Chapter 6

Bezier Curves and Surfaces

6.1 Curves

A curve is a path through space.

From a computer graphics modeling point of view, the advantage of a curve is using a small number of points to define a complex object.

6.1.1 Control Points and Blending Functions

Curves are often defined parametrically, i.e, as the path of a point over time. As an example a curve could be defined using the polynomials

$$\begin{aligned} x(t) &= 3t^3 - 4t + 2 \\ y(t) &= -t^2 + t - 3 \\ z(t) &= t - 1 \end{aligned}$$

But it is very nonintuitive what this curve would look like. It is also unclear in general what changing the coefficients of the polynomials would do to the shape of the curve.

A common restriction of the problem is to take a small set of points, called *control points* that are multiplied by functions and summed for a final result. For example, using the control points P_0, P_1, and P_2, each with their own x, y, and z coordinates,

$$\begin{aligned} x(t) &= x_0(1-t)^2 + x_1 t^2 - 3x_2 t \\ y(t) &= y_0(t+1) + y_2 t \\ z(t) &= z_0(t^2 - 3t + 7) + z_1(\sin(t)) \end{aligned}$$

Another common restriction is to use a small set of functions, called *blending functions* which are multiplied equally by all of the coordinates of a particular control point. The curve is then the sum of the control points multiplied by the blending functions, i.e.,

$$P(t) = \sum P_i f_i(t) = P_0 f_0(t) + P_1 f_1(t) + \cdots + P_n f_n(t)$$

Remember that this formula is evaluated simultaneously in x, y, and z. (In fact there is no limit on the number of coordinates that can be assigned to a control point, making it possible to define curves of high dimension.)

The blending functions often come in sets that are associated with the number of control points involved. The number of control points (or blending functions) is usually called the *degree* of the curve (with an adjustment by 1).

6.1.2 Desirable Properties

Many different types of blending functions have been studied. There is no set of blending functions that works well in all situations. However, some desirable properties for controlling curves have been identified that are satisfied with varying degrees of success by different blending functions.

1. Interpolation
2. Convex Hull
3. Linear Independence
4. Variation Diminishing
5. Coordinate System Independence
6. Symmetry
7. and more

6.2 Lagrange Curves

Lagrange curves give a way of finding polynomial curves that exactly interpolate a set of points. The set of points $P_0 ... P_n$ determine a curve of degree n. Associated with each control point is a specific parameter value of t, $t_0 ... t_n$. For each degree n there are $n+1$ Lagrange basis functions of the form

$$L_i^n(t) = \frac{(t-t_0)(t-t_1)\cdots(t-t_{i-1})(t-t_{i+1})\cdots(t-t_n)}{(t_i-t_0)(t_i-t_1)\cdots(t_i-t_{i-1})(t_i-t_{i+1})\cdots(t_i-t_n)} = \prod_{k=0,k\neq i}^{k=n} \frac{(t-t_k)}{(t_i-t_k)}$$

This basis function has the property that it has the value of 1 whenever $t = t_i$, and the value of 0 for any other of the t control values used in the curve.

The curve itself takes the standard form for curves using control points and basis functions:

$$P(t) = \sum_{i=0}^{n} P_i L_i^n(t)$$

Lagrange curves do not satisfy the convex hull property or the variation diminishing property. They wiggle too much. For example, the Langrange curve going through a set of control points all in a straight line will be a wiggly curve.

6.3 Bezier Curves

Developed independently by Pierre Bezier - Rénault Pierre de Casteljau - Citroën DeCasteljau developed it first, but didn't publish. Bezier published, and so has his name associated with the curves. Bezier curves are founded on the Bernstein basis polynomials.

6.3.1 Bernstein Basis Polynomials

The Bernstein basis polynomials can be developed by looking at the powers of 1, more specifically by looking at the powers of $[(1-t)+t]^n$.

By using the binomial theorem, this expression can be expanded to

$$[(1-t)+t]^n = \binom{n}{0}(1-t)^n t^0 + \binom{n}{1}(1-t)^{n-1}t^1 + \cdots + \binom{n}{i}(1-t)^{n-i}t^i + \cdots + \binom{n}{n}(1-t)^0 t^n$$

where $\binom{n}{i}$ is the binomial coefficient for the term and can be expressed by the formula $n!/i!(n-i)!$.

