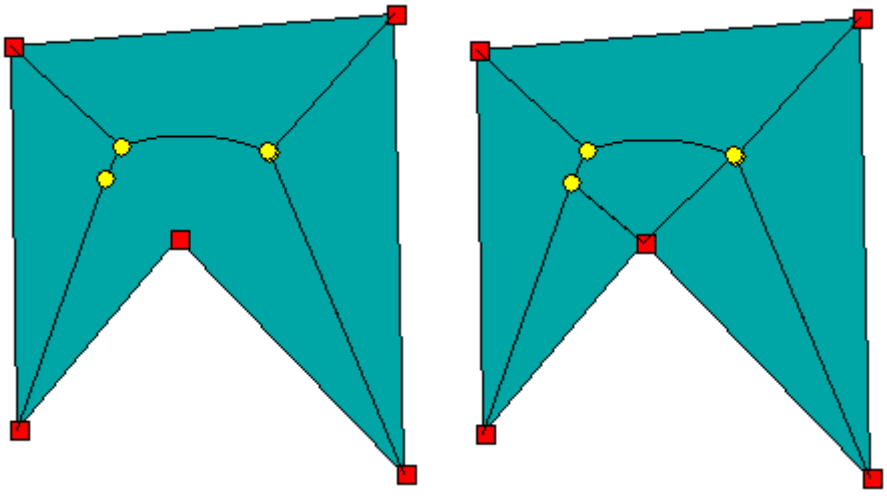
多边形中轴提取

胡博 036001 杨帆 036000

一、问题背景

Voronoi 图是计算几何中广泛研究的对象。最先被研究的是欧氏二维空间点集的 Voronoi 图。在二维空间的情况下，这些研究可以应用于分析结晶结构，宇宙分布，建筑物间的道路分布，运动规划，优化定位中。但是在这些应用中，应用对象不仅仅是单一的点，更为普遍的情况是直线，曲线，及直线、直线段、曲线、曲线段的连接等。其中一种重要的情况，就是在一个简单多边形内部，关于该多边形各顶点和边的 Voronoi 图。这种 Voronoi 图也被称为中轴或骨架。

严格的说，多边形的 Voronoi 图和多边形的中轴是不一样的，中轴应该是在 Voronoi 的基础上裁减得到，裁减的方法是去掉 Voronoi 图中由多边形凹顶点产生的两条 Voronoi 边，如下图所示：



多边形的中轴和 Voronoi 图

对于求简单多边形的中轴问题，1982 年有 D.T Lee[4]提出了复杂度为 O(nlogn)的分治算法，之后很多人在为得到 O(n)的算法而努力，其中的主导思想是将多边形的直方图划分，将问题转化成直方图划分，直方图的 Voronoi 求解及

合并这三个部分。在经历了几种接近 O(n)的随机构造算法后，由 Chin, Snoeyink, Wang[1]提出了 O(n)的确定算法。

实验的进行是以 Chin, Snoeyink, Wang 的算法为出发点，学习该算法和几个随机构造算法[2], [3]后，经过综合时间和实现难度的可行性分析后，最终实现的是 D.T Lee 的分治算法。

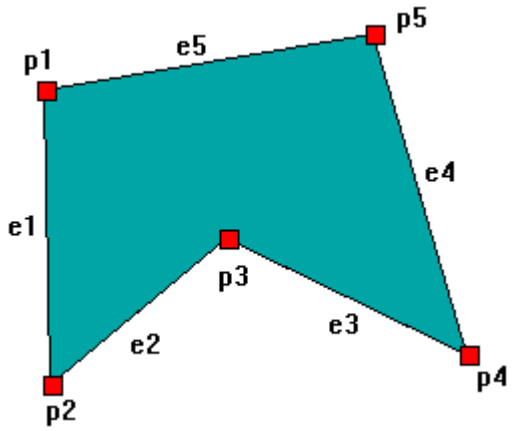
二、算法及原理

程序使用分治算法。首先将多边形的边界（点、边）分成若干个链，分别对其求 Voronoi 图，然后进行合并，直到整个多边形的边界被合成一条链时算法结束。

(1)边界拆分

将边界拆分成链的原则是，对于一个凹顶点（内角大于 180 度），它和它的左右相邻边在同一个链内，除此以外，一条边就构成一个链。初始状况下，链的 Voronoi 图可以直接得到。

