

# 四面体网格生成方法的研究与实现

田素垒, 张志毅, 陈敏, 张娴

(西北农林科技大学 信息工程学院, 陕西 杨凌 712100)

**摘要:** 针对四面体网格生成过程中需要人工构造背景网格和指定尺寸信息的问题, 提出了一种自动计算网格尺寸的方法。该方法通过按层次推进产生四面体网格, 根据周围前沿面的尺寸和法线信息, 计算新生成四面体的尺寸, 使四面体网格在尺寸上能够均匀分布。在网格生成过程中, 无需人工指定相关信息, 并能保证新生成四面体的质量。联立直线和平面参数方程, 根据方程组解的情况判断线段和三角形是否相交, 并对相交的条件作了详细的分析。使用空间八叉树管理前沿面, 降低与前沿面相关操作的复杂度。数值算例表明, 该方法能够生成较高质量的四面体网格。

**关键词:** 四面体网格; 层次推进; 尺寸计算; 相交测试; 八叉树

**中图分类号:** TP391.41 **文献标识码:** A **文章编号:** 1000-7024 (2012) 11-4416-06

## Research and implementation of methods for tetrahedral mesh generation

TIAN Su-lei, ZHANG Zhi-yi, CHEN Min, ZHANG Xian

(College of Information Engineering, Northwest Agriculture and Forest University, Yangling 712100, China)

**Abstract:** To address the issue that background mesh is constructed and mesh size is specified manually in the process of generation, a method of automatically calculating mesh size is proposed. Though tetrahedral mesh generation by each layer based on the surrounding front and normal information, the size of tetrahedron is calculated and smooth mesh is generated. In the process of mesh generation, related information specified by manually is unnecessary and the quality is also guaranteed. Though simultaneous parameter equations of the line and plane, the relationship between a segment and a triangle is judged. The intersection condition is given in detail. Octree space division is employed to manage the front, which reduce complexity of operations associated with front. The examples demonstrate that high quality tetrahedral mesh is generated.

**Key words:** tetrahedral mesh; layer; size calculation; intersection test; octree space division

### 0 引言

多面体网格在医学虚拟手术、地质结构仿真、工程数值模拟等领域存在着广泛应用。网格化是将表面模型所包围的连续空间离散成有限个单元的过程。网格化时选用四面体单元能够很好的逼近模型的边界。在四面体网格生成算法中, 比较有代表性的有 Delaunay 方法和前沿推进方法<sup>[1-2]</sup>。Delaunay 方法基于空球准则生成四面体网格, 具有严格的数学理论作为基础, 能够生成高质量的网格, 速度较快。该方法的主要问题为边界恢复, 因为利用空球准则生成的四面体网格是一个凸包, 并不保持表面模型的原始边界。对于表面形状较复杂的模型, 边界恢复算法也很复

杂<sup>[3-5]</sup>。前沿推进算法以待剖分区域的表面三角形作为初始前沿面, 通过插入点, 向内部推进, 生成四面体网格。算法在插入点的同时能够生成四面体, 从而能够灵活的指定四面体的尺寸和形状。该算法的优点是剖分区域的表面拓扑形状被自动保持, 不需要进行边界恢复, 主要缺点是在网格生成的过程中, 为了得到高质量的网格, 需要人工构造背景网格, 存储四面体单元的尺寸信息, 并且需要大量添加、删除、查找、相交检测等操作, 这些操作占用了算法的大部分时间<sup>[6-8]</sup>。

在上述的研究基础上, 本文针对前沿推进算法中需要人工指定相关信息的问题, 提出了一种自动计算新生成四面体尺寸的方法, 给出了实现算法的具体步骤。数值算例表明,

收稿日期: 2011-11-22; 修订日期: 2012-01-25

基金项目: 教育部留学回国人员科研基金项目 (K314020901); 中央高校科研业务费专项基金项目 (Z109021004)

