SLEVEs for planar spline curves

Jörg Peters [a,*], Xiaobin Wu [a]

[a] *University of Florida*

Abstract

Given a planar spline curve and local tolerances, a matched pair of polygons is computed that encloses the curve and whose width (distance between corresponding break points) is below the tolerances. This is the simplest instance of a subdividable linear efficient variety enclosure, short sleve.

We develop general criteria, that certify correctness of a global, polygonal enclosure built from a sequence of individual bounding boxes by extending and intersecting their edges. These criteria prove correctness of the sleve construction.

Key words: spline curve, one-sided bounds, approximate implicitization, sleve, boundedness certificate.

1 Motivation

Nonlinear splines are essential for function and design, but they can present challenges for accurate and robust analysis and use. Since piecewise linear geometry is simpler to work with, an obvious approach is to approximate splines piecewise linearly. Although typically used in rendering, approximation by connecting point samples does not guarantee a bound on the approximation error and it is not a safe approximation for purposes like intersection testing since it does not conservatively bound the spline from one side. The control polygon does better in that the local convex hull provides a safe enclosure; but the width of the enclosure is typically far from minimal. Suboptimal width of the enclosure increases the cost of interrogation in applications such as interference testing or ray casting.

For a large class of refinable functions, in one or several variables, in particular for splines, Lutterkort and Peters (2001a,b) developed subdividable linear efficient

[*] Corresponding author, NSF 9457806-CCR
Email addresses: jorg@cise.ufl.edu (Jörg Peters), xwu@cise.ufl.edu (Xiaobin Wu).

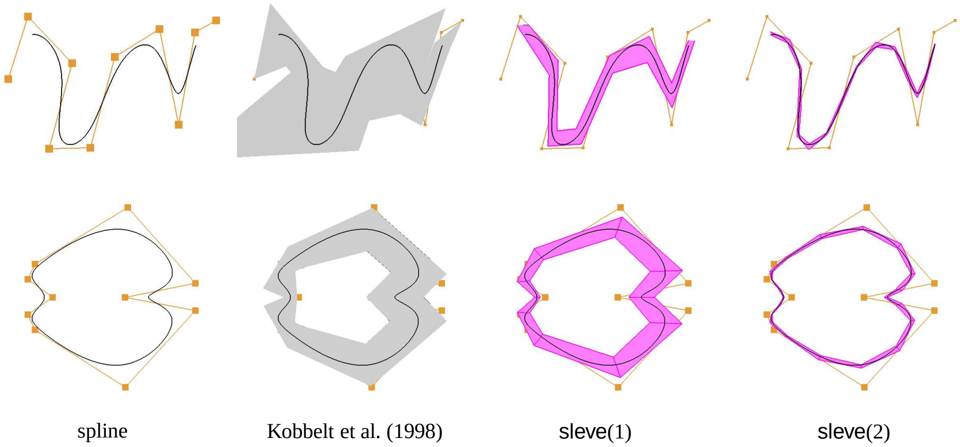


Fig. 1. **Enclosing uniform cubic splines**. *from left to right* B-spline control polygon and curve; envelope constructed according to Kobbelt et al. (1998); sleve with the same number of segments, sleve with twice the number of segments.

function enclosures, short slefes (pronounced like shirt sleeves), that provide two-sided bounds. Their width is typically much less than that of convex hulls or bounding boxes while using roughly the same number of operations. We say 'typically' since the constructions agree for a horizontal line segment and are asymptotically equivalent with an $O(h^2)$ convergence. The combination of accuracy and low cost is possible, because Lutterkort et al. precompute offline and tabulate a costly optimization specific to the class of splines. The runtime cost is therefore dominated by summing a few second differences of the control points.

There are two challenges when leveraging slefes: first, to bound each segment by selecting the correct combination of upper and lower bounds of the x and the y component (it is not true that the combination of the upper bound on x and the upper bound on y always bound the curve from one side – the corresponding broken line may well intersect the curve!); and second, to combine the resulting segment bounding boxes to form a matched pair of polygons that contiguously bound the spline curve. The second challenge is not specific to slefes and so we develop it for bounding boxes in general, classifying the failure modes and formulating certificates that verify correctness. The resulting framework helps repair the envelopes proposed in (Kobbelt, 1998; Kobbelt et al., 1998; Shen and Patrikalakis, 1998) shown in Figures 3, 4, 5.

The structure of the paper is as follows.
Section 3 shows how to efficiently and tightly bound a single polynomial segment that might be a piece of the x-component of the curve.
Section 4 explains why these bounds can not directly be leveraged to yield an enclosure of the curve.
Section 5 gives an efficient way of computing a piece of the enclosure. This construction is specific to sleves.
Section 6 gives the criterion for correctly joining two adjacent bounding boxes by

extending and intersecting their edges.
Section 7 gives the criterion for correctly completing a sequence of bounding boxes pairs. Both multi-box criteria are generic and apply to other constructions besides sleves.
Section 9 states the algorithm formally and in detail and Section 10 concludes with applications.

2 Related bounding constructs

The theory of slefes has its roots in bounds on the distance of piecewise polynomials to their Bézier or B-spline control net (Nairn, Peters, and Lutterkort, 1999; Reif, 2000). Compared to these constructions, slefes yield dramatically tighter bounds for the underlying functions since they need not enclose the control polygon. Lutterkort and Peters (2001a,b) develop slefes for splines, in particular for splines in Bézier form. In a nutshell, the slefe of a function x is an explicit two-sided approximation $\overline{x}, \underline{x}$ so that $\underline{x} \leq x \leq \overline{x}$ over the domain of interest. Key to the approach are the *a priori* knowledge of the finite spline basis and the fact that (after a careful change of basis), we can precompute optimal bounds on basis functions and tabulate them once and for all. Thus the work is factored into a best max-norm approximation, done off-line – and the computation of second differences for a specific instance of a spline. Bounds based on slefes are observed to be very tight. Indeed, while the simple linear slefe construction cannot be expected to solve the hard non-linear problem of best two-sided max-norm approximation, the detailed analysis in (Peters and Wu, 2003) confirms near-optimality for cubics that have no inflection in the interval of interest: the slefe differs by less than 7% from the optimal.

Approximation theory has long recognized the problems of one-sided approximation and two-sided approximation (Buck, 1965). Algorithmically, though, according to the seminal monograph (Pinkus, 1989), page 181, the convergence of the proposed Remez-type algorithms is already in one variable ‘generally very slow’. The only termination guarantee is that a subsequence must exist that converges. By contrast, the slefes provide a solution with an explicit error very fast and with a guarantee of error reduction under refinement.

If we distinguish between elementary bounding constructs and (hierarchical) structures that employ these elementary bounding constructs as their oracles, sleves are a structure and its elementary pieces are hexagons $\mathbb{H}_\mu$ defined in Section 4. A gallery of elementary bounding constructs is shown in Figure 2 including axis-aligned bounding boxes (AABB), oriented bounding boxes (OBB), quantized bounding boxes also called ‘k-dops’ or discrete orientation polytopes (convex polytopes whose facets are determined by half spaces whose outward normals come from a small fixed set of k orientations) (Crosnier and Rossignac, 1999; Kay and Ka-

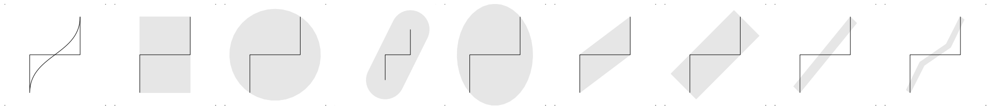


Fig. 2. **Enclosures** based on control points: less grey is better! (*from left to right:*) cubic curve with control polygon, axis-aligned box, bounding circle, Filip et al. bound (scaled by 1/2), bounding ellipse convex hull, oriented bounding box, ‘fat arc’, 3-segment sleve.

jiya, 1986; Klosowski et al., 1998), fat arcs (Sederberg et al., 1989), the bound of Filip et al. (1986) which is based on Taylor expansion, convex hulls, bounding spheres and minimal enclosing ellipsoids (Welzl, 1991). Gottschalk et al. (1996) and Klosowski (1998) give a good overview of how elementary bounding constructs are used in the context of hierarchical interference detection (for space partitioning methods see e.g. (Basch, 1999)): simpler constructs like AABBs and spheres provide fast rejection tests in sparse arrangements, while more expensive k-dops and OBBs perform better on complex objects in close proximity. With adaptive resolution, sleves are best suited for *curved, non-polyhedral objects in close proximity* (cf. Figure 20).

Cohen et al. (1996) and Sander et al. (2000) modify surface simplification for triangulated surfaces to generate (locally) inner and outer hulls. This requires solving a sequence of linear programs at runtime and applies to already triangulated surfaces.

Kobbelt (1998); Kobbelt et al. (1998), assemble oriented bounding boxes into a structure called ‘envelope’. The goal is to hierarchically support accurate ray-tracing of complex curved objects represented as subdivision curves or surfaces. While the focus is on surfaces, Kobbelt (1998) and Kobbelt et al. (1998) use cubic curves as the motivating case study. Envelope boxes and slefe boxes differ in that an envelope depends, via evaluation and normals, non-linearly on the coefficients of the

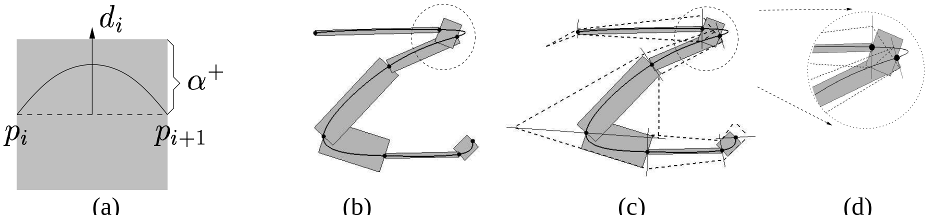


Fig. 3. **Unsafe construction** proposed in (Kobbelt, 1998; Kobbelt et al., 1998) applied to a uniform cubic curve. (a,b) For each polynomial curve segment with endpoints p_i and p_{i+1} and defined by coefficients c_j, $j = i-1, \dots, i+2$, a (*grey*) box is constructed with corners $p_i + \alpha^+ d_i, p_i - \alpha^- d_i, p_{i+1} + \alpha^+ d_i, p_{i+1} - \alpha^- d_i$, were d_i is the direction perpendicular to the chord p_i, p_{i+1} and α^+ is a bound on the distance of the curve from the line through p_i and p_{i+1} in the direction d_i, α^- is a bound in the $-d_i$ direction. (c) The grey boxes are extended, clipped against the line in the curve’s normal direction at p_i and the extreme intersections are selected. This yields the dashed boundaries of the envelope defined in (Kobbelt, 1998; Kobbelt et al., 1998) (d) The enlarged detail shows that the bounding boxes and hence the envelope do not fully enclose the curve.

