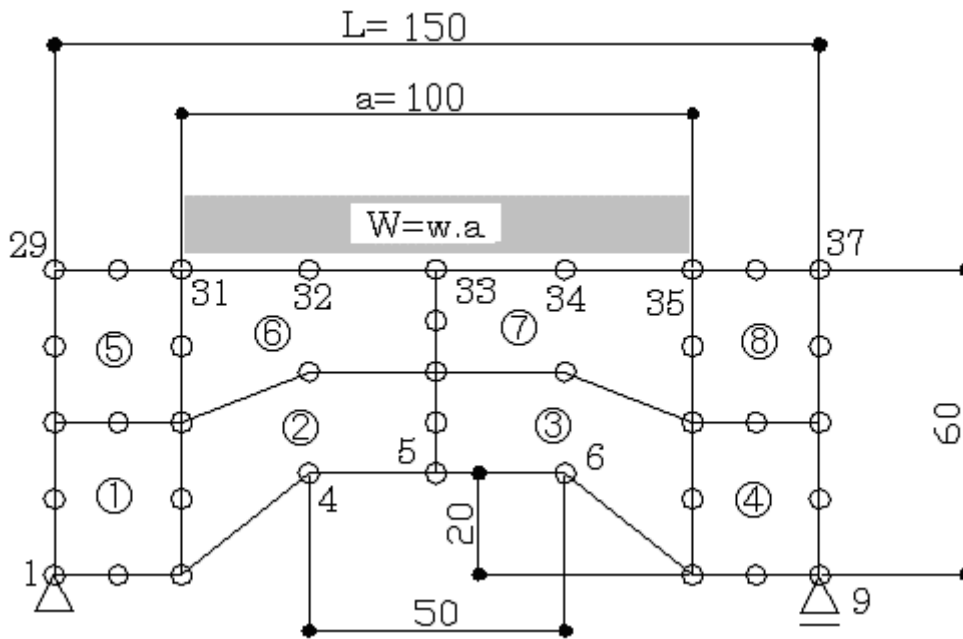
4 .



上図のように問題1 - 2の鋼板の下部隅を切り取ることによって軽量化をはかった。

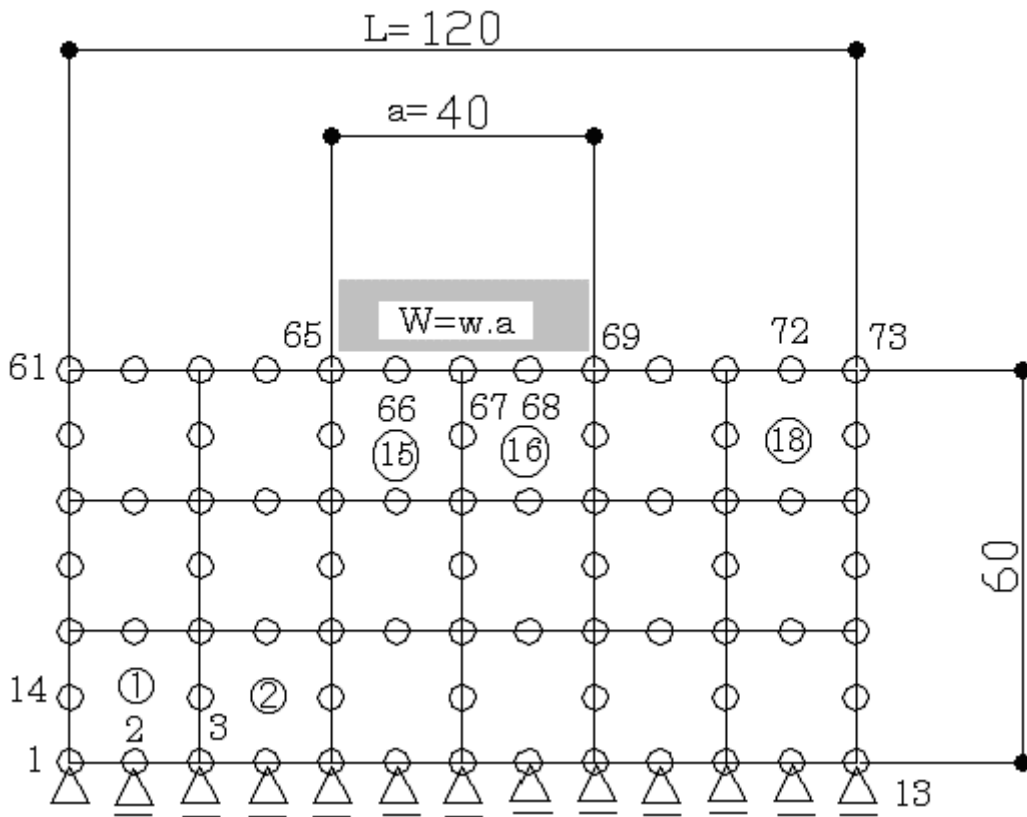
節点33の y 方向のたわみはいくらになるか。

5 .



