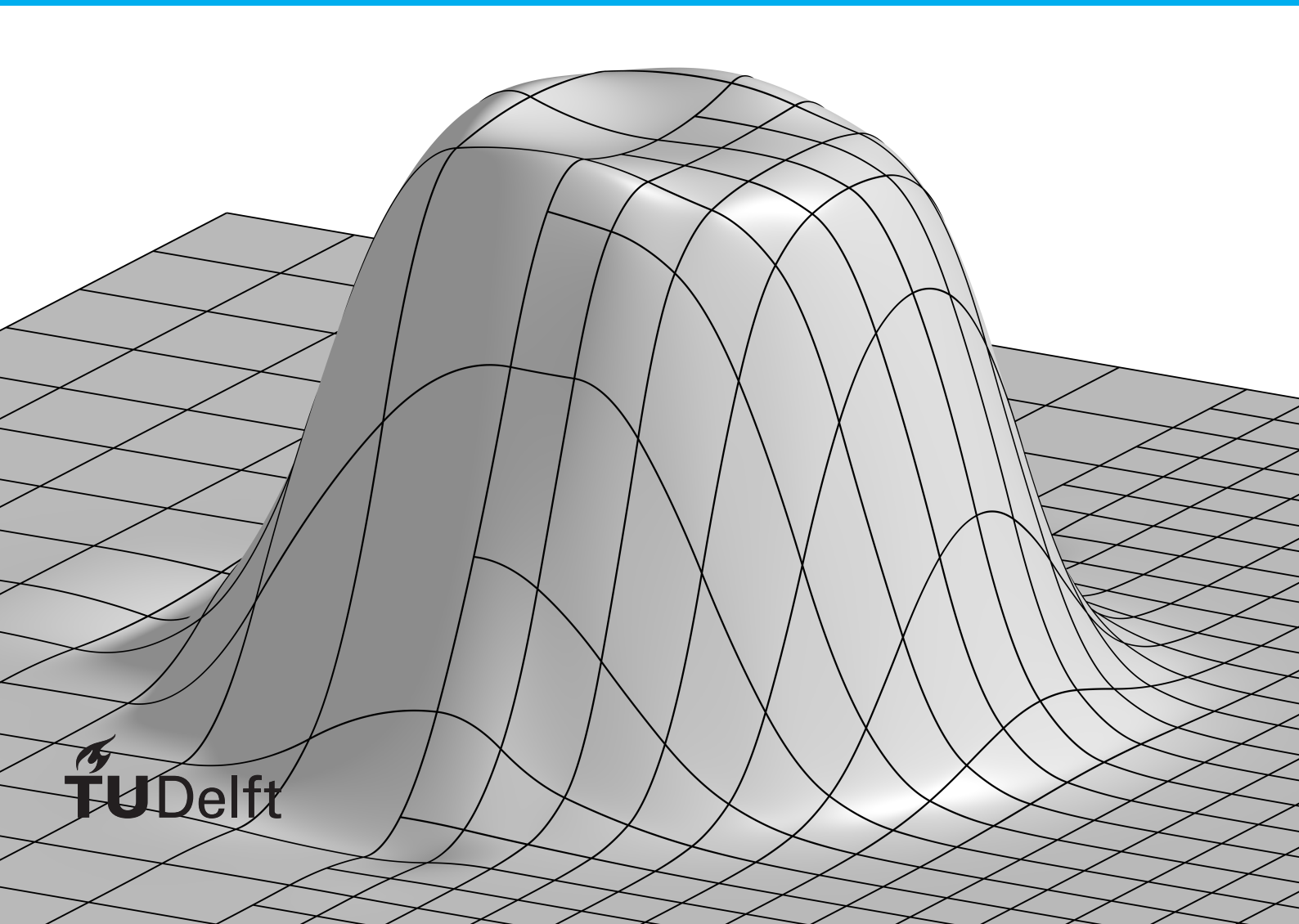
Local (Commuting) THB-spline Projectors

K.W. Dijkstra



TU Delft

Local (Commuting) THB-spline Projectors

by

K.W. Dijkstra

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Wednesday November 30, 2022 at 10:00 AM.

Student number: 4558235
Project duration: December 1, 2021 – November 30, 2022
Thesis committee: Dr. D. Toshniwal, TU Delft, supervisor
Prof. Dr. K. Vuik, TU Delft
Dr. P. Visser, TU Delft

This thesis is confidential and cannot be made public until November 30, 2023.

TU Delft

Contents

1

Introduction

In recent years, isogeometric analysis 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 (DOF) (Beirão da Veiga et al., 2011; Sande et al., 2020). Additionally, this allows the domain to be more accurately imported from Computer Aided Design (CAD) software, which results in a reduction/elimination of domain meshing errors (Cottrell et al., 2009; Hughes et al., 2005). Isogeometric analysis can be used to solve problems with tools like ("FEAP", n.d.). However, at this time, these tools are only used for research purposes. For this reason, B-splines are commonly used as basis functions. However, these B-splines are unable to be refined locally. For this reason Truncated Hierarchical B-splines have been developed (Lyche et al., 2018).

One of the tools which can be used to analyse the finite element method is Finite Element Exterior Calculus (D. N. Arnold, 2018). This approach helps us design and understand why certain methods work better than others. Finite Element Exterior Calculus concerns itself with the hodge Laplacian. This is a generalization of the Laplace problem in higher dimensions. The hodge Laplacian shows up in many problems. One can formulate the hodge Laplace problem as Stokes flow or the Maxwell equations. In (D. N. Arnold, 2018), they show that the mixed weak formulation of the hodge Laplacian is consistent, stable and converges, under certain assumption. One of these assumption being, existence of a commuting projector. A commuting projector is an operator that commutes with the (exterior) derivative, when mapping from the infinite dimensional complex to the finite dimensional complex. To give an example, initially projecting a H^1 function on to the finite dimensional space and then calculating the divergence must give the same result as first calculating the divergence and then projecting it onto the finite dimensional space. Lastly, in order for the projector to be efficient in practical use, we desire the projector to be local. This means that the coefficient of a basis function are only dependent on function data around the support of the basis function. For these reasons, it is interesting to develop (or at least show existence of) Local commuting Truncated Hierarchical B-spline projectors. These projectors do currently not exist in literature. Truncated Hierarchical B-spline projectors exist, see for example (Giust et al., 2020) and there exists local L^2 bounded commuting FEEC projectors (D. Arnold & Guzmán, 2021). However, no local commuting Truncated Hierarchical B-spline projectors exist in academic literature.

In this paper, we introduce a Truncated Hierarchical B-spline projector by extending the local Bezier projector from (Thomas et al., 2015) to Truncated Hierarchical B-splines. For this, a local notion of linear independence is required for the Truncated Hierarchical B-spline spaces. Additionally, by using the same approach as in (Buffa et al., 2011) for the 1D setting, a commuting local Truncated Hierarchical B-spline projector is constructed. In Chapter 2 the Truncated Hierarchical B-spline space is summarized followed by an overview of Finite Element Exterior Calculus in Chapter 3. In Chapter 4 the commuting Truncated Hierarchical B-spline projector is introduced for the univariate case and in Chapter 5 the Truncated Hierarchical B-spline projector for the multivariate case. Chapter 6 contains numerical results where we compare our proposed projector to (Giust et al., 2020) in the 1D setting and show that the theoretical convergence results hold. We conclude with our findings in Chapter 7 of this projector.

2

General Spline Theory

In this chapter we will introduce splines, in particular one-dimensional B-splines. At the end of this chapter we will expand this notion to higher dimensional splines by taking a tensor product of multiple one dimensional splines and locally refine these splines spaces to create (T)HB-splines. This section is highly inspired by (Lyche et al., 2018). Many Theorem are taken from (Lyche et al., 2018), and for proofs of all the claims in this section, we refer the reader to (Lyche et al., 2018).

2.1. B-spline

Definition 2.1. A **knot-sequence** $\boldsymbol{\xi}$ is a sequence of non-decreasing real numbers,

$$\boldsymbol{\xi} = \{\xi_i\}_{i=1}^{m} = \{\xi_1 \leq \xi_2 \leq \cdots \leq \xi_m\}, \qquad m \in \mathbb{N}$$

The elements ξ_i are called **knots**.

Assuming that $m \geq p + 1 \geq 0$, we can define B-splines of degree p over the knot-sequence $\boldsymbol{\xi}$:

Definition 2.2. Given a non-negative integer p and an integer j such that $\xi_j \leq \xi_{j+1} \leq \cdots \leq \xi_{j+p+1}$ are $p + 2$ real numbers taken from a knot-sequence $\boldsymbol{\xi}$. Then the j^{th} **B-spline** is defined to be zero in the case $\xi_j = \xi_{j+p+1}$, otherwise recursively by:

$$b_{j,p,\xi}(x) := \frac{x - \xi_j}{\xi_{j+p} - \xi_j} b_{j,p-1,\xi}(x) + \frac{\xi_{j+p+1} - x}{\xi_{j+p+1} - \xi_{j+1}} b_{j+1,p-1,\xi}(x) \tag{2.1}$$

Starting with:

$$b_{j,0,\xi}(x) := \begin{cases} 1, & \text{if } x \in [\xi_j, \xi_{j+1}), \\ 0, & \text{otherwise.} \end{cases}$$

Here, the convention is to let the fraction be zero, in case of a zero denominator.

These B-splines have the following properties:

Property 2.3. Given a knot-sequence $\boldsymbol{\xi}$ of m knots, a degree p and $1 \leq j \leq n = m - p - 1$. Then the n B-splines have the following properties:

- **Local support**. A B-spline is locally supported:

$$b_{j,p,\xi}(x) = 0, \quad x \notin [\xi_j, \xi_{j+p+1})$$

- **Non-negativity**. All n B-splines are non negative everywhere and strictly positive on the open local support. Meaning:

$$b_{j,p,\xi}(x) \geq 0, x \in \mathbb{R}, \quad b_{j,p,\xi}(x) > 0, x \in (\xi_j, \xi_{j+p+1})$$

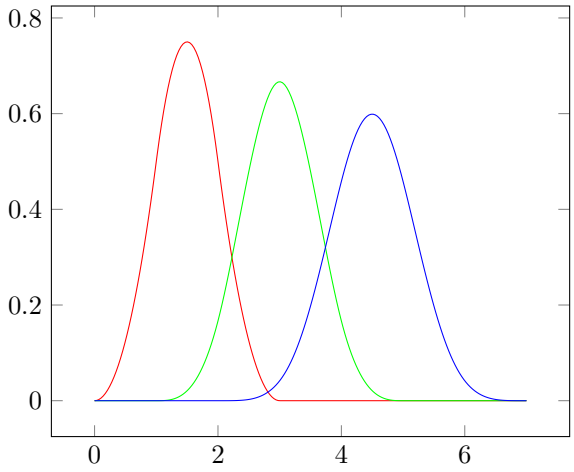


Figure 2.1: An example of a B spline. These are second degree B splines defined over the knot-sequence $\boldsymbol{\xi} = \{0, 1, 2, 3, 4, 5, 6, 7\}$ and $j \in \{1, 2, 3\}$.

- **Piecewise Structure**. B-splines have a polynomial piecewise structure. Specifically, restricting a B-spline to $I_i = [\xi_i, \xi_{i+1})$, the B-spline is a polynomial of degree p:

$$b^i_{j,p,\boldsymbol{\xi}}(x) := b_{j,p,\boldsymbol{\xi}}(x)_{|_{I_i}} \in \mathbb{P}_p(I_i), \quad i =, j, j+1, \dots, j+p$$

  where $\mathbb{P}_p(\Omega)$ denotes the space of polynomials of degree up to p defined over domain Ω.

- **Local Partition of Unity**. The B-splines with support on $I_k = [\xi_k, \xi_{k+1})$, for $p+1 \leq k \leq n$, sum to 1:

$$\sum_{i=k-p}^{k} b_{i,p,\boldsymbol{\xi}}(x) = 1, \quad x \in I_k = [\xi_k, \xi_{k+1})$$

- **Local Linear Independence**. Given a domain $I_k = [\xi_k, \xi_{k+1})$ containing at least $p+1$ distinct points, the set $\{b_{i,p,\boldsymbol{\xi}}(x)\}_{i=k-p}^{k}$ forms a basis for the polynomial space $\mathbb{P}_p(I_k)$.

An example of B-splines can be seen in Figure 2.1. Here, three B-splines are shown for the knot-sequence $\boldsymbol{\xi} = \{1, 2, 3, 4, 5, 6, 7\}$ of different degrees.

2.1.1. Smoothness, Differentiation and Integration

B-splines are a way to enforce higher regularity at element boundaries. For example, as we have seen, B-splines are locally a degree p polynomial and thus, p regular on every subdomain $I_k = [\xi_k, \xi_{k+1})$. However, they also exhibit high regularity at boundaries. To show this, the concept of multiplicity is introduced. This is a counting function μ that counts the amount of knots present in the knot-sequence. It is defined as:

Definition 2.4. The counting function $\mu : \mathbb{R} \to \mathbb{N}$ is defined as follows:

$$\mu(x) := \begin{cases} m, & \text{if } x \text{ appears } m \text{ distinct times in the knot-sequence } \boldsymbol{\xi}, \\ 0, & \text{otherwise.} \end{cases} \tag{2.2}$$

We will also call a knot ξ_m with multiplicity $\mu = \mu(\xi_m)$ a knot of multiplicity μ.

B-splines exhibit the following regularity at the knots of knot-sequence:

Theorem 2.1. *If ξ_m is a knot of the knot-sequence $\boldsymbol{\xi}$ of multiplicity $\mu(\xi_m) \leq p+1$, then:*

$$b_{j,p,\boldsymbol{\xi}} \in C^{p-\mu(\xi_m)}(\xi_m), \tag{2.3}$$

From this theorem, we deduce that the regularity of B-splines can be influenced by choosing an appropriate knot-sequence. Additionally, if all knots are chosen to be of multiplicity 1, we see that the B-splines are $p-1$ regular.

Additionally, the derivative of a B-spline can be explicitly described. For this, we denote the right derivative with D_+ and the left derivative by D_-. In this case, we the derivative is given as:

Theorem 2.2. *Given a knot-sequence $\boldsymbol{\xi}$ and degree $p > 1$. The right derivative of a B-spline $b_{j,p,\boldsymbol{\xi}}(x)$ is given by:*

$$D_+ b_{j,p,\boldsymbol{\xi}}(x) = p\left(\frac{b_{j,p-1,\boldsymbol{\xi}}(x)}{\xi_{j+p} - \xi_j} - \frac{b_{j+1,p-1,\boldsymbol{\xi}}(x)}{\xi_{j+p+1} - \xi_{j+1}}\right), \quad p > 1 \tag{2.4}$$

where fractions with zero denominator are taken to be zero.

2.1.2. B-spline function space

Given a domain $\Omega = [a, b]$, we can partition this space with a knot-sequence ξ where $\xi_1 = a$ and $\xi_m = b$ with m being the number of knots. We will introduce terminology with respect to the knot-sequence, which is crucial for our study/construction of the B-spline spaces.

Definition 2.5. We call a knot-sequence $\boldsymbol{\xi}$ of m knots with $n = m - p - 1$:

- $(p+1)$**-regular** if $\xi_j < \xi_{j+p+1}$ for all $j = 1, \dots, n$.
- $(p+1)$**-basic** if it is $(p+1)$ regular and $\xi_{p+1} < \xi_{p+2}$ and $\xi_n < \xi_{n+1}$.
- $(p+1)$**-open** on an interval $[a, b]$ if it is $(p+1)$-regular and the end knots have multiplicity $p+1$. Thus:
$$a := \xi_1 = \cdots = \xi_{p+1} < \xi_{p+2} \leq \cdots \leq \xi_n < \xi_{n+1} = \cdots = \xi_{n+p+1} =: b$$

Definition 2.6. Given a knot-sequence $\boldsymbol{\xi}$ of m knots and a degree p. Then, the B-spline space can be defined as $\mathbb{S}_{p,\boldsymbol{\xi}}$ of $n = m - p - 1$ B-splines:

$$\mathbb{S}_{p,\boldsymbol{\xi}} = \operatorname{span}\left\{b_{i,p,\boldsymbol{\xi}}(x)\right\}_{i=1}^{n}. \tag{2.5}$$

Additionally, define the basis of the B-spline space as $\mathcal{S}_{p,\boldsymbol{\xi}} = \{b_{i,p,\boldsymbol{\xi}}(x)\}_{i=1}^{n}$.

In practice, we will usually use a $(p+1)$-open knot-sequence. In this case, the spline space $\mathbb{S}_{p,\boldsymbol{\xi}}$ will be **linearly independent**. This is a consequence of the following theorem:

Theorem 2.3. *(Lyche et al., 2018, Sec. 1.3.1)*
If $\boldsymbol{\xi}$ is $(p+1)$-basic, then the spline space $\mathbb{S}_{p,\boldsymbol{\xi}}$ is linear independent over the basic domain.

The spline space $\mathbb{S}_{p,\boldsymbol{\xi}}$ seems arbitrary, but in fact, this space coincides with the space $\mathbb{S}_p^r(\Delta)$, the space of piecewise polynomials of degree p by a given sequence of break points and some prescribed smoothness. However, for the two spaces to coincide, the knot-sequence must be suitably chosen from the break points and the smoothness conditions.

The space of piecewise polynomials $\mathbb{S}_p^r(\Delta)$ is defined as follows. Let $\Delta := \{\eta_0 < \eta_1 < \cdots < \eta_{l+1}\}$ be a sequence of distinct real numbers that describe the bounds of the elements of the piecewise polynomials. Moreover, let $\mathbf{r} := (r_1, \dots, r_{l+1})$ be a vector of integers for which $-1 \leq r_i \leq p$ and where each r_i describes the regularity at boundary η_i. Then the space of piecewise polynomials is defined as:

$$\begin{aligned}\mathbb{S}_p^{\mathbf{r}}(\Delta) := \{s : [\eta_0, \eta_{l+1}] \to \mathbb{R} | s \in \mathbb{P}_p([\eta_i, \eta_{i+1})), i = 0, \dots, l-1, \\ s \in \mathbb{P}_p([\eta_l, \eta_{l+1}]), s \in C^{r_i}(\eta_i), i = 1, \dots, l\}.\end{aligned} \tag{2.6}$$

As stated before, this space $\mathbb{S}_p^{\mathbf{r}}(\Delta)$ and the B-spline space $\mathbb{S}_{p,\boldsymbol{\xi}}$ are equivalent, given correctly chosen knot-sequence $\boldsymbol{\xi}$, element bounds Δ and boundary regularity $\mathbf{r}$:

Theorem 2.4. *(Lyche et al., 2018, Thm 6)*
The piecewise polynomial space $\mathbb{S}_p^{\mathbf{r}}(\Delta)$ is characterized in terms of B-splines by

$$\mathbb{S}_p^{\mathbf{r}}(\Delta) = \mathbb{S}_{p,\boldsymbol{\xi}}$$

Where the knot-sequence $\xi := \{\xi\}_{i=1}^{n+p+1}$ with $n := dim(\mathbb{S}_p^{\mathbf{r}}(\Delta))$ is constructed such that

$$\xi_1 \leq \cdots \leq \xi_{p+1} := \eta_0, \quad \eta_{l+1} =: \xi_{n+1} \leq \cdots \leq \xi_{n+p+1}$$

and

$$\xi_{p+2}, \ldots, \xi_n := \overbrace{\eta_1, \ldots, \eta_1}^{p-r_1}, \ldots, \overbrace{\eta_l, \ldots, \eta_l}^{p-r_l}$$

Next, we will characterize the derivative spline space by the following theorem:

Theorem 2.5. *(Lyche et al., 2018, Thm 7) Given a knot-sequence $\boldsymbol{\xi} := \{\xi_i\}_{i=1}^{n+p+1}$, we have for $0 \leq r \leq p$,*

$$D_+^r \mathbb{S}_{p,\boldsymbol{\xi}} = \mathbb{S}_{p-r,\boldsymbol{\xi}_r}$$

where $\xi_r := \{\xi_i\}_{i=r+1}^{n+p+1-r}$.

In practice, we will take the knot-sequence $\boldsymbol{\xi}$ to be equidistant. So, partitioning the unit interval in N domains, the knots are spaced $1/N$ apart, with the multiplicity of the end knots being $p+1$, to form a $(p+1)$-open knot-sequence.

2.1.3. Knot Insertion

Given a knot-sequence $\boldsymbol{\xi}$ and a real number $\tilde{\xi}$ such that $\xi_1 \leq \tilde{\xi} \leq \xi_n$. The sequence $\tilde{\boldsymbol{\xi}}$ derived from adding the knot ξ is again a knot-sequence for which we can define a spline space $\mathbb{S}_{p,\tilde{\boldsymbol{\xi}}}$. Additionally, each element $s(x) \in \mathbb{S}_{p,\boldsymbol{\xi}}$ can be mapped in to $\mathbb{S}_{p,\tilde{\boldsymbol{\xi}}}$ with the following theorem:

Theorem 2.6. *(Lyche et al., 2018, Thm. 10)*
Let the $(p+1)$-basic knot-sequence $\tilde{\boldsymbol{\xi}} := \{\tilde{\xi}_i\}_{i=1}^{n+p+2}$ be obtained from the $(p+1)$-basic knot-sequence $\boldsymbol{\xi} := \{\xi_i\}_{i=1}^{n+p+1}$ by inserting one knot ξ such that $\xi_m \leq \xi < \xi_{m+1}$. Then,

$$s(x) = \sum_{j=1}^{n} c_j b_{j,p,\boldsymbol{\xi}}(x) = \sum_{i=1}^{n+1} \tilde{c}_i b_{i,p,\tilde{\boldsymbol{\xi}}}(x), \quad x \in [\xi_{p+1}, \xi_{n+1}], \tag{2.7}$$

where

$$\tilde{c}_i = \begin{cases} c_i, & \text{if } i \leq m-p, \\ \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} c_i + \frac{\xi_{i+p} - \xi}{\xi_{i+p} - \xi_i} c_{i-1}, & \text{if } m-p < i \leq m, \\ c_{i-1}, & \text{if } i > m. \end{cases} \tag{2.8}$$

As the knot-sequence $\tilde{\boldsymbol{\xi}}$ is derived from the knot-sequence $\boldsymbol{\xi}$, by adding new knots. It must be that $\boldsymbol{\xi} \subset \tilde{\boldsymbol{\xi}}$, meaning that by Theorem 2.6, we have that $\mathbb{S}_{p,\boldsymbol{\xi}} \subset \mathbb{S}_{p,\tilde{\boldsymbol{\xi}}}$. This process thus creates nested B-spline spaces.

2.1.4. Choosing appropriate knot-sequences

To show the effect the knot-sequence can have, the following problem is solved with the finite element method with various different B-spline spaces.

$$\begin{aligned} D^2 u &= |\sin(2\pi x)|, \quad \forall u \in (0,1) \\ u(0) &= u(1) = 0 \end{aligned} \tag{2.9}$$

The results can be seen in Table 2.1. Note that while keeping the number of degree of freedom n the same for all B-spline spaces, the general trend is that increasing the polynomial degree improves the solution. However, for case d, the spline space of degree 4 is less accurate than case c, with degree 3. This is a result of the underlying problem. In (2.9), the final solution must be C^2 regular at the point $x = 1/2$. However, the B-spline space in case d is C^3 regular. We can improve upon this, by making

case	p	n	ξ	error
a	1	9	$\{0, 0, 1/8, 1/4, 3/8, 1/2, 5/8, 3/4, 7/8, 1, 1\}$	0.0837
b	2	9	$\{0, 0, 0, 1/7, 2/7, 3/7, 4/7, 5/7, 6/7, 1, 1, 1\}$	0.0093
c	3	9	$\{0, 0, 0, 0, 1/6, 1/3, 1/2, 2/3, 5/6, 1, 1, 1, 1\}$	0.0031
d	4	9	$\{0, 0, 0, 0, 0, 1/5, 2/5, 3/5, 4/5, 1, 1, 1, 1, 1\}$	0.0043
e	4	9	$\{0, 0, 0, 0, 0, 1/4, 1/2, 1/2, 3/4, 1, 1, 1, 1, 1\}$	0.0021

Table 2.1: Problem (2.9) has been solved with various different B-spline spaces of various degrees p and different knot-sequences ξ while keeping the number of degrees of freedom n constant. One can see that the error reduces with increasing degree. However, for $p = 4$, the error is worse than $p = 3$. This is due to the fact that solution must be C^2 regular at the point $x = 1/2$. In option e, we have altered the knot-sequence such that the B-splines of the resulting spline space are C^2 regular at the point $x = 1/2$.

sure that the B-spline space is C^2 regular at the point $x = 1/2$ by setting the multiplicity of the knot $\xi = 1/2$ to be $\mu(\xi) = p - 2$. See case e.

