Curves for CAGD

FSP–Tutorial
A segment of a **polynomial curve** of degree $n$ in $\mathbb{R}^d$ ($d = 2, 3$) is parametrized with respect to the **monomial basis** $t^0, t^1, \ldots, t^n$ by

$$b(t) = \sum_{i=0}^{n} b_i t^i, \ t \in [a, b].$$

The coefficient vectors $b_i$ do not have a geometric interpretation.

Example: Parabola $b(t) = b_0 t^0 + b_1 t^1 + b_2 t^2$
Bézier Curves and Bernstein Basis

Given \( n + 1 \) points \( b_0, b_1, \ldots, b_n \) in space \( \mathbb{R}^d \), the Bézier curve defined by these control points is

\[
b(t) = \sum_{i=0}^{n} b_i B_i^n(t), \quad t \in [0, 1]. \quad (1)
\]

\( b(t) \) is a polynomial curve parametrized in the Bernstein basis

\[
B_i^n(t) = \binom{n}{i}(1 - t)^{n-i}t^i. \quad (2)
\]
Geometric Properties of the Bernstein Basis

$t \in [0, 1]$

- Recursion: $B_i^n(t) = (1 - t)B_i^{n-1}(t) + tB_{i-1}^{n-1}(t)$, \(i = 0, \ldots, n\).
- Derivatives: \(\frac{d}{dt}B_i^n(t) = n \left(B_{i-1}^{n-1}(t) - B_i^{n-1}(t)\right)\).
- Non-negativity: $B_i^n(t) \geq 0$.
- Partition of unity: $\sum_{i=0}^n B_i^n(t) = 1$.
- Symmetry: $B_i^n(t) = B_{n-i}^n(1 - t)$.
- $B_i^n(t)$ has a \(i\)-fold zero at $t = 0$, and a $(n - i)$-fold zero at $t = 1$.
- $B_i^n(t)$ has exactly one maximum in $I := [0, 1]$ at $t = \frac{i}{n}$. 
Derivatives of Bézier Curves

Derivative $\dot{b}(t)$ of a Bézier curve:

$$\dot{b}(t) = \sum_{j=0}^{n-1} B_j^{n-1}(t)(b_{j+1} - b_j) = n \sum_{j=0}^{n-1} B_j^{n-1}(t) \Delta b_j.$$ 

Tangent vectors at $t = 0$ and $t = 1$:

$$\dot{b}(0) = n(b_1 - b_0), \quad \dot{b}(1) = n(b_n - b_{n-1}),$$

Higher order derivatives can be computed in a similar way.
Control point $b_k$ is moved to

$$\tilde{b}_k = b_k + v.$$ 

The curve changes to

$$\tilde{b}(t) = \sum_{i=0}^{n} B_i^n(t)b_i + B_k^n(t)v,$$

$$= b(t) + B_k^n(t)v.$$ 

The shape of the curve changes globally.
Algorithm of de Casteljau

• Input: control polygon $b_0, \ldots, b_n$ and $t \in [0, 1]$

• Initialize: $b_i^0 := b_i, i = 0, \ldots, n$.

• Recursion:
  \[ b_i^r = (1 - t)b_i^{r-1} + tb_{i+1}^{r-1}, \quad r = 1, \ldots, n, \quad i = 0, \ldots, n - r \]

• Result: $b_0^n = b(t)$
• $b(t) = \sum_i B^n_i(t)b_i$ is a polynomial curve of degree $n$.
• End point interpolation: $b(0) = b_0, b(1) = b_n$
• Affine invariance: Constructing the curve from the image points $b'_i$ is equivalent to applying an affine mapping $\alpha : x' = A \cdot x + a$ to the curve $b(t)$.

\[
\tilde{b}(t) = \sum B^n_i(t)b'_i = \sum B^n_i(t)(A \cdot b_i + a) \\
= A \cdot \left( \sum B^n_i(t)b_i \right) + \sum B^n_i(t)a = A \cdot b(t) + a = b(t)'.
\]
• Convex hull property: The Bézier curve $b(t)$ lies completely in the convex hull of the control points $b_i$, $i = 0, \ldots, n$.

