



TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

**NON-UNIFORM DEGREE AND
TCHEBYCHEFFIAN SPLINE TECHNOLOGIES
FOR ADAPTIVE ISOGEOMETRIC ANALYSIS**

Krunal Raval

Supervisors:

Prof. Carla Manni

Prof. Hendrik Speleers

**NON-UNIFORM DEGREE AND
TCHEBYCHEFFIAN SPLINE TECHNOLOGIES
FOR ADAPTIVE ISOGEOMETRIC ANALYSIS**

by

Krunal Raval

DISSERTATION

Presented to the Department of Mathematics of

The University of Rome ‘Tor Vergata’

for the Degree of

DOCTOR OF PHILOSOPHY

Supervisors:

Prof. Carla Manni

Prof. Hendrik Speleers

Defense committee:

Prof. Tom Lyche

Prof. Maria Lucia Sampoli

Prof. Carla Manni

June 2024

Abstract

Tchebycheffian splines are smooth piecewise functions whose pieces are drawn from (possibly different) Tchebycheff spaces, a natural generalization of algebraic polynomial spaces. They enjoy most of the properties known in the polynomial spline case. In particular, under suitable assumptions, Tchebycheffian splines admit a representation in terms of basis functions, called Tchebycheffian B-splines (TB-splines), completely analogous to polynomial B-splines. A particularly interesting subclass consists of Tchebycheffian splines with pieces belonging to null-spaces of constant-coefficient linear differential operators. They grant the freedom of combining polynomials with exponential and trigonometric functions with any number of individual shape parameters. Moreover, they have been recently equipped with efficient evaluation and manipulation procedures. We consider the use of TB-splines with pieces belonging to null-spaces of constant-coefficient linear differential operators as an attractive substitute for standard polynomial B-splines and rational NURBS in isogeometric Galerkin methods. We discuss how to exploit the large flexibility of the geometrical and analytical features of the underlying Tchebycheff spaces according to problem-driven selection strategies. TB-splines offer a wide and robust environment for the isogeometric paradigm beyond the limits of the rational NURBS model.

Locally Refined B-splines (LR B-splines) have proven to be a very flexible and powerful framework in several application areas, such as approximation and data reconstruction, computer aided design, and isogeometric analysis.

We introduce Locally Refined Tchebycheffian B-splines (LR TB-splines) as a generalization of LR B-splines from the algebraic polynomial setting to the broad Tchebycheffian setting. We focus again on the particularly interesting class of Tchebycheffian splines whose pieces belong to null-spaces of constant-coefficient linear differential operators. We consider their application in the context of isogeometric analysis and discuss related adaptive refinement, adopting the so-called structured mesh refinement strategy, widely used and analyzed in the classical polynomial case.

Finally, we extend LR B-splines to allow for non-uniform polynomial degree. This capability, combined with local h -refinement, also allows for local hp -refinement, that is, increasing the polynomial degree as well as the smoothness at newly introduced knot lines. The novel refinement

schemes are based on degree elevation of individual basis functions by means of certain two-scale relations, and lead to nested spaces. We apply the introduced spaces to several two-dimensional model problems, illustrating the efficacy of the proposed adaptive refinement methodology in the context of isogeometric analysis.

To my parents and my brother, Devang.

Acknowledgements

I would like to express my warmest gratitude to my supervisors, Carla Manni and Hendrik Speleers. Their guidance and mentorship have not only ensured the success of my PhD but have also transformed the experience into a journey of profound personal growth. It is thanks to their unwavering support that my journey through my PhD was remarkably pleasant and enjoyable. They have taught me exceptionally high standards of research and introduced me to the world-class scientific exploration.

I extend my thanks to Tor Dokken, Georg Muntingh, and Panagiotis Kaklis for graciously hosting me at your institute during the secondments. Special appreciation goes to Tor Dokken for his guidance into the complexities of local refinement and for facilitating an engaging collaboration with Kjetil André Johannessen and René Hiemstra.

Furthermore, I am deeply grateful to my colleagues, who have become dear friends: Artūrs Bērziņš, Michelangelo Marsala, and Salah Eddargani. Their friendship has made conferences and other scientific activities not only productive but also incredibly enjoyable.

Lastly, but certainly not least, I wish to thank my parents and my brother, Devang, for their unwavering support. Their presence in my life has profoundly shaped me into the person I am today, and for that, I am forever grateful.



This thesis was funded by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 860843.

Marie Skłodowska-Curie
Actions

Contents

Abstract	iii
Acknowledgements	vii
Introduction	1
State of the art and contributions	3
Tchebycheffian B-splines	3
Splines over T-meshes	6
Tchebycheffian splines over T-meshes	9
Outline of the thesis	11
Publications	12
1 Preliminaries	13
1.1 Polynomial spline spaces	13
1.1.1 Polynomial B-splines	14
1.1.2 Local representation of polynomial B-splines	18
1.1.3 Knot insertion for polynomial B-splines	20
1.1.4 Degree elevation of polynomial B-splines	22
1.1.5 Tensor-product B-splines	23
1.2 Isogeometric Galerkin methods	24
2 A primer on Tchebycheffian splines	29
2.1 Extended Tchebycheff spaces	29
2.1.1 Extended complete Tchebycheff spaces	30
2.1.2 Null-spaces of linear differential operators	33
2.2 Tchebycheffian spline spaces	36
2.2.1 Tchebycheffian B-splines	37
2.2.2 Approximation properties of Tchebycheffian splines .	47
2.3 Multi-degree Tchebycheffian spline spaces	48
3 Isogeometric analysis with Tchebycheffian B-splines	53
3.1 Isogeometric TB-spline Galerkin methods	53
3.2 Selection of the ECT-spaces	54
3.3 Numerical results of IgA with TB-splines	56
3.3.1 Case study 1: Poisson problem on a domain bounded by circular arcs	57

3.3.2	Case study 2: Poisson problem on a symmetric domain bounded by circular arcs	59
3.3.3	Case study 3: Advection–diffusion problem with radial advection flow	62
3.3.4	Case study 4: Advection–diffusion problem with tangential advection flow	65
3.3.5	Case study 5: Advection–diffusion problem with internal sharp layer on a square	68
3.4	Conclusive remarks	72
4	Locally refined Tchebycheffian B-splines	75
4.1	Local representation of TB-splines	76
4.1.1	Knot insertion for TB-splines	77
4.1.2	Multivariate setting for TB-splines	79
4.2	TB-splines on LR-meshes	80
4.2.1	LR-meshes	80
4.2.2	LR TB-splines	84
4.2.3	Constructive example of LR TB-spline	88
4.3	Isogeometric analysis with LR TB-splines	97
4.3.1	Adaptive strategy	98
4.4	Numerical results of adaptive IgA with LR TB-splines	101
4.4.1	Case study 6: Poisson problem on a unit square	102
4.4.2	Case study 7: Reaction-diffusion problem on a square	104
4.4.3	Case study 8: Advection–diffusion problem with internal sharp layer on a square	108
4.5	Conclusive remarks	109
5	Non-uniform degree LR-splines	113
5.1	Non-uniform degree B-splines on LR-meshes	114
5.1.1	<i>hp</i> -refinement algorithm for LR B-splines	114
5.1.2	Constructive example of <i>hp</i> -refinement	117
5.2	Data-structures	121
5.3	Numerical results of adaptive IgA with non-uniform degree LR B-splines	122
5.3.1	Case study 9: Smooth Poisson problem on a unit square	125
5.3.2	Case study 10: Advection–diffusion problem with internal sharp layer on a square	128
5.4	Conclusive remarks	134
Outlook		135
Bibliography		138

Introduction

Isogeometric Analysis (IgA) [39, 55] emerged about two decades ago as a methodology aimed at streamlining the interactions between geometric modeling and numerical simulation. It sought to address significant discrepancies between Computer-Aided Design (CAD) packages, used for modeling geometries, and Finite Element Analysis (FEA) software developed in a complete independent way for numerical simulation. Indeed, traditionally geometric modeling relied on highly smooth piecewise polynomial functions known as splines, whereas numerical simulations utilized C^0 finite elements. By bringing FEA and CAD together, IgA enhanced efficiency in solving complex engineering problems, providing a truly integrated design-through-analysis process. The idea behind IgA revolves around employing an isoparametric approach, utilizing the same kind of functions both to represent the geometry of the domain and to approximate the solution of the differential problem defined on it. This approach yields several advantages over classical FEA:

- Enhanced accuracy in describing the geometry: It provides a better approximation of complex geometries, and even achieves exact representation in the case of conic sections; leading to increased precision in analysis.
- Streamlined refinement process: With IgA, the description of geometry is exact (or well approximated) at the coarsest mesh level and it does not change with refinement; eliminating the necessity of communication with CAD systems during refinement.
- Higher accuracy per degree of freedom: For a given degree, spline functions of high and low smoothness provide spaces with the same approximation order but of different dimensions: with a clear gain for the former ones, [21, 101]. This rejuvenated the study of higher order methods, proving to be a superior alternative in various applications; see [55, 57], and references therein.

These advantages have attracted significant interest in the isogeometric paradigm across various applications, see [3, 29, 43, 57, 58], validating the claim that IgA is *a viable alternative to standard, polynomial based finite element analysis*, [55].

The initial classical formulation of IgA is based on tensor-product B-splines and their rational extension (NURBS), [12, 94]. As a consequence, it inherited some drawbacks and limitations of these tools. For instance, NURBS are a legacy of CAD industry where they are of main interest because they allow for exact representation of conic sections. Still NURBS lack an exact description of transcendental curves of interest in application, and their parametrization of conic sections does not correspond to –and it is often far from– the arc length. In addition, they are defined in a rational form along with “weights”, resulting into a very poor behavior under differentiation and integration, which are key operations in analysis.

However, the properties of B-splines and NURBS which are crucial for design (positivity, partition of unity, compact support,...) are not confined to the polynomial setting, and can be extended in an elegant way beyond piecewise polynomial and/or rational functions.

In this thesis, we explore the use in IgA of a generalization of the polynomial splines, called Tchebycheffian splines, both to overcome the drawbacks of the rational model and to exploit their superior description of the fundamental solutions of some differential operators.

Another limitation in the primitive formulation of IgA pertains to local refinement in the general multivariate setting and it is due to the tensor-product structures, typically adopted to generate multivariate splines from univariate spline basis functions. Tensor-product spline spaces can be refined through knot insertion and/or degree elevation in each coordinate direction separately. However, this often results in excessive refinement in areas that do not require it due to the global nature of the refinement process. This lack of locality in refinement of tensor-product splines is visualized in Figure 1.

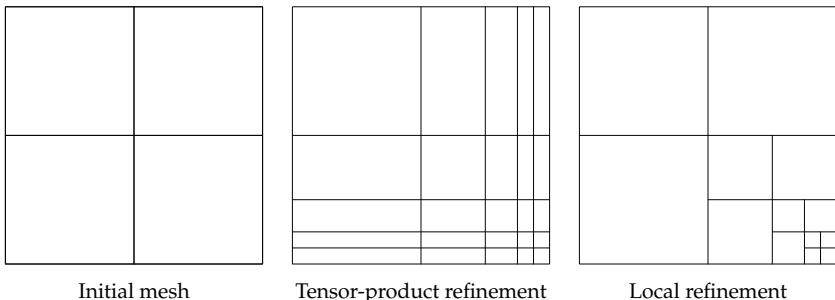


Figure 1: Lack of local refinement in tensor-product splines.

To address the limitations associated with local refinement, various strategies have been devised. Here we focus on local tensor-product struc-

tures and, in particular, on Locally Refined (LR) meshes and on the construction of different spline spaces over such meshes. Leveraging the structural similarities between standard polynomial B-splines and Tchebycheffian B-splines, we can construct Tchebycheffian spline spaces over LR meshes, combining the benefits inherent to Tchebycheffian spline setting and to local tensor-product structures. This is the second line of investigation presented in this thesis.

Furthermore, when tackling higher-order problems with localized features, integrating higher degree B-splines into the solution space can be advantageous. However, incorporating local degree elevation in B-splines presents a challenge, as uniformly higher degree splines throughout the domain may be unnecessary. In such scenarios, LR meshes with non-uniform degree B-splines, constructed through a combination of local h - and p -refinement, could offer a more efficient solution. This is a third line of investigation carried out in this thesis.

State of the art and contributions

In this section we shortly summarize the state of the art in the fields of (1) Tchebycheffian (B-)splines as a generalization of the standard polynomial (B-)splines and (2) spline constructions on unstructured meshes with local tensor-product structure for adaptive refinement, and we sum up our contributions on the use of Tchebycheffian B-splines and non-uniform degree splines in isogeometric (adaptive) Galerkin methods.

Tchebycheffian B-splines

Spline functions are ubiquitous in numerical methods. Besides their theoretical interest, they have application in several branches of the sciences including geometric modeling, signal processing, data analysis, visualization and numerical simulation just to mention a few. Splines, in the broad sense of the term, are functions consisting of pieces of smooth functions glued together in a certain smooth way. There is a large variety of spline species, often referred to as the zoo of splines. The most popular species is the one where the pieces are algebraic polynomials of a given degree p and inter-smoothness is imposed by means of equality of derivatives up to a certain order. Their popularity can be mainly attributed to their representation in terms of the so-called B-splines. B-splines enjoy properties such as local linear independence, minimal support, non-negativity, and partition of unity; they can be computed through a stable recurrence relation and can even be seen as the geometrically optimal basis for piecewise polynomial spaces.

Tchebycheffian splines are smooth piecewise functions whose pieces are drawn from (possibly different) Extended Tchebycheff (ET-) spaces which are natural generalizations of algebraic polynomial spaces [64, 76, 87, 104]. Any non-trivial element of an ET-space of dimension $p + 1$ has at most p zeros counting multiplicity. We will refer to p as the degree, in analogy with the polynomial case. Extended Complete Tchebycheff (ECT-) spaces are an important subclass that can be generated through a set of positive weight functions [86, 104] and are spanned by generalized power functions [76]; the latter are the natural extension of the monomial basis functions for algebraic polynomials; see Section 2.1.1. Relevant examples are null-spaces of linear differential operators on suitable intervals [104]; see Section 2.1.2. On bounded and closed intervals the concepts of ET-space and ECT-space coincide [85], so from now on we will focus on ECT-spaces.

Most of the results known for splines in the polynomial case extend in a natural way to the Tchebycheffian setting. In particular, under suitable assumptions on the involved ECT-spaces, Tchebycheffian splines admit a representation in terms of basis functions, called Tchebycheffian B-splines (TB-splines), with similar properties to polynomial B-splines. TB-splines were introduced in 1968 by Karlin [63] using generalized divided differences. There are several other ways to define them, including Hermite interpolation [24, 90], de Boor-like recurrence relations [42, 74], integral recurrence relations [11, 76], and blossoming [51, 87]; see also the historical notes in [104, Chapters 9 and 11] for further details. Each of these definitions has advantages according to the properties to be proved and lead to the same functions, up to a proper scaling.

Multivariate extensions of Tchebycheffian splines can be easily obtained by means of tensor-product structures.

Due to the richness of ECT-spaces, Tchebycheffian splines can find applications in several contexts including data approximation/interpolation, geometric modeling and numerical simulation; see [76] and references therein. Thanks to their structural similarities, TB-splines are theoretically plug-to-plug compatible with classical polynomial B-splines, so they can be potentially easily incorporated in any software library supporting polynomial B-splines to enrich its capability.

Unfortunately, despite their theoretical interest and applicative potential, TB-splines have not gained much attention in practice so far. The reason behind this is that TB-splines are generally difficult to compute. The classical approaches mentioned above, based on generalized divided differences, Hermite interpolation, or repeated integration, are computationally expensive and/or numerically unstable. An important step forward was recently made in [54] where the authors proposed a strategy that represents TB-splines as linear combinations of local Tchebycheffian Bernstein

functions through a suitable extraction operator. The local Tchebycheffian Bernstein functions form a basis of the local ECT-spaces involved in the definition of the TB-splines. In the polynomial case, these are nothing but the classical Bernstein polynomial basis functions. Following the approach in [54], an object-oriented Matlab toolbox has been developed in [115] for the construction and manipulation of TB-splines whenever they exist. The toolbox supports TB-splines whose pieces belong to ECT-spaces that are null-spaces of constant-coefficient linear differential operators, and is publicly available. Note that both [54] and [115] address the more general setting of Multi-Degree TB-splines (MDTB-splines) where the local ECT-spaces of the Tchebycheffian spline space are not required to be of the same dimension. Having at our disposal a publicly available implementation for such a large class of TB-splines paves the path for their effective use in practical applications. An alternative strategy has been proposed in [8] but no implementation is available.

As mentioned before, the advantages of the isogeometric paradigm are not a distinguishing property of B-splines/NURBS, and B-splines/NURBS are not a requisite ingredient in IgA. An interesting subclass of TB-splines, the so-called Generalized B-splines (GB-splines), has been proposed as an alternative to classical polynomial and rational splines in the context of IgA; see [78, 81, 82] and references therein. GB-splines can be seen as the minimal extension of polynomial B-splines towards the wide variety of TB-splines: their pieces belong to ECT-spaces obtained by enriching an algebraic polynomial space with a pair of functions, typically hyperbolic or trigonometric functions identified by a single shape parameter. GB-splines overcome two issues of the rational NURBS model: they allow for an (almost) exact arc-length parameterization of conic sections and behave with respect to differentiation and integration as nicely as polynomial B-splines (for instance, the derivative of a trigonometric generalized spline with a given phase parameter β and degree p is again a trigonometric generalized spline with the same shape parameter β and degree $p - 1$). These properties make GB-splines an interesting tool to face both geometrical and analytical hurdles in IgA. The effectiveness of hyperbolic or trigonometric GB-splines in isogeometric Galerkin and collocation methods has been illustrated in a sequence of papers, where their properties have been exploited to obtain exact representations of common geometries [1, 78, 80] or to beneficially deal with advection-dominated problems [79, 81]; see also [32, 99] for the related spectral properties.

Being a so minimal extension of the polynomial setting, however, GB-splines are not always flexible enough for practical applications. In particular, in a tensor-product GB-spline space only two hyperbolic or trigonometric functions identified by the same shape parameter are added to polynomials along each parametric direction. Therefore, a given tensor-

product GB-spline space does not allow for:

- an exact representation of different arcs of conic sections at opposite sides;
- a proper treatment of different analytic features (like sharp layers) along a given parametric direction;
- a simultaneous treatment of geometrical and analytical features along the same parametric direction.

Tchebycheffian splines with pieces belonging to ECT-spaces that are null-spaces of constant-coefficient linear differential operators are an extension of hyperbolic and trigonometric generalized polynomial splines. They enjoy all the geometrical and analytical features that motivate the interest in GB-splines without suffering from the above mentioned restrictions. They grant the freedom of combining polynomials with exponential and trigonometric functions with any number of individual shape parameters. For this class of Tchebycheffian splines, when the various pieces are drawn from a single ECT-space which contains constants, the existence of TB-splines is always ensured, possibly with some restriction on the partition; see Chapter 2 for more details. Furthermore, they are supported by the Matlab toolbox available in [115]. In summary, TB-splines with pieces belonging to ECT-spaces that are null-spaces of constant-coefficient linear differential operators offer a suitable balance between the immense variety of ECT-spaces and the practical needs of a problem-driven space selection and efficient evaluation procedures for the space elements.

In Chapter 3 of this thesis we investigate the use of tensor-product TB-splines, whose pieces belong to ECT-spaces that are null-spaces of constant-coefficient linear differential operators, in Galerkin isogeometric methods. It turns out that such a class of TB-splines provides a powerful and flexible environment for the IgA paradigm, beyond the limits of the rational NURBS model.

Splines over T-meshes

Tensor-product B-splines are probably the most well-known multivariate spline basis functions. They have been profitably applied in different contexts including geometric modeling, approximation theory, and numerical simulation. Their popularity roots in their simple, elegant and efficient construction: they are nothing but tensor-products of univariate B-splines; see, e.g., [75, 98] and references therein.

The tensor structure of the underlying mesh, however, is the major weakness of tensor-product B-splines as it hinders adequate local refinement, forcing the use of unnecessarily large discrete spaces and leading to

a significant loss in efficiency. This has been seen as a severe limitation in the context of isogeometric analysis.

To overcome this limitation, in the last decades many alternative spline technologies have been developed for so-called T-meshes. Such meshes are still axis-aligned but T-vertices are allowed in the interior of the domain, in order to support local refinement, while preserving locally the simplicity of the tensor approach; see [105] and references therein. T-splines [109], (truncated) hierarchical B-splines [49], and locally refined B-splines [41] are popular examples of such spline technologies. All these approaches have their own strengths (and weaknesses) depending on the context they are intended to be used.

T-splines were introduced by Sederberg [108, 109] to enable local refinement of spline surfaces and handle complex topology, by treatment of so called extraordinary points in the context of CAD. Initially, T-splines were limited to bi-cubic polynomial degree and C^2 continuity between elements. However, the technology has since been extended to arbitrary uniform polynomial degrees [35] and supports mixed continuity [25]. Various applications of T-splines in isogeometric analysis have been reported [4, 43]. To ensure nested spaces and maintain linear independence, a restricted class of analysis-suitable T-splines was introduced [26, 68, 70, 106]. Subsequently, these were characterized as dual-compatible T-splines [9, 10]. Several local refinement algorithms for analysis-suitable T-splines with optimal convergence have been proposed in [59, 69].

Hierarchical B-splines were introduced in 1988 for local h -refinement in geometric modeling [46]. The construction of the hierarchical B-splines guarantees nested spaces and linear independence of the basis functions [66]. The application of hierarchical constructions in isogeometric analysis has been very promising [45, 83, 102, 123]. However, the standard hierarchical construction does not preserve the partition of unity property. To address this limitation, a new set of hierarchical basis functions, called Truncated Hierarchical B-splines (THB-splines), were introduced [49]. THB-splines, defined as a linear combinations of refined B-splines, form a convex partition of unity, exhibit good stability and approximation properties [47, 50, 116]. By providing a way to define an adaptive extension of the B-spline framework which is also suitable for geometric modeling applications, THB-splines satisfy both the demands of adaptive numerical simulation and geometric design, making them well suited for isogeometric analysis; see [48, 52, 53, 73, 123]. An comprehensive study has been conducted to develop a posteriori error estimators for devising optimal refinement strategies with hierarchical B-splines; see [27, 28, 67, 113].

The definition of LR B-splines is inspired by the knot insertion refinement process of univariate (and tensor-product) B-splines. Their formulation bears a large similarity to classical tensor-product B-splines and this

makes them one of the most elegant extensions of univariate B-splines towards T-meshes. Since their introduction in [41], LR B-splines have found interesting applications in several contexts ranging from data approximation [65, 112] to numerical simulations [60, 61], also considering their rational version [128]. More theoretical aspects, mainly related to the issue of linear and local linear independence and related adaptive refinement strategies, have been investigated in [19, 91, 92, 93]. A comparison between LR B-splines, hierarchical, and truncated hierarchical B-splines can be found in [62], while combinations of the LR B-spline framework with the hierarchical approach have been explored in [20, 119].

LR B-spline refinement starts from a multivariate tensor-product spline space spanned by tensor-product B-splines. The refinement is performed successively by splitting individual tensor-product B-splines in one parameter direction at a time. In the bivariate case the splitting is done by introducing an axis parallel line segment with multiplicity. The line segment must split the support of at least one tensor-product B-spline into two disjoint parts. Very often the line segment also splits the support of other tensor-product B-splines. The refinement process is continued until all remaining B-splines have a certain minimal support property as defined in [19, 41, 93].

Figure 2 depicts examples of a tensor-mesh, a T-mesh that is not an LR-mesh, a T-mesh that is also an LR-mesh, and a mesh that is neither a T-mesh nor an LR-mesh.

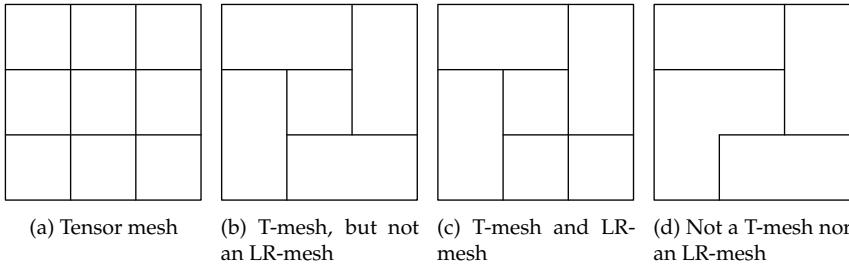


Figure 2: The T-mesh in (b) can not be created from single line insertions starting from a tensor mesh. The T-mesh in (c) can, and is therefore also a valid LR-mesh.

While T-splines have a straightforward construction only for odd degrees, hierarchical B-splines and LR B-splines offer independence from the polynomial degree. Moreover, both the definition and implementation of hierarchical B-splines and LR B-splines can be made dimension-independent.

The notion of mixing multiple degrees in (univariate) spline constructions was introduced first in the context of approximation [90] and later

rejuvenated in geometric modelling [110]. Such an extension of uniform degree B-splines, allowing for spaces of different dimensions in different intervals glued together with certain smoothness, are called Multi-Degree B-splines (MDB-splines). More recently, an efficient algorithm for the computation of MDB-splines was proposed in [121] and further developed in [114, 122]. The potential of the multi-degree spline spaces was also actualized in applications [118]. Some alternative constructions to the multi-degree B-splines can be found in [7, 13, 111].

The construction of multivariate spline generalizations combining locally different degrees, with the same flexibility as MDB-splines, is a hard and ambitious task. Already in the early work on hierarchical splines it was noted that different degrees could be mixed in the (truncated) hierarchical framework; see [50, Remark 4]. A full extension of the (truncated) hierarchical setting towards local mesh refinement and degree elevation is described in [44]. Even though not supported by the standard definition of T-splines, their flexibility can be increased through local p -refinement by introducing weighted B-splines of hybrid degrees [71]. Additionally, the study of the effectiveness of k -refinement [40] and its comparison with p -refinement [57] in isogeometric analysis has garnered considerable attention for its capability to address high-order differential problems accompanied by hp -refined meshes.

In the context of LR splines, the construction of non-uniform degrees utilizes the degree elevation operation or p -refinement, in which the degree of a spline is increased, while preserving its smoothness [95], in contrast to the k -refinement where the maximal smoothness is maintained at the loss of nestedness of the space. With the local tensor-product structure of LR B-splines, it becomes feasible to perform degree elevation on an individual B-spline without introducing new degrees of freedom along an entire hyper-plane in the parameter domain. This capability allows for local degree elevation, enabling the construction of a solution space with non-uniform degrees, which proves advantageous in addressing higher-order isogeometric methods involving local features. Moreover, due to the two-scale relation, low degree functions are consistently replaced with new higher degree ones, ensuring the nestedness of the space.

In this thesis, see Chapter 5, we introduce non-uniform degree LR-splines that allow local adaptive h -refinement as well as local p - and hp -refinement, by combining local knot insertion with local degree elevation, and we discuss their application in isogeometric analysis.

Tchebycheffian splines over T-meshes

As already mentioned, most of the results known for univariate polynomial splines extend in a natural way to the Tchebycheffian setting. As

illustrated in the analysis presented in Chapter 3, see also [97], tensor-product TB-splines are able to outperform tensor-product polynomial B-splines in isogeometric Galerkin methods whenever appropriate problem-driven selection strategies for the underlying ECT-spaces are applied. Nevertheless, tensor-product TB-splines present the same drawbacks of any tensor-product structure when local refinement is of need.

The structural similarity between ECT-spaces and algebraic polynomial spaces enables us to extend popular local refinement technologies, based on local tensor products, towards the Tchebycheffian setting. Tchebycheffian spline spaces over T-meshes have been introduced in their full generality in [17]. The structure of ECT-spaces has been exploited in [18] to fully extend the dimension study carried out in the polynomial case in [89]. Some earlier generalizations of the polynomial setting towards particular Tchebycheffian spline spaces or peculiar T-meshes have been considered in [14, 15, 16, 80]. In particular, [16] outlines the construction of GB-splines on LR-meshes while hierarchical GB-splines have been presented in [80].

In Chapter 4 of this thesis, we define LR TB-splines as a generalization of LR B-splines and we analyze their performance in the context of adaptive isogeometric Galerkin methods.

The definition of LR TB-splines is driven by the knot insertion refinement process of tensor-product TB-splines, in complete analogy to the polynomial setting. In the bivariate tensor case, inserting a new knot in a pair of (global) knot vectors results in inserting a line segment in the mesh crossing the entire domain, thus refining all the TB-splines whose supports are crossed. On the contrary, LR TB-splines are defined on local knot vectors, and consequently the insertion of a new knot is always performed with respect to a particular LR TB-spline and results in refining only few basis functions.

The theoretical construction of LR TB-splines is independent of the particular ECT-spaces where the various pieces are drawn from. However, in the applicative context we confine ourselves to Tchebycheffian splines identified by ECT-spaces that are null-spaces of constant-coefficient linear differential operators containing constants because, as already mentioned:

- they already grant the freedom of combining algebraic polynomials with exponential and trigonometric functions with any number of individual shape parameters;
- when the various pieces are drawn from a single ECT-space, the existence of TB-splines is always ensured, possibly with some restriction on the partition; see [97, Section 2.4];
- the corresponding TB-splines are supported by the Matlab toolbox available in [115].

Outline of the thesis

This thesis is organized in the following chapters:

Chapter 1 revisits the fundamentals of B-splines. We discuss polynomial B-splines and the polynomial spline space they span, along with their properties. Additionally, we explore a local representation of B-splines, which is essential for describing their construction on locally refined meshes. Furthermore, we review some important operations for spline manipulation, such as knot insertion and degree elevation. We also touch briefly on the multivariate setting through the tensor-product structure. Finally, we outline the isogeometric Galerkin method, accompanied by different spline spaces as the solution space, which we will employ for solving differential problems in all case studies presented here.

Chapter 2 provides an overview of Tchebycheffian spline theory. We introduce the notion of ECT-spaces and explore an important large class of ECT-spaces, which are the null-space of linear differential operators. ECT-spaces can be glued together to form piecewise functions, known as Tchebycheffian spline spaces. We introduce the basis of such spaces, known as Tchebycheffian B-splines, outlining their properties similar to B-splines and their approximation properties, which are further numerically analyzed in subsequent chapters. Additionally, we briefly revisit the multi-degree construction of Tchebycheffian spline spaces.

Chapter 3 presents the application of isogeometric analysis with Tchebycheffian B-splines. We formulate the isogeometric Galerkin approach based on tensor-product TB-splines and devise a problem-driven strategy to select the ECT-spaces and their shape parameters in order to exploit the full flexibility of Tchebycheffian splines. Later, we present a bunch of case studies illustrating the performance of the proposed approach in isogeometric analysis. Finally, we present some conclusive remarks from the case studies.

Chapter 4 introduces LR TB-splines as a generalization of LR B-splines. We commence with the local representation of TB-splines and knot insertion in a univariate setting and extend it to bivariate TB-splines defined by local knot vectors. Subsequently, we revisit the basic construction of LR-meshes and define TB-splines on such meshes, demonstrating their construction through successive line insertion. We then explore the application of Tchebycheffian B-splines on LR-meshes in the context of the isogeometric Galerkin method, discussing related adaptive refinement strategies. Finally, we present various numerical results, showcasing the

performance of adaptive isogeometric analysis based on LR TB-splines for the solution of few classical benchmark differential problems.

Chapter 5 presents non-uniform degree B-splines on LR-meshes. In this perspective, the considered LR-meshes allow for different order of continuity across different meshlines. We introduce an hp -refinement algorithm aimed at constructing non-uniform degree splines through LR-meshes and we discuss the data structures used for the implementation. Later we test the effectiveness of the proposed non-uniform degree B-splines on LR-meshes by presenting their performances in some benchmark problems in isogeometric analysis.

Finally, we give an overview of the contributions, conclusions and suggestions for further research.

Publications

The contributions of the thesis are based on the following works:

- K. Raval, C. Manni, and H. Speleers, *Tchebycheffian B-splines in isogeometric Galerkin methods*, Computer Methods in Applied Mechanics and Engineering **403** (2023), 115648.
- K. Raval, C. Manni, and H. Speleers, *Adaptive isogeometric analysis based on Tchebycheffian splines*, Computer Methods in Applied Mechanics and Engineering, Accepted.
- K. Raval, R. R. Hiemstra, K. A. Johannessen, T. Dokken, and A. Raffo, *Local h -, p - and k - adaptivity with LR B-splines*, In preparation.

Preliminaries

Splines typically refer to univariate piecewise (algebraic) polynomial functions with certain smoothness. Their popularity stems from their representation in terms of the B-splines.

In Section 1.1, after giving a formal definition of splines, we revisit the standard polynomial B-splines, which enjoy properties like local linear independence, minimal support, non-negativity and partition of unity and can be computed through a stable recurrence relation. Subsequently we summarize the main aspects of two fundamental operations in B-spline manipulation: knot insertion and degree elevation.

In Section 1.2 we outline the isogeometric Galerkin method for solving differential equations and present the notations for the case studies.

1.1 Polynomial spline spaces

Here, we define the polynomial spline space as a space of piecewise polynomials of degree p , defined by a specified sequence of breakpoints and certain prescribed smoothness at these breakpoints.

Let \mathcal{M} be a sequence of breakpoints on the interval $[a, b] \in \mathbb{R}$,

$$\mathcal{M} := \{a =: x_0 < x_1 < \dots < x_{m-1} < x_m := b\}. \quad (1.1)$$

We set $J_i := [x_{i-1}, x_i]$, $i = 1, \dots, m - 1$, and $J_m := [x_{m-1}, x_m]$. For $i = 1, \dots, m$, let \mathbb{P}_p be the space of algebraic polynomials of degree less than or equal to p . This space has dimension $p + 1$. Moreover, consider a sequence of $m - 1$ integers

$$\mathbf{r} := \left\{ r_i \in \mathbb{Z} : -1 \leq r_i \leq p - 1, i = 1, \dots, m - 1 \right\}. \quad (1.2)$$

The space $\mathbb{S}_{\mathcal{M}}^{\mathbf{r}}$ of polynomial splines of degree p with smoothness \mathbf{r}

over the partition \mathcal{M} is defined as

$$\begin{aligned} \mathbb{S}_p^r(\mathcal{M}) := & \left\{ f : [a, b] \rightarrow \mathbb{R} : f|_{J_i} \in \mathbb{P}_p, i = 1, \dots, m; \right. \\ & \left. D_{-}^l f(x_i) = D_{+}^l f(x_i), l = 0, \dots, r_i, i = 1, \dots, m - 1 \right\}. \end{aligned} \quad (1.3)$$

Its dimension is given by

$$n := p + 1 + \sum_{i=1}^{m-1} (p - r_i). \quad (1.4)$$

We refer the reader to [12, 75, 104] for details on the foundation of spline theory.

1.1.1 Polynomial B-splines

In this section, we revisit one of the most powerful tools in computer aided geometric design and approximation theory: B-splines. We define notation for standard B-splines and recall some of their key properties. For further details and proofs of the properties presented in this section, we refer the reader to [12, 75, 94].

In order to define B-splines, we need the concept of knot vector. For a given interval $[a, b]$ and $n \in \mathbb{N}$, let us put

$$\begin{aligned} \Xi := (\xi_k)_{k=1}^{n+p+1} &= (\xi_1, \xi_2, \dots, \xi_{n+p+1}), \\ \xi_k &\leq \xi_{k+1}, \\ \xi_1 = \dots = \xi_{p+1} &= a, \quad b = \xi_{n+1} = \dots = \xi_{n+p+1}. \end{aligned} \quad (1.5)$$

Ξ is called an (open) knot vector of degree p and length $n + p + 1$. The elements of the sequence Ξ are called knots. They can be repeated: the number of times a knot value ξ_k is repeated in the knot vector is referred to as the knot's multiplicity. The multiplicity of a knot is assumed to be less than or equal to $p + 1$. The knot vector (1.5) allows us to define n B-splines of degree p according to the so-called Cox-de Boor recurrence relation as stated in the following definition¹

Definition 1.1 (B-splines). For a given open knot vector (1.5) of degree p , the corresponding B-splines $\{N_{k,p} : k = 1, \dots, n\}$ are defined by the

¹There are several ways to define B-splines, based on recurrence, divided differences, integration, etc. Each of those definitions has certain advantages according to the problem one has to deal with. We refer to [104] for an extended bibliography on this topic. In Chapter 2 a definition via an integral recurrence relation will be presented.

1.1. Polynomial spline spaces

following recursion relation

$$N_{k,p}(x) := \frac{x - \xi_k}{\xi_{k+p} - \xi_k} N_{k,p-1}(x) + \frac{\xi_{k+p+1} - x}{\xi_{k+p+1} - \xi_{k+1}} N_{k+1,p-1}(x), \quad x \in [a, b], \quad (1.6)$$

starting from

$$N_{k,0}(x) := \begin{cases} 1, & x \in [\xi_k, \xi_{k+1}), \\ 0, & \text{otherwise,} \end{cases}$$

and under the convention that fractions with a zero denominator have value zero².

Example 1.2. Figure 1.1 illustrates the evaluation of a quadratic ($p = 2$) B-spline using the Cox-de Boor recurrence relation as in Definition 1.1.

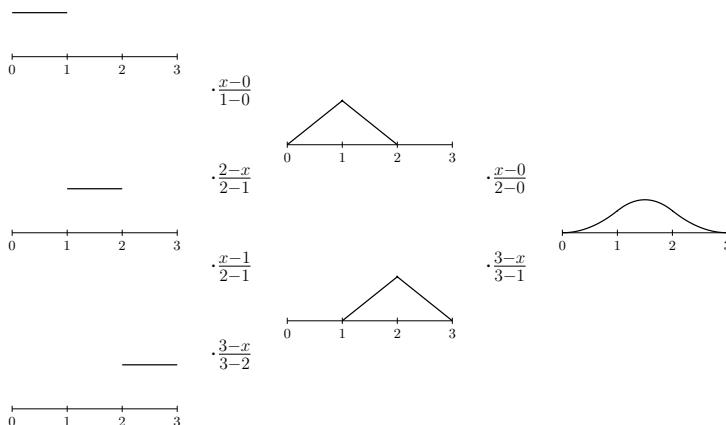


Figure 1.1: Recursive evaluation of a quadratic B-spline on $\Xi = (0, 0, 0, 1, 2, 3, 3, 3)$.

The interval $[a, b]$ is called *basic interval*. For practical purposes it is convenient to properly redefine the B-splines at the last point of the basic interval to avoid that they all vanish there. From now on we assume

$$N_{k,p}(b) := \lim_{x \rightarrow b^-} N_{k,p}(x), \quad k = 1, \dots, n.$$

The B-splines enjoy the properties collected in the following proposition.

²This defines B-splines on the interval $[a, b]$. Of course, they can be also interpreted as functions $\mathbb{R} \rightarrow \mathbb{R}$ just setting their value equal 0 outside the interval $[a, b]$. This interpretation will be used sometimes for notational convenience.

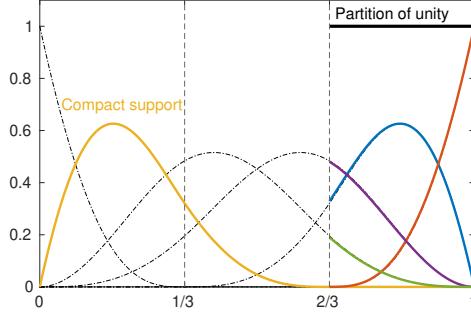


Figure 1.2: Illustration of B-spline properties.

Proposition 1.3 (B-spline properties). *Let $\{N_{k,p} : k = 1, \dots, n\}$ be the functions in Definition 1.1. Then, the following properties hold (See Figure 1.2).*

- *Non-negativity:* $N_{k,p}(x) > 0$ for all $x \in (\xi_k, \xi_{k+p+1})$;
- *Local support:* $N_{k,p}(x) = 0$ for all $x \notin [\xi_k, \xi_{k+p+1}]$;
- *Local partition of unity:* $\sum_{k=j-p}^j N_{k,p}(x) = 1$ for all $x \in [\xi_j, \xi_{j+1}]$;
- *Local linear independence:* the functions $\{N_{k,p}\}_{k=j-p}^j$ form a basis of polynomial space \mathbb{P}_p on $[\xi_j, \xi_{j+1}]$ for $p + 1 \leq j \leq n$,
- *Interpolation at end-points:*

$$\begin{aligned} N_{1,p}(a) &= 1, & N_{k,p}(a) &= 0, & k &= 2, \dots, n; \\ N_{n,p}(b) &= 1, & N_{k,p}(b) &= 0, & k &= 1, \dots, n-1. \end{aligned}$$

As stated by the Curry-Schoenberg theorem, see e.g. [12], with a proper selection of the knot vector B-splines form a basis of the spline space (1.3). More precisely, let a sequence of breakpoints \mathcal{M} as in (1.1) and a vector \mathbf{r} as in (1.2) be given and let n be as in (1.4). We consider the knot vector

$$\begin{aligned} \Xi &:= (\xi_k)_{k=1}^{n+p+1} \\ &:= \left(\underbrace{x_0, \dots, x_0}_{p+1 \text{ times}}, \underbrace{x_1, \dots, x_1}_{p-r_1 \text{ times}}, \dots, \underbrace{x_{m-1}, \dots, x_{m-1}}_{p-r_{m-1} \text{ times}}, \underbrace{x_m, \dots, x_m}_{p+1 \text{ times}} \right) \\ &= \left(x_0^{<\mu_0>}, x_1^{<\mu_1>}, \dots, x_{m-1}^{<\mu_{m-1}>}, x_m^{<\mu_m>} \right), \end{aligned} \quad (1.7)$$

where $\mu_0 := \mu_m := p + 1$, $\mu_l := p - r_l$, $l = 1, \dots, m - 1$, and $x_l^{<\mu_l>}$ means that x_l is repeated μ_l times, so that

$$\sum_{l=0}^m \mu_l = n + p + 1.$$

Then we have,

Theorem 1.4 (Curry-Schoenberg). *The B-splines $\{N_{k,p} : k = 1, \dots, n\}$ given in Definition 1.1 by the knot vector (1.7) are linearly independent and form a basis of the spline space (1.3).*

The above theorem ensures that any element $f \in \mathbb{S}_p^r(\mathcal{M})$ can be represented in the form

$$f(x) := \sum_{k=1}^n c_k N_{k,p}(x), \quad x \in [a, b], \quad c_k \in \mathbb{R}. \quad (1.8)$$

In particular, for $p \geq 1$ the identity function x belongs to the space (1.3) and we have

$$x = \sum_{k=1}^n \xi_k^* N_{k,p}(x), \quad x \in [a, b], \quad (1.9)$$

for some $\xi_k^* \in \mathbb{R}$. The coefficients ξ_k^* are called *Greville abscissae* and have the following expression

$$\xi_k^* = \frac{\xi_{k+1} + \dots + \xi_{k+p}}{p}. \quad (1.10)$$

Example 1.5. *The Greville abscissae for B-splines can be used to define a simple approximating operator to any function g defined on a given interval $[a, b]$:*

$$\mathfrak{V}_p g(x) := \sum_{k=1}^n g(\xi_k^*) N_{k,p}(x), \quad p \geq 1. \quad (1.11)$$

This operator is known as the Schoenberg operator (or Schoenberg quasi interpolant) [75, 103]. From (1.9) we have that the Schoenberg operator reproduces polynomials of first degree. It turns out that it provides an approximation which is second order accurate for a sufficiently smooth function g . In addition, the Schoenberg operator produces a spline function which mimics the shape of g . In particular, from the positivity and partition of unity property of B-splines, we have

$$\min_{x \in [a,b]} g(x) \leq \mathfrak{V}_p g(x) \leq \max_{x \in [a,b]} g(x). \quad (1.12)$$

Smooth piecewise polynomial parametric curves in \mathbb{R}^d can be easily

and efficiently obtained by means of B-splines. Let $\{\mathbf{P}_k \in \mathbb{R}^d\}_{k=0}^n$ be given, the parametric curve

$$\mathbf{C}(x) := \sum_{k=1}^n \mathbf{P}_k N_{k,p}(x), \quad x \in [a, b], \quad (1.13)$$

is called B-spline curve and the coefficients $\{\mathbf{P}_k \in \mathbb{R}^d\}_{k=0}^n$ are called *control points*. Connecting the control points we obtain the *control polygon* of the curve (1.13). Similarly, the vectors $(\xi_k^*, c_k)^T, k = 1, \dots, n$ are called control points for the spline function (1.8).

Thanks to the properties stated in Proposition 1.3, any B-spline curve (or graph of a spline function) belongs to the convex hull of its control points. More precisely, for $x \in [\xi_j, \xi_{j+1}]$ $\mathbf{C}(x)$ belongs to the convex hull of $\{\mathbf{P}_k\}_{k=j-p}^j$. This shows how the control points strongly determine the shape of a B-spline curve and motivate their names.

Figure 1.3, illustrates an example of a quadratic spline constructed from the set of B-splines on an open knot vector with smoothness $\mathbf{r} = \{1, \dots, 1\}$ at the breakpoints, along with its control points $\mathbf{P}_k \in \mathbb{R}^2$.

1.1.2 Local representation of polynomial B-splines

Typically, univariate splines of polynomial degree p are defined by partitioning an interval into a sequence of breakpoints \mathcal{M} as in (1.1) and assigning a vector of local smoothness \mathbf{r} as in (1.2). A global knot vector (1.7), which incorporates the smoothness at the breakpoints, defines a set of B-splines that are a basis for the polynomial spline space (1.3).

However, from Definition 1.1 it is evident that a single B-spline $N_{k,p}$ only depends on a part of the knots. Therefore, to analyze and describe the properties of a single B-spline it is appropriate to isolate the part of the global knot vector (1.5) that defines each B-spline separately. In this perspective we introduce some additional notations.

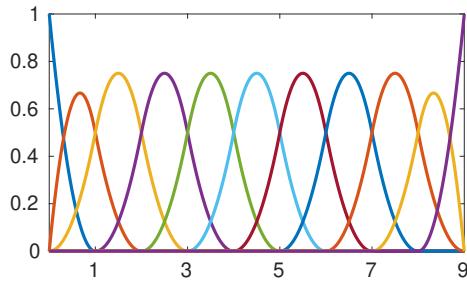
Definition 1.6 (Local knot vector). Given a sequence of breakpoints \mathcal{M} (1.1), a local knot vector is a non-decreasing sequence of $p + 2$ real numbers,

$$\Xi_k := (\xi_{k,1}, \xi_{k,2}, \dots, \xi_{k,p+2}) := (\xi_k, \dots, \xi_{k+p+1}) = (x_i^{<\mu_i>}, \dots, x_j^{<\mu_j>}),$$

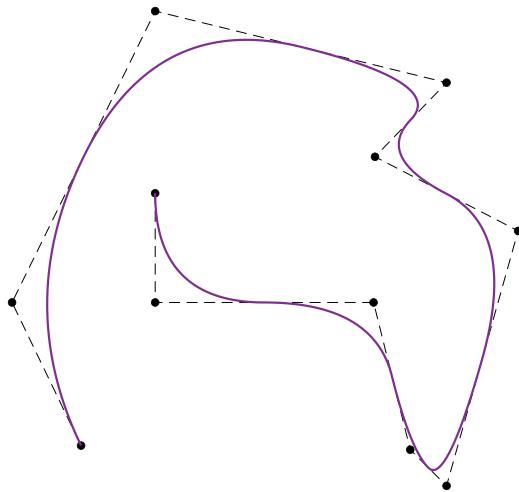
where $\mu_l, l = i, \dots, j$, denotes how many times x_l is repeated in the local knot vector, the indices i, j can be deduced from (1.7) and

$$\sum_{l=i}^j \mu_l = p + 2.$$

1.1. Polynomial spline spaces



(a) Set of quadratic B-spline basis on an open knot vector $\Xi = (0, 0, 0, 1, 2, \dots, 8, 9, 9, 9)$ with smoothness $r = \{1, \dots, 1\}$.



