

# 基于地图信息的三维仿真

邹 衡 (2009210895) 王晓欢 (2010210878) 陈树勇 (2010210861)

计算机科学与技术系 清华大学 北京 100084

## 一、实验背景

实现三维数字地形建模及地形的三维显示是地理信息系统的关键技术之一。在三维数字地形建模中，利用地图的等高线信息，提取散乱点及高程信息，并对散乱点进行剖分、三维建模、地理信息匹配和还原、真实场景漫游等，实现对地图信息的真实还原和操作。我们结合计算几何课程的相关知识，对这个问题进行了深入研究。我们应用双向链接边表（DCEL）数据结构对 Delaunay 三角剖分（分治算法）算法进行了实现，并应用 OpenGL 绘制出了真实感三维地形的基本信息。下面我们对实验的内容、算法、数据结构、结果及相关问题进行阐述。

## 二、实验过程描述

在本实验中，我们利用从地图等高线中提取出的离散点信息进行建模，下面是实验的实现的步骤及过程：

（一）我们从地图等高线信息中，提取出三维散乱点数据和地形高程数据，将此数据投影到二维欧氏空间，形成离散的数据点。再应用 Delaunay 三角剖分分治算法对二维散乱点进行剖分，得到具有最小内角最大化性质的最优三角剖分。

（二）根据每个离散点高程信息，将剖分后的结果还原到三维空间，建立起与真实地理信息相仿的三维空间的三角剖分。

再对所得三维空间剖分进行纹理绘制，还原出真实三维地形信息。

(三) 在还原得到的三维地形中，通过漫游可以得到真实的地形感受和场景仿真。

### 三、算法分析

#### (一) Delaunay 算法分析

Delaunay 算法实现大致分为三种：分治算法、逐点插入法、边生长算法。由下表可以看出，边生长算法在 80 年代中期以后就很少用到，较常见的是分治算法和逐点插入法，虽然逐点插入法实现过程简单，所需内存较小，但它的时间复杂度高。所以从时间复杂度看，分治算法最好。

表1 几种Delaunay三角网生成算法的时间复杂度

算法		一般情况	最坏情况
分治算法	Lewis和Robinson(1987)	$O(N\log N)$	$O(N_2)$
	Lee和Schachlter(1980)	$O(N\log N)$	$O(N\log N)$
	Dwyer(1987)	$O(N\log\log N)$	$O(N\log N)$
	Chew(1989)	$O(N\log N)$	$O(N\log N)$
逐点插入法	Lawson(1977)	$O(N_{4/3})$	$O(N_2)$
	Lee和Schachlter(1980)	$O(N_{3/2})$	$O(N_2)$
	Bowyer(1981)	$O(N_{3/2})$	$O(N_2)$
	Watson(1981)	$O(N_{3/2})$	$O(N_2)$
	Sloan(1987)	$O(N_{5/4})$	$O(N_2)$
边生长算法	Green和Sibson(1987)	$O(N_{3/2})$	$O(N_2)$
	Brassel和Reif(1979)	$O(N_{3/2})$	$O(N_2)$
	MaCullagh和Ross(1980)	$O(N_{3/2})$	$O(N_2)$
	Mirante和Weigarten(1982)	$O(N_{3/2})$	$O(N_2)$

#### (二) 数据结构：双向链接边表 (Doubly-Connected Edge

## List, DCEL)

DCEL 结构由点、面、半边三类记录构成，其详细说明 如下：

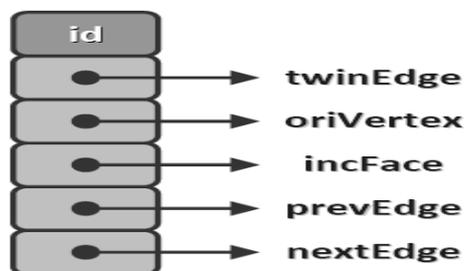
顶点记录，包含三个域 ID、Coordinates、IncidentEdge，分别存放顶点索引、顶点坐标、指向以该顶点为起点的某一半边的指针；



