Lecture 8:

Curves and Surfaces

Computer Graphics 2025

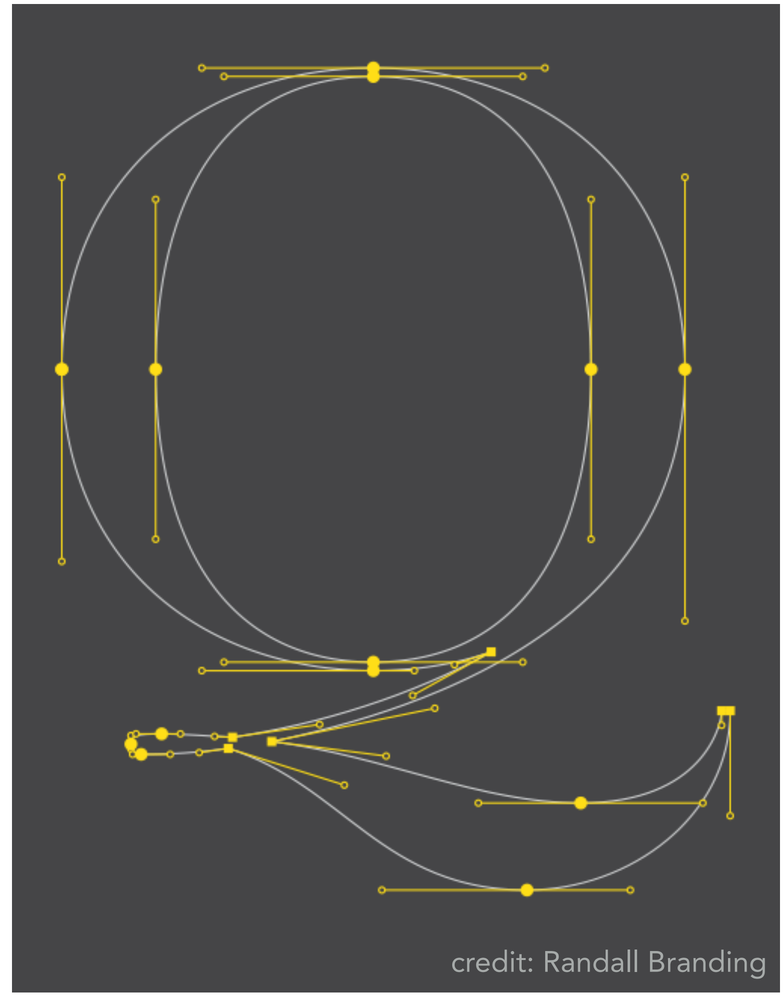
Fuzhou University - Computer Science

Curves

Vector Fonts

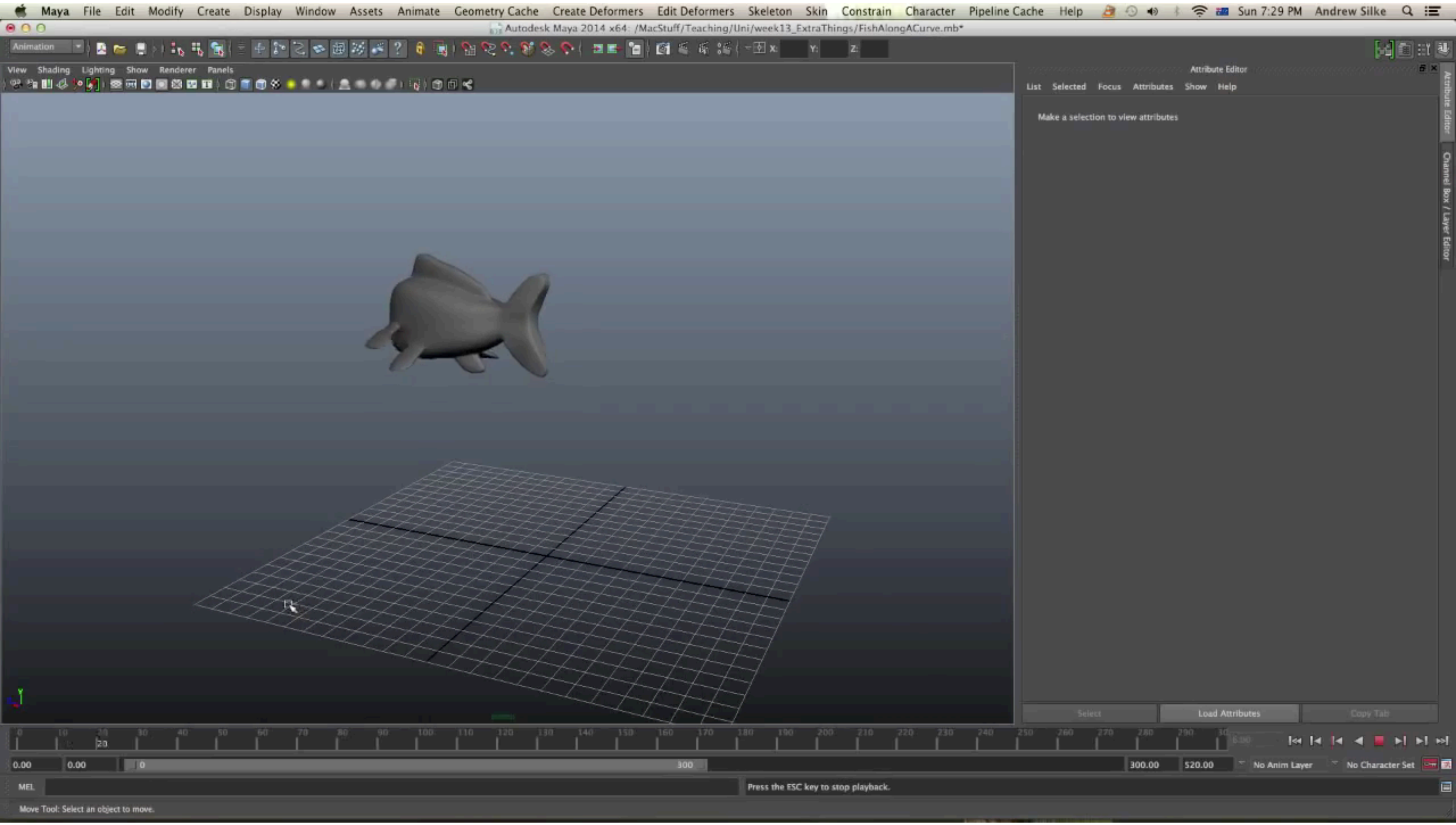
The Quick Brown
Fox Jumps Over
The Lazy Dog

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz 0123456789



Baskerville font - represented as cubic Bézier splines

Animation Curves



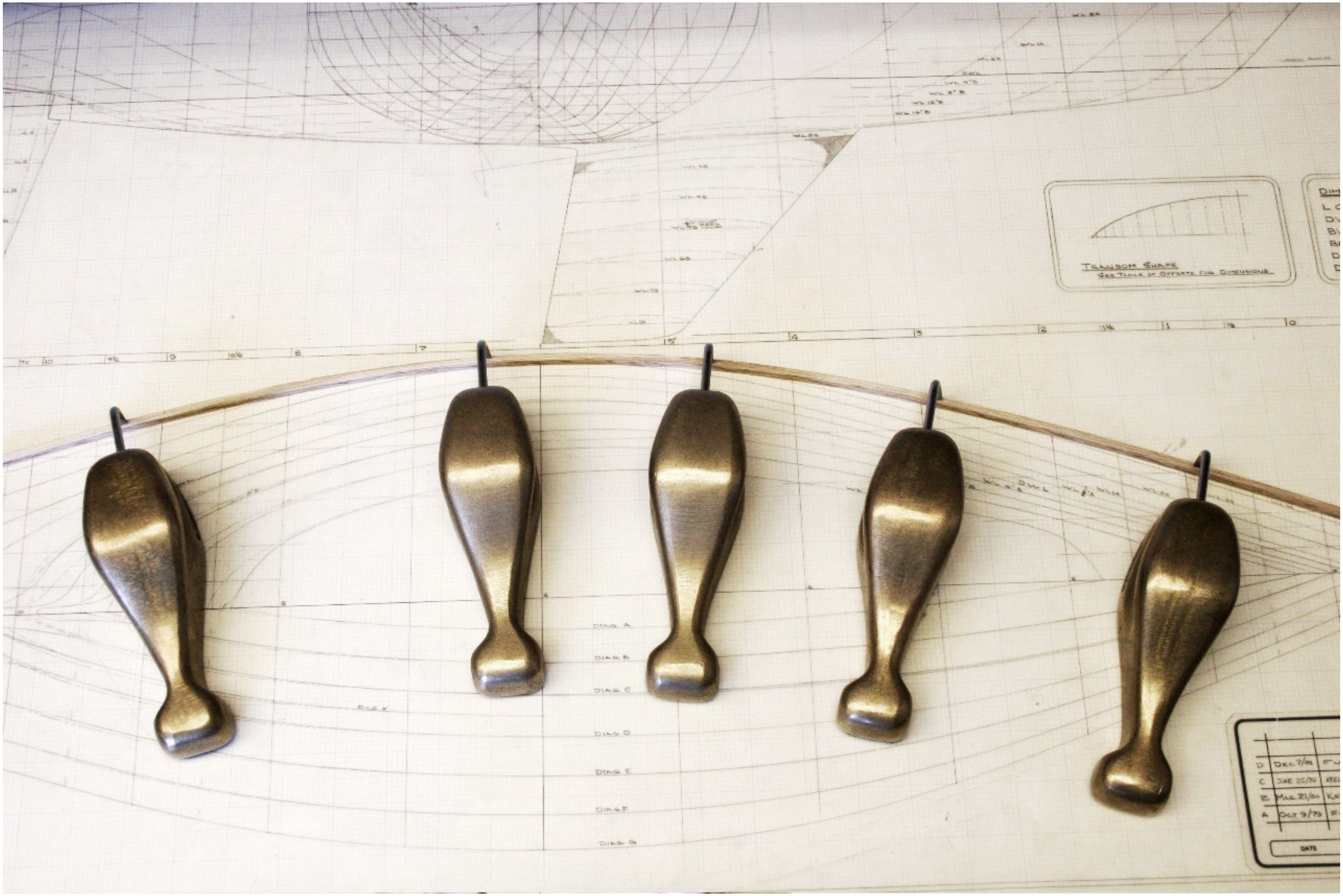
Maya Animation Tutorial: https://youtu.be/b-o5wtZlJPc

CAD Design

3D Car Modeling with Rhinoceros

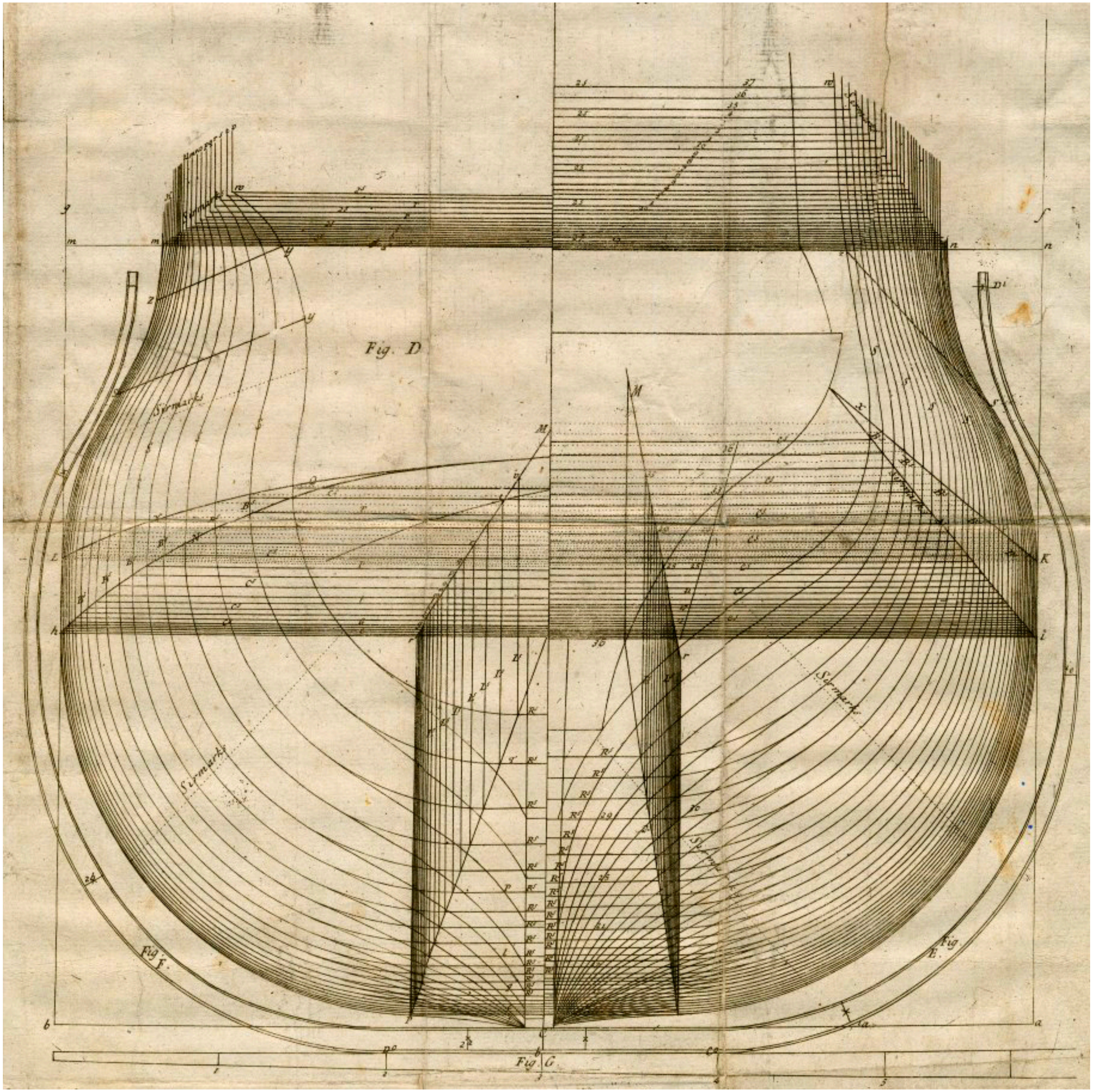
Splines 样条

A Real Draftsperson’s Spline



http://www.alatown.com/spline-history-architecture/

A ship comprised of splines



A ship’s body plan comprised of splines

Spline Topics

Defined by a set of points

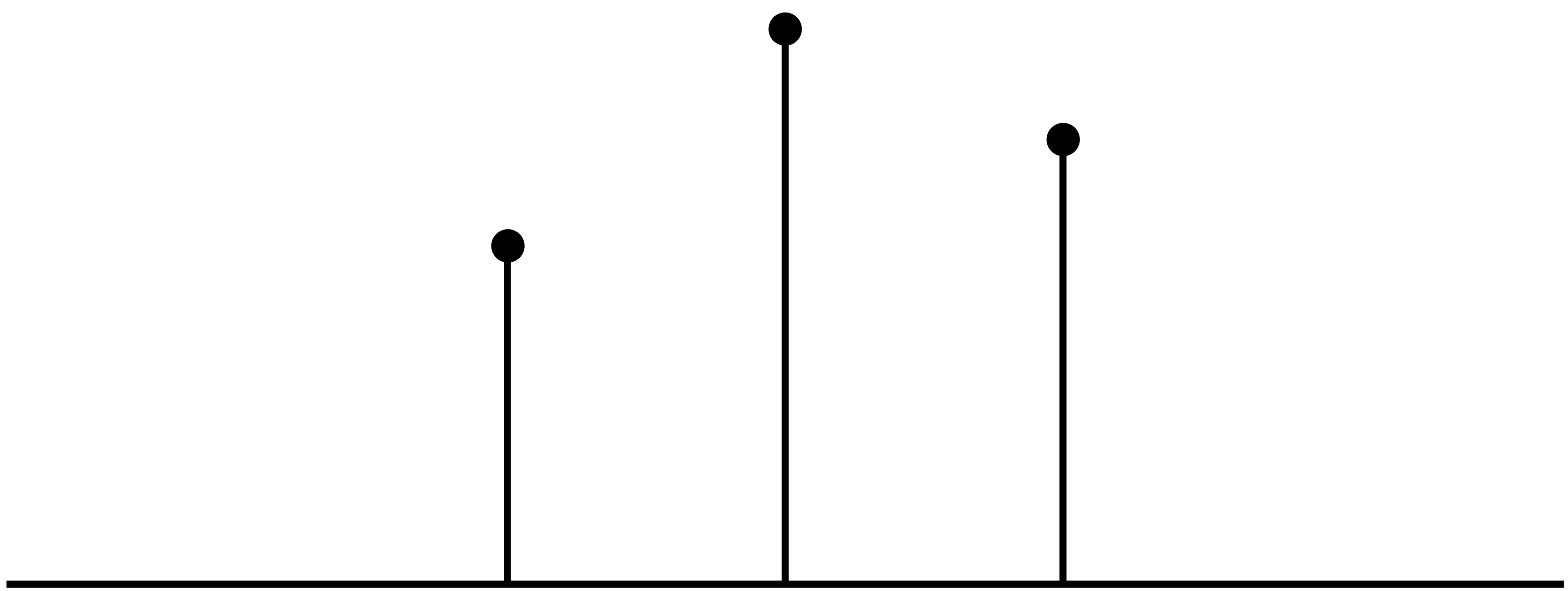
Interpolation between points

- Cubic Hermite interpolation
- Catmull-Rom interpolation

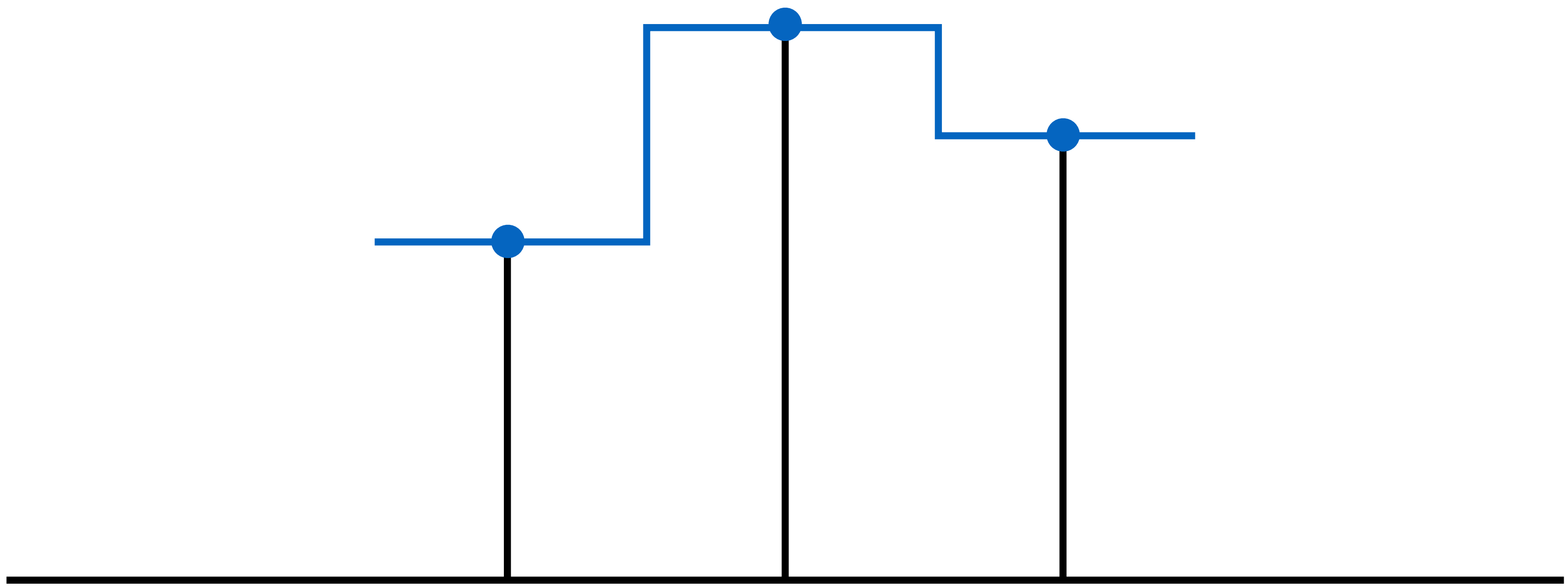
Bezier curves

Bezier surfaces

Goal: Interpolate Values

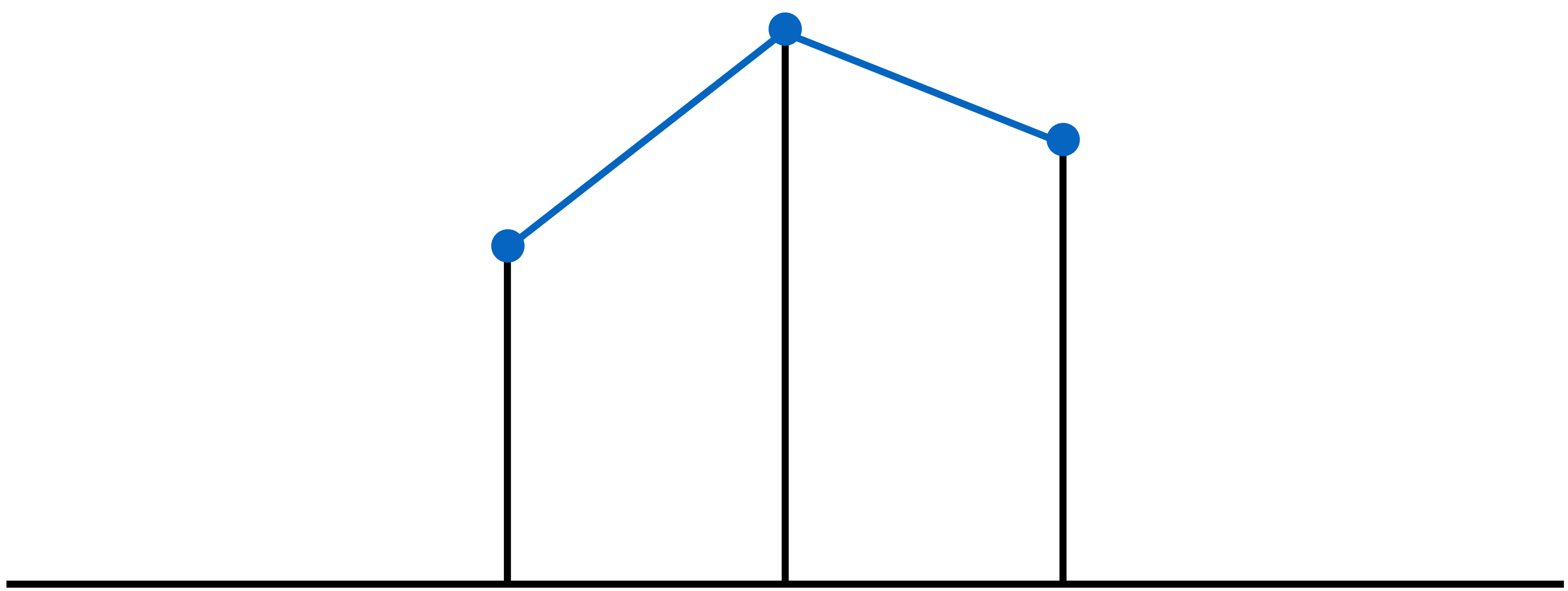


Nearest Neighbor Interpolation



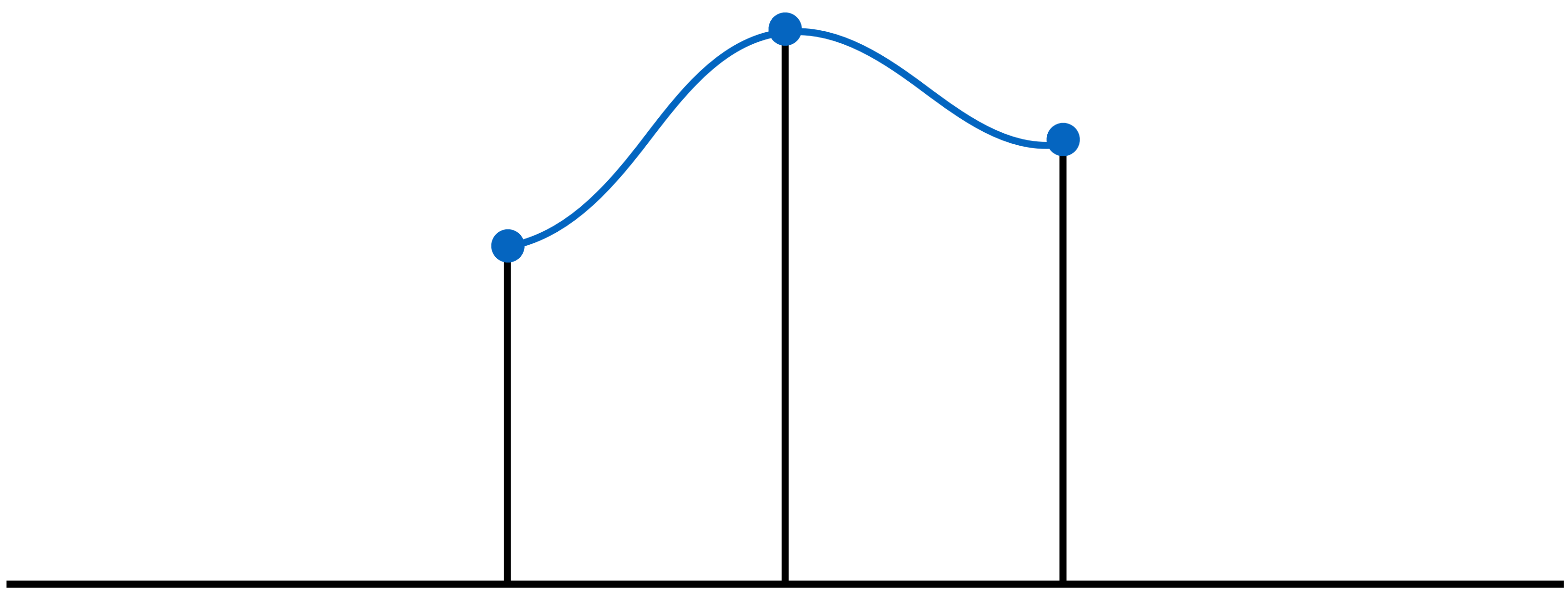
Problem: values not continuous

Linear Interpolation



Problem: derivatives not continuous

Smooth Interpolation?

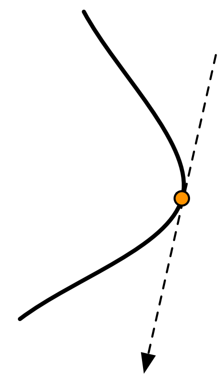


Cubic Hermite Interpolation

Some Differential Geometry

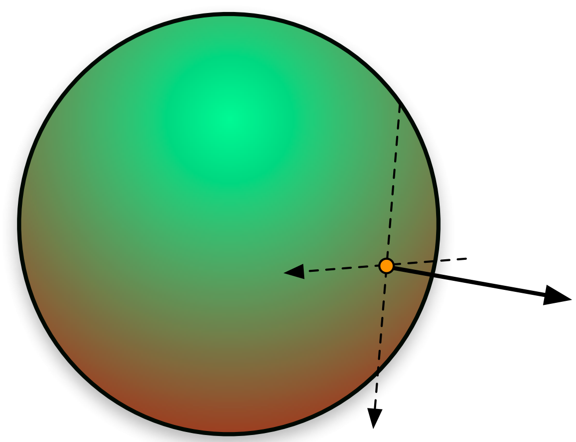
Tangent to curve

$$\boldsymbol{t}(u) = \left.\frac{\partial \boldsymbol{x}}{\partial u}\right|_u$$



Tangents to surface

$$\boldsymbol{t}_u(u,v) = \left.\frac{\partial \boldsymbol{x}}{\partial u}\right|_{u,v} \qquad \boldsymbol{t}_v(u,v) = \left.\frac{\partial \boldsymbol{x}}{\partial v}\right|_{u,v}$$

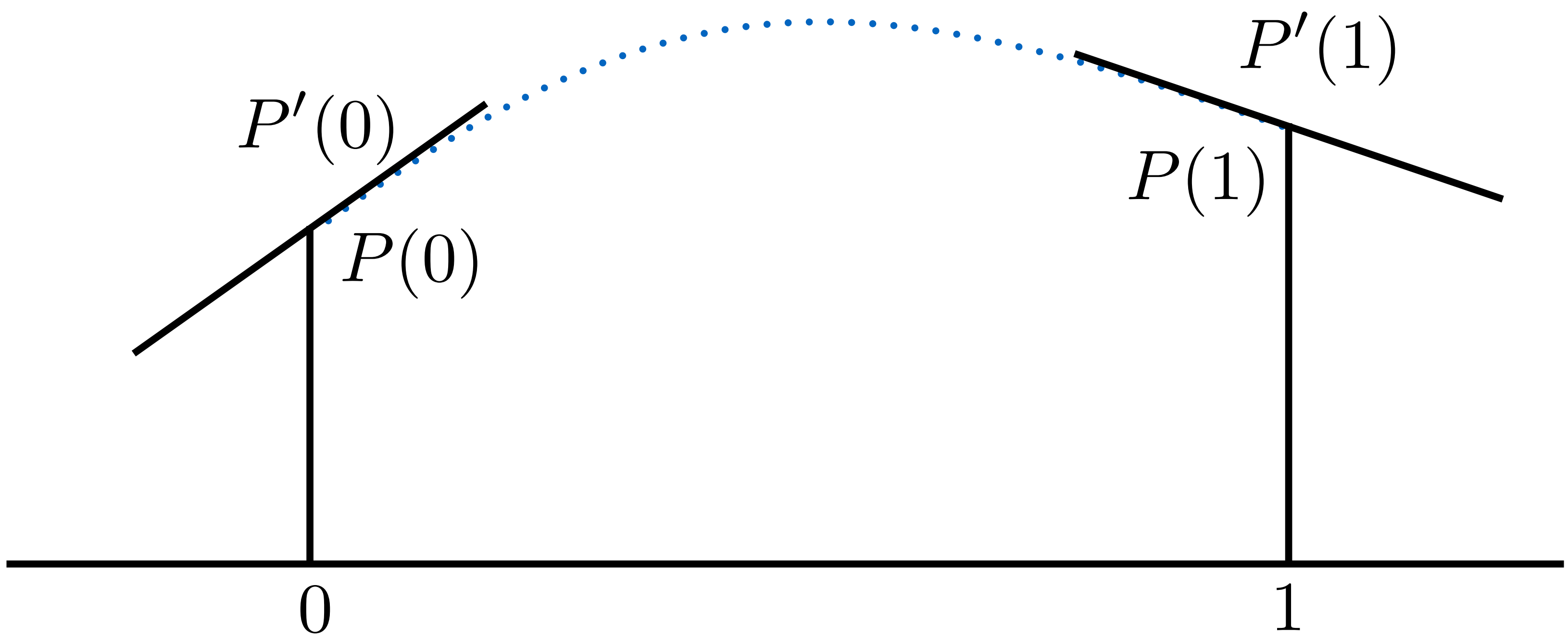


Normal of surface

$$\hat{\boldsymbol{n}} = \frac{\boldsymbol{t}_u \times \boldsymbol{t}_v}{||\boldsymbol{t}_u \times \boldsymbol{t}_v||}$$

Degeneracies: $\partial \boldsymbol{x}/\partial u = 0$ **or** $\boldsymbol{t}_u \times \boldsymbol{t}_v = 0$

Cubic Hermite Interpolation



Inputs: values and derivatives at endpoints

Polynomial Basis Functions

Power Basis

$$x(u) = \sum_{i=0}^{d} c_i \, u^i$$

e.g. a cubic polynomial

$$x(u) = c_0 + c_1 u + c_2 u^2 + c_3 u^3$$

$$x(u) = \boldsymbol{C} \cdot \boldsymbol{\mathcal{P}}^d$$

$$\boldsymbol{C} = [c_0, c_1, c_2, \ldots, c_d]$$

$$\boldsymbol{\mathcal{P}}^d = [1, u, u^2, \ldots, u^d]$$

The elements of $\boldsymbol{\mathcal{P}}^d$ are ***linearly independent***

i.e. no good approximation $u^k \not\approx \sum_{i \neq k} c_i \, u^i$

Specifying a Curve

Given desired values (constraints) how do we determine the coefficients for cubic power basis?

For now, assume

$$u_0 = 0 \quad u_1 = 1$$

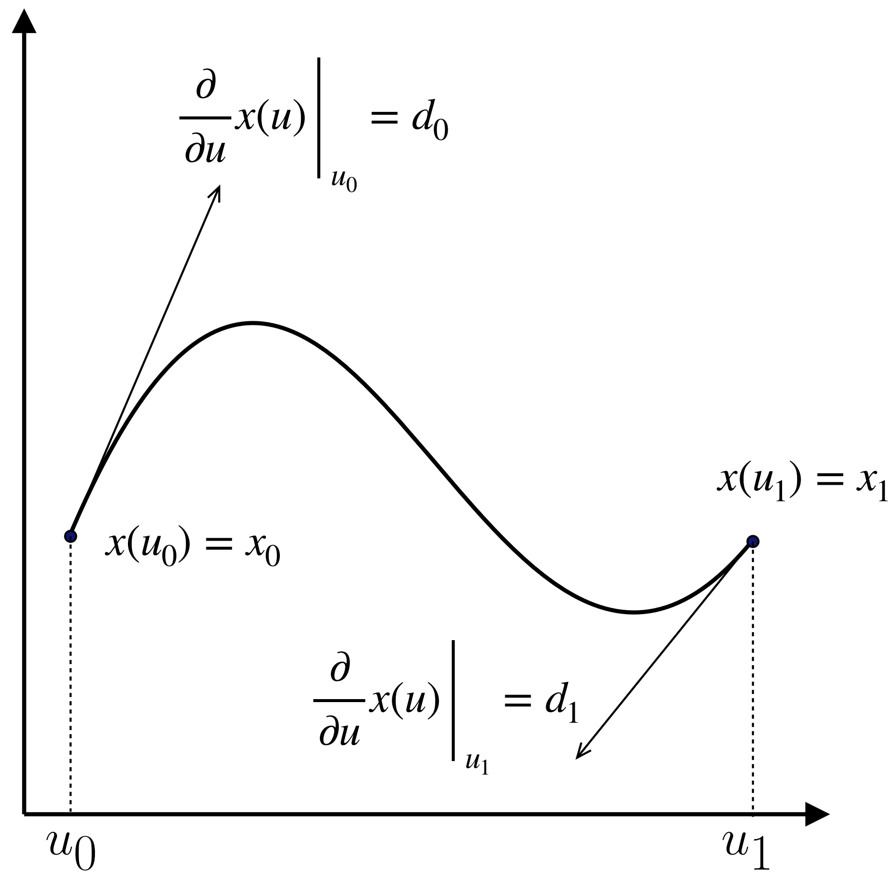
$$x(u) = c_0 + c_1 u + c_2 u^2 + c_3 u^3$$

$$x(u_0) = x_0$$

$$x(u_1) = x_1$$

$$\left.\frac{\partial}{\partial u} x(u)\right|_{u_0} = d_0$$

$$\left.\frac{\partial}{\partial u} x(u)\right|_{u_1} = d_1$$



Specifying a Curve

Given desired values (constraints) how do we determine the coefficients for cubic power basis?

$$x(u) = c_0 + c_1 u + c_2 u^2 + c_3 u^3$$
$$x'(u) = c_1 + 2c_2\, u + 3c_3\, u^2$$

$$x(u_0) = x_0$$

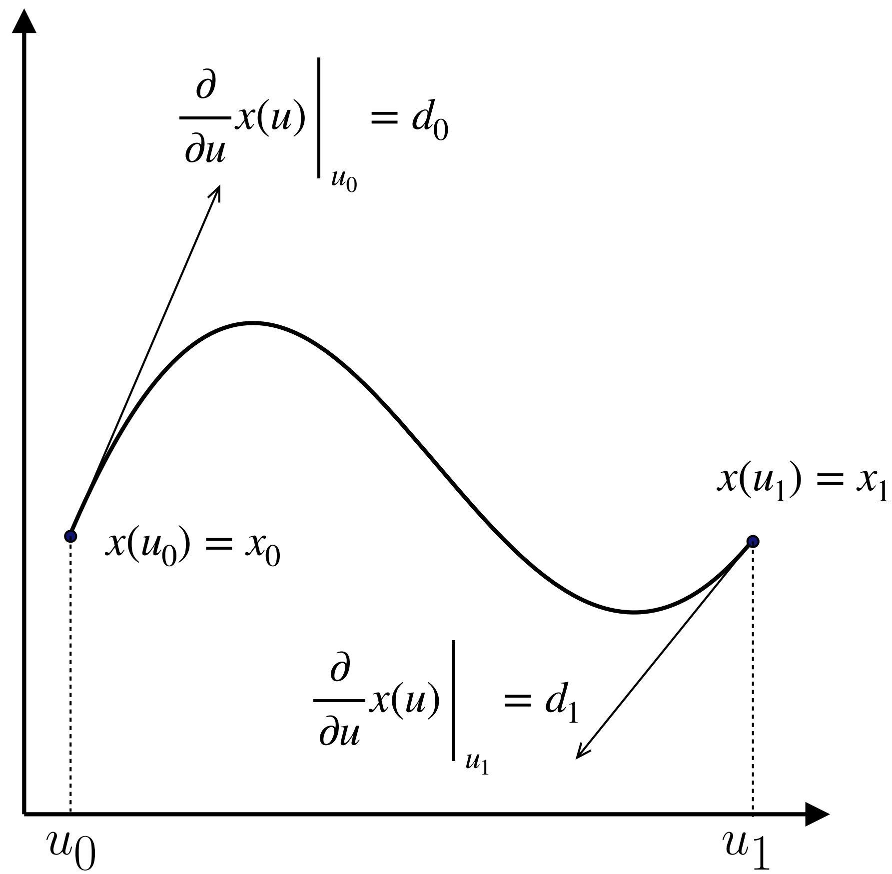
$$x(u_1) = x_1$$

$$x'(u_0) = d_0$$

$$x'(u_1) = d_1$$

For now, assume

$u_0 = 0 \quad u_1 = 1$



Specifying a Curve

Given desired values (constraints) how do we determine the coefficients for cubic power basis?