(b) A spline curve represented in terms of the same B-spline basis in (a), along with its control polygon (dashed black line).

Figure 1.3: B-spline basis functions and a corresponding B-spline planar curve.

Any B-spline $N_{\Xi_k, p}$ of degree p is uniquely identified by its local knot vector Ξ_k of length $p+2$ with the Cox-de Boor recurrence relation in Definition 1.1. Remarkably, $N_{\Xi_k, p}$ is of class $C^{p-\mu_l}$ at x_l .

Example 1.7. The recursive evaluation of a quadratic B-spline $N_{(0,1,2,3),2}$ in Example 1.2 can be achieved with the sole input of a local knot vector $\Xi_k = (0, 1, 2, 3)$ of length 4.

Throughout this thesis, we use the notation $N_{\Xi_k, p}$ to represent the B-spline constructed from the local knot vector Ξ_k , when we want to highlight its local construction. While for simplicity, we use the notation $N_{k, p}$ to denote B-splines whenever we are more interested in their role as basis functions for the spline space (1.3) constructed on a global knot vector Ξ as in (1.7). The distinction between global and local construction will be evident from the context or will be specified in individual chapters.

1.1.3 Knot insertion for polynomial B-splines

Knot insertion is a fundamental operation in B-spline manipulation: it consists of inserting a new knot into an existing knot vector. From Theorem 1.4 it is evident that knot insertion provides a spline space containing the space (1.3). If the new knot is already present in the initial knot vector, then the smoothness is reduced at the corresponding breakpoint. Otherwise, a new breakpoint is inserted in the partition. This leads to an explicit two-scale relation for B-splines under knot insertion, as follows.

Proposition 1.8 (Knot insertion in B-splines). *Let $N_{\Xi_k, p}$ be a B-spline as in Proposition 1.3, identified by the local knot vector $\Xi_k := (\xi_{k,1}, \dots, \xi_{k,p+2})$. Adding a knot $\hat{\xi} \in [\xi_{k,1}, \xi_{k,p+2}]$ results in two B-splines*

$$N_{(\xi_{k,1}, \dots, \hat{\xi}, \dots, \xi_{k,p+1}), p} \text{ and } N_{(\xi_{k,2}, \dots, \hat{\xi}, \dots, \xi_{k,p+2}), p}$$

such that

$$N_{(\xi_{k,1}, \dots, \xi_{k,p+2}), p} = v_k^{(1)} N_{(\xi_{k,1}, \dots, \hat{\xi}, \dots, \xi_{k,p+1}), p} + v_k^{(2)} N_{(\xi_{k,2}, \dots, \hat{\xi}, \dots, \xi_{k,p+2}), p} \quad (1.14)$$

with the coefficients $v_k^{(1)}, v_k^{(2)} \in [0, 1]$ given as

$$v_k^{(1)} := \begin{cases} \frac{\hat{\xi} - \xi_{k,1}}{\xi_{k,p+1} - \xi_{k,1}} & \xi_{k,1} < \hat{\xi} < \xi_{k,p+1}, \\ 1 & \xi_{k,p+1} \leq \hat{\xi} < \xi_{k,p+2}, \end{cases}$$

$$v_k^{(2)} := \begin{cases} 1 & \xi_{k,1} < \hat{\xi} \leq \xi_{k,2}, \\ \frac{\xi_{k,p+2} - \hat{\xi}}{\xi_{k,p+2} - \xi_{k,2}} & \xi_{k,2} < \hat{\xi} < \xi_{k,p+2}. \end{cases}$$

1.1. Polynomial spline spaces

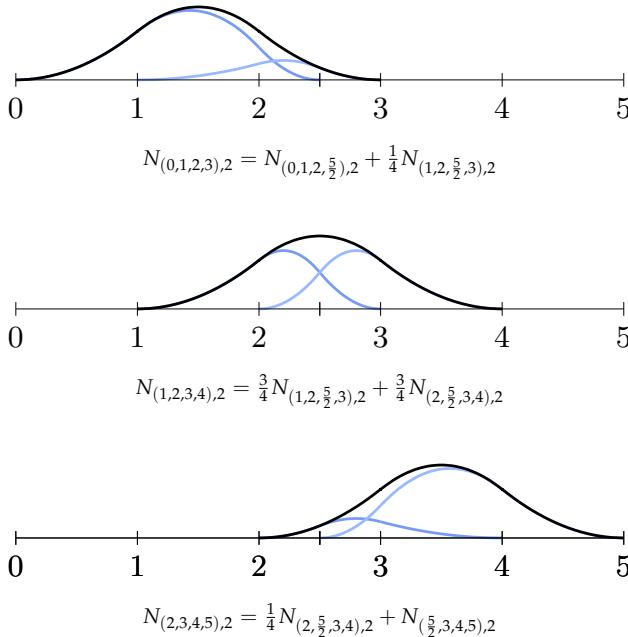


Figure 1.4: Inserting a knot $\hat{\xi} = \frac{5}{2}$ for different B-splines. Each existing spline (thick black line) is described as a combination of the refined B-splines (thin blue lines) with their corresponding scaling factors as given in (1.14).

Note that for simplicity we use the notation $(\xi_{k,1}, \dots, \hat{\xi}, \dots, \xi_{k,p+1})$ or $(\xi_{k,2}, \dots, \hat{\xi}, \dots, \xi_{k,p+2})$ to indicate the knot vector that is obtained after adding the knot $\hat{\xi}$ to the knot vector $(\xi_{k,1}, \dots, \xi_{k,p+1})$ or $(\xi_{k,2}, \dots, \xi_{k,p+2})$ respectively. This implicitly assumes that the resulting vector is in non-decreasing order. In particular, $\hat{\xi}$ can be the first or the last item in the vector according to its position w.r.t. $\xi_{k,2}$ or $\xi_{k,p+1}$ respectively.

Example 1.9. Figure 1.4 illustrates knot insertion in some B-splines as they are split into new refined B-splines. The linear mapping between the original functions and new functions is the two-scale relation that relates two different scales of refinement following (1.14).

Knot insertion provides an efficient way to refine the spline space (1.3) and it is of primal interest in different contexts. In particular, in this thesis knot insertion will be exploited to improve the accuracy of approximated solutions of differential problems obtained by isogeometric Galerkin methods. Knot insertion can also be performed by adding more than one knot at the time, see [36] for further details.

1.1.4 Degree elevation of polynomial B-splines

Degree elevation is another fundamental operation in spline manipulation: it consists of increasing the polynomial degree of a spline while maintaining the smoothness at the breakpoints. Any B-spline of degree p can be expressed as a linear combination of B-splines of degree $p + 1$. The degree elevation process is achieved by increasing the multiplicity of the breakpoints by one. Eventually leading toward nested spaces of splines.

More in detail, given a B-spline on a local knot vector Ξ_k of length $p + 2$

$$\Xi_k = (\xi_{k,l})_{l=1}^{p+2} = (x_i^{<\mu_i>}, \dots, x_j^{<\mu_j>}),$$

we increase the multiplicity of all breakpoints by 1, resulting into a new knot vector of length $n + p + 2$

$$\widehat{\Xi}_k := (\hat{\xi}_{k,l})_{l=1}^{n+p+2} = (x_i^{<\mu_i+1>}, \dots, x_j^{<\mu_j+1>}), \quad n := j - i + 1.$$

The knot vector $\widehat{\Xi}_k$ allows to construct n B-splines of degree $p + 1$. The B-spline $N_{\Xi_k, p}$ can be expressed in terms of these B-splines of degree $p + 1$ as follows

$$N_{\Xi_k, p} = \sum_{l=1}^n v_k^{(l)} N_{(\hat{\xi}_{k,l}, \hat{\xi}_{k,l+1}, \dots, \hat{\xi}_{k,l+p+2}), p+1}, \quad (1.15)$$

with $v_k^{(l)} \in \mathbb{R}$, $l = 1, \dots, n$. For further detail on degree elevation algorithms, we refer the reader to [37, 38, 95, 120] and references therein. Here we limit ourselves to remark that the splitting coefficients $v_k^{(l)}$ can be obtained in an elegant way by means of blossoming, [95, 96]. However, for practical purposes, these splitting coefficients can be simply computed by solving the linear system obtained by evaluating both sides of (1.15) sides at n points

$$\hat{x}_1 < \hat{x}_2 < \dots < \hat{x}_n$$

satisfying the Schoenberg-Whitney's Theorem, [12], i.e.

$$\hat{x}_l \in (\hat{\xi}_{k,l}, \hat{\xi}_{k,l+p+2}), \quad l = 1, \dots, n.$$

Example 1.10. Figure 1.5 presents degree elevation of some quadratic B-splines to cubic B-splines, with the linear mapping between the lower and higher degree B-splines as the two-scale relation that relates two different scales of refinement following (1.15).

1.1. Polynomial spline spaces

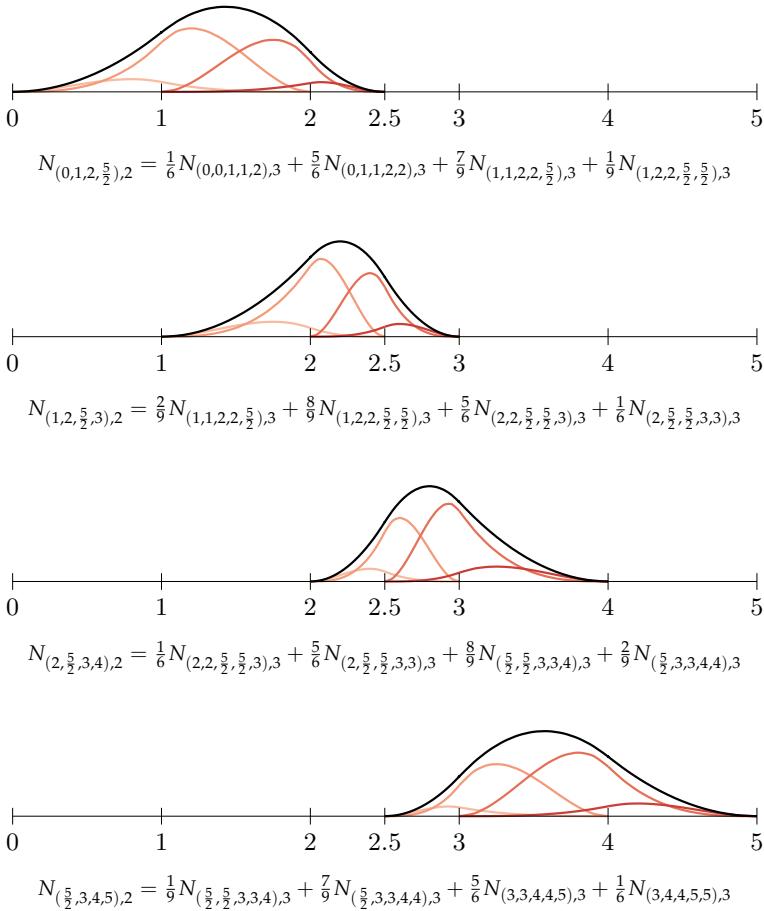


Figure 1.5: Degree elevation for different B-splines. Each quadratic spline is described as a combination of the cubic B-splines with their corresponding scaling factors as given in (1.15).

1.1.5 Tensor-product B-splines

A multivariate extension of B-splines can be simply obtained by considering the tensor-product approach.

Let $\mathbf{x} := (x_1, \dots, x_d)$ and $\mathbf{p} := (p_1, \dots, p_d)$. We define the multivariate B-spline $N_{\Xi_k, \mathbf{p}} : \mathbb{R}^d \rightarrow \mathbb{R}$, with local knot vectors

$$\Xi_k := \left(\Xi_{k_1}^1, \Xi_{k_2}^2, \dots, \Xi_{k_d}^d \right),$$

as

$$N_{\Xi_k, \mathbf{p}}(\mathbf{x}) := \prod_{i=1}^d N_{\Xi_{k_i}, p_i}(x_i) = \prod_{i=1}^d N_{k_i, p_i}(x_i), \quad (1.16)$$

where $N_{\Xi_{k_i}, p_i} = N_{k_i, p_i}$ is a univariate B-spline of degree p_i , for $i = 1, \dots, d$. The function $N_{\Xi_k, \mathbf{p}}$ belongs piecewisely to the space $\mathbb{P}_{p_1} \otimes \mathbb{P}_{p_2} \otimes \dots \otimes \mathbb{P}_{p_d}$ and its support is given by the Cartesian product of the supports of the local knot vectors in each direction

$$\text{supp}(N_{\Xi_k, \mathbf{p}}) = [\xi_{k_1, 1}^1, \xi_{k_1, p_1+2}^1] \times \dots \times [\xi_{k_d, 1}^d, \xi_{k_d, p_d+2}^d].$$

The d -variate tensor-product spline space is given by

$$\left\langle \prod_{i=1}^d N_{k_i, p_i} : k_i = 1, \dots, n_i, i = 1, \dots, d \right\rangle, \quad (1.17)$$

where n_i , $i = 1, \dots, d$ are the dimensions of the d univariate spline spaces.

Linear combinations of the basis function in (1.16) with coefficients in \mathbb{R}^q provide functions $\mathbb{R}^d \rightarrow \mathbb{R}^q$.

Example 1.11. For notational convenience for $d = 2$ the basis functions (1.16) can be shortly indexed as

$$N_i := N_{k_i, p_i} N_{l_i, p_i}, \quad i := (l - 1)n_1 + k, \quad k = 1, \dots, n_1, \quad l = 1, \dots, n_2. \quad (1.18)$$

With this notation

$$\sum_{i=1}^{n_1 n_2} \mathbf{P}_i N_i(\mathbf{x}), \quad \mathbf{P}_i \in \mathbb{R}^2,$$

describes a bounded region of \mathbb{R}^2 .

1.2 Isogeometric Galerkin methods

In this section, we outline the isogeometric Galerkin method, which uses splines as the basis functions. The flexibility of this method lies in the ability to vary the selection of spline spaces to suit the problem at hand. Throughout this thesis, we will explore different spline spaces while solving similar differential problems across various chapters. For the sake of simplicity, we concentrate on second-order elliptic differential problems with homogeneous Dirichlet boundary conditions.

Let \mathcal{L} be a linear second-order elliptic differential operator on the domain $\Omega \subset \mathbb{R}^d$ with Lipschitz boundary $\partial\Omega$. We consider the differential

problem

$$\begin{cases} \mathcal{L}u = f, & \text{in } \Omega, \\ u = 0, & \text{on } \partial\Omega, \end{cases} \quad (1.19)$$

whose weak form reads as follow:

$$\text{find } u \in \mathbb{V} \quad \text{such that} \quad a(u, v) = F(v), \quad \forall v \in \mathbb{V}, \quad (1.20)$$

where \mathbb{V} is a suitable function space, $a : \mathbb{V} \times \mathbb{V} \rightarrow \mathbb{R}$ is a bilinear form depending on \mathcal{L} , and $F : \mathbb{V} \rightarrow \mathbb{R}$ is a linear form depending on f .

The Galerkin approach to approximate the solution of (1.19) is based on the weak form (1.20). We select a finite-dimensional approximation space on Ω ,

$$\mathbb{W} := \langle \varphi_1, \varphi_2, \dots, \varphi_{n_W} \rangle \subset \mathbb{V}, \quad \dim(\mathbb{W}) = n_W, \quad (1.21)$$

and we look for

$$u_W \in \mathbb{W} \quad \text{such that} \quad a(u_W, w) = F(w), \quad \forall w \in \mathbb{W}.$$

Taking

$$u_W = \sum_{i=1}^{n_W} c_i \varphi_i$$

gives rise to a linear system $A\mathbf{c} = \mathbf{F}$, where the matrix A and the vector \mathbf{F} are defined as

$$A_{i,j} := a(\varphi_j, \varphi_i), \quad i, j = 1, \dots, n_W, \quad F_i := F(\varphi_i), \quad i = 1, \dots, n_W. \quad (1.22)$$

Different Galerkin methods correspond to different choices of the subspace \mathbb{W} .

In the standard formulation of Isogeometric Ananlysis (IgA), the physical domain Ω is represented by means of an invertible global geometry. We define the geometry map \mathbf{G} from the closure of the parametric domain $\widehat{\Omega} := (0, 1)^d$ to the closure of the physical domain Ω as (see Figure 1.6 and Equation (1.18))

$$\mathbf{G}(\hat{\mathbf{x}}) := \sum_{i=1}^{n_G} \mathbf{P}_i \hat{\varphi}_i(\hat{\mathbf{x}}), \quad \mathbf{P}_i \in \mathbb{R}^d, \quad (1.23)$$

where the basis functions

$$\{\hat{\varphi}_1, \dots, \hat{\varphi}_{n_G}\} \quad (1.24)$$

have to be selected so as to produce an exact (or at least very accurate) representation of the geometry (and to allow imposition of homogeneous

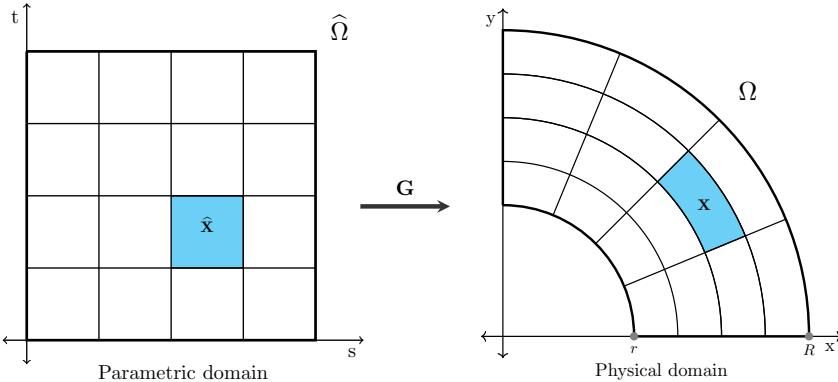


Figure 1.6: Isogeometric Galerkin method. Description of the physical domain Ω by means of a global geometry map \mathbf{G} .

boundary conditions). Following the isoparametric approach, the fields of interest are described with the same basis functions as the geometry in IgA. The space \mathbb{W} in (1.21), which incorporates the homogeneous boundary conditions, is then spanned by the isogeometric functions

$$\varphi_i(\mathbf{x}) := \hat{\varphi}_{j_i} \circ \mathbf{G}^{-1}(\mathbf{x}) = \hat{\varphi}_{j_i}(\hat{\mathbf{x}}), \quad i = 1, \dots, n_{\mathbb{W}}, \quad \mathbf{x} = \mathbf{G}(\hat{\mathbf{x}}), \quad (1.25)$$

where $n_{\mathbb{W}} < n_{\mathbf{G}}$ and $j_i : \{1, \dots, n_{\mathbf{G}}\} \rightarrow \{1, \dots, n_{\mathbb{W}}\}$ are a suitable rearrangement of the indices.

In the most common isogeometric formulation, the functions in (1.24) are chosen to be tensor-product B-splines or NURBS, [94]. Nevertheless, B-splines/NURBS are not a requisite ingredient in the isogeometric paradigm and several alternatives have been presented in the literature. In this thesis we exploit the richness of the various extensions of polynomial splines presented in the next chapter as discretization spaces (in the parametric domain) for isogeometric Galerkin methods.

More specifically, in this thesis we focus on the general second-order problem of the form

$$\begin{cases} -\nabla \cdot (\kappa \nabla u) + \mathbf{a} \cdot \nabla u + c u = f, & \text{in } \Omega = \mathbf{G}((0, 1)^2), \\ u = g, & \text{in } \partial\Omega, \end{cases} \quad (1.26)$$

where \mathbf{a} is the advection flow velocity, κ the diffusivity, c the reaction coefficient, f the prescribed source function, and g the Dirichlet boundary data. The physical domain Ω is described by the geometry map \mathbf{G} .

We define different constants representing the domination of a certain effect in the system depicting the nature of the solution for the differential

problem. These constants are as below.

- **Péclet number :** The dominance of advection over diffusion is defined by the global Péclet number

$$\mathbf{Pe}_g := \frac{\|\mathbf{a}\|}{\kappa}. \quad (1.27)$$

- **Damköhler number :** The dominance of reaction over diffusion is defined by the Damköhler number, defined as the ratio of the reaction coefficient (c) to the diffusion coefficient (κ).

A primer on Tchebycheffian splines

Similar to the polynomial case, Tchebycheffian splines are smooth piecewise functions whose pieces are drawn from Tchebycheff spaces, a natural generalization of algebraic polynomial spaces. Tchebycheffian splines share many properties with the classical polynomial splines but also offer a much more flexible framework due to the wide variety of Tchebycheff spaces.

In this chapter we introduce the so-called Extended Complete Tchebycheff (ECT-) spaces and we summarize some of their main properties. In particular, we focus on a subclass identified by linear differential operators with real constant coefficients that will be our target class for isogeometric analysis. Afterwards, we introduce Tchebycheffian splines – piecewise functions where the pieces belong to given ECT-spaces glued together with certain smoothness – and we briefly discuss the construction of TB-splines, i.e., B-spline like basis functions for Tchebycheffian splines. We provide a (theoretical) recursive definition of TB-splines with similar characteristics to the polynomial B-spline basis and highlight certain pitfalls of working with ECT-spaces.

Finally, we outline a further generalization of Tchebycheffian splines, the so-called multi-degree Tchebycheffian splines that allow for pieces belonging to Tchebycheff spaces of different dimension.

We refer the reader to [76, 104] for in-depth information about Tchebycheffian splines.

2.1 Extended Tchebycheff spaces

We consider a class of $(p + 1)$ -dimensional spaces that are a natural extension of the space of algebraic polynomials of degree less than or equal to p .

Definition 2.1 (Extended Tchebycheff space). Given an interval J and an integer $p \geq 0$, a space $\mathbb{T}_p(J) \subset C^p(J)$ of dimension $p + 1$ is an Extended Tchebycheff (ET-) space on J , if any Hermite interpolation problem with

$p + 1$ data on J has a unique solution in $\mathbb{T}_p(J)$. More in detail, let $z_1, z_2, \dots, z_{\bar{m}}$ be distinct points on J , with any positive integer \bar{m} and $d_1, d_2, \dots, d_{\bar{m}}$ be non-negative integers such that $p + 1 = \sum_{i=1}^{\bar{m}} (d_i + 1)$. Then for any $f_{i,j} \in \mathbb{R}$, there exists a unique $g \in \mathbb{T}_p(J)$ such that

$$D^j g(z_i) = f_{i,j}, \quad j = 0, \dots, d_i, \quad i = 1, \dots, \bar{m}.$$

As an immediate consequence of the previous definition, we have that any non-trivial element of an ET-space on J has at most p roots in J , counting multiplicity. This property shows the strong interconnection between ET-spaces and the space of algebraic polynomials.

2.1.1 Extended complete Tchebycheff spaces

We are now going to identify an interesting subclass of ET-spaces that strengthens the similarity with algebraic polynomials even further. We do this by generalizing the monomial basis functions, which form the so-called power basis of the polynomial space. We denote by $\langle g_0, \dots, g_k \rangle$ the linear space spanned by $\{g_0, \dots, g_k\}$.

Definition 2.2 (Extended complete Tchebycheff space). Given an interval J and an integer $p \geq 0$, a space $\mathbb{T}_p(J) \subset C^p(J)$ of dimension $p + 1$ is an Extended Complete Tchebycheff (ECT-) space on J , if there exist functions g_0, \dots, g_p such that $\mathbb{T}_p(J) = \langle g_0, \dots, g_p \rangle$ and every subspace $\langle g_0, \dots, g_k \rangle$ for $k = 0, \dots, p$ is an ET-space on J . The basis $\{g_0, \dots, g_p\}$ is called an ECT-system.

Example 2.3. The space of algebraic polynomials $\mathbb{P}_p = \langle 1, x, \dots, x^p \rangle$ is an ECT-space on the real line. Indeed, with any fixed point $z \in \mathbb{R}$, it can be accounted as the span of the ECT-system

$$\left\{ 1, x - z, \frac{(x - z)^2}{2}, \dots, \frac{(x - z)^p}{p!} \right\}. \quad (2.1)$$

It is evident from Definition 2.2 that every ECT-space on the interval J is an ET-space of the same dimension on the same interval. The contrary case is not always true. However, any ET-space on a bounded and closed interval is an ECT-space on the same interval [85]. In the scope of this thesis, we are only interested in bounded closed intervals. Hence, the notions of ET-space and ECT-space are interchangeable. Throughout the thesis, we consider ECT-spaces to allow for a neat construction of a proper basis of related spline spaces. In this perspective, we recall some additional properties of ECT-spaces.

A space $\mathbb{T}_p(J) \subset C^p(J)$ of dimension $p + 1$ is an ECT-space on J if and only if there exist positive functions $w_j \in C^{p-j}(J)$, $j = 0, 1, \dots, p$ such that $\mathbb{T}_p(J)$ is spanned by the following functions [76, 104]:

$$\begin{cases} g_0(x) := w_0(x), \\ g_1(x) := w_0(x) \int_z^x w_1(y_1) dy_1, \\ \vdots \\ g_p(x) := w_0(x) \int_z^x w_1(y_1) \int_z^{y_1} w_2(y_2) dy_2 \cdots \int_z^{y_{p-1}} w_p(y_p) dy_p dy_{p-1} \cdots dy_1, \end{cases} \quad (2.2)$$

where z is any point in J . The functions g_0, \dots, g_p are called generalized power functions, the functions w_0, \dots, w_p are usually referred to as weights, and the set $\{w_0, \dots, w_p\}$ is called a weight system generating $\mathbb{T}_p(J)$.

Example 2.4. A weight system for the space of algebraic polynomials \mathbb{P}_p is given by $\{w_j = 1 : j = 0, \dots, p\}$. Through (2.2) these weights provide the basis (2.1).

Example 2.5. Let $1 \leq \ell \leq p$ and $\alpha_1, \dots, \alpha_\ell \in \mathbb{R}$ with $\alpha_i \neq \alpha_j$ for each $i \neq j$. Then, we consider the functions

$$\begin{aligned} w_0(x) &= \cdots = w_{p-\ell}(x) = 1, \\ w_{p-\ell+1}(x) &= e^{\alpha_1 x}, \quad w_{p-\ell+k}(x) = e^{(\alpha_k - \alpha_{k-1})x}, \quad k = 2, \dots, \ell. \end{aligned} \quad (2.3)$$

For any $z \in \mathbb{R}$, the identified generalized power functions in (2.2) span the space

$$\left\langle 1, x, \dots, x^{p-\ell}, e^{\alpha_1 x}, \dots, e^{\alpha_\ell x} \right\rangle.$$

Since the functions in (2.3) are positive on \mathbb{R} , the above space is a $(p + 1)$ -dimensional ECT-space on \mathbb{R} for any $p \geq 1$.

Example 2.6. For $\beta \in \mathbb{R}$, $\beta > 0$, and $x \in J$ with

$$J = \left(-\frac{\pi}{2\beta}, \frac{\pi}{2\beta} \right),$$

we consider the functions

$$\begin{aligned} w_0(x) &= \cdots = w_{p-2}(x) = 1, \\ w_{p-1}(x) &= \cos(\beta x), \quad w_p(x) = \frac{1}{\cos^2(\beta x)}, \quad p \geq 1. \end{aligned} \quad (2.4)$$

For any $z \in J$, the identified generalized power functions in (2.2) span the space

$$\left\langle 1, x, \dots, x^{p-2}, \cos(\beta x), \sin(\beta x) \right\rangle. \quad (2.5)$$

Since the functions in (2.4) are positive on J , the above space is a $(p+1)$ -dimensional ECT-space on J for any $p \geq 1$. The space in (2.5) is often referred to as cycloidal space. Actually, for $p \geq 2$, it can be shown that the space in (2.5) is an ECT-space on any interval of length less than $2\pi/\beta$; see [34] and also [76, Section 2].

Remark 2.7. A given ECT-space on an interval J can be identified by different systems of weights; see [76, Section 2] for details and examples. In particular, it is clear that the two weight systems $\{w_0, \dots, w_p\}$ and $\{K_0 w_0, \dots, K_p w_p\}$, where K_0, \dots, K_p are positive constants, identify the same ECT-space.

Remark 2.8. Let \mathbb{T}_p be a space of dimension $p+1$ on J . Assume that the derivative space of \mathbb{T}_p , i.e., the space spanned by the derivatives of functions in \mathbb{T}_p , is a p -dimensional ECT-space on the interval J generated by the weights $\{w_1, \dots, w_p\}$. Then, \mathbb{T}_p is a $(p+1)$ -dimensional ECT-space on J generated by the weights $\{1, w_1, \dots, w_p\}$.

Many properties of polynomials involve working with derivatives. For dealing with ECT-spaces generated by the weight system $\{w_0, \dots, w_p\}$ it is often convenient to replace the usual derivatives by some related differential operators. We define $D_0 f := f$ and

$$D_j f := D \left(\frac{f}{w_{j-1}} \right), \quad j = 1, \dots, p+1,$$

and we set

$$L_j := D_j D_{j-1} \cdots D_0, \quad j = 0, \dots, p+1. \quad (2.6)$$

The operator L_j can be seen as a natural substitute for D^j ; see [104, Section 9.1]. In particular, we have

$$L_j g_k(x) = \begin{cases} w_j(x) \int_z^x w_{j+1}(y_{j+1}) \int_z^{y_{j+1}} \cdots \int_z^{y_{k-1}} w_k(y_k) dy_k dy_{k-1} \cdots dy_{j+1}, & k = j, \dots, p, \\ 0, & k = 0, \dots, j-1. \end{cases}$$

where g_k are defined in (2.2). This confirms that the generalized power functions are the natural extension of the monomial basis functions for algebraic polynomials.

2.1.2 Null-spaces of linear differential operators

A large and interesting class of ECT-spaces is given by the null-spaces of linear differential operators with real constant coefficients. More precisely, let us consider the differential operator defined by

$$\mathcal{L}_p f := D^{p+1}f + \sum_{j=0}^p a_j D^j f, \quad f \in C^{p+1}(\mathbb{R}), \quad a_j \in \mathbb{R}, \quad j = 0, \dots, p, \quad (2.7)$$

and let

$$\mathfrak{p}_p(\omega) := \omega^{p+1} + \sum_{j=0}^p a_j \omega^j \quad (2.8)$$

be its characteristic polynomial and \mathbb{N}_p its null-space.

It is well known that the null-space of the differential operator (2.7) is easily described by means of the roots of the characteristic polynomial (2.8). Let $\omega = \alpha + i\beta$ be a root of multiplicity $\mu \geq 1$ of the polynomial in (2.8) for $\alpha, \beta \in \mathbb{R}$ and $i := \sqrt{-1}$. Then, this root generates the following fundamental subspace:

- if $\beta = 0$, then

$$\left\langle x^k e^{\alpha x} : k = 0, \dots, \mu - 1 \right\rangle \subseteq \mathbb{N}_p;$$

- if $\beta \neq 0$, then the complex conjugate of ω is also a root of multiplicity μ , and

$$\left\langle x^k e^{\alpha x} \cos(\beta x), x^k e^{\alpha x} \sin(\beta x) : k = 0, \dots, \mu - 1 \right\rangle \subseteq \mathbb{N}_p.$$

The above subspaces for all roots together form the null-space \mathbb{N}_p . To ensure that the constants belong to \mathbb{N}_p , we have to assume that $\omega = 0$ is a root of the characteristic polynomial in (2.8).

Example 2.9. Setting $a_j = 0$, $j = 0, \dots, p$ in (2.7) gives the linear differential operator $\mathcal{L}_p f = D^{p+1}f$. In this case, the null-space is the space of algebraic polynomials $\mathbb{N}_p = \mathbb{P}_p$; see also Example 2.3.

Remark 2.10. If the polynomial (2.8) has only real roots, then \mathbb{N}_p is an ECT-space on the whole real line, and in particular on any bounded interval $[a, b]$. On the other hand, if the characteristic polynomial has also complex roots, then \mathbb{N}_p is an ECT-space on sufficiently small intervals; see [104]. More precisely, \mathbb{N}_p is an ECT-space on any interval $[a, b]$ such that

$$b - a < \mathfrak{l}_p.$$

The value ℓ_p is referred to as critical length and it can be bounded from below as

$$\ell_p \geq \pi/B > 0, \quad (2.9)$$

where B denotes the maximum imaginary part of all non-real roots of the characteristic polynomial. For a detailed study about the critical lengths of such ECT-spaces, the reader is referred to [6] and [33].

In the rest of the thesis we will mainly focus on $(p+1)$ -dimensional ECT-spaces that are null-spaces of linear differential operators with real constant coefficients, as in (2.7). The reference interval J ensuring the ECT-property will be specified when necessary. We will always assume that constants belong to the considered ECT-spaces. Let us now introduce the notation we are going to use for such spaces and give some examples.

Consider linear differential operators whose characteristic polynomial has roots $\omega_k = \alpha_k + i\beta_k$ of multiplicity $\mu_k \geq 1$, with $k = 0, \dots, M$, such that $\sum_{k=0}^M \mu_k = p+1$, and $\omega_0 = 0$. Under this assumption, the corresponding null-space is uniquely characterized by the following vector with $p+1-\mu_0$ components:

$$\mathcal{W} := \left(\underbrace{\omega_1, \dots, \omega_1}_{\mu_1 \text{ times}}, \underbrace{\omega_2, \dots, \omega_2}_{\mu_2 \text{ times}}, \dots, \underbrace{\omega_M, \dots, \omega_M}_{\mu_M \text{ times}} \right). \quad (2.10)$$

To highlight the dependence of the null-space on the roots (2.10), we use the notation

$$\mathbb{P}_p^{\mathcal{W}} = \mathbb{P}_p^{(\alpha_1+i\beta_1, \dots, \alpha_1+i\beta_1, \alpha_2+i\beta_2, \dots, \alpha_2+i\beta_2, \dots, \alpha_M+i\beta_M)}. \quad (2.11)$$

The ECT-spaces constructed from the null-spaces of linear differential operators with real constant coefficients offer great flexibility through the shape parameters to be chosen in (2.10). A particular selection of these shape parameters results into familiar subclasses of ECT-spaces. We now list some popular examples.

- The class of algebraic polynomial spaces is the most established subclass (see Example 2.3):

$$\mathbb{P}_p = \langle 1, x, \dots, x^p \rangle.$$

- Generalized polynomial spaces, see [76, 78], enrich algebraic polynomial spaces by a pair of functions. An important class of generalized polynomial spaces can be obtained as null-spaces of linear differential operators with constant coefficients. A first example is the algebraic polynomial space of degree $p-2$ enriched with two

exponential functions. Taking $\omega_1 = \alpha$, $\omega_2 = -\alpha$, and $\mu_1 = \mu_2 = 1$ gives the following null-space:

$$\begin{aligned}\mathbb{P}_p^{(\alpha, -\alpha)} &= \left\langle 1, x, \dots, x^{p-2}, e^{\alpha x}, e^{-\alpha x} \right\rangle \\ &= \left\langle 1, x, \dots, x^{p-2}, \cosh(\alpha x), \sinh(\alpha x) \right\rangle, \quad p \geq 2;\end{aligned}\tag{2.12}$$

see also Example 2.5. This is an ECT-space on any interval according to Remark 2.10. Another instance of generalized polynomial spaces is the combination of the algebraic polynomial space of degree $p-2$ with two trigonometric functions sharing the same phase $\beta > 0$. Taking $\omega_1 = i\beta$, $\omega_2 = -i\beta$, and $\mu_1 = \mu_2 = 1$ gives the following null-space:

$$\mathbb{P}_p^{(i\beta, -i\beta)} = \left\langle 1, x, \dots, x^{p-2}, \cos(\beta x), \sin(\beta x) \right\rangle, \quad p \geq 2,\tag{2.13}$$

which is a cycloidal space; see also Example 2.6. Remark 2.10 provides a lower bound on its critical length, while it can be bounded from above by

$$l_p \leq \frac{2\pi}{\beta} \left\lfloor \frac{p}{2} \right\rfloor, \quad p \geq 2;$$

see [33]. A detailed and precise study on the critical length of cycloidal spaces can be found in [34].

- An interesting subclass of ECT-spaces obtained as null-spaces of linear differential operators are the following exponential and trigonometric spaces, both defined for $p = 2q$:

$$\begin{aligned}\mathbb{P}_p^{(\alpha, -\alpha, \dots, q\alpha, -q\alpha)} &= \left\langle 1, e^{\alpha x}, e^{-\alpha x}, e^{2\alpha x}, e^{-2\alpha x}, \dots, e^{q\alpha x}, e^{-q\alpha x} \right\rangle \\ &= \left\langle 1, \cosh(\alpha x), \sinh(\alpha x), \dots, \cosh(q\alpha x), \sinh(q\alpha x) \right\rangle,\end{aligned}$$

and

$$\mathbb{P}_p^{(i\beta, -i\beta, \dots, iq\beta, -iq\beta)} = \left\langle 1, \cos(\beta x), \sin(\beta x), \dots, \cos(q\beta x), \sin(q\beta x) \right\rangle,$$

identified by the vector (2.10) with $\omega_k = \pm k\alpha$ and $\omega_k = \pm ik\beta$, $k = 1, \dots, q$, respectively. In the former case, the space is an ECT-space on any interval, while in the latter case, the space is an ECT-space on any interval of length less than $\frac{\pi}{\beta}$ for $\beta > 0$; see [84, 100].

All the above examples are special instances of the following subclass of ECT-spaces that combines polynomial, exponential, and trigonometric

functions. Let $0 \leq \ell \leq p$ and $\alpha_1, \dots, \alpha_\ell \in \mathbb{R}$ with $\alpha_i \neq \alpha_j$ for each $i \neq j$. Moreover, let $0 \leq 2q \leq p - \ell$ and $\beta_1, \dots, \beta_q \in \mathbb{R}$ with $\beta_i \neq \beta_j$ for each $i \neq j$. Then, let us consider the space

$$\mathbb{P}_p^{(\alpha_1, \dots, \alpha_\ell, i\beta_1, -i\beta_1, \dots, i\beta_q, -i\beta_q)} = \left\langle 1, x, \dots, x^{p-\ell-2q}, e^{\alpha_1 x}, \dots, e^{\alpha_\ell x}, \cos(\beta_1 x), \sin(\beta_1 x), \dots, \cos(\beta_q x), \sin(\beta_q x) \right\rangle, \quad (2.14)$$

defined on a reference interval ensuring the ECT-property. Such space is invariant under translation of the interval. We are going to utilize this comprehensive subclass of ECT-spaces in the case studies presented in Section 3.3. In the following we will refer to p as the degree of the ECT-space.

Remark 2.11. *The subclass of ECT-spaces in (2.14) is a proper subclass of null-spaces of differential operators with real constant coefficients, see (2.11), but still grants the freedom of combining polynomials with exponential and trigonometric functions with any number of individual shape parameters to be selected according to (automatic) problem-driven strategies. Moreover, derivatives and integrals of functions belonging to spaces of the form (2.14) belong to spaces of the same form, possibly not including constants anymore.*

2.2 Tchebycheffian spline spaces

In a complete similarity with polynomial splines, Tchebycheffian splines are piecewise functions with pieces belonging to ECT-spaces which can be glued together with certain prescribed smoothness. More precisely, let \mathcal{M} be a partition of the interval $[a, b] \subset \mathbb{R}$ specified in terms of a sequence of breakpoints,

$$\mathcal{M} := \{a =: x_0 < x_1 < \dots < x_{m-1} < x_m =: b\}.$$

We set $J_i := [x_{i-1}, x_i]$, $i = 1, \dots, m-1$, and $J_m := [x_{m-1}, x_m]$. Furthermore, for $i = 1, \dots, m$ let $\mathbb{T}_{p,i}$ be an ECT-space of dimension $p+1$ on the closed interval $[x_{i-1}, x_i]$. Finally, let us consider a sequence of $m-1$ integers

$$\mathbf{r} := \left\{ r_i \in \mathbb{Z} : -1 \leq r_i \leq p-1, i = 1, \dots, m-1 \right\}.$$

The elements of the set¹

$$\begin{aligned} \mathbb{S}_p^{\mathbf{r}}(\mathcal{M}) := & \left\{ f : [a, b] \rightarrow \mathbb{R} : f|_{J_i} \in \mathbb{T}_{p,i}, i = 1, \dots, m; \right. \\ & \left. D_-^l f(x_i) = D_+^l f(x_i), l = 0, \dots, r_i, i = 1, \dots, m-1 \right\} \end{aligned} \quad (2.15)$$

are the Tchebycheffian splines of degree p and smoothness \mathbf{r} on the partition \mathcal{M} of the interval $[a, b]$. As for the polynomial case, see (1.4), the dimension of the space $\mathbb{S}_p^{\mathbf{r}}(\mathcal{M})$ is given by

$$n := p + 1 + \sum_{i=1}^{m-1} (p - r_i). \quad (2.16)$$

Of course, the spaces (2.15) and (1.3) agree if $\mathbb{T}_{p,i} = \mathbb{P}_p$ for $i = 1, \dots, m$. The reader is referred to [76, 104] for more details.

When building the spline space $\mathbb{S}_p^{\mathbf{r}}(\mathcal{M})$, all the pieces of the spline functions can be taken from the “same” ECT-space on the whole interval $[a, b]$, or from different ECT-spaces considered on the different intervals. The latter case provides a more general framework which allows us to locally exploit the full richness of ECT-spaces; however, it also entails theoretical difficulties, mainly concerning the existence of a basis possessing all the nice properties of classical polynomial B-splines. This will be discussed in the next subsection.

2.2.1 Tchebycheffian B-splines

We now introduce a basis of the space $\mathbb{S}_p^{\mathbf{r}}(\mathcal{M})$ that possesses all the characterizing properties of classical polynomial B-splines. In particular, the elements of this basis form a non-negative partition of unity and have minimal support. These basis functions will be referred to as Tchebycheffian B-splines (TB-splines).

The existence of TB-splines requires some assumptions on the sequence of ECT-spaces $\mathbb{T}_{p,i}$, $i = 1, \dots, m$ in (2.15) which can be expressed in terms of weight systems; see Section 2.1.1.

Definition 2.12 (Admissible weights). For $i = 1, \dots, m$, let $\{w_{j,i} : j = 0, \dots, p\}$ be a weight system generating the ECT-space $\mathbb{T}_{p,i}$ on the interval $[x_{i-1}, x_i]$. These weight systems are admissible for the space $\mathbb{S}_p^{\mathbf{r}}(\mathcal{M})$ in (2.15), if

$$w_{0,i} = 1, \quad i = 1, \dots, m,$$

¹A more precise notation for the space of Tchebycheffian splines should include also the ECT-spaces $\mathbb{T}_{p,i}$ to which the different pieces belong. However, to avoid too heavy symbols and to stress the similarity with the polynomial case, we prefer to skip the explicit indication of the involved ECT-spaces. It will be clear from the context which spaces $\mathbb{T}_{p,i}$ are used.

and for $i = 1, \dots, m - 1$ and $j = 0, \dots, r_i$,

$$D_-^l w_{j,i}(x_i) = D_+^l w_{j,i+1}(x_i), \quad l = 0, \dots, r_i - j.$$

Given a set of admissible weight systems for the space $\mathbb{S}_p^r(\mathcal{M})$, we define the global weight system $\{w_j : j = 0, \dots, p\}$ as

$$w_j(x) := w_{j,i}(x), \quad x \in J_i, \quad i = 1, \dots, m. \quad (2.17)$$

Similar to the standard polynomial B-splines, see (1.7), Tchebycheffian B-splines can be defined using a vector of non-decreasing (open) knots,

$$\begin{aligned} \Xi &:= (\xi_k)_{k=1}^{n+p+1} \\ &:= \left(\underbrace{x_0, \dots, x_0}_{p+1 \text{ times}}, \underbrace{x_1, \dots, x_1}_{p-r_1 \text{ times}}, \dots, \underbrace{x_{m-1}, \dots, x_{m-1}}_{p-r_{m-1} \text{ times}}, \underbrace{x_m, \dots, x_m}_{p+1 \text{ times}} \right), \end{aligned} \quad (2.18)$$

with n being the dimension of the space $\mathbb{S}_p^r(\mathcal{M})$ in (2.16).

Definition 2.13 (TB-splines). Assume there exist admissible weight systems as in Definition 2.12 for the space $\mathbb{S}_p^r(\mathcal{M})$. Given the knot vector (2.18) and the global weight system (2.17), the TB-splines²

$$\{N_{k,p} : k = 1, \dots, n\}$$

are defined recursively as follows [76]: for $x \in [a, b]$, $q = 0, \dots, p$ and $k = 1, \dots, n + p - q$,

$$N_{k,0}(x) := \begin{cases} w_p(x), & x \in [\xi_k, \xi_{k+1}), \\ 0, & \text{otherwise,} \end{cases}$$

and

$$N_{k,q}(x) := w_{p-q}(x) \int_a^x \left[\frac{N_{k,q-1}(y)}{d_{k,q-1}} - \frac{N_{k+1,q-1}(y)}{d_{k+1,q-1}} \right] dy, \quad q > 0,$$

where

$$d_{j,q-1} := \int_a^b N_{j,q-1}(y) dy,$$

²As for the space (2.15), a more precise notation for TB-splines should include also the ECT-spaces $\mathbb{T}_{p,i}$ we deal with. Nevertheless, to avoid too heavy symbols and to stress the similarity with the polynomial case, we prefer again to skip the explicit indication of the involved ECT-spaces. It will be clear from the context which spaces $\mathbb{T}_{p,i}$ are used. Moreover, as for the polynomial case, we will use the notation $N_{\Xi,p}$ instead of $N_{k,p}$ whenever we need to emphasizing the local knot vector identifying the TB-spline.

2.2. Tchebycheffian spline spaces

and if $d_{j,q} = 0$ then

$$\int_a^x \frac{N_{j,q}(y)}{d_{j,q}} dy := \begin{cases} 1, & x \geq \xi_{j+q+1}, \\ 0, & \text{otherwise.} \end{cases}$$

At the right end point b , the spline $N_{k,q}$ is defined by taking the limit from the left, that is $N_{k,q}(b) := \lim_{x \rightarrow b, x < b} N_{k,q}(x)$.

TB-splines exhibit a complete structural similarity with polynomial B-splines. In particular, they enjoy all the nice properties of polynomial B-splines outlined in Proposition 1.3, as collected in the following proposition. For more details on the properties of TB-splines, see [76, 87, 90] and references therein.

Proposition 2.14 (TB-spline properties). *Let $\{N_{k,p} : k = 1, \dots, n\}$ be the functions in Definition 2.13. Then,*

$$\mathbb{S}_p^r(\mathcal{M}) = \langle N_{k,p} : k = 1, \dots, n \rangle,$$

and the following properties hold (see Figure 2.1).

