

Computer Aided Geometric Design

Fall Semester 2025

Bézier Curves (continue)

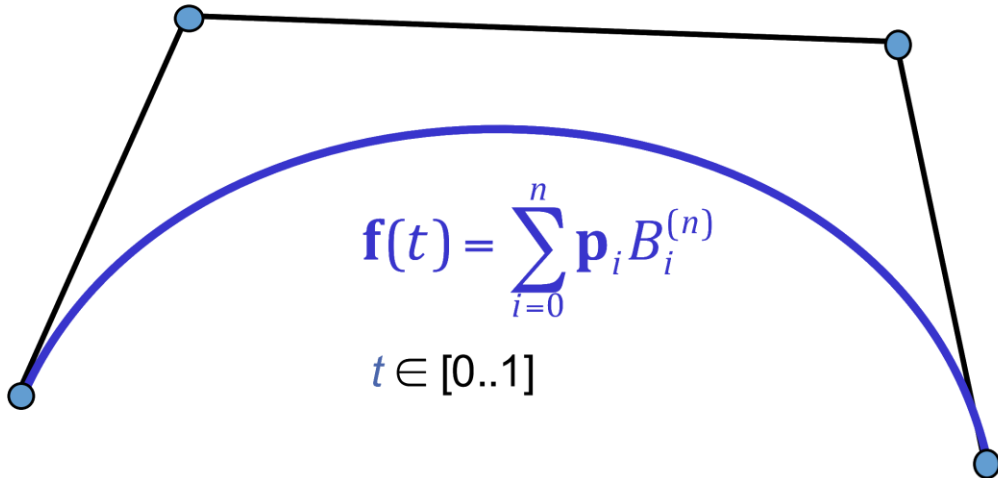
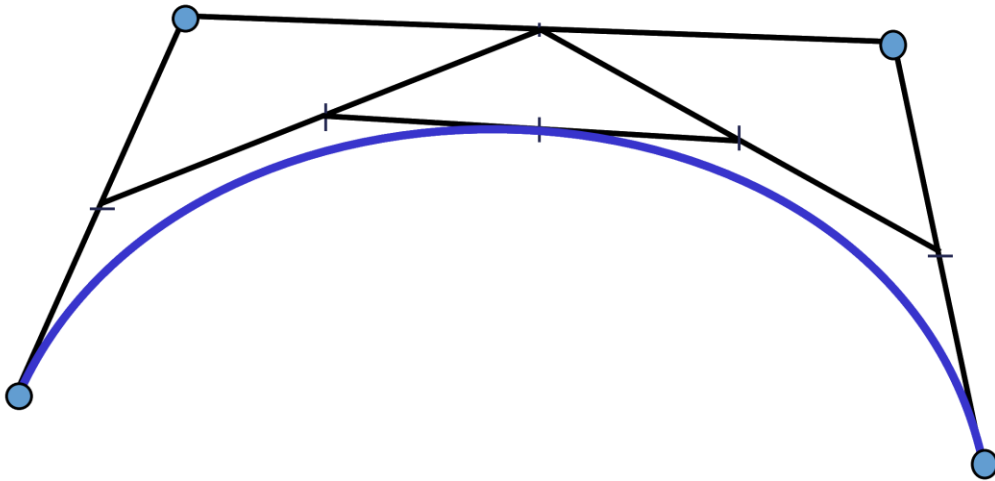
陈仁杰

renjiec@ustc.edu.cn

<http://staff.ustc.edu.cn/~renjiec>

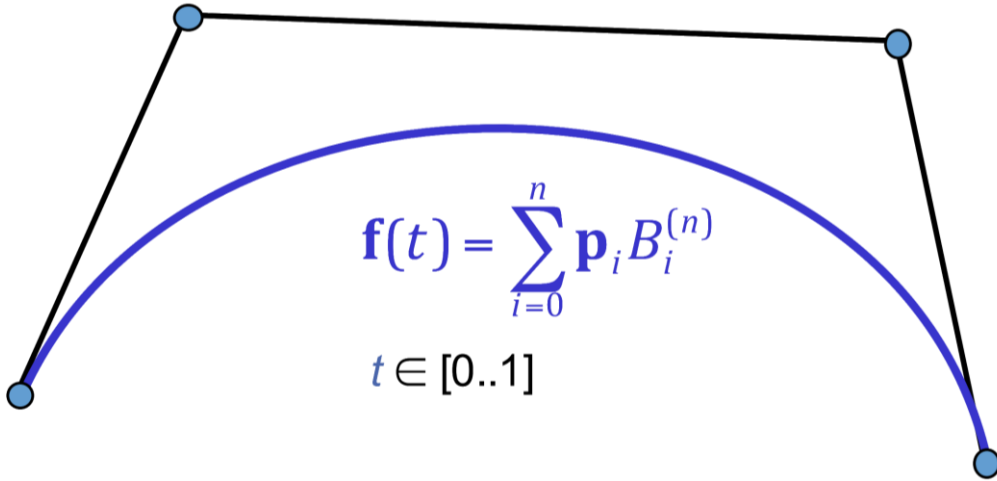
Recap

de Casteljau algorithm



Bernstein form

Recap



Bernstein form

Curve basis function control points

$\mathbf{f}(t) = \sum_{i=0}^n B_i^{(n)}(t) \mathbf{p}_i$

Dashed arrows point from the labels to the corresponding parts of the equation: 'Curve' to $\mathbf{f}(t)$, 'basis function' to $B_i^{(n)}(t)$, and 'control points' to \mathbf{p}_i .

Useful properties for basis functions

- Smoothness
- Local control / support
- Affine invariance
- Convex hull property

Degree elevation

- Given: $b_0, \dots, b_n \rightarrow x(t)$
- Wanted: $\bar{b}_0, \dots, \bar{b}_n, \bar{b}_{n+1} \rightarrow \bar{x}(t)$ with $x = \bar{x}$
- Solution:

Degree elevation

- Given: $\mathbf{b}_0, \dots, \mathbf{b}_n \rightarrow \mathbf{x}(t)$
- Wanted: $\bar{\mathbf{b}}_0, \dots, \bar{\mathbf{b}}_n, \bar{\mathbf{b}}_{n+1} \rightarrow \bar{\mathbf{x}}(t)$ with $\mathbf{x} = \bar{\mathbf{x}}$
- Solution:

$$\bar{\mathbf{b}}_0 = \mathbf{b}_0$$

$$\bar{\mathbf{b}}_{n+1} = \mathbf{b}_n$$

$$\bar{\mathbf{b}}_j = \frac{j}{n+1} \mathbf{b}_{j-1} + \left(1 - \frac{j}{n+1}\right) \mathbf{b}_j \quad \text{for } j = 1, \dots, n$$

