

# 四叉树法网格划分的数据结构及算法设计

孔铁全, 任钧国

(国防科技大学 航天与材料工程学院, 湖南 长沙 410073)



**摘要:**目前, 四叉树法在平面网格划分领域中被广泛地使用。本文应用数据结构描述了四叉树的程序实现过程, 并详细介绍了四叉树过程中切割单元的处理方法。最后通过算例证明了四叉树法具有快速自动生成质量较好的全四边形单元且很容易通过程序实现等优点。

**关键词:** 四叉树法; 切割单元; 数据结构

**中图分类号:** TP302

**文献标识码:** A

## 引言

四叉树法在平面网格划分领域中被广泛地使用, 它源于计算几何造型中一门十分重要的技术——空间离散技术。它的基本思想是: 取平面实体模型的外接正四边形, 把它等分成四个小的正四边形, 然后递归划分没达到网格划分精度要求且部分或全部包含在实体模型内的小正四边形。

## 1 四叉树法网格划分的数据结构设计

取剖分区域的外接正四边形, 用根结点表示。子结点表示父结点经过剖分后的子区域, 每个结点的度数或是 0 或是 4, 度数是结点的子结点个数。结点可分为三类: 第一类为黑结点, 表示此结点代表的四边形区域完全包含在分析域内; 第二类为白结点, 表示此结点代表的四边形区域全部在分析域外; 第三类为灰结点, 表示此结点代表的四边形区域部分包含于分析域内。结点设计为结构, 表示如下:

```
struct quadtree
{
    dot vertex[ 4 ], center[ 5 ];
    int order;
    int flag;
    struct quadtree *child[ 4 ], *parent; };

```

其中, vertex 为存储该四边形四个顶点数据的数组, center 为存储四边形中心和各边中点数据的数组, order 为该结点在整个剖分区域内的顺序码, flag 为结点的类型(0 代表白结点; 1 代表黑结点; 2 代表灰结点), child 为指向四个子结点的指针数组, 若是叶子结点则指针为零, parent 为指向父结点的指针, 若为根结点, 则指针为零。结点顺序的编码方法是根结点为 0, 子结点的顺序码是在其父结点顺序码的后面加上它在父结点中的位置码, 顺序码的位数为层数。点的结构定义为:

```
struct dot
{
    float x, y;
    int type; };

```

(x, y) 为点的坐标, type 为点的类型: 0 代表点在实体模型外, 1 代表点在实体模型内, 2 代表点在实体模型的边界上。

## 2 四叉树法网格划分的算法设计

### 2.1 树的生成

四叉树过程就是由根结点生成整个树结构从而达到对分析域进行剖分的过程。其算法采用递归调用, 表示如下:

if( 结点不是白结点)

if( 结点不满足精度要求)

{ for( $i=0, i<4, i++$ )

{ 生成四个子结点;

for( $i=0, i<4, i++$ )

子结点递归调用该算法, 继续生成子树; }

其中, 生成子结点即计算子结点所代表四边形的顶点、中心、各边中点坐标及类型、结点类型及顺序码。其过程为:

for( $i=0; i<4; i++$ )

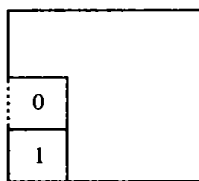
{ 申请一块 struct quadtree 大小的内存空间;

child[  $i$  ] 指向该空间;

填写新生成结构各项内容; }

## 2.2 结点类型的确定

根结点设为灰接点。其他结点类型的判断用它所代表的正四边形 4 个顶点的类型来近似判断。如果四边形的所有顶点都在分析域外, 则该结点为白结点; 如果部分顶点在分析域内、部分顶点在分析域外, 则该结点为灰结点; 如果所有顶点都在分析域内或边界上, 且至少有一个顶点为非边界上点, 则结点为黑结点。当所有顶点都在边界上, 结点可为白结点或黑结点, 如图 1 所示。此时可以通过判断它中心的类型来确定结点的类型, 中心在分析域外, 则为白结点, 否则为黑结点。



(0 表示单元为白节点, 1 表示单元为黑节点)

图1 结点分类示意图

## 3 四叉树法切割单元的处理

四叉树法划分后, 结点类型为 2 的单元被分析域边界切割, 这些单元称为切割单元。切割单元的形状可能很复杂, 而且有多种情况。为了简化切割后单元形状, 只考虑切割点在切割前小正四边形各边端点和中点的情况。切割单元根据被切割情况找与自己形状相近的情况替换。为了使网格划分后边

界与实体模型边界拟合得更好, 可考虑切割点在小正四边形各边上其他点甚至在四边形内的情况, 但它成倍增加了切割单元的形状, 也增加了把切割单元统一成同一类型单元的难度。表 1 是仅考虑实体模型与小正四边形在各边端点和中点切割的情况及修改结果。修改过程中有新节点产生, 新节点的选取要避免取在非切割单元的边上。

表 1 四边形切割情况表

情况	1	2	3	4	5
切割单元					
处理结果					
情况	6	7	8	9	10
切割单元					
处理结果					
情况	11	12	13	14	15
切割单元					
处理结果					

注:  $\odot$  为实体模型外点

表 2 选点转换成四边形单元处理表

取点	处理结果		
	如果A是实体模型内点	如果B是实体模型内点	如果A、B不是实体模型内点

表1中, 13、14、15三种切割情况, 由于剩下在分析域中的中点和顶点构不成二维几何形状, 因此近似地假定去掉该单元, 6、7、11、12四种切割情况, 修改后变为三角形单元, 可通过对三角形单元周围取点把它转换成四边形单元, 如表2所示, 对每种三角形单元在它周围选取了三个点, 判断三点是否在分析域内, 选取最接近实体边界的一点与三顶点构成一四边形单元。

