第 16 卷 第 1 期 计算机辅助设计与图形学学报 Vol. 16, No. 1
2004 年 1 月 JOURNAL OF COMPUTER-AIDED DESIGN & COMPUTER GRAPHICS Jan., 2004

平面扩展多边形分层表示的构造方法及简单应用

严冬明 董未名 孙家广
(清华大学计算机科学与技术系 北京 100084)

摘 要 平面多边形的分层表示(L-REP)是一种基于三角形片的多边形表示模型,具有构造简单、鲁棒性强等优点,并且在许多问题上都有着很好的应用. 文中在这一工作的基础上进行扩展,使其可以应用到带圆锥曲线边的平面扩展多边形上,提出平面扩展多边形的分层表示方法(CL-REP),并给出了完整的数学模型和两种典型的构造算法. 最后给出了使用该方法的几个简单应用,主要是布尔运算和包容测试等,可见使用 CL-REP 能够简单、有效地解决这些问题.

关键词 扩展多边形;分层表示;从属关系
中图法分类号 TP391

Construction of Layer-Based Representation for Curved-Edge Polygons and Its Applications

Yan Dongming Dong Weiming Sun Jiaguang
(Department of Computer Science and Technology, Tsinghua University, Beijing 100084)

Abstract The layer representation (L-REP) is a triangle-based representation for polygons. It is simple and robust, and presents interesting properties for several applications. We extend this algorithm to curved-edge polygons and present CL-REP for Layer-based representation of curved-edge polygons. A formal mathematical model of it is defined and two representative algorithms to construct it are given. Several simple applications of this model are presented, with a special emphasis on the Boolean operations and point-in-polygon inclusion test, which can be easily and efficiently solved using the CL-REP.

Key words curved-edge polygons; layer-based representation; subordination

1 引 言

平面多边形的各种分解表示方法[1]在计算机几何造型领域中有着广泛的应用. 最近,Rueda 等提出了平面多边形的分层表示方法(Layer Representation, L-REP),并且给出了构造 L-ERP 的多种算法和它的一些应用[2],如布尔运算[3]和点的包容测试[4]等. 本文在文献[2,5]的基础上,给出了带圆锥曲线边的平面扩展多边形分层表示方法的定义和构造算法,以及以该模型为基础的两个具体应用.

2 扩展多边形的分层表示

首先给出本文主要研究的扩展多边形的一些定义.

原稿收到日期:2002-12-09;修改稿收到日期:2003-04-08. 本课题得到国家重点基础研究发展计划(2002CB312106)国家"八六三"高技术研究发展计划(2003AA421010)资助. 严冬明,男,1979 年生,硕士研究生,主要研究方向为 CAD &CG、数字图像处理. 董未名,男,1979 年生,硕士研究生,主要研究方向为 CAD &CG. 孙家广,男,1946 年生,教授,博士生导师,中国工程院院士,主要研究方向为 CAD &CG、产品数据管理与企业信息化.

© 1995-2004 Tsinghua Tongfang Optical Disc Co., Ltd. All rights reserved.

定义 1. 至少有一条边为圆锥曲线(包括圆弧、椭圆弧、双曲线段和抛物线段)的平面简单多边形称为扩展多边形. 由直线段构成的边称为扩展多边形的直线边,由曲线段构成的边称为扩展多边形的曲线边. 我们可以用如下方式表示一个扩展多边形

$$CP = \{t_1, t_2, \cdots, t_m\} \cup \{c_1, c_2, \cdots, c_n\}.$$

其中, t 表示直线边; c 表示曲线边, $c = (f, V_0, V_1, k)$, f 是曲线的方程, V_0 和 V_1 为两端点, k 的取值为 1 或 - 1,分别表示曲线位于向量 V_0V_1 的右侧或左侧[6]. 如图 1 所示, $\{c, d, e, f, g\} \cup \{a, b, h\}$ 即为一个扩展多边形.