作者简介: 田素垒 (1987-), 男, 山东巨野人, 硕士研究生, 研究方向为计算机图形学; 张志毅 (1974-), 男, 山西夏县人, 博士, 副教授, 研究方向为计算机图形学; 陈敏 (1986-), 女, 陕西扶风人, 硕士研究生, 研究方向为计算机图形学; 张娴 (1987-), 女, 贵州贵阳人, 硕士研究生, 研究方向为计算机图形学。E-mail: suleitian0805@163.com

该方法能够满足实际需求。

## 1 四面体网格的生成

前沿推进算法以实体表面三角形网格作为算法的输入, 生成模型所占区域的四面体网格。算法是一个通过插入点不断向模型内部推进、反复迭代的过程, 直到整个前沿面队列为空, 算法的基本流程如下:

(1) 初始化表面三角形网格, 使每个三角形都有指向实体内部的法线。

(2) 以表面三角形网格初始化前沿面队列, 进入下述反复迭代过程, 直到前沿面队列为空: ①从前沿面队列中取出一前沿面作为当前面, 计算最优点; ②取得最优点临近空间的前沿点(在前沿面上的点)作为候选点, 按照一定的顺序放入候选队列, 最优点放到队列的最后; ③在候选队列中找到一个通过有效性检查的点, 使该点和当前面组成一个四面体单元; ④更新前沿面队列; ⑤检查前沿面队列是否为空; 若为空, 则算法结束, 否则跳转到步骤①。

### 1.1 初始化实体表面三角形网格

在生成四面体网格前, 需要对实体表面三角形进行初始化:

(1) 计算表面法线。四面体必须朝着实体内部的方向生成, 因此需要计算每一个三角形指向实体内部的法线。表面三角形网格必须是定向的, 即从实体内部观察每一个三角形, 三个顶点都按一个方向环绕, 逆时针或顺时针。假设三个顶点都按逆时针环绕(由模型内部向外观察), 如图1所示, 则 $\triangle ABC$ 指向实体内部的法向量 $n$ 可通过向量 $AB$ 与 $AC$ 的向量积获得, 即

$$n = \overline{AB} \times \overline{AC}$$

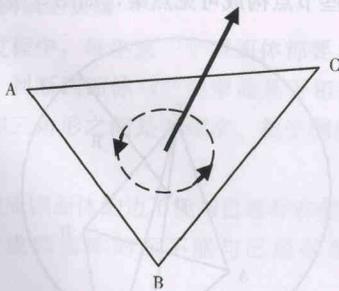


图1 三角形的朝向

(2) 封闭性检查, 即模型的表面三角形是否围成三维空间封闭的空间。如果表面三角形的每一条边都被两个三角形共享, 说明模型是封闭的, 否则不封闭, 四面体剖分算法无法进行。

### 1.2 四面体形状评价

四面体形状和尺寸对数值模拟等应用有重要的影响。一般而言, 四面体的形状越接近正四面体, 四面体的尺寸

分布越均匀, 或均匀过度, 模拟精度越高<sup>[9-10]</sup>。

通过式(1)评价四面体的质量

$$Q = \frac{|G-S|}{R} \quad (1)$$

式中:  $Q$ ——四面体的质量系数,  $G$ ——几何重心,  $S$ ——四面体外接球的球心,  $R$ ——外接球的半径, 即质量系数为四面体几何重心、外接球球心之间的距离与外接球半径的比值, 其值在0到1之间, 该值越小, 表示四面体的质量越高。当该值为0时, 四面体为正四面体。通过质量评价公式, 监督网格的生成过程, 使四面体的质量系数尽可能的高。

### 1.3 前沿推进与尺寸控制

在传统的前沿推进算法中, 为了得到尺寸均匀的四面体网格, 一般采用背景网格法。背景网格可使用结构化或非结构化的网格, 其必须覆盖表面三角形网格所占的空间, 并将要新生成的四面体尺寸存储在背景网格单元的节点上<sup>[11]</sup>。背景网格单元可为四面体、八面体等形状。背景网格生成以后, 对表面模型所占区域内任意一点处的四面体尺寸, 可通过如下方法得到(以背景网格单元为四面体为例):