- *Non-negativity:* $N_{k,p}(x) > 0$ for all $x \in (\xi_k, \xi_{k+p+1})$;
- *Local support:* $N_{k,p}(x) = 0$ for all $x \notin [\xi_k, \xi_{k+p+1}]$;
- *Partition of unity:* $\sum_{k=1}^n N_{k,p}(x) = 1$ for all $x \in [a, b]$;
- *Local linear independence:* the set $\{N_{k,p}\}_{k=j-p}^j$ forms a basis of \mathbb{T}_{p,i_j+1} on J_{i_j+1} for all $p + 1 \leq j \leq n$, where i_j denotes the index such that $\xi_j = x_{i_j}$;
- *Interpolation at end-points:*

$$\begin{aligned} N_{1,p}(a) &= 1, & N_{k,p}(a) &= 0, & k &= 2, \dots, n; \\ N_{n,p}(b) &= 1, & N_{k,p}(b) &= 0, & k &= 1, \dots, n-1. \end{aligned}$$

When $m = 1$, so that the partition \mathcal{M} has no interior breakpoints, the space $\mathbb{S}_p^r(\mathcal{M})$ reduces to the ECT-space $\mathbb{T}_p = \mathbb{T}_{p,1}$. The corresponding TB-splines are called Tchebycheffian Bernstein functions, and play the Tchebycheffian counterpart of Bernstein polynomials in case of classical algebraic polynomials. Tchebycheffian Bernstein functions provide an interesting alternative basis to the generalized power functions of the space \mathbb{T}_p given in (2.2).

Some examples of the Tchebycheffian Bernstein basis for different ECT-spaces are depicted in Figure 2.2, and considering a partition consisting of

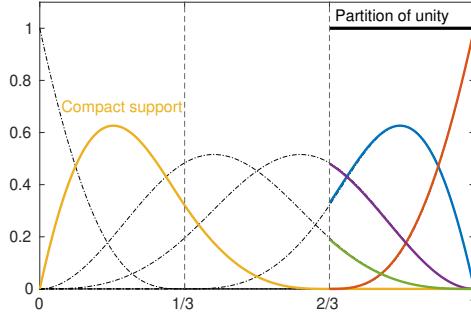


Figure 2.1: Illustration of TB-spline properties.

three intervals, the corresponding C^2 TB-splines based on the same ECT-spaces are given in Figure 2.3.

Definition 2.13 allows for the use of different ECT-spaces on different intervals. However, the requirement of admissible weight systems states that the various ECT-spaces we are dealing with should be identified by a sequence of weights which connect smoothly across the different segments. In general, the construction of such weights is complicated; see [5, 88]. Some comments about the existence of a TB-spline basis of the space $\mathbb{S}_p^r(\mathcal{M})$ are in order.

- Assume that \mathbb{T}_p is a $(p+1)$ -dimensional ECT-space on the interval $[a, b]$ and it contains constants. Moreover, assume that the derivative space of \mathbb{T}_p is a p -dimensional ECT-space on $[a, b]$ generated by the weights $\{w_1, \dots, w_p\}$. If all the spaces $\mathbb{T}_{p,i}$, $i = 1, \dots, m$ in (2.15) are selected as the restriction of \mathbb{T}_p , then from Remark 2.8 it follows that the weights $\{1, w_1, \dots, w_p\}$ restricted to the intervals $[x_{i-1}, x_i]$, $i = 1, \dots, m$ form admissible weight systems for the space $\mathbb{S}_p^r(\mathcal{M})$. In other words, if all the pieces of the Tchebycheffian spline functions are taken from a single space \mathbb{T}_p which is an ECT-space on $[a, b]$ containing constants, then the space $\mathbb{S}_p^r(\mathcal{M})$ admits a TB-spline basis provided that the derivative space of \mathbb{T}_p is also an ECT-space on $[a, b]$; see [76, 84, 104] and references therein.
- Assume now that \mathbb{T}_p is a $(p+1)$ -dimensional ECT-space on each interval $[x_{i-1}, x_i]$ separately, but not necessarily on the entire interval $[a, b]$. Moreover, assume that \mathbb{T}_p is the null-space of a linear differential operator as in (2.7) and it contains constants. If all the pieces of the Tchebycheffian spline functions are taken from such space \mathbb{T}_p , then the space $\mathbb{S}_p^r(\mathcal{M})$ admits a TB-spline basis on condition that $h_{\mathcal{M}} := \max_{i=1, \dots, m} (x_i - x_{i-1})$ is sufficiently small. In par-

2.2. Tchebycheffian spline spaces

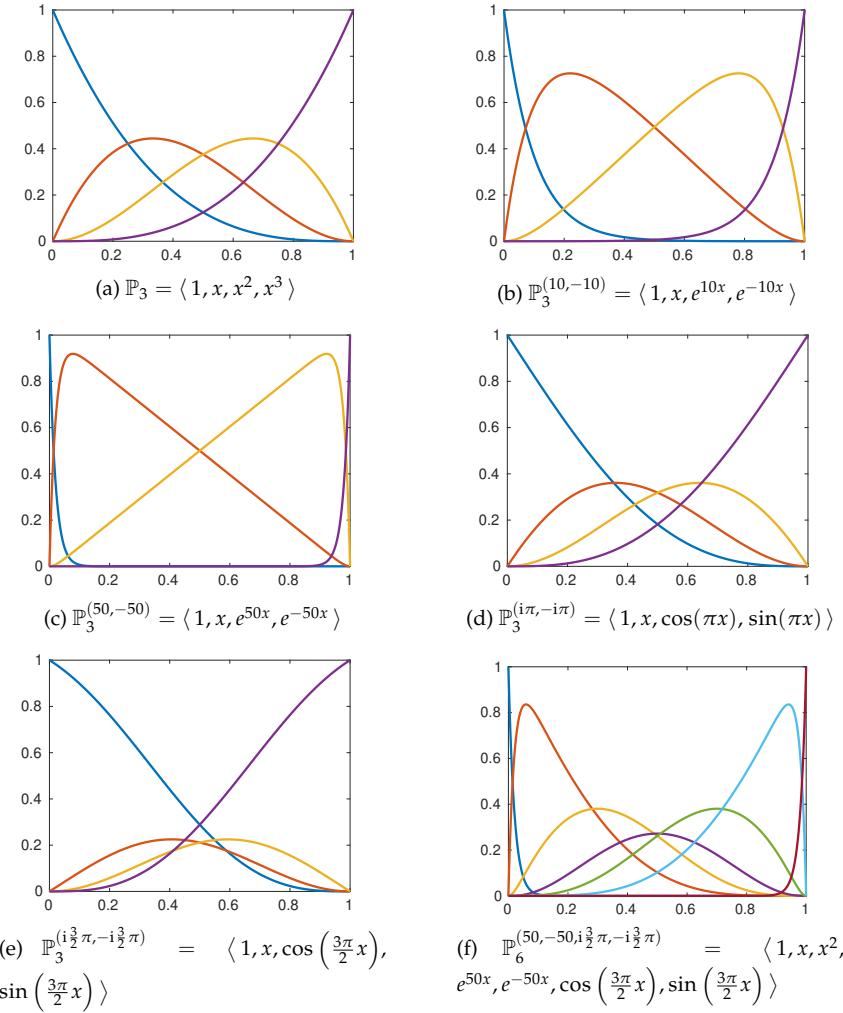


Figure 2.2: Tchebycheffian Bernstein basis on the interval $[0, 1]$ for different ECT-spaces.

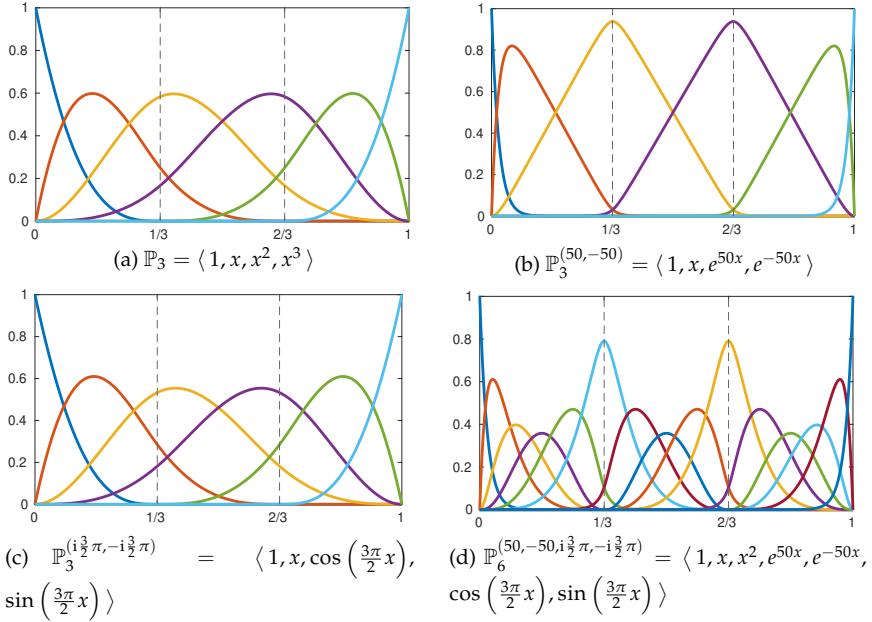


Figure 2.3: TB-splines on the partition $\mathcal{M} = \{0, 1/3, 2/3, 1\}$ with $\mathbf{r} = \{2, 2\}$ for different local ECT-spaces.

ticular, it admits a TB-spline basis when the derivative space of \mathbb{T}_p is an ECT-space on each interval $[\xi_{k+1}, \xi_{k+p}]$, where $\xi_{k+1} < \xi_{k+p}$ and $1 \leq k \leq n$. We refer the reader to [22] for details (and also [77]). Note that these intervals are closely related to the supports of the TB-splines; see Proposition 2.14.

- Admissible weight systems can be easily obtained for the interesting subclass of generalized polynomial splines, i.e., splines where the pieces are taken from generalized polynomial spaces (such as (2.12) and (2.13)), even considering different spaces on different intervals. This ensures the existence of TB-splines, called Generalized B-splines (GB-splines) in this setting, under very mild restrictions. In particular, the space $\mathbb{S}_p^{\mathbf{r}}(\mathcal{M})$ admits a TB-spline basis when the $(p-1)$ -th order derivative space of $\mathbb{T}_{p,i}$ in (2.15) is an ECT-space on $[x_{i-1}, x_i]$, $i = 1, \dots, m$. We refer the reader to [76, Section 4] for details.

Example 2.15. Similar to polynomial splines, a Tchebycheffian spline $f \in \mathbb{S}_p^{\mathbf{r}}(\mathcal{M})$ can be characterized by its coefficients $\{\mathbf{P}_k\}_{k=1}^n$ as in (1.13). We consider a Tchebycheffian spline curve analogous to the polynomial one in Figure 2.4 us-

ing the same set of control points but with different Tchebycheff spaces for different intervals in $\Xi = (0, 0, 0, 1, 2, \dots, 8, 9, 9, 9)$ with smoothness $\mathbf{r} = \{1, \dots, 1\}$. Specifically, the interval $[0, 2]$ with trigonometric functions $\mathbb{P}_2^{(i\frac{\pi}{2}, -i\frac{\pi}{2})}$, $[2, 7]$ with standard polynomials \mathbb{P}_2 and $[7, 9]$ with exponential functions $\mathbb{P}_2^{(30, -30)}$. With suitable shape parameters, this set of TB-splines as in Figure 2.4a, give an exact representation of a circular segment, polynomial segment and a hyperbolic segment in their respective intervals, see Figure 2.4b.

Example 2.16. Let $\mathbb{S}_p^{\mathbf{r}}(\mathcal{M})$ be a spline space as in (2.15) where

$$\mathbb{T}_{p,i} = \mathbb{P}_p^{(\alpha_1, \dots, \alpha_\ell)} = \left\langle 1, x, \dots, x^{p-\ell}, e^{\alpha_1 x}, \dots, e^{\alpha_\ell x} \right\rangle, \quad p \geq \ell \geq 0,$$

for $i = 1, \dots, m$. From Remark 2.10 we deduce that $\mathbb{P}_p^{(\alpha_1, \dots, \alpha_\ell)}$ is an ECT-space on the whole real line, and in particular on the interval $[a, b]$. Moreover, the derivative space of $\mathbb{P}_p^{(\alpha_1, \dots, \alpha_\ell)}$ is $\mathbb{P}_{p-1}^{(\alpha_1, \dots, \alpha_\ell)}$ (a space without constants when $p = \ell$), which is also an ECT-space on the whole real line. We conclude that the corresponding spline space $\mathbb{S}_p^{\mathbf{r}}(\mathcal{M})$ admits a TB-spline basis on any partition of the interval $[a, b]$, without any restrictions.

Example 2.17. Let $\mathbb{S}_p^{\mathbf{r}}(\mathcal{M})$ be a spline space as in (2.15) where

$$\begin{aligned} \mathbb{T}_{p,i} = & \left\langle 1, x, \dots, x^{p-\ell-2q}, e^{\alpha_1 x}, \dots, e^{\alpha_\ell x}, \right. \\ & \left. \cos(\beta_1 x), \sin(\beta_1 x), \dots, \cos(\beta_q x), \sin(\beta_q x) \right\rangle, \end{aligned}$$

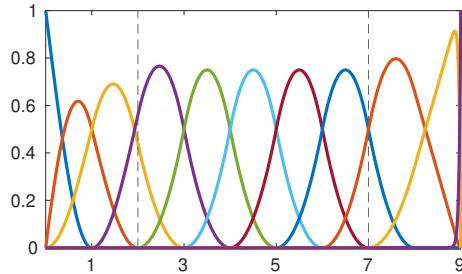
for $i = 1, \dots, m$, i.e., the space considered in (2.14). The case $q = 0$ is covered in Example 2.16. Hence, let us here focus on the case $q > 0$ and without loss of generality we assume $\beta_1 > 0, \dots, \beta_q > 0$. Since the pieces of the spline functions belong to the null-space of a linear differential operator and taking into account Remark 2.10, we conclude that the space $\mathbb{S}_p^{\mathbf{r}}(\mathcal{M})$ admits a TB-spline basis on the interval $[a, b]$ if the knot sequence Ξ in (2.18) satisfies

$$\xi_{k+p} - \xi_{k+1} < \frac{\pi}{\max\{\beta_1, \dots, \beta_q\}}, \quad k = 1, \dots, n.$$

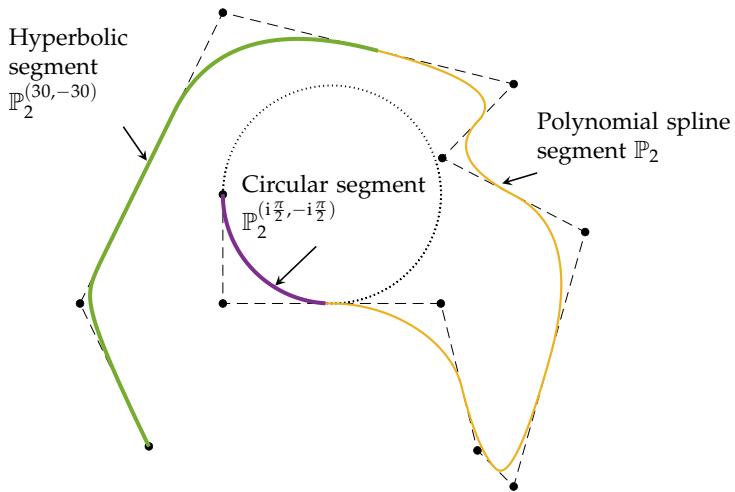
Example 2.18. Let $\mathbb{S}_p^{\mathbf{r}}(\mathcal{M})$ be a spline space as in (2.15) where

$$\mathbb{T}_{p,i} = \left\langle 1, \cos(\beta x), \sin(\beta x), \dots, \cos(q\beta x), \sin(q\beta x) \right\rangle, \quad \beta > 0, \quad p = 2q \geq 0,$$

for $i = 1, \dots, m$. As discussed in the previous example, such spline space admits a TB-spline basis on the interval $[a, b]$ if the knot sequence Ξ in (2.18) satisfies $\xi_{k+p} - \xi_{k+1} < \pi/(q\beta)$, $k = 1, \dots, n$. Actually, this bound can be improved:



(a) Set of quadratic TB-splines on $\Xi = (0, 0, 0, 1, 2, \dots, 8, 9, 9, 9)$ with space $\mathbb{P}_2^{(i\frac{\pi}{2}, -i\frac{\pi}{2})} = \langle 1, \cos(\frac{\pi}{2}x), \sin(\frac{\pi}{2}x) \rangle$ in $[0, 2]$, $\mathbb{P}_2 = \langle 1, x, x^2 \rangle$ in $[2, 7]$ and $\mathbb{P}_2^{(30, -30)} = \langle 1, e^{30x}, e^{-30x} \rangle$ in $[7, 9]$, with smoothness $r = 1$.



(b) A Tchebycheffian spline with its control polygon (black), represented in terms of the TB-splines in Figure 2.4a, exactly representing a quarter of a circle, a segment of polynomial spline and a hyperbola segment.

Figure 2.4: A quadratic Tchebycheffian spline analogous to the polynomial spline presented in Figure 1.3 along with its TB-spline with different local ECT-spaces.

the space $\mathbb{S}_p^r(\mathcal{M})$ admits a TB-spline basis if $\xi_{k+p} - \xi_{k+1} < \pi/\beta$, $k = 1, \dots, n$; see [84, 100].

Example 2.19. Let $\mathbb{S}_p^r(\mathcal{M})$ be a spline space as in (2.15) where

$$\mathbb{T}_{p,i} = \left\langle 1, x, \dots, x^{p-2}, \cos(\beta x), \sin(\beta x) \right\rangle, \quad \beta > 0, \quad p \geq 2,$$

for $i = 1, \dots, m$. As discussed in Example 2.17, such spline space admits a TB-spline basis on the interval $[a, b]$ if the knot sequence Ξ in (2.18) satisfies $\xi_{k+p} - \xi_{k+1} < \pi/\beta$, $k = 1, \dots, n$; see also Example 2.6. Actually, it is a generalized polynomial spline space and it can be shown to have a GB-spline basis if $x_i - x_{i-1} < \pi/\beta$, $i = 1, \dots, m$; see [76, Section 4].

In the rest of the thesis we will only consider Tchebycheffian spline spaces as in (2.15) obtained by considering the same ECT-space \mathbb{P}_p^W , defined in (2.14), on each interval of the partition \mathcal{M} . In view of the above discussion, this setting ensures the existence of a TB-spline basis whenever the derivative space of \mathbb{P}_p^W is an ECT-space on $[a, b]$. If this is not the case, a TB-spline basis can be obtained when a sufficiently fine partition of the given interval is considered. As numerically illustrated with case studies in Section 3.3, this simplified Tchebycheffian setting allows already for such a wide variety of ECT-spaces and related shape parameters to be extremely profitable in the context of IgA.

Remark 2.20. The recurrence relation in Definition 2.13 is the main theoretical tool to define and analyze TB-splines. However, it is of limited practical interest because it does not lead to an efficient evaluation strategy. Fortunately, this deficiency has been recently overcome: efficient routines to construct and manipulate TB-splines with pieces belonging to ECT-spaces as in Section 2.1.2 have been provided in [115] by implementing the approach described in [54]. This approach starts from the Tchebycheffian Bernstein functions on each interval $[x_{i-1}, x_i]$, $i = 1, \dots, m$, and, by using the smoothness constraints at all breakpoints, generates an extraction matrix expressing the TB-spline basis locally in terms of the Tchebycheffian Bernstein functions. The latter can be computed directly through Hermite interpolation (see Example 2.22 for an illustration). No weights are required along the process. Although the evaluation of TB-splines for extreme values of the shape parameters (i.e., the roots of the characteristic polynomial (2.8)), high degrees, high smoothness, and highly non-uniform partitions still faces numerical challenges [115], the above mentioned routines provide reliable tools for a large class of TB-splines of interest in practical applications. A Matlab implementation is publicly available through the ACM Collected Algorithms (CALGO) library [115].

Remark 2.21. The TB-spline construction procedure developed in [54, 115] does not require any knowledge of weights. In case the space $\mathbb{S}_p^r(\mathcal{M})$ does not possess

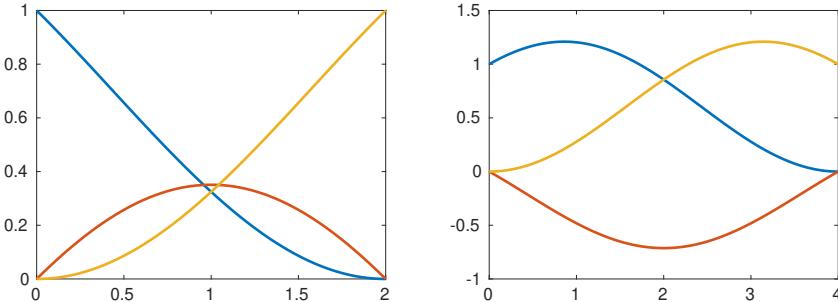


Figure 2.5: The functions satisfying (2.19)–(2.21) in the space $\mathbb{P}_2^{(i,-i)} = \langle 1, \cos(x), \sin(x) \rangle$. Left: interval $[0, 2]$, Right: interval $[0, 4]$.

admissible weight systems, the procedure simply returns a sequence of functions that might not enjoy the properties listed in Proposition 2.14. In this perspective, the computational process can also be used in practice to numerically test the (non-)existence of a TB-spline basis for a given Tchebycheffian spline space.

Example 2.22. *The space*

$$\mathbb{P}_2^{(i,-i)} = \langle 1, \cos(x), \sin(x) \rangle$$

is an ECT-space of dimension 3 on any interval of length less than 2π ; see Example 2.6. However, the space only admits a Tchebycheffian Bernstein basis on intervals of smaller length, namely less than π ; see Examples 2.18 and 2.19. If such a basis exists, then the corresponding basis elements $B_{j,2}$, $j = 0, 1, 2$, can be computed by solving the following Hermite interpolation problems:

$$B_{0,2}(a) = 1, \quad B_{0,2}(b) = 0, \quad DB_{0,2}(b) = 0; \quad (2.19)$$

$$B_{1,2}(a) = 0, \quad DB_{1,2}(a) = -DB_{0,2}(a), \quad B_{1,2}(b) = 0; \quad (2.20)$$

$$B_{2,2}(a) = 0, \quad DB_{2,2}(a) = 0, \quad B_{2,2}(b) = 1. \quad (2.21)$$

These interpolation problems have a unique solution whenever $b - a < 2\pi$, since $\mathbb{P}_2^{(i,-i)}$ is an ECT-space then. However, the unisolvency of the above interpolation problems does not ensure that the corresponding functions $B_{j,2}$, $j = 0, 1, 2$ enjoy all the properties listed in Proposition 2.14, in particular their non-negativity. This is illustrated in Figure 2.5 for the interval $[a, b] = [0, 2]$ (so $b - a < \pi$) and the interval $[a, b] = [0, 4]$ (so $\pi < b - a < 2\pi$).

2.2.2 Approximation properties of Tchebycheffian splines

Being interested in the application of Tchebycheffian splines in IgA, it is important to understand the approximation power of the space in (2.15). It turns out that, in general, smooth functions can be approximated by Tchebycheffian splines with the same orders of approximation as in the case of classical polynomial splines. The reader is referred to [104, Section 9.7] for details and we state a simplified version of [104, Theorem 9.38] below. Let $\mathbb{W}^{r,q}([a,b])$ be the standard Sobolev space on $[a,b]$ and let $\|\cdot\|_q$ denote the standard L^q -norm on $[a,b]$ for $1 \leq q \leq \infty$.

Theorem 2.23. *Let \mathbb{T}_p be a $(p+1)$ -dimensional ECT-space on the interval $[a,b]$ and let $\mathbb{S}_p^r(\mathcal{M})$ be a Tchebycheffian spline space where $\mathbb{T}_{p,i} = \mathbb{T}_p$ for $i = 1, \dots, m$. For any $f \in \mathbb{W}^{p+1,q}([a,b])$ there exists a spline $Qf \in \mathbb{S}_p^r(\mathcal{M})$ such that*

$$\|f - Qf\|_q \leq C (h_{\mathcal{M}})^{p+1} \|L_{p+1}f\|_q,$$

where L_{p+1} is defined in (2.6),

$$h_{\mathcal{M}} := \max_{i=1,\dots,m} (x_i - x_{i-1}),$$

and C is a constant independent of f and \mathcal{M} .

When considering Tchebycheffian spline spaces as in (2.15) such that each $\mathbb{T}_{p,i}$ contains the identity function, x , we have that $x \in \mathbb{S}_p^r(\mathcal{M})$. Therefore, if $\mathbb{S}_p^r(\mathcal{M})$ possesses a TB-spline basis, x can be represented as linear combination of TB-splines. With some abuse of notation we write

$$x = \sum_{k=1}^n \xi_k^* N_{k,p}(x), \quad x \in [a,b], \tag{2.22}$$

where the coefficients are again called Greville abscissae. Of course, their expression depends on the Tchebycheff spaces we are using to build $\mathbb{S}_p^r(\mathcal{M})$ and does not agree anymore with (1.10). Explicit expressions for Greville abscissae for TB-splines can be found in [76] and their values can be computed by means of the software in [115].

As in the polynomial case, Greville abscissae play an important role when dealing with Tchebycheffian splines. In particular, they allow to extend to the Tchebycheffian setting the Schoenberg operator introduced in Example 1.5 by considering exactly the same expression as in (1.11). Since TB-splines enjoy the same properties as classical polynomial B-splines, also in the Tchebycheffian setting the Schoenberg operator produces a spline function which mimics the shape of the function to be approximated. In particular, (1.12) still holds true. For these reasons the Schoenberg operator will sometimes be used in the next chapters for constructing

suitable Tchebycheffian spline approximations to Dirichlet boundary data of certain differential problems.

2.3 Multi-degree Tchebycheffian spline spaces

In the previous section we have discussed Tchebycheffian splines and their possible representation in terms of basis functions that enjoy all the nice properties of polynomial B-splines. Tchebycheffian (B-)splines are an extremely powerful extension of classical polynomial (B-)splines, which give access to the large variety of Tchebycheff spaces.

A further interesting extension of polynomial (B-)splines can be obtained by considering smooth piecewise functions whose pieces can be drawn from arbitrary ECT-spaces of possibly different dimensions. This leads to a larger class of splines that we are going to define and briefly discuss in this section.

Let us consider a partition \mathcal{M} of the interval $[a, b]$ as in (1.1). We also define an ECT-space of dimension $p_i + 1$ on each closed interval $[x_{i-1}, x_i]$, $i = 1, \dots, m$:

$$\mathbb{T}_{p_i, i} := \langle g_{0,i}, \dots, g_{p_i, i} \rangle, \quad g_{j,i} \in C^{p_i}([x_{i-1}, x_i]), \quad j = 0, \dots, p_i,$$

where $g_{0,i}, \dots, g_{p_i, i}$ are generalized powers defined in terms of positive weight functions $w_{j,i} \in C^{p_i-j}([x_{i-1}, x_i])$, $j = 0, \dots, p_i$ as in (2.2). By prescribing the smoothness at the break points, as in (1.2) we define the following space.

Definition 2.24 (MDT-spline space). Given the sets of integers

$$\mathbf{p} := \{p_1, \dots, p_m\}$$

and

$$\mathbf{r} := \left\{ r_i \in \mathbb{Z} : -1 \leq r_i \leq \min\{p_i, p_{i+1}\}, i = 1, \dots, m-1, r_0 = r_m = -1 \right\},$$

we define

$$\begin{aligned} \mathbb{S}_{\mathbf{p}}^{\mathbf{r}}(\mathcal{M}) := & \left\{ s : [a, b] \rightarrow \mathbb{R} : s|_{J_i} \in \mathbb{T}_{p_i, i}, i = 1, \dots, m, \right. \\ & \left. D_-^j s(x_i) = D_+^j s(x_i), j = 0, \dots, r_i \text{ and } i = 1, \dots, m-1 \right\}. \end{aligned} \tag{2.23}$$

This space is called Multi-Degree Tchebycheffian spline (MDT-spline) space.

The dimension of $\mathbb{S}_p^r(\mathcal{M})$ is given by

$$n := \sum_{i=0}^{m-1} (p_{i+1} - r_i) = \sum_{i=1}^m (p_i - r_i). \quad (2.24)$$

Remark 2.25. If $p_i = p$, $i = 1, \dots, m$, then the spline space in (2.23) reduces to the one in (2.15) and (2.24) agrees with (2.16) and so with (1.4). On the other hand, if $\mathbb{T}_{p_i, i} = \mathbb{P}_{p_i}$, $i = 1, \dots, m$ the space in (2.23) consists of smooth piecewise polynomial functions whose pieces can have different maximum degree; see [114, 122] and references therein.

We now introduce basis functions for the MDT-spline space $\mathbb{S}_p^r(\mathcal{M})$ that possess all the characterizing properties of classical polynomial B-splines. We call the corresponding functions Multi-Degree Tchebycheffian B-splines (MDTB-splines) to stress the fact that ECT-spaces of different dimensions can be employed on different intervals, in analogy with the polynomial MDB-splines considered in [114, 122].

The construction and analysis of MDTB-splines can be eased by considering two knot vectors,

$$\begin{aligned} \mathbf{u} &:= (u_k)_{k=1}^n := \left(\underbrace{x_0, \dots, x_0}_{p_1 - r_0 \text{ times}}, \dots, \underbrace{x_i, \dots, x_i}_{p_{i+1} - r_i \text{ times}}, \dots, \underbrace{x_{m-1}, \dots, x_{m-1}}_{p_m - r_{m-1} \text{ times}} \right), \\ \mathbf{v} &:= (v_k)_{k=1}^n := \left(\underbrace{x_1, \dots, x_1}_{p_1 - r_1 \text{ times}}, \dots, \underbrace{x_i, \dots, x_i}_{p_i - r_i \text{ times}}, \dots, \underbrace{x_m, \dots, x_m}_{p_m - r_m \text{ times}} \right). \end{aligned}$$

Assume there exist admissible weights for the space $\mathbb{S}_p^r(\mathcal{M})$ as in Definition 2.12. Let us define

$$p := \max_{1 \leq i \leq m} p_i,$$

then, the set of MDTB-splines $\{N_{k,p} : k = 1, \dots, n\}$ can be computed through an integral recurrence relation completely similar to the one of the Tchebycheffian B-splines in Definition 2.13. To this end, we define a global set of weight functions $\{w_j : j = 0, \dots, p\}$ by

$$w_j(x) := \begin{cases} w_{j,i}(x), & j \leq p_i, \\ 0, & \text{otherwise,} \end{cases} \quad x \in [x_{i-1}, x_i), \quad i = 1, \dots, m.$$

Then, the MDTB-splines³ $N_{k,p}$, $k = 1, \dots, n$, can be defined recursively as follows. For $q = 0, \dots, p$ and $k = p - q + 1, \dots, n$, the spline $N_{k,q}$ is

³Once again, with some abuse of notation, we use the same notation for MTDB-splines as for the polynomial and the Tchebycheffian case.

supported on the interval $[u_k, v_{k-p+q}]$, and is defined at $x \in [a, b)$ as

$$N_{k,0}(x) := \begin{cases} w_p(x), & x \in [x_{i-1}, x_i), \\ 0, & \text{otherwise,} \end{cases} \quad (2.25)$$

and

$$N_{k,q}(x) := w_{p-q}(x) \int_a^x \left[\frac{N_{k,q-1}(y)}{d_{k,q-1}} - \frac{N_{k+1,q-1}(y)}{d_{k+1,q-1}} \right] dy, \quad q > 0, \quad (2.26)$$

where

$$d_{j,q-1} := \int_a^b N_{j,q-1}(y) dy.$$

In the above we assumed that any undefined $N_{j,q-1}$ with $j < p - q + 2$ or $j > n$ must be regarded as the zero function, and we used the convention that if $d_{j,q-1} = 0$ then

$$\int_a^x \frac{N_{j,q-1}(y)}{d_{j,q-1}} dy := \begin{cases} 1, & x \geq u_j \text{ and } j \leq n, \\ 0, & \text{otherwise.} \end{cases}$$

At the right end point b , the spline $N_{k,q}$ is defined by taking the limit from the left, that is $N_{k,q}(b) := \lim_{x \rightarrow b, x < b} N_{k,q}(x)$. We refer the reader to [54, 90] for alternative definitions.

The MDTB-spline basis enjoys several nice properties.

Proposition 2.26. *Assume there exist admissible weights for the space $\mathbb{S}_p^r(\mathcal{M})$. Then, the set $\{N_{k,p} : k = 1, \dots, n\}$ is a basis of the space $\mathbb{S}_p^r(\mathcal{M})$, with the following properties:*

- *local support:*

$$\text{supp}(N_{k,p}) = [u_k, v_k], \quad k = 1, \dots, n;$$

- *non-negative partition of unity:*

$$N_{k,p}(x) \geq 0, \quad k = 1, \dots, n, \quad \sum_{k=1}^n N_{k,p}(x) = 1, \quad x \in [a, b];$$

- *interpolation at the end points:*

$$\begin{aligned} N_{1,p}(a) &= 1, & N_{k,p}(a) &= 0, & k &= 2, \dots, n, \\ N_{n,p}(b) &= 1, & N_{k,p}(b) &= 0, & k &= 1, \dots, n-1. \end{aligned}$$

2.3. Multi-degree Tchebycheffian spline spaces

These properties are of interest in both geometric modeling and isogeometric analysis; they make the set of MDTB-splines $\{N_{k,p} : k = 1, \dots, n\}$ a very appealing basis for the space $\mathbb{S}_p^r(\mathcal{M})$ in those applications. However, using the recurrence relation in (2.25)–(2.26) for their construction is not an option from the practical point of view. As for TB-splines, an alternative, more efficient construction, can be obtained by means of knot insertion; see Remark 2.20 and [54, 115].

3

Isogeometric analysis with Tchebycheffian B-splines

The properties of TB-splines show that TB-splines are plug-to-plug compatible with classical (polynomial) B-splines, making them a viable alternative basis for constructing isogeometric discretization spaces. In this chapter, we demonstrate the use of TB-splines in isogeometric analysis.

Section 3.1 presents the isogeometric Galerkin method based on tensor-product TB-splines and Section 3.2 describes a general road map for the selection of suitable ECT-spaces from a large variety of combinations of polynomial, exponential, and trigonometric functions equipped with a wide spectrum of shape parameters to fully exploit the flexibility of the ECT-spaces. Finally, Section 3.3 presents a series of case studies illustrating the performance of the TB-splines compared to the classical polynomial B-splines.

3.1 Isogeometric TB-spline Galerkin methods

We elaborate the isogeometric Galerkin formulation from Section 1.2 based on Tchebycheffian splines. We focus on TB-splines from the ECT-spaces of the form (2.14).

Here we provide some notation that will be useful in the case studies reported in Section 3.3. We deal with two-dimensional problems, where the bivariate TB-splines on the parametric domain are constructed by taking the tensor-product of univariate TB-splines considering a single ECT-space on the interval $[0, 1]$. Let $\mathcal{M}_1 \times \mathcal{M}_2$ be a rectangular grid in $[0, 1]^2$, $\mathbf{r}_{\mathcal{M}_1}, \mathbf{r}_{\mathcal{M}_2}$ be two smoothness vectors at the corresponding breakpoints, $p_1, p_2 \geq 1$ be two degrees, and w_1, w_2 be two root vectors as in (2.10).

The results in this chapter are published in:

K. Raval, C. Manni, and H. Speleers, *Tchebycheffian B-splines in isogeometric Galerkin methods*, Computer Methods in Applied Mechanics and Engineering **403** (2023), 115648.

Then, the tensor-product Tchebycheffian spline space is given by

$$\left\langle N_{k,p_1} N_{l,p_2} : k = 1, \dots, n_1, l = 1, \dots, n_2 \right\rangle, \quad (3.1)$$

where n_1 and n_2 are the dimensions of the two univariate Tchebycheffian spline spaces. For notational convenience, see also Example 1.11, the basis functions are also shortly indexed as

$$N_i := N_{k,p_1} N_{l,p_2}, \quad i := (l-1)n_1 + k, \quad k = 1, \dots, n_1, \quad l = 1, \dots, n_2.$$

A point in (the closure of) the physical domain Ω is denoted as $\mathbf{x} = (x, y)$ and a point in (the closure of) the parametric domain $\widehat{\Omega}$ as $\hat{\mathbf{x}} = (s, t)$. The geometry map in (1.23) is given by

$$\mathbf{x} = \mathbf{G}(\hat{\mathbf{x}}) = \sum_{i=1}^{n_G} \mathbf{P}_i N_i(\hat{\mathbf{x}}), \quad \mathbf{P}_i \in \mathbb{R}^2, \quad n_G = n_1 n_2, \quad (3.2)$$

while the basis functions in (1.25) are

$$\phi_i(\mathbf{x}) := N_{j_i} \circ \mathbf{G}^{-1}(\mathbf{x}) = N_{j_i}(\hat{\mathbf{x}}),$$

where, in order to incorporate the homogeneous boundary conditions, the tensor-product TB-splines

$$\begin{aligned} N_{k,p_1} N_{l,p_2}, & \quad k = 1, n_1, \quad l = 1, \dots, n_2, \\ N_{k,p_1} N_{l,p_2}, & \quad k = 1, \dots, n_1, \quad l = 1, n_2, \end{aligned}$$

are excluded according to the last property in Proposition 2.14. Thus we have $n_W = (n_1 - 2)(n_2 - 2)$.

3.2 Selection of the ECT-spaces

As mentioned above, we are interested in testing the performance of isogeometric Galerkin methods based on TB-splines from ECT-spaces of the form (2.14). It is clear that such a class provides a large variety of combinations of polynomial, exponential, and trigonometric functions equipped with a wide spectrum of shape parameters. Of course, it is crucial to properly choose the structure of the space (including different kinds of functions) and the related shape parameters to fully exploit the functionality of ECT-spaces. Particularly, there are two aspects that should be considered in such a selection to profit from a non-polynomial structure in the reference ECT-spaces.

- The first is the geometric aspect of the problem based on the geometry mapping at hand. Often, the boundary of the physical domain Ω is built from (arcs of) conic sections. Therefore, it can be exactly described in terms of polynomials, exponential, and trigonometric functions with suitable shape parameters. In the pure isogeometric philosophy, the geometry of the physical domain is exact from the beginning of the procedure and does not change along the process when possible refinements are required. In this perspective, it is then natural to select ECT-spaces (2.14) that allow for an exact representation of the geometry. In its classical formulation, IgA is based on B-splines or NURBS. Even though (arcs of) conic sections can be exactly represented in terms of NURBS, it is not an arc-length parameterization and smooth representations of (closed) conic sections require high degrees. As an example, a C^r NURBS parameterization of a circle requires at least degree $2(r + 1)$; see [2]. Moreover, NURBS behave poorly when dealing with differentiation and integration. On the other hand, spaces of the form (2.14) allow for natural smooth parameterizations of conic sections and their “structure” does not change when dealing with differentiation and integration, in a complete similarity with polynomial spaces; see also Remark 2.11.

- The second, of interest in the numerical treatment of PDEs, is the fact that functions in ECT-spaces of the form (2.14) are fundamental solutions of important ordinary differential operators, meaning that the shape parameters are immediately connected to the associated differential problems. Although the extension to the multivariate case is not straightforward, the use of TB-splines with pieces drawn from suitable ECT-spaces can still be beneficial. As an example, they can lead to a significant reduction, or even a complete removal, of extraneous oscillations in advection-dominated problems, without the need of stabilization techniques. Once again, a proper selection of the shape parameters is crucial but can be driven by the problem setting.

In the next section we present some case studies that show how TB-splines can outperform classical polynomial B-splines in isogeometric Galerkin discretizations of Poisson and advection-diffusion problems. As discussed above, the selection of the underlying ECT-spaces has to be done according to a problem-driven strategy; this will be detailed in each of the case studies.

3.3 Numerical results of IgA with TB-splines

In this section we discuss the performance of the isogeometric Galerkin method based on TB-splines through different case studies. For all the numerical experiments in this chapter we use the following setup.

Bivariate tensor-product Tchebycheffian spline spaces of the form (3.1) are taken as discretization spaces in the isogeometric Galerkin method. The corresponding ECT-spaces in the two directions will be specified in each experiment by writing

$$\mathbb{P}_{p_1}^{\mathcal{W}_1} \otimes \mathbb{P}_{p_2}^{\mathcal{W}_2},$$

where p_1, p_2 are the degrees and $\mathcal{W}_1, \mathcal{W}_2$ are the root vectors as in (2.10). We only consider uniform partitions $\mathcal{M}_1 = \mathcal{M}_2$ with breakpoints

$$x_i = \frac{i}{m}, \quad i = 0, \dots, m,$$

and denote by h the (uniform) distance between two consecutive breakpoints, i.e., $h = \frac{1}{m}$. We assume maximal smoothness at the breakpoints:

$$\mathbf{r}_{\mathcal{M}_1} = \{p_1 - 1, \dots, p_1 - 1\}, \quad \mathbf{r}_{\mathcal{M}_2} = \{p_2 - 1, \dots, p_2 - 1\}.$$

This amounts to univariate Tchebycheffian spline spaces of dimension $n_1 = m + p_1$ and $n_2 = m + p_2$, respectively.

We focus on the general advection-diffusion problem of the form in Equation (1.26). For simplicity, we fix the diffusion coefficient $\kappa = 1$ and reaction coefficient $c = 0$ in all the case studies.

As usual, homogeneous boundary conditions are satisfied pointwise exactly. In the non-homogeneous case, the boundary function is approximated in the underlying Tchebycheffian spline space by a suitable approximation strategy (e.g., least-squares approximation or quasi-interpolation) and subsequently the reduction to the homogeneous case is considered, so dealing again with a special instance of the problem (1.19).

For an accurate numerical approximation of the integrals needed in the construction of the matrix and the vector in (1.22), we use element-wise Gaussian quadrature rules. The selection of quadrature points for discretizations based on TB-splines is not trivial and quadrature rules of higher order compared to the algebraic polynomial splines are often required. In the presented case studies we use $3p$ quadrature points in each element for TB-splines only involving trigonometric functions (see Sections 3.3.1 and 3.3.2), and $5p$ quadrature points in each element when also exponential functions with large shape parameters are considered (see Sections 3.3.3 to 3.3.5).

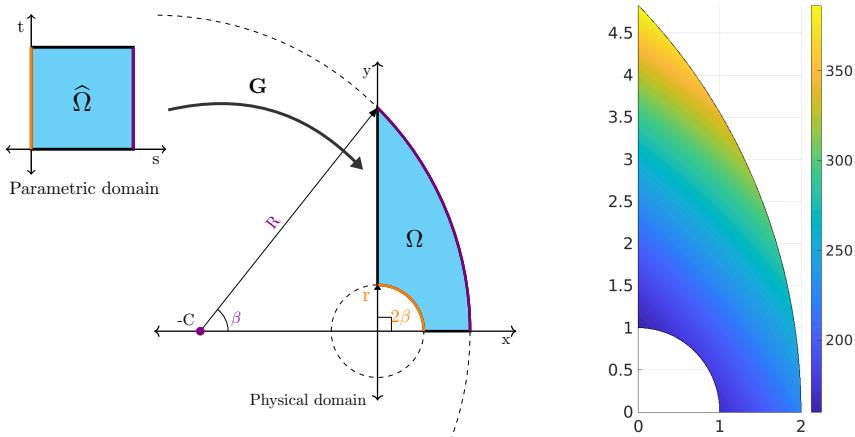


Figure 3.1: Case study 1. Left: Geometry map presented in (3.3) as a combination of curves with phases β and 2β . Right: Plot of the approximate solution obtained by using tensor-product TB-splines identified by $\mathbb{P}_4 \otimes \mathbb{P}_4^{(i\frac{\pi}{4}, -i\frac{\pi}{4}, i\frac{\pi}{2}, -i\frac{\pi}{2})}$ with $m = 8$ and dof = 144.

3.3.1 Case study 1: Poisson problem on a domain bounded by circular arcs

In this case study we address a Poisson problem by considering (1.26) with $\kappa = 1$, $\mathbf{a} = \mathbf{0}$, $c = 0$ and f obtained from the exact solution

$$u(x, y) = 5((x + 4)^2 + (y + 3)^2) + xy.$$

The parametric domain $\hat{\Omega} = (0, 1)^2$ is mapped to the physical domain Ω through the geometry map given by

$$\begin{pmatrix} x \\ y \end{pmatrix} = \mathbf{G} \begin{pmatrix} s \\ t \end{pmatrix} = (1 - s)r \begin{pmatrix} \cos(2\beta t) \\ \sin(2\beta t) \end{pmatrix} + s \begin{pmatrix} -C + R \cos(\beta t) \\ R \sin(\beta t) \end{pmatrix}, \quad (3.3)$$

with

$$r = 1, \quad \beta = \frac{\pi}{4}, \quad C = \frac{2r \cos(\beta)}{1 - \cos(\beta)}, \quad R = C + 2r,$$

which leads to the domain depicted in Figure 3.1, left. It is evident that the geometry can be exactly represented in the form (3.2) by considering in the parametric t -direction Tchebycheffian splines identified by an ECT-space that contains the trigonometric functions with phases β and 2β . Hence we

consider in the t -direction the space of degree $p_2 = 4$,

$$\mathbb{P}_4^{(i\beta, -i\beta, i2\beta, -i2\beta)} = \left\langle 1, \cos\left(\frac{\pi}{4}t\right), \sin\left(\frac{\pi}{4}t\right), \cos\left(\frac{\pi}{2}t\right), \sin\left(\frac{\pi}{2}t\right) \right\rangle.$$

The Tchebycheffian Bernstein basis of this ECT-space is illustrated in Figure 3.2.

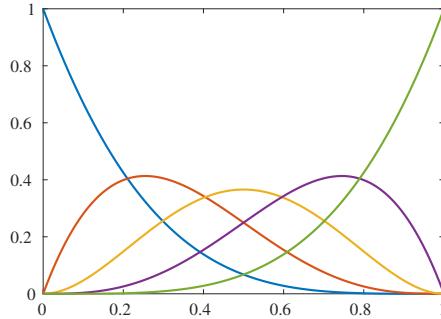


Figure 3.2: Case study 1. Tchebycheffian Bernstein basis of $\mathbb{P}_4^{(i\beta, -i\beta, i2\beta, -i2\beta)}$ on the interval $[0, 1]$.

From Example 2.18 we know that the corresponding Tchebycheffian spline space admits a TB-spline basis on any partition of the interval $[0, 1]$ since $\beta < \pi$. In the s -direction, we simply consider algebraic polynomials of degree $p_1 = 4$, i.e.,

$$\mathbb{P}_4 = \left\langle 1, s, s^2, s^3, s^4 \right\rangle.$$

This gives the tensor-product space

$$\mathbb{P}_4 \otimes \mathbb{P}_4^{(i\frac{\pi}{4}, -i\frac{\pi}{4}, i\frac{\pi}{2}, -i\frac{\pi}{2})}.$$

To obtain the control coefficients \mathbf{P}_i of the geometry in the form (3.2) we simply do interpolation on the grid obtained by the tensor-product of the Greville points of polynomial B-splines of the same degree 4. We remark that this geometry cannot be exactly represented by using generalized polynomial B-splines (see [78]) with pieces in ECT-spaces of the form (2.13).

