

**有限元方法及应用大作业**

作 业 题 目： 楔形板的有限元分析

学 生 姓 名： 余根锐

学 生 学 号： S230200240

任 课 教 师： 王琥

**目录**

[**一、问题描述 3**](#_Toc62228127)

[**二、有限元方法流程 4**](#_Toc62228128)

[**三、有限元编程分析 4**](#_Toc62228129)

[**3.1 网格化分 4**](#_Toc62228130)

[**3.2 建立单元刚度矩阵并组建系统刚度矩阵 6**](#_Toc62228131)

[**3.3 加载边界条件及求解 7**](#_Toc62228132)

[**3.4 计算解位移和应力 7**](#_Toc62228133)

[**3.5 输出结果到plt文件中 8**](#_Toc62228134)

[**3.6刚度矩阵和B矩阵计算 9**](#_Toc62228135)

[**四、后处理 10**](#_Toc62228136)

# 一、问题描述

本文以楔形板为例，进行模型的有限元程序的设计和分析。楔形板的模型如图1-1所示。根据工程实际考虑，楔形板的拉伸应力应变有限元分析可以在保证强度和使用范围的情况下尽可能的减轻材料重量和厚度，并对实际应用具有重要的参考价值。

如图1-1所示物体，长为160mm，宽为40mm，两端高为40mm，中间高为20mm，材料的弹性模量，泊松比。楔形块左端固定，右端最高处施加1 kN/m的线载荷，如图1-2所示。运用等参元法对其进行有限元分析。

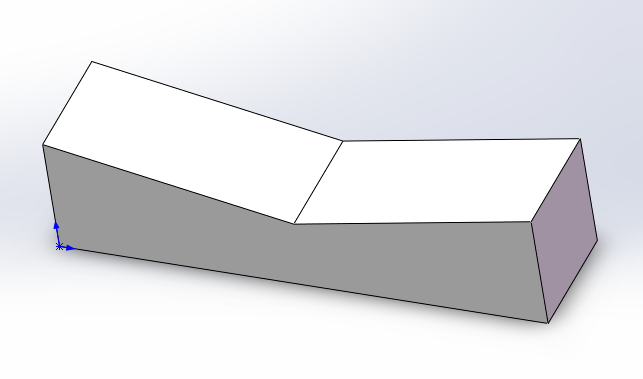


图1-1 零件图

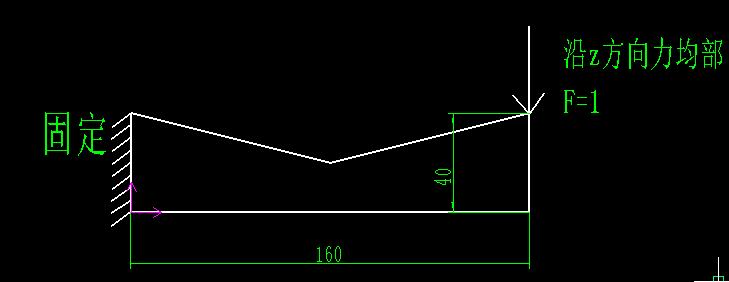
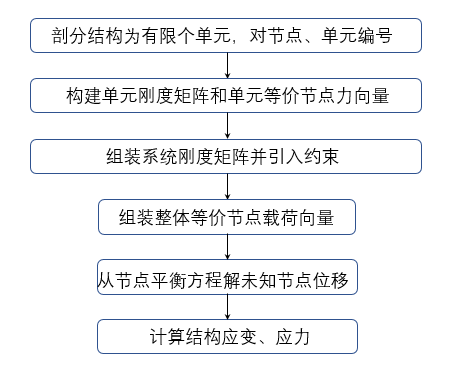


图1-2 x-y截面图，受力图

# 二、有限元方法流程



# 三、有限元编程分析

## 3.1 网格化分

利用tecplot软件对模型进行**前处理**划分网格后的模型情况如图3-1所示，网格单元为六面体单元，楔形板长为160mm，宽为40mm，两端高为40mm，单元总数为个，每个单元有8个节点，节点总数为个，该系统每个节点的自由度数为3，系统总自由度数为个。

