THB–splines: the truncated basis for hierarchical splines

Carlotta Giannelli[1], Bert Jüttler[1], Hendrik Speleers[2]

[1] Institute of Applied Geometry, Johannes Kepler University Linz, Austria
[2] Department of Computer Science, Katholieke Universiteit Leuven, Belgium

Abstract

The construction of classical hierarchical B–splines can be suitably modified in order to define locally supported basis functions that form a *partition of unity*. We will show that this property can be obtained by reducing the support of basis functions defined on coarse grids, according to finer levels in the hierarchy of splines. This *truncation* not only decreases the overlapping of supports related to basis functions arising from different hierarchical levels, but it also improves the numerical properties of the corresponding hierarchical basis — which is denoted as *truncated hierarchical B–spline (THB–spline) basis*. Several computed examples will illustrate the adaptive approximation behavior obtained by using a refinement algorithm based on THB–splines.

1 Introduction

The key requirement of an effective mesh refinement algorithm is to provide a *local* and *adaptive* procedure which enables to refine the underlying geometric representations so that only specific local regions of the domain are affected (see, e.g., Figure 1, top left). Standard tensor–product representations, as classical B–splines and NURBS models, preclude a strictly localized editing of the mesh. This kind of refinement is naturally supported by the hierarchical spline model [4], where different levels of details are identified by means of a hierarchy of tensor–product B–splines.

The hierarchical model controls the locality of the refinement through an adaptive procedure that is exclusively based on local basis refinement. In order to construct a basis of the hierarchical spline space, a specific selection mechanism was proposed in [11] and extended in [20]. More precisely, the so-called *hierarchical B–splines* (HB–splines)

- are locally supported, linearly independent, and non-negative;
- allow an effective local control of the refinement.

Surface reconstruction schemes for solving interpolation and approximation problems by using multilevel B–splines were originally discussed by Forsey and Bartels [5], and shortly after, additionaly investigated by Greiner and Hormann [8]. Kraft developed a quasi–interpolation algorithm based on hierarchical B–splines in [11]. There is an increasing interest in hierarchical splines coming from recent studies related to isogeometric analysis [20].

To address the problem of local mesh refinement, also other extensions of tensor–product representations were developed. T–splines [17, 18] and PHT–splines [2] are defined over *T–meshes*, where T–junctions between axis aligned segments are allowed. The T–spline framework has already shown its potential as a powerful modeling tool for advanced computer aided geometric design problems. Complex CAD geometries can be efficiently represented as a watertight T–spline model. Related plug–ins are already included in commercial modeling software for designers.
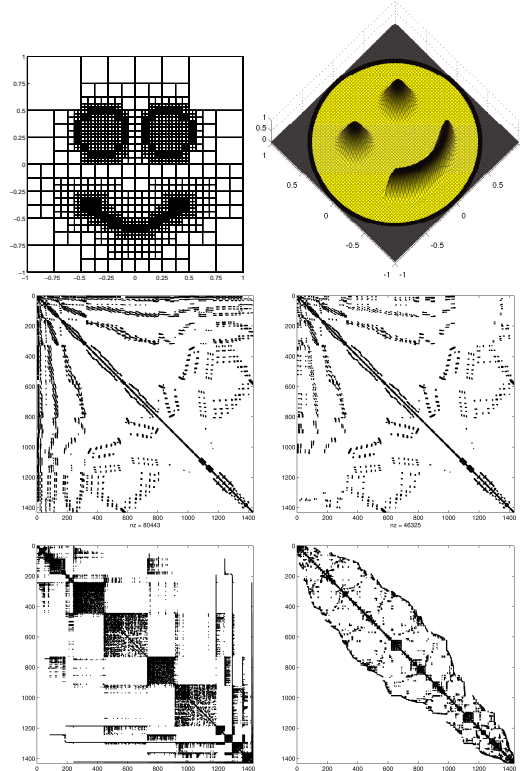

Figure 1: An example of a locally refined hierarchical mesh and the corresponding hierarchical spline representation (top). The structure of the least–squares matrices associated with the HB–spline basis (middle left) and the THB–spline basis (middle right) are also shown, together with their reordered version according to the reverse Cuthill–McKee algorithm (bottom). The THB/HB ratio with respect to sparsity is 57.59%.

Nevertheless, the corresponding refinement procedure may cause a propagation of the refinement beyond the regions marked by the error estimator [3]. In particular, the linear independence of the T–spline blending functions can be guaranteed only by considering a restricted subset of T–splines [1, 13, 16]. In order to allow this, a more involved algorithm — which reduces, but still does not eliminate, the unwanted propagation of the refinement — has to be considered and has been studied in [16].

In the context of PHT–splines, i.e. cubic splines over hierarchical T–meshes with reduced C^1 regularity, no refinement propagation is observed — see for example the original papers [2, 12] or their recent application for the numerical solution of partial differential equations [9, 14]. Despite a proper local refinement behavior in this case, the reduced regularity leads to a higher number of degrees of freedom to achieve a certain given accuracy.

The goal of the present paper is the definition and analysis of a normalized basis for hierarchically refined spline

spaces. The construction of hierarchical B-splines can be modified by suitably *truncating* basis functions according to finer levels in the hierarchy. After representing each coarse hierarchical basis function in terms of basis functions defined on a finer grid, we eliminate from this representation the contribution corresponding to the subset of finer basis functions that are effectively included in the hierarchical basis. The truncation mechanism preserves all the nice properties of hierarchical B–splines, such as linear independence and non–negativity. In addition, *truncated hierarchical B–splines* (THB–splines)

- have *smaller support*;
- form a *partition of unity*.

The potential of the THB–spline model with respect to the locality of the refinement and also to the sparsity of corresponding matrices is illustrated in Figure 1. A similar truncation approach has been proposed in [19] to define a normalized basis for hierarchical spline spaces defined on Powell–Sabin triangulations.

The paper is organized as follows. Section 2 introduces the definition of HB– and THB–splines, while Section 3 outlines the key properties of the THB–spline basis in the general multivariate setting. As an application we consider an adaptive approximation algorithm based on the hierarchical model in Section 4. We use this framework to demonstrate the good numerical properties of the THB–spline basis. Finally, Section 5 makes some concluding remarks and summarizes the key results of this paper.

2 From HB–splines to the THB–spline basis

In order to define the background machinery for the construction of spline hierarchies, let

$$V^0 \subset V^1 \subset \ldots \subset V^{N-1}$$

be a nested sequence of n–variate B–spline function spaces defined on the domain Ω^0. Each spline space V^ℓ, $\ell = 0, \ldots, N-1$, is spanned by a given normalized B–spline basis $\mathcal{B}^\ell$.

For each level ℓ, with $\ell = 0, \ldots, N-1$, let d be the degree in any of the n dimensions of the spline space $V^\ell \subset C^{d-1}(\Omega^0)$ spanned by the tensor–product B–spline basis $\mathcal{B}^\ell$ defined on corresponding knot sequences. To guarantee the nested nature of the spline spaces, these knot sequences are also assumed to be nested.