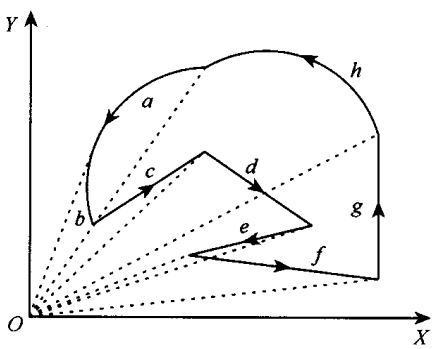


图 1 一个简单的平面扩展多边形

定义 2. 对位于笛卡尔坐标系第一象限的扩展多边形 CP,将其每一条直线边 t 的两个端点分别与坐标系原点 O 相连,构成一个闭合三角形,称为三角片,称 t 为三角片的非原点边,与原点相连的边称为三角片的原边. 将每一条曲线边 c 的两个端点分别与坐标系的原点相连,构成闭合扩展多边形,称为曲线片;同样地,称 c 为曲线片的非原点边,与原点相连的边称为曲线片的原边.

如图 1 所示,将扩展多边形各边的顶点与原点相连,便可以得到一系列三角片或曲线片.

有一种特殊情况需要考虑. 如图 2 a 所示,在构造曲线片 C_1 时,它的原边和非原点边之间产生了交点,这在计算从属关系时会发生问题,所以这种情况是不允许出现的. 因此,我们进行如下处理:

如果从原点引出的任意一条射线与曲线片的非原点边有多于一个的交点(包括顶点),则将曲线片分割成更小的片. 求出从原点出发的射线与曲线片原边的切点,以切点作为分割点将曲线段分割,将分割后的曲线段作为新的边代替原来的边加入到扩展多边形中. 这样,分割后将产生多个曲线片,这些曲线片的原边与非原点边不存在除顶点外的交点,且从原点引出的任意一条射线与曲线边都最多有一个交点,如图 2 b 所示. 在本文后面的论述中,如不作特殊说明,所研究的都是经过分割之后的曲线片.

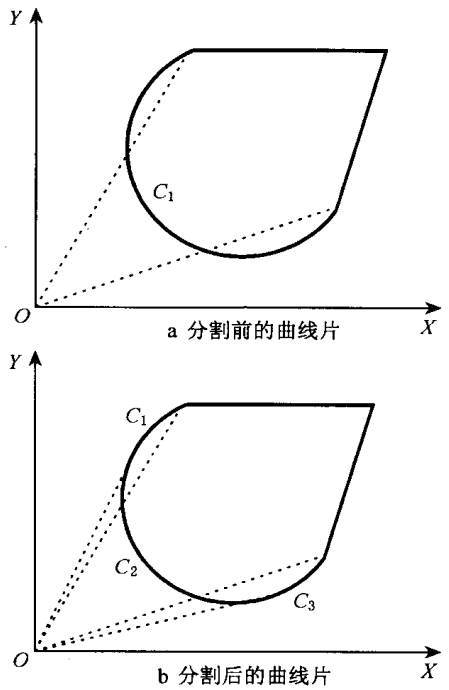


图 2 曲线片的非原点边与原边有交点时的处理方法

我们将三角片和曲线片的集合统一用集合 S 来表示, $S = \{S_1, S_2, \cdots S_{m+n}\} = \{T_1, T_2, \cdots, T_m\} \cup \{C_1, C_2, \cdots, C_n\}$,统称为简单片.

以任意点为坐标原点,扩展多边形的每一条边为非原点边,可以定义一组简单片,这些简单片叫作"原简单片". 因为它们都有一个顶点是坐标原点,所以每个原简单片都含有两条由原点和扩展多边形顶点决定的边(原边),并且有一条属于扩展多边形的非原点边.

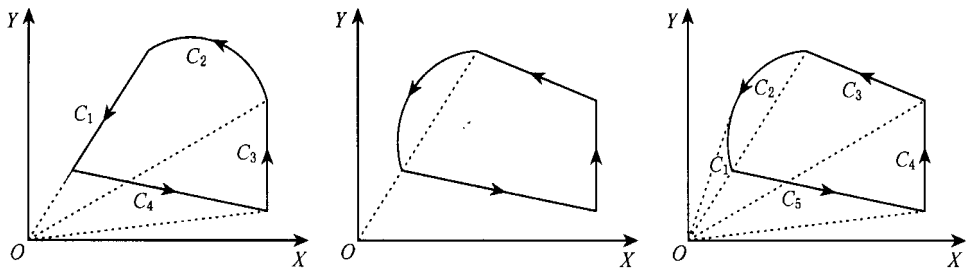
简单片顶点的排列方向决定了它们的符号,如果是逆时针排列则符号为 + 1,否则为 - 1. 若原点和三角片的另外两个顶点在同一直线上,则这个退化的三角片符号为 0,如图 3 a 所示. 如果原点和曲线片的另外两个顶点在同一直线上,则应该对这个曲线片进行分割,如图 3 b, 3 c 所示,所以曲线片的符号不会为 0. 给定简单片 $S = OP_1P_2$,其符号用 $\mathrm{sign}(S) = \begin{vmatrix} P_{1x} & P_{1y} \\ P_{2x} & P_{2y} \end{vmatrix}$ 计算.

