地理信息系统理论与应用丛书

计算几何：空间数据处理算法

闫浩文　王明孝　王中辉 编著

科学出版社

地理信息系统理论与应用丛书

计算几何:空间数据处理算法

闫浩文　王明孝　王中辉　编著

科学出版社

北京

内 容 简 介

计算几何是研究几何图形的计算机存储、表达、处理和分析等的理论与技术。而地理信息系统(GIS)处理的主要对象是空间数据(图形数据)。因此,如何运用计算几何的基本理论、方法、算法等为空间数据处理服务,已成为近年来GIS领域研究的焦点问题之一。

本书首先介绍计算几何基元及算法,然后依据计算几何在空间数据处理中的不同作用,分别论述空间分析算法、空间查询算法、空间数据可视化算法、空间关系表达算法及地图自动综合算法。本书注重介绍算法的基本原理与具体的实现过程。其论述深入浅出、图文并茂,便于读者理解与掌握。

本书适合于地理、地图、测量、城建等领域的广大研究人员和技术工作者阅读参考,也可作为地理学科、测绘学科及其他相关学科本科生、研究生的教学用书。

图书在版编目(CIP)数据

计算几何:空间数据处理算法/闫浩文,王明孝,王中辉编著.—北京:科学出版社,2012

(地理信息系统理论与应用丛书)

ISBN 978-7-03-036015-1

Ⅰ.①计… Ⅱ.①闫…②王…③王… Ⅲ.①计算几何-应用-空间测量-数据处理 Ⅳ.①018②P236

中国版本图书馆CIP数据核字(2012)第268481号

责任编辑:韩 鹏 朱海燕 吕晨旭/责任校对:刘亚琦

责任印制:钱玉芬/封面设计:王 浩

科学出版社 出版

北京东黄城根北街16号

邮政编码:100717

http://www.sciencep.com

新科印刷有限公司 印刷

科学出版社发行 各地新华书店经销

*

2012年11月第 一 版 开本:787×1092 1/16

2012年11月第一次印刷 印张:12 3/4

字数:302 000

定价:59.00元

(如有印装质量问题,我社负责调换)

前　言

早在2000年笔者撰写博士学位论文时就已经产生撰写本书的想法。当时，笔者在做空间关系理论问题的研究时，就发现地图上目标之间的空间关系计算问题，即在许多情况下都需要借助于凸包、Delaunay三角网、Voronoi图等来实现，而查阅当时的文献，中、外文书籍数量了了，于是心中闪过撰写本书的念头。后来，笔者在香港理工大学和瑞士苏黎世大学进行地图自动综合方面的研究时，发现计算几何在地图综合的空间描述和算法设计中应用同样广泛，这更加剧了笔者撰写本书的冲动。因此，从2006年开始，笔者尝试在兰州交通大学地图学与地理信息系统专业为硕士研究生开设了两门相关的课程："计算几何"和"地图数据智能化处理"，这使得笔者有机会投入专门的精力为本书写作积累资料、心得和人力支持。经过近5年多的探索，笔者大致理清了本书的思路。于是，在2011年上半年，笔者撰写了本书的提纲，邀请了王明孝博士、王中辉博士作为合作者，共同启动了本书初稿的撰写。

本书的撰写有两个目的：一是综括计算几何在空间数据处理算法方面的最新研究成果，为地图学、地理信息科学等领域的科研人员提供方法和技术手段；二是比较系统地论述计算几何的基本原理及其在空间数据处理中的经典算法，为地理信息学科的高年级本科生、硕士和博士研究生提供课本或课程参考资料。因而，本书在组织上着重于系统性，内容上强调新颖性。

为了论述上的系统性，本书在第1章提出并回答了计算几何与空间数据处理的关系问题，为后续各章的展开进行了导引；然后在第2章介绍了计算几何的基元和相关算法，为把计算几何算法应用到空间数据处理中奠定了基础。接下来进入了本书的重点章节：依据计算几何在空间数据处理中的不同作用，把这些算法归为4类，分别是：第3章"空间分析与空间查询算法"；第4章"空间数据可视化算法"；第5章"空间关系表达算法"；第6章"地图自动综合算法"。最后，第7章对全书进行了总结和展望。

本书的新颖性体现在4个方面。其一是书中的引文追求时效性，即针对某一具体问题的论述，尽量引用最近期的研究成果；其二是把计算几何和空间数据处理结合在一起进行系统论述的构想，到目前为止为本书所仅见；其三是书中给出了一个全新的计算几何在空间数据处理中的算法分类体系；其四是本书的第7章列出了计算几何在空间数据处理中潜在的研究方向，以便同行学者寻找感兴趣的研究课题。