上図のように問題1 - 2の鋼鈹の下中央部を切り取ることによって軽量化をはかった。
節点33の y 方向のたわみはいくらになるか。

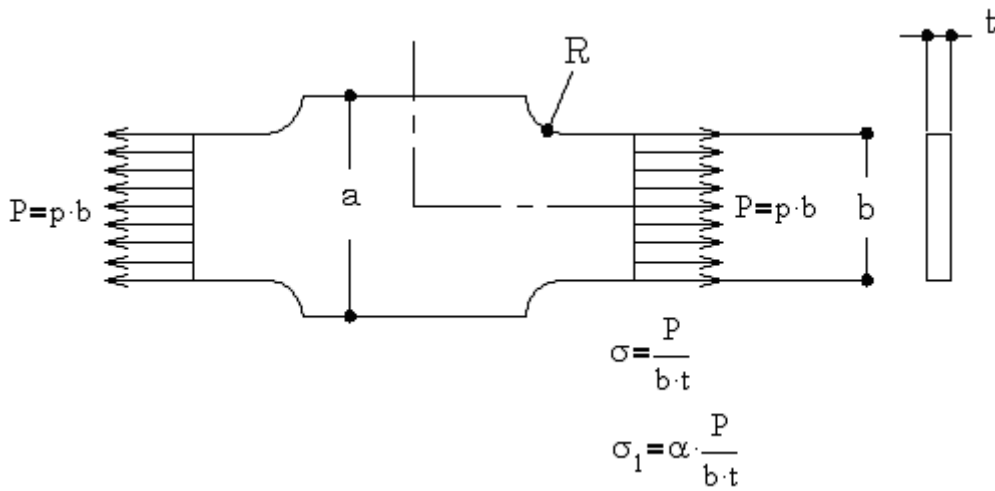
6.



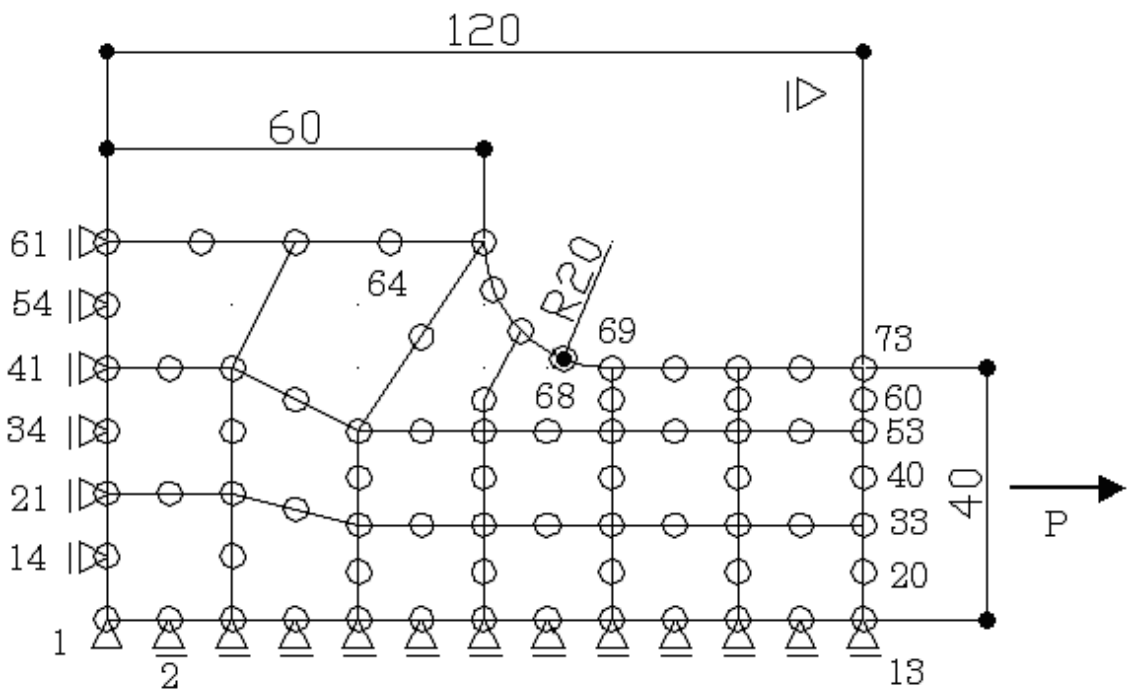
上図のような板の上面中央部には、単位長さあたり 10000 N/cm の等分布荷重がかかり、下面は節点1は x 、 y 両方向に0に拘束され、節点2 --- 13は y 方向のみに0に拘束されている。上面(節点61から73まで)の y 方向の変位はどのようになるか？ 板の厚さ $t = 1 \text{ cm}$ として計算せよ。

