Curves and curved surfaces

GEO1004:
3D modelling of the built environment

https://3d.bk.tudelft.nl/courses/geo1004

3D geoinformation
Department of Urbanism
Faculty of Architecture and the Built Environment
Delft University of Technology

Midterm exam

- 1 hour, 6 open questions -> ~10 minutes per question
- Lessons 1.1 (intro) - 4.1 (mat / gmaps)
- Open book, open laptop
- No LLMs (apart from AI snippets in searches)
- No communication with others, no phone

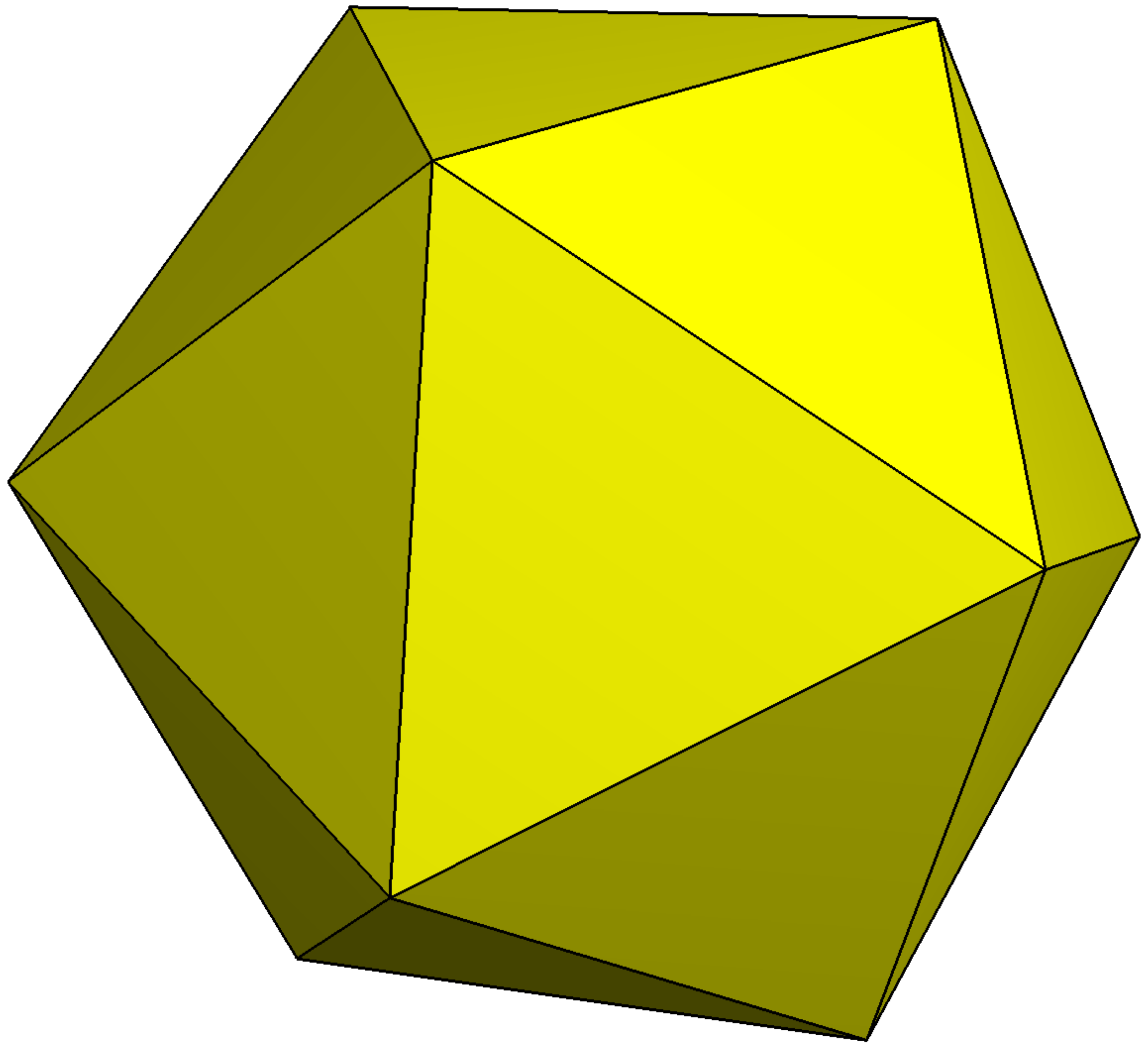
Homework 1 feedback

- Overall: very well done, all code worked, most reports were very good
- Some standouts:
  - Well accomplished pseudocode or higher level description with insight into method
  - Identification of usual and exceptional cases
  - Analysis using metrics for quality of results, simplification degree and performance

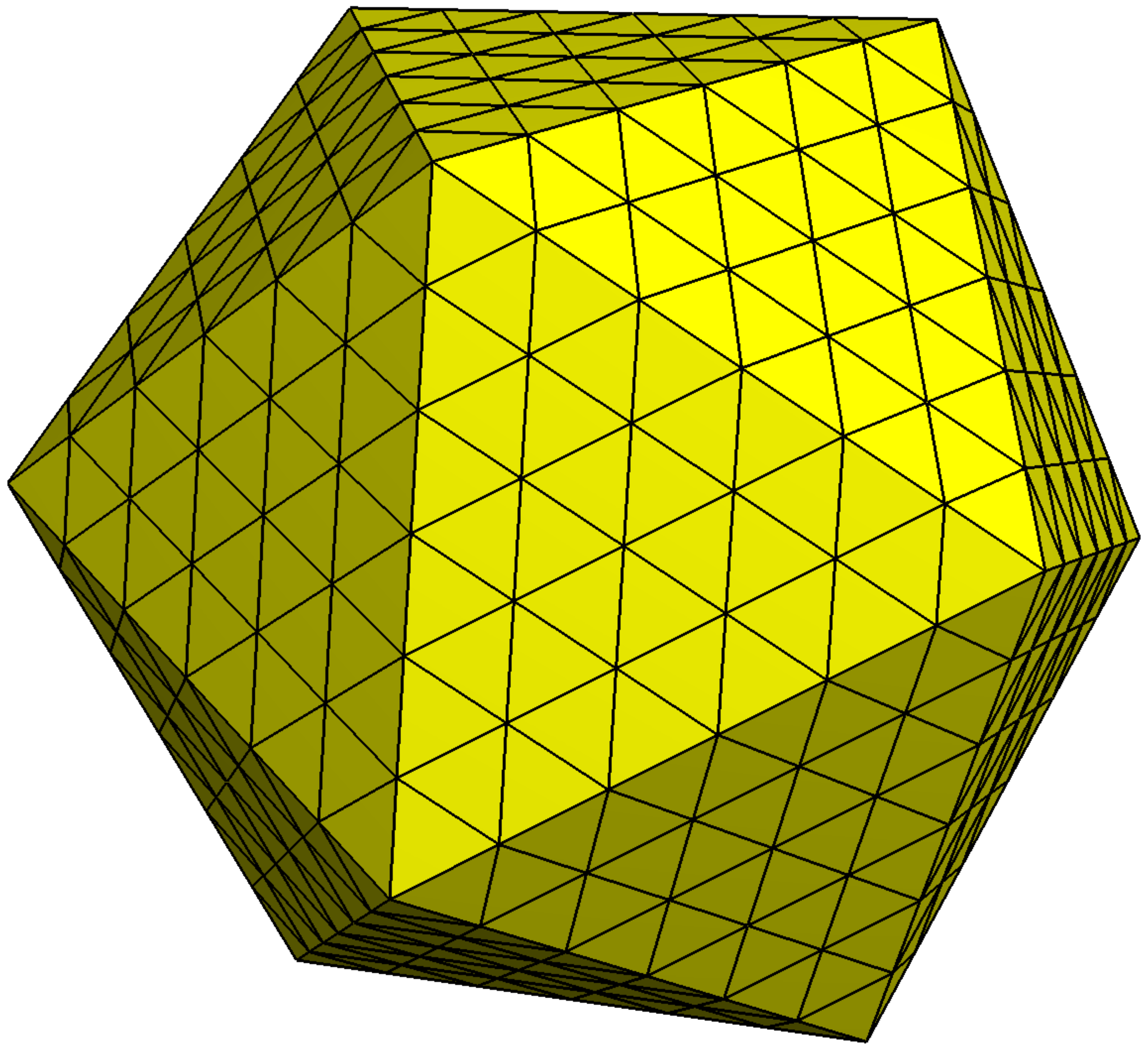
Oral authenticity check

- I’ll send you an email to agree on a time
  - 3: ??? & ???
  - 12: Marcel-Purcel & ???
  - 19: stinger

