

Geometrische Datenverarbeitung

Ulrich Reif

Sommersemester 2010

Aufgaben:

- **Modellierung** Methoden zur interaktiven Erzeugung von Kurven und Flächen auf dem Computer.
- **Visualisierung** Darstellung geometrischer Objekte auf dem Computer.
- **Interpolation/Approximation** Algorithmen zur Interpolation und Approximation von Messdaten für Visualisierung, Analyse und Rekonstruktion.
- **Geometriebeschreibung** Darstellung der Geometrie technischer (*Auto*) und fiktiver (*Dino*) Objekte für Analyse, Simulation und Unterhaltung.
- **Bildverarbeitung/Datenkompression**

Anwendungsbereiche:

- CAD/CAM-Systeme
- Qualitätskontrolle
- Film- und Werbeindustrie
- FE-Systeme

Methoden:

- Polynome
- Bézierkurven und -flächen
- Splinekurven und -flächen ("NURBS")
- Differentialgeometrie
- Variationsansätze
- Unterteilungsalgorithmen
- Multiskalenmethoden ("Wavelets")

Literatur:

- *J. Hoschek und D. Lasser*, Grundlagen der geometrischen Datenverarbeitung, Teubner
- *H. Prautzsch, W. Boehm und M. Paluszny*, Bézier- and B-Spline Techniques, Springer
- *G. Farin*, Curves and Surfaces for CAGD, Academic Press
- *L. Schumaker*, Spline Functions Basic Theory, Krieger
- *C. de Boor*, A Practical Guide to Splines, Springer

1. Polynome

1.1 Grundlagen

Vorteile:

- einfache Struktur
- Vektorraum
- schnelle Auswertung
- leicht zu integrieren und differenzieren
- ...

Definition: Polynom der Ordnung n

Ein (reelles algebraisches) *Polynom der Ordnung n* ist eine Linearkombination der Monome t^{n-1}, \dots, t^0 mit Koeffizienten $a_1, \dots, a_n \in \mathbb{R}$,

$$p(t) = \sum_{i=1}^n a_i t^{n-i} = a_1 t^{n-1} + a_2 t^{n-2} + \dots + a_{n-1} t + a_n .$$

Dies ist die *Monomform* von p . Der Raum aller Polynome der Ordnung n wird mit \mathbb{P}_n bezeichnet oder, wenn das Definitionsgebiet auf die Menge $D \subset \mathbb{R}$ eingeschränkt ist, mit $\mathbb{P}_n(D)$.

Theorem: Nullstellen von Polynomen

Ein Polynom ist genau dann identisch 0, wenn alle Koeffizienten der Monomform 0 sind. Ein von 0 verschiedenes Polynom der Ordnung n hat weniger als n Nullstellen.

Algorithmus: $p = \text{PolMonVal}(A, t)$

Wert p der Monomform mit Koeffizienten A in t mittels Horner-Schema.

```
 $n = \text{Length}(A)$   
 $p = A(1)$   
for  $i = 2 : n$   
     $p = p * t + A(i)$   
end
```

Definition: Taylor-Form

Die *Taylor-Form* eines Polynoms p der Ordnung n im Entwicklungspunkt t_1 ist

$$p(t) = \sum_{i=1}^n a_i (t - t_1)^{n-i}$$
$$= a_1 (t - t_1)^{n-1} + \dots + a_{n-1} (t - t_1) + a_n,$$

wobei $(n - i)! a_i = D^{n-i} p(t_1)$.

Das *Taylorpolynom* der Ordnung n einer Funktion f im Entwicklungspunkt t_1 ist gegeben durch

$$D^{n-i} f(t_1) = D^{n-i} p(t_1) = (n - i)! a_i, \quad i = 1 : n.$$

1.2 Interpolation

Definition: Lagrange-Form

Die *Lagrange-Form* des Polynoms $p \in \mathbb{P}_n$, das die Daten f_1, \dots, f_n in den paarweise verschiedenen Stützstellen t_1, \dots, t_n interpoliert, ist

$$p(t) = \sum_{i=1}^n f_i q_i(t), \quad q_i(t) = \prod_{j=1:n \setminus i} \frac{t - t_j}{t_i - t_j}.$$

Für mehrfache Interpolationspunkte schreibt man

$$t \mu = \underbrace{t, \dots, t}_{\mu \text{ mal}}.$$

Jedem Index i wird die Anzahl der *identischen Nachfolger* in der Folge t_i zugewiesen,

$$\#i := \#\{j \in \mathbb{N} : j > i \text{ und } t_j = t_i\}.$$

Beispiel: $T = [4\$3, 0, 1\$2] = [4, 4, 4, 0, 1, 1]$
 $\Rightarrow \#1 = 2, \#2 = \#5 = 1, \#3 = \#4 = \#6 = 0.$

Ein Vektor $T = [t_1, \dots, t_n]$ heißt *gruppiert*, wenn es paarweise verschiedene Zahlen $s_1, \dots, s_k \in \mathbb{R}$ gibt mit

$$T = [s_1 \$ \mu_1, s_2 \$ \mu_2, \dots, s_k \$ \mu_k].$$

Definition: Interpolationsproblem

Sei $T = [t_1, \dots, t_n]$ ein gruppierter Vektor von *Stützstellen*. Dann besteht das *Interpolationsproblem* zu den Daten (T, F) darin, ein Polynom $p \in \mathbb{P}_n$ zu finden mit

$$D^{\#i}p(t_i) = f_i, \quad i = 1 : n.$$

Theorem: Lösbarkeit des Interpolationsproblems

Das Interpolationsproblem zu n Datenpunkten (T, F) ist stets eindeutig durch ein Polynom $p \in \mathbb{P}_n$ lösbar.

Für die praktische Berechnung von p verwendet man *dividierte Differenzen*. Damit erhält man p in *Newton-Form*. Achtung, für große Werte von n neigt das Interpolationspolynom zu Oszillationen.

Die MATLAB-Befehle für Auswertung und Interpolation heißen `polyval` und `polyfit`.

1.4 Interpolationsfehler

Theorem: Interpolationsfehler

Für eine glatte Funktion f gilt für den Fehler des interpolierenden Polynoms p der Ordnung n zu den Stützstellen $T = [t_1, \dots, t_n]$ an der Stelle t_0

$$f(t_0) - p(t_0) = \frac{D^n f(s)}{n!} (t_0 - t_1) \cdots (t_0 - t_n),$$

wobei $s \in \text{conv}(t_0, \dots, t_n)$.

Einfluss der einzelnen Größen auf den Fehler:

- Eine hohe Ordnung n ist i.A. notwendig, aber nicht hinreichend für einen kleinen Fehler.
- Der Faktor $D^n f(s)$ kann i.A. nur abgeschätzt, aber nicht beeinflusst werden.
- Das Polynom $w_n(t_0) := (t_0 - t_1) \cdots (t_0 - t_n)$ hängt von der Wahl der Stützstellen ab. Für das Intervall $[-1, 1]$ stellen die *Chebyshevunkte*

$$T_n^{\text{ch}} := \sin([-1 + h : 2h : 1 - h]\pi/2), \quad h = \frac{1}{n}$$

eine optimale Wahl dar. Hier hat das Maximum von $|w_n|$ den minimal möglichen Wert, nämlich 2^{1-n} . Skalierung auf das Intervall $[a, b]$ ergibt die Fehlerabschätzung

$$\max_{t \in [a, b]} |f(t) - p(t)| \leq \frac{(b-a)^n}{n! 2^{2n-1}} \max_{s \in [a, b]} |D^n f(s)|.$$

1.5 Polynomiale Approximation

Häufig werden Daten nicht interpoliert, sondern approximiert. Das heißt, das gesuchte Polynom geht nicht notwendigerweise exakt durch die Datenpunkte, sondern lediglich möglichst nahe daran vorbei,

$$\sum_{i=1}^m |f_i - p(t_i)|^2 \rightarrow \min, \quad m \geq n.$$

Dieser Ansatz wird im Folgenden etwas verallgemeinert.

Definition: \mathbb{P}_n -Positivität für Skalarprodukt und Norm

Ein nicht-negative symmetrische Bilinearform $\langle \cdot, \cdot \rangle$ heißt \mathbb{P}_n -positives Skalarprodukt, wenn

$$\langle p, p \rangle > 0, \quad p \in \mathbb{P}_n \setminus \{0\}.$$

Die zugehörige \mathbb{P}_n -positive Norm ist

$$\|f\| := \sqrt{\langle f, f \rangle}.$$

Definition: Beste Approximation

Die *beste Approximation* $p \in \mathbb{P}_n$ der Funktion f bezüglich der \mathbb{P}_n -positiven Norm $\|\cdot\|$ ist definiert durch

$$\|f - p\| = \min_{r \in \mathbb{P}_n} \|f - r\|.$$

Übliche \mathbb{P}_n -positive Skalarprodukte sind

$$\langle f, g \rangle = \sum_{i=1}^m f(t_i)g(t_i), \quad n \leq m$$

und

$$\langle f, g \rangle = \int_a^b f(t)g(t)w(t) dt,$$

wobei die *Gewichtsfunktion* w fast überall positiv ist.

Theorem: Approximation

Sei $\langle \cdot, \cdot \rangle$ ein \mathbb{P}_n -positives Skalarprodukt. Dann existiert genau ein best-approximierendes Polynom $p \in \mathbb{P}_n$ bezüglich der zugehörigen Norm $\| \cdot \|$. Dies ist charakterisiert durch die *Orthogonalitätsrelation*

$$\langle f - p, q \rangle = 0, \quad q \in \mathbb{P}_n.$$

Aufgrund der Linearität genügt es, die Orthogonalitätsrelation für eine Basis $\{q_1, \dots, q_n\}$ von \mathbb{P}_n zu gewährleisten. Dies ergibt ein LGS für die Koeffizienten A des gesuchten Polynoms $p = \sum_i a_i q_i$,

$$\sum_{j=1:n} \langle q_i, q_j \rangle a_j = \langle f, q_i \rangle, \quad i = 1 : n.$$

Die $(n \times n)$ -Matrix des Systems ist *positiv definit* und heißt *Gramsche Matrix*.

1.7 Satz von Weierstrass

Theorem: Satz von Weierstrass

Sei $f : [0, 1] \rightarrow \mathbb{R}$ eine stetige Funktion, dann gibt es für alle $\varepsilon > 0$ ein Polynom p , so dass

$$\max_{t \in [0, 1]} |f(t) - p(t)| < \varepsilon.$$

Binomialkoeffizienten:

$$\binom{n}{i} = \begin{cases} \frac{n!}{(n-i)!i!} & \text{falls } 0 \leq i \leq n \\ 0 & \text{sonst.} \end{cases}$$

Rekursion:

$$\binom{n}{i} = \binom{n-1}{i-1} + \binom{n-1}{i}, \quad n \in \mathbb{N}, \quad i \in \mathbb{Z}.$$

Binomischer Satz:

$$(s + t)^n = \sum_i \binom{n}{i} s^{n-i} t^i.$$

Bernstein-Polynome der Ordnung $n + 1$:

$$b_{i+1}^{n+1}(t) := \binom{n}{i} (1-t)^{n-i} t^i, \quad i \in \mathbb{Z}.$$

Bernstein-Identität:

$$\frac{t(1-t)}{n-1} = \sum_i (t_i - t)^2 b_i^n(t), \quad t_i := \frac{i-1}{n-1}.$$

Bernstein-Operator:

$$B^n f := \sum_i f(t_i) b_i^n \in \mathbb{P}_n.$$

f ist gleichmäßig stetig, es gibt also für ein beliebiges $\varepsilon > 0$ ein $\delta > 0$ so, dass $|f(s) - f(t)| < \varepsilon/2$ für alle $|s - t| < \delta$.
 Ferner sei $\varrho := \max_t |f(t)|$.

Spalte den Fehler

$$f - B^n f = \sum_{i=1}^n (f - f(t_i)) b_i^n$$

für festes t in zwei Teilsummen Δ_I, Δ_J auf mit Indizes

$$I := \{i : |t - t_i| < \delta\} \quad \text{und} \quad J := \{i : |t - t_i| \geq \delta\}.$$

Die Bernsteinpolynome bilden auf $[0, 1]$ eine nicht-negative Partition der 1,

$$\sum_k b_k^n(t) \equiv 1, \quad b_k^n(t) \geq 0, \quad t \in [0, 1].$$

Es folgt damit

$$|\Delta_I(t)| \leq \sum_{i \in I} |f(t) - f(t_i)| b_i^n(t) \leq \frac{\varepsilon}{2} \sum_i b_i^n(t) = \frac{\varepsilon}{2}$$

und

$$\begin{aligned} |\Delta_J(t)| &\leq \sum_{i \in J} |f(t) - f(t_i)| b_i^n(t) \\ &\leq \frac{2\varrho}{\delta^2} \sum_i (t_i - t)^2 b_i^n(t) = \frac{2\varrho t(1-t)}{(n-1)\delta^2}. \end{aligned}$$

Wählt man $n - 1 \geq \varrho/(\varepsilon\delta^2)$, so gilt

$$|f(t) - B^n f(t)| \leq |\Delta_I(t)| + |\Delta_J(t)| \leq \varepsilon.$$

2. Bézierkurven

2.1 Grundlagen

Sei $\mathbf{P} := [\mathbf{p}_1; \dots; \mathbf{p}_n]$ ein Spaltenvektor von Punkten in \mathbb{R}^d und $W := [w_1, \dots, w_n]$ ein Zeilenvektor von Gewichten. Der Punkt $\mathbf{p} = \sum_i w_i \mathbf{p}_i = W\mathbf{P}$ ist eine

- *Linearkombination* von \mathbf{P} .
- *Affinkombination* von \mathbf{P} , wenn $\sum_i w_i = 1$.
- *Konvexkombination* von \mathbf{P} , wenn $\sum_i w_i = 1$ und $W \geq 0$.

Linearkombinationen sind linear invariant.

Affinkombinationen sind affin invariant.

Eine Teilmenge $M \subset \mathbb{R}^d$ heißt *konvex*, wenn mit zwei beliebigen Punkten $\mathbf{p}_1, \mathbf{p}_2 \in M$ auch deren geradlinige Verbindung in M liegt,

$$(1 - t)\mathbf{p}_1 + t\mathbf{p}_2 \in M, \quad t \in [0, 1].$$

Die konvexe Hülle $\text{conv } M$ einer Menge $M \subset \mathbb{R}^d$ ist die kleinste konvexe Obermenge von M .

Theorem: Konvexe Hülle

Sei $\mathbf{P} := [\mathbf{p}_1; \dots; \mathbf{p}_n]$, dann ist $\text{conv } \mathbf{P}$ gleich der Menge aller Konvexkombinationen von \mathbf{P} . Für $\mathbf{p}_i \in \mathbb{R}^2$ wird $\text{conv } \mathbf{P}$ von einem Polygon $\mathbf{p}_{i_1}, \dots, \mathbf{p}_{i_m}$ berandet.

Eine *polynomiale Kurve* der Ordnung n ist gegeben durch

$$\mathbf{c}(t) = Q(t)\mathbf{P} = \sum_{i=1}^n q_i(t)\mathbf{p}_i.$$

Dabei ist $Q(t) = [q_1(t), \dots, q_n(t)]$ ein Zeilenvektor von Basisfunktionen des Polynomraums \mathbb{P}_n und $\mathbf{P} = [\mathbf{p}_1; \dots; \mathbf{p}_n]$ ein Spaltenvektor von Koeffizienten.

Schränkt man das Definitionsgebiet der Kurve auf das Intervall $[0, 1]$ ein und wählt man die Bernsteinpolynome als Basis, dann ist

$$\mathbf{c}(t) = \sum_i b_i^n(t)\mathbf{p}_i = B^n(t)\mathbf{P}, \quad t \in [0, 1], \quad \mathbf{p}_i \in \mathbb{R}^d$$

eine *Bézierkurve* in \mathbb{R}^d . Die Koeffizienten \mathbf{P} werden *Kontrollpunkte* genannt und deren geradlinige Verbindung bildet das *Kontrollpolygon*.

- **Affine Invarianz:** Für eine affine Abbildung A gilt

$$A(\mathbf{c}) = B^n A(\mathbf{P}).$$

- **Konvexe Hülle:**

$$\mathbf{c}(t) \in \text{conv}(\mathbf{P}), \quad t \in [0, 1].$$

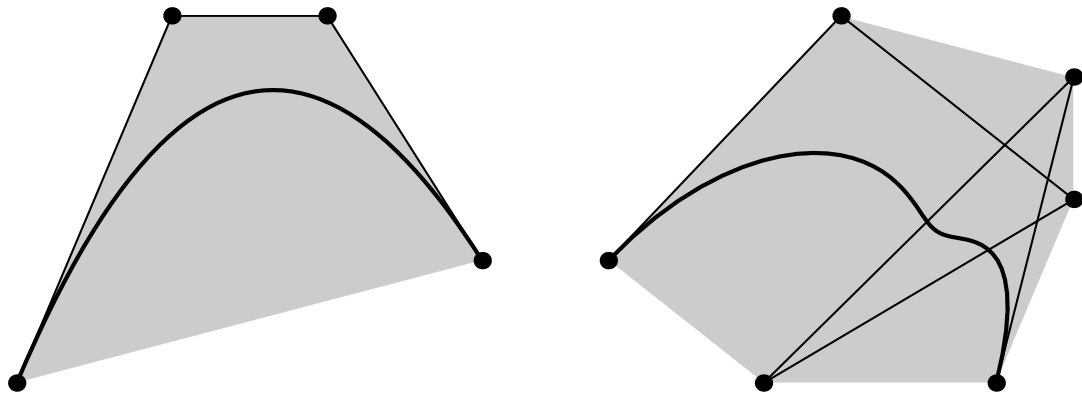
- **Randinterpolation:**

$$\mathbf{c}(0) = \mathbf{p}_1, \quad \mathbf{c}(1) = \mathbf{p}_n$$

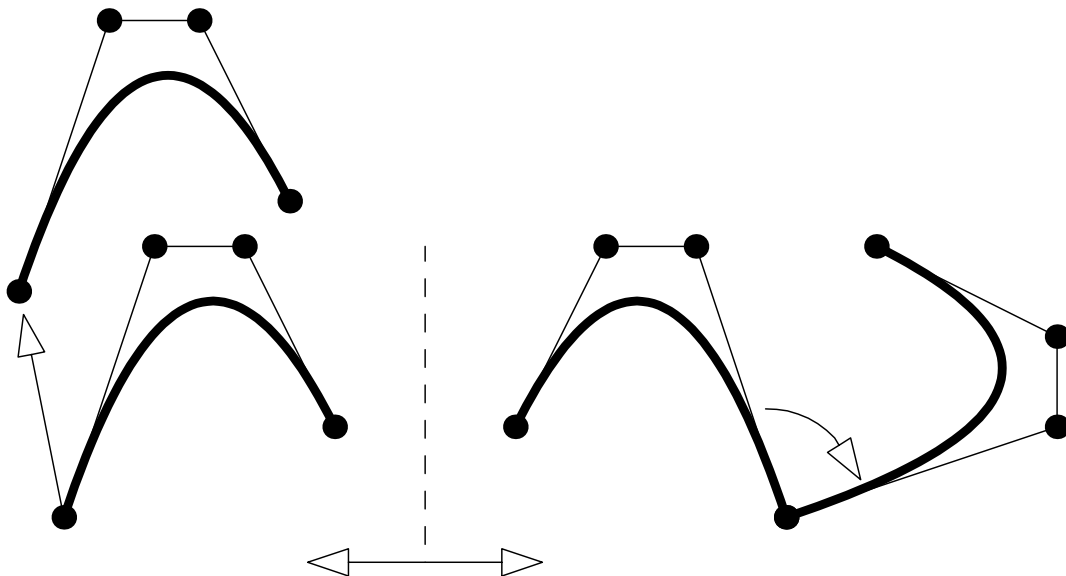
- **Symmetrie:**

$$\mathbf{c}(1-t) = \sum_i b_i^n(t)\mathbf{p}_{n+1-i}$$

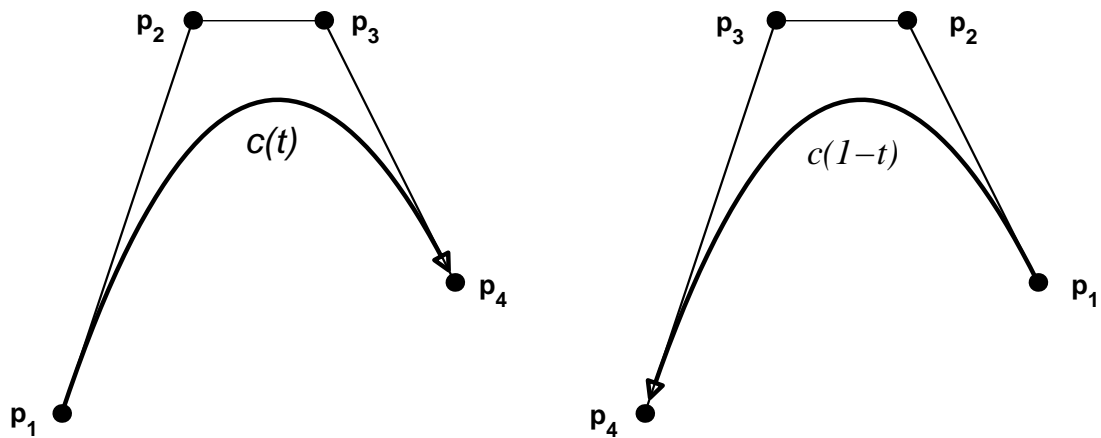
- Konvexe Hülle



- Affine Invarianz

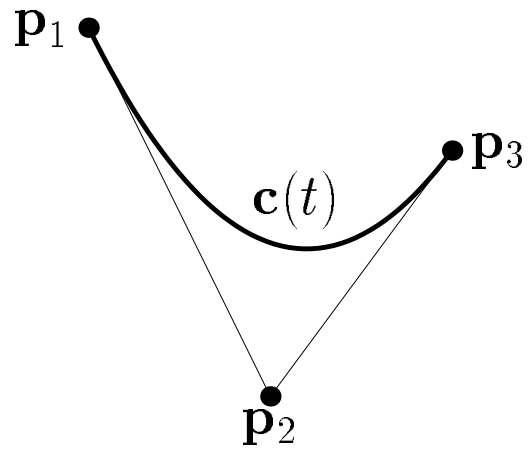
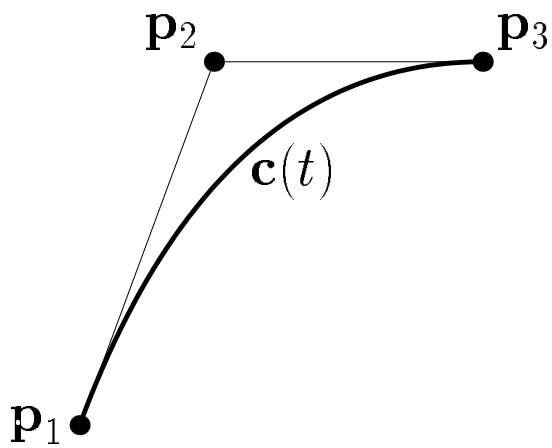
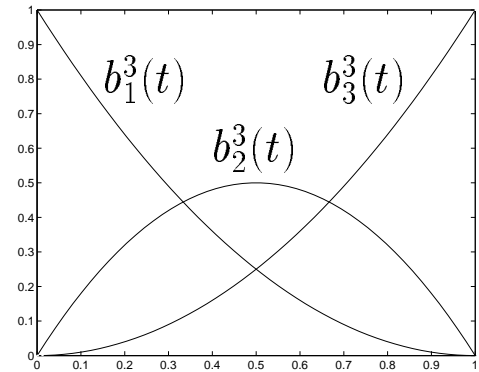