(注) 図の板のメッシュ図で、トポロジーテーブルはそのまま使用し、節点の座標を変更すると、デフォルメされた形の板が得られる。境界条件を種種与えて、以下の問題群が計算される。

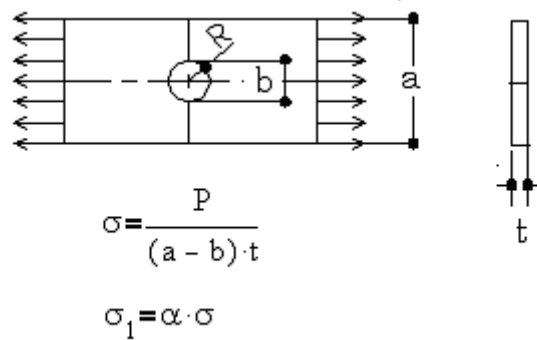
7.



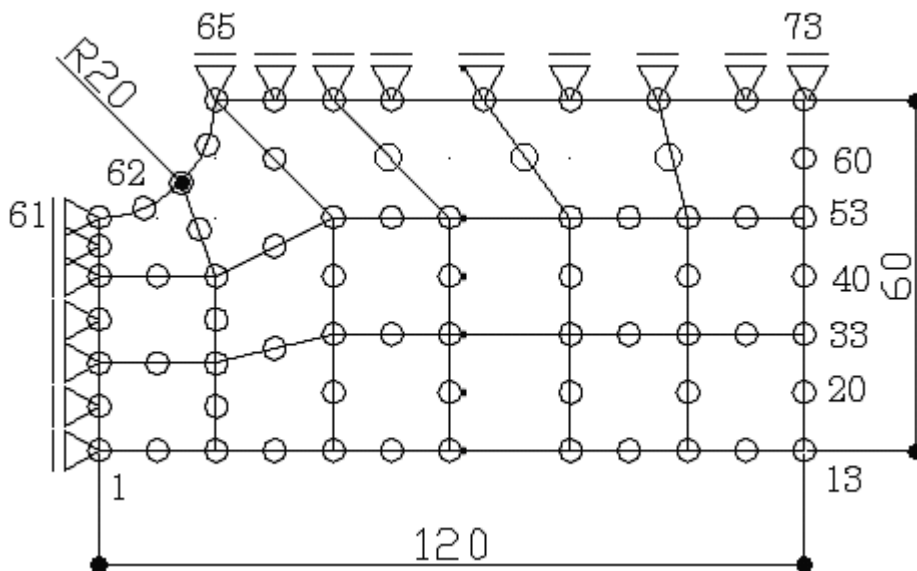
上図に幅が変化する板をしめしている。このような板に引張荷重をかける場合、幅の狭い部分では、公称引張り応力 σ は P/bt のように荷重を断面積で割ったものとして与えられるが、実際にはフィレット部(Rの部分)の応力 σ_1 はその公称応力の α 倍となる。この α を形状係数と呼ぶ。上の図の 1/4 の部分を下图のメッシュ図によって計算し、 α を求めよ。ただし、 $t=1\text{ cm}$ として、節点13 --- 73の右端に単位長さあたり、5000 N/cm の等分布荷重がかかるものとして計算せよ。



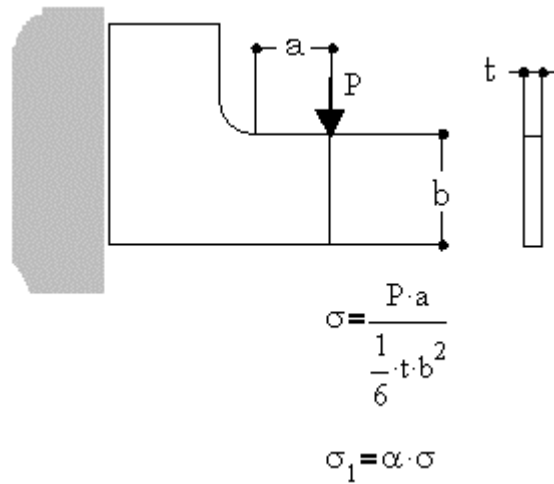
8.



