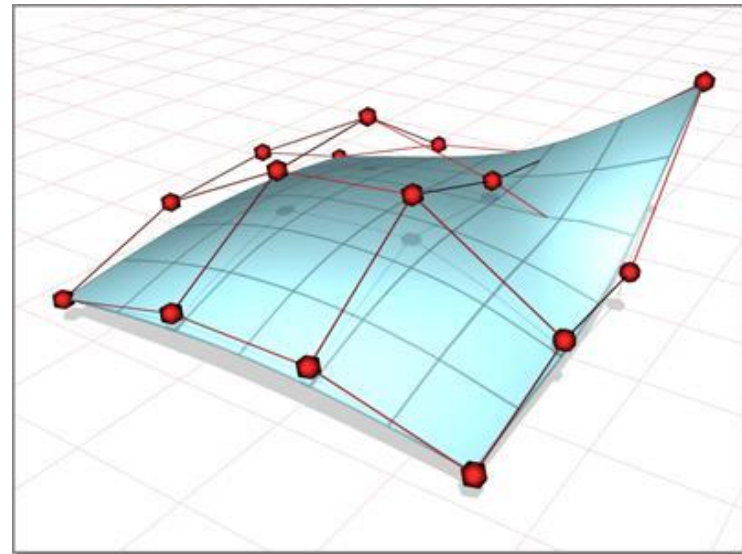


Modellierung mit Kurven





Inhalt

- [Polynombasen](#)
- [Parametrische Kurven](#)
- [Beziér Kurven](#)
- [Interpolierende Kurven](#)
- [Hermite Splines](#)
- [Stetiges Aneinanderfügen](#)
- [Basis-Splines](#)
- [Übersicht](#)

- Wir betrachten hier nur Polynome in einer Variablen t . Für Flächen werden Polynome in zwei Variablen benötigt.
- Die einfachste Darstellung von Polynomen ist in der Monombasis $\{ 1, t, t^2, \dots, t^g \}$
- Das Monom mit der höchsten Potenz dessen Koeffizient ungleich null ist definiert den Grad g des Polynoms
- Die Koeffizienten an die Basisfunktionen können als Vektor interpretiert werden und bilden den $(g+1)$ -dimensionalen Vektorraum der Polynome

$$\begin{aligned} f(t) &= a_0 + a_1 t + a_2 t^2 + \dots + a_g t^g \\ &= \sum_{i=0}^g a_i t^i = \langle \vec{a}, \vec{M}^g(t) \rangle \\ &= \vec{a}^T \vec{M}^g(t) = \vec{M}^g(t)^T \vec{a} \end{aligned}$$

Monombasis:

$$M_i^g(t) = t^i \quad \text{oder} \quad \vec{M}^g(t) = \begin{pmatrix} 1 \\ t \\ \vdots \\ t^g \end{pmatrix}$$

Koeffizientenvektor $\vec{a} = \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_g \end{pmatrix} \in \mathbf{R}^{g+1}$



- Die Fakultät einer positiven ganzen Zahl n ergibt sich aus dem Produkt aller ganzen Zahlen kleiner gleich n
- Binomialkoeffizienten sind mit Hilfe der Fakultät definiert
- Optional können die Binomialkoeffizienten auch über das Pascal'sche Dreieck berechnet werden
- Bernsteinbasis:

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

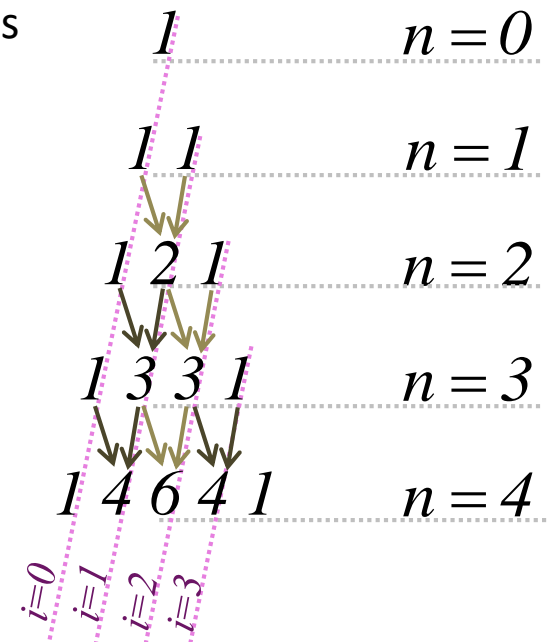
$$\vec{B}^2(t) = \begin{pmatrix} (1-t)^2 & 2(1-t)t & t^2 \end{pmatrix}^T$$

$$\vec{B}^3(t) = \begin{pmatrix} (1-t)^3 & 3(1-t)^2t & 3(1-t)t^2 & t^3 \end{pmatrix}^T$$

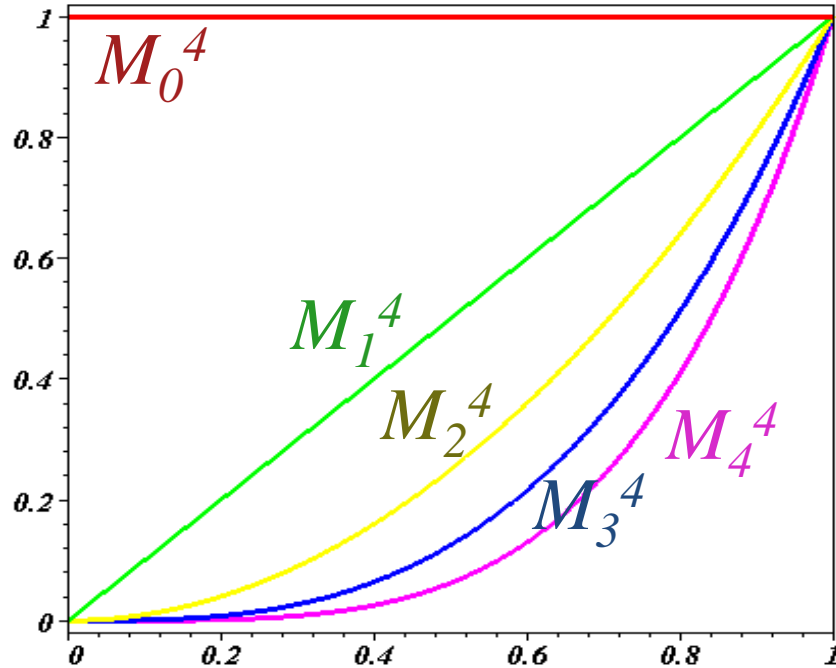
Fakultät: $n! = n \cdot (n-1) \cdot \dots \cdot 2 \cdot 1$

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

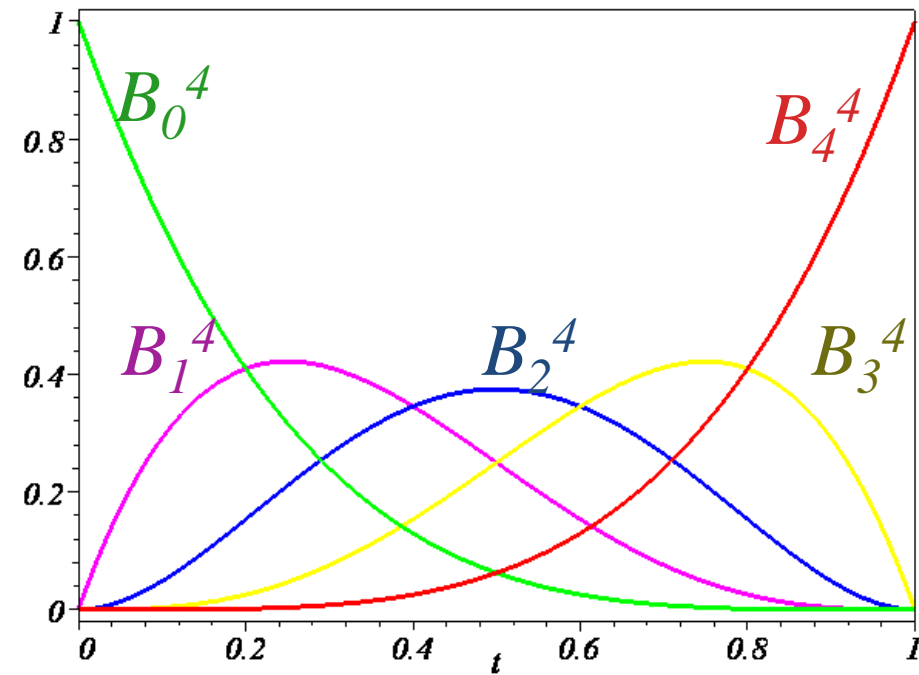
Pascal'sches
Dreieck:



Monombasis



Bernsteinbasis



Wichtig ist, dass nur der Definitionsbereich $t \in [0,1]$ betrachtet wird

Polynombasen

Basistransformation



- Dasselbe Polynome kann in verschiedenen Basen dargestellt werden. Dadurch ändert sich der Koeffizientenvektor
- Die Umrechnung des Koeffizientenvektors in eine andere Basis heißt Basistransformation, diese kann mit einer $(g+1) \times (g+1)$ -dimensionalen Matrix dargestellt werden
- Die Basistransformation erfolgt durch Matrix-Vektor-Multiplikation
- Die inverse Matrix transformiert den Koeffizientenvektor in umgekehrter Richtung

$$f(t) = \vec{a}^T \vec{M}^g(t) \stackrel{(1)}{=} \vec{b}^T \vec{B}^g(t)$$

Basistransformation:

$$\vec{b} = T_{B \leftarrow M} \vec{a}, \quad T_{B \leftarrow M} \in \mathbf{R}^{(g+1) \times (g+1)}$$