- Symmetrie

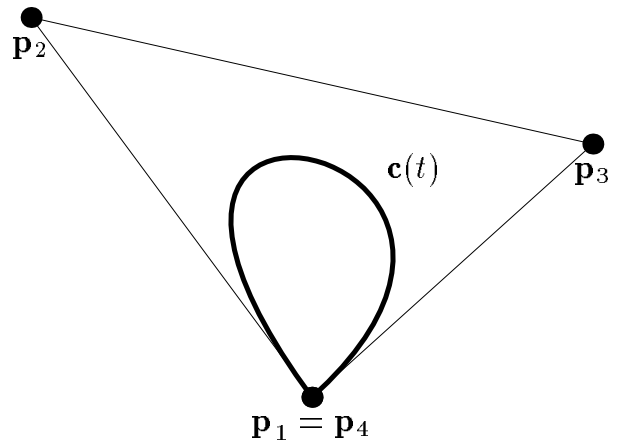
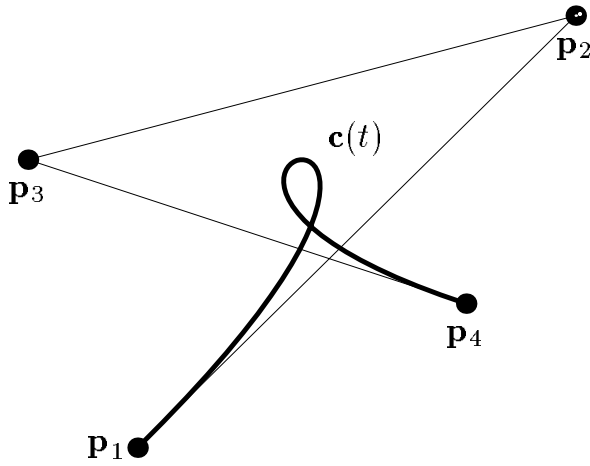
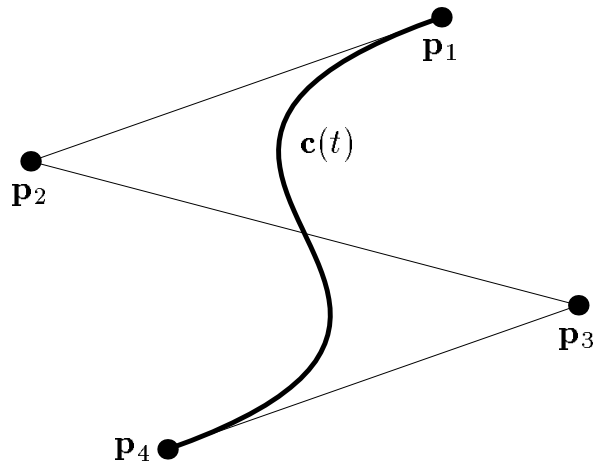
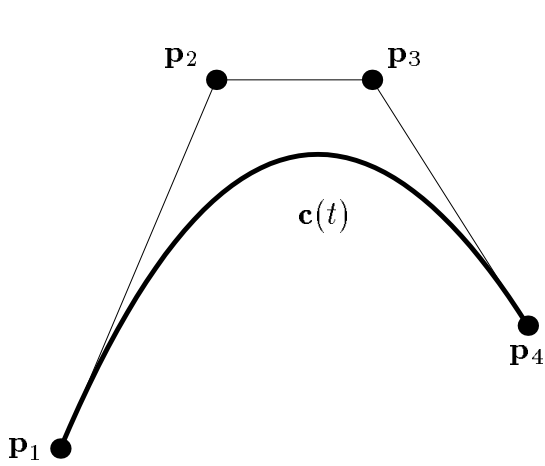
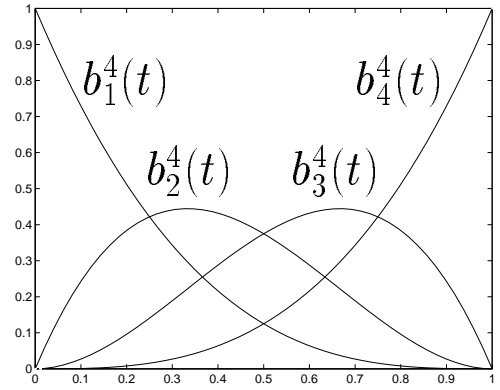


Beispiele:

$$n = 3 \Rightarrow \begin{aligned} b_1^3(t) &= (1 - t)^2 \\ b_2^3(t) &= 2(1 - t)t \\ b_3^3(t) &= t^2 \end{aligned}$$



$$n = 4 \Rightarrow \begin{aligned} b_1^4(t) &= (1 - t)^3 \\ b_2^4(t) &= 3(1 - t)^2 t \\ b_3^4(t) &= 3(1 - t) t^2 \\ b_4^4(t) &= t^3 \end{aligned}$$



2.2 Basiswechsel

Theorem: Basiswechsel

$$b_{j+1}^{n+1}(t) = \sum_{k=j}^n (-1)^{k+j} \binom{k}{j} \binom{n}{k} t^k$$
$$t^j = \sum_{k=j}^n \binom{k}{j} / \binom{n}{j} b_{k+1}^{n+1}(t) .$$

Sei $M^n := [t^{n-1}, \dots, t, 1]$ der Zeilenvektor der Monombasis, dann lässt sich der Basiswechsel darstellen als

$$M^n(t) = B^n(t)T_M^n, \quad B^n(t) = M^n(t)T_B^n.$$

Dabei sind T_M^n und T_B^n zueinander inverse Dreiecksmatrizen der Dimension $n \times n$.

Beispiele:

$$T_M^2 = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$$

$$T_B^2 = \begin{bmatrix} -1 & 1 \\ 1 & 0 \end{bmatrix}$$

$$T_M^3 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1/2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$T_B^3 = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 2 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$T_M^4 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1/3 & 1 \\ 0 & 1/3 & 2/3 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$$T_B^4 = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Sei

$$\mathbf{c}(t) = B^n(t)\mathbf{P} = M^n(t)\mathbf{A}$$

eine polynomiale Kurve in Bézier- bzw. Monomdarstellung, dann gilt

$$\mathbf{A} = T_B^n \mathbf{P}, \quad \mathbf{P} = T_M^n \mathbf{A}.$$

Beispiel:

$$\mathbf{c}(t) = t^3 \begin{bmatrix} 0 \\ -8 \end{bmatrix}^t + t^2 \begin{bmatrix} -6 \\ 6 \end{bmatrix}^t + t \begin{bmatrix} 6 \\ 3 \end{bmatrix}^t + \begin{bmatrix} 1 \\ 1 \end{bmatrix}^t$$

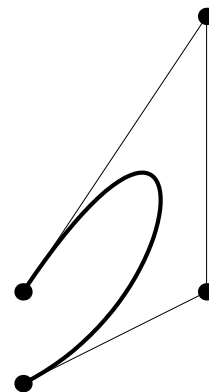
$$\mathbf{A} = \begin{bmatrix} 0 & -6 & 6 & 1 \\ -8 & 6 & 3 & 1 \end{bmatrix}^t, \quad \mathbf{P} = T_M^4 \mathbf{A} = \begin{bmatrix} 1 & 3 & 3 & 1 \\ 1 & 2 & 5 & 2 \end{bmatrix}^t$$

$$\mathbf{c}(t) = b_1^4(t) \begin{bmatrix} 1 \\ 1 \end{bmatrix}^t + b_2^4(t) \begin{bmatrix} 3 \\ 2 \end{bmatrix}^t + b_3^4(t) \begin{bmatrix} 3 \\ 5 \end{bmatrix}^t + b_4^4(t) \begin{bmatrix} 1 \\ 2 \end{bmatrix}^t$$

$$\mathbf{c}(1-t) = b_1^4(t) \begin{bmatrix} 1 \\ 2 \end{bmatrix}^t + b_2^4(t) \begin{bmatrix} 3 \\ 5 \end{bmatrix}^t + b_3^4(t) \begin{bmatrix} 3 \\ 2 \end{bmatrix}^t + b_4^4(t) \begin{bmatrix} 1 \\ 1 \end{bmatrix}^t$$

$$\mathbf{c}(0) = \begin{bmatrix} 1 \\ 1 \end{bmatrix}^t, \quad \mathbf{c}(1) = \begin{bmatrix} 1 \\ 2 \end{bmatrix}^t$$

$$\mathbf{c}'(0) = \begin{bmatrix} 6 \\ 3 \end{bmatrix}^t, \quad \mathbf{c}'(1) = \begin{bmatrix} -6 \\ -9 \end{bmatrix}^t$$



2.3 Auswertung

Es gilt

$$b_j^n(t) = (1-t)^{n-1} \binom{n-1}{j-1} u^{j-1}, \quad u := t/(1-t)$$

$$b_j^n(t) = t^{n-1} \binom{n-1}{j-1} v^{n-j+1}, \quad v := (1-t)/t.$$

Entsprechend gilt für eine Bézierkurve $\mathbf{c}(t) = B^n(t)\mathbf{P}$

$$\mathbf{c}(t) = (1-t)^{n-1} \sum_{j=1}^n \binom{n-1}{n-j} \mathbf{P}_{n-j+1} u^{n-j}$$

$$\mathbf{c}(t) = t^{n-1} \sum_{j=1}^n \binom{n-1}{j-1} \mathbf{P}_j v^{n-j}$$

Aus Stabilitätsgründen ist die erste Form für $t \in [0, 1/2]$ und die zweite Form für $t \in [1/2, 1]$ vorzuziehen.

Alternative: Es gilt die Rekursion

$$b_j^1(t) \equiv \delta_{j,1}$$

$$b_j^n(t) = (1-t) b_j^{n-1}(t) + t b_{j-1}^{n-1}(t)$$

und entsprechend für eine Bézierkurve $\mathbf{c}(t) = B^n(t)\mathbf{P}$

$$\begin{aligned} \mathbf{c}(t_0) &= \sum_j b_j^n(t_0) \mathbf{p}_j = \sum_j b_j^{n-1}(t_0) ((1-t_0) \mathbf{p}_j + t_0 \mathbf{p}_{j+1}) \\ &=: \sum_j b_j^{n-1}(t_0) \tilde{\mathbf{p}}_j. \end{aligned}$$

Die Kurven $\mathbf{c}(t) = B^n(t)\mathbf{P}$ und $\tilde{\mathbf{c}}(t) = B^{n-1}(t)\tilde{\mathbf{P}}$ stimmen für $t = t_0$ überein. $\tilde{\mathbf{P}}$ erhält man aus \mathbf{P} durch Affinkombination aufeinanderfolgender Kontrollpunkte.

Theorem: De Casteljau-Algorithmus

Die Rekursion

$$\mathbf{P}^{(1)} = \mathbf{P}$$

for $k = 1 : n - 1$

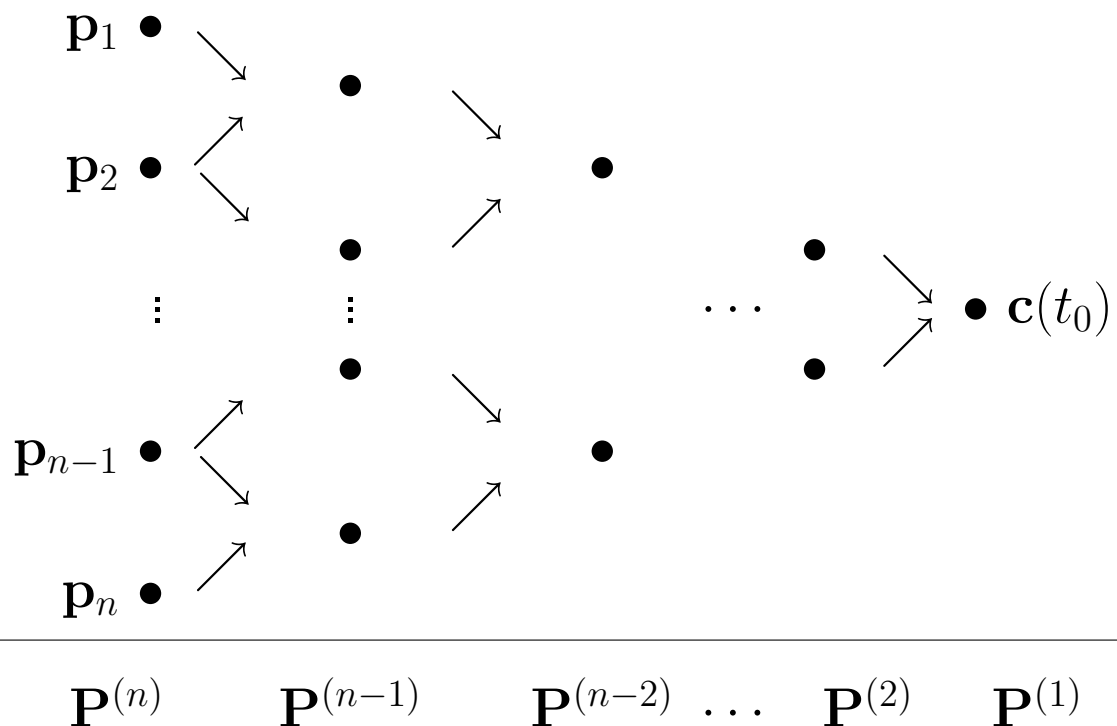
$I = 1 : n - k$

$$\mathbf{P}^{(k+1)} = (1 - t_0) \mathbf{P}^{(k)}(I) + t_0 \mathbf{P}^{(k)}(I + 1)$$

end

berechnet den Wert $\mathbf{P}^{(n)}(1) = B^n(t_0)\mathbf{P} = \mathbf{c}(t_0)$.

Schematische Darstellung:



Der Pfeil \nearrow steht für Multiplikation mit t_0 .

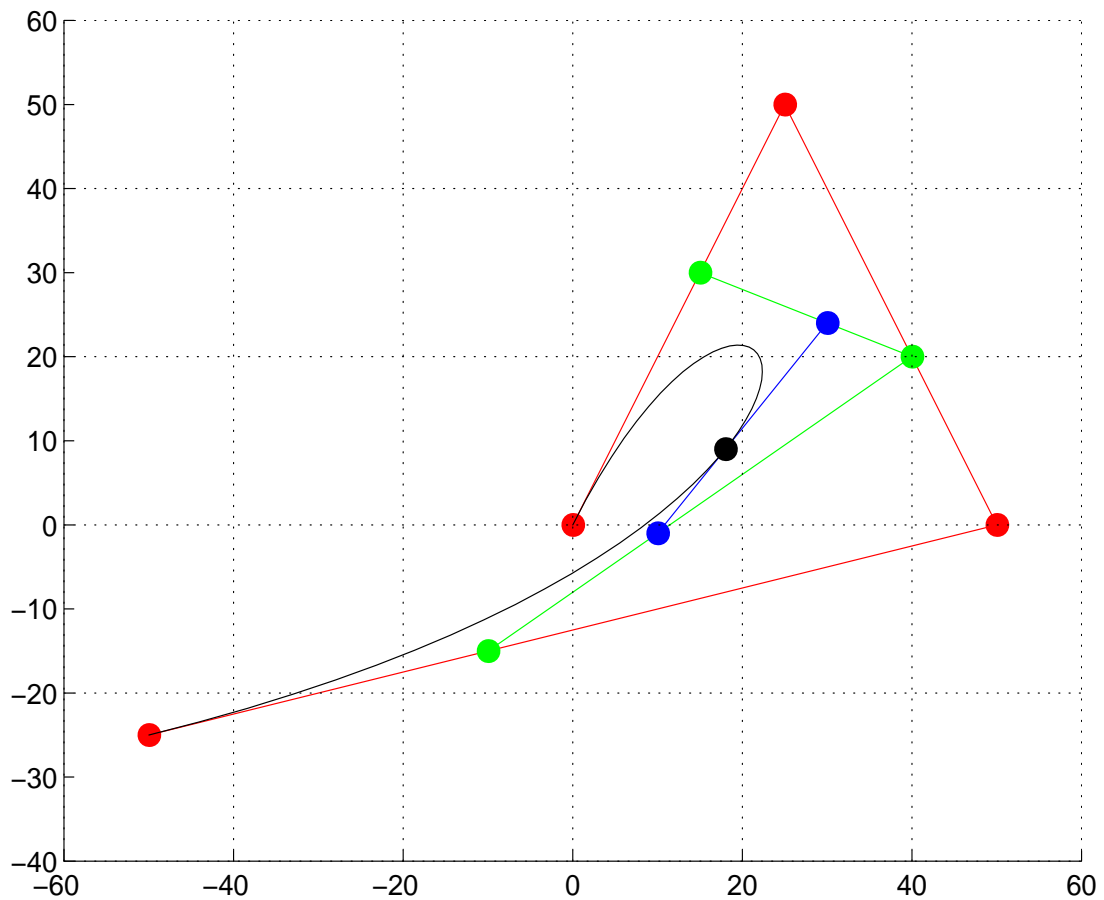
Der Pfeil \searrow steht für Multiplikation mit $(1 - t_0)$.

Die Kontrollpunkte $\mathbf{P}^{(k)}$ definieren eine Bézierkurve $\mathbf{c}^{(k)} := B^k \mathbf{P}^{(k)}$ der Ordnung k mit $\mathbf{c}^{(k)}(t_0) = \mathbf{c}(t_0)$.

Beispiel: De Casteljau für eine kubische Bézierkurve

$$\mathbf{P} = \begin{bmatrix} 0 & 25 & 50 & -50 \\ 0 & 50 & 0 & -25 \end{bmatrix}^t, \quad t_0 = 3/5$$

0				0			
	15				30		
25		30		50		24	
	40		18		20		9
50		10		0		-1	
	-10				-15		
-50				-25			



2.4 Differenziation und Integration

Für die Ableitung eines Bernsteinpolynoms gilt

$$Db_{j+1}^{n+1}(t) = n(b_j^n(t) - b_{j+1}^n(t)) .$$

Entsprechend gilt für eine Bézierkurve $\mathbf{c}(t) = B^{n+1}(t)\mathbf{P}$:

$$\mathbf{c}'(t) = n \sum_j b_j^n(t) (\mathbf{p}_{j+1} - \mathbf{p}_j) =: n B^n(t) \Delta \mathbf{P} .$$

Δ ist der *Vorwärtsdifferenzenoperator*.

Für das Integral eines Bernsteinpolynoms gilt

$$D^{-1}b_j^n(t) := \int_0^t b_j^n(\tau) d\tau = \frac{1}{n} \sum_{k>j} b_k^{n+1}(t) , \quad j \geq 1 .$$

Entsprechend gilt für eine Bézierkurve $\mathbf{c}(t) = B^n(t)\mathbf{P}$:

$$D^{-1}\mathbf{c}(t) = \frac{1}{n} \sum_j b_j^{n+1}(t) \sum_{k<j} \mathbf{p}_k =: \frac{1}{n} B^{n+1} \Delta^{-1} \mathbf{P} .$$

Δ^{-1} ist der *Summenoperator* und es gilt

$$\Delta \Delta^{-1} = \text{Id} .$$

Für beliebige $\nu \leq n$ und $\mathbf{c} = B^{n+1}\mathbf{P}$ gilt

$$D^\nu \mathbf{c}(t) = \frac{n!}{(n-\nu)!} B^{n+1-\nu}(t) \Delta^\nu \mathbf{P} .$$

Randpunkte: An den Randpunkten $t \in \{0, 1\}$ gilt für die Ableitungen

$$D^\nu b_j^n(0) = 0 \Leftrightarrow \nu \leq j - 2$$

$$D^\nu b_j^n(1) = 0 \Leftrightarrow \nu \leq n - 1 - j.$$

Folglich hängt für die Kurve $\mathbf{c}(t) = B^n(t)\mathbf{P}$

$$D^\nu \mathbf{c}(0) \text{ nur von } \mathbf{p}_1, \dots, \mathbf{p}_{\nu+1}$$

$$D^\nu \mathbf{c}(1) \text{ nur von } \mathbf{p}_{n-\nu}, \dots, \mathbf{p}_n$$

ab. Insbesondere gilt

$$\mathbf{c}(0) = \mathbf{p}_1, \quad \mathbf{c}'(0) = (n-1)(\mathbf{p}_2 - \mathbf{p}_1)$$

$$\mathbf{c}(1) = \mathbf{p}_n, \quad \mathbf{c}'(1) = (n-1)(\mathbf{p}_n - \mathbf{p}_{n-1}).$$

Beispiel (siehe Seite 20):

$$\mathbf{c}(t) = B^4(t) \begin{bmatrix} 1 & 3 & 3 & 1 \\ 1 & 2 & 5 & 2 \end{bmatrix}^t$$

$$\Delta \mathbf{P} = \begin{bmatrix} 2 & 0 & -2 \\ 1 & 3 & -3 \end{bmatrix}^t, \quad \Delta^{-1} \mathbf{P} = \begin{bmatrix} 0 & 1 & 4 & 7 & 8 \\ 0 & 1 & 3 & 8 & 10 \end{bmatrix}^t$$

$$D\mathbf{c}(t) = B^3(t) \begin{bmatrix} 6 & 0 & -6 \\ 3 & 9 & -9 \end{bmatrix}^t$$

$$D^{-1}\mathbf{c}(t) = B^5(t) \begin{bmatrix} 0 & 1/4 & 1 & 7/4 & 2 \\ 0 & 1/4 & 3/4 & 2 & 5/2 \end{bmatrix}^t$$

2.5 Graderhöhung

... sollte eigentlich „Ordnungserhöhung“ heißen ...