$$x(u) = c_0 + c_1 u + c_2 u^2 + c_3 u^3$$
$$x'(u) = c_1 + 2c_2\, u + 3c_3\, u^2$$

$$c_0 + c_1 u_0 + c_2 u_0^2 + c_3 u_0^3 = x_0$$

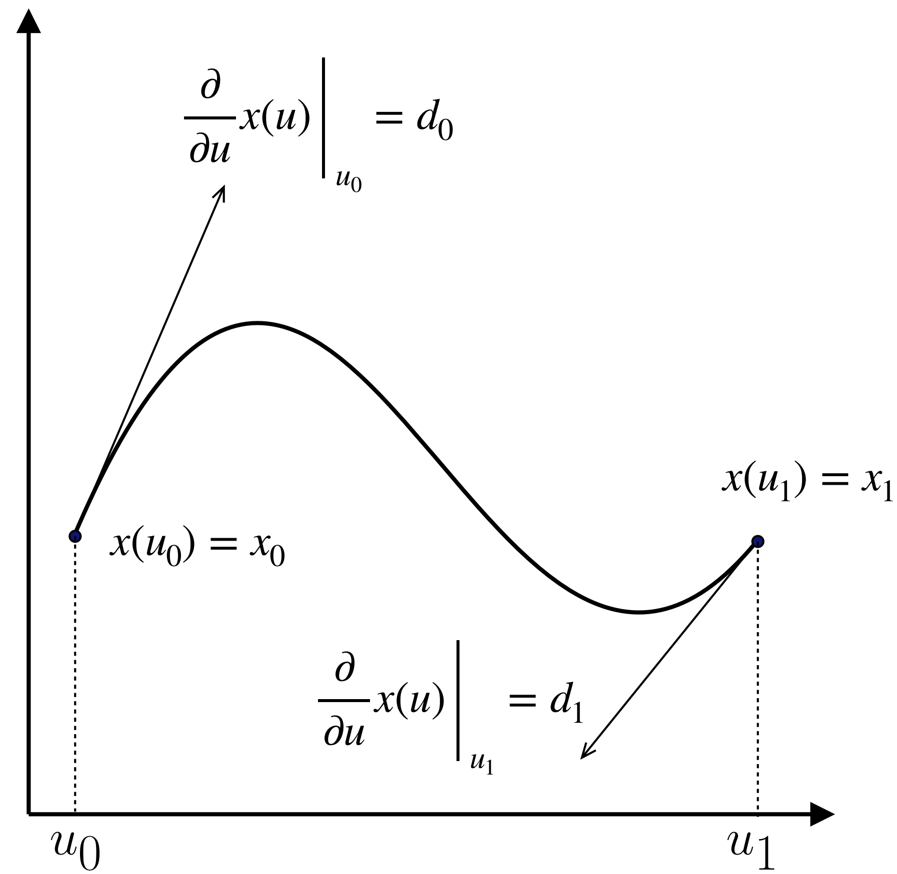
$$c_0 + c_1 u_1 + c_2 u_1^2 + c_3 u_1^3 = x_1$$

$$c_1 + 2c_2 u_0 + 3c_3 u_0^2 = d_0$$

$$c_1 + 2c_2 u_1 + 3c_3 u_1^2 = d_1$$

For now, assume

$u_0 = 0 \quad u_1 = 1$



Specifying a Curve

Given desired values (constraints) how do we determine the coefficients for cubic power basis?

$$x(u) = c_0 + c_1 u + c_2 u^2 + c_3 u^3$$
$$x'(u) = c_1 + 2c_2\, u + 3c_3\, u^2$$

$$c_0 + c_1\, 0 + c_2\, 0 + c_3\, 0 = x_0$$

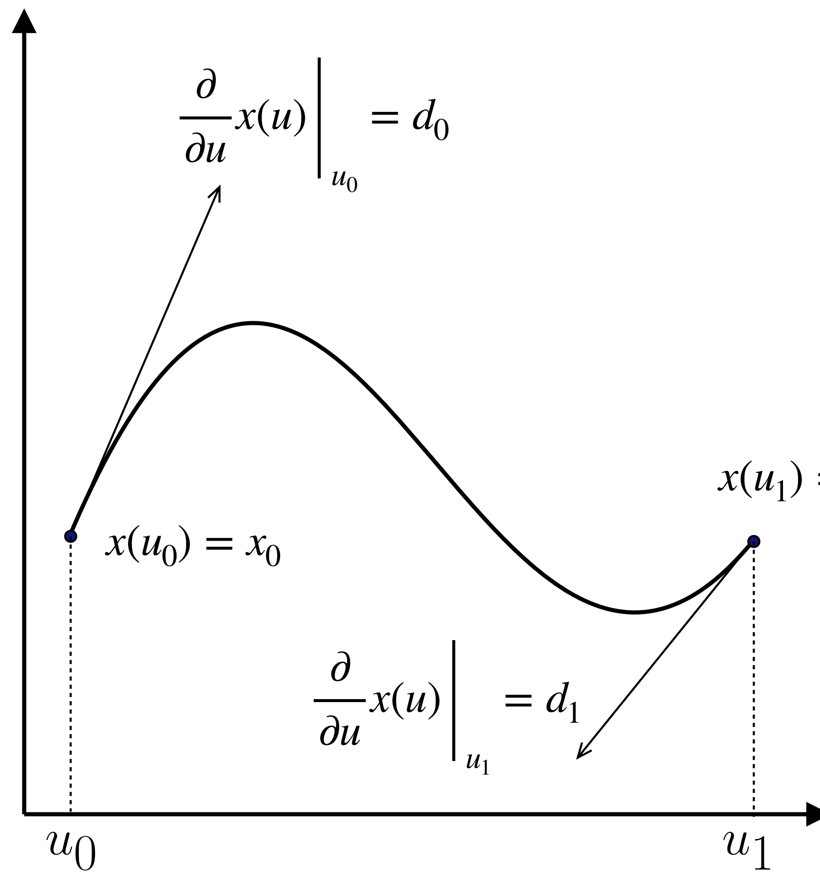
$$c_0 + c_1\, 1 + c_2\, 1 + c_3\, 1 = x_1$$

$$c_1 + 2c_2\, 0 + 3c_3\, 0 = d_0$$

$$c_1 + 2c_2\, 1 + 3c_3\, 1 = d_1$$

For now, assume

$u_0 = 0 \quad u_1 = 1$



Specifying a Curve

Given desired values (constraints) how do we determine the coefficients for cubic power basis?

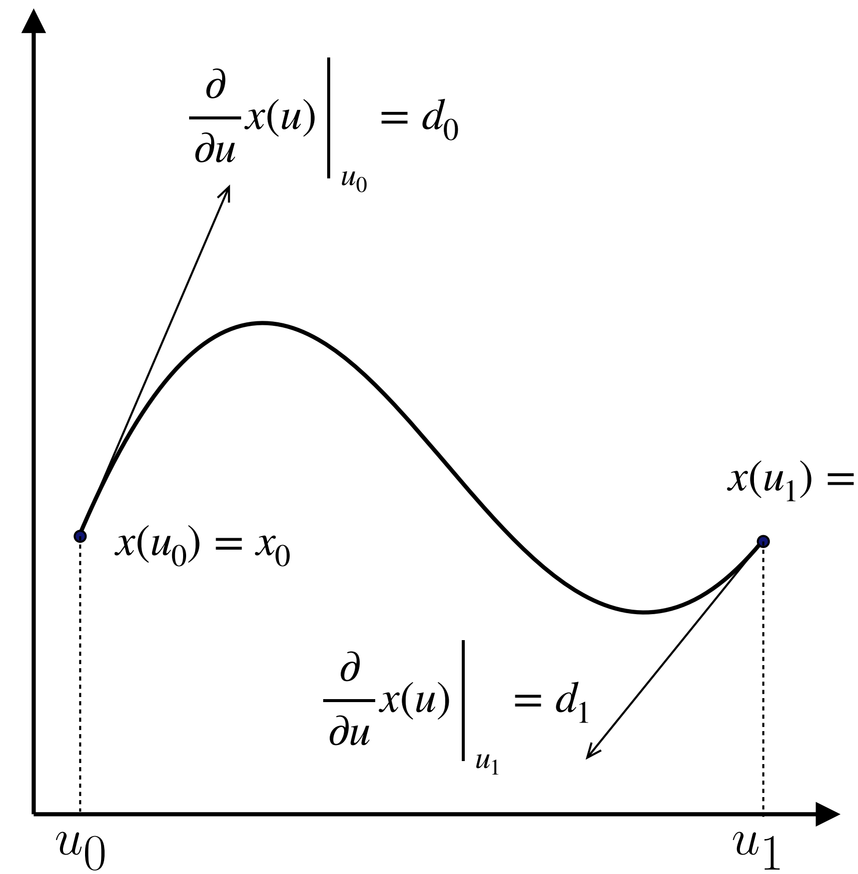
$$x(u) = c_0 + c_1 u + c_2 u^2 + c_3 u^3$$

$$x'(u) = c_1 + 2c_2\, u + 3c_3\, u^2$$

$$\begin{aligned} c_0 &= x_0 \\ c_0 + c_1 + c_2 + c_3 &= x_1 \\ c_1 &= d_0 \\ c_1 + 2c_2 + 3c_3 &= d_1 \end{aligned}$$

For now, assume

$u_0 = 0 \quad u_1 = 1$



Specifying a Curve

(From previous slide for reference.)

$$
\begin{aligned}
c_0 &= x_0 \\
c_0 + c_1 + c_2 + c_3 &= x_1 \\
c_1 &= d_0 \\
c_1 + 2c_2 + 3c_3 &= d_1
\end{aligned}
$$

Given desired values (constraints) how do we determine the coefficients for cubic power basis?

$$x(u) = c_0 + c_1 u + c_2 u^2 + c_3 u^3$$

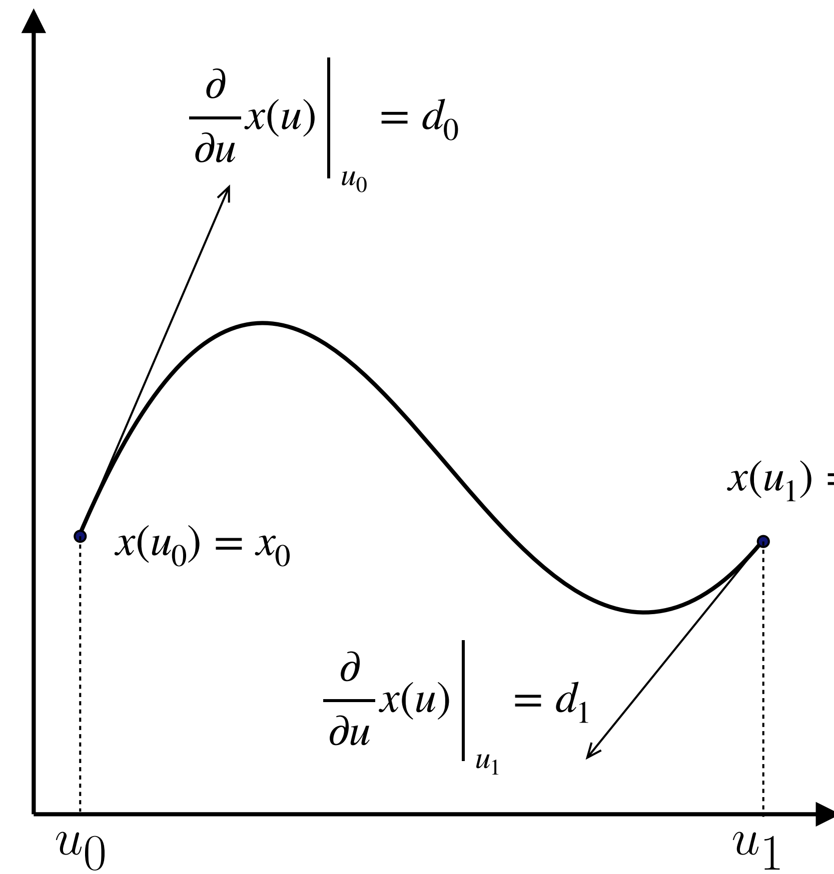
$$x'(u) = c_1 + 2c_2\, u + 3c_3\, u^2$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & 3 \end{bmatrix} \cdot \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} x_0 \\ x_1 \\ d_0 \\ d_1 \end{bmatrix}$$

Matrix

Coefficients (unknown) **Constraint Values**

For now, assume
$u_0 = 0 \quad u_1 = 1$



Specifying a Curve

Given desired values (constraints) how do we determine the coefficients for cubic power basis?

$$x(u) = c_0 + c_1 u + c_2 u^2 + c_3 u^3$$

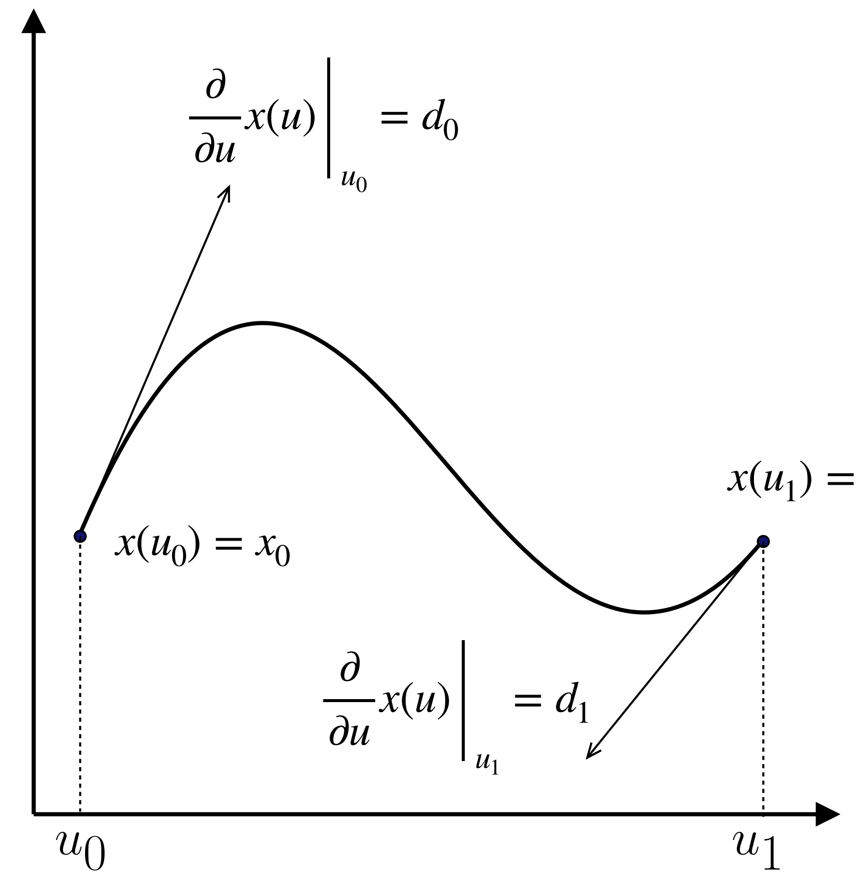
$$x'(u) = c_1 + 2c_2\, u + 3c_3\, u^2$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & 3 \end{bmatrix} \cdot \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} x_0 \\ x_1 \\ d_0 \\ d_1 \end{bmatrix}$$

$$\mathbf{B} \cdot \mathbf{c} = \mathbf{h}$$

Often called "control points", but note the *d* in this case are directions, not points.

For now, assume
$u_0 = 0 \quad u_1 = 1$

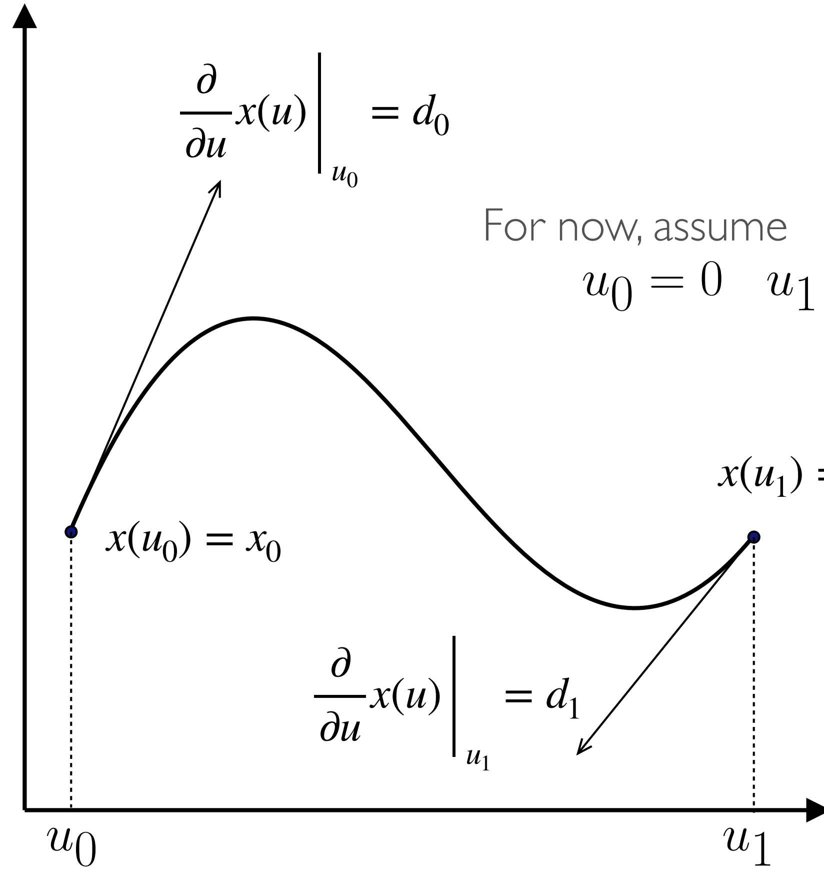


Specifying a Curve

$$x(u) = c_0 + c_1 u + c_2 u^2 + c_3 u^3$$

$$x'(u) = c_1 + 2c_2\,u + 3c_3\,u^2$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & 3 \end{bmatrix} \cdot \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} x_0 \\ x_1 \\ d_0 \\ d_1 \end{bmatrix}$$



For now, assume

$u_0 = 0 \quad u_1 = 1$

$$\mathbf{B} \cdot \mathbf{c} = \mathbf{h}$$

$$\mathbf{c} = \mathbf{B}^{-1} \cdot \mathbf{h}$$

$$\mathbf{c} = \beta_H \cdot \mathbf{h}$$

$$\begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -3 & 3 & -2 & -1 \\ 2 & -2 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \\ d_0 \\ d_1 \end{bmatrix}$$

Cubic Hermite Basis

$$\mathbf{c} = \boldsymbol{\beta}_H \cdot \mathbf{h}$$

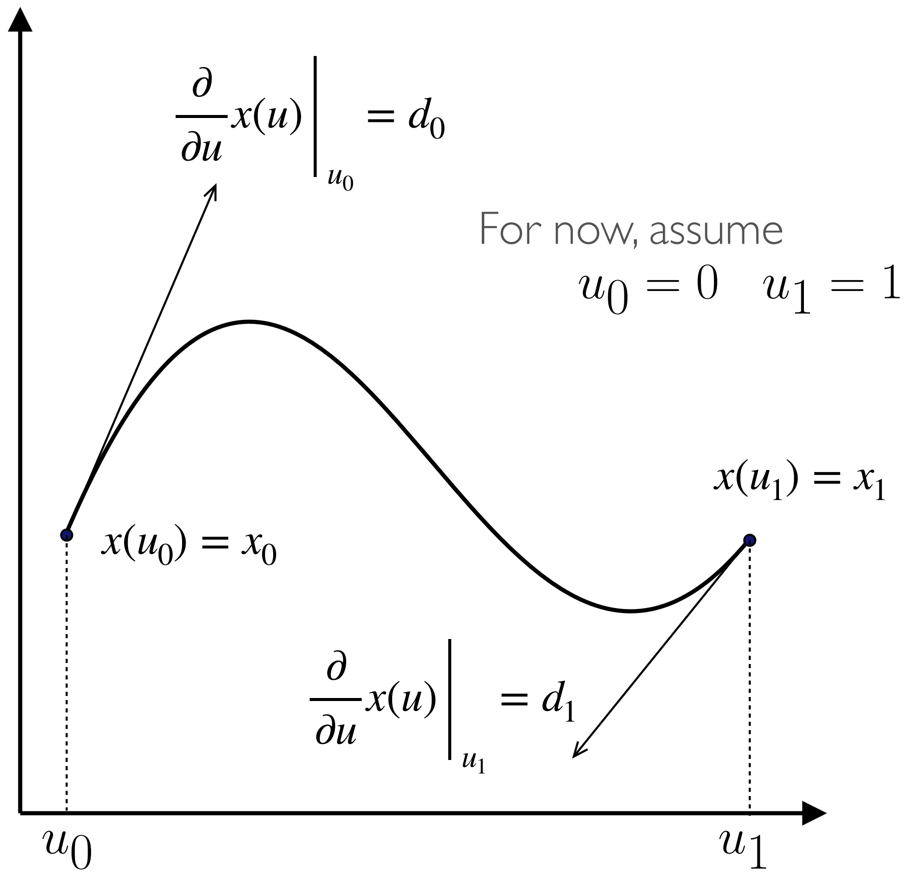
$$\begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -3 & 3 & -2 & -1 \\ 2 & -2 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \\ d_0 \\ d_1 \end{bmatrix}$$

$$x(u) = c_0 + c_1 u + c_2 u^2 + c_3 u^3$$

$$= \mathscr{P}^3 \cdot \mathbf{c}$$

$$= \mathscr{P}^3 \cdot \boldsymbol{\beta}_H \cdot \mathbf{h}$$

$$\boxed{x(u) = \mathscr{P}^3(u) \cdot \boldsymbol{\beta}_H \cdot \mathbf{h}}$$



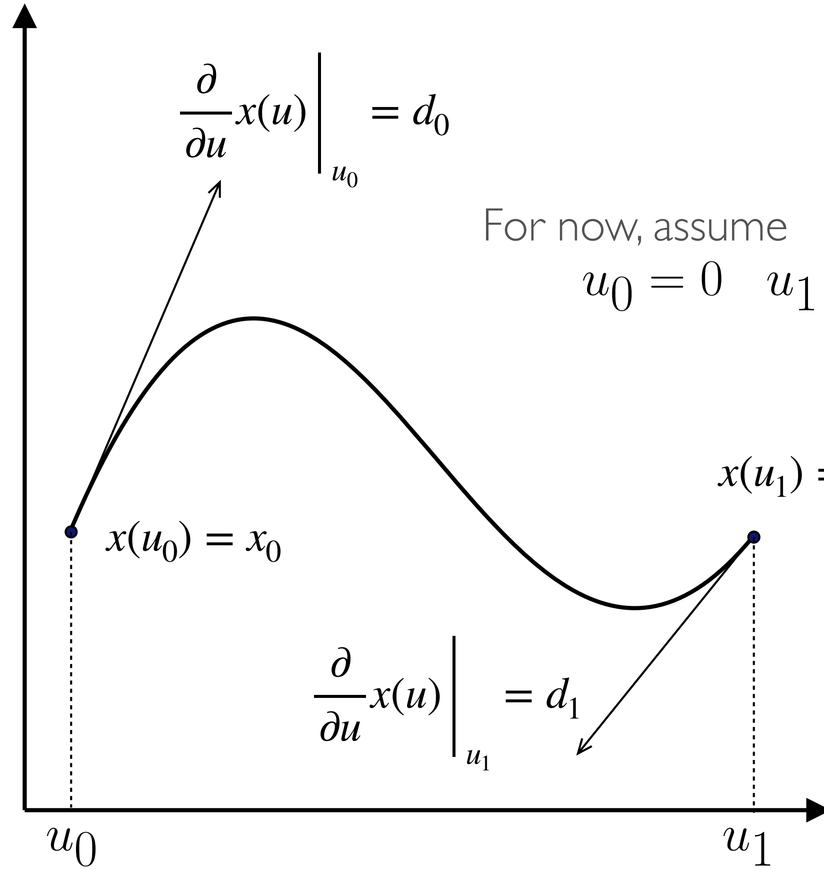
$\boldsymbol{\beta}_H \longleftarrow$ **Hermite Basis Matrix**

$$\mathscr{P}^3(u) = [1 \ u \ u^2 \ u^3]$$

Cubic Hermite Basis

$$\mathbf{c} = \boldsymbol{\beta}_H \cdot \mathbf{h}$$

$$\begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -3 & 3 & -2 & -1 \\ 2 & -2 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \\ d_0 \\ d_1 \end{bmatrix}$$



For now, assume
$u_0 = 0 \quad u_1 = 1$

$$x(u) = \boxed{\mathscr{P}^3(u) \cdot \boldsymbol{\beta}_H} \cdot \mathbf{h}$$

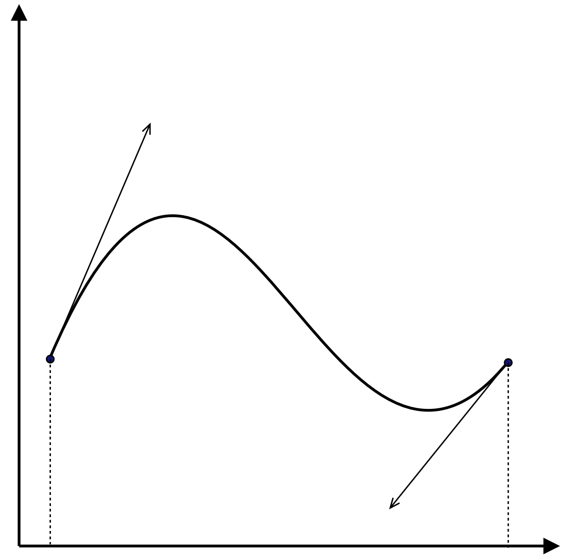
$$x(u) = \begin{bmatrix} 1 + 0\,u - 3\,u^2 + 2\,u^3 \\ 0 + 0\,u + 3\,u^2 - 2\,u^3 \\ 0 + 1\,u - 2\,u^2 + 1\,u^3 \\ 0 + 0\,u - 1\,u^2 + 1\,u^3 \end{bmatrix}^{\mathsf{T}} \cdot \mathbf{h}$$

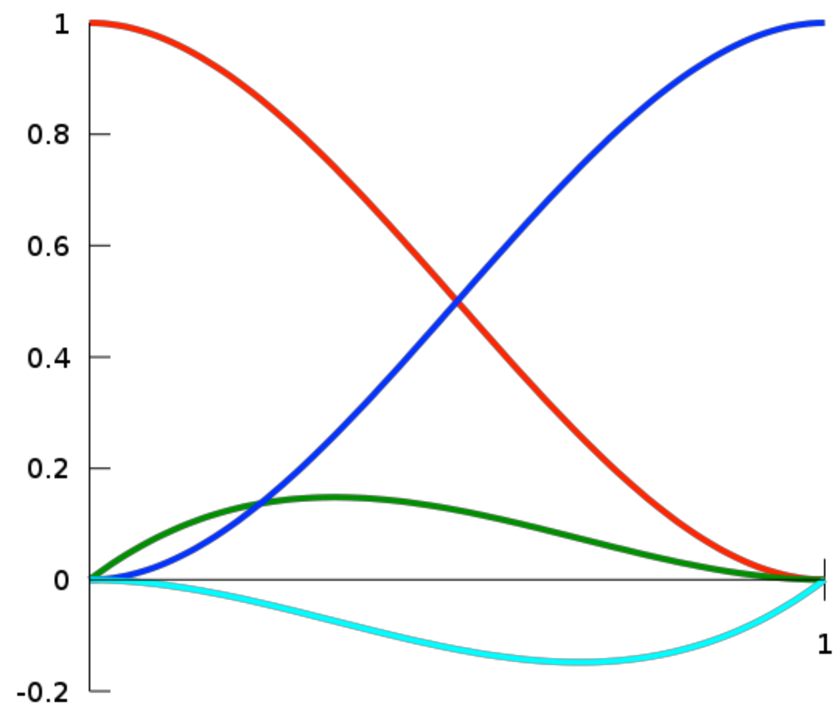
Cubic Hermite Basis

$$x(u) = \boxed{\mathscr{P}^3(u) \cdot \boldsymbol{\beta}_H} \cdot \mathbf{h}$$

$$= \begin{bmatrix} 1 + 0\,u - 3\,u^2 + 2\,u^3 \\ 0 + 0\,u + 3\,u^2 - 2\,u^3 \\ 0 + 1\,u - 2\,u^2 + 1\,u^3 \\ 0 + 0\,u - 1\,u^2 + 1\,u^3 \end{bmatrix}^{\mathsf{T}} \cdot \mathbf{h}$$

Hermite Basis Functions

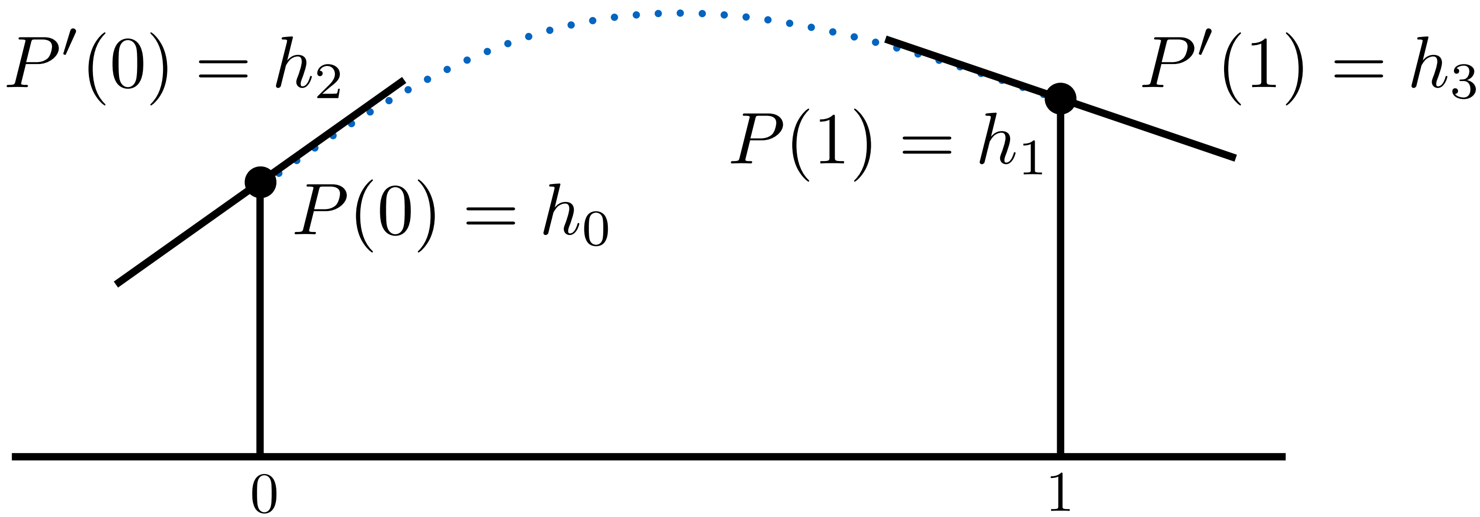




Basis matrix depends on u_0 and u_1

We could use this same approach for higher order than cubic.

Recap: Cubic Hermite Interpolation



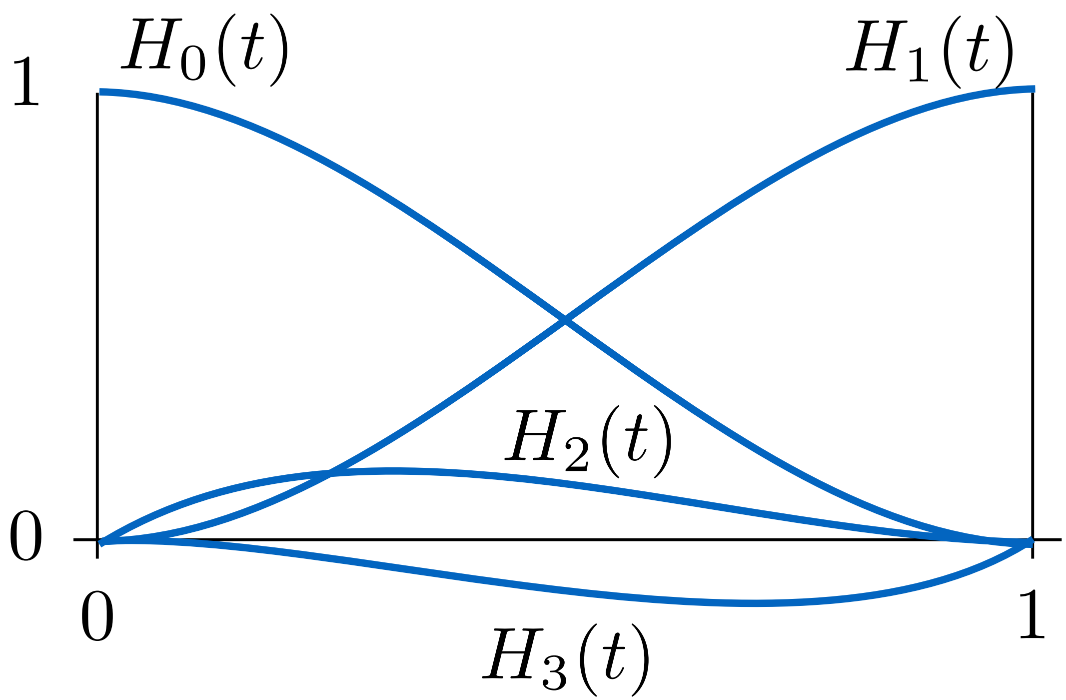
Inputs: values and derivatives at endpoints

Output: cubic polynomial that interpolates

Solution: weighted sum of Hermite basis functions

$$P(t) = h_0\,H_0(t) + h_1\,H_1(t) + h_2\,H_2(t) + h_3\,H_3(t)$$

Recap: Cubic Hermite Interpolation



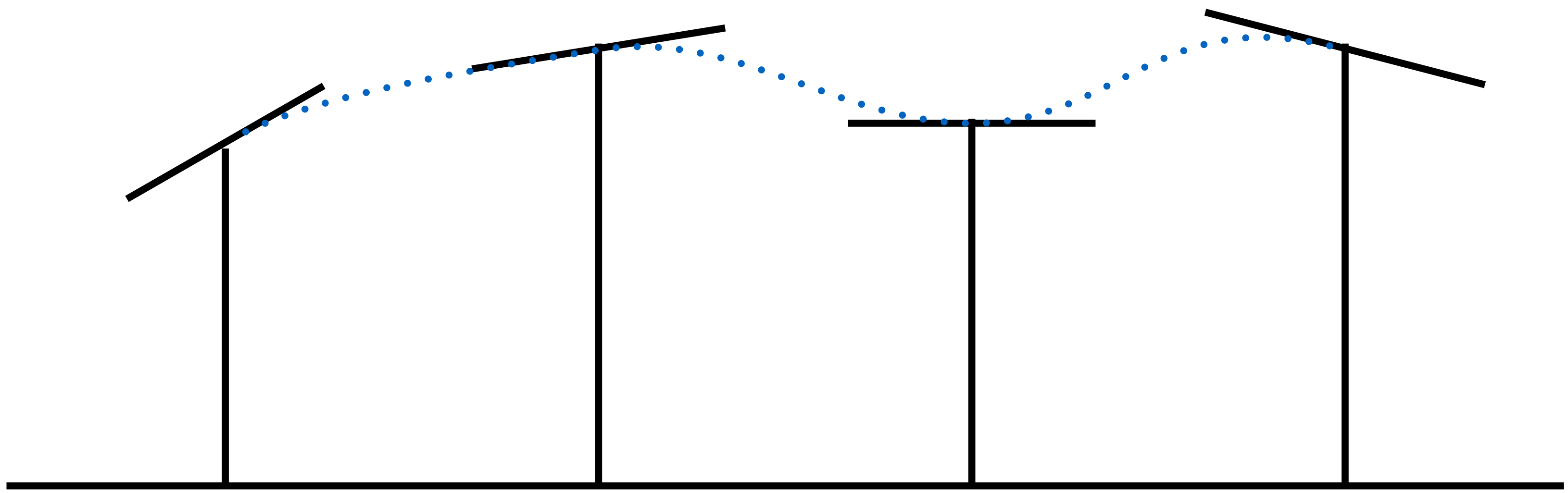
$$H_0(t) = 2t^3 - 3t^2 + 1$$

$$H_1(t) = -2t^3 + 3t^2$$

$$H_2(t) = t^3 - 2t^2 + t$$

$$H_3(t) = t^3 - t^2$$

Hermite Spline Interpolation

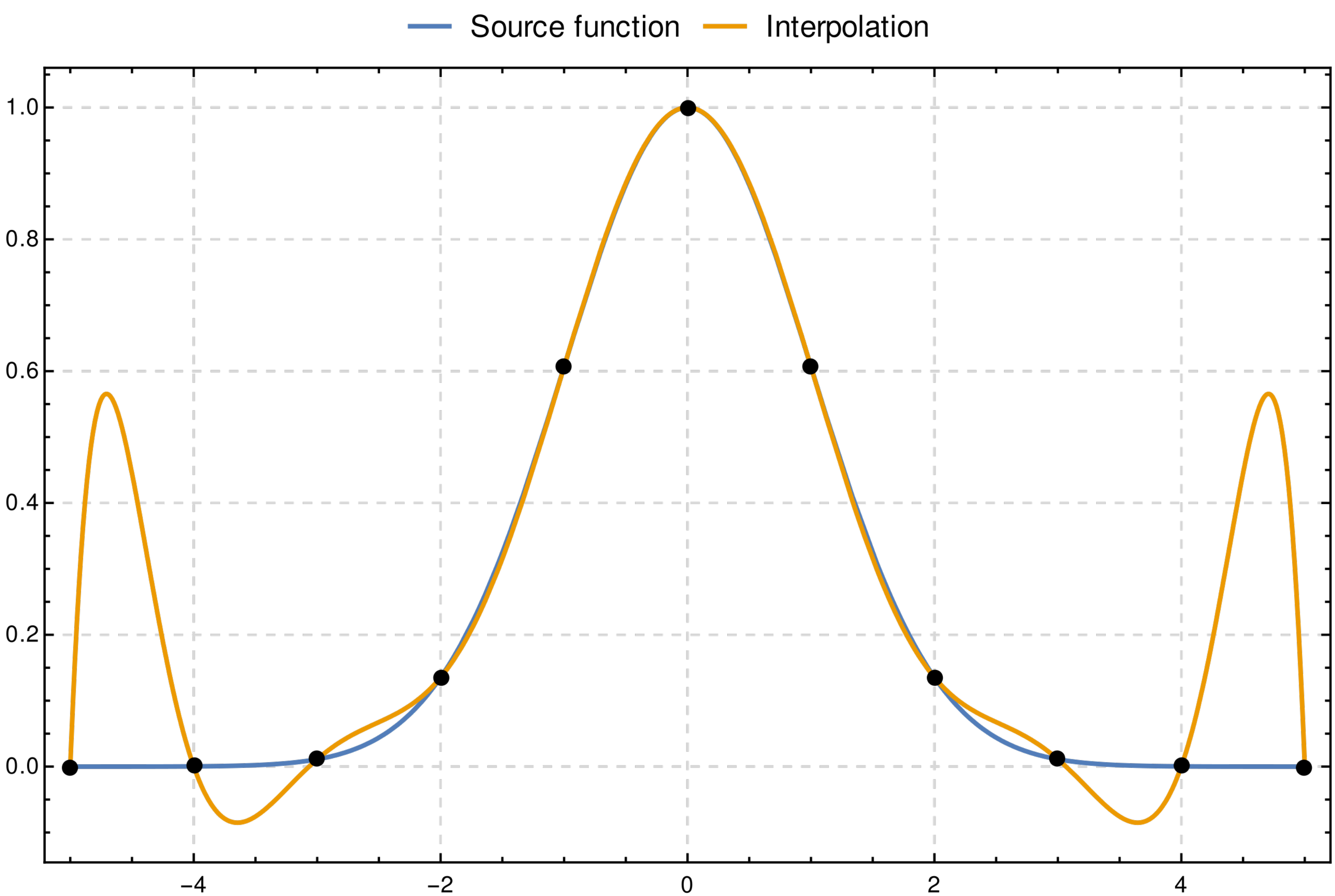


Inputs: sequence of values and derivatives

Question: Why not use higher-order polynomial?

