

## 11. Darstellung von Kurven und Flächen

### Bézier–Kurven.

Unser Ziel ist es, polynomiale Kurven auf dem Rechner möglichst effizient darzustellen. Hierzu nutzen wir die Basisdarstellung mit Hilfe der Bernstein-Polynome aus; man vergleiche hierzu auch Abschnitt 4.

#### Definition (11.1)

Für  $n \in \mathbb{N}_0$  sind die *Bernstein-Polynome* vom Grad  $n$  definiert durch

$$B_k^n(t) := \binom{n}{k} t^k (1-t)^{n-k}, \quad k = 0, 1, \dots, n.$$

#### Satz (11.2)

Die Bernstein-Polynome erfüllen die folgenden elementaren Eigenschaften

- a)  $B_k^n(t)$  besitzt eine  $k$ -fache Nullstelle in  $t = 0$  und eine  $(n-k)$ -fache Nullstelle in  $t = 1$ . Ferner hat man die Symmetrie

$$B_k^n(t) = B_{n-k}^n(1-t), \quad t \in \mathbb{R}, \quad k = 0, 1, \dots, n.$$

- b) Die  $B_k^n(t)$  sind auf dem Intervall  $[0, 1]$  nichtnegativ und haben in  $t_k^n := k/n$  ein striktes globales Maximum (bezogen auf  $[0, 1]$ ).

- c) Es gelten für  $t \in \mathbb{R}$

$$1 = \sum_{k=0}^n B_k^n(t), \quad t = \sum_{k=0}^n \frac{k}{n} B_k^n(t), \quad t^2 = \sum_{k=0}^n \frac{k(k-1)}{n(n-1)} B_k^n(t). \quad (11.3)$$

Insbesondere bilden die  $B_k^n$ ,  $k = 0, \dots, n$ , eine *Zerlegung der Eins*.

- d) Die Bernstein-Polynome lassen sich rekursiv über den folgenden Neville-artigen Algorithmus auswerten

$$B_0^0(t) := 1,$$

für  $m = 1, \dots, n$

$$B_{-1}^{m-1}(t) := B_m^{m-1}(t) := 0, \quad (11.4)$$

$$B_k^m(t) := t B_{k-1}^{m-1}(t) + (1-t) B_k^{m-1}(t), \quad k = 0, \dots, m,$$

end  $m$ .

e) *Weierstraßscher Approximationssatz:*

Für  $f \in C[0, 1]$  konvergieren die Bernstein-Approximationen

$$B_n(f)(t) := \sum_{k=0}^n f\left(\frac{k}{n}\right) B_k^n(t), \quad 0 \leq t \leq 1,$$

für  $n \rightarrow \infty$  gleichmäßig auf  $[0, 1]$  gegen die Funktion  $f$ .

f) Die Bernstein-Polynome  $(B_0^n, \dots, B_n^n)$  bilden eine Basis des Polynomraums  $\Pi_n$ .

