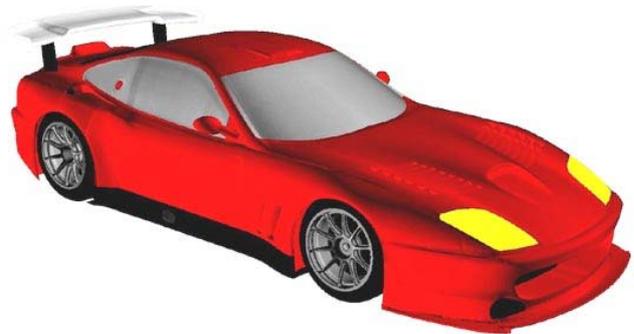
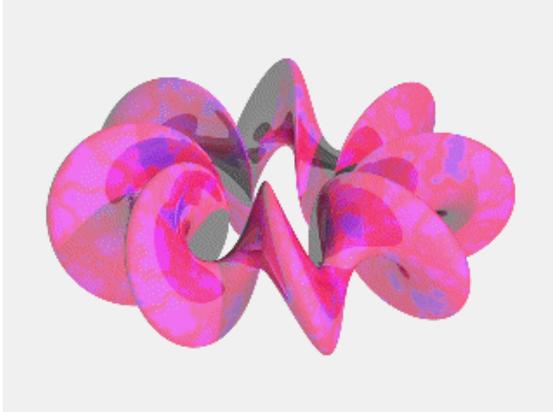


Kurven und Flächen

Viele Anwendungen der graphischen Datenverarbeitung benötigen neben den einfachen Elementen auch die Möglichkeit, beliebige allgemeine Flächen (*Freiformflächen*) zu modellieren und darzustellen. Es werden zuerst an Hand von Freiformkurven die Prinzipien erläutert, da hier die Mathematik etwas einfacher ist. Die dabei entwickelten Verfahren lassen sich leicht auf Flächen erweitern.



■ Kurven

Allgemeine Kurven können entweder durch eine *Formel* (analytische Darstellung) definiert werden oder durch Angabe einer Menge von *Stützpunkten* oder von *Kontrollpunkten*, die den Verlauf der Kurve bestimmen. Da eine analytische Definition für den Benutzer viel schwerer verständlich ist, wird normalerweise die zweite Methode verwendet.

Folgende Eigenschaften charakterisieren die verschiedenen Kurventypen:

- *interpolierend* (Kurve verläuft durch ihre *Stützpunkte*) versus *approximierend* (*Kontrollpunkte* liegen neben der Kurve)
- *Stetigkeitsgrad* an Verbindungen
- *globaler* Einfluss (alle Punkte beeinflussen jeden Kurvenpunkt) versus *lokaler* Einfluss (Punkte beeinflussen nur nahe Kurventeile)
- *achsenabhängige* Darstellung (Drehung des Koordinatensystems verändert Kurve) versus *achsenunabhängige* Darstellung (Drehung des Koordinatensystems verändert Kurve nicht)
- Tendenz zur *Dämpfung* an Ecken versus Tendenz zum *Überschwingen*
- mögliche Kurvenformen, Einschränkungen, doppelte Punkte, geschlossene Kurven usw.

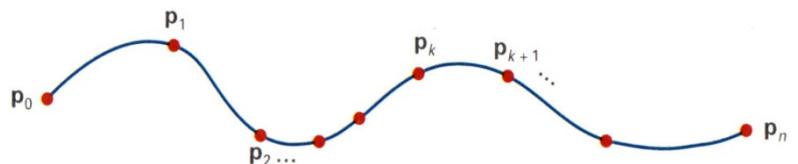
Kurven, die durch Stütz- oder Kontrollpunkte definiert sind, nennt man auch *Splines*. Im Folgenden werden einige einfache gebräuchliche Spline-Verfahren beschrieben.

