Computer Graphics

Freeform Curves 2

Mark Pauly

Geometric Computing Laboratory

Bezier Curves

- Bezier curves use Bernstein polynomials as basis:

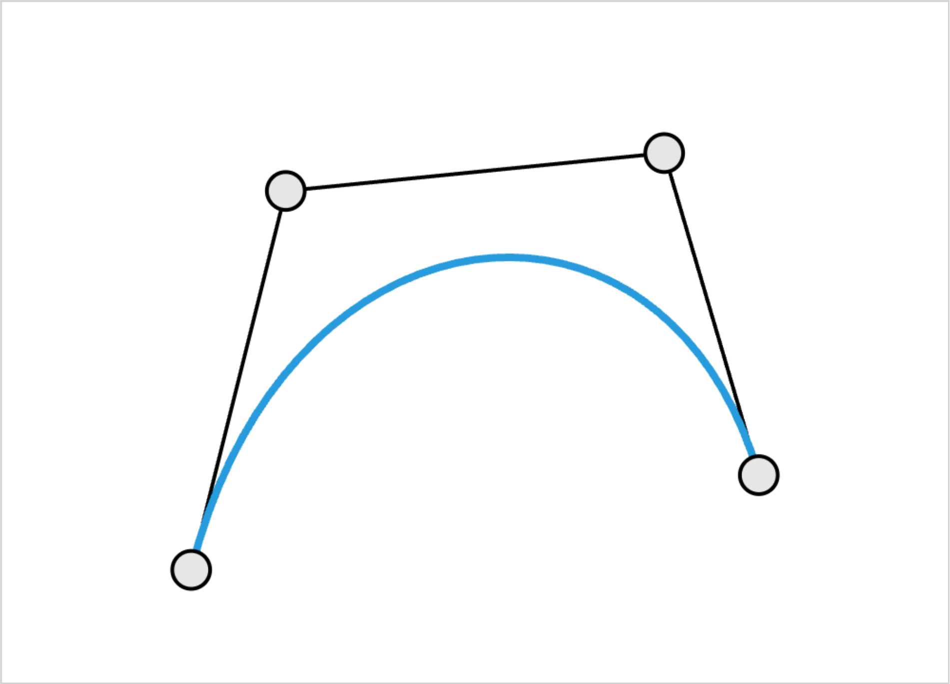
$$\mathbf{x}(t) = \sum_{i=0}^{n} \mathbf{b}_i \, B_i^n(t)$$

  - Coefficients $\mathbf{b}_i$ are called *control points*
  - Points $(\mathbf{b}_0, \ldots, \mathbf{b}_n)$ define the *control polygon*.

Pierre Bézier,
1910–1999

Properties of Bezier Curves

- **Affine combination:** Point $\mathbf{x}(t)$ is an affine combination of control points. Control points have geometric meaning!
- **Convex hull:** Curve $\mathbf{x}(t)$ lies in convex hull of control points
- **Endpoint interpolation:** Curve $\mathbf{x}(t)$ starts at $\mathbf{b}_0$ and ends at $\mathbf{b}_n$.
- **Symmetry:** Curve defined by $(\mathbf{b}_n, \mathbf{b}_{n-1}, \ldots, \mathbf{b}_0)$ is the same as the one defined by $(\mathbf{b}_0, \mathbf{b}_1, \ldots, \mathbf{b}_n)$, just in reverse order.
- **Pseudo-local control:** Control point $\mathbf{b}_i$ has its maximum effect on the curve at $t = i/n$.