• Variation diminishing property: The number of the intersections of $b(t)$ with a hyperplane $H$ ($d = 2$: line; $d = 3$: plane) is not greater than the number of the intersections of the control polygon $b_0, \ldots, b_n$ with $H$.

• A convex control polygon implies a convex Bézier curve.
Examples

Approximating Bézier curve for the quarter of a circle

Approximating Bézier curve for an ellipse
Polar Form

Generalized algorithm of de Casteljau

\[ b_r^i(t_1, \ldots, t_r) := (1 - t_r)b_r^{i-1}(t_1, \ldots, t_{r-1}) + t_r b_{i+1}^{r-1}(t_1, \ldots, t_{r-1}), \]
for \( r = 1, \ldots, n, i = 0, \ldots, n - r. \)

\[ B^n : \mathbb{R}^n \rightarrow \mathbb{R}^d \]
\[ (t_1, \ldots, t_n) \mapsto B^n(t_1, \ldots, t_n) \]
\[ := b_0^n(t_1, \ldots, t_n). \]

- \( B^n \) is symmetric.
- \( B^n \) is multiaffine.
• $B^n$ is *symmetric*: After the first $i - 1$ steps in the de Casteljau scheme we have the polygon $b_{0}^{i-1}, \ldots, b_{n-i+1}^{i-1}$. Now we find:

\[
\begin{align*}
    b_j^i &= (1 - t_i)b_{j}^{i-1} + t_i b_{j+1}^{i-1}, \\
    b_{j+1}^{i} &= (1 - t_{i+1}) [(1 - t_i)b_j^{i-1} + t_i b_{j+1}^{i-1}] + t_{i+1} [(1 - t_i)b_{j+1}^{i-1} + t_i b_{j+2}^{i-1}] \\
    &= (1 - t_i)(1 - t_{i+1})b_j^{i-1} + [(1 - t_{i+1})t_i + (1 - t_i)t_{i+1}] b_{j+1}^{i-1} + t_it_{i+1} b_{j+2}^{i-1}.
\end{align*}
\]

• $B^n$ is *multi-affine*:

\[
B^n(t_1, \ldots, t_{i-1}, (1 - \alpha)a_1 + \alpha a_2, t_{i+1}, \ldots, t_n) = (1 - \alpha)B^n(t_1, \ldots, t_{i-1}, a_1, t_{i+1}, \ldots, t_n) + \alpha B^n(t_1, \ldots, t_{i-1}, a_2, t_{i+1}, \ldots, t_n).
\]
Elementary Symmetric Functions

\[ S_i(t_1, \ldots, t_n) = \sum_{1 \leq j_1 < j_2 < \ldots < j_i \leq n} t_{j_1} \cdots t_{j_i}, \quad i = 0, \ldots, n. \]

\[ n = 3 : \quad S_0(t_1, t_2, t_3) = 1, \]
\[ S_1(t_1, t_2, t_3) = t_1 + t_2 + t_3, \]
\[ S_2(t_1, t_2, t_3) = t_1 t_2 + t_1 t_3 + t_2 t_3, \]
\[ S_3(t_1, t_2, t_3) = t_1 t_2 t_3. \]

- Any linear combination \( \sum_{i=0}^{n} S_i(t_1, \ldots, t_n)c_i, c_i \in \mathbb{R}^d \) is a symmetric multiaffine map \( \mathbb{R}^n \rightarrow \mathbb{R}^d \).

- A map \( F : \mathbb{R}^n \rightarrow \mathbb{R}^d \) is multiaffine and symmetric exactly if it is a linear combination \( F = \sum_{i=0}^{n} S_i(t_1, \ldots, t_n)c_i \) of elementary symmetric functions.
To each polynomial curve \( f : \mathbb{R} \to \mathbb{R}^d \) of degree \( n \) there exists exactly one symmetric multiaffine map \( F : \mathbb{R}^n \to \mathbb{R}^d \) with \( F(u, \ldots, u) = f(u) \). \( F \) is called \textit{polar form} of \( f \).

