

## 基于边/面遮挡关联性的多面体凸剖分方法\*

李 静<sup>1+</sup>, 王文成<sup>1</sup>, 吴恩华<sup>1,2</sup>

<sup>1</sup>(中国科学院软件研究所 计算机科学国家重点实验室,北京 100190)

<sup>2</sup>(澳门大学 科技学院 计算机与信息科学系,澳门)

### Convex Decomposition of Polyhedrons Using Occlusion Relations Among Edges/Facets

LI Jing<sup>1+</sup>, WANG Wen-Cheng<sup>1</sup>, WU En-Hua<sup>1,2</sup>

<sup>1</sup>(State Key Laboratory of Computer Science, Institute of Software, The Chinese Academy of Sciences, Beijing 100190, China)

<sup>2</sup>(Department of Computer and Information Science, University of Macau, Macao, China)

+ Corresponding author: E-mail: lij@ios.ac.cn

**Li J, Wang WC, Wu EH. Convex decomposition of polyhedrons using occlusion relations among edges/facets.**

*Journal of Software*, 2008,19(7):1766–1782. <http://www.jos.org.cn/1000-9825/19/1766.htm>

**Abstract:** This paper presents a new method for convex decomposition of polyhedrons. In comparison with existing methods, the new method can improve the efficiency greatly with complexities reduced in many aspects of execution time, storage and added new vertices, and it is more advantageous in treating the polyhedrons with reflex edges in higher numbers. Its strategy is to gradually decompose a polyhedron in local operations by the occlusion relationships between facets and edges along certain orthogonal directions. In treating general polyhedrons in practice, the new method has its time and storage complexities both in  $O(n)$  approximately, and its produced new vertices are in a number not more than  $O(r+n^{0.5})$ , here  $n$  is the vertex number and  $r$  is the reflex edge number. By testing a large number of complex polyhedrons, it shows that, compared with the popularly used “cutting & splitting” method, this new method can run 14~120 times faster, reduce the storage requirement to 1/2.3~1/7.4, and reduce the new points to at most 1/28, and even needs no new point in some cases. Because most convex polyhedrons decomposed by the method are tetrahedrons, the resulted convex polyhedrons by the new method are more than those by the “cutting & splitting” method. However, if convex polyhedrons are required to be further decomposed to tetrahedrons, the new method can produce much fewer tetrahedrons, due to much fewer added vertices for decomposition by the new method. Besides, the new method can be conveniently used to treat the polyhedrons with holes, or even the non-manifold polyhedrons that contain isolated facets, edges or vertices.

**Key words:** occlusion; convex decomposition; polyhedron; tetrahedron; polygon

**摘要:** 提出一种多面体凸剖分的方法,与国际上已有的工作相比,在计算速度、空间需求和新增顶点等方面均降低了复杂度,有大幅的效率提高,且在处理凹边很多的多面体时具有更大的优越性.其工作步骤是根据多面体的面、

\* Supported by the National Natural Science Foundation of China under Grant No.60773026 (国家自然科学基金); the National High-Tech Research and Development Plan of China under Grant No.2006AA01Z306 (国家高技术研究发展计划(863)); the National Basic Research Program of China under Grant No.2002CB312102 (国家重点基础研究发展计划(973))

Received 2007-10-09; Accepted 2007-12-28

边沿某些方向正投影时面与面之间、边与边之间的遮挡关系进行局部化操作,以渐进地凸剖分多面体.它对应用中的常见模型表现出的时间复杂度、空间复杂度皆近似为 $O(n)$ ,而新点数不超过 $O(r+n^{0.5})$ ,这里, $n$ 为模型的点数, $r$ 为凹边数.实验结果表明,与目前国际上常用的“切割分裂”方法相比,新方法的速度提高了14~120倍,空间下降至“切割分裂”方法的1/2.3~1/7.4,而新增加的点数则最多为“切割分裂”方法的1/28,甚至有些情况下无须增加新点就能完成凸剖分.新方法剖分出的凸多面体绝大多数是四面体,多于“切割分裂”方法所得凸多面体数量.但是,很多应用是要求多面体被剖分为四面体的.如果进一步将凸多面体四面体化,则新方法的结果个数将明显少于“切割分裂”方法,因为新方法的剖分过程中所增加的新点要少很多.新方法还能方便地处理包含空洞的多面体,甚至是包含孤立面、孤立边和孤立点的非流形多面体.

关键词: 遮挡;凸剖分;多面体;四面体;多边形

中图法分类号: TP391 文献标识码: A

## 1 简介

很多领域的工作都要进行关于多面体的计算,如碰撞检测<sup>[1]</sup>、CSG(constructive solid geometry)建模<sup>[2]</sup>、特征提取<sup>[3]</sup>、网格生成<sup>[4]</sup>等.由于凸多面体的一些优越性质,将任意的多面体剖分成一些凸多面体再对这些凸多面体进行相关的计算,可以极大地提高关于多面体计算的效率.因此,多面体的凸剖分得到了广泛的关注.多面体的凸剖分就是将一个复杂的多面体分割成一些相互间没有重叠的凸多面体,即这些凸多面体只有公共的顶点、边和多边形面,并且它们的并集就是原来的多面体.

关于多面体的凸剖分,已有了很多的研究.文献[5]中指出,将一个多面体凸剖分成个数最少的凸多面体是一个NP-难问题.因此,人们努力研究更有效的凸剖分方法,而不是去寻求最佳剖分的方法.由于多面体的非凸特征是由凹边引起的(如果一条边相关的两个面在多面体内的夹角大于 $180^\circ$ ,则该边是一条凹边),目前的各种凸剖分方法基本上是按照Chazelle的“切割分裂”策略进行的<sup>[6,7]</sup>,即根据凹边形成一些切割面对多面体进行分裂操作,以消除各个凹边的非凸性质.但这些方法会生成大量新点,在处理凹边较多的复杂多面体时会使时空复杂度增长很快.

实际应用中,往往会遇到含有很多凹边的复杂多面体.为了提高这些复杂多面体的凸剖分效率,提出一种新的凸剖分方法,以尽可能地进行局部操作来减少时空开销,降低复杂度.新方法利用多面体被观察时其各个元素(点、边、面)之间的遮挡关系将多面体分割成一些单层多面体(如果沿着某个方向正投影地看一个多面体,任一视线与该多面体的交只是一条线段、一个点或者没有交,则该多面体称为相对于该方向的一个单层多面体),然后再对各个单层多面体进行凸剖分.对于应用中的常用模型,新方法的时间复杂度、空间复杂度、剖分结果凸多面体个数皆近似为 $O(n)$ ,其所新增加顶点数的复杂度近似为 $O(r+n^{1/2})$ ,这里, $n$ 为多面体的点数, $r$ 为多面体凹边个数.由于该方法的复杂度对凹边数目不是很敏感,因此它在处理凹边较多的复杂多面体时具有很好的效率.特别是,新方法能够很方便地处理包含空洞的多面体、甚至是包含孤立面、孤立边和孤立点的非流形多面体.

本文第2节介绍已有的一些高效多面体凸剖分方法.第3节讨论二维情况下将多边形凸剖分成凸多边形的方法(虽然该方法比已有的一些高效多边形凸剖分方法具有更高的时间复杂度,但它能较直观地讨论基于边与边之间的遮挡关系进行凸剖分的思路,以便为讨论多面体的凸剖分新方法做一些准备).第4节详细讨论多面体的凸剖分新方法.第5节进行实验结果的分析.最后进行总结.

## 2 相关工作

与二维情况下多边形的凸剖分相比,三维多面体凸剖分的方法不是很多,它们基本上是按照“切割分裂”的思路来处理凹边的.该思路最早由Chazelle<sup>[6]</sup>提出,即:多面体的非凸性质是由凹边反映出来的;于是,为一条凹边生成一个切割面分裂多面体,则该边在各个分裂的多面体中将都不是凹边;这样,当所有凹边都被消除,就完成了多面体的凸剖分.

这种简单、直观的方法可能会产生很多凸多面体.为了减少剖分出的凸多面体个数,并降低凸剖分的复杂度,Chazelle提出了改进的方法<sup>[6,7]</sup>,即所有原属于同一条凹边但被其他分割面分开的凹边对应的分割面必须共面.这样,每条凹边的切割面最多会与 $r-1$ 个其他凹边的切割面相交,所以剖分出的凸多面体数最多为 $O(r^2)$ .Chazelle还给出了剖分凸多面体个数的下限,即存在一类多面体,若该多面体的凹边数为 $r$ ,则剖分该多面体生成的凸多面体个数总不少于 $O(r^2)$ .该方法的最坏时间和空间复杂度分别为 $O(nr^2(r+\log n))$ 和 $O(nr^2)$ .经过一些复杂的预处理,时间复杂度可降至 $O(nr^3)$ .

文献[8]提出了将非流形多面体转化为流形再进行凸剖分的方法,将文献[6]的方法推广到能够处理包含任意亏格genus(handle)和shell(internal void)的流形多面体.它同时对某些子算法进行了优化,使得其最大时间和空间复杂度分别为 $O(nr^2+r^{7/2})$ 和 $O(nr+r^{5/2})$ ,并生成 $O(r^2)$ 个凸多面体.

Hershberger和Snoeyink<sup>[2]</sup>认为切割分裂凸剖分方法<sup>[6]</sup>的时间最坏复杂度上限应为 $O(nr+r^{7/3})$ ,并且给出了一种时空复杂度均为 $O(nr+r^3)$ 的切割分裂凸剖分算法.