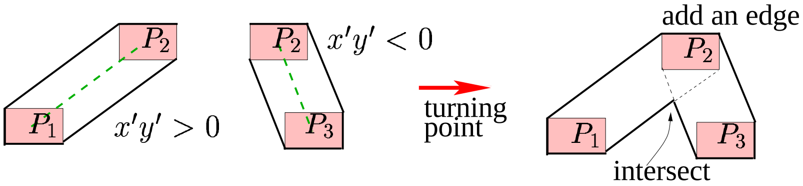


Fig. 4. **Unsafe construction** of outer boundaries (solid line segments) for a two-piece linear interval spline according to (Shen and Patrikalakis, 1998, page 51). (*left two*) the outer bounds (solid line segments) are selected for each piece according to the sign combination of the partial derivatives of the central curve (*dashed*) of the spline; (*right*) At the turning point, an edge is added and outer bounds are intersected and trimmed.

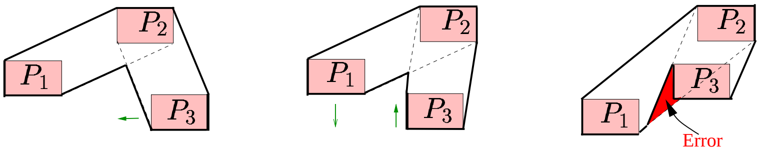


Fig. 5. Modifying the example of Figure 4. From (*left*) to (*middle*), the envelope edges of P_3 switch from right and lower to left and lower adding one more edge. From (*middle*) to (*right*), as P_1 and P_3 exchange their y positions, the boundaries not longer enclose the first linear interval. In particular, a dark (red) **wedge is not covered**.

spline. By contrast, the slefes are pseudo-linear, i.e. linear except for a min-max selection. This allows solving inverse problems such as fitting spline curves into prescribed channels (Lutterkort and Peters, 1999). The envelope construction steps and their failure to enclose a planar cubic curve are illustrated in Figure 3. We can repair the failure of the local bounding box by extending the grey box of the ith curve segment to enclose all four corresponding control points c_j; or by somehow chopping the curve into pieces so small that its orthogonal projection onto the chord is 1-1. But that does not solve the second, more complex problem that breaks the construction discussed next, namely connecting the boxes by extending and intersecting their edges to form a globally valid enclosure.

Hu et al. (1996a,b,c); Tuohy et al. (1997) promote the use of *interval spline representation* (Sederberg and Farouki, 1992) for tolerancing, error maintenance and data fitting. The key ingredient of this use of interval arithmetic are AABBs based on the positivity and partition of unity property of the B-splines. Among many other contributions, Shen and Patrikalakis (1998) propose to bound any interval spline of uniform width (for each component, the difference between upper bound and lower bound is constant for all parameters) by the construction sketched in Figure 4, whose result is also called ‘envelope’. The failure of the approach for a 2-piece interval spline curve of degree 1 is illustrated in Figure 5. The initial bounding boxes, here intervals, are correct, but the algorithm by which their union is formed is flawed since it violates certificate (2) derived in Section 6.

3 Subdividable Linear Efficient Function Enclosures

The subdividable linear efficient function enclosure, or slefe of a possibly multivariate function x with respect to a domain U is a piecewise linear pair, $\overline{x}, \underline{x}$, of upper and lower bounds that sandwich the function on U: $\overline{x} \geq x \geq \underline{x}$. Definition 9.1 states a specific algorithm to compute $\overline{x}$ and $\underline{x}$ and Section 3.1 below explains the derivation. The goal is to minimize the width,

$$w(x,U) := \overline{x} - \underline{x},$$

in the *recursively applied L^∞ norm*: the width is as small as possible where it is maximal – and, having fixed the breakpoint values where the maximal width is taken on (zeroth and first breakpoint in Fig. 6), the width at the remaining breakpoints is recursively minimized subject to matching the already fixed break point values.

3.1 The general slefe *construction*

The slefe construction is based on the two general lemmas from (Lutterkort and Peters, 2001a; Lutterkort, 2000) and the once-and-for-all tabulation $a[..,..]$ of best recursive L^∞ enclosures of a small set of functions, $\mathbf{a}_\nu$, $\nu = 1, \ldots, s$. These tables are available via (Wu and Peters, 2004). Understanding the section is helpful but not necessary for understanding the remainder of this paper.

The change-of-basis lemma below is valid for a large class of functions and approximating spaces, and it is best proven in the general setting since specialization only adds notation. For our purposes, it suffices to think of B as the space of B-splines or Bézier polynomials of a certain degree, H the hat functions that generate the control polygon (or just the linear functions, in the case of Bézier polynomials), L the mapping to the control polygon (or just to the chord connecting the end points, in the case of Bézier polynomials), Δ the second differences of the control points, s the degree minus one and the s functions $\mathbf{a}_\nu$ collected into a vector $\mathbf{a}$. A concrete example, is given right after the lemma.

Lemma 3.1 (change of basis) *Given two finite-dimensional vector spaces of functions, $B \neq H$, $s := \dim B - \dim(B \cap H)$, $(\mathbf{b}_k)_{k=1,\ldots,\dim B}$ a basis of B, $\mathbf{a}_\nu$, $\nu = 1, \ldots, s$ functions in B, and linear maps*

$$L : B \to H, \qquad \Delta : B \to \mathbb{R}^s,$$

such that (i) $(\Delta_j \mathbf{a}_\nu)_{\nu,j}$ is the identity in $\mathbb{R}^{s\times s}$ and (ii) $\ker \Delta = \ker(E - L)$ (where E is the embedding identity) then for any $x := \mathbf{b} \cdot \mathtt{x} \in B$,

$$(\mathbf{b} - L\mathbf{b}) \cdot \mathtt{x} = (\mathbf{a} - L\mathbf{a}) \cdot (\Delta x).$$

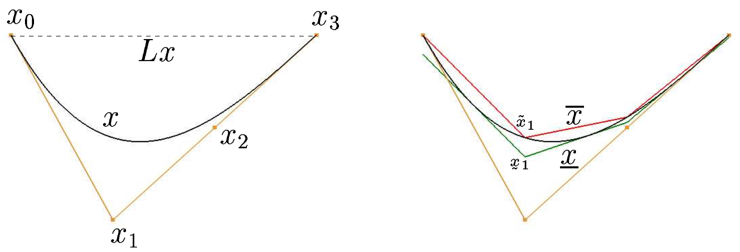


Fig. 6. The function $x := \mathbf{a}_1^3 = -(2\mathbf{b}_1^3 + \mathbf{b}_2^3)/3$ with (*left*) its control polygon, its linear interpolant Lx and (*right*) its slefe $(\underline{x}, \overline{x})$.

Proof By (i) $\Delta(I - \mathbf{a}\Delta)x = 0$ and hence, by (ii), $(E - L)(I - \mathbf{a}\Delta)x = 0$.

The bounds, Lemma 3.2, follow by interval arithmetic.

Lemma 3.2 (bounds) *With the definitions of Lemma 3.1, if* $\underline{\mathbf{a}_\nu - L\mathbf{a}_\nu} \leq \mathbf{a}_\nu - L\mathbf{a}_\nu \leq \overline{\mathbf{a}_\nu - L\mathbf{a}_\nu}$ *on every point of* U *then*

$$\underline{x} := Lx + \sum_\nu \underline{\mathbf{a}_\nu - L\mathbf{a}_\nu} \max\{0, \Delta x(\nu)\} + \sum_\nu \overline{\mathbf{a}_\nu - L\mathbf{a}_\nu} \min\{0, \Delta x(\nu)\},$$
$$\overline{x} := Lx + \sum_\nu \overline{\mathbf{a}_\nu - L\mathbf{a}_\nu} \max\{0, \Delta x(\nu)\} + \sum_\nu \underline{\mathbf{a}_\nu - L\mathbf{a}_\nu} \min\{0, \Delta x(\nu)\}.$$

sandwich x *on* U*:* $\quad \underline{x} \leq x \leq \overline{x}$.

Given $x \in B$ the slefe is constructed in five steps. The first four of these can be precomputed based on B. Only the fifth step is specific to x.

(1) Choose U, the domain of interest, and the space H of enclosing functions.
(2) Choose a difference operator $\Delta : B \mapsto \mathbb{R}^s$, with $\ker \Delta = B \cap H$.
(3) Compute $\mathbf{a} : \mathbb{R}^s \mapsto B$ so that $\Delta\mathbf{a}$ is the identity on $\mathbb{R}^s$ and each $\mathbf{a}_i$ matches the same $\dim(B \cap H)$ additional independent constraints.
(4) Compute $\underline{\mathbf{a}_\nu - L\mathbf{a}_\nu}$ and $\overline{\mathbf{a}_\nu - L\mathbf{a}_\nu} \in H$.
(5) Compute Δx and assemble $\underline{x}$ and $\overline{x}$ according to Lemma 3.2.