Proof

- Let's consider

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

Similarly

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

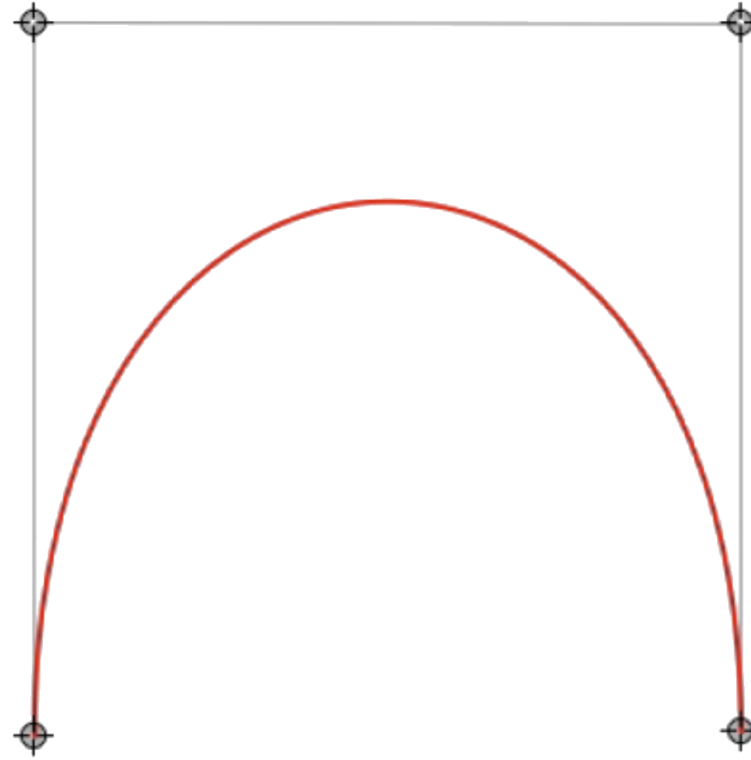
Proof

Using results from
last slide

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

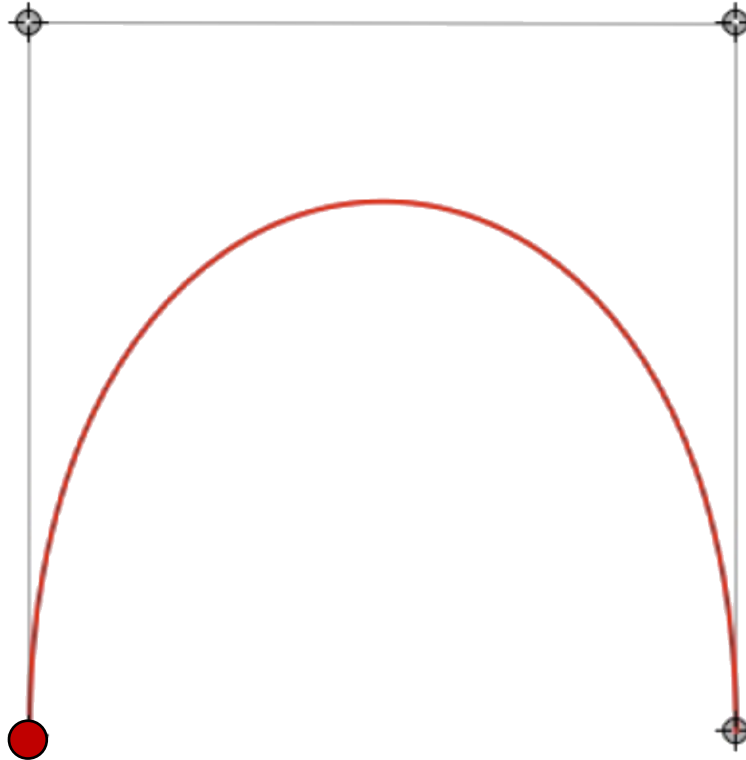
Adding null terms, $i = n+1, i = 0$

Degree elevation: Example



- $\bar{b}_0 = b_0$
 - $\bar{b}_{n+1} = b_n$
- $$\bar{b}_j = \frac{j}{n+1} b_{j-1} + \left(1 - \frac{j}{n+1}\right) b_j$$
- $$j = 1, \dots, n$$

Degree elevation: Example

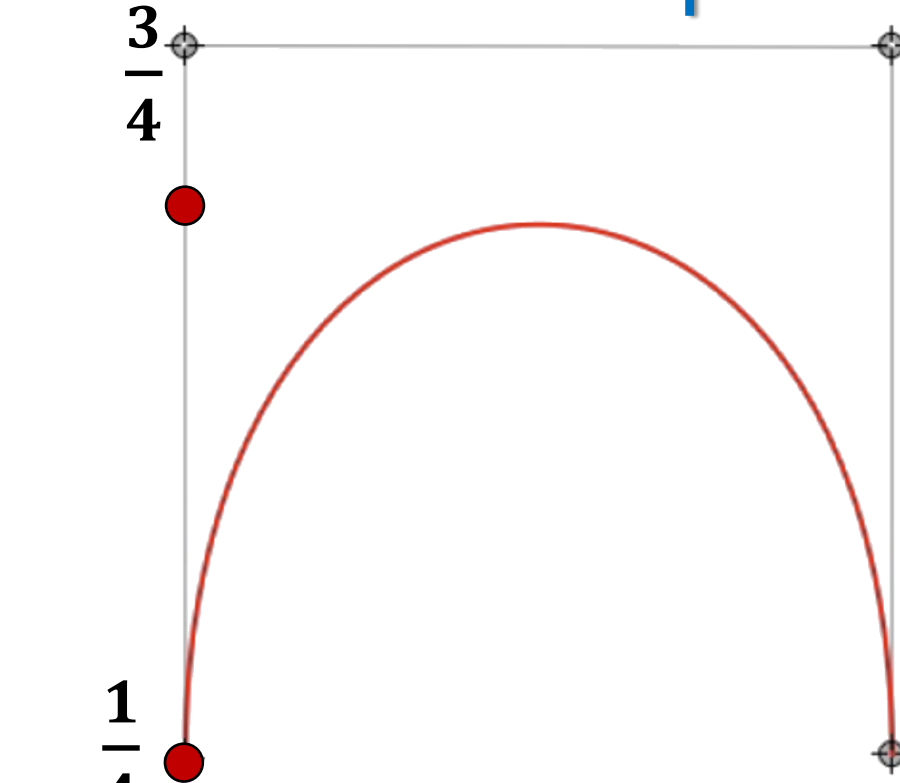


- $\bar{b}_0 = b_0$

- $\bar{b}_{n+1} = b_n$

$$\bar{b}_j = \frac{j}{n+1} b_{j-1} + \left(1 - \frac{j}{n+1}\right) b_j$$
$$j = 1, \dots, n$$