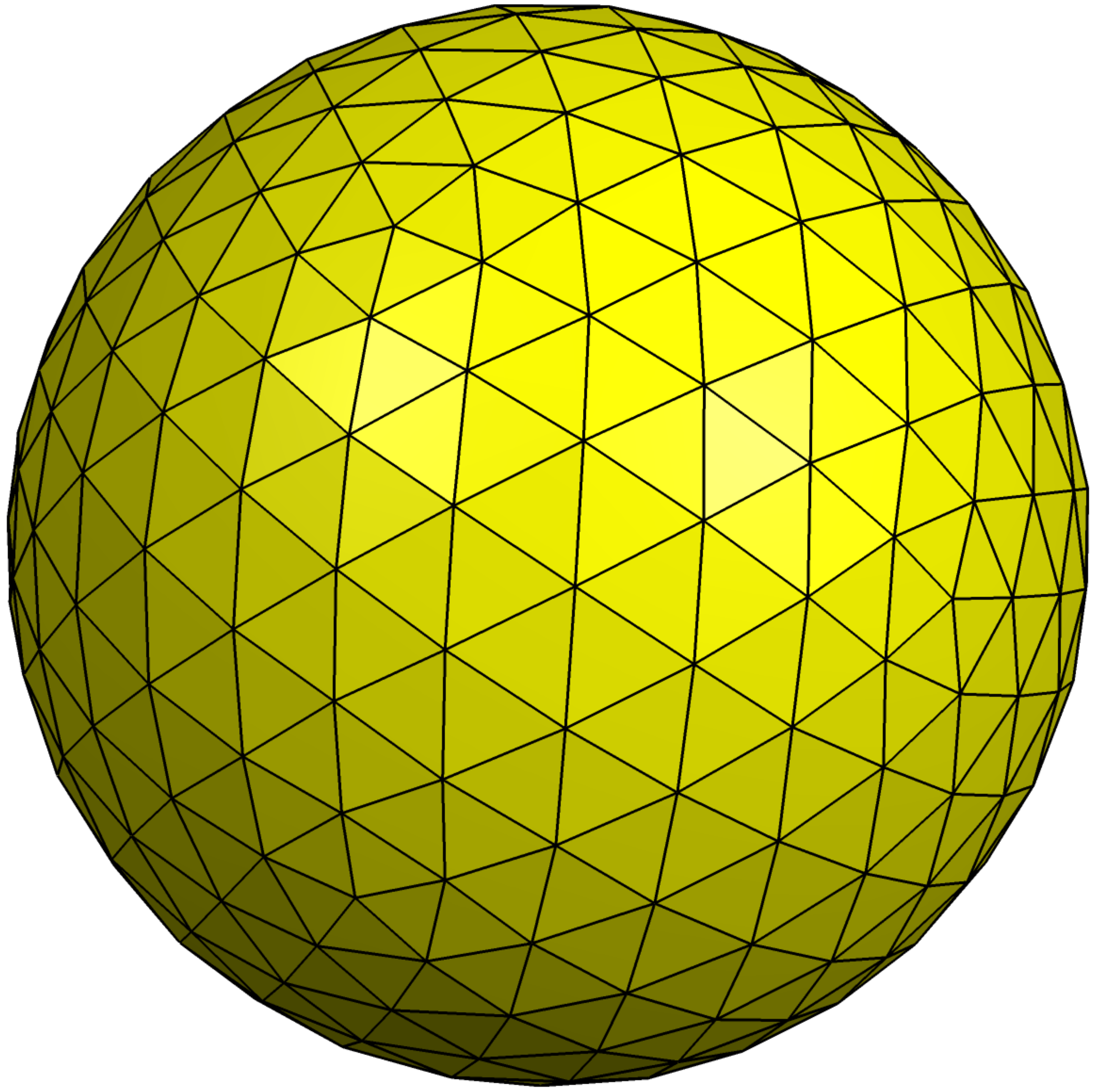
Representing a sphere using triangles



Icosahedron



6-frequency subdivision

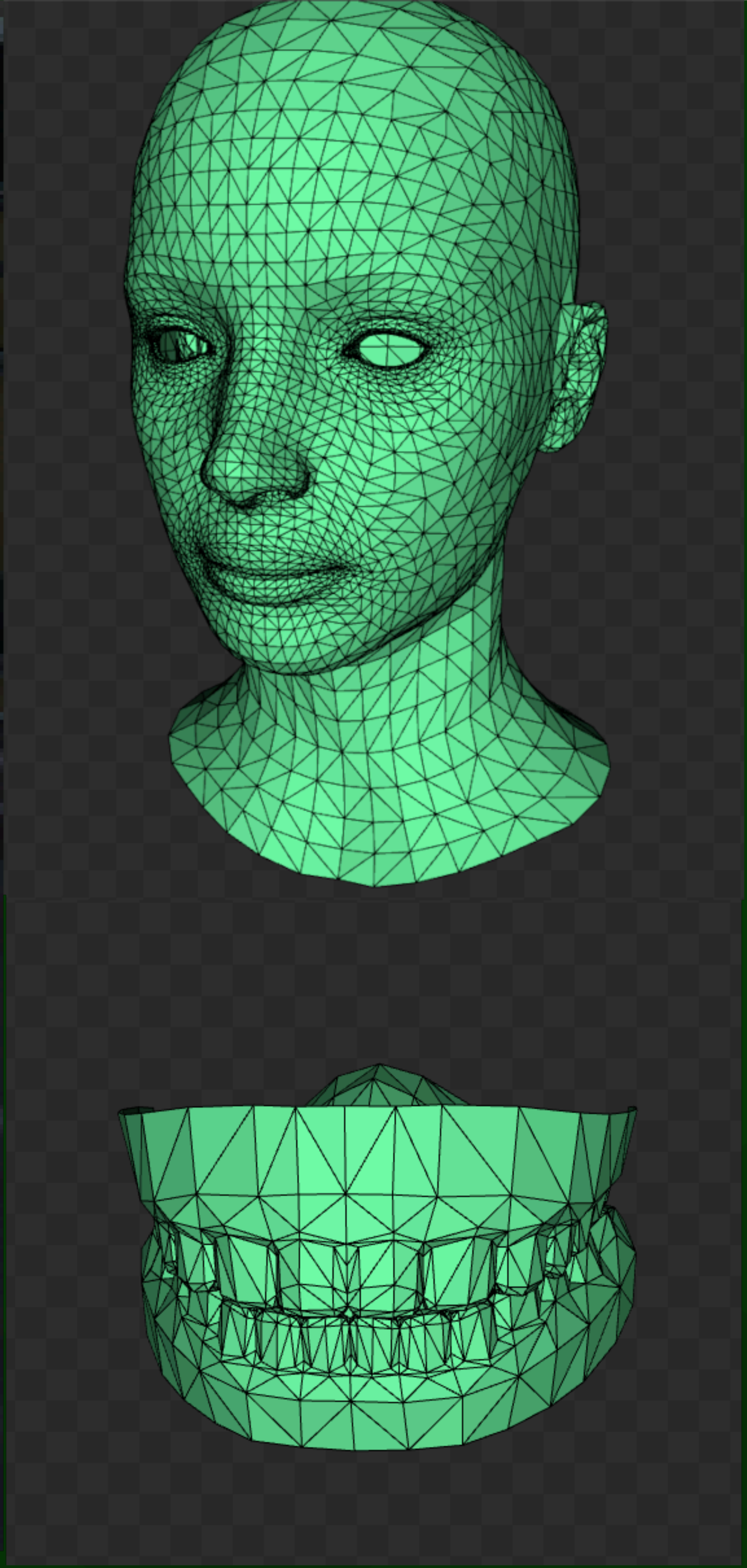


Vertices projected onto sphere

Representing a sphere using triangles

Subdivision level	Triangles	Edge length	Deviation
0	20	1.051	0.255
1	80	0.546	0.066
2	320	0.280	0.017
3	1 280	0.142	0.004
4	5 120	0.072	0.001

where 1.0 is the radius of the sphere



https://www.reddit.com/r/CitiesSkylines/comments/17gfq13/the_game_does_render_individual_teeth_with_no_lod/
https://www.youtube.com/watch?v=-BNDwgMhWEg

Why curves?

- Better approximation to reality with fewer primitives
- Discretisation creates problems (precision, inconsistencies, errors in software, etc.)
- Easier to manipulate -> used in CAD, BIM and other modelling software

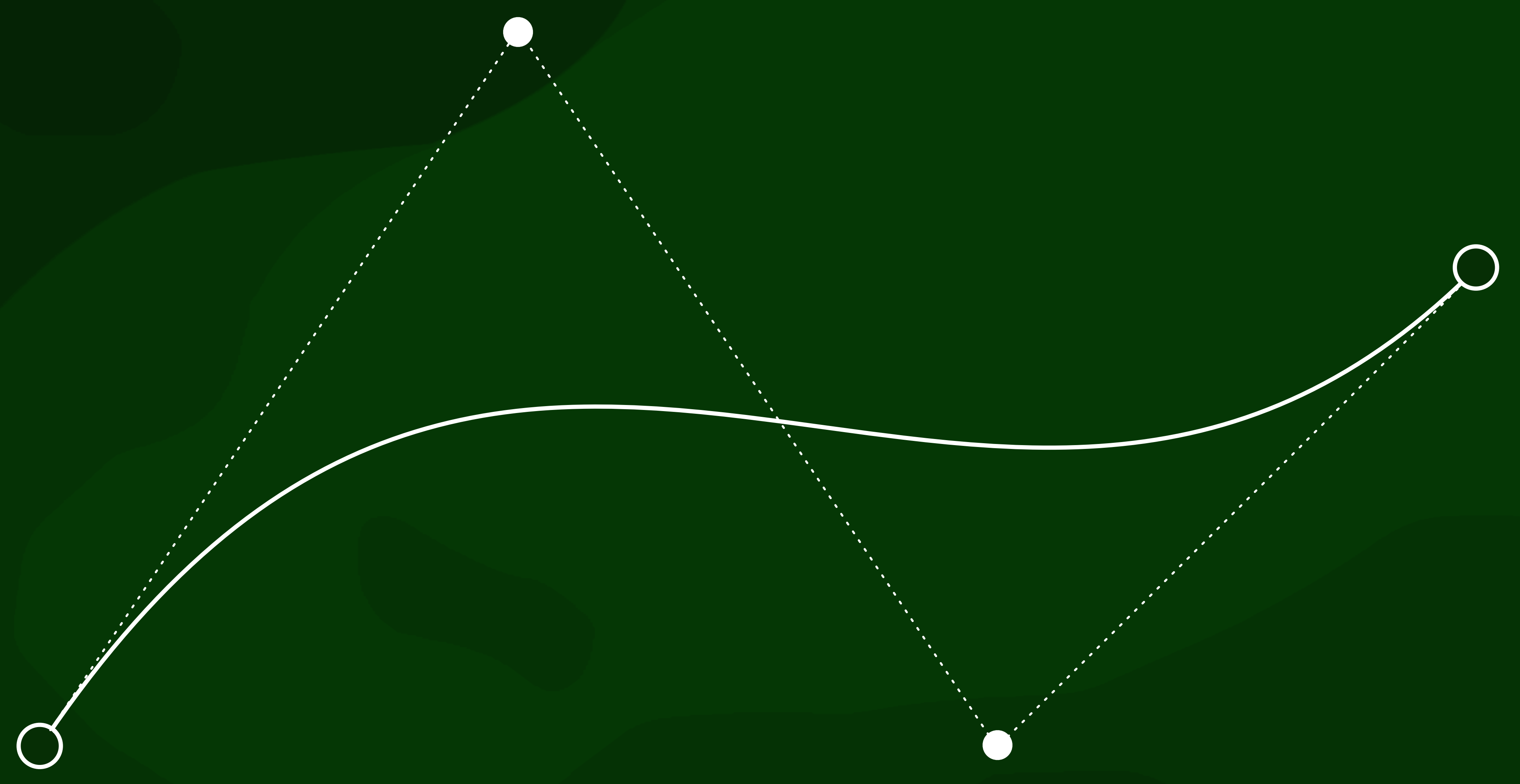
Modelling of curves

- Parametric solids (eg spheres, cylinders and cones in CSG) -> Wednesday
- Bézier curves and surfaces -> today
- Basis splines (uniform, rational, NURBS, etc.) -> today
- Sweeps -> BIM lecture

