CAGD(2) | Bezier曲线

1. 曲线的表示

1.1. 隐式表示

曲线隐式表示为$f(x, y) = 0$，该方法有诸多局限性：

- 对于同一个x横坐标值对应多个纵坐标值
- 存在一些位置，导数$\frac{\mathrm{d}y}{\mathrm{d}x}$没有定义
- 关于轴变换非不变

1.2. 参数表示

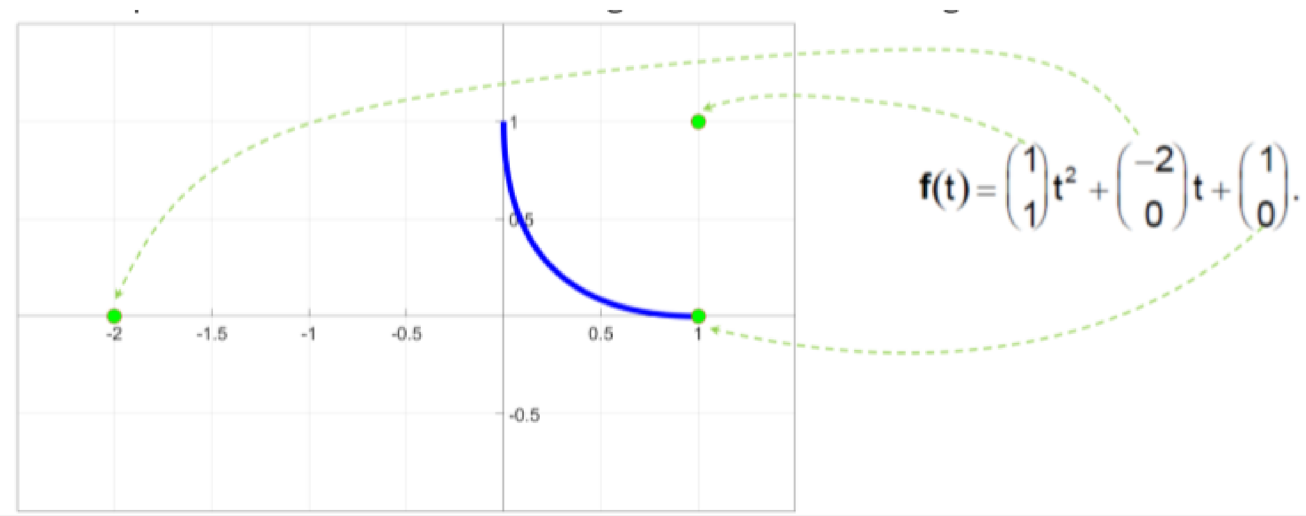
曲线参数表示为$c(t) = (x(t), y(t))$

- 求值方便
- 参数t可作为时间进行插值
- 曲线可以理解为运动粒子的运动轨迹跟踪

2. 曲线建模举例

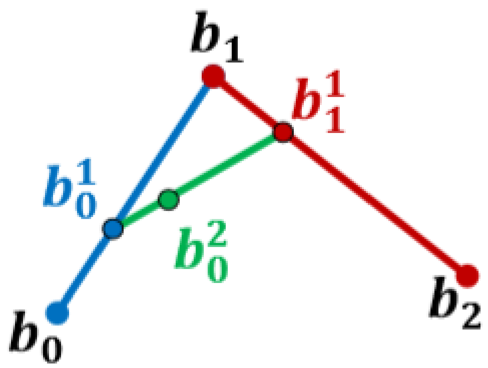
2.1. 用幂函数基进行建模

以抛物线$\boldsymbol{f}(t)=\boldsymbol{a}t^2+\boldsymbol{b}t+\boldsymbol{c}$为例



- 幂函数基的系数缺少直觉上的几何意义

2.2. 一种改进的画法



其中，

- $\boldsymbol{b}_0^1=(1-t)\boldsymbol{b}_0+t\boldsymbol{b}_1$
- $\boldsymbol{b}_1^1=(1-t)\boldsymbol{b}_1+t\boldsymbol{b}_2$

$$\begin{aligned}\boldsymbol{b}_0^2&=(1-t)\boldsymbol{b}_0^1+t\boldsymbol{b}_1^1\\&=(1-t)[(1-t)\boldsymbol{b}_0+t\boldsymbol{b}_1]+t[(1-t)\boldsymbol{b}_1+t\boldsymbol{b}_2]\\&=(1-t)^2\boldsymbol{b}_0+2t(1-t)\boldsymbol{b}_1+t^2\boldsymbol{b}_2\end{aligned}$$

