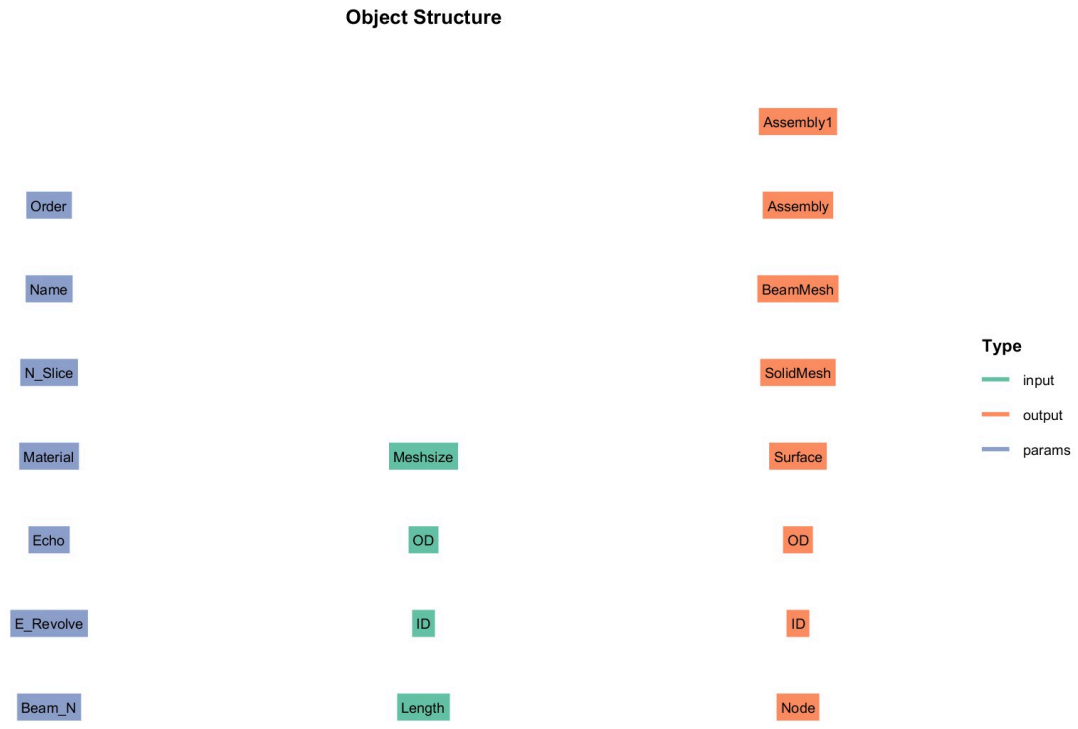
# Commonshaft

Xie Yu

## 1 介绍

Commonshaft用来设计轴，依据输入的长度、外径和内径可生成轴的3D网格和梁网格。

## 2 类结构




输入 input：

- Meshsize : 单元尺寸
- OD : 轴外径
- ID : 轴内径
- Length : 轴长度

参数 params:

- Order： 单元阶数
- Name : 名称
- N_Slice: 梁单元轴向网格划分数量
- E_Revolve: 实体单元旋转方向网格划分数量
- Material : 轴材料

- Beam_N : 梁单元截面环向划分

输出 output :

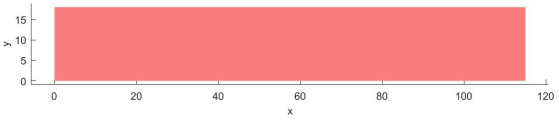
- Assembly1 : 梁单元装配
- Assembly : 实体单元装配
- BeamMesh : 梁网格
- SolidMesh : 实体网格
- OD : 梁各个单元对应外径
- ID : 梁各个单元对应内径
- Node : 梁单元节点

