ELSEVIER

Journal of Computational and Applied Mathematics 73 (1996) 45–64

JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS

On a class of polynomial triangular macro-elements

Paolo Costantini[a,*], Carla Manni[b]

[a] *Dipartimento di Matematica, Via del Capitano 15, 53100 Siena, Italy*
[b] *Dipartimento di Energetica, Via C. Lombroso 6/17, 50134 Firenze, Italy*

Received 11 September 1995; revised 24 November 1995

Abstract

In this paper we present a new class of polynomial triangular macro-elements of arbitrary degree which are an extension of the classical Clough–Tocher cubic scheme. Their most important property is that the degree plays the role of a tension parameter, since these macro elements tend to the plane interpolating the vertices data. Graphical examples showing their use in scattered data interpolation are reported.

Keywords: Bernstein–Bézier nets; Bernstein/Bézier polynomials; Scattered data interpolation

1. Introduction

The Clough–Tocher cubic macro-elements were originally introduced for finite element methods (see, e.g., [7, 31]), but it was soon realized that they could also provide a useful tool for the interpolation of scattered data. We refer to [2, 5, 17] for an introduction to their basic properties, and we mention [1, 16, 26] as examples of further generalizations and modifications. Here we simply recall that these basic cubic macro-elements can be locally defined using the function and the gradient values at the vertices of the triangle and they form a globally C^1 function. The resulting interpolating surface has a simple polynomial representation and can be evaluated with low computational cost, but it does exhibit a high dependence on the scheme adopted for triangulating the data. In addition we note that the Clough–Tocher interpolant is uniquely defined by the data and so no modification can be made of its shape. The relevance of this observation relies on the comparison with the tension methods, which have been widely used both in one- and two-dimensional tensor-product interpolation, to control the form of the interpolant, typically for visual or shape-preserving purposes. We refer to [6, 24, 28, 30] as examples of some popular methods, and recall that the usual tension approach consists of using a class of

* Corresponding author.

0377-0427/96 $15.00
PII S0377-0427(96)00033-7

parameter-dependent interpolants, in which the parameter is used to straighten the function toward a linear (one-dimensional case) or bilinear (two-dimensional case) form.

To our knowledge there is no tension-like method for scattered data interpolation, although the desirability of controlling the shape remains unchanged. In this paper we try to partially fill the gap, proposing a new class of triangular macro-elements of arbitrary degree n, which, in the case $n = 3$, coincide with the classical Clough–Tocher macro-element, and, as n tends to infinity, tend to the plane interpolating the data points at the triangle vertices.

This "tension feature" enables us to locally preserve the monotonicity and/or the convexity of the data inside the subtriangles of the Clough–Tocher split along directions parallel to the edges of the main triangulation.

The macro-elements are a triangular extension of a one-dimensional idea developed some years ago, and which we briefly recall for an easier understanding of the method. Let $l(x; n)$, $n \in \mathbb{N}$, $n \geqslant 3$, be a piecewise linear function defined in the interval $[0, 1]$ by

$$l(x;n)=\begin{cases} f_0 + f_0' x & x \in \left[0, \dfrac{1}{n}\right), \\ f_0 + \dfrac{f_0'}{n} + \dfrac{1}{n-2}\left[n(f_1 - f_0) - (f_0' + f_1')\right]\left(x - \dfrac{1}{n}\right), & x \in \left[\dfrac{1}{n}, \dfrac{n-1}{n}\right), \\ f_1 + f_1'(x-1), & x \in \left[\dfrac{n-1}{n}, 1\right], \end{cases} \tag{1.1}$$

and let $b(x; n)$ be its corresponding Bernstein–Bézier polynomial of degree n:

$$b(x;n) := \sum_{j=0}^{n} \binom{n}{j} l\left(\frac{j}{n}; n\right) x^j (1-x)^{n-j} \tag{1.2}$$

as shown in Fig. 1.

It is immediate to check that, for any n

$$b(0;n) = f_0, \quad b(1;n) = f_1, \quad \frac{\mathrm{d}}{\mathrm{d}x} b(x;n)|_{x=0} = f_0', \quad \frac{\mathrm{d}}{\mathrm{d}x} b(x;n)|_{x=1} = f_1',$$

and that, as n increases, $b(.; n)$ tends to uniformly approximate the straight line joining $(0, f_0)$ and $(1, f_1)$. A straightforward consequence is that, for sufficiently large n, $b(x; n)$ has the shape induced by the data f_0, f_1, f_0', f_1'. We observe that $b(x; n)$ depends, for any n, upon four parameters (actually, it belongs to a four-dimensional linear space) and high degrees do not effect the stability of the method (we need not worry about the high oscillations of classical polynomial interpolation) and produce a very limited increase in computational time [10]. In other words, the degree is nothing more than a tension parameter.

On the other hand, this tension scheme is polynomial with a very simple Bézier net which implies very simple checks on the shape of the function, and, more important, it allows extensions to bivariate interpolation. As a consequence, starting from basic one-dimensional results [8, 9], some two-dimensional schemes for the interpolation of data on a rectangular grid have been proposed in [11–13], while in [14] a method for interpolation of data distributed in a general tensor-product

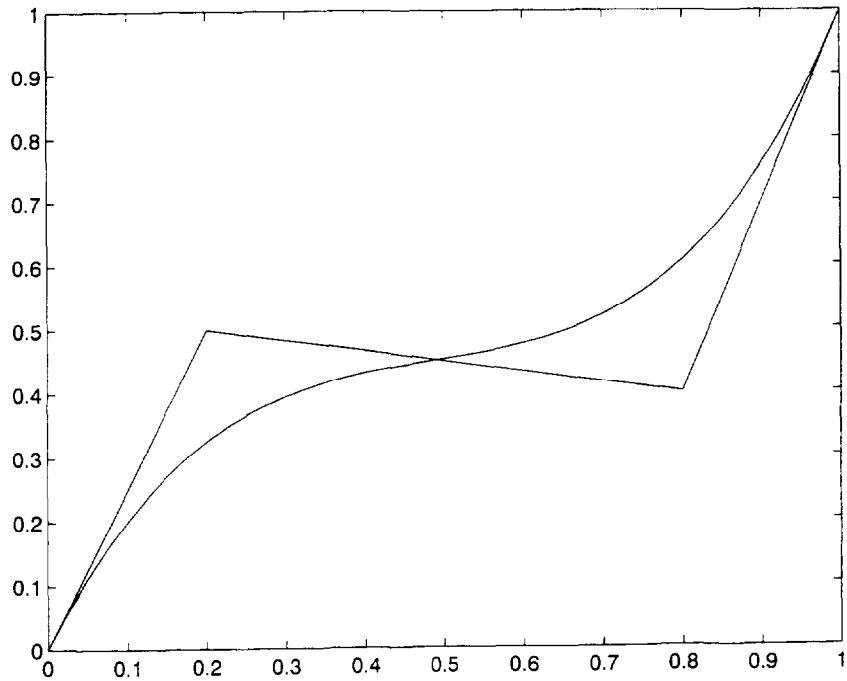

Fig. 1. An example of $b(x; n)$ and its net $l(x; n)$, $n = 5$.

topology has been considered. We note also that polynomials of the form (1.2) have been used in [20, 21] mainly from the parametric curve-fitting point of view.

This paper is divided into four sections. In the next one we describe the n-degree macro-element, and in Section 3 we show its shape-preserving and approximation properties. Finally, in Section 4, we develop a simple algorithm for computing the degree in order to satisfy the shape-preserving property and show its performance with some graphical examples.

2. The n-degree macro-element

It is now standard practice in computer-aided geometric design or in approximation theory to describe polynomials using their Bernstein–Bézier form, which allows significant simplifications in the description of their geometric structure. We start by briefly recalling some basic notations and properties. Let

$$P_r = \begin{bmatrix} x_r \\ y_r \end{bmatrix}, \quad r = 1, 2, 3,$$

be three noncollinear points in $\mathbb{R}^2$, and let T denote the triangle they form. An n-degree Bernstein–Bézier polynomial has the form (see, e.g., [2, 18])

$$b(x, y; n) = b(u, v, w; n) := \sum_{i+j+k=n} \frac{n!}{i!\,j!\,k!}\, l_{i,j,k}\, u^i v^j w^k, \tag{2.1}$$

where, setting

$$P = \begin{bmatrix} x \\ y \end{bmatrix} \in T, \quad u = u(x, y),\ v = v(x, y),\ w = w(x, y),$$

are the barycentric coordinates defined by

$$P = uP_1 + vP_2 + wP_3; \quad u + v + w = 1,$$

and $l_{i,j,k}$, $i + j + k = n$, are the *Bézier ordinates* of $b(.,.,.;n)$. If we take $x = x(u, v, w)$, $y = y(u, v, w)$ and connect in $\mathbb{R}^3$ the points (called *control points*)

$$L_{i,j,k.} := \begin{bmatrix} x\left(\frac{i}{n}, \frac{j}{n}, \frac{k}{n}\right) \\ y\left(\frac{i}{n}, \frac{j}{n}, \frac{k}{n}\right) \\ l_{i,j,k} \end{bmatrix}, \quad i, j, k \geqslant 0,\ i + j + k = n, \tag{2.2}$$

with triangular linear patches, we get the *control net*, $L = L(u, v, w)$. It plays a fundamental role in this paper.

