

Optimized Refinable Enclosures of Multivariate Polynomial Pieces

David Lutterkort and Jörg Peters

June 26, 2001

Abstract

An enclosure is a two-sided approximation of a uni- or multivariate function $\mathbf{b} \in \mathcal{B}$ by a pair of typically simpler functions $\mathbf{b}^+, \mathbf{b}^- \in \mathcal{H} \neq \mathcal{B}$ such that $\mathbf{b}^- \leq \mathbf{b} \leq \mathbf{b}^+$ over the domain U of interest. Enclosures are optimized by minimizing the width $\max_U \mathbf{b}^+ - \mathbf{b}^-$ and refined by enlarging the space \mathcal{H} . This paper develops a framework for efficiently computing enclosures for multivariate polynomials and, in particular, derives piecewise bilinear enclosures for bivariate polynomials in tensor-product Bézier form. Runtime computation of enclosures consists of looking up $s < \dim \mathcal{B}$ pre-optimized enclosures and linearly combining them with the second differences of \mathbf{b} . The width of these enclosures scales by a factor 1/4 under midpoint subdivision.

1 Introduction

An enclosure is an explicit two-sided approximation $\mathbf{b}^+, \mathbf{b}^-$ of a function \mathbf{b} so that $\mathbf{b}^- \leq \mathbf{b} \leq \mathbf{b}^+$ over the domain U of interest. If \mathbf{b}^+ and \mathbf{b}^- are simpler functions the enclosure offers the advantages of any simpler but tight bounding construct: easier point classification and intersection testing. The approximation is the more useful the smaller $\max_U \mathbf{b}^+ - \mathbf{b}^-$, the width of the enclosure.

In [7, 6, 9] the distance of piecewise polynomials to their Bézier or B-spline control net was bounded in terms of second differences. We extend these results but do not insist on enclosing the control polygon. Compared to [7, 9] this yields, often dramatically, tighter enclosures for \mathbf{b} .

The enclosures defined in this paper are *efficiently computable* and their width is affinely invariant. The key idea is to write the difference between $\mathbf{b} \in \mathcal{B}$ and a function $\mathbf{h} \in \mathcal{H}$, say the control polygon of \mathbf{b} , as a linear combination of a fixed set of functions whose enclosures have been precomputed and tabulated once and for all; tables for bivariate polynomials in tensor-product Bézier form and Matlab routines for generating tables are available at [3]. The tabulation reduces the cost for computing the enclosure of a specific \mathbf{b} to retrieving the values and linearly combining them with second differences of \mathbf{b} .

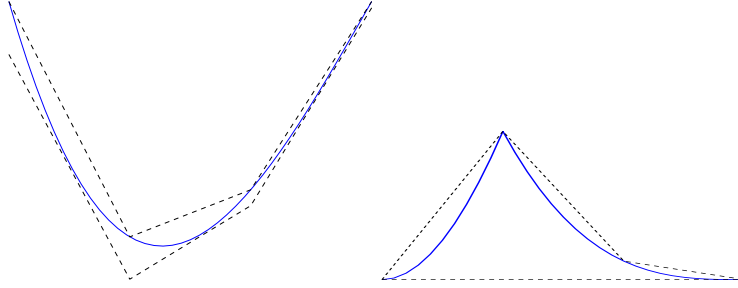


Figure 1: (left) Piecewise linear upper and lower bounds of the univariate cubic function \mathbf{b} with coefficients $0, -2/3, -1/3, 0$. (right) An enclosure of $\mathbf{b} - L\mathbf{b}$, the function minus its control polygon.

2 An example in one variable

We consider a cubic polynomial \mathbf{b} in Bézier representation [1]

$$\mathbf{b}(u) := \sum_{i=0}^3 b_i B_i(u) =: B(u)b, \quad B_i(u) := \frac{3!}{(3-i)!i!} (1-u)^{3-i} u^i.$$

In the short notation $\mathbf{b} := Bb$, B is the vector of basis function with $B(i) = B_i$ and $b(i) = b_i \in \mathbb{R}$ are the Bézier control points. Let $\{H_i\}, i = 0, 1, 2, 3$ be a basis for the space \mathcal{H} of piecewise linear functions with break points at $i/3$ (page 5 gives a formal definition). On the interval $U = [0, 1]$ we want to enclose \mathbf{b} from above and from below by functions $\mathbf{h} \in \mathcal{H}$ like the one drawn as dashed lines in Figures 1 and 2. Let $\lfloor B \rfloor$ be a vector of four functions from \mathcal{H} so that the i th function bounds the i th basis function, B_i , from below on the interval U . Similarly, $\lceil B \rceil(i) \geq B_i$ on U and

$$b_-(i) := \min\{b_i, 0\}, \quad b_+(i) := \max\{b_i, 0\}.$$

Then (recall our short notation) it is easily checked that on U

$$\lfloor B \rfloor b_+ + \lceil B \rceil b_- \leq \mathbf{b} \leq \lfloor B \rfloor b_- + \lceil B \rceil b_+.$$

However, the difference between upper and lower bound,

$$\text{diff}(\mathbf{b}) := (\lceil B \rceil - \lfloor B \rfloor)|b|, \quad |b|(i) := \text{abs}(b_i),$$

is not invariant even under the addition of constant functions: for example, if $k \gg \text{abs}(b_i)$ for all i , then

$$\text{diff}(\mathbf{b} + k) \approx k \sum_i \lceil B_i \rceil - \lfloor B_i \rfloor$$

increases linearly with k .