定义 3. 给定简单片 S_i,定义其系数

$$\alpha_i = \mathrm{sign}([S_i]).$$

定义 4. 给定简单片 $S_1 = OQ_1Q_2$ 和 $S_2 = OQ_1'Q_2'$,我们称 S_1 从属于 S_2 或者 S_2 包含 S_1 ($S_1 \lhd S_2$) 当且仅当 $S_1 = S_2$,或在 S_1 的非原点边 Q_1 和 Q_2 上存在一点 Q, Q 在 S_2 的内部. 若非原点边 Q_1Q_2 全部在 S_2 的内部,称 S_1 完全从属于 S_2.

© 1995-2004 Tsinghua Tongfang Optical Disc Co., Ltd. All rights reserved.



a 三角片　　b 曲线片　　c 分割后的曲线片

图 3　三角片或曲线片的两个顶点和原点在同一直线上的情况

判断两三角片从属关系的方法见文献[6],加入曲线片后的方法基本相同,惟一的区别就是可能需要进行点和带曲线边的多边形的包容测试,相关算法见文献[4].

定义 5. 定义从属链 $S = \{S_0, S_1, \cdots, S_n\}$ 为一个简单片的集合,其中 $S_i \lhd S_{i+1}, 0 \leq i < n$.

定义 6. 给定简单片 T 和 S,如果我们可以找到一个从属链连接它们,$S = \{S_0, S_1, \cdots, S_n\}$,其中 $S_0 = T, S_n = S$,则称 $T \leq S$.

定义 7. 定义一个简单片 S_i 的序号 $\mathrm{ind}(S_i)$ 为在扩展多边形 CP 的简单片集合中从简单片 S_i 开始的最长从属链的长度.

简单片可以按照它们的序号存储在不同的层次中,这就是扩展多边形的分层表示法,下面给出它的定义.

定义 8. 定义扩展多边形 CP 的分层表示法 CL-REP 为一个包含相同序号的简单片的层次的集合,即 $CR = \{L_1, L_2, \cdots, L_l\}$,其中 $L_i = \{S_j : \mathrm{ind}(S_j) = i\}$ 包含所有序号为 i 的简单片 S_j,l 是扩展多边形 CP 的原简单片的最大层数.

从图 1 可以很清楚地看到各层所包含的简单片,其中的扩展多边形的分层表示如图 4 所示.

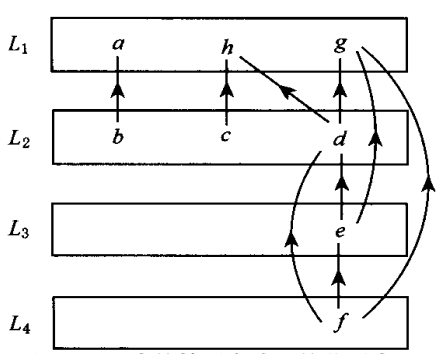


图 4　图 1 中的扩展多边形的分层表示

根据上面的定义,我们给出下面的引理.

引理 1. 同层的两个简单片的交为空.

完全由直线边构成的普通多边形的定理证明见文献[2]. 下面我们来看加入曲线边后的扩展多边形的情况.

证明. 用反证法. 设 S 和 T 为 L_i 层的两个简单片,若 S 和 T 的交不为空,那么 S 的非原点边必有一部分包含在 T 内(反之同理),即 $S \lhd T$. 如果 $T \in L_i$, $\mathrm{ind}(T) = i$,即 T 的最大从属链的长度为 i. 但此时 S 的最大从属链长度应该为 $i+1$,与 $S \in L_i$ 矛盾,引理得证.　　证毕.