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

Bernstein & Bezier

- **Bernstein polynomials**

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

  - Partition of Unity
  - Non-Negativity
  - Symmetry
  - Endpoint values
  - Local maximum

- **Bezier curves**

$$\mathbf{x}(t) \;=\; \sum_{i=0}^{n} \mathbf{b}_i \, B_i^n(t)$$

  - Affine combination
  - Convex hull
  - Symmetry
  - Endpoint interpolation
  - Pseudo-local control

Bezier Curves

- Check requirements for geometry representation:
  - ✓ Approximation power
  - ✗ Efficient evaluation of positions & derivatives ???
  - ✓ Ease of manipulation
  - ✗ Ease of implementation ???
- How to efficiently implement binomial coefficient and factorial?

$$B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i} \quad \text{and} \quad \binom{n}{i} = \begin{cases} \frac{n!}{i!(n-i)!} & \text{if } 0 \leq i \leq n, \\ 0 & \text{otherwise.} \end{cases}$$

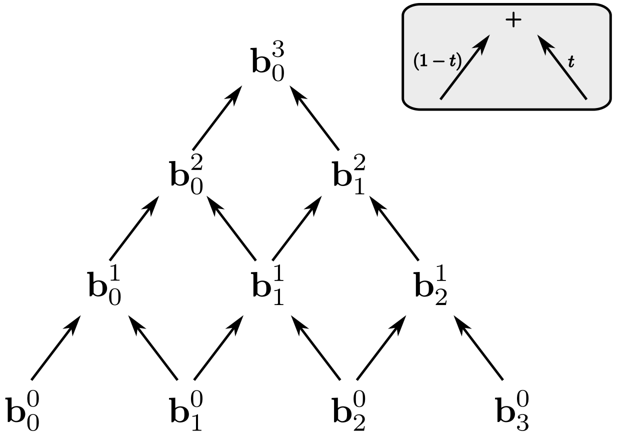
de Casteljau Algorithm

- Given:
  - Control polygon $\mathbf{b}_0, \ldots, \mathbf{b}_n$
  - Curve parameter t
- Initialization: $\mathbf{b}_i^0 = \mathbf{b}_i \quad i = 0, \ldots, n$
- Recursion:

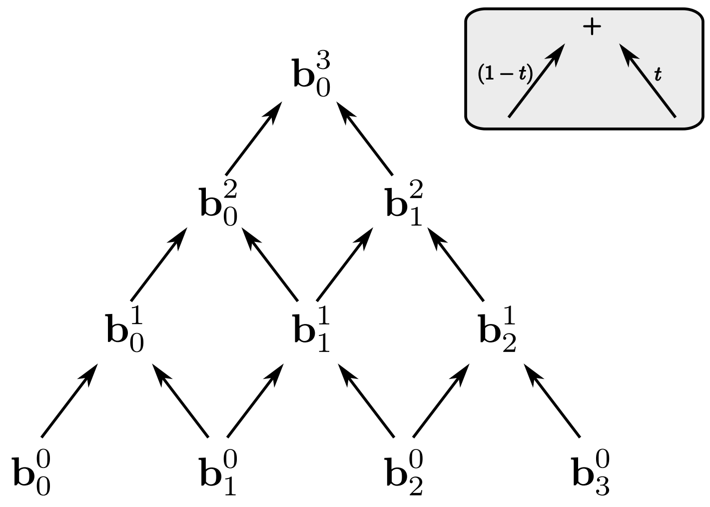
$$\mathbf{b}_i^k = (1-t)\,\mathbf{b}_i^{k-1}(t) + t\,\mathbf{b}_{i+1}^{k-1}(t)$$