We approximate the L^∞ norm of the error by sampling the approximate and exact solutions on a uniform grid in the parametric domain consisting of 501 points along each direction. Table 3.1 shows the L^∞ error obtained for different levels of refinements, starting with a very coarse

mesh of only one interval in each parametric direction, comparing the TB-splines against the classical B-splines for spaces of the same dimension. We see an improvement of about a factor 3.5 that we achieve just by exactly representing the geometry using TB-splines. The contour plot of the approximate solution partitioning the domain in 8×8 elements is illustrated in Figure 3.1, right.

m	dof	$\mathbb{P}_4 \otimes \mathbb{P}_4^{(i\frac{\pi}{4}, -i\frac{\pi}{4}, i\frac{\pi}{2}, -i\frac{\pi}{2})}$	$\mathbb{P}_4 \otimes \mathbb{P}_4$
1	25	7.9294×10^{-4}	5.0629×10^{-3}
2	36	3.4854×10^{-4}	5.9308×10^{-4}
4	64	2.9655×10^{-5}	8.4760×10^{-5}
8	144	8.3556×10^{-7}	2.7812×10^{-6}
16	400	2.6336×10^{-8}	9.2305×10^{-8}
32	1296	8.4881×10^{-9}	2.9782×10^{-9}

Table 3.1: Case study 1. Comparison of the L^∞ error of tensor-product TB-splines and B-splines of degrees $p_1 = p_2 = 4$ for different numbers of intervals m in each direction.

3.3.2 Case study 2: Poisson problem on a symmetric domain bounded by circular arcs

Here we focus again on a Poisson problem by considering (1.26) with $\kappa = 1$, $\mathbf{a} = \mathbf{0}$, $\mathbf{c} = 0$ and \mathbf{f} obtained from the exact solution

$$u(x, y) = 4((x + 4)^2 + y^2) + x(y - 4),$$

to exhibit the evolution of error under uniform refinement and verify the consistency of the TB-splines for optimal convergence. The geometry map from the parametric domain $\hat{\Omega} = (0, 1)^2$ to the physical domain Ω is given by

$$\begin{pmatrix} x \\ y \end{pmatrix} = \mathbf{G} \begin{pmatrix} s \\ t \end{pmatrix} = (1 - s)r \begin{pmatrix} \cos(\beta t) \\ \sin(\beta t) \end{pmatrix} + sR \begin{pmatrix} C + \cos(2\beta t - \gamma) \\ C + \sin(2\beta t - \gamma) \end{pmatrix}, \quad (3.4)$$

with

$$r = 1, \quad R = 2, \quad C = 1/\sqrt{2}, \quad \gamma = \frac{\pi}{4}, \quad \beta = \frac{\pi}{2},$$

which leads to the domain depicted in Figure 3.3, left. Similar to the previous case study, the geometry consists of two arcs with phases β and 2β ,

hence a suitable selection of the ECT-space in the t -direction is

$$\mathbb{P}_{p_2}^{(i\beta, -i\beta, \dots, iq\beta, -iq\beta)} = \left\langle 1, \cos(\beta t), \sin(\beta t), \dots, \cos(q\beta t), \sin(q\beta t) \right\rangle, \quad (3.5)$$

$$p_2 = 2q \geq 4.$$

The Tchebycheffian Bernstein basis of this ECT-space is illustrated in Figure 3.4 for $p_2 = 6$. From Example 2.18 we know that the corresponding Tchebycheffian spline space admits a TB-spline basis on any partition of the interval $[0, 1]$ since $\beta < \pi$. In the s -direction, we simply consider algebraic polynomials of the same degree as in the t -direction. This gives the tensor-product space

$$\mathbb{P}_{p_1} \otimes \mathbb{P}_{p_2}^{(i\frac{\pi}{2}, -i\frac{\pi}{2}, \dots, iq\frac{\pi}{2}, -iq\frac{\pi}{2})}, \quad p_1 = p_2 = p = 2q.$$

Again, this geometry cannot be exactly represented by using generalized polynomial B-splines (see [78]) with pieces in ECT-spaces of the form (2.13).

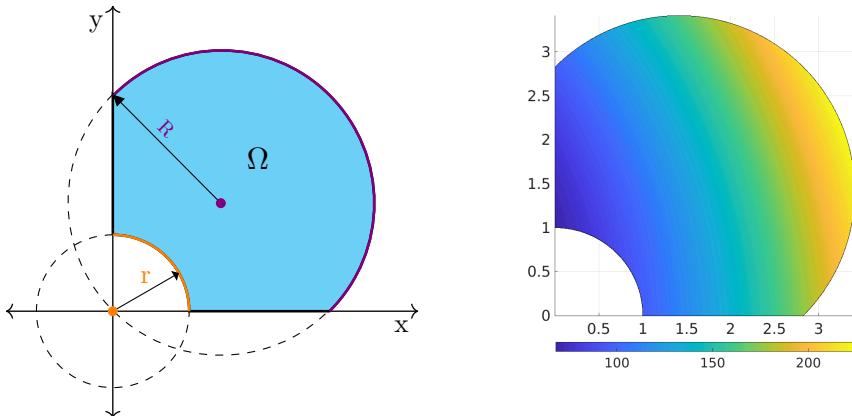


Figure 3.3: Case study 2. Left: Geometry map presented in (3.4) as a combination of curves with phases β and 2β . Right: Plot of the approximate solution obtained by using tensor-product TB-splines identified by $\mathbb{P}_6 \otimes \mathbb{P}_6^{(i\beta, -i\beta, i2\beta, -i2\beta, i3\beta, -i3\beta)}$, $\beta = \frac{\pi}{2}$ with $m = 8$ and $dof = 196$.

Figure 3.5 illustrates the convergence of the error in L^∞ norm, which has been approximated by sampling the approximate and exact solutions on a uniform grid in the parametric domain consisting of 501 points along each direction. The plot clearly shows that the approximate solution obtained by TB-splines exhibits the optimal convergence of order $p + 1$ as

3.3. Numerical results of IgA with TB-splines

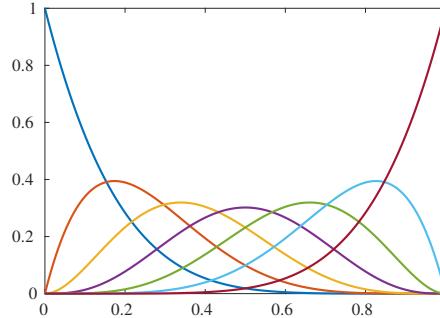


Figure 3.4: Case study 2. Tchebycheffian Bernstein basis of $\mathbb{P}_6^{(i\beta, -i\beta, i2\beta, -i2\beta, i3\beta, -i3\beta)}$, $\beta = \frac{\pi}{2}$, on the interval $[0, 1]$.

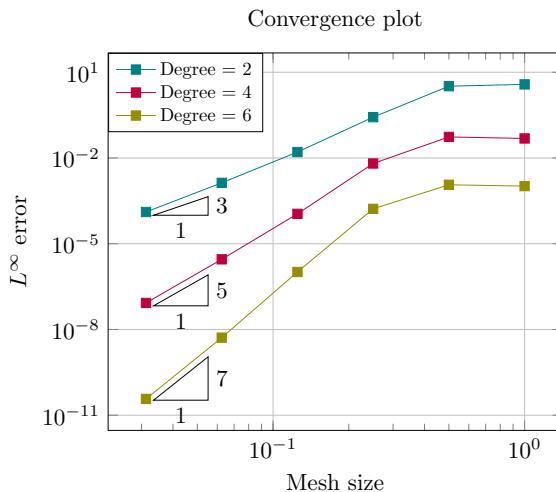


Figure 3.5: Case study 2. Convergence plot of the L^∞ error of tensor-product TB-splines identified by $\mathbb{P}_p \otimes \mathbb{P}_p^{(i\beta, -i\beta, i2\beta, -i2\beta, \dots, iq\beta, -iq\beta)}$, $\beta = \frac{\pi}{2}$, for $p = 2q$, $q = \{1, 2, 3\}$, and different resolution of the mesh $h = 1, \frac{1}{2}, \frac{1}{2^2}, \dots, \frac{1}{2^5}$.

classical polynomial B-splines in agreement with the theoretical expectations; see Theorem 2.23. This establishes the reliability of the Tchebycheffian spline spaces from the accuracy point of view. The contour plot of the approximate solution with $p = 6$ on a mesh consisting of 8×8 elements is presented in Figure 3.3, right.

3.3.3 Case study 3: Advection-diffusion problem with radial advection flow

In the experiments presented so far, the selection of the ECT-spaces has been driven by their geometrical features. Now we will also take into account some analytical features of the problem. Let us consider the advection-diffusion problem (1.26) with

$$\kappa = 1, \quad \mathbf{a}(x, y) := \mathbf{a} \begin{pmatrix} \frac{x}{\sqrt{x^2+y^2}} \\ \frac{y}{\sqrt{x^2+y^2}} \end{pmatrix}, \quad \mathbf{a} = 100, \quad c = 0, \quad f(x, y) = 1, \quad (3.6)$$

and homogeneous boundary conditions, on the same geometry used in the previous case study given by the geometry map (3.4).

It is well known that for problems with high Péclet number ($\text{Pe}_g \gg 1$) approximate solutions belonging to piecewise polynomial spaces tend to exhibit spurious oscillations unless the discretization is fine enough. The above mentioned issue can be efficiently addressed by turning to Tchebycheffian splines built from exponential functions with suitable parameters. Their use for solving advection-dominated advection-diffusion problems reduces, and sometimes even eliminates, the spurious oscillations while still providing an accurate localization of the sharp layer(s) without demanding extremely fine meshes and so keeping the number of degrees of freedom (dof) reasonably low.

For an advection-dominated system with advection flow in the radial direction as in (3.6), we can exploit exponential functions in the parametric s -direction with suitable shape parameters to represent the sharp layer generated near the homogeneous boundary. To construct effective Tchebycheffian splines with exponential functions in the s -direction, we consider the ECT-space

$$\mathbb{P}_{p_1}^{(\alpha_1, \alpha_2, \dots, \alpha_\ell)} = \left\langle 1, s, \dots, s^{p_1-\ell}, e^{\alpha_1 s}, \dots, e^{\alpha_\ell s} \right\rangle, \quad p_1 \geq \ell \geq 1,$$

where for $i = 1, \dots, \ell$,

$$\alpha_i := \begin{cases} \frac{1}{2}(\alpha_{max} + \alpha_{min}), & \ell = 1, \\ \alpha_{min} + \frac{i-1}{\ell-1}(\alpha_{max} - \alpha_{min}), & \ell \geq 2, \end{cases} \quad (3.7)$$

and

$$\alpha_{min} := \mathbf{a}(\sqrt{2R} - r), \quad \alpha_{max} := \mathbf{a}(2R - r).$$

The governing factors in the selection of the shape parameters here are the global Péclet number (1.27), which is $\text{Pe}_g = \mathbf{a}$, and the range of distances traveled by the advection flow in the radial direction. Figure 3.6

3.3. Numerical results of IgA with TB-splines

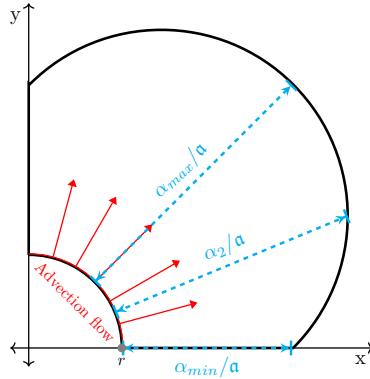


Figure 3.6: Case study 3. Illustration of the selection strategy of the shape parameters $\{\alpha_i : i = 1, \dots, \ell\}$ of the exponential functions to accommodate for different distances traveled by the advection flow for $\ell = 3$ in (3.7). The extremes of the chosen shape parameters $\alpha_1 = \alpha_{min} = a(\sqrt{2R} - r)$ and $\alpha_3 = \alpha_{max} = a(2R - r)$ are evidently related to the maximum and minimum radial distances of the domain, where a is the modulus of the advection coefficient.

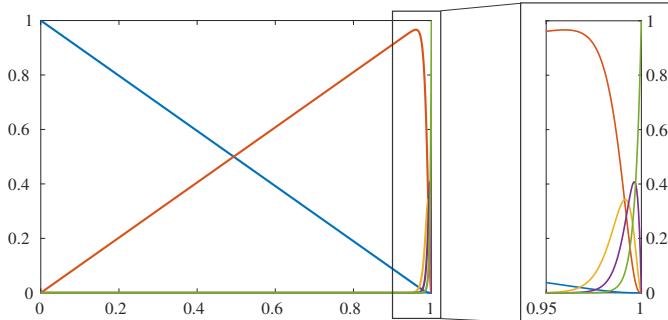


Figure 3.7: Case study 3. Tchebycheffian Bernstein basis of $\mathbb{P}_4^{(\alpha_1, \alpha_2, \alpha_3)}$ on the interval $[0, 1]$, with a magnified view of the functions on the interval $[0.95, 1]$.

illustrates this approach for the selection of the shape parameters by considering $\ell = 3$ different radial distances traveled by the advection flow. The Tchebycheffian Bernstein basis of this ECT-space is illustrated in Figure 3.7 for $p_1 = 4$. From Example 2.16 we know that the corresponding Tchebycheffian spline space admits a TB-spline basis on any partition of the interval $[0, 1]$.

As discussed in Section 3.3.2, to exactly reproduce the geometry, we consider in the t -direction the ECT-space (3.5) of degree $p_2 = 2q \geq 4$,

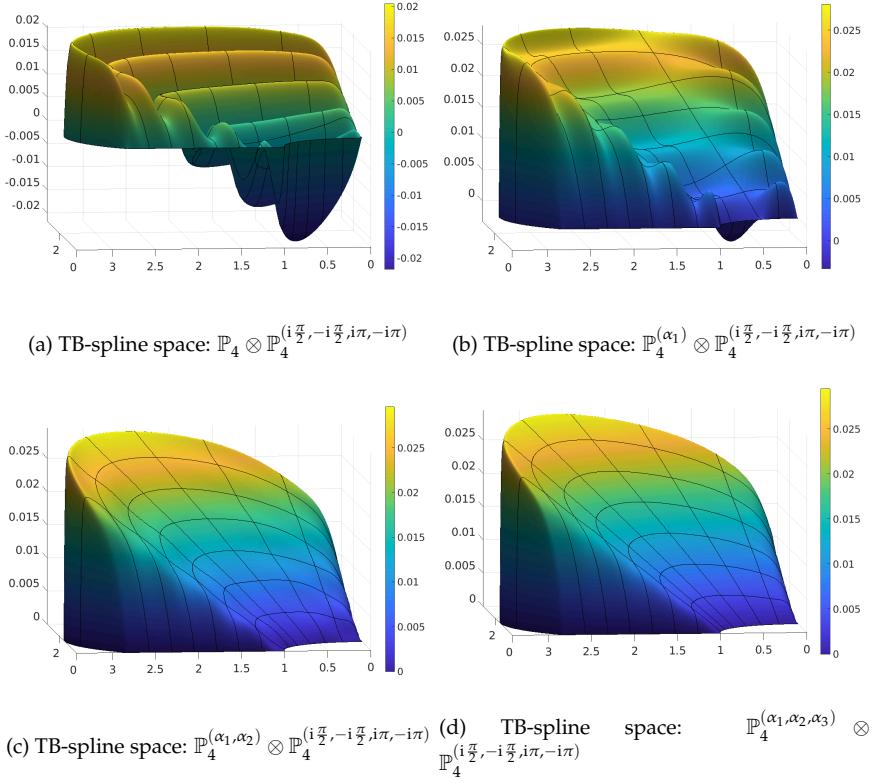


Figure 3.8: Case study 3. Plots of the approximate solution to the advection-diffusion problem with global Péclet number $\text{Pe}_g = 100$, obtained by using tensor-product TB-splines with $p_1 = p_2 = 4$, $m = 8$, dof = 144. Increasing the number of exponential functions with shape parameters $\{\alpha_i : i = 1, \dots, \ell\}$ according to (3.7) in the TB-spline space results into a reduction of oscillations in the solution.

which allows for a Tchebycheffian spline space equipped with a TB-spline basis on the interval $[0, 1]$. The resulting tensor-product space is

$$\mathbb{P}_{p_1}^{(\alpha_1, \alpha_2, \dots, \alpha_\ell)} \otimes \mathbb{P}_{p_2}^{(i\frac{\pi}{2}, -i\frac{\pi}{2}, \dots, iq\frac{\pi}{2}, -iq\frac{\pi}{2})}, \quad p_2 = 2q,$$

with both spaces of the form (2.14).

In Figure 3.8 we present some results obtained by using tensor-product TB-splines of degrees $p_1 = p_2 = 4$ on a mesh consisting of 8×8 elements. To exhibit the effect of the presence of exponential functions in the TB-spline space for addressing the advection-dominant problem (1.26) with

(3.6), we vary the number of shape parameters in the selection from (3.7). The outcome of using only algebraic polynomial splines (so without any exponential functions) in the s -direction is shown in Figure 3.8a. Even though the geometry is described exactly with the trigonometric functions present in the t -direction, the absence of the exponential functions results in spurious oscillations in the solution. However, taking one exponential function in the TB-spline space somewhat reduces the oscillations as shown in Figure 3.8b, and taking two exponential functions in the space almost removes all the over- and undershoots as shown in Figure 3.8c. Following the trend, with three exponential functions in the TB-spline space, the oscillations are completely eliminated from the solution (at least visually) as illustrated in Figure 3.8d.

3.3.4 Case study 4: Advection–diffusion problem with tangential advection flow

The ECT-spaces used in the previous case studies do not exploit the full functionality of mixing exponential and trigonometric functions with polynomials, as in (2.14). Hence, now we consider a problem with analytical and geometrical features in the same parametric direction. Let us consider the advection-diffusion problem (1.26) with

$$\kappa = 1, \quad \mathbf{a}(x, y) := \mathbf{a} \left(\begin{array}{c} \frac{-y}{\sqrt{x^2+y^2}} \\ \frac{x}{\sqrt{x^2+y^2}} \end{array} \right), \quad \alpha = 100, \quad c = 0, \quad f(x, y) = 1, \quad (3.8)$$

and homogeneous boundary conditions, on a classical quarter of an annulus given by the geometry map

$$\begin{pmatrix} x \\ y \end{pmatrix} = \mathbf{G} \begin{pmatrix} s \\ t \end{pmatrix} = ((1-s)r + sR) \begin{pmatrix} \cos(\frac{\pi}{2}t) \\ \sin(\frac{\pi}{2}t) \end{pmatrix}, \quad r = 1, \quad R = 2. \quad (3.9)$$

which leads to the domain depicted in Figure 1.6, right.

To build suitable TB-spline spaces for the above problem, we need to consider both the geometry map in (3.9) and the advection coefficient \mathbf{a} in (3.8). The geometry map requires trigonometric functions in the t -direction. Moreover, since we have a tangential flow along the t -direction, see Figure 3.9, it is natural to add exponential functions in that direction as well to accurately represent the sharp layer. The ECT-space in the t -

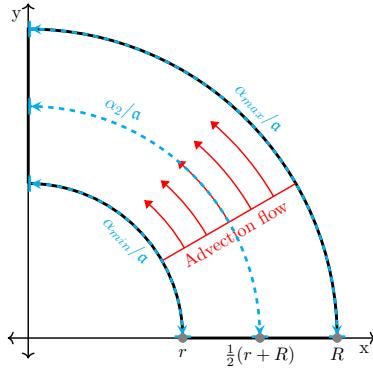


Figure 3.9: Case study 4. Illustration of the selection strategy of the shape parameters $\{\alpha_i : i = 1, \dots, \ell\}$ of the exponential functions to accommodate for different distances traveled by the advection flow for $\ell = 3$ in (3.11). The extremes of the chosen shape parameters $\alpha_1 = \alpha_{min} = \alpha \frac{\pi}{2} r$ and $\alpha_3 = \alpha_{max} = \alpha \frac{\pi}{2} R$ are evidently related to the two extreme arc lengths of the domain, where α is the modulus of the advection coefficient.

direction can be chosen as

$$\mathbb{P}_{p_2}^{(\alpha_1, \alpha_2, \dots, \alpha_\ell, i\frac{\pi}{2}, -i\frac{\pi}{2})} = \left\langle 1, t, \dots, t^{p_2-2-\ell}, e^{\alpha_1 t}, \dots, e^{\alpha_\ell t}, \cos\left(\frac{\pi}{2}t\right), \sin\left(\frac{\pi}{2}t\right) \right\rangle, \\ p_2 \geq \ell + 2 \geq 3, \quad (3.10)$$

where, similar to (3.7), for $i = 1, \dots, \ell$,

$$\alpha_i := \begin{cases} \frac{1}{2}(\alpha_{max} + \alpha_{min}), & \ell = 1, \\ \alpha_{min} + \frac{i-1}{\ell-1}(\alpha_{max} - \alpha_{min}), & \ell \geq 2, \end{cases} \quad (3.11)$$

and

$$\alpha_{min} := \alpha \frac{\pi}{2} r, \quad \alpha_{max} := \alpha \frac{\pi}{2} R.$$

The governing factors in the selection of the shape parameters here are again the global Péclet number (1.27), which is $\text{Pe}_g = \alpha$, and the range of distances traveled by the advection flow along circular arcs. Figure 3.9 illustrates this approach for the selection of the shape parameters by considering $\ell = 3$ different distances along arcs traveled by the advection flow. The Tchebycheffian Bernstein basis of this ECT-space is illustrated in Figure 3.10 for $p_2 = 6$. In the s -direction, we simply consider algebraic

3.3. Numerical results of IgA with TB-splines

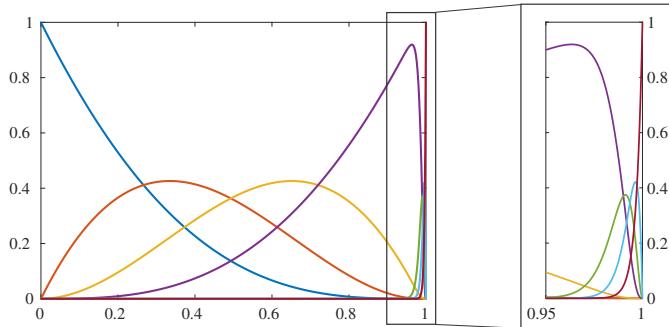


Figure 3.10: Case study 4. Tchebycheffian Bernstein basis of $\mathbb{P}_6^{(\alpha_1, \alpha_2, \alpha_3, i\frac{\pi}{2}, -i\frac{\pi}{2})}$ on the interval $[0, 1]$, with a magnified view of the functions on the interval $[0.95, 1]$.

polynomials of degree $p_1 = 2$. The resulting tensor-product space is

$$\mathbb{P}_{p_1} \otimes \mathbb{P}_{p_2}^{(\alpha_1, \alpha_2, \dots, \alpha_\ell, i\frac{\pi}{2}, -i\frac{\pi}{2})}.$$

For the sake of comparison, we also consider in the t -direction the generalized polynomial space obtained by adding to the polynomial space the pair of trigonometric functions necessary for an exact representation of the geometry, see (2.13), i.e.,

$$\mathbb{P}_{p_2}^{(i\frac{\pi}{2}, -i\frac{\pi}{2})} = \left\langle 1, t, \dots, t^{p_2-2}, \cos\left(\frac{\pi}{2}t\right), \sin\left(\frac{\pi}{2}t\right) \right\rangle, \quad (3.12)$$

so that the resulting tensor-product space used for the comparison is

$$\mathbb{P}_{p_1} \otimes \mathbb{P}_{p_2}^{(i\frac{\pi}{2}, -i\frac{\pi}{2})}.$$

Note that both the ECT-spaces in (3.10) and (3.12) are of the form (2.14). From Example 2.17 we know that the corresponding Tchebycheffian spline spaces admit a TB-spline basis on any partition of the interval $[0, 1]$.

In Figure 3.11 we present some results obtained by using tensor-product TB-splines of degrees $p_1 = 2$ and $p_2 = 6$ on a mesh consisting of 6×6 elements. To exhibit the effect of the presence of exponential functions in the TB-spline space for addressing the advection-dominant problem (1.26) with (3.8), we vary the number of shape parameters in the selection from (3.11). We can see in Figure 3.11a that the GB-spline space with only trigonometric functions leads to spurious oscillations in the solution. However, taking just one exponential function in the TB-spline space al-

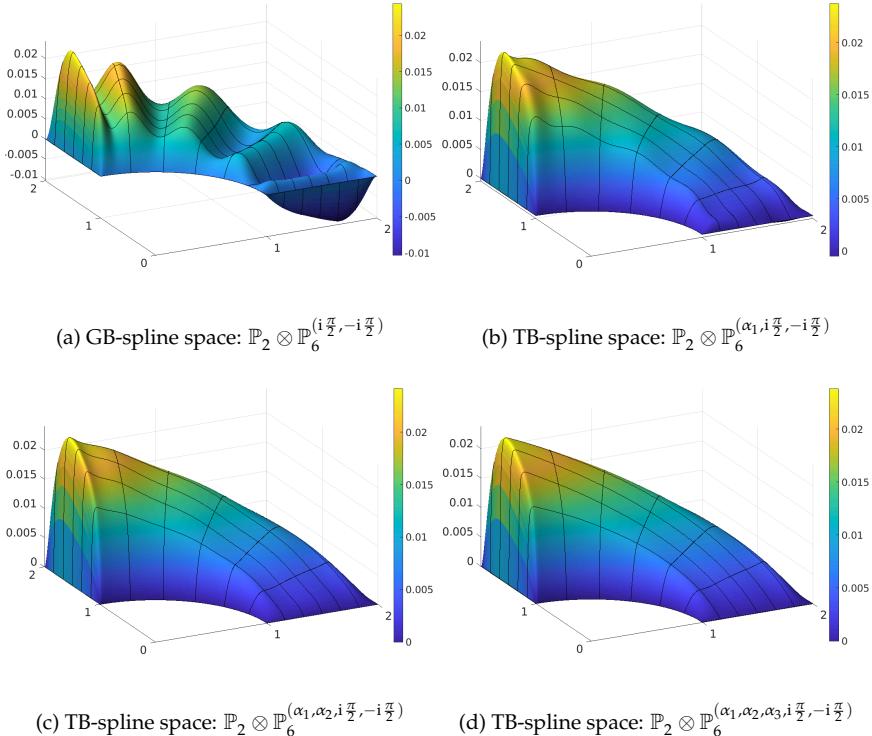


Figure 3.11: Case study 4. Plots of the approximate solution to the advection-diffusion problem with global Péclet number $\text{Pe}_g = 100$, obtained by using tensor-product TB-splines with $p_1 = 2$, $p_2 = 6$, $m = 6$, dof = 96. Increasing the number of exponential functions with different shape parameters $\{\alpha_i : i = 1, \dots, \ell\}$ according to (3.11) in the TB-spline space results into a reduction of oscillations in the solution.

ready substantially reduces the oscillations as shown in Figure 3.11b, and with two exponential functions all the over- and undershoots almost disappear as shown in Figure 3.11c. The trend continues to follow and increasing the number of exponential functions to 3 ultimately gives us a solution with no oscillations (at least visually) as illustrated in Figure 3.11d.

3.3.5 Case study 5: Advection-diffusion problem with internal sharp layer on a square

In the previous two case studies the advection flow travels exactly along one of the parametric directions. Here we consider a classical benchmark

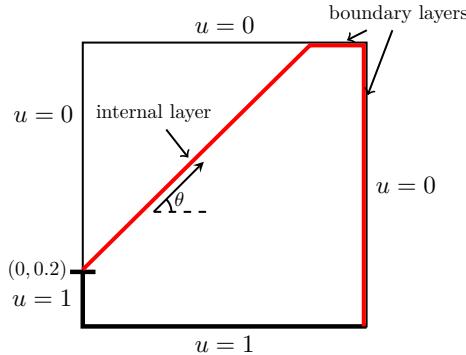


Figure 3.12: Case study 5. The domain with the Dirichlet boundary conditions, the sharp internal layer (red) generated along the advection flow at angle θ and the boundary layers (red).

problem with advection flow not parallel to any parametric direction [55, 79, 117].

We want to solve the advection-diffusion problem (1.26) on a square, hence the geometry map is just the identity, with

$$\kappa = 1, \quad \mathbf{a} = \mathbf{a}(\cos(\theta), \sin(\theta))^T, \quad \theta = \frac{\pi}{4}, \quad \mathbf{a} = 10^4, \quad c = 0, \quad f(x, y) = 0,$$

and the discontinuous Dirichlet boundary conditions as shown in Figure 3.12. The solution exhibits sharp boundary and internal layers depicted in red in the same figure. In particular, the sudden jump in the boundary conditions at the point $(0, 0.2)$ generates an inner sharp layer aligned with the advection flow direction identified by $(\cos(\theta), \sin(\theta))$.

As already mentioned in Section 3.3.3, since the problem has a large global Péclet number, namely $\mathbf{Pe}_g = \mathbf{a} = 10^4$, approximate solutions belonging to piecewise polynomial spaces exhibit spurious oscillations until the discretization is fine enough to resolve the sharp layers featured by the exact solution. To overcome this issue, it is common to rely on stabilization methods such as the SUPG and GLS method; see [23, 56]. Typically, stabilization methods remove spurious oscillations but at the same time “smooth out” the layers featured by the exact solution. Their efficiency deeply depends on a careful choice of some parameters appearing in the various stabilization methods.

Tchebycheffian splines can offer a flexible alternative for the treatment of the above advection-dominated problem without the need for stabilization. To catch the sharp layers, it is natural to consider Tchebycheffian spline spaces that contain exponential functions with suitable parameters.

Since the advection flow velocity is constant here, we can simply take its two components as parameters in the two (parameter) directions:

$$\begin{aligned}\mathbb{P}_{p_1}^{(\alpha \cos(\theta))} &= \left\langle 1, x, \dots, x^{p_1-1}, e^{\alpha \cos(\theta)x} \right\rangle, \\ \mathbb{P}_{p_2}^{(\alpha \sin(\theta))} &= \left\langle 1, y, \dots, y^{p_2-1}, e^{\alpha \sin(\theta)y} \right\rangle.\end{aligned}\quad (3.13)$$

Both these spaces are of the form (2.14) and, since they just contain polynomial and exponential functions, from Example 2.16 we know that the corresponding Tchebycheffian spline spaces admit a TB-spline basis on any partition of the interval $[0, 1]$. Note that, since the physical domain is simply a square, the parameter selection in (3.13) is not governed by the geometry.

It is important to remark that a proper treatment of the boundary conditions is imperative due to its effect on the approximate solution. In order to avoid oscillations along the boundary, instead of using a least squares approximation as in the previous case studies, we have considered quasi-interpolation. More precisely, in order to exploit the shape-preserving property of the (T)B-spline representation, following the approach in [55], we approximate the boundary function through the Schoenberg operator introduced in Example 1.5 and Section 2.2.2 (see [75, 103]) along each edge. In other words, we consider the linear combination of the boundary TB-splines whose coefficients are obtained by evaluating the boundary function at the corresponding Greville abscissae (under the assumptions that linear polynomials belong to the ECT-space of interest).

In Figure 3.13a we illustrate the approximate solution obtained by using tensor-product TB-splines, identified by (3.13), of degrees $p_1 = p_2 = 6$ on a mesh consisting of 50×50 elements. If we use classical tensor-product B-splines of the same degrees on a mesh of only 50×50 elements, then the result will exhibit huge spurious oscillations near the layers. Hence, it makes sense to compare the stabilized solution of the B-splines against the TB-splines (without stabilization). Figure 3.13b shows the result obtained by using tensor-product B-splines with SUPG stabilization, where the stabilization constant is given by

$$\tau = \frac{h_a}{2\|\mathbf{a}\|}, \quad \text{with} \quad h_a = \frac{h}{\max\{\cos(\theta), \sin(\theta)\}}. \quad (3.14)$$

Note that h_a is the element length in the direction of the advection flow; see [55] and [39, Section 9.3.2]. As a reference solution we are considering the one obtained by classical tensor-product B-splines of the same degrees on a very fine mesh of 1500×1500 elements, as shown in Figure 3.13c.

Table 3.2 collects the values of maximal over- and undershoot in the

3.3. Numerical results of IgA with TB-splines

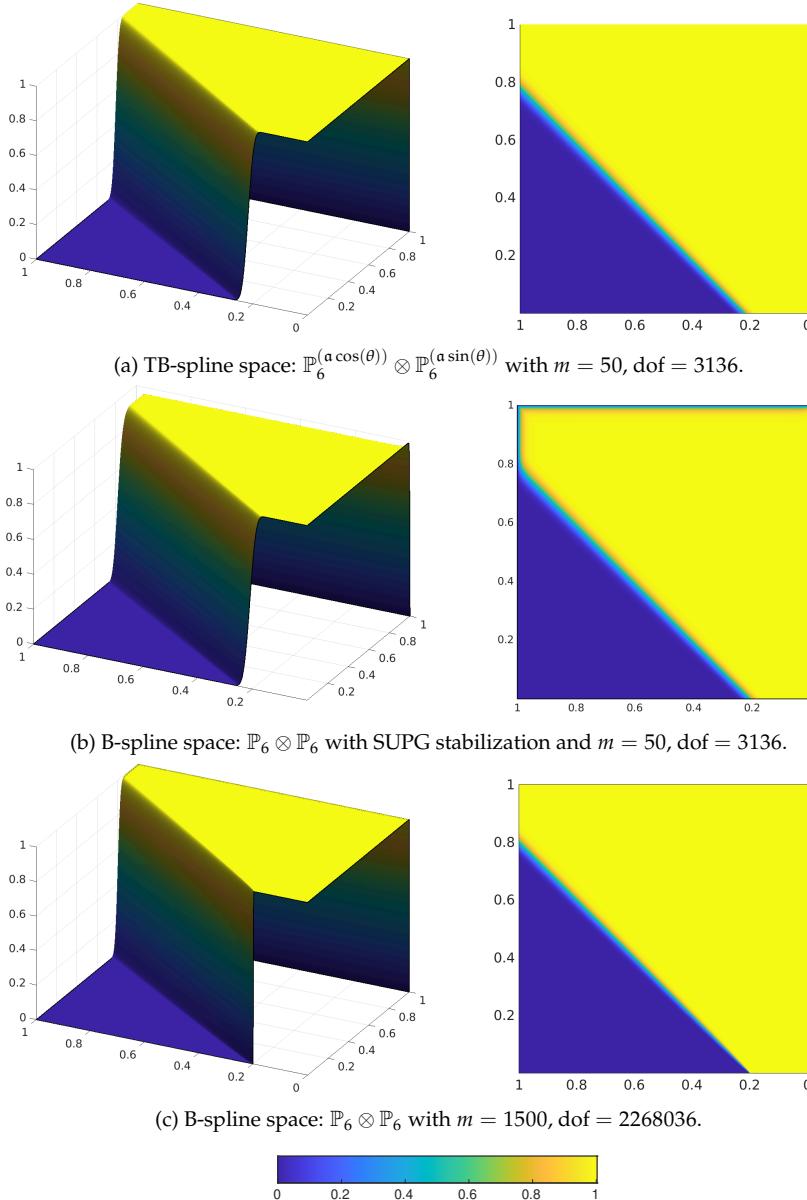


Figure 3.13: Case study 5. Plots of the approximate solution to the advection-diffusion problem with advection skew at $\theta = 45^\circ$ and global Péclet number $\mathbf{Pe}_g = 10^4$, obtained by using tensor-product spline spaces with $p_1 = p_2 = 6$; the number of elements m in each direction and the dof are mentioned in the sub-captions.

TB-spline space	SUPG	m	dof	max	min
$\mathbb{P}_6^{(\alpha \cos(\theta))} \otimes \mathbb{P}_6^{(\alpha \sin(\theta))}$	No	50	3136	1.0016	-1.6385×10^{-3}
$\mathbb{P}_6 \otimes \mathbb{P}_6$	Yes	50	3136	1.0009	-9.4671×10^{-4}
$\mathbb{P}_6 \otimes \mathbb{P}_6$	No	1500	2268036	1.0022	-1.8761×10^{-6}

Table 3.2: Case study 5. Comparison of maximum and minimum of the TB-spline solution with exponential functions in the space against the polynomial B-spline solution with SUPG stabilization on the same mesh and the polynomial B-spline solution on a really high number of intervals with $p_1 = p_2 = 6$ for $\alpha = 10^4$.

neighborhood of the layers evaluated on a uniform grid of 3001 points along each direction for the three considered spline spaces, together with the used number of elements and the total number of dof.

When comparing the TB-spline solution with the stabilized B-spline solution, we observe that both approaches undoubtedly eliminate the spurious oscillations. However, stabilization produces a too smooth solution and the localization of the internal and boundary layers is less accurate than in the case of the TB-spline solution. On the other hand, the unstabilized B-spline solution on a very fine mesh is able to localize accurately the layers but at a very high cost (i.e., a large number of dof). In conclusion, TB-splines on coarse meshes are able to predict the layers precisely, resulting in very few required dof while still giving stable results and eliminating the need of stabilization.

3.4 Conclusive remarks

Tchebycheffian splines are smooth piecewise functions whose pieces are drawn from ECT-spaces, a natural generalization of algebraic polynomial spaces. Under suitable assumptions, they admit a basis of functions equipped with all the nice properties of classical polynomial B-splines, the so-called TB-splines. Theoretically, TB-splines are completely plug-to-plug compatible with polynomial B-splines. Unfortunately, TB-splines in their most generality still lack practical algorithms for their evaluation and manipulation. For this reason, despite their great potential, they did not gain much attention in applications so far.

On the contrary, for the restricted class of TB-splines with pieces belonging to ECT-spaces that are null-spaces of constant-coefficient linear differential operators, efficient manipulation routines have been developed recently and made publicly available in a Matlab toolbox [115], so that they can be easily incorporated in any software library supporting polynomial B-splines to enrich its capability. This subclass of TB-splines

already provides a large variety of combinations of polynomial, exponential, and trigonometric functions equipped with a wide spectrum of shape parameters. They allow for an exact representation of profiles of interest in applications, they behave nicely with respect to differentiation and integration and, by construction, they include fundamental solutions of interesting differential operators. Therefore, they offer a valid alternative to classical polynomial B-splines and NURBS in Galerkin isogeometric methods.

It turns out that TB-splines can outperform polynomial B-splines whenever appropriate problem-driven selection strategies for the underlying ECT-spaces are applied. Typically, there are two aspects that can be considered to properly select and profit from a non-polynomial structure in the reference ECT-spaces: the geometric aspect of the problem based on an exact geometry mapping by using suitable ECT-spaces, and the analytical behavior of the solution that can be better captured, avoiding spurious oscillations, by including in the discretization spaces functions related to fundamental solutions of the given differential operator.

The proposed setting is promising but also presents some theoretical, computational, and numerical issues that require more attention and deserve further investigation in future research.

- The existence of a TB-spline basis is ensured when the different pieces are taken from a $(p + 1)$ -dimensional ECT-space on the entire interval $[a, b]$ which contains constants and its derivative space is a p -dimensional ECT-space on the same interval. Since our interest is confined to null-spaces of constant-coefficient linear differential operators, we can weaken this restriction and we only need to check that the critical length of the derivative space we are dealing with exceeds $\xi_{k+p} - \xi_{k+1}$ for $k = 1, \dots, n$. Besides the results concerning the critical length recalled in Sections 2.1.1 and 2.1.2, we remark that a numerical procedure to estimate the critical length for a given ECT-space is provided in [6]. In the more general setting where the various pieces of the Tchebycheffian splines are drawn from different ECT-spaces, the existence of a TB-spline basis requires some theoretical assumptions. A numerical characterization for the existence can be found in [5]. Alternatively, for practical purposes one can simply use the Matlab toolbox in [115] and directly check whether the produced set of functions possess the properties of interest (non-negativity and partition of unity).
- The Matlab toolbox in [115] provides efficient evaluation routines for a wide range of TB-splines, with pieces belonging to ECT-spaces that are null-spaces of constant-coefficient linear differential operators, of interest in practical applications. However, as discussed and

illustrated in [115, Section 6.2], possible numerical instabilities can arise when extreme values for shape parameters (i.e., of the roots of the characteristic polynomial (2.8)), high degrees, high smoothness, and highly non-uniform partitions are considered. A “fidelity” check for the obtained basis is recommended when dealing with extreme configurations. Such choices are anyway less relevant in most practical applications. Nevertheless, providing more robust implementations is an interesting but challenging line of further research.

- As for any Galerkin method, accurate quadrature rules are imperative for isogeometric Galerkin methods based on TB-splines. When using classical elementwise Gaussian quadrature, the same order of quadrature used for polynomial B-splines produces satisfactory results only for mild values of the shape parameters. For larger values of the shape parameters, however, the amount of quadrature points needs to be significantly increased (up to five times). The construction of tailored quadrature rules for TB-splines, in the spirit of [30] and references therein, is definitely worth to be investigated.

Finally, we observe that, due to the complete structural similarity with polynomial B-splines, the most popular local tensor-product structures supporting local refinement (such as T-splines [109], (truncated) hierarchical B-splines [49], and locally refined splines [41]) can be extended to the Tchebycheffian setting. In particular, [16] outlines the construction of GB-splines on LR-meshes while hierarchical GB-splines have been presented in [80]. In the next chapter we present the construction of LR TB-splines.

4

Locally refined Tchebycheffian B-splines

In this chapter, we define Locally Refined (LR) TB-splines as a generalization of LR B-splines and we analyze their performance in the context of adaptive isogeometric analysis. The definition of LR TB-splines is driven by the knot insertion refinement process of tensor-product TB-splines, in complete analogy to the polynomial setting. In the bivariate tensor case, inserting a new knot in a pair of (global) knot vectors results in inserting a line segment in the mesh crossing the entire domain, thus refining all the TB-splines whose supports are crossed. On the contrary, LR TB-splines are defined on local knot vectors, and consequently the insertion of a new knot is always performed with respect to a particular LR TB-spline and results in refining only few basis functions.

The theoretical construction of LR TB-splines is independent of the particular ECT-spaces where the various pieces are drawn from. However, in the applicative context we confine ourselves to TB-splines from the ECT-spaces of the form (2.14).

In this chapter, first we present a local representation of the TB-splines and introduce some fundamental concepts such as knot insertion for TB-splines in Section 4.1 and extend this phenomenon to a multivariate setting. In Section 4.2 we define LR-meshes in a bivariate setup and construct TB-splines on such meshes, called LR TB-splines. Section 4.3 presents the adaptive scheme for solving differential problems using LR TB-splines and Section 4.4 collects various numerical results, showcasing the performance of adaptive isogeometric analysis based on LR TB-splines for the solution of few classical benchmark differential problems.

For the sake of simplicity, in this chapter we confine ourselves to select all the pieces of the spline space in (2.15) from a single space \mathbb{T}_p , which is the null-space of a linear differential operator with constant coefficients as

The results in this chapter are published in:

K. Raval, C. Manni, and H. Speleers, *Adaptive isogeometric analysis based on Tchebycheffian splines*, submitted

in (2.14). We assume that \mathbb{T}_p (and its derivative space) is an ECT-space on each interval $[x_{i-1}, x_i]$ separately, but not necessarily on the entire interval $[a, b]$. As it will be shown in the rest of the chapter, also in this simplified setting, the framework we are dealing with is very rich and offers enough flexibility to allow a proper treatment for a wide class of problems. Moreover, we assume maximal smoothness at the internal breakpoints in the partition, i.e., $\mathbf{r} := \{p-1, \dots, p-1\}$, and we drop the smoothness \mathbf{r} from the notation of the Tchebycheffian spline space. Thus, the Tchebycheffian spline space $\mathbb{S}_p(\mathcal{M})$ of degree p where all the pieces are drawn from the space \mathbb{T}_p and glued with maximal smoothness on a partition \mathcal{M} is given as

$$\begin{aligned} \mathbb{S}_p(\mathcal{M}) := & \left\{ f : [a, b] \rightarrow \mathbb{R} : f|_{[x_{i-1}, x_i]} \in \mathbb{T}_p, i = 1, \dots, m; \right. \\ & \left. D_-^l f(x_i) = D_+^l f(x_i), l = 0, \dots, p-1, i = 1, \dots, m-1 \right\}. \end{aligned} \quad (4.1)$$

4.1 Local representation of TB-splines

The spline space in (4.1) admits a set of basis functions which enjoy all the main properties of B-splines and therefore are called Tchebycheffian B-splines (TB-splines). In the rest of the chapter we assume that the space (4.1) admits a TB-spline basis. We refer the reader to Section 2.2.1, and references therein, for a detailed discussion about the existence and definition of such a basis.

As already mentioned in Section 2.2.1, like in the polynomial B-spline case, the set of TB-splines of the space in (4.1) can be defined using a vector of non-decreasing (open) knots,

$$\Xi := (\xi_k)_{k=1}^{n+p+1} := \left(\underbrace{x_0, \dots, x_0}_{p+1 \text{ times}}, x_1, \dots, x_{m-1}, \underbrace{x_m, \dots, x_m}_{p+1 \text{ times}} \right), \quad (4.2)$$

with $n := m + p$ being the dimension of the Tchebycheffian spline space $\mathbb{S}_p(\mathcal{M})$, and any TB-spline $N_{\Xi, k} : \mathbb{R} \rightarrow \mathbb{R}$ of degree p can be uniquely identified by a local knot vector of length $p + 2$,

$$\Xi_k := (\xi_{k,1}, \dots, \xi_{k,p+2}),$$

where $\xi_{k,1}, \dots, \xi_{k,p+2}$ are consecutive knots in (2.18).

4.1.1 Knot insertion for TB-splines

Similar to polynomial splines, see Section 1.1.3, we can represent a Tchebycheffian spline on a refined knot sequence by inserting a single knot at a time. This follows from the following proposition; see, e.g., [76, Section 3.4] for a proof.

