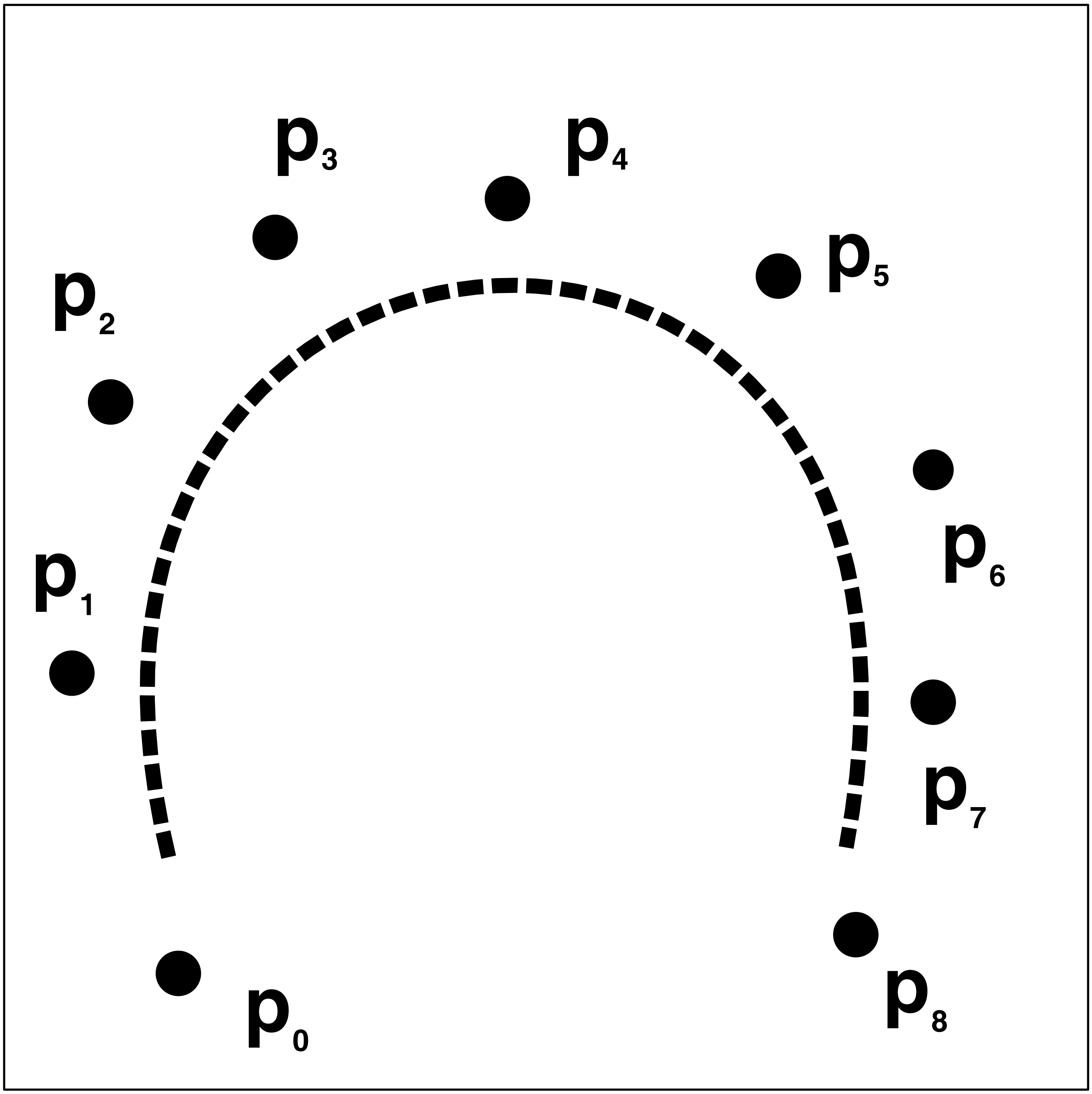
C^2 Approximating Splines



Uniform B-Spline

- **B-spline control points:** $\mathbf{p}_0, \ldots, \mathbf{p}_n$
- **Piecewise Bezier curves with C^2 continuity at joints**
- **Bezier control points:**

$$\mathbf{v}_1 = \frac{\mathbf{2p_1 + p_2}}{\mathbf{3}}$$

$$\mathbf{v}_2 = \frac{\mathbf{p_1 + 2p_2}}{\mathbf{3}}$$

$$\mathbf{v}_0 = \frac{1}{2}\left(\frac{\mathbf{p}_0 + 2\mathbf{p}_1}{3} + \frac{2\mathbf{p}_1 + \mathbf{p}_2}{3}\right)$$

$$= \frac{1}{6}(\mathbf{p}_0 + 4\mathbf{p}_1 + \mathbf{p}_2)$$

$$\mathbf{v}_3 = \frac{1}{6}(\mathbf{p}_1 + 4\mathbf{p}_2 + \mathbf{p}_3)$$

- **In general, i-th segment of B-splines is determined by**

$\mathbf{p}_i, \mathbf{p}_{i+1}, \mathbf{p}_{i+2}, \mathbf{p}_{i+3}$

$$\mathbf{v}_1 = \frac{2\mathbf{p}_{i+1} + \mathbf{p}_{i+2}}{3}$$

$$\mathbf{v}_2 = \frac{\mathbf{p}_{i+1} + 2\mathbf{p}_{i+2}}{3}$$

$$\mathbf{v}_0 = \frac{1}{6}(\mathbf{p}_i + 4\mathbf{p}_{i+1} + \mathbf{p}_{i+2})$$

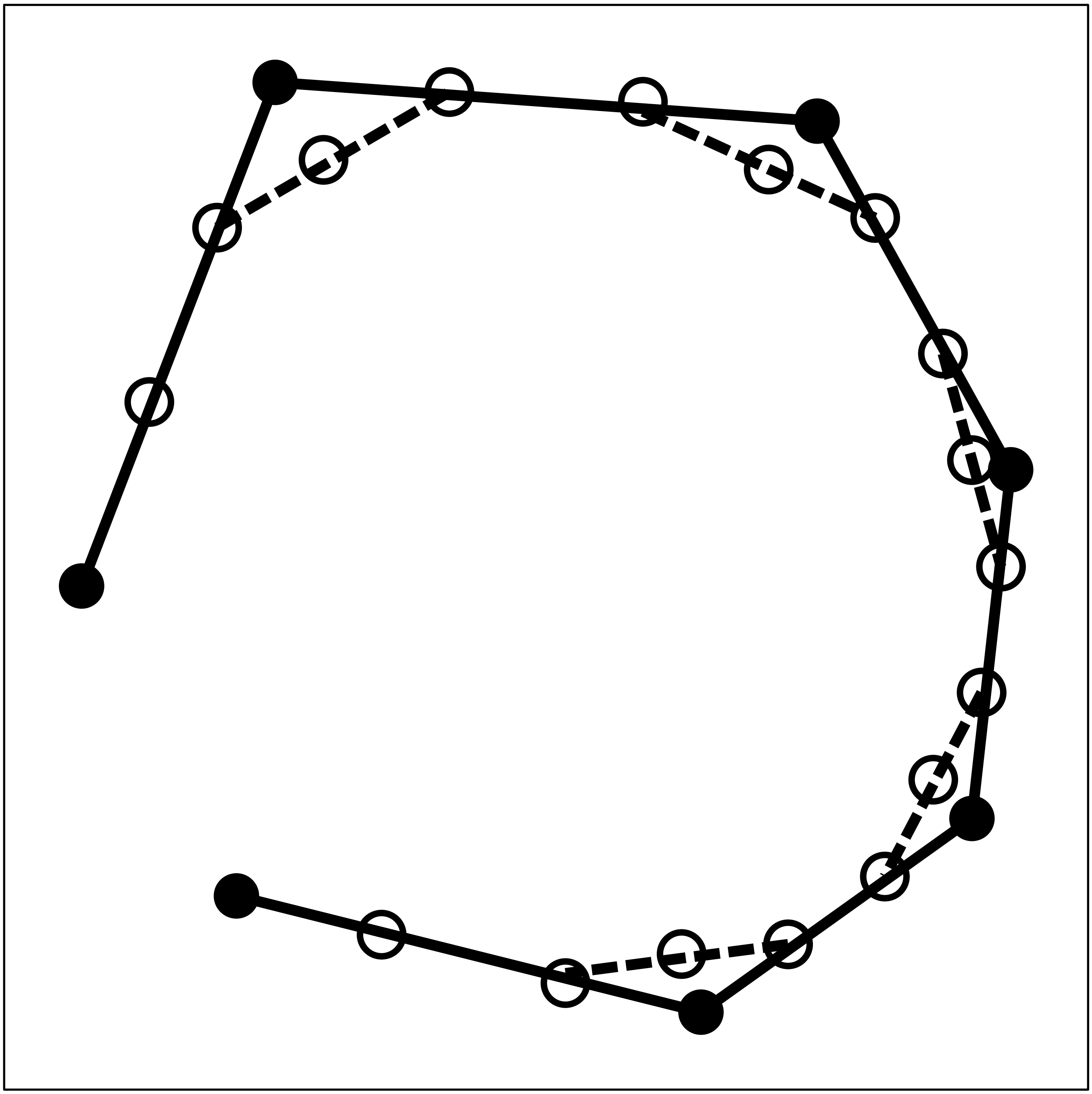
$$\mathbf{v}_3 = \frac{1}{6}(\mathbf{p}_{i+1} + 4\mathbf{p}_{i+2} + \mathbf{p}_{i+3})$$

- **In matrix form**

$$\begin{bmatrix} \mathbf{v}_0 \\ \mathbf{v}_1 \\ \mathbf{v}_2 \\ \mathbf{v}_3 \end{bmatrix} = \frac{1}{6} \begin{bmatrix} 1 & 4 & 1 & 0 \\ 0 & 4 & 2 & 0 \\ 0 & 2 & 4 & 0 \\ 0 & 1 & 4 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{p}_i \\ \mathbf{p}_{i+1} \\ \mathbf{p}_{i+2} \\ \mathbf{p}_{i+3} \end{bmatrix}$$

