

文章编号: 1005—0523(2005)02—0018—04

OpenGL 中基于 NURBS 的圆弧形墙体窗体的三维建模研究

徐佳磊, 张维锦, 李海姣

(华东交通大学 土木建筑学院, 江西 南昌 330013)

摘要: 在 VC++6.0 开发平台下, 应用 OpenGL 图形库为建筑工程中弧形结构的三维建模提供了一种新的解决方法. 采用圆弧的二次非均匀有理 B 样条(NURBS)表示方法, 实现了圆弧墙体的绘制, 并根据 NURBS 曲面的裁剪方法在墙体上剪切出窗口.

关 键 词: 非均匀有理 B 样条; 圆弧曲面; 裁剪; 三维建模; OpenGL

中图分类号: TP391

文献标识码: A

1 引 言

在建筑造型设计中经常遇到许多由二次曲线与曲面表示的形状, 例如曲面墙体和窗体其中包含有自由型曲面和二次曲面. 建筑师为了丰富建筑的造型, 常把建筑屋面、墙面、窗户设计成各种曲面的形状, 然而更常用的是圆弧构件. 圆弧构件用 OpenGL 中三角形、四边形非常容易构造其三维模型, 但某一圆弧墙体中包含有若干个形状复杂的窗洞, 其三维模型的构造就比较复杂.

NURBS (Non—Uniform Rational B—spline) 已成为计算机辅助设计领域的曲线和曲面的标准描述方式, 圆弧曲线可以采用二次 NURBS 表示. OpenGL 的库函数中提供了 NURBS 函数接口, NURBS 函数能够比较容易地构造复杂的曲面, 并且能够在复杂的曲面上裁剪出窗口. 所以建筑工程中的圆弧墙体和窗体适合采用计算机辅助设计建模.

2 B 样条基函数和 NURBS 曲线曲面的定义

在区间 $[u_i, u_{i+k+1}]$ 上的第 k 次 B 样条定义为:

$$B_{i,0}(u) = \begin{cases} 1, & \text{若 } u_i \leq u \leq u_{i+1} \\ 0, & \text{其他} \end{cases}$$
$$B_{i,k}(u) = \frac{u - u_i}{u_{i+k} - u_i} B_{i,k-1}(u) + \frac{u_{i+k+1} - u}{u_{i+k+1} - u_{i+1}} B_{i+1,k-1}(u) \quad (1)$$

$B_{i,k}(u)$ 的双下标中第一下标 i 表示序号, 第二下标 k 表示 B 样条多项式的次数. 区间 $[u_i, u_{i+k+1}]$ 被称为 $B_{i,k}(u)$ 的支撑区间, 该区间所含节点的并集就是定义这一组 B 样条基函数的节点矢量 $U = [u_0, u_1, \dots, u_{n+k+1}]$. 节点矢量的概念在后面的有理样条中有重要的实际作用.

一条 k 次 NURBS 曲线可以表示为两个样条函数之比:

$$p(u) = \frac{\sum_{i=0}^n \omega_i d_i B_{i,k}(u)}{\sum_{i=0}^n \omega_i B_{i,k}(u)} \quad (2)$$

其中: $\omega_i (i=0, 1, L, n)$ 称为权因子 (weights); $d_i (i=0, 1, L, n)$ 表示一组控制点, 而且与权因子相联系, 一个特定的 ω_i 越大, 曲线越靠近该控制点 d_i ; k 表示曲线参数; $U = [u_0, u_1, \dots, u_{n+k+1}]$ 表示节点矢量. 需要注意的是: 节点矢量的个数 = 控制点个数 + 曲线次数 + 1.

相应的, 一张 $k \times 1$ 次 NURBS 曲面的有理分式

收稿日期: 2004—12—03

作者简介: 徐佳磊 (1981—), 男, 山东莱阳人, 硕士研究生, 研究方向: 计算机辅助设计.

表示如下:

$$p(u, v) = \frac{\sum_{i=0}^m \sum_{j=0}^n \omega_{i,j} d_{i,j} B_{i,k}(u) B_{j,l}(v)}{\sum_{i=0}^m \sum_{j=0}^n \omega_{i,j} B_{i,k}(u) B_{j,l}(v)} \quad (3)$$

$N_{i,k}(u)$, $i=0, 1, \dots, m$ 和 $N_{j,l}(v)$, $j=0, 1, \dots, n$ 分别为 u 向 k 次和 v 向 l 次的规范 B 样条基. $p(u, v)$ 可看成从参数空间 (u, v) 平面到三维空间 R^3 的一个映射.

