

Euler Bèzier 曲线与 Euler B-Spline 曲线

王庶霖

目录

摘要	2
1 平面 Euler Bézier 曲线与 Euler B-Spline 曲线	3
2 利用 Euler Bézier 曲线和 Euler B-Spline 曲线构造 G2 连续圆角曲线	3
2.1 平面 Euler Bézier 曲线	3
2.2 平面 Euler B-Spline 曲线	6
3 空间 Euler Bézier 曲线与 Euler B-Spline 曲线	7
3.1 空间 Euler 螺线	7
3.2 空间 Euler 多边形	9
3.3 空间 Euler Bézier 曲线	10
3.4 空间 Euler B-Spline 曲线	13
4 实验结果	15
参考文献	15

摘要

这里是摘要内容。这里是摘要内容。这里是摘要内容。这里是摘要内容。这里是摘要内容。这里是摘要内容。这里是摘要内容。这里是摘要内容。这里是摘要内容。

关键词: 关键词 1 关键词 2 关键词 3 ...

1 平面 Euler Bézier 曲线与 Euler B-Spline 曲线

定义 1.1 (Euler 多边形) 如果平面多边形 $P_0 \dots P_n$ 满足

- (1). $\|P_i P_{i+1}\| (i = 0, 1, \dots, n-1)$ 是定值;
 - (2). $\theta_i (i = 1, 2, \dots, n-1)$ 是等差数列,
- (1)

则称该多边形为 *Euler 多边形*.

定义 1.2 (Euler Bézier 曲线) 如果平面 Bézier 曲线 $P(t) = \sum_{i=0}^n P_i B_{i,n}(t), t \in [0, 1]$ 的控制多边形是 *Euler 多边形*, 则称曲线 $P(t)$ 是 *Euler Bézier 曲线*.

定义 1.3 (Euler B-Spline 曲线) 记平面上的 k 阶均匀节点 *B-Spline* 曲线 $P(t) = \sum_{i=0}^n P_i N_{i,k}(t), t \in [t_{k-1}, t_{n+1}]$, 节点向量 $t_i = i \ (i = 0, 1, \dots, n+k)$. 如果其控制多边形 P_i 是 *Euler 多边形*, 则称曲线 $P(t)$ 是 *Euler B-Spline 曲线*.

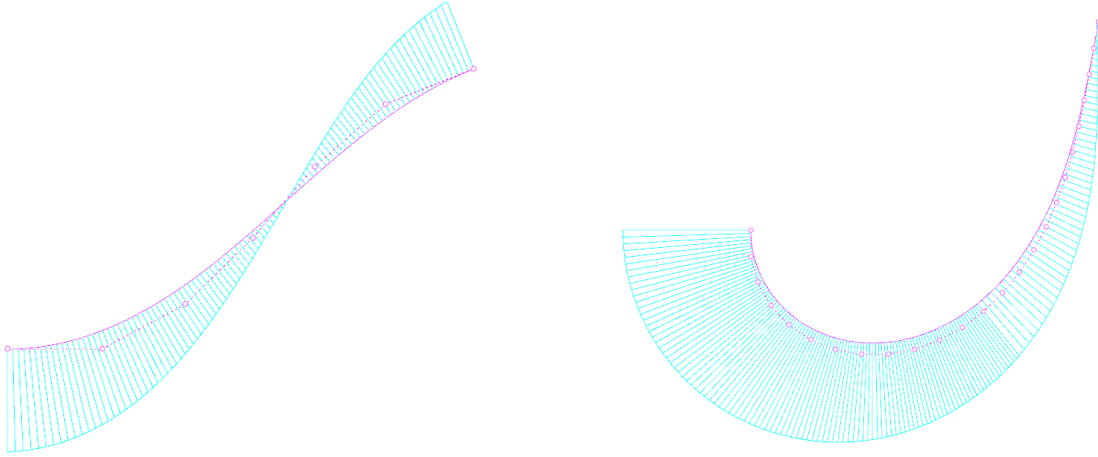


Figure 1.1: Euler Bézier 曲线

2 利用 Euler Bézier 曲线和 Euler B-Spline 曲线构造 G^2 连续圆角曲线

2.1 平面 Euler Bézier 曲线

对于一个给定的多边形, 对转角做圆角处理是非常常见的需求, 传统的方法是使用圆弧, 通过确定圆弧的半径来控制圆角的程度. 由于圆弧是固定曲率的, 这种传统的方法只能构造 G^1 连续的圆角, 使用 Euler Bézier 曲线则可以构造出 G^2 连续的圆角曲线, 并且可以保证有且仅有一个曲率极值点.

考虑多边形的一个转角 ABC , 向量 AB 到向量 BC 的旋转角为 $\alpha, |\alpha| \in (0, \pi)$, 我们对转角 B 构造圆角曲线. 由于 Euler Bézier 螺线是曲率单调的, 而多边形的圆角曲线要达到 G^2 连续需要保证两端的曲率为 0, 所以我们使用对称的两条 Euler Bézier 螺线拼接得到圆角曲线. 设圆角曲线的起点 P_S 在边 AB 上, 由于要构造对称的两条曲线, 我们把一边曲线的终点 P_E 取在多边形的内角平分线上. 为保

证 G^1 连续, 起点 P_S 处的切方向为向量 \mathbf{AB} 的方向, 记为 $\mathbf{T}_S = \mathbf{AB}/\|\mathbf{AB}\|$. 终点 P_E 处的切向 \mathbf{T}_E 应当与多边形的内角平分线垂直. 假设 Bézier 曲线 $\mathbf{P}(t) = \sum_{i=0}^n \mathbf{P}_i B_{i,n}(t)$, $t \in [0, 1]$ 插值了这两个边界条件, $P_i P_{i+1}$ 到 $P_{i+1} P_{i+2}$ 的角为 θ_{i+1} ($i = 0, 1, \dots, n-2$). 那么可以得到

$$\prod_{i=1}^{n-1} R(\theta_i) \mathbf{T}_S = \mathbf{T}_E \quad (2)$$

其中 $R(\theta)$ 表示旋转角为 θ 的旋转矩阵. 根据定义 (1.2), θ_i 应当为等差数列. 考虑 Bézier 曲线的起点曲率

$$\kappa(0) = \frac{n-1}{n} \frac{\sin(\theta_1)}{l}$$

保证 G^2 连续要令 $\kappa(0) = 0$, 则有 $\theta_1 = 0$, 故设 $\theta_i = (i-1)\Delta\theta$ ($i = 1, \dots, n-1$), 为了保证对称性, \mathbf{T}_E 应当与多边形内角平分线垂直, 则 \mathbf{T}_S 到 \mathbf{T}_E 的夹角为 $\alpha/2$, 所以有

$$\sum_{i=1}^{n-1} \theta_i = \frac{(n-2)(n-1)}{2} \Delta\theta = \frac{\alpha}{2} \quad (3)$$

计算得 $\Delta\theta = \frac{\alpha}{(n-2)(n-1)}$, 从而计算每个 θ_i . Euler 多边形顶点的序列可以递推给出:

$$\begin{aligned} \mathbf{P}_0 \mathbf{P}_1 &= l \mathbf{T}_S \\ \mathbf{P}_1 \mathbf{P}_2 &= R(\theta_1) \mathbf{P}_0 \mathbf{P}_1 \\ &\dots \\ \mathbf{P}_i \mathbf{P}_{i+1} &= R(\theta_i) \mathbf{P}_{i-1} \mathbf{P}_i \end{aligned} \quad (4)$$

