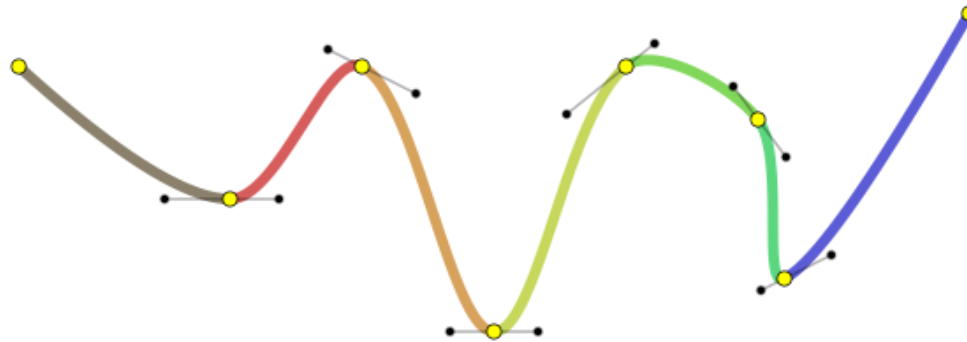


# Spline

---

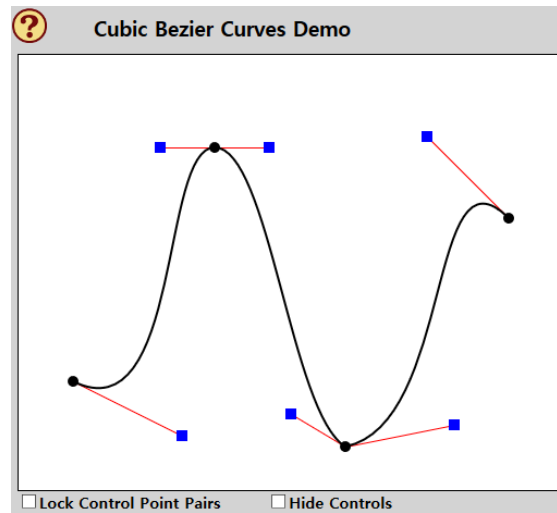
- Spline: *piecewise* polynomial



- Three issues:
  - How to connect these pieces *continuously*?
  - How easy is it to "*control*" the shape of a spline?
  - Does a spline have to *pass through* specific points?

# Continuity

- Let's try another Bezier demo: Bezier spline



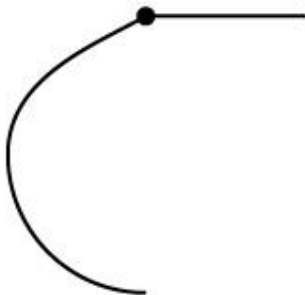
<http://math.hws.edu/graphicsbook/demos/c2/cubic-bezier.html>

- How to “smooth” the spline?

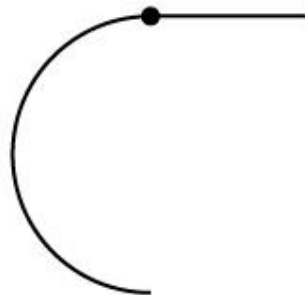
# Continuity

- Smoothness can be described by degree of continuity
  - zero-order ( $C^0$ ): position matches from both sides
  - first-order ( $C^1$ ): position and 1<sup>st</sup> derivative (velocity) match from both sides
  - second-order ( $C^2$ ): position and 1<sup>st</sup> & 2<sup>nd</sup> derivatives (velocity & acceleration) match from both sides