(1) 找出包含该点的背景单元网格。

(2) 对该点的尺寸进行线性插值, 插值公式为

$$h_P = (h_D \times V_{ABCP} + h_C \times V_{ADBP} + h_B \times V_{ACDP} + h_A \times V_{BDPC}) / V_{ABCD} \quad (2)$$

式中:  $h_P$ ——插值点 $P$ 处的尺寸,  $h_A$ 、 $h_B$ 、 $h_C$ 、 $h_D$ ——背景网格单元4个顶点 $A$ 、 $B$ 、 $C$ 、 $D$ 处的尺寸,  $V_{ABCD}$ ——该四面体的体积,  $V_{ABCP}$ 、 $V_{ADBP}$ 、 $V_{ACDP}$ 、 $V_{BDPC}$ ——点 $P$ 与四面体的4个面所构成的四面体的体积。背景网格法通过插值得到了均匀变化的尺寸信息, 但需要人工的构造一个背景网格, 且需要指定节点处的尺寸信息, 尤其是背景网格单元较多时, 会导致算法的自动化程度不高。本文采用按层推进的方式, 对表面模型进行四面体剖分, 新生成四面体的尺寸由当前面附近的前沿面决定, 无需构造背景网格, 能够生成局部尺寸均匀分布、全局尺寸均匀过度的网格。

采用按层推进对模型进行网格化时, 四面体网格由表面模型的三角形网格开始, 逐层向实体的内部推进生成<sup>[12]</sup>。要实现按层次生成四面体网格, 需要对前沿面进行分类: 在第 $n+1$ 层网格生成中, 能够被选中向内部推进生成新四面体的前沿面为活跃前沿面, 其为第 $n$ 层四面体的三角形面; 在生成第 $n+1$ 层四面体时, 而新生成的三角形为非活跃前沿面, 不能用来推进生成新四面体。两类前沿面交替使用: 当第 $n+1$ 层四面体完全生成之后, 非活跃前沿面成为可继续向内推进的活跃前沿面。为了得到均匀变化的网格, 在生成第 $n+1$ 层四面体时, 使用第 $n$ 层四面体的尺寸和法向量信息。在整个剖分过程中需要动态的计算与第 $n$ 层前沿面上的节点 $N_i$ 相关以下数据:

- (1) 参照高度  $h_i$  : 用来计算新生成四面体的高度。
- (2) 单位法向量  $\bar{N}_i$  : 用来计算网格推进方向。
- (3) 所连面的集合  $F_i$  : 记录当前的连接关系。
- (4) 搜索半径  $R_i$  : 控节点制影响区域的大小。

参照高度的计算方法为

$$h_i = 1.24 \sqrt{\bar{a}_i} \quad (3)$$

$$\bar{a}_i = \sum_{k=1}^m a_k / m \quad (4)$$

式(3)中,  $h_i$  表示节点处四面体的高度(公式中的常数来自于正四面体的情形: 高度为  $2\sqrt{2}\sqrt{3}/3$  与底面面积的乘积)。式(4)中,  $a_k$  为节点所连接的第  $k$  个面的面积,  $m$  为所连接的面数。 $\bar{a}_i$  为与节点  $N_i$  相连的  $m$  个前沿面的平均面积。 $\bar{N}_i$  的单位法向量的计算方法为

$$\bar{N}_i = \frac{\sum_{k=1}^m \bar{N}_k}{\left| \sum_{k=1}^m \bar{N}_k \right|} \quad (5)$$