2003年,Lien和Amato提出对多面体进行近似凸的凸剖分方法<sup>[9]</sup>,即对凹边进行重要性的度量和排序,并渐次处理重要的凹边直至误差精度满足要求,而那些不重要的凹边可以保留在剖分的结果中.与精确凸剖分相比,这种近似的凸剖分可以减少剖分的多面体数量,提高剖分的效率.其具体的凸剖分方法依然是根据“切割分裂”的思路,并采用文献[8]的方法.该方法能够很好地表达模型的重要结构特征,并能生成多面体的细节层次表示.

多面体的四面体化问题是与多面体凸剖分密切相关的一个研究课题.根据文献[10]中的论述,Chazelle等人在文献[11]中给出的将多面体剖分成四面体的方法是非常高效的.它的时间复杂度为 $O(nr+r^2\log r)$ ,产生的四面体个数为 $O(nr+r^2)$ .但它仅能处理与球同构的多面体,不能处理包含内部空洞的情况.1991年,Dey<sup>[12]</sup>给出了对带空洞多面体四面体化的方法.该方法先以“切割分裂”方法凸剖分多面体,然后再对凸多面体进行四面体化.其时间复杂度为 $O(nr^2+r^3\log r)$ ,空间复杂度为 $O(nr+r^3)$ .1992年,Ruppert和Seidel<sup>[13]</sup>证明了确定一个多面体是否能够被四面体化(不加点)是一个NP完全问题.近期的研究多集中在对凸多面体的四面体化问题上<sup>[14-16]</sup>.但这些问题的前提是输入为凸多面体,实际上,这正是多面体凸剖分算法所要解决的.而本文提出的针对复杂多面体的高效凸剖分算法可以很好地满足这一需求.

### 3 二维平面上多边形的凸剖分

在讨论基于边/面遮挡关系的多面体凸剖分算法之前,我们先讨论基于边之间遮挡关联性的多边形凸剖分算法,以便下节简洁、明了地讨论三维多面体凸剖分的新方法,因为它们遵循相同的思路.虽然多边形的凸剖分问题已有较全面的研究<sup>[10]</sup>,并且一些高效的算法比本文论述的多边形凸剖分算法具有更好的复杂度,但遵循相同思路的多面体凸剖分算法比已有的凸剖分算法具有更高的效率.

如果一个多边形中所有的内部夹角均不大于 $180^\circ$ ,则它是一个凸多边形;否则,它就是凹多边形.凹多边形中大于 $180^\circ$ 的夹角所在的顶点称为凹点.多边形的凸剖分,就是将一个非凸的多边形分割成一些相互之间不交叉的凸多边形,且这些凸多边形的并集就是原来的非凸多边形.

这里,我们给出一种基于边之间遮挡关系的多边形凸剖分方法.由于二维凸剖分并非本文讨论的重点,本节介绍的方法只是为了引入一些概念、分析,以便为后续三维情况的讨论打下基础,因此本文对此多边形凸剖分方法不进行算法复杂度的分析.

不失一般性,在下面的讨论中,假设平面的笛卡尔坐标系的 $X$ 轴是由左至右, $Y$ 轴是由下向上.

#### 3.1 多边形的凸剖分方法

在讨论多边形的凸剖分之前,我们先定义几个概念.

- $Y$ 单调边序列:对于各条边依次连接的一个边的序列,如果其各个顶点的 $Y$ 坐标值沿着其排列序列是单调变化的,则这个边序列称为 $Y$ 单调边序列,下文简称单调边序列.
- $Y$ 单调多边形:如果一个多边形基于它的 $y$ 坐标值最大和最小的两个顶点可以将它的边分成两个单调边序列,则该多边形称为 $Y$ 单调多边形,下文简称单调多边形,如图1所示.

- 遮挡关系:沿着某个方向观察两个目标  $a$  和  $b$ (目标可以是点、边),如果有一条视线与它们都有交点,且与  $a$  的交点比与  $b$  的交点离视点更近,则  $a$  能完全可见,而  $b$  至少有一部分被  $a$  挡住而不可见.此时,称  $a$  遮挡了  $b$ .

为了提高剖分的效率,我们将尽可能地进行局部操作,并生成尽可能少的新边.不失一般性,本文假设沿着平行  $X$  轴方向由左向右正投影地观察多边形以形成边与边之间的遮挡关系.我们的主要思路是:根据边与边之间的遮挡关系,先将多边形剖分成一些  $Y$  单调多边形,然后再利用文献[10]中提出的方法对非凸的各个单调多边形进行三角化或用类似的方法对其进行凸剖分.

将一个多边形剖分成一些单调多边形的步骤如下:

- (1) 先找到多边形中最左边的一个顶点(即  $x$  坐标最小的顶点).
- (2) 根据所找到的顶点,迭代搜索相邻的边,以得到一个局部最长的单调边序列(称为入射边单元).
- (3) 找到只被入射边单元完全遮挡的边,即遮挡这些边的边一定是在入射边单元中.
- 如果这些边彼此相连,则它们必然为一个单调边序列,称为出射边单元.该出射边单元可与那个入射边单元一起生成一个单调多边形.
- 如果这些边不完全相连,就在它们间断的每一个地方生成一条位于多边形内部的新边(在第 3.1.2 节讨论),将它们连接成一个单调边序列,再与入射边单元连接成一个单调多边形.
- (4) 每生成一个单调多边形,就从原多边形中分离出去;而为生成这个单调多边形所生成的新边,将被复制到分离后的多边形中,形成分离后的多边形.
- (5) 以上步骤迭代进行,就能将多边形剖分成一些单调多边形.

如图 2 所示,它的第 1 层入射边包括  $\{AB,BC,CD,DE\}$ ,它们构成了一个入射边单元.在第 2 层的边中,只被该入射边单元遮挡的边为  $\{EF,LM,MA\}$ .将它们顺序排列并将中断的地方加以连接,即连接点  $L$  与  $F$  生成新边后,得到一个与入射边单元对应的出射边单元  $\{EF,FL,LM,MA\}$ .此时出、入射边单元共同构成一个单调多边形,即图 2(a)中的斑马条纹区域.将它分离后的情形如图 2(b)所示,这时进行下一次剖分的入射边单元将为  $\{KL,LF,FG\}$ .如此迭代地进行,就能将该多边形剖分成一些单调多边形,如图 2(c)所示.

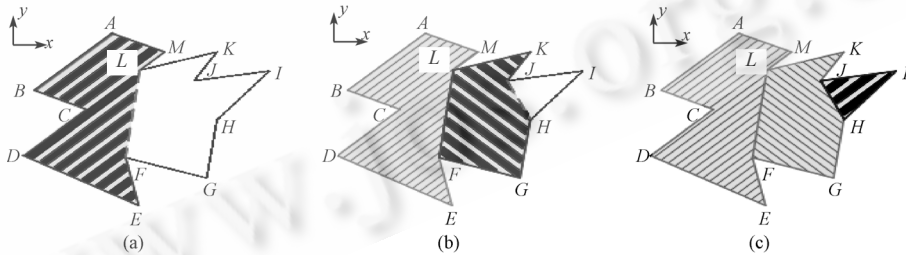


Fig.2 Decompose a polygon into monotone polygons gradually using occlusion relations among edges

图 2 按边之间遮挡关系将多边形逐渐地剖分成一些单调多边形

下面,我们讨论如何高效地寻找入射边单元和生成出射边单元中的新边,并进行有效性证明,即任何时候都会有一个极大单调边序列适宜作为入射边单元,并能进行单调多边形的剖分.

### 3.1.1 寻找入射边单元

为了高效地寻找入射边单元,我们先根据遮挡关系将多边形的边分为一些层次,即没有被遮挡的边分为第 1 层次,只被第 1 层次的边所遮挡的边分为第 2 层次,只被第 1 层次和第 2 层次的边所遮挡的边分为第 3 层次,依此类推.在考察任意两条边之间的遮挡关系时,先判断它们各自的  $y$  坐标值区间有没有重叠的部分,如果没有,则它们之间无遮挡关系;如果有,就以通过重叠部分的一条射线来判断它们之间的遮挡关系.

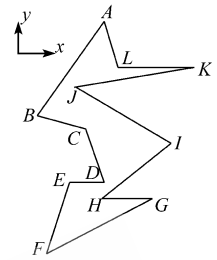


Fig.1 A monotone polygon consists of two monotone edge sequences  
图 1 一个单调多边形由两个单调边序列组成

根据多边形扫描转换算法中关于扫描线的讨论,不难证明:只有奇数层次的边才会形成入射边单元,只有偶数层次的边才会形成出射边单元.

于是,先根据第 1 层次中的边找到一个入射边单元,然后根据它所遮挡的第 2 层次的边并生成一些相应的新边(具体操作见第 3.1.2 节),就得到对应的一个出射边单元.由于新边只被入射边单元所遮挡,当分离出该入射边单元对应的单调多边形后,这些新边将是分离后多边形的第 1 层次的边.而该出射边单元中原多边形的边,将不再存在于分离后的多边形中,且那些只被它们所遮挡的第 3 层次的边,将成为分离后多边形的第 1 层次的边.这样,每次寻找入射边单元的操作,只要在当时的第 1 层次的边中进行即可.

### 3.1.2 生成出射边单元中的新边

为分割出单调多边形而生成新边,我们采用下面的 3 个步骤:

(1) 根据入射边单元,找到那些只被它们所遮挡的边.将这些边按其  $Y$  坐标顺序排列,对于排列中间断的地方,将  $Y$  坐标相邻的两个顶点连接成一条新边,于是就构成了一个与入射边单元对应的出射边单元.该出射边单元与入射边单元可以形成一个单调多边形.

