

# 二维不规则图形和 印制板电路区域运算的算法研究\*

周明天 徐波

(电子科技大学计算机科学与工程学院, 成都 610054)

**摘要** 区域运算是轮廓求解算法集中的核心算法. 本文介绍适用于复杂 PCB 电路和二维不规则图案的一种区域运算算法, 阐述了它的数学模型、算法描述、正确性证明和复杂性分析. 与典型的算法比较, 它有明显的优越性.

**关键词** 计算机辅助设计/制造, 区域运算, 计算机图形学, 印制电路板, 铭牌.

区域运算算法是轮廓求解算法集中的核心算法, 它应对 PCB 电路轮廓求解具有高效性并且适用于美工装饰中的各种不规则图案. 目前, 国内外<sup>[1,2]</sup>尚无能够满足这些要求的算法可以直接采用. 本文讨论我们提出的适用于复杂 PCB 电路和不规则图案轮廓求解的区域运算算法, 给出它的数学模型、算法描述、正确性证明和复杂性分析.

## 1 算法描述

### 1.1 区域运算的分解

**定义 1.1.** 二维欧氏空间  $E^2$  中的区域  $D$  定义为以下运算的结果:

$$D = O - H_1 - H_2 \cdots H_n$$

其中  $O$  为连通闭集,  $H_1, H_2, \dots, H_n$  是两两隔离的非空连通开集, 且  $H_i \subset O, i = 1, 2, \dots, n$ . 区域  $D$  记为  $D(O, H_1, H_2, \dots, H_n)$ ,  $O$  称为小岛,  $H_i$  称为孔洞(注意区域可以没有孔洞).

其余有关定义详见文献[1].

轮廓求解中的区域运算是建立在边界为简单多边形的区域集合上的封闭运算, 其结果仍为边界是简单多边形的区域. 设有区域  $A(O_A, H_{A1}, H_{A2}, \dots, H_{Am}), B(O_B, H_{B1}, H_{B2}, \dots, H_{Bn})$ , 根据区域定义,  $A = O_A - (H_{A1} \cup H_{A2} \cdots \cup H_{Am}), B = O_B - (H_{B1} \cup H_{B2} \cdots \cup H_{Bn})$ .  $A$  与  $B$  直接求并显然是很困难的, 应想办法把  $A$  与  $B$  求并分解为小岛与孔洞之间的布尔运算.

$$\begin{aligned} A \cup B &= (O_A - (H_{A1} \cup H_{A2} \cdots \cup H_{Am})) \cup (O_B - (H_{B1} \cup H_{B2} \cdots \cup H_{Bn})) \\ &= O_A \cup O_B - \cup (H_{Ai} - O_B) - \cup (H_{Bj} - O_A) - \cup \cup (H_{Ai} \cap H_{Bj}) \end{aligned}$$

\* 本文 1994-01-15 收到, 1994-04-18 定稿

本文的工作得到电子科学基金(SFED)的资助. 作者周明天, 1939年生, 教授, 主要研究领域为开放分布系统环境, 计算机应用, 计算机网络. 徐波, 1969年生, 博士生, 主要研究领域为计算机应用, 计算机网络, 开放分布处理环境.

本文通讯联系人: 周明天, 成都 610054, 电子科技大学计算机科学与工程学院

可以证明上述多项式中各项正是  $A \cup B$  的小岛和孔洞. 这样区域间的求并运算分解为多边形的并、交、差运算, 即小岛与小岛求并, 孔洞与孔洞求交, 孔洞与小岛求差.

### 1.2 多边形的并、交、差运算

为便于运算, 规定小岛和孔洞的边界取向符合右手规则, 即沿边界前进时区域恒在边界的左侧. 按这一规定, 小岛多边形取向为逆时针方向, 孔洞多边形取向为顺时针方向.

