

# The mixed directional difference-summation algorithm for generating the Bézier net of a trivariate four-direction Box-spline

G. Casciola<sup>a</sup> E. Franchini<sup>b</sup> L. Romani<sup>a,\*</sup>

<sup>a</sup>*Department of Mathematics, University of Bologna,  
P.zza di Porta San Donato 5, 40127 Bologna, Italy.*

<sup>b</sup>*Department of Pure and Applied Mathematics, University of Padova,  
Via G. Belzoni 7, 35131 Padova, Italy.*

---

## Abstract

Trivariate Box-splines lack an efficient and general exact evaluation technique. This paper presents one possible and underexploited approach to solving this problem. The algorithm we propose is based on mixed directional differences and summations for computing the Bézier net coefficients of all trivariate four-direction Box-splines of any degree over tetrahedral tessellations of the domain.

A Matlab package, called MDDS, for computing the Bézier net both in the trivariate and bivariate cases, is also provided.

*Key words:* Trivariate Box-splines, Recurrence Relations, Exact Evaluation, Tetrahedral Bézier Volume Decomposition, B-net.

*AMS Classification:* 65D17 68U05 33F99 65D07

---

## 1 Introduction

During the past twenty years, much research has been undertaken to study Bézier representations of bivariate Box-spline basis functions ([6], [7], [8], [24]). In contrast, Bézier representation of higher dimensional Box-splines has received much less attention as an effective and powerful exact evaluation tool.

---

\* Corresponding author.

*Email addresses:* `casciola@dm.unibo.it` (G. Casciola),  
`elena.franchini@inwind.it` (E. Franchini), `romani@dm.unibo.it` (L. Romani).

In this paper we focalize our attention on trivariate Box-splines and we propose a new efficient computational scheme for exact evaluating the four-direction class of such functions by computation of their Bézier net over a tetrahedral tessellation of the domain.

Trivariate Box-splines are a trivariate extension of uniform univariate B-splines. Trivariate Box-spline evaluation is, in general, more difficult than univariate B-spline evaluation. General evaluation algorithms can exploit one or more of the following properties of Box-splines: (i) two-scale subdivision, (ii) Fourier transform, (iii) recurrence definition formula, or (iv) tetrahedral Bézier volume decomposition.

Subdivision leads to an approximate iterative technique which converges quadratically. The continuous inverse Fourier transform of the Box-spline can only be approximated with a discrete inverse Fourier transform, so inverse FFT evaluation is also an approximation [22]. Both of the two first techniques share the property that a large number of samples are computed at once and have similar asymptotic complexity measures.

While procedures (i) and (ii) provide an efficient computational scheme only if an approximate Box-spline is required, of the relevant evaluation techniques, just (iii) and (iv) are exact. Unfortunately, (iii) can be very expensive: the value of every Box-spline at any point is in fact computed one at a time. Only fairly low degrees Box-splines can be computed without too much difficulty and for completely general Box-splines the cost increases combinatorically with the number of direction vectors. With care, on restricted classes of Box-splines, this explosion can be contained somewhat, but a large amount of arithmetic is still needed for every sample.

Strangely, evaluation via tetrahedral Bézier volume decomposition has not been looked at in the literature. Although for graphic display purposes it is worthwhile using an approximate evaluation procedure, for analytical purposes it is sometimes necessary to know the exact formulation of the polynomial pieces of a trivariate Box spline basis function. Additionally, in computer aided geometric design it is common to represent a polynomial on an interval or triangle by its Bézier points, since this is very useful (both in theory and in applications) as the appropriate piecewise linear interpolant of the Bézier points, the so-called Bézier net, reflects the shape of the polynomial curve or surface. Curiously, these results have not previously been extended to more than two dimensions. Therefore it is worthwhile to investigate this approach in the trivariate case too. Additionally, once the B-net of each polynomial piece is available, the box can be evaluated and its exact value at any point can be determined by using the regular Bézier polynomial evaluator ([25], [30]). For the purpose of graphically displaying the volume, it is worthwhile implementing the B-net subdivision algorithm, not only for efficiency reasons, but also for the reason that in using B-net subdivision, we have exact values of the volume at all the vertices of the subdivided tetrahedra. Thus, any interpolating volume based on translates of a trivariate Box-spline, could be displayed by our algorithm to preserve its interpolatory property.

This paper is organized as follows. Sections 2 and 3 are both introductory sections where we give notations and preliminary details. In particular, Section 2 is devoted to the analysis of the class of trivariate Box-spline volumes to which the

four-direction Box-splines belong, while Section 3 acquaints the reader with trivariate polynomials on tetrahedra, underlining the properties we need to work out the mathematics of our algorithm. In Section 4 we describe our computational scheme for generating the Bézier net coefficients of an arbitrary trivariate four-direction Box-spline. Details of computation and clarifying examples are presented. Then in Section 5 we show that a restriction of our computational scheme to the two-dimensional case provides an alternative procedure to the computational scheme proposed by Lai [24], Chui and Lai [6], [7], [8], for computing the Bézier net of bivariate three-direction Box-splines. Finally Section 6 is concerned with the description of a Matlab implementation of our algorithm for generating the B-nets of both trivariate and bivariate Box-splines.

## 2 Trivariate Box-spline volumes: notations and preliminary details

Box-splines represent a generalization of univariate uniform B-splines to several variables. Therefore we can think about a Box-spline as a multivariate piecewise polynomial function of some chosen degree, i.e. as a function that consists of different polynomial pieces of the same degree, defined over different parts of its domain, that join together with a certain order of continuity.

Since Box-splines were introduced by de Boor and De Vore [13], a rich theory has been developed and collected in “the box spline book” [18] which serves as a reference for the following exposition of these piecewise polynomial functions.

As underlined in this book there are many ways to derive Box-splines. Here we choose the recurrence definition and we confine ourselves to analyze the trivariate case only.

**Definition 1** *Let the degree- $m$ ,  $m \geq 0$ , trivariate Box-spline  $M_{D_n}^m : \mathbb{R}^3 \rightarrow \mathbb{R}$ , associated with the direction matrix  $D_n = [\mathbf{d}_1 \ \mathbf{d}_2 \ \dots \ \mathbf{d}_n] \in \mathbb{R}^{3 \times n}$ ,  $n = m + 3$ , with*

$$\text{span}(D_n) \equiv \text{span}(\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_n) = \mathbb{R}^3, \quad \mathbf{d}_h \in \mathbb{R}^3 \setminus \{0\} \quad \forall h = 1, \dots, n,$$

be defined  $\forall n > 3$  ( $m > 0$ ) by the recurrence relation

$$M_{D_n}^m(\mathbf{x}) = \frac{1}{n-3} \sum_{h=1}^n [t_h M_{D_n \setminus \{\mathbf{d}_h\}}^{m-1}(\mathbf{x}) + (1-t_h) M_{D_n \setminus \{\mathbf{d}_h\}}^{m-1}(\mathbf{x} - \mathbf{d}_h)] \quad (1)$$

where  $\mathbf{x} = \sum_{h=1}^n t_h \mathbf{d}_h$  is a representation of  $\mathbf{x} \in \mathbb{R}^3$  by some linear combination of

the columns  $\{\mathbf{d}_h\}_{h=1,\dots,n}$  of  $D_n$ , e.g. the least norm solution

$$\begin{bmatrix} t_1 \\ t_2 \\ \vdots \\ t_n \end{bmatrix} = D_n^t (D_n D_n^t)^{-1} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}.$$

The base case of this recursion occurs when the matrix  $D_n$  is square, that is  $n = 3$  ( $m = 0$ ). In this case  $M_{D_3}^0$  is the normalized characteristic function of the projected half-open parallelepiped  $H^3$ , spanned by the three columns of  $D_3$ :

$$M_{D_3}^0(\mathbf{x}) = \begin{cases} \frac{1}{|\det(D_3)|} & \text{if } \mathbf{x} \in H^3 \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

This Box-spline is piecewise constant, has degree  $m = 0$  and is discontinuous. In general, by the  $n$  columns of  $D_n$  (which may be interpreted as directions in  $\mathbb{R}^3$ ) we can determine the support of the piecewise polynomial and its continuity properties. In particular it has been proved that:

- The support of  $M_{D_n}^m$  is the set sum of the columns contained in  $D_n$ .
- $M_{D_n}^m$  is a piecewise polynomial of degree  $m = n - 3$ .
- $M_{D_n}^m$  is  $\rho - 2$  times continuously differentiable, where  $\rho$  is the minimal number of columns that need to be removed from  $D_n$  to obtain a matrix whose columns do not span  $\mathbb{R}^3$ .
- $M_{D_n}^m$  reproduces all polynomials of degree  $\rho - 1$  and none of degree higher than  $m = n - 3$ .

The class of trivariate Box-splines we are going to analyze is that defined by the direction vectors

$$e_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad e_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad e_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad e_{123} = e_1 + e_2 + e_3 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \quad (3)$$

which form a regular partition of  $\mathbb{R}^3$  into regular tetrahedra.

$E_3 \equiv \{e_1, e_2, e_3, e_{123}\}$  is only a subset of the complete direction set

$$\left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}, \begin{bmatrix} -1 \\ -1 \\ 1 \end{bmatrix} \right\}$$

containing all the 7 unit vectors of the domain of a trivariate Box-spline [27] (see figure 1).

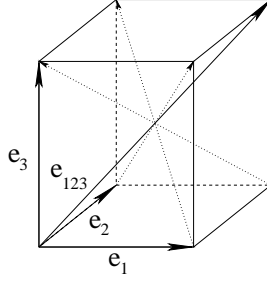


Fig. 1. The seven directions of the trivariate Box-spline.

The direction matrices  $D_n$  that we will associate with  $E_3$  are therefore matrices that contain the vectors  $e_1, e_2, e_3, e_{123}$  with their repetitions. More precisely, we will use the symbol  $M_{\nu_1\nu_2\nu_3\nu_4}^m$  to denote the four-direction Box-spline  $M_{D_n}^m$  associated with the direction matrix

$$D_n = \left[ \underbrace{e_1, \dots, e_1}_{\nu_1}, \underbrace{e_2, \dots, e_2}_{\nu_2}, \underbrace{e_3, \dots, e_3}_{\nu_3}, \underbrace{e_{123}, \dots, e_{123}}_{\nu_4} \right]$$

where  $n = \nu_1 + \nu_2 + \nu_3 + \nu_4$ ,  $\nu_h \in \mathbb{Z}_+ \setminus \{0\} \forall h = 1, \dots, 4$ .

