开发文档

## 功能介绍

1. 读取.txt数据文件。
2. 正确求出导入数据散点集的凸包点。自定义网格的大小，在网格中绘制出凸包图形，并可保存为.dxf文件。
3. 可以实现凸包图片在程序窗口的基本操作。
4. 可自定义基准高程，求出凸包内高于基准高程的体积。并形成报告，可以输出保存为.txt文件。
5. 在软件的左下角有切换窗口的快捷按钮，可通过这里进行切换数据窗口，图形窗口，报告窗口。

## 算法设计

求出凸包算法一——Graham’s Scan法 （S是凸包点集）

1. 取散点集（Q）中Y最小，如果Y坐标相同最其中X坐标最小的点为p0.
2. 把（p1,p2,p3,p4,……，pn）按照极角从小到大排序（p0为极点），极角相同的点按照到p0的距离从小到大排序。
3. PUSH(p0,S)
4. PUSH(p1,S)
5. PUSH(p2,S)
6. For I 🡨3 to n
7. Do while S的次顶点和顶点和pi形成右转
8. Do POP(S)
9. PUSH(pi,S)
10. Return S

求出凸包算法二——快速凸包法 （S是凸包点集，Q是散点集,C是散点集,L为线段（L.top,L.tail））

1. 求出Q中的X最小的点p1，Y最大的点p2，X最大的点p3，Y最小的点p4.
2. PUSH(p1,S)
3. For i 🡨1 to 4
4. L.top=pi, L.tail=p(i+1)
5. 点集C 🡨 在Q中且在L左侧的点
6. If C为空
7. PUSH( p(i+1),S)
8. else
9. 快速凸包( C,L )
10. return S
11. Void 快速凸包( 点集C, 线段L){
12. If C为空
13. PUSH( L.tail,S) break
14. 选取 C 中距离 L 最远的 点 Y
15. 向量 A <- { L.top , Y } ; 向量 B <- { Y , L.tail }
16. 点集 D <- 在 C 中 且在 A 左侧的点
17. 点集 R <- 在 C 中 且在 B 左侧的点
18. 快速凸包(D,A)
19. 快速凸包(R,B)
20. }

## 主要函数和变量说明

1. 主要变量

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 变量名 | 类型 | 含义 | 变量名 | 类型 | 含义 |
| g\_point | given\_point | 已知散点 | convex\_hull | given\_point | 凸包点 |
| height\_datum | float | 基准高程 | minmax | float[,] | 散点中的最大最小的x，y值 |
| grid\_spacing | float | 网格边长 |  |  |  |

1. 结构体

given\_point 已知点的结构体（点名，X坐标，Y坐标，H高程）

{ public string point\_name;

public float X;

public float Y;

public float H;}

point //建立新的结构体(有关基点到各点的信息)

{ public given\_point p;

public double p0\_pn\_D; //到各点的信息

public double p0\_pn\_angle; //到各点的角度 }

1. 类及主要函数

public partial class 规则网格进行体积计算

{ void diff\_grid\_size\_report(int grid\_spacing) //生成不同网格边长的报告

void openfile\_Click(object sender, EventArgs e) //打开读取数据文件

void savereport\_Click(object sender, EventArgs e) //保存报告

void save\_DXF\_Click(object sender, EventArgs e) //保存图形为.dxf文件

void 生成凸包\_Click(object sender, EventArgs e) //图形显示

}

class 体积计算

{ int get\_in\_gird\_num(List<given\_point> convex\_hull) //输出有多少格网点在凸包内

float[,] out\_matrix(float[,] minmax, float grid\_spacing) //输出外包矩形的四角坐标

int judge(grid\_point a, given\_point b, given\_point c) //判断点a是否在b—c线的左边

double get\_V(double r, float grid\_spacing, float height\_datum, List<given\_point> given\_point) //得到总的体积

}

class 生成凸包

{ point get\_p0() //得到p0基点

double get\_D(point a, point b) //得到两点之间的距离

double get\_angle(point a, point b) //得到两点的向量与x轴的夹角

void po\_pn\_sort() //对向量<p0,pi>进行排序

double judge\_l\_r(point a, point b, point c) //判断左转还是右转

List <given\_point > get\_convex\_hull() // Graham’s Scan法得到凸包点集

List<given\_point> conver\_h1(List<given\_point> k\_point) //快速凸包法得到凸包点集

int judge\_left(given\_point p1, given\_point p2, given\_point p3) //判断点是否在线的左边

}

class DRAW

{ Bitmap get\_picture(List<given\_point> g\_point, List<given\_point> convex\_hull,float grid\_spacing,float[,]minmax) //在画布上画出图形

void save\_dxf(string path ,List<given\_point> g\_point, List<given\_point> convex\_hull, float[,] minmax,float grid\_size) //保存数据为.dxf文件

void point(StreamWriter sw, float ax, float ay, string layer\_name) //.dxf文件中画点

void point(StreamWriter sw, float ax, float ay, string layer\_name)//.dxf文件中画点

void text(StreamWriter sw, float ax, float ay, string layer\_name, string txt, float size)//.dxf文件中写文字

void line(StreamWriter sw, float ax, float ay, float bx, float by, string layer\_name)//.dxf文件中画线