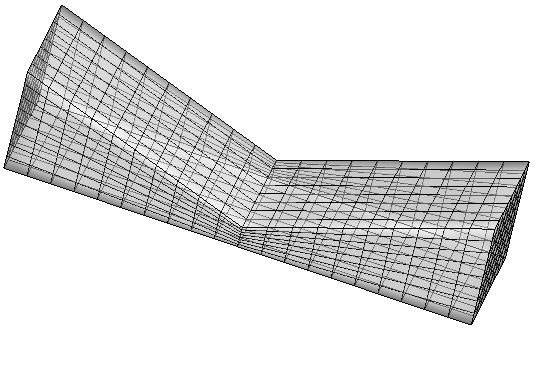


图3-1建模并划分网格图（tecplot所绘）

%-------------------------------------------------------------------------

%前处理

%建立模型，网格划分

%-------------------------------------------------------------------------

ly=8; % X轴的元素总量

lx=20; %Y轴的元素总量

lz=8; %Z轴的元素总量

lengthx=160;

lengthy=40;

lengthz=40;

nel=lx\*ly\*lz; %元素总量

nnel=8; %每个元素的节点数

% 定义材料

E=210000; %弹性模量E

NU=0.3; %泊松比μ

%--------------------------

grad=-0.4; %斜率

%--------------------------

%%节点坐标

%--------------------------

x0=[];

for k=1:lz+1

for i=1:lx+1

if i<=lx/2+1

for j=1:ly+1

x0=[x0; (i-1)\*lengthx/lx(j-1)\*(lengthy+(i-1)\*lengthx\*grad/lx)/ly(k-1)\* lengthz/lz];

end

else

for j=1:ly+1

x0=[x0; (i-1)\*lengthx/lx(j-1)\*(lengthy+0.5\*lengthx\*grad-(i-lx/2-1) \*lengthx\*grad/lx)/ly(k-1)\*lengthz/lz ];

end

end

end

end

%%节点编号

nodes=[];

for k=1:lz

for i=1:lx

for j=1:ly

nodes=[nodes; (k-1)\*(lx+1)\*(ly+1)+(ly+1)\*(i-1)+j (k-1)\*(lx+1)\*(ly+1)+(ly+1)\*i+j (k-1)\*(lx+1)\*(ly+1)+(ly+1)\*i+j+1 (k-1)\*(lx+1)\*(ly+1)+(ly+1)\*(i-1)+j+1 k\*(lx+1)\*(ly+1)+(ly+1)\*(i-1)+j k\*(lx+1)\*(ly+1)+(ly+1)\*i+j k\*(lx+1)\*(ly+1)+(ly+1)\*i+j+1 k\*(lx+1)\*(ly+1)+(ly+1)\*(i-1)+j+1;];

end

end

end

%--------------------------------------------------------------------

## 3.2 建立单元刚度矩阵并组建系统刚度矩阵

%-------------------------------------------------------------------------

%%%计算单元刚度矩阵、刚度矩阵组装

%-------------------------------------------------------------------------

K=sparse(3\*(ly+1)\*(lx+1)\*(lz+1),3\*(ly+1)\*(lx+1)\*(lz+1));%单元刚度矩阵

F=sparse(3\*(ly+1)\*(lx+1)\*(lz+1),1); %系统载荷向量

disp=zeros(3\*(ly+1)\*(lx+1)\*(lz+1),1); %定义位移矩阵

for ii0=1:nel %读取各单元的节点编号

xi1= x0(nodes(ii0,1),1);yi1=x0(nodes(ii0,1),2);zi1=x0(nodes(ii0,1),3);

xi2=x0(nodes(ii0,2),1);yi2=x0(nodes(ii0,2),2);zi2=x0(nodes(ii0,2),3);

xi3= x0(nodes(ii0,3),1);yi3=x0(nodes(ii0,3),2);zi3=x0(nodes(ii0,3),3);

xi4= x0(nodes(ii0,4),1);yi4=x0(nodes(ii0,4),2);zi4=x0(nodes(ii0,4),3);

xi5=x0(nodes(ii0,5),1);yi5=x0(nodes(ii0,5),2);zi5=x0(nodes(ii0,5),3);

xi6= x0(nodes(ii0,6),1);yi6=x0(nodes(ii0,6),2);zi6=x0(nodes(ii0,6),3);

xi7=x0(nodes(ii0,7),1);yi7=x0(nodes(ii0,7),2);zi7=x0(nodes(ii0,7),3);

xi8= x0(nodes(ii0,8),1);yi8=x0(nodes(ii0,8),2);zi8=x0(nodes(ii0,8),3);

Loc=[xi1,yi1,zi1;xi2,yi2,zi2;xi3,yi3,zi3;xi4,yi4,zi4;xi5,yi5,zi5;xi6,yi6,zi6;xi7,yi7,zi7;xi8,yi8,zi8;];

