A Characterization of Linear Independence of THB-Splines in $\mathbb{R}^n$ and Application to Bézier Projection

Kevin W. Dijkstra and Deepesh Toshniwal

1 Introduction

In recent years, isogeometric analysis [16] has been an active topic of research in numerical mathematics. Using higher regularity finite dimensional spaces for the finite element method allows for better approximation power per degree of freedom [1, 24]. Additionally, this allows the domain to be more accurately imported from Computer Aided Design software, which results in a reduction/elimination of domain meshing errors [7, 16]. For this reason, B-splines are commonly used as basis functions. However, B-splines cannot be refined locally. Different types of splines spaces have been introduced that can be locally refined, for example T-splines [5, 26], LR-splines [4, 9, 17, 22], S-splines [18] and HB-splines [12, 23, 30]. HB-splines have later been modified in [13] to form the Truncated Hierarchical B-splines (THB-splines), a spline space that has the partition of unity property, and where the basis functions have smaller support compared to the HB-splines. While B-splines, (T)HB-splines [21], T-splines [5], LR-splines [4, 22] and S-splines [18] are all linearly independent, only B-splines and—under suitable assumptions on the underlying mesh—T-splines and LR-splines are locally linearly independent. This paper considers the space of THB-splines, characterizes the pseudo-local linear independence of the THB-spline basis on certain macro elements, and uses that to introduce a local projector from $L^2(\Omega)$ onto the spline space.

Such projectors are of great value for numerical analysis and scientific computing. For example, they can be used for curve and surface fitting, enforcement of boundary conditions, solution methods with non-conforming meshes, multi-level solver and data compression for image, signal and data processing.

K. W. Dijkstra (✉) · D. Toshniwal
Delft Institute of Applied Mathematics, Delft University of Technology, Delft, The Netherlands
e-mail: k.w.dijkstra@tudelft.nl

D. Toshniwal
e-mail: d.toshniwal@tudelft.nl

M. Lanini et al. (eds.), *Approximation Theory and Numerical Analysis Meet Algebra, Geometry, Topology*, Springer INdAM Series 60,
https://doi.org/10.1007/978-981-97-6508-9_6

Hierarchical spline fitting has been an active topic of research since the introduction of Hierarchical B-splines. Global fitting methods have been investigated in [15] while quasi-interpolants have first been extended to hierarchical splines in [23]. In [27, 28], efficient quasi-interpolation has been introduced for THB-splines. In particular the latter sets up a general framework to construct quasi-interpolants in the Sobolev spaces W_q^{p+1}, $1 \leq q \leq \infty$. However, only explicitly constructed interpolates from the space of continuous functions $C(\Omega)$ onto the THB-spline space can be actually found in these two works. More recently, a different local THB-spline projector has been introduced in [14] that requires less function evaluations per degree of freedom then the projector of [28] while being slightly more accurate. Lastly, in [11, 19], a (T)HB-spline approximant that uses the Bézier projection philosophy was proposed in [29]. Note that this approximant is not a projector as it fails to preserve THB-splines.

We introduce a Bézier projector for considered THB-spline spaces. This projector is an extension of the Bézier projector introduced by [29] for B-splines and analysis-suitable T-splines. This local projector consists of two steps. For all mesh elements, an initial local L^2 projection onto the local polynomial space on that element. All of these element-wise projections together define a global C^{-1} approximation. Then, these C^{-1} projections are smoothed to produce a global spline projection. In this second step, local linear independence of the THB-splines is required for every mesh element. Unfortunately, this property is not verified in general by THB-splines. In this paper, under suitable assumptions on the mesh such as graded refinements, we construct local collections of mesh elements, called *projection elements*, such that the THB-splines are linearly independent over them. These collections are local in the sense that they consist of adjacent mesh elements. Furthermore, the number of mesh elements included in a projection element is only dependent on the spline degree. To be able to talk about pseudo-local linear independence of functions on collections of mesh elements, we introduce the following definition of overloading.

Definition 1.1 For a given set of functions, $\mathbb{S}$, defined on Ω, we say that $\widetilde{\Omega} \subset \Omega$ is **overloaded** if the functions supported on it are linearly dependent; else, $\widetilde{\Omega}$ is not overloaded.

Therefore, the projection elements that we will introduce will not be overloaded for THB-splines, allowing the formulation of a Bézier projection. See Fig. 1 for an example of application of the projector. Various assumptions are required to construct these projection elements. Some of them appear in the literature, for example, mesh grading [6], others are new. One of these new assumptions is non-constructive in the sense that if a mesh violates it, it is unclear which mesh modifications should be performed to satisfy it. Therefore, we also provide a set of stronger and constructive assumptions in two dimensions for quadratic and cubic THB-splines. They are used to formulate a first adaptive refinement scheme.

The remaining sections of the paper are organized as follows. Section 2 recaps the Bézier projector from [29] for multivariate B-splines. In Sect. 3 we introduce non-

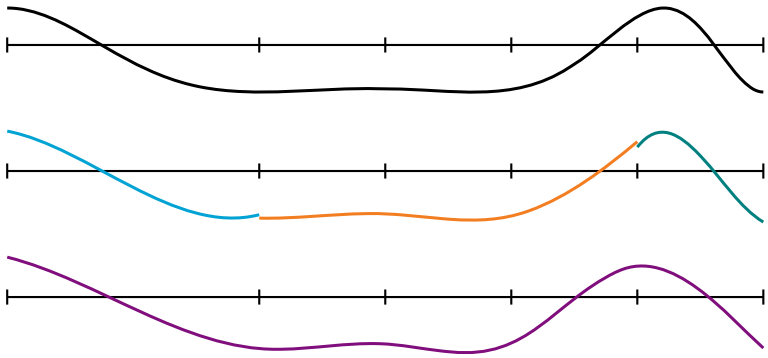


Fig. 1 An initial projection onto projection elements, that is, suitable local collections of mesh elements, and a global smoothing step to produce a global projection of the target onto a THB-spline space

overloaded macro-elements that are used in Sect. 4 to formulate the Bézier projector for THB-splines. In Sect. 5 the local error estimates are validated numerically, and the adaptive refinement routine is compared to [14].

2 B-Splines and Bézier Projection

In this section, B-splines will be briefly introduced and a small recap of [29] will be given on Bézier projection for multivariate B-splines. For an introductory text on B-splines, we refer the reader to [20].

2.1 Univariate B-Splines

Consider the unit domain $\Omega = (0, 1)$, a polynomial degree $p \in \mathbb{N}_0$, and an increasing sequence of $m \in \mathbb{N}$ real numbers over Ω as follows:

$$\begin{aligned} \boldsymbol{\xi} &:= \{\xi_1 \leq \xi_2 \leq \cdots \leq \xi_m\}, \\ &= \{\underbrace{0, \ldots, 0}_{p+1 \text{ times}} < \xi_{p+2} < \cdots < \xi_{m-p-1} < \underbrace{1, \ldots, 1}_{p+1 \text{ times}}\}. \end{aligned} \tag{1}$$

Such a sequence is called a $(p + 1)$-open **knot sequence**, where every individual value is called a **knot**; note that we assume that all internal knots are distinct. This knot sequence describes a partition of Ω, where the unique knots indicate element boundaries for the partition. For a given knot, the number of occurrences is denoted by its multiplicity. Thus, for our purposes, every knot either has multiplicity of 1 or

$p+1$. By convention, all non-empty open **knot intervals** (ξ_k, ξ_{k+1}) constitute the mesh elements.

For the knot sequence $\boldsymbol{\xi}$, we can define $M := m - p - 1$ **B-splines** recursively,

$$B_{j,p,\boldsymbol{\xi}} := \frac{x - \xi_j}{\xi_{j+p} - \xi_j} B_{j,p-1,\boldsymbol{\xi}} + \frac{\xi_{j+p+1} - x}{\xi_{j+p+1} - \xi_{j+1}} B_{j+1,p-1,\boldsymbol{\xi}}, \ j = 1, \ldots, M. \quad (2)$$

Here we assume the fractions to be zero when the denominator is zero. The base case $p = 0$ is defined by the following unit functions:

$$\begin{aligned} B_{j,0,\boldsymbol{\xi}} &:= \begin{cases} 1 & \text{if } x \in [\xi_j, \xi_{j+1}), \\ 0 & \text{otherwise,} \end{cases} \qquad \text{for} j \neq m - p - 1, \\ B_{m-p-1,0,\boldsymbol{\xi}} &:= \begin{cases} 1 & \text{if } x \in [\xi_{m-p-1}, 1], \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \quad (3)$$

The functions $B_{j,p,\boldsymbol{\xi}}$ are non-negative, form a partition of unity on $\bar{\Omega}$ and the collection of B-splines with support on any given mesh element $\Omega^e = (\xi_k, \xi_{k+1})$ is linearly independent and can reproduce any polynomial of degree p on Ω^e. Therefore, none of the mesh elements are overloaded with respect to B-splines. The span of the B-spline set $\mathcal{B}_{p,\boldsymbol{\xi}}$ is called the B-spline space and is denoted as:

$$\mathbb{B}_{p,\boldsymbol{\xi}} := \text{span}\left\{\mathcal{B}_{p,\boldsymbol{\xi}}\right\}. \quad (4)$$

With the spline degree p fixed, we will omit it from the notations to simply write $B_{j,\boldsymbol{\xi}}$, $\mathcal{B}_{\boldsymbol{\xi}}$ and $\mathbb{B}_{\boldsymbol{\xi}}$.

2.2 Multivariate B-Splines

In the case of a multivariate domain $\Omega = (0, 1)^n$, where n denotes the dimension, let the vector $\mathbf{p} := (p^1, \ldots, p^n)$ denote the polynomial degrees per dimension and let $\Xi = (\boldsymbol{\xi}^1, \ldots, \boldsymbol{\xi}^n)$ collect the knot sequences in each dimension. In this setting, the multivariate B-spline space $\mathbb{B}_{\mathbf{p},\Xi}$ is defined as the span of tensor product B-splines over knot sequences $\boldsymbol{\xi}^i$, $i = 1, \ldots, n$. The B-splines $B_{\mathbf{j},\mathbf{p},\Xi}$ that span $\mathbb{B}_{\mathbf{p},\Xi}$ are given by:

$$B_{\mathbf{j},\mathbf{p},\Xi}(x^1, \ldots, x^n) := B_{j^1,p^1,\boldsymbol{\xi}^1}(x^1) \times \cdots \times B_{j^n,p^n,\boldsymbol{\xi}^n}(x^n). \quad (5)$$

Just like in the univariate case, we will assume that the degree $\mathbf{p}$ is fixed and drop it from all notation and simply denote the B-splines as $B_{\mathbf{j},\Xi}$, the set of all B-splines as $\mathcal{B}_{\Xi}$, and the spline space as $\mathbb{B}_{\Xi}$.

Here, we also introduce the following notation for the mesh corresponding to Ξ. All $\mathbf{k}$ such that $p^i + 1 \leq k^i \leq m^i - p^i - 1$, define a mesh element $\Omega^e = \bigtimes_{i=1}^{n} (\xi^i_{k^i}, \xi^i_{k^i+1})$. We will index all of these mesh elements Ω^e in a sequential order and collect the mesh element indices e in the set $\mathcal{M}_\Xi$. Since the relation between e and $\mathbf{k}$ relies on Ξ, and our main goal is to have spline spaces of different levels of refinement derived from different Ξ, explicitly denoting this relation will quickly clutter notation. Hence we will use index e to refer to an element, where the underlying relation to Ξ and $\mathbf{k}$ is implicit.

2.3 *Bézier Projection for B-Splines*

In this section we introduce the Bézier projection for B-splines introduced in [29], which is a local approach to project functions in $L^2(\Omega)$ onto the space of B-splines, $\mathbb{B}_\Xi$. There are two main steps in this approach, see Fig. 2.

Given the target function $f \in L^2(\Omega)$, the first step is to project f onto the space of C^{-1} splines of degree $\mathbf{p}$. In the second step, this piece-wise defined C^{-1} approximation is smoothed out to construct a spline in $\mathbb{B}_\Xi$ that approximates f.

The initial C^{-1} spline projection can be performed on each element separately. For this, consider element $\Omega^e = \bigtimes_{i=1}^{n} \left(\xi^i_{k^i}, \xi^i_{k^i+1}\right)$ and let the local knot sequence for each dimension be given by

$$\boldsymbol{\xi}^{i,e} := \Big\{ \underbrace{\xi^i_{k^i}, \ldots, \xi^i_{k^i}}_{p^i+1 \text{ times}}, \ \underbrace{\xi^i_{k^i+1}, \ldots, \xi^i_{k^i+1}}_{p^i+1 \text{ times}} \Big\}, \ i = 1, \ldots, n. \tag{6}$$

Let $\Xi^e = (\boldsymbol{\xi}^{1,e}, \ldots, \boldsymbol{\xi}^{n,e})$ collect these local knot sequences such that $\mathcal{B}_{\Xi^e}$ defines the local spline space on element Ω^e.