In addition, let

$$\Omega^0 \supseteq \Omega^1 \supseteq \ldots \supseteq \Omega^{N-1} \qquad (1)$$

be a sequence of nested domains. Each $\Omega^\ell \in \mathbb{R}^n$ represents the region selected to be refined at level ℓ and its boundary $\partial\Omega^\ell$ is aligned with the knot lines of V^ℓ. The proposed setup allows us to consider any kind of refinement. However, in our examples we will only focus on dyadic cell refinement. Figure 2(a) shows a nested sequence of domains, together with the corresponding knot line segments on Ω^0. Their union — shown in Figure 2(b) — will be referred to as *hierarchical mesh*.
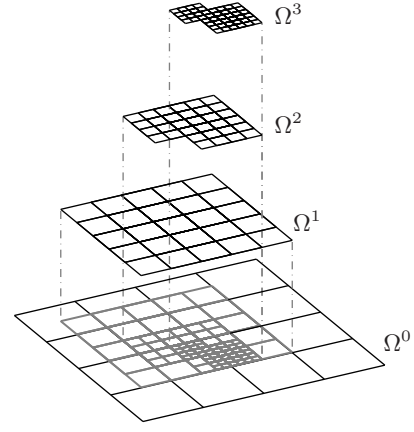
By modifying the classical support definition so that

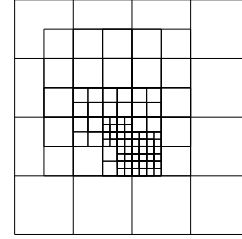$$\text{supp} f = \{(x,y) : f(x,y) \neq 0 \ \wedge \ (x,y) \in \Omega^0\}, \quad (2)$$

hierarchical B–splines can be obtained as described below.

Definition 1. *The hierarchical B–spline basis $\mathcal{H}$ is recursively constructed as follows.*

(I) Initialization: $\mathcal{H}^0 = \left\{ \beta \in \mathcal{B}^0 : \text{supp} \beta \neq \emptyset \right\}$.

(a) nested domains

(b) hierarchical mesh

Figure 2: A nested sequence of domains for the construction of the spline hierarchy according to relation (1), i.e., $\Omega^\ell \supseteq \Omega^{\ell+1}$ for $\ell = 0, \ldots, 2$, for the two–dimensional case.

(II) Recursive case: $\mathcal{H}^{\ell+1} = \mathcal{H}_A^{\ell+1} \cup \mathcal{H}_B^{\ell+1}$, *for* $\ell = 0, \ldots, N-2$, *where*

$$\mathcal{H}_A^{\ell+1} = \left\{ \beta \in \mathcal{H}^\ell : \text{supp} \beta \not\subseteq \Omega^{\ell+1} \right\},$$

and

$$\mathcal{H}_B^{\ell+1} = \left\{ \beta \in \mathcal{B}^{\ell+1} : \text{supp} \beta \subseteq \Omega^{\ell+1} \right\}.$$

(III) $\mathcal{H} = \mathcal{H}^{N-1}$.

The original definition, proposed by Kraft in [11] for the bivariate setting, imposed the following restrictions

$$\Omega^{\ell+1} = \bigcup_{\beta \in S^\ell, S^\ell \subseteq \mathcal{B}^\ell} \text{supp} \beta, \qquad \partial\Omega^\ell \cap \partial\Omega^{\ell+1} = \emptyset,$$

on the hierarchical sub–domains, i.e., $\Omega^{\ell+1}$ was defined as the union of supports of B–splines of level ℓ and characterized by disjoint boundaries with respect to Ω^ℓ. These restrictions are not required anymore by Definition 1. In particular, the modified support definition (2) — firstly considered in [20] — allows for refinement close to $\partial\Omega^0$, which was not covered by the original construction presented in [11].

Hierarchical B–splines, however, do not satisfy the *partition of unity* property. In addition, the number of overlapping basis functions associated to different hierarchical levels easily increases. This motivates the construction of another basis for the hierarchical spline space. The key idea behind the proposed approach relies on the following definition.

Definition 2. *Let $\tau \in V^\ell$ and let*

$$\tau = \sum_{\beta \in \mathcal{B}^{\ell+1}} c_\beta^{\ell+1}(\tau)\beta, \qquad c_\beta^{\ell+1} \in \mathbb{R}, \qquad (3)$$

be its representation with respect to the finer basis of $V^{\ell+1}$. The truncation of τ with respect to $\mathcal{B}^{\ell+1}$ and $\Omega^{\ell+1}$ is defined as

$$\text{trunc}^{\ell+1}\,\tau = \sum_{\beta \in \mathcal{B}^{\ell+1},\,\text{supp}\,\beta \not\subseteq \Omega^{\ell+1}} c_\beta^{\ell+1}(\tau)\beta. \qquad (4)$$

By applying the truncation mechanism to hierarchical B–splines of coarse levels, we can introduce the THB–spline basis.

Definition 3. *The truncated hierarchical B–spline basis $\mathcal{T}$ is recursively constructed as follows.*

(I) *Initialization:* $\mathcal{T}^0 = \mathcal{H}^0$.

(II) *Recursive case:* $\mathcal{T}^{\ell+1} = \mathcal{T}_A^{\ell+1} \cup \mathcal{T}_B^{\ell+1}$, *for* $\ell = 0,\dots,N-2$, *where*

$$\mathcal{T}_A^{\ell+1} = \left\{ \text{trunc}^{\ell+1}\tau : \tau \in \mathcal{T}^\ell \wedge \text{supp}\,\tau \not\subseteq \Omega^{\ell+1} \right\},$$

and $\mathcal{T}_B^{\ell+1} = \mathcal{H}_B^{\ell+1}$.

(III) $\mathcal{T} = \mathcal{T}^{N-1}$.

In the classical hierarchical construction, coarse basis functions of a certain level ℓ whose support is completely covered by finer B–splines of level $\ell+1$ (which will be added to the hierarchical basis) are replaced. For THB–splines, the replacement works as in the hierarchical case. In addition, coarse basis functions whose support has a non–empty overlap with $\Omega^{\ell+1}$ are truncated.

Example 4. The iterative constructions introduced in Definitions 1 and 3 are illustrated in Figure 3 for the one–dimensional quadratic case.

Example 5. Figure 4 shows bi–quadratic THB–splines defined over the hierarchical mesh introduced in Figure 2 for levels 1, 2, and 3. In particular, Figure 4(a) shows the 9 truncated basis functions whose support is completely contained in Ω^1 but not in Ω^2. From the 12 B–splines of level 2 whose support is completely contained in Ω^2, one will be replaced (i.e., eliminated) and the other 11 will be truncated as shown in Figure 4(b). Finally, the 20 B–splines which cover Ω^3 will be added to the truncated B–spline basis. Assuming that the domain is not further refined, there is no truncation in this case. Figure 4(c) shows three of these basis functions on the finest level.