Types of points

- Data points: points that the curve/surface needs to pass through

- Control points: points that have some influence over the shape of the curve/surface, but through which the curve/surface does not necessarily pass. Intuitively, they `pull' the curve in their direction.



- Note: sometimes both data points and control points are simply called control points

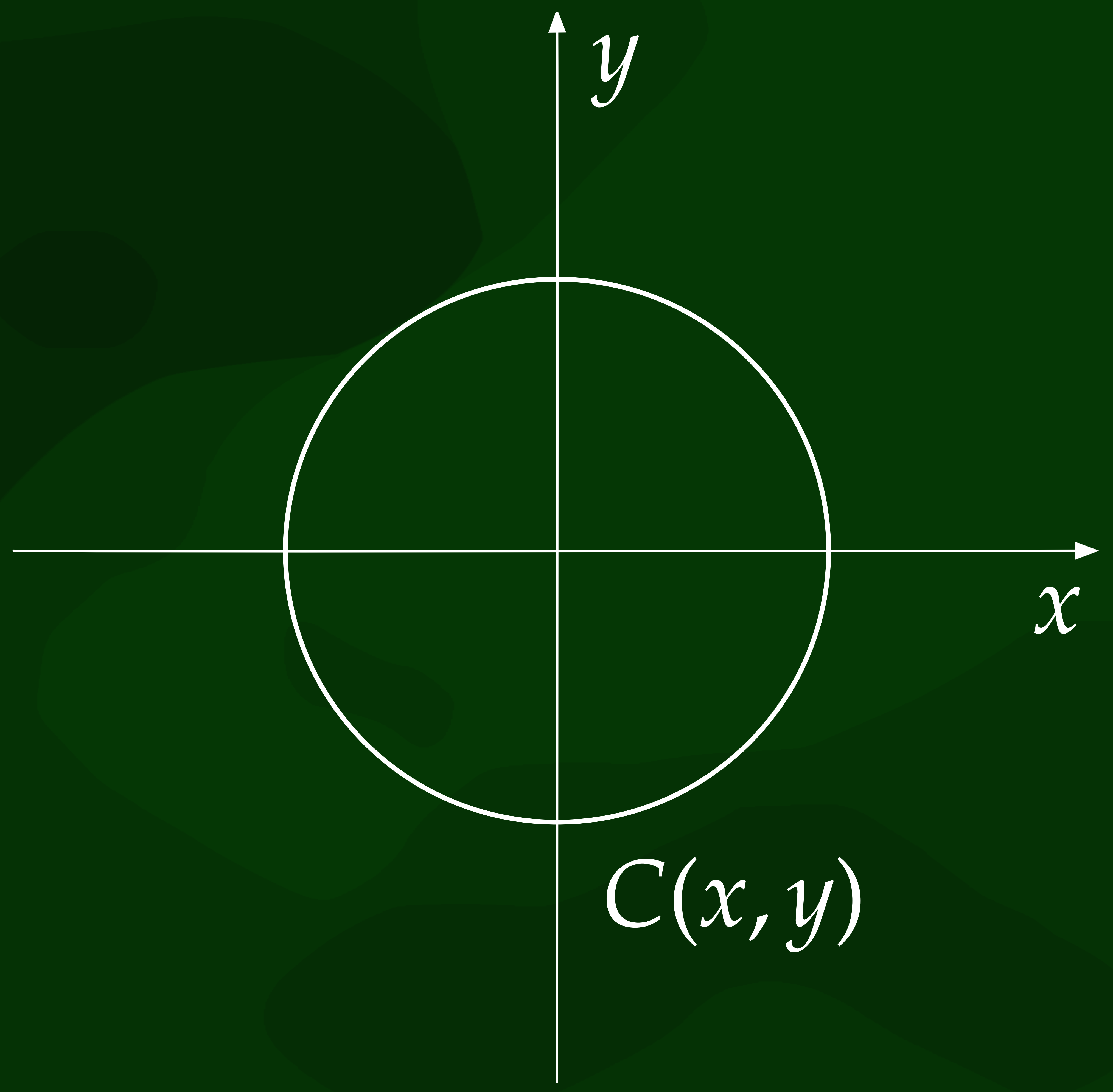
Curves using parametric equations

- Most methods to store curves and curved surfaces use parametric equations, ie equations that are defined using non-coordinate variables as parameters.
- For example, we can define a unit circle as:

$$C(x, y) = (\cos t, \sin t)$$

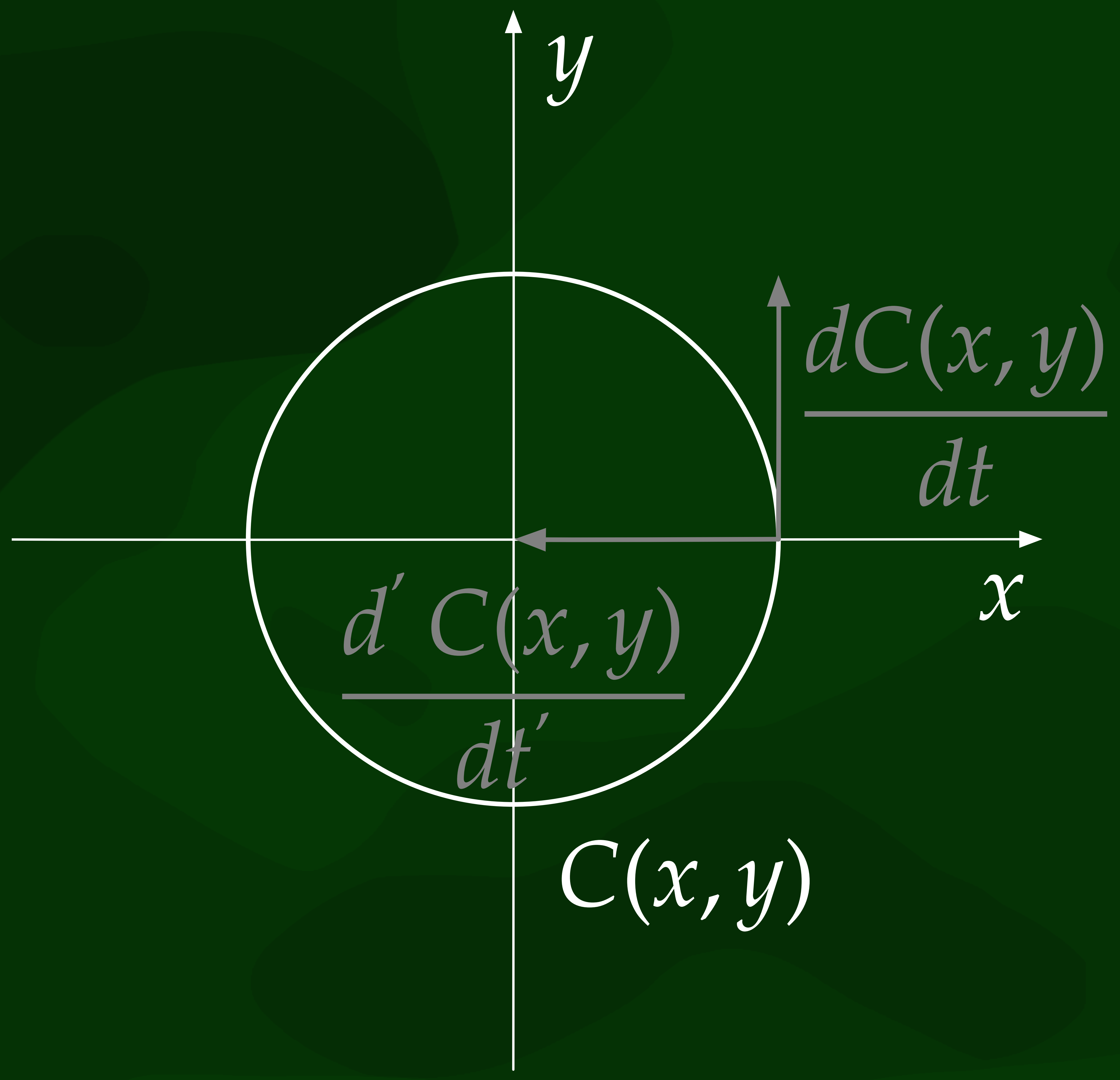
- where:

