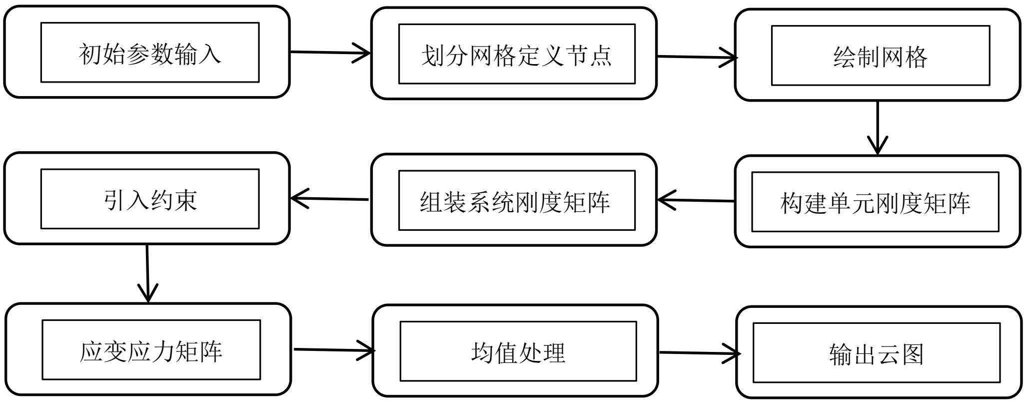
有限元课程作业

1.算例描述

一长方体，长为 2m，宽为 1m，高为 0.5m，弹性模量为 2.1e5MPa，泊松比为 0.3。其底面固定，在顶面中心施加大小为 1000N 的力，划分网格为 x*y*z，其中 x=y=z=10 得到 1000 个单元。求解得到网格划分图，位移变形图，位移云图，应力云图。

2.程序说明



根据上述的算例，绘制了包括网格划分图，位移变形图，位移云图，应力云图，以直观展示算例的模拟结果。具体节点的应力应变数据在 txt 文件中给出。在此基础上，采用了强大的有限元分析软件 Abaqus 对同一算例进行了额外计算，并在本文档内详细比较了使用 Abaqus 和 Matlab 两种工具得出的计算结果，以验证算例的准确性和可靠性。