To make the framework concrete, we specialize it to one variable. Let

- B be the space of univariate polynomials of degree d, in Bézier form

$$x(u) := \textstyle\sum_{k=0}^{d} \mathrm{x}_k \mathbf{b}_k^d(u), \; \mathbf{b}_k^d(u) := \frac{d!}{(d-k)!k!}(1-u)^{d-k}u^k.$$

  In Figure 6, $d = 3$.
- H the space of piecewise linear functions $\mathbf{h}_\mu$ with break points at μ/m, $\mu \in \{0, \ldots, m\}$. We write

$$\overline{\mathbf{h}}^{m} \in H$$

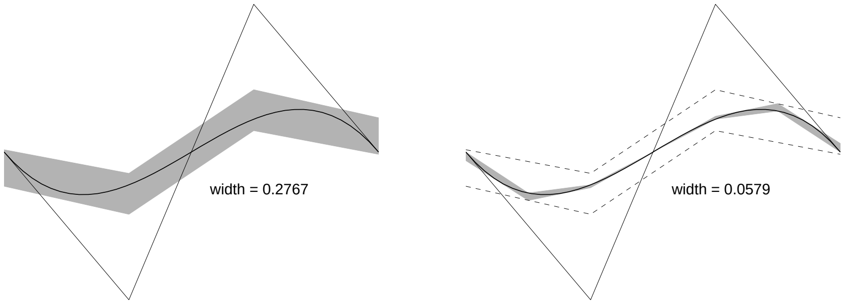


Fig. 7. (*left*) A cubic B´ezier segment with coefficients $0, -1, 1, 0$. The control polygon exaggerates the curve far more than the *grey* 3-piece slefe. (*right*) After one **subdivision at the midpoint**, the width of the slefe (*grey*) is roughly 1/4th of the width of the unsubdivided slefe (*dashed*).

to indicate that there are m linear segments joined to form the upper bound. In Figure 6, we use $m = 3$, but we could choose $m = 10$ or $m = 1$ just as well.

- Δx the $s = d - 1 = 2$ second differences of the Bézier coefficients

$$\Delta x := \begin{bmatrix} \Delta_1 x \\ \Delta_2 x \end{bmatrix} := \begin{bmatrix} x_0 - 2x_1 + x_2 \\ x_1 - 2x_2 + x_3 \end{bmatrix}$$

(Note that Δ denotes a second difference while the subscript in Δ_2 picks out the 2nd entry in the vector Δx of second differences.)

- $Lx(u) := x_0(1 - u) + x_d u$, (by contrast, for B-splines, L maps to the control polygon),
- $U = [0..1]$.

This yields

$$\mathbf{a}_1^d := -(2\mathbf{b}_1^d + \mathbf{b}_2^d)/d, \quad \mathbf{a}_2^d := -(\mathbf{b}_1^d + 2\mathbf{b}_2^d)/d,$$

and since, for the special choice of L, we have $\mathbf{a}_\nu^d - L\mathbf{a}_\nu^d = \mathbf{a}_\nu^d$, the equation of Lemma 3.1 reads

$$x - Lx = \mathbf{a}_1^d \Delta_1 x + \mathbf{a}_2^d \Delta_2 x.$$

By the symmetry $\mathbf{a}_1^3(1-t) = \mathbf{a}_2^3(t)$. To apply Lemma 3.2, it is sufficient to precompute the *optimal* enclosures for $\mathbf{a}_1^3$. Due to the convexity of $\mathbf{a}_1^3$ (see Fig. 6), the piecewise linear interpolant at μ/m is an upper bound and the table $a[d, m, sgn, \nu, \mu]$ storing the $m + 1$ breakpoint values of $\overline{\mathbf{a}_1^3}^m$, is for $m = 3$

$$a[3, 3, +1, 1, ..] = [0, -10/27, -8/27, 0].$$

For example, the value of $\overline{\mathbf{a}_1^3}$ at $1/3$ is $-10/27$. The optimal lower bound is computed by recursive minimization, according to Lemma 5 of (Peters and Wu, 2003). This procedure yields the $m + 1$ break point values

$$a[3, 3, -1, 1, ..] = [-.0695, -.4399, -.3154, -.0087].$$

Since Δ is a second difference,

- the width

$$w_{\text{slefe}}(x;U) := \max_U \overline{x} - \underline{x} = \max_U \sum_{\nu=1}^{d-1} (\overline{\mathbf{a}_\nu - L\mathbf{a}_\nu}^m - \underline{\mathbf{a}_\nu - L\mathbf{a}_\nu}_m)\, |\Delta_\nu x|$$

  is invariant under addition of constant and linear terms to x and
- one (DeCasteljau) subdivision step (see (Prautzsch et al., 2002; Farin, 1997, e.g.)) at the midpoint, $t = 1/2$, cuts the width to roughly *a quarter* (see Figure 7), because the width $(\overline{\mathbf{a}_\nu - L\mathbf{a}_\nu}^m - \underline{\mathbf{a}_\nu - L\mathbf{a}_\nu}_m)$ of the $\mathbf{a}_\nu$ stays fixed but the maximal $\Delta_\nu x$ shrinks to $1/4$ its size.

4 Constructing planar sleves

The slefes of the x and the y component of a planar curve each have an upper and a lower bound. These combine to four potential bounds for the μ^{th} segment $[t_\mu..t_{\mu+1}]$, $t_\mu := \frac{\mu}{m}$ of the spline $x(t), y(t)$:

$$\begin{aligned}
\ell^{++} &:= (1-t)\begin{bmatrix}\overline{x}(t_\mu)\\ \overline{y}(t_\mu)\end{bmatrix} + t\begin{bmatrix}\overline{x}(t_{\mu+1})\\ \overline{y}(t_{\mu+1})\end{bmatrix}, &\quad \ell^{+-} &:= (1-t)\begin{bmatrix}\overline{x}(t_\mu)\\ \underline{y}(t_\mu)\end{bmatrix} + t\begin{bmatrix}\overline{x}(t_{\mu+1})\\ \underline{y}(t_{\mu+1})\end{bmatrix},\\
\ell^{-+} &:= (1-t)\begin{bmatrix}\underline{x}(t_\mu)\\ \overline{y}(t_\mu)\end{bmatrix} + t\begin{bmatrix}\underline{x}(t_{\mu+1})\\ \overline{y}(t_{\mu+1})\end{bmatrix}, &\quad \ell^{--} &:= (1-t)\begin{bmatrix}\underline{x}(t_\mu)\\ \underline{y}(t_\mu)\end{bmatrix} + t\begin{bmatrix}\underline{x}(t_{\mu+1})\\ \underline{y}(t_{\mu+1})\end{bmatrix}.
\end{aligned}$$

Figures 8 and 9 illustrate the resulting collection of possibly intersecting line segments. Evidently, this collection does not enclose the curve due to the gaps. However, thanks to linearity, each piece $\mathbb{H}_\mu$ of the enclosure is a convex combination of consecutive point enclosures $\square_\mu$ and $\square_{\mu+1}$ where (Figure 9, *right*)

$$\square_\mu \text{ is the quadrilateral with the vertices } \begin{bmatrix}\overline{x}(t_\mu)\\ \overline{y}(t_\mu)\end{bmatrix}, \begin{bmatrix}\underline{x}(t_\mu)\\ \overline{y}(t_\mu)\end{bmatrix}, \begin{bmatrix}\overline{x}(t_\mu)\\ \underline{y}(t_\mu)\end{bmatrix}, \begin{bmatrix}\underline{x}(t_\mu)\\ \underline{y}(t_\mu)\end{bmatrix}.$$

Each *point enclosure* $\square_\mu$ is an axis-parallel rectangle or box and the curve is enclosed by the *linear interval splines of the point enclosures*. Precisely, with

$$\begin{aligned}
v_\mu(\alpha,\beta) := &(1-\alpha)(1-\beta)\begin{bmatrix}\overline{x}(t_\mu)\\ \overline{y}(t_\mu)\end{bmatrix} + \alpha(1-\beta)\begin{bmatrix}\underline{x}(t_\mu)\\ \overline{y}(t_\mu)\end{bmatrix}\\
&+ (1-\alpha)\beta\begin{bmatrix}\overline{x}(t_\mu)\\ \underline{y}(t_\mu)\end{bmatrix} + \alpha\beta\begin{bmatrix}\underline{x}(t_\mu)\\ \underline{y}(t_\mu)\end{bmatrix}, \quad \alpha,\beta \in [0..1],
\end{aligned}$$

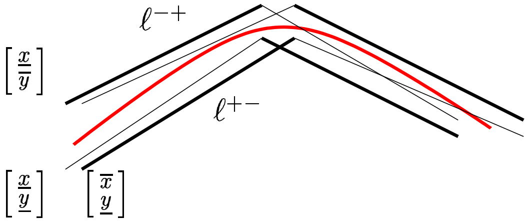

ℓ^{-+}
$\begin{bmatrix}\underline{x}\\ \overline{y}\end{bmatrix}$
ℓ^{+-}
$\begin{bmatrix}\underline{x}\\ \underline{y}\end{bmatrix}$
$\begin{bmatrix}\overline{x}\\ \underline{y}\end{bmatrix}$

Fig. 8. The **extreme**, outermost slefe **components** of the curve $\mathbf{x}$ are emphasized as fat line segments. Note the gap and the intersection between consecutive extreme segments.

