

Approximation and geometric modeling with simplex B-splines associated with irregular triangles

S. Auerbach

Department of Geology, University of Bonn, FRG (presently with BASYS Software, Aachen, FRG)

R.H.J. Gmelig Meyling

Shell Research Laboratory, Rijswijk Z-H, The Netherlands

M. Neamtu

Department of Applied Mathematics, Twente University, Enschede, The Netherlands

H. Schaeben

Department of Geology, University of Bonn, FRG (presently with BASYS Software, Aachen, FRG)

Received October 1989

Revised April 1990

Abstract

Auerbach, S., R.H.J. Gmelig Meyling, M. Neamtu and H. Schaeben. Approximation and geometric modeling with simplex B-splines associated with irregular triangles, *Computer Aided Geometric Design* 8 (1991) 67–87.

Bivariate quadratic simplicial B-splines defined by their corresponding set of knots derived from a (suboptimal) constrained Delaunay triangulation of the domain are employed to obtain a C^1 -smooth surface. The generation of triangle vertices is adjusted to the areal distribution of the data in the domain. We emphasize here that the vertices of the triangles initially define the knots of the B-splines and do generally not coincide with the abscissae of the data. Thus, this approach is well suited to process scattered data.

With each vertex of a given triangle we associate two additional points which give rise to six configurations of five knots defining six linearly independent bivariate quadratic B-splines supported on the convex hull of the corresponding five knots.

If we consider the vertices of the triangulation as threefold knots, the bivariate quadratic B-splines turn into the well known bivariate quadratic Bernstein–Bézier-form polynomials on triangles. Thus we might be led to think of B-splines as of smoothed Bernstein–Bézier polynomials with respect to the entire domain. From the degenerate Bernstein–Bézier situation we deduce rules how to locate the additional points associated with each vertex to establish knot configurations that allow the modeling of discontinuities of the function itself or any of its directional derivatives. We find that four collinear knots out of the set of five defining an individual quadratic B-spline generate a discontinuity in the surface along the line they constitute, and that analogously three collinear knots generate a discontinuity in a first derivative.

Finally, the coefficients of the linear combinations of normalized simplicial B-splines are visualized as geometric control points satisfying the convex hull property.

Thus, bivariate quadratic B-splines associated with irregular triangles provide a great flexibility to approximate and model fast changing or even functions with any given discontinuities from scattered data.

An example for least squares approximation with simplex splines is presented.

Keywords: Scattered data, constrained triangulation, simplicial B-splines, approximation, interpolation, geometric modeling

1. Multivariate simplex splines

The theory of multivariate polynomial, simplicial B-splines utilized here has essentially been developed by [Dahmen & Micchelli '82a, b, '83a, b], and [Höllig '82, '86], and adapted for numerical approximation by [Gmelig Meyling '86], [Grandine '86, '87, '88], [Traas '90].

For any spatial dimension s and polynomial degree k let $n = s + k$. Let $K = \{x^0, \dots, x^n\}$ be a set of points in \mathbb{R}^s such that the convex hull $[x^0, \dots, x^n]$ has dimension s . The standard n -simplex S^n is given by

$$S^n = \left\{ (t_0, \dots, t_n) \mid \sum_{i=0}^n t_i = 1, t_i \geq 0, i = 0, \dots, n \right\}. \quad (1)$$

Using the Hermite–Genocchi formula for divided differences the multivariate B-spline $M(x \mid x^0, \dots, x^n)$, or briefly $M(x \mid K)$, is implicitly defined by the identity

$$\int_{\mathbb{R}^s} f(x) M(x \mid x^0, \dots, x^n) dx_1 \cdots dx_s = n! \int_{S^n} f(t_0 x^0 + \cdots + t_n x^n) dt_1 \cdots dt_n \quad (2)$$

which must hold for all $f \in C(\mathbb{R}^s)$, [Micchelli '79].

There is a geometric interpretation of the s -variate B-spline $M(x \mid K)$. Let $\sigma = [v^0, \dots, v^n]$ be any n -simplex such that

$$\text{vol}_n(\sigma) > 0, \quad (3)$$

$$v^i \mid_{\mathbb{R}^s} = x^i, \quad i = 0, \dots, n. \quad (4)$$

Then Schoenberg's well known volume formula is recovered

$$M(x \mid K) = \text{vol}_k(\{v \in \sigma: v \mid_{\mathbb{R}^s} = x\}) / \text{vol}_n(\sigma), \quad x \in \mathbb{R}^s \quad (5)$$

which holds independently of σ .

Hence, the B-spline $M(x \mid K)$ is nonnegative, and its support is the convex hull $[K]$. Moreover, it is a piecewise polynomial. Choosing an appropriate collection of knot configurations the corresponding set of B-splines forms a basis of a space of splines defined on \mathbb{R}^s , [Dahmen & Micchelli '82b], [Höllig '82].

As in the case of univariate B-splines there are recurrence relations to facilitate the numerical evaluation of multivariate B-splines, first derived in [Micchelli '80]; they read for B-splines if $n > s$,

$$M(x \mid x^0, \dots, x^n) = \frac{n}{n-s} \sum_{i=0}^n \lambda_i M(x \mid x^0, \dots, x^{i-1}, x^{i+1}, \dots, x^n) \quad (6)$$

where $\sum_{i=0}^n \lambda_i = 1$ and $\sum_{i=0}^n \lambda_i x^i = x$,
or if $n = s$,

$$M(x \mid x^0, \dots, x^s) = \begin{cases} 1/\text{vol}_s([x^0, \dots, x^s]) & \text{if } x \in \text{int}([x^0, \dots, x^s]), \\ 0 & \text{if } x \notin [x^0, \dots, x^s] \end{cases} \quad (7)$$

where a particular decision depending on the specific situation is required in eq. (7) for the evaluation on the boundary;

for directional derivatives if $n > s$,

$$D_z M(x \mid x^0, \dots, x^n) = n \sum_{i=0}^n \mu_i M(x \mid x^0, \dots, x^{i-1}, x^{i+1}, \dots, x^n) \quad (8)$$

where $\sum_{i=0}^n \mu_i = 0$, $\sum_{i=0}^n \mu_i x^i = z$, and $z \in \mathbb{R}^s$.

1.1. Bivariate quadratic simplex B-splines associated with irregular triangles

In order to obtain a C^1 -smooth surface we employ piecewise quadratic simplicial B-splines in two variables over $\Omega \subset \mathbb{R}^2$.

By virtue of Schoenberg's volume-projection formula each bivariate quadratic simplicial B-spline may be thought of as being defined by the vertices of a 4-simplex projected onto \mathbb{R}^2 . Now suppose \mathcal{P} is the standard simplex in \mathbb{R}^2 , and let \mathcal{T} be an appropriate partition into 4-simplices of the cylinder set $\Omega \times \mathcal{P} \subset \mathbb{R}^4$. Each of the 4-simplices defines a bivariate quadratic spline by its vertices projected in \mathbb{R}^2 ; thus a set of knot configurations $\mathcal{C}(\mathcal{T})$ can be associated with \mathcal{T} . Hence the triangulation \mathcal{T} of $\Omega \times \mathcal{P} \subset \mathbb{R}^4$ gives rise to the spline space $\mathcal{S}(\mathcal{T}) = \text{span}\{M(x|K): K \in \mathcal{C}(\mathcal{T})\}$. Choosing a canonical triangulation \mathcal{T}^* of $\Omega \times \mathcal{P}$, the set $\{M(x|K): K \in \mathcal{C}(\mathcal{T}^*)\}$ provides a basis of $\mathcal{S}(\mathcal{T}^*)$, cf. [Dahmen & Micchelli '82a].

Reversing the view of projection from $\Omega \times \mathcal{P} \subset \mathbb{R}^4$ in \mathbb{R}^2 , we consider now the domain $\Omega \subset \mathbb{R}^2$ and any suitable triangulation Δ of Ω . Suppose Δ consists of T triangles with V vertices. We denote the vertices of Δ by $x^i, i = 1, \dots, V$ and the triangles by $[x^i, x^j, x^k]$.

With every vertex $x^{i,0} = x^i$ we associate two additional points $x^{i,1}, x^{i,2}$. These additional points have to be defined to satisfy some reasonable constraints, see Section 3.3. They can be thought of as being pulled apart from $x^{i,0}$. The vertices of the triangles and their two associates will become the knots defining the bivariate quadratic B-spline.