Proposition 4.1 (Knot insertion in TB-splines). *Let $N_{\Xi_k, p}$ be a TB-spline as in Definition 2.13, identified by the local knot vector $\Xi_k := (\xi_{k,1}, \dots, \xi_{k,p+2})$. Adding a knot $\hat{\xi} \in [\xi_{k,1}, \xi_{k,p+2}]$ results in two TB-splines*

$$N_{(\xi_{k,1}, \dots, \hat{\xi}, \dots, \xi_{k,p+1}), p}, \quad N_{(\xi_{k,2}, \dots, \hat{\xi}, \dots, \xi_{k,p+2}), p},$$

such that

$$N_{(\xi_{k,1}, \dots, \xi_{k,p+2}), p} = \nu_k^{(1)} N_{(\xi_{k,1}, \dots, \hat{\xi}, \dots, \xi_{k,p+1}), p} + \nu_k^{(2)} N_{(\xi_{k,2}, \dots, \hat{\xi}, \dots, \xi_{k,p+2}), p}, \quad (4.3)$$

with

$$\begin{cases} \nu_k^{(1)} = 0, \nu_k^{(2)} = 1, & \text{if } \hat{\xi} = \xi_{k,1}, \\ \nu_k^{(1)} > 0, \nu_k^{(2)} > 0, & \text{if } \xi_{k,1} < \hat{\xi} < \xi_{k,p+2}, \\ \nu_k^{(1)} = 1, \nu_k^{(2)} = 0, & \text{if } \hat{\xi} = \xi_{k,p+2}. \end{cases}$$

The coefficients $\nu_k^{(1)}, \nu_k^{(2)} \in [0, 1]$, connecting two different scales of TB-splines, can be explicitly expressed in an analytical form involving the underlying space \mathbb{T}_p ; we refer the reader to [76, Section 3.4] for details. Alternatively, for practical purposes, they can be simply computed by solving the linear system obtained from (4.3) by evaluating the two sides at any pair of distinct points \hat{x}_1, \hat{x}_2 such that

$$\hat{x}_j \in (\xi_{k,j}, \xi_{k,p+j}), \quad j = 1, 2.$$

Example 4.2 (Knot insertion). Consider a Tchebycheffian spline space $\mathbb{S}_p(\mathcal{M})$ on the mesh $\mathcal{M} = \{0, 1, 2, 3\}$ related to the ECT-space $\mathbb{P}_2^{(\alpha)} = \langle 1, x, e^{\alpha x} \rangle$ of degree $p = 2$. We refine the TB-spline $N_{(0,0,1,2),2}$ by inserting the knot $\hat{\xi} = 3/2$. The knot insertion formula (4.3) gives

$$N_{(0,0,1,2),2} = N_{(0,0,1,3/2),2} + \left(\frac{e^{\alpha/2}}{2e^{\alpha/2} + 2} \right) N_{(0,1,3/2,2),2}.$$

Figure 4.1(a) illustrates the original TB-spline split into two new TB-splines after knot insertion. Figure 4.1(b) exhibits another example where, as a result of knot

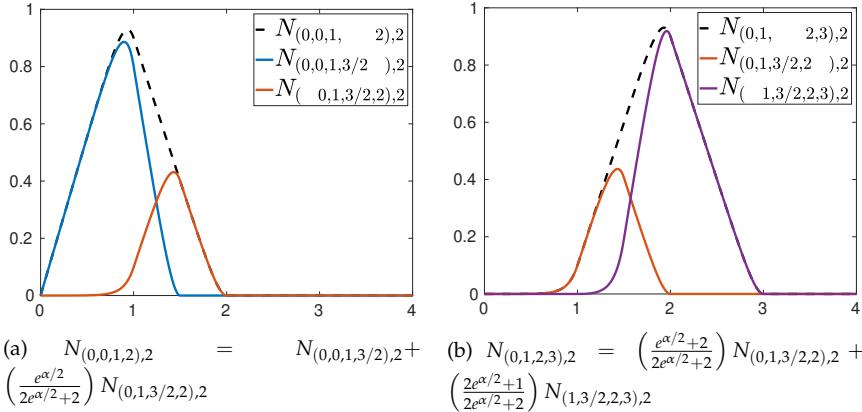


Figure 4.1: Inserting a knot $\hat{\xi} = 3/2$ for different TB-splines related to the ECT-space $\mathbb{P}_2^{(\alpha)} = \langle 1, x, e^{\alpha x} \rangle$ with $\alpha = 10$. Each existing spline is described as a combination of the refined TB-splines with their corresponding scaling factors as given in (4.3).

insertion at $\hat{\xi} = 3/2$, the TB-spline $N_{(0,1,2,3),2}$ splits as

$$N_{(0,1,2,3),2} = \left(\frac{e^{\alpha/2}+2}{2e^{\alpha/2}+2}\right) N_{(0,1,3/2,2),2} + \left(\frac{2e^{\alpha/2}+1}{2e^{\alpha/2}+2}\right) N_{(1,3/2,2,3),2}.$$

Since we are interested in spline spaces of maximal smoothness, from now on we only consider the case where the inserted knot $\hat{\xi}$ does not coincide with any existing knot in Ξ .

Remark 4.3. Let $\mathcal{S}_p(\mathcal{M})$ be the set of TB-splines spanning the Tchebycheffian spline space $\mathbb{S}_p(\mathcal{M})$ on the mesh \mathcal{M} . From (4.2) it follows that the cardinality of $\mathcal{S}_p(\mathcal{M})$ equals n . Let $\hat{\mathcal{M}}$ be the refined mesh obtained from \mathcal{M} by adding $\hat{\xi}$ as additional breakpoint in the interior of the interval $[a, b]$. On this refined mesh, the new set of TB-splines can be constructed by the following procedure.

1. Initialize the set by $\mathcal{S}_p(\hat{\mathcal{M}}) \leftarrow \mathcal{S}_p(\mathcal{M})$.
2. As long as there exists $N_{(\xi_{k,1}, \dots, \xi_{k,p+2}), p} \in \mathcal{S}_p(\hat{\mathcal{M}})$ with $\hat{\xi} \in [\xi_{k,1}, \xi_{k,p+2}]$ but $\hat{\xi} \notin \{\xi_{k,1}, \dots, \xi_{k,p+2}\}$:

4.1. Local representation of TB-splines

(a) Apply knot insertion as in (4.3):

$$N_{(\xi_{k,1}, \dots, \xi_{k,p+2}), p}(x) = v_k^{(1)} N_{(\xi_{k,1}, \dots, \hat{\xi}, \dots, \xi_{k,p+1}), p}(x) + v_k^{(2)} N_{(\xi_{k,2}, \dots, \hat{\xi}, \dots, \xi_{k,p+2}), p}(x).$$

(b) Update the set:

$$\begin{aligned} \mathcal{S}_p(\hat{\mathcal{M}}) &\leftarrow \left(\mathcal{S}_p(\hat{\mathcal{M}}) \setminus \{N_{(\xi_{k,1}, \dots, \xi_{k,p+2}), p}\} \right) \cup \\ &\quad \left\{ N_{(\xi_{k,1}, \dots, \hat{\xi}, \dots, \xi_{k,p+1}), p}, N_{(\xi_{k,2}, \dots, \hat{\xi}, \dots, \xi_{k,p+2}), p} \right\}. \end{aligned}$$

It can be verified that the obtained set $\mathcal{S}_p(\hat{\mathcal{M}})$ spans the $(n+1)$ -dimensional Tchebycheffian spline space $\mathbb{S}_p(\hat{\mathcal{M}})$ on the refined mesh $\hat{\mathcal{M}}$ and forms actually the TB-spline basis of this space. The above procedure is a simplified version of the LR TB-spline construction that will be discussed in Section 4.2 (see Definition 4.9).

4.1.2 Multivariate setting for TB-splines

As for the polynomial case, see Section 1.1.5, the local representation of univariate TB-splines can be easily extended to the multivariate case by taking their tensor-product. More precisely, let $x := (x_1, \dots, x_d)$, given degrees $p := (p_1, \dots, p_d)$ and root vectors w_1, w_2, \dots, w_d as in (2.10), we define the multivariate TB-spline $N_{\Xi_k, p} : \mathbb{R}^d \rightarrow \mathbb{R}$, with local knot vectors

$$\Xi_k := (\Xi_{k_1}^1, \Xi_{k_2}^2, \dots, \Xi_{k_d}^d),$$

as

$$N_{\Xi_k, p}(x) := \prod_{i=1}^d N_{\Xi_{k_i}^i, p_i}(x_i), \quad (4.4)$$

where $N_{\Xi_{k_i}^i, p_i}$ is a univariate TB-spline related to the ECT-space $\mathbb{P}_{p_i}^{\mathcal{W}_i}$ of degree p_i and root vector w_i as in (2.11) for $i = 1, \dots, d$. The function $N_{\Xi_k, p}$ belongs piecewisely to the space $\mathbb{P}_{p_1}^{\mathcal{W}_1} \otimes \mathbb{P}_{p_2}^{\mathcal{W}_2} \otimes \dots \otimes \mathbb{P}_{p_d}^{\mathcal{W}_d}$ and its support is given by the Cartesian product of the supports of the local knot vectors in each direction as

$$\text{supp}(N_{\Xi_k, p}) := [\xi_{k_1, 1}^1, \xi_{k_1, p_1+2}^1] \times \dots \times [\xi_{k_d, 1}^d, \xi_{k_d, p_d+2}^d].$$

In the following we restrict our attention to the bivariate case $d = 2$. Expanding the concept of knot insertion from a univariate to a bivariate TB-spline can be intuitively compared to inserting an axis-aligned line.

The sole necessary condition for the inserted line is that it must split the complete support of the TB-spline undergoing the refinement process. The bivariate TB-spline $N_{\Xi_k, \mathbf{p}}$, with local knot vector

$$\Xi_k = \left(\Xi_{k_1}^1, \Xi_{k_2}^2 \right) = \left((\xi_{k_1,1}^1, \dots, \xi_{k_1,p_1+2}^1), (\xi_{k_2,1}^2, \dots, \xi_{k_2,p_2+2}^2) \right),$$

can be refined by inserting, e.g., a vertical line $\{\hat{\xi}\} \times [\xi_{k_2,1}^2, \xi_{k_2,p_2+2}^2]$, with $\hat{\xi} \in (\xi_{k_1,1}^1, \xi_{k_1,p_1+2}^1)$. Analogous to knot insertion, the line insertion results in splitting the given TB-spline in two new TB-splines as

$$\begin{aligned} N_{\Xi_k, \mathbf{p}}(\mathbf{x}) &= N_{(\xi_{k_1,1}^1, \dots, \xi_{k_1,p_1+2}^1), p_1}(x_1) N_{(\xi_{k_2,1}^2, \dots, \xi_{k_2,p_2+2}^2), p_2}(x_2) \\ &= \left(\nu_k^{(1)} N_{(\xi_{k_1,1}^1, \dots, \hat{\xi}, \dots, \xi_{k_1,p_1+1}^1), p_1}(x_1) + \nu_k^{(2)} N_{(\xi_{k_1,2}^1, \dots, \hat{\xi}, \dots, \xi_{k_1,p_1+2}^1), p_1}(x_1) \right) \\ &\quad N_{(\xi_{k_2,1}^2, \dots, \xi_{k_2,p_2+2}^2), p_2}(x_2) \\ &=: \nu_k^{(1)} N_{\Xi_{l_1}, \mathbf{p}}(\mathbf{x}) + \nu_k^{(2)} N_{\Xi_{l_2}, \mathbf{p}}(\mathbf{x}). \end{aligned} \quad (4.5)$$

The non-negative scaling factors $\nu_k^{(1)}, \nu_k^{(2)} \in [0, 1]$ are computed in the same manner as in (4.3). It is clear that the scaling relation for insertion of a horizontal line is alike.

4.2 TB-splines on LR-meshes

In this section, we discuss the construction of TB-splines on Locally Refined meshes (LR-meshes), which are particular axis-aligned box-partitions. First, we explain how these meshes can be obtained by means of successive line insertion. Subsequently, we define LR TB-splines as an extension of polynomial LR B-splines. We refer the reader to [41] for in-depth information about splines over locally refined box-partitions. Our presentation closely follows [93, Section 2], extending to the Tchebycheff setting the concepts presented therein.

4.2.1 LR-meshes

Throughout this chapter we keep the description of LR-meshes limited to the bivariate case. Note that several definitions we are going to state are simplified for an easier reading because we confine ourselves to the case of maximal smoothness. However, the Tchebycheff setting allows for full generality as in the algebraic polynomial case.

Definition 4.4 (Box-partition and mesh). Given an axis-aligned rectangle

$\Omega \subseteq \mathbb{R}^2$, a box-partition of Ω is a finite collection \mathcal{E} of axis-aligned rectangles, called elements σ , such that

- $\bigcup_{\sigma \in \mathcal{E}} \sigma = \Omega$,
- $\sigma_1^\circ \cap \sigma_2^\circ = \emptyset$ for any $\sigma_1, \sigma_2 \in \mathcal{E}$, with $\sigma_1 \neq \sigma_2$,

where the interior of an element σ is denoted by σ° . We define the set \mathcal{V} consisting of all vertices from all the elements of \mathcal{E} as the vertices of \mathcal{E} . A meshline γ of \mathcal{E} is an axis-aligned segment contained in an element edge, connecting two and only two vertices of \mathcal{V} at its end-points. The set \mathcal{M} of all the meshlines γ of \mathcal{E} is called the mesh of \mathcal{E} .

We can represent a meshline γ as the Cartesian product of a point in \mathbb{R} and a finite interval. For example, a vertical meshline can be described as $\{\xi_i^1\} \times [\xi_j^2, \xi_{j+1}^2]$, while a horizontal meshline can be described as $[\xi_i^1, \xi_{i+1}^1] \times \{\xi_j^2\}$.

Definition 4.5 (μ -extended mesh). Given a mesh \mathcal{M} and bi-degree $\mathbf{p} = (p_1, p_2)$, the meshline multiplicity function $\mu : \mathcal{M} \mapsto \mathbb{Z}_{>0}$ is a function that associates a positive integer with every meshline, called multiplicity of the meshline. A mesh that has an assigned multiplicity function μ is called μ -extended mesh. The meshline multiplicity is assumed to be maximally $p_k + 1$ in its corresponding direction k . A mesh \mathcal{M} is considered an open mesh if every boundary meshline has maximal multiplicity. Moreover, when all the meshlines of the box-partition \mathcal{E} have the same multiplicity μ except on the boundary of the domain, we can say that the mesh \mathcal{M} has multiplicity μ .

In this chapter we fix the multiplicity of all meshlines to one except on the boundary, where it is fixed to maximum. Hence, all the considered meshes will be of multiplicity $\mu = 1$ and on the boundaries they will be open. Figure 4.2 illustrates an example of a box-partition \mathcal{E} of Ω and related μ -extended mesh \mathcal{M} .

A vertex of \mathcal{E} is called T-vertex if it is the intersection of two collinear meshlines and a single orthogonal meshline. A tensor mesh is a special case of μ -extended mesh, where the T-vertices occur only on the boundaries of the domain and all collinear meshlines have the same multiplicity μ . We can describe any bivariate tensor mesh on two knot vectors $\Xi = ((\xi_1^1, \dots, \xi_s^1), (\xi_1^2, \dots, \xi_t^2))$ as

$$\begin{aligned} \mathcal{M}(\Xi) := & \left\{ \{\xi_i^1\} \times [\xi_j^2, \xi_{j+1}^2] : i = 1, \dots, s; j = 1, \dots, t - 1 \right\} \\ & \cup \left\{ [\xi_i^1, \xi_{i+1}^1] \times \{\xi_j^2\} : i = 1, \dots, s - 1; j = 1, \dots, t \right\}. \end{aligned}$$

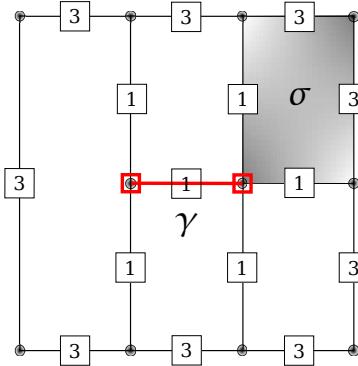


Figure 4.2: A box-partition \mathcal{E} of the domain Ω with elements σ and meshlines γ . The vertices of \mathcal{V} are marked with circles. The μ -extended mesh \mathcal{M} of \mathcal{E} for bi-degree $\mathbf{p} = (2, 2)$ assumes that the boundary meshlines have maximal multiplicity ($\mu = 3$) and all internal meshlines have a fixed multiplicity equal to $\mu = 1$. The multiplicities associated with all meshlines are depicted in the squares along the lines.

The multiplicities of the meshlines in $\mathcal{M}(\Xi)$ are given by the multiplicities of the knots in Ξ .

Definition 4.6 (Support). Given a mesh \mathcal{M} and a bivariate TB-spline $N_{\Xi_k, \mathbf{p}}$, we say that $N_{\Xi_k, \mathbf{p}}$ has support on \mathcal{M} if

- the meshlines in $\mathcal{M}(\Xi_k)$ can be obtained as the union of meshlines in \mathcal{M} , and
- the multiplicities of the meshlines in $\mathcal{M}(\Xi_k)$ are less than or equal to the multiplicities of the corresponding meshlines in \mathcal{M} .

Furthermore, we say that $N_{\Xi_k, \mathbf{p}}$ has minimal support if

- the multiplicities of the interior meshlines in $\mathcal{M}(\Xi_k)$ are equal to the multiplicities of the corresponding meshlines in \mathcal{M} , and
- there is no collection of collinear meshlines γ in $\mathcal{M} \setminus \mathcal{M}(\Xi_k)$ such that $\text{supp}(N_{\Xi_k, \mathbf{p}}) \setminus \gamma$ is not connected.

Figure 4.3 illustrates some examples of TB-splines of bi-degree $\mathbf{p} = (2, 2)$ with minimal support on a mesh \mathcal{M} with fixed multiplicity $\mu = 1$. Note that not all TB-splines that have support on \mathcal{M} have minimal support on \mathcal{M} .

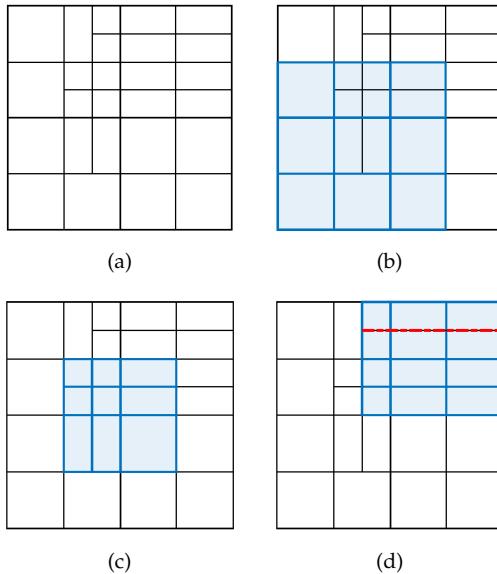


Figure 4.3: Some supports of TB-splines of bi-degree $p = (2, 2)$ on the given mesh \mathcal{M} in (a) with multiplicity $\mu = 1$. The TB-splines in (b) and (c) have minimal support on \mathcal{M} , with their knots highlighted by thicker lines. The TB-spline in (d) does not have minimal support on \mathcal{M} , as the dashed line disconnects its support.

Definition 4.7 (Split). Given a box-partition \mathcal{E} and an axis-aligned segment γ , we say that γ traverses $\sigma \in \mathcal{E}$ if $\gamma \subset \sigma$ and the interior of σ is divided into two parts by γ , i.e., $\sigma \setminus \gamma$ is not connected. A split is defined as a finite union of contiguous and collinear axis-aligned segments $\gamma = \cup_i \gamma_i$ such that every γ_i is either a meshline of \mathcal{E} or it traverses some $\sigma \in \mathcal{E}$.

LR-meshes are meshes obtained from a tensor mesh by successively inserting (sufficiently long) splits and LR TB-splines are defined as a specific set of TB-splines with minimal support on an LR-mesh. When a split γ is inserted into a box-partition \mathcal{E} , any traversed rectangle $\sigma \in \mathcal{E}$ is replaced with two sub-rectangles given by the closures of the connected components of $\sigma \setminus \gamma$. The resulting new box-partition will be denoted by $\mathcal{E} + \gamma$ and its corresponding mesh by $\mathcal{M} + \gamma$. We focus only on meshes of multiplicity $\mu = 1$ that are open on the boundaries. Hence, when we refine such a mesh \mathcal{M} with a split γ , the multiplicities of the meshlines shared by both \mathcal{M} and $\mathcal{M} + \gamma$ remain the same, while multiplicity one is assigned to the new meshlines in γ .

Definition 4.8 (Traversal of a TB-spline). Given an open mesh \mathcal{M} of mul-

tiplicity $\mu = 1$, a TB-spline $N_{\Xi_k, p}$ with support on \mathcal{M} , and a split γ , we say that γ traverses $N_{\Xi_k, p}$ if the interior of $\text{supp}(N_{\Xi_k, p})$ is divided in two disjoint parts by γ , i.e., $\text{supp}(N_{\Xi_k, p}) \setminus \gamma$ is not connected, and γ is in $\mathcal{M} \setminus \mathcal{M}(\Xi_k)$.

4.2.2 LR TB-splines

At this point we have all fundamentals to construct LR TB-splines. We start with an open tensor mesh and the corresponding set of tensor TB-splines. Then, we refine the mesh by inserting splits, one at a time, and whenever a TB-spline in the considered set has no longer minimal support during the mesh refinement process, we refine it by using the knot insertion procedure.

Definition 4.9 (LR TB-splines). Let \mathcal{M}_0 be an open tensor mesh of multiplicity $\mu = 1$ and let $\mathcal{S}_p(\mathcal{M}_0)$ be the corresponding set of tensor-product TB-splines of bi-degree p (and maximal smoothness) on \mathcal{M}_0 . We then define a sequence of meshes $\mathcal{M}_1, \mathcal{M}_2, \dots$ of multiplicity $\mu = 1$ and a sequence of sets of TB-splines $\mathcal{S}_p(\mathcal{M}_1), \mathcal{S}_p(\mathcal{M}_2), \dots$ as follows. For $j = 1, 2, \dots$, let γ_j be a split such that $\mathcal{M}_j = \mathcal{M}_{j-1} + \gamma_j$ and such that at least one TB-spline in $\mathcal{S}_p(\mathcal{M}_{j-1})$ is traversed by a split in \mathcal{M}_j . On this refined mesh \mathcal{M}_j the new set of TB-splines is constructed by the following procedure.

1. Initialize the set by $\mathcal{S}_p(\mathcal{M}_j) \leftarrow \mathcal{S}_p(\mathcal{M}_{j-1})$.
2. As long as there exists $N_{\Xi_k, p} \in \mathcal{S}_p(\mathcal{M}_j)$ with no minimal support on \mathcal{M}_j :
 - (a) Apply knot insertion as in (4.5): $N_{\Xi_k, p} = \nu_k^{(1)} N_{\Xi_{l_1}, p} + \nu_k^{(2)} N_{\Xi_{l_2}, p}$.
 - (b) Update the set: $\mathcal{S}_p(\mathcal{M}_j) \leftarrow (\mathcal{S}_p(\mathcal{M}_j) \setminus N_{\Xi_k, p}) \cup \{N_{\Xi_{l_1}, p}, N_{\Xi_{l_2}, p}\}$.

The generated mesh \mathcal{M}_j is referred to as an LR-mesh and the corresponding set $\mathcal{S}_p(\mathcal{M}_j)$ is designated as a set of LR TB-splines.

Remark 4.10. The splitting procedure described in Step 2 of Definition 4.9 to construct a set of TB-splines on the refined mesh \mathcal{M}_j (obtained by inserting split γ_j into mesh \mathcal{M}_{j-1}) can be subdivided into two steps.

- *Primary split:* Refine any existing TB-spline whose support is traversed by γ_j or by the union of γ_j and any existing split in \mathcal{M}_{j-1} that intersects and is collinear with γ_j .
- *Secondary split:* Refine any newly created TB-spline whose support is traversed by any existing split in \mathcal{M}_{j-1} .

4.2. TB-splines on LR-meshes

After the primary split step, the supports of the newly created TB-splines are expected to be smaller than the original ones. This, in turn, opens up the possibility that existing splits in \mathcal{M}_{j-1} might traverse these new TB-splines. Therefore, the secondary split step is designed to check for this possibility. If there is such a split, the secondary split step is executed once more, and this process continues until there are no remaining new TB-splines that are traversed by any split in \mathcal{M}_{j-1} . This implementation of LR TB-splines is summarized in Algorithm 1.

Algorithm 1: LR TB-spline refinement

Input :	$\mathcal{S}_p(\mathcal{M}_{j-1})$ TB-spline set γ_j Split
Output:	$\mathcal{S}_p(\mathcal{M}_j)$ Refined TB-spline set

```

1:  $\mathcal{S}_p(\mathcal{M}_j) \leftarrow \mathcal{S}_p(\mathcal{M}_{j-1})$  and  $\hat{\mathcal{S}}_p(\mathcal{M}_j) \leftarrow \emptyset$ ;
   /* Primary split: */
2: Extend  $\gamma_j$  by any existing split in  $\mathcal{M}_{j-1}$  that intersects and is
   collinear with  $\gamma_j$ ;
3: for every TB-spline  $N_{\Xi_k, p} \in \mathcal{S}_p(\mathcal{M}_{j-1})$  do
4:   if  $\gamma_j$  traverses support of  $N_{\Xi_k, p}$  then
5:     Refine  $N_{\Xi_k, p}$  and compute splitting coefficients  $v_k^{(1)}, v_k^{(2)}$ 
       from (4.3);
6:     Update the weights according to (4.6);
7:     Remove  $N_{\Xi_k, p}$  from  $\mathcal{S}_p(\mathcal{M}_j)$  and add new TB-splines to
        $\mathcal{S}_p(\mathcal{M}_j)$  and  $\hat{\mathcal{S}}_p(\mathcal{M}_j)$ ;
8:   end
9: end
   /* Secondary split: */
10: while  $\hat{\mathcal{S}}_p(\mathcal{M}_j)$  is non-empty do
11:   Select any TB-spline  $N_{\Xi_k, p} \in \hat{\mathcal{S}}_p(\mathcal{M}_j)$ ;
12:   for every existing split  $\gamma_\ell$  in  $\mathcal{M}_{j-1}$  do
13:     if  $\gamma_\ell$  traverses support of  $N_{\Xi_k, p}$  then
14:       Refine  $N_{\Xi_k, p}$  and compute splitting coefficients
           $v_k^{(1)}, v_k^{(2)}$  from (4.3);
15:       Update the weights according to (4.6);
16:       Remove  $N_{\Xi_k, p}$  from  $\mathcal{S}_p(\mathcal{M}_j)$  and add new TB-splines to
           $\mathcal{S}_p(\mathcal{M}_j)$  and  $\hat{\mathcal{S}}_p(\mathcal{M}_j)$ ;
17:     end
18:     Remove  $N_{\Xi_k, p}$  from  $\hat{\mathcal{S}}_p(\mathcal{M}_j)$ ;
19:   end
20: end

```

Remark 4.11. Any refinement sequence producing a given LR-mesh \mathcal{M}_j is not inherently unique, as the order of split insertions can often be altered. Nonetheless, the set of LR TB-splines $\mathcal{S}_p(\mathcal{M}_j)$ is well defined on \mathcal{M}_j because it is independent of such insertion ordering, as proved in [41, Theorem 3.4] for polynomial LR B-splines but the proof also extends to LR TB-splines. Therefore, once the set

of LR TB-splines on \mathcal{M}_j has been constructed, it is safe to discard any data linked to the previous iteration, including the mesh \mathcal{M}_{j-1} .

As in the polynomial case, due to the local refinement in new LR TB-splines, the partition of unity property is lost. This property is essential for interpreting the LR TB-spline coefficients as control points and ensuring the convex hull property. Restoration of the partition of unity in the new set of LR TB-splines can be achieved by a proper scaling, resulting in the so-called weighted LR TB-splines.

Definition 4.12 (Weighted LR TB-splines). Given a set of LR TB-splines $\mathcal{S}_p(\mathcal{M}_j)$ constructed according to Definition 4.9, the set of weighted LR TB-splines satisfies

$$\sum_{N_{\Xi_k, \mathbf{p}} \in \mathcal{S}_p(\mathcal{M}_j)} N_{\Xi_k, \mathbf{p}}^\sigma(\mathbf{x}) := \sum_{N_{\Xi_k, \mathbf{p}} \in \mathcal{S}_p(\mathcal{M}_j)} \sigma_k N_{\Xi_k, \mathbf{p}}(\mathbf{x}) = 1, \quad \mathbf{x} \in \Omega,$$

for some positive weights $\sigma_k \in \mathbb{R}$, defined as follows for $j = 0, 1, 2, \dots$. The initial weights corresponding to the tensor-product TB-splines on the tensor mesh \mathcal{M}_0 are set to $\sigma_k = 1$. When we advance refining $N_{\Xi_k, \mathbf{p}}$ (with weight σ_k) as in Step 2 of Definition 4.9, the weights σ_{l_1} and σ_{l_2} corresponding to the refined TB-splines $N_{\Xi_{l_1}, \mathbf{p}}$ and $N_{\Xi_{l_2}, \mathbf{p}}$ on \mathcal{M}_j are updated as

$$\sigma_{l_i} \leftarrow \sigma_k^{(i)} := \begin{cases} \nu_k^{(i)} \sigma_k, & \text{if } N_{\Xi_{l_i}, \mathbf{p}} \notin \mathcal{S}_p(\mathcal{M}_j), \\ \sigma_{l_i} + \nu_k^{(i)} \sigma_k, & \text{if } N_{\Xi_{l_i}, \mathbf{p}} \in \mathcal{S}_p(\mathcal{M}_j), \end{cases} \quad i = 1, 2. \quad (4.6)$$

Due to the structural similarity between the Tchebycheff and the polynomial setting (see Proposition 2.14), the LR TB-splines considered in Definitions 4.9 and 4.12 enjoy the same properties as polynomial LR B-splines; see [19, 41]. In particular, from their construction it follows immediately that they are non-negative, have minimal support, and the weighted LR TB-splines sum up to one. We collect them in the following proposition. These properties are direct consequence of the construction except for the weighted partition of unity and the well definition that can be proved with the same line of arguments as in the polynomial case; see [41, Appendix].

Proposition 4.13 (LR TB-spline properties). Let $\mathbb{S}_p(\mathcal{M}_j) = \langle N_{\Xi_k, \mathbf{p}} \in \mathcal{S}_p(\mathcal{M}_j) \rangle$ be the TB-spline space spanned by the set of TB-splines $\mathcal{S}_p(\mathcal{M}_j)$ as in Definition 4.9 on a sequence of meshes $\mathcal{M}_0, \mathcal{M}_1, \mathcal{M}_2, \dots$ such that $\mathcal{M}_{j+1} = \mathcal{M}_j + \gamma_j$, then the following properties hold.

- Non-negativity: $N_{\Xi_k, \mathbf{p}}(\mathbf{x}) > 0$ for all $\mathbf{x} \in (\xi_{k,1}^1, \xi_{k,p+2}^1) \times (\xi_{k,1}^2, \xi_{k,p+2}^2)$;
- Local support: $N_{\Xi_k, \mathbf{p}}$ has minimal support on \mathcal{M}_j ;

- *Weighted partition of unity*: $\sum_{N_{\Xi_k, \mathbf{p}} \in \mathcal{S}_{\mathbf{p}}(\mathcal{M})} \sigma_k N_{\Xi_k, \mathbf{p}}(x) = 1$, with $\sigma_k \in \mathbb{R}^+$;
- *Nestedness*: $\mathbb{S}_{\mathbf{p}}(\mathcal{M}_j) \subseteq \mathbb{S}_{\mathbf{p}}(\mathcal{M}_{j+1})$;
- *Well defined*: for any variation in the sequence of meshline insertion, as long the final mesh is equal $\mathcal{M} = \hat{\mathcal{M}}$, then the TB-spline set $\mathbb{S}_{\mathbf{p}}(\mathcal{M}) = \mathbb{S}_{\mathbf{p}}(\hat{\mathcal{M}})$.

Unfortunately, LR (T)B-splines are not always linearly independent. Extensive studies have been conducted to investigate the linear dependency for polynomial LR B-splines, and several refinement strategies have been proposed to address this issue; see, e.g., [91, 92, 93]. Similar strategies are also applicable for TB-splines on LR-meshes.

Figure 4.4 illustrates some bi-quadratic LR TB-splines obtained from the local space $\mathbb{P}_2^{(50)} \otimes \mathbb{P}_2^{(50)} = \langle 1, x_1, e^{50x_1} \rangle \otimes \langle 1, x_2, e^{50x_2} \rangle$ on a given LR-mesh \mathcal{M} .

4.2.3 Constructive example of LR TB-spline

We replicate the example presented in [61, Section 2.2.4] for the local space $\mathbb{P}_2^{(\alpha)} \otimes \mathbb{P}_2^{(\alpha)} = \langle 1, x_1, e^{\alpha x_1} \rangle \otimes \langle 1, x_2, e^{\alpha x_2} \rangle$ of bi-degree $\mathbf{p} = (2, 2)$. We start from the open tensor mesh \mathcal{M}_0 given by the global knot vectors

$$\Xi = ((0, 0, 0, 1, 2, 4, 5, 6, 6, 6), (0, 0, 0, 1, 2, 4, 5, 6, 6, 6));$$

see Figure 4.5. All the tensor TB-splines in the set $\mathcal{S}_{\mathbf{p}}(\mathcal{M}_0)$ have weight $\sigma_k = 1$. In the example we insert two splits and we monitor the evolution of the weights corresponding to each TB-spline refinement according to Definition 4.12. Note that a related knot insertion procedure in the univariate case has been elaborated in Example 4.2.

First, we introduce a vertical split $\gamma_1 = \{3\} \times [1, 5]$ into \mathcal{M}_0 , resulting in \mathcal{M}_1 . This affects three TB-splines as illustrated in Figure 4.6, creating

4.2. TB-splines on LR-meshes

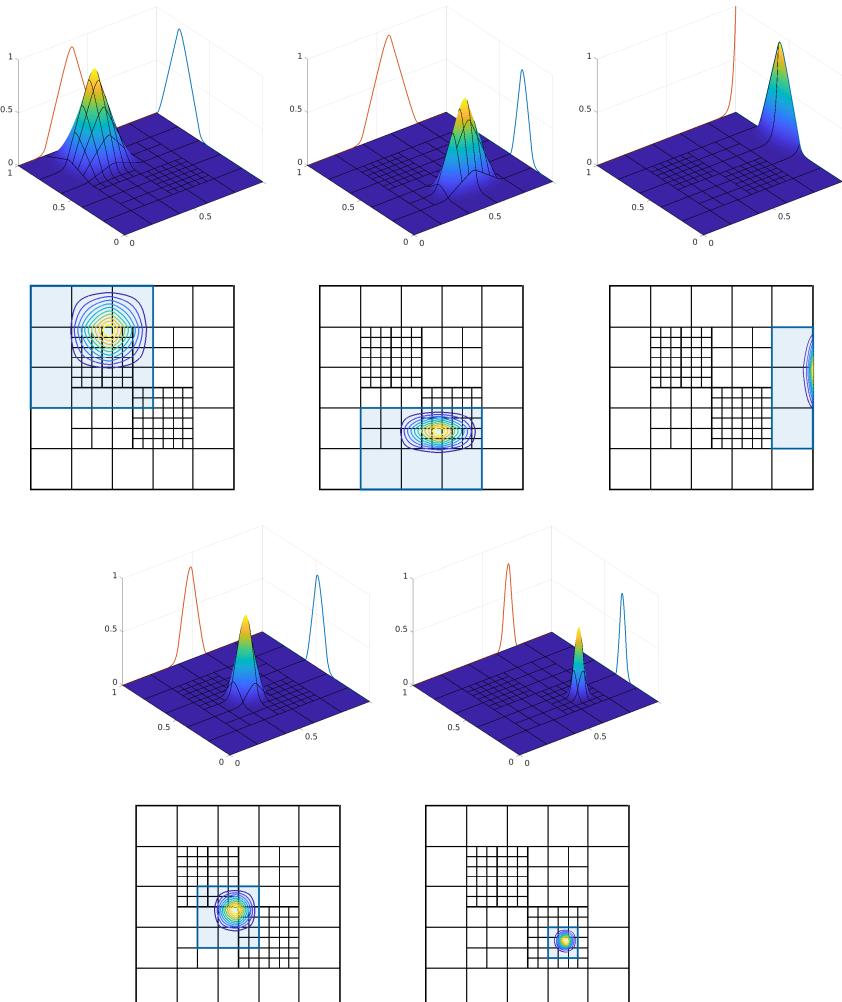


Figure 4.4: Some bi-quadratic LR TB-splines related to the local space $\mathbb{P}_2^{(50)} \otimes \mathbb{P}_2^{(50)}$ on a given LR-mesh \mathcal{M} . These splines, visualized along with their contour plots, are tensor products of univariate TB-splines, which are depicted on the coordinate planes.

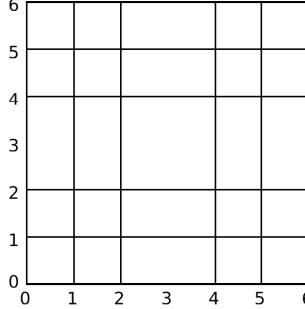


Figure 4.5: Initial tensor mesh \mathcal{M}_0 used to generate the TB-spline set $\mathcal{S}_{\mathbf{p}}(\mathcal{M}_0)$ related to the local space $\mathbb{P}_2^{(\alpha)} \otimes \mathbb{P}_2^{(\alpha)}$ of bi-degree $\mathbf{p} = (2, 2)$.

four new refined TB-splines,

$$\begin{aligned}
 N_{((0,1,2,4),(1,2,4,5)),(2,2)} &= N_{((0,1,2,3),(1,2,4,5)),(2,2)} \\
 &\quad + \left(\frac{e^\alpha}{2e^\alpha + 1} \right) N_{((1,2,3,4),(1,2,4,5)),(2,2)}, \\
 N_{((1,2,4,5),(1,2,4,5)),(2,2)} &= \left(\frac{e^\alpha + 1}{2e^\alpha + 1} \right) N_{((1,2,3,4),(1,2,4,5)),(2,2)} \\
 &\quad + \left(\frac{e^\alpha + 1}{e^\alpha + 2} \right) N_{((2,3,4,5),(1,2,4,5)),(2,2)}, \\
 N_{((2,4,5,6),(1,2,4,5)),(2,2)} &= \left(\frac{1}{e^\alpha + 2} \right) N_{((2,3,4,5),(1,2,4,5)),(2,2)} \\
 &\quad + N_{((3,4,5,6),(1,2,4,5)),(2,2)}. \tag{4.7}
 \end{aligned}$$

As a result of the split of $N_{((0,1,2,4),(1,2,4,5)),(2,2)}$ in (4.7), we achieve two new TB-splines $N_{((0,1,2,3),(1,2,4,5)),(2,2)}$ and $N_{((1,2,3,4),(1,2,4,5)),(2,2)}$ with the scaling coefficients 1 and $\left(\frac{e^\alpha}{2e^\alpha + 1} \right)$, respectively. Since both TB-splines are new and the weight of $N_{((0,1,2,4),(1,2,4,5)),(2,2)}$ from which they split is one, the resulting weights according to (4.6) are the same as their scaling coefficients as shown in Column 2 of Table 4.1. Next, the split of $N_{((1,2,4,5),(1,2,4,5)),(2,2)}$ in (4.7) results in one of the existing TB-splines given as $N_{((1,2,3,4),(1,2,4,5)),(2,2)}$ with scaling coefficient $\left(\frac{e^\alpha + 1}{2e^\alpha + 1} \right)$. Here the weight of the existing spline is updated according to (4.6) as presented in Column 3 in Table 4.1,

$$\sigma_2^{(1)} = \sigma_1^{(2)} + \left(\frac{e^\alpha + 1}{2e^\alpha + 1} \right) \cdot \sigma_2 = \left(\frac{e^\alpha}{2e^\alpha + 1} \right) + \left(\frac{e^\alpha + 1}{2e^\alpha + 1} \right) \cdot 1 = 1,$$

4.2. TB-splines on LR-meshes

where the existing weight of the TB-spline is $\left(\frac{e^\alpha}{2e^\alpha+1}\right)$. Similarly, with the split of $N_{((2,4,5,6),(1,2,4,5)),(2,2)}$ we complete the primary split step (see Remark 4.10), resulting in four new TB-splines and their corresponding weights as tabulated in Column 5 in Table 4.1.

Proceeding with the secondary split step (see Remark 4.10) in the refinement process, every new TB-spline listed in Column 1 in Table 4.1 is tested against traversal by any existing split in \mathcal{M}_0 . It turns out that there are no TB-splines traversed here, hence the set of all TB-splines after the primary split step is the final set $\mathcal{S}_p(\mathcal{M}_1)$.

Second, we insert a horizontal split $\gamma_2 = [1, 5] \times \{3\}$ into \mathcal{M}_1 , resulting in \mathcal{M}_2 . This affects four TB-splines as illustrated in Figure 4.7, creating eight new refined TB-splines,

$$\begin{aligned}
 N_{((1,2,3,4),(1,2,4,5)),(2,2)} &= \left(\frac{e^\alpha + 1}{2e^\alpha + 1}\right) N_{((1,2,3,4),(1,2,3,4)),(2,2)} \\
 &\quad + \left(\frac{e^\alpha + 1}{e^\alpha + 2}\right) N_{((1,2,3,4),(2,3,4,5)),(2,2)}, \\
 N_{((1,2,4,5),(0,1,2,4)),(2,2)} &= N_{((1,2,4,5),(0,1,2,3)),(2,2)} \\
 &\quad + \left(\frac{e^\alpha}{2e^\alpha + 1}\right) N_{((1,2,4,5),(1,2,3,4)),(2,2)}, \\
 N_{((1,2,4,5),(2,4,5,6)),(2,2)} &= \left(\frac{1}{e^\alpha + 2}\right) N_{((1,2,4,5),(2,3,4,5)),(2,2)} \\
 &\quad + N_{((1,2,4,5),(3,4,5,6)),(2,2)}, \\
 N_{((2,3,4,5),(1,2,4,5)),(2,2)} &= \left(\frac{e^\alpha + 1}{2e^\alpha + 1}\right) N_{((2,3,4,5),(1,2,3,4)),(2,2)} \\
 &\quad + \left(\frac{e^\alpha + 1}{e^\alpha + 2}\right) N_{((2,3,4,5),(2,3,4,5)),(2,2)}.
 \end{aligned} \tag{4.8}$$

Since the created TB-splines as a result of the primary split step are all new and different, their weights are the same as their scaling coefficients in (4.8). We list these eight TB-splines with their corresponding weights in Column 1 of Table 4.2. We now proceed with the secondary split step, where the new TB-splines are tested against traversal by any existing split in \mathcal{M}_1 . It turns out that two of them are traversed by a split. These two

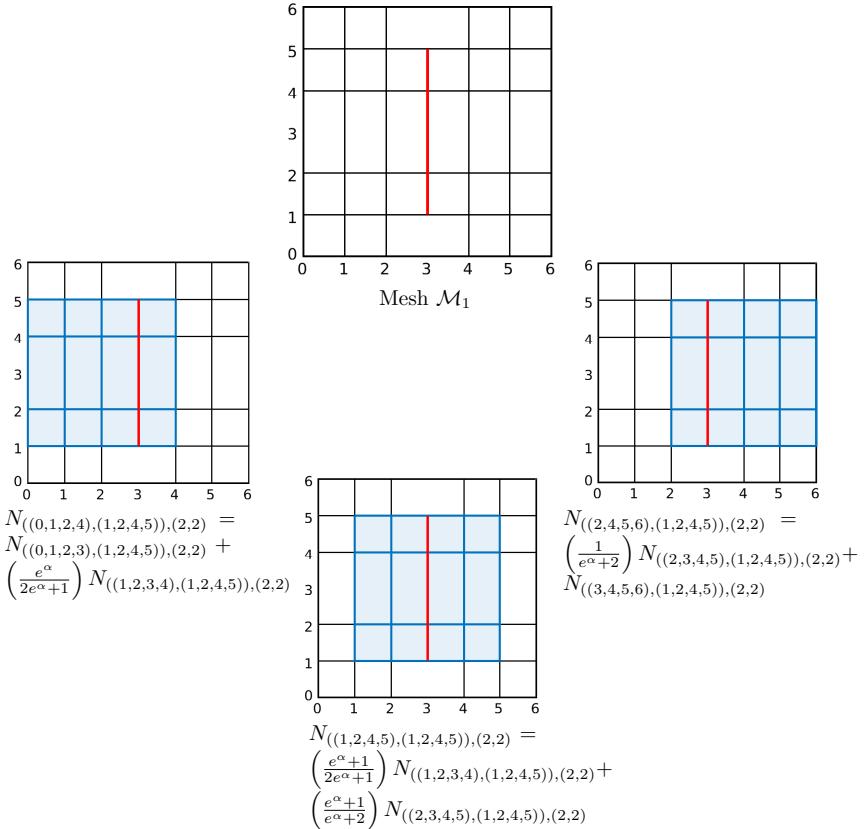


Figure 4.6: Split insertion (Iteration 1): The mesh \mathcal{M}_1 obtained by inserting a vertical split $\gamma_1 = \{3\} \times [1, 5]$ into the tensor mesh \mathcal{M}_0 illustrated in Figure 4.5, together with all three TB-splines traversed by γ_1 . The supports of the TB-splines are highlighted and annotated with the corresponding scaling relation between the original TB-spline and the two new TB-splines into which it has been refined, with their scaling factors for the local space $\mathbb{P}_2^{(\alpha)} \otimes \mathbb{P}_2^{(\alpha)}$ of bi-degree $p = (2, 2)$.

4.2. TB-splines on LR-meshes

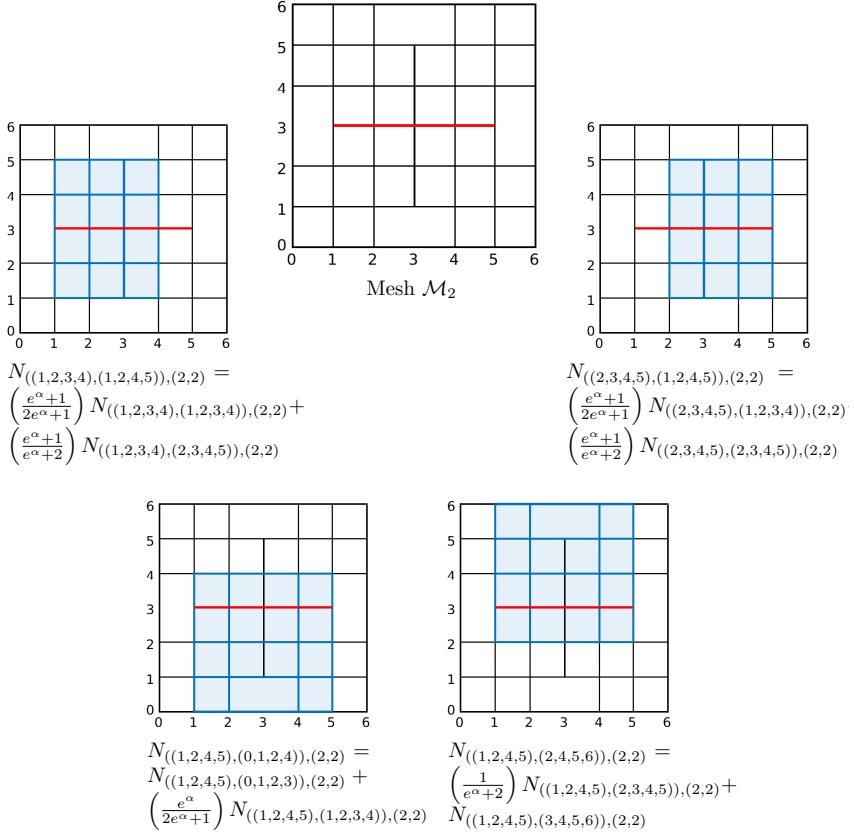
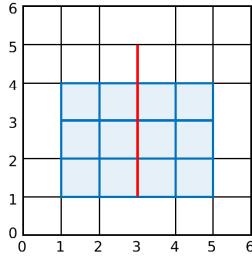
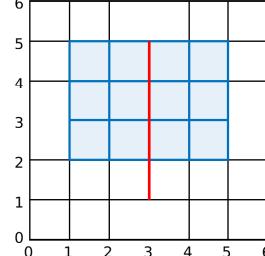


Figure 4.7: Split insertion (Iteration 2 - Primary split): The mesh \mathcal{M}_2 obtained by inserting a horizontal split $\gamma_2 = [1, 5] \times \{3\}$ into the mesh \mathcal{M}_1 illustrated in Figure 4.6, together with all four TB-splines traversed by γ_2 . The supports of the TB-splines are highlighted and annotated with the corresponding scaling relation between the original TB-spline and the two new TB-splines into which it has been refined, with their scaling factors for the local space $\mathbb{P}_2^{(\alpha)} \otimes \mathbb{P}_2^{(\alpha)}$ of bi-degree $p = (2, 2)$.



$$N_{((1,2,4,5),(1,2,3,4)),(2,2)} = \left(\frac{e^\alpha + 1}{2e^\alpha + 1}\right) N_{((1,2,3,4),(1,2,3,4)),(2,2)} + \left(\frac{e^\alpha + 1}{e^\alpha + 2}\right) N_{((2,3,4,5),(1,2,3,4)),(2,2)}$$



$$N_{((1,2,4,5),(2,3,4,5)),(2,2)} = \left(\frac{e^\alpha + 1}{2e^\alpha + 1}\right) N_{((1,2,3,4),(2,3,4,5)),(2,2)} + \left(\frac{e^\alpha + 1}{e^\alpha + 2}\right) N_{((2,3,4,5),(2,3,4,5)),(2,2)}$$

Figure 4.8: Split insertion (Iteration 2 - Secondary split): After the primary split step on the mesh \mathcal{M}_2 as depicted in Figure 4.7, eight new TB-splines have been created of which two are further traversed by $\gamma_1 = \{3\} \times [1, 5]$. The supports of these two TB-splines are highlighted and annotated with the corresponding scaling relation between the original TB-spline and the two new TB-splines into which it has been refined, with their scaling factors for the local space $\mathbb{P}_2^{(\alpha)} \otimes \mathbb{P}_2^{(\alpha)}$ of bi-degree $\mathbf{p} = (2, 2)$.