In the bivariate case, the analogous of this class of Box-splines is the class of the three-direction Box-splines studied in [7], [8], which are defined by the direction vectors

$$E_2 = \left\{ \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\}$$

(see figure 2 right).

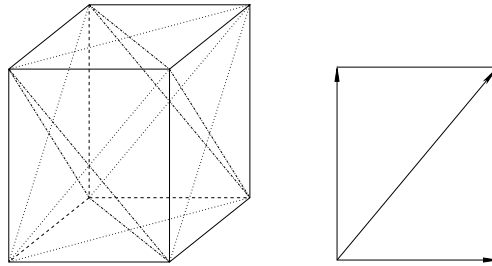


Fig. 2. Unit vectors on faces (left) - The bivariate three-direction set (right).

To represent a four-direction Box-spline by its Bézier net, we partition its support into cubes by a regular lattice and we consider a tessellation of each cube into six tetrahedra, as represented in figure 3. The tessellation of each cube into tetrahedra is the one which, once projected, gives the tessellation of each square of the support of a three-directional bivariate Box-spline into two regular triangles.

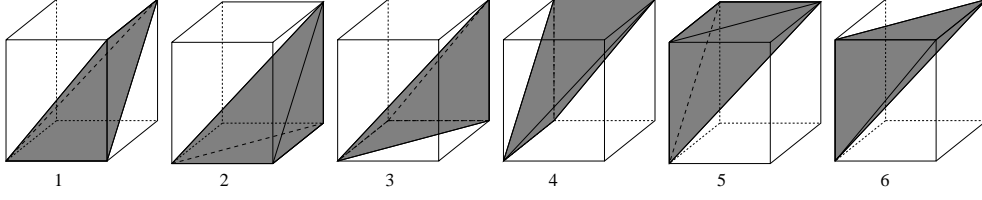


Fig. 3. Tetrahedral tessellation of a unit cube.

### 3 B-form of trivariate polynomials on tetrahedra: TB volumes

Given a trivariate Box-spline, i.e. a piecewise trivariate polynomial defined on a collection of tetrahedra, one can have the B-form of each polynomial piece restricted to one tetrahedron. In this section we acquaint the reader with the Bernstein representation of polynomials on tetrahedra. An introduction to this topic can be found in [16].

**Definition 2** Let  $\mathbf{v}^0, \mathbf{v}^1, \mathbf{v}^2, \mathbf{v}^3$  be four vertices in  $\mathbb{R}^3$ , whose Cartesian coordinates  $(x^h, y^h, z^h) = \mathbf{v}^h$ ,  $h = 0, \dots, 3$ , satisfy

$$\begin{vmatrix} 1 & x^0 & y^0 & z^0 \\ 1 & x^1 & y^1 & z^1 \\ 1 & x^2 & y^2 & z^2 \\ 1 & x^3 & y^3 & z^3 \end{vmatrix} \neq 0. \quad (4)$$

Thus, any non-degenerate tetrahedron  $T$ , with non-zero (signed) volume

$$\text{vol}(T) = \frac{1}{6} \begin{vmatrix} 1 & x^0 & y^0 & z^0 \\ 1 & x^1 & y^1 & z^1 \\ 1 & x^2 & y^2 & z^2 \\ 1 & x^3 & y^3 & z^3 \end{vmatrix}, \quad (5)$$

is defined by the convex hull of the four affine independent vertices  $\mathbf{v}^0, \mathbf{v}^1, \mathbf{v}^2, \mathbf{v}^3$ , as follows:

$$T := \langle \mathbf{v}^0, \mathbf{v}^1, \mathbf{v}^2, \mathbf{v}^3 \rangle = \left\{ \mathbf{v} \in \mathbb{R}^3 : \mathbf{v} = \sum_{h=0}^3 \lambda_h \mathbf{v}^h, 0 \leq \lambda_h \leq 1 \forall h, \sum_{h=0}^3 \lambda_h = 1 \right\}. \quad (6)$$

The quadruple  $(\lambda_0, \lambda_1, \lambda_2, \lambda_3)$  in (6) identifies the so-called barycentric coordinates of the arbitrary point  $\mathbf{v} = (x, y, z) \in \mathbb{R}^3$  relative to the tetrahedron  $T = \langle \mathbf{v}^0, \mathbf{v}^1, \mathbf{v}^2, \mathbf{v}^3 \rangle$ .

**Remark 3** *Since each*

$$\lambda_h := \lambda_h(\mathbf{v}) = \frac{\text{vol}(\hat{T}_h)}{\text{vol}(T)}, \quad h = 0, \dots, 3 \quad (7)$$

with  $\hat{T}_h$  denoting the tetrahedron of vertices  $\mathbf{v}^0, \dots, \mathbf{v}^{h-1}, \mathbf{v}, \mathbf{v}^{h+1}, \dots, \mathbf{v}^3$ , is a linear polynomial in  $\mathbf{v} = (x, y, z)$  with value 1 at the vertex  $\mathbf{v}^h$ , which vanishes on the face opposite to  $\mathbf{v}^h$ , in the interior of the tetrahedron  $T$  we have  $\lambda_h > 0, \forall h = 0, \dots, 3$ .

We next introduce the trivariate Bernstein polynomials as follows:

$$B_{i,j,k,l}^m(\lambda) = \frac{(i+j+k+l)!}{i!j!k!l!} \lambda_0^i \lambda_1^j \lambda_2^k \lambda_3^l, \quad m = i+j+k+l. \quad (8)$$

Since each  $\lambda_h$  is a linear polynomial, clearly  $B_{i,j,k,l}^m$  are polynomials of degree  $m$ . In fact we have that the set  $\{B_{i,j,k,l}^m(\lambda), i+j+k+l=m\}$  is a basis for the space of all polynomials of total degree  $\leq m$ . As a consequence any polynomial  $p$  of degree  $m$  on  $T$  can be written uniquely in terms of  $B_{i,j,k,l}^m$ 's i.e.,

$$p(\mathbf{v}) = \sum_{i,j,k,l \geq 0, i+j+k+l=m} P_{i,j,k,l}^m B_{i,j,k,l}^m(\lambda), \quad P_{i,j,k,l}^m \in \mathbb{R}. \quad (9)$$

The representation (9) for polynomials is referred to as the Bézier form (B-form for short) of  $p$  over  $T$ . The  $P_{i,j,k,l}^m$  are the Bézier coefficients of  $p$  with respect to  $T$ .

**Remark 4** *To simplify notation, we use  $p(\mathbf{v})$ ,  $p(x, y, z)$ ,  $p(\lambda)$  and  $p(\lambda_0, \lambda_1, \lambda_2, \lambda_3)$  to denote the same trivariate function  $p$ .  $p(\mathbf{v})$  is also commonly called a tetrahedral Bézier volume (TB volume).*

**Remark 5** *It is interesting to notice a property of trivariate Bernstein Bézier volumes. Let inner Bézier control vertices denote control points  $P_{i,j,k,l}^m$  with all  $i, j, k, l \neq 0$ . The outer control points consequently denote points on the faces of the tetrahedron. We can state that all tetrahedral Bézier volumes of degree  $m$  have no inner control points for  $m \leq 3$ . This follows trivially since  $i+j+k+l=m$  implies one of  $i, j, k, l$  equal to zero if  $m \leq 3$ .*

Since the indices  $i, j, k, l$  are respectively associated with the vertices  $\mathbf{v}^0, \mathbf{v}^1, \mathbf{v}^2, \mathbf{v}^3$  in  $T$ , then we can associate each polynomial  $p(v)$  with its Bézier coefficients set  $\{P_{i,j,k,l}^m, i+j+k+l=m\}$  (see figures 4-5 as example).

**Definition 6** *The ordinates  $P_{i,j,k,l}^m \in \mathbb{R}$  associated with the barycentric abscissae  $\xi_{i,j,k,l} = \frac{i\mathbf{v}^0 + j\mathbf{v}^1 + k\mathbf{v}^2 + l\mathbf{v}^3}{i+j+k+l} \in \mathbb{R}^3$  with  $i, j, k, l \geq 0, i+j+k+l=m$ ,*

$$\left\{ \left( \frac{i\mathbf{v}^0 + j\mathbf{v}^1 + k\mathbf{v}^2 + l\mathbf{v}^3}{m}, P_{i,j,k,l}^m \right) : i+j+k+l=m \right\} \in \mathbb{R}^4,$$

constitute the Bézier net of  $p(\mathbf{v})$  in  $\mathbb{R}^4$ , which uniquely determines the polynomial  $p(\mathbf{v})$  and contains information about the geometric feature of the volume.

In fact, like in the univariate/bivariate cases, the appropriate piecewise linear interpolant of the four dimensional Bézier points  $(\xi_{i,j,k,l}, P_{i,j,k,l}^m) \in \mathbb{R}^4$  (obtained by connecting them with straight lines according to their natural order), commonly called the Bézier net, reflects the shape of the polynomial volume. This is one of the reasons that polynomials in Bézier form have been widely used in Computer Aided Geometric Design.

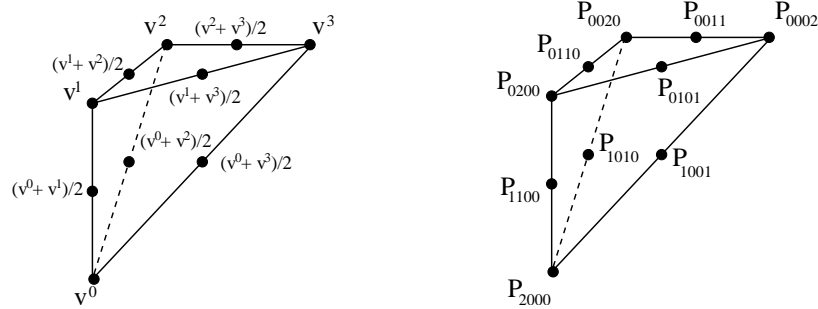


Fig. 4. Domain points on a degree-2 tetrahedron (left) and the corresponding Bézier coefficients (right).

