Programming #3: Part IV (4190.410)

Due: October 28, 2013

A cubic Bézier curve $C(t) = \sum_{l=0}^{3} \mathbf{b}_l B_l^3(t)$, $0 \le t \le 1$, can be approximated by a polygonal curve $L^h(t)$ connecting a sequence of curve points $C(t_i^h) = C(i/2^h)$, for $i = 0, \cdots, 2^h$, within an approximation error bound (Filip et al., CAGD 1986):

$$\|C(t) - L^h(t)\| \le \frac{3}{4} \cdot \frac{1}{4^h} \cdot \max\left(\|\mathbf{b}_0 - 2\mathbf{b}_1 + \mathbf{b}_2\|, \|\mathbf{b}_1 - 2\mathbf{b}_2 + \mathbf{b}_3\|\right) = \epsilon_h.$$

More precisely, each line segment $L_i^h(t)$, $(t_{i-1}^h \le t \le t_i^h)$, approximates the corresponding curve segment $C_i^h(t) = C(t)$, $(t_{i-1}^h \le t \le t_i^h)$, within the error bound $\epsilon_h \ge 0$.

Part I: Design an interactive system that can show the BVH structure (i.e., the AABB tree and the LSS tree) for the Bézier curve $C(t)$, $0 \le t \le 1$. Display the rounded AABB bounding volumes generated by expanding the AABB containing each line segment $L_i^h(t)$ by ϵ_h. Moreover, display the LSS bounding volumes generated by sweeping a disc of radious ϵ_h along each line segment $L_i^h(t)$, for the levels $h = 2, \cdots 10$.

Part II: 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 a recursive algorithm for computing the projection line from $\mathbf{Q}$ to the nearest point on the curve $C(t)$. Display the bounding volumes that have been used in the search for the nearest point $C(\hat{t})$ by the recursive algorithm.

Part III: Design an interactive system that can control the shape of two cubic Bézier curves $C(t)$ and $D(s)$ by dragging their control points. Moreover, implement an algorithm for computing the shortest distance between the two curves. Display the bounding volumes that have been used in the search for the nearest points between the two curves.

Part IV: Design an interactive system that can control the shape of two cubic Bézier space curves $C(t)$ and $D(s)$ by dragging their control points projected onto the xy, yz, and zx-planes. Moreover, implement an algorithm for computing the shortest distance between the two space curves.