Degree elevation: Example



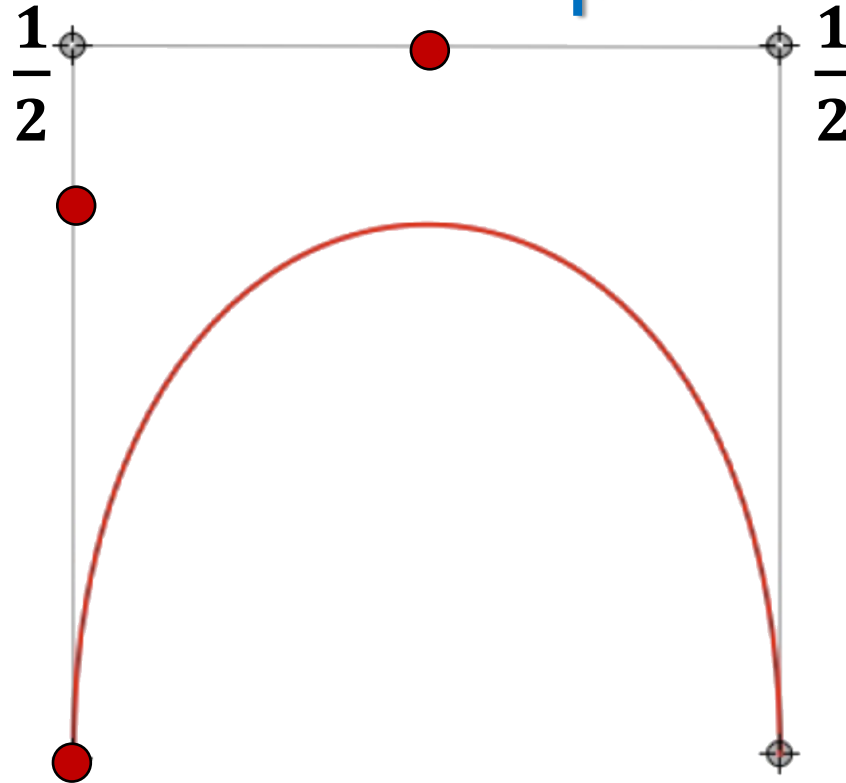
- $\bar{b}_0 = b_0$

$$\bar{b}_j = \frac{j}{n+1} b_{j-1} + \left(1 - \frac{j}{n+1}\right) b_j$$

- $\bar{b}_{n+1} = b_n$

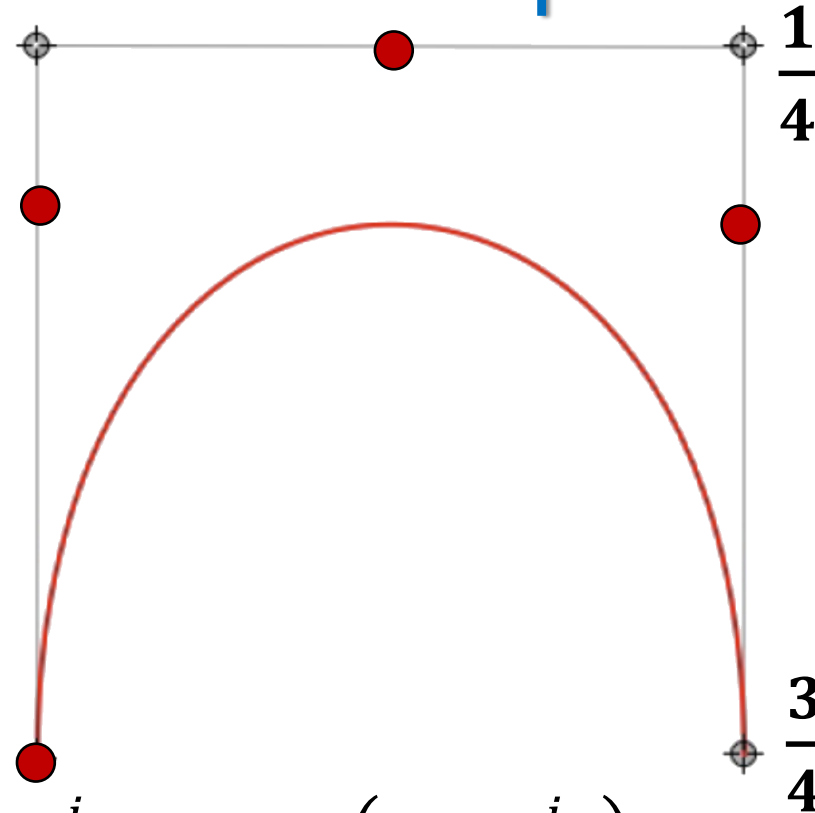
$$j = 1, \dots, n$$

Degree elevation: Example



- $\bar{b}_0 = b_0$
 - $\bar{b}_{n+1} = b_n$
- $$\bar{b}_j = \frac{j}{n+1} b_{j-1} + \left(1 - \frac{j}{n+1}\right) b_j$$
- $$j = 1, \dots, n$$