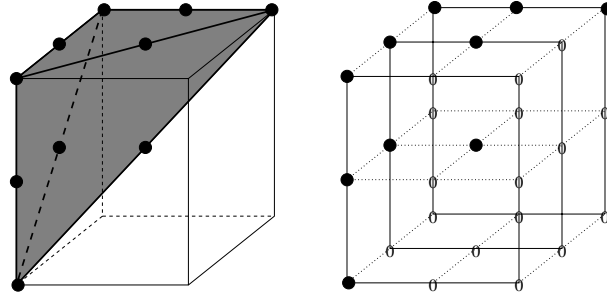


Fig. 5. A degree-2 trivariate matrix corresponding to tetrahedron no 5 in Fig.3.

Because of the linear dependence of the barycentric coordinates, the partial derivatives of a TB volume do not have an obvious geometric interpretation; indeed, the partial derivatives with respect to  $\lambda_0, \lambda_1, \lambda_2, \lambda_3$ , do not coincide with the derivatives along parametric lines. Instead of working with partial derivatives, we must therefore use directional derivatives. Now we are going to give formulas for the directional derivatives of  $p$  in a direction defined by a vector  $\mathbf{w}$ . In particular, for

$$p(\mathbf{v}) = \sum_{i,j,k,l \geq 0} P_{i,j,k,l}^m B_{i,j,k,l}^m(\lambda)$$

we have

$$D_{\mathbf{w}}p(\mathbf{v}) = m \sum_{i,j,k,l \geq 0} \Delta P_{i,j,k,l}^{m-1}(\mathbf{a}) B_{i,j,k,l}^{m-1}(\lambda) \quad (10)$$



with  $\mathbf{a} = (a_1, a_2, a_3, a_4)$ ,  $a_1 + a_2 + a_3 + a_4 = 0$ , and the following recurrence relation on the coefficients:

$$\Delta P_{i,j,k,l}^{m-1}(\mathbf{a}) = a_1 P_{i+1,j,k,l}^m + a_2 P_{i,j+1,k,l}^m + a_3 P_{i,j,k+1,l}^m + a_4 P_{i,j,k,l+1}^m. \quad (11)$$

Here the  $a_h$ ,  $h = 1, \dots, 4$  are the so-called  $T$ -coordinates of  $\mathbf{w}$ . That is, if  $\mathbf{w} = w_2 - w_1$  is a direction vector with  $(\beta_1, \beta_2, \beta_3, \beta_4)$  and  $(\gamma_1, \gamma_2, \gamma_3, \gamma_4)$  being the barycentric coordinates of  $w_1$  and  $w_2$  with respect to  $T$ , the  $T$ -coordinates of  $\mathbf{w}$  are  $\mathbf{a} = (\gamma_1 - \beta_1, \gamma_2 - \beta_2, \gamma_3 - \beta_3, \gamma_4 - \beta_4) = (a_1, a_2, a_3, a_4)$ .

#### 4 The mixed directional difference-summation (MDDS) algorithm

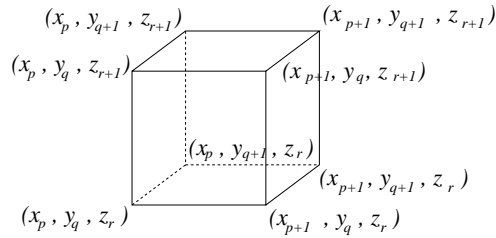
As mentioned in the introduction, algorithms for evaluating Box-spline functions can be classified into three types, depending on

- it is computed the value of the Box-spline function exactly for each fixed point using the recurrence definition in (1);
- it is given an efficient approximation scheme which is based on another recurrence relation, obtained via two-scale subdivision or inverse FFT;
- it is given an explicit representation, say in terms of the Bézier net, of each polynomial piece.

Algorithms of the third type depend on yet another form of recurrence relation (11). In this section we represent a trivariate Box-spline basis function over a regular tetrahedron partition  $\Delta$  of its polygonal domain  $\Omega \in \mathbb{R}^3$ , with piecewise planar boundary  $\partial\Omega$ , and we give a computational scheme to determine the Bézier coefficients of each polynomial piece of the trivariate function.

To be more precise, trivariate Box-spline basis functions  $M_{\nu_1\nu_2\nu_3\nu_4}^m$ ,  $\nu_1 + \nu_2 + \nu_3 + \nu_4 = n$ , are piecewise polynomials in three variables that consist of  $6 * (\nu_1 + \nu_4)(\nu_2 + \nu_4)(\nu_3 + \nu_4)$  different polynomial pieces of the same degree  $m = n - 3$ , defined over a tetrahedral tessellation of their domain.

Let  $x_0 < \dots < x_p < x_{p+1} < \dots < x_{\nu_1+\nu_4-1}$ ,  $y_0 < \dots < y_q < y_{q+1} < \dots < y_{\nu_2+\nu_4-1}$ ,  $z_0 < \dots < z_r < z_{r+1} < \dots < z_{\nu_3+\nu_4-1}$  and let the planes  $x = x_p$ ,  $y = y_q$ ,  $z = z_r$ ,  $p = 0, \dots, \nu_1 + \nu_4 - 1$ ,  $q = 0, \dots, \nu_2 + \nu_4 - 1$ ,  $r = 0, \dots, \nu_3 + \nu_4 - 1$  partition the 3D space into cubic cells. Drawing in all cells, the tetrahedral partition described in figure 3, we get a tetrahedral tessellation of the Box-spline domain.



Consider the cube of vertices  $(x_p, y_q, z_r), (x_p, y_q, z_{r+1}), (x_p, y_{q+1}, z_r), (x_p, y_{q+1}, z_{r+1}), (x_{p+1}, y_q, z_r), (x_{p+1}, y_q, z_{r+1}), (x_{p+1}, y_{q+1}, z_r), (x_{p+1}, y_{q+1}, z_{r+1})$  and denote by  $P_{i,j,k,l}^m$ ,  $i + j + k + l = m$ , the Bézier coefficients of the degree- $m$  trivariate polynomial  $p(\mathbf{v})$ , defined by restricting  $M_{\nu_1\nu_2\nu_3\nu_4}$ ,  $\nu_1 + \nu_2 + \nu_3 + \nu_4 = m + 3$ , to one of the six tetrahedra in the cube.

Hence it trivially follows

$$D_{\mathbf{d}_h} M_{D_n}(\mathbf{x}) = \sum_{\substack{i,j,k,l \geq 0 \\ i+j+k+l = m-1}} Q_{i,j,k,l}^{m-1} B_{i,j,k,l}^{m-1}(\lambda)$$

where  $\lambda \equiv (\lambda_0, \lambda_1, \lambda_2, \lambda_3)$  are the barycentric coordinates associated with  $\mathbf{x} \in T$ . Now recalling that a Box-spline is a linear combination of Bézier basis functions, we can exploit (10)-(11) and write

$$m \sum_{\substack{i,j,k,l \geq 0 \\ i+j+k+l = m-1}} \Delta P_{i,j,k,l}^{m-1}(\mathbf{a}) B_{i,j,k,l}^{m-1}(\lambda) = \sum_{\substack{i,j,k,l \geq 0 \\ i+j+k+l = m-1}} Q_{i,j,k,l}^{m-1} B_{i,j,k,l}^{m-1}(\lambda). \quad (12)$$

From (12) it can be easily derived

$$Q_{i,j,k,l}^{m-1} = m(a_1 P_{i+1,j,k,l}^m + a_2 P_{i,j+1,k,l}^m + a_3 P_{i,j,k+1,l}^m + a_4 P_{i,j,k,l+1}^m) \quad (13)$$

where  $\mathbf{a} \equiv (a_1, a_2, a_3, a_4)$  is a vector with the T-coordinates of the direction  $\mathbf{d}_h = \mathbf{x} - (\mathbf{x} - \mathbf{d}_h)$ :

$$a_i = \gamma_i - \beta_i \quad i = 1, \dots, 4,$$

with  $(\gamma_1, \gamma_2, \gamma_3, \gamma_4)$  and  $(\beta_1, \beta_2, \beta_3, \beta_4)$  being the barycentric coordinates of  $\mathbf{x}$  and  $\mathbf{x} - \mathbf{d}_h$  respectively<sup>1</sup>. The construction of the B-net requires to work out the unknowns  $P_{i,j,k,l}^m$  from equation (13). From now on, the factor  $m$  is omitted in order to keep the numbers integral (see Prop.7).

In conclusion, the mathematical ideas and formulations of finding the B-nets of a trivariate four-direction Box-spline can be summerized as follows.

Starting with the B-net of the degree-1 Box-spline  $M_{1111}^1$ , (see figure 6) we can compute the B-net of any other Box-spline  $M_{\nu_1\nu_2\nu_3\nu_4}^m$ , with  $\nu_1 + \nu_2 + \nu_3 + \nu_4 > 4$ , by applying the algorithm summarized in the following steps:

- (1) Express each polynomial piece of the Box-spline in (unknown) Bézier representation;
- (2) Use the recurrence (see [29])

<sup>1</sup> Since we always integrate along the directions  $e_1, e_2, e_3, e_{123}$ , generated by tetrahedra, we only have two non-zero barycentric coordinates which typically assume values 1 and -1.

$$D_{\mathbf{d}_h} M_{D_n}^m(\mathbf{x}) = M_{D_n \setminus \{\mathbf{d}_h\}}^{m-1}(\mathbf{x}) - M_{D_n \setminus \{\mathbf{d}_h\}}^{m-1}(\mathbf{x} - \mathbf{d}_h) \quad h = 1, \dots, n \quad (14)$$

to represent the degree- $m$  Box-spline with respect to the degree- $(m-1)$  one generated by the direction matrix  $D_n \setminus \{\mathbf{d}_h\}$ , with  $\text{span}(D_n \setminus \{\mathbf{d}_h\}) = \mathbb{R}^3$ ;

- (3) Determine the coefficients  $P^m$  in the B-net of  $M_{D_n}^m$  by using the derivative formula of TB-polynomials in (11), with  $(a_1, a_2, a_3, a_4)$  being the T-coordinates of  $\mathbf{d}_h$  and  $Q^{m-1}$  the coefficients of the B-net of  $M_{D_n \setminus \{\mathbf{d}_h\}}^{m-1}$ .

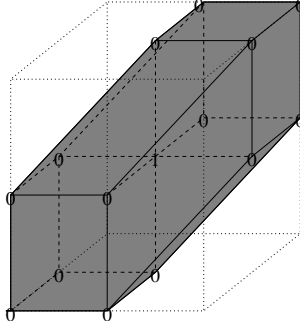


Fig. 6. B-net of  $M_{1111}^1$ .