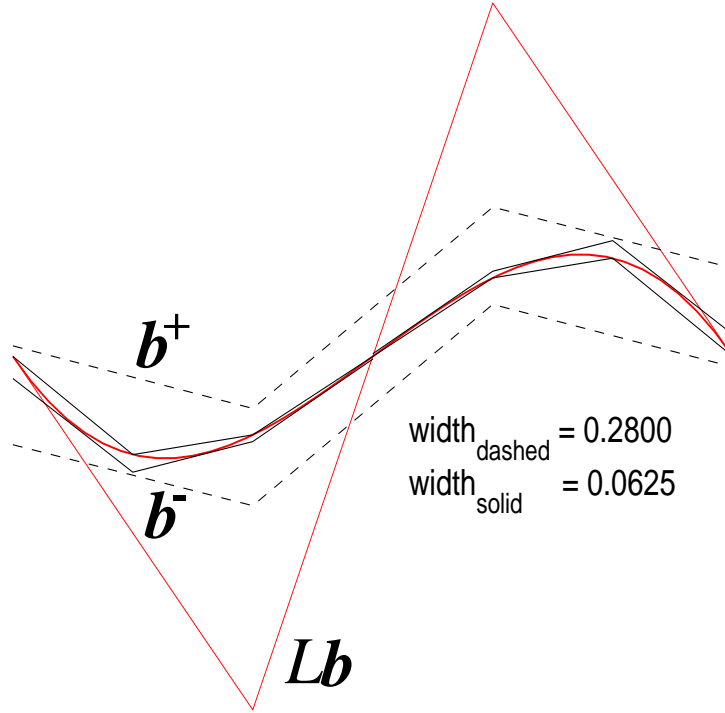


Figure 2: A cubic Bézier segment with coefficients $0, -1, 1, 0$. The control polygon $L\mathbf{b}$ exaggerates the curve far more than the *dashed* enclosure $\mathbf{b}^+, \mathbf{b}^-$. Midpoint subdivision yields an even tighter enclosure displayed as the matched pair of *solid* polygons enclosing the cubic.

Since an enclosure that becomes arbitrarily wide under mere translation is not desirable we define an enclosure using (the vector of) second differences

$$\Delta\mathbf{b} := \begin{bmatrix} \Delta_1\mathbf{b} \\ \Delta_2\mathbf{b} \end{bmatrix} := \begin{bmatrix} b_0 - 2b_1 + b_2 \\ b_1 - 2b_2 + b_3 \end{bmatrix}$$

which maps constant and linear functions (in cubic Bézier representation) to the tuple $\begin{bmatrix} 0 \\ 0 \end{bmatrix}$. We also define $A := [A_1, A_2]$ as a 2-vector of cubic polynomials whose first and last Bézier coefficient are zero. Specifically, A_1 has coefficients $0, -2/3, -1/3, 0$ (Figure 1) and A_2 coefficients $0, -1/3, -2/3, 0$. We check that $\Delta A_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and $\Delta A_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, i.e. $\Delta_i A_j = 1$ for $i = j$ and 0 else. Then $\Delta A \Delta\mathbf{b} = \Delta\mathbf{b}$ and with $\ell(\mathbf{b})$ the linear interpolant to $\mathbf{b}(0)$ and $\mathbf{b}(1)$

$$\mathbf{b} - \ell(\mathbf{b}) = A \Delta\mathbf{b} = A_1 \Delta_1\mathbf{b} + A_2 \Delta_2\mathbf{b}.$$

Finally, let $L\mathbf{b}$ be the control polygon of \mathbf{b} , i.e. the function in \mathcal{H} that interpolates the control points of \mathbf{b} . Since linear functions are exactly reproduced by their control

polygon, i.e. $L\ell(\mathbf{b}) = \ell(\mathbf{b})$,

$$\begin{aligned}\mathbf{b} - \ell(\mathbf{b}) - L\mathbf{b} &= A\Delta\mathbf{b} - L(\ell(\mathbf{b}) + A\Delta\mathbf{b}) \\ \iff \mathbf{b} - L\mathbf{b} &= (A - LA)\Delta\mathbf{b}.\end{aligned}$$

We can now apply the earlier double-inequality estimate, this time to $\mathbf{b} - L\mathbf{b}$. Adding $L\mathbf{b}$ to all three terms of the result yields

$$\begin{aligned}\mathbf{b}^+ &:= L\mathbf{b} + \lfloor A - LA \rfloor (\Delta\mathbf{b})_- + \lceil A - LA \rceil (\Delta\mathbf{b})_+ \\ &\geq \mathbf{b} \geq \\ \mathbf{b}^- &:= L\mathbf{b} + \lfloor A - LA \rfloor (\Delta\mathbf{b})_+ + \lceil A - LA \rceil (\Delta\mathbf{b})_-, \end{aligned}$$

where e.g. $\lfloor A - LA \rfloor (\Delta\mathbf{b})_+ = \sum_k \lfloor A_{\kappa} - LA_{\kappa} \rfloor \max\{\Delta_k \mathbf{b}, 0\}$ and $\lfloor A_{\kappa} - LA_{\kappa} \rfloor$ is a piecewise linear function in \mathcal{H} that bounds $A_{\kappa} - LA_{\kappa}$ from below. Figure 1, *right* shows an enclosure of $A_1 - LA_1$ whose upper piece is tabulated as the four values 0, 8/27, 1/27, 0 at the break points. Defining $|\Delta\mathbf{b}|(i) := \text{abs}(\Delta_i \mathbf{b})$ we get

$$\text{width}_U(\mathbf{b}) := \max_U \mathbf{b}^+ - \mathbf{b}^- = \max_U (\lceil A - LA \rceil - \lfloor A - LA \rfloor) |\Delta\mathbf{b}|,$$

invariant under addition of constant and linear terms to \mathbf{b} ; and $(\mathbf{b}^+ + \mathbf{b}^-)/2$ is an approximation to \mathbf{b} with a maximal error $\text{width}_U(\mathbf{b})/2$. Moreover, with \mathbf{b}^1 and \mathbf{b}^2 the cubic pieces resulting from de Castel'jau evaluation of \mathbf{b} at 1/2 (midpoint subdivision) and $\|\Delta\mathbf{b}\|_{\infty} := \max\{|\Delta_1 \mathbf{b}|, |\Delta_2 \mathbf{b}|\}$, Lemma 6.1 of [7] shows that

$$\max\{\|\Delta\mathbf{b}^1\|_{\infty}, \|\Delta\mathbf{b}^2\|_{\infty}\} \leq \frac{1}{4} \|\Delta\mathbf{b}\|_{\infty}.$$

That is, one subdivision cuts the width to *a quarter or less* as illustrated by the comparison of the dashed with the solid enclosure in Figure 2. Figure 2 also shows that if a component of $\lfloor A - LA \rfloor$ takes on positive values or a component of $\lceil A - LA \rceil$ takes on negative values then the control polygon $L\mathbf{b}$ need not lie in the enclosure and \mathbf{b}^+ and \mathbf{b}^- can approximate \mathbf{b} with *less* error than $L\mathbf{b}$.

