Übung zur Algorithmik kontinuierlicher Systeme

Übung 6 – Bézierkurven und Interpolation

Sommersemester 2021

Florian Frank

Friedrich-Alexander-Universität Erlangen-Nürnberg

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG
TECHNISCHE FAKULTÄT

Was machen wir heute?

Werbung: Gather.Town

Wo stehen wir?

Rekapitulation — Globale Interpolation

Aufgabe 1 — Polynominterpolation (I)

Rekapitulation — Lokale Interpolation

Aufgabe 1 — Polynominterpolation (II)

Rekapitulation — Freiformmodellierung mit Bézierkurven

Rekapitulation — Multivariate Interpolation

Hausaufgabe — Baryzentrische Koordinaten und Bézierkurven

Werbung: Gather.Town

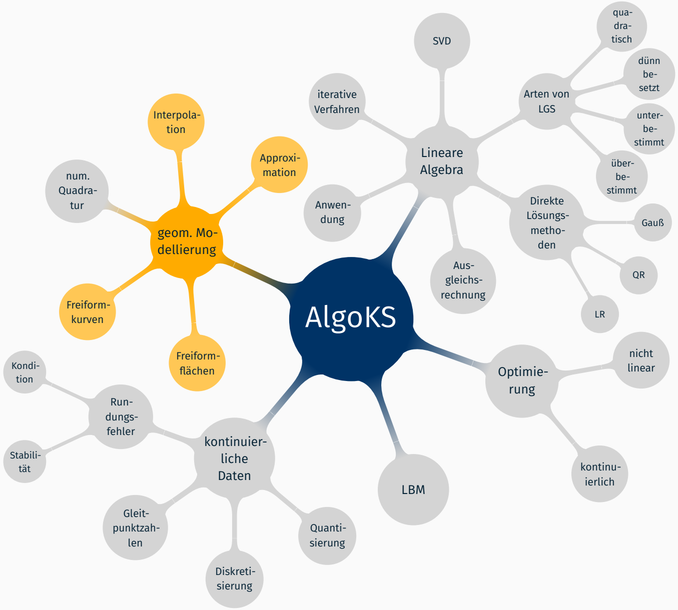

Gather
Video chat in fully customizable and 2D interactive spaces

gather.town — Euer „Space“ während der Übung

- Wir haben für euch einen eigenen Raum auf `gather.town` angelegt. Joint und diskutiert mit anderen AlgoKS-Studierenden über die Aufgaben oder die Vorlesungen :)
- Dazu ist nichts weiter notwendig als ein aktueller Internet-Browser und der folgende Link: https://gather.town/app/ssOyWsnPjAXesbZ4/AlgoKS

Wo stehen wir?

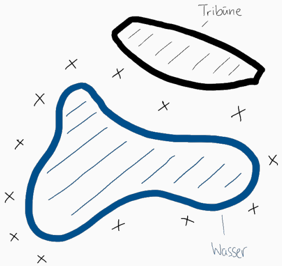
Wo stehen wir?



Rekapitulation — Globale Interpolation

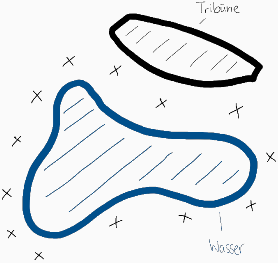
Motivation — Rudi, die rabiate Rennmaus

Rudi, ein Verwandter von Speedy Gonzales, möchte eine Rennstrecke um einen See erstellen. Er gibt einige Punkte vor (in der Skizze durch Kreuze markiert), durch die die Strecke verlaufen soll. Gleichzeitig möchte er aber einen möglichst glatten Streckenverlauf ohne Spitzen.



Motivation — Rudi, die rabiate Rennmaus

Rudi, ein Verwandter von Speedy Gonzales, möchte eine Rennstrecke um einen See erstellen. Er gibt einige Punkte vor (in der Skizze durch Kreuze markiert), durch die die Strecke verlaufen soll. Gleichzeitig möchte er aber einen möglichst glatten Streckenverlauf ohne Spitzen.



Ziel

Bestimme eine „glatte“ Funktion $\Phi : \mathbb{R} \to \mathbb{R}^2$, welche durch die Stützstellen verläuft.

Problemstellung: Interpolation

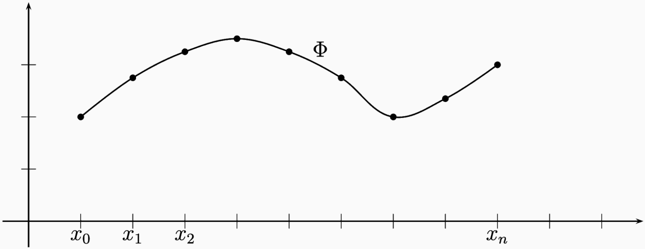
Problemstellung

Seien $n+1$ Punkte $(x_i, y_i) \in \mathbb{R} \times \mathbb{R}$ gegeben, dann ist eine Funktion

$$\Phi : \mathbb{R} \to \mathbb{R} \quad \text{mit} \quad \Phi(x_i) = y_i$$

einer bestimmten „Bauart“ gesucht.

Diese Bauart wird durch die Angabe eines Funktionenraums V_n vorgegeben, man fordert dann zusätzlich noch $\Phi \in V_n$.



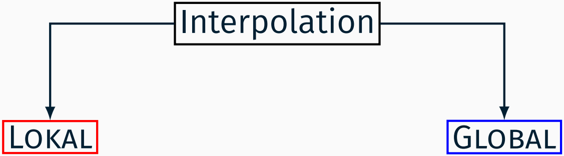
Interpolationsverfahren

Interpolation

Lokal

Global

Interpolationsverfahren

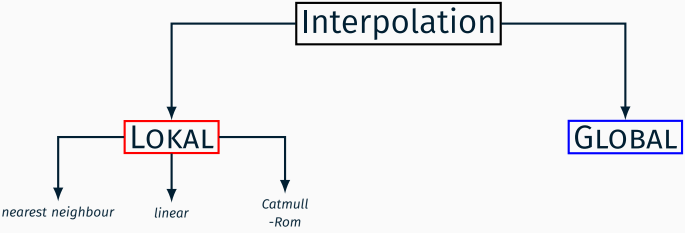


Definition 6.1

Ein Interpolationsverfahren heißt …

… **lokal**, wenn der interpolierte Wert nur von **benachbarten Werten** abhängt.

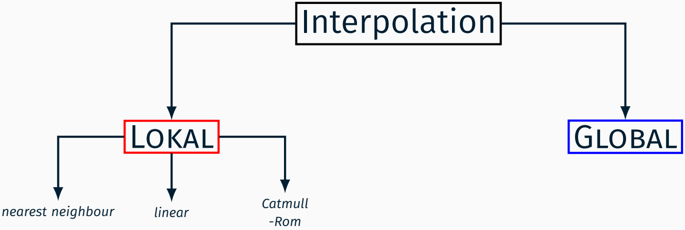
Interpolationsverfahren



Definition 6.1

Ein Interpolationsverfahren heißt ...

... **lokal**, wenn der interpolierte Wert nur von **benachbarten Werten** abhängt.



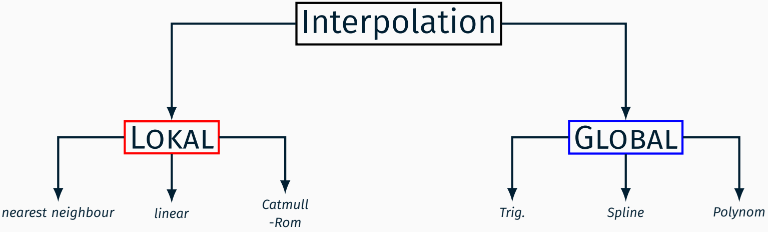
Definition 6.1

Ein Interpolationsverfahren heißt ...

... **lokal**, wenn der interpolierte Wert nur von **benachbarten Werten** abhängt.

... **global**, wenn die Interpolationsfunktion von **allen** Stützstellen abhängt.

Interpolationsverfahren



Definition 6.1

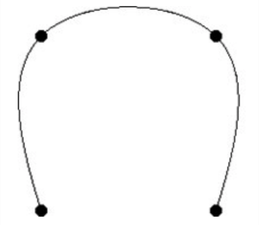
Ein Interpolationsverfahren heißt …

… **lokal**, wenn der interpolierte Wert nur von **benachbarten Werten** abhängt.

… **global**, wenn die Interpolationsfunktion von **allen** Stützstellen abhängt.

Unterschied: Interpolation $\longleftrightarrow$ Approximation

Interpolation



$\longleftrightarrow$

Die Kurve **muss** durch jeden Kontrollpunkt gehen.

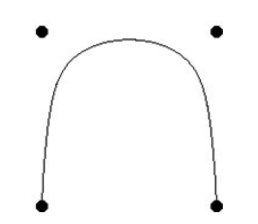
Unterschied: Interpolation $\longleftrightarrow$ Approximation

Interpolation

Die Kurve **muss** durch jeden Kontrollpunkt gehen.

