12 - Designing Curves: Polynomials and Interpolation

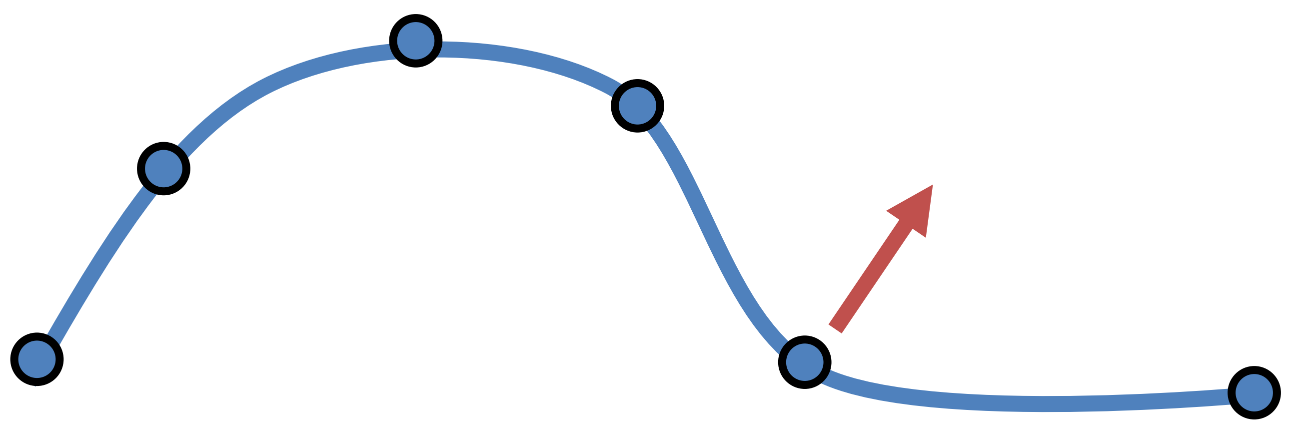
Acknowledgement: Daniele Panozzo, Olga Sorkine-Hornung,
Alexander Sorkine-Hornung, Ilya Baran

Florida State University

Basic idea for curve design

- User gives us points. We need to connect the dots in a smooth way.

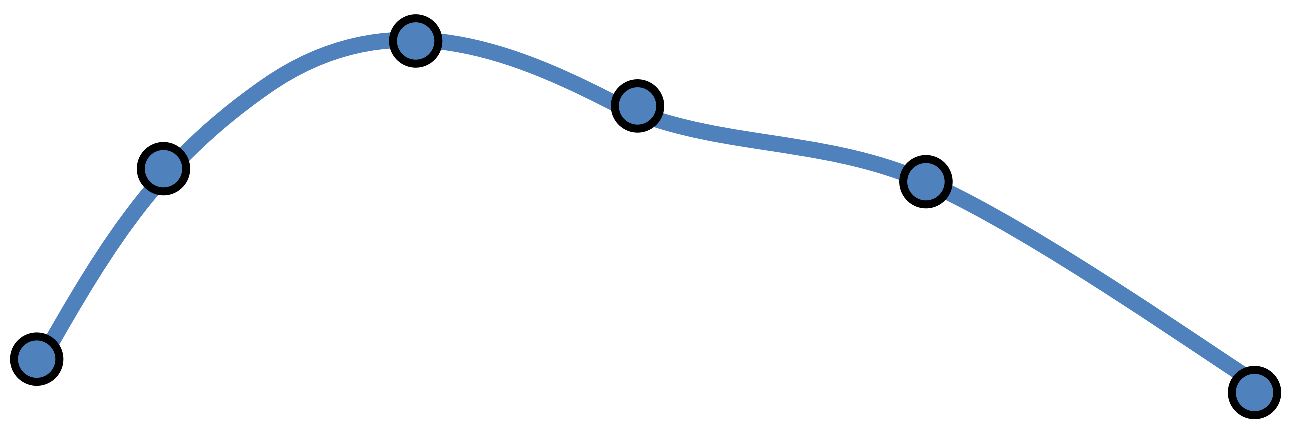
- The dots stay as “handles” on the curve. User can move the dots and the curve follows.



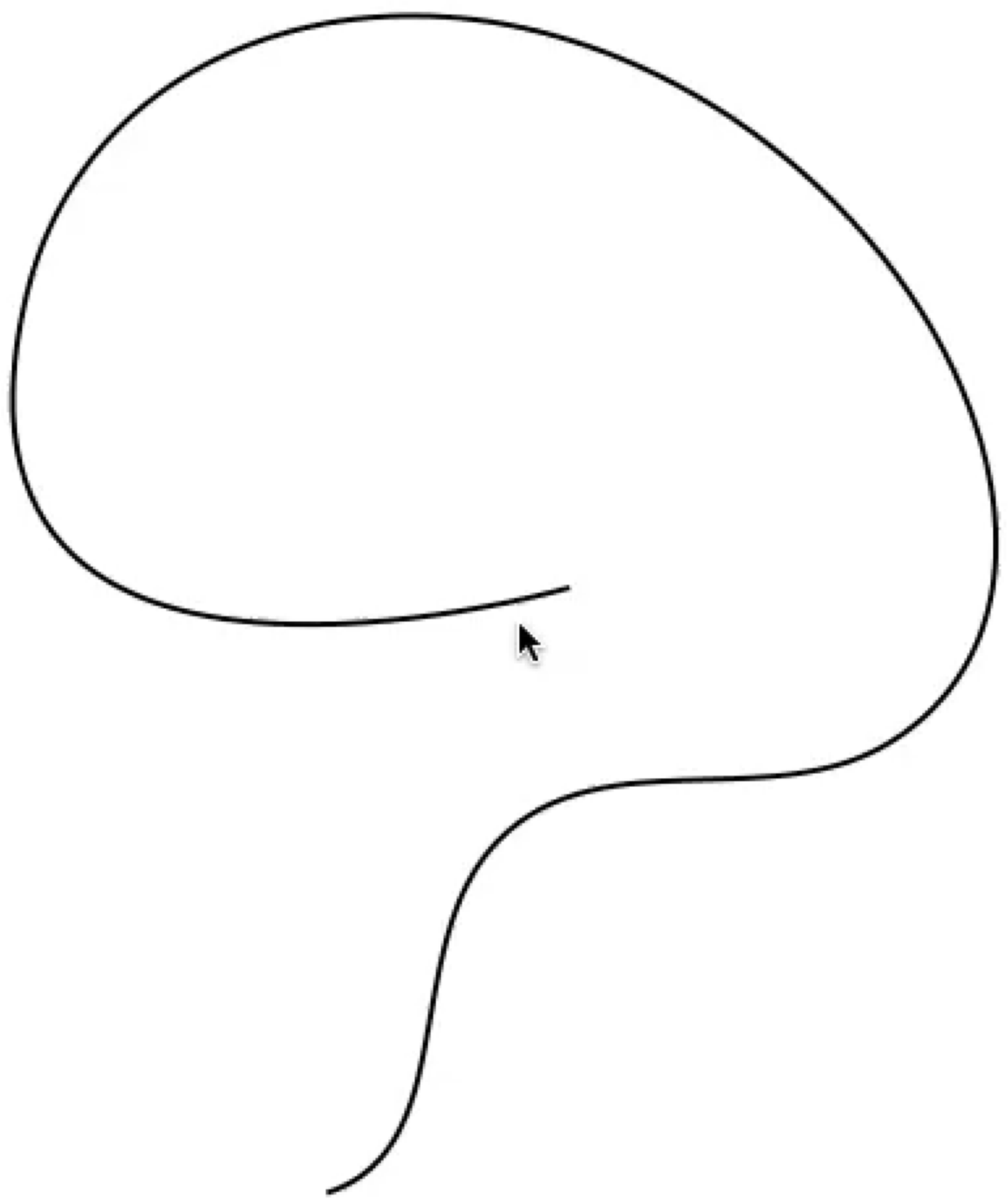
Basic idea for curve design

- User gives us points. We need to connect the dots in a smooth way.

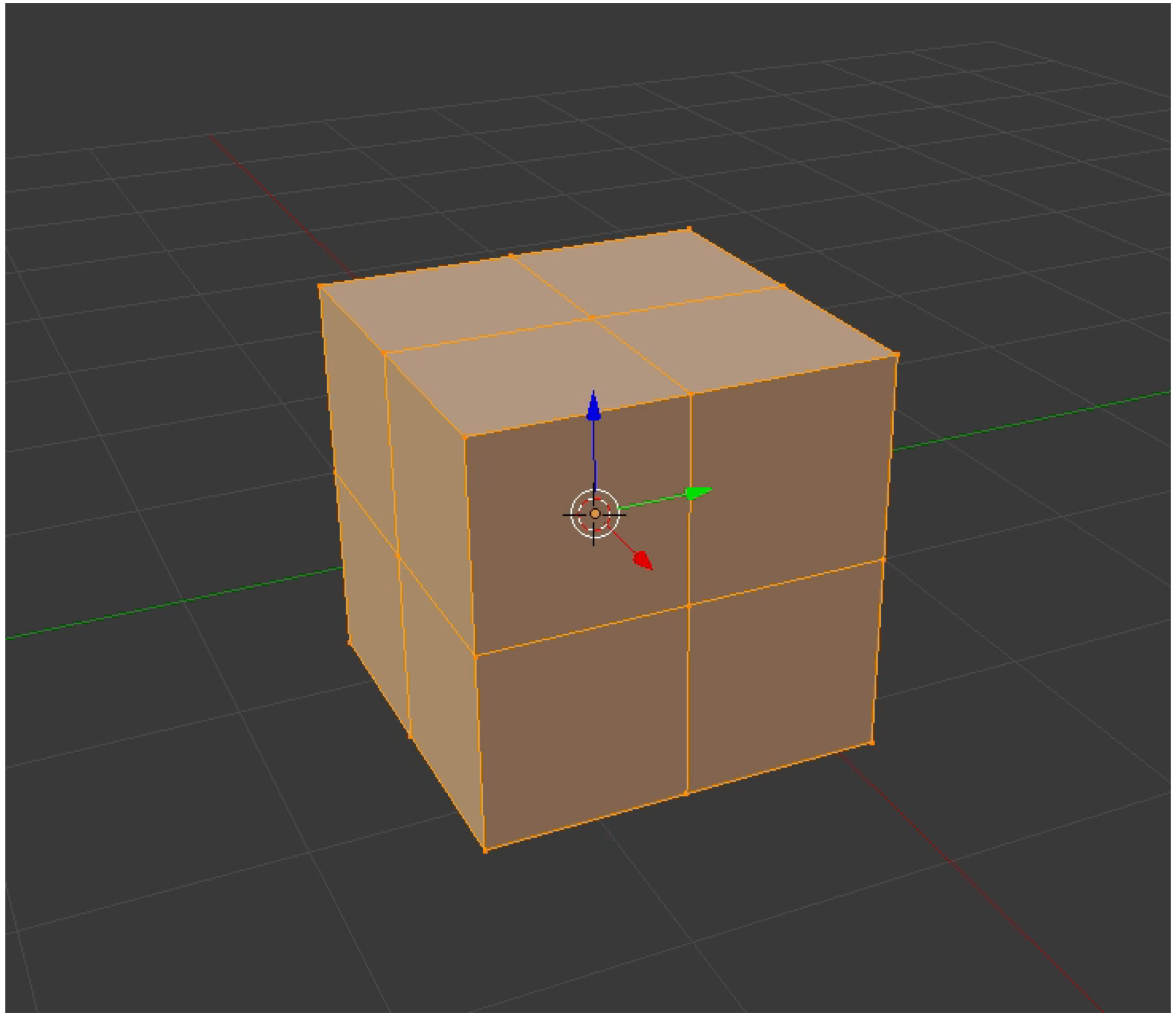
- The dots stay as “handles” on the curve. User can move the dots and the curve follows.