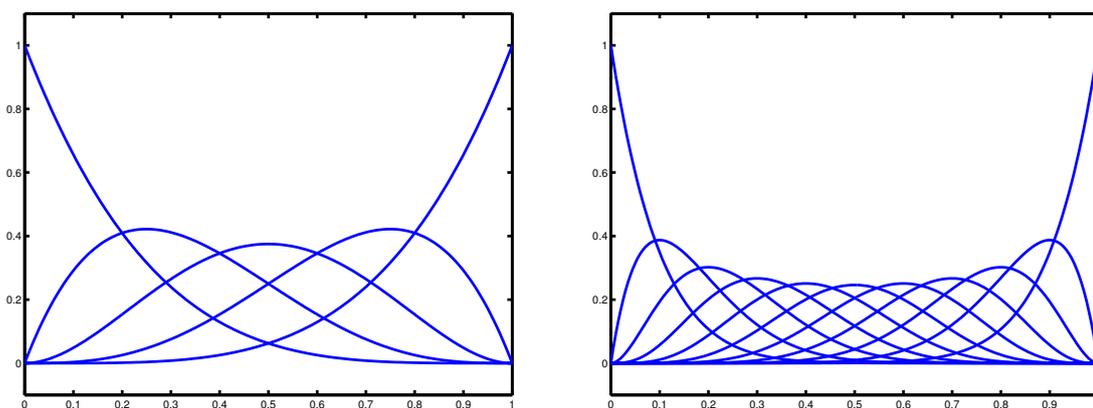


Abb. 11.1 Bernstein-Polynome  $B_k^n$  für  $n = 4$  und  $n = 10$

### Definition (11.5)

Aufgrund des Satzes (11.2) f) hat jedes Polynom  $p \in \Pi_n$  eine eindeutig bestimmte Darstellung als Linearkombination der Bernstein Polynome

$$p(t) = \sum_{k=0}^n a_k B_k^n(t) \tag{11.6}$$

Die Darstellung (11.6) heißt die *Bézier-Darstellung*<sup>1</sup> des Polynoms  $p$ .

Die Koeffizienten  $a_0, \dots, a_n$  heißen die *Bézier-Punkte* von  $p$ , das Polygon mit den Ecken  $(k/n, a_k)$ ,  $k = 0, \dots, n$ , heißt das *Bézier-Polygon* zu  $p$ .

### Bemerkung (11.7)

Für die Ableitung der Bernstein-Polynome ergibt sich aus (11.1)

$$\frac{d}{dt} B_k^n(t) := n (B_{k-1}^{n-1}(t) - B_k^{n-1}(t)), \quad k = 0, \dots, n, \tag{11.8}$$

wobei wie in (11.4)  $B_{-1}^{n-1}(t) := B_n^{n-1}(t) := 0$  gesetzt wird.

<sup>1</sup>Pierre Etienne Bézier (1910-1999); Paris

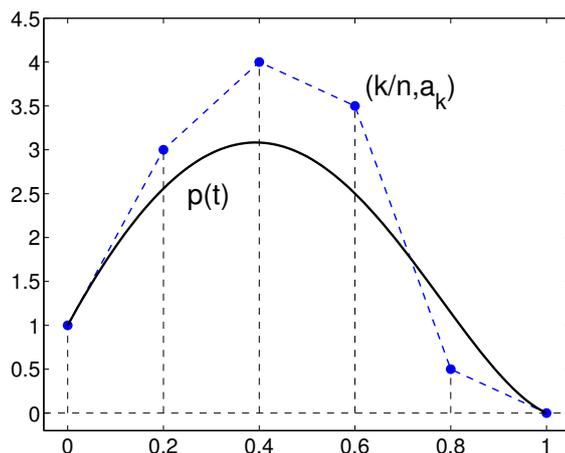


Abb. 11.2 Bézier-Polygon und zugehöriges Polynom

Verwendet man die Beziehung (11.8) zur Differentiation des Polynoms  $p(t)$ , so ergibt sich

$$p'(t) := n \sum_{k=0}^{n-1} (a_{k+1} - a_k) B_k^{n-1}(t). \quad (11.9)$$

An dieser Beziehung liest man ab, dass das Bézier-Polynom in den Randpunkten  $t = 0$  und  $t = 1$  tangential am Bézier-Polygon verläuft, vgl. Abbildung 11.2.

Weiterhin folgt aus der Bézier-Darstellung (11.6) und den Eigenschaften  $B_k^n(t) \geq 0$  sowie  $\sum_{k=0}^n B_k^n(t) = 1$ , dass die Werte  $p(t)$  ganz in der konvexen Hülle der Bézier-Punkte  $(a_0, \dots, a_n)$  verlaufen müssen:

$$p(t) \in \text{conv}(a_0, \dots, a_n). \quad (11.10)$$

Bei Kenntnis der Bézier-Punkte  $(a_0, \dots, a_n)$  ist damit der ungefähre Verlauf des Bézier-Polynoms einzuschätzen. Ferner lässt sich durch Veränderung einzelner Bézier-Punkte gezielt Einfluß auf den Verlauf des Bézier-Polynoms nehmen. Man nennt die Bézier-Punkte daher auch *Kontrollpunkte*.