$\longleftrightarrow$

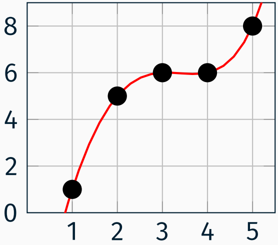
Approximation



Die Kurve wird von den Kontrollpunkten **beeinflusst**!

Unterschied: Interpolation $\longleftrightarrow$ Approximation

Interpolation

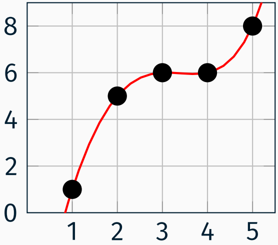


$\longleftrightarrow$

Die Kurve **muss** durch jeden Kontrollpunkt gehen.

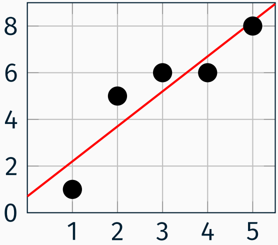
Unterschied: Interpolation $\longleftrightarrow$ Approximation

Interpolation



$\longleftrightarrow$

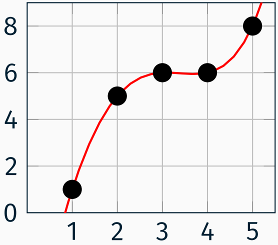
Approximation



Die Kurve **muss** durch jeden Kontrollpunkt gehen.

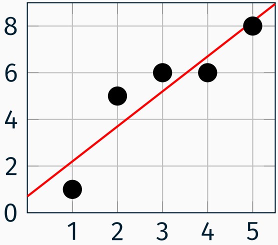
Die Kurve wird von den Kontrollpunkten **beeinflusst**!

Interpolation



⟷

Approximation



Die Kurve **muss** durch jeden Kontrollpunkt gehen.

Die Kurve wird von den Kontrollpunkten **beeinflusst**!

Daumenregel

Interpolation ist bei **exakten, korrekten** Daten sinnvoll, man möchte diese als solche auch behandeln. Kam es allerdings zu Messfehlern, oder sind die Daten nur *Approximationen*, so ist eine *Approximation* sinnvoller

Globale Polynominterpolation (I) — Vandermonde

Wir betrachten die Monombasis

$$\mathbb{P}_n = \operatorname{span}\left\{ x^i \;\middle|\; 0 \leqslant i \leqslant n \right\}$$

des Polynomraums $\mathbb{P}_n$ aller reellen Polynome vom Grad $\leqslant n$.

Globale Polynominterpolation (I) — Vandermonde

Wir betrachten die Monombasis

$$\mathbb{P}_n = \operatorname{span}\left\{ x^i \;\middle|\; 0 \leqslant i \leqslant n \right\}$$

des Polynomraums $\mathbb{P}_n$ aller reellen Polynome vom Grad $\leqslant n$. Die Koeffizienten der Interpolationsfunktion für $n+1$ Stützstellen sind die Lösung des linearen Gleichungssystems

$$\underbrace{\begin{pmatrix} 1 & x_1 & \cdots & x_1^n \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n+1} & \cdots & x_{n+1}^n \end{pmatrix}}_{\textbf{Vandermondematrix } V} \cdot \underbrace{\begin{pmatrix} c_1 \\ \vdots \\ c_{n+1} \end{pmatrix}}_{=:\, c} = \underbrace{\begin{pmatrix} y_1 \\ \vdots \\ y_{n+1} \end{pmatrix}}_{=:\, y} .$$

Globale Polynominterpolation (I) — Vandermonde

Gegeben die Monombasis

$$\mathbb{P}_n = \operatorname{span}\left\{ x^i \;\middle|\; 0 \leqslant i \leqslant n \right\},$$

so lösen wir das LGS

$$V \cdot c = y. \tag{1}$$

Singularität

Sind alle $n+1$ Stützstellen paarweise verschieden, so ist die Vandermonde-matrix V nicht singulär. (Interpolationstheorem aus der Mathematik oder Determinantenformel)

Globale Polynominterpolation (I) — Vandermonde

Gegeben die Monombasis

$$\mathbb{P}_n = \operatorname{span}\left\{ x^i \;\middle|\; 0 \leqslant i \leqslant n \right\},$$

so lösen wir das LGS

$$V \cdot c = y. \tag{1}$$

Problem

Dennoch ist die Matrix voll besetzt und schlecht konditioniert, so ist für $n = 20$ bei äquidistanten Stützstellen die Kondition bereits bei $\kappa_\infty \approx 11000$.

Globale Polynominterpolation (I) — Vandermonde

Gegeben die Monombasis

$$\mathbb{P}_n = \operatorname{span}\left\{ x^i \;\middle|\; 0 \leqslant i \leqslant n \right\},$$

so lösen wir das LGS

$$V \cdot c = y. \tag{1}$$

Singularität

Sind alle $n+1$ Stützstellen paarweise verschieden, so ist die Vandermonde-matrix V nicht singulär. (Interpolationstheorem aus der Mathematik oder Determinantenformel)

Problem

Dennoch ist die Matrix voll besetzt und schlecht konditioniert, so ist für $n = 20$ bei äquidistanten Stützstellen die Kondition bereits bei $\kappa_\infty \approx 11000$.

Satz 6.2 (Determinatenformel für die Vandermondematrix)

Sei mit

$$V_n = \begin{pmatrix} 1 & x_1 & \cdots & x_1^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \cdots & x_n^{n-1} \end{pmatrix} \in \mathbb{R}^{n\times n}$$

die n-dimensionale Vandermondematrix bezeichnet, so berechnet sich deren Determinante durch

$$\det(V_n) = \prod_{1\leqslant i<j\leqslant n} \left(x_j - x_i\right) .$$

Ziel der Lagrangebasis

Wähle Basisfunktionen φ_i, für die die Matrix in Gleichung (1) gut konditioniert ist.

Globale Polynominterpolation (II) — Lagrangebasis

Ziel der Lagrangebasis

Wähle Basisfunktionen φ_i, für die die Matrix in Gleichung (1) gut konditioniert ist. Eine Möglichkeit ist es $V = \mathbb{1}_n$ zu wählen.

Ziel der Lagrangebasis

Wähle Basisfunktionen φ_i, für die die Matrix in Gleichung (1) gut konditioniert ist. Eine Möglichkeit ist es $V = \mathbb{1}_n$ zu wählen.

Es ergeben sich Basisfunktionen für alle $i, j = 1, \dots, n+1$ mit

$$\varphi_i(x_j) = \delta_{ji} = \begin{cases} 1 & \text{, falls } i = j \\ 0 & \text{sonst.} \end{cases} \tag{2}$$

Globale Polynominterpolation (II) — Lagrangebasis

Definition 6.3 (Lagrangebasis)

Eine Basis des Raumes $\mathbb{P}_n$, welche die Bedingung (2) erfüllt, ist die **Lagrangebasis**:

$$\varphi_j(x) = L_j(x) := \frac{\prod\limits_{i=1,\ i\neq j}^{n+1} (x - x_i)}{\prod\limits_{i=1,\ i\neq j}^{n+1} (x_j - x_i)} \qquad \text{für } j = 1, \ldots, n+1$$

Satz 6.4 (Koeffizienten der Lagrangebasis)

Die Koeffizienten des Interpolationspolynoms sind genau die Stützwerte, es gilt also für das Interpolationspolynom:

$$p_L(x) = \sum_{i=1}^{n+1} c_i \cdot L_i(x) = \sum_{i=1}^{n+1} y_i \cdot L_i(x).$$

Globale Polynominterpolation (II) — Lagrangebasis

Lagrangebasis:

$$\varphi_j(x) = L_j(x) := \frac{\prod\limits_{i=1,\ i\neq j}^{n+1} (x - x_i)}{\prod\limits_{i=1,\ i\neq j}^{n+1} (x_j - x_i)} \qquad \text{für } j = 1, \ldots, n+1$$

Probleme

Jede Evaluation des Interpolationspolynoms benötigt $\mathcal{O}\left(n^2\right)$ Operationen, dies kann aber zu $\mathcal{O}\left(n\right)$ abgeschwächt werden.

Bei Hinzufügen eines neuen Punktes (x_{n+2}, y_{n+2}) muss **alles** neu berechnet werden, da **jedes** L_j von *allen* Stützpunkten abhängt.

Definition 6.5 (Newtonbasis)

Eine Basis des Raumes $\mathbb{P}_n$, welche den zweiten Nachteil von Lagrange behebt, ist die **Newtonbasis**:

$$\varphi_j(x) = N_j(x) := \prod_{i=1}^{j-1}(x - x_i)$$

Globale Polynominterpolation (III) — Newtonbasis

Definition 6.5 (Newtonbasis)

Eine Basis des Raumes $\mathbb{P}_n$, welche den zweiten Nachteil von Lagrange behebt, ist die **Newtonbasis**:

$$\varphi_j(x) = N_j(x) := \prod_{i=1}^{j-1}(x - x_i)$$

Wir stellen fest, dass N_j nur von den ersten $j-1$ Stützstellen abhängt. Die Koeffizienten ergeben sich dann als Lösung des linearen Gleichungssystems

$$\begin{pmatrix} 1 & 0 & \cdots & 0 \\ 1 & (x_2 - x_1) & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ 1 & (x_{n+1} - x_1) & \cdots & \prod_{i=1}^{n+1}(x_{n+1} - x_i) \end{pmatrix} \cdot \begin{pmatrix} c_1 \\ \vdots \\ \vdots \\ c_{n+1} \end{pmatrix} = \begin{pmatrix} y_1 \\ \vdots \\ \vdots \\ y_{n+1} \end{pmatrix}.$$

Globale Polynominterpolation (III) — Aitken-Neville

Eine Möglichkeit das Gleichungssystem von Newton zu lösen ist der Algorithmus von Aitken-Neville:

Algorithmus 6.1 — Algorithmus von Aitken-Neville

Eingabe : Stützstellen x_i, Stützwerte y_i mit $1 \leqslant i \leqslant n$.