Thus, starting with the known B-net of  $M_{1111}^1$ , the key idea to construct the B-net of an arbitrary degree Box-spline, is applying the derivative formula (14) recursively. In this way the Bézier coefficients  $Q_{i,j,k,l}^{m-1}$  of  $D_{\mathbf{d}_h} M_{D_n}^m$  can be obtained by applying a shift and subtract procedure on the B-net of  $M_{D_n \setminus \{\mathbf{d}_h\}}^{m-1}(\mathbf{x})$ .

Note that the B-nets of all trivariate four-direction Box-splines generated by the procedure described above can be computed once for all. This derives from the following proposition.

**Proposition 7** *Let  $M_{D_n}^m$  be a trivariate four-direction Box-spline. Then the B-net of its polynomial pieces consists of rational numbers.*

**Proof.** This trivially derives from the fact that we start our method from the linear B-net with all integral coefficients and we combine the  $\frac{1}{m}$  factors, coming from (13), into a single  $\frac{1}{m!}$  which can be carried out at the end of a sequence of operations which are indeed only integer additions and subtractions.  $\square$

**Corollary 8** *Thus we can compute the B-net of those Box-splines in exact arithmetic. This guarantees the numerical stability property of the proposed algorithm.*

To clarify the presentation, we first illustrate the procedure above with the following example and successively we present a Matlab package for computing the B-net coefficients of an arbitrary trivariate four-direction Box-spline (see Section 6).

**Example.** Consider the degree-2 Box-spline defined by the direction matrix  $D_5 = \{e_1, e_1, e_2, e_3, e_{123}\}$ . Following the algorithm given above, the B-net of  $M_{D_5}^2$  turns out to be defined by shifting and subtracting the B-net of  $M_{D_4}^1$  (see figure 6), with  $D_4 \equiv E_3 = \{e_1, e_2, e_3, e_{123}\}$ , along the direction  $e_1$  (see figure 7) and integrating

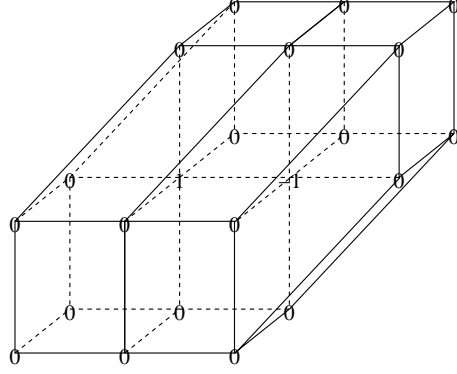


Fig. 7. Shift and subtraction of the coefficients of the linear Box-spline  $M_{D_4}^1$ .

the resulting coefficients along the same direction. This last step (which allows us to work out the Bézier coefficients of  $M_{D_5}^2$ ) is expressed in relation (13) with  $(a_1, a_2, a_3, a_4)$  being the T-coordinates of  $e_1$ , that is

$$\begin{aligned}
 a_1 &= \gamma_1 - \beta_1 = 0 - 1 = -1 \\
 a_2 &= \gamma_2 - \beta_2 = 1 - 0 = 1 \\
 a_3 &= \gamma_3 - \beta_3 = 0 - 0 = 0 \\
 a_4 &= \gamma_4 - \beta_4 = 0 - 0 = 0.
 \end{aligned}$$

In other words, relation (13) results simplified as follows:

$$\begin{aligned}
 Q_{1000}^1 &= P_{1100}^2 - P_{2000}^2 \\
 Q_{0100}^1 &= P_{0200}^2 - P_{1100}^2 \\
 Q_{0010}^1 &= P_{0110}^2 - P_{1010}^2 \\
 Q_{0001}^1 &= P_{0101}^2 - P_{1001}^2
 \end{aligned} \tag{15}$$

and thus

$$\begin{aligned}
 P_{1100}^2 &= Q_{1000}^1 + P_{2000}^2 \\
 P_{0200}^2 &= Q_{0100}^1 + P_{1100}^2 \\
 P_{0110}^2 &= Q_{0010}^1 + P_{1010}^2 \\
 P_{0101}^2 &= Q_{0001}^1 + P_{1001}^2.
 \end{aligned} \tag{16}$$

Since the integral of a TB-volume of degree 1 is a TB-volume of degree 2, equation (16) can be viewed as an integration of the  $Q^1$  coefficients, starting along the triangular face of the tetrahedron with vertices  $P_{2000}^2, P_{1010}^2, P_{1001}^2, P_{0020}^2, P_{0011}^2, P_{0002}^2$ . However, since the value of such constants of integration is unknown, we make them assume the coefficient values in the adjacent tetrahedron sharing the face containing those constants, until we get the box boundary, where the coefficients are equal to zero (as we are looking for basis functions with a bounded non-zero region). In this way we can compute the expression for the remaining four coefficients  $P_{1100}^2, P_{0200}^2,$

$P_{0110}^2, P_{0101}^2$  by a sequence of successive additions, like:

$$\begin{aligned} P_{1100}^2 &= Q_{1000}^1 + P_{2000}^2 \\ P_{0200}^2 &= Q_{0100}^1 + Q_{1000}^1 + P_{2000}^2 \\ P_{0110}^2 &= Q_{0010}^1 + P_{1010}^2 \\ P_{0101}^2 &= Q_{0001}^1 + P_{1001}^2. \end{aligned}$$

Applying these equalities to the whole Box-spline  $M_{D_5}^2$  and exploiting adjacencies of tetrahedra, we are capable to work out all the Bézier coefficients of its B-net. From now on we will use the double apex 2,  $(c_1, c_2)$  to denote the Bézier coefficients of the  $c_2$ -th degree-2 tetrahedral volume belonging to the cube  $c_1$ . For example, in order to compute the coefficient  $P_{0101}^{2,(1,5)}$  of the 5-th tetrahedron in cube 1 (the one on the right in figure 8), we exploit the relation

$$P_{0101}^{2,(1,5)} = Q_{0001}^{1,(1,5)} + P_{1001}^{2,(1,5)}$$

where  $P_{1001}^{2,(1,5)}$  coincides with the coefficient  $P_{0101}^{2,(2,1)}$  of the 1-st tetrahedron in the 2-nd cube (the one on the left in figure 8).

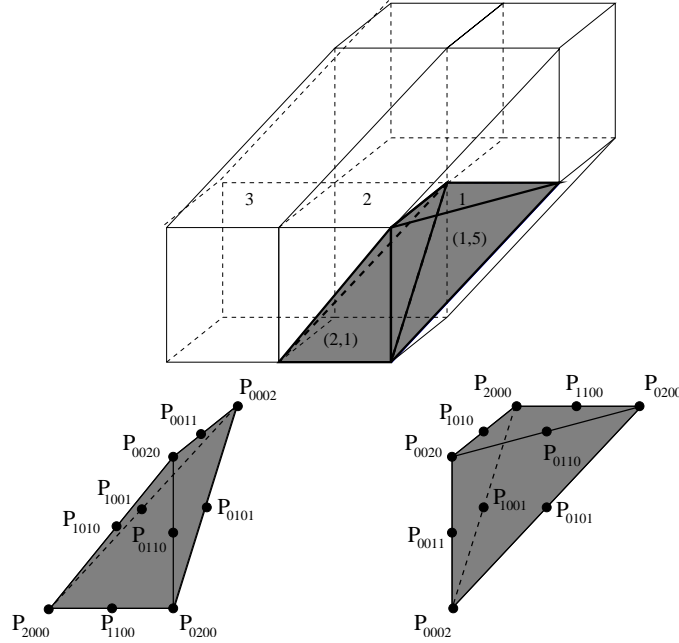


Fig. 8.

Therefore

$$P_{0101}^{2,(1,5)} = Q_{0001}^{1,(1,5)} + Q_{0001}^{1,(2,1)} + P_{1001}^{2,(2,1)}$$

and going on substituting until integration constant reaches the box boundary, we get

$$P_{0101}^{2,(1,5)} = Q_{0001}^{1,(1,5)} + Q_{0001}^{1,(2,1)} + Q_{0001}^{1,(2,5)} + Q_{0001}^{1,(3,1)} + Q_{0001}^{1,(3,5)} + P_{1001}^{2,(3,5)}. \quad (17)$$

In this way all the intermediate coefficients are easily computed:

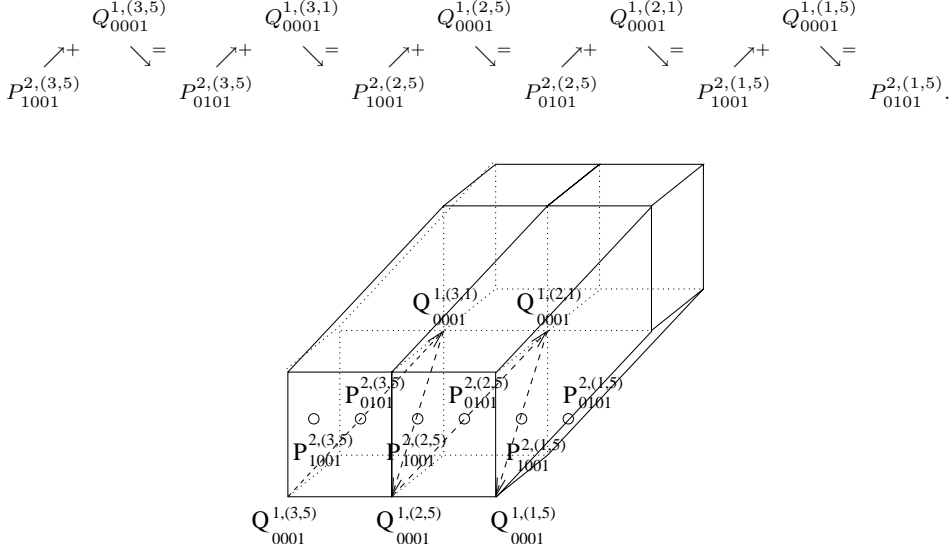


Fig. 9. The mixed direction worked out for determining the Bézier coefficients  $P_{1001}^{2,(3,5)}$ ,  $P_{0101}^{2,(3,5)}$ ,  $P_{1001}^{2,(2,5)}$ ,  $P_{0101}^{2,(2,5)}$ ,  $P_{1001}^{2,(1,5)}$ ,  $P_{0101}^{2,(1,5)}$  of  $M_{2111}^2$ .

