【知识点 9.5】典型空间问题的 MATLAB 编程

【MATLAB 程序】 4 节点四面体单元的有限元分析程序(Tetrahedron3D4Node)

编写空间 4 节点四面体单元的单元刚度矩阵、单元组装、单元应力的计算程序。

解答：编写的空间 4 节点四面体单元的三个 MATLAB 函数如下。

Tetrahedron3D4Node_Stiffness(E,NU,xi,yi,zi,xj,yj,zj,xm,ym,zm,xn,yn,zn)

该函数计算单元的刚度矩阵，输入弹性模量 E，泊松比 NU，4 个节点 i、j、m、n 的坐标 xi,yi,zi,xj,yj,zj,xm,ym,zm,xn,yn,zn，输出单元刚度矩阵 k(12X12)。

Tetrahedron3D4Node_Assembly(KK,k,i,j,m,n)

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、n，输出整体刚度矩阵 KK。

Tetrahedron3D4Node_Stress(E,NU,xi,yi,zi,xj,yj,zj,xm,ym,zm,xn,yn,zn,u)

该函数计算单元的应力，输入弹性模量 E，泊松比 NU，4 个节点 i、j、m、n 的坐标 xi,yi,zi,xj,yj,zj,xm,ym,zm,xn,yn,zn，单元的位移列阵 u(12X1)，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,Sz,Sxy,Syz,Szx。