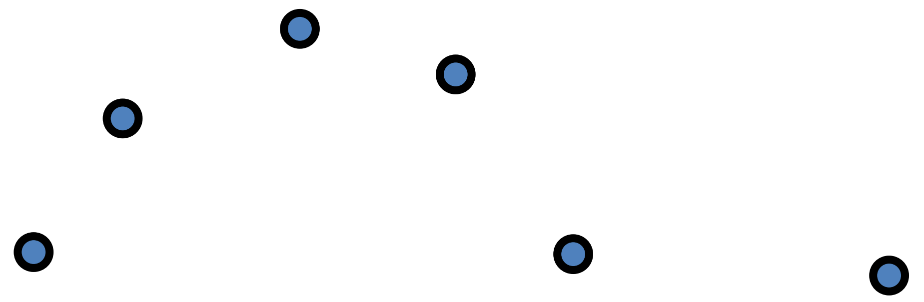
Keynote Demo



Blender Demo



Polynomial curves



- Polynomials

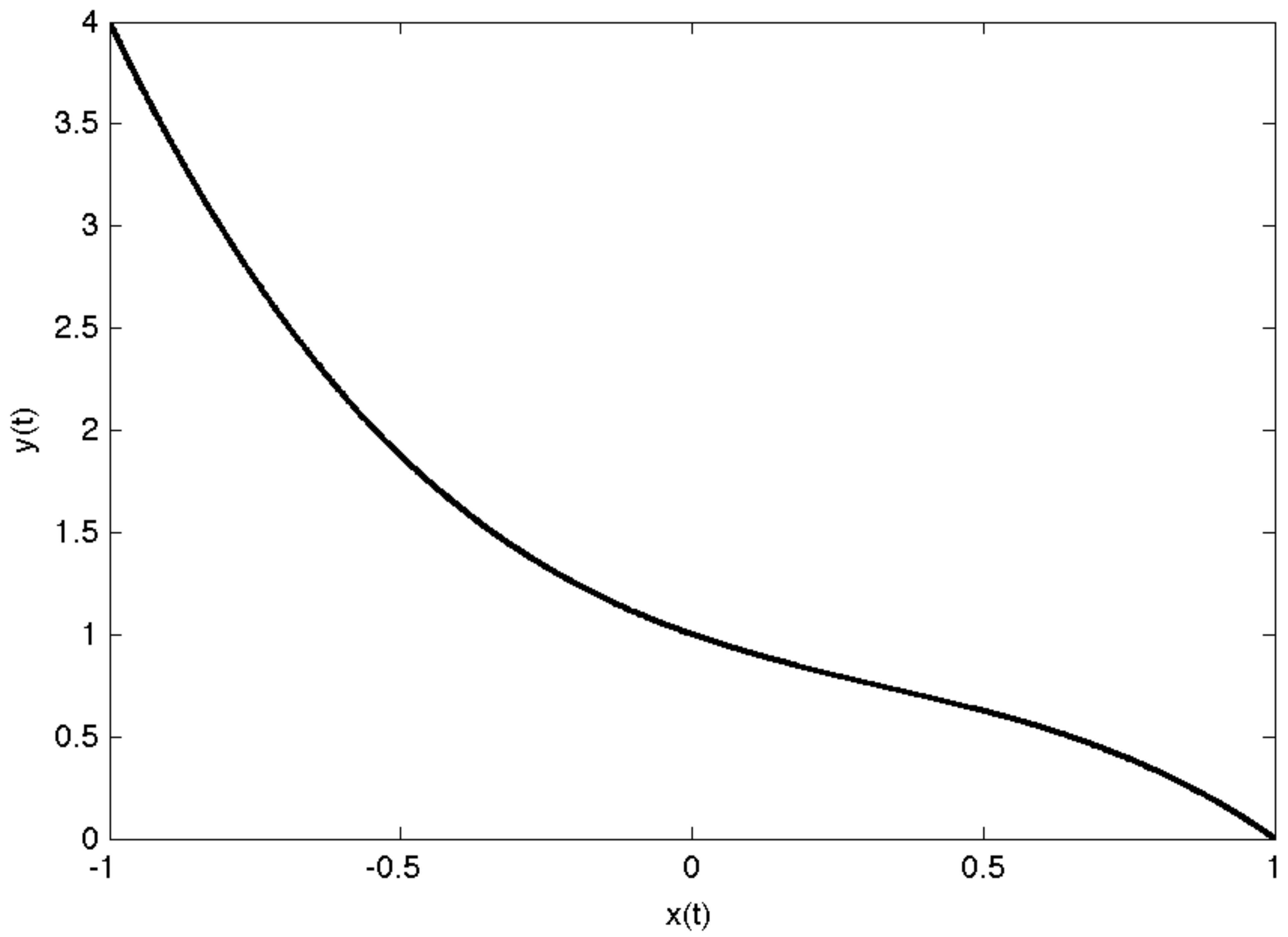
$$\mathbf{p}(t) = \begin{pmatrix} x_0 + x_1 t + x_2 t^2 + \cdots \\ y_0 + y_1 t + y_2 t^2 + \cdots \end{pmatrix} = $$
$$= \mathbf{p}_0 + \mathbf{p}_1 t + \mathbf{p}_2 t^2 + \cdots$$

- For degree d we need $d + 1$ points (“coefficients”)

Polynomials

- Parametric form with polynomials

$$\begin{aligned} x(t) &= t \\ y(t) &= 1 - t + t^2 - t^3 \end{aligned}$$

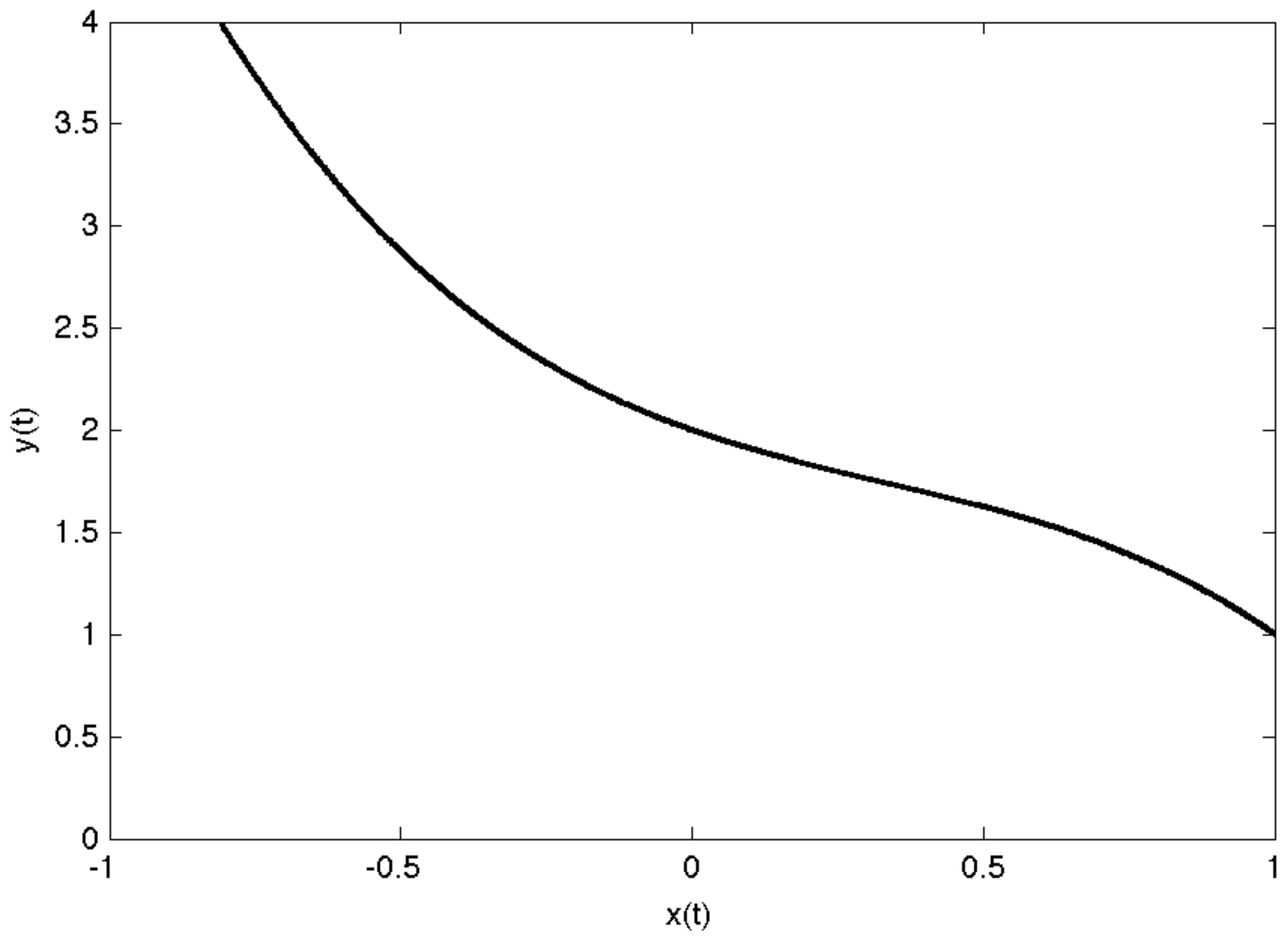


- Control shape?

Polynomials

- Parametric form with polynomials

$$x(t) = t$$
$$y(t) = 2 - t + t^2 - t^3$$

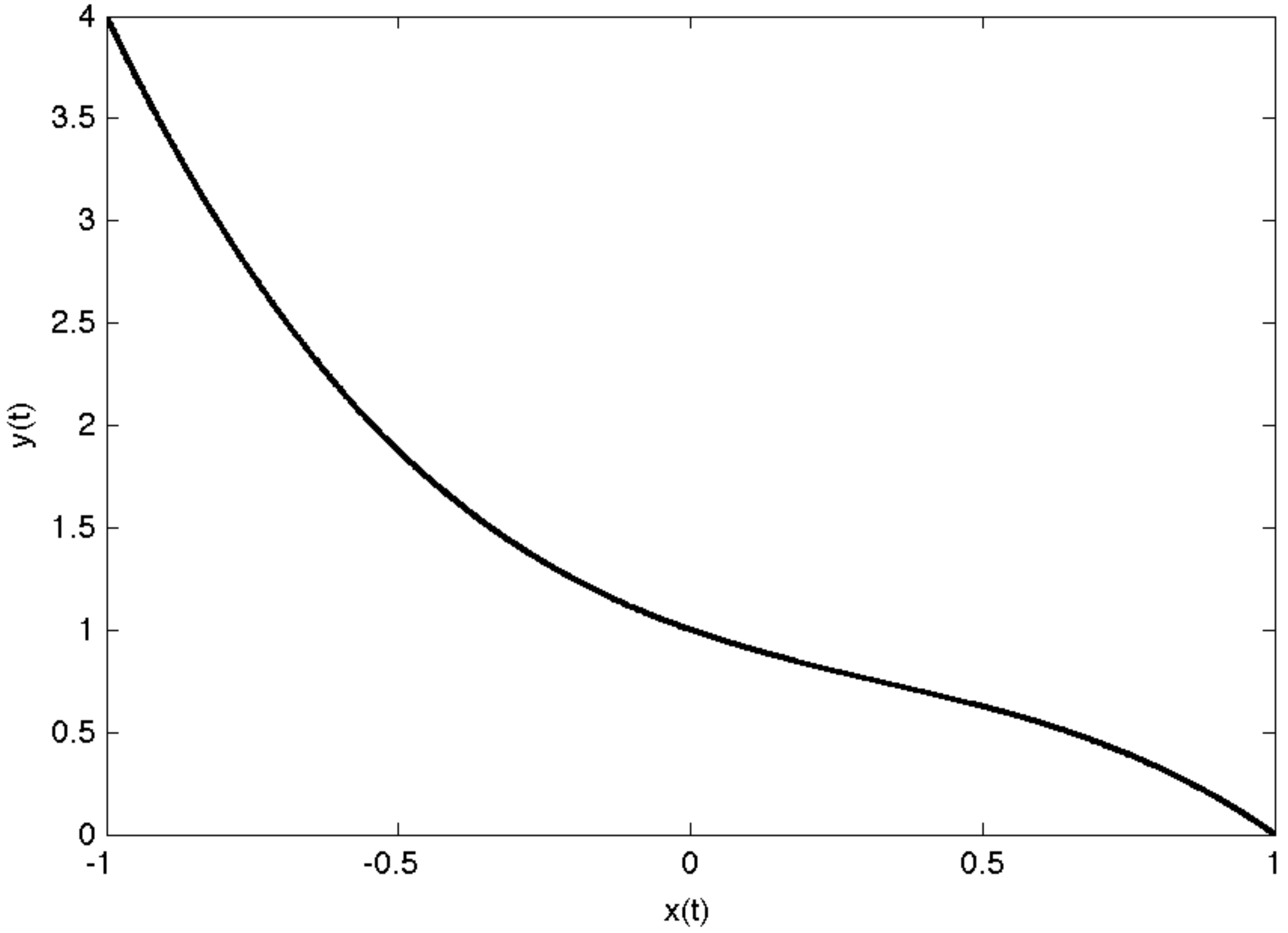


- Control shape?