为求得运算后的新多边形, 首先要求出两多边形的所有交点, 同时还需要决定多边形经过交点时是进入还是离开另一个多边形的区域. 以小岛与小岛求并为例,  $A$  的有向边到交点  $p$  ( $A$  的入点) 时, 就不能进入  $B$ , 而是沿  $B$  的边界前进, 到达下个交点  $q$  ( $B$  的入点) 时又要改变走向, 沿  $A$  前进, 如图 1 所示.  $S(p, q)$  上除  $p$  和  $q$  的任一点  $T$  称为  $p$  的前向邻点. 显然, 若  $T$  在  $B$  以外, 则  $p$  为出点, 若  $T$  在  $B$  以内或  $B$  边界上, 则  $p$  为入点.

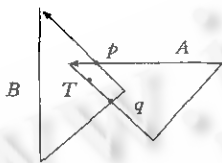


图1 出点、入点、前向邻点

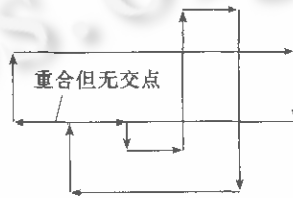


图2 孔洞边界重合时可能跳段

当多边形边界有重合时情形就比较复杂了. 重合边界可能出现在结果多边形中, 也可能不出现在结果多边形中, 边界重合也并不意味着两多边形有交点. 比如孔洞与孔洞求交 (如图 2 所示), 由于孔洞是开集, 虽然  $A, B$  有边界重合, 但实际上在重合段无交点, 遇到这种情况应该跳过重合段直到下一个交点再继续出入点判定. 如果两多边形在重合的一段处走向相同, 则称重合段起点为同向重合点, 否则称重合段对于两多边形的起点为反向重合点. 表 1 列出了三类多边形运算对于出入点、重合点的换向对策.

表 1 多边形布尔运算基于交点分类换向对策

	入点	出点	同向重合点	反向重合点
小岛 $\cup$ 小岛	换向	不换向	不换向	换向
孔洞 $\cap$ 孔洞	不换向	换向	换向	跳段
孔洞—小岛 ( $I$ )	不换向	换向	不换向	跳段
孔洞—小岛 ( $H$ )	换向	不换向	换向	跳段

( $I$ ) 表示当前多边形是小岛, ( $H$ ) 表示当前多边形是孔洞

### 1.3 多边形集合运算算法描述

设两多边形交于  $k$  个交点 ( $k \geq 2$ ), 则这  $k$  个交点将两多边形各分为  $k$  段, 每一段称为一个片段. 根据片段相对于多边形的位置关系, 我们将片段分为内部片段、外部片段、同向重合片段和反向重合片段, 并为每一类片段规定一个权值, 内部片段和反向重合片段权值为 0, 外部片段和重合同向片段权值为 1. 为便于算法描述和正确性证明, 我们还定义了片段图.

定义 1.2. 片段图  $FP$  是一有序偶  $\langle V, E \rangle$  即  $FP = \langle V, E \rangle$ . 其中  $V = \{c_1, c_2, \dots, c_n\}$  是两多边形的交点集合,  $E = \{ \langle a, b, d \rangle \mid \langle a, b, d \rangle \in V \times V \times \{0, 1\} \}$ , 且  $a, b$  分别为  $A$  和  $B$  中某一片段的起点和终点,  $d$  为该片段的权值.  $V$  中每一元素称为  $FP$  图的顶点,  $E$  中每一元素称为

$FP$  图的弧. 这里定义权值为 1 的弧称为有效弧, 权值为 0 的弧称为无效弧.

根据  $FP$  图定义,  $A, B$  中每一外部片段、内部片段以及反向重合边在  $FP$  图中有唯一一条弧与之对应, 对于一对同向重合片段, 权值都为 1, 但它们表示的是同一段几何曲线, 因此在  $FP$  图中也有唯一弧与之对应.

多边形运算的实质就是按特定规则遍历  $FP$  图, 并用所经历的弧重构结果多边形. 在遍历之前首先要选择  $FP$  图中合适的顶点作为起点, 以使遍历能够起动和终止. 比如多边形求并时必须选择引出弧权值为 1 的顶点作为遍历起点. 表 2 列出了各种多边形运算时, 起始顶点的选择策略.

