

第13章の問題 解答例

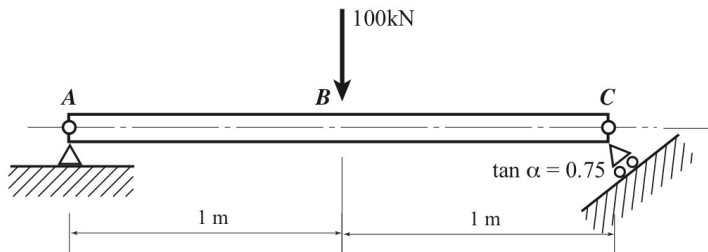
□ 1

```
clear; format shortG
```

目的

次のピン支点と傾いたローラー支点により指示される、長さ2メートルで中央に100kN の下向きの力が作用するはりの応力を変位法により求める。

部材の断面積は $A = 8 \times 10^3 \text{ mm}^2$ 、断面二次モーメントが $I = 100 \times 10^6 \text{ mm}^4$ 、ヤング率が $E = 20 \times 10^3 \text{ kN/mm}^2$ 、せん断弾性係数が $G = 9 \times 10^3 \text{ kN/mm}^2$ とする。ローラーの傾きの $\tan \alpha$ は 0.75 とする。



構造物の基本データ

```
title = 'ex13_1'; % 課題名称 (解析の名前などの情報)
unitl = 'mm'; % 長さの単位
unitf = 'kN'; % 力の単位
nnode = 3; % 節点数
nmemb = 2; % 部材数
nload = 1; % 荷重条件の数
```

材料に関するデータ

すべての部材の材料定数は、次の値を使って計算される。G = 0 の場合には、部材のせん断変形は無視される。

```
E = 20000; % 材料のヤング率 (単位: kN/mm^2)
G = 9000; % 材料のせん断弾性係数 (単位: kN/mm^2)
```

節点に関するデータ

節点ごとに節点に関するデータを設定する。節点番号は1からはじめて nnode までの値とする。

- 設定する値は、節点ごとに、(1) 節点番号、(2) 節点の支持条件を表す数値、(3) 基準座標系での節点の位置の x 座標 (右向き正)、(4) 基準座標系での節点の位置の y 座標 (上向き正)、(5) 支持点のローラーの方向と基準座標系の x 軸の方向の角度 θ の $\sin \theta$ の値、の順とする。
- 節点の支持条件を表す数値は、自由節点: 0、固定支持: 1、ピン支持: 2、ローラー支持: 3 とする。

```
node(1,:) = [1 2 0 0 0]; % 節点 1(A) の設定値
node(2,:) = [2 0 1000 0 0]; % 節点 2(中央) の設定値
node(3,:) = [3 3 2000 0 0.75]; % 節点 3(B) の設定値
```

部材に関するデータ

部材ごとに、部材に関するデータを設定する。部材番号は、1からはじめて nmemb までの値とする。

- 設定する値は、部材ごとに、(1) 部材番号、(2) 部材の 1-端の節点番号、(3) 部材の 2-端の節点番号、(4) 部材の端部の接続条件を表す数値、(5) 部材の断面積、(6) 部材の断面二次モーメント、(7) 部材の 1-端の剛域長さ、(8) 部材の 2-端の剛域長さ、(9) 部材のせん断変形に関する形状係数 κ の順とする。
- 部材の端部の接続条件を表す数値は、0 から 3 の整数から選ぶ。1) 両端剛: 0、2) 1-端がピン、2-端が剛接: 1、3) 2-端がピン、1-端が剛接: 2、4) 両端がピン: 3 とする。
- 部材のせん断変形に関する形状係数 κ は、断面形状により異なる値をとることが知られている。なお、長方形 (正方形) 断面の場合には 1.2 である。

```
member(1,:) = [1 1 2 0 8e3 200e6 0 0 1.2]; % 部材 1 の設定値
member(2,:) = [2 2 3 0 8e3 200e6 0 0 1.2]; % 部材 2 の設定値
```

