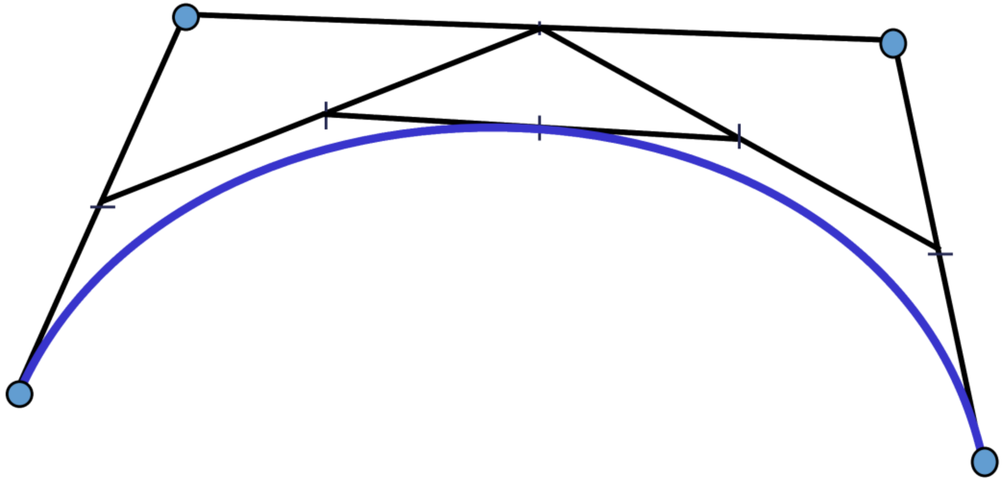
计算机辅助几何设计
2023秋学期

Bézier Splines

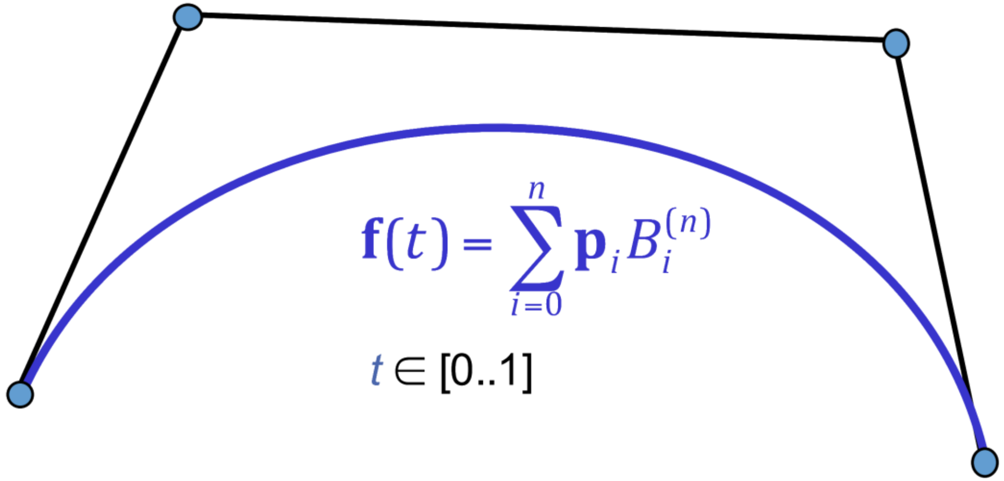
陈仁杰

中国科学技术大学

Recap



de Casteljau algorithm



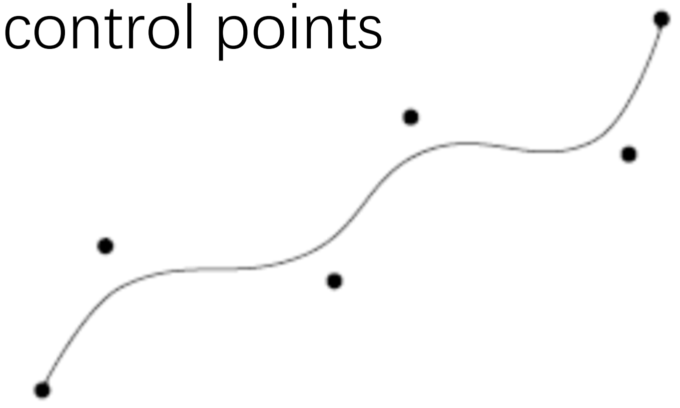
$$\mathbf{f}(t) = \sum_{i=0}^{n} \mathbf{p}_i B_i^{(n)}$$

$t \in [0..1]$

Bernstein form

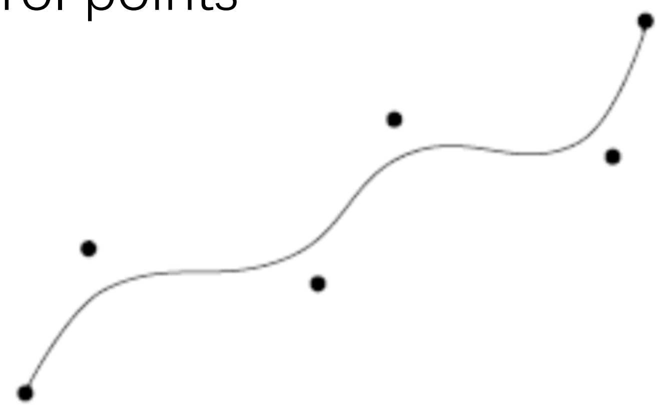
Recap

- **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



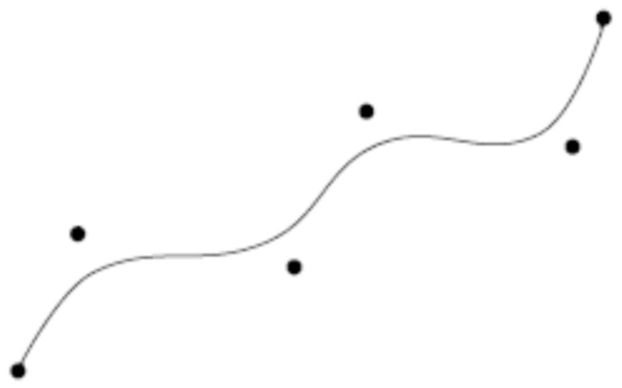
Recap

- **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

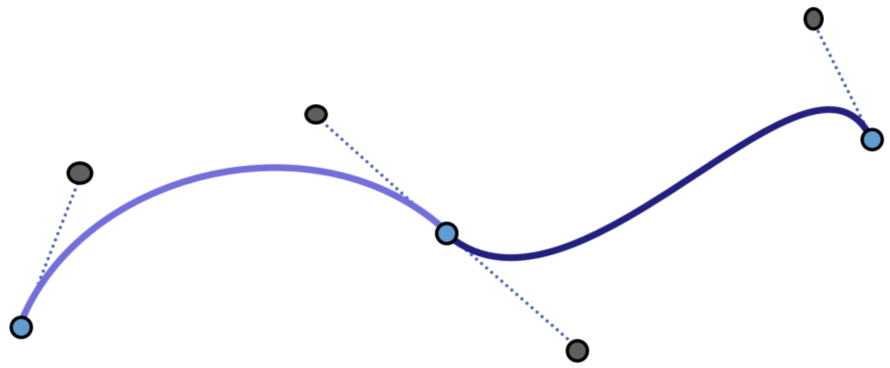


  - Problems:
    - High polynomial degree
    - Moving a control point can change the whole curve
    - Interpolation of points
    - →**Bézier splines**

Recap



Approximation ⟶ Interpolation



Towards Bézier Splines

- **Interpolation problems:**
  - given:

    $\boldsymbol{k}_0, \ldots, \boldsymbol{k}_n \in \mathbb{R}^3$ control points

    $t_0, \ldots, t_n \in \mathbb{R}$ knot sequence

    $t_i < t_{i+1}$, for $i = 0, \ldots, n-1$
  - wanted
    - Interpolating curve $\boldsymbol{x}(i)$, i.e. $\boldsymbol{x}(t_i) = \boldsymbol{k}_i$ for $i = 0, \ldots, n$
  - Approach: “Joining” of n Bézier curves with certain intersection conditions

Towards Bézier Splines

- **The following issues arise when stitching together Bézier curves:**
  - Continuity
  - Parameterization
  - Degree

Bézier Splines

Parametric and Geometric Continuity

Parametric Continuity

Joining curves – continuity

- Given: 2 curves

  $\boldsymbol{x}_1(t)$ over $[t_0, t_1]$

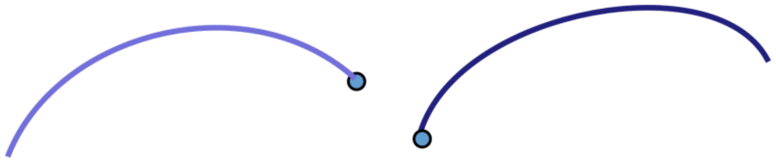
  $\boldsymbol{x}_2(t)$ over $[t_1, t_2]$

- $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are C^r continuous at t_1, if all their 0^{th} to r^{th} derivative vectors coincides at t_1

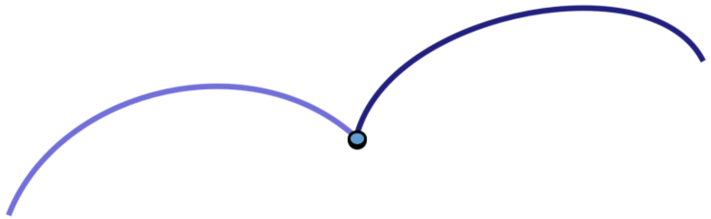
Parametric Continuity

- C^0: **position varies continuously**
- C^1: **First derivative is continuous across junction**
  - In other words: the velocity vector remains the same
- C^2: **Second derivative is continuous across junction**
  - The acceleration vector remains the same

