Coefficient–Based Spline Data Reduction by Hierarchical Spaces

Cesare Bracco, Carlotta Giannelli(✉), and Alessandra Sestini

Dipartimento di Matematica e Informatica "U. Dini", Università di Firenze, Viale Morgagni 67/A, 50134 Firenze, Italy
{cesare.bracco,carlotta.giannelli,alessandra.sestini}@unifi.it

Abstract. We present a data reduction scheme for efficient surface storage, by introducing a coefficient–based least squares spline operator that does not require any pointwise evaluation to approximate (in a lower dimension spline space) a given bivariate B–spline function. In order to define an accurate approximation of the target spline with a significant reduction of the space dimension, this operator is subsequently combined with the hierarchical spline framework to design an adaptive method that exploits the capabilities of truncated hierarchical B–splines (THB–splines). The resulting THB–spline simplification approach is validated by several numerical tests. The target B–spline surfaces include approximations of functions whose analytical expression is available, reconstructions of geographic data and parametric surfaces.

Keywords: Data reduction · Quasi–interpolation · Hierarchical splines · THB–splines

1 Introduction

A general data reduction scheme indicates any process that enables to store a certain set of information by (strongly) decreasing the amount of data needed for its reliable reconstruction. For example, an image compression algorithm represents a data reduction approach for images. A natural choice in this context relies in considering a *reference spline* representation that has to be previously generated in a suitably large spline space in order to guarantee a certain accuracy of the approximation. The data reduction scheme can then be applied to reduce the dimension of the spline space while preserving the quality of the approximation. Examples that consider an initial reference spline in the univariate case may be found in [3,17,29]. In these schemes the dimension reduction of the spline space was obtained through *simplification* of the reference spline by placing/removing the knots according to the shape of interest.

We here consider the problem of data reduction for efficient surface representation, see e.g., [20], by assuming an initial description of the target surface in standard tensor–product B–spline form. We then look for a new *spline data reduction* approach for surfaces that can also allow us to deal with complex shapes when extended to multi–patch B–spline descriptions. Obviously,

© Springer International Publishing AG 2017
M. Floater et al. (Eds.): MMCS 2016, LNCS 10521, pp. 23–41, 2017.
https://doi.org/10.1007/978-3-319-67885-6_2

when combined with a preliminary spline approximation phase, this kind of data reduction approach can also be applied to different surface representation formats, as for example gridded sets of space points that define geographic areas described by scanner acquisitions. In order to design a localized data reduction algorithm in a multivariate spline setting, different adaptive spline constructions may be considered. We mention T–splines [22], spline spaces over T–meshes [5] or locally refined (LR) box–partitions [7], as well as hierarchical splines [8]. In the bivariate context shape simplification with T–splines and polynomial splines over hierarchical T–meshes were discussed in [21] and [6], respectively. The use of LR B–splines for large data sets approximations was recently proposed [23].

Hierarchical B–splines were introduced as one of the first generalizations of tensor–product B–spline representations by considering a multilevel approach [8]. The idea of exploiting a multi–resolution spline scheme constitutes a powerful framework for data fitting with local refinements [9,14] and adaptive surface reconstruction [10,15]. The hierarchical levels are identified in terms of nested sequences of refined areas that define the domain hierarchy. A basis of hierarchical spline spaces may be easily constructed by selecting basis functions from different refinement levels according to the domain hierarchy [16]. By assuming mild conditions on the hierarchical mesh configuration, suitable choices of hierarchical B–spline bases span the entire space of piecewise polynomial functions of a certain degree and smoothness that are defined on the underlying grids, see e.g., [1,19]. A renewed interest in this kind of construction has been prompted by the introduction of the truncated basis for hierarchical splines [12]. Truncated hierarchical B–splines (THB–splines) slightly modify the selection mechanism for the hierarchical basis construction to recover the partition of unity property and reduce the influence of coarser basis functions in refined areas. Additional properties of the truncated basis have been derived by also considering a more general hierarchical setting, not necessarily restricted to the tensor–product B–spline model [13]. A relevant peculiarity of the truncated basis consists in facilitating the construction of hierarchical quasi–interpolants [25]. For example, a bivariate hierarchical Hermite quasi–interpolation scheme based on THB–splines was proposed in [2]. Additional results and examples within this approach were recently discussed [24].

By exploiting the truncated basis for hierarchical splines, we propose a data reduction approach by combining multilevel spline spaces with a coefficient–based operator applicable to spline functions. In particular our quasi–interpolant is based on a local least squares operator which uses only the de Boor coefficients of the target spline, and, consequently, no pointwise function evaluation is required. Its formulation in hierarchically refined spline spaces ensures a high level of data reduction, while simultaneously preserving the shape details of the given spline.

The structure of the paper is as follows. The coefficient–based spline operator is introduced in Sect. 2 while the construction and properties of (truncated) hierarchical B–splines are recalled in Sect. 3. Section 4 presents the THB–spline formulation of the new coefficient–based operator and the related spline simplification scheme. Finally, Sect. 5 provides several examples, including data reduction for functions whose analytical expression is available, geographic data approximation and geometric models, and Sect. 6 concludes the paper.

2 Coefficient–Based Data Reduction Operator

Let V be the multivariate tensor–product spline space of degree $\mathbf{d} = (d_1, d_2, ..., d_r)$, $r \in \mathbb{N}$ and $r \geq 1$, defined on a tensor–product mesh $\mathcal{G}$, with the associated basis of tensor–product B–splines

$$\mathcal{B}_{\mathbf{d}} := \{B_J,\, J \in \Gamma_{\mathbf{d}}\},$$

for the multi–index set $\Gamma_{\mathbf{d}}$.

Let us consider a spline $F \in V$ in the B–spline form

$$F = \sum_{J \in \Gamma_{\mathbf{d}}} c_J B_J \,,$$

with each $c_J \in \mathbb{R}$.

Let $\bar{V} \subseteq V$ be another space of splines of degree $\mathbf{d}$ defined on a tensor–product mesh $\bar{\mathcal{G}}$, and let

$$\bar{\mathcal{B}}_{\mathbf{d}} := \{\bar{B}_J, J \in \bar{\Gamma}_{\mathbf{d}}\}$$

be the corresponding B–spline basis.

Since $\bar{V} \subseteq V$, we have a linear relation between the basis of the two spaces:

$$\bar{\mathbf{B}}^{(\mathbf{d})} = R^T\, \mathbf{B}^{(\mathbf{d})},$$

where

$$\mathbf{B}^{(\mathbf{d})} := [B_J]_{J \in \Gamma_{\mathbf{d}}} \qquad \text{and} \qquad \bar{\mathbf{B}}^{(\mathbf{d})} := [\bar{B}_J]_{J \in \bar{\Gamma}_{\mathbf{d}}}$$

are vectors of length $|\Gamma_{\mathbf{d}}|$ and $|\bar{\Gamma}_{\mathbf{d}}|$, respectively, while R is the matrix of size $|\Gamma_{\mathbf{d}}| \times |\bar{\Gamma}_{\mathbf{d}}|$ obtained by using the knot insertion formula to move from $\bar{V}$ to V (see, e.g., [4]). We define the operator $Q : V \to \bar{V}$ as follows,

$$Q(F) := \sum_{J \in \bar{\Gamma}_{\mathbf{d}}} \bar{c}_J\, \bar{B}_J \,, \tag{1}$$

with each coefficient $\bar{c}_J$ obtained by setting $\bar{c}_J = d_J^J$, where d_J^J is the component of index J of the set of coefficients $\{d_K^J\}_{K \in \bar{L}_J}$ solution of the local least squares problem

$$\min_{d_K^J :\, K \in \bar{L}_J} \sum_{H \in L_J} \left[\left(\sum_{K \in \bar{L}_J} r_{H,K}\, d_K^J \right) - c_H \right]^2 , \tag{2}$$

with $r_{H,K}$ denoting the element of R in the H-th row and K-th column, and

$$\begin{aligned} \bar{L}_J &:= K \in \bar{\Gamma}_{\mathbf{d}} : \operatorname{supp}(\bar{B}_K) \cap \operatorname{supp}(\bar{B}_J) \neq \emptyset, \\ L_J &:= H \in \Gamma_{\mathbf{d}} : \operatorname{supp}(B_H) \cap \operatorname{supp}(\bar{B}_J) \neq \emptyset. \end{aligned} \tag{3}$$