由此可见,加入曲线边后的扩展多边形的证明方法与普通多边形完全一样. 事实上,我们可以将曲线边离散为若干长度很小的直线边,这时扩展多边形就转化为普通多边形,可直接使用原来的证明. 下面的引理 2 和定理 1 的证明方法也与扩展前的普通多边形的相应定理的证明方法完全一致. 由于篇幅有限,对后面的引理 2 和定理 1 不再给出详细证明,请参考文献[3,6].

引理 2. 给定层 L_i,如果 i 为奇数,则该层的简单片符号为正,反之为负.

全部为直线边的普通多边形的相关定理的证明见文献[2]. 如前所述,加入曲线边后的证明方法与加入前完全为直线边的证明方法一致,此处略. 由引理我们可以得到下面的定理.

定理 1. 设 CP 为一个扩展多边形,$\mathrm{CR} = \{L_1, L_2, \cdots, L_l\}$ 是它的 CL-REP 层,扩展多边形 CP 可以由 CR 中每一层中的简单片通过布尔运算来构造:

如果 l 为奇数,$CP = (\ldots(((L_1 - L_2) \cup L_3) - L_4) \cdots \cup L_l)$;

否则,$CP = (\ldots(((L_1 - L_2) \cup L_3) - L_4) \cdots - L_l)$.

加入曲线边后的证明方法与扩展前完全一致,此处略.

3　CL-REP 的构造和实现

扩展多边形的分层表示实现起来并不复杂,我们采用面向对象的方法. 定义了扩展多边形类 *CPolygonEx2D*,它包含一个三角片数组 *arrTriangle2D*

© 1995-2004 Tsinghua Tongfang Optical Disc Co., Ltd.　All rights reserved.

和一个环结构 *loop*;环包括内环和外环,环结构将多边形的所有顶点按照逆时针方向(外环,内环定点按照顺时针方向存放)存放在一个点数组 *arrVertext2D* 中. 我们仅给出基本的三角片和多边形的数据结构,大量的成员函数的定义及声明略.

(1) 多边形点的数据结构

```
struct Point2D
{
    CPoint p;
    bool IsVertex;  //标志该点是顶点还是交点
};
```

(2) 三角形片的数据结构

```
struct trian
{
    Point2D p1, p2;
        //三角形对应多边形边的两个顶点(start, end)
    int coef;  //三角形片的参数(1,0,-1)
    int level;  //三角形片的 level
    int num1;  //三角形片的标号
};
```

(3) 多边形环的数据结构

```
struct loop
{
    CArray<Point2D, Point2D&> arrVertext2D;  //存储环上各顶点
    bool IsHole;  //标志环是否为洞
};
```

CSimpleChipNode 为图的节点类,存放每个节点的层数和它的父、子节点,以及对应的扩展多边形的非原点边.

构造扩展多边形的分层表示有许多方法,如基于图矩阵的方法[2];另外,文献[4]中提出的一些方法在这里也适用. 本文提出两个更加简单的算法.

3.1 深度优先算法

首先定义一个有向图 G,它的节点是三角片 $S_0 S_1 S_2 \cdots S_n$,如果 $S_i \lhd S_{i+1}$,则在节点 i 和 j 之间加入一条有向边,如图 5 所示为图 1 中简单多边形对应的有向图. 每个节点中记录的从属于该节点的那些节点称为其子节点,该节点称为它们的父节点.

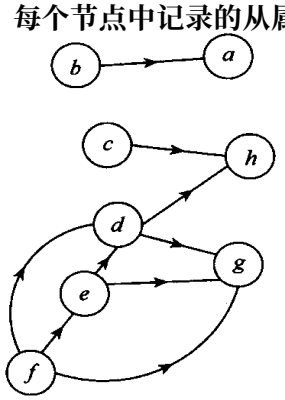


图 5 简单片对应的有向图

简单片 S_i 的序号定义为从 S_i 出发到某一节点 S_j 的最长的从属链的长度,等价于找一条从节点 S_i 出发到另一节点的最长路径. 我们采用深度优先算法,对所有节点使用深度优先搜索,找出每个节点到层数为 1 节点的最大距离即为它们的层数. 所有出度为 0 的节点,即没有父节点的节点,它们的层数为 1. 由于有向图的深度优先的算法复杂度为 $O(\max\{|V|,|E|\})$[7],其中,$|V|$ 为图的顶点数,$|E|$ 为图的边数,因此,该算法的平均复杂度不会超过 $O(\max\{n,|E|\})$,$|E|$ 为所有的从属关系的数目. 该算法的形式描述如下:

