实验5: 贝塞尔曲线
Bézier Curve

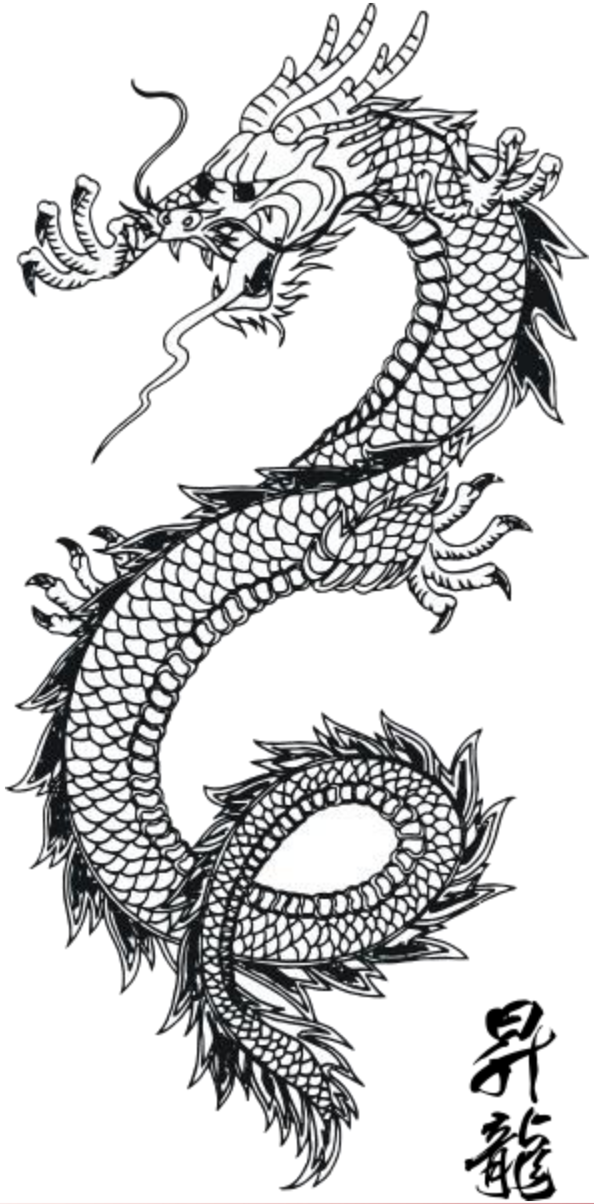
华东师范大学计算机科学与技术学院

李晨 副研究员

cli@cs.ecnu.edu.cn

Today

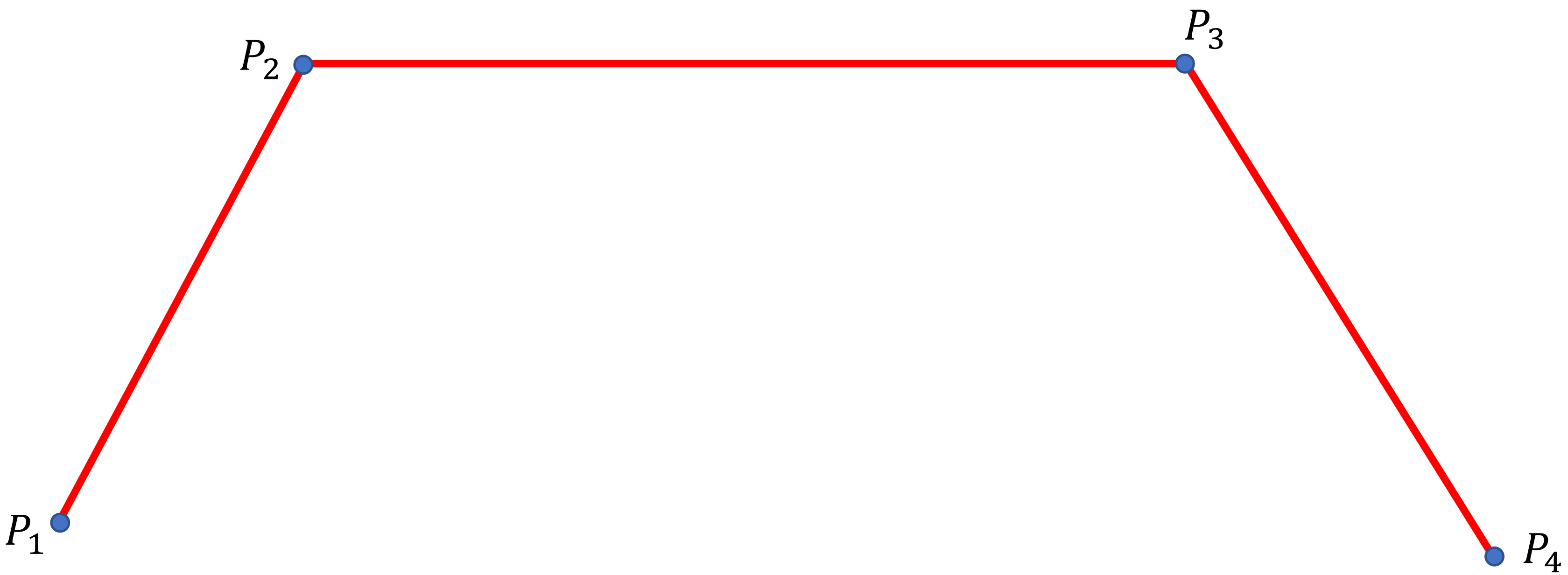
- Smooth curves in 2D
    - 2D illustration
    - Fonts
    - 3D modeling
    - Animation: trajectories