Polynomials

- Parametric form with polynomials

$$\begin{aligned} x(t) &= t \\ y(t) &= 1 - t + t^2 - t^3 \end{aligned}$$

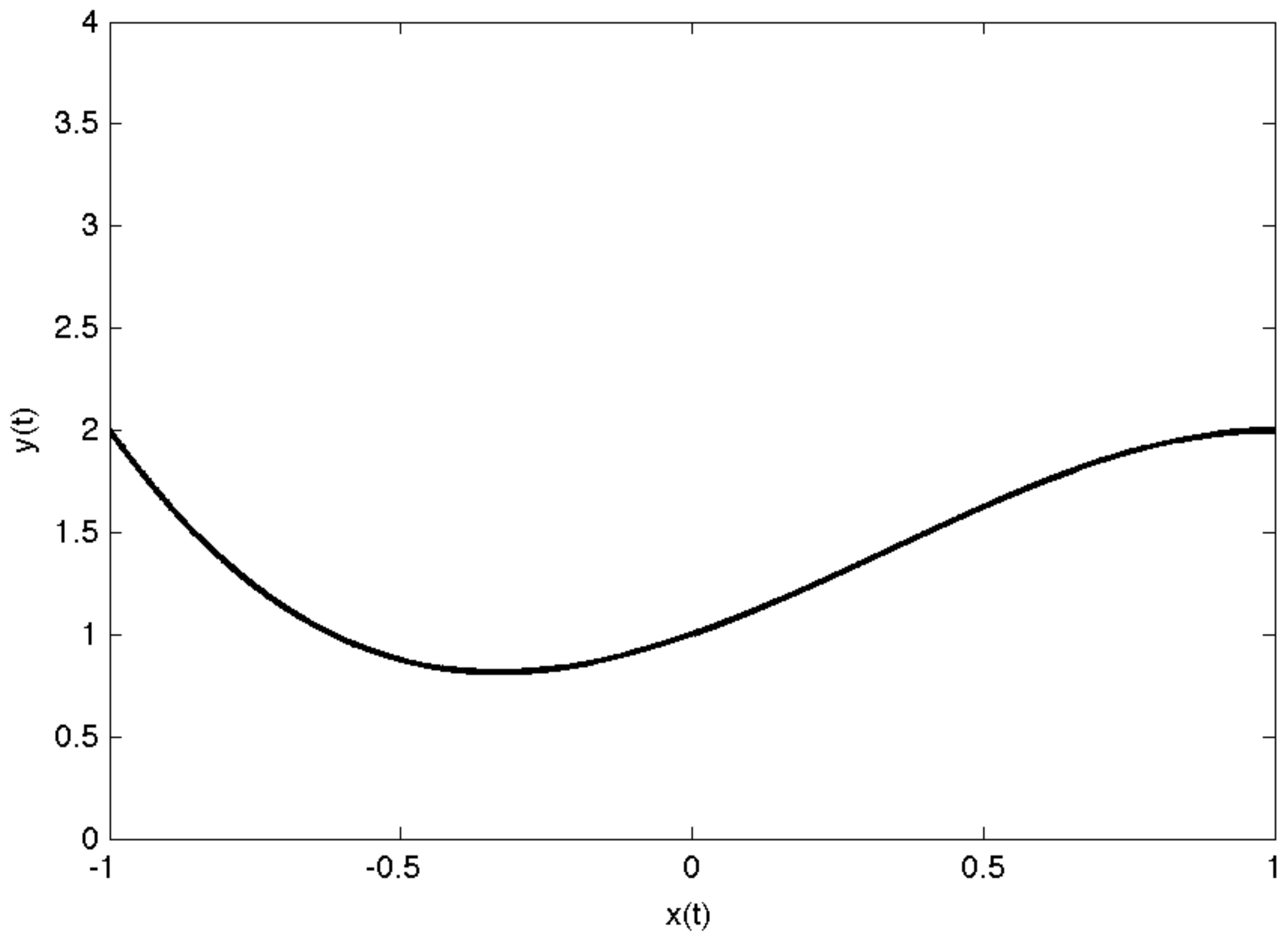


- Control shape?

Polynomials

- Parametric form with polynomials

$$
\begin{aligned}
x(t) &= t \\
y(t) &= 1 + t + t^2 - t^3
\end{aligned}
$$

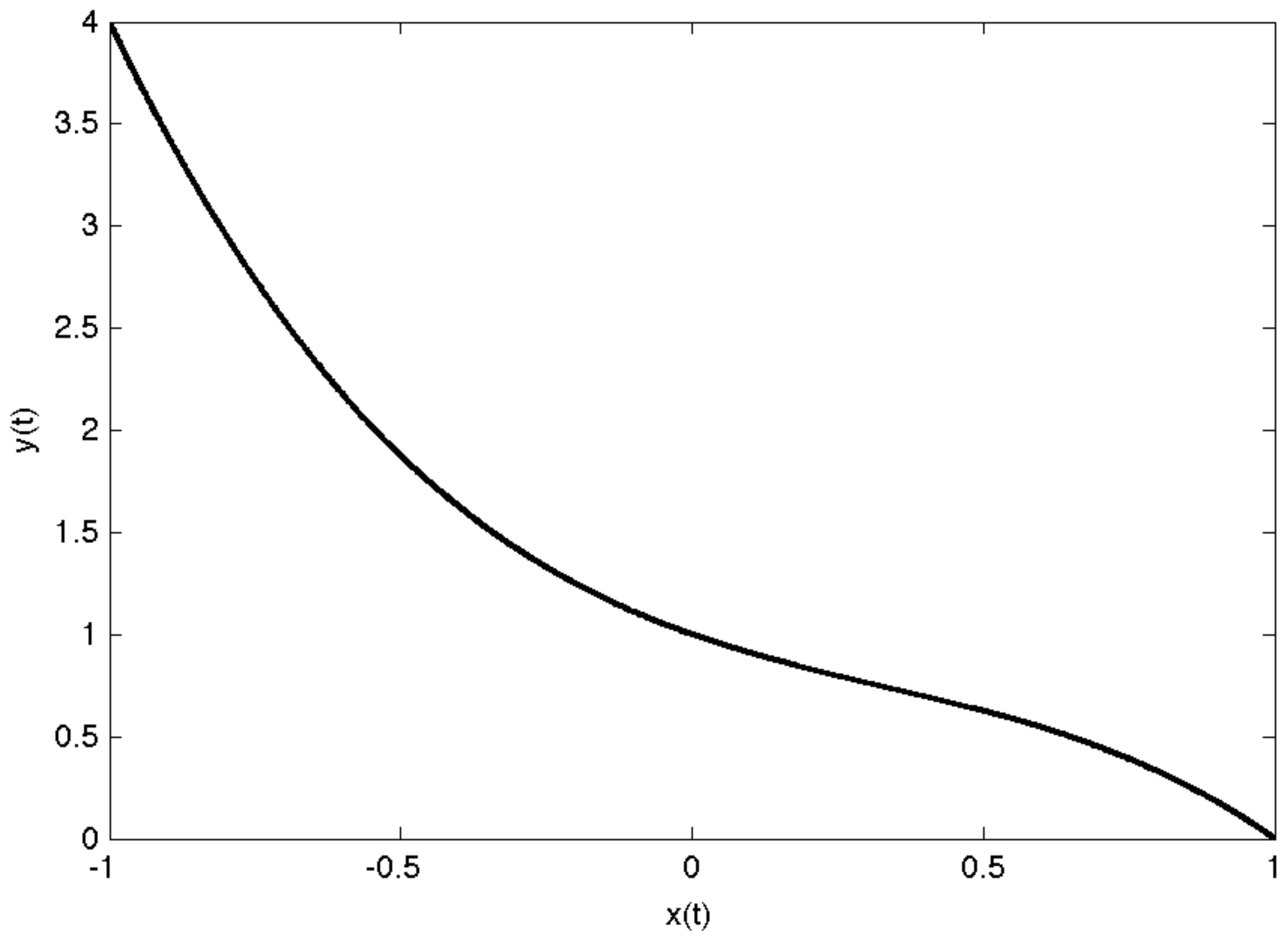


- Control shape?

Polynomials

- Parametric form with polynomials

$$\begin{aligned} x(t) &= t \\ y(t) &= 1 - t + t^2 - t^3 \end{aligned}$$



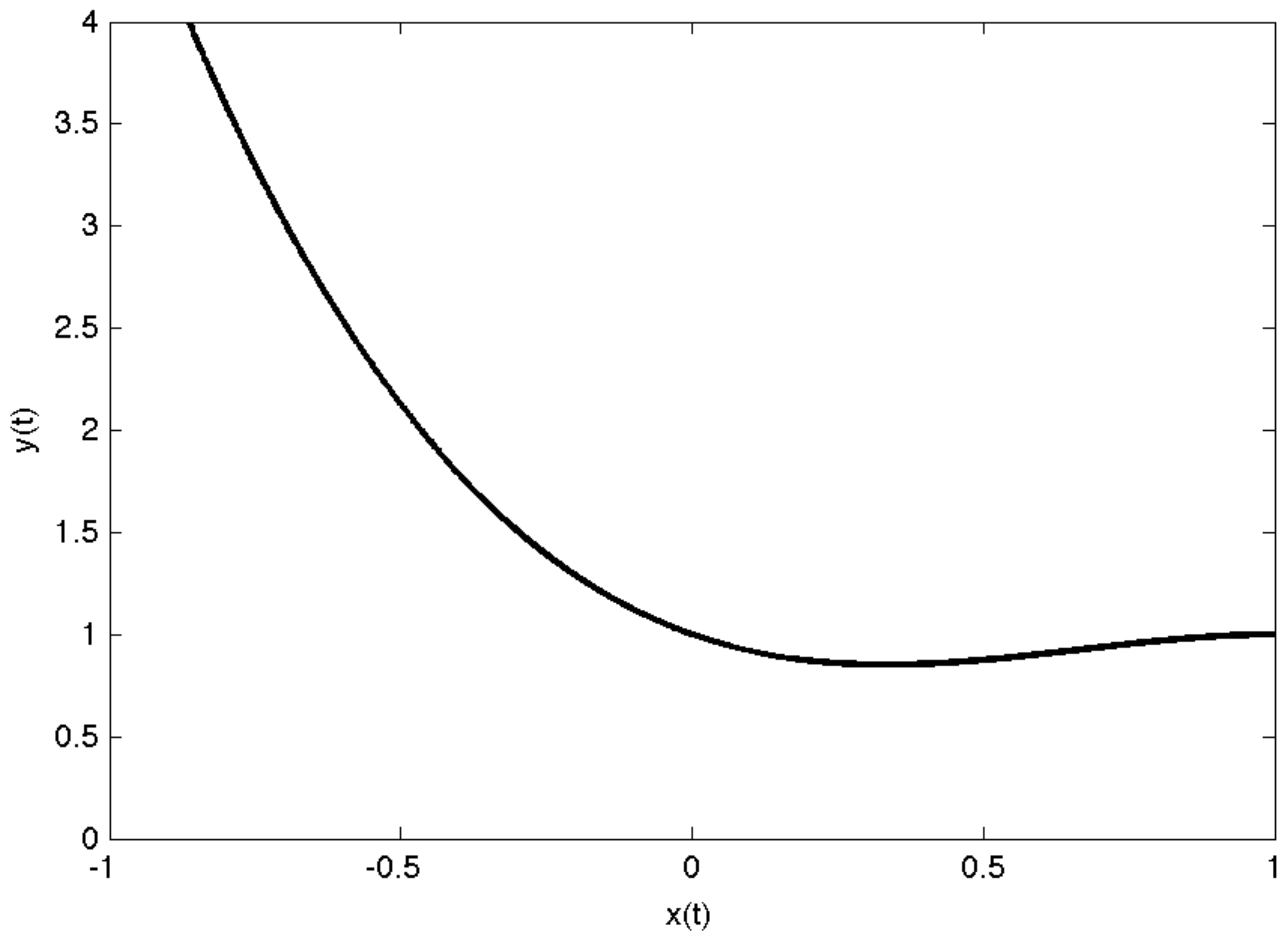
- Control shape?