- **Question: how many bezier segments???**

From B-Splines to Bezier



B-Spline Properties

- C^2 continuity
- Approximation
- Local control
- Each segment are determined by four control points
- Convex hull
- Questions: what happens if we put more than one control points in the same location???
  - double vertices
  - triple vertices
  - collinear vertices
- End conditions
  - double endpoint: curve will be tangent to line between first distinct points

- triple endpoint: curve interpolate endpoint, start w
  a line segment

- Display B-splines: transform it to Bezier curves

Catmull-Rom Splines

- Keep interpolation
- Give up C^2 continuity
- Control tangents locally
- Idea: Bezier curve between successive points
- How to determine two internal vertices: $\mathbf{v}_1, \mathbf{v}_2$
- $\mathbf{c}(0) = \mathbf{p}_i = \mathbf{v}_0$
- $\mathbf{c}(1) = \mathbf{p}_{i+1} = \mathbf{v}_3$
- $\mathbf{c}'(0) = \frac{\mathbf{p}_{i+1} - \mathbf{p}_{i-1}}{2} = 3(\mathbf{v}_1 - \mathbf{v}_0)$
- $\mathbf{c}'(1) = \frac{\mathbf{p}_{i+2} - \mathbf{p}_{i}}{2} = 3(\mathbf{v}_3 - \mathbf{v}_2)$
- $\mathbf{v}_1 = \frac{\mathbf{p}_{i+1} + 6\mathbf{p}_i - \mathbf{p}_{i-1}}{6}$
- $\mathbf{v}_2 = \frac{-\mathbf{p}_{i+2} + 6\mathbf{p}_{i+1} + \mathbf{p}_i}{6}$

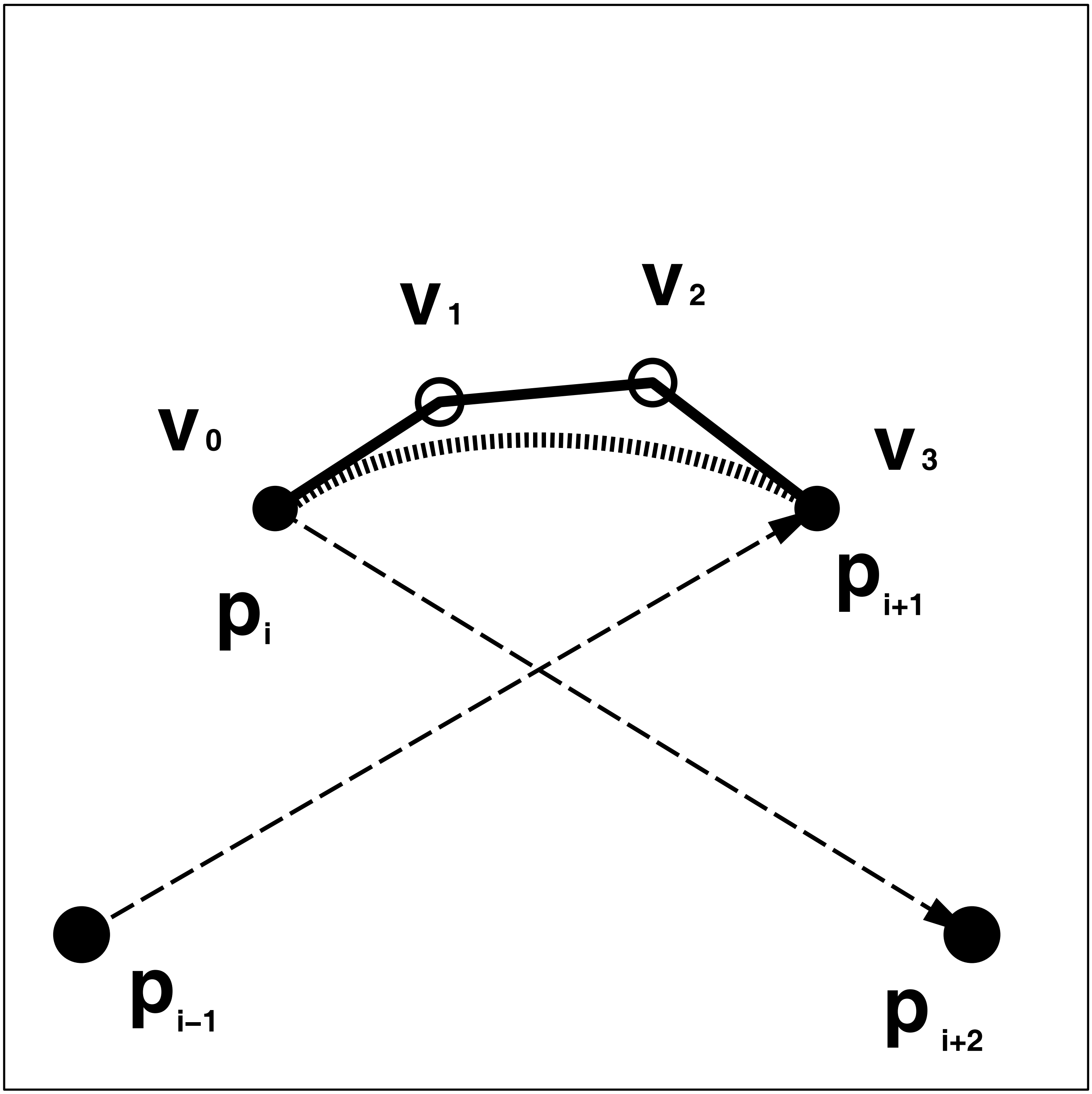
- **In matrix form**

$$\begin{bmatrix} \mathbf{v}_0 \\ \mathbf{v}_1 \\ \mathbf{v}_2 \\ \mathbf{v}_3 \end{bmatrix} = \frac{1}{6} \begin{bmatrix} 0 & 6 & 0 & 0 \\ -1 & 6 & 1 & 0 \\ 0 & 1 & 6 & -1 \\ 0 & 0 & 6 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{p}_{i-1} \\ \mathbf{p}_i \\ \mathbf{p}_{i+1} \\ \mathbf{p}_{i+2} \end{bmatrix}$$

- **Problem: boundary conditions**

- **Properties: C^1, interpolation, local control, non-convex-hull**

Catmull-Rom Splines



B-Splines

- **Mathematics**

$$\mathbf{c}(u) = \sum_{i=0}^{n} \mathbf{p}_i B_{i,k}(u)$$

- **Control points:** $\mathbf{p}_i$
- **Basis functions of degree** $k-1$**:** $B_{i,k}(u)$
- **Piecewise polynomials**
- **How to formulate** $B_{i,k}(u)$
- **First we introduce a knot sequence (n+k+1) in a non-decreasing order**

$$u_0, u_1, u_2, \ldots, u_{n+k}$$

- **The basis function are defined recursively**

$$B_{i,1}(u) = \begin{cases} 1 & u_i <= u < u_{i+1} \\ 0 & otherwise \end{cases}$$

$$B_{i,k}(u) = \frac{u - u_i}{u_{i+k-1} - u_i} B_{i,k-1}(u) + \frac{u_{i+k} - u}{u_{i+k} - u_{i+1}} B_{i+1,k-1}(u)$$

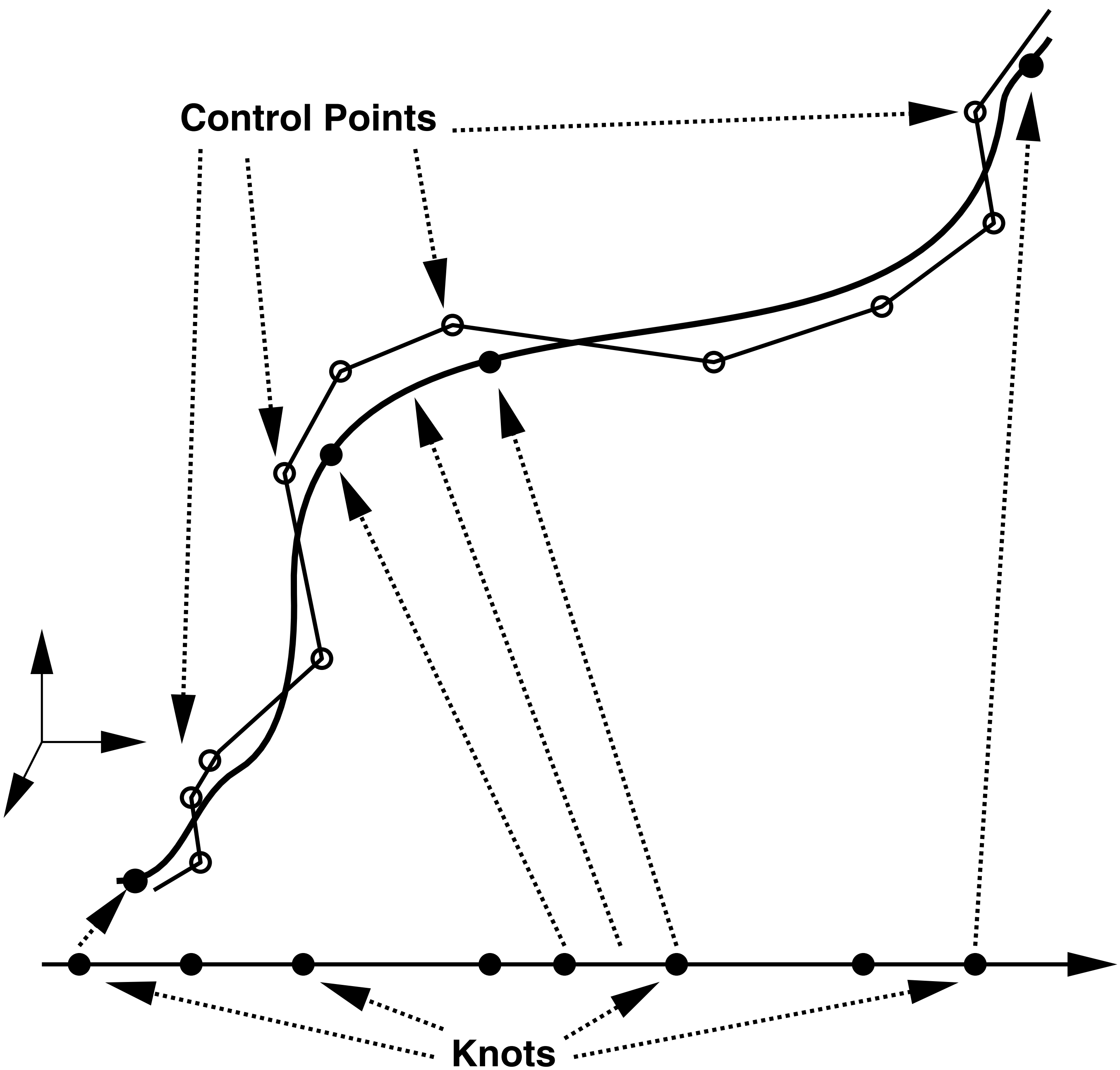
- **The parametric domain of B-spline is:**