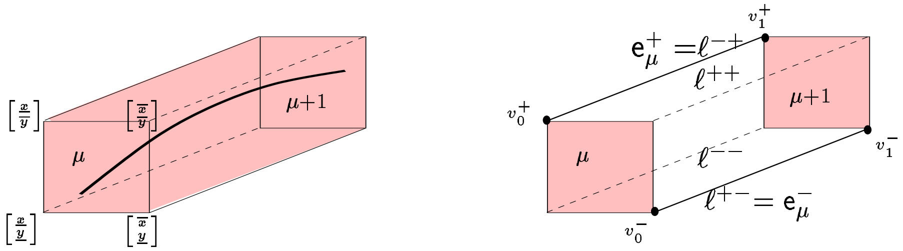
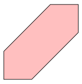

$\begin{bmatrix}\underline{x}\\\overline{y}\end{bmatrix}$
$\begin{bmatrix}\overline{x}\\\overline{y}\end{bmatrix}$
$\mu+1$
μ
$\begin{bmatrix}\underline{x}\\\underline{y}\end{bmatrix}$
$\begin{bmatrix}\overline{x}\\\underline{y}\end{bmatrix}$
v_1^+
$\mathsf{e}_\mu^+ = \ell^{-+}$
ℓ^{++}
$\mu+1$
v_0^+
μ
ℓ^{--}
v_1^-
$\ell^{+-} = \mathsf{e}_\mu^-$
v_0^-

Fig. 9. (*left*)The curve segment stays inside $\mathbb{H}_\mu$, the **convex combination of two neighboring point enclosures**. (*right*) Two extreme line segments, $\mathsf{e}_\mu^+, \mathsf{e}_\mu^-$, are picked among the four lines $\ell^{--}, \ell^{+-}, \ell^{++}, \ell^{-+}$ connecting corresponding corners of the two point enclosures. The hexagon $\mathbb{H}_\mu$ is generated by two extreme segments and edges of $\square_\mu$ and $\square_{\mu+1}$.

a bilinear average of the corners of the point enclosure, we define (a multiple cover of) $\mathbb{H}_\mu$ as

$$\mathbb{H}_\mu(\alpha, \beta, t) := (1-t)v_\mu(\alpha,\beta) + tv_{\mu+1}(\alpha,\beta).$$



Such interval enclosures can be used, say for intersection testing. However, just like other local enclosures, they have two shortcomings for effective use: multiplicity and gaps. *Multiplicity* refers to the fact that we need to test against all possible bounding segments. For example, to check interference between two interval enclosures, we need to check the four potential boundary curves of one against those of the other and even this is not enough due to gaps. *Gaps* refers to the fact that adjacent bounding segments do not necessarily meet up or they intersect early. We would like to prescribe the number of segments per polynomial piece at the outset. However, filling the gaps either requires more pieces by including edges of the point enclosures, or trimming to an intersection. (There could be up to three additional segments for one breakpoint.) Fortunately, if we are careful, the obvious strategy of extending or clipping ‘extreme’ segments can be guaranteed to work efficiently and stably.

5 Multiplicity: computing local bounding hexagons

This section explains a highly efficient way of computing $\mathbb{H}_\mu$ via a decision table. We focus on two adjacent point enclosures $\square_\mu$ and $\square_{\mu+1}$ as shown in Figure 9. We want to identify a pair of *extreme line segments*

$$\{\mathsf{e}_\mu^+, \mathsf{e}_\mu^-\} \subseteqq \{\ell^{--}, \ell^{+-}, \ell^{++}, \ell^{-+}\}$$

that, together with four edges of the two point enclosures, bound a hexagon that in turn covers $\mathbb{H}_\mu$. (For some configurations, the hexagon will only have 4 or 5 extreme points.)

Definition 5.1 *Let $\ell \in \{\ell^{--}, \ell^{+-}, \ell^{++}, \ell^{-+}\}$, i.e. $\ell(t) := (1-t)v_{\mu 0} + tv_{\mu 1}$ where $v_{\mu 0}$ is a corner of $\Box_\mu$ and $v_{\mu 1}$ the same corner in $\Box_{\mu+1}$. The normal of the ℓ is defined as $\mathsf{n}(\ell) := \begin{bmatrix} -(v_{\mu 1}-v_{\mu 0})_y \\ (v_{\mu 1}-v_{\mu 0})_x \end{bmatrix}$ and*

$$d_\mu(v) := \mathsf{n}(\ell) \cdot (v_{\mu 0} - v) = \mathsf{n}(\ell) \cdot (v_{\mu 1} - v) \, .$$

Then ℓ is extreme *for the interval $[t_\mu..t_{\mu+1}]$ if and $d_\mu(v)$ is of one sign for all $v \in \mathbb{H}_\mu$.*

Here and in the following, we interpret $sign(0)$ to match both $+$ and $-$ so that we can write: for all $v \in \Box_\mu$, $sign(v_{\mu 0} - v) = (-,-)$, even though choosing $v := v_{\mu 0}$ yields $v_{\mu 0} - v = (0,0)$.

Now consider ℓ^{--}. Since it is based on the lower slefe boundaries of x and y, $sign(v_{\mu 0} - v) = (-,-)$ for all $v \in \Box_\mu$. If the normal of ℓ^{--} also has signs $(-,-)$ then $d_\mu(v) \geq 0$. This motivates the following lemma:

Lemma 5.1 *A line segment $\ell \in \{\ell^{--}, \ell^{+-}, \ell^{++}, \ell^{-+}\}$ is extreme if and only if the signs of the x and y components of its normal both agree or both disagree with the superscripts of ℓ.*

Proof Suppose the signs agree. For $v \in \Box_\mu$, we get $d_\mu(v) = \mathsf{n}(\ell) \cdot (v_{\mu 0} - v) \geq 0$ since the vector $(v_{\mu 0} - v)$ has the same signs as the superscript of ℓ. For $v \in \Box_{\mu+1}$, $d_\mu(v) = \mathsf{n}(\ell) \cdot (v_{\mu 1} - v) \geq 0$ since $(v_{\mu 1} - v)$ has the same signs as the superscript of ℓ. For any $v \in \mathbb{H}_\mu$, we have $v = (1-t)v_0 + tv_1$, where $v_0 \in \Box_\mu$, $v_1 \in \Box_{\mu+1}$ and $t \in [0..1]$; and therefore $d_\mu(v) = (1-t)d_\mu(v_0) + td_\mu(v_1) \geq 0$. If both signs are opposite, the arguments applies with ≥ 0 replaced by ≤ 0.

Conversely, if one component of n has the same sign as $(v_{\mu 0} - v)$ and the other has the opposite sign, then choosing v on an edge of $\Box_\mu$ with $v_{\mu 0}$ as corner will set one component of the vector $(v_{\mu 0} - v)$ to zero. We zero out the component that had the same sign to show that $d_\mu(v) \leq 0$ can happen and we zero out the component with the opposite sign to show that also $d_\mu(v) \geq 0$ for some $v \in \mathbb{H}_\mu$, i.e. ℓ is not extreme. $|||$

We superscript the extreme line segments as $\mathsf{e}_\mu^{sign(d_\mu)}$. Algorithmically, e_μ^+ and e_μ^- can be detected by computing the signs of the differences of the component slefes:

$$\sigma := sign[\underline{y}(t_\mu) - \underline{y}(t_{\mu+1}),\ \overline{y}(t_\mu) - \overline{y}(t_{\mu+1}),\ \underline{x}(t_{\mu+1}) - \underline{x}(t_\mu),\ \overline{x}(t_{\mu+1}) - \overline{x}(t_\mu)]$$

and applying the decision table

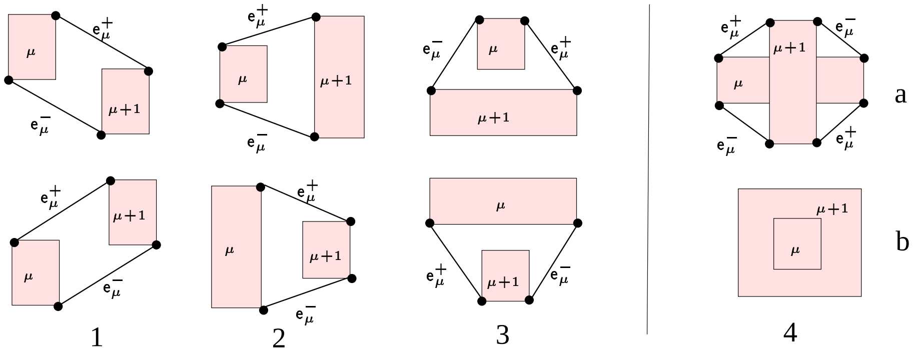


Fig. 10. **Eight point enclosure configurations**. Configurations 4a and 4b trigger subdivision.

T:

$\sigma_0 =$	$-$	$+$		
$\sigma_1 =$			$-$	$+$
$\sigma_2 =$	$-$		$+$	
$\sigma_3 =$		$-$		$+$
$\mathsf{e}_\mu^+ \leftarrow$	ℓ^{--}	ℓ^{+-}	ℓ^{-+}	ℓ^{++}

That is, to determine e_μ^+ we proceed as follows. If $\underline{y}(t_\mu)-\underline{y}(t_{\mu+1}) \leq 0$ and $\underline{x}(t_{\mu+1})-\underline{x}(t_\mu) \leq 0$, and hence $\sigma_0 = -1 = \sigma_2$ then ℓ^{--} is an extreme segment and is named e_μ^+. To determine e_μ^-, we negating the signs in the vector σ. Thus, if both the sign of the x and of the y component of $\mathsf{n}(\ell)$ are the opposite of the superscripts of ℓ then $\ell \rightarrow \mathsf{e}_\mu^-$.

Only cases (4a) and (4b) of Figure 10 provide no assignment or two competing assignments. Often, we can eliminate this case by replacing $\Box_\mu$ and $\Box_{\mu+1}$ by the smallest enclosing box if that allows a construction within the tolerance.