Polynomials

- Parametric form with polynomials

$$x(t) = t$$
$$y(t) = 1 - t + 2t^2 - t$$

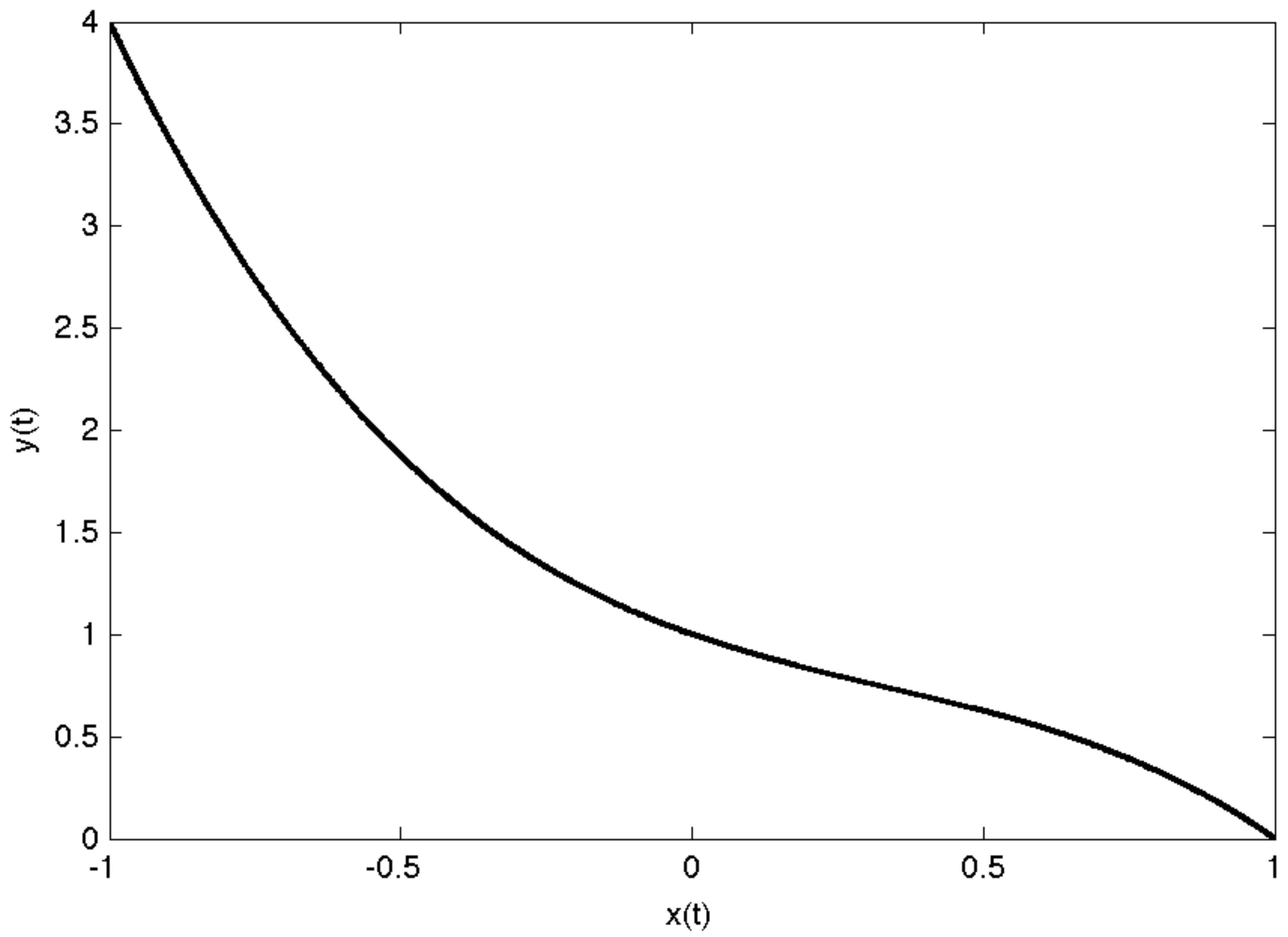
- Control shape?



Polynomials

- Parametric form with polynomials

$$
\begin{aligned}
x(t) &= t \\
y(t) &= 1 - t + t^2 - t^3
\end{aligned}
$$

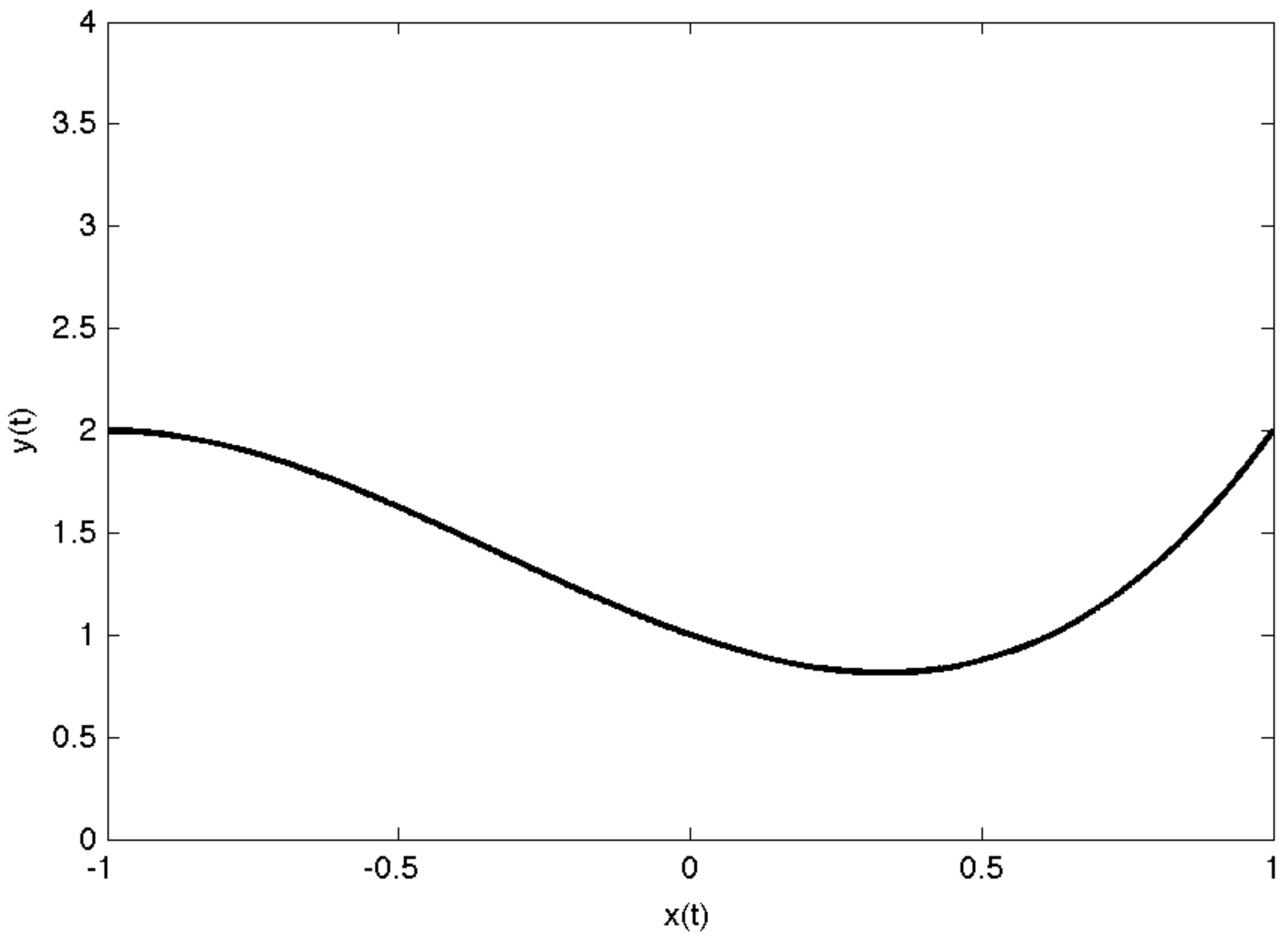


- Control shape?

Polynomials

- Parametric form with polynomials

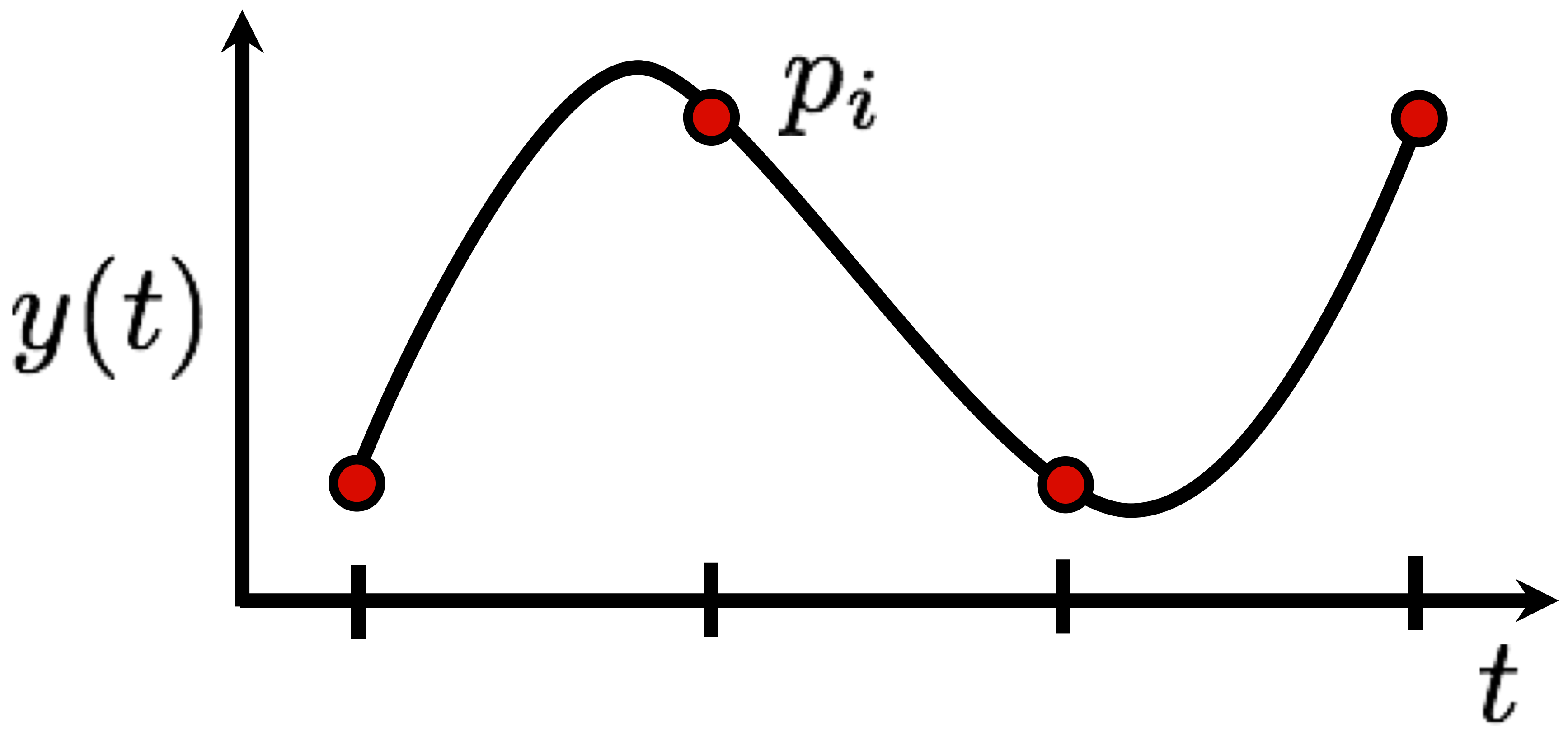
$$
\begin{aligned}
x(t) &= t \\
y(t) &= 1 - t + t^2 + t^3
\end{aligned}
$$



- Control shape?
- Monomial coefficients are not intuitive

Interpolation (1D)