$\mathcal{B}_{\Xi^e}$ is the Bernstein polynomial basis on Ω^e and they span $\mathbb{P}_{\mathbf{p}}(\Omega^e)$. The initial L^2 projection of $f|_{\Omega^e}$ onto the space $\mathbb{P}_{\mathbf{p}}(\Omega^e)$ gives us

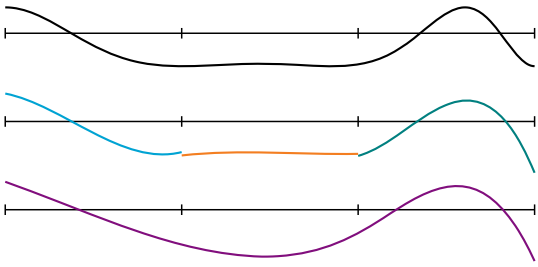


Fig. 2 The two main steps of Univariate Bézier B-spline projection. The Target function is initially projected on to a local polynomial basis of degree p, on every element. These C^{-1} spline functions are smoothed to obtain a global projection in the B-spline space $\mathbb{B}_{\boldsymbol{\xi}}$. Note the similarity to Fig. 1, however, instead of using macro projection elements, the initial step is performed element-wise

$$\Pi^e f := \sum_{j^1=1}^{p^1+1} \cdots \sum_{j^n=1}^{p^n+1} a^e_{\mathbf{j}} B_{\mathbf{j},\Xi^e}, \tag{7}$$

where Π^e f is the solution to the minimization problem

$$\Pi^e f := \underset{f_h \in \mathcal{B}_{\Xi^e}}{\operatorname{argmin}} \| f - f_h \|_{L^2(\Omega^e)}.$$

Let now $\mathcal{I}^e(\mathcal{B}_\Xi)$ denote the set of indices of B-splines whose support contains Ω^e, that is,

$$\mathcal{I}^e(\mathcal{B}_\Xi) := \left\{ \mathbf{j} \ : \ B_{\mathbf{j},\Xi} \in \mathcal{B}_\Xi, \ \ \operatorname{supp}(B_{\mathbf{j},\Xi}) \cap \Omega^e \neq \emptyset \right\}. \tag{8}$$

The B-spline basis of $\mathbb{B}_\Xi$, restricted to element Ω^e, also spans $\mathbb{P}_{\mathbf{p}}(\Omega^e)$. Hence, a mapping exists such that $B_{\mathbf{j},\Xi} = \sum_{\mathbf{i}} \mathrm{C}^e_{\mathbf{j},\mathbf{i}} B_{\mathbf{i},\Xi^e}$ for each $\mathbf{j} \in \mathcal{I}^e(\mathcal{B}_\Xi)$. This matrix C^e is called the Bézier extraction matrix [2, 25] and describes each B-spline as a linear combination of Bernstein polynomials on element Ω^e. Exploiting this relation, we can express approximation (7) as follows, where $\mathbf{b}^e := (\mathrm{C}^e)^{-T} \mathbf{a}^e$,

$$\Pi^e f = \sum_{\mathbf{j} \in \mathcal{I}^e(\mathcal{B}_\Xi)} b^e_{\mathbf{j}} B_{\mathbf{j},\Xi}\big|_{\Omega^e}. \tag{9}$$

Performing the above local projections for each element, we get piecewise-polynomial descriptions of the form (9) on each mesh element. However, if $\mathbf{j} \in \mathcal{I}^e(\mathcal{B}_\Xi) \cap \mathcal{I}^{e'}(\mathcal{B}_\Xi)$, then in general $b^e_{\mathbf{j}} \neq b^{e'}_{\mathbf{j}}$. See Fig. 2. Consequently, the piecewise descriptions together do not define a spline in $\mathbb{B}_\Xi$. This is rectified by a smoothing operation that, for a given B-spline $B_{\mathbf{j},\Xi}$, takes all associated coefficients $b^e_{\mathbf{j}}$ and performs a weighted averaging to yield a single coefficient $b_{\mathbf{j}}$,

$$b_{\mathbf{j}} := \sum_{e \in \mathcal{E}_{\mathbf{j}}(\mathcal{B}_\Xi)} \omega^e_{\mathbf{j}} b^e_{\mathbf{j}}, \tag{10}$$

where $\omega^e_{\mathbf{j}}$ are the averaging weights and $\mathcal{E}_{\mathbf{j}}(\mathcal{B}_\Xi)$ is the set of all elements where $B_{\mathbf{j},\Xi}$ is supported,that is,

$$\mathcal{E}_{\mathbf{j}}(\mathcal{B}_\Xi) := \left\{ e \in \mathcal{M}_\Xi \ : \ B_{\mathbf{j},\Xi} \in \mathcal{B}_\Xi, \ \Omega^e \cap \operatorname{supp}(B_{\mathbf{j},\Xi}) \neq \emptyset \right\}. \tag{11}$$

They can be defined in different ways, we follow the recommendations of [29] and choose them as

$$\omega^e_{\mathbf{j}} := \frac{\int_{\Omega^e} B_{\mathbf{j},\Xi} dx}{\int_{\Omega} B_{\mathbf{j},\Xi} dx}. \tag{12}$$

Finally, the global Bézier projection operator $\Pi : L^2(\Omega) \to \mathbb{B}_\Xi$ is defined by

$$\Pi f := \sum_{\mathbf{j}: B_{\mathbf{j},\Xi} \in \mathcal{B}_\Xi} b_{\mathbf{j}} B_{\mathbf{j},\Xi}. \tag{13}$$

Observe that $\sum_{e \in \mathcal{E}_{\mathbf{j}}(\mathcal{B}_\Xi)} \omega_{\mathbf{j}}^e = 1$. In Lemma 3.2 of [29], it is shown that this is required for (13) to be a projector.

3 Pseudo-Local Linear Independence of THB-Splines

While B-splines exhibit many nice properties, they lack the ability to be locally refined. With this in mind, Hierarchical B-splines (HB-splines) and, later, Truncated Hierarchical B-splines (THB-splines) have been introduced. These are defined in terms of a sequence of nested multivariate B-spline spaces and a sequence of nested regions. However, in contrast to the B-spline spaces, certain mesh elements will be overloaded for both HB-splines and THB-splines (see Definition 1.1). In this section, under suitable assumptions on the mesh, we identify the non-overloaded elements of the mesh and use them to partition the mesh into non-overloaded macro-elements.

3.1 Construction of THB-Splines

Consider a nested sequence of multivariate B-spline spaces over domain $\Omega = (0, 1)^n$:

$$\mathbb{B}_1 \subset \mathbb{B}_2 \subset \cdots \subset \mathbb{B}_L. \tag{14}$$

Here the multivariate B-spline space $\mathbb{B}_\ell := \mathbb{B}_{\Xi_\ell}$ containing the B-splines $B_{\mathbf{j},\ell} := B_{\mathbf{j}} \in \mathbb{B}_{\Xi_\ell}$, is a level-$\ell$ tensor-product B-spline space defined over the knot sequences $\left\{\boldsymbol{\xi}_\ell^i\right\}_{i=1}^n =: \Xi_\ell$. As before, these knot sequences $\boldsymbol{\xi}_\ell^i$ are all $(p+1)$-open knot sequences where the interior knots have multiplicity 1. In this setting, nestedness of the B-spline spaces is ensured if and only if:

$$\boldsymbol{\xi}_1^i \subset \boldsymbol{\xi}_2^i \subset \cdots \subset \boldsymbol{\xi}_L^i, \qquad i = 1, \ldots, n. \tag{15}$$

We will denote the sets containing level-ℓ mesh element indices as $\mathcal{M}_\ell := \mathcal{M}_{\Xi_\ell}$. Next, consider a sequence of nested, closed subsets of $\overline{\Omega}$:

$$\Omega_L \subseteq \cdots \subseteq \Omega_1 := \overline{\Omega}, \tag{16}$$

where Ω_ℓ is the closure of the union of mesh elements Ω^e for some $e \in \mathcal{M}_{\ell-1}$. The collection of those subsets is denoted by:

$$\Psi := \{\Omega_1, \ldots, \Omega_L\}, \tag{17}$$

and will be referred to as the **domain hierarchy** on Ω. For a given level ℓ, we can use this hierarchy to split the B-spline basis functions $\mathcal{B}_\ell$:

$$\mathcal{B}_\ell^{\text{in}} := \left\{ B_{\mathbf{i},\ell} \in \mathcal{B}_\ell \ : \ \text{supp}(B_{\mathbf{i},\ell}) \subseteq \Omega_\ell \right\}, \tag{18}$$

$$\mathcal{B}_\ell^{\text{ex}} := \left\{ B_{\mathbf{i},\ell} \in \mathcal{B}_\ell \ : \ \text{supp}(B_{\mathbf{i},\ell}) \nsubseteq \Omega_\ell \right\}. \tag{19}$$

Using the above, the following defines the Hierarchical B-spline basis functions, or HB-spline basis functions.

Definition 3.1 Given a domain hierarchy Ψ, the corresponding set of **HB-spline** basis functions is denoted by $\mathcal{H}_\Psi$ and defined recursively as follows:

1. $\mathcal{H}_1 := \mathcal{B}_1$,
2. for $\ell = 2, \ldots, L$:

$$\mathcal{H}_\ell := \mathcal{H}_\ell^C \cup \mathcal{H}_\ell^F,$$

where

$$\mathcal{H}_\ell^C := \left\{ B_{\mathbf{j},k} \in \mathcal{H}_{\ell-1} : \text{supp}(B_{\mathbf{j},k}) \nsubseteq \Omega_\ell \right\},$$
$$\mathcal{H}_\ell^F := \mathcal{B}_\ell^{\text{in}},$$

3. $\mathcal{H}_\Psi := \mathcal{H}_L$.

See the top plot of Fig. 3 for an example. HB-spline basis functions lack the partition of unity property, which is shown in red in Fig. 3. In addition, the number of overlapping basis functions associated with different hierarchical levels increases easily. This motivates the construction of a different basis, for example, the THB-spline basis, based on the following truncation mechanism.

Definition 3.2 Given $\ell = 2, \ldots, L$. Let $f \in \mathbb{B}_{\ell-1}$ be represented in the B-spline basis $\mathcal{B}_\ell$:

$$f = \sum_{\mathbf{j}: B_{\mathbf{j},\ell} \in \mathcal{B}_\ell} c_{\mathbf{j},\ell} B_{\mathbf{j},\ell}. \tag{20}$$

The **truncation** of f at hierarchical level ℓ is defined as the the sum of the terms appearing in (20) corresponding to the B-splines in $\mathcal{B}_\ell^{\text{ex}}$:

$$\text{trunc}_\ell(f) := \sum_{\mathbf{j}: B_{\mathbf{j},\ell} \in \mathcal{B}_\ell^{\text{ex}}} c_{\mathbf{j},\ell} B_{\mathbf{j},\ell}. \tag{21}$$

The THB-spline basis functions are constructed by successively truncating the functions from Definition 3.1.

Definition 3.3 Given a domain hierarchy Ψ, the corresponding set of **THB-splines** basis is denoted by $\mathcal{T}_\Psi$ and defined recursively as follows:

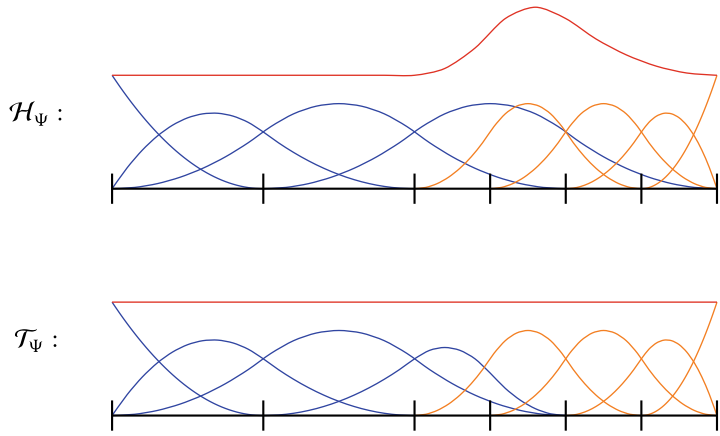


Fig. 3 An example of an HB-spline basis (top) and a corresponding THB-spline basis (bottom). The blue splines are from the first level and the orange splines are from the second one on a two level hierarchy. Notice that the total sum of all splines (red line) is 1 only for the THB-spline basis

1. $\mathcal{T}_1 := \mathcal{B}_1$,
2. for $\ell = 2, \ldots, L$:

$$\mathcal{T}_\ell := \mathcal{T}_l^C \cup \mathcal{T}_\ell^F,$$

where

$$\mathcal{T}_\ell^C := \left\{ \operatorname{trunc}_\ell(T) \ : \ T \in \mathcal{T}_{\ell-1}, \operatorname{supp}(T) \not\subseteq \Omega_\ell \right\},$$
$$\mathcal{T}_\ell^F := \mathcal{B}_\ell^{\text{in}},$$

3. $\mathcal{T}_\Psi := \mathcal{T}_L$.

