Geometric Modeling
(WS 2018/19)

Martin Held

FB Computerwissenschaften
Universität Salzburg
A-5020 Salzburg, Austria
held@cosy.sbg.ac.at

January 18, 2019
Personalia

Instructor (VO+PS): M. Held.
Email: held@cs.sbg.ac.at.
Base-URL: https://www.cosy.sbg.ac.at/~held.
Office: Universität Salzburg, Computerwissenschaften, Rm. 1.20, Jakob-Haringer Str. 2, 5020 Salzburg-Itzling.
Phone number (office): (0662) 8044-6304.
Phone number (secr.): (0662) 8044-6328.
Formalia

URL of course (VO+PS): Base-URL/teaching/geom_mod/geom_mod.html.

Lecture times (VO): Friday 12\textsuperscript{10}–13\textsuperscript{50}.
Venue (VO): T01, Computerwissenschaften, Jakob-Haringer Str. 2.

Lecture times (PS): Friday 11\textsuperscript{00}–11\textsuperscript{55}.
Venue (PS): T01, Computerwissenschaften, Jakob-Haringer Str. 2.

Note — PS is graded according to continuous-assessment mode!
— VO+PS on 21-Dec-2018 and 01-Feb-2019!
Electronic Slides and Online Material

In addition to these slides, you are encouraged to consult the WWW home-page of this lecture:

https://www.cosy.sbg.ac.at/~held/teaching/geom_mod/geom_mod.html.

In particular, this WWW page contains up-to-date information on the course, plus links to online notes, slides and (possibly) sample code.
A Few Words of Warning

- I hope that these slides will serve as a practice-minded introduction to various aspects of geometric modeling. I would like to warn you explicitly not to regard these slides as the sole source of information on the topics of my course. It may and will happen that I’ll use the lecture for talking about subtle details that need not be covered in these slides! In particular, the slides won’t contain all sample calculations, proofs of theorems, demonstrations of algorithms, or solutions to problems posed during my lecture. That is, by making these slides available to you I do not intend to encourage you to attend the lecture on an irregular basis.

- See also In Praise of Lectures by T.W. Körner.

- A basic knowledge of calculus, linear algebra, discrete mathematics, and geometric computing, as taught in standard undergraduate CS courses, should suffice to take this course. It is my sincere intention to start at such a hypothetical low level of “typical prior undergrad knowledge”. Still, it is obvious that different educational backgrounds will result in different levels of prior knowledge. Hence, you might realize that you do already know some items covered in this course, while you lack a decent understanding of some other items which I seem to presuppose. In such a case I do expect you to refresh or fill in those missing items on your own!
Acknowledgments

A small portion of these slides is based on notes and slides originally prepared by students — most notably Dominik Kaaser, Kamran Safdar, and Marko Šulejić — on topics related to geometric modeling. I would like to express my thankfulness to all of them for their help. This revision and extension was carried out by myself, and I am responsible for all errors.

I am also happy to acknowledge that I benefited from material published by colleagues on diverse topics that are partially covered in this lecture. While some of the material used for this lecture was originally presented in traditional-style publications (such as textbooks), some other material has its roots in non-standard publication outlets (such as online documentations, electronic course notes, or user manuals).

Salzburg, September 2018

Martin Held
Legal Fine Print and Disclaimer

To the best of my knowledge, these slides do not violate or infringe upon somebody else’s copyrights. If copyrighted material appears in these slides then it was considered to be available in a non-profit manner and as an educational tool for teaching at an academic institution, within the limits of the “fair use” policy. For copyrighted material we strive to give references to the copyright holders (if known). Of course, any trademarks mentioned in these slides are properties of their respective owners.

Please note that these slides are copyrighted. The copyright holder grants you the right to download and print the slides for your personal use. Any other use, including instructional use at non-profit academic institutions and re-distribution in electronic or printed form of significant portions, beyond the limits of “fair use”, requires the explicit permission of the copyright holder. All rights reserved.

These slides are made available without warrant of any kind, either express or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. In no event shall the copyright holder and/or his respective employer be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, arising out of or in connection with the use of information provided in these slides.
Recommended Textbooks I

G. Farin.
*Curves and Surfaces for CAGD: A Practical Guide.*

R.H. Bartels, J.C. Beatty, B.A. Barsky.
*An Introduction to Splines for Use in Computer Graphics and Geometric Modeling.*

H. Prautzsch, W. Boehm, M. Paluszny.
*Bézier and B-spline Techniques.*

J. Gallier.
*Curves and Surfaces in Geometric Modeling*
http://www.cis.upenn.edu/~jean/gbooks/geom1.html

N.M. Patrikalakis, T. Maekawa, W. Cho.
*Shape Interrogation for Computer Aided Design and Manufacturing.*
http://web.mit.edu/hyperbook/Patrikalakis-Maekawa-Cho/
Recommended Textbooks II

M. Botsch, L. Kobbelt, M. Pauly, P. Alliez, B. Levy.
*Polygon Mesh Processing.*
http://www.pmp-book.org/

G.E. Farin, D. Hansford.
*Practical Linear Algebra: A Geometry Toolbox.*

M.E. Mortenson.

A. Dickenstein, I.Z. Emiris (eds.).
Table of Content

Introduction

Mathematics for Geometric Modeling

Bézier Curves and Surfaces

B-Spline Curves and Surfaces

Approximation and Interpolation
Introduction
Motivation
Notation
Motivation: Evaluation of a Polynomial

Assume that we have an intuitive understanding of polynomials and consider a polynomial in $x$ of degree $n$ with coefficients $a_0, a_1, \ldots, a_n \in \mathbb{R}$, with $a_n \neq 0$:

$$p(x) := \sum_{i=0}^{n} a_i x^i = a_0 + a_1 x + a_2 x^2 + \ldots + a_{n-1} x^{n-1} + a_n x^n.$$ 

A straightforward polynomial evaluation of $p$ for a given parameter $x_0$ — i.e., the computation of $p(x_0)$ — results in $k$ multiplications for a monomial of degree $k$, plus a total of $n$ additions.

Hence, we would get

$$0 + 1 + 2 + \ldots + n = \frac{n(n+1)}{2}$$

multiplications (and $n$ additions).

Can we do better?

Yes, we can: Horner’s Algorithm consumes only $n$ multiplications and $n$ additions to evaluate a polynomial of degree $n$!
Motivation: Smoothness of a Curve

▶ What is a characteristic difference between the three curves shown below?

Answer: The green curve has tangential discontinuities at the vertices, the blue curve consists of straight-line segments and circular arcs and is tangent-continuous, while the red curve is a cubic B-spline and is curvature-continuous.

▶ By the way, when precisely is a geometric object a “curve”?
Motivation: Tangent to a Curve

What is a parameterization of the tangent line at a point $\gamma(t_0)$ of a curve $\gamma$?

Answer: If $\gamma$ is differentiable then a parameterization of the tangent line $\ell$ that passes through $\gamma(t_0)$ is given by

$$\ell(\lambda) = \gamma(t_0) + \lambda \gamma'(t_0) \quad \text{with } \lambda \in \mathbb{R}.$$  

How can we obtain $\gamma'(t)$ for $\gamma: \mathbb{R} \to \mathbb{R}^d$?
Motivation: Bézier Curve

- How can we model a “smooth” polynomial curve in $\mathbb{R}^2$ by specifying so-called “control points”. (E.g., the points $p_0, p_1, \ldots, p_{10}$ in the figure.)

- One (widely used) option is to generate a Bézier curve. (The figure shows a Bézier curve of degree 10 with 11 control points.)

- What is the degree of a Bézier curve? Which geometric and mathematical properties do Bézier curves exhibit?
Motivation: B-Spline Curve

How can we model a (piecewise) polynomial curve in $\mathbb{R}^2$ by specifying so-called “control points” such that a modification of one control point affects only a “small” portion of the curve?

Answer: Use B-spline curves.

Which geometric and mathematical properties do B-spline curves exhibit?
Motivation: NURBS

- Is it possible to parameterize a circular arc by means of a polynomial term? Or by a rational term?
- Yes, this is possible by means of a rational term:
  \[
  \left( \frac{1 - t^2}{1 + t^2}, \frac{2t}{1 + t^2} \right) \quad \text{for } t \in \mathbb{R}.
  \]
- More generally, NURBS can be used to model all types of conics by means of rational parameterizations.
Motivation: Approximation of a Continuous Function

- How can we approximate a continuous function by a polynomial?
- Answer: We can use a Bernstein approximation.
- Sample Bernstein approximations of a continuous function:

\[ f : [0, 1] \rightarrow \mathbb{R} \quad f(x) := \sin(\pi x) + \frac{1}{5} \sin(6\pi x + \pi x^2) \]

- One can prove that the Bernstein approximation \( B_{n,f} \) converges uniformly to \( f \) on the interval \([0, 1]\) as \( n \) increases, for every continuous function \( f \).
Notation: Numbers and Sets

Numbers:

- The set \( \{1, 2, 3, \ldots \} \) of natural numbers is denoted by \( \mathbb{N} \), with \( \mathbb{N}_0 := \mathbb{N} \cup \{0\} \).
- The set \( \{2, 3, 5, 7, 11, 13, \ldots \} \subset \mathbb{N} \) of prime numbers is denoted by \( \mathbb{P} \).
- The (positive and negative) integers are denoted by \( \mathbb{Z} \).
- \( \mathbb{Z}_n := \{0, 1, 2, \ldots, n - 1\} \) and \( \mathbb{Z}_n^+ := \{1, 2, \ldots, n - 1\} \) for \( n \in \mathbb{N} \).
- The reals are denoted by \( \mathbb{R} \); the non-negative reals are denoted by \( \mathbb{R}_0^+ \), and the positive reals by \( \mathbb{R}^+ \).
- Open or closed intervals \( I \subset \mathbb{R} \) are denoted using square brackets: e.g., \( I_1 = [a_1, b_1] \) or \( I_2 = [a_2, b_2[ \), with \( a_1, a_2, b_1, b_2 \in \mathbb{R} \), where the right-hand “[” indicates that the value \( b_2 \) is not included in \( I_2 \).
- The set of all elements \( a \in A \) with property \( P(a) \), for some set \( A \) and some predicate \( P \), is denoted by
  \[
  \{ x \in A : P(x) \} \quad \text{or} \quad \{ x : x \in A \land P(x) \}
  \]
  or
  \[
  \{ x \in A | P(x) \} \quad \text{or} \quad \{ x | x \in A \land P(x) \}.
  \]

Quantifiers: The universal quantifier is denoted by \( \forall \), and \( \exists \) denotes the existential quantifier.

Bold capital letters, such as \( \mathbf{M} \), are used for matrices.

The set of all (real) \( m \times n \) matrices is denoted by \( M_{m \times n} \).
Notation: Vectors

- Points are denoted by letters written in italics: \( p, q \) or, occasionally, \( P, Q \). We do not distinguish between a point and its position vector.

- The coordinates of a vector are denoted by using indices (or numbers): e.g., \( \mathbf{v} = (v_x, v_y) \) for \( \mathbf{v} \in \mathbb{R}^2 \), or \( \mathbf{v} = (v_1, v_2, \ldots, v_n) \) for \( \mathbf{v} \in \mathbb{R}^n \).

- In order to state \( \mathbf{v} \in \mathbb{R}^n \) in vector form we will mix column and row vectors freely unless a specific form is required, such as for matrix multiplication.

- The vector dot product of two vectors \( \mathbf{v}, \mathbf{w} \in \mathbb{R}^n \) is denoted by \( \langle \mathbf{v}, \mathbf{w} \rangle \). That is, \( \langle \mathbf{v}, \mathbf{w} \rangle = \sum_{i=1}^{n} v_i \cdot w_i \) for \( \mathbf{v}, \mathbf{w} \in \mathbb{R}^n \).

- The vector cross-product (in \( \mathbb{R}^3 \)) is denoted by a cross: \( \mathbf{v} \times \mathbf{w} \).

- The length of a vector \( \mathbf{v} \) is denoted by \( \| \mathbf{v} \| \).

- The straight-line segment between the points \( p \) and \( q \) is denoted by \( \overline{pq} \).

- The supporting line of the points \( p \) and \( q \) is denoted by \( \ell(p, q) \). 
Notation: Sum and Product

Consider $k$ real numbers $a_1, a_2, \ldots, a_k \in \mathbb{R}$, together with some $m, n \in \mathbb{N}$ such that $1 \leq m, n \leq k$.

$$\sum_{i=m}^{n} a_i := \begin{cases} 0 & \text{if } n < m \\ a_m & \text{if } n = m \\ (\sum_{i=m}^{n-1} a_i) + a_n & \text{if } n > m \end{cases}$$

$$\prod_{i=m}^{n} a_i := \begin{cases} 1 & \text{if } n < m \\ a_m & \text{if } n = m \\ (\prod_{i=m}^{n-1} a_i) \cdot a_n & \text{if } n > m \end{cases}$$
Mathematics for Geometric Modeling

Factorial and Binomial Coefficient
Polynomials
Elementary Differential Calculus
Elementary Differential Geometry of Curves
Elementary Differential Geometry of Surfaces
Factorial and Binomial Coefficient

Definition 1 (Factorial, Dt.: Fakultät, Faktorielle)

For $n \in \mathbb{N}_0$,

$$n! := \begin{cases} 
1 & \text{if } n \leq 1, \\
n \cdot (n - 1)! & \text{if } n > 1.
\end{cases}$$

▶ Note that $0! = 1$ by definition!

Definition 2 (Binomial coefficient, Dt.: Binomialkoeffizient)

Let $n \in \mathbb{N}_0$ and $k \in \mathbb{Z}$. Then the binomial coefficient $\binom{n}{k}$ of $n$ and $k$ is defined as follows:

$$\binom{n}{k} := \begin{cases} 
0 & \text{if } k < 0, \\
\frac{n!}{k! \cdot (n-k)!} & \text{if } 0 \leq k \leq n, \\
0 & \text{if } k > n.
\end{cases}$$

▶ The binomial coefficient $\binom{n}{k}$ is pronounced as “$n$ choose $k$”; Dt.: “$n$ über $k$”.
Factorial and Binomial Coefficient

- The following table contains the non-zero values of $\binom{n}{k}$ for $0 \leq n, k \leq 6$.

<table>
<thead>
<tr>
<th>$n$</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>4</td>
<td>6</td>
<td>4</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>5</td>
<td>10</td>
<td>10</td>
<td>5</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>6</td>
<td>15</td>
<td>20</td>
<td>15</td>
<td>6</td>
<td>1</td>
</tr>
</tbody>
</table>

- Trivial to observe:
  - Each row begins and ends with 1.
  - Initially each row contains increasing numbers till its middle but then the numbers start to decrease.
  - Each row's first half is exactly the mirror image of its second half.

### Lemma 3

Let $n \in \mathbb{N}_0$ and $k \in \mathbb{Z}$.

\[
\binom{n}{0} = \binom{n}{n} = 1 \quad \binom{n}{1} = \binom{n}{n-1} = n \quad \binom{n}{k} = \binom{n}{n-k}
\]
Factorial and Binomial Coefficient

- A simple rearrangement of the previous table yields what is known as *Pascal's Triangle* in the Western world (Blaise Pascal, 1623–1662). But it was already studied in India in the 10th century, and discussed by Omar Khayyam (1048–1131)!

<table>
<thead>
<tr>
<th>1</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>15</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>20</td>
</tr>
<tr>
<td>1</td>
<td>15</td>
<td>5</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>1</td>
</tr>
</tbody>
</table>

- All entries in this triangle, except for the left-most and right-most entries per row, are the sum of the two entries above them in the previous row.

**Theorem 4** (*Khayyam, Yang Hui, Tartaglia, Pascal*)

For $n \in \mathbb{N}$ and $k \in \mathbb{Z}$,

\[
\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}.
\]
Theorem 5 (Binomial Theorem, Dt.: Binomischer Lehrsatz)

For all $n \in \mathbb{N}_0$ and $a, b \in \mathbb{R}$,

$$(a + b)^n = \binom{n}{0} a^n + \binom{n}{1} a^{n-1} b + \cdots + \binom{n}{n} b^n$$

or, equivalently,

$$(a + b)^n = \sum_{i=0}^{n} \binom{n}{i} a^{n-i} b^i.$$
Polynomials

Definition 6 (*Monomial, Dt.: Monom*)

A (real) *monomial* in $m$ variables $x_1, x_2, \ldots, x_m$ is a product of a coefficient $c \in \mathbb{R}$ and powers of the variables $x_i$ with exponents $k_i \in \mathbb{N}_0$:

$$c \prod_{i=1}^{m} x_i^{k_i} = c \cdot x_1^{k_1} \cdot x_2^{k_2} \cdot \ldots \cdot x_m^{k_m}.$$ 

The *degree of the monomial* is given by $\sum_{i=1}^{m} k_i$.

Definition 7 (*Polynomial, Dt.: Polynom*)

A (real) *polynomial* in $m$ variables $x_1, x_2, \ldots, x_m$ is a finite sum of monomials in $x_1, x_2, \ldots, x_m$.

A polynomial is *univariate* if $m = 1$, *bivariate* if $m = 2$, and *multivariate* otherwise.

Definition 8 (*Degree, Dt.: Grad*)

The *degree of a polynomial* is the maximum degree of its monomials.
Polynomials

- Hence, a univariate polynomial over $\mathbb{R}$ with variable $x$ is a term of the form

$$a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0,$$

with coefficients $a_0, \ldots, a_n \in \mathbb{R}$ and $a_n \neq 0$.

- It is a convention to drop all monomials whose coefficients are zero.

- Univariate polynomials are usually written according to a decreasing order of exponents of their monomials.

- In that case, the first term is the \textit{leading term} which indicates the degree of the polynomial; its coefficient is the \textit{leading coefficient}.

- Univariate polynomials of degree

  0. are called constant polynomials,
  1. are called linear polynomials,
  2. are called quadratic polynomials,
  3. are called cubic polynomials,
  4. are called quartic polynomials,
  5. are called quintic polynomials.
Polynomial Arithmetic

- We define the addition of (univariate) polynomials based on the pairwise addition of corresponding coefficients:

\[
\left( \sum_{i=0}^{n} a_i x^i \right) + \left( \sum_{i=0}^{n} b_i x^i \right) := \sum_{i=0}^{n} (a_i + b_i) x^i
\]

- The multiplication of polynomials is based on the multiplication within \( \mathbb{R} \), distributivity, and the rules

\[ ax = xa \quad \text{and} \quad x^m x^k = x^{m+k} \]

for all \( a \in \mathbb{R} \) and \( m, k \in \mathbb{N} \):

\[
\left( \sum_{i=0}^{n} a_i x^i \right) \cdot \left( \sum_{j=0}^{m} b_j x^j \right) := \sum_{i=0}^{n} \sum_{j=0}^{m} (a_i b_j) x^{i+j}
\]

- Elementary properties of polynomials: One can prove easily that the addition, multiplication and composition of two polynomials as well as their derivative and antiderivative (indefinite integral) again yield a polynomial.

- Same for multivariate polynomials.
Polynomial Arithmetic

Instead of \( \mathbb{R} \) any commutative ring \((R, +, \cdot)\) and symbols \(x, y, \ldots\) that are not contained in \(R\) would do. E.g.,

\[
a_{2,3}x^2y^3 + a_{1,1}xy + a_{0,1}y + a_{0,0}
\]
with \( a_{2,3}, a_{1,1}, a_{0,1}, a_{0,0} \in R \).

Lemma 9

The set of all polynomials with coefficients in the commutative ring \((R, +, \cdot)\) and a symbol (variable) \(x \notin R\) forms a commutative ring, the ring of polynomials over \(R\), commonly denoted by \(R[x]\).

Multivariate polynomials can also be seen as univariate polynomials with coefficients out of a ring of polynomials. E.g.,

\[
a_{2,3}x^2y^3 + a_{1,1}xy + a_{0,1}y + a_{0,0} = (a_{2,3}x^2)y^3 + (a_{1,1}x + a_{0,1})y + a_{0,0}
\]
is an element of \(R[x, y] := (R[x])[y]\).

Definition 10

Two polynomials are equal if and only if the sequences of their coefficients (arranged in some specific order) are equal.
Theorem 11

The univariate polynomials of $\mathbb{R}[x]$ form an infinite vector space over $\mathbb{R}$. The so-called power basis of this vector space is given by the monomials $1, x, x^2, x^3, \ldots$.

- The monomials $1, x, x^2, x^3, \ldots, x^n$ form a basis of the vector space of polynomials of degree up to $n$ over $\mathbb{R}$, for all $n \in \mathbb{N}_0$.

- Recall: The fact that the monomials $1, x, x^2, x^3, \ldots, x^n$ form a basis of the polynomials of degree up to $n$ over $\mathbb{R}$ means that
  1. every polynomial $p \in \mathbb{R}[x]$ of degree at most $n$ can be expressed as a linear combination of those monomials: there exist $a_0, a_1, \ldots, a_n \in \mathbb{R}$ such that
     \[ p = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0, \]
  2. none of those monomials can be expressed as a linear combination of the other monomials, i.e., the monomials are linearly independent.

- The power basis is not the only meaningful basis of the polynomials $\mathbb{R}[x]$. See, e.g., the Bernstein polynomials that are used to form Bézier curves.
Polynomials: Roots

Definition 12 (Polynomial equation)

A polynomial equation (aka algebraic equation) is an equation in which a polynomial is set equal to another polynomial.

Definition 13 (Root, Dt.: Wurzel)

The polynomial \( p \in \mathbb{R}[x] \) has a root (aka zero) \( r \in \mathbb{R} \) if \( (x - r) \) divides \( p \).

Hence, if \( r \) is a root of \( p \) then \( p = (x - r) \cdot p_1 \) for some \( p_1 \in \mathbb{R}[x] \).

Definition 14 (Multiplicity, Dt.: Vielfachheit)

A root \( r \) of a polynomial \( p \in \mathbb{R}[x] \) is of multiplicity \( k \) if \( k \in \mathbb{N} \) is the maximum integer such that \( (x - r)^k \) divides \( p \).

Theorem 15 (Fundamental Theorem of Algebra)

The number of complex roots of a polynomial with real coefficients may not exceed its degree. It equals the degree if all roots are counted with their multiplicities.
Polynomials: Roots

- Recall the quadratic formula taught in secondary school for solving second-degree polynomial equations: For \( a \in \mathbb{R} \setminus \{0\} \) and \( b, c \in \mathbb{R} \),

\[
x_{1,2} := \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}
\]

yields the two (possibly complex) roots \( x_1 \) and \( x_2 \) of \( ax^2 + bx + c \).

- Similar (albeit more complex) formulas exist for cubic and quartic polynomials.

Theorem 16 (Abel-Ruffini (1824))

No algebraic solution for the roots of an arbitrary polynomial of degree five or higher exists.

- An algebraic solution is a closed-form expressions in terms of the coefficients of the polynomial that relies only on addition, subtraction, multiplication, division, raising to integer powers, and computing \( k \)-th roots (square roots, cube roots, and other integer roots).

- A closed-form expression is an expression that can be evaluated in a finite number of operations.
Theorem 17 (Vieta’s formula)

Let \( n, k \in \mathbb{N} \) with \( 1 \leq k \leq n \), and \( a_0, a_1, \ldots, a_n \in \mathbb{R} \) with \( a_n \neq 0 \). Then the coefficient \( a_{n-k} \) of the degree-\( n \) polynomial

\[
a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x + a_0
\]

is related to a signed sum of all possible \( k \)-at-a-time subproducts of roots:

\[
\sum_{1 \leq i_1 < i_2 < \ldots < i_k \leq n} x_{i_1} \cdot x_{i_2} \cdot \ldots \cdot x_{i_k} = (-1)^k \frac{a_{n-k}}{a_n}
\]

where \( x_1, x_2, \ldots, x_n \) are the \( n \) (possibly complex) roots of the polynomial.

François Viète (Franciscus Vieta, 1540–1603).
Polynomials: Roots

Corollary 18

For \( a, b, c \in \mathbb{R} \), the roots \( r_1, r_2 \) of the quadratic polynomial \( ax^2 + bx + c \) satisfy

\[
\begin{align*}
    r_1 + r_2 &= -\frac{b}{a} \\
    r_1 \cdot r_2 &= \frac{c}{a}.
\end{align*}
\]

Corollary 19

For \( a, b, c, d \in \mathbb{R} \), the roots \( r_1, r_2, r_3 \) of the cubic polynomial \( ax^3 + bx^2 + cx + d \) satisfy

\[
\begin{align*}
    r_1 + r_2 + r_3 &= -\frac{b}{a} \\
    r_1 \cdot r_2 + r_1 \cdot r_3 + r_2 \cdot r_3 &= \frac{c}{a} \\
    r_1 \cdot r_2 \cdot r_3 &= -\frac{d}{a}.
\end{align*}
\]

Polynomials: Function

Definition 20 (*Polynomial function*; Dt.: *Polynomfunktion*)

A (univariate real) function \( f : \mathbb{R} \to \mathbb{R} \) in one argument \( x \) is a *polynomial function* if there exist \( n \in \mathbb{N}_0 \) and \( a_0, a_1, \ldots, a_n \in \mathbb{R} \) such that

\[
f(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0 \quad \text{for all } x \in \mathbb{R}.
\]

- As usual, two (polynomial) functions are identical if their values are identical for all arguments in \( \mathbb{R} \).
- **Note:** Two different polynomials may result in the same polynomial function! (E.g., over finite fields.)
- **Note:** While some fields of mathematics (e.g., abstract algebra) make a clear distinction between polynomials and polynomial functions, we will freely mix these two terms. Also, unless noted explicitly, we will only deal with polynomials over \( \mathbb{R} \).
- **Note:** Polynomial functions may come in disguise: \( f(x) := \cos(2 \arccos(x)) \) is a polynomial function over \([-1, 1]\), since we have \( f(x) = 2x^2 - 1 \) for all \( x \in [-1, 1] \).
Consider a polynomial $p \in \mathbb{R}[x]$ of degree $n$ with coefficients $a_0, a_1, \ldots, a_n \in \mathbb{R}$, with $a_n \neq 0$:

$$p(x) := \sum_{i=0}^{n} a_i x^i = a_0 + a_1 x + a_2 x^2 + \ldots + a_{n-1} x^{n-1} + a_n x^n.$$ 

A straightforward polynomial evaluation of $p$ for a given parameter $x_0$ results in $k$ multiplications for a monomial of degree $k$, plus a total of $n$ additions.

Hence, we would get

$$0 + 1 + 2 + \ldots + n = \frac{n(n+1)}{2}$$

multiplications (and $n$ additions).

Can we do better?

Obviously, we can reduce the number of multiplications to $O(n \log n)$ by resorting to exponentiation by squaring:

$$x^n = \begin{cases} 
  x \left(x^2\right)^{\frac{n-1}{2}} & \text{if } n \text{ is odd}, \\
  \left(x^2\right)^{\frac{n}{2}} & \text{if } n \text{ is even}.
\end{cases}$$

Can we do even better?
**Horner’s Algorithm for Evaluation of a Polynomial**

**Horner’s Algorithm**: The idea is to rewrite the polynomial such that

\[
p(x) = a_0 + x(a_1 + x(a_2 + \ldots + x(a_{n-2} + x(a_{n-1} + x a_n))\ldots))
\]

and compute the result \( h_0 := p(x_0) \) as follows:

\[
h_n := a_n \\
h_i := x_0 \cdot h_{i+1} + a_i \quad \text{for } i = 0, 1, 2, \ldots, n - 1
\]

```c
/** Evaluates a polynomial of degree n at point x *
 * @param p: array of n+1 coefficients *
 * @param n: the degree of the polynomial *
 * @param x: the point of evaluation *
 * @return the evaluation result *
 */
double evaluate(double *p, int n, double x)
{
    double h = p[n];

    for (int i = n - 1; i >= 0; --i)
        h = x * h + p[i];

    return h;
}
```
Horner’s Algorithm for Evaluation of a Polynomial

Lemma 21

Horner’s Algorithm consumes \( n \) multiplications and \( n \) additions to evaluate a polynomial of degree \( n \).

Caveat

Subtractive cancellation could occur at any time, and there is no easy way to determine a priori whether and for which data it will indeed occur.

- Subtractive cancellation: Subtracting two nearly equal numbers (on a conventional IEEE-754 floating-point arithmetic) may yield a result with few or no meaningful digits. Aka: catastrophic cancellation.
Forward Differencing

- If a polynomial has to be evaluated at \( k + 1 \) evenly spaced points \( x_0, x_1, \ldots, x_k \), with \( x_{i+1} = x_i + \delta \) for \( 0 \leq i < k \), then forward differencing is faster than Horner’s Algorithm.

- Consider a polynomial of degree one:

\[
p(x) = a_0 + a_1 x
\]

- The difference in the function values \( p(x_i) \) and \( p(x_{i+1}) \) of two neighboring points \( x_i \) and \( x_{i+1} \) is

\[
\Delta := p(x_{i+1}) - p(x_i) = p(x_i + \delta) - p(x_i) = a_0 + a_1(x_i + \delta) - (a_0 + a_1 x_i) = a_1 \delta.
\]

- Hence, to evaluate the polynomial at several points, we may start with the evaluation of \( p(x_0) \) and recursively compute

\[
p(x_{i+1}) = p(x_i + \delta) = p(x_i) + \Delta, \quad \text{with} \quad \Delta := a_1 \delta.
\]
Forward Differencing

- For a quadratic polynomial $p(x) = a_0 + a_1 x + a_2 x^2$ the difference of the function values of neighboring points is

  $$
  \Delta_1(x_i) := p(x_{i+1}) - p(x_i) = p(x_i + \delta) - p(x_i)
  $$

  $$
  = a_0 + a_1 (x_i + \delta) + a_2 (x_i + \delta)^2 - (a_0 + a_1 x_i + a_2 x_i^2)
  $$

  $$
  = a_1 \delta + a_2 \delta^2 + 2a_2 x_i \delta
  $$

- As $\Delta_1(x)$ is itself a linear polynomial, it can also be evaluated using forward differencing:

  $$
  \Delta_2(x) := \Delta_1(x + \delta) - \Delta_1(x) = 2a_2 \delta^2
  $$

- This approach can be extended to polynomials of any degree.

- One can conclude that for a polynomial of degree $n$ each successive evaluation requires $n$ additions.
Differentiation of Functions of One Variable

Definition 22 (*Derivative, Dt.: Ableitung*)

Let \( S \subseteq \mathbb{R} \) be an open set. A (scalar-valued) function \( f : S \to \mathbb{R} \) is *differentiable* at an interior point \( x_0 \in S \) if

\[
\lim_{h \to 0} \frac{f(x_0 + h) - f(x_0)}{h}
\]

exists, in which case the limit is called the *derivative* of \( f \) at \( x_0 \), denoted by \( f'(x_0) \).