TB-splines are depicted in Figure 4.8 and lead to

$$\begin{aligned} N_{((1,2,4,5),(1,2,3,4)),(2,2)} &= \left(\frac{e^\alpha + 1}{2e^\alpha + 1}\right) N_{((1,2,3,4),(1,2,3,4)),(2,2)} \\ &\quad + \left(\frac{e^\alpha + 1}{e^\alpha + 2}\right) N_{((2,3,4,5),(1,2,3,4)),(2,2)}, \\ N_{((1,2,4,5),(2,3,4,5)),(2,2)} &= \left(\frac{e^\alpha + 1}{2e^\alpha + 1}\right) N_{((1,2,3,4),(2,3,4,5)),(2,2)} \\ &\quad + \left(\frac{e^\alpha + 1}{e^\alpha + 2}\right) N_{((2,3,4,5),(2,3,4,5)),(2,2)}. \end{aligned} \tag{4.9}$$

After the secondary split step, we arrive at the final set $\mathcal{S}_p(\mathcal{M}_2)$. The resulting weights of the TB-splines in this set are tabulated in Table 4.2.

Primary split				Final
	$N_{\Xi_1,2}^\sigma = N_{((0,1,2,4),(1,2,4,5)),(2,2)}^\sigma$ $\sigma_1 = 1$	$N_{\Xi_2,2}^\sigma = N_{((1,2,4,5),(1,2,4,5)),(2,2)}^\sigma$ $\sigma_2 = 1$	$N_{\Xi_3,2}^\sigma = N_{((2,4,5),(1,2,4,5)),(2,2)}^\sigma$ $\sigma_3 = 1$	σ_k
$N_{((0,1,2,3),(1,2,4,5)),(2,2)}^\sigma$	$\sigma_1^{(1)} = 1 \cdot \sigma_1$	$\sigma_2^{(1)} = \sigma_1^{(2)} + \left(\frac{e^\alpha+1}{2^{\alpha+1}}\right) \cdot \sigma_2$		1
$N_{((1,2,3,4),(1,2,4,5)),(2,2)}^\sigma$	$\sigma_1^{(2)} = \left(\frac{e^\alpha}{2^{\alpha+1}}\right) \cdot \sigma_1$	$\sigma_2^{(2)} = \left(\frac{e^\alpha+1}{e^\alpha+2}\right) \cdot \sigma_2$	$\sigma_3^{(1)} = \sigma_2^{(2)} + \left(\frac{1}{e^\alpha+2}\right) \cdot \sigma_3$	1
$N_{((2,3,4,5),(1,2,4,5)),(2,2)}^\sigma$			$\sigma_3^{(2)} = 1 \cdot \sigma_3$	1
$N_{((3,4,5,6),(1,2,4,5)),(2,2)}^\sigma$				1

Table 4.1: Split insertion (Iteration 1): The evolution of the weights for the three TB-splines that are traversed by the split γ_1 , converting \mathcal{M}_0 into \mathcal{M}_1 and leading to the creation of four new TB-splines, as illustrated in Figure 4.6. The scaling relations of the TB-splines that underwent splitting are given in (4.7). This iteration only involves the primary split step described in Remark 4.10 since no TB-splines undergo a secondary split.

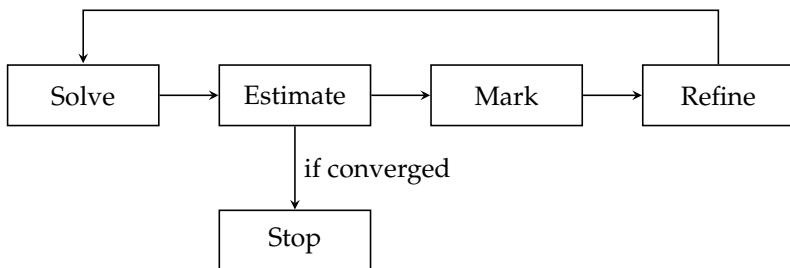
Primary split	Secondary split	
$N_{((1,2,3,4),(1,2,4,5)),(2,2)}^\sigma, N_{((1,2,4,5),(0,1,2,4)),(2,2)}^\sigma$	$N_{\Xi_4,2}^\sigma = N_{((1,2,4,5),(1,2,3,4)),(2,2)}^\sigma$	$N_{\Xi_5,2}^\sigma = N_{((1,2,4,5),(2,3,4,5)),(2,2)}^\sigma$
$N_{((1,2,4,5),(2,4,5,6)),(2,2)}, N_{((2,3,4,5),(1,2,4,5)),(2,2)}^\sigma$	$\sigma_4 = \left(\frac{e^{\alpha}}{2e^{\alpha}+1}\right)$	$\sigma_5 = \left(\frac{1}{e^{\alpha}+2}\right)$
		Final σ_k
$N_{\Xi_1,2}^\sigma = N_{((1,2,3,4),(1,2,3,4)),(2,2)}^\sigma$	$\sigma_1 = \frac{e^{\alpha}+1}{2e^{\alpha}+1}$	$\frac{3e^{2\alpha}+4e^{\alpha}+1}{(2e^{\alpha}+1)^2}$
$N_{\Xi_2,2}^\sigma = N_{((1,2,3,4),(2,3,4,5)),(2,2)}^\sigma$	$\sigma_2 = \left(\frac{e^{\alpha}+1}{e^{\alpha}+2}\right)$	$\frac{2(e^{\alpha}+1)^2}{2e^{2\alpha}+5e^{\alpha}+2}$
$N_{\Xi_3,2}^\sigma = N_{((1,2,4,5),(0,1,2,3)),(2,2)}^\sigma$	$\sigma_3 = 1$	1
$N_{\Xi_4,2}^\sigma = N_{((1,2,4,5),(1,2,3,4)),(2,2)}^\sigma$	$\sigma_4 = \frac{e^{\alpha}}{2e^{\alpha}+1}$	$\sigma_4^{(1)} = \sigma_1 + \left(\frac{e^{\alpha}+1}{2e^{\alpha}+1}\right) \cdot \sigma_4$
$N_{\Xi_5,2}^\sigma = N_{((1,2,4,5),(2,3,4,5)),(2,2)}^\sigma$	$\sigma_5 = \frac{1}{e^{\alpha}+2}$	$\sigma_5^{(1)} = \sigma_2 + \left(\frac{e^{\alpha}+1}{2e^{\alpha}+1}\right) \cdot \sigma_5$
		0
	Remove $N_{\Xi_4,2}^\sigma$	0
		0
$N_{\Xi_6,2}^\sigma = N_{((1,2,4,5),(3,4,5,6)),(2,2)}^\sigma$	$\sigma_6 = 1$	1
$N_{\Xi_7,2}^\sigma = N_{((2,3,4,5),(1,2,3,4)),(2,2)}^\sigma$	$\sigma_7 = \frac{e^{\alpha}+1}{2e^{\alpha}+1}$	$\frac{2(e^{\alpha}+1)^2}{2e^{2\alpha}+5e^{\alpha}+2}$
$N_{\Xi_8,2}^\sigma = N_{((2,3,4,5),(2,3,4,5)),(2,2)}^\sigma$	$\sigma_8 = \frac{e^{\alpha}+1}{e^{\alpha}+2}$	$\frac{e^{2\alpha}+4e^{\alpha}+3}{(e^{\alpha}+2)^2} \cdot \sigma_5$
		$\sigma_5^{(2)} = \sigma_8 + \left(\frac{e^{\alpha}+1}{e^{\alpha}+2}\right) \cdot \sigma_5$

Table 4.2: Split insertion (Iteration 2): The evolution of the weights for the four TB-splines that are traversed by the split γ_2 , converting \mathcal{M}_1 into \mathcal{M}_2 . The primary split step creates eight new TB-splines, as illustrated in Figure 4.7. The scaling relations of the TB-splines that underwent splitting are given in (4.8), and the resulting weights are tabulated in the first column. Subsequently, when the newly created TB-splines undergo a secondary split, two TB-splines are further divided as described by the scaling relations given in (4.9) and illustrated in Figure 4.8. This table serves to illustrate the evolving weights of TB-splines across consecutive refinement steps.

In conclusion, we want to emphasize that (4.7), (4.8), (4.9) present the two-scale relations for split insertion in an individual TB-spline, however, they do not take into account a possible existing weight of the TB-spline. The algorithmic evolution of the weights corresponding to each TB-spline after every split insertion is summarized in Tables 4.1 and 4.2.

4.3 Isogeometric analysis with LR TB-splines

In this section, we review the standard adaptive scheme for solving differential equations, which comprises the four (cyclic) key steps outlined below.



For the first step in the adaptive cycle, we consider the isogeometric Galerkin method based on LR TB-splines. We focus on the same second-order elliptic differential problems with homogeneous Dirichlet boundary conditions presented in (1.19). Then, we introduce different marking strategies based on either an ad hoc approach or a residual-based error estimator, depending on the type of problem we are handling. Finally, we present the refinement strategy we have incorporated to refine the functions marked as the result of previous steps in the adaptive cycle.

For isogeometric Galerkin methods, as described in Section 1.2, the selection of different subspaces \mathbb{W} in (1.21) results in different Galerkin methods and complying with the isogeometric approach, the fields of interest are described by means of spline basis functions. These basis functions in (1.21) are traditionally selected as tensor-product B-splines or NURBS. However, the characteristics outlined in Proposition 2.14 indicate that TB-splines are plug-to-plug compatible with classical (polynomial) B-splines and turn out to be a good choice as well Chapter 3. Furthermore, the set of LR TB-splines, as presented in Definition 4.9, is a suitable choice for incorporating adaptive refinement.

Let us select a space of bivariate LR Tchebycheffian splines as the approximation space in (1.21),

$$\mathbb{W} = \mathbb{S}_p(\mathcal{M}) = \left\langle N_{\Xi_k, p} \in \mathcal{S}_p(\mathcal{M}) \right\rangle. \quad (4.10)$$

This space is spanned by the set of TB-splines $\mathcal{S}_p(\mathcal{M})$ related to the local space $\mathbb{P}_{p_1}^{\mathcal{W}_1} \otimes \mathbb{P}_{p_2}^{\mathcal{W}_2}$ of bi-degree $p = (p_1, p_2)$ with maximal smoothness and roots \mathcal{W}_1 and \mathcal{W}_2 corresponding to the ECT-spaces in each parametric direction; see Definition 4.9 and Proposition 4.13. These TB-splines are constructed on an LR-mesh \mathcal{M} defined on the domain $\Omega = [0, 1]^2$.

The univariate building blocks for our basis functions are univariate TB-splines identified by ECT-spaces of the form (2.14). They offer a diverse array of algebraic polynomial, exponential, and trigonometric function combinations, equipped with a wide spectrum of shape parameters. Properly selecting the space's structure, incorporating various function types, along with appropriate shape parameters, is of vital importance to effectively harness the capabilities of the ECT-spaces. An in-depth study for the selection of ECT-spaces has been presented in Section 3.2 based on an (automatic) problem-driven strategy.

4.3.1 Adaptive strategy

Ever since the advent of polynomial LR B-splines, a prominent issue under discussion has been the quest for an optimal refinement strategy that aligns with objectives such as achieving linear independence, ensuring mesh quality, and accommodating grading and shape regularities. The same issue arises in the LR TB-spline setting. Thanks to the structural similarities between polynomial B-splines and TB-splines, the refinement strategies developed for polynomial LR B-splines are also applicable for LR TB-splines.

Local refinement strategy

We adopt the local refinement strategy introduced for polynomial LR B-splines in [61, Section 4.2] and known as structured mesh refinement. As opposed to the classical finite element method, in structured mesh refinement we select the LR TB-splines contributing most to the approximation error rather than the box-partition elements contributing most to the error and refine those LR TB-splines. This approach proves to be a more sensible choice since in LR-mesh refinement any newly inserted split must traverse the support of at least one LR TB-spline.

The structured mesh refinement is a dyadic refinement where all selected LR TB-splines are refined by halving all of the intervals in their local knot vectors, resulting in the insertion of a net of splits (located in the TB-spline supports) in accordance with Definition 4.9 (see also Algorithm 1). An LR-mesh obtained with the structured mesh refinement strategy is called a structured LR-mesh.

Figure 4.9 illustrates an example of (three iterations of) structured mesh

refinement for LR TB-splines of bi-degree $p = (2, 2)$. Specifically, Figure 4.9 (c) exhibits how the maximal smoothness is maintained in the LR-meshes across the meshlines. If the split already partially exists in the mesh then instead of increasing the multiplicity of already existing meshlines, we only elongate it to traverse the complete support of the marked TB-spline. And if the split already exists completely then instead of increasing the multiplicity of the corresponding meshlines, we neglect that split insertion, staying true to the configuration of LR-meshes with fixed multiplicity of 1 for all internal meshlines (as assumed in Section 4.2.1).

Remark 4.14. *One of the shortcomings of the structured mesh strategy is that it might produce a linearly dependent set of LR (T)B-splines. Many efforts have been made to achieve linear independence of LR B-splines. One of the suggested solutions was achieving local linear independence on each element in the mesh by enforcing the non-nested support (N_2S) property; see [19, 93]. These LR B-splines possess nice properties, such as*

- *the number of non-zero LR B-splines over an element σ is $(p_1 + 1)(p_2 + 1)$,*
- *they span the full local spline space,*
- *they form a partition of unity, without the use of scaling weights.*

Besides the function-based refinement strategies, there are also some box-based strategies, similar to the classical finite element approach, where the elements contributing the most to the error are selected for refinement. These strategies are hierarchical locally refined (HLR) meshing [19] and effective grading (EG) refinement [91], which produce LR-meshes with good grading while ensuring local linear independence. All of the aforementioned refinement strategies are developed for polynomial B-splines on LR-meshes, but are also applicable for LR TB-splines. It should be noted, however, that prioritizing the local linear independence of the LR (T)B-splines imposes many constraints on the refinement strategy. On the other hand, with the structured mesh refinement the generated meshes, at least in the region away from the boundary, are locally tensor meshes. Hence, the LR (T)B-splines defined in these zones of the mesh behave like the standard (T)B-splines and are locally linearly independent.

Marking strategy

After discussing the Solve and Refine steps in the adaptive cycle, we now focus on the remaining Estimate and Mark steps. As explained in Section 4.3.1, the structured mesh refinement strategy is based on refining LR TB-splines, hence we need to mark the TB-splines contributing maximally to the approximation error. This is estimated by two different strategies listed below.

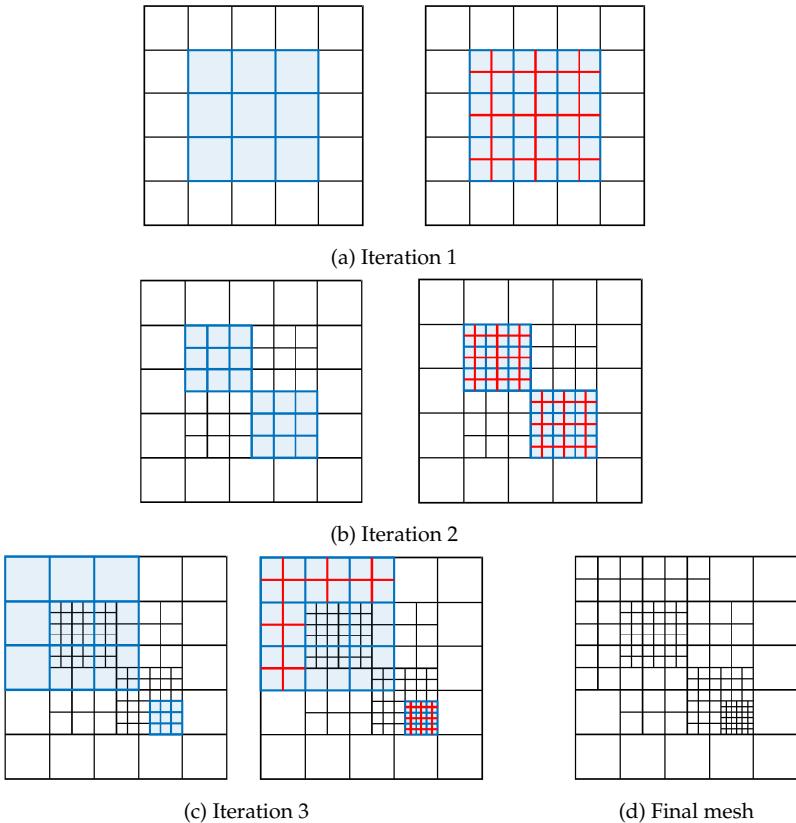


Figure 4.9: Example of structured mesh refinement for LR TB-splines of bi-degree $p = (2, 2)$. Figure (a), left: Initial tensor mesh \mathcal{M}_0 , with highlighted the support of the LR TB-spline to be refined. Figure (a), right: Set of splits to be inserted to accomplish structured mesh refinement, halving each interval in the support of the marked TB-spline. Figure (b): Refined LR-mesh \mathcal{M}_1 after the first iteration, with highlighted the supports of the two LR TB-splines to be refined in Iteration 2 and the set of splits to be inserted in this iteration. Figure (c), left: Refined LR-mesh \mathcal{M}_2 after the second iteration, with highlighted the supports of the two LR TB-splines to be refined in Iteration 3. Here the marked TB-splines have a different knot resolution. Figure (c), right: Illustration of how to maintain the multiplicity of the meshlines to one in the mesh. If the split to be inserted already partially exists in the mesh then it is only extended for the remaining part of the support of the TB-spline to be refined. Figure (d): Final LR-mesh \mathcal{M}_3 obtained.

- Problem-oriented ad hoc refinement: This method involves leveraging a priori knowledge about the problem's behavior. By understanding the problem's geometric feature and identifying specific regions in the domain that require refinement, we can selectively mark functions with support in those areas.
- Error-based automatic refinement: This method uses the residual-based error indicator in the L^2 -norm. Assuming we have computed the discrete solution $u_{\mathbb{W}}$ of the model problem (1.19), the error indicator η_σ on an element σ within an LR-mesh \mathcal{M} is given by

$$\eta_\sigma := \|f - \mathcal{L}u_{\mathbb{W}}\|_{L^2(\sigma)}. \quad (4.11)$$

This quantity is defined element-wise, while we need to mark basis functions for the structured mesh refinement strategy. Therefore, we first mark elements with high error and then mark all the TB-splines that have support over those elements. Our element selection is based on a proper threshold criterion,

$$\eta_\sigma \geq \psi \cdot \max_\delta \eta_\delta, \quad \psi \geq 0. \quad (4.12)$$

The marking parameter ψ facilitates refinement in the spectrum from global refinement to no refinement at all by selecting $\psi = 0$ and $\psi > 1$, respectively. Note that the proportion of elements marked can vary with each step since the computation of the threshold depends only on the magnitude of the maximum of the error, without considering the distribution of estimated errors.

Remark 4.15. *As an alternative to marking elements with errors exceeding a certain threshold as in (4.12), we can also label a specific percentile of elements based on their error distribution. In this chapter, we just use the criterion defined in (4.12) and maintain the same value for the marking parameter ψ across all iterations. This choice is made because the nature of all the case studies discussed in Section 4.4 involves small regions with substantial error magnitudes. Consequently, this strategy effectively restricts the refinement region to a minimum extent.*

4.4 Numerical results of adaptive IgA with LR TB-splines

In this section, we present some case studies using LR TB-splines in isogeometric Galerkin discretizations of second-order problems. In all case studies, the approximation spaces are taken as bivariate Tchebycheffian

spline spaces of the form (4.10) over LR-meshes, related to the local space $\mathbb{P}_{p_1}^{\mathcal{W}_1} \otimes \mathbb{P}_{p_2}^{\mathcal{W}_2}$ of bi-degree $p = (p_1, p_2)$ with roots w_1 and w_2 corresponding to each parametric direction. The LR-mesh refinement is carried out with the structured mesh refinement strategy described in Section 4.3.1. The initial mesh \mathcal{M}_0 is a tensor mesh that partitions the domain with $m + 1$ equidistant breakpoints, i.e., the distance between two consecutive breakpoints is $h_0 = 1/m$. Given that the structured mesh refinement follows a dyadic pattern, the minimum distance between two consecutive mesh-lines along one parametric direction for each iteration of refinement advances as

$$h_\ell = h_{\ell-1}/2, \quad \ell = 1, 2, \dots$$

Again, we focus on the general second-order problem described in (1.26), where similar to the tensor-product case in Section 3.3, the homogeneous boundary conditions are satisfied pointwise exactly. While, in the non-homogeneous case, the boundary function is approximated in the underlying LR Tchebycheffian spline space by a suitable approximation strategy and subsequently the reduction to the homogeneous case is considered, so dealing again with a special instance of the problem (1.19). It is important to remark that, when discontinuous Dirichlet boundary conditions are involved, a proper boundary treatment is imperative. In such a case, in order to avoid oscillations along the boundary, we approximate the boundary function by means of the shape-preserving Schoenberg quasi-interpolant (see Section 2.2.2 and [75, 103]), i.e, we consider the linear combination of the boundary TB-splines whose coefficients are obtained by evaluating the boundary function at the corresponding Greville abscissae (assuming that linear polynomials belong to the ECT-space of interest).

For the (numerical) computation of the integrals required in the construction of the linear system in (1.22), we employ element-wise Gaussian quadrature rules. The choice of quadrature points for discretizations involving TB-splines is a non-trivial task, often necessitating quadrature rules of higher order compared to those used for classical polynomial splines. In all case studies presented we use $5p$ quadrature points in each element for LR TB-splines consisting of exponential functions with large shape parameters.

4.4.1 Case study 6: Poisson problem on a unit square

In this case study we address a Poisson problem commonly used in the literature as a benchmark for adaptive refinement schemes; see [31, 61, 93]. We consider (1.26) with $\kappa = 1$, $c = 0$, $\mathbf{a} = \mathbf{0}$, and f obtained from the exact

solution

$$u(x, y) = \arctan \left(100 \left(\sqrt{(x - 1.25)^2 + (y + 0.25)^2} - \frac{\pi}{3} \right) \right). \quad (4.13)$$

While the problem may seem mathematically smooth since we are just solving for the Laplacian of the analytical solution, the highly varying right-hand side introduces complexity in the solution. The solution has rapid changes across the arc of the circumference

$$(x - 1.25)^2 + (y + 0.25)^2 = (\pi/3)^2, \quad (4.14)$$

traversing the domain $\Omega = [0, 1]^2$ and resulting into an internal sharp layer.

In the literature, bi-quadratic polynomial splines are prominently used for the computation of approximate solutions of this problem. Here, we select Tchebycheffian splines of the same degree to prove the efficacy of Tchebycheffian splines. Since the problem exhibits an internal sharp layer, a Tchebycheffian spline space with exponential functions (with suitable parameters) is a good choice. Following (2.14), we take the local space as

$$\mathbb{P}_2^{(\alpha, -\alpha)} \otimes \mathbb{P}_2^{(\alpha, -\alpha)} = \langle 1, e^{\alpha x}, e^{-\alpha x} \rangle \otimes \langle 1, e^{\alpha y}, e^{-\alpha y} \rangle, \quad \text{with } \alpha = 10.$$

We use an ad hoc refinement strategy as outlined in Section 4.3.1, where we utilize a priori knowledge about the problem to identify the specific regions in the domain that require refinement. More precisely, we mark the TB-splines that have support touching/overlapping the arc (4.14) and refine them according to the structured mesh refinement strategy. The plots of some LR-meshes are illustrated in Figure 4.10 and an approximate solution is visualized in Figure 4.11.

Figure 4.12 illustrates the convergence of the error in L^∞ -norm, computed by sampling the approximate and exact solutions on a uniform grid consisting of 1001 points along each direction in the domain. We assess the convergence rates of Tchebycheffian splines in comparison to classical polynomial splines on LR-meshes, starting from a coarse tensor mesh with $h_0 = 1/8$ to $h_5 = 1/256$ after five iterations of local refinement. We also compare the results of their tensor mesh counterpart with the same resolution. The plot clearly shows that the approximate solution obtained by TB-splines on LR-meshes converges faster than their tensor mesh equivalent of the same resolution. Table 4.3 shows the L^∞ -error for all LR-mesh iterations and it highlights smaller error for TB-splines compared to B-splines for spaces of equivalent dimension, especially on the coarser meshes.

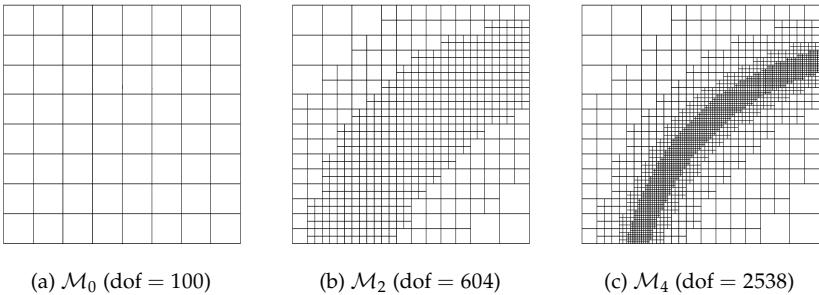


Figure 4.10: Case study 6. Some LR-meshes for the solution of the Poisson problem in (4.13), obtained with an ad hoc refinement strategy. The corresponding number of degrees of freedom (dof) for bi-degree $p = (2, 2)$ are mentioned.

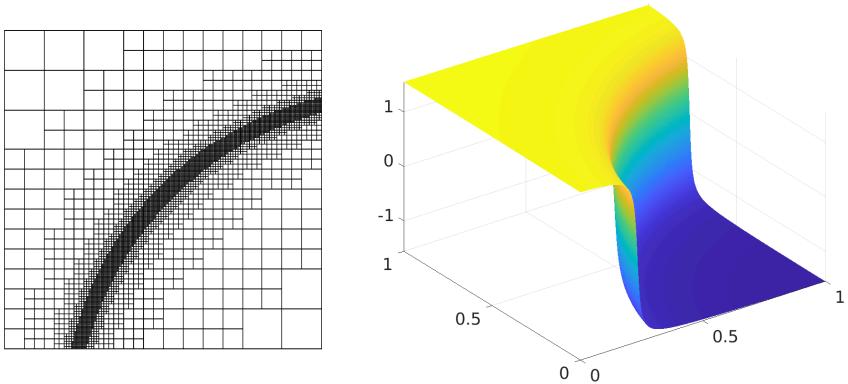


Figure 4.11: Case study 6. Plot of the approximate solution (right) for the Poisson problem with exact solution (4.13), obtained by LR TB-splines related to the local space $\mathbb{P}_2^{(10,-10)} \otimes \mathbb{P}_2^{(10,-10)}$ on the mesh \mathcal{M}_5 with dof = 5101 (left).

4.4.2 Case study 7: Reaction-diffusion problem on a square

Here we solve a reaction-diffusion problem (1.26) on a square, with

$$\kappa = 10^{-3}, \quad c = 1, \quad \mathbf{a} = \mathbf{0}, \quad \mathbf{f} = \mathbf{0}; \quad (4.15)$$

see [4, 80]. The boundary condition for this problem is set to zero except near the corners as illustrated in Figure 4.13 (left). The treatment of this discontinuous Dirichlet boundary condition is carried out with a Schoenberg quasi-interpolant, see Section 2.2.2.

The dominance of reaction over diffusion is defined by the Damköhler number, defined as the ratio of the reaction coefficient (c) to the diffusion

4.4. Numerical results of adaptive IgA with LR TB-splines

h_ℓ	dof	$\mathbb{P}_2^{(10,-10)} \otimes \mathbb{P}_2^{(10,-10)}$	$\mathbb{P}_2 \otimes \mathbb{P}_2$
1/8	100	1.5873929998	7.9295697929
1/16	277	0.6163964505	1.7534324399
1/32	604	0.2982445346	0.3859733442
1/64	1249	0.0937572015	0.0961967520
1/128	2538	0.0139797634	0.0139829152
1/256	5101	0.0012123306	0.0012128007

Table 4.3: Case study 6. Comparison of the L^∞ error for the Poisson problem with exact solution in (4.13) when considering TB-splines and B-splines of bi-degree $p = (2, 2)$ on a sequence of LR-meshes, obtained with an ad hoc refinement strategy, some of them are illustrated in Figures 4.10 and 4.11.

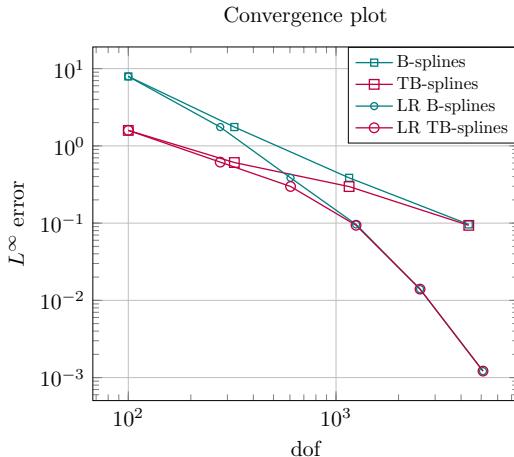


Figure 4.12: Case study 6. Convergence plot of the L^∞ error for the Poisson problem with exact solution in (4.13) when considering TB-splines and B-splines of bi-degree $p = (2, 2)$ on tensor meshes and LR-meshes.

coefficient (κ). In this case study, the system is strongly reaction dominated, as the Damköhler number is 10^3 . Therefore, we expect the solution to be zero throughout the domain, except near the corners, where enforcing the boundary condition rapidly spikes the solution to one.

The approximate solution belonging to a polynomial spline space tends to exhibit spurious oscillations around the jump, unless the discretization is fine enough. Using locally refined structures can help with suppressing the oscillations and resolve the spikes near the corners. In addition, the Tchebycheffian spline's ability in capturing sharp layers can further reduce such spurious oscillations, without refining too much. Therefore,

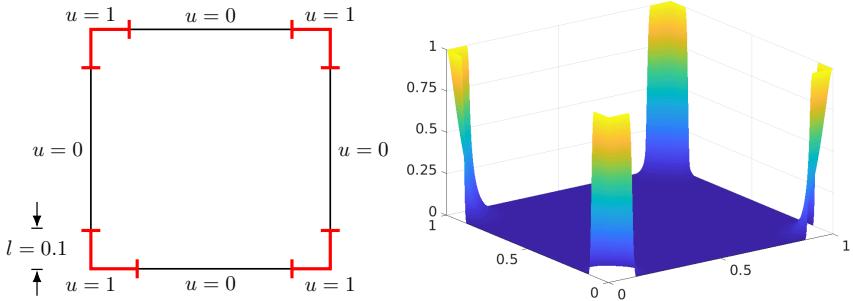


Figure 4.13: Case study 7. Left: The domain $\Omega = [0, 1]^2$ with the Dirichlet boundary conditions, where all corners are set to $u = 1$ up to the length $l = 0.1$ (red) and the rest of the boundary is set to $u = 0$. Right: Plot of an approximate solution for the reaction-diffusion problem specified in (4.15), which is computed on a very fine tensor mesh with B-splines of bi-degree $p = (3, 3)$ with $h = 1/1000$ and dof = 1006009. We use this solution as a reference to compare the other results in this case study.

we select again a Tchebycheffian spline space with exponential functions, where the local space is given by

$$\mathbb{P}_3^{(\alpha, -\alpha)} \otimes \mathbb{P}_3^{(\alpha, -\alpha)} = \left\langle 1, x, e^{\alpha x}, e^{-\alpha x} \right\rangle \otimes \left\langle 1, y, e^{\alpha y}, e^{-\alpha y} \right\rangle, \quad \text{with } \alpha = 35.$$

We use a residual-based error indicator to guide the local refinement. Given that $f = 0$, the element-wise error estimator in (4.11) becomes

$$\eta_\sigma = \|\nabla \cdot (\kappa \nabla u_W) - c u_W\|_{L^2(\sigma)}, \quad (4.16)$$

and we set the marking parameter to $\psi = 0.95$ in the marking criterion (4.12).

In the absence of an explicit expression for the exact solution of this problem, we check the accuracy of the approximate solutions by comparing them with a high-fidelity B-spline solution. This B-spline solution, depicted in Figure 4.13 (right), is computed on a tensor mesh with an element size of $h = 1/1000$, with B-splines of bi-degree $p = (3, 3)$ and dof = 1006009. The contour plots in Figure 4.14 illustrate the error in solutions obtained by LR TB-splines and LR B-splines, both with respect to the reference B-spline solution, along with the LR-meshes on which the system is solved. The approximate solutions are evaluated on a uniform grid of 2001×2001 points. Analyzing the contour plots, it shows that LR TB-spline solutions exhibit smaller oscillation areas, and the magnitude of

4.4. Numerical results of adaptive IgA with LR TB-splines

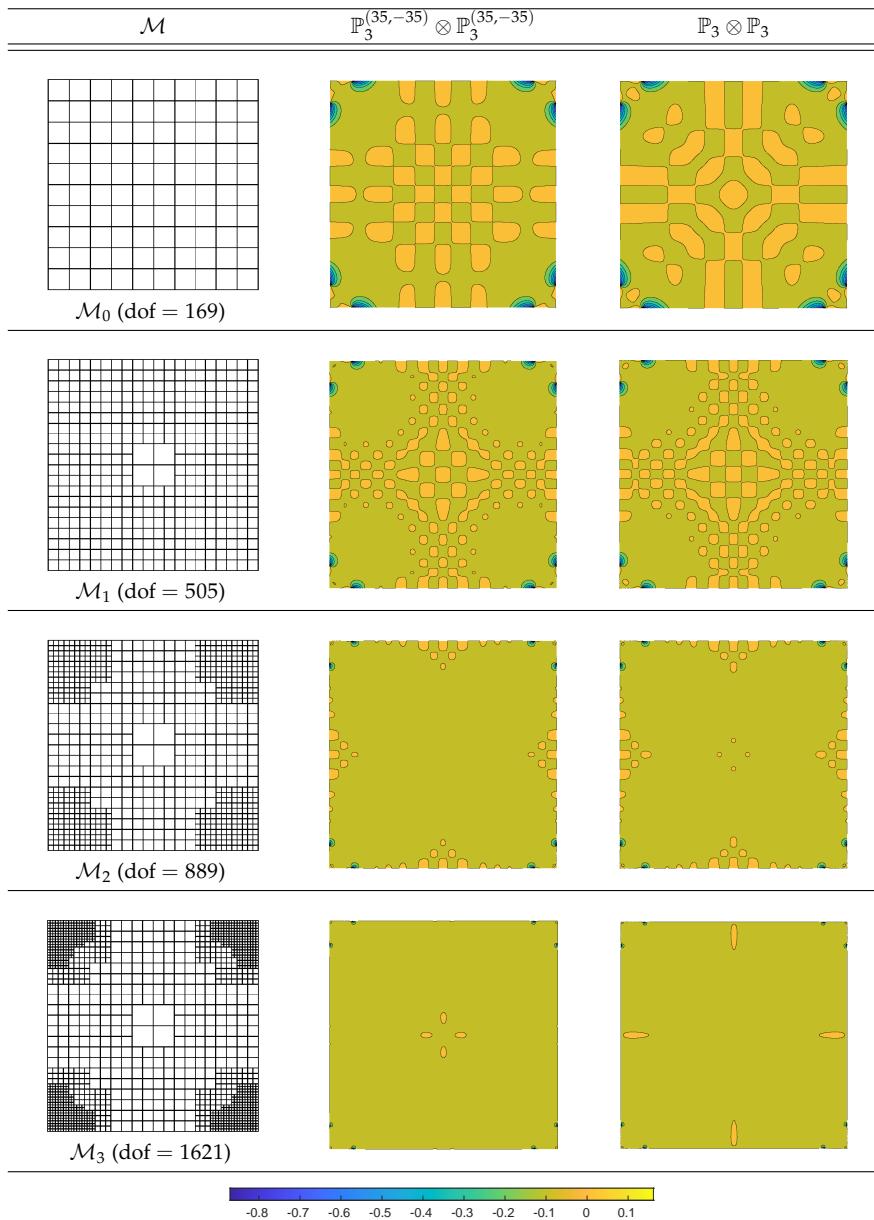


Figure 4.14: Case study 7. Some LR-meshes for the solution of the reaction-diffusion problem specified in (4.15), where the refinement is guided by the residual-based error indicator in (4.16) with marking parameter $\psi = 0.95$, along with the contour plots of the error in solutions obtained by TB-splines and B-splines of bi-degree $p = (3, 3)$, both with respect to the reference solution shown in Figure 4.13.

these oscillations is also reduced, especially on the coarser meshes. This comparison confirms the capability of TB-splines in capturing sharp layers while eliminating unwanted spurious oscillations.

4.4.3 Case study 8: Advection–diffusion problem with internal sharp layer on a square

In this case study we consider a standard benchmark problem with advection flow skew to any parametric direction [55, 79, 97, 117]. We solve an advection-diffusion problem (1.26) on a square, with

$$\kappa = 1, \quad c = 0, \quad \mathbf{a} = \alpha (\cos(\theta), \sin(\theta))^T, \quad \theta = \frac{\pi}{4}, \quad \alpha = 10^4, \quad f = 0, \quad (4.17)$$

and the discontinuous Dirichlet boundary conditions along with the sharp internal layer generated along the advection flow at angle θ and the boundary layers as shown in Figure 3.12. The jump in the boundary conditions at the point $(0, 0.2)$ creates an inner sharp layer aligned with the advection flow direction identified by $(\cos(\theta), \sin(\theta))$. Additionally, the solution also exhibits a sharp boundary layer.

This problem is advection-dominated since the global Péclet number (1.27) is $\mathbf{Pe}_g = \alpha = 10^4$. Such problems tend to exhibit spurious oscillations in their approximate solutions when polynomial splines are used, until the discretization is fine enough to resolve the sharp layers featured by the exact solution. A common practice to overcome this issue is to use stabilization methods such as the SUPG and GLS method; see [23, 56]. While these stabilization methods effectively eliminate spurious oscillations, it is also important to note that they tend to “smooth out” the layers featured by the exact solution. Moreover, their effectiveness is hugely dependent on the choice of some parameters appearing in the various stabilization methods. It has been showcased in Section 3.3.5 that Tchebycheffian splines can offer a flexible alternative for such advection-dominated problems without the need for stabilization and better localization of boundary and internal layers.

To capture the sharp layers of the problem specified in (4.17), an adequate choice of Tchebycheffian spline space contains exponential functions with suitable parameters [97]. These parameters are dependent on the components of the advection flow velocity in the two parametric directions, given as

$$\begin{aligned} & \mathbb{P}_4^{(\alpha \cos(\theta))} \otimes \mathbb{P}_4^{(\alpha \sin(\theta))} \\ &= \langle 1, x, x^2, x^3, e^{\alpha \cos(\theta)x} \rangle \otimes \langle 1, y, y^2, y^3, e^{\alpha \sin(\theta)y} \rangle. \end{aligned} \quad (4.18)$$

For the adaptive refinement we use the ad hoc refinement strategy described in Section 4.3.1. Since we expect the internal and boundary layers to be sharp, we can refine along these layers. Figure 4.15 illustrates the LR-mesh \mathcal{M}_2 for $p = (4, 4)$ obtained with this ad hoc strategy after two iterations, along with the approximate solutions using the LR TB-spline space related to the local space (4.18) and the LR B-spline space on the mesh \mathcal{M}_2 . Since this mesh is not fine enough, the solution with plain LR B-splines would exhibit huge spurious oscillations. Hence, it is necessary to use stabilization in the LR B-spline solution to have a comparable solution against the unstabilized LR TB-spline solution. The discontinuous Dirichlet boundary condition in all the cases is imposed by the Schoenberg quasi-interpolant.

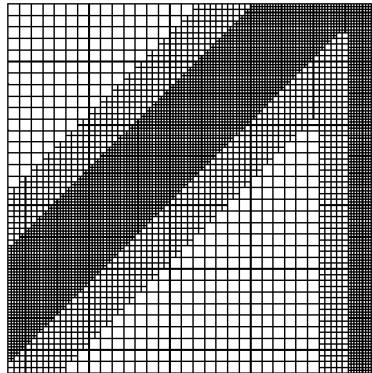
For comparison, we also solve the same problem with B-splines on an LR-mesh after many refinement steps to obtain a fine enough mesh such that the stabilization of the solution is not required anymore. The resulting mesh \mathcal{M}_6 is presented in Figure 4.16, where the mesh is obtained using the same ad hoc refinement strategy as described before for $p = (4, 4)$.

Table 4.4 summarizes the values of maximal over- and undershoot in the neighborhood of the layers evaluated on a uniform grid of 3001×3001 points along each direction for the different setups of the meshes and spline spaces, together with the resolution of the mesh h_ℓ and the number of degrees of freedom (dof).

The approximate solution with LR TB-splines on an LR-mesh gives a similar result compared to the TB-spline solution on a tensor mesh with the same $h_2 = 1/128$, showing that the use of an LR-mesh for TB-splines results in the same level of accuracy with fewer degrees of freedom (about 2.5 times fewer in this case). The stabilized LR B-spline solution on mesh \mathcal{M}_2 does eliminate the spurious oscillations but the corresponding solution is too smooth, resulting in a poor localization of the internal and boundary layers. On the other hand, the unstabilized LR B-spline solution on a very fine mesh \mathcal{M}_6 localizes the layers accurately but at a very high cost. In conclusion, TB-splines on an LR-mesh are able to eliminate spurious oscillations without the need for stabilization in the approximate solution, while still capturing the sharp layers precisely.

4.5 Conclusive remarks

Multivariate versions of TB-splines can be easily obtained by taking tensor products and popular local refinement technologies, based on local tensor products, can also be applied in the Tchebycheffian setting. For example, the knot insertion procedure allows for defining LR TB-splines as a natural generalization of LR B-splines.



(a) LR-mesh \mathcal{M}_2 for $p = (4, 4)$ with $h_2 = 1/128$ and dof = 6957 obtained with ad-hoc refinement strategy.

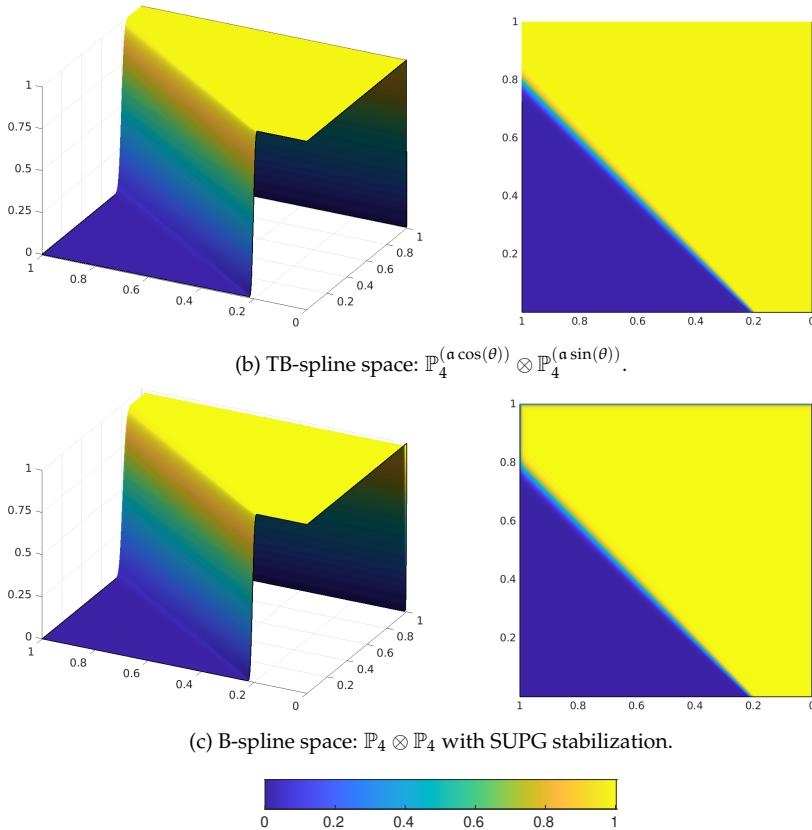


Figure 4.15: Case study 8. Plot of the LR-mesh \mathcal{M}_2 and the corresponding approximate solutions using different TB-spline spaces of $p = (4, 4)$ for the advection-diffusion problem specified in (4.17) with advection skew at $\theta = 45^\circ$ and global Péclet number $\text{Pe}_g = 10^4$.