6.3.2 Curve Evaluation

DeCasteljau's Algorithm

6.3.3 Curve Subdivision

6.3.4 Degree Elevation

Given the control points P_i of a Bezier curve P of degree n, a new curve Q can be created which has degree $n+1$ and looks identical to the original curve. This process is called $d$egree elevation. The control points of Q_i of the degree elevated curve are created using the following formula:

Important Point

For degree elevation:

$$Q_i = \frac{n+1-i}{n+1}P_i + \frac{i}{n+1}P_{i-1}; \qquad 0 \le i \le n+1$$

or

$$Q_0 = P_0; \qquad Q_{n+1} = P_n; \qquad Q_i = P_{i-1} + \frac{(n+1-i)}{(n+1)}(P_i - P_{i-1}); \qquad 1 \le i \le n$$

As an example of degree elevation from degree 2 to degree 3, $Q_0 = P_0$; Q_1 is $\frac{2}{3}$ of the way from P_0 to P_1; Q_2 is $\frac{1}{3}$ of the way from P_1 to P_2; and $Q_3 = P_2$.

Derivation of Degree Elevation

The key to the derivation of a curve of degree $n+1$ which is identical to $P(t)$ is to mulitply it by a special form of 1, namely the quantity $[(1-t)+t]$.

$$\begin{aligned}[(1-t)+t]P(t) &= [(1-t)+t]\sum_{i=0}^{n} P_i B_i^n(t) \\ &= (1-t)\sum_{i=0}^{n} P_i B_i^n(t) + t\sum_{i=0}^{n} P_i B_i^n(t)\end{aligned}$$

Expanding the Bernstein polynomials,

$$P(t) = (1-t)\sum_{i=0}^{n} P_i \binom{n}{i}(1-t)^{n-i}t^i + t\sum_{i=0}^{n} P_i \binom{n}{i}(1-t)^{n-i}t^i$$

Expanding the sums into two rows and collecting the like factors

$$\begin{aligned}P(t) = \; & P_0\binom{n}{0}(1-t)^{n+1} + P_1\binom{n}{1}(1-t)^n t + \ldots + P_n\binom{n}{n}(1-t)t^n \\ & + P_0\binom{n}{0}(1-t)^n t + \ldots + P_{n-1}\binom{n}{n-1}(1-t)t^n + P_n\binom{n}{n}t^{n+1}\end{aligned}$$

The key to recollecting these terms into $B_i^{n+1}(t)$ is to look at combinations involving $n+1$ items instead of n. So it becomes necessary to look at how $\binom{n}{i}$ relates to $\binom{n+1}{i}$ and how $\binom{n}{i-1}$ relates to $\binom{n+1}{i}$.

For the first relationship,

$$\begin{aligned}\binom{n}{i}\Big/\binom{n+1}{i} &= \frac{n!}{i!(n-i)!}\Big/\frac{(n+1)!}{i!(n+1-i)!} \\ &= \frac{n!}{i!(n-i)!}\frac{i!(n+1-i)!}{(n+1)!} = \frac{n+1-i}{n+1}\end{aligned}$$

so

$$\binom{n}{i} = \frac{n+1-i}{n+1}\binom{n+1}{i}$$

In a similar fashion,

$$\binom{n}{i-1} \Big/ \binom{n+1}{i} = \frac{n!}{(i-1)!(n-i+1)!} \Big/ \frac{(n+1)!}{i!(n+1-i)!}$$

$$= \frac{n!}{(i-1)!(n-i+1)!} \frac{i!(n+1-i)!}{(n+1)!} = \frac{i}{n+1}$$

so that

$$\binom{n}{i-1} = \frac{i}{n+1} \binom{n+1}{i}$$

Plugging these expressions into the double row sum for $P(t)$ and reforming the Bernstein polynomials of degree $n+1$,

$$\begin{aligned} P(t) = {} & P_0 B_0^{n+1}(t) + \frac{n}{n+1} P_1 B_1^{n+1}(t) + ... + \frac{1}{n+1} P_n B_n^{n+1}(t) \\ & + \frac{1}{n+1} P_0 B_1^{n+1}(t) + ... + \frac{n}{n+1} P_{n-1} B_n^{n+1}(t) + P_n B_{n+1}^{n+1}(t) \end{aligned}$$

Combining like terms gives the control points Q_i of the elevated curve as noted above.

