

Mesh2D

Xie Yu

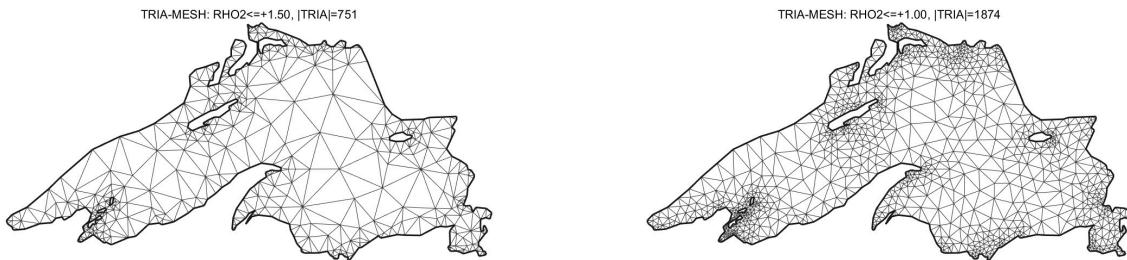
1 介绍

Mesh2D用来建立二维网格，可以导入Surface2D来进行网格划分。

2 原理

2.1 MESH2D: Delaunay-based unstructured mesh-generation

Baffalo 采用MESH2D^[1]作为2D网格划分求解器，它可基于给定的边界采用Delaunay三角形进行网格划分。



3 案例

3.1 Create gear obj (Flag=1)

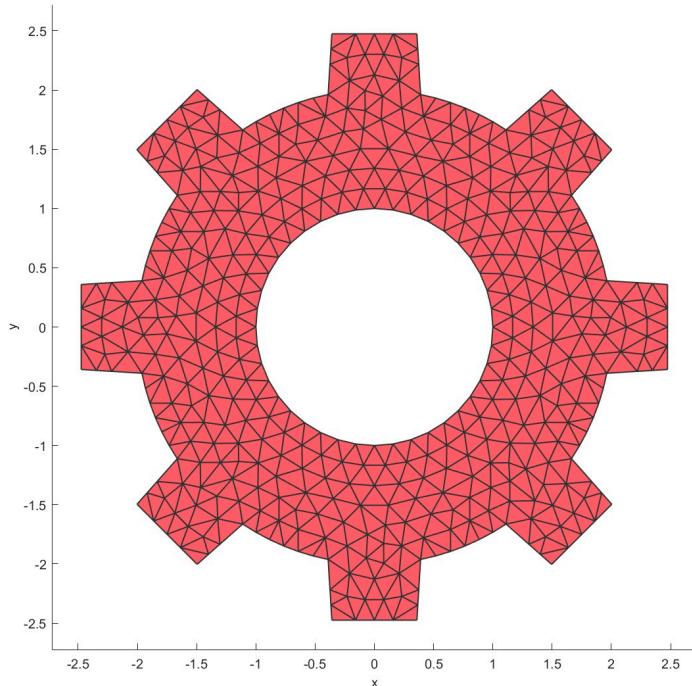
```
1 a=Point2D('Point Ass1');
2 neg=8;
3 OR1=2.5;OR2=2;IR=1;
4 D_theta=3/180*pi;
5 a=AddPoint(a,0,0);
6 for i=1:neg
7 x1=OR1*cos(pi/neg/2-D_theta-pi/neg*2*(i-1));
8 y1=OR1*sin(pi/neg/2-D_theta-pi/neg*2*(i-1));
9 x2=OR1*cos(-pi/neg/2+D_theta-pi/neg*2*(i-1));
10 y2=OR1*sin(-pi/neg/2+D_theta-pi/neg*2*(i-1));
11 x3=OR2*cos(-pi/neg/2-pi/neg*2*(i-1));
12 y3=OR2*sin(-pi/neg/2-pi/neg*2*(i-1));
13 x4=OR2*cos(-pi/neg/2*3-pi/neg*2*(i-1));
14 y4=OR2*sin(-pi/neg/2*3-pi/neg*2*(i-1));
15 x5=OR1*cos(pi/neg/2-D_theta-pi/neg*2*i);
16 y5=OR1*sin(pi/neg/2-D_theta-pi/neg*2*i);
17 xx=[x1;x2];yy=[y1;y2];
18 a=AddPoint(a,xx,yy);
19 xx=[x2;x3];yy=[y2;y3];
20 a=AddPoint(a,xx,yy);
21 xx=[x3;x4];yy=[y3;y4];
22 a=AddPoint(a,xx,yy);
23 xx=[x4;x5];yy=[y4;y5];
```

```

24 a=AddPoint(a,xx,yy);
25 end
26 b=Line2D('Line Ass1');
27 for i=1:neg
28 b=AddLine(b,a,2+(i-1)*4);
29 b=AddLine(b,a,3+(i-1)*4);
30 b=AddCircle(b,OR2,a,1,'sang',-180/neg/2-180/neg*2*(i-1),'ang',-180/neg);
31 b=AddLine(b,a,5+(i-1)*4);
32 end
33 S=Surface2D(b);
34 a1=Point2D('Point Group2');
35 a1=AddPoint(a1,0,0);
36 h=Line2D('Hole Group1');
37 h=AddCircle(h,IR,a1,1);
38 S= AddHole(S,h);
39 m=Mesh2D('Mesh1');
40 m=AddSurface(m,S);
41 m=SetSize(m,0.2);
42 m=Mesh(m);
43 Plot(m);

```

对一个类齿轮的封闭平面网格划分：



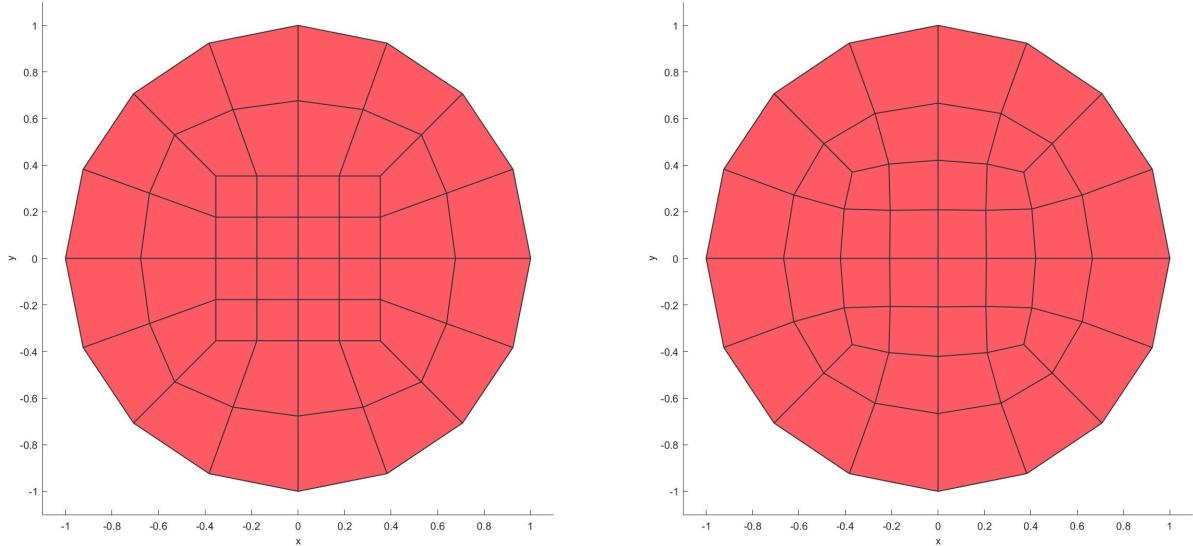
3.2 Create quad circle (Flag=2)

```

1 m=Mesh2D('Mesh1');
2 m=MeshQuadCircle(m,'n',4);
3 Plot(m);
4 m=SmoothFace(m,100);
5 Plot(m);