Schreibe eine Bézierkurve der Ordnung n als Bézierkurve der Ordnung $n + 1$,

$$\sum_j b_j^n(t) \mathbf{p}_j = \sum_j b_j^{n+1}(t) \tilde{\mathbf{p}}_j.$$

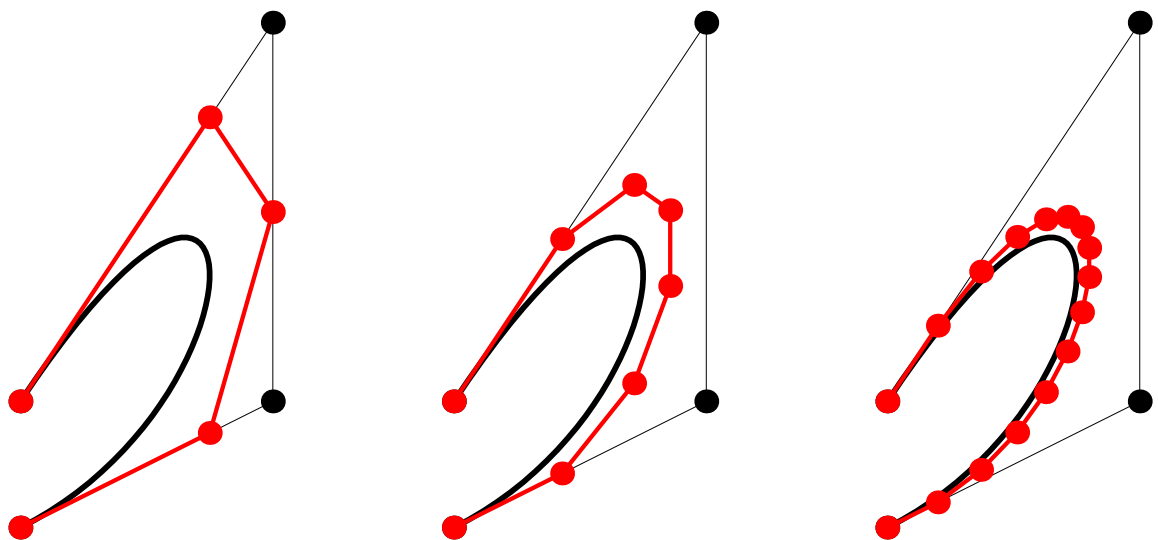
Multiplikation der linken Seite mit $(1 - t) + t$ ergibt:

Die Bézierkurven $\mathbf{c} = B^n \mathbf{P}$ und $\tilde{\mathbf{c}} = B^{n+1}(t) \tilde{\mathbf{P}}$ stimmen genau dann überein, wenn

$$\tilde{\mathbf{p}}_{j+1} = \left(1 - \frac{j}{n}\right) \mathbf{p}_{j+1} + \frac{j}{n} \mathbf{p}_j, \quad j = 0 : n.$$

$\tilde{\mathbf{P}}$ liegt in der konvexen Hülle von \mathbf{P} , also

$$\mathbf{c}(t) \in \text{conv}(\tilde{\mathbf{P}}) \subset \text{conv}(\mathbf{P}).$$



Der Übergang von \mathbf{P} nach $\tilde{\mathbf{P}}$ ist linear und kann durch eine Matrix E_n beschrieben werden,

$$\mathbf{c} = B^n \mathbf{P} = B^{n+1} E_n \mathbf{P} .$$

Es ist z.B.

$$E_3 = \frac{1}{3} \begin{bmatrix} 3 & 0 & 0 \\ 1 & 2 & 0 \\ 0 & 2 & 1 \\ 0 & 0 & 3 \end{bmatrix}, \quad E_4 = \frac{1}{4} \begin{bmatrix} 4 & 0 & 0 & 0 \\ 1 & 3 & 0 & 0 \\ 0 & 2 & 2 & 0 \\ 0 & 0 & 3 & 1 \\ 0 & 0 & 0 & 4 \end{bmatrix} .$$

Für mehrfache Graderhöhung schreiben wir

$$E_{m-1} \cdots E_{n+1} E_n =: E_n^m .$$

2.6 Gradreduktion

Problem: Bestimme Kontrollpunkte $\tilde{\mathbf{P}}$ so, dass

$$\|B^n \mathbf{P} - B^m \tilde{\mathbf{P}}\| \rightarrow \min, \quad 1 \leq m < n .$$

Das Resultat hängt i.A. von der gewählten Norm ab und die Approximation wird i.d.R. getrennt in den räumlichen Koordinaten vorgenommen.

Man kann z.B. die Abweichung der Kontrollpunkte als Maß für die Approximationsgüte verwenden,

$$\langle B^n P, B^n Q \rangle_E = \langle P, Q \rangle_2 = \sum_{i=1}^n p_i q_i .$$

Damit lautet das Approximationsproblem

$$\|P - E_m^n Q\|_2 \rightarrow \min .$$

Die Lösung ist gegeben durch

$$Q = \left((E_m^n)^t E_m^n \right)^{-1} (E_m^n)^t P =: R_n^m P$$

bzw. in \mathbb{R}^d durch

$$\mathbf{Q} = R_n^m \mathbf{P}.$$

Beispiel:

$$R_3^2 = \frac{1}{6} \begin{bmatrix} 5 & 2 & -1 \\ -1 & 2 & 5 \end{bmatrix}, \quad R_4^3 = \frac{1}{20} \begin{bmatrix} 19 & 3 & -3 & 1 \\ -5 & 15 & 15 & -5 \\ 1 & -3 & 3 & 19 \end{bmatrix}.$$

$$R_4^2 = R_3^2 R_4^3 = \frac{1}{10} \begin{bmatrix} 7 & 4 & 1 & -2 \\ -2 & 1 & 4 & 7 \end{bmatrix}.$$

Theorem: Gradreduktion

Gradreduktion bezüglich der L^2 -Norm $\| \cdot \|_L$ und der Koeffizientennorm $\| \cdot \|_E$ ist äquivalent.

Für die Gradreduktionsmatrizen R_n^m gilt daher

$$R_n^m = R_\ell^m R_n^\ell, \quad m \leq \ell \leq n$$

und

$$R_n^m = R_{m+1}^m \cdots R_{n-1}^{n-2} R_n^{n-1}.$$

Beweis: Sei $Q^n = [q_1^n, q_2^n, \dots, q_n^n]$ der Vektor der Lagrange-Polynome zu den Stützstellen $T = 1 : n$. Dann gilt

$$B^n P \in \mathbb{P}_m \quad \Leftrightarrow \quad Q^n P \in \mathbb{P}_m.$$

Die orthogonalen Komplemente $\mathbb{P}_{m,n}^E$ und $\mathbb{P}_{m,n}^L$ von \mathbb{P}_m in \mathbb{P}_n bezüglich $\langle \cdot, \cdot \rangle_E$ bzw. $\langle \cdot, \cdot \rangle_L$ stimmen überein. Dies folgt aus $\mathbb{P}_{m,n}^E \subset \mathbb{P}_{m,n}^L$, was sich wie folgt beweisen lässt: Um zu zeigen dass

$$\langle B^n W, t^i \rangle_L = 0, \quad B^n W \in \mathbb{P}_{m,n}^E, \quad 0 \leq i < m,$$

definieren wir den Vektor V^i gemäß

$$v_k^i := \int_0^1 b_k^n(t) t^i dt, \quad k = 1 : n.$$

Dann ist

$$\langle B^n W, t^i \rangle_L = \langle B^n W, B^n V^i \rangle_E = 0$$

genau dann, wenn $B^n V^i \in \mathbb{P}_m$. Wegen

$$v_k^i = \frac{\binom{n-1}{k-1}}{\binom{n+i-1}{k+i-1}} \int_0^1 b_{k+i}^{n+i}(t) dt = \frac{(n-1)!}{(n+i)!} \prod_{\ell=0}^{i-1} (k+\ell).$$

folgt dies aber aus $Q^n V^i \in \mathbb{P}_m$.

2.7 Operatorkalkül und de Casteljau-Algorithmus

Definiere den Shift-Operator

$$S : [\mathbf{p}_1; \mathbf{p}_2; \dots; \mathbf{p}_n] \mapsto [\mathbf{p}_2; \mathbf{p}_3; \dots; \mathbf{p}_n; 0]$$

und den Vektor

$$E := [1, 0, \dots, 0],$$

dann kann eine Béziékurve $\mathbf{c}(t) = B^{n+1}(t)\mathbf{P}$ auch dargestellt werden als

$$\mathbf{c}(t) = E((1-t) + tS)^n \mathbf{P}.$$

Daraus folgt zB einfach die Ableitungsformel

$$D\mathbf{c}(t) = nE((1-t) + tS)^{n-1}(S-1)\mathbf{P}$$

oder der de Casteljau-Algorithmus

$$\begin{aligned} \mathbf{c}(t) &= E((1-t) + tS)^{n-1}((1-t) + tS)\mathbf{P} \\ &= E((1-t) + tS)^{n-k}((1-t) + tS)^k \mathbf{P}. \end{aligned}$$

Die Kontrollpunkte in der k -ten Spalte des Schemas sind für den Auswertungspunkt $t = t_0$

$$\begin{aligned} \mathbf{P}^{(k)} &= ((1-t_0) + t_0S)\mathbf{P}^{(k-1)} \\ &= ((1-t_0) + t_0S)^{k-1}\mathbf{P}^{(1)}. \end{aligned}$$

Sei $\mathbf{c} = B^{n+1}\mathbf{P}^{(1)}$. Betrachte die Bézierkurve, die durch die Kontrollpunkte in der obersten Reihe des Dreieckschemas definiert wird,

$$\begin{aligned}
\mathbf{c}_\ell(t) &= E \sum_{j=0}^n b_{j+1}^{n+1}(t) \mathbf{P}^{(j+1)} \\
&= E \sum_{j=0}^n \binom{n}{j} (1-t)^{n-j} t^j ((1-t_0) + t_0 S)^j \mathbf{P}^{(1)} \\
&= E \sum_{j=0}^n \binom{n}{j} (1-t)^{n-j} (t(1-t_0) + t t_0 S)^j \mathbf{P}^{(1)} \\
&= E ((1-t) + (t(1-t_0) + t t_0 S))^n \mathbf{P}^{(1)} \\
&= E ((1-t t_0) + t t_0 S)^n \mathbf{P}^{(1)} = \mathbf{c}(t t_0).
\end{aligned}$$

Die oberste Reihe von Kontrollpunkten des de Casteljau-Dreiecks definiert das dem Intervall $[0, t_0]$ entsprechende Segment der gegebenen Kurve $\mathbf{c} = B^n \mathbf{P}$,

$$\mathbf{c}_\ell(t) := \mathbf{c}(t t_0) = \sum_{j=1}^n b_j^n(t) \mathbf{P}_1^{(j)} = B^n(t) \mathbf{P}_\ell.$$

Aus Symmetriegründen gilt: Die unterste Reihe von Kontrollpunkten des de Casteljau-Dreiecks (von rechts nach links gelesen) definiert das dem Intervall $[t_0, 1]$ entsprechende Segment,

$$\mathbf{c}_r(t) := \mathbf{c}(t_0 + (t(1-t_0))) = \sum_{j=1}^n b_j^n(t) \mathbf{P}_{n+1-j}^{(j)} = B^n(t) \mathbf{P}_r.$$

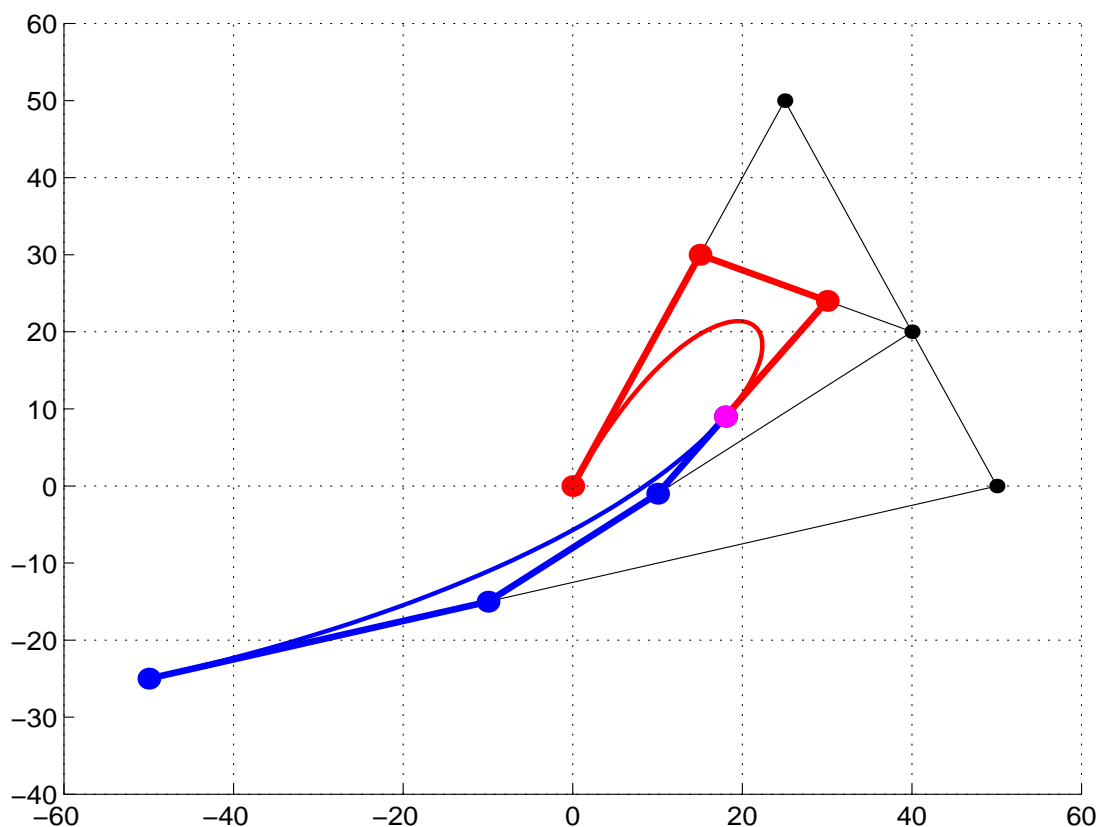
Beispiel: siehe oben

$$\mathbf{p} = \begin{bmatrix} 0 & 25 & 50 & -50 \\ 0 & 50 & 0 & -25 \end{bmatrix}^t, \quad t_0 = 3/5$$

$$\begin{array}{cccc} 0 & & & 0 \\ & 15 & & 30 \\ 25 & & 30 & 50 \\ & 40 & 18 & 20 \\ 50 & & 10 & 0 \\ & -10 & & -15 \\ -50 & & & -25 \end{array}$$

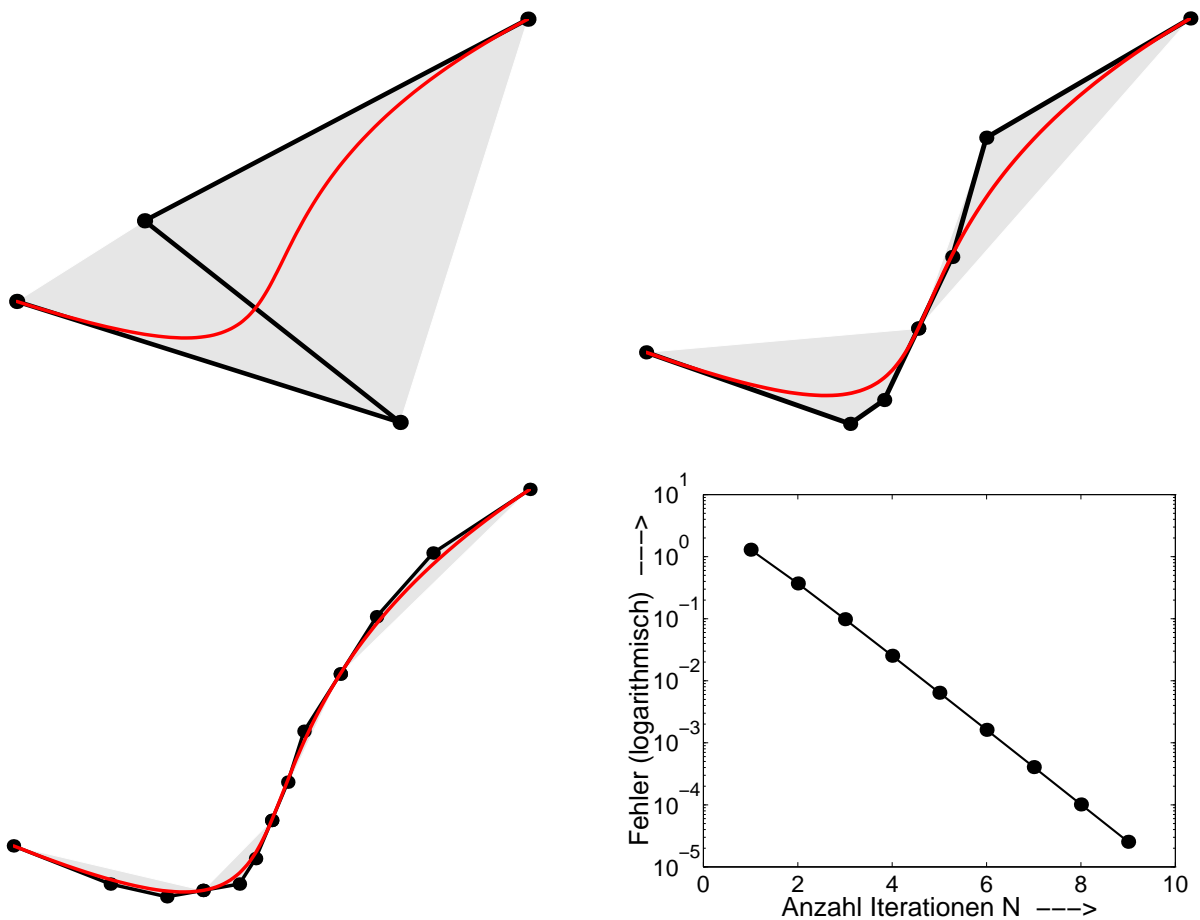
$$\mathbf{c}_\ell(t) = \mathbf{c}(3t/5) = B^4(t)\mathbf{P}_\ell \quad \mathbf{c}_r(t) = \mathbf{c}(3/5+2t/5) = B^4(t)\mathbf{P}_r$$

$$\mathbf{p}_\ell = \begin{bmatrix} 0 & 15 & 30 & 18 \\ 0 & 30 & 24 & 9 \end{bmatrix}^t, \quad \mathbf{p}_r = \begin{bmatrix} 18 & 10 & -10 & -50 \\ 9 & -1 & -15 & -25 \end{bmatrix}^t$$



Iterierte Unterteilung: Bei iterierter Unterteilung konvergiert die Folge der Kontrollpolygone rasch gegen die gegebene Kurve. Die Segmentierung kann dabei adaptiv gesteuert werden. Anwendungen ergeben sich u.A. bei Plotten von Kurven oder bei der Bestimmung von Nullstellen.

Beispiel: Binäre Unterteilung



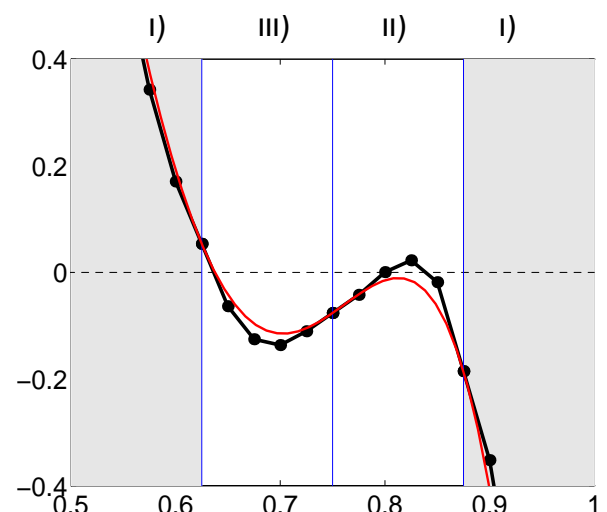
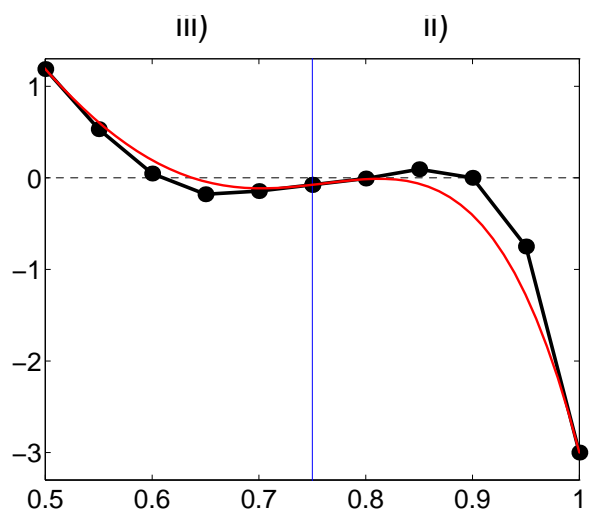
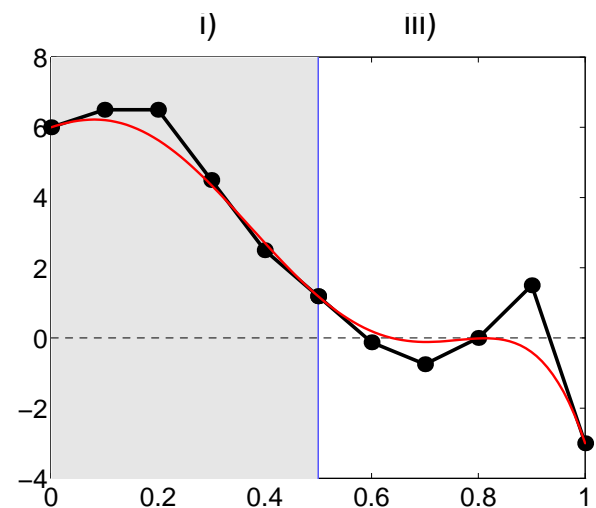
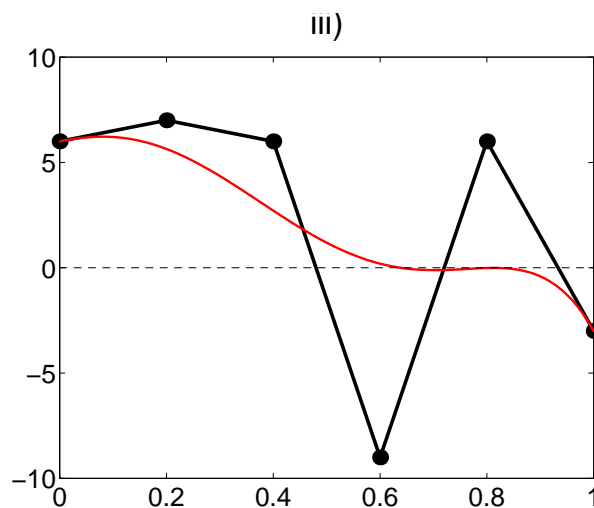
Anwendung: Nullstellensuche im Einheitsintervall für ein Polynom in Bernstein-Bézierform.

Es gilt

- i) $\min P > 0$ oder $\max P < 0 \Rightarrow$ keine Nullstelle.
- ii) $\min P < 0$ und $\max P > 0 \Rightarrow$ Nullstelle möglich.
- iii) $p_1 p_n \leq 0 \Rightarrow$ Nullstelle sicher.

Man iteriert den binären Unterteilungsalgorithmus für alle Teilintervalle, für die nicht i) gilt, bis eine vorgegebene Toleranz erreicht ist.

Beispiel: $P = [6, 7, 6, -9, 6, -3]^t$



2.8 Abstand Bézierkurve – Kontrollpolygon

Sei $c = B^n \mathbf{P}$ und $T^n := 0 : 1/(n-1) : 1$, dann parametrisiert die Kurve $\mathbf{p} : [0, 1] \rightarrow \mathbb{R}^d$ mit

$$\mathbf{p}(t) = \frac{(t - t_j^n) \mathbf{p}_{j+1} + (t_{j+1}^n - t) \mathbf{p}_j}{t_{j+1}^n - t_j^n}, \quad t \in [t_j^n, t_{j+1}^n]$$

das Kontrollpolygon.

