Bézier Curve

项楚成 PB20010392

1 实验概述

本次实验中, 我们探究了 Bézier 曲线和 Bézier 样条的构造, 用交互的方式指定控制点, 然后绘制对应的 Bézier 曲线和 Bézier 样条, 并且实现了实时更改控制点的功能.

2 算法描述

2.1 Bézier Curve

设 $b_0, b_1, \cdots, b_n$ 为给定的控制点, $\vec{l}(t) = (X(t), Y(t)), t \in [0,1]$ 为相应的 Bézier 曲线, 我们可以通过迭代算法求解 $\forall t_0 \in [0,1]$, $\vec{l}(t_0)$ 的值.

Algorithm 1 De Casteljau algorithm

Input: $\boldsymbol{b} = (b_0, b_1, \cdots, b_n)$, t_0
Output: $\vec{l}(t_0)$

```
procedure BEZIER(b, t_0)
    for r=1:n do
        for i=0:n-r do
            b_i^(r) ← (1 − t) b_i^(r−1) + t b_{i+1}^(r−1)
        end for
    end for
    return b_0^(n)
end procedure
```

算法的时间复杂度为 $\mathcal{O}(n^2)$, 空间复杂度为 $\mathcal{O}(n)$, 其中 n 为控制点的数目.

2.2 Bézier Spline

Bézier 曲线有一些缺点, 如: 只有当多项式的次数较高时, 曲线才对控制点有较好的逼近效果; 改变一个控制点的位置会影响到整条曲线; 对控制点不能做到插值效果. 于是引入了 Bézier 样条.

给定插值点 $k_1, k_2, \cdots, k_n$, 和节点序列 $t_1, t_2, \cdots, t_n$, $t_i < t_{i+1}$. 我们希望得到分片的三阶 Bézier 样条曲线 $\vec{l}(t)$, 满足 $\vec{l}(t_i) = k_i$. 每四个 Bézier 控制点可以得到一条三阶 Bézier 曲线 $\vec{l}_i(t) = (1-t)^3 k_i + 3(1-t)^2 t x_i + 3(1-t) t^2 y_i + t^3 k_{i+1}$, 其中 x_i, y_i 是待定的控制点, 在连接处 k_j, 有如下的连续性要求:

- C^1 连续性要求: $\dfrac{k_j - y_{j-1}}{t_j - t_{j-1}} = \dfrac{x_j - k_j}{t_{j+1} - t_j},\ j = 2, \cdots, n-1.$
- C^2 连续性要求: $\dfrac{k_j - 2y_{j-1} + x_{j-1}}{(t_j - t_{j-1})^2} = \dfrac{y_j - 2x_j + k_j}{(t_{j+1} - t_j)^2},\ j = 2, \cdots, n-1.$

对节点序列 t_j 的选取也有多种方式, 例如均匀参数化, 弧长参数化, 向心参数化 (Centripetal), Foley 参数化, 仿射不变参数化等.

均匀参数化

令 $t_j = j$, 代入连续性要求, 得到方程组:

$$\begin{cases} x_j + y_{j-1} = 2k_j,\ j = 2, \cdots, n-1 \\ 2x_j + x_{j-1} = y_j + 2y_{j-1},\ j = 2, \cdots, n-1 \end{cases}$$

消去 y_j, 得到

$$x_{j-1} + 4x_j + x_{j+1} = 4k_j + 2k_{j+1},\ j = 2, \cdots, n-2$$

注意到一共有 $n-1$ 个变量 x_j, 但只有 $n-3$ 个约束条件, 所以自由度为 2, 我们加入自然边界条件, 即 $\vec{l}_1''(0) = 0, \vec{l}_{n-1}''(1) = 0$. 联立方程:

$$\begin{cases} x_{j-1} + 4x_j + x_{j+1} = 4k_j + 2k_{j+1},\ j = 2, \cdots, n-2 \\ 2x_1 + x_2 = k_1 + 2k_2 \\ 7x_{n-1} + 2x_{n-2} = k_n + 8k_{n-1} \end{cases}$$