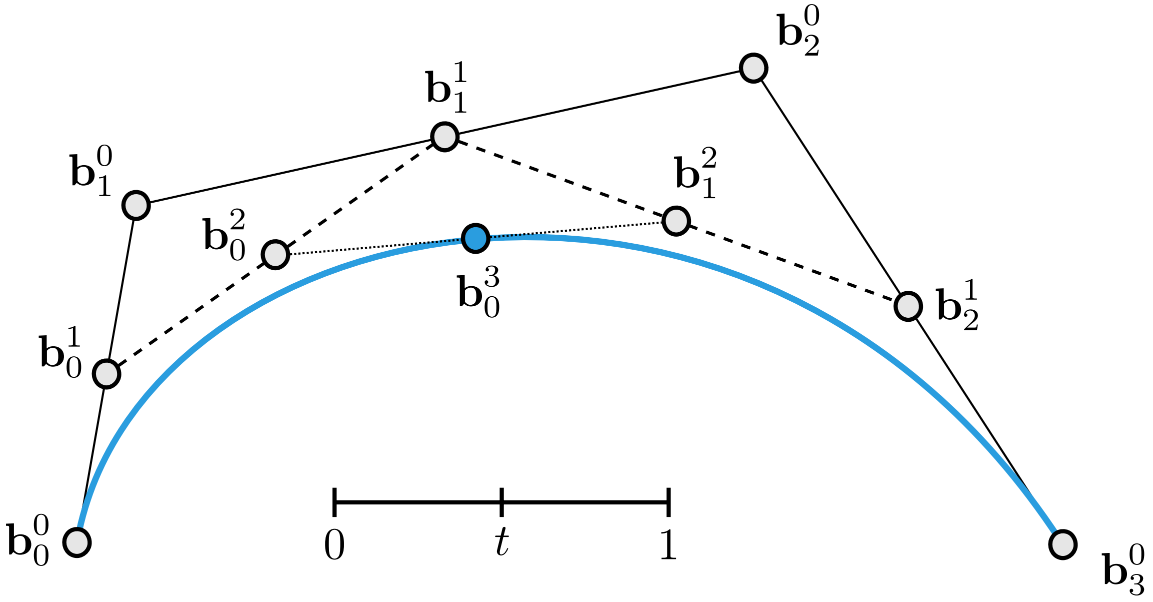
  where $k = 1, \ldots, n$ and $i = 0, \ldots, n-k$
- Result: $\mathbf{b}_0^n = \sum_{i=0}^n \mathbf{b}_i B_i^n(t) = \mathbf{x}(t)$