基于前面的基本公式，可以编写出具体实现以上每个函数的 MATLAB 程序如下。

```
%%%%%%%%%% Tetrahedron3D4Node %%% begin  %%%%%%%%%%
function k = Tetrahedron3D4Node_Stiffness(E,NU,x1,y1,z1,x2,y2,z2,x3,y3,z3,x4,y4,z4)
%该函数计算单元的刚度矩阵
%输入弹性模量 E，泊松比 NU
%输入 4 个节点 i、j、m、n 的坐标 xi,yi,zi,xj,yj,zj,xm,ym,zm,xn,yn,zn
%输出单元刚度矩阵 k(12X12)
%---------------------------------------------------------------
%以下数组与书中的对应关系
% betai - bi
% gammai  - ci
% deltai - di
% i = 1:4
xyz = [1 x1 y1 z1 ; 1 x2 y2 z2 ; 1 x3 y3 z3 ; 1 x4 y4 z4];
V = det(xyz)/6;
mbeta1 = [1 y2 z2 ; 1 y3 z3 ; 1 y4 z4];
mbeta2 = [1 y1 z1 ; 1 y3 z3 ; 1 y4 z4];
mbeta3 = [1 y1 z1 ; 1 y2 z2 ; 1 y4 z4];
mbeta4 = [1 y1 z1 ; 1 y2 z2 ; 1 y3 z3];
mgamma1  = [1 x2 z2 ; 1 x3 z3 ; 1 x4 z4];
mgamma2  = [1 x1 z1 ; 1 x3 z3 ; 1 x4 z4];
mgamma3  = [1 x1 z1 ; 1 x2 z2 ; 1 x4 z4];
mgamma4  = [1 x1 z1 ; 1 x2 z2 ; 1 x3 z3];
mdelta1 = [1 x2 y2 ; 1 x3 y3 ; 1 x4 y4];
mdelta2 = [1 x1 y1 ; 1 x3 y3 ; 1 x4 y4];
mdelta3 = [1 x1 y1 ; 1 x2 y2 ; 1 x4 y4];
mdelta4 = [1 x1 y1 ; 1 x2 y2 ; 1 x3 y3];
beta1 = -1*det(mbeta1);
beta2 = det(mbeta2);
beta3 = -1*det(mbeta3);
beta4 = det(mbeta4);
gamma1  = det(mgamma1);
gamma2  = -1*det(mgamma2);
gamma3  = det(mgamma3);
gamma4  = -1*det(mgamma4);
```

```
delta1 = -1*det(mdelta1);
delta2 = det(mdelta2);
delta3 = -1*det(mdelta3);
delta4 = det(mdelta4);
B1 = [beta1 0 0 ; 0 gamma1  0 ; 0 0 delta1 ;
    gamma1  beta1 0 ; 0 delta1 gamma1  ; delta1 0 beta1];
B2 = [beta2 0 0 ; 0 gamma2  0 ; 0 0 delta2 ;
    gamma2  beta2 0 ; 0 delta2 gamma2  ; delta2 0 beta2];
B3 = [beta3 0 0 ; 0 gamma3  0 ; 0 0 delta3 ;
    gamma3  beta3 0 ; 0 delta3 gamma3  ; delta3 0 beta3];
B4 = [beta4 0 0 ; 0 gamma4  0 ; 0 0 delta4 ;
    gamma4  beta4 0 ; 0 delta4 gamma4  ; delta4 0 beta4];
B = [B1 B2 B3 B4]/(6*V);
D = (E/((1+NU)*(1-2*NU)))*[1-NU NU NU 0 0 0 ; NU 1-NU NU 0 0 0 ; NU NU 1-NU 0 0 0 ;
    0 0 0 (1-2*NU)/2 0 0 ; 0 0 0 0 (1-2*NU)/2 0 ; 0 0 0 0 0 (1-2*NU)/2];
y = abs(V)*B'*D*B;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function y = Tetrahedron3D4Node_Assembly(KK,k,i,j,m,n)
%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、n
%输出整体刚度矩阵 KK
%---------------------------------------------------------------
DOF = [3*i-2:3*i,3*j-2:3*j,3*m-2:3*m,3*n-2:3*n];
for n1=1:12
    for n2=1:12
        KK(DOF(n1),DOF(n2))= KK(DOF(n1),DOF(n2))+k(n1,n2);
    end
end
y=KK;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function y = Tetrahedron3D4Node_Stress(E,NU,x1,y1,z1,x2,y2,z2,x3,y3,z3,x4,y4,z4,u)
%该函数计算单元的应力
%输入弹性模量 E，泊松比 NU
%输入 4 个节点 i、j、m、n 的坐标 xi,yi,zi,xj,yj,zj,xm,ym,zm,xn,yn,zn
%输入单元的位移列阵 u(12X1)
%输出单元的应力 stress(6X1)
%由于它为常应力单元，应力分量为 Sx,Sy,Sz,Sxy,Syz,Szx
%---------------------------------------------------------------
xyz = [1 x1 y1 z1 ; 1 x2 y2 z2 ; 1 x3 y3 z3 ; 1 x4 y4 z4];
V = det(xyz)/6;
mbeta1 = [1 y2 z2 ; 1 y3 z3 ; 1 y4 z4];
mbeta2 = [1 y1 z1 ; 1 y3 z3 ; 1 y4 z4];
mbeta3 = [1 y1 z1 ; 1 y2 z2 ; 1 y4 z4];
mbeta4 = [1 y1 z1 ; 1 y2 z2 ; 1 y3 z3];
mgamma1  = [1 x2 z2 ; 1 x3 z3 ; 1 x4 z4];
mgamma2  = [1 x1 z1 ; 1 x3 z3 ; 1 x4 z4];
mgamma3  = [1 x1 z1 ; 1 x2 z2 ; 1 x4 z4];
mgamma4  = [1 x1 z1 ; 1 x2 z2 ; 1 x3 z3];
mdelta1 = [1 x2 y2 ; 1 x3 y3 ; 1 x4 y4];
mdelta2 = [1 x1 y1 ; 1 x3 y3 ; 1 x4 y4];
mdelta3 = [1 x1 y1 ; 1 x2 y2 ; 1 x4 y4];
mdelta4 = [1 x1 y1 ; 1 x2 y2 ; 1 x3 y3];
beta1 = -1*det(mbeta1);
beta2 = det(mbeta2);
```

```
beta3 = -1*det(mbeta3);
beta4 = det(mbeta4);
gamma1  = det(mgamma1);
gamma2  = -1*det(mgamma2);
gamma3  = det(mgamma3);
gamma4  = -1*det(mgamma4);
delta1 = -1*det(mdelta1);
delta2 = det(mdelta2);
delta3 = -1*det(mdelta3);
delta4 = det(mdelta4);
B1 = [beta1 0 0 ; 0 gamma1  0 ; 0 0 delta1 ;
    gamma1  beta1 0 ; 0 delta1 gamma1  ; delta1 0 beta1];
B2 = [beta2 0 0 ; 0 gamma2  0 ; 0 0 delta2 ;
    gamma2  beta2 0 ; 0 delta2 gamma2  ; delta2 0 beta2];
B3 = [beta3 0 0 ; 0 gamma3  0 ; 0 0 delta3 ;
    gamma3  beta3 0 ; 0 delta3 gamma3  ; delta3 0 beta3];
B4 = [beta4 0 0 ; 0 gamma4  0 ; 0 0 delta4 ;
    gamma4  beta4 0 ; 0 delta4 gamma4  ; delta4 0 beta4];
B = [B1 B2 B3 B4]/(6*V);
D = (E/((1+NU)*(1-2*NU)))*[1-NU NU NU 0 0 0 ; NU 1-NU NU 0 0 0 ; NU NU 1-NU 0 0 0 ;
    0 0 0 (1-2*NU)/2 0 0 ; 0 0 0 0 (1-2*NU)/2 0 ; 0 0 0 0 0 (1-2*NU)/2];
y = D*B*u;
%%%%%%%%%%% Tetrahedron3D4Node %%% end %%%%%%%%%%
```