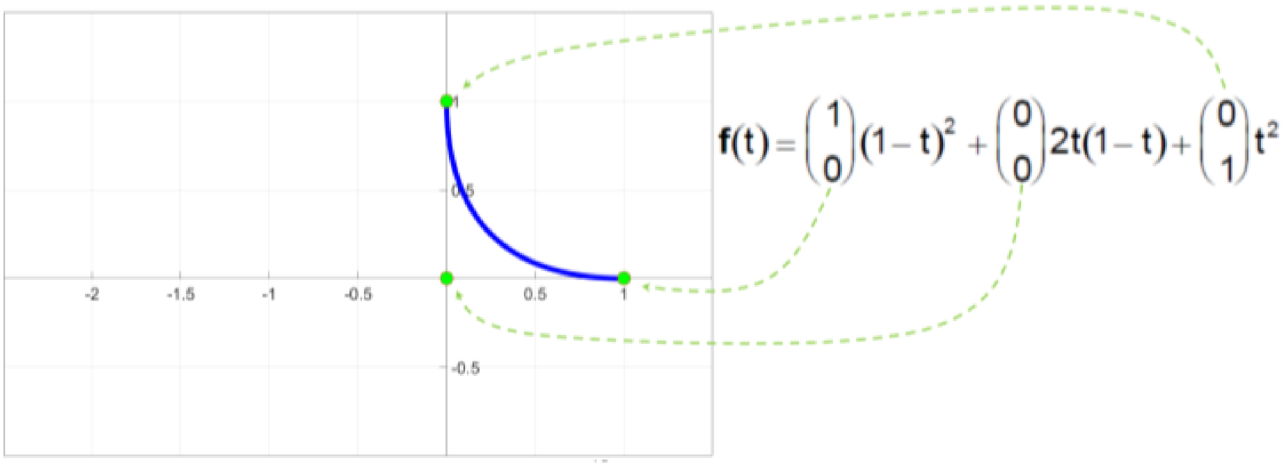
以之前的例子

$$\boldsymbol{f}(t)=\begin{pmatrix}1\\1\end{pmatrix}t^2+\begin{pmatrix}-2\\0\end{pmatrix}t+\begin{pmatrix}1\\0\end{pmatrix}$$

转为上述形式：

$$\boldsymbol{f}(t)=\begin{pmatrix}1\\0\end{pmatrix}(1-t)^2+\begin{pmatrix}0\\0\end{pmatrix}2t(1-t)+\begin{pmatrix}0\\1\end{pmatrix}t^2$$

在图上表示为



- 各系数均有相应的几何意义
- 更多直觉上的曲线操作

对于四个控制点，同样有：

$$\boldsymbol{p}_0^0(t) = \boldsymbol{p}_0,\quad \boldsymbol{p}_1^0(t) = \boldsymbol{p}_1,\quad \boldsymbol{p}_2^0(t) = \boldsymbol{p}_2,\quad \boldsymbol{p}_3^0(t) = \boldsymbol{p}_3$$

第一次迭代：

$$\begin{aligned}\boldsymbol{p}_0^1 &= (1-t)\boldsymbol{p}_0 + t\boldsymbol{p}_1\\ \boldsymbol{p}_1^1 &= (1-t)\boldsymbol{p}_1 + t\boldsymbol{p}_2\\ \boldsymbol{p}_2^1 &= (1-t)\boldsymbol{p}_2 + t\boldsymbol{p}_3\end{aligned}$$

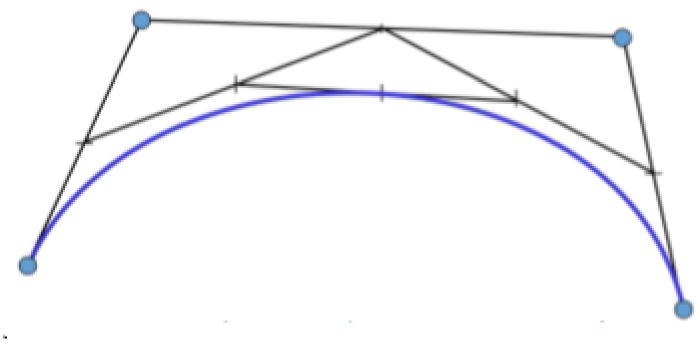
第二次迭代：

$$\begin{aligned}\boldsymbol{p}_0^2 &= (1-t)^2\boldsymbol{p}_0 + 2t(1-t)\boldsymbol{p}_1 + t^2\boldsymbol{p}_2\\ \boldsymbol{p}_1^2 &= (1-t)^2\boldsymbol{p}_1 + 2t(1-t)\boldsymbol{p}_2 + t^2\boldsymbol{p}_3\end{aligned}$$

最终得到的曲线方程：

$$\boldsymbol{c}(t) = (1-t)^3\boldsymbol{p}_0 + 3t(1-t)^2\boldsymbol{p}_1 + 3t^2(1-t)\boldsymbol{p}_2 + t^3\boldsymbol{p}_3$$