Definition 23

Let \( S \subseteq \mathbb{R} \) be an open set. A (scalar-valued) function \( f : S \to \mathbb{R} \) is *differentiable on* \( S \) if it is differentiable at every point of \( S \).

If \( f \) is differentiable on \( S \) and \( f' \) is continuous on \( S \) then \( f \) is *continuously differentiable on* \( S \). In this case \( f \) is said to be of *differentiability class* \( C^1 \).

- By taking one-sided limits one can also consider one-sided derivatives on the boundary of closed sets \( S \).
- By applying differentiation to \( f' \), a second derivative \( f'' \) of \( f \) can be defined. Inductively, we obtain \( f^{(n)} \) by differentiating \( f^{(n-1)} \).
Differentiation of Functions of One Variable

Definition 24 (\(C^k\), Dt.: \(k\)-mal stetig differenzierbar)

Let \(S \subseteq \mathbb{R}\) be an open set. A function \(f: S \to \mathbb{R}\) that has \(k\) successive derivatives is called \(k\) \textit{times differentiable}. If, in addition, the \(k\)-th derivative is continuous, then the function is said to be of \textit{differentiability class} \(C^k\).

- If the \(k\)-th derivative of \(f\) exists then the continuity of \(f^{(0)}, f^{(1)}, \ldots, f^{(k-1)}\) is implied.

Definition 25 (\textit{Smooth}, Dt.: \textit{glatt})

Let \(S \subseteq \mathbb{R}\) be an open set. A function \(f: S \to \mathbb{R}\) is called \textit{smooth} if it has infinitely many derivatives, denoted by the class \(C^\infty\).

- We have \(C^\infty \subset C^i \subset C^j\), for all \(i, j \in \mathbb{N}_0\) if \(i > j\).

- Notation:
  - \(f^{(0)}(x) := f(x)\) for convenience purposes.
  - \(f'(x) = f^{(1)}(x) = \frac{d}{dx} f(x) = \frac{df}{dx}(x)\).
  - \(f''(x) = f^{(2)}(x) = \frac{d^2}{dx^2} f(x) = \frac{d^2 f}{dx^2}(x)\).
  - \(f'''(x) = f^{(3)}(x) = \frac{d^3}{dx^3} f(x) = \frac{d^3 f}{dx^3}(x)\).
  - \(f^{(n)}(x) = f^{(n)}(x) = \frac{d^n}{dx^n} f(x) = \frac{d^n f}{dx^n}(x)\).
Differentiation of Functions of One Variable

Theorem 26 (Chain Rule, Dt.: Kettenregel)
Suppose that \( f : S \rightarrow \mathbb{R} \) is differentiable at \( x_0 \) and \( g : f(S) \rightarrow \mathbb{R} \) is differentiable at \( f(x_0) \). Then the composite function \( h := g \circ f \), defined as \( h(x) := g(f(x)) \), is differentiable at \( x_0 \), and we have
\[
h'(x_0) = g'(f(x_0)) \cdot f'(x_0).
\]

Theorem 27 (Rolle’s Theorem, Dt.: Satz von Rolle)
If \( f : [a, b] \rightarrow \mathbb{R} \) is continuous on the closed interval \( [a, b] \), with \( a \neq b \), and differentiable on the open interval \( ]a, b[ \), and if \( f(a) = f(b) \) then there exists \( c \in ]a, b[ \) such that \( f'(c) = 0 \).

Theorem 28 (Mean Value Theorem, Dt.: Mittelwertsatz der Differentialrechnung)
If \( f : [a, b] \rightarrow \mathbb{R} \) is continuous on the closed interval \( [a, b] \), with \( a \neq b \), and differentiable on the open interval \( ]a, b[ \), then there exists \( c \in ]a, b[ \) such that
\[
f'(c) = \frac{f(b) - f(a)}{b - a}.
\]
Differentiation of Functions of One Variable

Definition 29

For $n \in \mathbb{N}$ consider $n$ functions $f_i : S \to \mathbb{R}$ (with $1 \leq i \leq n$) and define $f : S \to \mathbb{R}^n$ as

$$f(x) := \begin{pmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_n(x) \end{pmatrix}.$$ 

Then the (vector-valued) function $f$ is differentiable at an interior point $x_0 \in S$ if and only if $f_i$ is differentiable at $x_0$, for all $i \in \{1, 2, \ldots, n\}$. The derivative of $f$ at $x_0$ is given by

$$f'(x_0) := \begin{pmatrix} f'_1(x_0) \\ f'_2(x_0) \\ \vdots \\ f'_n(x_0) \end{pmatrix}.$$ 

All other definitions related to differentiability carry over from scalar-valued functions to vector-valued functions of one variable in a natural way.
Definition 30 (Partial derivative, Dt.: partielle Ableitung)

Let \( S \subseteq \mathbb{R}^m \) be an open set. The partial derivative of a (vector-valued) function \( f : S \to \mathbb{R}^n \) at point \((a_1, a_2, \ldots, a_m) \in S\) with respect to the \(i\)-th coordinate \(x_i\) is defined as

\[
\frac{\partial f}{\partial x_i}(a_1, a_2, \ldots, a_m) := \lim_{h \to 0} \frac{f(a_1, a_2, \ldots, a_i + h, \ldots, a_m) - f(a_1, a_2, \ldots, a_i, \ldots, a_m)}{h},
\]

if this limit exists.

- Hence, for a partial derivative with respect to \(x_i\) we simply differentiate \(f\) with respect to \(x_i\) according to the rules for ordinary differentiation, while regarding all other variables as constants.
- That is, for the purpose of the partial derivative with respect to \(x_i\) we regard \(f\) as univariate function in \(x_i\) and apply standard differentiation rules.
- Some authors prefer to write \(f_x\) instead of \(\frac{\partial f}{\partial x}\).
- We will mix notations as we find it convenient.

Note

A function of \(m\) variables may have all first-order partial derivatives at a point \((a_1, \ldots, a_m)\) but still need not be continuous at \((a_1, \ldots, a_m)\).
Differentiation of Functions of Several Variables

Higher-order partial derivatives of $f$ are obtained by repeated computation of a partial derivative of a (higher-order) partial derivative of $f$.

Does it matter in which sequence we compute higher-order partial derivatives?

Theorem 31

Suppose that $\frac{\partial f}{\partial x}$, $\frac{\partial f}{\partial y}$, and $\frac{\partial^2 f}{\partial y \partial x}$ exist on a neighborhood of $(x_0, y_0)$, and that $\frac{\partial^2 f}{\partial y \partial x}$ is continuous at $(x_0, y_0)$. Then $\frac{\partial^2 f}{\partial x \partial y}(x_0, y_0)$ exists, and we have

$$\frac{\partial^2 f}{\partial x \partial y}(x_0, y_0) = \frac{\partial^2 f}{\partial y \partial x}(x_0, y_0).$$

Note the difference between the Leibniz notation and the subscript notation for higher-order mixed partial derivatives!

$$\frac{\partial^2 f}{\partial x \partial y}(x, y) := \frac{\partial}{\partial x} \left( \frac{\partial f}{\partial y} \right)(x, y) \quad \text{but} \quad f_{xy}(x, y) := (f_x)_y(x, y)$$

Also, note that Theorem 31 does not imply that it could not be simpler to compute, say, $\frac{\partial^2 f}{\partial y \partial x}$ rather than $\frac{\partial^2 f}{\partial x \partial y}$:

$$f(x, y) := xe^{2y} + \sqrt{e^y \sin(y \tan(\log y))} + \sqrt{1 + y^2 \cos^2 y}$$
Differentiation of Functions of Several Variables

**Definition 32** (*Differentiable, Dt.: total differenzierbar*)

A function $f : S \rightarrow \mathbb{R}^n$ of $m$ variables is *differentiable* at a point $a := (a_1, \ldots, a_m) \in S$ if there exists an $n \times m$ matrix $\mathbf{J}$ such that

$$\lim_{x \to a} \frac{f(x) - f(a) - \mathbf{J}(x - a)}{\|x - a\|} = 0.$$

**Theorem 33**

If a function $f : S \rightarrow \mathbb{R}^n$ of $m$ variables is differentiable at a point $a \in S$ then the coefficients $a_{ij}$ of the matrix $\mathbf{J}$ of Def. 32 are given by

$$a_{ij} = \frac{\partial f_i}{\partial x_j}(a) \quad \text{for} \ i \in \{1, 2, \ldots, n\} \text{ and } j \in \{1, 2, \ldots, m\}.$$

The matrix $\mathbf{J}$ is called *Jacobi matrix* of $f$ at $a$. 
Differentiation of Functions of Several Variables

Theorem 34

If a function \( f : S \to \mathbb{R}^n \) of \( m \) variables is differentiable at a point \( a \in S \) then it is continuous at \( a \).

Theorem 35

If \( \frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots, \frac{\partial f}{\partial x_m} \) exist for a function \( f : S \to \mathbb{R}^n \) of \( m \) variables on a neighborhood of a point \( a \in S \) and are continuous at \( a \) then \( f \) is differentiable at \( a \).

Definition 36 (Continuously differentiable, Dt.: stetig differenzierbar)

We say that a function \( f : S \to \mathbb{R}^n \) of \( m \) variables is continuously differentiable on an open subset \( S \) of \( \mathbb{R}^m \) if \( \frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots, \frac{\partial f}{\partial x_m} \) exist and are continuous on \( S \).
Curves

- Intuitively, a curve in $\mathbb{R}^2$ is generated by a continuous motion of a pencil on a sheet of paper.
- A formal mathematical definition is not entirely straightforward, and the term “curve” is associated with two closely related notions: kinematic and geometric.
- In the kinematic setting, a (parameterized) curve is a function of one real variable.
- In the geometric setting, a curve, also called an arc, is a 1-dimensional subset of space.
- Both notions are related:
  - The image of a parameterized curve describes an arc.
  - Conversely, an arc admits a parameterization.
- Since the kinematic setting is easier to introduce, we resort to a kinematic definition of “curve”.
- Note that fairly counter-intuitive curves exist: e.g., space-filling curves like the Sierpinski curve.
Caveat: Sierpinski Curves

- Sierpinski curves are a sequence of recursively defined continuous and closed curves in $\mathbb{R}^2$.
- Sierpinski curve of orders 1–3:

Their limit curve, the Sierpinski curve, is a space-filling curve: It fills the unit square completely!

- The Euclidean length $L_n$ of the order-$n$ Sierpinski curve is

$$L_n = \frac{2}{3} (1 + \sqrt{2}) 2^n - \frac{1}{3} (2 - \sqrt{2}) \frac{1}{2^n}.$$

Hence, its length grows exponentially and unboundedly as $n$ grows.
Curves in $\mathbb{R}^n$

**Definition 37 (Curve, Dt.: Kurve)**

Let $I \subseteq \mathbb{R}$ be an interval of the real line. A continuous (vector-valued) mapping $\gamma: I \rightarrow \mathbb{R}^n$ is called a *parameterization* of $\gamma(I)$, and $\gamma(I)$ is called the (parametric) *curve* parameterized by $\gamma$.

- Well-known examples of parameterized curves include a straight-line segment, a circular arc, and a helix.
- E.g., $\gamma: [0, 1] \rightarrow \mathbb{R}^3$ with

$$
\gamma(t) := \begin{pmatrix}
    p_x + t \cdot (q_x - p_x) \\
    p_y + t \cdot (q_y - p_y) \\
    p_z + t \cdot (q_z - p_z)
\end{pmatrix}
$$

maps $[0, 1]$ to a straight-line segment from point $p$ to $q$.

- Sometimes the interval $I$ is called the *domain* of $\gamma$, and $\gamma(I)$ is called *image*.
Curves in $\mathbb{R}^n$

Definition 38 (Start and end point)
If $I$ is a closed interval $[a, b]$, for some $a, b \in \mathbb{R}$, then we call $\gamma(a)$ the start point and $\gamma(b)$ the end point of the curve $\gamma: I \to \mathbb{R}^n$.

Definition 39 (Closed, Dt.: geschlossen)
A parameterization $\gamma: I \to \mathbb{R}^n$ is said to be closed (or a loop) if $I$ is a closed interval $[a, b]$, for some $a, b \in \mathbb{R}$, and $\gamma(a) = \gamma(b)$.

Definition 40 (Simple, Dt.: einfach)
A parameterization $\gamma: I \to \mathbb{R}^n$ is said to be simple if $\gamma(t_1) = \gamma(t_2)$ for $t_1 \neq t_2 \in I$ implies $\{t_1, t_2\} = \{a, b\}$ and $I = [a, b]$, for some $a, b \in \mathbb{R}$.

- Hence, if $\gamma: I \to \mathbb{R}^n$ is simple then it is injective on $\text{int}(I)$.
- A simple and closed curve in $\mathbb{R}^2$ is also called Jordan curve.
Curves in $\mathbb{R}^n$

- Many properties of curves can also be stated independently of a specific parameterization. E.g., we can regard a curve $C$ to be simple if there exists one parameterization of $C$ that is simple.
- Similarly, we will frequently speak about a closed curve rather than about a closed parameterization of a curve.
- Hence, the distinction between a curve and (one of) its parameterizations is often blurred.
- For the sake of simplicity, we will not distinguish between a curve $C$ and one of its parameterizations $\gamma$ if the meaning is clear.
- Similarly, we will frequently call $\gamma$ a curve.
Differentiable Curves

Definition 41 (**C\(^r\)**-parameterization)

If \( \gamma: I \rightarrow \mathbb{R}^n \) is \( r \) times continuously differentiable then \( \gamma \) is called a parametric curve of class \( C^r \), or a \( C^r \)-parameterization of \( \gamma(I) \), or simply a \( C^r \)-curve.

If \( I = [a, b] \), then \( \gamma \) is called a closed \( C^r \)-parameterization if \( \gamma^{(k)}(a) = \gamma^{(k)}(b) \) for all \( 0 \leq k \leq r \).

▶ One-sided differentiability is meant at the endpoints of \( I \) if \( I \) is a closed interval.

Definition 42 (Regular, Dt.: regulär)

A \( C^r \)-curve \( \gamma: I \rightarrow \mathbb{R}^n \) is called regular of order \( k \) if for all \( t \in I \) the vectors \( \{\gamma'(t), \gamma''(t), \ldots, \gamma^{(k)}(t)\} \) are linearly independent, for some \( 0 < k \leq r \).

In particular, \( \gamma \) is called regular if \( \gamma'(t) \neq 0 \in \mathbb{R}^n \) for all \( t \in I \).

Definition 43 (Singular, Dt.: singulär)

For a \( C^1 \)-curve \( \gamma: I \rightarrow \mathbb{R}^n \) and \( t_0 \in I \), the point \( \gamma(t_0) \) is called a singular point of \( \gamma \) if \( \gamma'(t_0) = 0 \).

▶ Note: Regularity and singularity depend on the parameterization!
Definition 44 (*Smooth curve, Dt.: glatte Kurve*)

If \( \gamma: I \rightarrow \mathbb{R}^n \) has derivatives of all orders then \( \gamma \) is (the parameterization of) a smooth curve (or of class \( C^\infty \)).

Definition 45 (*Piecewise smooth curve, Dt.: stückweise glatte Kurve*)

If \( I \) is the union of a finite number of sub-intervals over each of which \( \gamma: I \rightarrow \mathbb{R}^n \) is smooth then \( \gamma \) is piecewise smooth.

▶ Note: Smoothness depends on the parameterization!
▶ There do exist curves which are continuous everywhere but differentiable nowhere.
Equivalence of Parameterizations in $\mathbb{R}^n$

Note that parameterizations of a curve (regarded as a set $C \subset \mathbb{R}^n$) need not be unique: Two different parameterizations $\gamma: I \rightarrow \mathbb{R}^n$ and $\beta: J \rightarrow \mathbb{R}^n$ may exist such that $C = \gamma(I) = \beta(J)$.

\[
\gamma(t) := \begin{pmatrix} \cos 2\pi t \\ \sin 2\pi t \end{pmatrix}
\]

\[
\beta(t) := \begin{pmatrix} \cos 2\pi t \\ -\sin 2\pi t \end{pmatrix}
\]

Figure: $\gamma(t)$ for $t \in [0, 0.9]$

Figure: $\beta(t)$ for $t \in [0, 0.9]$
Equivalence of Parameterizations in \( \mathbb{R}^n \)

**Definition 46 (Reparameterization, Dt.: Umparameterisierung)**

Let \( \gamma : I \rightarrow \mathbb{R}^n \) and \( \beta : J \rightarrow \mathbb{R}^n \) both be \( C^r \)-curves, for some \( r \in \mathbb{N}_0 \). We consider \( \gamma \) and \( \beta \) as *equivalent* if a function \( \phi : I \rightarrow J \) exists, such that

\[
\beta(\phi(t)) = \gamma(t) \quad \forall t \in I,
\]

and

1. \( \phi \) is continuous, strictly monotonously increasing and bijective,
2. both \( \phi \) and \( \phi^{-1} \) are \( r \) times continuously differentiable.

In this case the parametric curve \( \beta \) is called a *reparameterization* of \( \gamma \).

**Caveat**

There is no universally accepted definition of a reparameterization! Some authors drop the monotonicity or the differentiability of \( \phi \), while others even require \( \phi \) to be smooth.
Theorem 47

Let $\beta: J \rightarrow \mathbb{R}^n$ be a reparameterization of $\gamma: I \rightarrow \mathbb{R}^n$. If $\gamma$ is regular then $\beta$ is regular.

Proof: Let $\gamma$ be regular. Hence, $\gamma$ is at least a $C^1$-curve such that $\gamma'(t) \neq 0$ for all $t \in I$. Let $\phi: I \rightarrow J$ be the mapping from $I$ to $J$ that establishes the equivalence of $\gamma$ and $\beta$. Hence, $\phi$ and $\phi^{-1}$ are continuous, strictly monotonously increasing, bijective and at least once continuously differentiable.

Since $\phi(\phi^{-1}(u)) = u$ for all $u \in J$, the Chain Rule 26 implies

$$\phi'(\phi^{-1}(u)) \cdot \frac{d}{du} \phi^{-1}(u) = 1,$$

i.e.,

$$\frac{d}{du} \phi^{-1}(u) \neq 0,$$

for all $u \in J$.

Since $\beta(u) = \gamma(\phi^{-1}(u))$, we know that $\beta(u)$ is continuously differentiable, and once again by applying the Chain Rule 26 we get

$$\beta'(u) = \gamma'(\phi^{-1}(u)) \cdot \frac{d}{du} \phi^{-1}(u) \neq 0$$

for all $u \in J$. 

$\square$
Arc Length

Definition 48 (Decomposition, Dt.: Unterteilung)

Consider \( \gamma : I \rightarrow \mathbb{R}^n \), with \( I := [a, b] \). A decomposition, \( P \), of the closed interval \( I \) is a sequence of \( m + 1 \) numbers \( t_0, t_1, t_2, \ldots, t_m \), for some \( m \in \mathbb{N} \), such that

\[
a = t_0 < t_1 < t_2 < \cdots < t_m = b.
\]

The length \( L_P(\gamma) \) of the polygonal chain \( (\gamma(t_0), \gamma(t_1), \gamma(t_2), \ldots, \gamma(t_m)) \) that corresponds to the decomposition \( t_0, t_1, t_2, \ldots, t_m \) is given by

\[
L_P(\gamma) := \sum_{j=0}^{m-1} \| \gamma(t_{j+1}) - \gamma(t_j) \|
\]

\[
= \| \gamma(t_1) - \gamma(t_0) \| + \| \gamma(t_2) - \gamma(t_1) \| + \cdots + \| \gamma(t_m) - \gamma(t_{m-1}) \|.
\]

We denote the set of all decompositions of \([a, b]\) by \( \mathcal{P}[a, b] \).
Arc Length

Definition 49 (Arc length, Dt.: Bogenlänge)
Consider $\gamma : I \rightarrow \mathbb{R}^n$, with $I := [a, b]$. The arc length of $\gamma(I)$ is given by

$$\sup \{ L_P(\gamma) : P \in \mathcal{P}[a, b] \},$$

i.e., by the supremum (over all decompositions $t_0, t_1, t_2, \ldots, t_m$ of $I$) of the length of the polygonal chain defined by $\gamma(t_0), \gamma(t_1), \gamma(t_2), \ldots, \gamma(t_m)$.

Definition 50 (Rectifiable, Dt.: rektifizierbar)
If the arc length of $\gamma : I \rightarrow \mathbb{R}^n$ is a finite number then $\gamma(I)$ is called rectifiable.

Lemma 51
The arc length of a curve does not change for equivalent parameterizations.

Sketch of Proof: Suppose that $\gamma(t) = \beta(\phi(t))$ for all $t \in I$, for $\beta : J \rightarrow \mathbb{R}^n$. Every decomposition $t_0, t_1, t_2, \ldots, t_m$ of $I$ maps to a decomposition $\phi(t_0), \phi(t_1), \phi(t_2), \ldots, \phi(t_m)$ of $J$ such that $\gamma(t_i) = \beta(\phi(t_i))$ for all $1 \leq i \leq m$. Hence, there is a bijection from the set of decompositions of $I$ to the set of decompositions of $J$, and it does not matter which set is used for determining the supremum of all possible chain lengths. $\square$
Arc Length: Non-Rectifiable Curve

- Curves exist that are non-rectifiable, i.e., for which there is no upper bound on the length of their polygonal approximations.
- Example of a non-rectifiable curve: The graph of the function defined by \( f(0) := 0 \) and \( f(x) := x \sin \left( \frac{1}{x} \right) \) for \( 0 < x \leq a \), for some \( a \in \mathbb{R}^+ \). It defines a curve \( \gamma(t) := \begin{pmatrix} t \\ f(t) \end{pmatrix} \).

![Graph of function](image)

The graph of \( f(x) := x \sin \left( \frac{1}{x} \right) \) for \( x \in [0, 2] \)
Curves exist that are non-rectifiable, i.e., for which there is no upper bound on the length of their polygonal approximations.

Example of a non-rectifiable curve: The graph of the function defined by $f(0) := 0$ and $f(x) := x \sin \left( \frac{1}{x} \right)$ for $0 < x \leq a$, for some $a \in \mathbb{R}^+$. It defines a curve $\gamma(t) := \left( t \atop f(t) \right)$.

The graph of $f(x) := x \sin \left( \frac{1}{x} \right)$ for $x \in [0, \frac{1}{2}]$
Arc Length: Non-Rectifiable Curve

- Curves exist that are non-rectifiable, i.e., for which there is no upper bound on the length of their polygonal approximations.
- Example of a non-rectifiable curve: The graph of the function defined by \( f(0) := 0 \) and \( f(x) := x \sin \left( \frac{1}{x} \right) \) for \( 0 < x \leq a \), for some \( a \in \mathbb{R}^+ \). It defines a curve 
  \[ \gamma(t) := \begin{pmatrix} t \\ f(t) \end{pmatrix}. \]

The graph of \( f(x) := x \sin \left( \frac{1}{x} \right) \) for \( x \in [0, \frac{1}{32}] \)
Arc Length: Non-Rectifiable Curve

- Example of a non-rectifiable closed curve: The *Koch snowflake* [Koch 1904].

The length of the curve after the $n$-th iteration is $(4/3)^n$ of the original triangle perimeter. (Its fractal dimension is $\log 4/\log 3 \approx 1.261$.)
Lemma 52

If \( \gamma: I \rightarrow \mathbb{R}^n \) is a \( C^1 \)-curve then \( \gamma(I) \) is rectifiable.

Proof: Let \( t_0, t_1, t_2, \ldots, t_m \) be a decomposition \( P \) of \( I := [a, b] \). Since \( \gamma'(t) \) exists and is continuous on all of \([a, b]\), we know that there exists \( M_k \) which serves as an upper bound for \( |\gamma'_k| \) on \([a, b]\), for all \( 1 \leq k \leq n \). Hence,

\[
L_P(\gamma) = \sum_{j=0}^{m-1} \| \gamma(t_{j+1}) - \gamma(t_j) \| \leq \sum_{j=0}^{m-1} \sum_{k=1}^{n} |\gamma_k(t_{j+1}) - \gamma_k(t_j)|
\]

\[
(*) \sum_{j=0}^{m-1} \sum_{k=1}^{n} |\gamma'_k(x_{kj})||t_{j+1} - t_j| \quad \text{for suitable } x_{kj}
\]

\[
\leq \sum_{j=0}^{m-1} \sum_{k=1}^{n} M_k(t_{j+1} - t_j) = \sum_{k=1}^{n} M_k \sum_{j=0}^{m-1} (t_{j+1} - t_j)
\]

\[
= (b - a) \sum_{k=1}^{n} M_k
\]

where equality at \((*)\) holds due to Mean Value Theorem 28.
Arc Length

Theorem 53
Let $\gamma : I \to \mathbb{R}^n$ be a $C^1$ curve, with $I := [a, b]$. Then the arc length of $\gamma(I)$ is given by

$$\int_a^b \|\gamma'(u)\| \, du.$$ 

Corollary 54
Let $\gamma : I \to \mathbb{R}^n$ be a $C^1$ curve, and $[a, b] \subseteq I$. Then the arc length of $\gamma([a, b])$ is given by

$$\int_a^b \|\gamma'(u)\| \, du.$$
Arc Length: Unit Speed

Definition 55 (Speed, Dt.: Geschwindigkeit)

If \( \gamma: I \rightarrow \mathbb{R}^n \) is a \( C^1 \)-curve then the vector \( \gamma'(t) \) is the velocity vector at parameter \( t \), and \( \|\gamma'(t)\| \) gives the speed at parameter \( t \), for all \( t \in I \).

Definition 56 (Natural parameterization)

A \( C^1 \)-curve \( \gamma: I \rightarrow \mathbb{R}^n \) is called natural (or at unit speed) if \( \|\gamma'(t)\| = 1 \) for all \( t \in I \).

Theorem 57

If \( \gamma: I \rightarrow \mathbb{R}^n \), with \( I := [a, b] \), is a regular curve then there exists an equivalent reparameterization \( \tilde{\gamma} \) that has unit speed.
If \( \gamma(t_0) \) is a fixed point on the curve \( \gamma \), and \( \gamma(t_1) \), with \( t_1 > t_0 \), is another point, then the vector from \( \gamma(t_0) \) to \( \gamma(t_1) \) approaches the tangent vector to \( \gamma \) at \( \gamma(t_0) \) as the distance between \( t_1 \) and \( t_0 \) is decreased.

The infinite line through \( \gamma(t_0) \) that is parallel to this vector is known as the tangent line to the curve \( \gamma \) at point \( \gamma(t_0) \).

If we disregard the orientation of the tangent vector then we would like to obtain the same result for the tangent line by considering a point \( \gamma(t_1) \) with \( t_1 < t_0 \).
Tangent Vector

Definition 58 (Tangent vector)

Let \( \gamma : I \to \mathbb{R}^n \) be a \( C^1 \)-curve. If \( \gamma'(t) \neq 0 \) for \( t \in I \) then \( \gamma'(t) \) forms the tangent vector at the point \( \gamma(t) \) of \( \gamma \).

- The tangent vector indicates the forward direction of \( \gamma \) relative to increasing parameter values.
- If \( \gamma \) is at unit speed then \( \gamma'(t) \) forms a unit vector.
- A parameterization of the tangent line \( \ell \) that passes through \( \gamma(t) \) is given by
  \[
  \ell(\lambda) = \gamma(t) + \lambda \gamma'(t) \quad \text{with} \quad \lambda \in \mathbb{R}.
  \]
- If
  \[
  \gamma(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}
  \]
  is a curve in \( \mathbb{R}^2 \) then the vector
  \[
  \begin{pmatrix} -y'(t) \\ x'(t) \end{pmatrix}
  \]
  is normal on the tangent line at \( \gamma(t) \).
Frenet Frame for Curves in $\mathbb{R}^3$

Definition 59 (*Frenet frame, Dt.: begleitendes Dreibein*)

Let $\gamma: I \rightarrow \mathbb{R}^3$ be a $C^2$ curve that is regular of order two. Then the *Frenet frame* (aka *moving trihedron*) at $\gamma(t)$ is defined as an orthonormal basis of vectors $T(t), N(t), B(t)$ as follows:

- $T(t) := \frac{\gamma'(t)}{||\gamma'(t)||}$ \hspace{1cm} unit tangent;
- $N(t) := \frac{T'(t)}{||T'(t)||}$ \hspace{1cm} unit (principal) normal;
- $B(t) := T(t) \times N(t)$ \hspace{1cm} unit binormal.

Lemma 60

Let $\gamma: I \rightarrow \mathbb{R}^3$ be a $C^2$ curve that is regular of order two, and define $T, N, B$ as in Def. 59. We get for all $t \in I$:

- $N(t)$ is normal to $T(t)$, and
- $B(t)$ is a unit vector.
Osculating Plane for a Curve in $\mathbb{R}^3$

- Let $\gamma(t_0)$ be a fixed point on $\gamma$, and two other points $\gamma(t_1)$ and $\gamma(t_2)$ that move along $\gamma$.
- Obviously, $\gamma(t_0)$, $\gamma(t_1)$ and $\gamma(t_2)$ define a plane.
- As both $\gamma(t_1)$ and $\gamma(t_2)$ approach $\gamma(t_0)$, the plane determined by them approaches a limiting position.
- This limiting plane is known as the osculating plane to the curve $\gamma$ at point $\gamma(t_0)$.

**Definition 61 (Osculating plane, Dt.: Schmiegeebene)**

Let $\gamma: I \rightarrow \mathbb{R}^3$ be a $C^2$ curve that is regular of order two. The **osculating plane** at the point $\gamma(t)$ is the plane spanned by $T(t)$ and $N(t)$, as defined in Def. 59.

- Of course, the osculating plane contains the tangent line to $\gamma$ at $\gamma(t_0)$.
- The osculating plane at the point $\gamma(t_0)$ contains also $\gamma''(t_0)$.
- Thus, the binormal is the normal vector of the osculating plane.
Curvature of Curves in $\mathbb{R}^3$

- The curvature at a given point of a curve is a measure of how quickly the curve changes direction at that point relative to the speed of the curve.