3.Matlab 程序（主程序）

```
clear;
clc;
%参数
lx=2;%长方体，长 2，宽 1，高 0.5，
ly=1;
lz=0.5;
E=2.1e5;%弹性模量
mu=0.3;%泊松比
ng=2;%高斯积分点数
% 获得高斯积分点坐标值和加权系数
[pi,wi]=Gauss(ng);
%弹性矩阵
D=E/((1+mu)*(1-2*mu))*...
    [1-mu mu mu 0 0 0
    mu 1-mu mu 0 0 0
    mu mu 1-mu 0 0 0
```

```
    0 0 0 (1-2*mu)/2 0 0
    0 0 0 0 (1-2*mu)/2 0
    0 0 0 0 0 (1-2*mu)/2];
%划分节点与单元
coord=zeros(11*11*11,3);
e=0;
for z=0:10
    for y=0:10
        for x=0:10
            e=e+1;
            coord(e,:)=[x,y,z];
        end
    end
end
ele=zeros(10*10*10,8);
for z=0:9
    for y=0:9
        for x=0:9
            n=x+1+10*y+10*10*z;
            ele(n,:)=[x+1+11*y+11*11*z ...
                      x+1+1+11*y+11*11*z ...
                      x+1+1+11*(y+1)+11*11*z ...
                      x+1+11*(y+1)+11*11*z ...
                      x+1+11*y+11*11*(z+1) ...
                      x+1+1+11*y+11*11*(z+1) ...
                      x+1+1+11*(y+1)+11*11*(z+1) ...
                      x+1+11*(y+1)+11*11*(z+1)];
        end
    end
end
nc=size(coord,1);%节点数量
ne=size(ele,1);%单元数量
nne=size(ele,2);%单元节点数
nd=3;%维度
ad=nc*nd;%整体自由度
ed=nne*nd;%每个单元的自由度
%初始化矩阵
K=sparse(ad,ad);%总体刚度矩阵
stress=zeros(nc,7);%应力矩阵
nnode=zeros(nc,1);%节点次数矩阵
mstress=zeros(nne,1);%Mises 应力矩阵
%初始化坐标
X=zeros(nne,1);
Y=zeros(nne,1);
```

```
Z=zeros(nne,1);
%计算总体刚度矩阵
for ine=1:ne
    ndi=ele(ine,:);
    for i=1:nne
        X(i)=coord(ndi(i),1);
        Y(i)=coord(ndi(i),2);
        Z(i)=coord(ndi(i),3);
    end
    k=zeros(ed,ed);
    for iz=1:ng
        z=pi(iz);
        wz=wi(iz);
        for iy=1:ng
            y=pi(iy);
            wy=wi(iy);
            for ix=1:ng
                x=pi(ix);
                wx=wi(ix);
                [dndr,dnds,dndt]=Shape(x,y,z);%构建型函数
                J=Jacobain(nne,dndr,dnds,dndt,X,Y,Z);%雅可比矩阵
                B=Strain(J,nne,dndr,dnds,dndt);%单元应变矩阵
                k=k+(B')*D*B*wx*wy*wz*det(J);%计算单元刚度矩阵
            end
        end
    end
    K=Assemble(K,ndi,k,nne,nd);
end
%添加边界条件，在顶面中心施加一个大小均为 1000N 力，底面固定
F=zeros(ad,1);
F(200)=1000;
bc=[];
for ibc=1:nc
    if coord(ibc,2)==0
        bc=[bc ibc];
    end
end
nb=size(bc,2);
K=Boundary(bc,nb,K,nd);
displacement=K\F;
%计算节点次数矩阵
for ini=1:ne
    for inj=1:nne
        nnode(ele(ini,inj))=nnode(ele(ini,inj))+1;%组装单元
```

