Programming #1: Part III (4190.410)

Due: October 10, 2016

Given a cubic Bézier curve $C(t) = \sum_{l=0}^{3} \mathbf{b}_l B_l^3(t)$, $0 \le t \le 1$, with four control points $\mathbf{b}_0, \mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3$, we have the following differential properties:

$$\begin{aligned}
C(t) = (x(t), y(t)) &= (1-t)^3\ \mathbf{b}_0 + 3(1-t)^2 t\ \mathbf{b}_1 + 3(1-t)t^2\ \mathbf{b}_2 + t^3\ \mathbf{b}_3, \\
C'(t)/3 = (x'(t), y'(t))/3 &= (1-t)^2\ (\mathbf{b}_1 - \mathbf{b}_0) + 2(1-t)t\ (\mathbf{b}_2 - \mathbf{b}_1) + t^2\ (\mathbf{b}_3 - \mathbf{b}_2) \\
&= (\mathbf{b}_1 - \mathbf{b}_0) + t\ (2\mathbf{b}_2 - 4\mathbf{b}_1 + 2\mathbf{b}_0) + t^2\ (\mathbf{b}_3 - 3\mathbf{b}_2 + 3\mathbf{b}_1 - \mathbf{b}_0), \\
C''(t)/6 = (x''(t), y''(t))/6 &= (1-t)\ (\mathbf{b}_2 - 2\mathbf{b}_1 + \mathbf{b}_0) + t\ (\mathbf{b}_3 - 2\mathbf{b}_2 + \mathbf{b}_1) \\
&= (\mathbf{b}_2 - 2\mathbf{b}_1 + \mathbf{b}_0) + t\ (\mathbf{b}_3 - 3\mathbf{b}_2 + 3\mathbf{b}_1 - \mathbf{b}_0).
\end{aligned}$$

Part I: Compute all x and y-extreme points as well as all inflection points of $C(t)$, $0 \le t \le 1$, by solving the following three quadratic equations:

$$x'(t)/3 = 0, \quad y'(t)/3 = 0, \quad (x'(t)y''(t) - x''(t)y'(t))/18 = 0.$$

Moreover, compute all curvature extreme points by solving the following quintic equation:

$$a'(t)b(t) - 3a(t)(b'(t)/2) = 0,$$

where $a(t) = x'(t)y''(t) - x''(t)y'(t)$ and $b(t) = x'(t)^2 + y'(t)^2$.

Part II: For each curve segment subdivided at the extra points detected above, compute the chord, and bounding circular arcs for the segment. Next, subdivide each curve segment into two pieces by computing the curve location at the mid-parameter value, and repeat the same construction of chords and circular arcs for the new curve segments.

Part III: Design an interactive system that can control the position of a query point $\mathbf{Q}$ and the shape of $C(t)$ by dragging the four control points $\mathbf{b}_l$. Moreover, implement an algorithm for computing the projection line from $\mathbf{Q}$ to the nearest point on the curve $C(t)$. Display the BCA bounding volumes that have been used in the search for the nearest point $C(\hat{t})$ by the algorithm.