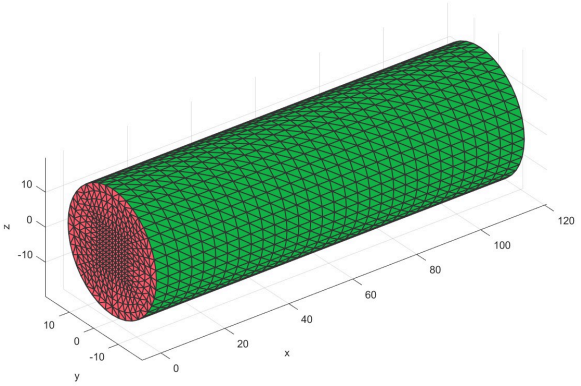
# 3 案例

## 3.1 Shaft1 (Flag=1)

```
% Shaft 1
inputshaft1.Length = 115;
inputshaft1.ID = [0,0];
inputshaft1.OD = [36,36];
paramsshaft1 = struct();
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
Plot2D(obj1);
Plot3D(obj1);
```

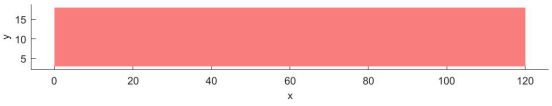
建立轴时定义好长度，内径和外径，即可生成轴网格：

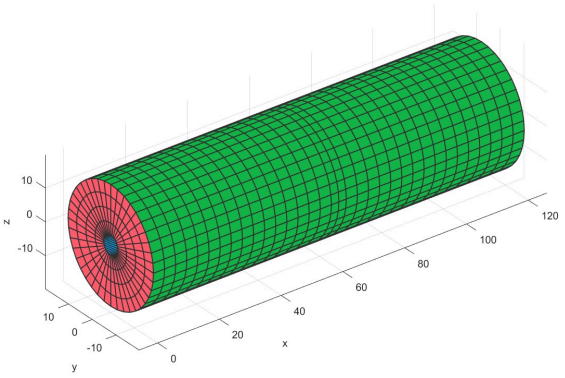


### 3.2 Shaft2 (Flag=2)

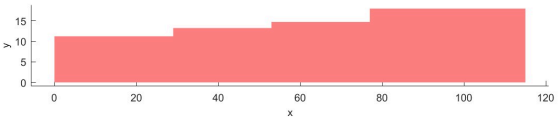
```
% Shaft 2
inputshaft1.Length = 120;
inputshaft1.ID = [6,6];
inputshaft1.OD = [36,36];
paramsshaft1 = struct();
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
Plot2D(obj1);
Plot3D(obj1);
```

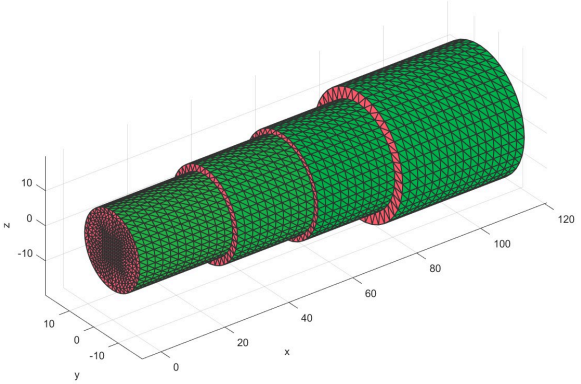



### 3.3 Shaft3 (Flag=3)

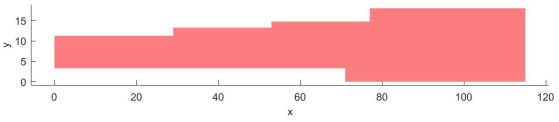
```
% Shaft 3
inputshaft1.Length = [29;53;77;115];
inputshaft1.ID = [[0,0];[0,0];[0,0];[0,0]];
inputshaft1.OD = [[22.5,22.5];[26.5,26.5];[29.5,29.5];[36,36]];
paramsshaft1 = struct();
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
Plot2D(obj1);
Plot3D(obj1);
```