Ausgabe : Koeffizienten a_i mit $1 \leqslant i \leqslant n$.

// Komplexität von $\mathcal{O}\left(n^2\right)$

(1) $P \leftarrow (p_{ik})_{\substack{1 \leqslant i \leqslant n-k \\ 0 \leqslant k \leqslant n-1}}$

(2) $p_{1:n,0} \leftarrow y_{1:n}$.

(3) **for** $k \leftarrow 1$ **to** n **do**

(4) **for** $i \leftarrow 1$ **to** $n-k$ **do**

(5) $p_{i,k} \leftarrow \dfrac{p_{i+1,k-1} - p_{i,k-1}}{x_{i+k} - x_i}$

(6) **end for**

(7) **end for**

(8) $a \leftarrow p_1$

Aufgabe 1 — Polynominterpolation (I)

Aufgabe 1 — Polynominterpolation (I)

a) Gegeben sind die Stützstellen $x_1 = -2, x_2 = 0, x_3 = 1, x_4 = 4$.
Bestimmen Sie die Lagrange- und Newton-Polynombasen zu diesen Stützstellen.

Aufgabe 1 — Polynominterpolation (I)

a) Gegeben sind die Stützstellen $x_1 = -2, x_2 = 0, x_3 = 1, x_4 = 4$. Bestimmen Sie die Lagrange- und Newton-Polynombasen zu diesen Stützstellen.

b) An diesen Stützstellen sind die Werte $\{4, -2, 1, 4\}$ gegeben. Bestimmen Sie die Koeffizienten des Interpolationspolynoms bzgl. der Lagrange- und Newton-Basis.

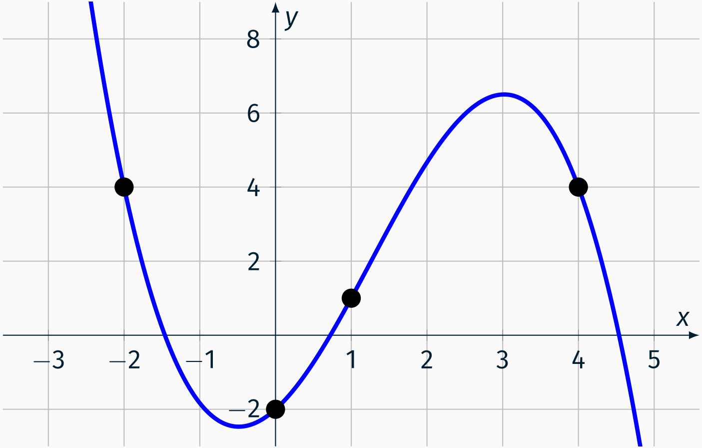
Aufgabe 1 — Polynominterpolation (I)

a) Gegeben sind die Stützstellen $x_1 = -2, x_2 = 0, x_3 = 1, x_4 = 4$. Bestimmen Sie die Lagrange- und Newton-Polynombasen zu diesen Stützstellen.

b) An diesen Stützstellen sind die Werte $\{4, -2, 1, 4\}$ gegeben. Bestimmen Sie die Koeffizienten des Interpolationspolynoms bzgl. der Lagrange- und Newton-Basis.

c) Bestimmen Sie den Wert der Polynome mit den Werten aus Teilaufgabe b an der Stelle $x = 0$.

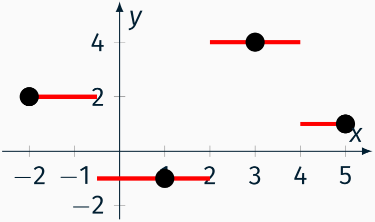
Aufgabe 1 — Polynominterpolation (I): Polynom



Rekapitulation — Lokale Interpolation

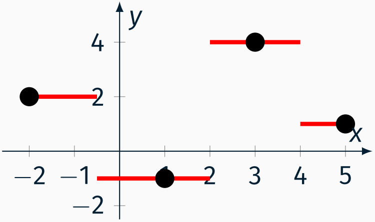
Lokale Interpolation (I) — *nearest neighbour*

Um den Wert an der Stelle x anzunähern, suche den nähesten Nachbarn, sprich die nächst gelegene Stützstelle x_j. Der interpolierte Wert ist dann mit y_j gegeben.



Lokale Interpolation (I) — *nearest neighbour*

Um den Wert an der Stelle x anzunähern, suche den nähesten Nachbarn, sprich die nächst gelegene Stützstelle x_j. Der interpolierte Wert ist dann mit y_j gegeben.

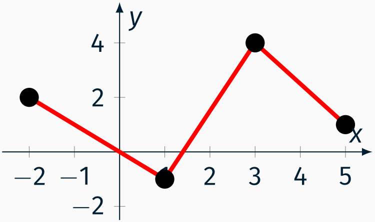


Für das Polynom gilt dann:

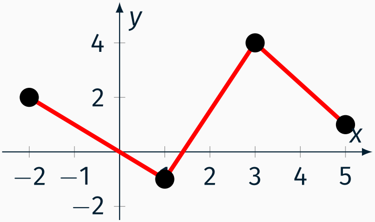
$$p(x) = \begin{cases} y_1 & \text{für } x_1 \leqslant x \leqslant \frac{1}{2}\left(x_1 + x_2\right) \\ y_i & \text{für } \frac{1}{2}\left(x_{i-1} + x_i\right) < x \leqslant \frac{1}{2}\left(x_i + x_{i+1}\right) \quad \text{für alle } 2 \leqslant i \leqslant n-1 \\ y_n & \text{für } \frac{1}{2}\left(x_{n-1} + x_n\right) < x \leqslant x_n \end{cases}$$

Lokale Interpolation (II) — Stückweise linear

Um den Wert an der Stelle x hierbei anzunähern, suchen wir zuerst die *nächst gelegene* linke und rechte Stützstelle x_i und x_{i+1}, so dass $x_i \leqslant x \leqslant x_{i+1}$ gilt, und interpolieren dann im Intervall $[x_i, x_{i+1}]$ linear.



Um den Wert an der Stelle x hierbei anzunähern, suchen wir zuerst die *nächst gelegene* linke und rechte Stützstelle x_i und x_{i+1}, so dass $x_i \leqslant x \leqslant x_{i+1}$ gilt, und interpolieren dann im Intervall $[x_i, x_{i+1}]$ linear.



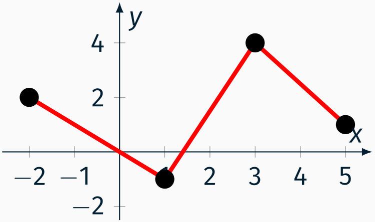
Für $1 \leqslant i < n$ stelle die Polynome

$$p_i(x) = m_i \cdot (x - x_i) + y_i = \underbrace{\frac{y_{i+1} - y_i}{x_{i+1} - x_i}}_{=:m_i} \cdot (x - x_i) + y_i$$

auf.

Lokale Interpolation (II) — Stückweise linear

Um den Wert an der Stelle x hierbei anzunähern, suchen wir zuerst die *nächst gelegene* linke und rechte Stützstelle x_i und x_{i+1}, so dass $x_i \leqslant x \leqslant x_{i+1}$ gilt, und interpolieren dann im Intervall $[x_i, x_{i+1}]$ linear.



Dann gilt insgesamt:

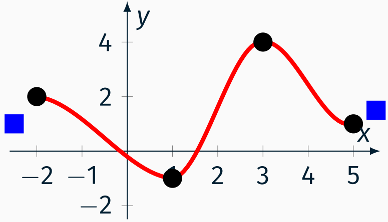
$$p(x) = \left(\sum_{i=1}^{n-2} p_i(x) \cdot \mathbb{1}_{[x_i, x_{i+1})}(x) \right) + p_n(x) \cdot \mathbb{1}_{[x_{n-1}, x_n]}(x).$$

Lokale Interpolation (III) — Catmull-Rom

Wir finden dann auf jedem Teilintervall $T_i := [x_i, x_{i+1}]$ das (*eindeutige*) kubische Polynom, welches in den Endpunkten neben den Stützwerten auch noch die geschätzten Ableitungen interpoliert. Ableitungen werden durch

$$y_i' = \frac{y_{i+1} - y_{i-1}}{x_{i+1} - x_{i-1}}$$

abgeschätzt.