The $N := |\mathcal{T}_\Psi|$ THB-spline basis functions are linearly indexed by some ordering $i = 1, \ldots, N$:

$$\mathcal{T}_\Psi = \{T_i\}_{i=1}^N. \tag{22}$$

The space of THB-splines will be denoted as $\mathbb{T}_\Psi$. Finally, we define the set of active level-ℓ mesh elements $\mathcal{M}_\ell^{\text{in}}$ as

$$\mathcal{M}_\ell^{\text{in}} := \left\{ e \in \mathcal{M}_\ell \ : \ \Omega^e \subset \Omega_\ell \text{ and } \Omega^e \cap \Omega_{\ell+1} = \emptyset \right\}. \tag{23}$$

In Fig. 3 a comparison between THB-splines and HB-splines is given.

3.2 Some Assumption on the Hierarchical Meshes

For our purpose, we assume that from this point forward all THB-spline spaces conform to the the following assumptions.

Assumption 3.1 The mesh element sizes at level $\ell = 1$ satisfies **quasi-uniformity**. That is, there exists a constant $\eta \geq 1$ such that for any two adjacent non-empty knot intervals, the mesh-size ratio is bounded as

$$\eta^{-1} \leq \frac{\xi^i_{j+1,1} - \xi^i_{j,1}}{\xi^i_{j,1} - \xi^i_{j-1,1}} \leq \eta, \quad i = 1, \ldots, d. \tag{24}$$

Moreover, for $\ell > 1$, the level-ℓ knot sequence in each direction is obtained by bisecting each non-zero knot interval of the corresponding level-$(\ell - 1)$ knot sequence.

Assumption 3.2 Given a mesh element Ω^e, $e \in \mathcal{M}^{\text{in}}_\ell$, the only THB-splines that can be supported on Ω^e correspond to (truncated) B-splines from at most two levels: $\ell - 1$ and ℓ.

Assumption 3.2 is a particular class of mesh admissibility introduced in [6], specifically, the mesh is of admissibility class 2.

3.3 An Example of Overloaded Mesh Elements in $\mathbb{R}^1$

In contrast to B-splines, mesh elements can be overloaded even for univariate THB-splines. In Fig. 4 an example of an overloaded element is depicted in blue for a quadratic THB-spline space.

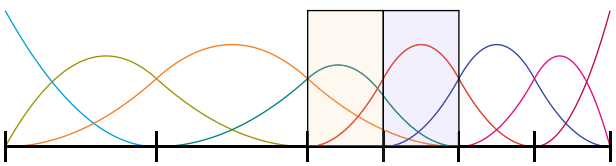


Fig. 4 The basis functions of a quadratic THB-spline space consisting of two levels are shown. While the blue coloured element is overloaded, the combination of the blue and orange is not

This univariate example motivates the following observations, which drive the developments in the next sections in the general multivariate setting.

- As a result of the truncation mechanism and Assumption 3.2, at most p elements that are adjacent to the border of Ω_ℓ can be overloaded.
- The undesirable overloading on the blue element can be resolved by creating a macro-element that combines the blue and the orange elements.

Then, in the next section, we will characterize the non-overloaded elements for THB-splines and use them to create macro-elements, called projection elements, that are not overloaded.

3.4 Characterizing Non-overloaded Mesh Elements in $\mathbb{R}^n$

As shown in Sect. 3.3, THB-spline spaces suffer from overloading even under Assumption 3.2. While characterizing the overloaded elements in 1D is simple, the situation in $\mathbb{R}^n$, $n > 1$, is more involved. We perform this characterization in this section under suitable assumptions. These assumption help us prove a general statement for $\mathbb{R}^n$, $n \geq 1$.

Consider $e \in \mathcal{M}_\ell^{\text{in}}$, $\ell > 1$. The element Ω^e can only be overloaded if:

$$\Omega^e \subset \operatorname{supp}(B_{\mathbf{j},\ell}) \text{ for some } B_{\mathbf{j},\ell} \in \mathcal{B}_\ell^{\text{ex}}. \tag{25}$$

Each such $B_{\mathbf{j},\ell}$ will contribute to some THB-spline that is obtained by truncating a level-$(\ell-1)$ B-spline and is supported on Ω^e. Our characterization of non-overloadedness assumes that such THB-splines do not vanish at $\partial\Omega^e \cap (\partial\Omega_\ell \backslash \partial\Omega)$. Unfortunately, this requirement may not be met, for instance, if the refinement domain is too "thin".

For example, consider the situation shown in Fig. 5 for the cross-hatched level-ℓ element Ω^e. On this Ω^e, B is some level-ℓ B-spline that clearly vanishes at $\partial\Omega^e \cap (\partial\Omega_\ell \backslash \partial\Omega)$ and, therefore, there may be a level-$(\ell-1)$ THB-spline which also vanishes at this boundary.

To prevent such cases, we impose two additional assumptions on the mesh. This leads to the desired characterization of non-overloaded elements in $\mathbb{R}^n$ and forms the basis for the construction of non-overloaded macro elements. We begin this discussion by decomposing the boundary of Ω^e as follows.

Definition 3.4 Given $\Omega^e = \bigtimes_{i=1}^n \left(\xi^i_{k^i,\ell}\,,\ \xi^i_{k^i+1,\ell}\right)$, $e \in \mathcal{M}_\ell$, the following boundary facets,

$$\begin{aligned}
&\bigtimes_{i=1}^{r-1}\left[\xi^i_{k^i,\ell}\,,\ \xi^i_{k^i+1,\ell}\right] \times \left\{\xi^r_{k^r,\ell}\right\} \bigtimes_{i=r+1}^{n}\left[\xi^i_{k^i,\ell}\,,\ \xi^i_{k^i+1,\ell}\right],\\
&\bigtimes_{i=1}^{r-1}\left[\xi^i_{k^i,\ell}\,,\ \xi^i_{k^i+1,\ell}\right] \times \left\{\xi^r_{k^r+1,\ell}\right\} \bigtimes_{i=r+1}^{n}\left[\xi^i_{k^i,\ell}\,,\ \xi^i_{k^i+1,\ell}\right],
\end{aligned} \tag{26}$$

are called r**-normal facets** of Ω^e.

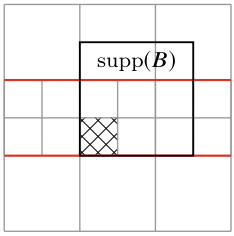


Fig. 5 A possible scenario for $\mathbf{p} = [2, 2]^T$ where a level-ℓ B-spline $B \in \mathcal{B}_\ell^{\text{ex}}$ that is supported on a level-ℓ element Ω^e (cross-hatched) may vanish at $\partial\Omega^e \cap (\partial\Omega_\ell \backslash \partial\Omega)$. Such cases are ruled out by the additional assumptions introduced in Sect. 3.4

See Fig. 6a, b for an example of facets in 2 dimensions. In addition, we will need to refer to elements obtained by translations of Ω^e in index space. To quantify this, we introduce these element translations.

Definition 3.5 Given $\Omega^e = \bigtimes_{i=1}^n \left(\xi^i_{k^i,\ell}\ ,\ \xi^i_{k^i+1,\ell}\right), e \in \mathcal{M}_\ell$, the translation of Ω^e by $\mathbf{t} = (t^1, \ldots, t^n)$ is defined as the element $\tau_{\mathbf{t}}(\Omega^e) := \bigtimes_{i=1}^n \left(\xi^i_{k^i+t^i,\ell}\ ,\ \xi^i_{k^i+t^i+1,\ell}\right)$.

For every overloaded element, (25) holds. However, in general there will exist elements for which (25) holds but which are in fact not overloaded. We will call these well-behaved border elements and they are defined as follows.

Definition 3.6 Given $e \in \mathcal{M}_\ell^{\text{in}}$ and $1 \le i \le n$, let $d_\ell^{i,e} \in \mathbb{Z}_{\ge 0}$ be the smallest number such that one of the following translations $\mathbf{t}$ of Ω^e,

$$\Big(0,\ \ldots,\ 0,\ \underbrace{d_\ell^{i,e}}_{i\text{-th position}} 0,\ \ldots,\ 0\Big) \quad \text{or} \quad \Big(0,\ \ldots,\ 0,\ \underbrace{-d_\ell^{i,e}}_{i\text{-th position}} 0,\ \ldots,\ 0\Big). \tag{27}$$

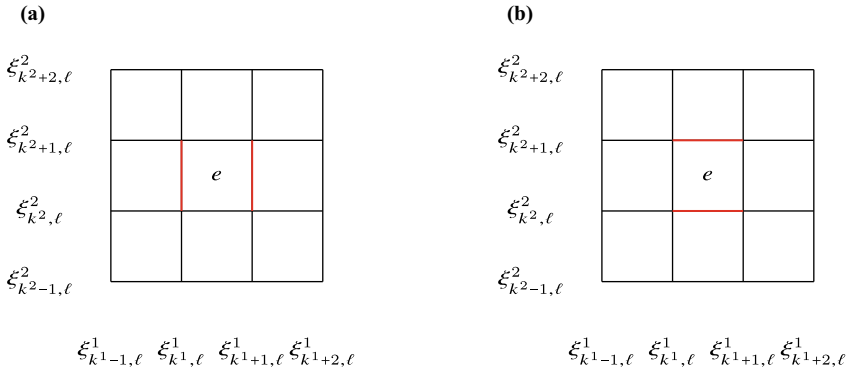


Fig. 6 Given element $\Omega^e = \bigtimes_{i=1}^n \left(\xi^i_{k^i,\ell}\ ,\ \xi^i_{k^i+1,\ell}\right)$, an example of the 1-normal facet is shown in **a** and an example of the 2-normal facet in **b**

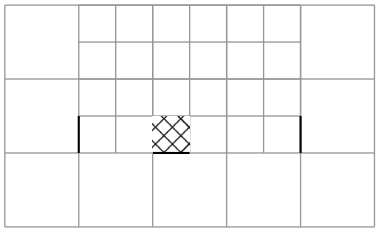


Fig. 7 An example of Definition 3.6. For the element Ω^e denoted with the cross-hatch, $d_\ell^{1,e} = 2$ and $d_\ell^{2,e} = 0$. Thus, it is a well-behaved border element if $p^1 \leq 2$

gives an element $\tau_{\mathbf{t}}(\Omega^e)$ where one of its i-normal facets is contained in $\partial\Omega_\ell \backslash \partial\Omega$. If such a $d_\ell^{i,e}$ does not exists, we define it be ∞. Then Ω^e is called a **well-behaved border element** if

- $\partial\Omega^e \cap (\partial\Omega_\ell \backslash \partial\Omega) \neq \emptyset$, and,
- for all i, $d_\ell^{i,e} = 0$ or $d_\ell^{i,e} \geq p^i$.

See Fig. 7 for an example of a well-behaved border element. The well-behaved border elements touch the border $\partial\Omega_\ell$ in a specific manner. The next assumptions that we will shortly impose will ensure that cases such as the one shown in Fig. 5 do not exist. This will in turn imply that well-behaved border elements are not overloaded. As the r-normal facets are closed sets, the intersection of any two of them is either empty or is a subset of $\partial\Omega^e$ (e.g., a vertex, an edge, a face, and so on). This allows us to describe the set $\partial\Omega^e \cap (\partial\Omega_\ell \backslash \partial\Omega)$ as a union of certain intersections of r-normal facets as follows. Given a set A containing normal facets of Ω^e, let $\mathrm{intrsct}(A)$ denote the intersection of all facets in A. Then, for a well-behaved border element Ω^e, let F^e be the set containing the minimal number of r-normal facets of Ω^e such that the union of all possible intersections coincides with $\partial\Omega^e \cap \partial\Omega_\ell \backslash \partial\Omega$.

$$F^e = \arg\min_{\mathrm{F}} \left\{ |F| \ : \bigcup_{\substack{A \subset F \\ \mathrm{intrsct}(A) \subset \partial\Omega_\ell \backslash \partial\Omega}} \mathrm{intrsct}(A) = \partial\Omega^e \cap (\partial\Omega_\ell \backslash \partial\Omega) \right\}. \tag{28}$$

Then, we associate a direction to Ω^e, denoted $\mathbf{n}_\ell^e$, and define it as the sum of unit normals for each facet in F^e (by convention, we assume that each unit normal is pointing into Ω^e). See Fig. 8 for an example in 3D. Note that, since Ω_ℓ is the closure of the union of level-$(\ell - 1)$ mesh elements, $\mathbf{n}_\ell^e$ is always a non-zero vector as a result of Assumption 3.1. Indeed, this implies that, for a fixed r, at most one r-normal facet of Ω^e can be in F^e.

As motivated at the beginning of this section, we desire that all B-splines $B_{\mathbf{j},\ell} \in \mathcal{B}_\ell^{\mathrm{ex}}$ with support on a well-behaved border element Ω^e do not vanish on the boundary $\partial\Omega^e$. For this we will assume that Ω_ℓ contains enough elements so that it can never be too "thin."