Die genannten Eigenschaften von Bézier-Polynomen lassen sich unmittelbar auf den vektorwertigen Fall einer polynomialen Kurve  $\mathbf{p}(t) \in \mathbb{R}^m$  übertragen. Die Bézier-Darstellung lautet dann

$$\mathbf{p}(t) = \sum_{k=0}^n \mathbf{a}_k B_k^n(t), \quad (11.11)$$

wobei die Bézier-Punkte nun Vektoren  $\mathbf{a}_k \in \mathbb{R}^m$  sind. Das Bézier-Polygon lässt sich dann als ein Polygonzug im  $\mathbb{R}^m$  interpretieren mit den Ecken  $(\mathbf{a}_0, \dots, \mathbf{a}_n)$  und es gelten analog zum skalaren Fall:

$$\begin{aligned} \mathbf{p}(t) &\in \text{conv}(\mathbf{a}_0, \dots, \mathbf{a}_n). \\ \mathbf{p}'(0) &\parallel (\mathbf{a}_1 - \mathbf{a}_0), \quad \mathbf{p}'(1) \parallel (\mathbf{a}_n - \mathbf{a}_{n-1}). \end{aligned} \quad (11.12)$$

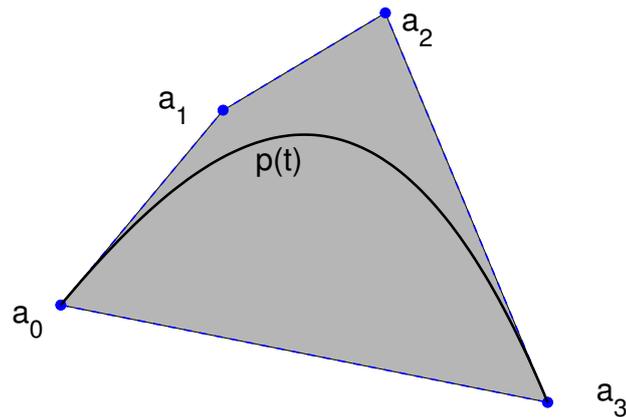


Abb. 11.3 Bézier-Polygon und zugehöriges Polynom im  $\mathbb{R}^2$

## Der Algorithmus von de Casteljau<sup>2</sup>.

Der Wert  $p(t)$  eines Bézier-Polynoms (11.6) lässt sich durch fortgesetzte lineare Interpolationen mit Hilfe eines Neville-artigen Algorithmus berechnen. Der Einfachheit halber betrachten wir hier wieder den skalaren Fall und definieren in Verallgemeinerung von (11.6) die folgenden Bézier-Polynome

$$a_i^m(t) := \sum_{k=0}^m a_{i+k} B_k^m(t), \quad m = 0, \dots, n, \quad i = 0, \dots, n - m. \quad (11.13)$$

Offensichtlich sind die  $a_i^m(t)$  Polynome aus  $\Pi_m$ . Es gilt  $a_i^0(t) = a_i$ ,  $i = 0, \dots, n$ , sowie  $a_0^n(t) = p(t)$ .

### Satz (11.14)

Die Bézier-Polynome  $a_i^m(t)$ ,  $0 \leq t \leq 1$ , lassen sich wie folgt rekursiv berechnen:

$$\begin{aligned} a_i^0(t) &:= a_i, \quad i = 0, \dots, n, \\ a_i^m(t) &:= (1-t) \cdot a_i^{m-1}(t) + t \cdot a_{i+1}^{m-1}(t), \quad m = 1, \dots, n, \quad i = 0, \dots, n - m. \end{aligned}$$

Damit ist  $p(t) = a_0^n(t)$ .

Ferner gilt für die Ableitung des Bézier-Polynoms  $p'(t) = n \cdot (a_1^{n-1}(t) - a_0^{n-1}(t))$ .