3 圆弧曲线的二次 Nurbs 表示

圆弧曲线可以采用二次或高于二次的 NURBS 表示, 高次曲线要求更多的控制点与权因子来定义, 仅在特殊场合如构造组合曲线或生成某些复杂曲面时用到. 本文中选用二次 NURBS 表示圆弧曲线, 应用 NURBS 曲线的插入节点算法^[1], 可以推导出 $90^\circ < |\theta| \leq 180^\circ$ 圆弧曲线的权因子和控制点坐标, 如下所示:

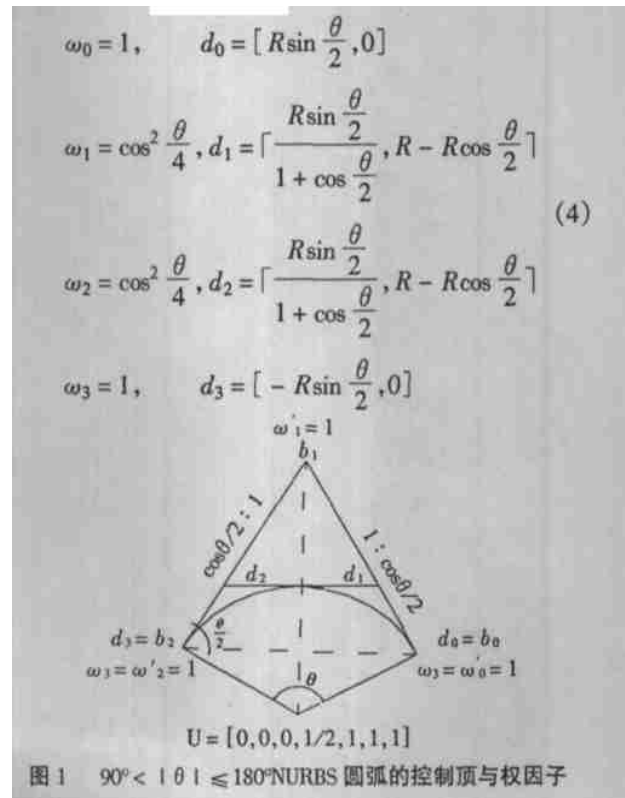


图1 $90^\circ < |\theta| \leq 180^\circ$ NURBS 圆弧的控制顶与权因子

4 应用 OpenGL 库函数绘制圆弧曲面墙体和窗体

OpenGL (Open Graphics Library), 是 SGI 公司开发的一个非常优秀的开放式、可独立于操作系统和硬件环境的三维图形软件库, 已经是事实上的高性能三维图形处理和交互式视景处理的工业标准.

OpenGL 在硬件、窗口、操作系统方面是相互独立的. 本文应用的是 WindowsXP 下 Visual C++6.0 的开发环境, 有关编程连接时的设置和 OpenGL 初试环境设置已有大量文献资料介绍, 在此不再赘述.

4.1 应用 Nurbs 函数绘制圆弧曲面墙体

OpenGL 辅助库中的 Nurbs 函数是一种强有力的曲面(线)绘制工具^[2], 其参数相当复杂, 运用该函数绘制 Nurbs 曲面(线)的具体操作步骤如下(曲线的绘制仅需要 u 方向的参数, 较为简单, 从略):

1) 首先创建一个要引用的 NURBS 对象:

```
GLUnurbsObj * pNurb; //NURBS 对象的指针
theNurb = gluNewNurbsRenderer();
```

2) NURBS 曲面的定义需要分别确定参数 u, v 方向的节点矢量、曲面阶数和控制点个数, 如下所示:

```
gluBeginSurface(pNurb); // 开始 NURBS 曲面绘制
```

```
gluNurbsSurface(pNurb, // NURBS 对象
GLint uknotCount, GLfloat * uKnots,
// 参数 u 方向的节点个数和节点矢量数组
GLint vknotCount, GLfloat * vKnots,
// 参数 v 方向的节点个数和节点矢量数组
GLint ustride,
// u 方向每两个相邻点之间的间距
GLint vstride,
// v 方向每两个相邻点之间的间距
GLfloat * ctrlArray,
// 控制点数组
GLint uorder, GLint vorder,
// u 和 v 方向的阶数
GLenum type);
// 曲面类型
gluEndSurface(pNurb);
// 结束 NURBS 曲面绘制
```

根据前面的 $90^\circ < |\theta| \leq 180^\circ$ 圆弧 NURBS 表示方法, 为了计算简便在此取 $\theta=120^\circ, R=2$, 建立图示坐标系.

