学习报告

作者:周铁军

2019年7月19日

1 均匀 B-Spline 曲线

1.1 问题重现

控制顶点：$d_0, d_1, d_2, d_3, ..., d_m$，这里为了闭合，我是使 $d_m = d_0$

诚然，对于控制顶点而言，确实实现了闭合。

我使用的算法一是：

$$\boldsymbol{p}(u) = \sum_{j=0}^{n} \boldsymbol{d}_j \mathbf{N}_{j,k}(u) = \sum_{j=i-k}^{i} \boldsymbol{d}_j \mathbf{N}_{j,k}(u) \tag{1}$$

其中 d_i 为控制顶点，$N_{ij}(u)$ 为基函数。

基函数计算采用 De Boor-Cox 递推公式。

$$\begin{cases} \mathrm{N}_{i,0}(u) = \begin{cases} 1, & \text{若 } u_i \leqslant u < u_{i+1} \\ 0, & \text{其他} \end{cases} \\ \mathrm{N}_{i,k}(u) = \dfrac{u - u_i}{u_{i+k} - u_i} \mathrm{N}_{i,k-1}(u) + \dfrac{u_{i+k+1} - u}{u_{i+k+1} - u_{i+1}} \mathrm{N}_{i+1,k-1}(u) \\ \text{规定 } \dfrac{0}{0} = 0 \end{cases}$$

图 1: De Boor - Cox 递推公式

得到的结果如下：

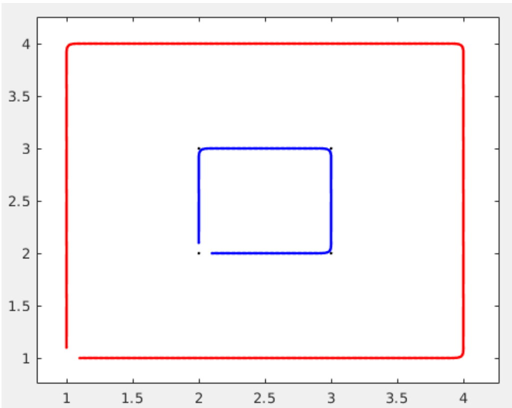

图 2: 算法一

为了验证算法结果，我采用了另一种算法 (只是为了检验算法一，故只验证了大矩形)，算法如下：

先对参数变换：

$$u = u(t) = (1-t)u_i + tu_{i+1}, \quad t \in [0,1]; i = k, k+1, \cdots, n \tag{2}$$

则 B 样条曲线方程改写为：

$$s_i(t) = p(u(t)) = \sum_{j=i=k}^{i} d_j \mathbf{N}_{j,k}(u(t)), \quad t \in [0,1]; i = k, k+1, \cdots, n \tag{3}$$

改成矩阵形式：

$$s_i(t) = \begin{bmatrix} 1 & t & t^2 & \cdots & t^k \end{bmatrix} \boldsymbol{M}_k \begin{bmatrix} \boldsymbol{d}_{i-k} \\ \boldsymbol{d}_{i-k+1} \\ \vdots \\ \boldsymbol{d}_i \end{bmatrix}, \quad t \in [0,1]; i = k, k+1, \cdots, n \tag{4}$$

这里矩阵 M_k 用的是书上给出的, 验证了计算了部分值，没有出错，网上查找资料时，亦为该矩阵。

$$M_3 = \frac{1}{6} \begin{bmatrix} 1 & 4 & 1 & 0 \\ -3 & 0 & 3 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix} \tag{5}$$