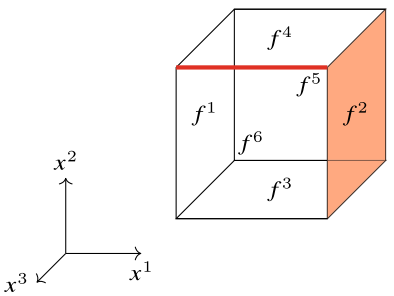


Fig. 8 An example of the facet set F^e for an element Ω^e where f^i are the facets such that f^1, f^2 are the left and right sides respectively, f^3, f^4 the bottom and top sides and f^5, f^6 the sides behind and in front. Let $\partial\Omega^e \cap \partial\Omega_\ell \backslash \partial\Omega$ be the region shown in red (i.e., it consists of a face and an edge). Then, the set $F^e = \{f^2, f^4, f^6\}$ is the smallest set that can represent $\partial\Omega^e \cap \partial\Omega_\ell \backslash \partial\Omega = f^2 \cup \left(f^4 \cap f^6\right)$. For this element Ω^e, the direction is then given by $\mathbf{n}_\ell^e = (-1, -1, -1)^T$

Assumption 3.3 For any well-behaved border element $\Omega^e = \bigtimes_{i=1}^n \left(\xi^i_{k^i,\ell}\,,\ \xi^i_{k^i+1,\ell}\right)$ and the direction $\mathbf{n}_\ell^e$ associated to it, the following inclusion must hold,

$$\bigcup_{\mathbf{t}\in\bigtimes_{i=1}^n G^i} \tau_{\mathbf{t}}\left(\Omega^e\right) \subseteq \Omega_\ell, \tag{29}$$

where

$$G^i := \begin{cases} \{0\}, & n_\ell^{i,e} = 0, \\ \left\{0, 1, \ldots, 2p^i - 1\right\}, & n_\ell^{i,e} = 1, \\ \left\{0, -1, \ldots, -(2p^i - 1)\right\}, & n_\ell^{i,e} = -1. \end{cases}$$

Note that Assumption 3.3 is stronger than strictly necessary for obtaining that all coarse, truncated B-splines on Ω^e do not vanish on its border. However, we will also need the latter fact to hold for certain macro elements that contain Ω^e, and this is why the stronger statement is assumed.

Definition 3.7 Given a well-behaved border element $\Omega^e = \bigtimes_{i=1}^n \left(\xi^i_{k^i,\ell}\,,\ \xi^i_{k^i+1,\ell}\right)$, and the direction $\mathbf{n}_\ell^e$ associated to it, we define the **projection element** generated by Ω^e as

$$\hat{\Omega}^e = \bigcup_{\mathbf{t}\in\bigtimes_{i=1}^n G^i} \tau_{\mathbf{t}}(\Omega^e), \tag{30}$$

where

$$G^i := \begin{cases} \{0\}, & n_\ell^{i,e} = 0, \\ \left\{0, 1, \ldots, p^i - 1\right\}, & n_\ell^{i,e} = 1, \\ \left\{0, -1, \ldots, -(p^i - 1)\right\}, & n_\ell^{i,e} = -1. \end{cases}$$

Note that as a direct consequence of Assumption 3.3, a level-ℓ well-behaved border element Ω^e will generate a projection element which is fully contained in $\Omega_\ell \supseteq \hat{\Omega}^e$. In addition, we desire that any possible overloaded mesh element, see (25), is contained in a unique projection element.

Assumption 3.4 Let Ω^e, $e \in \mathcal{M}_\ell^{\text{in}}$, be so that its $\mathbf{t} = (t^1, \ldots, t^n)$ translation, where $|t^i| < p^i$ for all i, satisfies $\partial\tau_{\mathbf{t}}(\Omega^e) \cap (\partial\Omega_\ell \backslash \partial\Omega) \neq \emptyset$. Then, there **exists** a **unique** projection element that contains Ω^e.

Note that overlapping projection elements do occur in general, and that's why the above assumption is needed. See Fig. 9 for an example in 2D. Here there are two projection elements (generated by the cross-hatched well-behaved border elements) that overlap. In 2D, this is due to "stair-like" refinements, i.e., when the boundary of Ω_ℓ has stair-like steps which have smaller widths / heights than projection elements.

We start by showing a technical lemma that will be useful later for proving the non-overloadedness of well-behaved border elements.

Lemma 3.1 *Let Ω^e be a well-behaved border element, and let $\tau_{\mathbf{t}_1}(\Omega^e), \tau_{\mathbf{t}}(\Omega^e) \subset \hat{\Omega}^e$, $|t_1^i| \geq |t^i|$ for all i, such that they are both contained in the same level-$(\ell-1)$ element. Let $B_{r,\ell-1}$ be a level-$(\ell-1)$ B-spline containing this level-$(\ell-1)$ element, and let $T \in \mathcal{T}_\Psi$ be its truncation. Then if T does not vanish on $\tau_{\mathbf{t}_1}(\Omega^e)$ then it does not vanish on $\tau_{\mathbf{t}}(\Omega^e)$ either.*

Proof Assume without loss of generality that all components of $\mathbf{n}_\ell^e$ are non-negative, implying that $t_1^i \geq t^i \geq 0$ for all i.

As both $\tau_{\mathbf{t}_1}(\Omega^e)$ and $\tau_{\mathbf{t}}(\Omega^e)$ are contained in the same level-$(\ell-1)$ element, $B_{\mathbf{r},\ell-1}$ has support on both elements. Then, for some $s^1, \ldots, s^n$, we can express $B_{\mathbf{r},\ell-1}$ as

$$B_{\mathbf{r},\ell-1} = \sum_{q^1=s^1}^{s^1+p^1+1} \cdots \sum_{q^n=s^n}^{s^n+p^n+1} c_{\mathbf{q}} B_{\mathbf{q},\ell}. \tag{31}$$

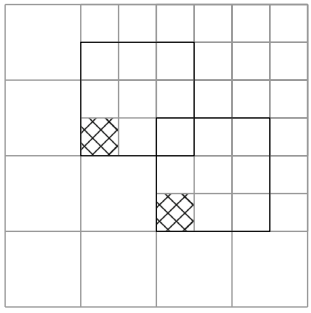


Fig. 9 Without imposing Assumption 3.4, it is possible to generate refinement domains, for which projection elements can overlap. For $p^1 = p^2 = 3$, the two well-behaved border elements (cross-hatched) shown above generate projection elements that overlap

Since T has support on $\tau_{\mathbf{t}_1}(\Omega^e)$, there is a non-zero $c_{\mathbf{q}_1}$ for $B_{\mathbf{q}_1,\ell} \in \mathcal{B}_\ell^{\text{ex}}$ with support on $\tau_{\mathbf{t}_1}(\Omega^e)$. If $B_{\mathbf{q}_1,\ell}$ has support on $\tau_{\mathbf{t}}(\Omega^e)$, we are done. Otherwise,

$$q_1^i \geq s^i + t_1^i \geq s^i + t_1^i - t^i, \qquad i = 1, \ldots, n. \tag{32}$$

As $B_{\mathbf{q}_1,\ell} \in \mathcal{B}_\ell^{\text{ex}}$ has support on some element of index $e' \in \mathcal{M}_{\ell-1}^{\text{in}}$ and since $0 \leq t_1^i - t^i \leq 1$, we have that $B_{\mathbf{q}_1-\mathbf{t}_1+\mathbf{t},\ell}$ has support on the same level-$(\ell-1)$ element $\Omega^{e'}$ and thus $B_{\mathbf{q}_1-\mathbf{t}_1+\mathbf{t},\ell} \in \mathcal{B}_\ell^{\text{ex}}$. But then, $B_{\mathbf{q}_1-\mathbf{t}_1+\mathbf{t},\ell}$ and by extension T have support on $\tau_{\mathbf{t}}(\Omega^e)$. □

We are now ready to show that well-behaved border elements and projection elements are non-overloaded. Both are a direct consequence of the following proposition.

Proposition 3.1 *Consider a well-behaved border element Ω^e and consider any $\mathbf{t}_1 = (t_1^1, \ldots, t_1^n)$ such that $\tau_{\mathbf{t}_1}(\Omega^e) \subset \hat{\Omega}^e$. Then, the following set of THB-splines on $\tau_{\mathbf{t}_1}(\Omega^e)$ is linearly independent,*

$$\Big\{ T_j \ : \ \text{supp}(T_j) \cap \tau_{\mathbf{t}_1}(\Omega^e) \neq \emptyset \text{, and,} \\ \text{supp}(T_j) \cap \tau_{\mathbf{t}}(\Omega^e) = \emptyset \text{ where } \tau_{\mathbf{t}}(\Omega^e) \subset \hat{\Omega}^e, \mathbf{t}_1 \neq \mathbf{t} \text{ and } |t_1^i| \geq |t^i| \text{ for all } i \Big\}. \tag{33}$$

Proof Assume without loss of generality that $\mathbf{n}_\ell^e$ is such that all of its components are non-negative, and let $\Omega^e = \bigtimes_{i=1}^n \left(\xi_{k^i,\ell}^i \ , \ \xi_{k^i+1,\ell}^i \right)$. In the following, we use $\mathbf{t}$ to denote a translate vector such that $\tau_{\mathbf{t}}(\Omega^e) \subset \hat{\Omega}^e$, $\mathbf{t}_1 \neq \mathbf{t}$, $t_1^i \geq t^i$ for all i.

The claim is immediate for the special case when there exists a $\mathbf{t}$ such that $\tau_{\mathbf{t}_1}(\Omega^e)$ and $\tau_{\mathbf{t}}(\Omega^e)$ are contained in a common element $e' \in \mathcal{M}_{\ell-1}$. In this case, from Lemma 3.1, the THB-splines that contain $\tau_{\mathbf{t}_1}(\Omega^e)$ in their support but not $\tau_{\mathbf{t}}(\Omega^e)$ only correspond to level-ℓ B-splines. The claim then follows from the linear independence of level-ℓ B-splines.

The only other case to consider is when $\tau_{\mathbf{t}_1}(\Omega^e)$ is not contained in a common element of level $(\ell-1)$, with any $\tau_{\mathbf{t}}(\Omega^e)$ satisfying the above conditions. In particular, this means that all t_1^i are even numbers due to Assumption 3.1. Here, we will show that the desired set of THB-splines (33) spans a subspace of $\mathbb{P}_{\mathbf{p}}\left(\tau_{\mathbf{t}_1}(\Omega^e)\right)$, where each polynomial in the subspace vanishes p^i times at the hyperplane $\xi^i = \xi_{k^i+t_1^i,\ell}^i$ if $t_1^i > 0$. This will be shown, by first constructing a basis which spans the subset. Then, each of these basis functions will be related to a particular THB-spline, showing that the THB-splines span the same space.

Let the element $e' \in \mathcal{M}_{\ell-1}$ contain $\Omega^e \subset \Omega^{e'}$ and let the level-$(\ell-1)$ B-splines supported on $\Omega^{e'}$, be $B_{\mathbf{q},\ell-1}$, where

$$\mathbf{q} \in \mathcal{I}^{e'}(\mathcal{B}_{\ell-1}) = \bigtimes_{i=1}^n \left\{ r^i, \ldots, r^i + p^i \right\}. \tag{34}$$

Then, the B-splines supported on $\tau_{\mathbf{t}_1}(\Omega^e)$ and not on any $\tau_{\mathbf{t}}(\Omega^e)$ as above correspond to $\mathbf{q} \in \bigtimes_{i=1}^n S_{\ell-1}^i$:

$$S_{\ell-1}^i := \begin{cases} \left\{r^i + p^i + \frac{t_1^i}{2}\right\}, & t_1^i > 0, \\ \left\{r^i, \ldots, r^i + p^i\right\}, & \text{else.} \end{cases} \tag{35}$$

Similarly, let the level-ℓ B-splines supported on Ω^e be $B_{\mathbf{q},\ell}$, where

$$\mathbf{q} \in \mathcal{I}^e(\mathcal{B}_\ell) = \bigtimes_{i=1}^n \left\{s^i, \ldots, s^i + p^i\right\}. \tag{36}$$

Then, the B-spline supported on $\tau_{\mathbf{t}_1}(\Omega^e)$ and not any $\tau_{\mathbf{t}}(\Omega^e)$ correspond to $\mathbf{q} \in \bigtimes_{i=1}^n S_\ell^i$ where

$$S_\ell^i := \begin{cases} \left\{s^i + p^i + t_1^i\right\}, & t_1^i > 0, \\ \left\{s^i, \ldots, s^i + p^i\right\}, & \text{else.} \end{cases} \tag{37}$$

Moreover, the active level-ℓ B-splines from the above set correspond to $B_{\mathbf{q},\ell}$, $\mathbf{q} \in \bigcup_{l=1}^n \bigtimes_{i=1}^n A_\ell^{i,l}$, where

$$A_\ell^{i,l} := \begin{cases} \left\{s^i + p^i + t_1^i\right\}, & t_1^i > 0, \\ \left\{s^i + p^i\right\}, & t_1^i = 0 \text{ and } d_\ell^{i,e} = 0 \text{ or } i = l, \\ \left\{s^i, \ \ldots, \ s^i + p^i\right\}, & t_1^i = 0 \text{ and } d_\ell^{i,e} > p^i \text{ and } i \neq l. \end{cases} \tag{38}$$

As a specific example, in Fig. 10a, b the supports of the level-ℓ B-splines from $\bigtimes_{i=1}^n A_\ell^{i,1}$ and $\bigtimes_{i=1}^n A_\ell^{i,2}$ can respectively be seen for a well-behaved border element. Combining these B-spline index sets, one obtains the full index set of level-ℓ B-splines with support on the well-behaved border element, but no support outside Ω_ℓ.