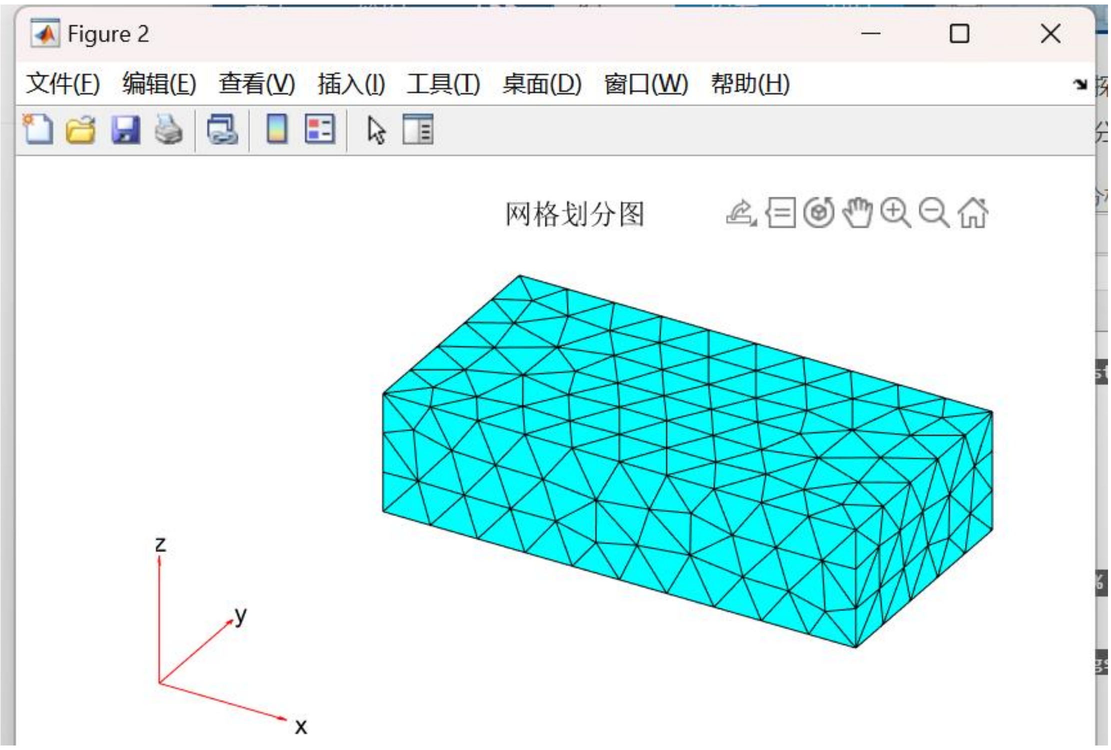
```
    end
end
%计算应力矩阵
for ine=1:ne
    ndi=ele(ine,:);
    for i=1:nne
        X(i)=coord(ndi(i),1);
        Y(i)=coord(ndi(i),2);
        Z(i)=coord(ndi(i),3);
    end
    stress2=[];
    for iz=1:ng
        z=pi(iz);
        for iy=1:ng
            y=pi(iy);
            for ix=1:ng
                x=pi(ix);
                [dndr,dnds,dndt]=Shape(x,y,z);%构建型函数
                J=Jacobain(nne,dndr,dnds,dndt,X,Y,Z);%雅可比矩阵
                B=Strain(J,nne,dndr,dnds,dndt);%单元应变矩阵
                stress1=Stress(displacement,nd,ndi,B,D);%应力矩阵
                stress2=[stress2 stress1];
            end
        end
    end
    stress2=[stress2(:,5),stress2(:,6),stress2(:,8),stress2(:,7),...
        stress2(:,1),stress2(:,2),stress2(:,4),stress2(:,3)];
    h=sqrt(3);
    R=[Recovery(-h,-h, h)
       Recovery( h,-h, h)
       Recovery( h, h, h)
       Recovery(-h, h, h)
       Recovery(-h,-h,-h)
       Recovery( h,-h,-h)
       Recovery( h, h,-h)
       Recovery(-h, h,-h)];%应力恢复矩阵
    estress=R*transpose(stress2);
    for mi=1:nne
        s1=estress(mi,1);
        s2=estress(mi,2);
        s3=estress(mi,3);
        mstress(mi)=(((s1-s2)^2+(s2-s3)^2+(s3-s1)^2)/2)^0.5;
    end
    estress=[estress,mstress];
```

```
    for is=1:nne
        stress(ndi(is),:)=stress(ndi(is),:)+estress(is,:);
    end
end
for inode=1:nc
    an=nnode(inode);
    stress(inode,:)=stress(inode,:)/an;
end
xd=zeros(nc,1);
yd=zeros(nc,1);
zd=zeros(nc,1);

xd0=zeros(nc,1);
yd0=zeros(nc,1);
zd0=zeros(nc,1);
%每一个节点 x,y,z 三个方向的应变
for i=1:nc
    xd(i)=displacement(3*i-2);
    yd(i)=displacement(3*i-1);
    zd(i)=displacement(3*i);
end
%%网格划分图
figure(1)
for iel=1:ne
    for i=1:nne
        ndi(i)=ele(iel,i);
        xcoord(i)=coord(ndi(i),1);
        ycoord(i)=coord(ndi(i),2);
        zcoord(i)=coord(ndi(i),3);
    end

plot3([xcoord(1);xcoord(2);xcoord(3);xcoord(4);xcoord(1);xcoord(5);xcoord(6);xcoord(7);xcoord(8);xcoord(5);xcoord(8);xcoord(4);xcoord(3);xcoord(7);xcoord(3);xcoord(2);xcoord(6)],[ycoord(1);ycoord(2);ycoord(3);ycoord(4);ycoord(1);ycoord(5);ycoord(6);ycoord(7);ycoord(8);ycoord(5);ycoord(8);ycoord(4);ycoord(3);ycoord(7);ycoord(3);ycoord(2);ycoord(6)],[zcoord(1);zcoord(2);zcoord(3);zcoord(4);zcoord(1);zcoord(5);zcoord(6);zcoord(7);zcoord(8);zcoord(5);zcoord(8);zcoord(4);zcoord(3);zcoord(7);zcoord(3);zcoord(2);zcoord(6)],'b-');
    title('网格划分图');
    hold on
end
%%位移变形图
figure(2)
```