In addition, a polynomial of the form (2.1) has the following interpolatory properties:

$$b(x_r, y_r; n) = l(x_r, y_r; n), \quad \nabla b(x_r, y_r; n) = \nabla l(x_r, y_r; n), \qquad r = 1, 2, 3, \tag{2.3}$$

where $l(x, y; n)$ is the piecewise continuous linear function made by $\frac{1}{2}n(n+1)$ patches such that

$$l\left(x\left(\frac{i}{n}, \frac{j}{n}, \frac{k}{n}\right), y\left(\frac{i}{n}, \frac{j}{n}, \frac{k}{n}\right); n\right) = l_{i,j,k.}, \quad i + j + k = n. \tag{2.4}$$

We will use the shorter notation $l(x, y) = l(x, y; n)$, $b(x, y) = b(x, y; n)$, whenever possible.

Now let the triples P_1, P_2, P_0 and Q_1, Q_2, Q_0, where $P_0 = Q_0$, $P_2, = Q_1$, form two adjacent triangles $T^{(1)}$ and $T^{(2)}$ (see also Fig. 2), and $L^{(1)}_{i,j,k}$, $L^{(2)}_{i,j,k}$ be the corresponding Bézier control points. It is well known that the two Bézier polynomials $b^{(1)}$ and $b^{(2)}$ form a C^0 surface across the common edge $P_0 P_2$ if the corresponding control points coincide along the edge, that is

$$L^{(1)}_{0,j,n-j} = L^{(2)}_{j,0,n-j}, \quad 0 \leqslant j \leqslant n,$$

and that they are C^1 across the above-mentioned edge if the triples

$$L^{(1)}_{0,j+1,n-j-1}, \quad L^{(1)}_{0,j,n-j}, \quad L^{(1)}_{1,j,n-j-1},$$

and

$$L^{(2)}_{j+1,0,n-j-1}, \quad L^{(2)}_{j,0,n-j}, \quad L^{(2)}_{j,1,n-j-1}, \qquad j = n-1, \ldots, 0,$$

lie on the same plane.

We finally recall the so-called *Clough–Tocher split* of a given triangle, T, which consists of dividing $T := P_1 P_2 P_3$, called in this context *macro-triangle*, in three *mini-triangles*

$$T^{(1)} := P_1 P_2 P_0, \quad T^{(2)} := P_2 P_3 P_0, \quad T^{(3)} := P_3 P_1 P_0,$$

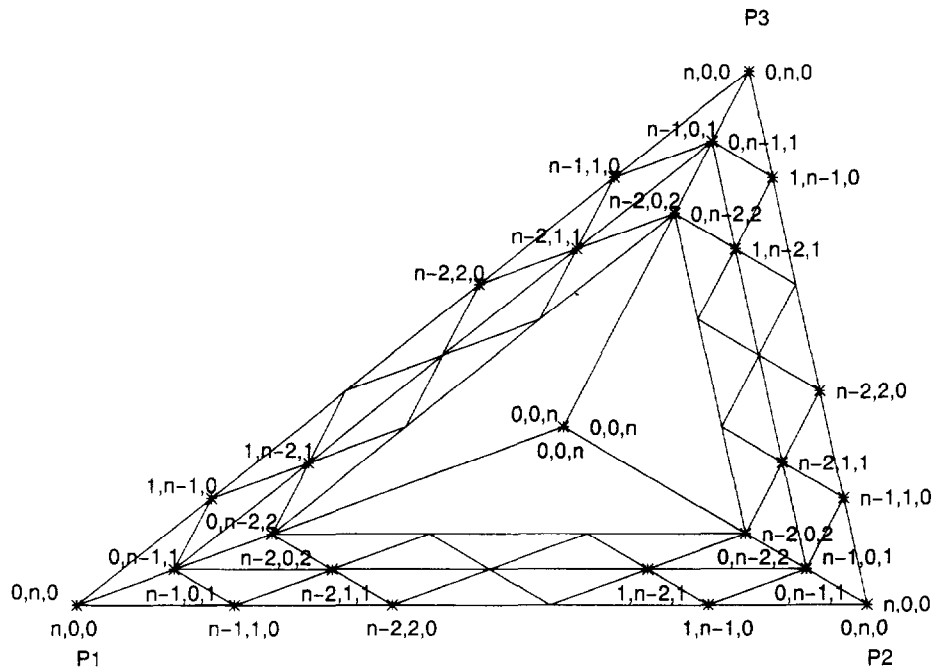

Fig. 2. The Clough–Tocher split of a macro-triangle with the distribution of the indices of the control points in each mini-triangle.

where P_0 is a point internal to T, often the barycenter of T (see Fig. 2).

Our main idea is to take an n-degree macro-element defined by control points $L^{(r)}_{i,j,k}$, of the form (2.2) in $T^{(r)}$, $r = 1, 2, 3$.

To construct the macro-element we will assume that the central control points $L^{(r)}_{i,j,k}$, $k \geqslant 2$, $r = 1, 2, 3$, all lie on the same plane, that is (see Figs. 2 and 3)

$$L^{(r)}_{0,0,n} = \beta_1 L^{(1)}_{n-2,0,2} + \beta_2 L^{(2)}_{n-2,0,2} + \beta_3 L^{(3)}_{n-2,0,2},$$

$$L^{(r)}_{i,j,k} = \frac{i}{n-2} L^{(r)}_{n-2,0,2} + \frac{j}{n-2} L^{(r)}_{0,n-2,2} + \frac{k-2}{n-2} L^{(r)}_{0,0,n}, \quad k \geqslant 2, \tag{2.5}$$

where β_r, $r = 1, 2, 3$, denote the barycentric coordinate of P_0, that is

$$P_0 = \beta_1 P_1 + \beta_2 P_2 + \beta_3 P_3, \beta_1 + \beta_2 + \beta_3 = 1.$$

In addition, we will require that the conditions for C^1 continuity hold for the remaining triples adjacent to a common internal edge, that is (see Figs. 2 and 3)

$$L^{(r)}_{1,n-1,0}, L^{(r)}_{0,n,0}, L^{(r)}_{0,n-1,1}, L^{(r+1)}_{n,0,0}, L^{(r+1)}_{n-1,1,0}, L^{(r+1)}_{n-1,0,1},$$

lie in the same plane, and the same is true for

$$L^{(r)}_{1,n-2,1}, L^{(r)}_{0,n-1,1}, L^{(r)}_{0,n-2,2}, L^{(r+1)}_{n-1,0,1}, L^{(r+1)}_{n-2,1,1}, L^{(r+1)}_{n-2,0,2},$$

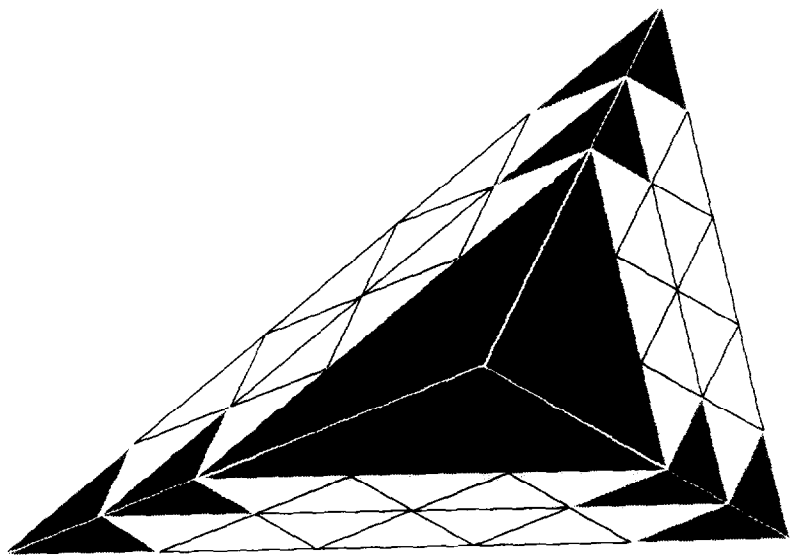

Fig. 3. The structure of the (two-dimensional projection of) the macro-element; the shaded parts correspond to planar patches.

where, here and in the following, $r-1 \in \mathbb{Z}_3$. These requirements imply the internal C^1 continuity of the n-degree macro-element constituted by the three Bézier polynomials $b^{(r)}$, $r = 1, 2, 3$, of the form (2.1). We emphasize that for $n = 3$ we obtain the classical Clough–Tocher cubic macro-element.