但是,由于凹点的存在,这些新边可能会与一些原多边形的边相交,使得所形成的单调多边形有些部分在原多边形的外面.这种凹点可以分为两类:来自入射边单元的凹点(第 1 类),只被入射边单元所遮挡的凹点(第 2 类).如图 3 所示,点  $C$  是第 1 类凹点,点  $H$  是第 2 类凹点.

于是,将这些凹点与新边进行考察,即以一个凹点的  $Y$  坐标形成一条与  $X$  轴平行的线,求与新边的交点.然后比较交点的  $X$  坐标与凹点的  $X$  坐标,如果凹点是第 1 类的,而交点的  $X$  坐标小于凹点的  $X$  坐标,则新边会与此凹点关联的边相交;如果凹点是第 2 类的,而交点的  $X$  坐标大于凹点的  $X$  坐标,则新边会与此凹点关联的边相交.

如果检测后,所有新边与多边形的边没有相交,则该单调多边形就能成功地分割出来.否则就要进行下面的步骤,修改有相交情况的新边部分,以保证新边都是位于多边形内的.此时,对于没有相交情况的新边部分,则不要修改.

(2) 如果有第 1 类的凹点关联的边与一条新边相交,则去掉该新边,并将该新边关联的两个顶点、其  $Y$  坐标位于这两个顶点  $Y$  坐标区间中的第 1 类的凹点按照  $Y$  坐标顺序排列,然后根据这个排列,顺序地在两点之间连成新边.显然,这些新边不会与第 1 类凹点关联的边相交,因为关联同一个顶点的边不会相交.

但是,这些新边还是有可能与第 2 类凹点关联的边相交.于是,将这些新边与对应的第 2 类凹点进行相交检测,即那些其  $Y$  坐标位于新边的  $Y$  坐标区间内的第 2 类凹点.如果没有相交的,则这些新边能够成功地分割出一个单调多边形.如果有相交的,则对这些新边进行下一步的修改.当然,没有相交情况的新边部分就不要修改.

(3) 对有相交情况对应的第 2 类凹点和新边,仿照上面(2)的操作,可以生成一些与这些凹点关联的边不相交的新边.

由此,这些新边都在多边形内部,可以成功地分割出一个单调多边形.

如图 3(a)所示,由于第 1 类凹点  $C$  和第 2 类凹点  $H$  的存在,新边  $FL$  将与多边形的原有边产生异常相交.

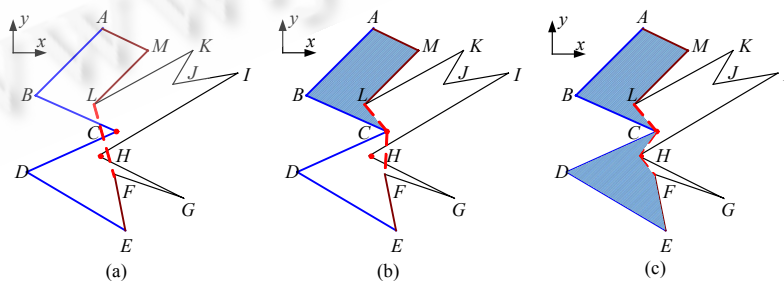


Fig.3 Handle intersections between original polygon's edges and new edges induced by concave vertices

图 3 对凹点引起的新边与原多边形的边有相交的情况的处理

首先针对凹点  $C$  进行调整,将新边  $FL$  的两个顶点  $F, L$  与  $C$  按其  $Y$  坐标排序,再顺序连接出新边  $CL$  和  $FC$ ,

如图 3(b)所示.此时,由于第 2 类凹点  $H$  的作用,新边  $FC$  仍与  $H$  关联的边相交.与处理点  $C$  的方法类似,将点  $C, H, F$  按其  $Y$  坐标排序,再形成新边  $FH$  和  $HC$ .这样,最终生成的这些新边不会与原多边形的边相交,如图 3(c)所示.

#### 4 三维多面体的凸剖分

本文讨论的多面体是由许多面构成的一个三维空间中的封闭体,并且它的任意两个面只在它们共享的顶点和边处相交.它的每个面都是由一些共面的点生成的平面多边形,而那些点就是该多面体的顶点,那些多边形的边就构成多面体的边.显然,一个顶点可以与多个面关联,而一条边只能与两个面关联.

上一节讨论了基于多边形元素(点、边)之间的遮挡关联性对多边形凸剖分的方法.运用相同的思路,本节将探讨基于多面体元素(点、边、面)之间的遮挡关联性进行多面体凸剖分的方法.在此,先给出几个定义:

- 单层多面体:如果对于某平面  $p$ ,任一垂直于  $p$  的直线  $l$  与一个多面体的交只是一个区间,即为一条线段、一个点或为空,则称该多面体为关于平面  $p$  的单层多面体.若某多面体关于  $XY$  平面是单层的,则称为  $XY$ -单层多面体.更直观的描述是在沿  $Z$  轴方向正投影地观察一个  $XY$ -单层多面体时,任一视线与该多面体的交只是一条线段、一个点或者没有交.显然,对一个单层多面体而言,它的面只可以分成遮挡的和被遮挡的两个集合.图 4 中显示了一个  $XY$ -单层多面体,其遮挡面集合为  $\{P_1P_2C, P_2P_3C, P_3P_4C, P_4P_1C\}$ ,被遮挡面集合为  $\{Q_1Q_2P_2P_1, Q_2Q_3P_3P_2, Q_3Q_4P_4P_3, P_4Q_4Q_1P_1, Q_4Q_3Q_2Q_1\}$ .在本文后面的叙述中,  $XY$  单层多面体都简称为单层多面体.本文亦据此讨论新方法,而不失一般性.
- 凹点:若一个点是某个凹边的顶点,则称这个点为凹点.

当我们沿着某个轴方向正投影地观察一个多面体时(本文是沿着  $Z$  轴方向),多面体的各个面、边和点之间存在遮挡关系.基于这种遮挡关联性,一个多面体可以剖分成一些单层多面体.如果单层多面体不是凸的,再对它进行凸剖分.

为简化后面的讨论,我们假设多面体的各个面都是三角面片(如果不是,则先三角剖分之),并且它的各个面在形成遮挡关系时不存在循环遮挡的现象(循环遮挡的现象,在处理二维多边形时不会发生,但在处理三维多面体时可能出现,详见参考文献[17]).

下面,我们先讨论多面体的凸剖分方法,然后进行多面体凸剖分算法的分析.

##### 4.1 多面体的凸剖分

为论述简便,本文着重讨论简单多面体的凸剖分,而对包含空洞的多面体的凸剖分,甚至是包含孤立面、孤立边和孤立点的非流形多面体的凸剖分,本文方法也可类似地处理,这里不再赘述.凸剖分主要有两个步骤:先将它剖分成一些单层多面体(本文是沿着  $Z$  轴方向剖分成一些  $XY$  单层多面体),然后,对于非凸的单层多面体进行进一步的凸剖分.

仿照第 3 节的讨论,在将简单多面体剖分成单层多面体时,首先在多面体中找到一个没有被遮挡的极大面单元(称为入射面单元),即该单元中的面都没有被遮挡,都是基于共享边连通的,且其他与它们基于共享边连通的面都会被遮挡.然后,找到那些只被入射面单元所遮挡的面,以及据此生成的一些在多面体内的新面,形成对应此入射面单元的一个面集合,称为出射面单元.于是,此入射面单元和出射面单元就能生成一个单层多面体.

类似于第 3 节的相关证明,我们有以下结论:

- 简单多面体中,总能找到一些面形成入射面单元,并能成功地分离出一个单层多面体.
- 为了形成一个单层多面体而生成的位于多面体内的新面,在分离出单层多面体后将复制在留下的多面体中,且将属于没有被遮挡的面.在分离后的多面体中,新面所遮挡的面,就是原入射面单元中遮挡新面的面所遮挡的面.

迭代地进行单层多面体的剖分,就能将一个多面体剖分成一些单层多面体,因为每个单层多面体中都有原多面体的一些面,而原多面体的每个面只会被划分到一个单层多面体中.

下面,我们详细讨论如何高效地寻找入射面单元以及生成出射面单元.

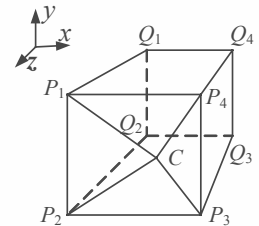


Fig. 4 A single layer polyhedron

图 4 单层多面体示意图

#### 4.1.1 寻找入射面单元

根据面与面的遮挡关系,可以将它们进行排序分类,即:没有被其他面片所遮挡的面归为第 1 层;只被第 1 层面片所遮挡的面片归为第 2 层;依此类推,就将面片归入了不同的层次.比较任意两个面之间的遮挡关系的操作是:求出它们各自在  $XY$  坐标面上的正投影,如果其投影之间没有重叠,则它们之间没有遮挡关系;如果有重叠,则从重叠区域中找到一点生成一条平行  $Z$  轴的线,以求与这两个面的交点,通过比较交点的  $Z$  坐标就可以知道它们之间的遮挡关系.

与第 3 节的讨论类似,我们可以推知:奇数层次的面才可能成为入射面单元,偶数层次的面才可能成为出射面单元.于是,对于奇数层次的面片,根据连通性将它们进一步分为许多单元,并为各个单元记录只被其所遮挡的面片和偶数层次的面中的凹边(只要有一部分被直接遮挡的凹边即是,这是为生成出射面单元而用的,将在第 4.1.2 节中加以讨论).

由于分离出一个单层多面体后,新面在分离后的多面体仍然属于未被遮挡的面.于是,寻找入射面单元只要在多面体当时的第 1 层面片中进行即可.