$$u \ in [u_{k-1}, u_{n+1}]$$

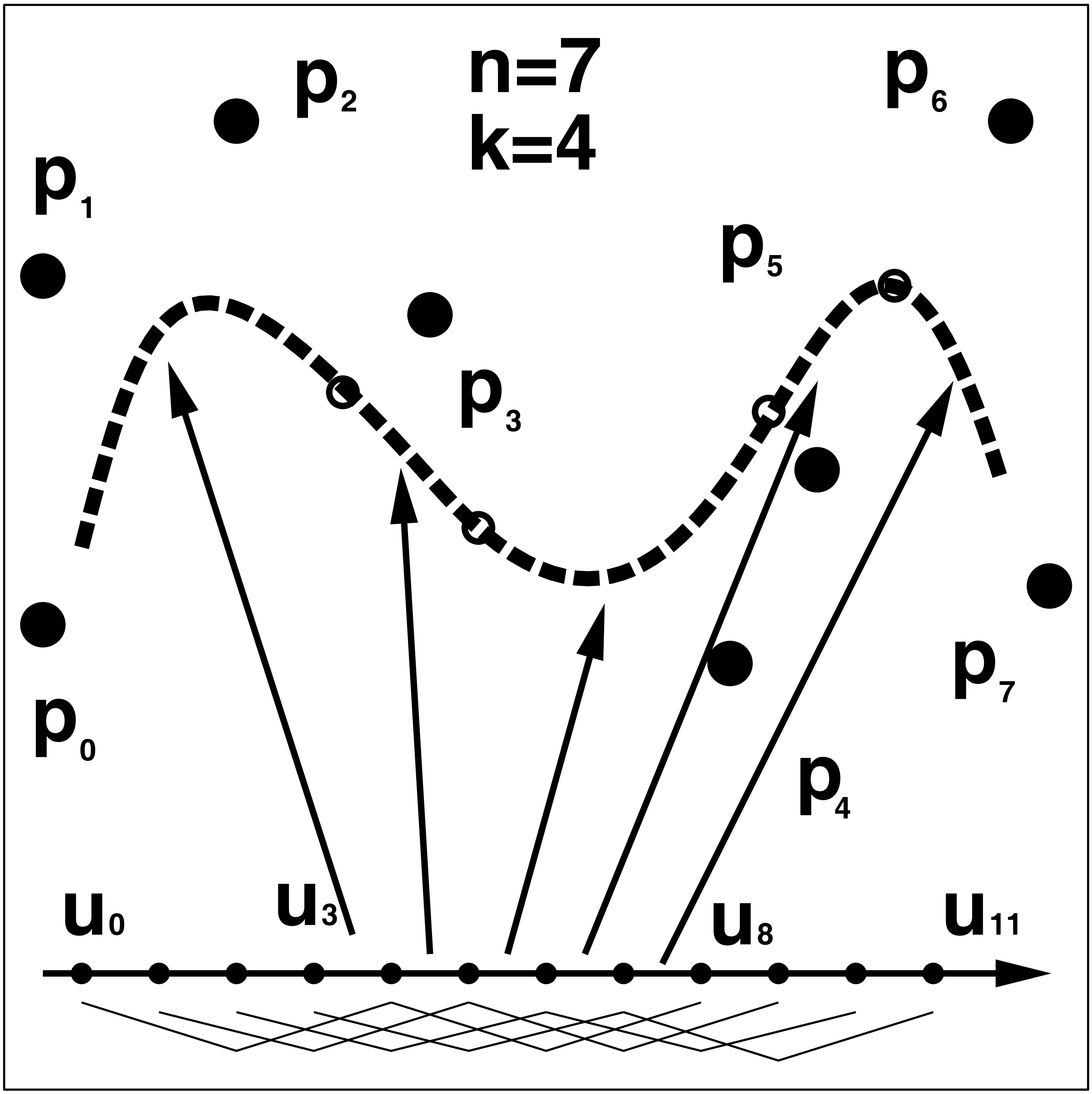
B-spline Curves

- **Curve:** $\mathbf{c}(u) = \sum_{i=0}^{n} \mathbf{p}_i B_{i,k}(u)$
- **Parametric variable:** u
- $n+1$ **control points:** $\mathbf{p}_i$
- $n+k+1$ **non-decreasing knots:** $u_0 \leq u_1 \leq \ldots \leq u_{n+k}$
- **Basis functions of degree** $k-1$**:** $B_{i,k}(u)$
- **Parametric domain:** $u_{k-1} \leq u \leq u_{n+1}$

Piecewise B-splines



B-spline Example



B-Spline Properties

- $\mathbf{c}(u)$: piecewise polynomial of degree $k-1$
- Continuity at joints: C^{k-2}
- $n+1$ control points
- $n+1$ basis (blending) functions
- $B_{i,k}(u)$ is defined over $[u_i, u_{i+k}]$, which includes $k+1$ knots and k sub-intervals
- $n+k+1$ knots
- Knot sequence divide the parametric axis into $n+k$ sub-intervals
- Parametric domain: $[u_{k-1}, u_{n+1}]$
- There are $(n+1)-(k-1)=n-k+2$ sub-intervals wit the parametric domain

- **There are $n-k+2$ piecewise polynomials**
- **Each curve span is influenced by k control points**
- **Each control points at most affects k curve spans**
- **Local control!!!**
- **Convex hull**
- **The degree of B-spline polynomial can be independen[t] from the the number of control point**
- **Compare B-spline with Bezier!!!**
- **Key components: control points, basis functions, ord[er], knots, parametric domain, local vs. global control, continuity**

Basis Functions

- **Linear examples**

$$B_{0,2}(u) = \begin{cases} u & u \in [0,1] \\ 2-u & u \in [1,2] \end{cases}$$

$$B_{1,2}(u) = \begin{cases} u-1 & u \in [1,2] \\ 3-u & u \in [2,3] \end{cases}$$

$$B_{2,2}(u) = \begin{cases} u-2 & u \in [2,3] \\ 4-u & u \in [3,4] \end{cases}$$

- **How does it look like???**

- **Quadric examples**

$$B_{0,3}(u) = \begin{cases} \frac{1}{2}u^2 & u \in [0,1] \\ \frac{1}{2}u(2-u) + \frac{1}{2}(3-u)(u-1) & u \in [1,2] \\ \frac{1}{2}(3-u)^2 & u \in [2,3] \end{cases}$$

$$B_{1,3}(u) = \begin{cases} \frac{1}{2}(u-1)^2 & u \in [1,2] \\ \frac{1}{2}(u-1)(3-u) + \frac{1}{2}(4-u)(u-2) & u \in [2,3] \\ \frac{1}{2}(4-u)^2 & u \in [3,4] \end{cases}$$

- **How does it look like???**

- **Cubic example**

$$B_{0,4}(u) = \begin{cases} \frac{1}{6}u^3 \\ \frac{1}{6}(u^2(2-u) + u(3-u)(u-1) + (4-u)(u-1)^2 \\ \frac{1}{6}(u(3-u)^2 + (4-u)(u-1)(3-u) + (4-u)^2(\\ \frac{1}{6}(4-u)^3 \end{cases}$$