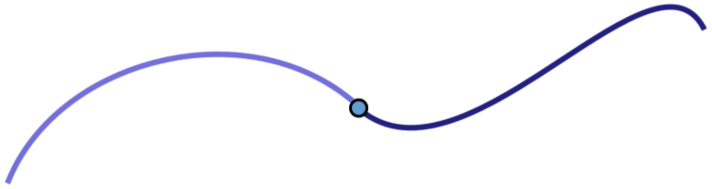
Parametric Continuity



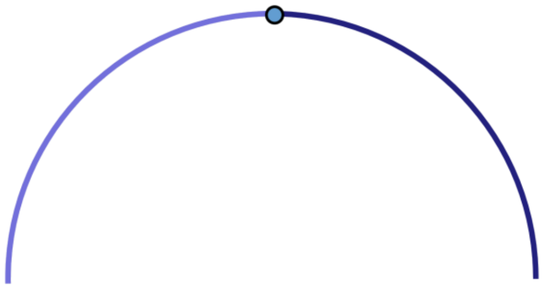
C^{-1} continuity



C^{0} continuity



C^{1} continuity



C^{2} continuity

Continuity

Parametric Continuity C^r:

- C^0, C^1, C^2 ⋯ continuity
- Does a particle moving on this curve have a smooth trajectory (position, velocity, acceleration, ⋯)?
- Depends on parameterization
- Useful for animation (object movement, camera paths)

Geometric Continuity G^r:

- Is the curve itself smooth?
- Independent of parameterization
- More relevant for modeling (curve design)

Geometric continuity:

Geometric continuity of curves

- Given: 2 curves

  $\boldsymbol{x}_1(t)$ over $[t_0, t_1]$

  $\boldsymbol{x}_2(t)$ over $[t_1, t_2]$

- $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are G^r continuous in t_1, if they can be reparameterized in such a way that they are C^r continuous in t_1

Geometric continuity:

- $G^0 = C^0$**: position varies continuously (connected)**

- G^1**: tangent direction varies continuously (same tangent)**
  - In other words: the **normalized** tangent varies continuously
  - Equivalently: The curve can be reparameterzed so that it becomes C^1
  - Also equivalent: A unit speed parameterization would be C^1

- G^2**: curvature varies continuously (same tangent and curvature)**
  - Equivalently: The curve can be reparameterized so that it becomes C^2
  - Also equivalent: A unit speed parameterization would be C^2

$$\kappa = \|c''\|$$

Bézier Splines

Parameterization

Bézier spline curves

Local and global parameters:

- Given:
  - $b_0, \cdots, b_n$
  - $y(u)$: Bézier curve in interval $[0,1]$
  - $x(t)$: Bézier curve in interval $[t_i, t_{i+1}]$

- Setting $u(t) = \frac{t - t_i}{t_{i+1} - t_i}$

- Results in $x(t) = y\big(u(t)\big)$

The *local* parameter u runs from 0 to 1,
while the *global* parameter t runs from t_i to t_{i+1}

Bézier spline curves

$$u(t) = \frac{t - t_i}{t_{i+1} - t_i}$$

$$x(t) = y\big(u(t)\big)$$

Derivatives:

$$x'(t) = \quad y'\big(u(t)\big) \cdot u'(t) = \frac{y'\big(u(t)\big)}{t_{i+1} - t_i}$$

$$x''(t) = \quad y''\big(u(t)\big) \cdot \big(u'(t)\big)^2 + y'\big(u(t)\big) \cdot u''(t) = \frac{y''\big(u(t)\big)}{(t_{i+1} - t_i)^2}$$

…

$$x^{[n]}(t) = \frac{y^{[n]}\big(u(t)\big)}{(t_{i+1} - t_i)^n}$$

Bézier Curve

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

- Function value at $\{0,1\}$:

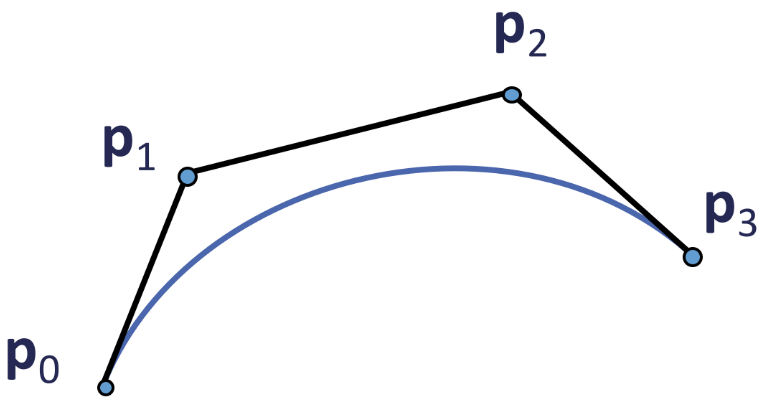
$$\boldsymbol{f}(0) = \boldsymbol{p}_0$$
$$\boldsymbol{f}(1) = \boldsymbol{p}_1$$

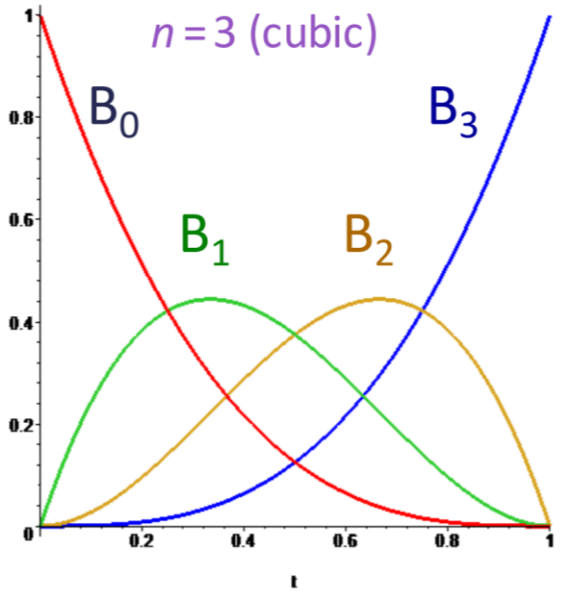
- First derivative vector at $\{0,1\}$

$$\boldsymbol{f}'(0) = n[\boldsymbol{p}_1 - \boldsymbol{p}_0]$$
$$\boldsymbol{f}'(1) = n[\boldsymbol{p}_n - \boldsymbol{p}_{n-1}]$$

- Second derivative vector at $\{0,1\}$

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





Bézier spline curves

Special cases:

$$\boldsymbol{x}'(t_i) = \frac{n\cdot(\boldsymbol{p}_1 - \boldsymbol{p}_0)}{t_{i+1} - t_i}$$

$$\boldsymbol{x}'(t_{i+1}) = \frac{n\cdot(\boldsymbol{p}_n - \boldsymbol{p}_{n-1})}{t_{i+1} - t_i}$$

$$\boldsymbol{x}''(t_i) = \frac{n\cdot(n-1)\cdot(\boldsymbol{p}_2 - 2\boldsymbol{p}_1 + \boldsymbol{p}_0)}{(t_{i+1} - t_i)^2}$$

$$\boldsymbol{x}''(t_{i+1}) = \frac{n\cdot(n-1)\cdot(\boldsymbol{p}_n - 2\boldsymbol{p}_{n-1} + \boldsymbol{p}_{n-2})}{(t_{i+1} - t_i)^2}$$

Bézier Splines

General Case

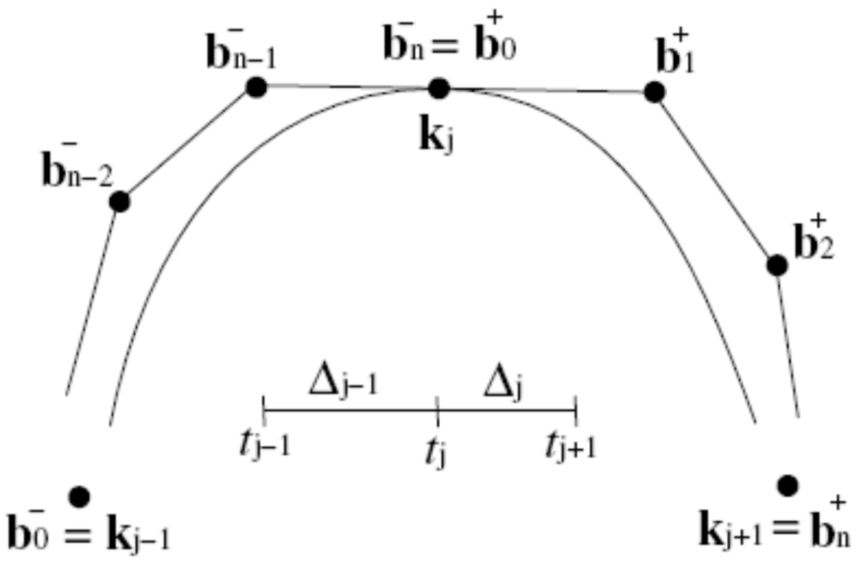
Bézier spline curves

Joining Bézier curves:

- Given: 2 Bézier curves of degree n through

$$\boldsymbol{k}_{j-1} = \boldsymbol{b}_0^-, \boldsymbol{b}_1^-, \dots, \boldsymbol{b}_n^- = \boldsymbol{k}_j$$

$$\boldsymbol{k}_j = \boldsymbol{b}_0^+, \boldsymbol{b}_1^+, \dots, \boldsymbol{b}_n^+ = \boldsymbol{k}_{j+1}$$

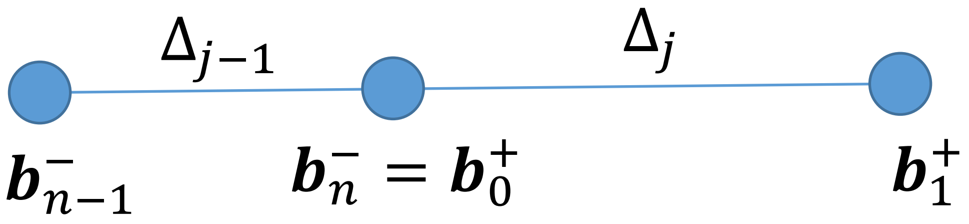


$$\boldsymbol{x}'(t_i) = \frac{n \cdot (\boldsymbol{b}_1 - \boldsymbol{b}_0)}{t_{i+1} - t_i}$$

Bézier spline curves

- Required: C^1-continuity at $\boldsymbol{k}_j$:
- $\boldsymbol{b}_{n-1}^-, \boldsymbol{k}_j, \boldsymbol{b}_1^+$ collinear and

$$\frac{\boldsymbol{b}_n^- - \boldsymbol{b}_{n-1}^-}{t_j - t_{j-1}} = \frac{\boldsymbol{b}_1^+ - \boldsymbol{b}_0^+}{t_{j+1} - t_j}$$



