

[文章编号] 1003-4684(2022)05-0033-05

# 基于减治的点与凸多边形位置关系判定算法

张 浩<sup>1</sup>, 沈 华<sup>1,2</sup>, 谌 刚<sup>1</sup>

(1 湖北工业大学计算机学院, 湖北 武汉 430068; 2 桂林电子科技大学广西可信软件重点实验室, 广西 桂林 541004)

[摘 要] 人们利用移动设备享用的很多位置服务涉及点与凸多边形位置判定问题。移动设备资源受限的客观条件使得设计轻量级算法解决问题成为当务之急。寻找一种轻量级的判定算法是必要的,减少与点进行操作的边的条数成为一种可行思路。因此,基于减治思想提出了一种轻量级点与凸多边形位置关系判定算法。算法包括三个模块:区域划分、点的区域判断和点与凸多边形的位置关系判断。算法通过将点与凸多边形的位置关系判断转化为点与凸多边形的一部分区域位置关系判断,减少了时间开销。通过将凸多边形的顶点编序并划分为多个子区域作为算法的预处理部分,算法的时间开销可以达到  $O(\log \sqrt{n})$ 。本算法可以适用在移动设备资源受限的场景下快速进行点与凸多边形的位置关系判断。

[关键词] 减治法; 凸多边形; 区域判断; 位置关系判定算法

[中图分类号] TP3 [文献标识码] A

随着位置感知移动终端的普及,基于位置的服务(Location-Based Service, LBS)给人们的生活带来了极大的便利<sup>[1]</sup>。近年来,几何范围查询在 LBS 应用中被广泛应用。判断点与凸多边形的位置关系是几何范围查询的一种基本操作。例如 LBS 的典型应用——近邻检测服务,该服务允许用户在地图上选择一个特定的几何范围,然后询问他的朋友是否在这个范围内。通常将用户选定的几何范围抽象为凸多边形,将用户朋友的位置抽象为一个点,通过判断点与凸多边形的位置关系来解决近邻检测问题。目前判断点与凸多边形位置关系的方法有:计算点的方位<sup>[2]</sup>、射线法<sup>[3]</sup>、夹角法<sup>[4]</sup>、面积法<sup>[5]</sup>等。相关工作见文献[6-17]。但这些方法普遍存在计算效率不高的问题。目前,人们通常通过移动设备(如智能手机、平板电脑、手环等)去请求 LBS,移动设备存在资源受限的特点,因此,上述方法都不适合直接应用于 LBS。在射线法中,通过给定点引出一条射线,计算该射线与凸多边形的交点数,若交点的数为奇数,则坐标点在凸多边形内部,否则坐标点在凸多边形外。但当给定点位于凸多边形边上时,该方法存在误判的可能性。因此,设计轻量级算法正确求解点与凸多边形位置关系判定问题是一个值得研究的问题。

为了减少算法的计算开销,本文运用了减治思

想,通过减少问题求解过程中与点进行位置关系判断的边的条数来达到提高问题求解效率的目标。基于该思路,本文提出了一种轻量级点与凸多边形位置关系判定算法(称为 LRDA)。算法 LRDA 首先提取凸多边形的四个特征顶点,根据这四个特征顶点对凸多边形进行区域划分,通过判断给定点落在哪个分区内将原问题的判断范围从凸多边形缩小到凸多边形的局部区域,然后调用该分区对应的判断条件对给定点进行位置判定,最终得到点与凸多边形的位置关系。

## 1 算法阐述

### 1.1 问题描述

假设给定的凸多边形  $L$  具有  $n$  个顶点,从纵坐标最大的顶点开始依次进行逆时针编号,假设为  $\{P_1, P_2, \dots, P_n\}$ ,其坐标分别为  $(x_i, y_i), i = 1, 2, \dots, n$ ,并给定一个点  $P(x_p, y_p)$ ,判断点  $P$  与该凸多边形之间的位置关系,即判断点  $P$  是位于凸多边形外部还是位于凸多边形内部。点  $P$  位于凸多边形内部包括点  $P$  在凸多边形边上的情况。

### 1.2 算法设计

1.2.1 凸多边形区域划分 首先找出凸多边形  $L$  的最上顶点、最左顶点、最下顶点、最右顶点四个特征顶点,分别表示为  $P_1, P_l, P_d, P_r$ (图 1)。根据问