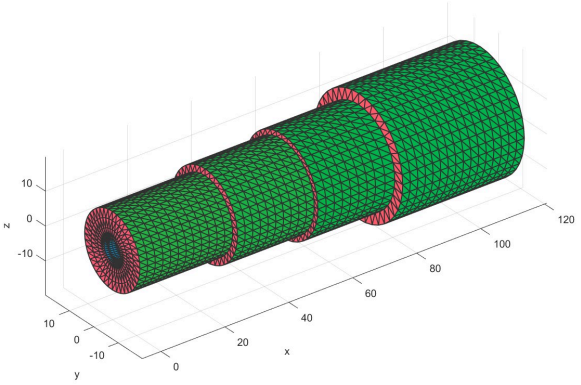



## 3.4 Shaft4 (Flag=4)

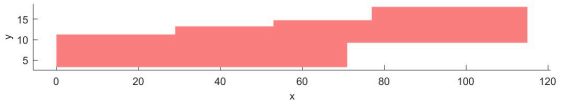
```
% Shaft 4
inputshaft1.Length = [29;53;71;77;115];
inputshaft1.ID = [[6.6,6.6];[6.6,6.6];[6.6,6.6];[0,0];[0,0]];
inputshaft1.OD = [[22.5,22.5];[26.5,26.5];[29.5,29.5];[29.5,29.5];[36,36]];
paramsshaft1 = struct();
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
Plot2D(obj1);
Plot3D(obj1);
```

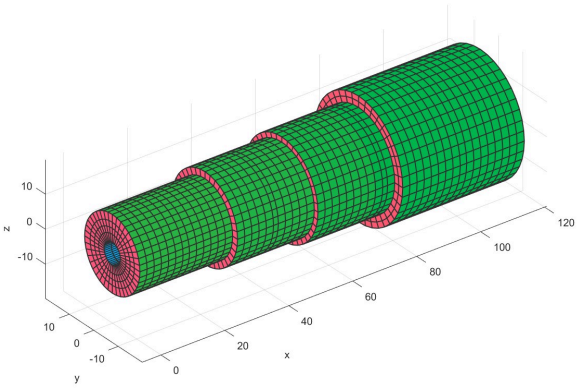


### 3.5 Shaft 5 (Flag=5)

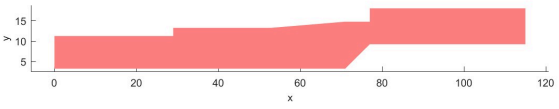
```
% Shaft 5
inputshaft1.Length = [29;53;71;77;115];
inputshaft1.ID = [[6.6,6.6];[6.6,6.6];[6.6,6.6];[18.5,18.5];[18.5,18.5]];
inputshaft1.OD = [[22.5,22.5];[26.5,26.5];[29.5,29.5];[29.5,29.5];[36,36]];
paramsshaft1 = struct();
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
Plot2D(obj1);
Plot3D(obj1);
```

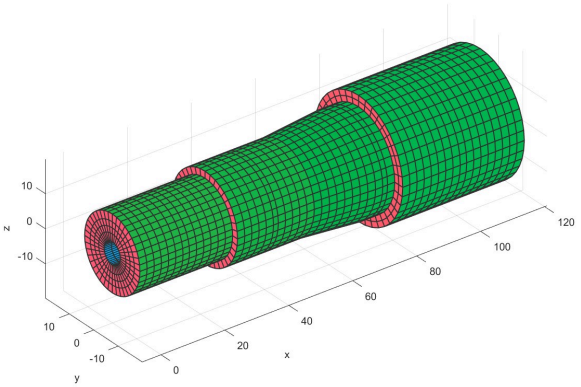



### 3.6 Shaft 6(Flag=6)

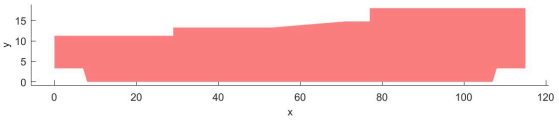
```
% Shaft 6
inputshaft1.Length = [29;53;71;77;115];
inputshaft1.ID = [[6.6,6.6];[6.6,6.6];[6.6,6.6];[6.6,18.5];[18.5,18.5]];
inputshaft1.OD = [[22.5,22.5];[26.5,26.5];[26.5,29.5];[29.5,29.5];[36,36]];
paramsshaft1 = struct();
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
Plot2D(obj1);
Plot3D(obj1);
```