#### 4.1.2 生成出射面单元

为一个入射面单元生成其对应的出射面单元,可按照以下方式进行:

首先,根据各个面在  $XY$  坐标面上的投影,找到那些只被入射面单元所遮挡的面(称为原遮挡面单元);对这些面与入射面单元在  $XY$  坐标面上的投影进行求差,找到那些位于入射面单元投影内而不位于原遮挡面单元投影内的部分(称为差区域);将差区域进行不加新点的三角化;然后,将三角化中新生成的三角形对应到三维空间中生成新的面(称为新遮挡面单元),即将这些三角形的边根据各自顶点在三维空间的对应点生成它们在三维空间对应的边.这称为第 1 次三角化.

然后,检测这些第 1 次三角化得到的新遮挡面是否位于多面体内部.如果新遮挡面都位于多面体内部,则原遮挡面单元与新遮挡面单元就可以构成该入射面单元对应的出射面单元,一起生成一个单层多面体.否则,就要进行进一步的操作,以保证新遮挡面位于多面体内部.

在讨论进一步的操作之前,我们先讨论如何检测新遮挡面是否位于多面体内部.

由于任意一个新生成的面都是三角形,则它肯定是一个平面片;而它的顶点也是多面体的顶点,因此,如果它不位于多面体内部,则一定有多面体的边与它相交,并且这样的边只能是下面两种情况之一.

- 入射面单元中至少有部分片段位于差区域中的凹边.因为只有这样的凹边才可能比对应的新生成的面更深入多面体内部,由此使得它所关联的边与新生成的三角形面相交.
- 不属于入射面单元且满足后面条件的一些凹边,即该凹边中至少有 1 段是只被入射面单元所遮挡的.因为与新遮挡面相交的边,它至少有一段位于新遮挡面与入射面之间,则该段肯定只被入射面所遮挡;由于这段的作用,该边相关的两个面都各有一部分只被入射面所遮挡,而这两部分也就一定会遮挡部分新遮挡面,又由于这两个面不属于原遮挡面,则它们肯定各有一部分被新遮挡面所遮挡,于是,这两个面在多面体内的夹角一定大于  $180^\circ$ ,所以该边是凹边.

于是,检测新遮挡面是否位于多面体内部,按照以下步骤进行:

(1) 找出入射面单元中至少有部分片段位于差区域的凹边,以及至少有一部分只被入射面单元所遮挡的凹边.

(2) 根据它们在  $XY$  坐标面上的投影,找到与各自投影有重叠的新遮挡面,以进行边与面的相交检测.

如果有新遮挡面与一些多面体的边有相交,则将这些新遮挡面在  $XY$  坐标面上的覆盖区域重新进行三角化,而其余的新遮挡面则予以保留.对于需要重新三角化的区域,上述的两种凹边可能会同时出现,且这两种凹边在  $XY$  坐标面上的投影可能相交.为此,我们不对这两种凹边同时处理,而是先处理第 1 种凹边的情况,使得重新三角化的面不会与这种凹边相交,然后对这些新面与第 2 种凹边进行相交检测,如果有相交情况,再处理这些第 2 种凹边,以作进一步的三角化.这样,就能保证新三角面不会与各种凹边相交.不失一般性,假设先处理入射面中的凹边,其操作如下:

(1) 与第 1 次三角化的面有相交的属于入射面单元的凹边,投影到差区域中,并计算它在差区域中的片段(凹边片段),然后对差区域中相关部分进行三角化并生成新三角面(第 2 次三角化).显然,这些新三角面不会与入射面单元的凹边相交,因为有共享顶点的两个平面片要么分离,要么相邻,不会相互穿刺.

(2) 对第 2 次三角化的面,检测它们是否与那些有部分片段只被入射面单元遮挡的凹边相交.如果没有相交情况,则完成了新三角面的生成;如果有相交,则确定这些有相交的三角面的区域为新的差区域.

(3) 对那些引起边/面相交的凹边,投影到  $XY$  坐标面上的新差区域,并计算出它们在新差区域的片段.然后,根据这些凹边片段,对新差区域进行三角化(第 3 次三角化).显然,这时生成的新三角面不会与这些凹边相交,也不会与引起第 2 次三角化的凹边相交(因为,引起边/面相交的两种凹边在三维空间是不会相交的,这使得对新差区域三角化时不会改变第 2 次三角化时由入射面单元中的凹边片段生成的新三角面的边).

在  $XY$  坐标面上三角化时,那些在  $XY$  坐标面上新加入的凹边片段,它们的端点将是新三角化后三角形的顶点,无论它们是否与原来已有的在  $XY$  坐标面的边或顶点重叠;并且重新三角化后生成对应的三角形面时,这些顶点都是对应它们原来在三维空间的位置.在此过程中,由于有些凹边只有一部分片段参与了新的三角化,所以这些边上会生成新的顶点,基于这些新的顶点生成的新边也可能与原来的边共线,但它们是属于不同的面的.

这样重新生成的三角形面片一定位于多面体内部,否则,就有多面体的边与它们相交,且任意一条这样的边都会至少有一部分只被入射面单元所遮挡,但是这样的边在此时不可能存在.

下面,我们以图 5 为例解释多面体剖分成单层多面体的过程,并在图 6 中解释如何处理凹边与第 1 次三角化的新面相交的情况,而第 2 次三角化的处理是类似的,这里不再赘述.

如图 5 所示,对于入射面单元  $\{P_1P_2C, P_2P_3C, P_3P_4C, P_4P_1C\}$ ,先找到只被它的面片所遮挡的原遮挡面单元  $\{Q_1Q_2P_2P_1, Q_2Q_3P_3P_2, Q_3Q_4P_4P_3, P_4Q_4Q_1P_1\}$ .两者在  $XY$  平面上的投影分别如图 5(b)、图 5(c)所示.对这两者的差区域(它们在  $XY$  坐标面上投影的不重叠区域)进行三角化,如图 5(d)所示.最后生成相应的新遮挡面单元  $\{Q_1Q_3Q_2, Q_1Q_4Q_3\}$ .于是,原遮挡面单元与新遮挡面单元一起构成了出射面单元  $\{Q_1Q_2P_2P_1, Q_2Q_3P_3P_2, Q_3Q_4P_4P_3, P_4Q_4Q_1P_1, Q_1Q_3Q_2, Q_1Q_4Q_3\}$ .它们与入射面单元一起生成一个单层多面体,从原多面体中分离出来,而新遮挡面单元则加入分离后的原多面体构成原多面体分离后的多面体(如图 5(e)所示).

如图 6 所示,凹点  $C$  位于模型的后半部分内,其入射面单元及原遮挡面单元在  $XY$  平面上的投影及差区域及第 1 次三角化均与图 5(b)~图 5(d)相同.此时,属于入射面单元的凹边  $\{P_1C, P_2C, P_3C, P_4C\}$  与新遮挡面单元  $\{Q_1Q_3Q_2, Q_1Q_4Q_3\}$  将相交.因此应将与新面  $Q_1Q_3Q_2$  和  $Q_1Q_4Q_3$  有相交的凹边  $\{P_1C, P_2C, P_3C, P_4C\}$  投影到差区域中,并计算它们在差区域中的片段(凹边片段)  $\{N_1'C, N_2'C, N_3'C, N_4'C\}$ ,如图 6(b)所示.其中,  $N_1', N_2', N_3', N_4'$  对应为各凹边上的新点.随后对差区域进行三角化并生成新三角面(第 2 次三角化),如图 6(c)所示.最后按照图 6(d)生成对应的新遮挡面.值得注意的是,有些边在投影平面上是重叠的但在空间中却并不重叠,它们在三角化过程中都要作为不同的个体,如  $Q_1'N_1', N_1'Q_4'$  和  $Q_4'Q_1'$  在投影平面上是重合的,但它们在  $XY$  坐标面上三角化为一个退化的三角形,并对应到三维空间组成一个三角面  $Q_1N_1Q_4$ .

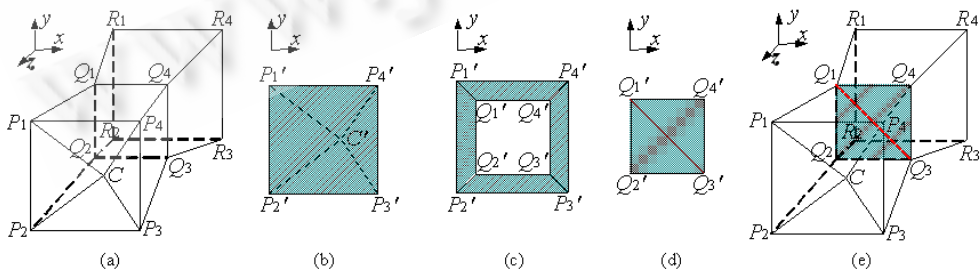


Fig.5 Decompose a polyhedron into single layer polyhedrons using occlusion relations

图 5 根据遮挡关系,将简单多面体剖分成一些单层多面体的基本过程



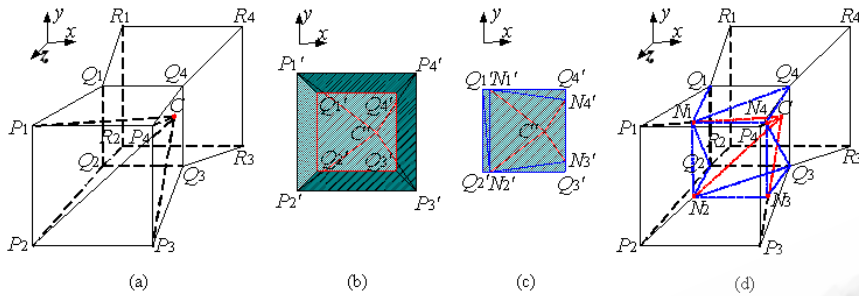


Fig.6 Handle intersections between new faces and reflex edges

图 6 新遮挡面与凹边异常相交情况的处理

4.1.3 单层多面体的凸剖分

对单层多面体进行凸剖分,我们将尽量剖分出一些四面体.

