

保凸均匀三次 B 样条插值曲线*

王文仲 方 逵

(国防科技大学系统工程与数学系 长沙 410073)

摘 要 均匀三次 B 样条曲线虽然具有保凸性,但曲线不通过任何控制顶点,我们在相邻两个控制点之间插入两个新的控制顶点后,所产生的新的均匀三次 B 样条曲线不但插值原来所有控制顶点,而且还保凸。本文描述的曲线可以作局部修改,给出了两个数值例子。

关键词 计算几何; B 样条曲线; 保凸插值

分类号 O242.7

B 样条曲线已广泛地应用于自由型曲线曲面设计, B 样条曲线曲面虽然能很方便地设计自由型曲线曲面,但用它设计插值曲线曲面是比较困难的。Yamagushi 用叠代法来近以计算插值 B 样条的控制顶点^[1],这种方法的缺点是插值曲线不精确,曲线的形状不能修改,Barsky 和谭建荣分别用反算的方法来计算控制顶点^[2,3],这种方法虽能精确的插值型值点,但计算费用较大,且曲线的形状不能局部修改。本文介绍一种具有保凸性的均匀三次 B 样条插值曲线,在每两个型值点之间插入两个控制点,这些控制点连同所有的型值点构成一个新的控制点列,由这一新的点列所产生的均匀三次 B 样条曲线插值所有的型值点,且曲线具有保凸性。

1 预备知识

给定 $m+1$ ($m \geq n$) 个空间点 \vec{r}_i ($i=0, 1, \dots, m$) 后,它对应的 n 次均匀 B 样条曲线可以分段表示为

$$\vec{r}_l(u) = \sum_{i=0}^n E_{i,n}(u) \vec{r}_{l+i}, \quad l = 0, 1, \dots, m-n$$

$$0 \leq u \leq 1, \quad (1)$$

式中 $E_{i,n}(u)$ 是均匀 n 次 B 样条基函数,且有

$$E_{i,n}(u) = \sum_{j=0}^{n-i} \frac{(-1)^j C_{n+1}^j (u+n-i-j)^n}{n!}, \quad i = 0, 1, \dots, n$$

特别当 $n=3$ 时,相应的基函数可表示为

$$E_{0,3}(u) = (-u^3 + 3u^2 - 3u + 1)/3!; \quad E_{1,3}(u) = (3u^3 - 6u^2 + 4)/3!,$$

$$E_{2,3}(u) = (-3u^3 + 3u^2 + 3u + 1)/3!; \quad E_{3,3}(u) = u^3/3!$$

* 1994年5月3日收稿

而对应的均匀三次 B 样条曲线是

$$\vec{r}_l(u) = \sum_{j=0}^3 E_{l,j,3}(u) \vec{r}_{l+j}, \quad l = 0, 1, \dots, n-3, \quad 0 \leq u \leq 1, \quad (2)$$

由 (2) 式容易求得均匀三次 B 样条曲线有如下的端点性质^[5]

$$\begin{aligned} \vec{r}_l(0) &= \frac{1}{6}(\vec{r}_l + 4\vec{r}_{l+1} + \vec{r}_{l+2}); & \vec{r}_l(1) &= \frac{1}{6}(\vec{r}_{l+1} + 4\vec{r}_{l+2} + \vec{r}_{l+3}), \\ \vec{r}_l'(0) &= \frac{1}{2}(\vec{r}_{l+2} - \vec{r}_l); & \vec{r}_l'(1) &= \frac{1}{2}(\vec{r}_{l+3} - \vec{r}_{l+1}). \end{aligned}$$

由上面的端点性质，立即得到下面的结论。

结论 若 $\vec{r}_l, \vec{r}_{l+1}, \vec{r}_{l+2}$ 三点共线，且有

$$\vec{r}_{l+1} = (\vec{r}_l + \vec{r}_{l+2})/2$$

则有

$$\vec{r}_{l-1}(1) = \vec{r}_l(0) = \vec{r}_{l+1}$$

即三次均匀 B 样条曲线插值 \vec{r}_{l+1} 。

2 保凸均匀三次 B 样条插值曲线的构造

在平面上给出一组有序点列 $\vec{V}_i (i=0, 1, \dots, n)$ ，把每相邻两个型值点用直线段连接起来形成一个多边形 $\langle \vec{V}_0 \vec{V}_1 \dots \vec{V}_n \rangle$ 。当多边形是凸的时，我们希望构造一条凸的均匀三次 B 样条插值曲线，并使这条曲线通过每一个点 \vec{V}_i 。

