

**【知识点 8.5】 平面矩形薄板分析的 MATLAB 编程****【MATLAB 程序】 3 节点三角形单元的有限元分析程序(Triangle2D3Node)**

编写平面 3 节点三角形单元的单元刚度矩阵、单元组装、单元应力的计算程序。

解答：编写的平面 3 节点三角形单元的三个 MATLAB 函数如下。

Triangle2D3Node_Stiffness(E,NU,t,xi,yi,xj,yj,xm,ym,ID)

该函数计算单元的刚度矩阵，输入弹性模量 E，泊松比 NU，厚度 t，三个节点 i、j、m 的坐标 xi,yi,xj,yj,xm,ym，平面问题性质指示参数 ID(1 为平面应力，2 为平面应变)，输出单元刚度矩阵 k(6X6)。

Triangle2D3Node_Assembly(KK,k,i,j,m)

该函数进行单元刚度矩阵的组装，输入单元刚度矩阵 k，单元的节点编号 I、j、m，输出整体刚度矩阵 KK。

Triangle2D3Node_Stress(E,NU,xi,yi,xj,yj,xm,ym,u,ID)

该函数计算单元的应力，输入弹性模量 E，泊松比 NU，厚度 t，三个节点 i、j、m 的坐标 xi,yi,xj,yj,xm,ym，平面问题性质指示参数 ID(1 为平面应力，2 为平面应变)，单元的位移列阵 u(6X1)，输出单元的应力 stress，由于它为常应力单元，则单元的应力分量为 Sx,Sy,Sxy。

编写出具体实现以上每个函数的 MATLAB 程序如下。

```

%%%%%%%%%% Triangle2D3Node begin %%%%%%%%%%%
function k=Triangle2D3Node_Stiffness(E,NU,t,xi,yi,xj,yj,xm,ym,ID)
%该函数计算单元的刚度矩阵
%输入弹性模量 E，泊松比 NU，厚度 t
%输入三个节点 i、j、m 的坐标 xi,yi,xj,yj,xm,ym
%输入平面问题性质指示参数 ID(1 为平面应力，2 为平面应变)
%输出单元刚度矩阵 k(6X6)
%-----
A = (xi*(yj-ym) + xj*(ym-yi) + xm*(yi-yj))/2;
betai = yj-ym;
betaj = ym-yi;
betam = yi-yj;
gammai = xm-xj;
gammaj = xi-xm;
gammam = xj-xi;
B = [betai 0 betaj 0 betam 0 ;
      0 gammai 0 gammaj 0 gammam ;
      gammai betai gammaj betaj gammam betam]/(2*A);
if ID == 1
    D = (E/(1-NU*NU))*[1 NU 0 ; NU 1 0 ; 0 0 (1-NU)/2];
elseif ID == 2
    D = (E/(1+NU)/(1-2*NU))*[1-NU NU 0 ; NU 1-NU 0 ; 0 0 (1-2*NU)/2];
end
k= t*A*B'*D*B;
%%%%%%%%%%
function z = Triangle2D3Node_Assembly(KK,k,i,j,m)
%该函数进行单元刚度矩阵的组装
%输入单元刚度矩阵 k
%输入单元的节点编号 I、j、m
%输出整体刚度矩阵 KK
%-----
DOF(1)=2*i-1;

```



```

DOF(2)=2*i;
DOF(3)=2*j-1;
DOF(4)=2*j;
DOF(5)=2*m-1;
DOF(6)=2*m;
for n1=1:6
    for n2=1:6
        KK(DOF(n1),DOF(n2))= KK(DOF(n1),DOF(n2))+k(n1,n2);
    end
end
z=KK;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function stress=Triangle2D3Node_Stress(E,NU,xi,yi,xj,yj,xm,ym,u,ID)
%该函数计算单元的应力
%输入弹性模量 E, 泊松比 NU, 厚度 t
%输入三个节点 i、j、m 的坐标 xi,yi,xj,yj,xm,ym
%输入平面问题性质指示参数 ID(1 为平面应力, 2 为平面应变), 单元的位移列阵 u(6X1)
%输出单元的应力 stress(3X1), 由于它为常应力单元, 则单元的应力分量为 Sx,Sy,Sxy
%-----
A = (xi*(yj-ym) + xj*(ym-yi) + xm*(yi-yj))/2;
betai = yj-ym;
betaj = ym-yi;
betam = yi-yj;
gammai = xm-xj;
gammaj = xi-xm;
gammam = xj-xi;
B = [betai 0 betaj 0 betam 0 ;
      0 gammai 0 gammaj 0 gammam ;
      gammai betai gammaj betaj gammam betam]/(2*A);
if ID == 1
    D = (E/(1-NU*NU))*[1 NU 0 ; NU 1 0 ; 0 0 (1-NU)/2];
elseif ID == 2
    D = (E/(1+NU))/(1-2*NU))*[1-NU NU 0 ; NU 1-NU 0 ; 0 0 (1-2*NU)/2];
end
stress = D*B*u;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Triangle2D3Node %%% end %%%