[收稿日期] 2021-12-04

[基金项目] 国家自然科学基金(61702168,62072134);广西可信软件重点实验室研究课题(Kx202014)

[第一作者] 张 浩(1997-),男,湖北孝感人,湖北工业大学硕士研究生,研究方向为位置隐私

[通信作者] 沈 华(1978-),女,江苏兴化人,工学博士,湖北工业大学副教授,研究方向为信息安全、隐私保护

题描述中的顶点编号规则可知  $1 \leq l \leq d \leq r \leq n$ 。然后,过最左顶点  $P_l(x_l, y_l)$  和最右顶点  $P_r(x_r, y_r)$  作与  $X$  轴平行的直线,过最上顶点  $P_1(x_1, y_1)$  和最下顶点  $P_d(x_d, y_d)$  作与  $Y$  轴平行的直线,将凸多边形  $L$  划分为图 2 所示的 5 个区域。第 1 个分区对应于图 2 中的空白区域,如果点  $P$  落入这个区域,则说明点  $P$  在凸多边形的外部。第 2 个分区对应于图 2 中标注为 I 的阴影区域、第 3 个分区对应于图 2 中标注为 II 的阴影区域、第 4 个分区对应于图 2 中标注为 III 的阴影区域、第 5 个分区对应于图 2 中标注为 IV 的阴影区域,如果点  $P$  落入这些区域,则需要进一步判断点  $P$  与凸多边形的位置关系。

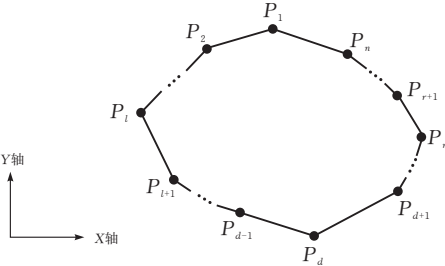


图 1 凸多边形示意图

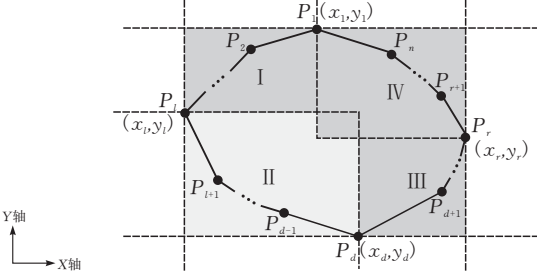


图 2 凸多边形划分示意图

1.2.2 点的区域判断 这里分别给出点  $P$  落入上述 5 个区域的判断条件。

判断条件 1:点落入空白区域。如果  $y_p < y_d$  或者  $y_p > y_1$  或者  $x_p < x_1$  或者  $x_p > x_r$ ,那么点  $P(x_p, y_p)$  落入空白区域(图 3)。

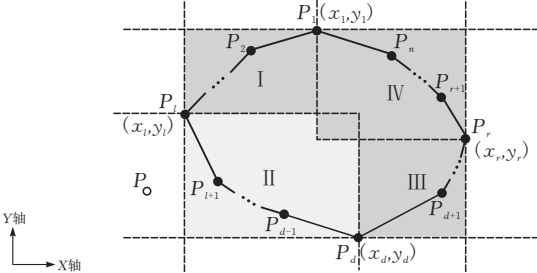
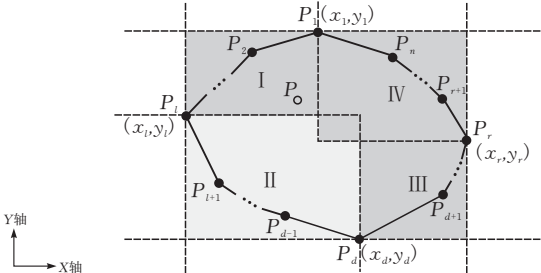


图 3 点  $P$  落入空白区域示意图

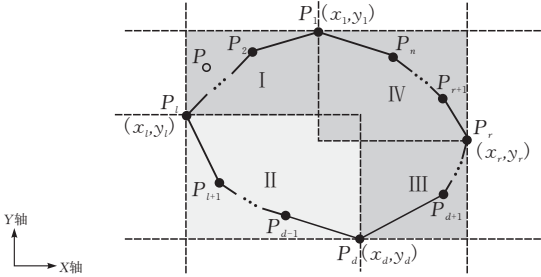
判断条件 2:点落入阴影区域 I。如果  $x_i \leq x_p \leq x_1$  且  $y_l \leq y_p \leq y_i$ ,那么点  $P(x_p, y_p)$  落入阴影区域 I(图 4)。

