

一种多边形道路网络拓扑生成算法的设计与实现

李 杰¹, 张文栋¹, 张 樾²

(1 中北大学仪器科学与动态测试教育部重点实验室, 中北大学电子科学与技术系, 山西太原 030051; 2 中北大学自动控制系, 山西太原 030051)

摘要: 在分析多边形道路网络组成特点的基础上, 提出一种生成多边形道路网络拓扑结构的有效算法, 给出了该算法详细的设计与实现过程. 算法的基本思想是先对多边形道路网络进行三角剖分, 然后生成其中轴线, 最后由中轴线生成其拓扑结构. 实验结果表明所提算法是正确和有效的.

关键词: 道路网络; 多边形; 三角剖分

中图分类号: TP301.6 TP393.02 **文献标识码:** A **文章编号:** 0372-2112 (2006) 08-1396-05

Design and Implementation of an Algorithm Creating the Topology of the Polygon Road Net

LI Jie¹, ZHANG Wen-dong¹, ZHANG Xi²

(1 Key Laboratory of Instrumentation Science and Dynamic Measurement (North University of China), Ministry of Education, Department of Electronic Science and Technology, North University of China, Taiyuan, Shanxi 030051, China; 2 Department of Automatic Control, North University of China, Taiyuan, Shanxi 030051, China)

Abstract Based on analyzing to the composing feature of the polygon road net, an effective algorithm of creating the topology of the polygon road net is proposed and its design and implementation process is given detailedly. The basic thought of the algorithm is first to triangulate the polygon road net, then to create its media axis, finally to generate its topology structure. The experimental results show the algorithm presented in this paper is correct and effective.

Key words road net, polygon, polygon, triangulation

1 引言

道路网络的拓扑结构描述的是道路网络中节点与节点、节点与路段及路段与路段之间的相互连接关系, 具有拓扑结构的道路网络是实现车辆定位导航系统中路径规划及地图匹配等功能的基础^[1]. 道路网络拓扑生成算法是用来生成道路网络拓扑结构的算法, 它是车辆定位导航系统中的主要技术问题之一.

目前, 国内外对道路网络拓扑生成算法的研究主要是针对折线道路网络 (即用折线表示的道路网络)^[2], 应用这些算法时要求道路网络中的每条道路由代表其中心线的折线段表示. 然而实际应用中所能得到的数字地图多数是由航测图片、遥感图片或纸质地图等经过扫描后再矢量化而成. 由于从这种数字地图中能直接获取的数据通常是道路的轮廓线数据, 而不是其中心线数据, 即得到的道路网络是多边形道路网络, 因此, 现有的折线道路网络拓扑生成算法不再适用. 本文针对这一问题, 分析了多边形道路网络的组成特点, 提出了一种基于三角剖分思想的多边形道路网络拓扑生成算法.

2 预备知识

(1) 道路网络中的路段在计算机中一般由带有宽度值的折线段或多边形表示, 相应的道路网络称之为折线段道路网络或多边形道路网络. 在多边形道路网络中, 不仅每条道路本身是简单多边形区域, 而且由若干多边形道路经多边形的交、并、差等运算后所形成的多边形道路网络也同样是简单多边形区域^[7].

(2) 简单多边形区域是指由若干互不相交的简单多边形所围成的区域^[5], 它可能由一个或多个简单多边形围成, 当它由多个简单多边形围成时, 通常将位于区域最外层的简单多边形称作外环多边形, 将位于其内部的简单多边形统称为内环多边形. 简单多边形则被定义为具有下列性质的多边形: 所有顶点各不相同; 任意顶点只属于它所在的边; 任意两边互不相交. 本文将其中任意两边互不相交进一步推广为任意两边可以重叠但不相交, 从而得到广义的简单多边形及简单多边形区域.