The level-$(\ell-1)$ B-splines from $\bigtimes_{i=1}^n S_{\ell-1}^i$ that are truncated to 0 on $\tau_{\mathbf{t}_1}(\Omega^e)$ by the above active level-ℓ B-splines are $B_{\mathbf{q},\ell-1}$, $\mathbf{q} \in \bigcup_{l=1}^n \bigtimes_{i=1}^n I_{\ell-1}^{i,l}$, where

$$I_{\ell-1}^{i,l} := \begin{cases} \left\{r^i + p^i + \frac{t_1^i}{2}\right\}, & t_1^i > 0, \\ \left\{r^i + p^i\right\}, & t_1^i = 0 \text{ and } d_\ell^{i,e} = 0 \text{ or } i = l, \\ \left\{r^i, \ \ldots, \ r^i + p^i\right\}, & t_1^i = 0 \text{ and } d_\ell^{i,e} > p^i \text{ and } i = l, \end{cases} \tag{39}$$

From the above, we notice that the cardinalities of $\bigtimes_{i=1}^n S_{\ell-1}^i$ and $\bigtimes_{i=1}^n S_\ell^i$ are the same, and the number of level-$(\ell-1)$ THB-splines truncated to zero above is exactly equal to the number of active level-ℓ THB-splines. Consequently, the complete set of THB-splines supported on $\tau_{\mathbf{t}_1}(\Omega^e)$ and not $\tau_{\mathbf{t}}(\Omega^e)$ is linearly independent. Each of them is either a truncated level-$(\ell-1)$ B-splines from the following set,

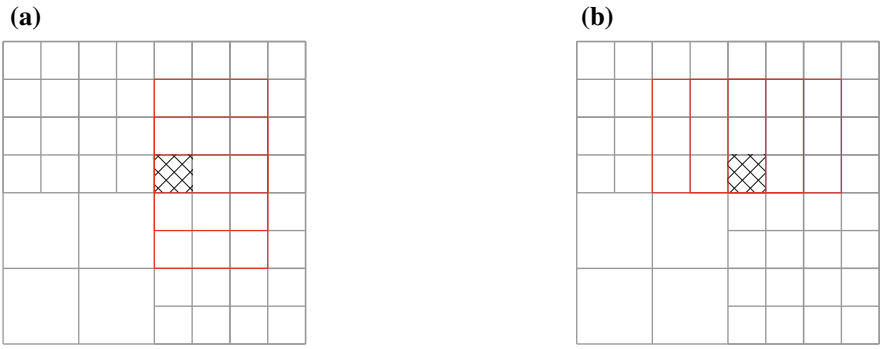


Fig. 10 **a** For a well-behaved border element with $\mathbf{p} = [2, 2]^T$, the supports of the level-ℓ B-splines $B_{\mathbf{q},\ell}$ with $\mathbf{q} \in \bigtimes_{i=1}^{d} A_\ell^{i,1} = \{s^1 + p^1\} \times \{s^2, \ldots, s^2 + p^2\}$ are outlined in red. **b** For a well-behaved border element with $\mathbf{p} = [2, 2]^T$, the supports of the level-ℓ B-splines $B_{\mathbf{q},\ell}$ with $\mathbf{q} \in \bigtimes_{i=1}^{d} A_\ell^{i,2} = \{s^1 + p^1\} \times \{s^2, \ldots, s^2 + p^2\}$ are outlined in red

$$\bigtimes_{i=1}^{n} S_{\ell-1}^{i} \Bigg\backslash \bigcup_{l=1}^{n} \bigtimes_{i=1}^{n} I_{\ell-1}^{i,l}, \tag{40}$$

or a level-ℓ B-spline from the following set,

$$\bigcup_{l=1}^{n} \bigtimes_{i=1}^{n} A_\ell^{i,l}. \tag{41}$$

□

Corollary 3.1 *Well-behaved border elements are not overloaded.*

Proof The result follows by considering $\mathbf{t} = (0, \ldots, 0)$ in Proposition 3.1. □

Theorem 3.1 *Let Ω^e be a well-behaved border element. Then, $\hat{\Omega}^e$ is not overloaded.*

Proof Consider the following representation of the zero function on $\hat{\Omega}^e$ as a linear combination of THB-splines on $\hat{\Omega}^e$,

$$0 = \sum_j c_j T_j|_{\hat{\Omega}^e}. \tag{42}$$

Starting from the linear independence on the well-behaved border element Ω^e shown in Corollary 3.1, we can conclude that some of the c_j in the above sum should be zero. Consequently, repeated applications of Proposition 3.1 can be used to show that all the remaining c_j should be zero as well since every element in $\hat{\Omega}^e$ can be obtained by translations of Ω^e. □

We finish this section by defining well-behaved interior elements and the corresponding projection elements.

Definition 3.8 Given $e \in \mathcal{M}_\ell^{\text{in}}$, Ω^e is called a **well-behaved interior element** if the THB-splines supported on it only correspond to level-ℓ B-splines. The corresponding projection element is defined as $\hat{\Omega}^e := \Omega^e$.

Clearly, a well-behaved interior element is not overloaded. Moreover, as a result of Assumption 3.4, each mesh element is contained in a unique projection element generated by a well-behaved (interior or border) element.

4 Bézier Projection for THB-Splines

Based on the main result of the previous section, we formulate a Bézier projection for THB-splines by using the notion of projection elements. There are various ways to extend the Bézier projector from [29], and we will show theoretical error estimates for these extensions.

4.1 General Formulation of the Projector

From the results and assumptions from Sect. 3, we can make two observations:

- each mesh element is contained in a unique projection element $\hat{\Omega}^e$ (by Definitions 3.7 and 3.8 and Assumptions 3.3 and 3.4), and,
- none of the projection elements are overloaded (by Theorem 3.1).

As a consequence, we can formulate the following generalized Bézier projection methodology for THB-splines.

Step 1: Local projections on $\hat{\Omega}^e$
Given an arbitrary well-behaved (border or interior) element Ω^e, $e \in \mathcal{M}_\ell^{\text{in}}$, denote with $\mathbb{V}(\hat{\Omega}^e)$ some local spline space defined on $\hat{\Omega}^e$; we will make the choice of this spline space concrete in the next subsection. Then, in the first step of the projection, and given a target function $f \in L^2(\Omega)$, we project $f|_{\hat{\Omega}^e}$ onto the space $\mathbb{V}(\hat{\Omega}^e)$. Denote this projection operator as $\hat{\Pi}_0^e$. Denote by $\hat{M}^e := \dim(\mathbb{V}(\hat{\Omega}^e))$ the dimension of $\mathbb{V}(\hat{\Omega}^e)$ and let $\left\{\hat{V}_j\right\}_{j=1}^{\hat{M}^e}$ be a basis of $\mathbb{V}(\hat{\Omega}^e)$. Then we can write

$$\hat{\Pi}_0^e f = \sum_{j=1}^{\hat{M}^e} \hat{f}_j^e \hat{V}_j. \tag{43}$$

In the next step, we apply a subsequent L^2-projection $\hat{\Pi}_1^e : \mathbb{V}(\hat{\Omega}^e) \rightarrow \hat{\mathbb{T}}_\Psi^e$, where $\hat{\mathbb{T}}_\Psi^e$ is defined to be the restriction of the THB-spline space $\mathbb{T}_\Psi$ to the projection element $\hat{\Omega}^e$. Let $\hat{\mathcal{I}}^e(\mathcal{T}_\Psi)$ contain the indices of THB-spline basis functions supported on $\hat{\Omega}^e$. These are linearly independent by Theorem 3.1. The action of $\hat{\Pi}_1^e$ can be encoded in a matrix $\hat{\mathrm{D}}^e$ such that $\hat{\mathbf{c}}^e = \hat{\mathrm{D}}^e \hat{\mathbf{f}}^e$ and:

$$\hat{\Pi}_1^e \hat{\Pi}_0^e f = \sum_{j \in \hat{\mathcal{I}}^e(\mathcal{T}_\Psi)} \hat{c}_j^e T_j|_{\hat{\Omega}^e}. \tag{44}$$

Step 2: Weighted-averaging to form the global projection on Ω
For a THB-spline T_j, the above process gives a coefficient $\hat{c}_j^e$ for each projection element $\hat{\Omega}^e$ on which T_j does not vanish. Then, as in the B-spline case, we perform a weighted averaging of the coefficients $\hat{c}_j^e$ to define a unique coefficient for T_j,

$$c_j := \sum_{e \in \hat{\mathcal{E}}_j(\mathcal{T}_\Psi)} \hat{\omega}_j^e \hat{c}_j^e, \tag{45}$$

where $\hat{\omega}_j^e$ are the averaging weights and $\hat{\mathcal{E}}_j(\mathcal{T}_\Psi)$ is the set of all projection elements where T_j does not vanish,

$$\hat{\mathcal{E}}_j(\mathcal{T}_\Psi) := \left\{ e \ : \ \hat{\Omega}^e \cap \operatorname{supp}\left(T_j\right) \text{ with } T_j \in \mathcal{T}_\Psi \right\}. \tag{46}$$

In particular, similarly to B-splines, we choose the weights to be,

$$\hat{\omega}_j^e := \frac{\int_{\hat{\Omega}^e} T_j \, dx}{\int_\Omega T_j \, dx}. \tag{47}$$

Observe that by Assumption 3.4, the sum over all $e \in \hat{\mathcal{E}}_j(\mathcal{T}_\Psi)$ of $\hat{\omega}_j^e$ is one. This leads to the following definition of the Bézier projector for THB-splines, denote $\Pi : L^2(\Omega) \rightarrow \mathbb{T}_\Psi$,

$$\Pi f := \sum_{j=1}^{N} c_j T_j. \tag{48}$$

4.2 Possible Choices for $\mathbb{V}(\hat{\Omega}^e)$

The above general methodology provides several options for formulating the Bézier projection operator for THB-splines by varying the choice of $\mathbb{V}(\hat{\Omega}^e)$ on each projection element $\hat{\Omega}^e$. We outline three such choices over here. Two common features of the following choices are that:

- they lead to local projections $\hat{\Pi}_1^e \hat{\Pi}_0^e$ that yield optimal a priori error estimates with respect to the mesh size, and,
- the corresponding matrices $\hat{\mathrm{D}}^e$ are independent of the mesh size, in the sense that they stay invariant when the domain and the hierarchical mesh are both scaled up or down.

Choice 1: discontinuous piecewise-polynomials on $\hat{\Omega}^e$
The first choice we propose is to pick $\mathbb{V}(\hat{\Omega}^e) = \mathbb{B}^{-1}_{\mathbf{p},\hat{\Omega}^e}$, where $\mathbb{B}^{-1}_{\mathbf{p},\hat{\Omega}^e}$ is defined to be the space of discontinuous piecewise-polynomials on $\hat{\Omega}^e$. With this choice, the projection $\hat{\Pi}_0^e$ can be formulated as a combination of L^2-projections onto $\mathbb{P}_{\mathbf{p}}(\Omega^{e'})$, see (7), for every $\Omega^{e'} \subset \hat{\Omega}^e$. In general, $\mathbb{V}(\hat{\Omega}^e) \supset \hat{\mathbb{T}}^e_{\Psi}$, and so the projection $\hat{\Pi}_1^e$ is equivalent to a least-squares problem and the matrix $\hat{\mathrm{D}}^e$ is a pseudo-inverse of the corresponding linear system. This choice most resembles the Bézier projector from [29] as it starts with local element-wise approximations that can be computed very efficiently and the matrix $\hat{\mathrm{D}}^e$ is constructed by collecting the transposed inverse Bézier extraction matrices of all the elements $\Omega^{e'} \subset \hat{\Omega}^e$. In practice, these inverses are not explicitly needed.

Choice 2: THB-splines on $\hat{\Omega}^e$
An alternative is to directly project onto the local THB-spline space by choosing $\mathbb{V}(\hat{\Omega}^e) = \hat{\mathbb{T}}^e_{\Psi}$. Here, the projection $\hat{\Pi}_0^e$ can be formulated as in (7) using the basis functions $T_j|_{\hat{\Omega}^e}$. Consequently, the projection $\hat{\Pi}_1^e$ is the identity and the corresponding matrix $\hat{\mathrm{D}}^e$ is an identity matrix. The numerical results of the next section indicate that for high degrees, this approach performs worse than choice 1 due to an ill-conditioned projection onto the THB-splines restricted to $\hat{\Omega}^e$.