節点荷重に関するデータ

節点に働く荷重もしくは強制変位の大きさを設定する。複数の節点に荷重もしくは強制変位が作用するときには、必要なだけの数のデータを追加する。荷重条件番号が同じ荷重は、それらはすべて同じ荷重条件の一部として取り扱われる。節点荷重がひとつもない場合でも、節点荷重をすべてゼロとしたダミーのデータが一つ以上必要であるので注意する。

- 設定する値は、節点に働く荷重ごとに (1) 荷重条件番号、(2) 節点番号、(3) 基準座標系で表された荷重の x 方向成分 P_x の数値、(4) 基準座標系で表された荷重の y 方向成分 P_y の数値、(5) 基準座標系で表された荷重のモーメント M_z の数値、(6) 基準座標系で表された節点の x 方向の強制変位、(7) 基準座標系で表された節点の y 方向の強制変位、(8) 基準座標系で表された節点の強制回転角とする。

```
load1(1,:) = [1      2      0 -100      0      0      0      0]; % 1 番目の節点荷重の設定値
```

部材荷重に関するデータ

中間荷重の作用する部材は、中間荷重の大きさを、部材ごとに設定する。中間荷重の作用する部材がない場合でも、部材荷重をすべてゼロとしたダミーのデータが一つ以上必要であるので注意する。

- 複数の部材に荷重が作用するときには、必要なだけの数のデータを追加できる。荷重条件番号が同じ荷重は、それらはすべて同じ荷重条件の一部として取り扱われる。
- 本スクリプトでは、中間荷重の大きさを、固定端反力の数値で指定する。
- 設定するデータは、部材ごとに、(1) 荷重条件番号、(2) 部材番号、(3) 部材座標系で表した 1-端の固定端反力の x 方向成分 P_{1x} の数値、(4) 部材座標系で表した 1-端の固定端反力の y 方向成分 P_{1y} の数値、(5) 部材座標系で表した 1-端の固定端モーメント M_{1z} の数値、(6) 部材座標系で表した 2-端の固定端反力の x 方向成分 P_{2x} の数値、(7) 部材座標系で表した 2-端の固定端反力の y 方向成分 P_{2y} の数値、(8) 部材座標系で表した 2-端の固定端モーメント M_{2z} の数値である。
- 部材座標系の x -軸は、部材の 1-端から 2-端に向かう向きとする。部材座標系の y -軸は、 x -軸から反時計回りに 90 度回転した向きとする。

```
load2(1,:) = [1      1      0      0      0      0      0      0]; % 1 番目の部材荷重の設定値
```

変位法による解析の実行

% 使用する主な配列

```
load    = zeros(nnode*3,nload); % 節点荷重ベクトル
displ   = zeros(nnode*3,nload); % 節点変位ベクトル
sk      = zeros(6,6);           % 部材の剛性マトリックス
skk     = zeros(nnode*3,nnode*3); % 構造物の剛性マトリックス
mload   = zeros(6,nmemb,nload); % 部材荷重の固定端反力（入力データ）
mforce  = zeros(6,nmemb,nload); % 部材端力（出力データ）
support = zeros(1,nnode*3);     % 支持条件（境界条件）を表すベクトル
```

節点荷重ベクトルと節点変位ベクトルの作成

```
% 節点荷重と節点変位
[szr,~] = size(load1);
if szr >= 1
    for i=1:szr
        il = load1(i,1);
        n  = load1(i,2);
        ind = [3*n-2 3*n-1 3*n];
        load(ind,il) = load(ind,il) + load1(i,3:5)'; % 節点荷重
        displ(ind,il) = displ(ind,il) + load1(i,6:8)'; % 節点変位
    end
end
```