(3) 简单多边形的方向由其顶点排列顺序决定的^[3-6, 8], 若一个简单多边形的顶点是按逆时针方向排列,

则其为逆时针方向或正向, 否则为顺时针方向或负向. 简单多边形顶点凹凸性的判断方法是^[5]: 设简单多边形的任意一个顶点 P_i 与其前后两个顶点 P_{i-1} 和 P_{i+1} 所形成的角度为 $\angle P_{i-1}P_iP_{i+1}$, 从多边形区域内部看去, 若 $\angle P_{i-1}P_iP_{i+1} < 180^\circ$, 则顶点 P_i 是凸顶点, 否则, P_i 是凹顶点.

(4)简单多边形区域的三角剖分是指将简单多边形区域 P 分解为一系列互不相交的三角形, 同时不产生新的顶点^[3-6]. 简单多边形区域的 $Delaunay$ 三角剖分是一种特殊的三角剖分, 它满足最小角最大的性质, 即所剖分出的三角形集合中的最小内角较其他三角剖分对应的最小内角大^[3-6]. 简单多边形区域的任一三角剖分, 总可以通过对所剖分出三角形中不满足局部优化性质的非边界边进行有限次局部优化操作后转化为 $Delaunay$ 三角剖分.

(5)依据三角形中三条边是简单多边形区域边界边的数目, 可将简单多边形区域三角剖分得到的三角形分为三种类型^[9]: I类三角形 (有两条边属于边界边, 通常位于道路的始末端点); II类三角形 (有一条边属于边界边, 通常位于道路的中间); III类三角形 (三条边均不属于边界, 通常位于道路的交叉口).

(6)一个三角形的权值定义为其三个内角的最小值.

3 多边形道路网络拓扑生成算法描述

本文提出的多边形道路网络拓扑生成算法的基本思想是, 首先通过多边形运算得到与多边形道路网络对应的多边形区域, 然后对获得的多边形区域进行 $Delaunay$ 三角剖分, 进而提生成多边形区域的中轴线, 最后由中轴线生成多边形道路网络的拓扑结构. 整个程序设计的基本流程如图 1 所示. 限于文章篇幅, 同时考虑到目前有关多边形运算的算法已经比较成熟, 本文对其不作详细介绍, 感兴趣的读者可以参考文献^[7], 此处重点介绍简单多边形区域的 $Delaunay$ 三角剖分、多边形道路网络中轴线的生成及由中轴线生成多边形道路网络拓扑结构的方法.

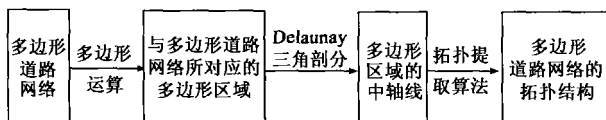


图 1 多边形道路网络拓扑生成算法的程序流程框图

3.1 基于凹凸顶点判定的简单多边形区域 $Delaunay$ 三角剖分

为了对与多边形道路网络对应的简单多边形区域进行 $Delaunay$ 三角剖分, 首先需进行以下 3 步预处理:

(1)简单多边形的正向化处理. 目的是将围成简单多边形区域的所有简单多边形的方向转换为正向, 过程是先从简单多边形的最左点 V_{min} 、最右点 V_{max} 、最下点 V_{min} 、最上点 V_{max} 中任意选择一凸点 P_i , 然后判断点 P_i 在有向线段 $P_{i-1}P_{i+1}$ 的哪一侧, 如果在右侧, 说明该简单多边形的方向是逆时针方向, 如果在左侧, 说明其方向是顺时针方向, 依此将围成简单多

形区域的所有简单多边形方向转换为正向.

(2)简单多边形的边界加密处理. 目的是尽可能地避免 $Delaunay$ 三角剖分时出现狭长三角形, 以便提取出的中轴线更精确, 得到比较准确的多边形道路网络拓扑结构. 方法是给定道路网络中所有道路的最小宽度值作为指标 d , 在长度大于 d 的边界边上以 d 为间距均匀地插入新的边界点, 从而使最终任意一条边界边的长度不超过 d .