3 Bivariate function enclosures

A tensor-product polynomial $\mathbf{b}(u, v)$ of degree d_1, d_2 is in Bézier form on the unit square $[0, 1]^2$ if

$$\begin{aligned}\mathbf{b} : [0, 1]^2 &\rightarrow \mathbb{R}, \\ \mathbf{b}(u, v) &:= \sum_i^{d_1} \sum_j^{d_2} b_{ij} B_i^{d_1}(u) B_j^{d_2}(v), \\ \text{where } B_k^d(t) &:= \frac{d!}{(d-k)!k!} (1-t)^{d-k} t^k.\end{aligned}$$

We abbreviate $\mathbf{b} = Bb$ where b is the vector of Bézier coefficients b_{ij} and B the vector of basis functions $B_{i,j}^{d_1, d_2} := B_i^{d_1} B_j^{d_2}$. We denote the space spanned by the functions $B_i^{d_1} B_j^{d_2}$ as \mathcal{B} .

We obtain the *control net* of the patch by connecting b_{ij} to $b_{i+1,j}$, $b_{i,j+1}$, $b_{i-1,j}$ and $b_{i,j-1}$ whenever all subscripts lie between 0 and the degree. The piecewise *bilinear* interpolant to each quadrilateral of the control net is the *control polyhedron* $L\mathbf{b}$

$$L\mathbf{b} : [0, 1]^2 \rightarrow \mathbb{R},$$

$$L\mathbf{b}(u, v) := bH(u, v) := \sum_i^{d_1} \sum_j^{d_2} b_{ij} H_i^{d_1}(u) H_j^{d_2}(v),$$

$$\text{where } H_k^d(t) := \begin{cases} dt - (k-1) & \text{if } \frac{k-1}{d} \leq t \leq \frac{k}{d} \\ (k+1) - dt & \text{if } \frac{k}{d} \leq t \leq \frac{k+1}{d} \\ 0 & \text{else.} \end{cases}$$

The space $\mathcal{H} := \text{span} \{H_i^{d_1}(u) H_j^{d_2}(v)\}$ is spanned by the piecewise bilinear hat functions with breakpoints at the Greville abscissae $t_{\alpha\beta} = (\alpha/d_1, \beta/d_2)$, $\alpha \in \{0, 1, \dots, d_1\}$, $\beta \in \{0, 1, \dots, d_2\}$.

If we want to enclose tensor-product polynomials over the unit square by piecewise bilinear functions, several direct approaches to tensoring the univariate enclosure construction do not succeed. We cannot use the enclosure of $\sum_i^{d_1} b_{ij} B_i^{d_1}(u)$ to provide a quantity q_j that can then be used for enclosing $\sum_j^{d_2} q_j B_j^{d_2}(v)$ since the enclosure is a piecewise linear function. We also cannot compute $[A_\kappa - LA_\kappa]$ from the univariate enclosures since tensoring the basis A_i of the previous section yields a space of dimension 4 while the space of bicubic functions that are not bilinear is of dimension 12. Finally, we note that in two or more variables there are several distinct types of second difference operators that all annihilate the bilinear functions in $\mathcal{B} \cap \mathcal{H}$. Different choices of Δ imply different choices of A , and, at least in the two cases discussed below, neither choice leads consistently to smaller widths.

Generalizing the approach to a **bilinear enclosure construction** is nevertheless possible as follows:

1. Choose a vector Δ of second difference operators Δ_i that annihilate bilinear polynomials, i.e. the functions in $\mathcal{B} \cap \mathcal{H}$.
2. Compute a basis A of size $s = (d_1 + 1)(d_2 + 1) - 4$ dual to Δ from the requirements
 - (i) $\Delta_i A_j = 1$ if $i = j$ and $\Delta_i A_j = 0$ otherwise,
 - (ii) $A_j(0, 0) = A_j(0, 1) = A_j(1, 0) = A_j(1, 1) = 0$.
3. Compute (and tabulate) piecewise bilinear upper and lower bounds for $\kappa = 1, \dots, s$

$$[A_\kappa - LA_\kappa] \leq A_\kappa - LA_\kappa \leq [A_\kappa - LA_\kappa] \text{ on } U.$$

(For example, the piecewise bilinear lower bound function $[A_\kappa - LA_\kappa]$ of a bicubic A_k is tabulated by recording its 16 values at the breakpoints $(\alpha/3, \beta/3)$.)

4. Weigh and sum the upper and lower bounds according to the sign of the second differences to obtain \mathbf{b}^- and \mathbf{b}^+ , e.g.
$$\mathbf{b}^+ := L\mathbf{b} + [A - LA](\Delta\mathbf{b})_- + [A - LA](\Delta\mathbf{b})_+.$$

We consider two families of difference operators in detail. The first, Δ^{cnv} , is of interest because the corresponding A is directionally convex and easily approximated by bilinear functions. The second, Δ^{gra} , minimizes the computational effort at runtime. Both for Δ^{cnv} and Δ^{gra} we associate one difference operator with each index (ij) of coefficients b_{ij} except for the corners (00) , $(d_1 0)$, $(0 d_2)$, $(d_1 d_2)$ where $b_{ij} = 0$ by 2(ii) above. To formalize this association, let κ be a map defined on the indices of all non-corner control points that maps bijectively into $\{1, \dots, s\}$, i.e. the $\kappa(i, j)$ -th difference operator Δ_κ is associated with the control point b_{ij} .

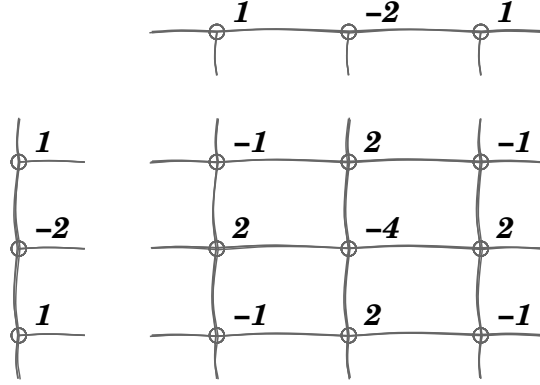


Figure 3: Second difference masks for tensor-products: the univariate second differences Δ^{uni} in either parameter direction (*top and left*) and the tensor-product of the masks (*bottom right*).