本书撰写的具体分工如下：前言、第1章、第6章和第7章由闫浩文撰写；第4章的第4、5节和第5章由王明孝撰写；第2章、第3章和第4章的第1、2、3节由王中辉撰写。全书由闫浩文统一组织和统稿。

本书的出版得到兰州军区信息工程科技创新工作站、国家自然科学基金(40871208)、863重大项目(2009AA121404)、教育部创新团队资助计划(IRT0966)等的支持。本书完成后的初稿，曾在兰州交通大学地图学与地理信息系统专业2011级研究生中试用。期间，张宁、窦鹏、李双元、易珍言、程亚辉等同学在文字方面进行了修订，在此表示诚挚的谢

意。特别感谢科学出版社领导和韩鹏先生在本书编辑出版中付出的辛勤劳动。

把计算几何理论和空间数据处理的实践结合在一起进行系统论述是空间信息科学中极具挑战性的课题。囿于作者的学识与经验，本书撰写虽然尽心尽力，但论述问题不免挂一漏万，遣词造句可能贻笑大方。文责尽在作者，欢迎同行批评指正。

闫浩文

2012年2月

目　　录

第 1 章　绪　　论

让我们用一个简单的事例开始本书的论述。

设想你是一个大学新生，初次进入陌生的大学校园，炎热难耐之际，想要在校园找个地方购买冷饮。学校里面零零落落地分布了数个冷饮小店。此时，你从哪里知道这些冷饮店的位置，并确定距离你最近的一个呢？一张校园地图无疑是非常合适的选择。在校园地图上，你可以很快确定自己的位置，明白你所在的校园分区，进而用简单的判断确定最佳的冷饮店位置(图 1.1)。

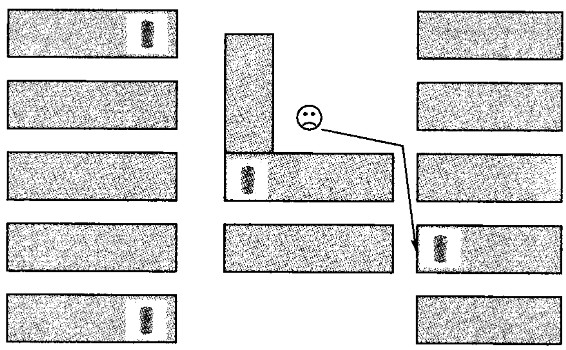


图 1.1　借助校园地图寻找冷饮店

大学新生借助地图寻找校园冷饮店的过程，如果换作机器人来自动实现，就包含了障碍物的识别、最短路径搜寻和计算等问题。简言之，该类问题可归纳为借助于计算机算法对几何图形进行计算的范畴，即属于计算几何(Computational Geometry，CG)所研究的问题。

计算几何的产生只有短短 30 多年的历史，但其发展势头极其迅猛，应用范围也非常广泛。目前，计算几何已成为计算机辅助设计、计算机辅助制造、运动规划、空间数据挖掘、图像处理、地图自动综合等领域的重要研究手段。

1.1　计算几何的概念

计算机科学(Preparata and Shamos，1988)和几何学(罗钟铉等，2010)的学者从各自的研究角度出发，均认为计算几何是自己学科的发展分支，从而各有侧重地对计算几何进行了定义。总括这两类定义，可以认为：计算几何是研究几何图形(或数据、模型)的计算机存储、表达、分析、综合、管理和处理等的理论与技术的总括。由于计算几何更偏重于算法的研究，因此，在许多情况下计算几何也被称为算法几何学(Algorithmic Geometry)或者几何算法学(Geometric Algorithms)。