Higher order polynomials tend to wiggle excessively.

Would like to keep effect of edits "local".



Example of interpolation divergence for a set of Lagrange polynomials.

https://en.wikipedia.org/wiki/Lagrange_polynomial

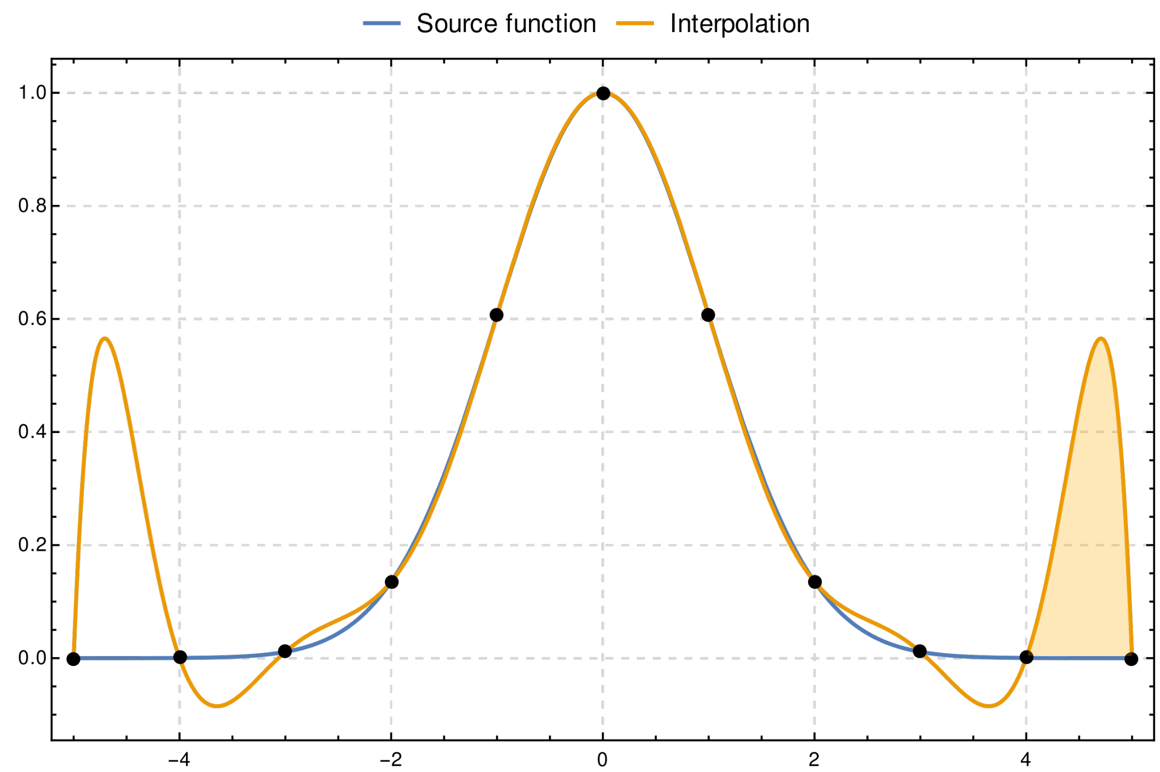
Question: Why not use higher-order polynomial?

Higher order polynomials tend to wiggle excessively.

Higher-Order Polynomial?

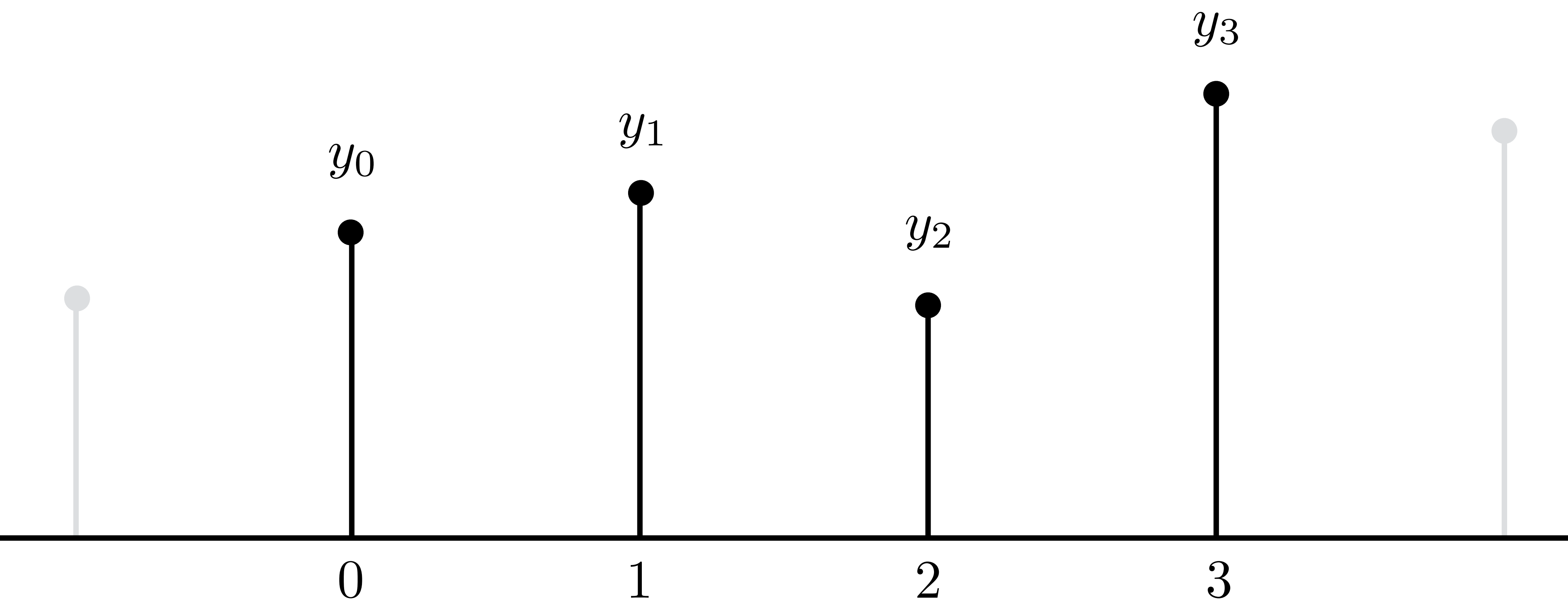
High-degree polynomials don't behave well

- Unstable
- No local edits
- Expensive



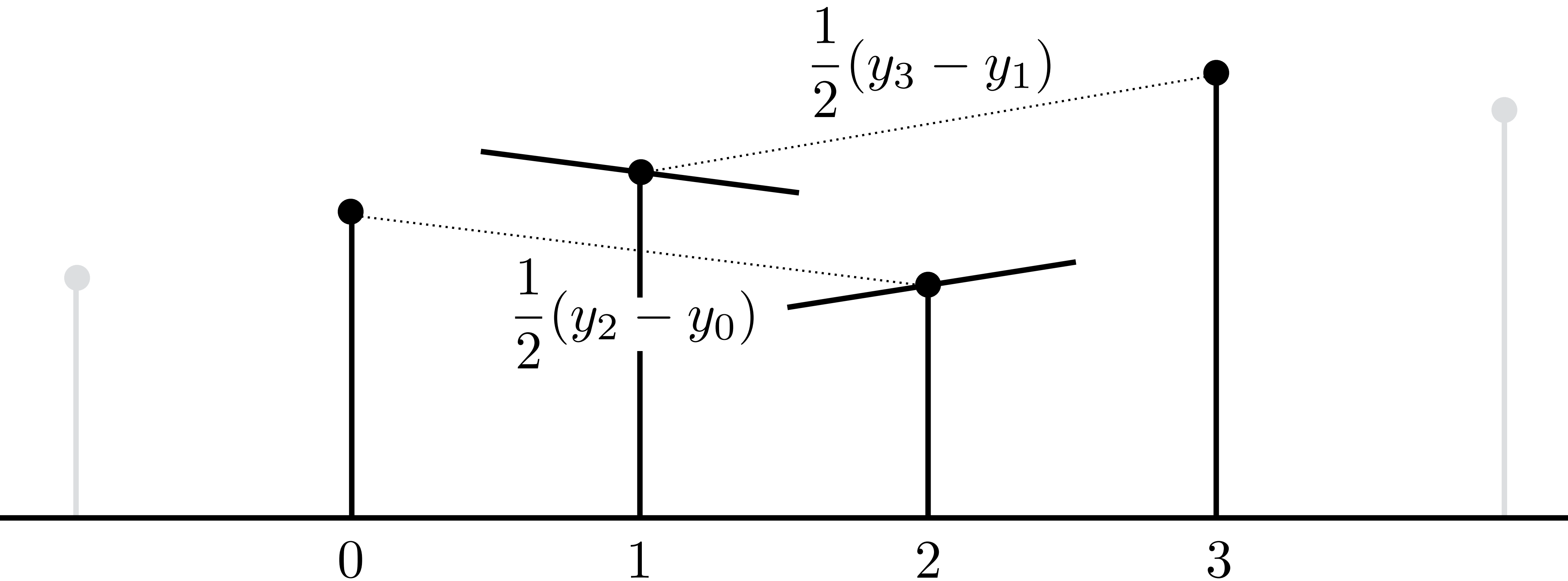
Catmull-Rom Interpolation

Catmull-Rom Interpolation



Inputs: sequence of values

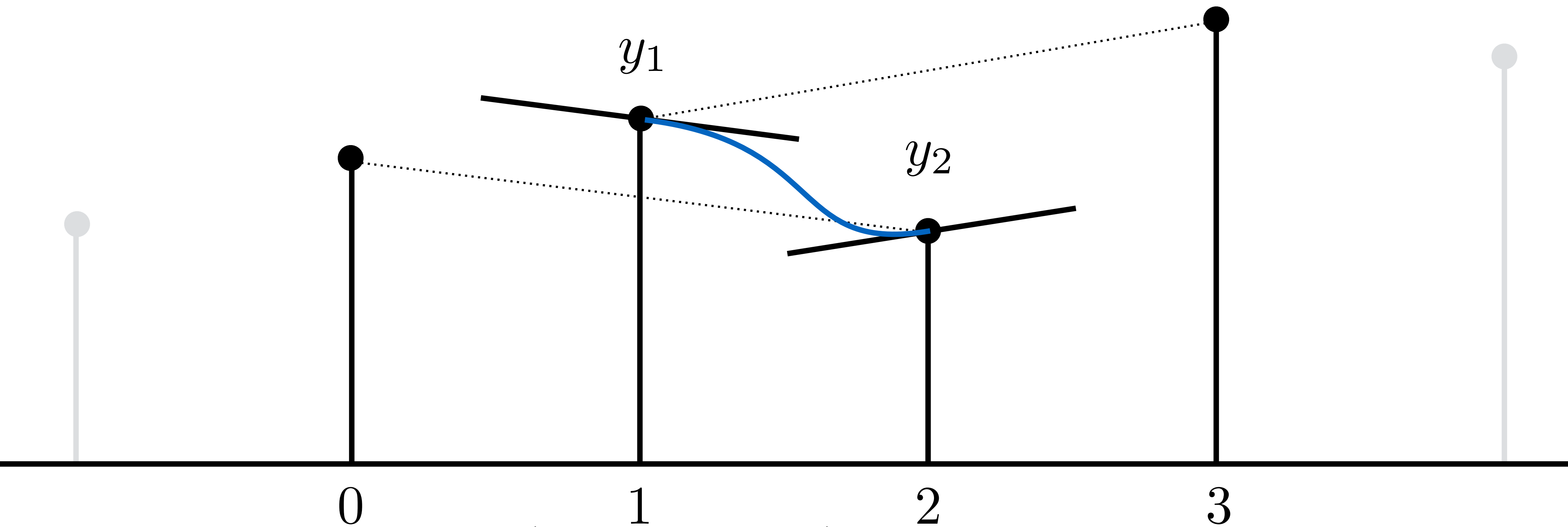
Catmull-Rom Interpolation



Rule for derivatives:

Match slope between previous and next values

Catmull-Rom Interpolation



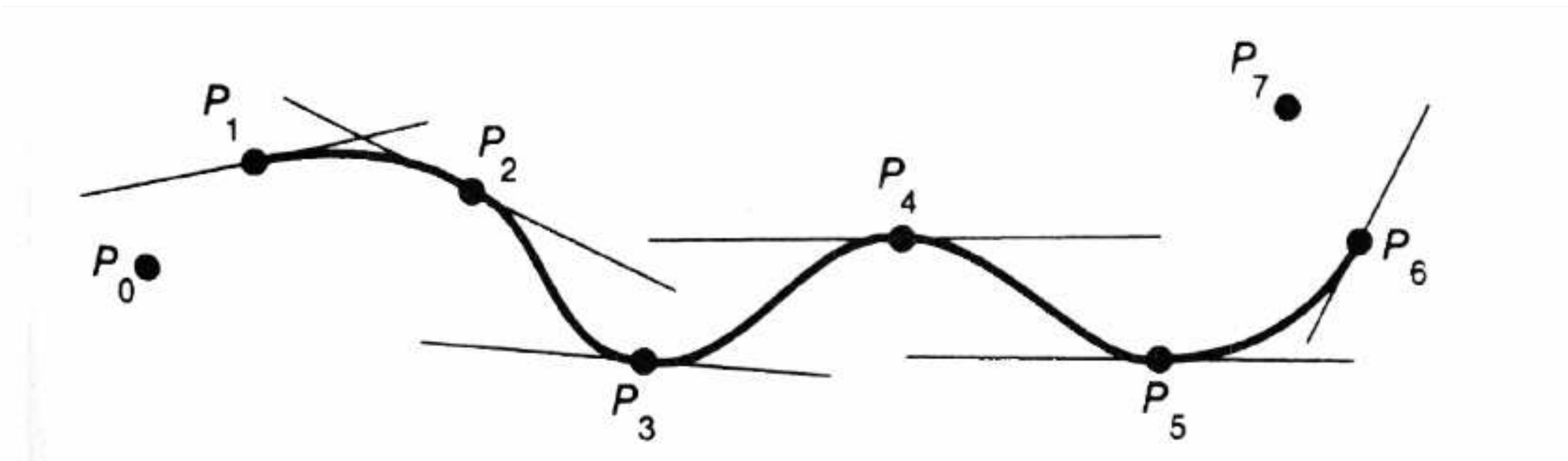
$$\frac{1}{2}(y_2 - y_0) \quad \frac{1}{2}(y_3 - y_1)$$

Then use Hermite interpolation

Catmull-Rom Spline

Input: sequence of points

Output: spline that interpolates all points with C1 continuity

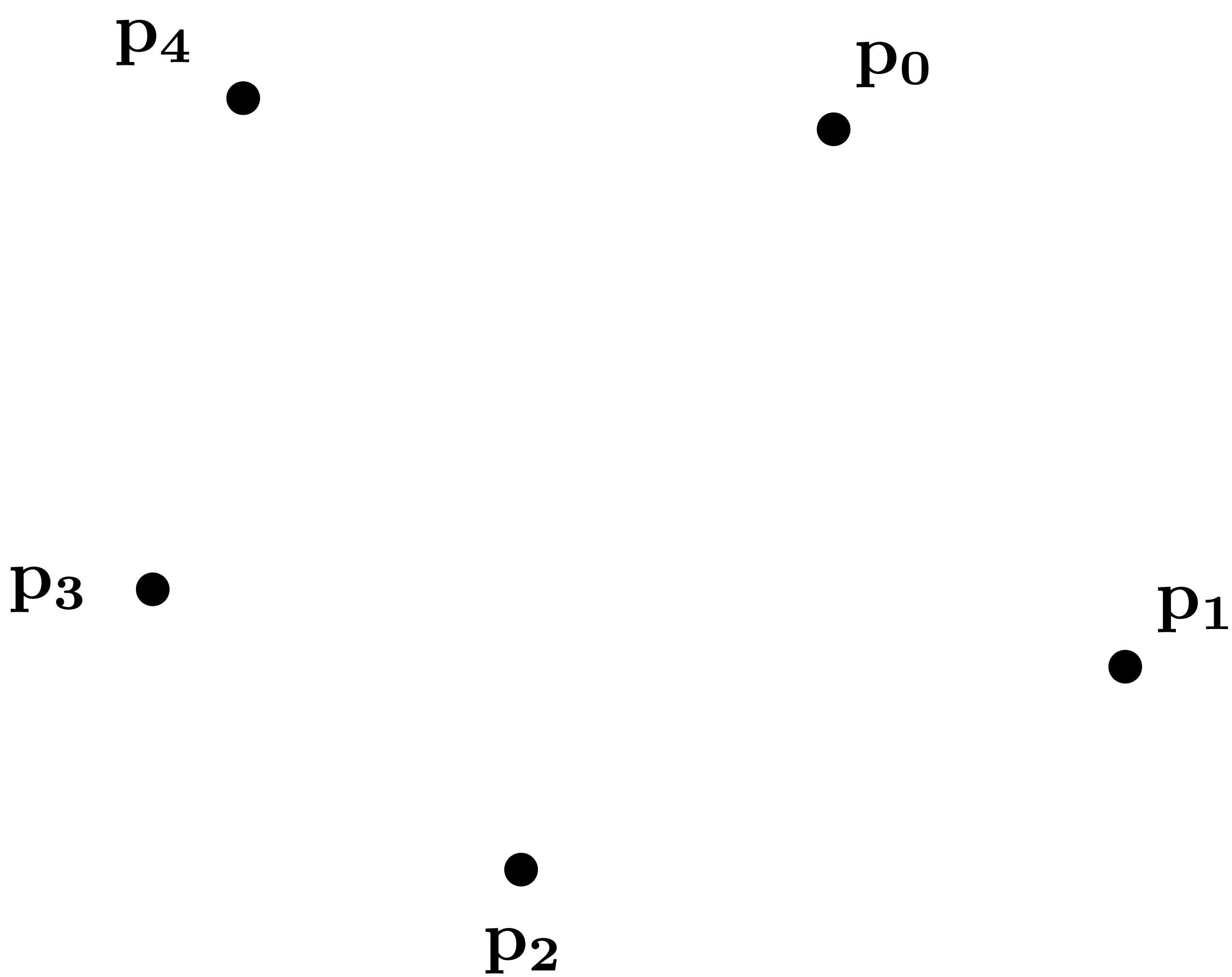


- Piecewise Cubic Curve
- C1 continuous!

Interpolating Points & Vectors

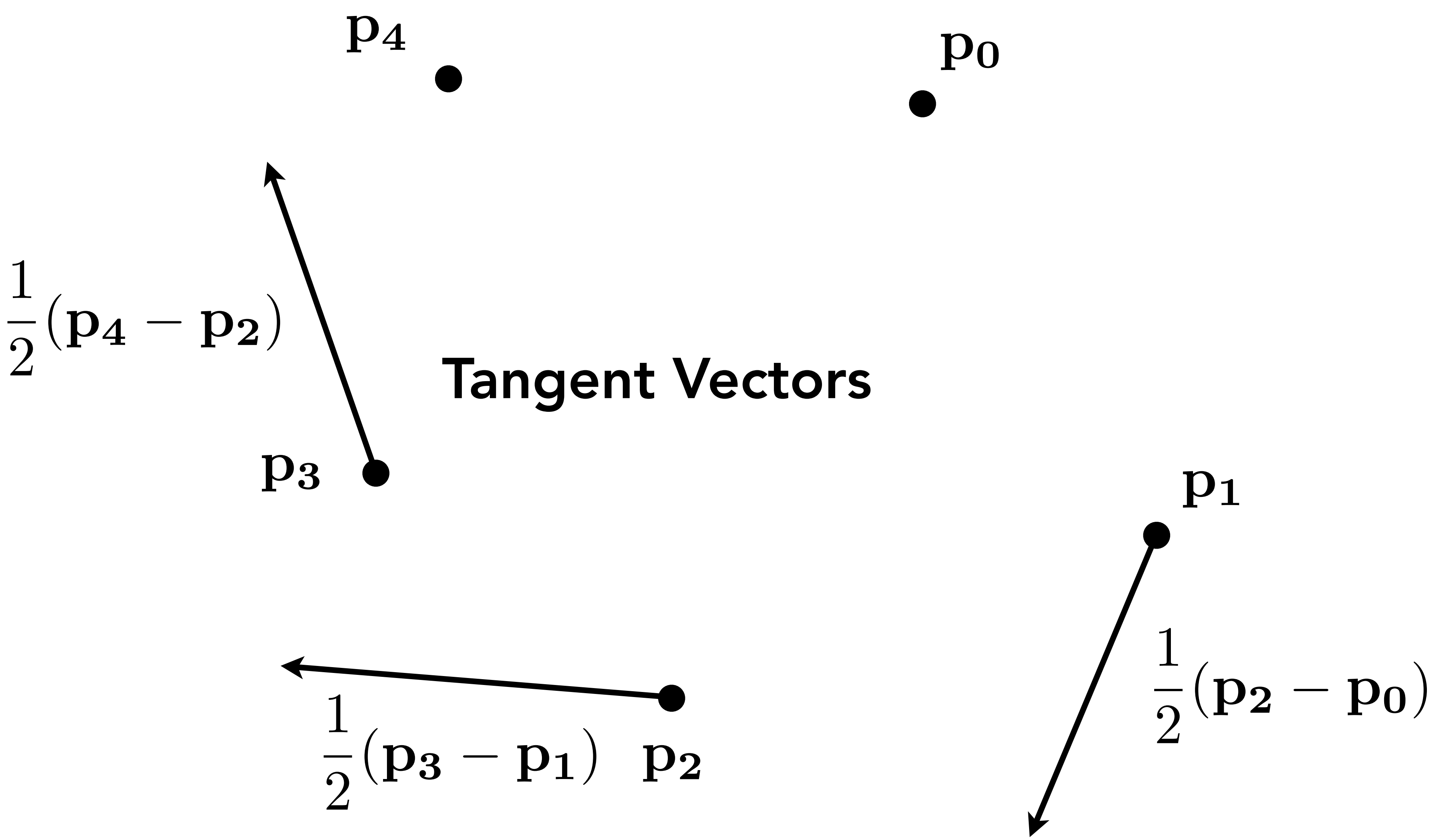
Can Interpolate Points As Easily As Values

E.g. point (0,1,3) in 3D space, or even a general vector in N dimensions



Catmull-Rom 3D spline control points

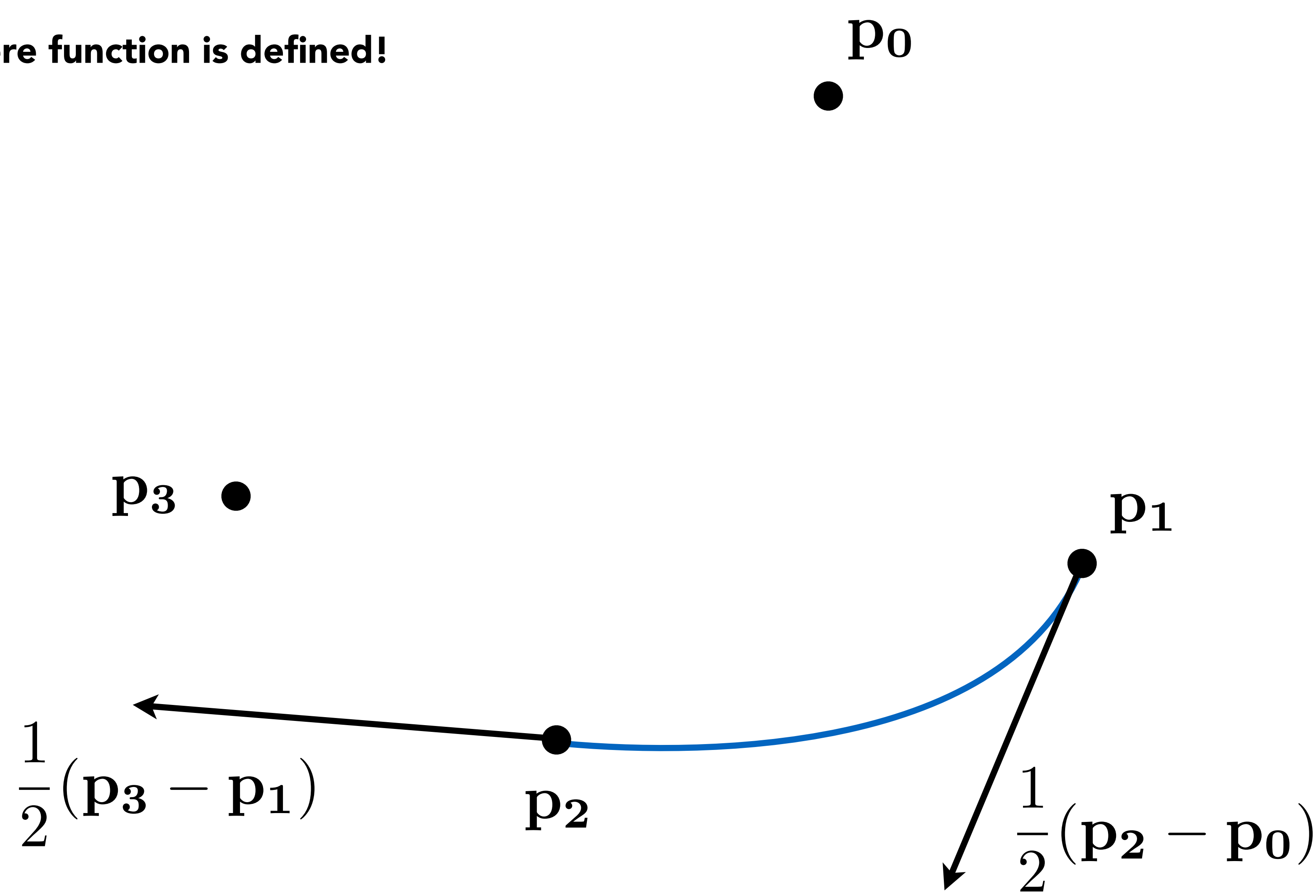
Can Interpolate Points As Easily As Values



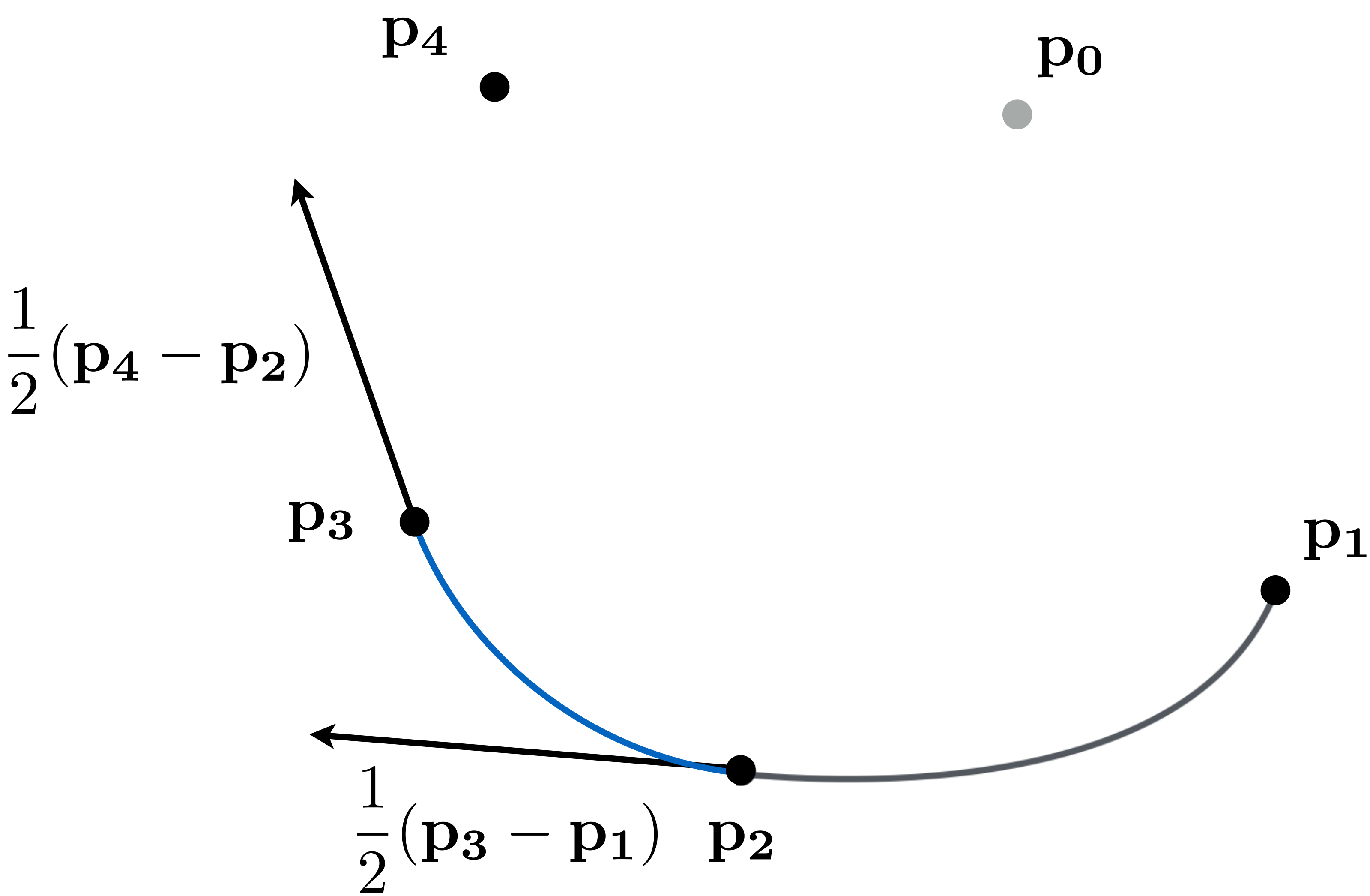
Catmull-Rom 3D tangent vectors

Can Interpolate Points As Easily As Values

Notice where function is defined!



Can Interpolate Points As Easily As Values



Catmull-Rom 3D space curve

Use Basis Functions to Define Curves

General formula for a particular interpolation scheme: $\mathbf{p}(t) = \sum_{i=0}^{n} \mathbf{p}_i F_i(t)$

$$x(t) = \sum_{i=0}^{n} x_i F_i(t) \qquad y(t) = \sum_{i=0}^{n} y_i F_i(t) \qquad z(t) = \sum_{i=0}^{n} z_i F_i(t)$$

Coefficients $\mathbf{p}_i$ can be points & vectors, not just values

$F_i(t)$ are basis functions for the interpolation scheme.

Use Basis Functions to Define Curves

$$P(t) = h_0\, H_0(t) + h_1\, H_1(t) + h_2\, H_2(t) + h_3\, H_3(t)$$

$$x(t) = \sum_{i=0}^{n} x_i F_i(t) \qquad y(t) = \sum_{i=0}^{n} y_i F_i(t) \qquad z(t) = \sum_{i=0}^{n} z_i F_i(t)$$

Saw $H_i(t)$ for Hermite interpolation earlier. Will see $C_i(t)$ for Catmull-Rom shortly, and $B_i(t)$ for Bézier scheme later. The basis functions are properties of the interpolation scheme.

Matrix Form of Catmull-Rom Space Curve?

Use Hermite matrix form

- Points & tangents given by Catmull-Rom rules

Power Basis

Hermite Basis Matrix β_H

Hermite Control Values

Hermite points

$$\mathbf{h}_0 = \mathbf{p}_1$$

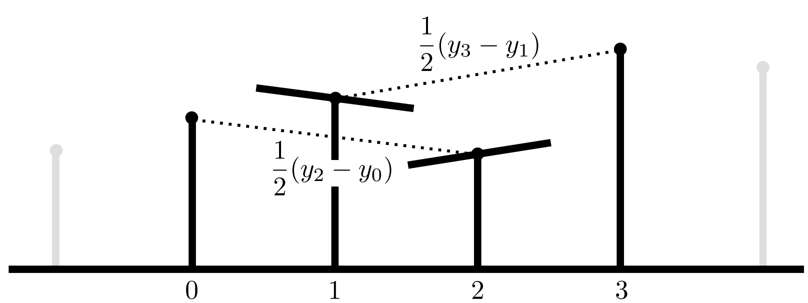
$$\mathbf{h}_1 = \mathbf{p}_2$$

Hermite tangents

$$\mathbf{h}_2 = \tfrac{1}{2}(\mathbf{p}_2 - \mathbf{p}_0)$$

$$\mathbf{h}_3 = \tfrac{1}{2}(\mathbf{p}_3 - \mathbf{p}_1)$$

$$\mathbf{p}(u) = \begin{bmatrix} 1 \\ u \\ u^2 \\ u^3 \end{bmatrix}^{\mathsf{T}} \cdot \beta_H \cdot \begin{bmatrix} \mathbf{h}_0 \\ \mathbf{h}_1 \\ \mathbf{h}_2 \\ \mathbf{h}_3 \end{bmatrix}$$



Matrix Form of Catmull-Rom Space Curve?

Use Hermite matrix form

- Points & tangents given by Catmull-Rom rules

Power Basis

Hermite Basis Matrix β_H

Convert Catmull-Rom to Hermite

Catmull-Rom Control Points

Hermite points

$$\mathbf{h}_0 = \mathbf{p}_1$$

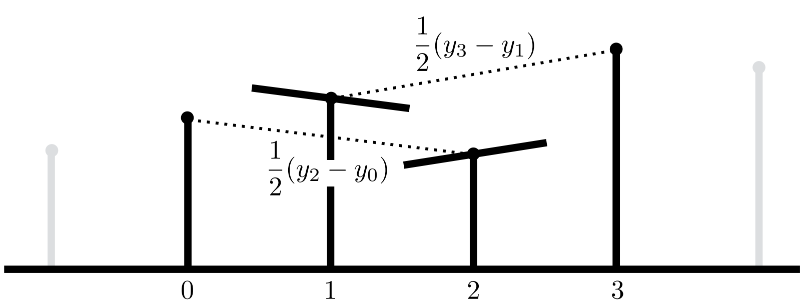
$$\mathbf{h}_1 = \mathbf{p}_2$$

Hermite tangents

$$\mathbf{h}_2 = \frac{1}{2}(\mathbf{p}_2 - \mathbf{p}_0)$$

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

$$\mathbf{p}(u) = \begin{bmatrix} 1 \\ u \\ u^2 \\ u^3 \end{bmatrix}^{\mathsf{T}} \cdot \beta_H \cdot \mathbf{M}_{CR \to H} \cdot \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{bmatrix}$$



Matrix Form of Catmull-Rom Space Curve?

Use Hermite matrix form

- Points & tangents given by Catmull-Rom rules

Power Basis

Hermite Basis Matrix β_H

Convert Catmull-Rom to Hermite

Catmull-Rom Control Points

Hermite points

$$\mathbf{h}_0 = \mathbf{p}_1$$

$$\mathbf{h}_1 = \mathbf{p}_2$$

Hermite tangents

$$\mathbf{h}_2 = \tfrac{1}{2}(\mathbf{p}_2 - \mathbf{p}_0)$$

$$\mathbf{h}_3 = \tfrac{1}{2}(\mathbf{p}_3 - \mathbf{p}_1)$$

$$\mathbf{p}(u) = \begin{bmatrix} 1 \\ u \\ u^2 \\ u^3 \end{bmatrix}^{\mathsf{T}} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -3 & 3 & -2 & -1 \\ 2 & -2 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \frac{-1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & \frac{-1}{2} & 0 & \frac{1}{2} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{bmatrix}$$

Matrix Form of Catmull-Rom Space Curve?

Use Hermite matrix form

- Points & tangents given by Catmull-Rom rules

Power Basis

Catmull-Rom Basis Matrix β_{CR}

Catmull-Rom Control Points

Hermite points

$$\mathbf{h}_0 = \mathbf{p}_1$$

$$\mathbf{h}_1 = \mathbf{p}_2$$

Hermite tangents

$$\mathbf{h}_2 = \tfrac{1}{2}(\mathbf{p}_2 - \mathbf{p}_0)$$

$$\mathbf{h}_3 = \tfrac{1}{2}(\mathbf{p}_3 - \mathbf{p}_1)$$

$$\mathbf{p}(u) = \begin{bmatrix} 1 \\ u \\ u^2 \\ u^3 \end{bmatrix}^{\mathsf{T}} \cdot \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{-1}{2} & 0 & \frac{1}{2} & 0 \\ 1 & -2\frac{1}{2} & 2 & \frac{-1}{2} \\ \frac{-1}{2} & 1\frac{1}{2} & -1\frac{1}{2} & \frac{1}{2} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{bmatrix}$$

Matrix Form of Catmull-Rom Space Curve?