**Definition 62 (Curvature, Dt.: Krümmung)**

Let $\gamma : I \to \mathbb{R}^3$ be a $C^2$ curve that is regular. The *curvature* $\kappa(t)$ of $\gamma$ at the point $\gamma(t)$ is defined as

$$
\kappa(t) := \frac{\|\gamma'(t) \times \gamma''(t)\|}{\|\gamma'(t)\|^3}.
$$

**Definition 63 (Radius of curvature, Dt.: Krümmungsradius)**

Let $\gamma : I \to \mathbb{R}^3$ be a $C^2$ curve that is regular. If $\kappa(t) > 0$ then the *radius of curvature* $\rho(t)$ at the point $\gamma(t)$ is defined as

$$
\rho(t) := \frac{1}{\kappa(t)}.
$$
Curvature of Curves in $\mathbb{R}^3$ 

- Consider the circle that
  1. passes through $\gamma(t_0)$,
  2. touches the tangent at $\gamma(t_0)$, and that
  3. passes through another point $\gamma(t_1)$.

- Now imagine that the difference between $t_1$ and $t_0$ is decreased.

- In the limit, for $t_1 = t_0$, we get the so-called circle of curvature.

- One can prove that
  - the circle of curvature (aka “osculating circle”) lies in the osculating plane of $\gamma(t_0)$,
  - its radius, the radius of curvature, is given by $\rho(t_0)$, and that
  - its center lies on the ray with direction vector $N(t_0)$ that starts at $\gamma(t_0)$.
Curvature of Curves in $\mathbb{R}^3$: Inflection

Definition 64 (Point of inflection, Dt.: Wendepunkt)

Let $\gamma : I \to \mathbb{R}^3$ be a $C^2$-curve that is regular. If for all $t \in I$ the second derivative $\gamma''$ does not vanish, i.e., if $\gamma''(t) \neq 0$, then a point $\gamma(t)$ for which $\kappa(t) = 0$ holds is called a point of inflection of $\gamma$.

Lemma 65

Let $\gamma : I \to \mathbb{R}^3$ be a $C^2$-curve that is regular such that for all $t \in I$ the second derivative $\gamma''$ does not vanish. Then $\gamma(t)$ is a point of inflection of $\gamma$ if and only if $\gamma'(t)$ and $\gamma''(t)$ are collinear.

Hence, at a point of inflection the radius of curvature is infinite and the circle of curvature degenerates to the tangent.
Lemma 66

Let $\gamma : I \to \mathbb{R}^3$ be a $C^3$-curve at unit speed that is regular of order two. Then the following simplified formula holds:

$$\kappa(t) = \|\gamma''(t)\|$$

Sketch of Proof: Recall that, in general,

$$\kappa(t) = \frac{\|\gamma'(t) \times \gamma''(t)\|}{\|\gamma'(t)\|^3}.$$
Curvature of Curves in $\mathbb{R}^2$

Lemma 67

Let $\gamma : I \to \mathbb{R}^2$ be a $C^2$-curve that is regular, with $\gamma(t) = (x(t), y(t))$. Then $\kappa(t)$ of $\gamma$ at the point $\gamma(t)$ is given as

$$\kappa(t) = \frac{|x'(t)y''(t) - x''(t)y'(t)|}{((x'(t))^2 + (y'(t))^2)^{3/2}}.$$  

Sketch of Proof: Recall that, in general,

$$\kappa(t) = \frac{\|\gamma'(t) \times \gamma''(t)\|}{\|\gamma'(t)\|^3}.$$  

Corollary 68

Let $\gamma : I \to \mathbb{R}^2$ be a $C^2$-curve that is regular, with $\gamma(t) = (t, y(t))$. Then $\kappa(t)$ of $\gamma$ at the point $\gamma(t)$ is given as

$$\kappa(t) = \frac{|y''(t)|}{((1 + (y'(t))^2)^{3/2})}.$$  

Parametric Continuity of a Curve

Consider two curves $\beta : [a, b] \to \mathbb{R}^n$ and $\gamma : [c, d] \to \mathbb{R}^n$.

Suppose that $\beta(b) = \gamma(c) =: p$.

We are interested in checking how “smoothly” $\beta$ and $\gamma$ join at the joint $p$.

Definition 69 ($C^k$-continuous at joint, Dt.: $C^k$-stetiger Übergang)

Let $\beta : [a, b] \to \mathbb{R}^n$ and $\gamma : [c, d] \to \mathbb{R}^n$ be $C^k$-curves. If

$$\beta^{(i)}(b) = \gamma^{(i)}(c) \quad \text{for all } i \in \{0, \ldots, k\}$$

then $\beta$ and $\gamma$ are $C^k$-continuous at joint $p := \beta(b)$.

Of course, one-sided derivatives are to be considered in Def. 69.
Parametric Continuity of a Curve

- $C^0$-continuity implies that the end point of one curve is the start point of the second curve, i.e., they share a common joint.
- $C^1$-continuity implies that the speed does not change at $p$.
- $C^2$-continuity implies that the acceleration does not change at $p$.
- Parametric continuity is important for animations: If an object moves along $\beta$ with constant parametric speed, then there should be no sudden jump once it moves along $\gamma$.

Definition 70 (Curvature continuous, Dt.: krümmungsstetig)

Let $\beta: [a, b] \rightarrow \mathbb{R}^3$ and $\gamma: [c, d] \rightarrow \mathbb{R}^3$ be $C^2$-curves, with $\beta(b) = \gamma(c) =: p$. If the curvatures of $\beta$ and $\gamma$ are equal at $p$ then $\beta$ and $\gamma$ are said to be curvature continuous at $p$.

Caveat

$C^1$-continuity plus curvature continuity need not imply $C^2$-continuity!

- Unfortunately, this important fact is missed frequently, and curvature continuity is often (wrongly) taken as a synonym for $C^2$-continuity . . .
Problems with Parametric Continuity

▶ Note that parametric continuity depends on the particular parameterizations of $\beta$ and $\gamma$.

▶ Consider three collinear points $p$, $q$, and $r$ which define two straight-line segments joined at their common endpoint $q$:

\[
\beta(t) := p + t(q - p), \quad t \in [0, 1] \\
\gamma(t) := q + (t - 1)(r - q), \quad t \in [1, 2]
\]

▶ Of course, $\beta$ and $\gamma$ are $C^0$-continuous at $q$.

▶ However, $\beta'(1) = q - p$ while $\gamma'(1) = r - q$. Thus, in general, $\beta$ and $\gamma$ will not be $C^1$-continuous at $q$.

▶ $C^1$-continuity at $q$ could be achieved by resorting to arc-length parameterizations for $\beta$ and $\gamma$:

\[
\beta(t) := p + \frac{t}{\|q - p\|}(q - p), \quad t \in [0, \|q - p\|] \\
\gamma(t) := q + \frac{t - \|q - p\|}{\|r - q\|}(r - q), \quad t \in [\|q - p\|, \|q - p\| + \|r - q\|]
\]
Geometric Continuity

- $G^0$-continuity equals $C^0$-continuity: The curves $\beta$ and $\gamma$ share a common joint $p$.

**Definition 71 ($G^1$-continuous at joint, Dt.: $G^1$-stetiger Übergang)**

Let $\beta : [a, b] \rightarrow \mathbb{R}^n$ and $\gamma : [c, d] \rightarrow \mathbb{R}^n$ be $C^1$-curves, with $\beta(b) = \gamma(c) =: p$. If

$$0 \neq \beta'(b) = \lambda \cdot \gamma'(c) \quad \text{for some } \lambda \in \mathbb{R}^+$$

then $\beta$ and $\gamma$ are $G^1$-continuous at joint $p$.

- $G^1$-continuity means that $\beta$ and $\gamma$ share the tangent line at $p$.
- Higher-order geometric continuities are a bit tricky to define formally for $k \geq 2$.
- $G^2$-continuity means that $\beta$ and $\gamma$ share the tangent line and also the same center of curvature at $p$.
- In general, $G^k$-continuity exists at $p$ if $\beta$ and $\gamma$ can be reparameterized such that they join with $C^k$-continuity at $p$.
- $C^k$-continuity implies $G^k$-continuity.
- Note: Reflections on a surface (e.g., a car body) will not appear smooth unless $G^2$-continuity is achieved.
**Parametric Surface in** $\mathbb{R}^3$

**Definition 72 (Parametric surface)**

Let $\Omega \subseteq \mathbb{R}^2$. A continuous mapping $\alpha : \Omega \rightarrow \mathbb{R}^3$ is called a *parameterization* of $\alpha(\Omega)$, and $\alpha(\Omega)$ is called the (parametric) *surface* parameterized by $\alpha$.

- For instance, every point on the surface of Earth can be described by the geographic coordinates longitude and latitude.
- Note that parameterizations of a surface (regarded as a set $S \subseteq \mathbb{R}^3$) need not be unique: two different parameterizations $\alpha$ and $\beta$ may exist such that $S = \alpha(\Omega_1) = \beta(\Omega_2)$.
- For simplicity, we will not distinguish between a surface and one of its parameterizations if the meaning is clear.
Sample Parametric Surface: Frustum of a Paraboloid

\[ \alpha: [0, 1] \times [0, 2\pi] \rightarrow \mathbb{R}^3 \]

\[ \alpha(u, v) := \begin{pmatrix} u \cos v \\ u \sin v \\ 2u^2 \end{pmatrix} \]
Sample Parametric Surface: Torus

\[ \alpha : [0, 2\pi]^2 \rightarrow \mathbb{R}^3 \]

\[ \alpha(u, v) := \begin{pmatrix} (2 + \cos v) \cos u \\ (2 + \cos v) \sin u \\ \sin v \end{pmatrix} \]
Basic Definitions for Parametric Surfaces

Definition 73 (Regular parameterization, Dt.: reguläre (od. zulässige) Param.)

Let \( \Omega \subseteq \mathbb{R}^2 \). A continuous mapping \( \alpha : \Omega \rightarrow \mathbb{R}^3 \) in the variables \( u \) and \( v \) is called a regular parameterization of \( \alpha(\Omega) \) if

1. \( \alpha \) is differentiable on \( \Omega \),
2. \( \frac{\partial \alpha}{\partial u}(u_0, v_0) \) and \( \frac{\partial \alpha}{\partial v}(u_0, v_0) \) are linearly independent for all \( (u_0, v_0) \) in \( \Omega \).

▶ Note that \( \frac{\partial \alpha}{\partial u}(u_0, v_0) \) and \( \frac{\partial \alpha}{\partial v}(u_0, v_0) \) are linearly independent if and only if

\[
\frac{\partial \alpha}{\partial u}(u_0, v_0) \times \frac{\partial \alpha}{\partial v}(u_0, v_0) \neq 0.
\]

Definition 74 (Singular point, Dt.: singulärer Punkt)

Let \( \Omega \subseteq \mathbb{R}^2 \). A point \( (u_0, v_0) \in \Omega \) is a singular point of a differentiable parameterization \( \alpha : \Omega \rightarrow \mathbb{R}^3 \) if \( \frac{\partial \alpha}{\partial u}(u_0, v_0) \) and \( \frac{\partial \alpha}{\partial v}(u_0, v_0) \) are linearly dependent.
Curves on Surfaces

- Consider a planar parametric $C^1$ curve with coordinates $(u(t), v(t))$, for $t \in I \subseteq \mathbb{R}$, in the parametric domain $\Omega$ of a parametric surface $\alpha : \Omega \to \mathbb{R}^3$.

- Then $\gamma : I \to \mathbb{R}^3$ with $\gamma(t) := \alpha(u(t), v(t))$ is a parametric curve on the surface $\alpha(\Omega)$.

- Suppose that $\alpha$ is differentiable on $\Omega$. Differentiating $\gamma(t)$ (by means of a multi-dimensional analogue of the Chain Rule 26) with respect to $t$ yields the tangent vector of the curve on the surface:

\[
\gamma'(t) = \frac{\partial \alpha}{\partial u}(u(t), v(t)) \cdot u'(t) + \frac{\partial \alpha}{\partial v}(u(t), v(t)) \cdot v'(t)
\]

- Hence, the tangent vector at $\gamma(t)$ is a linear combination of

\[
\frac{\partial \alpha}{\partial u}(u(t), v(t)) \quad \text{and} \quad \frac{\partial \alpha}{\partial v}(u(t), v(t)).
\]
Curves on Surfaces

- Suppose that $\Omega = [u_{\text{min}}, u_{\text{max}}] \times [v_{\text{min}}, v_{\text{max}}]$

- If we fix $v := v_0 \in [v_{\text{min}}, v_{\text{max}}]$ and let $u$ vary, then $\alpha(u, v_0)$ depends on one parameter; it is called an *isoparametric curve* or, more specifically, the *$u$-parameter curve*.

- Likewise, we can fix $u := u_0 \in [u_{\text{min}}, u_{\text{max}}]$ and let $v$ vary to obtain the *$v$-parameter curve* $\alpha(u_0, v)$.

- Tangent vectors for the $u$-parameter and $v$-parameter curves are computed by partial derivatives of $\alpha$ with respect to $u$ and $v$, respectively:

\[
\frac{\partial \alpha}{\partial u}(u, v) \quad \text{for } v = v_0
\]

\[
\frac{\partial \alpha}{\partial v}(u, v) \quad \text{for } u = u_0
\]
Curves on Surfaces

The parameterization of the unit sphere is given by

\[ \alpha(u, v) := \begin{pmatrix} \cos u \cdot \cos v \\ \sin u \cdot \cos v \\ \sin v \end{pmatrix} \]  
with \((u, v) \in [0, 2\pi] \times [-\frac{\pi}{2}, \frac{\pi}{2}]\).

We get

\[ \frac{\partial \alpha}{\partial u}(u, v_0) = \begin{pmatrix} -\sin u \cdot \cos v_0 \\ \cos u \cdot \cos v_0 \\ 0 \end{pmatrix} \]

and

\[ \frac{\partial \alpha}{\partial v}(u_0, v) = \begin{pmatrix} -\cos u_0 \cdot \sin v \\ -\sin u_0 \cdot \sin v \\ \cos v \end{pmatrix}. \]

Note that \( \frac{\partial \alpha}{\partial u}(u_0, v_0) \perp \frac{\partial \alpha}{\partial v}(u_0, v_0) \).

Also, note that \( \frac{\partial \alpha}{\partial u}(u, v_0) \) vanishes for \( v_0 := \pm \frac{\pi}{2} \). Hence, the north and south poles are singular points of this parameterization.
Tangent Plane and Normal Vector

Recall that the tangent vector at a point \( \gamma(t) \) of a parametric curve on a surface \( \alpha(\Omega) \) is given by partial derivatives of \( \alpha \). This motivates the following definition.

**Definition 75 (Tangent plane, Dt.: Tangentialebene)**

Consider a regular parameterization \( \alpha : \Omega \to \mathbb{R}^3 \) of a surface \( S \). For \( (u, v) \in \Omega \), the **tangent plane** \( \varepsilon(u, v) \) of \( S \) at \( \alpha(u, v) \) is the plane through \( \alpha(u, v) \) that is spanned by the vectors

\[
\frac{\partial \alpha}{\partial u}(u, v) \quad \text{and} \quad \frac{\partial \alpha}{\partial v}(u, v).
\]

**Definition 76 (Normal vector, Dt.: Normalvektor)**

Consider a regular parameterization \( \alpha : \Omega \to \mathbb{R}^3 \) of a surface \( S \). For \( (u, v) \in \Omega \), the **normal vector** \( N(u, v) \) of \( S \) at \( \alpha(u, v) \) is given by

\[
N(u, v) := \frac{\partial \alpha}{\partial u}(u, v) \times \frac{\partial \alpha}{\partial v}(u, v).
\]
Bézier Curves and Surfaces
Bernstein Basis Polynomials
Bézier Curves
Bézier Surfaces
Bernstein Basis Polynomials

Definition 77 (*Bernstein basis polynomials*)

The $n + 1$ *Bernstein basis polynomials* of degree $n$, for $n \in \mathbb{N}_0$, are defined as

$$B_{k,n}(x) := \binom{n}{k} x^k (1 - x)^{n-k} \quad \text{for} \quad k \in \{0, 1, \ldots, n\}.$$

- We use the convention $0^0 := 1$.
- For convenience purposes, we define $B_{k,n}(x) := 0$ for $k < 0$ or $k > n$.
- $B_{0,0}(x) = 1$.
- $B_{0,1}(x) = 1 - x$ and $B_{1,1}(x) = x$.
- $B_{0,2}(x) = (1 - x)^2$ and $B_{1,2}(x) = 2x(1 - x)$ and $B_{2,2}(x) = x^2$.

- Introduced by Sergei N. Bernstein in 1911 for a constructive proof of Weierstrass’ Approximation Theorem 181.
All Bernstein basis polynomials of degree $n = 0$ over the interval $[0, 1]$:

1
Bernstein Basis Polynomials

- All Bernstein basis polynomials of degree $n = 1$ over the interval $[0, 1]$:

  \[ 1 - x \quad \quad \quad \quad \quad \quad x \]
Bernstein Basis Polynomials

- All Bernstein basis polynomials of degree $n = 2$ over the interval $[0, 1]$:

\[(1 - x)^2 \quad 2x(1 - x) \quad x^2\]
All Bernstein basis polynomials of degree $n = 3$ over the interval $[0, 1]$: 

$(1 - x)^3 \quad 3x(1 - x)^2 \quad 3x^2(1 - x) \quad x^3$
Bernstein Basis Polynomials

- All Bernstein basis polynomials of degree $n = 4$ over the interval $[0, 1]$: 

\[
(1 - x)^4 \quad 4x(1 - x)^3 \quad 6x^2(1 - x)^2 \quad 4x^3(1 - x) \quad x^4
\]
Bernstein Basis Polynomials

- All Bernstein basis polynomials of degree $n = 5$ over the interval $[0, 1]$: 

\[
(1 - x)^5 \quad 5x(1 - x)^4 \quad 10x^2(1 - x)^3 \quad 10x^3(1 - x)^2 \quad 5x^4(1 - x) \quad x^5
\]
Recursion Formula for Bernstein Basis Polynomials

Lemma 78

For all \( n \in \mathbb{N} \) and \( k \in \mathbb{N}_0 \) with \( k \leq n \), the Bernstein basis polynomial \( B_{k,n}(x) \) of degree \( n \) can be written as the sum of two basis polynomials of degree \( n - 1 \):

\[
B_{k,n}(x) = x \cdot B_{k-1,n-1}(x) + (1 - x) \cdot B_{k,n-1}(x)
\]

Proof: Let \( n \in \mathbb{N} \) and \( k \in \mathbb{N}_0 \) with \( k \leq n \) be arbitrary but fixed, and recall that

\[
B_{k,n}(x) \overset{\text{Def. 77}}{=} \binom{n}{k} x^k (1 - x)^{n-k}
\]

and

\[
\binom{n}{k} \overset{\text{Thm. 4}}{=} \binom{n - 1}{k - 1} + \binom{n - 1}{k}.
\]

We get

\[
B_{k,n}(x) = \binom{n - 1}{k - 1} x^k (1 - x)^{n-k} + \binom{n - 1}{k} x^k (1 - x)^{n-k}
\]

\[
= x \binom{n - 1}{k - 1} x^{k-1} (1 - x)^{(n-1)-(k-1)} + (1 - x) \binom{n - 1}{k} x^k (1 - x)^{(n-1)-k}
\]

\[
= x \cdot B_{k-1,n-1}(x) + (1 - x) \cdot B_{k,n-1}(x).
\]
Properties of Bernstein Basis Polynomials

Lemma 79
For all $n \in \mathbb{N}$ and $k \in \mathbb{N}_0$ with $k \leq n$, the Bernstein basis polynomial $B_{k,n-1}$ can be written as linear combination of Bernstein basis polynomials of degree $n$:

$$B_{k,n-1}(x) = \frac{n-k}{n} B_{k,n}(x) + \frac{k+1}{n} B_{k+1,n}(x).$$

Lemma 80
For all $n, k \in \mathbb{N}_0$ with $k \leq n$, the Bernstein basis polynomial $B_{k,n}$ is non-negative over the unit interval:

$$B_{k,n}(x) \geq 0 \text{ for all } x \in [0, 1].$$

Proof: Recall the definition of the Bernstein basis polynomials:

$$B_{k,n}(x) \overset{\text{Def. 77}}{=} \binom{n}{k} x^k (1-x)^{n-k} \geq 0 \text{ for all } x \in [0, 1].$$
Properties of Bernstein Basis Polynomials

Lemma 81 (Partition of unity, Dt.: Zerlegung der Eins)

For all $n \in \mathbb{N}_0$, the $n + 1$ Bernstein basis polynomials of degree $n$ form a partition of unity, i.e., they sum up to one:

$$\sum_{k=0}^{n} B_{k,n}(x) = 1 \quad \text{for all } x \in [0, 1].$$

Proof: Trivial for $n = 0$. Now recall the Binomial Theorem 5, for $a, b \in \mathbb{R}$ and $n \in \mathbb{N}$:

$$(a + b)^n = \sum_{k=0}^{n} \binom{n}{k} a^k b^{n-k}$$

Then the claim is an immediate consequence by setting $a := x$ and $b := 1 - x$:

$$1 = (x + (1 - x))^n = \sum_{k=0}^{n} \binom{n}{k} x^k (1 - x)^{n-k} = \sum_{k=0}^{n} B_{k,n}(x).$$
Properties of Bernstein Basis Polynomials

Lemma 82

For all $n \in \mathbb{N}_0$ and any set of $n + 1$ points in $\mathbb{R}^2$ with position vectors $p_0, p_1, p_2, \ldots, p_n$, the term

$$p_0 B_{0,n}(t) + p_1 B_{1,n}(t) + \cdots + p_n B_{n,n}(t)$$

forms a convex combination of these points for all $t \in [0, 1]$.

Proof: This is an immediate consequence of Lem. 80 and Lem. 81.

Corollary 83 (Convex hull property)

For all $n \in \mathbb{N}_0$ and any set of $n + 1$ points in $\mathbb{R}^2$ with position vectors $p_0, p_1, p_2, \ldots, p_n$, the point

$$p_0 B_{0,n}(t) + p_1 B_{1,n}(t) + \cdots + p_n B_{n,n}(t)$$

lies within $CH(\{p_0, p_1, p_2, \ldots, p_n\})$ for all $t \in [0, 1]$.

Proof: Recall that $CH(\{p_0, p_1, p_2, \ldots, p_n\})$ equals the set of all convex combinations of $p_0, p_1, p_2, \ldots, p_n$. 

\end{document}
Derivatives of Bernstein Basis Polynomials

Lemma 84

For \( n, k \in \mathbb{N}_0 \) and \( i \in \mathbb{N} \) with \( i \leq n \), the \( i \)-th derivative of \( B_{k,n}(x) \) can be written as a linear combination of Bernstein basis polynomials of degree \( n - i \):

\[
B_{k,n}^{(i)}(x) = \frac{n!}{(n-i)!} \sum_{j=0}^{i} (-1)^{i-j} \binom{i}{j} B_{k-j,n-i}(x)
\]

Corollary 85

For \( n, k \in \mathbb{N}_0 \), the first and second derivative of \( B_{k,n}(x) \) are given as follows:

\[
B_{k,n}'(x) = n \left( B_{k-1,n-1}(x) - B_{k,n-1}(x) \right)
\]
\[
B_{k,n}''(x) = n(n-1) \left( B_{k-2,n-2}(x) - 2B_{k-1,n-2}(x) + B_{k,n-2}(x) \right)
\]
Lemma 86

The \( n + 1 \) Bernstein basis polynomials \( B_{0,n}, B_{1,n}, \ldots, B_{n,n} \) are linearly independent, for all \( n \in \mathbb{N}_0 \).

Proof: We do a proof by induction.

I.B.: The claim is obviously true for \( n = 0 \) and \( n = 1 \).

I.H.: Suppose that the claim is true for an arbitrary but fixed \( n - 1 \in \mathbb{N}_0 \), i.e., that
\[
\sum_{k=0}^{n-1} \lambda_k B_{k,n-1}(x) = 0 \quad \text{implies} \quad \lambda_0 = \lambda_1 = \ldots = \lambda_{n-1} = 0.
\]

I.S.: Suppose that \( \sum_{k=0}^{n} \lambda_k B_{k,n}(x) = 0 \) for some \( \lambda_0, \lambda_1, \ldots, \lambda_n \in \mathbb{R} \). Then we get

\[
0 = \sum_{k=0}^{n} \lambda_k B'_{k,n}(x) = 84 \sum_{k=0}^{n} \lambda_k \cdot n \cdot (B_{k-1,n-1}(x) - B_{k,n-1}(x))
\]

\[
= n \left( \sum_{k=0}^{n-1} \lambda_{k+1} B_{k,n-1}(x) - \sum_{k=0}^{n-1} \lambda_k B_{k,n-1}(x) \right)
\]

\[
= n \sum_{k=0}^{n-1} \mu_k B_{k,n-1}(x) \quad \text{with} \quad \mu_k := \lambda_{k+1} - \lambda_k \quad \text{for} \quad 0 \leq k \leq n - 1.
\]

The I.H. implies \( \mu_0 = \mu_1 = \cdots \mu_{n-1} = 0 \) and, thus, \( \lambda_0 = \lambda_1 = \cdots \lambda_n \), which implies \( \lambda_0 = \lambda_1 = \cdots \lambda_n = 0 \). (Recall Partition of Unity, Lem. 81.)
Lemma 87
For all $n, i \in \mathbb{N}_0$, with $i \leq n$, we have

$$x^i = \sum_{k=i}^{n} \binom{k}{i} \binom{n}{i} B_{k,n}(x).$$

Theorem 88
The Bernstein basis polynomials of degree $n$ form a basis of the vector space of polynomials of degree up to $n$ over $\mathbb{R}$, for all $n \in \mathbb{N}_0$.

Proof: This is an immediate consequence of either Lem. 86 or Lem. 87.
Bézier Curves

- Discovered in the late 1950s by Paul de Casteljau at Citroën and in the early 1960s by Pierre E. Bézier at Renault, and first published by Bézier in 1962. (Citroën allowed de Casteljau to publish his results in 1974 for the first time.)
- The idea is to specify a curve by using points which control its shape: control points. The figure shows a Bézier curve of degree 10 with 11 control points.
- Bézier curves formed the foundations of Citroën’s UNISURF CAD/CAM system.
- TrueType fonts use font descriptions made of composite quadratic Bézier curves; PostScript, METAFONT, and SVG use composite Béziers made of cubic Bézier curves.
Bézier Curves

Definition 89 (Bézier curve)

Suppose that we are given \( n + 1 \) control points with position vectors \( p_0, p_1, \ldots, p_n \) in the plane \( \mathbb{R}^2 \), for \( n \in \mathbb{N} \). The Bézier curve \( B : [0, 1] \rightarrow \mathbb{R}^2 \) defined by \( p_0, p_1, \ldots, p_n \) is given by

\[
B(t) := \sum_{i=0}^{n} B_{i,n}(t) p_i \quad \text{for } t \in [0, 1],
\]

where \( B_{i,n}(t) = \binom{n}{i} t^i (1 - t)^{n-i} \) is the \( i \)-th Bernstein basis polynomial of degree \( n \).

- The weighted average of all control points gives a location on the curve relative to the parameter \( t \). The weights are given by the coefficients \( B_{i,n} \).
- The polygonal chain \( p_0, p_1, p_3, \ldots, p_{n-1}, p_n \) is called the control polygon, and its individual segments are referred to as legs.
- Although not explicitly required, it is generally assumed that the control points are distinct, except for possibly \( p_0 \) and \( p_n \) being identical.
- Of course, the same definition and the subsequent math can be applied to \( p_0, p_1, \ldots, p_n \in \mathbb{R}^d \) for some \( d \in \mathbb{N} \) with \( d > 2 \).
Properties of Bézier Curves

Lemma 90

A Bézier curve defined by \( n + 1 \) control points is (coordinate-wise) a polynomial of degree \( n \).

**Proof:** It is the sum of \( n + 1 \) Bernstein basis polynomials of degree \( n \). □

Lemma 91

A Bézier curve starts in the first control point and ends in the last control point.

**Proof:** Recall that

\[
B_{i,n}(0) = \begin{cases} 
1 & \text{for } i = 0, \\
0 & \text{for } i > 0.
\end{cases}
\]

Hence, \( B(0) = B_{0,n}(0)p_0 = p_0 \). Similarly for \( B_{i,n}(1) \) and \( B(1) \). □
Properties of Bézier Curves

Lemma 92 (Convex hull property)
A Bézier curve lies completely inside the convex hull of its control points.

Proof: This is nothing but a re-formulation of Cor. 83.

Lemma 93 (Variation diminishing property)
If a straight line intersects the control polygon of a Bézier curve \( k \) times then it intersects the actual Bézier curve at most \( k \) times.

Lemma 94 (Symmetry property)
The following identity holds:

\[
\sum_{i=0}^{n} B_{i,n}(t)p_i = \sum_{i=0}^{n} B_{i,n}(1-t)p_{n-i}.
\]
Properties of Bézier Curves

Lemma 95 (Affine invariance)

Any Bézier representation is affinely invariant, i.e., given any affine map \( \pi \), the image curve \( \pi(B) \) of a Bézier curve \( B: [0, 1] \rightarrow \mathbb{R}^2 \) with control points \( p_0, p_1, \ldots, p_n \) has the control points \( \pi(p_0), \pi(p_1), \ldots, \pi(p_n) \) over \([0, 1]\).

Proof: Consider an affine map \( \pi: \mathbb{R}^2 \rightarrow \mathbb{R}^2 \). Hence, \( \pi(x) = A \cdot x + v \), for some \( 2 \times 2 \) matrix \( A \), and \( x, v \in \mathbb{R}^2 \). We get

\[
\pi(B(t)) = \pi \left( \sum_{i=0}^{n} B_{i,n}(t)p_i \right) = A \cdot \left( \sum_{i=0}^{n} B_{i,n}(t)p_i \right) + v
\]

\[
= \sum_{i=0}^{n} B_{i,n}(t)A \cdot p_i + \sum_{i=0}^{n} B_{i,n}(t)v = \sum_{i=0}^{n} B_{i,n}(t)(A \cdot p_i + v)
\]

\[
= \sum_{i=0}^{n} B_{i,n}(t)\pi(p_i).
\]

\( \square \)
Modifying a Control Point

▶ Suppose that we shift one control point $p_j$ to a new location $p_j + v$.