To see the differences in regularity between the cases, in Figure 2.2 one can see the second derivative for the cases c,d and e. Here it becomes clear that case e is the best option, as it is able to approximate the forcing term the best away from $x = 1/2$, while also being able to deal with the discontinuous derivative at $x = 1/2$.

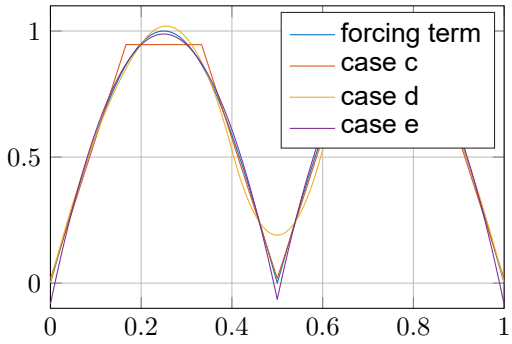


Figure 2.2: The second derivative of the solutions to problem (2.9) for cases c,d and e of Table 2.1. Here it is clear that case e is the best option, as case e is able to approximate the forcing term the best over the whole domain, including the point $x = 1/2$ where case d has issues with the discontinuous derivative.

2.1.5. Multivariate spline spaces

The multivatiate B-splines are constructed as a tensor product of the previously discussed one dimensional B-splines. Thus, we obtain spline functions $B(x, y)$ that are defined as:

$$b_{i,p_x,\xi_x,j,p_y,\xi_y}(x, y) := b_{i,p_x,\xi_x}(x) b_{j,p_y,\xi_y}(y), \tag{2.10}$$

and similarly for more variables. Note that p_x, p_y are the degree of the splines in the x direction and the y direction. Additionally we denote n_x, n_y as the amount of splines in each direction. As there are multiple knot-sequences, a domain rectangle can be constructed as follows:

$$R := [\xi_{x,p_x+1}, \xi_{x,n_x+1}] \times [\xi_{y,p_y+1}, \xi_{y,n_y+1}]$$

See Figure 2.3 for an example of a multivariate B-spline. It is clear from the tensor product nature of the multivariate B-spline spaces that the they inherit all the features of the univariate B-spline spaces. Additionally, when indexing the elements by e, we can define a mesh regularity constant μ_e, which is the ratio of the elements smallest edge, and its diameter h_e. In the case of B-splines with equidistant knot spacing, this constant is the same for every element, as will also be the case for (T)HB-spline spaces.

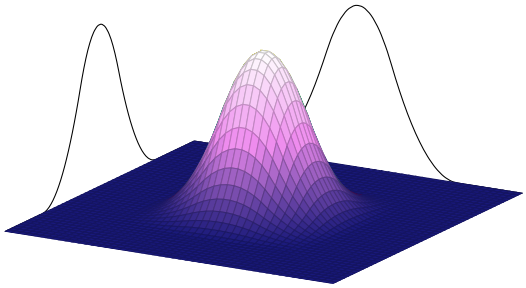


Figure 2.3: A multivariate B-spline constructed as the tensor product of two univariate B-splines.

2.2. Hierarchical B-Spline Spaces

A big disadvantage of the multivariate B-spline spaces, is that the accuracy has to be determined in a global sense (see for example Figure 2.4). Here, we desire higher accuracy in the lower left corner, and additional knots have been inserted to accomplish this. However, the top left and bottom right corners also have increased knots, and thus an undesired increase of accuracy. This increased accuracy comes at the cost more degrees of freedom of the final spline space, and will thus have a negative impact on computation time.

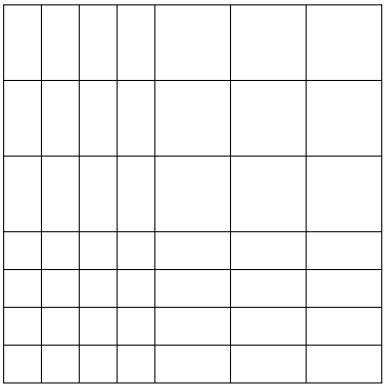


Figure 2.4: Given a domain rectangle R and the knot-sequences $\boldsymbol{\xi}_x = \boldsymbol{\xi}_y = \{0, 1/2, 1, 3/2, 2, 3, 4, 5\}$. This choice of knot-sequences allows the resulting multivariate B splines complex allows for higher accuracy in the lower left quadrant of the domain rectangle. However the top left and bottom quadrants are also higher accuracy as a result. This might not be desired since this will require additional resources.

To remedy the undesired increase in inaccuracy / degrees of freedom, a hierarchical B-spline space will be introduced. These splines will initially be introduced in the univariate case, but can be extended to the multivariate case very naturally.

2.2.1. Univariate Hierarchical B-Spline Spaces

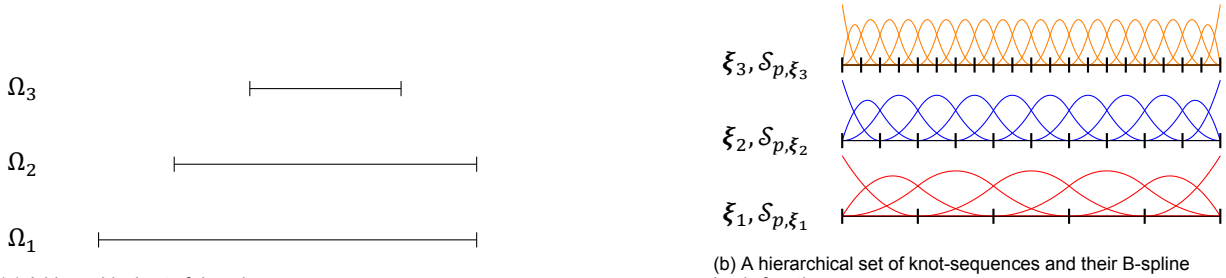
In order to locally refine the mesh (and thus locally increase the accuracy), we will define a sequence of domains, Ω_l for $l = 1, \dots, L$ and $\Omega_{l+1} \subset \Omega_l$. See Figure 2.5a for such a collection of domains. These refinement domains, must be chosen such that they are a collection of present mesh elements e. For example, Ω_2 in Figure 2.5a is the second, third, fourth and fifth element of the B-spline space $\mathbb{S}_{p,\boldsymbol{\xi}_1}$.

Additionally, on each domain, we will define a $(p+1)$-open knot-sequence $\boldsymbol{\xi}_l$ that refines the previous level knot-sequence $\boldsymbol{\xi}_{l-1}$ by splitting every element in two equal parts. This results in a nested knot-sequence:

$$\boldsymbol{\xi}_1 \subset \boldsymbol{\xi}_2 \subset \cdots \subset \boldsymbol{\xi}_L$$

See Figure 2.5b for an example of nested knot-sequences. Additionally, as a result from Theorem 2.6, the associated B-spline spaces are nested:

$$\mathbb{S}_{p,\boldsymbol{\xi}_1} \subset \mathbb{S}_{p,\boldsymbol{\xi}_2} \subset \cdots \subset \mathbb{S}_{p,\boldsymbol{\xi}_L}$$



(a) A hierarchical set of domains.

(b) A hierarchical set of knot-sequences and their B-spline basis functions.

Figure 2.5: A set set of hierarchical domains of three levels and the knot-sequences defined on each level. Note that the knot-sequences are nested, and the next level of knot-sequences is defined by splitting all elements of the previous level in two.

The way to construct the hierarchical B-spline space, is to pick the right splines from all these B-spline spaces of different levels, such that on domain Ω_l, the level l B-spline whose support is entirely contained in Ω_l, are present. For this, denote the collection of sub domains as $\mathbf{\Omega} := \{\Omega_1, \Omega_2, \dots, \Omega_L\}$, which we will refer to as the **domain hierarchy**. On this domain hierarchy, a finite element space can defined, see Definition 2.7.

Definition 2.7. Given a domain hierarchy $\mathbf{\Omega}$, the corresponding basis set of HB-splines is denoted by $\mathcal{H}_\Omega$ and defined recursively as follows:

(i) $\mathcal{H}_1 := \left\{b_{j,1}(x) \in \mathcal{B}_{p,\xi_1} : \operatorname{supp}\left(b_{j,1}(x)\right) \neq \emptyset\right\}$

(ii) for $l = 2, \dots, L$:

$$\mathcal{H}_l := \mathcal{H}_L^C \cup \mathcal{H}_l^F.$$

where

$$\mathcal{H}_l^C := \left\{b_{j,k}(x) \in \mathcal{H}_{l-1} : \operatorname{supp}\left(b_{j,l}(x)\right) \nsubseteq \Omega_l\right\},$$
$$\mathcal{H}_l^F := \left\{b_{j,l}(x) \in \mathcal{B}_{p,\xi_l} : \operatorname{supp}\left(b_{j,l}(x)\right) \subseteq \Omega_l\right\}.$$

(iii) $\mathcal{H}_{\mathbf{\Omega}} := \mathcal{H}_L$.

Additionally, the hierarchical B-spline space is defined as $\mathbb{H}_{\mathbf{\Omega}} := \operatorname{span}\left(\mathcal{H}_{\mathbf{\Omega}}\right)$

To summarize this definition, for the initial level we take all the splines of $\mathcal{B}_1$ that have non vanishing support on Ω_1. Next, for every recursive level $l > 1$, we take all the splines from the previous level $\mathcal{H}_{l-1}$ whose support is not entirely contained in Ω_l. To this set, add all the splines of $\mathcal{B}_l$ whose support is entirely contained in Ω_l. In Figure 2.6, the resulting HB-spline space can be seen from the example Domain Hierarchy from Figure 2.5.

2.2.2. Multivariate Hierarchical B-Spline Spaces

For the Multivariate case, notice that Definition 2.9 can be extended to any number of dimensions, by replacing all univariate B-spline spaces with multivariate B-spline spaces. By doing so, the domain hierarchy is also extended to the multidimensional case, and we obtain a multidimensional mesh. See Figure 2.7 for an example. In this example, the lower left corner is refined.

2.2.3. Properties

HB-spline spaces have the following properties:

Property 2.8. Given a Hierarchical B-spline space $\mathbb{H}_{\mathbf{\Omega}}$, the function space has the following properties:

- **Local support**. An HB-spline is locally supported.
- **Non-negativity**. All HB-splines are non-negative everywhere and strictly positive on the open local support.
- **Piecewise Structure**. HB-splines have a polynomial piecewise structure.

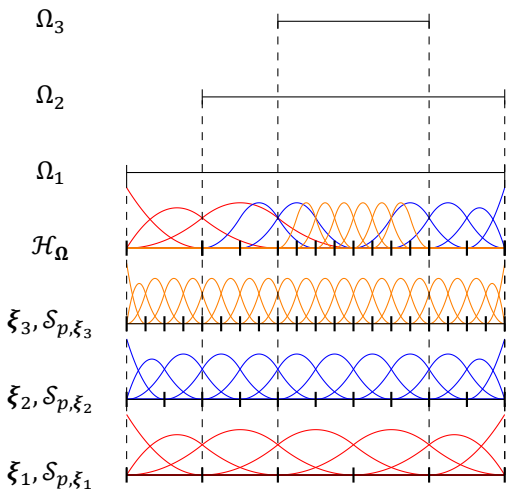


Figure 2.6: The resulting hierarchical B-spline space $\mathbb{H}_{\mathbf{\Omega}}$ resulting from the domain hierarchy from Figure 2.5.

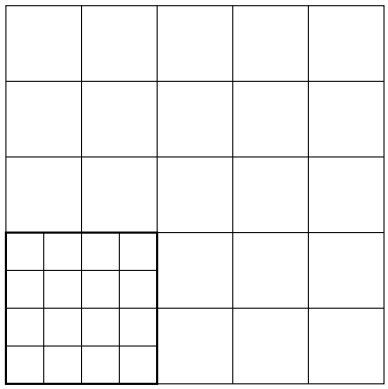


Figure 2.7: Local refinement possible by using hierarchical domains.

- **Global Linear Independence**. Over the entirety of the domain Ω_1, the collection of all HB-splines are linearly independent.

Of this list of properties, only the global linear independence is non-trivial. See (Lyche et al., 2018) for a proof. Comparing this list to the list of properties of B-splines, two things stand out. First, the HB-spline space is only globally linearly independent than locally on every element. Secondly, the HB-spline space no longer has the partition of unity property. To obtain a Hierarchical B-spline space that does have the partition of unity property, the Hierarchical B-spline spaces are altered to create a Truncated Hierarchical B-spline space.

2.3. Truncated Hierarchical B-splines spaces

Truncated Hierarchical B-spline spaces are an altered Hierarchical B-spline space, such that the space has the partition of unity property. For this space, we will first introduce the truncation operator defined over level l:

$$\mathrm{trunc}_{l,\mathbf{\Omega}} : \mathbb{S}_{p_l,\xi_l} \to \mathbb{S}_{p_l,\xi_l}, \tag{2.11}$$

$$\mathrm{trunc}_{l,\mathbf{\Omega}}\left(\sum_{j=1}^{n_l} c_{j,l} b_{j,l}(x)\right) := \sum_{j:\mathrm{supp}_{\mathbf{\Omega}}(b_{j,l}(x)) \not\subseteq \Omega_l} c_{j,l} b_{j,l}(x). \tag{2.12}$$

This truncation operator, when refining to level l, disregards all information belonging the basis functions of $\mathbb{S}_{p_l,\xi_l}$ which are elements of the intermediary HB-spline basis. Additionally, as for the construction of the B-spline spaces, we have that:

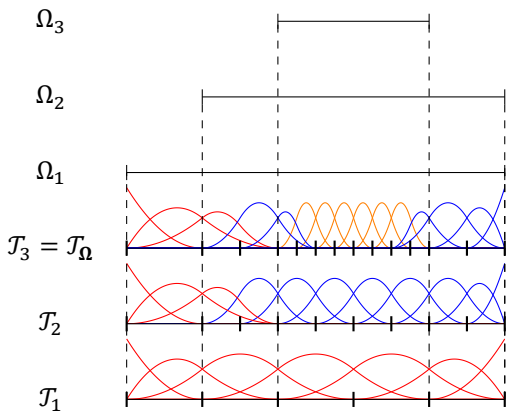


Figure 2.8: The resulting Truncated Hierarchical B-spline space $\mathbb{T}_\Omega$ resulting from the domain hierarchy from Figure 2.5.

$$\mathbb{S}_{p,\xi_1} \subset \mathbb{S}_{p,\xi_2} \subset \cdots \subset \mathbb{S}_{p,\xi_L}$$

The truncation operator can be extended to all B-spline spaces of level l and lower. This truncation operator can be used to construct Hierarchical B-spline spaces that have the partition of unity property. To see why, when a level $l-1$ B-spline is projected into the level l B-splines, we obtain the following linear combination:

$$b_{i,l-1}(x) = \sum_{j=1}^{n_l} c_{i,j,l} b_{j,l}(x), \quad x \in \Omega_1. \tag{2.13}$$

It can happen that the lower level B-spline $b_{i,l-1}(x)$, is dependent on a level l B-spline $b_{j,l}(x)$, that is an element of the HB-spline basis. Then, the unit sum of all HB-spline basis functions up to level l, must be equal or greater then $1 + c_{i,j,l} b_{j,l}(x) > 1$. Note that for the level l B-spline space, these $c_{i,j,l}$ can be calculated. Clearly, this can happen for many level l B-splines. However, by truncating all the lower level HB-splines, these additional components, $c_{i,j,l}$, vanish. Doing this in for every intermediary level in the HB-spline space construction, we obtain the Truncated Hierarchical B-spline space:

Definition 2.9. Given a domain hierarchy $\mathbf{\Omega}$ the corresponding set of THB-splines basis functions is denoted by $\mathcal{T}_\Omega$ and defined recursively as:

(i) $\mathcal{T}_1 := \{b_{j,1}(x) \in \mathcal{B}_1 : \operatorname{supp}_\Omega(b_{j,1}(x)) \neq \emptyset\}$

(ii) for $l = 2, \ldots, L$:

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

where

$$\mathcal{T}_l^C := \{\operatorname{trunc}_{l,\Omega}(b_{j,k}(x)) : b_{j,k}(x) \in \mathcal{T}_{l-1}, \operatorname{supp}_\Omega\left(b_{j,k}(x)\right) \not\subseteq \Omega_l\}$$
$$\mathcal{T}_l^F := \{b_{j,l}(x) \in \mathcal{B}_l : \operatorname{supp}_\Omega\left(b_{j,l}(x)\right) \subseteq \Omega_l\}$$

(iii) $\mathcal{T}_\Omega := \mathcal{T}_L$

Then, the space of Truncated Hierarchical B-splines is given by $\mathbb{T}_\Omega := \operatorname{span}\{\mathcal{T}_\Omega\}$.

Note that from the definition of the truncated hierarchical B splines, the number of basis functions is the same as the number of basis functions for the hierarchical B splines. See Figure 2.8, for an example of a Truncated Hierarchical B-spline space.

2.3.1. Properties

THB-spline spaces have the following properties.

Property 2.10. Given a Hierarchical B-spline space $\mathbb{H}_\Omega$, the function space has the following properties:

- **Local support**. A THB-spline is locally supported.
- **Non-negativity**. All THB-splines are non negative everywhere and strictly positive on the open local support.
- **Piecewise Structure**. THB-splines have a polynomial piecewise structure.
- **Partition of Unity**. All the basis functions of $\mathcal{T}_\Omega$ sum to the unit function over Ω_1.
- **Global Linear Independence**. Over the entirety of the domain Ω_1, the collection of all THB-splines are linearly independent.

2.4. Bernstein Polynomials

Additionally, a particular set of B-splines will be introduced, namely the Bernstein polynomials. These are a special subset of B-splines for which the inverse Gramian matrices are known. The Bernstein polynomials are defined as:

$$\mathfrak{b}_{i,p}(x) := \binom{p}{i-1}(x)^{i-1}(1-x)^{p+1-i}, \quad i = \{1, 2, \dots, p+1\}, x \in [0, 1]. \tag{2.14}$$

See Figure 2.9 for an example of degree 4 Bernstein polynomials.

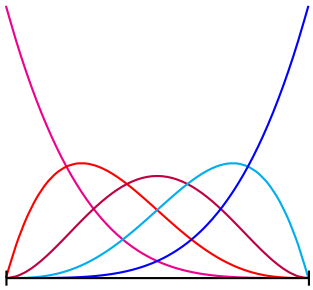


Figure 2.9: Degree 4 Bernstein polynomials over a unit interval.

As stated before, the Bernstein polynomials are a special set of B-splines. They are defined for the knot-sequence $\boldsymbol{\xi} := \{0, \dots, 0, 1, \dots, 1\}$, where both knots have multiplicity $p+1$. We will proof this with induction. This clearly holds for $p = 0$. For the induction step, assume that $\mathfrak{b}_{i,p-1}(x) = b_{i,p-1,\boldsymbol{\xi}_{p-1}}(x)$. Then, we have:

$$b_{j,p,\boldsymbol{\xi}_p}(x) = \frac{x-\xi_j}{\xi_{j+p}-\xi_j} b_{j,p-1,\boldsymbol{\xi}_p}(x) + \frac{\xi_{j+p+1}-x}{\xi_{j+p+1}-\xi_{j+1}} b_{j+1,p-1,\boldsymbol{\xi}_p}(x) \tag{2.15}$$

$$= x b_{j-1,p-1,\boldsymbol{\xi}_{p-1}}(x) + (1-x) b_{j,p-1,\boldsymbol{\xi}_{p-1}}(x) \tag{2.16}$$

$$= x\binom{p-1}{j-2}x^{j-2}(1-x)^{p-j+1} + (1-x)\binom{p-1}{j-1}x^{j-1}(1-x)^{p-j} \tag{2.17}$$

$$= \binom{p-1}{j-2}x^{j-1}(1-x)^{p-j+1} + \binom{p-1}{j-1}x^{j-1}(1-x)^{p-j+1} \tag{2.18}$$

$$= \binom{p}{j-1}x^{j-1}(1-x)^{p-j+1} \tag{2.19}$$

$$= \mathfrak{b}_{j,p}(x) \tag{2.20}$$

The B-splines are renumbered, since the degree $p-1$ knot-sequence, has two knots less. Additionally, the binomial recursion equation has been used. Lastly, for certain j, the fractions do not exist. However, exactly for those fractions, the binomial coefficients do not exist as well. These terms are interpreted to be zero. The relation will thus still hold.

For a product of Bernstein polynomials over the unit interval, we have the following result:

$$\int_0^1 \mathfrak{b}_{i,p}(x)\mathfrak{b}_{j,p}(x)dx = \frac{1}{2p+1}\binom{2p}{i+j-4}^{-1}\binom{p}{i-2}\binom{p}{j-2} \tag{2.21}$$

From this result, the Gramian matrix can be constructed:

$$[G]_{i,j} = \int_0^1 \mathfrak{b}_{i,p}(x)\mathfrak{b}_{j,p}(x)dx \tag{2.22}$$

And the entires of the inverse of the Gramian matrix are given by:

$$\begin{aligned}\left[G^{-1}\right]_{i,j} =&(-1)^{i+j-2}\left[\binom{p}{i-2}\binom{p}{j-2}\right]^{-1}\times \\ &\sum_{k=1}^{\min(i-1,j-1)}(2k-1)\binom{p-k-1}{p-i-2}\binom{p-k-1}{p-j-2}\binom{p+k}{p-i-2}\binom{p+k}{p-j-2}\end{aligned} \tag{2.23}$$

See (Jüttler, 1998), for proofs of these statements.

2.5. Supplemental theory

We will finish this chapter with some external results for splines. Starting with the following lemma from (Bazilevs et al., 2006), which is an approximation result for a given B-spline space $\mathbb{S}$ and an element e, on the support extension $\tilde{e}$. The support extension is defined as:

Definition 2.11. The **support extension** $\tilde{e}$ of an element e, is the union of the supports of all splines of some spline space $\mathbb{S}$ with non-empty support on e:

$$\tilde{e} := \bigcup_{\substack{b(x)\in\mathbb{P} \\ \Omega^e\subset\mathrm{supp}(b(x))}} \mathrm{supp}(b_j(x)). \tag{2.24}$$

Then, we can approximate $v \in H^l(\Omega^{\tilde{e}})$ by the following lemma:

Lemma 2.7 (Lemma 3.1, (Bazilevs et al., 2006))**.** *Let k and l be integer indices with $0 \le k \le l \le p+1$. Given e, and support extension $\tilde{e}$, $v \in H^l\left(\Omega^{\tilde{e}}\right)$, there exists an $s \in \mathbb{S}$ such that:*

$$|v-s|_{H^k(\Omega^{\tilde{e}})} \le C h_e^{l-k}|v|_{H^l(\Omega^{\tilde{e}})}. \tag{2.25}$$

Here h_e, is the diameter of element element e and C is a constant independent of h_e, but possibly dependent on l, k and μ, the mesh shape constant.

Additionally, on an element e, the local L^2 projection onto the Bernstein polynomials, with the help of the Gramian matrices introduced in Section 2.4, can be bound by the following lemma from (Thomas et al., 2015).

Lemma 2.8 (Lemma A.5, (Thomas et al., 2015))**.** *For each element e, the local Bernstein coefficient vector $\boldsymbol{\beta}^e$ associated with the local L^2-projection of a function $f \in L^2(\Omega^e)$ onto the space of polynomials of degree p satisfies the inequality:*

$$\|\boldsymbol{\beta}^e\|_\infty \le \frac{C_p}{|\Omega^e|^{1/2}}\|f\|_{L^2(\Omega^e)}, \tag{2.26}$$

where C_p is a constant only dependent upon the polynomial degree p.