3   Properties of the THB–spline basis

In virtue of the non–negativity property of the B–spline basis, hierarchical B–spline basis functions are also non–negative. In addition, the property of local linear independence and the nested nature of the spline spaces are inherited from the background normalized tensor–product B–splines — see Lemmas 2 and 3 in [20]. The partition of unity property, instead, is lost during the hierarchical construction. From Definition 3 we may observe that

- for each truncated basis function τ introduced at level ℓ, there exists one B–spline $\beta \in \mathcal{B}^\ell$ which satisfies

$$\tau = \text{trunc}^{N-1}\left(\text{trunc}^{N-2}\dots\left(\text{trunc}^{\ell+1}(\beta)\right)\dots\right) \quad (5)$$
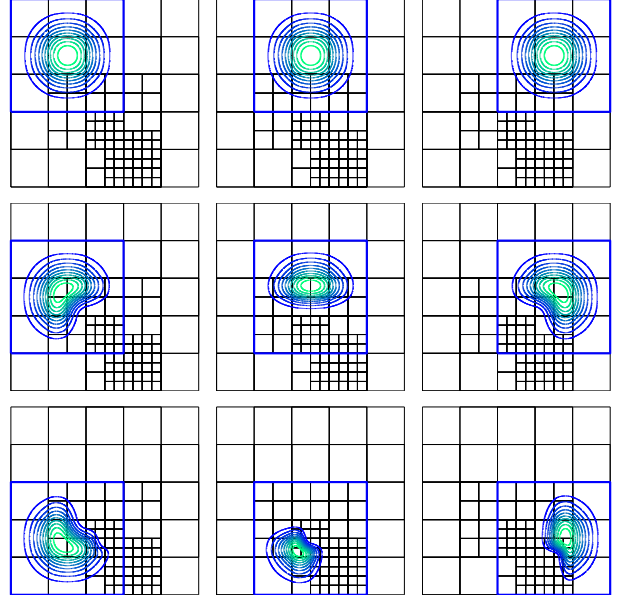
and

$$\tau\big|_{\Omega^\ell \setminus \Omega^{\ell+1}} = \beta\big|_{\Omega^\ell \setminus \Omega^{\ell+1}}; \qquad (6)$$
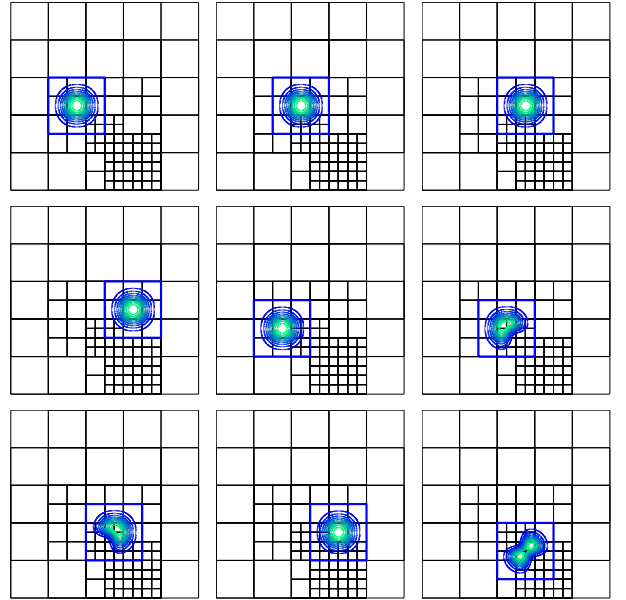
- the property of non–negativity remains true.

The B-spline β related to the truncated basis function τ as in (5) will be called the *mother* B-spline of τ and will be indicated by $\text{mot}(\tau)$. Moreover,
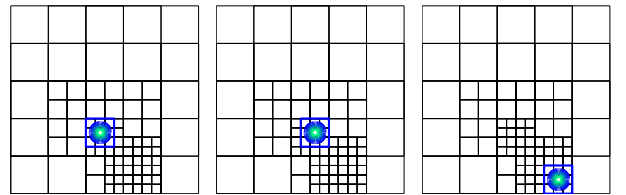
- $\text{span}\,\mathcal{H} = \text{span}\,\mathcal{T}$;

(a) bi–quadratic THB–splines of level 1

(b) bi–quadratic THB–splines of level 2

(c) some bi–quadratic THB–splines of level 3

Figure 4: Bi–quadratic THB–splines for Example 5.

(a) nested intervals of level 0,1,2, and the hierarchical mesh   (b) HB–splines of level 1 and 2   (c) THB–splines of level 1 and 2
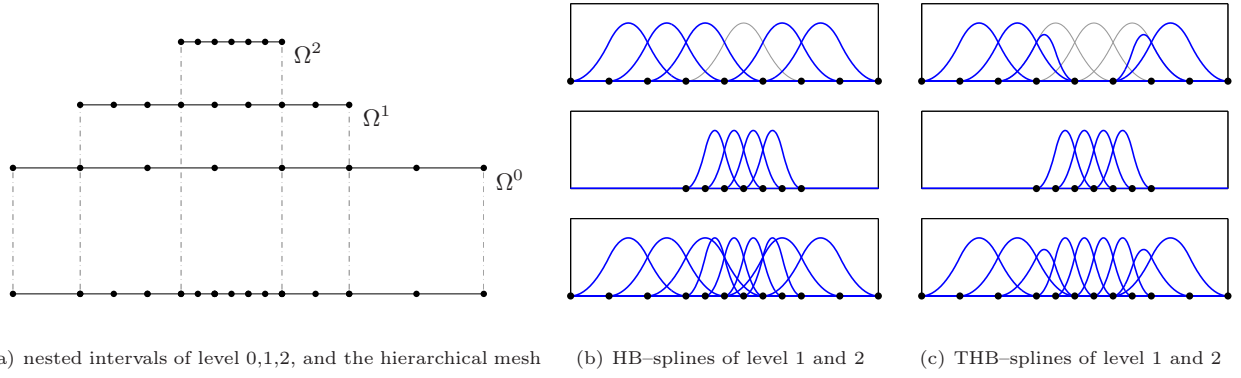
Figure 3: Univariate quadratic HB– and THB–splines defined on the hierarchical mesh shown in (a). For cases (b) and (c), top: basis functions of level 1 (B–splines of level 1 that are replaced or modified are depicted in grey lines); middle: finer basis functions of level 2; bottom: combination of basis functions from these two hierarchical levels.

- the truncated basis functions are linearly independent and form a partition of unity.

The linear independence and partition of unity will be discussed in Theorems 6 and 10, respectively. The equivalence of the two hierarchical spline spaces will be shown in Theorem 9.

Theorem 6. *The functions in $\mathcal{T}$ are linearly independent.*

Proof. We have to prove that

$$\sum_{\tau \in \mathcal{T}} c_\tau \tau = 0 \quad \Rightarrow \quad c_\tau = 0. \tag{7}$$

We may decompose the sum on the left in (7) according to the mother B-splines introduced at a certain level ℓ, namely

$$\sum_{\tau \in \mathcal{T},\, \mathrm{mot}(\tau) \in \mathcal{B}^0} c_\tau \tau + \sum_{\tau \in \mathcal{T},\, \mathrm{mot}(\tau) \in \mathcal{B}^1} c_\tau \tau + \dots$$
$$\dots + \sum_{\tau \in \mathcal{T} \cap \mathcal{B}^{N-1}} c_\tau \tau = 0. \tag{8}$$

The basis functions collected by the first sum in (8) are the only non–zero functions acting on the region given by $\Omega^0 \setminus \Omega^1$. In virtue of (5)–(6) and of the local linear independence of the B–spline basis, these functions are locally linearly independent on $\Omega^0 \setminus \Omega^1$, and thus the corresponding coefficients c_τ must be zero.