(3)外环简单多边形和内环简单多边形的合并处理. 目的是将围成简单多边形区域的外环和内环简单多边形合并为一个广义简单多边形. 方法是将所有内环按其最高点 (即每个内环中 y 值最大的点) 的 y 值从大到小排序, 取当前排序最前的内环, 在外环中寻找 y 值比该内环最高点的 y 值大, 且到最高点距离最小的点, 将该点与最高点相连, 内环并入外环. 如此逐个处理内环, 最后将带有内环的简单多边形区域转化为只有一个环的广义简单多边形区域.

在完成以上 3 步预处理后依次执行下面两个算法得到多边形道路网络的 $Delaunay$ 三角剖分.

算法 1 基于凹凸顶点判定的广义简单多边形区域 $Delaunay$ 三角剖分^[5]

输入: 经过预处理之后的广义简单多边形区域 P 的边界顶点序列 P_1, P_2, \dots, P_n

输出: 广义简单多边形区域 P 的 $Delaunay$ 三角剖分结果

步 1: 读入 P 的顶点, 建立表示结点的双向循环链表 $NodeInkList$

步 2 计算 $NodeInkList$ 中每个结点的凹凸性;

步 3 对 $NodeInkList$ 中的每个凸结点 Q , 设由 Q 与其前后结点 P 和 R 组成的三角形为 $\triangle PQR$ (方向为正向), 若 $\triangle PQR$ 内不包含多边形的其他顶点, 则求出该三角形的权值, 从这类三角形中求出权值最大的三角形, 设其为 $\triangle ABC$, 把 $\triangle ABC$ 的顶点序号保存到三角形链表 $TriangleInkList$ 表中, 并从结点链表 $NodeInkList$ 中删除结点 B ;

步 4 若 $NodeInkList$ 中存在 3 个以上结点, 则转步 2, 否则转步 5;

步 5 由 $NodeInkList$ 中最后 3 个结点所对应的多边形顶点构成一个三角形, 将该三角形的顶点序号保存到 $TriangleInkList$ 中, 同时删除 $NodeInkList$ 中的最后 3 个结点;

步 6 按最小角最大准则, 通过局部变换, 得到 P 的 $Delaunay$ 三角剖分 (见算法 2).

算法 2 三角剖分结果的局部优化^[3]

输入: 三角剖分得到的三角形链表 $TriangleInkList$

输出: 满足最小角最大性质的三角形链表 $TriangleInkList$

步 1: 由 $TriangleInkList$ 生成三角形的边链表 $EdgeInkList$

步 2 对 $EdgeInkList$ 中的每一条非边界边, 取出以该边为邻接边的 2 个相邻三角形构成一个四边形, 判断该四边形的凹凸性, 如果是凹的, 不做任何处理, 如果是凸的,

计算 2 个三角形的最小内角 A , 交换该四边形的两条对角线后计算出 2 个新三角形的最小内角 B , 如果 $B > A$, 则以交换对角线后所得到的 2 个新三角形替换原来的两个三角形, 并按新三角形修改 $TriangleLinkList$ 同时, 统计边交换次数的变量 $ChangeTime$ 累加 1 (初始值为 0);

步 3 判断 $ChangeTime$ 的值, 若等于 Q 表明对所有非边界边的局部优化操作完毕, 程序转步 4 否则, 转步 1;

步 4 返回 $TriangleLinkList$

3.2 简单多边形区域的中轴线提取

简单多边形区域的中轴线提取, 是通过寻找其 Delaunay 三角剖分所得三角形的中轴线来实现的, 具体过程见算法 3

算法 3 简单多边形区域的中轴线提取^[9]

输入: 简单多边形区域 Delaunay 三角剖分所形成的三角形链表 $TriangleLinkList$

输出: 简单多边形区域的中轴线 $MediaAxis$

步 1 将指针 Ptr 指向 $TriangleLinkList$ 的首结点;

步 2 判断 Ptr 是否为空, 若为空, 表示 $TriangleLinkList$ 中所有三角形的中轴线都提取完毕, 程序转步 4 否则, 取出 Ptr 所指向的三角形 T , 同时将指针 Ptr 指向其下一结点;