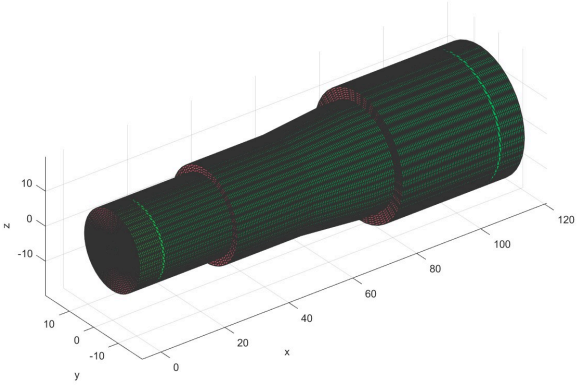



## 3.7 Shaft 7(Flag=7)

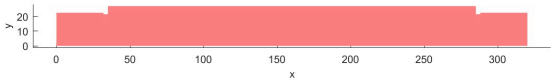
```
% Shaft 7
inputshaft1.Length = [7;8;29;53;71;77;107;108;115];
inputshaft1.ID = [[6.6,6.6];[6.6,0];[0,0];[0,0];[0,0];[0,0];[0,0];[0,6.6];[6.6,6.6]];
inputshaft1.OD = [[22.5,22.5];[22.5,22.5];[22.5,22.5];[26.5,26.5];[26.5,29.5];
[29.5,29.5];...
[36,36];[36,36];[36,36]];
inputshaft1.Meshsize=0.5;
paramsshaft1 = struct();
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
Plot2D(obj1);
Plot3D(obj1);
```

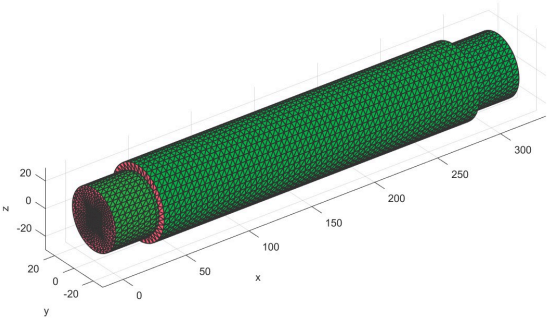


### 3.8 Shaft8(Flag=8)

```
% Shaft 8
inputshaft1.Length = [32;35;320-35;320-32;320];
inputshaft1.ID = [[0,0];[0,0];[0,0];[0,0];[0,0]];
inputshaft1.OD = [[45,45];[43,43];[54,54];[43,43];[45,45]];
paramsshaft1.Order = 1;
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
Plot2D(obj1);
Plot3D(obj1);
```