With the combinatorial recipe by [Dahmen & Micchelli '82a] we choose six configurations of five knots each for each triangle $t \in \Delta$ corresponding to the canonical triangulation \mathcal{T}^* of $t \times \mathcal{P}$, and thus obtain six linearly independent bivariate quadratic B-splines for every triangle $t \in \Delta$. Now let us study an arbitrary individual triangle $t = [x^i, x^j, x^k], i < j < k$, see Fig. 1. The six B-splines belonging to this triangle are respectively: $M(x|K_{t,m}), m = 1, \dots, 6, x = (x_1, x_2) \in \mathbb{R}^2$, with

$$\begin{aligned}
 K_{t,1} &= \{x^{i,0}, x^{j,0}, x^{k,0}, x^{k,1}, x^{k,2}\}, \\
 K_{t,2} &= \{x^{i,0}, x^{j,0}, x^{j,1}, x^{k,1}, x^{k,2}\}, \\
 K_{t,3} &= \{x^{i,0}, x^{j,0}, x^{j,1}, x^{j,2}, x^{k,2}\}, \\
 K_{t,4} &= \{x^{i,0}, x^{i,1}, x^{j,1}, x^{k,1}, x^{k,2}\}, \\
 K_{t,5} &= \{x^{i,0}, x^{i,1}, x^{j,1}, x^{j,2}, x^{k,2}\}, \\
 K_{t,6} &= \{x^{i,0}, x^{i,1}, x^{i,2}, x^{j,2}, x^{k,2}\}.
 \end{aligned}
 \tag{9}$$

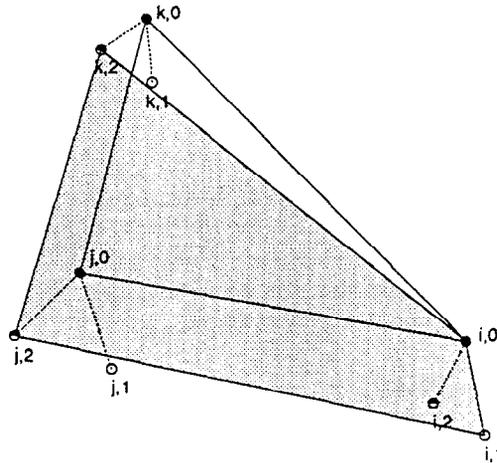


Fig. 1. Triangle $[x^i, x^j, x^k] = t \in \Delta$, with vertices x^i, x^j, x^k labeled such that $i < j < k$, and associated points $x^{i,1}, x^{j,1}, x^{k,1}, x^{i,2}, x^{j,2}, x^{k,2}$ generating knot sets $K_{t,m}, m = 1, \dots, 6$, of which $K_{t,6}$ is shown shaded.

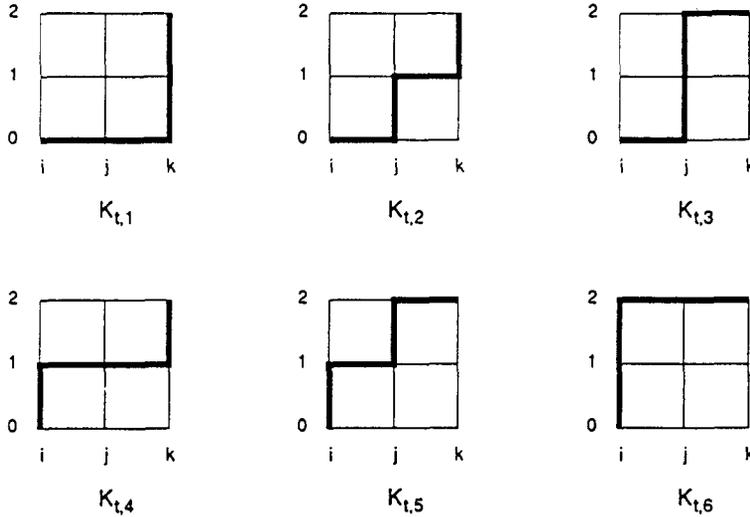


Fig. 2. Nondescending paths p_r along grid lines from $(0, 0)$ to $(2, 2)$ in the lattice of labels i, j, k of knots and labels $q \in \{0, 1, 2\}$ of faces describing the indices of the knot sets $K_{t,m}$, $m = 1, \dots, 6$.

These six knot configurations, i.e. their indices, can be represented as nondescending paths p_r , $r = 1, \dots, 6$, along grid lines from $(0, 0)$ to $(2, 2)$ in the lattice of (local) knot numbers $i \in \{0, 1, 2\}$ and their second superscripts $q \in \{0, 1, 2\}$, as depicted in Fig. 2, cf. [Dahmen & Micchelli '82a], [Höllig '82].

Summarizing the properties of a B-spline $M(x | K_{t,m})$, it is

1. supported on the convex hull of its defining five knots,
2. nonnegative on its support,
3. piecewise quadratic with respect to the 'cut-regions' defined as sets in the support set $[K]$ which are bounded but not intersected by any 1-simplex spanned by 2 points from $\{x^0, \dots, x^4\}$, i.e. by the line segments joining the elements of K , see Fig. 3,
4. C^1 -smooth provided that the five knots are in general position, i.e. every three knots span a nondegenerate triangle.

The entire triangular mesh Δ produces $6T$ B-splines which are linearly independent and span all quadratics over Ω ; for the proof the reader is referred to [Dahmen & Micchelli '82b].

We can now define a surface over Ω by the bivariate function

$$s(x) = \sum_{t=1}^T \sum_{m=1}^6 c_{t,m} M(x | K_{t,m}) \tag{10}$$

which is a linear combination of the splines $M(x | K_{t,m})$ with coefficients $c_{t,m}$.

In the following, our aim is a solution to the problem of finding a suitable way to locally control and manipulate the shape of the surface given by eq. (10).

To this end, we must first assure that the basisfunctions $M(x | K_{t,m})$ are normalized such that they form a partition of unity for all $x \in \Omega$, i.e.

$$\sum_{t=1}^T \sum_{m=1}^6 \tilde{M}(x | K_{t,m}) = 1 \tag{11}$$

for all $x \in \Omega$, with

$$\tilde{M}(x | K_{t,m}) = d_{t,m} M(x | K_{t,m}). \tag{12}$$

There is a simple numerical way to compute the coefficients $d_{t,m}$ by solving an appropriate linear system of the form $Md = 1$. An efficient way is to exploit existence and properties of a subregion $B_t \in t$ where only the six B-splines associated with t are different from zero. If the pulling apart of the knots is done in a proper way, there exists a nonempty subregion $B_t \subset t$ inside every triangle t (see Fig. 4) defined by

$$B_t = \bigcap_{(q_0, q_1, q_2) \in Q} [x^{j_0, q_0}, x^{j_1, q_1}, x^{j_2, q_2}] \tag{13}$$

for each triangle t , $t = [x^{j_0}, x^{j_1}, x^{j_2}]$, $j_0 < j_1 < j_2$, and $Q = \{(q_0, q_1, q_2) \in \mathbb{Z}^3 \mid 0 \leq q_0 \leq q_1 \leq q_2 \leq 2\}$, see [Dahmen & Micchelli '82a], [Gmelig Meyling '86]. The set $B_t \subset t$ has the following two major properties:

1. For all $m = 1, \dots, 6$ the spline $M(x | K_{t,m})$ is a polynomial on B_t .
2. In every point $x \in B_t$ only the six B-splines $M(x | K_{t,m})$, $m = 1, \dots, 6$, induced by t are incident.

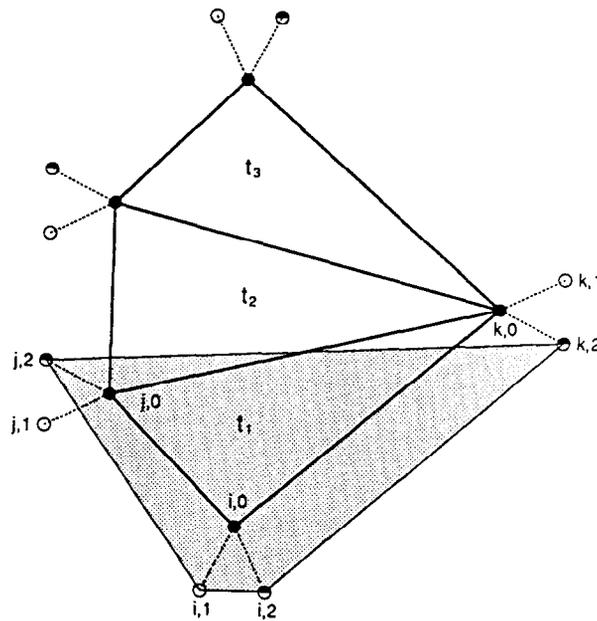


Fig. 3a. Triangulated vertices x^i and their two associated points $x^{i,q}$, $q = 1, 2$. Shaded region: Support $K_{t_1,6}$ of one individual B-spline associated with triangle t_1 .