3

Finite Element Exterior Calculus

Finite Element Exterior Calculus (FEEC) concerns itself with choosing appropriate finite dimensional spaces for the so called abstract Hodge Laplace problem. This is a generalization of the Laplacian problem, which in 1D is given by the problem of finding $u(x) \in H_0^2([0,1])$:

$$\begin{aligned} -\frac{d^2}{dx^2}u(x) &= f(x), \quad x \in (0,1), \\ u(0) = u(1) &= 0. \end{aligned} \tag{3.1}$$

For a given forcing function $f(x) \in L^2([0,1])$. One way to solve this problem, is by writing the problem in the mixed weak form, where we look for the function pair $(u, \sigma) \in L^2([0,1]) \times H^1([0,1])$:

$$\begin{aligned} \int_0^1 \tau(x)\sigma(x)dx - \int_0^1 \frac{d}{dx}\tau(x)u(x)dx &= 0, && \forall \tau(x) \in H^1([0,1]), \\ \int_0^1 v(x)\frac{d}{dx}\sigma(x)dx &= \int_0^1 v(x)f(x)dx, && \forall v(x) \in L^2([0,1]). \end{aligned} \tag{3.2}$$

In this formulation, the Dirichlet boundary conditions are implicitly enforced in the formulation. When we solve this problem with Galerkin's method, a choice has to be made for which finite dimensional spaces to use. Depending on this choice, the method might be stable, or unstable. For example, for a given partition Δ of the unit domain, we can define the local piecewise polynomial spaces:

$$\mathbb{F}_p^{-1} := \left\{ v \in L^2([0,1]) : v_{|_I}(x) \in \mathbb{P}_p(I), \forall I \in \Delta \right\} \tag{3.3}$$

Then, in Figure 3.1, two choices are shown. The left graph shows the solution to u and the right graph the solution for σ. In both graphs, three lines are plotted, a blue line which represents the analytical solution, and two solutions, shown with green and red lines. Both represent a different choice of finite dimensional spaces. In the case of the green solution, u is approximated in the space F_0^{-1}, and σ in the space F_1^{-1}. This choice approximates the analytical solution (the blue line is almost perfectly covered by the green line in the right graph). The red solution, where the space F_0^{-1} is used for u, and F_2^{-1} for σ, performs a lot worse, despite, theoretically, allowing for better approximations. Finite Element Exterior Calculus concerns it self with this issue.

We will first introduce the Hilbert Complex. This is a collection of function spaces, coupled by the exterior derivative (in practice, these are regular derivatives, or vector derivatives like grad and curl). On this Hilbert Complex, we will introduce the abstract Hodge Laplacian, which is (as we will see) a generalization of the (vector) Laplace problem. Next, the abstract Hodge Laplace problem will be formulated as a mixed weak formulation. Similarly as was done for the example. For the mixed weak formulation, we will see that the abstract Hodge Laplacian, under three requirements on the finite dimensional spaces, are well posed and stable. We will end this Chapter by looking closer at what kind of function spaces the Hilbert Complex is made of, introduce some polynomial Finite element spaces, and see what the current state of THB-splines used in FEEC is.

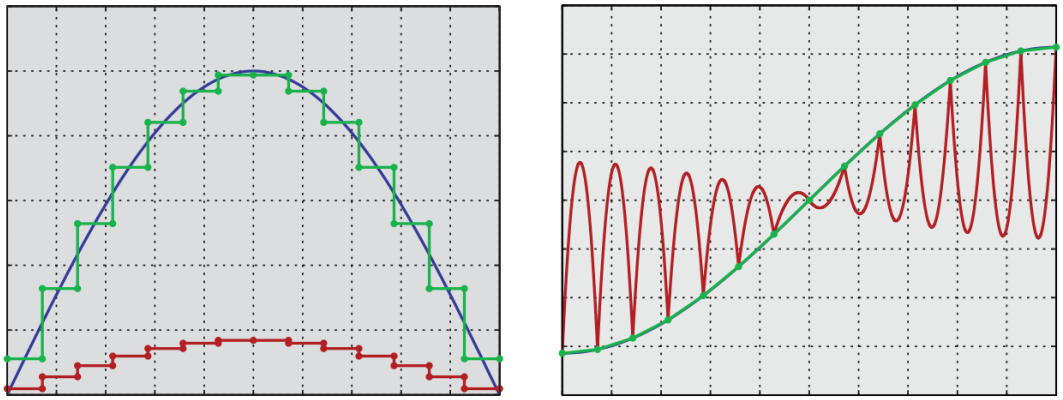


Figure 3.1: The solution to the mixed weak problem in (3.2). The left graph shows the solution of function u and the right graph shows the solution to the σ function. In both graphs, three lines. The blue being the analytical solution. The red and green being solutions for a particular choice of finite element spaces. The green solution belongs to the choice of F_0^{-1} for u, and F_1^{-1} for σ, while the red solution belongs to the choice of F_0^{-1} for u, and F_2^{-1} for σ. Even though, the red solution should have better approximation power, the red solution performs a lot worse. Image taken from (D. N. Arnold et al., 2010).

3.1. Hilbert Complex

We start by introducing a Hilbert complex:

Definition 3.1. A **Hilbert complex** is a sequence of Hilbert spaces W^k and a sequence of closed densely defined linear operators d^k from W^k to W^{k+1} such that $\mathcal{R}(d^k) \subset \mathcal{N}(d^{k+1})$.

We call the linear operator d^k the **exterior derivative** and the exterior derivative has the property that $\mathcal{R}(d^k) \subset \mathcal{N}(d^{k+1})$ implies that $d^{k+1} \circ d^k = 0$. The simplest example of a Hilbert complex can be constructed in 1D as follows:

Example 1. In one dimension, we take the domain to be the unit interval $\Omega = (0, 1)$. Let W^0 be the Hilbert space $L^2(\Omega)$ and $W^1 = L^2(\Omega)$. Let the exterior derivative d^0 be the classical derivative. Then the Hilbert complex is given by:

$$L^2(\Omega) \xrightarrow{\frac{d}{dx}} L^2(\Omega)$$

Note, that in this case, since there is only one exterior derivative (namely d/dx), the required property $d^{k+1} \circ d^k = 0$ is always satisfied.

Note that in the previous example, there is only one exterior derivative. The requirement that $\mathcal{R}(d^k) \subset \mathcal{N}(d^{k+1})$, is thus always satisfied. However, we can construct complexes where this requirement plays a role. For example, the following complex in three dimensions, for the three dimensional unit cube $\Omega = (0, 1)^3$:

$$L^2(\Omega) \xrightarrow{\text{grad}} L^2(\Omega; \mathbb{R}^3) \xrightarrow{\text{curl}} L^2(\Omega; \mathbb{R}^3) \xrightarrow{\text{div}} L^2(\Omega)$$

Due to the vector identities $\text{curl}(\text{grad}(f)) = 0$ and $\text{div}(\text{curl}(\vec{g})) = 0$ and a density argument, we find that this is in fact, a Hilbert complex.

Due to the closed dense property of the exterior derivative, the whole space is not the domain of the exterior derivative. For this, we can additionally define the domain complex, as:

Definition 3.2. Given a Hilbert complex W^k and the exterior derivatives $d^k : W^k \to W^{k+1}$, define the **domain complex** as:

$$V^k := D(d^k). \tag{3.4}$$

Where $D(d^k)$ is the domain of the operator d^k in W^k.

Given a Hilbert and a domain complex, note the similar structure for both:

$$\ldots \xrightarrow{d} W^{k-1} \xrightarrow{d} W^{k} \xrightarrow{d} W^{k+1} \xrightarrow{d} \ldots \tag{3.5}$$

$$\ldots \xrightarrow{d} V^{k-1} \xrightarrow{d} V^{k} \xrightarrow{d} V^{k+1} \xrightarrow{d} \ldots \tag{3.6}$$

Example 2. The three dimensional L^2 de Rham complex. Given the three dimensional Hilbert complex, the domain complex is given by:

$$H^1(\Omega) \xrightarrow{\text{grad}} H(\Omega, \text{curl}) \xrightarrow{\text{curl}} H(\Omega, \text{div}) \xrightarrow{\text{div}} L^2(\Omega) \tag{3.7}$$

Here $H(\Omega, \text{curl})$ are the $L^2(\Omega, \mathbb{R}^3)$ functions such that the curl exists and is an element of $L^2(\Omega, \mathbb{R}^3)$. Similarly, $H(\Omega, \text{div})$ are the $L^2(\Omega, \mathbb{R}^3)$ functions such that the divergence is an element of $L^2(\Omega)$.

3.1.1. The adjoint of the exterior derivative

Since by definition, d^k is a densely defined linear operator from W^k to W^{k+1}, we may define an unbounded linear operator from W^{k+1} to W^k called the adjoint of d^k, which we denote by d^*_{k+1}. To define the adjoint, consider any $w \in W^{k+1}$. Then, since every W^k is a Hilbert space, we can take the inner product to define a linear mapping $D(d^k) \to \mathbb{R}$:

$$v \to \left\langle w, d^k v \right\rangle_{W^{k+1}}, \quad v \in D(d^k) \tag{3.8}$$

Next, we consider the set of $w \in W^{k+1}$ for which the above linear functional (3.8) is bounded in the W^k norm. For example, the elements of $w \in W^{k+1}$ for which there exists a c_w, such that:

$$\left|\left\langle w, d^k v \right\rangle_{W^{k+1}}\right| \leq c_w ||v||_{W^k}, \quad \forall v \in D(d^k)$$

Define this set of $w \in W^{k+1}$ as the domain of d^*_{k+1}, denoted by $D(d^*_{k+1})$. For any of such $w \in D(d^*_{k+1})$, we can extend (3.8) to a bounded linear functional over the whole domain W^k, since $D(d^k)$ is dense in W^k. By now applying the Riesz representation theorem, there must exist a unique element $x \in W^k$ such that:

$$\left\langle w, d^k v \right\rangle_{W^{k+1}} = \langle x, v \rangle_{W^k}, \quad \forall v \in D(d^k)$$

Then we define the adjoint d^*_{k+1} by setting $d^*_{k+1} w = x$. Meaning:

$$\left\langle d^*_{k+1} w, v \right\rangle_{W^k} = \left\langle w, d^k v \right\rangle_{W^{k+1}} \tag{3.9}$$

And the domain $w \in D(d^*_{k+1})$ is exactly those elements of W^{k+1}, for which there exists a unique element of $x \in W^k$. This element x fulfills the above equation for $d^*_{k+1} w = x$.

Example 3. In the 1D case, the adjoint of the exterior derivative d/dx, is again given by d/dx. The domain of the adjoint is $H^1_0([0,1])$, where the subscript 0 denotes zero boundary information. To see that this is a bounded adjoint, take $v \in H^1([0,1])$ and $w \in H^1_0([0,1])$, then:

$$\begin{aligned}
\left|\left\langle w, \frac{dv}{dx} \right\rangle_{L^2([0,1])}\right| &= \left|\int_0^1 w \frac{dv}{dx} dx\right| \\
&= \left|\int_0^1 v \frac{dw}{dx} dx\right| \quad \text{(Integration by parts)} \\
&\leq \|w\|_{H^1([0,1])} \|v\|_{L^2([0,1])}.
\end{aligned}$$

Then, we have that $c_w = \|w\|_{H^1([0,1])}$. Note, the domain of the adjoint cannot be larger than $H^1_0([0,1])$. For instance, if we allow boundary information, by integration by parts, we obtain a term that is unable to be written in the regular L^2 norm. Likewise, if we take $w \in L^2([0,1])$, we can not perform integration by parts, meaning that the r.h.s. is bounded by $\|dv/dx\|_{L^2([0,1])}$, which is not the regular L^2 norm.

3.1.2. The dual complex

The notation of the adjoint suggests that we can also create a Hilbert complex, where the adjoint operators map in the reversed direction. This is indeed the result of the following two theorems. The first shows that the adjoints are closed densly defined operators, while the second relates the range/kernel of d^k to the kernel/range of the adjoing d_k^*. See (D. N. Arnold et al., 2010) for proofs.

Theorem 3.1. *If d^k is a closed densely defined unbounded operator from W^k to W^{k+1}, then d_{k+1}^* is a closed densely defined operator from W^{k+1} to W^k.*

Theorem 3.2. *Given d^k a closed densely defined operator W^k to W^{k+1}. Then*

$$\mathcal{R}(d^k)^\perp = \mathcal{N}(d_{k+1}^*), \quad \mathcal{N}(d^k)^\perp = \overline{\mathcal{R}(d_{k+1}^*)}, \quad \mathcal{R}(d_{k+1}^*)^\perp = \mathcal{N}(d^k), \quad \mathcal{N}(d_{k+1}^*)^\perp = \overline{\mathcal{R}(d^k)} \tag{3.10}$$

As a direct result of Theorem 3.2, we find that $\mathcal{R}(d_{k+1}^*) \subset \mathcal{N}(d_k^*)$. This is shown as follows:

$$\mathcal{R}(d_{k+1}^*) \subset \overline{\mathcal{R}(d_{k+1}^*)} = \mathcal{N}(d^k)^\perp \subset \mathcal{R}(d^{k-1})^\perp = \mathcal{N}(d_k^*)$$

Combining these results, we find that we can define the dual complex of a Hilbert complex denoted by (W, d^*):

$$\ldots \xleftarrow[d_{k-1}^*]{} W^{k-1} \xleftarrow[d_k^*]{} W^k \xleftarrow[d_{k+1}^*]{} W^{k+1} \xleftarrow[d_{k+2}^*]{} \ldots \tag{3.11}$$

And the dual domain complex:

$$\ldots \xleftarrow[d_{k-1}^*]{} V_{k-1}^* \xleftarrow[d_k^*]{} V_k^* \xleftarrow[d_{k+1}^*]{} V_{k+1}^* \xleftarrow[d_{k+2}^*]{} \ldots \tag{3.12}$$

where $V_k^* = D(d_k^*)$.

Example 4. Given the domain complex for the three dimensional L^2 de Rham complex, the dual complex is given by:

$$L^2(\Omega) \xleftarrow[-\operatorname{div}]{} \mathring{H}(\Omega, \operatorname{div}) \xleftarrow[\operatorname{curl}]{} \mathring{H}(\Omega, \operatorname{curl}) \xleftarrow[\text{-grad}]{} \mathring{H}^1(\Omega) \tag{3.13}$$

The small circles, denote reduced boundary information. For $\mathring{H}^1(\Omega)$, this means that the functions have zero boundary information. For $\mathring{H}(\Omega, \operatorname{curl})$, the functions have zero in tangential boundary information. Lastly, for $\mathring{H}(\Omega, \operatorname{div})$, the functions have zero normal boundary information. To see that the divergence is the adjoint of the gradient, we can integrate $\operatorname{div}(f\vec{g})$ with $f \in H^1(\Omega)$ and $\vec{g} \in H(\Omega, \operatorname{div})$. Then by stokes, we find:

$$\int_\Omega \operatorname{grad}(f) \cdot \vec{g} dV + \int_\Omega f \operatorname{div}(\vec{g}) dV = \int_{\partial\Omega} \hat{n} \cdot (f\vec{g})\, dS$$

In case that $\vec{g}$ vanishes at the boundary, the right hand side disappears and it becomes obvious that the divergence is the adjoint of the gradient. For a more extensive argument, on why these are the adjoints, see (D. N. Arnold, 2018, section 3.4).

3.2. The (abstract) Hodge Laplacian problem

Now, we are able to define the abstract Hodge Laplacian, which, as we will see is a generalization of the Laplace operator. Given a closed Hilbert complex (W, d), we can define an operator $L : W \to W$ called the (abstract) Hodge Laplacian as follows:

$$L^k u = d_{k+1}^* d^k u + d^{k-1} d_k^* u, \tag{3.14}$$

Where the (abstract) Hodge Laplacian is defined on the space:

$$D(L^k) := \left\{ u \in V^k \cap V_k^* : d_k u \in V_{k+1}^*, d_k^* u \in V^{k-1} \right\}, \tag{3.15}$$

A major subject within FEEC are the numerical solutions to the Hodge Laplace problem $L^k u = f$ for a given $f \in W^k$.

Example 5. In the case of $k = 0$ and the L^2 de Rham complex. (3.14) is equivalent to the ordinary scalar Laplace problem:

$$L^0 u = -\Delta u = f, \quad f \in L^2(\Omega) \tag{3.16}$$

For the case of $k = 1$, we have:

$$L^1 \vec{u} = d_2^* d^1 \vec{u} + d^0 d_1^* \vec{u} = \operatorname{curl}\operatorname{curl} \vec{u} - \operatorname{grad}\operatorname{div} \vec{u} = \vec{f}, \quad \vec{f} \in L^2\left(\Omega; \mathbb{R}^3\right). \tag{3.17}$$

This example also shows us why studying the numerical solution to the Hodge Laplace equation is relevant. Many pde's have components based on the abstract Hodge Laplace operator. For example stokes flow and the Maxwell equations.

The abstract Hodge Laplace problem can be written in three different formulations:

- The strong formulation
- The primal weak form
- The mixed weak form

All three formulations are equivalent. However, for designing numerical methods, the mixed weak form is the preferred choice. For the mixed weak form, one can proof consistency and stability for the Galerkin method. The mixed weak form is given as the following problem:

Given $f \in W^k$, find $\sigma \in V^{k-1}, u \in V^k$ and $p \in \mathfrak{H}^k$ such that:

$$\begin{aligned} \langle \sigma, \tau \rangle - \left\langle u, d^k \tau \right\rangle &= 0, & \tau &\in V^{k-1}, \\ \left\langle d^k \sigma, v \right\rangle + \left\langle d^k u, d^k v \right\rangle + \langle p, v \rangle &= \langle f, v \rangle, & v &\in V^k, \\ \langle u, q \rangle &= 0, & q &\in \mathfrak{H}^k. \end{aligned} \tag{3.18}$$

Here the space $\mathfrak{H}^k$ is defined as $\mathfrak{H}^k := \mathcal{N}(d^k) \cup \mathcal{R}(d^{k-1})^\perp$. In the case of a closed Hilbert complex (as we have assumed) this space is in fact isomorphic to the cohomology space $\mathcal{H}^k$ see (D. N. Arnold et al., 2010) for more details. Note that in this formulation, only the exterior derivative d^k appears. The d_k^* terms are treated weakly in this formulation.

3.3. Approximating Hilbert Complexes

The next step in creating a numerical method, is to choose finite element sub-spaces for the spaces V^k and $\mathfrak{H}^k$. However, note that the space $\mathfrak{H}^k$ is by definition dependent on the spaces V^k and the exterior derivative:

$$\mathfrak{H}^k = \mathcal{N}(d^k) \cup \mathcal{R}(d^{k-1})^\perp = \{u \in V^k : d^k v = 0, v \perp du, \forall u \in V^{k-1}\}$$

Thus, when we approximate $V_h^k \subset V^k$, we construct the finite element space for $\mathfrak{H}^k$ by:

$$\mathfrak{H}_h^k = \{u \in V_h^k : d^k v = 0, v \perp du, \forall u \in V_h^{k-1}\}$$

However, for this choice, it might happen that $\mathfrak{H}_h^k \not\subset \mathfrak{H}^k$. If this is the case, the discretization introduced is not a standard Galerkin discretization, but a generalized Galerkin method. For a generalized Galerkin method, the exact solution u to (3.18) does not satisfy the general Galerkin discretization. However, often it is the case that $\mathfrak{H}_h^k = \mathfrak{H}^k = 0$ (as we will assume from now on), meaning that we can use the standard Galerkin discretization:

Given a $f \in W^k$, find $\sigma_h \in V_h^{k-1}, u_h \in V_h^k$ and $p_h \in \mathfrak{H}_h^k$ such that:

$$\begin{aligned} \langle \sigma_h, \tau \rangle - \left\langle u_h, d^k \tau \right\rangle &= 0, & \tau &\in V_h^{k-1}, \\ \left\langle d^k \sigma_h, v \right\rangle + \left\langle d^k u_h, d^k v \right\rangle + \langle p_h, v \rangle &= \langle f, v \rangle, & v &\in V_h^k, \\ \langle u_h, q \rangle &= 0, & q &\in \mathfrak{H}_h^k. \end{aligned} \tag{3.19}$$

This Galerkin discretization is consistent and stable given the following three properties:

3.3.1. Approximation property

The first property, the approximation property, is straightforward. We require that in order for the solutions σ_h, u_h of (3.19) to approximate the exact solutions σ, u of (3.18), the finite dimensional sub spaces V_h^k must approximate the full spaces V^k. For example, we might require that for any k:

$$\lim_{h\to 0} \inf_{v\in V_h^k} ||w - v||_{V^k} = 0, \quad w \in V^k. \tag{3.20}$$

3.3.2. Subcomplex property

For the second property, we require that $dV_h^{k-1} \subset V_h^k$. In this case, we find that the finite dimensional spaces, form a Hilbert complex:

$$\ldots \xrightarrow{d^{k-2}} V_h^{k-1} \xrightarrow{d^{k-1}} V_h^k \xrightarrow{d^k} V_h^{k+1} \xrightarrow{d^{k+1}} \ldots \tag{3.21}$$

3.3.3. Bounded cochain projections

Requiring the same structure for the finite dimensional complex and the original Hilbert complex, is a very natural property. However, the two complexes are not “linked” so to say. For this, we require a bounded cochain projection property. Here, we require the existence of projectors between the complex (3.6) to the finite dimensional complex (3.21). Denote these projectors as Π_h^k. The full complex and the sub-complex can now be combined in to the following diagram:

$$\begin{array}{ccccccccc} \ldots & \xrightarrow{d^{k-2}} & V^{k-1} & \xrightarrow{d^{k-1}} & V^k & \xrightarrow{d^k} & V^{k+1} & \xrightarrow{d^{k+1}} & \ldots \\ & & \downarrow{\scriptstyle \Pi_h^{k-1}} & & \downarrow{\scriptstyle \Pi_h^k} & & \downarrow{\scriptstyle \Pi_h^{k+1}} & & \\ \ldots & \xrightarrow{d^{k-2}} & V_h^{k-1} & \xrightarrow{d^{k-1}} & V_h^k & \xrightarrow{d^k} & V_h^{k+1} & \xrightarrow{d^{k+1}} & \ldots \end{array} \tag{3.22}$$

Additionally, we require that the projectors commute with the exterior derivatives:

$$d^{k-1}\Pi_h^{k-1} v = \Pi_h^k d^{k-1} v, \quad \forall v \in V^{k-1}. \tag{3.23}$$

The commuting property means that in the diagram in (3.22), it does not matter in which order one moves within the diagram. For example, going from V^{k-1} to V_h^k, both the top path and bottom path return the same result.

3.4. Differential forms

Uptill now, the function spaces V^k have been assumed to be abstract spaces, or given examples in 1D and 3D of scalar/vector spaces. The vector approach, can be used to give useful examples in 2D. However, for generating an n-dimensional approach, we will have to introduce differential k-forms.

3.4.1. Alternating multilinear forms

Given a finite dimensional vector space U and a non negative number k, we consider the vector space $\text{Alt}^k U$ consisting of all k-linear maps:

$$\omega : \overbrace{U \times \cdots \times U}^{k \text{ times}} \to \mathbb{R}.$$

And that changes sign, when interchanging two variables:

$$\omega(u_1, \ldots, u_i, \ldots, u_j, \ldots, u_k) = -\omega(u_1, \ldots, u_j, \ldots, u_i, \ldots, u_k), \quad 1 \le i < j \le n, \quad u_1, \ldots, u_n \in U.$$

Example 6. The $\text{Alt}^1 U$ space is in fact the dual space of U. Any element $\omega \in \text{Alt}^1 U$, applied to an element $u \in U$ must be a real number $\omega(u) \in \mathbb{R}$, and ω must be linear functional.

Definition 3.3. The exterior product of $\omega \in \text{Alt}^j U$ and $\mu \in \text{Alt}^k U$ is given by:

$$(\omega \wedge \mu)(u_1, \ldots, u_{j+k}) = \sum_{\sigma} \text{sign}(\sigma)\omega(u_{\sigma_1}, \ldots, u_{\sigma_j})\mu(u_{\sigma_{k+1}}, \ldots, u_{\sigma_{j+k}}), \tag{3.24}$$

Where the sum is over all $\sigma = (\sigma_1, \ldots, \sigma_{j+k})$ for which $\sigma_1 < \cdots < \sigma_j$ and $\sigma_{j+1} < \cdots < \sigma_{j+k}$.