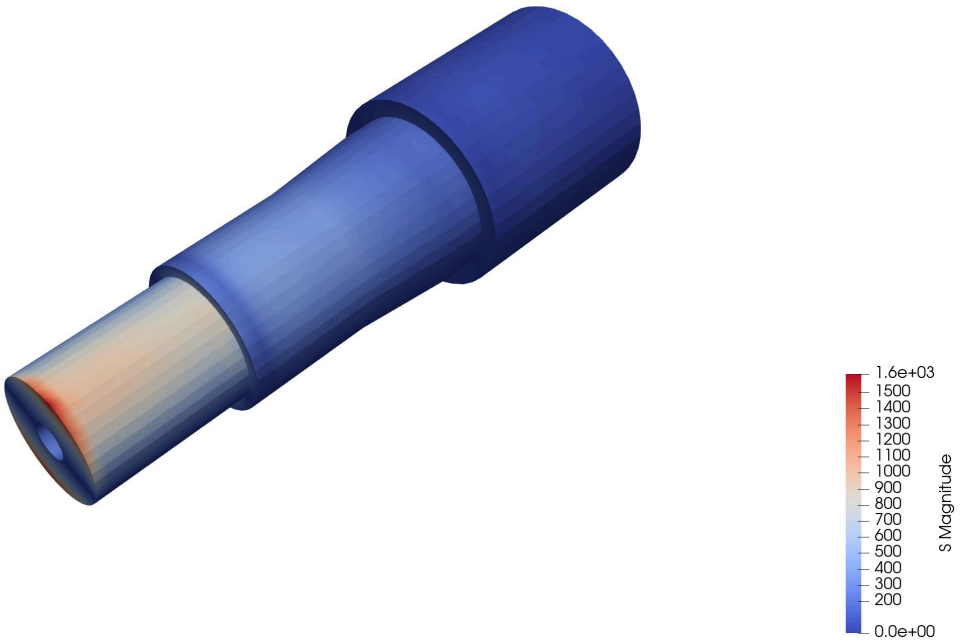




### 3.9 OutputSoliodModel to ANSYS (Flag=9)

```
% Shaft 7
inputshaft1.Length = [7;8;29;53;71;77;107;108;115];
inputshaft1.ID = [[6.6,6.6];[6.6,0];[0,0];[0,0];[0,0];[0,0];[0,0];[0,6.6];[6.6,6.6]];
inputshaft1.OD = [[22.5,22.5];[22.5,22.5];[22.5,22.5];[26.5,26.5];[26.5,29.5];
[29.5,29.5];...
[36,36];[36,36];[36,36]];
paramsshaft1.Order = 2;
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
Plot2D(obj1);
Plot3D(obj1);
%% Assembly
m=obj1.output.SolidMesh;
Ass=Assembly('Shaft_Assembly');
Ass=AddPart(Ass,m.Meshoutput);
% Boundary
Ass=AddBoundary(Ass,1,'No',302);
Bound1=[1,1,1,0,0,0];
Ass=SetBoundaryType(Ass,1,Bound1);
% Load
Ass=AddLoad(Ass,1,'No',305);
Load1=[0,-1e4,0,0,0,0];
Ass=SetLoad(Ass,1,Load1);
```

```
Plot(Ass,'boundary',1,'load',1);
% Material
mat.Name='Steel';
mat.table=["DENS",7.85e-9;"EX",210000;"NUXY",0.3];
Ass=AddMaterial(Ass,mat);
Ass=SetMaterial(Ass,1,1);
% Element type
if paramsshaft1.Order==1
ET.name='185';
else
ET.name='186';
end
ET.opt=[];
ET.R=[];
Ass=AddET(Ass,ET);
Ass=SetET(Ass,1,1);
Ass=AddSensor(Ass,'Stress',1);
opt.ANTYPE=0;
Ass=AddSolu(Ass,opt);
%% Output to ANSYS
ANSYS_Output(Ass);
ANSYSSolve(Ass)
PlotSensor(Ass,1)
```

建立轴模型，通过ANSYS分析计算，最终将结果导出到Paraview中显示：




### 3.10 OutputBeamModel to ANSYS (Flag=10)

```
% Shaft 7
inputshaft1.Length = [7;8;29;53;71;77;107;108;115];
inputshaft1.ID = [[6.6,6.6];[6.6,0];[0,0];[0,0];[0,0];[0,0];[0,0];[0,6.6];[6.6,6.6]];
inputshaft1.OD = [[22.5,22.5];[22.5,22.5];[22.5,22.5];[26.5,26.5];[26.5,29.5];
[29.5,29.5];...
[36,36];[36,36];[36,36]];
paramsshaft1 = struct();
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
```

```
%% Add assembly
Ass=Assembly('Common_shaft_BeamModel');
Ass=AddPart(Ass,obj1.output.BeamMesh.Meshoutput);
% Boundary
Ass=AddBoundary(Ass,1,'No',1);
Bound1=[1,1,1,1,1,1];
Ass=SetBoundaryType(Ass,1,Bound1);
% Material
mat.table=["EX",2.1e5;"PRXY",0.3];
Ass=AddMaterial(Ass,mat);
Ass=SetMaterial(Ass,1,1);
% Add load
Ass=AddLoad(Ass,1,'No',101);
Load1=[0,-1e4,0,0,0,0];
Ass=SetLoad(Ass,1,Load1);
% Element type
ET.name='188';
ET.opt=[];
ET.R=[];
Ass=AddET(Ass,ET);
Ass=SetET(Ass,1,1);
Ass=BeamK(Ass,1);
% Section
Section=obj1.output.BeamMesh.Section;
Ass=DividePart(Ass,1,mat2cell((1:size(Section,1))',ones(1,size(Section,1))));
for i=1:size(Section,1)
Ass=AddSection(Ass,Section{i,1});
Ass=SetSection(Ass,i,i);
end
Plot(Ass,'boundary',1,'load',1,'load_scale',0.1);
%% Output to ANSYS
Ass=AddSensor(Ass,'U',1);
opt.ANTYPE=0;
Ass=AddSolu(Ass,opt);
%% Output to ANSYS
ANSYS_Output(Ass);
ANSYSSolve(Ass)
PlotSensor(Ass,1)
```