Inverse Basistransformation:

$$\vec{a} = T_{M \leftarrow B} \vec{b} = (T_{B \leftarrow M})^{-1} \vec{b} \quad (2)$$

Zusammenhang zw. Basistransformation und Transformation der Basen:

$$f(t) = \vec{a}^T \vec{M}^g(t) \stackrel{(2)}{=} \vec{b}^T \left((T_{B \leftarrow M})^{-1} \right)^T \vec{M}^g(t)$$

$$\stackrel{(1)}{\Rightarrow} \vec{B}^g(t) = A_{B \leftarrow M} \vec{M}^g(t)$$

$$A_{B \leftarrow M} = \left((T_{B \leftarrow M})^{-1} \right)^T$$

Polynombasen

Beispieltransformation

- Auch die Basen stehen in einem linearen Zusammenhang zueinander, der durch eine Matrix $\mathbf{A}_{B \leftarrow M}$ darstellbar ist
- Diese Matrix kann bei Transformation von der Monombasis durch ausmultiplizieren der anderen Basis abgelesen werden
- Rechts ist dies für die Bernsteinbasis durchgeführt
- Die Transformationsmatrix für den Koeffizientenvektor ergibt sich durch Transponieren und Invertieren.

$$\begin{aligned} \vec{B}^3(t) &= \begin{pmatrix} (1-t)^3 \\ 3(1-t)^2 t \\ 3(1-t)t^2 \\ t^3 \end{pmatrix} = \begin{pmatrix} 1-3t+3t^2-t^3 \\ 3t-6t^2+3t^3 \\ 3t^2-3t^3 \\ t^3 \end{pmatrix} \\ &= \underbrace{\begin{pmatrix} 1 & -3 & 3 & -1 \\ 0 & 3 & -6 & 3 \\ 0 & 0 & 3 & -3 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{\mathbf{A}_{B \leftarrow M}} \begin{pmatrix} 1 \\ t \\ t^2 \\ t^3 \end{pmatrix} \end{aligned}$$

$$\Rightarrow \mathbf{T}_{B \leftarrow M} = \left((\mathbf{A}_{B \leftarrow M})^T \right)^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & \frac{1}{3} & 0 & 0 \\ 1 & \frac{2}{3} & \frac{1}{3} & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$



- Für die Auswertung von Polynomen gibt es je nach Basis mehrere Möglichkeiten deren Laufzeit sich in Bezug auf den Grad unterscheidet.
- Für die Monombasis benötigt eine direkte Umsetzung der Formel im Grad quadratisch viele Rechenoperationen
- Mit dem Horner Schema kann, die Anzahl der Rechenoperationen reduziert werden, so dass nur noch proportional zum Grad viele Operationen notwendig sind ist.

$$f(t) = a_0 + a_1 t + a_2 t^2 + \dots + a_g t^g$$

Pseudo-Code Standardauswertung:

```
f = a0
for i = 1..g do
    Mi = t
    for j = 2..i do
        Mi *= t
    f += ai * Mi
```

Horner Schema

$$f(t) = a_0 + t(a_1 + t(\dots + t(a_{g-1} + ta_g)\dots))$$

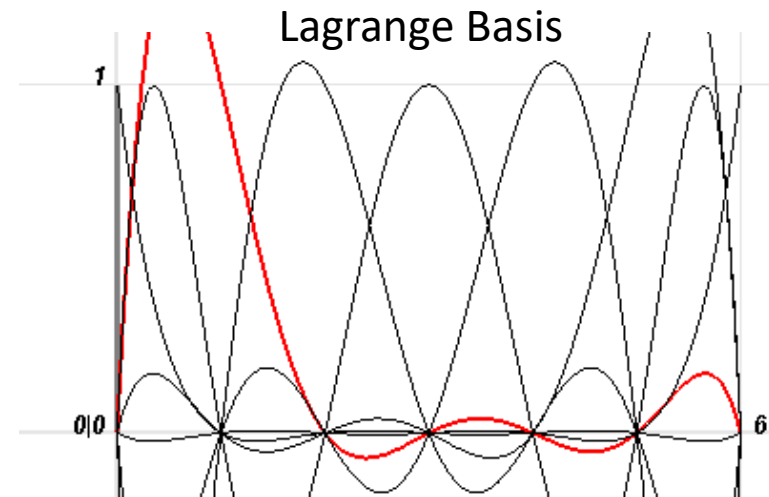
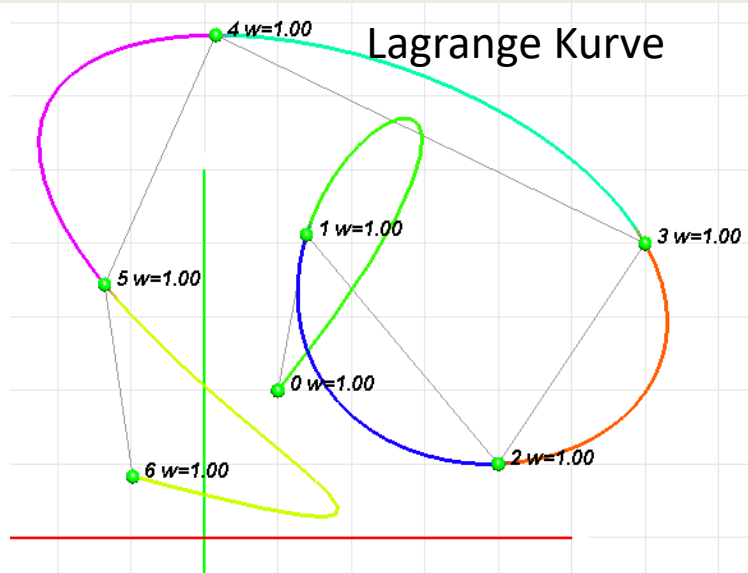
Pseudo-Code Horner Schema

```
f = ag
for i = g-1..0 do
    f = ai + t*f
```



Parametrische Kurven

Modellierung von Kurven



Kontrollpunktparadigma

- Kontrollpolygon aus Kontrollpunkten definiert den Kurvenverlauf
- Gewichte geben bei rationalen Basen zusätzliche Freiheitsgrade zum modellieren

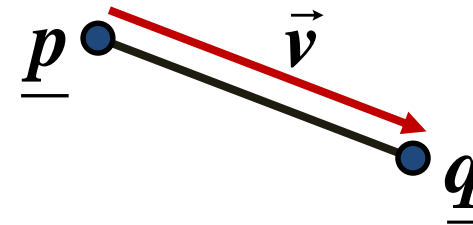
Basisfunktionen

- Der Einfluss der verschiedenen Kontrollpunkte wird durch polynomiale Basisfunktionen gesteuert
- Je nach Wahl der Basis hat die resultierende Kurve andere Eigenschaften

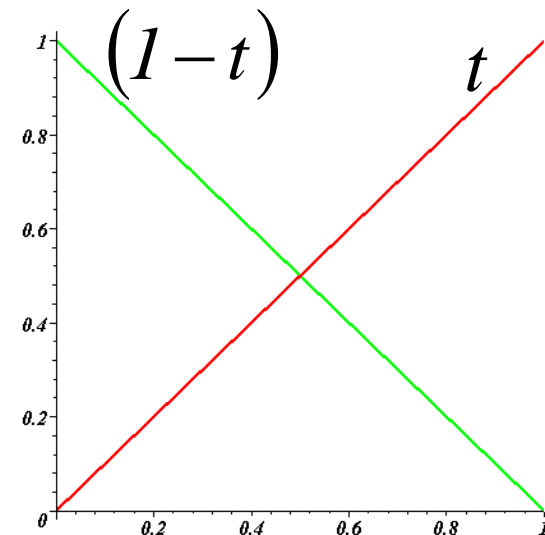
Parametrische Kurven

Motivation der Basisdarstellung

- Die einfachste parametrische Kurve ist durch ein geradliniges Segment zwischen zwei Punkten definiert
- Dazu werden nur lineare Basisfunktionen benötigt
 - Monombasis: $\{1, t\}$
wird an Anfangspunkt und Differenzvektor multipliziert
 - Bernsteinbasis: $\{1-t, t\}$
wird direkt an die Kontrollpunkte multipliziert
 - Bei $t=0$ und $t=1$ ist eine Basis 1 und die andere 0 \rightarrow Endpunkte werden interpoliert
 - Die Basis summiert sich überall zu $1-t+t=1$

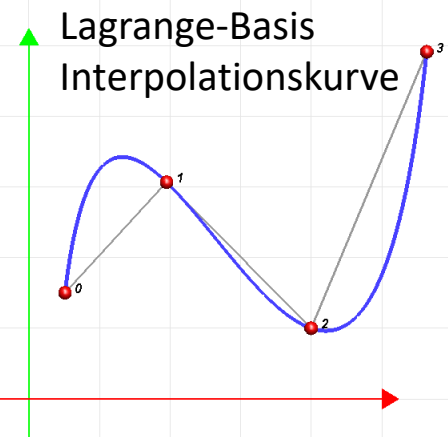
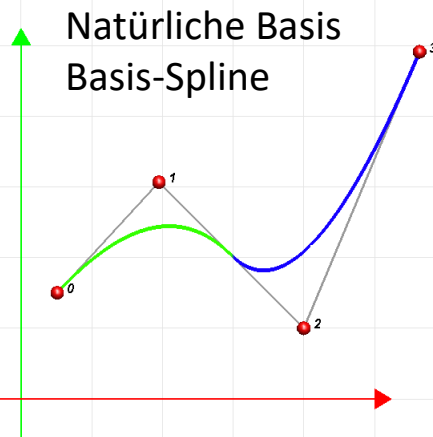
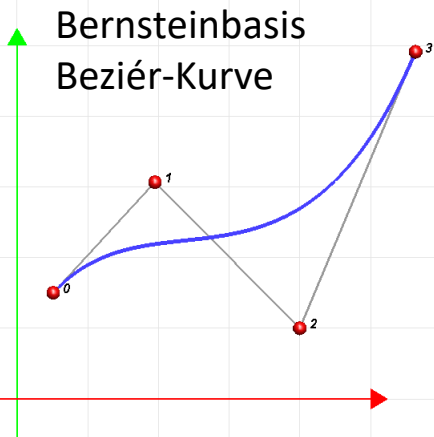


$$\begin{aligned}\underline{c}(t) &= \underline{p} + \vec{v}t \\ &= (1-t)\underline{p} + t\underline{q}\end{aligned}$$