A simple inspection (see Fig. 3) shows that the macro-net has $3[n + (n-2)]$ degrees of freedom because the control points

$$\begin{matrix} L^{(r)}_{i,n-i,0}, & i = n, \ldots, 1, \\ L^{(r)}_{i,n-1-i,1}, & i = n-2, \ldots, 1, \end{matrix} \qquad r = 1, 2, 3,$$

can be arbitrarily chosen while the other are given by the planar conditions.

In order to reduce the number of free parameters, we assume that part of the control points of the first two rows in each mini-triangle lie on a straight line. To be more precise, we assume, for $r = 1, 2, 3$, (see Fig. 4)

$$L^{(r)}_{i,n-i,0} = \frac{i-1}{n-2} L^{(r)}_{n-1,1,0} + \frac{(n-2)-(i-1)}{n-2} L^{(r)}_{1,n-1,0}, \quad i = n-1, \ldots, 1,$$

$$L^{(r)}_{i,n-i-1,1} = \frac{i-1}{n-3} L^{(r)}_{n-2,1,1} + \frac{(n-3)-(i-1)}{n-3} L^{(r)}_{1,n-2,1}, \quad i = n-2, \ldots, 1. \tag{2.6}$$

In this way we have defined a linear space of polynomial macro-elements having dimension 15 for any $n > 3$ and dimension 12 for $n = 3$. We note also that the slices of the three mini-nets for $k = 0, 1$ have the form (1.1) and that the slices of Bézier polynomials at $u = 0$, $v = 0$ and $w = 0$ are one-dimensional polynomials of the form (1.2).

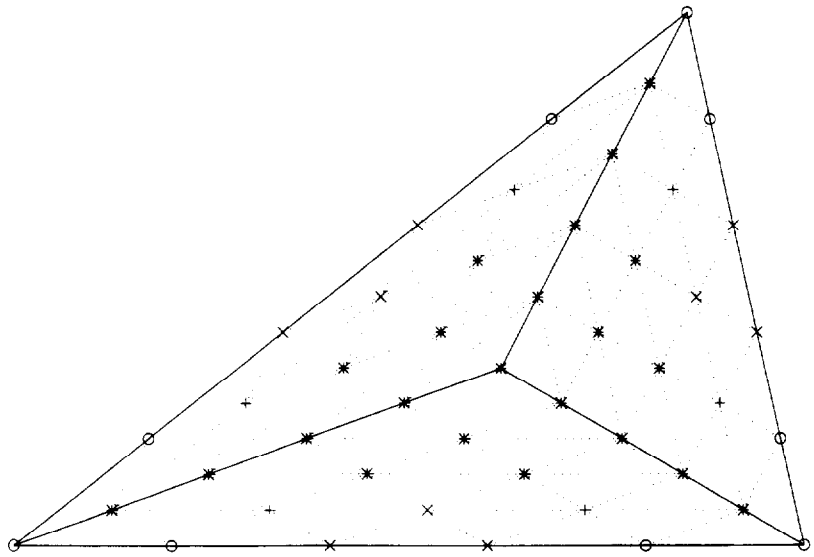

Fig. 4. How to compute Bézier ordinates of the macro-net: (○) from tangent planes (Eqs. (2.8)); (+) from cross boundary normal derivatives (Eqs. (2.9)); (×) from linear conditions (Eqs. (2.6)); (∗) from C^1 continuity conditions and planar conditions (Eqs. (2.10), (2.11) and (2.5)).

Now let us consider the interpolation problem and suppose that the data

$$f_r := f(P_r),\ \nabla f_r := \nabla f(P_r) = \begin{bmatrix} \dfrac{\partial f}{\partial x}(x_r, y_r) \\ \dfrac{\partial f}{\partial y}(x_r, y_r) \end{bmatrix}, \quad r = 1, 2, 3, \tag{2.7}$$

are given, where $f \in C^1(T)$.

Let Π_r be the orthogonal projection of P_0 onto the straight line through P_r and P_{r+1} and set

$$e_r := P_{r+1} - P_r, \quad t_r = P_0 - \Pi_r, \quad \rho_r := \Pi_r - P_r = \rho_r e_r.$$

Then we can uniquely define a macro-net, according to the previous assumptions, using the following conditions (see Fig. 4):

$$\begin{aligned} l^{(r)}_{n,0,0} &= f_r, \quad l^{(r)}_{n-1,1,0} = f_r + \frac{1}{n}\langle \nabla f_r, e_r \rangle, \\ l^{(r)}_{0,n,0} &= f_{r+1}, \quad l^{(r)}_{1,n-1,0} = f_{r+1} - \frac{1}{n}\langle \nabla f_{r-1}, e_r \rangle, \end{aligned} \tag{2.8}$$

and (see Fig. 4)

$$
\begin{aligned}
l^{(r)}_{n-2,1,1} &= f_r + \frac{1}{n}\langle \nabla f_r, e_r\rangle + \frac{1}{n}\left\langle \frac{n-2}{n-1}\nabla f_r + \frac{1}{n-1}\nabla f_{r+1}, t_r\right\rangle \\
&\quad + \frac{\rho_r}{n-2}\left(f_{r+1} - \frac{1}{n}\langle \nabla f_{r+1}, e_r\rangle - f_r - \frac{1}{n}\langle \nabla f_r, e_r\rangle\right), \\
l^{(r)}_{1,n-2,1} &= f_{r+1} - \frac{1}{n}\langle \nabla f_{r+1}, e_r\rangle + \frac{1}{n}\left\langle \frac{1}{n-1}\nabla f_r + \frac{n-2}{n-1}\nabla f_{r+1}, t_r\right\rangle \\
&\quad - \frac{1-\rho_r}{n-2}\left(f_{r+1} - \frac{1}{n}\langle \nabla f_{r+1}, e_r\rangle - f_r - \frac{1}{n}\langle \nabla f_r, e_r\rangle\right).
\end{aligned}
\tag{2.9}
$$

In addition, setting

$$\lambda^{(r)}(\alpha) := (1-\alpha)\langle \nabla f_r, t_r\rangle + \alpha\langle \nabla f_{r+1}, t_r\rangle,\ \alpha \in [0,1],$$

we observe that (2.9) are equivalent to say that the control point $L^{(r)}_{n-2,1,1}$ ($L^{(r)}_{1,n-2,1}$) lies on the plane through $L^{(r)}_{n-1,1,0}$ and $L^{(r)}_{n-2,2,0}$ ($L^{(r)}_{2,n-2,0}$ and $L^{(r)}_{1,n-1,0}$) having cross boundary normal derivative (along the edge e_r) given by $\lambda^{(r)}(1/(n-1))$ ($\lambda^{(r)}((n-2)/(n-1))$). Using (2.6) we also see that any control point in the second row, $L^{(r)}_{i,n-i-1,1}$, $i = n-2, \ldots, 1$, belongs to the plane through $L^{(r)}_{i+1,n-i-1,0}$, and $L^{(r)}_{i,n-i,0}$ having cross boundary normal derivative (along the edge e_r) given by $\lambda^{(r)}((n-1-i)/(n-1))$, $i = n-2, \ldots, 1$. We have assumed, in other words, that the cross boundary normal derivative of the control net is a linear function along the edge $P_r P_{r+1}$.

Note that in the case $n = 3$, the two equations in (2.9) do coincide and produce the standard conditions for cross boundary normal derivatives used in the Clough–Tocher interpolation scheme.