Theorem: Abstand zum Kontrollpolygon

Sei $d = 1$. Für den Abstand zwischen $c = B^n P$ und p gilt

$$\|c - p\|_\infty \leq \Gamma(n) \|\Delta^2 P\|_\infty,$$

wobei

$$\Gamma(n) = \begin{cases} \frac{n-1}{8} & \text{falls } n \text{ ungerade} \\ \left(\frac{n-1}{8} - \frac{1}{8(n-1)} \right) & \text{falls } n \text{ gerade.} \end{cases}$$

Diese Abschätzung ist für $c \in \mathbb{P}_3$ scharf.

Beweis: Sei o.B.d.A. $\|\Delta^2 P\|_\infty = 8$ und

$$\|c - p\|_\infty = c(t^*) - p(t^*)$$

maximal. Für $t^* \in [t_j^n, t_{j+1}^n]$ betrachte p_j und p_{j+1} als fest.

Dann ist $c(t^*) - p(t^*)$ genau dann maximal, wenn alle $p_k, k \notin \{j, j+1\}$ so groß wie möglich sind. Dies ist genau dann der Fall, wenn

$$p_{k+1} - 2p_k + p_{k-1} = 8, \quad k = 2 : (n-1).$$

Dies bedeutet, dass $p_k = 4k^2 + \alpha k + \beta$ für gewisse Konstanten α, β . Folglich ist $c \in \mathbb{P}_3$ und $t^* \in T^n$.

Betrachte also

$$c(t_j^n) - p_j = 4((t_j^n(1 - t_j^n))(n-1)).$$

Dieser Ausdruck nimmt sein Maximum an, wenn t_j^n so nahe wie möglich bei $1/2$ liegt, also für

$$t_j^n = \begin{cases} 1/2 & \text{falls } n \text{ ungerade} \\ 1/2 - 1/2(n-1) & \text{falls } n \text{ gerade.} \end{cases}$$

Bei Graderhöhung $B^n(t)P = B^m(t)\tilde{P}, m > n$, gilt

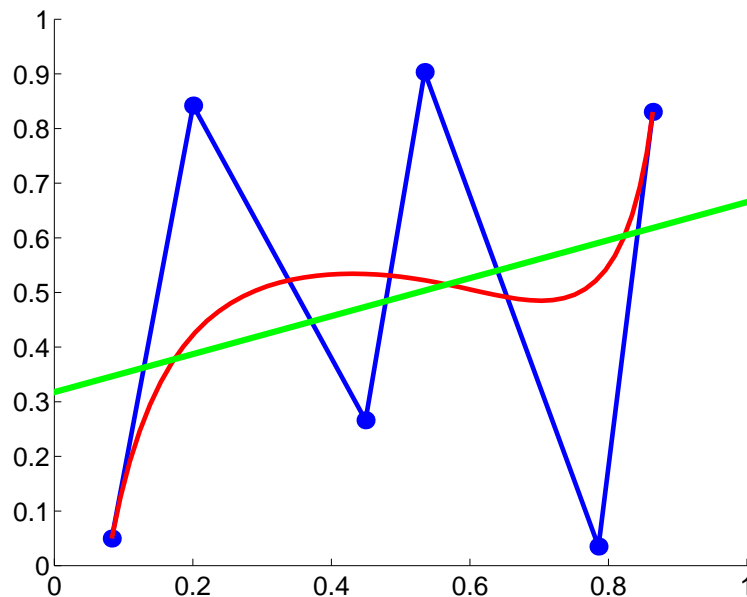
$$\|c - \tilde{p}\|_\infty \leq \frac{(n-1)(n-2)}{8(m-2)} \|\Delta^2 P\|_\infty.$$

Bei Subdivision $B^n(a+th)P = B^n(t)\tilde{P}, [a, a+h] \subset [0, 1]$, gilt

$$\|\tilde{c} - \tilde{p}\|_\infty \leq \frac{h^2(n-1)}{8} \|\Delta^2 P\|_\infty.$$

2.9 Variationsminderung

Die Anzahl der Schnitte einer ebenen Kurve mit einer gegebenen Geraden kann als Maß für deren „Welligkeit“ angesehen werden. In diesem Sinne ist eine Bézierkurve nie welliger als das zugehörige Kontrollpolygon.



Theorem: Variationsminderung

Bézierkurven sind variationsmindernd. Das heißt, für eine Bézierkurve c mit zugehörigem Kontrollpolygon p und eine beliebige Hyperebene $h : \mathbf{x}a^t - d = 0$ gilt

$$\#VZW(\mathbf{c}a^t - d) \leq \#VZW(\mathbf{p}a^t - d).$$

Beweis: i) Lineare Interpolation ist variationsmindernd. ii) Graderhöhung kann als fortgesetzte lineare Interpolation des Kontrollpolygons angesehen werden, ist also variationsmindernd. iii) Bei fortgesetzter Graderhöhung konvergiert die Folge der Kontrollpolygone gegen die Bézierkurve. Das Resultat folgt dann aus Stetigkeitsgründen.

2.10 Rationale Bézierkurven

Motivation: Polynomiale Bézierkurven haben den Nachteil, dass sich Kegelschnitte (insbesondere Kreise) nicht exakt modellieren lassen.

Idee: Verwende rationale Basisfunktionen.

Ansatz: Die Gewichte $W = [w_1, \dots, w_n]^t$ definieren die Gewichtsfunktion $w(t)$ gemäß

$$w(t) = B^n(t)W.$$

Damit bilden für $\prod_j w_j \neq 0$ die rationalen Basisfunktionen

$$b_{W,j}^n(t) := w_j b_j^n(t)/w(t), \quad B_W^n(t) := [b_{W,1}^n(t), \dots, b_{W,n}^n(t)]$$

eine Partition der Eins, $\sum_j b_{W,j}^n(t) \equiv 1$.

Definition: Rationale Bézierkurve

Sei $W = [w_1, \dots, w_n]$ und $\prod_j w_j \neq 0$, dann heißt

$$\mathbf{c}(t) = B_W^n(t)\mathbf{P} = \frac{\sum_{j=1}^n b_j^n(t)w_j\mathbf{p}_j}{\sum_{j=1}^n b_j^n(t)w_j}$$

rationale Bézierkurve der Ordnung n .

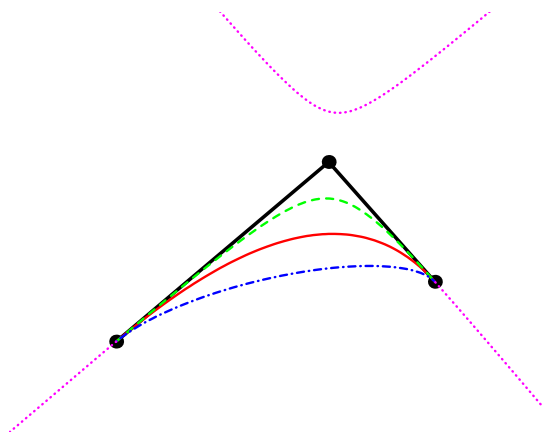
Es folgt unmittelbar für $\mathbf{c}(t) = B_W^n(t)\mathbf{P}$ und $\prod_j w_j \neq 0$

- **affine Invarianz** $A\mathbf{c}(t) + \mathbf{a} = B_W^n(t)(A\mathbf{P} + \mathbf{a})$.
- **konvexe Hülle** $\mathbf{c}(t) \in \text{conv}(\mathbf{P})$, falls $W > 0$.
- **Randinterpolation** $\mathbf{c}(0) = \mathbf{p}_1$.
- **Randableitung** $D\mathbf{c}(0) = (n-1)(\mathbf{p}_2 - \mathbf{p}_1)w_2/w_1$.

Beispiele:

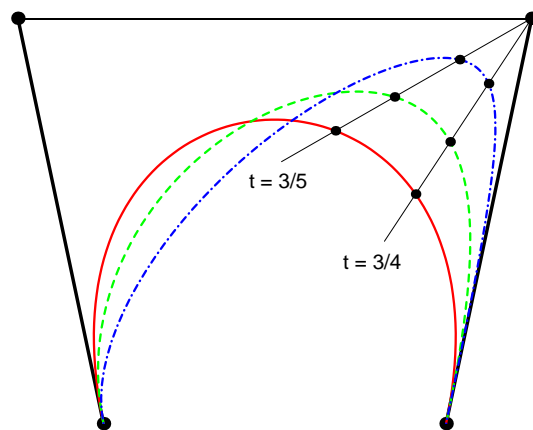
$$n = 3, W = [1, \alpha, 1]^t$$

$$\alpha = 1, 3, 1/3, -4$$



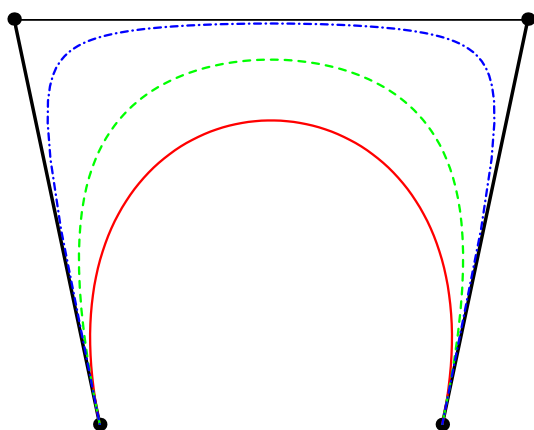
$$n = 4, W = [1, 1, \alpha, 1]^t$$

$$\alpha = 1, 2, 5,$$

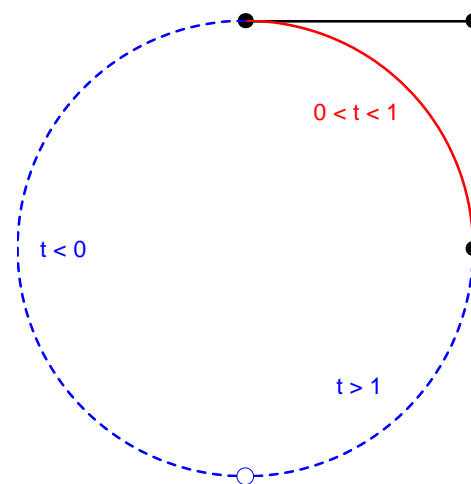


$$n = 4, W = [1, \alpha, \alpha, 1]^t$$

$$\alpha = 1, 3, 30,$$



$$n = 3, W = [1, 1, 2]^t$$



Homogene Koordinaten: Identifiziere euklidische Koordinaten $[x^1, \dots, x^d]$ in \mathbb{R}^d mit homogenen Koordinaten in \mathbb{R}^{d+1} gemäß

$$[x^1, \dots, x^d] \sim [\alpha x^1, \dots, \alpha x^d, \alpha], \quad \alpha \in \mathbb{R} \setminus 0 \text{ beliebig,}$$

bzw.

$$\mathbf{x} \sim [\alpha \mathbf{x}, \alpha] \sim [\mathbf{x}, 1].$$

Beispielsweise gilt in \mathbb{R}^3

$$[x/w, y/w, z/w] \sim [x, y, z, w].$$

Für eine rationale Bézierkurve gilt:

$$\begin{aligned} \mathbf{c}(t) &\sim \left[\sum_{j=1}^n b_j^n(t) w_j \mathbf{p}_j, \sum_{j=1}^n b_j^n(t) w_j \right] \\ &\sim \sum_{j=1}^n b_j^n(t) [w_j \mathbf{p}_j, w_j] = B^n(t) \mathbf{P}_W. \end{aligned}$$

Die homogenen Koordinaten einer rationalen Bézierkurve erhält man durch die Auswertung der Bézierkurve mit Kontrollpunkten $\mathbf{p}_{W,j} = [w_j \mathbf{p}_j, w_j]$.

Anwendung: Viele Algorithmen für integrale Bézierkurven (Auswertung, Graderhöhung, Gradreduktion, Unterteilung etc.) lassen sich unmittelbar auf rationale Bézierkurven übertragen. *Ausnahme:* Integration, Differenziation.

Beispiel: siehe Seite 39, rechts unten

$$\mathbf{P} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}^t, \quad W = [1 \ 1 \ 2]^t.$$

Matrix der homogenen Kontrollpunkte:

$$\mathbf{P}_W = \begin{bmatrix} 0 & 1 & 2 \\ 1 & 1 & 0 \\ 1 & 1 & 2 \end{bmatrix}^t$$

Graderhöhung:

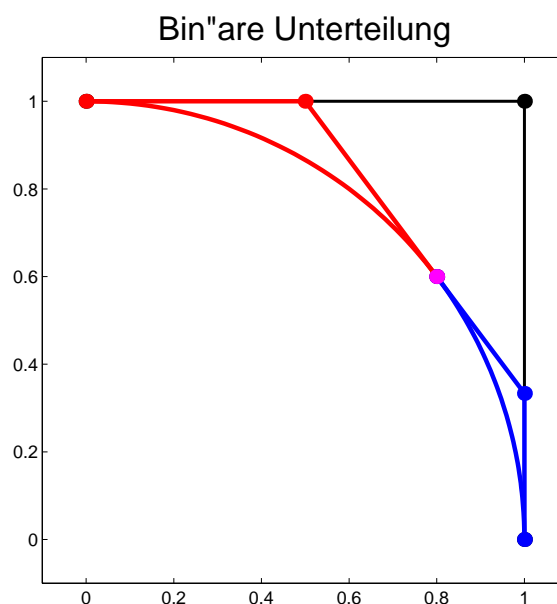
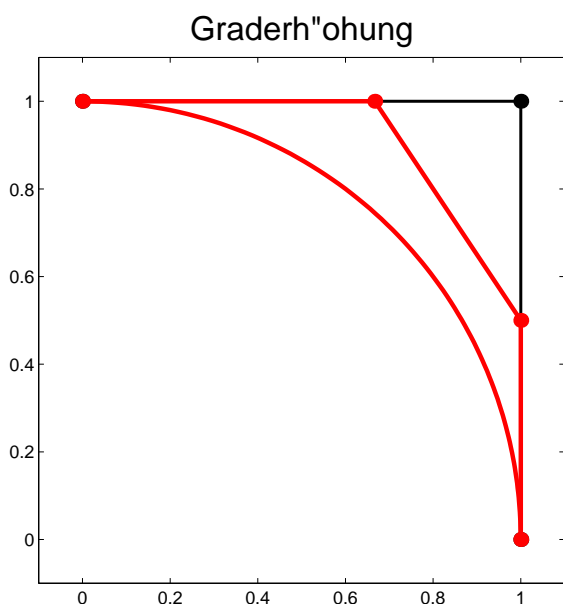
$$\tilde{\mathbf{P}}_W = \frac{1}{3} \begin{bmatrix} 0 & 2 & 4 & 6 \\ 3 & 3 & 2 & 0 \\ 3 & 3 & 4 & 6 \end{bmatrix}^t, \quad \tilde{\mathbf{P}} = \frac{1}{6} \begin{bmatrix} 0 & 4 & 6 & 6 \\ 6 & 6 & 3 & 0 \end{bmatrix}^t$$

$$\tilde{W} = \frac{1}{3} [3 \ 3 \ 4 \ 6]^t$$

Binäre Unterteilung:

$$\mathbf{P}_{W,\ell} = \frac{1}{4} \begin{bmatrix} 0 & 2 & 4 \\ 4 & 4 & 3 \\ 4 & 4 & 5 \end{bmatrix}^t, \quad \mathbf{P}_\ell = \frac{1}{10} \begin{bmatrix} 0 & 5 & 8 \\ 10 & 10 & 6 \end{bmatrix}^t$$

$$\tilde{W}_\ell = \frac{1}{4} [4 \ 4 \ 5]^t$$



Kegelschnitte: Die Darstellung des Kegelschnitts

$$ax^2 + 2bxy + cy^2 + 2dx + 2ey + f = 0$$

in homogener Form lautet

$$[x \ y \ w] \begin{bmatrix} a & b & d \\ b & c & e \\ d & e & f \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix} =: [x, y, w] Q [x, y, w]^t = 0.$$

Die Tangente im Punkt $[x_0, y_0, w_0]$ ist gegeben durch

$$[x, y, w] Q [x_0, y_0, w_0]^t = 0.$$

Für $w = 1$ erhält man jeweils die euklidische Form zurück.

Theorem: Jede quadratische rationale Bézierkurve in \mathbb{R}^2 ist Teilmenge eines Kegelschnitts.

Beweis: Für beliebige quadratische Polynome $x(t)$, $y(t)$, $w(t)$ ist $[x(t), y(t), w(t)] Q [x(t), y(t), w(t)]^t$ ein Polynom der Ordnung 5, dessen Koeffizienten linear von a, \dots, f abhängen. Nullsetzen ergibt 5 homogene lineare Gleichungen für 6 Variable, es gibt also nichttriviale Lösungen.

Beispiel: wie zuvor

$$x(t) = 2t, \quad y(t) = 1 - t^2, \quad w(t) = 1 + t^2$$

$$\begin{bmatrix} 0 & 0 & 1 & 0 & -2 & 1 \\ 0 & -4 & 0 & 4 & 0 & 0 \\ 4 & 0 & -2 & 0 & 0 & 2 \\ 0 & 4 & 0 & 4 & 0 & 0 \\ 0 & 0 & 1 & 0 & 2 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \end{bmatrix} = 0 \Rightarrow Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix}.$$

Die Kurve ist also Teilmenge des Einheitskreises.

Erweitert man den Definitionsbereich der quadratischen Bézierkurve auf $t \in \mathbb{R} \cup \infty$, so wird der gesamte Kegelschnitt durchlaufen. Dabei definiert man den *Pol* als

$$\mathbf{c}(\infty) := \lim_{t \rightarrow \infty} \mathbf{c}(t) = \frac{w_1 \mathbf{p}_1 - 2w_2 \mathbf{p}_2 + w_3 \mathbf{p}_3}{w_1 - 2w_2 + w_3}.$$

Theorem: Jeder nichtentartete Kegelschnitt (Ellipse, Hyperbel, Parabel) läßt sich als erweiterte quadratische Bézierkurve darstellen.

Beweis: Transformiere den Kegelschnitt auf eine der drei Normalformen, $x^2 + y^2 = 1$, $xy = 1$, $y = x^2$. Dafür lassen sich elementar Bézierkurven angeben. Die Behauptung folgt dann aus der affinen Invarianz der Bézierdarstellung.

Die Parametrisierung eines Kegelschnitts durch eine quadratische Bézierkurve ist nicht eindeutig. Es können beispielsweise zusätzlich die Orte zu den Parameterwerten $t = 0, 1, \infty$ spezifiziert werden.

Konstruktion:

1. Setze $\mathbf{p}_1 := \mathbf{c}(0)$, $\mathbf{p}_3 := \mathbf{c}(1)$.
2. Bestimme \mathbf{p}_2 als Schnittpunkt der Tangenten in $\mathbf{c}(0)$, $\mathbf{c}(1)$ (falls existent),

$$\begin{bmatrix} \mathbf{p}_1, 1 \\ \mathbf{p}_3, 1 \end{bmatrix} Q \begin{bmatrix} \mathbf{p}_2^t \\ 1 \end{bmatrix} = 0.$$

3. Bestimme die Gewichte W als nichttriviale Lösung von

$$\left[\mathbf{c}(\infty) - \mathbf{p}_1; -2(\mathbf{c}(\infty) - \mathbf{p}_2); \mathbf{c}(\infty) - \mathbf{p}_3 \right]^t W = 0.$$

Akzeptiere die Lösung, falls $w_1 w_2 w_3 \neq 0$.

Beispiel: Hyperbel $xy = 4$,

$$\mathbf{c}(0) = [2, 2], \quad \mathbf{c}(1) = [-4, -1], \quad \mathbf{c}(\infty) = [1, 4].$$

Homogene Form des Kegelschnitts:

$$[x \ y \ w] \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -8 \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix} = 0.$$

1. Schritt: $\mathbf{p}_1 := \mathbf{c}(0) = [2, 2]$, $\mathbf{p}_3 := \mathbf{c}(1) = [-4, -1]$.

2. Schritt: $\mathbf{p}_2 := [8, -4]$ ist Lösung von

$$\begin{bmatrix} 2 & 2 & 1 \\ -4 & -1 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -8 \end{bmatrix} \begin{bmatrix} \mathbf{p}_2^t \\ 1 \end{bmatrix} = \begin{bmatrix} 2 & 2 & -8 \\ -1 & -4 & -8 \end{bmatrix} \begin{bmatrix} \mathbf{p}_2^t \\ 1 \end{bmatrix} = 0.$$

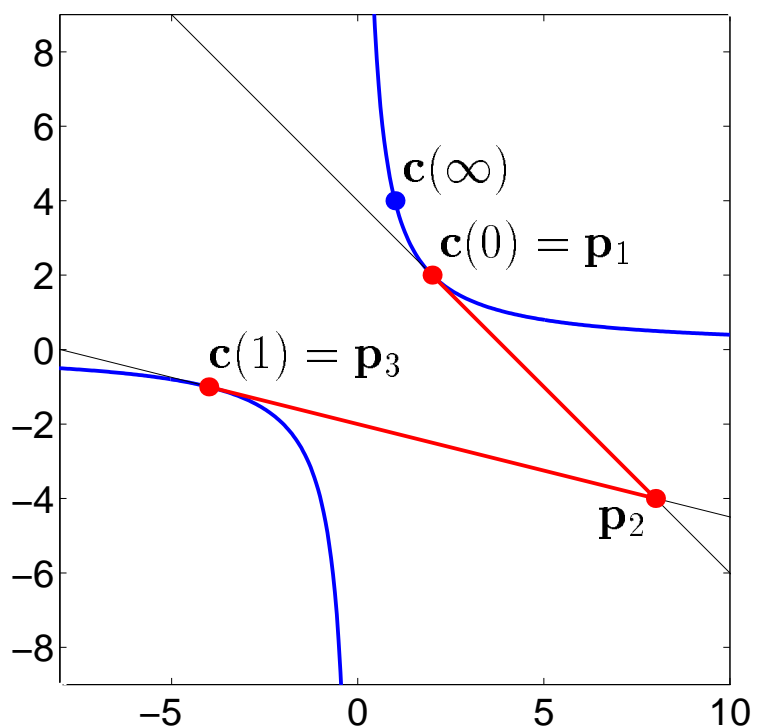
3. Schritt: $W := [50, 5, -4]^t$ ist Lösung von

$$\begin{bmatrix} -1 & 14 & 5 \\ 2 & -16 & 5 \end{bmatrix} W = 0.$$

Lösung:

$$\mathbf{P} = \begin{bmatrix} 2 & 8 & -4 \\ 2 & -4 & -1 \end{bmatrix}^t$$

$$W = [50 \quad 5 \quad -4]^t$$

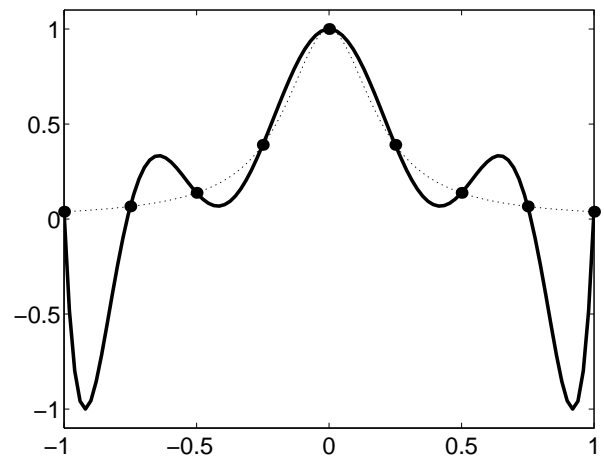


3. Splines

Nachteile polynomialer Funktionen und Kurven:

- Globale Abhängigkeit der Funktion/Kurve von den Koeffizienten.
- Interpolation von Funktionenwerten liefert u.U. schlechte Approximation.