Use Hermite matrix form

- Points & tangents given by Catmull-Rom rules

Power Basis

Catmull-Rom Basis Matrix β_{CR}

Catmull-Rom Control Points

Hermite points

$$\mathbf{h}_0 = \mathbf{p}_1$$

$$\mathbf{h}_1 = \mathbf{p}_2$$

Hermite tangents

$$\mathbf{h}_2 = \tfrac{1}{2}(\mathbf{p}_2 - \mathbf{p}_0)$$

$$\mathbf{h}_3 = \tfrac{1}{2}(\mathbf{p}_3 - \mathbf{p}_1)$$

$$\mathbf{p}(u) = \begin{bmatrix} 1 \\ u \\ u^2 \\ u^3 \end{bmatrix}^{\mathsf{T}} \cdot \beta_{CR} \cdot \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{bmatrix}$$

$$\mathbf{p}(u) = \mathscr{P}^3(u) \cdot \beta_{CR} \cdot \mathbf{p}$$

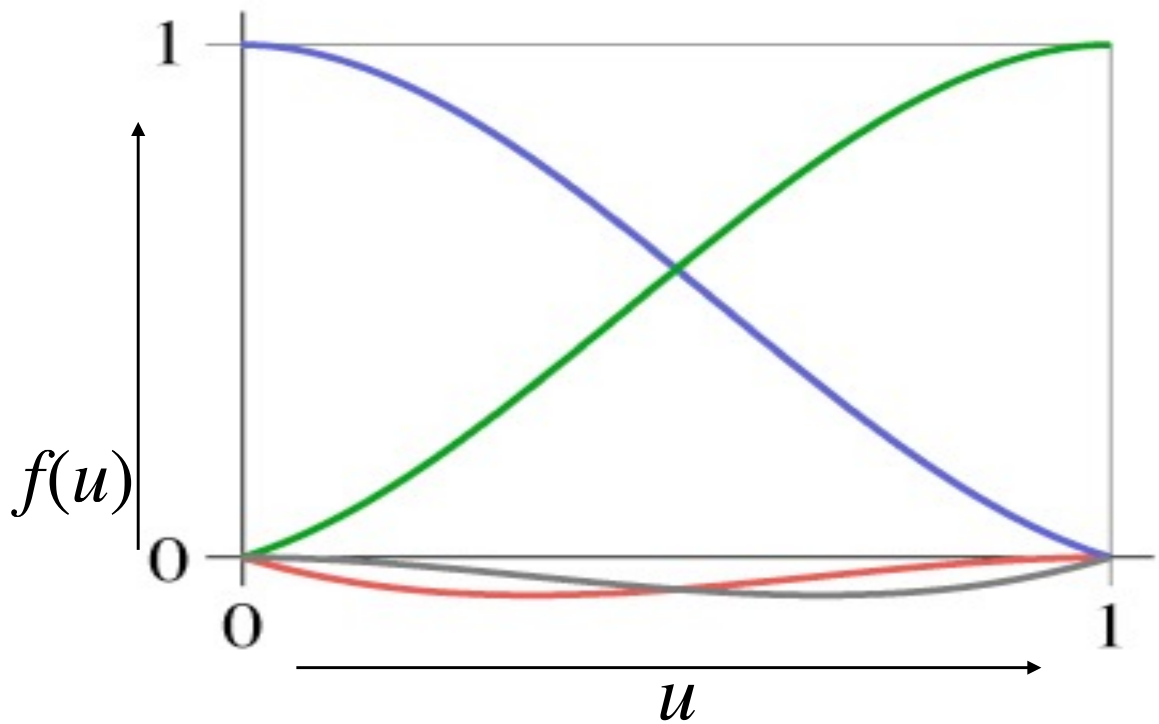
Catmull-Rom Basis Functions

$$\mathbf{p}(u) = \begin{bmatrix} 0 - \frac{1}{2}u + 1u^2 - \frac{1}{2}u^3 \\ 1 + 0u - 2\frac{1}{2}u^2 + 1\frac{1}{2}u^3 \\ 0 + \frac{1}{2}u + 2u^2 - 1\frac{1}{2}u^3 \\ 0 + 0u - \frac{1}{2}u^2 + \frac{1}{2}u^3 \end{bmatrix}^{\mathsf{T}} \cdot \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{bmatrix}$$

$$\mathbf{p}(u) = \mathscr{P}^3(u) \cdot \boldsymbol{\beta}_{CR} \cdot \mathbf{p}$$

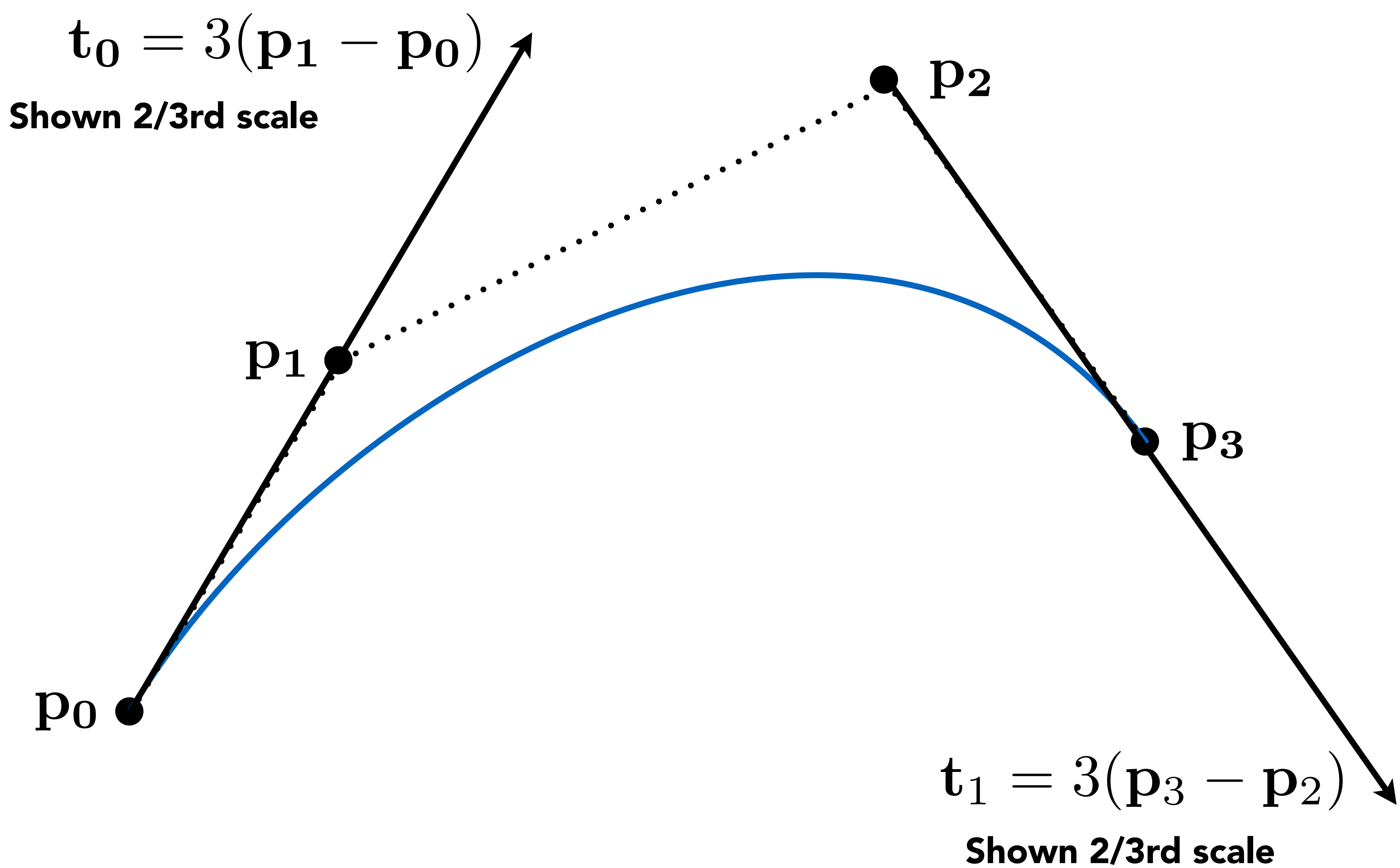
Matrix columns are Catmull-Rom basis functions

Compare to Hermite



Bézier Curves

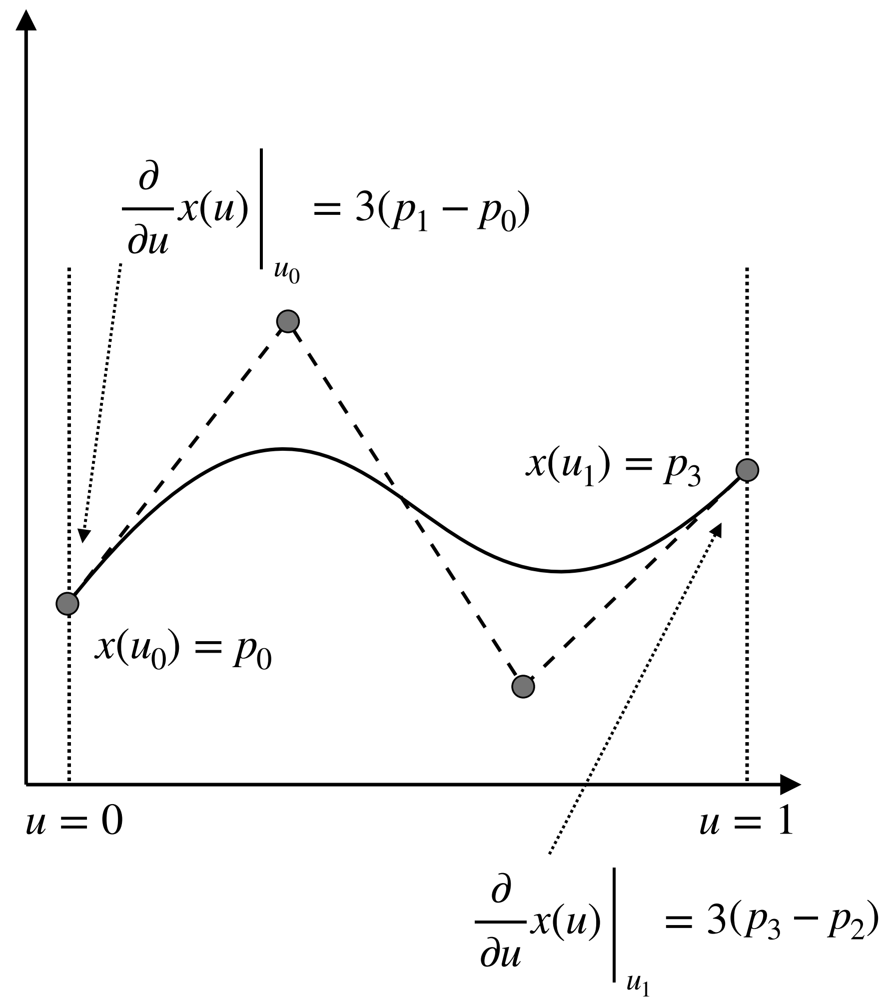
Defining Cubic Bézier Curve With Tangents



Cubic Bézier

Similar to Hermite, but specify tangents indirectly

$$x(u) = c_0 + c_1 u + c_2 u^2 + c_3 u^3$$



Cubic Bézier

Similar to Hermite, but specify tangents indirectly

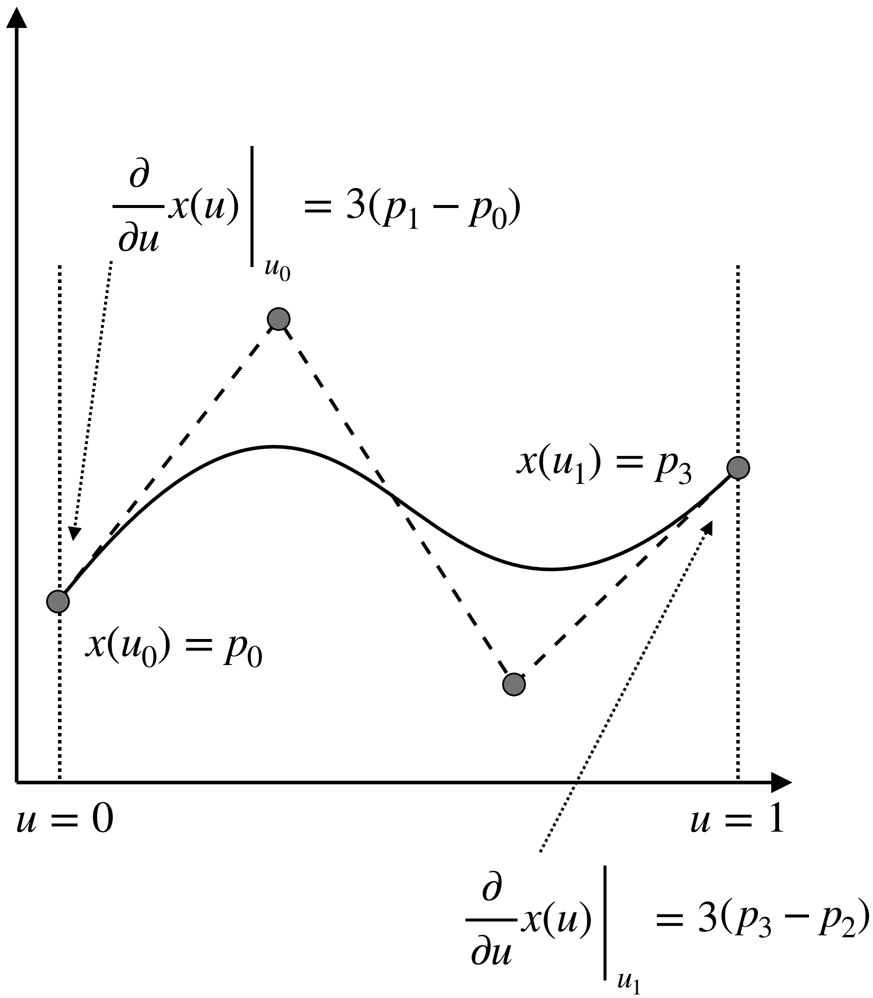
$$x(u) = c_0 + c_1 u + c_2 u^2 + c_3 u^3$$

$$x(u_0) = h_0 = p_0$$

$$x(u_1) = h_1 = p_3$$

$$x'(u_0) = h_2 = 3(p_1 - p_2)$$

$$x'(u_1) = h_3 = 3(p_3 - p_2)$$



Cubic Bézier

Similar to Hermite, but specify tangents indirectly

$$x(u) = c_0 + c_1 u + c_2 u^2 + c_3 u^3$$

$$x(u_0) = h_0 = p_0$$

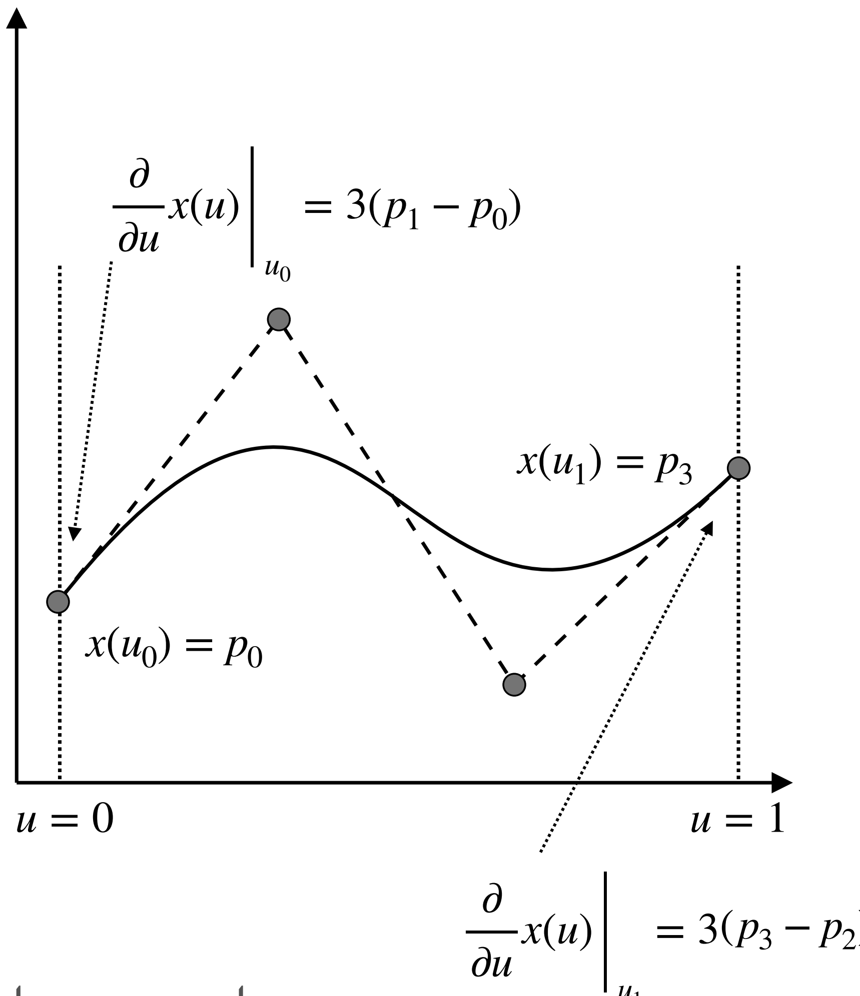
$$x(u_1) = h_1 = p_3$$

$$x'(u_0) = h_2 = 3(p_1 - p_0)$$

$$x'(u_1) = h_3 = 3(p_3 - p_2)$$

$$\mathbf{x}(u) = \mathscr{P}^3(u) \cdot \boldsymbol{\beta}_H \cdot \mathbf{M}_{Z\rightarrow H} \cdot \mathbf{p}$$

Bézier



Note: all the control points are actually points in space, no tangents.

Cubic Bézier

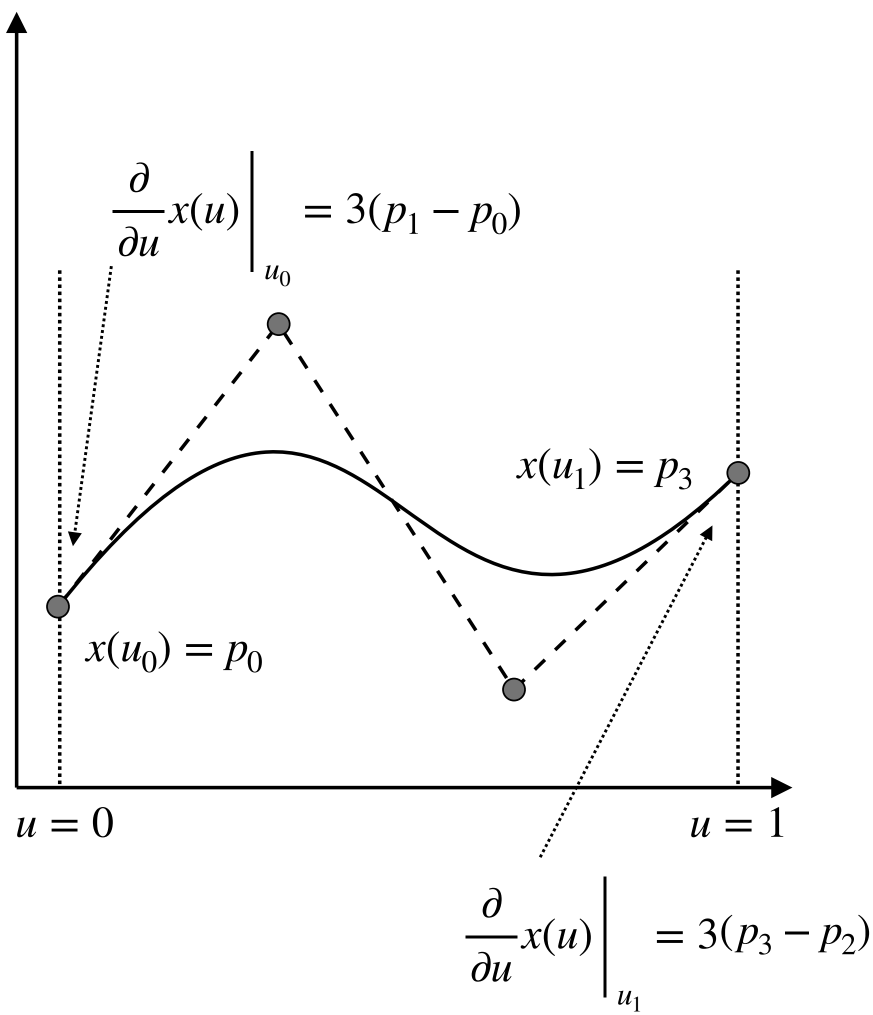
Similar to Hermite, but specify tangents indirectly

$$x(u) = c_0 + c_1 u + c_2 u^2 + c_3 u^3$$

$$x(u_0) = h_0 = p_0$$
$$x(u_1) = h_1 = p_3$$
$$x'(u_0) = h_2 = 3(p_1 - p_0)$$
$$x'(u_1) = h_3 = 3(p_3 - p_2)$$

$$\mathbf{M}_{Z\to H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -3 & 3 & 0 & 0 \\ 0 & 0 & -3 & 3 \end{bmatrix}$$

$$\mathbf{x}(u) = \mathscr{P}^3(u) \cdot \boldsymbol{\beta}_H \cdot \mathbf{M}_{Z\to H} \cdot \mathbf{p}$$



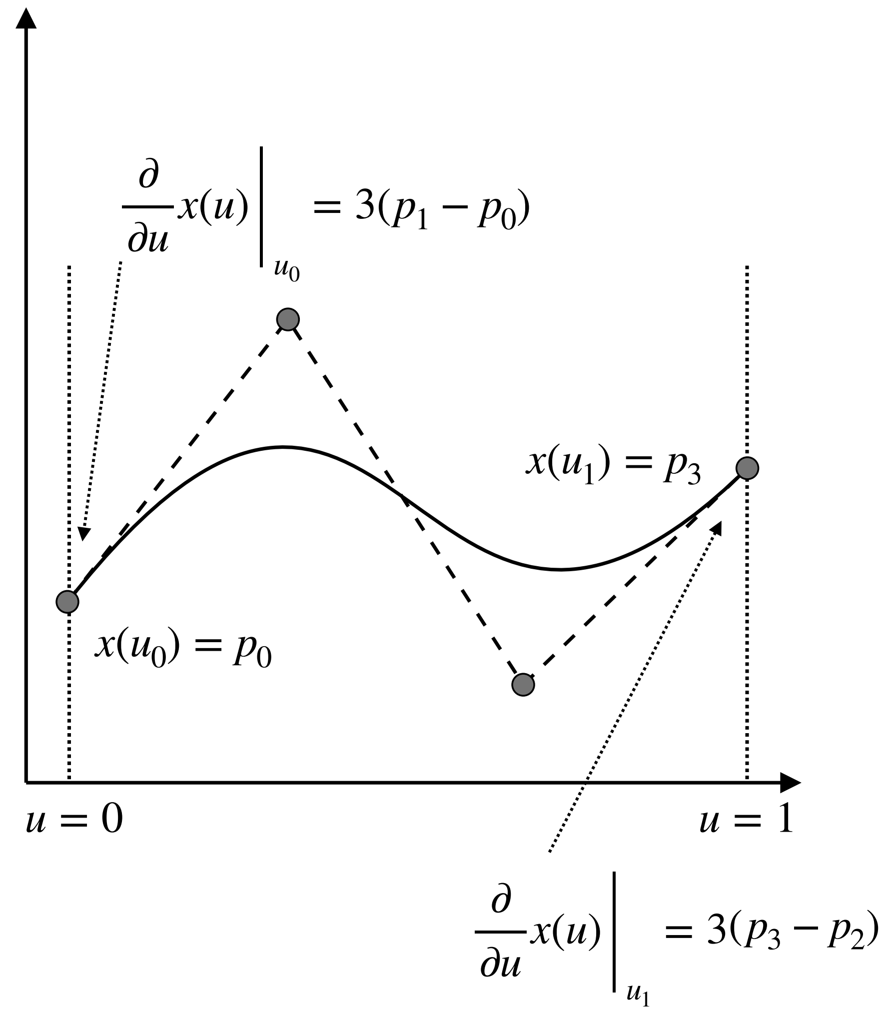
Cubic Bézier

Similar to Hermite, but specify tangents indirectly

$$\mathbf{x}(u) = \mathscr{P}^3(u) \cdot \boldsymbol{\beta}_H \cdot \mathbf{M}_{Z\rightarrow H} \cdot \mathbf{p}$$

$$\mathbf{x}(u) = \mathscr{P}^3(u) \cdot \boldsymbol{\beta}_Z \cdot \mathbf{p}$$

$$\boldsymbol{\beta}_Z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}$$



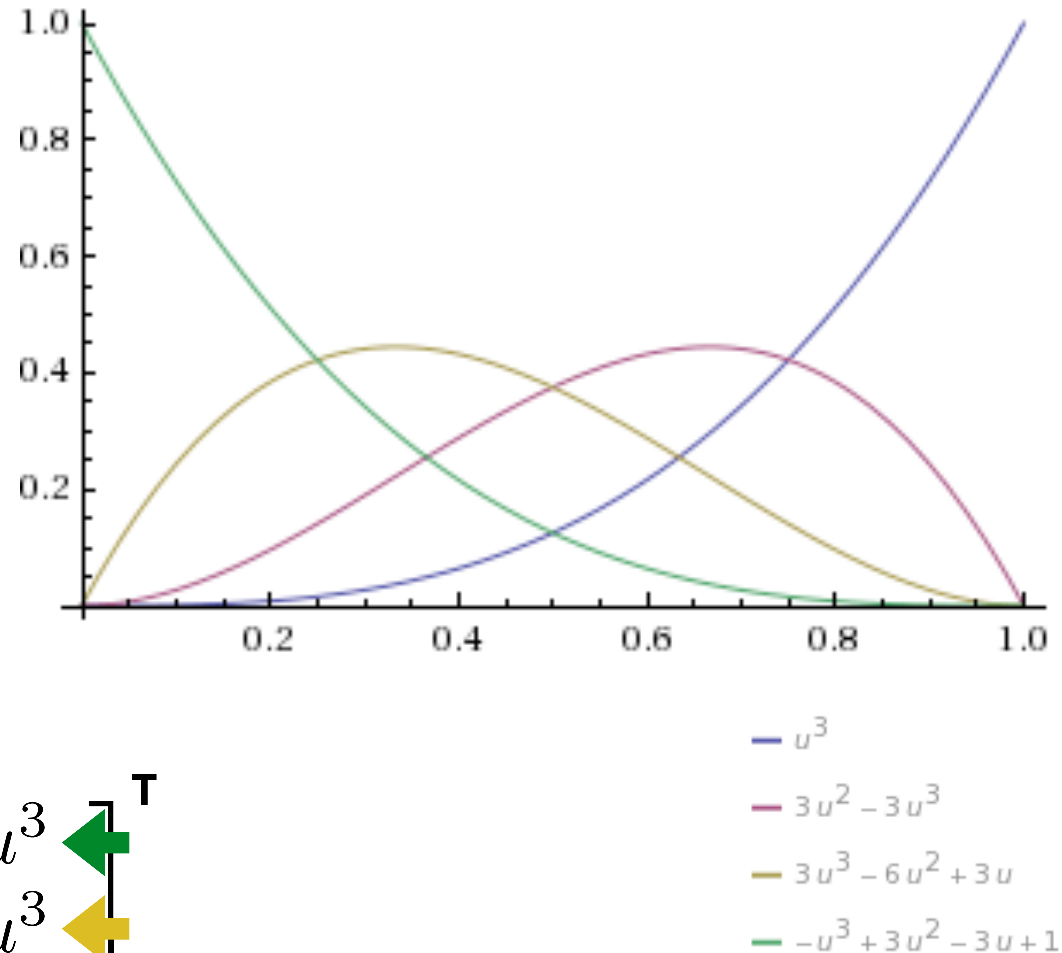
Cubic Bézier

Bézier basis functions

$$\mathbf{x}(u) = \mathscr{P}^3(u) \cdot \boldsymbol{\beta}_Z \cdot \mathbf{p}$$

$$\boldsymbol{\beta}_Z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}$$

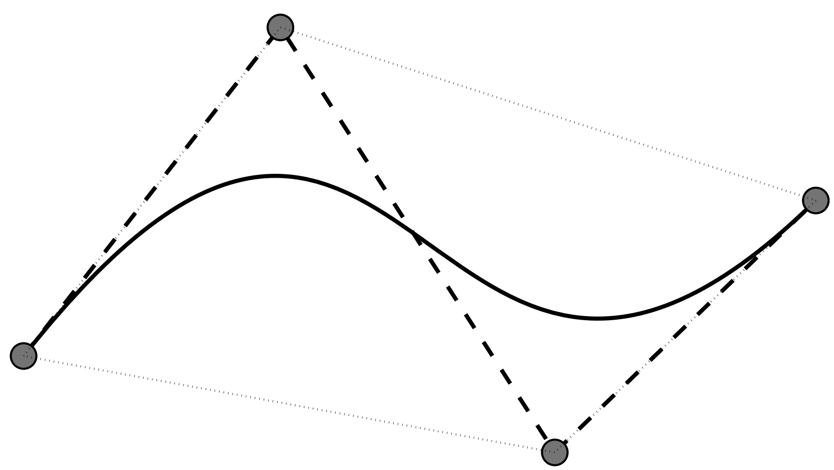
$$\mathbf{X}(u) = \begin{bmatrix} 1 - 3u + 3u^2 - 1u^3 \\ 0 + 3u - 6u^2 + 3u^3 \\ 0 + 0u + 3u^2 - 3u^3 \\ 0 + 0u + 0u^2 + 1u^3 \end{bmatrix}^{\mathsf{T}} \mathbf{p}$$

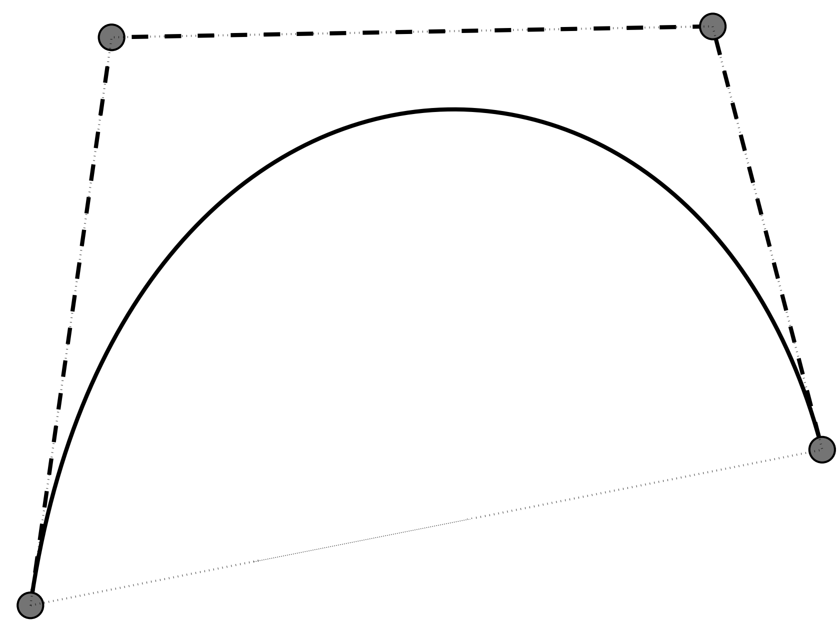


Useful Properties of a Basis

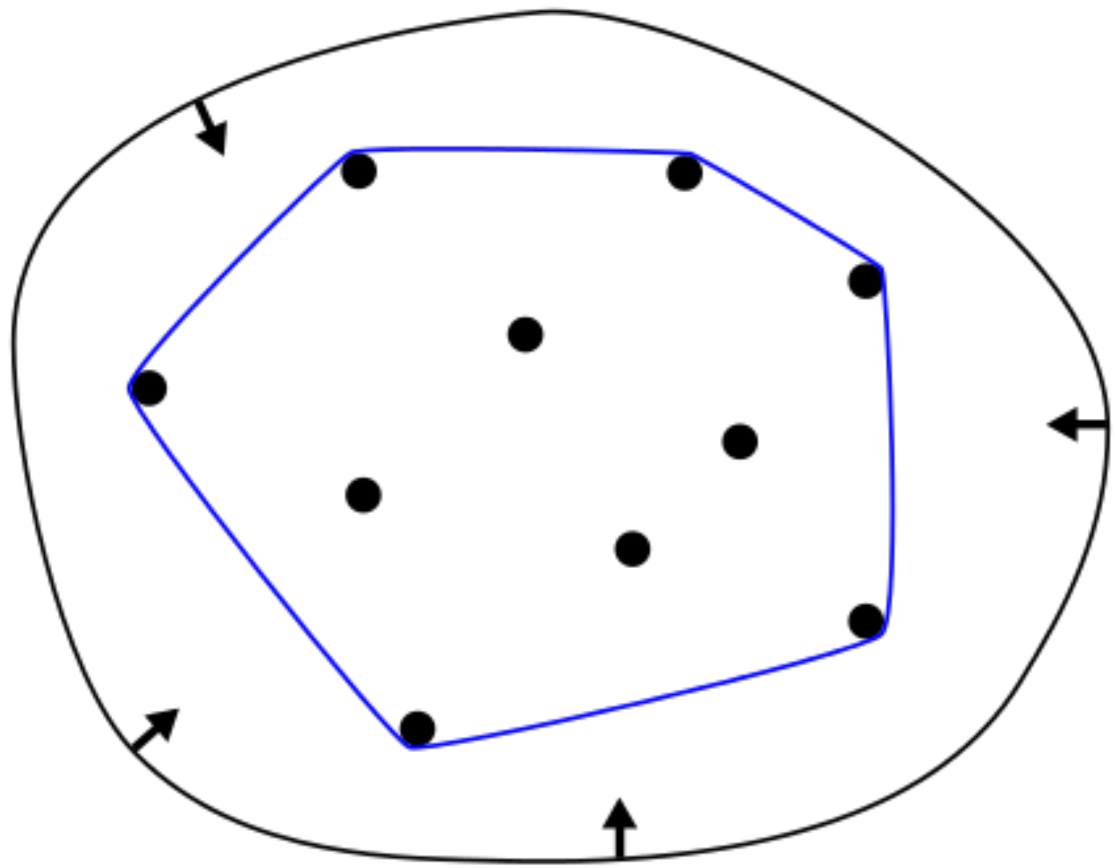
Convex Hull Property

- All points on curve inside convex hull of control points





What's a Convex Hull

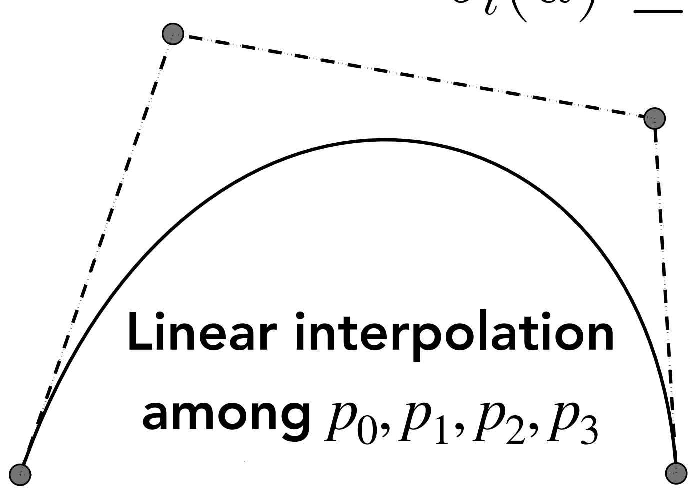


[from Wikipedia]

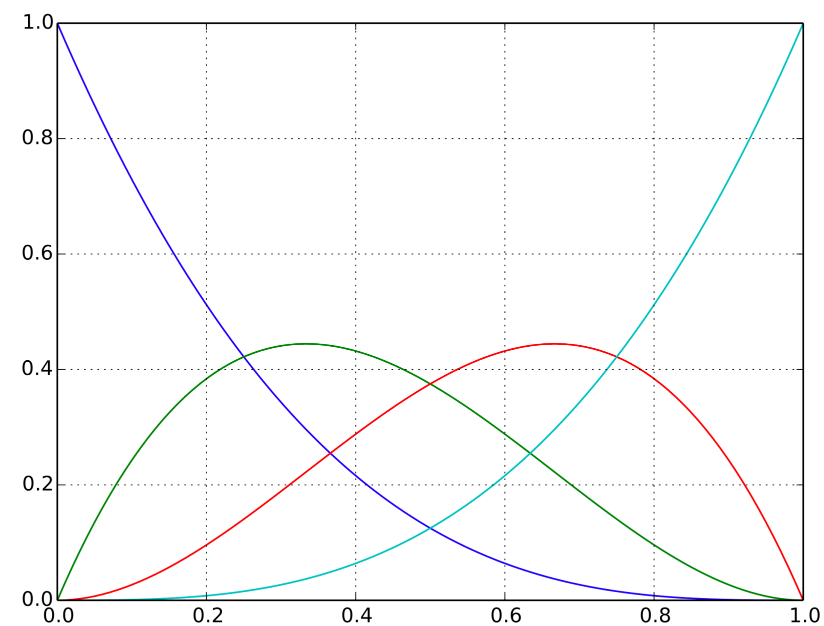
Useful Properties of a Basis

Convex Hull Property

- All points on curve inside convex hull of control points
- Bézier basis has convex hull property!
- Convex hull if the following are true:
  - Basis functions sum to one $$\sum_i b_i(u) = 1$$
  - Basis functions are positive over domain
    $$b_i(u) \geq 0 \qquad \forall u \in \Omega$$



$$p(u) = b_0(u)p_0 + b_1(u)p_1 + b_2(u)p_2 + b_3(u)p_3$$



Useful Properties of a Basis

Invariance under class of transforms

- Transforming curve is same as transforming control points
- Bézier basis invariant w.r.t. *affine* transforms

$$Af(x) = f(Ax)$$ 仿射不变性

$$\mathbf{x}(u) = \mathscr{P}^3(u) \cdot \boldsymbol{\beta}_Z \cdot \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{bmatrix} \quad \Longleftrightarrow \quad \mathbf{T} \cdot \mathbf{x}(u) = \mathscr{P}^3(u) \cdot \boldsymbol{\beta}_Z \cdot \begin{bmatrix} \mathbf{T} \cdot \mathbf{p}_0 \\ \mathbf{T} \cdot \mathbf{p}_1 \\ \mathbf{T} \cdot \mathbf{p}_2 \\ \mathbf{T} \cdot \mathbf{p}_3 \end{bmatrix}$$