Lokale Interpolation (III) — Catmull-Rom

Man kann die kubischen Polynome auch explizit bestimmen durch die Funktionen

$$p_i(x) = a_0 \cdot (x_{i+1} - x)^3 + a_1 \cdot (x_{i+1} - x)^2 \cdot (x - x_i) + a_2 \cdot (x_{i+1} - x) \cdot (x - x_i)^2 + a_3 \cdot (x - x_i)^3,$$

wobei sich die Koeffizienten durch

$$a_0 = \frac{y_i}{(x_{i+1} - x_i)^3}, \quad a_1 = 3 \cdot a_0 + \frac{y_i'}{(x_{i+1} - x_i)^2},$$

$$a_2 = 3 \cdot a_3 - \frac{y_{i+1}'}{(x_{i+1} - x_i)^2} \quad \text{und} \quad a_3 = \frac{y_{i+1}}{(x_{i+1} - x_i)^3}$$

bestimmen lassen.

Lokale Interpolation (IV) — Fehlerabschätzungen

Sei h gegeben durch

$$h = \max_{1 \leqslant i \leqslant n} \left\{ x_{i+1} - x_i \right\}.$$

Dann kann man zeigen:

Verfahrensname	*obere Fehlerschranke*	*„Fehlerkomplexität“*
Konstante Interpolation (*nearest neighbor*)	$\leqslant \frac{\lvert f'(\xi) \rvert}{2 \cdot h}$	$\in \mathcal{O}(h)$
Lineare Interpolation	$\leqslant \frac{\lvert f''(\xi) \rvert}{8 \cdot h^2}$	$\in \mathcal{O}(h^2)$
Kubische Interpolation (*catmull rom*)	$\leqslant \frac{\lvert f'''(\xi) \rvert}{24 \cdot h^3}$	$\in \mathcal{O}(h^3)$

Aufgabe 1 — Polynominterpolation (II)

Aufgabe 1 — Polynominterpolation (II)

d) Nun sind an den Stützstellen $\{-3, -1, 1, 3\}$ die Werte $\{-1, 1, 2, 2\}$ gegeben. Skizzieren Sie das mittlere Segment des Catmull-Rom-Interpolanten.

Rekapitulation — Freiformmodellierung mit Bézierkurven

Definition 6.6 (Bernsteinpolynombasen)

Die **Bernsteinpolynome** B_i^n sind Polynome mit

$$B_i^n(x) := \binom{n}{i} \cdot (1-x)^{n-i} \cdot x^i$$

für $0 \leqslant i \leqslant n$. Sie spannen ebenfalls den Polynomraum $\mathbb{P}_n$ auf.

Definition 6.6 (Bernsteinpolynombasen)

Die **Bernsteinpolynome** B_i^n sind Polynome mit

$$B_i^n(x) := \binom{n}{i} \cdot (1-x)^{n-i} \cdot x^i$$

für $0 \leqslant i \leqslant n$. Sie spannen ebenfalls den Polynomraum $\mathbb{P}_n$ auf.

Satz 6.7 (Eigenschaften der Bernsteinpolynome)

- Für $t \in [0,1]$ gilt, dass $0 \leqslant B_i^n(t) \leqslant 1$

Bernsteinpolynombasen

Definition 6.6 (Bernsteinpolynombasen)

Die **Bernsteinpolynome** B_i^n sind Polynome mit

$$B_i^n(x) := \binom{n}{i} \cdot (1-x)^{n-i} \cdot x^i$$

für $0 \leqslant i \leqslant n$. Sie spannen ebenfalls den Polynomraum $\mathbb{P}_n$ auf.

Satz 6.7 (Eigenschaften der Bernsteinpolynome)

- Für $t \in [0,1]$ gilt, dass $0 \leqslant B_i^n(t) \leqslant 1$
- $B_i^n(t)$ hat eine $i-$fache Nullstelle in $t = 0$ und $(n-i)$-fache in $t = 1$.

Bernsteinpolynombasen

Definition 6.6 (Bernsteinpolynombasen)

Die **Bernsteinpolynome** B_i^n sind Polynome mit

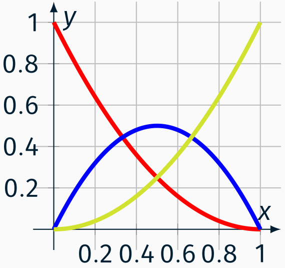
$$B_i^n(x) := \binom{n}{i} \cdot (1-x)^{n-i} \cdot x^i$$

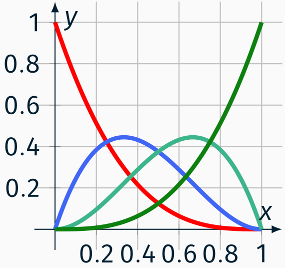
für $0 \leqslant i \leqslant n$. Sie spannen ebenfalls den Polynomraum $\mathbb{P}_n$ auf.

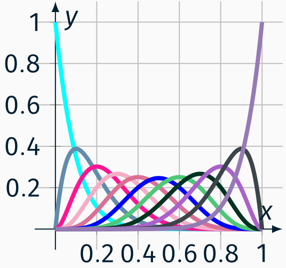
Satz 6.7 (Eigenschaften der Bernsteinpolynome)

- Für $t \in [0,1]$ gilt, dass $0 \leqslant B_i^n(t) \leqslant 1$
- $B_i^n(t)$ hat eine i–fache Nullstelle in $t = 0$ und $(n-i)$-fache in $t = 1$.
- Für alle t gilt, dass $\sum_{i=0}^n B_i^n(t) = 1$

Bernsteinpolynombasen







Bézierkurven

Ziel

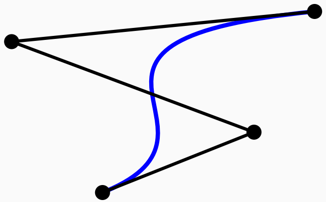
Kleine Änderungen an **einzelnen** Stützwerten, sollen nur *geringe* **lokale** Auswirkungen haben.

Definition 6.8 (Bézierkurve)

Gegeben eine Menge an Kontrollpunkten $b_i \in \mathbb{R}^d$ mit $0 \leqslant i \leqslant n$, nennen wir die Kurve $C \subseteq \mathbb{R}^d$ Bézierkurve vom Grad n, wenn

$$C(t) = \sum_{i=0}^{n} b_i \cdot B_i^n(t) \qquad \text{mit } t \in [0, 1].$$

Der Polygonzug aus den Kontrollpunkten bezeichnet man als das zur Bézierkurve gehöhrende **Kontrollpolygon**.



(1) **Interpolation der Endpunkte** Es muss gelten, dass

$$C(0) = b_0 \qquad \text{und} \qquad C(1) = b_n.$$

(2) **Tangentiale Tangentenbedingung** In den Endpunkten nähert sich die Kurve tangential an das Kontrollpolygon an, es gilt also

$$C'(0) = n(b_1 - b_0) \quad \text{und } C'(1) = n(b_n - b_{n-1})$$

(3) **Konvexe Hülle** Die Bézierkurve liegt in der **konvexen Hülle** der Kontrollpunkte. Die **konvexe Hülle** einer Menge M ist die kleinste Obermenge M' von M, welche ebenfalls konvex ist.

(4) **Affine Invarianz** Eine affine Abbildung ist mit $\Phi(x) = Ax + b$ gegeben, um eine Bézierkurve affin zu transformieren müssen **bloß die Kontrollpunkte affin transformiert** werden.

Bézierkurven — Eigenschaften

(5) **Variationsreduzierend** Für jede Gerade g gilt, dass die Anzahl an Schnittpunkten von g mit der Kurve kleiner oder gleich der Anzahl an Schnittpunkten von g mit dem Kontrollpolygon ist.

Bézierkurven — Eigenschaften

(1) **Interpolation der Endpunkte** Es muss gelten, dass

$$C(0) = b_0 \qquad \text{und} \qquad C(1) = b_n.$$

(2) **Tangentiale Tangentenbedingung** In den Endpunkten nähert sich die Kurve tangential an das Kontrollpolygon an, es gilt also

$$C'(0) = n(b_1 - b_0) \quad \text{und } C'(1) = n(b_n - b_{n-1})$$

(3) **Konvexe Hülle** Die Bézierkurve liegt in der **konvexen Hülle** der Kontrollpunkte. Die **konvexe Hülle** einer Menge M ist die kleinste Obermenge M' von M, welche ebenfalls konvex ist.

(4) **Affine Invarianz** Eine affine Abbildung ist mit $\Phi(x) = Ax + b$ gegeben, um eine Bézierkurve affin zu transformieren müssen **bloß die Kontrollpunkte affin transformiert** werden.

(5) **Variationsreduzierend** Für jede Gerade g gilt, dass die Anzahl an Schnittpunkten von g mit der Kurve kleiner oder gleich der Anzahl an Schnittpunkten von g mit dem Kontrollpolygon ist.

Algorithmus 6.2 — Auswertungsalgorithmus nach DE CASTELJAU

Eingabe : Kontrollpunkte b_i mit $0 \leqslant i \leqslant n$ und Stelle $t \in [0, 1]$.