▶ The corresponding Bézier curve $\mathcal{B}$ is transformed to $\mathcal{B}^*$ as follows:

$$\mathcal{B}^*(t) = \left( \sum_{i=0}^{j-1} B_{i,n}(t) p_i \right) + B_{j,n}(t)(p_j + v) + \left( \sum_{i=j+1}^{n} B_{i,n}(t) p_i \right) =$$

$$= \sum_{i=0}^{n} B_{i,n}(t) p_i + B_{j,n}(t)v = \mathcal{B}(t) + B_{j,n}(t)v$$

▶ Now recall that $B_{j,n}(t) \neq 0$ for all $t$ with $0 < t < 1$. Hence, a modification of just one control point results in a global change of the entire Bézier curve.
For $0 < t < 1$ we can locate a point $q$ on a line segment $\overline{pr}$ such that it divides the line segment into portions of relative length $t$ and $1 - t$, i.e., according to the ratio $t : 1 - t$.

Of course, $q$ is given by

$$q = (1 - t) \cdot p + t \cdot r.$$

Similarly, we can compute a point on a Bézier curve such that the curve is split into portions of relative length $t$ and $1 - t$.

On every leg $\overline{p_{j-1}p_j}$ of the control polygon we compute a point $p_{1j}$ which divides it according to the ratio $t : 1 - t$.

In total we get $n$ new points which define a new polygonal chain with $n - 1$ legs.

This new polygonal chain can be used to construct another polygonal chain with $n - 2$ legs.

This process can be repeated $n$ times, i.e., until we are left with a single point.

It was proved by de Casteljau that this point corresponds to the point $B(t)$ sought.
De Casteljau’s Algorithm

- Sample run of de Casteljau’s algorithm for $t := 1/4$.
- The points are indexed in the form $i, j$, where $i$ denotes the number of the iteration and $j + 1$ denotes the leg defined by the control points $p_{i,j}$ and $p_{i,j+1}$.
De Casteljau’s Algorithm

- Sample run of de Casteljau’s algorithm for $t := 1/4$.
- The points are indexed in the form $i, j$, where $i$ denotes the number of the iteration and $j + 1$ denotes the leg defined by the control points $p_{i,j}$ and $p_{i,j+1}$.
De Casteljau’s Algorithm

- Sample run of de Casteljau’s algorithm for $t := 1/4$.
- The points are indexed in the form $i, j$, where $i$ denotes the number of the iteration and $j + 1$ denotes the leg defined by the control points $p_{i,j}$ and $p_{i,j+1}$.
De Casteljau’s Algorithm

- Sample run of de Casteljau’s algorithm for \( t := \frac{1}{4} \).
- The points are indexed in the form \( i, j \), where \( i \) denotes the number of the iteration and \( j + 1 \) denotes the leg defined by the control points \( p_{i,j} \) and \( p_{i,j+1} \).
De Casteljau’s Algorithm

- Sample run of de Casteljau’s algorithm for $t := \frac{1}{4}$.
- The points are indexed in the form $i, j$, where $i$ denotes the number of the iteration and $j + 1$ denotes the leg defined by the control points $p_{i,j}$ and $p_{i,j+1}$.
De Casteljau’s Algorithm

- Sample run of de Casteljau’s algorithm for $t := 1/4$.
- The points are indexed in the form $i, j$, where $i$ denotes the number of the iteration and $j + 1$ denotes the leg defined by the control points $p_{i,j}$ and $p_{i,j+1}$.

```
P0 =: p00
P1 =: p01
P2 =: p02
P3 =: p03
P4 =: p04
P5 =: p05
```

```
P00
P01
P02
P03
P04
P05
```

```
P10
P11
P12
P13
P14
```

```
P20
P21
P22
P23
```

```
P30
P31
P32
P33
```

```
P40
P41
P42
P43
P44
```

```
P5
```

```
P15
P14
```

© M. Held (Univ. Salzburg)  Geometric Modeling (WS 2018/19) 114/289
De Casteljau’s Algorithm

- Sample run of de Casteljau’s algorithm for $t := 1/4$.
- The points are indexed in the form $i, j$, where $i$ denotes the number of the iteration and $j + 1$ denotes the leg defined by the control points $p_{i,j}$ and $p_{i,j+1}$.
De Casteljau’s Algorithm

- Numerically very stable, since only convex combinations are used!

```c
/** Evaluates a Bezier curve at parameter t by applying de Casteljau’s algorithm
 * @param p: array of n+1 control points
 * @param n: the degree of the Bezier curve
 * @param t: the parameter
 * @return the evaluation result
 */
point DeCasteljau(point *p, int n, double t)
{
    for (int i = 1; i <= n; ++i)
        for (int j = 0; j <= n-i; ++j)
            p[j] = (1-t) * p[j] + t * p[j+1];

    return p[0];
}
```

\[
\begin{align*}
    p_0 &= p_{00} \\
    p_1 &= p_{01} \\
    p_2 &= p_{02} \\
    p_3 &= p_{03} \\
    p_4 &= p_{04} \\
    p_5 &= p_{05}
\end{align*}
\]
De Casteljau’s Algorithm: Correctness

- The point \( p_{10} \) is obtained as
  \[
  p_{10} = (1 - t) \cdot p_{00} + t \cdot p_{01}.
  \]

- Hence, the contribution of \( p_{01} \) to \( p_{10} \) is \( t \cdot p_{01} \).

- Since \( p_{20} \) is obtained as
  \[
  p_{20} = (1 - t) \cdot p_{10} + t \cdot p_{11},
  \]
  the contribution of \( p_{01} \) to \( p_{20} \) via \( p_{10} \) is
  \[
  (1 - t)p_{10} = t(1 - t) \cdot p_{01}.
  \]

- Similarly, the contribution of \( p_{01} \) to \( p_{20} \) via \( p_{11} \) is
  \[
  t(1 - t) \cdot p_{01}.
  \]
De Casteljau’s Algorithm: Correctness

▶ Each path from \(p_{0i}\) to \(p_{n0}\) is constrained to a diamond shape anchored at \(p_{0i}\) and \(p_{n0}\).

▶ An inductive argument shows that each path from \(p_{0i}\) to \(p_{n0}\) consists of \(i\) north-east arrows, i.e., multiplications by \(t\), and \(n - i\) south-east arrows, i.e., multiplications by \((1 - t)\).

▶ Thus, the contribution of \(p_{0i}\) to \(p_{n0}\) is

\[
t^i (1 - t)^{n-i} \cdot p_{0i},
\]

along each path from \(p_{0i}\) to \(p_{n0}\).
De Casteljau’s Algorithm: Correctness

▶ How many different paths exist from $p_{oi}$ to $p_{n0}$? This is equivalent to asking “how many different ways exist to place $i$ north-east arrows on a total of $n$ possible positions?”, and the answer is given by $\binom{n}{i}$.

▶ Thus, the total contribution of $p_{oi}$ to $p_{n0}$, along all paths from $p_{oi}$ to $p_{n0}$, is

$$\binom{n}{i} \cdot t^i (1 - t)^{n-i} p_{oi}.$$ 

This is, however, precisely the weight of $p_{oi}$ in the definition of a Bézier curve (Def. 89).
Evaluation of a Bézier Curve Using Horner’s Scheme

- Horner’s scheme can also be used for evaluating a Bézier curve.

- After rewriting $B(t)$ as

$$B(t) = \sum_{i=0}^{n} B_{i,n}(t) p_i = \sum_{i=0}^{n} \binom{n}{i} t^i (1 - t)^{n-i} p_i$$

$$= (1 - t)^n \left( \sum_{i=0}^{n} \binom{n}{i} \left( \frac{t}{1 - t} \right)^i p_i \right),$$

one evaluates the sum for the value $\frac{t}{1 - t}$, and then multiplies by $(1 - t)^n$.

- This method becomes unstable if $t$ is close to one. In this case, one can resort to Lem. 94, which gives the identity

$$B(t) = t^n \left( \sum_{i=0}^{n} \binom{n}{i} \left( \frac{1 - t}{t} \right)^i p_{n-i} \right).$$

- In any case, Horner’s scheme tends to be faster but numerically more problematic than de Casteljau’s algorithm.
Bernstein Polynomials and Polar Forms

Theorem 96

Let \( n, d \in \mathbb{N} \). For every polynomial function \( F: \mathbb{R} \rightarrow \mathbb{R}^d \) of degree at most \( n \) there exists exactly one symmetric and multi-affine function \( f: \mathbb{R}^n \rightarrow \mathbb{R}^d \), which is called the polar form (aka “blossom”, Dt.: Polarform) of \( F \), such that

1. for all \( i \in \{1, 2, \ldots, n\} \), all \( x_1, x_2, \ldots, x_n \in \mathbb{R} \), all \( k \in \mathbb{N} \), all \( y_1, y_2, \ldots, y_k \in \mathbb{R} \) and all \( \alpha_1, \alpha_2, \ldots, \alpha_k \in \mathbb{R} \) with \( \sum_{j=1}^{k} \alpha_j = 1 \)

\[
f(x_1, \ldots, x_{i-1}, \sum_{j=1}^{k} \alpha_j y_j, x_{i+1}, \ldots, x_n) = \sum_{j=1}^{k} \alpha_j f(x_1, \ldots, x_{i-1}, y_j, x_{i+1}, \ldots, x_n)
\]

2. for all \( i, j \in \{1, 2, \ldots, n\} \)

\[
f(x_1, x_2, \ldots, x_{i-1}, x_i, x_{i+1}, \ldots, x_{j-1}, x_j, x_{j+1}, \ldots, x_n) = f(x_1, x_2, \ldots, x_{i-1}, x_j, x_{i+1}, \ldots, x_{j-1}, x_i, x_{j+1}, \ldots, x_n),
\]

3. for all \( x \in \mathbb{R} \)

\[
F(x) = f(x, x, \ldots, x), \quad \text{i.e., } F \text{ is the diagonal of } f.
\]
Bernstein Polynomials and Polar Forms

Lemma 97

Let \( n \in \mathbb{N} \) and \( a_0, a_1, \ldots, a_n \in \mathbb{R} \), and \( F(x) := \sum_{i=0}^{n} a_i x^i \). Then \( f : \mathbb{R}^n \to \mathbb{R} \) with

\[
  f(x_1, x_2, \ldots, x_n) := \sum_{i=0}^{n} a_i \frac{1}{\binom{n}{i}} \left( \sum_{\substack{I \subseteq \{1, \ldots, n\} \\vert \, \vert I \vert = i}} \prod_{j \in I} x_j \right)
\]

is the polar form of \( F \).

<table>
<thead>
<tr>
<th>( n )</th>
<th>( a_i )</th>
<th>( F(x) )</th>
<th>( f(x_1, \ldots, x_n) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( a_0 := 1, a_1 := 0 ) &lt;br&gt; ( a_0 := 0, a_1 := 1 )</td>
<td>1 &lt;br&gt; ( x )</td>
<td>1 &lt;br&gt; ( x_1 )</td>
</tr>
<tr>
<td>2</td>
<td>( a_0 := 1, a_1 := 0, a_2 := 0 ) &lt;br&gt; ( a_0 := 0, a_1 := 1, a_2 := 0 ) &lt;br&gt; ( a_0 := 0, a_1 := 0, a_2 := 1 )</td>
<td>1 &lt;br&gt; ( x ) &lt;br&gt; ( x^2 )</td>
<td>1 &lt;br&gt; ( \frac{1}{2}(x_1 + x_2) ) &lt;br&gt; ( x_1 x_2 )</td>
</tr>
<tr>
<td>3</td>
<td>( a_0 := 1, a_1 := 0, a_2 := 0, a_3 := 0 ) &lt;br&gt; ( a_0 := 0, a_1 := 1, a_2 := 0, a_3 := 0 ) &lt;br&gt; ( a_0 := 0, a_1 := 0, a_2 := 1, a_3 := 0 ) &lt;br&gt; ( a_0 := 0, a_1 := 0, a_2 := 0, a_3 := 1 )</td>
<td>1 &lt;br&gt; ( x ) &lt;br&gt; ( x^2 ) &lt;br&gt; ( x^3 )</td>
<td>1 &lt;br&gt; ( \frac{1}{3}(x_1 + x_2 + x_3) ) &lt;br&gt; ( \frac{1}{3}(x_1 x_2 + x_1 x_3 + x_2 x_3) ) &lt;br&gt; ( x_1 x_2 x_3 )</td>
</tr>
</tbody>
</table>
Bernstein Polynomials and Polar Forms

Let \( F(x) := \begin{pmatrix} x \\ \frac{1}{2} x^2 \end{pmatrix} \). Hence \( f(x_1, x_2) = \begin{pmatrix} \frac{1}{2} (x_1 + x_2) \\ \frac{1}{2} x_1 x_2 \end{pmatrix} \), and we get

\[
\begin{align*}
  f(0, 0) &= \begin{pmatrix} 0 \\ 0 \end{pmatrix} &
  f(0, 1) &= \begin{pmatrix} \frac{1}{2} \\ 0 \end{pmatrix} = f(1, 0) &
  f(1, 1) &= \begin{pmatrix} 1 \\ \frac{1}{2} \end{pmatrix}.
\end{align*}
\]

Furthermore, \( F(t) = f(t, t) \), with

\[
\begin{align*}
  f(t, t) &= f((1 - t) \cdot 0 + t \cdot 1, t) = (1 - t)f(0, t) + tf(1, t) \\
  &= (1 - t)[(1 - t) \cdot f(0, 0) + t \cdot f(0, 1))] + t[(1 - t) \cdot f(1, 0) + t \cdot f(1, 1))] \\
  &= (1 - t)^2 f(0, 0) + 2t(1 - t)f(0, 1) + t^2 f(1, 1) \\
  &= B_{0,2}(t)f(0, 0) + B_{1,2}(t)f(0, 1) + B_{2,2}(t)f(1, 1).
\end{align*}
\]

Hence, there is a close connection between the polar form and the Bernstein polynomials: \( f(0, 0), f(0, 1), f(1, 1) \) form the coefficients (i.e., control points) of \( F \) relative to the Bernstein basis.
Bernstein Polynomials and Polar Forms

Lemma 98

Every polynomial can be expressed in Bezier form. That is, for every polynomial \( P : \mathbb{R} \to \mathbb{R}^2 \) of degree \( n \) there exist control points \( p_0, p_1, \ldots, p_n \in \mathbb{R}^2 \) such that the Bezier curve defined by them matches \( P|_{[0,1]} \).

Sketch of Proof: Let \( f \) be the polarform of \( P \), and let

\[
p_k := f(0, \ldots, 0, 1, \ldots, 1) \quad \text{for } k = 0, 1, \ldots, n.
\]

Polar forms are useful because they provide a uniform and simple means for computing values of a polynomial using a variety of representations (Bezier, B-spline, NURBS, etc.).

For this reason, some authors prefer to introduce Bezier curves in their polar form.
Derivatives of a Bézier Curve

Lemma 99

Let $B$ be a Bézier curve of degree $n$ with $n + 1$ control points $p_0, p_1, \ldots, p_n$. Its first derivative, which is sometimes called *hodograph*, is a Bézier curve of degree $n - 1$,

$$B'(t) = \sum_{i=0}^{n-1} B_{i,n-1}(t)(n(p_{i+1} - p_i)),$$

whose $n$ control points are given by $n(p_1 - p_0), n(p_2 - p_1), \cdots, n(p_n - p_{n-1})$.

**Proof**: Since the control points are constants, computing the derivative of a Bézier curve is reduced to computing the derivatives of the Bernstein basis polynomials.

$$B'(t) = \frac{d}{dt} \left( \sum_{i=0}^{n} B_{i,n}(t)p_i \right) = \sum_{i=0}^{n} B'_{i,n}(t)p_i \overset{\text{Cor. 85}}{=} n \left( \sum_{i=0}^{n} (B_{i-1,n-1}(t) - B_{i,n-1}(t))p_i \right)$$

$$= n \cdot \left( \sum_{i=1}^{n} B_{i-1,n-1}(t)p_i - \sum_{i=0}^{n-1} B_{i,n-1}(t)p_i \right)$$

$$= n \cdot \left( \sum_{i=0}^{n-1} B_{i,n-1}(t)p_{i+1} - \sum_{i=0}^{n-1} B_{i,n-1}(t)p_i \right) = \sum_{i=0}^{n-1} B_{i,n-1}(t)(n(p_{i+1} - p_i)) \quad \square$$
Lemma 100

A Bézier curve is tangent to the control polygon at the endpoints.

Proof: This is readily proved by computing $B'(0)$ and $B'(1)$.

- Hence, joining two Bézier curves in a $G^1$-continuous way is easy.
- Let $p_0, p_1, \ldots, p_n$ and $p_0^*, p_1^*, \ldots, p_m^*$ be the control points of two Bézier curves $B$ and $B^*$. In order to achieve $C^1$-continuity, we need (in addition to $p_n = p_0^*$)

$$B'(1) = (B^*)'(0) \quad \text{i.e.,} \quad n(p_n - p_{n-1}) = m(p_1^* - p_0^*).$$

- This has an interesting consequence for closed Bézier curves with $p_0 = B(0) = B(1) = p_n$:
  - We get $G^1$-continuity at $p_0$ if $p_0, p_1, p_{n-1}$ are collinear.
  - We get $C^1$-continuity at $p_0$ if $p_1 - p_0 = p_n - p_{n-1}$. 
Subdivision of a Bézier Curve

One can subdivide an \( n \)-degree Bézier curve \( B \) into two curves, at a point \( B(t_0) \) for a given parameter \( t_0 \), such that the newly obtained Bézier curves \( B_1 \) and \( B_2 \) have their own set of control points and are of degree \( n \) each:

- First, we use de Casteljau’s algorithm to compute \( B(t_0) \).
- The subdivided control polygon can then be used to generate the new control polygons for \( B_1 \) and \( B_2 \).
- Note that \( B_1 \) and \( B_2 \) join in a \( G^1 \)-continuous way.
Subdivision of a Bézier Curve

Lemma 101

Let \( p_0, p_1, \ldots, p_n \) be the control points of the Bézier curve \( B \), and let \( p_{i,j} \) denote the control points obtained by de Casteljau’s algorithm for some \( t_0 \in ]0, 1[ \). We define new control points as follows:

\[
\begin{align*}
p_i^* & := p_{i,0} \quad \text{for } i = 0, 1, \ldots, n \\
p_{i}^{**} & := p_{n-i,i} \quad \text{for } i = 0, 1, \ldots, n
\end{align*}
\]

Let \( B^* \) (\( B^{**} \), resp.) denote the Bézier curve defined by \( p_0^*, p_1^*, \ldots, p_n^* \) (\( p_0^{**}, p_1^{**}, \ldots, p_n^{**} \), resp.). Then \( B^* \) and \( B^{**} \) join in a tangent-continuous way at point \( p_n^* = p_0^{**} \), and we have

\[
B^* = B|_{[0,t_0]} \quad \text{and} \quad B^{**} = B|_{[t_0,1]}.
\]

- Note: With every subdivision the control polygons get closer to the Bézier curve. And the approximation is very fast: For \( k \) subdivision steps, the maximum distance \( \varepsilon \) between the resulting control polygon and the curve is

\[
\varepsilon < \frac{c}{2^k}
\]

for some positive constant \( c \).
Degree Elevation of a Bézier Curve

- An increase of the number of control points of a Bézier curve increases the flexibility in designing shapes.
- The key goal is to preserve the shape of the curve. (Recall that Bézier curves change globally if one control point is relocated!)
- Of course, adding one control point means increasing the degree of a Bézier curve by one.
- Let $p_0, p_1, \ldots, p_n$ be the old control points, and $p_0^*, p_1^*, \ldots, p_n^*, p_{n+1}^*$ be the new control points, and denote the Bézier curves defined by them by $B$ and $B^*$.
- How can we guarantee $B(t) = B^*(t)$ for all $t \in [0, 1]$?
- Obviously, we will need

$$p_0 = p_0^* \quad \text{and} \quad p_n = p_{n+1}^*$$

in order to ensure that at least the start and end points of $B$ and $B^*$ match.
- In the sequel, we will find it convenient to extend the index range of the control points of $B$ and introduce (arbitrary) points $p_{-1}$ and $p_{n+1}$. (Both points will be multiplied with factors that equal zero, anyway.)
Degree Elevation of a Bézier Curve

▶ Standard equalities:

\[
\binom{n+1}{i}(1-t) \cdot B_{i,n}(t) = \binom{n+1}{i}(1-t)\binom{n}{i}t^i(1-t)^{n-i}
\]

\[
= \binom{n}{i} \binom{n+1}{i} t^i(1-t)^{n+1-i} = \binom{n}{i} B_{i,n+1}(t)
\]

and

\[
\binom{n+1}{i+1} t \cdot B_{i,n}(t) = \binom{n+1}{i+1} t \binom{n}{i} t^i(1-t)^{n-i}
\]

\[
= \binom{n}{i} \binom{n+1}{i+1} t^{i+1}(1-t)^{n-i} = \binom{n}{i} B_{i+1,n+1}(t)
\]

▶ Hence,

\[
(1-t) \cdot B_{i,n}(t) = \frac{n+1-i}{n+1} B_{i,n+1}(t) \quad \text{and} \quad t \cdot B_{i,n}(t) = \frac{i+1}{n+1} B_{i+1,n+1}(t).
\]
Degree Elevation of a Bézier Curve

\[ B(t) = \sum_{i=0}^{n} B_{i,n}(t)p_i = ((1 - t) + t) \sum_{i=0}^{n} B_{i,n}(t)p_i \]

\[ = (1 - t) \sum_{i=0}^{n} B_{i,n}(t)p_i + t \sum_{i=0}^{n} B_{i,n}(t)p_i = \sum_{i=0}^{n} (1 - t) \cdot B_{i,n}(t)p_i + \sum_{i=0}^{n} t \cdot B_{i,n}(t)p_i \]

\[ = \sum_{i=0}^{n} \frac{n + 1 - i}{n + 1} B_{i,n+1}(t)p_i + \sum_{i=0}^{n} \frac{i + 1}{n + 1} B_{i+1,n+1}(t)p_i \]

\[ = \sum_{i=0}^{n+1} \frac{n + 1 - i}{n + 1} B_{i,n+1}(t)p_i + \sum_{i=0}^{n} \frac{i + 1}{n + 1} B_{i+1,n+1}(t)p_i \]

\[ = \sum_{i=0}^{n+1} \frac{n + 1 - i}{n + 1} B_{i,n+1}(t)p_i + \sum_{i=0}^{n+1} \frac{i}{n + 1} B_{i,n+1}(t)p_{i-1} \]

\[ = \sum_{i=0}^{n+1} \frac{i}{n + 1} \left( \frac{i}{n + 1} p_{i-1} + \frac{n + 1 - i}{n + 1} p_i \right) = \sum_{i=0}^{n+1} B_{i,n+1}(t)p_i^* =: B^*(t) \]

with

\[ p_i^* := \frac{i}{n + 1} p_{i-1} + \frac{n + 1 - i}{n + 1} p_i, \quad i = 0, \ldots, n + 1. \]
Degree Elevation of a Bézier Curve

Lemma 102

Let $p_0, p_1, \ldots, p_n$ be the control points of the degree-$n$ Bézier curve $B$. If we use

$$p_i^* := \left( \frac{i}{n+1} \right) p_{i-1} + \left( 1 - \frac{i}{n+1} \right) p_i$$

for $i = 0, 1, \ldots, n+1$ as control points for the Bézier curve $B^*$ of degree $n+1$, then

$$B(t) = B^*(t) \quad \text{for all } t \in [0, 1].$$
Degree Elevation of a Bézier Curve

- Note that all newly created control points lie on the edges of the previous control polygon.
- Effectively, the corners of the previous control polygon are cut off.
- Degree elevation can be used repeatedly, e.g., in order to arrive at the same degrees for two Bézier curves that join.
- As the degree keeps increasing, the control polygon approaches the Bézier curve and has it as a limiting position.
Bézier Surfaces

Definition 103 (*Bézier surface*)

Suppose that we are given a set of \((n + 1) \cdot (m + 1)\) control points in \(\mathbb{R}^3\), with \(0 \leq i \leq n\) and \(0 \leq j \leq m\), where the control point on the \(i\)-th row and \(j\)-th column is denoted by \(p_{i,j}\). The *Bézier surface* \(S: [0, 1] \times [0, 1] \rightarrow \mathbb{R}^3\) defined by \(p_{i,j}\) is given by

\[
S(u, v) := \sum_{i=0}^{n} \sum_{j=0}^{m} B_{i,n}(u) B_{j,m}(v) p_{i,j} \quad \text{for } (u, v) \in [0, 1] \times [0, 1],
\]

where \(B_{k,d}(x) := \binom{d}{k} x^k (1 - x)^{d-k}\) is the \(k\)-th Bernstein basis polynomial of degree \(d\).

► Since \(B_{i,n}(u)\) and \(B_{j,m}(v)\) are polynomials of degree \(n\) and \(m\), this is called a *Bézier surface of degree* \((n, m)\).

► The set of control points is called a *Bézier net* or *control net*. 
Properties of Bézier Surfaces

Lemma 104

For all \( n, m \in \mathbb{N}_0 \) and all \( 0 \leq i \leq n \) and \( 0 \leq j \leq m \), and all \((u, v) \in [0, 1] \times [0, 1]\), the term \( B_{i,n}(u)B_{j,m}(v) \) is non-negative.

Lemma 105 (Partition of unity)

For all \( m, n \in \mathbb{N}_0 \), the sum of all \( B_{i,n}(u)B_{j,m}(v) \) is one:

\[
\sum_{i=0}^{n} \sum_{j=0}^{m} B_{i,n}(u)B_{j,m}(v) = 1 \quad \text{for all } (u, v) \in [0, 1] \times [0, 1].
\]

Proof: We have for all \( m, n \in \mathbb{N}_0 \) and all \((u, v) \in [0, 1] \times [0, 1]\)

\[
\sum_{i=0}^{n} \sum_{j=0}^{m} B_{i,n}(u)B_{j,m}(v) = \sum_{i=0}^{n} B_{i,n}(u) \left( \sum_{j=0}^{m} B_{j,m}(v) \right) = \sum_{i=0}^{n} B_{i,n}(u) = 1.
\]
Properties of Bézier Surfaces

Lemma 106 *(Convex hull property)*

A Bézier surface lies completely inside the convex hull of its control points.

*Proof:* Recall that $S(u, v)$ is the linear combination of all its control points with non-negative coefficients whose sum is one.

Lemma 107

A Bézier surface passes through the four corners $p_{0,0}$, $p_{n,0}$, $p_{0,m}$ and $p_{n,m}$.

*Proof:* Recall that

$$B_{i,n}(0) = \begin{cases} 1 & \text{for } i = 0, \\ 0 & \text{for } i > 0, \end{cases} \quad \text{and} \quad B_{j,m}(0) = \begin{cases} 1 & \text{for } j = 0, \\ 0 & \text{for } j > 0. \end{cases}$$

Hence, $S(0, 0) = B_{0,n}(0)B_{0,m}(0)p_{0,0} = p_{0,0}$. Similarly for the other corners.

Lemma 108

Applying an affine transformation to the control points results in the same transformation as obtained by transforming the surface’s equation.
Isoparametric Curves of Bézier Surfaces

Lemma 109

Consider a Bézier surface \( S : [0, 1] \times [0, 1] \rightarrow \mathbb{R}^3 \) defined by \((n + 1) \cdot (m + 1)\) control points \( p_{i,j} \), with \( 0 \leq i \leq n \) and \( 0 \leq j \leq m \), and let \( v_0 \in [0, 1] \) be fixed. Then \( C : [0, 1] \rightarrow \mathbb{R}^3 \) defined as

\[
C(u) := \sum_{i=0}^{n} \sum_{j=0}^{m} B_{i,n}(u) B_{j,m}(v_0) p_{i,j} \quad \text{for } u \in [0, 1]
\]

is a Bézier curve defined by the \( n + 1 \) control points \( q_0, q_1, \ldots, q_n \in \mathbb{R}^3 \), where

\[
q_i := \sum_{j=0}^{m} B_{j,m}(v_0) p_{i,j} \quad \text{for } 0 \leq i \leq n.
\]

Proof: We have for all \( u \in [0, 1] \)

\[
C(u) = \sum_{i=0}^{n} \sum_{j=0}^{m} B_{i,n}(u) B_{j,m}(v_0) p_{i,j} = \sum_{i=0}^{n} B_{i,n}(u) \left( \sum_{j=0}^{m} B_{j,m}(v_0) p_{i,j} \right) = \sum_{i=0}^{n} B_{i,n}(u) q_i.
\]

Analogously for fixed \( u_0 \).
Isoparametric Curves of Bézier Surfaces

Corollary 110

The boundary curves of a Bézier surface are Bézier curves defined by the boundary points of its control net.

Lemma 111 (Tangency in the corner points)

Consider a Bézier surface \( S: [0, 1] \times [0, 1] \to \mathbb{R}^3 \) defined by \((n + 1) \cdot (m + 1)\) control points \( p_{i,j} \), with \( 0 \leq i \leq n \) and \( 0 \leq j \leq m \). The tangent plane at \( S(0, 0) = p_{0,0} \) is spanned by the vectors \( p_{1,0} - p_{0,0} \) and \( p_{0,1} - p_{0,0} \).
Bézier Surface as Tensor-Product Surface

- A Bézier surface is generated by “multiplying” two Bézier curves: tensor product surface.

Lemma 112

Consider a Bézier surface \( S: [0, 1] \times [0, 1] \rightarrow \mathbb{R}^3 \) defined by \((n + 1) \cdot (m + 1)\) control points \( p_{i,j} \), with \( 0 \leq i \leq n \) and \( 0 \leq j \leq m \). Then \( S \) is a tensor-product surface:

\[
S(u, v) = (B_{0,n}(u), B_{1,n}(u), \ldots, B_{n,n}(u)) \cdot \begin{pmatrix}
  p_{0,0} & p_{0,1} & \cdots & p_{0,m} \\
  p_{1,0} & p_{1,1} & \cdots & p_{1,m} \\
  \vdots & \vdots & \ddots & \vdots \\
  p_{n,0} & p_{n,1} & \cdots & p_{n,m}
\end{pmatrix} \cdot \begin{pmatrix}
  B_{0,m}(v) \\
  B_{1,m}(v) \\
  \vdots \\
  B_{m,m}(v)
\end{pmatrix}
\]

Proof: Just do the math!
B-Spline Curves and Surfaces

- Shortcomings of Bézier Curves
- B-Spline Basis Functions
- B-Spline Curves
- B-Spline Surfaces
- Non-Uniform Rational B-Spline Curves and Surfaces
Shortcomings of Bézier Curves

- Modifying the vertex \( p_j \) of a Bézier curve causes a global change of the entire curve:

\[
B^*(t) = B(t) + B_{j,n}(t)v
\]

- But \( B_{j,n}(t) \neq 0 \) for all \( t \) with \( 0 < t < 1 \)!
Shortcomings of Bézier Curves

- While it is easy to join two Bézier curves with $G^1$ continuity, achieving $C^2$ or even higher continuity is quite cumbersome.
Shortcomings of Bézier Curves

- While it is easy to join two Bézier curves with $G^1$ continuity, achieving $C^2$ or even higher continuity is quite cumbersome.
- Even worse, changing the common end point of two consecutive Bézier curves destroys $G^1$ continuity.

\[ p_0 \rightarrow p_1 \rightarrow p_2 \rightarrow p_3 = p_0^* \]
Shortcomings of Bézier Curves

- While it is easy to join two Bézier curves with $G^1$ continuity, achieving $C^2$ or even higher continuity is quite cumbersome.
- This will be easier for B-spline curves. (Depicted are two cubic B-splines.)
Shortcomings of Bézier Curves

- While it is easy to join two Bézier curves with $G^1$ continuity, achieving $C^2$ or even higher continuity is quite cumbersome.
- This will be easier for B-spline curves. (Depicted are two cubic B-splines.)
Shortcomings of Bézier Curves

- It is fairly difficult to squeeze a Bézier curve close to a sharp corner of the control polygon.
Shortcomings of Bézier Curves

- Adding additional control vertices hardly helps but increases the degree of the Bézier curve, which may result in oscillation and cause numerical instability.

three new vertices
Curves consisting of just one segment have several drawbacks:
- To satisfy all given constraints, often a high polynomial degree is required.
- The number of control points is directly related to the degree.
- Interactive shape design is inaccurate or requires high computational costs.