Note that, considering (2) and (3), we can state that the coefficient $\bar{c}_J$ is the central coefficient of a local approximation of the restriction of F to the support of $\bar{B}_J$ defined on the analogous restriction of $\bar{V}$.

Since $\bar{V} \subset V$, we are approximating a spline surface with another spline surface belonging to a coarser space. Moreover, note that the computation of the coefficients of $Q(F)$ does not require any evaluation of the target spline F to be approximated. The next Proposition proves that Q is a projector into $\bar{V}$.

Proposition 1. *For any $F \in \bar{V}$, $Q(F) = F$.*

Proof. Since $F \in \bar{V}$, we have

$$F = \sum_{K \in \bar{\Gamma}_{\mathbf{d}}} \bar{a}_K \, \bar{B}_K \, ,$$

which can also be written in the form

$$F = \sum_{H \in \Gamma_{\mathbf{d}}} c_H B_H \, ,$$

with

$$c_H = \sum_{K:\, r_{H,K} > 0} r_{H,K} \bar{a}_K, \qquad H \in \Gamma_{\mathbf{d}}.$$

Note that for any $J \in \bar{\Gamma}_{\mathbf{d}}$, by the definitions of L_J and $\bar{L}_J$ in (3), if $H \in L_J$ it is

$$\{K \in \bar{\Gamma}_{\mathbf{d}} : r_{H,K} > 0\} = \{K \in \bar{L}_J : r_{H,K} > 0\}.$$

Therefore, we get

$$c_H = \sum_{K \in \bar{L}_J :\, r_{H,K} > 0} r_{H,K} \bar{a}_K = \sum_{K \in \bar{L}_J} r_{H,K} \bar{a}_K, \qquad H \in L_J.$$

This implies that, for any $J \in \bar{\Gamma}_{\mathbf{d}}$, it is

$$\sum_{H \in L_J} \left[\left(\sum_{K \in \bar{L}_J} r_{H,K} \bar{a}_K \right) - c_H \right]^2 = 0.$$

Since each coefficient $\bar{c}_J$ in (1) is obtained by solving the minimum problem (2), we must have $\bar{c}_J = \bar{a}_J$ for any $J \in \bar{\Gamma}_{\mathbf{d}}$, and, consequently, $Q(F) = F$.

3 Hierarchical Spline Spaces

This section briefly reviews (truncated) hierarchical B–spline—(T)HB–spline — construction and quasi–interpolation in hierarchical spline spaces. For a detailed introduction to (T)HB–splines and hierarchical quasi–interpolation, we refer to [12,13] and [2,24,25], respectively.

3.1 Hierarchical B–spline Bases

Let $V^{\ell-1} \subset V^\ell$ and $\Omega^{\ell-1} \supseteq \Omega^\ell$, $\ell = 1, \ldots, M$ be two nested sequences of multivariate tensor–product spline spaces and closed domains, respectively. By starting from an initial tensor–product configuration, each spline space V^ℓ is defined over a grid of level ℓ, obtained through h-refinement of the grid of level $\ell - 1$. The B–spline basis of degree $\mathbf{d}$ that spans the space V^ℓ is indicated as

$$\mathcal{B}^\ell_{\mathbf{d}} := \left\{ B^\ell_J,\ J \in \Gamma^\ell_{\mathbf{d}} \right\},$$

for a certain multi–index set $\Gamma^\ell_{\mathbf{d}}$. We assume $\Omega^0 = \Omega$ and $\Omega^M = \emptyset$. Each Ω^ℓ is defined as a collection of cells with respect to the tensor–product grid of level $\ell - 1$.

At each level ℓ, the set of B–splines B^ℓ_J whose support is completely inside Ω^ℓ but not in successive refined domains is included in the hierarchical B–spline (HB–spline) basis [16, 28].

Definition 1. *The hierarchical B–spline basis $\mathcal{H}_{\mathbf{d}}(\mathcal{G}_{\mathcal{H}})$ of degree $\mathbf{d}$ with respect to the mesh $\mathcal{G}_{\mathcal{H}}$ is defined as*

$$\mathcal{H}_{\mathbf{d}}(\mathcal{G}_{\mathcal{H}}) := \left\{ B^\ell_J \in \mathcal{B}^\ell_{\mathbf{d}} \ :\ J \in A^\ell_{\mathbf{d}},\ \ell = 0, ..., M-1 \right\},$$

where

$$A^\ell_{\mathbf{d}} := \{ J \in \Gamma^\ell_{\mathbf{d}} \ :\ \operatorname{supp} B^\ell_J \subseteq \Omega^\ell \ \wedge\ \operatorname{supp} B^\ell_J \not\subseteq \Omega^{\ell+1} \},$$

is the active set of multi–indices of level ℓ, $A^\ell_{\mathbf{d}} \subseteq \Gamma^\ell_{\mathbf{d}}$, and $\operatorname{supp} B^\ell_J$ denotes the intersection of the support of B^ℓ_J with Ω^0.

In view of the linear independence of hierarchical B–splines, they form a basis for the space $S_{\mathcal{H}} := \operatorname{span} \mathcal{H}_{\mathbf{d}}(\mathcal{G}_{\mathcal{H}})$ associated to the mesh $\mathcal{G}_{\mathcal{H}}$.

Definition 2. *Let*

$$s = \sum_{J \in \Gamma^{\ell+1}_{\mathbf{d}}} \sigma^{\ell+1}_J B^{\ell+1}_J,$$

be the representation in the B–spline basis of $V^{\ell+1} \supset V^\ell$ of $s \in V^\ell$. The truncation operators

$$\operatorname{trunc}^{\ell+1} : V^\ell \to V^{\ell+1} \quad \textit{and} \quad \operatorname{Trunc}^{\ell+1} : V^\ell \to S_{\mathcal{H}} \subseteq V^{M-1}$$

are defined as

$$\operatorname{trunc}^{\ell+1} s := \sum_{J \in \Gamma^{\ell+1}_{\mathbf{d}} \,:\, \operatorname{supp} B^{\ell+1}_J \not\subseteq \Omega^{\ell+1}} \sigma^{\ell+1}_J B^{\ell+1}_J, \quad \ell = 0, \ldots, M-1,$$

and

$$\operatorname{Trunc}^{\ell+1} := \operatorname{trunc}^{M-1}(\operatorname{trunc}^{M-2}(\cdots(\operatorname{trunc}^{\ell+1}(s))\cdots)), \qquad \ell = 0, \ldots, M-1,$$

respectively.