3.1 The directionally convex basis A^{cnv}

A tensor-product Bernstein polynomial \mathbf{b} is directionally convex if every isoparameter curve $\mathbf{b}(u, \cdot)$ and $\mathbf{b}(\cdot, v)$ is convex [10]. This is the case when the control polygon is convex in both parameter directions. In terms of the vector of univariate difference masks $b_{i-1,j} - 2b_{ij} + b_{i+1,j}$ and $b_{i,j-1} - 2b_{ij} + b_{i,j+1}$ (c.f. Figure 3) \mathbf{b} is directionally convex if $\Delta^{\text{uni}}\mathbf{b} \geq 0$. Such polynomials are bounded from above by their piecewise bilinear interpolant at the Greville abscissa and from below by their control polygon.

We define Δ^{cnv} using the difference masks shown in Figure 3. A difference operator $\Delta_{\kappa(ij)}^{\text{cnv}}$ on the boundary is the univariate second difference along that boundary centered at b_{ij} . Interior differences are the tensor-products of two univariate differences centered at b_{ij} . We define a basis A with $\Delta^{\text{cnv}}A = I$ with the help of ramp-like function r_i^d from $\{0, \dots, d\}$ to the integers:

$$r_i^d(\alpha) := \begin{cases} \alpha(d-i) & \alpha \leq i \\ i(d-\alpha) & \text{otherwise.} \end{cases}$$

For example, r_1^3 takes on the values 0, 2, 1, 0. Then the coefficients with index (α, β)

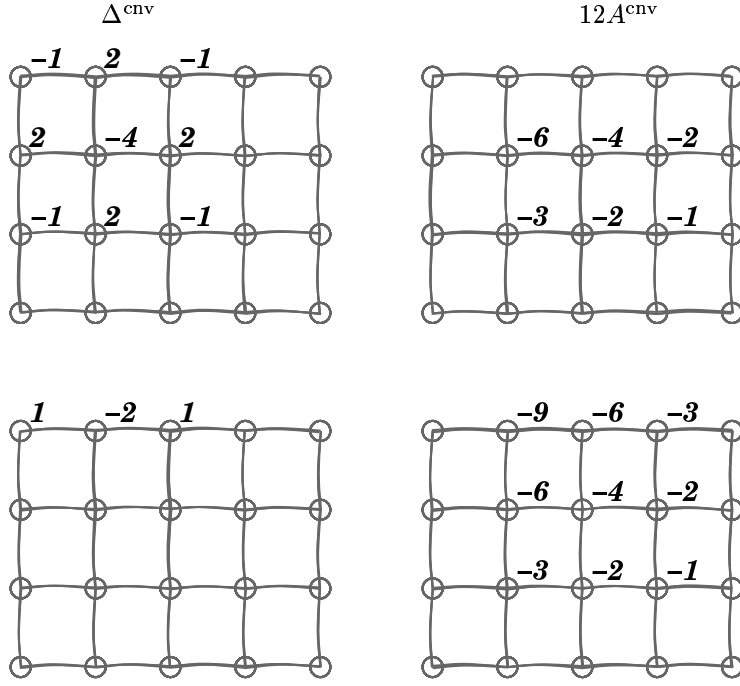


Figure 4: Two second difference masks $\Delta_{\kappa}^{\text{cnv}}$ (left) and the corresponding control nets A_{κ}^{cnv} (right) for tensor product polynomials of bidegree 3×4 . The control nets A_{κ}^{cnv} are scaled by 12.

of A_{κ} for $\kappa = \kappa(i, j)$ is defined by the control points (c.f. Figure 4, right)

$$A_{\kappa(i,j),\alpha,\beta}^{\text{cnv}} := \frac{-1}{d_1 d_2} \begin{cases} (d_2 - \beta) r_i^{d_1}(\alpha) & j = 0, \\ \beta r_i^{d_1}(\alpha) & j = d_2, \\ (d_1 - \alpha) r_j^{d_2}(\beta) & i = 0, \\ \alpha r_j^{d_2}(\beta) & i = d_1, \\ r_i^{d_1}(\alpha) r_j^{d_2}(\beta) & \text{in the interior.} \end{cases}$$

Since A_{κ} is linear in at least one direction except at t_{ij} , $\Delta_{\mu} A_{\kappa} = 0$ for $\kappa \neq \mu$ and $\Delta_{\kappa} A_{\kappa} = 1$ as required. (Check also in Figure 4 that each difference mask applied to the diagonally opposite A_{κ} yields zero and while application to the A_{κ} in the same row yields 1.) Since $\Delta^{\text{uni}} A \geq 0$, i.e. the functions A induced by Δ^{cnv} are directionally convex and

$$[A_{\kappa} - LA_{\kappa}]^* := 0 \quad [A_{\kappa} - LA_{\kappa}]^* := \sum_{i=0}^{d_1} \sum_{j=0}^{d_2} (A_{\kappa}(t_{ij}) - A_{\kappa(i,j)}^{\text{cnv}}) H_i^{d_1} H_j^{d_2}$$

form a piecewise bilinear enclosure of $A_{\kappa} - LA_{\kappa}$. In fact $[A_{\kappa} - LA_{\kappa}]^*$ is optimal