Excluding the functions already considered in this first sum, the basis functions collected by the second sum are the only non–zero functions which act on $\Omega^1 \setminus \Omega^2$. As before, also the corresponding coefficients c_τ must be zero. We can repeat the same argument for the sums related to the basis functions whose mother B-splines belong to $\mathcal{B}^l$, $l = 2, \dots, N-2$.

On the other hand, the basis functions $\tau \in \mathcal{T} \cap \mathcal{B}^{N-1}$ represent just the subset of B–splines in $\mathcal{B}^{N-1}$ whose support is completely contained in Ω^{N-1}. Hence, they are locally linearly independent on this last subdomain, and also the coefficients c_τ in the last sum must be zero. □

Lemma 7. *Let $\mathcal{H}$ and $\mathcal{T}$ be the two bases introduced in Definition 1 and 3, respectively. We have*

$$\#\mathcal{H} = \#\mathcal{T}.$$

Proof. It is easy to see that $\#\mathcal{T}_A^\ell = \#\mathcal{H}_A^\ell$ and $\#\mathcal{T}_B^\ell = \#\mathcal{H}_B^\ell$, for all $\ell = 0, \dots, N-1$. □

Lemma 8. *Let $\mathcal{T}$ be the basis introduced in Definition 3. We have*

$$\mathrm{span}\, \mathcal{T}^\ell \subset \mathrm{span}\, \mathcal{T}^{\ell+1}, \qquad \ell = 0, \dots, N-2.$$

Proof. Any $f \in \mathrm{span}\, \mathcal{T}^\ell$ can be expressed as

$$f = \sum_{\tau \in \mathcal{T}^\ell} d_\tau(f)\tau$$
$$= \sum_{\tau \in \mathcal{T}^\ell,\, \mathrm{supp}\,\tau \not\subseteq \Omega^{\ell+1}} d_\tau(f)\tau + \sum_{\tau \in \mathcal{T}^\ell,\, \mathrm{supp}\,\tau \subseteq \Omega^{\ell+1}} d_\tau(f)\tau$$
$$= \sum_{\tau \in \mathcal{T}^\ell,\, \mathrm{supp}\,\tau \not\subseteq \Omega^{\ell+1}} d_\tau(f) \left(\sum_{\beta \in \mathcal{B}^{\ell+1},\, \mathrm{supp}\,\beta \not\subseteq \Omega^{\ell+1}} c_\beta^{\ell+1}(\tau)\beta \right.$$
$$\left. + \sum_{\beta \in \mathcal{B}^{\ell+1},\, \mathrm{supp}\,\beta \subseteq \Omega^{\ell+1}} c_\beta^{\ell+1}(\tau)\beta \right)$$
$$+ \sum_{\tau \in \mathcal{T}^\ell,\, \mathrm{supp}\,\tau \subseteq \Omega^{\ell+1}} d_\tau(f) \left(\sum_{\beta \in \mathcal{B}^{\ell+1},\, \mathrm{supp}\,\beta \subseteq \Omega^{\ell+1}} c_\beta^{\ell+1}(\tau)\beta \right)$$
$$= \sum_{\tau \in \mathcal{T}^\ell,\, \mathrm{supp}\,\tau \not\subseteq \Omega^{\ell+1}} d_\tau(f)\, \mathrm{trunc}^{\ell+1}\tau$$
$$+ \sum_{\beta \in \mathcal{B}^{\ell+1},\, \mathrm{supp}\,\beta \subseteq \Omega^{\ell+1}} \left(\sum_{\tau \in \mathcal{T}^\ell} d_\tau(f) c_\beta^{\ell+1}(\tau) \right) \beta$$
$$= \sum_{\tau \in \mathcal{T}_A^{\ell+1}} d_\tau(f)\, \mathrm{trunc}^{\ell+1}\tau + \sum_{\beta \in \mathcal{T}_B^{\ell+1}} d_\beta(f)\beta,$$

where $c_\beta^{\ell+1}(\tau)$ has been introduced in (3)–(4). The first sum in the last line belongs to the span of $\mathcal{T}_A^{\ell+1}$, while the second sum to the one of $\mathcal{T}_B^{\ell+1}$. Hence, $f \in \mathrm{span}\, \mathcal{T}^{\ell+1}$. □

Theorem 9. *Let $\mathcal{H}$ and $\mathcal{T}$ be the bases introduced in Definitions 1 and 3, respectively. We have*

$$\mathrm{span}\, \mathcal{H} = \mathrm{span}\, \mathcal{T}.$$

Proof. By induction on ℓ, we prove that

$$\mathrm{span}\, \mathcal{H}^\ell = \mathrm{span}\, \mathcal{T}^\ell \quad \Rightarrow \quad \mathrm{span}\, \mathcal{H}^{\ell+1} = \mathrm{span}\, \mathcal{T}^{\ell+1}.$$

For $\ell = 0$ it follows directly from Definition 3. By considering Definition 1, the inductive hypothesis, and Lemma 8, we may observe that

$$\mathrm{span}\, \mathcal{H}_A^{\ell+1} \subseteq \mathrm{span}\, \mathcal{H}^\ell = \mathrm{span}\, \mathcal{T}^\ell \subset \mathrm{span}\, \mathcal{T}^{\ell+1},$$

4

while, by Definition 3 again,

$$\operatorname{span}\mathcal{H}_B^{\ell+1} = \operatorname{span}\mathcal{T}_B^{\ell+1} \subseteq \operatorname{span}\mathcal{T}^{\ell+1}.$$

Then

$$\operatorname{span}\mathcal{H}_A^{\ell+1} \cup \operatorname{span}\mathcal{H}_B^{\ell+1} \subseteq \operatorname{span}\mathcal{T}^{\ell+1},$$

i.e.,

$$\operatorname{span}\mathcal{H}^{\ell+1} \subseteq \operatorname{span}\mathcal{T}^{\ell+1}.$$

In view of the linear independence of the two bases and of Lemma 7, we may then conclude that $\operatorname{span}\mathcal{H}^{\ell+1} = \operatorname{span}\mathcal{T}^{\ell+1}$. □

Theorem 10. *The truncated hierarchical B–spline basis $\mathcal{T}$ forms a partition of unity:*

$$\sum_{\tau \in \mathcal{T}^\ell} \tau = 1 \quad on \quad \Omega^0, \qquad \ell = 0, 1, \dots, N-1.$$

Proof. We recall that

$$\sum_{\beta \in \mathcal{B}^\ell} \beta = 1 \quad on \quad \Omega^0, \qquad \ell = 0, 1, \dots, N-1. \quad (9)$$