B-spline Basis Functions

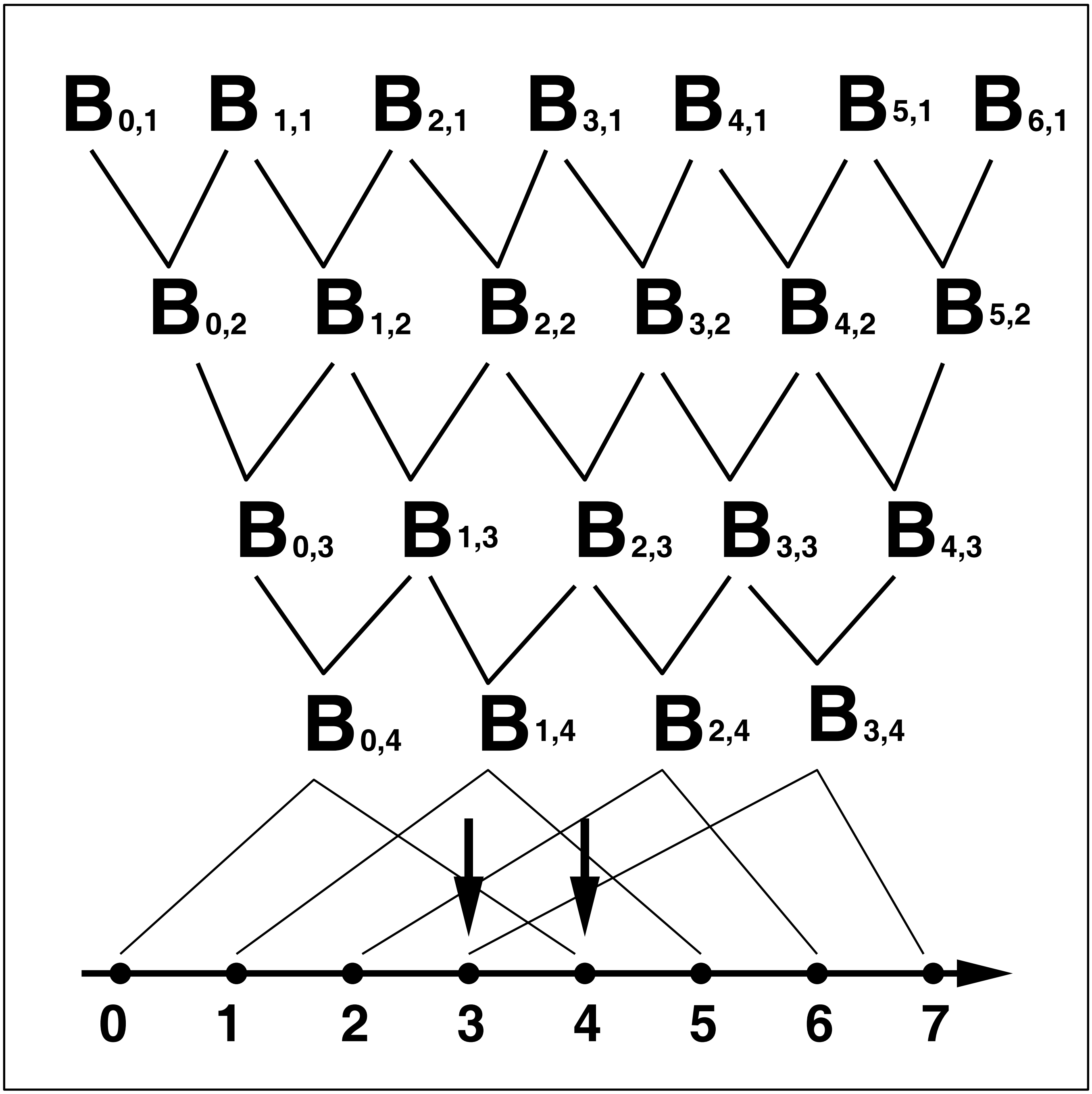
- **One example: quadratic**
- **Knot vector:** $[0, 1, 2, 3, 4, 5, 6]$
- $$B_{0,3}(u) = \begin{cases} \frac{1}{2}u^2, & \textbf{for } 0 \leq u < 1 \\ \frac{1}{2}u(2-u) + \frac{1}{2}(u-1)(3-u), & \textbf{for } 1 \leq u < 2 \\ \frac{1}{2}(3-u)^2 & \textbf{for } 2 \leq u < 3 \end{cases}$$
- $$B_{1,3}(u) = \begin{cases} \frac{1}{2}(u-1)^2, & \textbf{for } 1 \leq u < \\ \frac{1}{2}(u-1)(3-u) + \frac{1}{2}(u-2)(4-u), & \textbf{for } 2 \leq u < \\ \frac{1}{2}(4-u)^2, & \textbf{for } 3 \leq u < \end{cases}$$
- $B_{2,3}(u) = \ldots$
- $B_{3,3}(u) = \ldots$

B-spline Basis Functions

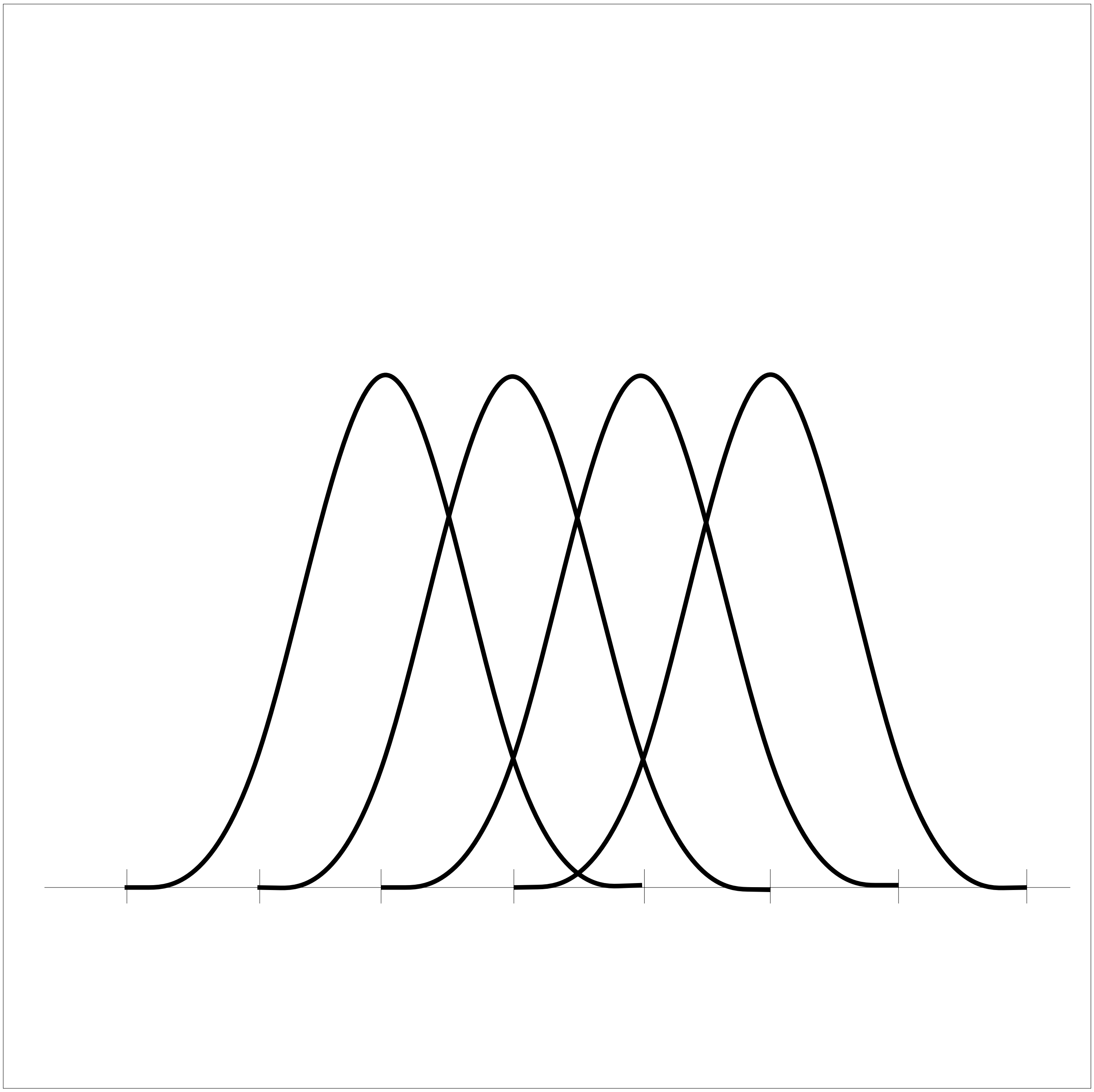
- $B_{i,1}(u) = \begin{cases} 1 & \textbf{for } u_i \leq u < u_{i+1} \\ 0 & \textbf{otherwise} \end{cases}$,

- $B_{i,k}(u) = \frac{u-u_i}{u_{i+k-1}-u_i} B_{i,k-1}(u) + \frac{u_{i+k}-u}{u_{i+k}-u_{i+1}} B_{i+1,k-1}(u).$

Basis Functions



B-Spline Basis Functions



B-spline Properties

- Control points and basis functions: $n + 1$
- Degree of basis functions: $k - 1$
- Partition of unity, positivity, and recursive evaluation basis functions
- The degree is independent of the number of control points
- The curve is $C^{(k-2)}$ continuous in general
- Local control
- Special case: Bezier splines
- Efficient algorithms & tools
  - evaluation
  - knot insertion
  - degree elevation

  - **derivative, integration**
  - **continuity**

- **Composite Bezier curves for B-splines**

Another formulation

- **Uniform B-spline**
- **Matrix representation**
- **Normalization to** $u \in [0, 1]$
- **End-point positions and tangents**

$$\mathbf{c}(0) = \frac{1}{6}(\mathbf{p}_0 + 4\mathbf{p}_1 + \mathbf{p}_2)$$

$$\mathbf{c}(1) = \frac{1}{6}(\mathbf{p}_1 + 4\mathbf{p}_2 + \mathbf{p}_3)$$

$$\mathbf{c}'(0) = \frac{1}{2}(\mathbf{p}_2 - \mathbf{p}_0)$$

$$\mathbf{c}'(1) = \frac{1}{2}(\mathbf{p}_3 - \mathbf{p}_1)$$

$$\mathbf{c}(u) = UM_h \begin{bmatrix} \mathbf{c}(0) \\ \mathbf{c}(1) \\ \mathbf{c}'(0) \\ \mathbf{c}'(1) \end{bmatrix}$$