3. De Casteljau算法



3.1. 动机

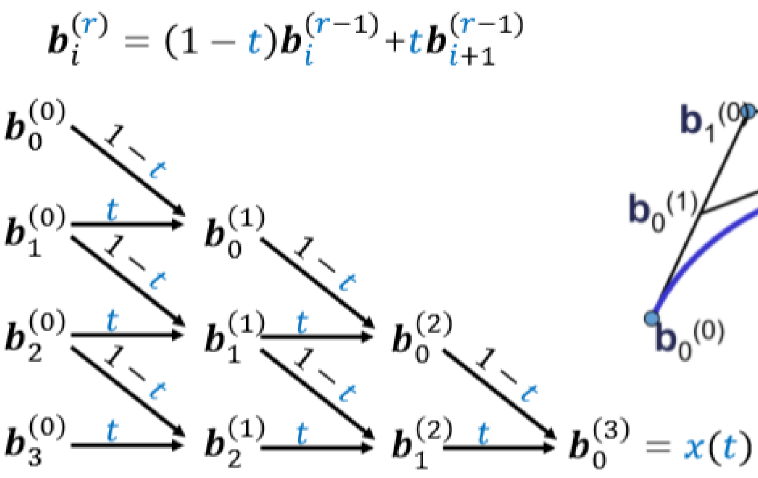
- 对给定t计算$\boldsymbol{x}(t)$
  - 按比例$t:(1-t)$平分控制多边形
  - 用线连接新点（相邻线段）
  - 用相同比例进行插值
  - 迭代，直到只剩下一个点

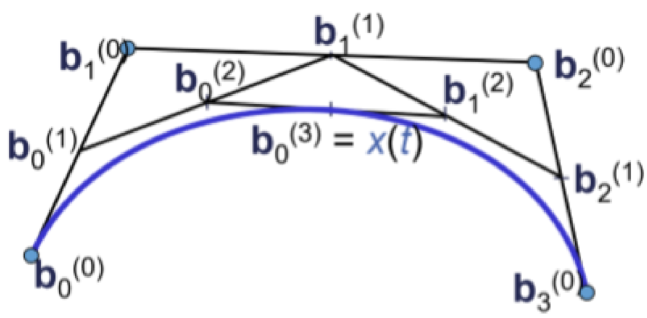
3.2. 算法描述

- 输入点：$\boldsymbol{b}_0, \boldsymbol{b}_1, \cdots, \boldsymbol{b}_n \in \mathbb{R}^3$
- 输出曲线：$\boldsymbol{x}(t), t \in [0,1]$
- 对给定t进行点$\boldsymbol{x}(t)$的几何构造

$$
\begin{aligned}
&\boldsymbol{b}_i^0(t) = \boldsymbol{b}_i, \quad i = 0, \cdots, n \\
&\boldsymbol{b}_i^r(t) = (1-t)\boldsymbol{b}_i^{r-1}(t) + t\boldsymbol{b}_{i+1}^{r-1}(t) \\
&r = 1, \cdots, n \quad i = 0, \cdots, n-r
\end{aligned}
$$

- 最后，$\boldsymbol{b}_0^n(t)$为所找的曲线点$\boldsymbol{x}(t)$在参数值t的取值





所有系数可写为下三角矩阵：

$$
\begin{array}{lllllll}
\boldsymbol{b}_0 = \boldsymbol{b}_0^0 & & & & & & \\
\boldsymbol{b}_1 = \boldsymbol{b}_1^0 & \boldsymbol{b}_0^1 & & & & & \\
\boldsymbol{b}_2 = \boldsymbol{b}_2^0 & \boldsymbol{b}_1^1 & \boldsymbol{b}_0^2 & & & & \\
\boldsymbol{b}_3 = \boldsymbol{b}_3^0 & \boldsymbol{b}_2^1 & \boldsymbol{b}_1^2 & \boldsymbol{b}_0^3 & & & \\
\quad\vdots & \vdots & \vdots & \ddots & \ddots & & \\
\boldsymbol{b}_{n-1} = \boldsymbol{b}_{n-1}^0 & \boldsymbol{b}_{n-2}^1 & \cdots & \cdots & \cdots & \boldsymbol{b}_0^{n-1} & \\
\boldsymbol{b}_n = \boldsymbol{b}_n^0 & \boldsymbol{b}_{n-1}^1 & \cdots & \cdots & \cdots & \boldsymbol{b}_1^{n-1} & \boldsymbol{b}_0^n = \boldsymbol{x}(t)
\end{array}
$$

伪代码：

Algorithm:

for r=1..n

  for i=0..n-r

$$\boldsymbol{b}_i^{(r)} = (1-t)\,\boldsymbol{b}_i^{(r-1)} + t\,\boldsymbol{b}_{i+1}^{(r-1)}$$

  end

end

return $\boldsymbol{b}_0^{(n)}$

The whole algorithm consists only of repeated linear interpolations.

