

Voronoi 区域面积的计算方法

A Computational Method for the Area of Voronoi Region

张有会 耿肇英

(河北师范大学计算机科学系 石家庄050016)

Abstract Voronoi diagram is an important branch of computation geometry. The area is an important property of a Voronoi diagram. In this paper, a computational method for the area of a Voronoi diagram in a rectangle is proposed. With this method, given the coordinates of the generating points and the corners of the rectangle, the area of any Voronoi diagram could be calculated.

Keywords Voronoi diagram, Computational geometry, Area, Computational method

1 引言

在计算几何中, Voronoi 图理论成功地解决了找最近点, 求最大空圆, 求 n 个点的凸包, 求最小树等问题. 此外, Voronoi 图还在生态研究、城市规划以及优化配置等许多领域有重要应用.

为简化书写, 本文在下面的叙述中, 将“Voronoi”简记为“V-”, 如“V-图”指的是“Voronoi 图”, “V-区域”指的是“Voronoi 区域”, 等等.

V-图是在平面上给定若干个点, 依据这些点所在的位置, 对平面进行的一种分割(如图1). 平面被分割成若干个区域, 称为 V-区域. V-区域的面积是 V-图的一个非常重要的物理属性, 该属性在 V-图的的实际应用中起着很大作用. 如 V-图用于城市规划及优化配置时, 就需要计算 V-区域的面积^[1,2]. 本文给出了矩形范围内 V-图的 V-区域面积的计算方法. 利用该方法, 只要给出矩形顶点和 V-图母点的坐标, 便可计算出各 V-区域的面积.

2 V-图的定义

V-图的数学定义是: 设 $p_i (i=1, 2, \dots, n)$ 为二维欧氏空间(平面)上的 n 个互不相同的点, 称

$$V_n(p_i) = \bigcap_{j=1}^n \{p | d(p, p_i) < d(p, p_j)\} \quad (1)$$

为点 p_i 的 V-区域, 其中 $d(p, p_i)$ 为 p 和 p_i 间的 Euclid 距离. 如果 $i \neq j$ 时, $V_n(p_i) \cap V_n(p_j)$ 非空且非单点集, 则称 $V_n(p_i) \cap V_n(p_j)$ 为 p_i 和 p_j 间的 V-边, 其中 $V_n(p_i)$ 是 $V_n(p_i)$ 的闭包. 两条以上 V-边的交点, 称为 V-点, 也就是 V-区域的顶点. 将 $V_n(p_i) (i=1, 2, \dots, n)$ 及其边界称为以 $p_i (i=1, 2, \dots, n)$ 为母点(或生成元)的 V-

图, 通常简称为 V-图, 记作 V_n (见图1).

现在介绍 V-图与本文内容有关的几个简单性质如下:

- (1) p_i 和 p_j 间的 V-边是 p_i 和 p_j 间的垂直平分线的一部分;
- (2) 由(1)式可知, V-区域是 $n-1$ 个半平面的交集, 是凸集;
- (3) 母点与 V-区域一一对应;
- (4) 位于全体母点的最小凸包边界上的母点, 其所在 V-区域的面积为无穷大.

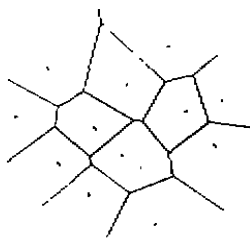


图1 Voronoi 图

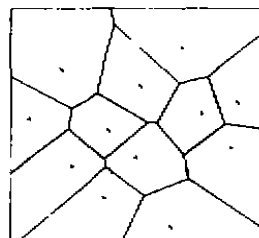


图2

为突出实用性, 将 V-图限定在一矩形区域 D 内(图2), 此时, 所有 V-区域均为由闭合折线段围成的有限凸集, 其边界是凸多边形.

3 V-区域面积的计算

计算各 V-区域面积的基本思想是, 对给定的 n 个母点, 先用逐点添加的方法求出各 V-区域的顶点坐标, 将其存放在各自的循环链表中, 然后再根据保存在循环链表中的顶点坐标, 计算 V-区域的面积.