The solution is to use a sequence of polynomial or rational curves to form one continuous curve: spline.

Historically, the term spline (Dt.: Straklatte) was used for elastic wooden strips in the shipbuilding industry, which pass through given constrained points called ducks such that the strain of the strip is minimized.

Mathematical splines were introduced by Isaac Jacob Schoenberg in 1946.

**Warning**
The terminology and the definitions used for B-splines vary from author to author! Thus, make sure to check carefully the definitions given in textbooks or research papers.
### Definition 113 (Spline)

A curve $C(t) : [a, b] \rightarrow \mathbb{R}^2$ is called a spline of degree $k$ (and order $k+1$), for $k \in \mathbb{N}$, if there exist

- $m$ polynomials $P_1, P_2, \ldots, P_m$ of degree $k$, for $m \in \mathbb{N}$, and
- $m+1$ parameters $t_0, \ldots, t_m \in \mathbb{R}$

such that

1. $a = t_0 < t_1 < \ldots < t_{m-1} < t_m = b$,
2. $C|_{[t_{i-1}, t_i]} = P_i|_{[t_{i-1}, t_i]}$ for all $i \in \{1, 2, \ldots, m\}$.
Introduction to B-Splines

- The numbers $t_0, \ldots, t_m$ are called breakpoints or knots.
- The definition implies
  
  $$P_i(t_i) = P_{i+1}(t_i) \quad \text{for all } i \in \{1, 2, \ldots, m - 1\}.$$

- Special case $k = 1$: We get a polygonal curve.

- The polynomials join with some unknown degree of continuity at the breakpoints. (We have at least $C^0$-continuity.)
- Obvious problem: How can we achieve a reasonable degree of continuity?
Knot Vector

Definition 114 (Knot vector, Dt.: Knotenvektor)

In general, a knot vector is a sequence of non-decreasing real numbers. A finite knot vector is a sequence of \( m + 1 \) real numbers \( \tau := (t_0, t_1, t_2, \ldots, t_m) \), for some \( m \in \mathbb{N} \), such that \( t_i \leq t_{i+1} \) for all \( 0 \leq i < m \).

An infinite knot vector is a sequence of real numbers \( \tau := (t_0, t_1, t_2, \ldots) \) such that \( t_i \leq t_{i+1} \) for all \( i \in \mathbb{N}_0 \).

A bi-infinite knot vector is a sequence of real numbers \( \tau := (\ldots, t_{-2}, t_{-1}, t_0, t_1, t_2, \ldots) \) such that \( t_i \leq t_{i+1} \) for all \( i \in \mathbb{Z} \).

The numbers \( t_i \) are called knots, and the \( i \)-th knot span is given by the (half-open) interval \( [t_i, t_{i+1}[ \subset \mathbb{R} \).

For (bi)infinite knot vectors we assume \( \sup_{i \to \infty} t_i = \infty \) and \( \inf_{i \to -\infty} t_i = -\infty \).

For some of the subsequent definitions we will find it convenient to deal with (bi)infinite knot vectors. With some extra care for “boundary conditions” one could replace all (bi)infinite knot vectors by finite knot vectors.
Knot Vector

Definition 115 (Multiplicity of a knot, Dt.: Vielfachheit eines Knotens)

Let $\tau$ be a finite or (bi)infinite knot vector. If a knot $t_i$ appears exactly $k > 1$ times in $\tau$, for a permissible value of $i \in \mathbb{Z}$, i.e., if $t_{i-1} < t_i = t_{i+1} = \cdots = t_{i+k-1} < t_{i+k}$, then $t_i$ is a multiple knot of multiplicity $k$. Otherwise, if $t_i$ appears only once in $\tau$ then $t_i$ is a simple knot.

Definition 116 (Uniform knot vector)

A finite or (bi)infinite knot vector is uniform if there exists $c \in \mathbb{R}^+$ such that $t_{i+1} - t_i = c$ for all (permissible) values of $i \in \mathbb{Z}$, except for possibly the first and last knots of higher multiplicity in case of a finite knot sequence. Otherwise, the knot vector is non-uniform.
B-Spline Basis Functions

► We define the B-spline basis functions analytically, using the recurrence formula by de Boor, Cox and Mansfield.

Definition 117 (B-spline basis function)

Let $\tau$ be a finite or (bi)infinite knot vector. For all (permissible) $i \in \mathbb{Z}$ and $k \in \mathbb{N}_0$, the $i$-th B-spline basis function, $N_{i,k,\tau}(t)$, of degree $k$ (and order $k + 1$) relative to $\tau$ is defined as,

if $k = 0$,

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

and if $k > 0$ as

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

► In case of multiple knots, indeterminate terms of the form $\frac{0}{0}$ are taken as zero!
► Alternatively, one can demand $t_i < t_{i+k}$ for all (permissible) $i \in \mathbb{Z}$.
► Aka: Normalized B(asic)-Spline Blending Functions.
B-Spline Basis Functions

Plugging into the definition yields

\[
N_{i,1,\tau}(t) = \frac{t - t_i}{t_{i+1} - t_i} N_{i,0,\tau}(t) + \frac{t_{i+2} - t}{t_{i+2} - t_{i+1}} N_{i+1,0,\tau}(t)
\]

\[
= \begin{cases} 
0 & \text{if } t \not\in [t_i, t_{i+2}], \\
\frac{t - t_i}{t_{i+1} - t_i} & \text{if } t \in [t_i, t_{i+1}], \\
\frac{t_{i+2} - t}{t_{i+2} - t_{i+1}} & \text{if } t \in [t_{i+1}, t_{i+2}].
\end{cases}
\]

The functions \( N_{i,1,\tau}(t) \) are called hat functions or chapeau functions. They are widely used in signal processing and finite-element techniques.

Note that \( N_{i,1,\tau}(t) \) is continuous at \( t_{i+1} \).

For a uniform knot vector \( \tau \) with \( c := t_{i+1} - t_i \) this simplifies to

\[
N_{i,1,\tau}(t) = \begin{cases} 
0 & \text{if } t \not\in [t_i, t_{i+2}], \\
\frac{1}{c}(t - t_i) & \text{if } t \in [t_i, t_{i+1}], \\
\frac{1}{c}(t_{i+2} - t) & \text{if } t \in [t_{i+1}, t_{i+2}].
\end{cases}
\]
B-Spline Basis Functions

- Basis functions $N_{i,k,\tau}$.

$N_{i,0}$ is a step function that is 1 over the knot span $[t_i, t_{i+1}]$.

$N_{i,1}$ is a piecewise linear function that is non-zero over two knot spans $[t_i, t_{i+2}]$ and goes from 0 to 1 and back.

$N_{i,2}$ is a piecewise quadratic function that is non-zero over three knot spans $[t_i, t_{i+3}]$. 
Sample B-Spline Basis Functions

- Basis functions of degree 0:

  \[ N_{0,0} \]
  \[
  \begin{array}{ccccccc}
  t_0 & t_1 & t_2 & t_3 & t_4 & t_5 \\
  \end{array}
  \]

  \[ N_{1,0} \]
  \[
  \begin{array}{ccccccc}
  t_0 & t_1 & t_2 & t_3 & t_4 & t_5 \\
  \end{array}
  \]

  \[ N_{2,0} \]
  \[
  \begin{array}{ccccccc}
  t_0 & t_1 & t_2 & t_3 & t_4 & t_5 \\
  \end{array}
  \]

  \[ N_{3,0} \]
  \[
  \begin{array}{ccccccc}
  t_0 & t_1 & t_2 & t_3 & t_4 & t_5 \\
  \end{array}
  \]

  \[ N_{4,0} \]
  \[
  \begin{array}{ccccccc}
  t_0 & t_1 & t_2 & t_3 & t_4 & t_5 \\
  \end{array}
  \]
Sample B-Spline Basis Functions

- Basis functions of degree 1:

\[ N_{0,1}, N_{1,1}, N_{2,1}, N_{3,1} \]
Sample B-Spline Basis Functions

Basis functions of degree 2:

- $N_{0,2}$
- $N_{1,2}$
- $N_{2,2}$

Graphs showing the basis functions $N_{0,2}$, $N_{1,2}$, and $N_{2,2}$ with control points at $t_0$, $t_1$, $t_2$, $t_3$, $t_4$, and $t_5$. The graphs illustrate how the basis functions vary within the interval defined by the control points, with $N_{0,2}$ and $N_{2,2}$ being non-zero in the interval $(t_0, t_5)$ and $N_{1,2}$ being non-zero in $(t_1, t_4)$. The shaded region indicates the overlapping influence of two basis functions.
Sample B-Spline Basis Functions

- Uniform knot vector \((0, \frac{1}{9}, \frac{2}{9}, \frac{3}{9}, \frac{4}{9}, \frac{5}{9}, \frac{6}{9}, \frac{7}{9}, \frac{8}{9}, 1)\) with ten knots.
Sample B-Spline Basis Functions

- Clamped uniform knot vector \((0, 0, 0, \frac{1}{5}, \frac{2}{5}, \frac{3}{5}, \frac{4}{5}, 1, 1, 1)\) with ten knots.
Sample B-Spline Basis Functions

- Non-uniform knot vector \((0, \frac{1}{6}, \frac{1}{3}, \frac{1}{2}, \frac{1}{2}, \frac{2}{3}, \frac{5}{6}, 1)\) with ten knots.

\[ N_{0,0}, N_{1,0}, N_{2,0}, N_{6,0}, N_{7,0}, N_{8,0} \]

\[ N_{0,1}, N_{1,1}, N_{2,1}, N_{5,1}, N_{6,1}, N_{7,1} \]

\[ N_{0,2}, N_{1,2}, N_{2,2}, N_{4,2}, N_{5,2}, N_{6,2} \]

\[ N_{0,3}, N_{1,3}, \ldots, N_{5,3} \]
Properties of B-Spline Basis Functions

▶ It is common to omit the explicit mentioning of the dependency of $N_{i,k,\tau}(t)$ on $\tau$, and to write simply $N_{i,k}(t)$. (And sometimes we simply write $N_{i,k} \ldots$)

▶ For $k > 0$, each $N_{i,k,\tau}(t)$ is a linear combination of two B-spline basis functions of degree $k - 1$: $N_{i,k-1,\tau}(t)$ and $N_{i+1,k-1,\tau}(t)$.

▶ This suggests a recursive analysis of the dependencies.

\[
\begin{align*}
N_{0,0} & \\
N_{0,1} & \\
N_{1,0} & N_{0,2} \\
N_{1,1} & N_{0,3} \\
N_{2,0} & N_{1,2} \\
N_{2,1} & N_{1,3} \\
N_{3,0} & N_{2,2} \\
N_{3,1} & \\
N_{4,0} & 
\end{align*}
\]
Properties of B-Spline Basis Functions

- It is common to omit the explicit mentioning of the dependency of \( N_{i,k,\tau}(t) \) on \( \tau \), and to write simply \( N_{i,k}(t) \). (And sometimes we simply write \( N_{i,k} \) . . .)
- For \( k > 0 \), each \( N_{i,k,\tau}(t) \) is a linear combination of two B-spline basis functions of degree \( k - 1 \): \( N_{i,k-1,\tau}(t) \) and \( N_{i+1,k-1,\tau}(t) \).
- \( N_{i,k,\tau}(t) \) depends on \( N_{i,0,\tau}(t), N_{i+1,0,\tau}(t), \ldots, N_{i+k,0,\tau}(t) \).
Properties of B-Spline Basis Functions

- It is common to omit the explicit mentioning of the dependency of $N_{i,k,\tau}(t)$ on $\tau$, and to write simply $N_{i,k}(t)$. (And sometimes we simply write $N_{i,k}$.)

- For $k > 0$, each $N_{i,k,\tau}(t)$ is a linear combination of two B-spline basis functions of degree $k - 1$: $N_{i,k-1,\tau}(t)$ and $N_{i+1,k-1,\tau}(t)$.

- $N_{i,k,\tau}(t)$ is non-zero only for $t \in [t_i, t_{i+k+1}]$. 

\[
\begin{align*}
[t_0, t_1] & : N_{0,0} & N_{0,1} & N_{0,2} & N_{0,3} & N_{0,4} \\
[t_1, t_2] & : N_{1,0} & N_{1,1} & N_{1,2} & N_{1,3} \\
[t_2, t_3] & : N_{2,0} & N_{2,1} & N_{2,2} \\
[t_3, t_4] & : N_{3,0} & N_{3,1} \\
[t_4, t_5] & : N_{4,0}
\end{align*}
\]
Properties of B-Spline Basis Functions

Lemma 118 (Local support)

Let \( \tau := (\ldots, t_{-2}, t_{-1}, t_0, t_1, t_2, \ldots) \) be a (bi)infinite knot vector. For all (permissible) \( i \in \mathbb{Z} \) and \( k \in \mathbb{N}_0 \) we have

\[
N_{i,k,\tau}(t) = 0 \quad \text{if} \quad t \notin [t_i, t_{i+k+1}].
\]

Proof: We do a proof by induction on \( k \). Let \( i \in \mathbb{Z} \) be arbitrary but fixed.

I.B.: By definition, this claim is correct for \( k = 0 \).

I.H.: Suppose that it is true for all basis functions of degree \( k - 1 \), for some arbitrary but fixed \( k \in \mathbb{N} \). I.e., \( N_{i,k-1,\tau}(t) = 0 \) if \( t \notin [t_i, t_{i+k}] \).

I.S.: Recall that

\[
N_{i,k,\tau}(t) = \frac{t - t_i}{t_{i+k} - t_i} N_{i,k-1,\tau}(t) + \frac{t_{i+k+1} - t}{t_{i+k+1} - t_{i+1}} N_{i+1,k-1,\tau}(t).
\]

Hence, \( N_{i,k,\tau}(t) = 0 \) if \( t \notin ([t_i, t_{i+k}] \cup [t_{i+1}, t_{i+k+1}]), \) i.e., if \( t \notin [t_i, t_{i+k+1}] \). \( \square \)
Properties of B-Spline Basis Functions

Lemma 119 (Non-negativity)

We have \( N_{i,k,\tau}(t) \geq 0 \) for all (permissible) \( i \in \mathbb{Z} \) and \( k \in \mathbb{N}_0 \), and all real \( t \).

Proof: Again we do a proof by induction on \( k \). Let \( i \in \mathbb{Z} \) be arbitrary but fixed.
I.B.: By definition, this claim is correct for \( k = 0 \).
I.H.: Suppose that it is true for all basis functions of degree \( k - 1 \), for some arbitrary but fixed \( k \in \mathbb{N} \).
I.S.: Lemma 118 tells us that \( N_{i,k,\tau}(t) = 0 \) if \( t \notin [t_i, t_{i+k+1}] \). Hence, we can focus on \( t \in [t_i, t_{i+k+1}] \) and get

\[
N_{i,k,\tau}(t) = \left( \frac{t - t_i}{t_{i+k} - t_i} \right) \cdot N_{i,k-1,\tau}(t) + \left( \frac{t_{i+k+1} - t}{t_{i+k+1} - t_{i+1}} \right) \cdot N_{i+1,k-1,\tau}(t)
\]

\[
\geq 0 \quad \text{for } t \in [t_i, t_{i+k+1}] \quad \geq 0 \quad \text{(I.H.)}
\]

\[
\geq 0.
\]

Lemma 120

For all \( k \in \mathbb{N} \), all B-spline basis functions of degree \( k \) are continuous.
Lemma 121

Let $\tau := (\ldots, t_{-2}, t_{-1}, t_0, t_1, t_2, \ldots)$ be a (bi)infinite knot vector. For all (permissible) $i \in \mathbb{Z}$ and $k \in \mathbb{N}_0$, the basis functions

$$N_{i-k, k, \tau}(t), \ N_{i-k+1, k, \tau}(t), \ldots, \ N_{i, k, \tau}(t)$$

are the only (at most) $k + 1$ basis functions of degree $k$ that are (possibly) non-zero over the interval $[t_i, t_{i+1}]$. 
Properties of B-Spline Basis Functions

Lemma 122

Let $\tau := (\ldots, t_{-2}, t_{-1}, t_0, t_1, t_2, \ldots)$ be a (bi)infinite knot vector. For all (permissible) $i \in \mathbb{Z}$ and $k \in \mathbb{N}_0$, the basis functions

$$N_{i-k,k,\tau}(t), \ N_{i-k+1,k,\tau}(t), \ldots, \ N_{i,k,\tau}(t)$$

are the only (at most) $k + 1$ basis functions of degree $k$ that are (possibly) non-zero over the interval $[t_i, t_{i+1}]$.

Proof: The Local Support Lemma 118 tells us that

$$N_{j,k,\tau}(t) = 0 \quad \text{if} \ t \notin [t_j, t_{j+k+1}]$$

and, thus, possibly non-zero only if $t \in [t_j, t_{j+k+1}]$. Hence, $N_{j,k,\tau}(t) \neq 0$ over $[t_i, t_{i+1}]$ only if $i \geq j$ and $i + 1 \leq j + k + 1$, i.e., if $j \leq i$ and $j \geq i - k$. Thus,

$$N_{i-k,k,\tau}(t), \ N_{i-k+1,k,\tau}(t), \ldots, \ N_{i,k,\tau}(t)$$

are the only B-spline basis functions that are (possibly) non-zero over $[t_i, t_{i+1}]$. 
Properties of B-Spline Basis Functions

Lemma 123

All B-spline basis functions of degree $k$ are piecewise polynomials of degree $k$.

Lemma 124

All B-spline basis functions of degree $k$ are $k - r$ times continuously differentiable at a knot of multiplicity $r$, and $k - 1$ times continuously differentiable everywhere else. The first derivative of $N_{i,k}(t)$ is given as follows:

$$N'_{i,k}(t) = \frac{k}{t_{i+k} - t_i} N_{i,k-1}(t) - \frac{k}{t_{i+k+1} - t_{i+1}} N_{i+1,k-1}(t)$$
Properties of B-Spline Basis Functions

Lemma 125
For a uniform knot vector \( \tau \) all B-spline basis functions of the same degree are shifted copies of each other: For all (permissible) \( i \in \mathbb{Z} \) and \( k \in \mathbb{N}_0 \) we have
\[
N_{i,k,\tau}(t) = N_{0,k,\tau}(t - i \cdot c),
\]
where \( c := t_1 - t_0 \).

Lemma 126 (Partition of unity)
Let \( \tau = (t_0, t_1, t_2, \ldots, t_m) \) be a finite knot vector, and \( k \in \mathbb{N}_0 \) with \( k < \frac{m}{2} \). Then,
\[
\sum_{i=0}^{m-k-1} N_{i,k,\tau}(t) = 1 \quad \text{for all } t \in [t_k, t_{m-k}].
\]

Corollary 127
Let \( \tau = (t_0, t_1, t_2, \ldots, t_{n+k+1}) \) be a finite knot vector, for some \( k \in \mathbb{N}_0 \) with \( k \leq n \). Then,
\[
\sum_{i=0}^{n} N_{i,k,\tau}(t) = 1 \quad \text{for all } t \in [t_k, t_{n+1}].
\]
Properties of B-Spline Basis Functions

Proof of Lemma 126 (Partition of Unity): We do a proof by induction on \( k \).

I.B.: By definition, this claim is correct for \( k = 0 \).

I.H.: Suppose that it is true for degree \( k - 1 \), for some arbitrary but fixed \( k \in \mathbb{N} \) such that \( k < \frac{m}{2} \). I.e., suppose that \( \sum_{i=0}^{m-k} N_{i,k-1,\tau}(t) = 1 \) for all \( t \in [t_{k-1}, t_{m-k+1}] \).

I.S.: Recall that (by Lem. 118)

\[
N_{0,k-1,\tau}(t) = 0 \text{ for } t \notin [t_0, t_k] \quad \text{and} \quad N_{m-k,k-1,\tau}(t) = 0 \text{ for } t \notin [t_{m-k}, t_m].
\]

Let \( t \in [t_k, t_{m-k}] \) be arbitrary but fixed. Applying the recursion yields

\[
\sum_{i=0}^{m-k-1} N_{i,k,\tau}(t) = \sum_{i=0}^{m-k-1} \left( \frac{t-t_i}{t_{i+k} - t_i} N_{i,k-1,\tau}(t) + \frac{t_{i+k+1} - t}{t_{i+k+1} - t_{i+1}} N_{i+1,k-1,\tau}(t) \right)
\]

\[
= \sum_{i=1}^{m-k-1} \frac{t-t_i}{t_{i+k} - t_i} N_{i,k-1,\tau}(t) + \sum_{i=0}^{m-k-2} \frac{t_{i+k+1} - t}{t_{i+k+1} - t_{i+1}} N_{i+1,k-1,\tau}(t)
\]

\[
= \sum_{i=1}^{m-k-1} \frac{t-t_i}{t_{i+k} - t_i} N_{i,k-1,\tau}(t) + \sum_{i=0}^{m-k-1} \frac{t_{i+k} - t}{t_{i+k} - t_i} N_{i,k-1,\tau}(t)
\]

\[
= \sum_{i=1}^{m-k-1} N_{i,k-1,\tau}(t) = \sum_{i=0}^{m-k} N_{i,k-1,\tau}(t) \quad t \in [t_{k-1}, t_{m-k+1}] = 1.
\]
B-Spline Curves

Definition 128 (B-spline curve)

For \( n \in \mathbb{N} \) and \( k \in \mathbb{N}_0 \) with \( k \leq n \), consider a set of \( n + 1 \) control points with position vectors \( p_0, p_1, \ldots, p_n \) in the plane, and let \( \tau := (t_0, t_1, \ldots, t_{n+k+1}) \) be a knot vector. Then the B-spline curve of degree \( k \) (and order \( k + 1 \)) relative to \( \tau \) with control points \( p_0, p_1, \ldots, p_n \) is given by

\[
P(t) := \sum_{i=0}^{n} N_{i,k,\tau}(t)p_i \quad \text{for} \ t \in [t_k, t_{n+1}],
\]

where \( N_{i,k,\tau} \) is the \( i \)-th B-spline basis function of degree \( k \) relative to \( \tau \).

- The degree \( k \) is (except for \( k \leq n \)) independent of the number \( n + 1 \) of control points!
- The restriction of \( t \) to the interval \( [t_k, t_{n+1}] \) guarantees that the basis functions sum up to 1 for all (permissible) values of \( t \). (Recall the Partition of Unity, Cor. 127.)
Clamped and Unclamped B-Spline Curves

Definition 129 (*Clamped B-spline*)

Let $\mathcal{P}$ be a B-spline curve of degree $k$ defined by $n + 1$ control points with position vectors $p_0, p_1, \ldots, p_n$, over the knot vector $\tau := (t_0, t_1, \ldots, t_{n+k+1})$, for $n \in \mathbb{N}$ and $k \in \mathbb{N}_0$ with $k \leq n$. If $t_0 = t_1 = \ldots = t_k < t_{k+1}$ and $t_n < t_{n+1} = \ldots = t_{n+k+1}$ then we say that the knot vector and the B-spline curve are *clamped*.

- Recall that the Partition of Unity (Cor. 127) holds for all $t \in [t_k, t_{n+1}]$.
- Typically, for a clamped knot vector,

$$0 = t_0 = t_1 = \ldots = t_k \quad \text{and} \quad t_{n+1} = \ldots = t_{n+k+1} = 1.$$ 

Lemma 130

Let $\mathcal{P}$ be a B-spline curve of degree $k$ defined by $n + 1$ control points with position vectors $p_0, p_1, \ldots, p_n$, over the clamped knot vector $\tau := (t_0, t_1, \ldots, t_{n+k+1})$, for $n \in \mathbb{N}$ and $k \in \mathbb{N}_0$ with $k \leq n$. Then $\mathcal{P}$ starts in $p_0$ and ends in $p_n$. 

© M. Held (Univ. Salzburg) Geometric Modeling (WS 2018/19)
Clamped and Unclamped B-Spline Curves

Control points: \[
\left\{ \left( \begin{array}{c} 0 \\ 0 \end{array} \right), \left( \begin{array}{c} 0 \\ 2 \end{array} \right), \left( \begin{array}{c} 2 \\ 3 \end{array} \right), \left( \begin{array}{c} 4 \\ 0 \end{array} \right), \left( \begin{array}{c} 6 \\ 3 \end{array} \right), \left( \begin{array}{c} 8 \\ 2 \end{array} \right), \left( \begin{array}{c} 8 \\ 0 \end{array} \right) \right\}.
\]

uniform unclamped cubic B-spline: \( \tau = (0, \frac{1}{10}, \frac{1}{5}, \frac{3}{10}, \frac{2}{5}, \frac{1}{2}, \frac{3}{5}, \frac{7}{10}, \frac{4}{5}, \frac{9}{10}, 1) \)

uniform clamped cubic B-spline: \( \tau = (0, 0, 0, 0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1, 1, 1, 1) \).
Proof of Lemma 130

We prove $\mathcal{P}(t_k) = p_0$. Recall that $N_{0,k}(t)$ is non-zero only for $t \in [t_0, t_{k+1}]$.

However, for a clamped knot vector with $t_0 = t_1 = \ldots = t_k < t_{k+1}$ we have

$$N_{0,0}(t) = N_{1,0}(t) = \ldots = N_{k-1,0}(t) = 0 \quad \text{for all $t$, and $N_{k,0}(t_k) = 1$.}$$

The recursion formula for the B-spline basis functions yields

$$N_{i,j}(t) = 0 \quad \text{for all $i, j$ with $i + j \leq k - 1$ and for all $t$.}$$
Proof of Lemma 130

Applying the standard recursion for the B-spline basis functions at parameter \( t_k \),

\[
N_{i,k}(t_k) = \frac{t_k - t_i}{t_{i+k} - t_i} N_{i,k-1}(t_k) + \frac{t_{i+k+1} - t_k}{t_{i+k+1} - t_{i+1}} N_{i+1,k-1}(t_k),
\]

for \( i = 0 \) (and subsequently for \( i = j \) and \( k - j \) for \( j \in \{1, \ldots, k - 1\} \)) yields

\[
N_{0,k}(t_k) = \frac{t_k - t_0}{t_k - t_0} N_{0,k-1}(t_k) + \frac{t_{k+1} - t_k}{t_{k+1} - t_1} N_{1,k-1}(t_k) \\
= \frac{t_{k+1} - t_k}{t_{k+1} - t_k} N_{1,k-1}(t_k) = N_{1,k-1}(t_k) \\
= \frac{t_k - t_1}{t_k - t_1} N_{1,k-2}(t_k) + \frac{t_{k+1} - t_k}{t_{k+1} - t_2} N_{2,k-2}(t_k) \\
= N_{2,k-2}(t_k) = \cdots = N_{k,0}(t_k) \\
= 1.
\]

Hence, due to the Partition of Unity, Cor. 127, \( N_{i,k}(t_k) = 0 \) for \( i > 0 \) and we get

\[
\sum_{i=0}^{n} N_{i,k}(t_k)p_i = N_{0,k}(t_k)p_0 = p_0.
\]

\[\square\]
Generation of Knot Vector

- Suppose that a B-spline curve over \([0, 1]\) has \(n + 1\) control points \(\{p_0, p_1, \ldots, p_n\}\) and degree \(k\).
- We need \(m + 1\) knots, where \(m = n + k + 1\).
- If the B-spline curve is clamped then we get
  \[
  t_0 = t_1 = \ldots = t_k = 0 \quad \text{and} \quad t_{n+1} = t_{n+2} = \ldots = t_{n+k+1} = 1.
  \]
- The remaining \(n - k\) knots can be spaced uniformly or non-uniformly.
- For uniformly spaced internal knots the interval \([0, 1]\) is divided into \(n - k + 1\) subintervals. In this case the knots are given as follows:
  \[
  t_0 = t_1 = \ldots = t_k = 0
  \]
  \[
  t_{k+j} = \frac{j}{n - k + 1} \quad \text{for} \ j = 1, 2, \ldots, n - k
  \]
  \[
  t_{n+1} = t_{n+2} = \ldots = t_{n+k+1} = 1
  \]
Generation of Knot Vector

- Suppose that $n = 6$, i.e., that we have seven control points $p_0, \ldots, p_6$ and want to construct a clamped cubic B-spline curve. (Hence, $k = 3$.)
- We have in total $m + 1 = n + k + 2 = 6 + 3 + 2 = 11$ knots and get

$$\tau := (0, 0, 0, 0, \frac{1}{4}, \frac{2}{4}, \frac{3}{4}, 1, 1, 1, 1)$$

as uniform knot vector.
- For $\{p_0, \ldots, p_6\} := \left\{ \begin{pmatrix} 0 \\ 0 \\ \end{pmatrix}, \begin{pmatrix} 0.2 \\ 2 \\ \end{pmatrix}, \begin{pmatrix} 2 \\ 3 \\ \end{pmatrix}, \begin{pmatrix} 3 \\ 2 \\ \end{pmatrix}, \begin{pmatrix} 3 \\ 0 \\ \end{pmatrix}, \begin{pmatrix} 5 \\ 0 \\ \end{pmatrix}, \begin{pmatrix} 7 \\ 1 \\ \end{pmatrix} \right\}$ we get the following clamped, cubic and $C^2$-continuous B-spline curve:
Properties of B-Spline Curves

Lemma 131

The lower the degree of a B-spline curve, the closer it follows its control polygon.