Kubische Spline-Interpolation

Gegeben sind $n+1$ Stützpunkte $\mathbf{p}_k = (x_k, y_k, z_k)$, $k=0, \dots, n$. Eine Interpolationskurve, die zwischen je 2 Stützpunkten aus einem kubischen Polynom besteht, nennt man kubischen Spline. Zwischen Stützpunkten \mathbf{p}_k und \mathbf{p}_{k+1} beschreibt ein Parameter u die Kurve:

$$\mathbf{P}_k(u) = \mathbf{a}_k u^3 + \mathbf{b}_k u^2 + \mathbf{c}_k u + \mathbf{d}_k$$

mit $k = 0, 1, 2, \dots, n-1$ und $0 \leq u \leq 1$



(Achtung: $\mathbf{a}_k, \mathbf{b}_k, \mathbf{c}_k, \mathbf{d}_k$ sind dabei Vektoren).

Um ein Kurvenstück zwischen 2 Stützpunkten zu berechnen braucht man 4 Angabestücke. Wählt man die Definition an den Stützpunkten dabei so, dass die kubischen Polynome dort sowohl C^1 -stetig (differenzierbar) als auch C^2 -stetig (2x differenzierbar, also gleiche Krümmung) verbunden sind, so spricht

man von *natürlichen kubischen Splines*. Diese erhält man wenn man ein Gleichungssystem mit $4n$ Variablen löst und dabei an den Rändern 2 Nebenbedingungen vorgibt, z.B. Krümmung am Anfang und am Ende ist null. Kubische Splines haben den Nachteil, dass jeder Stützpunkt einen Einfluss auf den gesamten Kurvenverlauf hat, also globalen Einfluss.

Hermite-Interpolation

Eine spezielle Form der kubischen Splines ist die Hermite-Interpolation. Dabei werden neben den Stützpunkten \mathbf{p}_k auch die Ableitungen \mathbf{Dp}_k an den Stützpunkten vorgegeben. Das kubische Interpolationspolynom $\mathbf{P}_k(u)$, $0 \leq u \leq 1$, zwischen Punkten \mathbf{p}_k und \mathbf{p}_{k+1} lässt sich dann aus den 4 Bestimmungsstücken eindeutig berechnen:

$\mathbf{P}_k(0) = \mathbf{p}_k$, $\mathbf{P}_k(1) = \mathbf{p}_{k+1}$, $\mathbf{P}'_k(0) = \mathbf{Dp}_k$, $\mathbf{P}'_k(1) = \mathbf{Dp}_{k+1}$ für $k = 0, \dots, n-1$
 $\mathbf{P}_k(u) = \mathbf{a}_k u^3 + \mathbf{b}_k u^2 + \mathbf{c}_k u + \mathbf{d}_k$ lässt sich auch in Matrixschreibweise anschreiben, ebenso die erste Ableitung dieser Kurve $\mathbf{P}'_k(u) = 3\mathbf{a}_k u^2 + 2\mathbf{b}_k u + \mathbf{c}_k$. Daraus kann man die 4 Bestimmungsstücke \mathbf{p}_k , \mathbf{p}_{k+1} , \mathbf{Dp}_k , \mathbf{Dp}_{k+1} formulieren:

$$\mathbf{P}_k(u) = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{a}_k \\ \mathbf{b}_k \\ \mathbf{c}_k \\ \mathbf{d}_k \end{bmatrix} \quad \mathbf{P}'_k(u) = \begin{bmatrix} 3u^2 & 2u & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{a}_k \\ \mathbf{b}_k \\ \mathbf{c}_k \\ \mathbf{d}_k \end{bmatrix} \quad \begin{bmatrix} \mathbf{p}_k \\ \mathbf{p}_{k+1} \\ \mathbf{Dp}_k \\ \mathbf{Dp}_{k+1} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{a}_k \\ \mathbf{b}_k \\ \mathbf{c}_k \\ \mathbf{d}_k \end{bmatrix}$$