Example: The cubic polynomial curve

\[
f(u) = a_0 + ua_1 + u^2a_2 + u^3a_3
\]

possesses the polar form

\[
F(u_1, u_2, u_3) = a_0 + \frac{1}{3}(u_1+u_2+u_3)a_1 + \frac{1}{3}(u_1u_2+u_1u_3+u_2u_3)a_2 + u_1u_2u_3a_3.
\]
To each Bézier curve (polynomial curve) $b^n \subset \mathbb{R}^d$ there exists a unique polar form $B^n : \mathbb{R}^n \to \mathbb{R}^d$ with

$$B^n(t, \ldots, t) = b^n(t).$$

The Bézier curve segment to the parameter interval $[0, 1]$ possesses the Bézier points

$$b_i = B^n_{n-i, i}.$$ 

Further, we see that

$$b^i_0 = B^n_{n-i, i}(0, \ldots, 0, t, \ldots, t),$$

$$b^{n-i}_i = B^n_{n-i, i}(t, \ldots, t, 1, \ldots, 1).$$
Subdivision of Bézier Curves

Bézier curve $b^n(u)$ with control points $(b_0, \ldots, b_n)$. The de Casteljau algorithm yields the control points $(c_0, \ldots, c_n)$ and $(d_0, \ldots, d_n)$ of $b^n(u)$ over the subintervals $[0, u]$ and $[u, 1]$, respectively. We have

$$c_i = b^i_0(u), \quad d_i = b^{n-i}_i(u), \quad i = 0, \ldots, n.$$  

Repeated subdivision yields a sequence of polygons that converges to the curve. The refinement is a corner cutting.
B–Spline Curves – Introduction

Bézier curves have fixed degree depending on the number of control points. B–Spline curves are composed of Bézier curve segments.

Control polygons and B–Spline curves of different degrees
A B–Spline curve

\[ s(u) = \sum_{i=0}^{m} N_i^n(u)d_i \]

is determined by the control points \( d_i \) and the degree of the basis functions \( N_i^n(u) \).

regular case: \( \# \) control points = degree + \( \# \) segments
B–Spline Basis Functions

Constant and linear B–Splines for the knots 0, 1, 3, 4, 6 and 0, 0, 1, 3, 4, 6, 6.

Quadratic and cubic B–Splines for the knots 0, 0, 0, 1, 3, 4, 6, 6, 6 and 0(4), 1, 3, 4, 6(4).
Properties of B–Splines

\[ N_i^0(u) := \begin{cases} 
1 & \text{for} \quad t_i \leq u < t_{i+1} \\
0 & \text{else} 
\end{cases} \]

\[ N_i^r(u) := \frac{u - t_i}{t_{i+r} - t_i} N_i^{r-1}(u) + \frac{t_{i+r+1} - u}{t_{i+r+1} - t_{i+1}} N_{i+1}^{r-1}(u), \quad 1 \leq r \leq n. \]

- \( N_i^p(u) \) is a function composed of polynomials.
- The restriction of \( N_i^p(u) \) to \((t_j, t_{j+1})\) is a polynomial of degree \( p \).
- Nonnegativity:
  \[ N_i^p(u) > 0 \text{ for } u \in (t_i, t_{i+n+1}), \quad N_i^p(u) = 0 \text{ for } u \notin [t_i, t_{i+n+1}]. \]
- The interval \([t_i, t_{i+p+1}]\) is called support of \( N_i^p(u) \).
Properties of B–Splines 2

- Partition of unity: \( \sum_{j=i-p}^{i} N_j^p(u) = 1. \)
- \( N_i^p(u) \) is \( C^{p-k} \)-continuous at a knot of multiplicity \( k \).