Sketch of Proof: The lower the degree, the fewer control points contribute to \( P(t) \). For \( k = 1 \) it is simply the convex combination of pairs of control points.

Clamped uniform B-spline of degree 10 for a control polygon with 14 vertices:

\[
\{(1, 1), (1, 3), (3, 5), (5, 5), (6, 4), (5, 2), (3, 2), (3, 1), (11, 1), (8, 3), (8, 5), (10, 6), (4, 7), (1, 5)\}
\]
Properties of B-Spline Curves

Lemma 132 (Variation diminishing property)

If a straight line intersects the control polygon of a B-spline curve \( m \) times then it intersects the actual B-spline curve at most \( m \) times.

Lemma 133 (Affine invariance)

Any B-spline representation is affinely invariant, i.e., given any affine map \( \pi \), the image curve \( \pi(P) \) of a B-spline curve \( P \) with control points \( p_0, p_1, \ldots, p_n \) has the control points \( \pi(p_0), \pi(p_1), \ldots, \pi(p_n) \).

Sketch of Proof: The proof is identical to the proof of the affine invariance of Bézier curves, recall Lem. 95.
Properties of B-Spline Curves

Lemma 134

Let $P$ be a clamped B-spline curve of degree $k$ over $[0, 1]$ defined by $k + 1$ control points with position vectors $p_0, p_1, \ldots, p_k$ and the knot vector $\tau := (t_0, t_1, \ldots, t_{2k+1})$, for $k \in \mathbb{N}_0$. Then $P$ is a Bézier curve of degree $k$.

- Note: This implies $0 = t_0 = t_1 = \ldots = t_k$ and $1 = t_{k+1} = \ldots = t_{2k} = t_{2k+1}$.
- Of course, this lemma can also be formulated for a parameter interval other than $[0, 1]$.
- Clamped (uniform) B-spline of degree 3 for knot vector $(0, 0, 0, 0, 1, 1, 1, 1)$ and control polygon $\left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 3 \end{pmatrix}, \begin{pmatrix} 3 \\ 3 \end{pmatrix}, \begin{pmatrix} 4 \\ 0 \end{pmatrix} \right\}$:
Derivatives of B-Spline Curves

Lemma 135

Let \( \mathcal{P} \) be a B-spline curve of degree \( k \) defined by \( n + 1 \) control points with position vectors \( p_0, p_1, \ldots, p_n \), and the knot vector \( \tau := (t_0, t_1, \ldots, t_{n+k+1}) \), for \( n \in \mathbb{N} \) and \( k \in \mathbb{N}_0 \) with \( k \leq n \). Then

\[
\mathcal{P}'(t) = \sum_{i=0}^{n-1} N_{i+1,k-1}(t) q_i \quad \text{for } t \in [t_k, t_{n+1}[, 
\]

where

\[
q_i := \frac{k}{t_{i+k+1} - t_{i+1}} \left(p_{i+1} - p_i\right) \quad \text{for } i \in \{0, 1, \ldots, n - 1\}
\]

and the knot vector \( \tau \) remains unchanged.

Sketch of Proof: This is a consequence of Lem. 124 and some (lengthy) analysis.

\[\square\]

- Since the first derivative of a B-spline curve is another B-spline curve, one can apply this technique recursively to compute higher-order derivatives.
Derivatives of B-Spline Curves

Lemma 136

Let $\mathcal{P}$ be a B-spline curve of degree $k$ defined by $n + 1$ control points with position vectors $p_0, p_1, \ldots, p_n$ and the clamped knot vector $\tau := (t_0, t_1, \ldots, t_{n+k+1})$, for $n \in \mathbb{N}$ and $k \in \mathbb{N}_0$ with $k \leq n$. Then, for the new knot vector $\tau' := (t_1, t_2, \ldots, t_{n+k-1}, t_{n+k})$,

$$\mathcal{P}'(t) = \sum_{i=0}^{n-1} N_{i, k-1, \tau'}(t)q_i \quad \text{for } t \in [t_k, t_{n+1}[, $$

where

$$q_i := \frac{k}{t_{i+k+1} - t_{i+1}}(p_{i+1} - p_i) \quad \text{for } i \in \{0, 1, \ldots, n - 1\}. $$

Sketch of Proof: One can show that $N_{i+1, k-1, \tau}(t)$ is equal to $N_{i, k-1, \tau'}(t)$ for all $t \in [t_k, t_{n+1}[$. 

\[ \square \]
Derivatives of B-Spline Curves

Corollary 137

A clamped B-spline curve is tangent to the first leg and the last leg of its control polygon.

Sketch of Proof: Recall that, by Lem. 136, the first derivative of a clamped B-spline curve $\mathcal{P}$ of degree $k$ is a clamped B-spline curve of degree $k - 1$ over essentially the same knot vector but with new control points of the form

$$q_i := \frac{k}{t_{i+k+1} - t_{i+1}}(p_{i+1} - p_i) \quad \text{for } i \in \{0, 1, \ldots, n - 1\}.$$

Hence, by arguments similar to those used in the proof of Lem. 130, one can show that $\mathcal{P}'(t_k)$ starts in $q_0$ and, thus, the tangent of $\mathcal{P}$ in the start point $\mathcal{P}(t_k)$ is parallel to $p_1 - p_0$. \qed
Lemma 138 (Strong convex hull property)

Let \( \mathcal{P} \) be a B-spline curve of degree \( k \) defined by \( n + 1 \) control points with position vectors \( p_0, p_1, \ldots, p_n \) and the knot vector \( \tau := (t_0, t_1, \ldots, t_{n+k+1}) \), for \( n \in \mathbb{N} \) and \( k \in \mathbb{N}_0 \) with \( k \leq n \). For \( i \in \mathbb{N} \) with \( k \leq i \leq n \), we have

\[
\mathcal{P}|_{[t_i, t_{i+1}]} \subset \text{CH}(\{p_{i-k}, p_{i-k+1}, \ldots, p_{i-1}, p_i\}).
\]

Proof: Lemma 121 tells us that \( N_{i-k,k}, N_{i-k+1,k}, \ldots, N_{i-1,k}, N_{i,k} \) are the only B-spline basis functions that can be non-zero over \( [t_i, t_{i+1}] \), for \( k \leq i \leq n \), while all other basis functions are zero (Lem. 119). Together with Cor. 127, Partition of Unity, we get

\[
1 = \sum_{j=0}^{n} N_{j,k}(t) = \sum_{j=i-k}^{i} N_{j,k}(t) \quad \text{for all} \ t \in [t_i, t_{i+1}].
\]

Hence,

\[
\mathcal{P}(t) = \sum_{j=0}^{n} N_{j,k}(t)p_j = \sum_{j=i-k}^{i} N_{j,k}(t)p_j \quad \text{for all} \ t \in [t_i, t_{i+1}]
\]

is a convex combination of \( p_{i-k}, p_{i-k+1}, \ldots, p_{i-1}, p_i \).
**Strong Convex Hull Property**

**Lemma 138 (Strong convex hull property)**

Let $\mathcal{P}$ be a B-spline curve of degree $k$ defined by $n + 1$ control points with position vectors $p_0, p_1, \ldots, p_n$ and the knot vector $\tau := (t_0, t_1, \ldots, t_{n+k+1})$, for $n \in \mathbb{N}$ and $k \in \mathbb{N}_0$ with $k \leq n$. For $i \in \mathbb{N}$ with $k \leq i \leq n$, we have

$$\mathcal{P}|_{[t_i, t_{i+1}]} \subset \text{CH}(\{p_{i-k}, p_{i-k+1}, \ldots, p_{i-1}, p_i\}).$$

Second knot span of a cubic B-spline contained in $\text{CH}(\{p_1, p_2, p_3, p_4\})$. 
Local Control and Modification

Lemma 139 \((\text{Local control})\)

Let \( \mathcal{P} \) be a B-spline curve of degree \( k \) defined by \( n + 1 \) control points with position vectors \( p_0, p_1, \ldots, p_n \) and the knot vector \( \tau := (t_0, t_1, \ldots, t_{n+k+1}) \), for \( n \in \mathbb{N} \) and \( k \in \mathbb{N}_0 \) with \( k \leq n \). Then the B-spline curve \( \mathcal{P} \) restricted to \( [t_i, t_{i+1}] \) depends only on the positions of \( p_{i-k}, p_{i-k+1}, \ldots, p_{i-1}, p_i \).

\textbf{Proof:} By Lem. 121, and as in the proof of Lem. 138,

\[
\mathcal{P}|_{[t_i, t_{i+1}]}(t) = \sum_{j=i-k}^{i} N_{j,k}(t)p_j.
\]

Lemma 140 \((\text{Local modification scheme})\)

Let \( \mathcal{P} \) be a B-spline curve of degree \( k \) defined by \( n + 1 \) control points with position vectors \( p_0, p_1, \ldots, p_n \) and the knot vector \( \tau := (t_0, t_1, \ldots, t_{n+k+1}) \), for \( n \in \mathbb{N} \) and \( k \in \mathbb{N}_0 \) with \( k \leq n \). Then a modification of the position of \( p_i \) changes \( \mathcal{P} \) only in the parameter interval \( [t_i, t_{i+k+1}] \), for \( i \in \{0, 1, \ldots, n\} \).

\textbf{Proof:} The Local Support Lemma 118 tells us that

\[
N_{i,k}(t) = 0 \quad \text{if } t \notin [t_i, t_{i+k+1}].
\]
Local Control and Modification

- Clamped uniform B-spline of degree three with knot vector
  \[ \tau := (0, 0, 0, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 11, 11, 11) \]
  for a control polygon with 14 vertices:
  \[ \left\{ \left( \begin{array}{c} 1 \\ 1 \end{array} \right), \left( \begin{array}{c} 1 \\ 3 \end{array} \right), \left( \begin{array}{c} 3 \\ 5 \end{array} \right), \left( \begin{array}{c} 5 \\ 6 \end{array} \right), \left( \begin{array}{c} 5 \\ 2 \end{array} \right), \left( \begin{array}{c} 3 \\ 1 \end{array} \right), \left( \begin{array}{c} 11 \\ 8 \end{array} \right), \left( \begin{array}{c} 8 \\ 5 \end{array} \right), \left( \begin{array}{c} 10 \\ 6 \end{array} \right), \left( \begin{array}{c} 4 \\ 7 \end{array} \right), \left( \begin{array}{c} 1 \\ 5 \end{array} \right) \right\} \]
Local Control and Modification

- Clamped uniform B-spline of degree three with knot vector
  \[ \tau := (0, 0, 0, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 11, 11, 11) \]
  for a control polygon with 14 vertices:
  \[ \{(1, 1), (1, 3), (3, 5), (5, 6), (5, 2), (3, 4), (1, 1), (1, 3), (5, 8), (8, 8), (10, 6), (4, 7), (1, 5)\} \]
Multiple Control Points

Lemma 141 (Multiple control points)

Let $\mathcal{P}$ be a B-spline curve of degree $k$ defined by $n + 1$ control points with position vectors $p_0, p_1, \ldots, p_n$ and the knot vector $\tau := (t_0, t_1, \ldots, t_{n+k+1})$, for $n \in \mathbb{N}$ and $k \in \mathbb{N}_0$ with $k \leq n$.

1. If $k$ control points $p_{i-k+1}, p_{i-k+2}, \ldots, p_i$ coincide, i.e., if $p_{i-k+1} = p_{i-k+2} = \ldots = p_i$ then $\mathcal{P}$ contains $p_i$ and is tangent to the legs $p_{i-k}p_{i-k+1}$ and $p_ip_{i+1}$ of the control polygon, for $i \in \mathbb{N}$ with $k \leq i < n$.

2. If $k$ control points $p_{i-k+1}, p_{i-k+2}, \ldots, p_i$ are collinear then $\mathcal{P}$ touches a leg of the control polygon, for $i \in \mathbb{N}$ with $k \leq i < n$.

3. If $k + 1$ control points $p_{i-k}, p_{i-k+1}, \ldots, p_i$ are collinear then $\mathcal{P}$ coincides with a leg of the control polygon, for $i \in \mathbb{N}$ with $k < i < n$.

Sketch of Proof: This is a consequence of Lemma 121 and of the Strong Convex Hull Property (Lem. 138).

Note that this implies that a degree-$k$ B-spline $\mathcal{P}$ starts at $p_0$ if $p_0 = p_1 = \ldots = p_{k-1}$.
Multiple Control Points

Clamped cubic B-spline with control points

\[
\begin{pmatrix}
2 \\ 0 \\
0 \\ 4 \\
4 \\ 5 \\
4 \\ 8 \\
8 \\ 2 \\
8 \\ 8 \\
6 \\ 0
\end{pmatrix}, \begin{pmatrix}
0 \\ 4 \\
4 \\ 8 \\
5 \\ 4 \\
8 \\ 2 \\
8 \\ 8 \\
6 \\ 0
\end{pmatrix}
\]

and uniform knot vector \(0, 0, 0, 0, 1, 2, 3, 3, 3, 3\):
Multiple Control Points

Clamped cubic B-spline with control points

\[
\begin{pmatrix} 2 \\ 0 \\ 4 \\ 5 \\ 4 \\ 5 \\ 4 \\ 8 \\ 8 \\ 2 \\ 0 \end{pmatrix}
\]

and uniform knot vector \((0, 0, 0, 0, 1, 2, 3, 4, 5, 5, 5, 5)\):
Multiple Control Points

Clamped cubic B-spline with control points

\[
\begin{pmatrix}
2 \\
0 \\
4 \\
5 \\
6 \\
8 \\
8 \\
2 \\
6
\end{pmatrix},
\begin{pmatrix}
0 \\
4 \\
5 \\
\frac{9}{2} \\
4 \\
8 \\
8 \\
2 \\
0
\end{pmatrix}
\]

and uniform knot vector \((0, 0, 0, 0, 1, 2, 3, 4, 4, 4, 4)\):

![Diagram of a cubic B-spline curve with control points labeled as described in the text. The curve is a smooth arc connecting the control points, illustrating the geometric modeling concept.](image-url)
Multiple Control Points

Clamped cubic B-spline with control points

\[
\begin{pmatrix}
2 \\
0
\end{pmatrix}, \begin{pmatrix}
0 \\
4
\end{pmatrix}, \begin{pmatrix}
4 \\
5
\end{pmatrix}, \begin{pmatrix}
5 \\
\frac{19}{4}
\end{pmatrix}, \begin{pmatrix}
7 \\
\frac{17}{4}
\end{pmatrix}, \begin{pmatrix}
8 \\
4
\end{pmatrix}, \begin{pmatrix}
8 \\
2
\end{pmatrix}, \begin{pmatrix}
6 \\
0
\end{pmatrix}
\]

and uniform knot vector \((0, 0, 0, 0, 1, 2, 3, 4, 5, 5, 5, 5)\):

![Control Points Diagram](Image)
**Lemma 142 (Multiple knots)**

Let $\mathcal{P}$ be a B-spline curve of degree $k$ defined by $n + 1$ control points with position vectors $p_0, p_1, \ldots, p_n$ and the knot vector $\tau := (t_0, t_1, \ldots, t_{n+k+1})$, for $n \in \mathbb{N}$ and $k \in \mathbb{N}_0$ with $k \leq n$. Let $i \in \mathbb{N}$ with $k + 1 \leq i \leq n - k$. If $t_i$ is a knot of multiplicity $k$, i.e., if $t_i = t_{i+1} = \ldots = t_{i+k-1}$ then $\mathcal{P}(t_i) = p_{i-1}$ and $\mathcal{P}$ is tangent to the legs $\overline{p_{i-2}p_{i-1}}$ and $\overline{p_{i-1}p_i}$ of the control polygon.
Multiple Knots

- Clamped uniform B-spline of degree three for a control polygon with nine vertices:

\[
\left\{ \left( \begin{array}{c} -1 \\ 0 \end{array} \right), \left( \begin{array}{c} 0 \\ 0 \end{array} \right), \left( \begin{array}{c} 0 \\ 2 \end{array} \right), \left( \begin{array}{c} 2 \\ 3 \end{array} \right), \left( \begin{array}{c} 4 \\ 0 \end{array} \right), \left( \begin{array}{c} 6 \\ 3 \end{array} \right), \left( \begin{array}{c} 8 \\ 2 \end{array} \right), \left( \begin{array}{c} 8 \\ 0 \end{array} \right), \left( \begin{array}{c} 9 \\ 0 \end{array} \right) \end{array} \right\}
\]

Knot vector:

\[\tau := (0, 0, 0, 0, 1, 2, 3, 4, 5, 6, 6, 6, 6)\]
Multiple Knots

- Clamped uniform B-spline of degree three for a control polygon with nine vertices:

\[
\left\{ \left( \begin{array}{c} -1 \\ 0 \end{array} \right), \left( \begin{array}{c} 0 \\ 0 \end{array} \right), \left( \begin{array}{c} 2 \\ 2 \end{array} \right), \left( \begin{array}{c} 3 \\ 3 \end{array} \right), \left( \begin{array}{c} 4 \\ 2 \end{array} \right), \left( \begin{array}{c} 6 \\ 0 \end{array} \right), \left( \begin{array}{c} 8 \\ 0 \end{array} \right), \left( \begin{array}{c} 8 \\ 0 \end{array} \right), \left( \begin{array}{c} 9 \\ 0 \end{array} \right) \right\}
\]

Knot vector:

\[ \tau := (0, 0, 0, 0, 1, 2, 2, 2, 3, 4, 4, 4, 4) \]
Motivation for de Boor’s Algorithm

- Can we express $P(t)$ in terms of $N_{i,0}(t)$?
- We exploit the recursive definition of $N_{i,k}$ in order to determine $P(t)$ in terms of $N_{i,k-1}(t)$, recalling that $t \in [t_k, t_{n+1}]$.

\[
P(t) = \sum_{i=0}^{n} N_{i,k}(t)p_i = \sum_{i=0}^{n} \left( \frac{t - t_i}{t_{i+k} - t_i} N_{i,k-1}(t) + \frac{t_{i+k+1} - t}{t_{i+k+1} - t_{i+1}} N_{i+1,k-1}(t) \right) p_i
\]

\[
= \sum_{i=0}^{n} \frac{t - t_i}{t_{i+k} - t_i} N_{i,k-1}(t)p_i + \sum_{i=0}^{n} \frac{t_{i+k+1} - t}{t_{i+k+1} - t_{i+1}} N_{i+1,k-1}(t)p_i
\]

\[
= \frac{t - t_0}{t_k - t_0} N_{0,k-1}(t)p_0 + \sum_{i=1}^{n} \frac{t - t_i}{t_{i+k} - t_i} N_{i,k-1}(t)p_i + \frac{t_{n+k+1} - t}{t_{n+k+1} - t_{n+1}} N_{n+1,k-1}(t)p_n + \sum_{i=0}^{n-1} \frac{t_{i+k+1} - t}{t_{i+k+1} - t_{i+1}} N_{i+1,k-1}(t)p_i
\]

\[
= \sum_{i=1}^{n} \frac{t - t_i}{t_{i+k} - t_i} N_{i,k-1}(t)p_i + \sum_{i=1}^{n} \frac{t_{i+k} - t}{t_{i+k} - t_i} N_{i,k-1}(t)p_{i-1}
\]

\[
= \sum_{i=1}^{n} N_{i,k-1}(t) \left( \frac{t_{i+k} - t}{t_{i+k} - t_i} p_{i-1} + \frac{t - t_i}{t_{i+k} - t_i} p_i \right) =: \sum_{i=1}^{n} N_{i,k-1}(t)p_{i,1}(t)
\]
Motivation for de Boor's Algorithm

Equality at $\star$ holds since each basis function $N_{i,k}$ is non-zero only over $[t_i, t_{i+k+1}]$ (Local Support Lem. 118):

\[ N_{0,k-1,\tau}(t) = 0 \text{ for } t \notin [t_0, t_k] \quad \text{and} \quad N_{n+1,k-1,\tau}(t) = 0 \text{ for } t \notin [t_{n+1}, t_{n+k+1}] \]

For $1 \leq i \leq n$, we have

\[ p_{i,1}(t) := (1 - \alpha_{i,1}) \ p_{i-1} + \alpha_{i,1} \ p_i \quad \text{with } \alpha_{i,1} := \frac{t - t_i}{t_{i+k} - t_i}, \]

thus expressing $P(t)$ in terms of basis functions of degree $k - 1$ and modified (parameter-dependent!) new control points.

Repeating this process yields

\[ P(t) = \sum_{i=2}^{n} N_{i,k-2}(t)p_{i,2}(t), \]

where, for $2 \leq i \leq n$,

\[ p_{i,2}(t) := (1 - \alpha_{i,2}) \ p_{i-1,1}(t) + \alpha_{i,2} \ p_{i,1}(t) \quad \text{with } \alpha_{i,2} := \frac{t - t_i}{t_{i+k-1} - t_i}, \]
De Boor's Algorithm

Theorem 143 (de Boor's algorithm)

Let \( \mathcal{P} \) be a B-spline curve of degree \( k \) with control points \( p_0, p_1, \ldots, p_n \) and knot vector \( \tau := (t_0, t_1, \ldots, t_{n+k+1}) \). If we define

\[
p_{i,j}(t) := \begin{cases} 
    p_i & \text{if } j = 0, \\
    (1 - \alpha_{i,j}) p_{i-1,j-1}(t) + \alpha_{i,j} p_{i,j-1}(t) & \text{if } j > 0,
\end{cases}
\]

where

\[
\alpha_{i,j} := \frac{t - t_i}{t_i + k + 1 - j - t_i},
\]

then

\[
\mathcal{P}(t) = \sum_{i=k}^{n} N_{i,0}(t) p_{i,k}(t) \quad \text{for } t \in [t_k, t_{n+1}].
\]

Corollary 144

Let \( \mathcal{P} \) be a B-spline curve of degree \( k \) with control points \( p_0, p_1, \ldots, p_n \) and knot vector \( \tau := (t_0, t_1, \ldots, t_{n+k+1}) \). If \( t \in [t_i, t_{i+1}] \), for \( i \in \{k, k+1, \ldots, n\} \), then

\[
\mathcal{P}(t) = p_{i,k}(t).
\]
Sample Run of de Boor’s Algorithm

- Clamped uniform B-spline of degree three for seven control points:
  \[
  \{ (0,0), (2,2), (4,3), (6,0), (8,2), (8,0) \}
  \]

  Knot vector with eleven knots: \( \tau := (0, 0, 0, 0, 1, 2, 3, 4, 4, 4) \).

- B-spline curve with \( p_{i,0}(0.7) \), with \( 0.7 \in [t_3, t_4] \):
  \[
  \{ (0,0), (2,2), (4,0), (6,3), (8,2), (8,0) \}
  \]
Sample Run of de Boor's Algorithm

- Clamped uniform B-spline of degree three for seven control points:

\[ \{ (0,0), (2,3), (4,0), (6,3), (8,2), (8,0) \} \]

Knot vector with eleven knots: \( \tau := (0,0,0,0,1,2,3,4,4,4,4) \).

- B-spline curve with \( p_{i,1}(0.7) \), with \( 0.7 \in [t_3, t_4] \):

\[ \{ (0.14, 0.7, 2.4667), (0.7, 2.35, 2.3) \} \]
Sample Run of de Boor's Algorithm

- Clamped uniform B-spline of degree three for seven control points:

\[
\left\{ \begin{pmatrix} 0 \\ 0 \\ 2 \\ 3 \\ 0 \\ 3 \\ 2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \\ 4 \\ 6 \\ 8 \\ 2 \\ 0 \end{pmatrix} \right\}
\]

Knot vector with eleven knots: \( \tau := (0, 0, 0, 0, 1, 2, 3, 4, 4, 4, 4) \).

- B-spline curve with \( p_{i,2}(0.7) \), with \( 0.7 \in [t_3, t_4] \):

\[
\left\{ \begin{pmatrix} 0.49 \\ 2.065 \end{pmatrix}, \begin{pmatrix} 1.3183 \\ 2.3325 \end{pmatrix} \right\}
\]
Sample Run of de Boor’s Algorithm

- Clamped uniform B-spline of degree three for seven control points:

\[
\left\{ \begin{pmatrix} 0 \\ 0 \\ 2 \\ 3 \\ 4 \\ 6 \\ 8 \\ 8 \end{pmatrix} \right\}
\]

Knot vector with eleven knots: \( \tau := (0, 0, 0, 0, 1, 2, 3, 4, 4, 4, 4) \).

- B-spline curve with \( p_{i,3}(0.7) \), with \( 0.7 \in [t_3, t_4] \):

\[
\left\{ \begin{pmatrix} 1.0698 \\ 2.2523 \end{pmatrix} \right\} = \mathcal{P}(0.7)
\]
De Boor’s Algorithm for Subdividing a B-Spline Curve

- Clamped uniform B-spline of degree three for seven control points:

\[
\{ (0,0), (0,2), (2,3), (4,0), (6,3), (8,2), (8,0) \}
\]

Knot vector with eleven knots: \( \tau := (0,0,0,0,1,2,3,4,4,4,4) \).

- New control polygons for \( t^* := 0.7 \):

\[
(p_{0,0}, p_{1,1}, p_{2,2}, p_{3,3}) \quad \text{and} \quad (p_{3,3}, p_{3,2}, p_{3,1}, p_{3,0}, p_{4,0}, p_{5,0}, p_{6,0})
\]

- New knot vectors for \( t^* := 0.7 \):

\[
(0,0,0,0,0.7,0.7,0.7,0.7) \quad \text{and} \quad (0.7,0.7,0.7,0.7,1,2,3,4,4,4,4)
\]
De Boor’s Algorithm for Subdividing a B-Spline Curve

Definition 145

Let \( \mathcal{P} \) be a clamped B-spline curve of degree \( k \) defined by \( n + 1 \) control points with position vectors \( p_0, p_1, \ldots, p_n \) and the clamped knot vector \( \tau := (t_0, t_1, \ldots, t_{n+k+1}) \), for \( n \in \mathbb{N} \) and \( k \in \mathbb{N}_0 \) with \( k \leq n \). For some \( t^* \in [t_i, t_{i+1}[ \), with \( i \in \{k, \ldots, n\} \), we define two new knot vectors \( \tau^*, \tau^{**} \) and two new control polygons \( P^*, P^{**} \) as follows:

If \( t^* \neq t_i \) then \( m := i \) else \( m := i - 1 \).

\[
\begin{align*}
\tau^* &:= (t_0, t_1, \ldots, t_m, t^*, \ldots, t^*) \quad \text{and} \quad \tau^{**} := (t^*, \ldots, t^*, t_{m+1}, \ldots, t_{n+k+1}), \\
P^*(t^*) &:= (p_{0,0}(t^*), p_{1,0}(t^*), \ldots, p_{m-k,0}(t^*), p_{1,1}(t^*), p_{2,2}(t^*), \ldots, p_{k,k}(t^*)), \\
P^{**}(t^*) &:= (p_{k,k}(t^*), p_{k,k-1}(t^*), \ldots, p_{k,1}(t^*), p_{m,0}(t^*), p_{m+1,0}(t^*), \ldots, p_{n,0}(t^*)),
\end{align*}
\]

where the new control points \( p_{i,j}(t^*) \) are obtained by de Boor’s algorithm (Thm 145).
De Boor’s Algorithm for Subdividing a B-Spline Curve

Lemma 146

Let \( P \) be a clamped B-spline curve of degree \( k \) defined by \( n + 1 \) control points with position vectors \( p_0, p_1, \ldots, p_n \) and the clamped knot vector \( \tau := (t_0, t_1, \ldots, t_{n+k+1}) \), for \( n \in \mathbb{N} \) and \( k \in \mathbb{N}_0 \) with \( k \leq n \). (Hence, \( t_0 = t_1 = \ldots = t_k < t_{k+1} \) and \( t_n < t_{n+1} = \ldots = t_{n+k+1} \).) For some \( t^* \in [t_i, t_{i+1}[ \), with \( i \in \{k, \ldots, n\} \), we define two new knot vectors \( \tau^*, \tau^{**} \) and two new control polygons \( P^*, P^{**} \) as in Def. 145. Then we get two new B-spline curves \( P^* \) and \( P^{**} \) of degree \( k \) with control polygon \( P^* \) (\( P^{**} \), resp.) and knot vector \( \tau^* \) (\( \tau^{**} \), resp.) that join in a tangent-continuous way at point \( p_{kk}(t^*) = P(t^*) \), such that

\[
P^* = P|_{[t_k,t^*[} \quad \text{and} \quad P^{**} = P|_{[t^*,t_{n+1}[}.
\]
De Boor’s Algorithm for Splitting a B-Spline Curve into Bézier Segments

Corollary 147

Let \( \mathcal{P} \) be a clamped B-spline curve of degree \( k \) defined by \( n + 1 \) control points with position vectors \( p_0, p_1, \ldots, p_n \) and the clamped knot vector \( \tau := (t_0, t_1, \ldots, t_{n+k+1}) \), for \( n \in \mathbb{N} \) and \( k \in \mathbb{N}_0 \) with \( k \leq n \). (Hence, \( t_0 = t_1 = \ldots = t_k < t_{k+1} \) and \( t_n < t_{n+1} = \ldots = t_{n+k+1} \).) Subdividing \( \mathcal{P} \) at the knot values \( \{t_{k+1}, t_{k+2}, \ldots, t_{n-1}, t_n\} \), as outlined in Def. 145, splits \( \mathcal{P} \) into \( n - k + 1 \) Bézier curves of degree \( k \).

Sketch of Proof: Lemma 146 ensures that each of the resulting curves is a B-spline curve of degree \( k \), where the \( m \)-th curve is defined over \([t_{k+m}, t_{k+m+1}]\), for \( m \in \{0, 1, \ldots, n - k\} \). Each curve has knot vectors of length \( 2k + 2 \), with start and end knots of multiplicity \( k + 1 \) but no interior knots. After mapping \([t_{k+m}, t_{k+m+1}]\) to \([0, 1]\) we can apply Lem. 134 and conclude that the resulting B-spline curve is a Bézier curve of degree \( k \).
De Boor’s Algorithm for Splitting a B-Spline Curve into Bézier Segments

- Clamped uniform B-spline of degree three for seven control points:
  \[
  \{(0,0), (0,2), (2,3), (4,0), (6,3), (8,2), (8,0)\}
  \]
  Knot vector with eleven knots: \(\tau := (0, 0, 0, 0, 1, 2, 3, 4, 4, 4, 4)\).

- Third Bézier segment over \([2, 3]\).