Um die Koeffizienten \mathbf{a}_k , \mathbf{b}_k , \mathbf{c}_k , \mathbf{d}_k von $\mathbf{a}_k u^3 + \mathbf{b}_k u^2 + \mathbf{c}_k u + \mathbf{d}_k$ zu berechnen, invertiert man diese Matrix, die resultierende Matrix heißt Hermite-Matrix \mathbf{M}_H :

$$\begin{bmatrix} \mathbf{a}_k \\ \mathbf{b}_k \\ \mathbf{c}_k \\ \mathbf{d}_k \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix}^{-1} \cdot \begin{bmatrix} \mathbf{p}_k \\ \mathbf{p}_{k+1} \\ \mathbf{Dp}_k \\ \mathbf{Dp}_{k+1} \end{bmatrix} = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{p}_k \\ \mathbf{p}_{k+1} \\ \mathbf{Dp}_k \\ \mathbf{Dp}_{k+1} \end{bmatrix} \quad \mathbf{P}_k(u) = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \cdot \mathbf{M}_H \cdot \begin{bmatrix} \mathbf{p}_k \\ \mathbf{p}_{k+1} \\ \mathbf{Dp}_k \\ \mathbf{Dp}_{k+1} \end{bmatrix}$$

Bezier-Kurven

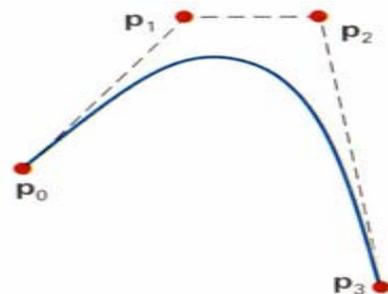
Von Pierre Bezier wurde etwa 1960 für die Beschreibung von Autokarosserien bei Renault eine *approximierende* Kurve erfunden, bei der die sogenannten Bernstein-Polynome ($\text{BEZ}_{k,n}$) als Gewichtsfunktionen für die Kontrollpunkte verwendet werden. Jeder Kurvenpunkt ist das gewichtete Mittel aller Kontrollpunkte:

$$\mathbf{P}(u) = \sum_{k=0}^n \mathbf{p}_k \text{BEZ}_{k,n}(u) \quad 0 \leq u \leq 1 \quad \text{wobei} \quad \text{BEZ}_{k,n}(u) = \binom{n}{k} u^k (1-u)^{n-k}$$

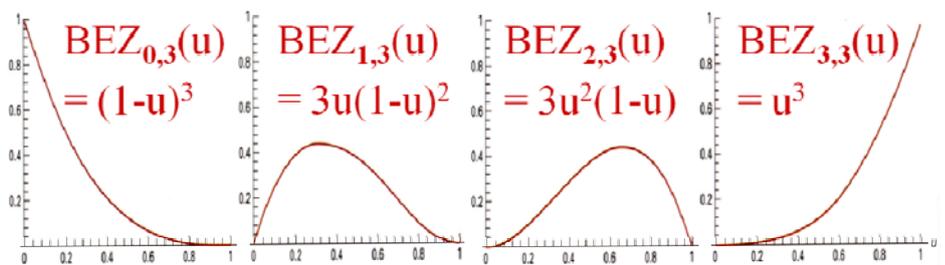
Für 4 Kontrollpunkte (also $n=3$) ergibt sich z.B.
 $\mathbf{P}(u) = (1-u)^3 \cdot \mathbf{p}_0 + 3u(1-u)^2 \cdot \mathbf{p}_1 + 3u^2(1-u) \cdot \mathbf{p}_2 + u^3 \cdot \mathbf{p}_3$

Eigenschaften von Bezier-Kurven:

- bei $n+1$ Kontrollpunkten ist $\mathbf{P}(u)$ vom Grad n
- jeder Kontrollpunkt zieht die Kurve wie mit einem Gummiband an
- *globaler Einfluss* (Gewichtsfunktion fast überall >0)
- \mathbf{p}_0 und \mathbf{p}_n liegen auf der Kurve
- die Tangenten in \mathbf{p}_0 und \mathbf{p}_n sind die Verbindung zu den nächsten Punkten \mathbf{p}_1 und \mathbf{p}_{n-1}
- die Kurve liegt zur Gänze in der konvexen Hülle der Kontrollpunkte (die konvexe Hülle ist das kleinste konvexe Polygon, das alle Kontrollpunkte enthält)