![Knot vector 
[0, 0, 1, 2, 2, 4, 4]](image)

- On any interval \([t_i, t_{i+1})\) at most \( p + 1 \) basis functions of degree \( p \) are non-zero, \( N_{i-p}^p(u), N_{i-p+1}^p(u), \ldots, \) and \( N_i^p(u) \).
The restriction of $N_i^n(u)$ to $(t_j, t_{j+1})$ is a polynomial $N_{i,j}^n(u)$.

⇒ associated polar form $P_{i,j}^n(u_1, \ldots, u_n)$.

The polar form $P_{i,j}^n$ of $N_{i,j}^n$ is given by

\[
P_{i,j}^0() = \delta_{i,j},
\]

\[
P_{i,j}^r(u_1, \ldots, u_r) = \frac{u_r-t_i}{t_{i+r}-t_i} P_{i,j}^{r-1}(u_1, \ldots, u_{r-1}) + \frac{t_{i+r+1}-u_r}{t_{i+r+1}-t_{i+1}} P_{i+1,j}^{r-1}(u_1, \ldots, u_{r-1}).
\]

$P_{i,j}^r$ is symmetric and multiaffine and $P_{i,j}^r(u, \ldots, u) = N_{i,j}(u)$. 

---

FSP-Seminar, Graz, November 2005 22
Main theorem on B–Spline curves

B–Spline curve

\[ s(u) = \sum_{i=0}^{m} N^n_i(u) d_i, \quad d_i \in \mathbb{R}^d, \]

knots \( T = (t_0, \ldots, t_{m+n+1}) \). Polar form \( S_j(u_1, \ldots, u_n) \) of the restriction of \( s \) onto \( (t_j, t_{j+1}) \).

The control points \( d_i \) (for \( l \leq j \leq l+n \)) can be expressed as

\[ d_i = S_j(t_{i+1}, \ldots, t_{i+n}). \]

The normalized B-Splines \( \{N^n_i| i = 0, \ldots, m\} \) are \textit{linearly independent}. 

\[ \mathbf{T} = (0^4, 1, 2, 4, 5, 6^4) \]
Geometric Properties of B–Spline Curves

De Boor’s algorithm: $d_i^0 := d_i,$

$$d_i^r := (1 - \frac{u - t_i}{t_i + n + 1 - r - t_i})d_{i-1}^{r-1} + \frac{u - t_i}{t_i + n + 1 - r - t_i}d_{i+1}^{r-1} \text{ for } l - n + r \leq i \leq l.$$  

$\Rightarrow$ Curve point $s(u) = d_i^n.$

- **Affine invariance**: true because of $\sum_{i=0}^{m} N_i^n(u) = 1.$

- **Convex hull property**.

- **Subdivision property**.

- **Local control**: Changing a control point $d_i$ influences at most $n + 1$ segments ($(t_i, t_{i+n+1})$).

- In each recursion depth $r \ (r \geq 1)$ the knot $u$ is inserted as new knot. At the end, $u$ is inserted $n$–fold.
Knot Insertion

B–spline curve \( s(u) = \sum_{i=0}^{m} N_i^n(u) d_i \) over \( T = (t_0, \ldots, t_{m+n+1}) \)

Insertion of a new knot \( t \) at \( t_l \leq t < t_{l+1} \) yields

\[
s(u) = \sum_{i=0}^{m+1} N_i^{n*}(u) d_i^*
\]

over the new knot vector \( T^* = (t_0, \ldots, t_l, t, t_{l+1}, \ldots, t_{m+n+1}) \).

With

\[
t = (1 - \frac{t-t_l}{t_{i+n}-t_l}) t_i + \frac{t-t_l}{t_{i+n}-t_l} t_{i+n} = (1 - a_i) t_i + a_i t_{i+n}
\]

we obtain

\[
d_i^* = (1 - a_i) d_{i-1} + a_i d_i
\]

with \( a_i = 1 \) if \( i \leq l - n \), \( a_i = 0 \) if \( l + 1 \leq i \),
else \( a_i = t - t_i/t_{i+n} - t_i \).
B–Spline Curve Subdivision

Given a control polygon $P$ of a closed B–Spline curve.