在剖分过程中,为了减少对边的分割而生成新顶点的次数,我们尽量进行局部化的操作.为此,单层多面体的凸剖分按下面两个步骤进行.

(1) 面-点剖分.

对每个入射面单元的面,考察它在 XY 坐标面上的投影是否含有出射面单元中的顶点在 XY 坐标面上的投影.如果有,则以该入射面生成一个四面体,并保证分离出该四面体后的多面体还是单层多面体.对于每个出射面单元的面也进行这样的处理.如此迭代地进行,就能剖分出很多四面体,且分离后的多面体还是单层多面体.

在此剖分过程中,每分离出一个四面体,该四面体的某些面就要复制以形成分离后的多面体.此时,对这些面所关联的边或顶点在分离后的多面体中进行考察,检测它们是否可以断开以分离出多面体而不影响多面体的形状.如果能够分离,就分离出多个单层多面体,再对这些多面体分别进行进一步的凸剖分.如图 7 所示,在将四面体 {CFB,CDF,CBD,FDB} 分离后,原多面体(如图 7(a)所示)的剩余部分为基于一条共享边 CF 相连的两个四面体 {ABC,CBF,CFA,AFB} 和 {CFD,CDE,CEF,EDF} 的组合(如图 7(b)所示),此时从边 CF 处断开就能得到分离的两个四面体.

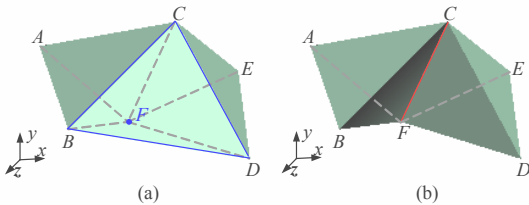


Fig.7 An example of separation

图 7 分离情况示例

当这种操作不能再进行时,剩下的单层多面体中入射面单元的顶点都与出射面单元的顶点共点(称为共点单层多面体),如后文的图 9(a)所示.

(2) 切割剖分.

经过上面操作后得到的多面体中,有些还是会有凹边的.此时,就以这些凹边形成一些平行 Z 轴的面去切割多面体.由于这时的多面体的规模都比较小,所以因切割而生成的新顶点不会很多,并且由于共点单层多面体的特征,最多只要为每条凹边生成一个切割面,就能完成多面体的凸剖分.

下面,我们对这两个步骤进行详细的讨论.

面-点剖分.我们对入射面单元和出射面单元中的面分别处理,即对于一个入射面,在出射面单元中找到一个与其顶点相异的顶点,一起生成一个四面体;对于一个出射面,就在入射面单元中找到一个与其顶点相异的顶点,一起生成一个四面体.其操作如下:

(1) 对于一个入射面,如果有一些出射面的顶点在 XY 坐标面上的投影位于它在 XY 坐标面上的投影区域内,且这些顶点与入射面的顶点不共点,就选择其中任意一个与该入射面一起生成一个四面体.

(2) 虽然这些出射面的顶点关联的面之间没有遮挡关系,但由于凹边的存在,上面生成的四面体可能与出射面单元中的一些凹边相交,并且相交的面一定是新生成的三角面(这很容易根据剖分多面体生成单层多面体的讨论推知).此时,求出那些位于新生成三角面内部的交点(不与三角面的边重叠的交点).如果没有这样的交

点,则表明(1)中生成的四面体可以分离出去;如果有,则转(3).

(3) 舍弃(1)中生成的四面体,然后入射面尝试与这些交点生成四面体,并检测所生成的四面体是否还与出射面单元中的凹边相交.最后,选定没有任何相交的一个四面体作为该入射面生成的四面体,并分离出去.此时,新生成的三角面一定是在多面体内部(根据剖分多面体生成单层多面体的讨论可以推知).而这些新生成的三角面将进行复制以得到分离后的多面体.由于这些新生成的三角面在XY坐标面上投影与那个入射面一样,而它们也只被那个入射面遮挡,所以,它们在分离后的多面体中仍然属于入射面单元.因此,分离后的多面体还是单层多面体.

(4) 对于一个出射面也采用类似的方法,可以分离出一个四面体,且新生成的三角面仍然属于出射面单元.

以图8为例,解释面-点剖分的操作.图8(a)、图8(b)分别为某单层多面体的入射面单元和出射面单元,图8(c)是它们在XY平面上的投影,边AE为出射面单元中的一条凹边.对于入射面CBD来说,出射面顶点F的投影F'落在其投影C'B'D'内,故可选择点F与它共同生成一个四面体{CBD,CFB,CDF,FDB}.但由于凹边AE的存在,新面CFB和CDF将分别与AE交于点G和H,如图8(d)、图8(e)所示(图8(d)为XY投影图,且为了显示清晰,图8(e)未画出入射面).故应舍弃该点面构成方式,重新选择交点G或H与CBD形成新的四面体{CBD,CGB,CDG,GDB},如图8(f)、图8(g)所示(其中图8(f)为XY投影图,且图8(g)未画出入射面).新四面体不再与出射凹边相交,因此可以将该四面体分离出去.

**切割剖分.**对于一个共点单层多面体,它的所有顶点在XY坐标面上的投影一定是位于多面体投影的边界上.而根据单层多面体的定义可知:入射面单元中各条边之间是不会相交的,而出射面单元中各条边之间也是不会相交的.

于是,采用Chazelle的“切割分裂”策略进行剖分,即根据入射面单元中一条凹边形成一个平行Z轴的切割面去分割多面体.它与入射面单元中的边都不会相交,而只会与出射面单元中的边相交.根据该切割面,可将多面体剖分成两个部分.由于这个面是平行于Z轴的,而这条凹边相关的两个面在XY坐标面上的投影没有重叠,所以,该切割面分别与这条凹边相关的两个面的夹角都不会大于180°.于是,这条边在剖分的两个多面体中都不再是凹边.

在剖分的这两个多面体中,切割面会与出射面单元中的一些边相交而生成一些新点.将这些新点分别与引起切割面的凹边的顶点相连,就可以生成许多共面的边.而对于与切割面相交的出射面单元的边,每一条都会被分割成两段而分别加入两个剖分后的多面体中.如果它是凸的,那么它的两段边在剖分后的两个多面体中依然也会是凸的.对于凹边,也是这样.于是,剖分后的各个多面体相对于剖分前,它们的凹边数目总是减少的.如此迭代地进行,最后就能消除所有凹边.

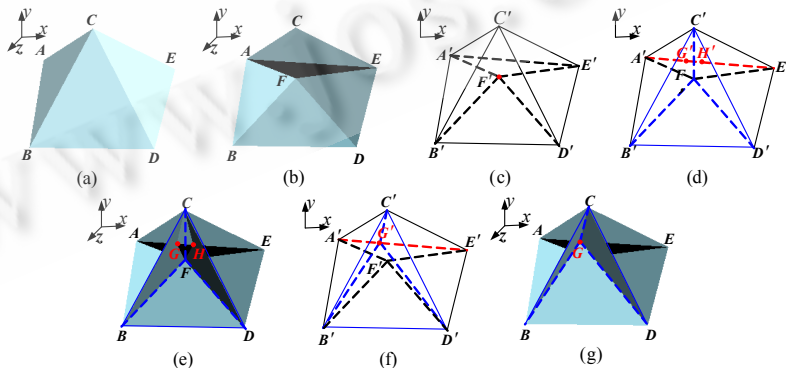


Fig.8 An example of face-vertex decomposition

图8 面点剖分示意图

入射面单元中的凹边处理完成后,出射面单元的凹边也进行同样的处理.这样,就能将一个单层多面体剖分成一些凸多面体.

如图 9(a)所示,切割剖分的操作如下:面 $\{ABC, ACD, ADE\}$ 为入射面,面 $\{AEB, BED, BDC\}$ 为出射面,两者共同构成一个单层多面体.其中,边 $AD$ 为入射面单元的凹边,边 $BD$ 为出射面单元的凹边.过凹边 $AD$ 作平行于 $Z$ 轴的面切割该多面体,与边 $BE$ 相交,生成新点 $F$ ,如图 9(b)所示.此时各个剖分体的入射面单元中已无凹边,但有一个剖分体的出射面单元中还存在凹边 $BD$ ,因此再过 $BD$ 作平行于 $Z$ 轴的切割面,与边 $AC$ 相交,生成新点 $G$ ,如图 9(c)所示.经过上述两次操作,所有凹边均已被分解,原单层多面体被剖分为 3 个凸多面体.

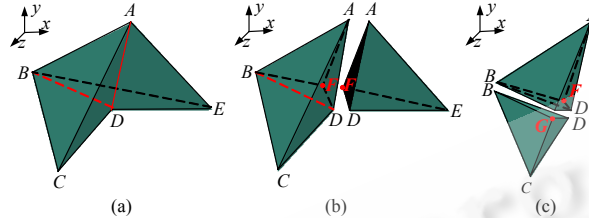


Fig.9 An example of cutting decomposition against a common vertex single layer polyhedron

图 9 切割剖分一个共点单层多面体的示意图

## 4.2 算法分析

在此,我们讨论凸剖分新方法的时间复杂度、空间复杂度、剖分生成的凸多面体个数的复杂度和剖分中需增加新的顶点个数的复杂度.在下面的讨论中,多面体的点、边和面片的数目均用 $n$ 表示,因为欧拉公式表明它们之间是线性相关的.