3.3. De Casteljau算法性质

- 包含点$\boldsymbol{b}_0,\cdots,\boldsymbol{b}_n$的多边形称为**Bezier多边形**
- 点$\boldsymbol{b}_i$称为**Bezier点**（控制点）
- 由Bezier点$\boldsymbol{b}_0,\cdots,\boldsymbol{b}_n$和De Casteljau算法所定义的曲线称为Bezier曲线
- De Casteljau算法是数值稳定的，因为只使用了凸组合
- De Casteljau算法复杂度
  - 时间复杂度$O(n^2)$
  - 空间复杂度$O(n)$
  - 其中n为Bezier点的数量
- **Bezier曲线的性质**
  - 给定Bezier点$\boldsymbol{b}_0,\cdots,\boldsymbol{b}_n$和Bezier曲线$\boldsymbol{x}(t)$
  - Bezier曲线是n阶多项式曲线
  - 端点$\boldsymbol{x}(0)=\boldsymbol{b}_0,\boldsymbol{x}(1)=\boldsymbol{b}_n$插值，其余的Bezier点仅仅是大致的近似值
  - 凸包性质：
    - Bezier曲线完全在其Bezier多边形的凸包内部
  - 变化减少
    - 没有直线与Bezier曲线的交点比Bezier多边形多
  - Bezier点的影响：全局但伪局部
    - 全局：移动Bezier点会改变整个曲线的形状
    - 伪局部：点$\boldsymbol{b}_i$对$x(t)$在$t=\frac{i}{n}$有最大的影响
  - 仿射不变性
    - Bazier曲线和Bazier多边形在仿射变换下不变
  - 仿射参数变换不变性
  - 对称性
    - 以下两条Bezier曲线重合，它们仅在相反的方向上移动：

$$\boldsymbol{x}(t)=[\boldsymbol{b}_0,\cdots,\boldsymbol{b}_n] \quad \boldsymbol{x}'(t)=[\boldsymbol{b}_n,\cdots,\boldsymbol{b}_0]$$

  - 线性精确
    - 当$\boldsymbol{b}_0,\cdots,\boldsymbol{b}_n$共线时，Bazier曲线为线段
  - 重心组合下的不变性

4. Bazier曲线

Bazier曲线表示为基函数组合：

$$\boldsymbol{x}(t)=\sum_{i=0}^{n}\boldsymbol{B}_i^n(t)\cdot b_i$$

4.1. 期望特性

- 对基底的要求：
  - 良好性质的曲线
    - **光滑**的基函数
  - **局部控制**（或者至少半局部）
    - 紧致的基函数
  - **仿射不变性**
    - 对控制点或曲线进行仿射变换$\boldsymbol{x}=A\boldsymbol{x}+b$应该有相同的效果
    - 例如：旋转、平移
    - 否则交互式编辑曲线将非常困难
  - **凸包性质**
    - 曲线处于其控制点的凸包内
    - 至少能够避免奇怪的震荡
  - 优点
    - 计算优势（递归相交测试）
    - 更多可预测的行为

4.2. 仿射不变性

- 仿射变换：$\boldsymbol{x}\to A\boldsymbol{x}+\boldsymbol{b}$

4.2.1. 线性不变性

Bazier曲线的线性不变性是显然的，Bazier曲线表示为基函数的线性组合：

$$\boldsymbol{f}(t)=\sum_{i=1}^{n}b_i(t)\boldsymbol{p}_i=\sum_{i=1}^{n}b_i(t)\begin{pmatrix}p_i^{(x)}\\p_i^{(y)}\\p_i^{(z)}\end{pmatrix}$$

因此

$$A(\boldsymbol{f}(t))=A\left(\sum_{i=1}^{n}b_i(t)\boldsymbol{p}_i\right)=\sum_{i=1}^{n}b_i(t)(A\boldsymbol{p}_i)$$

4.2.2. 平移不变性

$$\sum_{i=1}^{n}b_i(t)(\boldsymbol{p}_i+\boldsymbol{b})=\sum_{i=1}^{n}b_i(t)\boldsymbol{p}_i+\sum_{i=1}^{n}b_i(t)\boldsymbol{b}=\boldsymbol{f}(t)+\left(\sum_{i=1}^{n}b_i(t)\right)\boldsymbol{b}$$

- 为了满足平移不变性，基函数的和应恒为1
- 这也称为"partition of unity property"，单位划分性质
- $\boldsymbol{b}_i$是控制点$\boldsymbol{p}_i$的“仿射组合”
- 该性质对建模非常重要

4.3. 凸包性质

- **凸组合**
    - 点集$\{\boldsymbol{p}_1, \cdots, \boldsymbol{p}_n\}$的一个凸组合为如下形式

$$\sum_{i=1}^{n} \lambda_i \boldsymbol{p}_i \text{ with} \sum_{i=1}^{n} \lambda_i = 1 \text{ and } \forall i = 1, \cdots, n : 0 \le \lambda_i \le 1$$

    - 所有允许的凸组合的集合形成点集的凸包
        - 凸包是包含所有点集$\{\boldsymbol{p}_1, \cdots, \boldsymbol{p}_n\}$以及集合中两个元素之间的每条完整连线的最小集合
- 相应地
    - 如果我们有性质：

$$\forall t \in \Omega : \sum_{i=1}^{n} b_i(t) = 1 \text{ and } \forall t \in \Omega, \forall i : b_i(t) \ge 0$$