$$0 \leq t \leq 2\pi$$



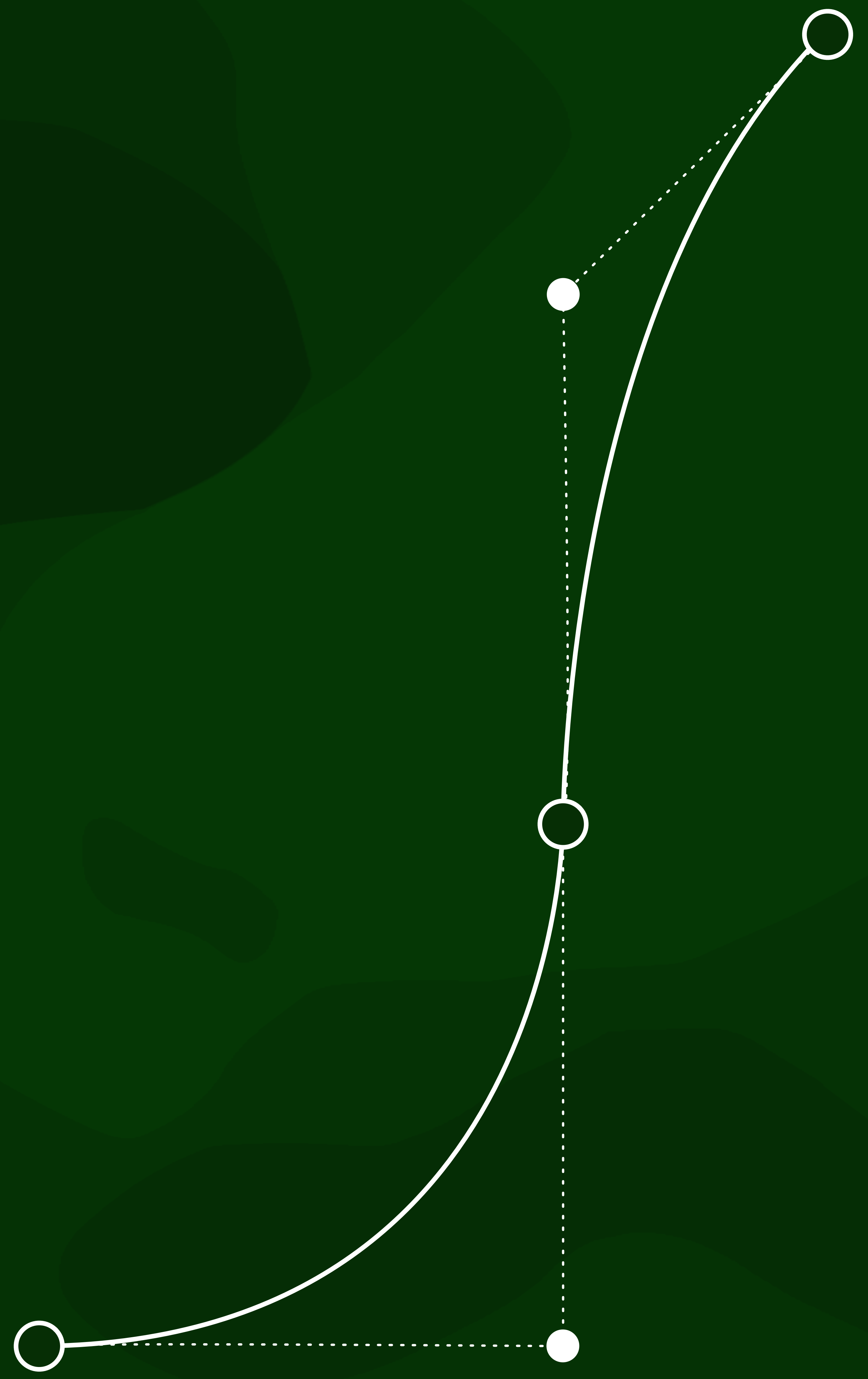
Derivatives of parametric equations

- If we look at the derivatives of these equations,
  - the first derivative is a tangent vector
  - the second derivative tells us the rate of change of the curvature

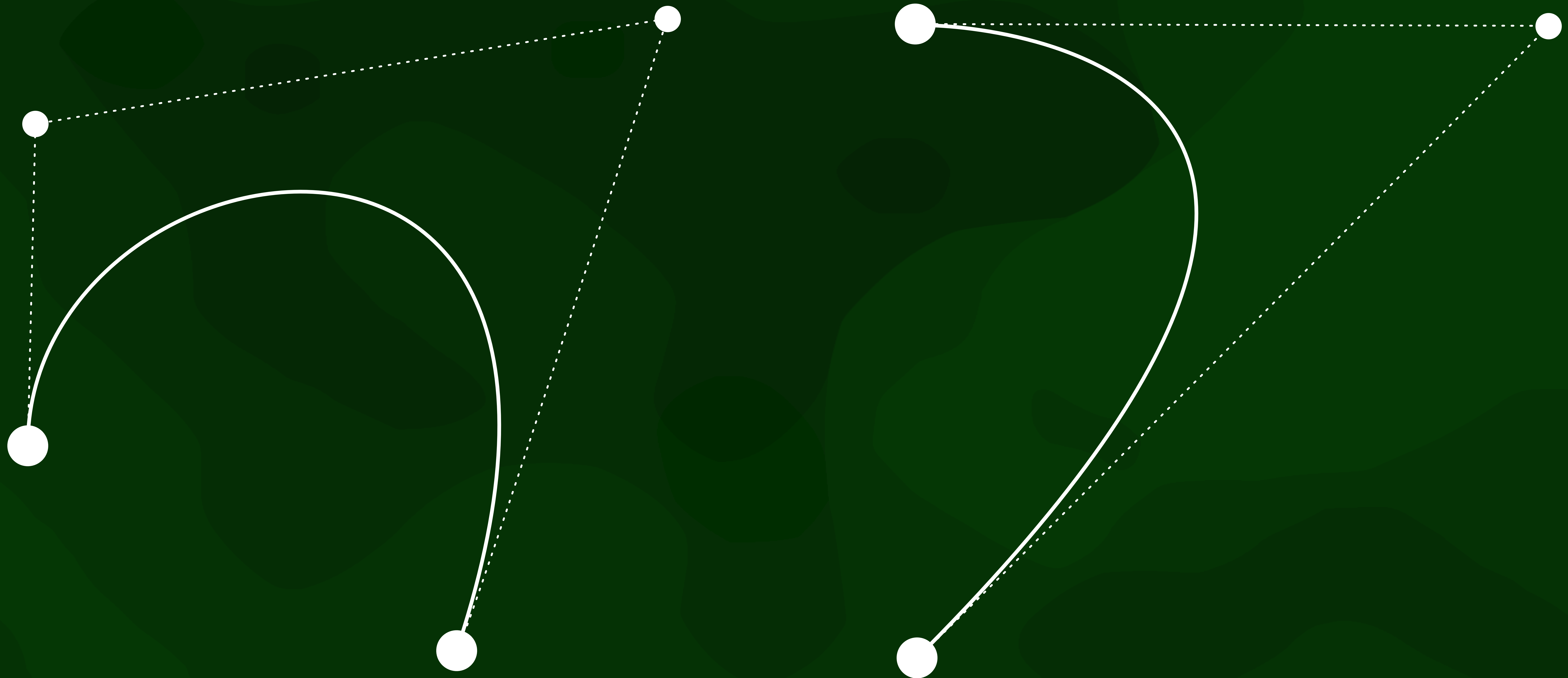


Derivatives and continuity

- Positional (G^0) continuity: the boundary of a curve (ie a point) or surface (ie a curve) matches that of its neighbours, ie there are no gaps at common boundaries;
- Tangential (G^1) continuity: the angles of curves or surfaces match those of its neighbours at their common boundaries, ie no sharp edges (creases / folds) at common boundaries;
- Curvature (G^2) continuity: the curvature of a curve or surface matches that of its neighbours at their common boundaries, ie not even 'soft' or rounded edges at common boundaries.
- A curve has G^n continuity at a boundary point when the n -th derivatives have the same direction at that point. If they have the same magnitude as well, it has C^n continuity too.