Ausgabe : Wert $C(t)$

(1) $\mathfrak{B} \leftarrow (\mathfrak{b}_{ik})_{\substack{0 \leqslant i \leqslant n \\ 0 \leqslant k \leqslant n}}$

(2) $\mathfrak{b}^0_{0:n} \leftarrow b_{0:n}$

(3) **for** $k \leftarrow 1$ **to** n **do**

(4)   **for** $i \leftarrow k$ **to** n **do**

(5) $\mathfrak{b}^k_i \leftarrow (1 - t) \cdot \mathfrak{b}^{k-1}_{i-1} + t \cdot \mathfrak{b}^{k-1}_i$

(6)   **end for**

(7) **end for**

(8) **return** $\mathfrak{b}^n_n$

Bézierkurven — Auswertung über de Casteljau (II)

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:

$$\begin{array}{cccc}
\begin{pmatrix} \\ \\ \end{pmatrix} & & & \\
\begin{pmatrix} \\ \\ \end{pmatrix} & \begin{pmatrix} \\ \\ \end{pmatrix} & & \\
\begin{pmatrix} \\ \\ \end{pmatrix} & \begin{pmatrix} \\ \\ \end{pmatrix} & \begin{pmatrix} \\ \\ \end{pmatrix} & \\
\begin{pmatrix} \\ \\ \end{pmatrix} & \begin{pmatrix} \\ \\ \end{pmatrix} & \begin{pmatrix} \\ \\ \end{pmatrix} & \begin{pmatrix} \\ \\ \end{pmatrix}
\end{array}$$

$\mathfrak{B} \leftarrow (\mathfrak{b}_{ik})_{\substack{0 \leqslant i \leqslant n \\ 0 \leqslant k \leqslant n}}$

$\mathfrak{b}^0_{0:n} \leftarrow b_{0:n}$

for $k \leftarrow 1$ **to** n **do**

  for $i \leftarrow k$ **to** n **do**

    $\mathfrak{b}^k_i \leftarrow (1-t) \cdot \mathfrak{b}^{k-1}_{i-1} + t \cdot \mathfrak{b}^{k-1}_i$

  end for

end for

return $\mathfrak{b}^n_n$

Bézierkurven — Auswertung über DE CASTELJAU (II)

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:

$$\begin{array}{cccc}
\begin{pmatrix} 0 \\ 9 \end{pmatrix} & & & \\
\begin{pmatrix} 0 \\ 0 \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix} & & \\
\begin{pmatrix} 9 \\ 0 \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix} & \\
\begin{pmatrix} 12 \\ 6 \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix}
\end{array}$$

$\mathfrak{B} \leftarrow (\mathfrak{b}_{ik})_{\substack{0 \leqslant i \leqslant n \\ 0 \leqslant k \leqslant n}}$

$\mathfrak{b}^0_{0:n} \leftarrow b_{0:n}$

for $k \leftarrow 1$ **to** n **do**

  for $i \leftarrow k$ **to** n **do**

    $\mathfrak{b}^k_i \leftarrow (1-t) \cdot \mathfrak{b}^{k-1}_{i-1} + t \cdot \mathfrak{b}^{k-1}_i$

  end for

end for

return $\mathfrak{b}^n_n$

Bézierkurven — Auswertung über de Casteljau (II)

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:

$$\begin{array}{cccc}
\begin{pmatrix} 0 \\ 9 \end{pmatrix} & & & \\
\begin{pmatrix} 0 \\ 0 \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix} & & \\
\begin{pmatrix} 9 \\ 0 \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix} & \\
\begin{pmatrix} 12 \\ 6 \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix}
\end{array}$$

```
𝔅 ← (𝔟_ik) 0⩽i⩽n, 0⩽k⩽n
𝔟^0_{0:n} ← b_{0:n}
for k ← 1 to n do
    for i ← k to n do
        𝔟^k_i ← (1−t)·𝔟^{k−1}_{i−1} + t·𝔟^{k−1}_i
    end for
end for
return 𝔟^n_n
```

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:

$$\begin{array}{cccc}
\begin{pmatrix} 0 \\ 9 \end{pmatrix} & & & \\
\begin{pmatrix} 0 \\ 0 \end{pmatrix} & \begin{pmatrix} \ \\ \ \end{pmatrix} & & \\
\begin{pmatrix} 9 \\ 0 \end{pmatrix} & \begin{pmatrix} \ \\ \ \end{pmatrix} & \begin{pmatrix} \ \\ \ \end{pmatrix} & \\
\begin{pmatrix} 12 \\ 6 \end{pmatrix} & \begin{pmatrix} \ \\ \ \end{pmatrix} & \begin{pmatrix} \ \\ \ \end{pmatrix} & \begin{pmatrix} \ \\ \ \end{pmatrix}
\end{array}$$

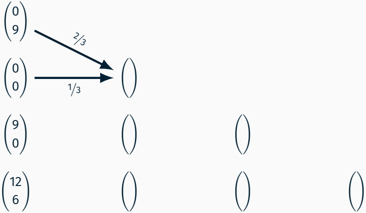
(1) $\mathfrak{B} \leftarrow (\mathfrak{b}_{ik})_{\substack{0 \leqslant i \leqslant n \\ 0 \leqslant k \leqslant n}}$
(2) $\mathfrak{b}^0_{0:n} \leftarrow b_{0:n}$
(3) **for** $k \leftarrow 1$ **to** n **do** `// k = 1`
(4) **for** $i \leftarrow k$ **to** n **do**
(5) $\mathfrak{b}^k_i \leftarrow (1-t) \cdot \mathfrak{b}^{k-1}_{i-1} + t \cdot \mathfrak{b}^{k-1}_i$
(6) **end for**
(7) **end for**
(8) **return** $\mathfrak{b}^n_n$

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:



$\mathfrak{B} \leftarrow (\mathfrak{b}_{ik})_{\substack{0 \leqslant i \leqslant n \\ 0 \leqslant k \leqslant n}}$

$\mathfrak{b}^0_{0:n} \leftarrow b_{0:n}$

for $k \leftarrow 1$ **to** n **do** `// k = 1`

  for $i \leftarrow k$ **to** n **do** `// i = 1`

    $\mathfrak{b}^k_i \leftarrow (1-t) \cdot \mathfrak{b}^{k-1}_{i-1} + t \cdot \mathfrak{b}^{k-1}_i$

  end for

end for

return $\mathfrak{b}^n_n$

Bézierkurven — Auswertung über DE CASTELJAU (II)

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:

$$\begin{array}{cccc}
\begin{pmatrix} 0 \\ 9 \end{pmatrix} & & & \\
\begin{pmatrix} 0 \\ 0 \end{pmatrix} & \begin{pmatrix} 0 \\ 6 \end{pmatrix} & & \\
\begin{pmatrix} 9 \\ 0 \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix} & \\
\begin{pmatrix} 12 \\ 6 \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix}
\end{array}$$

$\mathfrak{B} \leftarrow (\mathfrak{b}_{ik})_{\substack{0 \leqslant i \leqslant n \\ 0 \leqslant k \leqslant n}}$

$\mathfrak{b}_{0:n}^0 \leftarrow b_{0:n}$

for $k \leftarrow 1$ **to** n **do** // k = 1

  for $i \leftarrow k$ **to** n **do**

    $\mathfrak{b}_i^k \leftarrow (1-t) \cdot \mathfrak{b}_{i-1}^{k-1} + t \cdot \mathfrak{b}_i^{k-1}$

  end for

end for

return $\mathfrak{b}_n^n$

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:

$$\begin{tikzcd}
\begin{pmatrix} 0 \\ 9 \end{pmatrix} \arrow[dr] & & & \\
\begin{pmatrix} 0 \\ 0 \end{pmatrix} \arrow[r] \arrow[dr, "2/3"'] & \begin{pmatrix} 0 \\ 6 \end{pmatrix} & & \\
\begin{pmatrix} 9 \\ 0 \end{pmatrix} \arrow[r, "1/3"'] & \left(\right) & \left(\right) & \\
\begin{pmatrix} 12 \\ 6 \end{pmatrix} & \left(\right) & \left(\right) & \left(\right)
\end{tikzcd}$$

$\mathfrak{B} \leftarrow (\mathfrak{b}_{ik})_{\substack{0\leqslant i\leqslant n \\ 0\leqslant k\leqslant n}}$

$\mathfrak{b}^0_{0:n} \leftarrow b_{0:n}$

for $k \leftarrow 1$ **to** n **do** `// k = 1`

  for $i \leftarrow k$ **to** n **do** `// i = 2`

    $\mathfrak{b}^k_i \leftarrow (1-t)\cdot\mathfrak{b}^{k-1}_{i-1} + t\cdot\mathfrak{b}^{k-1}_i$

  end for

end for

return $\mathfrak{b}^n_n$

Bézierkurven — Auswertung über DE CASTELJAU (II)