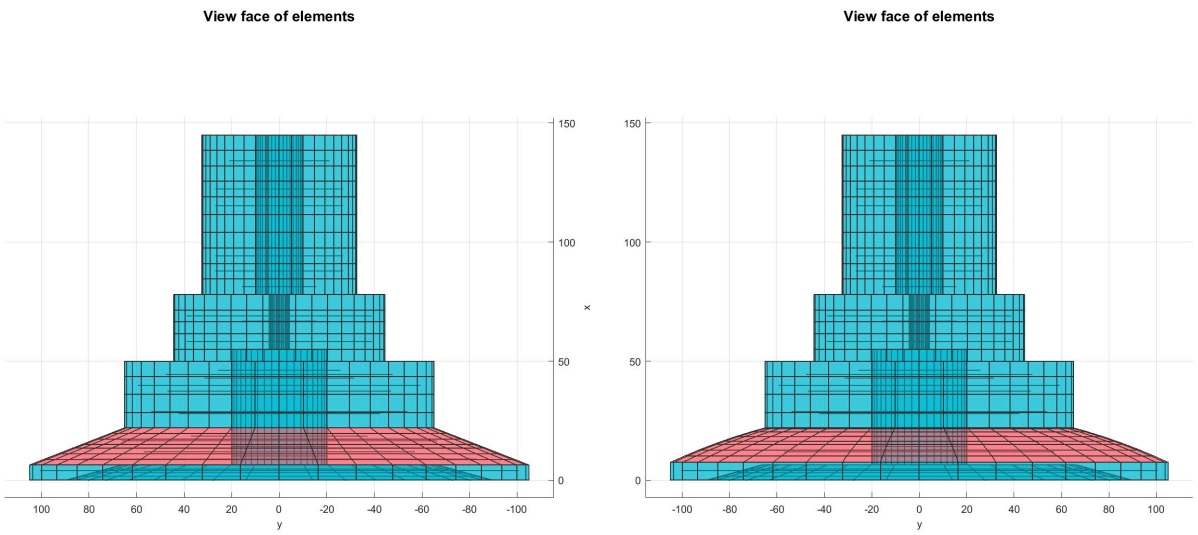
将轴网格导出到Paraview中显示：

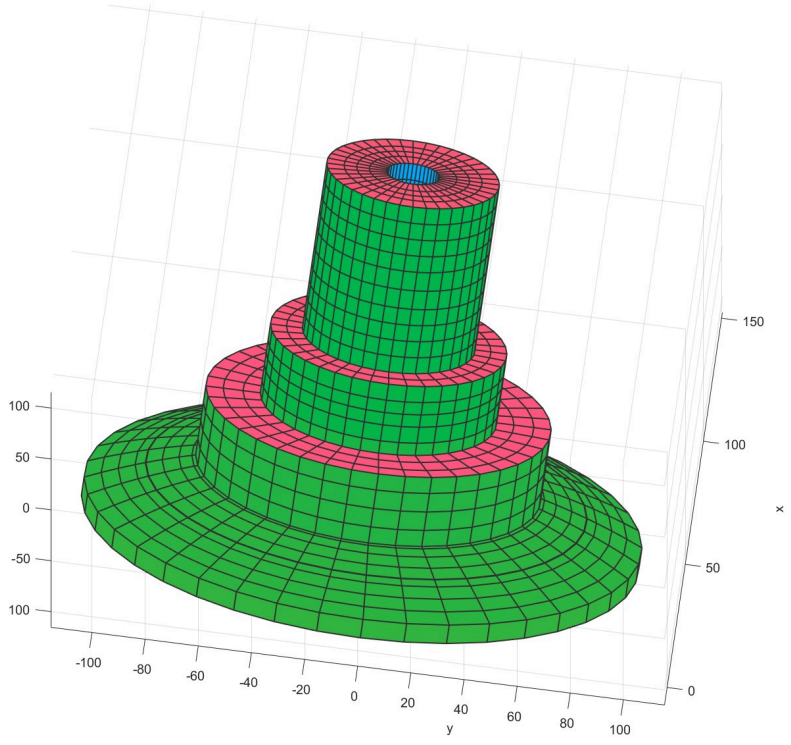

### 3.11 Deform Face (Flag=11)

```
inputShaft1.Length = [6.4;22;50;55;78;145];
inputShaft1.ID = [[180,133];[40,40];[40,40];[40,40];[8.5,8.5];[20,20]];
inputShaft1.OD = [[210,210];[210,130];[130,130];[89,89];[89,89];[65,65]];
paramsShaft1 = struct();
obj1 = shaft.Commonshaft(paramsShaft1, inputShaft1);
obj1 = obj1.solve();
Plot3D(obj1,'faceno',102);
f=@(r)sqrt(250^2-r.^2)+22-sqrt(250^2-65^2);
obj1=DeformFace(obj1,102,f);
Plot3D(obj1);
```