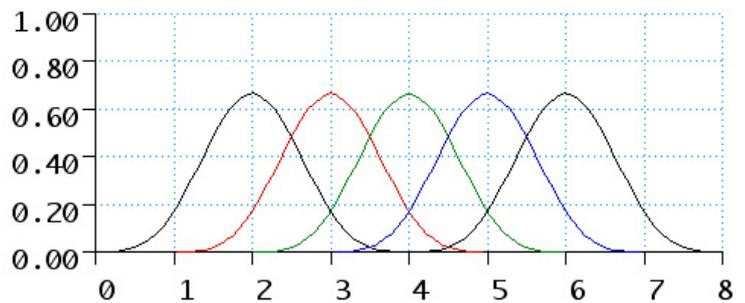


Einen Teil dieser Eigenschaften erkennt man aus der Form der Bernsteinpolynome $BEZ_{k,n}$:



B-Spline-Kurven

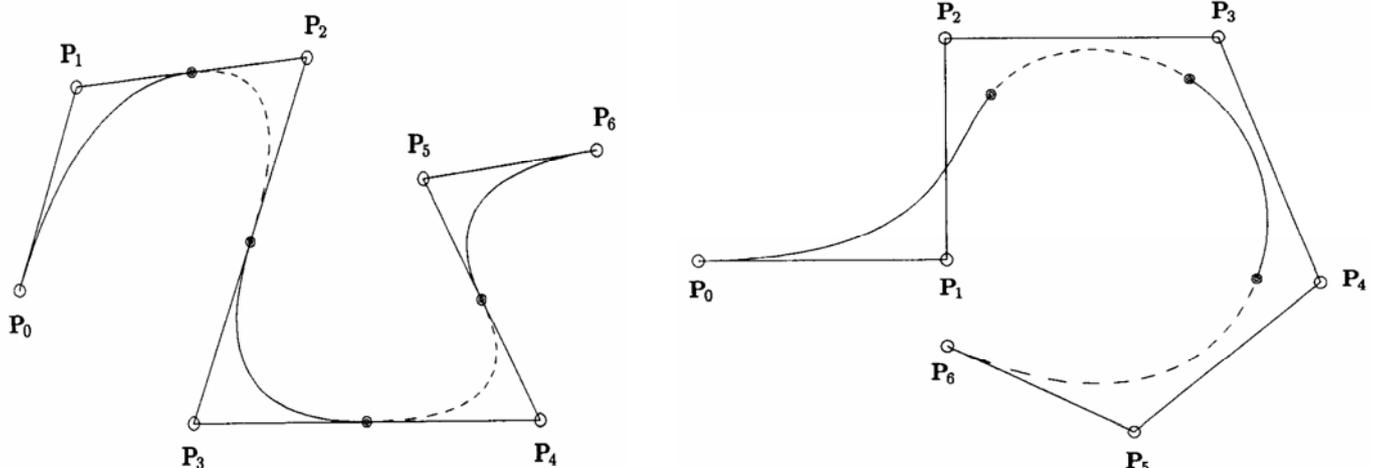
Der Hauptnachteil der Bezierkurven ist der globale Einfluss der Kontrollpunkte auf die ganze Kurve. Dies hat zwei Hauptnachteile: (1) jede Veränderung der Kontrollpunkte (Einfügen, Entfernen, Verschieben) verändert das Aussehen der Kurve an allen Stellen, und (2) die Rechenzeit für große Kontrollpunktmengen ist höher. Die Ursache liegt in der Form der Gewichtsfunktionen. Die sogenannten *B-Splines* sind ebenso wie die Bezier-Splines approximierende Kurven, jedoch sind die Bernsteinpolynome durch B-Spline-Polynome $B_{k,d}$ ersetzt. Diese beschränken die Anzahl der Kontrollpunkte, die einen Kurvenpunkt beeinflussen, auf d . Die Berechnung der $B_{k,d}$ ist etwas komplexer und erfolgt rekursiv, für das Verständnis reicht es allerdings, die Form der B-Spline-Polynome zu sehen. Man erkennt, dass jede Kurve nur in einem begrenzten Bereich ungleich null ist, dass also jeder Punkt über weite Bereiche keinen Einfluss auf die Kurve hat.