%计算单元刚度

[KE,B,D]=gangdujuzh(E,NU,Loc);

edof=[3\*nodes(ii0,1)-2,3\*nodes(ii0,1)-1,3\*nodes(ii0,1),3\*nodes(ii0,2)-2,3\*nodes(ii0,2)-1,3\*nodes(ii0,2),3\*nodes(ii0,3)-2,3\*nodes(ii0,3)-1,3\*nodes(ii0,3),3\*nodes(ii0,4)-2,3\*nodes(ii0,4)-1,3\*nodes(ii0,4),3\*nodes(ii0,5)-2,3\*nodes(ii0,5)-1,3\*nodes(ii0,5),3\*nodes(ii0,6)-2,3\*nodes(ii0,6)-1,3\*nodes(ii0,6),3\*nodes(ii0,7)-2,3\*nodes(ii0,7)-1,3\*nodes(ii0,7) ,3\*nodes(ii0,8)-2,3\*nodes(ii0,8)-1,3\*nodes(ii0,8)];

%%选出节点、自由度编号、并组装刚度矩阵

K(edof,edof)=K(edof,edof)+KE;

end

%-------------------------------------------------------------------------

## 3.3 加载边界条件及求解

根据模型实际工况加载恰当的边界条件及力。

%-------------------------------------------------------------------

% 边界处理并求解：

%-------------------------------------------------------------------

for i=1:lz+1

F(3\*(ly+1)\*(lx+1)-1+(i-1)\*3\*(ly+1)\*(lx+1),1) = -1000; %力

end

alldofs= [1:3\*(ly+1)\*(lx+1)\*(lz+1)];

for i=1:lz+1

fixeddofs= [1+(i-1)\*3\*(ly+1)\*(lx+1):1:3\*(ly+1)+(i-1)\*3\*(ly+1)\*(lx+1)];%固定自由度

end

## 3.4 计算解位移和应力

%------------------------------------------------------------------------

% 计算解位移和应力

%------------------------------------------------------------------------

%计算位移

freedofs= setdiff(alldofs,fixeddofs);

disp(freedofs,:) = K(freedofs,freedofs) \ F(freedofs,:);

stresplt=zeros(3\*(lx+1)\*(ly+1)\*(lz+1),1);

for ii0=1:nel

xi1= x0(nodes(ii0,1),1);yi1=x0(nodes(ii0,1),2);zi1=x0(nodes(ii0,1),3);

xi2=x0(nodes(ii0,2),1);yi2=x0(nodes(ii0,2),2);zi2=x0(nodes(ii0,2),3);

xi3= x0(nodes(ii0,3),1);yi3=x0(nodes(ii0,3),2);zi3=x0(nodes(ii0,3),3);

xi4= x0(nodes(ii0,4),1);yi4=x0(nodes(ii0,4),2);zi4=x0(nodes(ii0,4),3);

xi5=x0(nodes(ii0,5),1);yi5=x0(nodes(ii0,5),2);zi5=x0(nodes(ii0,5),3);

xi6= x0(nodes(ii0,6),1);yi6=x0(nodes(ii0,6),2);zi6=x0(nodes(ii0,6),3);

xi7=x0(nodes(ii0,7),1);yi7=x0(nodes(ii0,7),2);zi7=x0(nodes(ii0,7),3);

xi8= x0(nodes(ii0,8),1);yi8=x0(nodes(ii0,8),2);zi8=x0(nodes(ii0,8),3);

Loc=[xi1,yi1,zi1;xi2,yi2,zi2;xi3,yi3,zi3;xi4,yi4,zi4;xi5,yi5,zi5;xi6,yi6,zi6;xi7,yi7,zi7;xi8,yi8,zi8;];

[KE,B,D]=gangdujuzh(E,NU,Loc); %计算应变矩阵

edof=[3\*nodes(ii0,1)-2,3\*nodes(ii0,1)-1,3\*nodes(ii0,1),3\*nodes(ii0,2)-2,3\*nodes(ii0,2)-1,3\*nodes(ii0,2),3\*nodes(ii0,3)-2,3\*nodes(ii0,3)-1,3\*nodes(ii0,3),3\*nodes(ii0,4)-2,3\*nodes(ii0,4)-1,3\*nodes(ii0,4),3\*nodes(ii0,5)-2,3\*nodes(ii0,5)-1,3\*nodes(ii0,5),3\*nodes(ii0,6)-2,3\*nodes(ii0,6)-1,3\*nodes(ii0,6) ,3\*nodes(ii0,7)-2,3\*nodes(ii0,7)-1,3\*nodes(ii0,7) ,3\*nodes(ii0,8)-2,3\*nodes(ii0,8)-1,3\*nodes(ii0,8)];