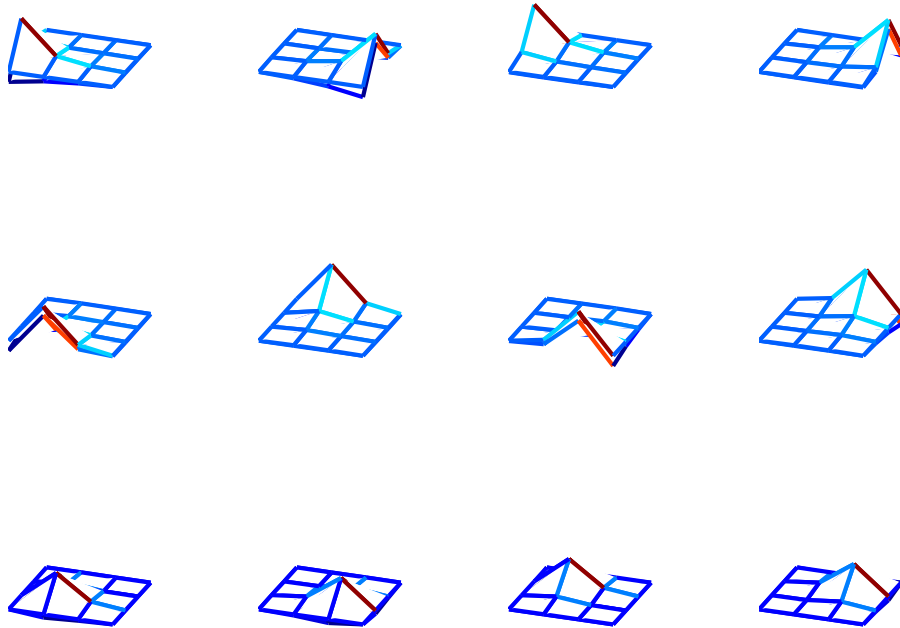


Figure 5: Control nets of the piecewise bilinear upper and lower bounds A_{κ}^{cnv} of degree 3×3 .

for directionally convex polynomials since it interpolates $A_{\kappa} - LA_{\kappa}$ at the Greville abscissa. The * bounds are essentially the estimates derived in [7, 9]. The lower bound can, however, be considerably improved. For example, for the functions in Figure 6 the respective enclosure widths using the * bounds are 1.7, 0.56 and 1.58, while the improved estimates based on the subdivide-and-bound algorithm of Section 5 have widths 0.31, 0.13 and 0.30. The *left* column of Figure 6 shows examples of enclosures based on the improved bounds for Δ^{cnv} .

3.2 Univariate second differences

Once $\lfloor A - LA \rfloor$ and $\lceil A - LA \rceil$ are tabulated the main effort for computing enclosures goes into computing differences of the control net. It is therefore desirable to compute enclosures based on differences with small masks. The cost is minimal if we choose an independent subset of the differences Δ^{uni} : we pick a parallel *grating* (like the teeth of a comb) in the i -direction. For $\kappa(i, j)$ with $0 < i < d_1$, we pick $b_{i-1, j} - 2b_{i, j} + b_{i+1, j}$ and for $\kappa(0, j)$ and $\kappa(d_1, j)$ with $0 < j < d_2$ on the boundary of the control net we pick the univariate second difference in the j -direction.

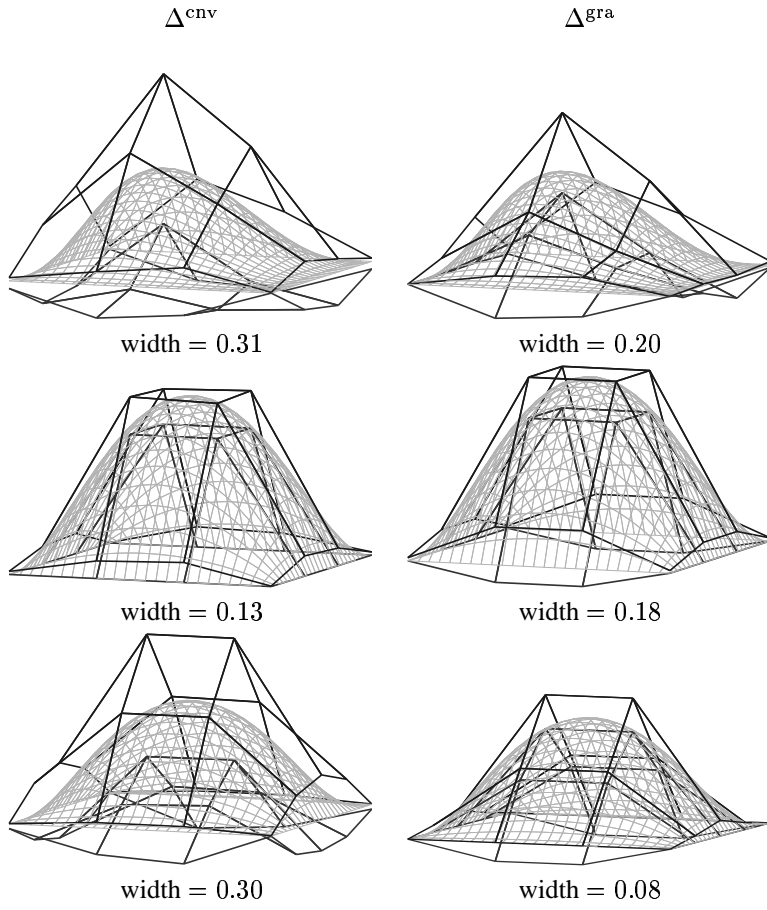


Figure 6: Envelopes of $B_{1,1}^{3,3}$, $B_{1,1}^{3,3} + B_{2,1}^{3,3} + B_{1,2}^{3,3} + B_{2,2}^{3,3}$, and $B_{1,2}^{3,4} + B_{2,2}^{3,4}$ with respect to Δ^{cnv} (left) and enclosures with respect to Δ^{gra} (right). Each of the six figures shows the black control nets of the piecewise bilinear enclosure functions \mathbf{b}^+ and \mathbf{b}^- that sandwich the grey, finely gridded graph of \mathbf{b} .

The Bézier polynomial A_κ corresponding to $\Delta_\kappa^{\text{gra}}$ has the coefficients

$$A_{\kappa,\alpha,\beta} = A_{\kappa(i,j),\alpha,\beta}^{\text{gra}} = \frac{-1}{d_1 d_2} \begin{cases} 0 & 0 < i < d_1, \beta \neq j \\ d_2 r_i^{d_1}(\alpha) & 0 < i < d_1, \beta = j \\ (d_1 - \alpha)r_j^{d_2}(\beta) & i = 0, 0 < j < d_2 \\ \alpha r_j^{d_2}(\beta) & i = d_1, 0 < j < d_2 \end{cases}$$

For example, if $d_1 = 4, d_2 = 3, A_\kappa$ for $\kappa = \kappa(1, 3)$ corresponds to the difference $b_{03} - 2b_{13} + b_{23}$ and has all coefficients zero except $-12A_\kappa(\cdot, 3) = 0, 3, 2, 1, 0$. We check $\Delta^{\text{gra}} A = I$ but the functions $A_\kappa - LA_\kappa$ are not directionally convex and we need to employ the subdivide-and-bound algorithm of Section 5 to compute the vectors of lower and upper bounds $\lfloor A_\kappa - LA_\kappa \rfloor$ and $\lceil A_\kappa - LA_\kappa \rceil$. The *right* column of Figure 6 shows examples of enclosures based on Δ^{gra} .