Chazelle已然证明,其基于“切割分裂”的算法是最坏情况下的最佳(凸多面体数量)<sup>[7]</sup>.但在实际应用中,算法对常见模型表现出的复杂度常常比最坏复杂度更有意义<sup>[18]</sup>,因此,下面将不讨论最坏情况下的复杂度,而是根据

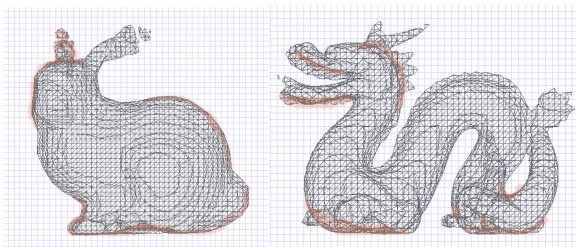


Fig.10 Common models

图 10 常见模型示意图

应用中常见模型的性质来进行针对性的分析.应用中的模型通常具有如下性质:(a) 完全或绝大部分由三角面片或者四边形构成;(b) 面片大小比较均匀;(c) 大多数区域遮挡层次数很少,为 $O(1)$ 量级,仅在少数形状变化较剧烈的皱褶区域才存在多层次的遮挡关系,且很少存在循环遮挡的情况.如图 10 所示,以Bunny和Dragon为例,可以比较明显地看出这 3 点性质.图中颜色较深的边缘区域即为遮挡层次较多的部分.根据上述 3 点性质,下面将具体分析新方法在实际剖分中表现出的时

间、空间、剖分结果个数以及新点个数 4 项指标的复杂度.

新的多面体凸剖分方法包括以下几个运算步骤:对非三角面片进行三角剖分;对面片进行排序;将多面体剖分为单层多面体;对各个单层多面体进行剖分.这些步骤各自的复杂度分析如下:

- 对非三角面片进行三角剖分

由于四边形的三角剖分需要时间为 $O(1)$ ,而根据性质(a)多面体的面片绝大部分为四边形或三角形,所以本步骤所需时间为 $O(n)$ .

- 对面片进行排序

最坏情况下,任意两个面之间都要进行遮挡检测,所以复杂度为 $O(n^2)$ .但在实现中,我们可以采用网格法加速遮挡关系的判定,即将多面体包围盒、多面体面片向某投影面(通常为坐标平面)进行正投影;然后基于投影面的均匀网格剖分,对每个网格记录所有投影落入其中的三角面片;之后,每个三角面片仅与投影落入同一网格的三角面片进行遮挡关系的判定.由于三角面片大小比较均匀,所以每个面片平均只与近似 $O(1)$ 个面片有遮挡关系.这样,计算所有遮挡关系需要的时间约 $O(n)$ ,存储遮挡关系所需空间也约为 $O(n)$ .

- 将多面体剖分为单层多面体

单层多面体的封闭过程主要是对差区域多边形的三角剖分,该多边形为入射面单元投影与出射面单元投影的差,可能包含空洞并可能存在空洞嵌套.根据常用多面体这样的分析前提,我们下面将推导三角剖分差区域  $DIFF_i$  的复杂度.

为描述方便,首先给出一些标识.假设我们封闭第  $i$  层的单层多面体,其入、出射面单元分别为  $IN_i, OUT_i$ ; 它们的投影分别为  $IN'_i, OUT'_i$ ; 差区域为  $DIFF_i$ ;  $IN'_i$  和  $OUT'_i$  的边界分别表示为  $\partial IN'_i$  和  $\partial OUT'_i$ . 令  $S[x]$  表示  $x$  的面积;  $TNum[x]$  表示  $x$  包含的三角形个数;  $ENum[x]$  表示  $x$  包含的三角形总边数. 其中,  $x$  可以是  $IN_i, OUT_i, IN'_i, OUT'_i, \partial IN'_i$  和  $\partial OUT'_i$  中的任意一项.

首先,我们分析差区域不包含嵌套空洞的情况.  $DIFF_i$  的外边界为  $\partial IN'_i$ ,  $DIFF_i$  的内边界为  $\partial OUT'_i$ , 因此  $DIFF_i$  的边数为两者之和. 根据前述的常用应用模型性质有:

$$O(TNum[IN'_i]) = O(ENum[IN'_i]) = O(S[IN'_i])$$

$$O(TNum[OUT'_i]) = O(ENum[OUT'_i]) = O(S[OUT'_i]).$$

并且,由面积与周长的关系有:

$$O(ENum[\partial IN'_i]) = O(TNum[\partial IN'_i]) = O((S[IN'_i])^{1/2}) = O((TNum[IN'_i])^{1/2})$$

类似地有  $O(ENum[\partial OUT'_i]) = O((TNum[OUT'_i])^{1/2})$ . 也就是说面单元边界处的边数复杂度与面单元中三角形个数复杂度呈二次根号关系(以矩形均匀网格为例,一个有  $n$  个均匀网格单元的矩形,其边界处的单元数为  $n^{1/2}$ ).

由前提条件我们还可以推出以下结论:差区域  $DIFF_i$  大小具有局限性,即  $IN'_i$  与  $OUT'_i$  的面积差不会太大(具体分析见文献[17]). 因此,有

$$O(S[\partial IN'_i]) \approx O(S[\partial OUT'_i] + (S[\partial OUT'_i])^{1/2}) = O(S[\partial OUT'_i]) = O(TNum[\partial OUT'_i]) = O(ENum[\partial OUT'_i]).$$

仍以矩形均匀网格为例,一个有  $n$  个均匀网格单元的矩形,其边界处的网格单元数为  $O(n^{1/2})$ , 该边界向外扩充 1 个单元宽度的大小,则扩充后的总网格单元数为  $O(n+n^{1/2})$ .

那么,差区域的总边数近似为  $O(TNum[\partial OUT'_i])$ , 剖分后得到的三角面片数量近似为  $O(TNum[\partial OUT'_i])$ , 剖分时间近似为

$$O(TNum[\partial OUT'_i] \times \log(TNum[\partial OUT'_i])) = O(TNum[OUT'_i]^{1/2} \times \log(TNum[OUT'_i]^{1/2})) < O(TNum[OUT'_i]).$$

另外,从  $IN'_i, OUT'_i$  中搜索出外部边界需要时间为  $O(TNum[OUT'_i])$ , 因此生成一个单层多面体所需时间为  $O(TNum[OUT'_i])$ . 由于  $\sum TNum[OUT'_i] = O(n)$ , 因此不考虑凹边特殊处理的情况下所有单层多面体的剖分过程需要时间近似为  $O(n)$ , 并产生  $O(n^{1/2})$  个新面片. 如果考虑到凹边,根据分析前提,每条凹边平均可与近似  $O(1)$  个新生三角面片发生关联. 除原有凹边外,新生的三角面片会生成新的凹边,故所有单层多面体的凹边总数近似为  $O(r+n^{0.5})$ , 因此生成单层多面体过程中处理凹边所需时间近似为  $O(r+n^{0.5})$ , 生成新点数量也近似为  $O(r+n^{0.5})$ .

若差区域包含嵌套空洞,则根据分析前提,空洞的数量不会很多,不会达到与  $OUT'_i$  的边数一个数量级. 因此,面单元边界处的边数复杂度与面单元中三角形个数复杂度的二次根号关系依然成立. 类似地,我们也可以证明存在嵌套空洞关系下,差区域大小的局限性仍然成立(证明略). 因此,对于差区域包含嵌套空洞的情况,上述复杂度分析结果不变.

- 单层多面体的凸剖分

在面点剖分阶段,若不考虑凹边,则每个面片与对侧的  $O(1)$  个点组合生成  $O(1)$  个新面片,共需要  $O(n)$  时间,产生  $O(n)$  新面片. 每条凹边与近似  $O(1)$  个三角面片关联,故处理凹边时产生  $O(r+n^{0.5})$  个新点,  $O(r+n^{0.5})$  个新面片,需要  $O(r+n^{0.5})$  时间. 共点单层多面体的切割阶段的分析也类似.

综合以上各个步骤分析,新方法的剖分时间近似为  $O(n)$ , 所生成新面片数量近似为  $O(n)$ , 新点数量近似为  $O(r+n^{0.5})$ , 当凹边数量很多,达到  $O(n)$  时,新点数量即近似为  $O(n)$ . 由于凸多面体数量不会超过三角面片数量,因此近似为  $O(n)$ . 同理,空间复杂度亦为近似  $O(n)$ .

### 5 实验结果

我们用 Visual C++7.0 实现了本文的多面体凸剖分方法,并实现了文献[8]中基于“切割分离”的凸剖分方法作为对比,因为,根据文献[9]中的讨论,文献[8]是当前各种凸剖分方法的基础,并是精确剖分方法中效率最高的方法.

我们在一台微机上进行了实验,其配置为一个 AMD Athlon 3200+的 CPU 和 1G 内存.

为检测新方法各方面的性质,我们设计了如下 3 项实验:

- (1) 测试多面体规模增长时,算法各项指标的变化情况,以验证复杂度分析.
- (2) 对比不同情况下新旧算法的性能.
- (3) 测试新方法对复杂模型的剖分情况.

在进行第 1 项实验时,我们用 3DMAX 生成了两组模型,一组为 TorusKnot 模型(如图 11 所示),其凹边比例(即多面体凹边数在总边数中所占比例)保持在 20%~33%,面片数从 200 增长到 102 400.另一组为带空洞的嵌套 NoiseSphere 模型(如图 13 所示),其凹边比例为 50%,面片数从 4 000 增长到 16 000.两组模型的实验结果分别如图 12 和图 14 所示.

