第36卷 第3期
2007年8月
测 绘 学 报
ACTA GEODAETICA et CARTOGRAPHICA SINICA
Vol.36, No.3
Aug, 2007

文章编号:1001-1595(2007)03-0358-05 中图分类号:P208 文献标识码:A

Delaunay 三角形构网的分治扫描线算法

芮一康, 王结臣
(南京大学 地理信息科学系, 江苏 南京 210093)

A New Study of Compound Algorithm Based on Sweepline and Divide-and-conquer Algorithms for Constructing Delaunay Triangulation

RUI Yi-kang, WANG Jie-chen
(Department of Geographic Information Science, Nanjing University, Nanjing 210093, China)

Abstract: As one of the most important DTM model, Delaunay triangulation is widely applied in manifold fields. A wide variety of algorithms have been proposed to construct Delaunay triangulation, such as divide-and-conquer, incremental insertion, trangulation growth, and so on. The compound algorithm is also researched to construct Delaunay triangulation, and prevalently it is mainly based on divide-and-conquer and incremental insertion algorithms. This paper simply reviews and assesses sweepline and divide-and-conquer algorithms, based on which a new compound algorithm is provided after studying the sweepline algorithm seriously. To start with, this new compound algorithm divides a set of points into several grid tiles with different dividing methods by divide-and-conquer algorithm, and then constructs subnet in each grid tile by sweepline algorithm. Finally these subnets are recursively merged into a whole Delaunay triangulation with a simplified efficient LOP algorithm. For topological structure is important to temporal and spatial efficiency of this algorithm, we only store data about vertex and triangle, thus edge is impliedly expressed by two adjacent triangles. In order to fit two subnets merging better, we optimize some data structure of sweepline algorithm. For instance, frontline and baseline of triangulation are combined to one line, and four pointers point to where maximum and minimum of x axis and y axis are in this outline. The test shows that this new compound algorithm has better efficiency, stability and robustness than divide-and-conquer and sweepline algorithms. Especially if we find the right dividing method reply to different circumstance, its superiority is remarkable.

Key words: Delaunay triangulation; compound algorithm; sweepline algorithm; divide-and-conquer algorithm

摘 要:Delaunay 三角网作为一种主要的DTM表示法,具有极其广泛的用途。基于分治算法和逐点插入法的合成算法是目前研究较多的用于生成Delaunay三角网的合成算法。简要介绍和评价扫描线算法和分治算法后,提出一种新的基于这两种算法的合成算法。该方法兼顾空间与时间性能,稳定性较高,分别较扫描线算法和分治算法,运行效率和鲁棒性更优。

关键词:Delaunay 三角网;合成算法;扫描线算法;分治算法

1 引 言

2维平面域内任意离散点集的不规则三角网(TIN-Triangular Irregular Network)的构建是GIS数据表达、管理、集成和可视化的一项重要内容,也是地学分析、计算机视觉、表面目标重构、有限元分析、道路CAD等领域的一项重要的应用技术。在所有生成TIN的方法中,Delaunay三角网最优,它尽可能避免了病态三角形的出现,常常被用来生成TIN。Delaunay三角网是Voronoi图的直线对偶图,即是连接所有相邻的Voronoi多边形的生长中心所形成的三角网。它有以下两条重要性质[1]:空外接圆性质,即由点集所形成的三角网中,每个三角形的外接圆均不包含点集中的

收稿日期:2006-06-21;修回日期:2007-02-06
基金项目:国家自然科学基金(40401046)
作者简介:芮一康(1983-),男,江苏溧阳人,研究生,主要从事地理信息系统理论与应用研究。

其他任意点;最大最小角性质,即由点集所能形成的所有三角网中,Delaunay 三角网中三角形的最小内角角度是最大的。

目前常见的构建 Delaunay 三角网的算法有[2~4]:分治算法,逐点插入算法,生长算法,扫描线算法和凸壳算法。由于这些算法各有优劣,为了体现各个算法的优势,提高算法在处理大量数据时的时空性能,一个很自然的想法就是将算法综合起来,相互取长补短。目前研究较多的合成算法是把逐点插入法植入到分治算法中[5~7],该算法编码简易,且有较高的执行效率。考虑到在上述方法中,扫描线算法较生长、插入等方法具有更好的稳定性和算法复杂度,作为一次研究尝试,本文在研究扫描线算法和分治算法后,提出了分治扫描法,即把扫描线算法和分治法结合起来,以期获得较好的算法性能。