3.3 Example: enclosure of a Bézier polynomial of degree 2×2

We compute enclosures for $\mathbf{b}(u, v) = B_1^2(u)B_2^2(v)$. The five differences of $\Delta^{\text{cnv}} \mathbf{b}$ are

$$\begin{aligned} b_{00} - 2b_{01} + b_{02} &= 0, & b_{20} - 2b_{21} + b_{22} &= 0, \\ b_{02} - 2b_{12} + b_{22} &= -2, & b_{00} - 2b_{10} + b_{20} &= 0, \\ -b_{00} + 2b_{10} - b_{20} + 2b_{01} - 4b_{11} + 2b_{21} - b_{02} + 2b_{12} - b_{22} &= 2 \end{aligned}$$

and the five differences of $\Delta^{\text{gra}} \mathbf{b}$ are

$$\begin{aligned} b_{00} - 2b_{01} + b_{02} &= 0, & b_{20} - 2b_{21} + b_{22} &= 0, \\ b_{02} - 2b_{12} + b_{22} &= -2, & b_{00} - 2b_{10} + b_{20} &= 0, \\ b_{01} - 2b_{11} + b_{21} &= 0. \end{aligned}$$

The piecewise bilinear upper and lower bounding functions $\lceil A_\kappa - LA_\kappa \rceil, \lfloor A_\kappa - LA_\kappa \rfloor$ are obtained from [3] as tabulated, precomputed 9-tuples of scalars that represent their values at the Greville abscissae. Initializing \mathbf{b}^- and \mathbf{b}^+ to $L\mathbf{b}$,

we add $\Delta_\kappa \mathbf{b} \lceil A_\kappa - LA_\kappa \rceil$ to \mathbf{b}^+ if $\Delta_\kappa \mathbf{b} > 0$ or to \mathbf{b}^- if $\Delta_\kappa \mathbf{b} < 0$; and we add $\Delta_\kappa \mathbf{b} \lfloor A_\kappa - LA_\kappa \rfloor$ to \mathbf{b}^- if $\Delta_\kappa \mathbf{b} > 0$ or to \mathbf{b}^+ if $\Delta_\kappa \mathbf{b} < 0$.

We obtain the control 9-tuples (breakpoints of a piecewise bilinear function consisting of 4 pieces)

$$\begin{array}{l} \mathbf{b}_{\text{cnv}}^+ : \begin{array}{ccc} 0.1407 & 0.6104 & 0.1407 \\ 0.0528 & 0.1664 & 0.0528, \\ 0.0000 & 0.0000 & 0.0000 \end{array} \quad \mathbf{b}_{\text{cnv}}^- : \begin{array}{ccc} -0.0000 & 0.4560 & -0.0000 \\ -0.0257 & 0.0848 & -0.0257, \\ -0.0000 & -0.0440 & -0.0000 \end{array} \\ \mathbf{b}_{\text{gra}}^+ : \begin{array}{ccc} 0.1422 & 0.6089 & 0.1422 \\ 0.0711 & 0.1794 & 0.0711, \\ 0.0000 & 0.0000 & 0.0000 \end{array} \quad \mathbf{b}_{\text{gra}}^- : \begin{array}{ccc} -0.0000 & 0.4577 & -0.0000 \\ -0.0532 & 0.0635 & -0.0532. \\ 0.0000 & -0.0423 & 0 \end{array} \end{array}$$

The width with respect to Δ^{cnv} is 0.1544 and with respect to Δ^{gra} is 0.1512.

4 A general framework for enclosures

This section formalizes the approach of the previous two sections and defines the construction for arbitrary polynomial bases. As before we want to find two functions $\mathbf{b}^+, \mathbf{b}^-$ from \mathcal{H} that tightly enclose a given polynomial $\mathbf{b} \in \mathcal{B} \neq \mathcal{H}$. The initial transformation in Lemma 1 makes the enclosure invariant under addition of elements from $\mathcal{B} \cap \mathcal{H}$. The corollary records the inequalities needed for constructing the enclosure.

Lemma 1 (Change of basis). *Given two spaces of functions, $\mathcal{B} \neq \mathcal{H}$, the embedding identity $E : \mathcal{B} \rightarrow \mathcal{B} + \mathcal{H}$, and linear maps $L : \mathcal{B} \rightarrow \mathcal{H} \subset \mathcal{B} + \mathcal{H}$, $\Delta : \mathcal{B} \rightarrow \mathbb{R}^s$, $A : \mathbb{R}^s \rightarrow \mathcal{B}$, where $s = \dim \mathcal{B} - \dim(\mathcal{B} \cap \mathcal{H})$ such that*

(i) $\Delta A = I$, the identity in \mathbb{R}^s , and

(ii) $\ker \Delta = \ker(E - L)$.

Then for any $\mathbf{b} \in \mathcal{B}$, $(E - L)\mathbf{b} =$

$$\mathbf{b} - L\mathbf{b} = (A - LA)(\Delta\mathbf{b}). \quad (1)$$

Proof. By (i) $\Delta(I - A\Delta)\mathbf{b} = 0$ and hence by (ii) $(E - L)(I - A\Delta)\mathbf{b} = 0$, i.e. (1). \square

Here as earlier $(A - LA)(\Delta\mathbf{b})$ is a short notation for the linear combination $\sum_k (A_{\kappa} - LA_{\kappa})\Delta_k\mathbf{b}$. Spaces \mathcal{B} and \mathcal{H} may be infinite dimensional, e.g. if they represent splines over an infinite knot sequence. However, for practical computation, the linear combination $(A - LA)(\Delta\mathbf{b})$ has to have only finitely many terms, e.g. $s < \infty$. Items (i) and (ii) make $I - A\Delta$ a projector into a space invariant under L . In (ii), $\ker \Delta \subset \ker(E - L)$ is needed since for any $\mathbf{b} \in \ker \Delta \setminus \ker(E - L)$ the right side of Equation (1) is zero, but not the left. Since the width of the enclosure changes under addition of any $k \in \ker(E - L) \setminus \ker \Delta$ we also want $\ker(E - L) \subset \ker \Delta$.