Lemma 5.2 *If*

Table T determines unique segment e_μ^+ and e_μ^- or (1)
two choices lie on the same line (and either may be picked)

then e_μ^+ and e_μ^- and four edges taken from $\Box_\mu$ and $\Box_{\mu+1}$ form a polygon that covers $\mathbb{H}_\mu$, or the smallest enclosing box are enclosed.

Proof By definition, the segments e_μ^+ and e_μ^- corresponding to $[t_\mu..t_{\mu+1}]$ do not cross. There are 2^4 possible values for σ. The configurations characterized by eight of these are depicted in Figure 10 and the other eight are obtained by exchanging the subscripts μ and $\mu+1$. Only configurations 4a and 4b do not yield unique extreme lines. |||

Here and later, we argue that subdivision at the midpoint eventually removes the offending cases, which correspond to sharp turns or self intersection.

To simplify the argument, we rule out curves with cusps and corners for now but will include them in the final argument.

Lemma 5.3 *If the curve is regularly parametrized, subdivision at the midpoint in the limit (i) orders any triple $\square_{\mu-1}$, $\square_\mu$ and $\square_{\mu+1}$ with respect to the tangent direction and (ii) separates any pair $\square_\mu$ and $\square_{\mu+1}$.*

Proof Since the parametrization is not singular, there is a finite proportionality between the length of the parameter interval and the arclength. Since the width of each component slefe is proportional to the second differences of the control points and the second differences decrease like $O(h^2)$ when the parameter interval is shortened to h, the maximal width of $\square_\mu$ shrinks faster than the parameter interval; therefore the point enclosures shrink more rapidly than the curve length. Since the parametrization is not singular, in a small neighborhood of t_μ, the curve is a function over the tangent line at t_μ and therefore orders the triple. $|||$

For example, if the tangent line is the x-axis then, in the limit, the minimum x component of $\square_{\mu+1}$ will be larger than the maximum x component of $\square_\mu$.

While separation of $\square_\mu$ and $\square_{\mu+1}$ rules out cases 4a and 4b, a weaker condition suffices since only complete interpenetration of the point enclosures has to be prevented. In particular, (ii) holds already when the curve *pieces* are regularly parametrized, allowing for corners at joints between the polynomial pieces. So, if we can assure that we eventually subdivide at all parameters that correspond to a singularity, (ii) also holds for curves with isolated singularities. A piece of zero extent presents no problem since the point enclosures have zero width. Note that two adjacent curves may need to be subdivided if we are at an end point, i.e. if $\mu = 0$ or $\mu = m_i$.

6 Gaps and Intersections: the correct pairwise joining of bounding regions

If we would simply take the union of the bounding regions constructed in the previous section, we would have a correct enclosure of the curve. However, as pointed out earlier, that union is considerably more complex than two line segments per curve segment and it is not cheap to compute compared to extending and intersecting the edges of the regions. This section gives a criterion for correctly joining two adjacent bounding regions by extending and intersecting their edges. It applies, in particular, to bounding regions generated as the convex hull of two boxes, *not necessarily axis-aligned*. Oriented bounding boxes fall into this framework as degenerate hexagons, where the box is reduced to a line segment. In particular, schemes

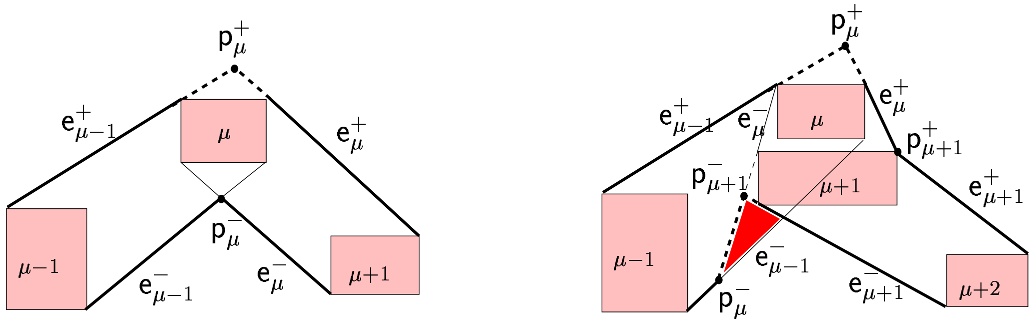


Fig. 11. (*left*) Two adjacent extreme segments are intersected to get a vertex p_μ^+ on the sleve. The sleve polygons are shown as bold lines. (*right*) **Certificate (2)** fails for this configuration of point enclosures and, as indicated by the dark (red) triangle, the sequence of extreme segments no longer encloses the slefe hexagon $\mathbb{H}_{\mu-1}$.

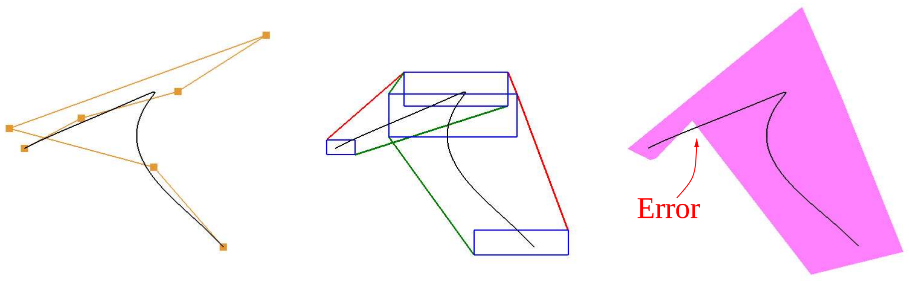


Fig. 12. A concrete example of a curve (with $d = 6$ and $m = 3$) for which **certificate (2) fails**. The polygons with breakpoints $\mathsf{p}_\mu^+, \mathsf{p}_\mu^-$ do not enclose the curve.

like (Kobbelt, 1998; Kobbelt et al., 1998; Shen and Patrikalakis, 1998) are covered.

In general, gaps and intersection appear between the individual bounding boxes (see Figure 11, *left*). For sleves this occurs when different corners of the point enclosure are chosen by two adjacent segments. (This, in turn, corresponds to differing superscripts ρ, τ of $\ell^{\rho,\tau}$ for the extreme segments μ and $\mu + 1$.)

Denote the intersections of $\mathsf{e}_{\mu-1}^+$ with e_μ^+, respectively of $\mathsf{e}_{\mu-1}^-$ with e_μ^- as

$$\mathsf{p}_\mu^+ := \mathsf{e}_{\mu-1}^+(t_{\mu-1,1}^+) = \mathsf{e}_\mu^+(t_{\mu,0}^+) \quad \text{and} \quad \mathsf{p}_\mu^- := \mathsf{e}_{\mu-1}^-(t_{\mu-1,1}^-) = \mathsf{e}_\mu^-(t_{\mu,0}^-).$$

If the lines coincide, we choose the midpoint of the edge of $\square_\mu$. If they are parallel but not coincident, we return infinity so that the subsequent tolerance test triggers refinement (see Section 9). To see that such parallel lines are possible, although not likely, see Figure 15, *left*.

It is now tempting to connect the resulting sequence of vertices (p_μ^+), respectively (p_μ^-) to form the boundaries of the sleve. However Figure 11, *right*, proves that this does not yield an enclosure of all hexagons $\mathbb{H}_\mu$ and the curve would not be safely enclosed. The problem is that the segment $\mathsf{p}_\mu^-\mathsf{p}_{\mu+1}^-$ does not correspond to a proper segment of the line through e_μ^- since the parameter of intersection is decreasing rather than increasing. To rule out geometrically and combinatorially complex cases, we require that

$$t^+_{\mu,1} \geq t^+_{\mu,0}. \quad \text{and} \quad t^-_{\mu,1} \geq t^-_{\mu,0}. \tag{2}$$

Note that it is not correct to demand that the parameters lie in $[0, 1]$: this will not hold, even after repeated subdivision, for extension points such as p^+_μ in Figure 11 *left*. Since the segments are now well-defined, the quadrilateral

$$Q_\mu \quad \text{with vertices } \mathsf{p}^-_\mu, \mathsf{p}^+_\mu \mathsf{p}^+_{\mu+1}, \mathsf{p}^-_{\mu+1}$$

is well-defined and bounded by e^+_μ, $q_{\mu+1}$, e^-_μ and q_μ where

$$q_\mu := (1-s)\mathsf{p}^+_\mu + s\mathsf{p}^-_\mu,\ s \in [0..1].$$

We can conclude the following.

Lemma 6.1 *If (2) holds then each $\mathbb{H}_\mu$ is covered by a central quadrilateral Q_μ plus two, possibly empty, remainder pieces.*

7 Completion of a sequence of bounding box pairs

Figure 13 (b) and (c) show that certificate (2), which certifies pairwise correct bounds, is not sufficient to validate the enclosure. While the specific example could be flagged and tested, proving in the general case that $\mathbb{H}_\mu \backslash Q_\mu$ is covered is difficult. The main reason is that it can involve an arbitrary number of segments. Figure 13 shows two more examples where the immediately adjacent $Q_{\mu+1}$ does not cover the remainder of $\mathbb{H}_\mu$.

The challenge is then to formulate a certificate that is simple to check and is not too restrictive. To this end, we denote

$$\text{the relative interior of } \square_\mu \text{ by } \square^\circ_\mu,$$

e.g. the interior of the line segment if $\underline{x} = \overline{x}$ or $\underline{y} = \overline{y}$. The sufficient certificate (3)

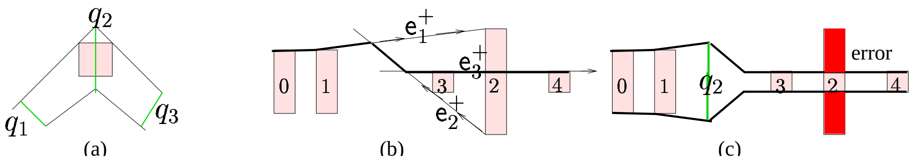


Fig. 13. (a) Generic case satisfying **certificate (3)**: $q_{\mu-1}$ and $q_{\mu+1}$ do not intersect $\square_\mu$ but q_μ does. (b) Sequence of e^+ that keep the $\square_\mu$ to their right. (Note the ordering!) The resulting sequence of intersected boundaries with increasing parameters is rendered thicker. It satisfies certificate (2) but not certificate (3)! (c) The upper and symmetric lower sequence incorrectly fails to enclose the red parts of $\square_2$.