Degree elevation: Example



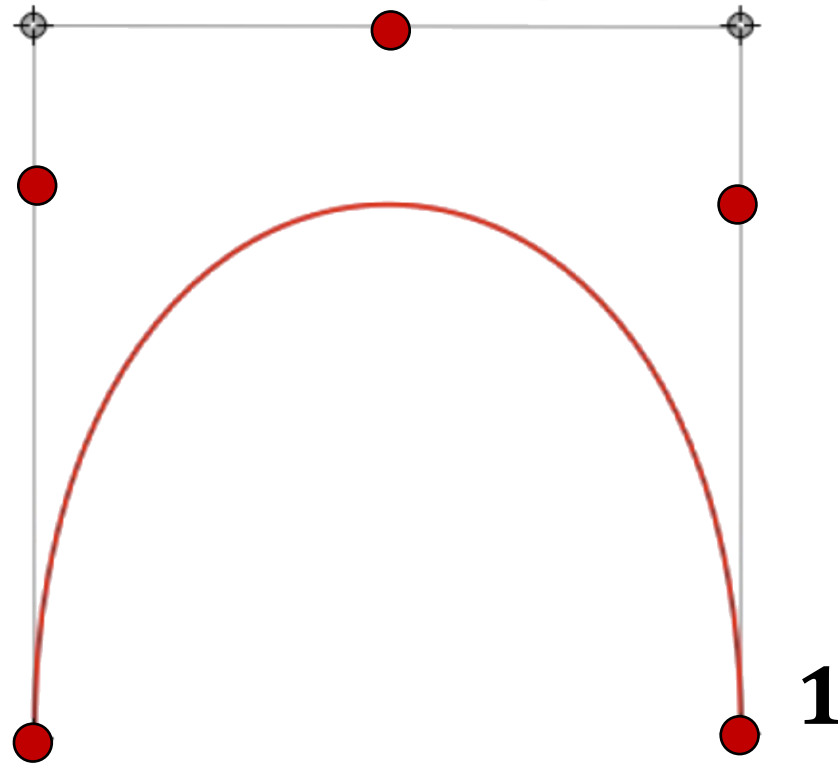
- $\bar{b}_0 = b_0$

- $\bar{b}_{n+1} = b_n$

$$\bar{b}_j = \frac{j}{n+1} b_{j-1} + \left(1 - \frac{j}{n+1}\right) b_j$$

$$j = 1, \dots, n$$

Degree elevation: Example

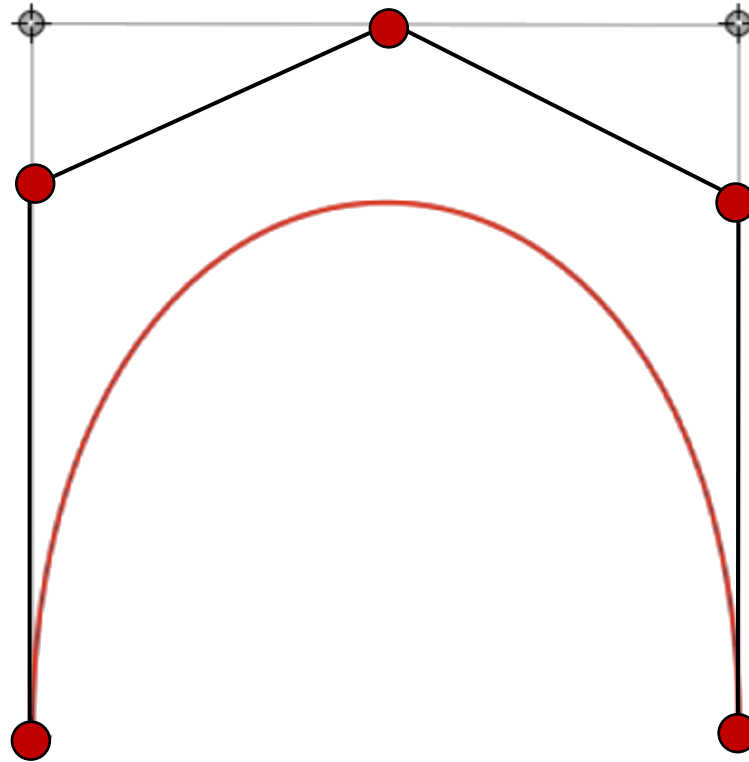


- $\bar{b}_0 = b_0$

- $\bar{b}_{n+1} = b_n$

$$\bar{b}_j = \frac{j}{n+1} b_{j-1} + \left(1 - \frac{j}{n+1}\right) b_j$$
$$j = 1, \dots, n$$

Degree elevation: Example

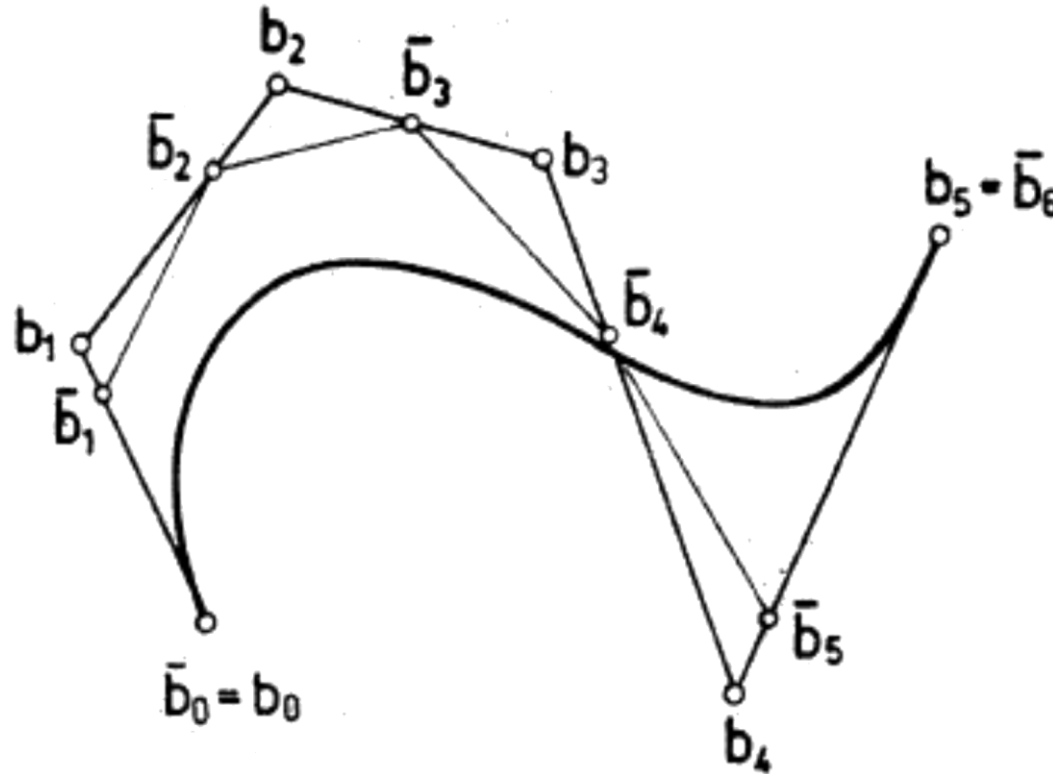


- $\bar{b}_0 = b_0$

- $\bar{b}_{n+1} = b_n$

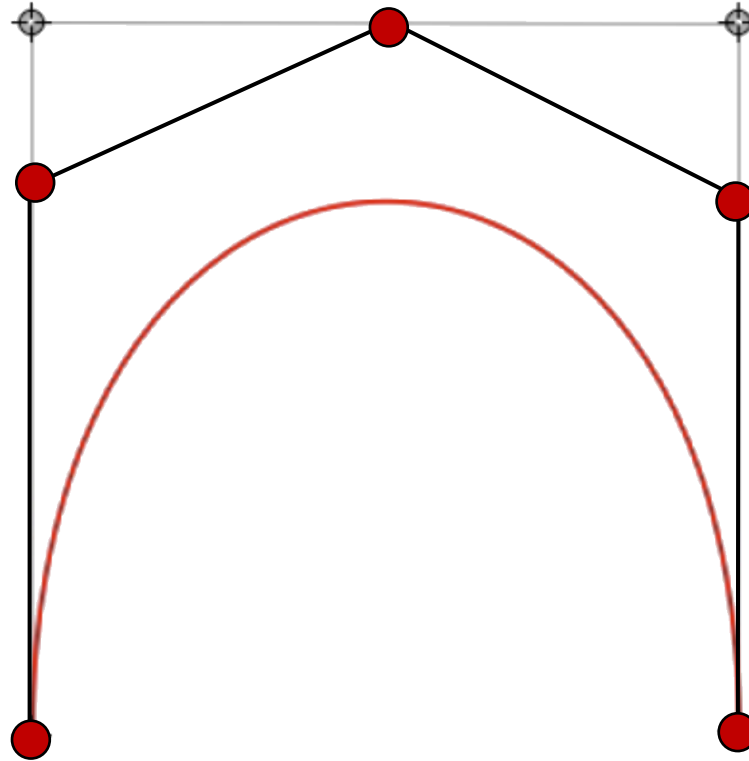
$$\bar{b}_j = \frac{j}{n+1} b_{j-1} + \left(1 - \frac{j}{n+1}\right) b_j$$
$$j = 1, \dots, n$$

Degree elevation



For repeated degree elevation, the Bézier polygon converges to the Bézier curve. (slow convergence)

Degree elevation



- $\bar{b}_0 = b_0$

- $\bar{b}_{n+1} = b_n$

$$\bar{b}_j = \frac{j}{n+1} b_{j-1} + \left(1 - \frac{j}{n+1}\right) b_j$$
$$j = 1, \dots, n$$