In order to complete the definition of the macro-net, we obtain from C^1 continuity conditions and from (2.8), (2.9)

$$
\begin{aligned}
l^{(r)}_{n-1,0,1} &= \beta_r\, l^{(r)}_{n,0,0} + \beta_{r+1}\, l^{(r)}_{n-1,1,0} + \beta_{r+2}\, l^{(r+2)}_{1,n-1,0} = f_r + \frac{\rho_r}{n}\langle \nabla f_r, e_r\rangle + \frac{1}{n}\langle \nabla f_r, t_r\rangle, \\
l^{(r)}_{0,n-1,1} &= \beta_r\, l^{(r)}_{1,n-1,0} + \beta_{r-1}\, l^{(r)}_{0,n,0} + \beta_{r+2}\, l^{(r+1)}_{n-1,1,0} = f_{r+1} - \frac{1-\rho_r}{n}\langle \nabla f_{r+1}, e_r\rangle \\
&\quad + \frac{1}{n}\langle \nabla f_{r+1}, t_r\rangle,
\end{aligned}
\tag{2.10}
$$

$$
\begin{aligned}
l^{(r)}_{n-2,0,2} &= \beta_r\, l^{(r)}_{n-1,0,1} + \beta_{r+1}\, l^{(r)}_{n-2,1,1} + \beta_{r+2}\, l^{(r+2)}_{1,n-2,1}, \\
l^{(r)}_{0,n-2,2} &= \beta_r\, l^{(r)}_{1,n-2,1} + \beta_{r+1}\, l^{(r)}_{0,n-1,1} + \beta_{r+2}\, l^{(r+1)}_{n-2,1,1}.
\end{aligned}
\tag{2.11}
$$

It is simple to verify that (2.8), (2.10) corresponds to

$$l^{(r)}(P_r) = f_r,\quad \nabla l^{(r)}(P_r) = \nabla f_r,\quad l^{(r)}(P_{r+1}) = f_{r+1},\quad \nabla l^{(r)}(P_{r+1}) = \nabla f_{r+1}.$$

Then from (2.3) we immediately have that each Bézier polynomial $b^{(r)}$ of the form (2.1) satisfies

$$b^{(r)}(P_r) = f_r,\quad \nabla b^{(r)}(P_r) = \nabla f_r,\quad b^{(r)}(P_{r+1}) = f_{r+1},\quad \nabla b^{(r)}(P_{r+1}) = \nabla f_{r+1}. \tag{2.12}$$

In addition, since $(\partial b^{(r)}/\partial t_r)(u, v, w)$, along the edge e_r, is a Bézier polynomial of degree $n-1$ with Bézier ordinates

$$\lambda^{(r)}\left(\frac{n-1-i}{n-1}\right), \quad i = n-1, \ldots, 0$$

(see [18]) it turns out that, via (2.9), we have imposed for $b^{(r)}$ linear cross boundary normal derivatives along the edge $P_r P_{r+1}$.

3. Some properties of the macro-net

In this section we will collect some properties of the n-degree macro-element, which are stated in the form of theorems just for notational purposes. Let $\mathscr{B}(x, y; n)$, $(x, y) \in T$, be the bivariate piecewise polynomial such that $\mathscr{B}(x, y; n)_{|T^{(r)}} = b^{(r)}(x, y; n)$, where $b^{(r)}$ is the bivariate Bernstein polynomial of the form (2.1) obtained via (2.8), (2.9), by the interpolatory conditions (2.12). Let $\mathscr{L}(x, y; n)$, $\mathscr{L}(x, y; n)_{|T^{(r)}} = l^{(r)}(x, y; n)$, be the corresponding z-component of the control net defined in (2.5). From the internal continuity conditions $\mathscr{B}(x, y; n) \in C^1(T)$ and, from (2.12), it interpolates the data (2.7) at the vertices of T.

Let us now put

$$\|g\|_{T,\infty} = \max_{(x,y)\in T} |g(x, y)|, \quad g \in C(T).$$

Then we have

Theorem 3.1. *Let $\Lambda = \Lambda(x, y)$ the plane interpolating the data (P_r, f_r), $r = 1, 2, 3$. Then*

$$\lim_{n\to\infty} \|b(.,.;n) - \Lambda\|_{T,\infty} = \lim_{n\to\infty} \|\mathscr{L}(.,.;n) - \Lambda\|_{T,\infty} = 0. \tag{3.1}$$

Proof. Since Λ, represented in the Bézier form, coincides with its control net for each value of n, it suffices to prove that

$$\lim_{n\to\infty} l^{(r)}_{i,j,k} = \Lambda(P^{(r)}_{i,j,k}), \; i + j + k = n,$$

where $P^{(r)}_{i,j,k}$ denotes the projection of $L^{(r)}_{i,j,k}$ on the $x - y$ plane. From (2.8)–(2.11) we have

$$\lim_{n\to\infty} l^{(r)}_{n-1,1,0} = \lim_{n\to\infty} l^{(r)}_{n-2,1,1} = \lim_{n\to\infty} l^{(r)}_{n-1,0,1} = \lim_{n\to\infty} l^{(r)}_{n-2,0,2} = f_r,$$

$$\lim_{n\to\infty} l^{(r)}_{1,n-1,0} = \lim_{n\to\infty} l^{(r)}_{1,n-2,1} = \lim_{n\to\infty} l^{(r)}_{0,n-1,1} = \lim_{n\to\infty} l^{(r)}_{0,n-2,2} = f_{r+1},$$

$$r = 1, 2, 3,$$

and, from (2.2), the control points $L^{(r)}_{i,j,k}$ with the above indices tend, respectively, to (P_r, f_r), (P_{r+1}, f_{r+1}). The assertion then follows from (2.6) observing that the central control points $L^{(r)}_{i,j,k}$, $k \geqslant 2$ lie on the plane through $L^{(r)}_{n-2,0,2}$, $r = 1, 2, 3$. □

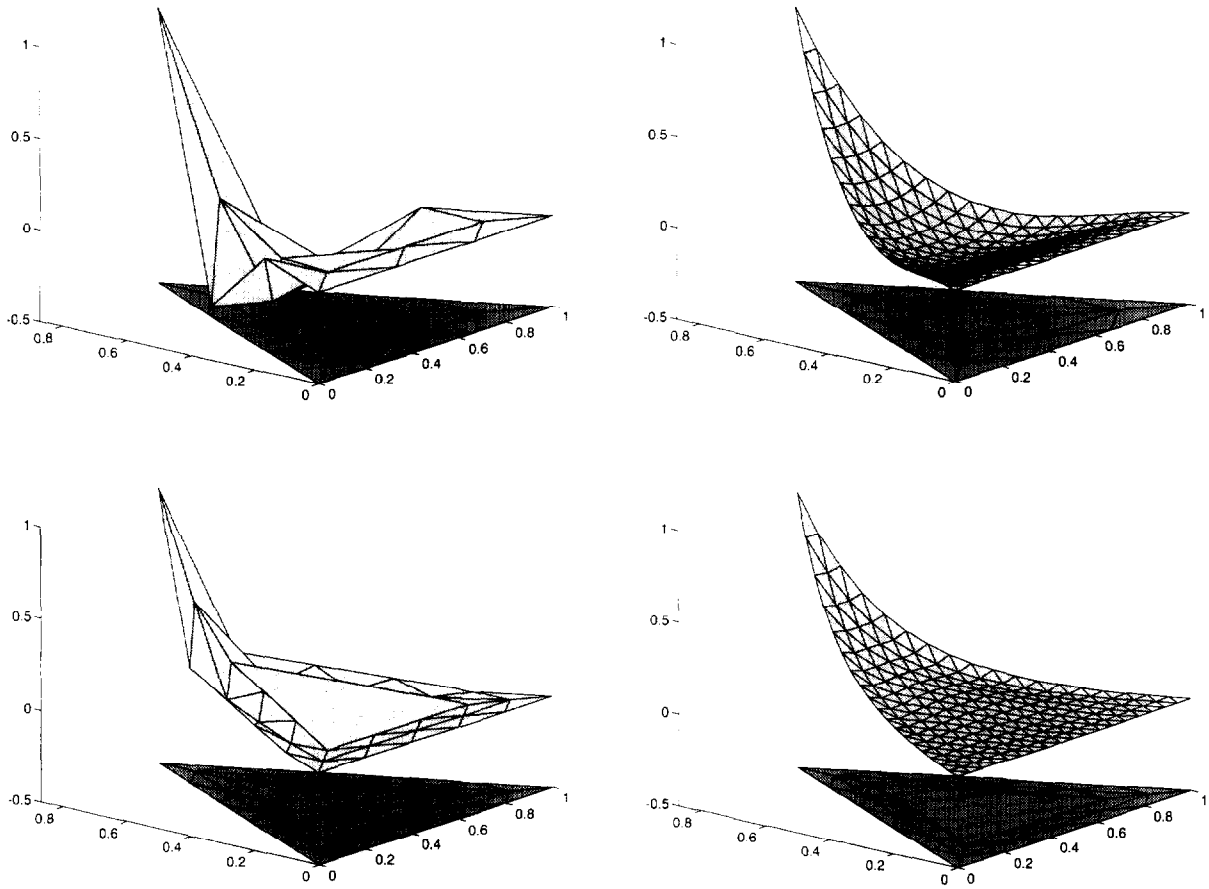

Fig. 5. The effect of the degree on the shape of the function. Top: The macro-net (left) and the corresponding polynomials for $n = 3$. Bottom: The macro-net (left) and the corresponding polynomials for $n = 7$.