Paul de Casteljau, *1930–2022*

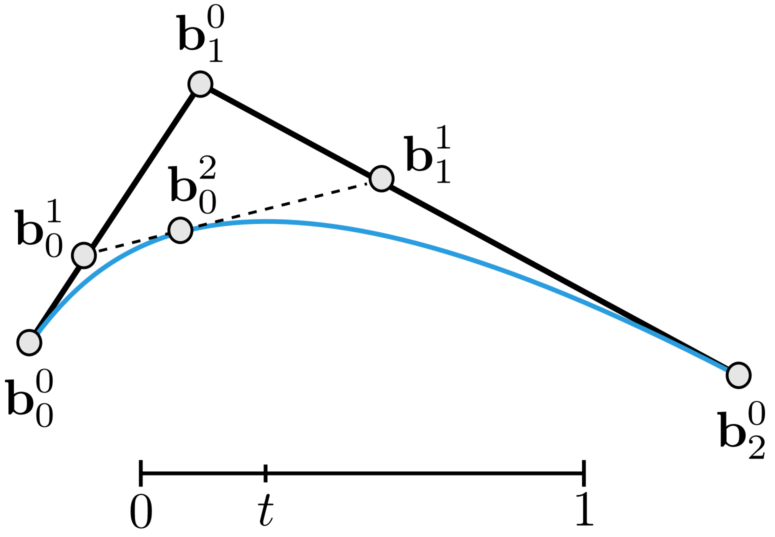


Geometric Interpretation





de Casteljau and Bernstein Basis



Bernstein Recursion

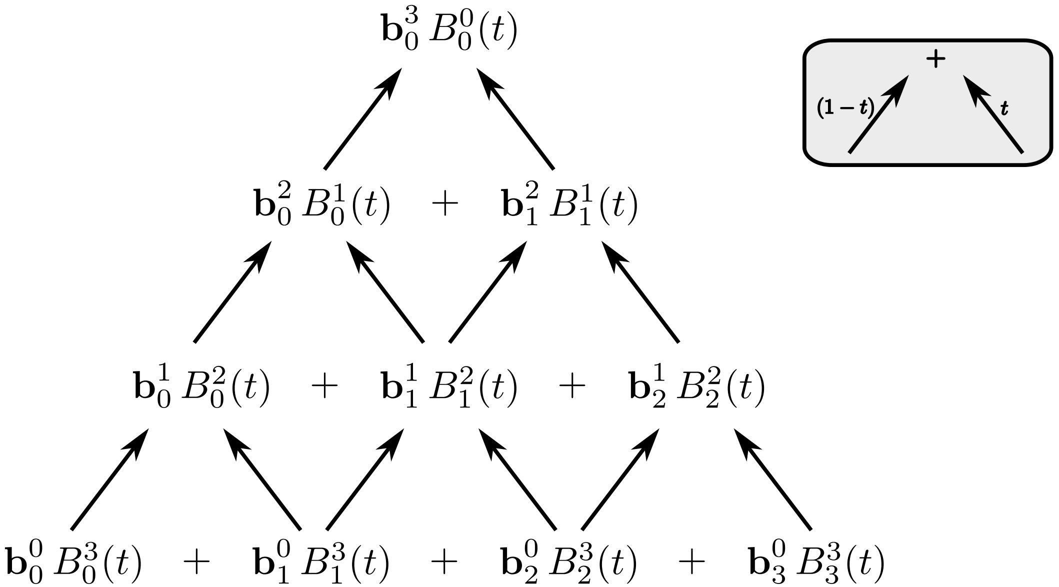
- Bernstein polynomials can be evaluated through a recursion:

$$B_i^n(t) = (1-t)\,B_i^{n-1}(t) \; + \; t\,B_{i-1}^{n-1}(t)$$

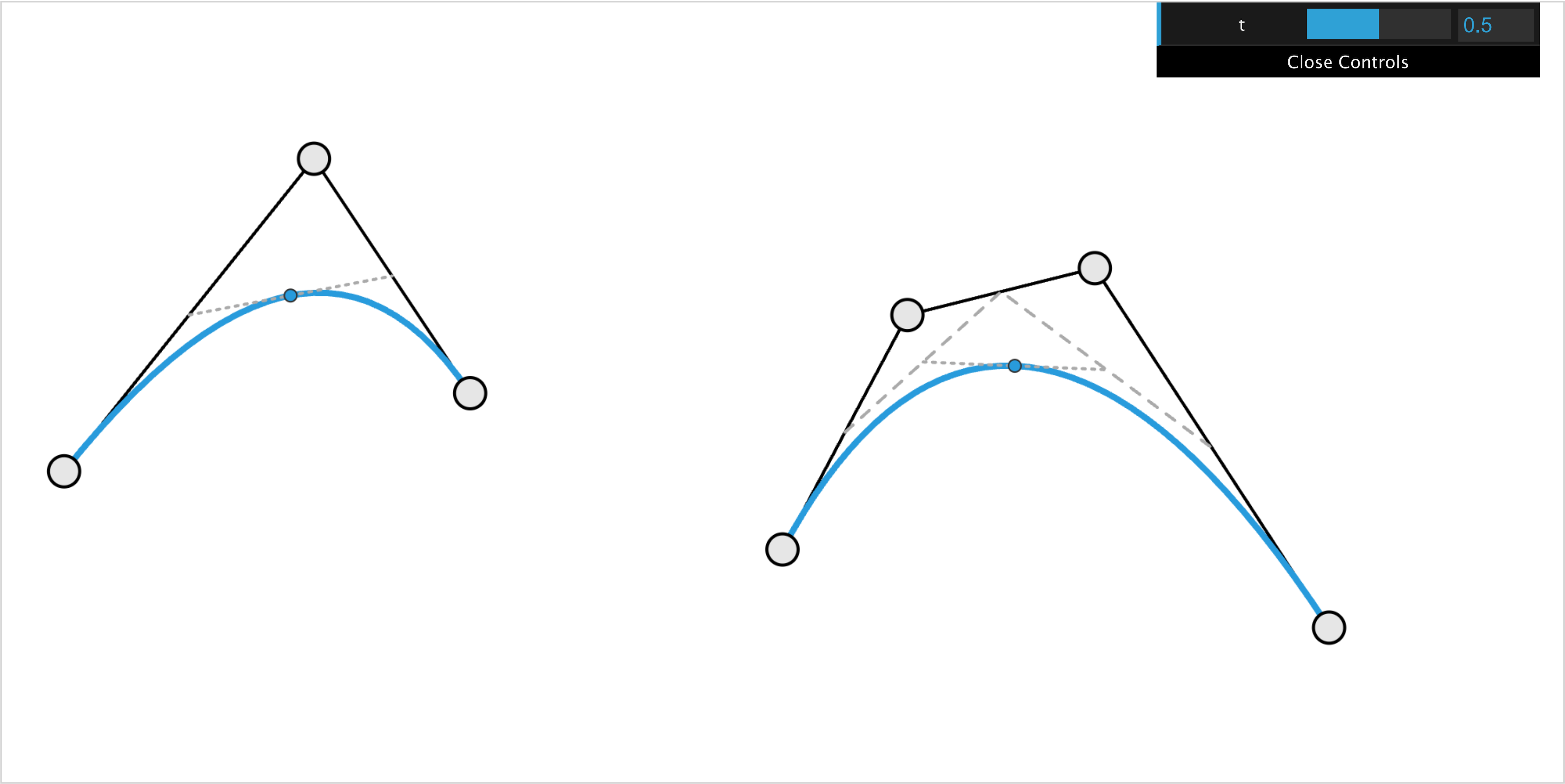
with $B_0^0(t) = 1$ and $B_i^n(t) = 0$ for $i \notin \{0, \ldots, n\}$

