

非结构网格教学程序使用说明

Part 1: Delauney 三角形网格生成程序

主程序文件 **2008Delauney.for** 运行时, 首先从 **INPUT.txt** 文件中读取边界节点, 随后提示用户输入要插入的节点数目。运行结束后, 生成网格文件 **GRID.dat**。此外程序中增加的函数 *subroutine TECPLOT* 会在运行结束后生成一个名为 **TECPLOT.dat** 的文件, 该文件可用绘图软件 Tecplot10.0 打开, 直接查看生成的网格。



图 1 程序运行提示

1. INPUT 文件结构说明

```

XXXXXX .....输入节点总数 (=各条边界上节点数之和+包容矩形节点)
XX  YY.....第 1 条边界上的第 1 个节点坐标
XX  YY.....第 1 条边界上的第 2 个节点坐标
XX  YY.....第 1 条边界上的第 3 个节点坐标
.....
.....
XX  YY.....第 2 条边界上的第 1 个节点坐标
XX  YY.....第 2 条边界上的第 2 个节点坐标
.....
.....
XX  YY.....第 N 条边界上的第 1 个节点坐标
XX  YY.....第 N 条边界上的第 2 个节点坐标
.....
.....
XX  YY.....包容矩形第 1 点坐标
XX  YY.....包容矩形第 2 点坐标
XX  YY.....包容矩形第 3 点坐标
XX  YY.....包容矩形第 4 点坐标
XXXXXX .....区域边界数目
XXXXXX .....第 1 条边界上的节点数目
  
```

XXXXX	第 2 条边界上的节点数目
.....	
.....	
XXXXX	第 N 条边界上的节点数目
XXXXX	区域总的边界节点数 (=各条边界上节点数之和)

2. 网格生成实例

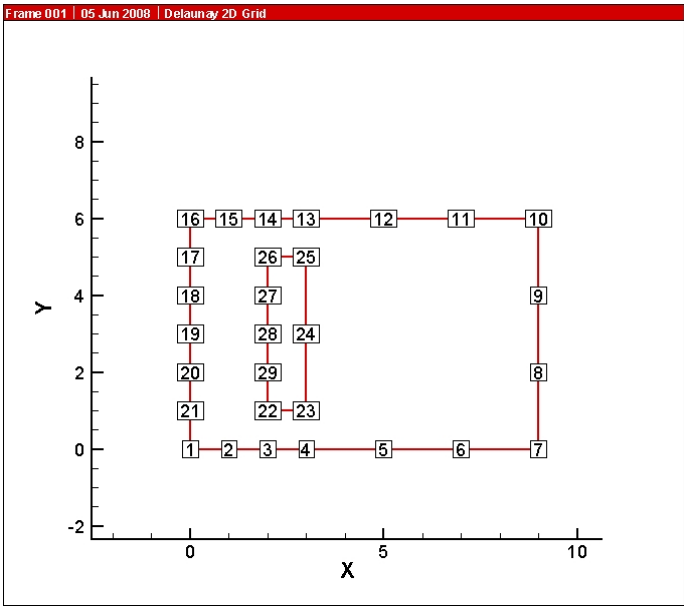


图 2 网格划分区域边界节点设置

以图 2 所示网格划分区域为例，该区域有两条边界：边界 1 上有个 21 节点（点 1 至点 21），边界 2 上有 8 个节点（点 22 至点 29）。

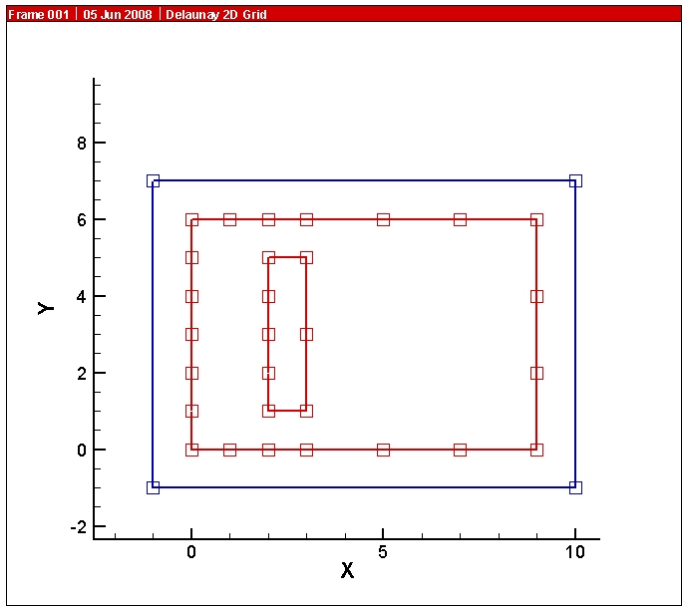


图 3 包容矩形示意

根据 Delauney 初始化的要求，设置包容矩形 4 个节点坐标（图 4 蓝色点），使包容矩形完全覆盖网格划分区域。

该划分区域的 **INPUT.dat** 文件如下。

```
33          // 输入点数 NPP = 所有边界点 + 包容矩形
0 0          // 第 1 条边界：21 个边界点
1 0
2 0
3 0
5 0
7 0
9 0
9 2
9 4
9 6
7 6
5 6
3 6
2 6
1 6
0 6
0 5
0 4
0 3
0 2
0 1
2 1          // 第 2 条边界：8 个边界点
3 1
3 3
3 5
2 5
2 4
2 3
2 2
-1 -1        // 初始化包容矩形
10 -1
10 7
-1 7
2           // 边界数目 N
21          // 各边界点数 NDOMAIN
8
29          // 边界点总数 NBOUND
```