```

采用四边形网格划分圆。



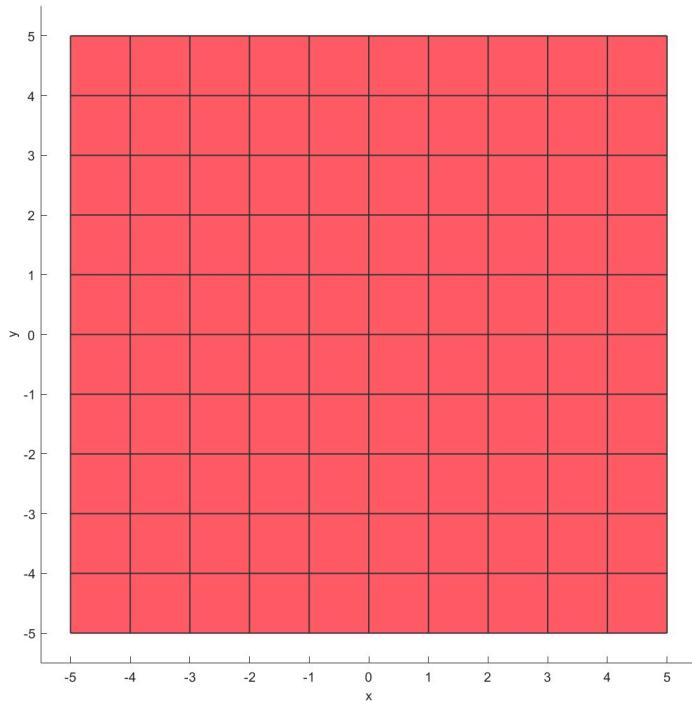
3.3 Create quad plate (Flag=3)

```

1 m=Mesh2D('Mesh1');
2 m=MeshQuadPlate(m,[10,10]);
3 Plot(m);

```

矩形平板四边形网格划分：



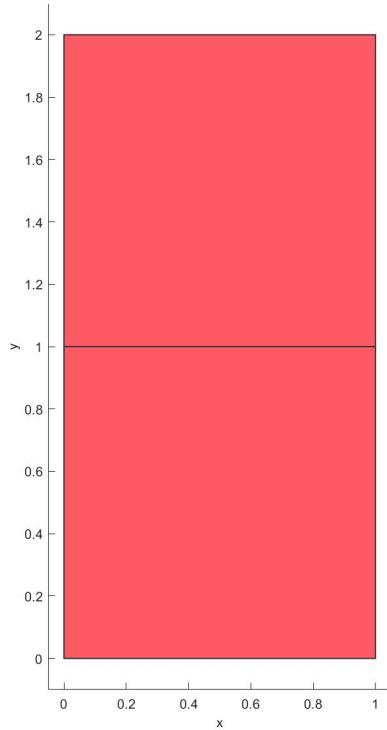
3.4 Add Elements (Flag=4)

```

1 m=Mesh2D('Mesh1');
2 P=[0,0;0,1;1,1;1,0;0,2;1,2];
3 m.Vert=P;
4 m=AddElements(m,[1,2,3,4;2,5,6,3]);
5 Plot(m);

```

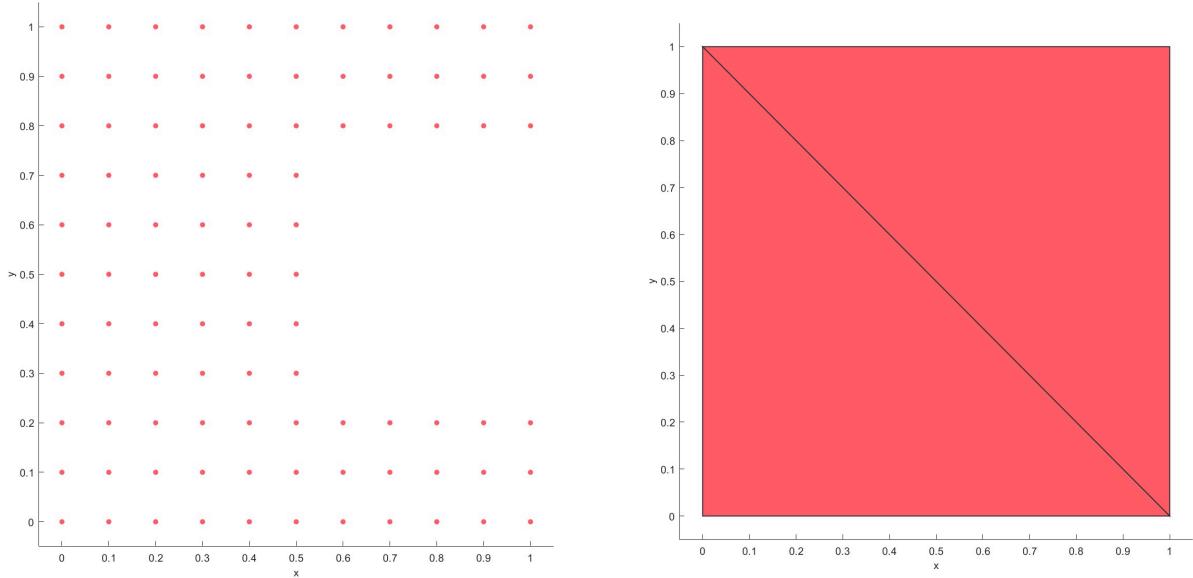
加入网格，采用AddElements加入网格时需要自己定义好网格的节点。



3.5 Convhull (Flag=5)

```
1 [X,Y]=meshgrid(0:0.1:1);
2 points1=[X(:),Y(:)];
3 a=Point2D('Point Ass1');
4 a=AddPoint(a, points1(:,1), points1(:,2));
5 f=@(x,y)(x>0.5 &y>0.2 & y<0.8);
6 a=DeletePoint(a,1,'fun',f);
7 Plot(a);
8 m=Mesh2D('Mesh1');
9 m=Convhull(m,a,'keep',0,'simplicity',true);
10 Plot(m);
```