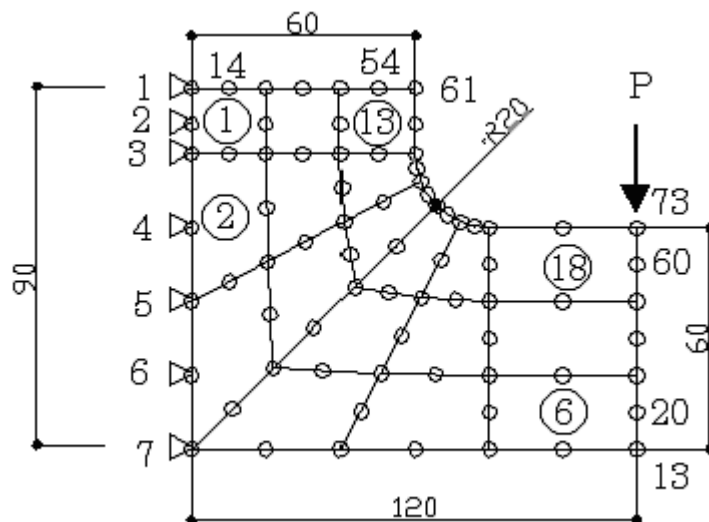
図に中央に丸穴のある板をしめしている。このような板に引張荷重をかける場合、もっとも幅の狭い部分では、公称引張り応力 σ は $P/(a-b)t$ のように荷重を断面積で割ったものとして与えられるが、実際にはフィレット部 (R の部分) の応力 σ_1 はその公称応力の α 倍となる。この α を形状係数と呼ぶ。上の図の 1/4 の部分を下图のメッシュ図によって計算し、 α を求めよ。ただし、 $t = 1 \text{ cm}$ として、節点 13 --- 73 の右端に単位長さあたり、 5000 N/cm の等分布荷重がかかるものとして計算せよ。



9 .



図に幅が変化する板をしめしている。このような板の先端に下向きの荷重をかける場合、フィレットの部分の、公称曲げ応力 σ は図中の式で与えられるが、実際にはフィレット部(Rの部分)の応力 σ_1 はその公称応力の α 倍となる。この α を形状係数と呼ぶ。上の図の 1/4 の部分を下のメッシュ図によって計算し、 α を求めよ。ただし、 $t=1 \text{ cm}$ として、節点 13 --- 73 の右端に単位長さあたり、5000 N/cm の等分布荷重がかかるものとして計算せよ。



解答へのヒント

1. トポロジーテーブル、境界条件、節点 41 における y 方向たわみ u_{82} は次のとおりである。

	1	2	3	4
17	19	20	29	28
18	20	21	30	29
19	21	22	31	30
20	22	23	32	31
21	23	24	33	32
22	24	25	34	33
23	25	26	35	34
24	26	27	36	35
25	28	29	38	37
26	29	30	39	38
27	30	31	40	39
28	31	32	41	40
29	32	33	42	41
30	33	34	43	42
31	34	35	44	43

$$\text{Top} =$$

$$\text{Restrains} := \begin{bmatrix} 1 & 0 \\ 2 & 0 \\ 18 & 0 \end{bmatrix} \quad L := 100 \quad P := 200000 \quad ff := \frac{P}{L}$$

$$\text{Loads} := \begin{bmatrix} 78 & -\frac{1}{2} \cdot ff \cdot 25 \\ 80 & -\frac{1}{2} \cdot ff \cdot 25 - \frac{1}{2} \cdot ff \cdot 25 \\ 82 & -\frac{1}{2} \cdot ff \cdot 25 - \frac{1}{2} \cdot ff \cdot 25 \\ 84 & -\frac{1}{2} \cdot ff \cdot 25 - \frac{1}{2} \cdot ff \cdot 25 \\ 86 & -\frac{1}{2} \cdot ff \cdot 25 \end{bmatrix} \quad u_{82} = -0.011$$