步 3 依据 T 的类型提取其相应的中轴线 (I 类三角形的中轴线定义为其重心与非边界边中点的连线, II 类三角形的中轴线定义为其两条非边界边各自中点的连线, III 类三角形的中轴线定义为其重心分别与三条非边界边中点的连线), 将所得到的中轴线保存到 $MediaAxis$ 中;

步 4 返回 $MediaAxis$

应用算法 3 所提取出的中轴线通常还存在以下 3 个问题: (1) 存在伪分支, 形成的原因是道路网络边界本身的不平滑; (2) 存在冗余点, 形成的主要原因是预处理阶段的边界加密处理; (3) 交叉路口的交点不唯一, 十字路口等多条道路的交叉处往往出现多个三叉路口而非一个多叉路口, 形成的原因是基于三角剖分提取出的中轴线, 其中的任意一个结点最多与其他三个结点相连.

考虑到以上 3 个问题相互独立, 此处采用以下 3 个算法分别对算法 3 获得的中轴线进一步优化处理.

算法 4 删除中轴线中的伪分支

输入: 中轴线 $MediaAxis$ 及分支长度的下限值 L_{min}

输出: 删除伪分支后的中轴线 $MediaAxis$

步 1 标记 $MediaAxis$ 中所有节点为未读;

步 2 判断 $MediaAxis$ 中未读节点的数目, 若为 Q 转步 7

步 3 取 $MediaAxis$ 中任意未读节点 v_i , 标记为已读, 若 v_i 是非端节点, 转步 2

步 4 从 v_i 沿 $MediaAxis$ 寻找所有二度节点, 直至遇到一个非二度节点 v_j 为止, 得到一条分支 $v_i, v_1, v_2, \dots, v_{j-1}, v_j$, 计算该分支的长度 L , 同时, 将该分支上的所有节点标记为已读;

步 5 比较 L 与 L_{min} , 若 $L < L_{min}$, 则从 $MediaAxis$ 中删除 v_i, v_1, \dots, v_{j-1} 及其相关连接, 转步 2

步 6 返回 $MediaAxis$

算法 5 删除中轴线中的冗余点^[10]

输入: 中轴线 $MediaAxis$ 角度指标 α 及密集度指标 d

输出: 删除冗余点后的中轴线 $MediaAxis$

步 1 标记 $MediaAxis$ 中所有节点为未读;

步 2 若 $MediaAxis$ 中不存在度数为 2 的未读节点, 转步 6

步 3 取任意未读的 2 度节点 v , 由 $MediaAxis$ 中节点间邻接关系生成 v 所在的一条分支 (该分支除了两端点是非 2 度节点外, 其余节点都是 2 度节点), 同时将该分支上的节点标记为已读;

步 4 根据角度指标 α 删除该分支上的冗余点;

步 5 根据密集度指标 d 合并该分支上的密集点, 转步 2

步 6 返回 $MediaAxis$

算法 6 合并中轴线中交叉路口处的多个三叉路口为一个多叉路口

输入: 中轴线 $MediaAxis$

输出: 交叉路口用一个多叉路口表示的中轴线 $MediaAxis$

步 1 标记 $MediaAxis$ 的所有节点为未读;

步 2 若 $MediaAxis$ 中已无未读节点, 转步 9

步 3 取 $MediaAxis$ 中任意未读节点 v_i , 标记 v_i 为已读, 若 v_i 非 3 度节点, 转步 2

步 4 清空节点集 S , 将节点 v_i 存入 S ;

步 5 若存在不属于 S , 且与 S 内某个节点相邻的三度节点 v_j (定义两个 3 度节点 v_1, v_2 相邻为它们通过某个 2 度节点相连, 且记该 2 度节点为 $v_{1,2}$), 标记 v_j 为已读, 转步 6 否则, 转步 7

步 6 将节点 v_j 和 $v_{1,2}$ 存入 S , 转步 5

步 7 构造点集 S 中所有 3 度节点的凸壳 CH_S ^[11], 并计算 CH_S 的最大内切圆的圆心 o ^[12] (若 S 中仅有一个 3 度节点, 则将其作为点 o , 若 S 中仅有两个 3 度节点, 则将二者连线中点作为点 o);