**Beweis:** Wir verwenden die Rekursion (11.4) für die Bernstein-Polynome:

<sup>2</sup>Paul de Faget de Casteljau (geb. 1930 in Besançon); Paris

$$\begin{aligned}
a_i^m(t) &= \sum_{k=0}^m a_{i+k} B_k^m(t) \\
&= \sum_{k=0}^m a_{i+k} (t \cdot B_{k-1}^{m-1}(t) + (1-t) \cdot B_k^{m-1}(t)) \\
&= t \cdot \sum_{k=0}^m a_{i+k} B_{k-1}^{m-1}(t) + (1-t) \cdot \sum_{k=0}^m a_{i+k} B_k^{m-1}(t) \\
&= t \cdot \sum_{k=0}^{m-1} a_{i+1+k} B_k^{m-1}(t) + (1-t) \cdot \sum_{k=0}^{m-1} a_{i+k} B_k^{m-1}(t) \\
&= t \cdot a_{i+1}^{m-1}(t) + (1-t) \cdot a_i^{m-1}(t).
\end{aligned}$$

Für die Ableitung des Bézier-Polynoms ergibt sich mit (11.9)

$$p'(t) = n \sum_{k=0}^{n-1} (a_{k+1} - a_k) B_k^{n-1}(t) = n (a_1^{n-1}(t) - a_0^{n-1}(t)). \quad \square$$

### Tableau von de Casteljau (11.15)

$$\begin{array}{ccccccc}
& & & & & & a_0 \\
& & & & & & a_1 & a_0^1(t) \\
& & & & & & a_2 & a_1^1(t) & \ddots \\
& & & & & & \vdots & \vdots & & a_0^{n-1}(t) \\
& & & & & & a_n & a_{n-1}^1(t) & \dots & a_1^{n-1}(t) & a_0^n(t)
\end{array}$$

### Konkretes Zahlenbeispiel: ( $n = 3$ )

$$\begin{aligned}
p(t) &= 1 \cdot B_0^3(t) + 4 \cdot B_1^3(t) + 3 \cdot B_2^3(t) + 0 \cdot B_3^3(t) \\
&= (1-t)^3 + 12 \cdot (1-t)^2 t + 9 \cdot (1-t) t^2
\end{aligned}$$

Für  $t = 0.4$  (also  $1-t = 0.6$ ) erhält man das Tableau:

$$\begin{array}{ccccccc}
& & & & & & 1 \\
& & & & & & 4 & 2.2 \\
& & & & & & 3 & 3.6 & 2.76 \\
& & & & & & 0 & 1.8 & 2.88 & 2.808
\end{array}$$

Somit ist  $p(0.4) = 2.808$  und  $p'(0.4) = 3 \cdot (2.88 - 2.76) = 0.36$ .

Der de Casteljau–Algorithmus in (11.14) beschreibt in der Tat eine iterierte lineare Interpolation, wobei jeweils zwei bereits konstruierte benachbarte Punkte im gleichen Verhältnis  $t$  unterteilt werden.

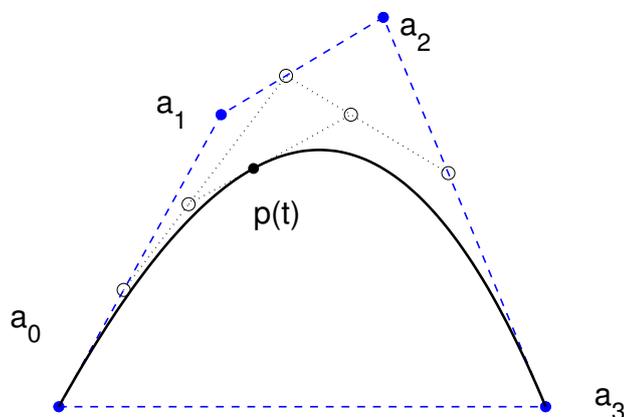


Abb. 11.4 Konstruktion von de Casteljau für  $t = 0.4$

Iteriert man diese Konstruktion, wie in Abb. 11.4 angedeutet, so erhält man einen Punkt des Bézier-Polynoms und die Tangente in diesem Punkt. Damit hat man nun einen sehr einfachen geometrischen Werkzeug, um geeignete Kurven zu konstruieren. Diese Hilfsmittel wird vielfach im 'Computer Aided Geometric Design' (CAGD) eingesetzt.