Bézier spline curves

- Required: G^1-continuity at $\boldsymbol{k}_j$:
  - $\boldsymbol{b}_{n-1}^-$, $\boldsymbol{k}_j$, $\boldsymbol{b}_1^+$ collinear
- Less restrictive than C^1-continuity

Bézier Splines

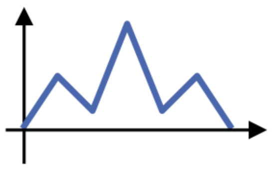
Choosing the degree

Choosing the Degree

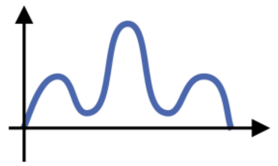
Candidates:

- $d = 0$ (piecewise constant) : not smooth

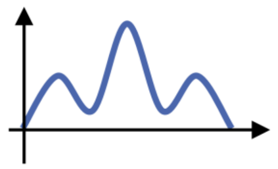
- $d = 1$ (piecewise linear) : not smooth enough



- $d = 2$ (piecewise quadratic) : constant 2nd derivative, still too inflexible



- $d = 3$ (piecewise cubic): degree of choice for computer graphics applications



Cubic Splines

Cubic piecewise polynomials:

- We can attain C^2 continuity without fixing the second derivative throughout the curve

Cubic Splines

Cubic piecewise polynomials:

- We can attain C^2 continuity without fixing the second derivative throughout the curve
- C^2 continuity is perceptually important
  - Motion: continuous position, velocity & acceleration
    Discontinuous acceleration noticeable (object/camera motion)
  - We can see second order shading discontinuities
    (esp.: reflective objects)

Cubic Splines

Cubic piecewise polynomials

- We can attain C^2 continuity without fixing the second derivative throughout the curve
- C^2 continuity is perceptually important
  - We can see second order shading discontinuities

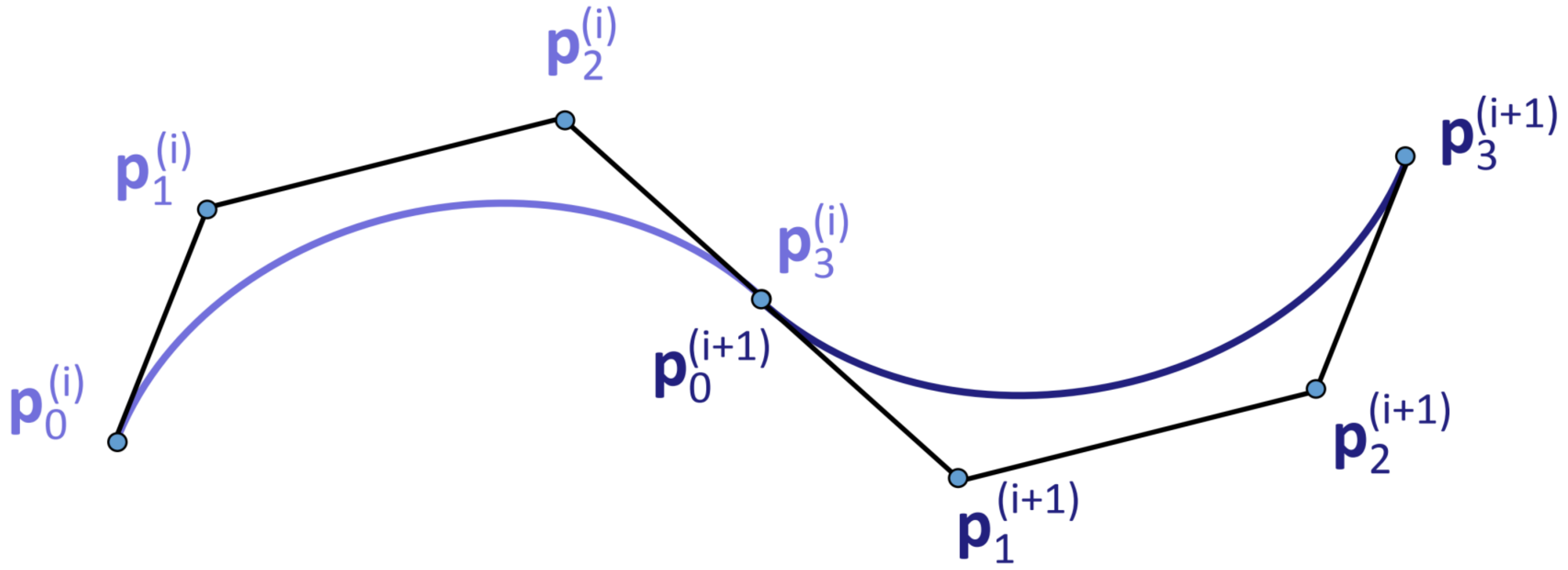
    (esp.: reflective objects)
  - Motion: continuous *position, velocity* & *acceleration*

    Discontinuous acceleration noticeable (object/camera motion)
- One more argument for cubics:
  - Among all C^2 curves that interpolate a set of points (and obey to the same end condition), a piecewise cubic curve has the least integral acceleration ("smoothest curve you can get").

Bézier Splines

Local control: Bézier splines

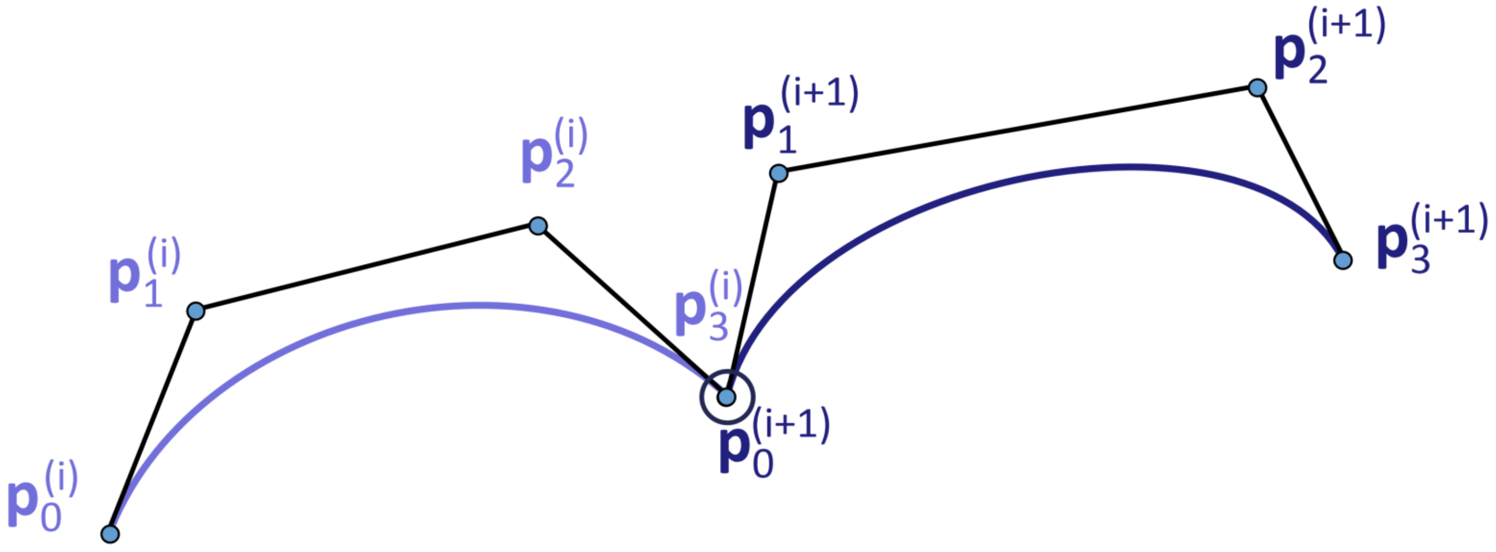
- Concatenate several curve segments
- Question: Which constraints to place upon the control points in order to get C^{-1}, C^{0}, C^{1}, C^{2} continuity?



Bézier Spline Continuity

Rules for Bézier spline continuity:

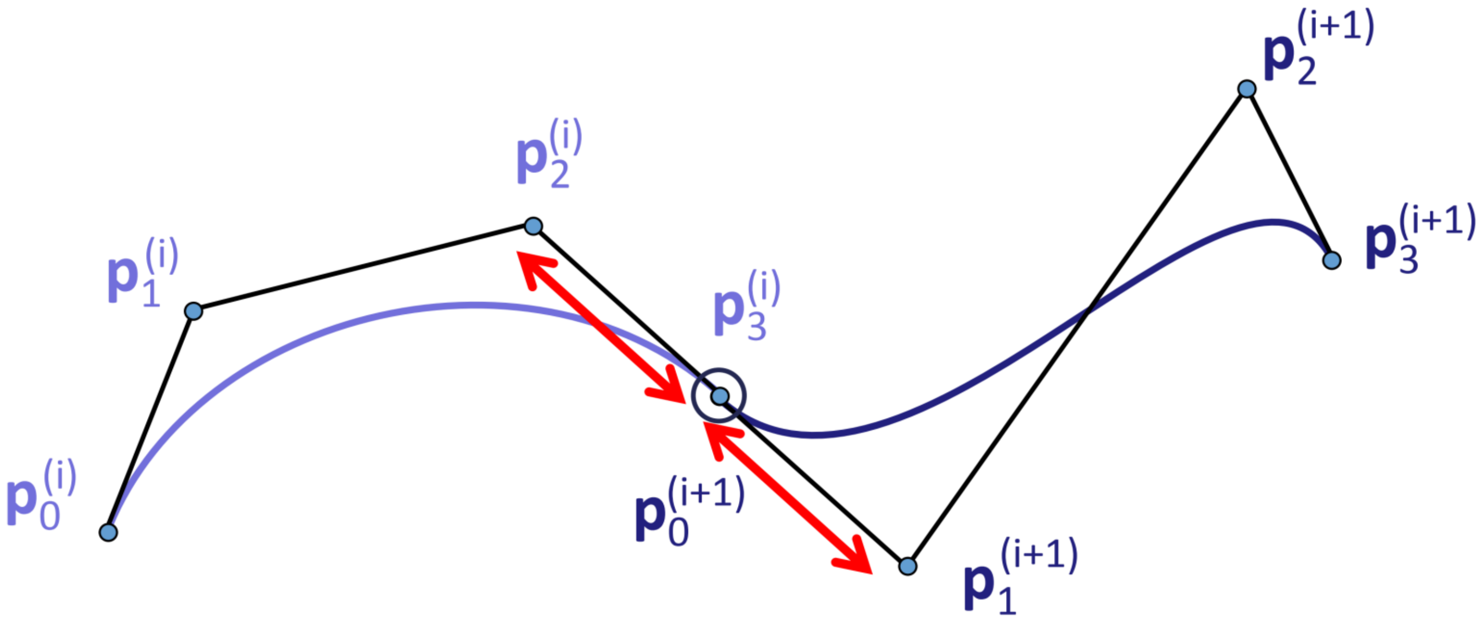
- C^0 continuity:
  - Each spline segment interpolates the first and last control point
  - Therefore: Points of neighboring segments have to coincide for C^0 continuity