de Casteljau Construction

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



Try it yourself!



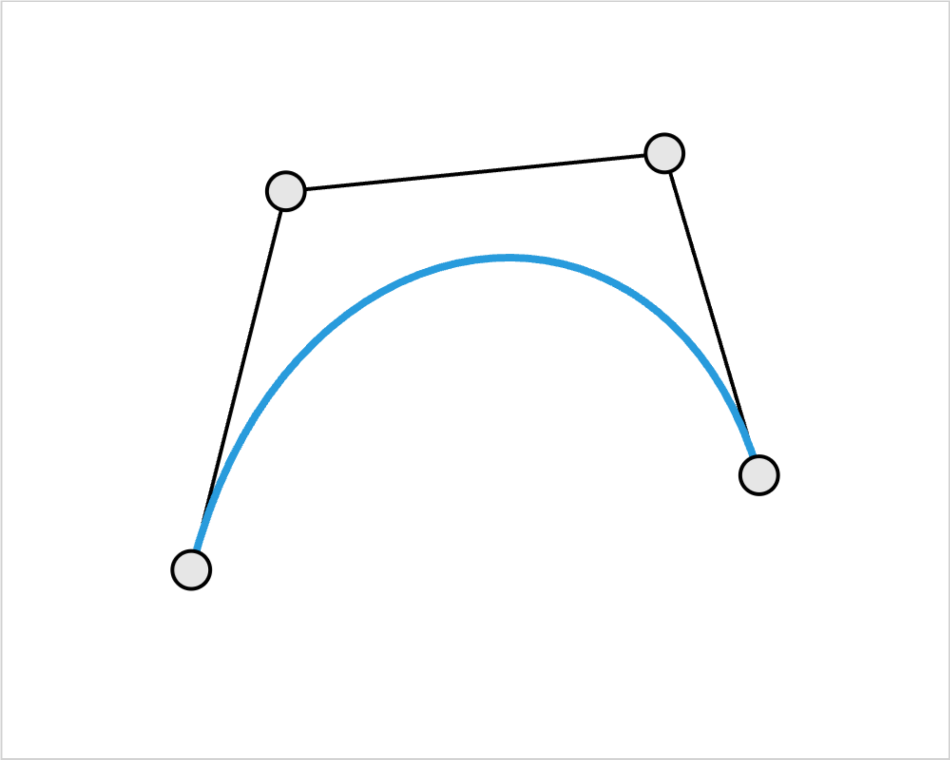
Quiz: Bezier Curves

What is the polynomial degree of a Bezier curve with four control points?