節点荷重ベクトルに部材荷重を考慮

```
% 部材荷重
[szr,~] = size(load2);
if szr >= 1
    for i = 1:szr
        il = load2(i,1);
        m  = load2(i,2);
        mload(1:6,m,il) = mload(1:6,m,il) + load2(i,3:8)'; % 固定端反力
    end
```

```
end
```

```
% 固定端反力の符号を逆にしたものを節点荷重データに加える
for m=1:nmemb
    n1 = member(m,2); % 部材端 (1) の節点番号
    n2 = member(m,3); % 部材端 (2) の節点番号
    delx=node(n2,3)-node(n1,3);
    dely=node(n2,4)-node(n1,4);
    al = sqrt(delx*delx+dely*dely);
    cs = delx/al;
    sn = dely/al;
    T = [cs -sn 0; sn cs 0; 0 0 1]; % 座標変換マトリクス
    for il = 1:nload
        pm1 = mload(1:3,m,il);
        pm2 = mload(4:6,m,il);
        in1 = [3*n1-2 3*n1-1 3*n1];
        in2 = [3*n2-2 3*n2-1 3*n2];
        % 部材の固定端反力を基準座標系に変換し符号を逆にして節点荷重に加算する
        load(in1,il) = load(in1,il) - T*pm1;
        load(in2,il) = load(in2,il) - T*pm2;
    end
end
```

構造物の剛性マトリクスの作成

```
% 基準座標系による構造物の剛性マトリクスの作成
for m=1:nmemb
    [k,T] = stiff(m,member,node,E,G); % 部材の剛性マトリクスと部材座標系の座標変換マトリクス
    n1 = member(m,2); % 部材端 (1) の節点番号
    n2 = member(m,3); % 部材端 (2) の節点番号
    index = [3*n1-2 3*n1-1 3*n1 3*n2-2 3*n2-1 3*n2]; % 部材端の自由度の番号
    skk(index,index) = skk(index,index) + T*k*T'; % 構造物の剛性マトリクス
end
% ローラー支点到節点座標系を適用した剛性マトリクスに変換
for nn = 1:nnode % 節点
    if node(nn,2) == 3
        sn = node(nn,5);
        cs = sqrt(1-sn*sn);
        T = [cs -sn 0; sn cs 0; 0 0 1]; % 節点座標系の座標変換マトリクス
        index = [3*nn-2 3*nn-1 3*nn];
        displ(index,:) = T' * displ(index,:); % 節点変位を節点座標系に変換
        load(index,:) = T' * load(index,:); % 節点荷重を節点座標系に変換
        skk(index,:) = T' * skk(index,:); % 全体構造物の剛性マトリクスを節点座標系に変換
        skk(:,index) = skk(:,index) * T; % 全体構造物の剛性マトリクスを節点座標系に変換
    end
end
```

自由節点変位と支点反力の解法

```
% 節点の支持条件を表すベクトル support を作成
for n=1:nnode
    is = node(n,2);
    index = [3*n-2 3*n-1 3*n];
    if is == 0 % 自由節点の場合
        support(index)=[0 0 0];
    elseif is == 1 % 固定支点の場合
        support(index)=[1 1 1];
    elseif is == 2 % ピン支点の場合
        support(index)=[1 1 0];
    elseif is == 3 % ピン・ローラー支持の場合
        support(index)=[0 1 0];
    end
end
% 変位が未知である自由度のならびを表す数列 indf
indf = find(support==0);

% 変位が既知の自由度のならびを表す数列 inds
inds = find(support);
```

```

% 未知の変位と反力を求める
kff = ssk(indf,indf);
kfs = ssk(indf,inds);
ksf = ssk(inds,indf);
kss = ssk(inds,inds);
displ(indf,:)= kff\((load(indf,:)-kfs*displ(inds,:)); % 未知の節点変位を求める
load(inds,:)= ksf*displ(indf,:)+kss*displ(inds,:); % 未知の節点反力を求める

% ローラー支点の節点座標で表された変位と節点力を基準座標系に戻す
for nn=1:nnode % 節点
    if node(nn,2) == 3
        sn = node(nn,5);
        cs = sqrt(1-sn*sn);
        T = [cs -sn 0; sn cs 0; 0 0 1]; % 節点座標系の座標変換マトリクス
        index = [3*nn-2 3*nn-1 3*nn];
        displ(index,:) = T * displ(index,:); % 節点変位を基準座標系に変換
        load(index,:) = T * load(index,:); % 節点荷重を基準座標系に変換
    end
end