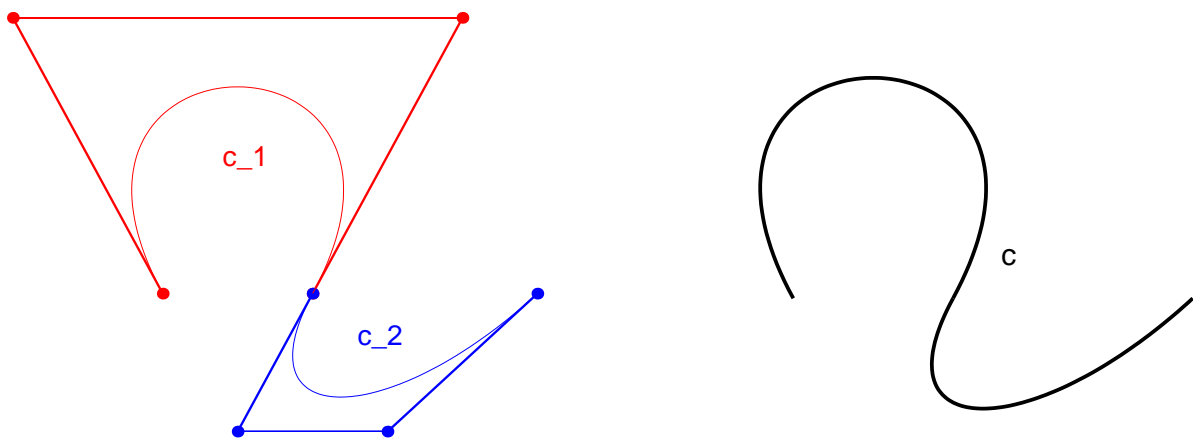
Beispiel: Interpolation von $f(x) = 1/(1+25x^2)$ an 9 äquidistanten Stützstellen.



- Erzwungene Glattheit.

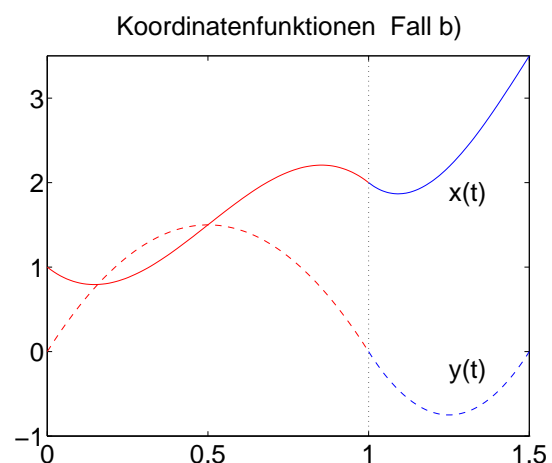
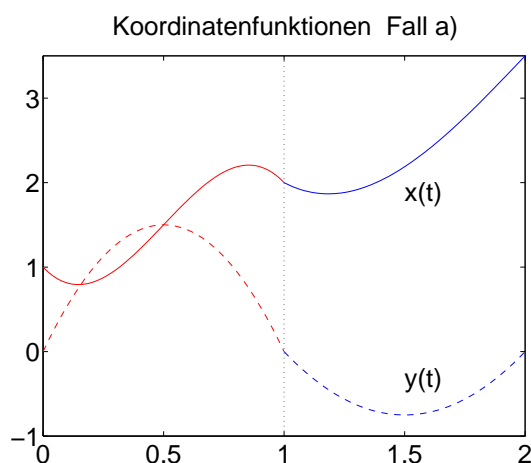
Idee: Verwende stückweise polynomiale Funktionen.

Beispiel: Zusammenfügen zweier Bézierkurven.



Es gibt viele Möglichkeiten, die zusammengesetzte Kurve zu parametrisieren, z.B.:

$$\begin{aligned} \text{a)} \quad \mathbf{c}(t) &= \begin{cases} \mathbf{c}_1(t) & \text{für } 0 \leq t \leq 1 \\ \mathbf{c}_2(t - 1) & \text{für } 1 \leq t \leq 2 \end{cases} \\ \text{b)} \quad \mathbf{c}(t) &= \begin{cases} \mathbf{c}_1(t) & \text{für } 0 \leq t \leq 1 \\ \mathbf{c}_2(2t - 2) & \text{für } 1 \leq t \leq 3/2 \end{cases} \end{aligned}$$



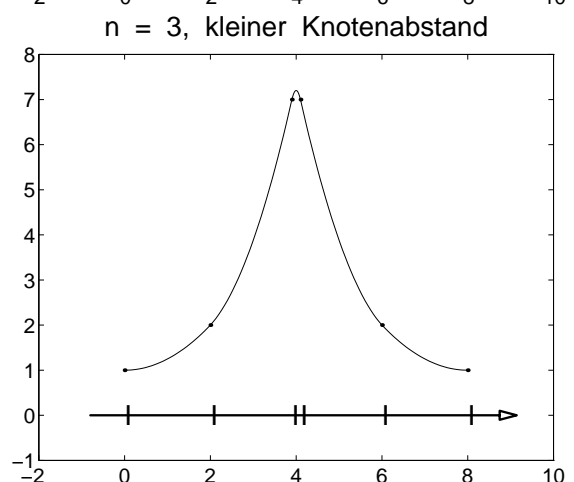
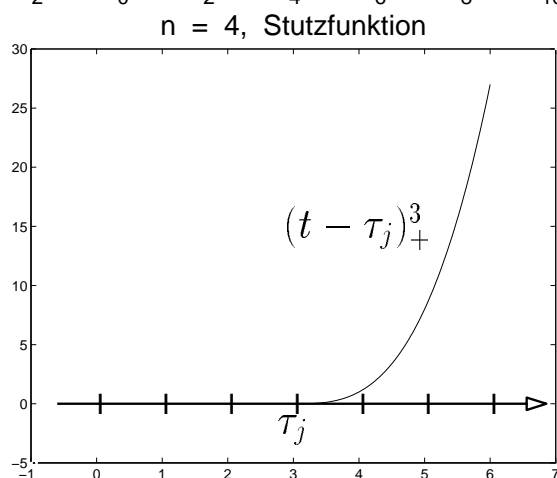
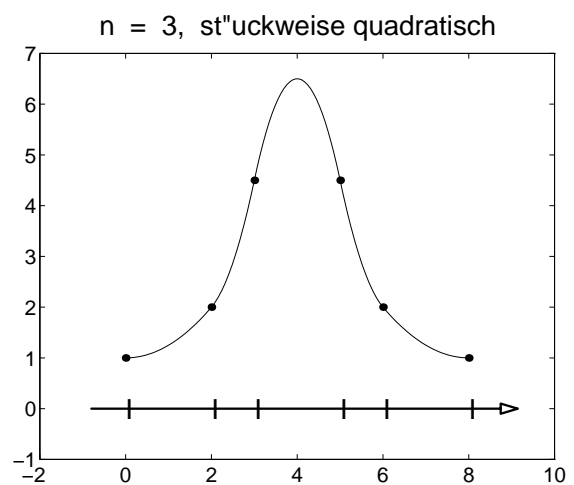
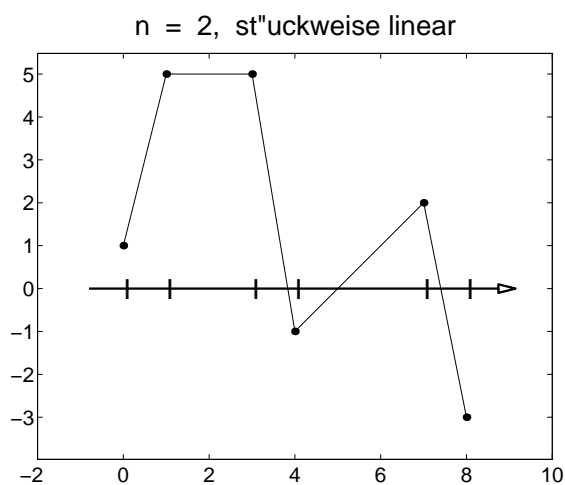
Die Koordinatenfunktionen sind stückweise polynomial und im Fall b) stetig differenzierbar.

Ein *Spline* der Ordnung n ist eine Funktion, die stückweise aus Polynomen der Ordnung n besteht. Die Abszissen τ_j , die die Polynomsegmente trennen, heißen *Knoten*. Die Knoten werden zu der *Knotenfolge* $T = \{\tau_j\}$ zusammengefasst.

Welche Glattheit kann man an den Knoten erwarten? Wenn zwei Polynomsegmente der Ordnung n $(n - 1)$ -mal stetig differenzierbar verbunden sind, dann stellen sie ein und dasselbe Polynom dar.

Man vereinbart: An einfachen Knoten ist ein Spline mindestens $(n - 2)$ -mal stetig differenzierbar.

Beispiele:



Frage: Betrachte nun Splines der Ordnung $n + 1$. Für jeden Knoten τ_j ist die *abgebrochene Potenz*

$$(t - \tau_j)_+^n, \quad z_+ := \max\{0, z\},$$

ein Element des Splineraums. Betrachte o.B.d.A. die Knoten $\tau_j := 0$ und $\tau_{j+1} := h$, dann sind

$$t_+^n \quad \text{und} \quad (t - h)_+^n$$

Splines und folglich auch die Linearkombination

$$f_h(t) = \frac{t_+^n - (t - h)_+^n}{nh} = \begin{cases} 0 & , t \leq 0 \\ t^n / (nh) & , t \in (0, h) \\ (t^n - (t - h)^n) / (nh), & t \geq h. \end{cases}$$

Im Grenzübergang gilt $\tau_j = \tau_{j+1}$ und

$$\lim_{h \rightarrow 0} f_h(t) = t_+^{n-1},$$

die Differenzierbarkeitsordnung ist also nur noch $n - 2$.

Man vereinbart: An einem k -fachen Knoten, ist ein Spline der Ordnung $n + 1$ $(n - k)$ -mal stetig differenzierbar.

Stetigkeitsargument: Sei $\tau_j = \dots = \tau_{j+k-1} = 0$ ein k -facher Knoten und $\tau_{j+k} = h$, dann sind per Induktionsannahme die abgebrochenen Potenzen

$$t_+^n, t_+^{n-1}, \dots, t_+^{n-k+1} \quad \text{und} \quad (t - h)_+^n$$

Elemente des Splineraums. Betrachte die Splinefunktion

$$f_h(t) := \frac{(t - h)_+^n - \sum_{\mu=0}^{k-1} \binom{n}{\mu} (-h)^\mu t_+^{n-\mu}}{\binom{n}{k} (-h)^k},$$

dann folgt für $k \leq n$ mit Hilfe der binomischen Formel

$$\lim_{h \rightarrow 0} f_h(t) = t_+^{n-k}.$$

Die *Knotenfolge* T eines Splineraums der Ordnung n ist eine endliche, monoton wachsende Folge reeller Zahlen der Form

$$T := \tau_1, \dots, \tau_{m+n} .$$

Die Knotenfolge heißt *nicht entartet*, wenn

$$\tau_n < \tau_{n+1}, \quad \tau_m < \tau_{m+1}, \quad \tau_j < \tau_{j+n} \quad \forall j, \quad m \geq n .$$

Ein *Spline* der Ordnung n über der Knotenfolge T ist eine Funktion $f : \mathbb{R} \supset D \mapsto \mathbb{R}$, die auf den Intervallen $[\tau_j, \tau_{j+1})$ mit Polynomen der Ordnung n übereinstimmt und an k -fachen Knoten $(n - k - 1)$ -mal stetig differenzierbar ist.

Der Raum aller Splines der Ordnung n über einer Knotenfolge T wird mit $\mathcal{S}_{n,T}(D)$ bezeichnet.

Konventionen: Außerhalb des Intervalls $[\tau_1, \tau_{m+n}]$ sind alle Splines identisch Null.

Meist wählt man den Definitionsbereich $D = D(T) = [\tau_n, \tau_{m+1})$.

Die Anzahl der identischen Nachfolger eines Knotens ist

$$\#j := \max\{k : \tau_{j+k} = \tau_j\} .$$

Für die k -fache Wiederholung eines Knotens schreibt man

$$\tau_j \text{\$} k := \underbrace{\tau_j, \tau_j, \dots, \tau_j}_{k\text{-mal}} .$$

Beispiel: $T = -1, 0, 1, 2, 3, 4$.

T	-1	0	0	1	1	1	2	3	4
$n = 1, m = 8$	τ_1	τ_2	τ_3	τ_4	τ_5	τ_6	τ_7	τ_8	τ_9
$n = 2, m = 7$	τ_1	τ_2	τ_3	τ_4	τ_5	τ_6	τ_7	τ_8	τ_9
$n = 3, m = 6$	τ_1	τ_2	τ_3	τ_4	τ_5	τ_6	τ_7	τ_8	τ_9
$n = 4, m = 5$	τ_1	τ_2	τ_3	τ_4	τ_5	τ_6	τ_7	τ_8	τ_9

T ist entartet für $n = 1, 2, 4$ und nicht entartet für $n = 3$.

Theorem: Der Raum $\mathcal{S}_{n,T}$ ist linear. Für eine nicht-entartete Knotenfolge T ist eine Basis des Raums $\mathcal{S}_{n,T}(D(T))$ gegeben durch

$$a_j^n(t) := (t - \tau_j)_+^{n-1-\#j}, \quad j = 1 : m.$$

Insbesondere ist dann $\dim \mathcal{S}_{n,T}(D(T)) = m$.

Beweis: Die Linearität ist trivial. Zunächst wird die Dimensionsformel per vollständiger Induktion bewiesen:

Induktionsanfang: Für $m = n$ ist $\dim \mathcal{S}_{n,T}(D(T)) = n$.

Induktionsannahme: Die Behauptung sei richtig für m .

Induktionsschluss: Füge in eine Knotenfolge T der Länge $m + n$ einen weiteren Knoten $\tau^* \in (\tau_n, \tau_{m+1})$ ein. Ist k die Vielfachheit des neuen Knotens τ^* , dann gilt

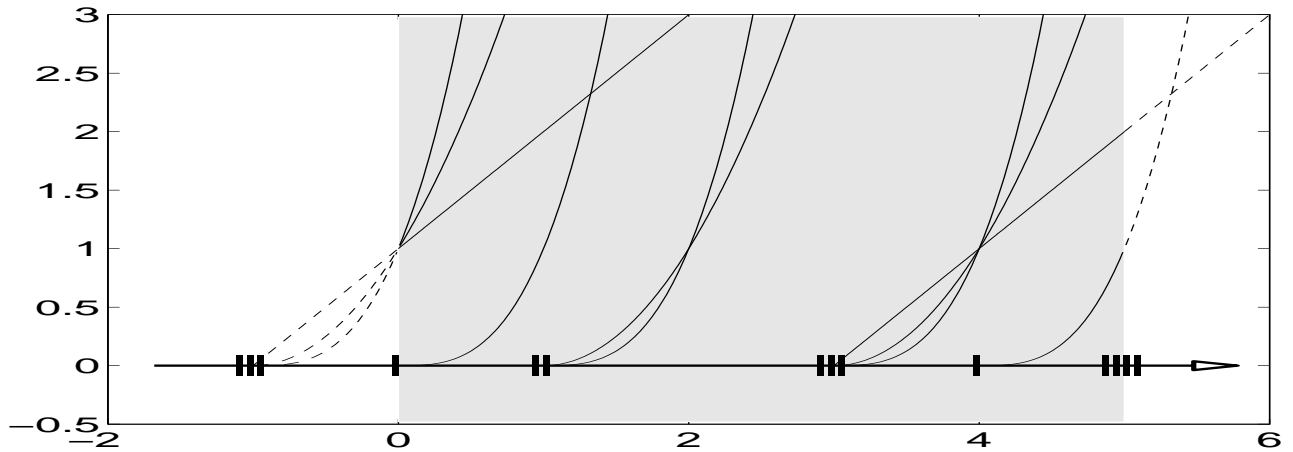
$$\mathcal{S}_{n,T^*}(D(T)) = \mathcal{S}_{n,T}(D(T)) \oplus \text{span}(t - \tau^*)_+^{n-k},$$

also ist $\dim \mathcal{S}_{n,T^*}(D(T)) = m + 1$.

Die m Funktionen $a_j^n(t)$ sind linear unabhängig und liegen in $\mathcal{S}_{n,T}(D(T))$. Folglich bilden sie eine Basis.

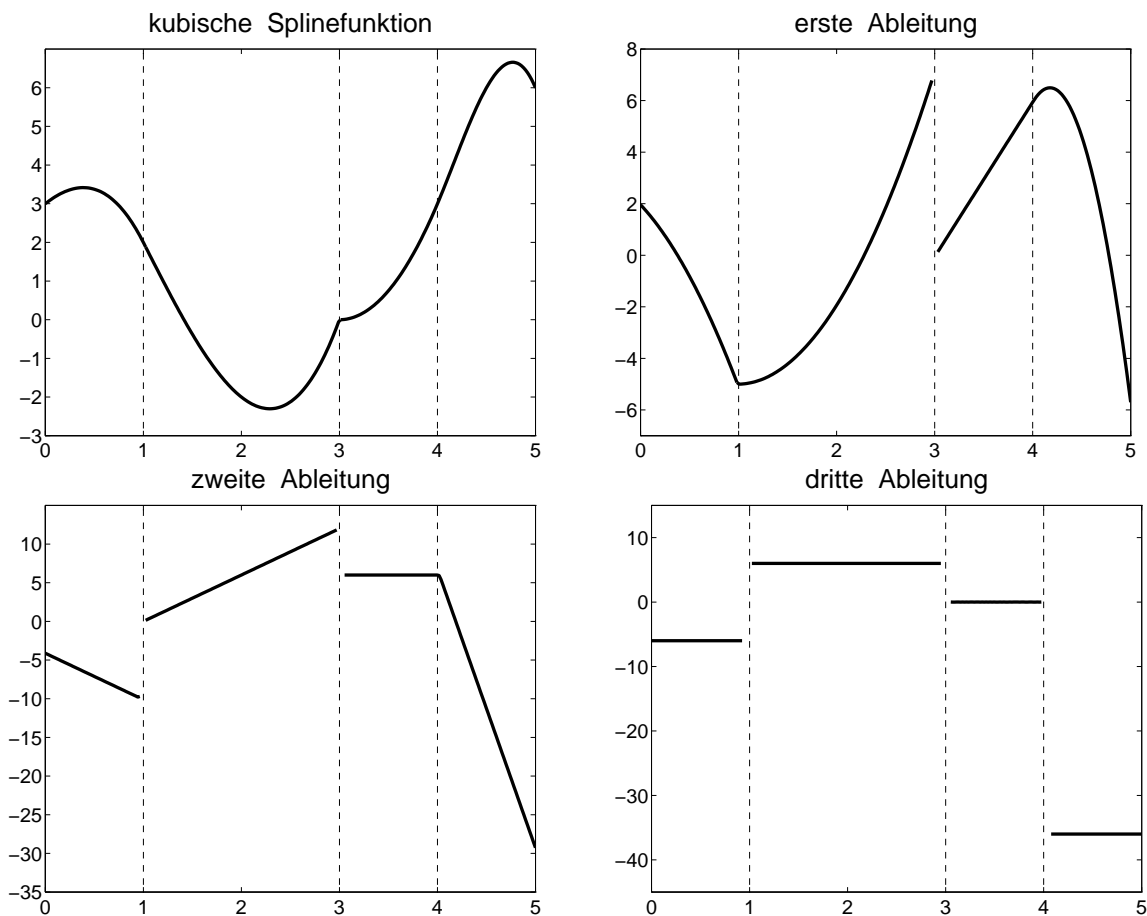
Beispiel: Basis $\{a_j^n(t)\}_{j=1}^m$ aus abgebrochenen Potenzen für $n = 4$ und die Knotenfolge

$$T = -1, -1, -1, 0, 1, 1, 3, 3, 3, 4, 5, 5, 5, 5.$$



Splinefunktion $f(t) = \sum_j q_j a_j^n(t)$ mit Koeffizienten

$$Q = [3, 1, -1, 0, 5, 2, -7, -3, -1, -6].$$



Nachteile der Basis aus abgebrochenen Potenzen:

- keine Symmetrie
- globaler Charakter
- keine erkennbare Relation zwischen Koeffizienten und Geometrie
- mangelnde Stabilität (Koeffizienten klein, Funktion groß)

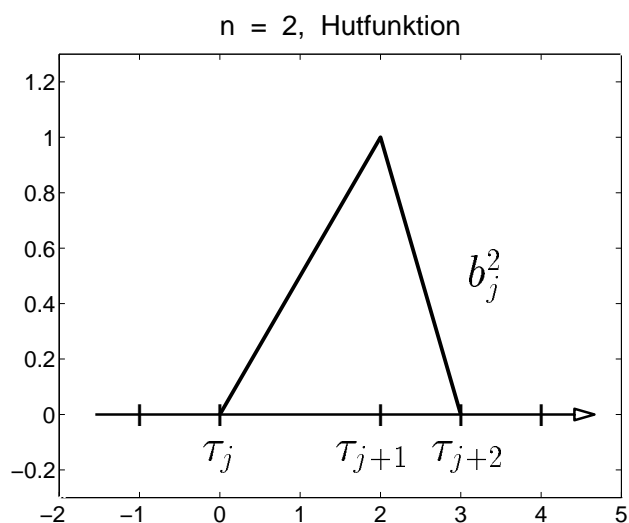
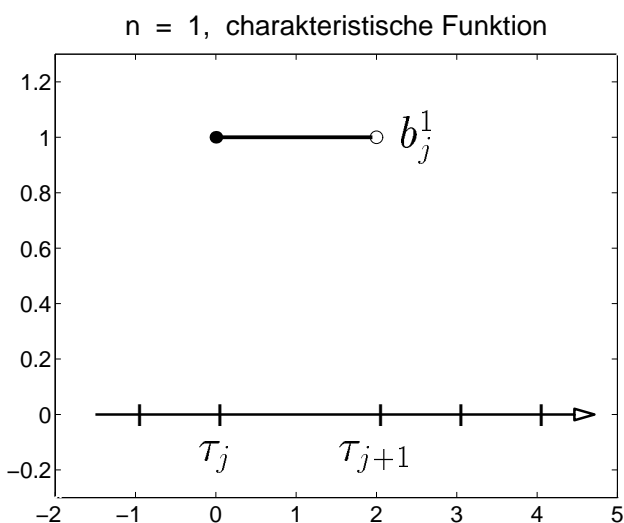
Konstruktion einer Basis mit günstigeren Eigenschaften:

$n = 1$, wähle charakteristische Funktionen der Segmente:

$$b_j^1(t) = \chi([\tau_j, \tau_{j+1})) = \begin{cases} 1 & \text{für } \tau_j \leq t < \tau_{j+1} \\ 0 & \text{sonst .} \end{cases}$$