根据起点终点条件有 $\mathbf{P}_0 = \mathbf{P}_S$, $\mathbf{P}_n = \mathbf{P}_E$, 结合递推式 (4) 得到:

$$\mathbf{P}_S \mathbf{P}_E = \sum_{i=0}^{n-1} \mathbf{P}_i \mathbf{P}_{i+1} = l \left[\sum_{i=0}^{n-1} R(\phi_i) \right] \mathbf{T}_S \quad (5)$$

其中 $\phi_i = \sum_{j=1}^i \theta_j$ ($i = 1, \dots, n-1$), $\phi_0 = 0$. 对于一个给定正整数 n , 我们可以计算出向量 $\mathbf{D} = \sum_{i=0}^{n-1} R(\phi_i) \mathbf{T}_S$, 这是一个常向量, 从起点到终点的连线必须与它平行. 将向量 \mathbf{D} 与向量 \mathbf{T}_S 的夹角记为 β (注意 β 也是有符号角, 应当与 α 符号相同), 则有

$$\frac{\|\mathbf{OP}_S\|}{\cos(\frac{\alpha}{2} - \beta)} = \frac{\|\mathbf{OP}_E\|}{\sin(\beta)} = \frac{\|\mathbf{P}_S \mathbf{P}_E\|}{\cos(\frac{\alpha}{2})} \quad (6)$$

根据方程 (6), 我们可以在给定圆角起点 \mathbf{P}_S 或者给定角平分线上的经过点 \mathbf{P}_E 时求得唯一的 Euler 多边形. 对于 $n+1$ 个顶点的 Euler 多边形, 判断以它们为控制顶点的 Bézier 曲线是否为 Euler Bézier 曲线, 如果没有满足 Euler Bézier 曲线的条件, 则使 n 自加一再次重复上述过程, 直到满足 Euler Bézier 曲线的条件或者控制顶点数达到了预先设定的最大值停止迭代. 如果迭代结束曲线满足了螺线条件, 就得到了一半的圆角曲线. 再将该曲线沿着转角的平分线作对称变换即可得到另一半圆角曲线, 该算法的伪代码实现见 Algorithm 1.

图 (2.1) 是对正六边形 (黑色直线段) 应用 Algorithm 1 的结果, 六个圆角曲线 (品红色) 均为对称的两条四次 Bézier 曲线拼接而成, 其圆角起点与终点都是六边形边的三等分点. 图中的浅蓝色曲率梳表明了该圆角的 G^2 连续性, 并且每一个圆角曲线自身对称, 有且仅有一个曲率极值点, 位于对称轴上.

Data: 构成 Corner 的三个顶点 A, B, C , BP_S 的长度 L_S 或 BP_E 的长度 L_E

Result: Bézier 曲线 $P(t)$ 的控制顶点序列 P_0, P_1, \dots, P_n 和对称的顶点序列

```

1  $n = 4$ ;
2 初始化 Bézier 曲线  $P(t)$ ;
3  $T_S = \text{Normalize}(B - P_S)$ ;
4  $D = T_S$ ;
5  $temp = T_S$ ;
6 while EulerBézierSpiralCheck( $P(t)$ ) == FALSE and  $n < 20$  do
7   计算  $AB$  与  $BC$  的夹角  $\alpha$ ;
8    $\Delta\theta = \frac{\alpha}{(n-2)(n-1)}$ ;
9   for  $i = 0, 1, \dots, n-2$  do
10     $temp.\text{Rotate}(i * \Delta\theta)$ ;
11     $D+ = temp$ ;
12  end
13  计算  $AB$  与  $D$  的夹角  $\beta$ ;
14   $length = \cos(\alpha/2) / \cos(\alpha/2 - \beta) * \|B - P_S\| / \|D\|$ ;
15   $temp = T_S * length$ ;
16   $P_0 = P_S$ ;
17  for  $i = 1, \dots, n-1$  do
18     $P_i = P_{i-1} + temp$ ;
19     $temp.\text{Rotate}((i-1) * \Delta\theta)$ ;
20  end
21   $n++$ ;
22 end

```

Algorithm 1: Euler Bézier 圆角

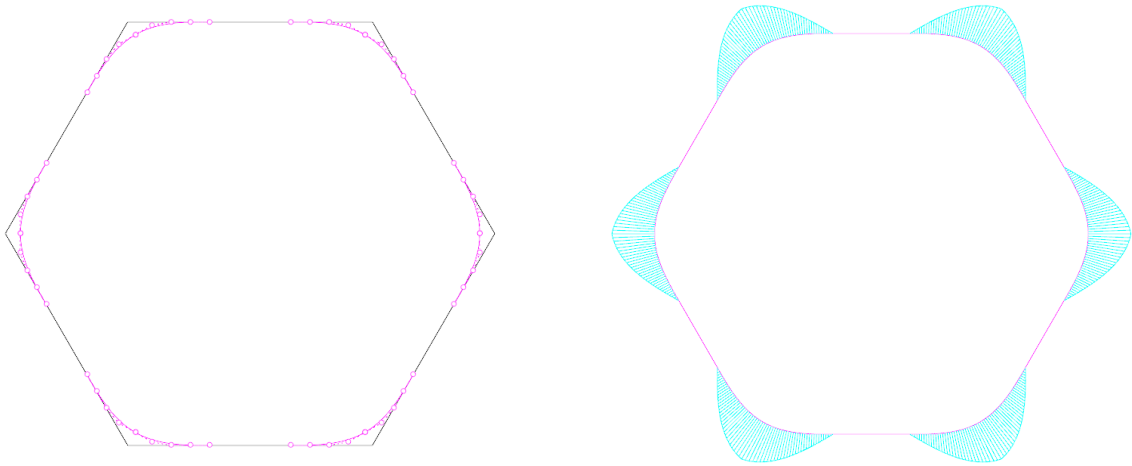


Figure 2.1: 算法 (1) 得到的 G^2 连续 Euler Bèzier 螺线圆角

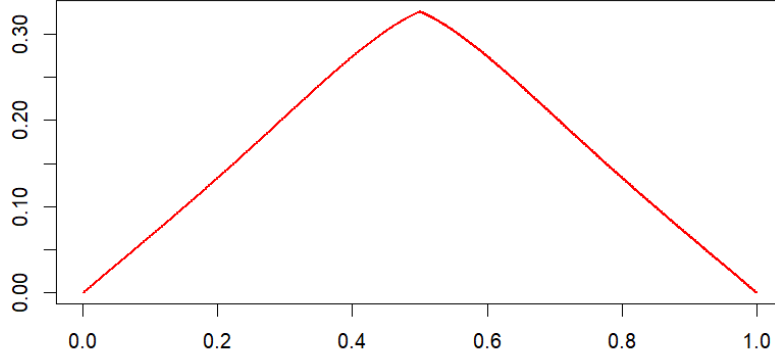


Figure 2.2: 图 (2.1) 中一个圆角曲线的曲率.