## Bézier–Flächen.

Unter einer *Parameterdarstellung einer Fläche* im  $\mathbb{R}^3$  versteht man eine stetig differenzierbare Abbildung

$$\Phi : [0, 1]^2 \rightarrow \mathbb{R}^3, \quad (u, v)^T \mapsto \mathbf{x} = \Phi(u, v).$$

Zur Approximation von Flächen verwendet man häufig komponentenweise und lokal Polynomräume in zwei Variablen

$$\Pi_{n,m} := \left\{ p(u, v) = \sum_{i=0}^n \sum_{j=0}^m a_{ij} u^i v^j : u, v \in \mathbb{R} \right\}. \quad (11.16)$$

Offensichtlich ist  $\Pi_{n,m}$  ein  $(n + 1) \cdot (m + 1)$ –dimensionaler  $\mathbb{R}$ –Vektorraum und man sieht unmittelbar, dass die Produktpolynome

$$B_k^n(u) \cdot B_\ell^m(v), \quad 0 \leq k \leq n, \quad 0 \leq \ell \leq m,$$

eine Basis von  $\Pi_{n,m}$  bilden.

Jedes Vektorpolynom  $\mathbf{p}(u, v) \in \mathbb{R}^3$  besitzt daher eine eindeutig bestimmte *Bézier-Darstellung*

$$\mathbf{p}(u, v) = \sum_{i=0}^n \sum_{j=0}^m \mathbf{a}_{ij} \cdot B_i^n(u) \cdot B_j^m(v). \quad (11.17)$$

Wiederum heißen die Koeffizienten  $\mathbf{a}_{ij} \in \mathbb{R}^3$  die *Bézier-Punkte* von (11.17) und die durch die Parameterdarstellung  $\mathbf{x} = \mathbf{p}(u, v)$  definierte Fläche heißt *Bézier-Fläche*.

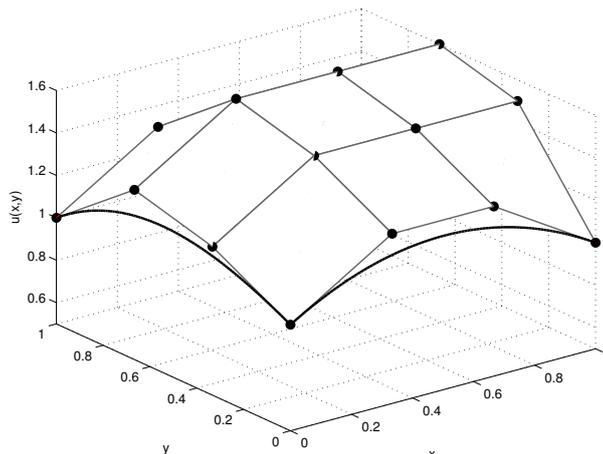


Abb. 11.5 Bikubische Bézier-Fläche im  $\mathbb{R}^3$

Die Berechnung der Flächenpunkte  $\mathbf{p}(u, v)$  bei vorgegebenen Parametern  $(u, v)$  erfolgt durch iterative Verwendung des Algorithmus von de Casteljau für den eindimensionalen Fall. Dazu schreibt man (11.17) wie folgt um

$$\begin{aligned} \mathbf{p}_i(v) &= \sum_{j=0}^m \mathbf{a}_{ij} \cdot B_j^m(v), \quad i = 0, \dots, n, \\ \mathbf{p}(u, v) &= \sum_{i=0}^n \mathbf{p}_i(v) \cdot B_i^n(u). \end{aligned} \quad (11.18)$$

Jeder dieser Ausdrücke ist ein eindimensionales Bézier-Polynom zu den Bézier-Punkten  $\mathbf{a}_{i0}, \dots, \mathbf{a}_{im}$ , bzw.  $\mathbf{p}_0(v), \dots, \mathbf{p}_n(v)$ . Es sind also zur Auswertung von (11.17) jeweils  $(n+2)$  eindimensionale Bézier-Polynome im  $\mathbb{R}^3$  mittels (11.14) zu berechnen.

Im Folgenden gehen wir noch auf kurz auf zwei wichtige Eigenschaften von Bézier-Flächen ein.