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:

$$\begin{tikzcd}
\begin{pmatrix} 0 \\ 9 \end{pmatrix} \arrow[dr] & & & \\
\begin{pmatrix} 0 \\ 0 \end{pmatrix} \arrow[r] \arrow[dr] & \begin{pmatrix} 0 \\ 6 \end{pmatrix} & & \\
\begin{pmatrix} 9 \\ 0 \end{pmatrix} \arrow[r] & \begin{pmatrix} 3 \\ 0 \end{pmatrix} & \begin{pmatrix} \ \\ \ \end{pmatrix} & \\
\begin{pmatrix} 12 \\ 6 \end{pmatrix} & \begin{pmatrix} \ \\ \ \end{pmatrix} & \begin{pmatrix} \ \\ \ \end{pmatrix} & \begin{pmatrix} \ \\ \ \end{pmatrix}
\end{tikzcd}$$

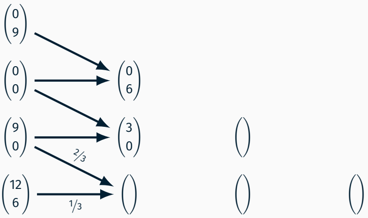
$\mathfrak{B} \leftarrow (\mathfrak{b}_{ik})_{\substack{0 \leqslant i \leqslant n \\ 0 \leqslant k \leqslant n}}$
$\mathfrak{b}^0_{0:n} \leftarrow b_{0:n}$
for $k \leftarrow 1$ **to** n **do**   `// k = 1`
    for $i \leftarrow k$ **to** n **do**
        $\mathfrak{b}^k_i \leftarrow (1-t) \cdot \mathfrak{b}^{k-1}_{i-1} + t \cdot \mathfrak{b}^{k-1}_i$
    end for
end for
return $\mathfrak{b}^n_n$

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:



$\mathfrak{B} \leftarrow (\mathfrak{b}_{ik})_{\substack{0 \leqslant i \leqslant n \\ 0 \leqslant k \leqslant n}}$
$\mathfrak{b}^0_{0:n} \leftarrow b_{0:n}$
for $k \leftarrow 1$ **to** n **do** // k = 1
  for $i \leftarrow k$ **to** n **do** // i = 3
    $\mathfrak{b}^k_i \leftarrow (1-t) \cdot \mathfrak{b}^{k-1}_{i-1} + t \cdot \mathfrak{b}^{k-1}_i$
  end for
end for
return $\mathfrak{b}^n_n$

Bézierkurven — Auswertung über de Casteljau (II)

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:

$$\begin{array}{ccccccc}
\begin{pmatrix} 0 \\ 9 \end{pmatrix} & & & & & & \\
 & \searrow & & & & & \\
\begin{pmatrix} 0 \\ 0 \end{pmatrix} & \rightarrow & \begin{pmatrix} 0 \\ 6 \end{pmatrix} & & & & \\
 & \searrow & & & & & \\
\begin{pmatrix} 9 \\ 0 \end{pmatrix} & \rightarrow & \begin{pmatrix} 3 \\ 0 \end{pmatrix} & & \begin{pmatrix} \\ \end{pmatrix} & & \\
 & \searrow & & & & & \\
\begin{pmatrix} 12 \\ 6 \end{pmatrix} & \rightarrow & \begin{pmatrix} 10 \\ 2 \end{pmatrix} & & \begin{pmatrix} \\ \end{pmatrix} & & \begin{pmatrix} \\ \end{pmatrix}
\end{array}$$

$\mathfrak{B} \leftarrow (\mathfrak{b}_{ik})_{\substack{0 \leqslant i \leqslant n \\ 0 \leqslant k \leqslant n}}$
$\mathfrak{b}^0_{0:n} \leftarrow b_{0:n}$
for $k \leftarrow 1$ **to** n **do**
  for $i \leftarrow k$ **to** n **do**
    $\mathfrak{b}^k_i \leftarrow (1-t) \cdot \mathfrak{b}^{k-1}_{i-1} + t \cdot \mathfrak{b}^{k-1}_i$
  end for
end for
return $\mathfrak{b}^n_n$

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:

$$\begin{tikzcd}
\begin{pmatrix} 0 \\ 9 \end{pmatrix} \arrow[dr] & & & \\
\begin{pmatrix} 0 \\ 0 \end{pmatrix} \arrow[r] \arrow[dr] & \begin{pmatrix} 0 \\ 6 \end{pmatrix} & & \\
\begin{pmatrix} 9 \\ 0 \end{pmatrix} \arrow[r] \arrow[dr] & \begin{pmatrix} 3 \\ 0 \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix} & \\
\begin{pmatrix} 12 \\ 6 \end{pmatrix} \arrow[r] & \begin{pmatrix} 10 \\ 2 \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix}
\end{tikzcd}$$

```
𝔅 ← (𝔟_ik) 0≤i≤n, 0≤k≤n
𝔟^0_{0:n} ← b_{0:n}
for k ← 1 to n do                         // k = 2
    for i ← k to n do
        𝔟^k_i ← (1−t)·𝔟^{k−1}_{i−1} + t·𝔟^{k−1}_i
    end for
end for
return 𝔟^n_n
```

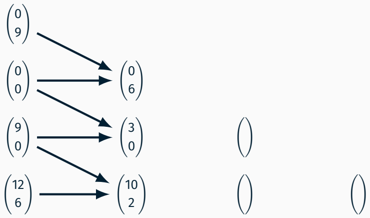
Bézierkurven — Auswertung über DE CASTELJAU (II)

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:




```
𝔅 ← (𝔟_ik)_{0≤i≤n, 0≤k≤n}
𝔟^0_{0:n} ← b_{0:n}
for k ← 1 to n do                                  // k = 2
    for i ← k to n do
        𝔟^k_i ← (1−t)·𝔟^{k−1}_{i−1} + t·𝔟^{k−1}_i
    end for
end for
return 𝔟^n_n
```

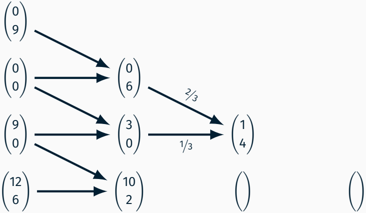
Bézierkurven — Auswertung über DE CASTELJAU (II)

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:



$\mathfrak{B} \leftarrow (\mathfrak{b}_{ik})_{\substack{0 \leqslant i \leqslant n \\ 0 \leqslant k \leqslant n}}$

$\mathfrak{b}^0_{0:n} \leftarrow b_{0:n}$

for $k \leftarrow 1$ **to** n **do** `// k = 2`

  for $i \leftarrow k$ **to** n **do** `// i = 2`

    $\mathfrak{b}^k_i \leftarrow (1-t) \cdot \mathfrak{b}^{k-1}_{i-1} + t \cdot \mathfrak{b}^{k-1}_i$

  end for

end for

return $\mathfrak{b}^n_n$

Bézierkurven — Auswertung über de Casteljau (II)

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:

$$\begin{tikzcd}
\begin{pmatrix} 0 \\ 9 \end{pmatrix} \arrow[dr] & & & \\
\begin{pmatrix} 0 \\ 0 \end{pmatrix} \arrow[r] \arrow[dr] & \begin{pmatrix} 0 \\ 6 \end{pmatrix} \arrow[dr] & & \\
\begin{pmatrix} 9 \\ 0 \end{pmatrix} \arrow[r] \arrow[dr] & \begin{pmatrix} 3 \\ 0 \end{pmatrix} \arrow[r] & \begin{pmatrix} 1 \\ 4 \end{pmatrix} & \\
\begin{pmatrix} 12 \\ 6 \end{pmatrix} \arrow[r] & \begin{pmatrix} 10 \\ 2 \end{pmatrix} & \begin{pmatrix} \ \end{pmatrix} & \begin{pmatrix} \ \end{pmatrix}
\end{tikzcd}$$

$\mathfrak{B} \leftarrow (\mathfrak{b}_{ik})_{\substack{0 \leqslant i \leqslant n \\ 0 \leqslant k \leqslant n}}$

$\mathfrak{b}^0_{0:n} \leftarrow b_{0:n}$

for $k \leftarrow 1$ **to** n **do** // k = 2

  for $i \leftarrow k$ **to** n **do**

    $\mathfrak{b}^k_i \leftarrow (1-t) \cdot \mathfrak{b}^{k-1}_{i-1} + t \cdot \mathfrak{b}^{k-1}_i$

  end for

end for

return $\mathfrak{b}^n_n$

Bézierkurven — Auswertung über de Casteljau (II)

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:

$$\begin{tikzcd}
\begin{pmatrix} 0 \\ 9 \end{pmatrix} \arrow[dr] & & & \\
\begin{pmatrix} 0 \\ 0 \end{pmatrix} \arrow[r] \arrow[dr] & \begin{pmatrix} 0 \\ 6 \end{pmatrix} \arrow[dr] & & \\
\begin{pmatrix} 9 \\ 0 \end{pmatrix} \arrow[r] \arrow[dr] & \begin{pmatrix} 3 \\ 0 \end{pmatrix} \arrow[r] \arrow[dr, "2/3"'] & \begin{pmatrix} 1 \\ 4 \end{pmatrix} & \\
\begin{pmatrix} 12 \\ 6 \end{pmatrix} \arrow[r] & \begin{pmatrix} 10 \\ 2 \end{pmatrix} \arrow[r, "1/3"'] & \begin{pmatrix} 5\,1/3 \\ 2/3 \end{pmatrix} & \begin{pmatrix} \; \\ \; \end{pmatrix}
\end{tikzcd}$$