Property (3.1), which is a trivial consequence of the construction of the macro-element, is the most relevant in our application. In fact, (3.1) says that the degree, n, is a tension parameter and that we can use it to control the shape of the net and of the corresponding Bézier polynomial. Fig. 5 shows the effect of enlarging the degree.

Theorem 3.1 suggests the idea of using the degree to reproduce the shape of the data. With this goal in mind we state the following definitions. Note that the second one is closely related but weaker than the notion of *axial convexity* given in [27].

Definition 3.2. The data (2.7) are increasing (decreasing) along the edge $P_r P_{r+1}$ if

$$f_r < f_{r+1}, \langle \nabla f_r, e_r \rangle > 0, \langle \nabla f_{r+1}, e_r \rangle > 0,$$

$$(f_r > f_{r+1}, \langle \nabla f_r, e_r \rangle < 0, \langle \nabla f_{r+1}, e_r \rangle < 0).$$

Definition 3.3. The data (2.7) are convex (concave) along the edge $P_r P_{r+1}$ if

$$\langle \nabla f_r, e_r \rangle < (f_{r+1} - f_r) < \langle \nabla f_{r+1}, e_r \rangle,$$

$$(\langle \nabla f_r, e_r \rangle > (f_{r+1} - f_r) > \langle \nabla f_{r+1}, e_r \rangle).$$

Then we have the following result.

Theorem 3.4. *Let $P_0 \in T$ be such that $\rho_r \in [0, 1]$, $r = 1, 2, 3$, and let the data (2.7) be increasing (decreasing) and/or convex (concave) along the edge $P_r P_{r+1}$. There exists a threshold degree, $\bar{n}$, such that for any $n \geqslant \bar{n}$ the interpolating piecewise Bézier polynomial $\mathscr{B}(.,.;n)$ is increasing (decreasing) and/or convex (concave) in the subtriangle $T^{(r)}$ along the lines parallel to the edge $P_r P_{r+1}$.*

Proof. Suppose the data are increasing along e_r (in a similar way we conclude in the decreasing case). From (2.8) there exists $\bar{n}_0$, such that for any $n \geqslant \bar{n}_0$,

$$l^{(r)}_{n,0,0} \leqslant l^{(r)}_{n-1,1,0} \leqslant l^{(r)}_{1,n-1,0} \leqslant l^{(r)}_{0,n,0}. \tag{3.2}$$

From the second expressions of (2.10) and from (2.9)

$$l^{(r)}_{n-2,1,1} - l^{(r)}_{n-1,0,1} = \frac{1-\rho_r}{n} \langle \nabla f_r, e_r \rangle + \frac{\rho_r}{n-2} (f_{r+1} - f_r) + \frac{1}{n(n-1)} \langle \nabla f_{r+1} - \nabla f_r, t_r \rangle - \frac{\rho_r}{n(n-2)} \langle \nabla f_{r+1} + \nabla f_r, e_r \rangle,$$

$$l^{(r)}_{0,n-1,1} - l^{(r)}_{1,n-2,1} = \frac{\rho_r}{n} \langle \nabla f_{r+1}, e_r \rangle + \frac{1-\rho_r}{n-2} (f_{r+1} - f_r) + \frac{1}{n(n-1)} \langle \nabla f_{r+1} - \nabla f_r, t_r \rangle - \frac{1-\rho_r}{n(n-2)} \langle \nabla f_{r+1} + \nabla f_r, e_r \rangle,$$

That is, (2.9) implies

$$l^{(r)}_{n-2,1,1} - l^{(r)}_{n-1,0,1} = \frac{1-\rho_r}{n} \langle \nabla f_r, e_r \rangle + \frac{\rho_r}{n} (f_{r+1} - f_r) + O\left(\frac{1}{n^2}\right),$$

$$l^{(r)}_{1,n-2,1} - l^{(r)}_{n-2,1,1} = (f_{r+1} - f_r) + O\left(\frac{1}{n}\right),$$

$$l^{(r)}_{0,n-1,1} - l^{(r)}_{1,n-2,1} = \frac{\rho_r}{n} \langle \nabla f_{r-1}, e_r \rangle + \frac{1-\rho_r}{n} (f_{r+1} - f_r) + O\left(\frac{1}{n^2}\right).$$

Then since the data are increasing along e_r (see Definition 3.2), and $\rho_r \in [0, 1]$, there exists $\bar{n}_1$, such that for any $n \geqslant \bar{n}_1$,

$$l^{(r)}_{n-1,0,1} \leqslant l^{(r)}_{n-2,1,1} \leqslant l^{(r)}_{1,n-2,1} \leqslant l^{(r)}_{0,n-1,1}. \tag{3.4}$$

Finally, from (2.11)

$$l^{(r)}_{0,n-2,2} - l^{(r)}_{n-2,0,2} = (f_{r+1} - f_r) + O\left(\frac{1}{n}\right),$$

so there exists $\bar{n}_2$, such that for any $n \geqslant \bar{n}_2$,

$$l^{(r)}_{n-2,0,2} < l^{(r)}_{0,n-2,2}. \tag{3.5}$$

Let us assume now the data (2.7) are convex along the edge e_r (similar results hold if the data are concave). From (2.8) and (2.6)

$$l^{(r)}_{n-1,1,0} - l^{(r)}_{n,0,0} = \frac{1}{n}\langle \nabla f_r, e_r\rangle,$$

$$l^{(r)}_{n-2,2,0} - l^{(r)}_{n-1,1,0} = \frac{1}{n}(f_{r+1} - f_r) + \mathrm{O}\left(\frac{1}{n^2}\right),$$

$$l^{(r)}_{0,n,0} - l^{(r)}_{1,n-1,0} = \frac{1}{n}\langle \nabla f_{r+1}, e_r\rangle,$$

$$l^{(r)}_{1,n-1,0} - l^{(r)}_{2,n-2,0} = \frac{1}{n}(f_{r+1} - f_r) + \mathrm{O}\left(\frac{1}{n^2}\right)$$

and

$$l^{(r)}_{n-i,i,0} - l^{(r)}_{n-i+1,i-1,0} = l^{(r)}_{n-i-1,i+1,0} - l^{(r)}_{n-i,i,0},\ i = 2, \ldots, n-2.$$

Then since the data are convex along e_r (see Definition (3.3)) there exists $\bar{n}_3$, such that for any $n \geqslant \bar{n}_3$,

$$l^{(r)}_{n-i-1,i+1,0} - 2l^{(r)}_{n-i,i,0} + l^{(r)}_{n-i+1,i-1,0} \geqslant 0,\ i = 1, \ldots, n-1. \tag{3.6}$$

In the same way from (3.3), recalling (2.6),

$$l^{(r)}_{n-3,2,1} - 2l^{(r)}_{n-2,1,1} + l^{(r)}_{n-1,0,1} = \frac{1-\rho_r}{n}[(f_{r+1} - f_r) - \langle \nabla f_r, e_r\rangle] + \mathrm{O}\left(\frac{1}{n^2}\right),$$

$$l^{(r)}_{0,n-1,1} - 2l^{(r)}_{1,n-2,1} + l^{(r)}_{2,n-3,1} = \frac{\rho_r}{n}[\langle \nabla f_{r+1}, e_r\rangle - (f_{r+1} - f_r)] + \mathrm{O}\left(\frac{1}{n^2}\right)$$

and

$$l^{(r)}_{n-i,i-1,1} - l^{(r)}_{n-i-1,i-2,1} = l^{(r)}_{n-i-1,i,1} - l^{(r)}_{n-i,i-1,1},\ i = 3, \ldots, n-2.$$

Then, since the data are convex along e_r and $\rho_r \in [0, 1]$, there exists $\bar{n}_4$, such that for any $n \geqslant \bar{n}_4$,

$$l^{(r)}_{n-i-1,i,1} - 2l^{(r)}_{n-i,i-1,1} + l^{(r)}_{n-i+1,i-2,1} \geqslant 0,\ i = 2, \ldots, n-1. \tag{3.7}$$

Now, it we set

$$\bar{n} = \max_{i=0,\ldots,4}\{\bar{n}_i\}$$

then the subnet $L^{(r)}$ is increasing and/or convex along the direction of the edge e_r for any $n \geqslant \bar{n}$ because of (2.5) and (2.6).