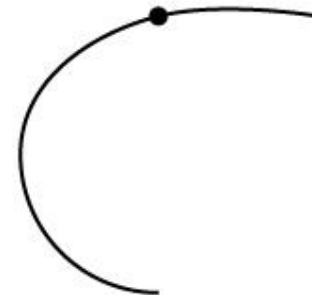
zero order



first order

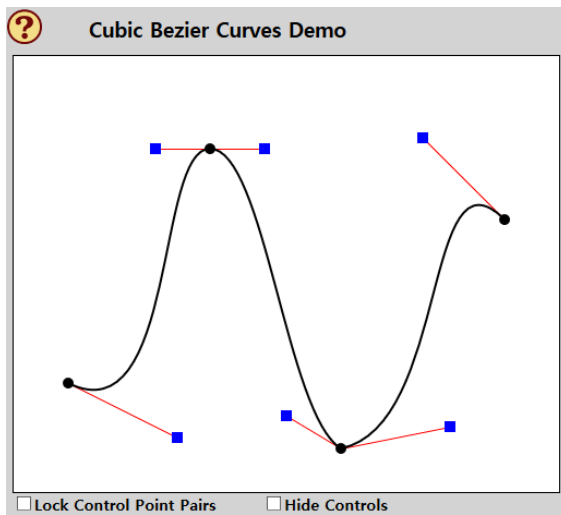


second order

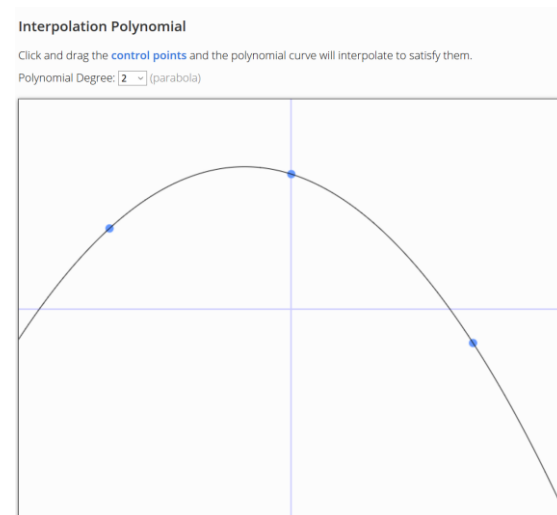


# Control

- Let's say you want to make a specific shape using these two curves. Which one is more controllable?



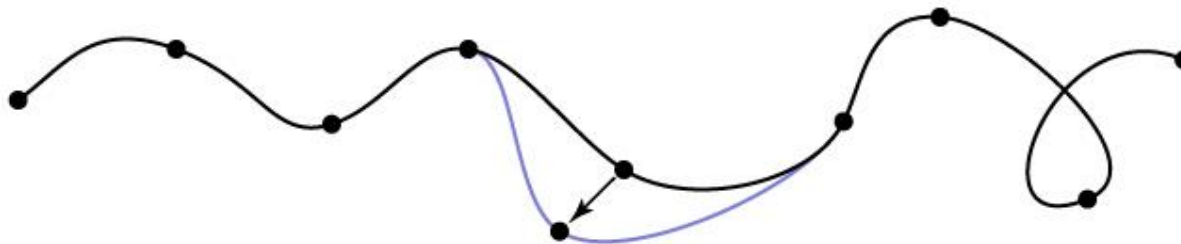
<http://math.hws.edu/graphicsbook/demos/c2/cubic-bezier.html>



<https://www.benjoffe.com/code/demos/interpolate>

# Control

- Local control
  - changing control point only affects a **limited part** of spline
  - without this, splines are very difficult to use
  - many likely formulations lack this
    - natural spline
    - polynomial fits