2.2 平面 Euler B-Spline 曲线

虽然 Bézier 曲线已经能得到满足条件的圆角曲线, 但是通常控制顶点较多导致次数较高, 而使用 B-Spline 曲线可以保持次数为三次, 方便进行曲线的拼接处理. 利用二维 Euler B-Spline 曲线构造圆角大体与 Bézier 曲线的构造一致, 只是生成 Euler 多边形时考虑不同的边界条件. 设四阶 (三次) 均匀 B-Spline 曲线为 $\mathbf{P}(t) = \sum_{i=0}^n \mathbf{P}_i N_{i,4}(t)$, 节点向量为 $t_j = j, j = 0, 1, \dots, n+4$

在 Euler Bézier 圆角的构造中, 我们将起点 \mathbf{P}_0 设置在圆角起点 \mathbf{P}_S 上, 我们令向量 $\mathbf{P}_0\mathbf{P}_1$ 与 \mathbf{T}_S 同向, 令 $\mathbf{P}_0\mathbf{P}_1\mathbf{P}_2$ 三点共线, 这些都是基于 Bézier 曲线的端点各阶导数与控制顶点的关系得到的. 而对于所求的均匀节点 B-Spline 曲线 $\mathbf{P}(t), t \in [3, n+1]$, 我们有

$$\begin{aligned} \mathbf{P}(3) &= \frac{\mathbf{P}_0 + 4\mathbf{P}_1 + \mathbf{P}_2}{6} \\ \mathbf{P}'(3) &= \frac{\mathbf{P}_2 - \mathbf{P}_0}{2} \\ \mathbf{P}''(3) &= \mathbf{P}_0 - 2\mathbf{P}_1 + \mathbf{P}_2 \end{aligned} \tag{7}$$

为满足 G^2 连续条件, 必须有

$$\begin{aligned} \frac{\mathbf{P}_0 + 4\mathbf{P}_1 + \mathbf{P}_2}{6} &= \mathbf{P}_S \\ \mathbf{P}_2 - \mathbf{P}_0 &= \lambda \mathbf{T}_S, \text{ for some } \lambda > 0 \\ \mathbf{P}_0\mathbf{P}_2 \times (\mathbf{P}_1\mathbf{P}_2 - \mathbf{P}_0\mathbf{P}_1) &= \mathbf{0} \end{aligned} \tag{8}$$

第三个条件等价于 $\mathbf{P}_0\mathbf{P}_2 \times \mathbf{P}_0\mathbf{P}_1 = \mathbf{0}$, 所以等式成立当且仅当 $\mathbf{P}_0, \mathbf{P}_1$ 与 \mathbf{P}_2 三点共线. 注意到 Euler 多边形各边等长, 故方程 (8) 的解为

$$\begin{cases} \mathbf{P}_0 = \mathbf{P}_S - l\mathbf{T}_S \\ \mathbf{P}_1 = \mathbf{P}_S \\ \mathbf{P}_2 = \mathbf{P}_S + l\mathbf{T}_S \end{cases} \tag{9}$$

其中 $l > 0$ 为多边形的边长.

另外 $\Delta\theta$ 与常向量 \mathbf{D} 的定义也会发生变化, 由于终点的导数为

$$\mathbf{P}'(n+1) = \frac{\mathbf{P}_n - \mathbf{P}_{n-2}}{2} \quad (10)$$

所以 θ_i 与 α 的关系变为

$$\sum_{i=2}^{n-2} \theta_i + \frac{1}{2}\theta_{n-1} = \frac{\alpha}{2} \quad (11)$$

结合 $\theta_i = (i-1)\Delta\theta$, 得到

$$\Delta\theta = \frac{\alpha}{(n-2)^2} \quad (12)$$

记 $\phi_i = \sum_{j=1}^i \theta_j (i=1, \dots, n-1), \phi_0 = 0$. 可以表示控制多边形每一个顶点的坐标

$$\mathbf{P}_k = \mathbf{P}_0 + l \sum_{i=0}^{k-1} \mathbf{R}(\phi_i) \mathbf{T}_S, \quad k \geq 1 \quad (13)$$

根据终点处的边界条件

$$\mathbf{P}_E = \frac{1}{6}(\mathbf{P}_{n-2} + 4\mathbf{P}_{n-1} + \mathbf{P}_n) \quad (14)$$

结合式 (13), (14) 和 (9) 得到

$$\mathbf{P}_S \mathbf{P}_E = l \left[\frac{1}{6}(\mathbf{D}_{n-2} + 4\mathbf{D}_{n-1} + \mathbf{D}_n) - \mathbf{T}_S \right] \quad (15)$$

其中

$$\mathbf{D}_k = \sum_{i=0}^{k-1} \mathbf{R}(\phi_i) \mathbf{T}_S, \quad k = 1, \dots, n \quad (16)$$

对于给定的 n , 每个 \mathbf{D}_k 都是定值, 最后我们求解 l , 只需要和 Bézier 的方法类似, 确定了 \mathbf{P}_S 或 \mathbf{P}_E 的位置后, 通过求解三角形得到 $\mathbf{P}_S \mathbf{P}_E$ 的长即可. 之后再进行与算法 (1) 类似的迭代过程即得到一半的 Euler B-Spline 圆角曲线, 将其对称并拼接可以得到一整条 B-Spline 圆角曲线.

3 空间 Euler Bézier 曲线与 Euler B-Spline 曲线

3.1 空间 Euler 螺线

平面的各种螺线很早以前就有各种定义和性质的研究, 一般而言可以概括为曲率 $\kappa(s)^\alpha = as + b$, 即曲率的某个方幂与弧长是线性关系, $\alpha = 1$ 为 Euler 螺线, $\alpha = -1$ 是对数螺线, $\alpha = -2$ 为圆的渐开线, 这样的曲线其显著特征就是曲率的单调性. 在空间中对于螺线的定义应该同时包括曲率和挠率, 我们采用 Euler 螺线定义方法, 将曲率 κ , 挠率 τ 均与弧长呈线性关系,

定义 3.1 (空间 Euler 螺线) 设 $\mathbf{P}(t), t \in [t_0, t_1]$ 是空间中一段曲线, 设 s 为其弧长参数, 如果其曲率 $\kappa(s)$ 和挠率 $\tau(s)$ 满足 $\kappa(s) = as + b, \tau(s) = cs + d$, 其中 a, b, c, d 均是常数, 则称 $\mathbf{P}(t)$ 是空间 Euler 螺线.