Choice 3: global polynomials on $\hat{\Omega}^e$
As a last choice, one can pick $\mathbb{V}(\hat{\Omega}^e) = \mathbb{P}_{\mathbf{p}}(\hat{\Omega}^e)$. With this choice, the projection $\hat{\Pi}_0^e$ can be formulated exactly as in (7), albeit for a projection element $\hat{\Omega}^e$ instead of a mesh element Ω^e. Moreover, since $\mathbb{P}_{\mathbf{p}}(\hat{\Omega}^e) \subset \hat{\mathbb{T}}^e_{\Psi}$, the projection $\hat{\Pi}_1^e$ is the identity and knot insertion can be used to assemble the corresponding matrix $\hat{\mathrm{D}}^e$. However, we do not consider this choice because the resulting Π does not preserve THB-splines, and hence is not a projector. Note that this choice can still be used to define a THB-spline approximant and is similar in philosophy to [19]. To see that the latter is also not a projector, we can apply its reconstruction approach to a spline in the THB-spline space. In that case, restricted to Ω^e, the control points of the target spline are mapped to those of its approximation using the following map:

$$\left(\mathrm{L}^e\right)^T \left(\mathrm{L}^e \mathrm{C}^e\right)^{-T} \left(\mathrm{C}^e\right)^T, \tag{49}$$

where the so-called transmission matrix L^e encodes an intermediate projection onto a coarser space. Since $(\mathrm{C}^e)^T$ is a rectangular matrix with more columns than rows and $\mathrm{L}^e\mathrm{C}^e$ is square, in general the above map will not preserve the control points of the THB-spline.

4.3 A Priori Local Error Estimates

We begin this section by defining the support extensions of elements, and showing that the assumptions placed so far imply that the size of the support extension is bounded by the local mesh size.

Definition 4.1 We define the **support extension** of Ω^e, $e \in \mathcal{M}_\ell^{\text{in}}$, with respect to $\mathcal{T}_\Psi$ as:

$$\widetilde{\Omega}^e := \bigcup_{j \in \mathcal{I}^e(\mathcal{T}_\Psi)} \bigcup_{e' \in \hat{\mathcal{E}}_j(\mathcal{T}_\Psi)} \text{clos}\left(\hat{\Omega}^{e'}\right), \tag{50}$$

Define the **mesh size** associated to an element $\Omega^e = \bigtimes_{i=1}^n \left(\xi^i_{k^i,\ell}\ ,\ \xi^i_{k^i+1,\ell}\right)$, $e \in \mathcal{M}_\ell^{\text{in}}$, as

$$h^e := \max_i \left(\xi^i_{k^i+1,\ell} - \xi^i_{k^i,\ell}\right), \tag{51}$$

and of a projection element $\hat{\Omega}^e = \bigtimes_{i=1}^n \left(\xi^i_{k^i,\ell}\ ,\ \xi^i_{k^i+b^i,\ell}\right)$, as

$$\hat{h}^e := \max_i \left(\xi^i_{k^i+b^i,\ell} - \xi^i_{k^i,\ell}\right), \tag{52}$$

where b^i only depends on p^i. From Assumptions 3.1 and 3.2 placed on the hierarchical mesh, and by definition of the projection elements, there exists a constant C_0, dependent only on the degree, such that the mesh size of a projection element is

$$h^e \leq \hat{h}^e \leq C_0 h^e. \tag{53}$$

Let the corresponding support extension $\widetilde{\Omega}^e$ be contained in the smallest bounding box $\text{cube}(\widetilde{\Omega}^e) := \bigtimes_{i=1}^n \left(\widetilde{\xi}^i_0\ ,\ \widetilde{\xi}^i_1\right) \supseteq \widetilde{\Omega}^e$. Then, the mesh size for $\widetilde{\Omega}^e$ is defined as

$$\widetilde{h}^e := \max_i \left(\widetilde{\xi}^i_1 - \widetilde{\xi}^i_0\right). \tag{54}$$

Again, there is a constant, C_1, independent of h^e, such that

$$h^e \leq \widetilde{h}^e \leq C_1 h^e. \tag{55}$$

Using the above, we now state the following local error estimates for the Bézier projector.

Theorem 4.1 *For $e \in \mathcal{M}_\ell^{\text{in}}$, $0 \leq k \leq m \leq \min(\mathbf{p}) + 1$ and $f \in H^m(\widetilde{\Omega}^e)$:*

$$|f - \Pi f|_{H^k(\Omega^e)} \leq C (h^e)^{m-k} |f|_{H^m(cube(\widetilde{\Omega}^e))}, \tag{56}$$

where C is a constant independent of the mesh size h^e and cube$(\widetilde{\Omega}^e)$ is the smallest hypercube that contains $\widetilde{\Omega}^e$.

The proof for this theorem is an adaptation of the proof given in [29] and relies on standard polynomial approximation estimates, and spline-reproduction and stability of the projection Π. We first prove the latter results in Lemma 4.1, and thereafter provide a proof of Theorem 4.1.

Lemma 4.1 *The projector Π has the following properties:*

- $\Pi f = f$ *for* $f \in \mathbb{T}_\Psi$, *(spline reproduction)*
- $\|\Pi f\|_{L^2(\Omega^e)} \leq C_{stab} \|f\|_{L^2(\widetilde{\Omega}^e)}$ *for* $f \in L^2(\widetilde{\Omega}^e)$, *(local stability)*

where C_{stab} is independent of the mesh size h^e.

Proof Spline preservation follows trivially by construction since every step (and with Choices 1 and 2 for the local spaces) perfectly preserves the piecewise-polynomial representation of the THB-spline.[1] For local stability, observe that on Ω^e the global projection Πf combines the local approximations on the elements in the support extension $\widetilde{\Omega}^e$:

$$\Pi f|_{\Omega^e} = \sum_{j \in \mathcal{I}^e(\mathcal{T}_\Psi)} \left(\sum_{e \in \hat{\mathcal{E}}_j(\mathcal{T}_\Psi)} \hat{\omega}_j^e \hat{c}_j^e \right) T_j.$$

Then,

$$\left| \Pi f|_{\Omega^e} \right| \leq \max_{j \in \mathcal{I}^e(\mathcal{T}_\Psi)} \left| \sum_{e' \in \hat{\mathcal{E}}_j(\mathcal{T}_\Psi)} \hat{\omega}_j^{e'} \hat{c}_j^{e'} \right| \sum_{j \in \mathcal{I}^e(\mathcal{T}_\Psi)} T_j \leq \max_{\substack{j \in \mathcal{I}^e(\mathcal{T}_\Psi) \\ e' \in \hat{\mathcal{E}}_j(\mathcal{T}_\Psi)}} \left| \hat{c}_j^{e'} \right|, \tag{57}$$

where we have used partition-of-unity of THB-splines. For any given e, $\hat{c}_j^e$ are computed using (44) and, since the matrix $\hat{\mathrm{D}}^e$ is mesh size independent, $\|\hat{\mathrm{D}}^e\|_\infty$ is mesh-size independent. Moreover, the values $\hat{f}_j^e$ in (43) are computed via an L^2-projection. For an L^2-projection on a single element Ω^e to a finite element polynomial space that is non-negative and obeys partition of unity, we can bound the coefficients by

$$\max_k |f_k^e| \leq \frac{C}{\sqrt{(h^e)^n}} \|f\|_{L^2(\Omega^e)}, \tag{58}$$

where C is a constant independent of mesh size. See [29, Lemma A.5] for a proof for the Bernstein polynomials in case of choice 1. This proof can be trivially extended to hold for a finite element polynomial space which is non-negative and obeys partition of unity. For a projection element $\hat{\Omega}^e$, this bound becomes

[1] Choice 3 will instead preserve the polynomials of $\mathbb{P}(\hat{\Omega}^e)$.

$$\max_k |\hat{f}_k^e| \leq \frac{C}{\sqrt{(h^e)^n}} \|f\|_{L^2(\hat{\Omega}^e)}. \tag{59}$$

Here we used that the number of elements in a projection element is dependent on spline degree $\mathbf{p}$, and their size is bounded from below by h^e. Combining the above and integrating over Ω^e:

$$\|\Pi f\|_{L^2(\Omega^e)} \leq C_{\text{stab}} \|f\|_{L^2(\widetilde{\Omega}^e)},$$

for a constant C_{stab} that is independent of h^e. □

Proof (Theorem 4.1) For any polynomial $f_h \in \mathbb{P}_{\mathbf{p}}(\widetilde{\Omega}^e)$,

$$\begin{aligned} |f - \Pi f|_{H^k(\Omega^e)} &= |f - f_h + f_h - \Pi f|_{H^k(\Omega^e)} \\ &\leq |f - f_h|_{H^k(\Omega^e)} + |\Pi(f - f_h)|_{H^k(\Omega^e)} \\ &\leq |f - f_h|_{H^k(\widetilde{\Omega}^e)} + C_{\text{inv}} C_{\text{stab}} (h^e)^{-k} \|f - f_h\|_{L^2(\widetilde{\Omega}^e)}, \end{aligned}$$

where we have used the inverse inequality for polynomials with constant C_{inv} and local stability of Π from Lemma 4.1. Consequently, using standard polynomial approximation estimates for both terms on the hypercube cube$(\widetilde{\Omega}^e)$, we obtain the desired claim for a constant C that depends on C_{stab}, C_{inv} and the polynomial degree. □

5 Numerical Results

We perform several numerical experiments to validate our findings. Here we limit ourselves to the two dimensional THB-spline spaces of various polynomial degrees. First, the local estimate from Theorem 4.1 is numerically validated for Choice 1 and 2 from Sect. 4.2. Secondly, we introduce a global refinement scheme and perform tests to compare the proposed Bézier projection to [14].

5.1 Verification of Error Estimates

To validate the error estimates of Theorem 4.1, the target function $f(x, y) = \sin(\pi x)\sin(\pi y)$ is projected onto $\mathbb{T}_\Psi$ with $L = 2$ levels, where

$$\Omega_1 := \overline{\Omega}, \qquad \Omega_2 := \{(x, y) \in \Omega_1 : x \geq 0.5, y \geq 0.5\}. \tag{60}$$

On the above domain hierarchy, THB-spline spaces $\mathbb{T}_\Psi$ are constructed for each $\mathbf{p} \in \{(p, p) : p = 1, \ldots, 5\}$. The coarsest mesh consists of 2×2 elements, and the subsequent refinements are built by bisecting the meshes at all levels. The results for

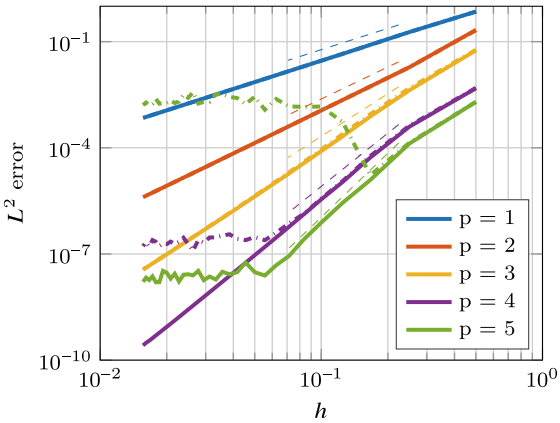


Fig. 11 Numerical Convergence rates for projection of $f(x, y) = \sin(\pi x)\sin(\pi y)$ onto the THB-spline space built for the domain hierarchy from (60). For both choice 1 (solid line) and choice 2 (dash dotted line) are compared to Theorem 4.1 (dashed line). The elements sizes per dimension vary between $h = 2^{-1}$ and 2^{-6} (for level 1) while ensuring an even number of elements. Note that choice 2 performs worse choice 1

both choice 1 and choice 2 can be seen in Fig. 11, the error converges optimally with respect to the mesh size. However, for choice 1 (for $p = 5$) and choice 2 (for $p = 4$ and $p = 5$), the accuracy stagnates for a small enough mesh size. This problem is worse for choice 2, as we have observed that the matrices $\hat{\mathrm{D}}^e$ from Sect. 4.1 exhibit a worse conditioning number for choice 2, compared to choice 1. For example, for degree 5 and for the projection element touching the lower-left corner of Ω_2, the condition number for the L^2 projection matrix for choice 2 is 1.7×10^{16}, while the matrix $\hat{\mathrm{D}}^e$ of choice 1 has a condition number of 2.4×10^7 (provided one uses a QR-decomposition).

5.2 *Adaptive Refinement Tests*

In this section, we compare our projector to the one from [14] on an adaptive refinement test borrowed from the latter. To do this, we propose an adaptive refinement scheme that ensures that the hierarchical refinements satisfy all assumptions required for the formulation of the Bézier projector.

The numerical results show comparable performance in relation to the results from [14]. For practical purposes, a first implementation of the refinement scheme was formulated for the restricted setting of quadratic and cubic THB-splines, and for a stronger set of assumptions that automatically imply Assumptions 3.3 and 3.4.

New assumptions on the mesh

The first new assumption is borrowed from [30] and it states that the refinement domains Ω_ℓ must be equal to the union of supports of a subset of level-$(\ell-1)$ splines.