华东师范大学计算机科学与技术学院
School of Computer Science and Technology

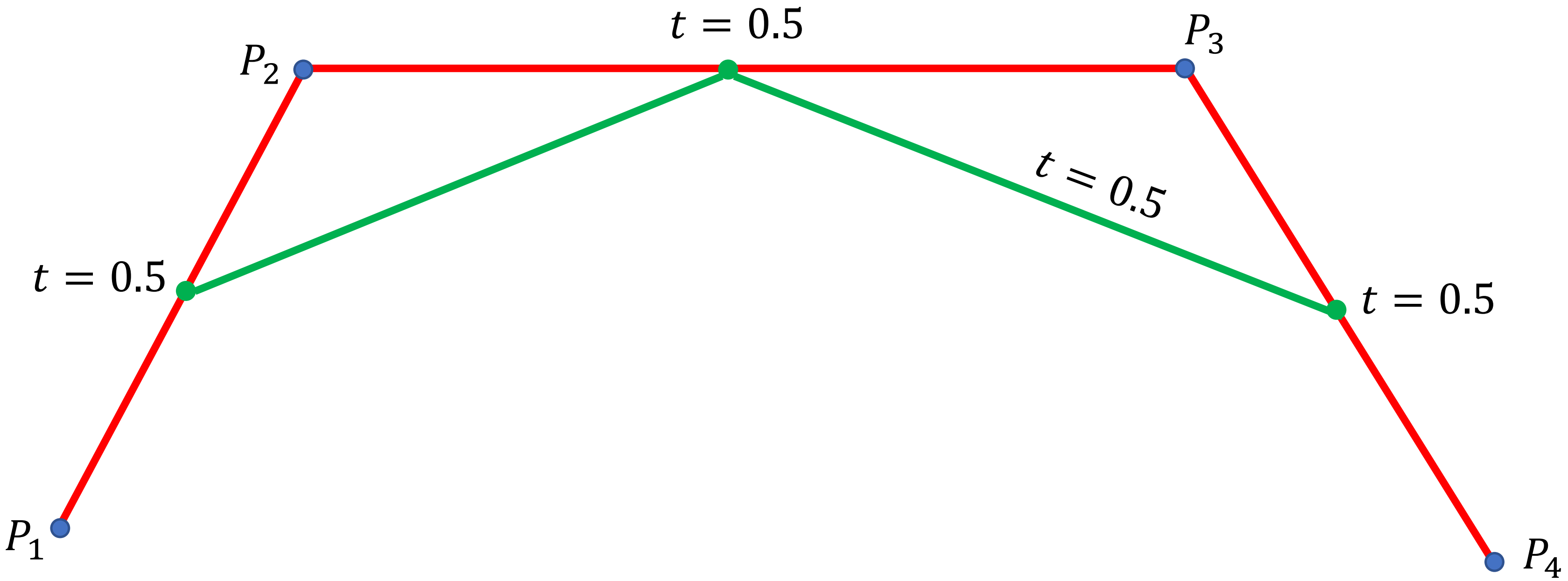
Bézier Curve-de Casteljau Algorithm

- Lets say that I have four control points



Bézier Curve-de Casteljau Algorithm