Useful Properties of a Basis

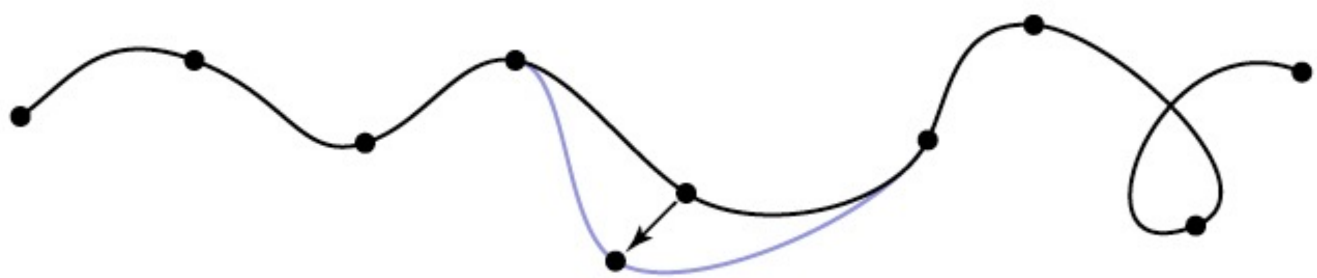
Invariance under class of transforms

- Transforming curve is same as transforming control points
- Bézier basis invariant w.r.t. *affine* transforms
- Bézier NOT invariant for perspective transforms
  - NURBS are invariant w.r.t. perspective transforms

$$\mathbf{x}(u) = \mathscr{P}^3(u) \cdot \boldsymbol{\beta}_Z \cdot \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{bmatrix} \Leftrightarrow \mathbf{T} \cdot \mathbf{x}(u) = \mathscr{P}^3(u) \cdot \boldsymbol{\beta}_Z \cdot \begin{bmatrix} \mathbf{T} \cdot \mathbf{p}_0 \\ \mathbf{T} \cdot \mathbf{p}_1 \\ \mathbf{T} \cdot \mathbf{p}_2 \\ \mathbf{T} \cdot \mathbf{p}_3 \end{bmatrix}$$

More Useful Properties of a Basis

- **Local support**
  - Changing one control point does not change the entire curve



Properties of Bézier Curves

Interpolates endpoints

- For cubic Bézier: $\mathbf{b}(0) = \mathbf{b}_0; \quad \mathbf{b}(1) = \mathbf{b}_3$

Tangent to end segments

- Cubic case: $\mathbf{b}'(0) = 3(\mathbf{b}_1 - \mathbf{b}_0); \quad \mathbf{b}'(1) = 3(\mathbf{b}_3 - \mathbf{b}_2)$

Affine transformation property

- Transform curve by transforming control points

Convex hull property

- Curve is within convex hull of control points

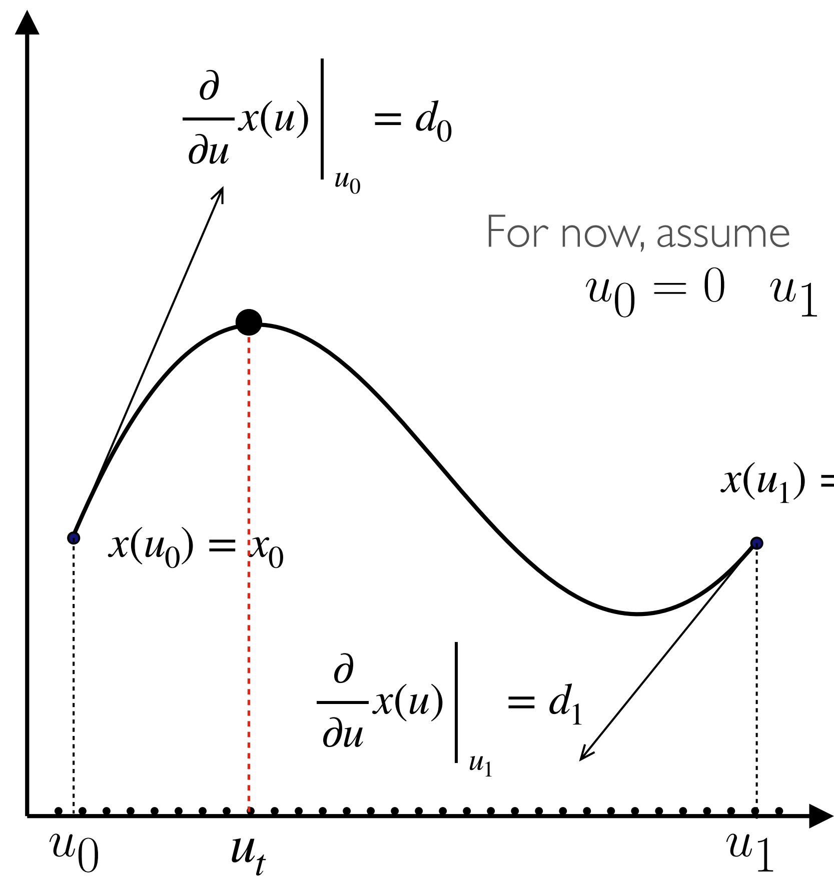
Well behaved

Local control (when joining)

Evaluating Bézier Curves
Matrix Formula

Bézier Curve – Matrix Formula

$$\mathbf{x}(u) = \mathscr{P}^3(u) \cdot \boldsymbol{\beta}_Z \cdot \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{bmatrix} \qquad \boldsymbol{\beta}_Z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}$$

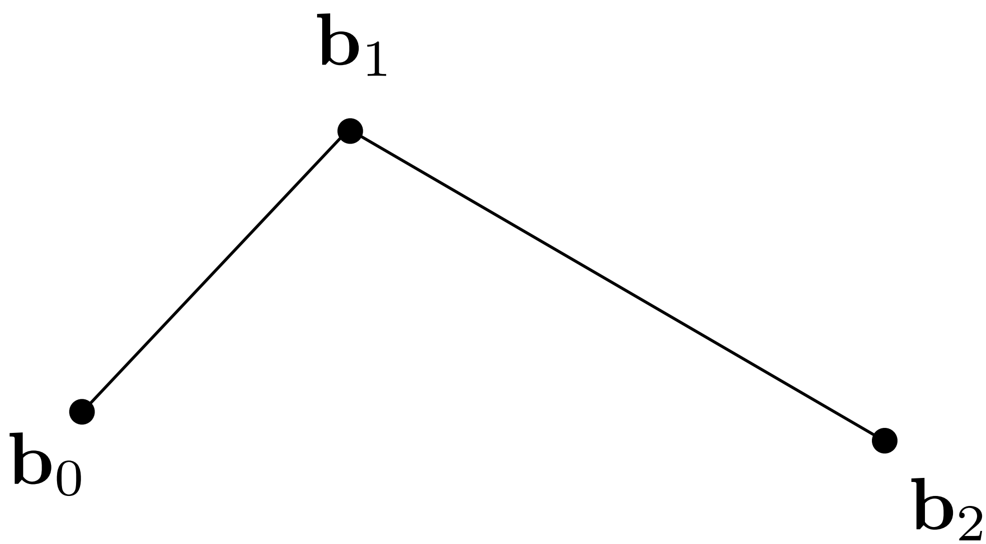


For now, assume
$u_0 = 0 \quad u_1 = 1$

Evaluating Bézier Curves
De Casteljau Algorithm

Bézier Curves – de Casteljau Algorithm

Consider three points (quadratic Bezier)

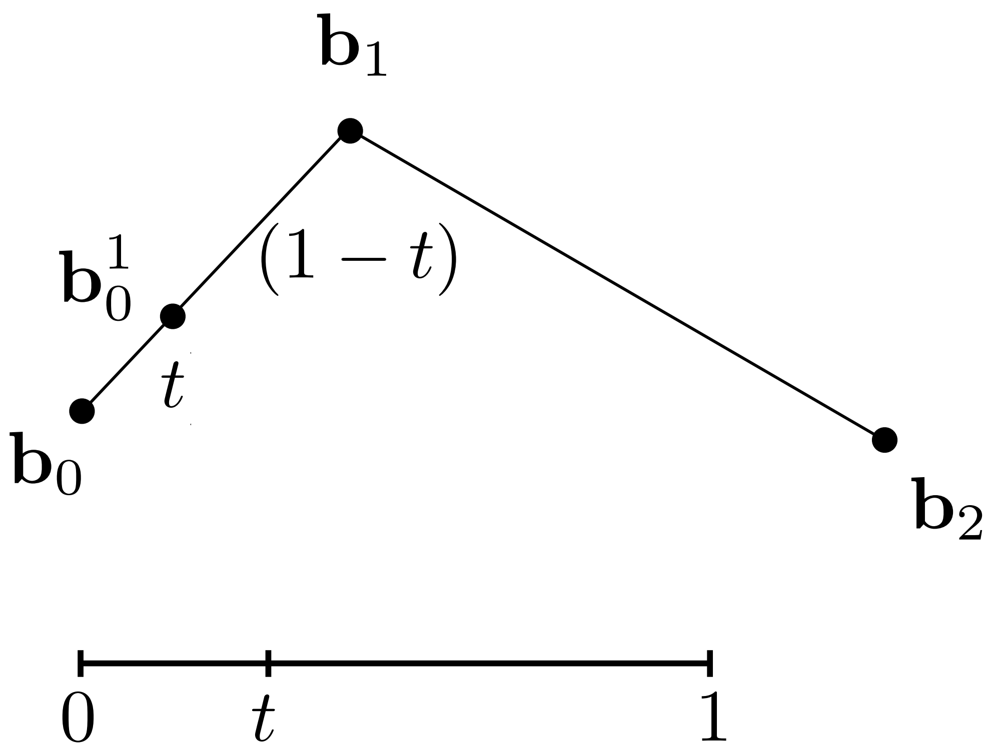


Pierre Bézier
1910 – 1999

Paul de Casteljau
1930 – 2022

Bézier Curves – de Casteljau Algorithm

Insert a point using linear interpolation

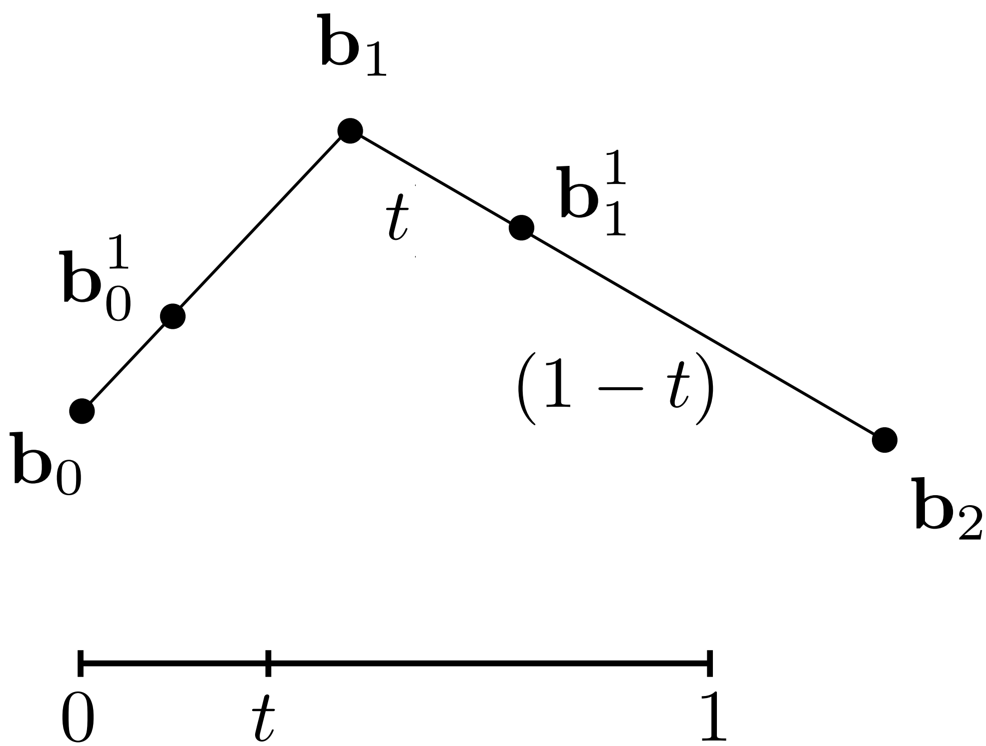


Pierre Bézier
1910 – 1999

Paul de Casteljau
1930 – 2022

Bézier Curves – de Casteljau Algorithm

Insert on both edges

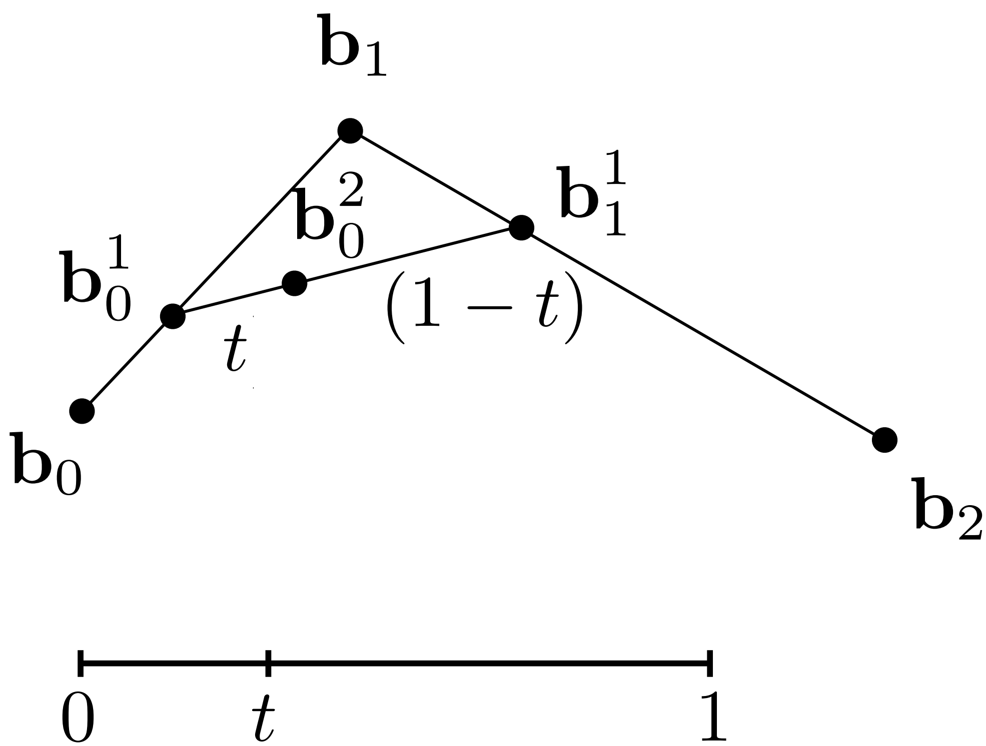


Pierre Bézier
1910 – 1999

Paul de Casteljau
1930 – 2022

Bézier Curves – de Casteljau Algorithm

Repeat recursively

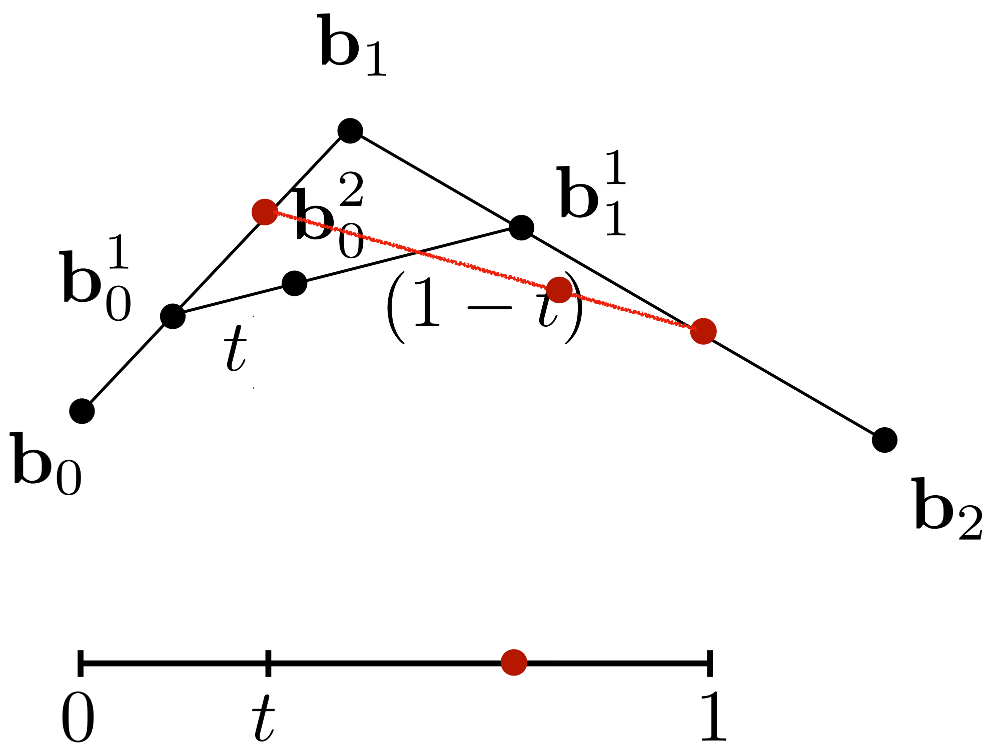


Pierre Bézier
1910 – 1999

Paul de Casteljau
1930 – 2022

Bézier Curves – de Casteljau Algorithm

Repeat recursively

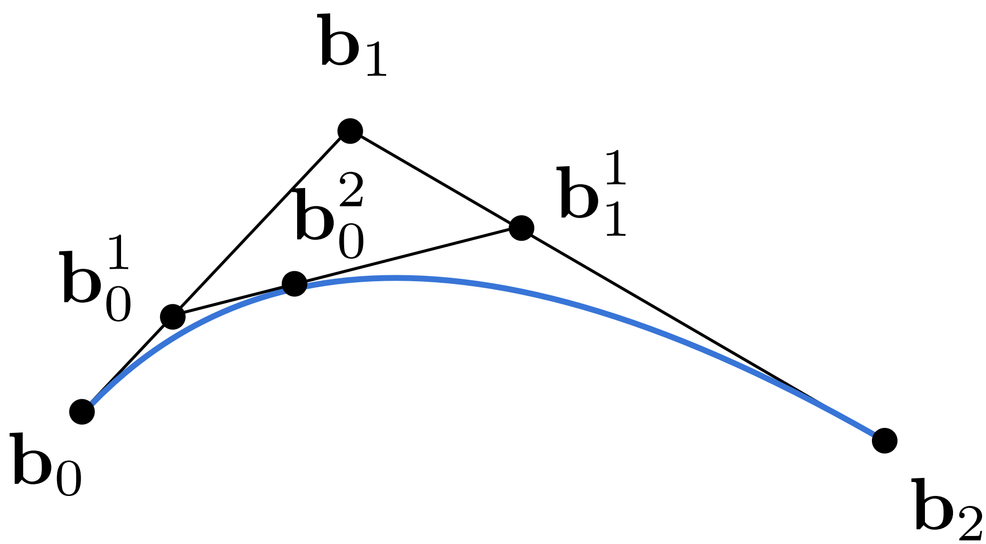


Pierre Bézier
1910 – 1999

Paul de Casteljau
1930 – 2022

Bézier Curves – de Casteljau Algorithm

Algorithm defines the curve



Pierre Bézier
1910 – 1999

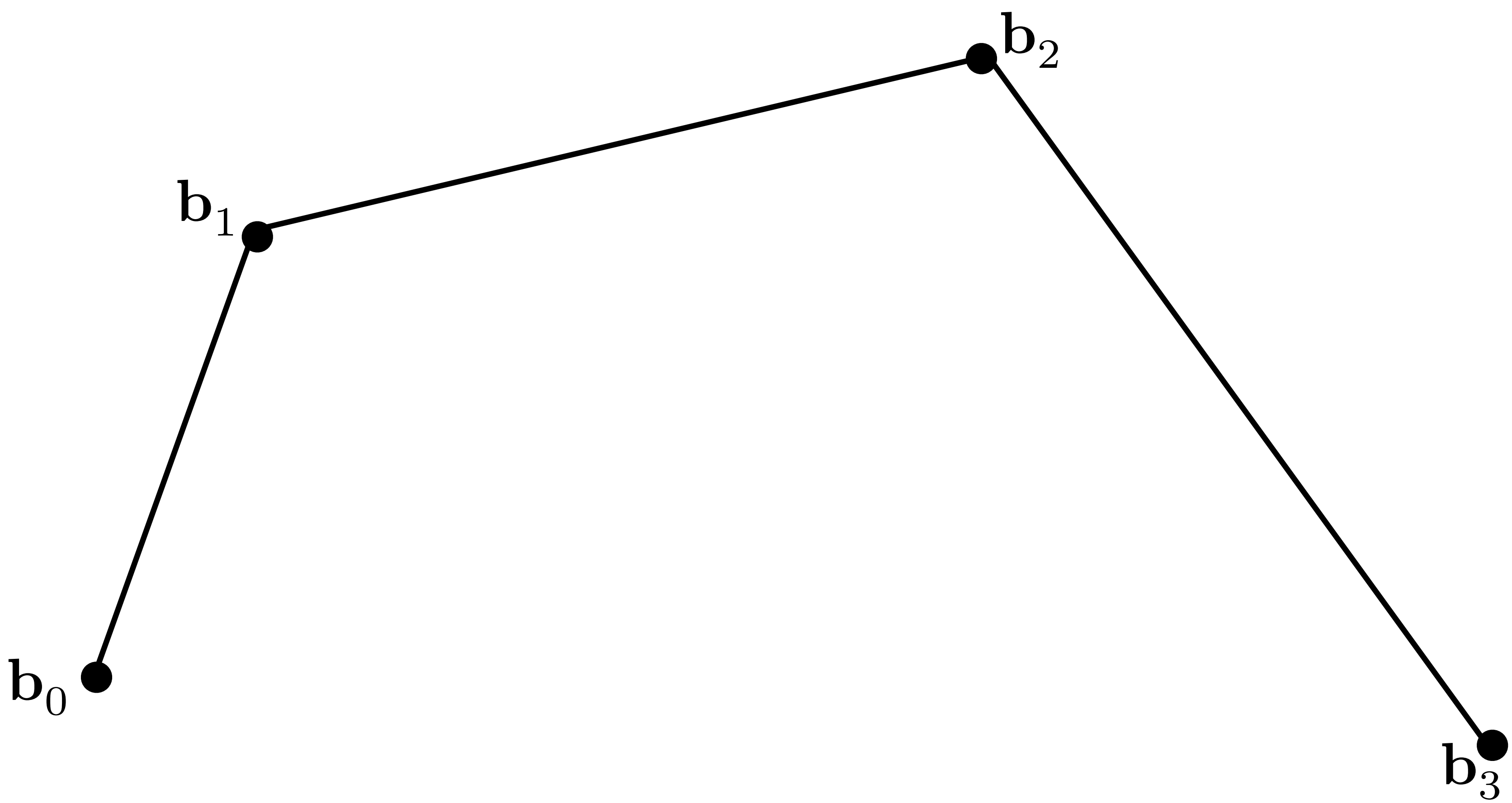
Paul de Casteljau
1930 – 2022

"Corner cutting" recursive subdivision

Successive linear interpolation

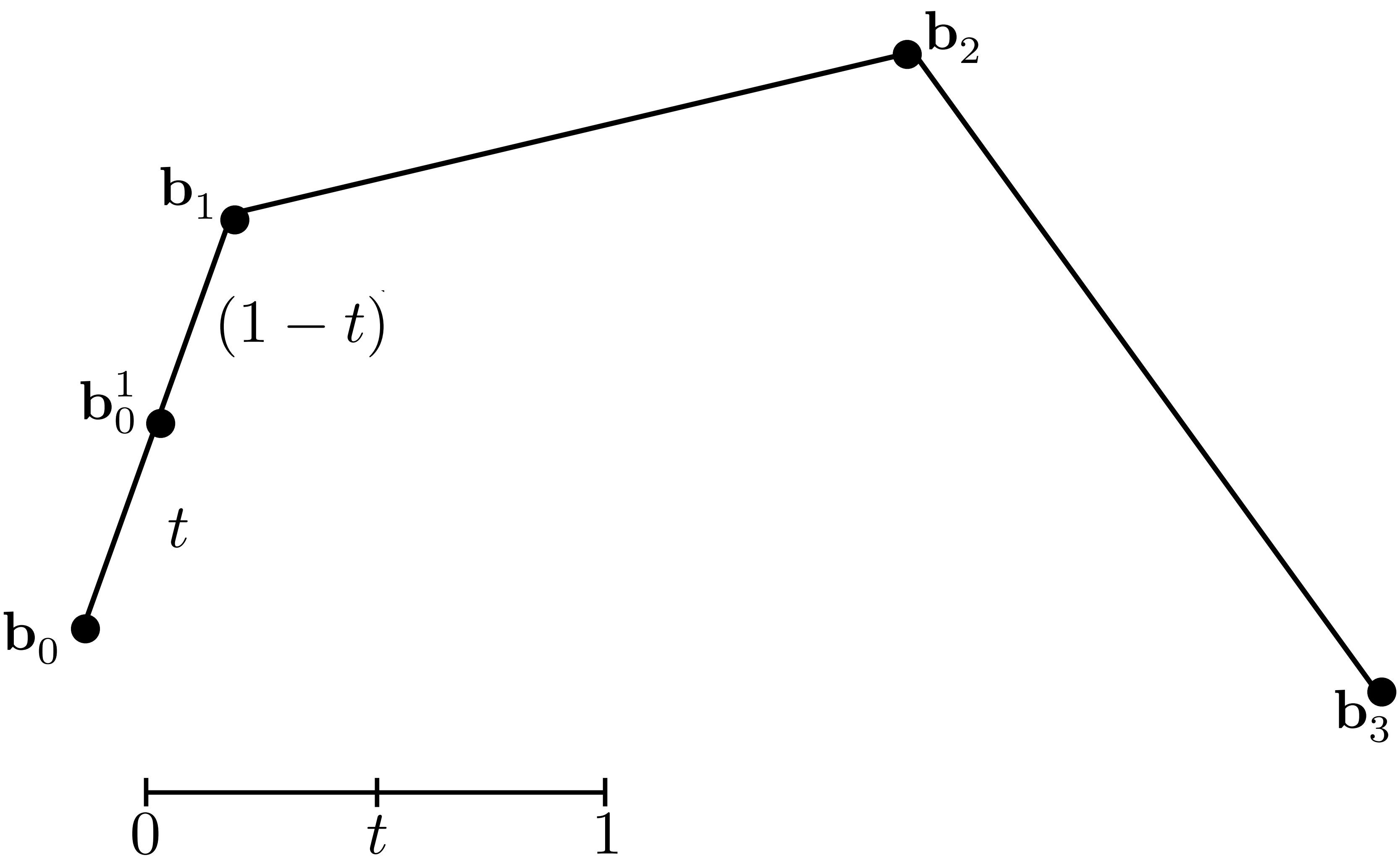
Cubic Bézier Curve – de Casteljau

Consider four points
Same recursive linear interpolations



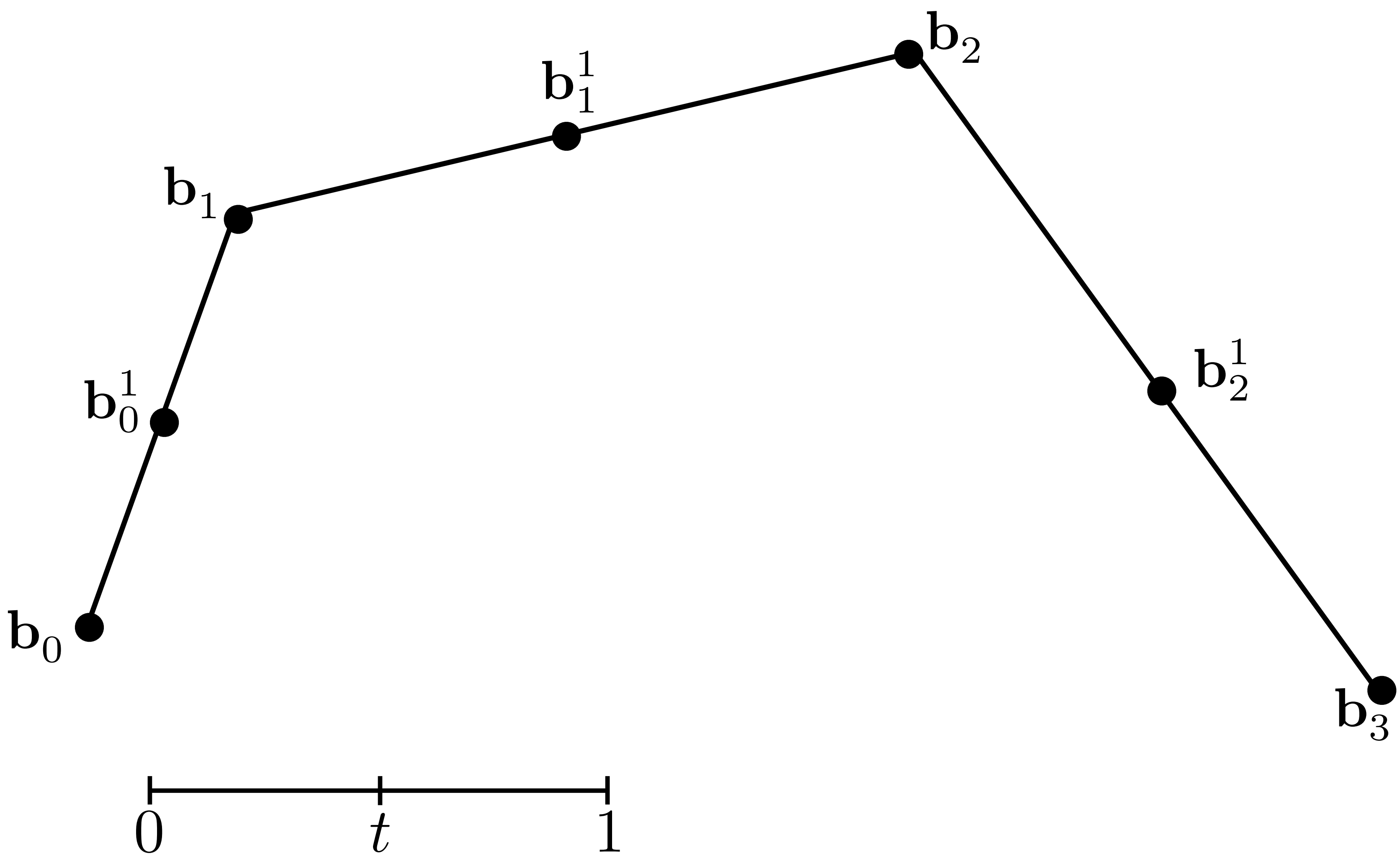
Cubic Bézier Curve – de Casteljau

Consider four points
Same recursive linear interpolations



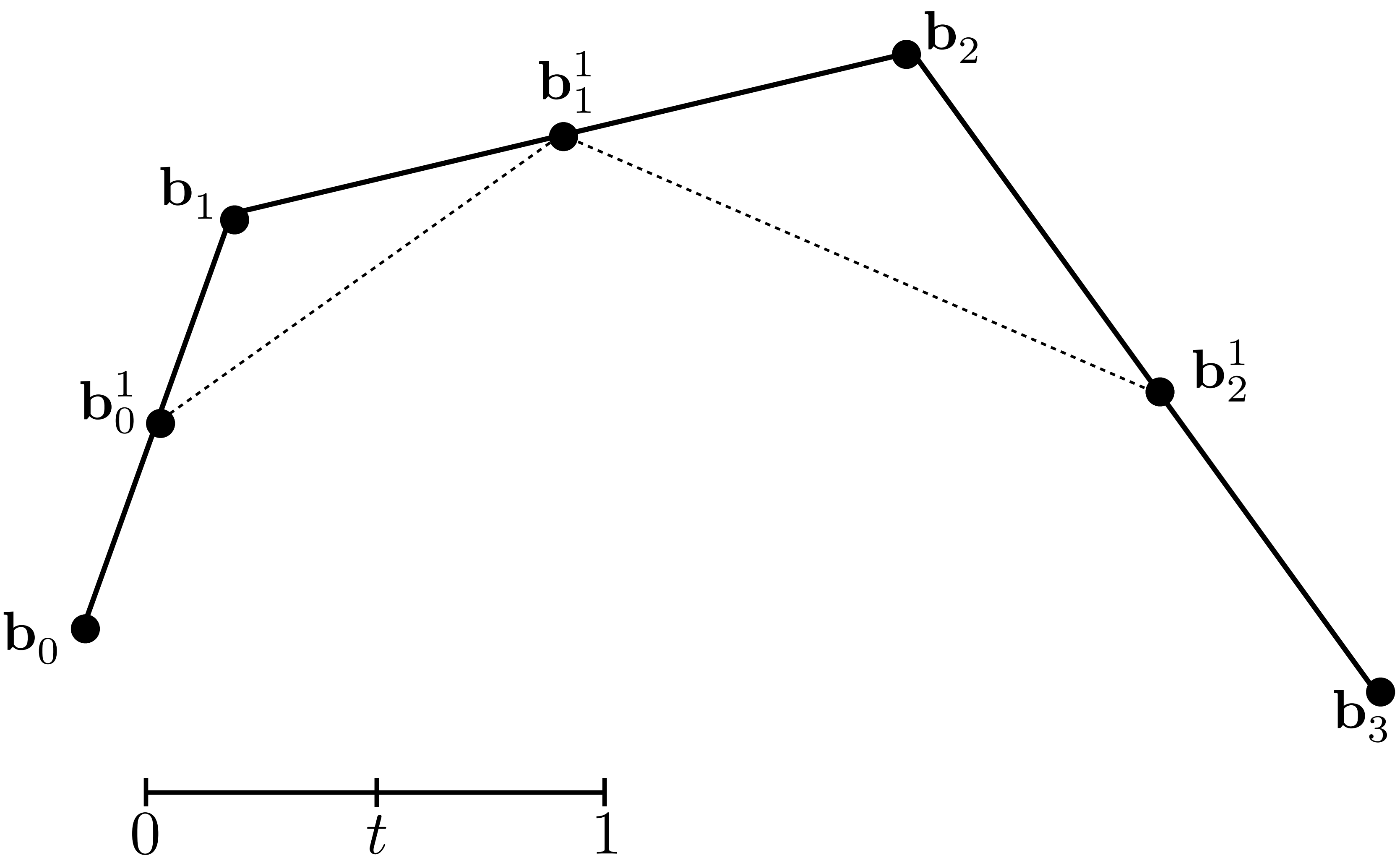
Cubic Bézier Curve – de Casteljau

Consider four points
Same recursive linear interpolations



Cubic Bézier Curve – de Casteljau

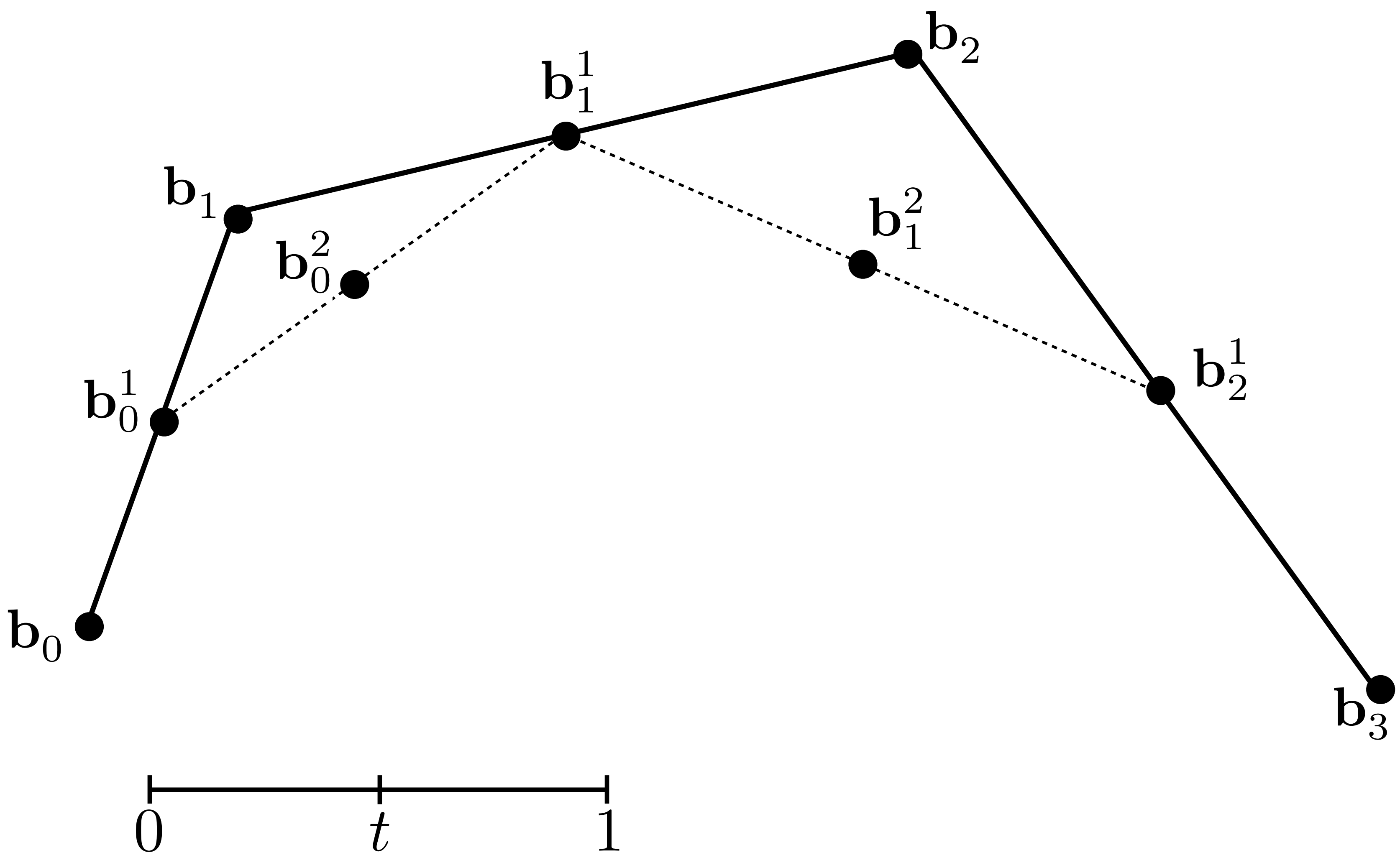
Consider four points
Same recursive linear interpolations



