Beziers Curves

Milind Sohoni

http://www.cse.iitb.ac.in/~sohoni
Recall

Lets recall a few things:

1. $f : [0, 1] \rightarrow \mathbb{R}$ is a function.

2. $f_0, \ldots, f_i, \ldots, f_n$ are observations of $f$ with $f_i = f\left(\frac{i}{n}\right)$.

3. $B^n(f) = \sum_i f_i B^n_i(t)$ is a polynomial of degree $n$.

4. The plot of $B^n(f)$ looks like this:

![Plot of Bernstein approximator for n=4](image-url)
A Computation

- $\sum_{i=0}^{n} B_i^n(t) = 1$.

This follows from binomial expansion of

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

Thus for all $t$, $B^n(f)(t)$ is a convex combination of the observations $f_i$.

- $\sum_{i=0}^{n} \frac{i}{n} B_i^n(t) = t$.

This is more delicate. Suppose we choose $f(t)$ as $t$ itself, then $f(\frac{t}{n}) = \frac{i}{n}$. Thus what is being computed is the Bernstein approximation to $f(t) = t$. And what this says is that the approximation $B^n(f)$ is $f$ itself!

*WARNING* This is not true even for $f(t) = t^2$
Computation Continued...

We begin with the expression:

\[ t = \int 1 \, dt = \sum_{i=0}^{n-1} \int B_{i}^{n-1}(t) \, dt \]

Now we solve this, and also eliminate the constant of integration. For this note that

\[ \int B_{i}^{n-1}(t) \, dt = \frac{1}{n} B_{i+1}^{n}(t) + \int B_{i+1}^{n-1}(t) \, dt \]

This easily telescopes into the desired result.
An Alternate Expression

Treating both $y = f(t)$ and $y = B^n(f)(t)$ as curves in $\mathbb{R}^2$, we can give a parametrization:

$$\begin{bmatrix} t \\ B^n(f)(t) \end{bmatrix} = \begin{bmatrix} 0/n & 1/n & \cdots & n/n \\ f_0 & f_1 & \cdots & f_n \end{bmatrix} \begin{bmatrix} B^n_0(t) \\ B^n_1(t) \\ \vdots \\ B^n_n(t) \end{bmatrix}$$

---

original function

bernstein approximator for $n=4$
The Bezier Curve

In general, just as the \( y \)-coordinates were general, we may put general \( x \)-coordinates, instead of \( \frac{i}{n} \) to get:

\[
\begin{bmatrix}
  x(t) \\
y(t)
\end{bmatrix} = \begin{bmatrix}
  x_0 & x_1 & \ldots & x_n \\
y_0 & y_1 & \ldots & y_n
\end{bmatrix} \begin{bmatrix}
  B_0^n(t) \\
  B_1^n(t) \\
  \vdots \\
  B_n^n(t)
\end{bmatrix}
\]

\( p_0=(x_0,y_0) \quad p_3=(x_3,y_3) \)

\( p_1=(x_1,y_1) \quad p_2=(x_2,y_2) \)

\( (x(0.7),y(0.7)) \)
The Bezier Curve: Control Polygon

In general, if we have a sequence \( P = [p_0, \ldots, p_n] \) of points \( p_i = [x_i, y_i] \in \mathbb{R}^2 \), we may define

\[
x(t) = \sum_{i=0}^{n} x_i B_i^n(t)
\]
\[
y(t) = \sum_{i=0}^{n} y_i B_i^n(t)
\]

or in general

\[
p(t) = \sum_{i=0}^{n} p_i B_i^n(t)
\]

\( p(t) \) has nice properties such as \( p(0) = p_0 \), \( p(1) = p_n \) and more.

The sequence \( P = [p_0, \ldots, p_n] \) is called the control polygon.
A New Scheme

This gives us a new paradigm: Draw curves in space via the control polygon.

Bezier Curve: Using the Bernstein Basis and Control Polygons
The Construction of Given Curves

But what about approximation of already given curves?

- Given a curve $C$ in $\mathbb{R}^3$, sample points $P^n = [p_0, \ldots, p_n]$ equi-distant along curve-length.
- Form $P^n(t) = \sum_i p_i B_i^n(t)$.

Theorem: For every $\epsilon > 0$, there is an $n$ such that $P^n(t)$ is within the $\epsilon$-envelop of $C$. 