A: Degree 3

B: Degree 4

C: Degree 5



Quiz: Bezier Curves

True or False?

Bezier curves interpolate all their control points.

A: True

B: False

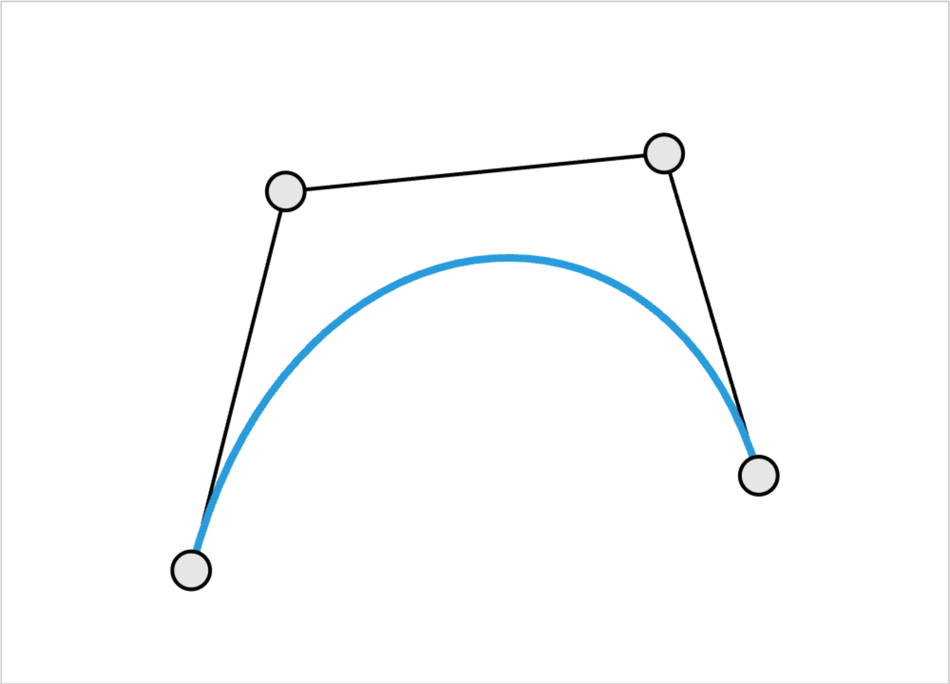
Quiz: Bezier Curves

True or False?

If two control points of a Bezier curve coincide, then the curve must have a sharp corner.