    所构造地曲线/曲面将满足：
        - 仿射不变性（平移，线性映射）
        - 限制在控制点的凸包中
    - 推论：曲线将有**linear precision**（线性精确）
        - 当所有控制点共线时，曲线为直线段
        - 具有平面控制点的曲面也将是平面
- 凸包性质在实践中十分有用
    - 避免不良震荡
        - 被限制在凸包内，不像多项式插值
    - 线性精确性质比较直观（用户友好）
    - 可用于快速范围检查
        - 相交测试可以先对凸包进行，然后再对物体进行
        - 递归相交算法与细分规则结合使用

4.4. Bezier曲线的多项式描述

- 给定$(n+1)$个控制点$\boldsymbol{b}_0, \cdots, \boldsymbol{b}_n$
- 目标：Bazier曲线$\boldsymbol{x}(t)$，其中$t \in [0, 1]$
- 定义$n+1$个基函数，通过其线性组合来描述一个Bezier曲线

$$B_0^n(t), \cdots, B_n^n(t) \text{ over } [0, 1]$$

$$\boldsymbol{x}(t) = \sum_{i=0}^{n} B_i^n(t) \cdot \boldsymbol{b}_i$$

4.4.1. Bernstein基函数

Bernstein基函数：$B = \{B_0^{(n)}, B_1^{(n)}, \cdots, B_n^{(n)}\}$

- n次Bernstein基函数

$$B_i^{(n)}(t) = \binom{n}{i} t^i (1-t)^{n-i}$$

  其中，二项式系数

$$\binom{n}{i} = \begin{cases} \dfrac{n!}{(n-i)!!} & \text{for } 0 \le i \le n \\ 0 & \text{otherwise} \end{cases}$$

  杨氏三角：

1

1 1

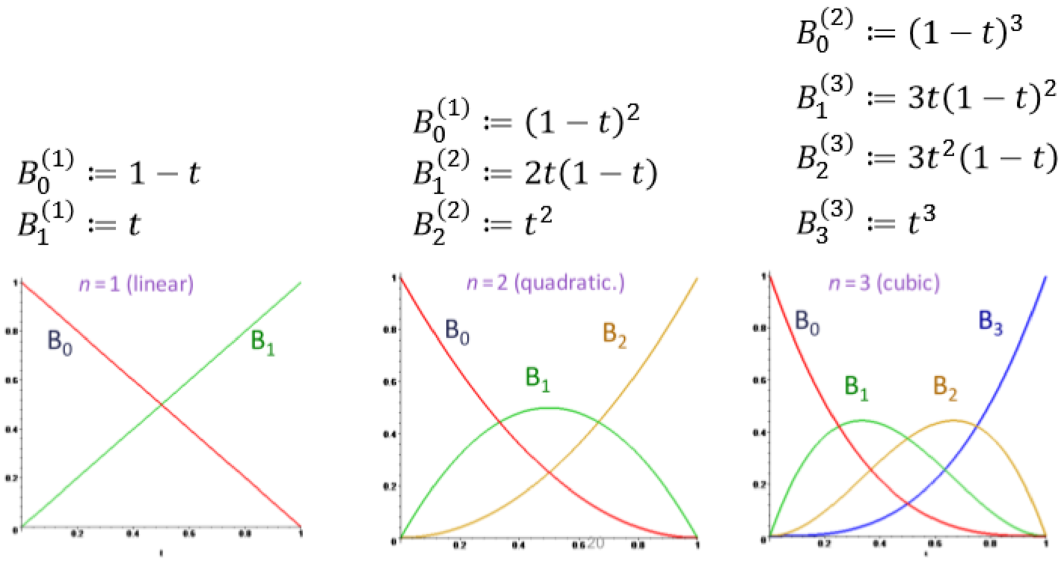
1 2 1

1 3 3 1

1 4 6 4 1

1 5 10 10 5 1

前三组Bernstein基函数



4.4.2. Bernstein基函数的性质

$$B=\{B_0^{(n)},B_1^{(n)},\cdots,B_n^{(n)}\},\quad B_i^{(n)}=\binom{n}{i}t^i(1-t)^{n-i}$$

光滑性

基函数为n次多项式——显然光滑

局部控制

每个基函数$B_i^{(n)}$在$t=\dfrac{i}{n}$处取最大值——对该处有最大影响

凸包性质和仿射不变性

组合数的性质:

$$\sum_{i=0}^{n}B_i^{(n)}(t)=(t+(1-t))^n=1$$

递归计算特性

$$B_i^n(t):=(1-t)B_i^{(n-1)}(t)+tB_{i-1}^{(n-1)}(1-t)$$
$$\text{with } B_0^0(t)=1, B_i^n(t)=0 \text{ for } i\notin\{0,\cdots,n\}$$