The operators introduced in Definition 2 allow us to define an alternative basis for the hierarchical spline space $S_{\mathcal{H}}$, known as truncated hierarchical B–spline (THB–spline) basis [12].

Definition 3. *The truncated hierarchical B–spline basis* $\mathcal{T}_{\mathbf{d}}(\mathcal{G}_{\mathcal{H}})$ *of degree* $\mathbf{d}$ *with respect to the mesh* $\mathcal{G}_{\mathcal{H}}$ *is defined as*

$$\mathcal{T}_{\mathbf{d}}(\mathcal{G}_{\mathcal{H}}) := \left\{T_J^\ell \,:\, J \in A_{\mathbf{d}}^\ell,\ \ell = 0,...,M-1\right\}, \quad \textit{with} \quad T_J^\ell := \mathrm{Trunc}^{\ell+1}(B_J^\ell)\,.$$

In view of the B–spline refinement rule and the non–negativity of HB–splines, by subtracting from coarser THB–splines the values of B–splines inserted at subsequent hierarchical levels, the truncated basis forms a convex partition of unity [12]. The truncation also guarantees the property of coefficient preservation: THB–splines preserve the coefficients of functions represented with respect to one of the bases $\mathcal{B}_{\mathbf{d}}^\ell$. This property is stated in [13, Theorem 12] and can be summarized as follows. Let $s|_{D^\ell}$ be the restriction of $s \in \operatorname{span} \mathcal{T}_{\mathbf{d}}(\mathcal{G}_H)$ to $D^\ell = \Omega^\ell \setminus \Omega^{\ell+1}$ and consider its representation with respect to $\mathcal{T}_{\mathbf{d}}(\mathcal{G}_H)$ and $\mathcal{B}_{\mathbf{d}}^\ell$,

$$s|_{D^\ell} = \sum_{k=0}^{M-1} \sum_{I \in A_{\mathbf{d}}^k} d_I^k T_I^k = \sum_{J \in \Gamma_{\mathbf{d}}^\ell} c_J^\ell B_J^\ell.$$

The coefficient d_I^ℓ of each THB–spline T_I^ℓ of level ℓ is equal to the coefficient c_I^ℓ of the B–spline B_I^ℓ from which T_I^ℓ is originated via truncation, namely $d_I^\ell = c_I^\ell$, $I \in A_{\mathbf{d}}^\ell$. In addition, THB–splines form a strongly stable basis: the constants arising in the stability analysis of the basis do not depend on the number of refinement levels, see [13, Theorem 19].

3.2 THB–Spline Quasi–Interpolation

The property of coefficient preservation mentioned at the end of the previous section directly leads to the generalization of any quasi–interpolation operator to the hierarchical setting [25]. Let $f \in C(\Omega^0)$ and let

$$Q^\ell(f) := \sum_{J \in \Gamma_{\mathbf{d}}^\ell} \lambda_J^\ell(f)\, B_J^\ell, \qquad \ell = 0, \ldots, M-1,$$

be a sequence of quasi–interpolants defined in terms of certain linear functionals $\lambda_J^\ell(f)$. Let also the B–spline B_J^ℓ related to the truncated basis function $T_J^\ell = \mathrm{Trunc}^{\ell+1}(B_J^\ell)$ through Definition 3, be the *mother* B–spline of T_J^ℓ. Thanks to the preservation of coefficients, the hierarchical quasi–interpolant is simply defined by associating at each THB–spline the linear functional of its mother function, namely

$$Q_{\mathcal{H}}(f) := \sum_{\ell=0}^{M-1} \sum_{J \in A_{\mathbf{d}}^\ell} \lambda_J^\ell(f)\, T_J^\ell.$$

Note that the property of reproducing polynomials is preserved by the hierarchical construction:

$$Q(p) = p \quad \Rightarrow \quad Q_{\mathcal{H}}(p) = p, \quad \forall p \in \mathbb{P}^{\mathbf{d}},$$

where $\mathbb{P}^{\mathbf{d}}$ is the space of tensor–product polynomials of degree $\mathbf{d}$. While [25] introduced the general framework for hierarchical quasi–interpolation based on the truncated basis together with the related properties, the hierarchical Hermite BS quasi–interpolation scheme was presented in [2]. THB–spline quasi–interpolation was recently discussed also in [24].

4 THB–Spline Simplification

Given a tensor–product B–spline function, possibly obtained by approximation of a set of gridded data or by interactive modeling and processing, our data reduction scheme produces an accurate THB–spline approximation with a strongly reduced number of degrees of freedom. This result is obtained by locally applying to the original B–spline function the coefficient–based operator introduced in Sect. 2 to compute the coefficient associated with each truncated basis function. Note that, in the case of regular grids, the refinement matrices which express the relation between the coefficients on different levels of the hierarchy and are needed by the least–squares operator depend only on the spline degree. Consequently, they can be computed once and for all in the implementation of the method.

4.1 The Hierarchical Coefficient–Based Operator

Let $\mathcal{G}_{\mathcal{H}}$ be a hierarchical mesh with M levels, and let $V^0 \subset \cdots \subset V^{M-1}$ be the sequence of associated nested tensor–product spline spaces with $V^{M-1} \subseteq V$. We recall from the previous section that $\mathcal{B}^{\ell}_{\mathbf{d}}$ is the B–spline basis of V^{ℓ}, while $\mathcal{G}^{\ell}$ is the associated tensor-product mesh. For any $F \in V$, of the form

$$F = \sum_{H \in \Gamma_{\mathbf{d}}} c_H B_H \,, \tag{4}$$

we define the hierarchical operator

$$Q_{\mathcal{H}}(F) := \sum_{\ell=0}^{M-1} \sum_{J \in A^{\ell}_{\mathbf{d}}} c^{\ell}_J \, T^{\ell}_J \,, \tag{5}$$

where each c^{ℓ}_J is the coefficient of the corresponding tensor–product operator of type (1) defined in the space V^{ℓ} and expressed as

$$Q^{\ell}(F) := \sum_{J \in \Gamma^{\ell}_{\mathbf{d}}} c^{\ell}_J \, B^{\ell}_J \,.$$

Analogously to the tensor–product case, each coefficient c_J^ℓ is obtained by solving the local least squares problem

$$\min_{c_K^\ell :\, K\in \bar{L}_J^\ell} \sum_{H\in L_J^\ell} \left[\left(\sum_{K\in \bar{L}_J^\ell} r_{H,K}^\ell c_K^\ell \right) - c_H \right]^2, \tag{6}$$

where c_H are the coefficients in the tensor–product B–spline representation of F provided by (4),

$$\bar{L}_J^\ell := K \in \Gamma_{\mathbf{d}}^\ell : \operatorname{supp}(B_K^\ell) \cap \operatorname{supp}(B_J^\ell) \neq \emptyset,$$
$$L_J^\ell := H \in \Gamma_{\mathbf{d}} : \operatorname{supp}(B_H) \cap \operatorname{supp}(B_J^\ell) \neq \emptyset,$$

and $r_{H,K}^\ell$ is the element in the H-th row and K-th column of the matrix R^ℓ so that

$$\mathbf{B}^{(\mathbf{d},\ell)} = (R^\ell)^T\, \mathbf{B}^{(\mathbf{d})}, \tag{7}$$

