Lecture 3: Bézier Curves II [1]

The points on a Bézier curve can be computed using a recursive algorithm due to de Casteljau.

de Casteljau algorithm

Lets assume that we have three control points $\mathbf{p}_0, \mathbf{p}_1$ and $\mathbf{p}_2$ to define a second degree Bézier curve. Suppose we want to compute the point $\mathbf{p}(u)$ for the parameter value u.

For this we first compute the point $\mathbf{x}$ on $\mathbf{p}_0\mathbf{p}_1$ given by $\mathbf{x} = \mathbf{p}_0 + u(\mathbf{p}_1 - \mathbf{p}_0)$. Compute the point $\mathbf{y} = \mathbf{p}_1 + u(\mathbf{p}_2 - \mathbf{p}_1)$. The point $\mathbf{z} = \mathbf{x} + u(\mathbf{y} - \mathbf{x})$ lies on the curve.

To see this expand the equation for $\mathbf{z}$.

$$\begin{aligned}
\mathbf{z} &= \mathbf{x} + u(\mathbf{y} - \mathbf{x}) \\
&= \mathbf{p}_0 + u(\mathbf{p}_1 - \mathbf{p}_0) + u((\mathbf{p}_1 + u(\mathbf{p}_2 - \mathbf{p}_1)) - \mathbf{p}_0 - u(\mathbf{p}_1 - \mathbf{p}_0)) \\
&= (1-u)^2\mathbf{p}_0 + 2u(1-u)\mathbf{p}_1 + u^2\mathbf{p}_2
\end{aligned}$$

So, recursively we can express the algorithm as:

Casteljau($\mathbf{p}_0, ..., \mathbf{p}_n$)
    if ($n = 0$) output $\mathbf{p}_0$;
    else
        Compute n points $\mathbf{p}'_0, ..., \mathbf{p}'_{n-1}$
        where $\mathbf{p}'_i = \mathbf{p}_i + u(\mathbf{p}_{i+1} - \mathbf{p}_i)$
        Call Casteljau($\mathbf{p}'_0, ..., \mathbf{p}'_{n-1}$)
    endif
end

Show an example with a cubic Bézier curve.

Basis functions

We already know that the curve is inside the convex hull of the control points since $\Sigma_{i=0}^{n} B_{i,n}(u) = 1$ and each basis function is nonnegative everywhere in the interval $[0, 1]$. This gives an easy way to quickly determine if two curves are not intersecting. A min-max rectangle is fit around the control polygon and if the two such rectangles do not intersect, the curves inside cannot intersect.

Moving points. See the plots of the basis functions given in the book, page 92. We observe the following. $B_{0,n}$ and $B_{n,n}$ passes through $u = 1$ and $u = 0$ respectively. $B_{0,n}$ is the most influential at $u = 0$ since all other basis functions vanish there. Similarly, $B_{n,n}$ is most influential at $u = 1$. Thus control points $\mathbf{p}_0$ and $\mathbf{p}_n$ are most infulential at $u = 0$ and $u = 1$ respectively. It can be calculated that each $B_{i,n}$ reaches its maximum at $u = i/n$. So, $\mathbf{p}_i$ exerts its influence most at $u = i/n$. So, changing $\mathbf{p}_i$ will have most effect around $u = i/n$.

See the picture on page 94 to see the effect of moving a point. We can specify multiple points at the same position. This will pull the curve towards that point.

[1]Note by Tamal K. Dey

Closed curves. If the first and last points of the control polygon coincide we have a closed curve. If the edges $\mathbf{p}_1\mathbf{p}_0$ and $\mathbf{p}_{n-1}\mathbf{p}_n$ are collinear with $\mathbf{p}_0 = \mathbf{p}_n$ the curve will have G^1 continuity.

Degree elevation

We can increase the degree of a curve by adding more points. This might be desired to achieve more freedom in approximating a shape. But, initially we would like to elevate the degree without changing the shape. Then, we can move points to change the shape.

So, we want

$$\Sigma_{i=0}^{n+1} \mathbf{p}'_i B_{i,n+1}(u) = \Sigma_{i=0}^{n} \mathbf{p}_i B_{i,n}(u)$$

within the interval $[0, 1]$. Multiply the right hand side by $u + (1 - u)$ to increase the degree. Notice that this keeps the curve same within the interval $[0, 1]$. After equating the coefficients from both sides, it can be shown that

$$\mathbf{p}'_i = \frac{i}{n+1}\mathbf{p}_{i-1} + (1 - \frac{i}{n+1})\mathbf{p}_i$$

for $i = 0, ..., n + 1$.

This means we can compute a new set of control points $\mathbf{p}'_i$ using the original control points and piecewise linear interpolation at the parameter values $i/(n + 1)$.