The assertion then follows from the expression of the directional derivatives of the Bézier polynomials [18]. □

Remark 3.5. The hypothesis $\rho_r \in [0, 1]$, that is Π_r belonging to the edge $P_r P_{r+1}$, is not restrictive since it is always possible to choose P_0 such that it is satisfied; for example, we can take P_0 as the center of the circle inscribed in T. However, from the computational point of view, it is better to consider the barycenter of the triangle whenever it satisfies the mentioned hypothesis.

Remark 3.6. The previous theorem ensures that the macro-net has shape-preserving properties along the directions of the edges of T, and we can observe that, especially for convexity, global properties should be preferable (see, for example, [19]). These, however, cannot be achieved for arbitrary data. If we consider the complete macro-net we have in fact the Grandine's result [19] showing that the only convex macro-nets are the planar ones. However the following example shows that even the single mini-net can be nonconvex for arbitrary data values.

Example 3.7. Suppose a set of convex data is given. To ensure a mini-net is convex the following condition is necessary:

$$\frac{l^{(r)}_{n-1,0,1} + l^{(r)}_{n-2,1,1}}{2} \leqslant \frac{l^{(r)}_{n-1,1,0} + l^{(r)}_{n-2,0,2}}{2}. \tag{3.8}$$

Let us now consider the paraboloid

$$f(x, y) = (x - 1)^2 + \left(y - \frac{\sqrt{3}}{3}\right)^2,$$

the equilateral triangle T with vertices $P_1 = (0, 0)$, $P_2 = (2, 0)$, $P_3 = (1, \sqrt{3})$, and let P_0 be the barycenter of T. In this case, with a straightforward manipulation (3.8) becomes, for $r = 1$,

$$\langle \nabla f_1, e_1 \rangle \tfrac{1}{3} + \langle \nabla f_1, e_3 \rangle \tfrac{1}{6} + O\left(\frac{1}{n}\right) \geqslant 0,$$

but, substituting the gradient value, the left member of the inequality is equal to $-\frac{4}{3} + \frac{4}{6} + O(1/n)$, hence (3.8) cannot hold for large values of n, and the mini-net is not globally convex.

We conclude this section with the following elementary property.

Theorem 3.8. *Let $f \in C^2(T)$. Then for any n,*

$$\| f(.,.) - \mathscr{B}(.,.; n) \|_{T,\infty} = O(h^2), \quad h := \max_{r=1,2,3} \| P_{r+1} - P_r \|.$$

Proof. For r arbitrary but fixed, let $\pi(x, y) = f_r + \langle \nabla f_r, [x - x_r, y - y_r]^T \rangle$ the truncated Taylor expansion of f around P_r. Then for each $(x, y) \in T^{(r)}$

$$\begin{aligned} |f(x, y) - \mathscr{B}(x, y; n)| &= |f(x, y) - b^{(r)}(x, y; n)| \\ &\leqslant |f(x, y) - \pi(x, y)| + |\pi(x, y) - b^{(r)}(x, y; n)| = O(h^2), \end{aligned}$$

since, from (2.12), π is also the truncated Taylor expansion for $b^{(r)}$. □

4. Scattered data interpolation

As previously said in the introduction, this section is devoted to developing a simple algorithm for computing a C^1 surface interpolating a set of scattered data. The surface is composed of triangular polynomial macro-elements and the degree (the same for each patch) is computed in a global automatic way, according to Theorem (3.4), and it reproduces, as far as possible, the shape of the data.

Given a set of scattered data

$$P_\eta = (x_\eta, y_\eta), \quad f_\eta = f(P_\eta), \quad \nabla f_\eta = \nabla f(P_\eta), \qquad \eta = 1, \ldots, N_P, \tag{4.1}$$

let

$$T_\mu = P_{\mu_1} P_{\mu_2} P_{\mu_3}, \quad \mu = 1, \ldots, N_T, \tag{4.2}$$

be a corresponding set of non-overlapping triangles (the practical aspects concerning the gradients, supposed to be known a priori, and the choice of a triangulation method will be discussed later).

For any $n \in \mathbb{N}$, using (2.5), (2.8)–(2.11), it is immediate to construct, for any triangle T_μ, $\mu = 1, \ldots, N_T$, a macro-polynomial $\mathscr{B}_\mu(x, y; n)$ such that

$$\mathscr{B}_\mu(x_{\mu_r}, y_{\mu_r}; n) = f_{\mu_r}, \quad \nabla \mathscr{B}_\mu(x_{\mu_r}, y_{\mu_r}; n) = \nabla f_{\mu_r}, \quad r = 1, 2, 3, \tag{4.3}$$

and the surface $s(x, y; n)$ given by

$$s(x, y; n) := \mathscr{B}_\mu(x, y; n), (x, y) \in T_\mu,$$

turns out to be a C^1 function interpolating the data (4.1). In fact, the control nets of two adjacent triangles have the first two rows parallel to the common edge given by (2.6), (2.8) and (2.9) which define the same conditions for the two macro-nets. This, in turn, implies that contiguous triples of control points, adjacent to the same edge, lie on the same plane. Figure 6 shows an example of such a composite surface. The data reported in the top have been interpolated assuming zero gradients and the data points. The graphs obtained with $n = 3$ and $n = 15$, depicted in the center and in the bottom respectively, clearly show the visual consequences of the use of large degrees.

As Theorem (3.1) suggests, the degree n of the polynomials can be used as a parameter to obtain the desired tension of the surface. If we are interested in reproducing the shape of the data, we can compute the degree so that the hypotheses of Theorem (3.4) are satisfied for all the triangles T_μ using the following scheme

Algorithm 4.1.

1. *Let the data* (4.1) *and the triangulation* (4.2) *be given.*
2. *For* $\mu = 1, \ldots, N_T$
   - 2.1. *For* $r = 1, 2, 3$
     - 2.1.1. *Check if the data are increasing (decreasing) and/or convex (concave) according to Definitions* 3.2 *and* 3.3 *along the edge* $P_{\mu_r} P_{\mu_{r+1}}$.
     - 2.1.2. *Compute the threshold degree* $\bar{n}_{\mu_r}$ *according to Theorem* 3.4.
   - 2.2. *Set* $\bar{n}_\mu := \max\{\bar{n}_{\mu_1}, \bar{n}_{\mu_2}, \bar{n}_{\mu_3}\}$
3. *Set* $n := \max\{\bar{n}_\mu, \mu = 1, \ldots, N_T\}$
4. *For* $\mu = 1, \ldots, N_T$
   - 4.1. *Compute the macro-element* $\mathscr{B}_\mu(x, y; n)$ *which satisfies* (4.3).

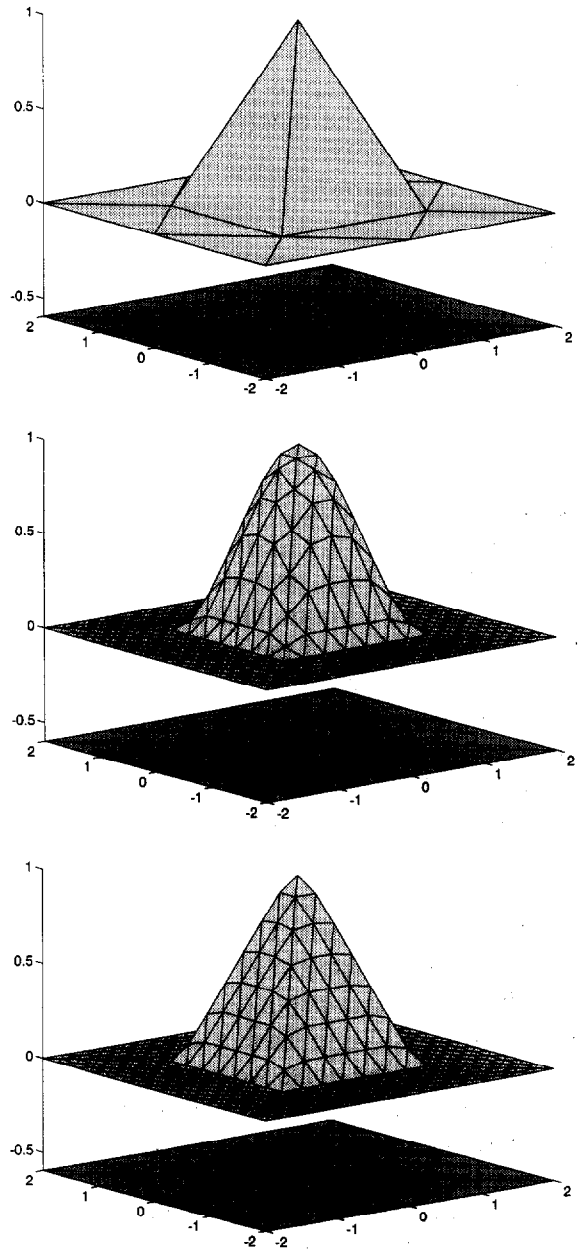

Fig. 6. An example of the composite surface. Top: the data. Center: the interpolating surface for $n = 3$. Bottom: The interpolating surface for $n = 15$.