Bézier Curves

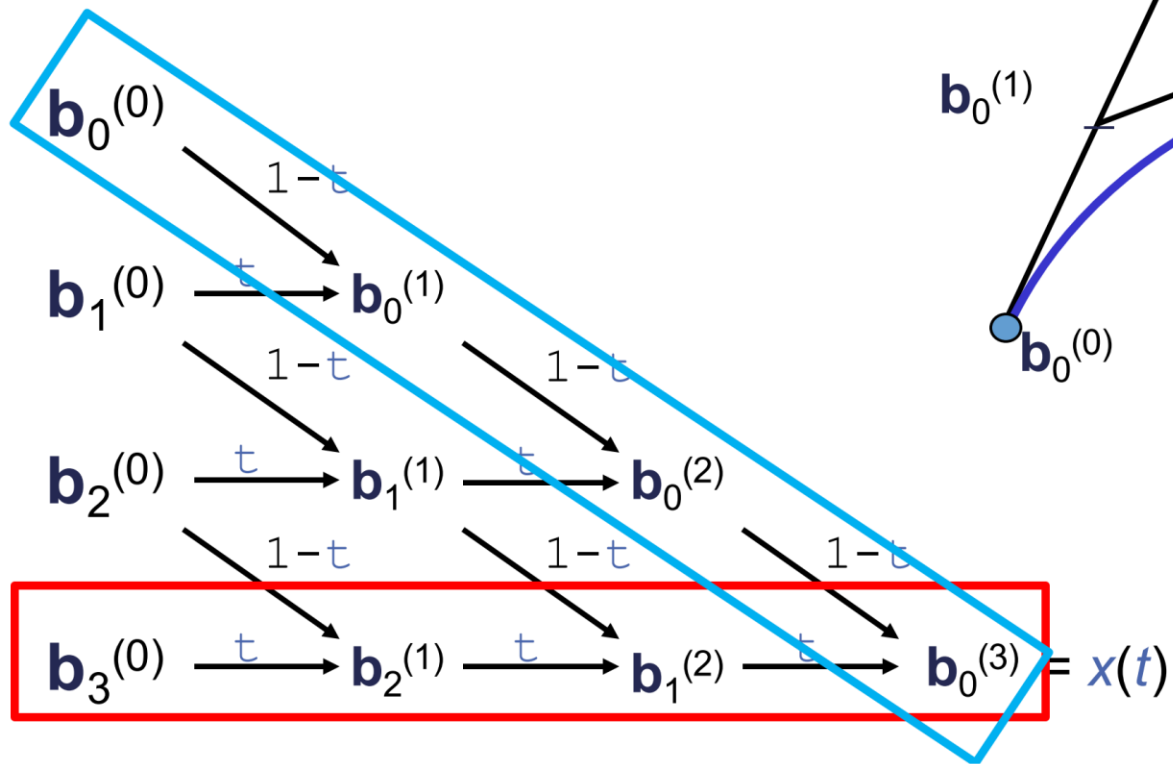
Subdivision

Subdivision

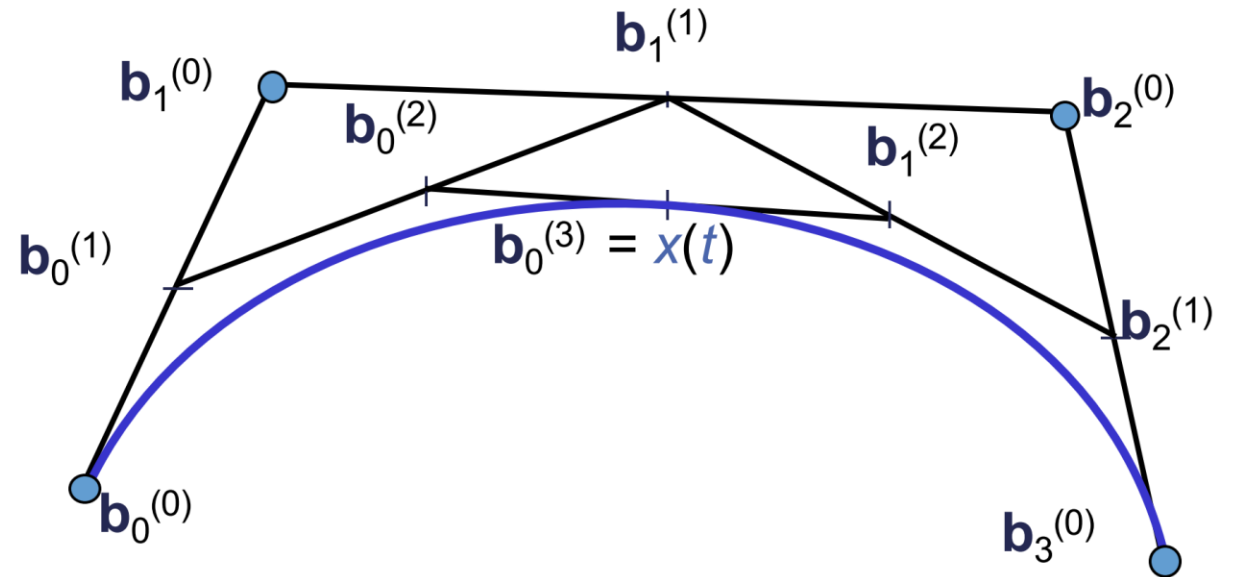
- **Given:** $b_0, \dots, b_n \rightarrow x(t), t \in [0,1]$
- **Wanted:** $b_0^{(1)}, \dots, b_n^{(1)} \rightarrow x^{(1)}(t),$
 $b_0^{(2)}, \dots, b_n^{(2)} \rightarrow x^{(2)}(t),$