Corollary 2 (Range Estimation). *With \mathcal{B} , \mathcal{H} , s , A and Δ as in Lemma 1, and the maps $\lfloor \cdot \rfloor, \lceil \cdot \rceil : \mathcal{B}^s \rightarrow \mathcal{H}^s$ such that the following inequalities hold pointwise and componentwise: $\lfloor A - LA \rfloor \leq A - LA \leq \lceil A - LA \rceil$. Then*

$$\begin{aligned} \mathbf{b}^- &:= L\mathbf{b} + \lfloor A - LA \rfloor (\Delta\mathbf{b})_+ + \lceil A - LA \rceil (\Delta\mathbf{b})_- \\ &\leq \mathbf{b} \leq \\ \mathbf{b}^+ &:= L\mathbf{b} + \lfloor A - LA \rfloor (\Delta\mathbf{b})_- + \lceil A - LA \rceil (\Delta\mathbf{b})_+. \end{aligned}$$

Together this justifies the following general **enclosure construction**:

- (0) Choose U , the domain of interest, and the space \mathcal{H} of enclosure functions.
- (1) Choose a difference operator $\Delta : \mathcal{B} \rightarrow \mathbb{R}^s$, with $\ker \Delta = \mathcal{B} \cap \mathcal{H}$.
- (2) Compute $A : \mathbb{R}^s \rightarrow \mathcal{B}$ so that ΔA is the identity on \mathbb{R}^s and each A_{κ} matches the same $\dim(\mathcal{B} \cap \mathcal{H})$ additional independent constraints.
- (3) Compute $\lfloor A - LA \rfloor$ and $\lceil A - LA \rceil$.

(4) Compute $(\Delta \mathbf{b})_+$ and $(\Delta \mathbf{b})_-$ and assemble \mathbf{b}^- and \mathbf{b}^+ .

For polynomials, steps (2-4) can be automated as implemented in [3]. The strength of the approach is that any, possibly expensive optimization of the enclosures is encapsulated in step (3) which can be computed, once and for all, for a given pair \mathcal{B} , \mathcal{H} as a preprocessing step. The runtime enclosure computation then consists only of (4).

5 Automatic generation of enclosures for $A - LA$

Since $\lfloor A_\kappa - LA_\kappa \rfloor$ and $\lceil A_\kappa - LA_\kappa \rceil$ are tabulated once and for all we can expend much energy to obtain best bounds. Still it is convenient to have a good and fast algorithm at hand. Off hand, for $\mathbf{h} := A_\kappa - LA_\kappa$, we need to solve a *continuous minimization problem* for all $u \in U$:

$$\begin{aligned} \min_{\lceil \mathbf{h} \rceil, \lfloor \mathbf{h} \rfloor \in \mathcal{H}} \quad & \|\lceil \mathbf{h} \rceil - \lfloor \mathbf{h} \rfloor\|_{\infty, U} \\ & \lfloor \mathbf{h} \rfloor \leq \mathbf{h} \\ & \lceil \mathbf{h} \rceil \geq \mathbf{h}. \end{aligned}$$

The subdivide-and-bound algorithm discretizes the problem and iteratively generates an increasing sequence of lower bounding functions

$$\lfloor \mathbf{h} \rfloor^0 \leq \dots \leq \lfloor \mathbf{h} \rfloor^{n-1} \leq \lfloor \mathbf{h} \rfloor^n \leq \mathbf{h}$$

and a decreasing sequence of upper bounding functions from \mathcal{H}

$$\lceil \mathbf{h} \rceil^0 \geq \dots \geq \lceil \mathbf{h} \rceil^{n-1} \geq \lceil \mathbf{h} \rceil^n \geq \mathbf{h}.$$

The superscript indicates a change (improvement) of the maps $\lfloor \cdot \rfloor, \lceil \cdot \rceil : \mathcal{B} \rightarrow \mathcal{H}$. Matlab code of the following **subdivide-and-bound algorithm** is available from [3].

Pre-Condition For any $\mathbf{h}_1, \mathbf{h}_2 \in \mathcal{H}$, *finitely* many linear functionals $F_\kappa(\mathbf{h}_1) \leq F_\kappa(\mathbf{h}_2)$ decide $\mathbf{h}_1 \leq \mathbf{h}_2$.

(0) *Initialization*: Set the maximal number of iterations n_{max} , $\sigma \in \mathbb{N}$ and pick maps $\lfloor \cdot \rfloor^0$ and $\lceil \cdot \rceil^0$ such that $\lfloor A_\kappa - LA_\kappa \rfloor^0 \leq A_\kappa - LA_\kappa \leq \lceil A_\kappa - LA_\kappa \rceil^0$ for $\kappa = 1, \dots, s$. (For example, for polynomials in Bézier representation, $\lfloor \cdot \rfloor^0$ can map to the constant function equal to the maximal Bézier coefficient as illustrated in Figure 7 under (0).)

For $n = 0, \dots, n_{max}$,

$$\text{For } \kappa = 1, \dots, s, \quad \mathbf{h} := A_\kappa - LA_\kappa$$

(1) *Enclose the refined representation*: Split \mathbf{h} into σ pieces \mathbf{h}_i^σ and enclose each piece

$$\begin{aligned} (\mathbf{h}_i^\sigma)^- &:= L(\mathbf{h}_i^\sigma) + \lfloor A - LA \rfloor^n (\Delta \mathbf{h}_i^\sigma)_+ + \lceil A - LA \rceil^n (\Delta \mathbf{h}_i^\sigma)_- \\ &\leq \mathbf{h}_i^\sigma \leq \\ (\mathbf{h}_i^\sigma)^+ &:= L(\mathbf{h}_i^\sigma) + \lceil A - LA \rceil^n (\Delta \mathbf{h}_i^\sigma)_- + \lfloor A - LA \rfloor^n (\Delta \mathbf{h}_i^\sigma)_+. \end{aligned}$$

(Then the union \mathbf{h}^σ of the pieces \mathbf{h}_i^σ satisfies $\lfloor \mathbf{h}^\sigma \rfloor^n \leq \mathbf{h} \leq \lceil \mathbf{h}^\sigma \rceil^n$ but $\mathbf{h}^\sigma \notin \mathcal{H}$. Note that this step is relative cheap and therefore one can try several different σ .)