$$= U M_h M' \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{bmatrix}$$

$$= U M \mathbf{p}$$

- **Basis matrix**

$$M = \frac{1}{6} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix}$$

- **Basis functions (**$u \in [0, 1]$**):**

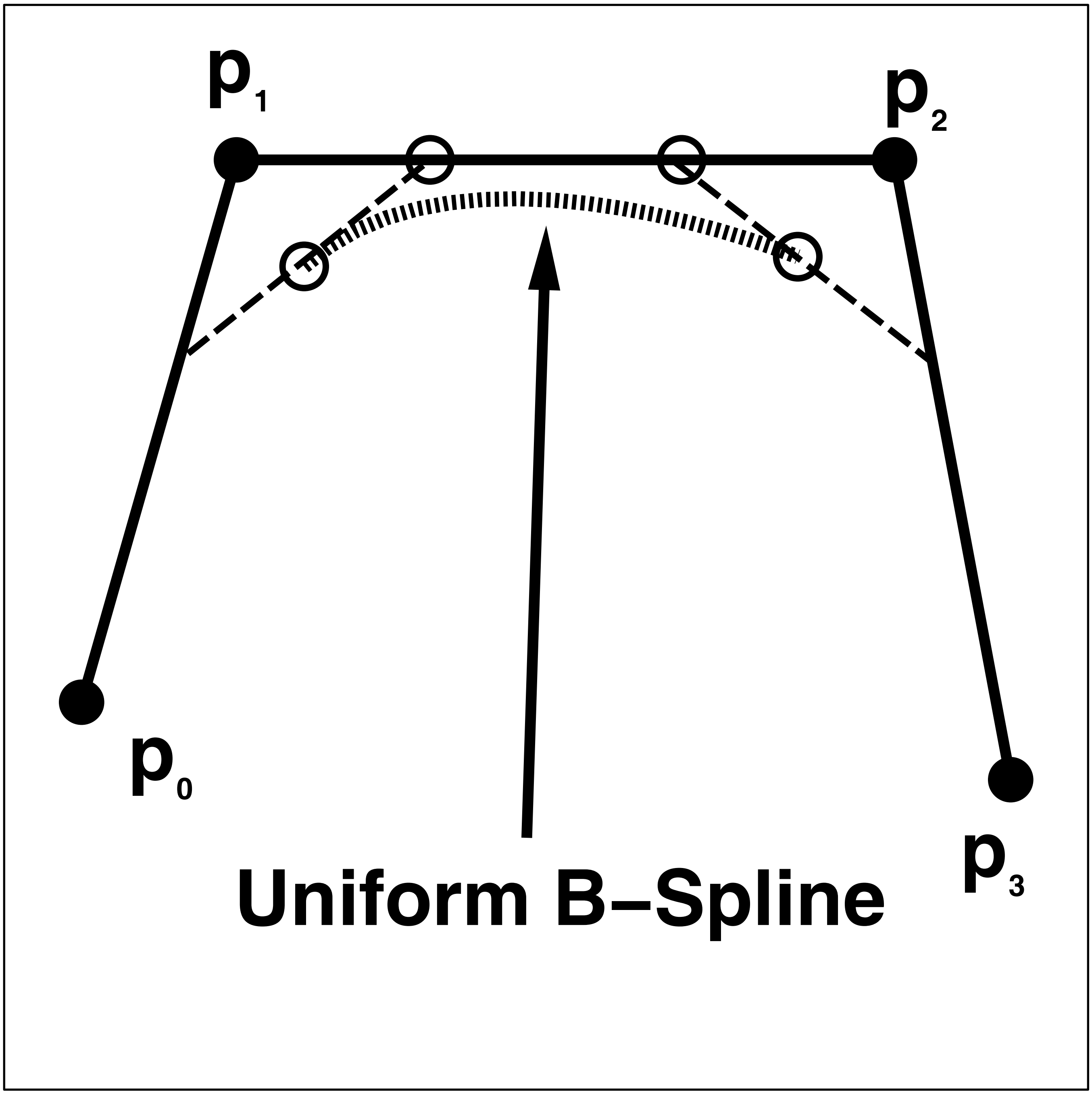
$$B_{0,4}(u) = \frac{1}{6}(1 - u)^3$$

$$B_{1,4}(u) = \frac{1}{6}(3u^3 - 6u^2 + 4)$$

$$B_{2,4}(u) = \frac{1}{6}(-3u^3 + 3u^2 + 3u + 1)$$

$$B_{3,4}(u) = \frac{1}{6}u^3$$

Uniform B-Spline



B-Spline Rendering

- Transform it to a set of Bezier curves
- Display B-Spline ($\{\mathrm{p}_0, \ldots, \mathrm{p}_n\}$)
- For $i = 0$ to $n - k + 1$ do
- Convert i-span controlled by $\mathrm{p}_i, \ldots, \mathrm{p}_{i+k-1}$ into a bezie curve represented by $\mathrm{v}_0, \ldots, \mathrm{v}_{k-1}$
- Display Bezier Curve ($\{\mathrm{v}_0, \ldots, \mathrm{v}_{k-1}\}$)

From B-splines to NURBS

- What are NURBS???
- Non Uniform Rational B-Splines (NURBS)
- Rational curve motivation
- Polynomial-based splines can not represent commonly-used analytic shapes such as conic section circles, ellipses, parabolas
- Rational splines can achieve this goal
- NURBS are a unified representation
  - polynomial, conic section, etc
  - industry standard
- NURBS Mathematics

$$\mathbf{c}(u) = \frac{\sum_{i=0}^{n} \mathbf{p}_i w_i B_{i,k}(u)}{\sum_{i=0}^{n} w_i B_{i,k}(u)}$$

- **Geometric Meaning — Projection!**

- **B-splines**

$$\mathbf{c}(u) = \sum_{i=0}^{n} \begin{bmatrix} \mathbf{p}_{i,x} w_i \\ \mathbf{p}_{i,y} w_i \\ \mathbf{p}_{i,z} w_i \\ w_i \end{bmatrix} B_{i,k}(u)$$

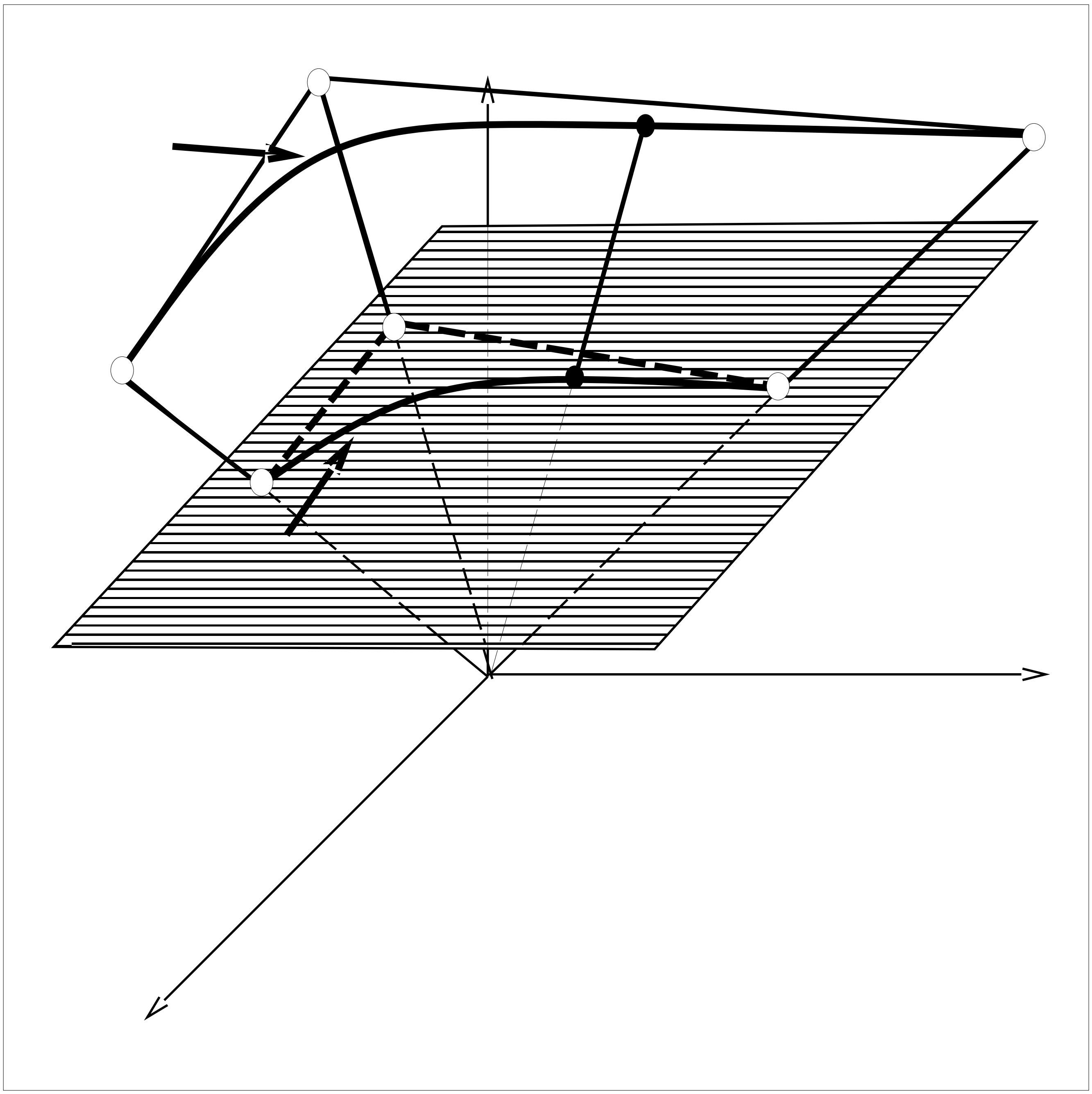
- **Homogeneous representation**

$$\begin{bmatrix} x(u) \\ y(u) \\ z(u) \\ w(u) \end{bmatrix} = \sum_{i=0}^{n} \begin{bmatrix} \mathbf{p}_i w_i \\ \mathbf{w}_i \end{bmatrix} B_{i,k}(u)$$