切割单元处理后, 把切割单元的 flag 值置为 0, 建立一个单元链表, 把切割单元处理后得到的新单元存储到单元链表中。

## 4 单元遍历和节点遍历算法

四叉树生成后, 单元链表中单元和所有叶子结点才是最后的网格单元。因此, 要对四叉树进行遍历, 找到这些叶子结点, 把它们储存在单元链表中。其算法仍采用递归调用, 如下:

```
if ( 结点不为空)
{
    if ( 结点是叶子结点)
    {
        if( 结点不为白结点)
            存入单元链表; }
    else
    {
        for (i=0; i<4; i++)
            对子结点递归调用该算法; }}
```

生成最终单元链表后, 对单元进行遍历可得到节点的信息, 把节点信息储存到新建节点链表中。然后, 删除节点链表中的重复节点, 便得到最终节点链表, 其算法如下:

```
for ( 遍历单元链表中单元)
{
    for(i=0; i<4; i++)
        把单元的顶点存到节点链表中; }
    for(i=0; i<节点个数减1; i++)
    {
        for(j=i+1; j<节点个数; j++)
            if( 两节点重复)
                {j 后面的节点依次往前移一步;
                 节点个数减1; }}
```

## 5 举例

基于前面的四叉树法数据结构理论及算法, 利

用 C++ 编制了网格全自动生成程序, 只要给出剖分域的边界解析方程和网格划分的层数, 该程序就可自动生成全四边形单元网格。图2分别对一两边带半圆槽的长方形板和一中心带圆孔的长方形板进行了四叉树划分。

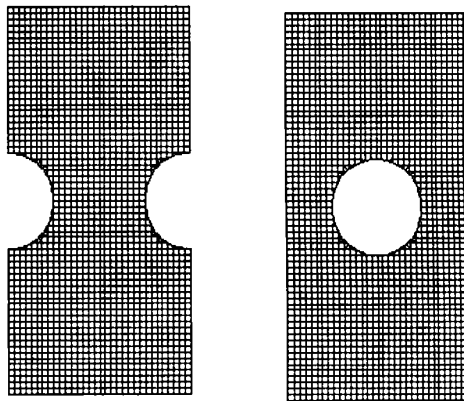


图2 四叉树网格划分举例

## 6 结束语

四叉树法是一种被广泛使用的平面网格划分方法, 它形成的单元形状好, 且边界适应性强, 很容易通过程序实现, 因此它在平面网格划分领域被广泛地使用。本文主要讲述了四叉树法的数据结构设计及网格划分的算法实现, 并详细列出了切割点在正四边形顶点和各边中点的切割情况及对切割单元的处理方法, 最终达到了对平面区域的网格划分。

参考文献:

- [1] 杨名生, 张立京. 基于四叉树的有限元网格自动剖分[J]. 大连理工大学学报, 1997, 37(5): 614—617.
- [2] 魏红宁, 周本宽. 自适应有限元分析的网格自动生成方法的选择[J]. 西安交通大学学报, 1997, 10, (5): 477—481.
- [3] MARK A YERRY, MARK S. SHEPHARD. Automatic Three-dimensional Mesh Generation by the Modified-octree technique[J]. Int J. numer. methods eng., 1984 (20): 1965—1990.

(下转第89页)

重用性、扩充性和易构性。最后介绍了应用文中提出的报表工具模型进行西安电子科技大学网络财务系统中报表工具的开发。

#### 参考文献:

[1] 王元珍, 汪皓. 达梦智能报表工具设计与实现[J]. 计

算机工程与应用. 2001, 4, (4): 65—67.

[2] 刘军强. 通用报表生成工具及其与 MIS 的集成[J]. 小型微型计算机系统. 1996, 5(17): 73—77.

[3] 万琳, 陈传波. 智能报表生成系统模型的研究与实现[J]. 计算机应用研究. 2000, 5(5): 25—26

## Study on Reporting Technologies and Implementation of a Reporting Tool Generator

LIU Zhuang, ZHANG Lei, ZHANG Li, JIN Yi-min

(Software Engineering Institute, Xidian University, Xi'an 710071, China)

**Abstract:** Currently, only a few reporting tools under UNIX, which support database application development, are available and they do not fulfill the developer's and user's needs well. We identify and discuss the three main problems that reporting tools have to address, which are report data, report format and report output. Then, we construct a reporting tool model. In our model, report data definition language (RDDL) and report format description language (RFDL) are used with interpreters to interpreter them. Last, we build a reporting tool under UNIX based on the model.

**Key words:** reporting tool; reporting tool model; report data definition language; report format description language

(上接第 84 页)

## Data Structure and Algorithm of Mesh Generation by Quadtree Approach

KONG Tie-quan, REN Jun-guo

(College of Aerospace and Materials Engineering, National University of Defence Technology, Changsha 410073, China)

**Abstract:** At the present time, quadtree approach is extensively used in the field of planar mesh generation. In this paper, data structure and algorithm of quadtree approach are described. It is a key step how to deal with dissected elements in mesh generation, so it is also described in detail at the end of this paper. At last, two examples indicate that quadtree approach can automatically generate planar structures with all—quadrangular and good quality elements in high speed, and it is easy to realize by program.

**Key words:** quadtree approach; dissected mesh; data structure