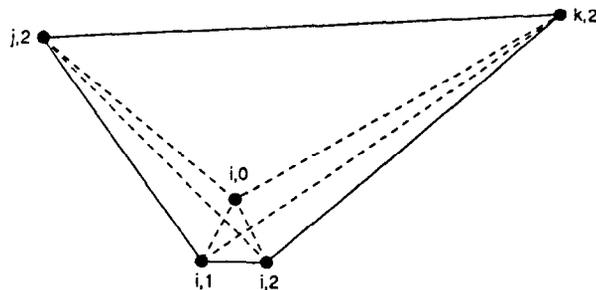


Fig. 3b. 'Cut-regions' of the support $K_{t_1,6}$ with respect to which its bivariate quadratic, simplicial B-spline is a piecewise quadratic polynomial.

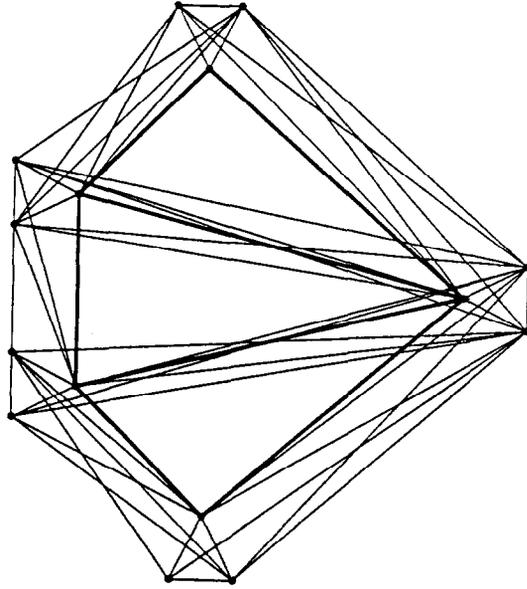


Fig. 3c. 'Cut-regions' of the set $\Omega = t_1 \cup t_2 \cup t_3$.

These imply that it is possible to compute the coefficients $d_{t,m}$ for each individual triangle consecutively. Suppose we take six different points $y^{t,k}$, $k = 1, \dots, 6$, in B_t which do not lie on the same conic section. Then obviously

$$\sum_{m=1}^6 \tilde{M}(x | K_{t,m}) = \sum_{m=1}^6 d_{t,m} M(x | K_{t,m}) = 1 \tag{14}$$

holds for all $x \in B_t$, if

$$\sum_{m=1}^6 d_{t,m} M(y^{t,k} | K_{t,m}) = 1 \tag{15}$$

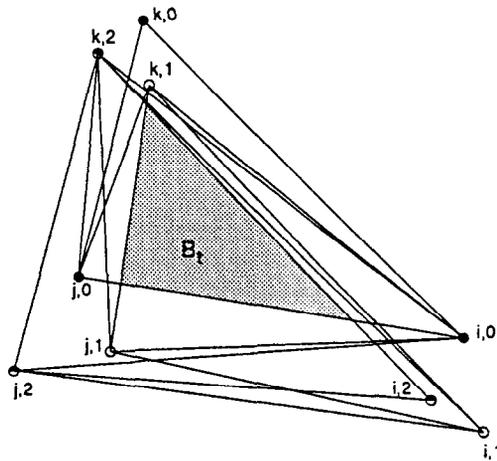


Fig. 4. Subset $B_t \subset t \in \Delta$ where only the six B-splines associated with t are different from zero defined by the intersection of the ten triangles $[x^{i,q_0}, x^{j,q_1}, x^{k,q_2}]$ associated with triangle $t = [x^i, x^j, x^k]$, with $i < j < k$, via $Q = \{(q_0, q_1, q_2) \in \mathbf{Z}^3 | 0 \leq q_0 \leq q_1 \leq q_2 \leq 2\}$.

for all $k = 1, \dots, 6$. Now we solve for the coefficients $d_{t,m}$ from the linear system

$$Md = \mathbf{1} \quad (16)$$

with matrix elements $m_{i,j} = M(\mathbf{y}^{t,i} | K_{t,j})$, $i = 1, \dots, 6$, $j = 1, \dots, 6$, and the column vector \mathbf{d} of unknowns $d_{t,i}$, $i = 1, \dots, 6$, and for every triangle $t \in \Delta$. Eq. (14) may be read as the first of Marsden's identities, see [Dahmen '79], [Goodman & Lee '81].

The condition not to lie on the same conic section ensures that the coefficient matrix of this linear system is nonsingular. With proper row and column scaling, this approach provides a reliable numerical method to compute the coefficients $d_{t,m}$. Doing so for every triangle $t \in \Delta$, we automatically get

$$\sum_{t=1}^T \sum_{m=1}^6 \tilde{M}(\mathbf{x} | K_{t,m}) = 1 \quad (17)$$

for all $\mathbf{x} \in \Omega$.

Now we turn to the problem of defining control points for manipulating the surface.

[Goodman & Lee '81] have given an analytical solution to this problem. If we address the second upper index of a knot $\mathbf{x}^{i,l}$, $i = 1, \dots, V$, $l = 0, 1, 2$, as label of the face the knot belongs to, then Goodman and Lee observed that two different cases can occur: either three of the five knots defining one of the six B-splines associated with the triangle t are in one face, or there are two different faces with two knots each. In the first case the abscissa of the corresponding control point is given by the mean of the remaining two knots, in the second case it is given as the mean of the remaining knot and the point of intersection of the two lines defined by the two knots of the same face, [Goodman & Lee '81].

Here, we suggest a numerical alternative which proved itself efficient in practical applications. Suppose we take again the six different points $\mathbf{y}^{t,k}$, $\mathbf{y}^{t,k} = (y_1^{t,k}, y_2^{t,k})$, $k = 1, \dots, 6$, in B_t which do not lie on the same conic section. Then we may set

$$\sum_{m=1}^6 x_{t,m} \tilde{M}(\mathbf{y}^{t,k} | K_{t,m}) = y_1^{t,k} \quad (18)$$

and

$$\sum_{m=1}^6 y_{t,m} \tilde{M}(\mathbf{y}^{t,k} | K_{t,m}) = y_2^{t,k} \quad (19)$$

for all $k = 1, \dots, 6$ and for every triangle $t \in \Delta$. Now we solve for the coefficients $x_{t,m}$, resp. $y_{t,m}$, from the two linear systems

$$Md = \mathbf{r} \quad (20)$$

with essentially the same matrix M as in eq. (16), column vectors \mathbf{d} of unknowns $d_{t,i} = x_{t,i}$, resp. $d_{t,i} = y_{t,i}$, $i = 1, \dots, 6$, and right-hand side column vectors \mathbf{r} with $r_{t,i} = y_1^{t,i}$, resp. $r_{t,i} = y_2^{t,i}$, $i = 1, \dots, 6$, and for every triangle $t \in \Delta$. Eq. (18), resp. eq. (19), may be read as application of the second of Marsden's identities, see [Dahmen '79], [Goodman & Lee '81].

The coefficients $x_{t,i}$, resp. $y_{t,i}$, of the linear systems (18), resp. (19), provide the abscissae of the control points $\mathbf{b}^{t,i}$ corresponding to the coefficients $b_{t,i} = (x_{t,i}, y_{t,i}, b_{t,i})$ with $b_{t,i}$ being the coefficient of the normalized B-splines $\tilde{M}(\mathbf{x} | K_{t,i})$.

We can now rewrite more appropriately

$$s(\mathbf{x}) = \sum_{t=1}^T \sum_{m=1}^6 b_{t,m} \tilde{M}(\mathbf{x} | K_{t,m}) \quad (21)$$

respectively

$$(\mathbf{x}, s(\mathbf{x})) = \sum_{t=1}^T \sum_{m=1}^6 \mathbf{b}^{t,m} \tilde{M}(\mathbf{x} | K_{t,m}) \quad (22)$$

where eq. (22) complementarily to eq. (10) may be read as convex combination of control points $\mathbf{b}^{t,m} \in \mathbb{R}^3$ weighted by $\tilde{M}(\mathbf{x} | K_{t,m})$.

There are now five important properties of the normalized B-splines $\tilde{M}(\mathbf{x} | K_{t,m})$ which allow to manipulate the surface $(\mathbf{x}, s(\mathbf{x}))$ by its corresponding control points $\mathbf{b}^{t,m}$, resp. their z -coordinates $b_{t,m}$; these are:

1. *Invariance of abscissae*

The points in \mathbb{R}^3 defined by the map from \mathbb{R}^2 into \mathbb{R}^3 given by the right-hand side of eq. (22) have the same x_1, x_2 -coordinates as the original points in \mathbb{R}^2 .