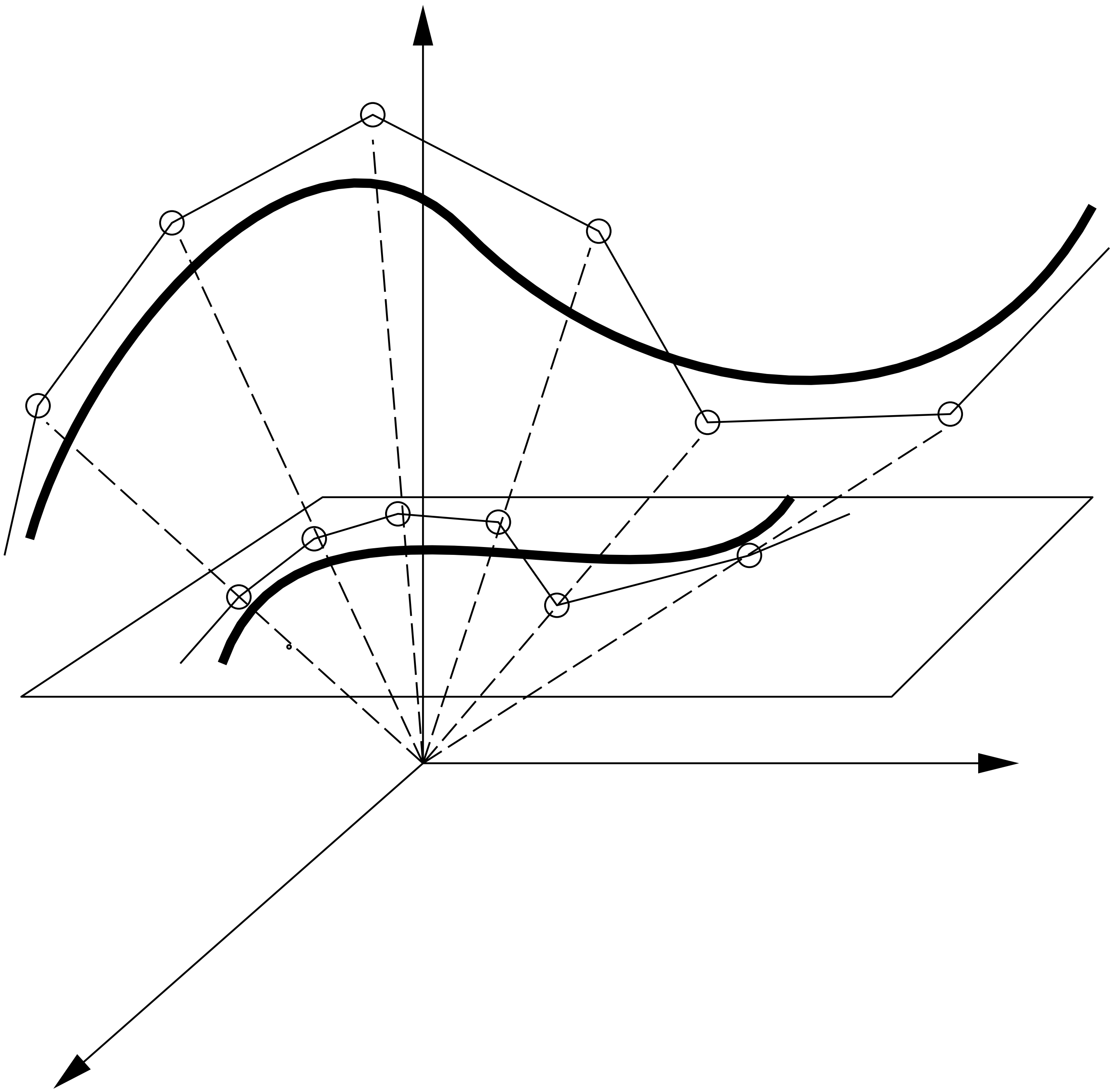
- **NURBS is obtained from projection**

- **Properties of NURBS: represent standard shapes, invariant under perspective projection B-spline is a special case, weights as extra degrees of freedom, common analytic shapes such as circles, clear geomet meaning of weights**

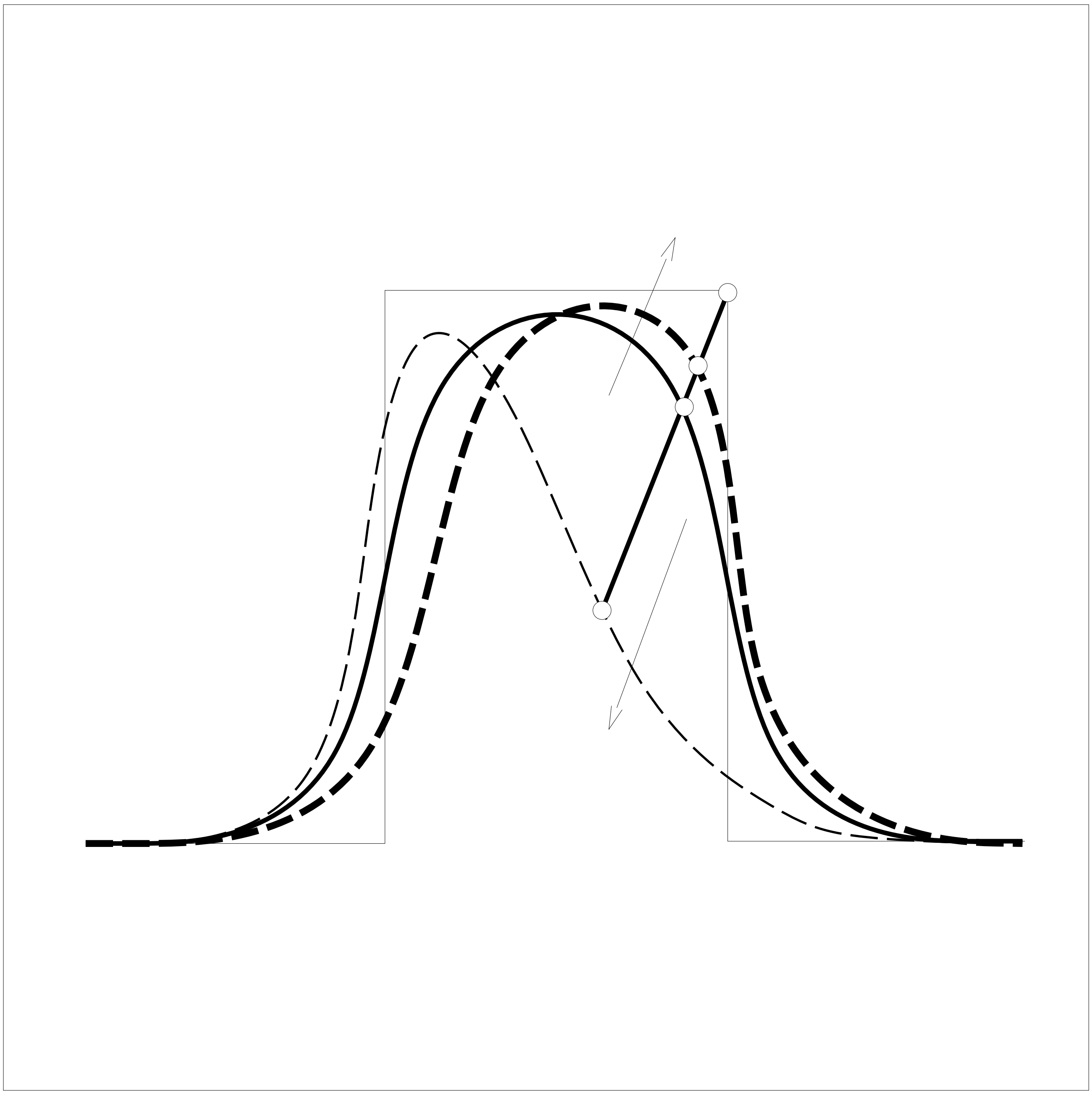
Rational Bezier Curve



From B-splines to NURBS



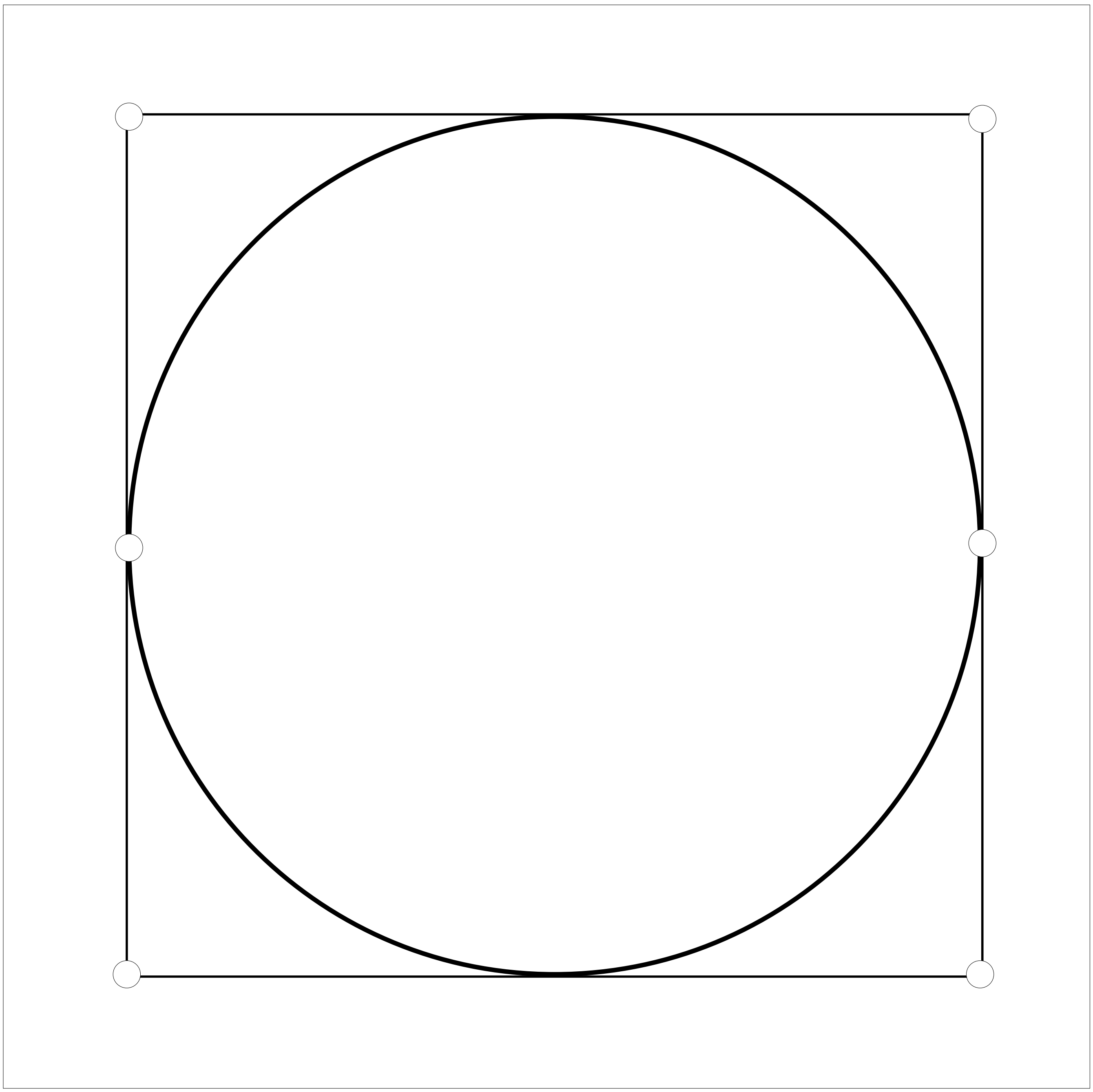
NURBS Weights



NURBS Properties

- Generalization of B-splines and Bezier splines
- Unified formulation for free-form and analytic shapes
- Weights as extra DOFs
- Various smoothness requirements
- Powerful geometric toolkits
- Efficient and fast evaluation algorithm
- Invariance under standard transformations
- Composite curves
- Continuity conditions