2 分治扫描法

Steven Fortune 于1987 年在 Algorithmica 上发表了一篇基于平面扫描思想的 V- 图生成算法的文章[8]。之后,J. R. Shewchuk 等人均按此思路实现了 D- 三角网的扫描线算法,但各有不同。此算法主要是把离散点先按某一坐标值(如 y 值)排序,然后用一条垂直于该坐标轴的直线扫过点集平面,并在所谓事件点(event points)处停下,处理每种事件,添加三角形,直到完毕。扫描线算法最坏情况的时间复杂度可达 O(Nlog N),且对内存要求不高。

Lewis 和 Robinson 首先将分治算法思想应用于生成 D- 三角网[9]。分治算法的思路是把整个离散点集分割为数个较小的点集,在各个小点集内生成小三角网,然后再逐级合并。整个过程是一个递归的过程。L. Guibas 和 J. Stolfi 给出了一个最坏情况下时间复杂度为 O(Nlog N)的算法[10]。分治算法的优点是时间效率高,但实现较为困难。由于大量使用递归,内存需求也较大。

与插入法和分治法结合相比,分治扫描算法除了具有良好的执行效率和稳定性外,也很好地解决了插入和分治合成算法子模块的一些"瓶颈"问题,如凸壳计算,因为扫描线算法在局部子网构建的同时也解决了凸壳生成问题。同时每块子网在优化后可释放部分内存,这为处理大数据量时,提供了一种缓解内存不足的途径。

分治扫描法的基本思路是:划分点集,扫描线法生成子网,子网合并。其关键过程及实现方法分述如下。

2.1 划分点集

分治算法分割子网的通常方案是:当点集内的点数小于一给定的阈值时,生成子网;当点集内的点数超过给定阈值时,如果点域 X 轴方向的长度大于 Y 轴方向的长度,则以 X 轴方向对半分割点集,否者以 Y 轴方向对半分割点集。这是一个递归调用的分割过程。

2.2 子网生成

2.2.1 扫描法算法的基本步骤

子网的生成主要使用扫描线算法。在三角网数据结构中,拓扑关系的表达最为重要。拓扑结构的组织关系到算法的执行效率和空间需求。在这方面有两类经典的数据组织:以边为主体的结构和以三角形为基本的结构。本方案使用基于三角形结构的数据组织。这种方案较为节省空间,仅存储顶点和三角形这两种结构,边结构由相邻三角形隐式表达。每个三角形存储三个顶点的序号,同时存储对应的三个相邻三角形的指针。

此算法主要步骤如下:首先把离散点先按某一坐标值(如 Y 值)排序,同时建初始三角网,把已生成的三角网的外围分为壳底和前线,壳底和前线的分界是已处理点集中 X 值最小和最大的点,以此两点为界,所谓下部的折线为壳底,它总是凸壳的一部分,上部的折线即前线,它把已扫描与未扫描的点分开。

然后用一条平行于该坐标轴的直线扫过点集平面,从最新扫入点往前线做垂直于 x 轴的直线,并在所谓事件点处停下,处理每种事件直到完毕。点事件流程图如图1 所示,分为四种具体的处理情况。在三角网中每加入一个新的三角形,都要建立与其相邻的三角形之间的拓扑关系,同时更新前线与底壳。为了保证生成的三角网满足 D- 三角网要求,需要进行空圆测试,这一步也可放在不规则三角网生成之后,进行整体优化。

在所有点都扫描之后,前线上各点必须充分地构造三角形,以满足 D- 三角网的外围是凸多边形这一性质。

凸壳完成,也就意味着 D- 三角网生成完毕。

2.2.2 扫描线算法中数据结构的改进

现有方法中底壳参与计算较少,因此其结构不如前线来得严谨。而在新方法中,考虑后面子网合并的需要,此处把前线和底壳合二为一条双

向链表,即为子网的外壳,链表上的每个节点都存储有指向其右侧的最外围的三角形的指针;同时用4 个指针分别定位外壳中 X,Y 值最大、最小的4 个离散点,既是用来区分外壳中的前线和底壳,又可作为子网合并的初始和结束点。