$\mathfrak{B} \leftarrow (\mathfrak{b}_{ik})_{\substack{0 \leqslant i \leqslant n \\ 0 \leqslant k \leqslant n}}$
$\mathfrak{b}^0_{0:n} \leftarrow b_{0:n}$
for $k \leftarrow 1$ **to** n **do** `// k = 2`
  for $i \leftarrow k$ **to** n **do** `// i = 3`
    $\mathfrak{b}^k_i \leftarrow (1-t) \cdot \mathfrak{b}^{k-1}_{i-1} + t \cdot \mathfrak{b}^{k-1}_i$
  end for
end for
return $\mathfrak{b}^n_n$

Bézierkurven — Auswertung über DE CASTELJAU (II)

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:

$$\begin{tikzcd}
\begin{pmatrix} 0 \\ 9 \end{pmatrix} \arrow[dr] & & & \\
\begin{pmatrix} 0 \\ 0 \end{pmatrix} \arrow[r] \arrow[dr] & \begin{pmatrix} 0 \\ 6 \end{pmatrix} \arrow[dr] & & \\
\begin{pmatrix} 9 \\ 0 \end{pmatrix} \arrow[r] \arrow[dr] & \begin{pmatrix} 3 \\ 0 \end{pmatrix} \arrow[r] \arrow[dr] & \begin{pmatrix} 1 \\ 4 \end{pmatrix} & \\
\begin{pmatrix} 12 \\ 6 \end{pmatrix} \arrow[r] & \begin{pmatrix} 10 \\ 2 \end{pmatrix} \arrow[r] & \begin{pmatrix} 5\,1/3 \\ 2/3 \end{pmatrix} & \begin{pmatrix} \; \\ \; \end{pmatrix}
\end{tikzcd}$$

$\mathfrak{B} \leftarrow (\mathfrak{b}_{ik})_{\substack{0 \leqslant i \leqslant n \\ 0 \leqslant k \leqslant n}}$

$\mathfrak{b}^0_{0:n} \leftarrow b_{0:n}$

for $k \leftarrow 1$ **to** n **do**

  for $i \leftarrow k$ **to** n **do**

    $\mathfrak{b}^k_i \leftarrow (1-t) \cdot \mathfrak{b}^{k-1}_{i-1} + t \cdot \mathfrak{b}^{k-1}_i$

  end for

end for

return $\mathfrak{b}^n_n$

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:

$$\begin{tikzcd}
\begin{pmatrix} 0 \\ 9 \end{pmatrix} \arrow[dr] & & & \\
\begin{pmatrix} 0 \\ 0 \end{pmatrix} \arrow[r] \arrow[dr] & \begin{pmatrix} 0 \\ 6 \end{pmatrix} \arrow[dr] & & \\
\begin{pmatrix} 9 \\ 0 \end{pmatrix} \arrow[r] \arrow[dr] & \begin{pmatrix} 3 \\ 0 \end{pmatrix} \arrow[r] \arrow[dr] & \begin{pmatrix} 1 \\ 4 \end{pmatrix} & \\
\begin{pmatrix} 12 \\ 6 \end{pmatrix} \arrow[r] & \begin{pmatrix} 10 \\ 2 \end{pmatrix} \arrow[r] & \begin{pmatrix} 5\,1/3 \\ 2/3 \end{pmatrix} & \begin{pmatrix} \\ \end{pmatrix}
\end{tikzcd}$$

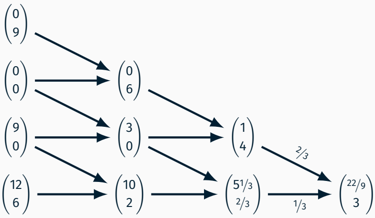
```
𝔅 ← (𝔟_ik)_{0⩽i⩽n, 0⩽k⩽n}
𝔟^0_{0:n} ← b_{0:n}
for k ← 1 to n do                           // k = 3
    for i ← k to n do
        𝔟^k_i ← (1−t)·𝔟^{k−1}_{i−1} + t·𝔟^{k−1}_i
    end for
end for
return 𝔟^n_n
```

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:



$\mathfrak{B} \leftarrow (\mathfrak{b}_{ik})_{\substack{0 \leqslant i \leqslant n \\ 0 \leqslant k \leqslant n}}$

$\mathfrak{b}^0_{0:n} \leftarrow b_{0:n}$

for $k \leftarrow 1$ **to** n **do** // k = 3

  for $i \leftarrow k$ **to** n **do** // i = 3

    $\mathfrak{b}^k_i \leftarrow (1-t) \cdot \mathfrak{b}^{k-1}_{i-1} + t \cdot \mathfrak{b}^{k-1}_i$

  end for

end for

return $\mathfrak{b}^n_n$

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 12 \\ 6 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Rechnerische Lösung:

$$\begin{tikzcd}
\binom{0}{9} \arrow[dr] & & & \\
\binom{0}{0} \arrow[r] \arrow[dr] & \binom{0}{6} \arrow[dr] & & \\
\binom{9}{0} \arrow[r] \arrow[dr] & \binom{3}{0} \arrow[r] \arrow[dr] & \binom{1}{4} \arrow[dr] & \\
\binom{12}{6} \arrow[r] & \binom{10}{2} \arrow[r] & \binom{5\,1/3}{2/3} \arrow[r] & \binom{22/9}{3}
\end{tikzcd}$$

Damit ist $C(1/3)$ durch $(22/9\ ,\ 3)^T$ gegeben.

(1) $\mathfrak{B} \leftarrow (\mathfrak{b}_{ik})_{\substack{0 \leqslant i \leqslant n \\ 0 \leqslant k \leqslant n}}$
(2) $\mathfrak{b}^0_{0:n} \leftarrow b_{0:n}$
(3) **for** $k \leftarrow 1$ **to** n **do**
(4) **for** $i \leftarrow k$ **to** n **do**
(5) $\mathfrak{b}^k_i \leftarrow (1-t) \cdot \mathfrak{b}^{k-1}_{i-1} + t \cdot \mathfrak{b}^{k-1}_i$
(6) **end for**
(7) **end for**
(8) **return** $\mathfrak{b}^n_n$

Bézierkurven — Auswertung über de Casteljau (II)

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 11 \\ 7 \end{pmatrix}$$

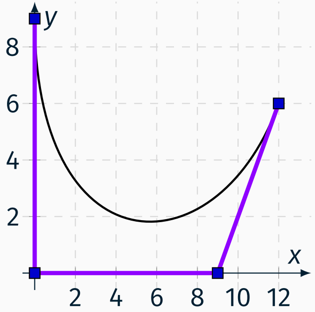
gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 11 \\ 7 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Graphische Lösung:

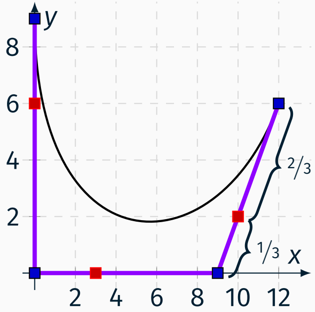


Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 11 \\ 7 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Graphische Lösung:



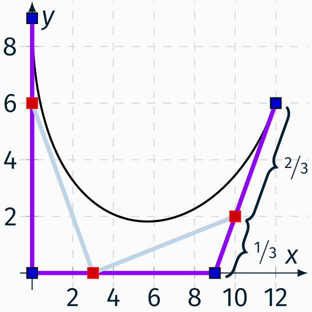
Bézierkurven — Auswertung über DE CASTELJAU (II)

Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 11 \\ 7 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Graphische Lösung:

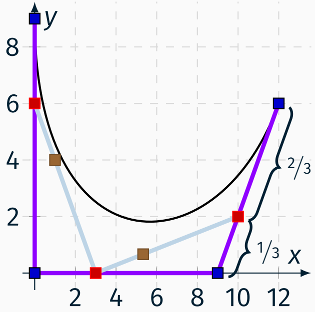


Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 11 \\ 7 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Graphische Lösung:

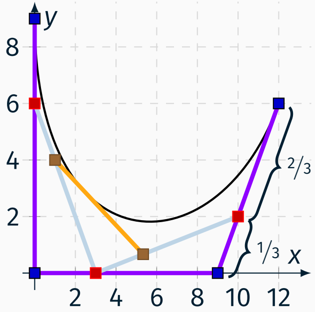


Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 11 \\ 7 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Graphische Lösung:

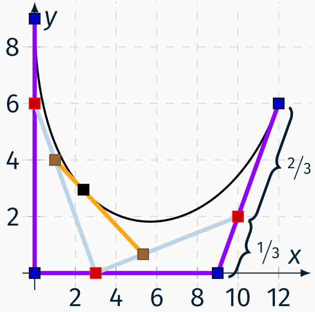


Sei eine kubische Bézier-Kurve mit den Kontrollpunkten

$$b_0 = \begin{pmatrix} 0 \\ 9 \end{pmatrix} \qquad b_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad b_2 = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \qquad \text{und} \qquad b_3 = \begin{pmatrix} 11 \\ 7 \end{pmatrix}$$

gegeben. Werten Sie die Kurve an $t = 1/3$ aus.