对称性

$$B_i^n(t)=B_{n-i}^n(1-t)$$

非负性

$$B_i^{(n)}(t) \geq 0 \text{ for } t \in [0, \cdots, 1]$$

$$B_i^{(n)}(t) > 0 \text{ for } 0 < t < 1$$

$$\begin{gathered}B_0^{(n)}(0) = 1, \quad B_1^{(n)}(0) = \cdots = B_n^{(n)}(0) = 0 \\ B_0^{(n)}(1) = \cdots = B_{n-1}^{(n)} = 0, \quad B_n^{(n)}(1) = 1\end{gathered}$$

导数

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t} B_i^{(n)}(t) &= \binom{n}{i} \left(it^{i-1}(1-t)^{n-i} - (n-i)t^i(1-t)^{n-i-1}\right) \\ &= \frac{n!}{(n-i)!i!} it^{i-1}(1-t)^{n-i} - \frac{n!}{(n-i)!i!}(n-i)t^i(1-t)^{n-i-1} \\ &= n\left[\binom{n-1}{i-1} t^{i-1}(1-t)^{n-i} - \binom{n-1}{i} t^i(1-t)^{n-i-1}\right] \\ &= n\left[B_{i-1}^{(n-1)}(t) - B_i^{(n-1)}(t)\right] \\ \frac{\mathrm{d}^2}{\mathrm{d}t^2} B_i^{(n)}(t) &= \frac{\mathrm{d}}{\mathrm{d}t}\left[B_{i-1}^{(n-1)}(t) - B_i^{(n-1)}(t)\right] \\ &= n\left[(n-1)\left(B_{i-2}^{(n-2)}(t) - B_{i-1}^{(n-2)}(t)\right) - (n-1)\left(B_{i-1}^{(n-2)}(t) - B_i^{(n-2)}(t)\right)\right] \\ &= n(n-1)\left[B_{i-2}^{(n-2)}(t) - 2B_{i-2}^{(n-2)}(t) + B_i^{(n-2)}(t)\right]\end{aligned}$$

4.5. Bezier曲线的性质

4.5.1. 前面提到过的性质：

- 仿射不变性
- 凸包性质
- 控制点影响性

4.5.2. 导数性质

对于$t \in [0,1]$，有

$$\begin{gathered}\boldsymbol{f}(t) = \sum_{i=0}^{n-1} \binom{n}{i} t^i (1-t)^{n-i} \boldsymbol{p}_i \\ \Rightarrow \boldsymbol{f}(0) = \boldsymbol{p}_0 \quad \boldsymbol{f}(1) = \boldsymbol{p}_1\end{gathered}$$

一阶导数

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{f}(t) &= n\sum_{i=0}^{n-1}\left[B_{i-1}^{(n-1)}(t) - B_i^{(n-1)}(t)\right]\boldsymbol{p}_i \\ &= n\left(\left[-B_0^{(n-1)}(t)\right]\boldsymbol{p}_0 + \left[B_0^{n-1}(t) - B_1^{(n-1)}(t)\right]\boldsymbol{p}_1 + \cdots\right)\end{aligned}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{f}(0) = n(\boldsymbol{p}_1 - \boldsymbol{p}_0) \quad \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{f}(1) = n(\boldsymbol{p}_n - \boldsymbol{p}_{n-1})$$

以此类推，对于边界点$\{0,1\}$，有

$$\begin{aligned}\boldsymbol{f}(0) &= \boldsymbol{p}_0 \\ \boldsymbol{f}(1) &= \boldsymbol{p}_1 \\ \boldsymbol{f}'(0) &= n[\boldsymbol{p}_1 - \boldsymbol{p}_0] \\ \boldsymbol{f}'(1) &= n[\boldsymbol{p}_n - \boldsymbol{p}_{n-1}] \\ \boldsymbol{f}''(0) &= n(n-1)[\boldsymbol{p}_2 - 2\boldsymbol{p}_1 + \boldsymbol{p}_0] \\ \boldsymbol{f}''(1) &= n(n-1)[\boldsymbol{p}_n - 2\boldsymbol{p}_{n-1} + \boldsymbol{p}_{n-2}]\end{aligned}$$

4.6. Bezier曲线升阶（Degree Evaluation）

- 给定：$\boldsymbol{b}_0, \cdots, \boldsymbol{b}_n \rightarrow \boldsymbol{x}(t)$
- 目标：$\overline{\boldsymbol{b}}_0, \cdots, \overline{\boldsymbol{b}_n}, \overline{\boldsymbol{b}}_{n+1} \rightarrow \overline{\boldsymbol{x}}(t)$ with $\boldsymbol{x} = \overline{\boldsymbol{x}}$
- 解决方法：

$$
\begin{aligned}
\overline{\boldsymbol{b}}_0 &= \boldsymbol{b}_0 \\
\overline{\boldsymbol{b}}_{n+1} &= \boldsymbol{b}_n \\
\overline{\boldsymbol{b}}_j &= \frac{j}{n+1}\boldsymbol{b}_{j-1} + \left(1 - \frac{j}{n+1}\right)\boldsymbol{b}_j \text{ for } j = 1, \cdots, n
\end{aligned}
$$

