ICES REPORT 17-27

October 2017

Improved Conditioning of Isogeometric Analysis Matrices for Trimmed Geometries

by

Benjamin Marussiga, René Hiemstra, and Thomas J. R. Hughes

The Institute for Computational Engineering and Sciences
The University of Texas at Austin
Austin, Texas 78712

Reference: Benjamin Marussiga, René Hiemstra, and Thomas J. R. Hughes, "Improved Conditioning of Isogeometric Analysis Matrices for Trimmed Geometries," ICES REPORT 17-27, The Institute for Computational Engineering and Sciences, The University of Texas at Austin, October 2017.

Improved Conditioning of Isogeometric Analysis Matrices for Trimmed Geometries

Benjamin Marussig[a,*], René Hiemstra[b], Thomas J. R. Hughes[b]

[a] *Institute of Structural Analysis, Graz University of Technology, Lessingstraße 25/II, 8010 Graz, Austria*
[b] *Institute for Computational Engineering and Sciences,The University of Texas at Austin, 201 East 24th St, Stop C0200,Austin, TX 78712, USA*

Abstract

A stable basis for isogeometric analysis of trimmed models is obtained by combining extended B-splines with truncated hierarchical B-splines. While extended B-splines guarantee that the condition number of system matrices is independent of the location of a trimming curve, local refinement is used to improve the robustness of the procedure and the accuracy of the numerical results. The present extended B-spline construction works in the context of Galerkin and collocation methods. The paper focuses on the latter and introduces a new collocation scheme for truncated hierarchical B-splines. A proper transition between refinement levels is assured by a novel balancing algorithm that employs a simple criterion. The enhanced performance of the locally refined stabilization is verified by scalar Laplace and linear elasticity problems analyzed by a collocation based isogeometric boundary element method. The proposed approach yields excellent results and requires few refinement levels to improve the stabilization procedure and accuracy along trimming curves.

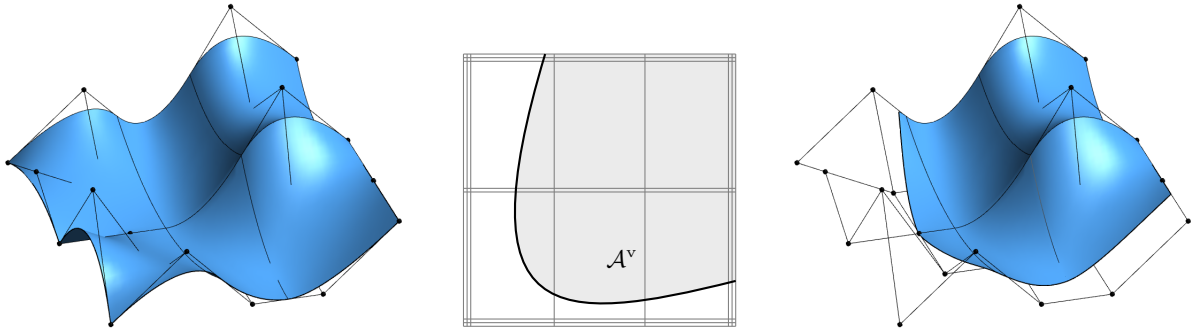
Keywords: Isogeometric Analysis, Trimming, Extended B-splines, WEB-splines, THB-splines, BEM, Collocation,

1. Introduction

Trimming is an ubiquitous procedure in Computer Aided Design (CAD) systems. It is used to represent models of arbitrary topological genus simply by displaying specified areas $\mathcal{A}^{\mathrm{v}}$ of a tensor product surface. In essence this is an optical illusion since the underlying mathematical description remains unaltered. Consider the surface depicted in Figure 1: instead of defining a set of control points and a corresponding parameterization that is aligned with the boundaries of the object sought 1(c), a tensor product surface 1(a) is used as a latent representation in the background and a trimming curve, defined in its parameter space 1(b), indicates the final shape shown by the graphics system. Unfortunately, trimming procedures lead to several severe problems within CAD and downstream applications [29]. The present paper is concerned with one of these issues – the stability of a trimmed basis. Trimming procedures may compromise the conditioning of system matrices if only a small portion of a associated basis function's support is within $\mathcal{A}^{\mathrm{v}}$. This problem arises as soon as a trimmed basis is incorporated into an analysis, where trimming is no longer a primarily visual process.