显然，计算几何的研究对象是几何形体和几何数据，借助的工具是计算机相关学科，

目的是实现图形处理的自动化和智能化(图 1.2)。

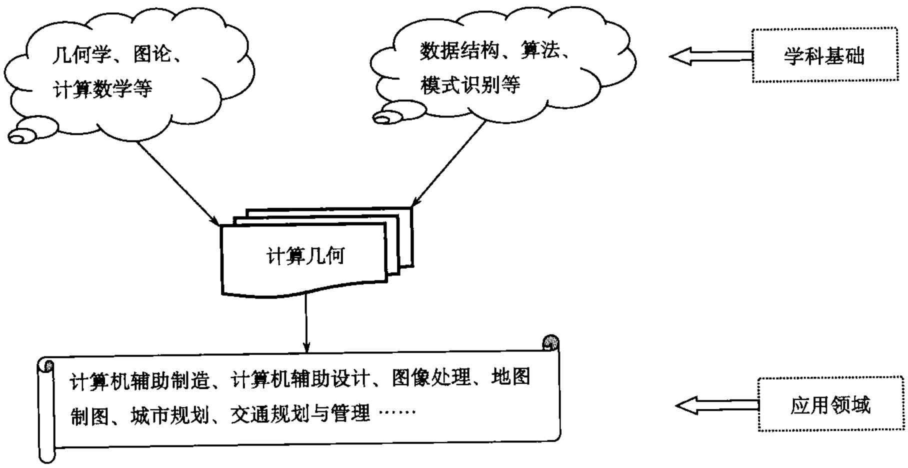


图 1.2 计算几何的相关学科及计算几何的应用领域

1.2 计算几何的缘起与发展

计算几何的"影子"可从 20 世纪 60 年代进行的样条曲线光滑、组合几何学等中窥见一斑。Minsky 与 Papert 在 1969 年出版的著作 *Perceptrons: An Introduction to Computational Geometry* 中最早提出"计算几何"这一术语,并进行了较为系统的论述。该书的出发点是模式识别。随后,Forrest(1971)在一篇名为"*Computational Geometry*"的会议论文中论述了几何曲线和曲面的建模问题。真正对当代计算几何学产生重大影响的标志性事件之一,应该是 Shamos 于 1978 年在耶鲁大学的博士学位论文"*Computational Geometry*"及其前后公开发表的一系列论文。他对计算几何的基本概念、工具和一些算法(如最近点问题、相交问题)进行了研究并给出了非常出色的解答。由于 Shamos 的突出贡献和他本人的计算机学科背景,后来者多把计算几何归结为理论计算机科学(Theoretical Computer Science)中的算法理论(Algorithm Theory)的一个分支。从 20 世纪 80 年代开始,计算几何的研究在世界范围内进入了一个非常活跃的时期。典型的事件,如 1983 年,关于计算几何的第一个国际学术研讨会召开;1985 年,关于计算几何的第一个国际学术大会(Annual Symposium on Computational Geometry)召开,并规定此大会每年举行 1 次;也是在这一年,第一本计算几何的教科书出版(Shamos and Preparata,1985)。

计算几何在中国的起源和发展几乎和世界同步,但是中国学者最初的研究方向却与其他国家有很大差异。1980 年,我国著名数学家苏步青、刘鼎元出版了《计算几何》一书。他们从计算机辅助设计的角度对样条曲线、样条曲面的光滑和变换问题进行了系统阐述;虽然这可视为计算几何在中国起源的标志性成果,但很显然,其着眼点与彼时的国际潮流并不一致。相比较而言,周培德(2000)撰写的著作《计算几何——算法设计与分析》,其内容及研究手段、研究思路、应用领域等与国际上计算几何研究的主流较为吻合。虽然,计算几何在中国已经得到了深入的研究和广泛的应用,也有部分高校开设了专门的计算几

何课程;但是到目前为止,中国尚未成立专门的计算几何学会或协会,也未见专门的计算几何学术期刊发行。然而,鉴于计算几何在计算机科学、数学、空间信息处理、图形分析与设计等领域的重要地位,相关学术组织和学术期刊的出现可以预期。

1.3 从空间数据处理到计算几何算法

对地理信息系统(Geographic Information System,GIS)而言,其处理的对象是地理空间数据(简称空间数据)。GIS 的空间数据一般可分为图形数据和属性数据两类,且图形数据是 GIS 研究的重点。从此意义上看,GIS 的研究对象和计算几何的研究对象(几何图形数据)存在着很大的共性。因此,对于 GIS 领域的学者而言,研究如何运用计算几何的理论、方法、算法等为空间数据处理服务,就成为自然而然的事情。下面从几个例子来概括说明计算几何在空间数据处理中的应用。

1.3.1 点击选取目标

计算机屏幕上的地图目标,就其几何形体而言,无非是点、线、面三类。在计算机屏幕所在的二维空间点击选取目标,本质上是确定鼠标点击时所指的点与欲选中的目标之间的位置关系(图 1.3)。