计算电集的凸包，根据凸包生成网格、

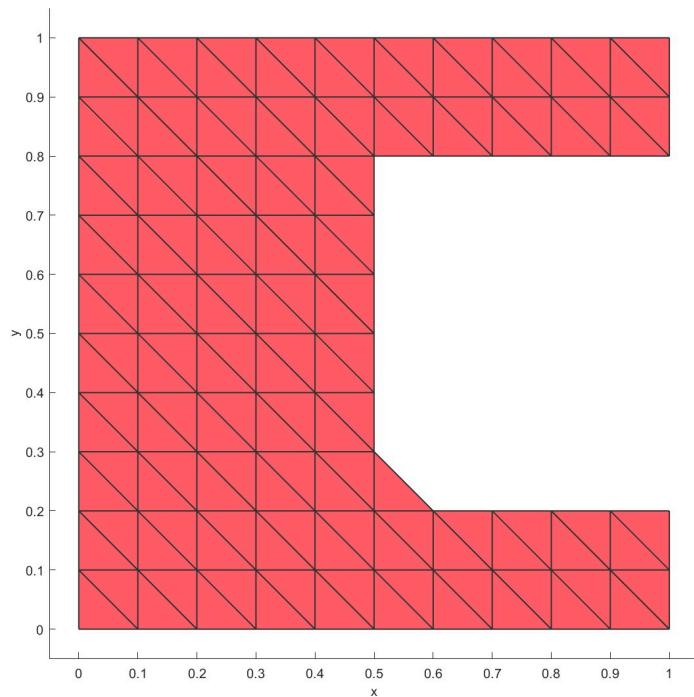


3.6 Convccave1 (Flag=6)

```

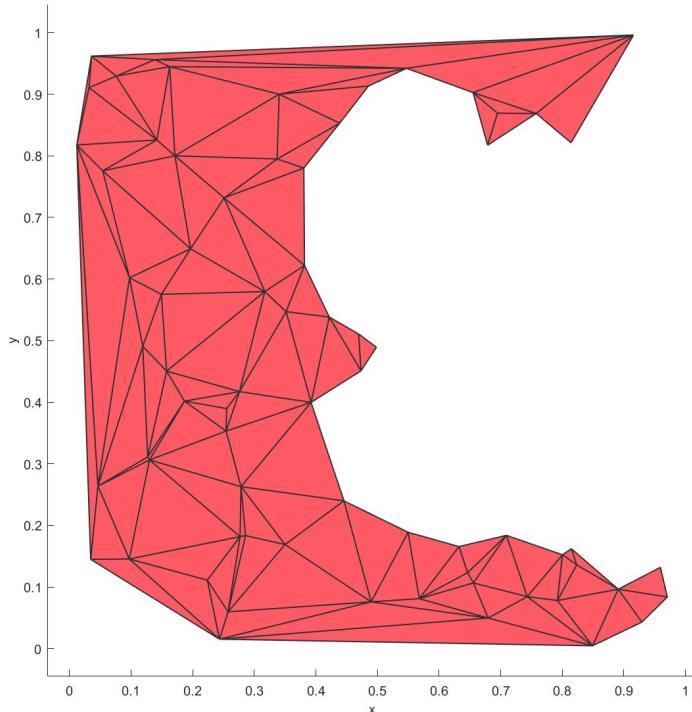
1 [X,Y]=meshgrid(0:0.1:1);
2 points1=[X(:,1),Y(:,1)];
3 a=Point2D('Point Ass1');
4 a=AddPoint(a, points1(:,1), points1(:,2));
5 logic=points1(:,1)>0.5 & points1(:,2)>0.2 & points1(:,2)<0.8;
6 m=Mesh2D('Mesh1');
7 m=Convccave(m,a,'logic',logic);
8 Plot(m);

```



3.7 Convcaue2 (Flag=7)

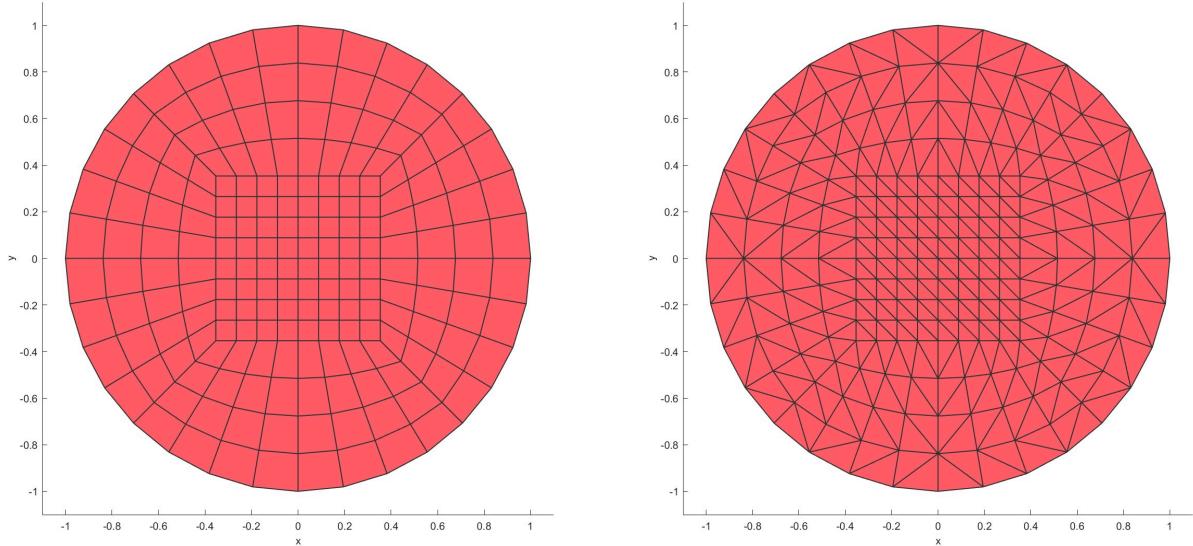
```
1 points1=rand(100,2);
2 a=Point2D('Point Ass1');
3 a=AddPoint(a, points1(:,1), points1(:,2));
4 logic=points1(:,1)>0.5 & points1(:,2)>0.2 & points1(:,2)<0.8;
5 m=Mesh2D('Mesh1');
6 m=Convcaue(m,a,'logic',logic);
7 Plot(m);
```



3.8 Covert quad to tri (Flag=8)

```
1 m=Mesh2D('Mesh1');
2 m=MeshQuadCircle(m,'n',8);
3 Plot(m);
4 m=Quad2Tri(m);
5 Plot(m);
```

将四边形网格转换为三角形单元。



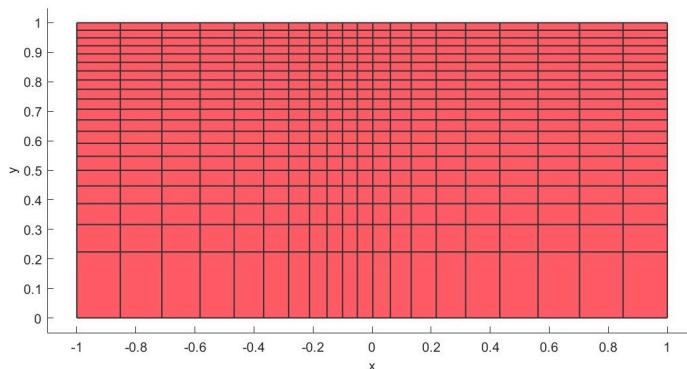
3.9 Mesh 2D Tensor Grid (Flag=9)

```

1 m=Mesh2D('Mesh1');
2 dx = 1-0.5*cos((-1:0.1:1)*pi);
3 x = -1.15+0.1*cumsum(dx);
4 y = 0:0.05:1;
5 m=MeshTensorGrid(m,x,sqrt(y));
6 Plot(m);