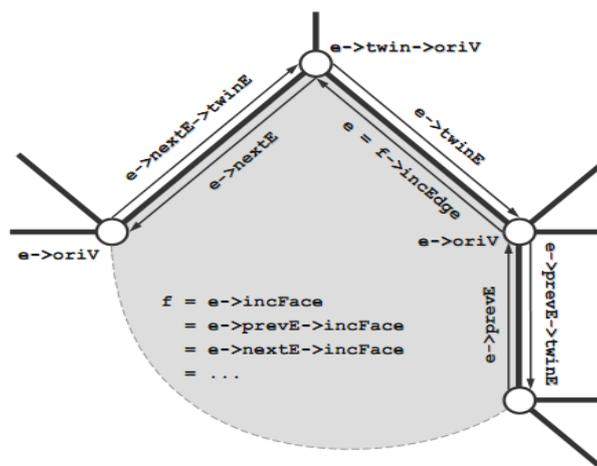
面记录，包含两个域 ID、OuterComponent，分别存放面索引、指向该面外边界上的任意一条半边。



半边记录，包含个域 ID、Origin、Twin、IncidentFace、Next、Prev，分别存放指向该半边的起点的指针、指向其对偶边 (Twin) 的指针、指向其参与围成面的指针、指向其后继边的指针、指向其前趋边的指针。

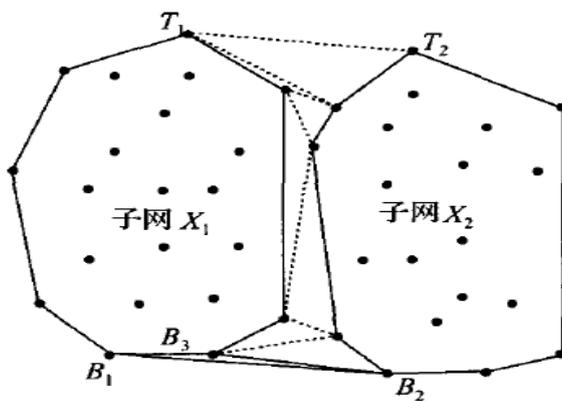


其点、面、半边数据关系如下图：



### (三) Delaunay 算法

Delaunay 分治算法：把点集  $P$  以横坐标为主，纵坐标为辅按升序排序，然后递归地执行以下步骤：把点集  $P$  分为近似相等的两个子集  $X_1$  和  $X_2$ ；在  $X_1$  和  $X_2$  中生成三角网；找出连接  $X_1$  和  $X_2$  中两个凸壳的底线和顶线；由底线至顶线合并  $X_1$  和  $X_2$  中两个三角网。算法示意图如下：