支持小幅度的网格节点偏移。

注意：当网格节点偏移时，网格会发生变形，可能引起网格读取错误



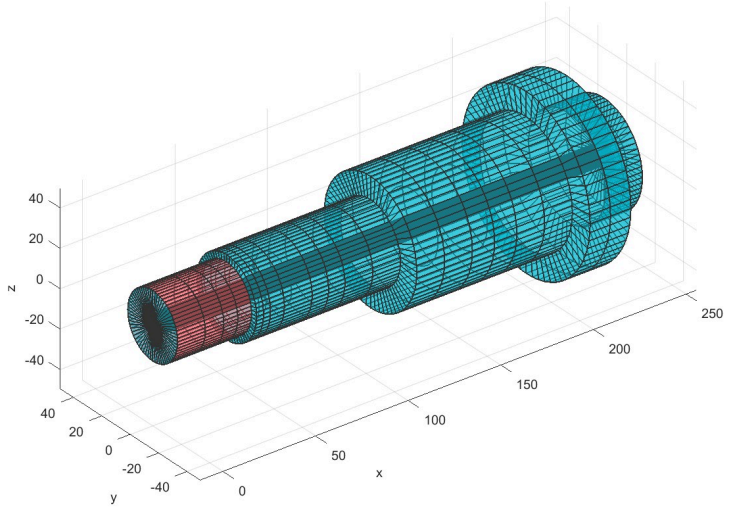




## 3.12 Plot faceno (Flag=12)

```
Shaft2_Height=243.7;
Shaft2_Step_Height=20;
inputShaft2.Length = [40;117.7;201.7;Shaft2_Height-Shaft2_Step_Height;Shaft2_Height];
inputShaft2.OD = [[35,35];[42,42];[65,65];[90,90];[54,54]];
inputShaft2.ID = [[10,10];[10,10];[10,10];[10,10];[10,10]];
inputShaft2.Meshsize=15;
paramsShaft2.E_Revolve = 60;
obj1 = shaft.Commonshaft(paramsShaft2, inputShaft2);
obj1 = obj1.solve();
Plot3D(obj1,'faceno',101);
```

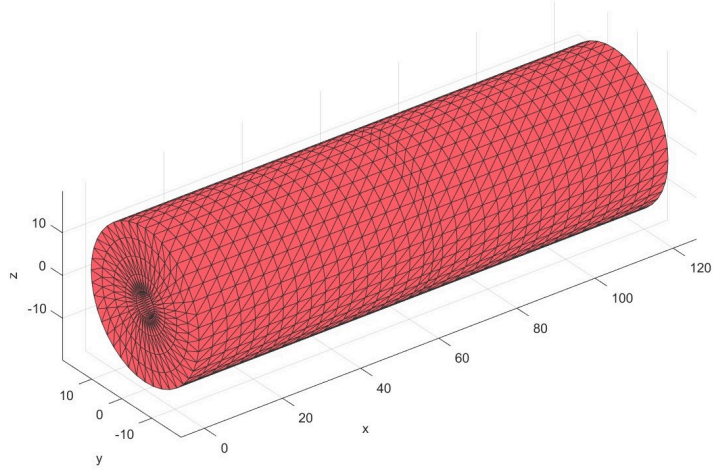
显示面编号：

View face of elements




## 3.13 Output STL (Flag=13)

```
% Shaft 2
inputshaft1.Length = 120;
inputshaft1.ID = [6,6];
inputshaft1.OD = [36,36];
paramsshaft1 = struct();
obj1 = shaft.Commonshaft(paramsshaft1, inputshaft1);
obj1 = obj1.solve();
Plot2D(obj1);
Plot3D(obj1);
OutputSTL(obj1)
% Load stl file
L=Layer('test');
Name=strcat(obj1.params.Name,'.stl');
L=STLRead(L,Name);
Plot(L);
end
```



# 4 参考文献