2. *Translation invariance*

$$(\mathbf{x}, \bar{s}(\mathbf{x})) = \sum_{t=1}^T \sum_{m=1}^6 (\mathbf{b}^{t,m} + \delta) \tilde{M}(\mathbf{x} | K_{t,m}) = (\mathbf{x}, s(\mathbf{x})) + \delta. \quad (23)$$

3. *Convex hull property*

$$(\mathbf{x}, s(\mathbf{x})) \in \left\{ \{ \mathbf{b}^{t,m} \mid \mathbf{x} \in [K_{t,m}] \} \right\} \quad (24)$$

for all $\mathbf{x} \in \Omega$. However, in general

$$\{ (\mathbf{x}, s(\mathbf{x})) \mid \mathbf{x} \in t \} \not\subset \{ \mathbf{b}^{t,m} \mid m = 1, \dots, 6 \}. \quad (25)$$

4. *Reproduction of linear functions*

If the control points $\mathbf{b}^{t,m}$ are in a given plane, then $(\mathbf{x}, s(\mathbf{x}))$ represents this plane exactly.

5. *C^1 -continuity inside Ω*

If all knots of each configuration are in general position, then for any choice of control points one obtains a C^1 -continuous surface over entire Ω . This is in contrast to Bernstein–Bézier representations.

Since C^1 -continuity over Ω is guaranteed only if all knots of each configuration are in general position, i.e. generate nondegenerate triangles, particular knot configurations are used to intentionally generate discontinuities in the function $s(\mathbf{x})$ or any of its directional derivatives. Thinking of the simplicial B-splines as of smoothed versions of Bernstein–Bézier forms over Ω we find the following configurations particularly useful. A knot configuration with four collinear knots will generate a discontinuity of the corresponding spline across the line given by these four knots, and a configuration with three collinear knots will generate a discontinuity in any directional derivative across the line given by these three knots. A threefold knot will give rise to a configuration with two sets of four collinear knots, see Figs. 5. Taking all vertices \mathbf{x}^i , $i = 1, \dots, V$, of the triangulation Δ as threefold knots, the bivariate quadratic, simplicial

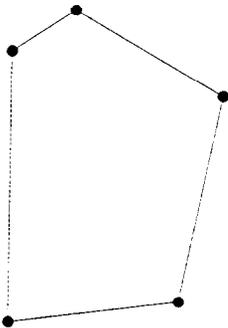


Fig. 5a. Set of knots in general positions and its corresponding individual spline.

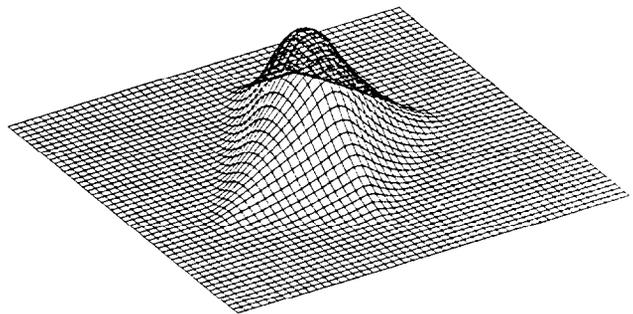


Fig. 5b. Perspective view of the spline defined by the knot configuration of Fig. 5a.

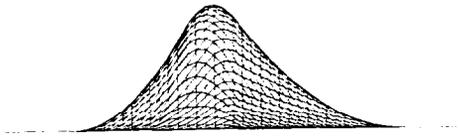


Fig. 5c. Side view of the spline defined by the knot configuration of Fig. 5a.

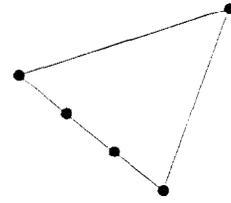


Fig. 5d. Knot configuration with four collinear knots generating a discontinuity in the spline over the line of collinearity.

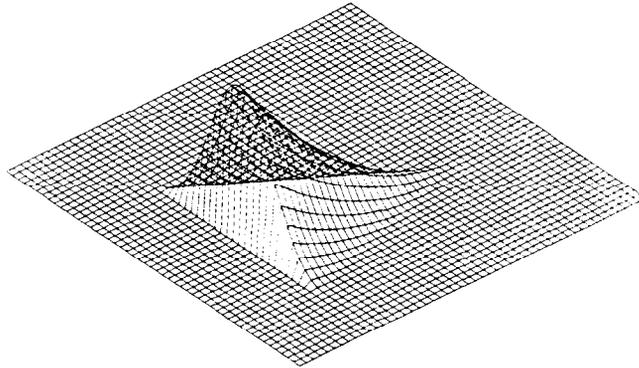


Fig. 5e. Perspective view of the spline defined by the knot configuration of Fig. 5d.

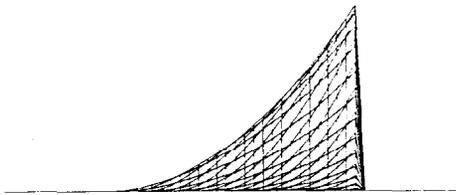


Fig. 5f. Side view of the spline defined by the knot configuration of Fig. 5d.

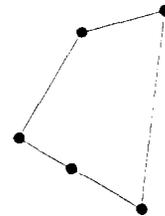


Fig. 5g. Knot configuration with three collinear knots generating a discontinuity in the directional derivatives of the spline across the line of collinearity.

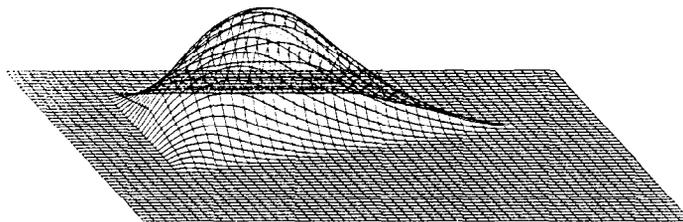


Fig. 5h. Perspective view of the spline defined by the knot configuration of Fig. 5d.

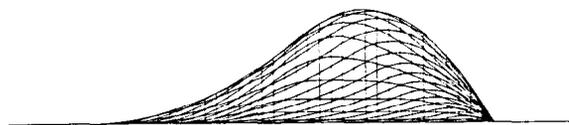


Fig. 5i. Side view of the spline defined by the knot configuration of Fig. 5d.

B-splines degenerate to the bivariate quadratic Bernstein–Bézier forms over the triangulation Δ of the data domain Ω , cf. [Boehm et al. '84], [Dahmen '86], [Farin '86]. The desirable ability to take discontinuities of the function to be approximated or modeled or any of its directional derivatives into account, therefore requires that the lines over which the discontinuities should occur become edges of some triangles in Δ . Thus, the problem encountered here is to find a triangulation appropriate for the purposes of approximation, once the vertices and some predefined edges are given. We termed its solution the (suboptimal) constrained Delaunay triangulation, to be presented in the next section.

2. Definition of knots for practical applications of bivariate quadratic B-splines

The knots, or more specifically the set of knot configurations associated with any triangle, are defined in three basic steps:

1. Define vertices of triangles according to
 - varying density of data, resp. information,
 - known positions of discontinuities of the surface or any of its directional derivatives (predefined edges).
2. Define edges to compose a Delaunay triangulation of the vertices (Dirichlet-, Thiessen tessellation, Voronoi polygon) subject to constraints, e.g. predefined edges, resulting in (suboptimal) constrained Delaunay triangulation of the vertices.
3. Define knots of splines by pulling apart threefold vertices of the triangulation subject to constraints concerning the smoothness of the surface to be approximated or modeled.

2.1. Generation of vertices

The vertices are generated primarily according to the varying density of data, and the available a priori information concerning the surface to be approximated or modeled. Their convex hull will usually define the domain of the surface. The vertices should be placed to generally avoid triangles with less than six data of position or gradient, as a rule of thumb there should be more than twelve data per triangle.

Subsets of points of the domain where discontinuities of the function itself or its directional derivatives occur must become edges in the next step of triangulation. Therefore, vertices must be defined such that their corresponding polygon describes these subsets, resp. edges. Placing the vertices one should always remember that their edges to be defined in the next step will cause data to belong to different triangles.

The generation of vertices is actually the starting point of the geometrical modeling process; it is also the first instance when the user can introduce ‘a priori’ knowledge about the surface to be approximated and modeled. Therefore, we recommend that the vertices are defined interactively with a graphics screen whenever their number is sufficiently small to be reasonably handled in this fashion. For large numbers of vertices to be generated procedures developed for finite element applications, e.g. [Cavendish '74], can be adopted.

2.2. Triangulation of the data domain

Each scheme of interpolation/approximation is conceptually constituted of a neighborhood relation between the location, i.e. the abscissae, of the data, and a rule to assign weights to neighbors. The first step generally addresses the problem which data should be considered for an actual calculation of the interpolant/approximant at a specified point of the domain, the

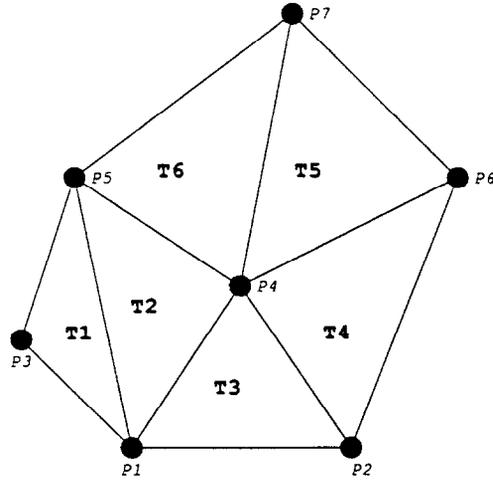


Fig. 6. 'Max-min'-optimal or Delaunay triangulation Δ^* of given vertices P_1, \dots, P_7 .

second step defines the rule how to blend the information inherent in the data; both have to be adequately and properly defined in mathematical terms.