表 2  $FP$  图遍历的起始顶点选择策略

多边形运算	起始顶点引出弧权值
小岛 $\cup$ 小岛	1
孔洞 $\cap$ 孔洞	0
孔洞—小岛	0 或 1

算法 1.1. 多边形  $A, B$  求交、并、差:

(1) 准备. 求出  $A$  和  $B$  的所有交点, 并分别按小岛或孔洞的顺序排好.  $A$  加上交点扩充成为  $A'$ , 同样,  $B$  加上交点扩充为  $B'$ . 如果没有交点, 则  $A' = A, B' = B$ , 然后转步骤(7).

(2) 确定  $A', B'$  中交点的出入点特征, 并置“未经历”标志.

(3) 根据  $FP$  图的起始顶点选择策略, 取  $A'$  的一个未经历交点作为起始点, 开始建立新多边形.

(4) 判断交点类型, 根据表 1 中换向对策实施换向, 并对  $A', B'$  中此交点置“已经历”标志.

(5) 沿正在进行的多边形, 取下一个点(顶点或交点). 如果是顶点, 进行本步骤; 如果是交点, 则判断是否回到首交点. 如果回到首交点, 转步骤(6); 否则转步骤(4).

(6) 判断还有没有未经历的交点, 若有, 则转步骤(3).

(7) 判断所得新多边形的包含关系. 若没有包含关系, 则结束. 若有, 则最外面多边形为小岛, 把在小岛内的同方向多边形删去, 异方向多边形即为孔洞.

## 2 算法正确性证明

两多边形的交点将这两个多边形各自分为若干片段, 基于交点分类算法的实质是对这些片段进行分类, 将其分为出现在结果多边形的和不出现在结果多边形的. 下面将证明算法 1.1 进行的片段分类对于简单多边形运算是正确的, 从而证明算法正确性. 我们着重讨论小岛与小岛求并的情况. 对于其它两类多边形运算证明是类似的.

**定理 2.1.** 给定两小岛多边形  $A, B$  以及片段  $e_A(a, b)$ , 当且仅当  $e_A(a, b)$  是外部片段或同向重合片段时,  $e_A(a, b)$  是并集多边形  $A \cup B$  中的边.

为证明定理 2.1, 先引入两个引理.

**引理 2.1.** 同向重合片段仍是并集多边形的边界.

引理 2.2. 反向重合片段不可能是并集多边形的边界.

证明(定理 2.1):先证充分性. 设  $e_A(a,b)$  是  $A$  中外部片段, 取  $e_A(a,b)$  上除端点外任意一点  $x$ , 由于  $x \in b(A)$ , 因此  $x$  的任一邻域  $N(x,r)$  既包含  $A$  中的点又包含  $\sim A$  中点. 又  $x \in o(B)$ , 因此存在  $x$  的邻域  $N(x,r_0) \subset \sim B$ , 于是有  $N(x,r) \cap N(x,r_0) \cap \sim A \neq \emptyset$ . 即是说, 存在点  $y \in N(x,r)$  使得  $y \in N(x,r_0)$  且  $y \in \sim A$ , 但  $N(x,r_0) \subset \sim B$ , 故  $y \in \sim B \cap \sim A$ , 即  $y \in \sim(A \cup B)$ . 这样  $N(x,r)$  中既包含  $A \cup B$  中的点又包含  $\sim(A \cup B)$  中的点, 所以  $x \in b(A \cup B)$ . 根据边界的连续性,  $a, b$  也将是  $A \cup B$  的边界点. 于是  $e_A(a,b)$  是并集多边形的边界. 根据引理 2.1, 同向重合片段是并集多边形中的边. 充分性得证.