void layer(StreamWriter sw, string layer\_name, int layer\_color) //.dxf文件中定义图层

}

## 主要程序运行界面

1. 1.报告：

-------------------------基本信息---------------------

基准高程： 9

网格间隔： 1

网格横格： 43

网格纵格： 26

总网格数： 1118

凸包内的网格数： 651

体积： 1217.252

-----------点位信息---------

外包矩形的顶点坐标：

X坐标 Y坐标

3757.095 2845.174

3757.095 2870.586

3799.283 2870.586

3799.283 2845.174

-------------------------基本信息---------------------

基准高程： 9

网格间隔： 5

网格横格： 9

网格纵格： 6

总网格数： 54

凸包内的网格数： 26

体积： 1220.828

-----------点位信息---------

外包矩形的顶点坐标：

X坐标 Y坐标

3757.095 2845.174

3757.095 2870.586

3799.283 2870.586

3799.283 2845.174

-------------------------基本信息---------------------

基准高程： 9

网格间隔： 10

网格横格： 5

网格纵格： 3

总网格数： 15

凸包内的网格数： 8

体积： 1469.775

-----------点位信息---------

外包矩形的顶点坐标：

X坐标 Y坐标

3757.095 2845.174

3757.095 2870.586

3799.283 2870.586

3799.283 2845.174

报告基点是:

点号 X坐标 Y坐标 H高程

P31 3757.095 2846.279 10.65

--------------------------凸包点----------------------

点号 X坐标 Y坐标 H高程

P31 3757.095 2846.279 10.65

P10 3779.115 2869.933 9.38

P11 3785.879 2870.586 9.528

P12 3792.979 2870.191 9.554

P02 3795.959 2868.952 9.615

P03 3799.283 2864.64 9.977

P04 3799.039 2856.23 10.102

P05 3795.293 2853.8 10.115

P28 3778.428 2845.174 10.354

P29 3772.222 2845.423 10.604

P31 3757.095 2846.279 10.65

4.2 用户界面



图1 数据读取显示



图2 报告界面

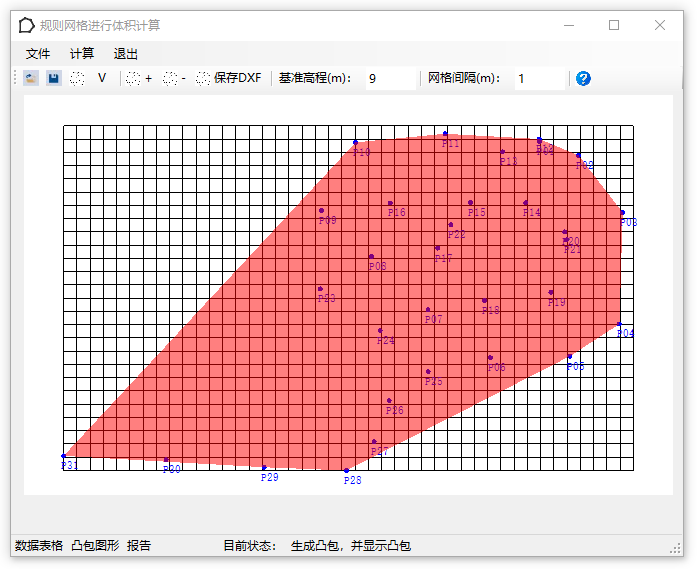


图3 凸包图形显示

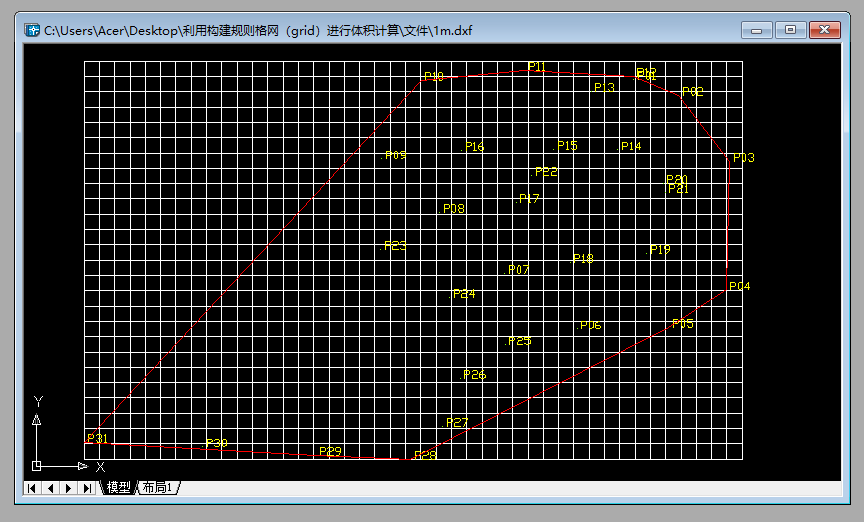


图4 在cad中打开的图形

## 使用说明

1. 打开软件，点击打开文件，出现文件对话框，把要计算的文件导入到软件中
2. 点击绘图，可以形成凸包图形，也可以点击体积计算，直接形成报告。
3. 在软件的左下角有切换窗口的快捷按钮，可通过这里进行切换数据窗口，图形窗口，报告窗口。
4. 点击保存按钮可以保存报告到.txt文件
5. 点DXF保存可以把图形输出保存到.dxf文件中。