with

$$\mathbf{B}^{(\mathbf{d},\ell)} := [B_J^\ell]_{J\in\Gamma_{\mathbf{d}}^\ell} \quad \text{and} \quad \mathbf{B}^{(\mathbf{d})}$$

representing the B–spline bases of V^ℓ and V, respectively. Note that, for given

$$0 \le \ell \le M-1 \quad \text{and} \quad J \in A_{\mathbf{d}}^\ell,$$

only a submatrix of R^ℓ is employed for computing the solution of (6), namely

$$R_J^\ell := [r_{H,K}]_{H\in L_J^\ell,\, K\in \bar{L}_J^\ell}.$$

This matrix can be obtained as the Kronecker product of matrices expressing the relation between univariate B–splines:

$$R_J^\ell = R_{J,1}^\ell \otimes R_{J,2}^\ell \otimes \cdots \otimes R_{J,r}^\ell,$$

where

$$\mathbf{B}_{J,h}^{(\mathbf{d},\ell)} = (R_{J,h}^\ell)^T\, \mathbf{B}_{J,h}^{(\mathbf{d})}$$

with $\mathbf{B}_{J,h}^{(\mathbf{d})}$ and $\mathbf{B}_{J,h}^{(\mathbf{d},\ell)}$ being the vectors containing the univariate B–splines whose tensor–product gives the r-variate B–splines B_H, $H \in L_J^\ell$ and B_K, $K \in \bar{L}_J^\ell$, respectively.

Remark 1. We observe that, when we consider uniform meshes on each level and $V = V^{M-1}$, each matrix $R_{J,h}^\ell$ depends only on the degree $\mathbf{d}$, and on the number of dyadic refinements needed to pass from $\mathbf{B}_{J,h}^{(\mathbf{d},\ell)}$ to $\mathbf{B}_{J,h}^{(\mathbf{d})}$, that is, $M-1-\ell$. For example, in the case of only single knots at all levels, when $r=2$, $\mathbf{d}=(2,2)$ and $M-1-\ell=1$, for any J, we have

$$R_{J,h}^\ell = \begin{bmatrix} 3/4 & 1/4 & 0 & 0 & 0 \\ 1/4 & 3/4 & 0 & 0 & 0 \\ 0 & 3/4 & 1/4 & 0 & 0 \\ 0 & 1/4 & 3/4 & 0 & 0 \\ 0 & 0 & 3/4 & 1/4 & 0 \\ 0 & 0 & 1/4 & 3/4 & 0 \\ 0 & 0 & 0 & 3/4 & 1/4 \\ 0 & 0 & 0 & 1/4 & 3/4 \end{bmatrix}, \qquad h = 1, ..., r.$$

The following proposition proves that $Q_{\mathcal{H}}$ reproduces the polynomial space $\mathbb{P}_{\mathbf{d}}$.

Proposition 2. *For any* $q \in \mathbb{P}_{\mathbf{d}}$, $Q_{\mathcal{H}}(q) = q$.

Proof. Note that, by Proposition 1 we have

$$Q^{\ell}(q) = q \quad \text{for any} \quad q \in \mathbb{P}_{\mathbf{d}}, \quad \ell = 0, ..., M-1.$$

As a consequence, by applying Theorem 3 in [25], we obtain the thesis.

In addition, the hierarchical operator $Q_{\mathcal{H}}$ reproduces all splines of the subspace V^0, as it is proved in Proposition 3 below.

Proposition 3. *For any* $F \in V^0$, $Q_{\mathcal{H}}(F) = F$.

Proof. Let us consider c_J^{ℓ} in (5) determined by solving problem (6). Since $F \in V^0 \subseteq V^{\ell} \subseteq ... \subseteq V^{M-1} \subseteq V$, we have

$$F = \sum_{K \in \Gamma_{\mathbf{d}}^0} a_K^0 \, B_K^0 = \sum_{K \in \Gamma_{\mathbf{d}}^{\ell}} a_K^{\ell} \, B_K^{\ell} = \sum_{H \in \Gamma_{\mathbf{d}}} c_H B_H \; ,$$

with

$$c_H = \sum_{K:\, r_{H,K}^{\ell} > 0} r_{H,K}^{\ell} a_K^{\ell}, \qquad H \in \Gamma_{\mathbf{d}},$$

where each $r_{H,K}^{\ell}$ is the element in the H-th row and K-th column of the matrix R^{ℓ} in (7). Analogously to the proof of Proposition 1, this is enough to prove that $c_J^{\ell} = a_J^{\ell}$. This in turn, by using the THB–spline property of coefficient preservation [13], implies that $Q_{\mathcal{H}}(F) = F$.

It is clear that the accuracy of the hierarchical approximation $Q_{\mathcal{H}}(F)$ of $F \in V$ strongly depends on the choice of the hierarchical mesh $\mathcal{G}_{\mathcal{H}}$, and a strategy for its automatic generation is crucial.

4.2 The Adaptive Data Reduction Scheme

Let V be a $\mathbf{d}$–degree tensor–product spline space, $\mathbf{d} = (d_1, d_2, ..., d_r)\,, r \in \mathbb{N}$ and $r \geq 1$. For simplicity, we assume that V is defined on a grid $\mathcal{G}^{M_{max}-1}$ obtained from a coarser grid $\mathcal{G}^0$ by applying $M_{max} - 1$ successive dyadic refinements. Consequently, the mesh $\mathcal{G}^{\ell}$ is obtained by one dyadic refinement of the cells of $\mathcal{G}^{\ell-1}$, $\ell = 1, ..., M-1$, with $M \leq M_{max}$. Let

$$\mathbf{T} := [T_K]_{K \in \Gamma_{\mathbf{d}}^{\mathcal{T}}} \quad \text{with} \quad \Gamma_{\mathbf{d}}^{\mathcal{T}} := \{(\ell, I) :\, I \in A_{\mathbf{d}}^{\ell}\,, 0 \leq \ell \leq M-1\}$$

be the set of THB–splines defined by the spline hierarchy. We can then write

$$\mathbf{T} = P^T \, \mathbf{B}^{\mathbf{d}}$$

where the transpose of P is the matrix that expresses the linear relation between the basis of the hierarchical spline space and the basis of V. We denote by $p_{J,K}$

the element of P in the J-th row and K-th column. For subsequent use, we also rewrite (5) as

$$Q_{\mathcal{H}}(F) = \sum_{K \in \Gamma_{\mathbf{d}}^{\mathcal{T}}} c_K^{\mathcal{T}} T_K \,. \tag{8}$$

The following ascending algorithm summarizes the main steps to compute a THB–spline with $M \leq M_{max}$ levels which approximates $F \in V$ with knots in $\mathcal{G}_{M_{max}-1}$ within a given tolerance ϵ. As previously mentioned, for simplicity, we assume that V is a spline space whose mesh can be dyadically simplified $M_{max} - 1$ times.

Input:

- the set of coefficients $\{c_J, J \in \Gamma_{\mathbf{d}}\}$ defining $F \in V$ with knots in $\mathcal{G}_{M_{max}-1}$;
- a dyadic coarsening $\mathcal{G}_0$ of $\mathcal{G}_{M_{max}-1}$;
- a maximum number of hierarchical levels $M \leq M_{max}$;
- the tolerance $\epsilon > 0$.