2. トポロジーテーブル、境界条件、板上面中央節点 33 における y 方向たわみ u_{66} は次のとおりである。

1	2	3	11	17	16	15	10
3	4	5	12	19	18	17	11
5	6	7	13	21	20	19	12
7	8	9	14	23	22	21	13
15	16	17	25	31	30	29	24
17	18	19	26	33	32	31	25
19	20	21	27	35	34	33	26
21	22	23	28	37	36	35	27

$$\text{Top} =$$

$$\text{Restrains} := \begin{bmatrix} 1 & 0 \\ 2 & 0 \\ 18 & 0 \end{bmatrix} \quad \text{Loads} := \begin{bmatrix} 62 & -\frac{1}{6} \cdot w \cdot L \\ 64 & -\frac{4}{6} \cdot w \cdot L \\ 66 & -\frac{1}{6} \cdot w \cdot L - \frac{1}{6} \cdot w \cdot L \\ 68 & -\frac{4}{6} \cdot w \cdot L \\ 70 & -\frac{1}{6} \cdot w \cdot L \end{bmatrix}$$

$$u_{66} = -0.012$$

3. 従来の計算で得られた上面中央のたわみ $= 0.013$ は、問題 1 における $u_{82} = -0.011$ と問題 2 における $u_{66} = -0.012$ とよく似ている。

4. 問題 2 の節点の座標を変更するだけで、トポロジーテーブル、境界条件がそのまま使用できる。 $u_{66} = -0.012$

5. $u_{66} = -0.026$ となり、問題 4 のたわみの 2 倍以上となる。中央下部の切り欠きはたわみの上で不利となる。

6. トポロジーテーブルの一部と境界条件は次のとおりである。

$$\text{Top1} := \begin{bmatrix} 1 & 2 & 3 & 15 & 23 & 22 & 21 & 14 \\ 3 & 4 & 5 & 16 & 25 & 24 & 23 & 15 \\ 5 & 6 & 7 & 17 & 27 & 26 & 25 & 16 \\ 7 & 8 & 9 & 18 & 29 & 28 & 27 & 17 \\ 9 & 10 & 11 & 19 & 31 & 30 & 29 & 18 \\ 11 & 12 & 13 & 20 & 33 & 32 & 31 & 19 \end{bmatrix}$$

$$i := 1..13$$

$$\text{Restrains}_{i,1} := 1$$

$$\text{Restrains}_{i+1,1} := 2 \cdot i$$

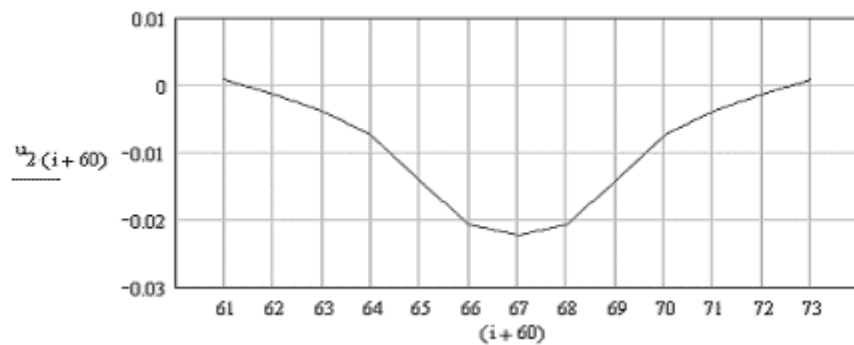
$$\text{Restrains}_{i+1,2} := 0$$

$$w := \frac{400000}{40}$$

$$L := 20$$

$$\text{Loads} := \begin{bmatrix} 130 & -\frac{1}{6} \cdot w \cdot L \\ 132 & -\frac{4}{6} \cdot w \cdot L \\ 134 & -\frac{1}{6} \cdot w \cdot L - \frac{1}{6} \cdot w \cdot L \\ 136 & -\frac{4}{6} \cdot w \cdot L \\ 138 & -\frac{1}{6} \cdot w \cdot L \end{bmatrix}$$