算法 1.

```
void Construct(CPolygonEx2D CP)
{
    cp.Split();  //对扩展多边形进行分割;
    Graph G;  //有向图
    CreateGraph(CP, G)  //按照简单片之间的从属关系生成图 G;
    for(i=0; i<G.GetNodeNum(); i++)
        //对 G 的每个节点用深度优先计算层数
    {
        DFS(G.GetNode(i));
    }
}
```

生成图的算法如下:

```
CreateGraph(CPolygonEx2D CP, Graph G)
{
    for(int i=0; i<CP.GetEdgeNum(); i++)
    {
        CSimpleChipNode si(CP.GetEdge(i));  //图的节点
        G.AddNode(si);  //生成有向图
        for(j=0; j<i<CP.GetEdgeNum(); j++)
        {
            CSimpleChipNode sj(CD.GetEdge(j));
            if(si.subordinates(sj))
            {
                si.AddFatherNode(sj);  //保存父节点
                sj.AddChildNode(si);  //保存子节点
            }
            else if(sj.subordinates(si))
            {
                sj.AddFatherNode(si);
                si.AddChildNode(sj);
            }
        }
    }
}
```

深度优先算法的描述如下:

© 1995-2004 Tsinghua Tongfang Optical Disc Co., Ltd. All rights reserved.

```
void DFS(CSimpleChipNode node)
{
    //如果没有子节点,则层数为 1
    if(node.GetChildNum() == 0)
      {
          node.level = 1;
          return;
      }
    //对 node 的每个子节点进行递归的搜索
    for(i = 0; i < node.GetChildNum(); i++)
      {
          DFS(node.GetChildNode(i);
      }
    //该节点的层数即为子节点的最大层数加 1
    node.level = max(node.GhildNode.level) + 1;
}
```

3.2　动态算法

利用图的特性,我们首先找到出度为 0 的节点,这些节点肯定是属于第一层的. 从图中删除这些节点以及指向它们的有向边,余下的出度为 0 的节点肯定属于第二层,依次类推,即可确定每个节点的层次. 该算法在本质上和算法 2 是相同的,所以的复杂度也相同. 算法 2 的形式描述如下:

算法 2.

```
void Construct(CPolygonEx2D CP)
{
      CP.Split();  //对扩展多边形进行分割
      Graph G;  //有向图
      CreateGraph(CP, G)  //按照简单片之间的从属
         关系生成图 G,同算法 1
      level = 1;
      while(!G.IsEmpty())
        {
            CSimpleChipNode arrChip[];
            FindSingleNodes(G, arrChip)  //找到 G 中
              出度为 0 的节点,存放在 arrChip 中
              //层数设为 level
            SetModifyFlag(G, arrChip)  //将上述节点
              作标记,从 G 中删除指向它们的边
            level++;
        }
}
```

4　一些应用

对扩展多边形使用分层表示之后,可以方便地求出扩展多边形的面积,只要用处于奇数层的简单片的面积之和减去处于偶数层的简单片的面积之和即可. 另外,我们认为利用这一模型可能可以计算扩展多边形的其他一些几何或物理特性,如质心和惯性距等,但实现这些功能还需要进行更深入的研究.

下面我们着重提出两个应用:平面扩展多边形的布尔运算和点的包容测试.

4.1　布尔运算

文献[5]的工作主要实现了平面扩展多边形的布尔运算,该算法根据文献[8]提出的平面简单多边形布尔运算算法扩展而成,相对于其他一些算法[9-10]比较新颖,并具有更强的鲁棒性,可以对平面扩展多边形进行并、交、差的运算,且无需考虑各种各样复杂的特殊情况. 具体模型和算法见文献[5]. 附图 1 所示为一个布尔运算的例子.