The exterior product satisfies the anticommutativity law:

$$\omega \wedge \mu = (-1)^{jk} \mu \wedge \omega, \quad \omega \in \text{Alt}^j U, \mu \in \text{Alt}^k U.$$

Basis for Alternating multilinear forms
Let $\{u_i\}$ be a basis for U and let $\{u^i\}$ be the corresponding dual basis. Then given $1 \leq \sigma_1 < \cdots < \sigma_k \leq n$, the element of $\text{Alt}^k U$ which takes the k-tuple $(u_{\sigma_1}, \ldots, u_{\sigma_k})$ to 1 and all other k-tuples with increasing indices to 0 is:

$$u^{\sigma_1} \wedge \cdots \wedge u^{\sigma_k}. \tag{3.25}$$

The set of all these elements, form a basis for $\text{Alt}^k U$. Any element $\omega \in \text{Alt}^k U$, can thus be written as:

$$\omega = \sum_{\sigma} a_\sigma u^{\sigma_1} \wedge \cdots \wedge u^{\sigma_k}, \tag{3.26}$$

with $a_\sigma \in \mathbb{R}$. In the case $U = \mathbb{R}^n$, the obvious choice for the basis is the canonical basis $e_1, \ldots, e_n \in \mathbb{R}^n$. As notation for the dual basis, instead of the previous introduced notation e^i, we will use the notation $dx^1, \ldots, dx^n$. From the definition of the canonical basis, it becomes clear that dx^i assigns to each vector $u \in \mathbb{R}^n$ its ith component. In this case, (3.26) becomes:

$$\omega = \sum_{\sigma} a_\sigma dx^{\sigma_1} \wedge \cdots \wedge dx^{\sigma_k}.$$

3.4.2. Differential forms

Given a manifold Ω of dimension n. At each point $x \in \Omega$, we can define the tangent space $T_x\Omega$. The tangent space is also a space of dimension n, and the collection of pairs (x, v) with $x \in \Omega$ and $v \in T_x\Omega$ defines the tangent bundle, a manifold of dimension $2n$. Coming back to the exterior algebra introduced in the previous section, define the kth exterior power of the cotangent bundle as the pairs (x, μ) with $\mu \in \text{Alt}^k T_x\Omega$. Then, we define the **differential k-forms** as the functions from Ω to an element of $\text{Alt}^k T_x\Omega$:

$$\omega_x(u_1, \ldots, u_k) \in \mathbb{R}, \quad \forall x \in \Omega, u_1, \ldots, u_k \in T_x\Omega, \tag{3.27}$$

In the case that Ω is a sub domain of $\mathbb{R}^n$, we can identify the tangent space with $\mathbb{R}^n$, and thus use the previously discussed notation dx^i as the basis for the tangent space:

$$\omega = \sum_{\sigma} a_\sigma dx^{\sigma_1} \wedge \cdots \wedge dx^{\sigma_k},$$

where a_σ is a real valued function on Ω.

We will denote these functions as $\Lambda^k(\Omega)$ for the general functions of this form. When we would like to specify the degree of smoothness of a function, we will designate this by appending the required smoothness in front of $\Lambda^k(\Omega)$. For example, if we want to denote the continuous functions we denote this space by $C\Lambda^k(\Omega)$ and the continuously differentiable functions by $C^1\Lambda^k(\Omega)$.

Exterior derivative
The exterior derivative, which maps $\Lambda^k(\Omega)$ into $\Lambda^{k+1}(\Omega)$ is the fundamental operator of exterior calculus. Given $\omega \in \Lambda^k(\Omega)$, the exterior derivative is defined as:

$$(d\omega)_x(u_0, \ldots, u_k) = \sum_{j=0}^{k} (-1)^j \partial_{u_j} \omega_x(u_0, \ldots, \hat{u}_j, \ldots, u_k) \tag{3.28}$$

Where $\hat{u}_j$ indicates that this vector is removed. This is well defined, given all partial differentials exist. Additionally, since $\omega \in \Lambda^k(\Omega)$, ω only takes k tangent vectors, while the left hand side of (3.28) is defined over $k+1$ tangent vectors, as is required for an element of $\Lambda^{k+1}(\Omega)$.

In the case when Ω is a sub domain of $\mathbb{R}^n$, (3.28) becomes:

$$d\omega = \sum_{\sigma} \sum_{j=0}^{n} \frac{\partial a_\sigma}{\partial x_j} dx^j \wedge dx^{\sigma_1} \wedge \cdots \wedge dx^{\sigma_k}.$$

Example 7. In the case of $k = 0$, the exterior derivative on a subdomain of $\mathbb{R}^n$ becomes:

$$d\omega = \sum_{j=1}^{n} \frac{\partial a}{\partial x_j} dx^j$$

If we identify the 1-form with the vectors of $\mathbb{R}^n$, we can see that the exterior derivative becomes the vector gradient.

Lastly, we introduce the space $H\Lambda^k(\Omega)$. $H\Lambda^k(\Omega)$ is the space of $\omega \in L^2\Lambda^k(\Omega)$, such that $d\omega \in L^2\Lambda^{k+1}(\Omega)$. Clearly, it must be that $H^1\Lambda^k(\Omega) \subset H\Lambda^k(\Omega)$.

3.5. Finite element differential forms

Next, we will introduce finite dimensional differential forms. These are based on polynomials. In this section, we will introduce two spaces, namely the regular polynomial complex and the space of trimmed polynomials. The latter being a subspace of the former, but with better properties.

Definition 3.4. The space of polynomial differential forms $\mathbb{P}_r\Lambda^k(\Omega)$ are the differential forms ω which can be written as:

$$\omega = \sum_{\sigma} p_\sigma dx^{\sigma_1} \wedge \cdots \wedge dx^{\sigma_k} \tag{3.29}$$

with $p_\sigma \in \mathbb{P}_r(\Omega)$ where $\mathbb{P}_r(\Omega)$ is the space of all polynomials up to degree r over the domain Ω. Here we define $\mathbb{P}_r\Lambda^k(\Omega)$ to be zero when $r < 0$.

Remark 1. The exterior derivative maps $\mathbb{P}_r\Lambda^k(\Omega)$ into $\mathbb{P}_{r-1}\Lambda^{k+1}(\Omega)$. Clearly, the derivative operator maps $\mathbb{P}_r$ into $\mathbb{P}_{r-1}$. This is thus a direct result from (3.28).

The space of polynomial differential forms is a Hilbert complex where the polynomial degree decreases as the index of the complex increases:

$$\mathbb{P}_r\Lambda^0(\Omega) \xrightarrow{d} \mathbb{P}_{r-1}\Lambda^1(\Omega) \xrightarrow{d} \dots \xrightarrow{d} \mathbb{P}_{r-n}\Lambda^n(\Omega) \tag{3.30}$$

Next we will introduce the space of trimmed polynomials. For this space, we will first define the Koszul operator κ, which is defined as:

$$(\kappa\omega)_x(u_1, \dots, u_{k-1}) = \omega_x(x, u_1, \dots, u_{k-1}), \quad x \in \Omega, u_1, \dots, u_{k-1} \in \mathbb{R}^n. \tag{3.31}$$

The Koszul operator turns out to map $\mathbb{P}_r\Lambda^k(\Omega)$ into $\mathbb{P}_{r+1}\Lambda^{k-1}(\Omega)$ and is thus the reverse of d. We can now define space of trimmed polynomial differential forms:

Definition 3.5. The trimmed space of polynomial forms of degree r is defined as:

$$\mathbb{P}_r^-\Lambda^k(\Omega) := \mathbb{P}_{r-1}\Lambda^k(\Omega) + \kappa\mathcal{H}_{r-1}\Lambda^{k+1}(\Omega), \tag{3.32}$$

where $\mathcal{H}_r\Lambda^k(\Omega)$ is the space of k-forms with homogeneous polynomial coefficients of degree r.

Just like the space of polynomial forms, the space of trimmed polynomial forms, is also a Hilbert complex:

$$\mathbb{P}_r^-\Lambda^0(\Omega) \xrightarrow{d} \mathbb{P}_r^-\Lambda^1(\Omega) \xrightarrow{d} \dots \xrightarrow{d} \mathbb{P}_r^-\Lambda^n(\Omega) \tag{3.33}$$

Note that in contrast to the polynomial forms, the trimmed polynomial forms do not decrease in degree.

3.5.1. Canonical Projections

Lastly, we will define the canonical projection $\Pi_h^{can} : C\Lambda^k(\Omega) \to \mathbb{P}_r^-\Lambda^k(\Omega)$ which is defined by the requirement that:

$$\int_f \text{tr}_f(\omega - \Pi_h^{can}\omega) \wedge \mu = 0, \quad \forall \mu \in \mathbb{P}_{r+k-d-1}\Lambda^{d-k}(f), \quad \forall f \in \Delta_d(T), \quad n \geq d \geq k. \tag{3.34}$$

Since these are all the degrees of freedom to functions in $\mathbb{P}_r^-\Lambda^k(\Omega)$, the canonical projection I_h is well defined. The following theorem allows us to extend this definition to a Sobolev space:

Theorem 3.3. *(D. N. Arnold, 2018, Thm. 7.6) Denote by I_h, the canonical projection of $I_h : C\Lambda^k(\Omega) \to \mathbb{P}_r^-\Lambda^k(\Omega)$. Let $1 \leq p \leq \infty$ and $(n-k)/p < s \leq r+1$. Then Π_h^{can} extends boundedly to $W_p^s\Lambda^k(\Omega)$, and there exists a constant C independent of h such that:*

$$\|\omega - \Pi_h^{can}\omega\|_{L^p\Lambda^k(\Omega)} \leq Ch^s|\omega|_{W_p^s\Lambda^k(\Omega)}, \quad \omega \in W_p^s\Lambda^k(\Omega). \tag{3.35}$$

However, this theorem is not perfect. For example, we cannot project from $L^2\Lambda^k(\Omega)$, as then $p = 2, s = 0$ which requires $k > n$. Which is impossible. A practical argument to see that this is impossible, is to consider the case that f is a single vertex $x \in \Omega$, then $\int_f \text{tr}_f(\omega)$ is the point evaluation of ω at x, which is in issue for $\omega \in L^2\Lambda^k(\Omega)$. This means that bounded commuting projectors, that are defined on the entirety of $L^2\Lambda^k(\Omega)$, are still needed/desired.

3.6. Remarks on FEEC and Splines

Returning to THB-splines, when we desire a numerical method for the abstract hodge laplace problem, which is consistent, stable, accurate and efficient, we desire a couple properties. First of all, the approximation property of THB-spline is a result of the fact that THB-splines are locally refined B-splines. Furthermore, for B-spline, it has been shown by (Sande et al., 2020) that B-spline have a better approximation power per degree of freedom than regular linear basis functions. This means that THB-splines are a good choice for finite element spaces, and the accuracy and efficiency requirements are immediately satisfied.

Additionally, we require the space to obey the subcomplex property. Here, we will restrict our efforts to exact complexes. These are complexes for which the range of d^k is equal to the kernel of d^{k+1}:

$$\mathcal{R}(d^k) = \mathcal{N}(d^{k+1}). \tag{3.36}$$

On the unit domain $\Omega = [0,1]^n$, Table 3.1 shows for which spline spaces there are assumptions, underwhich the spline space is exact. Here, the additional spline spaces, T-splines, LR-splines and simplex splines are mentioned. Even-though, they are not been covered, they have been added for completeness.

Dimension	1	2	3	$\mathbb{N}$
B-spline	✓	✓	✓:(Buffa et al., 2011)	-
(T)HB-spline	✓	✓:(Evans et al., 2020)	-	-
T-splines	✓	✓:(Buffa et al., 2014)	-	-
LR-splines	✓	✓:(Johannessen et al., 2015)	-	-
simplex splines	✓	-	-	-

Table 3.1: Exactness of the L^2 De Rham complex for a unit square domain $\Omega = (0,1)^n$. Here a - means that no literature has been found on the matter.

Lastly, a set of commuting projectors needs to exist. In Table 3.2, one can see for which space commuting projectors exist in academic literature. Additionally, there are columns to indicate whether they are local or not.

Dimension	1	local	2	local	3	local	$\mathbb{N}$	local
B-spline	✓	✓	✓	✓	✓:(Buffa et al., 2011)	✓	-	-
(T)HB-spline	✓	-	✗[1]:(Evans et al., 2020, p. 465)	-	-	-	-	-
T-splines	✓	-	-	-	-	-	-	-
LR-splines	✓	-	-	-	-	-	-	-
simplex splines	✓	-	-	-	-	-	-	-

Table 3.2: Existence of (local) commuting cochain projectors.

4

Univariate Projector

Our proposed projector consists of two steps. Initially, the target function $f \in L^2(\Omega)$ is projected into intermediary function space $\mathbb{F}$, a C^0 smooth locally polynomial space by the projector F. The second step is to project Ff onto the desired THB-spline function space $\mathbb{T}$ by the projector T. Then, the final projector THB-spline projector is given by $\Pi = TF$.

$$\begin{array}{ccc} L^2(\Omega) & & \\ \downarrow F & \searrow^{\Pi} & \\ \mathbb{F} & & \\ \downarrow T & & \\ \mathbb{T} & & \end{array} \tag{4.1}$$

The projector is dependent on the following three assumptions on the THB-mesh.

Assumption 1. Every refinement domain is given by the union of lower level spline supports. So for $l = 0, \dots, L$:

$$\Omega_{l+1} = \bigcup_{b_{i,l}(x) \in S} \operatorname{supp}\left(b_{i,l}(x)\right). \tag{4.2}$$

For a subset $S \subset \mathcal{S}_{\vec{p},l,\xi_l}$.

Assumption 2. Element grading. Only THB-splines from two consecutive refinement levels are present on any element e. Thus any element e has splines from:

- either level l, or
- either level l and level $l+1$, or
- either level l and level $l-1$.

Assumption 3. The THB-spline space is maximally smooth, meaning that, all the knots, besides the border knots, have a multiplicity of 1 for the required B-spline spaces $\mathbb{S}_{p,\xi_l}$.

Both Assumption 1 and Assumption 2 appear in academic literature. Assumption 1 appears in (Evans et al., 2020) as one of the assumptions to proof exactness of the THB-spline complex in 2D. Assumption 2 appears in a stronger form in (Giust et al., 2020). In (Giust et al., 2020) the requirement of two consecutive levels is required for the supports of all the spline basis functions[1].

Additionally, the projector developed in this section, will allow for multiple choices of the projector F on to the intermediary space $\mathbb{F}$ (this space is fixed, but the projector F can be chosen). We will develop assumptions for F, and for this paper we have chosen a projector that conforms to these assumptions.

[1] Since each (TH)B-spline is defined on at least $p+1$ consecutive elements.

4.1. Projection elements

In the case of B-splines, on every element there are $p+1$ B-splines which are all linearly independent. This means that there is a well defined mapping from a $p+1$ linearly independent polynomial space onto the B-splines. However, in the case of THB-splines, certain elements might have more than $p+1$ THB-splines, and in this case, these THB-splines will be linearly dependent. We call these elements **overloaded**. See Figure 4.1, where both a polynomial space and a THB-spline space of degree two can be seen. The fourth element from the left contains more THB-spline basis functions than the required three basis functions. This is an issue, because if we would try to map a function from the polynomial space $\mathbb{P}$ onto the THB-spline basis functions on this element, there is an extra degree of freedom that can be chosen arbitrarily. This means that this projection step is not well posed. To get around this issue, multiple elements will be grouped to form projection elements, so that on every projection element, this mapping is well posed.

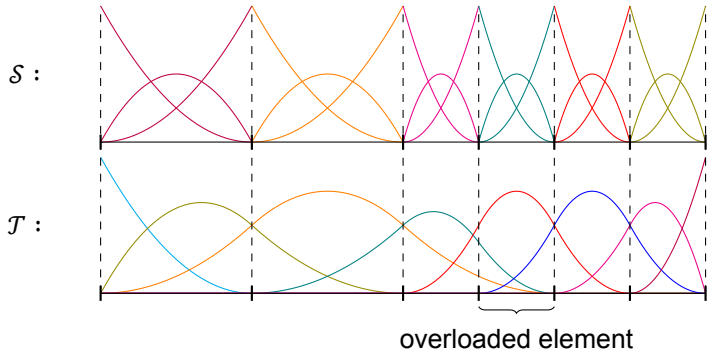


Figure 4.1: Two function spaces over a THB-mesh. The top space is a polynomial space of degree two over all elements. The bottom space is a THB-spline space. Note that for the THB-spline space, there are four THB-splines on one of the elements. This is an issue, as for any linearly independent polynomial space of degree two, there should be exactly three basis functions. We call these elements overloaded elements.

From Assumption 2, there are either elements with a single level of THB-splines, or elements with THB-splines from two levels. From the local Polynomial structure of the (TH)B-splines, the former elements are well defined. This is a result, from the fact, that on elements with a single level, the restriction of the THB-splines are B-splines. Every HB-spline, starts as a B-spline, and is only truncated on an element, if there is a refined HB-spline, from a different level. This is clearly not the case for elements with THB-spline from a single level. Meaning that the mapping from a polynomial space on to the THB-splines (for this element) is well posed. We will refer to these elements as **projection elements** of **type 0**.

For those elements with THB-splines from different levels, displayed in Figure 4.1, the THB-spline basis functions might be overloaded. Due to this overloading, the THB-spline space to map into is linearly dependent. To remedy this, the elements that contain splines from different levels, will be grouped to form multiple, bigger projection elements. These projection elements will be called **projection elements** of **type 1**. From Assumption 1 and Assumption 2, we will first show that these projection elements of type 1 can be chosen to be the first p mesh elements on the refined side of the boundary, which splits the coarse and refined domain. Secondly, we will show that this choice of type 1 projection elements results in a set of THB-splines which are linearly independent, which is required for a well posed projection.

4.1.1. Construction of projection elements of type 1

To see which elements have splines from two levels, we refer to the construction of the THB-spline space and in particular, the truncation step. In the truncation step, the HB-splines are initially mapped into the space of the refined B-splines over the entire domain. This results in the HB-splines being a linear combination of the refined B-splines. In Figure 4.2, a THB-spline space and a refined B-spline space can be seen. The basis functions of the refined B-spline space are colored red for those basis

functions that are added in the refinement step, and the other B-spline basis function are colored blue.

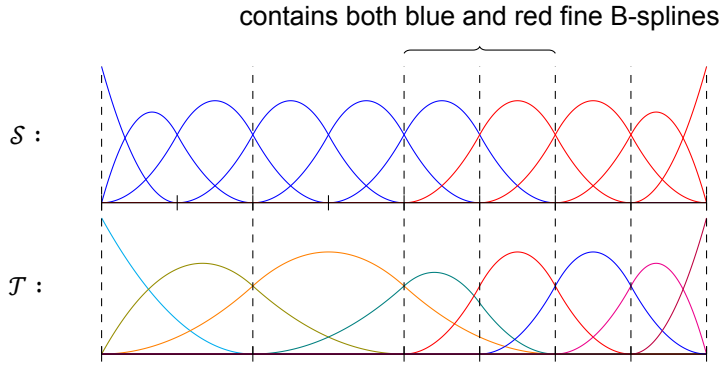


Figure 4.2: Two spline spaces of degree $p = 2$. In the bottom graph, a THB-spline space $\mathcal{T}$ with two levels of refinement. In the top graph, the B-spline space $\mathcal{S}$ of the level 2 refined spline of the bottom graph. In the top graph, the B-spline have been colored red, to indicate that these are also elements of $\mathcal{T}$, or blue to indicate that they have not been added. Note that there are 2 elements, on which both red and blue splines have support.

When a HB-spline is truncated, the components that are from the red finer B-spline basis functions, are discarded. The truncated HB-splines are a linear combination of the blue B-spline basis functions from Figure 4.2. These blue B-splines are the basis functions that are not entirely contained in the refinement domain. Since the support of B-splines is at most $p + 1$ consecutive elements, the greatest amount of elements these blue B-splines can reach in the refinement domain, are the first p consecutive elements, and by extension, the truncated HB-splines as well. For this reason, we define the projection elements of type 1 to be the first p consecutive refined elements from every refinement border.

However, up till now, we have only assumed that there are only two different levels. A coarse and a refined level. However, because of Assumption 2, only splines of two levels can be present in any given mesh element, and by extension of the above argument, on any type 1 projection element. Secondly, the case when the refined domain has two boundaries with the coarse domain, a left and right boundary, the two projection elements, related to the two boundaries, do not overlap. This is a direct consequence of Assumption 1, as the refinement domain contains at least a coarse spline. This means that the refinement domain must be at least $p + 1$ coarse elements, and in turn, the refinement domain has $2p + 2$ elements. Since the two projection elements of type 1 consist of p consecutive elements, they are separated by at least two projection elements of type 0.

From the same counting argument, on every type 1 projection element, there are p coarse THB-splines and p fine THB-splines. Clearly, from Assumption 1, we can restrict our attention to a single boundary between a coarse and a refined domain. For this single boundary, there are p (H)B-splines that have non-empty support on the refined domain. These (H)B-splines must have non-zero components of the blue B-splines in Figure 4.1, thus after truncation, there are still p coarse THB-splines. For the refined THB-splines, there must be p, as every element can be associated with a refined THB-splines as an edge of its support. This means that on every type 1 projection element, there are a total of $2p$ THB-splines with non-empty support. Contrast this with the $p(p + 1)$ polynomials basis functions with non-empty support on the type 1 projection element of $\mathbb{F}$ (the piecewise polynomial function space of degree p over every element).

4.1.2. Linear independence

For a well-posed projection, the set of THB-splines with non-empty support will need to be linearly independent over the projection element. For this we will first proof Lemma 4.1, which states that for any type 1 projection element, there is a single element on which the THB-splines with non-empty support are linearly independent. With the help of this Lemma, we can show linear independence over the full projection element in Theorem 4.2.

Lemma 4.1. *For a univariate THB-spline space conforming to Assumption 1 and Assumption 2. For all*

type 1 projection elements, the element e of the type 1 projection element that intersects the boundary, which separates two levels. The THB-splines with non-empty support on e are linearly independent on element e.

Proof. Let the two levels on element e be denoted by $l-1$ and l. Then, element e can be denoted as $\Omega^e = [\xi_{k,l}, \xi_{k+1,l}]$ with knots from the knot-sequence from level l. Assume that the boundary that separates level l from level $l-1$ is given by $\xi_{k,l}$. Over element e there are $p+1$ B-spline from level l with non-empty support, $b_{j,l}(x)$ for $j = 1, \dots, p+1$. Additionally, before adding the B-splines from level l, there are $p+1$ B-splines from level $l-1$ with non-empty support, $b_{j,l-1}(x)$ for $j = 1, \dots, p+1$. During the construction of the HB-spline space, in the step of refinement of level l, these B-splines cannot be truncated by Assumption 2. This means that the B-splines from level $l-1$, are linearly independent on element e. Since the knot-sequence of level l is derived from the knot-sequence of level $l-1$, all the B-splines from level $l-1$ can be written as a linear combination of the $p+1$ B-splines from level l:

$$b_{j,l-1}(x) = \sum_{i=1}^{p+1} c_{j,i} b_{i,l}(x), \quad x \in [\xi_{k,l}, \xi_{k+1,l}], \quad j = 1, \dots, p+1. \tag{4.3}$$

From the assumption that the boundary that separates level $l-1$ and l is given by $\xi_{k,l}$ and from Assumption 1, in the construction of the HB-spline space, the B-spline $b_{p+1,l-1}(x)$ is removed from the HB-spline basis, and is replaced by the B-spline $b_{p+1,l}(x)$. However, locally, on element e, these two functions coincide up to a scaling factor. This is the result of the linear combinations in (4.3). Even though these linear combinations only hold on element e, the B-splines $b_{j,l-1}(x)$ can be written to be a linear combination over the entire domain. This linear combination, restricted to element e, reproduces the combinations from (4.3). However, since the support of $b_{p+1,l-1} \subseteq [\xi_{k,l}, 1]$, this linear combination can only depend on B-splines of level l with $b_{j,l} \subseteq [\xi_{k,l}, 1]$, as these B-splines are linearly independent. So, on element e we have:

$$b_{p+1,l-1}(x) = \alpha b_{p+1,l}(x), \quad x \in [\xi_{k,l}, \xi_{k+1,l}]. \tag{4.4}$$

Note, $\alpha \neq 0$, as both splines are non-zero on e. But then the following spans are equivalant and of the same rank:

$$\operatorname{span}\{b_{1,l-1}(x), \dots, b_{p+1,l-1}(x)\} = \operatorname{span}\{b_{1,l-1}(x), \dots, b_{p,l-1}(x), b_{p+1,l}(x)\} \tag{4.5}$$

Thus, the HB-splines over element e are linearly independent. Additionally, the level $l-1$ B-splines can be truncated, as follows:

$$\tilde{b}_{i,l-1}(x) = b_{i,l-1}(x) - \alpha_i b_{p+1,l}(x) = b_{i,l-1}(x) - \frac{\alpha_i}{\alpha} b_{p+1,l-1}(x), \quad x \in [\xi_{k,l}, \xi_{k+1,l}], \quad \alpha_i \in \mathbb{R}. \tag{4.6}$$

But this redefinition of the level $l-1$ splines, is again a linear combination, and thus an element of the span. Meaning that:

$$\operatorname{span}\{b_{1,l-1}(x), \dots, b_{p+1,l-1}(x)\} = \operatorname{span}\{\tilde{b}_{1,l-1}(x), \dots, \tilde{b}_{p,l-1}(x), b_{p+1,l}(x)\}, \quad x \in [\xi_{k,l}, \xi_{k+1,l}]. \tag{4.7}$$

Again, both sides must have the same rank. From (4.6), these linear combinations can never map to zero by the linear independence of the level $l-1$ B-splines. Additionally, in (4.7), the basis for the r.h.s. are the THB-splines with non-empty support on e. Thus proofing the linear independence. For the case when $\xi_{k+1,l}$ is the boundary, the proof can be altered by first showing that $b_{1,l-1}(x) = \alpha b_{1,l}(x)$, and then showing that the altered spans are again all full rank. □

Theorem 4.2. *Given a univariate THB-spline space of degree p that conforms to Assumption 2 and Assumption 1 . THB-splines with non-empty support on a type 0 or 1 projection element are locally linearly independent on that projection element.*

Proof. The type 0 projection elements are all THB-splines from a single level. These THB-splines cannot be truncated, else there are splines from a higher level present, but then this projection element would be part of a type 1 projection element. Moreover, HB-splines from a single level over a single element must be B-splines, and thus locally linearly independent.