1. initialize $\mathcal{G}_{\mathcal{H}} = \mathcal{G}_0$ and, consequently, $\Gamma_{\mathbf{d}}^{\mathcal{T}}$ and P ;
2. compute the coefficients $c_K^{\mathcal{T}}$, $K \in \Gamma_{\mathbf{d}}^{\mathcal{T}}$, of $Q_{\mathcal{H}}(F)$ in (8) by solving for each of them the local least square system in (6);
3. while
$$\left| \sum_{K:\, p_{J,K}>0} p_{J,K} c_K^{\mathcal{T}} - c_J \right| \leq \epsilon \cdot \left(\max_{H \in \Gamma_{\mathbf{d}}} c_H - \min_{H \in \Gamma_{\mathbf{d}}} c_H \right) \tag{9}$$
is not satisfied for all $J \in \Gamma_{\mathbf{d}}$ and the current number of levels is less than M, repeat the following steps:
   (a) for all $J \in \Gamma_{\mathbf{d}}$ which do not satisfy (9), mark the cells which belong to $\mathrm{supp}(B_I^{\ell})$ for all $K = (\ell, I) \in \Gamma_{\mathbf{d}}^{\mathcal{T}}$ such that $p_{J,K} > 0$;
   (b) obtain the new mesh $\mathcal{G}_{\mathcal{H}}$ by dyadically refining each marked cell belonging to $\mathcal{G}_{\ell}, \ell < M - 1$ and update $\Gamma_{\mathbf{d}}^{\mathcal{T}}$ and P ;
   (c) compute the new coefficients $c_K^{\mathcal{T}}$, $K \in \Gamma_{\mathbf{d}}^{\mathcal{T}}$, of $Q_{\mathcal{H}}(F)$ in (8) by solving for each of them the local least square system in (6).

Output: THB–spline approximation $Q_{\mathcal{H}}(F)$ with $M \leq M_{max}$ levels of the form (8) approximating F within the given tolerance ϵ.

In the stopping criterion, the tolerance ϵ is compared with the error in the current hierarchical approximation of c_J, scaled with respect to the data, according to (9). The right–hand side of (9) vanishes if all the coefficients $c_H, H \in \Gamma_{\mathbf{d}}$, are equal to a constant. Even if this case is not of practical interest, we may note that it is still covered by the algorithm since the target spline is just a constant exactly represented already in V_0.

Note that, in step 3(a) of the algorithm, in order to avoid additional computations, instead of marking the cells in the support of the THB–splines associated to the coefficients that do not satisfy the desired tolerance, we simply consider the support of the corresponding B–splines. This is justified since the support of a THB–spline is contained in the one of its mother function, namely the B–spline from which the truncated basis functions is obtained by truncation.

Remark 1. *It is worth to mention that the whole algorithm can be naturally generalized for applying the data reduction scheme to tensor–product B–spline parametric surfaces where the coefficients are replaced by control points. Since* c_H *is now a vector and no more a scalar, the necessary changes consist in replacing the square brackets in* (2) *and* (6) *and the absolute value in* (9) *with the euclidean norm, and substituting the normalizing factor* $(\max_{H\in\Gamma_{\mathbf{d}}} c_H - \min_{H\in\Gamma_{\mathbf{d}}} c_H)$ *in* (9) *with* $\max_{H,K\in\Gamma_{\mathbf{d}}} \|c_H - c_K\|$.

Remark 2. *Note that, when* $M = M_{max}$*, the algorithm always succeeds at meeting any tolerance, at most by producing a hierarchical mesh with* M_{max} *levels. This is due to the fact that, at each iteration of the algorithm, if* (9) *is not satisfied for a certain* $J \in \Gamma_{\mathbf{d}}$*, all the cells belonging to the supports of the B–splines* B_I^ℓ*,* $K = (I,\ell)$*, such that* $p_{J,K} > 0$ *are refined. As a consequence, at each iteration the level* ℓ *of the indices* $K = (\ell, I)$ *such that* $p_{J,K} > 0$ *increases by* 1*. Eventually, in the worst case we will get* $\ell = M_{max} - 1$ *and* $\{K = (\ell, I) : p_{J,K} > 0\} = \{J\}$*, that is, the obtained hierarchical space is locally a tensor–product space. Therefore,* $c_J^{M_{max}-1} = c_J$*, which of course implies that* (9) *is satisfied for* c_J*.*

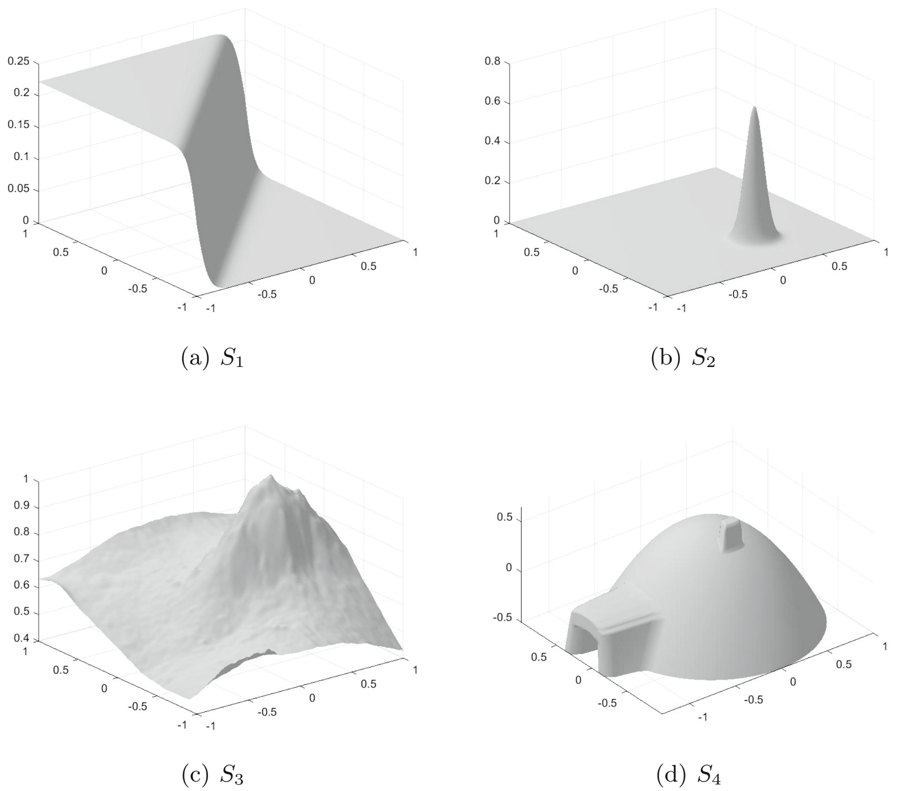
5 Numerical Experiments

For testing the proposed hierarchical data reduction scheme, we implemented the coefficient–based scheme in MATLAB and combined it with THB–splines by relying on the hierarchical B–spline implementation within the MATLAB package GeoPDEs, see [11,27]. Open knot vectors are considered for all the examples.