NURBS Circle



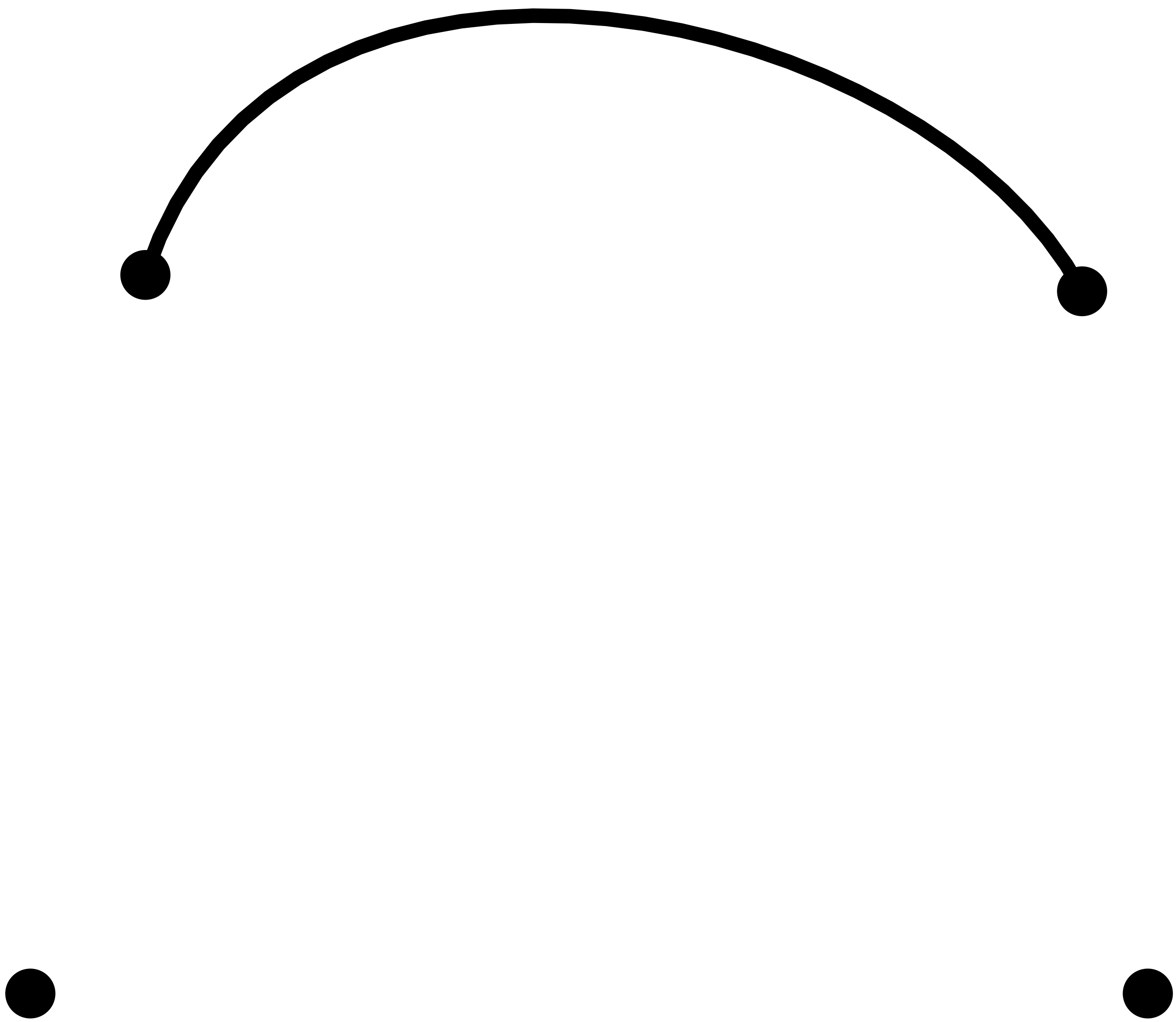
NURBS for Analytic Shapes

- **Conic Sections**
- **Natural quadrics**
- **Extruded surfaces**
- **Ruled surfaces**
- **Surfaces of revolution**

Cardinal Splines

- **Four vertices:** $\mathbf{v}_0$, $\mathbf{v}_1$, $\mathbf{v}_2$, **and** $\mathbf{v}_3$,
- $\mathbf{c}(0) = \mathbf{v}_1$, $\mathbf{c}(1) = \mathbf{v}_2$,
- $\mathbf{c}^{(1)}(0) = \frac{1}{2}(1-\alpha)(\mathbf{v}_2 - \mathbf{v}_0)$,
- $\mathbf{c}^{(1)}(1) = \frac{1}{2}(1-\alpha)(\mathbf{v}_3 - \mathbf{v}_1)$
- **Special case: Catmull-Rom splines when** $\alpha = 0$
- **More general case: Kochanek-Bartels splines**
  - **tension, bias, continuity parameters**

Cardinal Splines



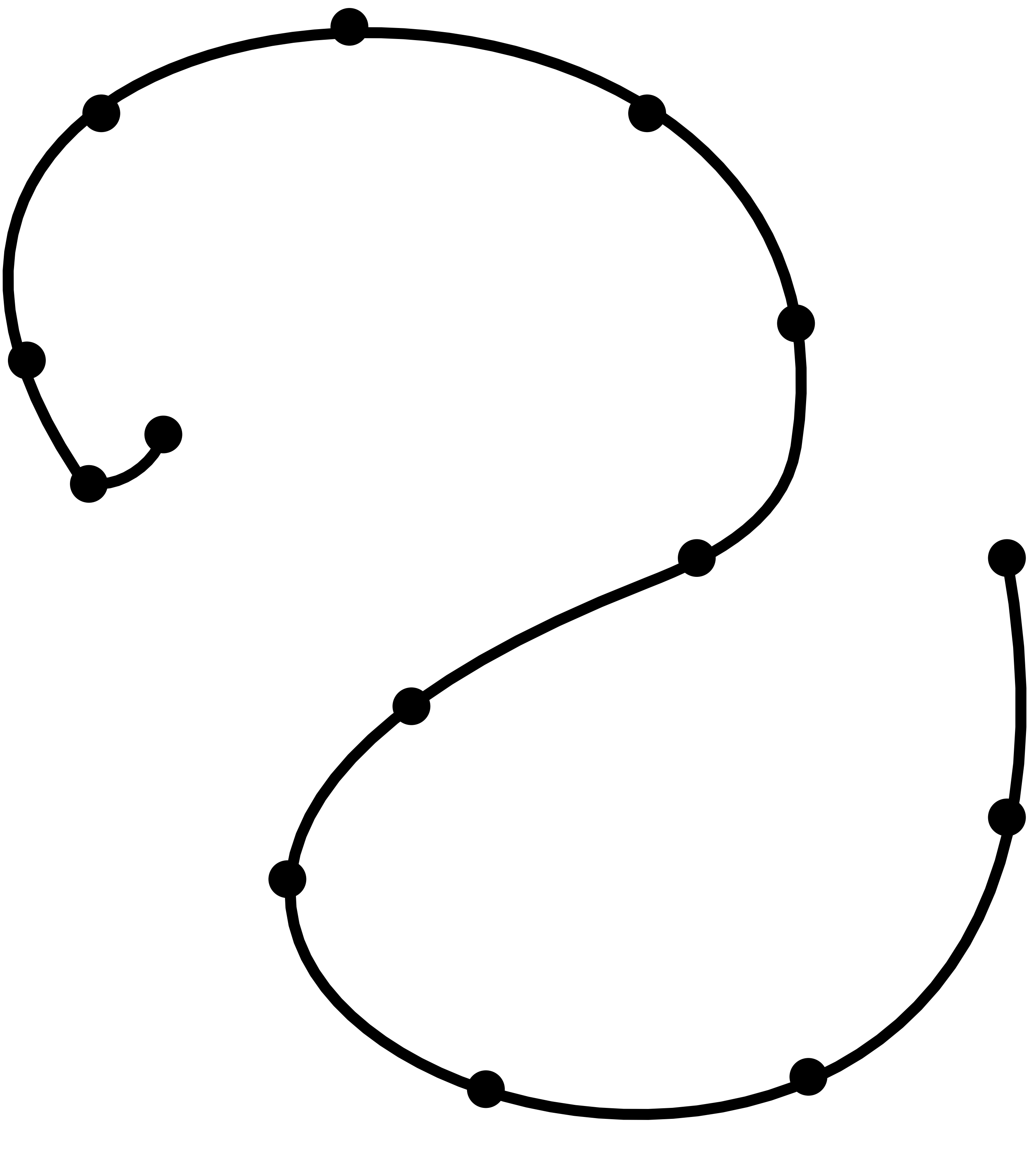
Kochanek-Bartels Splines

- **Four vertices:** $\mathbf{v}_0$, $\mathbf{v}_1$, $\mathbf{v}_2$, **and** $\mathbf{v}_3$
- $\mathbf{c}(0) = \mathbf{v}_1$, $\mathbf{c}(1) = \mathbf{v}_2$,
- $\mathbf{c}^{(1)}(0) =$
$\frac{1}{2}(1-\alpha)[(1+\beta)(1-\gamma)(\mathbf{v}_1 - \mathbf{v}_0) + (1-\beta)(1+\gamma)(\mathbf{v}_2 - \mathbf{v}_1)]$
- $\mathbf{c}^{(1)}(1) =$
$\frac{1}{2}(1-\alpha)[(1+\beta)(1+\gamma)(\mathbf{v}_2 - \mathbf{v}_1) + (1-\beta)(1-\gamma)(\mathbf{v}_3 - \mathbf{v}_2)]$
- **Tension parameter:** α
- **Bias parameter:** β
- **Continuity parameter:** γ