A: True

B: False



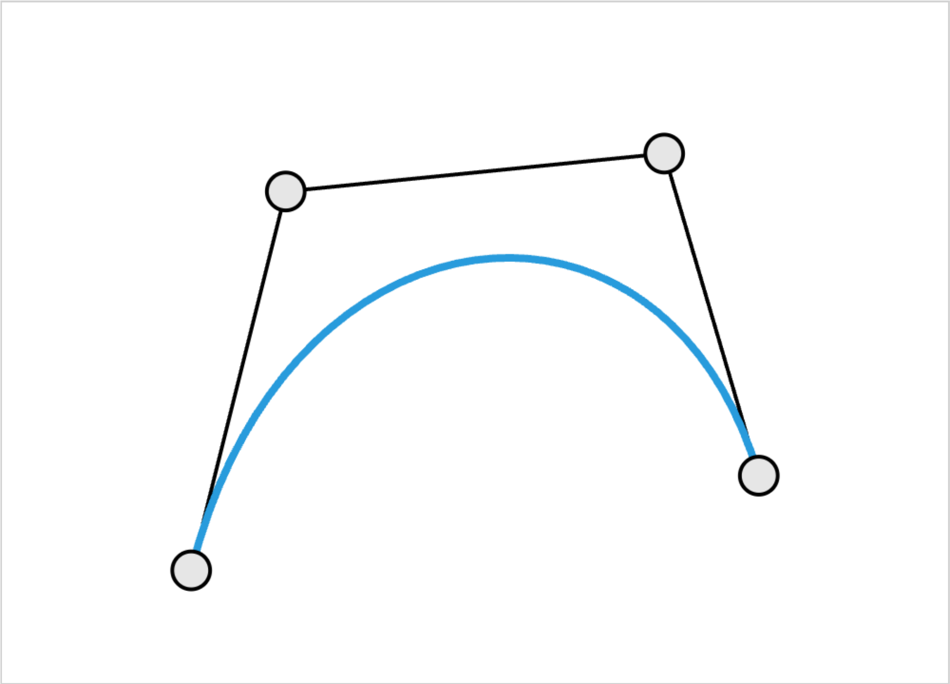
Quiz: Bezier Curves

True or False?

If a Bezier curve has no self-intersections, then its control polygon has no self-intersections.

A: True

B: False



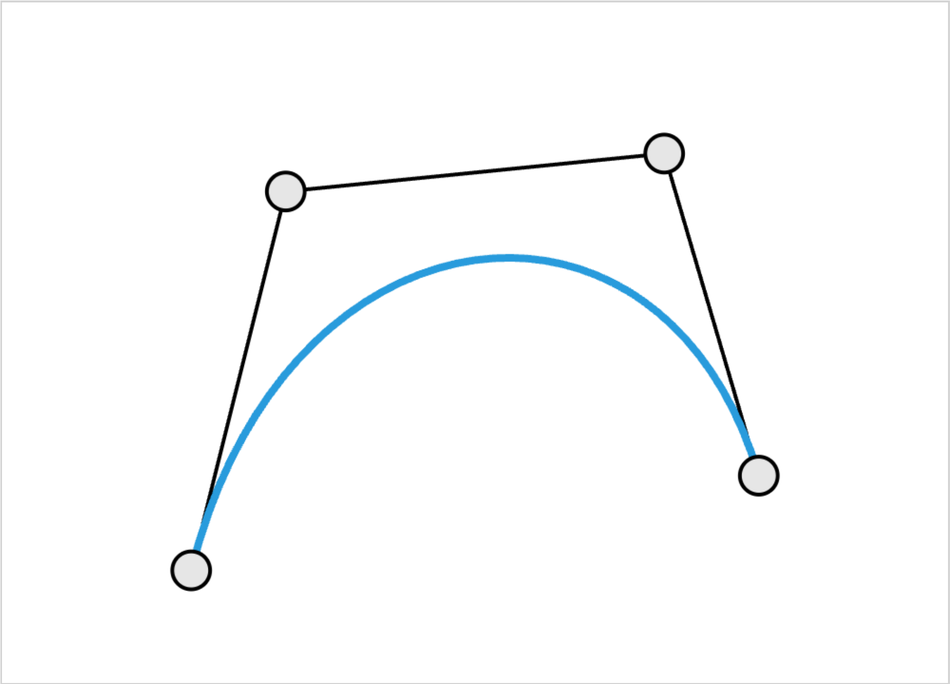
Quiz: Bezier Curves

True or False?

If all control points of a Bezier curve lie on a line, then the Bezier curve is injective.