```

部材端に作用する力

```

% 部材端力の計算
for nl=1:nload
    for m=1:nmemb
        [k,T] = stiff(m,member,node,E,G); % 部材の剛性マトリックス k と部材座標系の座標変換マトリクス T
        n1 = member(m,2);
        n2 = member(m,3);
        index = [3*n1-2, 3*n1-1, 3*n2-2, 3*n2-1, 3*n2];
        mforce(1:6,m,nl) = k*T'* displ(index,nl); % 部材端力
    end
end

% 部材端力に、部材の固定端荷重を重ね合わせる。
mforce = mforce + mload;

```

入力データの表示

構造物の基本データの表示

```
fprintf('\n ***** 課題名称 *****\n\n');
```

```
***** 課題名称 *****
```

```
fprintf(' %s\n\n', title);
```

```
ex13_1
```

```
fprintf(' ***** 入力データ *****\n\n');
```

```
***** 入力データ *****
```

```
fprintf(' 共通事項:\n\n');
```

```
共通事項:
```

```
fprintf(' 節点数 = %d 部材数 = %d 荷重条件数 = %d \n\n', ...
        nnode, nmemb, nload);
```

```
節点数 = 3 部材数 = 2 荷重条件数 = 1
```

材料に関するデータの表示

```
fprintf(' 材料特性データ:\n\n')
```

```
材料特性データ:
```

```
fprintf(' E = %8.3e ( %s / %s **2) \n G = %8.3e ( %s / %s **2) \n\n', ...
        E, unitf, unitl, G, unitf, unitl);
```

```
E = 2.000e+04 ( kN / mm **2)
G = 9.000e+03 ( kN / mm **2)
```

節点に関するデータの表示

```
fprintf(' 節点データ:\n\n')
```

節点データ:

```
fprintf([' 節点      支持      座標      ローラーの傾きの\n' ...
        ' 番号      条件      x(%4s)      y(%4s)      sin θ\n'], unitl,unitl);
```

節点 番号	支持 条件	座標 x(mm)	y(mm)	ローラーの傾きの sin θ
----------	----------	--------------	--------	-------------------

```
for i=1:nnode
    if node(i,2)==1
        fix = 'fix';
    elseif node(i,2)==2
        fix = 'pin';
    elseif node(i,2)==3
        fix = 'rol';
    else
        fix = ' ';
    end
    fprintf(1,' %d      %s      %8.1f %8.1f %8.1f \n', ...
            node(i,1), fix, node(i,3), node(i,4), node(i,5))
end
```

1	pin	0.0	0.0	0.0
2		1000.0	0.0	0.0
3	rol	2000.0	0.0	0.8

部材に関するデータの表示

```
fprintf('\n 部材データ1:\n\n')
```

部材データ1:

```
fprintf([' 部材  節点      接続条件  部材長さ  方向      剛域1  剛域2 \n' ...
        ' 番号  1端 2端  1端 2端      ( %s)  cos   sin   ( %s ) ( %s )\n'], ...
        unitl,unitl,unitl)
```

部材 番号	節点 1端 2端	接続条件 1端 2端	部材長さ (mm)	方向 cos sin	剛域1 (mm)	剛域2 (mm)
----------	-------------	---------------	---------------	---------------	--------------	--------------

```
for i=1:nmemb
    if member(i,4) == 0
        fix1='fix'; fix2='fix';
    elseif member(i,4) == 1
        fix1='pin'; fix2='fix';
    elseif member(i,4) == 2
        fix1='fix'; fix2='pin';
    elseif member(i,4) == 3
        fix1 = 'pin'; fix2='pin';
    end
    n1 = member(i,2);
    n2 = member(i,3);
    delx=node(n2,3)-node(n1,3);
    dely=node(n2,4)-node(n1,4);
    al = sqrt(delx*delx+dely*dely);
    cs = delx/al;
    sn = dely/al;
    rgd1 = member(i,7);
    rgd2 = member(i,8);
    fprintf(' %3d %3d %3d %4s %4s %6.1f %6.3f %6.3f %6.2f %6.2f\n',...
            member(i,1), n1,n2,fix1,fix2,al,cs,sn,rgd1,rgd2)
```

```
end
```