The coefficients  $Q_{i,j,k,l}^1$  in (17) identify a particular mixed direction in  $\mathbb{R}^3$  (see figure 9). Such direction allows us to store the  $Q_{i,j,k,l}^1$  coefficients in a vector that has to be integrated by applying a sequence of summations (that starts with the integration constant  $P_{1001}^{2,(3,5)}$ ), in order to generate the  $P_{0101}^{2,(1,5)}$  coefficient of the degree-2 Box-spline  $M_{2111}^2$ . The name "mixed direction" derives from the fact that the direction of integration, identified by the coefficients  $Q_{i,j,k,l}^1$ , is a composition of some linear combinations of the directions  $e_1$ ,  $e_2$ ,  $e_3$ ,  $e_{123}$  ( $e_{123}$ ,  $e_2 + e_3$  in this case). The same procedure can be applied for determining all the mixed directions we need to work out all the B-net coefficients of  $M_{2111}^2$ . Obviously, the mixed directions by which they are determined are not always the same, although there is a common pattern for all of them. In figure 10 you can see the mixed directions set originating the B-net of  $M_{2111}^2$ .

The number and path of such directions depend on the degree of the Box. As you can see, the mixed directions set is composed of one uni-dimensional direction (figure 10-a), two two-dimensional directions (with two branches each), respectively on the planes  $y = \text{constant}$  (figure 10-b) and  $z = \text{constant}$  (figure 10-c), and one three-dimensional direction (figure 10-d) composed of four branches. Each branch is just a translation in the vertices indicated in the pictures, of the given mixed direction with origin in the vertex marked by 0. The uni-dimensional one, defined by a single straight direction along  $e_1$ , corresponds to the direction  $\mathbf{d}_h$  along which

we are integrating. For this reason we are going to refer to such direction by the term *main direction*. In the given example, such direction identifies the coefficient  $P_{0200}^{2,(1,5)}$ , since its value is computed in the following way:

$$\begin{array}{cccccccc}
 & Q_{0001}^{1,(3,5)} & & Q_{0100}^{1,(3,5)} & & Q_{1000}^{1,(2,5)} & & Q_{0100}^{1,(2,5)} & & Q_{1000}^{1,(1,5)} & & Q_{0100}^{1,(1,5)} \\
 & \nearrow + & \searrow = & \nearrow + & \searrow = & \nearrow + & \searrow = & \nearrow + & \searrow = & \nearrow + & \searrow = & \nearrow + & \searrow = \\
 P_{2000}^{2,(3,5)} & & P_{1100}^{2,(3,5)} & & P_{2000}^{2,(2,5)} & & P_{1100}^{2,(2,5)} & & P_{2000}^{2,(1,5)} & & P_{1100}^{2,(1,5)} & & P_{0200}^{2,(1,5)}
 \end{array}$$

As it can be observed, because  $Q_{0100}^{1,(3,5)} = Q_{1000}^{1,(2,5)}$  and  $Q_{0100}^{1,(2,5)} = Q_{1000}^{1,(1,5)}$ , the main direction contains two duplicated coefficients. In general, a main direction for generating a degree- $m$  Box-spline, contains exactly  $m$  duplicated coefficients.

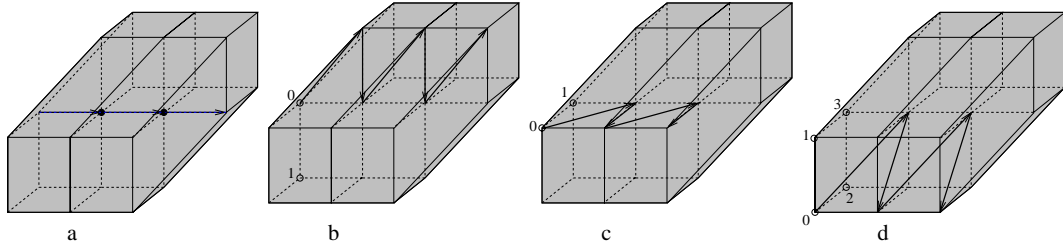


Fig. 10. All the integration paths (mixed directions) sufficient for computing the B-net of the quadratic trivariate Box-spline  $M_{2111}^2$ .

## 5 Restricting the scheme to the two-dimensional case

In this section we are going to show that a restriction of our computational scheme to the two-dimensional case provides an alternative procedure to the computational scheme proposed by Lai [24], Chui and Lai [6], [7], [8], for computing the Bézier net of bivariate three-direction Box-splines. In fact following the Box-spline definition proposed in [29] we can use mathematical induction on the degree  $m$  to prove that

$$M^m(x, y, z) \Big|_{plane} = M^m(u, v) \quad \forall m \geq 1$$

and in particular

$$\begin{aligned}
 M_{\nu_1 \nu_2 \nu_3}^{\nu_1 + \nu_3 - 1}(x, y, z) \Big|_{y=1} &= M_{\nu_1 \nu_3}^{\nu_1 + \nu_3 - 1}(x, z) \\
 M_{\nu_1 \nu_2 \nu_3}^{\nu_1 + \nu_2 - 1}(x, y, z) \Big|_{z=1} &= M_{\nu_1 \nu_2}^{\nu_1 + \nu_2 - 1}(x, y) \\
 M_{\nu_1 \nu_2 \nu_3}^{\nu_2 + \nu_3 - 1}(x, y, z) \Big|_{x=1} &= M_{\nu_2 \nu_3}^{\nu_2 + \nu_3 - 1}(y, z)
 \end{aligned}$$

$$M_{11\nu_3\nu_4}^{\nu_3+\nu_4-1}(x, y, z) \Big|_{x=y} = M_{1\nu_3\nu_4}^{\nu_3+\nu_4-1}(x, z).$$

We now confine ourselves to the first case and for sake of simplicity we choose  $\nu_3 = 1$ , focusing our attention on the restriction of the trivariate Box-spline  $M_{\nu_1 111}^{\nu_1}$  to the plane  $y = 1$ . The initial step is the linear case:

$$M_{E_3}^1(x, y, z) \Big|_{y=1} = M_{E_2}^1(x, z).$$

According to the recurrence Box-spline definition given in Section 2,

$$\begin{aligned} M_{E_3}^1(x, y, z) &= t_1 M_{E_3 \setminus \{e_1\}}^0(x, y, z) + (1 - t_1) M_{E_3 \setminus \{e_1\}}^0(x - 1, y, z) \\ &\quad + t_2 M_{E_3 \setminus \{e_2\}}^0(x, y, z) + (1 - t_2) M_{E_3 \setminus \{e_2\}}^0(x, y - 1, z) \\ &\quad + t_3 M_{E_3 \setminus \{e_3\}}^0(x, y, z) + (1 - t_3) M_{E_3 \setminus \{e_3\}}^0(x, y, z - 1) \\ &\quad + t_4 M_{E_3 \setminus \{e_{123}\}}^0(x, y, z) + (1 - t_4) M_{E_3 \setminus \{e_{123}\}}^0(x - 1, y - 1, z - 1). \end{aligned} \quad (18)$$

Figure 11 illustrates the computation in (18).

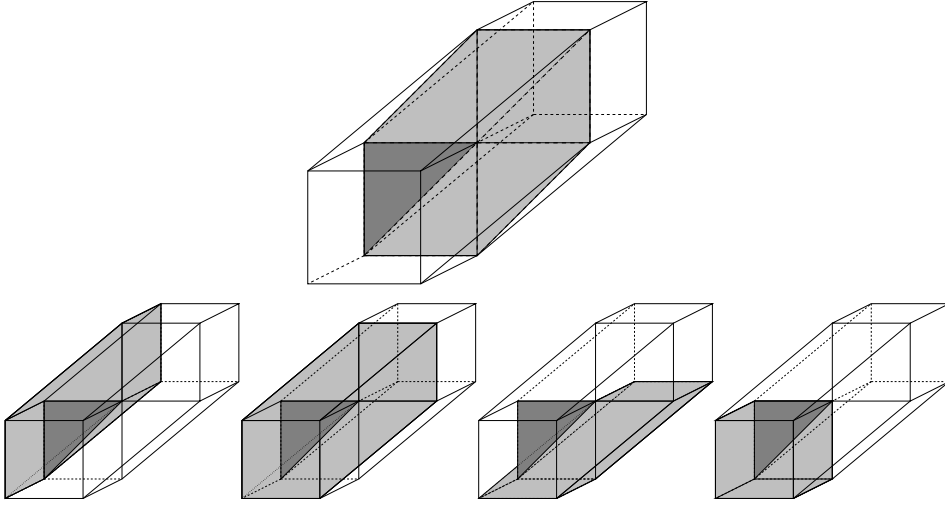


Fig. 11. The four branches related to  $t_1, t_2, t_3, t_4$  contributions.

Further, one notes that the only branches of the tree which contribute to the computation are those for which  $(x, y, z)$  lies in the support of the corresponding shifted Box-spline. In this case  $(x, y, z)$  is chosen in the area delimited by the gray hexagon (figure 11). To compare corresponding trivariate and bivariate Box-splines, we have to consider all the six triangles contained in the hexagonal marked area. In this paper we show, for example, the computation on one of them. In particular if  $(x, y, z)$  lies in the darkest triangle,  $M_{E_3}^1(x, y, z) = t_1 + 1 - t_2$ .



Applying relation (1) we get  $M_{E_3}^1(x, y, z) \Big|_{y=1} = x$ .

In a similar way we have evaluated the bivariate Box-spline in the same triangle, obtaining

$$M_{E_2}^1(x, z) = x.$$

Exploiting the inductive definition suggested in [29] we have

$$\begin{aligned} M_{E_3 \cup \{e_1\}}^2(x, y, z) \Big|_{y=1} &= \int_0^1 M_{E_3}^1(x-t, y, z) dt \Big|_{y=1} \\ &= \int_0^1 M_{E_3}^1(x-t, y, z) \Big|_{y=1} dt \quad (*) \end{aligned}$$

and for the inductive hypothesis  $M_{E_3}^1(x-t, 1, z) = M_{E_2}^1(x-t, z)$ ,

$$(*) = \int_0^1 M_{E_2}^1(x-t, z) dt = M_{E_2 \cup \{e_1\}}^2(x, z). \quad \square$$

From this property it easily derives that the B-net of a bivariate three-direction Box-spline  $M_{\nu_1, \nu_2, \nu_3}^m$  (with one of the multiplicities  $\nu_1, \nu_2, \nu_3$  set equal to 1) can be obtained by restriction of the Bézier coefficients of the corresponding trivariate four-direction Box-spline to a specified plane. For example the B-net of  $M_{211}^2$

$$\frac{1}{2} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

is exactly the restriction of the B-net of  $M_{2111}^2$  to the plane  $y = 1$ . This is explained by observing that in the bivariate case the mixed directions we use for generating the B-nets are exactly the 2D mixed directions of the corresponding trivariate Box-spline, belonging to the section plane.