以下图为例：



对该多边形分链结果如下：C1={e1}，C2={e2, p3, e3}，C3={e4}，C4={e5}。

(2)合并过程

每次合并两个相邻链 C1 和 C2 的 Voronoi 图，也就是求 Vor(C1)和 Vor(C2)的合并平分线。

假定 Vor(C1)和 Vor(C2)已知，C2 在 C1 的逆时针方向，C1 的最后一个元素和 C2 的第一个元素相邻，分别记为 ej 和 ej+1。

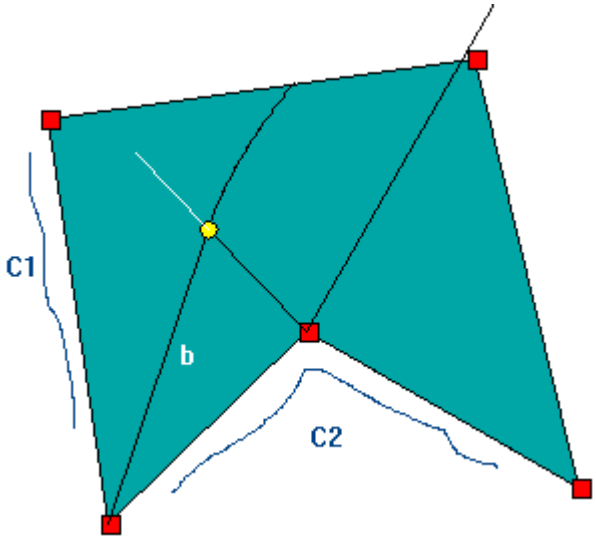
同时记 b(e1, e2)为 e1 和 e2 两个元素的平分线；定义 VR(e, C)为元素 e 相对链 C 的 Voronoi 区域，即该区域点到 e 的距离小于到 C 的距离。

显然 b(ej, ej+1)是合并平分线的一部分，从它出发进行扫描检查。

1. 逆时针检查 b(ej, ej+1)和 VR(ej, C1-ej)各边的交点，顺时针检查 b(ej, ej+1)和 VR(ej+1, C2-ej+1)各边的交点。如果 b(ej, ej+1)先与 b(ek, ej)(ek 属于 C1)相交，那合并平分线的下一段就是以 b(ek, ej+1)，交点是新的起点；如果 b(ej, ej+1)先和 b(ej+1, er)相交，那合并平分线的下一段是 b(ej, er)。
2. 不妨假定下一段平分线是 b(ek, ej+1)，则逆时针检查它与 VR(ek, C1-ek)各边的交点，顺时针检查与 VR(ej+1, C2-ej+1)的交点，重复这个过程直到满足终止条件。
3. 终止合并的条件如下：生成 C1 的第一个元素和 C2 最后一个元素的平分线，即已经遍历了两条链的所有元素，合并平分线终止；如果某次扫描检查中 b 和两个 VR 中的各边都没有交点，合并平分线结束。

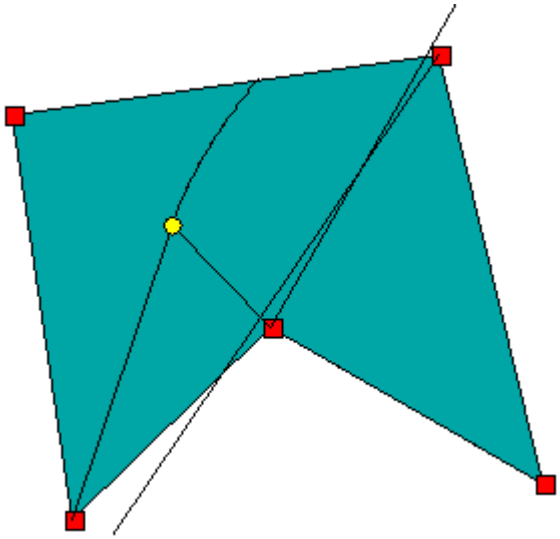
得到合并的平分线 b 后，舍弃 VR(C1)在 b 右侧的部分和 VR(C2)在 b 左侧的部分。合并之后 C1 和 C2 构成新的链。

以前面图的合并过程为例，先合并 C1={e1}和 C2={e2, p3, e3}，合并后结果如下：

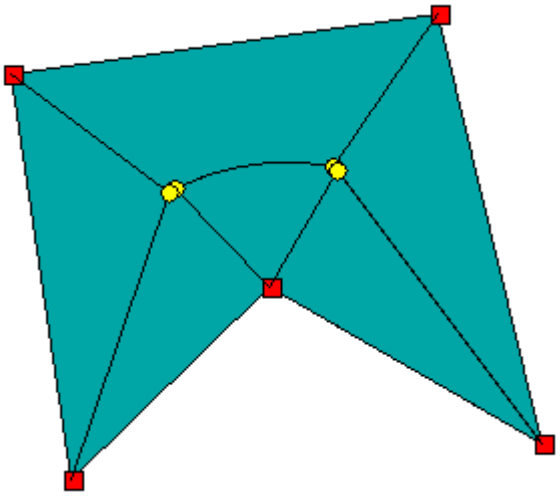


C1 的最后一个元素是 e1，C2 的第一个元素是 e2，b(e1, e2)为 e1 和 e2 的角平分线，与 VR(C2)有一个交点，舍弃 VR(C2)在 b(e1, e2)左侧的部分，合并平分线的下一段应该是 b(e1, p3)（图中黄色交点后的抛物线）；依次继续。