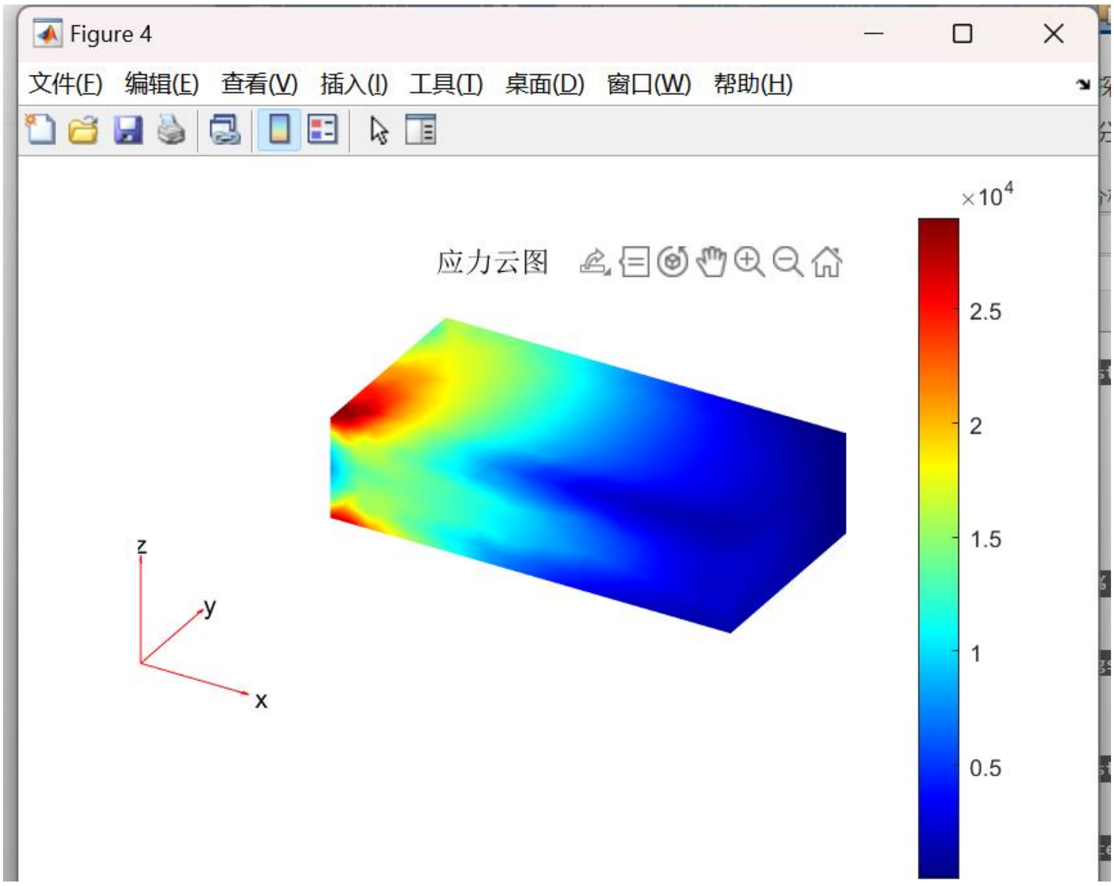
```
for iel=1:ne
    for i=1:nne
        ndi(i)=ele(iel,i);
        xcoord(i)=coord(ndi(i),1);
        ycoord(i)=coord(ndi(i),2);
        zcoord(i)=coord(ndi(i),3);

        xd0(i)=50*xd(ndi(i));
        yd0(i)=50*yd(ndi(i));
        zd0(i)=50*zd(ndi(i));
    end

plot3([xcoord(1)+xd0(1);xcoord(2)+xd0(2);xcoord(3)+xd0(3);xcoord(4)+xd0(4);xcoord(1)+xd0(1);xcoord(5)+xd0(5);xcoord(6)+xd0(6);xcoord(7)+xd0(7);xcoord(8)+xd0(8);xcoord(5)+xd0(5);xcoord(8)+xd0(8);xcoord(4)+xd0(4);xcoord(3)+xd0(3);xcoord(7)+xd0(7);xcoord(3)+xd0(3);xcoord(2)+xd0(2);xcoord(6)+xd0(6)],[ycoord(1)+yd0(1);ycoord(2)+yd0(2);ycoord(3)+yd0(3);ycoord(4)+yd0(4);ycoord(1)+yd0(1);ycoord(5)+yd0(5);ycoord(6)+yd0(6);ycoord(7)+yd0(7);ycoord(8)+yd0(8);ycoord(5)+yd0(5);ycoord(8)+yd0(8);ycoord(4)+yd0(4);ycoord(3)+yd0(3);ycoord(7)+yd0(7);ycoord(3)+yd0(3);ycoord(2)+yd0(2);ycoord(6)+yd0(6)],[zcoord(1)+zd0(1);zcoord(2)+zd0(2);zcoord(3)+zd0(3);zcoord(4)+zd0(4);zcoord(1)+zd0(1);zcoord(5)+zd0(5);zcoord(6)+zd0(6);zcoord(7)+zd0(7);zcoord(8)+zd0(8);zcoord(5)+zd0(5);zcoord(8)+zd0(8);zcoord(4)+zd0(4);zcoord(3)+zd0(3);zcoord(7)+zd0(7);zcoord(3)+zd0(3);zcoord(2)+zd0(2);zcoord(6)+zd0(6)],'r-');