Parametrische Kurven

Eigenschaften die aus Basis folgen



- Je nach den gewählten Basisfunktionen ergibt sich mit denselben Kontrollpunkten unterschiedliche Kurven mit anderen Eigenschaften
- Splines sind aus mehreren Kurvensegmenten so zusammengesetzt, dass ein glatter Übergang entsteht
- gewünschte Eigenschaften
 - Glattheit
 - Kontrollpunktinterpolation
 - Endtangentialinterpolation ... Tangenten der Kurvenendpunkte entsprechen Endsegmenten des Kontrollpolygons
 - Konvexe Hüllen Eigenschaft: Kurvenverlauf nur innerhalb der konvexen Hülle der Kontrollpunkte

Parametrische Kurven

Matrixdarstellung



- Im Vergleich zu Polynomen werden die skalaren Koeffizienten zu Kontrollpunkten und die Koeffizientenvektoren zu Kontrollpunktmatrizen
- Bei der Basistransformation bleibt alles gleich, außer dass die Kontrollpunktmatrizen durch Matrix-Matrix-Multiplikation transformiert werden
- Im 3D hat die Kontrollpunktmatrix einfach eine weitere Zeile

Beispiel Beziér-Kurve, die mit Bernsteinbasis definiert wird und deren Kontrollpunkte auch Beziér-Punkte genannt werden

$$\underline{c}(t) = \sum_{i=0}^g \underline{b}_i B_i^g(t) \\ = \underline{K}_b \cdot \mathbf{B}^g(t)$$

Kontrollpunktmatrix

Bsp.: $g=3$

$$\underline{c}(t) = \begin{pmatrix} b_{0,x} & b_{1,x} & b_{2,x} & b_{3,x} \\ b_{0,y} & b_{1,y} & b_{2,y} & b_{3,y} \end{pmatrix} \begin{pmatrix} (1-t)^3 \\ 3(1-t)^2 t \\ 3(1-t)t^2 \\ t^3 \end{pmatrix}$$



Beziér Kurven

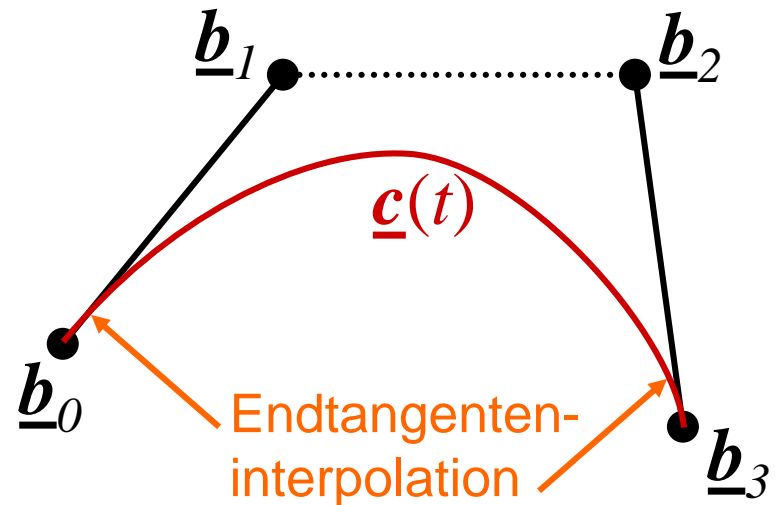
einleitendes Beispiel

- In den meisten Tools werden Bezier-Kurven dritten Grades für die Definition von Kurven verwendet
- Die Kurve ist durch vier Bezier-Punkte $\underline{b}_0, \dots, \underline{b}_3$ definiert
- $\underline{b}_0, \underline{b}_3$ definieren Start- und Endpunkt
- $\underline{b}_1, \underline{b}_2$ definieren die Tangenten in Start- und Endpunkt

Bezier-Kurve 3.Grades:

$$\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = (1-t)^3 \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} + 3t(1-t)^2 \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} + 3t^2(1-t) \begin{pmatrix} x_2 \\ y_2 \end{pmatrix} + t^3 \begin{pmatrix} x_3 \\ y_3 \end{pmatrix}$$

mit $t \in [0,1]$



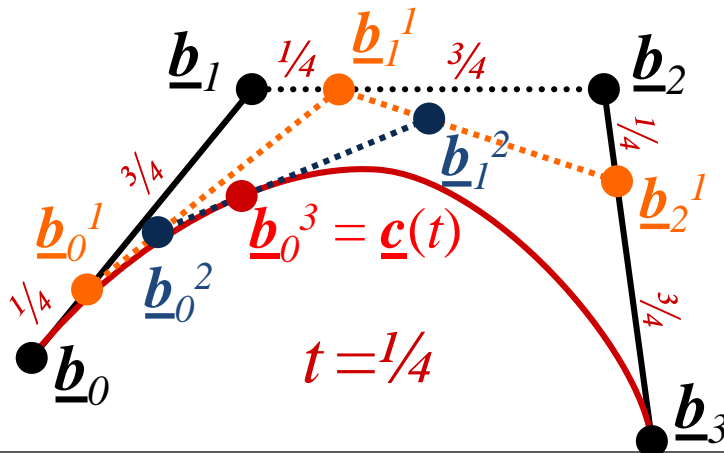
Beziér Kurven

Berechnung von Kurvenpunkten

Kurvenauswertung nach De Casteljau

- Die Produkte von 4-5 Zahlen beim Auswerten von Bezier-Kurven 3. Grades sind numerisch nicht stabil.
- Deshalb verwendet man das Schema nach de Casteljau:

$$\begin{aligned} \underline{c}(t) &= \underline{b}_0^s & \underline{b}_i^0 &= \underline{b}_i \\ \underline{b}_i^r &= (1-t)\underline{b}_i^{r-1} + t\underline{b}_{i+1}^{r-1} \end{aligned}$$



Beweis für de Casteljau:

$$\begin{aligned} \underline{c}(t) &= (1-t)^3 \underline{b}_0 + 3t(1-t)^2 \underline{b}_1 + \\ &\quad 3t^2(1-t) \underline{b}_2 + t^3 \underline{b}_3 \end{aligned}$$

$$\begin{aligned} &= (1-t)^2 ((1-t) \underline{b}_0 + t \underline{b}_1) + \\ &\quad 2t(1-t) ((1-t) \underline{b}_1 + t \underline{b}_2) + \\ &\quad t^2 ((1-t) \underline{b}_2 + t \underline{b}_3) \end{aligned}$$

$$= (1-t)^2 \underline{b}_0^1 + 2t(1-t) \underline{b}_1^1 + t^2 \underline{b}_2^1$$

$$\begin{aligned} &= (1-t) ((1-t) \underline{b}_0^1 + t \underline{b}_1^1) + \\ &\quad t ((1-t) \underline{b}_1^1 + t \underline{b}_2^1) \end{aligned}$$

$$= (1-t) \underline{b}_0^2 + t \underline{b}_1^2$$

$$= \underline{b}_0^3 = \underline{c}(t)$$

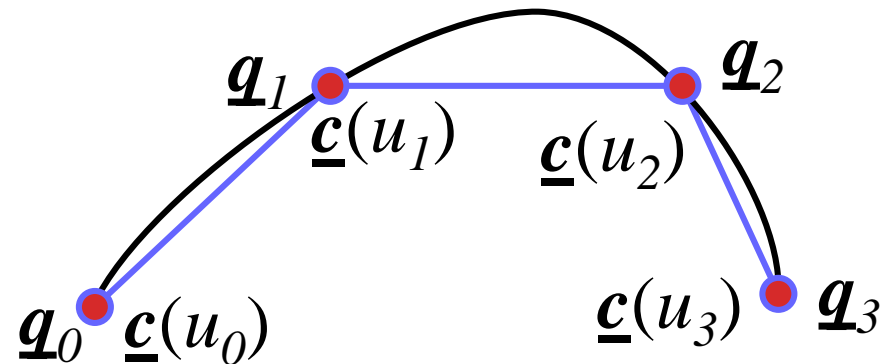


Interpolierende Kurven

Lagrange Interpolation



- Wenn die Kurve durch alle Kontrollpunkte gehen soll, (Interpolation) so muss man angeben für welche Parameterwerte $t=u_i$ dies geschehen soll.
- Die u_i nennt man Stützstellen und den Vektor U der aus einer Stützstelle pro Kontrollpunkt definiert ist Stützstellenvektor
- Aus dem Stützstellenvektor definiert man die Lagrange Basis $L_i^g(t)$ so, dass $L_i^g(u_i)=1$ ist und $L_i^g(u_{j \neq i})=0$. Daraus folgt die Interpolationseigenschaft.