Example 1. *We first consider two test tensor-product B–spline surfaces,* $S_i, i = 1, 2$*, shown in Fig. 1(a) and (b). Each of them was obtained with a preliminary spline approximation of a corresponding set of* 129×129 *uniformly gridded functional data. More precisely, the tensor–product extension of the BS Hermite QI scheme introduced in [18] was adopted for this aim. The two discrete data sets used to generate* S_1 *and* S_2 *were defined by uniformly sampling the following two test functions,*

$$f_1(x,y) = \frac{\tanh(9y - 9x) + 1}{9}, \qquad (x,y) \in [-1,1]^2,$$
$$f_2(x,y) = \frac{2}{3\exp{(10x-3)^2 + (10y+4)^2}}, \qquad (x,y) \in [-1,1]^2.$$

Example 2. *We applied the algorithm to a tensor–product B–spline surface* S_3 *approximating the set of geographic data available at [26] and describing the terrain elevation in a mountain region of the Hawaii Islands, see Fig. 1(c). The tensor–product surface was obtained with a modified version of the BS Hermite QI scheme (mentioned in [18]). Such variant, unlike the basic one, does not require the values of the first and second–order mixed partial derivatives of the approximated function on the rectangular mesh defining the spline knots.*



(a) S_1 (b) S_2

(c) S_3 (d) S_4

Fig. 1. The reference tensor–product spline surfaces $S_i, i = 1, 2, 3, 4$. The spline breakpoints are 129 uniformly spaced points in $[-1\,,\,1]$ with respect to both directions.

Example 3. *In this example, we applied the data reduction algorithm to the "igloo"model* S_4*, defined in a tensor–product B–spline space of degree* $(3, 3)$ *on a* 128×128 *uniform grid, see Fig. 1(d). In this case the reference parametric surface is obtained through control point modification and the control points* c_J *belong to* $\mathbb{R}^3$*.*

In all the experiments we set $M = M_{max}$. For each test, we report the spline degree, the number M of levels, the tolerance ϵ used for generating the hierarchical mesh and the dimension of the spaces $S_{\mathcal{H}}$ and V. In addition, the last column of the table shows the discrete approximation of the infinity norm of the error e_i

$$e_i := Q_{\mathcal{H}}(S_i) - S_i, \qquad i = 1, 2, 3, \qquad e_4 := \|Q_{\mathcal{H}}(S_4) - S_4\|_2,$$

computed by sampling the error at the vertices of the original tensor-product grid. It is clear that the data reduction approximation error can be controlled by setting a suitable tolerance for the marking strategy considered in the algorithm.

Note that, for any considered test, there is a significative reduction of the number of degrees of freedom, thanks to the local refinement capabilities of hierarchical spline spaces.

Table 1 shows the results obtained by applying the hierarchical operator to the four reference surfaces with different tolerance values (ϵ=5e-2,1e-2,5e-3). The adaptive nature of the refinements obtained with the application of the algorithm is evident from the hierarchical meshes generated by the THB–spline simplification approach, see Figs. 2, 3 and 4 (right). The comparison between the approximated surfaces shown in Figs. 2, 3 and 4 (left) and the original surfaces $S_i, i = 1, \ldots, 4$ of Fig. 1 suggests that the shape of the data is also well reproduced. The corresponding contour plots are shown in Figs. 5 and 6 which confirm the good quality of the approximations (only very minor differences between the original and the approximated contour plot are present). Different experiments with periodic (rather than open) knot vectors suggest that this choice leads to more refined meshes near the boundary (and consequently more degrees of freedom).

Table 1. Numerical results obtained by applying the hierarchical quasi-interpolation operator $Q_{\mathcal{H}}$ to the tensor–product splines S_1, S_2, S_3, S_4.

S_1 $(d_1 = d_2 = 3)$				
M	ϵ	$\dim(S_{\mathcal{H}})$	$\dim(V)$	$\|e_1\|_\infty$
3	5e-2	361	17161	5.519e-3
4	1e-2	973	17161	2.546e-4
4	5e-3	1027	17161	2.546e-4
S_2 $(d_1 = d_2 = 3)$				
M	ϵ	$\dim(S_{\mathcal{H}})$	$\dim(V)$	$\|e_2\|_\infty$
4	5e-2	550	17161	9.251e-3
5	1e-2	820	17161	1.609e-3
5	5e-3	928	17161	3.456e-4
S_3 $(d_1 = d_2 = 2)$				
M	ϵ	$\dim(S_{\mathcal{H}})$	$\dim(V)$	$\|e_3\|_\infty$
3	5e-2	190	16900	2.125e-2
6	1e-2	2485	16900	4.733e-3
6	5e-3	5718	16900	2.504e-3
S_4 $(d_1 = d_2 = 3)$				
M	ϵ	$\dim(S_{\mathcal{H}})$	$\dim(V)$	$\|e_4\|_\infty$
6	5e-2	2848	17161	1.373e-2
6	1e-2	3739	17161	3.835e-3
6	5e-3	3952	17161	3.334e-4

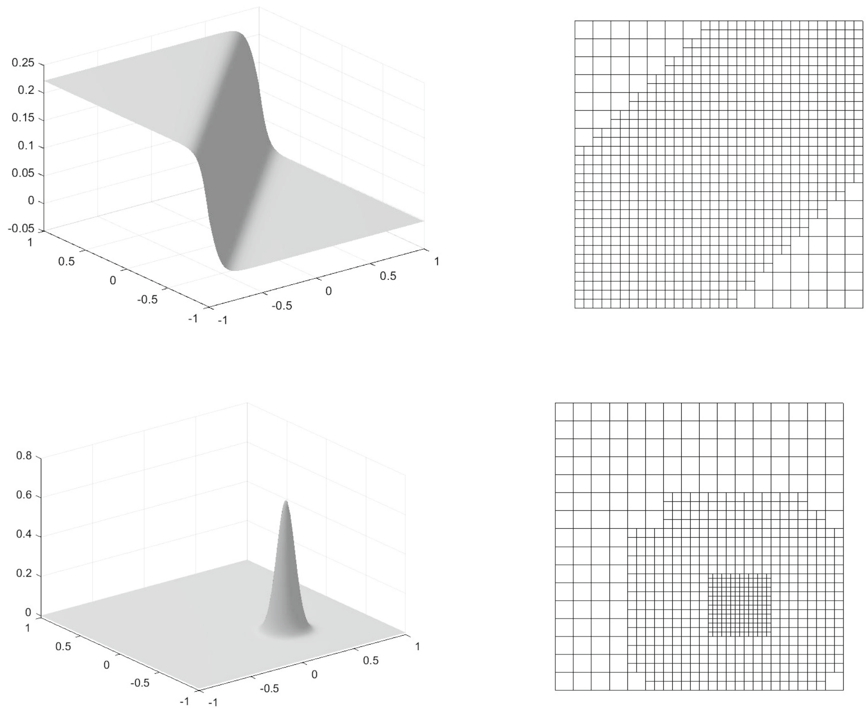


Fig. 2. THB–spline approximations (left) and corresponding hierarchical meshes (right) obtained by applying $Q_{\mathcal{H}}$ to the tensor–product splines of Example 1 with ϵ=1e-2.

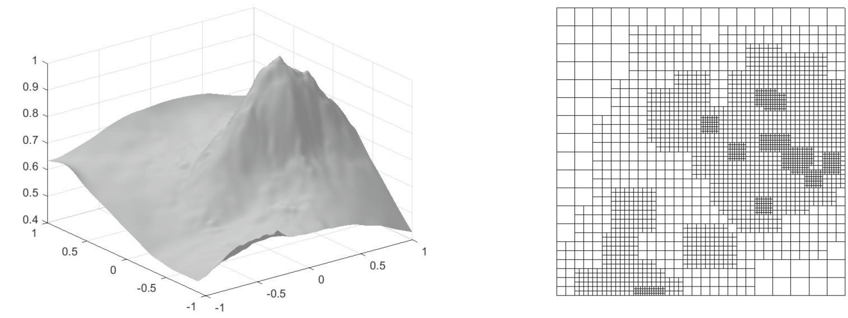


Fig. 3. THB–spline approximation (left) and corresponding hierarchical mesh (right) obtained by applying $Q_{\mathcal{H}}$ to the tensor–product spline of Example 2 with ϵ=1e-2.