```

MRST^[1]提供了很好用的四边形和六面体网格的函数，Buffalo借鉴了部分函数集成到网格划分中。

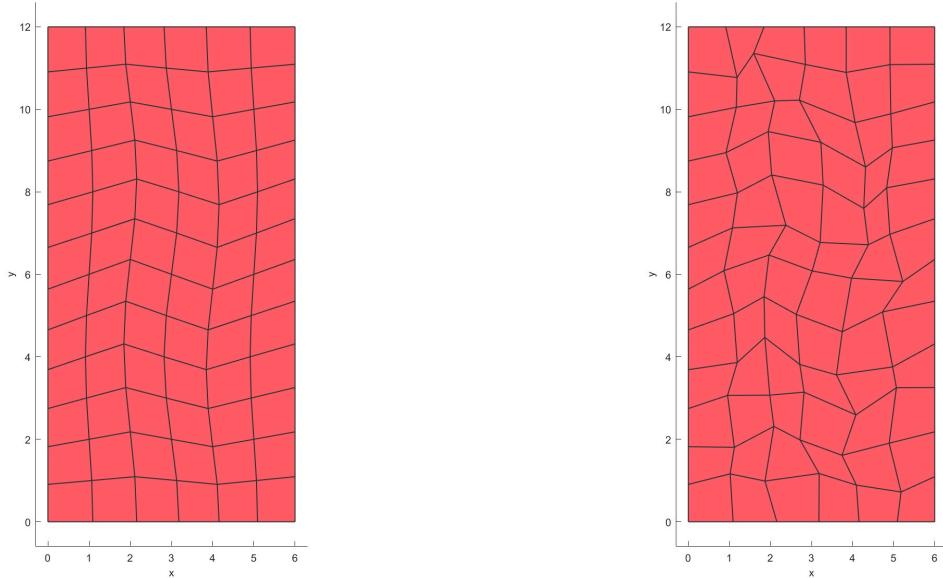


3.10 Mesh 2D Grid (Flag=10)

```

1 m=Mesh2D('Mesh1');
2 nx=6;ny=12;
3 lx=6;ly=12;
4 m=MeshGrid(m,[nx, ny],[lx,ly], 'twist', 0.03);
5 Plot(m);
6 c = m.Vert;
7 I = any(c==0,2) | any(c(:,1)==nx,2) | any(c(:,2)==ny,2);
8 m.Vert(~I,:) = c(~I,:)+0.6*rand(sum(~I),2)-0.3;
9 Plot(m);

```



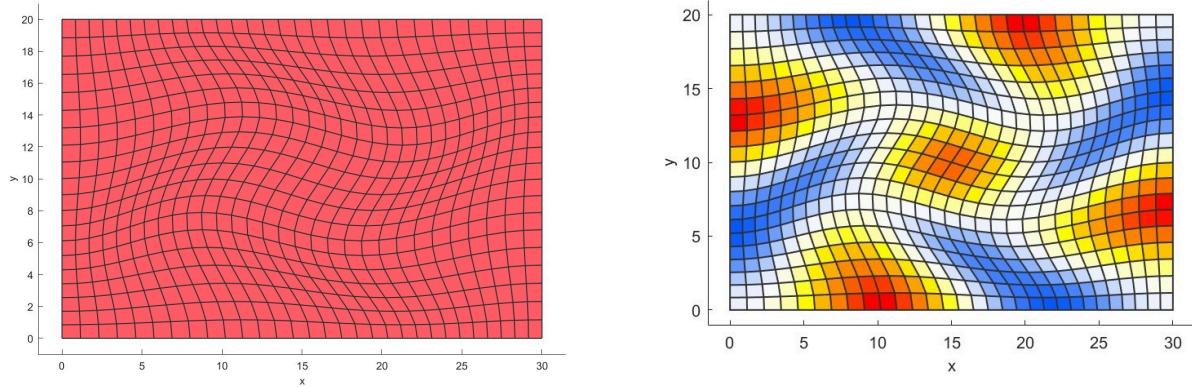
3.11 Compute MRST G (Flag=11)

```

1 m=Mesh2D('Mesh1');
2 nx=30;ny=20;
3 lx=30;ly=20;
4 m=MeshGrid(m,[nx, ny],[lx,ly], 'twist',0.05);
5 Plot(m);
6 m= ComputeGeometryG(m);
7 PlotG(m,'volume',1);

```

ComputeGeometryG用于计算四边形网格的集合信息（中心, 体积, 面积）。



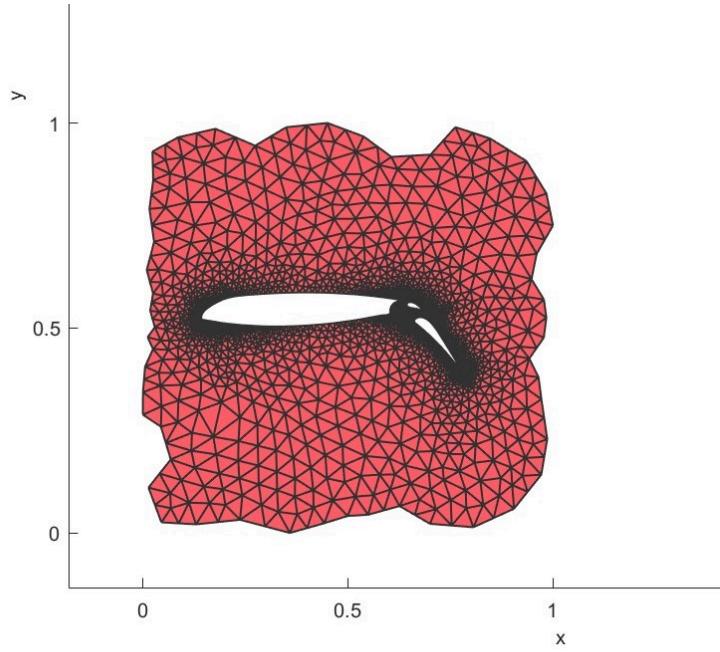
3.12 Load msh file (Flag=12)

```

1 m=Mesh2D('Mesh1');
2 m=LoadMsh(m,'airfoil.msh');
3 Plot(m);

```

读取msh格式网格。



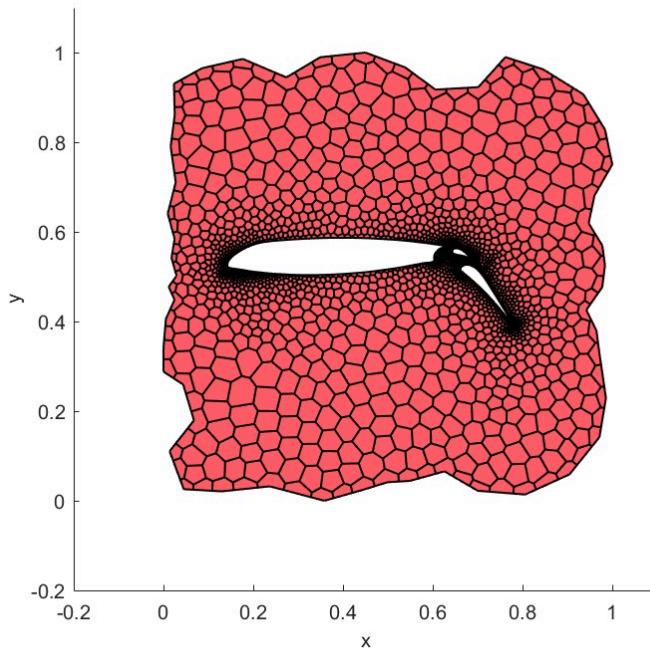
3.13 MeshDual (Flag=13)

```

1 m=Mesh2D('Mesh1');
2 m=LoadMsh(m,'airfoil.msh');
3 Plot(m,'xlim',[-0.2,1.1], 'ylim',[-0.2,1.1]);
4 m=MeshDual(m);
5 PlotDual(m,'xlim',[-0.2,1.1], 'ylim',[-0.2,1.1]);