In schemes based on triangulation the neighborhood is obviously defined by the triangles. Usually, the triangulation of some given set of points in the plane is done automatically according to an optimizing criterion, e.g. to Lawson's 'max-min' criterion [Lawson '77], [Cline & Renka '84] to maximize the smallest angle in each triangle. This criterion is aimed at a triangulation as uniform as possible given the vertices, thus basically avoiding degenerated, i.e. elongated, triangles. It is mathematically equivalent to Delaunay triangulation also coined Dirichlet, Thiessen or Voronoi tessellation [Schumaker '87], see Fig. 6.

Let \mathcal{P} be a set of $V \geq 3$ distinct points $P_i = x^i \in \mathbb{R}^2$, $i = 1, \dots, V$. Its convex hull $[\mathcal{P}]$ is defined to be the set

$$[\mathcal{P}] = \left\{ P \in \mathbb{R}^2 \mid P = \sum_{i=1}^V \omega_i P_i, \omega_i \geq 0, i = 1, \dots, V, \sum_{i=1}^V \omega_i = 1 \right\}. \quad (26)$$

Here, it should be sufficient to define a triangulation Δ of \mathcal{P} , resp. $[\mathcal{P}]$, as a partition of $[\mathcal{P}]$ into disjoint triangles t_n , $n = 1, \dots, T$, (except for their common vertices and edges) with vertices P_1, \dots, P_V . A more formal mathematical definition of a triangulation is omitted here because it is both notationally cumbersome and not more instructive. However, the interested reader is referred to [Schumaker '87].

An early and widely used procedure related to applications of the finite element method to generate vertices within a given domain and then triangulate them was given by [Cavendish '74]; a more general and quite comprehensive discussion of this matter can be found in [Haber et al. '81].

Given a set \mathcal{P} , its triangulation is not unique. For reasons of mathematical behaviour and numerical performance of approximation/interpolation schemes degenerated, i.e. elongated, triangles should be avoided.

2.2.1. Optimal Delaunay triangulation

An appropriate triangulation is generally chosen to satisfy some optimality criterion which in turn guarantees first of all a unique triangulation, possibly without elongated triangles. This triangulation is then the best one with respect to the given optimality criterion, and therefore called globally optimal.

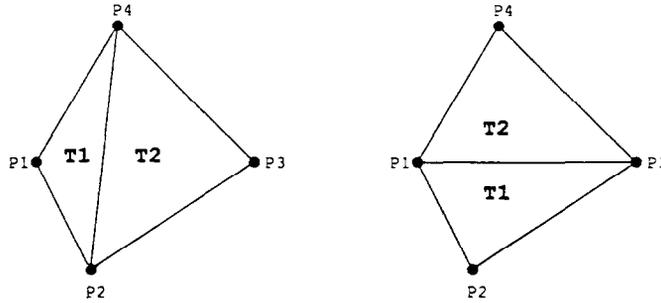


Fig. 7. $[P_{i_1}, \dots, P_{i_4}]$ showing the locally 'max min'-optimal triangulation on the right-hand side.

One special way to achieve a unique triangulation as uniform as possible, is to apply Lawson's 'max-min' criterion to choose the triangulation Δ^* which maximizes the smallest (interior) angle of all triangles [Lawson '77]. This triangulation Δ^* is then globally optimal with respect to the 'max-min' optimality criterion and at the same time it is uniquely defined except for neutral swaps.

A closed quadrilateral polygon Q defined by P_{i_1}, \dots, P_{i_4} is called a strictly convex quadrangle if $[Q \setminus P_{i_\nu}] \neq [Q]$ for all $\nu = 1, \dots, 4$. A triangulation of a strictly convex quadrangle is called locally optimal with respect to the 'max-min' criterion if the smallest of the six interior angles involved is maximized. A triangulation Δ of the domain $[\mathcal{P}]$ is called locally optimal with respect to the 'max-min' criterion, if every strictly convex quadrangle involved in the triangulation is optimally triangulated with respect to this criterion, see Fig. 7. For some arbitrary criterion of optimality a locally optimal triangulation is generally not unique, cf. [Schumaker '87].

The major advantage of applying the global 'max-min' optimality criterion to achieve an optimal triangulation over other triangulation procedures which may be optimal with respect to some other criterion is of algorithmic nature; more specifically, [Lawson '72, '77] and [Sibson '78] have shown that local optimization of a given triangulation with respect to the 'max-min' criterion eventually results in the unique globally 'max-min' optimal triangulation. The proof exploits the equivalence of Delaunay triangulation and Thiessen tessellation. This proposition can then be used to design an efficient algorithm to generate the 'max-min' optimal triangulation Δ^* of $[\mathcal{P}]$. The triangulation Δ^* can actually be constructed by consecutive construction of locally optimal (with respect to the 'max-min' criterion) triangulations of corresponding strictly convex quadrangles, starting with an appropriate $[P_{i_1}, \dots, P_{i_4}]$ and then adding P_{i_ν} , $\nu = 5, \dots, V$, successively. This successive procedure outperforms other alternatives as e.g. the one suggested by [Magnus et al. '83], and allows additional specifications to be incorporated in an easy way.

2.2.2. Constrained Delaunay triangulation

Our aim here is to provide the user who has got some a priori knowledge concerning the neighborhood relation with special means to incorporate it in a triangulation procedure because it would generally be lost otherwise. The basic idea is to adapt the triangulation procedure to honour natural neighborhood relations; thus it resembles features of a concept put forward by [Sibson '81].

This a priori knowledge can basically be expressed by marking given vertices that must not be joined to form an edge of any triangle, e.g. in order to avoid interpolation across the known location of a discontinuity of the function to be interpolated/approximated, or that must be joined to form an edge of some triangle, e.g. in order to preserve the natural neighborhood relation of the data with successive locations along a given line segment. In both cases, the a

priori knowledge concerning the neighborhood relation of data locations can actually be incorporated effectively by defining some edges previously to an automatic procedure which must then honour these predefined edges. We may in fact use an updated version of Lawson's triangulation generalized to preserve predefined edges. This new procedure is then a constrained triangulation which generates triangles which are as uniform as possible given the vertices and some predefined edges. We termed this new procedure (suboptimal) 'constrained Delaunay triangulation'.

If a triangulation Δ is to preserve a predefined edge $[P_{i_\nu}, P_{i_\mu}]$, $i_\nu, i_\mu = 1, \dots, V$, $i_\nu \neq i_\mu$, it is generally not globally optimal. However, given the edge $[P_{i_\nu}, P_{i_\mu}]$, a triangulation $\tilde{\Delta}^*$ can be constructed which is locally optimal for all strictly convex quadrangles involved. This uniquely defined triangulation $\tilde{\Delta}^*$ differs from the Delaunay triangulation Δ^* only locally; more specifically, it differs only in those N_C edges of Δ^* which are intersected by the predefined edge $[P_{i_\nu}, P_{i_\mu}]$ of $\tilde{\Delta}^*$.

The proof of this proposition uses again the properties of the Thiessen tessellation of $[\mathcal{P}]$ and its equivalence to the Delaunay triangulation Δ^* . It is omitted here because the proposition should be visually and intuitively clear (see Figs. 8); it can be found in [Auerbach '90].

Therefore, we termed the triangulation $\tilde{\Delta}^*$ the (suboptimal) constrained Delaunay triangulation of $[P]$ given the edge $[P_{i_\nu}, P_{i_\mu}]$. The same argument can be generalized to several edges which may also be connected with one another to form an open or closed polygon. Construction of the triangulation $\tilde{\Delta}^*$ requires modifications of the algorithm to construct the triangulation Δ^* , see [Auerbach '90] for two different variants.

Another algorithmic aspect to it is that to incorporate an additional edge subsequently into a given (sub)optimal triangulation requires its local updating only.

First applications to deriving digital elevation (terrain) models from digitized contour lines have been given in [Auerbach & Schaeben '90].