with $x = x^{(1)} \cup x^{(2)}$

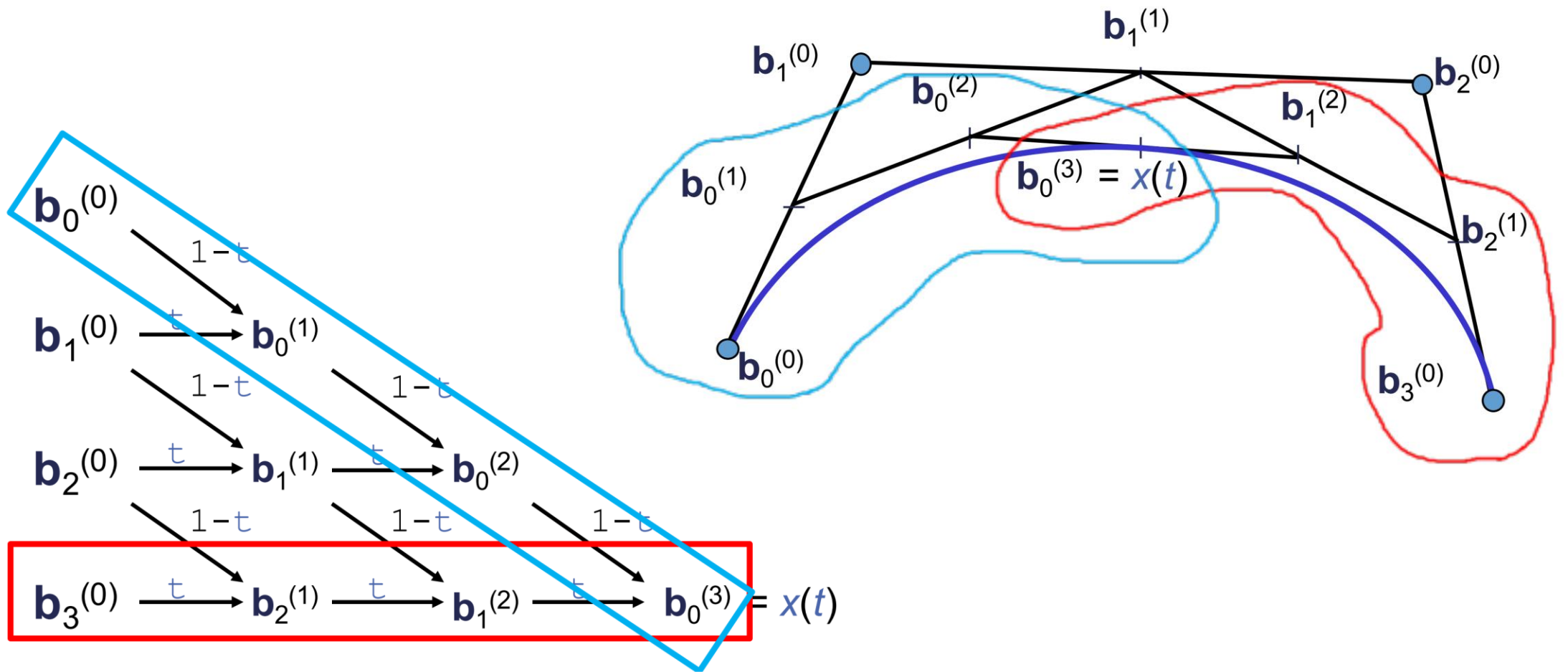
Subdivision: Example



de Casteljau scheme



Subdivision: Example

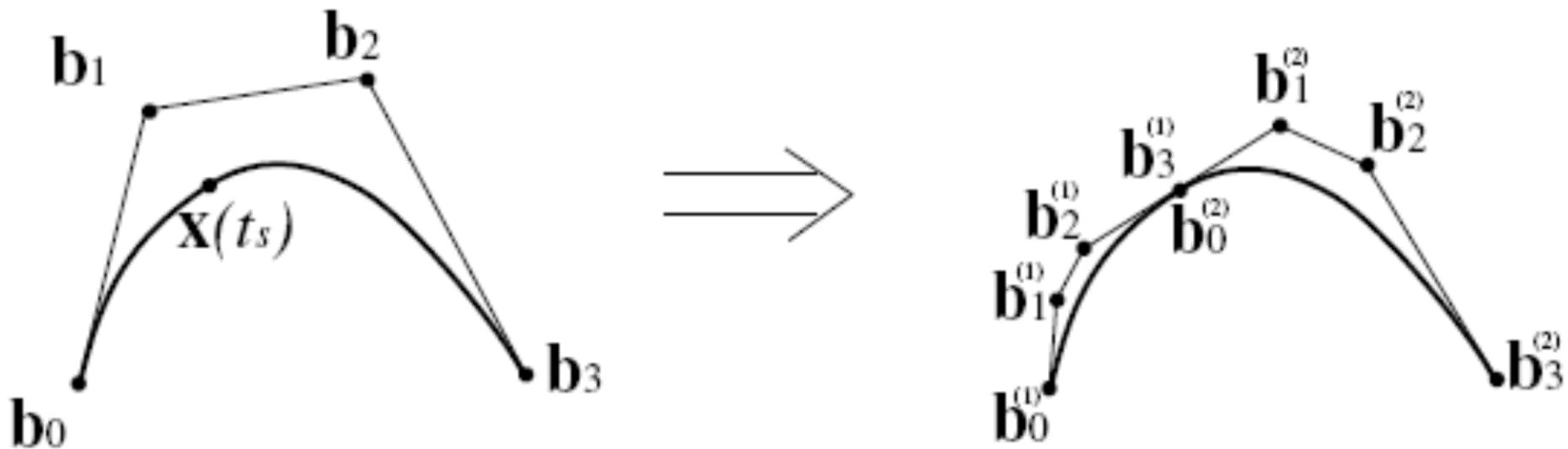


de Casteljau scheme

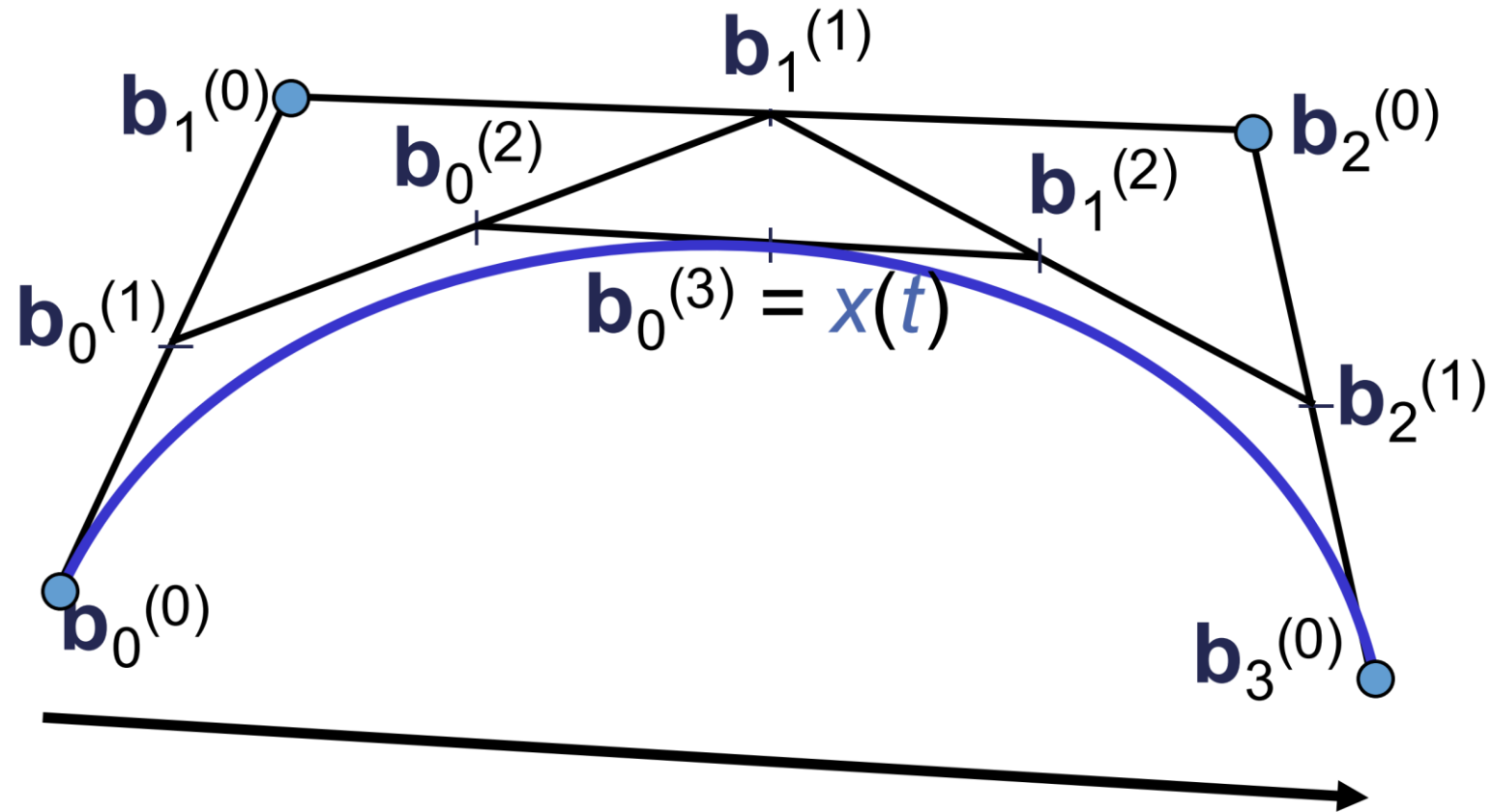
Subdivision

Solution: $b_i^{(1)} = b_0^i$, $b_i^{(2)} = b_i^{n-i}$ for $i = 0, \dots, n$

That means that the new points are intermediate points of the de Casteljau algorithm!