- Splitting: introduce midpoints of segments.
- Averaging: Replace new segments by their centers.

Subdivision: $Splitting + (k - 1)$-times $Averaging$ results in a refined control polygon of a B–Spline curve of degree $k$. 
Quadratic B–Spline Subdivision – Chaikin

Splitting

\[ (2, 4) \]
\[ (4, 6) \]
\[ [4, 4] \]
\[ [6, 6] \]
\[ [2, 2] \]
\[ (0, 2) = (8, 10) \]
\[ (6, 8) \]

\[ T = (\ldots, 0, 2, 4, 6, 8, \ldots) \]

Averaging

\[ (3, 4) \]
\[ (4, 5) \]
\[ (2, 3) \]
\[ (5, 6) \]
\[ (1, 2) \]
\[ (6, 7) \]
\[ (8, 9) \]
\[ (7, 8) \]

\[ T = (\ldots, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 \ldots) \]

\[ p_{2i}^{k+1} = \frac{3}{4} p_i^k + \frac{1}{4} p_{i+1}^k \]

\[ p_{2i+1}^{k+1} = \frac{1}{4} p_i^k + \frac{3}{4} p_{i+1}^k \]
Cubic B–Spline Subdivision

\[ T = (\ldots, 0, 2, 4, 6, 8, 10, \ldots) \quad T = (\ldots, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 \ldots) \]

\[
\begin{align*}
  p_{2i}^{k+1} & = \frac{4}{8} p_i^k + \frac{4}{8} p_{i+1}^k \\
  p_{2i+1}^{k+1} & = \frac{1}{8} p_i^k + \frac{6}{8} p_{i+1}^k + \frac{1}{8} p_{i+2}^k
\end{align*}
\]
Circles, ellipses, hyperbolas and piecewise rational curves cannot be represented exactly as (piecewise) polynomial curves. We introduce weights $w_i$ and rational parametrizations

$$b(t) = \frac{1}{\sum_i^n B_i^n(t) w_i} \sum_i^n B_i^n(t) b_i w_i,$$

Circle

Hyperbola

Parabola

Circle

Ellipse
Interpolation and Approximation

• Given:
  – $M$ data points $c_0, \ldots, c_{M-1}$ with $M$ corresponding parameter values $x_0, \ldots, x_{M-1}$
  – $m$–dimensional space with basis functions $N_0^n, \ldots, N_{m-1}^n$ over a knot vector $T$

• Find: Curve $s(u) = \sum_{i=0}^{m-1} N_i(u)^n d_i$ that interpolates / approximates the data points.

The given points $c_0, \ldots, c_{M-1}$ are to be interpolated / approximated at given parameter values $x_0, \ldots, x_{M-1}$
Interpolation and Approximation - Cases

- $M = m$: number of conditions = number of degree of freedom → interpolation
- $M > m$: overdetermined system of equations → approximation
- $M < m$: underdetermined system of equations → "constrained" interpolation / approximation (additional constraints)

Approximating and interpolating $B$-spline curve
\section*{B–Spline Interpolation}