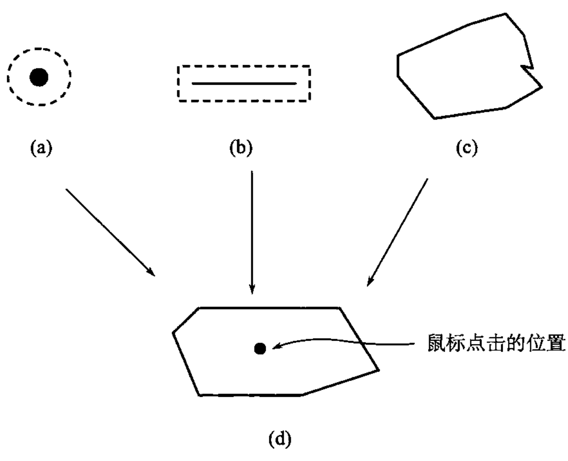


图 1.3 鼠标在计算机屏幕上点击选取目标

用鼠标点击点目标时,由于数学意义上的点在屏幕上面积为零,所以鼠标不能捕捉到点目标。在实际中用该点的一个缓冲圆形区域来代替原始点目标[图 1.3(a)];当鼠标点击的位置在缓冲圆中时,即认为原始点目标被选中。

用鼠标点击线目标时,同样地,几何意义上的线没有宽度,只是在线所经过的位置占据单位像元的位置,用鼠标极难“点准”。为了选中线目标,通常为线目标设定一个缓冲区,并认为鼠标点击到该缓冲区时,就代表该线目标被选中[图 1.3(b)]。

面目标则占据了屏幕上一定大小的实际区域,鼠标点击到该区域,就表示该面目标将

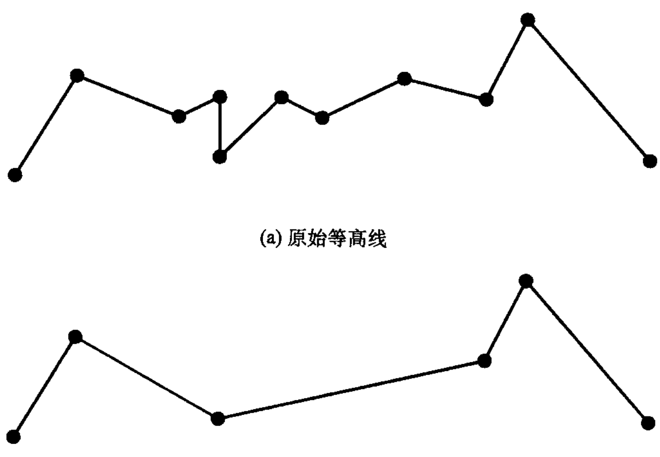
被选中[图 1.3(c)]。

总括以上点、线、面三类目标的选取原理可知，在计算机屏幕上用鼠标点击选取目标，其关键就是(鼠标)点与多边形(所点击目标的抽象)的包含关系判断[图 1.3(d)]。这显然是一个几何图形的算法问题，比较典型的答案可见于地图数据处理中的铅垂线内点算法。

1.3.2 曲线化简

据统计，地图上至少80%的符号为线状符号，因此曲线的处理是地图数据处理的主要内容。曲线化简是地图上线状目标可视化表达、地图数据网络传输等的常用运算。其目标是，在地图比例尺缩小时，去掉在视觉上多余的点，使地图上的线条符号和人类的视觉感受相一致。为此就需要区分在比例尺变化到一定幅度时，线目标上的哪些点仍然是特征点而需要保留，哪些点已不重要而需要删除。如果抛开曲线上点位的属性信息，这一问题就简化为寻找几何曲线上的特征点的问题，即计算几何算法问题。

到目前为止，曲线化简算法还以 Douglas 和 Peucker(1973)提出的算法最为经典，广为地图数据处理、图像处理、计算机辅助设计与制造等领域应用。在空间数据处理中，该算法被用在等高线化简、行政区域多边形化简等方面。图 1.4 是 Douglas-Peucker 算法化简单根等高线的一个例子。



(a) 原始等高线

(b) 化简后的等高线

图 1.4 基于 Douglas-Peucker 算法化简等高线

1.3.3 居民地综合

在大、中比例尺的地图上，居民地一般呈离散、面状分布。当比例尺缩小时，由于地图图幅面积的限制，居民地符号之间、居民地与其他地物符号会出现重叠和压盖，此时需要对居民地进行合并、化简等操作。这一问题交由计算机自动完成时，就需要计算居民地之