式中:  $\bar{N}_k$  ——节点所连接的第  $k$  个面的单位法向量。  
 $\bar{N}_i$  ——与点  $N_i$  相连的  $m$  个前沿面法向量的平均值, 表示这些面的推进方向。为了得到均匀的网格, 考虑周边区域对参考高度的影响。对于节点  $N_i$ , 定义一个球  $S(X_i, R_i)$  为影响区域, 球心在节点处, 半径为节点所连的最长边的3倍。搜索位于影响区域内的前沿面上的点, 若节点  $N_j$  位于  $N_i$  的影响区域内, 且满足条件

$$\cos^{-1}(\bar{N}_i \cdot \bar{N}_j) > 60^\circ \quad (6)$$

则节点  $N_j$  影响  $N_i$  的参考高度, 影响因子可通过式(7)计算

$$c_{ji} = \cos \langle \bar{N}_i, \bar{N}_j \rangle (h_j - h_i) / h_j \quad (7)$$

位于影响区域内的所有的前沿面节点对  $N_i$  的参考高度的影响因子通过式(8)计算

$$c_i = \sum_{k=1}^l c_{ki} / l \quad (8)$$

$l$  为位于影响区域内的节点且满足式(6)的节点个数。通过网格的生成方向, 控制位于影响区域内的节点对新生成网格的影响: 当  $N_j$  与  $N_i$  之间的角度大于  $60^\circ$  时, 节点  $N_j$  对  $h_i$  无影响, 否则由式(7)计算对  $h_i$  的影响。节点  $N_i$  最终的参照高度通过式(9)计算

$$h'_i = (1 + c_i) h_i \quad (9)$$

在推进过程中, 从活跃队列中取出一前沿三角形  $\triangle ABC$ , 新生成四面体的最优尺寸通过式(10)计算得到

$$h = \frac{h_A + h_B + h_C}{3} \quad (10)$$

式中:  $h_A, h_B, h_C$  ——节点 A、B、C 处参照高度。四面体的最优点可根据最优尺寸通过式(11)计算

$$E = D + h \bar{N} \quad (11)$$

式中:  $D$  ——计算最优点的基准点, 为  $\triangle ABC$  的几何重心,  $\bar{N}$  —— $\triangle ABC$  的单位法向量, 如图2所示。

在网格生成过程中, 通过影响区域内的尺寸和法向量

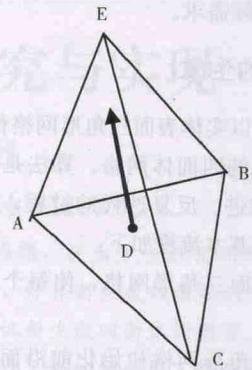


图2 计算最优点 E

信息, 第  $n$  层四面体影响第  $n+1$  层的尺寸, 保证最终得到尺寸均匀的网格。

算法在实现过程中维持 3 个队列: 活跃前沿面队列  $Q_C$ , 初始化为表面三角形; 非活跃前沿面队列  $Q_N$ , 初始化为空; 候选节点队列  $Q_{node}$ , 初始化为空。活跃前沿面队列中的每一个前沿面向实体内部推进, 算法如下:

(1) 从活跃前沿面队列  $Q_C$  中取一三角形记为  $\triangle ABC$  作为当前面。通过式(10)计算四面体的参照高度  $h$  和最优点 E 的位置, 并清空候选节点队列  $Q_{node}$ 。

(2) 构造可见点集。可见点集是当前面  $\triangle ABC$  附近空间前沿面上的节点。计算方法如下:

获取  $\triangle ABC$  的 3 个节点的搜索半径  $r_A, r_B, r_C$  分别为节点 A、B、C 所连最长边的长度。构造球  $S(O, R), R = \max(r_A, r_B, r_C)$ 。  $O = 0.4 \times D + 0.6 \times E$ ,  $D$  是  $\triangle ABC$  的几何重心。