- Note that the number of knots has increased drastically!
Knot Insertion

- Suppose that we would like to insert a new knot \( t^* \in [t_j, t_{j+1}] \), for some \( j \in \{k, k+1, \ldots, n\} \), into the knot vector

\[
\tau := (t_0, t_1, \ldots, t_j, t_{j+1}, \ldots, t_{n+k+1}),
\]

thus transforming \( \tau \) into a knot vector

\[
\tau^* := (t_0, t_1, \ldots, t_j, t^*, t_{j+1}, \ldots, t_{n+k+1}).
\]

- The fundamental equality \( m = n + k + 1 \), with \( m + 1 \) denoting the number of knots, tells us that we will have to either increase the number \( n \) of control points by one or to increase the degree \( k \) of the curve by one.

- Since an increase of the degree would change the shape of the B-spline globally, we opt for increasing the number of control points (and modifying some of them).

- How can we modify the control points such that the shape of the curve is preserved?
Knot Insertion

Lemma 148 \( (Boehm 1980) \)

Let \( \mathcal{P} \) be a B-spline curve of degree \( k \) defined by \( n + 1 \) control points with position vectors \( p_0, p_1, \ldots, p_n \) and the knot vector \( \tau := (t_0, t_1, \ldots, t_{n+k+1}) \), for \( n \in \mathbb{N} \) and \( k \in \mathbb{N}_0 \) with \( k \leq n \). Let \( t^* \in [t_j, t_{j+1}[ \), for some \( j \in \{k, k + 1, \ldots, n\} \), and define a knot vector \( \tau^* \) as \( \tau^* := (t_0, t_1, \ldots, t_j, t^*, t_{j+1}, \ldots, t_{n+k+1}) \). Then we have

\[
\mathcal{P}(t) = \sum_{i=0}^{n} N_{i,k,\tau}(t)p_i = \sum_{i=0}^{n+1} N_{i,k,\tau^*}(t)p_i^* =: \mathcal{P}^*(t) \quad \text{for all } t \in [t_k, t_{n+1}[
\]

if, for \( 0 \leq i \leq n + 1 \),

\[
p_i^* := \begin{cases} 
p_i & \text{if } i \leq j - k \\
(1 - \alpha_i)p_{i-1} + \alpha_ip_i & \text{if } j - k + 1 \leq i \leq j \\
p_{i-1} & \text{if } i \geq j + 1
\end{cases}
\]

and

\[
\alpha_i := \frac{t^* - t_i}{t_{i+k} - t_i} \quad \text{for } i \in \{j - k + 1, \ldots, j\}.
\]
Knot Insertion: Sample

- Clamped uniform B-spline of degree three for 14 control points and knot vector with 18 knots: \( \tau := (0, 0, 0, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 11, 11, 11) \).

\[ \{(1), (1), (3), (5), (5), (6), (2), (3), (11), (8), (8), (10), (4), (5)\} \]

- New 15 control points for 19 knots

\( \tau^* := (0, 0, 0, 0, 1, 2, 3, 4, 5.2, 5, 6, 7, 8, 9, 10, 11, 11, 11, 11) \):

\[ \{(1), (1), (3), (5), (5), (6), (2), (3), (1), (8), (8), (10), (4), (5)\} \]
Knot Insertion: Sample

▶ Old 18 knots:

<table>
<thead>
<tr>
<th>t₀</th>
<th>t₁</th>
<th>t₂</th>
<th>t₃</th>
<th>t₄</th>
<th>t₅</th>
<th>t₆</th>
<th>t₇</th>
<th>t₈</th>
<th>t₉</th>
<th>t₁₀</th>
<th>t₁₁</th>
<th>t₁₂</th>
<th>t₁₃</th>
<th>t₁₄</th>
<th>t₁₅</th>
<th>t₁₆</th>
<th>t₁₇</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
</tbody>
</table>

\[ t₀ = t₁ = t₂ = t₃ \quad t₄ \quad t₅ \quad t₆ \quad t₇ \quad t₈ \quad t₉ \quad t₁₀ \quad t₁₁ \quad t₁₂ \quad t₁₃ \quad t₁₄ = t₁₅ = t₁₆ = t₁₇ \]

\[ t^* = 4.2 \]

\[ \alpha₅ := \frac{t^* - t₅}{t₈ - t₅} = \frac{2\frac{1}{5}}{3} = \frac{11}{15} \quad p₅^* = (1 - \alpha₅)p₄ + \alpha₅p₅ = \frac{1}{15} \begin{pmatrix} 79 \\ 38 \end{pmatrix} \approx \begin{pmatrix} 5.2667 \\ 2.5333 \end{pmatrix} \]

\[ \alpha₆ := \frac{t^* - t₆}{t₉ - t₆} = \frac{1\frac{1}{5}}{3} = \frac{6}{15} \quad p₆^* = (1 - \alpha₆)p₅ + \alpha₆p₆ = \frac{1}{15} \begin{pmatrix} 63 \\ 30 \end{pmatrix} \approx \begin{pmatrix} 4.2 \\ 2 \end{pmatrix} \]

\[ \alpha₇ := \frac{t^* - t₇}{t₁₀ - t₇} = \frac{1\frac{1}{5}}{3} = \frac{1}{15} \quad p₇^* = (1 - \alpha₇)p₆ + \alpha₇p₇ = \frac{1}{15} \begin{pmatrix} 45 \\ 29 \end{pmatrix} \approx \begin{pmatrix} 3 \\ 1.9333 \end{pmatrix} \]

▶ For \( t^* := 4.2 = 4\frac{1}{5} \) we have \( j = 7 \) and \( j - k + 1 = 5 \), and get:
Knot Insertion and Deletion

- The so-called Oslo algorithm, developed by Cohen et al. [1980], is more general than Boehm’s algorithm: It allows the insertion of several (possibly multiple) knots into a knot vector. (It is also substantially more complex, though.)

- An algorithm for the removal of a knot is due to Tiller [1992]. However, as pointed out by Tiller, knot removal and degree reduction result in an overspecified problem which, in general, can only be solved within some tolerance.
Closed B-Spline Curves

Lemma 149

Let \( \mathcal{P} \) be a B-spline curve of degree \( k \) defined by \( n+1 \) control points with position vectors \( p_0, p_1, \ldots, p_n \) and the uniform knot vector \( \tau := (t_0, t_1, \ldots, t_{n+k+1}) \), for \( n \in \mathbb{N} \) and \( k \in \mathbb{N}_0 \) with \( k \leq n \). If

\[
p_0 = p_{n-k+1}, \quad p_1 = p_{n-k+2}, \ldots, \quad p_{k-2} = p_{n-1}, \quad p_{k-1} = p_n
\]

then \( \mathcal{P} \) is \( C^{k-1} \) at the joining point \( \mathcal{P}(t_k) = \mathcal{P}(t_{n+1}) \).

Lemma 150

Let \( \mathcal{P} \) be a B-spline curve of degree \( k \) defined by \( n+2 \) control points with position vectors \( p_0, p_1, \ldots, p_n, p_{n+1} \) and the (possibly non-uniform and periodic) knot vector \( \tau := (t_0, t_1, \ldots, t_{n+k+1}, t_{n+k+2}) \), for \( n \in \mathbb{N} \) and \( k \in \mathbb{N}_0 \) with \( k \leq n \). If

\[
p_0 = p_{n+1} \quad \text{and} \quad t_0 = t_{n+1}, \quad t_1 = t_{n+2}, \ldots, \quad t_k = t_{n+k+1}, \quad t_{k+1} = t_{n+k+2}
\]

then \( \mathcal{P} \) is \( C^{k-1} \) at the joining point \( \mathcal{P}(t_0) = \mathcal{P}(t_{n+1}) \).

\[\blacktriangleright\] Hence, wrapping around \( k \) control points or \( k + 2 \) knots achieves \( C^{k-1} \)-continuity at the joining point.
Closed B-Spline Curves via Wrapping of Control Points

Uniform B-spline of degree three for nine control points:
\[
\left\{ \begin{pmatrix} 4 \\ 3 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 4 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 7 \\ 7 \\ 2 \end{pmatrix}, \begin{pmatrix} 12 \\ 12 \\ 4 \\ 3 \end{pmatrix}, \begin{pmatrix} 4 \\ 0 \\ 4 \\ 0 \end{pmatrix} \right\}
\]

Knot vector with 13 knots: \( \tau := (0, \frac{1}{12}, \frac{2}{12}, \frac{3}{12}, \frac{4}{12}, \frac{5}{12}, \frac{6}{12}, \frac{7}{12}, \frac{8}{12}, \frac{9}{12}, \frac{10}{12}, \frac{11}{12}, 1) \).
B-Spline Surfaces

Definition 151 (B-spline surface)

For \( n, m \in \mathbb{N} \) and \( k\', k'' \in \mathbb{N}_0 \) with \( k' \leq n \) and \( k'' \leq m \), consider a set of \((n + 1) \times (m + 1)\) control points with position vectors \( p_{i,j} \in \mathbb{R}^3 \) for \( 0 \leq i \leq n \) and \( 0 \leq j \leq m \), and let \( \sigma := (s_0, s_1, \ldots, s_{n+k'+1}) \) and \( \tau := (t_0, t_1, \ldots, t_{m+k''+1}) \) be two knot vectors. Then the \textit{B-spline surface} relative to \( \sigma \) and \( \tau \) with control net \( (p_{i,j})_{i,j=0}^{n,m} \) is given by

\[
S(s, t) := \sum_{i=0}^{n} \sum_{j=0}^{m} N_{i,k'}, \sigma (s) N_{j,k''}, \tau (t) p_{i,j} \quad \text{for} \quad s \in [s_{k'}, s_{n+1}], \ t \in [t_{k''}, t_{m+1}],
\]

where \( N_{i,k'}, \sigma \) is the \( i \)-th B-spline basis function of degree \( k' \) relative to \( \sigma \), and \( N_{j,k''}, \tau \) is the \( j \)-th B-spline basis function of degree \( k'' \) relative to \( \tau \).

\( \blacktriangleright \) Hence, a B-spline surface is another example of a tensor-product surface.
Sample B-Spline Surfaces
Properties of B-Spline Surfaces

Lemma 152 (Non-negativity)
With the setting of Def. 151, we have \( N_{i,k',\sigma}(s)N_{j,k'',\tau}(t) \geq 0 \) for all (permissible) \( i, j \in \mathbb{Z} \) and \( k', k'' \in \mathbb{N}_0 \), and all real \( s, t \).

Lemma 153 (Partition of unity)
With the setting of Def. 151, we have
\[
\sum_{i=0}^{n} \sum_{j=0}^{m} N_{i,k',\sigma}(s)N_{j,k'',\tau}(t) = 1
\]
for all \( s \in [s_{k'}, s_{n+1}] \), \( t \in [t_{k''}, t_{m+1}] \).

Lemma 154 (Strong convex hull property)
With the setting of Def. 151, for \( i, j \in \mathbb{N} \) with \( k' \leq i \leq n \) and \( k'' \leq j \leq m \) we have
\[
S|_{[s_i, s_{i+1}] \times [t_j, t_{j+1}]} \subset \text{CH}(\{p_{l',l''} : i - k' \leq l' \leq i \land j - k'' \leq l'' \leq j\}).
\]
Properties of B-Spline Surfaces

Lemma 155 (Local control)
With the setting of Def. 151, for $i, j \in \mathbb{N}$ with $k' \leq i \leq n$ and $k'' \leq j \leq m$ we have that
\[
S|_{[s_i, s_{i+1}] \times [t_j, t_{j+1}]} \text{ depends only on } \{p_{i', j'} : i - k' \leq i' \leq i \land j - k'' \leq j' \leq j\}.
\]

Lemma 156 (Local modification scheme)
With the setting of Def. 151, a modification of the position of $p_{i,j}$ changes $S$ only in the parameter domain $[s_i, s_{i+k'+1}] \times [t_j, t_{j+k''+1}]$, for $i \in \{0, 1, \ldots, n\}$ and $j \in \{0, 1, \ldots, m\}$.

Lemma 157 (Affine invariance)
Any B-spline representation is affinely invariant, i.e., given any affine map $\pi$, the image surface $\pi(S)$ of a B-spline surface $S$ with control points $p_{i,j}$ has the control points $\pi(p_{i,j})$. 
Clamping of a B-Spline Surface

- A B-spline surface can be clamped by repeating the same knot values in one direction of the parameters (i.e., in s or t).
- We can also close the surface by recycling the control points.
- If a B-spline surface is closed in one direction, then the surface becomes a tube.
- If a B-spline surface is closed in two directions, then the surface becomes a torus.
- Other topologies are more difficult to handle, such as a ball or a double torus.
Evaluation of a B-Spline Surface

► Five easy steps to calculate a point on a B-spline patch for \((s, t)\)

1. Find the knot span in which \(s\) lies, i.e., find \(i\) such that \(s \in [s_i, s_{i+1}]\).
2. Evaluate the non-zero basis functions \(N_{i-k'}, k'(s), \ldots, N_{i, k'}(s)\).
3. Find the knot span in which \(t\) lies, i.e., find \(j\) such that \(t \in [t_j, t_{j+1}]\).
4. Evaluate the non-zero basis functions \(N_{j-k'', k''}(t), \ldots, N_{j, k''}(t)\).
5. Multiply \(N_{i', k'}(s)\) with \(N_{j', k''}(t)\) and with the control point \(p_{i', j'}\), for \(i' \in \{i - k', \ldots, i\}\) and \(j' \in \{j - k'', \ldots, j\}\).

► Alternatively, one can apply an appropriate generalization of de Boor's algorithm.
Motivation

- Can we use a B-spline curve to represent a circular arc?

\[
\left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\}
\]

- uniform knots, degree 10

- close to a circle, but still no circle!
Definition 158 (NURBS curve)

For \( n \in \mathbb{N} \) and \( k \in \mathbb{N}_0 \) with \( k \leq n \), consider a set of \( n + 1 \) control points with position vectors \( p_0, p_1, \ldots, p_n \) in the plane, and let \( \tau := (t_0, t_1, \ldots, t_{n+k+1}) \) be a knot vector. Then the (non-uniform) rational B-spline curve of degree \( k \) (and order \( k + 1 \)) relative to \( \tau \) with control points \( p_0, p_1, \ldots, p_n \) is given by

\[
\mathcal{N}(t) := \frac{\sum_{i=0}^{n} N_{i,k}(t) w_i p_i}{\sum_{i=0}^{n} N_{i,k}(t) w_i} \quad \text{for } t \in [t_k, t_{n+1}[, \\

where \( N_{i,k,\tau} \) is the \( i \)-th B-spline basis function of degree \( k \) relative to \( \tau \), with the weights \( w_i \in \mathbb{R}^+ \) for all \( i \in \{0, 1, \ldots, n\} \).

- If all \( w_i := 1 \) then we obtain the standard B-spline curve. (Recall the Partition of Unity, Cor. 127.)
- Both the numerator and the denominator are (piecewise) polynomials of degree \( k \). Hence, \( \mathcal{N} \) is a piecewise rational curve of degree \( k \).
- In general, the weights \( w_i \) are required to be positive; a zero weight effectively turns off a control point, and can be used for so-called infinite control points [Piegl 1987].
Geometric Interpretation of NURBS

We resort to homogeneous coordinates: For \( w \neq 0 \), \( \begin{pmatrix} u \\ v \\ w \end{pmatrix} \) are the homogeneous coordinates of \( \begin{pmatrix} x \\ y \end{pmatrix} \), and \( \begin{pmatrix} x \\ y \end{pmatrix} \) are the inhomogeneous coordinates of \( \begin{pmatrix} u \\ v \\ w \end{pmatrix} \iff [x = \frac{u}{w} \text{ and } y = \frac{v}{w}] \).

For \( p_i := \begin{pmatrix} x_i \\ y_i \end{pmatrix} \in \mathbb{R}^2 \), let \( p_i^w := \begin{pmatrix} w_i x_i \\ w_i y_i \\ w_i \end{pmatrix} \in \mathbb{R}^3 \), for all \( i \in \{0, 1, \ldots, n\} \).

Now consider \( \mathcal{N}^w(t) := \sum_{i=0}^{n} N_{i,k}(t) p_i^w = \begin{pmatrix} \sum_{i=0}^{n} N_{i,k}(t) w_i x_i \\ \sum_{i=0}^{n} N_{i,k}(t) w_i y_i \\ \sum_{i=0}^{n} N_{i,k}(t) w_i \end{pmatrix} \).

Dividing the first two components of \( \mathcal{N}^w \) by its third component equals the (perspective) projection of \( \mathcal{N}^w \) to the plane \( z = 1 \).

Hence, a NURBS curve in \( \mathbb{R}^d \) is the projection of a B-spline curve in \( \mathbb{R}^{d+1} \) and, thus, it inherits properties of B-spline curves.
**Geometric Interpretation of NURBS**

**Projection onto** \( z = 1 \)

A NURBS curve in \( \mathbb{R}^d \) is the projection of a B-spline curve in \( \mathbb{R}^{d+1} \).

\[
\begin{align*}
(x_0, y_0, 1) & \quad (w_0x_0, w_0y_0, w_0) \\
(x_1, y_1, 1) & \quad (w_1x_1, w_1y_1, w_1) \\
(x_2, y_2, 1) & \quad (w_2x_2, w_2y_2, w_2) \\
(x_3, y_3, 1) & \quad (w_3x_3, w_3y_3, w_3)
\end{align*}
\]

\( z = 1 \)

\( \uparrow \) **Note**: A projection tends to increase rather than decrease continuity!
Geometric Interpretation of NURBS

- Rational (inhomogeneous) parameterization of the unit circle in the plane:

\[ x(t) := \frac{1 - t^2}{1 + t^2} \]
\[ y(t) := \frac{2t}{1 + t^2} \]

with \( t \in \mathbb{R} \).

- Parameterization of the unit circle in the plane in homogeneous coordinates:

\[ u(t) := 1 - t^2 \]
\[ v(t) := 2t \]
\[ w(t) := 1 + t^2 \]
NURBS Basis Functions

Definition 159 (*NURBS basis function*)

For \( k \in \mathbb{N}_0 \), weights \( w_j > 0 \) for all \( j \in \{0, 1, \ldots, n\} \) and all (permissible) \( i \), we define the \( i \)-th *NURBS basis function* of degree \( k \) as

\[
R_{i,k}(t) := \frac{N_{i,k}(t)w_i}{\sum_{j=0}^{n} N_{j,k}(t)w_j}.
\]

We can re-write the equation for \( \mathcal{N}(t) \) as

\[
\mathcal{N}(t) = \sum_{i=0}^{n} R_{i,k}(t)p_i \quad \text{for } t \in [t_k, t_{n+1}].
\]

Since NURBS basis functions in \( \mathbb{R}^d \) are given by the projection of B-spline basis functions in \( \mathbb{R}^{d+1} \), we may expect that the properties of B-spline basis functions carry over to NURBS basis functions.
Properties of NURBS Basis Functions

Lemma 160

For $n \in \mathbb{N}$ and $k \in \mathbb{N}_0$ with $k \leq n$, let $\tau := (t_0, t_1, t_2, \ldots, t_{n+k+1})$ be a knot vector. Then the following properties hold for all (permissible) values of $i \in \mathbb{N}_0$:

**Non-negativity:**

$R_{i,k}(t) \geq 0$ for all real $t$.

**Local support:**

$R_{i,k}(t) = 0$ if $t \notin [t_i, t_{i+k+1}]$.

**Local influence:**

$R_{j,k}$ non-zero over $[t_i, t_{i+1}]$ \implies $j \in \{i - k, i - k + 1, \ldots, i\}$.

**Partition of unity:**

$$\sum_{j=0}^{n} R_{j,k}(t) = 1$$ for all $t \in [t_k, t_{n+1}]$.

**Continuity:**

All NURBS basis functions of degree $k$ are $k - r$ times continuously differentiable at a knot of multiplicity $r$, and $k - 1$ times continuously differentiable everywhere else.
Properties of NURBS Curves

Lemma 161

For \( n \in \mathbb{N} \) and \( k \in \mathbb{N}_0 \) with \( k \leq n \), consider a set of \( n + 1 \) control points with position vectors \( p_0, p_1, \ldots, p_n \) in the plane, and let \( \tau := (t_0, t_1, \ldots, t_{n+k+1}) \) be a knot vector. Then the following properties hold:

**Clamped interpolation:** If \( \tau \) is clamped then the NURBS curve \( \mathcal{N} \) starts in \( p_0 \) and ends in \( p_n \).

**Variation diminishing property:** If a straight line intersects the control polygon of \( \mathcal{N} \) \( m \) times then it intersects \( \mathcal{N} \) at most \( m \) times.

**Strong convex hull property:** For \( i \in \mathbb{N} \) with \( k \leq i \leq n \), we have

\[
\mathcal{N}|_{[t_i, t_{i+1}]} \subset \text{CH} \left( \{ p_{i-k}, p_{i-k+1}, \ldots, p_{i-1}, p_i \} \right).
\]

**Local control:** The NURBS curve \( \mathcal{N} \) restricted to \( [t_i, t_{i+1}] \) depends only on the positions of \( p_{i-k}, p_{i-k+1}, \ldots, p_{i-1}, p_i \).

**Local modification scheme:** A modification of the position of \( p_i \) changes \( \mathcal{N} \) only in the parameter interval \( [t_i, t_{i+k+1}] \), for \( i \in \{0, 1, \ldots, n\} \).
Properties of NURBS Curves

Lemma 162 (*Projective invariance*)

Any NURBS curve is projectively invariant, i.e., given any projective transformation $\pi$, the image curve $\pi(\mathcal{N})$ of a NURBS curve $\mathcal{N}$ with control points $p_0, p_1, \ldots, p_n$ has the control points $\pi(p_0), \pi(p_1), \ldots, \pi(p_n)$.

Lemma 163

For $n \in \mathbb{N}$ and $k \in \mathbb{N}_0$ with $k \leq n$, consider a set of $n + 1$ control points with position vectors $p_0, p_1, \ldots, p_n$ in the plane, and let $\tau := (t_0, t_1, \ldots, t_{n+k+1})$ be a knot vector and $w_0, w_1, \ldots, w_n$ be weights. Then the following properties hold for all $i \in \{0, 1, \ldots, n\}$:

1. The weight $w_i$ effects only the knot span $[t_i, t_{i+k+1}]$.
2. If $w_i$ decreases (relative to the other weights) then the NURBS curve is pushed away from $p_i$.
3. If $w_i = 0$ then $p_i$ does not contribute to the NURBS curve.
4. If $w_i$ increases (relative to the other weights) then the NURBS curve is pulled towards $p_i$. 
Sample NURBS Curve

- Clamped uniform rational B-spline of degree three for a control polygon with nine vertices:

\[
\begin{bmatrix}
-1 \\
0 \\
0 \\
2 \\
3 \\
4 \\
6 \\
8 \\
8 \\
9
\end{bmatrix},
\begin{bmatrix}
0 \\
0 \\
2 \\
3 \\
0 \\
3 \\
2 \\
0 \\
0 \\
0
\end{bmatrix}
\]

Knot vector:

\[\tau := (0, 0, 0, 0, 1, 2, 3, 4, 5, 6, 6, 6, 6)\]

Weights:

\[(1, 1, 1, 1, 1, 1, 1)\]
Sample NURBS Curve

- Clamped uniform rational B-spline of degree three for a control polygon with nine vertices:

\[
\left\{ \begin{pmatrix} -1 \\ 0 \\ 0 \\ 2 \\ 3 \\ 0 \\ 3 \\ 2 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 2 \\ 2 \\ 4 \\ 6 \\ 8 \\ 8 \\ 8 \\ 9 \end{pmatrix} \right\}
\]

Knot vector:

\[ \tau := (0, 0, 0, 0, 1, 2, 3, 4, 5, 6, 6, 6, 6) \]

Weights:

\((1, 1, 1, 10, 1, 1, 1)\)
Sample NURBS Curve

- Clamped uniform rational B-spline of degree three for a control polygon with nine vertices:

\[ \{ (-1, 0), (0, 2), (2, 4), (4, 6), (6, 8), (8, 8), (9, 0) \} \]

Knot vector:

\[ \tau := (0, 0, 0, 0, 1, 2, 3, 4, 5, 6, 6, 6, 6) \]

Weights:

\( (1, 1, 1, 0.1, 1, 1, 1) \)
Conics Modeled by NURBS

- NURBS can represent all conic curves — circle, ellipse, parabola, hyperbola — exactly.
- Conics are quadratic curves.
- Hence, consider three control points $p_0, p_1, p_2$ and the following quadratic NURBS curve

$$\mathcal{N}_2(t) := \frac{\sum_{i=0}^{2} N_{i,2}(t) w_i p_i}{\sum_{i=0}^{2} N_{i,2}(t) w_i}$$

with $\tau := (0, 0, 0, 1, 1, 1)$, i.e., a rational Bézier curve of degree two over $[0, 1]$.

- In expanded form we get

$$\mathcal{N}_2(t) = \frac{(1 - t^2)w_0 p_0 + 2t(1 - t)w_1 p_1 + t^2 w_2 p_2}{(1 - t^2)w_0 + 2t(1 - t)w_1 + t^2 w_2}.$$

- Can we come up with conditions for $w_0, w_1, w_2$ that allow to characterize the type of curve represented by $\mathcal{N}_2$?
Conics Modeled by NURBS

Lemma 164

The conic shape factor, \( \rho \), determines the type of conic represented by \( \mathcal{N}_2 \):

\[
\rho := \frac{w_1^2}{w_0 w_2} \begin{cases} 
< 1 & \ldots \text{ \( \mathcal{N}_2 \) is an elliptic curve,} \\
= 1 & \ldots \text{ \( \mathcal{N}_2 \) is a parabolic curve,} \\
> 1 & \ldots \text{ \( \mathcal{N}_2 \) is a hyperbolic curve.}
\end{cases}
\]

► Clamped uniform rational B-spline \( \mathcal{N}_2 \) of degree two with three control vertices

\[
\left\{ \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} \right\}
\]

and knots

\[
\tau := (0, 0, 0, 1, 1, 1)
\]

and weights:

\[
(1, 1/10, 1), \quad \text{hence } \rho < 1.
\]
Conics Modeled by NURBS

Lemma 165

The conic shape factor, \( \rho \), determines the type of conic represented by \( N_2 \):

\[
\rho := \frac{w_1^2}{w_0 w_2} \quad \begin{cases} 
< 1 & \ldots \quad N_2 \text{ is an elliptic curve,} \\
= 1 & \ldots \quad N_2 \text{ is a parabolic curve,} \\
> 1 & \ldots \quad N_2 \text{ is a hyperbolic curve.}
\end{cases}
\]

- Clamped uniform rational B-spline \( N_2 \) of degree two with three control vertices
  \[
  \left\{ \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} \right\}
  \]
  and knots
  \( \tau := (0, 0, 0, 1, 1, 1) \)
  and weights:
  \( (1, 1, 1), \) hence \( \rho = 1 \).
Conics Modeled by NURBS

Lemma 166

The conic shape factor, \( \rho \), determines the type of conic represented by \( \mathcal{N}_2 \):

\[
\rho := \frac{w_1^2}{w_0 w_2} \begin{cases} 
< 1 & \ldots \text{ \( \mathcal{N}_2 \) is an elliptic curve,} \\
= 1 & \ldots \text{ \( \mathcal{N}_2 \) is a parabolic curve,} \\
> 1 & \ldots \text{ \( \mathcal{N}_2 \) is a hyperbolic curve.}
\end{cases}
\]

- Clamped uniform rational B-spline \( \mathcal{N}_2 \) of degree two with three control vertices

\[
\binom{1}{0}, \binom{1}{1}, \binom{0}{1}
\]

and knots

\( \tau := (0, 0, 0, 1, 1, 1) \)

and weights:

\( (1, 5, 1) \), hence \( \rho > 1 \).
Lemma 167

The quadratic NURBS curve $N_2$ represents a circular arc

- if the control points $p_0, p_1, p_2$ form an isosceles triangle, and
- if the weights are set as follows:

$$w_0 := 1 \quad w_1 := \frac{\|p_0 - p_2\|}{2 \cdot \|p_0 - p_1\|} \quad w_2 := 1$$

- The weight $w_1$ is related to the central angle $\varphi$ subtended by the arc: $w_1 = \cos(\varphi/2)$.
- We can join four quarter-circle NURBS to form a full circle.
- In this case, the isosceles triangles defining the quarter circles need to add up to a square.
Conics Modeled by NURBS

- It is also possible to construct a circle by a single NURBS curve.

\[
\left\{ \begin{array}{c}
(1, 0), (1, 1), (0, 1), (-1, 0), (-1, 1), (0, -1), (1, -1), (1, 0) \\
\end{array} \right. 
\]

Knots:

\( (0, 0, 0, \frac{\pi}{2}, \frac{\pi}{2}, \pi, \frac{3\pi}{2}, \frac{3\pi}{2}, 2\pi, 2\pi, 2\pi) \)

Weights:

\( (1, \frac{1}{\sqrt{2}}, 1, \frac{1}{\sqrt{2}}, 1, \frac{1}{\sqrt{2}}, 1, \frac{1}{\sqrt{2}}, 1) \)

- Note: The positioning of the control points ensures that the first derivative is continuous, despite of double knots.

- Note: \( N'(t) \neq (\sin t, \cos t) \) for \( t \neq \frac{m \cdot \pi}{4} \).
Conics Modeled by NURBS

- Applying an affine transformation to the control points yields an ellipse.

\[
\left\{ \begin{pmatrix} 2 \\ 0 \end{pmatrix}, \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} -2 \\ 1 \end{pmatrix}, \begin{pmatrix} -2 \\ 0 \end{pmatrix}, \begin{pmatrix} -2 \\ -1 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \begin{pmatrix} 2 \\ -1 \end{pmatrix}, \begin{pmatrix} 2 \\ 0 \end{pmatrix} \right\}
\]

Knots: (0, 0, 0, 1, 1, 2, 2, 3, 3, 4, 4, 4)  Weights: (1, \( \frac{1}{\sqrt{2}} \), 1, \( \frac{1}{\sqrt{2}} \), 1, \( \frac{1}{\sqrt{2}} \), 1, \( \frac{1}{\sqrt{2}} \), 1)
Sample NURBS Surface
Approximation and Interpolation

- Distance Measures
- Interpolation of Point Data
- Bernstein Approximation of Functions
- Approximation of Polygonal Profiles
Hausdorff Distance

Let $A, B$ be two subsets of a metric space $X$ and let $d(p, q)$ denote the distance between two elements $p, q \in X$. E.g., take $\mathbb{R}^n$ and the (standard) Euclidean distance.

How can we measure how similar $A$ and $B$ are?