步 8 将 $MediaAxis$ 中不属于 S 且与 S 内 3 度节点相连的所有节点连接到 o , 从 $MediaAxis$ 中删除 S 内的所有节点, 同时调整 $MediaAxis$ 中各节点的相应属性, 转步 2

步 9 返回 $MediaAxis$

3.3 由中轴线生成多边形道路网络的拓扑结构^[1,2]

完成 3.2 后便得到了经过优化处理的多边形道路网络中轴线, 要由中轴线进一步生成多边形道路网络的拓扑结构, 首先需要定义多边形道路网络的拓扑结构, 本文采用现有的 Arc-Node 结构^[2], 具体的生成算法如下:

算法 7 由中轴线生成多边形道路网络拓扑

输入: 中轴线 $MediaAxis$

输出: 多边形道路网络拓扑 $Topology$ $Polynomial$ Net

步 1 清空 Topo bgy_ PolyRoadNet 标记 MediaAxis 中所有节点为未读;

步 2 若 MediaAxis 中不存在未读的非 2 度节点, 转步 7

步 3 取任意未读的非 2 度节点 v_i , 标记 v_i 为已读, 将 v_i 加入 Topo bgy_ PolyRoadNet

步 4 在 MediaAxis 中沿 v_i 发出的弧寻找节点, 若到遇到一个非 2 度节点 v_j , 则对该弧的搜索完成, 记沿 v_i 发出的弧搜索到的非 2 度结点集合为 S;

步 5 若 S 为空, 转步 2 否则, 取出 S 中的任意节点 v_j , 转步 6

步 6 将节点 v_j 及 v_i 到 v_i 的弧加入到 Topo bgy_ PolyRoadNet 中, 转步 5

步 7 返回 Topo bgy_ PolyRoadNet

4 关键数据结构定义

在生成多边形道路网络拓扑结构的过程中, 涉及到的几个关键数据结构定义如下 (注: 作者采用的是 Delphi 可视化程序设计语言)^[13]:

(1) 多边形区域顶点的双向循环链表 NodeLinkList 及其结点类型

```

Ptr vertex_node = ^vertex_node / 定义指向结构
NodeLinkList 的指针类型
vertex_node = record // 多边形顶点链表的结点类型
    Index integer // 顶点序号
    mark integer / 凹凸顶点标志, -1 是凹点, 1 是凸点
    weight single / 相关三角形的权值
    last Ptr vertex_node / 指向前一顶点的指针
    next Ptr vertex_node / 指向后一顶点的指针
end
NodeLinkList = Ptr vertex_node
    
```

(2) 三角形链表 TriAngleLinkList 及其结点类型

```

Ptr tri_node = ^tri_node / 定义指向结构 NodeLinkList
的指针类型
tri_node = record
    vt array[1..3] of integer / 存放三角形的三个顶点序号
    next Ptr tri_node / 指向下一个结点的指针
end
TriAngleLinkList = Ptr tri_node
    
```

(3) 三角形的边链表 EdgeLinkList 及其结点类型

```

Ptr EL = ^EL; / 定义指向结构 EdgeLinkList 的指针
EL = record
    FromNodeIndex integer / 边的起点
    ToNodeIndex integer / 边的终点
    IsBound Boolean / 表明该边是否边界边
    ItsTriangle array[1..2] of integer / 边所属三角形的
索引
end
    
```

```

EdgeLinkList = Ptr EL;
    
```

(4) 多边形道路区域的中轴线链表 MediaAxis 及其结点类型

```

Ptr MAPoint = MAPoint / 定义指向结构 MAPoint
的指针
MAPoint = record
    Index integer / 结点索引号
    X, Y: integer / 结点坐标
    Degree integer / 结点数度
    AdjacentPoints array[1..3] of integer / 该节点的
相邻结点
end
MediaAxis = Ptr MAPoint
    
```