接着合并 C3 和 C4，得到：



将 C1, C2 的合并结果和 C3, C4 的合并结果合并，得到最终的 Voronoi 图：



(3)两元素间的 Voronoi 区域分界边求法

在合并过程中，关键的一步是求两个元素间的 Voronoi 区域的分界边。两元素有以下几种情况：

两条不相邻边：如果两条边平行，则取与它们等距的平行线作为分界边，否则延长边直到相交并作角平分线，作为分界边。

两条相邻边：显然，其分界边是它们构成的多边形内角的角平分线。可看作两条不相邻边的退化情况。

凹顶点与一条不相邻边：分界线到点及直线的距离相等，为一段抛物线。

凹顶点与一条相邻边：过顶点做边的垂线，即为分界线，可看作凹顶点与一条不相邻边分界线的退化情况。

三、数据结构

程序中主要构造了两种数据结构，SkeletonRegion 主要是对 Voronoi 区域的存储和操作，SkeletonEdge 则是对 Voronoi 边的存储和操作。

SkeletonEdge 记录了该 Voronoi 边两边的 Voronoi 区域，以及其构成区域的上一条边、下一条边等，同时提供相交，截取等基本操作。

SkeletonRegion 以逆时针方向记录了围成该区域的所有 Voronoi 边，以及产生该区域的多边形元素（边或凹顶点）。可以方便地在其维护的边表中插入、删除边界。

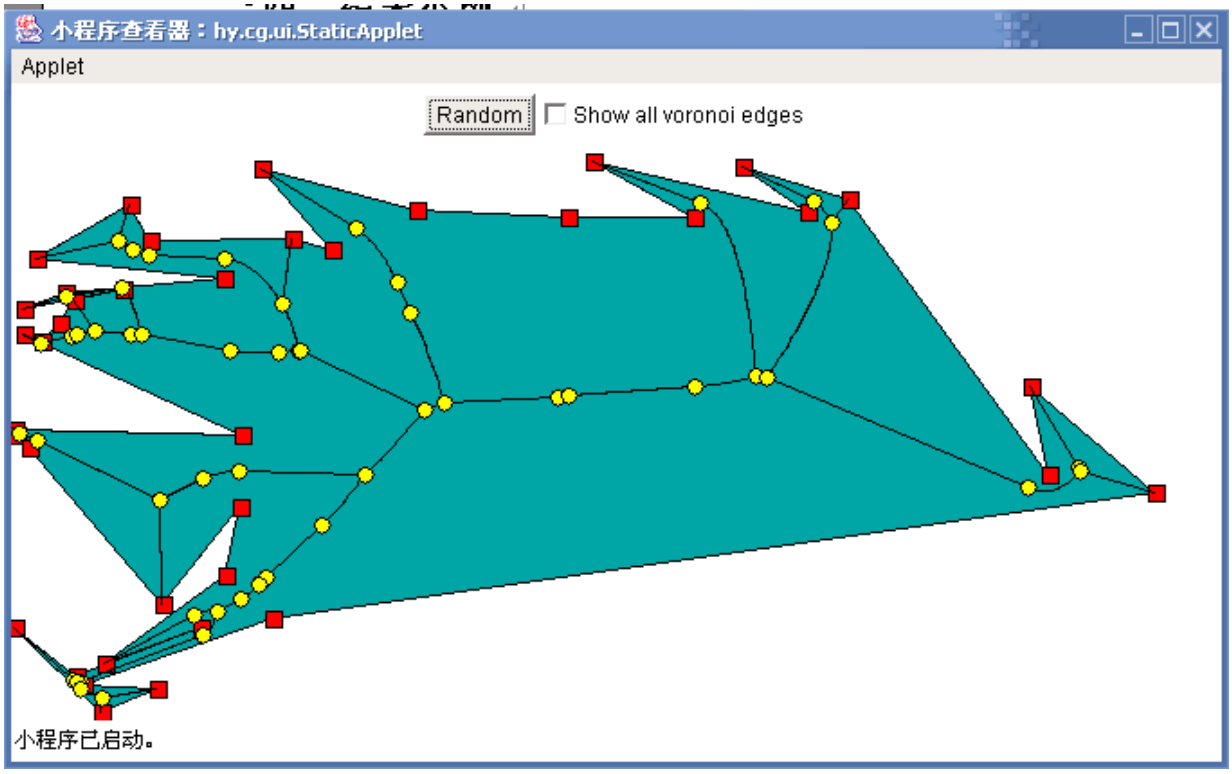
主程序过程中维护一个链表 chains，顺序记录了当前待合并的链，链表每个元素是如下形式：

chain	VR(chain)