Cubic Bézier Curve – de Casteljau

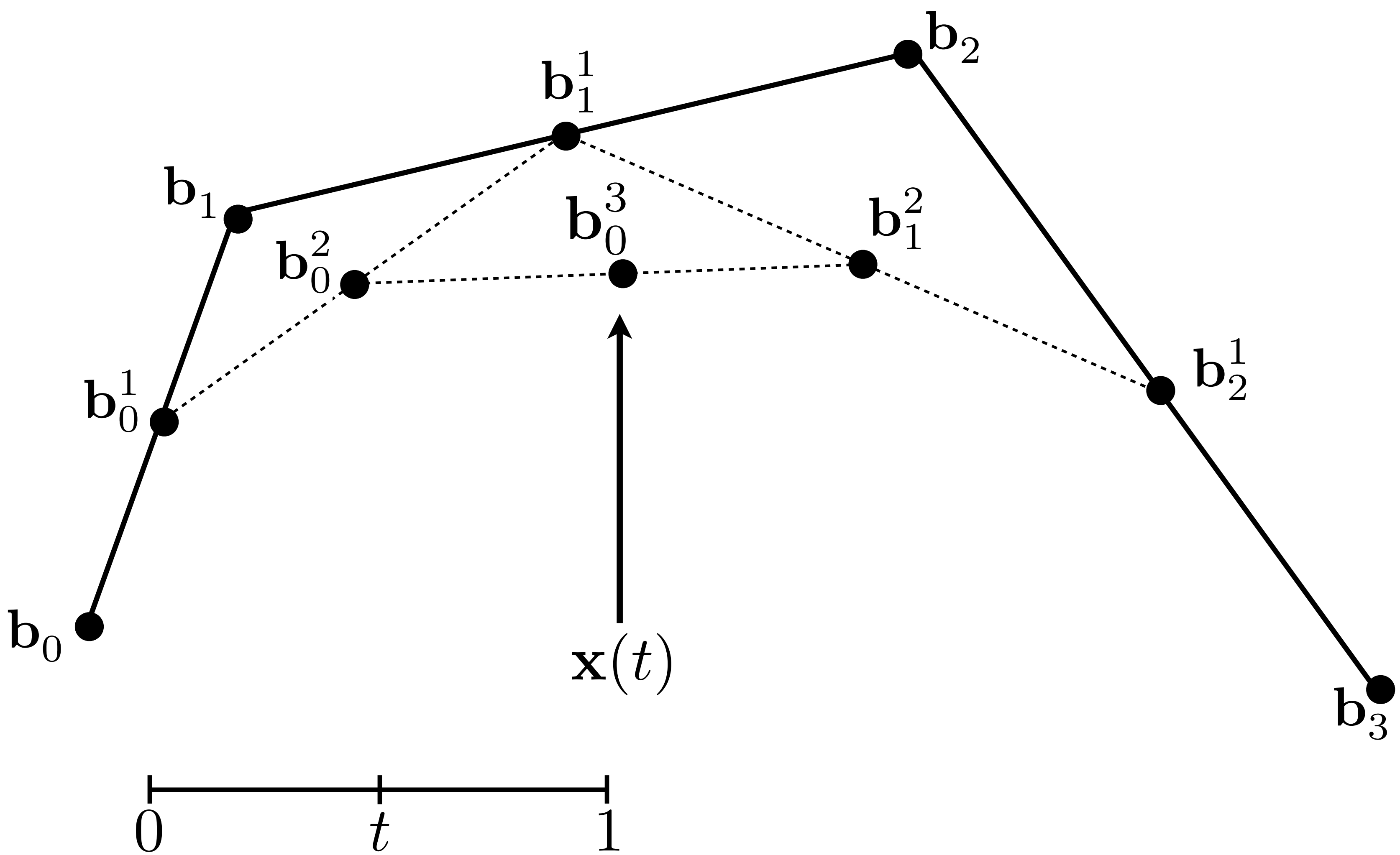
Consider four points

Same recursive linear interpolations



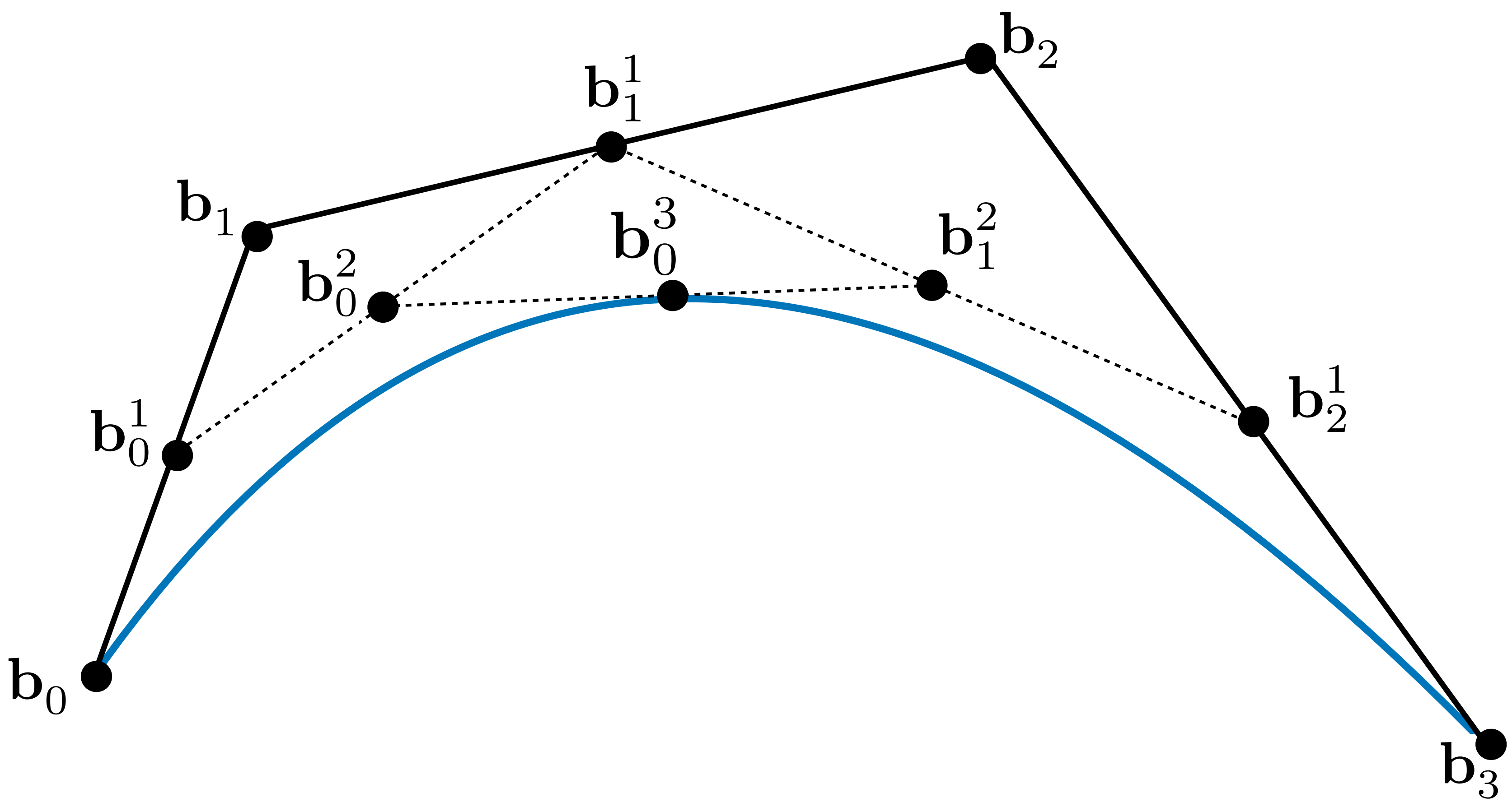
Cubic Bézier Curve – de Casteljau

Consider four points
Same recursive linear interpolations

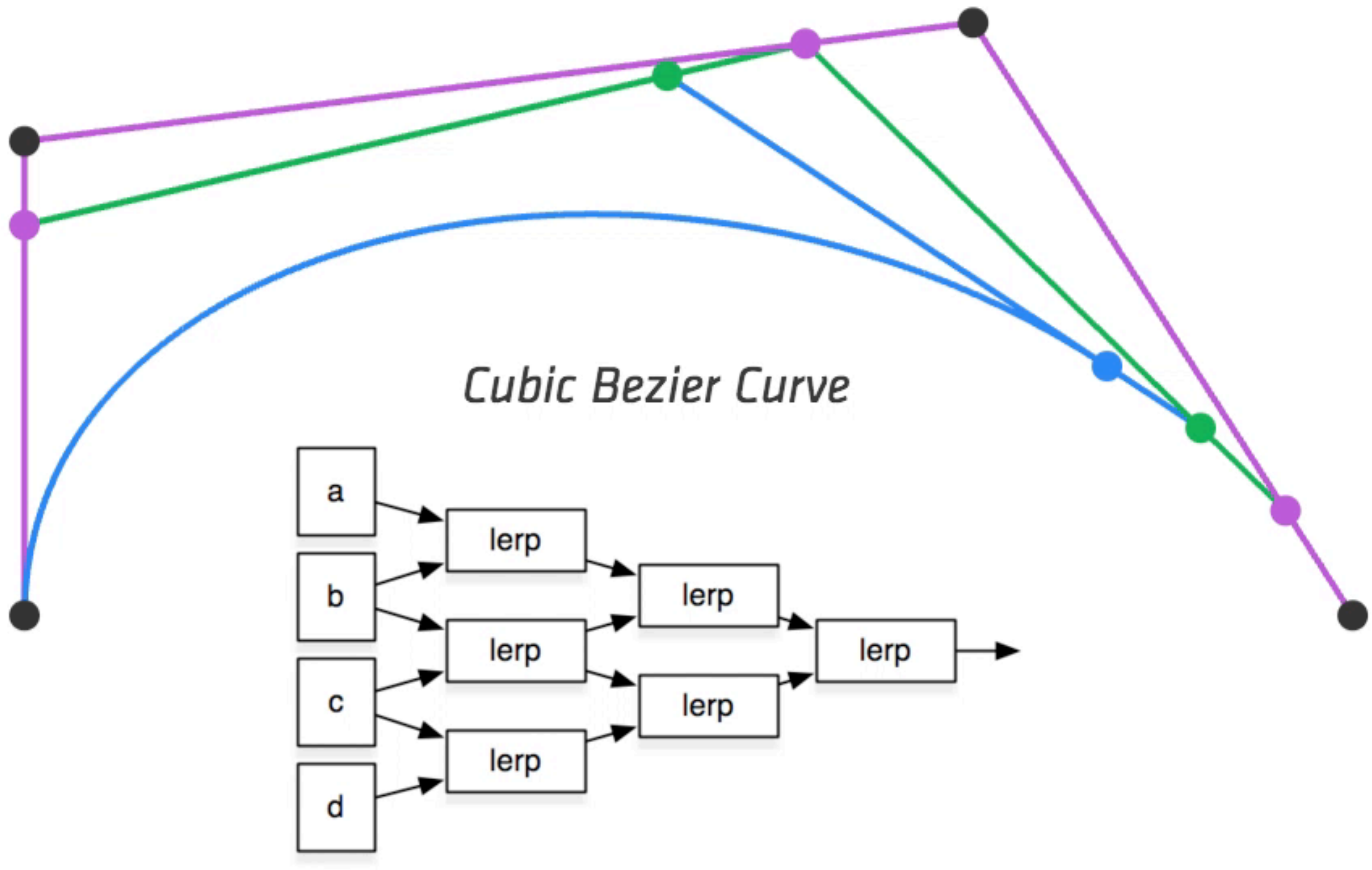


Cubic Bézier Curve – de Casteljau

Consider four points
Same recursive linear interpolations



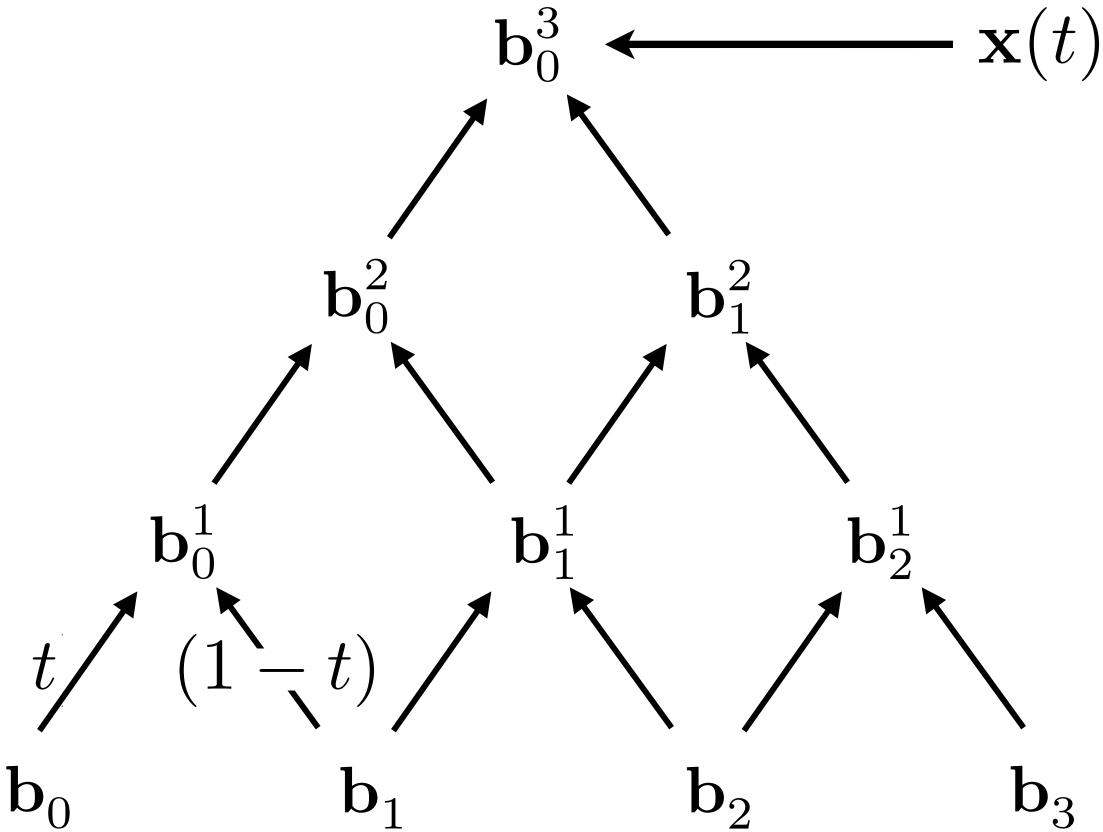
Visualizing de Casteljau Algorithm

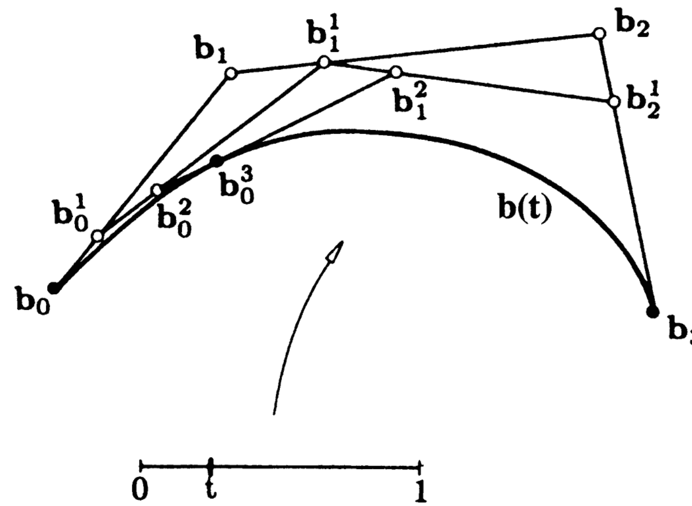


Animation: Steven Wittens, Making Things with Maths, http://acko.net

Bézier Curve – Algebraic Formula

de Casteljau algorithm gives a pyramid of coefficients



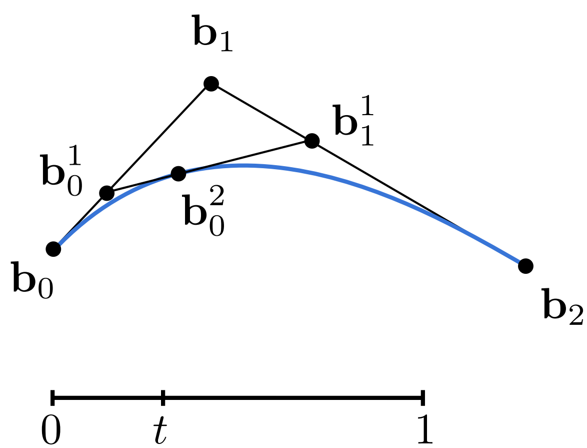


Every rightward arrow is multiplication by t,

Every leftward arrow by (1–t)

Bézier Curve – Algebraic Formula

Example: quadratic Bézier curve from three points



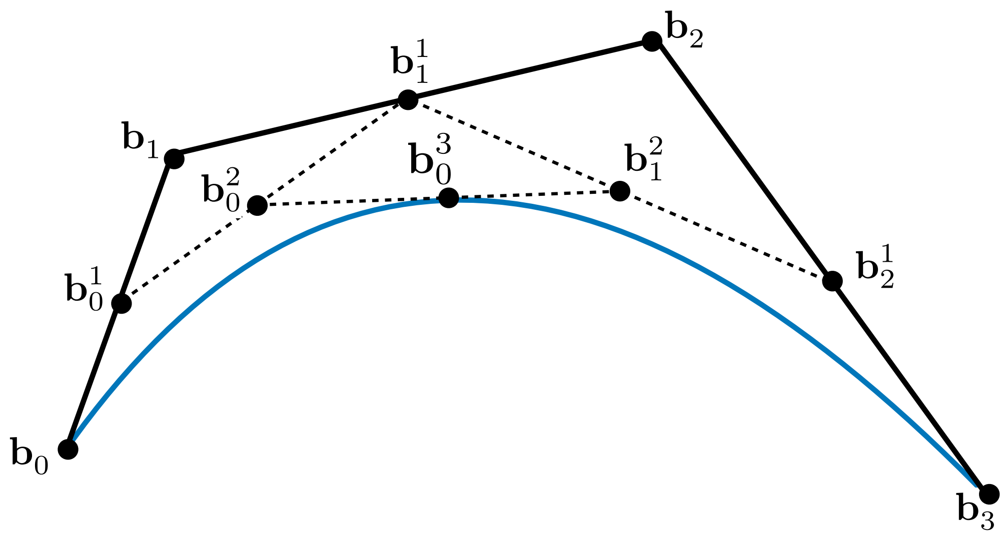
$$\mathbf{b}_0^1(t) = (1-t)\mathbf{b}_0 + t\mathbf{b}_1$$

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

$$\mathbf{b}_0^2(t) = (1-t)\mathbf{b}_0^1 + t\mathbf{b}_1^1$$

$$\mathbf{b}_0^2(t) = (1-t)^2\mathbf{b}_0 + 2t(1-t)\mathbf{b}_1 + t^2\mathbf{b}_2$$

Bézier Curve – Algebraic Formula



$$\mathbf{b}_0^3 = \mathbf{b}_0\,(1-t)^3 + \mathbf{b}_1\,3t(1-t)^2 + \mathbf{b}_2\,3t^2(1-t) + \mathbf{b}_3\,t^3$$

$$\mathbf{X}(u) = \begin{bmatrix} 1 - 3u + 3u^2 - 1u^3 \\ 0 + 3u - 6u^2 + 3u^3 \\ 0 + 0u + 3u^2 - 3u^3 \\ 0 + 0u + 0u^2 + 1u^3 \end{bmatrix}^{\mathsf{T}} \mathbf{p}$$

Bézier Curve - General Algebraic Formula

Bernstein form of a Bézier curve of order n:

$$\mathbf{b}^n(t) = \mathbf{b}_0^n(t) = \sum_{j=0}^{n} \mathbf{b}_j B_j^n(t)$$

Bézier curve order n
(vector polynomial of degree n)

Bernstein polynomial
(scalar polynomial of degree n)

Bézier control points
(vector in R^N)

Bernstein polynomials:

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

Bézier Curve - Algebraic Formula: Example

Bernstein form of a Bézier curve of order n:

$$\mathbf{b}^n(t) = \sum_{j=0}^{n} \mathbf{b}_j B_j^n(t)$$

Example: assume n = 3 and we are in R^3

i.e. we could have control points in 3D such as:

$$\mathbf{b}_0 = (0,2,3),\ \mathbf{b}_1 = (2,3,5),\ \mathbf{b}_2 = (6,7,9),\ \mathbf{b}_3 = (3,4,5)$$

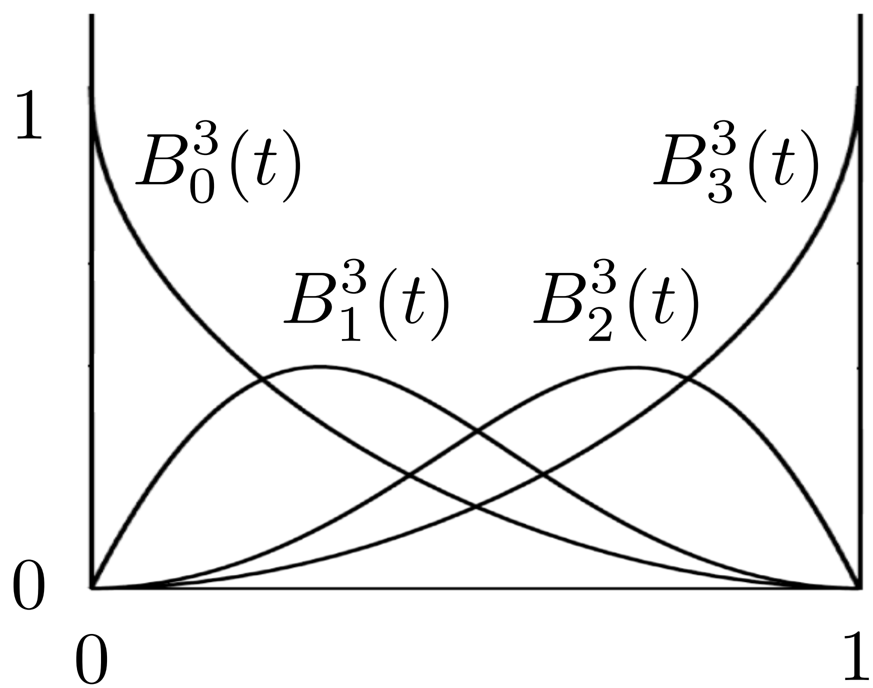
These points define a Bezier curve in 3D that is a cubic polynomial in t:

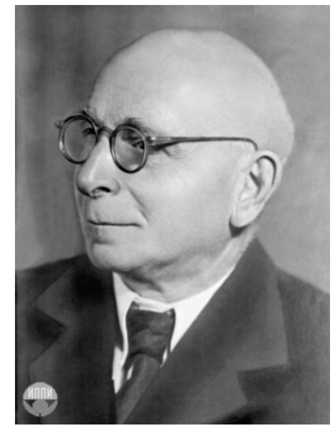
$$\mathbf{b}^n(t) = \ \mathbf{b}_0\,(1-t)^3 + \mathbf{b}_1\,3t(1-t)^2 + \mathbf{b}_2\,3t^2(1-t) + \mathbf{b}_3\,t^3$$

Cubic Bézier Basis Functions

Bernstein Polynomials

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

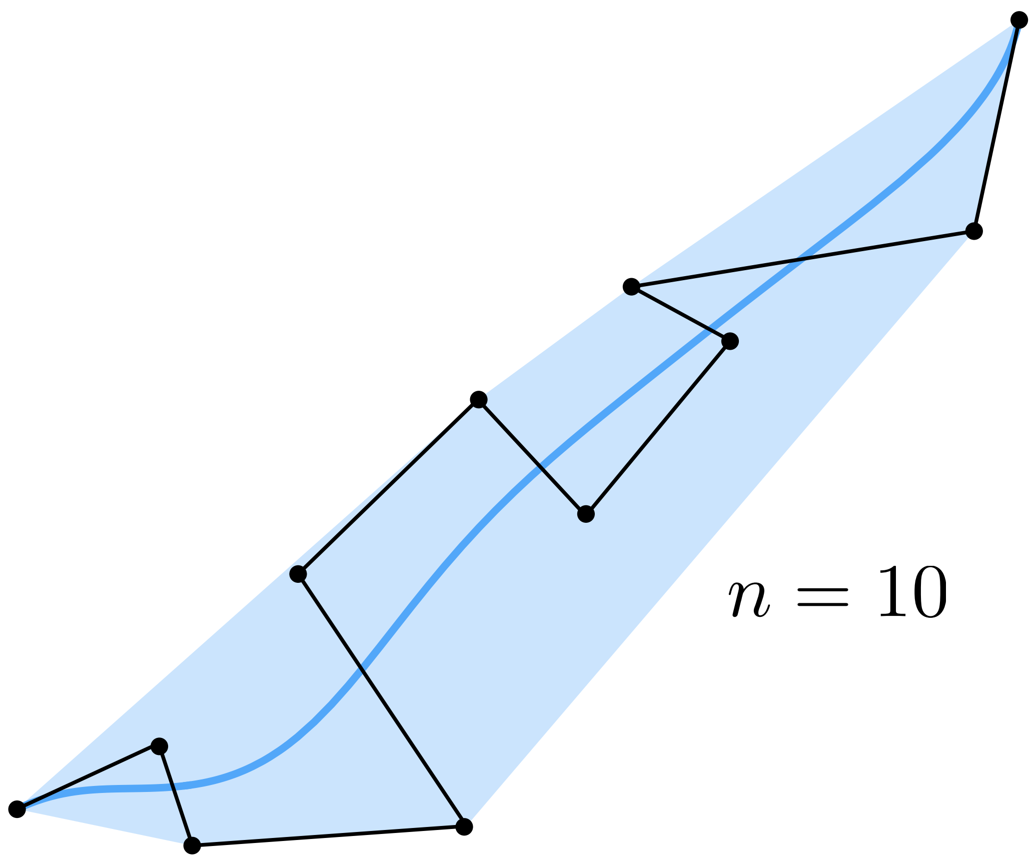




Sergei N. Bernstein
1880 – 1968

Piecewise Bézier Curves
(Bézier Spline)

Higher Orders Bézier Curves?



$n = 10$

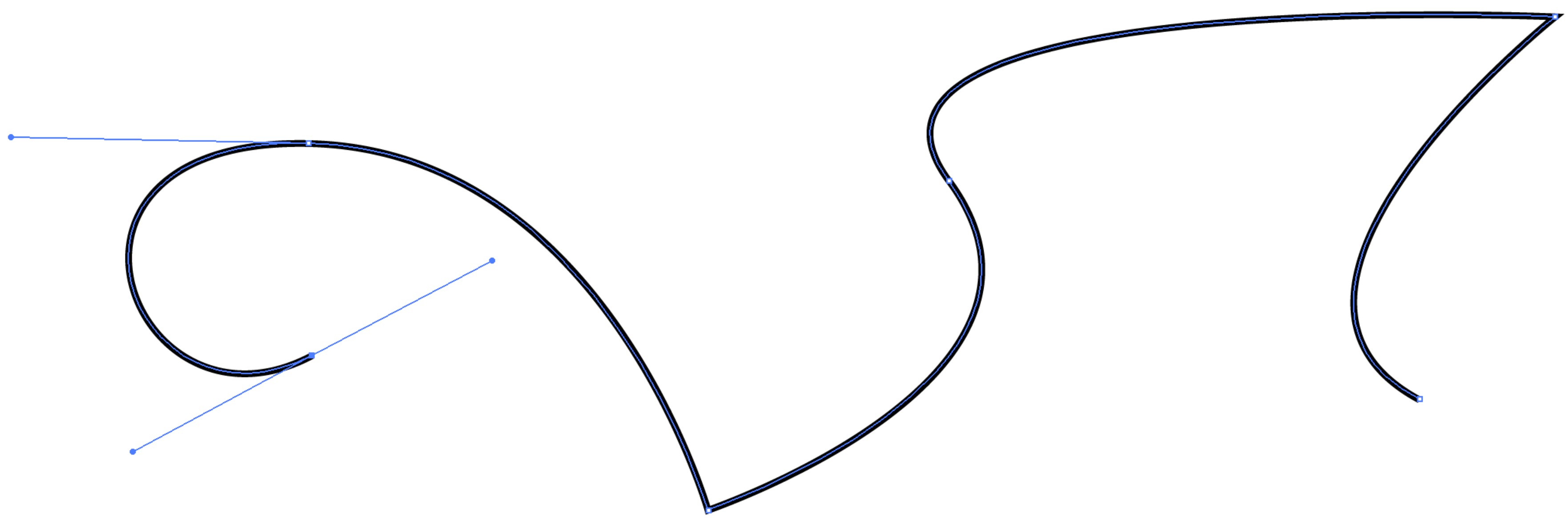
Very hard to control!

Uncommon

Piecewise Bézier Curves

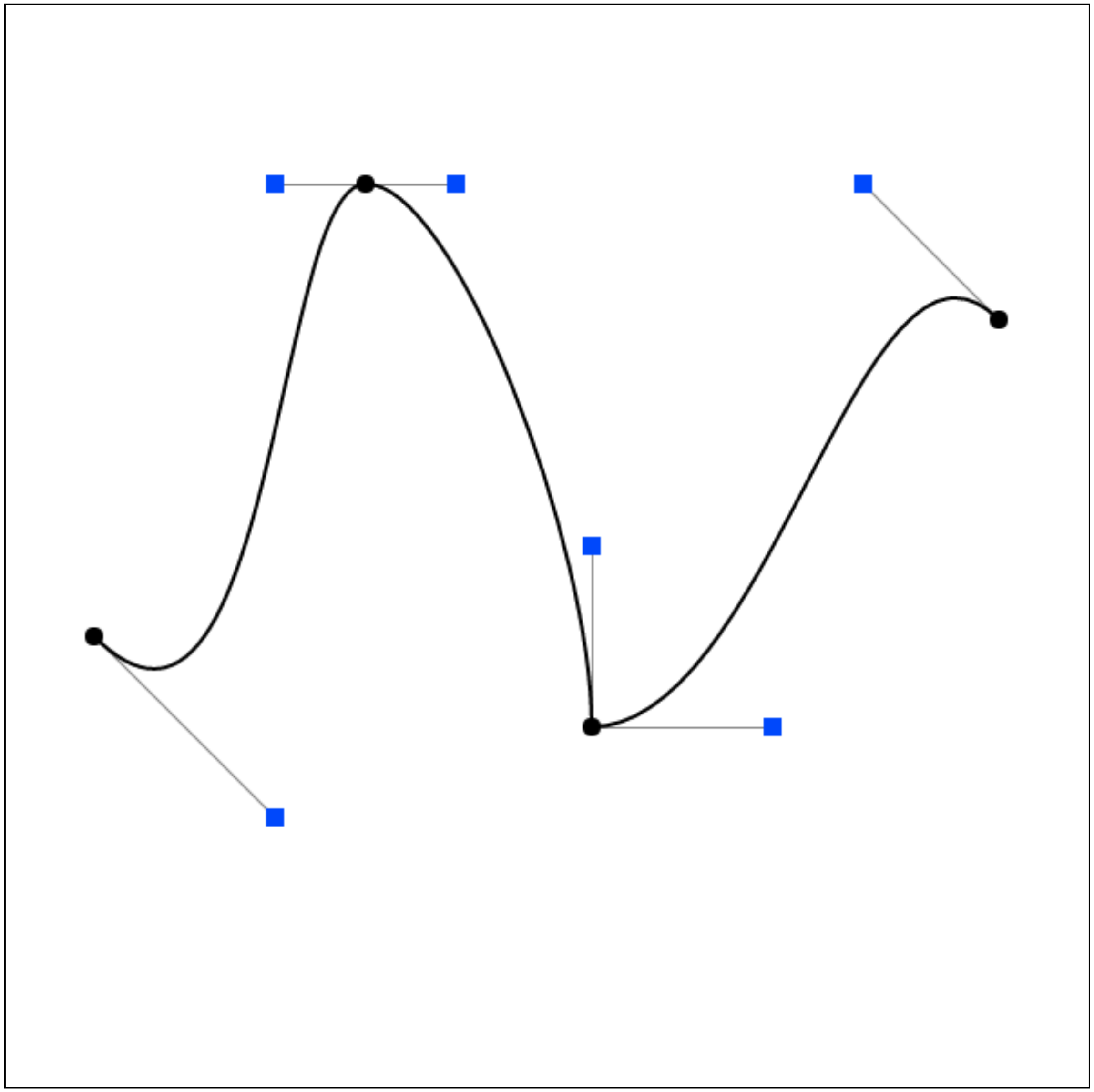
Instead, chain many low-order Bézier curve

Piecewise cubic Bézier the most common technique



Widely used (fonts, paths, Illustrator, Keynote, …)

Demo – Piecewise Cubic Bézier Curve



David Eck, http://math.hws.edu/eck/cs424/notes2013/canvas/bezier.html

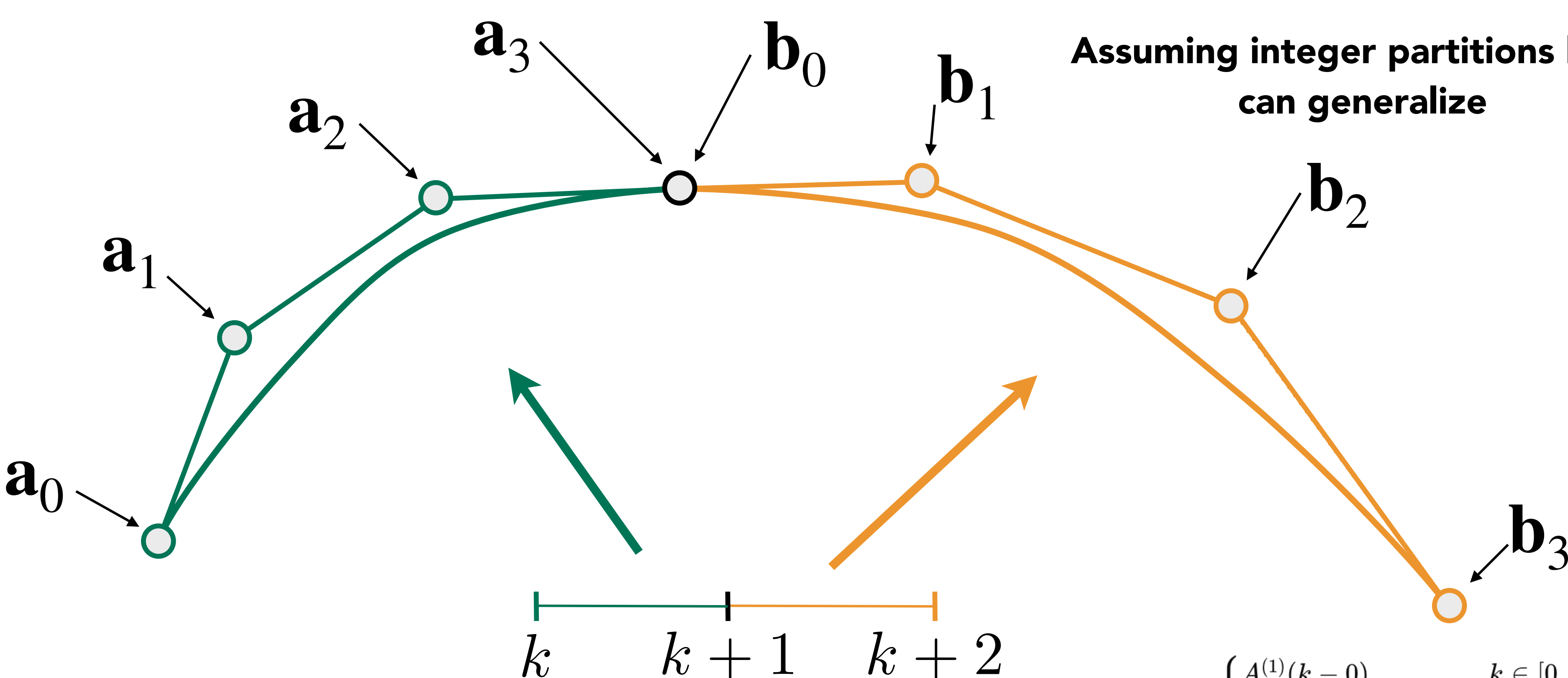
Piecewise Bézier Curve – Continuity

Two Bézier curves

$$\mathbf{a} : [k, k+1] \to \mathbb{R}^N$$

$$\mathbf{b} : [k+1, k+2] \to \mathbb{R}^N$$

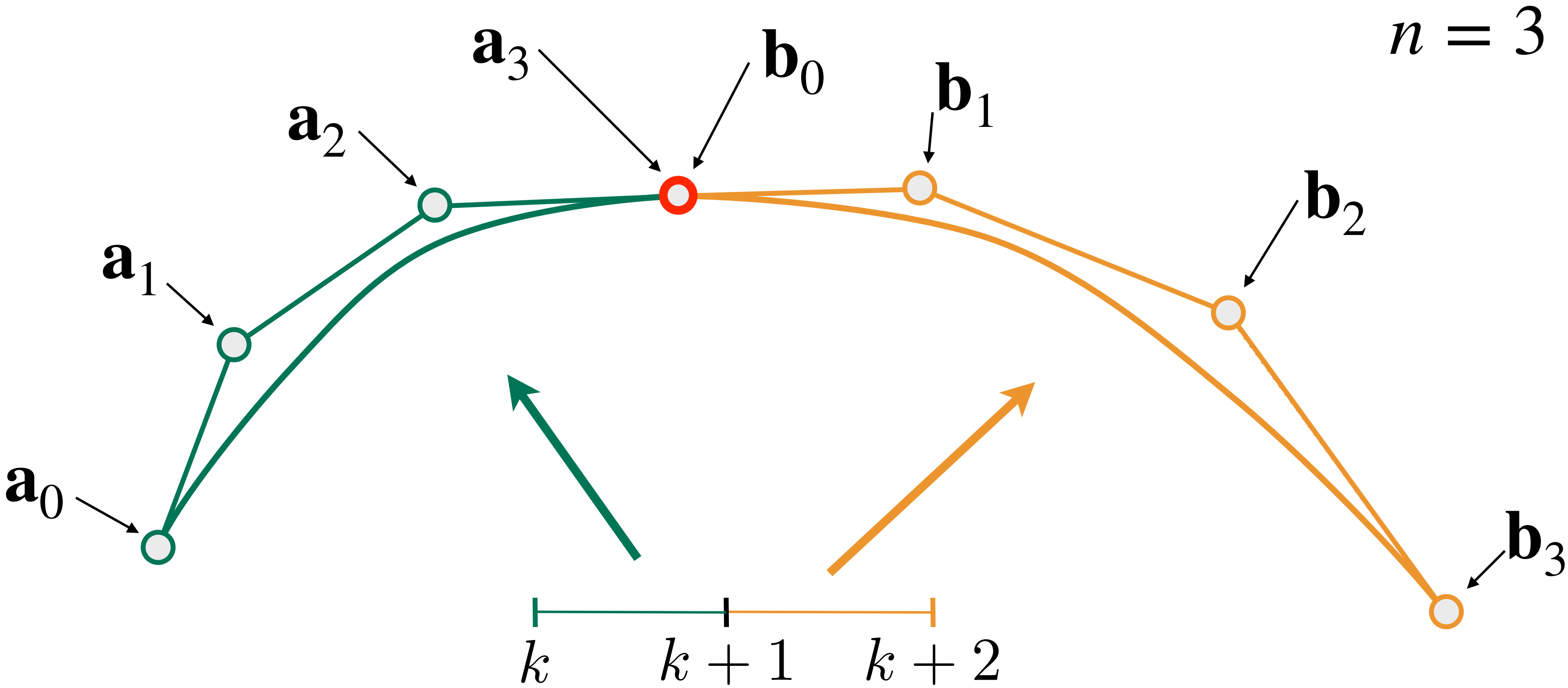
Assuming integer partitions here, can generalize