- Interpolate control points
- Find polynomial with $y(t_i) = p_i$

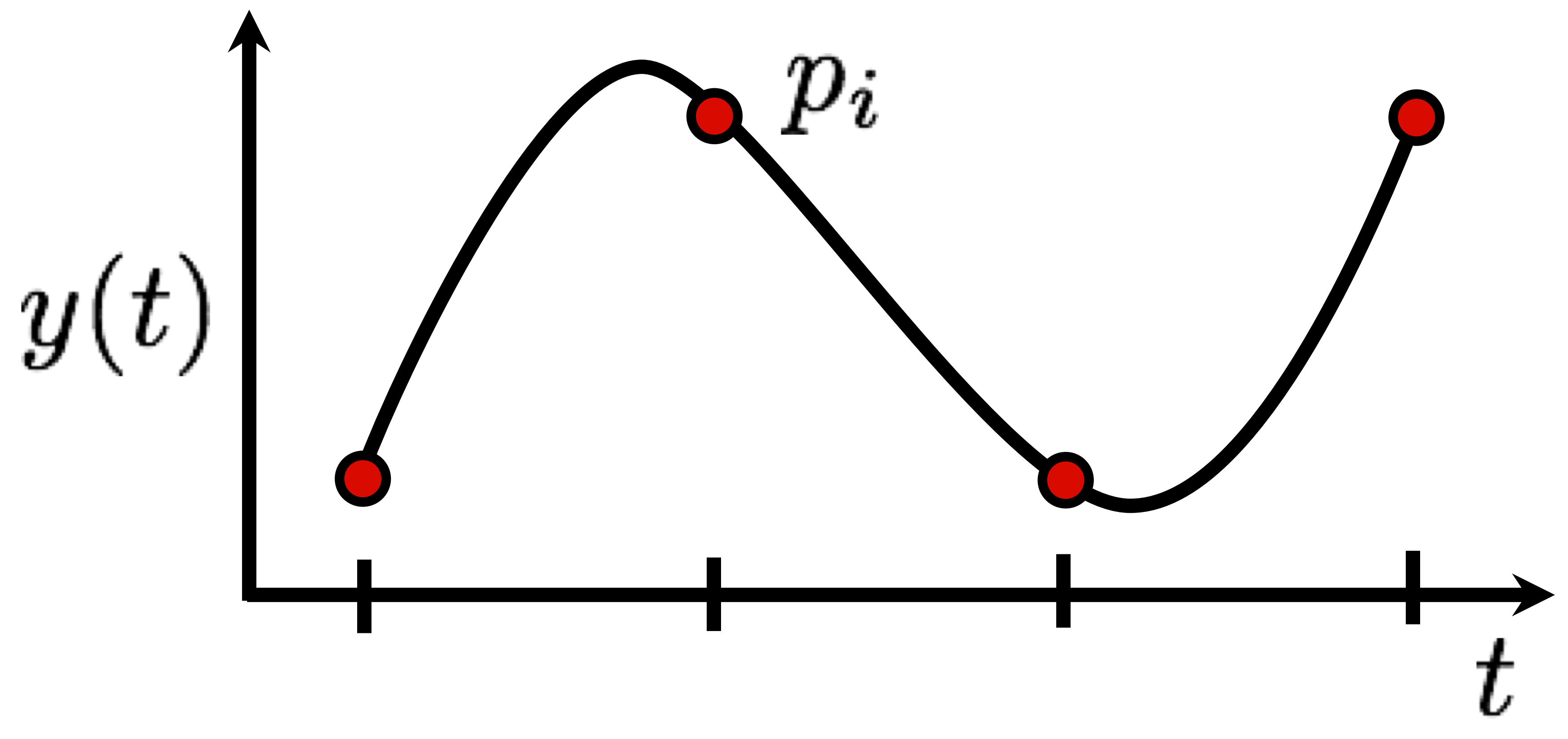


Interpolation (1D)

- Interpolate control points
- Find polynomial
  with $y(t_i) = p_i$

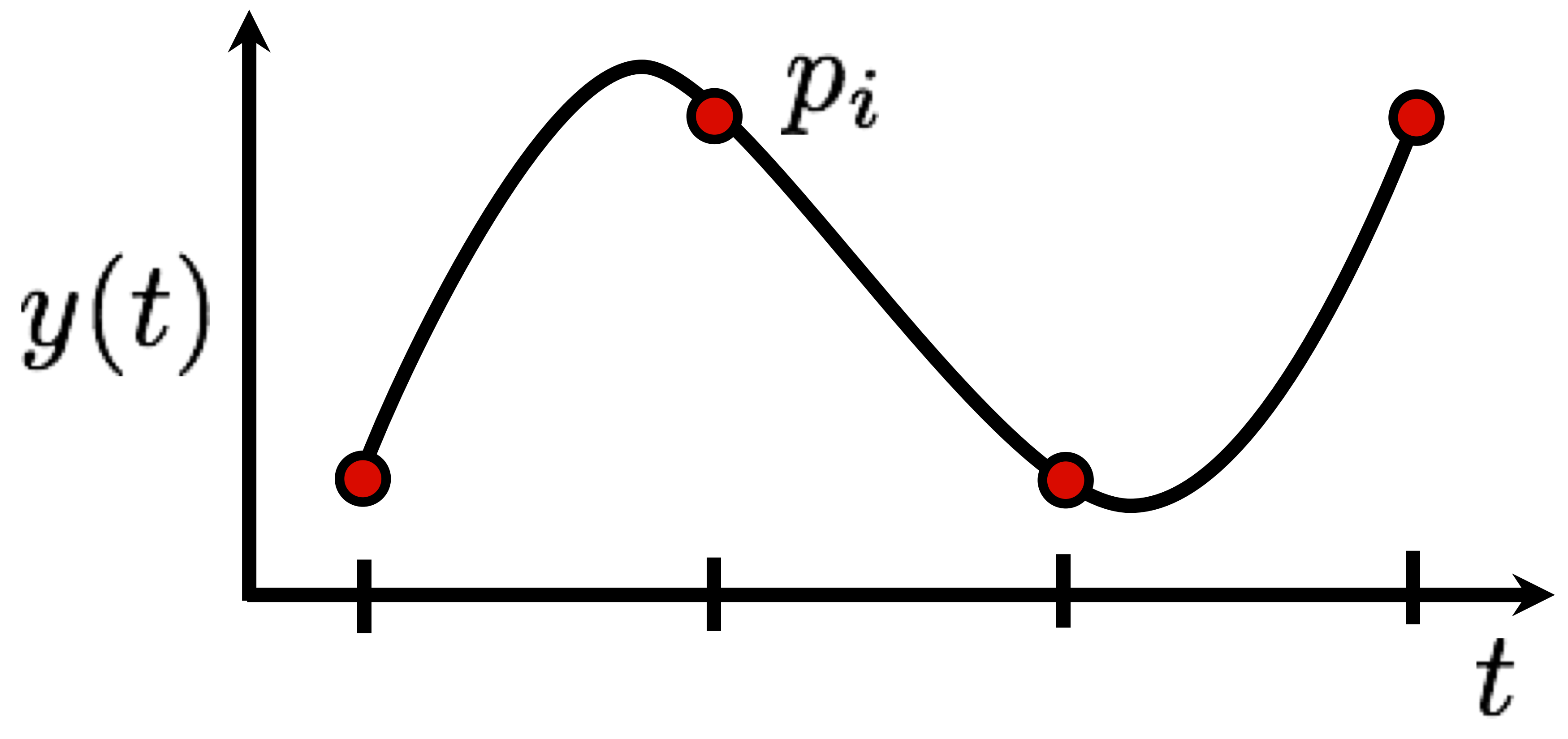
$$y(t) = c_0 + c_1 t + c_2 t^2 + c_3 t^3$$

$$\begin{pmatrix} 1 & t & t^2 & t^3 \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{pmatrix} = y(t)$$



Interpolation (1D)

- Interpolate control points
- Find polynomial
  with $y(t_i) = p_i$
- Solve

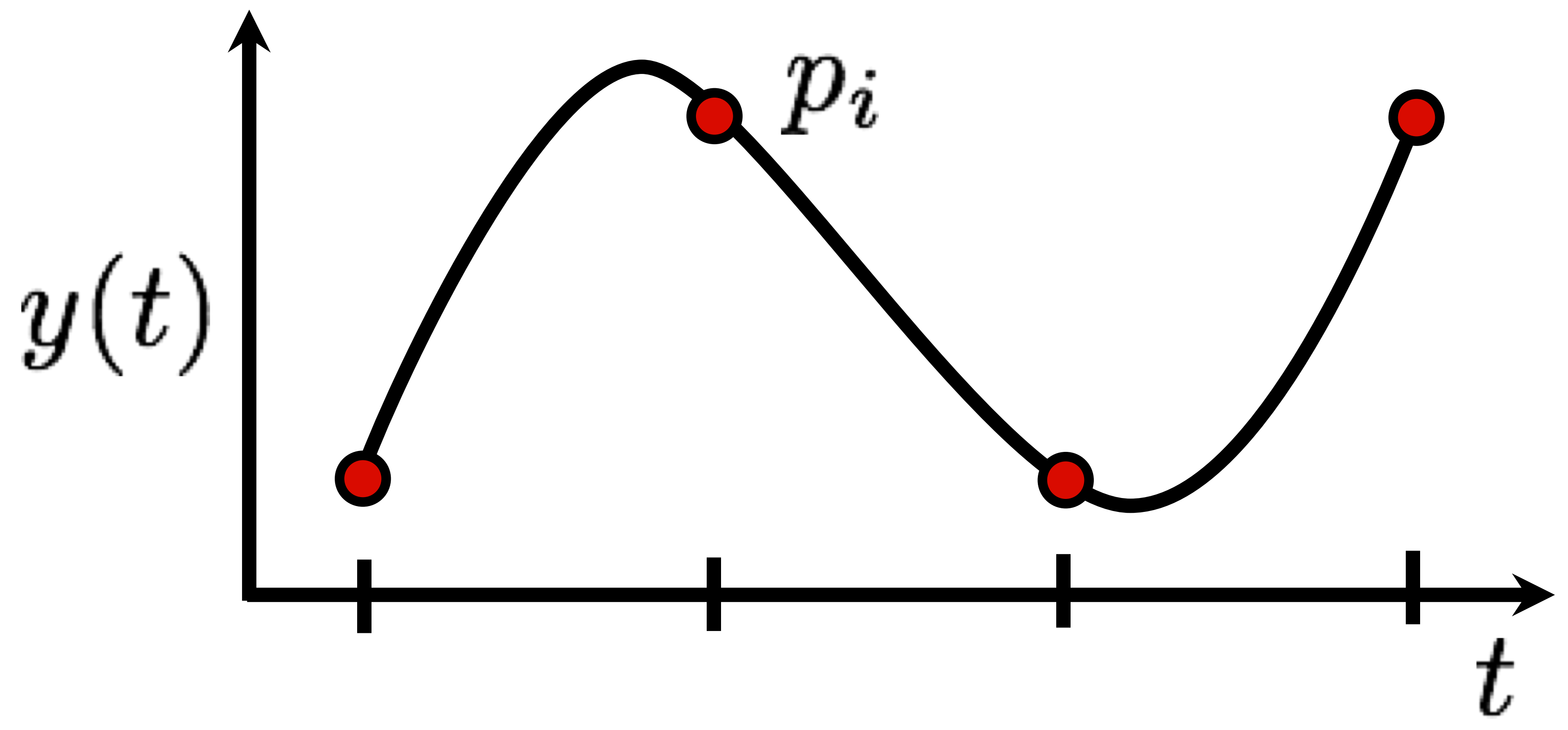


$$\begin{pmatrix} 1 & t_0 & t_0^2 & t_0^3 \\ 1 & t_1 & t_1^2 & t_1^3 \\ 1 & t_2 & t_2^2 & t_2^3 \\ 1 & t_3 & t_3^2 & t_3^3 \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{pmatrix}$$

Vandermonde matrix

Interpolation (1D)