搜索位于活跃前沿面队列  $Q_C$  中, 且落入球  $S(O, R)$  中的节点, 这些节点构成可见点集, 如图3所示。

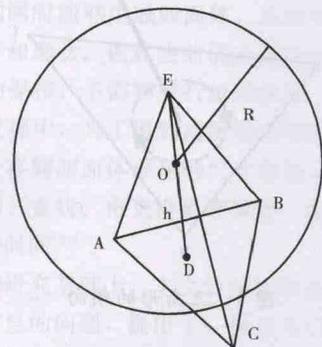


图3 候选点在球  $S(O, R)$  的内部

(3) 构造候选节点队列。根据四面体质量优先和已有节点优先的原则, 对位于可见点集中、且位于  $\triangle ABC$  上方节点(法线所指的一侧), 通过式(1)计算节点与  $\triangle ABC$  所形成四面体的质量系数, 按照新生成四面体质量的高低进行排序, 逐一放入候选队列  $Q_{node}$  中, 并将最优点放到队

列的最后。

在搜索候选节点时, 可以通过构造球  $S(O, R)$  的外接立方体包围盒, 加快搜索候选节点的过程。首先取得落在包围盒内的点, 再判断其是否在球  $S$  的内部。

(4) 在候选节点中查找满足要求的点。从候选队列中取一点  $P$ , 与当前面  $\triangle ABC$  构成一个新四面体  $ABCP$ 。四面体网格中, 任意两个四面体或者共享一个点, 或共享一条线段, 或共享一个三角形面。四面体  $ABCP$  可能与已经存在的四面体相交, 所以需要每一个新生成的四面体进行相交性测试。若点  $P$  没有通过相交性测试, 则从候选队列中选取下一个点进行相交性测试, 直到候选队列为空。

(5) 更新活跃前沿面、非活跃前沿面队列, 并计算新生成三角形指向四面体外部的法线。若通过相交性测试的点  $P$  为前沿面上的点, 则新生成的三角形  $\triangle ABP$ 、 $\triangle BCP$ 、 $\triangle CAP$  可能在活跃前沿面队列  $Q_c$  和非活跃前沿面队列  $Q_n$  中已经存在。因为在四面体网格中一个三角形至多被两个四面体共享, 如果新生成的三角形, 已经在前沿队列中存在, 说明此三角形被新形成的四面体和已经存在的四面体共享, 不能再作为前沿面向区域内部推进生成四面体, 需要将其从前沿队列中删除。如果不存在, 则将三角形放入非活跃前沿队列  $Q_n$  中。若通过相交性测试的点为最优点  $E$ , 则直接将三角形  $\triangle ABE$ 、 $\triangle BCE$ 、 $\triangle CAE$  放入非活跃前沿面队列  $Q_n$  中。在步骤 (1)、(3) 中都需要前沿面的法线信息, 因此, 每生成一个四面体, 都需要计算新生成的三角形指向四面体外部的法线。

步骤 (1) 一步骤 (5) 重复执行, 当活跃前沿面队列  $Q_c$  为空时, 激活非活跃前沿面队列  $Q_n$ , 使之成为活跃前沿面队列, 继续向实体内部推进, 直到两队列都为空。

### 1.4 相交性测试的实现

在推进过程中, 每生成一个四面体都要与前沿面进行相交性测试。判断四面体与三角形面是否相交, 可以转化成判断线段和三角形之间是否相交。相交测试可以分为两种情况:

- (1) 新生成四面体的边不能与已经存在的前沿面相交。
- (2) 新生成四面体的面不能与已经存在的前沿面的边相交。

通过联立直线和平面的参数方程, 可以判断线段和三角形的相交情况。直线的参数方程为

$$tJ_a + (1-t)J_b \quad t \in R$$

$$J_a = (x_a, y_a, z_a), J_b = (x_b, y_b, z_b) \quad (12)$$

式中:  $J_a$ 、 $J_b$  —— 线段的端点。

平面的参数方程为

$$uP_0 + vP_1 + (1-u-v)P_2$$

$$w = 1 - u - v$$

$$u, v \in R$$

$$P_k = (x_k, y_k, z_k) \quad k = 0, 1, 2 \quad (13)$$

式中:  $P_0$ 、 $P_1$ 、 $P_2$  —— 三角形的顶点。联立两参数方程, 可得到方程 (14)