Bézier Spline Continuity

Rules for Bézier spline continuity:

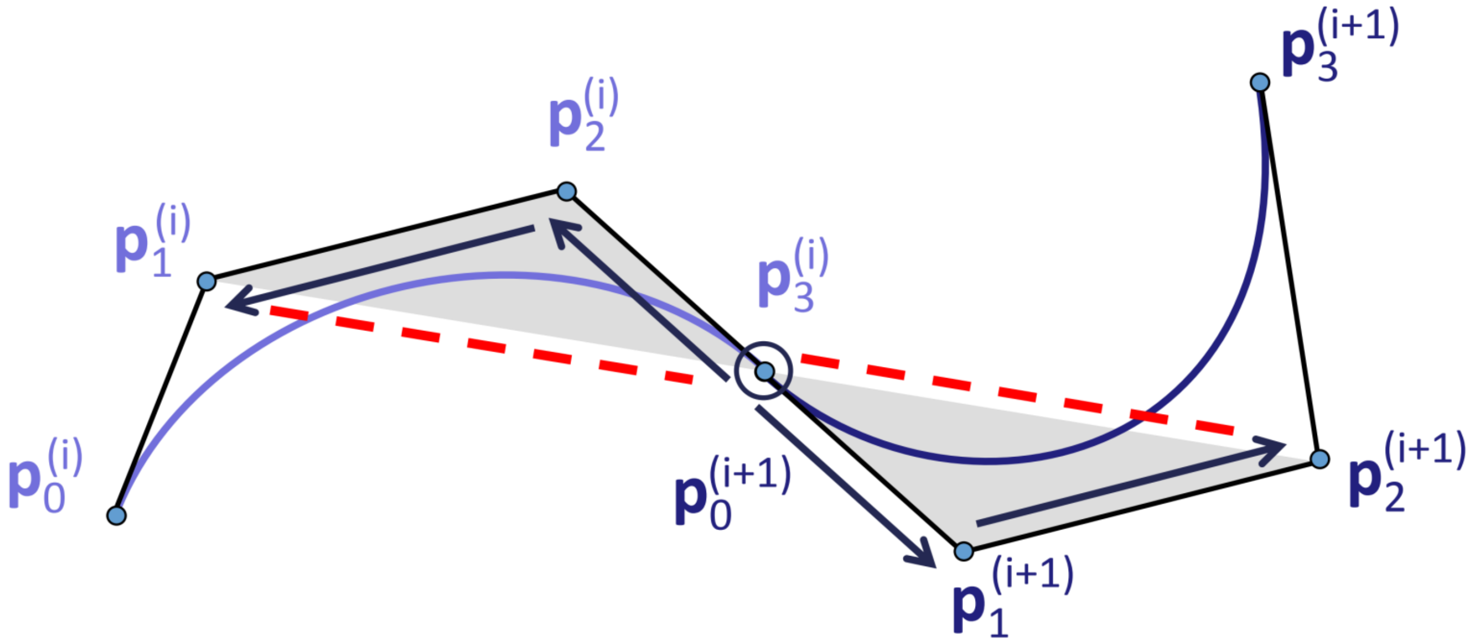
- Additional requirement for C^1 continuity:
  - Tangent vectors are proportional to differences $\boldsymbol{p}_1 - \boldsymbol{p}_0$, $\boldsymbol{p}_n - \boldsymbol{p}_{n-1}$
  - Therefore: These vectors must be identical for C^1 continuity



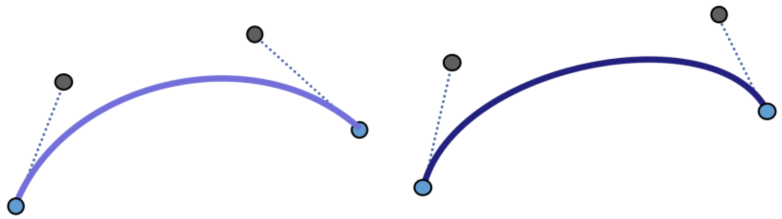
Bézier Spline Continuity

Rules for Bézier spline continuity

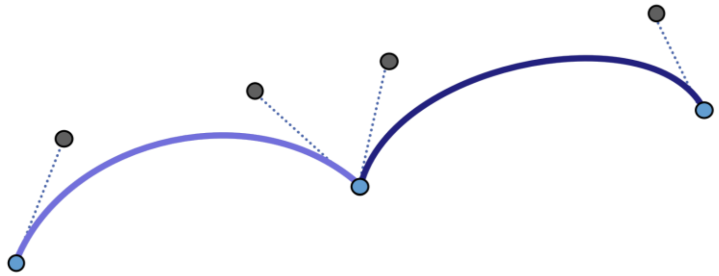
- Additional requirement for C^2 continuity:
  - d^2/dt^2 vectors are prop. to $\boldsymbol{p}_2 - 2\boldsymbol{p}_1 + \boldsymbol{p}_0$, $\boldsymbol{p}_n - 2\boldsymbol{p}_{n-1} + \boldsymbol{p}_{n-2}$
  - Tangents must be the same (C^2 implies C^1)



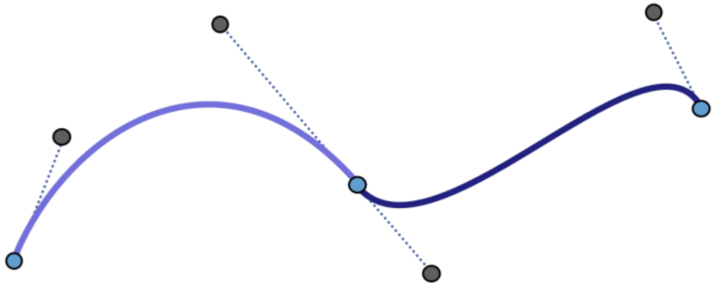
Continuity



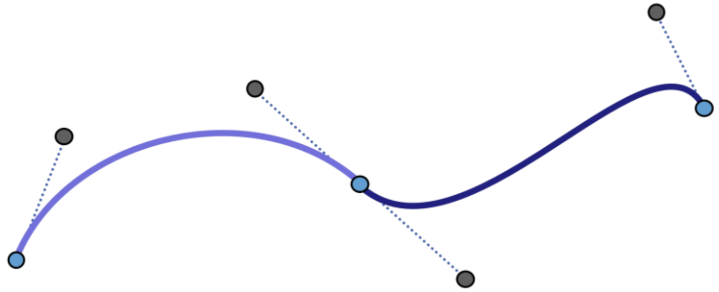
C^{-1} continuity



C^{0} continuity



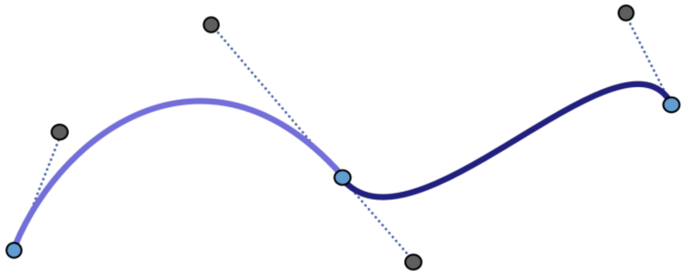
G^{1} continuity



C^{1} continuity

Continuity for Bézier Splines

This means



G^1 continuity

This Bézier curve is G^1: It can be reparameterized to become C^1.
(Just increase the speed for the second segment by ratio of tangent vector lengths)

In Practice

- Everyone is using cubic Bézier curves
- Higher degree are rarely used (some CAD/CAM applications)
- Typically: “points & handles” interface
- Four modes:
    - Discontinuous (two curves)
    - C^0 Continuous (points meet)
    - G^1 continuous: Tangent direction continuous
        - Handles point into the same direction, but different length
    - C^1 continuous
        - Handle points have symmetric vectors
- C^2 is more restrictive: control via k_i

Bézier spline curves

- Required: C^2-continuity at $\boldsymbol{k}_j$
- C^1 implies $$\frac{\boldsymbol{b}_n^- - \boldsymbol{b}_{n-1}^-}{t_j - t_{j-1}} = \frac{\boldsymbol{b}_1^+ - \boldsymbol{b}_0^+}{t_{j+1} - t_j}$$
- C^2 implies $$\frac{\boldsymbol{b}_n^- - 2\boldsymbol{b}_{n-1}^- + \boldsymbol{b}_{n-2}^-}{\left(t_j - t_{j-1}\right)^2} = \frac{\boldsymbol{b}_2^+ - 2\boldsymbol{b}_1^+ + \boldsymbol{b}_0^+}{\left(t_{j+1} - t_j\right)^2}$$

$$\frac{t_{j+1} - t_j}{t_j - t_{j-1}} = \frac{\Delta_j}{\Delta_{j-1}}$$

Bézier spline curves

- Required: C^2-continuity at $\boldsymbol{k}_j$:

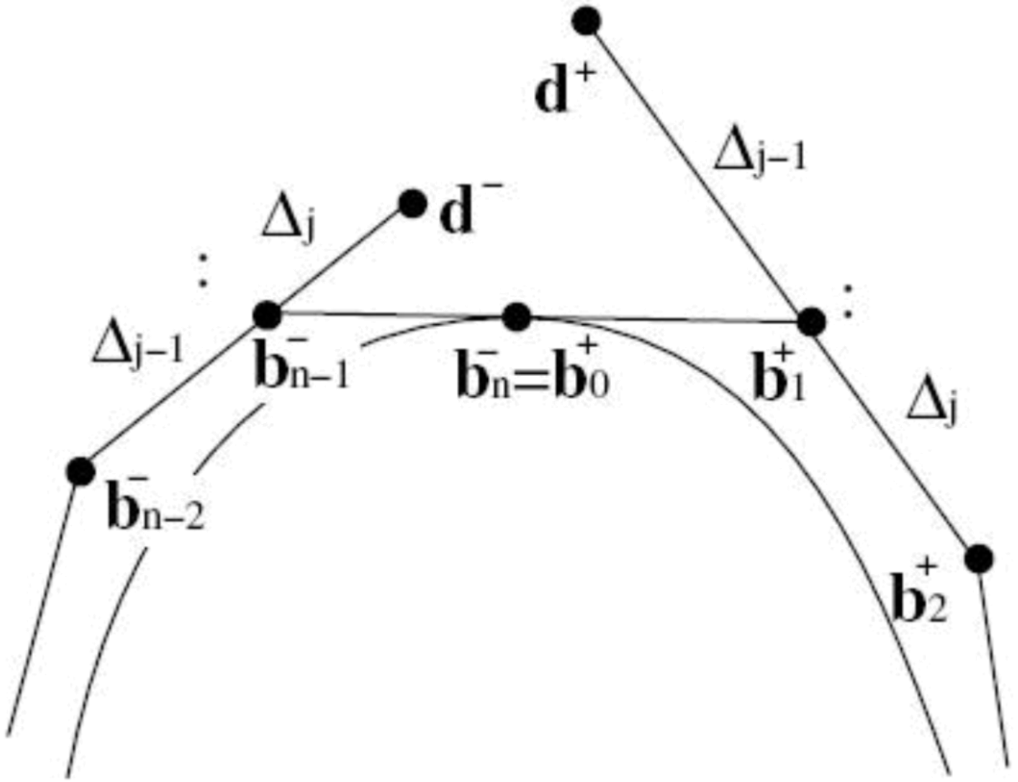
- Introduce $\boldsymbol{d}^- = \boldsymbol{b}_{n-1}^- + \frac{\Delta_j}{\Delta_{j-1}}(\boldsymbol{b}_{n-1}^- - \boldsymbol{b}_{n-2}^-)$

  and $\boldsymbol{d}^+ = \boldsymbol{b}_1^+ - \frac{\Delta_{j-1}}{\Delta_j}(\boldsymbol{b}_2^+ - \boldsymbol{b}_1^+)$

- By manipulating equation from the previous slides
- C^2-continuity $\Leftrightarrow$ C^1-continuity and $\boldsymbol{d}^- = \boldsymbol{d}^+$

Bézier spline curves

C^2-continuity $\Leftrightarrow$ C^1-continuity and $\boldsymbol{d}^- = \boldsymbol{d}^+$

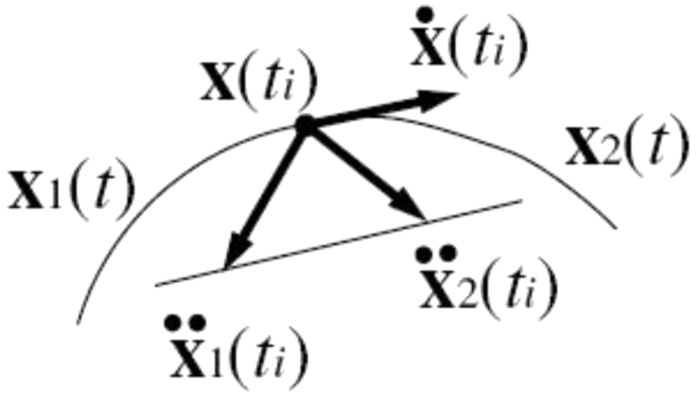


Bézier spline curves

- G^2-continuity in general (for all types of curves):
- Given:
  - $\boldsymbol{x}_1(t)$, $\boldsymbol{x}_2(t)$ with
  - $\boldsymbol{x}_1(t_i) = \boldsymbol{x}_2(t_i) = \boldsymbol{x}(t_i)$
  - $\boldsymbol{x}_1'(t_i) = \boldsymbol{x}_2'(t_i) = \boldsymbol{x}(t_i)$
- Then the requirement for G^2-continuity at $\boldsymbol{t = t_i}$:

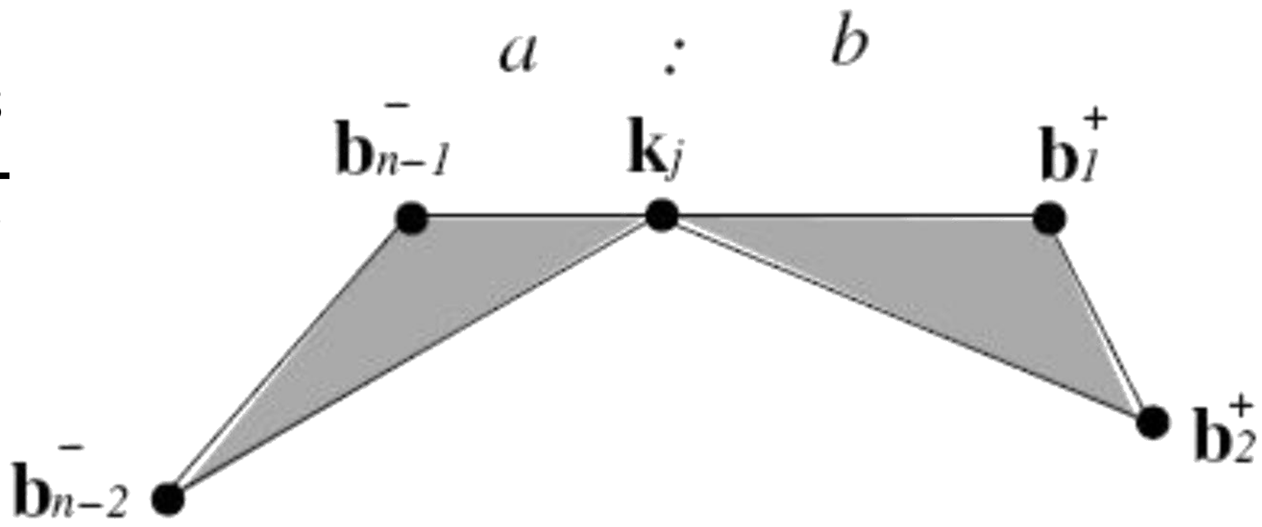
$$\boldsymbol{x}_2''(t_i) - \boldsymbol{x}_1''(t_i) \parallel \boldsymbol{x}'(t_i)$$

Parallel



Bézier spline curves

- Required: G^2-continuity at k_j:
- G^1-continuity
- Co-planarity for : $\boldsymbol{b}_{n-2}^{-}, \boldsymbol{b}_{n-1}^{-}, \boldsymbol{k}_j, \boldsymbol{b}_1^{+}, \boldsymbol{b}_2^{+}$
- And: $\dfrac{\text{area}(\boldsymbol{b}_{n-2}^{-}, \boldsymbol{b}_{n-1}^{-}, \boldsymbol{k}_j)}{\text{area}(\boldsymbol{k}_j, \boldsymbol{b}_1^{+}, \boldsymbol{b}_2^{+})} = \dfrac{a^3}{b^3}$



Bézier Splines

C^2 Cubic Bézier Splines

Cubic Bézier Splines

Cubic Bézier spline curves

- Given:

$$\boldsymbol{k}_0, \dots, \boldsymbol{k}_n \in \mathbb{R}^3 \qquad \text{control points}$$

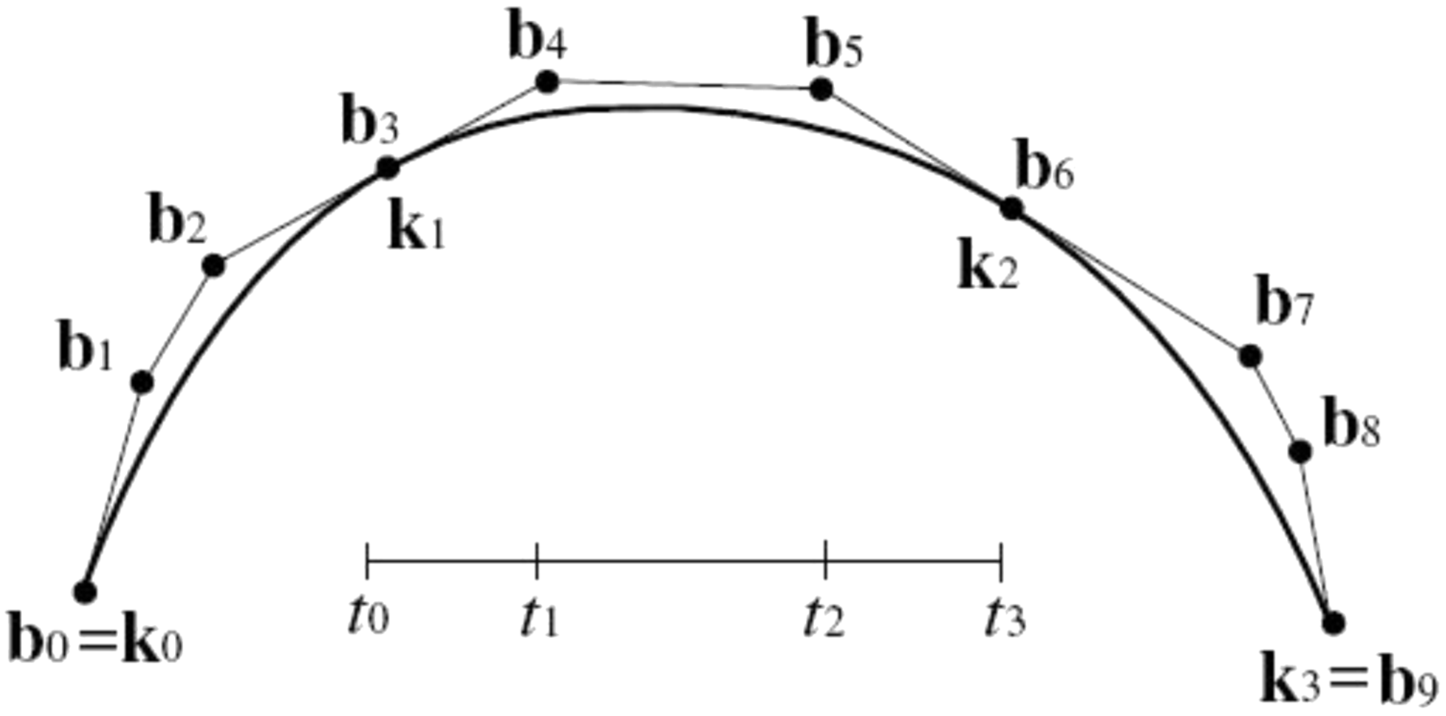
$$t_0, \dots, t_n \in \mathbb{R} \qquad \text{knot sequence}$$

$$t_i < t_{i+1}, \text{ for } i = 0, \dots, n_1$$

- Wanted: Bézier points $\boldsymbol{b}_0, \dots, \boldsymbol{b}_{3n}$ for an interpolating C^2-continuous piecewise cubic Bézier spline curve