3.1 求 V-区域顶点坐标的准备工作

在给出求 V-区域顶点坐标的一般方法之前,将求解过程中所用的一些基本方法(过程)介绍如下:

3.1.1 两点位于一直线异侧的判断方法 设直线 L 的方程为: $Ax + By = C$, 两点 p_1, p_2 的坐标分别为 (x_1, y_1) 和 (x_2, y_2) 。如果

$$\begin{aligned} \text{sign}(Ax_1 + By_1 - C) &\neq 0, \text{sign}(Ax_2 + By_2 - C) \neq 0 \\ \text{且 } \text{sign}(Ax_1 + By_1 - C) &\neq \text{sign}(Ax_2 + By_2 - C) \end{aligned}$$

则 p_1, p_2 位于 L 的异侧。

$$\text{其中 } \text{sign}(\cdot) \text{ 为符号函数: } \text{sign}(x) = \begin{cases} 1 & \cdots \cdots x > 0 \\ 0 & \cdots \cdots x = 0 \\ -1 & \cdots \cdots x < 0 \end{cases}$$

3.1.2 直线与线段是否相交的判断方法 如果线段 p_1p_2 的两端点 p_1, p_2 位于直线 L 的异侧,或位于直线 L 上,则直线 L 与线段 p_1p_2 相交,否则不相交。

3.1.3 求直线与线段的交点坐标的方法 为方便计算,设直线 L 的单位方向向量为 (α, β) 且过点 (x_0, y_0) , 则 L 可表为:

$$\begin{pmatrix} x \\ y \end{pmatrix} = t \begin{pmatrix} \alpha \\ \beta \end{pmatrix} + \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}, (-\infty < t < +\infty)$$

设线段 p_1p_2 的两端点 p_1, p_2 的坐标分别为 (x_1, y_1) 和 (x_2, y_2) 。如果 L 与线段 p_1p_2 相交,则交点坐标 (x, y) 由下式给出:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} + \lambda \begin{pmatrix} x_2 - x_1 \\ y_2 - y_1 \end{pmatrix}$$

$$\text{其中 } \lambda = \frac{\alpha(y_0 - y_1) - \beta(x_0 - x_1)}{\alpha(y_2 - y_1) - \beta(x_2 - x_1)}.$$

3.1.4 与一固定 V-区域相邻的全部 V-区域的查找方法 设 p_0 为 V-图 V_m 中的一个母点, $V_m(p_0)$ 的顶点为 $a_0a_1a_2 \cdots a_t$ (其中 $a_t = a_0$)。求 p_0 关于 a_i, a_{i+1} 所在直线的对称点 q_i ($i = 0, 1, 2, \cdots, t-1$)。则 q_i ($i = 0, 1, 2, \cdots, t-1$) 均为 V_m 中的母点, 线段 a_ia_{i+1} 为 p_0 和 q_i 之间的 V-边, q_i ($i = 0, 1, 2, \cdots, t-1$) 所在的 V-区域即为全部与 $V_m(p_0)$ 相邻的 V-区域。

3.1.5 查找一点所在 V-区域的方法 设由母点 p_i ($i = 1, 2, \cdots, m$) 所确定的 V-图为 V_m , 现对平面上的点 p , 求 p 所在的 V-区域。

(1) 在母点 p_i ($i = 1, 2, \cdots, m$) 中任选一母点, 不妨设为 p_1 , $k := 1$;

(2) 在 V_m 中对所有与 $V_m(p_k)$ 相邻的 V-区域 $V_m(p_i)$, 找满足 $d(p, p_i) < d(p, p_k)$ 的点 p_i ;

(3) 如果这样的 p_i 不存在, 则 p_k 即为所求, 结束; 否则, 令 $k := i$, 转(2)。

3.1.6 求一 V-区域的多边形边界中, 与一给定直线相交的线段 设 V-区域 v 的顶点为 $a_0a_1a_2 \cdots a_t$ (其中 $a_t = a_0$)。直线 L 与 v 有两个交点, 且 L 不过 v 的顶点。任选一顶点, 不妨设为 a_0 , 用第 3.1.1 节的方法将所

有顶点分为与 a_0 位于 L 的同侧和异侧两部分。由于 v 是凸多边形, 可设 $a_0a_1 \cdots a_{j_0}$ 位于 L 的一侧, 而其余顶点位于 L 的另一侧, 故与 L 相交的两个线段分别为线段 $a_{j_0-1}a_{j_0}$ 和线段 $a_{j_0}a_{j_0+1}$ 。