![Diagram of a curve and its approximations](image)
Beziers Curve Properties

We begin with the expression:

\[ P(t) = p_0 B^n_0(t) + p_1 B^n_1(t) + \ldots + p_n B^n_n(t) \]

- Putting \( t = 0 \), we see that \( B^n_i \) vanish for \( i > 0 \). and out pops \( p_0 \). Thus \( P(0) = p_0 \). Similarly \( P(1) = p_n \). Thus the curve behaves quite predictably at the end-points.

- Next, for any \( t \in [0, 1] \), we have \( B^n_i(t) \geq 0 \) and \( \sum B^n_i(t) = 1 \). Thus the curve \( P(t) \) lies in the convex hull of the control polygon.
Tangents

So given \( P = [p_0, \ldots, p_n] \), and \( P(t) = \sum_i p_i B^n_i(t) \).
What is the meaning of \( P'(t) = \frac{dP}{dt} \)?
\( P(t) = (x(t), y(t)) \) and thus \( P'(t) = (x'(t), y'(t)) \) is the tangent to the curve.

Also recall that \( \frac{dB^n_i(t)}{dt} = n(B^{n-1}_{i-1}(t) - B^{n-1}_i(t)) \).
End Tangents

Back-substituting, we get that:

\[ P'(t) = \sum_{i=0}^{n-1} q_i B_i^{n-1}(t) = \sum_{i=0}^{n-1} n(p_{i+1} - p_i)B_i^{n-1}(t) \]

Thus, the derivative/tangent to \( P(t) \) is a degree \( n - 1 \) bezier curve, whose control points are easily computed.

Whence evaluating \( P'(t) \) at 0, we see that \( P'(0) = q_0 = n(p_1 - p_0) \), i.e.,

\[
\begin{align*}
x'(0) &= n(x_1 - x_0) \\
y'(0) &= n(y_1 - y_0)
\end{align*}
\]

Thus \( P'(0) \), the tangent to the curve at 0 and is given by the line joining \( p_1 \) and \( p_0 \). The slope is clearly \( \frac{y_1 - y_0}{x_1 - x_0} \).
Thus the behaviour of $P(t)$ at the end-points is easily determined from the control polygon: $P(0) = p_0$ and the tangent $P'(0) = (p_1 - p_0)/n$. 
Thus the behaviour of \( P(t) \) at the end-points is easily determined from the control polygon: \( P(0) = p_0 \) and the tangent \( P'(0) = (p_1 - p_0)/n \).

**Caution:** If we just know the image of \( P(t) \), then \( p_0 \) is certainly determined as one of the end-points. From the tangent, we can just guess that \( p_1 \) lies on it.
Splicing

**Question:** Suppose $P$ is a control polygon and $P(t)$ its associated curve. We would like to splice another curve $Q(t)$ which extends $P(t)$ at $p_0$. Then how is the control polygon of $Q$ to be chosen?

**Smooth extension result:** The curve $Q(t)$ smoothly extends $P(t)$ if (i) $p_0 = q_m$ and (ii) $p_1 - p_0$ and $q_m - q_{m-1}$ are co-linear.
**Evaluation: The deCasteljeu Algorithm**

**Question**: How is one to evaluate $P(t)$, given $P = [p_0, \ldots, p_n]$ and the parameter value $t$.

The deCasteljeu scheme is $O(n^2)$, and quite efficient and stable.

Compare with evaluating $P(t) = \sum_{i=0}^{n} p_i \binom{n}{i} t^i (1 - t)^{n-i}$ directly.
The Geometric De-Casteljau

Thus every successive iteration of the algorithm is a sequence of convex combinations of the points generated in the previous phase. The final point $P[n]$ thus is also (as expected) a convex combination of the elements of $P = [p_0, \ldots, p_n]$ and therefore lies in the convex hull of $P$. 
Subdivision

Next consider the curve $C = P[t]$. Suppose that there is a surface $S$ (a plane in this case) which intersects the curve $C$. Suppose that we have determined the intersection point and that it takes the parameter value $c = 0.7$. The ‘useful’ part of the curve is $C''$ which is $C$ restricted to $t \in [0, 0.7]$.

**Question:** How is one to obtain the control points for $C''$ having those of $C$?
In terms of polynomials...

Suppose that \( f : [0, 1] \to \mathbb{R} \) is a polynomial. For a given \( c = 0.7 \), we require another polynomial \( g \) such that \( g(t) = f(ct) \).