stresplt(edof,:)= stresplt(edof,:)+k\*disp(edof,:);

stress(ii0,:)=D\*B\*disp(edof,1);

strain(ii0,:)=B\*disp(edof,1);

end

## 3.5 输出结果到plt文件中

%-------------------------------------------------

% 输出结果到plt文件

%-------------------------------------------------

fid\_out=fopen('Q8-result.plt','wt');

fprintf(fid\_out,'TITLE="test case governed by poisson equation"\n');

fprintf(fid\_out,'VARIABLES="x" "y" "z" "u" "v" "w" "s11" "s22" "s33"\n');

fprintf(fid\_out,'ZONE T="flow-field", N= %8d,E=%8d,ET=BRICK, F=FEPOINT\n',nnode,nel);

for i=1:nnode

fprintf(fid\_out,'%10.10f,%10.10f,%10.10f,%10.10f,%10.10f,%10.10f,%10.10f\n', x0(i,1), x0(i,2),x0(i,3), disp(3\*i-2,1), disp(3\*i-1,1),disp(3\*i,1), stresplt(3\*i-2,1),stresplt(3\*i-1,1),stresplt(3\*i,1));

end

## 3.6刚度矩阵和B矩阵计算

function [KE,B,D]=Stiffnesske(E,NU,Loc)

gsx=[-0.7745966692 0 0.7745966692];

gsw=[0.55555555556 0.888888888889 0.55555555556];

D=[1-NU NU NU 0 0 0;NU 1-NU NU 0 0 0;NU NU 1-NU 0 0 0;0 0 0 0.5-NU 0 0;0 0 0 0 0.5-NU 0;0 0 0 0 0 0.5-NU;];

D=D\*(E/((1+NU)\*(1-2\*NU)));

KE=zeros(24,24);

for ii=1:3

sx=gsx(ii);

sw=gsw(ii);

for jj=1:3

nx=gsx(jj);

nw=gsw(jj);

for kk=1:3

tx=gsx(kk);

tw=gsw(kk);

[BD,B,D]=BDcalc(sx,nx,tx,Loc,E,NU);

KE=KE+sw\*nw\*tw\*BD;

end

end

end

function [KE,B,D]=Stiffnesske(E,NU,Loc);

gsx=[-0.7745966692 0 0.7745966692];

gsw=[0.55555555556 0.888888888889 0.55555555556];

D=[1-NU NU NU 0 0 0;NU 1-NU NU 0 0 0;NU NU 1-NU 0 0 0;0 0 0 0.5-NU 0 0;0 0 0 0 0.5-NU 0;0 0 0 0 0 0.5-NU;];

D=D\*(E/((1+NU)\*(1-2\*NU)));

KE=zeros(24,24);

for ii=1:3

sx=gsx(ii);

sw=gsw(ii);

for jj=1:3

nx=gsx(jj);

nw=gsw(jj);

for kk=1:3

tx=gsx(kk);

tw=gsw(kk);

[BD,B,D]=BDcalc(sx,nx,tx,Loc,E,NU);

KE=KE+sw\*nw\*tw\*BD;