Given a type 1 projection element, the domain can be denoted by $[\xi_{k,l}, \xi_{k+p,l}]$. From Lemma 4.1, we know that there are p THB-splines $b_{j,l-1}(x)$ with non-empty support. Over the p element of the type 1 projection element, an additional p (TH)B-splines from level l are with non-empty support, $b_{j,l}(x)$. Let them satisfy the following equation for the coefficients $c_{j,l-1}, c_{j,l}$:

$$\sum_{j=1}^{p} c_{j,l-1} b_{j,l-1}(x) + \sum_{j=1}^{p} c_{j,l} b_{j,l}(x) = 0, \quad x \in [\xi_{k,l}, \xi_{k+p,l}]. \tag{4.8}$$

From Lemma 4.1, we must have that by the linear independence on one of the elements of the type 1 projection element, that $c_{j,l-1} = 0$ for $j = 1, \dots, p$. However, the level l THB-splines are B-splines on this type 1 projection element, and thus from the local linear independence of the B-splines, we have that $c_{j,l} = 0$ for $j = 1, \dots, p$. This shows the linear independence of the THB-splines with non-empty support on the type 1 projection elements. □

Lastly, we will differentiate projection elements from mesh elements. Let ϵ be the projection element index, belonging to the projection element $\{e_1, \dots, e_p\} =: E_\epsilon$, where E_ϵ is the set of mesh element indices that make up the projection element ϵ. The projection element indices are denoted by ϵ, and the set of all projection element indices is denoted by $\Upsilon = \{\epsilon_1, \dots, \epsilon_m\}$ for a THB-spline space with m projection elements.

4.1.3. Notation and Element extensions

The notion of element support extensions has been introduced in Section 2.5. Here, the element support extension of a mesh element with index e for a B-spline space $\mathbb{S}$ is given by:

$$\tilde{\Omega^e} = \bigcup_{\substack{b(x) \in \mathcal{S} \\ \Omega^e \subset \mathrm{supp}(b(x))}} \mathrm{supp}\left(b(x)\right). \tag{4.9}$$

Where $\mathcal{S}$ is the basis of the B-spline space $\mathbb{S}$. Here, we additionally define $\tilde{e} = \{e_1, \dots, e_m\}$, so that:

$$\tilde{\Omega^e} = \cup_{i=1}^{m} \Omega^{e_i}. \tag{4.10}$$

We extend this notation to any subset $U \subset E$, so that we have $\Omega^U = \cup_{e \in U} \Omega^e$. Then, we have that $\Omega^{\tilde{e}} = \tilde{\Omega^e}$. Additionally, in a similar fashion, we can create an extension based on projection elements:

Definition 4.1. The **projection extension** $\bar{e}$ of a mesh element index e, is the union of the projection elements, on which a spline has support of some spline space $\mathbb{S}$ with non-empty support on e:

$$\bar{e} := \bigcup_{\alpha \in \tilde{e}} \{e \in E_\epsilon : \forall \epsilon \in \Upsilon, \alpha \in E_\epsilon\} \tag{4.11}$$

The same notation can be used for the domain of the support extension and the projection extension. Lastly, we can also define the support extension of a projection extension (or any combination), by taking the union of the support extension of the individual elements of the projection extension:

$$\tilde{\bar{e}} = \bigcup_{\alpha \in \bar{e}} \tilde{\alpha}. \tag{4.12}$$

Note that, we have not explicitly used the spline space $\mathbb{S}$ in the definition. It should be obvious from the context, over which spline space $\mathbb{S}$ the extension is performed.

4.2. Sub projection F

The first projection, F, is a local projection on to the space $\mathbb{F}$. This function space is the space of $C^0(\Omega)$ functions, that locally, on every element e, are Bernstein polynomials.

$$\mathbb{F} := \left\{ s \in C^0(\Omega) : s|_e \in \mathbb{P}_p(\Omega^e), \forall e \in E \right\}. \tag{4.13}$$

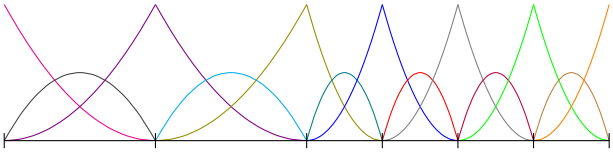


Figure 4.3: The space $\mathbb{F}$, of C^0 smooth local polynomial functions of degree $p = 2$.

See Figure 4.3 for an example of degree $p = 2$. The projection $F : L^2(\Omega) \to \mathbb{F}$ is an altered least square projection on every element on to the local Bernstein polynomial basis, where the boundary Bersntein polynomials are averaged based on a weighting scheme. The first step of F, is to perform the following integration's:

$$f_i^e := \int_{\Omega^e} f(x)\mathfrak{b}_{i,p}^e(x)dx, \quad i = \{1, 2, \dots, p+1\}, \quad e \in E. \tag{4.14}$$

Here, $\mathfrak{b}_{i,p}^e(x)$ is the Bernstein polynomial $\mathfrak{b}_{i,p}(x)$ (see Section 2.4) scaled and translated, so that instead of having domain $[0, 1]$, it has domain Ω^e. The integration can be done with various different numerical methods. We have used Legendre-Gauss Quadrature where the number of nodes N has to be chosen such that the error is on the level of machine precision. This is done by increasing N until it has no more effect on the accuracy. Then, on each element, the projection onto the local Bernstein basis is given by:

$$\vec{\beta}^e := \frac{1}{|\Omega^e|} G^{-1} \begin{bmatrix} f_1^e & f_2^e & \cdots & f_{p+1}^e \end{bmatrix}^T, \quad \forall e \in E. \tag{4.15}$$

Where $|\Omega^e|$ is the size of element e, and G^{-1} is the inverse Gramian Matrix given in (2.23). The local projection on element e is then given by:

$$s^e(x) = \sum_{i=1}^{p+1} \beta_i^e \mathfrak{b}_{i,p}^e(x), \quad x \in \Omega^e. \tag{4.16}$$

In order to make sure that the projection is C^0 smooth, we will have to regularize the projection on the element boundaries. Notice that in (2.14), only $\mathfrak{b}_{1,p}(x)$ is non-zero on $x = 0$, and thus is $\mathfrak{b}_{1,p}^e(x)$ the only polynomial that is non-zero on the left boundary on e. Likewise, $\mathfrak{b}_{p+1,p}(x)$ and $\mathfrak{b}_{p+1,p}^e(x)$ are the only polynomials that are non-zero on $x = 1$ and the right boundary. Now, let e_1 and e_2 be two subsequent elements such that e_1 is to the left of e_2. Denote the size of e_1 as $|\Omega^{e_1}|$ and the size of e_2 as $|\Omega^{e_2}|$. Then, we replace both the values $\beta_{p+1}^{e_1}$ and $\beta_1^{e_2}$ by:

$$\beta_{e_1,e_2} := \frac{|\Omega^{e_1}|\,\beta_{p+1}^{e_1} + |\Omega^{e_2}|\,\beta_1^{e_2}}{|\Omega^{e_1}| + |\Omega^{e_2}|}. \tag{4.17}$$

This forces the splines on e_1 and e_2 to coincide on the border between the two elements[2]. We repeat this step for all borders between two mesh elements. The combination of these two steps, the local projection, and the regularization over all element boundaries, defines the projector F for any element e, with "left" border element e_1, and "right" element border e_2:

$$s^e(x) = \beta_{e_1,e}\mathfrak{b}_{1,p}^e(x) + \sum_{i=2}^{p} \beta_i^e \mathfrak{b}_{i,p}^e(x) + \beta_{e,e_2}\mathfrak{b}_{p+1,p}^e(x), \quad x \in \Omega^e. \tag{4.18}$$

Additionally, since β_{e_1,e_2} is a linear combination with positive weights, it can be estimated as follows:

$$|\beta_{e_1,e_2}| \leq \max\left\{|\beta_{p+1}^{e_1}|, |\beta_1^{e_2}|\right\}. \tag{4.19}$$

[2]Note that this regularization step differs from the basis functions seen in Figure 4.3. In this space, the basis functions are C^0 smooth. However, in this proposed projector, we have set the values of the boundary functions to a weighted average. In theory this does not matter, as the resulting regularity is the same. However, the implementation is different as we have extra degrees of freedom. However, these extra degrees of freedom, assist in notation.

4.2.1. Estimates

For the projector F proposed, we will show some local estimate results. For this, we need the intermediary result:

Lemma 4.3. *We have:*

$$\|Fv\|_{L^2(\Omega^e)} \leq C_{bern}\|v\|_{L^2(\tilde{\Omega^e})}. \tag{4.20}$$

Where C_p is only dependent on mesh regularity μ and polynomial degree p and the support extension is taken from the spline space $\mathbb{F}$.

Proof. Given:

$$\|Fv\|^2_{L^2(\Omega^e)} = \left\| \beta_{e_1,e}(v)\,\mathfrak{b}_1(x) + \sum_{i=2}^{p} \beta_i^e(v)\,\mathfrak{b}_i(x) + \beta_{e,e_2}(v)\,\mathfrak{b}_{p+1}(x) \right\|^2_{L^2(\Omega^e)} \tag{4.21}$$

$$\leq \|\vec{\beta}^e(v)\|^2_\infty \left\| \sum_{i=1}^{p+1} \mathfrak{b}_i(x) \right\|^2_{L^2(\Omega^e)} \tag{4.22}$$

$$= |\Omega^e| \max\left\{|\beta_{e_1,e}|, |\beta_2^e|, \dots, |\beta_p^e|, |\beta_{e,e_2}|\right\}^2 \tag{4.23}$$

$$\leq |\Omega^e| \max\left\{|\beta_{p+1}^{e_1}|, |\beta_1^e|, |\beta_2^e|, \dots, |\beta_p^e|, |\beta_{p+1}^e|, |\beta_1^{e_2}|\right\}^2 \tag{4.24}$$

$$\leq C^2_{bern}\|v\|^2_{L^2(\tilde{\Omega^e})} \tag{4.25}$$

Here, we used that the Bernstein polynomials sum to one, the estimates from (4.19) and in the last step we used Lemma 2.8. Do note that the coefficient C_{bern} is the coefficient C_p of Lemma 2.8, multiplied by the difference in size of the mesh elements e_1, e_2 compared to e. This is solely dependent on the dimension, as these elements can at most differ by one level. □

Corollary 4.3.1. *For each element e, the local Bernstein coefficient vector $\vec{\beta}^e$ over the support extension $\tilde{e}$ associated with the altered local L^2-projection of a function $f \in L^2(\Omega^e)$ onto $\mathbb{F}$, the space of polynomials of degree p over the projection element satisfies the inequality:*

$$\|\vec{\beta}^e\|_\infty \leq \frac{C_{bern}}{|\Omega^e|^{1/2}}\|f\|_{L^2(\tilde{\Omega^e})}, \tag{4.26}$$

Here, $|\Omega^e|$ is the mesh size of element e. C_{bern} is a constant only dependent upon the polynomial degree p and mesh regularity μ.

Using this lemma, we can show the following local error estimate result:

Proposition 4.4. *Let k and l be integers with $0 \leq k \leq l \leq p+1$ and $l \leq \alpha + 1$. Then for each element e we have:*

$$|v - Fv|_{H^k(\Omega^e)} \leq C h_e^{l-k}|v|_{H^l(\tilde{\Omega^e})}, \quad \forall v \in H^k\left(\tilde{\Omega^e}\right). \tag{4.27}$$

where h_e is the diameter of element e. C is independent of h_e, but possibly dependent on shape regularity, polynomial degree, continuity and k, l.

Proof. Note that the space $\mathbb{F}$ is a B-spline space $\mathbb{S}$ where every internal knot has multiplicity p. Given s as in Lemma 2.7:

$$|v - Fv|_{H^k(\Omega^e)} = |v - s + s - Fv|_{H^k(\Omega^e)} \tag{4.28}$$

$$\leq |v - s|_{H^k(\Omega^e)} + |F(v - s)|_{H^k(\Omega^e)} \tag{4.29}$$

$$\leq I + II. \tag{4.30}$$

Then, with C_{lem} the constant from Lemma 2.7:

$$I \leq C_{lem} h_e^{l-k}\|v\|_{H^l(\tilde{\Omega^e})}. \tag{4.31}$$

The standard inverse inequality for polynomials yields:

$$II \leq C_{inv} h_e^{-k} \| F\,(v - s)\,\|_{L^2(\Omega^e)}. \tag{4.32}$$

And with Lemma 4.3 and Lemma 2.7 we find:

$$II \leq C_{inv} C_{bern} C_{lem} \|_{H^l(\Omega^{\tilde{e}})}. \tag{4.33}$$

Thus the proposition holds with $C = C_{lem}\,(1 + C_{inv} C_{bern})$. □

Lastly, we will show that the projector preserves all $v \in \mathbb{F}$:

Proposition 4.5. *The projector F preserves all elements in $\mathbb{F}$:*

$$Fv = v, \quad \forall v \in \mathbb{F}. \tag{4.34}$$

Proof. Given a $v \in \mathbb{F}$, on any element e, the local representation is given by:

$$v_{|_e} = \sum_{i=1}^{p+1} \beta_i^e \mathfrak{b}_{i,p}^e(x). \tag{4.35}$$

Then, the intermediary integrals v_j are:

$$v_j = \sum_{i=1}^{p+1} \beta_i^e \int_{\Omega^e} \mathfrak{b}_{i,p}(x) \mathfrak{b}_{j,p}(x) dx = \left[G \vec{\beta}^e \right]_j. \tag{4.36}$$

Here G is the Gramian matrix. Now, the resulting projection (before regularization) is given by the vector:

$$\vec{u} = G^{-1} \vec{v} = G^{-1} G \vec{\beta}^e. \tag{4.37}$$

Where $\left[\vec{\beta}^e\right]_i = \beta_i^e$. However, before regularization, the projection is the exact initial smooth function. Thus, the regularization step does not alter the coefficients. This means that the projection preserves $v \in \mathbb{F}$. □

4.3. Sub projection T

The second projector, $T : \mathbb{F} \to \mathbb{T}$ is defined in two steps. Initially, on every projection element, the local polynomials are projected on to the THB-splines with non-empty support. This produces a local projection. However, for THB-splines that span multiple projection elements, these THB-splines will have multiple coefficients, each linked to a particular projection element. These different coefficients are combined by a weighted average to obtain a single final coefficient for each THB-spline basis function.

For the first step, let $\epsilon \in \Upsilon$ be a projection element index and let $E_\epsilon \subset E$ be the set of elements that make up the projection element with index ϵ. Let J_ϵ be the index set of all THB-splines with non-empty support on ϵ. Since the THB-splines are local polynomials on all $e \in E_\epsilon$, the THB-splines can be written as a linear combination of local polynomials expressed as bernstein polynomials:

$$b_j(x)|_\epsilon = \sum_{e \in E_\epsilon} \sum_{i=1}^{p+1} C_{j,i}^e \mathfrak{b}_{i,p}^e(x), \quad \forall j \in J_\epsilon. \tag{4.38}$$

Here, $b_j(x)|_\epsilon$ is the restriction of THB-spline j to the projection element ϵ. In this case, by choosing an ordering of $e \in E_\epsilon$, a matrix C^ϵ can be constructed, where every column represents a THB-spline $j \in J_\epsilon$ and where the coefficients represent the linear combination given in (4.38). Thus, given a $\vec{\gamma}^\epsilon$ of a linear combination of THB-splines $j \in J_\epsilon$ with non-empty support on ϵ, this representation can be mapped to the local polynomial space $\mathbb{F}$, restricted to ϵ by:

$$\vec{\beta}^\epsilon = C^\epsilon\, \vec{\gamma}^\epsilon. \tag{4.39}$$

In the case of T, we desire to map the space of local polynomials in to the space of THB-splines with non-empty support on ϵ. In the case of type 0 projection elements, there are $p+1$ THB-splines and local polynomials on ϵ. For the case of type 1 projection elements, there are $2p$ THB-splines and $p(p+1)$ local polynomials on ϵ. Then the matrix C^ϵ is a matrix of size $m \times n$, with $m \geq n$. Additionally, as a consequence from Proposition 4.2, the columns of C^ϵ are linearly independent. This means that a Moore-Penrose pseudo-inverse exists and is given by:

$$C^{\epsilon\,+} := \left(C^{\epsilon\,T}\, C^\epsilon\right)^{-1} C^{\epsilon\,T}. \tag{4.40}$$

The Moore-Penrose pseudo-inverse is the solution to the following minimization problem:

$$x = \operatorname{argmin}_{u \in \mathbb{R}^n} \|v - C^\epsilon\, u\|_2 = C^{\epsilon\,+}\, v, \quad v \in \mathbb{R}^m. \tag{4.41}$$

By constructing the vector $\vec{\beta}^\epsilon$ from $\vec{\beta}^e, e \in E_\epsilon$ in the same order as C^ϵ, such that the i^{th} index of $\vec{\beta}^\epsilon$ and the i^{th} row of C^ϵ correlate to the same local polynomial function defined over ϵ, the local projection on projection element ϵ is given by:

$$\vec{\gamma}^\epsilon := C^{\epsilon\,+}\, \vec{\beta}^\epsilon, \quad \forall \epsilon \in E_\epsilon. \tag{4.42}$$

The vector $\vec{\gamma}^\epsilon$ is a local THB projection on a projection element ϵ. Thus on projection element ϵ, the local projection is given by:

$$s(x)|_\epsilon = \sum_{j \in J_\epsilon} \gamma_j^\epsilon b_j(x). \tag{4.43}$$

Here, $\gamma_j^\epsilon(x)$ are the THB-splines indexed by J_ϵ that have support on ϵ. In order to obtain a global projection for a given THB-spline j, all the coefficients that are related to the THB-spline j will need to be combined. For this, let $\Upsilon_j = \{\epsilon_1, \epsilon_2, \ldots, \epsilon_n\} \subset \Upsilon$ be the set of projection elements on which the THB-spline j has non-empty support. Then, the final coefficient γ_j for THB-spline j is given by:

$$\gamma_j := \sum_{\epsilon \in \Upsilon_j} \omega_j^\epsilon \gamma_j^\epsilon. \tag{4.44}$$

Where the weights ω_j^ϵ are given by:

$$\omega_j^\epsilon := \frac{\int_{\Omega^\epsilon} b_j(x)dx}{\int_\Omega b_j(x)dx}. \tag{4.45}$$

This produces the global THB-spline projection:

$$s(x) = \sum_j \gamma_j b_j(x). \tag{4.46}$$

4.3.1. Construction of the local projection matrices

The C^ϵ matrices can be constructed by knot insertion, see Section 2.1.3. Since the Bernstein polynomials are B-splines defined over a $(p+1)$-open knot-sequence with only two distinct knots (the boundaries), we can write all (T)(H)B-splines as local Bernstein polynomials by knot inserting all element boundaries till they have a multiplicity of $p+1$. However, in the case of (T)HB-splines, we have splines of multiple levels. Fortunately, all the splines are of at most two different levels on every projection element. Next to that, on the type 1 projection elements, the coarse THB-splines are already a known linear combination of the finer splines (as this is what the truncation process does). And on type 0 projection elements, the THB-splines are all locally B-splines. This means that on any projection element, the THB-splines are a linear combination of B-splines, all from the same level. Then, add all element boundaries of the projection element until the knots have multiplicity $p+1$. See Figure 4.4 for an example on a type 1 projection element.

This process relates any THB-spline j with non-empty support on ϵ to the local Bernstein polynomials. Doing this for all THB-splines with non-empty support, the matrix C^ϵ can be constructed. In the case of type 0 projection elements that are at least p elements away from the global boundary, the eigenvalues of the Moore-Penrose pseudo-inverse are given in Proposition 4.6.

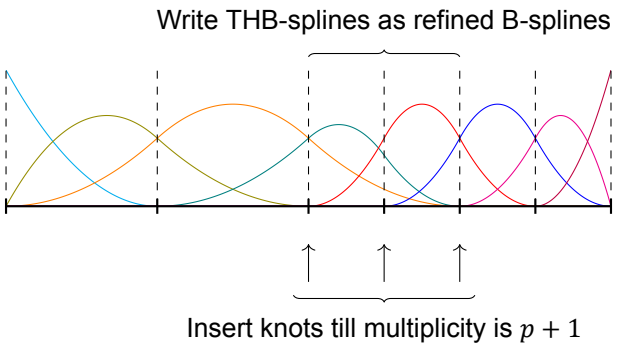


Figure 4.4: The construction process of C^{ϵ} for the only type 1 projection element over this THB-spline space. Initially, write all THB-splines with non-empty support as a linear combination of the refined B-splines. Secondly, alter these linear combination coefficients with knot insertion untill all the knots on the projection element have multiplicity $p+1$. By raising the multiplicity of the element boundaries, the knot-sequence on every element becomes $p+1$-open, and thus the B-splines over the elements are the local Bernstein polynomials.

Proposition 4.6. *Given an equidistant knot-sequence ξ and an element e which is at least p elements away from the global border, the projection matrix $C^{\epsilon}{}^{+}$ over the type 0 projection element ϵ over element e, from the local Bernstein polynomials to the B-splines with support on element e has the $p+1$ eigenvalues $1, p, p\cdot(p-1), \ldots, p!$.*

Proof. The projection matrix $C^{\epsilon}{}^{+}$ satisfies both of the following two equations.

$$\vec{\gamma} = C^{\epsilon}{}^{+}\vec{\beta}, \tag{4.47}$$

$$\sum_{i=1}^{p+1}\gamma_i b_i(x) = \sum_{i=1}^{p+1}\sum_{j=1}^{p+1}\left[C^{\epsilon}{}^{+}\right]_{i,j}\beta_j \mathfrak{b}_j(x), \quad x\in\Omega^{\epsilon}. \tag{4.48}$$

Since both the B-splines and the Bernstein polynomials satisfy the partition of unity, $\vec{1}$ must be an eigenvector with eigenvalue 1. For the other eigenvectors and values, we will integrate a $p-1$ degree B-spline/Bernstein polynomial vector. The operator that maps $I_p : \mathbb{S}_{p-1,\xi} \to \mathbb{S}_{p,\xi}$[3][4] by integration is given by the $p+1\times p$ matrix:

$$I_p = \frac{1}{p}\begin{bmatrix} \xi_{i+p}-\xi_i & & & \\ \xi_{i+p}-\xi_i & \xi_{i+p+1}-\xi_{i+1} & & \\ \xi_{i+p}-\xi_i & \xi_{i+p+1}-\xi_{i+1} & \xi_{i+p+2}-\xi_{i+2} & \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{4.49}$$

Here, i is chosen such that when element e is given by $\Omega^e = [\xi_m, \xi_{m+1}]$, we have that $\xi_{i+p} = \xi_{m+1}$. This integral operator corresponds to integrating the spline functions with the condition that the function is zero at the left boundary. Now note that in the case of B-splines, for which the element is at least p elements away from the border, we must have that all elements of (4.49) are $p|\Omega^e|$. Likewise, in the case of the Bernstein Polynomials, all elements are $|\Omega^e|$ because $\xi_{i+p} = \xi_{i+p+1} = \cdots = \xi_{m+1}$ and $\xi_i = \cdots = \xi_{i+p-1} = \xi_m$. This means that if we distinguish the integral operators for the B-splines as I_p^b and for the Bernstein polynomials as I_p^b, we have that $I_p^b = pI_p^p$.

The last step to obtain all the eigenvectors and eigenvalues, is to keep integrating the k'th dimensional vector $\vec{1}_k$ to the degree p spaces. Clearly, $\vec{1}_k \in \mathbb{S}_{k,\xi}$ corresponds (by partition of unity) to the unit

[3] Since the Bernstein polynomials are a special case B-spline space, we only have to do the derivation for the B-spline case.

[4] Strictly speaking, the knot-sequence needs to be altered, as the initial and last knots must appear exactly $p+1$ times. However, since the degree is known, we will assume that it is implied.

function, and likewise for the Bernstein polynomials. So we must have that:

$$\vec{\beta}_k = \left(\prod_{l=k}^{p} I_l^p\right)\vec{1}_k, \tag{4.50}$$

$$\vec{\gamma}_k = \left(\prod_{l=k}^{p} I_l^b\right)\vec{1}_k = \left(\prod_{l=k}^{p} l I_l^p\right)\vec{1}_k = \left(\prod_{l=k}^{p} l\right)\vec{\beta}_k. \tag{4.51}$$

However, both vectors must describe the same function, as we are integrating the unit function $p-k$ times in both cases. From this we conclude that $\vec{\beta}_k$ is an eigenvector with eigenvalue $\prod_{l=k}^{p} l$. □

For the other type 0 projection elements, no expression has been found. However, from numerical experiments the largest eigenvalue seems to coincide with $p!$. Likewise, for the type 1 projection elements, no bounds have been found. However, since the THB-splines and Bernstein polynomials are invariant under translation and scaling, these matrices should be independent on location and element diameter h_e, but can depend on spline degree p, shape regularity of the mesh and continuity.