2.3. Generation of knots

For arbitrary triangulations Δ there are no canonical positions for the points $x^{i,1}$ and $x^{i,2}$, $i = 1, \dots, V$, which ensure B-splines of optimal smoothness. However, the rules for pulling the knots apart should consider that:

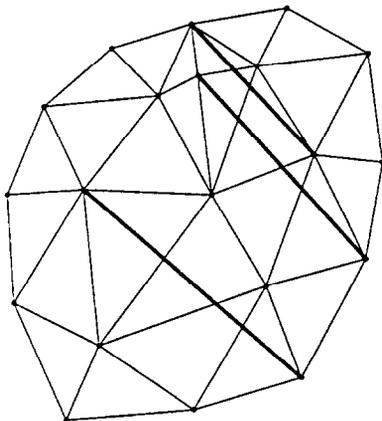


Fig. 8a. 'Max-min' optimal Delaunay triangulation Δ^* of given vertices, but not honoring predefined edges (bold).

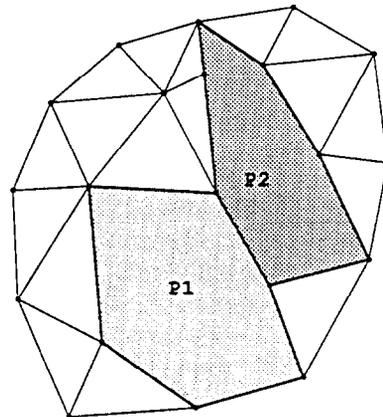


Fig. 8b. Regions P_1 , P_2 , where the triangulation Δ^* has to be changed in order to honor the predefined edges of Fig. 8a.

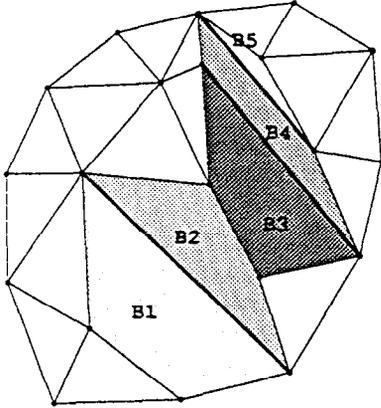


Fig. 8c. Locally 'max-min' optimal triangulation required for regions B_1, \dots, B_5 with $B_1 \cup B_2 = P_1$, $B_3 \cup B_4 \cup B_5 = P_2$.

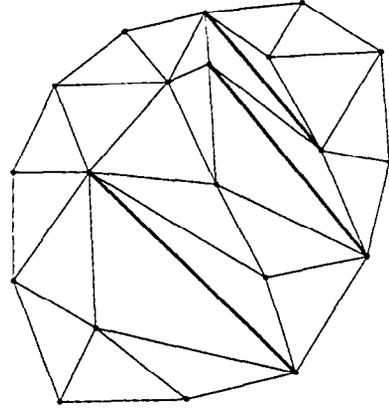


Fig. 8d. Constrained locally 'max-min' optimal triangulation $\tilde{\Delta}^*$ of given vertices and some given edges, or (suboptimal) constrained Delaunay triangulation which now honors predefined edges but is therefore no longer globally 'max-min' optimal.

• Numerical stability of the B-spline basis is preserved if the distances between knots $x^{i,q}$, $q = 1, 2$, and $x^{i,0}$ are kept restricted. To see this, we recall that a B-spline basis is stable if for all coefficients $b_{i,m}$

$$d \|\{b_{i,m}\}\|_{l_p} \leq \left\| \sum_{t=1}^T \sum_{m=1}^6 b_{i,m} \tilde{M}(x | K_{t,m}) \right\|_{L_p(\Omega)} \leq \|\{b_{i,m}\}\|_{l_p} \quad (27)$$

for all $x \in \Omega$ where $d > 0$. This is satisfied here if

$$\text{vol}_2(B_t) > c > 0 \quad (28)$$

for each triangle t , and a given constant $c > 0$, cf. [Dahmen & Micchelli '81, '82b].

• To facilitate the evaluation of linear combinations of B-splines at any given point in $\Omega \in \mathbb{R}^2$ the number of B-spline incidences should be kept small by using triangulations Δ as regular as possible avoiding situations where many triangles share the same vertex, and referring to a systematic numbering of knots. The number of incidences essentially depends on the numbering of vertices. For an appropriate numbering, like e.g. 'line numbering', we can introduce a classification of triangles with respect to each of their vertices. The triangle $[x^i, x^j, x^k]$ is

- of type T0 with respect to x^i if $i < j, k$,
- of type T1 with respect to x^i if $j < i < k$ or $k < i < j$,
- of type T2 with respect to x^i if $j, k < i$.

Numerical investigations by [Gmelig Meyling '86] have shown that the number of incidences is reduced if the knots $x^{i,1}, x^{i,2}$ are placed in triangles of type T2 with respect to x^i . In case of line numbering in an almost regular triangulation this may be interpreted as pulling the knots downwards with respect to the lines, see Fig. 11.

• To achieve optimal smoothness it is required that all knots from all knot sets in \mathcal{C} are in general position, i.e. each three knots of the same knot set must generate a nondegenerate triangle, and to achieve good approximation rates for derivatives nearly coalescent or nearly collinear points in the same knot set must be avoided.

In the following we give an heuristic univariate Monte Carlo algorithm to compute the actual locations of the knots $x^{i,1}$, $x^{i,2}$ associated with x^i which gives satisfying results concerning the criteria mentioned above. It may be understood as compromise between a fully interactive user controlled procedure as suggested by [Dahmen & Micchelli '82a], and a completely automatic procedure solving a sequence of optimization problems as realized by [Gmelig Meyling '86].

1. line numbering of given vertices x^i , $i = 1, \dots, V$;
2. find all triangles with x^i as one of its vertices;
3. find all triangles of type T2 with respect to x^i ;
4. for C^1 -smoothness, determine all 'forbidden lines' [Traas '90], i.e. lines of possible, however unintended collinearity of knots, passing through x^i and intersecting the union of T2-triangles, and their enclosed angles;
5. choose the two lines bisecting the largest and second largest of these angles as directions PD1 and PD2 of pulling apart;
6. to account for a given discontinuity over, resp. across an edge through the vertex x^i , choose this edge as PD1 and PD2;
7. define distances D1 and D2 along PD1, resp. PD2, as approximately one third of the length of the edge closest to PD1, resp. PD2;
8. place $x^{i,1}$, $x^{i,2}$ on the line PD1, resp. PD2 through x^i with distance D1, resp. D2 from x^i ;
9. check if all of the above criteria are satisfied; if so, proceed to next vertex, otherwise systematically reduce distances D1, D2 alternately until the above criteria are satisfied. Should this not be achieved in a few iterative steps the triangulation seems not to be appropriate and requires checking.

More sophisticated procedures following the general rules given above did not prove worth the additional effort in computing demands with respect to the finally computed surfaces.

To allow surfaces not vanishing on the boundary of Ω or with nonvanishing first-order directional derivatives, it is necessary to guarantee that the 'partition of unity' property holds in the entire set Ω . This can be easily obtained by moving the knots $x^{i,1}$, $x^{i,2}$ for which $x^{i,0}$ lies on the boundary outside of Ω , see Fig. 11.

3. Approximation and modeling

For purposes of practical approximation and modeling the general procedure is as follows:

- (suboptimal) constrained triangulation of data domain;
- provide six particular points for each triangle to
 - normalize the coresponding six B-splines, eq. (15),
 - compute the abscissae of their control points, eq. (18) and (19);
- calculate and store elements of matrix by evaluating corresponding splines at the coordinates of data points, thus establishing a sparse linear system

$$z_\nu = \sum_{i=1}^T \sum_{m=1}^6 b_{i,m} \tilde{M}((x_\nu, y_\nu) | K_{i,m}) \quad (29)$$

for the unknown coefficients $b_{i,m}$ with (x_ν, y_ν, z_ν) , $\nu = 1, 2, \dots$, covering the positional data;

- remove zero columns of matrix if there are any;
- add equations corresponding to additional information, e.g. gradients;
- solve the system in the sense of (weighted) least squares;

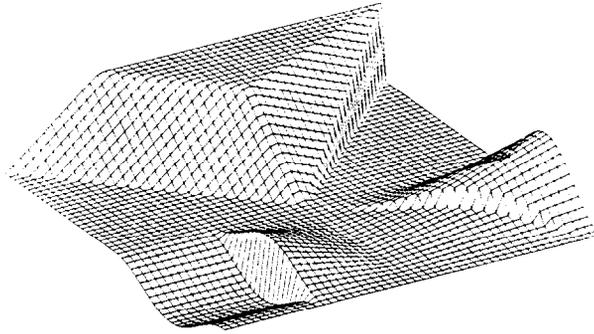


Fig. 9a. Mathematical test function: perspective view.