【MATLAB 算例】 基于 4 节点四面体单元的空间块体分析(Tetrahedron3D4Node)

如图 1 所示的一个块体，在右端面上端点受集中力 F 作用。基于 MATLAB 平台，计算各个节点位移、支反力以及单元的应力。取相关参数为：$E=1\times10^{10}$ Pa, μ=0.25，F=1×10^{5}N。

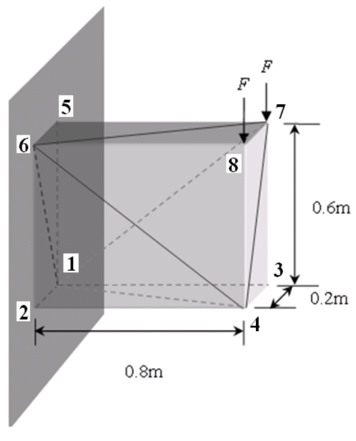

图 1 一个空间块体的分析

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 5 个 4 节点四面体单元，单元编号及节点编号和坐标如图 4-22 所示，连接关系见表 4-8，节点的坐标见表 4-9。

表 4-8 单元连接关系

单元号	节点号
1	1　4　2　6
2	1　4　3　7
3	6　7　5　1
4	6　7　8　4
5	1　4　6　7

表 4-9 节点的坐标

节点	节点坐标(m)		
	x	y	z
1	0	0	0
2	0.2	0	0
3	0	0.8	0
4	0.2	0.8	0
5	0	0	0.6
6	0.2	0	0.6
7	0	0.8	0.6
8	0.2	0.8	0.6

节点位移列阵

$$\mathbf{q}=\begin{bmatrix}u_1 & v_1 & w_1 & u_2 & v_2 & w_2 & \cdots & u_8 & v_8\end{bmatrix}^T \qquad (4\text{-}190)$$

节点外载列阵

$$\mathbf{F}=\begin{bmatrix}0 & 0 & \mathbf{F}_3^T & \mathbf{F}_4^T & 0 & 0 & \mathbf{F}_7^T & \mathbf{F}_8^T\end{bmatrix}^T \qquad (4\text{-}191)$$

其中

$$\mathbf{F}_3=\mathbf{F}_4=\begin{bmatrix}0\\0\\0\end{bmatrix} \qquad \mathbf{F}_7=\mathbf{F}_8=\begin{bmatrix}0\\0\\-1\times10^5 N\end{bmatrix}$$

约束的支反力列阵

$$\mathbf{R}=\begin{bmatrix}\mathbf{R}_1^T & \mathbf{R}_2^T & 0 & 0 & \mathbf{R}_5^T & \mathbf{R}_6^T & 0 & 0\end{bmatrix}^T \qquad (4\text{-}192$$

其中

$$\mathbf{R}_1=\begin{bmatrix}R_{1x}\\R_{1y}\\R_{1z}\end{bmatrix}\ \mathbf{R}_2=\begin{bmatrix}R_{2x}\\R_{2y}\\R_{2z}\end{bmatrix}\ \mathbf{R}_5=\begin{bmatrix}R_{5x}\\R_{5y}\\R_{5z}\end{bmatrix}\ \mathbf{R}_6=\begin{bmatrix}R_{6x}\\R_{6y}\\R_{6z}\end{bmatrix}$$

总的节点载荷列阵

$$\mathbf{P}=\mathbf{F}+\mathbf{R}=\mathbf{R}=\begin{bmatrix}\mathbf{R}_1^T & \mathbf{R}_2^T & \mathbf{F}_3^T & \mathbf{F}_4^T & \mathbf{R}_5^T & \mathbf{R}_6^T & \mathbf{F}_7^T & \mathbf{F}_8^T\end{bmatrix}^T \qquad (4\text{-}193)$$

（2）计算各单元的刚度矩阵(以国际标准单位)