记多边形 $\langle \vec{V}_0 \vec{V}_1 \dots \vec{V}_n \rangle$ 的边矢量为 $\vec{a}_i = \vec{V}_i - \vec{V}_{i-1} (i=1, 2, \dots, n)$ ，

我们目的是构造均匀三次 B 样条曲线的控制顶点列 \vec{r}_i ，使由 \vec{r}_i 构成的控制多边形满足下面的两个条件：

- (1) 当多边形 $\langle \vec{V}_0, \vec{V}_1, \dots, \vec{V}_n \rangle$ 是凸的时，由 \vec{r}_i 构成的控制多边形也是凸的；
- (2) 由该控制多边形定义的均匀三次 B 样条曲线通过每一个型值点 \vec{V}_i 。

这样由 B 样条曲线的保凸性可以保证该均匀三次 B 样条插值曲线是保形的。

设待构造的保凸均匀三次 B 样条插值曲线在每个型值点 \vec{V}_i 处的切矢为

$$\begin{cases} \vec{T}_0 = t_0(-\vec{a}_2) + (1-t_0)\vec{a}_1, \\ \vec{T}_i = t_i\vec{a}_i + (1-t_i)\vec{a}_{i+1}, (i=1, 2, \dots, n-1), \\ \vec{T}_n = t_n\vec{a}_n + (1-t_n)(-\vec{a}_{n-1}) \end{cases} \quad (3)$$

式中 $0 < t_i < 1$ 是切矢调节参数。

设过型值点 \vec{V}_i 且平行于切矢 \vec{T}_i 的切线为 l_i ，则 l_i 可表示为

$$l_i: \vec{r} = \vec{V}_i + u_i \vec{T}_i, \quad -\infty < u_i < +\infty \quad (4)$$

由于型值多边形 $\langle \vec{V}_0 \vec{V}_1 \dots \vec{V}_n \rangle$ 是凸的，所以相邻两条切线必相交于一点，例如，切线 l_i 和 l_{i+1} 相交于 \vec{V}_i^* （如图 1），且有

$$\vec{V}_i^* = \vec{V}_i + \frac{\vec{a}_{i+1} \times \vec{T}_{i+1}}{|\vec{T}_i \times \vec{T}_{i+1}|} \vec{T}_i, \quad i = 0, 1, \dots, n-1 \quad (5)$$

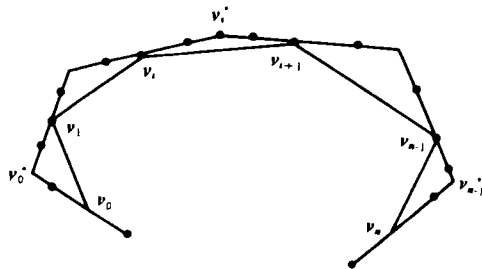


图 1

接下来我们按下面的原则确定控制顶点 \vec{r}_i , 首先取

$$\begin{cases} l_0 = |\vec{V}_1 - \vec{V}_0|, \\ l_i = \min\{|\vec{V}_i - \vec{V}_i|, |\vec{V}_i - \vec{V}_{i-1}|\}, \\ (i = 1, 2, \dots, n-1), \\ l_n = |\vec{V}_n - \vec{V}_{n-1}| \end{cases} \quad (6)$$

再令

$$\vec{r}_{3i} = \vec{V}_i, \quad (i = 0, 1, \dots, n), \quad (7)$$

$$\vec{r}_{3i+1} = \vec{V}_i + \lambda_i \frac{\vec{T}_i}{|\vec{T}_i|}, \quad 0 < \lambda_i < l_i, \\ i = 0, 1, \dots, n, \quad (8)$$

$$\vec{r}_{3i+2} = \vec{V}_{i+1} - \lambda_{i+1} \frac{\vec{T}_{i+1}}{|\vec{T}_{i+1}|}, \quad 0 < \lambda_{i+1} < l_{i+1}, \quad i = -1, 0, \dots, n-1, \quad (9)$$

由(6)~(9)式我们有

$$\vec{r}_{3i} = \frac{1}{2}(\vec{r}_{3i-1} + \vec{r}_{3i+1}), \quad i = 0, 1, \dots, n \quad (10)$$