```

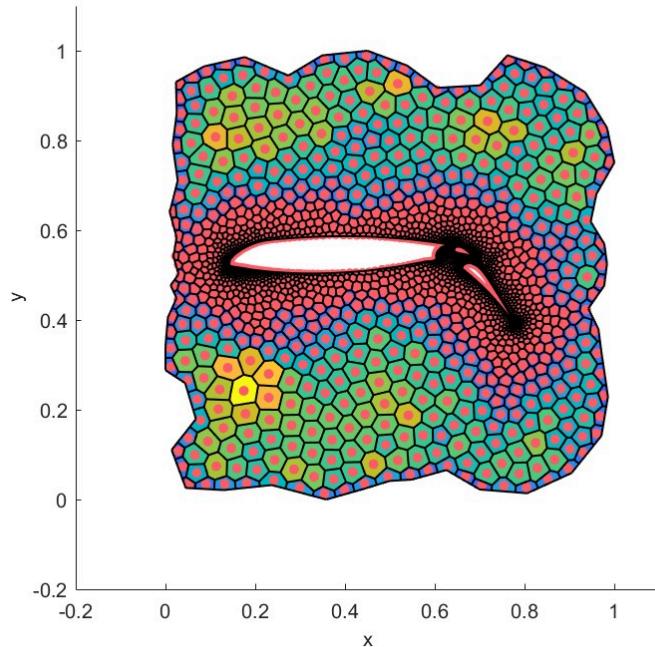
DUALMESH^[3]可以将三角形网格重新划分为多边形网格。



3.14 Plot Dual center and area (Flag=14)

```
1 m=Mesh2D('Mesh1');
2 m=LoadMsh(m,'airfoil.msh');
3 m=MeshDual(m);
4 [pc,ac]=ComputeGeometryDual(m); %#ok<ASGLU>
5 PlotDual(m,'xlim',[-0.2,1.1], 'ylim',[-0.2,1.1], 'area',1, 'center',1);
```

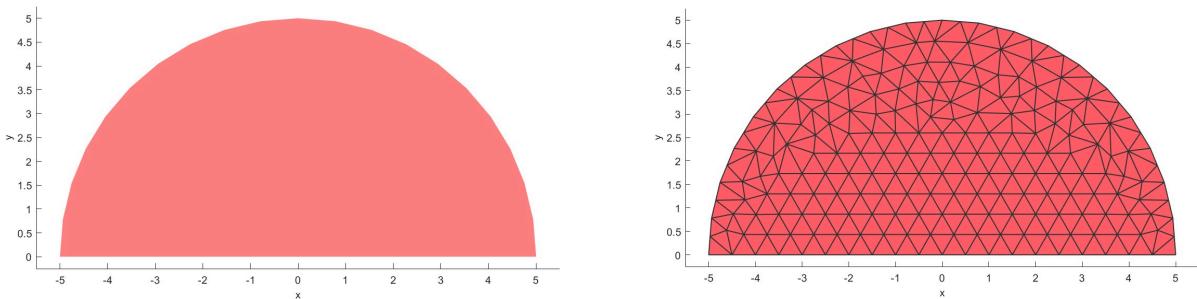
显示多边形网格的中心，并依据面积设置不同的颜色。



3.15 Calculate geometry information (Flag=15)

```
1 % Semi circle
2 a=Point2D('Circle center');
3 a=AddPoint(a,0,0);
4 a=AddPoint(a,[-5;5],[0;0]);
5 b=Line2D('Semi circle');
6 b=AddCircle(b,5,a,1,'ang',180);
7 b=AddLine(b,a,2);
8 S=Surface2D(b);
9 Plot(S);
10 m=Mesh2D('Mesh1');
11 m=AddSurface(m,S);
12 m=SetSize(m,0.5);
13 m=Mesh(m);
14 Plot(m);
15 [Area,Center,Ixx,Iyy,Ixy]= CalculateGeometry(m); %#ok<ASGLU>
```

计算网格截面的几何信息。



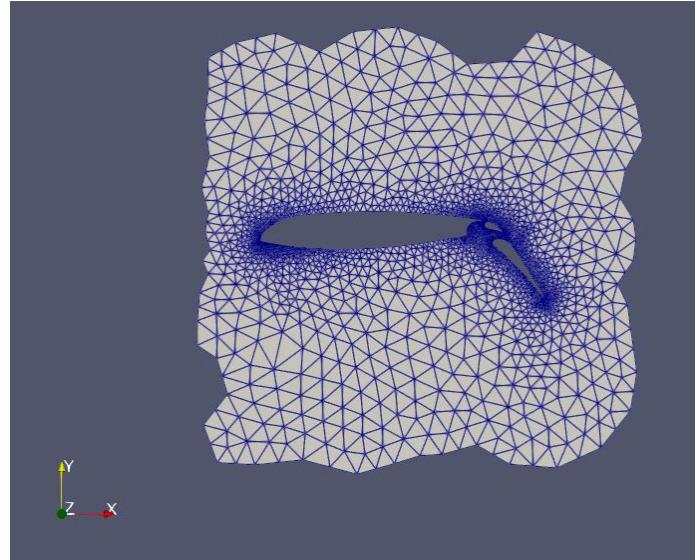
3.16 Plot Mesh2D in paraview (Flag=16)

```

1 m=Mesh2D('Mesh1');
2 m=LoadMsh(m,'airfoil.msh');
3 Plot2(m);

```

在Paraview中显示网格。



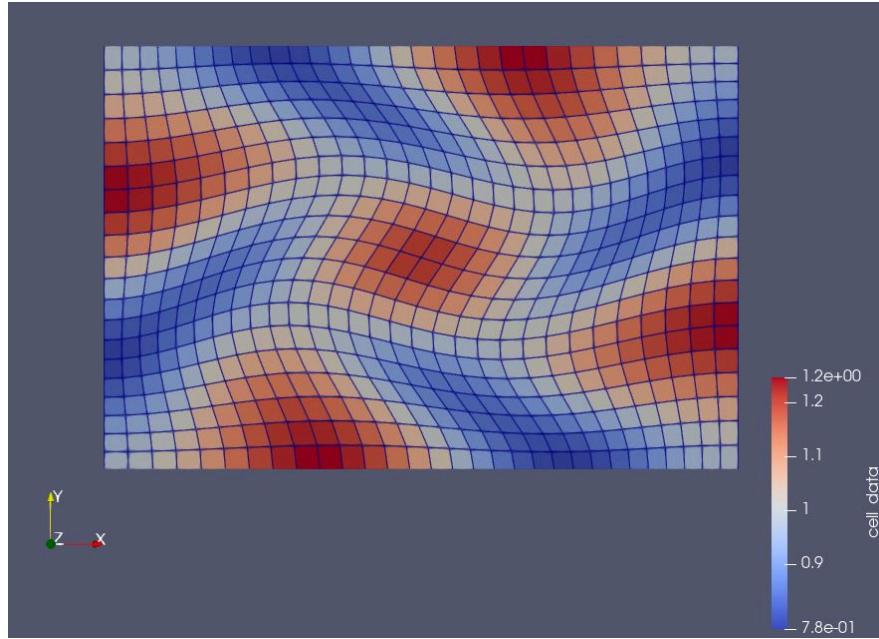
3.17 Plot Mesh2D G and celldata in paraview (Flag=17)

```

1 m=Mesh2D('Mesh1');
2 nx=30;ny=20;
3 lx=30;ly=20;
4 m=MeshGrid(m,[nx, ny],[lx,ly], 'twist', 0.05);
5 m= ComputeGeometryG(m);
6 volume=m.G.cells.volumes;
7 m=AddCellData(m,volume);
8 PlotG2(m);