Cubic Bézier Splines

Examples: $n = 3$:



Cubic Bézier Splines

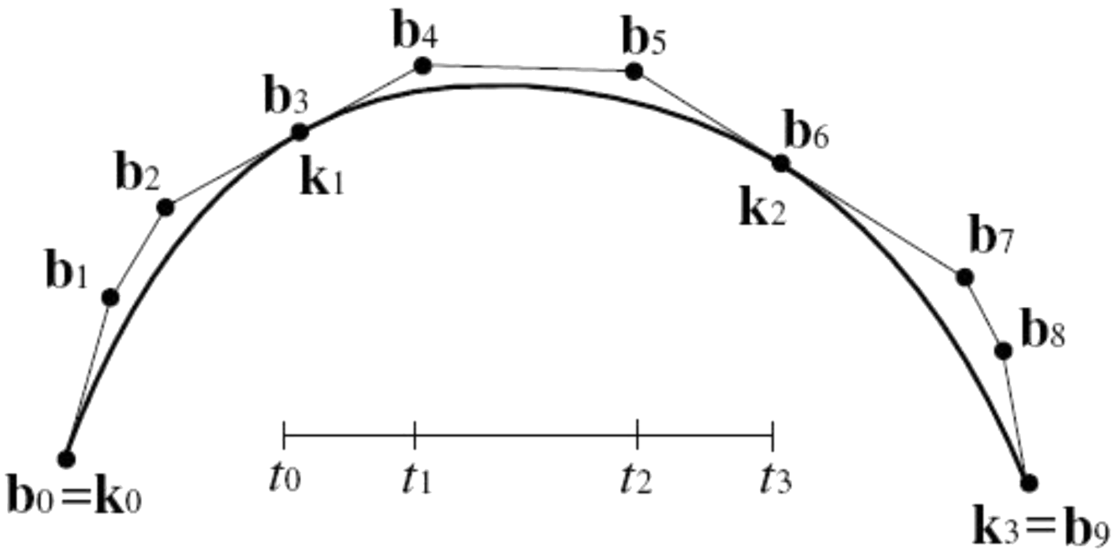
- $3n + 1$ unknown points
- $b_{3i} = k_i$ for $i = 0, \dots, n$

  $n + 1$ equations
- C^1 in points k_i for $i = 1, \dots, n - 1$

  $n - 1$ equations
- C^2 in points k_i for $i = 1, \dots, n - 1$

  $n - 1$ equations

$3n - 1$ equations



$\Rightarrow$ **2 additional conditions necessary: end conditions**

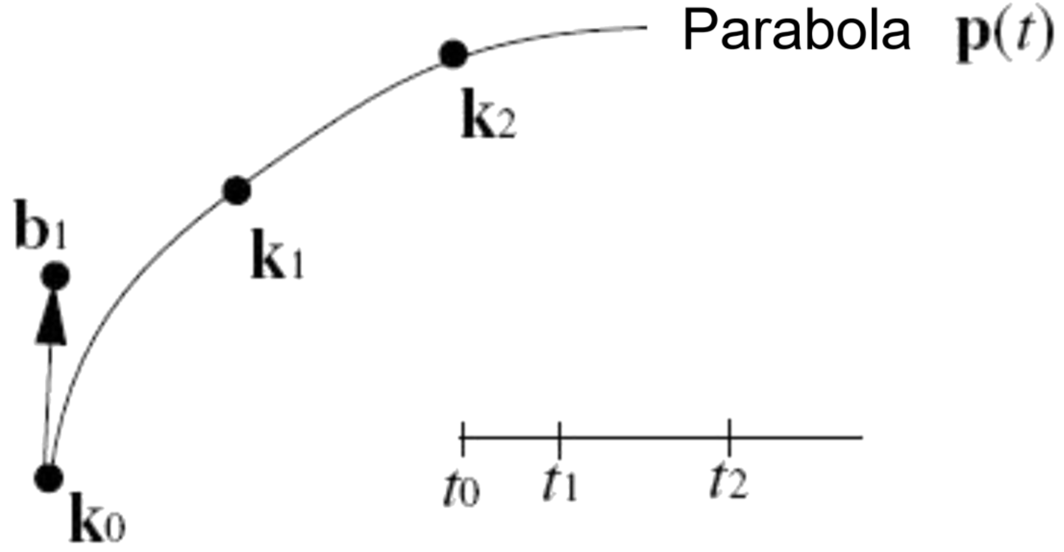
Bézier Splines

C^2 Cubic Bézier Splines: End conditions

Bézier spline curves: End conditions

Bessel's end condition

- The tangential vector in $\boldsymbol{k}_0$ is equivalent to the tangential vector of the parabola interpolating $\{\boldsymbol{k}_0, \boldsymbol{k}_1, \boldsymbol{k}_2\}$ at $\boldsymbol{k}_0$:



$$\dot{\boldsymbol{x}}(t_i) = \frac{n\cdot(\boldsymbol{b}_1 - \boldsymbol{b}_0)}{t_{i+1} - t_i}$$

Bézier spline curves: End conditions

Parabola Interpolating $\{\boldsymbol{k}_0, \boldsymbol{k}_1, \boldsymbol{k}_2\}$

$$\boldsymbol{p}(t) = \frac{(t_2 - t)(t_1 - t)}{(t_2 - t_0)(t_1 - t_0)}\boldsymbol{k}_0 + \frac{(t_2 - t)(t - t_0)}{(t_2 - t_1)(t_1 - t_0)}\boldsymbol{k}_1 + \frac{(t_0 - t)(t_1 - t)}{(t_2 - t_1)(t_2 - t_0)}\boldsymbol{k}_2$$

Its derivative

$$\boldsymbol{p}'(t_0) = -\frac{(t_2 - t_0) + (t_1 - t_0)}{(t_2 - t_0)(t_1 - t_0)}\boldsymbol{k}_0 + \frac{(t_2 - t_0)}{(t_2 - t_1)(t_1 - t_0)}\boldsymbol{k}_1 - \frac{(t_1 - t_0)}{(t_2 - t_1)(t_2 - t_0)}\boldsymbol{k}_2$$

Location of $\boldsymbol{b}_1$

$$\boldsymbol{b}_1 = \boldsymbol{b}_0 + \frac{t_1 - t_0}{3}\boldsymbol{p}'(t_0)$$

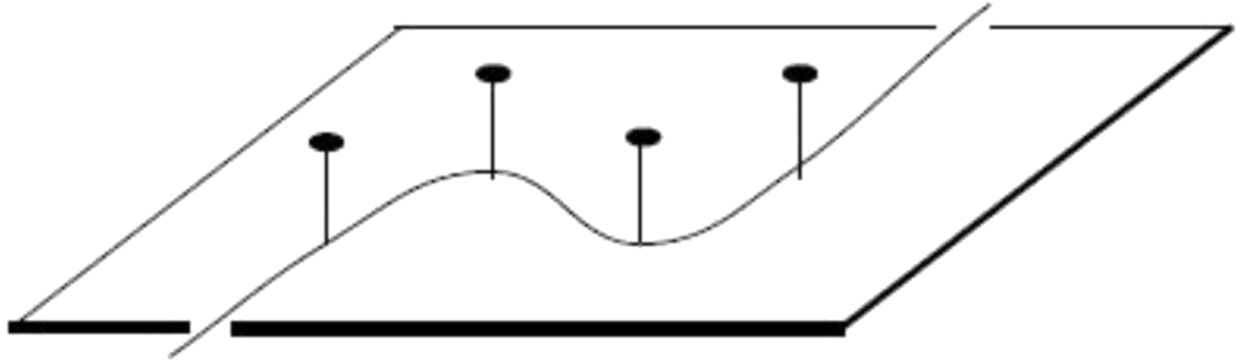
$$\ddot{\boldsymbol{x}}(t_i) = \frac{n \cdot (n-1) \cdot (\boldsymbol{b}_2 - 2\boldsymbol{b}_1 + \boldsymbol{b}_0)}{(t_{i+1} - t_i)^2}$$

Bézier spline curves: End conditions

- Natural end condition:

$$\boldsymbol{x}''(t_0) = 0 \Leftrightarrow \boldsymbol{b}_1 = \frac{\boldsymbol{b}_2 + \boldsymbol{b}_0}{2}$$

$$\boldsymbol{x}''(t_n) = 0 \Leftrightarrow \boldsymbol{b}_{3n-1} = \frac{\boldsymbol{b}_{3n-2} + \boldsymbol{b}_{3n}}{2}$$