1	1	2	fix	fix	1000.0	1.000	0.000	0.00	0.00
2	2	3	fix	fix	1000.0	1.000	0.000	0.00	0.00

```
fprintf('\n 部材データ2:\n\n')
```

部材データ2:

```
fprintf([' 部材          断面積  断面二次モーメント 有効せん断断面積 形状係数\n' ...  
        ' 番号          (%4s**2)    (%4s**4)    (%4s**2)      κ\n'], ...  
        unitl, unitl, unitl)
```

部材	断面積	断面二次モーメント	有効せん断断面積	形状係数
番号	(mm**2)	(mm**4)	(mm**2)	κ

```
for i=1:nmemb  
    a      = member(i,5);  
    ai     = member(i,6);  
    akappa = member(i,9);  
    n1     = member(i,2);  
    n2     = member(i,3);  
    delx   = node(n2,3)-node(n1,3);  
    dely   = node(n2,4)-node(n1,4);  
    al     = sqrt(delx*delx+dely*dely);  
    rgd1   = member(i,7);  
    rgd2   = member(i,8);  
    aln    = al - rgd1 - rgd2;  
    if akappa == 0  
        aq = 0;  
    elseif G == 0  
        aq = 0;  
    else  
        aq = a/akappa;  
    end  
    fprintf(1, '%3d    %10.3e    %10.3e    %10.3e    %6.3f \n', ...  
            member(i,1), a, ai, aq, akappa)  
    disp('')  
end
```

1	8.000e+03	2.000e+08	6.667e+03	1.200
2	8.000e+03	2.000e+08	6.667e+03	1.200

節点荷重に関するデータの表示

```
fprintf('\n 節点荷重データ:\n\n')
```

節点荷重データ:

```
fprintf([' 荷重    節点    節点荷重          強制変位\n' ...  
        ' 条件    番号    Px              Py              Mz              ux              uy              θz\n'], ...  
        θz)
```

荷重	節点	節点荷重			強制変位		
条件	番号	Px	Py	Mz	ux	uy	θz

```
[szr,~] = size(load1);  
for i=1:szr  
    il = load1(i,1);  
    n  = load1(i,2);  
    ind = [3*n-2 3*n-1 3*n];  
    fprintf(' %3d %3d    %10.3e    %10.3e    %10.3e    %10.3e    %10.3e    %10.3e\n', ...  
            il, n, load1(i,3), load1(i,4), load1(i,5), load1(i,6), load1(i,7), load1(i,8))  
end
```

1	2	0.000e+00	-1.000e+02	0.000e+00	0.000e+00	0.000e+00	0.000e+00
---	---	-----------	------------	-----------	-----------	-----------	-----------

部材荷重に関するデータの表示

```
fprintf('\n 部材荷重データ:\n\n')
```

部材荷重データ:

```
fprintf([' 荷重    部材    部材荷重(1)端          部材荷重(2)端\n' ...  
        ' 条件    番号    Px1          Py1          Mz1          Px2          Py2          Mz2\n'])
```

荷重 条件	部材 番号	部材荷重(1)端 Px1	Py1	Mz1	部材荷重(2)端 Px2	Py2	Mz2
----------	----------	-----------------	-----	-----	-----------------	-----	-----

```
[szr,~] = size(load2);  
for i = 1:szr  
    il = load2(i,1);  
    m = load2(i,2);  
    fprintf(' %3d %3d %10.3e %10.3e %10.3e %10.3e %10.3e %10.3e\n', ...  
            il, m, load2(i,3), load2(i,4), load2(i,5), load2(i,6), load2(i,7), load2(i,8))  
end
```

1	1	0.000e+00	0.000e+00	0.000e+00	0.000e+00	0.000e+00	0.000e+00
---	---	-----------	-----------	-----------	-----------	-----------	-----------