间的拓扑关系、距离关系、方向关系等，并由此确定应运用何种操作对相应的居民地群组进行合适的处理(闫浩文和王家耀，2009)。显然这主要是一个面状几何图形的计算问题。曾有学者(Yan et al.，2008)借助于计算几何中的 Delaunay 三角剖分，给出了这一问题的答案(图 1.5)。

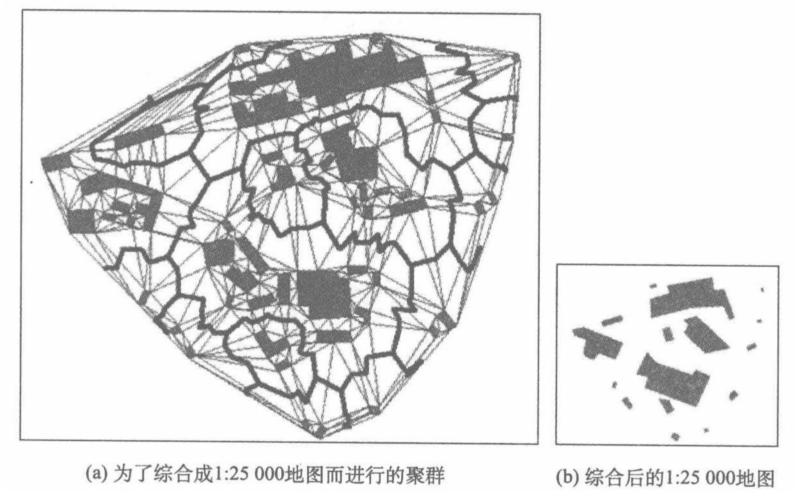


图 1.5　借助于 Delaunay 三角剖分的居民地综合方法
原图比例尺为 1∶10 000

计算几何在空间数据处理中的应用远非上述的例子所能概括。除了矢量数据，计算几何在栅格数据中同样应用广泛。国内外均有学者开发了专门的计算几何算法库，甚至把许多算法的控件库在网络上公布出来，免费让用户下载使用。

1.4　本书的组织和约定

在展开论述之前，对本书的研究对象、研究范围、研究重点等进行约定是必要的。

(1) 本书的研究对象是地理空间的矢量数据，除非特殊说明或部分章节的特殊需要，本书对栅格数据不作讨论。因此，本书所谓的地理空间数据可理解为表达地理空间的矢量数据。

(2) 本书研究的地理空间数据一般是二维的，且以地图数据作为研究重点。

(3) 本书不是计算几何的教科书，故无意涉猎计算几何的全部内容，而将专注于计算几何的基本理论和方法在地理空间数据处理中的算法方面。

本书的基本结构如下：

本章为绪论，主要论述计算几何的基本概念及其和空间数据处理的关系。

第 2 章论述计算几何的基元和相关算法，为后续章节中把计算几何算法应用到空间数据处理中奠定基础。

第 3～6 章是本书的重点。本书依据计算几何在空间数据处理中的不同作用，把这些算法分别归类到不同的章节，分别是：第 3 章“空间分析与空间查询算法”；第 4 章“空间数

据可视化算法”;第 5 章“空间关系表达算法”;第 6 章“地图自动综合算法”。当然,本书不可能穷尽相关的所有算法,而是着重于新的、国际认同度高的算法。

第 7 章是本书的结束语,一方面对全书进行总结;另一方面提出本书研究议题中潜在的许多难题、问题,以便和同行学者分享。

主要参考文献

罗钟铉,孟兆良,刘成明. 2010. 计算几何. 北京:科学出版社

苏步青,刘鼎元. 1980. 计算几何. 上海:上海科学技术出版社

闫浩文,王家耀. 2009. 地图群(组)目标描述与自动综合. 北京:科学出版社

周培德. 2000. 计算几何——算法设计与分析. 北京:清华大学出版社

Douglas D H, Peucker T K. 1973. Algorithms for the reduction of the number of points required to represent a digitised line or its caricature. The Canadian cartographer, 10(2):112～122

Forrest A R. 1971. Computational Geometry. *In*: Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences, 321(1545):187～195

Minsky M, Papert S. 1969. Perceptrons: An Introduction to Computational Geometry. Boston: The MIT Press

Preparata F P, Shamos M I. 1988. Computational Geometry: an introduction. New York: Springer-verlag

Shamos M I, Preparata F P. 1985. Computational Geometry: an introduction. Springer-verlag