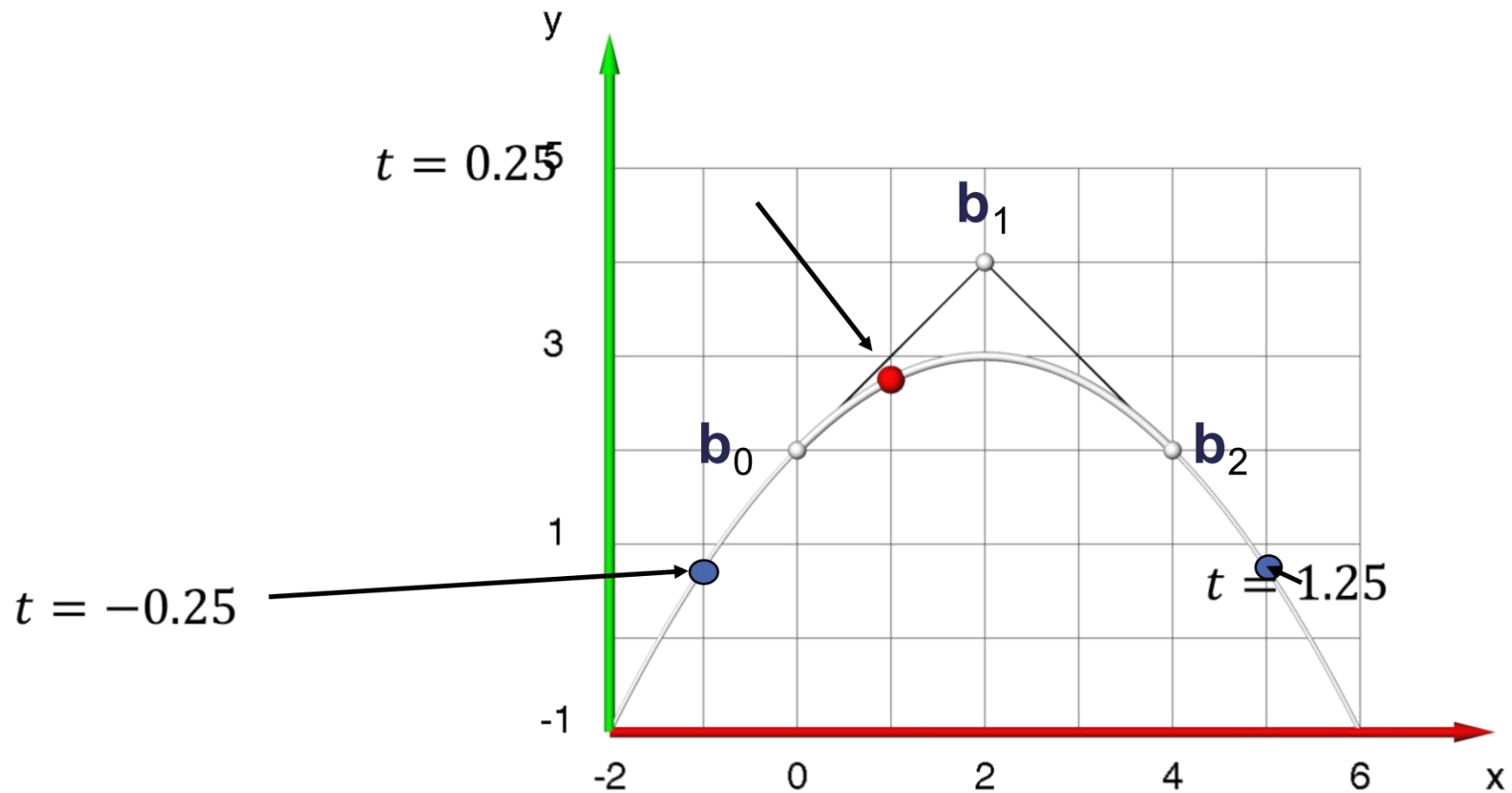


Curve range



parameterization: $t \in [0,1]$

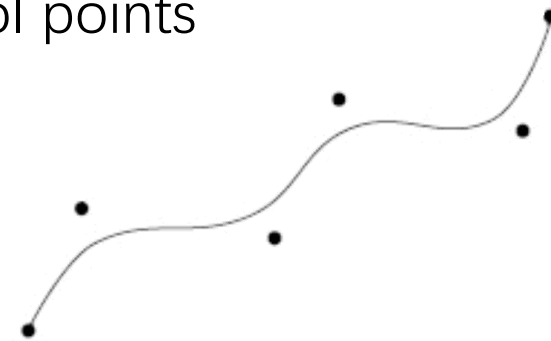
Curve range



Summary & Outlook

- **Bézier curves and curve design**

- The rough form is specified by the position of the control points
- Results: smooth curve approximating the control points
- Computation / Representation:
 - de Casteljau algorithm
 - Bernstein form



- Problems:
 - High polynomial degree
 - Moving a control point can change the whole curve
 - Interpolation of points
 - → **Bézier splines**

Matrix representations
(common in software implementations)

Homogeneous coordinates

$$[P] = \begin{pmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \\ \dots & \dots & \dots & \dots \\ x_n & y_n & z_n & 1 \end{pmatrix}$$

Transformations

- Basic representation $[P^*] = [P][T]$
 - $[P^*]$ is the new coordinates matrix
 - $[P]$ is the original coordinates matrix, or points matrix
 - $[T]$ is the transformation matrix

Transformations

- Translation (2D example)

$$[T_t] = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ x & y & 0 & 1 \end{pmatrix}$$

$$[P^*] = [P][T_t]$$

Transformations

- Basic representation $[P^*] = [P][T]$

$[P^*]$ is the new coordinates matrix

$[P]$ is the original coordinates matrix, or points matrix

$[T]$ is the transformation matrix

$$[P] = \begin{pmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \\ \dots & \dots & \dots \\ x_n & y_n & 1 \end{pmatrix}$$

Transformations

- Uniform scaling

$$[T] = \begin{pmatrix} s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \\ 0 & 0 & s & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Non-uniform scaling

$$[T] = \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Transformations

- Rotation 2D

$$\left. \begin{aligned} x &= r \cos \alpha \\ y &= r \sin \alpha \end{aligned} \right\} \text{Original coordinates of point } P$$

$$\left. \begin{aligned} x^* &= r \cos(\alpha + \theta) \\ y^* &= r \sin(\alpha + \theta) \end{aligned} \right\} \text{The new coordinates}$$

$$\begin{bmatrix} x^* & y^* & 0 & 1 \end{bmatrix} = \begin{bmatrix} x & y & 0 & 1 \end{bmatrix} \begin{pmatrix} \cos \theta & \sin \theta & 0 & 0 \\ -\sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Transformations