4.4. Local estimates of full projector

In this section, we will show that the combination $\Pi = TF$ is locally stable. For this we will first require that the initial projector conforms to the following assumptions:

Assumption 4. The initial projector $F : L^2(\Omega) \to \mathbb{F}$ conforms to the following assumptions for any $e \in E$. Here we have that the coefficients $\vec{\beta}^e$ are the coefficients related to $\mathfrak{b}_i^e(x)$ that represent the projected coefficient on element e.

$$Fs = s, \qquad \forall s, \in \mathbb{F}, \qquad \text{(continuous polynomial preserving)} \tag{4.52}$$

$$\|\vec{\beta}^e\|_\infty \leq \frac{C_{stab,F}}{|\Omega^e|^{1/2}}\|s\|_{L^2(\Omega^{\tilde{e}})}, \qquad \forall s \in L^2\left(\Omega^{\tilde{e}}\right). \qquad \text{(local stability property)} \tag{4.53}$$

Where $C_{stab,F}$ is a constant independent on $|\Omega^e|$ and the support extension is taken in the $\mathbb{F}$ spline space.

Note that the projector F in Section 4.2 conforms to this assumption by Corollary 4.3.1 and 4.5. However, any other projector conforming to these assumptions can also be chosen.

Then, the full projector Π is locally stable over the domain $e^* := \tilde{\tilde{e}}$, the support extension of the projection extension (see Section 4.1.3). Here, the projection extension is taken over $\mathbb{T}$ and the support extension over $\mathbb{F}$. Now, from the construction of $\mathbb{F}$, the support extension is equivalent to the collection of all elements neighbouring and including $\tilde{e}$. For this element, we have the following result:

Lemma 4.7. *Let k and l be integer indices with $0 \leq k \leq l \leq p+1$. Given e, and support extension $\tilde{e}$, $v \in H^l\left(\Omega^{e^*}\right)$, there exists an $s \in \mathbb{T}$ such that:*

$$|v - s|_{H^k(\Omega^{e^*})} \leq C_{int} h_e^{l-k} |v|_{H^l(\Omega^{e^*})}. \tag{4.54}$$

Where C_{int} is a constant independent on element diameter h_e, but possibly on p, k, l, μ.

Proof. The above proof relies on Lemma 2.7. However, this Lemma requires a B-spline space. For this, we will construct a B-spline space that is locally contained in $\mathbb{T}$, and for which the local B-splines defined over e^* are independent on level of refinement e.

For this, note that e^* consists of splines of at most 6 levels. To see why, note e has at most 2 levels by Assumption 2. Then, the projection extension can have splines of at most 4 levels, as each projection element can also have at most two levels of splines. The final support extension brings it up to at most 6 different levels. By taking l to be this lowest level, all the $b_{i,l,\xi_l}(x) \in \mathcal{S}_{\vec{p},\xi_l}$ with support on e^*, are elements of $\mathbb{T}$.

Next, there must be a level l element e_l such that e is contained in e_l. Then, the support extension $\tilde{e}_l$ might contain e^*. If it does, the $s \in \mathbb{S}_{\vec{p},\boldsymbol{\xi}_l}$ from Lemma 2.7 proofs the lemma. If it does not, we can remove knots of $\boldsymbol{\xi}_l$ to construct a new knot-sequence Ξ and B-spline space $\mathbb{S}_{\vec{p},\Xi} \subset \mathbb{S}_{\vec{p},\boldsymbol{\xi}_l}$.

When we remove knots that correspond to boundaries strictly within $\tilde{e}_l$, the underlying mesh changes. This means that there is a new element e_Ξ that contains e, but crucially, this must also contain e_l. Thus the support extension $\tilde{e}_\Xi$ contains $\tilde{e}_l$. By increasingly removing knots, the support extension $\tilde{e}_\Xi$ will contain e^* at some point. Also, note that the amount of elements (which is related to the number of internal knots) is bounded by some function of p. Then, choose $s \in \mathbb{S}_{\vec{p},\Xi}$ according to Lemma 2.7. Then we have that $s \in \mathbb{S}_{\vec{p},\Xi} \subset \mathbb{S}_{\vec{p},\boldsymbol{\xi}_l}$. Additionally, all the level l B-splines from s with support on e^* can be reproduced in $\mathbb{T}$, so $s \in \mathbb{T}$. □

Additionally, as a direct consequence of the local stability property:

Lemma 4.8. *Given a projector F as in Assumption 4, the local stability over a projection element is bounded by:*

$$\|\vec{\beta}^\epsilon\|_\infty \leq \frac{C_{stab,F}}{|\Omega^{e,\epsilon}|^{1/2}} \|s\|_{L^2(\Omega^{\tilde{\epsilon}})}, \quad \forall s \in L^2\left(\Omega^{\tilde{\epsilon}}\right). \tag{4.55}$$

Where $\Omega^{e,\epsilon}$ is the smallest mesh element of ϵ.

Proof. This is a direct result from the local stability property in Assumption 4. Note that by construction of the projection element ϵ, all the mesh elements $e \in E_\epsilon$ have the same size $|\Omega^{e,\epsilon}|$. □

Then, the full projector $\Pi = TF$ has the following two properties:

Proposition 4.9. *We have:*

$$\Pi s = s, \qquad \forall s \in \mathbb{T} \qquad \textit{(spline-preserving property)} \tag{4.56}$$

$$\|\Pi v\|_{L^2(\Omega^e)} \leq C_{stab} \|v\|_{L^2\left(\Omega^{e^*}\right)}, \qquad \forall v \in L^2\left(\Omega^{e^*}\right), \forall e \in E \qquad \textit{(local stability property)} \tag{4.57}$$

Here, e^*, is the support extension of the support extension of e. C_{stab} is a constant independent of h, but possibly dependent on the shape regularity of the mesh, polynomial degree and continuity.

Proof. The spline-preserving property holds trivially as the splines are also elements of $\mathbb{F}$, so F preserves them by Assumption 4. For T, the correct spline is given by the optimal pseudo inverse solution. This means that T preserves the splines. Hence, what remains to prove is the local stability property. Given any element e and $f \in L^2\left(\Omega^{e^*}\right)$, denote the set J_e as the set of THB-spline indices, which have non-empty support on Ω^e:

$$J_e := \left\{j : \Omega^e \subset \operatorname{supp}\left(b_j(x)\right)\right\} \tag{4.58}$$

Then we have that:

$$\Pi f|_e(x) = \sum_{j \in J_e} \left[\sum_{\epsilon \in \Upsilon_j} \omega_j^\epsilon \gamma_j^\epsilon(f) \right] b_j(x) \tag{4.59}$$

Here, ω_j^ϵ are the weighting coefficients of the recombination set that are all non negative and sum to one for any given splines j. Lastly, $\gamma_j^\epsilon(f)$ are the local THB-spline projection coefficients defined as:

$$\vec{\gamma}^\epsilon(f) = C^{\epsilon\,+} \vec{\beta}^\epsilon(f)$$

Where $C^{\epsilon\,+}$ is the local THB-spline projection matrix and $\vec{\beta}^\epsilon(f)$ the local Bernstein coefficients over projection element ϵ as a result of F. Since the projection matrices $C^{\epsilon\,+}$ only depend on p, shape regularity and continuity and by Lemma 4.8, we have:

$$\|\vec{\gamma}^\epsilon(f)\|_\infty \leq \frac{C_\lambda}{|\Omega^{e,\epsilon}|^{1/2}} \|f\|_{L^2(\Omega^{\tilde{\epsilon}})}$$

Here, $|\Omega_{e,\epsilon}|$ is the size of the smallest element of ϵ, and C_λ a constant which only depends on the polynomial degree p and the shape regularity of the parametric mesh. Next, similar as in (Thomas et al., 2015), we have that:

$$|\Pi f(x)| \leq \left| \sum_{j \in J_e} \left[\sum_{\epsilon \in \Upsilon_j} \omega_j^\epsilon \gamma_j^\epsilon (f) \right] b_j(x) \right| \tag{4.60}$$

$$\leq \left| \sum_{j \in J_e} \left[\sum_{\epsilon \in \Upsilon_j} \frac{C_\lambda}{|\Omega^{e,\epsilon}|^{1/2}} \|f\|_{L^2(\Omega^{\tilde{\epsilon}})} \right] b_j(x) \right| \tag{4.61}$$

$$\leq \max_{j \in J_e} \left[\sum_{\epsilon \in \Upsilon_j} \frac{C_\lambda}{|\Omega^{e,\epsilon}|^{1/2}} \|f\|_{L^2(\Omega^{\tilde{\epsilon}})} \right] \left| \sum_j b_j(x) \right| \tag{4.62}$$

$$\leq \max_{j \in J_e} \left[\sum_{\epsilon \in \Upsilon_j} \frac{C_\lambda}{|\Omega^{e,\epsilon}|^{1/2}} \|f\|_{L^2(\Omega^{\tilde{\epsilon}})} \right] \tag{4.63}$$

$$\leq \frac{C_\lambda^*}{|\Omega^e|^{1/2}} \|f\|_{L^2(\Omega^{e^*})} \tag{4.64}$$

Here, e^* is the projection extension of e. The constant C_λ^* does absorb some constants that relate the size of $|\Omega^e|$ to all the sizes of $|\Omega^{e,\epsilon}|$, but this is solely dependent on the mesh regularity μ and p and the maximal difference in levels (which is 6). Integrating over e we find:

$$\|\Pi f\|_{L^2(\Omega^e)} \leq \frac{C_\lambda^*}{|\Omega^e|^{1/2}} \|f\|_{L^2(\Omega^{e^*})} \left(\int_{\Omega^e} 1^2 dx \right)^{1/2} = C_\lambda^* \|f\|_{L^2(\Omega^{e^*})} \tag{4.65}$$

□

The local approximation error is then given by:

Theorem 4.10. *Let k and l be integer indices with $0 \leq k \leq l \leq p+1$. For each mesh element e, the following inequality holds:*

$$\|f - \Pi f\|_{H^k(\Omega^e)} \leq C_{app} h_e^{l-k} \|f\|_{H^l(\Omega^{e^*})}, \quad \forall f \in H^l\left(\Omega^{e^*}\right), \tag{4.66}$$

where h_e is the element diameter of e, e^* is the support extension of the support extension of e, and C_{int} is a constant independent of h_e but possibly dependent on the shape regularity of the mesh, polynomial degree, number of refinement levels L, continuity, and the parameters k and l.

Proof. Let s be as in Lemma 4.7. Then, by Proposition 4.9:

$$|v - \Pi v|_{H^k(\Omega^e)} = |v - s + s - \Pi v|_{H^k(\Omega^e)} \tag{4.67}$$

$$= |v - s + \Pi[s - v]|_{H^k(\Omega^e)} \tag{4.68}$$

$$\leq |v - s|_{H^k(\Omega^e)} + |\Pi[v - s]|_{H^k(\Omega^e)}. \tag{4.69}$$

Note that the second term is a polynomial (because of the projection) on element e. Thus by a standard inverse inequality for polynomials (see for example (Bernardi et al., 2007)), we get:

$$|\Pi[v - s]|_{H^k(\Omega^e)} \leq C_{inv} h_e^{-k} \|\Pi[v - s]\|_{L^2(\Omega^e)} \tag{4.70}$$

$$\leq C_{inv} C_{stab} h_e^{-k} \|v - s\|_{L^2(\Omega^{e^*})} \quad \text{(Proposition 4.9)} \tag{4.71}$$

$$\leq C_{inv} C_{stab} C_{int} h_e^{l-k} \|v\|_{H^l(\Omega^{e^*})}. \quad \text{(Lemma 4.7)} \tag{4.72}$$

The first term can immediately be estimated by Lemma 4.7, to obtain the result where,

$$C_{app} = C_{int} \left(1 + C_{inv} C_{stab}\right)$$

which only depends on k, l, p, μ. □

4.5. Commutativity

To construct a commuting projector, a similar approach that was introduced in (Buffa et al., 2011) will be used. Given the following diagram:

$$\begin{array}{ccc} H^1(\Omega) & \xrightarrow{\frac{d}{dx}} & L^2(\Omega) \\ \downarrow{\scriptstyle F^0} & & \downarrow{\scriptstyle F^1} \\ \mathbb{F}^0 & \xrightarrow{\frac{d}{dx}} & \mathbb{F}^1 \\ \downarrow{\scriptstyle T^0} & & \downarrow{\scriptstyle T^1} \\ \mathbb{T}^0 & \xrightarrow{\frac{d}{dx}} & \mathbb{T}^1 \end{array} \tag{4.73}$$

Here, $\mathbb{T}^0$ and $\mathbb{T}^1$ represent the THB-spline spaces on which to project. The first THB-spline space $\mathbb{T}^0$ has degree p, and the second THB-spline space $\mathbb{T}^1$ has degree $p-1$. Secondly, we have the intermediate spaces $\mathbb{F}^0$ and $\mathbb{F}^1$ where the former is of degree p and the latter of degree $p-1$. Additionally, F^0 and T^0 are the sub projectors derived in previous sections. For F^1, we can define the commuting projector F^1 inspired by (Buffa et al., 2011) as follows:

$$F^1[f](x) := \frac{d}{dx} F^0 \left[\int_0^x f(s) ds \right](x) \tag{4.74}$$

Lemma 4.11. *The projectors F^0 and F^1 commutes with the exterior derivative.*

$$\frac{d}{dx} F^0 f = F^1 \frac{d}{dx} f, \quad f \in H^1(\Omega). \tag{4.75}$$

Proof. Let $f \in H^1(\Omega)$, then:

$$\begin{aligned} F^1 \frac{d}{dx} f &= \frac{d}{dx} F^0 \left[\int_0^x \frac{d}{dx} f(s) ds \right] && (4.76) \\ &= \frac{d}{dx} F^0 [f + C] && (4.77) \\ &= \frac{d}{dx} F^0 f + \frac{d}{dx} C && (4.78) \\ &= \frac{d}{dx} F^0 f && (4.79) \end{aligned}$$

Here, C is a constant. The projector F^0 does not alter constants, since the projector F^0 does not alter splines. Since this was for an arbitrary $f \in H^1(\Omega)$, we conclude it holds for all $f \in H^1(\Omega)$. □

Likewise, the commuting projector T^1 is constructed as $T^1[f](x) := dT^0 \left[\int_0^x f(s) ds \right](x)$. Yet, since the domain of this projector is a finite dimensional function space, the local integral operator can be explicitly calculated (see equation (4.49)). However, this integral operator only acts locally. Fortunately, the functions in $\mathbb{F}^0$ are C^0 continuous. So, for any function $f \in \mathbb{F}^0$, after integrating $\frac{d}{dx} f$, we again obtain a C^0 smooth function. For this reason, by working from the left to right, we add constants on each element until the local integral matches the left integral on the border. Denote this finite dimensional integral operator by I, resulting in:

$$T^1[f](x) := dT^0 [If](x) \tag{4.80}$$

These projectors also commute.

Lemma 4.12. *The projectors T^0 and T^1 commute with the exterior derivative.*

$$\frac{d}{dx} T^0 f = T^1 \frac{d}{dx} f, \quad f \in \mathbb{F}^0. \tag{4.81}$$

The proof of this lemma is equivalent to the proof of lemma 4.11. Combining these results, we can construct local THB-spline projectors as:

$$\Pi^0 f(x) := T^0 F^0 f(x), \quad \forall f \in H^1(\Omega) \tag{4.82}$$

$$\Pi^1 f(x) := \frac{d}{dx} T^0 F^0 \left[\int_0^x f(s) ds \right](x), \quad \forall f \in L^2(\Omega) \tag{4.83}$$

And these projectors also commute with the exterior derivative.

Theorem 4.13. *The local THB-spline projectors Π^0 and Π^1 commute with the exterior derivative:*

$$\frac{d}{dx} \Pi^0 f = \Pi^1 \frac{d}{dx} f, \quad \forall f \in H^1(\Omega). \tag{4.84}$$

Proof. The theorem is the direct result from Lemma 4.11 and Lemma 4.12. □

5

Multivariate Bezier Projector

The multivariate projector is again structured in two subsequent projectors, starting with a local projector F on a Bernstein polynomial space $\mathbb{F}$ and followed by a subsequent THB-spline projector T on to $\mathbb{T}$. For the multivariate case, both will be an extension of the univariate projectors. Let Ω be a domain in n-dimensions, then the final projector $\Pi := TF$ is defined over the diagram:

$$\begin{array}{ccc} L^2(\Omega) & & \\ \downarrow F & \searrow & \\ \mathbb{F} & & \Pi \\ \downarrow T & \swarrow & \\ \mathbb{T} & & \end{array} \tag{5.1}$$

First, we will extend the notion of projection elements to higher dimensions. Followed by the construction of F and T. We will end this chapter with estimates for the projector Π in the multivariate setting.

5.1. Projection elements

In the multivariate setting, there will be more types of projection elements, instead of the two types for the univariate case (see Section 4.1). For an n-dimensional domain, we will have $n+1$ different projection elements, ranging from type 0 to type n projection elements. We will first introduce a process to determine generator elements. These are the elements from which a projection element is generated. These projection element generators will also aid us in proofing linear independence, and are thus a useful concept.

By Assumption 2, it suffices to look at a refinements from level $l-1$ to level l. Then, all refined elements e of level l, which intersect the boundary $\partial\Omega_l \backslash \partial\Omega_0$, are projection element generators and will be assigned a type. This will be done by looking at the refined B-splines that are inserted in the refinement step. Clearly these level l B-splines, are from the B-spline space $\mathbb{S}_{\vec{p},\vec{\xi}_l}$. In the B-spline space $\mathbb{S}_{\vec{p},\vec{\xi}_l}$, there are B-splines that have support on the element e, of which a subset is inserted in the refinement step to level l. See Figure 5.1 for an example with degree $\vec{p} = (2,2)$. Here, for different elements e, the B-splines of the B-spline space $\mathbb{S}_{\vec{p},\vec{\xi}_l}$ are marked in the center of their support. They are marked with either a $\times$ for a level l B-splines that is not inserted, or denoted with a $\circ$ if they are inserted to the THB-spline basis. See the construction of the HB-spline space in Section 2.2.

By the tensor product structure of the B-spline space, these B-splines can be denoted by a vector index $\vec{i}$:

$$\vec{i} = (i_1, i_2, \dots, i_n), \to b_{\vec{i},\vec{p},\vec{\xi}_l}(x) \in \mathbb{S}_{\vec{p},\vec{\xi}_l}, \quad \text{such that } \Omega^e \subset \operatorname{supp}\left(b_{\vec{i},\vec{p},\vec{\xi}_l}(x)\right). \tag{5.2}$$

Here each i_d can be (up to) $p_d + 1$ different values. Note that the collection of these indices will create a hyper-rectangle (see Figure 5.1, where the marked B-splines define a rectangle). Now, given some

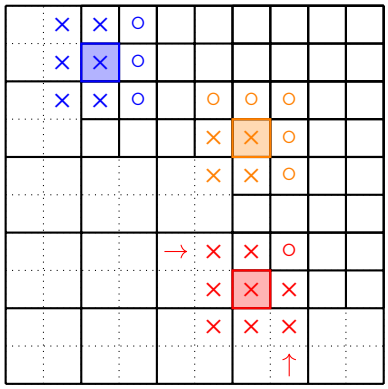


Figure 5.1: Given $\vec{p} = (2, 2)$ and a THB-spline space. For three different refined elements, that neighbour the border $\partial\Omega_l \backslash \partial\Omega_0$ (colored red, orange and blue), the nine B-splines of level $l-1$ that have support on each element. These B-splines have either been marked with a $\circ$ to denote that they are inserted in to the THB-spline basis, or by $\times$ when they are not. Here the $\times$ or $\circ$ is placed in the center of the support of the B-spline. Note that for $\vec{p} = (2, 2)$ the B-splines have a support on a collection of 3×3 elements of level l.

$\vec{i}$ and a dimension d, we can a create set of $p_d + 1$ different indices $I_{\vec{i},d}$, that only differ in the d'th component. Here the d'th component runs over all the possible i_d, such that these indices conform to (5.2). Note that in Figure 5.1, this corresponds to either a row or column of possible indices.

For a given dimension d, we call that dimension **refined** on element e, if there is a vector $\vec{i}$ for which the set $I_{\vec{i},d}$ contains exactly one B-spline of level l that is inserted in to the THB-spline basis. See for example the red arrows in Figure 5.1. To recap, a dimension d of element e is **refined** if:

$$\exists \vec{i} : \exists! \vec{j} \in I_{\vec{i},d} : \operatorname{supp}\left(b_{\vec{j},\vec{p},\vec{\xi}_l}(x)\right) \subseteq \Omega_l. \tag{5.3}$$

Let e be a refined element of level l that intersects the boundary $\partial\Omega_l \backslash \partial\Omega_0$. Element e is a **type-k projection element generator**, if there are exactly k refined dimensions. In figure 5.1, the red and orange elements are type-2 generators, and the blue element is a type-1 generator.

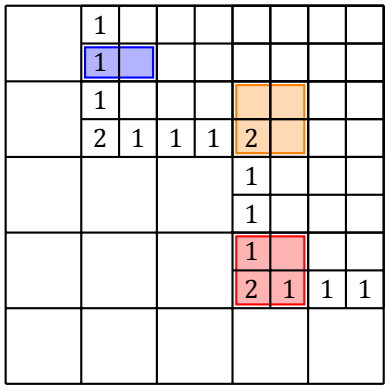


Figure 5.2: For the THB-spline space in Figure 5.1, the different types of projection element generators, for all the elements that intersect the refinement boundary $\partial\Omega_l \backslash \partial\Omega_0$. Additionally, the projection elements generated by the colored elements from Figure 5.1 are drawn in color.

These generator elements will generate projection elements. A type-k projection element is a rectangular collection of elements in the directions of the refined dimensions. If the dimensions $d_1, \ldots, d_k$ are refined, then one B-spline of level l that must be inserted. This B-spline is either the first or the last for a given set $I_{\vec{i},d_1}$. If it is the first, d_1 will be extended into the -1 direction, while if it is the last, it will be extended into the $+1$ direction. Starting from element e, the first p_1 refined elements[1] of level l in

[1] Note that the existence of a single refined B-spline in $I_{\vec{i},d_1}$, means that all these elements are in fact elements of the THB-spline space.

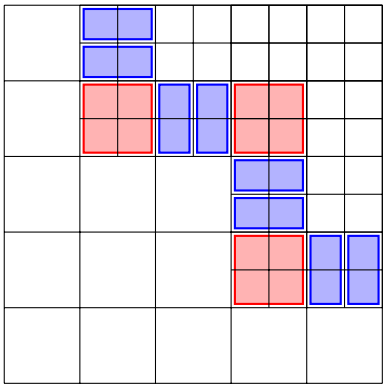


Figure 5.3: The projection elements for the THB-spline space of Figure 5.1. Here the red projection elements, are projection elements of type-2, the blue projection elements are type-1 and the remaining elements are all type-0.

the extension direction of d_1 are added. Then, the first p_2 refined elements of level l in the extension direction of d_2 are added. This process is repeated until this has been done for all refined dimensions. Finally, the minimum amount of elements are added, to create a rectangular collection of refined elements of level l.

In Figure 5.2, the projection elements generated by the colored generator elements of Figure 5.1 are shown. From this Figure, it becomes clear that some projection elements can/will overlap. For example, in Figure 5.2, the red projection element is a type-2 projection element, and will contain two projection elements of type-1. In this case, we will remove the lower type projection elements[2]. Additionally, if instead $\vec{p} = (2, 5)$ would have been chosen, the two type-2 projection elements overlap (see the red and orange projection elements in Figure 5.2). In the case that two projection elements of the same type overlap (after first removing all the contained lower type projection elements), the proof presented below for linear independence, fails. For this reason, we will require that the mesh does not produce these kind of projection elements. See the following assumption.

Assumption 5. Given a THB-spline mesh, after removing lower type projection elements, no projection elements of the same type overlap.

