NURBS

Justin Unger

May 14, 2022, Mosbach

Contents

1 Einleitung **1**

2 Vorbetrachtung **2**
2.1 Bézier-Kurven 2
2.2 Tensor Produkt Oberflächen 4

3 Definition NURBS **6**

4 Modellierung **6**

5 Fazit **6**

6 Anhang **6**
6.1 Umformung Bézier-Kurven mit P_i^w auf Bézier-Kurven mit $w_i * P_i$ 6

1 Einleitung

Produkte haben eine Vielzahl von Phasen, welche sie durchlaufen, bis sie in die Produktion kommen. Besonders bei Autos ist das Design sehr wichtig. Um so komplexe Formen zu generieren, gibt es verschiedene Möglichkeiten. Wenn diese digital modelliert werden, dann müssen diese im Hintergrund mathematisch dargestellt werden.

In dieser Ausarbeitung wird ein Standardverfahren für diese Mathematische Darstellung vorgestellt: *NURBS*. NURBS steht für Non-uniform rational B-Splines. Was diese einzelnen Bestandteile der Abkürzung mit dem Modell zu tun haben, wird im Laufe der Arbeit vorgestellt.

Bevor dies geschehen kann, müssen explizit zwei Vorbetrachtungen zu rate gezogen werden: die *Bézier-Kurven* und die *Tensor Produkt Oberflächen*. Dahingehend wird auch der Rationale Aspekt erläutert.

Anschließend werden im Abschnitt *3 Definition NURBS* die weiteren Bestandteile von NURBS dargestellt. Dabei werden auch B-Splines für sich selbst vorgestellt.

Im Kapitel *4 Modellierung* wird beispielhaft gezeigt, wie in einem Programm mit den NURBS gearbeitet werden kann. Diese Kapitel ist explizit nicht mathematisch. Es soll die Anwendung für Designer und nicht für Mathematiker darstellen.

2 Vorbetrachtung

2.1 Bézier-Kurven

Pierre Etienne Bézier arbeitete bei Renault. Dabei hat er eine mathematische Beschreibung für beliebige Kurven gefunden, welche sich anhand von Kontrollpunkten definieren lassen. Dabei muss die Kurve (Bézier-Kurve) nicht durch jeden Kontrollpunkt verlaufen. Dennoch wird die Form durch die Kontrollpunkte definiert.

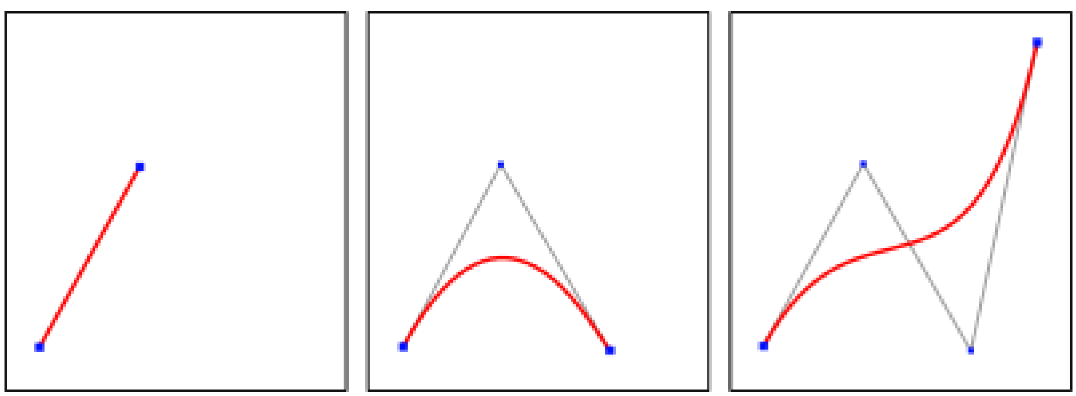

Figure 1: Bezier-Kurven vom 1. bis 3. Grad

Eine Bézier-Kurve hat immer einen bestimmten Grad n. Zudem hat sie $n+1$ Kontrollpunkte P.

$$C(u) = \sum_{i=0}^{n} B_{i,n}(u) * P_i \tag{1}$$

$$B_{i,n}(u) = \frac{n!}{i! * (n-i)!} * u^i * (1-u)^{n-i} \tag{2}$$

$$0 \leq u \leq 1 \tag{3}$$

Mit diesen Gleichungen lassen sich viele zwei dimensionale Formen darstellen. Jedoch nicht alle. Bézier-Kurven sind eine andere Darstellung für Polynome. Die "Grunddarstellung" von Polynomen sind:

$$x(u) = a_0 * u^0 + a_1 * u^1 + a_2 * u^2 + ... + a_n * u^n \tag{4}$$

$$x(u) = b_0 * u^0 + b_1 * u^1 + b_2 * u^2 + ... + b_n * u^n \tag{5}$$

Diese können jedoch keine Kreise, Ellipsen oder Hyperbeln exakt darstellen. Dazu sind gebrochen rationale Polynome (Englisch: rational polynomes) notwendig.