事实上, 根据曲线论的第一基本定理可知, 给定了常数 a, b, c, d 后, 这样定义的空间 Euler 螺线一定存在, 并且在相差一个刚体运动的意义上是唯一的. 不过, 直接试图求解这样的曲线并不是好的选择, 一方面它需要求解 Frenet 微分方程组得到离散结果, 不能得到准确的参数化表示, 更不能表示成 NURBS 曲

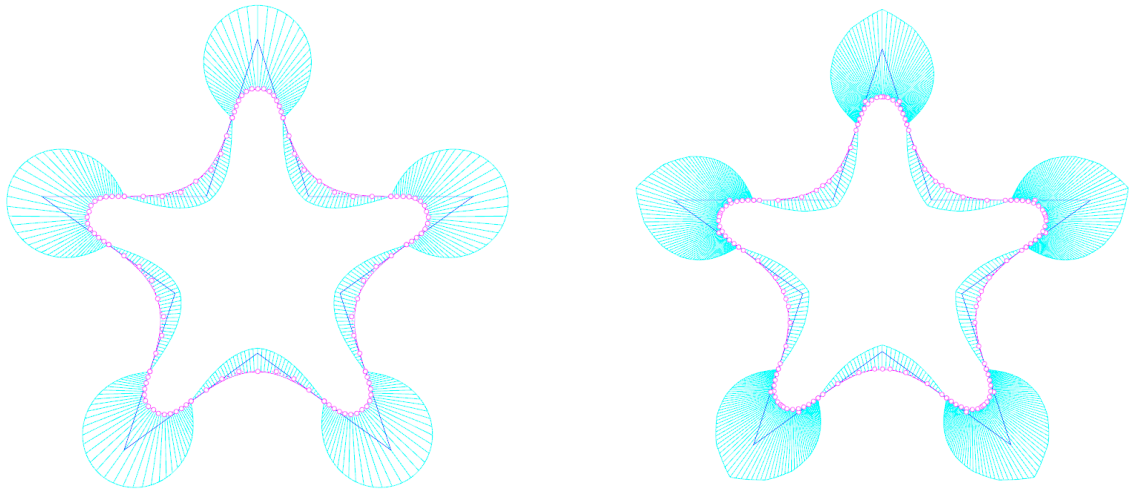


Figure 2.3: 对正五角星的每一个角做圆角, 圆角的起点和终点均为各边中点. 左图: 使用 Euler Bézier 曲线生成, 钝角的圆角曲线是两段 4 次曲线, 锐角的圆角为两段 7 次曲线. 右图: 使用 Euler B-Spline 曲线生成, 所有曲线均为三次. 两个例子都表现出了 G^2 连续性, Bézier 曲线次数更高, 每一段内的几何连续性也更高, B-Spline 曲线次数恒定, 但由于拼接, 几何连续性只能达到 G^2 .



Figure 2.4: 图 (2.3) 中两个 Bézier 圆角的曲率



Figure 2.5: 图 (2.3) 中两个 B-Spline 圆角的曲率

线；另一方面是实际应用中往往要构造满足 G^1 边值条件的曲线，在这种情况下这样严格定义的曲线未必是存在的。所以我们将空间 Euler 螺线的几何性质离散化，计算出一个可以逼近 Euler 螺线的离散多边形，再利用 B 样条曲线或 Bézier 曲线能逼近多边形控制顶点的性质，来构造能准确参数化的多项式曲线。因此我们和平面有类似的方法，引入空间 Euler 多边形。

3.2 空间 Euler 多边形

三维空间中插值 G^1 边界条件的曲率（以及挠率）单调曲线的存在性不像二维的情形有简单且完整的理论证明。现有的一些构造三维螺线的结果有对于离散多边形的非线性细分方法，有根据 Frenet 方程的离散形式进行计算的方法，这些方法得到的都是离散形式的曲线，需要进行多次迭代，或者计算大量的采样点，而我们的目的是使用显式的（分段）多项式曲线来尽可能的构造曲率单调曲线。将空间 Euler 螺线按弧长参数均匀采样点会得到一个多边形，我们可以去在多边形中寻找一些离散的量作为曲线曲率和挠率连续量的近似。曲率反映了切向的变化速度，因此我们可以用边之间的角度大小来衡量曲率的大小，但是不同于二维情况，三维的曲率是没有正负号，即曲率向量的方向可以是任意的，但是角度的大小不能体现这一点，所以我们将空间的 Euler 多边形看作平面的 Euler 多边形“扭曲”得到的。我们使用多边形相邻边的两种角度来反映曲率和挠率，一种是将两条边投影到 xOy 平面上，投影后两向量的符号角度，定义为“水平角度”；另一种是两条边的俯仰角的差值，定义为“竖直角”，参见图 (3.1)

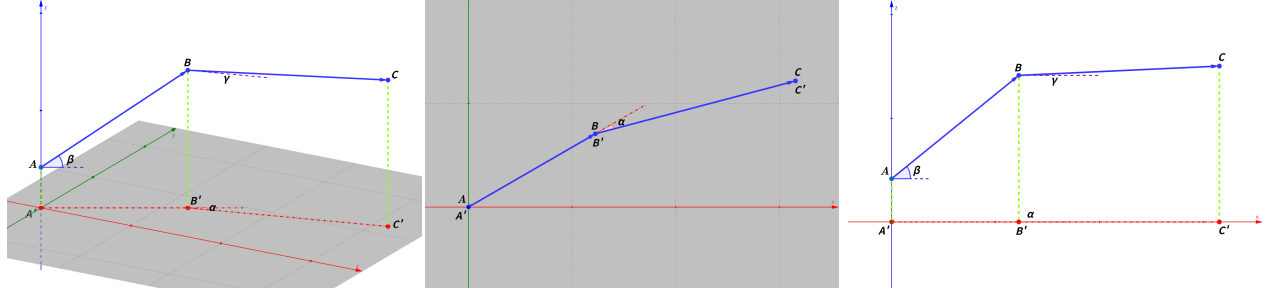


Figure 3.1: 左，中，右分别是一段空间多边形角点 ABC 的标准视图，顶视图和主视图， A', B', C' 分别为三点在 xOy 平面的投影。 $A'B'$ 与 $B'C'$ 的夹角 α 定义为多边形角点 B 的水平角 θ_B 。 β 和 γ 分别是向量 AB , BC 与 xOy 平面的夹角，我们定义点 B 处的竖直角 $\phi_B = \gamma - \beta$ 。

下面给出两种角的严格定义。假设有非零的三维向量 $\mathbf{v} = (v_x, v_y, v_z)$ ，将其化为极坐标得到 $\hat{\mathbf{v}} = (\rho_v, \theta_v, \phi_v)$ ，其中 $\rho_v > 0$ 是 \mathbf{v} 的模长， $\theta_v \in [0, 2\pi)$ 是 \mathbf{v} 在 xy 平面投影向量的极角， $\phi_v \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ 是 \mathbf{v} 的仰角，它们满足等式

$$\begin{aligned} \rho_v &= \sqrt{v_x^2 + v_y^2 + v_z^2} \\ \theta_v &= \begin{cases} \arccos\left(\frac{v_x}{\sqrt{v_x^2 + v_y^2}}\right), & v_x^2 + v_y^2 > 0, v_y \geq 0 \\ -\arccos\left(\frac{v_x}{\sqrt{v_x^2 + v_y^2}}\right) + 2\pi, & v_x^2 + v_y^2 > 0, v_y < 0 \\ 0, & v_x = v_y = 0 \end{cases} \\ \phi_v &= \begin{cases} \arctan\left(\frac{v_z}{\sqrt{v_x^2 + v_y^2}}\right), & v_x^2 + v_y^2 > 0 \\ \text{sign}(v_z) * \frac{\pi}{2}, & v_x = v_y = 0 \end{cases} \end{aligned} \quad (17)$$

对于一个空间多边形 P_0, \dots, P_n , 它的 n 条边 $P_i P_{i+1}$ 记为向量 $v_i (0 \leq i \leq n-1)$, 其极坐标表示的极角和俯角记为 θ_{v_i}, ϕ_{v_i} , 记相邻边 v_i, v_{i+1} 的水平夹角为 $\theta_i = \theta_{v_{i+1}} - \theta_{v_i}$, 相邻边 v_i, v_{i+1} 的竖直夹角为 $\phi_i = \phi_{v_{i+1}} - \phi_{v_i}$, $0 \leq i \leq n-2$ 我们使用这两种角定义三维的 Euler 多边形.