$$tJ_a + (1-t)J_b = uP_0 + vP_1 + (1-u-v)P_2 \quad (14)$$

将点的坐标代入上式, 整理可得 3 阶线性方程组

$$\begin{bmatrix} x_b - x_a & x_0 - x_2 & x_1 - x_2 \\ y_b - y_a & y_0 - y_2 & y_1 - y_2 \\ z_b - z_a & z_0 - z_2 & z_1 - z_2 \end{bmatrix} \begin{bmatrix} t \\ u \\ v \end{bmatrix} = \begin{bmatrix} x_b - x_2 \\ y_b - y_2 \\ z_b - z_2 \end{bmatrix}$$

通过方程组解的情况, 可以判断线段和三角形之间的位置关系:

- (1) 若方程组无解, 则表明线段所在的直线平行于三角形所在的平面。
- (2) 若方程组有无数组解, 则表明线段所在的直线位于三角形所在的平面上。此时, 判断线段和三角形是否相交可转换成线段和线段是否相交。
- (3) 若方程组有唯一解, 则表明线段所在的直线和三角形所在的平面相交。可分为如下 4 种情况。

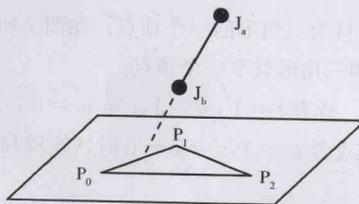


图 4 线段和三角形不相交

- 1) 线段和三角形不相交。

当  $t \notin [0, 1]$  时, 无论  $u, v, w$  取何值, 线段的延长线与三角形所在的平面相交, 线段和三角形都不会相交, 如图 4 所示。同理, 当  $u \notin [0, 1]$ , 或者  $v \notin [0, 1]$ , 或者  $w \notin [0, 1]$  时, 无论  $t$  取何值, 线段所在的直线与平面上三角形之外的区域相交, 线段和三角形都不会相交。

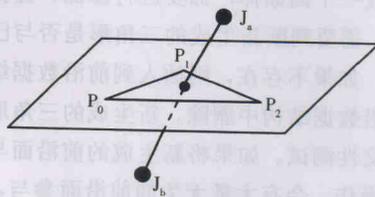


图 5 线段和三角形相交于内部

- 2) 线段和三角形相交与内部。

当  $t \in (0, 1), 0 < u, v, w < 1$  时, 线段和三角形相交, 交点在线段的内部, 并且在三角形的内部, 如图 5 所示。

- 3) 线段和三角形相交于边界处。

当  $t \in [0, 1], 0 < u, v < 1, w = 0$ , 或者  $0 < u, w < 1, v = 0$ , 或者  $0 < v, w < 1, u = 0$  时, 线段和三角形相交, 交点在线段内部, 并且在三角形一条边的内部。如图 6 所示。

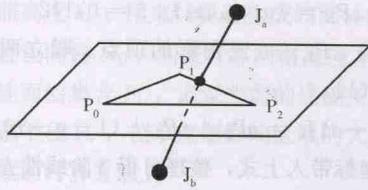


图6 线段和三角形的一条边相交

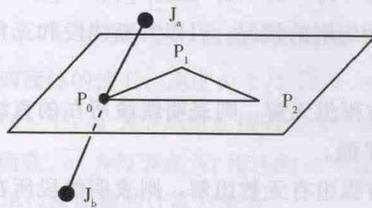


图7 线段和三角形的一个顶点相交

当  $t \in (0,1), u = 1, v = w = 0$ , 或者  $v = 1, u = w = 0$ , 或者  $w = 1, u = v = 0$ , 线段和三角形相交, 交点在线段的内部, 并且为三角形的一个顶点, 如图7所示。