求出 gluNurbsSurface 函数中各参数的值如下:

1) 三维空间中的控制点坐标数组可由(3)式求得:

```
GLfloat ctrlpoints[4][4][3] = {
```

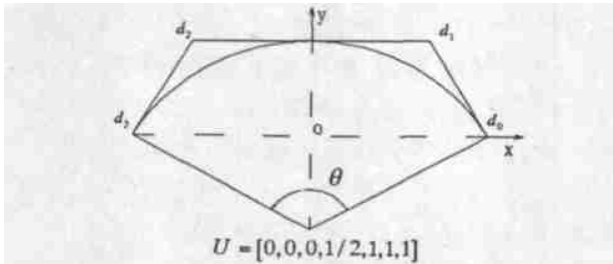


图2 $\theta = 120^\circ$ NURBS 圆弧的节点矢量和控制点

$\{-1.732, 0, 0\}, \{-1.155, 1, 0\}, \{1.155, 1, 0\}, \{1.732, 0, 0\},$
 $\{-1.732, 0, 2\}, \{-1.155, 1, 2\}, \{1.155, 1, 2\}, \{1.732, 0, 2\},$
 $\{-1.732, 0, 4\}, \{-1.155, 1, 4\}, \{1.155, 1, 4\}, \{1.732, 0, 4\},$
 $\{-1.732, 0, 6\}, \{-1.155, 1, 6\}, \{1.155, 1, 6\}, \{1.732, 0, 6\}\};$

2) u 方向的节点矢量 $\text{GLfloat uknots}[8] = \{0, 0, 0, 0, 1, 1, 1, 1\};$

3) v 方向的节点矢量 $\text{GLfloat vknots}[7] = \{0, 0, 0, 0.5, 1, 1, 1\};$

根据(2)式, 节点矢量个数=控制点个数+曲线次数+1; 即 节点矢量个数=控制点个数+阶数, 可以得出:

4) u 方向的阶数为 4;

5) v 方向的阶数为 3;

6) 曲面类型为: GL_MAP2_VERTEX_3

最后绘制的 $\theta = 120^\circ$ 圆弧曲面及其控制点如图:

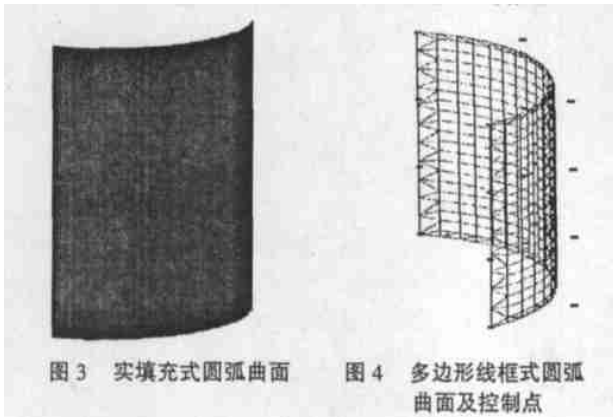


图3 实填充式圆弧曲面

图4 多边形线框式圆弧曲面及控制点

4.2 在墙体上裁剪出窗体

NURBS 裁剪方法在几何造型应用中范围广泛, 它可以快速地在曲面上构造窗口. NURBS 曲面的定义(3)式实际上是建立了从二维参数域到三维空间的映射关系^[3], NURBS 曲面裁剪的实质就是在保持映射关系不变的情况下, 改变曲面的有效参数

域^[4]. OpenGL 的库函数提供了 NURBS 曲面的修剪函数, 在绘制曲面的过程中, 也就是在 $\text{gluNurbsSurface}()$ 和 $\text{gluEndSurface}()$ 之间加入 NURBS 曲面修剪函数就可以实现对曲面的修剪.