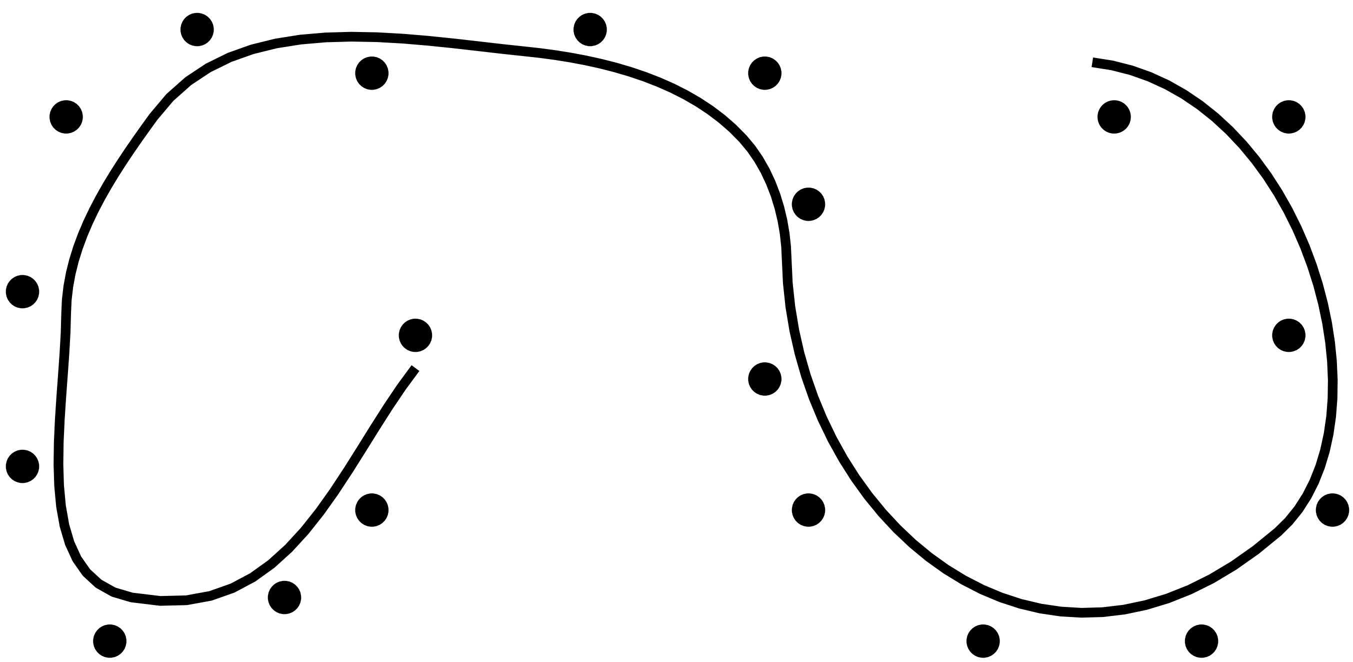
Summary

- Parametric curves and surfaces
- Polynomials and rational polynomials
- Free-form curves and surfaces
- Other commonly used geometric primitives sphere, ellipsoid, torus, superquadrics, blobby
- Motivation: fewer degrees of freedom more geometri coverage

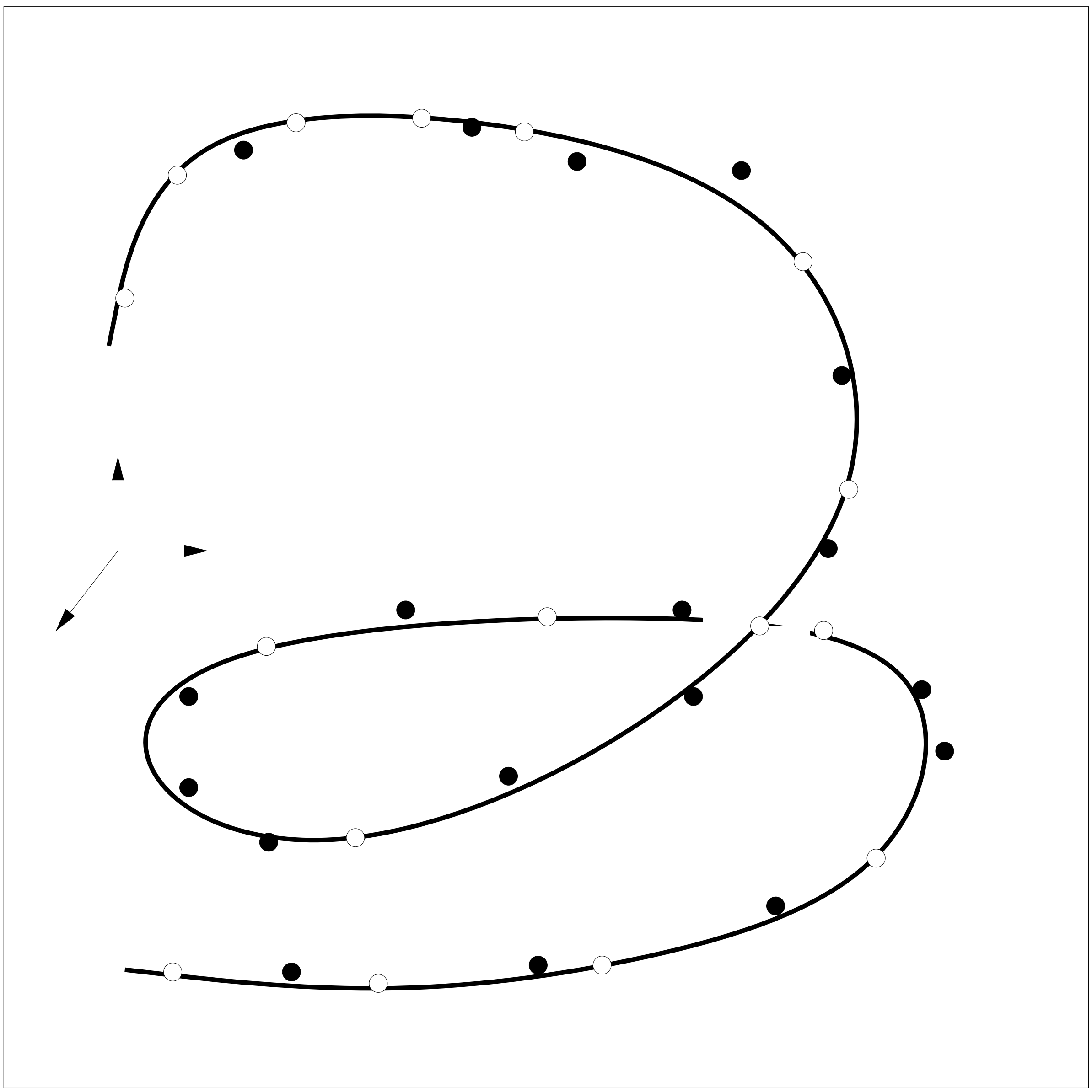
Interpolation Splines



Approximation Splines



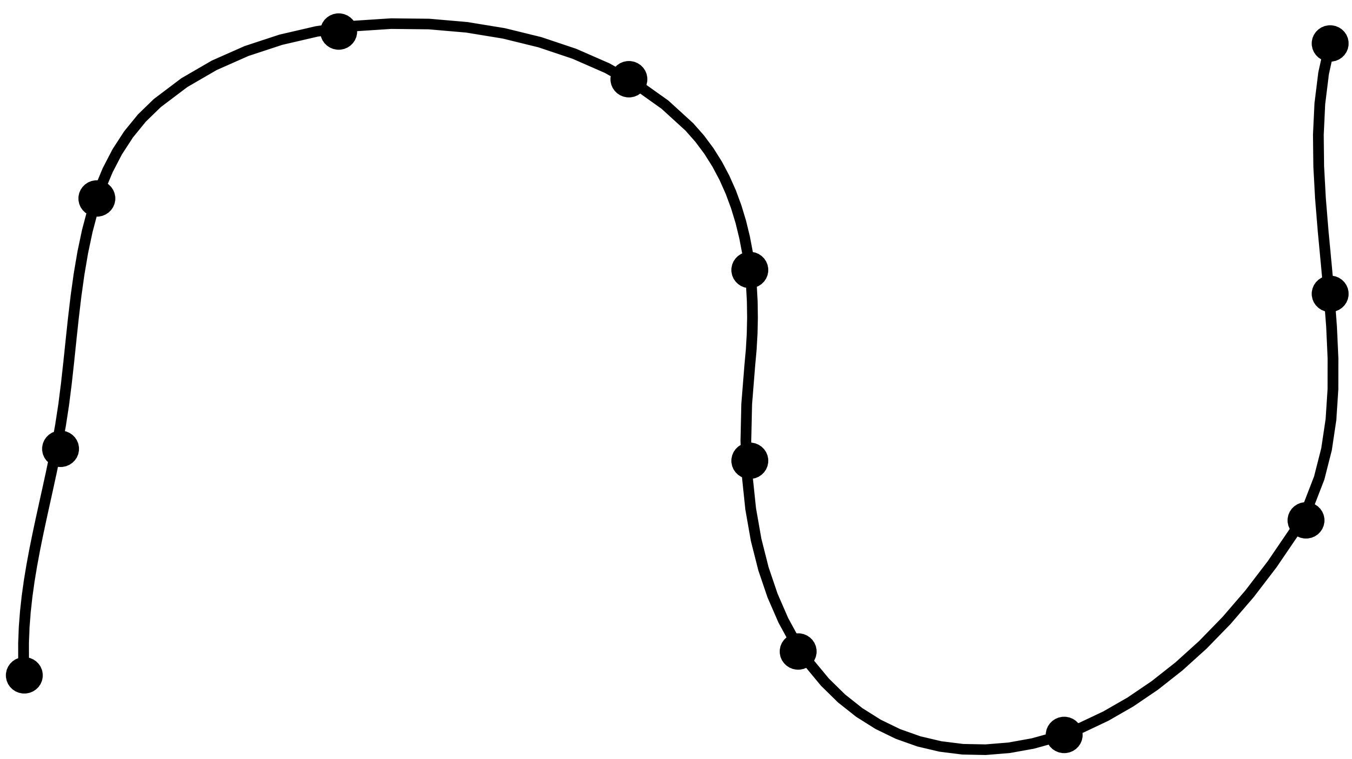
Interpolation/Approximation



Natural C^2 Cubic Splines

- **A set of piecewise cubic polynomials**
- C^2 **continuity at each vertex**
- $\mathbf{c}_i(u) = \begin{bmatrix} x(u) \\ y(u) \\ z(u) \end{bmatrix}$

Natural C^2 Cubic Splines



From Curve to Surface