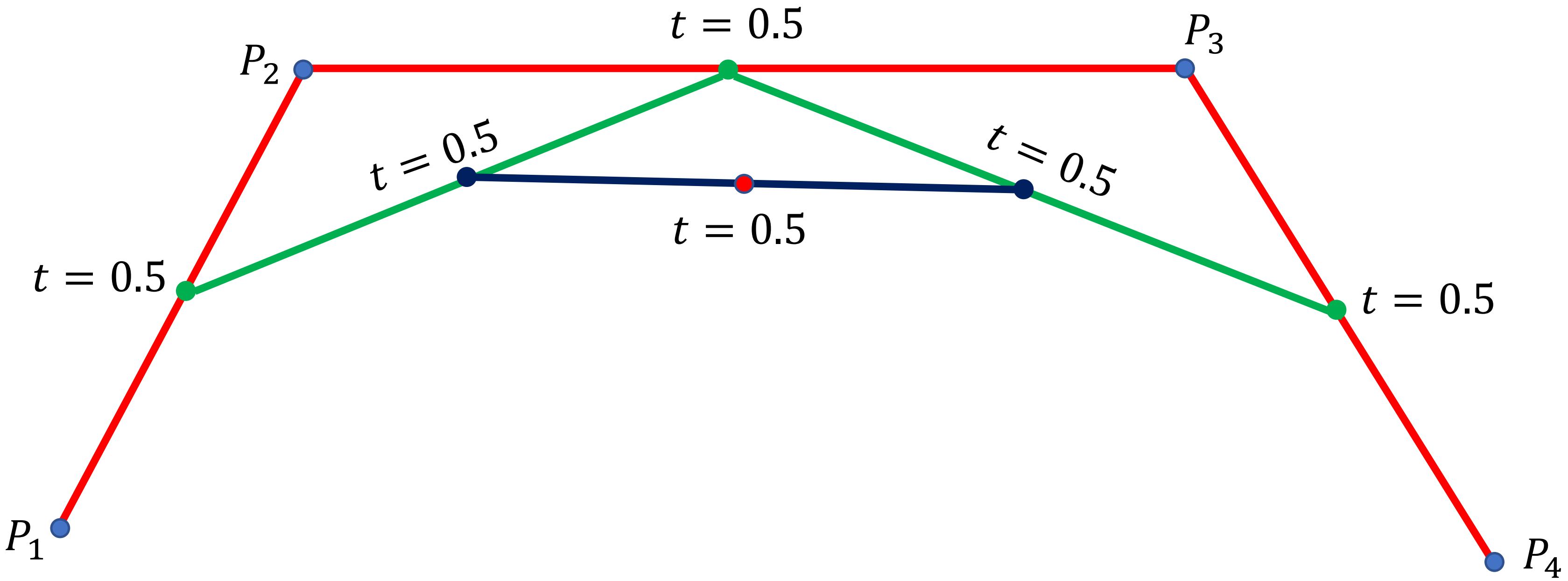
- To find the midpoint of the curve corresponding to those control points:



Connect the point between P_1 and P_2 where $t = 0.5$ with the point between P_3 & P_2 where $t = 0.5$; Do the same with P_2P_3&P_3P_4