\begin{itemize}
  \item Given:
    \begin{itemize}
      \item $m$ data points $\mathbf{c}_0, \ldots, \mathbf{c}_{m-1}$ with $m$ corresponding parameter values $x_0, \ldots, x_{m-1}$
      \item $m$–dimensional space with basis functions $N_0^n, \ldots, N_{m-1}^n$ over a knot vector $T = (t_0 \leq \ldots \leq t_{m+n})$
    \end{itemize}
  \item Find: Control points $\mathbf{d}_0, \ldots, \mathbf{d}_{m-1}$ such that $s(u) = \sum_{i=0}^{m-1} N_i^n(u) \mathbf{d}_i$ interpolates the data points $\mathbf{c}_0, \ldots, \mathbf{c}_{m-1}$ at parameter values $x_0, \ldots, x_{m-1}$: $s(x_i) = \mathbf{c}_i \quad \forall i$
    \begin{equation}
      \begin{pmatrix}
        N_0^n(x_0) & \ldots & N_{m-1}^n(x_0) \\
        \vdots & \ddots & \vdots \\
        N_0^n(x_{m-1}) & \ldots & N_{m-1}^n(x_{m-1})
      \end{pmatrix}
      \begin{pmatrix}
        \mathbf{d}_0 \\
        \vdots \\
        \mathbf{d}_{m-1}
      \end{pmatrix}
      =
      \begin{pmatrix}
        \mathbf{c}_0 \\
        \vdots \\
        \mathbf{c}_{m-1}
      \end{pmatrix}
    \end{equation}
  \item $A = (N_i^n)_{ij}$ has $n + 1$ non-zero entries per row (support of $N_i^n : [t_i, t_i + n + 1]$)
\end{itemize}
Special Case: Cubic Spline Interpolation

• Given:
  – Degree \( n = 3 \)
  – \( m - 2 \) data points \( c_0, \ldots, c_{m-3} \)
  – \( m \)-dimensional space of basis functions \( N^3_0, \ldots, N^3_{m-1} \)
  – knot vector \( T = (t_0 \leq \ldots \leq t_{m+3}) \)

• Find: cubic spline curve \( s(u) \) over \( T \) with \( s(t_{i+3}) = c_i \)
  (interpolating at the knots)

Possible additional conditions for the sake of uniqueness:

• Specification of the tangent at the endpoints

• In the case of closed curves: same 1st and 2nd derivatives at the endpoints \( t_3 \) and \( t_m \)

• Natural end conditions: vanishing 2nd derivatives at the endpoints
Special Case: Natural Cubic Splines

- Endpoint interpolation

- Uniform knot vector

\[ T = (0, 0, 0, 0, 1, \ldots, m - 3, m - 2, m - 2, m - 2, m - 2) \]

\[
\begin{pmatrix}
3/2 & -1/2 \\
1/4 & 7/12 & 1/6 \\
1/6 & 4/6 & 1/6 \\
\vdots
\end{pmatrix}
\begin{pmatrix}
d_1 \\
d_2 \\
\vdots \\
d_{m-2}
\end{pmatrix}
= 
\begin{pmatrix}
c_0 \\
c_1 \\
\vdots \\
c_{m-3}
\end{pmatrix}
\]

and \( d_0 = c_0, \; d_{m-1} = c_{m-3} \).

Natural cubic \( B \)-spline over uniform knot vector

FSP-Seminar, Graz, November 2005 34
Solve the overdetermined system of linear equations with the method of the Gaussian normal equations (for minimizing the error $||Ad - c||^2$):

$$||Ad - c||^2 \rightarrow \min \iff d = (A^T A)^{-1} A^T c$$

LSS-approximating cubic $B$–spline curve; uniform knot sequence
How to choose parameter values $x_i$ for data points $c_i$?

- Uniform (equidistant) parametrization ($x_i = i$) - disregards the position of $c_i$

- Take the positions of the data points into account, choose the distance of the parameters proportional to the distance of $c_i$:

$$s_i = ||c_i - c_{i-1}||^p, \quad i = 1, \ldots, m - 1, \quad p \in [0, 1]$$

$$x_0 = 0; \quad x_{i+1} = x_i + s_{i+1}, \quad i = 1, \ldots, m - 1$$

Wiggling interpolation curve (e.g. loops) due to parametrization ($x_i = i$)
$s_i = ||c_i - c_{i-1}||^p, \quad i = 1, \ldots, m - 1, \quad p \in [0, 1]$ 

The parametrization is called

- uniform for $p = 0$
- centripetal for $p = 1/2$ – simulates a driving car through the points
- chordal for $p = 1$
Influence of Parametrization

$B$–spline interpolation with different parametrization

uniform  chordal  centripetal  chordal  centripetal
Interpolation with $B$–splines is global: Changing the position of a single data point changes the shape of the interpolating curve globally.