4) 线段和三角形共享一个顶点。

当  $t = 0$ , 或者  $t = 1, u = 1, v = w = 0$ , 或者  $v = 1, u = w = 0$ , 或者  $w = 1, u = v = 0$  时, 线段和三角形共享一个顶点。

在四面体网格生成过程中, 情况1)和4)是允许的。情况4)发生时, 说明这个节点被若干个四面体共享。在相交检测算法实现过程中, 只需检测方程组的解是否满足情况1)和4)中的条件: 如果满足则新生成的四面体通过相交检测, 否则没有通过。

## 2 前沿面管理

在四面体网格生成过程中, 需要管理大量的前沿面三角形。每生成一个四面体, 都要进行添加、查找、删除和搜索等操作。需要判断新生成的三角形是否与已经存在的前沿面重复: 如果不存在, 则插入到前沿数据结构中, 否则需要从前沿数据结构中删除。新生成的三角形还需要与前沿面做相交性测试。如果将新生成的前沿面与所有的前沿面作上述操作, 会有大量无关的前沿面参与, 这在很大程度上降低了算法的效率。可通过空间八叉树管理前沿面, 减少上述操作的复杂度。构建模型的立方体包围盒, 以包围盒为根结点构建空间八叉树。在八叉树的每一个结点中存放其所包含的前沿面。在四面体网格生成过程中, 八叉树所包含的前沿面信息需要动态的更新。更新八叉树所包含的前沿面信息的过程如下:

(1) 确定前沿面所在的结点 (从根结点开始)。

1) 如果前沿面与8个子结点都相交, 则前沿面在根结点中, 否则进入以包含此前沿面的结点为根的八叉树中进

行判定。

2) 反复执行步骤1), 直到确定前沿面所在的最小的且与前沿面不相交的八叉树结点。

(2) 判断前沿面在结点中是否已经存在: 如果不存在则添加此前沿面, 否则从结点中删除此前沿面。

使用空间八叉树管理前沿面时, 相交性测试的过程如下:

(1) 判断新生成的前沿面所处的八叉树结点。

(2) 遍历以此结点为根的八叉树。新前沿面与八叉树中每个结点所包含的前沿面做相交性测试。

## 3 数值算例

本文算法在 VS2008 平台上由 C++ 语言实现, 计算环境为 2GHz 主频, 1G 内存的 PC 计算机。应用算法对人物头像、牛、球等模型进行四面剖分, 剖分结果如图8所示。左侧为表面模型, 右侧为四面体网格的剖视图。从图中可看出四面体网格尺寸在局部能够均匀分布, 在整体上能够均匀的过度。依据质量评价式(1)进行的四面体网格的质量系数统计, 结果如图9所示。统计结果表明约60%的四面体的质量系数小于0.4, 约80%的四面体的质量系数小于0.6, 说明算法能够生成高质量的四面体网格。表1是对网格单元的个数和剖分时间的统计, 结果表明平均一秒生成的四面体单元个数在700左右, 能够满足实际的需求。