```

将计算的体积信息导入到celldata中，即可在paraview中显示相关信息。



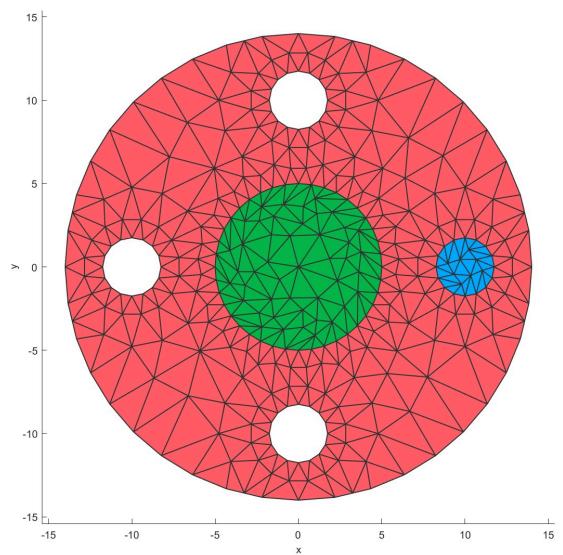
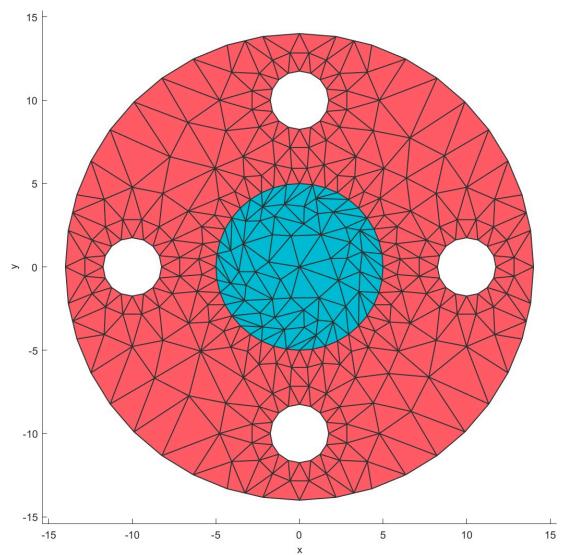
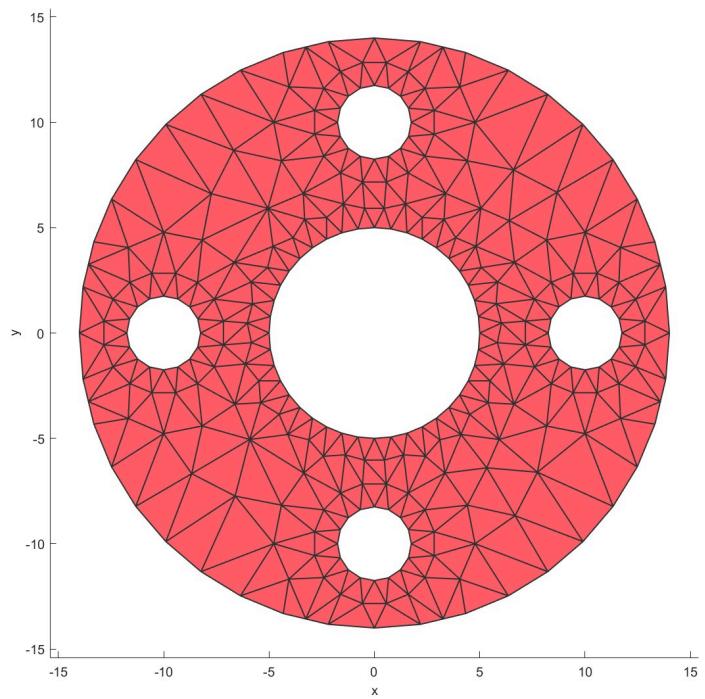
3.18 Mesh edge (Flag=18)

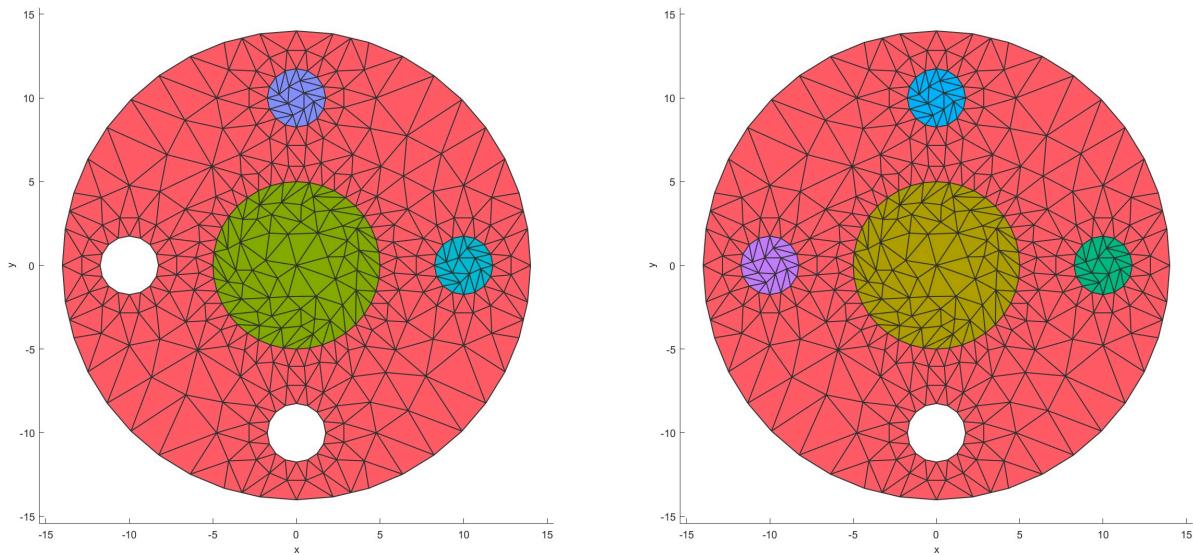
```

1 a=Point2D('Temp');
2 a=AddPoint(a,0,0);
3 % Add outline
4 b1=Line2D('Out');
5 b1=AddCircle(b1,14,a,1);
6 % Add innerline
7 b2=Line2D('Inner');
8 b2=AddCircle(b2,5,a,1,'seg',40);
9 % Add assembly hole
10 h=Line2D('Hole');
11 h=AddCircle(h,3.5/2,a,1,'seg',16);
12 S=Surface2D(b1);
13 S=AddHole(S,b2);
14 for i=1:4
15 S=AddHole(S,h,'dis',[10*cos(pi/2*(i-1)),10*sin(pi/2*(i-1))]);
16 end
17 m=Mesh2D('Temp');
18 m=AddSurface(m,S);
19 m=SetSize(m,5);
20 m=Mesh(m);
21 Plot(m)
22 % Mesh edge
23 m=MeshEdge(m,2);
24 Plot(m)
25 m=MeshEdge(m,2);
26 Plot(m)
27 m=MeshEdge(m,2);
28 Plot(m)
29 m=MeshEdge(m,2);
30 Plot(m)