4.2　二维扩展多边形点的包容测试

文献[4]提出了一个带曲线边的扩展多边形的点的包容测试算法. 该算法与原有的其他一些算法[11-12]相比,具有简单、鲁棒和适应性强的特点,而它的基础也是本文所描述的扩展多边形的分层表示,下面进行简单的描述. 与文献[1]中给出的普通多边形包容测试的算法类似,我们定义扩展多边形的边界表示为 $CP = \{e_1, e_2, \cdots, e_n\}$,其中 e_i 为扩展多边形的第 i 条边;定义 Q 为任意一点,定义

$$k_i = \begin{cases} 1, & \text{if } \operatorname{sign}(t_i) > 0 \text{ and } Q \in t_i \\ -1, & \text{if } \operatorname{sign}(t_i) < 0 \text{ and } Q \in t_i, \\ 0, & \text{otherwise} \end{cases}$$

其中,$\operatorname{sign}(t_i)$ 是简单片的符号,由它所对应的扩展多边形的边的方向决定,$Q \in CP$ 当且仅当 $\sum k_i > 0$. 因此,点的包容测试可以很方便地由点在简单片中的测试结果得到:给定一个任意点 Q 和原简单片 $T = \{OAB\}$,如果 $\operatorname{sign}(QOA) \geq 0$,$\operatorname{sign}(QAB) \geq 0$ 和 $\operatorname{sign}(QBO) \geq 0$,则 Q 在 T 中. 算法与文献[3]类似,此处略.

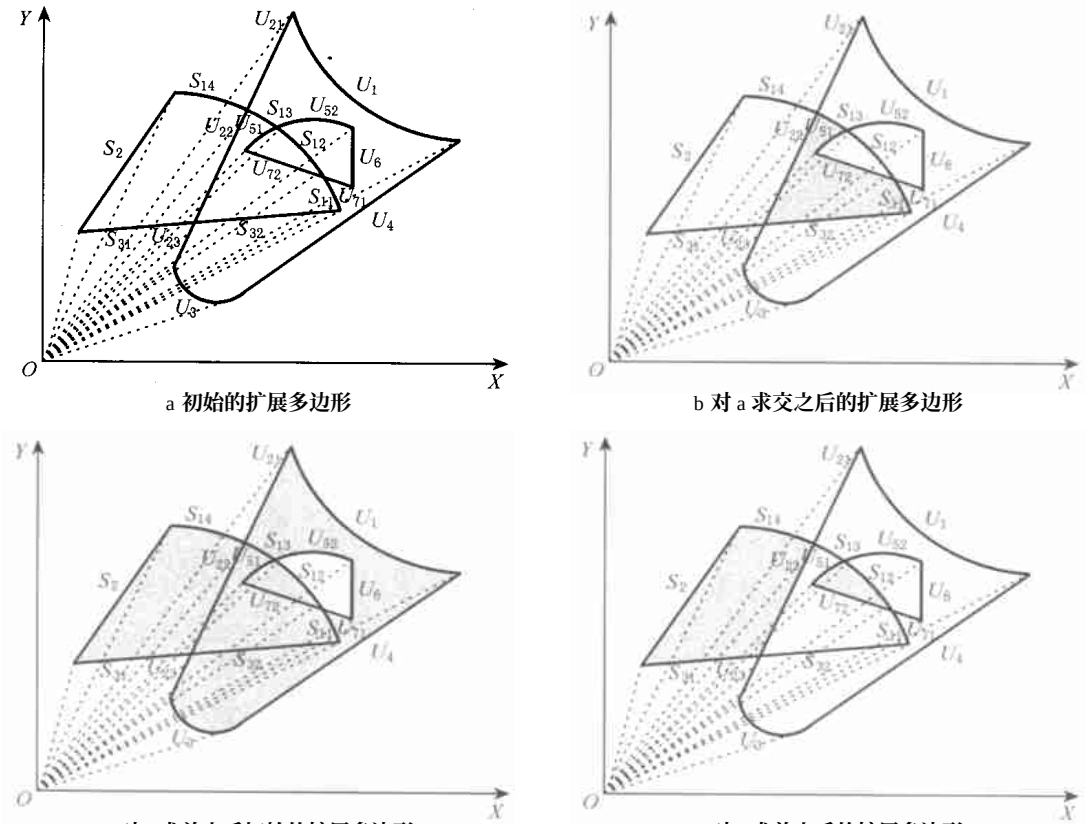
5　结论和下一步工作

本文提出了平面简单扩展多边形的分层表示方法,并给出了一些应用. 本文算法基于一些简单的概念,如三角片、曲线片、从属关系、从属链和层等,通俗易懂;同时,实现该模型所需要的数据结构也比较简单,效率比较高. 在应用上,CL-REP 在布尔运算和包容测试的计算中都有着很高的效率和稳定性,即使在扩展多边形的边很多的情况下也能很好的工作.