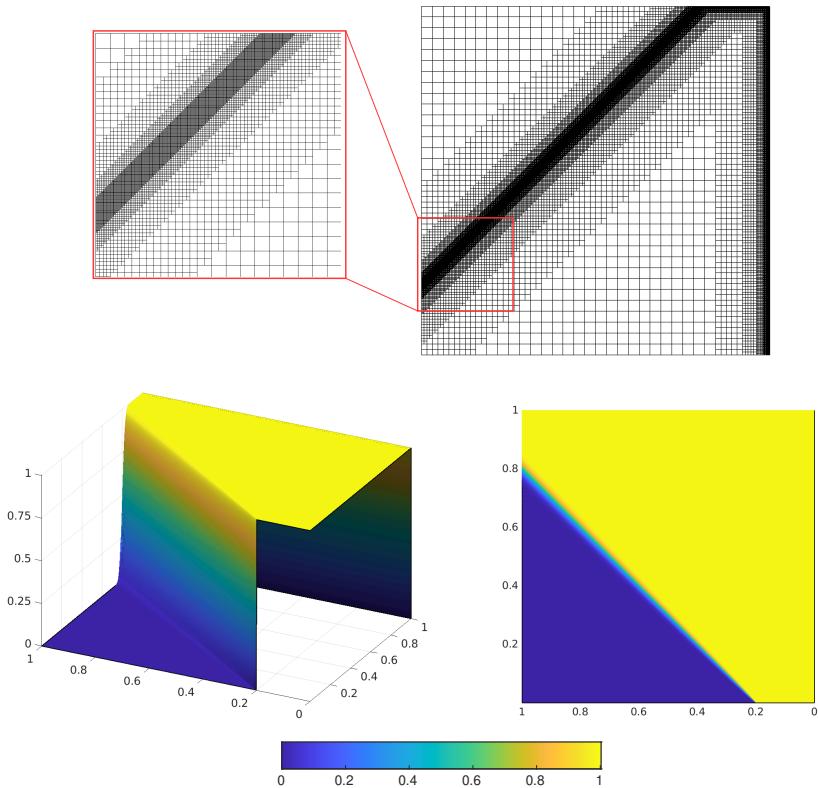


Figure 4.16: Case study 8. Plot of the LR-mesh \mathcal{M}_6 of $p = (4, 4)$ with $h_6 = 1/2048$ and $\text{dof} = 121283$, obtained with ad hoc refinement, and the corresponding approximate solution using the B-spline space (i.e., the local space is $\mathbb{P}_4 \otimes \mathbb{P}_4$) without SUPG stabilization for the advection-diffusion problem specified in (4.17) with advection skew at $\theta = 45^\circ$ and global Péclet number $\text{Pe}_g = 10^4$.

TB-spline space	Local refinement	SUPG stabilization	h_ℓ	dof	max	min
$\mathbb{P}_4^{(\alpha \cos(\theta))} \otimes \mathbb{P}_4^{(\alpha \sin(\theta))}$	Yes	No	1/128	6957	1.0058	-6.5089×10^{-3}
$\mathbb{P}_4^{(\alpha \cos(\theta))} \otimes \mathbb{P}_4^{(\alpha \sin(\theta))}$	No	No	1/128	17424	1.0059	-6.5036×10^{-3}
$\mathbb{P}_4 \otimes \mathbb{P}_4$	Yes	Yes	1/128	6957	1.0019	-1.6772×10^{-3}
$\mathbb{P}_4 \otimes \mathbb{P}_4$	Yes	No	1/2048	121283	1.0071	-1.3056×10^{-3}

Table 4.4: Case study 8. Comparison of maximum and minimum of the TB-spline solution (without stabilization) against the B-spline solution (with SUPG stabilization) on the same LR-mesh \mathcal{M}_2 as in Figure 4.10, the B-spline solution (without stabilization) on the very fine LR-mesh \mathcal{M}_6 as in Figure 4.16, and the TB-spline solution on a tensor mesh of the same resolution as in \mathcal{M}_2 for $p = (4, 4)$ and $\alpha = 10^4$.

In this chapter we have been focusing on Tchebycheffian splines whose pieces belong to null-spaces of constant-coefficient linear differential operators. Although this is a more restricted class, it offers the freedom of combining algebraic polynomial, exponential, and trigonometric functions, thus providing a large family of approximation spaces for isogeometric methods that are flexible and effective both from the analytical and geometrical point of view. They can be identified according to problem-oriented selection strategies [97]. Moreover, the corresponding (LR) TB-splines can be easily incorporated in any software library supporting polynomial (LR) B-splines to enrich its capability, thanks to efficient evaluation and manipulation routines recently developed for this class [115].

LR TB-splines are a viable spline technology on unstructured meshes and offer a valid alternative to classical LR B-splines in adaptive isogeometric analysis. We have shown that the adaptive strategy combined with problem-oriented approximation spaces may create a synergistic effect and may produce results of similar quality with less levels of refinement, and so fewer degrees of freedom, without the need for possible stabilization, compared to the polynomial setting.

Non-uniform degree LR-splines

The construction of non-uniform degree LR-splines utilizes the degree elevation operation, in which the degree of a spline is increased, while preserving its smoothness [95]. With the local tensor-product structure of LR B-splines, it becomes feasible to perform local degree elevation on an individual B-spline. This capability allows for local degree elevation, enabling the construction of a solution space with non-uniform degrees, which proves advantageous in addressing higher-order isogeometric methods involving local features. Moreover, due to the two-scale relation (see Section 1.1.4), the functions before refinement are consistently replaced with new refined ones, ensuring the nestedness of the space.

In this chapter, we devise an algorithmic approach to construct LR-meshes supporting non-uniform degree LR B-splines through sequential h - and p -refinement, also known as hp -refinement, by combining local knot insertion with local degree elevation, and we discuss their application in isogeometric analysis.

To construct such LR-meshes with non-uniform degree LR B-splines, we leverage the concept of polynomial splines over locally refined box-partitions in their full generality, as formulated in [41]. In this general setup, the LR-meshes allow for different order of continuity across different meshlines, which is in contrast to the restricted setup presented in Section 4.2.1, where all meshlines in an LR-mesh had a fixed multiplicity equal to one, except on the boundary, where it is fixed to maximum.

In Section 5.1 we introduce an hp -refinement algorithm aimed at constructing non-uniform degree splines through LR-meshes. We discuss the data structures used for the implementation of the locally refined B-splines in Section 5.2. Finally, Section 5.3 presents various numerical results, showcasing the performance of adaptive isogeometric analysis based on non-uniform degree LR B-splines for the solution of few classical benchmark differential problems.

The results in this chapter are published in:

K. Raval, R. R. Hiemstra, K. A. Johannessen, T. Dokken, and A. Raffo, *Local h -, p - and hp -adaptivity with LR B-splines*, In preparation.

5.1 Non-uniform degree B-splines on LR-meshes

In this section, we define non-uniform degree B-splines on Locally Refined meshes (LR-meshes) through an hp -refinement algorithm. Analogous to the construction of uniform degree LR B-splines, we begin with an open tensor mesh and its corresponding set of tensor B-splines. Subsequently, we refine the mesh by either inserting splits, one at a time (referred to as the h -refinement step), or by elevating the degree of a spline in a specific direction (referred to as the p -refinement step). This process of knot insertion is repeated until all B-splines in the considered set have minimal support (see Definition 4.6). By alternating these two operations on an LR-mesh, we implement the hp -refinement strategy. Below, we outline the precise steps involved in executing the hp -refinement strategy.

5.1.1 hp -refinement algorithm for LR B-splines

The standard LR B-spline framework (see, e.g., Definition 4.9) allows to sequentially perform local h -refinement via the insertion of local splits in an initial tensor mesh. The only assumption for the inserted split is that at least one tensor-product B-spline has its support fully traversed by it. Similarly, for p -refinement, it is natural to directly select B-splines to be refined. In order to have a consistent approach we will align the procedures for selecting B-splines for h -refinement with p -refinement. Local refinement of an LR B-spline yields a new refined set of LR B-splines. By basing the refinement rules on the two-scale relations presented in Sections 1.1.3 and 1.1.4, we achieve that the refinement is nested.

An additional desirable property is achieving a partition of unity, which can be accomplished by introducing weights to the LR B-splines. This ensures that for a given set of LR B-splines $\mathcal{S}_p(\mathcal{M}_j)$, the set of weighted LR B-splines satisfies:

$$\sum_{N_{\Xi_k,p} \in \mathcal{S}_p(\mathcal{M}_j)} N_{\Xi_k,p}^\sigma(x) := \sum_{N_{\Xi_k,p} \in \mathcal{S}_p(\mathcal{M}_j)} \sigma_k N_{\Xi_k,p}(x) = 1, \quad x \in \Omega,$$

for some positive weights $\sigma_k \in \mathbb{R}$. Because of the structural similarities of TB-splines with B-splines, the definition of weighted LR (T)B-splines in Definition 4.12 applies here as well, of course additionally taking into account the two-scale relation for p -refinement.

Let $\mathcal{S}_p(\mathcal{M}_{j-1})$ be a given set of bivariate minimal support B-splines on the LR-mesh \mathcal{M}_{j-1} . A simple choice could be a tensor mesh and the corresponding set of tensor-product B-splines. Then we obtain the new set $\mathcal{S}_p(\mathcal{M}_j)$ in the following manner, depending on the selection of some B-splines to be h -refined or p -refined.

We schematize h -refinement as follows. Given

1. a set $\mathcal{S}_p(\mathcal{M}_{j-1})$ of LR B-splines over a mesh \mathcal{M}_{j-1} ,
2. a set $\mathcal{R} \subset \mathcal{S}_p(\mathcal{M}_{j-1})$ of B-splines to be h -refined,
3. a parameter direction $i = 1, 2$ for splitting,
4. a knot value $\hat{\xi}$ to be used for the splitting,
5. the continuity C^k for the corresponding meshline,

then

Step 1 Intersect the union of the supports of the selected B-splines with the straight line determined by inputs (3-4). Update \mathcal{M}_j by adding the minimal meshlines produced by this intersection, if any.

Step 2 Apply the two scale relation of (1.14) to the B-splines in \mathcal{R} . Remove the above-split B-splines from $\mathcal{S}_p(\mathcal{M}_j)$. Add any newly created B-spline or, whether already existing, update its weight.

Step 3 Apply the two scale relations of (1.14) to all those B-spline in $\mathcal{S}_p(\mathcal{M}_j)$ without minimal support (i.e., lacking minimal support) on the updated mesh. Remove the above-split functions from $\mathcal{S}_p(\mathcal{M}_j)$. Add any newly created B-spline or, whether already existing, update its weight.

Moreover, the p -refinement works as follows. Given

1. a set $\mathcal{S}_p(\mathcal{M}_{j-1})$ of LR B-splines over a mesh \mathcal{M}_{j-1} ,
2. a set $\mathcal{R} \subset \mathcal{S}_p(\mathcal{M}_{j-1})$ of B-splines to be p -refined,
3. a parameter direction $i = 1, 2$ for degree elevation,

then

Step 1 Increase the multiplicity of all meshlines in the support of the selected B-spline for refinement, such that only the multiplicity of the line segment in the support is increased.

Step 2 Apply the two scale relation of (1.15) to all the B-splines in \mathcal{R} . Remove the above degree elevated B-splines from $\mathcal{S}_p(\mathcal{M}_j)$. Add any newly created B-spline or, whether already existing, update its weight.

The above procedure for hp -refinement is implemented in Algorithm 2.

Algorithm 2: *hp*-refinement of LR B-splines

Input : $\mathcal{S}_p(\mathcal{M}_{j-1})$ B-splines set
 $\mathcal{R} \subset \mathcal{S}_p(\mathcal{M}_{j-1})$ Subset of B-splines to be refined
 $i = 1, 2$ Parameter direction
 $\hat{\xi}$ Knot value
 r Continuity
 $\left. \begin{array}{l} \text{Parameter direction} \\ \text{Knot value} \\ \text{Continuity} \end{array} \right\} \gamma_j \text{ split}$

Output: $\mathcal{S}_p(\mathcal{M}_j)$ Refined TB-spline set

```

1:  $\mathcal{S}_p(\mathcal{M}_j) \leftarrow \mathcal{S}_p(\mathcal{M}_{j-1})$  and  $\hat{\mathcal{S}}_p(\mathcal{M}_j) \leftarrow \emptyset$ ;  

   /* STEP 1: update the mesh */  

2:  $\mathcal{M}_j = \text{UPDATE\_MESH}(\mathcal{M}_{j-1}, \mathcal{R}, i, \hat{\xi}, r)$ ; // Add minimal meshlines  

3: if p-refinement step then  

   /* STEP 2: B-splines degree elevation */  

4:   for  $\forall N_{\Xi_k, p} \in \mathcal{R}$  do  

5:      $\mathcal{S}_p(\mathcal{M}_j) = \{\mathcal{S}_p(\mathcal{M}_j) \setminus N_{\Xi_k, p}\} \cup \text{DEGREE\_RAISE}(N_{\Xi_k, p}, i)$ ;  

     // Apply (1.15) to  $N_{\Xi_k, p}$   

6:   end  

7: end  

8: else  

   /* STEP 2: B-splines split */  

9:    $\hat{\mathcal{S}}_p(\mathcal{M}_j) = \text{KNOT\_INSERTION}(\mathcal{R}, i, \hat{\xi}, r)$ ; // Apply (1.14) to  $\mathcal{R}$   

10:   $\mathcal{S}_p(\mathcal{M}_j) := \{\mathcal{S}_p(\mathcal{M}_{j-1}) \setminus \mathcal{R}\} \cup \hat{\mathcal{S}}_p(\mathcal{M}_j)$ ; // Remove old  

    B-splines + update weights  

   /* STEP 3: minimal support recovering */  

11:  while  $\exists N_{\Xi_k, p} \in \hat{\mathcal{S}}_p(\mathcal{M}_j)$  without minimal support in  $\mathcal{M}_j$  do  

12:    if  $N_{\Xi_k, p}$  has not h-minimal support then  

13:       $\hat{\mathcal{S}}_p(\mathcal{M}_j) =$   

       { $\hat{\mathcal{S}}_p(\mathcal{M}_j) \setminus N_{\Xi_k, p}$ }  $\cup \text{KNOT\_INSERTION}(N_{\Xi_k, p}, i, \hat{\xi}, r)$   

14:    end  

15:  end  

16: end

```

The span of the set of non-uniform degree B-splines on an LR-mesh is denoted by

$$\mathbb{S}_p(\mathcal{M}_j) := \langle N_{\Xi_k, p} \in \mathcal{S}_p(\mathcal{M}_j) \rangle.$$

Note that

1. all $N_{\Xi_k, p} \in \mathcal{S}_p(\mathcal{M}_j)$ have minimal support,
2. $\mathbb{S}_p(\mathcal{M}_j)|_\sigma \subset \mathbb{P}^{\max(p)_\sigma}$ for any $\sigma \in \mathcal{E}$,
3. $\mathbb{S}_p(\mathcal{M}_j)|_\sigma \supset \mathbb{P}^{\min(p)_\sigma}$ for any $\sigma \in \mathcal{E}$,

where $\max(p)_\sigma$ (resp., $\min(p)_\sigma$) means the maximum (resp., minimum) degree of the B-spline from $\mathcal{S}_p(\mathcal{M})$ with support over that element σ . Property three implies polynomial reproduction on the element.

Another desirable property to have in the LR spline spaces is the linear independence, but it is not build in by construction. Linear independence can be maintained by using certain refinement strategies. See Remark 4.14 and references therein for a detailed discussion upon the linear independence of the uniform degree LR B-splines. The non-uniform degree setting requires further study regarding this topic.

Figure 5.1a shows an example LR-mesh with prescribed continuity and maximum polynomial degree in each element $\max(p)_\sigma$ depicted in grayscale. The upper-right corner is refined using local hp -refinement. Figure 5.1b shows the Greville abscissae of the associated LR B-splines.

5.1.2 Constructive example of hp -refinement

Here we present an example of hp -refinement to construct a non-uniform degree spline space on an LR-mesh, following the algorithm presented in Section 5.1.1.

Let us begin with a tensor mesh \mathcal{M}_0 comprising quadratic B-splines, thus generating a set of B-splines $\mathcal{S}_p(\mathcal{M}_0)$ with $p = 2$. We mark a function for refinement, depicted in Figure 5.2, where all meshlines have a uniform multiplicity of 1. Following the hp -refinement algorithm, we first apply the p -refinement step. Here, we execute the degree-raising operation as detailed in Section 1.1.4, increasing the multiplicity of the meshlines within the support of the marked B-spline through sequential split insertion. Consequently, we obtain a set of non-uniform degree LR B-splines $\mathcal{S}_p(\mathcal{M}_1)$ with $p = \{2, 3\}$. The resulting LR-mesh with the degree marked elements along with the same mesh illustrating the Greville abscissae is shown in Figure 5.3. During the p -refinement process, the degree elevation step outlined in Section 5.1.1 enables degree elevation in each parametric direction independently. However, for the sake of simplicity, we execute all degree elevation steps in both parametric directions.

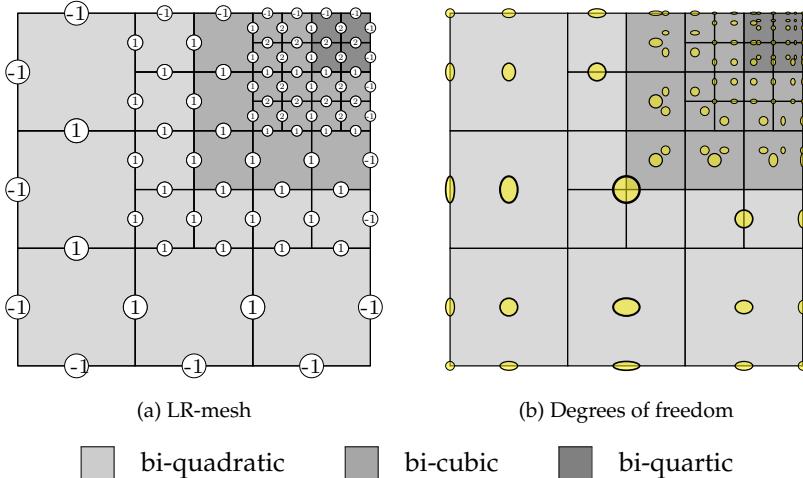


Figure 5.1: Non-uniform degree LR spline space spanned by a set of bi-quadratic, bi-cubic and bi-quartic B-splines. Figure (a) shows the non-uniform degree LR-mesh and the meshline continuity and Figure (b) depicts the Greville abscissae of the B-splines. Note that the LR spline space is C^1 across most mesh lines and C^2 over certain mesh-lines in the upper right corner, illustrating local hp -refinement. Along the boundary the meshlines are open with C^{-1} continuity.

For the second iteration, we apply the h -refinement step where, following the structured refinement approach (see Section 4.3.1), the resulting split insertion adheres to the following criteria:

- All knot spans of the marked B-spline are halved in both directions, with the coordinate of the split insertion positioned at the midpoint of each knot interval of the two local knot vectors.
 - The length of the split is determined to traverse the entire B-spline, ensuring that it follows the algorithm outlined in Section 5.1.1 to guarantee minimal support for all B-splines in the space.

The resulting h -refined mesh \mathcal{M}_2 produces a set of non-uniform degree B-splines $\mathcal{S}_p(\mathcal{M}_2)$ with the degree remaining consistent with the previous iteration, $p = \{2, 3\}$.

Subsequently, we proceed with another p -refinement step on a bi-cubic spline in \mathcal{M}_2 , as illustrated in the last row of Figure 5.3. Once again, following the degree elevation step in the refinement algorithm, this results in several bi-quartic B-splines in accordance with (1.15), leading to a new set of non-uniform degree LR B-splines denoted as $\mathcal{S}_p(\mathcal{M}_3)$ on \mathcal{M}_3 with $p = \{2, 4\}$.

5.1. Non-uniform degree B-splines on LR-meshes

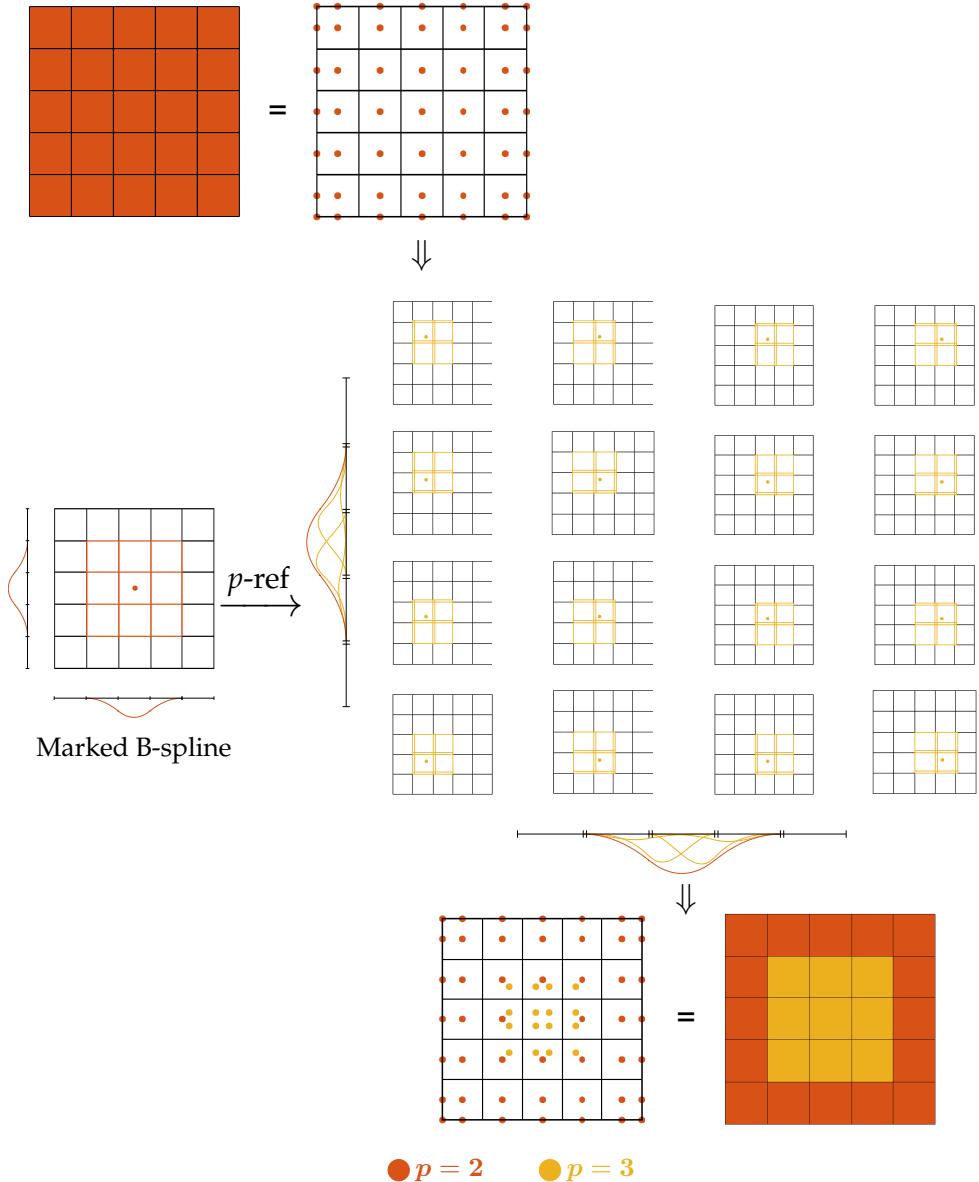


Figure 5.2: hp -refinement example (Iteration 1): Beginning with a tensor mesh \mathcal{M}_0 , which generates a set of B-splines $\mathcal{S}_p(\mathcal{M}_0)$ of uniform degree $p = 2$, as depicted in the first row. In the second row, a B-spline marked for refinement (orange) undergoes a p -refinement step, resulting in a few bi-cubic B-splines (yellow) following (1.15), leading to a new set of non-uniform degree LR B-spline set $\mathcal{S}_p(\mathcal{M}_1)$ on \mathcal{M}_1 with $p = \{2, 3\}$, as presented in the third row. All elements in the colored meshes are colored according to the maximal degree of the B-spline that has support over that element, alongside the same mesh illustrating the Greville abscissae for all B-splines.

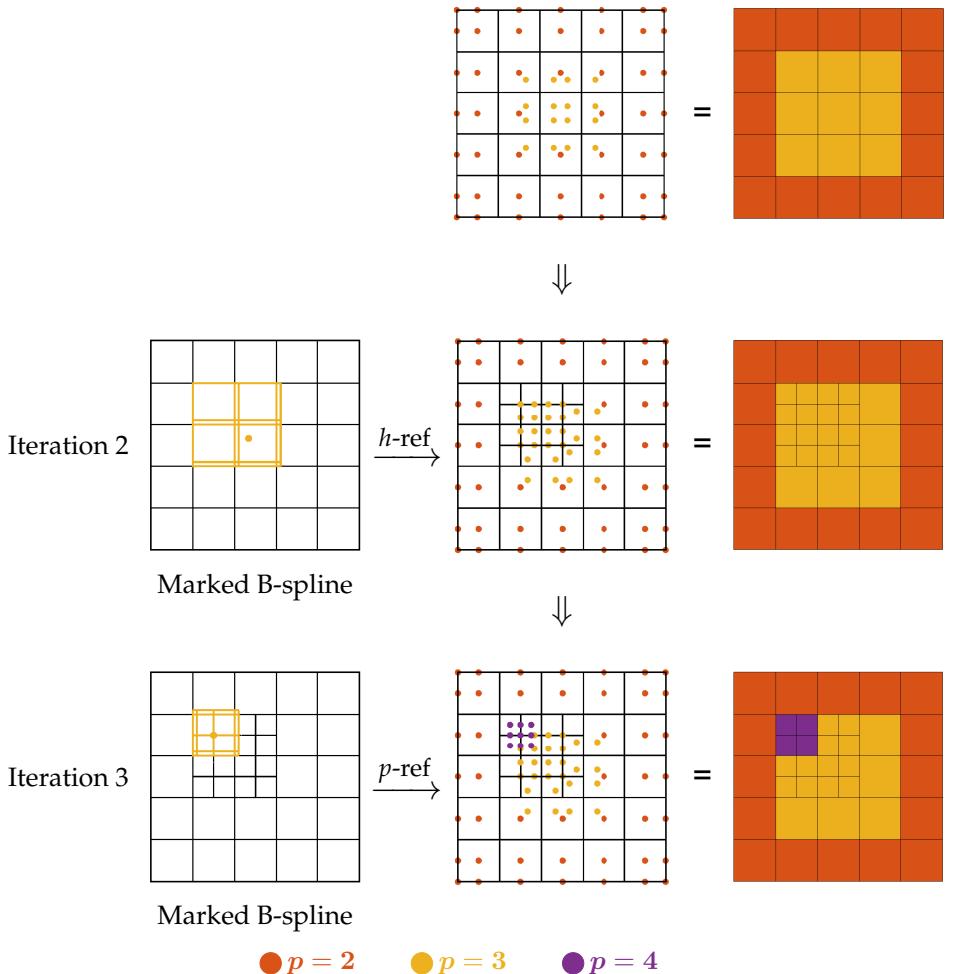


Figure 5.3: hp -refinement example (Iteration 2-3): Following the first iteration of p -refinement in Figure 5.2 we constructed \mathcal{M}_1 of $p = \{2, 3\}$. Now, in the second iteration we perform h -refinement over the marked B-spline as shown in the second row. Following the structured mesh refinement strategy, we introduce a net of splits in the support of the marked B-spline with multiplicity 1, as outlined in Algorithm 2, resulting in mesh \mathcal{M}_2 of $p = \{2, 3\}$, generating an hp -refined set of LR B-splines $\mathcal{S}_p(\mathcal{M}_2)$. Finally, we illustrated another iteration of local p -refinement on a bi-cubic spline, as presented in third row. After step of degree elevation following Algorithm 2, we achieve a set of hp -refined set of B-splines $\mathcal{S}_p(\mathcal{M}_3)$ with degree $p = \{2, 4\}$.

5.2. Data-structures

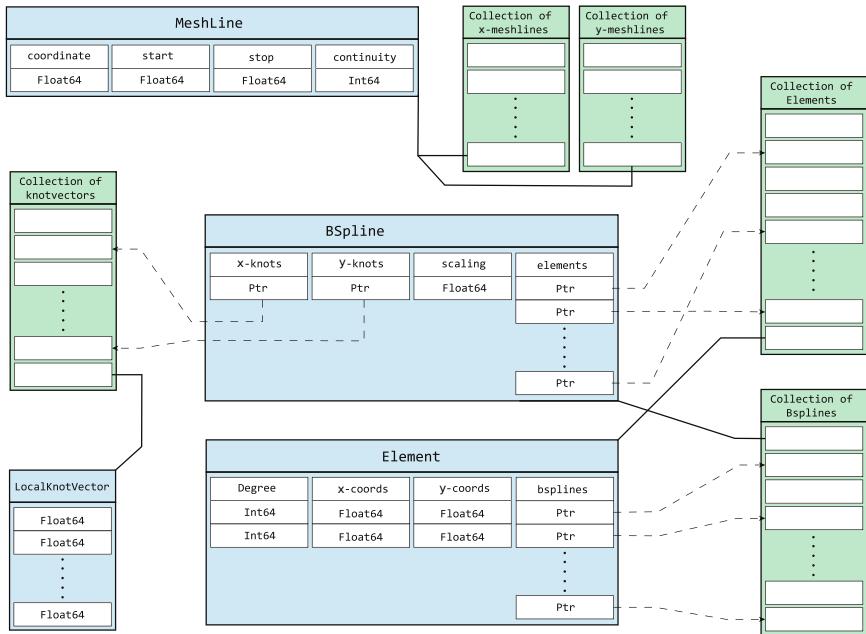


Figure 5.4: Main data-structures used in the implementation of non-uniform degree LR B-splines.

5.2 Data-structures

Implementation of spline technologies that enable local refinement, such as LR-splines or splines on any unstructured meshes, can be challenging. The additional flexibility of local degree elevation proposed with the non-uniform degree LR B-splines increases complexity further. Hence, it is important to make sound choices about the key data-structures in the implementation. Figure 5.4 summarizes these and their underlying relationships. We make a distinction between collections (green) and data-structures (blue):

- A **MeshLine** is defined by the coordinate that is kept constant and a start- and end-coordinate in the opposite direction. The continuity that is maintained across the mesh-line is also stored.
- A **LocalKnotVector** is a non-decreasing sequence of $p + 2$ floating point values.
- A **BSpline** function is defined by two local knot vectors and stores a set of pointers to the elements that are within its support.

- An `Element`, defined by its coordinates in the corresponding direction, stores the maximum polynomial degrees in both directions and maintains a set with all active BSplines.

The collections (in green) store all the unique objects of the LR-spline and are conveniently represented as hash-sets. Hash-sets maintain unique collections and provide $O(1)$ queries. The local collections `elements` in `BSpline` and `bsplines` in `Element` are similarly stored as hash-sets. The size of these sets should be flexible to allow the sets to change when local refinement is performed. The relationships between the different data-structures is conveniently implemented by making use of pointers to objects in the main collections.

Remark 5.1. *It could be convenient to introduce a quad-tree data-structure to track refinement (and coarsening) of elements.*

Remark 5.2. *It is not strictly necessary to store the maximum polynomial degrees per element, since they can be ascertained by inspecting the polynomial degrees of all active functions on the element.*

Remark 5.3. *It is not strictly necessary to store the meshlines since this information is contained in the local knot vectors pointed to by the B-splines.*

Figure 5.5 illustrates the mesh element structure of non-uniform degree LR B-splines. Here the Greville points of the B-splines that have support in the highlighted mesh element are shown. Refinement may lead to elements that are overloaded, which means that the dimension of the local space (on the element) is greater than the dimension of the polynomials. In essence, this means that local linear independence is lost. For most applications it is important that global linear independence is maintained.

Figure 5.6 depicts the support, knot-lines, and Greville abscissae of several tensor-product B-splines in the LR spline space. Some meshlines are repeated, which illustrates the accompanying reduction in continuity.

5.3 Numerical results of adaptive IgA with non-uniform degree LR B-splines

In this section, we present some case studies to assess the application of non-uniform degree LR B-splines on LR-meshes in solving PDEs. Employing the isogeometric approach within Galerkin methods as described in Section 1.2, we represent both the fields of interest and the geometry using the same set of B-splines. Specifically, employing the non-uniform degree LR B-splines constructed in Section 5.1.1 can serve as a viable strategy to construct an efficient approximation space for isogeometric analysis.

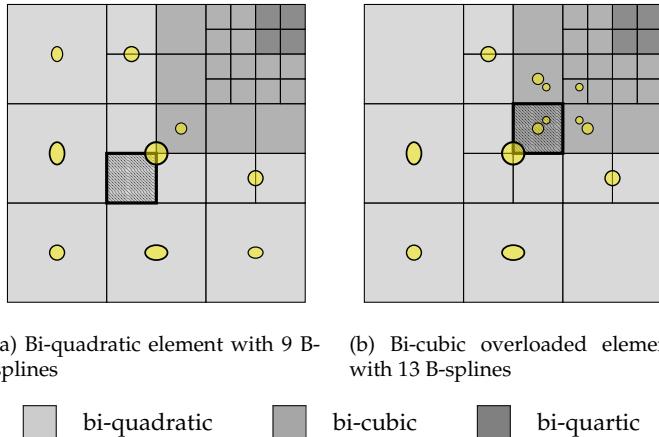


Figure 5.5: Element data-structure. Every mesh element points to a set of B-splines (grevillepoints are highlighted in yellow) whose support intersect the element under consideration. Figure (a) depicts a bi-quadratic element comprised of nine supporting B-splines. The element is not overloaded (the basis is locally linear independent) and spans the full space. Figure (b) shows a bi-cubic element comprising thirteen B-splines. Hence, this element is overloaded (the basis is not locally linear independent) and it spans the space of bi-quadratics but not the full space of bi-cubics.

We focus on the second-order problem in (1.26), and we select the non-uniform degree LR B-spline space as the approximation spaces in (1.21),

$$\mathbb{W} = \mathbb{S}_p(\mathcal{M}) = \left\langle N_{\Xi_k, p} \in \mathcal{S}_p(\mathcal{M}) \right\rangle.$$

This space is spanned by the set of B-splines $\mathcal{S}_p(\mathcal{M})$ constructed on an LR-mesh \mathcal{M} following the refinement algorithm in Section 5.1.1.

Following the same approach as in the application of LR TB-splines in IgA (see Section 4.4), in the context of non-uniform degree LR B-splines the homogeneous boundaries are satisfied pointwise exactly. In the non-homogeneous case, the boundary values are represented in the underlying spline space by suitable approximation strategies (e.g., least squares approximation) and subsequently the reduction to the homogeneous case is considered, so dealing again with a special instance of the problem (1.26).

For the numerical computation of the integrals for constructing the linear system in (1.22), we employ element-wise Gaussian quadrature rule. We adopt a straightforward approach for selecting quadrature points for discretizations involving non-uniform degree B-splines. In all presented

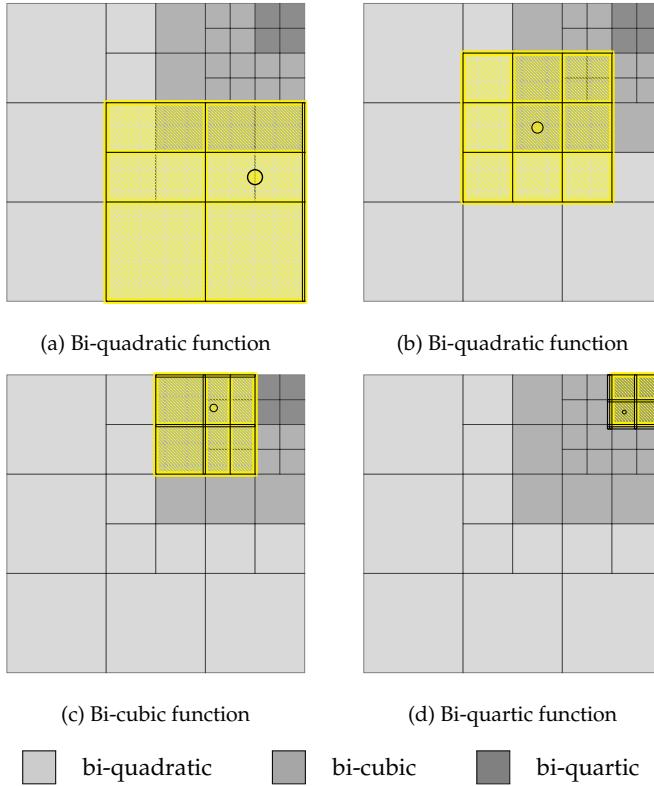


Figure 5.6: B-spline data-structure. The B-splines are defined by local knot vectors and store the elements in their support. Figures (a-d) show examples of B-splines of different polynomial degrees and varying smoothness highlighted by the drawn multiplicity.

case studies, we use $(\max(p)_\sigma + 1)$ quadrature points in each mesh element, where $\max(p)_\sigma$ represents the maximum degree of the B-spline from $S_p(\mathcal{M})$ with support over that element σ . For a fair comparison of computational cost for assembly of the linear systems, in all case studies we monitor both the total number of quadrature points (N_{QP}) and the degrees of freedom (dof).

For adaptive refinement we need to mark the B-splines contributing most to the approximation error. In this section, all case studies opt for an error-based automatic marking strategy as described in Section 4.3.1, where after computing the discrete solution $u_{\mathbb{W}}$ of the model problem (1.19), we introduce the error indicator η_σ in a suitable norm on a mesh element σ within an LR-mesh \mathcal{M} . We specify this indicator in each prob-

lem based on the availability of an exact solution to the problem. Moreover, we follow the strategy where we first mark the mesh elements that crosses the threshold criterion defined by the maximum element-wise error and a marking parameter ψ as in (4.12), then mark all the B-splines that have support over those elements. The marking strategy for functions to be refined is consistent for both h - and p -refinement, and thereby for hp -refinement, maintaining the same marking parameter ψ across all iterations.

The first case study in Section 5.3.1 considers heat conduction on the unit square, showcasing a globally smooth solution without large gradients, making global p -refinement typically the most effective choice. The second problem in Section 5.3.2 serves as a benchmark for adaptive refinement, addressing advection-dominated advection-diffusion on a unit square with sharp internal and boundary layers. In this context, local adaptive h - and p -refinement could be beneficial for resolving the behavior near the sharp layers. With the help of hp -refinement we aim to achieve fast convergence by finding the right balance between the local mesh refinement and degree elevation, while keeping the degrees of freedom of the solution space to minimal.

5.3.1 Case study 9: Smooth Poisson problem on a unit square

We consider the Poisson problem on the unit square $\Omega = [0, 1]^2$, where the parameters in (1.26) are $\kappa = 1$, $\mathbf{a} = 0$, $c = 0$ and the source term

$$f(x, y) = 8\pi^2 \sin(2\pi x) \cos(2\pi y),$$

with homogeneous Dirichlet boundary conditions, i.e. $g = 0$. The analytical solution satisfying the boundary value problem is

$$u(x, y) = \sin(2\pi x) \cos(2\pi y), \quad (5.1)$$

as illustrated in Figure 5.7.

Since we are solving for the Laplacian of a smooth field on a simple unit square domain with homogeneous boundary conditions, the solution is inherently smooth. Hence the potential benefits of local refinement are limited. However, using smoother higher-degree basis functions to construct the solution space can undoubtedly enhance accuracy.

Given the exact solution in (5.1), we can compute the error between the exact (u) and discrete solution ($u_{\mathbb{W}}$), giving the element-wise error estimator in energy norm

$$\eta_{\sigma} = \|\nabla u - \nabla u_{\mathbb{W}}\|_{E^2(\sigma)}. \quad (5.2)$$

We set the marking parameter to $\psi = 0.8$ for the marking criterion (4.12).

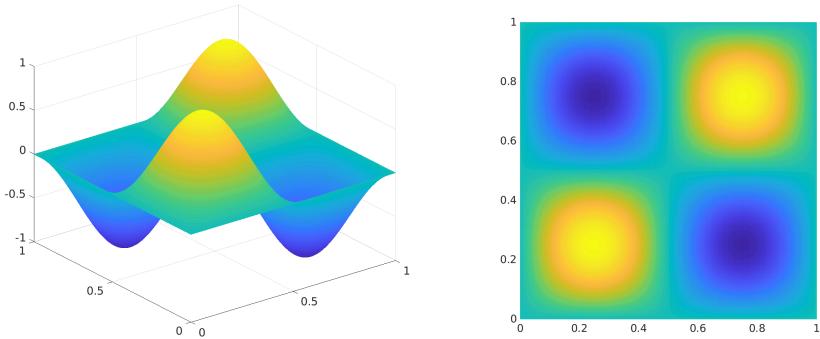


Figure 5.7: Case study 9. Exact solution of the Poisson problem as in (5.1).

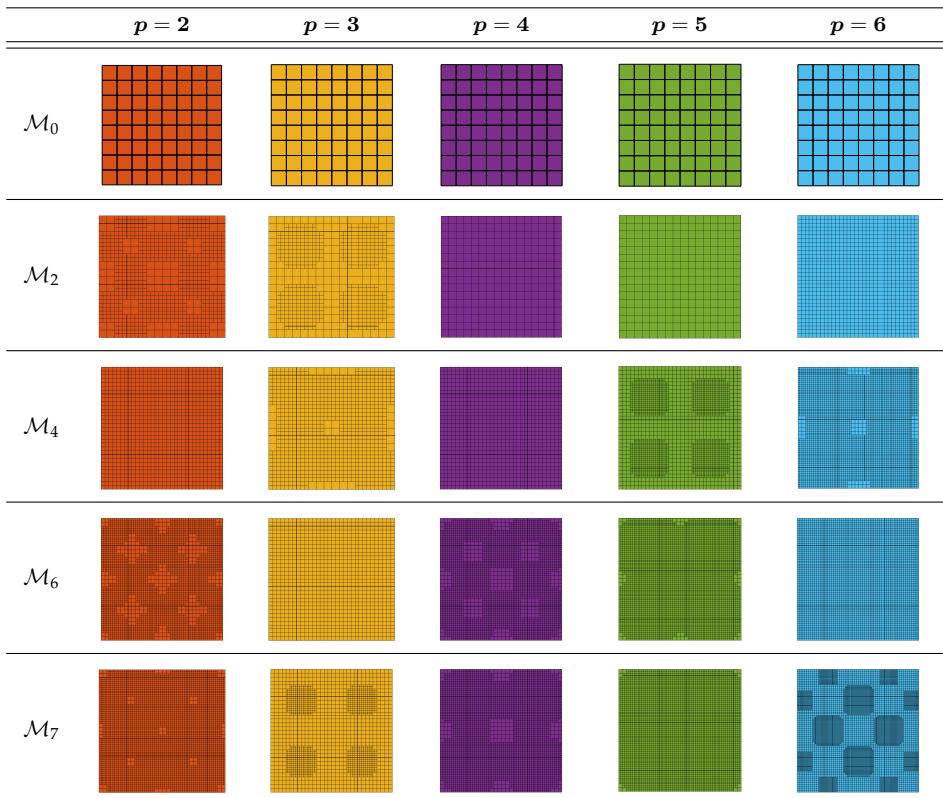
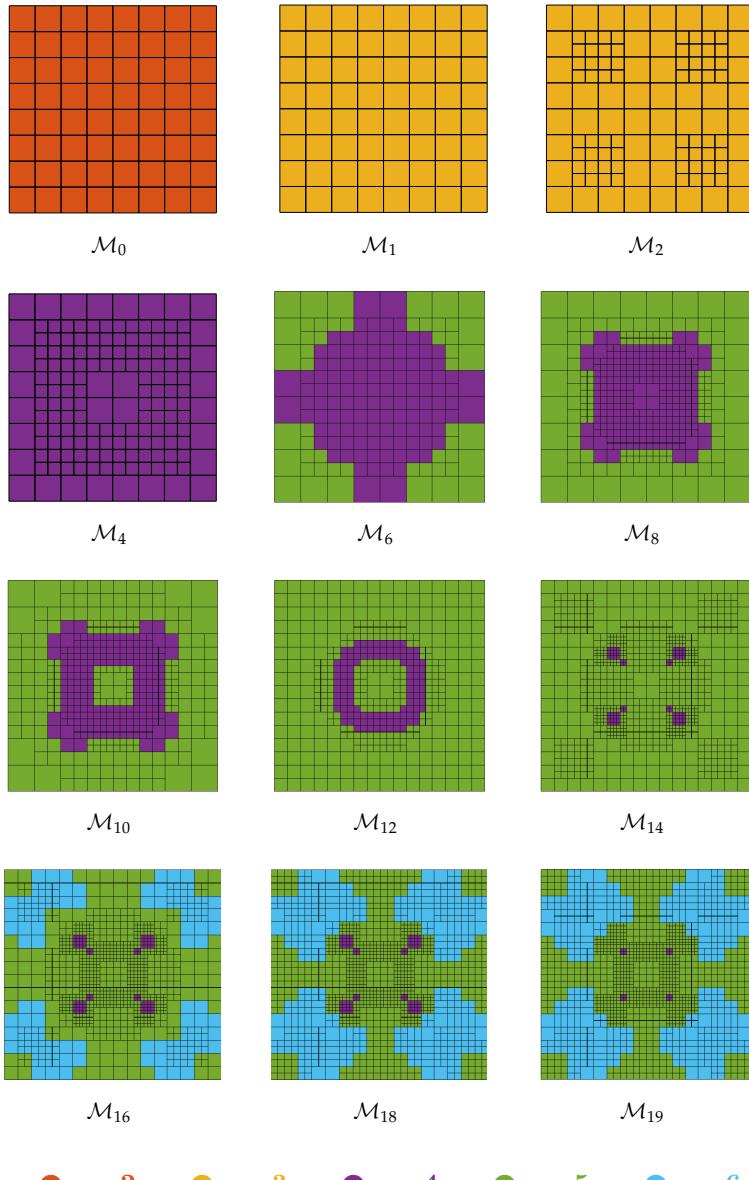


Figure 5.8: Case study 9. Uniform degree LR-meshes for the solution of the Poisson problem in (5.1), obtained by local h -refinement for $p = 2, 3, 4, 5, 6$, following the error indicator in (5.2) with marking parameter $\psi = 0.8$.



● $p = 2$ ● $p = 3$ ● $p = 4$ ● $p = 5$ ● $p = 6$

Figure 5.9: Case study 9. hp -refined LR-meshes for the Poisson problem with an exact solution in (5.1), refined using the error indicator in (5.2) with marking parameter $\psi = 0.8$, starting from \mathcal{M}_0 with $p = 2$ till the final mesh \mathcal{M}_{19} with $p = \{2, 6\}$.

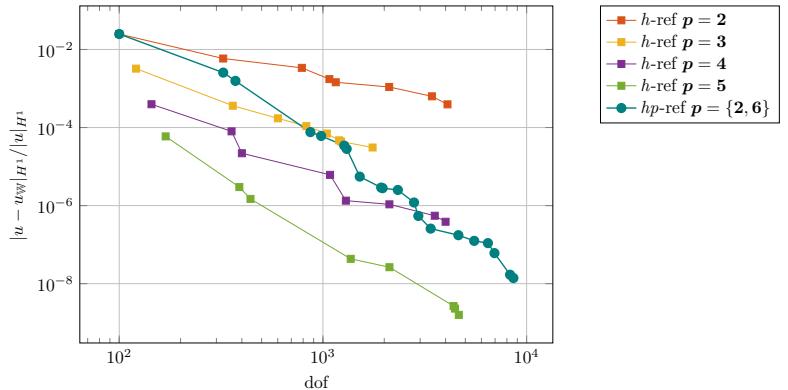
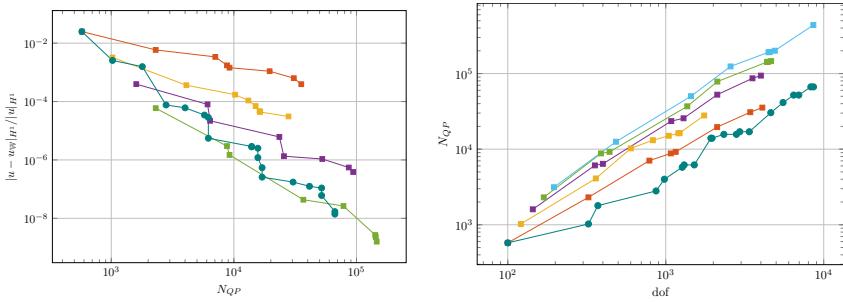
Following the refinement algorithm outlined in Section 5.1.1 with the error indicator in (5.2), we construct hp -refined meshes, by specifically performing p -refinement in the odd iterations and h -refinement in the even iterations. We compare their performance with uniform degree h -refined LR-meshes. The final hp -refined meshes are illustrated in Figure 5.9, starting from a tensor mesh \mathcal{M}_0 with $h_0 = 1/8$ for degree $p = 2$ to \mathcal{M}_{19} of $p = \{2, 6\}$. Similarly, we consider some h -refined meshes, beginning with a tensor mesh having $h = 1/8$ for degree $p = 2$ up to $p = 6$, as presented in Figure 5.8. However, due to the problem's smooth behavior, it does not significantly benefit from h -refinement, leading to LR-meshes that cover almost the entire domain in many iterations. In contrast, with hp -refinement, the p - and h -refined regions are more localized, as shown in Figure 5.9. This localization indicates a more efficient adaptation to the problem's characteristics.