写为矩阵的形式

$$\begin{bmatrix} 2 & 1 & & & & \\ 1 & 4 & 1 & & & \\ & 1 & 4 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & 4 & 1 \\ & & & & 2 & 7 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{n-2} \\ x_{n-1} \end{bmatrix} = \begin{bmatrix} k_1 + 2k_2 \\ 4k_2 + 2k_3 \\ 4k_3 + 2k_4 \\ \vdots \\ 4k_{n-2} + 2k_{n-1} \\ k_n + 8k_{n-1} \end{bmatrix}$$

解出 x_j 后, 再由 $y_j = 2k_{j+1} - x_{j+1},\ j = 1, 2, \cdots, n-2,\ y_{n-1} = 4x_{n-1} + x_{n-2} - 4k_{n-1}$ 解出 y_j. 最后用 $(k_j,\ x_j,\ y_j,\ k_{j+1})$, $j = 1, 2, \cdots, n-1$ 作为控制点, 作为绘制 Bézier 曲线的参数, 得到 $n-1$ 条 Bézier 曲线, 即所求的 C^2 连续的 Bézier 样条.

3 实验结果

以下图片从左到右分别展示了控制点的数目由少到多时, Bézier 曲线和 Bézier 样条对折线的逼近效果.

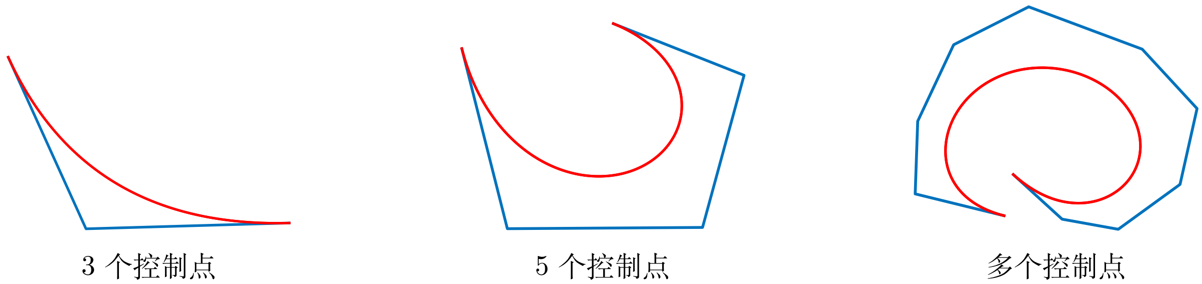

图 1: Bézier 曲线绘制

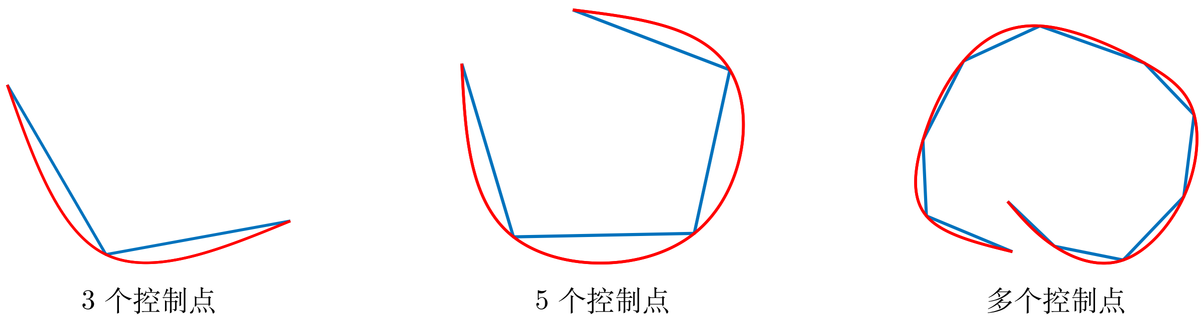

图 2: Bézier 样条绘制

结果分析

Bézier 曲线和 Bézier 样条都能对折线有较好的逼近效果, 且一般来说, 控制点数目越多, 逼近效果越好. 相比之下, Bézier 曲线只能在端点处插值折线, 而 Bézier 样条可以插值所有的控制点, 且对折线的拟合效果也更好.

4 实验总结

通过本次实验, 我们实现了 Bézier 曲线和 Bézier 样条的绘制, 并且可以交互式地选取控制点与插值点. 算法有较高的精确度和稳定性, 可以对给定的函数有很好的逼近效果.