```

MeshEdge在原始模型的边界划分网格。





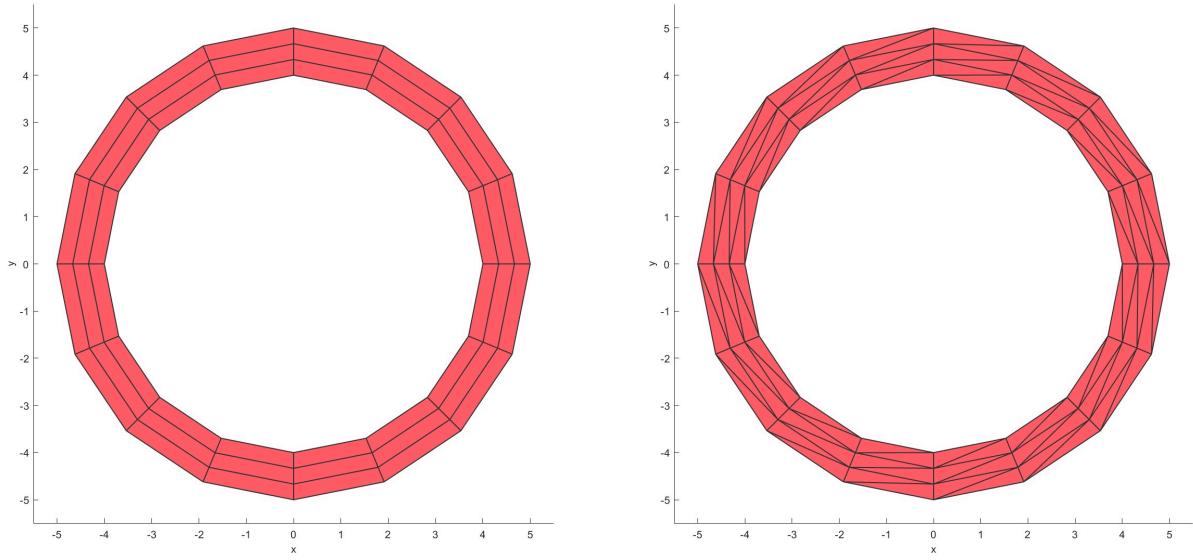
3.19 Mesh ring (Flag=19)

```

1 m1=Mesh2D('Mesh1');
2 m1=MeshRing(m1,4,5);
3 m2=Mesh2D('Mesh2');
4 m2=MeshRing(m2,4,5,'ElementType','tri');
5 Plot(m1);
6 Plot(m2);

```

绘制一个环。



3.20 Mesh edge layer (Flag=20)

```

1 a=Point2D('Temp');
2 a=AddPoint(a,0,0);
3 % Add outline
4 b1=Line2D('Out');
5 b1=AddCircle(b1,14,a,1);
6 % Add innerline

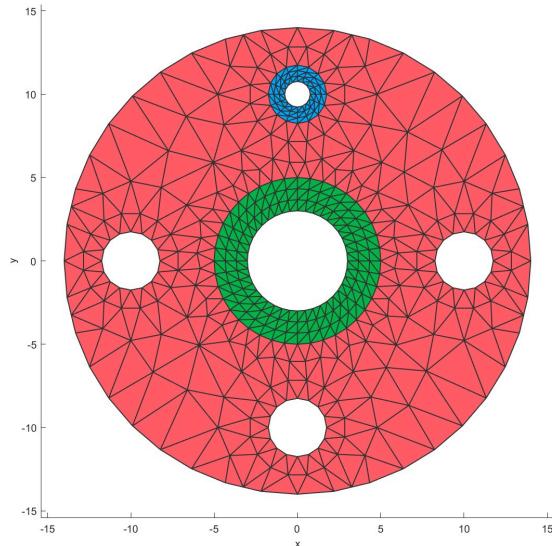
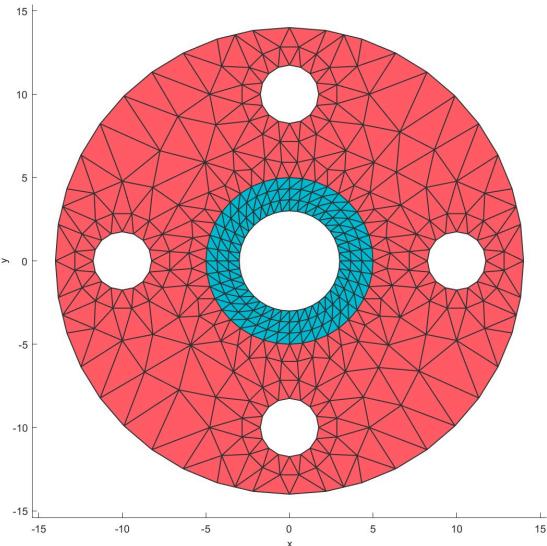
```

```

7 b2=Line2D('Inner');
8 b2=AddCircle(b2,5,a,1,'seg',40);
9 % Add assembly hole
10 h=Line2D('Hole');
11 h=AddCircle(h,3.5/2,a,1,'seg',16);
12 S=Surface2D(b1);
13 S=AddHole(S,b2);
14 for i=1:4
15 S=AddHole(S,h,'dis',[10*cos(pi/2*(i-1)),10*sin(pi/2*(i-1))]);
16 end
17 m=Mesh2D('Temp');
18 m=AddSurface(m,S);
19 m=SetSize(m,5);
20 m=Mesh(m);
21 Plot(m)
22 % Mesh Layer edge
23 m=MeshLayerEdge(m,2,2);
24 Plot(m)
25 % Mesh Layer edge
26 m=MeshLayerEdge(m,3,1);
27 Plot(m)

```

在现有网格的边界绘制层网格。



3.21 Buffalo logo (Flag=21)

```

1 Data=load('Data.mat').Data;
2 a=Point2D('Temp');
3 a=AddPoint(a,Data(:,1),Data(:,2));
4 % Add outline
5 b=Line2D('Out');
6 b=AddCurve(b,a,1);
7 S=Surface2D(b);
8 m=Mesh2D('Temp');
9 m=AddSurface(m,S);
10 m=SetSize(m,20);
11 m=Mesh(m);

```

```

12 | Center = CenterCal(m);
13 | m=AddCellData(m,Center(:,1).^2+Center(:,2).^2);
14 | Plot2(m)

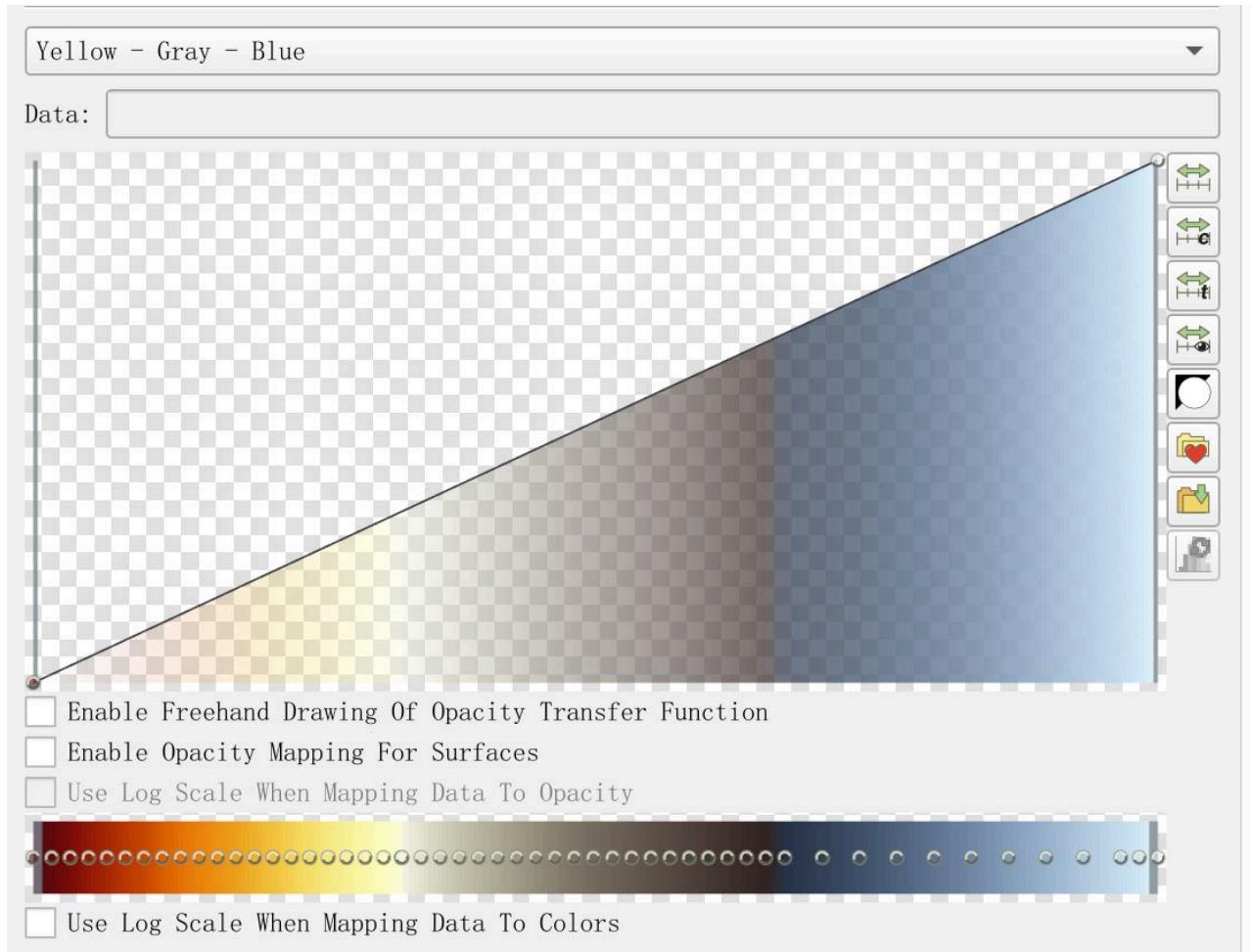
```