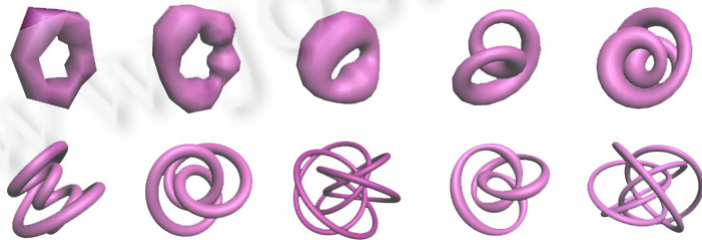


Fig.11 TorusKnot models used by the first experiment

图 11 用于第 1 项实验的 TorusKnot 模型

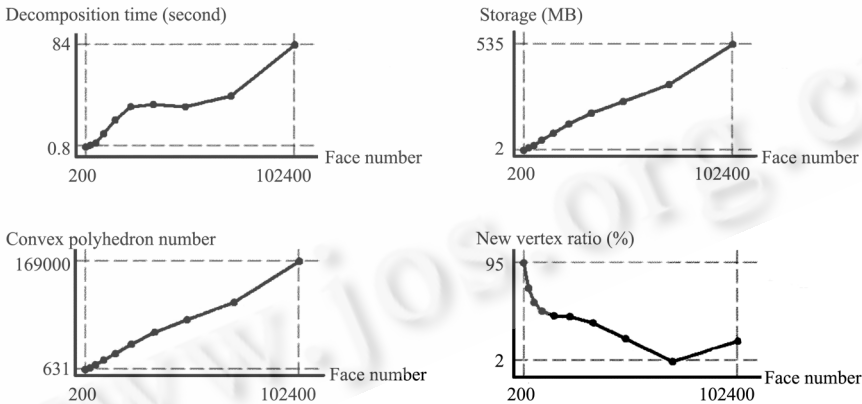


Fig.12 Results of the first experiment on TorusKnot models

图 12 第 1 项实验 TorusKnot 模型组测试结果

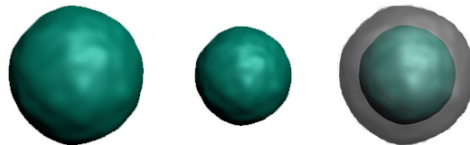


Fig.13 Nested NoiseSphere models which is used for testing decomposing efficiency on polyhedrons including internal voids. The right model is produced by embedding the middle model into the left model

图 13 嵌套 NoiseSphere 模型,用于测试对包含内部空洞的多面体的凸剖分效率.右边的模型是在左边的模型中嵌入一个由中间模型表达的空洞而生成的

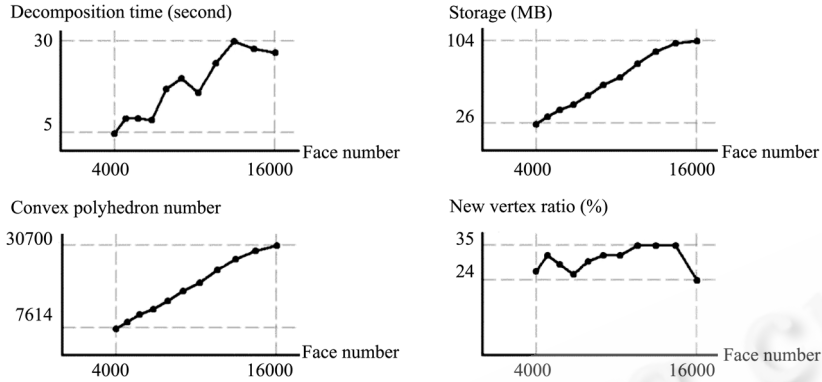


Fig. 14 Results of the first experiment on NoiseSphere models

图 14 第 1 组实验嵌套 NoiseSphere 模型组测试结果

由这组实验可知,在凹边比例基本稳定的情况下,新方法凸剖分的时间、空间和剖分结果凸多面体个数呈较为明显的线性变化趋势,而新点比例(新点个数/原顶点个数)也比较稳定,并不会随模型规模增长而增长,这说明,新点个数复杂度不会超过  $O(n)$ 。另外,与空间的数据曲线相比,剖分时间和新点比例呈现出较大的随机性,这是由于这两项指标很大程度上依赖于凹边与新生面片的相交次数,相交次数越多,所需的额外处理时间就越多,新点数也会越多。但是,这个相交次数本身也存在着较大的随机性——模型的形状、放置方向、凹边的分布等等都会影响到这一指标。

总之,本项实验较好地验证了上一节的复杂度分析——剖分时间  $O(n)$ ,空间  $O(n)$ ,凸多面体个数  $O(n)$ ,新点数不超过  $O(n)$ 。

在进行第 2 项实验时,我们同样用 3DMAX 生成了两组形如图 15 的 NoiseSphere 模型,其中一组凹边比例固定为 35%,面片数量从 2000~8000,另一组面片数量固定为 2 000,凹边比例从 5%~45%。每组模型都用新、旧两种方法进行剖分测试,获取时间、空间、凸多面体个数、四面体个数、新点比例等项数据。其中,四面体个数是对所有凸多面体再进行四面体化后的结果个数,它是一个估算的结果。根据文献[15],对于任何多面体,设顶点数为  $n$ ,四面体化需要的新边数为  $e$ ,四面体个数为  $t$ ,则有  $t=e+n-3$ 。当  $e=0$  时,得到这个问题的下限,即  $t=n-3$ 。因此,我们可以根据这个公式估算实验中的最终四面体个数。测试结果如图 16 和图 17 所示。

从第 2 项实验的第 1 组模型的实验结果来看,新方法与旧方法相比,剖分时间下降为 1/15~1/121,空间下降为 1/2.3~1/7.4,新点数下降为 1/28~1/99,在多项指标上有数量级的提高。但是,由于新方法会剖分出大量四面体,因此凸多面体数量增加了 1 倍左右。但如果将所有凸多面体四面体化,则新方法得到的四面体总数仅为旧方法的 1/2 左右。由于目前针对复杂多面体(包含空洞)的四面体化算法通常是先进行凸剖分,然后再对凸多面体进行四面体剖分<sup>[12]</sup>,因此,新方法非常有利于进一步的四面体化。

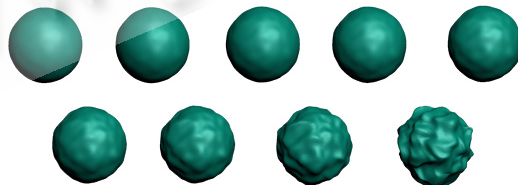


Fig. 15 NoiseSphere models used by the second experiment. The ratio of reflex edges ranges from 5% to 45%

图 15 用于第 2 组实验的 NoiseSphere 模型,凹边比例从 5%~45%

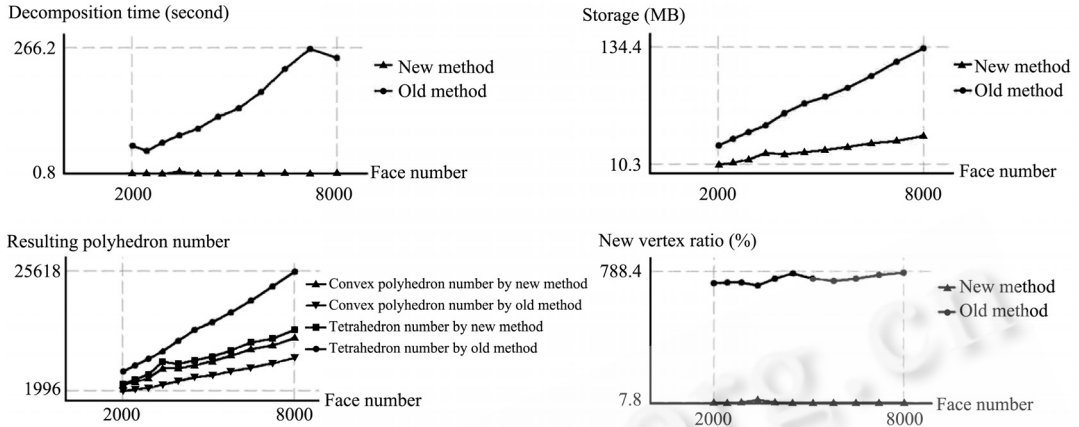


Fig.16 Test results of NoiseSphere models whose reflex edge ratio is 35% and the face number ranges from 2000 to 8000

图 16 凹边比例 35%,面片数从 2000~8000 的 NoiseSphere 模型组测试结果

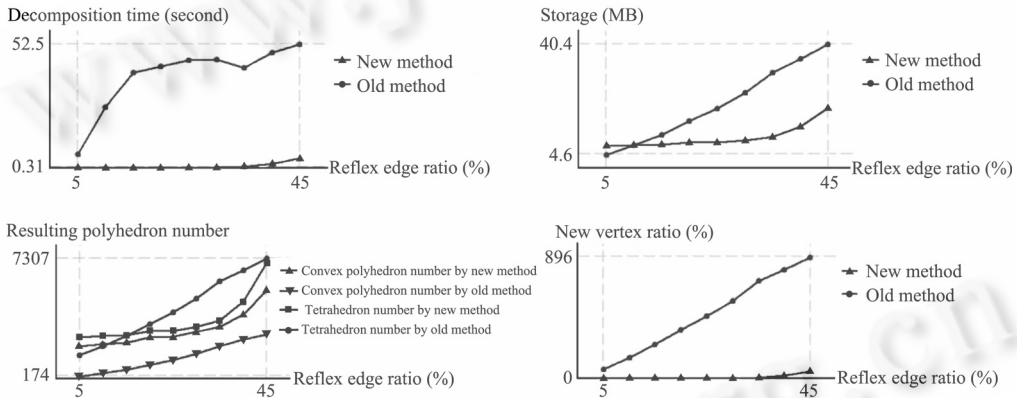


Fig.17 Test results of NoiseSphere models whose concave edge number is 2000 and the ratio of concave edges ranges from 5% to 45%

图 17 面片数 2000,凹边比例从 5%~45%的 NoiseSphere 模型组测试结果

从第 2 项实验的第 2 组模型的测试数据来看,在不同的凹边比例下,新方法和旧方法相比都具有相似的性质——速度大幅提高、新点数大幅下降、空间明显下降,凸剖分后凸多面体数量明显增加,完全四面体化后四面体数量明显减少.在除“生成凸多面体个数”外的所有指标中,新、旧算法之间的差距均随凹边比例的变大而趋向增加,这是因为旧方法对凹边数量的敏感程度要明显超过新方法.另外,从数据中可以看出,当凹边比例较低时,新方法有可能完全不产生或者仅生成极少个新点.例如,在这组实验中,当凹边比例为 5%、10%、15%时,新点总数分别为 0,3 和 8.