plot3([xcoord(1);xcoord(2);xcoord(3);xcoord(4);xcoord(1);xcoord(5);xcoord(6);xcoord(7);xcoord(8);xcoord(5);xcoord(8);xcoord(4);xcoord(3);xcoord(7);xcoord(3);xcoord(2);xcoord(6)],[ycoord(1);ycoord(2);ycoord(3);ycoord(4);ycoord(1);ycoord(5);ycoord(6);ycoord(7);ycoord(8);ycoord(5);ycoord(8);ycoord(4);ycoord(3);ycoord(7);ycoord(3);ycoord(2);ycoord(6)],[zcoord(1);zcoord(2);zcoord(3);zcoord(4);zcoord(1);zcoord(5);zcoord(6);zcoord(7);zcoord(8);zcoord(5);zcoord(8);zcoord(4);zcoord(3);zcoord(7);zcoord(3);zcoord(2);zcoord(6)],'b-');
    title('位移变形图');
    hold on
end

%%位移云图
figure(3)
lengthx=0.1;
lengthy=0.1;
lengthz=0.1;
redisp=reshape(displacement,3,length(displacement)/3);
```

```
 xyz=[];
 x=linspace(0,lengthx,lx+1)';
y=linspace(0,lengthy,ly+1)';
mi=1;
for i=1:ly+1
   for j=1:lx+1
Index(mi)=(i-1)*(lx+1)+j  ;
       z=norm(redisp(:,Index(mi)));
       xyz=[xyz;[x(j) y(i) z]];
       mi=mi+1;
   end
end
x=xyz(:,1);
y=xyz(:,2);
z=xyz(:,3);
[X,Y,Z]=griddata(x,y,z,linspace(min(x),max(x))',linspace(min(y),max(y)
),'v4');
surf(X,Y,Z);
shading interp;
title('位移云图');
colorbar;
%%应力云图
figure(4)
 xyz=[];
 x=linspace(0,lengthx,lx+1)';
y=linspace(0,lengthy,ly+1)';
mi=1;
for i=1:ly+1
   for j=1:lx+1
Index(mi)=(i-1)*(lx+1)+j  ;
         z=stn(stress(Index(mi),:));
       xyz=[xyz;[x(j) y(i) z]];
       mi=mi+1;
   end
end
x=xyz(:,1);
y=xyz(:,2);
z=xyz(:,3);
[X,Y,Z]=griddata(x,y,z,linspace(min(x),max(x))',linspace(min(y),max(y)
),'v4');
surf(X,Y,Z);
shading interp;
title('应力云图');
colorbar;
```