## 6 The MDDS package

This section is concerned with the description of a Matlab package, called MDDS (available in the NUMERALGO library [4]), designed to compute the B-net coefficients of all trivariate four-direction Box-splines and additionally to exact evaluate and visualize them.

The user interface is provided by three main functions:

1. the function `bnet_mdds()` that allows to compute the Bézier net coefficients of trivariate four-direction Box-splines of any degree;
2. the function `pointeval_mdds3d()` that allows to exact evaluate any trivariate four-direction Box-spline in an arbitrary set of 3D points;
3. the function `visual_mdds3d()` that allows to visualize any trivariate four-direction Box-spline  $M_{\nu_1\nu_2\nu_3\nu_4}^m$  either through the  $s$ -set ( $s \in \mathcal{Imm}(M_{\nu_1\nu_2\nu_3\nu_4}^m)$ ) extraction of the Box-spline volume or through the contour lines of the regions obtained by intersecting the Box-spline volume with three families of planes respectively parallel to  $yz$ ,  $xz$ ,  $xy$ .

The function `bnet_mdds()` implements the algorithm presented in Section 4 for generating the mixed directions set of  $M_{\nu_1\nu_2\nu_3\nu_4}^m$  and computing its Bézier net.

The general procedure proposed consists in representing the B-net of the linear Box-spline  $M_{1111}^1$  by the tri-dimensional array

$$B(:, :, 1) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B(:, :, 2) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B(:, :, 3) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

and generating the B-net of  $M_{\nu_1\nu_2\nu_3\nu_4}^m$  by successive differences-summations of the linear Box-spline along the four directions  $e_1, e_3, e_2, e_{123}$ , respectively  $\nu_1, \nu_3, \nu_2, \nu_4$  times:

$$M_{1111}^1 \xrightarrow{I} M_{\nu_1 111}^{\nu_1} \xrightarrow{II} M_{\nu_1 1 \nu_3 1}^{\nu_1 + \nu_3 - 1} \xrightarrow{III} M_{\nu_1 \nu_2 \nu_3 1}^{\nu_1 + \nu_2 + \nu_3 - 2} \xrightarrow{IV} M_{\nu_1 \nu_2 \nu_3 \nu_4}^m.$$

At the end of these four steps, the Bézier coefficients of  $M_{\nu_1\nu_2\nu_3\nu_4}^m$  are stored in the tri-dimensional array  $A(iA, jA, kA)$ , where  $1 \leq iA \leq m(\nu_3 + \nu_4) + 1$ ,  $1 \leq jA \leq m(\nu_1 + \nu_4) + 1$ ,  $1 \leq kA \leq m(\nu_2 + \nu_4) + 1$  (noting that we have zero coefficients outside of the support of  $M_{\nu_1\nu_2\nu_3\nu_4}^m$ ). Steps I, II, III, IV are summarized respectively in the four function modules `step1_mdds3d()`, `step2_mdds3d()`, `step3_mdds3d()`, `step4_mdds3d()`. They are devised to generate the mixed directions set MD needed to work out the intermediate Bézier coefficients set  $A$ , starting with the one degree less coefficients matrix  $B = A(\cdot | D_n \setminus \{\mathbf{d}_h\}) - A(\cdot - \mathbf{d}_h | D_n \setminus \{\mathbf{d}_h\})$ .

I. First step:

$$M_{1111} \longrightarrow M_{\nu_1 111}$$

**for**  $i=1, \dots, \nu_1$

A=step1\_mdds3d(i, 1, 1, 1, B)

B=A

**end**

Number of MD:  $m^2$

Number of branches:  $(2\nu_1 - 1)^2$

II. Second step:

$$M_{\nu_1 111} \longrightarrow M_{\nu_1 1 \nu_3 1}$$

**for**  $i=1, \dots, \nu_3$

A=step2\_mdds3d( $\nu_1, 1, i, 1, B$ )

B=A

**end**

Number of MD:  $m^2$

Number of branches:  $(2m - 1)(m(\nu_1 + 1) - 1)$

III. Third step:

$$M_{\nu_1 1 \nu_3 1} \longrightarrow M_{\nu_1 \nu_2 \nu_3 1}$$

**for**  $i=1, \dots, \nu_2$

A=step3\_mdds3d( $\nu_1, i, \nu_3, 1, B$ )

B=A

**end**

Number of MD:  $m^2$

Number of branches:  $(m(\nu_3 + 1) - 1)(m(\nu_1 + 1) - 1)$

IV. Fourth step:

$$M_{\nu_1 \nu_2 \nu_3 1} \longrightarrow M_{\nu_1 \nu_2 \nu_3 \nu_4}$$

**for**  $i=1, \dots, \nu_4$

A=step4\_mdds3d( $\nu_1, \nu_2, \nu_3, i, B$ )

B=A

**end**

Number of MD:  $3m(m - 1) + 1$

Number of branches:  $m^2(\nu_1 \nu_2 + \nu_2 \nu_3 + \nu_1 \nu_3) - m(\nu_1 + \nu_2 + \nu_3) + 1$

To perform the calculations in the four steps above, two additional modules named `dup1_mdds()` and `integ_mdds()`, are needed. While the first one is required for the

computation of the duplicated coefficients in the *main mixed directions* (as previously explained at the end of Section 4), the second one is required to compute the integration step along any direction.

To exact evaluate any trivariate four-direction Box-spline in an arbitrary set of 3D points, an additional function, named `pointeval_mdds3d()`, is provided. This function is supported by the function modules `findcube_mdds3d()` and `findtetra_mdds3d()`, which respectively identify the unit cube and the tetrahedron type (see figure 3) that contain the evaluation point.

If, instead, there is an interest to visualize a trivariate four-direction Box-spline, the function `visual_mdds3d()` will make the job. Exploiting the trivariate Bernstein-Bézier polynomials evaluation procedure in `computebez_mdds3d()`, it allows to visualize a  $s$ -set extraction (with  $s \in \mathcal{Imm}(M_{\nu_1\nu_2\nu_3\nu_4}^m)$ ) of the Box-spline volume  $M_{\nu_1\nu_2\nu_3\nu_4}^m$  and (if required) its contour lines.

To demonstrate the use of all such functions, the M-file `demo_mdds3d.m` is included. Three examples of four-direction Box-splines on  $\mathbb{R}^3$  are examined: the cubic Box-spline  $M_{2112}^3$ , the quintic Box-spline  $M_{2222}^5$  and the sextic Box-spline  $M_{3222}^6$ . Their Bézier coefficients matrix and their evaluation in an arbitrary point of the domain are provided. Furthermore a visualization through the  $s$ -set extraction and the three families of contour lines is available. We show here the 0-set and 0.01-set extraction of the Box-spline volume  $M_{2112}^3$  (figure 12), the 0-set and the 0.005-set extraction of  $M_{2222}^5$  (figure 13) and the 0-set and the 0.001-set extraction of  $M_{3222}^6$  (figure 14).

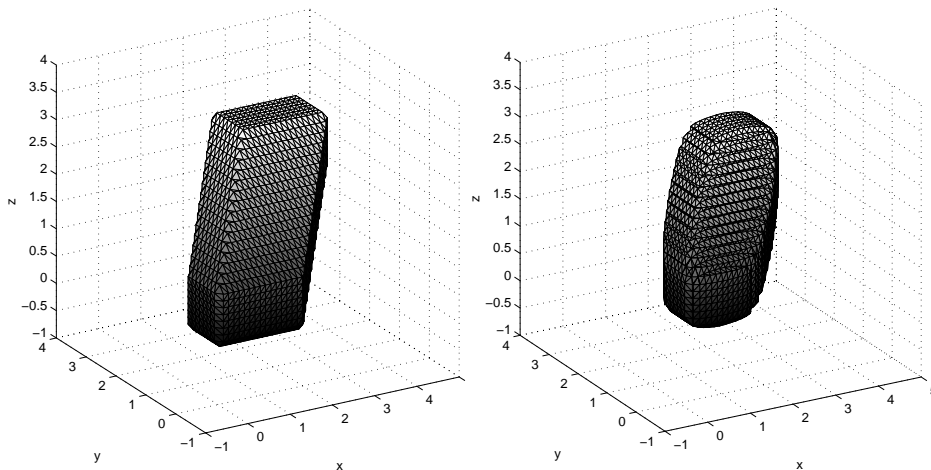


Fig. 12. Visualization through 0-set extraction (left) and 0.01-set extraction (right) of the Box-spline volume  $M_{2112}^3$ .

Figures 15 and 16 illustrate, instead, the mixed-directions set computed for generating the Bézier net of  $M_{2112}^3$ .

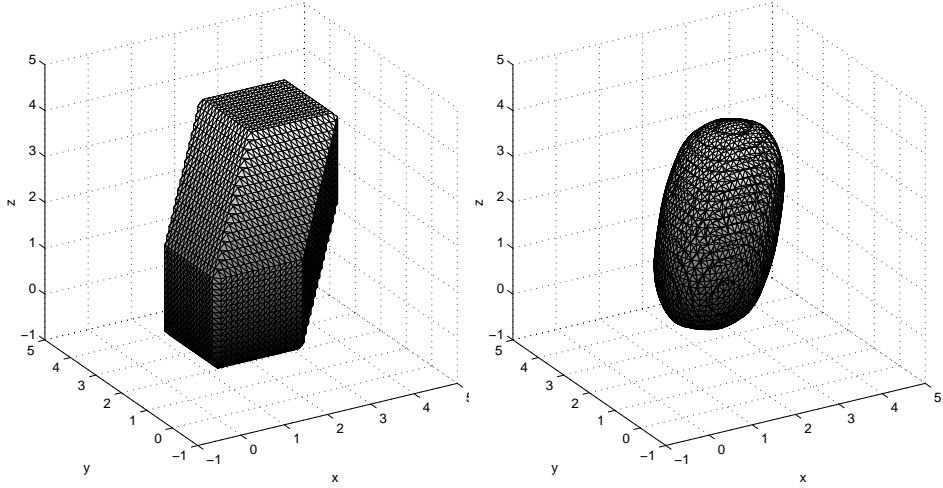


Fig. 13. Visualization through 0-set extraction (left) and 0.005-set extraction (right) of the Box-spline volume  $M_{2222}^5$ .