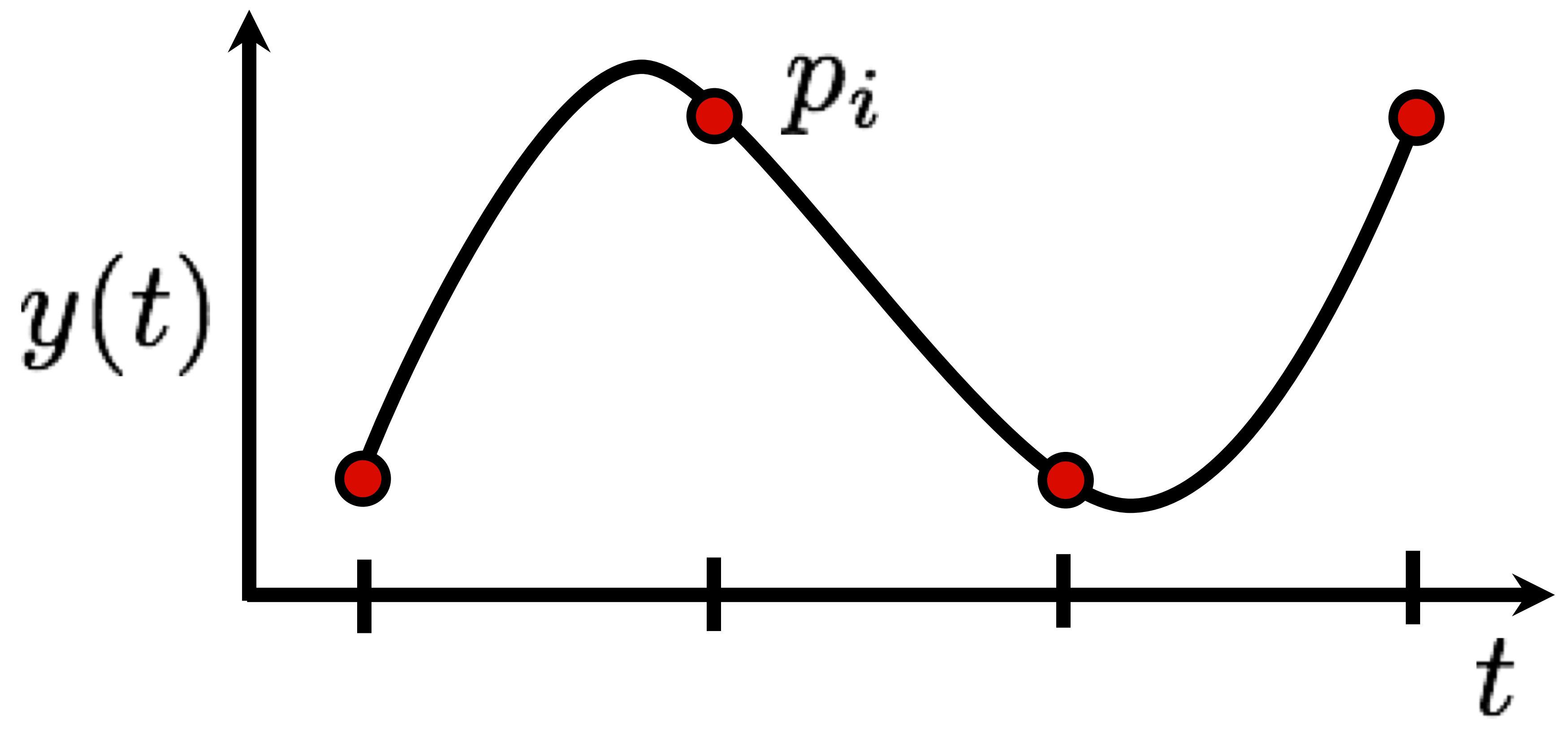
- Interpolate control points
- Find polynomial
  with $y(t_i) = p_i$
- Solve



$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 8 \\ 1 & 3 & 9 & 27 \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 3 \\ 1 \\ 3 \end{pmatrix}$$

Interpolation (1D)

- Interpolate control points
- Find polynomial with $y(t_i) = p_i$
- Solve



$$\begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} 1 & t_0 & t_0^2 & t_0^3 \\ 1 & t_1 & t_1^2 & t_1^3 \\ 1 & t_2 & t_2^2 & t_2^3 \\ 1 & t_3 & t_3^2 & t_3^3 \end{pmatrix}^{-1} \begin{pmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{pmatrix}$$

Interpolation (1D)

- Polynomial fitting can be done explicitly:

$$y(t) = \sum_i p_i \prod_{j \neq i} \frac{t - t_j}{t_i - t_j} = \sum_i p_i L_i(t)$$

- Check that $y(t_i) = p_i$

- Products $L_i(t)$ are called Lagrange polynomials

Example of 1D Interpolation

Recap

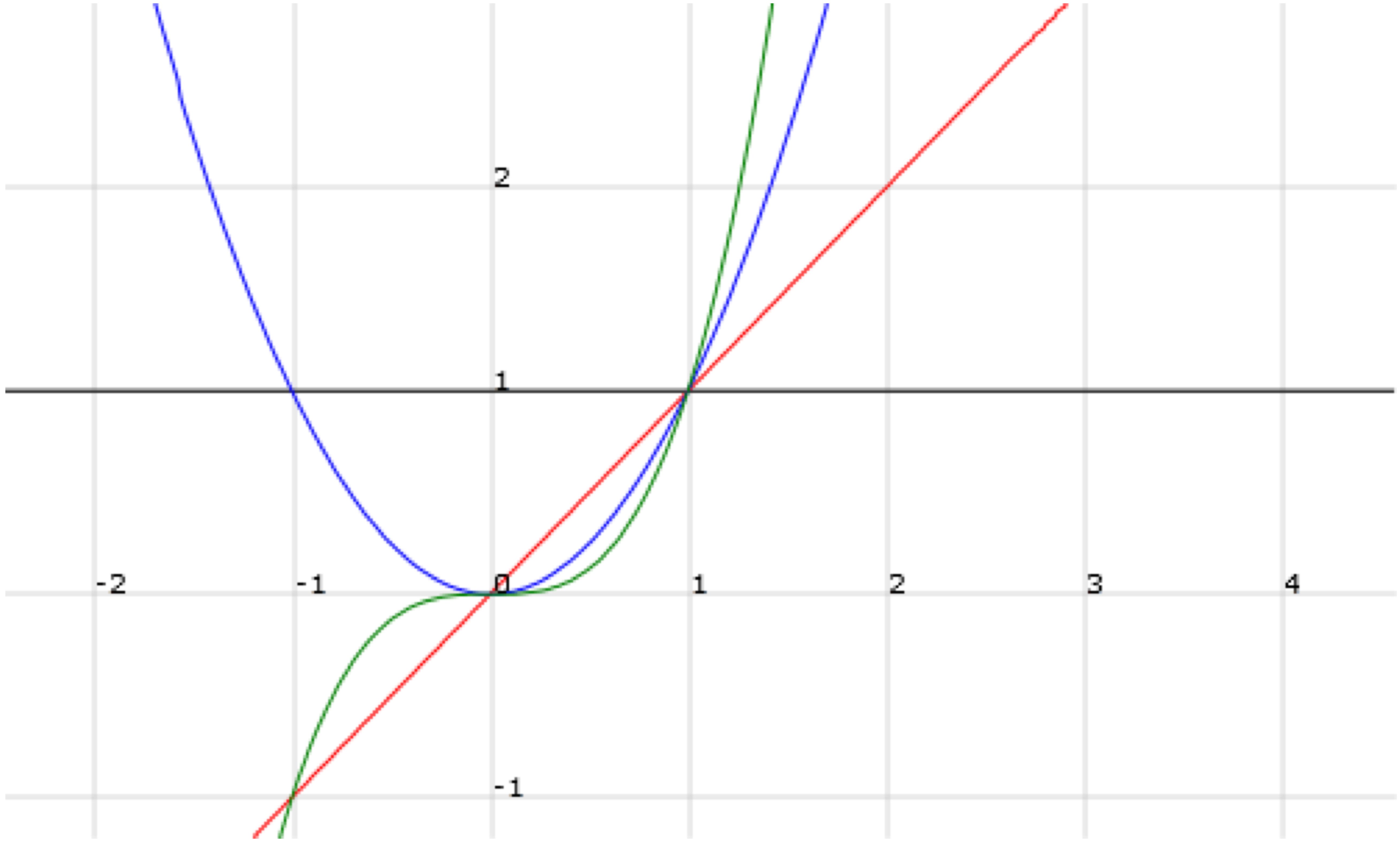
- Monomial coefficients unintuitive
- Lagrange interpolation is better, but there is a global effect

Polynomials form a vector space

- You can add and subtract them
- You can multiply them by real numbers
- These operations follow the usual rules
  - Associativity, commutativity
  - Distributivity for scalar multiplication
  - Etc.
- Anything you can do with vectors in general, you can do with polynomials.

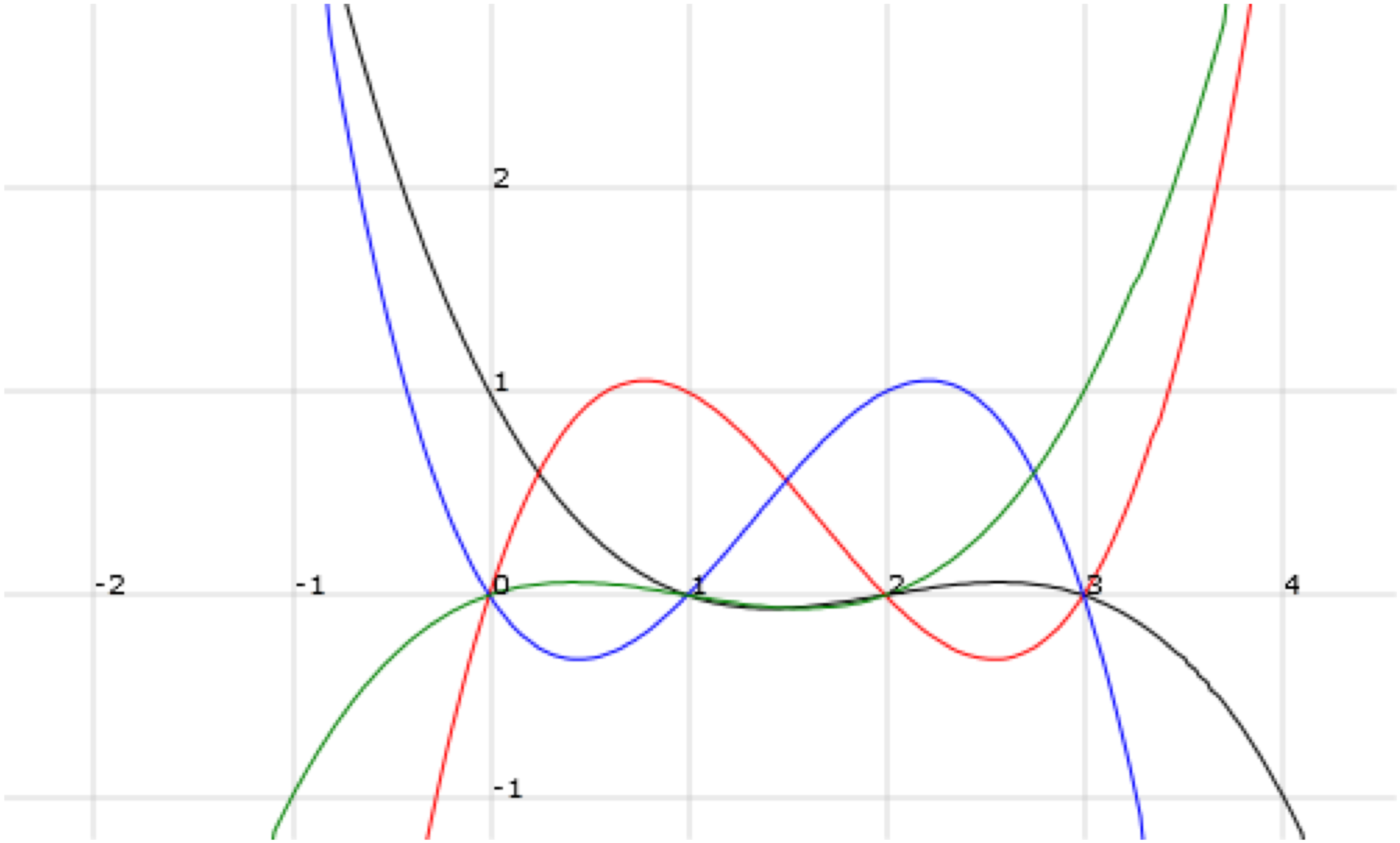
Monomial basis for cubics:

$$(\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3, \mathbf{b}_4) = (1, t, t^2, t^3)$$