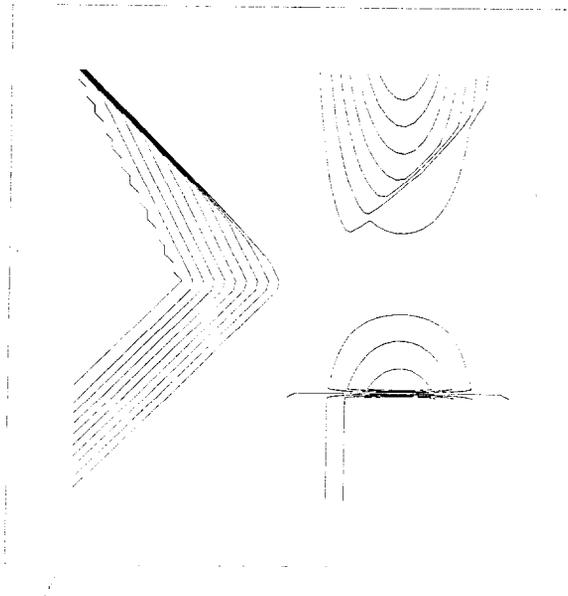


Fig. 9b. Contour lines of test function in Fig. 9a.

- repair for zero columns, i.e. for gaps in the data by estimating missing coefficients from neighboring coefficients;
- visualize the computed surface;
- visualize coefficients of the linear combination as control points;
- establish different levels of local control by optionally regrouping of coefficients, resp. control points, in different ways.

4. Example

To give a practical example of the versatility of this approach we applied the procedure outlined in the previous section to a mathematical test function composed of tilted planes and trigonometric functions such that the resulting surface is mathematically represented as a given bivariate function with various discontinuities as depicted in Figs. 9a and 9b. We would like to emphasize here that displaying the contour lines often provides a more sensitive control of the

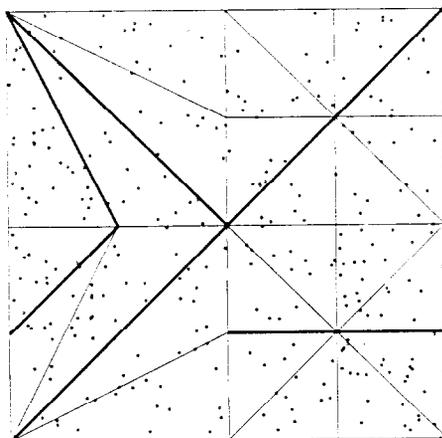


Fig. 10. Data sites (\circ) in the domain Ω , vertices of triangles adjusted to areal distribution of data sites, predefined edges reflecting a priori knowledge of the surface to be approximated or modeled (bold lines), and edges according to (suboptimal) constrained Delaunay triangulation honouring the predefined edges.

features of a surface even though we used a simple standard routine to compute them from a regular quadratic grid of calculated z -values.

The mathematical surface was sampled at 300 randomly generated data sites in the domain Ω , which was then triangulated, given the user defined vertices adjusted to the areal distribution of data sites and the predefined edges according to the known locations of the discontinuities, see Fig. 10.

In the next step the knots were defined by pulling the vertices apart; note collinear knots along predefined edges, Fig. 11.

In Fig. 12 the subregions B_t (shaded) of each triangle t are shown; any empty or degenerated B_t subset would indicate that some knots may not be most appropriately

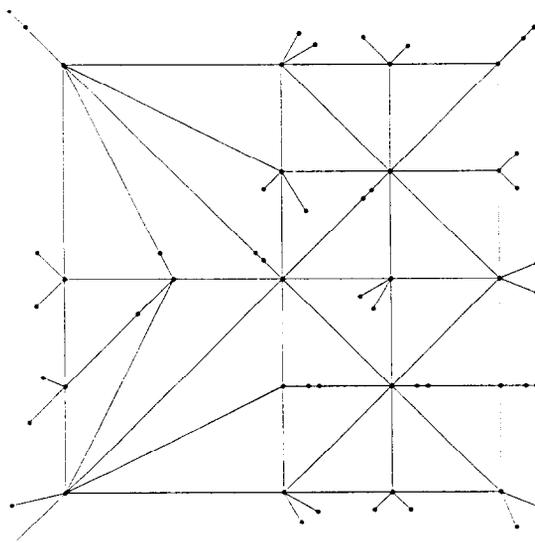


Fig. 11. Definition of knots by pulling vertices apart such that (i) partition of unity over Ω is guaranteed, (ii) all knots are in general position except for only those (iii) along predefined edges to reproduce the known discontinuities.

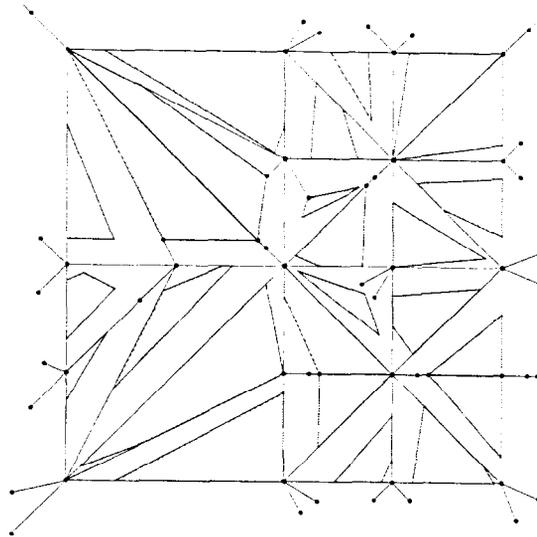


Fig. 12. Subregions B_i (shaded) of each triangle t where only the six B-splines associated with t are different from zero and all other vanish, and such that $s(x)|_{B_i}$ is a bivariate quadratic polynomial.

numbered or placed, and warn that the normalization and computation of the abscissae of control points may get numerically unstable.

The surface of the linear combination of quadratic B-splines approximating the given data in least squares sense is shown in Figs. 13a and 13b revealing small deviations of the initial surface only. A three-dimensional perspective view of the geometric control points is conveyed by Fig. 13c whereas Fig. 13d shows the abscissae of the control points in the data domain Ω .

5. Conclusions

Using simplicial B-splines associated with irregular triangles for the purposes of approximation and geometric modeling, the shortcomings of tensor product B-splines are overcome which

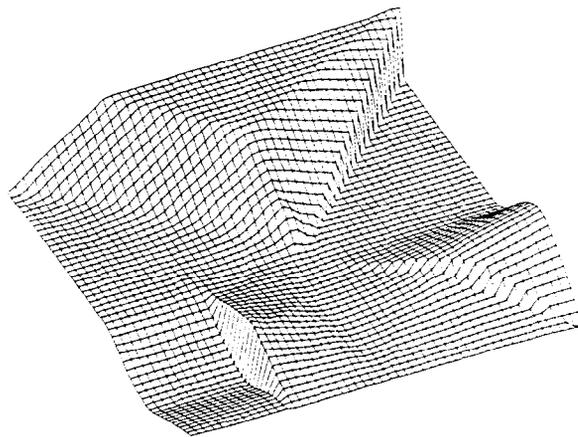


Fig. 13a. Computed surface of linear combination of quadratic B-splines: perspective view.

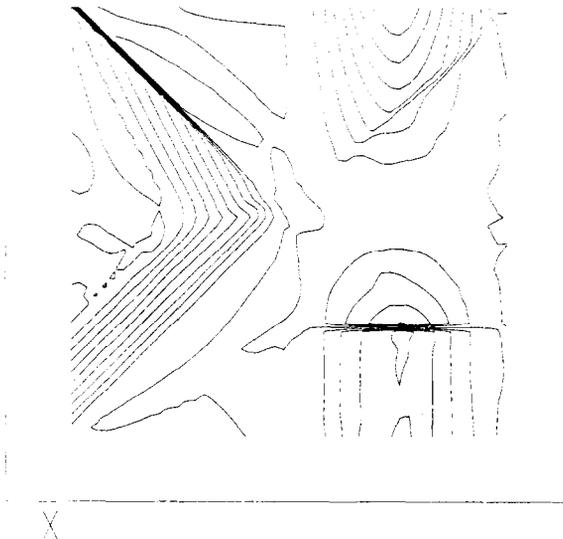


Fig. 13b. Contour lines of surface in Fig. 13a.

are implied by the fact that tensor product splines require a grid of knots topologically equivalent to a rectangular grid, while at the same time many of their favorable features are preserved. To be more specific, polynomial B-splines associated with irregular triangles provide a method combining all the properties required by geometric modeling of geologic surfaces and bodies:

- triangles are adjustable to the spatial distribution of data, resp. their abscissae;
- data providing positional (x , y , z) information can be processed;
- data providing information on any directional derivatives can be processed when available or accessible; however, they are not necessarily required;

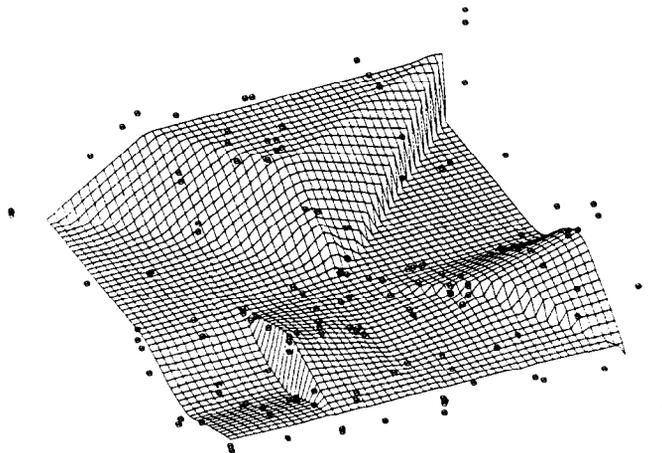


Fig. 13c. Surface of Fig. 13a with geometric control points.