且由 \vec{r}_i 构成的控制多边形是凸的。

于是 \vec{r}_j ($j = -1, 0, \dots, 3n+1$) 就是我们要找的均匀三次 B 样条曲线的控制点, 由这些控制点定义的均匀三次 B 样条曲线为

$$\vec{B}_i(u) = \sum_{l=0}^3 E_{i,3}(u) \vec{r}_{i+l-1} \quad (l = 0, 1, \dots, 3n-1), \quad (11)$$

根据第一节的结论和(6)~(10)式, 我们立即有

$$\vec{B}_{3j-1}(1) = \vec{V}_j, \quad j = 1, 2, \dots, n, \\ \vec{B}_{3j}(0) = \vec{V}_j, \quad j = 0, 1, \dots, n-1$$

于是我们知道均匀三次 B 样条插值曲线 (11) 式插值所有的型值点 \vec{V}_j ($j = 1, 2, \dots, n$)。又由 B 样条曲线的保凸性可知, 这条插值均匀三次 B 样条曲线是凸的, 即插值曲线具有保形性。如果要构造闭曲线, 只需令

$$\vec{V}_{n+1} = \vec{V}_0, \quad \vec{V}_{n+2} = \vec{V}_1$$

则可以完全类似的构造保凸的均匀三次 B 样条插值闭曲线。

综合上面的方法, 我们认为有下面的几条优点:

(1) B 样条插值曲线的控制顶点可以直接通过型值点的简单计算得到, 它不需用叠代法近似计算或是求解大型的矢量方程组反算控制顶点;

(2) 曲线是 C^2 连续的和保凸的;

(3) 曲线的形状可以通过调节切矢参数和控制点参数 l_i 来修改, 并且修改是十分方便有效的;

(4) 曲线的修改是局部的, 例如调节参数 l_i , 只影响与 l_i 有关的四段 B 样条曲线, 而与其它的曲线段形状无关。

3 数值例子

例 1 给定一凸数组

x	0	80	120	140
y	10	55	35	0

用本文的方法画出的保凸插值曲线如图 2.

例 2 给定另一组凸数组

x	0	37	57	90	90	50	10
y	18	0	0	20	50	60	50

画出的一条插值闭曲线如图 3. 从图可知, 用本文方法作出的图形是光滑的, 且视觉上是飘亮的.

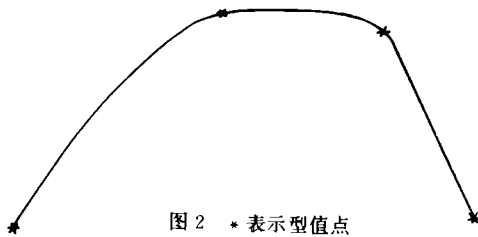


图 2 * 表示型值点

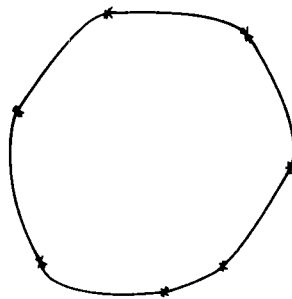


图 3 * 表示型值点

参 考 文 献

- 1 Yamaguchi F. A new curve fitting method using a CRT computer display. *Graphics and Image Processing*, 1978, 7: 425~437
- 2 Barsky B A. Determining a set of B-spline control Vertices to generate an interpolating surface. *Comput. Graphics and Image processing*, 1980, 9: 203~226
- 3 谭建荣. 均匀三次 B-spline 反求的一种新方法, *工程图学学报*, 1988, (2): 59~67
- 4 程正兴. 保凸插值参数三次曲线, *数学研究与评论*, 1983, 6 (2): 51~56
- 5 方遼. 闭 G^2 连续的保凸插值样条曲线. *工程数学学报*, 1992, 9 (4): 92~97
- 6 谭建荣. *工程曲线曲面的计算机辅助设计*. 河海大学出版社, 1993

Convex Preserving Uniform Cubic B-spline Interpolation Curve

Wang Wenzhong Fang Kui

(Department of System Engineering and Mathematics, NUDT, Changsha, 410073)

Abstract

Although uniform B-spline has convexity-preserving property, the curve doesn't interpolate any control points. If we insert two new control points between two consecutive control points, the new points will merge the old points to form a new set of control points. We show that cubic B-spline curve generated by the new set of points interpolates all old control points, and the interpolating curve is convexity preserving. The local modifications of the curve are possible. Finally, two numerical examples are given.

Key words computational geometry; B-spline curve, convexity preserving interpolation