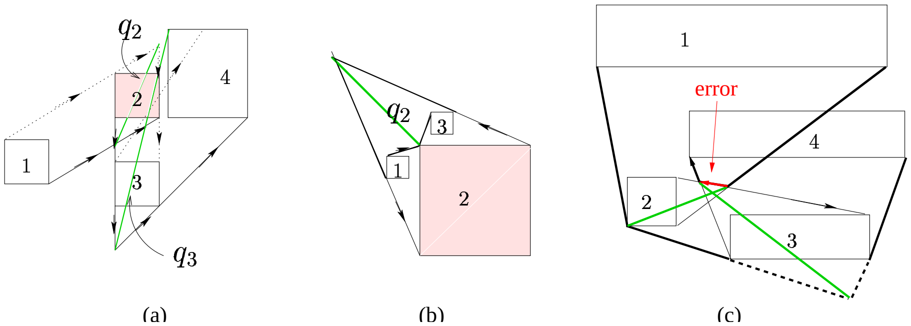


Fig. 14. (a) q_2 and q_3 intersect $\square_2$ so that $Q_{\mu-1}$ and Q_μ fail to bound $\square_\mu$. (b) q_2 does not intersect $\square_2$. (c) Certificate (3) is satisfied, but not certificate (2).

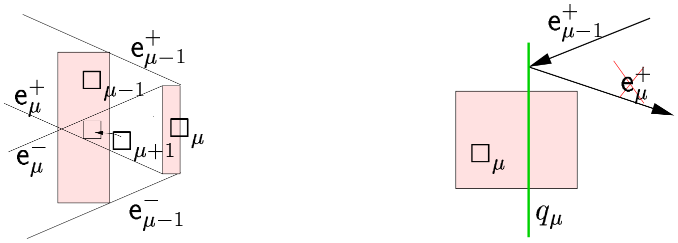


Fig. 15. (*left*) $\mathsf{e}^+_{\mu-1}$ and e^+_μ are parallel. (*right*) The displayed configuration of $\mathsf{e}^+_{\mu-1}$ and e^+_μ is incompatible with their definition: due to a reversal of orientation, $\square_\mu$ is 'to the left' of $\mathsf{e}^+_{\mu-1}$ but 'to the right' of e^+_μ.

is

$$q_\mu \text{ intersects } \square^\circ_\mu \quad \text{but} \quad q_{\mu-1} \text{ and } q_{\mu+1} \text{ do not intersect } \square^\circ_\mu. \tag{3}$$

It implies that $Q_{\mu-1}$ and Q_μ alone cover $\square_\mu$.

Lemma 7.1 *If (1), (2) and (3) hold then $\square_\mu$ is enclosed by $Q_{\mu-1}$ and Q_μ.*

Proof By assumption (3), q_μ partitions $\square_\mu$ into two non-empty parts. By (1), and (2), the quadrilaterals Q_μ are well-defined. By (3), the three other edges of Q_μ do not intersect $\square^\circ_\mu$. Therefore Q_μ encloses one part of $\square_\mu$. By the same logic, $Q_{\mu-1}$ also encloses one part of $\square_\mu$. If both parts were the same then $\mathsf{e}^+_{\mu-1}$ and e^+_μ lie on the same side of q_μ, but, since the direction is reversed, either $d_\mu(\mathsf{e}^+_\mu) \leq 0$ (see Figure 15, *middle*) or e^+_μ intersects $\square_\mu$. Both outcomes contradict the definition of e^+_μ. |||

Figure 13, *bottom*, shows examples where q_μ does not intersect $\square_\mu$ and therefore $d_\mu(\mathsf{e}^+_\mu) \geq 0$ but $\square_\mu$ is not covered.

Figure 13, *top,right*, shows that (2) does not imply (3). Conversely, Figure 15, *right*), shows an unsafe case where (2) is violated but (3) holds.

When one of the constraints fails we can either combine the offending point enclosures, if we are below the tolerance, or subdivide the curve, i.e. rerepresent it with a finer control structure and sleve over both half-domains.

Lemma 7.2 *If the curve is regularly parametrized, subdivision at the midpoint enforces Constraints (2) and (3) in the limit.*

Proof Without loss of generality the tangent line is the x-axis. Then, by Lemma 5.3, any triple $\square_{\mu-1}$, $\square_\mu$ and $\square_{\mu+1}$ is ordered implying that p^+_μ and p^-_μ are confined between the vertical lines through the minimum x of $\square_\mu$ and the maximum x of $\square_\mu$. Therefore the parameter of intersection is increasing and (2) holds; also, q_μ must intersect $\square_\mu$. Since the triple $\square_{\mu-1}$, $\square_\mu$ and $\square_{\mu+1}$ converges towards a straight line segment, q_μ can not intersect $\square_{\mu+1}$ in the limit. |||

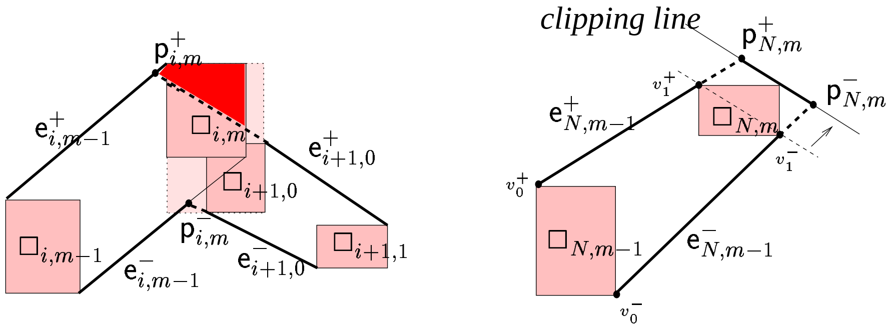


Fig. 16. (*left*) The two point enclosures need to be combined where **two curve pieces join**. Otherwise the connecting edge may incorrectly intersect as shown. (*right*) If a curve has **no adjacent curve** at its end, the line connecting $v^+_{m-1,1}$ and $v^-_{m-1,1}$ is parallel translated until $\square_{N,m}$ is enclosed. The line is called clipping line.

7.1 Endpoints of enclosures

When connecting a series of Bézier curves, linking p^ρ_{i,m_i} to $\mathsf{p}^\rho_{i+1,0}$ can leave a part of $\square_{i,m_i}$ uncovered as illustrated in Figure 16, *left*. We therefore enclose $\square_{i,m_i}$ and $\square_{i+1,0}$ in a minimal common axis-aligned point enclosure.

If $\square_{N,m_N}$ is the last point enclosure and has no neighbor then we select a line through a corner of $\square_{N,m_N}$ as a clipping line as shown in Figure 16, *right*. The clipping line is parallel to the line connecting v^+_1 and v^-_1 and maximizes the distance to v^+_0. The segment of the clipping line between the intersections p^-_m and p^+_m with the extreme line segments is the *clipping edge*. The clipping edge closes the sleve, so that Q_{N,m_N} covers $\square_{N,m_N}$.

8 The sleve of a piecewise Bézier spline

Definition 8.1 *Let* x *be a planar spline with pieces* $\mathsf{x}_i(t) := \sum_k \begin{bmatrix} x_{i,k} \\ y_{i,k} \end{bmatrix} \mathbf{b}_k(t)$ *for* $i = 1..N$. *If for all* $i = 1..N$ *and* $\mu = 0..m_i$, *the constraints (1), (2) and (3) hold,*

then the intersection points $\mathsf{p}^+_{i,\mu}$ and $\mathsf{p}^-_{i,\mu}$ are well-defined.
The sequence

$$\mathsf{p}^+_{0,0}, \mathsf{p}^+_{0,1}, \dots \mathsf{p}^+_{i,\mu}, \dots, \mathsf{p}^+_{N,m_N}, \mathsf{p}^-_{N,m_N}, \mathsf{p}^-_{N,m_N-1}, \dots, \mathsf{p}^-_{0,0}, \mathsf{p}^+_{0,0}$$

forms the sleve *of* x *with* x^+ *the polyline* $\mathsf{p}^+_{0,0}, \dots, \mathsf{p}^+_{N,m_N}$ *and* x^- *the polyline* $\mathsf{p}^-_{0,0}, \dots, \mathsf{p}^-_{N,m_N}$.

Lemma 8.1 *The* sleve *of a curve encloses the curve.*

Proof By Lemma 6.1, each Q_μ is well-defined. Since $Q_{\mu-1}$, Q_μ and $Q_{\mu+1}$ abut, Lemma 7.1 implies that all $\mathbb{H}_\mu$ are covered by the union of all Q_μ. |||

8.1 Tolerance test, subdivision or segmentation increase

Due to the extension to avoid gaps, the width of the sleve,

$$\max_{i,\mu}\{\|\mathsf{p}^+_{i,\mu} - \mathsf{p}^-_{i,\mu}\|_2\}$$

is hard to predict, initially. To give full control over tightness of the sleve, we check whether $\|\mathsf{p}^+_{i,\mu} - \mathsf{p}^-_{i,\mu}\|_2 \leq \mathtt{tol}_i$, where $\mathtt{tol}_i$ is the prescribed maximal width of the polynomial piece. If not, we trigger subdivision to meet the tolerance. Alternatively, somewhat analogous to the technique of ‘degree raising’, we can increase of number of segments, m_i.