End conditions: Examples

- *Bessel* end condition

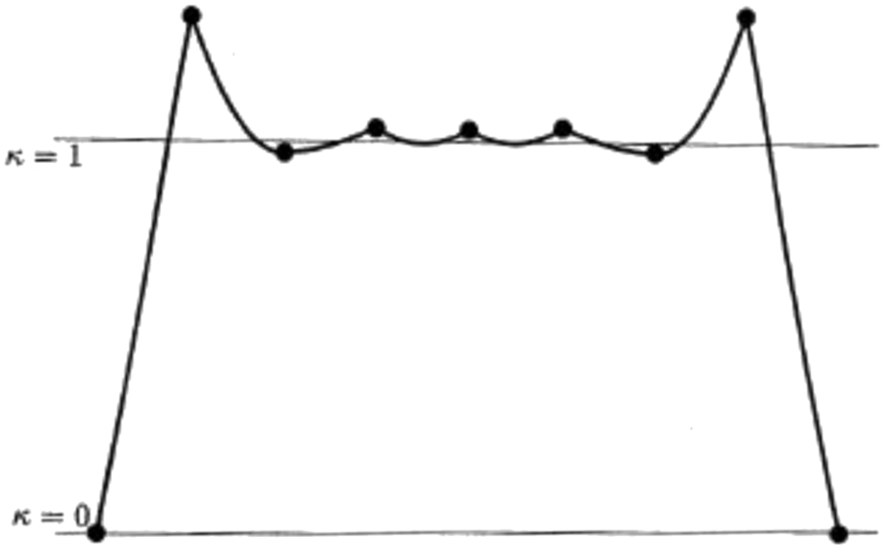
Curve: circle of radius 1

Curvature plot

End conditions: Examples

- *Natural* end condition

Curve: circle of radius 1



Curvature plot

Bézier Splines

C^2 Cubic Bézier Splines: parameterization

Bézier spline curves: Parameterization

Approach so far:

- Given: control points $\boldsymbol{k}_0, \dots, \boldsymbol{k}_n$ and knot sequence $t_0 < \cdots < t_n$

- Wanted: interpolating curve

- Problem: Normally, the knot sequence is not given, but it influences the curve

Bézier spline curves: Parameterization

- **Equidistant (uniform) parameterization**
  - $t_{i+1} - t_i = \text{const}$
  - e.g. $t_i = i$
  - Geometry of the data points is not considered
- **Chordal parameterization**
  - $t_{i+1} - t_i = \|\boldsymbol{k}_{i+1} - \boldsymbol{k}_i\|$
  - Parameter intervals proportional to the distances of neighbored control points

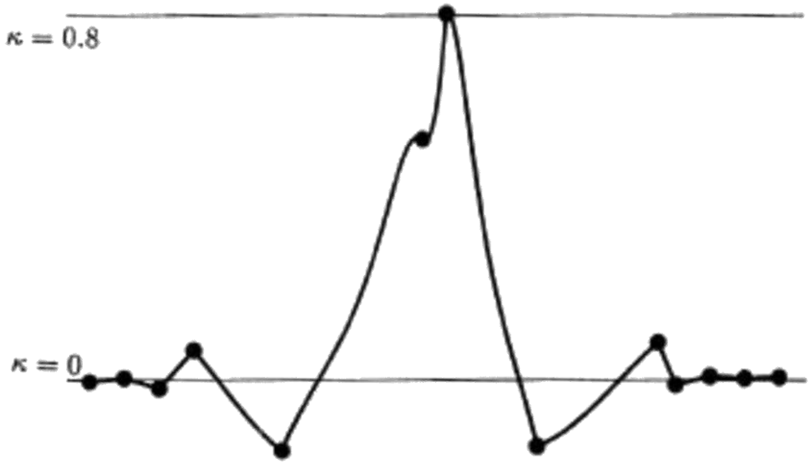
Bézier spline curves: Parameterization

- Centripetal parameterization
  - $t_{i+1} - t_i = \sqrt{\|\boldsymbol{k}_{i+1} - \boldsymbol{k}_i\|}$
- Foley parameterization
  - Involvement of angles in the control polygon
  - $t_{i+1} - t_i = \|\boldsymbol{k}_{i+1} - \boldsymbol{k}_i\| \cdot \left(1 + \frac{3}{2}\frac{\hat{\alpha}_i \|\boldsymbol{k}_i - \boldsymbol{k}_{i-1}\|}{\|\boldsymbol{k}_i - \boldsymbol{k}_{i-1}\| + \|\boldsymbol{k}_{i+1} - \boldsymbol{k}_i\|} + \frac{3}{2}\frac{\hat{\alpha}_{i+1} \|\boldsymbol{k}_{i+1} - \boldsymbol{k}_i\|}{\|\boldsymbol{k}_{i+1} - \boldsymbol{k}_i\| + \|\boldsymbol{k}_{i+2} - \boldsymbol{k}_{i+1}\|}\right)$
  - with $\hat{\alpha}_i = \min\left(\pi - \alpha_i, \frac{\pi}{2}\right)$
  - and $\alpha_i = \text{angle}(\boldsymbol{k}_{i-1}, \boldsymbol{k}_i, \boldsymbol{k}_{i+1})$
- Affine invariant parameterization
  - Parameterization on the basis of an affine invariant distance measure (e.g. G. Nielson)