Interpolationsproblem:

- geg: Kontrollpunkt $\underline{q}_{i=0..g}$ und Stützstellen $u_{i=0..g}$
- ges.: Kurve $\underline{c}(t)$ mit $\underline{c}(u_i)=\underline{q}_i$
- Lösung: Kurve in Lagrange-Basis:

$$\underline{c}(t) = \sum_{i=0}^g \underline{q}_i L_i^g(t)$$

$$L_i^g(t) = \prod_{0=k \neq i}^g \frac{t - u_k}{u_i - u_k} = \frac{(t - u_0) \cdots (t - u_{i-1}) \cdots (t - u_g)}{(u_i - u_0) \cdots (u_i - u_{i-1}) \cdots (u_i - u_g)}$$

Interpolierende Kurven

Aufstellen der Lagrange Basis



Beispiel:

- ◆ Gegeben: $\{u_i\}_i = \{0, 1, 3\}$
 $\{\underline{q}_i\}_i = \{(1, 0), (-1, 0), (3, 3)\}$

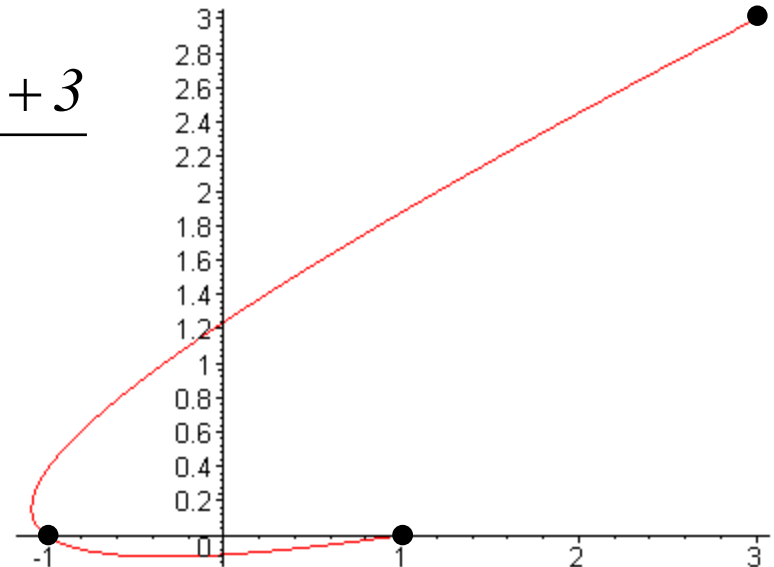
$$L_0^2(t) = \frac{(t-1)(t-3)}{(0-1)(0-3)} = \frac{t^2 - 4t + 3}{3}$$

$$L_1^2(t) = \frac{t(t-3)}{-2}$$

$$L_2^2(t) = \frac{t(t-1)}{6}$$

$$\underline{c}(t) = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \frac{t^2 - 4t + 3}{3} + \begin{pmatrix} -1 \\ 0 \end{pmatrix} \frac{t(t-3)}{-2} + \begin{pmatrix} 3 \\ 3 \end{pmatrix} \frac{t(t-1)}{6}$$

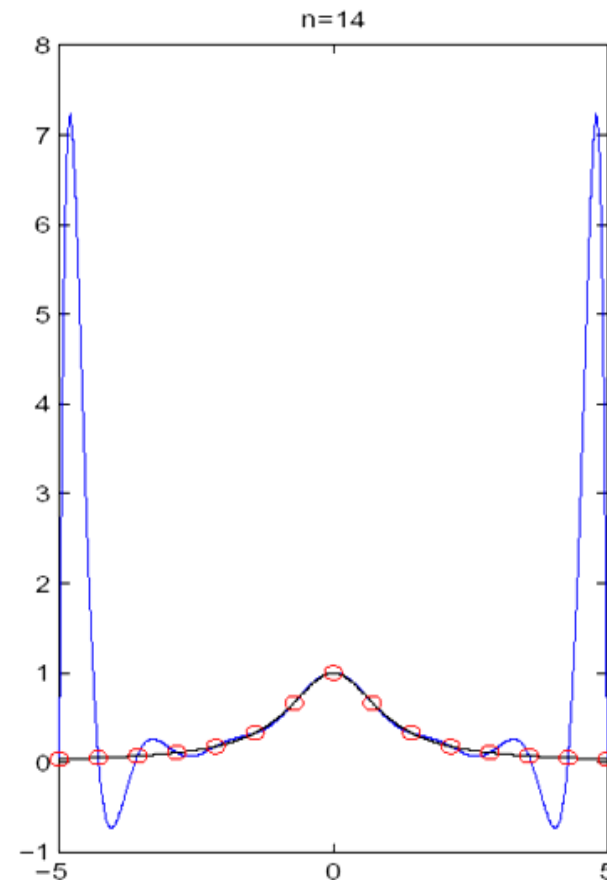
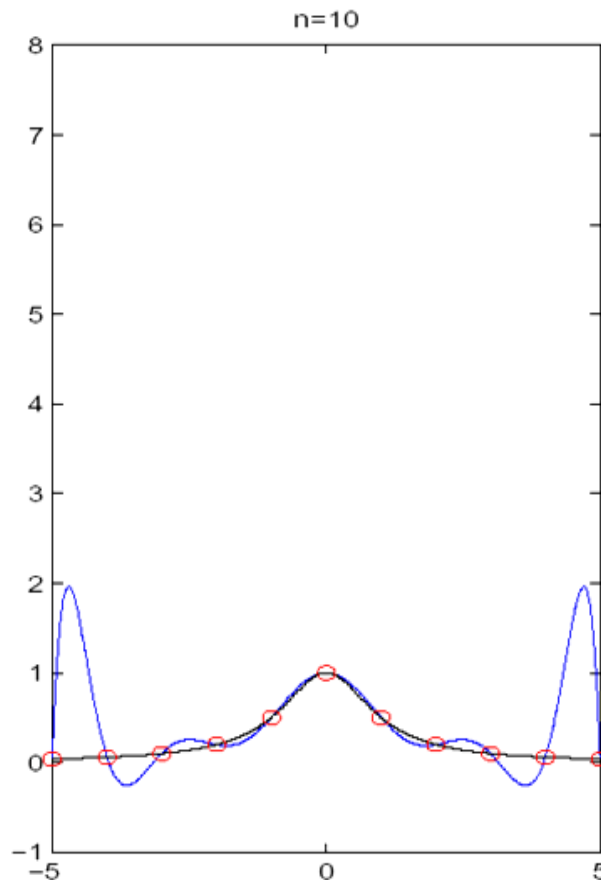
$$L_i^g(t) = \prod_{0=k \neq i}^g \frac{t - u_k}{u_i - u_k}$$



Interpolierende Kurven

Überschwingungsproblematik

- ◆ **Vorsicht:** bei vielen Knoten kommt es bei der Lagrange-Interpolation zu Überschwingen



Interpolierende Kurven

Weitere Eigenschaften



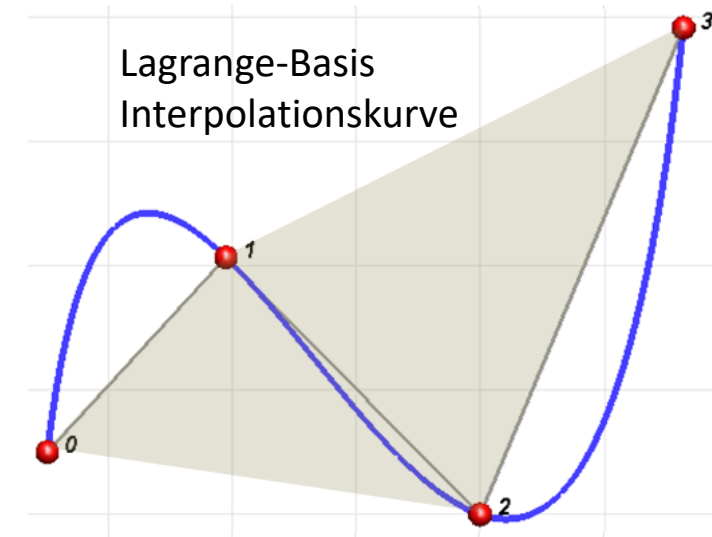
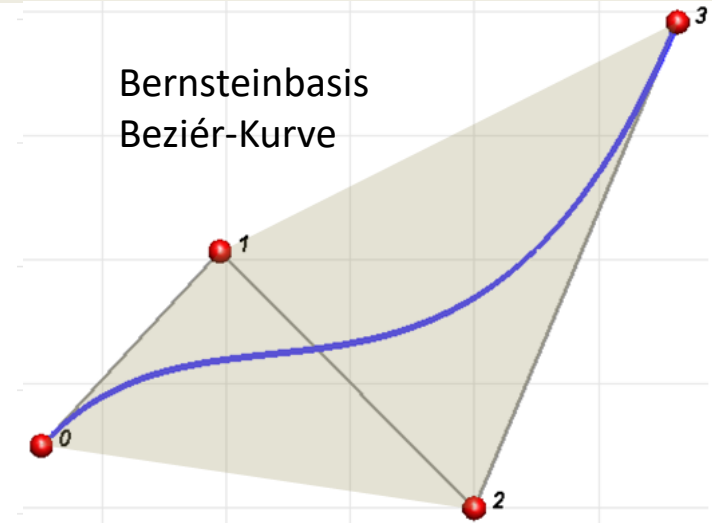
- Affine Invarianz ... die Basisfunktionen summieren sich für alle t zu eins:

$$\forall t, g, U : 1 = \sum_{i=0}^g B_i^g(t) = \sum_{i=0}^g L_i^g(t)$$

- Konvexe Hülleneigenschaft ... die Kurve liegt in der konv. Hülle der Kontrollpunkte. Immer bei affiner Invarianz zusammen mit positiven Basisfunktionen:

$$\forall t, i, g : B_i^g(t) \geq 0$$

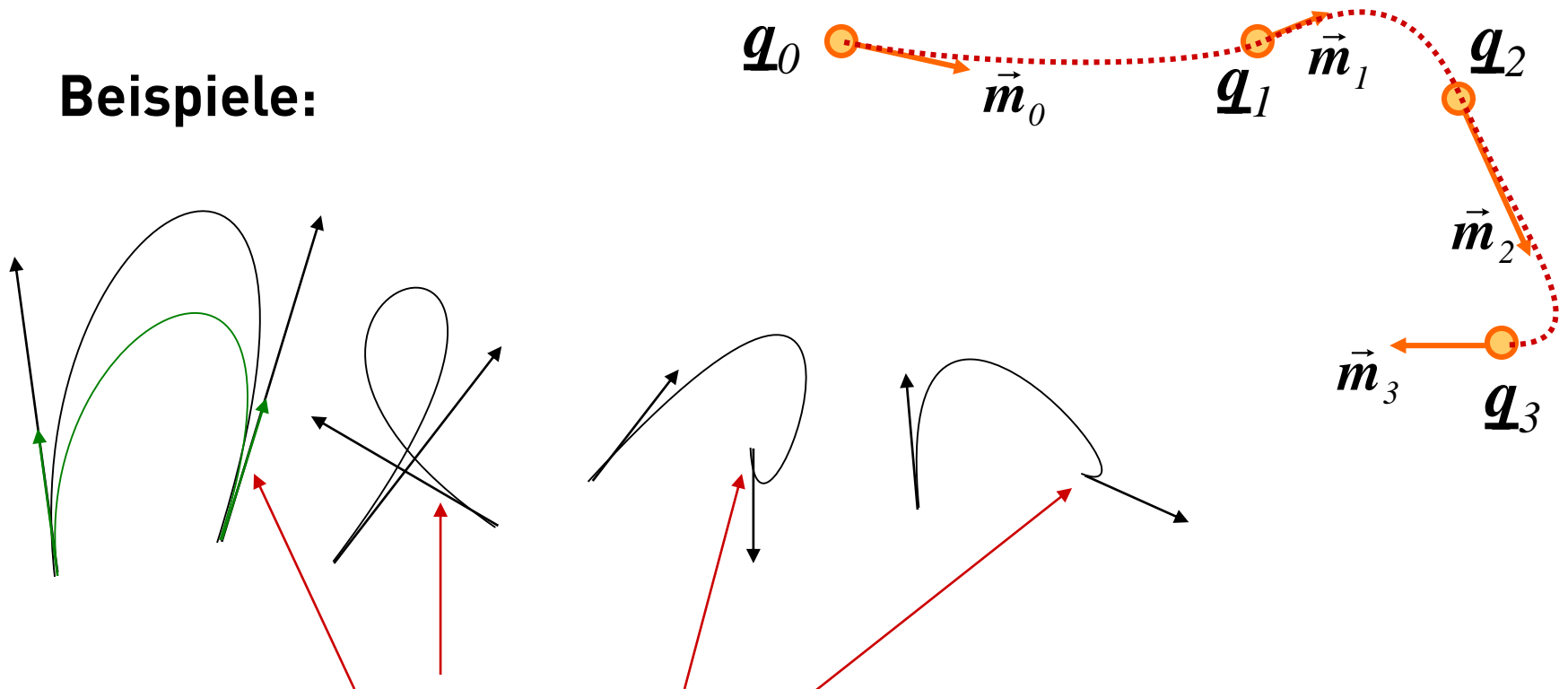
für Lagrange Basis nicht erfüllt.



Idee:

- Interpolation von Positions- und Tangenteninformation

Beispiele:



Vorsicht: In Darstellung zeigen die Pfeile am Ende entgegen der Vektoren \vec{m} , die für die Definition der Hermite-Kurve verwendet wird!!!



Gegeben:

- Parameter $u_0, u_1, \dots, u_n \in \mathbb{R}$,
- Punkte $\underline{q}_0, \underline{q}_1, \dots, \underline{q}_n \in \mathbb{R}^d$
- Tangenten $\vec{m}_0, \vec{m}_1, \dots, \vec{m}_n \in \mathbb{R}^d$

Gesucht:

- n Kurvensegmente $\underline{c}_i(t): [u_i, u_{i+1}] \rightarrow \mathbb{R}^d$
- mit $\underline{c}_i(u_i) = \underline{q}_i$, $\dot{\underline{c}}_i(u_i) = \vec{m}_i$
 $\underline{c}_i(u_{i+1}) = \underline{q}_{i+1}$, $\dot{\underline{c}}_i(u_{i+1}) = \vec{m}_{i+1}$

Lösung der Interpolationsaufgabe

(Kontrolltangenten-
interpolation)

$$\begin{aligned} \underline{c}_i(t) = & H_0^3(s_i) \underline{q}_i + \Delta_i \cdot H_1^3(s_i) \vec{m}_i + \\ & + \Delta_i \cdot H_2^3(s_i) \vec{m}_{i+1} + H_3^3(s_i) \underline{q}_{i+1}, t \in [u_i, u_{i+1}] \end{aligned}$$

wobei $s_i = (t - u_i) / \Delta_i$ und $\Delta_i = u_{i+1} - u_i$

in Klausur gegeben

Beispiel

- ◆ Hermite-Polynome vom Grad 3:

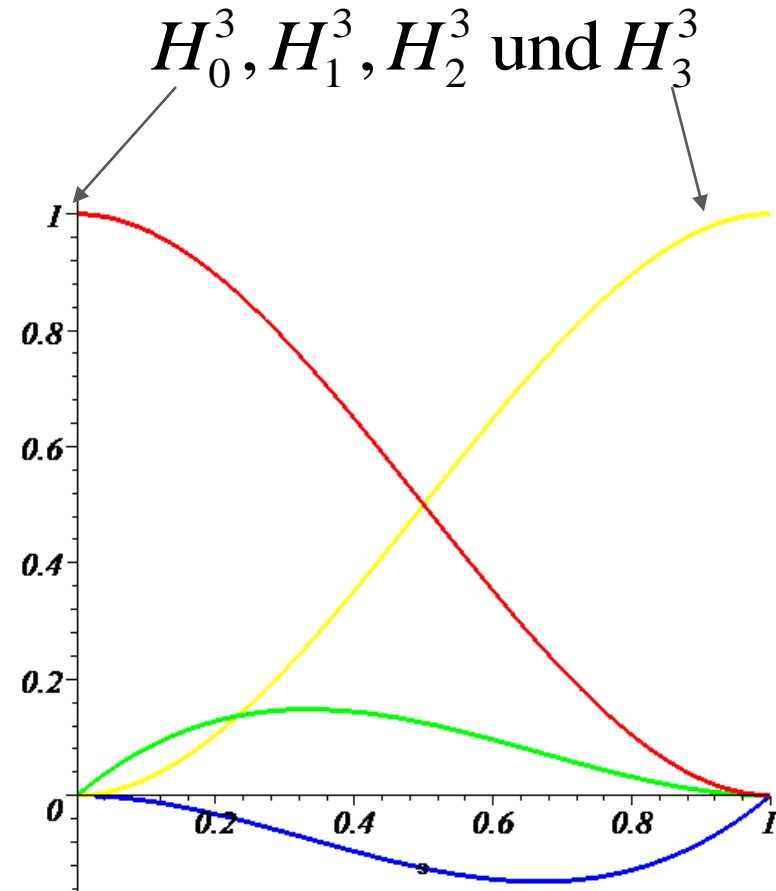
$$H_0^3(t) = (1-t)^2(1+2t)$$

$$H_1^3(t) = t(1-t)^2$$

$$H_2^3(t) = -t^2(1-t)$$

$$H_3^3(t) = (3-2t)t^2$$

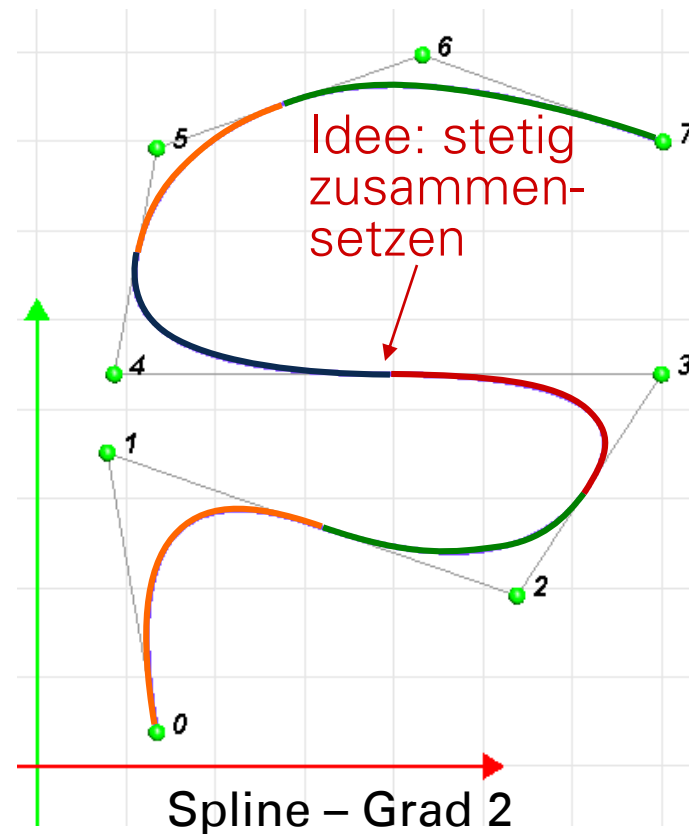
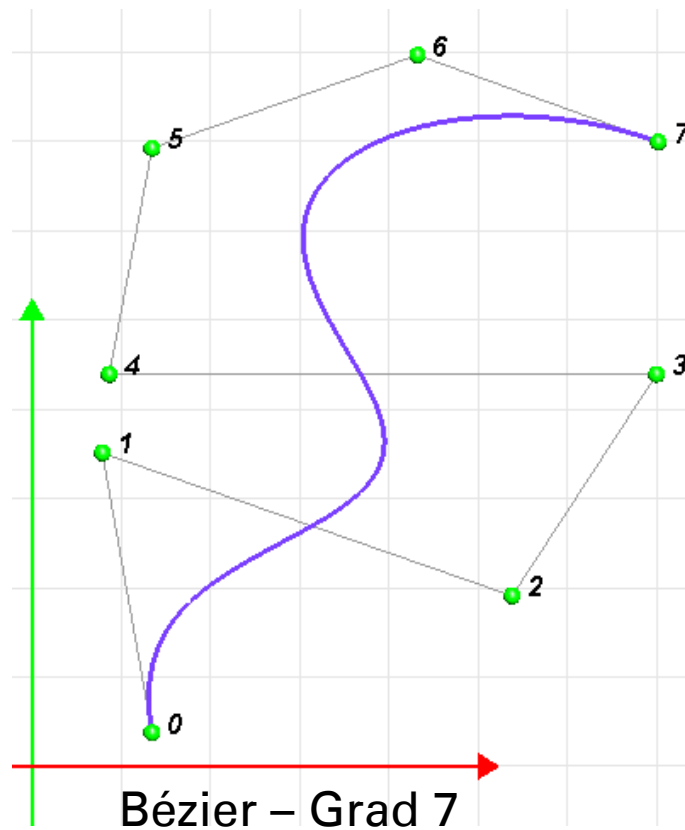
in Klausur gegeben