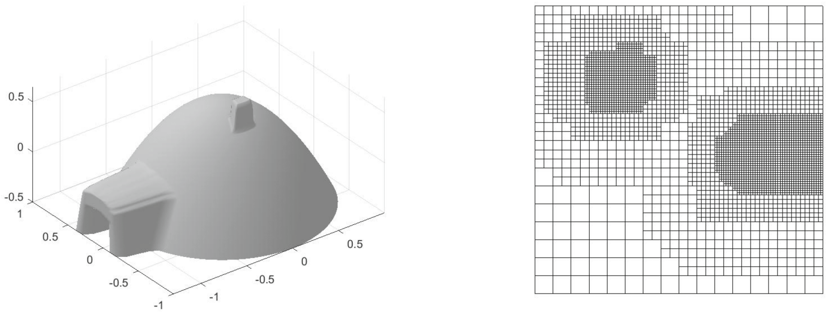


Fig. 4. THB–spline approximation (left) and corresponding hierarchical mesh (right) obtained by applying $Q_{\mathcal{H}}$ to the tensor–product spline of Example 3 with ϵ=1e-2.

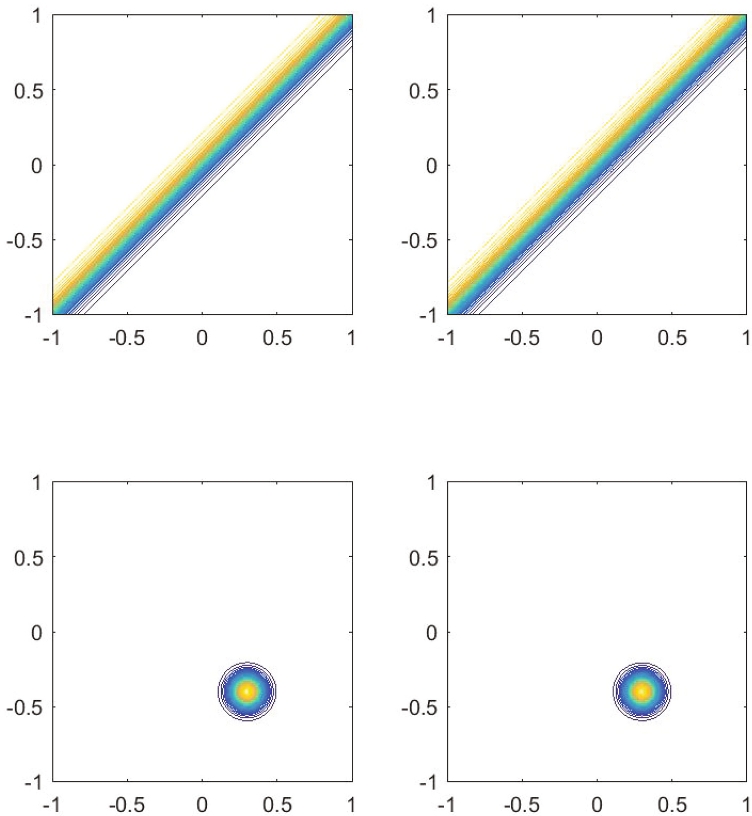


Fig. 5. Contour plots of the tensor–product splines (left) of Example 1: S_1 (top) and S_2 (bottom) and of their THB–spline approximations (right) obtained with ϵ=1e-2.

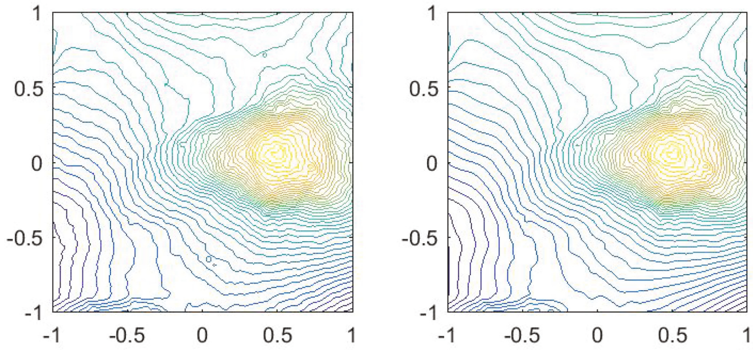


Fig. 6. Contour plots of the tensor–product spline of Example 2 (left) and of its THB–spline approximation (right) obtained with ϵ=1e-2.

Table 2. Numerical results obtained by applying the hierarchical quasi-interpolation operator $Q_{\mathcal{H}}$ to the tensor–product splines S_1, S_2, S_3, shown in Fig. 1(a), (b) and (c).

test	M	$d_1 = d_2$	ϵ	$\dim(S_{\mathcal{H}})$	$\dim(V)$	$\|e_i\|_\infty$
S_1	6	3	5.0e-6	7597	17161	1.702e-7
S_2	6	3	5.0e-6	3142	17161	5.157e-7
S_3	6	2	5.0e-3	5718	16900	2.504e-3

In order to show that the adaptive scheme can generate approximations with the same accuracy of the tensor–product case with a reduced number of degrees of freedom, we also present the results in Table 2. In this case the tolerance values were chosen of the same order of the error obtained by approximating the original data with tensor–product B–splines. The corresponding meshes are shown in Figs. 7 and 8.

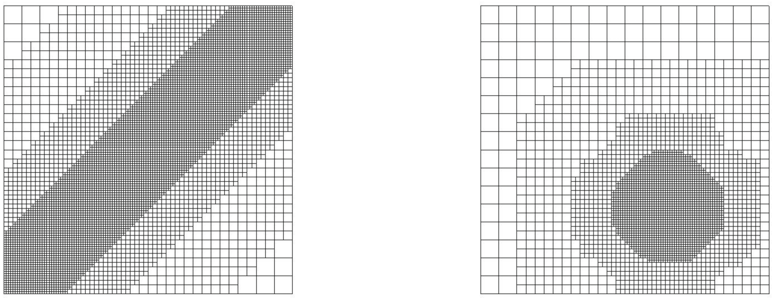


Fig. 7. Hierarchical meshes (right) obtained by applying $Q_{\mathcal{H}}$ to the tensor–product splines of Example 1 with ϵ=5e-6.

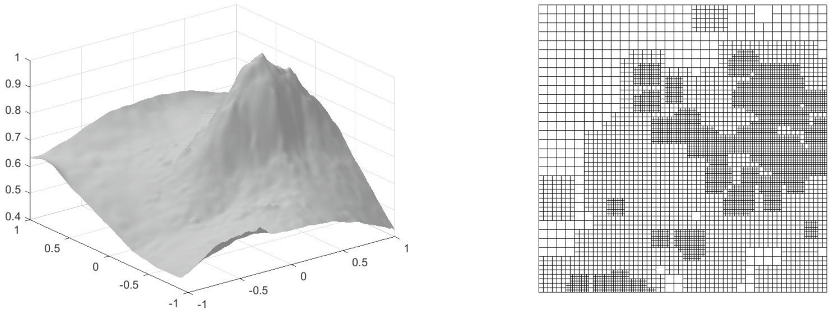


Fig. 8. THB–spline approximation (left) and corresponding hierarchical mesh (right) obtained by applying $Q_{\mathcal{H}}$ to the tensor–product spline of Example 2 with ϵ=5e-3.