本文所研究的扩展多边形在凸性上没有任何限制,并允许内环的存在. 但是,由于过于复杂的缘故,没有考虑边自交的情况,这方面目前我们正在做

© 1995-2004 Tsinghua Tongfang Optical Disc Co., Ltd.　All rights reserved.

进一步的研究;同时,还在对包容测试算法进行改进,争取将其时间复杂度减为 $O(\log l)$,其中 l 为 CL-REP 的最大层数. 另外,我们准备将该模型的思想扩展到三维空间,以使其有更广的应用范围.



a 初始的扩展多边形

b 对 a 求交之后的扩展多边形

c 对 a 求并之后初始的扩展多边形

d 对 a 求差之后的扩展多边形

附图 1 布尔运算举例

参 考 文 献

[1] Zhou Peide. Computational Geometry —The Analysis of the Algorithms and Design [M]. Beijing: Tsinghua University Press, 2000 (in Chinese)
(周培德. 计算几何 ——算法分析与设计[M]. 北京: 清华大学出版社, 2000)

[2] Rueda A J, Feito F R, Rivero M. A triangle-based representation for polygons and its applications [J]. Computers & Graphics, 2002, 26(5): 805～814

[3] Rivero M, Feito F R. Boolean operations on general planar polygons [J]. Computer & Graphics, 2000, 24(6): 881～896

[4] J Ruiz, de Miras, Feito F R. Inclusion test for curved-edged polygons [J]. Computer & Graphics, 1997, 21(6): 815～824

[5] Dong Weiming, Maira Babang, Zhou Dengwen, *et al*. Boolean operations on general planar curved-edge polygons [J]. Journal of Computer-Aided Design & Computer Graphics, 2003, 15(9): 1134～1140 (in Chinese)
(董未名, 玛依拉·巴榜, 周登文, 等. 平面简单扩展多边形的布尔运算[J]. 计算机辅助设计与图形学学报, 2003, 15(9): 1134～1140)

[6] Feito F, Rivero M L. Geometric modeling based on simplicial chains [J]. Computers & Graphics, 1998, 22(5): 611～619

[7] Lu Kaicheng. Introduction to Computer Algorithms —Design and Analysis [M]. Beijing: Tsinghua University Press, 2001 (in Chinese)
卢开澄. 计算机算法引导 ——设计与分析[M]. 北京: 清华大学出版社, 2001

[8] Feito F, Rivero M, Rueda A. Boolean representation for general planar polygons [A]. In: Proceedings of the 7th International Conference in Central Europe on Computer Graphics, Visualization and Interactive Digital Media, Pilsen, 1999. 87～92

[9] Wu Yunxing. Polygon Boolean operation based on edges recognition algorithm [J]. Journal of Computer-Aided Design & Computer Graphics, 1994, 6(4): 260～265 (in Chinese)
(武运兴. 基于边界识别的多边形的布尔运算[J]. 计算机辅助设计与图形学学报, 1994, 6(4): 260～265)

[10] Zhu Renzhi, Meng Weihua. A new algorithm on one-dimension Boolean operation in terms of geometric modeling [J]. Journal of Engineering Graphics, 1994 (1): 10～14 (in Chinese)
(朱仁芝, 孟卫华. 一种用于几何造型的一维布尔运算新算法[J]. 工程图学学报, 1994 (1): 10～14)

[11] Feito F, Torres J C, Urena A. Orientation, simplicity and inclusion test for planar polygons [J]. Computer & Graphics, 1995, 19(4): 595～600

[12] Haines E. Point in Polygons Strategies [M]. In: Paul Heckbert, ed. Graphics Gems IV. New York: Academic Press, 1994. 24～25

© 1995-2004 Tsinghua Tongfang Optical Disc Co., Ltd. All rights reserved.