Graphische Lösung:



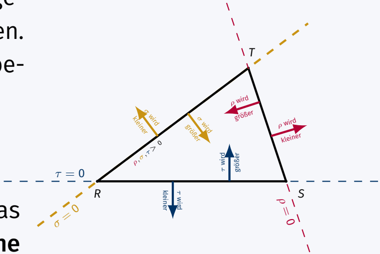
Rekapitulation — Multivariate Interpolation

Definition 6.9 (Baryzentrische Koordinaten)

Seien drei Punkte R, S, T in der Ebene gegeben, die **nicht** auf einer Geraden liegen. Dann lässt sich ein jeder Punkt P der Ebene *eindeutig* darstellen durch

$$P = \rho \cdot R + \sigma \cdot S + \tau \cdot T,$$

wobei $\rho + \sigma + \tau = 1$ gelten muss. Das Tupel (ρ, σ, τ) heißt dann **baryzentrische Koordinate** von P bezüglich $\Delta(R, S, T)$.



Baryzentrische Koordinaten

Die Koordinaten werden über das Lösen des Gleichungssystems

$$\begin{pmatrix} 1 & 1 & 1 \\ x_R & x_S & x_T \\ y_R & y_S & y_T \end{pmatrix} \cdot \begin{pmatrix} \rho \\ \sigma \\ \tau \end{pmatrix} = \begin{pmatrix} 1 \\ x_P \\ y_P \end{pmatrix}$$

mittels LR-Zerlegung.

Zudem gilt:

$$\rho = \frac{\mathbf{area}(\Delta(P,S,T))}{\mathbf{area}(\Delta(R,S,T))}, \quad \sigma = \frac{\mathbf{area}(\Delta(R,P,T))}{\mathbf{area}(\Delta(R,S,T))}$$

$$\text{und} \quad \tau = \frac{\mathbf{area}(\Delta(R,S,P))}{\mathbf{area}(\Delta(R,S,T))}$$

Baryzentrische Koordinaten — Lineare Interpolation

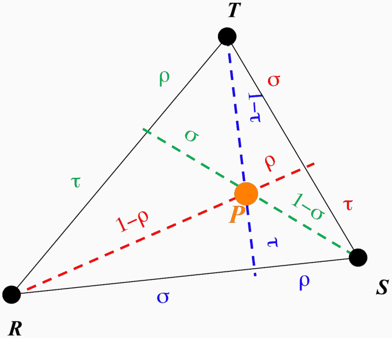
Lineare Interpolation

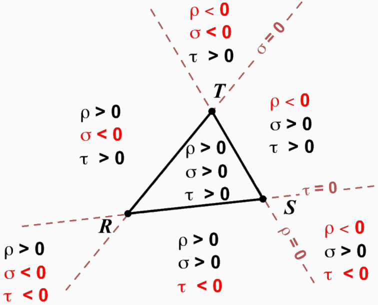
Seien nun in den Eckpunkten des Dreiecks $\Delta(R, S, T)$ die Daten f_R, f_S und f_T gegeben, so erhält man den Wert des linearen Interpolanten an einer Stelle $P \in \Delta(R, S, T)$ durch

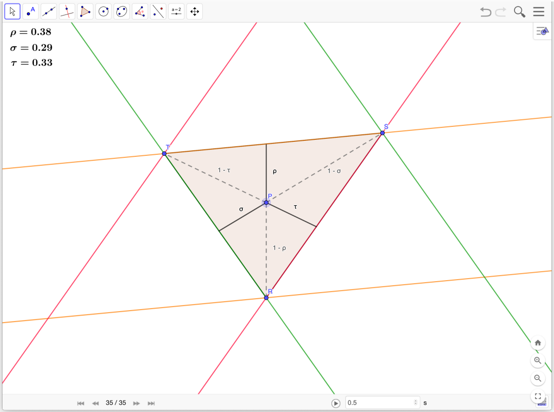
$$f_P = \rho \cdot f_R + \sigma \cdot f_S + \tau \cdot f_T,$$

wobei ρ, σ, τ die baryzentrischen Koordinaten von P bezüglich $\Delta(R, S, T)$ sind.

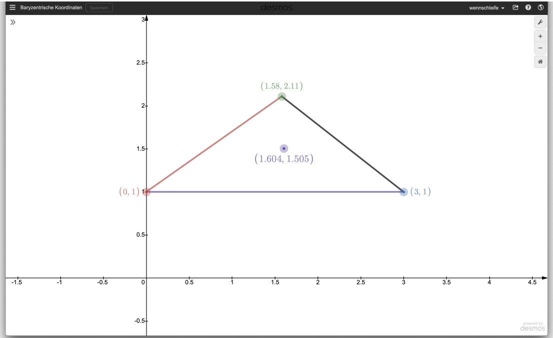
Baryzentrische Koordinaten — Teilungsverhältnis







GeoGebra-Applet für die Transformation
Punkt $\mapsto$ Baryzentrische Koordinaten
(https://www.geogebra.org/classic/rmxs2kwx)



Desmos-Applet für die Transformation
Baryzentrische Koordinaten $\mapsto$ Punkt
(https://www.desmos.com/calculator/pofgxrysn9)

Hausaufgabe — Baryzentrische Koordinaten und Bézierkurven

Hausaufgabe — Baryz. Koord. und Bézierkurven

Diese Hausaufgabe findet sich **komplett** auf StudOn!

Zusätzliche Literatur

Zusätzliche Literatur zur 6. Übung (1)

[1] J. Bergh und J. Löfström. *Interpolation Spaces*. Springer Berlin Heidelberg, 1976. DOI: 10.1007/978-3-642-66451-9. URL: https://doi.org/10.1007/978-3-642-66451-9.

[2] J.-P. Berrut und L. N. Trefethen. „Barycentric Lagrange Interpolation“. In: *SIAM Review* 46.3 (Jan. 2004), S. 501–517. DOI: 10.1137/s0036144502417715. URL: https://doi.org/10.1137/s0036144502417715.

[3] A. Bjorck und V. Pereyra. „Solution of Vandermonde Systems of Equations“. In: *Mathematics of Computation* 24.112 (Okt. 1970), S. 893. DOI: 10.2307/2004623. URL: https://doi.org/10.2307/2004623.

Zusätzliche Literatur zur 6. Übung (2)

[4] S. D. Conte und C. de Boor. *Elementary Numerical Analysis*. Society for Industrial und Applied Mathematics, Dez. 2017. DOI: 10.1137/1.9781611975208. URL: https://doi.org/10.1137/1.9781611975208.

[5] P. Davis. *Interpolation and Approximation*. Dover Books on Mathematics. Dover Publications, 1975. ISBN: 9780486624952. URL: https://books.google.de/books?id=228PAQAAMAAJ.

[6] D. C. Fraser. „Newton's Interpolation Formulas". In: *Journal of the Institute of Actuaries* 51.2 (Okt. 1918), S. 77–106. DOI: 10.1017/s0020268100028407. URL: https://doi.org/10.1017/s0020268100028407.

[7] G. H. Golub und C. F. van Loan. *Matrix Computations*. Fourth. JHU Press, 2013. ISBN: 9781421407944. URL: http://www.cs.cornell.edu/cv/GVL4/golubandvanloan.htm.

Zusätzliche Literatur zur 6. Übung (3)

[8] M. Hecht, K. B. Hoffmann, B. L. Cheeseman und I. F. Sbalzarini. *Multivariate Newton Interpolation*. 2020. arXiv: 1812.04256 [math.NA].

[9] R. A. Horn und C. R. Johnson. *Topics in Matrix Analysis*. Cambridge University Press, Apr. 1991. DOI: 10.1017/cbo9780511840371. URL: https://doi.org/10.1017/cbo9780511840371.

[10] J. L. de Lagrange. „Leçons élémentaires sur les mathématiques données à l'École Normale en 1795“. In: *Œuvres VII* (1877), S. 183–287.

[11] G. Mastroianni und G. V. Milovanović. *Interpolation Processes*. Springer Berlin Heidelberg, 2008. DOI: 10.1007/978-3-540-68349-0. URL: https://doi.org/10.1007/978-3-540-68349-0.

Zusätzliche Literatur zur 6. Übung (4)

[12] G. M. Phillips. *Interpolation and Approximation by Polynomials*. Springer New York, 2003. DOI: 10.1007/b97417. URL: https://doi.org/10.1007/b97417.

[13] G. P. Serb und I. P. Serb. „Newton's Interpolation Applied in the Practice of Designing a Salient Synchronous Generator". In: *2018 XIII International Conference on Electrical Machines (ICEM)*. IEEE, Sep. 2018. DOI: 10.1109/icelmach.2018.8507182. URL: https://doi.org/10.1109/icelmach.2018.8507182.

[14] „VII. Problems concerning interpolations". In: *Philosophical Transactions of the Royal Society of London* 69 (Dez. 1779), S. 59–67. DOI: 10.1098/rstl.1779.0008. URL: https://doi.org/10.1098/rstl.1779.0008.