The amount of work then depends on the number of subdivisions necessary to achieve the tolerances. Since the tables $a[..,..]$ are fixed (Wu and Peters, 2004), the number of subdivisions depend on the second differences. Each of the second differences reduces to $1/4$ after subdivision at $t = 1/2$.

8.2 Parametrization and coordinate system

Like many other spline computations, sleves depend on the spline’s parametrization rather than its intrinsic geometry. This can reduce the efficiency (without affecting the correctness) if the parametrization is more complex than the geometry. For example the optimal linear enclosure for $x(t) = y(t) = t^3$ with Bézier coefficients $\left[\begin{smallmatrix}0\\0\end{smallmatrix}\right]$, $\left[\begin{smallmatrix}0\\0\end{smallmatrix}\right]$, $\left[\begin{smallmatrix}0\\0\end{smallmatrix}\right]$, $\left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]$ on $U = [0..1]$ is the line segment itself but the width of the sleve computed from the parametrization is nonzero.

Certificates (2) and (3) did not depend on axis-align boxes $\square_\mu$. It then makes sense to consider whether it is computationally efficient to choose the local coordinate system to minimize the sleve width. This was suggested in (Karavelas et al., 2002). In the example, $x(t) = y(t) = t^3$, rotation of the x-axis to the interpolatory chord generates sleves with minimal width. However, when we conducted a statistical test with random orientations and modifications of a degree 4 curve, we did not observe a general reduction in width when the local coordinate system was rotated. Given the extra cost of such a rotation and increased complexity in forming sleves from rotated boxes, we decided not to make the rotation part of the basic algorithm.

9 The complete algorithm for planar curve sleves

To summarize the construction, we first characterize slefes from a computational point of view.

Definition 9.1 (univariate slefe) *Let $x := \sum \mathbf{b}_k x_k$ be a linear combination of basis functions $\mathbf{b}_k$ and coefficients x_k. For (small) integers d, m, $1 \leq \delta \leq d$, $1 \leq \mu \leq m$ and $sgn \in \{-1, +1\}$, let*

$$a[d, m, sgn, \delta, \mu]$$

be a table of real numbers derived according to Section 3. Then

$$\begin{aligned}
&\text{slefe}([x_0, \ldots, x_d], m, sgn) := [q_0, \ldots, q_m] \quad \textit{where} \\
&\quad q_\mu := x_0(1 - \frac{\mu}{m}) + x_d(\frac{\mu}{m}) + \sum_{\delta=1}^{d-1} \Delta_\delta x \; a[d, m, sign(\Delta_\delta x) \times sgn, \delta, \mu] \\
&\quad \Delta_\delta x := x_{\delta-1} - 2x_\delta + x_{\delta+1}.
\end{aligned}$$

Let $\mathbf{h}_\mu^m$ be the piecewise linear hat function with break points at $\frac{j}{m}$, $j = 0, \ldots, m$ that is 1 at $\frac{\mu}{m}$ and 0 at all other break points. Then the m-piecewise linear upper and lower component of the slefe are

$$\begin{aligned}
\overline{x}(t) &:= \sum_{\mu=0}^{m} \tilde{x}_\mu \mathbf{h}_\mu^m(t), \quad \textit{where} \quad [\tilde{x}_0, \ldots, \tilde{x}_m] := \text{slefe}([x_0, x_1, \ldots, x_d], m, +1) \\
\underline{x}(t) &:= \sum_{\mu=0}^{m} \underset{\sim}{x}_\mu \mathbf{h}_\mu^m(t), \quad \textit{where} \quad [\underset{\sim}{x}_0, \ldots, \underset{\sim}{x}_m] := \text{slefe}([x_0, x_1, \ldots, x_d], m, -1).
\end{aligned}$$

We now summarize the construction.

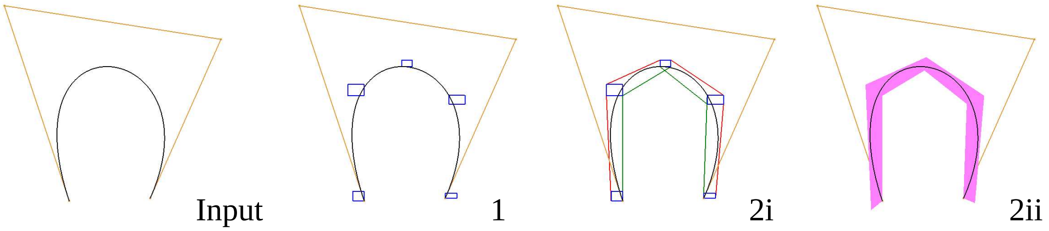


Fig. 17. Steps of the sleve construction, with one input curve piece and $d = 3, m = 4$.

Input:
A spline curve x represented as a linked list of its polynomial curve pieces

$$\mathsf{x}_i(t) := \sum_k \begin{bmatrix} x_{i,k} \\ y_{i,k} \end{bmatrix} \mathbf{b}_k(t),\ i = 1..N.$$

in order. Each piece is defined by
- its $d_i + 1$ Bézier coefficients,
- the number m_i of linear segments per curve piece, and
- the maximal width tolerance, $\mathtt{tol}_i$.

(Of course, d_i, m_i and $\mathtt{tol}_i$ may be the same for all pieces).

Output: A sleve (x^+,x^-) defined by breakpoints p^+_{i,μ_i} and p^-_{i,μ_i} for $i = 1..N$ and $\mu = 1..m_i$ such that max $\|\mathsf{p}^+_{i,\mu} - \mathsf{p}^-_{i,\mu}\|_2 \leq \mathtt{tol}_i$.

Initialize e^+_o and e^-_o to the start clipping edge and $t^+_o = t^-_o \leftarrow -\infty$.

(1) (create component enclosures)

For each Bézier curve piece x_i, for each component $\chi \in \{x, y\}$ use the tables $a[d_i, m_i, .., .., ..]$ (e.g. from (Wu and Peters, 2004)) to compute

$$[\overline{\chi_0}, \dots, \overline{\chi_{m_i}}] \leftarrow \mathsf{slefe}([\chi_0, \dots, \chi_{d_i}], m_i, +1),$$
$$[\underline{\chi_0}, \dots, \underline{\chi_{m_i}}] \leftarrow \mathsf{slefe}([\chi_0, \dots, \chi_{d_i}], m_i, -1).$$

If $i > 0$, replace $\square_{i-1,m_{i-1}}$ and $\square_{i,0}$ by their smallest enclosing axis-aligned box.

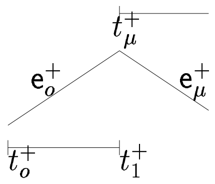
(2) For each Bézier curve piece x_i, for each segment $\mu = 0..m_i - 1$,

i (select extreme line segments)

Select e^+_μ and e^-_μ using Table T.
Test Certificate (1):
If (4a) or (4b) replace $\square_\mu$ and $\square_{\mu+1}$ by their smallest enclosing box.

ii (intersect two extreme line segments)

Intersect e^+_o and e^+_μ to get
$\mathsf{p}^+_\mu := \mathsf{e}^+_o(t^+_1) = \mathsf{e}^+_\mu(t^+_\mu)$ and
intersect e^-_o and e^-_μ to get
$\mathsf{p}^-_\mu := \mathsf{e}^-_o(t^-_1) = \mathsf{e}^-_\mu(t^-_\mu)$.



Test Certificate (2):
If ($(t^+_1 < t^+_o)$ or $(t^-_1 < t^-_o)$), Refine. \\ reverse edge

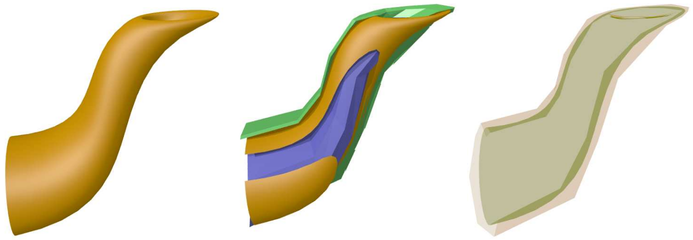


Fig. 18. Teapot spout and sleve. *left to right:* The surface, a cut through inner sleve and surface, transparent rendering of inner and out sleve and surface.

```
        If ($\|\mathsf{p}^+_\mu - \mathsf{p}^-_\mu\|_2 > \mathtt{tol}_\mu$), Refine.        \\ width too large
        Test Certificate (3):
        If $q_o$ does intersect $\Box^\circ_\mu$, Refine.
        If $q_\mu$ does not intersect $\Box^\circ_\mu$, Refine.
        If $q_\mu$ does intersect $\Box^\circ_{\mu-1}$, Refine.
    iii Update $t^+_o \leftarrow t^+_\mu, t^-_o \leftarrow t^-_\mu$,   $\mathsf{e}^+_o \leftarrow \mathsf{e}^+_\mu, \mathsf{e}^-_o \leftarrow \mathsf{e}^-_\mu$,   $q_o \leftarrow q_\mu$.
 Set $\mathsf{e}^+_\mu$, $\mathsf{e}^-_\mu$ to the end clipping edge and do step (2ii).
```

Refine stands for:
replace x by the two curves obtained by subdividing x at $t = 1/2$, and continue at (1) for each piece.

The tolerance requirement catches the case in 2.ii where lines are parallel but not equal and resolves cases 4a and 4b of table T.

10 Extensions and Applications

By knot insertion, the construction of NURBS sleves from sleves of splines in Bézier representation is immediate. For splines with a uniform knot sequence, however, it is worth using the slefes in (Lutterkort and Peters, 2001b). The midpath $\begin{bmatrix}\overline{x}(t)+\underline{x}(t)\\ \overline{y}(t)+\underline{y}(t)\end{bmatrix}/2$ provides a good linearization of the curve $(x(t), y(t))$ but does not require without the construction of a sleve. To help in intersection testing, sleves can be applied to the normal of a curve.