图1 扫入点情况判断流程图

Fig.1 Flowchart of processing new point

外壳前线部分点的搜索用二叉排序树来实现。尽管二叉平衡树的查找长度更稳定,但由于其在建树和动态插入上的“平衡化”问题,会增加相应的算法复杂度以及系统在时间和空间上的开销。而且考虑到扫描点是离散随机的,一般情况下可以认为其平均查找长度是和二叉平衡树等量级的。

2.3 子网合并

分治算法的合并是个递归的过程,包括左右子网合并和上下子网合并,方法类似。子网合并的主要步骤如下(以左右子网 X_1 和 X_2 合并为例),如图2 所示,假设子网内三角网已经构建,外壳可得:

1. 获取底边 B_1B_2(B_1 在子网 X_1 上,B_2 在子网 X_2 上)。扫描线算法中已经得到了两子网 X_1 右凸壳的最底点和 X_2 左凸壳的最底点,所以底边获取比较容易,但如果待合并的网间有点在线 B_1B_2 之下时,就需把 B_1 适当右移,或者 B_2 适当左移。

2. 获取顶边 T_1T_2,方法同(1)。

3. 从底边 B_1B_2 向上搜索新的顶点 B_3,产生新三角形 $B_1B_2B_3$,并更新底边,当 B_3 在子网 X_1 上时,新的底边为 B_2B_3,反之用 B_1B_3 取代底边。至于 B_3 的选择,尽量选取 X_1 右凸壳和 X_2 左凸壳上 y 值较小的点,这样能保证得到的三角形尽量最优,减少后面 LOP 优化的次数,同时要保证新的底边不能和凸壳有相交。三角形优化工作最后一并进行。凸壳上每个点都有指向其所在三角形的指针,在构建网间三角形时,也要建立新旧三角形之间的拓扑关系。

4. 重复执行该过程,当新的底边与顶边 T_1T_2 重合时终止。

图2 子网间的合并示意图

Fig.2 Sketch map of two subnets merging

2.4 构建外围及局部优化

每个子网都需要进行构建外围和局部优化操作。

由于D- 三角网的外围是凸多边形,所以外围上各点必须充分地构造三角形以满足此形状。方法简述如下:从外围上的一点开始,迭代地取连续3 点,判断其中第二点在一、三两点定义直线什么位置。若在下方,那么就构造新三角形,然后更新外壳。下一次就从外围的第二点取连续3 点,同上处理,直到回到开始构建的那一点结束。一般情况下只有前线在完成后会有不凸的情况,所以开始点和结束点可以定义为前线的两个端点。

初始三角网建成后,就要对所有三角形进行局部优化,需要遍历该三角网。鉴于三角形之间的拓扑关系,三角网可认为是个图结构,图的遍历有深度优先遍历和广度优先遍历算法。深度优先遍历算法采用递归法比较简单易懂,但是当离散点很多、数据量很大时,由于系统栈的大小受限,不可能无限的递归下去,超过一定的递归层数堆栈就会溢出。若采用广度优先算法的层次遍历,将未优化的三角形放入一个先进先出的链表,每个三角形与其三边的指针指向的三角形进行 LOP 优化。如果链表中排队等待优化的三角形被前面的三角形优化掉了,就要进行适当的标记,但删除操作得放在优化结束之后进行。由于初始

三角网的三角形是以 Y 轴方向依次建立的,可在建网时把这些三角形顺序串连起来,LOP 优化的方向控制在向下进行。

网间合并所生成的三角形也要放入链表中依次进行LOP 优化。删除每个子网中优化掉的三角形,也可以对每块子网中的三角形进行分类[11],释放掉那些在子网合并中不会受影响的三角形所占的内存,以便在处理大数据量时缓解内存不足问题。