Bézier Curve-de Casteljau Algorithm

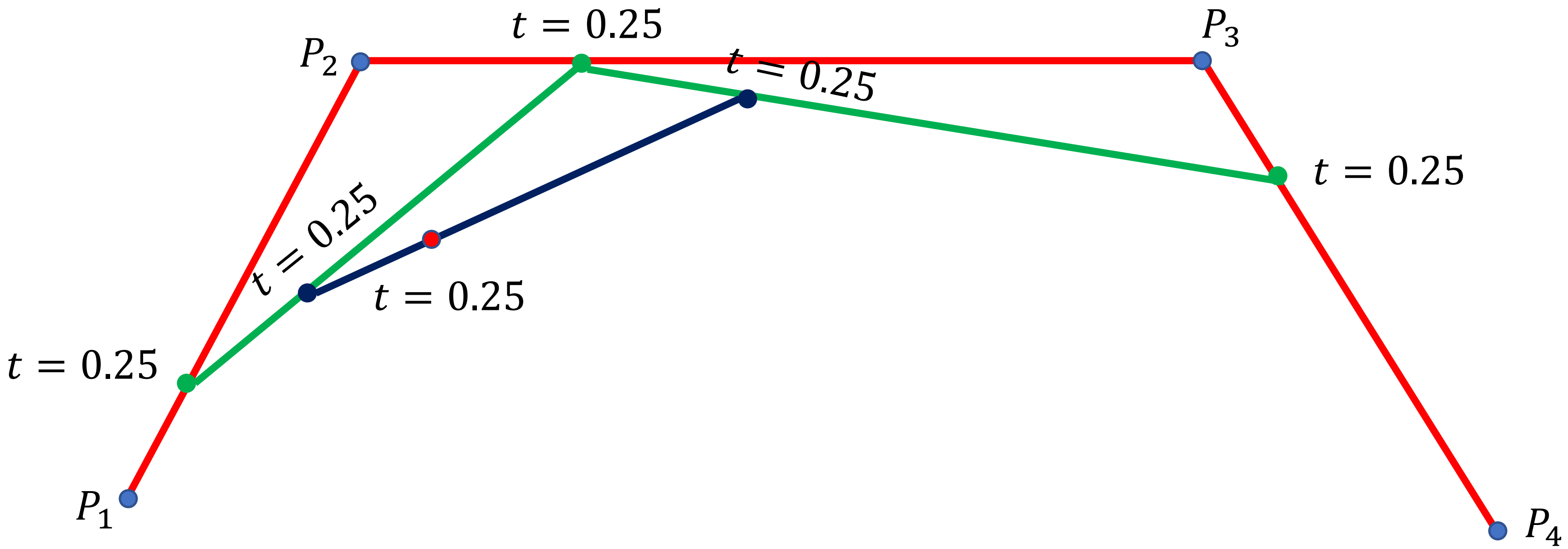
- Now, connect these two lines at their $t = 0.5$ points:



The $t = 0.5$ point on the resulting segment is the midpoint of some type of curve which is made up of weighted averages of the control points

Bézier Curve-de Casteljau Algorithm

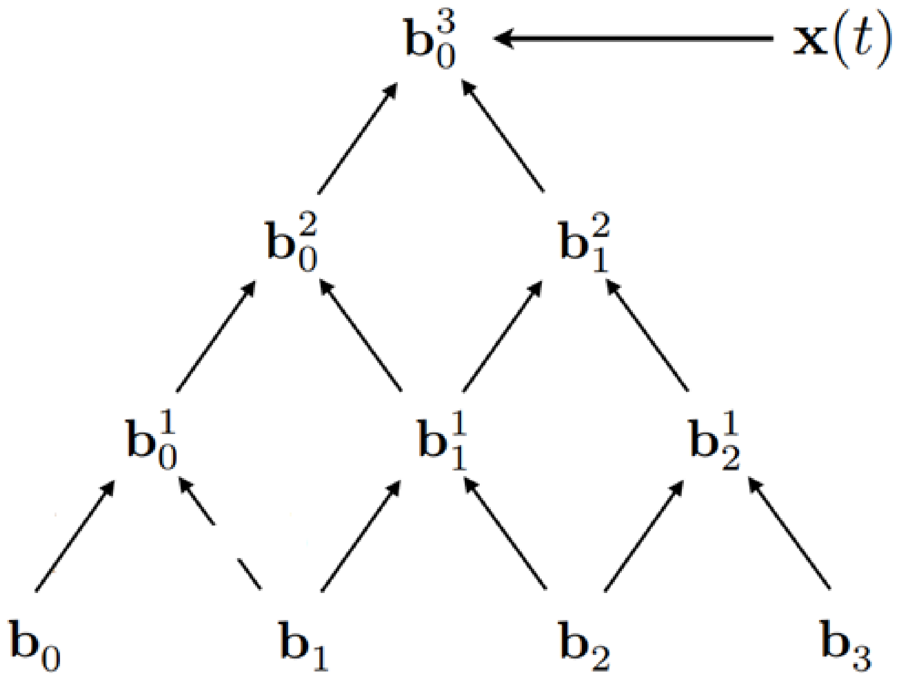
- We can extend this idea to any t value.