Interpolation basis for $t = (0, 1, 2, 3)$:

$$\frac{1}{6}(-t^3 + 6t^2 - 11t + 6, \quad 3t^3 - 15t^3 + 18t, \quad -3t^3 + 12t^2 - 9t, \quad t^3 - 3t^2 + 2t)$$

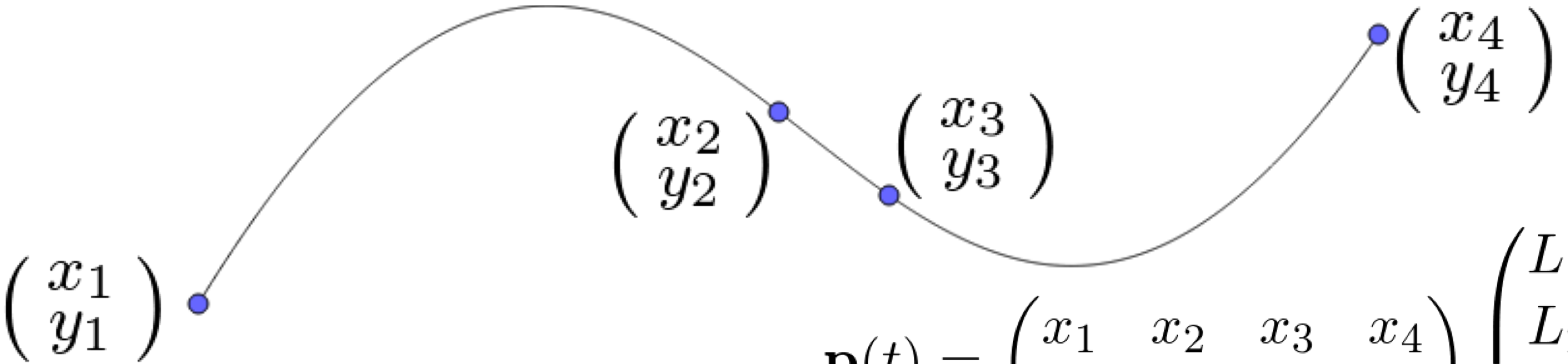


Remark

- Basis change is just a matrix multiplication
- We need to find a good basis for intuitively designing curves

Interpolation in 2D

- We now have pairs $\begin{pmatrix} x \\ y \end{pmatrix}$ we want to interpolate:



$$\mathbf{p}(t) = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \end{pmatrix} \begin{pmatrix} L_1(t) \\ L_2(t) \\ L_3(t) \\ L_4(t) \end{pmatrix}$$

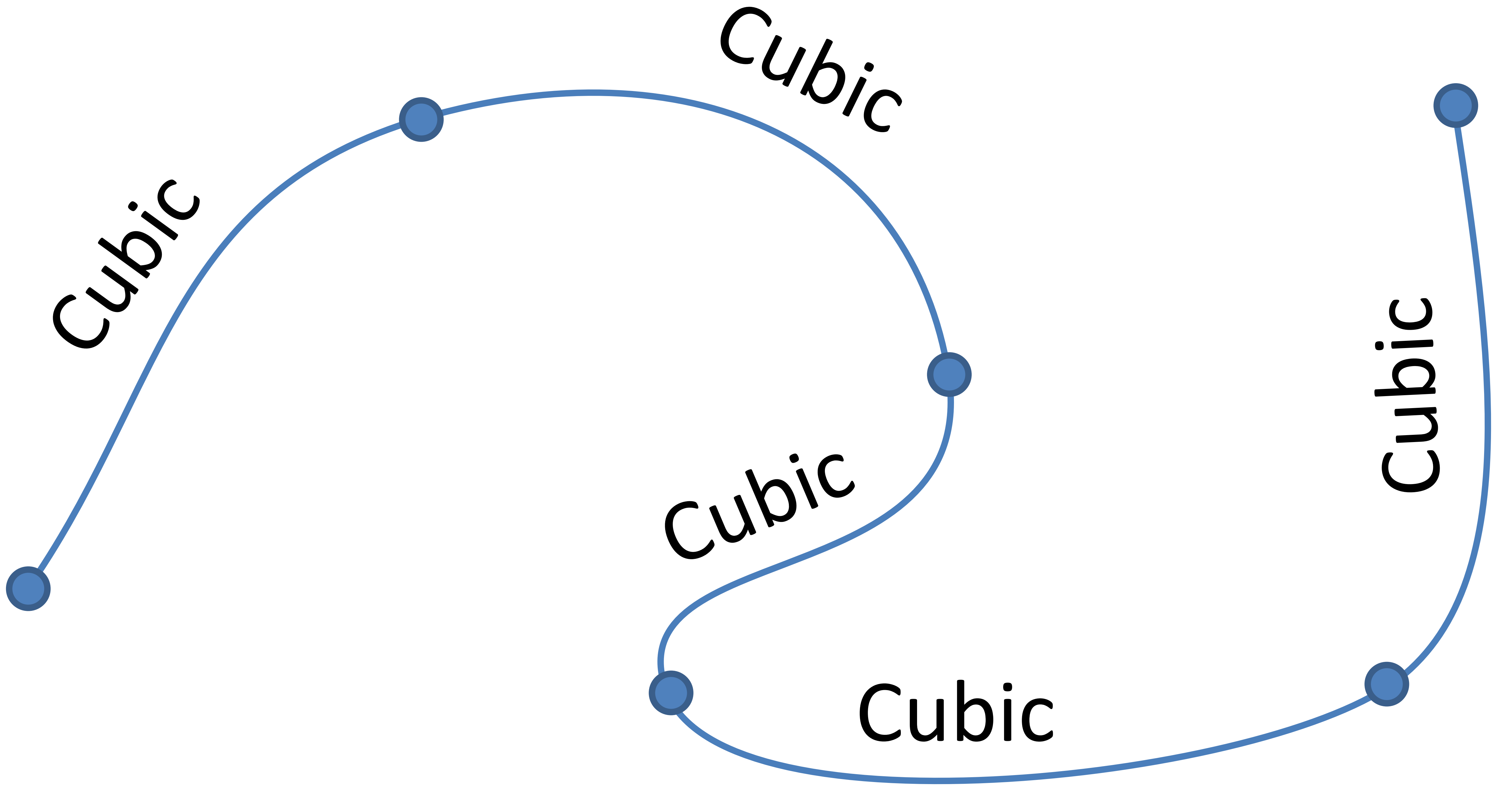
- How to choose t 's at which to specify L_i?

- Parameter space matters!

Runge's phenomenon

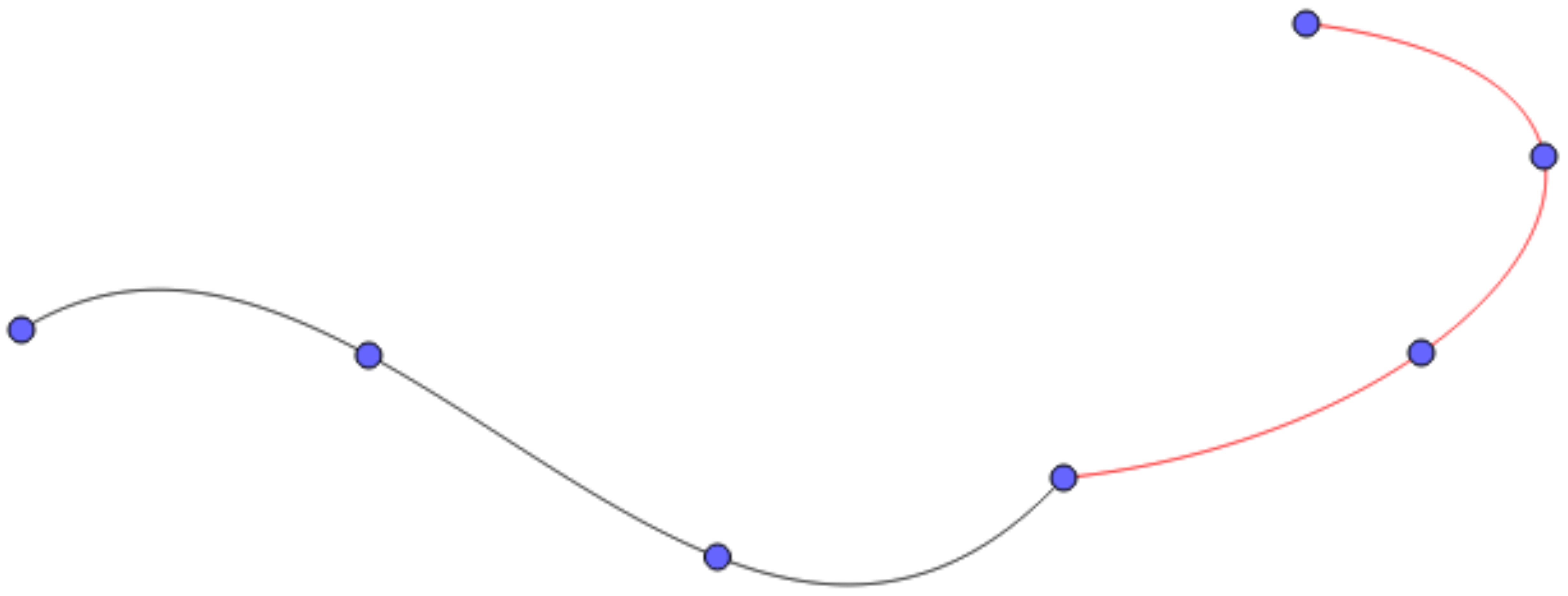
Splines

- Paste together low-degree polynomials



How do we get smoothness?

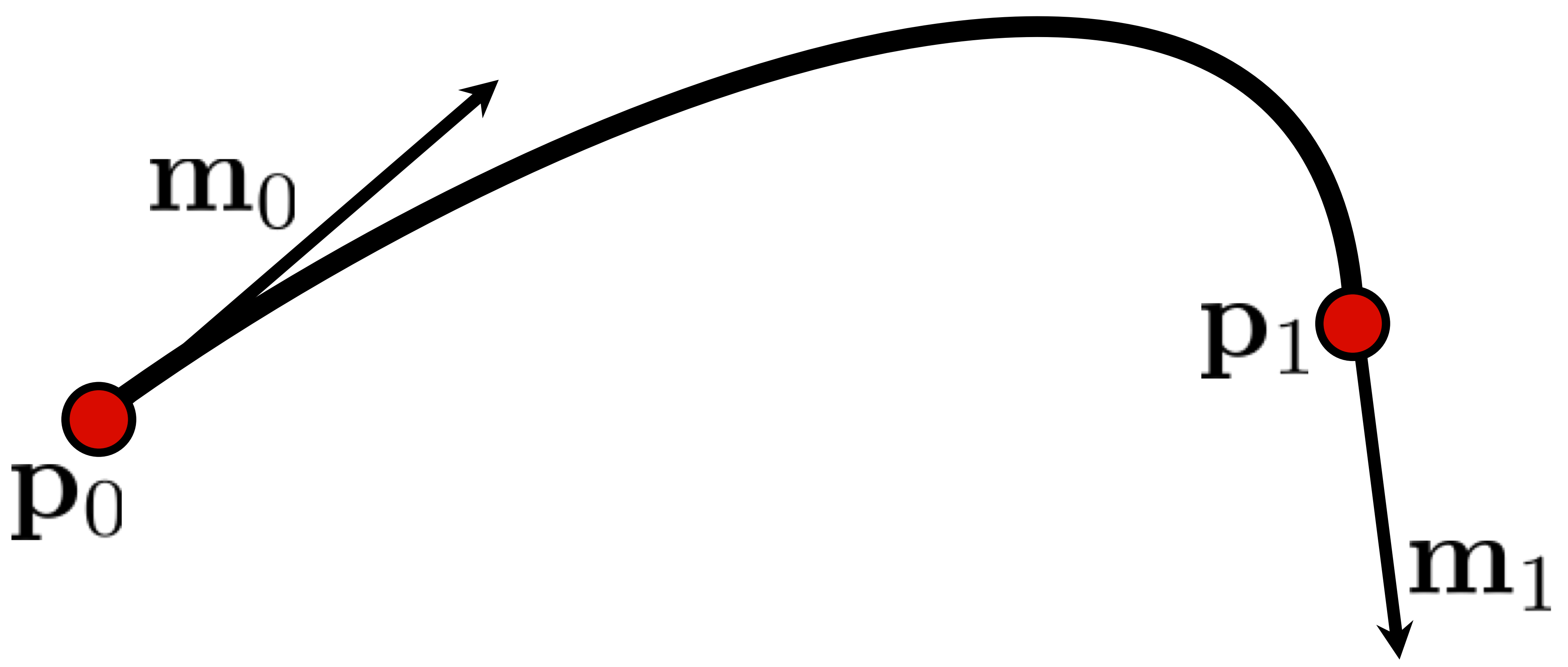
- With Lagrange polynomials, it's hard to get tangents to match up