- Rotation about an arbitrary axis (2D)
 - Translate the fixed axis so it coincides with z-axis
→ apply to object
 - Rotate object about the axis
 - Translate object back

$$[P^*] = [P][T_t][T_r][T_{-t}]$$

Transformations

- Rotation about an arbitrary axis (2D)

Step 1: Translate the fixed axis so it coincides with z-axis

Step 2: Rotate object about the axis

Step 3: Translate the fixed axis back to the original position

$$[P^*] = [P][T_t][T_r][T_{-t}]$$

Transformations

- Scaling with an arbitrary point (x, y)

$$\begin{aligned}[P^*] &= [P][T_t][T_s][T_{-t}] \\ &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -x & -y & 0 & 1 \end{pmatrix} \begin{pmatrix} s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \\ 0 & 0 & s & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ x & y & 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \\ 0 & 0 & s & 0 \\ x - sx & y - sy & 0 & 1 \end{pmatrix}\end{aligned}$$

Transformations

- Rotation about an arbitrary point (x, y)

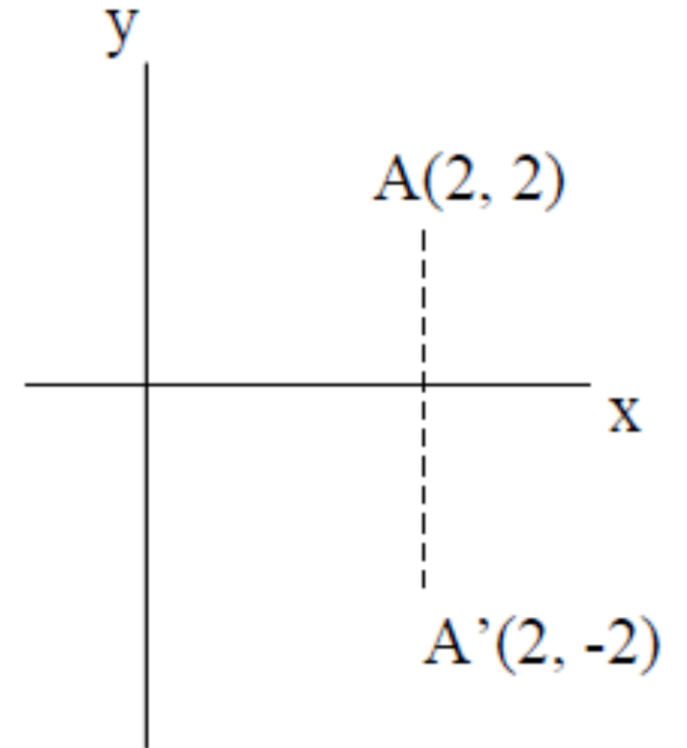
$$[T_{\text{cond}}] = [T_t][T_s][T_{-t}]$$

$$= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -x & -y & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos \theta & \sin \theta & 0 & 0 \\ -\sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ x & y & 0 & 1 \end{pmatrix}$$

Transformations

- Mirroring about x-axis (negative scaling along y-axis)

$$[P^*] = [2 \quad 2 \quad 0 \quad 1] \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$
$$= [2 \quad -2 \quad 0 \quad 1]$$



Transformations

- Mirroring about arbitrary axis
 - Translate line to pass through origin
 - Rotate axis to coincide with x -axis
 - Mirror about x -axis
 - Rotate back
 - Translate back to original position

$$[P^*] = [P][T_t][T_r][T_m][T_{-r}][T_{-t}]$$

Transformations

- Rotation about coordinates axes (3D)

$$[T_{rz}] = \begin{pmatrix} \cos \theta & \sin \theta & 0 & 0 \\ -\sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$[T_{rx}] = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta & \sin \theta & 0 \\ 0 & -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$[T_{ry}] = \begin{pmatrix} \cos \theta & 0 & -\sin \theta & 0 \\ 0 & 1 & 0 & 0 \\ \sin \theta & 0 & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Transformations

- Rotation θ about an arbitrary axis (3D)
 1. Translate the given line so that it will pass through the origin
 2. Rotate about the x -axis so that the line lies in the xz -plane (angle α)
 3. Rotate about the y -axis so that the line coincides with the z -axis (angle ϕ)
 4. Rotate the geometric object about the z -axis (angle θ – given rotation angle)
 5. Reverse of step 3
 6. Reverse of step 2
 7. Reverse of step 1

$$[P^*] = [P][T_t][T_r]_{\alpha}[T_r]_{\phi}[T_r]_{\theta}[T_r]_{-\phi}[T_r]_{-\alpha}[T_{-t}]$$

Alternatively, you can use Quaternions!

Bézier Curves

- Cubic Bézier curves $f(t) = P_0 B_0^{(3)} + P_1 B_1^{(3)} + P_2 B_2^{(3)} + P_3 B_3^{(3)}$

$$B_0^{(3)}(t) = \frac{3!}{0! 3!} t^0 (1 - t)^3 = (1 - t)^3$$

$$B_1^{(3)} = \frac{3!}{1! 2!} t^1 (1 - t)^2 = 3t(1 - t)^2$$

$$B_2^{(3)} = \frac{3!}{2! 1!} t^2 (1 - t)^1 = 3t^2(1 - t)$$

$$B_3^{(3)} = \frac{3!}{3! 0!} t^3 (1 - t)^0 = t^3$$

Bézier Curves

- Cubic Bézier curves

$$f(t) = P_0 B_0^{(3)} + P_1 B_1^{(3)} + P_2 B_2^{(3)} + P_3 B_3^{(3)}$$

- In Matrix form:

- -The curve

$$f(t) = [t^3 \quad t^2 \quad t \quad 1] \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix}$$

- -The tangent

$$f'(t) = [3t^2 \quad 2t \quad 1 \quad 0] \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix}$$

B-spline Curves (to be covered later)

- Uniform cubic B-Spline curve

$$f_i(t) = \frac{1}{6} [t^3 \quad t^2 \quad t \quad 1] \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{pmatrix} \begin{bmatrix} P_{i-1} \\ P_i \\ P_{i+1} \\ P_{i+2} \end{bmatrix}$$

B-spline Curves (to be covered later)

- Other splines:

- Catmull-Rom

$$f_i(t) = [t^3 \quad t^2 \quad t \quad 1] \frac{1}{2} \begin{pmatrix} -1 & 3 & -3 & 1 \\ 2 & -5 & 4 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{bmatrix} P_{i-1} \\ P_i \\ P_{i+1} \\ P_{i+2} \end{bmatrix}$$

- Cardinal splines

$$\begin{pmatrix} -a & 2-a & a-2 & a \\ 2a & a-3 & 3-2a & -a \\ -a & 0 & a & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

- Tensioned splines

$$\frac{1}{6} \begin{pmatrix} -a & 12-9a & 9a-12 & a \\ 2a & a-3 & 18-15a & -a \\ -3a & 0 & 3a & 0 \\ 0 & 6-2a & a & 0 \end{pmatrix}$$