定义 3.2 (空间 Euler 多边形) 设三维空间中有多边形 P_0, \dots, P_n , 顶点互不相同, 如果它们的相邻边的水平夹角 θ_i , 竖直夹角 ϕ_i 满足:

$$\begin{aligned} (1). \quad & \theta_i - 2\theta_{i+1} + \theta_{i+2} = 0, \quad i = 0, 1, \dots, n-4 \\ (2). \quad & \phi_i - 2\phi_{i+1} + \phi_{i+2} = 0, \quad i = 0, 1, \dots, n-4 \\ (3). \quad & \text{所有边长} \|v_i\| \text{相等} \end{aligned} \tag{18}$$

则称 P_0, \dots, P_n 是三维 Euler 多边形.

由空间 Euler 多边形作为控制顶点生成的 Bézier 曲线或整数节点 B 样条曲线称为空间 Euler Bézier 曲线或空间 Euler B-Spline 曲线.

假定有三维 G^1 条件: 起点 P_S , 终点 P_E , 以及起点终点切向 T_S, T_E , 其中 T_S, T_E 是单位向量, 并且与起点指向终点的向量 $P_S P_E$, 三个向量异面. 对于这样的 G^1 条件, 直接构造严格满足定义 (3.2) 的曲线是行不通的, 事实上在二维情况下都是不可行的, 所以我们依然使用一种非线性细分的方法对初始多边形进行迭代, 使其逐渐逼近三维的 Euler 多边形, 当然没有理论保证对于任意的边值条件都存在满足定义 (3.2) 的多边形, 但是如果这样的多边形存在, 我们的方法可以很好的逼近出这样的多边形, 以它们作为控制顶点的 Bézier 和 B-Spline 曲线是曲率单调的.

3.3 空间 Euler Bézier 曲线

对于一般的三维 G^1 边值条件, 我们先作刚体变换使得起点 P_S 位于原点, 初始切向变换为 x 轴正向的单位向量 $(1, 0, 0)$ 并且终点的端切向与 xOy 平面平行, 这样的变换是非常容易的, 只需要一次平移和旋转即可.

变换后的边值条件, 起点和起点端切向即为 $P_S = (0, 0, 0), T_S = (1, 0, 0)$, 终点的端切向与 xOy 平面平行. 我们很容易构造一个初始多边形, 使得该多边形生成的 Bézier 曲线满足该 G^1 边值条件, 这样的多边形至少要四个顶点, 可以表示为

$$\begin{aligned} P_0 &= P_S \\ P_1 &= P_S + \lambda T_S \\ P_2 &= P_E - \mu T_E \\ P_3 &= P_E \end{aligned} \tag{19}$$

其中 $\lambda, \mu > 0$ 是可变的常数, 我们可以选择设置为 $\lambda = \mu = \|P_S P_E\|/3$.

接下来, 我们从初始多边形开始进行保边界条件的迭代. 假设当前的多边形顶点按顺序为 P_0, \dots, P_n , 那么 P_0 和 P_n 的位置不变, P_1 和 P_{n-1} 的位置只依赖于多边形的边长, 因为它们用于满足端切向条件, 所以从 P_2 开始计算. 假定现有连续的三点 P_{i-1}, P_i, P_{i+1} , 其中 $i = 2, \dots, n-2$, 我们计算新的 \hat{P}_i 更新该多边形. 我们把多边形相交于顶点 P_i 的边 v_{i-1} 和 v_i 的水平, 竖直夹角称为点 P_i 处的水平, 竖直转角, 记为 θ_i 和 ϕ_i (定义 (3.2)). 为了在局部满足 Euler 多边形的条件, 应当让转角满足 $2\theta_i = \theta_{i-1} + \theta_{i+1}$,

以及 $2\phi_i = \phi_{i-1} + \phi_{i+1}$, 因此, 我们更新 \mathbf{P}_i 处的转角为:

$$\begin{aligned}\hat{\theta}_i &= \frac{\theta_{i-1} + \theta_i + \theta_{i+1}}{3} \\ \hat{\phi}_i &= \frac{\phi_{i-1} + \phi_i + \phi_{i+1}}{3}\end{aligned}\quad (20)$$

接下来, 我们通过已经更新的 $\hat{\theta}_i$ 和 $\hat{\phi}_i$ 来确定 $\hat{\mathbf{P}}_i$ 的位置. 设 \mathbf{P}_{i-1} 和 \mathbf{P}_{i+1} 的坐标分别为 (x_1, y_1, z_1) 和 (x_2, y_2, z_2) , \mathbf{P}_{i-1} 和 \mathbf{P}_{i+1} 在 xOy 平面上的投影记为 $\bar{\mathbf{P}}_{i-1}$ 和 $\bar{\mathbf{P}}_{i+1}$, 所求的新 $\hat{\mathbf{P}}_i$ 在 xOy 面上的投影为 $\bar{\hat{\mathbf{P}}}_i$. 我们要使得 $\bar{\mathbf{P}}_{i-1}$, $\bar{\hat{\mathbf{P}}}_i$ 和 $\bar{\mathbf{P}}_{i+1}$ 三点形成的转角为 $\hat{\theta}_i$, 从而点 $\bar{\hat{\mathbf{P}}}_i$ 应当位于一段圆弧上, 该圆弧的半径为

$$R = \frac{\|\bar{\mathbf{P}}_{i-1}\bar{\mathbf{P}}_{i+1}\|}{2\sin(\hat{\theta}_i)} \quad (21)$$

圆心的位置也可以由下式给出

$$C_i = \frac{\bar{\mathbf{P}}_{i-1} + \bar{\mathbf{P}}_{i+1}}{2} + R(\text{sign}(\hat{\theta}_i) \cdot \frac{\pi}{2}) \cdot \frac{\bar{\mathbf{P}}_{i-1}\bar{\mathbf{P}}_{i+1}}{\|\bar{\mathbf{P}}_{i-1}\bar{\mathbf{P}}_{i+1}\|} \cdot \sqrt{R^2 - \frac{1}{4}\|\bar{\mathbf{P}}_{i-1}\bar{\mathbf{P}}_{i+1}\|^2} \quad (22)$$

其中 $R(\theta)$ 表示在 xOy 平面绕原点旋转 θ 角度, 注意到 $\hat{\theta}_i$ 是有符号的, 并且圆心与点 $\bar{\hat{\mathbf{P}}}_i$ 位于 $\bar{\mathbf{P}}_{i-1}\bar{\mathbf{P}}_{i+1}$ 的两侧. 将 C_i 的坐标记为 (a, b) , 从而我们可以设 $\bar{\hat{\mathbf{P}}}_i = (a + R\cos(x), b + R\sin(x))$, 设 $\hat{\mathbf{P}}_i = (a + R\cos(x), b + R\sin(x), z)$.