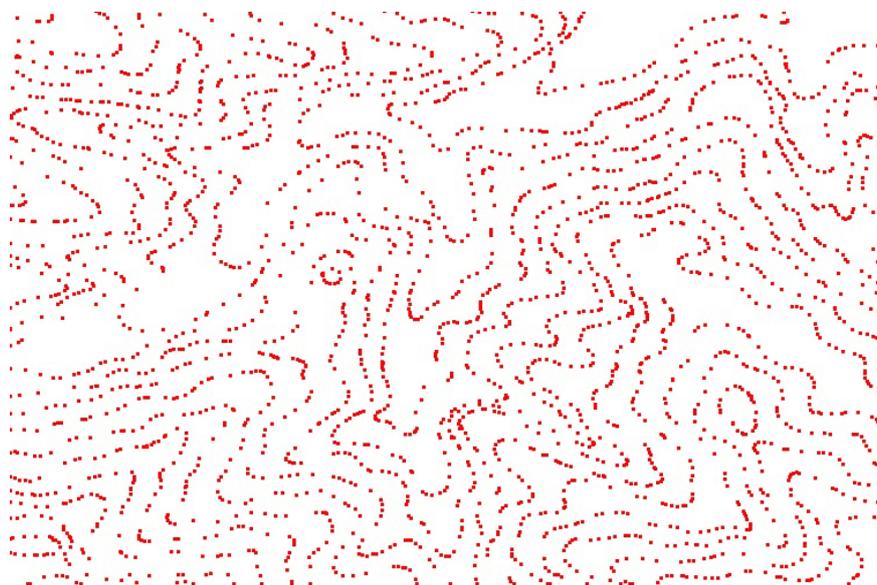
### (四) 复杂度

本实验对输入数据的处理上用线性时间即可完成。采用的 Delaunay 分治算法，运用局部最优全局最优的思想，其时间复杂度为  $O(n \log n)$ ，但需要更多的内存空间。对二维的剖分结果还原为三维地形模型，也只需线性时间，故本实验可在  $O(n \log n)$