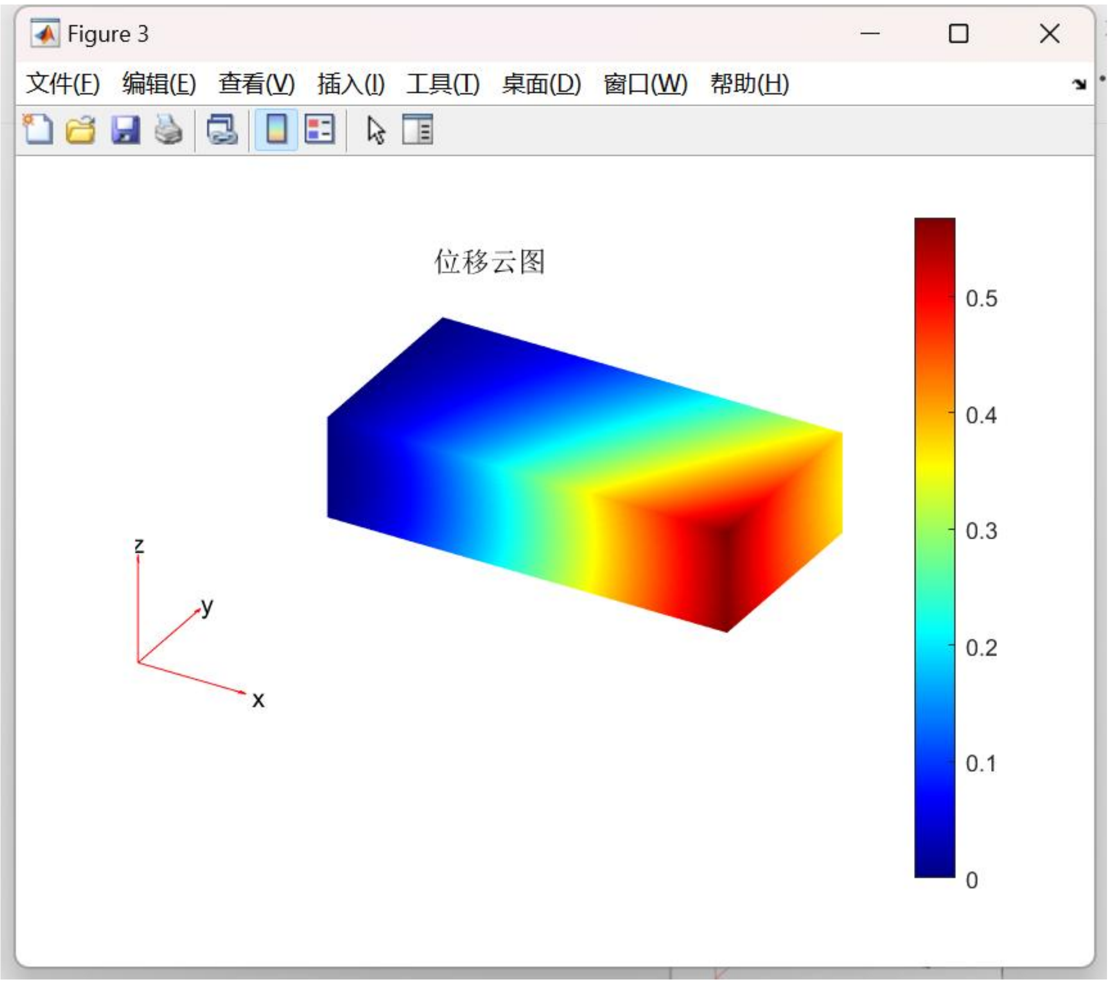
4.Matlab 仿真结果



Matlab 仿真网格划分图

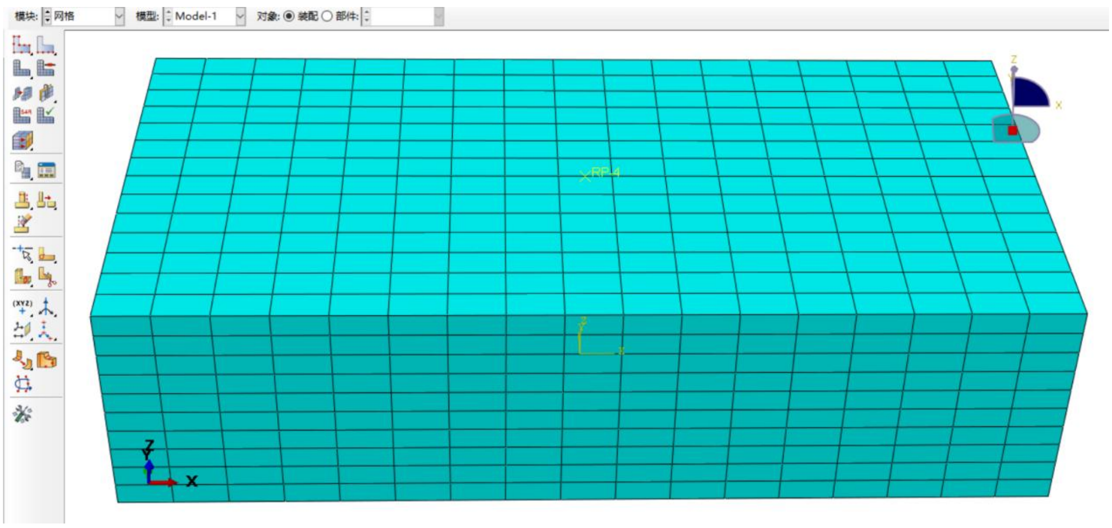


Matlab 仿真应力云图