Assumption 5.1 For all $\ell > 1$, $\exists S \subset \mathcal{B}_{\ell-1}$ such that

$$\Omega_\ell := \bigcup_{B_{\mathbf{j},\ell-1} \in S} \text{supp}\left(B_{\mathbf{j},\ell-1}\right), \quad \ell = 2, \ldots, L. \tag{61}$$

Assumption 5.1 was originally introduced so that the HB-spline functions can represent unity with strictly positive coefficients $w_{\mathbf{j},\ell} > 0$,

$$\sum_{(\mathbf{j},\ell):B_{\mathbf{j},\ell} \in \mathcal{H}_\Psi} w_{\mathbf{j},\ell} B_{\mathbf{j},\ell} = 1. \tag{62}$$

For the second assumption, we strengthen an assumption in [21]. For this, we introduce the **open complementary region**:

$$\Omega^c_{\ell+1} := \Omega \backslash \Omega_{\ell+1}. \tag{63}$$

Assumption 5.2 For each level $\ell = 1, \ldots, L-1$, and for any B-spline basis function $B_{\mathbf{j},\ell} \in \mathcal{B}_\ell$, the overlap defined as:

$$\text{overlap}\left(B_{\mathbf{j},\ell}\right) := \text{supp}\left(B_{\mathbf{j},\ell}\right) \cap \Omega^c_{\ell+1}, \tag{64}$$

is connected.

Assumption 5.2 is used in [21] to show that Hierarchical B-spline spaces are linearly independent and contains the space of piecewise polynomials over the same mesh.

The above two assumptions are sufficient for quadratic THB-splines. In the case either p^1 or p^2 is 3, projection elements can overlap (such as in Fig. 9). In this case, we impose the following additional assumption on the refinement domains Ω_ℓ.

Assumption 5.3 When either $p^1 = 3$ or $p^2 = 3$, and for each $\ell = 2, \ldots, L$, Ω_ℓ can be described as the following unions:

- if $p^1 = 3$ and $p^2 \leq 2$, then $\exists K_\ell^1 \subset \mathbb{N}^2$ such that:

$$\Omega_\ell = \bigcup_{(k^1,k^2)\in K_\ell^1} \left[\xi^1_{2k^1,\ell-1}, \xi^1_{2k^1+2,\ell-1}\right] \times \left[\xi^2_{k^2,\ell-1}, \xi^2_{k^2+1,\ell-1}\right]. \tag{65}$$

- if $p^1 \leq 2$ and $p^2 = 3$, then $\exists K_\ell^2 \subset \mathbb{N}^2$ such that:

$$\Omega_\ell = \bigcup_{(k^1,k^2)\in K_\ell^2} \left[\xi^1_{k^1,\ell-1}, \xi^1_{k^1+1,\ell-1}\right] \times \left[\xi^2_{2k^2,\ell-1}, \xi^2_{2k^2+2,\ell-1}\right]. \tag{66}$$

- if $p^1 = p^2 = 3$, then $\exists K_\ell^3 \subset \mathbb{N}^2$ such that:

$$\Omega_\ell = \bigcup_{(k^1,k^2)\in K_\ell^3} \left[\xi^1_{2k^1,\ell-1}, \xi^1_{2k^1+2,\ell-1}\right] \times \left[\xi^2_{2k^2,\ell-1}, \xi^2_{2k^2+2,\ell-1}\right]. \tag{67}$$

- if $p^1 \leq 2$, $p^2 \leq 2$, no change.

The following results, proved in the Appendix, show that these assumptions imply Assumptions 3.3 and 3.4 required for formulating the Bézier projector.

Proposition 5.1 *Assumptions 5.1, 5.2 and 5.3 imply Assumptions 3.3 and 3.4 in two dimensions.*

Adaptive refinement scheme

We now propose a simple refinement scheme consisting of four main routines:

- `Project` : builds THB-splines on the given mesh and applies the projector described in Sect. 4;
- `ElemError` : returns the maximum error for every mesh element;
- `MarkElem` : marks mesh elements for refinement based on Dörfler marking [10] and $0 < \theta < 1$;
- `ConformMesh` : updates the set of marked elements so that Assumptions 3.2, 5.1, 5.2 and 5.3 are satisfied.

The only non-standard routine here is `ConformMesh`. The pseudo-code for this is shown in Algorithm 1 and consists of three routines:

- `SupportCover` : for every level ℓ, initializes Ω_ℓ as a union of supports of $B_{\mathbf{j},\ell-1}$ that cover the marked level-ℓ elements;
- `GradeMesh` : conforms mesh to admissibility class 2 using [3];
- `ConnectedSupport` : for any ℓ and $B_{\mathbf{j},\ell-1} \in \mathcal{B}_{\ell-1}$ such that overlap$(B_{\mathbf{j},\ell-1})$ is not simply connected, marks the support of $B_{\mathbf{j},\ell-1}$ for refinement.

In addition, for cubic THB-splines, each of these routines is made to conform to Assumption 5.3.

```
input  : mesh to conform to Assumptions 3.2, 5.1, 5.2 and 5.3 while adding
         marked_elem.
output: mesh.
mesh ← SupportCover(marked_elem, mesh);
alter ← true;
while alter do
    mesh_new ← GradeMesh(mesh);
    mesh_new ← ConnectedSupport(mesh_new);
    if mesh_new = mesh then
        alter ← false;
    end
    mesh ← mesh_new;
end
```

Algorithm 1: Schematic for `ConformMesh`

Comparison with [14]

The performance of our Bézier projector with the proposed adaptive refinement scheme is compared to the local THB-spline projector introduced in [14]. As a test case, we use the following target function:

$$f(x, y) = 1 - \tanh\left(\frac{\sqrt{(2x-1)^2 + (2y-1)^2} - 0.3}{0.05\sqrt{2}}\right), \quad (x, y) \in [0, 1]^2. \quad (68)$$

The same function was used in [14, Example 1] albeit by transforming it from $[0, 1]^2$ to a scaled and translated domain $[-1, 1]^2$. This will not cause any problems for our comparison as we will measure the errors in the L^∞ norm.

As in [14], the quadratic scheme is allowed to refine up to level 5, while the cubic case refines up to level 4. In Fig. 12a, b, the maximum element error for our refinement scheme can be seen for the quadratic and cubic case for various choices of θ. The final refinement domain for the quadratic case and $\theta = 0.5$ can be seen in Fig. 13a, and for the cubic case and $\theta = 0.75$ in Fig. 13b. In [14], they use an adapted Dörfler marking scheme for $\theta = 1$, where they additionally add extra elements to control the locality of refinement.

While in this paper we focus on the characterization of pseudo-local linear independence for THB-splines, we can still compare our proposed projector to [14]. The main difference being that our method relies on Bézier extraction, for which efficient implementations exists, for instance, [8]. In the case of [14], for every degree of freedom, a local THB-spline projection is required. While the projection matrices can be precomputed (as can ours), this local projection is required for every DOF separately. In contrast to our method where each projection elements indirectly projects

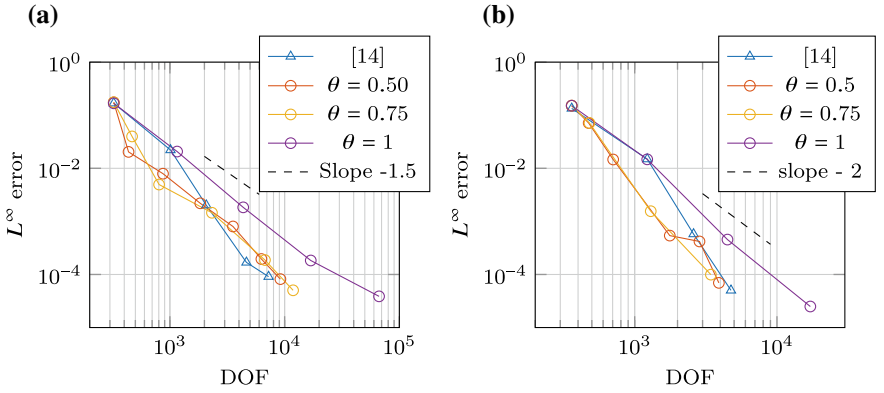


Fig. 12 Maximum element error for the adaptive refinement of Eq. (68) in the quadratic case **a** and the cubic case **b**

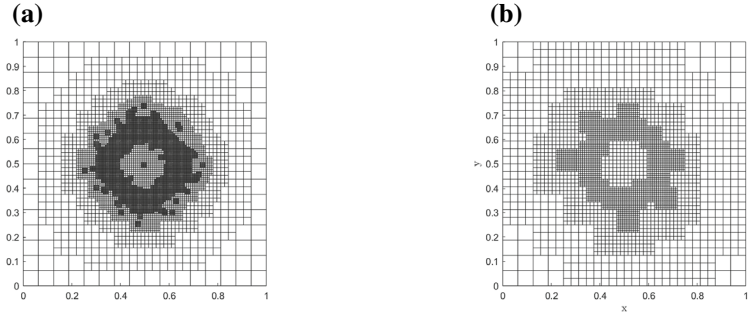


Fig. 13 The final meshes for $\theta = 0.5$ in the quadratic case **a** and for $\theta = 0.75$ in the cubic case **b**

to multiple DOF at a time, but multiple projection elements need to be computed before the final coefficient can be computed. The biggest advantage for [14], is that it requires less strict assumptions on the mesh.

6 Conclusion

Truncated Hierarchical B-spline spaces suffer from overloading of mesh elements, for example, there exist mesh elements restricted to which the basis functions are linearly dependent. In this paper, we considered these spaces in an arbitrary number of spatial dimensions. For hierarchically refined meshes that satisfy certain assumptions, we showed that we can identify the elements that are not overloaded. Using those non-overloaded elements, we partitioned the entire mesh into a set of local macro-elements, none of which are overloaded. These macro-elements, which we

call projection elements, are local in the sense that they consists of adjacent elements, and the number of elements in each macro-element solely depends on the spline degree.

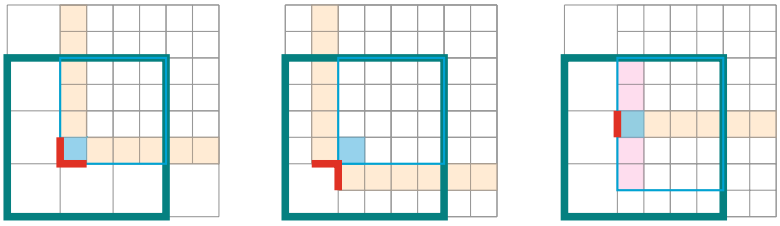
Using these projection elements, we extended the Bézier projector from [29] to the Truncated Hierarchical B-spline spaces. In particular, the notion of projection elements is useful for performing local projections onto the spline space restricted to those macro-elements. These local projections can be chosen in different ways, and we outlined some simple choices that stay close to the original Bézier projector from [29].

Finally, we derived optimal local error estimates for our proposed projector. These estimates were numerically validated in a two-dimensional setting. In addition, a first adaptive refinement scheme was proposed for quadratic and cubic splines that produces hierarchically refined meshes conforming to our assumptions. Note that this refinement scheme was only proposed for performing numerical comparison with the results of [14]; theoretical results (for example, on its complexity) are outside the scope of this paper. The results showed that the performance of the Bézier projector and the proposed refinement scheme are comparable to that of [14].

There are several generalizations of our results that may be of interest in applications, we list a couple of them here. First, we only considered hierarchical meshes of admissibility class 2 [3]; extending the characterization of local linear independence to higher admissibility classes will allow one to work with more general hierarchical refinements. Another interesting open question is formulation of refinement schemes with linear complexity, simple implementations, that can yield meshes conforming to our assumptions, and that work for arbitrary choices of degrees in both 2D and 3D. These and other questions will be the focus of our future work.

7 Appendix: Proof of Proposition 5.1

Assumption 3.3:
We first show that Assumptions 5.1 and 5.2 imply Assumptions 3.3 in two dimensions and spline degree $\mathbf{p} = (p^1, p^2)^T$. Here, there are three kinds of well-behaved border elements shown in the schematics below.



$d^1_\ell(e) = d^2_\ell(e) = 0$ $\quad$ $d^1_\ell(e) > p^1,\ d^2_\ell(e) > p^2$ $\quad$ $d^1_\ell(e) = 0,\ d^2_\ell(e) > p^2$

For the first two cases (left and centre), the well-behaved border elements shown in blue and the respective domain specified by Assumption 3.3 is shown by blue line. Due to Assumption 5.1, if $d_\ell^i(e) = 0$, the adjacent $2p^i + 2$ elements, must be contained in Ω_ℓ, which are shown in orange. Then, consider the level-$(\ell - 1)$ B-spline with the boundary of its support displayed with bold, teal lines (these are chosen to encompass the domain of Assumption 3.3 and the level-$(\ell - 1)$ elements sharing the border $\partial\Omega_\ell$ with the well-behaved border element of interest). Then, the level-$(\ell - 1)$ elements in its support and fenced by the light blue border must be contained in Ω_ℓ due to Assumption 5.2, and therefore Assumption 3.3 is satisfied. For the last case (right), assume without loss of generality that $d_\ell^2(e) > p^2$. In this case, $d_\ell^2(e) > p^2$, forces the pink coloured elements to be refined. While the orange elements are refined by the same reasoning as the other cases. In this setting, again the level-$(\ell - 1)$ elements contained in the domain specified by Assumption 3.3, must again all be contained in Ω_ℓ, as otherwise Assumption 5.2 is violated.

Assumption 3.4:
We will now show that Assumptions 5.1, 5.2 and 5.3 imply Assumptions 3.4 in two dimensions for the quadratic and cubic cases. We start by showing that for any of the possibly overloaded elements, there **exists** a projection element containing it. Lastly, we will show that no two projection elements can overlap, hence showing **uniqueness**.