每次选取两个相邻的链合并，删除原来的两个链，放回合并结果。到 chains 的长度为 1 时结束。

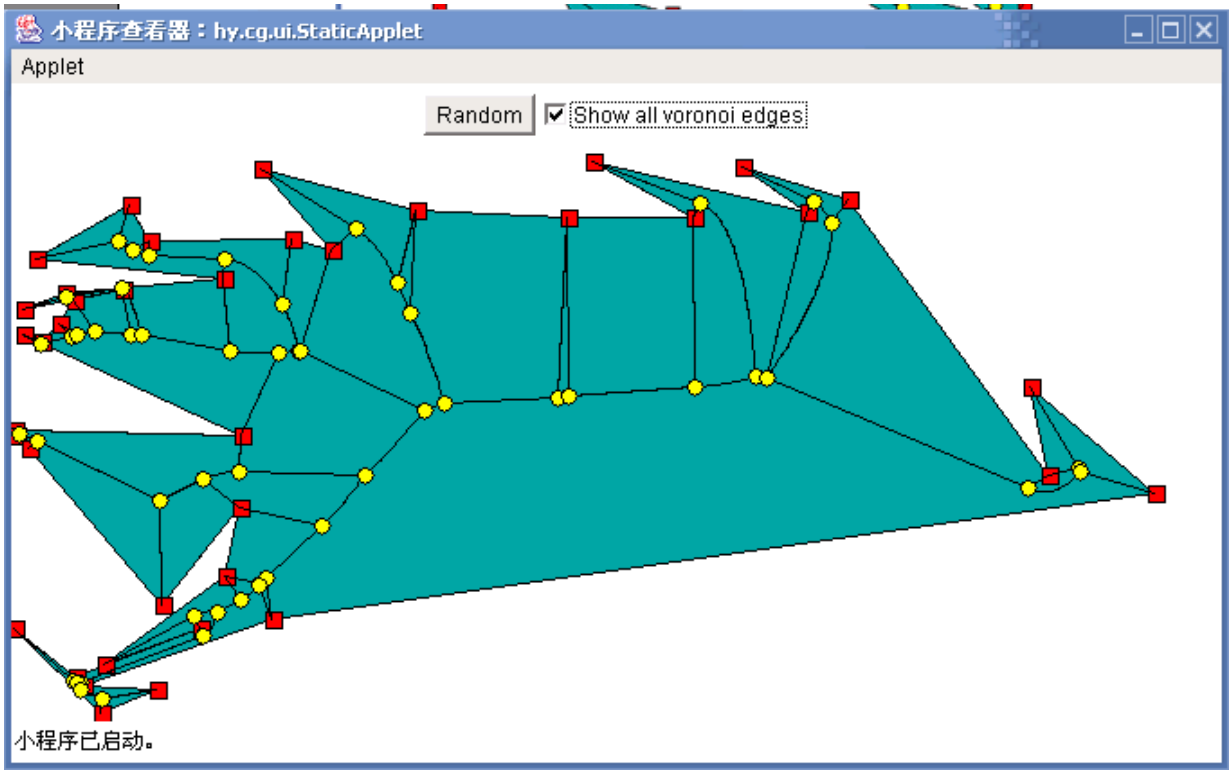
四、结果示例

随机生成一个简单多边形（使用的是非常简单的随机生成方法），显示它的中轴如下：



随机的多边形顶点数 39，凸顶点 20 个，凹顶点 19 个，计算用时 16ms。

对应的 Voronoi 图如下：



五、参考文献及参考代码

[1]. F. Chin, J. Snoeyink, and C. A. Wang. Finding the Medial Axis of a Simple Polygon in Linear Time.

[2]. R. Klein and A. Lingas. A linear-time randomized algorithm for the bounded Voronoi diagram of a simple polygon.

[3]. R. Klein and A. Lingas. Fast skeleton construction.

[4]. D. T. Lee. Medial axis transformation of a planar shape.