The notion of removing lower type projection elements that are contained in higher type projection elements, is quite natural. For example, in Figure 5.2, the red type-2 generator is next to two type-1 generators. These type-1 generators must have a dimension that does not contain a row (or column) with a singular refined B-spline. In fact, they have rows (or columns) with either none, or with two. For those rows (or columns) with two, the exact same row (or columns) in the next element over, contains a single refined B-spline. Thus, this dimension is refined in the next element over. This is the case for our type-2 starting element. For this reason, the type-2 projection element is in someway an extension of the type-1 elements that are removed. However, in the proof of linear independence, these type-1 projection element generators are still useful. Lastly, all the remaining elements that are not covered by any type-k with $k \geq 1$ projection elements, are all individual type-0 projection elements. In Figure 5.3 the final projection elements can be seen.

5.1.1. Linear independence

Linear independence is proven in multiple steps. We will first introduce some definitions which will lead to a local description of which B-splines are inserted and which are removed. Following this, we will proof that the projection element generators have a linearly independent set of basis functions with support on the generator. From this, we will show that the THB-splines with support over the rest of the projection element, are linearly independent.

We start with the introduction of the complement support of any projection element generator:

[2]From the construction, these lower types of projection elements that get removed, are always full contained in a projection element of a higher type.

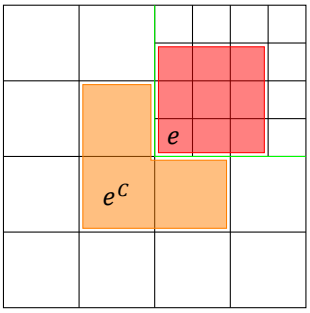


Figure 5.4: An example of a support complement of an element e.

Definition 5.1. Given a generator element e, related to the refinement of level $l-1$ to level l. The **complement support** e^C is given by the collection of elements from level $l-1$ such that they intersect e:

$$e^C := \left\{\alpha \in E^{l-1} : \Omega^e \cap \Omega^\alpha \cap \partial\Omega_l \neq \emptyset\right\} \tag{5.4}$$

And for the projection element ϵ related to the generator element e, the **complement support** of a projection element:

$$\epsilon^C := \left\{\alpha \in E^{l-1} : \Omega^\epsilon \cap \Omega^\alpha \cap \partial\Omega_l \neq \emptyset\right\} \tag{5.5}$$

See Figure 5.4 for an example of a complement support. This definition coupled with Assumption 5, allows us to characterise the B-splines that are removed, by whether they have support on the complement support of a projection element:

Lemma 5.1. *Given a projection element ϵ in a THB-spline space conforming to Assumptions 1, 2 and 5. A level $l-1$ B-spline still has support on ϵ after refinement and truncation to level l, if and only if it has support on ϵ^C.*

Proof. Clearly, if $b_{i,l-1}(x)$ has support on ϵ^C, its support is not entirely contained in Ω_l, and this spline will be truncated. After truncation, it is a linear combination of level l B-splines whose support is not entirely contained in Ω_l, and will thus still have support on ϵ.

However, if $b_{i,l-1}(x)$ has no support on ϵ^C, it is not directly obvious that it will be removed, or truncated until it has no support on ϵ. If it is removed, we are done, so let $b_{i,l-1}(x)$ be a spline with no support on ϵ^C, that is not removed. Then, by the construction of the THB-spline space, it must have support on some level $l-1$ elements $e_1, \dots, e_m$ that border the boundary $\partial\Omega_l \backslash \partial\Omega_0$:

$$\exists e_j \in E^{l-1} : \operatorname{supp}\left(b_{i,l-1}(x)\right) \cap \Omega^{e_j} \neq \emptyset. \tag{5.6}$$

However, these elements must all border projection elements, and this produces some list of projection elements $\epsilon_1, \dots, \epsilon_m$ on which $b_{i,l-1}(x)$ also has support.

$$\operatorname{supp}\left(b_{i,l-1}(x)\right) \cap \Omega^{\epsilon_j} \neq \emptyset. \tag{5.7}$$

But then, when we truncate $b_{i,l-1}(x)$, it becomes a linear combination of level l B-splines that must have support on some level $l-1$ elements, and whose support is a subset of the support of $b_{i,l-1}(x)$. This forces these level l B-splines to only have support on the projection elements $\epsilon_1, \dots, \epsilon_m$.

$$\operatorname{supp}\left[\operatorname{trunc}\left(b_{i,l-1}(x)\right)\right] \cap \Omega_l \subseteq \Omega^{\epsilon_1} \cup \dots \cup \Omega^{\epsilon_m}. \tag{5.8}$$

However, by Assumption 5, the r.h.s. does not intersect the original projection element ϵ. So, when $b_{i,l-1}(x)$ has no support on ϵ^C, and is not entirely contained in Ω_l, the truncation of $b_{i,l-1}(x)$ has no support on ϵ. □

Next, we proof that for any type-k generator element e, the THB-splines with support on e are linearly independent.

Lemma 5.2. *Given an n-dimensional THB-spline space that conforms to Assumptions 1, 2 and 5, then for any type k projection element, generated by element e, the THB-splines with support on e are linearly independent.*

Proof. Let the projection element ϵ contain THB-splines from level $l-1$ and l (by Assumption 2). Then, the generator element $e = e^l$ is a mesh element of refinement level l, which must be contained in some level $l-1$ mesh element e^{l-1}. In the refinement step to level l, we start with a collection of B-splines of level $l-1$ (at least, they must locally be B-splines on element e^{l-1}, as this element will be refined). But then, this set of B-spline must be linearly independent on element e^{l-1} (and thus also e^l) by properties of the B-splines:

$$\left\{b_{i,l-1}(x)\right\}. \tag{5.9}$$

This set of B-splines will be split in two. Those level $l-1$ B-splines that get removed, as they are fully contained in Ω_l, and those that remain, and will be truncated. Now, note that the B-splines that get removed, cannot have any support on the support complement of e, e^C by Lemma 5.1[3]. So we can denote the B-splines of level $l-1$ that are to be removed, by whether they have support on e^C:

$$\left\{b_{i,l-1}(x)\right\}_{i=1}^{n} = \left\{b_{i,l-1}^{C}(x)\right\}_{i=1}^{n-n_{l-1}^{\not C}} \cup \left\{b_{i,l-1}^{\not C}(x)\right\}_{i=1}^{n_{l-1}^{\not C}} \tag{5.10}$$

Here, superscript C denotes the splines with support on e^C, and $\not C$ if they have no support on e^C and $n_{l-1}^{\not C}$ is the number of splines without support on e^C. The latter set will be removed and replaced by level l B-splines that also have no support on e^C. Additionally, since both sets are B-spline spaces, where the level $l-1$ B-splines that are removed and the level l B-splines have the same requirement of not having support on e^C, they must consist of the same number of splines. So we have:

$$\left|\left\{b_{i,l-1}^{\not C}(x)\right\}_{i=1}^{n_{l-1}^{\not C}}\right| = \left|\left\{b_{i,l}^{\not C}(x)\right\}_{i=1}^{n_{l}^{\not C}}\right| \tag{5.11}$$

So $n_{l-1}^{\not C} = n_l^{\not C} = n^{\not C}$. Next, since we have that $\mathbb{S}_{\vec{p},\vec{\xi}_{l-1}} \subset \mathbb{S}_{\vec{p},\vec{\xi}_l}$, the removed B-splines $b_{i,l-1}^{\not C}(x)$ can be written as a linear combination of level l B-splines. However, again because of the support argument over e^C, we must have that:

$$b_{i,l-1}^{\not C}(x) = \sum_{j=1}^{n^{\not C}} c_{i,j}\, b_{j,l}^{\not C}(x), \quad \forall i = 1, \dots, n^{\not C}, \quad x \in \Omega^e. \tag{5.12}$$

But then, the map between the two sets must be full rank since none of these B-splines is zero over element e. Thus, we must have that the following set is linearly independent over element e:

$$\left\{b_{i,l-1}^{C}(x)\right\}_{i=1}^{n-n^{\not C}} \cup \left\{b_{i,l}^{\not C}(x)\right\}_{i=1}^{n^{\not C}}, \quad x \in \Omega^e. \tag{5.13}$$

To find the THB-spline basis, we have to truncate the B-spline of the first set by removing a linear combination of the second set. But this does not alter linearly independence. So we conclude that the THB-splines over the element e are linearly independent.

□

Note that the previous lemma only shows that the projection element generator is linearly independent. However, from the construction of a type-k projection element, it contains lower type projection elements. This allows us to partition the level $l-1$ THB-splines with support on ϵ as follows:

$$\left\{b_{i,l-1}(x)\right\}_{i=1}^{n} = \bigcup_{m=1}^{k} \left\{b_{j,l-1}^{m}(x)\right\}_{j=1}^{n_m} = \bigcup_{m=1}^{k} B_m. \tag{5.14}$$

[3]Strictly speaking, some level $l-1$ B-splines do not get removed. However, by truncation, they no longer have support on ϵ. The result is still the same, these splines get removed from the set of B-splines with support on ϵ.

Where $b^m_{i,l=1}(x)$ is a level $l-1$ B-spline that has support on a type-m generator element in ϵ, but not on any generator element of a type higher than m. To proof that the THB-splines on the projection element are linearly independent, we will show that for any linear combination of THB-splines that is zero:

$$\sum_{m=1}^{k}\sum_{j=1}^{n_m} c^m_{j,l-1} b^m_{j,l-1}(x) + \sum_{i=1}^{n} c_{i,l} b_{i,l}(x) = 0. \tag{5.15}$$

That the coefficients $c^m_{j,l-1}$ and $c_{i,l}$ are all zero. Clearly, by Lemma 5.2, all $c^k_{j,l-1}$ are zero. Next, showing that sequentially all $c^m_{j,l-1}$ are zero with first $m = k-1$ down to $k = 1$, the first sum vanishes. Then, since the second sum is over non truncated B-splines from level l, which must be linearly independent on these elements, we find that $c_{i,l}$ are all zero. This is the outline for the proof of linear independence over the projection elements.

To show this, we will call all the projection elements of type m with $m < k$, in a type-k projection element, a **sub projection element of type-k**. These are exactly the removed projection elements in the construction of the type-k projection elements. Next, introducing the concept of restricted linearly independence:

Definition 5.2. A sub projection element generator e of type $1 \leq m \leq k$ of a projection element with splines of level $l-1$ and level l of type k is called **restricted linearly independent**, if it is linearly independent over the following set:

$$B_{m|_e} \cup \left\{ b_{i,l}(x) : \Omega^e \subset \operatorname{supp}\left(b_{i,l}(x)\right) \subseteq \Omega_l \right\} \tag{5.16}$$

Here $B_{m|_e}$ are the elements of B_m that have support on Ω^e.

Then, the following lemma shows that the boundary elements of ϵ that intersect the border $\partial\Omega_l$ are restricted linearly independent.

Lemma 5.3. *Let ϵ be a projection element of type k in a THB-spline space conforming to Assumptions 1, 2 and 5 with splines of level $l-1$ and l. The sub projection elements of type $1 \leq m \leq k$ are restricted linearly independent.*

Proof. This lemma is proven with induction. The base case of $m = k$, is proven by Lemma 5.2. Only the induction step remains to be proven. That is, if the Lemma holds for sub projection elements of type-$m+1, \dots, k$, then it holds for sub projection elements of type-m. Assume that the sub projection elements of type-$m+1, \dots, k$ are all restricted linearly independent. Now, let e be the generator element of one of the sub projection elements of type-m. Then, over e, before refinement to level l, there are n B-splines of level $l-1$ defined over element e:

$$\left\{ b_{i,l-1}(x) \right\}_{i=1}^{n}. \tag{5.17}$$

This set must be linearly independent on element e. Now, we can split this set in two parts, those B-splines that will be truncated to THB-elements of the sets $B_{m+1}, \dots, B_k$, and the complement:

$$\left\{ b_{i,l-1}(x) \right\}_{i=1}^{n} = \left\{ b_{j,l-1}(x) : \operatorname{trunc}\left(b_{j,l-1}(x)\right) \in \bigcup_{\alpha=m+1}^{k} B_\alpha \right\} \cup \left\{ b^m_{i,l-1}(x) \right\}_{i=1}^{n_m}. \tag{5.18}$$

To show restricted linear independence, only the latter set is of importance. The B-splines of the latter set come in two flavours. Those B-splines that are entirely contained in Ω_l (and will thus be removed), or those B-splines that have support on e, but must also have support on the complement support e^C by Lemma 5.1. This means the latter set can be split in two:

$$\left\{ b^m_{i,l-1}(x) \right\}_{i=1}^{n_m} = \left\{ b^{m,C}_{i,l-1}(x) \right\}_{i=1}^{n^C_m} \cup \left\{ b^{m,\not C}_{i,l-1}(x) \right\}_{i=1}^{n^{\not C}_m}. \tag{5.19}$$

The latter set is replaced by a set of level l B-splines, entirely contained in Ω_l. Denote these splines by $b_{i,l}^{\mathcal{C}'}(x)$. For these level l B-splines, we must have that they are linearly independent, and span the same space as $b_{i,l-1}^{\mathcal{C}'}(x)$ (these are a bigger set then the level $l-1$ B-splines in (5.19)). Thus we have:

$$\operatorname{span}\left\{b_{i,l-1}^{\mathcal{C}'}(x)\right\}_{i=1}^{n^{\mathcal{C}'}} = \operatorname{span}\left\{b_{i,l}^{\mathcal{C}'}(x)\right\}_{i=1}^{n^{\mathcal{C}'}}, \quad x \in \Omega^e. \tag{5.20}$$

Next, both bases are linearly independent. But then, since $\left\{b_{i,l-1}^{m,\mathcal{C}'}(x)\right\}_{i=1}^{n_m^{\mathcal{C}'}}$ is a subset of the l.h.s. of (5.20), we must have that the following set of splines are linearly independent over element e by the splitting of (5.19):

$$\left\{b_{i,l-1}^{m,\mathcal{C}}(x)\right\}_{i=1}^{n_m^{\mathcal{C}}} \cup \left\{b_{i,l}^{\mathcal{C}'}(x)\right\}_{i=1}^{n^{\mathcal{C}'}}. \tag{5.21}$$

By now truncating the level $l-1$ B-splines in (5.21), the first set of (5.21) becomes the grade m restricted truncated THB-splines of level $l-1$, restricted to those splines with non-empty support on element e:

$$B_m^{res}|_e \cup \left\{b_{i,l}^{\mathcal{C}'}(x)\right\}_{i=1}^{n^{\mathcal{C}'}}. \tag{5.22}$$

As stated before, the truncation operations do not affect the linear independence of this set, as they produce new linear combinations that span the same space. This proves the restricted linear independence of grade m. So, by the induction hypotheses, this holds for all $m = 1, \dots, k$. □

We can now proof linear independence over the entire projection elements:

Proposition 5.4. *Given a THB-spline space conforming to Assumptions 1, 2 and 5 and let ϵ be a projection element of type k. Then this projection element is linearly independent.*

Proof. Split the THB-splines over ϵ in the truncated level $l-1$ THB-splines and the level l B-splines. Then, given a linear combination that sums to zero:

$$\sum_{i=1}^{n} c_{i,l-1} b_{i,l-1}(x) + \sum_{j=1}^{m} c_{j,l} b_{j,l}(x) = 0, \quad x \in \Omega^{\epsilon}. \tag{5.23}$$

From the splitting in (5.14), the first sum can be rewritten to be:

$$\sum_{m=1}^{k} \sum_{i=1}^{n_m} c_{i,l-1}^{m} b_{i,l-1}^{m}(x) + \sum_{j=1}^{m} c_{j,l} b_{j,l}(x) = 0, \quad x \in \Omega^{\epsilon}. \tag{5.24}$$

Starting with $m = k$, and any $b_{i,l_1}^{k}(x)$, this B-spline has support on some type-k generator element by definition. By Lemma 5.3 the coefficient $c_{i,l-1}^{k} = 0$. This holds for all $b_{i,l-1}^{k}(x) \in B_k$. But then, the same argument holds for all $b_{i,l-1}^{k-1} \in B_{k-1}$ by Lemma 5.3. Repeat this to find that all the $c_{i,l-1}^{m} = 0$. Resulting in:

$$\sum_{j=1}^{m} c_{j,l} b_{j,l}(x) = 0, \quad x \in \Omega^{\epsilon}. \tag{5.25}$$

However, these THB-splines are unaltered B-splines of level l, forcing the coefficients to be zero by the linear independence of the B-splines. This means that the THB-splines with support on ϵ are linear independent. □

5.2. Sub projection F

Given the mesh element e, the local Bernstein projection F is given by the tensor product of the one dimensional projection with a regularization step. This can be done since the mesh elements are all

higher dimensional rectangles that can be written as a tensor product of one dimensional domains. In this case, the local Bernstein polynomials basis is given by a tensor product:

$$\mathcal{P}_{\vec{p}} := \mathcal{P}_{p_1} \otimes \mathcal{P}_{p_2} \otimes \cdots \otimes \mathcal{P}_{p_n} \tag{5.26}$$

And thus, the basis functions are associated with the index vector $\vec{i} = (i_1, i_2, \dots, i_D)^T$ given by:

$$\mathfrak{b}_{\vec{i},\vec{p}}(x_1, \dots, x_n) := \prod_{d=1}^{n} \mathfrak{b}_{i_d,p_d}(x_d), \quad \mathfrak{b}_{i_d,p_d}(x_d) \in \mathcal{P}_{p_d}. \tag{5.27}$$

These $\vec{i}$ are generated by taking tensor products of the index set per dimension. Given dimension d of degree p_d, there are $p_d + 1$ local Bernstein polynomials and denote $I_d := \{1, 2, \dots, p_d + 1\}$ as the index set of these Bernstein polynomials. Then, take $I := I_1 \otimes I_2 \otimes \cdots \otimes I_n$ to be the multivariate index set. In the two dimensional case, with $\vec{p} = (3, 2)$, these can be associated with nodes on the domain as in Figure 5.5.

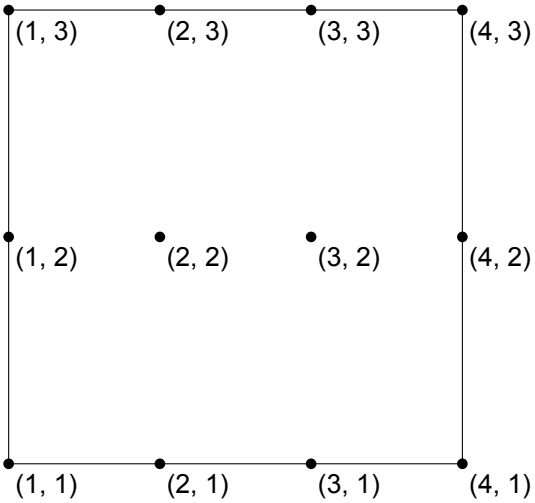


Figure 5.5: Nodal points of a degree $\vec{p} = (3, 2)$ 2D Bernstein polynomial space over some element e.

To construct the full dimensional projector $G_{\vec{p}}$, let G_{p_d} be the the one dimensional projector for dimension d, see (2.23). Then, the full dimensional local Bernstein projector is given by a tensor of the 1D projection matrices:

$$G_{\vec{p}} := G_{p_1} \otimes G_{p_2} \otimes \cdots \otimes G_{p_n} \tag{5.28}$$

Additionally, the integrals are given by:

$$f^e_{\vec{i}} = \int_{\Omega^e} \mathfrak{b}_{\vec{i},\vec{p}}(x_1, \dots, x_n) f(x_1, \dots, x_n) dx_1 \cdots dx_n \tag{5.29}$$

These are again calculated numerically by Legendre-Gauss Quadrature, where the number of nodal points will need to be chosen as such, that they have no effect on the accuracy of the later projections. Denote the full collection of all of these elements as $\vec{f}^e$. The local Bernstein polynomial projection is then given by:

$$\vec{\beta}^e = G_{\vec{p}} \vec{f}^e \tag{5.30}$$

5.2.1. Regularization

Secondly, the projection is regularized over the element boundaries, to produce a C^0 smooth projection. This smooth space $\mathbb{F}$, will be a particular THB-spline space. Until now, we have assumed that the B-spline spaces used to construct the THB-spline spaces are $(p + 1)$ open, where all knots (besides the border knots) have multiplicity 1, see Assumption 3. This produces a maximally smooth THB-spline space. However, if we create a different THB-spline space, where the B-spline spaces are again $(p+1)$

open, but all the interior knots have multiplicity p, we obtain the desired C^0 smooth space $\mathbb{F}$.

This THB-spline space is C^0 smooth, since all the basis functions are C^0 smooth. Additionally, any basis function of one of the B-spline spaces is a local Bernstein polynomial on most elements. The case where all knots have multiplicity $p+1$, is proven in Section 2.4. However, if we decrease the multiplicity, and thus increase the regularity from C^{-1} to C^0 on any element, only the B-spline on the borders are combined. However, these B-splines that are combined, must coincide on the border, and are thus not altered. This immediately means that any non truncated THB-spline in $\mathbb{F}$, restricted to any mesh element e, is a Bernstein polynomial. This is not the case for the truncated THB-splines, as on some elements, they are truncated/altered. However, since we desire a C^0 smooth space, we cannot expect this to be the case, as on the border between two levels, the refined Bernstein polynomials have more degrees of freedom. See Figure 5.6, here on the red and blue edges. The C^0 smooth functions imply that some of basis functions of $\mathbb{F}$, cannot be a local Bernstein polynomial, but must instead be some linear combination.

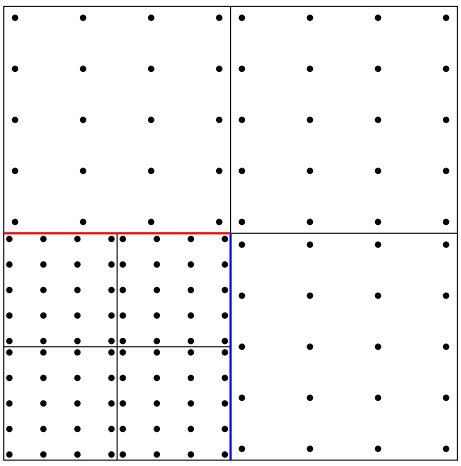


Figure 5.6: A grid of local Bernstein polynomial degrees of freedom. They have been shifted inwards, as otherwise degrees of freedom will overlap. Notice that on the border between the coarse and refined domain, the degrees of freedom only partially coincide.

Now, on any mesh element e, these THB-splines are in fact linearly independent. Any element of level l that does not intersect any refinement boundary $\partial\Omega_l \backslash \partial\Omega_0$, is obvious by the local Bernstein polynomial structure. For any element that does intersect a refinement boundary $\partial\Omega_l \backslash \partial\Omega_0$, the splines can be split whether they have support on this refinement boundary or not. Clearly, those without are linearly independent, and must be linearly independent to any spline with support on the boundary. The splines on the boundary, restricted to the boundary, must be Bernstein polynomials from level $l-1$ restricted to the boundary. They are thus linearly independent on the boundary, and hence on the element.

This means that we can do a Bezier projection from the local Bernstein polynomial space to this THB-space $\mathbb{F}$. As shown above, on any element e, the THB-splines of $\mathbb{F}$ are linearly independent, and we can thus do an element wise projection:

$$\vec{\lambda}^e = {C^e}^{-1} \vec{\beta}^e. \tag{5.31}$$

Where the columns of matrix C^e describe any basis function on element e of $f_j^e(x) \in \mathcal{F}$ as a linear combination of Bernstein polynomials $\mathfrak{b}_i^e(x)$. Secondly, this mapping is regularized as:

$$\lambda_j = \sum_{e \in E} \omega_j^e \lambda_j^e, \quad \omega_j^e := \frac{\int_{\Omega^e} f_j(x) dx}{\int_{\Omega} f_j(x) dx}. \tag{5.32}$$

Hence, we find an element $s \in \mathbb{F}$:

$$s(x) = \sum_j \lambda_j f_j(x). \tag{5.33}$$

This element can be mapped back to the local Bernstein polynomials on any element e by multiplying the coefficients by C^e. This is useful, as the actual THB projector T maps from the local Bernstein polynomial basis.

5.2.2. Estimates

We desire this projector to conform to the requirements of Assumption 4.

Proposition 5.5. *The projector $F : L^2(\Omega) \to \mathbb{F}$ conforms to the following properties for any $e \in E$. Let the coefficients $\vec{\beta}^e$ be the coefficients related to $\mathfrak{b}_i^e(x)$ that represent the projected coefficient on element e.*

$$Fs = s, \qquad \forall s \in \mathbb{F}, \qquad \textit{(continuous polynomial preserving)} \tag{5.34}$$

$$\|\vec{\beta}^e\|_\infty \leq \frac{C_{stab,F}}{|\Omega^e|^{1/2}} \|s\|_{L^2(\Omega^{\tilde{e}})}, \qquad \forall s \in L^2\left(\Omega^{\tilde{e}}\right). \qquad \textit{(local stability property)} \tag{5.35}$$

Where $C_{stab,F}$ is a constant independent on h. The support extension is taken in the $\mathbb{F}$ spline space.