- 证明：

  考虑

$$
\begin{aligned}
(1-t)B_i^n(t) &= (1-t)\binom{n}{i}(1-t)^{n-i}t^i \\
&= \binom{n}{i}(1-t)^{n+1-i}t^i \\
&= \frac{n+1-i}{n+1}\binom{n+1}{i}(1-t)^{n+1-i}t^i \\
&= \frac{n+1-i}{n+1}B_i^{n+1}(t)
\end{aligned}
$$

  类似地，

$$
tB_i^n(t) = \frac{i+1}{n+1}B_i^{n+1}(t)
$$

  从而有：

$$
\begin{aligned}
\boldsymbol{f}(t) &= [(1-t)+t]\boldsymbol{f}(t) \\
&= [(1-t)+t]\sum_{i=0}^{n} B_i^n(t)\boldsymbol{P}_i \\
&= \sum_{i=0}^{n}\left[(1-t)B_i^n(t) + tB_i^n(t)\right]\boldsymbol{P}_i \\
&= \sum_{i=0}^{n}\left[\frac{n+1-i}{n+1}B_i^{n+1}(t) + \frac{i+1}{n+1}B_{i+1}^{n+1}(t)\right]\boldsymbol{P}_i \\
&= \sum_{i=0}^{n}\frac{n+1-i}{n+1}B_i^{n+1}(t)\boldsymbol{P}_i + \sum_{i=0}^{n}\frac{i+1}{n+1}B_{i+1}^{n+1}(t)\boldsymbol{P}_i \\
&= \sum_{i=0}^{n}\frac{n+1-i}{n+1}B_i^{n+1}(t)\boldsymbol{P}_i + \sum_{i=1}^{n}\frac{i}{n+1}B_i^{n+1}(t)\boldsymbol{P}_{i-1} \\
&= \sum_{i=0}^{n+1}\frac{n+1-i}{n+1}B_i^{n+1}(t)\boldsymbol{P}_i + \sum_{i=0}^{n}\frac{i}{n+1}B_i^{n+1}(t)\boldsymbol{P}_{i-1} \\
&= \sum_{i=0}^{n+1}B_i^{n+1}(t)\left[\frac{n+1-i}{n+1}\boldsymbol{P}_i + \frac{i}{n+1}\boldsymbol{P}_{i-1}\right]
\end{aligned}
$$

4.7. 细分

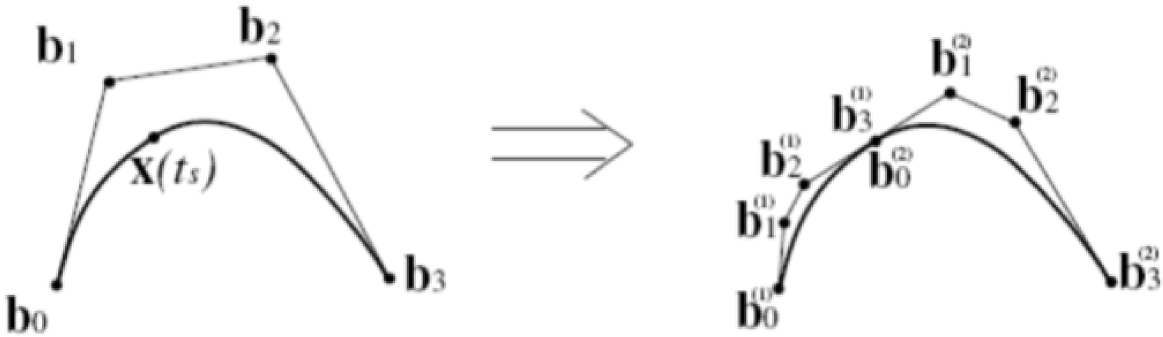
- 给定

$$
\boldsymbol{b}_0, \cdots, \boldsymbol{b}_n \rightarrow \boldsymbol{x}(t), t \in [0,1]
$$