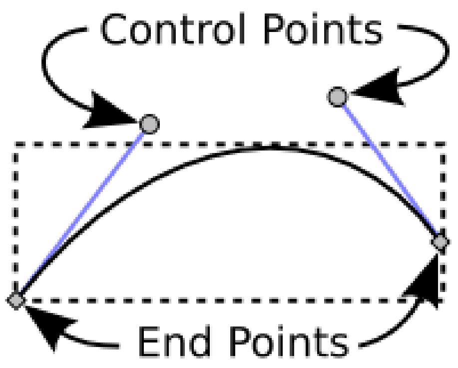
A: True

B: False

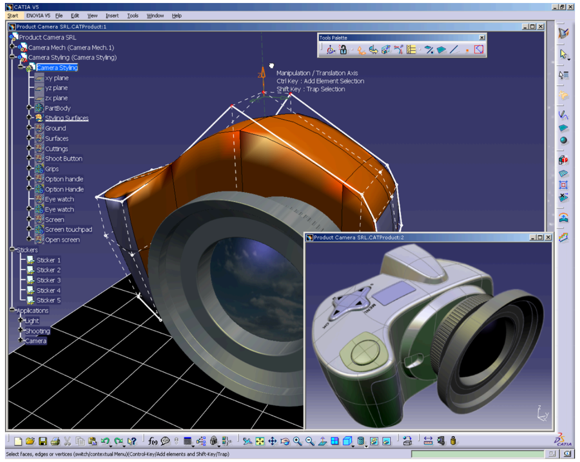


Applications of Bezier Curves

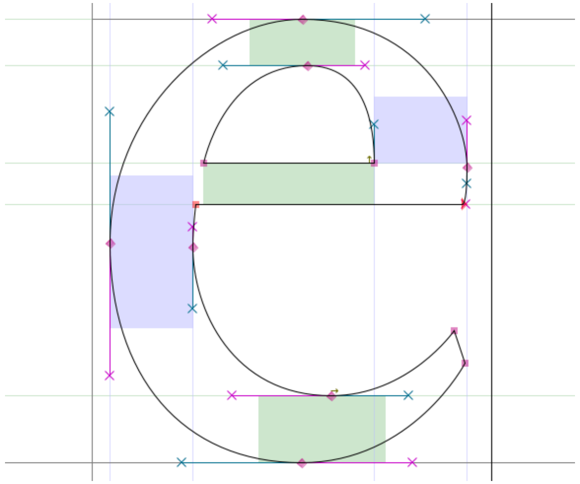
- Bezier curves are the most prominent curve representation:
  - Used for 2D vector drawing applications
  - Used for 3D computer-aided design (CAD)
  - Used for true-type fonts



Inkscape

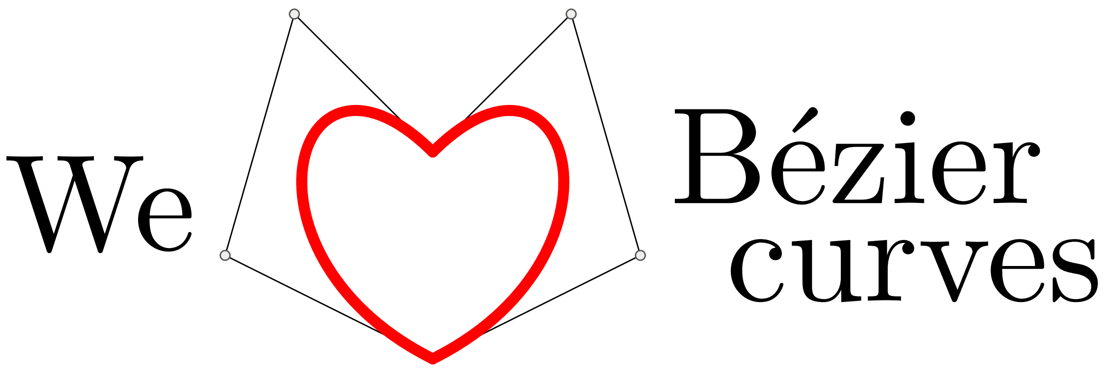


Dassault CATIA



FontForge

Applications of Bezier Curves



Summary

- How to represent polynomial curves?
- Monomial basis does not sum to one
- Bernstein basis does sum to one
- Bezier curves can intuitively be manipulated through control points
- de Casteljau algorithm allows for efficient evaluation

Literature

- Farin: *Curves and Surfaces for CAGD. A Practical Guide*, Morgan Kaufmann, 2001
    - Chapters 4 & 5