```

【MATLAB 算例】 基于 3 节点三角形单元的矩形薄板分析

如图所示为一矩形薄平板，在右端部受集中力 $F = 100000\text{N}$ 作用，材料常数为：弹性模量 $E = 1 \times 10^7 \text{Pa}$ 、泊松比 $\mu = 1/3$ ，板的厚度为 $t = 0.1\text{m}$ ，试按平面应力问题计算各个节点位移及支座反力。

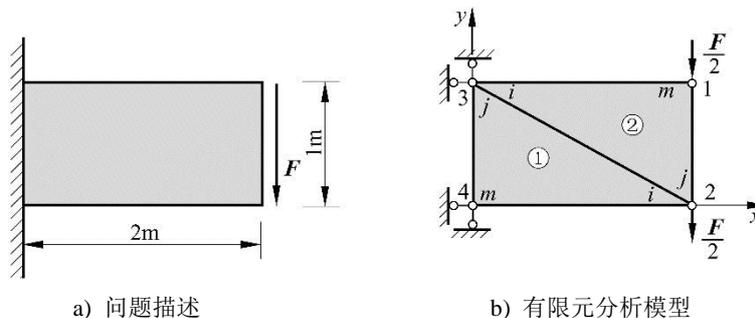


图 1 右端部受集中力作用的平面问题(高深梁)



解答：对该问题进行有限元分析的过程如下。

(1) 结构的离散化与编号

将结构离散为二个 3 节点三角形单元，单元编号及节点编号如图 1 所示。

(2) 计算各单元的刚度矩阵(以国际单位)

首先在 MATLAB 环境下，输入弹性模量 E、泊松比 NU，薄板厚度为 t，平面应力问题性质指示参数 ID，然后针对单元 1 和单元 2，分别调用两次函数 Triangle2D3Node_Stiffness，就可以得到单元的刚度矩阵 k1(6×6)和 k2(6×6)。

```
>> E=1e7;
>> NU=1/3;
>> t=0.1;
>> ID=1;
>> k1=Triangle2D3Node_Stiffness(E,NU,t,2,0,0,1,0,0,ID)
k1 = 1.0e+006 *
    0.2813     0         0     0.1875    -0.2813    -0.1875
         0     0.0938     0.1875         0    -0.1875    -0.0938
         0     0.1875     0.3750         0    -0.3750    -0.1875
    0.1875         0         0     1.1250    -0.1875    -1.1250
   -0.2813   -0.1875   -0.3750   -0.1875     0.6563     0.3750
   -0.1875   -0.0938   -0.1875   -1.1250     0.3750     1.2188
```

```
>>k2=Triangle2D3Node_Stiffness(E,NU,t,0,1,2,0,2,1,ID)
k2 = 1.0e+006 *
    0.2813     0         0     0.1875    -0.2813    -0.1875
         0     0.0938     0.1875         0    -0.1875    -0.0938
         0     0.1875     0.3750         0    -0.3750    -0.1875
    0.1875         0         0     1.1250    -0.1875    -1.1250
   -0.2813   -0.1875   -0.3750   -0.1875     0.6563     0.3750
   -0.1875   -0.0938   -0.1875   -1.1250     0.3750     1.2188
```