6 Conclusions

In order to reduce the computational costs connected with the reconstruction of large data sets, we introduced a data reduction operator that does not require any pointwise functional evaluation and its THB–spline generalization. Such operator can be applied to any initial (highly refined) standard bivariate spline, preliminarily constructed by suitable classical spline approximation, or alternatively obtained either by control point modification of an initial spline configuration, or as the result of modeling techniques. The THB–spline simplification algorithm ensures accurate spline representations with a strongly reduced number of degrees of freedom. The algorithm can also be exploited for interactive design and model simplification. In principle, the data reduction scheme can also be applied to other kind of Bernstein/B–spline-type representations, assuming to start with a target function represented in this alternative form. The analysis of the influence of the chosen representation on the final approximation is beyond the scope of this paper.

Acknowledgements. The support by MIUR "Futuro in Ricerca" programme through the project DREAMS (RBFR13FBI3) and by the Istituto Nazionale di Alta Matematica (INdAM) through Gruppo Nazionale per il Calcolo Scientifico (GNCS)—"Finanziamento Giovani Ricercatori" and "Progetti di ricerca" programmes—and Finanziamenti Premiali SUNRISE are gratefully acknowledged.

References

1. Berdinsky, D., Kim, T.-W., Bracco, C., Cho, D., Mourrain, B., Oh, M.-J., Kiatpanichgij, S.: Dimensions and bases of hierarchical tensor-product splines. J. Comput. Appl. Math. **257**, 86–104 (2014)
2. Bracco, C., Giannelli, C., Mazzia, F., Sestini, A.: Bivariate hierarchical Hermite spline quasi-interpolation. BIT **56**, 1165–1188 (2016)
3. Conti, C., Morandi, R., Rabut, C., Sestini, A.: Cubic spline data reduction choosing the knots from a third derivative criterion. Numer. Algorithms **28**, 45–61 (2001)

4. de Boor, C.: A Practical Guide to Splines. Springer, New York (2001). Revised ed
5. Deng, J., Chen, F., Feng, Y.: Dimensions of spline spaces over T-meshes. J. Comput. Appl. Math. **194**, 267–283 (2006)
6. Deng, J., Chen, F., Li, X., Hu, C., Tong, W., Yang, Z., Feng, Y.: Polynomial splines over hierarchical T-meshes. Graph. Models **70**, 76–86 (2008)
7. Dokken, T., Lyche, T., Pettersen, K.F.: Polynomial splines over locally refined box-partitions. Comput. Aided Geom. Des. **30**, 331–356 (2013)
8. Forsey, D.R., Bartels, R.H.: Hierarchical B-spline refinement. Comput. Graphics **22**, 205–212 (1988)
9. Forsey, D.R., Bartels, R.H.: Surface fitting with hierarchical splines. ACM Trans. Graphics **14**, 134–161 (1995)
10. Forsey, D.R., Wong, D.: Multiresolution surface reconstruction for hierarchical B-splines. In: Graphics, Interface, pp. 57–64 (1998)
11. Garau, E.M., Vázquez, R.: Algorithms for the implementation of adaptive isogeometric methods using hierarchical splines. Appl. Numer. Math. **123**, 58–87 (2018)
12. Giannelli, C., Jüttler, B., Speleers, H.: THB-splines: the truncated basis for hierarchical splines. Comput. Aided Geom. Des. **29**, 485–498 (2012)
13. Giannelli, C., Jüttler, B., Speleers, H.: Strongly stable bases for adaptively refined multilevel spline spaces. Adv. Comput. Math. **40**, 459–490 (2014)
14. Greiner, G., Hormann, K.: Interpolating and approximating scattered 3D-data with hierarchical tensor product B-splines. In: Le Méhauté, A., Rabut, C., Schumaker, L.L. (eds.) Surface Fitting and Multiresolution Methods, pp. 163–172. Vanderbilt University Press, Nashville (1997)
15. Kiss, G., Giannelli, C., Zore, U., Jüttler, B., Großmann, D., Barner, J.: Adaptive CAD model (re-)construction with THB-splines. Graph. Models **76**, 273–288 (2014)
16. Kraft, R.: Adaptive and linearly independent multilevel B-splines. In: Le Méhauté, A., Rabut, C., Schumaker, L.L. (eds.) Surface Fitting and Multiresolution Methods, pp. 209–218. Vanderbilt University Press, Nashville (1997)
17. Lyche, T., Mørken, K.: A data-reduction strategy for splines with applications to the approximation of functions and data. IMA J. Numer. Anal. **8**, 185–208 (1988)
18. Mazzia, F., Sestini, A.: The BS class of Hermite spline quasi-interpolants on nonuniform knot distribution. BIT **49**, 611–628 (2009)
19. Mokriš, D., Jüttler, B., Giannelli, C.: On the completeness of hierarchical tensor-product B-splines. J. Comput. Appl. Math. **271**, 53–70 (2014)
20. Morandi, R., Sestini, A.: Data reduction in surface approximation. In: Lyche, T., Schumaker, L. (eds.) Mathematical Methods for Curves and Surfaces: Oslo 2000, pp. 315–324. Vanderbilt University Press, Nashville (2001)
21. Sederberg, T.W., Cardon, D.L., Finnigan, G.T., North, N.S., Zheng, J., Lyche, T.: T-spline simplification and local refinement. ACM Trans. Graphics **23**, 276–283 (2004)
22. Sederberg, T.W., Zheng, J., Bakenov, A., Nasri, A.: T-splines and T-NURCCS. ACM Trans. Graphics **22**, 477–484 (2003)
23. Skytt, V., Barrowclough, O., Dokken, T.: Locally refined spline surfaces for representation of terrain data. Comput. Graphics **49**, 58–68 (2015)
24. Speleers, H.: Hierarchical spline spaces: quasi-interpolants and local approximation estimates. Adv. Comput. Math. **43**, 235–255 (2017)
25. Speleers, H., Manni, C.: Effortless quasi-interpolation in hierarchical spaces. Numer. Math. **132**, 155–184 (2016)
26. U.S. Geological Survey. https://www.usgs.gov/, http://dds.cr.usgs.gov/pub/data/nationalatlas/el_usa_hawaii.bil_nt00924.tar.gz

27. Vázquez, R.: A new design for the implementation of isogeometric analysis in Octave and Matlab: GeoPDEs 3.0. Comput. Math. Appl. **72**, 523–554 (2016)
28. Vuong, A.-V., Giannelli, C., Jüttler, B., Simeon, B.: A hierarchical approach to adaptive local refinement in isogeometric analysis. Comput. Methods Appl. Mech. Eng. **200**, 3554–3567 (2011)
29. Wever, U.A.: Global and local data reduction strategies for cubic splines. Comput. Aided Des. **23**, 127–132 (1991)

http://www.springer.com/978-3-319-67884-9

Mathematical Methods for Curves and Surfaces
9th International Conference, MMCS 2016, Tønsberg,
Norway, June 23–28, 2016, Revised Selected Papers
Floater, M.; Lyche, T.; Mazure, M.-L.; Mørken, K.;
SCHUMAKER, L. (Eds.)
2017, VIII, 325 p. 151 illus., Softcover
ISBN: 978-3-319-67884-9