We need some more details on step (2.1.2.). Let us consider for the sake of simplicity only the increasing and/or convex case. It is clear that the macro-net of $\mathcal{B}_\mu$ will have the same shape of the data if the corresponding mini-nets satisfy (3.2), (3.4)–(3.5) and/or (3.6)–(3.7). But Theorem (3.4) says that we have to look for a threshold degree $\bar{n}_{\mu_r}$ such that (3.2), (3.4)–(3.5) and/or (3.6)–(3.7) are

satisfied for any larger degree. It is unfortunately impossible to obtain from (3.2), (3.4)–(3.7) simple stronger inequalities ensuring the asymptotic behavior of the net (see, for comparison, the analogous problem in [12]), because they lead to polynomial inequalities in $1/n$. However, the computation of the required elements of the macro-net is very cheap and we can adopt the following simple scheme.

Algorithm 4.2.

1. *Let the data* $f_{\mu_r}, f_{\mu_{r+1}}$, ∇f_{μ_r}, $\nabla f_{\mu_{r+1}}$, *increasing and/or convex, and the integer* $n_{\max}$ *be given.*
2. *Set* $n_{\text{temp}} = n_{\max}$.
3. *While* ((3.2), (3.4)–(3.5) *and/or* (3.6)–(3.7) *are satisfied with* $n = n_{\text{temp}}$).
   3.1. $n_{\text{temp}} = n_{\text{temp}} - 1$.
4. *Set* $\bar{n}_{\mu_r} = n_{\text{temp}} + 1$.

Here $n_{\max}$ is the maximum degree allowed by the user, and in principle, $n_{\max} = +\infty$ (for example, in all the following tests we have used $n_{\max} = 20$). It is worthwhile to recall that we have no problem in dealing with large values of $n_{\max}$ since the macro-element tend uniformly to a plane, and so we do not have to worry about instabilities and oscillations. In addition, the structure of the macro-net makes possible to evaluate each polynomial using only $O(n)$ addenda in (2.1).

Algorithm 4.1 requires knowing the values of the gradients at the interpolation points, but, in practice, this information is often not available. It is clear that our resulting surface, as well as any other two dimensional shape preserving interpolant of gridded data, will depend heavily on the method we have adopted to recover the gradients from the data points. The problem of a good choice of the partial derivatives has been solved in some cases for tensor-product constrained interpolation, [3], but, to the best of our knowledge, has never been investigated in the scattered data setting.

Similarly, Algorithm 4.1 requires the data have been organized in a triangulation, and it is widely known how significantly the triangulation method affects the shape of the interpolant. The most famous method is the so-called Lawson or Delaunay scheme [22], based on the max–min angle criterion, but new algorithms, based on data-dependent strategies have been recently developed [4,15, 23, 25, 29]. Obviously, we would like to have a data-dependent triangulation, especially tailored for our macro-elements and for the goals we want to achieve.

Summarizing, a good shape-preserving method is given by three ingredients: (a) a good class of interpolating functions, (b) a good method to compute the gradients and (c) a good triangulation scheme. So far, we have obtained some results on item (a), but, for producing the examples of this section, we have used the Lawson method and we have computed the gradients as the weighted least-squares approximation of the neighboring data slopes. The development of better choices for (b) and (c), specialized for our problem, are under investigation.

In the first example we have interpolated a function originally introduced by Ritchie (see [15] for its analytical form) at a set of 25 scattered data points as shown in the top Fig. 7. In the center is shown the graph produced by the cubic Clough–Tocher interpolant. It is possible to see that its oscillations are strongly reduced in the bottom of Fig. 7, where the degree 18 produced by Algorithm 4.1 has been used.

In the second example, see Fig. 8, we have computed 36 scattered values of the sigmoidal function (see [9] for the analytical form) and we show the interpolating surface for $n = 3$ (center)

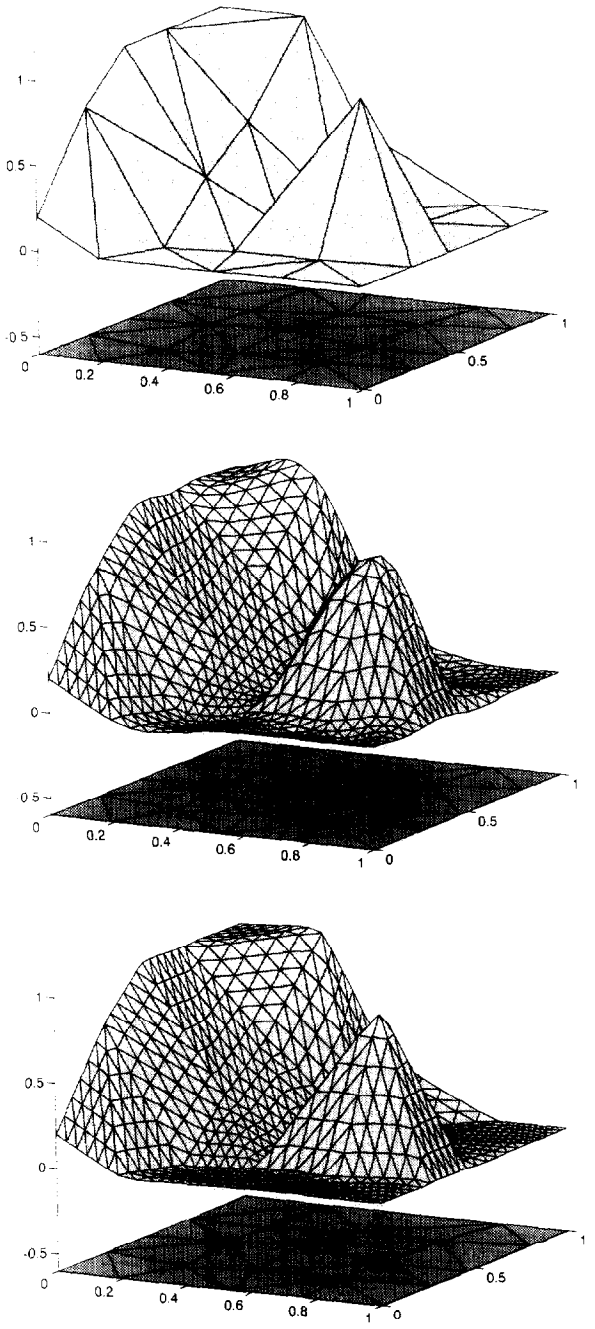

Fig. 7. Example 1. Top: The data. Center: The interpolating surface for $n = 3$. Bottom: The interpolating surface for $n = 18$.