判断条件 3:点落入阴影区域 II。如果  $x_i \leq x_p \leq x_d$  且  $y_d \leq y_p < y_l$ ,即点  $P(x_p, y_p)$  落入阴影区

域 II(图 5)。

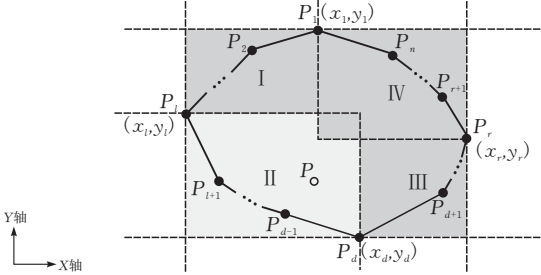


(a)点在区域 I 的凸多边形内

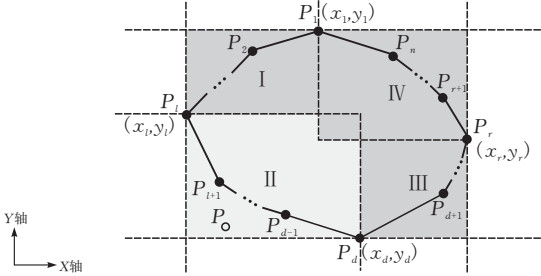


(b)点在区域 I 的凸多边形外

图 4 点  $P$  落入阴影区域 I 示意图



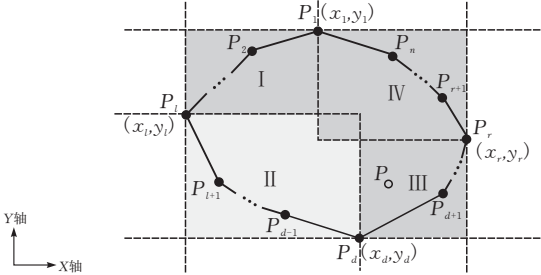
(a)点在区域 II 的凸多边形内



(b)点在区域 II 的凸多边形外

图 5 点  $P$  落入阴影区域 II 示意图

判断条件 4:点落入阴影区域 III。如果  $x_d \leq x_p \leq x_r$ ,即点  $P(x_p, y_p)$  落入阴影区域 III(图 6)。



(a)点在区域 III 的凸多边形内



Step2.判断点落在哪个区域;

Step3.如果点落在空白区域,则立即返回点  $P$  位于凸多边形外部;如果点  $P$  落在阴影区域 I,则调用子算法 1;如果点  $P$  落在阴影区域 II,则调用子算法 2;如果点  $P$  落在阴影区域 III,则调用子算法 3;如果点  $P$  落在阴影区域 IV,则调用子算法 4。

2 算法分析

本小节给出算法 LRDA 的时间复杂度分析。算法 LRDA 中 Step1 的时间开销主要花费在找最上顶点、最下顶点、最左顶点和最右顶点上。可以先将所有顶点按照  $x$  坐标和  $y$  坐标分别进行排序,此过程可以在  $O(n\log n)$  时间内完成;然后,我们可以在  $O(1)$  时间内找到上述四个特征顶点。因此,算法 LRDA 中 Step1 的时间复杂度为  $O(n\log n)$ 。显然,算法 LRDA 中 Step2 的时间开销为  $O(1)$ 。算法 LRDA 中 Step3 的时间开销分析需要考虑点落到 5 个分区的概率分布。用  $E_1$  表示点落在空白区域(即图 2 中的空白区域)的事件,用  $E_2$  表示点落在非空白区域(即图 2 中的 4 个阴影区域)的事件,则有:

$$P(E_1) = \frac{\text{空白区域面积}}{\text{空白区域面积} + \text{非空白区域面积}}$$
$$P(E_2) = \frac{\text{非空白区域面积}}{\text{空白区域面积} + \text{非空白区域面积}}$$