3.2 创建和更新循环链表的过程

设 V_m 是由 m 个母点 p_i ($i = 1, 2, \cdots, m$) 所确定的 V-图, 设 L_i 是循环链表, 用来存放母点 p_i 所在 V-区域的, 相对于 p_i 逆时针排列的顶点坐标序列, ($i = 1, 2, \cdots, m$)。现对 V_m 添加母点 p_{m+1} , 构造 V_{m+1} , 为此需更新部分相关的循环链表并创建新循环链表 L_{m+1} , 其方法如下。

在 V-图 V_m 中, 按第 3.1.5 节中方法找 p_{m+1} 所在的区域, 设为 $V_m(p_i)$; 按第 3.1.6 节中方法求两母点 p_{m+1}, p_i 的垂直平分线与 $V_m(p_i)$ 的边界的交点, 设为 q_1, q_2 ; 以下分三种情形讨论:

① q_1, q_2 均位于 D 的边界上。当 q_1, q_2 相对于 p_{m+1} 为逆时针顺序时(图 3), 设 q_1, q_2 分别位于 L_i 中的线段 $a_{j_0}a_{j_0+1}$ 和 $a_{j_1}a_{j_1+1}$ 上(当 q_1 或 q_2 恰与 L_i 中的某一点 a_{j_0} 重合时, 则将点 a_{j_0} 作为 L_i 中的一线段 $a_{j_0}a_{j_0+1}$ ($a_{j_0+1} = a_{j_0}$) 处理即可), 更新 L_i 为 $(q_2q_1a_{j_0+1} \cdots a_{j_1}q_2)$, 创建 L_{m+1} 为 $(q_1q_2a_{j_1+1} \cdots a_{j_0}q_1)$; 当 q_1, q_2 相对于 p_{m+1} 为顺时针顺序时, 类似处理。

② q_1, q_2 中有且仅有一点位于 D 的边界上。如果 q_1, q_2 中有且仅有一点, 不妨设 q_1 位于 D 的边界上, 更新和创建过程如下(图 4):

(1) $k := 2$;

(2) 设 q_k 与 L_i 的线段 $a_{j_0}a_{j_0+1}$ 相交;

(3) 求与 p_i 关于线段 $a_{j_0}a_{j_0+1}$ 相邻的母点, 设为 p_{j_0} , 更新 L_i (方法同①); 求 p_{m+1}, p_{j_0} 的垂直平分线与 L_{i_0} 相交于 q_k 和 q_{k+1} 两点;

(4) 如果 q_{k+1} 位于 D 的边界上, 转(5), 否则, $k := k+1, j := j_0$, 转(2);

(5) 创建 L_{m+1} : 如果顶点 q_1, q_2 关于 p_{m+1} 为逆时针排列, 则创建 L_{m+1} 为 $(q_1q_2 \cdots q_k a_{j_0} \cdots a_{j_1} q_1)$, 否则, 创建 L_{m+1} 为 $(q_1 a_{j_1} \cdots a_{j_0} q_1, q_{k+1} \cdots q_2 q_1)$, 其中 q_1 是最后一次求得的位于 D 的边界上的点, $a_{j_0} \cdots a_{j_1}$ 是 D 边界上位于从 q_k 到 q_1 之间的点。

③ 如果 q_1 和 q_2 均不在 D 的边界上(图 5)

(1) $k := 2$

(2) 设 q_k 与 L_i 的线段 $a_{j_0}a_{j_0+1}$ 相交;

(3) 求与 p_i 关于线段 $a_{j_0}a_{j_0+1}$ 相邻的母点, 设为 p_{j_0} , 更新 L_i (方法同①); 求 p_{m+1}, p_{j_0} 的垂直平分线与 L_{i_0} 相交于 q_k 和 q_{k+1} 两点; 如果 q_{k+1} 位于 D 的边界上, 则令 $q_1 := q_{k+1}, q_2 := q_k, j := j_0$, 转为情形②处理; 否则转(4);

(4) 如果 $q_{k+1} = q_1$, 转(5), 否则 $k := k+1, j := j_0$,