- 目标

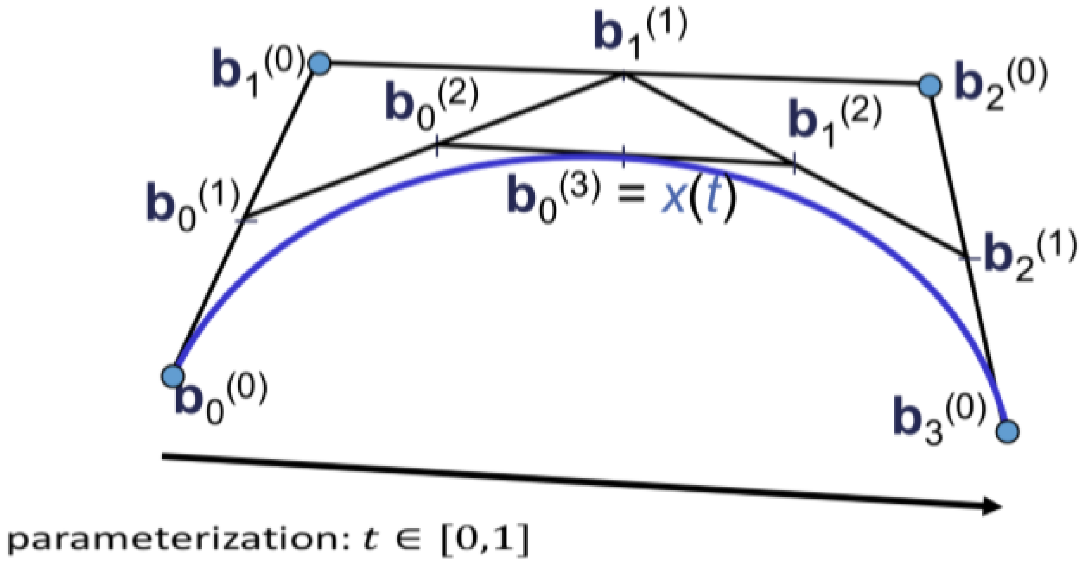
$$
\begin{gathered}
\boldsymbol{b}_0^{(1)}, \cdots, \boldsymbol{b}_n^{(1)} \rightarrow \boldsymbol{x}^{(1)}(t) \\
\boldsymbol{b}_0^{(2)}, \cdots, \boldsymbol{b}_n^{(2)} \rightarrow \boldsymbol{x}^{(2)}(t) \\
\text{with} \quad \boldsymbol{x} = \boldsymbol{x}^{(1)} \cup \boldsymbol{x}^{(2)}
\end{gathered}
$$

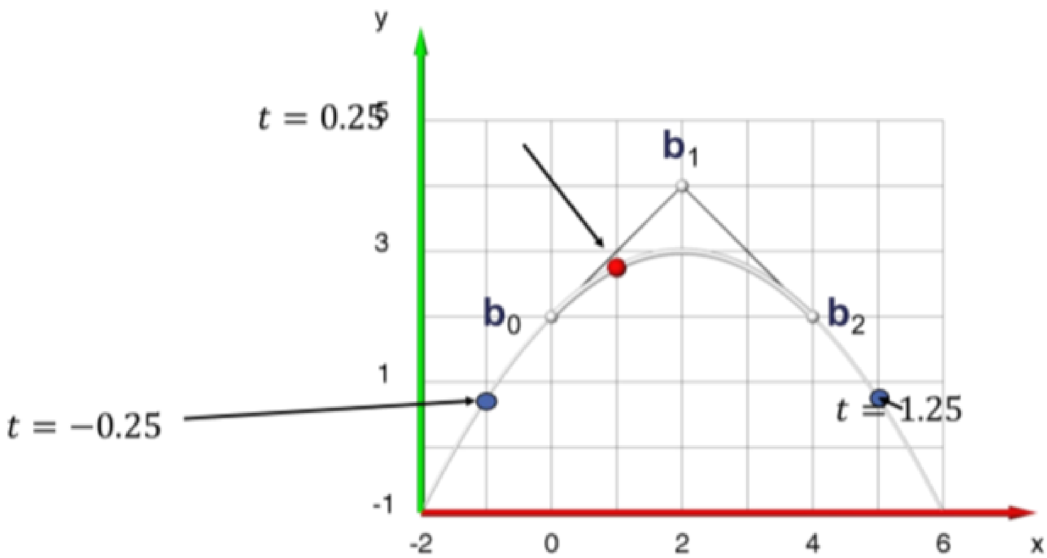
- 解决方法



$$b_i^{(1)} = b_0^i, b_i^{(2)} = b_0^{n-i} \text{ for } i = 0, \cdots, n$$

4.8. 曲线范围





4.9. 矩阵实现

三次Bazier曲线

$$P(t) = V_0 B_{0,3} + V_1 B_{1,3} + V_2 B_{2,3} + V_3 B_{3,3}$$

$$
\begin{aligned}
B_{0,3} &= \frac{3!}{0!3!}t^0(1-t)^3 = (1-t)^3 \\
B_{1,3} &= \frac{3!}{1!2!}t^1(1-t)^2 = 3t(1-t)^2 \\
B_{2,3} &= \frac{3!}{2!1!}t^2(1-t)^1 = 3t^2(1-t) \\
B_{3,3} &= \frac{3!}{3!0!}t^3(1-t)^0 = t^3
\end{aligned}
$$

矩阵形式表示：

$$
\boldsymbol{P}(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{pmatrix} \begin{bmatrix} V_{i-1} \\ V_i \\ V_{i+1} \\ V_{i+2} \end{bmatrix}
$$

导数的矩阵表示：

$$
\boldsymbol{P}'(t) = \begin{bmatrix} 3t^2 & 2t & 1 & 0 \end{bmatrix} \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{pmatrix} \begin{bmatrix} V_{i-1} \\ V_i \\ V_{i+1} \\ V_{i+2} \end{bmatrix}
$$