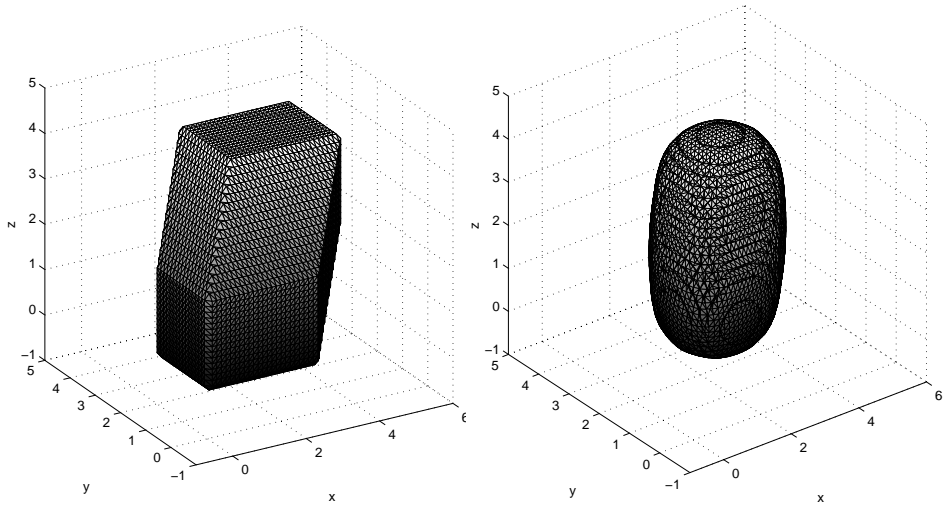


Fig. 14. Visualization through 0-set extraction (left) and 0.001-set extraction (right) of the Box-spline volume  $M_{3222}^6$ .

Indeed, through our mixed directional difference-summation algorithm, we can obtain also the B-net of any arbitrary bivariate three-direction Box-spline (see Section 5). In fact, also for computing the B-net of the very general Box-spline  $M_{\nu_1, \nu_2, \nu_3}^m$  we can exploit the idea proposed for generating the Bézier coefficients of an arbitrary trivariate Box-spline and work out a similar 2D mixed directions set. Such a procedure, alternative to the computational scheme proposed by Lai [24], Chui and Lai [6],[7],[8], instead of computing the complete B-net of a bivariate three-direction Box-spline by starting at the upstream edge and integrating successively triangle by triangle, identifies special mixed directions on the support region which allow to work out the integration step in the univariate case. Since calculations are reduced to univariate calculations along paths, our algorithm is shown to compare favorably over the other existing algorithms. For this reason in the MDDS package we have

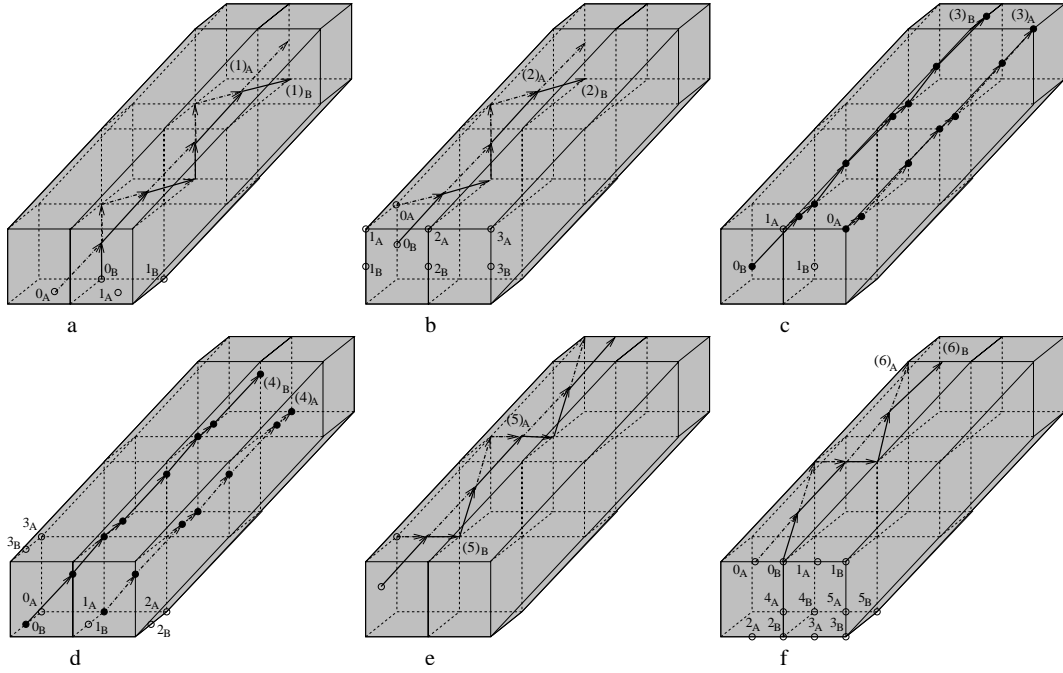


Fig. 15. The 2D mixed-directions set used for computing the B-net of the cubic trivariate Box-spline  $M_{2112}^3$ .

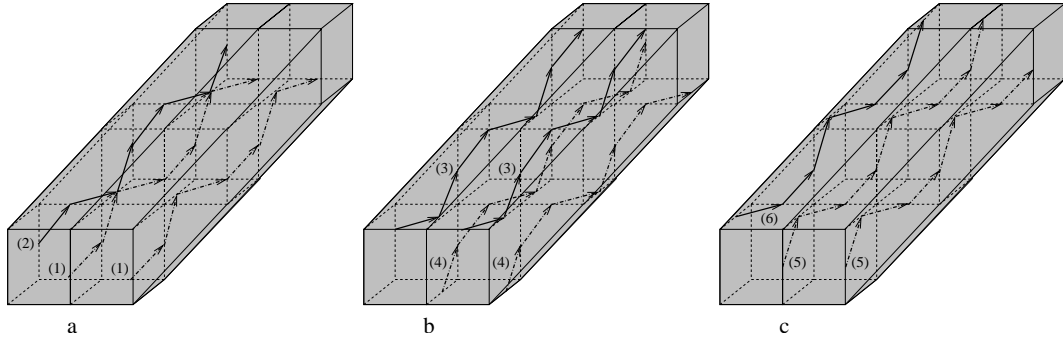


Fig. 16. The 3D mixed-directions set used for computing the B-net of the cubic trivariate Box-spline  $M_{2112}^3$ .

added a Matlab code [4] that exploits the mixed directions set procedure to compute the B-net coefficients of all bivariate three-direction Box-splines and additionally to exact evaluate and visualize them.

More precisely, the computation of the Bézier net coefficients of bivariate three-direction Box-splines of any degree, is performed by the main function `bnet_mdds()` itself. In fact, if the input parameter is a three-component vector with the multiplicities of the directions in  $E_2$ , the three function modules `step1_mdds2d()`, `step2_mdds2d()`, `step3_mdds2d()` are called in order to generate the mixed directions set MD in  $\mathbb{R}^2$ .

Conversely, the exact evaluation of any bivariate three-direction Box-spline in an

arbitrary set of 2D points, is implemented in the function `pointeval_mdds2d()`, by means of the functions `findsquare_mdds2d()` and `findtria_mdds2d()`, where the unit square and the triangle type (see figure 2, right) containing the evaluation point are respectively identified. Finally, the visualization of a bivariate three-direction Box-spline and its B-net, is carried out in the function `visual_mdds2d()`, which exploits the function `computebez_mdds2d()` for the evaluation of bivariate Bernstein-Bézier polynomials. The use of all such functions is demonstrated in the M-file `demo_mdds2d.m`, where the quadratic Box-spline  $M_{211}^2$ , the quartic Box-spline  $M_{222}^4$  and the quintic Box-spline  $M_{322}^5$  are taken as examples for the computation of the Bézier coefficients matrix and the evaluation in an arbitrary point of the domain. Furthermore a visualization of these Box-spline surfaces and their B-nets is provided, as shown in figures 17, 18 and 19. Figure 20, instead, illustrates the mixed directions set used to generate the B-net of  $M_{222}^4$ .

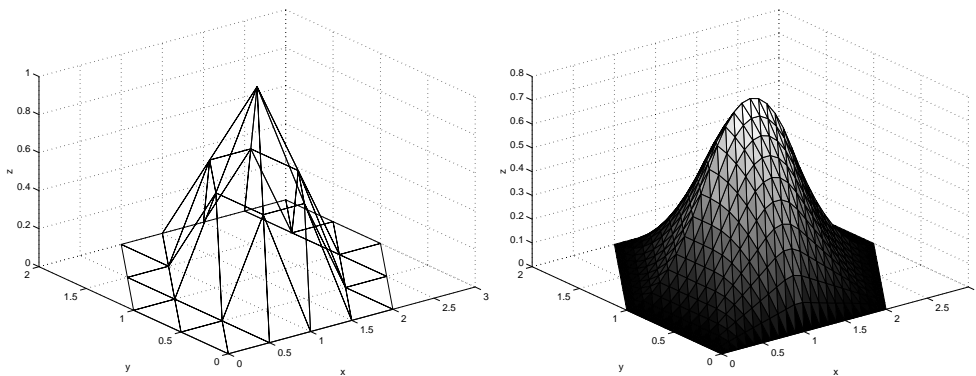


Fig. 17. B-net (left) and surface graph (right) of the bivariate quadratic three-direction Box-spline  $M_{211}^2$ .

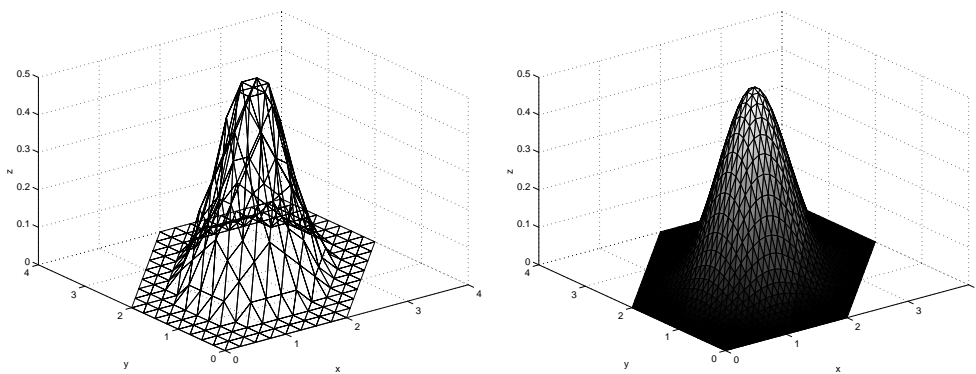


Fig. 18. B-net (left) and surface graph (right) of the bivariate quartic three-direction Box-spline  $M_{222}^4$ .