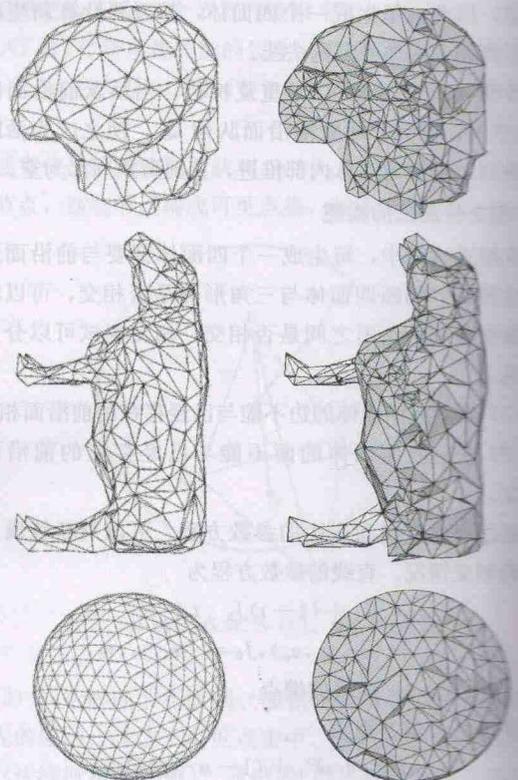


图8 表面模型及四面体网格剖视图

图9 四面体网格单元质量统计

表 1 时间统计

模型	节点数 N	单元数 E	剖分时间/s
头像	1157	5055	6.8
牛	1627	7839	12.3
球	1206	5791	7.8

#### 4 结束语

本文提出一种自动计算四面体单元尺寸、生成均匀四面体网格的方法。通过按层次推进产生四面体网格,利用周围前沿面的尺寸和法向量信息,控制新生成四面体尺寸的尺寸,达到生成均匀四面体网格的目的。使用直线和平面的参数方程对可能的相交情况作了详细的分析,给出了相交时的条件,并使用空间二叉树管理前沿面,减少相关操作的复杂度。算例表明算法可生成较高质量的网格,能够有效的满足实际的需要。

由于表面模型三角形网格会影响第一层四面体的质量,进而对实体内部其它层的四面体的质量产生影响,下一步将研究如何改善表面模型三角形网格的质量,进一步提高四面体网格的质量。

#### 参考文献:

- [1] Vladimir Liseikin. Grid generation methods [M]. Springer, Heidelberg, 2010.
- [2] CHENG S, Tamal Dey, Levine J. Theory of a practical delaunay meshing algorithm for a large class of domains [J]. Algorithms, Architecture and Information System security, 2008, 3 (12): 17-41.
- [3] YANG Y, YONG J, SUN J. An algorithm for tetrahedral mesh generation based on conforming constrained delaunay trahedralization [J]. Computer & graphic, 2005, 29 (4): 606-615.
- [4] CHENG S, Poon S. Three dimensional delaunay mesh generation [J]. Discrete Computational Geometry, 2006, 36 (3): 419-456.
- [5] DU Qiang, WANG Desheng. Recent progress in robust and quality delaunay mesh generation [J]. Computational and Applied Mathematics, 2006, 295 (1): 8-23.
- [6] Danilov A. Unstructured tetrahedral mesh generation technology [J]. Computational Mathematics and Mathematical Physics, 2010, 1 (50): 139-156.
- [7] Matthew Staten, Robert Kerr, Steven Owen, et al. Unconstrained plastering-hexahedral mesh generation via advancing front geometry decomposition [J]. Numerical Methods in Engineering, 2010, 81 (2): 135-171.
- [8] Foucault G, Cuillere J, Francois V, et al. An extension of the advancing front method to composite geometry [C]. Proceedings of the 16th International Meshing Roundtable, 2008: 287-314.
- [9] XU Kai, CHENG Zhiqian. Quality encoding for tetrahedral mesh optimization [J]. Computers & Graphics, 2009, 33 (3): 250-261.
- [10] Dompierre J, Vallet G, Labbe P, et al. An analysis of simplex shape measures for anisotropic meshes [J]. Computer Methods in Applied Mechanics and Engineering. 2005, 194 (48): 4895-4914.
- [11] Dardenne J, Valette S, Siauve N, et al. Variational tetrahedral mesh generation from discrete volume data [J]. Visual Computer, 2009, 25 (5): 401-410.
- [12] JIANG Hengyi, LI Shanqng, ZENG Jianmin, et al. A novel tetrahedral mesh generation algorithm for finite element analysis [J]. Advanced Materials Research, 2011, 189 (9): 545-548.