首先在 MATLAB 环境下，输入弹性模量 E、泊松比 NU，然后针对单元 1 和单元 2，分别调用 5 次函数 Tetrahedron3D4Node_Stiffness，就可以得到单元的刚度矩阵 k1(6×6) ～ k5(6×6)。

```
>> E=1e10;
>> NU=0.25;
>> k1 = Tetrahedron3D4Node_Stiffness(E,NU,0,0,0,0.2,0.8,0,0.2,0,0,0.2,0,0.6);
>> k2 = Tetrahedron3D4Node_Stiffness(E,NU,0,0,0,0.2,0.8,0,0,0.8,0,0,0.8,0.6);
>> k3 = Tetrahedron3D4Node_Stiffness(E,NU,0.2,0,0.6,0,0.8,0.6,0,0,0.6,0,0,0);
>> k4=Tetrahedron3D4Node_Stiffness(E,NU,0.2,0,0.6,0,0.8,0.6,0.2,0.8,0.6,0.2,0.8,0);
>> k5 = Tetrahedron3D4Node_Stiffness(E,NU,0,0,0,0.2,0.8,0,0.2,0,0.6,0,0.8,0.6);
```

（3） 建立整体刚度方程

由于该结构共有 8 个节点，则总共的自由度数为 24，因此，结构总的刚度矩阵为 KK(24×24)，先对 KK 清零，然后 5 次调用函数 Tetrahedron3D4Node_Assembly 进行刚度矩阵的组装。

```
>>KK = zeros(24);
>> KK = Tetrahedron3D4Node_Assembly(KK,k1,1,4,2,6);
>> KK = Tetrahedron3D4Node_Assembly(KK,k2,1,4,3,7);
>> KK = Tetrahedron3D4Node_Assembly(KK,k3,6,7,5,1);
>> KK = Tetrahedron3D4Node_Assembly(KK,k4,6,7,8,4);
>> KK = Tetrahedron3D4Node_Assembly(KK,k5,1,4,6,7);
```

（4） 边界条件的处理及刚度方程求解

由图 4-22 可以看出，节点 1、2、5、6 上三个方向的位移将为零，即 $u_1 = v_1 = w_1 = u_2 = v_2 = w_2 = u_5 = v_5 = w_5 = u_6 = v_6 = 0$。因此，将针对节点 3、4、7、8 的位移进行求解，节点 1、2、5、6 的位移将对应 KK 矩阵中的第 1 至 6 行，第 13 至 18 行和第 1 至 6 列，第 13 至 18 列，需从 KK(24×24)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([7:12,19:24],[7:12,19:24]);
>>p=[0,0,0,0,0,0,0,0,-1e5,0,0,-1e5]'
>>u=k\p
u =    1.0e-003 *
      0.1249    -0.0485    -0.4024    0.1343    -0.0715    -0.4031          [将列排成行]
      0.1314    0.0858    -0.4460    0.1353    0.0681    -0.4742          [将列排成行]
```

由此可以看出，所求得的位移结果见表 4-10。

表 4-10　空间块体的节点位移计算结果

$u_3 = 0.1249 \times 10^{-3}$	$u_7 = 0.1314 \times 10^{-3}$
$v_3 = -0.0485 \times 10^{-3}$	$v_7 = 0.0858 \times 10^{-3}$
$w_3 = -0.4024 \times 10^{-3}$	$w_7 = -0.4460 \times 10^{-3}$
$u_4 = 0.1343 \times 10^{-3}$	$u_8 = 0.1353 \times 10^{-3}$
$v_4 = -0.0715 \times 10^{-3}$	$v_8 = 0.0681 \times 10^{-3}$
$w_4 = -0.4031 \times 10^{-3}$	$w_8 = -0.4742 \times 10^{-3}$

（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(24×1)，再代回原整体刚度方程，计算出所有的节点力 P(24×1)，按式(4-192)的对应关系就可以找到对应的支反力。