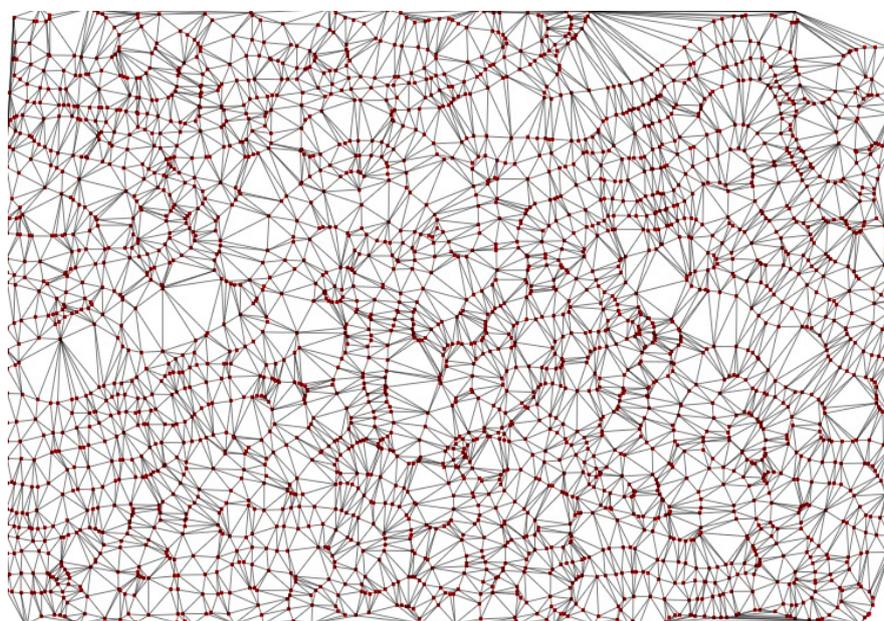
时间内完成对地图信息三维仿真。

#### 四、实验结果展示

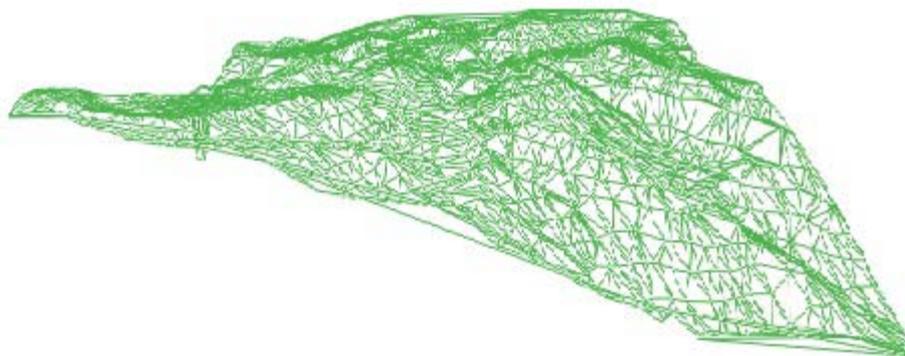
##### (一) 地形文件导入



##### (二) Delaunay 三角剖分



##### (三) 三维三角网显示



#### (四) 粘贴纹理后三维实体显示



按动方向键，可以在三维地形中漫游

### 五、前景展望

我们在实验中结合课堂知识对三维地形网格化及三维真实感绘制做了初步的探讨，实现了基于三维地形数据的三角剖分，应用 OpenGL 绘制出了三维真实感地形。利用我们实验的数据结构与结果可以很容易实现其它诸如三维地形漫游、场景仿真等操作，这些在工程实施、灾害评估、军事指挥中有非常广泛的应用。

### 六、实验总结

在这次实验过程中，我们组按照老师的要求，积极配合，认真努力，按期完成了实验任务。通过这次实验我们总结如下：

（一）实验中，我们对二维平面剖分采用了多种尝试：

1) 对平面点纵向平分，将每个区域内点纵向连接起来，得到一个单调链，将相邻两个单调链按单调多边形来进行剖分，再用 Delaunay 性质优化。

2) 采用三角网生长算法：以任一点为起点；找出与起始点最近的数据点相互连接形成三角形的一条边作为基线，找出与基线构成三角形的第三点；基线的两个端点与第三点相连，成为新的基线；迭代以上两步直至所有基线被处理。

最后，通过比较发现：上述两种算法在实现剖分效率上比分治算法相差较大。

尽管在实验过程中走了许多弯路，但每次发现问题、解决问题的经历，我们都在成长提高。

（二）注意利用理论知识去发掘、思考、理解并解决实际问题。我们一开始的选题仅是想做一下关于三维点集的三角剖分，如果仅是就某个具体问题进行的话，实验的内容不够丰富，实验中所考虑问题的面也比较窄，缺乏全面性，我们把实际地形问题引进来，在定义数据、设计算法时比仅考虑三维点集要更全面、合理、可靠。

（三）加强沟通、明确分工、积极配合。在实验开始阶段我们小组就明确了每个成员的分工，组长邹衡理论及开发水平

都较高，负责总体工作，具体负责总体框架的设计、主要数据结构的设计、主要算法的设计、数据结构算法实现的测试及修改等工作；组员王晓欢 C++语言应用较熟练，主要负责数据结构、算法的设计实现工作等；组员陈树勇用户应用开发较熟悉，主要负责界面的设计及文档草稿的编写等。在实验过程中我们组定在每周二晚上七点召开小组碰头会，总结上周工作进展并制定下周工作计划。同时，我们在实验过程中非常注意加强之间的协作，积极的互相学习，取长补短，做到每个人对实验的每个细节都掌握。我们在加强小组内部的交流的同时，也注意同其他组同学的交流，听取他们的意见建议，把他们好的方法经验吸收过来。与此同时我们更注重与老师的交流，利用答疑等机会，积极地向老师咨询我们的算法等设计的可行性及效率等，老师对我们的设计提出了很多好的意见建议，对我们改进算法复杂度、提高运行效率、加强系统的稳定性起到了非常关键的作用。

同时，在这次实验中我们也发现诸如时间进度安排不合理、各子任务工作量估计不足、对可能遇到的困难准备不充分等问题。总之，通过这次实验我们加深了对课堂知识的理解，研究了与之相关实际应用问题初步解决方案，发现了自身存在的不足与缺陷，收获了学习工作的经验方法，这对于我们以后的发展进步一定会有很大的帮助。

## 七、参考文献

[1]邓俊辉 计算几何讲义 2010

[2]Mark de Berg 等著, 邓俊辉译. 计算几何算法与应用(第 3 版). 清华大学出版社. 2009

[3] J. O' Rourke. Computational Geometry in C. Cambridge University Press, Dec. 1998, 2nd edition

[4]张永春 达飞鹏 宋文忠. 三维散乱点集的曲面三角剖分 中国图象图形学报 第 8 卷(A 版)第 12 期 P1379-1388, 2003