This is a frequently asked question in image processing, solid modeling, computer graphics and computational geometry.

Note that the classical \textit{minimin} function

$$D(A, B) := \inf_{a \in A} \left( \inf_{b \in B} d(a, b) \right)$$

is a very poor measure of similarity between $A$ and $B$: One can easily get $D(A, B) = 0$ although $A$ and $B$ need not be similar at all, according to any natural human interpretation of similarity.

So, can we do any better?
**Hausdorff Distance**

**Definition 168 (Hausdorff distance)**

Let \( A, B \) be two non-empty subsets of a metric space \( X \) and let \( d \) be any metric on \( X \). The *directed Hausdorff distance*, \( h(A, B) \), from \( A \) to \( B \) is a maximin function, defined as

\[
h(A, B) := \sup_{a \in A} \left( \inf_{b \in B} d(a, b) \right).
\]

The *(symmetric) Hausdorff distance*, \( H(A, B) \), between \( A \) and \( B \) is defined as

\[
H(A, B) := \max \{h(A, B), h(B, A)\}.
\]

- Introduced by Felix Hausdorff in 1914.
- If both \( A \) and \( B \) are bounded then \( H(A, B) \) is guaranteed to be finite.
- For compact sets we can replace \( \inf \) by \( \min \) and \( \sup \) by \( \max \).
- The function \( H \) defines a metric on the set of all non-empty compact subsets of a metric space \( X \).
- For sets of \( n \) points in \( \mathbb{R}^2 \), the Hausdorff distance can be computed in time \( O(n \log n) \), using a Voronoi-based approach \( \rightarrow \) computational geometry.
- A common variation is the *Hausdorff distance under translation*. 
Fréchet Distance

- The Hausdorff distance does not capture any form of orientation or continuity, as we might be interested in when matching curves or surfaces.

**Definition 169 (Fréchet distance)**

Consider a closed interval \( I \subset \mathbb{R} \) and two curves \( \beta, \gamma : I \rightarrow \mathbb{R}^n \). The Fréchet distance between \( \beta(I) \) and \( \gamma(I) \) is defined as

\[
\text{Fr}(\beta, \gamma) := \inf_{\sigma, \tau} \max_{t \in I} \|\beta(\sigma(t)) - \gamma(\tau(t))\|,
\]

where \( \sigma, \tau : I \rightarrow I \) range over all continuous and monotonously increasing functions that map \( I \) to \( I \) such that \( \sigma(I) = I \) and \( \tau(I) = I \).

- Popular interpretation [Alt&Godau 1995]: Suppose that a person is walking a dog. Assume the person is walking on one curve and the dog on another curve. Both can adjust their speeds but are not allowed to move backwards.
- We can think of the parameter \( t \) as time: Then \( \beta(\sigma(t)) \) is the position of the person and \( \gamma(\tau(t)) \) is the position of the dog at time \( t \). The length of the leash between them at time \( t \) is the distance between \( \beta(\sigma(t)) \) and \( \gamma(\tau(t)) \).
- Then the Fréchet distance of the two curves is the minimum leash length necessary to keep the person and the dog connected.
Fréchet Distance

- Note that we do not demand strict monotonicity for either $\sigma$ or $\tau$.
- While efficient algorithms are known for computing the Fréchet distance of polygonal curves, the same problem for triangulated surfaces is $\mathcal{NP}$-hard.
- However, a variant, the so-called weak Fréchet distance, can be computed in polynomial time [Alt&Buchin 2010].
Interpolation Versus Approximation

- For $m \in \mathbb{N}_0$, we are given $m + 1$ points $q_0, q_1, \ldots, q_m \in \mathbb{R}^n$, possibly with matching parameter values $u_0 < u_1 < \ldots < u_m$.

- For an interpolation of $q_0, q_1, \ldots, q_m$ we seek a curve $C$ such that either
  - $C(x_i) = q_i$ for arbitrary $x_i \in \mathbb{R}$, for all $i \in \{0, 1, \ldots, m\}$, or
  - $C(u_i) = q_i$ for all $i \in \{0, 1, \ldots, m\}$.

- For an approximation of $q_0, q_1, \ldots, q_m$ we seek a curve $C$ such that the distance between $C$ and $q_0, q_1, \ldots, q_m$ is smaller than a user-specified threshold relative to some distance measure.

- Similarly for approximation/interpolation by a surface rather than a curve.
Humorous View of Approximation

Curve-Fitting Methods and the Messages They Send

- **Linear**
  - "Hey, I did a regression."

- **Quadratic**
  - "I wanted a curved line, so I made one with math."

- **Logarithmic**
  - "Look, it's tapering off!"

- **Exponential**
  - "Look, it's growing uncontrollably!"

- **LOESS**
  - "I'm sophisticated, not like those bumbling polynomial people."

- **Linear, No Slope**
  - "I'm making a scatter plot but I don't want to."

- **Logistic**
  - "I need to connect these two lines, but my first idea didn't have enough math."

- **Confidence Interval**
  - "Listen, science is hard but I'm a serious person doing my best."

- **Piecewise**
  - "I have a theory, and this is the only data I could find."

- **Connecting Lines**
  - "I clicked 'smooth lines' in Excel."

- **Ad-Hoc Filter**
  - "I had an idea for how to clean up the data. What do you think?"

- **House of Cards**
  - "As you can see, this model smoothly fits the- wait? No, no, don't extend it aaaaaa!"

[Image credit: https://xkcd.com]
Lagrange Interpolation

Definition 170 (Lagrange polynomial)

For \( m \in \mathbb{N} \), consider \( m + 1 \) parameter values \( u_0 < u_1 < \ldots < u_m \) and let \( i \in \{0, 1, \ldots, m\} \). Then the \( i \)-th Lagrange polynomial of degree \( m \) is defined as

\[
L_{i,m}(u) := \prod_{j=0, i \neq j}^{m} \frac{u - u_j}{u_i - u_j}.
\]

Definition 171 (Lagrange interpolation)

For \( m \in \mathbb{N} \), consider \( m + 1 \) parameter values \( u_0 < u_1 < \ldots < u_m \) and \( m + 1 \) data points \( q_0, q_1, \ldots, q_m \). Then the Lagrange interpolation of \( q_0, q_1, \ldots, q_m \) is given by

\[
L(u) := \sum_{i=0}^{m} L_{i,m}(u)q_i.
\]
Lagrange Interpolation

Lemma 172

For \( m \in \mathbb{N} \), let \( \mathcal{L} \) be defined for \( m + 1 \) parameter values \( u_0 < u_1 < \ldots < u_m \) and \( m + 1 \) data points \( q_0, q_1, \ldots, q_m \), as given in Def. 171. Then \( \mathcal{L}(u_k) = q_k \) for all \( k \in \{0, 1, \ldots, m\} \).

Proof: For all \( k \in \{0, 1, \ldots, m\} \), we have

\[
L_{i,m}(u_k) = \prod_{j=0, i \neq j}^{m} \frac{u_k - u_j}{u_i - u_j} = \delta_{ik} = \begin{cases} 0 & \text{if } i \neq k, \\ 1 & \text{if } i = k. \end{cases}
\]

Hence,

\[
\mathcal{L}(u_k) = \sum_{i=0}^{m} L_{i,m}(u_k)q_i = q_k.
\]

Corollary 173

For \( m \in \mathbb{N} \), the Lagrange polynomials \( L_{0,m}, L_{1,m}, \ldots, L_{m,m} \) form a basis of the vector space of all polynomials of degree at most \( m \).

Sketch of Proof: Recall that exactly one polynomial of degree \( m \) interpolates \( m + 1 \) data points.
Newton Interpolation

Definition 174 (Newton polynomial)

For $m \in \mathbb{N}$, consider $m + 1$ parameter values $u_0 < u_1 < \ldots < u_m$ and let $i \in \{0, 1, \ldots, m\}$. Then the $i$-th Newton polynomial is defined as

$$l_i(u) := \prod_{j=0}^{i-1} (u - u_j)$$

with, by convention, $l_0(u) := 1$.

Definition 175 (Newton interpolation)

For $m \in \mathbb{N}$, consider $m + 1$ parameter values $u_0 < u_1 < \ldots < u_m$ and $m + 1$ data points $q_0, q_1, \ldots, q_m$. Then the Newton interpolation of $q_0, q_1, \ldots, q_m$ is given by

$$I(u) := \sum_{i=0}^{m} l_i(u)p_i,$$

with

$$p_i := \begin{cases} q_i & \text{for } i = 0, \\ \frac{q_i - \sum_{j=0}^{i-1} l_j(u_i)p_j}{l_i(u_i)} & \text{for } i > 0. \end{cases}$$
Newton Interpolation

Lemma 176

For \( m \in \mathbb{N} \), let \( I \) be defined for \( m + 1 \) parameter values \( u_0 < u_1 < \ldots < u_m \) and \( m + 1 \) data points \( q_0, q_1, \ldots, q_m \), as given in Def. 175. Then \( I(u_k) = q_k \) for all \( k \in \{0, 1, \ldots, m\} \).

Proof: For all \( k \in \{0, 1, \ldots, m\} \), we have for all \( i > 1 \)

\[
l_i(u_k) = \prod_{j=0}^{i-1} (u_k - u_j) \quad \begin{cases} = 0 & \text{if } k \leq i - 1, \\ \neq 0 & \text{if } k \geq i. \end{cases}
\]

We have

\[
I(u_0) = 1 \cdot p_0 = q_0,
\]

and for each \( 1 \leq k \leq m \)

\[
I(u_k) = \sum_{i=0}^{m} l_i(u_k)p_i = \sum_{i=0}^{k} l_i(u_k)p_i = \sum_{i=0}^{k-1} l_i(u_k)p_i + l_k(u_k)p_k
\]

\[
= \sum_{i=0}^{k-1} l_i(u_k)p_i + l_k(u_k) \cdot \frac{q_k - \sum_{j=0}^{k-1} l_j(u_k)p_j}{l_k(u_k)} = q_k.
\]

\[\square\]
Sampling of a function \( f \) and subsequent Lagrange interpolation may yield an extremely poor approximation of \( f \) even if \( f \) is continuously differentiable.

C. Runge: Consider \( f(x) := \frac{1}{1+x^2} \) and \( n + 1 \) uniform samples within \([-5, 5]\), with \( n := 20 \).

Similar problems occur for Newton interpolation.
B-Spline Interpolation

Let \( k \in \mathbb{N}_0 \) and suppose that we are looking for \( n + 1 \) control points \( p_0, p_1, \ldots, p_n \) and a knot vector \( \tau := (t_0, t_1, \ldots, t_{n+k+1}) \) such that the B-spline curve \( B \) of degree \( k \) defined by \( p_0, p_1, \ldots, p_n \) and \( \tau \) interpolates \( q_0, q_1, \ldots, q_m \), with \( B(u_i) = q_i \) for all \( i \in \{0, 1, \ldots, m\} \) and some given \( u_0 < u_1 < \ldots < u_m \).

If \( n = m \), then we get the following system of equations:

\[
\begin{bmatrix}
N_{0,k}(u_0) & \cdots & N_{n,k}(u_0) \\
\vdots & \ddots & \vdots \\
N_{0,k}(u_n) & \cdots & N_{n,k}(u_n)
\end{bmatrix}
\begin{bmatrix}
p_0 \\
p_1 \\
p_n
\end{bmatrix}
=:
\begin{bmatrix}
q_0 \\
q_1 \\
q_n
\end{bmatrix}
\]

Hence, the interpolation problem can be solved if the (quadratic) collocation matrix \( N \) is invertible.

**Lemma 177 (Schönberg-Whitney)**

The collocation matrix \( N \) is invertible if and only if all its diagonal elements are non-zero, i.e., if and only if \( t_i \leq u_i < t_{i+k+1} \), for all \( i \in \{0, 1, \ldots, n\} \).

**Lem. 118:** The matrix \( N \) is a sparse band matrix without negative elements.

Fast and numerically reliable algorithms exist for computing the inverse of \( N \).
B-Spline Interpolation

- Most applications do not require specific parameter values \( u_i \).
- In such a case, one can fix the knots \( t_i \), and choose \( u_i \) as follows ("Greville-abscissae"):

\[
   u_i := \frac{1}{k} \sum_{j=1}^{k} t_{i+j} \quad \text{for all } i \in \{0, 1, \ldots, n\}.
\]

- Note that \( t_i \) and \( t_{i+k+1} \) do not enter the definition of \( u_i \).
- Of course,

\[
   t_i \leq t_{i+1} \leq \frac{1}{k} (t_{i+1} + \cdots + t_{i+k}) \leq t_{i+k} \leq t_{i+k+1},
\]

thus meeting the Schönberg-Whitney condition of Lem. 177. Equality would only occur if an inner knot has multiplicity \( k + 1 \). (But then the B-spline would be discontinuous!)
Effects of Parameters and Knots

Since a B-spline has continuous speed and acceleration (for $k \geq 3$), it is obvious that the parameter values $u_i$ should bear a meaningful relation to the distances between the data points. Otherwise, overshooting is bound to occur!

Consider

$$u_0 := 0 \quad \text{and} \quad u_{i+1} := u_i + \Delta_i \quad \text{for all} \quad i \in \{1, \ldots, m - 1\},$$

with

$$\Delta_i := \|q_i - q_{i-1}\|^p \quad \text{for some} \quad p \in [0, 1] \quad \text{and all} \quad i \in \{1, \ldots, m - 1\}.$$ 

These parameter values are known as \textit{uniform} if $p = 0$, \textit{centripetal} if $p = \frac{1}{2}$, and \textit{chordal} if $p = 1$.

Suitable knots that meet the Schönberg-Whitney conditions (Lem. 177) are defined as follows:

$$t_i := \frac{1}{k} \left( u_{i-k} + u_{i-k+1} + \ldots + u_{i-1} \right)$$
B-Spline Approximation

- If \( m > n \), i.e., if there are more data points than control points, then the linear system \( \mathbf{Np} = q \) is over-determined and a solution need not exist.
- One popular option is a least-squares fit, which is achieved if
  \[
  \mathbf{N}^T \mathbf{Np} = \mathbf{N}^T q.
  \]
- Hence, if \( \mathbf{N}^T \mathbf{N} \) is invertible then we get
  \[
  p = (\mathbf{N}^T \mathbf{N})^{-1} \mathbf{N}^T q.
  \]
- An extension of the Schönberg-Whitney Lem. 177 tells us that the matrix \( \mathbf{N}^T \mathbf{N} \) is invertible exactly if the Schönberg-Whitney conditions are met:

**Lemma 178**

The matrix \( \mathbf{N}^T \mathbf{N} \) is invertible if and only if \( t_i \leq u_i < t_{i+k+1} \), for all \( i \in \{0, 1, \ldots, n\} \).
Bernstein Polynomials

Definition 179 (*Bernstein polynomial*)

A *Bernstein polynomial* of degree $n$ is a linear combination of Bernstein basis polynomials of degree $n$:

$$B_n(x) := \sum_{i=0}^{n} \mu_i B_{i,n}(x), \quad \text{with } \mu_0, \mu_1, \ldots, \mu_n \in \mathbb{R}.$$ 

- Hence, every polynomial (in power basis) can be seen as a Bernstein polynomial, albeit with unknown scalars for the linear combination.
- Can we select $\mu_i$ such that a decent approximation of a user-specified function is achieved?
Bernstein Approximation

Definition 180 (Bernstein approximation)

Consider a continuous function \( f : [0, 1] \rightarrow \mathbb{R} \). The Bernstein approximation with degree \( n \) of \( f \) is defined as

\[
B_{n,f}(x) := \sum_{i=0}^{n} f \left( \frac{i}{n} \right) B_{i,n}(x).
\]

• Hence, a Bernstein approximation is given by a Bernstein polynomial, with weights \( \mu_i := f \left( \frac{i}{n} \right) \).

Theorem 181 (Weierstrass 1885, Bernstein 1911)

The Bernstein approximation \( B_{n,f} \) converges uniformly to the continuous function \( f \) on the interval \([0, 1]\). That is, given a tolerance \( \varepsilon > 0 \), there exists \( n \in \mathbb{N} \) such that

\[
|f(x) - B_{n,f}(x)| \leq \varepsilon \quad \text{for all } x \in [0, 1].
\]

• Since \( x := \frac{t-a}{b-a} \) maps \( t \in [a, b] \) to \( x \in [0, 1] \), this approximation theorem extends to continuous functions \( f : [a, b] \rightarrow \mathbb{R} \).
Sample Bernstein Approximation

Sample Bernstein approximation of a continuous function:

\[ f: [0, 1] \rightarrow \mathbb{R} \]

\[ f(x) := \frac{1}{1 + (10x - 5)^2} \]
Sample Bernstein Approximation

Sample Bernstein approximation of a continuous function:

\[ f : [0, 1] \rightarrow \mathbb{R} \]

\[ f(x) := \frac{1}{1 + (10x - 5)^2} \]
Sample Bernstein Approximation

Sample Bernstein approximation of a continuous function:

\[ f: [0, 1] \rightarrow \mathbb{R} \]

\[ f(x) := \sin(\pi x) + \frac{1}{5} \sin(6\pi x + \pi x^2) \]
Sample Bernstein Approximation

Sample Bernstein approximation of a continuous function:

\[ f : [0, 1] \rightarrow \mathbb{R} \quad f(x) := \sin(\pi x) + \frac{1}{5} \sin\left(6\pi x + \pi x^2\right) \]
Approximation of Polygonal Profiles

Informal problem statement

- For a set of planar (polygonal) profiles $\mathcal{P}$
- and an approximation threshold given,
- compute a smooth approximation such that the input topology is maintained.
Sample Real-World Application: Tool-Path Generation

- Approximation of tool paths: Asymmetric tolerances may be required!
- In the example: 6,648 input line segments versus 252 circular arcs.
Sample Real-World Application: Conversion of Scanned Data

- Scanned data tends to be noisy.
- In the example: 526,489 segments (within 1,827 input polygons) versus 42,222 circular arcs or 14,462 cubic B-splines.
Sample Real-World Application: Recovering PCB Data

- Huge(!) data sets; maintaining the input topology is imperative.
- In the (toy) example: 81,984 input line segments versus 1,000 output line segments and 9,791 circular arcs.
Specifying a Tolerance

- Intuitively, for an input profile $P$ we seek a tolerance zone, $TZ(P, d_L, d_R)$, of $P$ with left tolerance $d_L$ and right tolerance $d_R$.

- Non-trivial tolerances classified as
  - symmetric if $-d_L = d_R > 0$,
  - asymmetric if $d_L < 0 \leq d_R$ or $d_L \leq 0 < d_R$, and
  - one-sided if $d_L < d_R < 0$ or $0 < d_L < d_R$.
Specifying a Tolerance

- It seems natural to pre-compute $\mathcal{TZ}(P, d_L, d_R)$ prior to approximation.
- But: Individual tolerance zones of the input profiles may intersect arbitrarily!

- Hence, we need to compute the tolerance zones for all profiles $P$ of the input $\mathcal{P}$ at once!
Cone of Influence

Definition 182 (Cone of influence)

The cone of influence, $CI(s)$, of

- a straight-line segment $s$ is the closure of the strip bounded by the normals through its endpoints;
- a circular arc $s$ is the closure of the cone bounded by the pair of rays originating in the arc’s center and extending through its endpoints;
- a point $s$ is the entire plane.
Signed Distance

**Definition 183 (Signed distance)**

We restrict our attention to the cone of influence of a site. Then, the *signed distance*, $d_s(s, p)$, of a point $p \in CI(s)$

- to an oriented straight-line segment or circular arc $s$ of $\mathcal{P}$ is given by the standard (Euclidean) distance of $p$ to $s$, multiplied by $-1$ if $p$ is on the left side of the supporting line or circle of $s$,
- to a vertex $s$ of $\mathcal{P}$ we take the standard distance between $p$ and $s$, and multiply it by $-1$ if the ray from $s$ to $p$ is locally on the left side of $s_1$ and $s_2$, where $s_1$ and $s_2$ are the sites of $\mathcal{P}$ that share $s$ as a common vertex.
Tolerance Zone

Definition 184 (*Tolerance zone*)

The tolerance zone of a site $s$ of $\mathcal{P}$ is defined as

$$T \mathcal{Z}_{\text{site}}(s, \mathcal{P}, d_L, d_R) := \{p \in \mathcal{V}(\mathcal{P}, s) : d_L < d_s(s, p) < d_R\}.$$ 

The tolerance zone of $\mathcal{P}$ is defined as the union of all tolerance zones of all sites:

$$T \mathcal{Z}(\mathcal{P}, d_L, d_R) := \bigcup_{s \in \mathcal{P}} T \mathcal{Z}_{\text{site}}(s, \mathcal{P}, d_L, d_R).$$
Problem Statement

Given: Input

- Set $\mathcal{P}$ of (open or closed) polygonal profiles that do not intersect pairwise;
- Left approximation tolerance $d_L$ and right approximation tolerance $d_R$, with $d_L < d_R$.

Compute: Approximation $\mathcal{A}$ of $\mathcal{P}$ such that

- $\mathcal{A}$ consists of $G^k$ curves, for some $k \in \mathbb{N}$,
- all curves of $\mathcal{A}$ are simple and pairwise disjoint,
- $\mathcal{A} \subset TZ(\mathcal{P}, d_L, d_R)$,
- $\mathcal{P} \subset TZ(\mathcal{A}, -d_R, -d_L)$ if requested by user,
- topology of $\mathcal{A}$ matches topology of $\mathcal{P}$.
Assume \(-d_L = d_R > 0\). We have
\[
\mathcal{A} \subset TZ(\mathcal{P}, -d_R, d_R) \land \mathcal{P} \subset TZ(\mathcal{A}, -d_R, d_R) \implies H(\mathcal{A}, \mathcal{P}) \leq d_R,
\]
where \(H(\mathcal{A}, \mathcal{P})\) denotes the Hausdorff distance between \(\mathcal{A}\) and \(\mathcal{P}\).

Assume \(-d_L = d_R > 0\). If each approximation curve \(A \in \mathcal{A}\) is “monotone” relative to its corresponding input curve \(P \in \mathcal{P}\), then
\[
\mathcal{A} \subset TZ(\mathcal{P}, -d_R, d_R) \land \mathcal{P} \subset TZ(\mathcal{A}, -d_R, d_R) \implies Fr(A, P) \leq d_R,
\]
where \(Fr(A, P)\) denotes the Fréchet distance between \(A\) and \(P\), for each \(A \in \mathcal{A}\) and corresponding \(P \in \mathcal{P}\).
Tolerance Zone and Distance Measures

- Omitting the second condition $\mathcal{P} \subset T\mathcal{Z}(\mathcal{A}, -d_R, -d_L)$ makes a difference!
Voronoi Diagram of Points

Voronoi experiment
Let’s throw a ball into “ideal” water (with no boundaries and no initial waves), and watch the wavefronts that emerge (within a finite portion of the water).
Voronoi Diagram of Points

Voronoi experiment

Let’s now throw two identical balls at the same time into the water: As the wavefronts meet, the *bisector* between the two balls is traced out.
Voronoi Diagram of Points

Voronoi experiment

We repeat the experiment with three identical balls thrown at the same time into the water: Again, the wavefronts trace out the bisectors between the balls as they meet.
Voronoi Diagram of Points

Voronoi regions
The blue bisectors defined by the three balls partition the water into Voronoi regions: Each region is the loci of points \( q \) closer to its defining ball than to any other ball.

\[
d(q, p_2) \leq \max\{d(q, p_1), d(q, p_3)\}
\]
Voronoi Diagram of Points: Sample Diagram

- Input set $S$ of points, Voronoi region, Voronoi diagram, Voronoi nodes.
Consider a set $S$ of $n$ points, straight-line segments, and circular arcs ("sites").

Intuitively, the Voronoi diagram of $S$ partitions the Euclidean plane into regions that are closer to one site than to any other.

Natural generalization of Voronoi diagrams of points, but Voronoi regions are now bounded by conics and need not be convex. (And non-trivial technicalities . . .)
VD-Based Computation of the Tolerance Zone

- [Held&Heimlich 2008; Held&Kaaser 2014] use the (generalized) Voronoi diagram of the input to compute the boundary of the tolerance zone.

- Tolerance zone computation for an input profile:
  1. Collect all nodes of Voronoi cells left of the profile.
  2. Skip nodes that are further away than $d_L$ from the profile.
  3. Remove trees within the tolerance zone and add spikes.
  4. Repeat this procedure for the right side of the profile w.r.t. $d_R$. 
Offset Spikes

- Offset spikes ensure that the directed Hausdorff distance from the input to the approximation curve does not exceed the user-specified maximum tolerance.
- Spikes are formed by portions of the Voronoi diagram; they can be computed in linear time.
Circular Biarcs

Definition 185 (Circular biarc)

Given are two points \( p_s \) and \( p_e \) together with two tangents \( t_s \) and \( t_e \). The (circular) biarc between \( p_s \) and \( p_e \) consists of two circular arcs \( a_1 \) and \( a_2 \) such that

1. Arc \( a_1 \) runs from \( p_s \) to a joint \( p_j \); arc \( a_2 \) runs from \( p_j \) to \( p_e \);
2. Tangent vector of \( a_1 \) in \( p_s \) is aligned with \( t_s \); tangent vector of \( a_2 \) in \( p_e \) is aligned with \( t_e \);
3. The arcs \( a_1 \) and \( a_2 \) are \( G^1 \)-continuous at the joint.
Circular Biarcs

- The joint $p_j$ is not uniquely determined.

**Lemma 186**

Consider two points $p_s$ and $p_e$ together with two tangents $t_s$ and $t_e$. Then all possible joints for the biarcs defined by $p_s, p_e, t_s, t_e$ lie on a circle.
Approximation Nodes

- Which points and tangents shall be used for the approximation?

- Note: Input vertices
  - do not lie in the tolerance zone for one-sided tolerances, and
  - may be far away from “good” approximation curves for asymmetric tolerances or if $\mathcal{P} \subset \mathcal{T}Z(A, -d_R, -d_L)$ is not required!
Approximation Nodes

1. [Held&Heimlich 2008] compute Voronoi diagram inside of the tolerance zone.
2. Place approximation nodes ("a-nodes") on medial axis of the boundaries of the components of the tolerance zone; tangents according to [Meek&Walton 1992]:

\[
\frac{(a_i - a_{i-1})}{||a_i - a_{i-1}||} + \frac{(a_{i+1} - a_i)}{||a_{i+1} - a_i||}.
\]
Doubling-and-Bisection Heuristic for Biarc Approximation

- Suppose we are to start in a-node $a_1$, with a-nodes numbered consecutively.
- Doubling: Successively try $k := 1, 2, 3, \ldots$ and check whether a valid biarc exists (within the tolerance zone) between $a_1$ and $a_{2k}$.
- Bisection: If biarc to $a_{2k}$ is valid and biarc to $a_{2k+1}$ is not valid then apply bisection between a-node indices $2^k$ and $2^{k+1}$.
- Validity check for a biarc can be restricted to a strip within the tolerance band!

Theorem 187 (Held&Heimlich 2008)

Let $n$ denote the number of vertices of a set $\mathcal{P}$ of polygonal profiles. Then a $G^1$ biarc approximation within an (asymmetric) user-specified tolerance that preserves the topology of $\mathcal{P}$ can be computed in $O(n \log n)$ time.
Sample Biarc Approximation
Sample Biarc Approximation

SALZBURG

SCIENTIFIC COMPUTING

SALZBURG

SCIENTIFIC COMPUTING
Theoretical Results on Biarc Approximation

**Theorem 188 ([Drysdale&Rote&Sturm 2008](#))**

Let $n$ denote the number of vertices of one open polygon $P$, and assume that a tolerance zone together with $a$-nodes are given. Then a $G^1$ biarc approximation of $P$ that uses the minimum number of biarcs (relative to the $a$-nodes given) can be computed in $O(n^2 \log n)$ time.

**Theorem 189 ([Maier 2010](#))**

Let $n$ denote the number of vertices of one open polygon $P$, and assume that a tolerance zone is given. Then a $G^1$ biarc approximation of $P$ that uses the minimum number of biarcs can be computed in $O(n^2)$ time.

**Theorem 190 ([Maier&Pisinger 2013](#))**

Let $n$ denote the number of vertices of one closed polygon $P$, and assume that a tolerance zone is given. Then a $G^1$ biarc approximation of $P$ that uses the minimum number of biarcs can be computed in $O(n^3)$ time.
Uniform Cubic B-Spline Approximation

- Since consecutive B-spline segments share a common set of control vertices rather than only one a-node, there is no obvious way how a greedy scheme is adapted to support a B-spline approximation.
- [Held&Kaaser 2014]: Recursive subdivision heuristic for finding maximal uniform cubic B-splines.
- For every profile of $\mathcal{P}$, compute an initial B-spline curve that consists of a small number of segments.
- This initial approximation curve is then refined by subsequently adding a-nodes as new control vertices until an approximation curve that fits through the tolerance zone is obtained.

Theorem 191 (Held&Kaaser 2014)

Let $n$ denote the number of vertices of a set $\mathcal{P}$ of polygonal profiles. Then a $C^2$ approximation by uniform cubic B-splines within an (asymmetric) user-specified tolerance that preserves the topology of $\mathcal{P}$ can be computed in $O(n \log n)$ time.
B-Spline Intersection Checking

▶ Use de Boor’s algorithm to subdivide a spline segment, and recurse on the two halves. If the maximum intersection tolerance is reached then report an intersection.

▶ One may have to check one B-spline segment $ucbs$ for intersection with several straight-line segments of the boundary of the tolerance zone.

▶ Recall that the B-spline subdivision follows a regular pattern.

▶ Hence, one might end up computing the same control vertices over and over again.

▶ One can save computational time by storing the control vertices in a binary tree $T$, where each node of $T$ corresponds to one subdivision.
  ▶ During an intersection check with a line segment one starts at the root of $T$ and descends $T$ in the same way as one recurse on portions of $ucbs$.
  ▶ If a particular intersection check requires a subdivision whose corresponding control vertices are not stored in $T$ then these vertices are computed and stored.
Sample Approximation by Uniform Cubic B-Splines
Sample Approximation by Uniform Cubic B-Splines
The Voronoi diagram allows to filter noisy input without preprocessing, and the tolerance zones allow to assert important properties of the approximation, but VD-based tolerance zones need not support optimum approximations!
Shortcomings of A-Nodes

- A-nodes are a convenient means for obtaining control points for B-splines or support for biarcs, but an inappropriate placement of the a-nodes may cause the number of approximation primitives to sky-rocket!
The End!

I hope that you enjoyed this course, and I wish you all the best for your future studies.