接下来考虑竖直角, 向量 $\mathbf{P}_{i-1}\hat{\mathbf{P}}_i$ 和 $\hat{\mathbf{P}}_i\mathbf{P}_{i+1}$ 的俯角应当满足

$$\begin{aligned}\tan(\phi_{\mathbf{P}_{i-1}\hat{\mathbf{P}}_i}) &= \frac{z - z_1}{L_1(x)} \\ \tan(\phi_{\hat{\mathbf{P}}_i\mathbf{P}_{i+1}}) &= \frac{z_2 - z}{L_2(x)}\end{aligned}\quad (23)$$

其中, $L_1(x)$ 和 $L_2(x)$ 分别表示点 $\bar{\hat{\mathbf{P}}}_i$ 到点 $\bar{\mathbf{P}}_{i-1}$ 和点 $\bar{\mathbf{P}}_{i+1}$ 的距离.

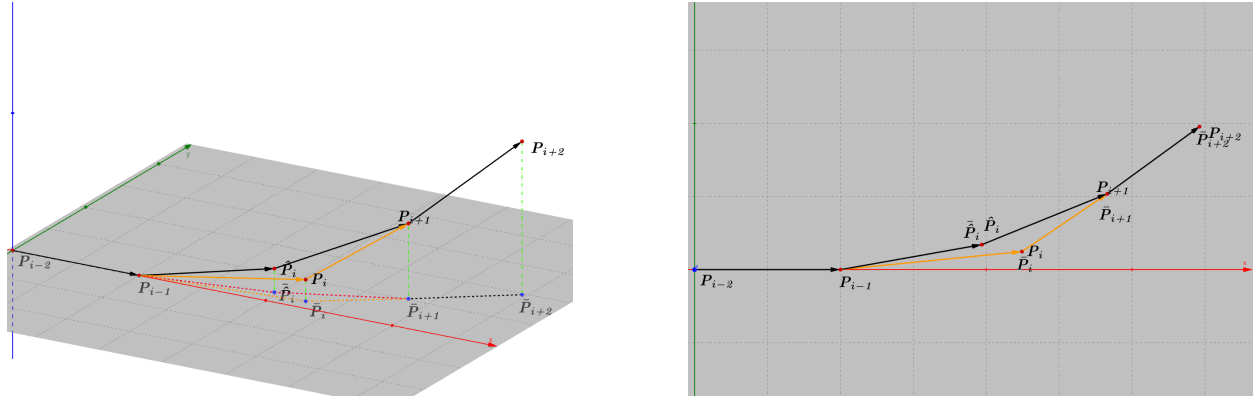


Figure 3.2: 算法图解.

我们要使 $\phi_{\hat{\mathbf{P}}_i\mathbf{P}_{i+1}} - \phi_{\mathbf{P}_{i-1}\hat{\mathbf{P}}_i} = \hat{\phi}_i$, 从而得到方程

$$\frac{z_2 - z}{L_2(x)} - \frac{z - z_1}{L_1(x)} = \tan(\hat{\phi}_i) \left(1 + \frac{(z_2 - z)(z - z_1)}{L_1(x)L_2(x)}\right) \quad (24)$$

最后再利用 Euler 多边形的各边相等, 要有 $\|\mathbf{P}_{i-1}\hat{\mathbf{P}}_i\| = \|\mathbf{P}_{i+1}\hat{\mathbf{P}}_i\|$, 则有方程

$$L_1(x)^2 + (z - z_1)^2 = L_2(x)^2 + (z_2 - z)^2 \quad (25)$$

注意到式 (25) 等价于

$$z = \frac{z_1 + z_2}{2} + \frac{L_1(x)^2 - L_2(x)^2}{2(z_1 - z_2)} \quad (26)$$

将其代入式 (24), 并记 $z_2 - z_1 = h$ 得到

$$\begin{aligned} & L_1(x)\left(\frac{h}{2} + \frac{L_1(x)^2 - L_2(x)^2}{2h}\right) - L_2(x)\left(\frac{h}{2} - \frac{L_1(x)^2 - L_2(x)^2}{2h}\right) \\ & - \tan(\hat{\phi}_i)(L_1(x)L_2(x) + \frac{h^2}{4} - \frac{(L_1(x)^2 - L_2(x)^2)^2}{4h^2}) = 0 \end{aligned} \quad (27)$$

将式 (27) 左侧的表达式记为 $\Phi(x)$, 我们即要求 $\Phi(x)$ 的零点.

关于函数 $\Phi(x)$ 的零点, 我们可以证明在一定条件下零点是存在的 (但是不一定唯一), 我们令 L 表示点 $\bar{\mathbf{P}}_{i-1}$ 到点 $\bar{\mathbf{P}}_{i+1}$ 的距离, 我们注意到当点 $\hat{\mathbf{P}}_i$ 与 $\bar{\mathbf{P}}_{i-1}$ 重合时, $L_1(x)$ 的值为 0, $L_2(x)$ 的值为 L , 将此时的 x 的值记为 x_1 ; 同理记另一端极限情况的 x 的值为 x_2 , 满足 $L_1(x_2) = L, L_2(x_2) = 0$, 不妨假设 $x_1 < x_2$. 我们可以得到一个使解存在的充分条件

定理 3.1 若 $|h| < L$, 则关于 x 的方程 (27) 在区间 $[x_1, x_2]$ 上一定有解.

证明

考虑函数 $\Phi(x)$ 在区间端点和中点处的取值, 我们有

$$\begin{aligned} \Phi(x_1) &= \frac{h^2 + L^2}{4} [\tan(\hat{\phi}_i)\left(\frac{L}{h}\right)^2 - 2\left(\frac{L}{h}\right) - \tan(\hat{\phi}_i)] \\ \Phi(x_2) &= \frac{h^2 + L^2}{4} [\tan(\hat{\phi}_i)\left(\frac{L}{h}\right)^2 + 2\left(\frac{L}{h}\right) - \tan(\hat{\phi}_i)] \\ \Phi(x_0) &= -\tan(\hat{\phi}_i)(L_0^2 + \frac{h^2}{4}) \end{aligned} \quad (28)$$

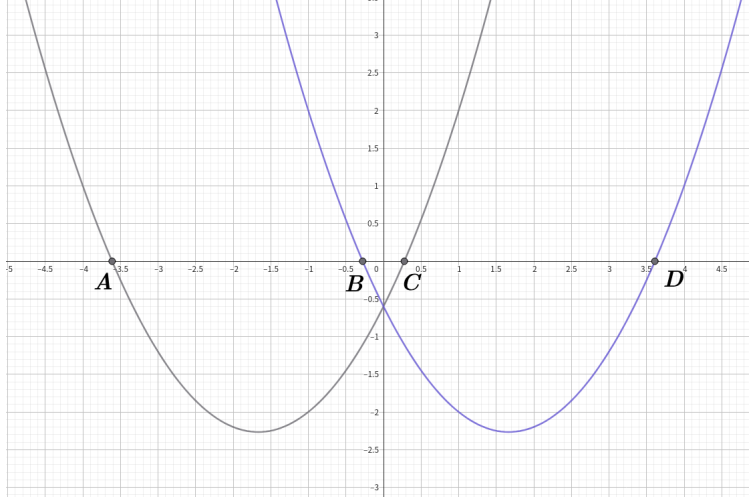
其中 $x_0 = \frac{x_1 + x_2}{2}$, 根据几何性质显然有 $L_1(x_0) = L_2(x_0)$, L_0 表示这一对相等的长度.