再证必要性. 假设  $A$  中有内部片段  $e_A(a,b)$  成为  $A \cup B$  的边界. 对  $e_A(a,b)$  上除  $a, b$  外任一点  $x, x \in i(B)$ , 因此存在邻域  $N(x,r) \subset B$ , 当然有  $N(x,r) \subset A \cup B$ , 这样  $x \in i(A \cup B)$ , 这与  $x \in b(A \cup B)$  矛盾. 根据引理 2.2, 反向重合片段不可能是并集多边形的边界. 必要性得证. □

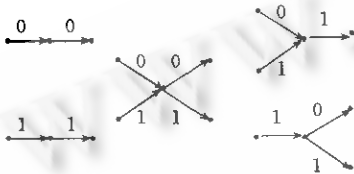


图3 简单多边形相交FP图的顶点弧模式

算法 1:1 的实质是对  $FP$  图的遍历. 下面证明算法 1.1 对  $FP$  图的遍历是可终止的、正确的和完备的. 认真研究两小岛相交所得  $FP$  图的特征可知, 对小岛相交,  $FP$  图中任意顶点只可能有以下弧模式存在, 如图 3.

引理 2.3. 设小岛  $A$  与小岛  $B$  相交并有片段图

$FP$ , 从任一有效引出弧顶点出发, 沿有效弧遍历该  $FP$  图, 则经过有限次遍历, 可以将且仅将  $FP$  所描述的全部外部片段和同向重合片段遍历完, 并且每个片段只经历一次.

定理 2.2. 算法 1.1 将所有外部片段和同向重合片段归入并集多边形, 所有内部片段和反向重合片段均未归入并集多边形.

再考虑算法 1.1 各步骤与引理 2.3 中遍历算法的对应关系. 换向对策决定了算法 1.1 总是遍历有效弧, 因此算法 1.1 将所有外部片段和同向重合片段归入并集多边形, 而所有内部片段和反向重合片段均未归入并集多边形. □

结合定理 2.1, 得出算法 1.1 生成的新多边形正是并集多边形.

3 算法复杂性分析

首先分析算法 1.1 最坏情况时间复杂性. 假设要计算多边形  $A$  和  $B$  的并集,  $n_A, n_B$  表示  $A$  和  $B$  的顶点数(边数),  $s$  表示  $A$  和  $B$  的交点数. 注意  $s$  最大可以是  $n_A \cdot n_B$ . 算法 1.1 各步骤的时间复杂性如下:

(1) 求  $A$  与  $B$  各边的交点. 该步骤时间开销为  $O(n_A \cdot n_B)$ .

(2) 对交点分类. 对每一交点取前向邻点时间开销为(1). 采用环绕法判断  $A$  中前向邻点对  $B$  的位置关系需要  $O(n_B)$  步, 判断  $B$  中一个前向邻点对  $A$  的位置关系需要  $O(n_A)$  步, 因此该步骤最坏情况时间开销为  $O(s + s \cdot \max\{n_A, n_B\})$ .

(3) 遍历  $FP$  图, 由于  $FP$  顶点数为  $s$ , 该步时间开销为  $O(s)$ .

将以上结果相加, 可知算法 1.1 最坏情况时间复杂性为:

$$O(n_A \cdot n_B + s + s \cdot \max\{n_A, n_B\}) \leq O(n_A \cdot n_B \cdot s)$$

算法 1.1 占有的空间由  $A$ 、 $B$  及  $A \cup B$  三个多边形链表决定,  $A$  多边形链表长度为  $O(n_A + s)$ ,  $B$  多边形链表长度为  $O(n_B + s)$ ,  $A \cup B$  多边形链表长度为  $O(n_A + n_B + s)$ , 所以该算法的空间复杂性为  $O(n_A + n_B + s)$ .

#### 4 同类算法的比较

基于顶点分类的多边形集合运算 Margalit 算法<sup>[1]</sup>是目前比较通用而有效的算法. 对 PCB 电路轮廓求解, 上述我们提出的基于交点分类的算法效率比 Margalit 算法高得多.