定义裁剪曲面的裁剪曲线有两种方法^[5]: 1) 用 $\text{gluPwlCurve}()$ 产生分段的线性曲线; 2) 用 $\text{gluNurbsCurve}()$ 产生 NURBS 曲线. 分段线性曲线实际上是一系列直线, 它与 NURBS 相交时会在一定误差范围内出现偏差. 为了保证裁剪出的窗口与原曲面相切, 本文中定义的裁剪曲线为 NURBS 曲线, 而且 NURBS 曲线必须位于参数空间(u, v)的单位正方形内. 为保证结果的正确, 应当考虑裁剪曲线的方向性: 1) 外边界裁剪曲线应该是逆时针方向, 它的定义范围要包含整个 NURBS 参数空间; 2) 内边界裁剪曲线应该是顺时针的, 它的定义范围决定了裁剪区域即窗口的大小、位置和形状.

本文以裁剪矩形窗口为例, 分别取 4 段 NURBS 曲线作为裁剪曲线来对应矩形窗口的 4 条边. 每条曲线采用如下表达方式:

节点矢量 = $[0, 0, 1, 1]$, 控制点两个: 起点和终点; 曲线阶数 2 阶.

结果如图:

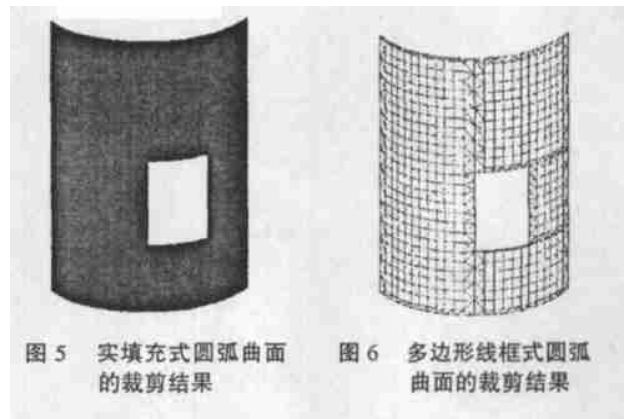


图5 实填充式圆弧曲面的裁剪结果

图6 多边形线框式圆弧曲面的裁剪结果

5 结 论

本文是将 Nurbs 方法应用于建筑工程领域三维建模的一次尝试, 应用圆弧曲线的 NURBS 表示方法, 通过 OpenGL 辅助库提供的 NURBS 绘制函数接口成功实现了 $\theta = 120^\circ$ 的圆弧曲面的建模, 同时应用 OpenGL 中的 NURBS 修剪函数在圆弧面上开窗口, 为今后复杂造型曲面的三维建模打下了基础.

参考文献:

[1] 施法中. 计算机辅助几何设计与非均匀有理 B 样条[M]. 北京:北京航空航天大学出版社, 1994.

[2] Richard S. Wright, Jr., Michael Sweet (潇湘工作室译) OpenGL 超级宝典[M]. 北京:人民邮电出版社, 2001.

[3] 唐敏,董金祥,何志均. 一个雕塑实体的布尔操作算法[J]. 计算机辅助设计与图形学学报, 1999, 11(5):459~463.

[4] 沈庆云, 周来水, 张乐年, 周儒荣. 一种 NURBS 曲面的裁剪方法[J]. 南京航空航天大学学报, 1997, (4).

[5] 吴斌,毕丽蕴. OpenGL 编程实例与技巧[M]. 北京:人民邮电出版社, 1999.

Study on the Cicular Arc-wall with Windows Modeling Based on Trimmed NURBS Through the Interface of OpenGL

XU Jia-lei, ZHANG Wei-jin, LI Hai-jiao

(School of Civil Engineering and Architecture, East China Jiaotong University, Nanchang 330013 , China)

Abstract: On the VC++6.0 platform, a new solution to constructing the three-dimenional model of architecture circular arc-surface was presented by applying Open Graphic Liabrary. An arc-surface wall was constructed through the method of representing circular arcs with quadratic NURBS, a window was also cut out by using trimmed NURBS.

Key words: NURBS; circular arc-surface; trimming; 3-dimensional modelling; OpenGL