Shamos M I. 1978. Ph. D Thesis. Yale University

Yan H W, Weibel R, Yang B S. 2008. A multi-parameter approach to automated building grouping and generalization. Geoinformatica, 12(1):73～89

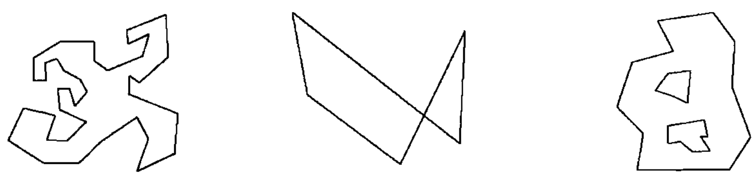
第2章　计算几何基元及算法

几何基元(Geometric Primitives)是计算几何研究的典型问题,包括多边形、凸壳、Voronoi图、Delaunay三角网、曲线和图等。它是研究计算几何其他问题的理论基础与必备前提,在空间数据处理算法中有着非常重要而广泛的应用,本章将就上述几类几何基元及其相关算法进行详细阐述。

2.1　多　边　形

2.1.1　简单多边形与复杂多边形

多边形分为简单多边形与复杂多边形。简单多边形是指边之间除顶点外不相交的多边形,即多边形的每个顶点的度数为2,如图2.1(a)所示。而复杂多边形则是指各边可以自相交,如图2.1(b)所示,或是带有"洞"的多边形,如图2.1(c)所示。对于复杂多边形,在矢量方法里,至今还没有好的解决办法。本书所讨论的多边形,如没有特别说明均指简单多边形。



(a) 简单多边形　　(b) 自相交的复杂多边形　　(c) 带"洞"的复杂多边形

图2.1　简单多边形和复杂多边形

简单多边形分为凹多边形与凸多边形两类。凹多边形如图2.1(a)所示。凸多边形是一种特殊的多边形,如图2.2所示。它具有许多特性,如凸多边形的所有顶点均在任一条边的同一侧,所有内角都小于π,所有顶点均为凸点等。因此在许多问题中常常将多边形分割成凸多边形,特别是分割成三角形。另外,在空间数据处理及分析中常常用到的凸壳也是一凸多边形。

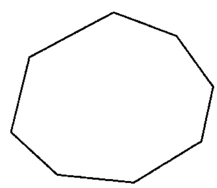


图2.2　凸多边形

2.1.2　多边形的方向及顶点凹凸性的判定算法

1. 基本概念

在对空间数据进行处理时,常常需要判断多边形的方向和顶点的凹凸性。为了叙述

上的方便，首先给出以下几个相关的定义。

定义 2.1 若多边形顶点 $P_1, P_2, P_3, \cdots, P_n$，依次按逆时针方向排列，且沿着多边形的任意一条有向边从起点走到终点时，左侧始终在多边形的内部，则称该多边形的方向为逆时针方向或正方向；否则称为顺时针方向或负方向。

定义 2.2 设 $P_1, P_2, P_3, \cdots, P_n, P_{n+1}=P_1$ 表示一个多边形。由 $P_{i-1}P_i, P_iP_{i+1}$ 所形成的内角（即多边形内部的角）记为 $\theta(\theta\neq 0, \pi, 2\pi)$，若多边形某顶点 P_i 的内角 $\theta\in(0,\pi)$ 则该顶点是凸顶点；若 $\theta\in(\pi, 2\pi)$，则该顶点是凹顶点。

定义 2.3 在多边形的顶点中，将 x 坐标最小（大）的顶点，y 坐标最小（大）的顶点统称为多边形的极值顶点，极值顶点必为凸顶点。

定义 2.4 多边形的所有边均可看作有向边，边的方向与顶点的串联方向一致。对于每个顶点，均可求其前后相邻边的矢量叉积，得到的矢量称为顶点矢量。

2. 算法原理

在判断多边形的方向及其顶点的凹凸性时，涉及两个基本问题：

(1) 已知多边形某一顶点的凹凸性，确定多边形的方向；

(2) 已知多边形的方向，确定多边形任一顶点的凹凸性。

本节算法将上述两个基本问题结合起来进行考虑，通过多边形方向和其任一凸顶点之间的内在联系，确定出多边形的方向，然后再进一步由多边形的方向计算出其余各顶点的凹凸性（王中辉，2009）。具体算法原理叙述如下。