The Figure 5.10 provides an overview of the performance of the non-uniform degree hp -refined meshes in Figure 5.9, compared to the uniform degree h -refined meshes in Figure 5.8. First, we compare the convergence of the error in the energy norm for different LR-meshes to examine the performance of the hp -refined mesh in Figure 5.10a. We observed that the hp -refined mesh exhibits faster convergence than h -refined meshes up to degree $p = 4$. However, for degrees $p = 5$ and 6 , the convergence appears to be slower, possibly due to lower smoothness in the hp -refined meshes. Secondly, we evaluate the computational efficiency in terms of the number of quadrature points (N_{QP}) required to construct the linear system in Figure 5.10b. It is evident that hp -refined meshes offer significant computational savings to achieve similar accuracy levels compared to higher degree h -refined meshes. The hp -refined mesh starts with the $p = 2$ in the first iteration and eventually converges to an equivalent of a higher degree solution with similar degrees of freedom. Finally, Figure 5.10c shows that the total number of quadrature points N_{QP} in the hp -refined meshes remains lower than the h -refined meshes.

In summary, utilizing hp -refined meshes, we can benefit with the computational efficiency in the lower degree iterations, while still converging to the higher degree solution in the subsequent iterations and exhibiting good convergence, since it gradually increases the polynomial degree.

5.3.2 Case study 10: Advection–diffusion problem with internal sharp layer on a square

In this case study, we tackle a classical benchmark problem involving advection–diffusion on the unit square $\Omega = [0, 1]^2$, where the advection flow is skewed to any parametric direction; see [43, 55]. The parameters


 (a) Relative error in H^1 -seminorm versus the number of degrees of freedom.

 (b) Relative error in H^1 -seminorm versus the number of quadrature points.

(c) Number of degrees of freedom versus the number of quadrature points.

Figure 5.10: Case study 9. Comparison of different solution parameters (relative error, degrees of freedom, and number of quadrature points) for the Poisson problem with an exact solution in (5.1), between adaptive local *h*-refinement and *hp*-refinement on a sequence of LR-meshes.

for the equation (1.26) are set as follows:

$$\kappa = 10^{-6}, \quad c = 0, \quad \mathbf{a} = (\cos \theta, \sin \theta), \quad \theta = \frac{\pi}{4}, \quad f(x, y) = 0,$$

and the discontinuous Dirichlet boundary conditions are as depicted in Figure 3.12. These conditions enforce a jump at the point $(0, 0.2)$, leading to a sharp internal layer along the advection flow at an angle θ , identified by $(\cos(\theta), \sin(\theta))$, and to boundary layers along the outflow boundaries.

In the regime of advection-dominated diffusion, characterized by a large Péclet number $\text{Pe}_g = a = 10^{-6}$, approximate solutions from piecewise polynomial spaces often exhibit spurious oscillations until the dis-

cretization is fine enough to resolve the sharp layers depicted by the exact solution. To address this issue, we employ streamline upwind Petrov-Galerkin (SUPG) stabilization; see [23, 56]. The stabilization constant used for the SUPG stabilization is as given in (3.14).

Due to the sensitivity of the problem, a proper treatment of the boundary conditions is imperative. To prevent oscillations in the approximated boundary conditions, we opt for quasi-interpolation and use the Schoenberg operator along each edge (see Example 1.5). This involves considering a linear combination of the given B-splines, where the coefficient of each basis function is obtained by evaluating the function to be approximated at the corresponding Greville abscissa.

For the adaptive refinement, we use the error based automatic refinement strategy described in Section 4.3.1, accompanied with the element-wise residual based error indicator computed for the discrete solution in the L^2 -norm,

$$\eta_\sigma = \| -\kappa \nabla^2 u_{\mathbb{W}} + \mathbf{a} \cdot \nabla u_{\mathbb{W}} - f \|_{L^2(\omega)} \quad (5.3)$$

We set $\psi = 0.99$ for the marking criterion in (4.12).

Following the refinement algorithm outlined in Section 5.1.1 with the residual based error indicator in (5.3), we construct hp -refined meshes, by alternating p - and h -refinement in the odd and even iterations, respectively. The resulting hp -refined mesh is illustrated in Figure 5.11a, starting from a tensor mesh \mathcal{M}_0 with $h_0 = 1/8$ and degree $p = 2$ to \mathcal{M}_8 of $p = \{2, 6\}$. For a fair comparison of the quality of the approximate solutions, we consider h -refined meshes of degree $p = 2$ and $p = 6$, up to the number of iterations such that the degrees of freedom in their corresponding sets of LR B-splines are in a similar range as in the hp -refined mesh. The resulting h -refined meshes are presented in Figure 5.8.

Figure 5.12 illustrates the approximate solution of this problem, and it is evident that the solution of advection-diffusion problem on hp -refined mesh in Figure 5.11a do not exhibit spurious oscillations and the internal and boundary layers are localized well. Table 5.1 summarizes the values of maximal over- and undershoot in the neighborhood of the layers evaluated on a uniform grid of 3001×3001 points along each direction for the different setups of the meshes, along with the number of degrees of freedom (dof) and number of quadrature points N_{QP} .

We compare the convergence of the approximate solutions by observing the minimum and maximum of the oscillations for different LR-meshes to examine the performance of the hp -refined mesh in Figure 5.13a. The problem is very slowly converging for uniform degree h -refined meshes, while for the hp -refined mesh, the solution in the first iteration is the same as the $p = 2$ solution, which improves drastically as we reach higher iterations. This results in a similar quality result as of the h -refined mesh

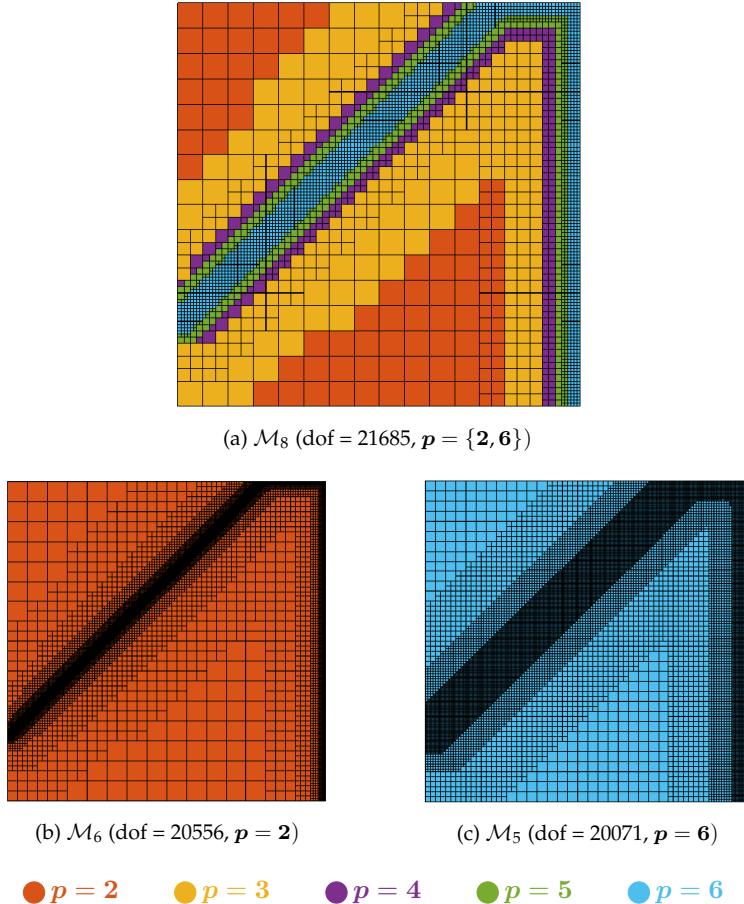


Figure 5.11: Case study 10. LR-meshes for the solution of the advection-diffusion problem. Figure 5.11a shows an hp -refined mesh \mathcal{M}_8 with $p = \{2, 6\}$), while Figure 5.11b and Figure 5.11c illustrate the h -refined LR-meshes of degrees $p = 2$ and $p = 6$, respectively. All LR-meshes are obtained by using the residual based error indicator in (5.3) with marking parameter $\psi = 0.99$.

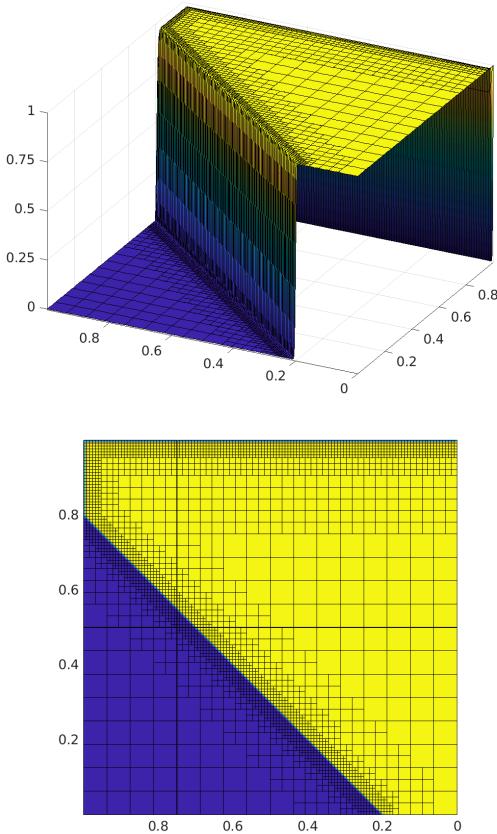


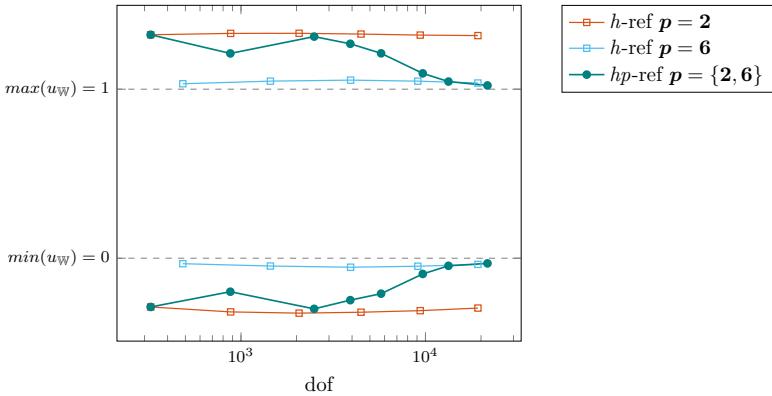
Figure 5.12: Case study 10. Plot of the SUPG stabilized approximate solution u_W of the advection-diffusion problem, on the hp -refined LR-mesh \mathcal{M}_8 of $p = \{2, 6\}$ with dof = 21685. Figure 5.11a illustrates the hp -refined LR-mesh.

of degree $p = 6$ for quite similar degrees of freedom. However, the benefits are most evident when we compare the number of quadrature points (N_{QP}) required to construct the linear system in Figure 5.13b. The number of quadrature points in the last iteration of the hp -refined mesh \mathcal{M}_8 is comparable to the last iteration \mathcal{M}_6 of degree $p = 2$ and the third iteration \mathcal{M}_3 of degree $p = 6$ in h -refined meshes. Moreover, following the same trend as in the last case study, the total number of quadrature points N_{QP} in the hp -refined meshes remains lower than the h -refined meshes, as shown in Figure 5.13c. Therefore, we gain drastically in terms of computational efficiency by using non-uniform degree LR B-splines, while still achieving the quality of solutions similar to higher degree solutions.

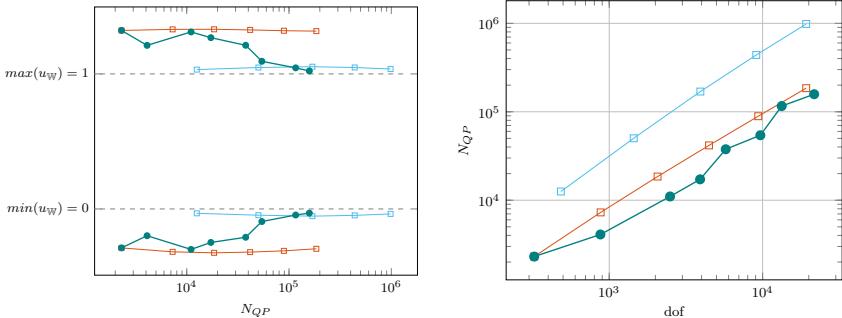
5.3. Numerical results of adaptive IgA with non-uniform degree LR B-splines

Refinement	p	dof	N_{QP}	max	min
hp -ref	{2, 6}	21685	157976	1.0044	-6.1310×10^{-3}
h -ref	2	19246	185004	1.0634	-5.9012×10^{-2}
h -ref	6	19243	983479	1.0073	-7.3458×10^{-3}

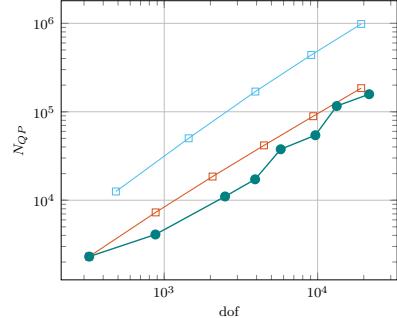
Table 5.1: Case study 10. Comparison of maximum and minimum of the SUPG stabilized discrete solutions u_W of the advection-diffusion problem, for h - and hp -refined LR-meshes presented in Figure 5.11.



(a) Extrema of the discrete solution versus the number of degrees of freedom.



(b) Extrema of the discrete solution versus the number of quadrature points.



(c) Number of degrees of freedom versus the number of quadrature points.

Figure 5.13: Case study 10. Convergence of the maximum and minimum of the oscillations in the discrete solution u_W of advection-diffusion problem with adaptive local h -refinement and hp -refinement and the comparison of the rate of increase in degrees of freedom against the number of quadrature points N_{QP} .

5.4 Conclusive remarks

Traditional tensor-product B-splines lack the ability of local refinement which is needed in order to achieve faster convergence in applications. In particular, higher order isogeometric methods based on tensor-product B-splines are not able to exploit the full potential offered by isogeometric analysis when applied to problems involving local features. Moreover, the uniform degree B-splines over LR-meshes do offer a lot of flexibility with the local refinement.

In this chapter, we introduced LR B-splines that allow for non-uniform polynomial degree. This capability, combined with local h -refinement, allows for a rich space of possible refinements. For example, it enables local hp -refinement, that is, increasing the polynomial degree at newly introduced meshlines. The new refinement schemes are based on well known two-scale relations under degree elevation of individual basis functions, which automatically implies that the refinements are nested.

We first introduced an h -refinement and a p -refinement algorithm for LR splines. Depending on the order of operations, combinations of these allow for hp -refinement. We then introduced a set of data-structures to manage the increased complexity due to the presence of non-uniform degrees.

The potential of local h -, p -, and hp -adaptive refinement was investigated in a variety of conditions. We investigated its efficacy for a Poisson problem with a smooth solution. We also investigated adaptivity in the context of an advection dominated advection-diffusion problem that features an internal layer skew to the mesh. As expected, the local combinations of h - and p -refinement restore faster rates of convergence compared to the uniform degree h -refined LR meshes. Furthermore, we showed that adaptive local hp -refinement can produce highly sharp layers without overshoots.

Outlook

This dissertation has explored a diverse set of extensions of the theory and application of isogeometric analysis.

- The idea of Tchebycheffian splines as a generalization of classical algebraic polynomial splines was investigated and a problem dependent strategy for solving differential problems through isogeometric analysis was devised to design a better suited solution space based on the problem at hand.
- The idea of splines on unstructured meshes, and in particular on LR-meshes, has been investigated in two directions: non-uniform degrees and Tchebycheffian spline spaces.

As for the generalization of splines beyond the polynomial setting, we have extensively considered the subclass of Tchebycheffian splines whose pieces are drawn from a single ECT-space which is the null space of a differential operator with real constant coefficients, and includes constants. This subclass of Tchebycheffian splines already provides a large variety of combinations of polynomial, exponential, and trigonometric functions equipped with a wide spectrum of shape parameters and can be represented in terms of a B-spline like basis (TB-splines) at least for sufficiently fine partitions. It turns out that, such TB-splines can outperform polynomial B-splines whenever appropriate problem-driven selection strategies for the underlying ECT-spaces are applied. In particular, they can be beneficial both from the geometrical and the analytical point of view offering a perfect fit to the isogeometric approach.

Several extensions of the developments presented (or alternative approaches) are feasible and will form the focus of future research. Some of them are outlined below.

Chapters 2 and 3 presented Tchebycheffian spline spaces derived from a large class of ECT-spaces and discussed their application in isogeometric analysis through their B-spline like basis, the TB-splines. Although, the large variety of ECT-spaces that are null-spaces of constant-coefficient linear differential operators already offers an extreme flexible environment for applications, an additional extension is provided by the so-called multi-degree Tchebycheffian B-splines (MDTB-splines); see [54, 90, 115]. MDTB-splines allow for ECT-spaces of different dimensions on different

intervals of the given partitions; they are also supported by the Matlab toolbox in [115]. However, finding (automatic) strategies to select both the dimension and the structure for different ECT-spaces seems to be a very challenging task.

In Chapter 4 we presented the seamless integration of TB-splines over LR-meshes as an extension to the standard LR B-splines. However, the complete structural similarity with the polynomial case also implies that LR TB-splines suffer from the same weaknesses as their polynomial counterpart. For example, LR TB-splines might be (locally) linearly dependent; see also Remark 4.14. Common refinement strategies proposed in the literature, such as the minimum span, full span, and structured refinement [61, 112], do not ensure linear independence of the LR B-splines. The so-called peeling algorithm can be adopted to remove redundant basis functions [41, 92] but refinement strategies ensuring linear independence are strongly preferred, both from the theoretical and algorithmic point of view. A complete characterization of linear independence for LR B-splines is still not known, but there exist features of the underlying LR mesh, like the so-called non-nested support (N_2S) property [19], that guarantee the stronger property of local linear independence. Refinement strategies ensuring the N_2S property for LR B-splines have been proposed in [20, 91, 93]. It is likely the case that the structural similarity paves again the path for an extension of the above properties from the polynomial to the Tchebycheffian setting. This can be an interesting topic for future investigation. In particular, the identification of refinement strategies ensuring (local) linear independence for LR (T)B-splines, not relying on peculiar properties of algebraic polynomials but only on structural properties of the involved spaces, may provide further insights and understandings of the LR paradigm, also in the polynomial case. Furthermore, a mixed LR-spline strategy combining TB-splines with polynomial B-splines could also assist with the better selection of the TB-spline basis, by potentially starting with an LR B-spline mesh and augmenting the TB-spline basis as needed based on solution behavior.

In Chapter 5 for the application of non-uniform degree LR B-splines in adaptive isogeometric analysis, the tagging of functions for refinement was based on a relatively unsophisticated procedure, and, in the context of hp -refinement, the h - and p -refinement steps were simply alternated. A holistic approach to a-posteriori error-estimation and adaptive local refinement would require an effective approach to choosing the type of refinement and effective criteria to choose functions to tag for refinement. Function coarsening is another issue that should be investigated in this context. Another important issue is global linear independence of non-uniform degree LR B-splines. It is currently not known under which conditions and for which type of refinements non-uniform degree LR B-

splines maintain global linear independence. It would be also interesting to apply non-uniform degree splines in different contexts, for example in computer aided design and data reconstruction.

In a more general context, many applications require the use of unstructured meshes to achieve greater flexibility. The study of unstructured meshes with extraordinary vertices provides a broader scope for such applications. Substantial work has been done for standard polynomial B-splines on unstructured meshes [127]. Additionally, the same issue has been discussed in the context of local mesh refinement, particularly with T-splines [72, 107, 124] and hierarchical splines [126]. Due to the structural similarities of TB-splines with polynomial B-splines, it can be interesting to extend this line of research to both tensor-product TB-splines and LR TB-splines.

Finally, the truncation mechanism has proven to be very effective in achieving highly localized refinement by reducing support overlapping in the context of hierarchical [49] and T-meshes [125]. The same concept can be investigated in the context of LR-meshes.

Bibliography

- [1] A. Aimi, M. Diligenti, M. L. Sampoli, and A. Sestini, *Non-polynomial spline alternatives in isogeometric symmetric Galerkin BEM*, Applied Numerical Mathematics **116** (2017), 10–23.
- [2] C. Bangert and H. Prautzsch, *Circle and sphere as rational splines*, Neural, Parallel & Scientific Computations **5** (1997), 153–162.
- [3] Y. Bazilevs, L. Beirão da Veiga, J. A. Cottrell, T. J. R. Hughes, and G. Sangalli, *Isogeometric analysis: approximation, stability and error estimates for h-refined meshes*, Mathematical Models and Methods in Applied Sciences **16** (2006), 1031–1090.
- [4] Y. Bazilevs, V. M. Calo, J. A. Cottrell, J. A. Evans, T. J. R. Hughes, S. Lipton, M. A. Scott, and T. W. Sederberg, *Isogeometric analysis using T-splines*, Computer Methods in Applied Mechanics and Engineering **199** (2010), 229–263.
- [5] C. V. Beccari, G. Casciola, and M. L. Mazure, *Design or not design? A numerical characterisation for piecewise Chebyshevian splines*, Numerical Algorithms **81** (2019), 1–31.
- [6] _____, *Critical length: An alternative approach*, Journal of Computational and Applied Mathematics **370** (2020), 112603.
- [7] C. V. Beccari, G. Casciola, and S. Morigi, *On multi-degree splines*, Computer Aided Geometric Design **58** (2017), 8 – 23.
- [8] C. V. Beccari, G. Casciola, and L. Romani, *A practical method for computing with piecewise Chebyshevian splines*, Journal of Computational and Applied Mathematics **406** (2022), 114051.
- [9] L. Beirão da Veiga, A. Buffa, D. Cho, and G. Sangalli, *Analysis-suitable T-splines are dual-compatible*, Computer Methods in Applied Mechanics and Engineering **249** (2012), 42–51.
- [10] L. Beirão da Veiga, A. Buffa, G. Sangalli, and R. Vázquez, *Analysis-suitable T-splines of arbitrary degree: definition, linear independence and approximation properties*, Mathematical Models and Methods in Applied Sciences **23** (2013), 1979–2003.

- [11] D. Bister and H. Prautzsch, *A new approach to Tchebycheffian B-splines*, Curves and Surfaces with Applications in CAGD (A. Le Méhauté, C. Rabut, and L. L. Schumaker, eds.), Vanderbilt University Press, Nashville, 1997, pp. 387–394.
- [12] C. de Boor, *A practical guide to splines*, vol. 27, Springer-Verlag New York, 1978.
- [13] M. Boushaba, S. Eddargani, M. J. Ibáñez, and A. Lamnii, *Normalized B-spline-like representation for low-degree Hermite osculatory interpolation problems*, Mathematics and Computers in Simulation **225** (2024), 98–110.
- [14] C. Bracco, D. Berdinsky, D. Cho, M. Oh, and T. Kim, *Trigonometric generalized T-splines*, Computer Methods in Applied Mechanics and Engineering **268** (2014), 540–556.
- [15] C. Bracco and D. Cho, *Generalized T-splines and VMCR T-meshes*, Computer Methods in Applied Mechanics and Engineering **280** (2014), 176–196.
- [16] C. Bracco, T. Lyche, C. Manni, F. Roman, and H. Speleers, *Generalized spline spaces over T-meshes: Dimension formula and locally refined generalized B-splines*, Applied Mathematics and Computation **272** (2016), 187–198.
- [17] _____, *On the dimension of Tchebycheffian spline spaces over planar T-meshes*, Computer Aided Geometric Design **45** (2016), 151–173.
- [18] C. Bracco, T. Lyche, C. Manni, and H. Speleers, *Tchebycheffian spline spaces over planar T-meshes: Dimension bounds and dimension instabilities*, Journal of Computational and Applied Mathematics **349** (2019), 265–278.
- [19] A. Bressan, *Some properties of LR-splines*, Computer Aided Geometric Design **30** (2013), 778–794.
- [20] A. Bressan and B. Jüttler, *A hierarchical construction of LR meshes in 2D*, Computer Aided Geometric Design **37** (2015), 9–24.
- [21] A. Bressan and E. Sande, *Approximation in FEM, DG and IGA: a theoretical comparison*, Numerische Mathematik **143** (2019), 923–942.
- [22] M. Brilleaud and M. L. Mazure, *Design with L-splines*, Numerical Algorithms **65** (2014), 91–124.

- [23] A. N. Brooks and T. J. R. Hughes, *Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations*, Computer Methods in Applied Mechanics and Engineering **32** (1982), 199–259.
- [24] B. Buchwald and G. Mühlbach, *Construction of B-splines for generalized spline spaces generated from local ECT-systems*, Journal of Computational and Applied Mathematics **159** (2003), 249–267.
- [25] A. Buffa, D. Cho, and M. Kumar, *Characterization of T-splines with reduced continuity order on T-meshes*, Computer Methods in Applied Mechanics and Engineering **201** (2012), 112–126.
- [26] A. Buffa, D. Cho, and G. Sangalli, *Linear independence of the T-spline blending functions associated with some particular T-meshes*, Computer Methods in Applied Mechanics and Engineering **199** (2010), 1437 – 1445.
- [27] A. Buffa and C. Giannelli, *Adaptive isogeometric methods with hierarchical splines: Error estimator and convergence*, Mathematical Models and Methods in Applied Sciences **26** (2016), 1–25.
- [28] _____, *Adaptive isogeometric methods with hierarchical splines: Optimality and convergence rates*, Mathematical Models and Methods in Applied Sciences **27** (2017), 2781–2802.
- [29] A. Buffa, G. Sangalli, and R. Vázquez, *Isogeometric analysis in electromagnetics: B-splines approximation*, Computer Methods in Applied Mechanics and Engineering **199** (2010), 1143–1152.
- [30] F. Calabrò, G. Loli, G. Sangalli, and M. Tani, *Quadrature rules in the isogeometric Galerkin method: State of the art and an introduction to weighted quadrature*, Advanced Methods for Geometric Modeling and Numerical Simulation (C. Giannelli and H. Speleers, eds.), Springer INdAM Series, vol. 35, Springer International Publishing, 2019, pp. 43–55.
- [31] J. Cao, Z. Chen, X. Wei, and Y. J. Zhang, *A finite element framework based on bivariate simplex splines on triangle configurations*, Computer Methods in Applied Mechanics and Engineering **357** (2019), 112598.
- [32] M. L. Cardinali, C. Garoni, C. Manni, and H. Speleers, *Isogeometric discretizations with generalized b-splines: Symbol-based spectral analysis*, Applied Numerical Mathematics **166** (2021), 288–312.
- [33] J. M. Carnicer, E. Mainar, and J. M. Peña, *Critical length for design purposes and extended Chebyshev spaces*, Constructive Approximation **20** (2003), 55–71.

-
- [34] ———, *Critical lengths of cycloidal spaces are zeros of Bessel functions*, *Calcolo* **54** (2017), 1521–1531.
 - [35] H. Casquero, L. Liu, Y. Zhang, A. Reali, and H. Gomez, *Isogeometric collocation using analysis-suitable T-splines of arbitrary degree*, *Computer Methods in Applied Mechanics and Engineering* **301** (2016), 164 – 186.
 - [36] E. Cohen, T. Lyche, and R. Riesenfeld, *Discrete B-splines and subdivision techniques in computer-aided geometric design and computer graphics*, *Computer Graphics and Image Processing* **14** (1980), 87–111.
 - [37] E. Cohen, T. Lyche, and L. L. Schumaker, *Algorithms for degree-raising of splines*, *ACM Transactions on Graphics* **4** (1985), 171–181.
 - [38] ———, *Degree raising for splines*, *Journal of Approximation Theory* **46** (1986), 170–181.
 - [39] J. A. Cottrell, T. J. R. Hughes, and Y. Bazilevs, *Isogeometric analysis: Toward integration of CAD and FEA*, John Wiley & Sons, 2009.
 - [40] J. A. Cottrell, T. J. R. Hughes, and A. Reali, *Studies of refinement and continuity in isogeometric structural analysis*, *Computer Methods in Applied Mechanics and Engineering* **196** (2007), 4160 – 4183.
 - [41] T. Dokken, T. Lyche, and K. F. Pettersen, *Polynomial splines over locally refined box-partitions*, *Computer Aided Geometric Design* **30** (2013), 331–356.
 - [42] N. Dyn and A. Ron, *Recurrence relation for Tchebycheffian B-splines*, *Journal d’Analyse Mathématique* **51** (1988), 118–138.
 - [43] M. R. Dörfel, B. Jüttler, and B. Simeon, *Adaptive isogeometric analysis by local h-refinement with T-splines*, *Computer Methods in Applied Mechanics and Engineering* **199** (2010), 264 – 275.
 - [44] N. Engleitner and B. Jüttler, *Patchwork B-spline refinement*, *Computer-Aided Design* **90** (2017), 168–179.
 - [45] E. J. Evans, M. A. Scott, X. Li, and D. C. Thomas, *Hierarchical T-splines: Analysis-suitability, Bézier extraction, and application as an adaptive basis for isogeometric analysis*, *Computer Methods in Applied Mechanics and Engineering* **284** (2015), 1 – 20.
 - [46] D. R. Forsey and R. H. Bartels, *Hierarchical B-spline refinement*, *ACM SIGGRAPH Computer Graphics* **22** (1988), 205–212.

- [47] C. Giannelli and B. Jüttler, *Bases and dimensions of bivariate hierarchical tensor-product splines*, Journal of Computational and Applied Mathematics **239** (2013), 162 – 178.
- [48] C. Giannelli, B. Jüttler, S. K. Kleiss, A. Mantzaflaris, B. Simeon, and J. Speh, *THB-splines: An effective mathematical technology for adaptive refinement in geometric design and isogeometric analysis*, Computer Methods in Applied Mechanics and Engineering **299** (2016), 337 – 365.
- [49] C. Giannelli, B. Jüttler, and H. Speleers, *THB-splines: The truncated basis for hierarchical splines*, Computer Aided Geometric Design **29** (2012), 485–498.
- [50] ———, *Strongly stable bases for adaptively refined multilevel spline spaces*, Advances in Computational Mathematics **40** (2014), 459–490.
- [51] T. Goodman and M. L. Mazure, *Blossoming beyond extended Chebyshev spaces*, Journal of Approximation Theory **109** (2001), 48–81.
- [52] P. Hennig, S. Muller, and M. Kastner, *Bézier extraction and adaptive refinement of truncated hierarchical NURBS*, Computer Methods in Applied Mechanics and Engineering **305** (2016), 316 – 339.
- [53] R. R. Hiemstra, F. Calabrò, D. Schillinger, and T. J. R. Hughes, *Optimal and reduced quadrature rules for tensor product and hierarchically refined splines in isogeometric analysis*, Computer Methods in Applied Mechanics and Engineering **316** (2017), 966 – 1004.
- [54] R. R. Hiemstra, T. J. R. Hughes, C. Manni, H. Speleers, and D. Toshniwal, *A Tchebycheffian extension of multidegree B-splines: Algorithmic computation and properties*, SIAM Journal on Numerical Analysis **58** (2020), 1138–1163.
- [55] T. J. R. Hughes, J. A. Cottrel, and Y. Basilev, *Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement*, Computer Methods in Applied Mechanics and Engineering **194** (2005), 4135–4195.
- [56] T. J. R. Hughes, L. P. Franca, and G. M. Hulbert, *A new finite element formulation for computational fluid dynamics: VIII. the Galerkin/least-squares method for advective-diffusive equations*, Computer Methods in Applied Mechanics and Engineering **73** (1989), 173–189.
- [57] T. J. R. Hughes, A. Reali, and G. Sangalli, *Duality and unified analysis of discrete approximations in structural dynamics and wave propagation:*

- comparison of p-method finite elements with k-method NURBS*, Computer Methods in Applied Mechanics and Engineering **197** (2008), 4104–4124.
- [58] ———, *Efficient quadrature for NURBS-based isogeometric analysis*, Computer Methods in Applied Mechanics and Engineering **199** (2010), 301–313.
- [59] Z. Jingjing and X. Li, *Local refinement for analysis-suitable++ T-splines*, Computer Methods in Applied Mechanics and Engineering **342** (2018), 32 – 45.
- [60] K. A. Johannessen, M. Kumar, and T. Kvamsdal, *Divergence-conforming discretization for Stokes problem on locally refined meshes using LR B-splines*, Computer Methods in Applied Mechanics and Engineering **293** (2015), 38–70.
- [61] K. A. Johannessen, T. Kvamsdal, and T. Dokken, *Isogeometric analysis using LR B-splines*, Computer Methods in Applied Mechanics and Engineering **269** (2014), 471–514.
- [62] K. A. Johannessen, F. Remonato, and T. Kvamsdal, *On the similarities and differences between classical hierarchical, truncated hierarchical and LR B-splines*, Computer Methods in Applied Mechanics and Engineering **291** (2015), 64–101.
- [63] S. Karlin, *Total positivity*, Stanford University Press, Stanford, 1968.
- [64] S. Karlin and W. J. Studden, *Tchebycheff systems: With applications in analysis and statistics*, Interscience Publishers, New York, 1966.
- [65] G. Kermarrec, V. Skytt, and T. Dokken, *Optimal surface fitting of point clouds using local refinement: Application to GIS data*, Springer Nature, 2023.
- [66] R. Kraft, *Adaptive and linearly independent multilevel B-splines*, SFB 404, Geschäftsstelle, 1997.
- [67] G. Kuru, C. V. Verhoosel, K. G. van der Zee, and E. H. van Brummelen, *Goal-adaptive isogeometric analysis with hierarchical splines*, Computer Methods in Applied Mechanics and Engineering **270** (2014), 270 – 292.
- [68] X. Li and M. A. Scott, *Analysis-suitable T-splines: Characterization, refineability, and approximation*, Mathematical Models and Methods in Applied Sciences **24** (2014), 1141–1164.

- [69] X. Li and J. Zhang, *AS++ T-splines: Linear independence and approximation*, Computer Methods in Applied Mechanics and Engineering **333** (2018), 462 – 474.
- [70] X. Li, J. Zheng, T. W. Sederberg, T. J. R. Hughes, and M. A. Scott, *On linear independence of T-spline blending functions*, Computer Aided Geometric Design **29** (2012), 63 – 76.
- [71] L. Liu, H. Casquero, H. Gomez, and Y. J. Zhang, *Hybrid-degree weighted T-splines and their application in isogeometric analysis*, Computers & Fluids **141** (2016), 42–53.
- [72] L. Liu, Y. J. Zhang, and X. Wei, *Handling extraordinary nodes with weighted T-spline basis functions*, Procedia Engineering **124** (2015), 161–173.
- [73] G. Lorenzo, M. A. Scott, K. Tew, T. J. R. Hughes, and H. Gomez, *Hierarchically refined and coarsened splines for moving interface problems, with particular application to phase-field models of prostate tumor growth*, Computer Methods in Applied Mechanics and Engineering **319** (2017), 515 – 548.
- [74] T. Lyche, *A recurrence relation for Chebyshevian B-splines*, Constructive Approximation **1** (1985), 155–173.
- [75] T. Lyche, C. Manni, and H. Speleers, *Foundations of spline theory: B-splines, spline approximation, and hierarchical refinement*, Splines and PDEs: From Approximation Theory to Numerical Linear Algebra (T. Lyche, C. Manni, and H. Speleers, eds.), Lecture Notes in Mathematics, vol. 2219, Springer International Publishing, 2018, pp. 1–76.
- [76] ———, *Tchebycheffian B-splines revisited: An introductory exposition*, Advanced Methods for Geometric Modeling and Numerical Simulation (C. Giannelli and H. Speleers, eds.), Springer INdAM Series, vol. 35, Springer International Publishing, 2019, pp. 179–216.
- [77] T. Lyche and L. L. Schumaker, *Total positivity properties of LB-splines*, Total Positivity and Its Applications (M. Gasca and C. A. Micchelli, eds.), Mathematics and Its Applications, vol. 359, Kluwer, 1996, pp. 35–46.
- [78] C. Manni, F. Pelosi, and M. L. Sampoli, *Generalized B-splines as a tool in isogeometric analysis*, Computer Methods in Applied Mechanics and Engineering **200** (2011), 867–881.
- [79] ———, *Isogeometric analysis in advection–diffusion problems: Tension splines approximation*, Journal of Computational and Applied Mathematics **236** (2011), 511–528.

-
- [80] C. Manni, F. Pelosi, and H. Speleers, *Local hierarchical h-refinements in IgA based on generalized B-splines*, Mathematical Methods for Curves and Surfaces 2012 (M. Floater, T. Lyche, M. L. Mazure, K. Mørken, and L. L. Schumaker, eds.), Lecture Notes in Computer Science, vol. 8177, Springer-Verlag, Heidelberg, 2014, pp. 341–363.
 - [81] C. Manni, A. Reali, and H. Speleers, *Isogeometric collocation methods with generalized B-splines*, Computers & Mathematics with Applications **70** (2015), 1659–1675.
 - [82] C. Manni, F. Roman, and H. Speleers, *Generalized B-splines in isogeometric analysis*, Approximation Theory XV: San Antonio 2016 (G. E. Fasshauer and L. L. Schumaker, eds.), Springer Proceedings in Mathematics & Statistics, vol. 201, Springer International Publishing, 2017, pp. 239–267.
 - [83] B. Marussig, R. Hiemstra, and T. J. R. Hughes, *Improved conditioning of isogeometric analysis matrices for trimmed geometries*, Computer Methods in Applied Mechanics and Engineering **334** (2018), 79 – 110.
 - [84] M. L. Mazure, *Chebyshev spaces and Bernstein bases*, Constructive Approximation **22** (2005), 347–363.
 - [85] ———, *Extended Chebyshev piecewise spaces characterised via weight functions*, Journal of Approximation Theory **145** (2007), 33–54.
 - [86] ———, *Finding all systems of weight functions associated with a given extended Chebyshev space*, Journal of Approximation Theory **163** (2011), 363–376.
 - [87] ———, *How to build all Chebyshevian spline spaces good for geometric design?*, Numerische Mathematik **119** (2011), 517–556.
 - [88] ———, *Constructing totally positive piecewise Chebyshevian B-spline bases*, Journal of Computational and Applied Mathematics **342** (2018), 550–586.
 - [89] B. Mourrain, *On the dimension of spline spaces on planar T-meshes*, Mathematics of Computation **83** (2014), 847–871.
 - [90] G. Nürnberger, L. L. Schumaker, M. Sommer, and H. Strauss, *Generalized Chebyshevian splines*, SIAM Journal on Mathematical Analysis **15** (1984), 790–804.
 - [91] F. Patrizi, *Effective grading refinement for locally linearly independent LR B-splines*, BIT Numerical Mathematics **62** (2022), 1745–1764.

- [92] F. Patrizi and T. Dokken, *Linear dependence of bivariate minimal support and locally refined B-splines over LR-meshes*, Computer Aided Geometric Design **77** (2020), 101803.
- [93] F. Patrizi, C. Manni, F. Pelosi, and H. Speleers, *Adaptive refinement with locally linearly independent LR B-splines: Theory and applications*, Computer Methods in Applied Mechanics and Engineering **369** (2020), 113230.
- [94] L. Piegl and W. Tiller, *The NURBS book*, Springer-Verlag, New York, 1995.
- [95] Hartmut Prautzsch, *Degree elevation of B-spline curves*, Computer Aided Geometric Design **1** (1984), 193–198.
- [96] L. Ramshaw, *Blossoms are polar forms*, Computer Aided Geometric Design **6** (1989), 323–358.
- [97] K. Raval, C. Manni, and H. Speleers, *Tchebycheffian B-splines in isogeometric Galerkin methods*, Computer Methods in Applied Mechanics and Engineering **403** (2023), 115648.
- [98] D. F. Rogers, *An introduction to NURBS: With historical perspective*, Morgan Kaufmann Publisher, San Francisco, 2001.
- [99] F. Roman, C. Manni, and H. Speleers, *Spectral analysis of matrices in Galerkin methods based on generalized B-splines with high smoothness*, Numerische Mathematik **135** (2017), 169–216.
- [100] J. Sánchez-Reyes, *Harmonic rational Bézier curves, p-Bézier curves and trigonometric polynomials*, Computer Aided Geometric Design **15** (1998), 909–923.
- [101] E. Sande, C. Manni, and H. Speleers, *Explicit error estimates for spline approximation of arbitrary smoothness in isogeometric analysis*, Numerische Mathematik **144** (2020), 889–929.
- [102] D. Schillinger, L. Dedè, M. A. Scott, J. A. Evans, M. J. Borden, E. Rank, and T. J. R. Hughes, *An isogeometric design-through-analysis methodology based on adaptive hierarchical refinement of NURBS, immersed boundary methods, and T-spline [CAD] surfaces*, Computer Methods in Applied Mechanics and Engineering **249–252** (2012), 116 – 150.
- [103] I. J. Schoenberg, *On spline functions*, Inequalities (O. Shisha, ed.), Academic Press, New York, 1967, pp. 255–286.

-
- [104] L. L. Schumaker, *Spline functions: Basic theory*, third ed., Cambridge University Press, Cambridge, 2007.
 - [105] L. L. Schumaker and L. Wang, *Approximation power of polynomial splines on T-meshes*, Computer Aided Geometric Design **29** (2012), 599–612.
 - [106] M. A. Scott, X. Li, T. W. Sederberg, and T. J. R. Hughes, *Local refinement of analysis-suitable T-splines*, Computer Methods in Applied Mechanics and Engineering **213–216** (2012), 206 – 222.
 - [107] M. A. Scott, R. N. Simpson, J. A. Evans, S. Lipton, S. P. A. Bordas, T. J. R. Hughes, and T. W. Sederberg, *Isogeometric boundary element analysis using unstructured T-splines*, Computer Methods in Applied Mechanics and Engineering **254** (2013), 197 – 221.
 - [108] T. W. Sederberg, D. L. Cardon, G. T. Finnigan, N. S. North, J. Zheng, and T. Lyche, *T-spline simplification and local refinement*, ACM Transactions on Graphics **23** (2004), 276–283.
 - [109] T. W. Sederberg, J. Zheng, A. Bakenov, and A. Nasri, *T-splines and T-NURCCs*, ACM Transactions on Graphics **22** (2003), 477–484.
 - [110] T. W. Sederberg, J. Zheng, and X. Song, *Knot intervals and multi-degree splines*, Computer Aided Geometric Design **20** (2003), 455–468.
 - [111] W. Shen and G. Wang, *A basis of multi-degree splines*, Computer Aided Geometric Design **27** (2010), 23 – 35.
 - [112] V. Skytt and T. Dokken, *Scattered data approximation by LR B-spline surfaces: A study on refinement strategies for efficient approximation*, Geometric Challenges in Isogeometric Analysis (C. Manni and H. Speleers, eds.), Springer INdAM Series, vol. 49, Springer, Cham, 2022, pp. 217–258.
 - [113] H. Speleers, *Hierarchical spline spaces: quasi-interpolants and local approximation estimates*, Advances in Computational Mathematics **43** (2017), 235–255.
 - [114] _____, *Algorithm 999: Computation of multi-degree B-splines*, ACM Transactions on Mathematical Software **45** (2019), 43.
 - [115] _____, *Algorithm 1020: Computation of multi-degree Tchebycheffian B-splines*, ACM Transactions on Mathematical Software **48** (2022), 12.
 - [116] H. Speleers and C. Manni, *Effortless quasi-interpolation in hierarchical spaces*, Numerische Mathematik **132** (2016), 155–184.

- [117] H. Speleers, C. Manni, F. Pelosi, and M. L. Sampoli, *Isogeometric analysis with Powell-Sabin splines for advection-diffusion-reaction problems*, Computer Methods in Applied Mechanics and Engineering **221–222** (2012), 132–148.
- [118] H. Speleers and D. Toshniwal, *A general class of C^1 smooth rational splines: Application to construction of exact ellipses and ellipsoids*, Computer-Aided Design **132** (2021), 102982.
- [119] I. Stangeby and T. Dokken, *Properties of spline spaces over structured hierarchical box partitions*, Isogeometric Analysis and Applications 2018 (H. van Brummelen et al., eds.), Lecture Notes in Computational Science and Engineering, vol. 133, Springer, Cham, 2021, pp. 177–207.
- [120] W. Tiller, *Rational B-splines for curve and surface representation*, IEEE Computer Graphics and Applications **3** (1983), 61–69.
- [121] D. Toshniwal, H. Speleers, R. R. Hiemstra, and T. J. R. Hughes, *Multi-degree smooth polar splines: A framework for geometric modeling and isogeometric analysis*, Computer Methods in Applied Mechanics and Engineering **316** (2017), 1005–1061.
- [122] D. Toshniwal, H. Speleers, R. R. Hiemstra, C. Manni, and T. J. R. Hughes, *Multi-degree B-splines: Algorithmic computation and properties*, Computer Aided Geometric Design **76** (2020), 101792.
- [123] A. V. Vuong, C. Giannelli, B. Jüttler, and B. Simeon, *A hierarchical approach to adaptive local refinement in isogeometric analysis*, Computer Methods in Applied Mechanics and Engineering **200** (2011), 3554 – 3567.
- [124] X. Wei, X. Li, K. Qian, T. J. R. Hughes, Y. J. Zhang, and H. Casquero, *Analysis-suitable unstructured t-splines: Multiple extraordinary points per face*, Computer Methods in Applied Mechanics and Engineering **391** (2022), 114494.
- [125] X. Wei, Y. Zhang, L. Liu, and T. J. R. Hughes, *Truncated T-splines: fundamentals and methods*, Computer Methods in Applied Mechanics and Engineering **316** (2017), 349–372.
- [126] X. Wei, Y. J. Zhang, and T. J. R. Hughes, *Truncated hierarchical tricubic C0 spline construction on unstructured hexahedral meshes for isogeometric analysis applications*, Computers & Mathematics with Applications **74** (2017), 2203 – 2220.

- [127] X. Wei, Y. J. Zhang, D. Toshniwal, H. Speleers, X. Li, C. Manni, J. A. Evans, and T. J. R. Hughes, *Blended B-spline construction on unstructured quadrilateral and hexahedral meshes with optimal convergence rates in isogeometric analysis*, Computer Methods in Applied Mechanics and Engineering **341** (2018), 609–639.
- [128] C. Zimmermann and R. A. Sauer, *Adaptive local surface refinement based on LR NURBS and its application to contact*, Computational Mechanics **60** (2017), 1011–1031.