我们不妨假设 $\hat{\phi}_i \in (0, \frac{\pi}{2})$, 记函数 $F_1(x) = \tan(\hat{\phi}_i)x^2 - 2x - \tan(\hat{\phi}_i)$, $F_2(x) = \tan(\hat{\phi}_i)x^2 + 2x - \tan(\hat{\phi}_i)$, 两个函数的图像如图 (3.3), 两函数共有四个零点, 我们可以求得它们从小到大分别为

$$-\frac{1}{\tan(\frac{\hat{\phi}_i}{2})}, -\tan(\frac{\hat{\phi}_i}{2}), \tan(\frac{\hat{\phi}_i}{2}), \frac{1}{\tan(\frac{\hat{\phi}_i}{2})}.$$

如果 L/h 的值在图中 AB 或 CD 之间, 即若 $\tan(\frac{\hat{\phi}_i}{2}) < |L/h| < \frac{1}{\tan(\frac{\hat{\phi}_i}{2})}$, 则 $\Phi(x_1)\Phi(x_2) < 0$, 那么方程 (27) 显然有解. 如果 $|L/h| > 1/\tan(\frac{\hat{\phi}_i}{2})$, 那么 $\Phi(x_1), \Phi(x_2) > 0$, 而 $\Phi(x_0) < 0$, 从而方程 (27) 仍然有解. 也就是说, 只要 $|L/h| > \tan(\frac{\hat{\phi}_i}{2})$, 解就一定存在, 由于我们可以细分多边形使得 $\hat{\phi}_i < \pi/2$, 从而 $\tan(\hat{\phi}_i/2) < 1$, 结论成立. \square

求得了满足条件的 x 后, 可以代入求得 $\hat{\mathbf{P}}_i$ 的坐标, 用新坐标更新顶点序列的 \mathbf{P}_i , 之后对于下一个顶点重复上述步骤. 最后计算 $\mathbf{P}_1, \mathbf{P}_{n-1}$ 的值, 我们计算当前多边形的平均边长 \bar{L} , 令 $\mathbf{P}_1 = \mathbf{P}_0 + \bar{L}\mathbf{T}_S$, $\mathbf{P}_{n-1} = \mathbf{P}_n - \bar{L}\mathbf{T}_E$ 计算出所有顶点.



每当对所有顶点完成一次迭代, 我们计算当前的角度误差量

$$E_\theta = \max_{2 \leq i \leq n-2} |2\theta_i - \theta_{i-1} - \theta_{i+1}|$$

$$E_\phi = \max_{2 \leq i \leq n-2} |2\phi_i - \phi_{i-1} - \phi_{i+1}|$$
(29)

以及所有边长长度的标准差 σ_L , 为这三个量分别设定阈值, 不断迭代直到三个量均小于阈值, 此时可以认为多边形是近似的 Euler 多边形. 迭代的过程中顶点数是不会增加的, 我们在每一次迭代至满足要求后对多边形升阶, 再迭代得到更逼近的结果, 升阶的方法为

$$\hat{P}_i = P_{i-1} \cdot \frac{i}{n+1} + P_i \cdot \left(1 - \frac{i}{n+1}\right), i = 0, 1, \dots, n+1$$
(30)

经过迭代和升阶, 最终可以得到满足要求的曲线, 该算法的伪代码表示见算法 (2)

3.4 空间 Euler B-Spline 曲线

我们只要适当修改上面算法中对于边界的处理就可以构造空间 Euler B-Spline 曲线. 输入初始的控制多边形 $P_0 \dots P_n$ 后, 我们依然按照算法 (2) 中迭代的方法计算 P_2, \dots, P_{n-2} , 我们只需要修改对于剩余四个边界点的计算.

注意 Euler B-Spline 的定义是均匀节点的 B 样条, 为了满足 G^1 条件, 我们要有

$$\frac{P_0 + 4P_1 + P_2}{6} = P_S$$

$$P_2 - P_0 = \lambda T_S$$
(31)

其中 $\lambda > 0$ 待定, 由于 $\|P_0 P_1\| = \|P_1 P_2\|$, 我们可以得到关于 λ 的方程

$$\left\| \frac{3}{2} P_2 P_S + \frac{5\lambda}{4} T_S \right\| = \left\| \frac{3}{2} P_2 P_S + \frac{\lambda}{4} T_S \right\|$$
(32)

求解该方程得到 $\lambda = 2P_S P_2 \cdot T_S$, 从而计算得到边界点

$$P_0 = P_2 - (2P_S P_2 \cdot T_S) T_S$$

$$P_1 = \frac{3}{2} P_S - \frac{1}{2} P_2 + \frac{P_S P_2 \cdot T_S}{2} T_S$$
(33)

Data: 初始的三维控制多边形 $P_0 \dots P_m$, 最大迭代次数 \max_n .

Result: 与初始曲线有相同 G^1 边值条件的 Euler Bézier 曲线.

```
1  $n = m - 1$ ;  
2 求旋转矩阵  $R$ , 使旋转后的两端切向与  $xOy$  平面平行;  
3 将旋转后的多边形重新记为  $P_0 \dots P_m$ , 用  $T_S, T_E$  表示起终点单位切向;  
4 声明两个角度序列 AngleTheta[n], AnglePhi[n];  
5 AngleTheta[0]=0.0, AngleTheta[n-1]=0.0;  
6 AnglePhi[0]=0.0, AnglePhi[n-1]=0.0;  
7 for ( $i = 1; i < n - 1; i = i + 1$ ) do  
8   | 根据式 (17) 计算 AngleTheta[i], AnglePhi[i];  
9 end  
10 while ( $n \leq \max\_n$ ) do  
11   | E_theta = 10.0, E_phi = 10.0, sigma_L = 10.0;  
12   | s_count = 1;  
13   | max_count = 10000;  
14   | while  
15     | ( $s\_count < \max\_count \&\& (E\_theta > 1e - 6 || E\_phi > 1e - 6 || sigma\_L > 1e - 6)$ )  
16     | do  
17       | for ( $i = 2; i < n - 2; i = i + 1$ ) do  
18         | AngleTheta[i] = (AngleTheta[i-1]+AngleTheta[i]+AngleTheta[i+1])/3;  
19         | AnglePhi[i] = (AnglePhi[i-1]+AnglePhi[i]+AnglePhi[i+1])/3;  
20         | 根据式 (21),(22) 计算相关量;  
21         | 求解方程 (27), 更新  $P_i$ ;  
22       | end  
23       | s_count +=1;  
24       | 根据式 (29) 计算 E_theta , E_phi;  
25       | 计算边长的标准差 sigma_L;  
26     | end  
27   |  $n++$ ;  
28 end  
29 将得到的多边形作  $R$  的逆变换, 得到满足条件的曲线.
```

Algorithm 2: 3D Euler Bézier 曲线

同样的过程求得另一端的边界点

$$\begin{aligned} P_n &= P_{n-2} - (2P_E P_{n-2} \cdot T_E) T_E \\ P_{n-1} &= \frac{3}{2} P_E - \frac{1}{2} P_{n-2} + \frac{P_E P_{n-2} \cdot T_E}{2} T_E \end{aligned} \quad (34)$$

其余的过程和构造 Euler Bézier 曲线是一致的, 两个方法的主要原理都是 Bézier 或 B-Spline 曲线会随着控制顶点的增多逐渐逼近控制多边形, 从而逼近我们定义的 Euler 多边形.