Hermite Basis

- Instead of four points, specify two points and two derivatives:

$$\mathbf{p}(t_0) = \mathbf{p}_0$$
$$\mathbf{p}(t_1) = \mathbf{p}_1$$
$$\mathbf{p}'(t_0) = \mathbf{m}_0$$
$$\mathbf{p}'(t_1) = \mathbf{m}_1$$



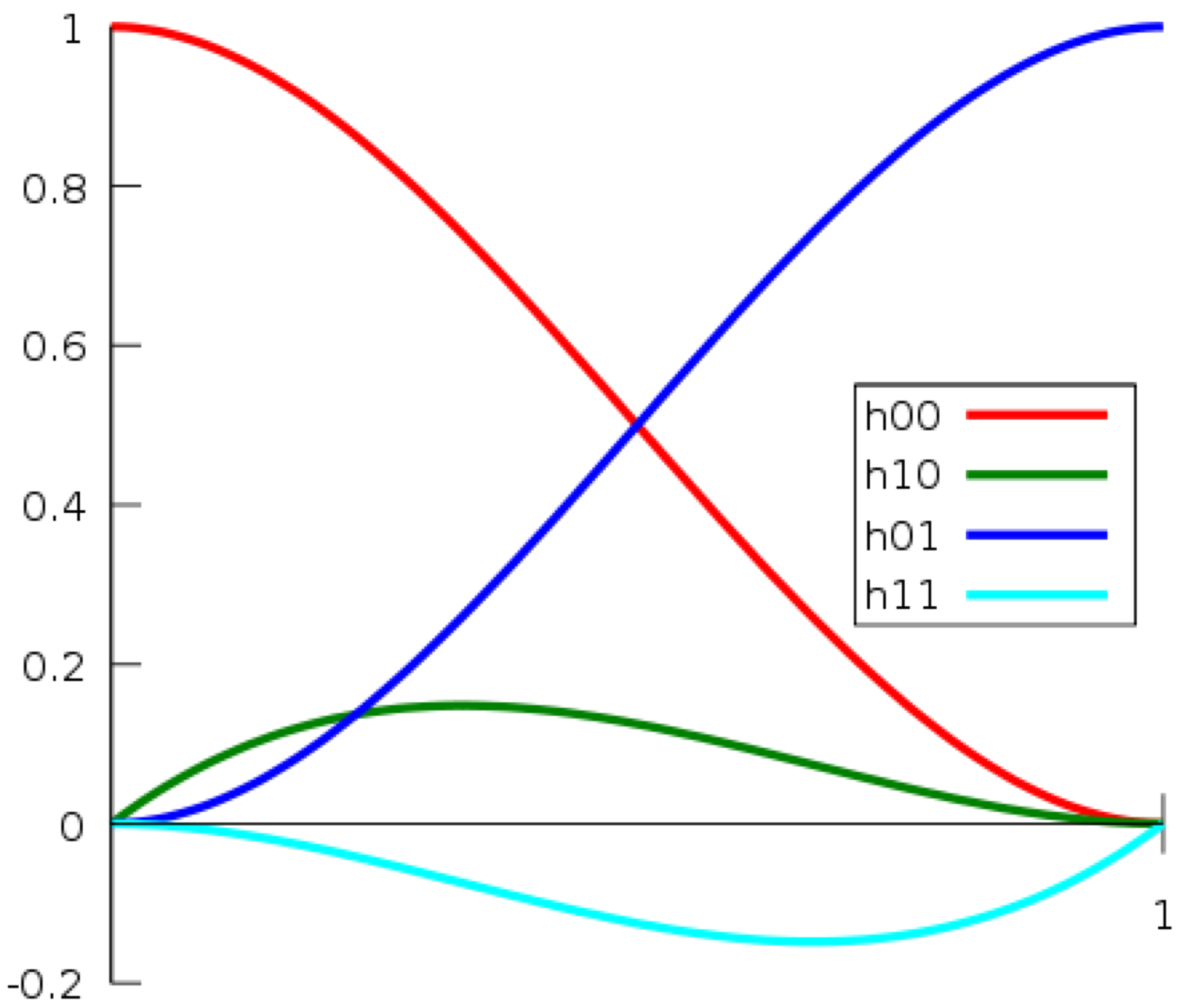
$$\mathbf{p}(t) = \mathbf{p}_0 h_{00}(t) + \mathbf{p}_1 h_{10}(t) + \mathbf{m}_0 h_{01}(t) + \mathbf{m}_1 h_{11}(t)$$

Cubic polynomials of the form $H(t) = h_0 + h_1 t + h_2 t^2 + h_3 t^3$

Hermite Basis

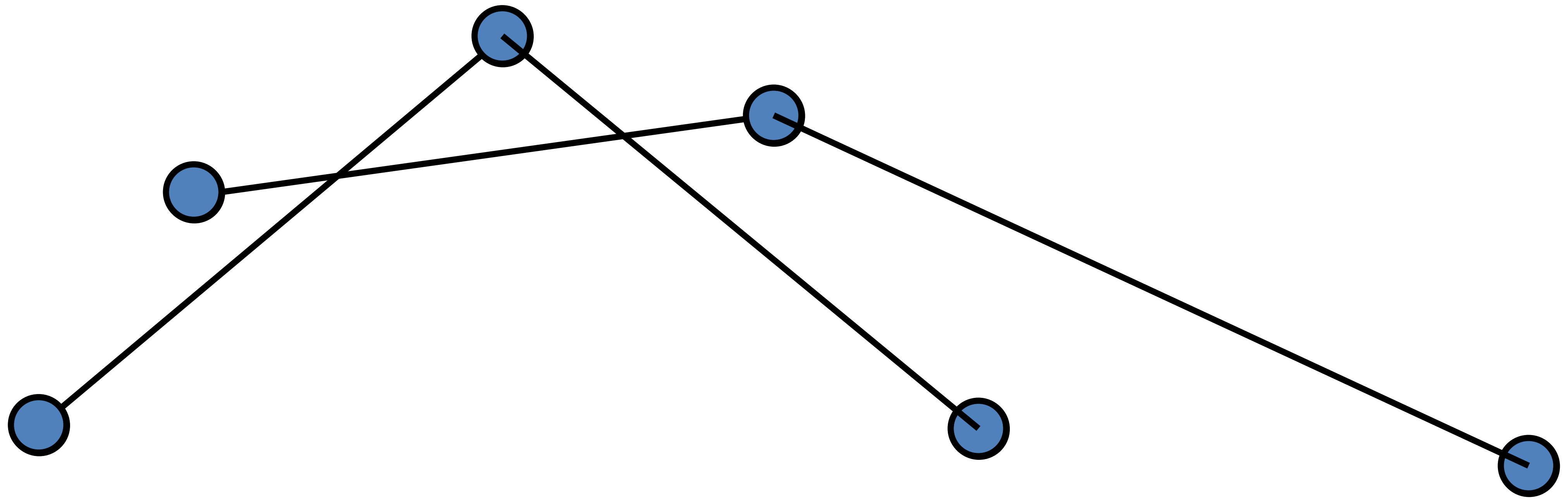
- Instead of four points, specify two points and two derivatives:

$$\begin{pmatrix} 1-3t^2+2t^3 \\ 3t^2-2t^3 \\ t-2t^2+t^3 \\ -t^2+t^3 \end{pmatrix}^T$$



Catmull-Rom splines

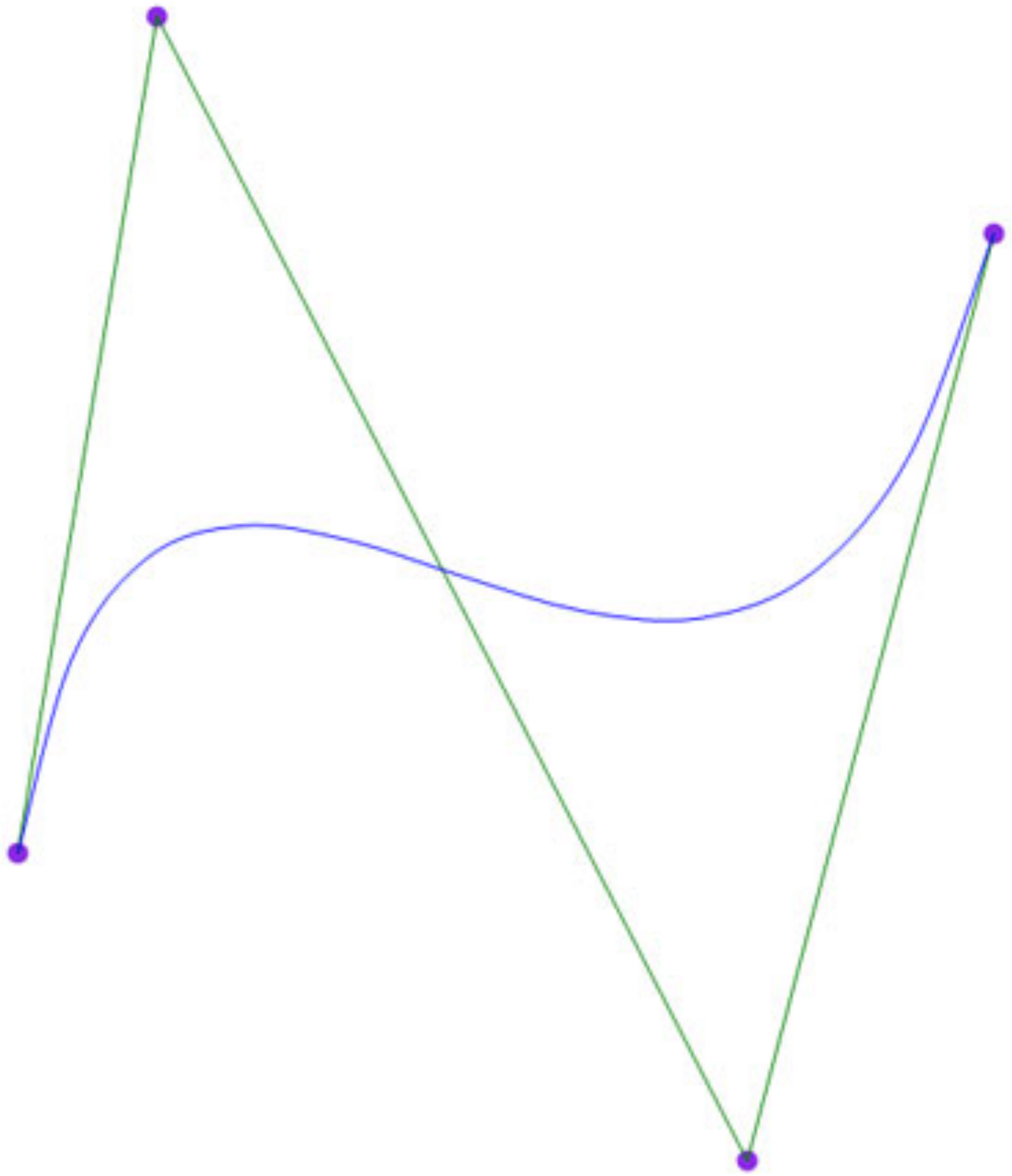
- If no tangents are explicitly specified – get them from the input points!



Tangent at $\mathbf{p}_i$ is parallel to $(\mathbf{p}_{i+1} - \mathbf{p}_{i-1})$

Next time

- Approximating Curves
  - Bezier splines
  - B-splines



References

Fundamentals of Computer Graphics, Fourth Edition
4th Edition **by Steve Marschner, Peter Shirley**

Chapter 15