# Interpolation / Approximation

---

- Interpolation: passes through points



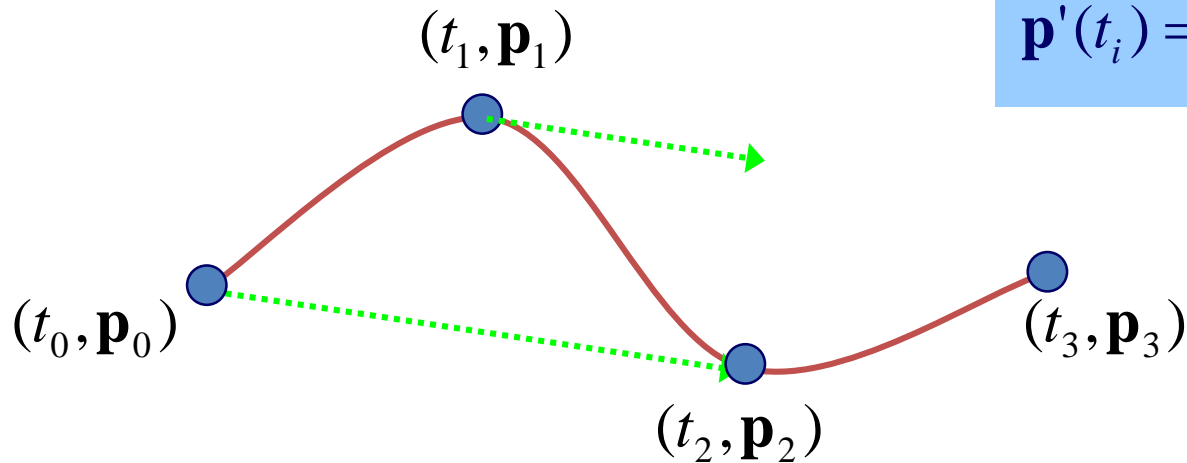
- Approximation: merely guided by points



- Interpolation properties are preferable, but not mandatory

# Catmull-Rom Spline

- A Bezier or Hermite curve interpolates two end points only
- What if we want a cubic spline interpolating all control points?
- Catmull-Rom Splines
  - One Hermite curve between two consecutive control points
  - Define end point derivatives using adjacent control points

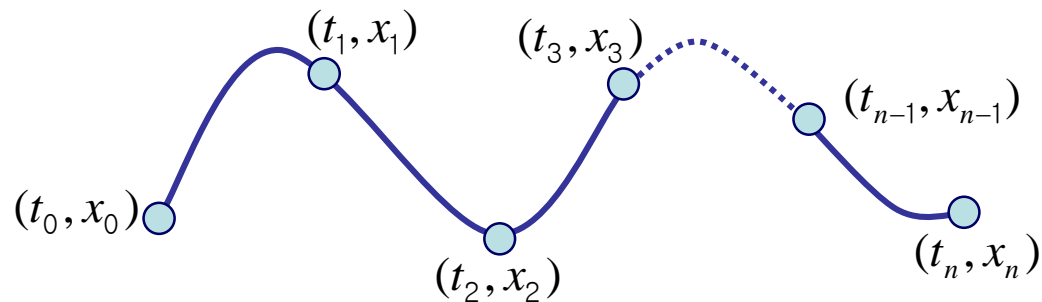


$$\mathbf{p}'(t_i) = \frac{\mathbf{p}_{i+1} - \mathbf{p}_{i-1}}{2}$$

- $C^1$  continuity, local controllability, interpolation

# Natural Cubic Splines

- We want to achieve higher continuity (at least  $C^2$ )
- $4n$  unknowns
  - $n$  Bezier curve segments (4 control points per each segment)
- $4n$  equations
  - $2n$  equations for end point interpolation
  - $(n-1)$  equations for tangential continuity
  - $(n-1)$  equations for second derivative continuity
  - 2 equations:  $x''(t_0) = x''(t_n) = 0$



- $C^2$  continuity, no local controllability, interpolation

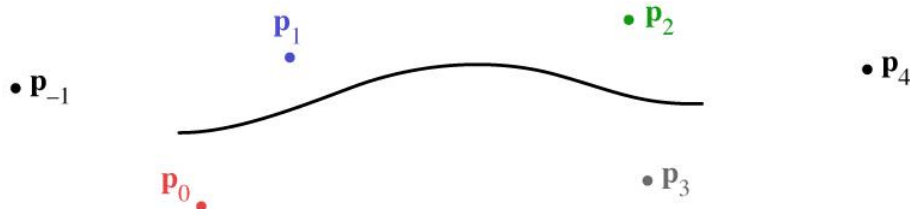


# B-splines (brief intro)

- Use 4 points, but approximate only middle two



- Draw curve with overlapping segments
  - 0-1-2-3, 1-2-3-4, 2-3-4-5, 3-4-5-6, etc



- $C^2$  continuity, local controllability, approximation