$$x(u) = \frac{X(u)}{W(u)} \tag{6}$$

$$y(u) = \frac{Y(u)}{W(u)} \tag{7}$$

Wenn Bézier-Kurven gebrochen rationale Polynome darstellen sollen, kann folgende die Gleichung verwendet werden. Die Funktion $B(u)$ ist die gleiche wie bei rationalen Bézier-Kurven.

$$C(u) = \sum_{i=0}^{n} R_{i,n}(u) * u_i * P_i \tag{8}$$

$$R_{i,n}(u) = \frac{B_{i,n}(u) * w_i}{\sum_{j=0}^{n} B_{j,n}(u) * w_j} \tag{9}$$

$$B_{i,n}(u) = \frac{n!}{i! * (n-i)!} * u^i * (1-u)^{n-i} \tag{10}$$

Bei $C(u)$ ist der Aufbau auch gleich, außer das $B_{i,n}(u)$ durch $R_{i,n}(u)$ ersetzt wurde. Die Funktion $R_{i,n}(u)$ ist der Bruch der Funktion. Dieser stellt das Verhältnis zwischen dem aktuellen Punkt und der Summe aller Punkte dar. Der Parameter w_i stellt die Gewichtung dar. Wenn die Abbildung 1 erneut betrachtet, kann die Gewichtung als Einflussstärke eines Kontrollpunktes verstanden werden.

Da die Gleichung für Bézier-Kurven immer identisch ist, kann jede mögliche Kurve durch zwei Listen darstellt werden. Eine Liste mit den Kontrollpunkten und eine Liste mit den Gewichtungen für die Punkte. Da jedoch dann immer beachtet werden muss, das beide Listen gleich groß sind, wird eine weitere Darstellungsmöglichkeit für die Punkte gewählt. Dabei werden die Punkte um eine Dimension erweitert. Zwei dimensionale Punkte werden zu drei dimensionalen Punkten.

$$P = (x, y) \tag{11}$$

$$P^w = (w * x, w * y, w) \tag{12}$$

$$H\{P^w\} = P \tag{13}$$

Dadurch ist die Gewichtung direkt mit im Punkt mit angegeben. Diese Umwandlung nennt sich Homogene Koordinaten. Die Geometrische Bedeutung ist das Zuweisen (Mappen) einer $n + 1$-dimensionalen Kurve auf eine n-dimensionale Kurve. Grafisch dargestellt ist dieses Mappen in einem $n = 2$ dimensionalen Beispiel in Figur 2. Die blaue Kurve ist die Kurve in einem n+1-Raum welche durch die Punkte P^w definiert werden. Die rote Kurve hingegen ist die Kurve, welche durch das Mappen generiert wird. Diese wird durch die Punkte P und die Gewichtungen w definiert.

Bézier-Kurven können an diese Punkte entsprechend angepasst werden, sodass sie nicht erst zurückgemapped werden müssen, sondern direkt die gewichteten Punkte P^w verwenden können.

$$C^w(u) = \sum_{i=0}^{n} B_{i,n}(u) * P_i^w \tag{14}$$

Diese Formel hat die exakt gleiche Bedeutung wie die gebrochen rationalen Bézier-Kurven. Der Beweis ist im Anhang vermerkt.

2.2 Tensor Produkt Oberflächen

Wie der Name bereits andeutet, sind Bézier-Kurven nur Kurven. Damit lassen sich noch keine Oberflächen oder Volumen-Körper darstellen. Dies ist mit *Tensor Produkt Oberflächen* (im Weiteren TPO) möglich Diese können sich wie Bézier-Kurven in zwei Richtungen vorgestellt werden. Diese TPOs werden durch das Symbol $S(u, v)$ dargestellt. Dabei sind u und v Elemente