$n = 2$, wähle Hutfunktionen:

$$b_j^2(t) = \begin{cases} \frac{t - \tau_j}{\tau_{j+1} - \tau_j} & \text{für } \tau_j \leq t < \tau_{j+1} \\ \frac{\tau_{j+2} - t}{\tau_{j+2} - \tau_{j+1}} & \text{für } \tau_{j+1} \leq t < \tau_{j+2} \\ 0 & \text{sonst .} \end{cases}$$



Sei T eine einfache Knotenfolge. Dann gibt es bis auf Skalierung eindeutig bestimmte Splines

$$b_j^n(t) = \sum_{k=j}^{j+n} q_{j,k} a_k^n(t)$$

der Ordnung n über T mit kompaktem nichtleeren Träger. Für diesen gilt

$$\text{supp } b_j^n = s_j^n := [\tau_j, \tau_{j+n}] .$$

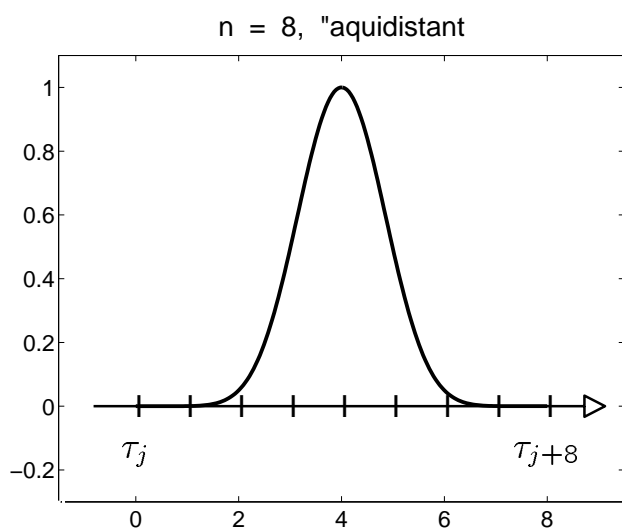
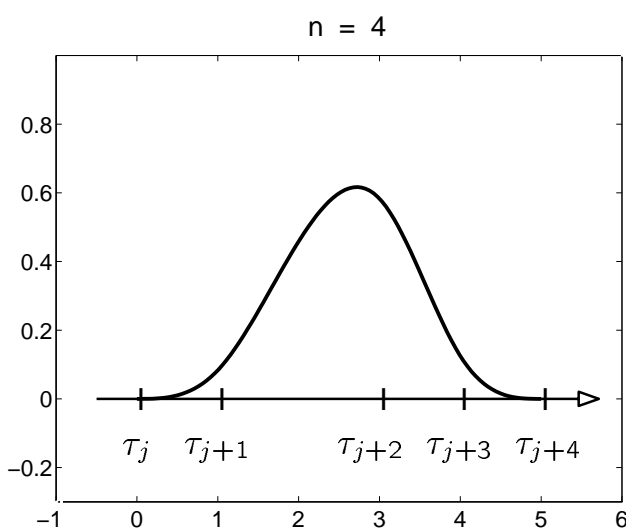
Beweis: Für festes $q_{j,j+r} \neq 0$ und $t \geq \tau_{j+r}$ ist das Gleichungssystem

$$\sum_{k=j}^{j+r-1} q_{j,k} a_k^n(t) = -q_{j,j+r} a_{j+r}^n(t) , \quad t > \tau_{j+r}$$

aufgrund der Basiseigenschaft der Funktionen

$$(t - \tau_j)^{n-1}, \dots, (t - \tau_{j+n-1})^{n-1}$$

für $r = n$ eindeutig lösbar und für $r < n$ unlösbar.



Theorem: Lokale Basis

Sei T einfach und $\{b_j^n, j = 1 : m\}$ eine Folge von Splines wie oben definiert, dann gilt

$$\sum_j b_j^n(t) p_j \equiv 0 \quad \text{für } t \in [\tau_k, \tau_{k+1})$$
$$\Leftrightarrow p_j = 0 \quad \text{für } j = k - n + 1 : k .$$

Beweis: Sei $j^* := \min\{j : k - n < j \leq k \wedge p_j \neq 0\}$.
Der Spline

$$f(t) := \begin{cases} \sum_{j=j^*}^k p_j b_j^n(t) & \text{für } t < \tau_k \\ 0 & \text{für } t \geq \tau_k \end{cases}$$

hat nichtleeren Träger in $[\tau_{j^*}, \tau_k]$. Widerspruch.

Theorem: Globale Basis

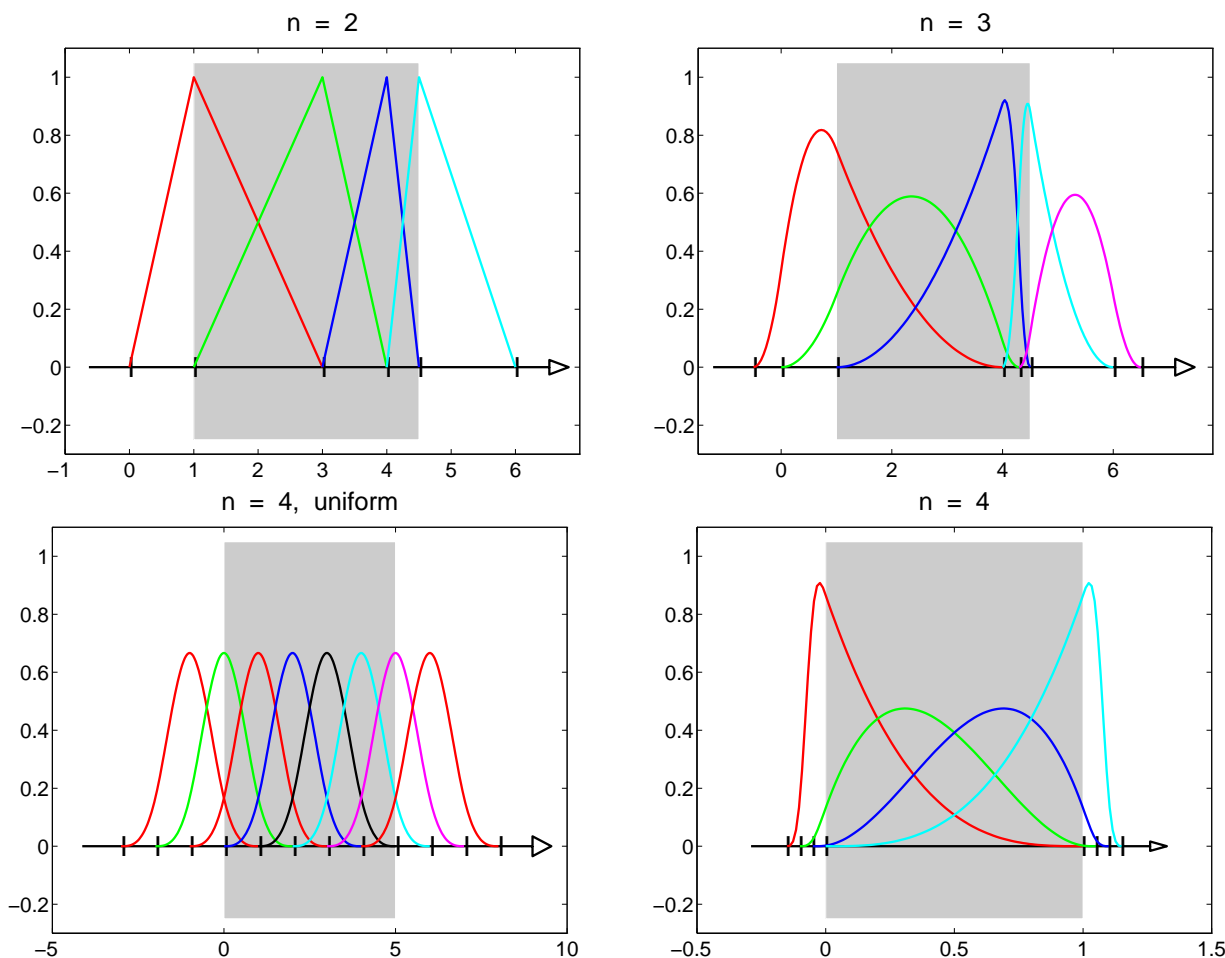
Sei T einfach. Jede Folge b_1^n, \dots, b_m^n von Splines gemäß der obigen Konstruktion bildet eine Basis des Spline-raums $\mathcal{S}_{n,T}(D(T))$.

Durch

$$\sum_{j=1}^m b_j^n(t) \equiv 1, \quad t \in D(T),$$

ist die Basis $\{b_j^n, j = 1 : m\}$ eindeutig bestimmt. Die Basisfunktionen heißen dann *B-Splines*.

Beispiele:



Für beliebiges $\tau \in \mathbb{R}$ gibt es Funktionen $\psi_j^n(\tau)$ mit

$$(t - \tau)^{n-1} = \sum_j b_j^n(t) \psi_j^n(\tau), \quad t \in D(T). \quad (1)$$

Die Funktionen $\psi_j^n(\tau)$ sind Polynome der Ordnung n , da

$$0 = D_\tau^n (t - \tau)^{n-1} = \sum_j b_j^n(t) D_\tau^n \psi_j^n(\tau),$$

und folglich $D_\tau^n \psi_j^n(\tau) \equiv 0$. Ist speziell $\tau = \tau_k$, dann gibt es eine Konstante p_k mit

$$(t - \tau_k)^{n-1} = b_k^n(t) p_k, \quad t \in [\tau_k, \tau_{k+1}).$$

Also ist

$$\sum_j b_j^n(t) \psi_j^n(\tau_k) - b_k^n(t) p_k \equiv 0, \quad t \in [\tau_k, \tau_{k+1})$$

und nach dem Satz „lokale Basis“ gilt $\psi_j^n(\tau_k) = 0$ für $k - n < j < k$, bzw.

$$\psi_j^n(\tau_k) = 0 \quad \text{für} \quad j < k < j + n.$$

Es gibt demnach eine Konstante c_j^n mit

$$\psi_j^n(\tau) = c_j^n (\tau_{j+1} - \tau) \cdots (\tau_{j+n-1} - \tau).$$

Dividiert man (1) durch $(-\tau)^{n-1}$ und betrachtet den Grenzwert $\tau \rightarrow \infty$, dann erhält man

$$1 = \sum_j b_j^n(t) c_j^n \quad \Rightarrow \quad c_j^n = 1.$$

Theorem: Marsden-Identität

Für einfaches T , $t \in D(T)$ und

$$\psi_j^n(\tau) := (\tau_{j+1} - \tau) \cdots (\tau_{j+n-1} - \tau)$$

gilt

$$(t - \tau)^{n-1} = \sum_j b_j^n(t) \psi_j^n(\tau).$$

Übernimmt man die Definition der Funktionen ψ_j^n für beliebiges T , dann sind die Funktionen $\psi_{k-n+1}^n, \dots, \psi_k^n$ linear unabhängig, sofern $\tau_k < \tau_{k+1}$. Damit kann die Marsden-Identität zur Definition von B-Splines für beliebige Knotenfolgen verwendet werden.

Definition: B-Splines

Sei $T = [\tau_1, \dots, \tau_{m+n}]$ ein beliebiger Knotenvektor. Die *B-Splines* b_j^n , $j = 1, \dots, m$ sind definiert durch

$$\text{supp } b_j^n \subset s_j^n := [\tau_j, \tau_{j+n}]$$

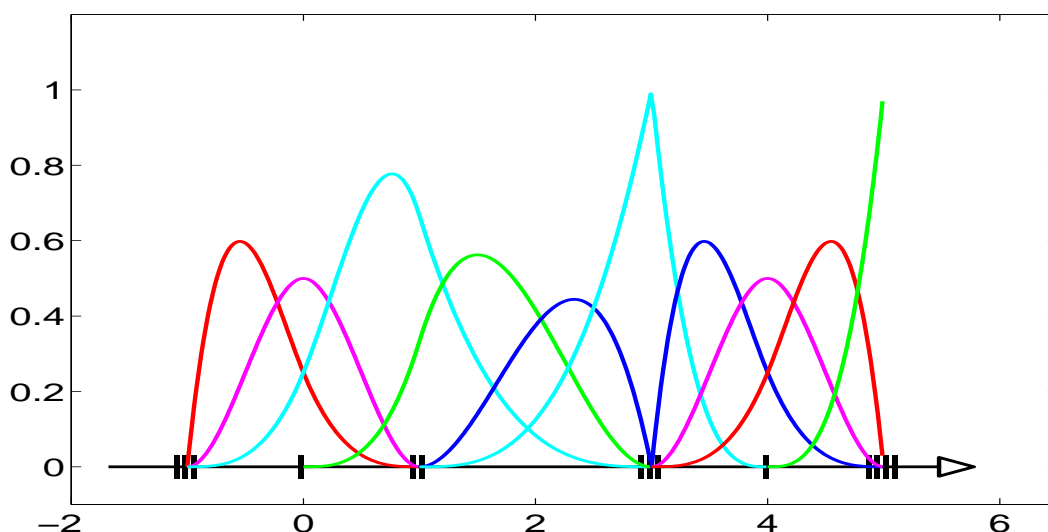
und

$$(t - \tau)^{n-1} = \sum_j b_j^n(t) \psi_j^n(\tau), \quad t \in D(T) = [\tau_n, \tau_{m+1}].$$

Bemerkungen:

- Für $t \in [\tau_k, \tau_{k+1})$ kann der Summationsindex auf $j = k - n + 1 : k$ eingeschränkt werden. Dadurch entsteht ein lineares Gleichungssystem für die Berechnung von b_j^n auf dem Intervall $[\tau_k, \tau_{k+1})$.
- Für $\tau_j = \tau_{j+n}$ ist $b_j^n = 0$.
- Für einfache Knoten stimmt die Definition mit der von Seite 54 überein.

$n = 4$, vgl. Seite 51



Theorem: Globale Basis

Sei $T = [\tau_1, \dots, \tau_{m+n}]$ nichtentartet, dann bilden die B-Splines b_1^n, \dots, b_m^n eine Basis des Splineraums $\mathcal{S}_{n,T}(D(T))$.

Beweis: Es wird gezeigt, dass sich alle abgebrochenen Potenzen a_k^n als Linearkombination von B-Splines darstellen lassen. Da die Anzahl der B-Splines mit der Raumdimension übereinstimmt, folgt die Behauptung.

Sei zunächst $\#k = 0$, dann gilt

$$a_k^n(t) = (t - \tau_k)_+^{n-1} = \sum_{j \geq k} b_j^n(t) \psi_j^n(\tau_k),$$

da $\psi_j^n(\tau_k) = 0$ für $k - n < j < k$.

Allgemein gilt für $\#k = \ell$:

$$a_k^n(t) = (t - \tau_k)_+^{n-1-\ell} = \frac{(-1)^\ell (n-1-\ell)!}{(n-1)!} \sum_{j \geq k} b_j^n(t) D_\tau^\ell \psi_j^n(\tau_k).$$

Dabei verwendet man, dass $\psi_j^n(\tau)$ für $k - n + \ell < j < k$ einen $(\ell + 1)$ -fachen Faktor $(\tau_k - \tau)$ besitzt und folglich $D_\tau^\ell \psi_j^n(\tau_k) = 0$ ist.

Lokale Basis: Einschränkung der Knotenfolge auf $\tau_{k-n+1}, \dots, \tau_{k+n}$ zeigt: Die B-Splines bilden eine lokale Basis im Sinne von Seite 54.

Da die Funktionen ψ_j stetig von den Knoten abhängen, hängt auch $b_j^n(t)$ stetig von den Knoten ab, sofern t kein Knoten mit Vielfachheit $\geq n$ ist. Dies ermöglicht es häufig, Eigenschaften für einfache Knoten abzuleiten und durch Grenzübergang auf beliebige Knotenfolgen zu übertragen. Z.B. kann man damit zeigen, dass $b_j^n(t)$ nur von den Knoten $\tau_j, \dots, \tau_{j+n}$ abhängt.

Partition der Eins: Dividiert man die Marsden-Identität durch $(-\tau)^{n-1}$, dann erhält man für $\tau \rightarrow \infty$

$$\sum_j b_j^n(t) \equiv 1, \quad t \in D(T).$$

Definition: Splines in B-Splineform

Ein *Spline in B-Splineform* ist gegeben durch

$$f(t) = \sum_j b_j^n(t) p_j.$$

Die Koeffizienten p_j heißen *Kontrollpunkte* (oder auch: *Boor-Punkte*, *B-Splinekoeffizienten*, *Steuerpunkte*).

Fasst man die B-Splines zu einem Zeilenvektor $B^n := [b_1^n, \dots, b_m^n]$ zusammen, so erhält man die Darstellung

$$f(t) = B^n P.$$

Verwendet man Kontrollpunkte $\mathbf{p}_j \in \mathbb{R}^d$, so entsteht eine *Splinekurve* in \mathbb{R}^d ,

$$\mathbf{f}(t) = B^n \mathbf{P}.$$

Differenziation von B-Splines: Aus $f \in \mathcal{S}_{n,T}$ folgt $Df \in \mathcal{S}_{n-1,T}$. Für die Ableitung eines B-Splines mit einfachen Knoten gilt

$$\text{supp } Db_j^n \subset [\tau_j, \tau_{j+n}] \quad \Rightarrow \quad Db_j^n = \alpha_j b_j^{n-1} + \beta_{j+1} b_{j+1}^{n-1}.$$

Die Konstanten lassen sich folgendermaßen bestimmen: Zunächst folgt aus der Partition der Eins

$$0 = D \sum_j b_j^n(t) = \sum_j Db_j^n(t) = \sum_j (\alpha_j + \beta_j) b_j^{n-1}(t)$$

und damit $\alpha_j = -\beta_j$. Die Marsden-Identität liefert nun

$$\begin{aligned} (n-1)t^{n-2} &= \sum_j \psi_j^n(0) Db_j^n(t) \\ &= \sum_j \alpha_j (\psi_j^n(0) - \psi_{j-1}^n(0)) b_j^{n-1}(t). \end{aligned}$$

Vergleich mit

$$(n-1)t^{n-2} = (n-1) \sum_j \psi_j^{n-1}(0) b_j^{n-1}(t)$$

ergibt

$$\alpha_j = \frac{(n-1) \psi_j^{n-1}(0)}{\psi_j^n(0) - \psi_{j-1}^n(0)} = \frac{n-1}{\tau_{j+n-1} - \tau_j}$$

und schließlich

$$Db_j^n = (n-1) \left(\frac{b_j^{n-1}}{|s_j^{n-1}|} - \frac{b_{j+1}^{n-1}}{|s_{j+1}^{n-1}|} \right).$$

Aufgrund der stetigen Abhängigkeit der B-Splines von den Knoten ist diese Gleichung für beliebige Knotenfolgen gültig (sofern an den Knoten eine Ableitung definiert ist). Ist einer der Träger degeneriert, dann wird der entsprechende Summand weggelassen.

Differenziation von Splines: Sei

$$f(t) = \sum_j b_j^n(t) p_j \in \mathcal{S}_{n,T}$$

ein Spline der Ordnung n über T , dann ist

$$Df(t) = \sum_j b_j^{n-1}(t) \tilde{p}_j$$

ein Spline der Ordnung $n - 1$ über T .

Für die Kontrollpunkte der Ableitung gilt

$$\tilde{p}_j = \frac{(n-1)(p_j - p_{j-1})}{|s_j^{n-1}|}.$$

Die Knotenfolge der Ableitung kann reduziert werden, wenn nur der Definitionsbereich $D(T)$ betrachtet wird: Zunächst ist der erste und der letzte Knoten überflüssig, außerdem kann ein n -facher Knoten zu einem $(n-1)$ -fachen Knoten reduziert werden. Es ist also $Df \in \mathcal{S}_{n-1,DT}$ mit

$$\begin{aligned} T &= \tau_1, \dots, \tau_k \S n, \dots, \tau_{m+n} \\ DT &= \tau_2, \dots, \tau_k \S (n-1), \dots, \tau_{m+n-1} \end{aligned}$$

Wenn T nicht entartet ist, dann ist auch DT nicht entartet.

Integration: Zur Integration eines Splines muss die Knotenfolge erweitert werden, beispielsweise durch Verdopplung des ersten und letzten Knotens:

$$D^{-1}T = \tau_1, \tau_1, \dots, \tau_{m+n}, \tau_{m+n} .$$

Es gilt

$$D^{-1}b_j^n(t) := \int_{-\infty}^t b_j^n(\tau) d\tau = \frac{|s_j^n|}{n} \sum_{k \geq j} b_k^{n+1}(t)$$

und insbesondere

$$\int_{-\infty}^{\infty} b_j^n(\tau) d\tau = \frac{|s_j^n|}{n} .$$

Formel für Spline $D^{-1}f(t)$ als Übung.

Rekursion: Es gilt

$$(t - \tau)^n = \sum_j b_j^{n+1}(t) \psi_j^{n+1}(\tau) = (t - \tau) \sum_j b_j^n(t) \psi_j^n(\tau) .$$

Substitution der Bézierdarstellung

$$(t - \tau) = \frac{(\tau_{j+n} - \tau)(t - \tau_j) + (\tau_j - \tau)(\tau_{j+n} - t)}{|s_j^n|}$$

ergibt

$$\begin{aligned} (t - \tau)^n &= \sum_j \left(\psi_j^{n+1}(\tau)(t - \tau_j) + \psi_{j-1}^{n+1}(\tau)(\tau_{j+n} - t) \right) \frac{b_j^n(t)}{|s_j^n|} \\ &= \sum_j \psi_j^{n+1}(\tau) \left(\frac{t - \tau_j}{|s_j^n|} b_j^n(t) + \frac{\tau_{j+n+1} - t}{|s_{j+1}^n|} b_{j+1}^n(t) \right) . \end{aligned}$$

Theorem: Rekursion

Mit

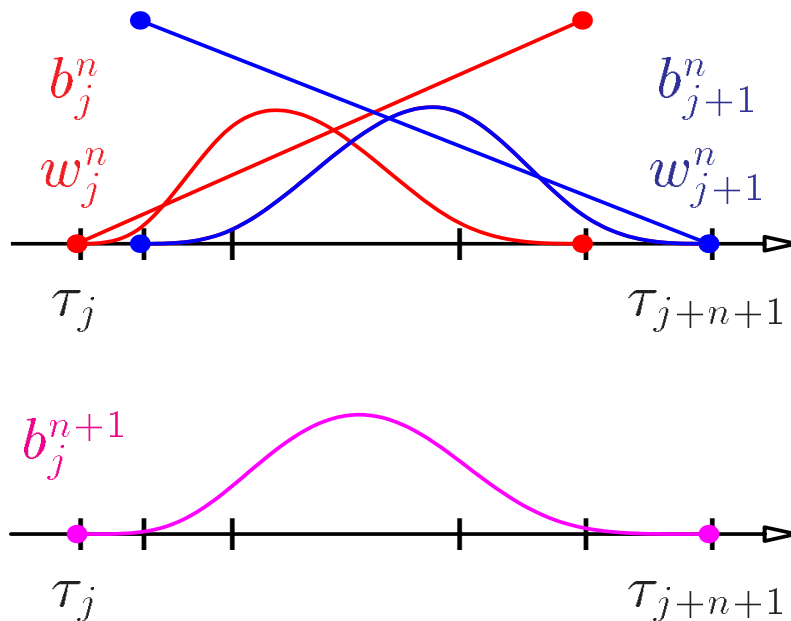
$$w_j^n(t) := \frac{t - \tau_j}{|s_j^n|}$$

gilt die Rekursionsformel

$$b_j^1 = \chi([\tau_j, \tau_{j+1}))$$

$$b_j^{n+1} = w_j^n b_j^n + (1 - w_{j+1}^n) b_{j+1}^n .$$

Summanden mit leerem Träger werden weggelassen.