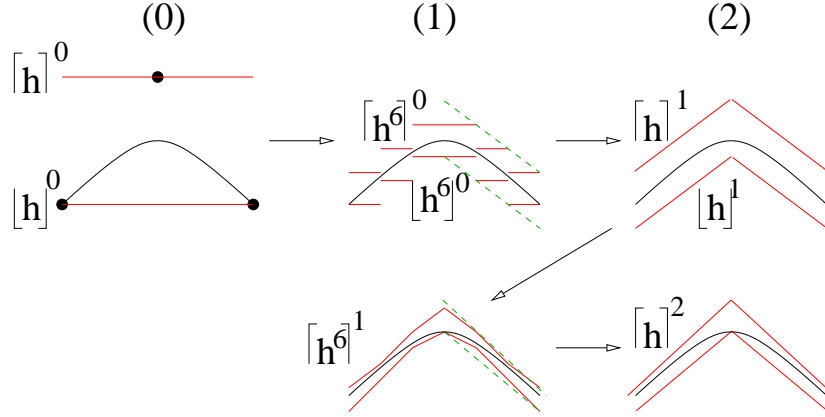


Figure 7: Initialization (0) and the first two iterations of the subdivide-and-bound algorithm for a piecewise linear \mathcal{H} with breakpoints at 0, .5, 1 on $U = [0, 1]$. The new upper and lower bounds (solid lines under (2)) are replicated as dashed lines under (1).

- (2) *Enclose the refined enclosure:* Solve the following linear program for $[\mathbf{h}]^{n+1}$ and $[\mathbf{h}]^{n+1}$. One possible goal function is to minimize

$$(LP) \min \sum_{\kappa} F_{\kappa}([\mathbf{h}]^{n+1}) - F_{\kappa}([\mathbf{h}]^{n+1})$$

$$[\mathbf{h}]^{n+1} \leq [\mathbf{h}^{\sigma}]^n$$

$$[\mathbf{h}]^{n+1} \geq [\mathbf{h}^{\sigma}]^n.$$

(For example, if $[\mathbf{h}]^{n+1}$ and $[\mathbf{h}]^{n+1}$ are piecewise bilinear, $F_{\kappa(ij)}([\mathbf{h}]^{n+1})$ can be the (ij) breakpoint of $[\mathbf{h}]^{n+1}$. Then the continuous inequalities $[\mathbf{h}]^{n+1} \leq [\mathbf{h}^{\sigma}]^n$ and $[\mathbf{h}]^{n+1} \geq [\mathbf{h}^{\sigma}]^n$ reduce to a finite number of inequalities on the breakpoints of \mathbf{h}^{σ} .)

- (3) Break if no improvement, i.e. if
 $[A - LA]^{n+1} - [A - LA]^{n+1} \approx [A - LA]^n - [A - LA]^n$.

Continue loops over κ and n .

(The inequalities $[\mathbf{h}^{\sigma}]^n \leq \mathbf{h} \leq [\mathbf{h}^{\sigma}]^n$ in (2) above are based on the estimator at iteration n . Each iteration improves the estimator.)

If \mathbf{h} is of bounded variation the approximation in step (1) becomes ever closer for larger σ . A closer approximation in step (1) in turn permits a tighter bound in step (2) improving the maps $[\cdot]^n$ or $[\cdot]^n$. The subdivide-and-bound algorithm is a heuristic since we do not prove that it converges towards a global optimum for any fixed σ . However, we observed that for tensor-product polynomials in Bézier form $n_{max} = 3$ and $\sigma = 3$, i.e. three iterations and a 3-fold subdivision, results in tight enclosures.

6 Other multivariate representations

Piecewise bilinear enclosures for *non-uniform b-splines* can be constructed in complete analogy to those for tensor-product Bernstein polynomials: the difference operators Δ^{uni} take the non-equidistant spacing of the Greville abscissae into account [6]. If $\lfloor A - LA \rfloor$ and $\lceil A - LA \rceil$ are not tabulated, the first computation would have to generate the table for repeated use. If the knot-sequence is not re-used it pays to instead convert the pieces to Bézier form and use existing tables.

Enclosures for *Bézier patches of total degree* are described in [4]. *Parametric enclosures* can be based on enclosures of the component functions ([5] and [8]). To make enclosures competitive with other bounding constructs such as [2] the challenge is to generate only as many pieces, say, as there are facets in the control net and no additional slivers.

References

- [1] Carl de Boor. *B-form basics*. In Gerald E. Farin, editor, *Geometric modeling*, pages 131–148. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, Pennsylvania, 1987.
- [2] Stefan Gottschalk, Ming Lin, and Dinesh Manocha. OBBTree: A hierarchical structure for rapid interference detection. In *Proceedings of the ACM Conference on Computer Graphics*, pages 171–180, New York, August 4–9 1996. ACM.
- [3] David Lutterkort. Matlab software for bivariate spline envelopes. tar-file at <http://www.cise.ufl.edu/research/SurfLab/papers>.
- [4] David Lutterkort. *Envelopes for Nonlinear Geometry*. PhD thesis, Purdue University, May 2000.
- [5] David Lutterkort and Jörg Peters. Linear envelopes for uniform B-spline curves. In L.L. Schumaker C. Rabut, editor, *Proceedings of Curves and Surfaces 1999, St. Malo*, pages 239–246, 2000.
- [6] David Lutterkort and Jörg Peters. Tight linear envelopes for splines. *Numerische Mathematik*, 200x. accepted Oct 1999.
- [7] Dean Nairn, Jörg Peters, and David Lutterkort. Sharp, quantitative bounds on the distance between a polynomial piece and its Bézier control polygon. *Computer Aided Geometric Design*, 16(7):613–631, 1999.
- [8] J. Peters. Surface envelopes. Technical Report CISE-TR-003, University of Florida, 2001.
- [9] Ulrich Reif. Best bounds on the approximation of polynomials and splines by their control polygon. *Computer Aided Geometric Design*, 17(6):579–589, July 2000.

- [10] Thomas Sauer. Multivariate Bernstein polynomials and convexity. *Computer Aided Geometric Design*, 8(6):465–478, 1991.