Bézier curves



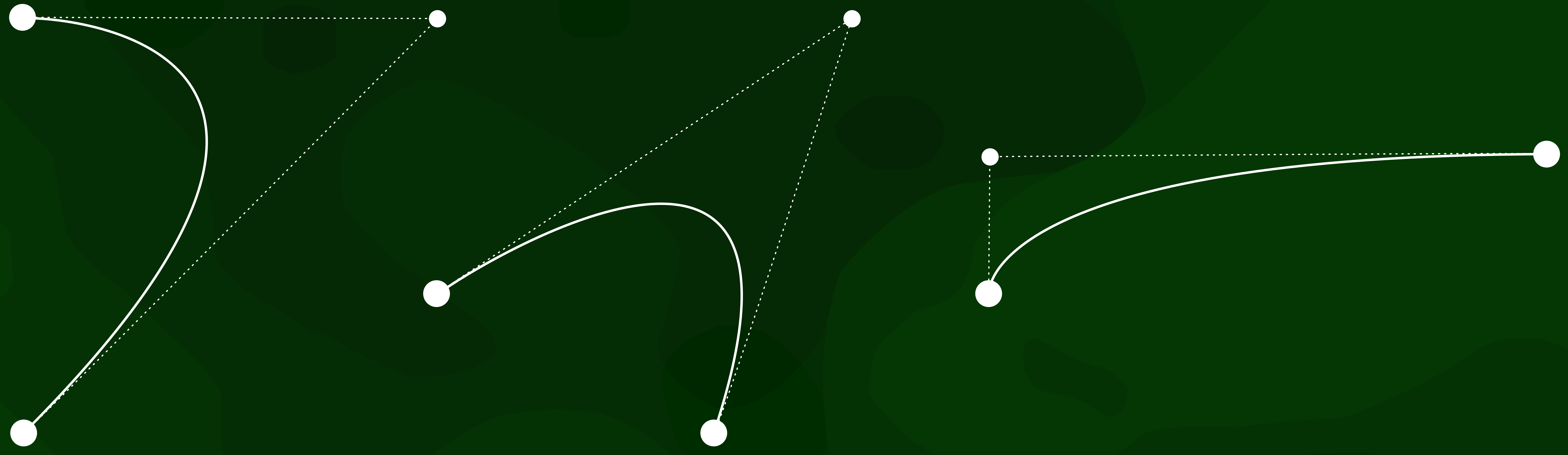
Bézier curves

- Parametric curves that are based on a polynomial function with one parameter

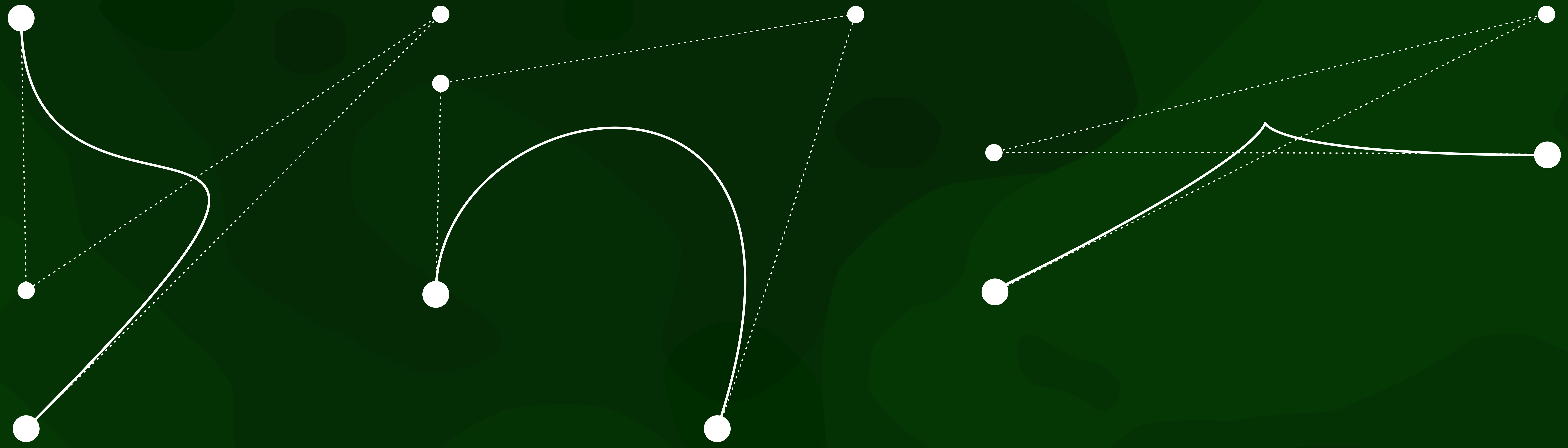
- Intuitively, a weighted average of its points $(p0, p1, \ldots, p_n)$:

$$C(t) = \sum_{i=0}^{n} B_i^n(t) p_i, \text{ for } 0 \le t \le 1$$

Quadratic Bézier curves



Cubic Bézier curves



Bézier curves

$$C(t) = \sum_{i=0}^{n} B_i^n(t) p_i, \text{ for } 0 \leq t \leq 1$$

where:

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

where:

$$\binom{n}{i} = \frac{n!}{i!\,(n-i)!}$$

or

$$\begin{matrix} 1 \\ 1 \quad 1 \\ 1 \quad 2 \quad 1 \\ 1 \quad 3 \quad 3 \quad 1 \\ 1 \quad 4 \quad 6 \quad 4 \quad 1 \\ 1 \quad 5 \quad 10 \quad 10 \quad 5 \quad 1 \end{matrix}$$

Bézier curves

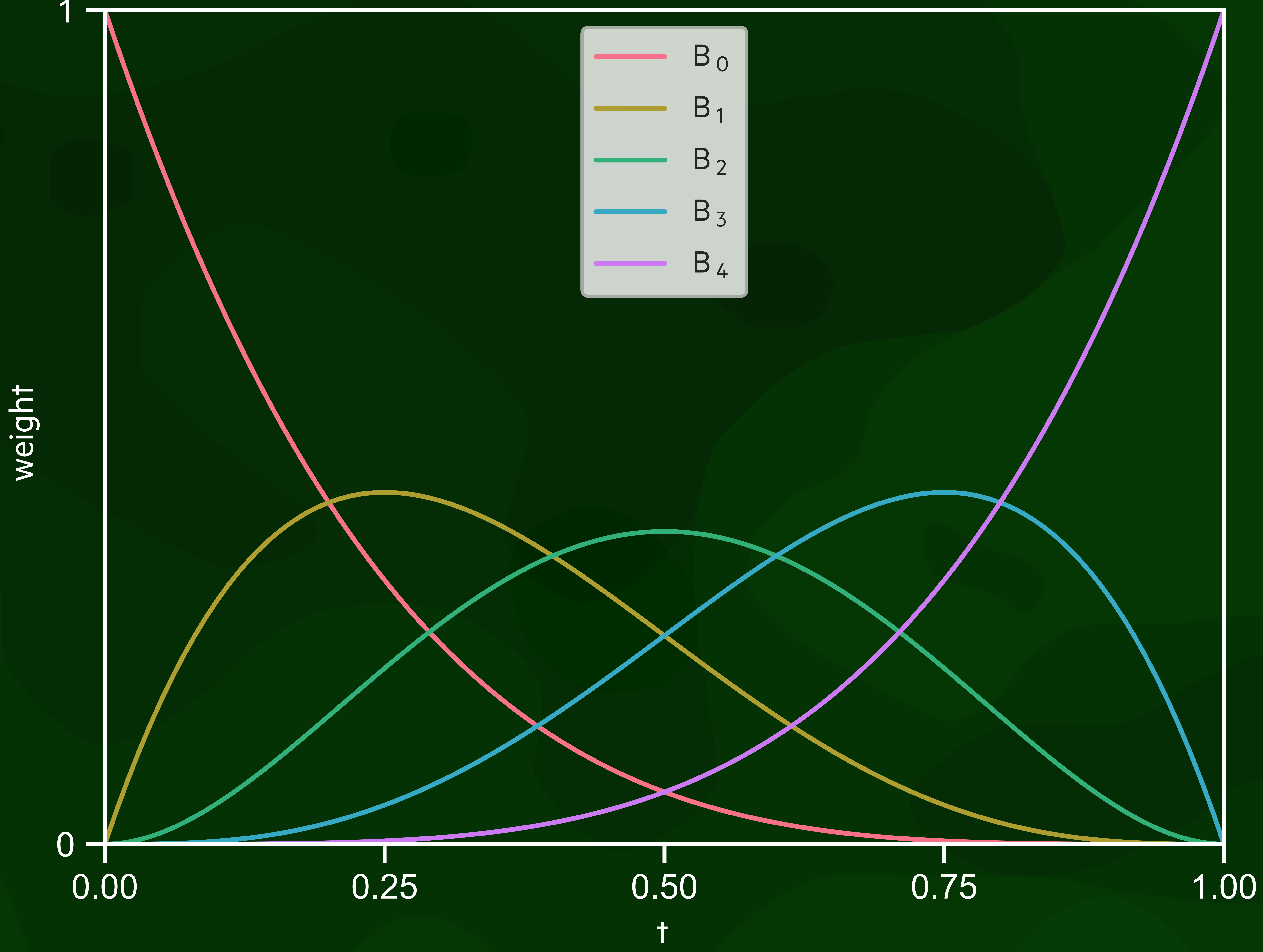
- Quadratic:

$$
\begin{aligned}
C(t) &= \sum_{i=0}^{2} B_i^2(t) p_i \\
&= (1-t)^2 p_0 + 2t(1-t) p_1 + t^2 p_2, \text{ for } 0 \le t \le 1.
\end{aligned}
$$

- Cubic:

$$
\begin{aligned}
C(t) &= \sum_{i=0}^{3} B_i^3(t) p_i \\
&= (1-t)^3 p_0 + 3t(1-t)^2 p_1 + 3t^2(1-t) p_2 + t^3 p_3, \text{ for } 0 \le t \le 1.
\end{aligned}
$$

Bernstein polynomials



Composite Bézier curves

- We could use high-degree polynomials to model very complex curves, but in a Bézier curve **all points** affect **all parts** of the curve
- Better to model complex curves as composites:

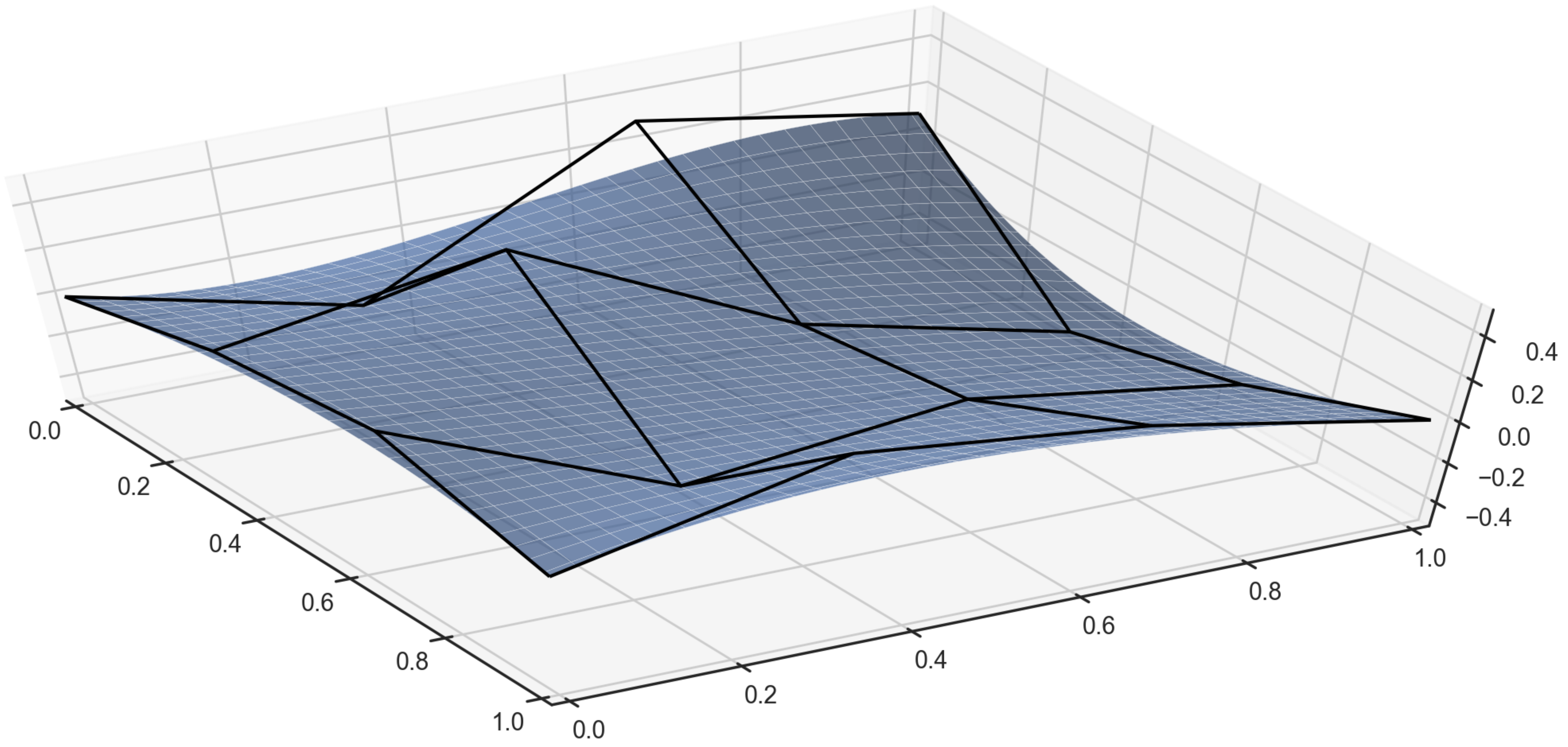
Bézier rectangles

$$p_{\text{biquadratic}} = \begin{pmatrix} p_{0,0} & p_{0,1} & p_{0,2} \\ p_{1,0} & p_{1,1} & p_{1,2} \\ p_{2,0} & p_{2,1} & p_{2,2} \end{pmatrix}, \text{ and}$$

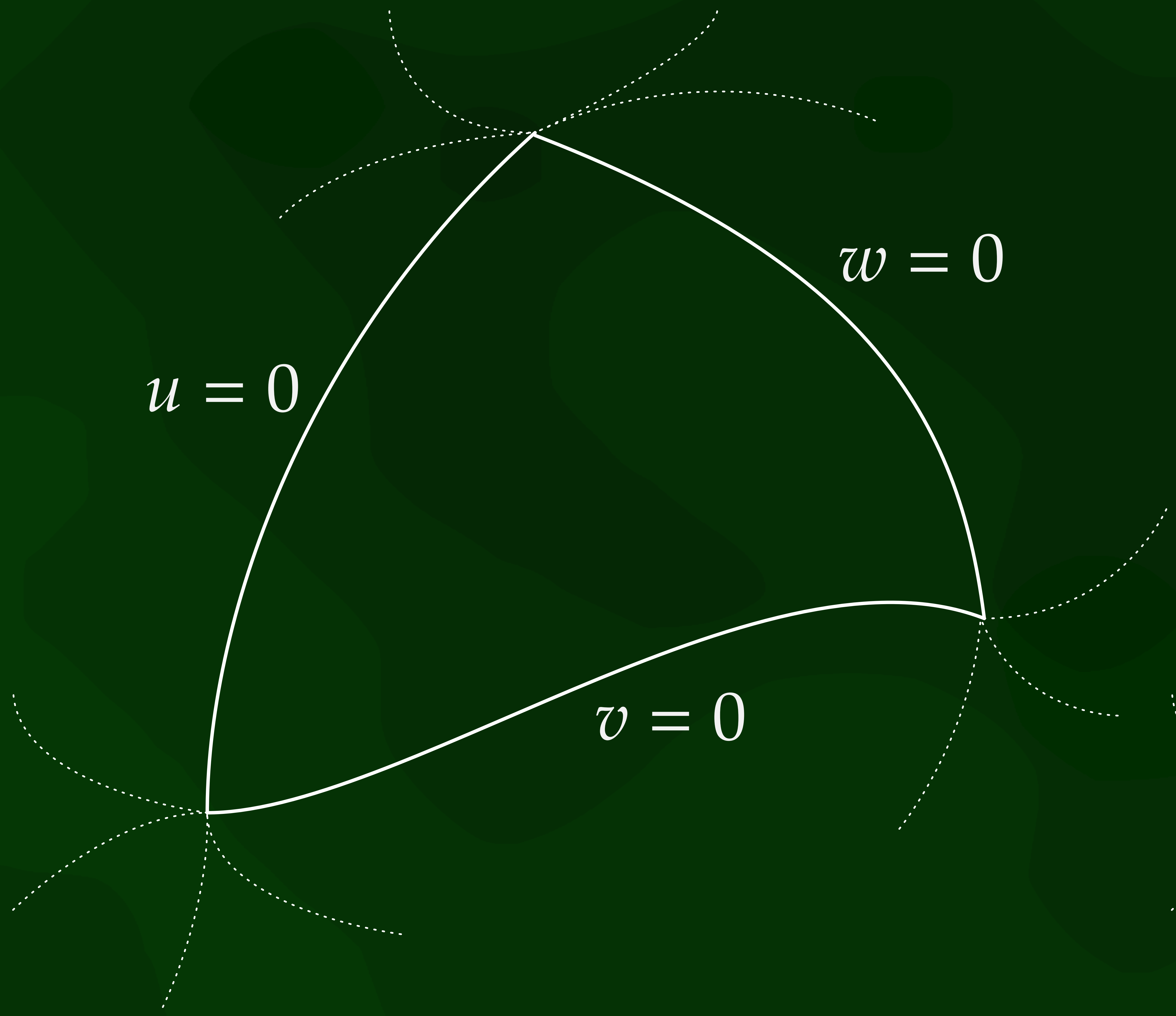
$$p_{\text{bicubic}} = \begin{pmatrix} p_{0,0} & p_{0,1} & p_{0,2} & p_{0,3} \\ p_{1,0} & p_{1,1} & p_{1,2} & p_{1,3} \\ p_{2,0} & p_{2,1} & p_{2,2} & p_{2,3} \\ p_{3,0} & p_{3,1} & p_{3,2} & p_{3,3} \end{pmatrix},$$

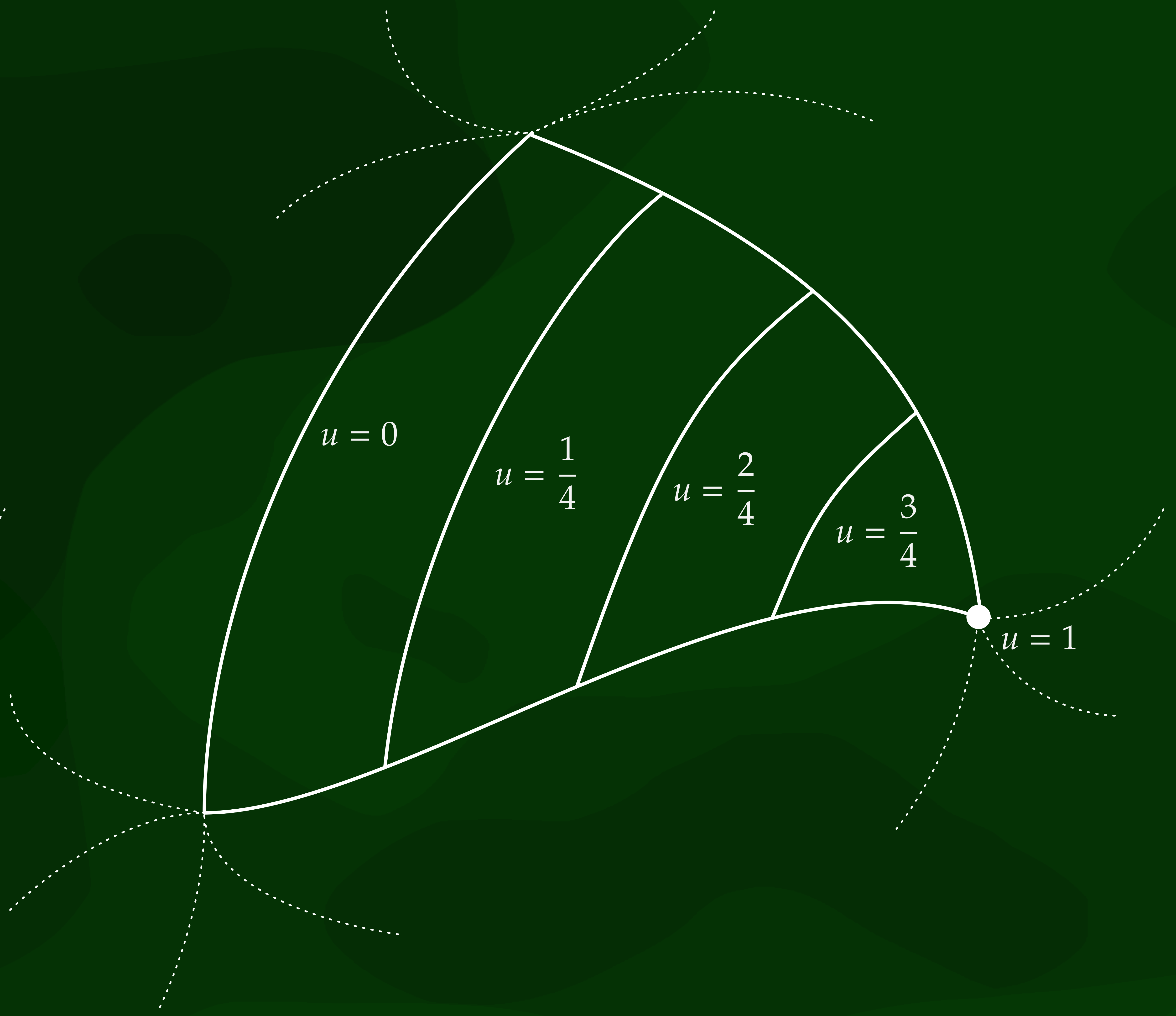
$$S(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} B_i^n(u) B_j^m(v) p_{i,j}, \text{ for } 0 \le u \le 1, 0 \le v \le 1.$$