Conceptually, surface sleves can be bootstrapped from curve sleves as illustrated in Figure 18. A new aspect is the determination of the vertices of the bounding triangulations which can no longer be computed simply by intersecting two lines.

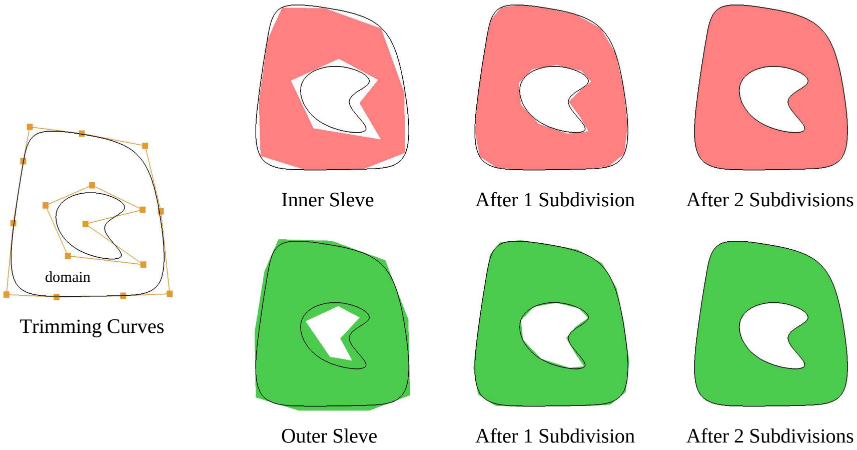


Fig. 19. (*left) two trimming curves; (upper row) the area between the inner* sleve*s; (lower row) the area between the outer* sleve*s.*

10.1 Application: Polygonalization of curve trimmed domain

Planar spline curves are frequently used in CAGD as trimming curves, in the parameter domain (see Figure 19 left). To render trimmed surfaces, the curved domain needs to be linearized. Linearizing by sampling can result both in missing and in extraneous surface pieces. Choosing the area between the two outer sleves (Figure 19, lower row) creates a polygonal region guaranteed to cover the entire domain. Using the inner sleves guarantees that no extraneous piece is rendered. Subdivision or increasing the number m of segments improves the approximation to any prescribed tolerance.

10.2 Application: separation test

For curved objects in close proximity, Figure 20, *left*, *middle*, illustrates the effectiveness of sleves. Here a convex-hull test fails to detect separation even after subdividing the pieces. However, narrowness of the bounding construct is just one of several criteria for usefulness in intersection testing: simpler constructs like axis-aligned bounding boxes and spheres provide faster rejection tests in sparse settings.

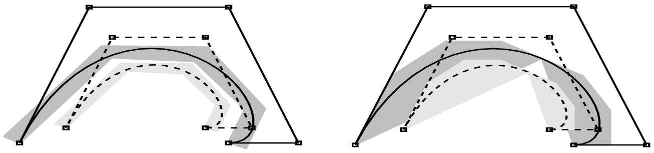


Fig. 20. **Separation test** between two nested curve segments of degree $d = 4$. (*left*) Two sleves with $m = 4$ (light grey and dark grey) immediately prove separation. (*right*) The convex hulls of the two curve segments intersect even after one subdivision.

We are currently testing whether tightness pays off in higher dimensions as the cost of refinement increases and precision pays.

11 Summary

Even for planar curves, the construction of a correct, tight piecewise linear enclosure is nontrivial. The sleve construction described in this paper comes with a proof and is relatively simple to implement. It provides bounds on the width and two mechanisms for adaptively reducing the width of the enclosure.

References

Basch, J., 1999. Kinetic data structures. Ph.D. thesis, Stanford University.

Buck, R. C., 1965. Applications of duality in approximation theory. In: Approximation of Functions (Proc. Sympos. General Motors Res. Lab., 1964). Elsevier Publ. Co., Amsterdam, pp. 27–42.

Cohen, J., Varshney, A., Manocha, D., Turk, G., Weber, H., Agarwal, P., Brooks, Jr., F. P., Wright, W., Aug. 1996. Simplification envelopes. In: Rushmeier, H. (Ed.), SIGGRAPH 96 Conference Proceedings. Annual Conference Series. ACM SIGGRAPH, Addison Wesley, pp. 119–128, held in New Orleans, Louisiana, 04-09 August 1996.

Crosnier, A., Rossignac, J. R., Jun. 1999. Technical section — tribox bounds for three-dimensional objects. Computers and Graphics 23 (3), 429–437.

Farin, G., 1997. Curves and Surfaces for Computer-Aided Geometric Design: A Practical Guide, 4th Edition. Academic Press, New York, NY, USA.

Filip, D., Magedson, R., Markot, R., 1986. Surface algorithms using bounds on derivatives. Computer Aided Geometric Design 3 (4), 295–311.

Gottschalk, S., Lin, M. C., Manocha, D., 1996. OBBTree: A hierarchical structure for rapid interference detection. Computer Graphics 30 (Annual Conference Series), 171–180.

Hu, C., Maekawa, T., Sherbrooke, E. C., Patrikalakis, N. M., 1996a. Robust interval algorithm for curve intersections. Computer-aided Design 28 (6-7), 495–506.

Hu, C., Patrikalakis, N. M., Ye, X., 1996b. Robust interval solid modelling part I: representations. Computer-aided Design 28 (10), 807–817.

Hu, C., Patrikalakis, N. M., Ye, X., 1996c. Robust interval solid modelling part II: boundary evaluation. Computer-aided Design 28 (10), 819–830.

Karavelas, M., Kaklis, P., Kostas, K., 2002. Alternative bounds on the distance between 2d parametric bezier curves and their control polygon. Tech. rep., presented at the Dagstuhl Seminar on Geometric Modelling.

Kay, T. L., Kajiya, J. T., Aug. 1986. Ray tracing complex scenes. In: Evans, D. C., Athay, R. J. (Eds.), Computer Graphics (SIGGRAPH '86 Proceedings). Vol. 20. pp. 269–278.

Klosowski, J. T., May 1998. Efficient collision detection for interactive 3d graphics and virtual environments. Ph.D. thesis, State Univ. of New York at Stony Brook.

Klosowski, J. T., Mitchell, J. S. B., Sowizral, H., Zikan, K., Jan. 1998. Efficient Collision Detection Using Bounding Volume Hierarchies of k-DOPs. IEEE Transactions on Visualization and Computer Graphics 4 (1), 21–36.

Kobbelt, L., Oct. 26–29 1998. Tight bounding volumes for subdivision surfaces. In: Werner, B. (Ed.), Proceedings of the Conference on Computer Graphics and Applications (Pacific-Graphics'98). IEEE, Los Alamitos, pp. 17–26.

Kobbelt, L., Daubert, K., Seidel, H.-P., Jun. 1998. Ray tracing of subdivision surfaces. In: Rendering Techniques '98 (Proceedings of the Eurographics Workshop). Springer-Verlag, New York, pp. 69–80.

Lutterkort, D., May 2000. Envelopes for nonlinear geometry. Ph.D. thesis, Purdue University.

Lutterkort, D., Peters, J., 1999. Smooth paths in a polygonal channel. In: Proceedings of the 15th annual symposium on Computational Geometry. pp. 316–321.

Lutterkort, D., Peters, J., 2001a. Optimized refinable enclosures of multivariate polynomial pieces. Comput. Aided Geom. Design 18 (9), 851–863.

Lutterkort, D., Peters, J., May 2001b. Tight linear bounds on the distance between a spline and its B-spline control polygon. Numerische Mathematik 89, 735–748.

Nairn, D., Peters, J., Lutterkort, D., Aug 1999. Sharp, quantitative bounds on the distance between a polynomial piece and its Bézier control polygon. Computer Aided Geometric Design 16 (7), 613–633.

Peters, J., Wu, X., 2003. On the optimality of piecewise linear max-norm enclosures based on slefes. In: Proceedings of the 2002 St Malo conference on Curves and Surfaces. pp. 335–344.

Pinkus, A. M., 1989. On L^1-approximation. Cambridge University Press, Cambridge.

Prautzsch, H., Boehm, W., Paluszny, M., 2002. Bézier and B-spline techniques. Springer Verlag, Berlin, Heidelberg.

Reif, U., 2000. Best bounds on the approximation of polynomials and splines by their control structure. Comput. Aided Geom. Design 17 (6), 579–589.

Sander, P., Gu, X., Gortler, S., Hoppe, H., Snyder, J., 2000. Silhouette clipping. Computer Graphics 34 (Annual Conference Series), 327–334.

Sederberg, T. W., Farouki, R. T., Sep. 1992. Approximation by interval bezier curves. IEEE Computer Graphics and Applications 12 (5), 87–95.

Sederberg, T. W., White, S. C., Zundel, A. K., 1989. Fat arcs: A bounding region with cubic convergence. Comput. Aided Geom. Design 6, 205–218.

Shen, G., Patrikalakis, M., 1998. Numerical and geometric properties of interval b-splines. In: International Journal of Shape Modeling. Vol. 4. pp. 35–62.

Tuohy, S. T., Maekawa, T., Shen, G., Patrikalakis, N. M., 1997. Approximation of measured data with interval B-splines. Computer-aided Design 29 (11), 791–799.

Welzl, E., Jun. 1991. Smallest enclosing disks (balls and ellipsoids). In: Maurer, H.

(Ed.), Proceedings of New Results and New Trends in Computer Science. Vol. 555 of LNCS. Springer, Berlin, Germany, pp. 359–370.

Wu, X., Peters, J., 2004. The SubLiME package (Subdividable Linear Maximum-norm Enclosure). Downloadable from http://www.cise.ufl.edu/research/SurfLab/SubLiME/.