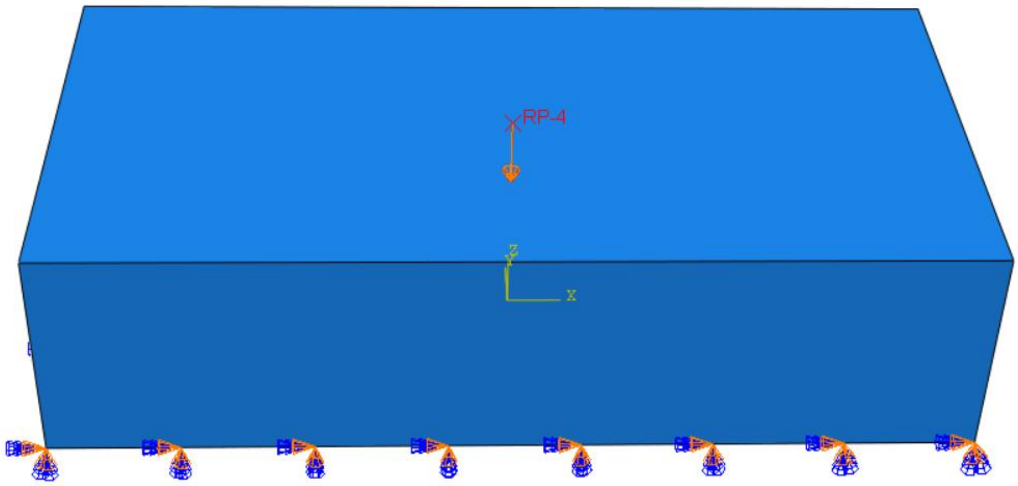


Matlab 仿真位移云图

5. Abaqus 仿真

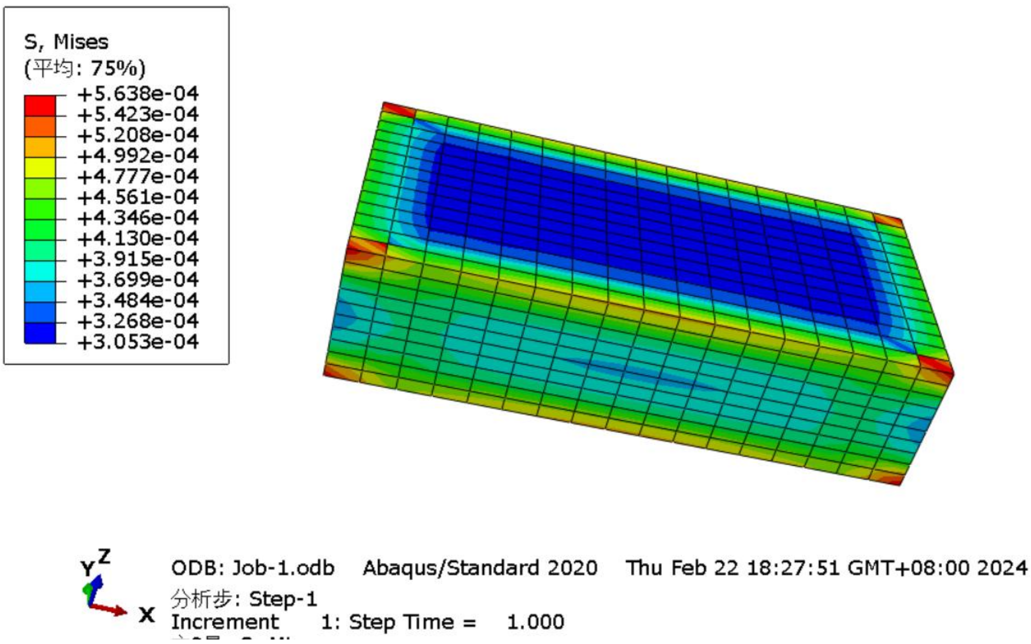


Abaqus 网格划分

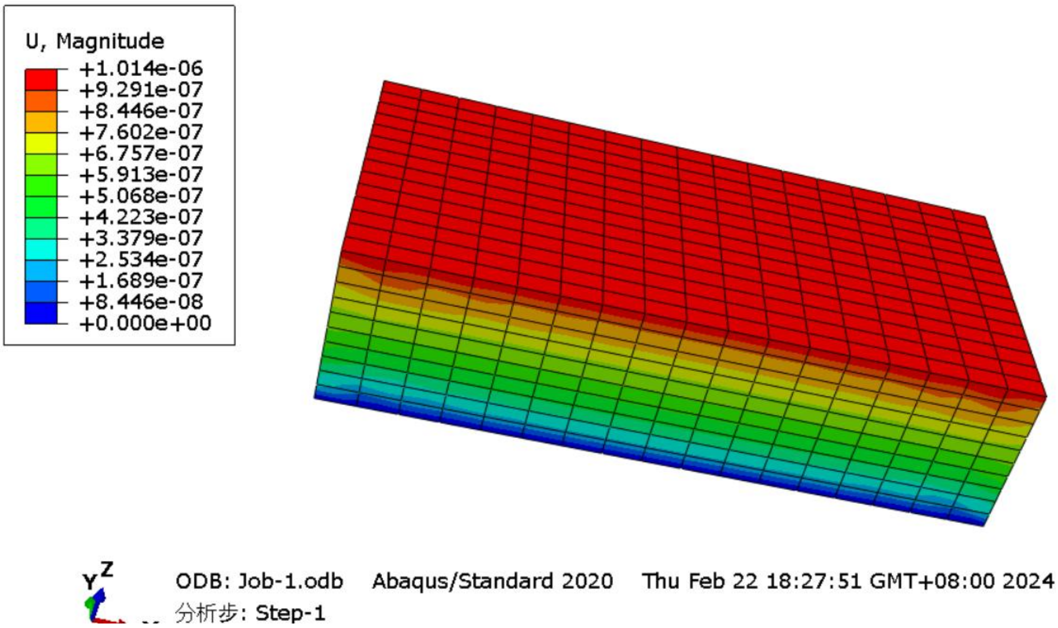


定义载荷与边界条件

6. Abaqus 仿真结果



应力云图



位移云图

7. 结论

通过对比网格划分图，位移变形图，位移云图，应力云图等，我们发现使用Matlab编写的程序与Abaqus软件所得的计算结果在趋势和应力分布上存在一定差异，Matlab程序还有待进一步修改。