計算結果の表示

節点荷重と反力の表示

```
fprintf('\n ***** 節点荷重と反力 *****\n\n')
```

***** 節点荷重と反力 *****

```
for nl=1:nload  
fprintf(' 荷重条件番号: %3d\n', nl)  
fprintf(' \n')  
fprintf(' 節点          Px          Py          Mz\n')  
fprintf(' 番号          (%4s)          (%4s)          (%4s*%4s) \n', ...  
        unitf, unitf, unitf, unitf)  
for nn = 1:nnode  
    fprintf(' %3d %10.3e %10.3e %10.3e \n', ...  
            nn, load(nn*3-2,nl), load(nn*3-1,nl), load(nn*3,nl))  
end  
fprintf(' \n')  
end
```

荷重条件番号: 1

節点 番号	Px (kN)	Py (kN)	Mz (kN* mm)
1	5.669e+01	5.000e+01	0.000e+00
2	0.000e+00	-1.000e+02	0.000e+00
3	-5.669e+01	5.000e+01	0.000e+00

節点変位の表示

```
fprintf('\n ***** 節点変位 *****\n\n')
```

***** 節点変位 *****

```
for nl=1:nload  
fprintf(' 荷重条件番号: %3d \n', nl);  
fprintf(' \n')  
fprintf(' 節点          ux          uy          θ\n')  
fprintf(' 番号          (%4s)          (%4s)          (rad)\n', ...  
        unitl, unitl)  
for nn = 1:nnode  
    fprintf(' %3d %10.3e %10.3e %10.3e \n', ...  
            nn, displ(nn*3-2,nl), displ(nn*3-1,nl), displ(nn*3,nl))  
end  
fprintf(' \n')  
end
```

荷重条件番号: 1

節点 番号	ux (mm)	uy (mm)	θ (rad)
1	0.000e+00	0.000e+00	-6.652e-06
2	-3.543e-04	-5.402e-03	-4.018e-07
3	-7.087e-04	-8.036e-04	5.848e-06

部材端力の表示

```
fprintf(' ***** 部材端力 *****\n\n')
```

```
***** 部材端力 *****
```

```
fprintf([' 荷重 部材      Px1      Py1      Mz1      Px2      Py2      Mz2\n' ...
' 条件 番号      (%4s)      (%4s) (%4s*%4s) (%4s)      (%4s) (%4s*%4s)\n'], ...
        unitf, unitf, unitf, unitf, unitf, unitf, unitf, unitf)
```

```
荷重 部材      Px1      Py1      Mz1      Px2      Py2      Mz2
条件 番号      ( kN)      ( kN) ( kN* mm) ( kN)      ( kN) ( kN* mm)
```

```
for nl= 1:nl
    for nm = 1:nmemb
        fprintf(' %3d %3d %9.2e %9.2e %9.2e %9.2e %9.2e %9.2e\n', ...
                nl, nm,mforce(1,nm,nl),mforce(2,nm,nl), mforce(3,nm,nl),...
                mforce(4,nm,nl),mforce(5,nm,nl), mforce(6,nm,nl))
    end
    fprintf(' \n')
end
```

```
1 1 5.67e+01 5.00e+01 6.82e-12 -5.67e+01 -5.00e+01 5.00e+04
1 2 5.67e+01 -5.00e+01 -5.00e+04 -5.67e+01 5.00e+01 -7.28e-12
```

```
% -----
fprintf(1,' ***** 計算終了 *****\n\n')
```

```
***** 計算終了 *****
```