Eine wichtige Eigenschaft der B-Spline-Gewichtsfunktionen ist die Tatsache, dass für jeden Kurvenpunkt ihre Summe genau 1 ist. Jeder Kurvenpunkt ist also ein *gewichteter Mittelwert* aus den Kontrollpunkten.

$$\sum_{k=0}^n B_{k,d}(u) = 1$$

Beispiele für B-Spline-Kurven mit $d=3$ (links) und $d=4$ (rechts):



Wenn man $d=n+1$ wählt, dann erhält man Bezier-Kurven, diese sind also ein Spezialfall der B-Splines.

Die Hauptunterschiede zu Bezierkurven sind:

- lokaler Einfluss der Kontrollpunkte
- Aufwand linear zur Anzahl der Kontrollpunkte (statt quadratisch bei Bezierkurven)

Die wichtigsten Erweiterungen dieser sogenannten uniformen B-Splines führen zu den *Non-Uniform Rational B-Splines*, besser bekannt als *NURBS*. Mit diesen können auch regelmäßige geometrische Formen konsistent repräsentiert werden.

Man beachte im Übrigen, dass alle beschriebenen Methoden für Punkte im Zweidimensionalen und im Dreidimensionalen gleichermaßen gilt. Grundsätzlich beschreiben also alle diese Splines räumliche Kurven im dreidimensionalen Raum.

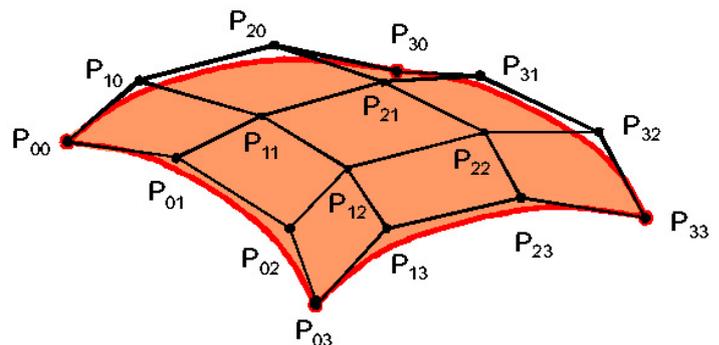
■ Freiformflächen

Bezier- und B-Spline-Flächen

Bildet man über einem zweidimensionalen Punkteraster das kartesische Produkt zweier Kurvenscharen, so erhält man auf natürliche Weise Freiformflächen. Je nach zugrundeliegender Kurvenart erhält man verschiedene Flächen, also etwa Bezierflächen aus Bezierkurven, B-Splineflächen aus B-Splinekurven usw.

$$\mathbf{P}(u, v) = \sum_{j=0}^m \sum_{k=0}^n \mathbf{p}_{j,k} \text{BEZ}_{j,m}(v) \text{BEZ}_{k,n}(u)$$

Jedes Parameterpaar (u,v) führt zu einem Punkt der entstehenden Fläche. Die Randkurven der Fläche sind in diesem Bezier-Beispiel wieder Bezierkurven. Auch die anderen Eigenschaften der Bezierkurven werden auf die Flächen übertragen.



Analog zu diesen Bezierflächen erhält man B-Splineflächen wenn man die B-Spline-Gewichtsfunktionen in die Formel einsetzt, oder etwa NURBS-Flächen für NURBS-Kurven.

Um Freiformflächen zu zeichnen können Dreiecksnetze aus den Flächen erzeugt werden, die wie B-Reps gerendert werden. Oder es werden Ray-Casting-Methoden eingesetzt; dazu muss man eine Prozedur implementieren, die den Schnittpunkt einer Geraden mit der Fläche möglichst genau berechnet.