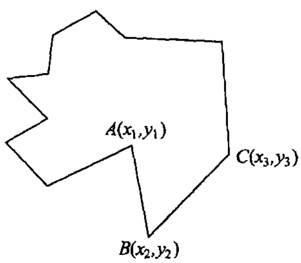


图 2.3 多边形方向的判断

如图 2.3 所示，设 $B(x_2, y_2)$ 是多边形的任意一个凸顶点，B 的前驱顶点为 $A(x_1, y_1)$，后继顶点为 $C(x_3, y_3)$。计算 B 的顶点矢量 K，得到式(2.1)：

$$K=(x_2-x_1)\times(y_3-y_2)-(x_3-x_2)\times(y_2-y_1) \tag{2.1}$$

若 $K>0$，多边形的方向为逆时针；反之，若 $K<0$，则多边形的方向为顺时针。

由多边形的性质可知，多边形的极值顶点必为凸顶点（本节算法中选择 y 坐标最小的极值点）。

同样，若已知多边形的方向为逆时针（顺时针），利用式(2.1)计算多边形各顶点的顶点矢量，若其值大于（小于）零，则该顶点为凸顶点；其值小于（大于）零，则为凹顶点。

3. 算法描述

设 $P_i(x_i, y_i)\{i=1,2,3,\cdots,n\}$ 为多边形的一组有序顶点。

算法 1：多边形方向的判定。

第一步，遍历多边形顶点 P_1 至 P_n，找到 y 坐标最小的顶点 B；

第二步，利用式(2.1)计算凸顶点 B 的顶点矢量 K，若 $K>0$，则 flag=1，若 $K<0$，则 flag=−1；

第三步，判断 flag 的值，若为 1，则多边形的方向为逆时针；若为−1，则多边形的方向

为顺时针。

算法 2:多边形顶点凹凸性的判定。

第一步,利用算法 1 得到 flag 的值。

第二步,利用式(2.1)计算 $P_i(x_i, y_i)\{i=1,2,3,\cdots,n\}$的顶点矢量 K_i。

第三步,如果 flag=1,若 $K_i>0$,则 P_i 为凸顶点,若 $K_i<0$,则 P_i 为凹顶点;如果 flag=-1,若 $K_i<0$,则 P_i 为凸顶点,若 $K_i>0$,则 P_i 为凹顶点。

2.2 凸 壳

2.2.1 凸壳的定义

定义 2.5 平面点集 S 的凸壳(Convex Hull)(或凸包、凸多边形)是指包含 S 的最小凸集,通常用 CH(S)来表示。从几何直观上判断,S 的凸壳表现为 S 中任意两点所连的线段全部位于 S 中。平面点集 S 的凸壳边界 BCH(S)是一个凸多边形,多边形的顶点必定为 S 中的点(图 2.4)。凸壳是计算几何中最普遍、最基本的一种结构。在应用中,许多实际问题都可以归结为凸壳问题,如运动路线规划中的货郎担问题(周培德,2000),制图综合中的群点化简问题(毋河海,2000)等,都可以用凸壳解决。

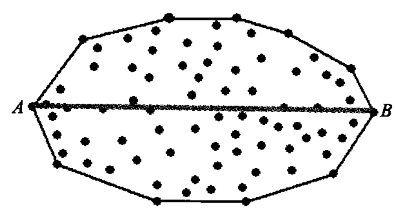


图 2.4 平面点集的凸壳

定义 2.6 凸多边形直径又称为平面点集的直径或平面点集凸壳的直径,定义为凸多边形顶点序列中距离最大的点对的连线,如图 2.4 中的线段 AB。

2.2.2 平面点集的凸壳求解算法

求解平面点集凸壳的算法有卷包裹法(Chand and KaPur,1970)、格雷厄姆算法(Graham,1972)、分治算法(PreParata and Hong,1977)、增量算法(Edelsbrunner,1982)及实时凸壳算法(PreParata,1979)等。

下面介绍格雷厄姆算法,该算法是求解群点凸壳的时间最优算法。

第一步,对于有 N 个点的点集 S,若 $N\leqslant 3$,输出 S,结束运算,否则转第二步。

第二步,找出 S 中 y 值最小的点(设为点 1),计算该点与其他点的连线和水平线的夹角,如图 2.5(a)所示。该夹角的起算边是 x 轴的正半轴,顺时针旋转为负,逆时针旋转为正。然后按照夹角大小和各点与点 1 的距离进行词典排序,得到序列 $1,2,\cdots,N$,依次连接这些点,得到一个多边形,如图 2.5(b)所示。1、2、N 必然是凸壳上的点,如图 2.5(c)所示。