The partition of unity can be shown by induction on the hierarchical level ℓ. The base case simply follows from (9) with $\ell = 0$. The inductive step

$$\sum_{\tau \in \mathcal{T}^\ell} \tau = 1 \quad on \quad \Omega^0 \quad \Rightarrow \quad \sum_{\tau \in \mathcal{T}^{\ell+1}} \tau = 1 \quad on \quad \Omega^0$$

can be proved by using (3) and re–arranging the sums as follows.

$$1 = \sum_{\tau \in \mathcal{T}^\ell} \tau = \sum_{\tau \in \mathcal{T}^\ell} \sum_{\beta \in \mathcal{B}^{\ell+1}} c_\beta^{\ell+1}(\tau)\,\beta$$

$$= \sum_{\tau \in \mathcal{T}^\ell} \left(\sum_{\beta \in \mathcal{B}^{\ell+1},\, \operatorname{supp}\beta \not\subseteq \Omega^{\ell+1}} c_\beta^{\ell+1}(\tau)\,\beta \right.$$

$$\left. + \sum_{\beta \in \mathcal{B}^{\ell+1},\, \operatorname{supp}\beta \subseteq \Omega^{\ell+1}} c_\beta^{\ell+1}(\tau)\,\beta \right)$$

$$= \sum_{\tau \in \mathcal{T}^\ell} \left(\sum_{\beta \in \mathcal{B}^{\ell+1},\, \operatorname{supp}\beta \not\subseteq \Omega^{\ell+1}} c_\beta^{\ell+1}(\tau)\,\beta \right)$$

$$+ \sum_{\beta \in \mathcal{B}^{\ell+1},\, \operatorname{supp}\beta \subseteq \Omega^{\ell+1}} \left(\sum_{\tau \in \mathcal{T}^\ell} c_\beta^{\ell+1}(\tau) \right) \beta. \quad (10)$$

The term in brackets in the first sum of the last equation is just $\operatorname{trunc}^{\ell+1}(\tau)$. Moreover, by (9) and by swapping the order of sums in the first line of (10), we have

$$\sum_{\beta \in \mathcal{B}^{\ell+1}} \beta = 1 \quad \text{as well as} \quad 1 = \sum_{\beta \in \mathcal{B}^{\ell+1}} \left(\sum_{\tau \in \mathcal{T}^\ell} c_\beta^{\ell+1}(\tau) \right) \beta$$

on Ω^0. By comparing coefficients and by the linear independence of the B-splines we can conclude that

$$\sum_{\tau \in \mathcal{T}^\ell} c_\beta^{\ell+1}(\tau) = 1,$$

for all $\beta \in \mathcal{B}^{\ell+1}$ and in particular for each β such that $\operatorname{supp}\beta \subseteq \Omega^{\ell+1}$. From (10), in view of Definition 3, we then obtain

$$1 = \sum_{\tau \in \mathcal{T}^\ell} \operatorname{trunc}^{\ell+1}(\tau) + \sum_{\beta \in \mathcal{B}^{\ell+1},\, \operatorname{supp}\beta \subseteq \Omega^{\ell+1}} \beta$$

$$= \sum_{\tau \in \mathcal{T}_A^{\ell+1}} \tau + \sum_{\tau \in \mathcal{T}_B^{\ell+1}} \tau = \sum_{\tau \in \mathcal{T}^{\ell+1}} \tau.$$

□

In view of the non–negativity of truncated basis functions and of the previous theorem, the functions in $\mathcal{T}$ form a convex partition of unity. Moreover, Theorem 9 allows us to apply the characterization of bivariate hierarchical B–spline spaces recently provided in [7] to the bivariate THB–spline model. By considering certain mild assumptions on the configuration of the hierarchical mesh, single knots, dyadic refinement, and $\partial\Omega^\ell$ aligned with the knot lines of $V^{\ell-1}$, the bivariate truncated hierarchical B–spline basis of bi–degree (d, d) covers the space of piecewise polynomials of the same degree which are C^{d-1} smooth on the considered domain — see [7] for the details.

4 Surface approximation with THB–splines

In this section we compare the local and adaptive behavior of hierarchical and truncated hierarchical B–splines for the solution of a classical approximation problem. A least–squares fitting is iteratively computed after the execution of a refinement procedure until the approximation error is satisfactory with respect to a given tolerance. Once the degree d in each direction of the background tensor–product B–spline basis has been chosen, the approximation procedure can be summarized as follows.

Hierarchical spline approximation

    Input: an initial domain Ω^0, together with a data set $(\mathbf{p}_i, f_i)$, $i = 0, \dots, m$, which associates real values f_i to a set of points $\mathbf{p}_i \in \Omega^0$, a certain tolerance ϵ, and a maximum value for the number of iteration steps $s_{\max}$.

    Algorithm:

    (a) define an initial tensor–product space V^0 on Ω^0, and initialize $s = 0$;

    (b) compute the least–squares approximation $f = \sum_{\beta \in \mathcal{H}} c_\beta \beta = \sum_{\tau \in \mathcal{T}} c_\tau \tau$, which minimizes

$$\sum_{i=0}^m (f_i - f(\mathbf{p}_i))^2$$

    according to the current hierarchical spline space;

    (c) if $\max|f_i - f(\mathbf{p}_i)| > \epsilon$ and $s < s_{\max}$, then set $s = s + 1$, and apply the hierarchical refinement according to the considered cell selection strategy, and repeat step (b).

    Output: the least–squares approximation f expressed in terms of HB–splines and THB–splines so that $\max|f_i - f(\mathbf{p}_i)| \leq \epsilon$ or $s = s_{\max}$.

Step (c) of the above algorithm can be described by the following *black–box* procedure — possible subdomain enlargement strategies will be discussed later on.

Hierarchical refinement

    Input: the current hierarchical sequence of nested domains (1), together with the regions marked for further refinement.

    Output: a (possibly enlarged) nested sequence of domains $\hat{\Omega}^0 \supseteq \hat{\Omega}^1 \supseteq \dots \supseteq \hat{\Omega}^N$ so that

$$\hat{\Omega}^0 \equiv \Omega^0, \qquad \hat{\Omega}^\ell \supseteq \Omega^\ell, \qquad \ell = 0, \dots, N-1,$$

    and the corresponding hierarchical sequence of bases $\hat{\mathcal{H}}^\ell$ and $\hat{\mathcal{T}}^\ell$, $\ell = 0, \dots, N$, constructed according to Definitions 1 and 3.

As already observed in [20], the nested nature of the spline space is preserved by the nested (enlargement of) domains constructed by the refinement procedure, namely span $\mathcal{H}^\ell \subseteq$ span $\hat{\mathcal{H}}^\ell$, for any level $\ell = 0, \ldots, N-1$. In combination with a suitable refinement strategy, this a priori knowledge automatically ensures that approximations defined on refined grids of a given function converge to that function with an increased level of details.

The good approximation properties of the truncated hierarchical basis with respect to classical hierarchical B–splines are illustrated in the two following bivariate examples. We only consider single knots and quadratic degree in both directions.