Definition $$S(k) = \begin{cases} A^{(1)}(k-0), & k \in [0,1], \\ A^{(2)}(k-1), & k \in [1,2], \\ \vdots \\ A^{(m)}(k-(m-1)), & k \in [m-1,m]. \end{cases}$$

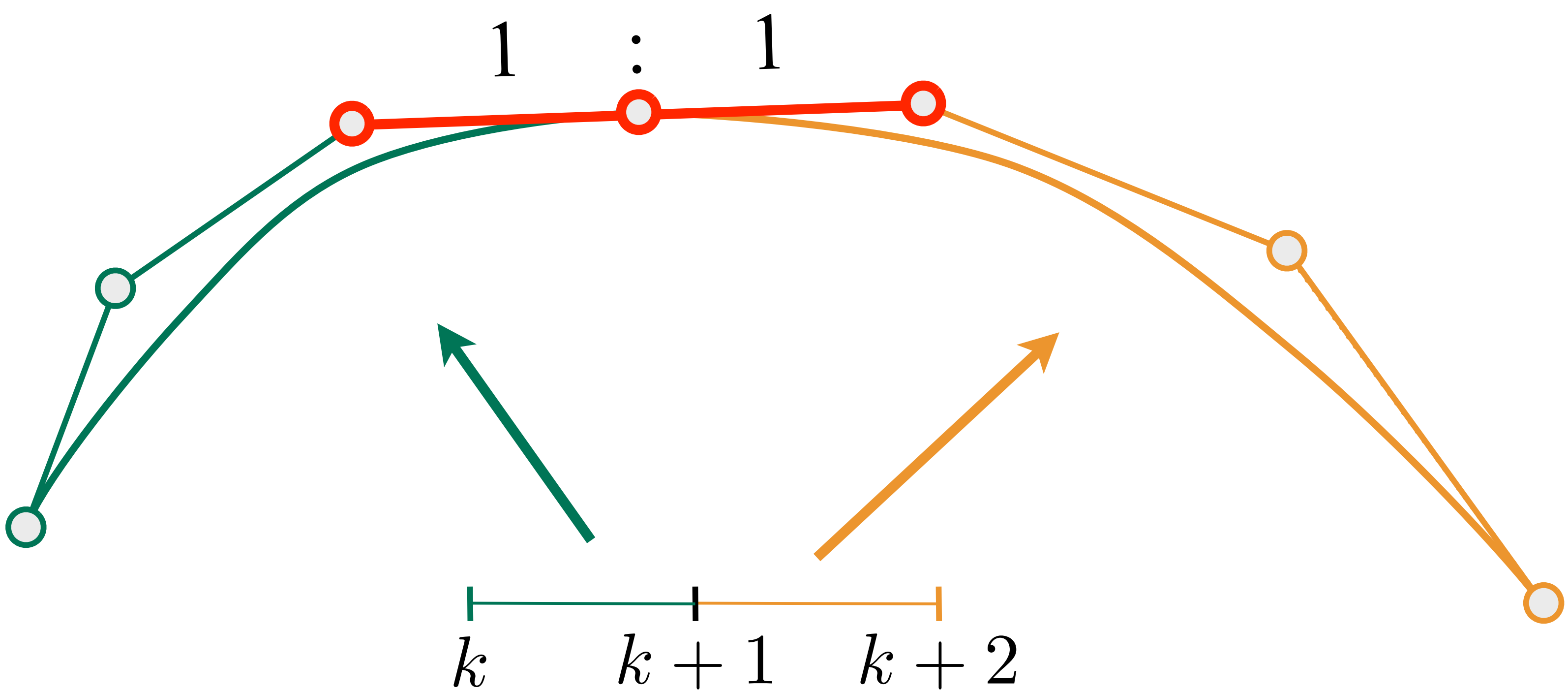
Piecewise Bézier Curve – Continuity

C^0 continuity: $\mathbf{a}_n = \mathbf{b}_0$



Piecewise Bézier Curve – Continuity

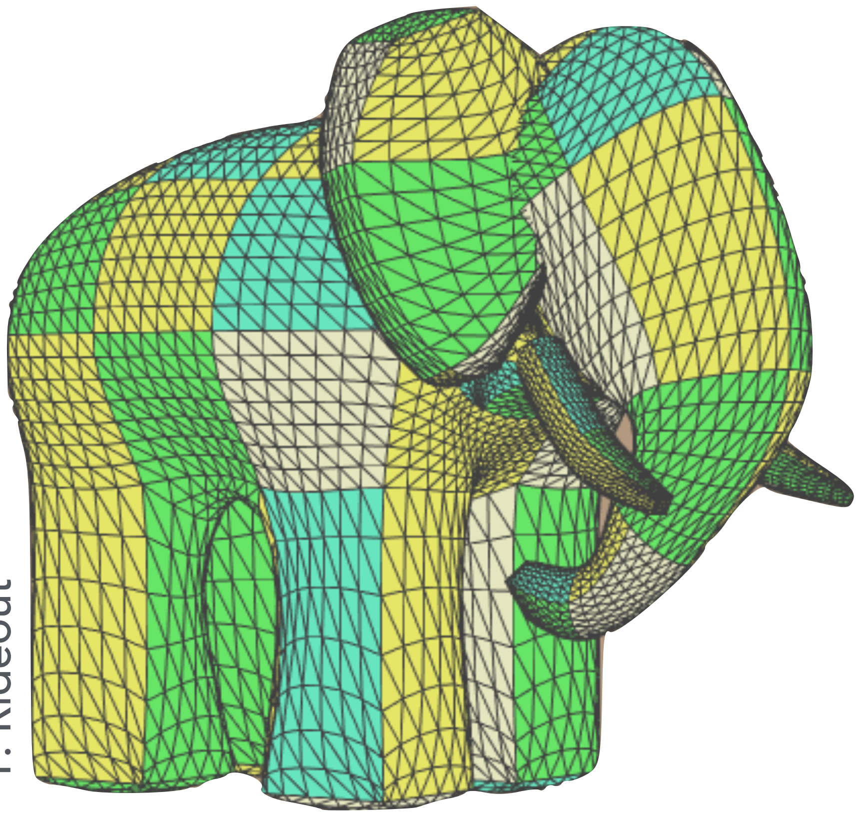
C¹ continuity: $$\mathbf{a}_n = \mathbf{b}_0 = \frac{1}{2}\left(\mathbf{a}_{n-1} + \mathbf{b}_1\right)$$



Bézier Surfaces

Bézier Surfaces

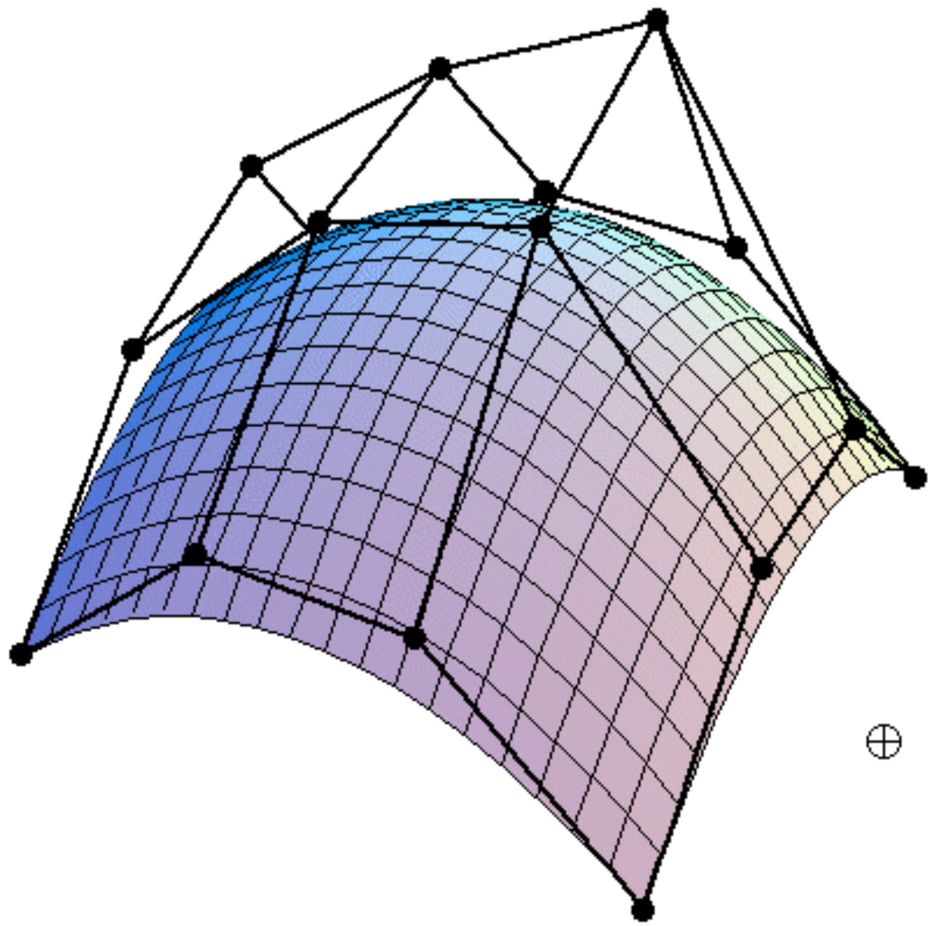
Extend Bézier curves to surfaces



Ed Catmull's "Gumbo" model

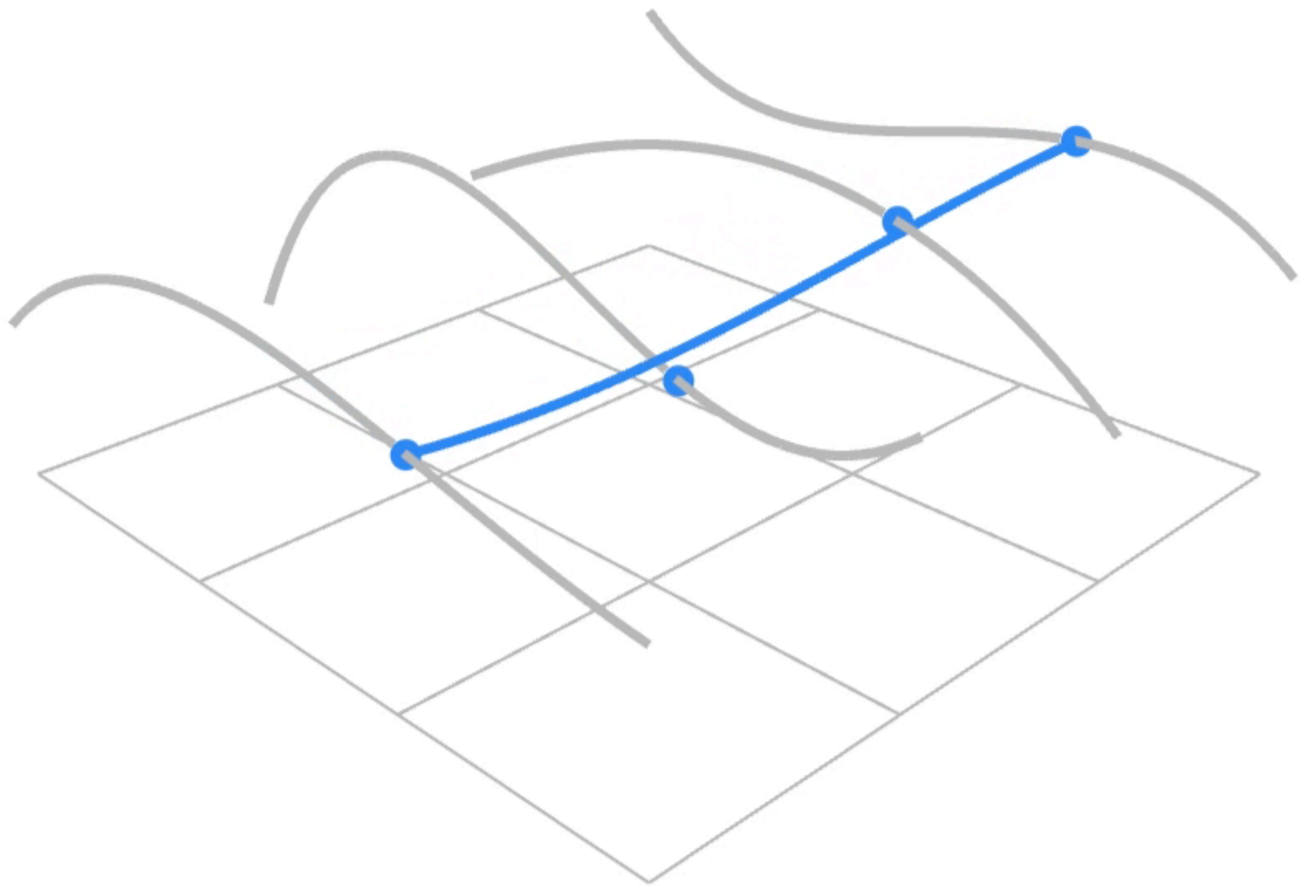
Utah Teapot

Bicubic Bézier Surface Patch



Bezier surface and 4 x 4 array of control points

Visualizing Bicubic Bézier Surface Patch



Animation: Steven Wittens, Making Things with Maths, http://acko.net

Visualizing Bicubic Bézier Surface Patch

4x4 control points

- Each 4x1 control points in u define a Bezier curve
  - (4 Bezier curves in u)
- Corresponding points on these 4 Bezier curves define 4 control points for a “moving curve” in v
  - This “moving” curve sweeps out the 2D surface

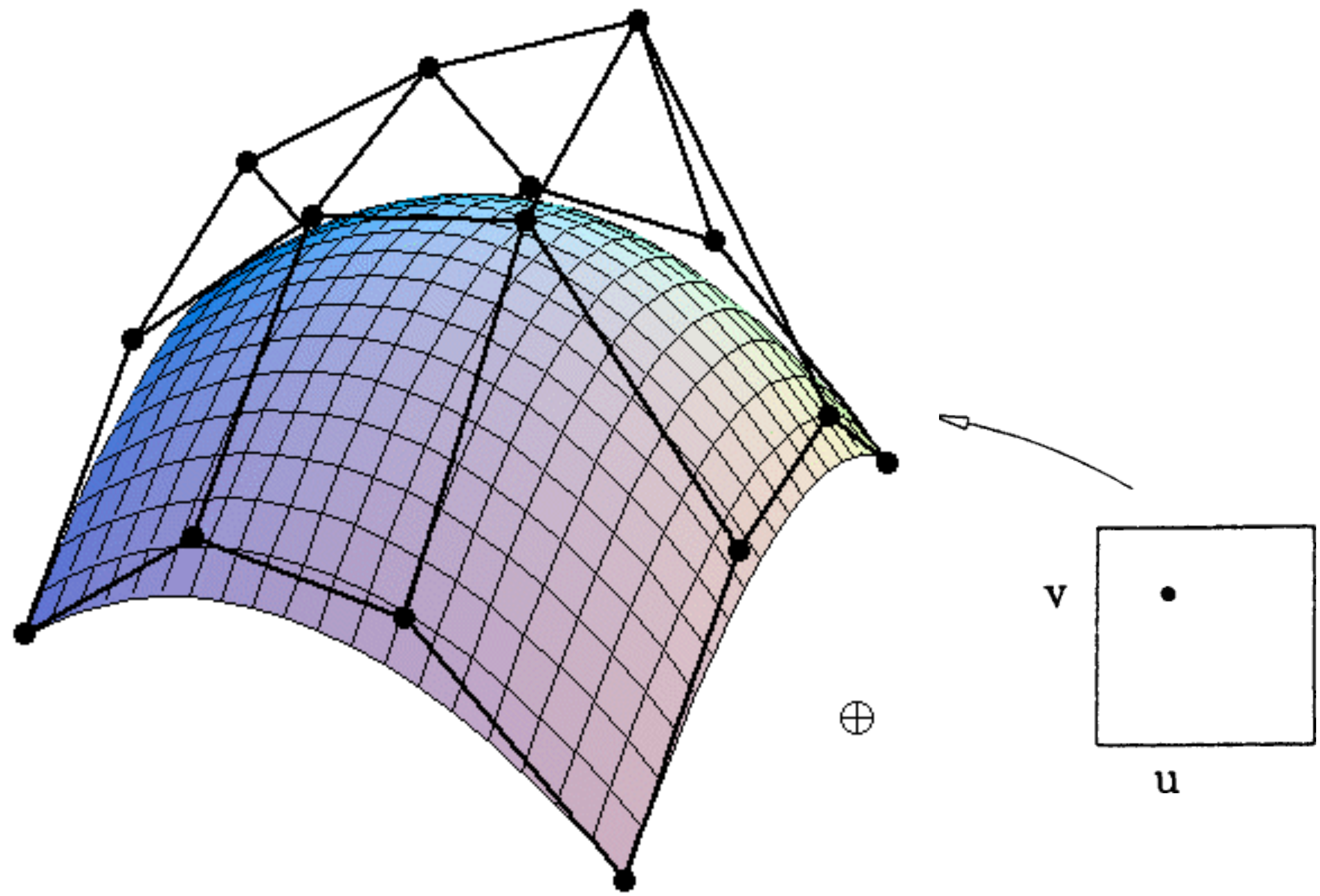
Evaluating Bézier Surfaces

Evaluating Surface Position For Parameters (u,v)

For bi-cubic Bezier surface patch,

Input: 4x4 control points

Output is 2D surface parameterized by (u,v) in $[0,1]^2$

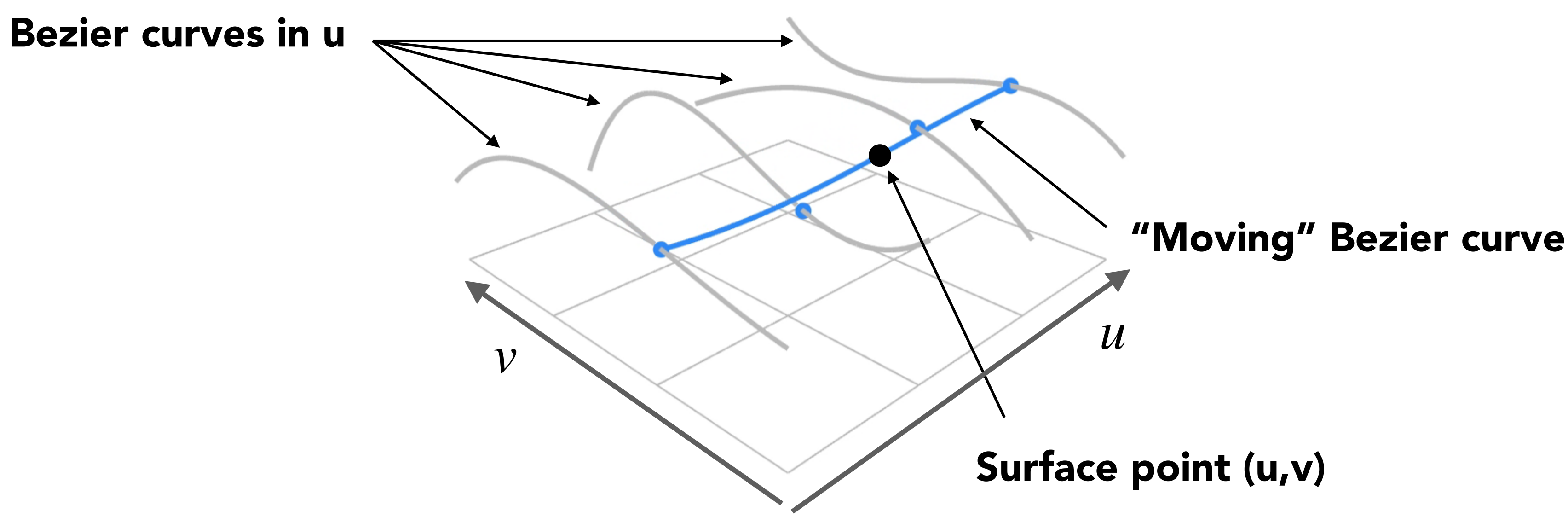


Method 1: Separable 1D de Casteljau Algorithm

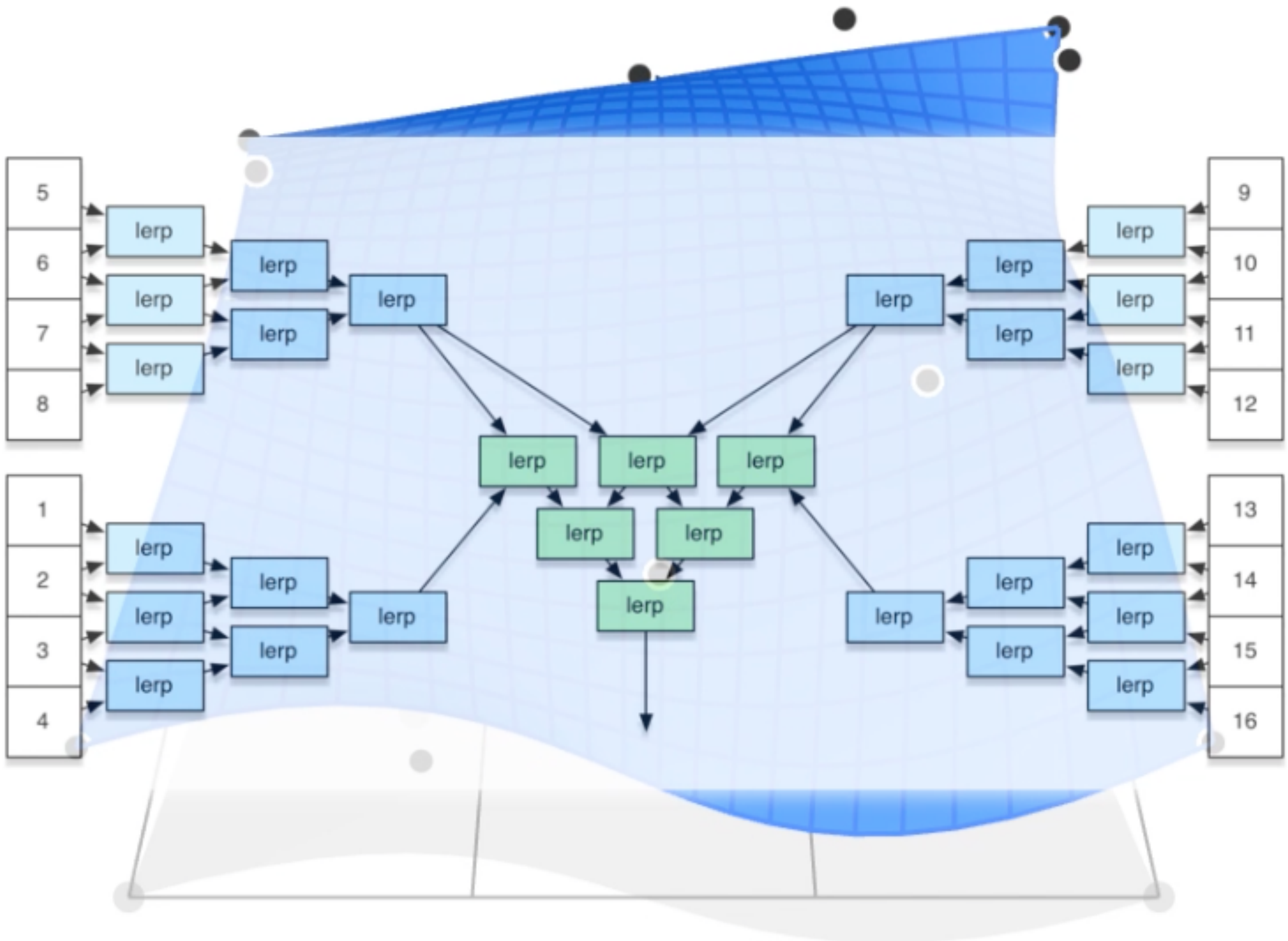
Goal: Evaluate surface position corresponding to (u,v)

(u,v)-separable application of de Casteljau algorithm

- Use de Casteljau to evaluate point u on each of the 4 Bezier curves in u. This gives 4 control points for the “moving” Bezier curve
- Use 1D de Casteljau to evaluate point v on the “moving” curve



Method 1: Separable 1D de Casteljau Algorithm



Method 2: Algebraic Evaluation

Let the moving curve be a degree m Bézier curve

$$\mathbf{b}^m(u) = \sum_{i=0}^{m} \mathbf{b}_i B_i^m(u)$$

(remember, Bernstein polynomials)

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

Let each control point $\mathbf{b}_i$ be moving along a Bézier curve of degree n

$$\mathbf{b}_i = \mathbf{b}_i(v) = \sum_{j=0}^{n} \mathbf{b}_{i,j} B_j^n(v)$$

Tensor product Bézier patch

$$\mathbf{b}^{m,n}(u,v) = \sum_{i=0}^{m} \sum_{j=0}^{n} \mathbf{b}_{i,j} B_i^m(u) B_j^n(v)$$

Method 3: Linear Algebra

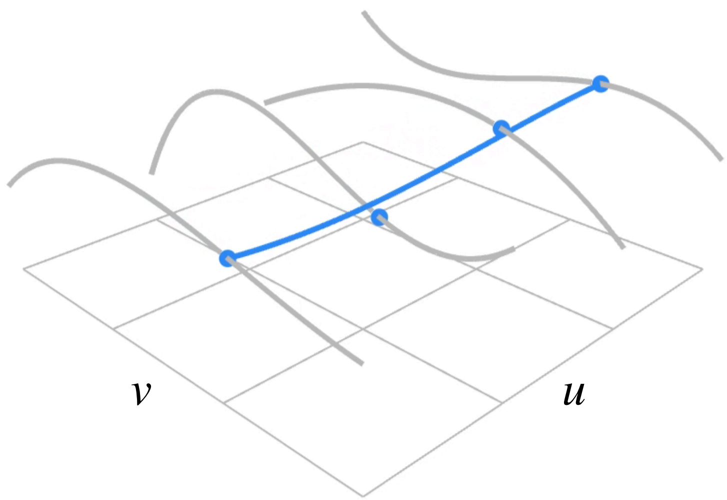
$$\mathbf{x}(u) = \mathscr{P}^3(u) \cdot \boldsymbol{\beta}_Z \cdot \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{bmatrix}$$

$$\mathbf{p}_0(v) = \mathscr{P}^3(v) \cdot \boldsymbol{\beta}_Z \cdot \begin{bmatrix} \mathbf{p}_{0,0} \\ \mathbf{p}_{0,1} \\ \mathbf{p}_{0,2} \\ \mathbf{p}_{0,3} \end{bmatrix}$$

$$\mathbf{p}_1(v) = \mathscr{P}^3(v) \cdot \boldsymbol{\beta}_Z \cdot \begin{bmatrix} \mathbf{p}_{1,0} \\ \mathbf{p}_{1,1} \\ \mathbf{p}_{1,2} \\ \mathbf{p}_{1,3} \end{bmatrix}$$

$$\mathbf{p}_2(v) = \mathscr{P}^3(v) \cdot \boldsymbol{\beta}_Z \cdot \begin{bmatrix} \mathbf{p}_{2,0} \\ \mathbf{p}_{2,1} \\ \mathbf{p}_{2,2} \\ \mathbf{p}_{2,3} \end{bmatrix}$$

$$\mathbf{p}_3(v) = \mathscr{P}^3(v) \cdot \boldsymbol{\beta}_Z \cdot \begin{bmatrix} \mathbf{p}_{3,0} \\ \mathbf{p}_{3,1} \\ \mathbf{p}_{3,2} \\ \mathbf{p}_{3,3} \end{bmatrix}$$



Method 3: Linear Algebra

$$\mathbf{x}(u) = \mathscr{P}^3(u) \cdot \boldsymbol{\beta}_Z \cdot \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{bmatrix}$$

$$\mathbf{p}_0(v) = \mathscr{P}^3(v) \cdot \boldsymbol{\beta}_Z \cdot \begin{bmatrix} \mathbf{p}_{0,0} \\ \mathbf{p}_{0,1} \\ \mathbf{p}_{0,2} \\ \mathbf{p}_{0,3} \end{bmatrix}$$

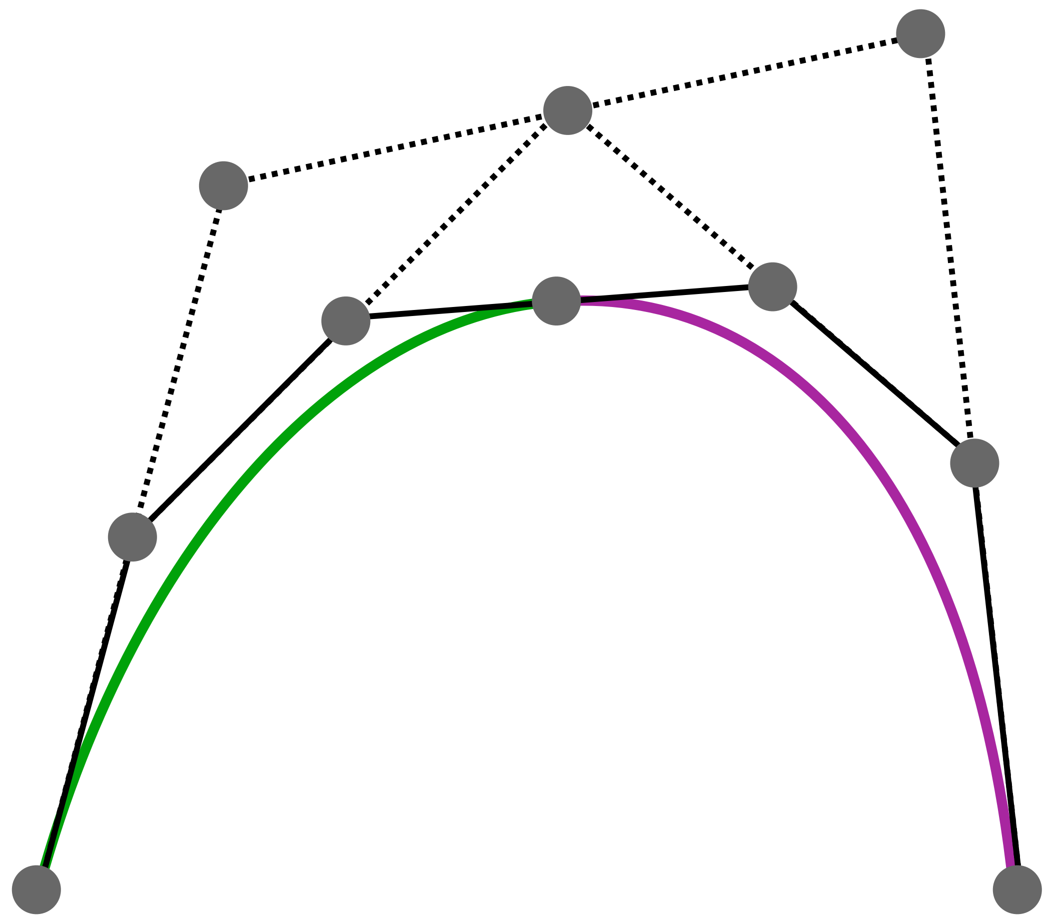
$$\mathbf{p}_1(v) = \mathscr{P}^3(v) \cdot \boldsymbol{\beta}_Z \cdot \begin{bmatrix} \mathbf{p}_{1,0} \\ \mathbf{p}_{1,1} \\ \mathbf{p}_{1,2} \\ \mathbf{p}_{1,3} \end{bmatrix}$$

$$\mathbf{p}_2(v) = \mathscr{P}^3(v) \cdot \boldsymbol{\beta}_Z \cdot \begin{bmatrix} \mathbf{p}_{2,0} \\ \mathbf{p}_{2,1} \\ \mathbf{p}_{2,2} \\ \mathbf{p}_{2,3} \end{bmatrix}$$

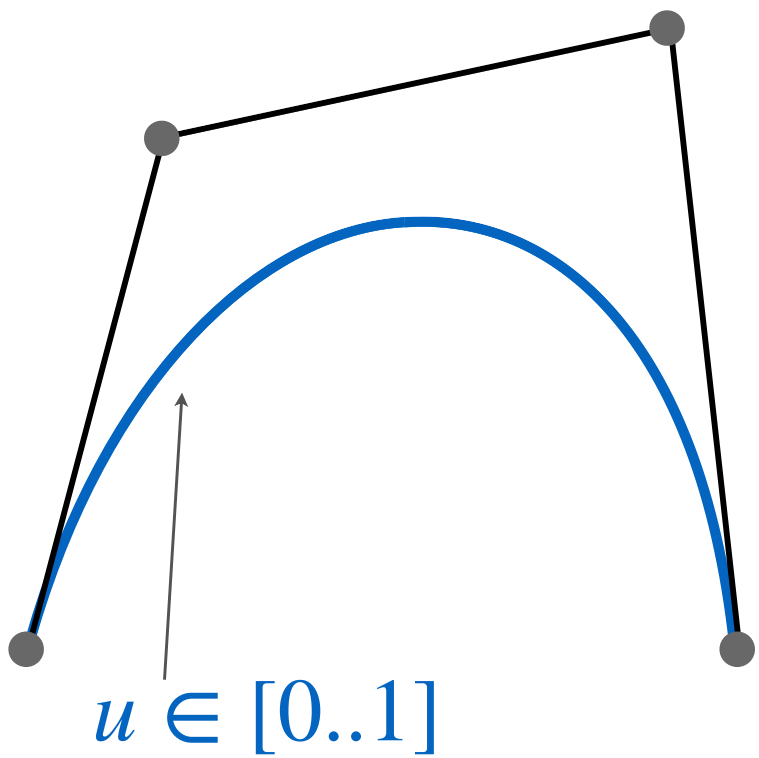
$$\mathbf{p}_3(v) = \mathscr{P}^3(v) \cdot \boldsymbol{\beta}_Z \cdot \begin{bmatrix} \mathbf{p}_{3,0} \\ \mathbf{p}_{3,1} \\ \mathbf{p}_{3,2} \\ \mathbf{p}_{3,3} \end{bmatrix}$$

$$\mathbf{x}(u, v) = \mathscr{P}^3(u) \cdot \boldsymbol{\beta}_Z \cdot \begin{bmatrix} \mathbf{p}_{0,0} & \mathbf{p}_{0,1} & \mathbf{p}_{0,2} & \mathbf{p}_{0,3} \\ \mathbf{p}_{1,0} & \mathbf{p}_{1,1} & \mathbf{p}_{1,2} & \mathbf{p}_{1,3} \\ \mathbf{p}_{2,0} & \mathbf{p}_{2,1} & \mathbf{p}_{2,2} & \mathbf{p}_{2,3} \\ \mathbf{p}_{3,0} & \mathbf{p}_{3,1} & \mathbf{p}_{3,2} & \mathbf{p}_{3,3} \end{bmatrix} \cdot \boldsymbol{\beta}_Z^\mathsf{T} \cdot \mathscr{P}^3(v)^\mathsf{T}$$

Method 4: Bézier Subdivision



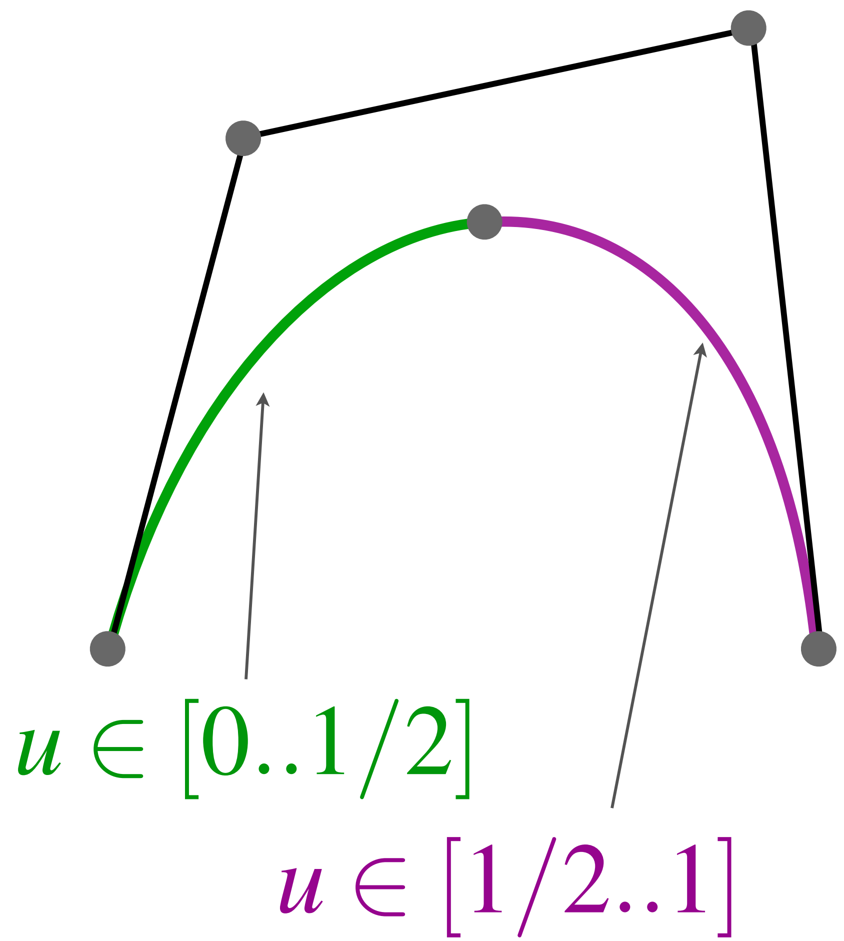
Bézier Subdivision



$$\mathbf{x}(u) = [1, u, u^2, u^3]\, \beta_Z \, \mathbf{P}$$

$$\beta_Z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}$$

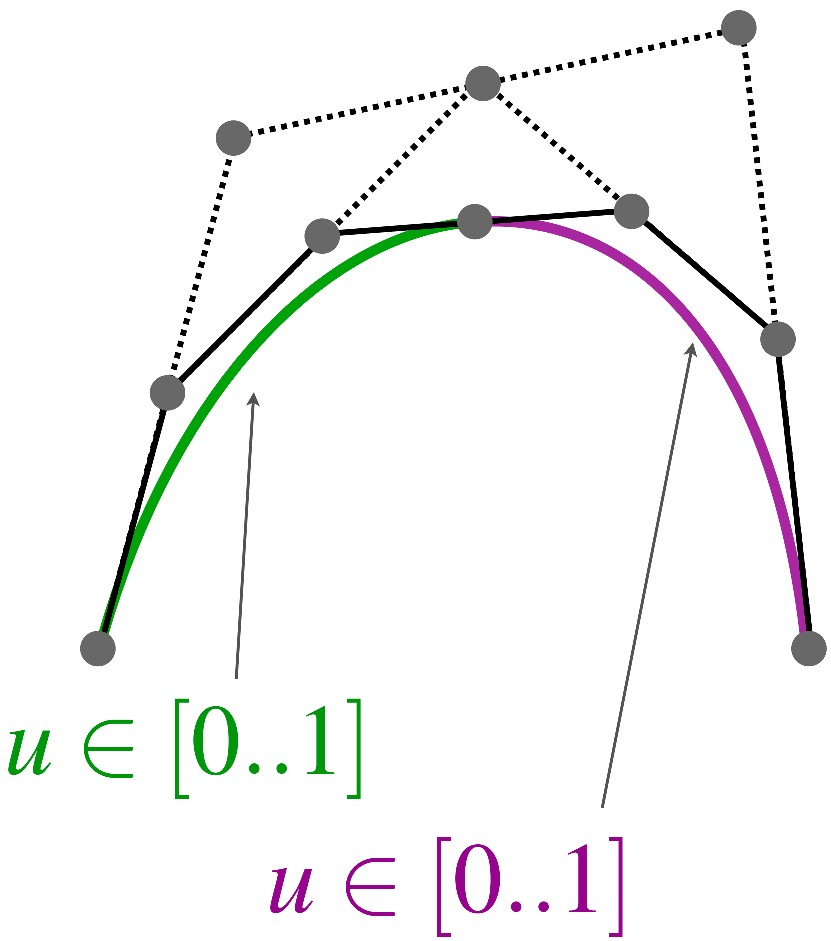
Bézier Subdivision



$$\mathbf{x}(u) = [1, u, u^2, u^3]\, \beta_Z \mathbf{P}$$

$$\beta_Z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}$$

Bézier Subdivision



$$\mathbf{x}(u) = [1, u, u^2, u^3]\, \beta_Z \mathbf{P}$$

Can't change these...