得到结果如下：

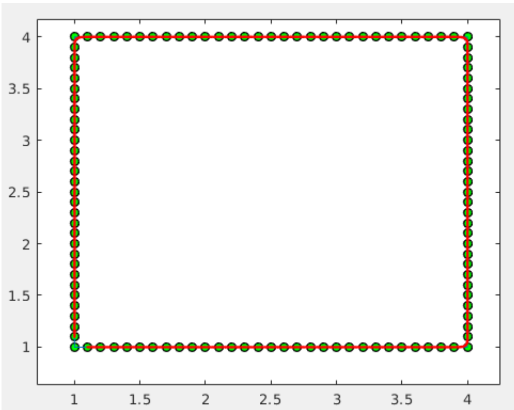

图 3: 算法二

算法二的结果与算法一如出一折，在左下有个缺口，**不过却也验证算法一的过程应该是没有出错。**

最后，在书上注意到如下性质：

1.2 k 次 B 样条闭曲线与开曲线的统一表示问题

给定控制顶点 $d_j(j = 0, 1, ..., m)$ 其中 $d_0 = d_m$

关于统一表示，可按如下步骤确定：

(1) 确定控制顶点下标的上界值 $n - m + k - r$, 其中 r 为重复度

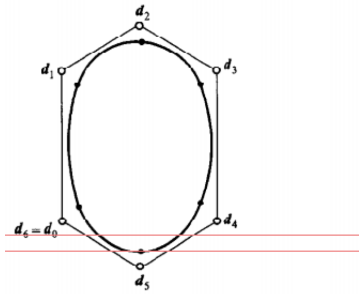

图 4: 与开曲线表示不统一的 B 样条曲线 (m=6,k=3)

(2) 按 $d_{m+j}-d_j(j=0,1,...,k-r)$ 决定 $k-r+1$ 个重顶点：$d_m=d_0,d_{m+1}=d_1,...,d_{m+k-r}=d_{k-r}$.

(3) 定义该闭曲线的控制顶点 $d_j(j=0,1,..,m,m+1,...,n)$.

即顶点下标的上界从给定的 m 增大到 n, 增加了 $k-r$ 个重顶点，加上原来已有的一个重顶点，共有 $k-r+1$ 个重顶点。按如上确定后，节点矢量与曲线定义域就与开曲线完全相同。

1.3 问题解决

从书中得知闭合的均匀 B 样条曲线的重复度 $r=1$, 则重顶点数应该为 $k-r+1=3$ 个。

即对于给定控制顶点 $d_j(j=0,1,...,m)$ 其中已经有 $d_m=d_0$，我们计算时需要增加 $d_{m+1}=d_1,d_{m+2}=d_2$ 来计算。

增加重顶点后，得到算法一图像：

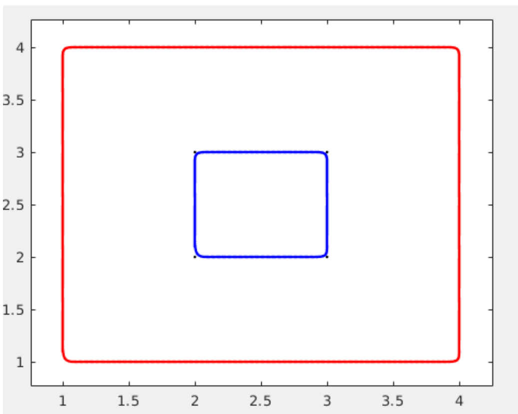

图 5: 算法一图像

得到算法二图像：

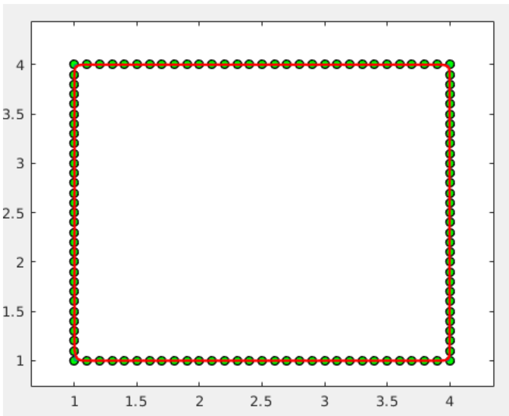

图 6: 算法二图像