end

end

end

# 四、后处理

对其模型离散划分网格，使用tecplot对plt格式文件做后处理。所得云图：

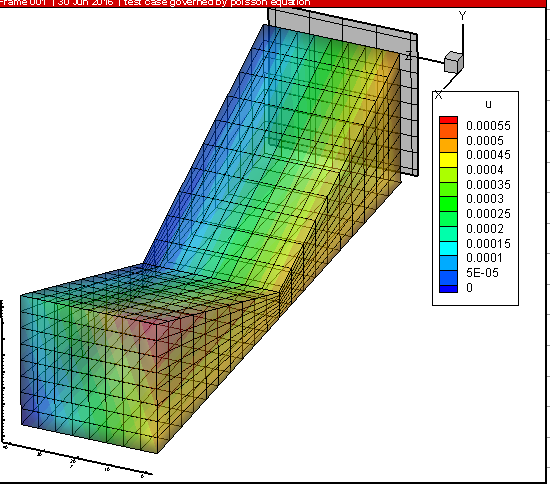


图4-3 X方向应变图

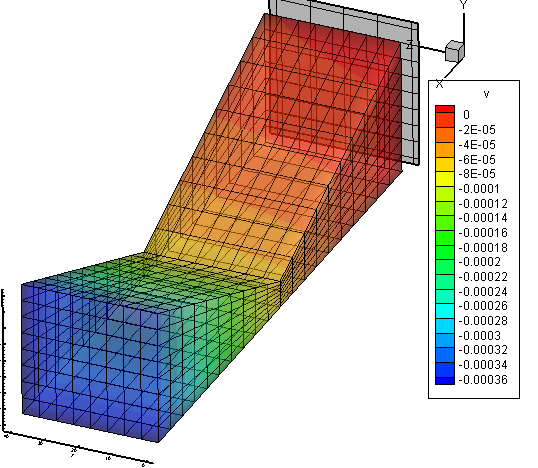


图4-4 Y方向应变图

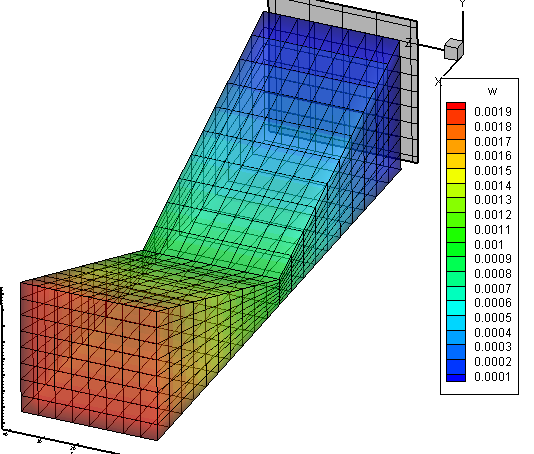


图4-4 Z方向应变图

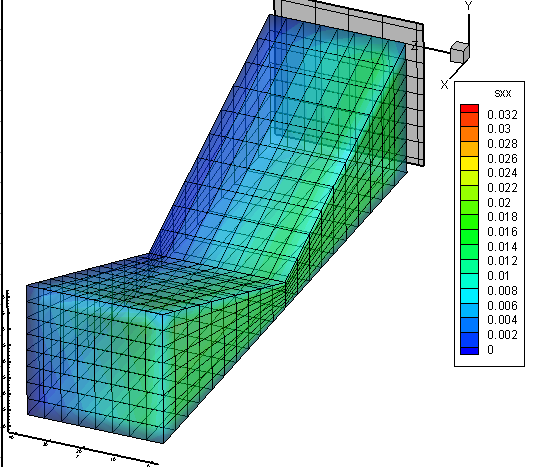


图4-5 应力图

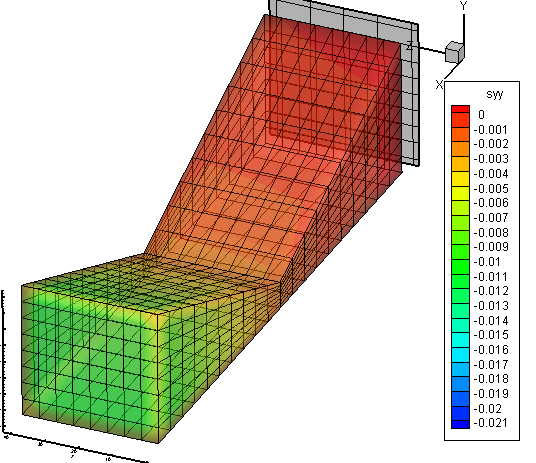


图4-6 应力图

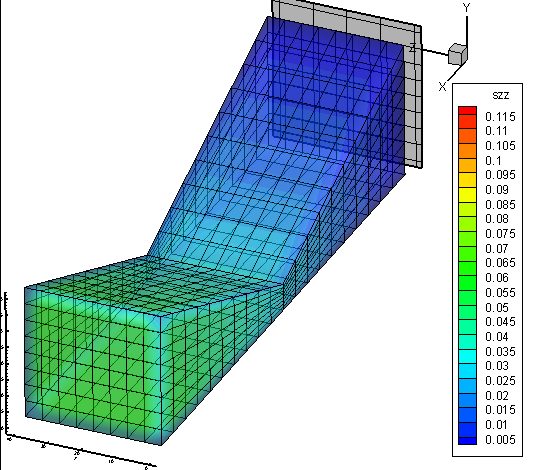


图4-7 应力图