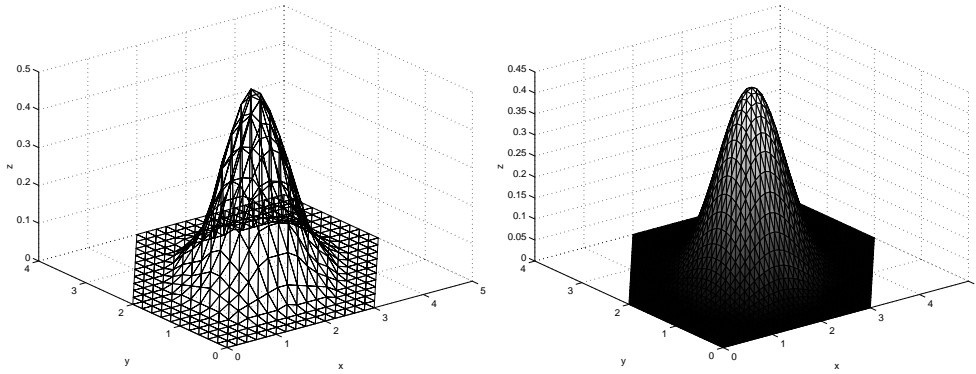


Fig. 19. B-net (left) and surface graph (right) of the bivariate quintic three-direction Box-spline  $M_{322}^5$ .

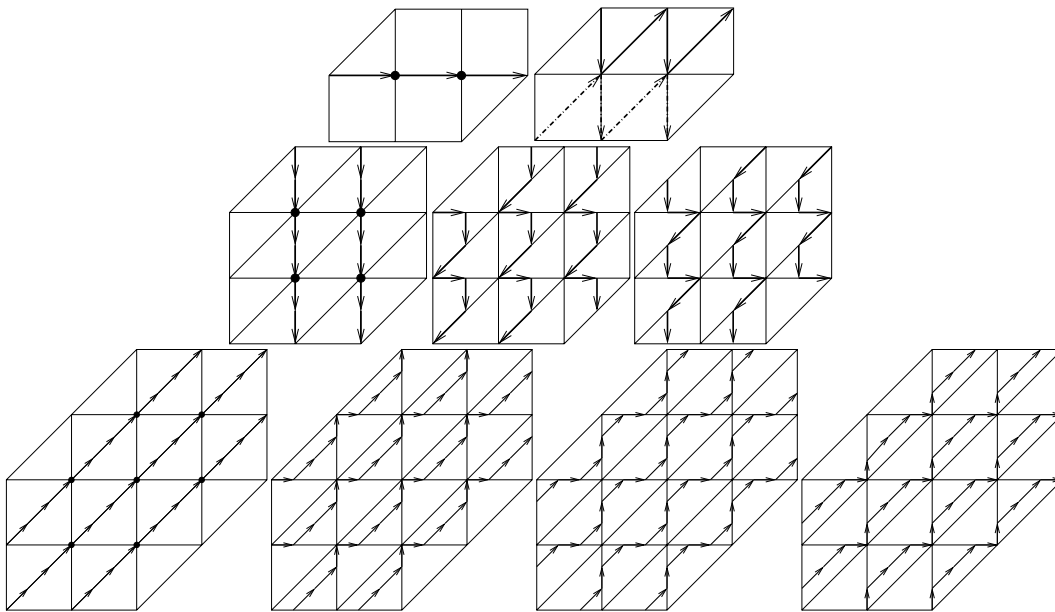


Fig. 20. Generation of the B-net of  $M_{222}^4$  by successive summations of the coefficients of  $D_{e_1}M_{111}^1$ ,  $D_{e_2}M_{211}^2$ ,  $D_{e_{12}}M_{221}^3$  along the marked mixed directions.

## 7 Conclusions

In this work we have proposed a new computational scheme for determining the Bézier net of any trivariate four-direction Box-spline and we have given a concise Matlab implementation of it [4]. The results obtained show that our algorithm is stable and efficient and, although reproducing consisting results with the existing evaluation algorithms [20], it has no comparison in the field of exact evaluation techniques.

In addition, if restricted to the two-dimensional case, it provides a novel and simplified procedure for computing the Bézier net coefficients of any bivariate three-direction Box-spline.



## Acknowledgements

This research was supported by MIUR-PRIN 2004 and by University of Bologna "Funds for selected research topics".

## References

- [1] T. Asahi, K. Ichige and R. Ishii, A new formulation for discrete box splines reducing computational cost and its evaluation, *IEICE Trans. Fundamentals*, Vol. E84-A, No. 3, March 2001.
- [2] J. Berchtold, I. Voiculescu and A. Bowyer, Multivariate Bernstein-form polynomials, Technical Report No. 31/98, School of Mechanical Engineering, University of Bath.
- [3] W. Boehm, Calculating with box splines, *Computer Aided Geometric Design* **1**(2) (1984), pp. 149-162.
- [4] G. Casciola, E. Franchini, L. Romani, The MDDS package, available at [www.netlib.org/numeralgo/na23](http://www.netlib.org/numeralgo/na23).
- [5] Y.-S. Chang, K.T. McDonnell and H. Qin, A new solid subdivision scheme based on box splines, *Proceedings of the 7th ACM Symposium on Solid Modeling and Applications* (2002), pp. 226-233.
- [6] C.K. Chui and M.J. Lai, Computation of box-splines and B-splines on triangulations of nonuniform rectangular partitions, *Approximation Theory & its Applications* **3** (1987), pp. 37-62.
- [7] C.K. Chui, *Multivariate Splines*, CBMS Lectures Series 54, SIAM, Philadelphia (1988).
- [8] C.K. Chui and M.J. Lai, Algorithms for generating B-nets and graphically displaying spline surfaces on three- and four- directional meshes, *Computer Aided Geometric Design* **8** (1991), pp. 479-493.
- [9] E. Cohen, T. Lyche and R. Riesenfeld, Discrete box splines and refinement algorithms, *Computer Aided Geometric Design* **1** (1984), pp. 131-141.
- [10] M. Dæhlen, On the evaluation of box splines, in: *Mathematical Methods in Computer Aided Geometric Design* T. Lyche and L. Schumaker (Eds.), Academic Press N.Y. (1989), pp. 167-179.
- [11] W. Dahmen and C.A. Micchelli, Recent progress in multivariate splines, in: *Approximation Theory IV* C.K. Chui, L.L. Schumaker and J.D. Ward (Eds.), Academic Press, New York (1983), pp. 27-121.

- [12] W. Dahmen, Bernstein-Bézier representation of polynomial surfaces, in: *Extension of B-spline curve algorithms to surfaces* Siggraph 86 Lecture Notes, ACM New York.
- [13] C. de Boor, R. De Vore, Approximation by smooth multivariate splines, *Trans. Amer. Math. Soc.* **276** (1983), pp. 775-788.
- [14] C. de Boor and K. Höllig, Recurrence relations for multivariate B-splines, *Proc. Amer. Math. Soc.* **85** (1982), pp. 397-400.
- [15] C. de Boor and K. Höllig, B-splines from parallelepipeds, *J. Analyse Math.* **42** (1982), pp. 99-115.
- [16] C. de Boor, B-form basics, in: *Geometric Modeling: Applications and New Trends* G. Farin (Ed.), SIAM Publication, Philadelphia (1987), pp. 131-148.
- [17] C. de Boor, On the evaluation of box splines, *Numerical Algorithms* **5** (1993), pp. 5-23.
- [18] C. de Boor and K. Höllig and S. Riemenschneider, *Box splines*, Springer Verlag, Berlin, 1993.
- [19] K. Höllig, Box splines, in: *Approximation Theory V* C.K. Chui, L.L. Schumaker and J.D. Ward (Eds.), Academic Press, New York (1986), pp. 71-95.
- [20] L. Kobbelt, Stable evaluation of box splines, *Numerical Algorithms* **14**(4) (1997), pp. 377-382.
- [21] D. Lasser, Bernstein-Bézier representation of volumes, *Computer Aided Geometric Design* **2**(1-3) (1985), pp. 145-150.
- [22] M.D. Mc Cool, Optimized evaluation of Box splines via the inverse FFT, *Proceedings of Graphics Interface*, Canadian Information Processing Society, Québec (1995), pp. 34-43.
- [23] T. Goodman and J. Peters, Bézier nets, convexity and subdivision on higher dimensional simplices, *Computer Aided Geometric Design* **12**(1) (1995), pp. 53-65.
- [24] M.J. Lai, Fortran subroutines for B-nets of box splines on three- and four-directional meshes, *Numerical Algorithms* **2** (1992), pp. 33-38.
- [25] J. Peters, Evaluation of multivariate Bernstein polynomials, CMS Technical Report No. 91-1, University of Wisconsin.
- [26] J. Peters, Evaluation and approximate evaluation of the multivariate Bernstein-Bézier form on a regularly partitioned simplex, *ACM Trans. Math. Softw.* **20**(4) (1994), pp. 460-480.
- [27] J. Peters,  $C^2$  surfaces built from zero sets of the 7-direction box spline, *Proceedings of the sixth IMA Conference on the Mathematics of Surfaces* (1994), pp. 463-474.

- [28] J. Peters and M. Wittman, Blending basic implicit shapes using trivariate box splines, Proceedings of the fourth ACM Symposium on Solid Modeling and Applications (1997), pp. 195-205.
- [29] H. Prautzsch and W. Boehm, Box Splines, in: *Handbook of Computer Aided Geometric Design* G. Farin, J. Hoschek and M.-S. Kim (Eds.), (2002), Chapter 10.
- [30] L.L. Schumaker and W. Volk, Efficient evaluation of multivariate polynomials, Computer Aided Geometric Design **3**(2) (1986), pp. 149-154.
- [31] J. Sun and K. Zhao, On the structure of Bézier nets, J. Computational Math. **5** (1987), pp. 376-383.