Existence
As in the premise of Assumption 3.4, consider an element $\Omega^{e'}$ for which we want to show the claim: the existence of Ω^e such that $\Omega^{e'} \subset \hat{\Omega}^e$. Let $\mathbf{t}$ be the minimal translation in l^1 norm to an element of $\mathcal{M}_\ell \backslash \mathcal{M}_\ell^{\text{in}}$. Without loss of generality, assume $t^i \leq 0$ for all i.

If both entries t^i are non-zero, then for $\Omega^e = \tau_{\mathbf{t}+[1,1]^T}(\Omega^{e'})$, we have:

- $\partial\Omega^e \cap (\partial\Omega_\ell \backslash \partial\Omega) \neq \emptyset$
- $d_\ell^i(e) > 0$ for all $i = 1, 2$.

But then, due to Assumption 5.3, for both the cubic and quadratic case:

$$d_\ell^i(e) > p^i, \qquad i = 1, 2. \tag{69}$$

Hence $\Omega^{e'}$ is contained in the projection element generated by well-behaved border element Ω^e.

In case both entries t^i are zero, $d_\ell^i(e') = 0$ and $\Omega^{e'}$ is a well-behaved border element and contained in the projection element $\hat{\Omega}^{e'}$.

The last case is less trivial, assume without loss of generality that $t^1 = 0$. Then the element at translation $\mathbf{t}_1 = (0, t^2 + 1)$ is the first candidate for Ω^e. Denote this element as $\Omega^{e^*} = \tau_{\mathbf{t}_1}(\Omega^{e'})$. If $d_\ell^1(e^*) > p^1$ or $d_\ell^1(e^*) = 0$, $\Omega^{e'} \subset \hat{\Omega}^{e^*}$. Else, assuming without loss of generality that the translation for which $d_\ell^1(e) > p^1$ is in the negative x-direction, we claim that the element at translation $t_2 = (-d_\ell^1(e^*), t^2 + 1)$ is the

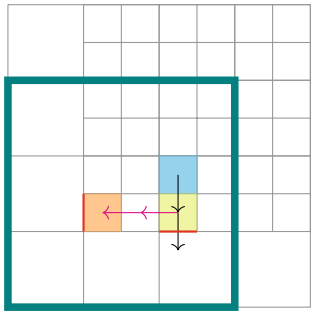


Fig. 14 For the starting element $\Omega^{e'}$ shown in blue, the candidate shown in green does not satisfy $d^1_\ell(e^*) > 3 = p^1$. Hence, the orange element is the true candidate Ω^e for which $d^1_\ell(e) = d^2_\ell(e) = 0$. If this were false, the teal B-spline would violate Assumption 5.2

desired well-behaved border element $\Omega^e = \tau_{t_2}(\Omega^{e'})$. See Fig. 14 for this case, the black arrows are the translation vector t_1, while the combination of the black and pink are t_2. This element Ω^e must be a well-behaved border elements, as otherwise Assumption 5.2 is violated for the level-$(\ell - 1)$ B-spline indicated.

Uniqueness:
To show uniqueness, it is sufficient to show that any arbitrary projection element cannot overlap with any other. In 2D, there are four possible projection elements. Namely, those generated by the condition that $d^i_\ell(e) = 0$ or $d^i_\ell(e) > p^i$ for $i = 1, 2$.

- $d^1_\ell(e) = 0,\ d^2_\ell(e) > p^2$ or $d^2_\ell(e) = 0,\ d^1_\ell(e) > p^1$:
  For this case, Assumption 3.3 ensures the projection element $\hat{\Omega}^e$ is unique. See Fig. 15a, the domain stated by Assumption 3.3 consists of projection elements, none of which can overlap with $\hat{\Omega}^e$. Notice that in Fig. 15a, for a different projection

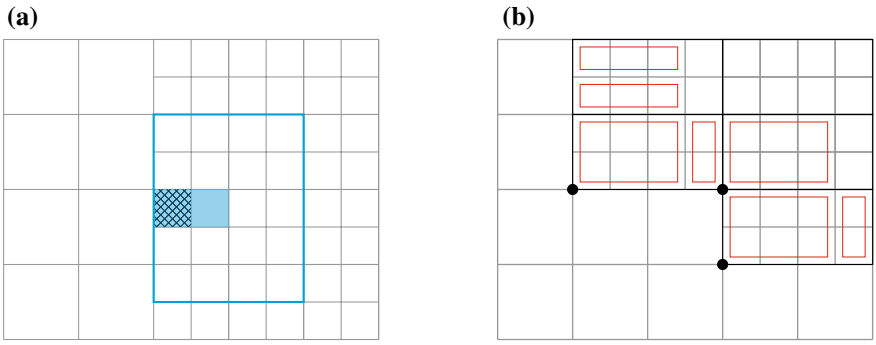


Fig. 15 **a** For a given well-behaved border element indicated by the crosshatch pattern with either $d^1_\ell(e) = 0,\ d^2_\ell(e) > p^2$ or $d^2_\ell(e) = 0,\ d^1_\ell(e) > p^1$, Assumption 3.3 guarantees that no other projection elements can overlap the projection element filled in with blue. **b** Assumption 5.3 makes sure that there is enough "space" between any vertex of Ω_ℓ. These vertices produce projection elements which are fully contained in the "blocks" from Assumption 5.3 (shown in black), hence non-overlapping

element $\hat{\Omega}^{e_1}$, to overlap with $\hat{\Omega}^e$, some part of $\partial\Omega^{e_1} \cap \partial\Omega_\ell \backslash \partial\Omega$ must lie within the domain specified by Assumption 3.3, which would violate Assumption 3.3.

- $d_\ell^1(e) = 0,\ d_\ell^2(e) = 0$ or $d_\ell^1(e) > p^1,\ d_\ell^2(e) > p^2$: the previous bullet, it suffices to check against other projection elements of these types. However, in 2D, these are all generated by the vertices of Ω_ℓ, and span p^i elements in each direction. Depending on spline degree, these are contained in either 1×1, 1×2, 2×1 or 2×2 level-$(\ell - 1)$ elements (since we are only considering at most bi-cubic splines). By Assumption 5.3, these blocks (generated by the vertices) can never overlap in conjunction with Assumption 5.1, hence these projection elements do not overlap. See Fig. 15b for an example in case of $p^1 = 3$, $p^2 = 2$ where the relevant blocks from Assumption 5.3 are given in red.

Acknowledgements K. Dijkstra is supported by the Peter Paul Pietrich Fellowship 2023–2026. The research of Deepesh Toshniwal is supported by project number 212.150 awarded through the Veni research programme by the Dutch Research Council (NWO).

References

1. Beirão da Veiga, L., Buffa, A., Rivas, J., Sangalli, G.: Some estimates for h-p-k-refinement in Isogeometric Analysis. Numerische Mathematik **118**(2), 271–305 (2011)
2. Borden, M.J., Scott, M.A., Evans, J.A., Hughes T.J.R.: Isogeometric finite element data structures based on Bézier extraction of NURBS. Int. J. Numer. Methods Eng. **87**(1–5):15–47 (2011)
3. Bracco, C., Giannelli, C., Vázquez, R.: Refinement algorithms for adaptive isogeometric methods with hierarchical splines. Axioms **7**(3) (2018). Publisher: MDPI AG
4. Bressan, A.: Some properties of LR-splines. Comput. Aided Geom. Des. **30**(8), 778–794 (2013)
5. Buffa, A., Cho, D., Sangalli, G.: Linear independence of the T-spline blending functions associated with some particular T-meshes. Comput. Methods Appl. Mech. Eng. **199**(23–24), 1437–1445 (2010)
6. Buffa A, Giannelli, C.: Adaptive isogeometric methods with hierarchical splines: error estimator and convergence. Math. Models Methods Appl. Sci. **26**(1), 1–25 (2016). arXiv: 1502.00565 Publisher: World Scientific Publishing Co. Pte Ltd
7. Austin Cottrell, J., Hughes, T.J.R., Bazilevs, Y.: Isogeometric Analysis. Wiley-Blackwell, Hoboken, NJ (2009)
8. D'Angella, D., Kollmannsberger, S., Rank, E., Reali, A.: Multi-level Bézier extraction for hierarchical local refinement of Isogeometric Analysis. Comput. Methods Appl. Mech. Eng. **328**, 147–174 (2018). Publisher: Elsevier B.V
9. Dokken, T., Lyche, T., Pettersen, K.F.: Polynomial splines over locally refined box-partitions. Comput. Aided Geom. Des. **30**(3), 331–356 (2013). Publisher: Elsevier B.V
10. Dörfler, W.: A convergent adaptive algorithm for poisson's equation. SIAM J. Numer. Anal. **33**(3), 1106–1124 (1996). https://doi.org/10.1137/0733054
11. D'Angella, D., Reali, A.: Efficient extraction of hierarchical B-Splines for local refinement and coarsening of isogeometric analysis. Comput. Methods Appl. Mech. Eng. **367**, 113131 (2020)
12. Forsey, D.R., Barrels, R.H.: Hierarchical B-spline refinement. SIGGRAPH Comput. Graph. **22**(4), 205–212 (1988). Issue: 4 Publication Title: Computer Graphics Volume: 22
13. Carlotta Giannelli, B., Speleers, H.: THB-splines: the truncated basis for hierarchical splines. Comput. Aided Geom. Des. **29**(7), 485–498 (2012). Issue: 7 ISSN: 01678396
14. Giust, A., Jüttler, B., Mantzaflaris, A.: Local (T)HB-spline projectors via restricted hierarchical spline fitting. Comput. Aided Geom. Des. **80**, 101865 (2020)

15. Greiner, G., Hormann, K.: Interpolating and approximating scattered 3D data with hierarchical tensor product splines. In: Surface Fitting and Multiresolution Methods (1998)
16. Hughes, T.J.R., Cottrell, J.A., Bazilevs, Y.: Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. Comput. Methods Appl. Mech. Eng. **194**(39–41), 4135–4195 (2005)
17. Johannessen, K.A., Kvamsdal, T., Dokken, T.: Isogeometric analysis using LR B-splines. Comput. Methods Appl. Mech. Eng. **269**, 471–514 (2014)
18. Li, X., Sederberg, T.W.: S-splines: a simple surface solution for IGA and CAD. Comput. Methods Appl. Mech. Eng. **350**, 664–678 (2019). Publisher: Elsevier B.V
19. Lorenzo, G., Scott, M.A., Tew, K., Hughes, T.J.R., Gomez, H.: Hierarchically refined and coarsened splines for moving interface problems, with particular application to phase-field models of prostate tumor growth. Comput. Methods Appl. Mech. Eng. **319**, 515–548 (2017). https://doi.org/10.1016/j.cma.2017.03.009
20. Lyche, T., Manni, C., Speleers, H.: Foundations of Spline Theory: B-Splines, Spline Approximation, and Hierarchical Refinement, vol. 2219. Springer International Publishing (2018). Publication Title: Lecture Notes in Mathematics ISSN: 00758434
21. Mokriš, D., Jüttler, B., Giannelli, C.: On the completeness of hierarchical tensor-product B-splines. J. Comput. Appl. Math. **271**, 53–70 (2014)
22. Patrizi, F., Dokken, T.: Linear dependence of bivariate Minimal Support and locally refined B-splines over LR-meshes. Comput. Aided Geom. Des. **77**, 101803 (2020)
23. Kraft, R.: Adaptive und linear unabh"angige Multilevel B-Splines und ihre Anwendungen. Ph.D. thesis, Univ. Stuttgard, Stuttgard, 1998. Publication Title: PhD Thesis
24. Sande, E., Manni, C., Speleers, H.: Explicit error estimates for spline approximation of arbitrary smoothness in isogeometric analysis. Numerische Mathematik **144**(4), 889–929 (2020). arXiv: 1909.03559 Publisher: Springer Berlin Heidelberg
25. Scott, M.A., Borden, M.J., Verhoosel, L.V., Sederberg, T.W., Hughes, T.J.R.: Isogeometric finite element data structures based on Bézier extraction of T-splines. Int. J. Numer. Methods Eng. **88**(2), 126–156 (2011)
26. Sederberg, T.W., Zheng, J., Bakenov, A., Nasri, A.: T-Splines and T-NURCCs. ACM Trans. Graph. **22**(3), 477–484 (2003). Place: New York, NY, USA Publisher: Association for Computing Machinery
27. Speleers, H.: Hierarchical spline spaces: quasi-interpolants and local approximation estimates. Adv. Comput. Math. **43**(2), 235–255 (2017)
28. Speleers, H., Manni, C.: Effortless quasi-interpolation in hierarchical spaces. Numerische Mathematik **132**(1), 155–184 (2016)
29. Thomas, D.C., Scott, M.A., Evans, J.A., Tew, K., Evans. , E.J.: Bézier projection: a unified approach for local projection and quadrature-free refinement and coarsening of NURBS and T-splines with particular application to isogeometric design and analysis. Comput. Methods Appl. Mech. Eng. **284**, 55–105 (2015). arXiv: 1404.7155 Publisher: Elsevier B.V
30. 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**(49–52), 3554–3567 (2011)