且  $P(E_1) + P(E_2) = 1$ 。显然从几何角度看,空白区域的面积远远大于非空白区域,但考虑到实际应用中,给定点不会距离目标凸多边形太远,因此假设  $P(E_1) = P(E_2) = 1/2$ 。事件  $E_1$  对应于判断条件 1 成立,事件  $E_2$  对应于判断条件 2 或判断条件 3 或判断条件 4 或判断条件 5 成立。如果判断条件 1 成立,则可以立即判定点在凸多边形外部,判定过程的时间开销为  $O(1)$ 。如果判断条件 2 或判断条件 3 或判断条件 4 或判断条件 5 成立,则需要调用子算法 1 或子算法 2 或子算法 3 或子算法 4 得到点与凸多边形的位置关系,判定过程的时间开销为子算法 1 或子算法 2 或子算法 3 或子算法 4 的时间开销。子算法 1、子算法 2、子算法 3 和子算法 4 的时间开销均主要由它们的 Step1 所贡献,其时间复杂度主要取决于 4 个阴影分区所包含的顶点数。假设每个阴影分区所包含的顶点数相同,均为  $n/4$ 。利用算法 LRDA 的 Step1 中顶点的排序结果进行折半搜索,搜索满足条件的顶点  $p_i$  和  $p_{i+1}$ ,则子算法中的 Step1 的时间复杂度为  $O(\log(n/4))$ 。因此,子算法 1、子算法 2、子算法 3 和子算法 4 的时间开销均为  $O(\log(n/4))$ 。因此,算法 LRDA 中 Step3 的时间开销为  $O(1) \times P(E_1) + O(\log(n/4)) \times P(E_2) =$

$O((1/2)\log(n/4))$ ,即为  $O(\log\sqrt{n})$ 。  
综上所述可知,算法 LRDA 的时间复杂度为  $O(n\log n) + O(1) + O(\log\sqrt{n}) = O(n\log n) + O(\log\sqrt{n})$ 。  
需要说明的是,在算法具体实施过程中,算法 LRDA 的时间复杂度可以进一步提高。例如,将凸多边形划分中对顶点按照  $x$  坐标和  $y$  坐标的排序过程作为算法的预处理过程,则算法 LRDA 的时间复杂度为  $O(\log\sqrt{n})$ 。再如,在多实例情况下(即多个点或凸多边形不同的实例),如果多个实例中所涉及的凸多边形相同,那么只有在第一个实例执行时需要进行凸多边形划分,这种情况下可以用平摊分析法对算法 LRDA 进行算法分析得到一个更优的时间性能。

3 算法实施

实施场景 1:基于位置服务中的近邻好友判断。  
场景描述:用户 A 在某商圈逛街,想知道其好友 B 是否也在该商圈范围内。此场景下,算法 LRDA 在位置服务器上运行,算法的输入分别来自用户 A 和用户 B,其中用户 A 的输入被抽象为一个凸多边形,用户 B 的输入被抽象为一个点。

实施场景 2:移动服务站最佳运营路线。  
场景描述:服务型公司 C 为了更好地服务于其客户,决定设置移动服务站点。为了充分发挥移动服务站的作用,需要给出移动服务站每个时间段的最佳服务地点。公司 C 可以通过执行算法 LRDA 获得移动服务站的最佳运营路线。首先,公司 C 确定时间段和选出几个候选区域范围;然后,公司 C 通过执行算法 LRDA 统计出不同时间段出现在候选区域内的客户人数;最后根据统计结果得到移动服务站的最佳运营路线。此场景下,公司 C 执行在给定的时间段内执行算法 LRDA,算法的输入分别为公司 C 选定的候选区域和客户的当前位置。

4 结束语

点与凸多边形位置判定是大多数基于位置服务需要解决的核心问题。针对移动设备资源受限的客观条件,本文基于减治的思想提出了一种轻量级点与凸多边形位置关系判定算法。该算法通过减少与点进行操作的边的条数以降低算法对计算资源的需求,实现了时间效率从  $O(n^2)$  到  $O(\log\sqrt{n})$  的提升。但本文提出的算法仅适用于凸多边形,设计一种轻量级点与任意多边形位置关系判定算法是进一步的工作方向。



[参 考 文 献]

[1] SHEN H, ZHANG M W, WANG H, et al. A light-weight privacy-preserving fair meeting location determination scheme[J]. IEEE Internet of Things Journal, 2020, 7(4): 3038-3093.