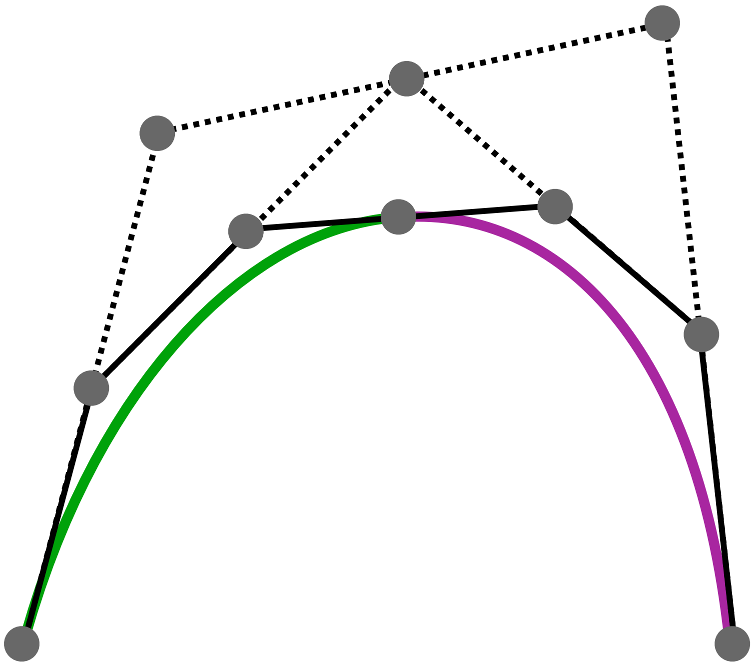
$$\beta_Z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}$$

Bézier Subdivision

$$\mathbf{x}(u) = [1, u, u^2, u^3]\,\beta_Z\,\mathbf{P} \qquad u \in [0..\tfrac{1}{2}]$$

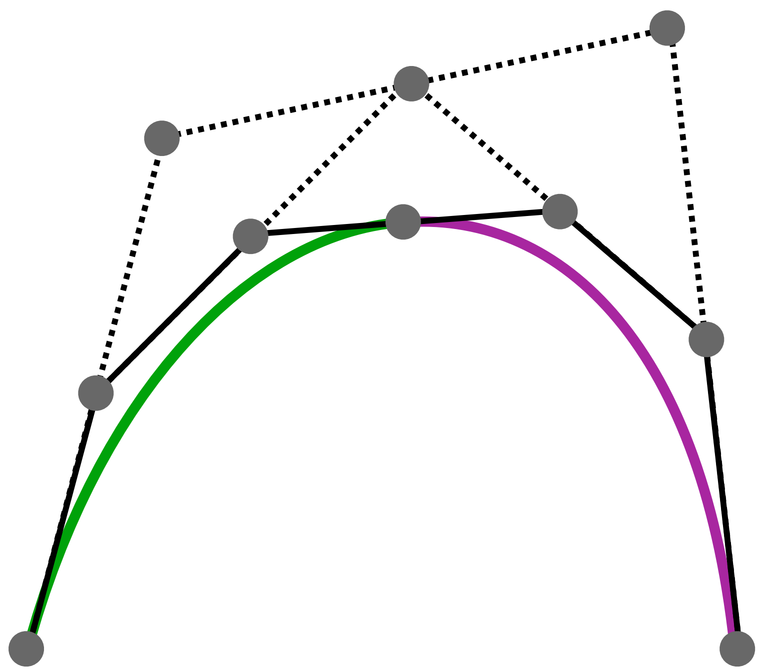
$$\mathbf{x}(u) = [1, \frac{u}{2}, \frac{u^2}{4}, \frac{u^3}{8}]\beta_Z\mathbf{P} \qquad u \in [0..1]$$

$$\mathbf{x}(u) = [1, u, u^2, u^3]\mathbf{S}_1\beta_Z\mathbf{P}$$



$$\mathbf{S}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 \\ 0 & 0 & 1/4 & 0 \\ 0 & 0 & 0 & 1/8 \end{bmatrix}$$

Bézier Subdivision



$$\mathbf{x}(u) = [1, u, u^2, u^3]\, \beta_Z\, \mathbf{P}$$ $u \in [0..\frac{1}{2}]$

$$\mathbf{x}(u) = [1, \frac{u}{2}, \frac{u^2}{4}, \frac{u^3}{8}] \beta_Z \mathbf{P}$$ $u \in [0..1]$

$$\mathbf{x}(u) = [1, u, u^2, u^3] \mathbf{S}_1 \beta_Z \mathbf{P}$$

$$\mathbf{x}(u) = [1, u, u^2, u^3] \beta_Z \beta_Z^{-1} \mathbf{S}_1 \beta_Z \mathbf{P}$$

$$\mathbf{x}(u) = [1, u, u^2, u^3] \beta_Z \mathbf{H}_{Z1} \mathbf{P}$$

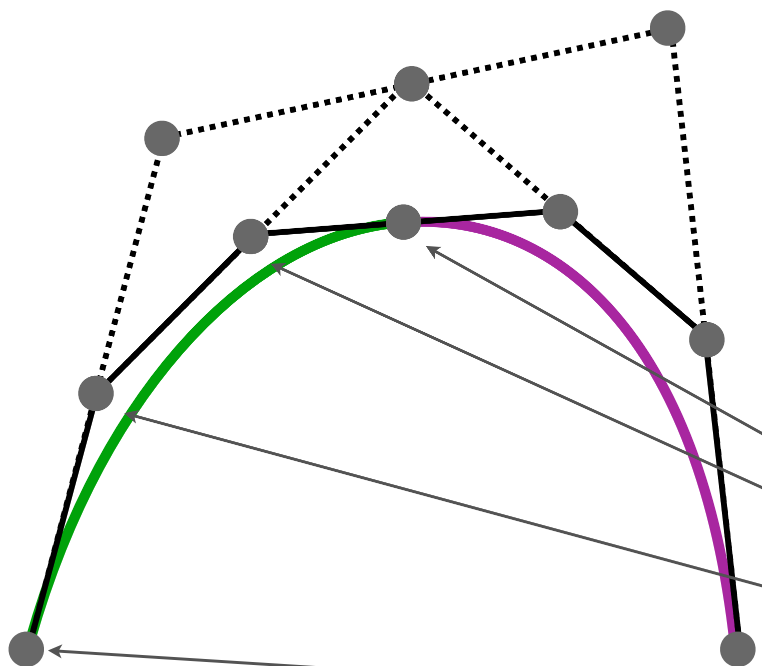
$$\mathbf{S}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 \\ 0 & 0 & 1/4 & 0 \\ 0 & 0 & 0 & 1/8 \end{bmatrix}$$

Bézier Subdivision

$$\mathbf{x}(u) = [1, u, u^2, u^3]\beta_Z \beta_Z^{-1}\mathbf{S}_1\beta_Z\mathbf{P}$$

$$\mathbf{x}(u) = [1, u, u^2, u^3]\beta_Z \mathbf{H}_{Z1}\mathbf{P}$$

$$\mathbf{x}(u) = [1, u, u^2, u^3]\beta_Z \mathbf{P}_1$$

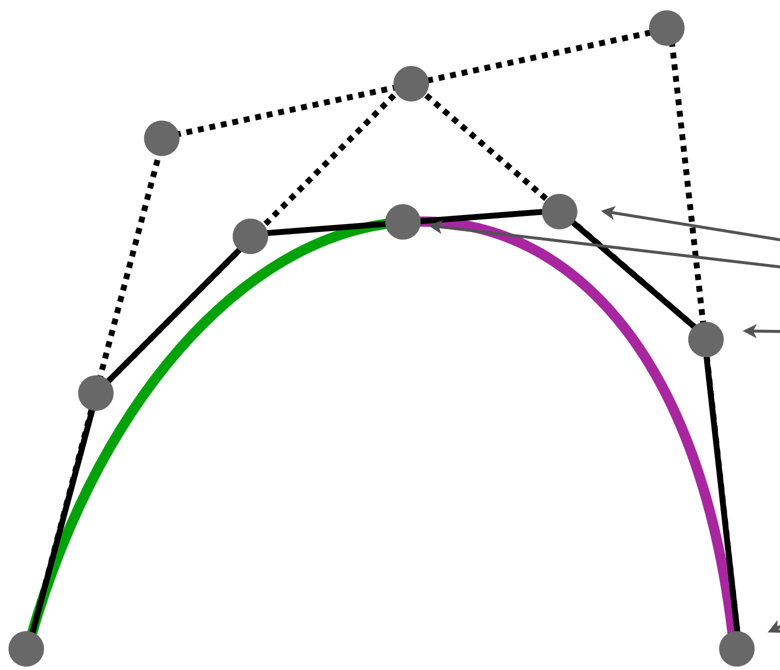


$$\mathbf{H}_{Z1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} & 0 \\ \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{bmatrix}$$

$$\mathbf{P}_1 = \mathbf{H}_{Z1}\mathbf{P}$$

$$\mathbf{S}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 \\ 0 & 0 & 1/4 & 0 \\ 0 & 0 & 0 & 1/8 \end{bmatrix}$$

Bézier Subdivision



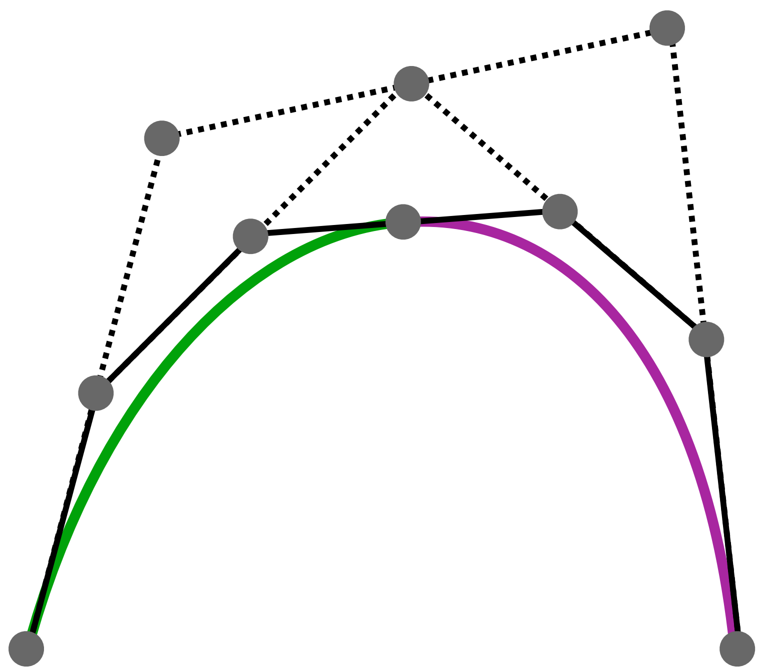
$$\mathbf{x}(u) = [1, u, u^2, u^3]\beta_Z \mathbf{P}_2$$

$$\mathbf{P}_2 = \mathbf{H}_{Z2}\mathbf{P}$$

$$\mathbf{H}_{Z2} = \begin{bmatrix} \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \\ 0 & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{S}_2 = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{4} & \frac{1}{8} \\ 0 & \frac{1}{2} & \frac{1}{2} & \frac{3}{8} \\ 0 & 0 & \frac{1}{4} & \frac{3}{8} \\ 0 & 0 & 0 & \frac{1}{8} \end{bmatrix}$$

Bézier Subdivision



$$\mathbf{P}_1 = \mathbf{H}_{Z1}\mathbf{P}$$

$$\mathbf{P}_2 = \mathbf{H}_{Z2}\mathbf{P}$$

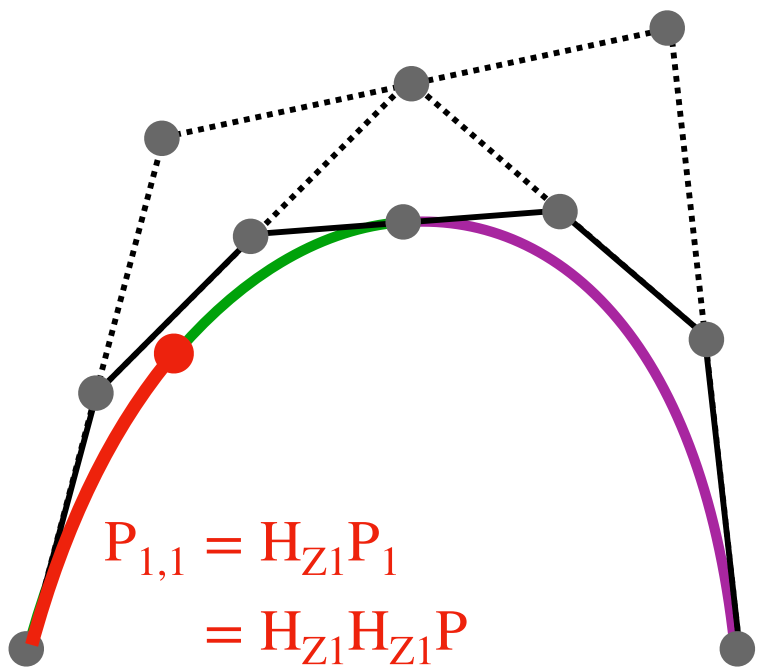
$$\mathbf{H}_{Z1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} & 0 \\ \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{bmatrix}$$

$$\mathbf{H}_{Z2} = \begin{bmatrix} \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \\ 0 & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rows of $\mathrm{H}_{Z.}$ sum to 1 and ≥ 0 $\Rightarrow$ Convex Hull $\Rightarrow$ $\mathbf{P}_1$ and $\mathbf{P}_2 \in$ ConvexHull$(\mathbf{P})$

Curve remains constant $\Rightarrow$ Control Mesh converges to surface!!

Bézier Subdivision



$$\mathbf{P}_1 = \mathbf{H}_{Z1}\mathbf{P}$$

$$\mathbf{P}_2 = \mathbf{H}_{Z2}\mathbf{P}$$

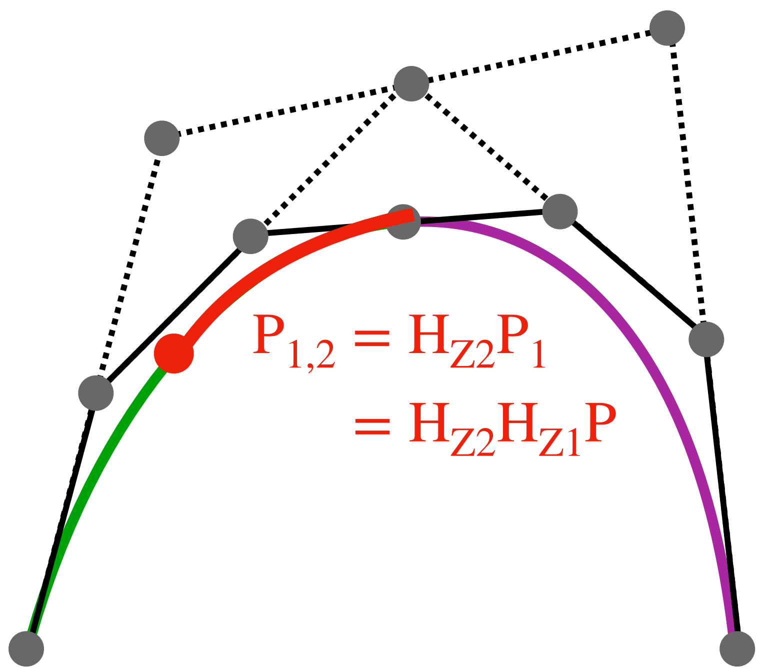
$$\mathbf{H}_{Z1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} & 0 \\ \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{bmatrix}$$

$$\mathbf{H}_{Z2} = \begin{bmatrix} \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \\ 0 & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rows of $\mathbf{H}_{Z.}$ sum to 1 and ≥ 0 $\Rightarrow$ Convex Hull $\Rightarrow$ $\mathbf{P}_1$ and $\mathbf{P}_2 \in$ ConvexHull$(\mathbf{P})$

Curve remains constant $\Rightarrow$ Control Mesh converges to surface!!

Bézier Subdivision



$$\mathbf{P}_1 = \mathbf{H}_{Z1}\mathbf{P}$$

$$\mathbf{P}_2 = \mathbf{H}_{Z2}\mathbf{P}$$

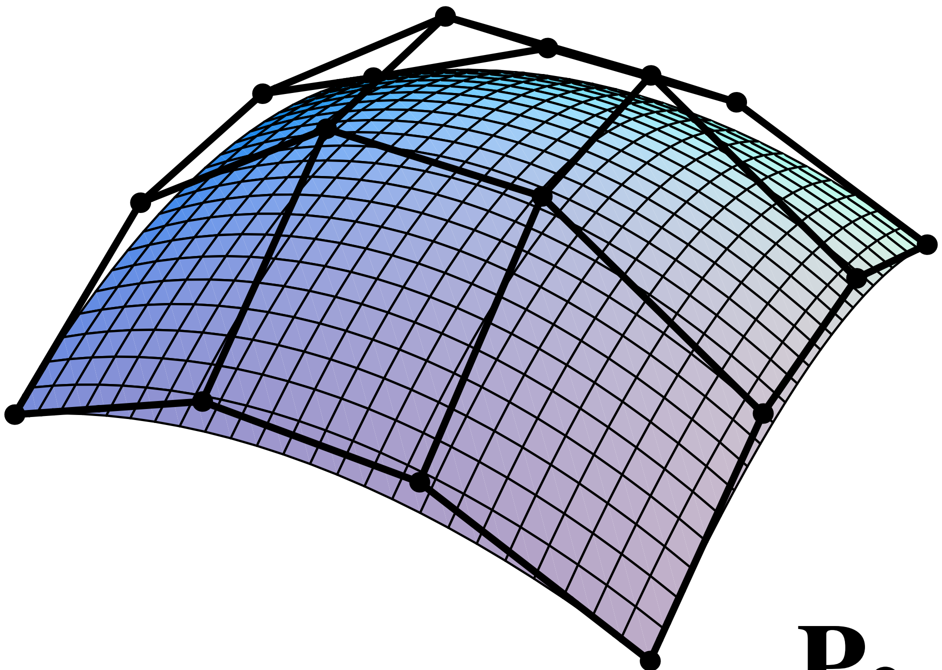
$$\mathbf{H}_{Z1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} & 0 \\ \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{bmatrix}$$

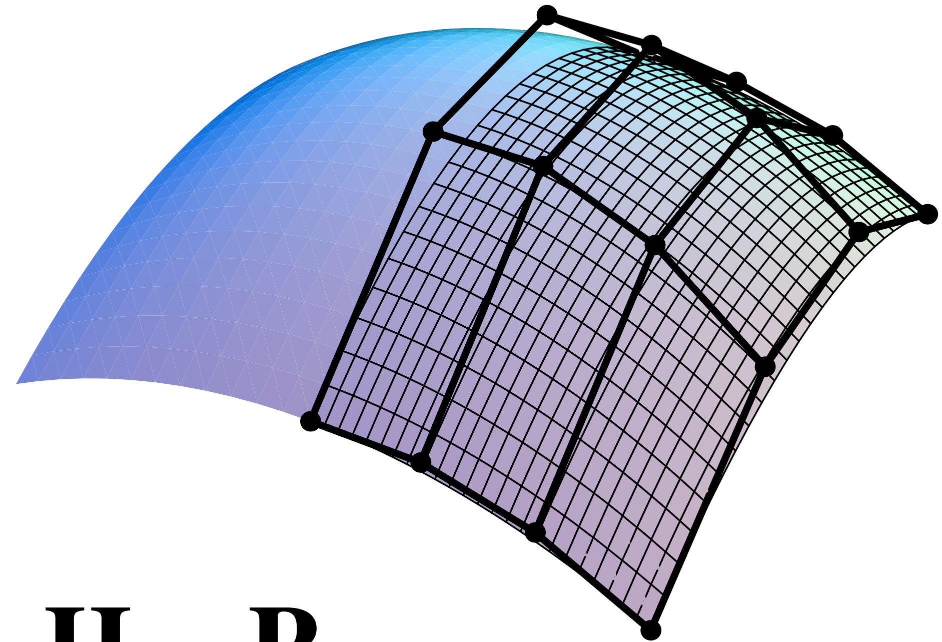
$$\mathbf{H}_{Z2} = \begin{bmatrix} \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \\ 0 & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rows of $\mathrm{H}_{Z.}$ sum to 1 and ≥ 0 $\Rightarrow$ Convex Hull $\Rightarrow$ P_1 and $\mathrm{P}_2 \in$ ConvexHull$(\mathbf{P})$

Curve remains constant $\Rightarrow$ Control Mesh converges to surface!!

Bézier Subdivision





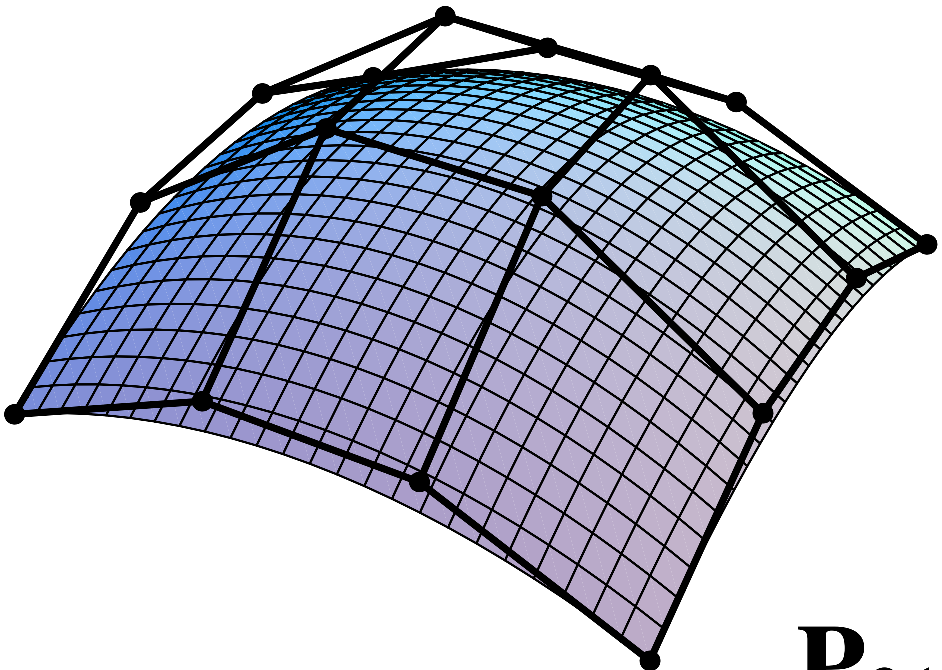
$$\mathbf{P}_{2\cdot} = \mathbf{H}_{Z2}\,\mathbf{P}$$

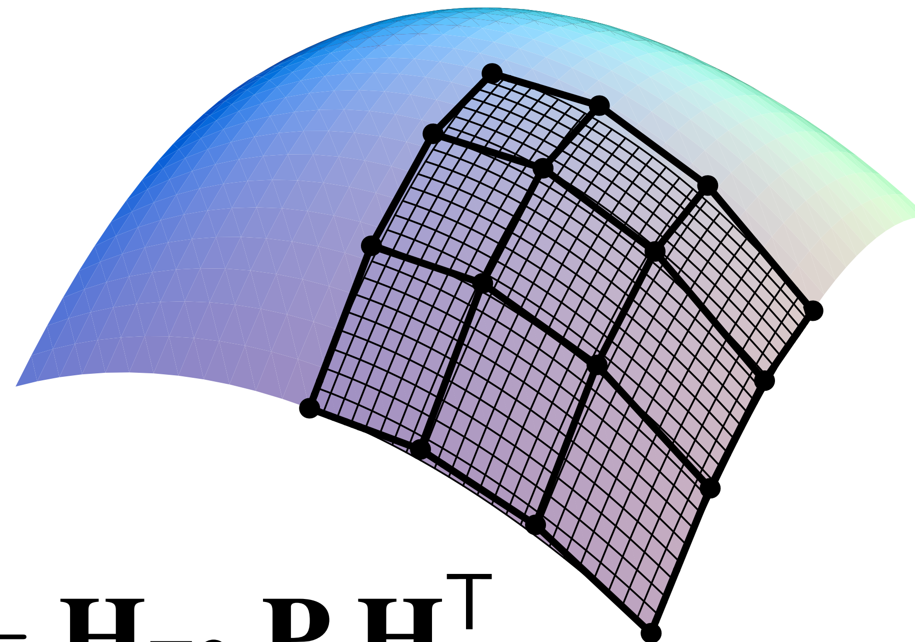
$$\mathbf{x}(u,v) = [1, u, u^2, u^3]\,\beta_Z \mathbf{P} \beta_Z^\top [1, v, v^2, v^3]^\top$$

4 x 4 matrix of control points

$$\mathbf{x}(u,v) = \mathscr{P}^3(u) \cdot \boldsymbol{\beta}_Z \cdot \begin{bmatrix} \mathbf{p}_{0,0} & \mathbf{p}_{0,1} & \mathbf{p}_{0,2} & \mathbf{p}_{0,3} \\ \mathbf{p}_{1,0} & \mathbf{p}_{1,1} & \mathbf{p}_{1,2} & \mathbf{p}_{1,3} \\ \mathbf{p}_{2,0} & \mathbf{p}_{2,1} & \mathbf{p}_{2,2} & \mathbf{p}_{2,3} \\ \mathbf{p}_{3,0} & \mathbf{p}_{3,1} & \mathbf{p}_{3,2} & \mathbf{p}_{3,3} \end{bmatrix} \cdot \boldsymbol{\beta}_Z^\top \cdot \mathscr{P}^3(v)^\top$$

Bézier Subdivision





$$\mathbf{P}_{21} = \mathbf{H}_{Z2}\,\mathbf{P}\,\mathbf{H}_{Z1}^{\mathsf{T}}$$

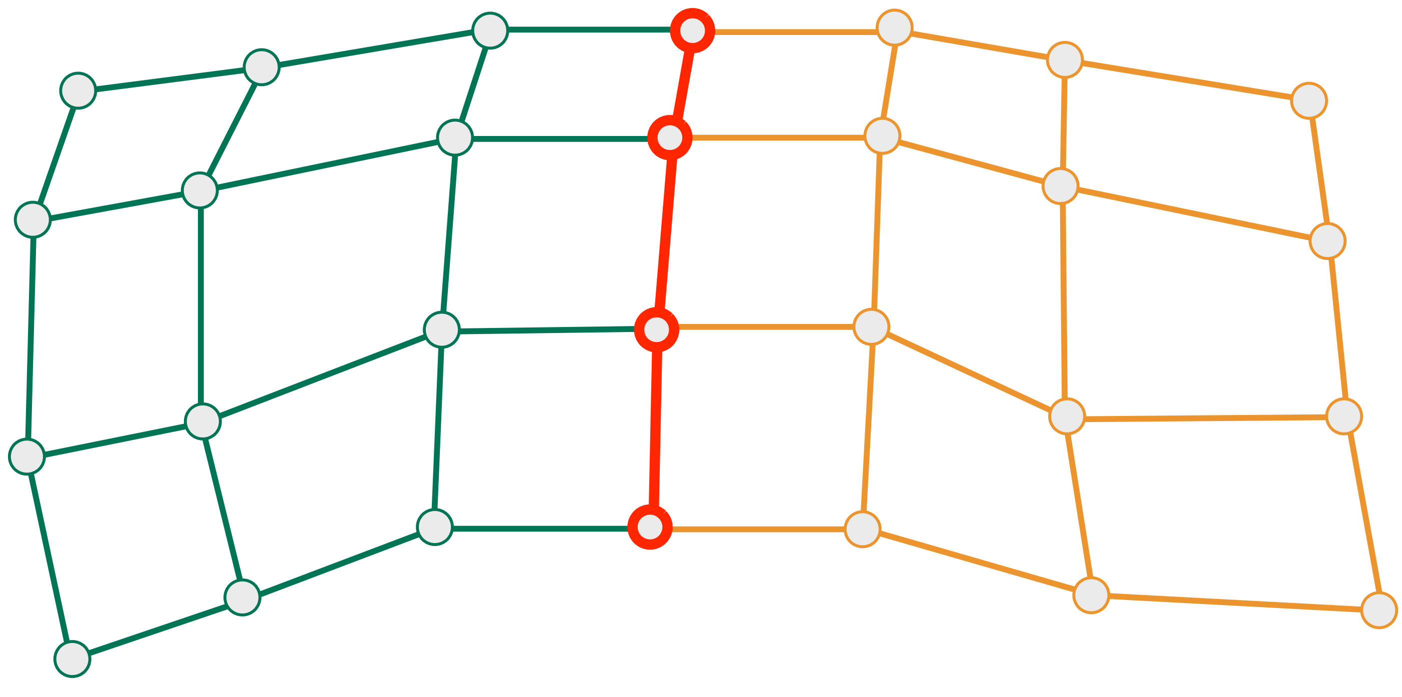
$$\mathbf{x}(u,v) = [1,u,u^2,u^3]\,\beta_Z \mathbf{P} \beta_Z^{\mathsf{T}}\,[1,v,v^2,v^3]^{\mathsf{T}}$$

4 x 4 matrix of control points

Bézier Surface Continuity

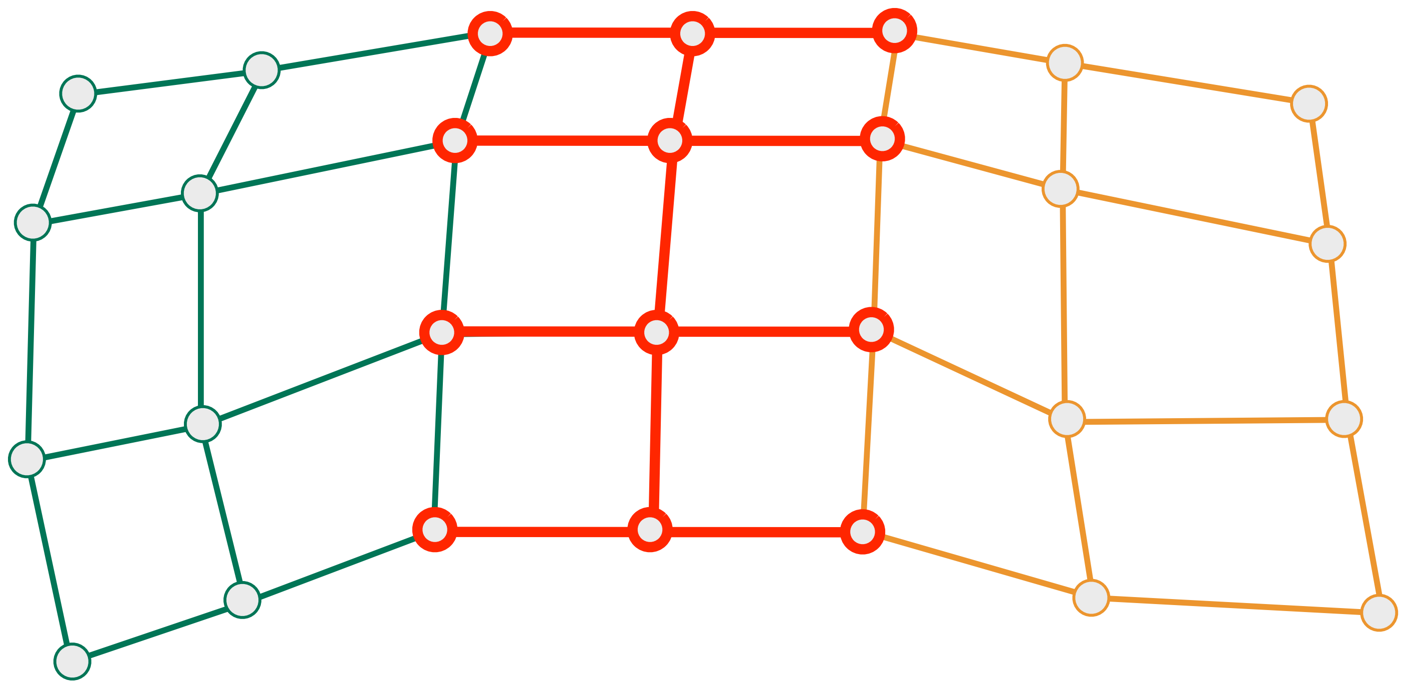
Piecewise Bézier Surfaces

C^0 continuity: Boundary curves



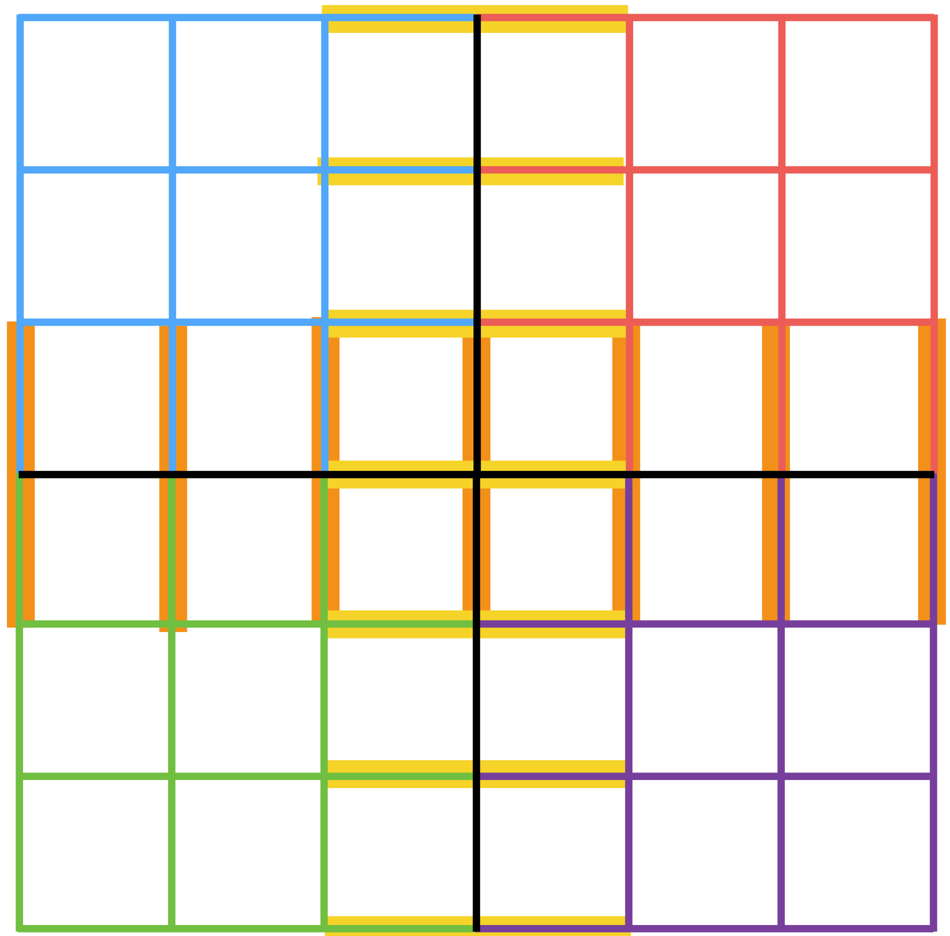
Piecewise Bézier Surfaces

C^1 continuity: Collinearity



Piecewise Bézier Surfaces

C^1 continuity: Collinearity



Things to Remember

Splines

- Cubic Hermite and Catmull-Rom interpolation
- Matrix representation of cubic polynomials

Bézier curves

- Easy-to-control spline
- Recursive linear interpolation – de Casteljau algorithm
- Properties of Bézier curves
- Piecewise Bézier curve – continuity types and how to achieve

Bézier surfaces

- Bicubic Bézier patches – tensor product surface
- 2D de Casteljau algorithm