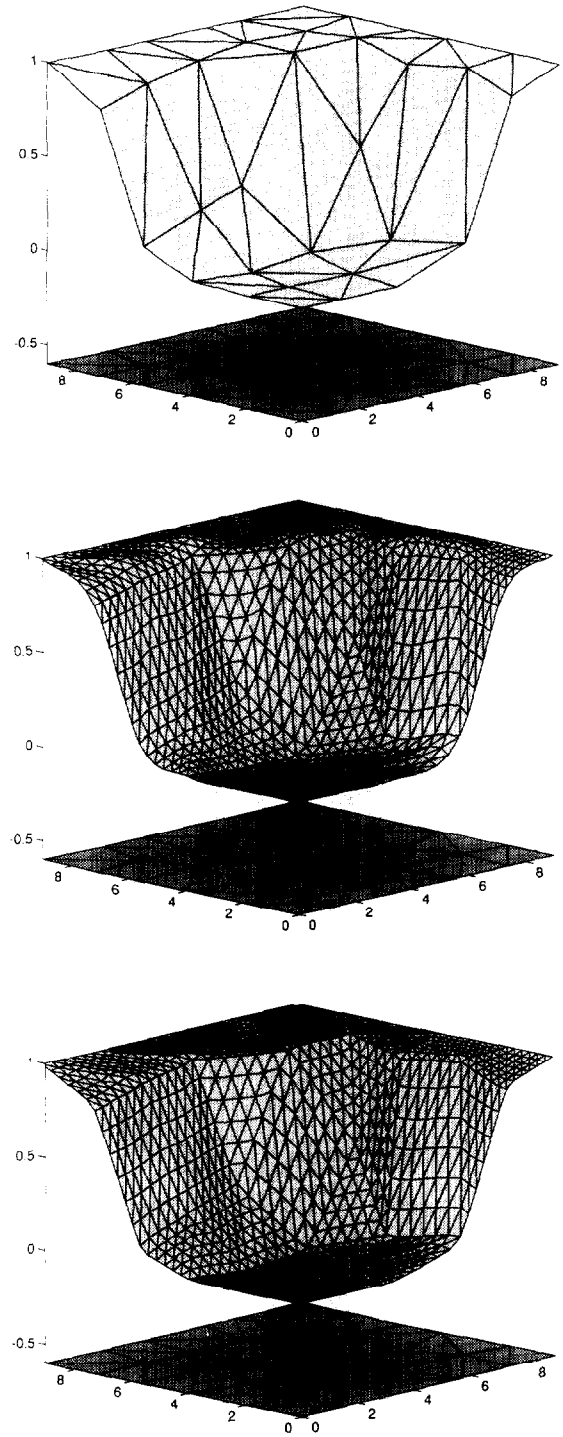

Fig. 8. Example 2. Top: The data. Center: The interpolating surface for $n = 3$. Bottom: The interpolating surface for $n = 13$.

and for $n = 13$, output value of Algorithm 4.1 (bottom). As in the previous test, we can see that the oscillations are reduced.

For both the examples only monotonicity constraints have been considered in Algorithm 4.1.

We point out that the surfaces in the bottom of Fig. 7 and 8 are not, strictly speaking, monotonicity preserving if compared with the data at the top. As previously said, this is due to the poor choice of the derivatives, which, following Definition (3.2), makes some triangles neither increasing nor decreasing in contrast to the shape of the data. This is the case, for example, of the triangles around the "hill" of Fig. 7.

We want to conclude this section with some comments concerning the major drawback of the present scheme, that is the global choice of the tension parameter. It is in fact clear that a large value of n, even if suitable for some "sharp" subset of data, does force the other triangles to accept useless strong tension factors, and therefore the resulting surface could, in some cases, not be visually pleasing. In addition, we have the obvious disadvantage that all the data have to be processed for a single evaluation of the interpolating function.

We anticipate that using some properties of the net and of the Bézier polynomials, it seems possible to modify our macro-element and obtain a local algorithm which provides a globally C^1 surface, where any triangular polynomial patch has its own, locally computed, degree. However, due to space limitations, the corresponding results, which are still under study, will be reported in a subsequent paper.

Acknowledgements

We want to express our thanks to Prof. F.I. Utreras and to Prof. P. Sablonnière for the helpful discussions we had with them at different stages of our work.

References

[1] P. Alfeld, A bivariate C^2 Clough–Tocher scheme, *Computer Aided Geometric Des.* **1** (1984) 257–267.

[2] W. Böhm, G. Farin and J. Kahman, A survey of curves and surfaces methods in CAGD, *Computer Aided Geometric Des.* **1** (1984) 1–60.

[3] R.E. Carlson and F.N. Fritsch, An algorithm for monotone piecewise bicubic interpolation, *SIAM J. Numer. Anal.* **26** (1989) 230–238.

[4] J.M. Carnicer and W. Dahmen, Convexity preserving interpolation and Powell–Sabin elements, *Comput. Aided Geometric Des.* **9** (1992) 279–289.

[5] P.G. Ciarlet, Sur l'élement de Clough et Tocher, *RAIRO Anal. Numér.* **R2** (1974) 19–27.

[6] A.K. Cline, Scalar and planar valued curve fitting using splines under tension, *Comm. ACM* (17), (1974) 218–223.

[7] R.W. Clough and J.L. Tocher, Finite element stiffness matrices for analysis of plates in bending, *Proc. Conf. on Matrix Methods in Structural Mechanics*, Wright–Patterson A.F.B., OH, 1965.

[8] P. Costantini, Co-monotone interpolating splines of arbitrary degree. A local approach, *SIAM J. Sci. Statist. Comput*, **8** (1987) 1026–1034.

[9] P. Costantini, An algorithm for computing shape-preserving interpolating splines of arbitrary degree, *J. Comput. Appl. Math.* **22** (1988) 89–136.

[10] P. Costantini, On some recent methods for shape-preserving bivariate interpolation, in: W. Haussmann and K. Jetter, Eds., *Multivariate Interpolation and Approximation*, International Series in Numerical Mathematics, Vol. 94 (Birkhäuser, Base, 1990) 59–68.

[11] P. Costantini and F. Fontanella, Shape-preserving bivariate interpolation, *SIAM J. Numer. Anal.* **27** (1990) 488–506.
[12] P. Costantini and C. Manni, A local scheme for bivariate comonotone interpolation, *Comput. Aided Geometric Des.* **8** (1991) 371–391.
[13] P. Costantini and C. Manni, A bicubic shape-preserving blending scheme, *Computer Aided Geometric Des.*, to appear.
[14] P. Costantini and C. Manni, Monotonicity-preserving interpolation of non-gridded data, *Comput. Aided Geometric Des.*, to appear.
[15] N. Dyn and S. Rippa, Data dependent triangulations for scattered data interpolation, *Appl. Numer. Math.* **12** (1993) 89–105.
[16] G. Farin, A modified Clough–Tocher interpolant, *Comput. Aided Geometric Des.* **2** (1985) 19–27.
[17] G. Farin, Triangular Bernstein–Bézier patches, *Comput. Aided Geometric Des.* **3** (1986) 83–127.
[18] G. Farin, *Curves and Surfaces for Computer Aided Geometric Design* (Academic Press, London, 1988).
[19] A. Grandine, On convexity of piecewise polynomial functions on triangulations, *Comput. Aided Geometric Des.* **6** (1989) 181–187.
[20] P.D. Kaklis and D.G. Pandelis, Convexity preserving polynomial splines of nonuniform degree, *IMA J. Numer. Anal.* **10** (1990) 223–234.
[21] P.D. Kaklis and N.S. Sapidis, Convexity-preserving interpolatory parametric splines of non-uniform polynomial degree, *Comput. Aided Geometric Des.* **12** (1995) 1–26.
[22] C.L. Lawson, Software for C^1 surface interpolation, in: J.R. Rice, Ed., *Mathematical Software III* (Academic Press, New York, 1977) 161–194
[23] E. Quak and L.L Schumaker, Cubic spline fitting using data dependent triangulations, *Comput. Aided Geometric Des.* **7** (1990) 293–301.
[24] R.J. Renka, Algorithm 716. TSAPACK: tension spline curve fitting package, *TOMS* **19** (1993) 81–94.
[25] S. Rippa, Long thin triangles can be good for linear interpolation, *SIAM J. Numer. Anal.* **20** (1992) 257–270.
[26] P. Sablonnière, Composite finite elements of class C^k, *J. Comput. Appl. Math.* **12&13** (1985) 541–550.
[27] T. Sauer, Multivariate Bernstein polynomials and convexity, *Comput. Aided Geometric Des.* **8** (1991) 465–478.
[28] J.W. Schmidt, Rational biquadratic C^1 splines in S-convex interpolation, *Computing* **47** (1991) 87–96.
[29] L.L. Schumaker, Computing optimal triangulations in polygonal domains, *Comput. Aided Geometric Des.* **10** (1993) 329–345.
[30] D.G. Schweikert, Interpolatory tension splines with automatic selection of tension factors, *J. Math. Phys.* **45** (1966) 312–317.
[31] G. Strang and G. Fix, *An Analysis of the Finite Element Method* (Prentice-Hall, Englewood Cliffs, NJ, 1973).