[2] FEITO F, TORRES J C, UREÑA A. Orientation, simplicity, and inclusion test for planar polygons[J]. Compute. Graph., 1995, 19(4): 595-600.

[3] GALETZKA M, GLAUNER P O. A simple and correct even-odd algorithm for the point-in-polygon problem for complex polygons[C]. //The 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2017). Porto, Portugal, 2017,175-178.

[4] HORMANN K, AGATHOS A. The point in polygon problem for arbitrary polygons[J]. Computational Geometry, 2001, 20(3): 131-144.

[5] 陈振华,李顺东,黄琼,等. 两个保密位置判断问题的新解法[J]. 计算机学报, 2018, 41(2): 336-348.

[6] ZHU H, WANG F, LU R, et al. Efficient and privacy-preserving proximity detection schemes for social applications[J]. IEEE Internet of Things Journal, 2018, 5(4): 2947-2957.

[7] 孙爱玲,赵光华,赵敏华,等. 基于 sign(x)函数的点在多边形内外判别算法及应用[J]. 计算机工程与科学, 2017, 39(4): 785-790.

[8] 马晨,张毅.一种改进的点与多边形关系的叉乘判别法[J]. 测绘科学, 2013, 38(1): 125-127.

[9] 柳珍. 一种夹角和法结合叉乘法判定点与任意多边形位置关系方法:中国, CN106991698A[P]. 2017-07-28.

[10] 沈陈华. 平面上点与多边形包含关系的 Q 算法[J]. 扬州大学学报(自然科学版), 1999(4): 24-26.

[11] 刘梁. 确定点与多边形以及多边形中顺时针或逆时针拓扑关系的优化算法[J]. 测绘与空间地理信息, 2007, 30(1): 84-86.

[12] 张宏,温永宁,刘爱利. 地理信息系统算法基础[M]. 北京:科学出版社, 2006: 129-131.

[13] 董秀山,刘润涛. 判断点与简单多边形位置关系的新算法[J]. 计算机工程与应用, 2009, 45(2): 185-186.

[14] 翟艳,徐卫亚,张强. 点与多边形或多面体的拓扑关系判断[J]. 计算机工程与设计, 2015(4): 972-976.

[15] 章磊,何芬,李鸿赞.一种基于奇异射线法检测点在多边形内的方法[J]. 计算机应用研究, 2020, 37(S2): 133-135.

[16] 李楠,肖克炎. 一种改进的点在多边形内外判断算法[J]. 计算机工程, 2012, 38(5): 30-34.

[17] 王盛春,王文成,等.加强局部简便计算的点在多边形内的高效判定[J]. 图学学报, 2019, 40(2): 267-273.

Judgment Algorithm for the Position Relationship between Point and Convex Polygon Based on Reduction and Rule

ZHANG Hao<sup>1</sup>, SHEN Hua<sup>1,2</sup>, SHEN Gang<sup>1</sup>

(1 School of Computer Science, Hubei Univ. of Tech., Wuhan 430068, China;

2 Guangxi Key Laboratory of Trusted Software, Guilin Univ. of Electronic Tech., Guilin 541004, China)

**Abstract:** Many location-based services used by people involve the problem of determining the positional relation between points and convex polygons. The limited resources of mobile devices make lightweight algorithms designed to resolve the above issue to be an urgent problem. Therefore, finding a lightweight decision algorithm is necessary, and reducing the number of edges that operate on points has become a feasible idea. Therefore, a lightweight algorithm of position relationship determination between a point and a convex polygon is proposed based on reduction and governance. The algorithm includes: region division, region judgment of a point, and position relationship judgment between a point and a convex polygon. The algorithm converts the position relationship judgment between a point and a convex polygon into the position relationship judgment between the point and a part of the convex polygon, which reduces the time cost. By sorting vertexes of the part of the convex polygon beforehand, the time cost of our algorithm can achieve  $O(\log \sqrt{n})$ . This algorithm can be applied to determine quickly the position relationship between points and convex polygons in scenarios where mobile device resources are limited.

**Keywords:** decrease-and-conquer; convex polygon; region judgment; position relation determination algorithm