为测试新方法对于复杂模型的剖分能力,我们测试了国际上常见的 5 个模型——Bunny,Dragon,Horse,Buddha,Cow,如图 18 所示.由于原始模型并非简单多面体,它们存在着不封闭、自相交等现象,无法进行剖分,因此我们使用了来自 <http://www.cs.wustl.edu/~taoju/code/polymender.htm> 的模型处理工具 Polymender 对它们进行了规整化处理,使之成为标准的简单多面体.由于该工具在处理过程中可能会删除一些纠结在一起无法矫正的面片,因此上面图中,马、牛关节处都有所缺失,但这并不影响测试结果.测试结果见表 1.

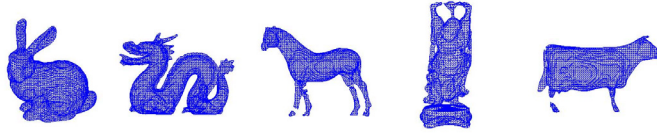


Fig.18 Common models including bunny, dragon, horse, happy buddha and cow

图 18 兔、龙、马、佛、牛五大模型

Table 1 Statistics of the decomposition of complex models

表 1 复杂模型的剖分数据

	Total face number	Ratio of reflex edges (%)	Decomposition time (s)	Storage (MB)	Convex polyhedron number	Ratio of new added vertex (%)
Bunny	15 812	37.2	11.5	103.0	31 629	21.0
Dragon	19 944	38.3	70.9	181.6	49 627	53.4
Horse	10 568	38.9	9.6	67.4	20 473	23
Buddha	10 594	40.9	22.9	101.1	27 807	50.9
Cow	9 732	37.7	9.4	61.1	18 288	22.6

由上面的数据可知,对于实际应用的复杂模型,新方法同样可以正确剖分,而且速度、空间需求等指标与前  
面简单形状模型的剖分数据相比都基本相当,符合算法分析的结果。

## 6 结束语

当前的复杂多面体往往含有许多凹边。为了提高这些复杂多面体的凸剖分效率,并尽量减少剖分产生的新  
点,本文提出了一种新的凸剖分方法。新方法主要是根据多面体沿某个方向正投影时它的边与边之间、面与面  
之间的遮挡关系,尽可能地进行局部操作以渐进地凸剖分多面体。它能够方便地处理包含空洞的多面体,甚至  
是包含孤立面、孤立边和孤立点的非流形多面体。经算法分析,并对包括Bunny,Dragon,Happy Buddha等复杂模型  
在内的各种多面体模型进行实验后可知:新方法的时间复杂度、空间复杂度皆近似为 $O(n)$ ,而新点数不超过  
 $O(r+n^{0.5})$ ,这些均优于已有的方法。在实验中,与目前国际上常用的“切割分裂”方法相比,新方法的速度提高了  
14~120倍,空间下降为1/2.3~1/7.4,而新增的点数则最多为“切割分裂”方法的1/28,甚至有些情况下不必增加新  
点即能完成凸剖分。另外,由于新方法剖分出的凸多面体绝大多数是四面体,因此剖分出的凸多面体数量近似为  
 $O(n)$ ,个数要多于“切割分裂”方法。但是,将两种方法剖分得到的凸多面体进一步四面体化,则新方法的结果个数  
将明显少于切割方法,这对很多需要四面体化的应用是有益的。新方法降低了剖分效率对凹边数量的敏感性,在  
处理凹边很多的复杂多面体时具有更高的效率。

## References:

- [1] Ehmann SA, Lin MC. Accelerated proximity queries between convex polyhedra by multi-level voronoi marching. In: Proc. of the Int'l Conf. on Intelligent Robots and Systems. IEEE Computer Society, 2000. 2101–2106. <http://ieeexplore.ieee.org/Xplore/login.jsp?url=/iel5/7177/19356/00895281.pdf?arnumber=895281>
- [2] Hershberger JE, Snoeyink JS. Erased arrangements of lines and convex decompositions of polyhedra. Computational Geometry: Theory and Applications, 1998,9(3):129–143.
- [3] Feng HYF, Pavlidis T. Decomposition of polygons into simpler components: Feature generation for syntactic pattern recognition. IEEE Trans. on Computers, 1975,24(6):636–650
- [4] Lu Y, Gadh R, Tautges TJ. Volume decomposition and feature recognition for hexahedral mesh generation. In: Proc. of the 8th Int'l Meshing Roundtable. 1999. 269–280.
- [5] O'Rourke J, Supowit KJ. Some NP-hard polygon decomposition problems. IEEE Trans. on Information Theory, 1983,29(2): 181–190.
- [6] Chazelle B. Convex decompositions of polyhedra. In: Proc. of the 13th Annual ACM Symp. on Theory of Computing. 1981. 70–79. <http://doi.acm.org/10.1145/800076.802459>



- [7] Chazelle B. Convex partitions of polyhedra: A lower bound and worst-case optimal algorithm. *SIAM Journal on Computing*, 1984, 13(3):488–507.
- [8] Bajaj CL, Dey TK. Convex decomposition of polyhedra and robustness. *SIAM Journal on Computing*, 1992,21(2):339–364
- [9] Lien J, Amato NM. Approximate CONVEX DECOMPOSITION. Technical Report, TR03-001, Parasol Lab., Texas A&M University, 2003.
- [10] de Berg M, van Kreveld M, Overmans M, Schwarzkopf O. Polygon triangulation. In: *Computational Geometry: Algorithms and Applications*, 2nd rev. ed. Berlin: Springer-Verlag, 2000. 45–61. [http://www.amazon.com/gp/reader/3540656200/ref=sib\\_fs\\_top/102-5730750-3280164?ie=UTF8&p=S00H&checksum=nu5Y6vfC4UFG38VkTzuqQSwvuK6HLGbk8117Ficiys%3D#reader-link](http://www.amazon.com/gp/reader/3540656200/ref=sib_fs_top/102-5730750-3280164?ie=UTF8&p=S00H&checksum=nu5Y6vfC4UFG38VkTzuqQSwvuK6HLGbk8117Ficiys%3D#reader-link)
- [11] Chazelle B, Palios L. Triangulating a non-convex polytope. *Discrete & Computational Geometry*, 1990,5(1):505–526.
- [12] Dey T. Triangulation and CSG representation of polyhedra with arbitrary genus. In: *Proc. of the 7th ACM Symp. Computational Geometry*. 1991. 364–372. <http://doi.acm.org/10.1145/109648.109689>
- [13] Rupert J, Seidel R. On the difficulty of tetrahedralizing 3-dimensional non-convex polyhedra. *Discrete and Computational Geometry*, 1992,7(1):227–253.
- [14] Below A, de Loera JA, Richter-Gebert J. Finding minimal triangulations of convex 3-polytopes is NP-hard. In: *Proc. of the 11th Annual ACM-SIAM Symp. on Discrete Algorithms*. Society for Industrial and Applied Mathematics, 2000. 65–66.
- [15] Chin FYL, Fung SYP, Wang CA. Approximation for minimum triangulation of convex polyhedra. In: *Proc. of the 12th Annual ACM-SIAM Symp. on Discrete Algorithms*. Society for Industrial and Applied Mathematics, 2001. 128–137.
- [16] Fung SPY, Chin FYL, Poon CK. Approximating the minimum triangulation of convex 3-polytopes with bounded degrees. *Computational Geometry: Theory and Applications*, 2005,32(1):1–12.
- [17] Jing Li. Research on several fundamental problems in computer graphics and ray tracing techniques [Ph.D. Thesis]. Beijing: Graduate University, the Chinese Academy of Sciences, 2007 (in Chinese with English abstract).
- [18] de Berg M, Katz J, Frank van der Stappen A, Vleugels J. Realistic input models for geometric. Algorithms. In: *Proc. of the 13th Annual Symp. on Computational Geometry*. 1997. 294–303. <http://doi.acm.org/10.1145/262839.262986>

#### 附中文参考文献:

- [17] 李静. 计算机图形学中若干基本问题及光线跟踪技术的研究[博士学位论文]. 北京: 中国科学院研究生院, 2007.



李静(1972—),女,北京人,博士,助理研究员,主要研究领域为计算机图形学,计算几何.



吴恩华(1947—),男,博士,教授,博士生导师,CCF高级会员,主要研究领域为真实感绘制,虚拟现实,科学计算可视化.



王文成(1967—),男,博士,研究员,CCF高级会员,主要研究领域为科学计算可视化,虚拟现实,计算机图形学.