首先准备外轮廓的点，绘制曲线和封闭曲面，利用Mesh2D生成网格，导入到Paraview中。

调整背景色，



并设置自己喜欢的色阶，





3.22 Mesh with constraints (Flag=22)

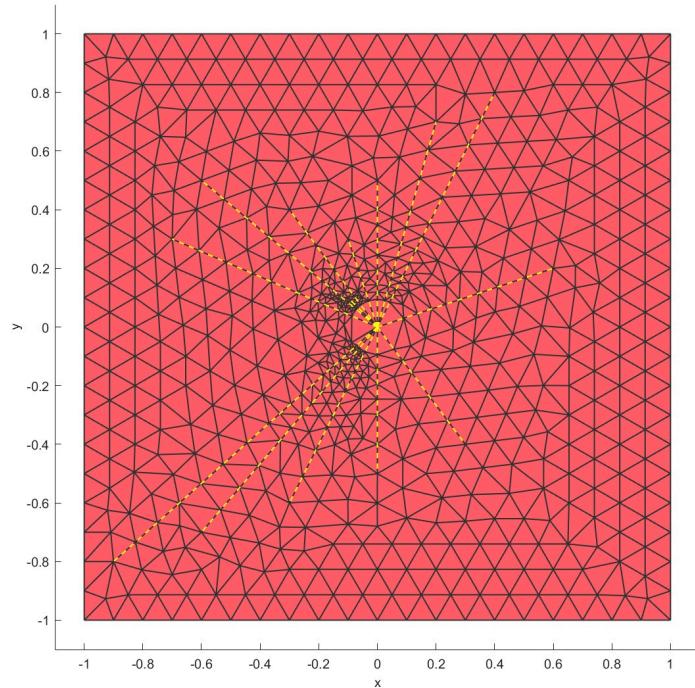
```

1 a=Point2D('Temp');
2 a=AddPoint(a,[1,1,-1,-1,1],[1,-1,-1,1,1]);
3 % Add outline
4 b=Line2D('Out');
5 b=AddCurve(b,a,1);
6 Plot(b)
7 S=Surface2D(b);
8 m=Mesh2D('Temp');
9 m=AddSurface(m,S);
10 m=SetSize(m,0.1);
11 Cnode = [
12     +.0, +.0; +.2, +.7
13     +.6, +.2; +.4, +.8
14     +0., +.5; -.7, +.3
15     -.1, +.1; -.6, +.5
16     -.9, -.8; -.6, -.7
17     -.3, -.6; +.0, -.5
18     +.3, -.4; -.3, +.4
19     -.1, +.3
20 ];
21 Cedge = [
22     1 , 2 ; 1 , 3
23     1 , 4 ; 1 , 5
24     1 , 6 ; 1 , 7
25     1 , 8 ; 1 , 9
26     1 , 10 ; 1 , 11
27     1 , 12 ; 1 , 13
28     1 , 14 ; 1 , 15
29 ];
30 m=AddCNode(m,Cnode);

```

```
31 | m=AddCEdge(m,Cedge);  
32 | m=Mesh(m);  
33 | Plot(m)
```

带约束的网格划分，图中虚线为约束的线段。



4 CheatTable

Name	Varargin	Description
AddCellData(obj,Data)		Add cell data to mesh
AddElements(obj,Face)	'position'	Add elements to Mesh2D
AddPointData(obj,Data)		Add point data to nodes
AddPointVector(obj,Vec)		Add Point vector to nodes
AddSurface(obj,Surface)		Add surface to Mesh2D
CalculateGeometry(obj)		Calculate mesh Geometry information
CenterCal(obj)		Calculate center of elements
ComputeGeometryDual(obj)		Compute Dual mesh information
ComputeGeometryG(obj)		Compute Geometry MRST G information
Convccave(obj,Point)	'logic'	Create convcave
Convert2Order2(obj)		Change element order
Convhull(obj,Point,varargin)	'simplity','keep'	Create convhull
DelNullElement(obj)		Delete Null elements and compress elements
LoadMsh(obj,name)		Load msh file to Mesh2D
FindBoundary(obj)		Find boundary of mesh
EdgeGroup(obj)		Find edge group
MergeNode(obj)		Merge obj face node
Mesh(obj)	'kind','ref1','ref2','siz1','siz2'	Mesh 2D elements
MeshEdge(obj,edgeno)		Mesh edge elemnts
MeshGrid(obj,n,l)	'twist'	Mesh grid elements
MeshLayerEdge(obj,edgeno,Thickness)	'Num','ElementType','Dir'	Mesh layer edge elements
MeshQuadCircle(obj)	'n','r','f'	Mesh a circle using quadrilateral elements
MeshQuadPlate(obj,plateDim,plateEl)		Mesh quad plate
MeshRing(obj,IR,OR)	'Num','ElementType','Seg'	Mesh ring
MeshTensorGrid(obj,x,y)	'twist'	Mesh tensor grid element
PatchBoundaryCentre(obj)		Get center of boundary
Plot(obj)	'grid','axe','equal','base_size', 'xlim','ylim'	Plot Mesh2D object
Plot2(obj)		Plot Mesh2D object in Paraview
PlotDual(obj)	'grid','area','center','axe', 'equal','base_size','xlim','ylim'	Plot dual mesh
PlotG(obj)	'volume','grid','name','axe', 'equal','base_size','xlim','ylim'	Plot MRST G
PlotG2(obj)		Plot MRST G in Paraview
Quad2Tri(obj)	'type'	Covert quad mesh to tri mesh
RemoveCells(obj,cells)		Rmove MRST G cells
ScaleMesh(obj,Scale)		Scale mesh
SetSize(obj,edgesize)		Set mesh edge size
SmoothFace(obj,N)		Smooth face elements
VTKWrite(obj)		Write VTK file of Mesh2D
VTKWriteG(obj)		Write VTK file of Mesh2D G
WriteSTL(obj)		Write STL file of mesh

5 参考文献

[1] https://www.mathworks.com/matlabcentral/fileexchange/25555-mesh2d-delaunay-based-unstructured-mesh-generation?s_tid=srchtile

[2] <https://www.sintef.no/projectweb/mrst/>

[3] https://www.mathworks.com/matlabcentral/fileexchange/48609-dualmesh-polygonal-mesh-construction?s_tid=srchtile