Stetiges Aneinanderfügen

Motivation

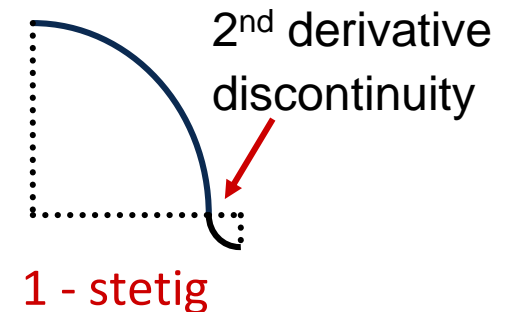
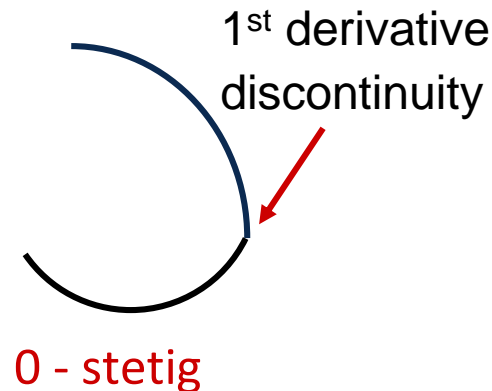
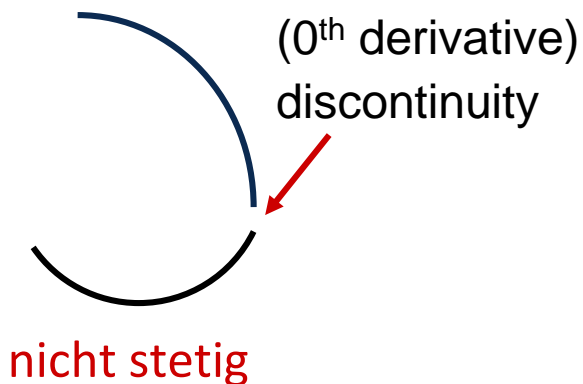
- ◆ Mit wachsendem Grad folgen Bézier-Kurven immer weniger dem Kontrollpolygon:



Stetiges Aneinanderfügen

Verschiedene Stetigkeiten

- Der Anschluss zwischen zwei Kurvensegmenten wird nach der Anzahl der übereinstimmenden Ableitungen klassifiziert:



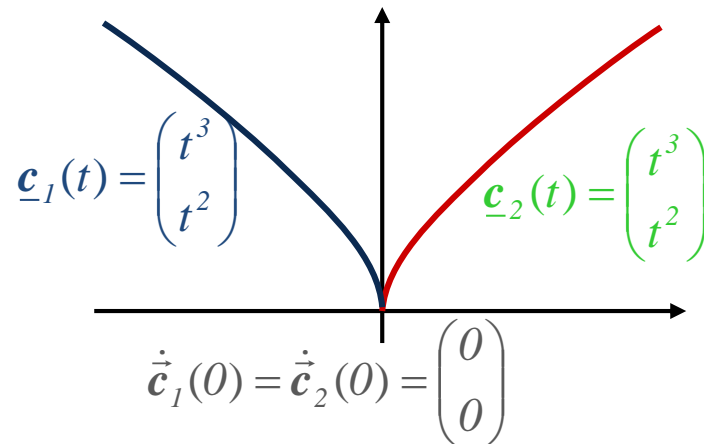
- Man unterscheidet zwischen parametrischer (C) und geometrischer (G) Stetigkeit

Stetiges Aneinanderfügen

Parametrisch versus Geometrisch

C^k -stetig

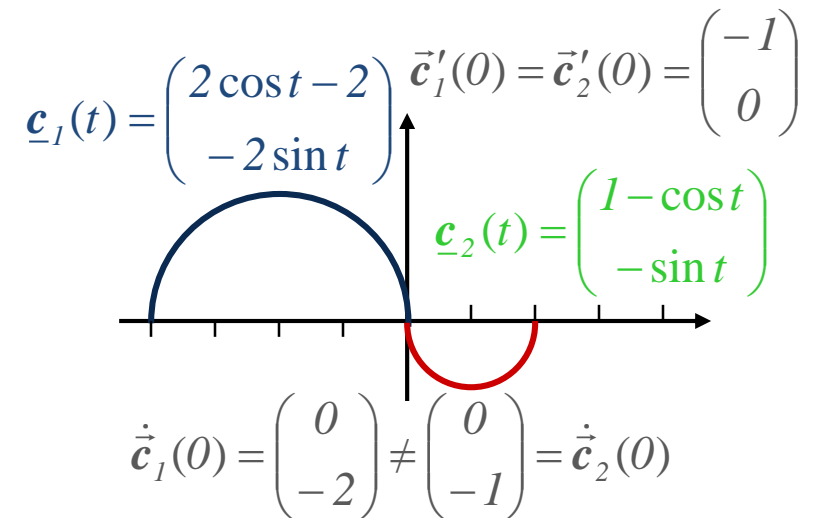
- Die ersten k Ableitungen nach dem Parameter sind identisch