#### Konvexe Hüllen Eigenschaft:

Es gelten  $B_i^n(u) \cdot B_j^m(v) \geq 0$  sowie  $\sum_{i=0}^n \sum_{j=0}^m B_i^n(u) \cdot B_j^m(v) = 1$ .

Damit ist  $\mathbf{p}(u, v) = \sum_{i=0}^n \sum_{j=0}^m \mathbf{a}_{ij} \cdot B_i^n(u) \cdot B_j^m(v)$  eine Konvexkombination der Bézier-Punkte  $\mathbf{a}_{ij}$ . Die Bézier-Fläche  $\mathbf{x} = \mathbf{p}(u, v)$  verläuft also ganz in der konvexen Hülle der Bézier-Punkte.

### Partielle Ableitungen:

Analog zum eindimensionalen Fall (vgl. Satz (11.14)) lassen sich die partiellen Ableitungen der Bézier-Fläche sehr leicht mit Hilfe des Algorithmus von de Casteljau berechnen. Es gilt nämlich

$$\begin{aligned} \frac{\partial}{\partial u} \mathbf{p}(u, v) &= n \cdot \sum_{i=0}^{n-1} \sum_{j=0}^m (\mathbf{a}_{i+1,j} - \mathbf{a}_{ij}) B_i^n(u) B_j^m(v) \\ \frac{\partial}{\partial v} \mathbf{p}(u, v) &= m \cdot \sum_{i=0}^n \sum_{j=0}^{m-1} (\mathbf{a}_{i,j+1} - \mathbf{a}_{ij}) B_i^n(u) B_j^m(v). \end{aligned} \quad (11.19)$$

Insbesondere lässt sich hiermit ein Einheits-Normalenvektor an die Bézier-Fläche wie folgt berechnen

$$\mathbf{n}(u, v) = \frac{\mathbf{p}_u(u, v) \times \mathbf{p}_v(u, v)}{\|\mathbf{p}_u(u, v) \times \mathbf{p}_v(u, v)\|}. \quad (11.20)$$

## B-Spline Kurven und Flächen.

Ganz ähnliche Darstellungen und Berechnungsmethoden erhält man, wenn man in den Gleichungen (11.11) und (11.17) die Bernstein-Polynome durch B-Splines zu einem festen Gitter

$$t_{-m} < t_{-m+1} < \dots < t_0 < \dots < t_n < \dots < t_{n+m}$$

ersetzt. Die zu (11.11) analoge Darstellung lautet dann

$$\mathbf{p}(t) = \sum_{j=-m}^{n-1} \mathbf{a}_{j+m} B_{mj}(t). \quad (11.21)$$

Wieder heißen die  $\mathbf{a}_0, \dots, \mathbf{a}_{m+n-1}$  *Kontrollpunkte* (manchmal auch *de Boor-Punkte*) der B-Spline Kurve (11.21).

Wegen  $B_{m,j}(t) \geq 0$  und  $\sum_{j=-m}^{n-1} B_{m,j}(t) = 1$ , vgl. (5.25) und (5.26), erfüllen B-Spline Kurven (und analog B-Spline Flächen) ebenfalls die Konvexe Hüllen Eigenschaft  $\mathbf{p}(t) \in \text{conv}(\mathbf{a}_0, \dots, \mathbf{a}_{m+n-1})$ .

Die einfache Berechnung mit Hilfe des de Casteljau-Algorithmus lässt sich ebenfalls auf B-Spline Kurven und Flächen übertragen. Anstelle der Dreiterm-Rekursion (11.4) für die Bernstein-Polynome verwendet man hierzu die Dreiterm-Rekursion (5.23) für die B-Splines.

Gegenüber den Bézier-Kurven und Fläche haben die B-Spline Kurven und Flächen einen wichtigen Vorteil, der sich aufgrund der kompakten Träger der B-Splines ergibt, vgl. (5.25): Fehler in einem Kontrollpunkt  $\mathbf{a}_{j+m}$  beeinflussen die B-Spline Kurve  $\mathbf{p}(t)$  nur lokal, nämlich im Intervall  $t_j \leq t \leq t_{j+m+1}$ , dem Träger von  $B_{m,j}$ .