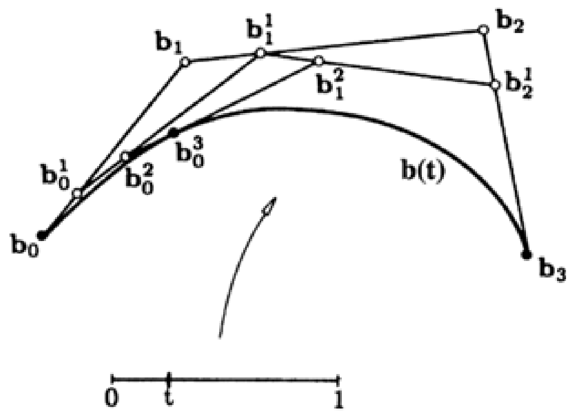


In this way, you can compute the 3rd-order curve for any value

Bézier Curve-Algebraic Formula

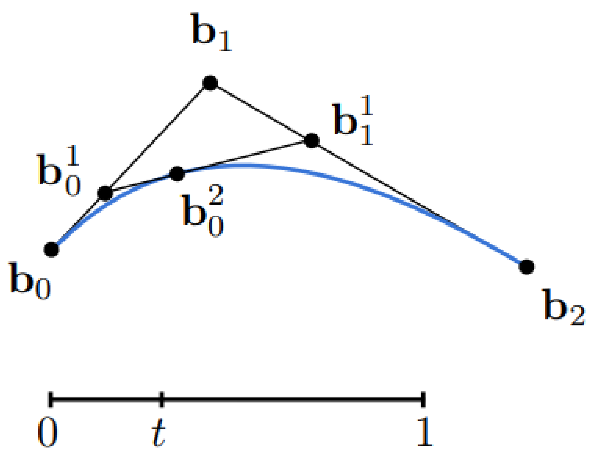
- de Casteljau algorithm gives a pyramid of coefficients





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-General Algebraic Formula

- Bernstein form of a Bézier curve of order *n*

Bézier curve order *n*
(vector polynomial of degree *n*)

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

Bézier control point
(vector in R^N)

Bernstein polynomial
(scalar polynomial of degree *n*)