- Kurve ist C^∞ – aber nicht einmal G^1 – stetig

G^k -stetig

- Die ersten k Ableitungen nach der Kurvenlänge sind identisch



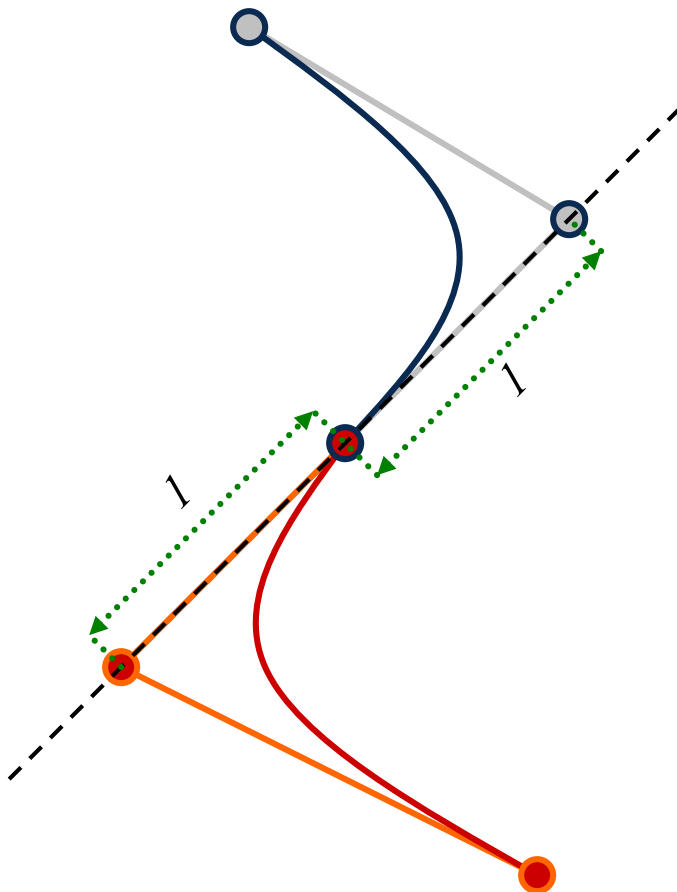
- Kurve ist G^1 – aber nur C^0 – stetig

Stetiges Aneinanderfügen

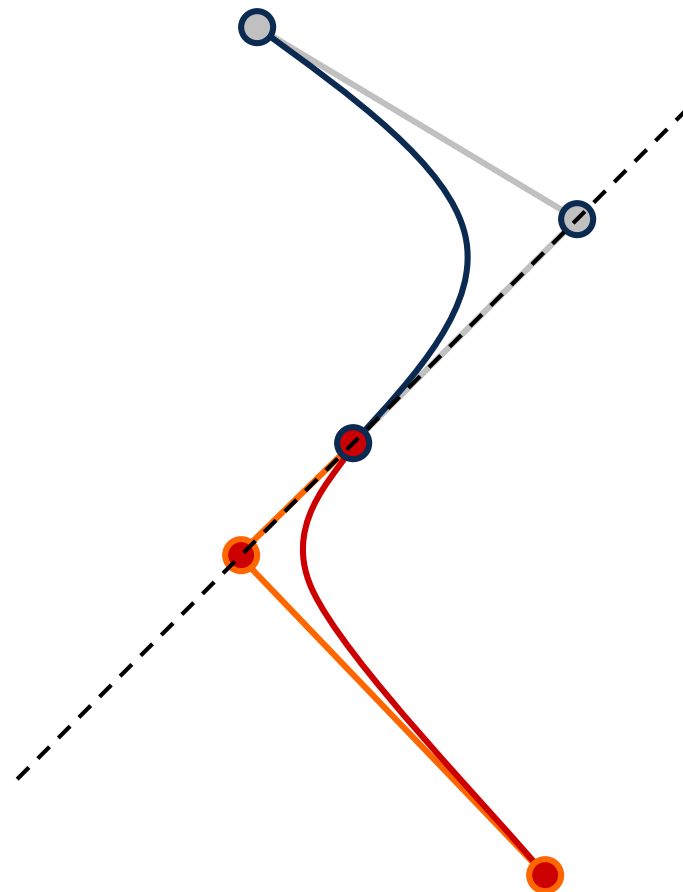
Tangentenstetigkeit



C^1 -stetiger Bézier Spline

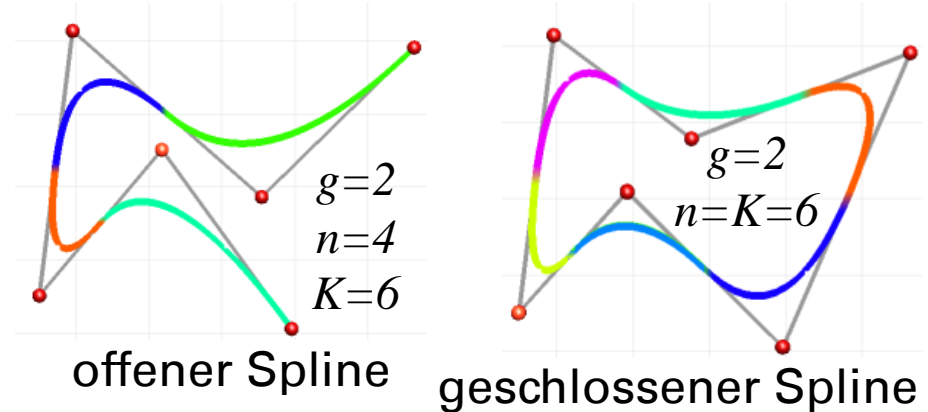


G^1 -stetiger Bézier Spline



- Basis-Splines vom Grad g sind aus n Kurvensegmenten zusammengesetzt, die $k=g-1$ stetig (C^k) aneinanderstoßen
- Um B-Splines wie andere polynomiale Kurven verwenden zu können, werden die natürlichen Basisfunktionen $N_i^g(t)$ mit der rekursiven Konstruktionsformel nach Cox und De Boor definiert
- Die Kontrollpunkte \underline{d}_i heißen De Boor Punkte
- Man unterscheidet offene und geschlossene Splines

$$\underline{c}(t) = \sum_{i=0}^{K-1} \underline{d}_i N_i^g(t)$$



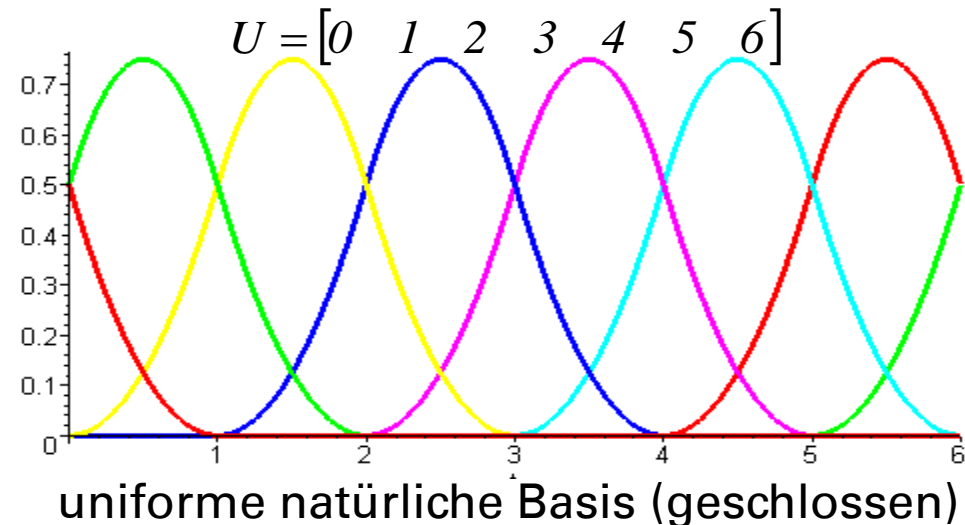
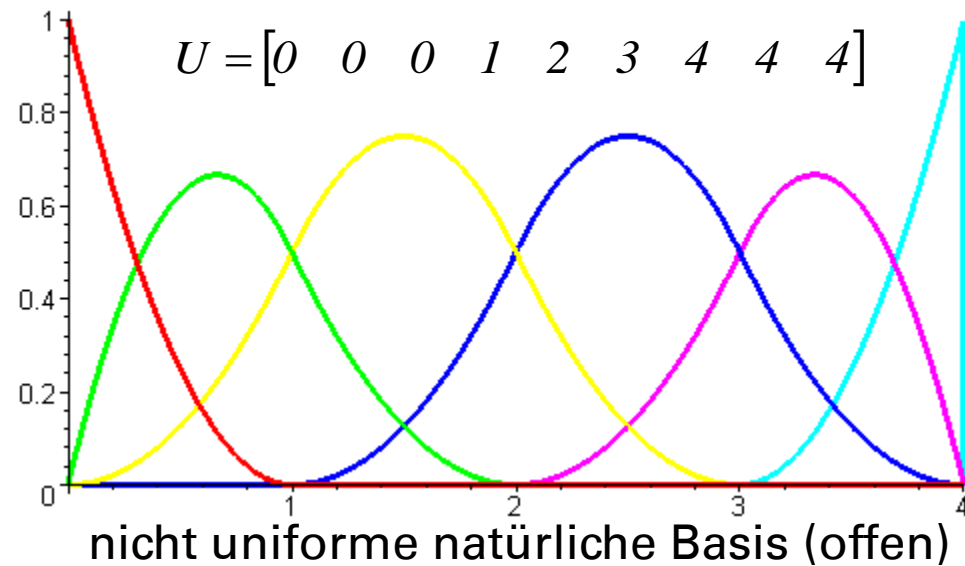
- Bei konstantem Grad g kann eine beliebige Anzahl $K > g$ von Kontrollpunkten genutzt werden
- es gilt
 - $K = n + g \dots$ offene Splines
 - $K = n \dots$ geschlossene Splines

Basis-Splines

Natürliche Basis



- Ähnlich zur Lagrange-Interpolation wird ein Stützstellenvektor U mit m Einträgen u_i verwendet, der den Parameterbereich in Intervalle einteilt, es gilt
 - $m=K+g+1$... offene Splines
 - $m=K$... geschlossene Splines
- anstelle von Stützstelle bzw. Stützstellenvektor werden oft die Begriffe Knoten und Knotenvektor verwendet
- Sind die Knoten $u_i=a \cdot i$ äquidistant, so spricht man von einem uniformen ansonsten von einem nicht uniformen Spline



Basis-Splines

Symbolübersicht



- i ... variabel eingesetzter Laufindex
- n ... Anzahl der Segmente
- g ... Grad der Kurvensegmente ($g+1$ Freiheitsgrade pro Kurvensegment)
- $k=g-1$... Stetigkeit zwischen Kurvensegmenten ($k+1=g$ Nebenbedingungen zwischen Kurvensegmenten)
- $K=n+g$ oder n ... Anzahl Kontrollpunkte
- $m=K+g+1$ oder n ... Anzahl Einträge im Knotenvektor

Basis-Splines

Cox De Boor-Rekursion

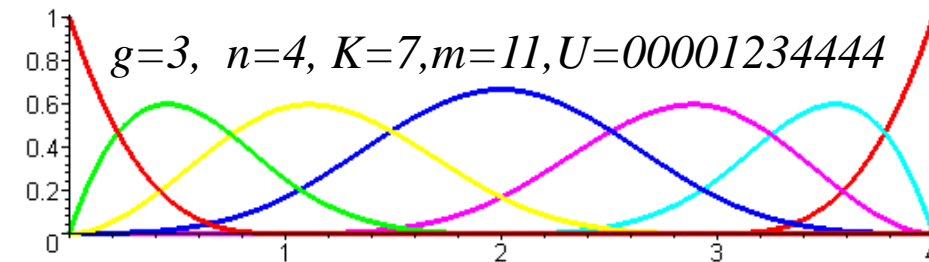
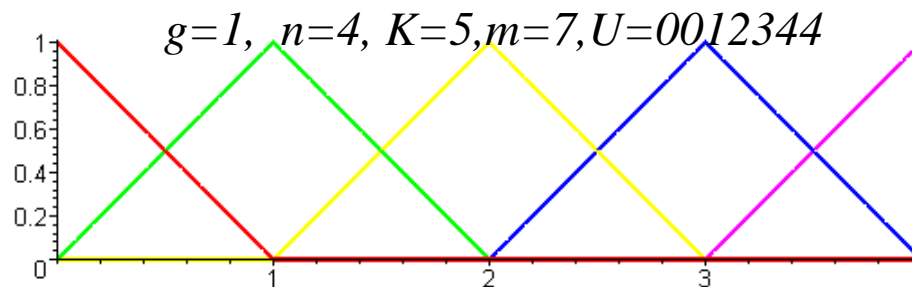
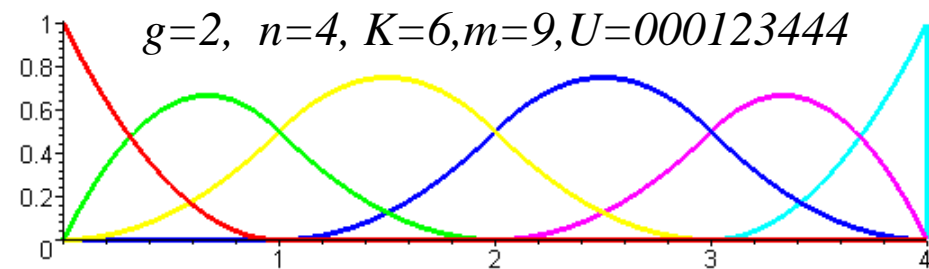
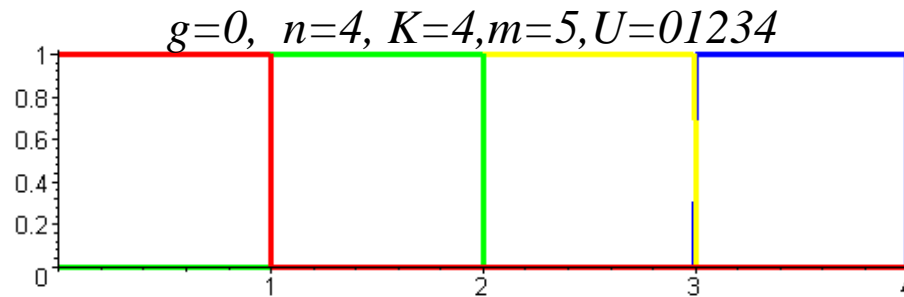
- Rekursion nach Cox De Boor am Bsp. offener Splinebasen