```
>>U=[zeros(6,1);u([1:6]);zeros(6,1);u(7:12)];
>>P=KK*U
P = 1.0e+005 *
    0.3372    1.3774    0.1904   -0.4202    1.2892    0.4984      [将列排成行]
   -0.0000    0.0000    0.0000   -0.0000   -0.0000   -0.0000      [将列排成行]
   -0.4745   -1.3774    0.5604    0.5575   -1.2892    0.7509      [将列排成行]
   -0.0000   -0.0000   -1.0000   -0.0000    0.0000   -1.0000      [将列排成行]
```

由式(4-193)的对应关系，可以得到对应的支反力见表 4-11。

表 4-11　空间块体的支反力计算结果

$R_{1x}=0.3372\times10^5 N$	$R_{5x}=-0.4745\times10^5 N$
$R_{1y}=1.3774\times10^5 N$	$R_{5y}=-1.3774\times10^5 N$
$R_{1z}=0.1904\times10^5 N$	$R_{5z}=0.5604\times10^5 N$
$R_{2x}=-0.4202\times10^5 N$	$R_{6x}=0.5575\times10^5 N$
$R_{2y}=1.2892\times10^5 N$	$R_{6y}=-1.2892\times10^5 N$
$R_{2z}=0.4984\times10^5 N$	$R_{6z}=0.7509\times10^5 N$

（6）各单元的应力计算

先从整体位移列阵 U(24×1)中提取出单元的位移列阵，然后，调用计算单元应力的函数 Tetrahedron3D4Node_Stress，就可以得到各个单元的应力分量。

```
>>u1=[U(1:3);U(10:12);U(4:6);U(16:18)];
>>stress1 = Tetrahedron3D4Node_Stress(E,NU,0,0,0,0.2,0.8,0,0.2,0,0,0.2,0,0.6,u1)
stress1 = 1.0e+006 *
   -0.3574   -1.0721   -0.3574    0.6717   -2.0155         0      [将列排成行]
>>u2=[U(1:3);U(10:12);U(7:9);U(19:21)];
>> stress2= Tetrahedron3D4Node_Stress(E,NU,0,0,0,0.2,0.8,0,0,0.8,0,0,0.8,0.6,u2)
stress2 = 1.0e+006 *
    0.0314   -0.8298   -0.9260    0.1649   -1.1170    0.0294      [将列排成行]
>> u3=[U(16:21);U(13:15);U(1:3)];
>> stress3=Tetrahedron3D4Node_Stress(E,NU,0.2,0,0.6,0,0.8,0.6,0,0,0.6,0,0,0,u3)
stress3 = 1.0e+006 *
    0.4289    1.2867    0.4289    0.6568   -2.2301         0      [将列排成行]
>> u4=[U(16:21);U(22:24);U(10:12)];
>> stress4=Tetrahedron3D4Node_Stress(E,NU,0.2,0,0.6,0,0.8,0.6,0.2,0.8,0.6,0.2,0.8,0,u4)
stress4 = 1.0e+006 *
    0.1046    0.6272   -1.0012    0.3233   -1.4402   -0.5562      [将列排成行]
>> u5=[U(1:3);U(10:12);U(16:21)];
>> stress5=Tetrahedron3D4Node_Stress(E,NU,0,0,0,0.2,0.8,0,0.2,0,0.6,0,0.8,0.6,u5)
stress5 = 1.0e+006 *
   -0.0179   -0.0060   -0.3636   -0.9083   -1.5986    0.4192      [将列排成行]
```

各个单元应力分量的计算结果列在表 4-12 中。

表 4-12　空间块体的各个单元应力分量的计算结果

1 号单元	$\sigma_x = -0.3574MPa$	$\sigma_y = -1.0721MPa$	$\sigma_z = -0.3574MPa$
	$\tau_{xy} = 0.6717MPa$	$\tau_{yz} = -2.0155MPa$	$\tau_{zx} = 0MPa$
2 号单元	$\sigma_x = 0.0314MPa$	$\sigma_y = -0.8298MPa$	$\sigma_z = -0.9260MPa$
	$\tau_{xy} = 0.1649MPa$	$\tau_{yz} = -1.1170MPa$	$\tau_{zx} = 0.0294MPa$
3 号单元	$\sigma_x = 0.4289MPa$	$\sigma_y = 1.2867MPa$	$\sigma_z = 0.4289MPa$
	$\tau_{xy} = 0.6568MPa$	$\tau_{yz} = -2.2301MPa$	$\tau_{zx} = 0MPa$
4 号单元	$\sigma_x = 0.1046MPa$	$\sigma_y = 0.6272MPa$	$\sigma_z = -1.0012MPa$
	$\tau_{xy} = 0.3233MPa$	$\tau_{yz} = -1.4402MPa$	$\tau_{zx} = -0.5562MPa$
5 号单元	$\sigma_x = -0.0179MPa$	$\sigma_y = -0.0060MPa$	$\sigma_z = -0.3636MPa$
	$\tau_{xy} = -0.9083MPa$	$\tau_{yz} = -1.5986MPa$	$\tau_{zx} = 0.4192MPa$