Bézier rectangles



Bézier triangles





Bézier triangles

$$p_{\text{linear}} = \begin{matrix} & p_{0,1,0} & \\ p_{0,0,1} & & p_{1,0,0} \end{matrix},$$

$$p_{\text{quadratic}} = \begin{matrix} & & p_{0,2,0} & & \\ & p_{0,1,1} & & p_{1,1,0} & \\ p_{0,0,2} & & p_{1,0,1} & & p_{2,0,0} \end{matrix},$$

$$p_{\text{cubic}} = \begin{matrix} & & & p_{0,3,0} & & & \\ & & p_{0,2,1} & & p_{1,2,0} & & \\ & p_{0,1,2} & & p_{1,1,1} & & p_{2,1,0} & \\ p_{0,0,3} & & p_{1,0,2} & & p_{2,0,1} & & p_{3,0,0} \end{matrix},$$

$$S(u, v, w) = \sum_{\substack{i+j+k=n \\ i,j,k \geq 0}} B^n_{i,j,k}(u, v, w) p_{i,j,k}$$

Bézier triangles

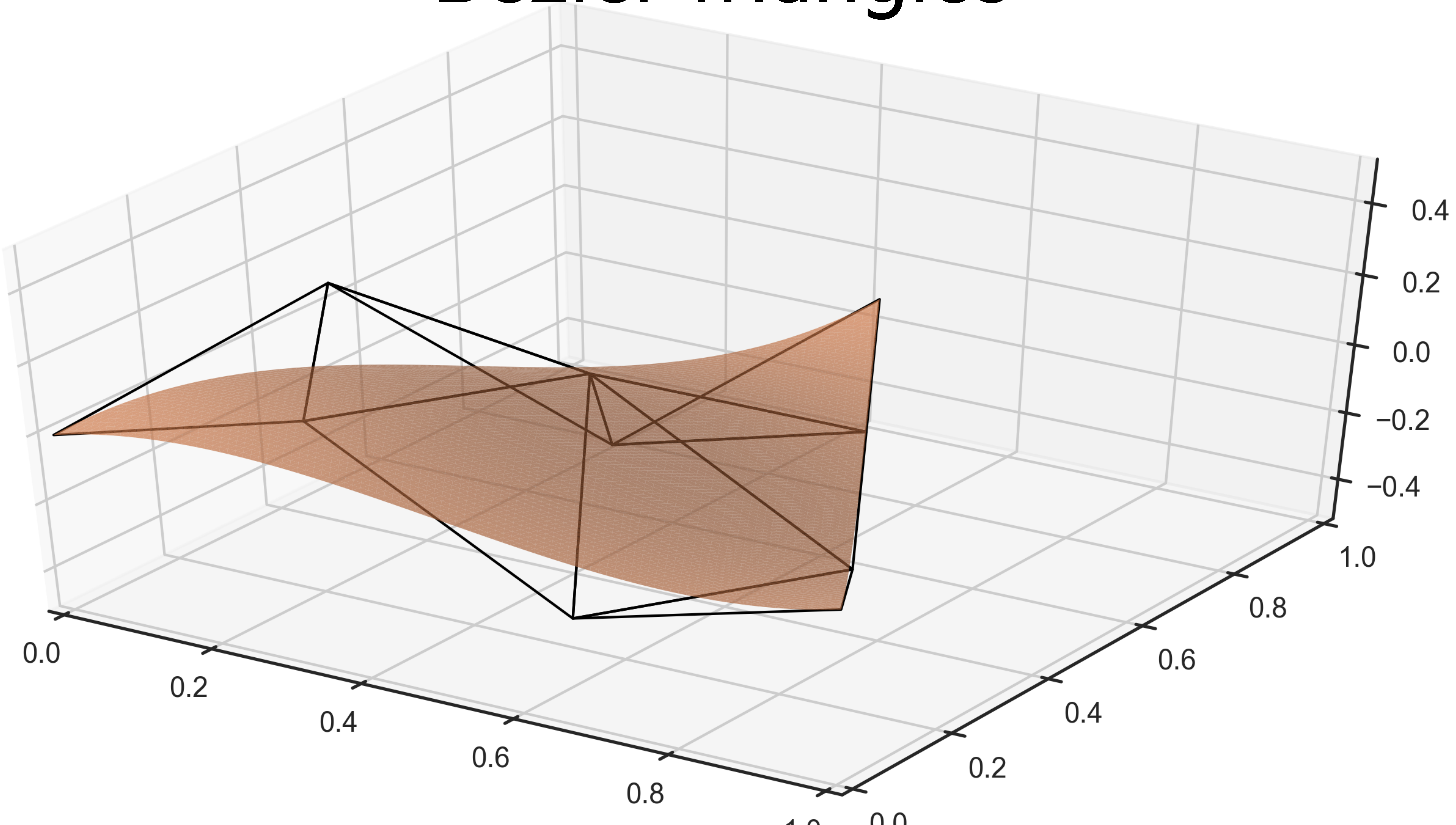
$$B^n_{i,j,k}(u,v,w) = \frac{n!}{i!j!k!}u^i v^j w^k$$

$$B^1_{i,j,k} = \begin{array}{ccc} & v & \\ w & & u \end{array},$$

$$B^2_{i,j,k} = \begin{array}{ccccc} & & v^2 & & \\ & 2vw & & 2uv & \\ w^2 & & 2uw & & u^2 \end{array},$$

$$B^3_{i,j,k} = \begin{array}{ccccccc} & & & v^3 & & & \\ & & 3v^2w & & 3uv^2 & & \\ & 3vw^2 & & 6uvw & & 3u^2v & \\ w^3 & & 3uw^2 & & 3u^2w & & u^3 \end{array}$$

Bézier triangles



Rational Bézier curves

- We extend Bézier curves with rational weights (w_i), where every point p_i is associated with a weight w_i
  - If $w_i = 1$, the point p_i has a normal weight and we get a typical Bézier curve
  - If $w_i = 0$, the point p_i has no weight, ie it doesn't pull the curve towards it
  - Larger or smaller values of w_i will affect the 'force' with which p_i pulls the point

$$C(t) = \frac{\sum_{i=0}^{n} B_i^n(t) w_i p_i}{\sum_{i=0}^{n} B_i^n(t) w_i}$$

B-splines

- Also, parametric curves that are based on a polynomial function with one parameter

- Also, intuitively, a weighted average of its points $(p_0, p_1, \ldots, p_n)$:

$$C(t) = \sum_{i=0}^{n} B_{i,k}(t) p_i$$

- But with the addition of a knot vector $(t_0, t_1, \ldots, t_m)$ that describes which control points affect which parts of the curve

- Knot vector -> only a few points affect each segment -> local control

B-spline basis functions

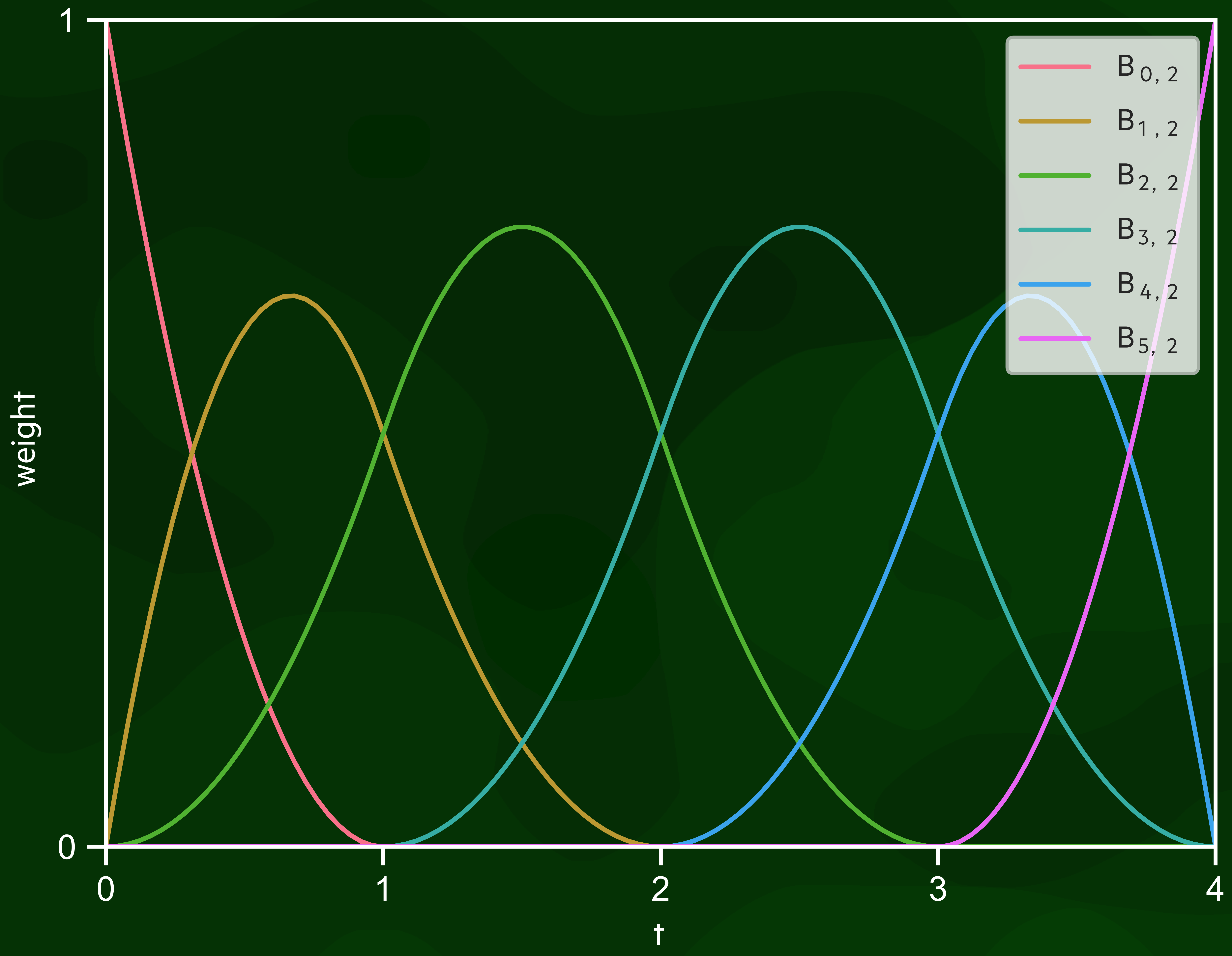
$$C(t) = \sum_{i=0}^{n} B_{i,k}(t) p_i$$