6.3.5 Derivatives

Given the control points P_i of a Bezier curve P of degree n, the derivative of that curve is given by another Bezier curve of reduced degree $n-1$:

Important Point

For finding the derivative of a Bezier curve:

$$Q_i = n\,(P_{i+1} - P_i)\,; \qquad 0 \leq i < n$$

Derivation of the Derivative

The derivative of a Bezier curve essentially comes down to combining combinations of the derivatives of the Bernstein polynomials, $B_i^n(t)$, i.e.

$$\frac{d}{dt}P(t) = \sum_{i=0}^{n} P_i \frac{d}{dt} B_i^n(t)$$

The derivative of the Bernstein basis function follows from an application of the product rule and chain rule:

$$\frac{d}{dt}\binom{n}{i}(1-t)^{n-i}t^i = \binom{n}{i}\left[i(1-t)^{n-i}t^{i-1} - (n-i)(1-t)^{n-i-1}t^i\right]$$

This derivative is substituted back into the original form and separated into two sums:

$$\frac{d}{dt}P(t) = \sum_{i=1}^{n} P_i \binom{n}{i} i(1-t)^{n-i}t^{i-1} - \sum_{i=0}^{n-1} P_i \binom{n}{i}(n-i)(1-t)^{n-i-1}t^i$$

Note that the range of indices has been changed slightly to eliminate two terms that evaluate to 0, and thus don't affect the sum.

Expanding the sums into two rows and collecting the like factors

$$\begin{array}{rcllllr} \frac{d}{dt}P(t) & = & P_1\binom{n}{1}1(1-t)^{n-1}t^0 & + & \ldots & + & P_n\binom{n}{n}n(1-t)^0t^{n-1} \\ & - & P_0\binom{n}{0}n(1-t)^{n-1}t^0 & - & \ldots & - & P_{n-1}\binom{n}{n-1}1(1-t)^0t^{n-1} \end{array}$$

The generalized term of this sum is

$$\left[P_{i+1}\binom{n}{i+1}(i+1) - P_i\binom{n}{i}(n-i)\right]\left[(1-t)^{n-i-1}t^i\right]$$

Looking at the factors associated with P_{i+1} and P_i in turn,

$$\binom{n}{i+1}(i+1) = \frac{n!}{(i+1)!(n-i-1)!}(i+1) = \frac{n!}{i!(n-1-i)!} = n\frac{(n-1)!}{i!(n-1-i)!} = n\binom{n-1}{i}$$

and

$$\binom{n}{i}(n-i) = \frac{n!}{i!(n-i)!}(n-i) = \frac{n!}{i!(n-i-1)!} = n\frac{(n-1)!}{i!(n-1-i)!} = n\binom{n-1}{i}$$

This makes the generalized term

$$n\left[P_{i+1} - P_i\right]\binom{n-1}{i}(1-t)^{n-i-1}t^i$$

leading to

$$\frac{d}{dt}P(t) = \sum_{i=0}^{n-1} n\left(P_{i+1} - P_i\right)B_i^{n-1}(t)$$

6.4 Hermite Curves

6.5 Bezier Patches

6.5.1 Leaving Attributes Alone

Sometimes it is necessary to assign control point values for particular attributes when in fact none are wanted. For example, a Bezier patch should be implicitly textured from 0 to 1. But the mechanism should exist to assign texture coordinates to control points as needed. So if the mechanism exists, what should the default control point values be when no explicit values are given.

The solution lies in degree elevating a line segment to the appropriate degree. This is simply an equal spacing of the points between 0 and 1, i.e., $P_i = i/n$. where n is the desired degree. What follows is a proof (probably unnecessary) that this is indeed the same as the original line parameterization from 0 to 1, namely $P(t) = t$.

The degree elevated curve, with $P_i = i/n$ is

$$P(t) = \sum_{i=0}^{i=n} \frac{i}{n}\binom{n}{i}(1-t)^{n-i}\,t^i$$

Now the term when $i = 0$ will be zero, so the summation can start at $i = 1$.

Expanding the binomial coefficient,

$$P(t) = \sum_{i=1}^{i=n} \frac{i}{n}\,\frac{n!}{(n-i)!\;i!}\,(1-t)^{n-i}\,t^i$$

and simplifying,

$$P(t) = \sum_{i=1}^{i=n} \frac{(n-1)!}{((n-1)-(i-1))!\,(i-1)!}\,(1-t)^{n-i}\,t^i$$

Now factoring out a t, making a small substitution, $i = j + 1$, and rearranging,

$$P(t) = t \sum_{j=0}^{j=n-1} \binom{n-1}{j} (1-t)^{n-j-1}\,t^j$$

but these are simply the Bernstein polynomials of degree $n - 1$, i.e.,

$$P(t) = t \sum_{j=0}^{j=n-1} B_j^{n-1}(t)$$

However, the Bernstein polynomials of a given degree all sum to 1, so $P(t) = t$.

Important Point

When forced to add uniform attribute values to a Bezier control polygon, increment the value evenly from control point to control point.

6.6 Rational Bezier Curves

6.7 B-Splines

6.8 NURBS