例如, 设基本图元顶点数总为  $a_0$ , 基本图元与连通域的交点数总为  $n_c$ , 总共有  $n$  个图元相交. 第  $i$  次求并后连通域顶点数为  $a_i = a_0 + i \cdot (a_0 + n_c)$ . 现在考虑第  $i+1$  次求并的时间复杂性. 对顶点分类算法, 总有

$$S_{i+1} = a_i \cdot a_0 + a_0 \cdot a_i$$

对交点分类算法有  $S'_{i+1} = (n_c/2) \cdot a_0 + (n_c/2) \cdot a_i + \lambda \cdot n_c$

其中  $\lambda \cdot n_c$  为求前向邻点的时间开销,  $\lambda$  是求单次前向邻点相对于进行单次内外判别的开销系数.

因为 取曲线段中点为前向邻点

所以  $\lambda \ll 1$

又因为  $a_0 \geq 2, a_1 \geq 2$

所以  $\frac{\lambda \cdot n_c}{(n_c/2) \cdot a_0 + (n_c/2) \cdot a_i} \approx 0$

所以  $S'_{i+1} \approx (n_c/2) \cdot a_0 + (n_c/2) \cdot a_i$

于是完成  $n$  个图元求并, 对顶点分类算法需要作内外判别次数

$$U = S_1 + S_2 + \dots + S_{n-1} = 2(n-1)a_0^2 + (n-2)(n-1)(a_0 + n_c)a_0$$

对交点分类算法只需

$$U' = S'_1 + S'_2 + \dots + S'_{n-1} = (n-1)n_c a_0 + (n-2)(n-1)(a_0 + n_c)n_c/4$$

设  $a_0 = n_c = 4$ , 表 3 列出图元数为 3 到 20 间两类算法的复杂性比较.

表 3 顶点分类与交点分类复杂性比较

相交图元数	内外判别次数		相交图元数	内外判别次数	
	顶点分类	交点分类		顶点分类	交点分类
3	128	48	12	3872	1056
4	288	96	13	4608	1248
5	512	160	14	5408	1456
6	800	240	15	6272	1680
7	1152	336	16	7200	1820
8	1568	448	17	8192	2176
9	2048	576	18	9248	2448
10	2592	720	19	10368	2736
11	3200	880	20	11552	3040

#### 5 结 论

本文提出了一种适用于二维复杂 PCB 电路和不规则图案的区域运算算法, 与我们在

轮廓求解所提出的算法一起构成了一组算法,包括曲线求交、包含判定、多边形运算和连通分割等.该算法和我们提出的环绕法已在一台基于 PC386 的小型(20000 个图元,500mm×500mm)多功能(集印制电路板、雕刻、光绘、钻孔和铭牌商标图案制作于一体)CAD/CAM 系统上实现,软件代码 C++ 源程序 15000 余行,半年多的实际运作表明,它使用方便,人机界面友好,其正确性、高效性和鲁棒性(Robustness)优于颇为知名的 Margalit 算法.

### 参考文献

- 1 Avraham Margalit. An algorithm for computing the union, intersection or difference of two polygons. *Computer & Graphics*, 1989, 13(2), 167-183.
- 2 应道宁. 多边形裁剪及布尔运算的新方法. *计算机学报*, 1987, 10(7), 444-449.

## AN ALGORITHM FOR REGION OPERATION ON 2D COMPLICATED PCB CIRCUIT AND IRREGULAR PATTERN

Zhou Mingtian Xu Bo

(College of Computer Science and Engineering, University of Electronic Science and Technology, Chengdu 610054)

**Abstract** Region operation is key part of algorithm suite for outline resolution. An algorithm the authors proposed for region operation on 2D complicated PCB circuit and irregular pattern is presented. Its mathematic model, algorithm description and implementation are described as well as correctness proof and complexity analysis of the algorithm. The gain shows sufficient in comparison with the typical algorithm in the case of PCB circuit.

**Key words** CAD/CAM, region operation, computer graphics, printed circuit board, inscription board.