利用Delaunay 三角网的空圆准则,对初始三角网进行LOP 优化,目前常见的做法是:利用第四点到其他三点构成的外接圆的圆心距离和外接圆的半径关系进行判定。这一过程中要多次执行乘、除等执行效率比较低的操作,因此有必要在保证计算精度和稳定性的前提下,提出一个简化的空圆测试的算法。有学者提出利用两矢量的数量积公式计算判断角的方法,以减少运算量[7]。

为了保证算法的稳定性,应注意当点落在三角形外接圆的边上时,计算该四边形的对角线,取其较短的一条即可。

3　测试与分析

下面用一个 MFC 程序测试本算法,测试对象为:随机离散分布的点集,测试平台为:Celeron CPU 2.8 GHz,可用物理内存近400 MB,操作系统是 Windows XP。时间并不包含输入/输出时间。

取分割阈值为随机离散点总数的1/3,即分裂2 次,子网个数为4。测试如表1 所示。

表1　合成算法的点数与处理时间对比表

Tab.1　Relationship between process time and point number of the compound algorithm

点数/万	5	10	20	30	40	50	100
运行时间/s	1.05	2.21	4.47	6.79	9.16	11.48	24.21

分析各主要算法的时间复杂度,扫描线算法和分治法的最坏情况的时间复杂度都为 $O(N\log N)$,而插入法和生长法的最坏情况下的时间复杂度都为 $O(N^2)$。可预见的是扫描线算法和分治法的结合会带来较优的时间复杂度。

经测试,显而易见的是,随着点数的增加,此合成算法的运行时间几乎呈线性增长,可见此算法具有较好的时间性能。图3 是由表1 所作的不同点数 Delaunay 三角形构网的线性拟合曲线。

图3　合成算法的时间性能测试

Fig.3　Performance of the compound algorithm

为了将此合成算法与分治算法和扫描线算法进行比较,由大到小取多个阈值进行测试。同时根据扫描线算法的特点,改变分治算法中点集分割的规则,进行了测试二测试。测试如表2 所示。

表2　合成算法在不同分割阈值下的处理时间

Tab.2　Performance of the compound algorithm under various split threshold

分割阈值		测试一	测试二	
分裂次数	子网个数	运行时间/s	运行时间/s	LOP 优化次数
0	0	26.92	29.84	823 281
1	2	25.53	26.71	796 245
2	4	24.31	24.12	778 932
3	8	24.02	21.84	765 782
4	16	25.98	20.69	760 842
5	32	26.85	19.61	769 127
6	64	28.12	22.98	791 665
7	128	29.16	26.29	830 354
8	256	30.13	29.79	892 709
9	512	31.64	33.56	961 452
10	1 024	33.15	37.97	1 050 017

在测试一中,对象为100 万个随机离散点,点重复概率在万分之三以内,点在 X 轴和 Y 轴方向上的分布是等概率的。分治算法采用比较常见方案,即是按点的个数划分。

当分裂次数为0 时,就是纯粹的扫描线算法;随着分裂次数的增加,逐渐趋向分治算法。测试一的结果表明,合成算法的时间效率要优于分治算法,与扫描线算法相当。由于两种算法的时间复杂度相同,且本机的硬件配置较好,为程序的执行提供了足够的内存空间,因此合成算法的效率提高不是很明显。

一般的分治算法是通过点数阈值来控制三角网的左右或上下分裂,对离散点不均匀的情况下,有很好的适用性,但是扫描线算法本身具有很好

的鲁棒性[12],即使在点不均匀的情况下仍然有很好的时间效率。考虑到扫描线算法在前线搜索方面比较耗时,可以考虑仅仅在 X 方向上进行左右分裂。测试二就是采用这种方案,对象仍为100万个随机离散点,点重复概率在万分之三以内,点在 X 轴和Y 轴方向上的分布采用一种特殊情况,即点在 X 轴方向上的分布可能性是在Y 轴方向上的10 倍。同为分裂次数为0 时,测试二的运行时间比测试一要长,这是因为分布形状的影响,导致前线搜索时间变长且不优的三角形增加。