Proof. The continuous polynomial preserving property is trivial. It clearly holds on the initial local elements projections, and in the regularization steps, for $s \in \mathbb{F}$, it must be continuous, and thus coincide on every element boundary. This means that the regularization step does not alter the initial local projection.

To show the local stability property:

$$\|\vec{\beta}^e\|_\infty \leq \frac{C_{stab,F}}{|\Omega^e|^{1/2}} \|f\|_{L^2(\Omega^{\tilde{e}})}, \quad \forall f \in L^2\left(\Omega^{\tilde{e}}\right). \tag{5.36}$$

We again start from Lemma 2.8, this bounds all the pre-regularized coefficients over all elements e, by:

$$\|\vec{\beta}^e\|_\infty \leq \frac{C_p}{|\Omega^e|^{1/2}} \|f\|_{L^2(\Omega^e)}. \tag{5.37}$$

In the regularization step, a Bezier projection is performed, where the local projection matrices $C^{e\,-1}$ of (5.31) must be independent of element size, possibly dependent on p and mesh regularity μ. But then, the regularized coefficients $\vec{\lambda}^e$ are bounded:

$$\|\vec{\lambda}^e\|_\infty \leq \frac{C}{|\Omega^e|^{1/2}} \|f\|_{L^2(\Omega^e)}. \tag{5.38}$$

Where this constant C is dependent on C_p, p and μ. Taking a weighted average, the resulting coefficients must be bounded by all of these λ^e and hence, the local stability property follows. □

5.3. Sub projection T

The second projector T is defined the exact same way is in the univariate setting (see section 4.3). In the multivariate case, we have more types of projection elements. However, every type is simply a collection of mesh elements, on which the THB-splines with support on the projection element, are linearly independent (see Lemma 5.2). Given some element e, all the THB-splines $b_i(x)$ can be written as a linear combination of Bernstein polynomials over e, denoted by $\mathfrak{b}_j^e$:

$$b_i(x) = \sum_j c_{i,j}^e \mathfrak{b}_j^e(x), \quad x \in \Omega^e. \tag{5.39}$$

We can extend this over any projection element $E_\epsilon = \{e_1, \dots, e_m\}$:

$$b_i(x) = \sum_{e \in E_\epsilon} \sum_j c_{i,j}^e \mathfrak{b}_j^e(x) = \sum_j c_{i,j}^\epsilon \mathfrak{b}_j^\epsilon(x), \quad x \in \Omega^\epsilon. \tag{5.40}$$

Then, locally on projection element ϵ, we project to the THB-spline basis with the pseudo inverse:

$$C^{\epsilon\,+} := \left(C^{\epsilon\,T}\, C^{\epsilon}\right) C^{\epsilon\,T}. \tag{5.41}$$

To obtain a local representation:

$$s(x)_{|\epsilon} \sum_i \gamma_i^{\epsilon} b_i(x), \quad x \in \Omega^{\epsilon}. \tag{5.42}$$

Finally, a recombination step produces the final projection for any THB-spline i:

$$\gamma_i = \sum_{\epsilon \in Y_j} \omega_i^{\epsilon} \gamma_i^{\epsilon}. \tag{5.43}$$

Where the weighting coefficients are given by:

$$\omega_i^{\epsilon} := \frac{\int_{\Omega^{\epsilon}} b_i(x)dx}{\int_{\Omega_0} b_i(x)dx}. \tag{5.44}$$

This produces a final THB-projection:

$$Ts = \sum_i \gamma_i b_i(x), \quad x \in \Omega_0, s \in \mathbb{F}. \tag{5.45}$$

Note, this is exactly the same as in Section 4.3.

5.4. Estimates

Defining the multivariate projector $\Pi := TF$. This projector has the following estimate result:

Theorem 5.6. *Let k and l be integer indices with $0 \leq k \leq l \leq p+1$. For each mesh element e, the following inequality holds:*

$$\|f - \Pi f\|_{H^k(\Omega^e)} \leq C_{app} h_e^{l-k} \|f\|_{H^l(\Omega^{e^*})}, \quad \forall f \in H^l\left(\Omega^{e^*}\right), \tag{5.46}$$

where h_e is the diameter of the element e. e^* is the support extension of the support extension of e. C_{int} is a constant independent of h_e, but possibly dependent on the shape regularity of the mesh, polynomial degree, number of refinement levels L, continuity, and the parameters k and l.

Proof. The proof and required lemmas from Theorem 4.10 also hold in the multivariate case.

□

6

Numerical Results

The proposed projector has been implemented in Matlab for both the univariate and multivariate case.

6.1. Univariate Projector

For the univariate projector, we have the following test cases:

$$f(x) = \sin(\pi x), \tag{6.1}$$
$$g(x) = 2\,|x - 1/2|\,, \tag{6.2}$$
$$h(x) = \cos(20\pi x), \tag{6.3}$$
$$z(x) = \sin(8\pi x) + 2. \tag{6.4}$$

6.1.1. Accuracy

The above mentioned test cases have been used to compare our proposed projector Π against a global THB-spline projection[1] and a different THB-spline projector introduced in (Giust et al., 2020). This was done for $p = 2$ in Figure 6.1 and for $p = 4$ in Figure 6.2. The first function $f(x)$ was chosen as a simple function to see how the projectors deal with the different levels of refinement. The second $g(x)$ was chosen to see how they would deal with C^0 smooth functions and $h(x)$ was chosen in such a way that the coarse splines will have difficulty capturing the finer details. In all cases, a THB-spline spaces was generated where the coarse spline space partitions the unit domain in sixteen elements, of which the middle eight have been refined.

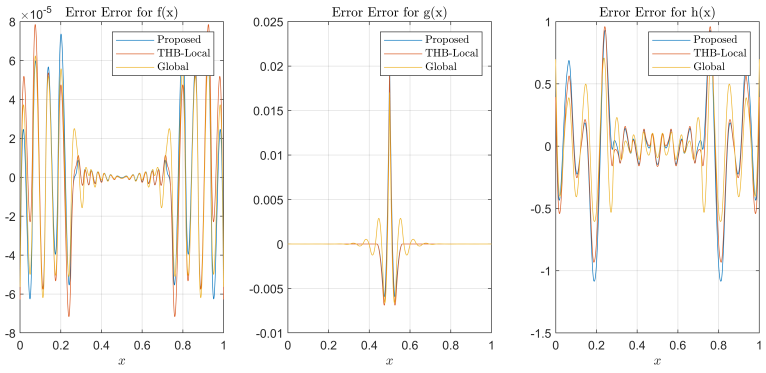


Figure 6.1: p=2, THB local is projector introduced in (Giust et al., 2020)

[1] The second step in Π, T has been replaced by a global THB projection, instead of the local projection element projections. This represents an "optimal" T projector.

In the case of $p = 2$ in Figure 6.1, there is not much difference between the proposed projector and the projector of (Giust et al., 2020). In the case of $p = 4$ in Figure 6.2, the differences become more pronounced. For the $sin(\pi x)$ case, the projector of (Giust et al., 2020) seems to struggle with the refinement boundaries at $x = \frac{1}{4}$ and $x = \frac{3}{4}$. In the case of $2\left|x - \frac{1}{2}\right|$, our proposed projector has a larger error at $x = \frac{1}{2}$, while the projector of (Giust et al., 2020) has a bigger overshoot. This indicates that neither one is better, but each excel for different problems.

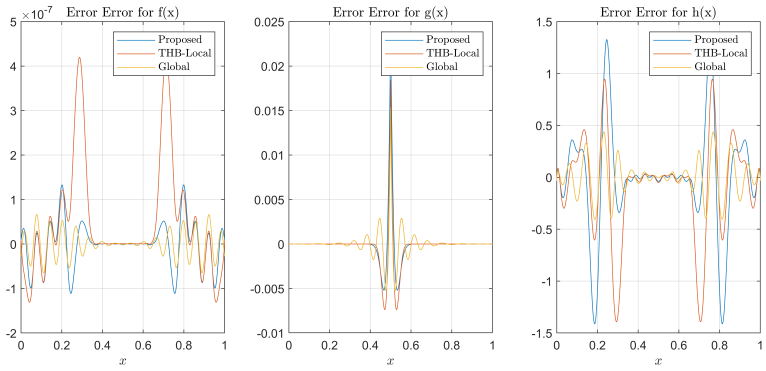


Figure 6.2: p=4, THB local is projector introduced in (Giust et al., 2020)

Additionally, convergence for both Π^0 and Π^1 are shown in Figure 6.3. To show this, we have a THB-spline space for which the middle is refined. The error is calculated for different element diameters h in order to check Theorem 4.10, which states that for $z(x) = \sin(8\pi x) + 2$, the projections should converge as $\mathcal{O}(h^{p+1})$ for Π^0. To see this, the theoretical slopes have also been plotted. For Π^1, no theoretical results on convergence are given, but the solution seems to converge as $\mathcal{O}(h^p)$.

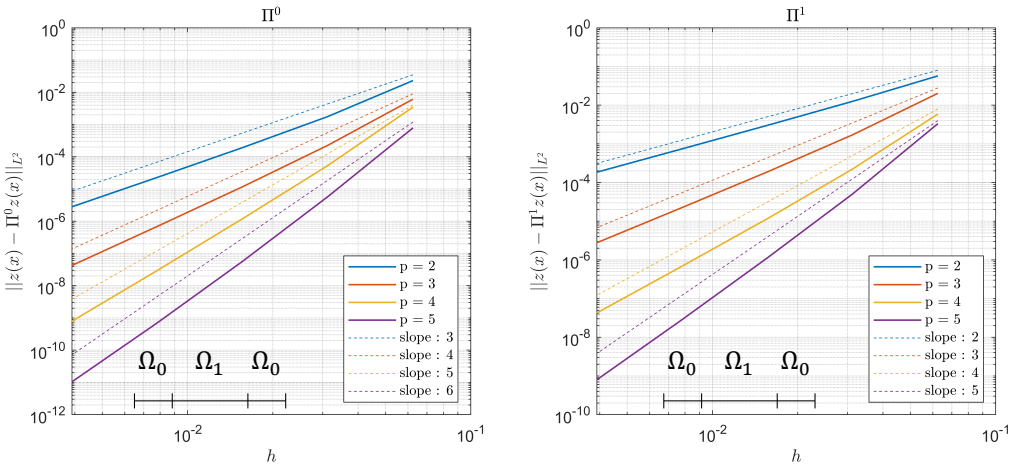


Figure 6.3: Convergence of the univariate projectors Π^0 and Π^1 for target function $z(x) = \sin(8\pi x) + 2$, over different element diameters and degrees. Here a THB-spline mesh is used where the middle half of the domain is refined. Additionally, the optimal convergence slopes are also plotted. Note that due to Theorem 4.10, only optimal convergence slopes for Π^0 are known, the slopes for Π^1 are the slopes that fit the best. Additionally, note that for Π^1, we project in to $\mathbb{T}^1$, but the degree denotes the degree of the THB-spline space $\mathbb{T}^0$.

6.1.2. Commutation

The commuting property of the projector has also been numerically confirmed in Figure 6.4. Here the function $f(x) = cos\,(4\pi x)$ is first projected on to the local Polynomial space $\mathbb{F}^0$ as is proposed in Section 4.2. In Figure 6.4, the error between $\frac{d}{dx}T^0F^0f(x)$ and $T^1\frac{d}{dx}F^0f(x)$ is shown. Only commutation

has been checked for T, as in F, the integrals have to be calculed analytically, thus the two to project function, differ by a constant. The commutation result is more involved for the projector T. The Figure shows that the second projector T^0 and T^1 commute with the derivative, and that the error is close to the order of machine precision for both the cases of $p = 2$ and $p = 4$. The not-exactly machine precision error is most likely a result of the numerical errors propagating through the projector T.

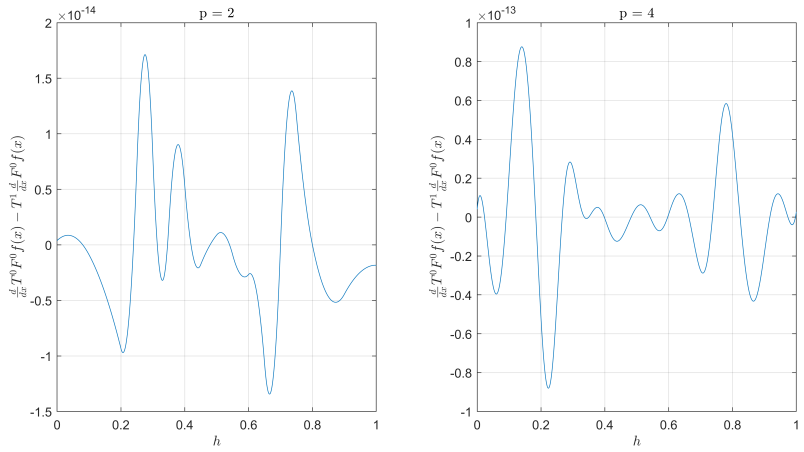


Figure 6.4: Given the projection of the function $f(x) = cos(4\pi x)$ in to the space $\mathbb{F}^0$ of (4.73). On the left the error between $\frac{d}{dx}T^0F^0f(x)$ and $T^1\frac{d}{dx}F^0f(x)$ for $p = 2$ and on the right for $p = 4$.

6.2. Multivariate projector

For the multivariate projector Π, we have the test case $f(x, y)$:

$$f(x, y) = \sin(\pi x)\sin(\pi y). \tag{6.5}$$

6.2.1. Convergence

In order to check convergence, a B-spline space will be used. Clearly, the L^2 error is mainly determined by the coarsest level. For this reason, analysing B-spline spaces, will give a better result. In Figure 6.5, the L^2 error for the function $f(x, y) = \sin(\pi x)\sin(\pi y)$ can be seen for various degrees $p = 1, \dots, 5$ over different element diameters h.

From Theorem 5.6, the error should converge as $\mathcal{O}(h^{p+1})$. In Figure 6.5, for $p = 1, \dots, 4$, the projection converges optimally. However, for $p = 5$, the accuracy of the projection halts, for $h = 6 \cdot 10^{-1}$. This is most likely the result of the numerical errors propagating through the projector.

Additionally, a similar Convergence plot has been made for THB-spline spaces for the function $g(x, y) = \sin(2\pi x^2)\sin(2\pi y^2)$. Due to the squares in the function, the domain $[0.5, 1]\times[0.5, 1]$ contains more detail and will benefit from refinement. The convergence of the error is given in Figure 6.6. In this Figure, it becomes clear that even for the THB-spline space, the convergence is optimal according to Theorem 5.6. But the optimal convergence rates require a minimal element diameter in order to converge properly.

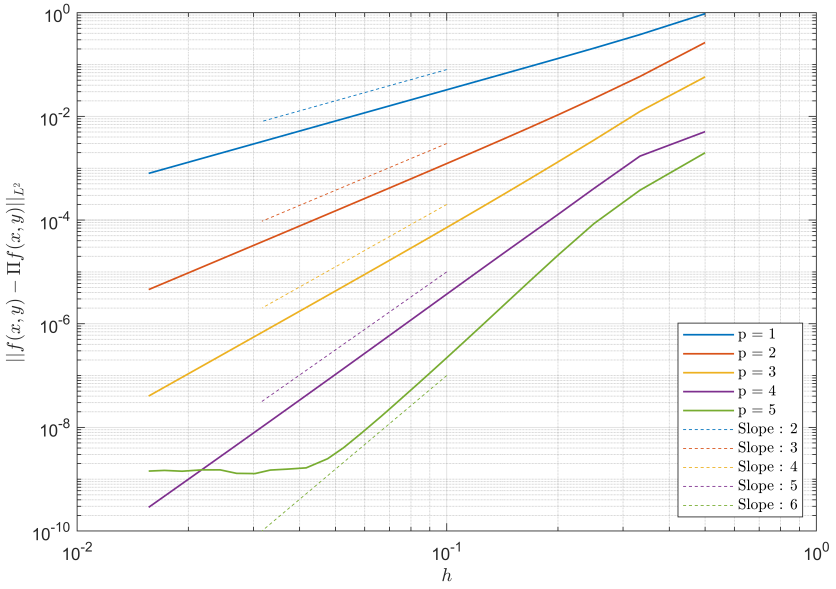


Figure 6.5: Projection error on to a B-spline space of the function $f(x,y) = \sin(\pi x)\sin(\pi y)$. For the degree values $p = 1, \dots, 5$, the error is calculated over different element diameters h. Additionally, the optimal $p+1$ slope lines are shown.

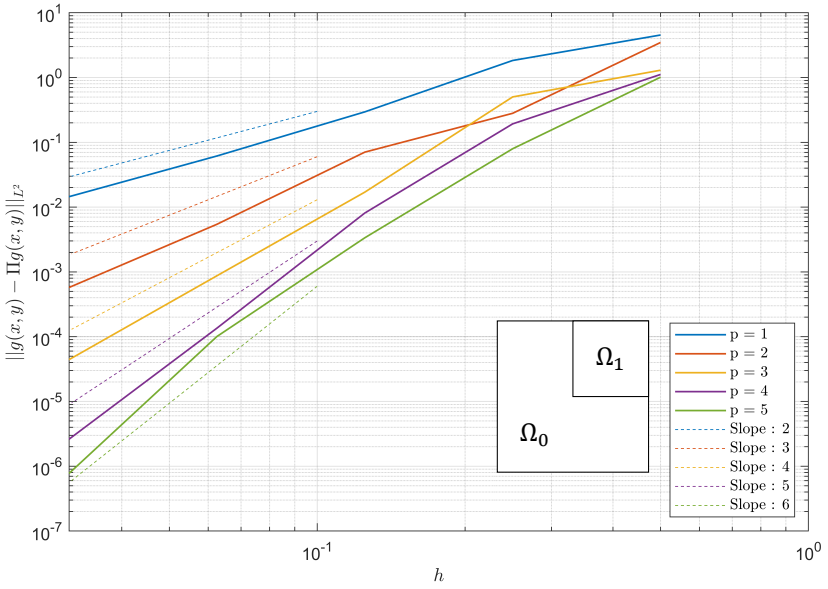


Figure 6.6: Projection error of $g(x,y) = \sin(2\pi x^2)\sin(2\pi y^2)$ in a THB-spline space. The mesh is refined for a single level on the domain $[0.5,1] \times [0.5,1]$, see the domain. The horizontal axis shows the element diameter h of the coarsest level. The error is given for $p = 1, \dots, 4$ and the optimal convergence slope lines $p+1$ are also shown.

7

Conclusion

Within the research field of Finite Element Exterior Calculus, that concerns it self with solving the abstract Hodge Laplacian problem numerically, one requires that the finite element complex has good approximation power, shares the underlying structure of the continuous complex, and the existence of a commuting projector. That is to say, an projector which commutes with the exterior derivative. THB-spline spaces are a natural choice for a finite element space/complex. They exhibit better approximation power per degree of freedom, compared to the regularly used linear basis functions. Additionally, they can be refined on subsets of the main domain to an arbitrary level or refinements. While local THB-spline projectors exist in literature, no commuting THB-spline projector exists.

Our proposed projector requires a local set of linearly independent THB-splines. Compared to B-splines, THB-splines can be overload on certain elements, which means that on certain mesh elements, the set of THB-spline basis functions with non-empty support are linearly dependent. To get around the overloaded elements, projection elements were introduced, that are collections of mesh elements for which we have shown that the set of THB-spline basis functions with non-empty support are linearly independent.

The local THB-spline projector Π consists of two sub-projectors, namely F and T. First, on every mesh element, the target function is projected on to a local Bernstein polynomial basis, which is then regularized to obtain a C^0 smooth projection. The second sub-projector is a Bezier projection over the projection elements. In this Thesis we have chosen the initial projector to be a L^2 projection on every element. However, this initial projection can be chosen, as long as they conform to Assumption 4.

Additionally, for the univariate setting, a set of commuting projectors was constructed. This was done by constructing a commuting set of sub-projectors. However, to show commutation for the second sub-projector, the intermediary space was required to be C^0 smooth, hence a regularization step. Unfortunately, this approach did not extend to the multivariate setting.

Lastly, the local estimates were proven for both the univariate and mutlivariate projectors. These estimates were numerically verified, as was the commutation property for the univarite case.

However, the projector does require various assumptions. Some of these appear in literature, like Assumption 1 and 2, which are used to show that THB-spline spaces are exact in 2D, or are necessary for other local THB-spline projectors. Additionally, we require that the THB-spline space is maximally smooth and that non of the projection elements overlap.

Bibliography

Arnold, D., & Guzmán, J. (2021). Local L 2-bounded commuting projections in FEEC. *ESAIM: Mathematical Modelling and Numerical Analysis*, *55*(5), 2169–2184. https://doi.org/10.1051/m2an/2021054

Arnold, D. N. (2018). *Finite Element Exterior Calculus*. https://doi.org/https://doi.org/10.1137/1.9781611975543

Arnold, D. N., Falk, R. S., & Winther, R. (2010). Finite Element Exterior Calculus: From Hodge. *Bulletin of the American Mathematical Society*, *47*(2), 281–354. http://www.ams.org/journal-getitem?pii=S0273-0979-10-01278-4

Bazilevs, Y., Beirão Da Veiga, L., Cottrell, J. A., Hughes, T. J., & Sangalli, G. (2006). Isogeometric analysis: Approximation, stability and error estimates for h-refined meshes. *Mathematical Models and Methods in Applied Sciences*, *16*(7), 1031–1090. https://doi.org/10.1142/S0218202506001455

Beirão da Veiga, L., Buffa, A., Rivas, J., & Sangalli, G. (2011). Some estimates for h-p-k-refinement in Isogeometric Analysis. *Numerische Mathematik*, *118*(2), 271–305. https://doi.org/10.1007/s00211-010-0338-z

Bernardi, C., Dauge, M., Maday, Y., Bernardi, C., Dauge, M., & Maday, Y. (2007). Polynomials in the Sobolev World To cite this version : HAL Id : hal-00153795.

Buffa, A., Rivas, J., Sangalli, G., & Vázquez, R. (2011). Isogeometric Discrete Differential Forms in Three Dimensions. *SIAM Journal on Numerical Analysis*, *49*(2), 818–844. https://doi.org/10.1137/100786708

Buffa, A., Sangalli, G., & Vázquez, R. (2014). Isogeometric methods for computational electromagnetics: B-spline and T-spline discretizations. *Journal of Computational Physics*, *257*(PB), 1291–1320. https://doi.org/10.1016/j.jcp.2013.08.015

Cottrell, J. A., Hughes, T. J. R., & Bazilevs, Y. (2009). *Isogeometric analysis*. Wiley-Blackwell.

Evans, J. A., Scott, M. A., Shepherd, K. M., Thomas, D. C., & Vázquez Hernández, R. (2020). Hierarchical B-spline complexes of discrete differential forms. *IMA Journal of Numerical Analysis*, *40*(1), 422–473. https://doi.org/10.1093/IMANUM/DRY077

FEAP. (n.d.). http://projects.ce.berkeley.edu/feap/

Giust, A., Jüttler, B., & Mantzaflaris, A. (2020). Local (T)HB-spline projectors via restricted hierarchical spline fitting. *Computer Aided Geometric Design*, *80*, 101865. https://doi.org/10.1016/J.CAGD.2020.101865

Hughes, T. J. R., Cottrell, J. A., & Bazilevs, Y. (2005). Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, *194*(39-41), 4135–4195. https://doi.org/10.1016/j.cma.2004.10.008

Johannessen, K. A., Kumar, M., & Kvamsdal, T. (2015). Divergence-conforming discretization for Stokes problem on locally refined meshes using LR B-splines. *Computer Methods in Applied Mechanics and Engineering*, *293*(December 2017), 38–70. https://doi.org/10.1016/j.cma.2015.03.028

Jüttler, B. (1998). The dual basis functions for the Bernstein polynomials. *Advances in Computational Mathematics*, *8*(4), 345–352. https://doi.org/10.1023/a:1018912801267

Lyche, T., Manni, C., & Speleers, H. (2018). *Foundations of spline theory: B-splines, spline approximation, and hierarchical refinement* (Vol. 2219). https://doi.org/10.1007/978-3-319-94911-6{_}1

Sande, E., Manni, C., & Speleers, H. (2020). Explicit error estimates for spline approximation of arbitrary smoothness in isogeometric analysis. *Numerische Mathematik*, *144*(4), 889–929. https://doi.org/10.1007/s00211-019-01097-9

Thomas, D. C., Scott, M. A., Evans, J. A., Tew, K., & Evans, E. J. (2015). 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. *Computer Methods in Applied Mechanics and Engineering*, *284*, 55–105. https://doi.org/10.1016/j.cma.2014.07.014