(3) 建立整体刚度方程

由于该结构共有 4 个节点，则总共的自由度数为 8，因此，结构总的刚度矩阵为 KK(8×8)，先对 KK 清零，然后两次调用函数 Triangle2D3Node_Assembly 进行刚度矩阵的组装。

```
>>KK = zeros(8,8);
>>KK=Triangle2D3Node_Assembly(KK,k1,2,3,4);
>>KK=Triangle2D3Node_Assembly(KK,k2,3,2,1)
KK = 1.0e+006 *
Columns 1 through 6
    0.6563    0.3750   -0.3750   -0.1875   -0.2813   -0.1875
    0.3750    1.2188   -0.1875   -1.1250   -0.1875   -0.0938
   -0.3750   -0.1875    0.6563         0         0         0.3750
   -0.1875   -1.1250         0     1.2188     0.3750         0
   -0.2813   -0.1875         0     0.3750     0.6563         0
   -0.1875   -0.0938     0.3750         0         0     1.2188
         0         0   -0.2813   -0.1875   -0.3750   -0.1875
         0         0   -0.1875   -0.0938   -0.1875   -1.1250

Columns 7 through 8
         0         0
         0         0
   -0.2813   -0.1875
```



```

-0.1875   -0.0938
-0.3750   -0.1875
-0.1875   -1.1250
 0.6563    0.3750
 0.3750    1.2188

```

(4) 边界条件的处理及刚度方程求解

由图 1(b) 可以看出，节点 3 和节点 4 的两个方向的位移将为零，即 $u_3 = 0, v_3 = 0, u_4 = 0, v_4 = 0$ 。因此，将针对节点 1 和节点 2 的位移进行求解，节点 1 和节点 2 的位移将对应 KK 矩阵中的前 4 行和前 4 列，则需从 $KK(8 \times 8)$ 中提出，置给 k ，然后生成对应的载荷列阵 p ，再采用高斯消去法进行求解，注意：MATLAB 中的反斜线符号“\”就是采用高斯消去法。

```

>>k=KK(1:4,1:4)
k = 1.0e+006 *
    0.6563    0.3750   -0.3750   -0.1875
    0.3750    1.2188   -0.1875   -1.1250
   -0.3750   -0.1875    0.6563     0
   -0.1875   -1.1250     0     1.2188

>>p=[0;-5000;0;-5000]
p =      0      -5000         0      -5000   [这里将列排成了一行，以节省篇幅]

>>u=k\p
u =  0.0188   -0.0899   -0.0150   -0.0842   [这里将列排成了一行，以节省篇幅]

```

由此可以看出，所求得的结果为： $u_1 = 0.0188, v_1 = -0.0899, u_2 = -0.0151, v_2 = -0.0842$ 。

(5) 支反力的计算

在得到整个结构的节点位移后，由原整体刚度方程就可以计算出对应的支反力；先将上面得到的位移结果与位移边界条件的节点位移进行组合(注意位置关系)，可以得到整体的位移列阵 $U(8 \times 1)$ ，再代回原整体刚度方程，计算出所有的节点力 $P(8 \times 1)$ ，按式对应关系就可以找到对应的支反力。

```

>>U=[u;0;0;0;0];
>>P=KK*U
P = 1.0e+004 *
   -0.0000  -0.5000  0  -0.5000  -2.0000  -0.0702  2.0000  1.0702   [这里将列排成了一行]

```

由式 (4-179) 的对应关系，可以得到对应的支反力为 $R_{x3} = -20000, R_{y3} = -702, R_{x4} = 20000, R_{y4} = 10702$ 。

(6) 各单元的应力计算

先从整体位移列阵 $U(8 \times 1)$ 中提取出单元的位移列阵，然后，调用计算单元应力的函数 `Triangle2D3Node_Stress`，就可以得到各个单元的应力分量。

```

>>u1=[U(3);U(4);U(5);U(6);U(7);U(8)]
u1 =  -0.0150   -0.0842     0     0     0     0

>>stress1=Triangle2D3Node_Stress(E,NU,2,0,0,1,0,0,u1,ID)
stress1 = 1.0e+005 *
   -0.8419   -0.2806   -1.5791   [这里将列排成了一行，以节省篇幅]

>>u2=[U(5);U(6);U(3);U(4);U(1);U(2)]

```



u2 = 0 0 -0.0150 -0.0842 0.0188 -0.0899 [这里将列排成了一行，以节省篇幅]

```
>>stress2=Triangle2D3Node_Stress(E,NU,0,1,2,0,2,1,u2,ID)
```

```
stress2 =1.0e+004 *
```

```
8.4187 -2.8953 -4.2094 [这里将列排成了一行，以节省篇幅]
```

可以看出：计算得到的单元 1 的应力分量为 $\sigma_x = -84190\text{Pa}$, $\sigma_y = -28060\text{Pa}$, $\tau_{xy} = -157910\text{Pa}$ ；单元 2 的应力分量为 $\sigma_x = 84187\text{Pa}$, $\sigma_y = -28953\text{Pa}$, $\tau_{xy} = -42094\text{Pa}$ 。