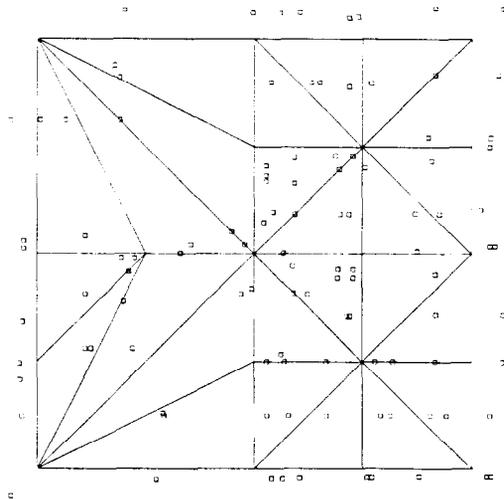


Fig. 13d. Abscissae of control points in Fig. 13c.

- data providing information on the geometry to be recovered/ modeled that can be formalized into discontinuities of the function itself or any of its directional derivatives can be processed;
- given a topology consistent with the data, i.e. an approximate parameterization of the geometry to be recovered/ modeled, parametric surfaces can be represented which initially cannot be represented by functions;
- local control of the geometry is provided by corresponding control points establishing the 'convex hull', thus interactive computer aided geometric modeling is possible.

Thus, after the constructive approach towards multivariate splines by [Dahmen & Micchelli '82a] had been elaborated for applications to practical problems in two- and three-dimensional approximation and interpolation, e.g. by [Gmelig Meyling '86], it may now also be used for practical applications of computer aided geometric design as here we exploit the interpretation of coefficients as geometric meaningful control points.

Acknowledgements

This communication is part of the results accomplished within the project 'computer aided design of geological maps' within the priority program 'digital geoscientific maps' of the German Science Foundation (DFG), which Prof. R. Vinken has been the scientific director of, cf. [Vinken '86]. Special thanks are due to Prof. W. Boehm, Braunschweig, for his introduction into the world of CAGD, and Prof. W. Dahmen, Berlin, Prof. K. Scherer, Bonn, and Prof. C.R. Traas, Twente, for many instructive discussions on splines and their practical applicability.

Two authors (S.A. and H.S.) thank for financial support by the WGerman Science Foundation, another (M.N.) thanks for financial support by the Netherlands Foundation of Mathematics.

References

- Auerbach, S. (1990). Approximation mit bivariaten B-splines über Dreiecken, Ms Thesis, Dept. of Applied Mathematics, Univ. of Bonn, FRG, to appear.

- Auerbach, S. and Schaeben, H. (1990). Processing digitized contour lines to digital surface representations, *Math. Geol.* 22, 723–742.
- Boehm, W., Farin, G. and Kahmann, J. (1984). A survey of curve and surface methods in CAGD, *Computer Aided Geometric Design* 1, 1–60.
- Cavendish, J.C. (1974). Automatic triangulation of arbitrary planar domains for the finite element method, *Intern. J. Numer. Meth. Eng.* 8, 679–696.
- Cline, A.K. and Renka, R.L. (1984). A storage efficient method for construction of a Thiessen triangulation, *Rocky Mountain J. Math.* 14, 119–139.
- Dahmen, W. (1979). Polynomials as linear combinations of multivariate B-splines, *Math. Z.* 169, 93–98.
- Dahmen, W. (1986). Bernstein–Bézier representation of polynomial surfaces, *ACM SIGGRAPH 86*, Dallas, TX, Aug. 18–22, 1986.
- Dahmen, W. and Micchelli, C.A. (1981). Numerical algorithms for least squares approximation by multivariate B-splines, in: Collatz, L., Meinardus, G., and Werner, H., eds., *Numerical Methods of Approximation Theory 6*, Birkhäuser, Basel, 85–114.
- Dahmen, W. and Micchelli, C.A. (1982a). Multivariate splines — a new constructive approach, in: Barnhill, R.E. and Boehm, W., eds., *Surfaces in Computer Aided Geometric Design*, Proc. Conf. at Oberwolfach, Apr. 25–80, 1982, North-Holland, Amsterdam 191–215.
- Dahmen, W. and Micchelli, C.A. (1982b). On the linear independence of multivariate B-splines: I. Triangulations of simploids, *SIAM J. Numer. Anal.* 19, 992–1012.
- Dahmen, W. and Micchelli, C.A. (1983a). On the linear independence of multivariate B-splines: II. Complete configurations, *Math. Comp.* 41, 143–163.
- Dahmen, W. and Micchelli, C.A. (1983b). Recent progress in multivariate splines, in: Chui, C.K., Schumaker, L.L. and Ward, J.D., eds., *Approximation Theory IV*, Academic Press, New York, 27–121.
- Farin, G. (1986). Triangular Bernstein–Bézier patches, *Computer Aided Geometric Design* 3, 83–127.
- Gmelig Meyling, R.H.J. (1986). Polynomial spline approximation in two variables, Ph.D. Thesis, University of Amsterdam.
- Goodman, T.N.T. and Lee, S.L. (1981). Spline approximation operators of Bernstein–Schoenberg type in one and two variables, *J. Approx. Theory* 33, 248–263.
- Grandine, T.A. (1986). Computing with multivariate simplex splines, in: Chui, C.K., Schumaker, L.L. and Ward, J.D., eds., *Approximation Theory V*, Academic Press, New York, 359–362.
- Grandine, T.A. (1987). The computational costs of simplex spline functions, *SIAM J. Numer. Anal.* 24, 887–890.
- Grandine, T.A. (1988). The stable evaluation of multivariate simplex splines, *Math. Comp.* 50, 197–205.
- Haber, R., Shepard, M.S., Abel, J.F., Gallagher, R.H. and Greenberg, D.P. (1981). A general two-dimensional, graphical finite element preprocessor utilizing discrete transfinite mappings, *Int. J. Num. Meth. Eng.* 17, 1015–1044.
- Höllig, K. (1982). Multivariate splines, *SIAM J. Num. Anal.* 19, 1013–1031.
- Höllig, K. (1986). Multivariate splines, Lecture Notes, AMS short course series, New Orleans.
- Lawson, C.L. (1972). Generation of a triangular grid with application to contour plotting, Jet Propulsion Laboratory, Internal Report 299, Pasadena.
- Lawson, C.L. (1977). Software for C1 surface interpolation, in: Rice, J.R., ed., *Mathematical Software III*, Proc., University of Wisconsin, Madison, March 28–30, 1977, Academic Press, New York, 161–194.
- Magnus, E.R., Joyce, C.C. and Scott, W.D. (1983). A spiral procedure for selecting a triangular grid from random data, *J. Appl. Math. Phys. (ZAMP)* 34, 231–235.
- Micchelli, C.A. (1979). On a numerically efficient method for computing multivariate B-spline, in: Schempp, W. and Zelker, K., eds., *Multivariate Approximation Theory*, Birkhäuser, Basel, 211–248.
- Micchelli, C.A. (1980). A constructive approach to Kergin interpolation in \mathbb{R}^k : Multivariate B-splines and Lagrange interpolation, *Rocky Mountain J. Math.* 10, 485–497.
- Schumaker, L.L. (1987). Triangulation methods, in: Chui, C.K., Schumaker, L.L. and Utreras, F.I., eds., *Topics in Multivariate Approximation*, Academic Press, New York, 219–232.
- Sibson, R. (1978). Locally equiangular triangulations, *Computer J.* 21, 243–245.
- Sibson, R. (1981). A brief description of natural neighbourhood interpolation, in: Barnett, V.D., ed., *Graphical Methods for Multivariate Data*, Wiley, Chichester, 21–36.
- Traas, C.R. (1990). Practice of bivariate quadratic simplicial splines, in: W. Dahmen, M. Gasca and C.A. Micchelli, eds., *Computation of Curves and Surfaces*, Kluwer Academic Publ., Dordrecht, 383–422.
- Vinken, R. (1986). Digital geoscientific maps: A priority program of the German society for the advancement of scientific research, *Math. Geol.* 18, 237–246.