5 实例与结果分析

为了验证本文所提出的多边形道路网络拓扑生成方法的正确性和有效性, 结合嵌入式车载导航系统的实际研发, 作者以海淀区多边形道路网络中的部分道路网络为例进行实验. 图 2 显示的是经过预处理的多边形道路网络, 图 3 显示的是其 Delaunay 三角剖分结果, 图 4 显示的是其优化前的中轴线, 图 5 显示的是其优化后的中轴线, 图 6 显示的是其拓扑结构. 其中, 图 2 至图 5 中的黑色线表示与多边形道路网络对应的简单多边形区域边界, 图 3 至图 5 中的红色线表示简单多边形区域 Delaunay 三角剖分所划分出的三角形边, 图 4 中的蓝色线表示优化前的中轴线, 图 5 中的绿色线表示优化后的中轴线, 图 6 中的绿色线表示多边形道路网络的拓扑结构.

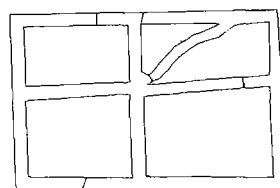


图 2 预处理后的多边形道路网络

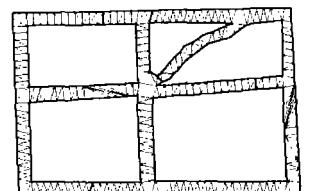


图 3 多边形道路网络 Delaunay 三角剖分结果

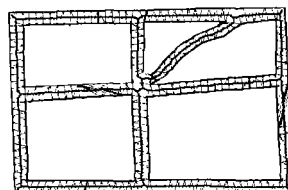


图 4 优化前的中轴线

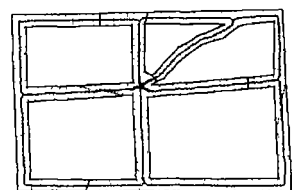


图 5 优化后的中轴线

对比图 2 和图 3 可以看出, Delaunay 三角剖分将对道路多边形的研究转化成了对所划分出的三角形的研究, 从图 4 可以看出优化前的中轴线通常存在伪分支及交叉口处出现多个三叉路口的问题, 从图 5 可以看出, 经过优化的中轴线不仅去除了长度过短的伪分支, 而且将多个三叉

路口合并成了一个多叉路口,使交叉路口的表示更加符合实际,从图6则可以看出,最终所获得的多边形道路网络拓扑结构,确实较准确地反映了多边形道路网络的拓扑连接关系,且与理论分析基本一致,说明算法是正确和有效的。

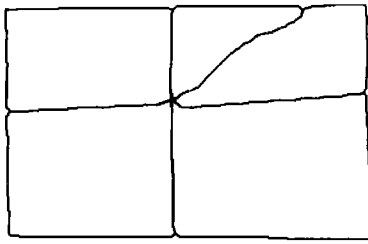


图6 多边形道路网络的拓扑

6 结束语

本文提出的多边形道路网络拓扑生成方法,弥补了现有折线道路网络生成方法不能解决多边形道路网络拓扑生成的不足,使目前许多实际可得的多边形道路网络数据更易于应用到车辆定位导航系统中,为车辆定位导航系统的进一步发展奠定了基础。此外,文中所涉及到的简单多边形区域 Delaunay 三角剖分及矢量数据的压缩方法等,也可以推广应用到其他领域。

参考文献:

- [1] Zhao Yilin Vehicle Location and Navigation Systems[M]. Boston: Artech House, 1997.
- [2] 张小国,王庆,王宁.电子地图路网模型及其自动生成算法研究[J].中国图像图形学报,2001,6(5):481-485
Zhang Xiaoguo, Wang Qing, Wang Ning. A Study on road network model in digital maps and the automatic generation algorithm of its database[J]. Journal of Image and Graphics, 2001, 6(5): 481-485 (in Chinese)
- [3] 丁永祥,夏巨谔,王英,等.任意多边形的Delaunay三角剖分[J].计算机学报,1994,17(4):270-275
Ding Yongxiang, Xia Jute, Wang Ying, et al. Delaunay triangulation of arbitrary polygons[J]. Chinese Journal of Computers, 1994, 17(4): 270-275 (in Chinese)
- [4] 徐春雷,李思昆.一种适用于任意平面多边形的三角剖分算法[J].国防科技大学学报,2000,20(2):82-85
Xu Chunlei, Li SiKun. A triangulation algorithm for general plane polygon[J]. Journal of National University of Defense Technology, 2000, 20(2): 82-85 (in Chinese)
- [5] 李岚.基于凹凸顶点判定的简单多边形区域的三角剖分[J].中国计量学院学报,2001,(12):164-166
Li Lan. The triangulation method of simple polygons based on judging the concave or convex[J]. Journal of China Institute of Metrology, 2001, (12): 164-166 (in Chinese)
- [6] 王树国,付宜力,吴菁,等.改进的统一于NIP的多边形三角剖分算法[J].计算机学报,1996,19(2):96-101
Wang Shuguo, Fu Yili, Wu Jing, et al. Improved Nip triangulation algorithm of arbitrary polygons[J]. Chinese Journal of Computers, 1996, 19(2): 96-101 (in Chinese)
- [7] 朱雅音,王化文,万丰,等.确定两个任意简单多边形交并差的算法[J].计算机研究与发展,2003,40(4):576-583
Zhu Ya Yin, Wang Hua Wen, Wan Feng, et al. An algorithm for determining the intersection, union and difference of any two polygons[J]. Journal of Computer Research and Development, 2003, 40(4): 576-583 (in Chinese)
- [8] 王志强,肖立瑾,洪家振.多边形简单性、方向及内外点的判别算法[J].计算机学报,1998,21(2):183-187.
Wang Zhiqiang, Xiao Lijin, Hong JiaZhen. Simplicity, Orientation, and inclusion test algorithms for polygons[J]. Chinese Journal of Computers, 1998, 21(2): 183-187. (in Chinese)
- [9] Ju Jiazou, Hung-hsin Chang, Hong Yan. A new skeletonization algorithm based on constrained delaunay triangulation [A]. Fifth International Symposium on Signal Processing and Its Applications[C]. Australia, 1999. 22-25.
- [10] 杨得志,王杰臣,阎国年.矢量数据压缩的Douglas-Peucker算法的实现与改进[J].测绘通报,2002,(7):18-22
Yang Dezhiz, Wang Jiechen, Liu Guonian. Study of realization method and improvement of douglas-peucker algorithm of vector data compressing[J]. Bulletin of Surveying and Mapping, 2002 (7): 18-22 (in Chinese)
- [11] 张大远,刘玉树.基于栅格划分构建平面点集凸壳的算法研究[J].微机发展,2004,14(7):106-108
Zhang DaYuan, Liu Yu-Shu. A grid-based algorithm for determining convex hull of a planar set[J]. Microcomputer Development, 2004, 14(7): 106-108 (in Chinese)
- [12] 郑梅生,陈宁,宋超.计算任意多边形最大内圆的一种算法[J].机械设计与制造,2003,(5):84-85
Zheng Mei-Sheng, Chen Ning, Song Chao. An algorithm for determining the largest internal circle in arbitrary polygons[J]. Machinery Design & Manufacture, 2003 (5): 84-85 (in Chinese)
- [13] 严蔚敏,吴伟民.数据结构[M].北京:清华大学出版社,1992

作者简介:

李杰 男,1976年出生于山西省岚县,2005年3月毕业于北京理工大学信息科学技术学院,获导航、制导与控制专业工学博士学位,目前主要研究兴趣为微系统集成理论与技术、惯性测控技术、组合导航理论、计算几何及智能信息处理等。E-mail: Lijie@nuc.edu.cn

张文栋 男,1962年出生于河南省太康县,教授、博士生导师。现任中北大学校长、“仪器科学与动态测试”教育部重点实验室常务主任,中北大学微米纳米技术中心主任。目前主要从事存储测试技术、MEMS技术、惯性测控技术及微系统集成理论与技术等方面的研究。

张樾 女,1977年出生于贵州省六盘水市,硕士研究生,现在中北大学自动化系导航、制导与控制专业攻读硕士学位,当前主要从事车辆定位与导航、组合导航理论与技术及计算几何等方面的研究工作。