4 实验结果

图 (4.1) 展示了七段曲线拼接而成的一段 G^1 连续的空间曲线, 其曲率是单调增加的, 形状接近圆锥螺线. 使用 Bézier 和 B-Spline 两种方法得到的结果无法直接观察出差异, 图 (4.2) 和图 (4.3) 均匀采样了曲线的曲率和挠率, 可以看出三次的 B-Spline 曲线挠率是不连续的, 每一段 Bézier 曲线挠率是连续单调的, 两者的曲率接近 G^2 连续, 且整体是单调的.

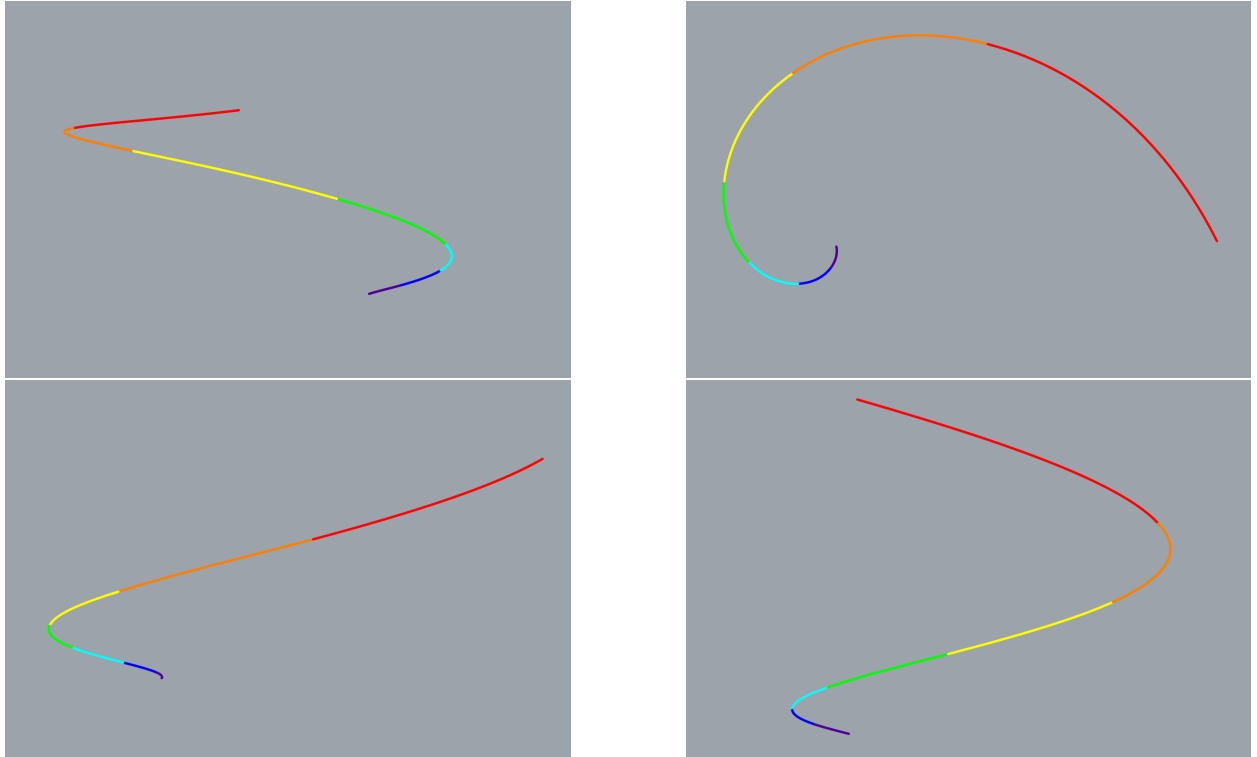


Figure 4.1: 七段 Euler Bézier 曲线 G^1 拼接而成.

参考文献

- [1] 第一篇文献.
- [2] 第二篇文献.

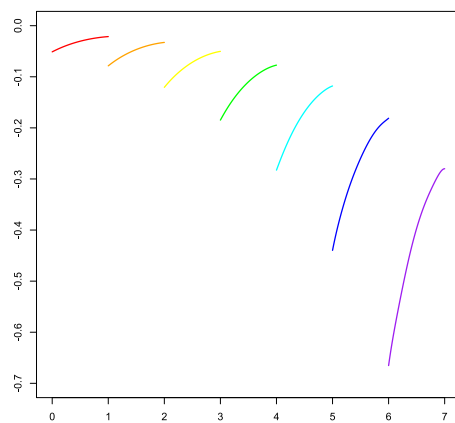
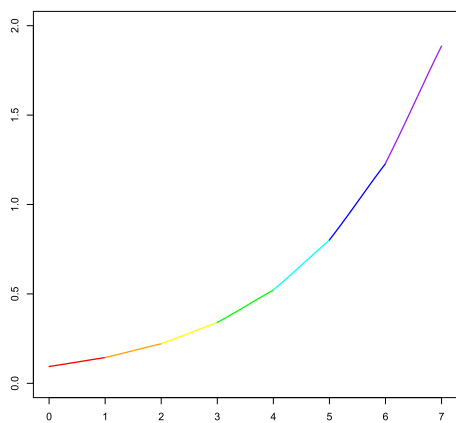


Figure 4.2: 左: 图 (4.1) 中七段 Bézier 曲线的曲率; 右: 图 (4.1) 中七段 Bézier 曲线的挠率.

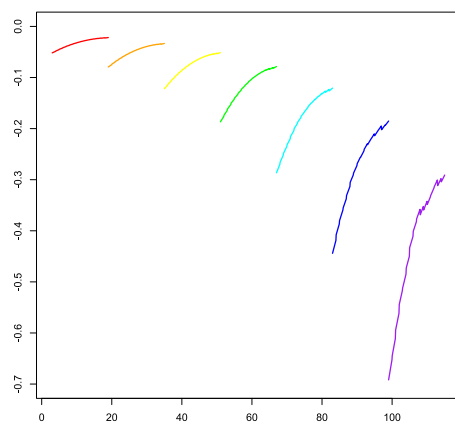
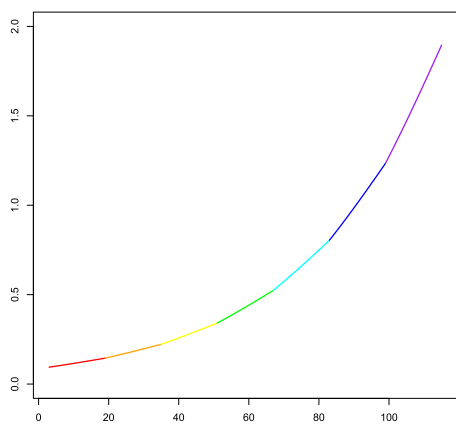


Figure 4.3: 左: 图 (4.1) 中七段 B-spline 曲线的曲率; 右: 图 (4.1) 中七段 B-Spline 曲线的挠率.

[3] 第三篇文献.

[4] 第四篇文献.