Example 11. We computed the least–squares approximations of the function

$$f = \frac{2}{3\exp(\sqrt{(10x-3)^2 + (10y-3)^2})}$$
$$+ \frac{2}{3\exp(\sqrt{(10x+3)^2 + (10y+3)^2})}$$
$$+ \frac{2}{3\exp(\sqrt{(10x)^2 + (10y)^2})},$$

shown on the left of Figure 5, by sampling the data points on a 150×150 uniform grid defined on the domain $[-1,1] \times [-1,1]$. Figure 6 collects several locally refined hierarchical meshes together with the corresponding bi–quadratic THB–spline approximations. Tables 1 and 2 show the comparison between the HB– and THB–spline approximations with respect to sparsity and condition number, respectively, of the matrices used to compute the least–squares approximations.
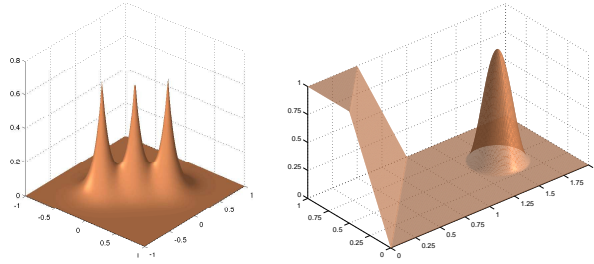

Figure 5: Functions considered in Examples 11–12 (left) and 13 (right).

step	HB	THB	THB/HB
0	576	576	100.00%
1	1,496	1,312	87.70%
2	3,380	2,696	79.76%
3	8,288	5,506	66.43%
4	24,058	13,492	56.08%
5	35,938	18,412	51.23%

Table 1: Number of non–zero matrix elements for the two matrices associated with the HB– and THB–spline least–squares approximations ($d = 2$) in Example 12.

Example 12. In order to have a closer look at the structure of the linear systems which characterize the considered approximation problem, we applied two reordering algorithms to the matrices obtained at the last refinement step in Example 11. The first one is the *symmetric reverse Cuthill–McKee algorithm* which permutes the rows and columns of a given matrix so that the non–zero elements are closer to the diagonal. The second one is the

step	HB	THB	THB/HB
0	1.938e+03	1.938e+03	100.00%
1	5.989e+03	4.037e+03	67.41%
2	4.879e+03	2.720e+03	55.75%
3	4.973e+03	2.772e+03	55.74%
4	9.599e+03	4.214e+03	43.90%
5	3.893e+04	1.747e+04	44.88%

Table 2: Condition number of the two matrices associated with the HB– and THB–spline least–squares approximations ($d = 2$) in Example 12.

symmetric approximate minimum degree permutation algorithm which, instead, computes a suitable permutation of rows and columns of a given matrix so that the non–zero elements of the Cholesky factor are reduced. The original matrices together with the two reordered versions are shown in Figure 7. We may observe that the Cuthill–McKee reordering performs particularly better when applied to the matrix associated with the THB–splines basis rather than to the one associated with hierarchical B–splines — see the two central plots of Figure 7. Figure 8 illustrates the corresponding Cholesky factorizations, and the related comparison with respect to sparsity is summarized in Table 3.

matrix	HB	THB	THB/HB
org	176,999	90,861	51.33%
rcm	95,136	75,240	79.09%
amd	38,252	28,925	75.62%

Table 3: Number of non–zero elements of the Cholesky factors considered in Example 12. The data related to the original factors (org) are shown together with the data of the Cholesky factors computed after the reordering according to the reverse Cuthill–McKee (rcm) algorithm, and to the approximate minimum degree (amd) algorithm.

By focusing again on the bivariate case with single knots, we now investigate the following three particular strategies for the construction of the sequence $\{\hat{\Omega}^\ell\}_{\ell=0,\ldots,N}$. Let $\Phi^0 = \emptyset$, and let Φ^ℓ be the region spanned by the cells of level $\ell-1$ selected to be refined by the error estimator, for $\ell = 1, \ldots, N$.

(S1) Each $\hat{\Omega}^\ell$ is the union of Ω^ℓ and Φ^ℓ.

(S2) Each $\hat{\Omega}^\ell$ is given by the domain constructed in (S1) plus an offset ring around Φ^ℓ of $\lceil \frac{d-1}{4} \rceil$ cells with respect to the knot mesh of V^ℓ in the horizontal, vertical, and diagonal directions.

(S3) Each $\hat{\Omega}^\ell$ is the union of disjoint boxes composed by $d \times d$ cells with respect to the knot mesh of $V^{\ell-1}$ containing the domain constructed in (S1).[1]

Obviously the first strategy may be too naive. It can create refined (isolated) regions that will not enrich the hierarchical spline space, because no additional finer basis functions act on these regions. The aim of strategies (S2) and (S3) is to identify refined regions which are large enough to ensure an improved hierarchical spline space. The width of the offset ring considered in strategy (S2) comes from the dyadic refinement we are considering. For example, an offset ring of one cell in each direction around any (isolated) cell marked by the error estimator guarantees to add at least one finer basis function of degree $d = 2, \ldots, 5$. The trade–off between the locality of the refinement and the

---

[1]This refinement strategy was proposed in [7].

(a) step 0: dimension = 36, max error = 4.493e-01

(b) step 1: dimension = 64, max error = 3.877e-01

(c) step 2: dimension = 112, max error = 2.223e-01

(d) step 3: dimension = 190, max error = 1.153e-01

(e) step 4: dimension = 456, max error = 3.047e-02

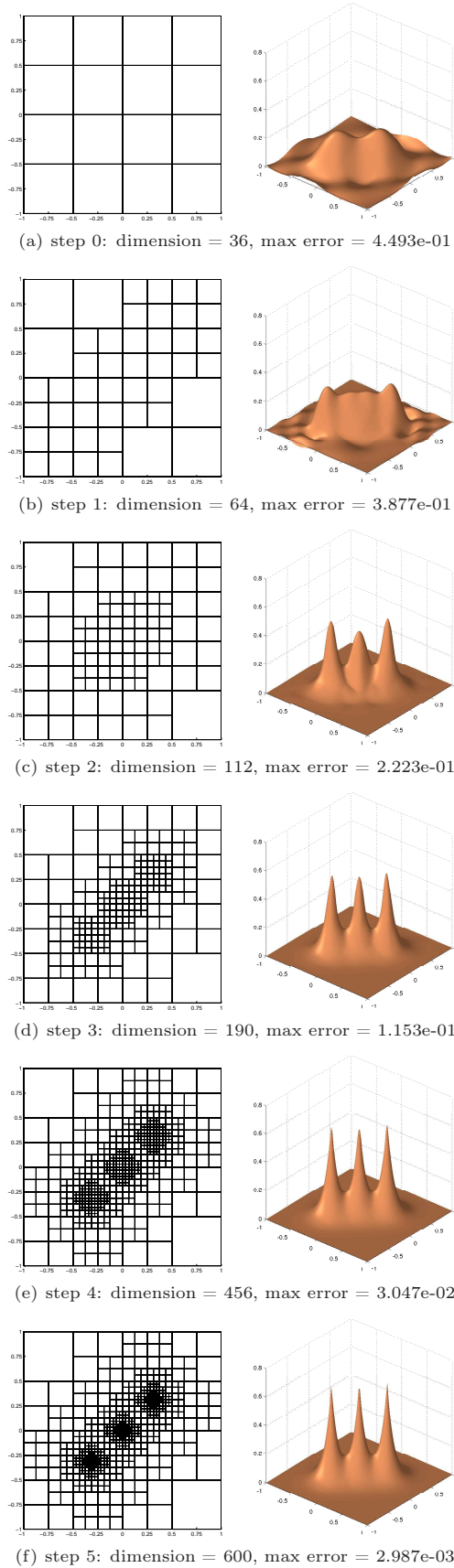(f) step 5: dimension = 600, max error = 2.987e-03