得られた応力は、軸力が -56.7kNで、曲げモーメントが中央の 50kNmであり 正しく求められている。

このライブスクリプト内で使用している関数の定義

```
% -----
```

```
function [k,T] = stiff(m,member,node,E,G)
% 部材剛性マトリックスと座標変換マトリックスの計算
% {p1} = [k11]*{d1} + [k12]*{d2}
% {p2} = [k21]*{d1} + [k22]*{d2}
% 上式により、部材座標系による部材端部の変位 {d1}, {d2}と材端部の応力 {p1}, {p2}
% の関係を表す部材の剛性方程式の係数
% k = [k11, k12; k21, k22] (6×6)
% を求める。
% また、
% {p'} = [T]{p} {d} = [T]{d'}
% となる、基準座標系と部材座標系の変換マトリクス[T]を求める。
```

```
n1      = member(m,2);    % 部材の (1)-端の節点番号
n2      = member(m,3);    % 部材の (2)-端の節点番号
ipin    = member(m,4);    % 部材の端部の接続条件
a       = member(m,5);    % 部材の断面積
ai      = member(m,6);    % 部材の断面 2 次モーメント
rgd1    = member(m,7);    % 部材の(1)-端の剛域長さ
rgd2    = member(m,8);    % 部材の(2)-端の剛域長さ
kappa   = member(m,9);    % 断面の形状係数 κ
x1      = node(n1,3);     % 部材の(1)-端の x 座標
x2      = node(n2,3);     % 部材の(2)-端の x 座標
y1      = node(n1,4);     % 部材の(1)-端の y 座標
y2      = node(n2,4);     % 部材の(2)-端の y 座標
delx    = x2 - x1;
dely    = y2 - y1;
```



```

al      = sqrt(delx * delx + dely * dely); % 部材長さ
cs      = delx/al;
sn      = dely/al;
ale     = al-rgd1-rgd2; % 剛域を除いた弾性部材の長さ
eal     = E*a/ale;
ei      = E*ai;

if kappa == 0
    aq = a;
    ss = 0;
elseif G == 0,
    aq = a;
    ss = 0;
else
    aq = a/kappa; % 有効せん断断面積
    ss = 1/(G*aq*ale);
end

% 柔性マトリクス [F] = [fb] + [fs] (静定基本部材は単純支持梁)
% [fb] 曲げ変形成分
% [fs] せん断変形成分
% indf 静定基本部材で変位を拘束していない自由度
% inds 静定基本部材で変位を拘束している自由度
% [He] 釣合マトリクス

% 部材の両端が連続
if ipin == 0
    indf = [4,3,6];
    inds = [1,2,5];
    fb = [1/eal,0,0; 0,ale/(3*ei),-ale/(6*ei); 0,-ale/(6*ei),ale/(3*ei)];
    fs = [0,0,0; 0,ss,ss; 0,ss,ss]; % せん断変形
    He = [1,0,0; 0,-1/ale,-1/ale; 0,1/ale,1/ale];
% (1)-端がピン、(2)-端が連続
elseif ipin == 1
    indf = [4,6];
    inds = [1,2,3,5];
    fb = [1/eal,0; 0,ale/(3*ei)];
    fs = [0,0; 0,ss];
    He = [1,0; 0,-1/ale; 0,0; 0,1/ale];
% (1)-端が連続、(2)-端がピン
elseif ipin == 2
    indf = [1,3];
    inds = [2,4,5,6];
    fb = [1/eal,0; 0,ale/(3*ei)];
    fs = [0,0; 0,ss];
    He = [0,-1/ale; 1,0; 0,1/ale; 0,0];
% 両端がピン (トラス部材)
elseif ipin == 3
    indf = 4;
    inds = [1,2,5,3,6];
    fb = [1/eal];
    fs = [0];
    He = [1; 0; 0; 0; 0];
end
F = fb + fs;

% 剛域を除いた弾性部材の剛性マトリクス [k]
k = zeros(6,6);
kff = inv(F);
k(indf,indf) = kff;
k(indf,inds) = -kff * He';
k(inds,indf) = -He * kff;
k(inds,inds) = He * kff * He';

% 剛域の端部の応力と弾性部材端部の応力の関係に関する釣り合いマトリクス [Hg]
Hg1 = [1 0 0; 0 1 0; 0 rgd1 1];
Hg2 = [1 0 0; 0 1 0; 0 -rgd2 1];
Hg = [Hg1, zeros(3,3); zeros(3,3), Hg2];

% 剛域を考慮した部材剛性マトリクス [K]

```

```
k = Hg * k * Hg';
```

```
% 座標変換マトリックス [T]
```

```
t = [cs -sn 0; sn cs 0; 0 0 1];
```

```
T = [t, zeros(3); zeros(3), t];
```

```
end
```