Da alle Gewichte w_j^n auf s_j^n nichtnegativ sind, folgt:

Theorem: Partition der Eins

Die B-Splines b_1^n, \dots, b_m^n bilden eine nichtnegative Partition der Eins auf dem Intervall $D(T)$.

Für einen Spline $f(t) = \sum_j b_j^n(t)p_j$ gilt

$$f(t) = \sum_j b_j^{n-1}(t) (w_j^{n-1}(t)p_j + (1 - w_j^{n-1}(t))p_{j-1}) .$$

Halte $t = t_0$ fest, dann stimmen die Splines $f(t)$ und

$$\tilde{f}(t) := \sum_j b_j^{n-1}(t) (w_j^{n-1}(t_0)p_j + (1 - w_j^{n-1}(t_0))p_{j-1})$$

an der Stelle $t = t_0$ überein.

- $\tilde{f}(t) = \sum_j \tilde{p}_j b_j^{n-1}(t)$ ist ein Spline der Ordnung $n - 1$ über T .

- Die neuen Kontrollpunkte

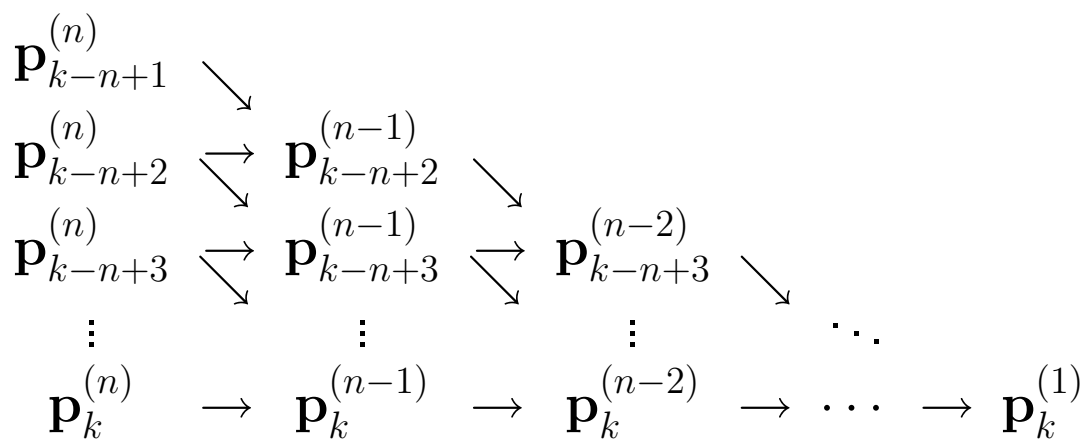
$$\tilde{p}_j := w_j^{n-1}(t_0)p_j + (1 - w_j^{n-1}(t_0))p_{j-1}$$

sind Konvexkombinationen der alten Kontrollpunkte.

- Ist $t_0 \in [\tau_k, \tau_{k+1})$, dann sind für den Spline f die Kontrollpunkte p_{k-n+1}, \dots, p_k relevant; für den Spline \tilde{f} sind die Kontrollpunkte $\tilde{p}_{k-n+2}, \dots, \tilde{p}_k$ relevant;
- Iteration ergibt ein Dreiecksschema analog zum de Casteljau-Algorithmus.

de Boor-Algorithmus: Auswertung des Splines $\mathbf{f}(t) = \sum_j b_j^n(t) \mathbf{p}_j$ der Ordnung n über T an der Stelle $t = t_0 \in D(T)$.

- Bestimme Index k so, dass $t_0 \in [\tau_k, \tau_{k+1})$.
- Initialisiere Dreiecksschema mit den Kontrollpunkten $\mathbf{p}^{(n)} := [\mathbf{p}_{k-n+1}, \dots, \mathbf{p}_k]$.
- Berechne rekursiv das Dreiecksschema



Zur Berechnung des Punktes $\mathbf{p}_j^{(\nu)}$ verwendet man das Gewicht

$$w_j^{(\nu)}(t_0) = \frac{t_0 - \tau_j}{|s_j^{(\nu)}|}.$$

Der Pfeil \rightarrow steht für Multiplikation mit $w_j^{(\nu)}(t_0)$.

Der Pfeil \searrow steht für Multiplikation mit $(1 - w_j^{(\nu)}(t_0))$.

Der Spline $\mathbf{f}^{(\nu)}(t) := \sum_j \mathbf{p}_j^{(\nu)} b_j^{(\nu)}(t)$ der Ordnung ν stimmt mit $\mathbf{f}(t)$ and der Stelle $t = t_0$ überein.

- Der gesuchte Wert ist $\mathbf{f}(t_0) = \mathbf{p}_k^{(1)}$.

Beispiel: $n = 4, t_0 = 2$

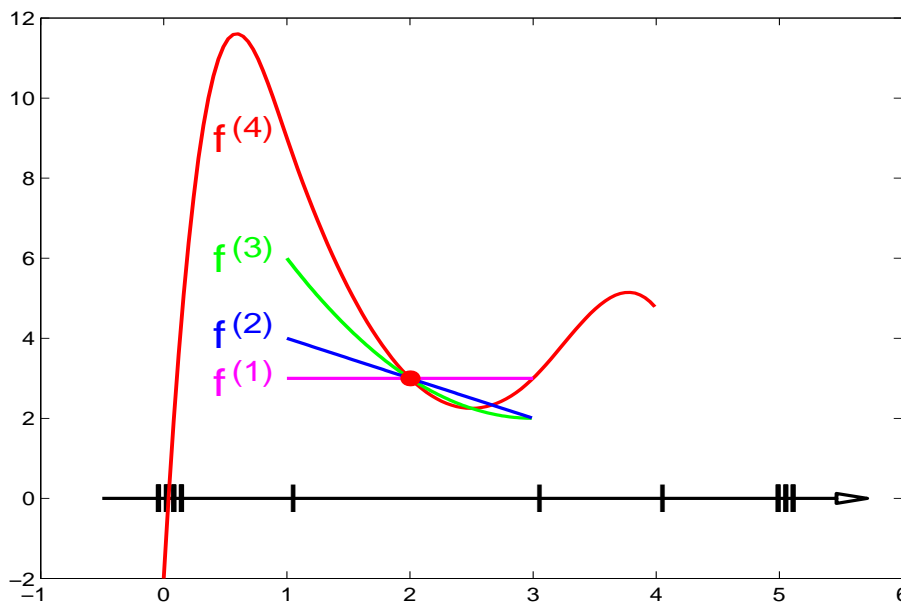
$$T = 0, 0, 0, 0, 1, 3, 4, 5, 5, 5$$

$$p = -2, 16, 4, 0, 8, -1$$

- Es gilt $2 \in [1, 3) \Rightarrow k = 5$.
- Die relevanten Kontrollpunkte sind $p^{(4)} = 16, 4, 0, 8$.
- Das Dreiecksschema ist

$$\begin{array}{cccc} 16 & & & \\ 4 & 8 & & \\ 0 & 2 & 4 & \\ 8 & 2 & 2 & 3 \end{array}$$

- Der gesuchte Funktionswert ist $f(2) = 3$.



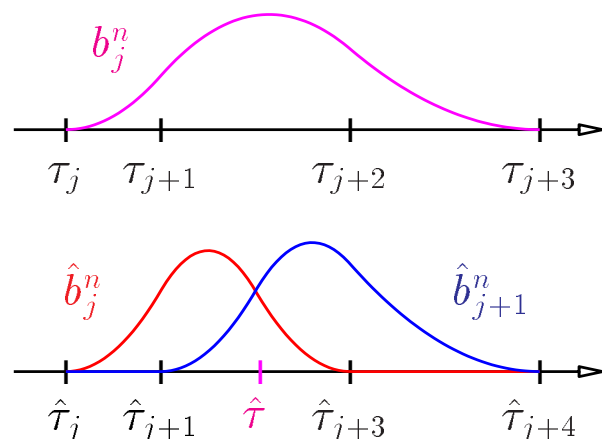
Knoteneinfügen: Sei \hat{T} die Knotenfolge, die entsteht, wenn in T der Knoten $\hat{\tau} \in [\tau_k, \tau_{k+1})$ eingefügt wird,

$$\hat{\tau}_j := \begin{cases} \tau_j & \text{für } j \leq k \\ \hat{\tau} & \text{für } j = k + 1 \\ \tau_{j-1} & \text{für } j > k + 1 . \end{cases}$$

Seien \hat{b}_j^n die B-Splines über \hat{T} , dann gilt wegen $\mathcal{S}_{n,T} \subset \mathcal{S}_{n,\hat{T}}$

$$b_j^n = \begin{cases} \hat{b}_j^n & \text{für } j \leq k - n \\ \alpha_j \hat{b}_j^n + \beta_j \hat{b}_{j+1}^n & \text{für } k - n < j \leq k \\ \hat{b}_{j+1}^n & \text{für } j > k . \end{cases}$$

Zur Bestimmung der Konstanten α_j, β_j sei zunächst wieder T einfach, das allgemeine Resultat folgt dann aus Stetigkeitsgründen.



Für $t \in [\tau_j, \tau_{j+1})$ gilt (siehe Seite 58)

$$b_j^n(t) = a_j^n(t) / \Psi_j^n(\tau_j), \quad \hat{b}_j^n(t) = a_j^n(t) / \hat{\Psi}_j^n(\hat{\tau}_j), \quad \hat{b}_{j+1}^n(t) = 0 .$$

Daraus folgt für α_j (und analog auch für β_j)

$$\alpha_j = \frac{\hat{\tau} - \hat{\tau}_j}{\hat{\tau}_{j+n} - \hat{\tau}_j} = \hat{w}_j^n(\hat{\tau}), \quad \beta_j = 1 - \hat{w}_{j+1}^n(\hat{\tau}) .$$

Für einen Spline $f(t) = \sum_j b_j^n(t)p_j$ ist

$$\begin{aligned}
 f(t) &= \sum_{j \leq k-n} b_j^n(t)p_j + \sum_{j > k} b_j^n(t)p_j + \sum_{j=k-n+1}^k b_j^n(t)p_j \\
 &= \sum_{j \leq k-n} \hat{b}_j^n(t)p_j + \sum_{j > k+1} \hat{b}_j^n(t)p_{j-1} \\
 &\quad + \sum_{j=k-n+1}^k (\hat{w}_j^n(\hat{\tau}) \hat{b}_j^n(t) + (1 - \hat{w}_{j+1}^n(\hat{\tau})) \hat{b}_{j+1}^n(t))p_j .
 \end{aligned}$$

Es gilt

$$\hat{w}_j^n(\hat{\tau}) = \begin{cases} 1 & \text{für } j = k - n + 1 \\ w_j^{n-1}(\hat{\tau}) & \text{für } k - n + 1 < j \leq k \\ 0 & \text{für } j = k + 1 \end{cases}$$

und damit

$$\begin{aligned}
 f(t) &= \sum_{j \leq k-n+1} \hat{b}_j^n(t)p_j + \sum_{j > k} \hat{b}_j^n(t)p_{j-1} \\
 &\quad + \sum_{j=k-n+2}^k \hat{b}_j^n(t) (w_j^{n-1}(\hat{\tau}) p_j + (1 - w_j^{n-1}(\hat{\tau})) p_{j-1}) .
 \end{aligned}$$

Die neuen Kontrollpunkte stehen also in der zweiten Spalte des de Boor-Dreiecks zur Auswertung von f an der Stelle $t = \hat{\tau}$.

Iteration ergibt: Der de Boor-Algorithmus zur Auswertung von f an der Stelle $t = \hat{\tau}$ entspricht dem Einfügen eines $(n - 1)$ -fachen Knotens $\hat{\tau}$. Insbesondere gilt

$$\#j = n - 2 \Rightarrow f(\tau_j) = p_{j-1} .$$

Beispiel: siehe Seite 66

$$T = 0, 0, 0, 0, 1, 3, 4, 5, 5, 5$$

$$p = -2, 16, 4, 0, 8, -1$$

$$\mapsto T = 0, 0, 0, 0, 1, 2, 3, 4, 5, 5, 5$$

$$p = -2, 16, 8, 2, 2, 8, -1$$

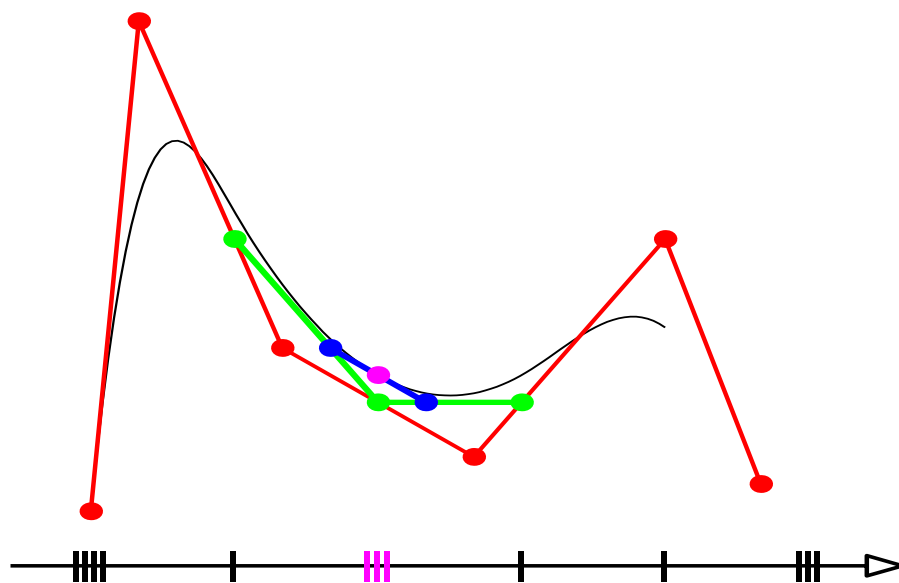
$$\mapsto T = 0, 0, 0, 0, 1, 2, 2, 3, 4, 5, 5, 5$$

$$p = -2, 16, 8, 4, 2, 2, 8, -1$$

$$\mapsto T = 0, 0, 0, 0, 1, 2, 2, 2, 3, 4, 5, 5, 5$$

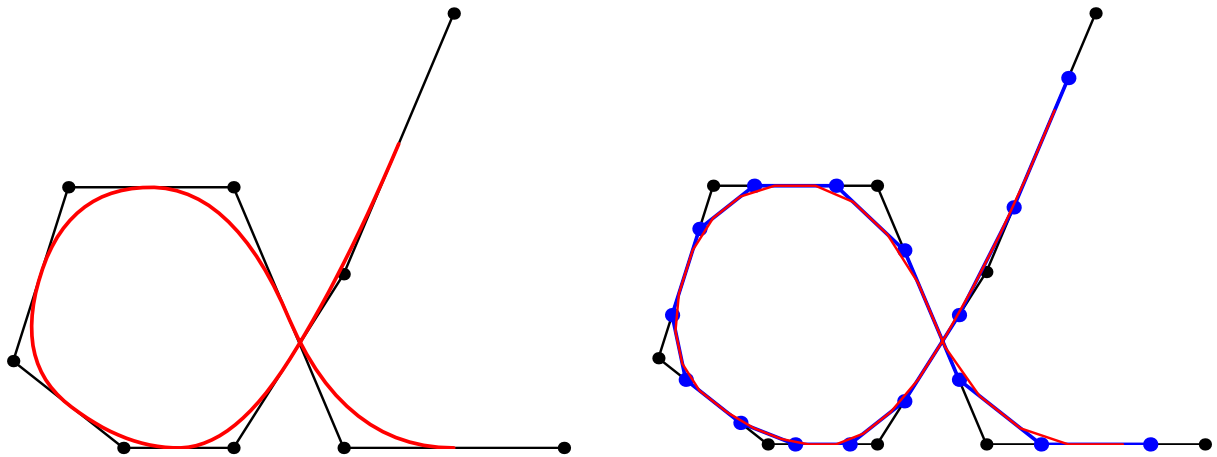
$$p = -2, 16, 8, 4, 3, 2, 2, 8, -1$$

$$\begin{array}{cccc}
 -2 & & -2 & & -2 & & -2 \\
 16 & & 16 & & 16 & & 16 \\
 4 & \mapsto & 4 & 8 & \mapsto & 4 & 8 & \mapsto & 4 & 8 \\
 0 & & 0 & 2 & \mapsto & 0 & 2 & 4 & \mapsto & 0 & 2 & 4 \\
 8 & & 8 & 2 & & 8 & 2 & 2 & & 8 & 2 & 2 & 3 \\
 -1 & & -1 & & -1 & & -1 & & -1 & & -1 & &
 \end{array}$$



Konvergenz: Wiederholt man das Knoteneinfügen so, dass der maximale Abstand zwischen zwei aufeinanderfolgenden Knoten klein wird, so konvergiert die Folge der erzeugten Kontrollpolygone rasch gegen die definierte Splinekurve.

Beispiel: $n = 3$, uniforme Knotenfolge, füge jeweils in die Mitte zwischen zwei Knoten einen neuen Knoten ein.



Knotenentfernen Das Entfernen von Knoten ist im Allgemeinen nur näherungsweise möglich. Man kann analog zur Gradreduktion bei Bézierkurven vorgehen. Soll ein Knoten τ_j entfernt werden, dann bestimmt man die neuen Kontrollpunkte $\tilde{\mathbf{p}}$ bezüglich der reduzierten Knotenfolge \tilde{T} so, dass Einfügen von τ_j in \tilde{T} eine minimale Abweichung von den ursprünglich gegebenen Kontrollpunkten erzeugt.

Ist I die $(n + 1) \times n$ -Matrix, die das Einfügen von τ_j für die relevanten Kontrollpunkte beschreibt, dann ist die $n \times (n + 1)$ -Matrix R für das Entfernen von τ_j gegeben durch

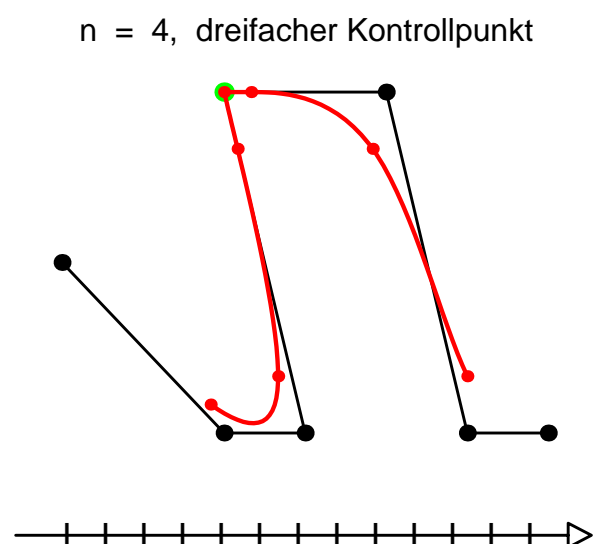
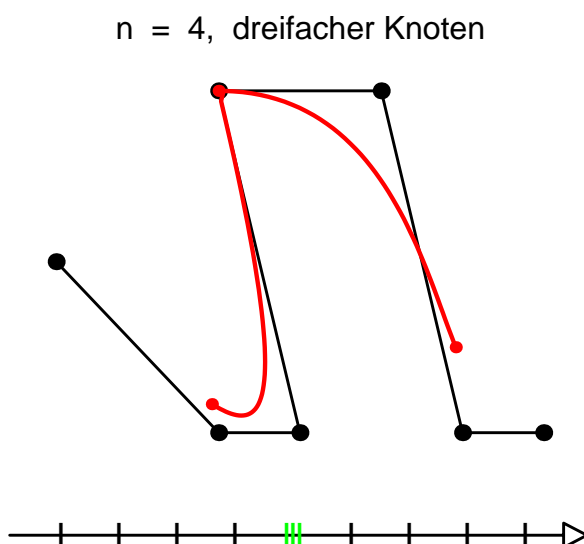
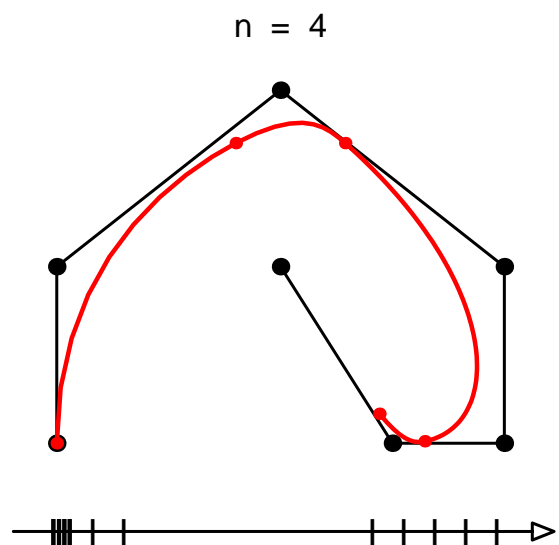
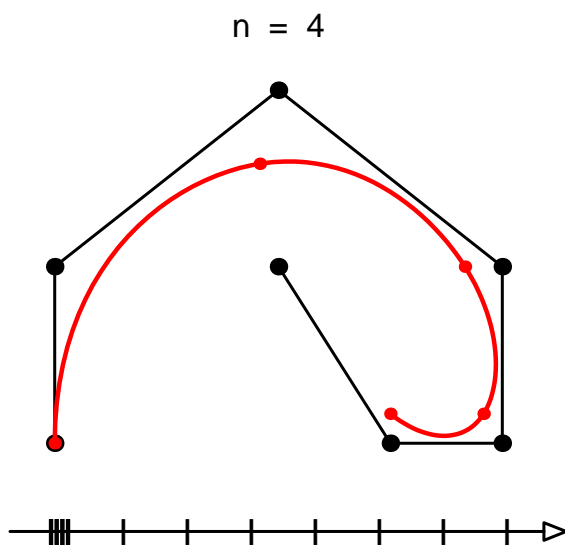
$$R := (I^t I)^{-1} I^t.$$

Geometrie von Splinekurven

Eine *B-Splinekurve* in \mathbb{R}^d hat die Form

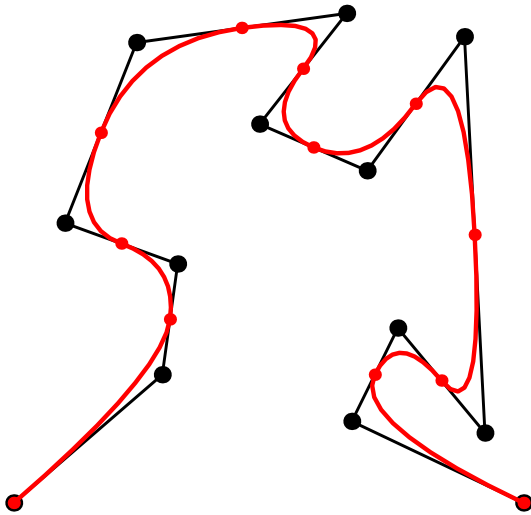
$$\mathbf{f}(t) = \sum_j b_j^n(t) \mathbf{p}_j = B^n(t) \mathbf{P} \quad \mathbf{p}_j \in \mathbb{R}^d .$$

Das Kontrollpolygon \mathbf{p} von \mathbf{f} ist die stückweise geradlinige Verbindung der Kontrollpunkte.