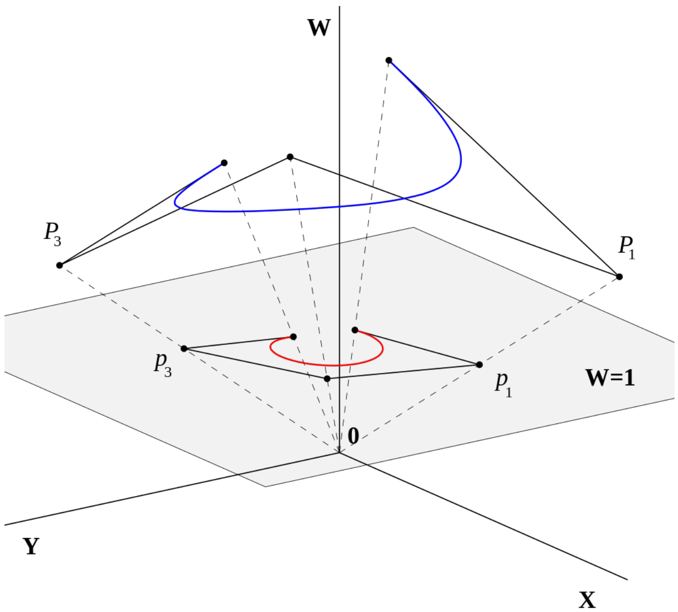

Figure 2: Mapping bei homogenen Koordinaten

einer Fläche im Koordinaten-System. Über dieser Fläche wird die TBO abgebildet.

$$S(u,v) = (x(u,v), y(u,v), z(u,v)) = \sum_{i=0}^{n} \sum_{j=0}^{m} f_i(u) * g_j(v) * b_{i,j} \tag{15}$$

Diese Gleichung kann auch als Matrix-Multiplikation verstanden werden. Statt mit i und j durch zu iterieren, kann eine nXm ermittelt werden, welche in das Koordinaten-System eingetragen werden kann. Dabei ist $[f_i(u)]^T$ ein Reihenvektor der Größe $(1)X(n+1)$. Dagegen ist $[g_j(u)]$ ein Spaltenvektor der Größe $(m+1)X(1)$. Die Punkte $[b_{i,j}]$ sind bereits in einer Matrix der Größe $(n+1)X(m+1)$ angeordnet. Somit verändert sich die Gleichung wie folgt:

$$S(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} f_i(u) * g_j(v) * b_{i,j} = [f_i(u)]^T * [b_{i,j}] * [g_j(u)] \tag{16}$$

Wenn diese allgemeine Form der TPO mit Bézier-Kurven als $f_i(u)$ und $g_j(v)$ verwendet werden, kann nahezu jede Oberfläche dargestellt werden.

$$S(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} B_{i,n}(u) * B_{j,m}(v) * P_{i,j} \tag{17}$$

Für gebrochen rationale Bézier-Kurven können erneut die homogenen Koordinaten verwendet werden, sodass folgende Gleichung entsteht.

$$S^w(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} B_{i,n}(u) * B_{j,m}(v) * P_{i,j}^w \tag{18}$$

Mit dieser Gleichung können alle Oberlfächen dargestellt werden. Jedoch bietet diese Gleichung das Problem, dass sie sehr schnell zu sehr hohen Graden (große n und m) von Bézier-Kurven ausarten. Dadurch ist die Berechnung sehr kompliziert und rechenintensiv. Eine Lösung dafür wird im folgenden Kapitel dargestellt.

3 Definition NURBS

4 Modellierung

5 Fazit

6 Anhang

6.1 Umformung Bézier-Kurven mit P_i^w auf Bézier-Kurven mit $w_i * P_i$

$$C^w(u) = \sum_{i=0}^{n} B_{i,n}(u) * P_i^w$$

$$X(u) = \sum_{i=0}^{n} B_{i,n}(u) * w_i * x_i$$

$$Y(u) = \sum_{i=0}^{n} B_{i,n}(u) * w_i * y_i$$

$$Z(u) = \sum_{i=0}^{n} B_{i,n}(u) * w_i * z_i$$

$$W(u) = \sum_{i=0}^{n} B_{i,n}(u) * w_i$$

$$x(u) = \frac{X(u)}{W(u)} = \frac{\sum_{i=0}^{n} B_{i,n}(u) * w_i * x_i}{\sum_{i=0}^{n} B_{i,n}(u) * w_i}$$

$$y(u) = \frac{Y(u)}{W(u)} = \frac{\sum_{i=0}^{n} B_{i,n}(u) * w_i * y_i}{\sum_{i=0}^{n} B_{i,n}(u) * w_i}$$

$$z(u) = \frac{Z(u)}{W(u)} = \frac{\sum_{i=0}^{n} B_{i,n}(u) * w_i * z_i}{\sum_{i=0}^{n} B_{i,n}(u) * w_i}$$

$$C(u) = (x(u), y(u), z(u)) = \frac{\sum_{i=0}^{n} B_{i,n}(u) * w_i * (x_i, y_i, z_i)}{\sum_{i=0}^{n} B_{i,n}(u) * w_i} = \frac{\sum_{i=0}^{n} B_{i,n}(u) * w_i * P_i}{\sum_{i=0}^{n} B_{i,n}(u) * w_i}$$