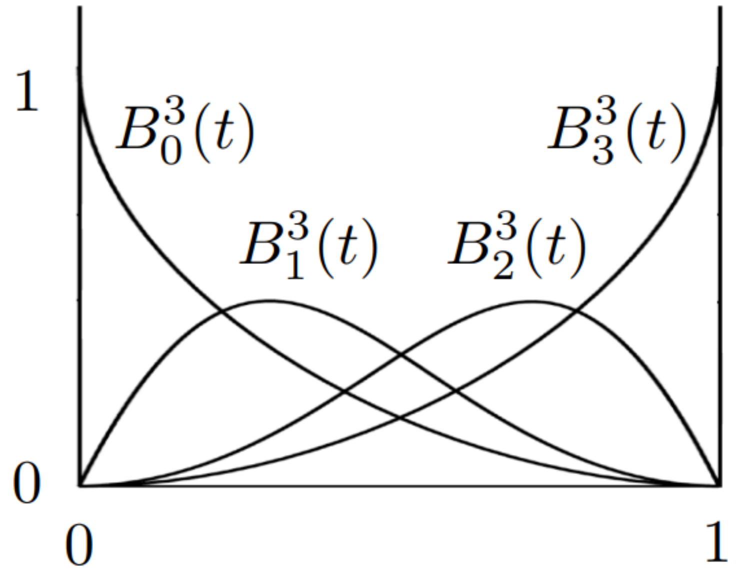
- Bernstein polynomials

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

Cubic Bézier Basis Functions

- Bernstein Polynomials

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

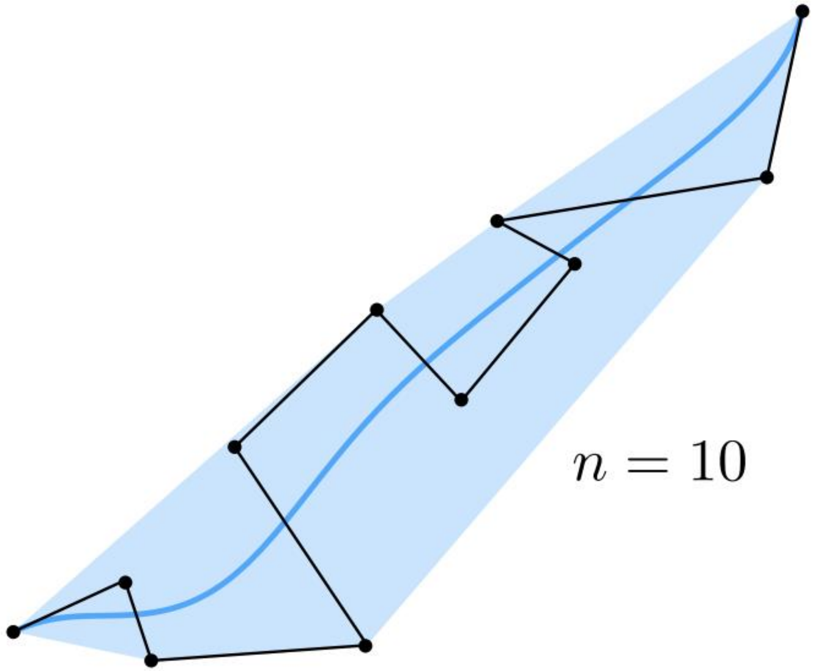


Properties of Bézier Curves

- Interpolates endpoints
  - For cubic Bézier: $b(0) = b_0, b(1) = b_3$
- Tangent to end segments
  - Cubic case: $b'(0) = 3(b_1 - b_0), b'(1) = 3(b_3 - b_2)$
- Affine transformation property
  - Transform curve by transforming control points
- Convex hull property
  - Curve is within convex hull of control points

Piecewise Bézier Curves

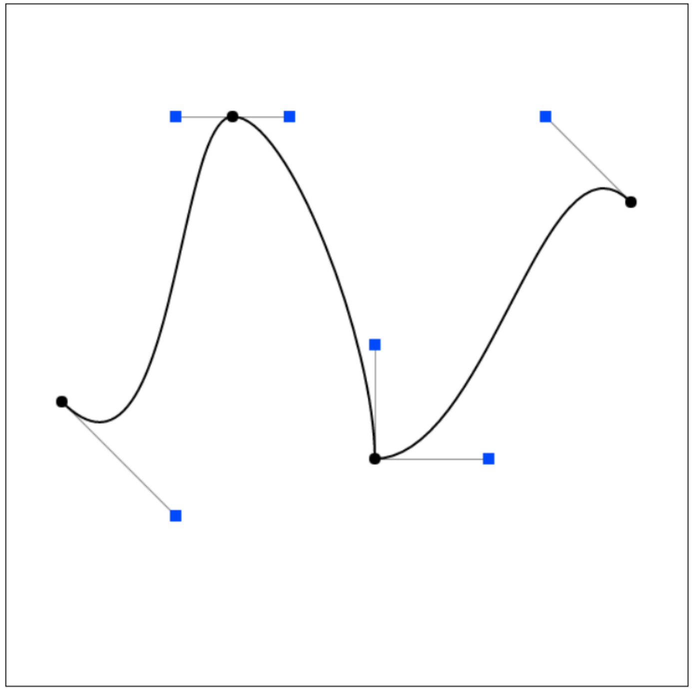
- Higher-order Bézier curves?