$$N_i^0(t) = \begin{cases} 1 & \text{falls } u_i \leq t < u_{i+1} \text{ und } u_i < u_{i+1} \\ 0 & \text{sonst} \end{cases}$$

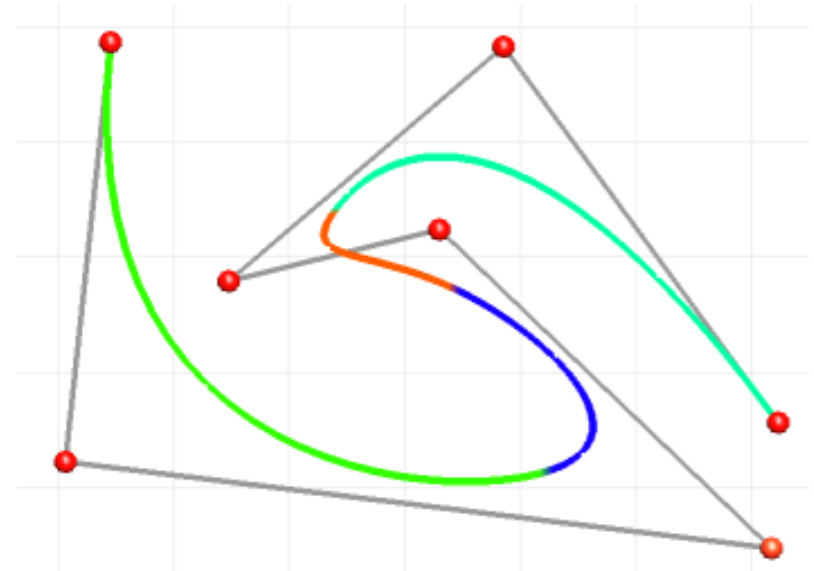
↑
pro t nur einmal $\neq 0$
↑
Einfluss nur von nicht degenerierten Intervallen

$$N_i^g(t) = \frac{t - u_i}{u_{i+g} - u_i} N_i^{g-1}(t) + \frac{u_{i+1+g} - t}{u_{i+1+g} - u_{i+1}} N_{i+1}^{g-1}(t) \quad u_i \leq t < u_{i+g+1}$$

in Klausur gegeben

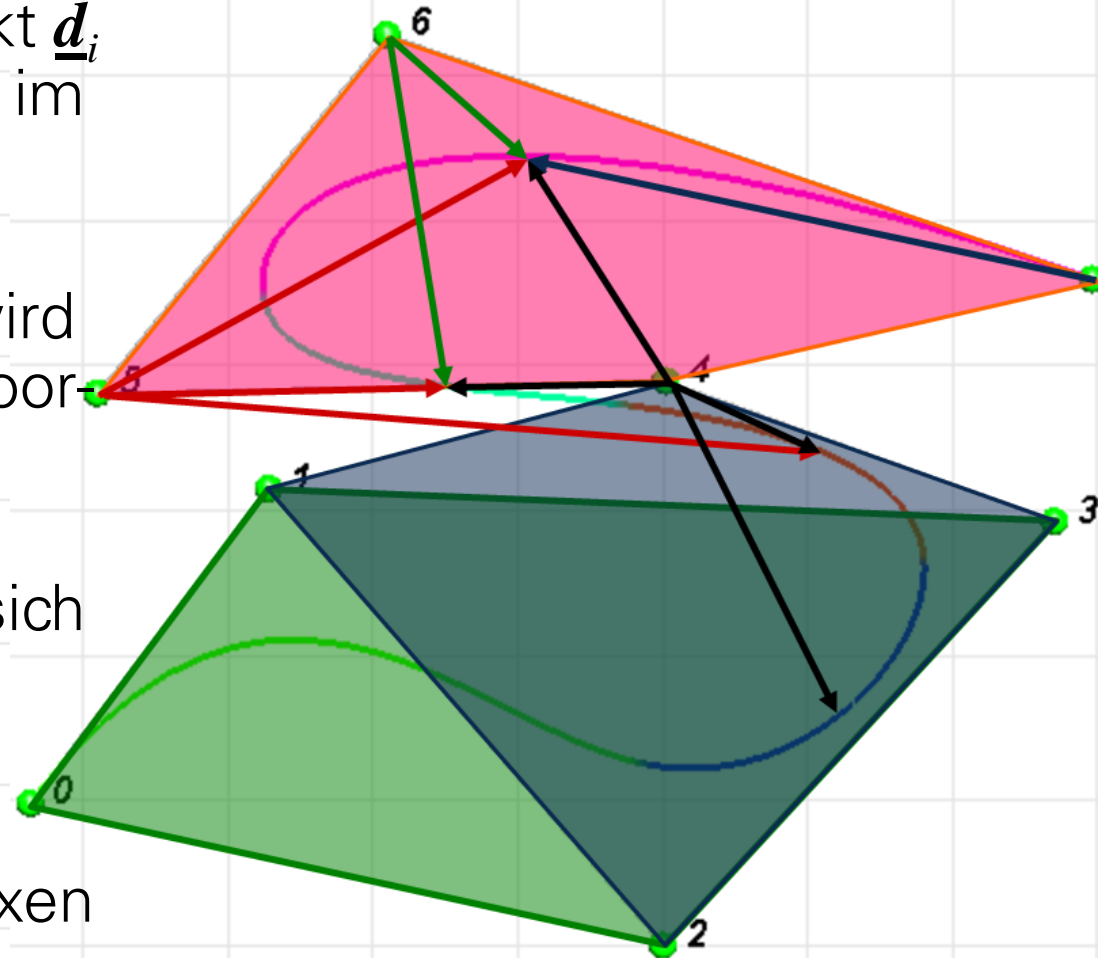


- Wählt man μ aufeinander folgende $u_{i \dots i+\mu-1}$ gleich, so nennt man u_i einen Knoten der Multiplizität μ .
- in u_i vermindert sich die Stetigkeit auf $C^{k-(\mu-1)} = g - \mu$
- bei einer $\mu=g$ Multiplizität von u_i wird d_i interpoliert.
- Bei $\mu=g+1$ bekommt der B-Spline einen Sprung
- Dies ist nur an den Endpunkten von offenen B-Splines erlaubt



Bsp.: $n=4, g=3, K=7, m=11,$
 $U=[0,0,0,0,1,2,3,4,4,4,4]$

- Die Basisfunktion N_i^g bzw. der De Boor Punkt \underline{d}_i beeinflusst den Spline im Parameterbereich $[u_i, u_{i+g+1})$
- Das Intervall $[u_i, u_{i+1})$ wird nur von den $g+1$ De Boor-Punkten $\underline{d}_{i-g}, \dots, \underline{d}_i$ beeinflusst
- Die $N_i^g(t)$ summieren sich zu 1 und sind größer gleich null
- Der Spline liegt in der Vereinigung der konvexen Hüllen von $\underline{d}_i, \underline{d}_{i+1}, \dots, \underline{d}_{i+g}$



$g=3, n=5, m=12, K=8, U=000012345555$



Übersicht über Basen und Kurven

• Bernstein-Basis

- affin invariant
- positiv

$$\sum_{i=0}^g \underline{b}_i B_i^g(t)$$

• Beziér-Kurve (Beziér-Punkte)

- approximierend
- Konvexhülleneigenschaft
- Endtangenteninterpolation

• Lagrange-Basis

- affin invariant
- unabhängig

$$\sum_{i=0}^g \underline{q}_i L_i^g(t)$$

• Lagrange-Kurve

- Kontrollpunktinterpolation

• Hermite-Basis

- lokale Definition

• Hermite-Spline

- Kontrolltangenteninterpolation
- lokaler Kontrolltangenteneinfluss
- C^1 -stetig

• natürliche Basis

- affin invariant
- positiv
- lokale Definition

$$\sum_{i=0}^{K-1} \underline{d}_i N_i^g(t)$$

• Basis-Splines (De Boor Punkte)

- Endtangenteninterpolation
- lokaler Kontrollpunkteinfluss
- Konvexhülleneigenschaft
- C^{g-1} -stetig