Bézier spline curves: Parameterization

- Examples: Chordal parameterization

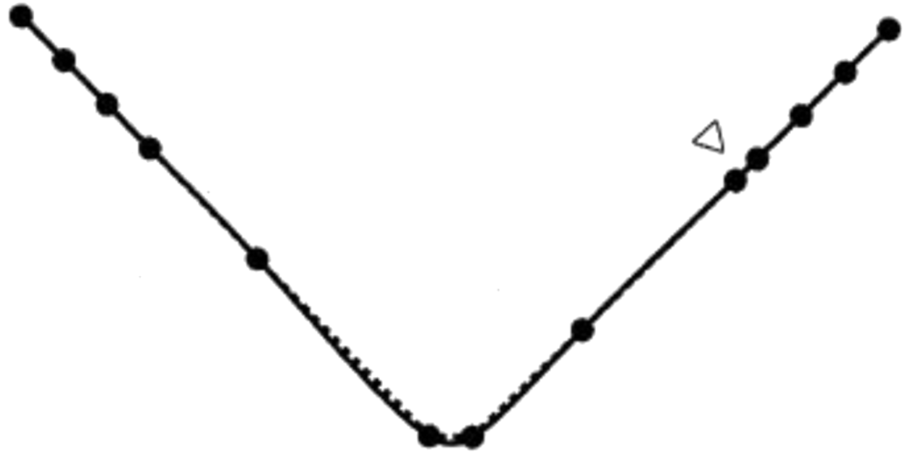
Curve



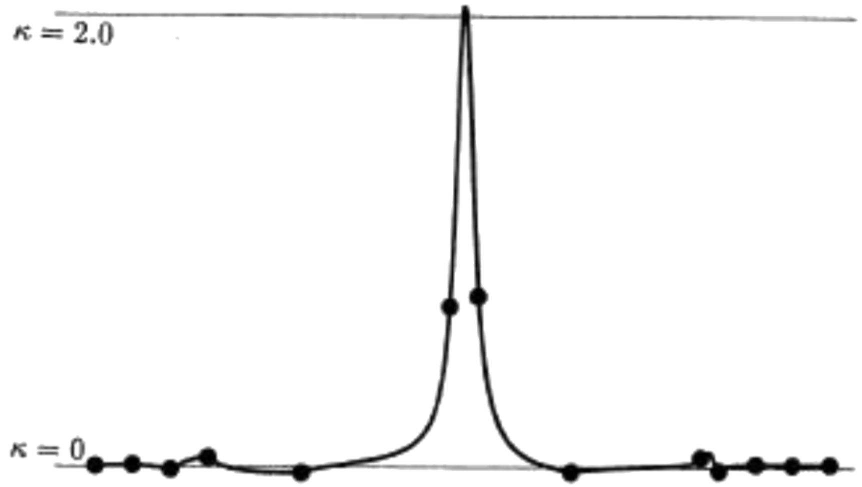
Curvature plot

Bézier spline curves: Parameterization

- Examples: Centripetal parameterization



Curve

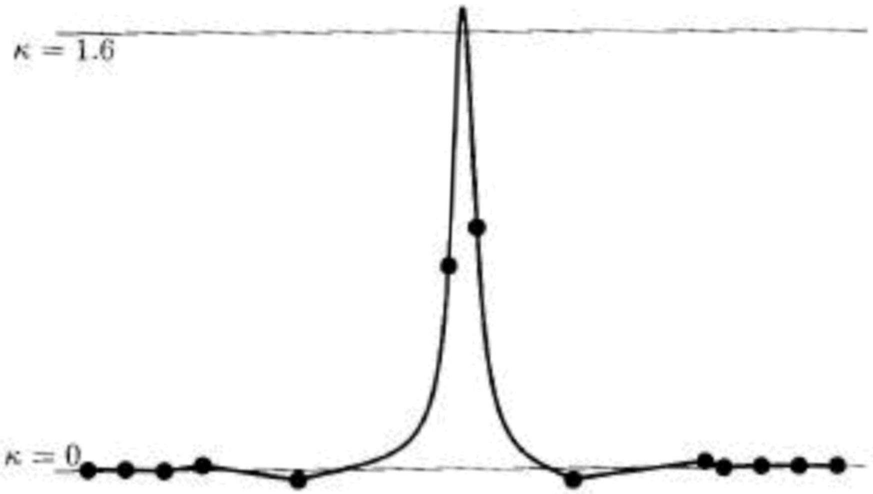


Curvature plot

Bézier spline curves: Parameterization

- Examples: Foley parameterization

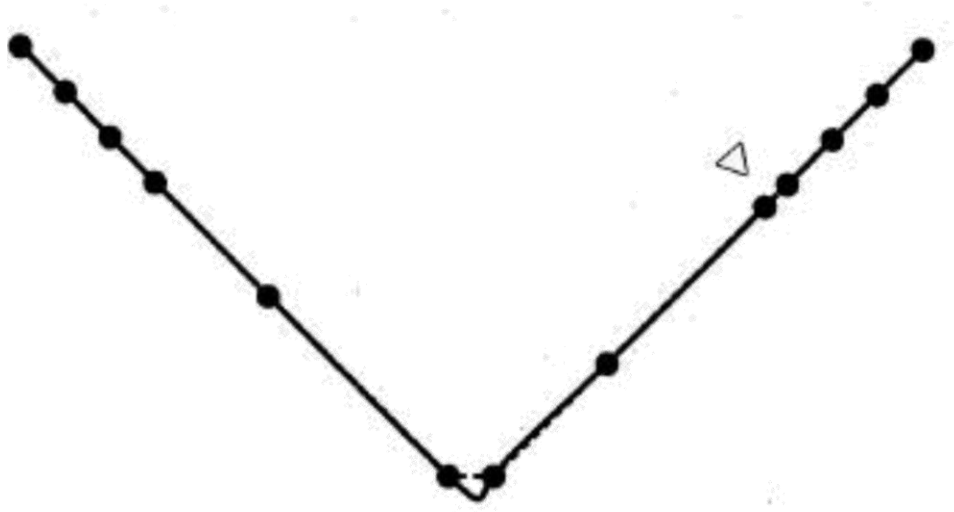
Curve



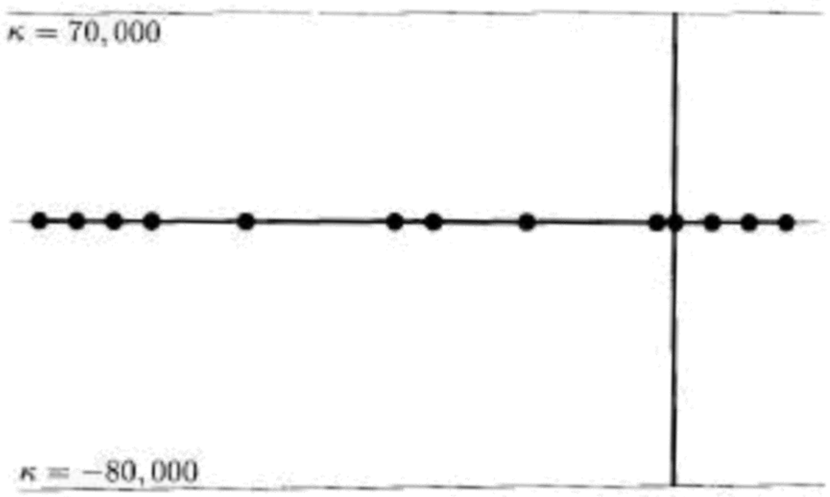
Curvature plot

Bézier spline curves: Parameterization

- Examples: Uniform parameterization



Curve



Curvature plot

Bézier Splines

C^2 Cubic Bézier Splines: closed curves

Closed cubic Bézier spline curves

Closed cubic Bézier spline curves

- Given:

  $\boldsymbol{k}_0, \dots, \boldsymbol{k}_{n-1}, \boldsymbol{k}_n = \boldsymbol{k}_0$: control points

  $t_0 < \cdots < t_n$: knot sequence

- As an "end condition" for the piecewise cubic curve we place:

$$\boldsymbol{x}'(t_0) = \boldsymbol{x}'(t_n)$$

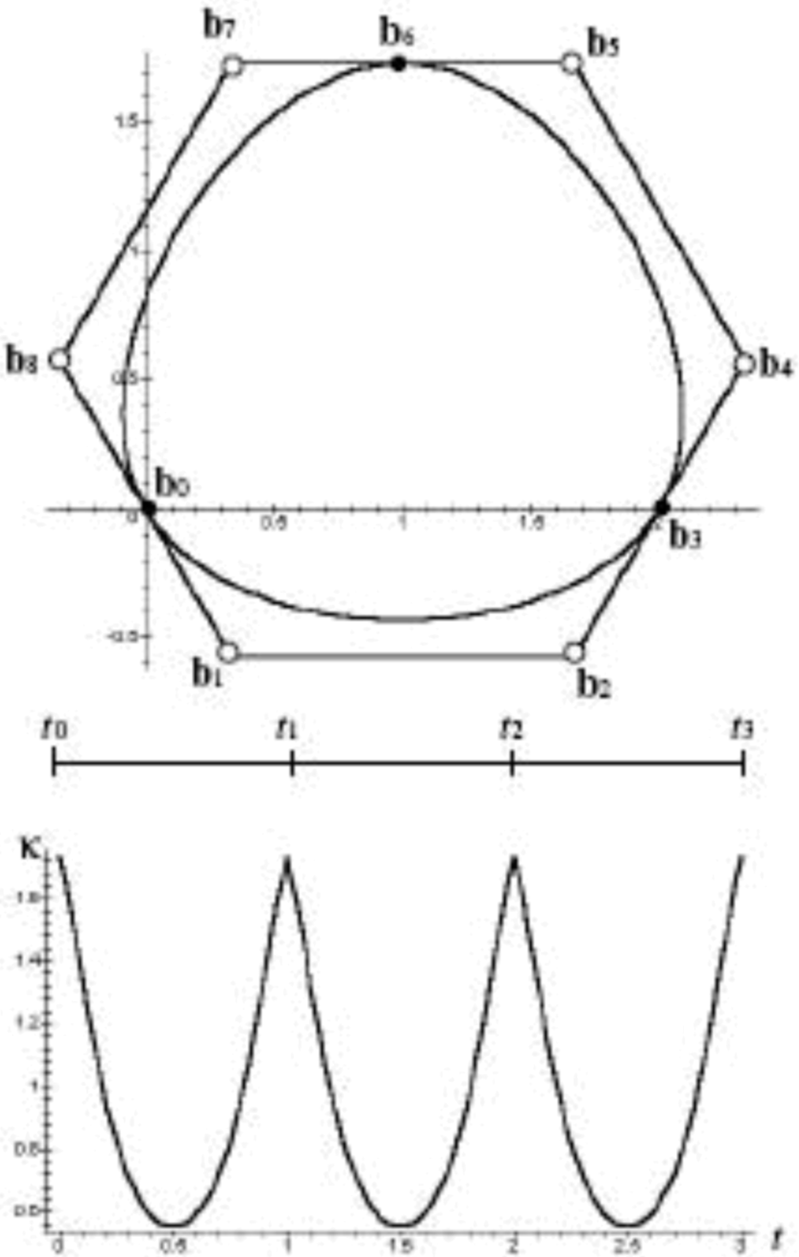
$$\boldsymbol{x}''(t_0) = \boldsymbol{x}''(t_n)$$

Closed cubic Bézier spline curves

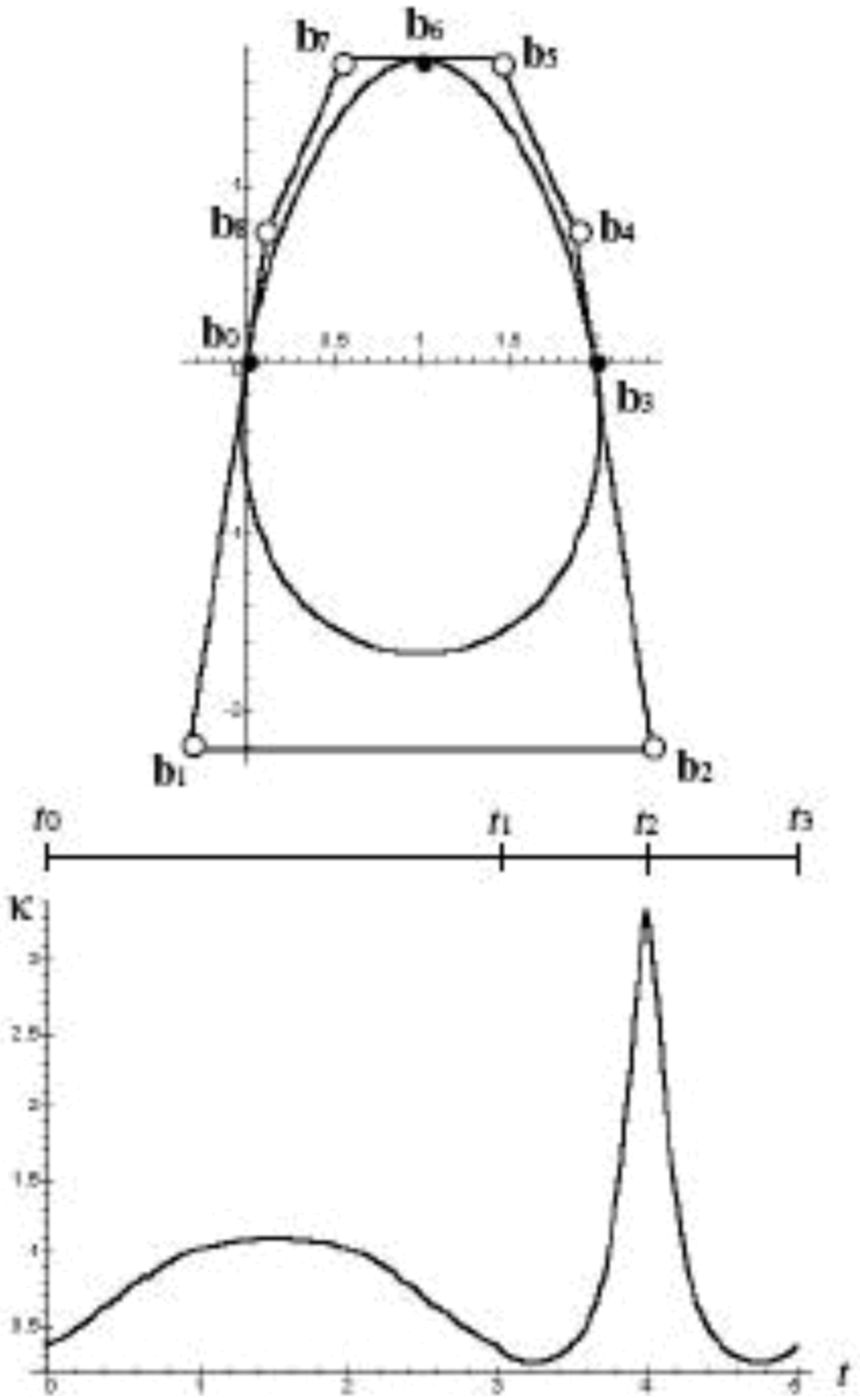
Closed cubic Bézier spline curves

- $\rightarrow C^2$-continuous and closed curve
- Advantage of closed curves: selection of the end condition is not necessary!
- Examples (on the next 3 slides): $n = 3$

Examples



Examples



Examples