Very hard to control!
Uncommon

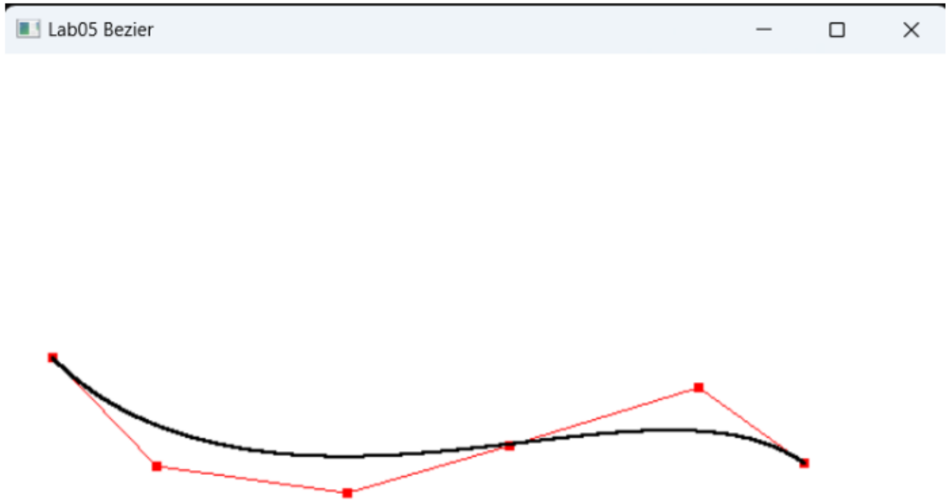
Piecewise Bézier Curves

- Instead, chain many low-order Bézier curve



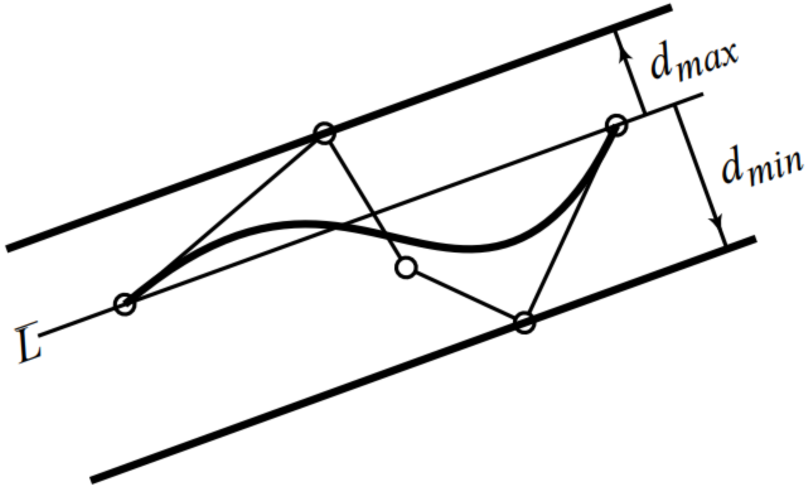
Assignment

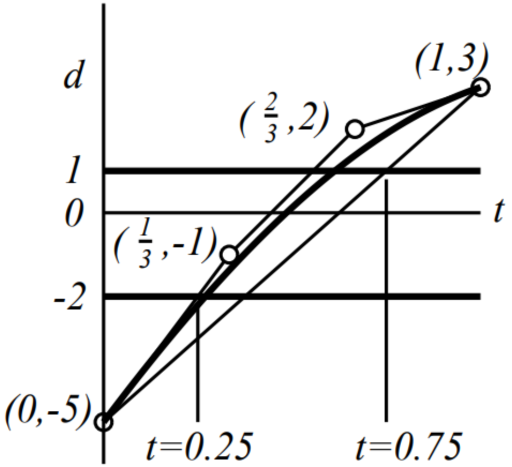
- 实验编号：4
- 实验名称：贝塞尔曲线
- 实验内容
  - 实现贝塞尔曲线绘制



Extra Credit

- Could you clip a Bézier curve?
  - Fat lines
  - Parameter *t*





https://scholarsarchive.byu.edu/cgi/viewcontent.cgi?article=1000&context=facpub

Reference

- https://en.wikipedia.org/wiki/B%C3%A9zier_curve
- https://en.wikipedia.org/wiki/De_Casteljau%27s_algorithm
- https://math.hws.edu/graphicsbook/demos/c2/cubic-bezier.html