板の上面（節点 61 --- 73）のたわみの分布は次のようになる。



7. トポロジーテーブルは問題 6 と同じである。境界条件は次のようになる。

```

i := 1..13
Restraints1i,1 := 2·i
Restraints1i,2 := 0
Restraints2 :=
[ 1 0
27 0
41 0
67 0
81 0
107 0
121 0 ]
Restraints := stack(Restraints1, Restraints2)
Restraints := csort(Restraints, 1)

```

```

ff := 200000 / 40      ff = 5·103
L1 := 15              L2 := 10
Loads :=
[ 25   1/6·ff·L1
39   4/6·ff·L1
65   1/6·ff·L1 + 1/6·ff·L1
79   4/6·ff·L1
105  1/6·ff·L1 + 1/6·ff·L2
119  4/6·ff·L2
145  1/6·ff·L2 ]

```

要素 のフィレット部のローカル節点番号 5 , 6 , 7 にたいする主応力を求めると次のようになり、

```

p := 5..7
σp,16
[ 8.834·103
7.209·103
4.038·103 ]

```

ローカル番号 5 番目 (グローバル節点番号 69) では 8838 となっており、公称引張応力 5000 の約 1.7 倍の応力となっている。したがってこの場合の形状係数は 1.7 程度であろう。

8. トポロジーテーブルは問題 6 と同じである。境界条件は次のようになる。

$$\begin{array}{l}
 i := 1..9 \\
 \text{Restrains1}_{i,1} := 2 \cdot (i + 64) \\
 \text{Restrains1}_{i,2} := 0 \qquad \text{Restrains2} :=
 \end{array}
 \begin{bmatrix}
 1 & 0 \\
 27 & 0 \\
 41 & 0 \\
 67 & 0 \\
 81 & 0 \\
 107 & 0 \\
 121 & 0
 \end{bmatrix}
 \begin{array}{l}
 \text{Restrains} := \text{stack}(\text{Restrains1}, \text{Restrains2}) \\
 \text{Restrains} := \text{csort}(\text{Restrains}, 1)
 \end{array}$$

$$\begin{array}{l}
 ff := \frac{300000}{60} \qquad ff = 5 \cdot 10^3 \\
 L := 20
 \end{array}
 \begin{array}{l}
 \text{Loads} :=
 \end{array}
 \begin{bmatrix}
 27 & \frac{1}{6} \cdot ff \cdot L \\
 39 & \frac{4}{6} \cdot ff \cdot L \\
 45 & \frac{1}{6} \cdot ff \cdot L + \frac{1}{6} \cdot ff \cdot L \\
 79 & \frac{4}{6} \cdot ff \cdot L \\
 105 & \frac{1}{6} \cdot ff \cdot L + \frac{1}{6} \cdot ff \cdot L \\
 119 & \frac{4}{6} \cdot ff \cdot L \\
 145 & \frac{1}{6} \cdot ff \cdot L
 \end{bmatrix}$$

要素 のフィレット部のローカル節点番号 1 --- 8 にたいする主応力を求めると次のようになり、

$$\begin{array}{l}
 p := 1..8 \qquad \sigma_{1,p,13}
 \end{array}
 \begin{array}{|c|}
 \hline
 8.174 \cdot 10^3 \\
 \hline
 7.133 \cdot 10^3 \\
 \hline
 6.285 \cdot 10^3 \\
 \hline
 5.725 \cdot 10^3 \\
 \hline
 4.072 \cdot 10^3 \\
 \hline
 1.146 \cdot 10^4 \\
 \hline
 1.503 \cdot 10^4 \\
 \hline
 9.69 \cdot 10^3 \\
 \hline
 \end{array}$$

ローカル番号 8 番目 (グローバル節点番号 63) では 15030 となっており、公

称引張応力 5000 の約 3 倍の応力となっている。 したがってこの場合の形状係数は 3 程度であろう。

9. トポロジーテーブルは問題 6 と同じである。 境界条件は次のようになる。

$$\begin{aligned} i &:= 1..14 \\ \text{Restrains}_{i,1} &:= i & P &:= 50000 \\ \text{Restrains}_{i,2} &:= 0 & \text{Loads} &:= (146 - P) \end{aligned}$$

下の主応力が最大で

$$\sigma_{1,5,16} = 6.762 \cdot 10^3 \quad \blacksquare$$

公称曲げ応力

$$\sigma := \frac{P \cdot 50}{\frac{1}{6} \cdot t \cdot h^2} \quad \sigma = 4.167 \cdot 10^3$$

と比較すると約 1.6 倍の応力となっている。 したがってこの場合の形状係数は 1.6 程度であろう。