第三步,删除 3 到 $N-1$ 中的凹点,方法是:从第 3 点开始,设它为当前点 i,判断当前点 i 的后一点 $i+1$ 与第 $i-2$ 点是否在当前点 i 与它前面一点 $i-1$ 连线的同侧位,若在同侧,保留当前点,否则删除当前点。

如图 2.5(c)中,点 3 为当前点时,由于 1、4 点位于 2、3 连线的两侧,故删除点 3,得到

图 2.5(d)的结果;同样可以判断,点 4、7 为当前点时,被保留,而点 5、6 为当前点时被删除,得到图 2.5(e)的结果。

第四步,顺序输出凸壳顶点,如图 2.5(f)所示。

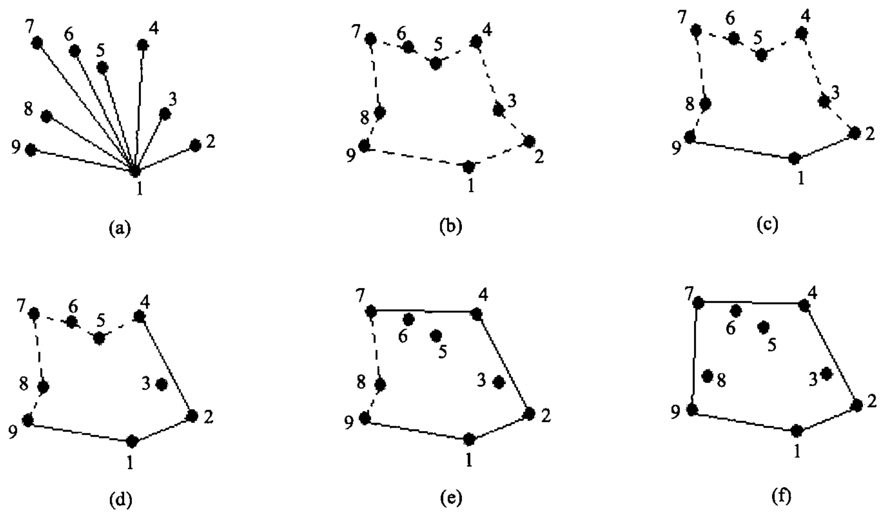


图 2.5　格雷厄姆算法求解凸壳的过程

2.2.3　平面点集的凸壳直径求解算法

多边形的直径是指连接多边形两顶点的线段的最长者。求凸 N 边形直径最简单、直观的方法是计算 $N(N-1)/2$ 个点对的距离,选取其中的最大者。这种算法显然需要 $O(N^2)$的时间复杂度。PreParata 和 Shamos(1988)设计了一种对趾点对算法,在预处理后时间复杂度为 $O(N)$。周培德(2000)提出的夹角序列算法与上面两种方法相比,不需要预处理,只需要 N 次求距离运算和 $N-1$ 次比较便可以得到凸 N 边形直径。

夹角序列算法是目前效率最高的算法。下面的三个引理包含了算法的主要思想,由于其证明很简单,故只附图给出引理,不作证明。

引理 1　图 2.6(a)中,D 是任意$\triangle ABC$ 的底边 BC 的中点,如果$\angle ADC<\pi/2$,则 $AC<AB$。

引理 2　设两个三角形$\triangle ABC$、$\triangle ACE$ 的位置如图 2.6(b)所示。D_1、D_2 分别是 BC、CE 的中点,如果$\angle AD_1B<\pi/2$,$\angle AD_2C<\pi/2$,则 $AB<AC<AE$。

引理 3　设 4 个三角形$\triangle ABC$、$\triangle ACE$、$\triangle AEF$、$\triangle AFG$ 的位置如图 2.6(c)所示。D_1、D_2、D_3、D_4 分别是所在三角形底边的中点,如果$\angle AD_1B<\pi/2$,$\angle AD_2C<\pi/2$,$\angle AD_3E>\pi/2$,$\angle AD_4F>\pi/2$,则有 $AB<AC<AE$,$AG<AF<AE$。

根据上面的引理,夹角序列算法如下:

第一步,计算凸壳各边中点的坐标。

第二步,从第 1 点起计算该点与各边中点连线和底边的夹角,并根据引理 3 判断循环中的一个距离最大值。

第三步,变换开始点,重复第二步,直到各点都被遍历。最后得到的距离最大值 $D=$