Thus \( g(0) = f(0) \) and \( g(1) = f(c) \), and \( g : [0, 1] \to \mathbb{R} \) defines the useful part of \( f \).

If \( f = a_0 + a_1 t^1 + \ldots + a_n t^n \), then

\[
 g(t) = f(ct) = a_0 + (c^1 a_1) t^1 + \ldots + (c^n a_n) t^n
\]

In other words, \( g = b_0 + b_1 t^1 + \ldots + b_n t^n \), where \( b_i = c^i a_i \) for all \( i \). Thus the expression of \( g \) in terms of the Taylor basis is clear when \( f \) is also similarly expressed.

So what happens when \( f(ct) = \sum_{i=0}^{n} p_i B_i^n(ct) \), is expressed in the bernstein basis?
Subdivision in the Bernstein basis

In other words, express $B^n_i(ct)$ in terms of $\{B^n_0(t), \ldots, B^n_n(t)\}$. Trying our hand, we see that:

$$B^n_n(ct) = \binom{n}{n} (ct)^n (1 - ct)^{n-n} = c^n t^n = c^n B^n_n(t)$$

$$B^n_{n-1}(ct) = n (ct)^{n-1} (1 - ct) = nc^{n-1} t^{n-1} [(1 - t) + t(1 - c)]$$

$$= c^n B^n_{n-1} + nc^{n-1} (1 - c) t^n$$

$$= B^n_{n-1}(t) B^n_{n-1}(c) + B^n_n(t) B^n_{n-1}(c)$$

In general, we have:

$$B^n_{n-k}(ct) = \sum_{j=0}^{k} B^n_{n-j}(t) B^n_{n-k-j}(c)$$
Back-Substituting...

Back-substituting that expression into \( P(ct) = \sum_{k=0}^{n} p_{n-k} B_{n-k}^{n}(ct) \), we get:

\[
P(ct) = \sum_{k=0}^{n} p_{n-k} \sum_{j=0}^{k} B_{n-j}^{n}(t) B_{n-k}^{n-j}(c) = \sum_{i=0}^{n} \left[ \sum_{r=0}^{i} p_{r} B_{r}^{i}(c) \right] B_{i}^{n}(t) = \sum_{i=0}^{n} q_{i}(c) B_{i}^{n}(t)
\]

But we know this quantity before...:

![Diagram with states and transitions]
And Geometrically speaking..

Thus, the control points for the sub-division are sitting there in the evaluation process.

Guess what are the control points for the second part of the curve, i.e., from $[c, 1]$?
Degree Elevation

Suppose $C = P[t]$ is given as a degree 3 parametrization. Thus $(x(t), y(t), z(t))$ are degree 3 polynomials. Then certainly, they are expressible in terms of $B^4_i(t)$! What is that expression?

In other words, given $P = [p_0, \ldots, p_n]$ compute $Q = [q_0, \ldots, q_n, q_{n+1}]$ so that:

$$\sum_{i=0}^{n} p_i B^i_n(t) = \sum_{j=0}^{n+1} q_j B^{n+1}_j(t)$$

To begin with, we see:

$$B^i_n(t) = \binom{n}{i} t^i (1 - t)^{n-i}$$

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

$$= \frac{i+1}{n+1} B^{n+1}_{i+1}(t) + \frac{n-i+1}{n+1} B^{n+1}_i(t)$$
Back-substituting, we see that:

\[ q_i = \frac{n - i + 1}{n + 1} p_i + \frac{i}{n + 1} p_{i-1} \]

The coefficients have the following pleasing interpretation:
Back-substituting, we see that:

\[ q_i = \frac{n - i + 1}{n + 1} p_i + \frac{i}{n + 1} p_{i-1} \]

The coefficients have the following pleasing interpretation:
And Geometrically speaking..

Thus, the control polygon $Q$ may be obtained as an appropriate interpolation of the control-polygon $P$. 
Wrap-Up

Recall that the edge geometry is stored as the tuple:

- an interval \([a, b]\), in this case \([0, 1]\).
- a map \(f : [a, b] \to \mathbb{R}^3\), in this case, as a sequence \(P = [p_0, \ldots, p_n]\).

Further:

1. The evaluation of \(f\) is given by the deCasteljau algorithm. Also note that \(f\) is a polynomial and is thus defined beyond \([0, 1]\).

2. The subdivision and elevation are basic kernel operations of modifying the function \(f\) to suit requirements.

3. The construction of a particular curve from points on it is enabled via the Bezier-Bernstein theorem.