Figure 6: Locally refined hierarchical meshes and corresponding THB–spline least–squares approximations ($d = 2$) to the function shown on the left of Figure 5 and considered in Example 11.
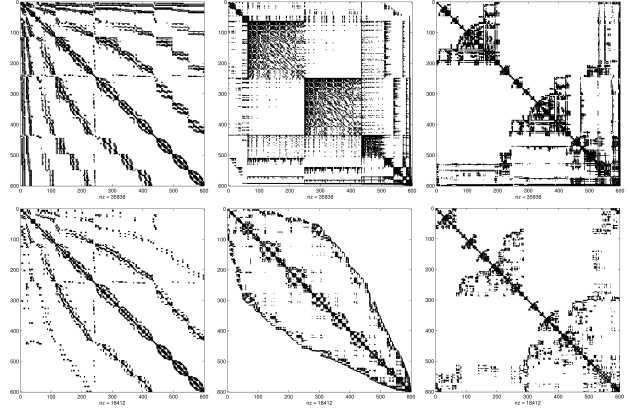

Figure 7: Structure of the matrices associated with the HB–spline basis (top) and the THB–spline basis (bottom) in Example 12. Left: original matrices; center: Cuthill–McKee reordering; right: approximate minimum degree permutation.
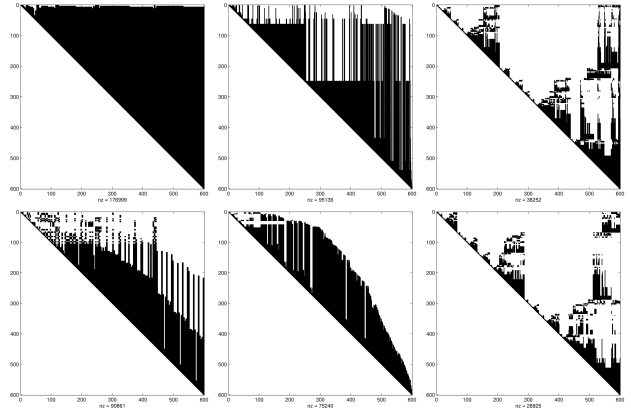

Figure 8: Cholesky factorizations of the matrices associated with the HB–spline basis (top) and the THB–spline basis (bottom) in Example 12. Left: original matrices; center: after Cuthill–McKee reordering; right: after approximate minimum degree permutation.

enlargement of $\hat{\Omega}^\ell$ may influence the quality of the final approximation.

In the last example we compare these three refinement strategies for a given test function.

Example 13. We sampled the data points on a 150×150 uniform grid from the function[2]

$$
f(x,y) = \begin{cases} 1, & y - x > 1/2, \\ 2(y - x), & 0 \leq y - x \leq 1/2, \\ 1/2 \cos(4\pi\sqrt{q(x,y)}) + 1/2, & q(x,y) \leq 1/16, \\ 0, & \text{otherwise,} \end{cases}
$$

where $q(x,y) = (x - 3/2)^2 + (y - 1/2)^2$, defined on the domain $[0,2] \times [0,1]$ — see Figure 5 on the right. Figure 9 presents the results obtained considering the refinement strategies mentioned above with $\epsilon = 5\mathrm{e}{-}3$, $s_{\max} = 4$, and $d = 2$. For each strategy, the hierarchical meshes obtained after $s = 4$ are shown together with a visualization of the error on the domain. We may observe that only with strategy (S2) the approximation error is less than the given tolerance ϵ. In each case the truncated basis performs better than the HB–spline basis.

---

[2]This test function is taken from [10, 15].

(a) strategy (S1) — max error: 8.762e-03, THB/HB ratio with respect to sparsity: 86.23%, and to condition number: 84.67%

(b) strategy (S2) — max error: 4.062e-03, THB/HB ratio with respect to sparsity: 76.37%, and to condition number: 83.53%

(c) strategy (S3) — max error: 9.415e-03, THB/HB ratio with respect to sparsity: 81.17%, and to condition number: 90.29%
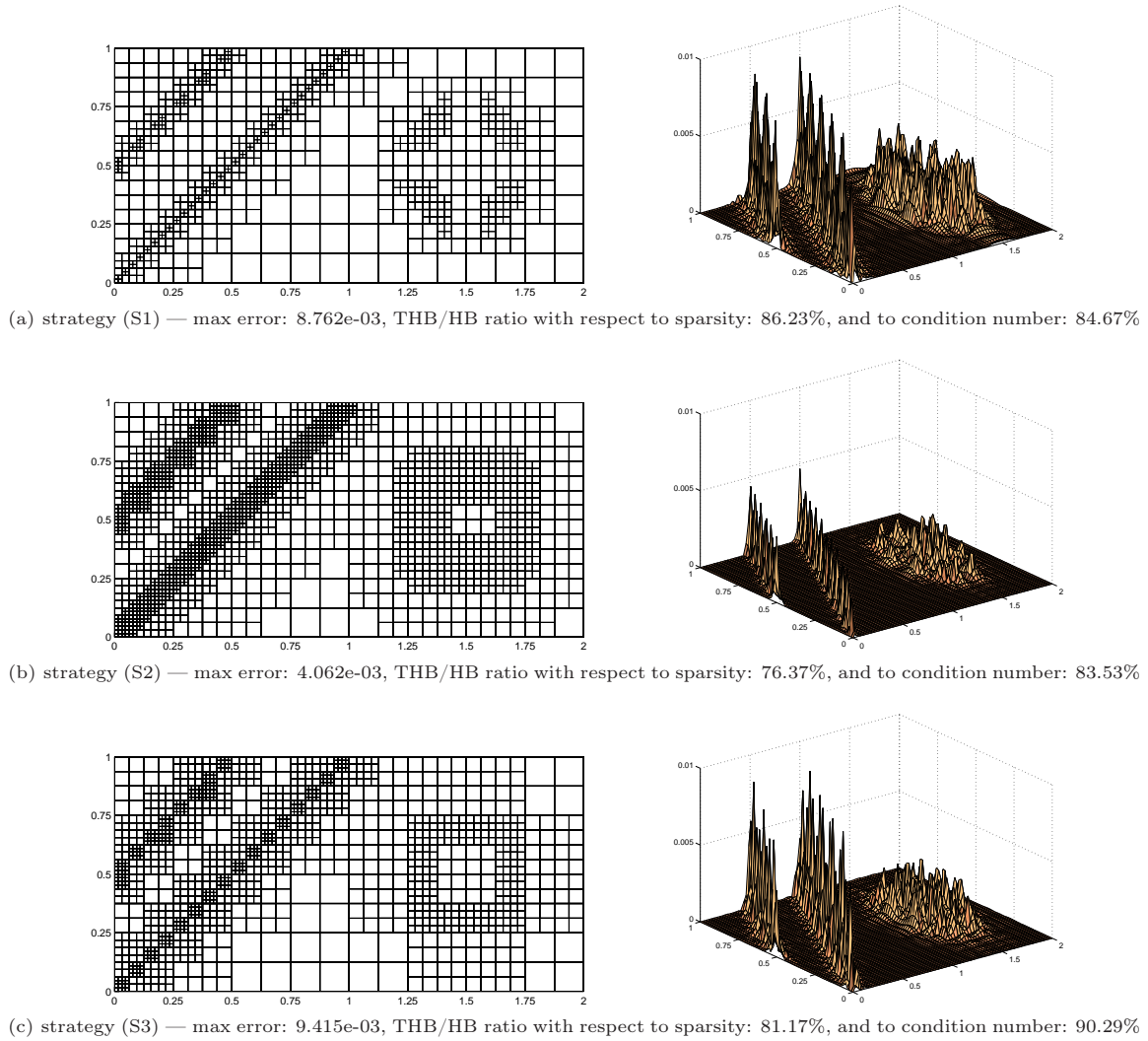
Figure 9: Locally refined hierarchical meshes (left) and approximation errors (right) obtained with the refinement strategies (S1), (S2), and (S3) for the function shown on the right of Figure 5 and considered in Example 13. The comparisons THB/HB with respect to sparsity and condition number of the corresponding matrices are also indicated.