程序运行结果见图 4，图 5

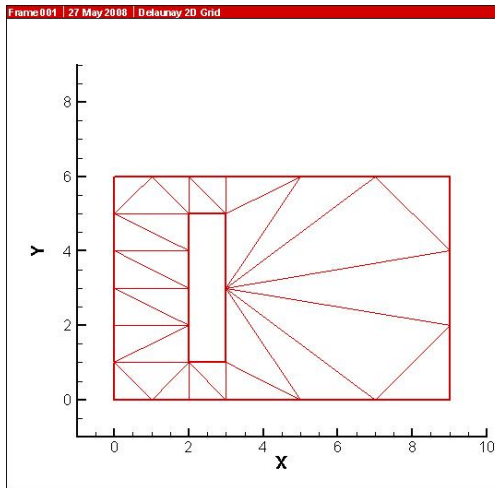


图 4 不插入节点的网格生成结果

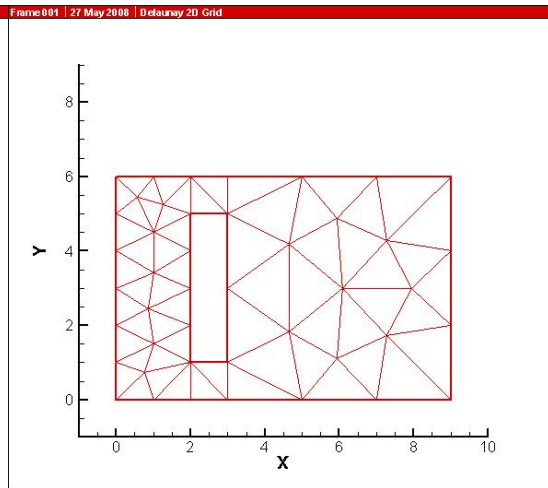


图 5 插入 15 个节点的网格生成结果

备注：程序运行中出现 "MISSHAPEN TRIANGLE WARNING"提示时，不会中断网格生成，只是在边界会产生质量较差的网格，说明需要增加边界节点的数目。

Part 2: 非结构化网格求解程序

主程序 **2008Unstruct.for**，除更改计算格式以外（程序提供 FUD, SUD, CD 三种格式），通常不需要改动，所有计算参数和边界条件在用户程序 **User.for** 中进行设置，计算区域的网格文件 **GRID.dat** 可由前述网格生成程序获得。程序所附的 **User.for** 和 **GRID.dat** 为顶盖驱动流算例，计算结果保存在 **RESULT.dat** 中，同时该程序中也提供 *subroutine* **TECPLOT** 函数，该函数生成 **TECFEILD.dat** 文件，用于计算结果在 Tecplot10.0 中的显示。

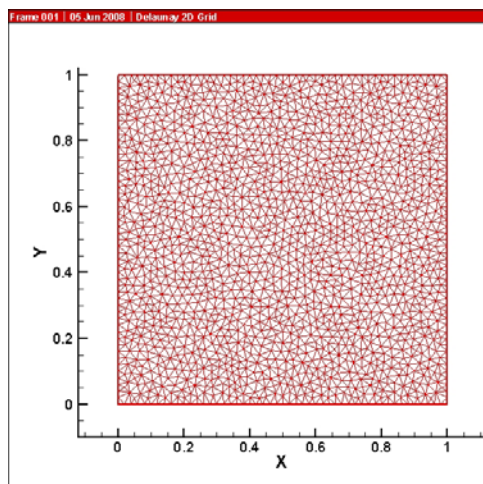


图 6 计算网格

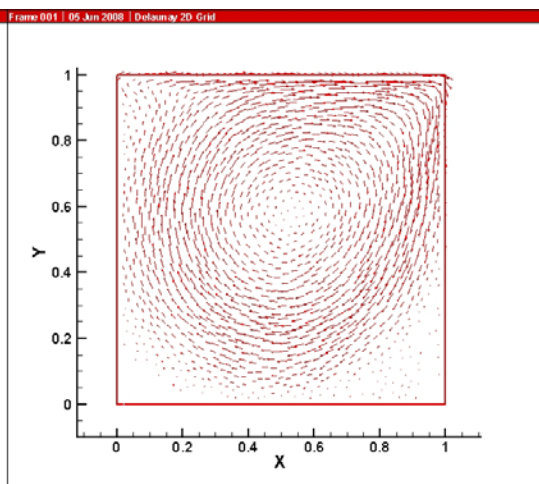


图 7 速度场计算结果