测试二的结果表明:随着左右分裂的子网个数的增加,局部LOP 优化的总次数是先减少再增加;同时对每个子网的扫描线算法来说,X 方向的点数变少,使得其前线处理的搜索耗时缩短。合成算法的处理时间在这个动态变化过程中,先减少再增加,当分裂次数为5 时,达到最优,较纯扫描线算法(即分裂次数为0 时)提高了1/3。

4 结 语

分治法和扫描线法都是时间效率比较高的算法。基于这两种算法的合成算法很好地继承两者的优点,同时改善了扫描线算法的数据结构,不仅使得其结构更为严谨,也为扫描线算法构网的网间快速合并提供了很好的借鉴。该算法充分考虑平面上离散点分布的随意性,针对特殊的离散点分布状况(如多点共圆,多点共线)进行了充分的修正;其阈值可以动态调节,以适应不同的点集分布情况和运行环境,因此具有很好的稳定性和鲁棒性。通常在实际应用中,部分离散点之间常常存在某种约束关系,如DTM 分析中的地形结构线,如山脊线、山谷线等。如何将本方法应用于这类约束条件下Delaunay 三角网的生成,将是下一步研究工作的重点。

参考文献:

[1] WU Li-xin, SHI Wen-zhong. Principle and Algorithm of GIS [M]. Beijing: Science Press, 2003. 264-283. (吴立新,史文中. 地理信息系统原理与算法[M]. 北京:科学出版社, 2003, 264-283.)

[2] SU P, ROBERT L, SCOT DRYSDALE. A Comparison of Sequential Delaunay Triangulation Algorithms[J]. Computational Geometry, 1997, (7): 361-385.

[3] LI Zhi-lin, ZHU Qing. Digital Elevation Model[M]. Wuhan: Wuhan Uniiversity Press, 2001: 34-56. (李志林,朱 庆. 数字高程模型[M]. 武汉:武汉大学出版社, 2001: 34-56.)

[4] DWYER R A. A Faster Divide-and-Conquer Algorithm for Constructing Delaunay Triangulations [J]. Algorithmica, 1987, 2(2): 137-151.

[5] WU Xiao-bo, WANG Shi-xin, XIAO Chun-sheng. A New Study of Delaunnay Triangulation Creation[J]. Acta Geodaetica et Cartographica Sinica, 1999, 28(1): 28-35. (武晓波,王世新,肖春生. Delaunay 三角网的生成算法研究[J]. 测绘学报, 1999, 28(1): 28-35.)

[6] WU Yu-xiao, ZHANG Deng-rong. Improved Algorithm for Building Delaunay Triangulation[J]. Journal of Zhejiang University (Science Edition), 2004, 31(3): 343-348. (吴宇晓,张登荣. 生成 Delaunay 三角网的快速合成算法[J]. 浙江大学学报(理学版), 2004, 31(3): 343-348.)

[7] FORTUNE S. A Sweepline Algorithm for Voronoi Diagrams [J]. Algorithmica 2, 1987, 153-174.

[8] SHEWCHUK J R. Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator[A]. First Workshop on Applied Computational Geometry[C]. [s.l.]: [s.n.], 1996, 124-133.

[9] LEWIS B A, ROBINSON J S. Triangulation of Planar Regions with Applications[J]. The Computer Journal, 1978, 21(4): 324-332.

[10] GUIBAS L, STOLFI J. Primitives for the Manipulation of General Subdivisions and the Computation of Voronoi Diagrams[J]. ACM Trans Graphics, 1985, 4(2): 75-123.

[11] HU Jin-xing, MA Zhao-ting, WU Huan-ping, *et al*. Massive Data Delaunay Triangulation Based on Grid Partition Method [J]. Acta Geodaetica et Cartographica Sinica, 2004, 33(2): 163-167. (胡金星,马照亭,吴焕萍,等. 基于网格划分的海量数据 Delaunay 三角剖分[J]. 测绘学报, 2004, 33(2): 163-167.)

[12] CORMEN T, LEISERSON C, RIVEST R, STEIN C. Introduction to Algorithms [M]. Cambridge: The MIT Press, 2001. 50-75.

(责任编辑:张燕燕)