5   Conclusion

A new normalized basis for multivariate hierarchical spline spaces has been proposed, and its performance for the computation of bivariate adaptive approximations has been tested through a selection of computed examples. The theoretical framework that modifies the hierarchical B–spline basis functions in order to define the THB–spline basis was introduced. Furthermore, the equivalence with hierarchical B–spline spaces, and the properties of linear independence and partition of unity were proved. The advantages of exploiting the truncation mechanism with respect to the original HB–spline model are illustrated in the context of data fitting.

The hierarchical approximation scheme can be enhanced to facilitate the modeling of detailed features by using some kind of *fairness* functionals as suggested in [6, 8]. Alternative error estimators and refinement strategies can also be considered to further improve the convergence rate, as well as the shape qualities of the resulting surface. Moreover, additional studies, such as a theoretical analysis of stability and related issues will be interesting subjects of future research. Finally, the n–variate THB–spline framework is well suited for related applications in

isogeometric analysis.

References

[1] A. Buffa, D. Cho, and G. Sangalli. Linear independence of the T-spline blending functions associated with some particular T-meshes. *Comput. Methods Appl. Mech. Engrg.*, 199:1437–1445, 2010.

[2] J. Deng, F. Chen, X. Li, C. Hu, T. W., Z. Yang, and Y. Feng. Polynomial splines over hierarchical T-meshes. *Graph. Models* , 70:76–86, 2008.

[3] M. R. Dörfel, B. Jüttler, and B. Simeon. Adaptive isogeometric analysis by local h-refinement with T-splines. *Comput. Methods Appl. Mech. Engrg.*, 199:264–275, 2010.

[4] D. R. Forsey and R. H. Bartels. Hierarchical B-spline refinement. *Comput. Graphics*, 22:205–212, 1988.

[5] D. R. Forsey and R. H. Bartels. Surface fitting with hierarchical splines. *ACM Trans. Graphics*, 14:134–161, 1995.

[6] D. R. Forsey and D. Wong. Multiresolution surface reconstruction for hierarchical B-splines. In W. A. Davis, K. S. Booth, and A. Fournier, editors, *Graphics Interface*, pages 57–64. Canadian Human-Computer Communications Society, 1998.

[7] C. Giannelli and B. Jüttler. Bases and dimensions of bivariate hierarchical tensor–product splines. Tech-

nical Report 2011–11, DK–Report, Johannes Kepler University Linz (available at https://www.dk-compmath.jku.at/publications/dk-reports/2011-10-04/), 2011.

[8] G. Greiner and K. Hormann. Interpolating and approximating scattered 3D-data with hierarchical tensor product B-splines. In A. L. Méhauté, C. Rabut, and L. L. Schumaker, editors, *Surface Fitting and Multiresolution Methods*, Innovations in Applied Mathematics, pages 163–172. Vanderbilt University Press, Nashville, TN, 1997.

[9] S. K. Kleiss, B. Jüttler, and W. Zulehner. Enhancing isogeometric analysis by a finite element–based local refinement strategy. *Comput. Methods Appl. Mech. Engrg.*, 213–216:168–182, 2012.

[10] V. P. Kong, B. H. Ong, and K. H. Saw. Range restricted interpolation using cubic Bézier triangles. In *Proceedings of Computer Graphics, Visualization and Computer Vision*, pages 125–132. Science Press, Union Agency, Plzen, 2004.

[11] R. Kraft. Adaptive and linearly independent multilevel B–splines. In A. Le Méhauté, C. Rabut, and L. L. Schumaker, editors, *Surface Fitting and Multiresolution Methods*, pages 209–218. Vanderbilt University Press, Nashville, 1997.

[12] X. Li, J. Deng, and F. Chen. Surface modeling with polynomial splines over hierarchical T-meshes. *Visual Comput.*, 23:1027–1033, 2007.

[13] X. Li, J. Zheng, T. W. Sederberg, T. J. R. Hughes, and M. A. Scott. On Linear Independence of T-spline blending functions. *Comput. Aided Geom. Design*, 29:63–76, 2012.

[14] N. Nguyen-Thanh, H. Nguyen-Xuan, S. P. A. Bordas, and T. Rabczuk. Isogeometric analysis using polynomial splines over hierarchical T-meshes for two-dimensional elastic solids. *Comput. Methods Appl. Mech. Engrg.*, 200:1892–1908, 2011.

[15] L. L. Schumaker and H. Speleers. Nonnegativity preserving macro-element interpolation of scattered data. *Comput. Aided Geom. Design*, 27:245–261, 2010.

[16] M. A. Scott, X. Li, T. W. Sederberg, and T. J. R. Hughes. Local refinement of analysis-suitable T-splines. *Comput. Methods Appl. Mech. Engrg.*, 213–216:206–222, 2012.

[17] T. W. Sederberg, D. L. Cardon, G. T. Finnigan, N. S. North, J. Zheng, and T. Lyche. T-spline simplification and local refinement. *ACM Trans. Graphics*, 23:276–283, 2004.

[18] T. W. Sederberg, J. Zheng, A. Bakenov, and A. Nasri. T-splines and T-NURCCS. *ACM Trans. Graphics*, 22:477–484, 2003.

[19] H. Speleers, P. Dierckx, and S. Vandewalle. Quasi–hierarchical Powell–Sabin B–splines. *Comput. Aided Geom. Design*, 26:174–191, 2009.

[20] A.-V. Vuong, C. Giannelli, B. Jüttler, and B. Simeon. A hierarchical approach to adaptive local refinement in isogeometric analysis. *Comput. Methods Appl. Mech. Engrg.*, 200:3554–3567, 2011.