where for a function of degree k:

$$B_{i,0}(t) = \begin{cases} 1 & \text{if } t_i \leq t < t_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

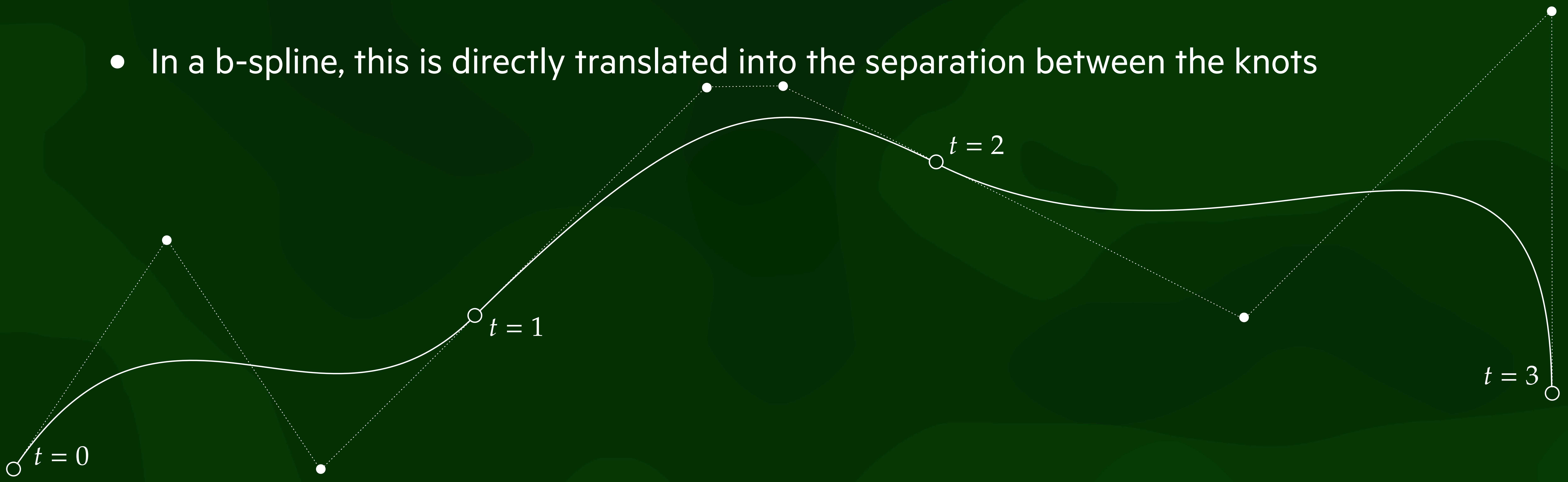
$$B_{i,k}(t) = \frac{t - t_i}{t_{i+k} - t_i} B_{i,k-1}(t) + \frac{t_{i+k+1} - t}{t_{i+k+1} - t_{i+1}} B_{i+1,k-1}(t)$$

B-spline basis functions



Uniform vs. non-uniform curves

- In a parametric curve, when the values of t are equally spaced, we call it a uniform curve.
- Non-uniform curves give us more control: we can add more parametric space where we need more detail
- In a b-spline, this is directly translated into the separation between the knots



Clamping

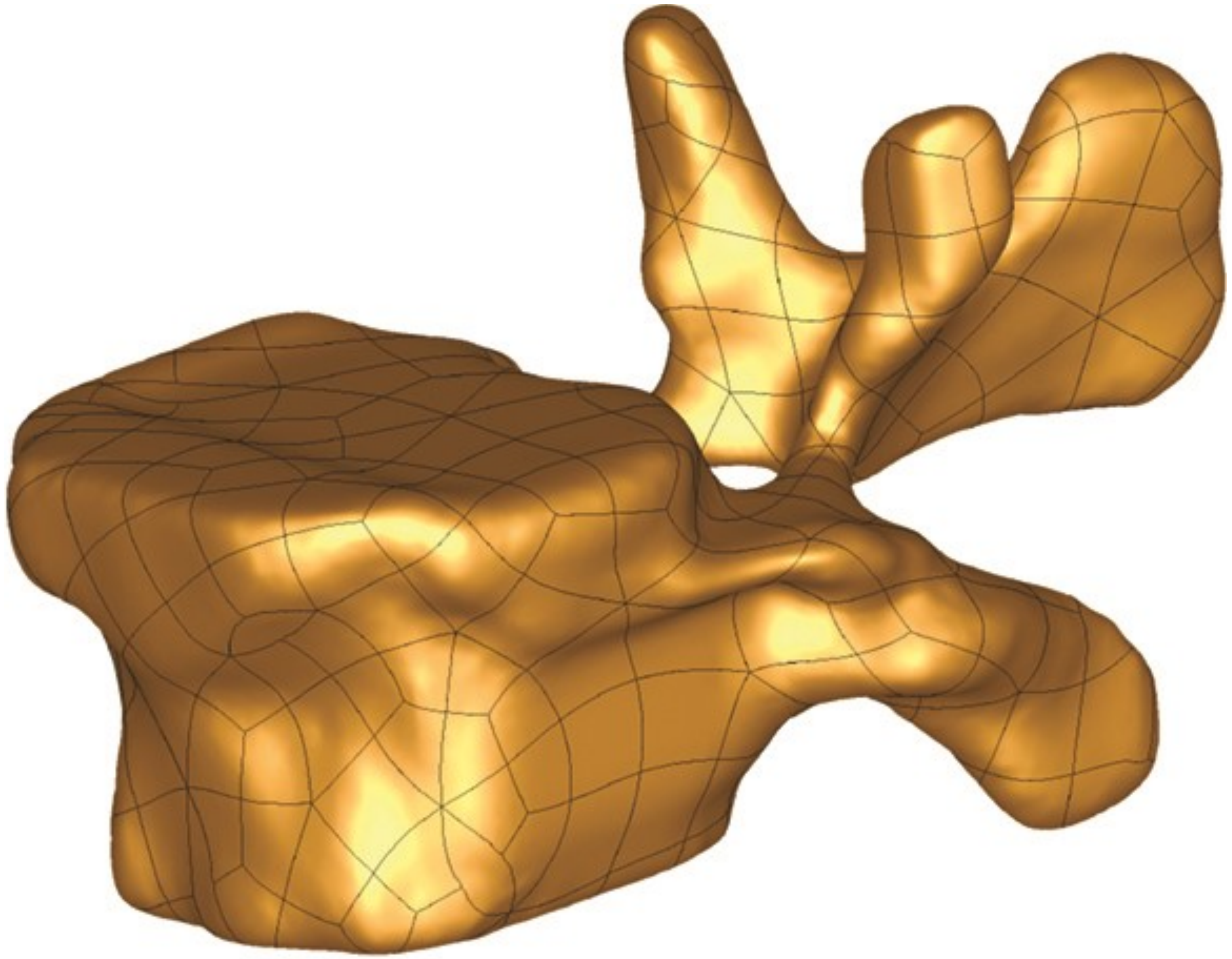
- In the extreme case, ie *clamping*, we can put multiple knots in the same location
- These reduce continuity at a point but can be used to ensure that the curve passes through a point.
- For example, for a cubic b-spline, the knot vector $(0,0,0,0,1,2,3,4,5,6,6,6,6)$ ensures that we pass through the start and end of a curve
- In practice, b-splines are almost always clamped

NURBS

- **N**on-**u**niform: local control, possible sharp features (corners / edges)
- **R**ational: possibility to present conic section (ie circle, ellipse, parabola or hyperbola)

$$C(t) = \frac{\sum_{i=0}^{n} N_{i,p}(t) w_i p_i}{\sum_{i=0}^{n} N_{i,p}(t) w_i}$$

- **B**-**s**plines



Perney, A., Bordas, S., & Kerfriden, P. (2023). NURBS-based surface generation from 3D images: spectral construction and data-driven model selection. *Journal of Computational Design and Engineering*, *10*(4), 1856-1867.

What to do next?

1. Today:
    - Go to geo1004 website and study today's lesson (**updated** 3D book Chapter 6)
    - Continue with Homework 2
2. Wednesday: CSG lecture, midterm feedback

https://3d.bk.tudelft.nl/courses/geo1004

3D geoinformation
Department of Urbanism
Faculty of Architecture and the Built Environment
Delft University of Technology