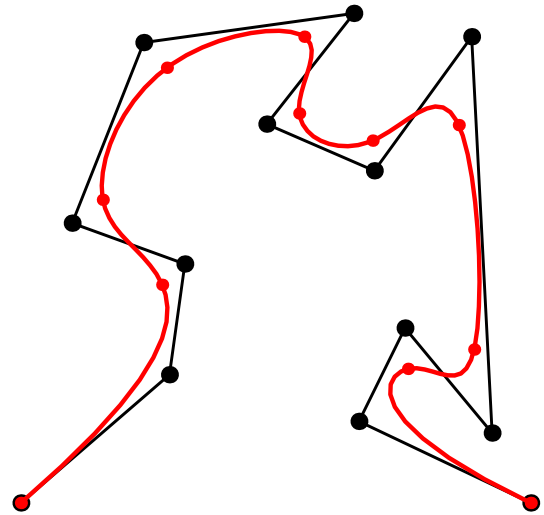


Wachsender Grad bei gleichbleibendem Kontrollpolygon bewirkt eine zunehmende Glättung der Splinekurve. Im Beispiel ist der erste und letzte Knoten jeweils n -fach, dazwischen ist die Unterteilung äquidistant.

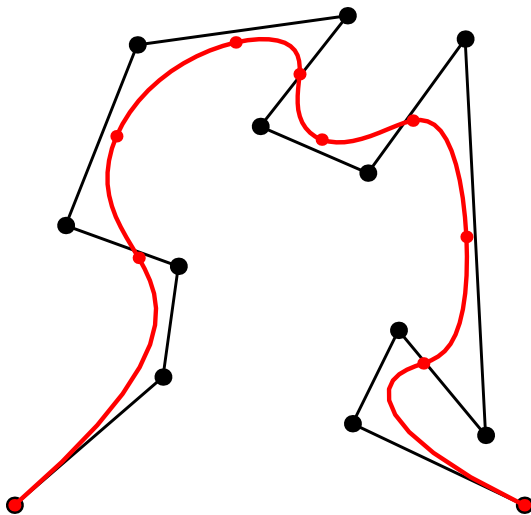
$n = 3$



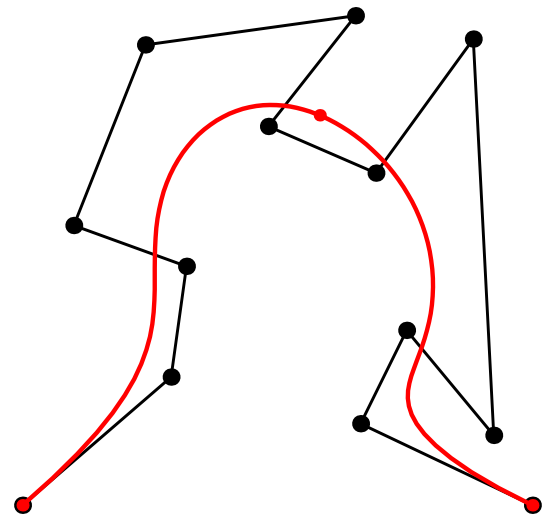
$n = 4$



$n = 5$



$n = 12$



Da B-Splines eine nicht-negative Partition der Eins bilden, haben B-Splinekurven unter anderem folgende Eigenschaften:

- **Konvexe Hülle** Die B-Splinekurve liegt in der konvexen Hülle des Kontrollpolygons,

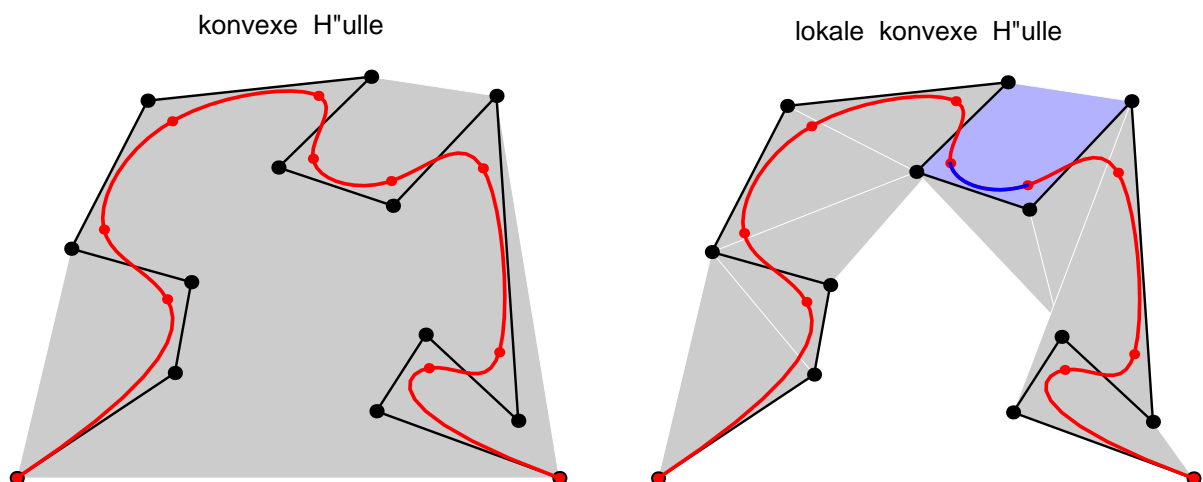
$$\mathbf{f}(t) \in \text{conv}(\mathbf{P}) , \quad t \in D(T) .$$

- **Lokale konvexe Hülle** Das Segment $\mathbf{f}([\tau_k, \tau_{k+1}))$ einer B-Splinekurve liegt in der konvexen Hülle der relevanten Kontrollpunkte,

$$\mathbf{f}(t) \in \text{conv}(\mathbf{p}_{k-n+1}, \dots, \mathbf{p}_k) , \quad t \in [\tau_k, \tau_{k+1}) .$$

- **Affine Invarianz** Die affine Abbildung der Kurve entspricht der affinen Abbildung der Kontrollpunkte,

$$A\mathbf{f}(t) + \mathbf{a} = \sum_j b_j^n(t)(A\mathbf{p}_j + \mathbf{a}) .$$



Lineare Präzision: Die *Greville-Abszissen* ("Mittel der inneren Knoten") sind definiert durch

$$\tau_j^* := (\tau_{j+1} + \dots + \tau_{j+n-1}) / (n - 1) .$$

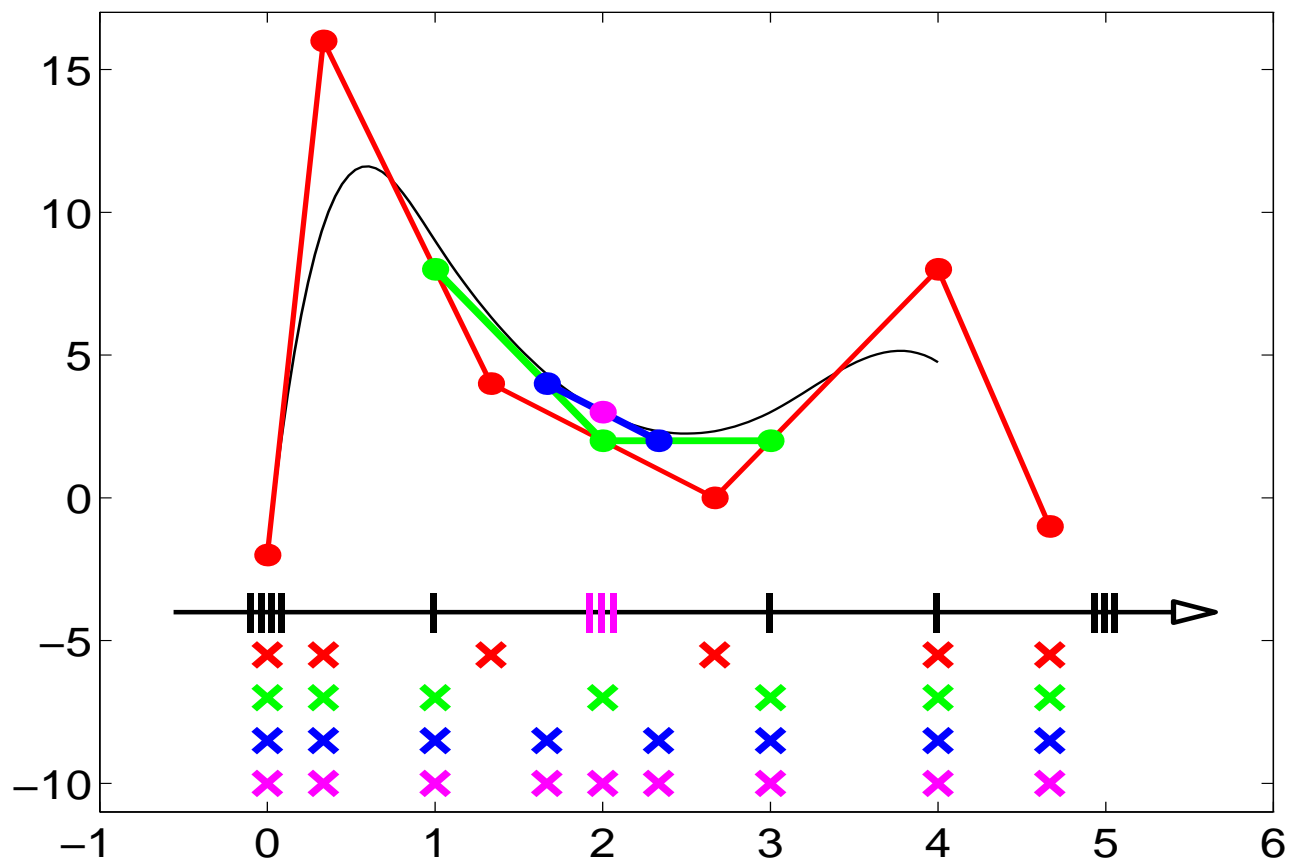
Damit gilt (siehe Aufgabe XX):

$$t = \sum_j b_j^n(t) \tau_j^* .$$

Skalare Kontrollpolygone: Im skalaren Fall $d = 1$ identifiziert man die Funktion $f(t)$ mit der Kurve $[t, f(t)]$,

$$f(t) = \sum_j b_j^n(t) p_j \longmapsto [t, \sum_j b_j^n(t) p_j] = \sum_j b_j^n(t) [\tau_j^*, p_j] .$$

Beispiel: Skalares Kontrollpolygon, de Boor-Algorithmus, siehe Seite 66.



Verallgemeinerte Bézierkurven: Sei

$$T = \dots, \tau_k, \tau_{k+1}, \dots,$$

dann sind alle Splinesegmente unabhängig voneinander. Auf einem Intervall $[\alpha, \beta) := [\tau_k, \tau_{k+1})$ sind genau die B-Splines $b_{k-n+1}^n, \dots, b_k^n$ von Null verschieden. Diese stimmen mit skalierten Bernsteinpolynomen \tilde{b}_j^n überein,

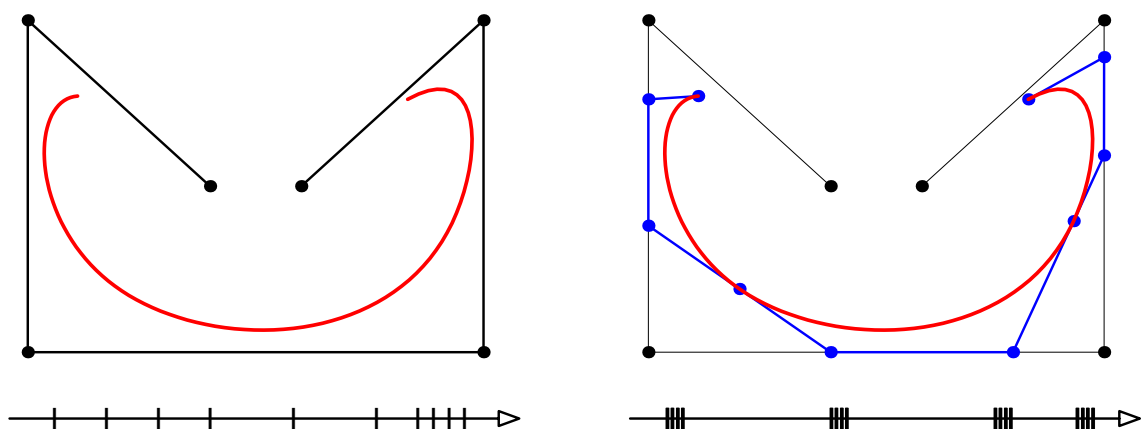
$$b_{k-n+j}^n(t) = \tilde{b}_j^n \left(\frac{t - \alpha}{\beta - \alpha} \right), \quad t \in [\tau_k, \tau_{k+1}).$$

Folglich gilt für $t \in [\alpha, \beta)$

$$\mathbf{f}(t) = \sum_{j=1}^n \tilde{b}_{k-n+j}^n(t) \mathbf{p}_{k-n+j} = \sum_{j=1}^n b_j^n \left(\frac{t - \alpha}{\beta - \alpha} \right) \mathbf{p}_{k-n+j}.$$

Das Splinesegment stimmt also (bis auf Umparametrisierung) mit der Bézierkurve zu den relevanten Kontrollpunkten überein.

Die Bézierdarstellung einer B-Splinekurve erhält man, indem man mit Hilfe des de Boor-Algorithmus alle Knoten auf die Vielfachheit n bringt.

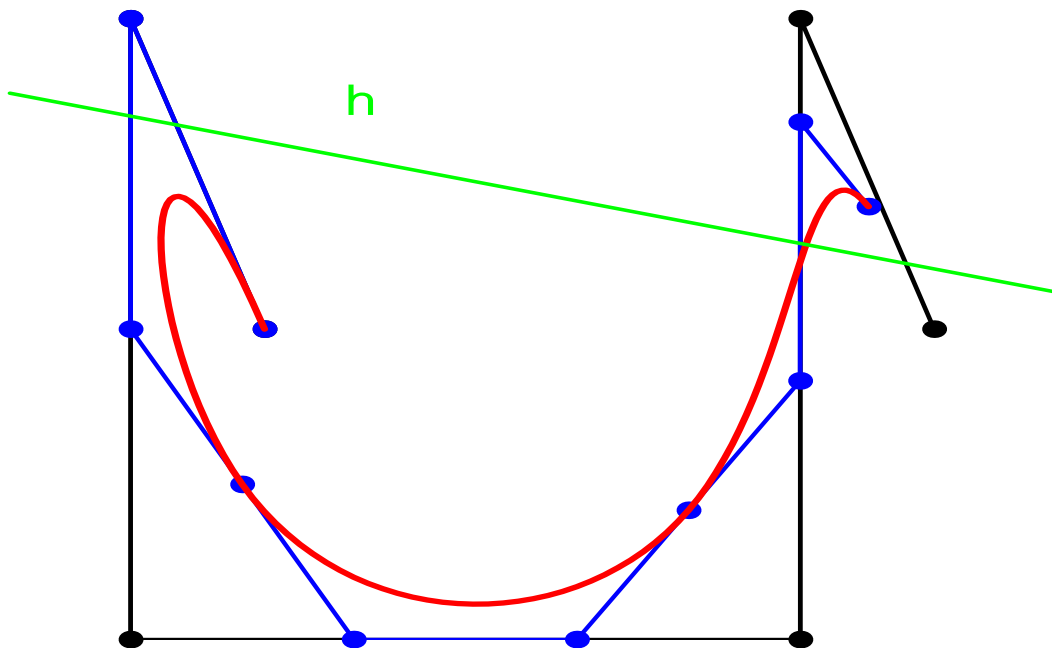


Theorem: Variationsminderung

B-Splinekurven sind variationsmindernd. Das heißt, für jede Hyperebene $h : \mathbf{x}\mathbf{a}^t - d = 0$ und eine Splinekurve $\mathbf{f} = B^n\mathbf{P}$ gilt

$$\#VZW(\mathbf{f}\mathbf{a}^t - d) \leq \#VZW(\mathbf{P}\mathbf{a}^t - d).$$

Beweis: Der de Boor-Algorithmus ist variationsmindernd, da nur Konvexkombinationen benachbarter Kontrollpunkte durchgeführt werden. Folglich ist auch die Transformation einer Splinekurve auf Bézierdarstellung variationsmindernd. Da Bézierkurven variationsmindernd sind, folgt die Behauptung.



Lokale Kontrolle: Ändert man den Wert eines einzelnen Kontrollpunkts \mathbf{p}_j , so ändert sich der zugehörige Spline der Ordnung n nur auf dem Intervall $[\tau_j, \tau_{j+n})$.

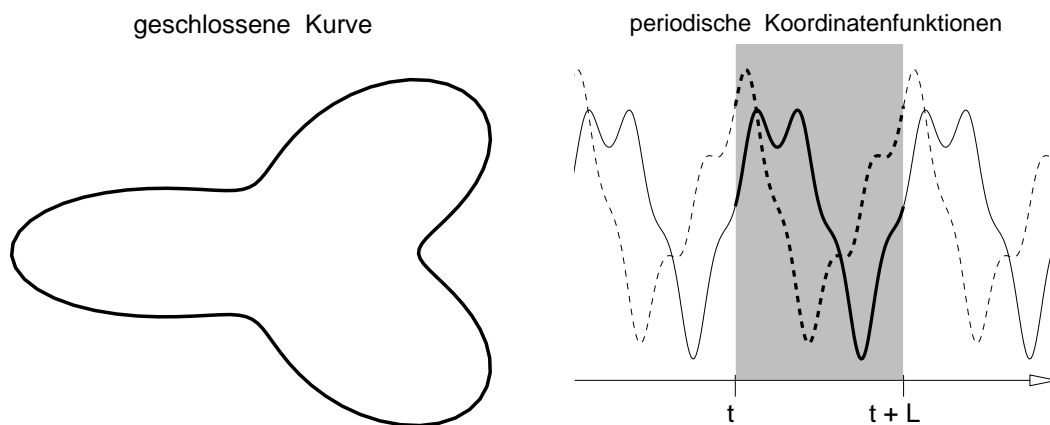
Geschlossene Splinekurven: Geschlossene Kurven werden durch periodische Funktionen dargestellt,

$$\mathbf{c}(t) = \mathbf{c}(t + L) .$$

Die kleinste positive Zahl L , für die diese Gleichung gilt, heißt *Periode* der Funktion. Alle Argumente, die sich um ein Vielfaches von L unterscheiden, werden identifizert:

$$\dots t - 2L \simeq t - L \simeq t \simeq t + L \simeq t + 2L \dots , \quad t \in \mathbb{R} .$$

Das dadurch entstehende Definitionsgebiet ist \mathbb{R}/L .



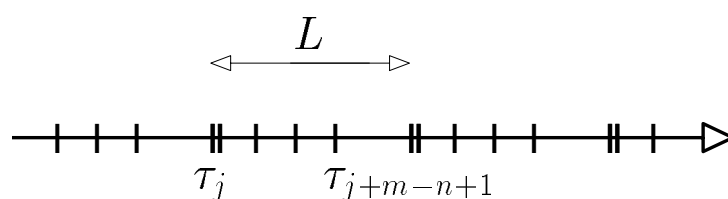
Für periodische Splines muss die Knotenfolge kompatibel zu \mathbb{R}/L sein. Dies ist der Fall, wenn das Definitionsgebiet $D(T)$ die Länge L hat und

$$\tau_{j+m-n+1} = \tau_j + L , \quad \text{für alle } j .$$

Somit müssen für einen periodischen Spline nur die Knoten

$$\tau_n, \dots, \tau_{m+1}$$

angegeben werden. Die Periode L ist dann $L = \tau_{m+1} - \tau_n$.



Periodische B-Splines der Ordnung n lassen sich sinnvoll definieren, wenn $m \geq 2n - 1$ ist. Sie entstehen durch periodische Fortsetzung, d.h. durch

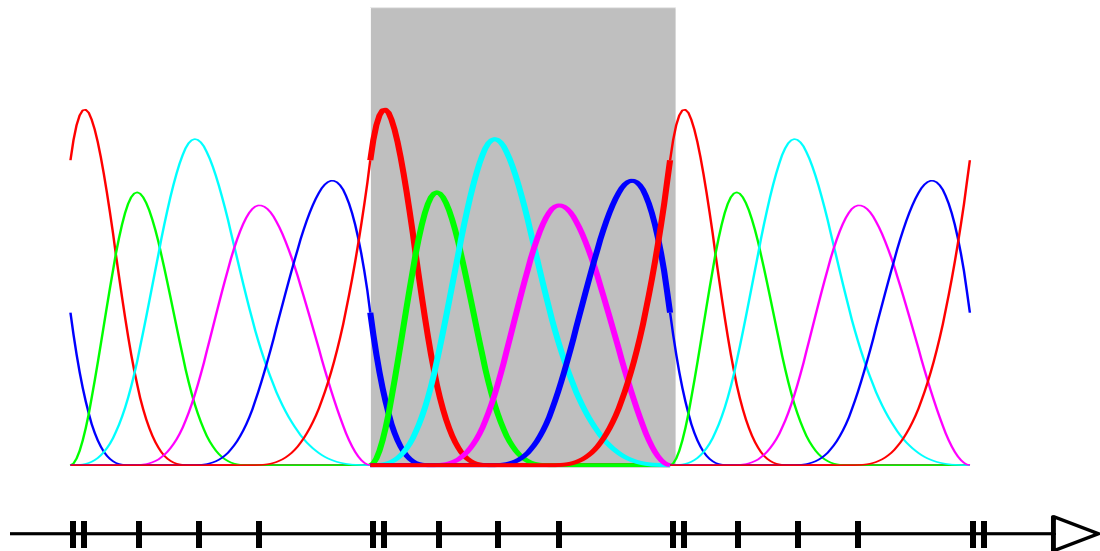
$$b_j^n \simeq \sum_{k \in \mathbb{Z}} b_{j+k(m-n+1)}^n.$$

Damit hat eine geschlossene B-Splinekurve die Form

$$\mathbf{f}(t) = \sum_{j=1}^{m+1-n} b_j^n(t) \mathbf{p}_j, \quad t \in [\tau_n, \tau_{m+1}).$$

Beispiel: $T = [0, 0, 1, 2, 3, 5], n = 4$

periodische B – Splines



Sei $m \geq 2n - 1$. Der Raum der periodischen Splines über der nichtentarteten Knotenfolge

$$T = \tau_n, \dots, \tau_{m+1}$$

hat die Dimension $m + 1 - n$, und die periodischen B-Splines

$$b_1^n, \dots, b_{m+1-n}^n$$

bilden eine Basis.

Beispiel: Approximation des Einheitskreises durch periodische uniforme Splines. Der Fehler err gibt die maximale radiale Abweichung an.