An elegant way to re-establish the stability of a trimmed basis is the utilization of *extended B-splines*. Originally, these splines have been developed in the context of a B-spline based fictitious domain method referred to as WEB-splines [17, 18, 19, 20]. Most extended B-spline applications are tailored to Galerkin formulations based on uniform parameter spaces, but the concept is not restricted to this setting. Non-uniform extended B-splines have been introduced in [18]. Recently, this stabilization has also been applied

*Corresponding author. mail: marussig@utexas.edu



(a) Original surface (b) Trimmed parameter space (c) Trimmed surface

Figure 1: Trimmed tensor product surface: (a) regular surface defined by a tensor product basis, (b) trimmed parameter space where a trimming curve specifies the visible part $\mathcal{A}^{\mathrm{v}}$ of (c) the resulting trimmed surface.

to collocation schemes for uniform [2] and non-uniform [31] parameter spaces. Regarding the extended B-splines approach, the main difference between Galerkin and collocation is the classification of stable and potentially unstable B-splines. The former usually relates the size of a support inside of $\mathcal{A}^{\mathrm{v}}$ to its exterior counterpart and sets a threshold for this ratio. The latter defines a B-spline as stable when its anchor, which specifies the position of the collocation point, is within $\mathcal{A}^{\mathrm{v}}$. The collocation based classification is a useful option also for Galerkin formulations due to its simplicity and ease of implementation.

In this work, the robustness of extended B-splines is enhanced by appropriate local refinement along the trimming curves. The refinement is carried out by truncated hierarchical B-splines (THB-splines) [14]. The combination of extended B-splines and classical hierarchical B-splines has already been considered in [17]. The benefits of the truncated basis, however, have not been utilized yet. The proposed *extended THB-spline* approach forms a partition of unity and the overlap of refinement levels is reduced, which improves the conditioning and stability of the refined basis. Furthermore, the concept decouples the extended B-spline stabilization from the local refinement process. To be precise, decoupling is accomplished by restricting the construction of extended B-splines to the last level of the hierarchical basis. This makes the implementation of extended THB-splines very simple. In fact, the stabilization itself is performed exactly the same way as described in [31]. A main part of the paper focuses on the application of the local refinement. Most importantly, a novel balancing algorithm is introduced that guarantees a proper transition from one level to another using a simple refinement rule. In addition, an alternative collocation scheme for THB-splines is proposed which uses the maximal values of truncated basis functions to specify the collocation points in the region where two levels overlap. As a result, every basis function is associated to only one collocation point in contrast to existing schemes [38].

The paper is structured as follows: First, the concept of extended B-splines is outlined in Section 2. The essential findings of [31] are recapitulated and the potential impact of the polynomial degree on the stabilization is discussed. Section 3 is dedicated to THB-splines, their implementation, and corresponding collocation concepts. Finally, these parts are united in Section 4. In Section 5, the resulting extended THB-spline basis is applied to various Laplace and elasticity problems solved by a collocated isogeometric boundary element method (BEM). The numerical results are compared to analytic results and reference solutions obtained by non-trimmed models as well as trimmed models discretized by conventional extended B-splines.

2. Extended B-splines

Stabilization of trimmed parameter spaces by means of extended B-splines is outlined. First the fundamental properties of B-splines are briefly reviewed and a classification of B-splines of a trimmed basis is provided. Based on this, the key points of extended B-splines are derived. The section closes with a

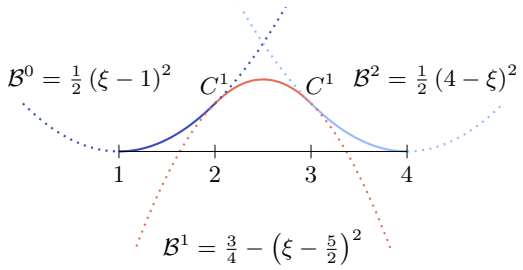


Figure 2: Polynomial segments $\mathcal{B}^s$ of a quadratic B-spline defined by $\Xi = \{1, 2, 3, 4\}$ and the continuity C between them. The resulting B-spline is illustrated by solid lines, whereas dotted lines indicate extensions of its segments.

discussion on the influence of the polynomial degree on an extended B-spline basis, which motivates the enhancement by local refinement.

2.1. Conventional B-splines

A B-spline $B_{i,p}$ is described by piecewise polynomial segments $\mathcal{B}_i^s$ of degree p. These $\mathcal{B}_i^s$ and the continuity between them are specified by the *knot vector* Ξ which is a non-decreasing sequence of parametric coordinates $\xi_j \leqslant \xi_{j+1}$. The components ξ_j of Ξ are called *knots* and they indicate the location where adjacent $\mathcal{B}_i^s$ join. The continuity at these *breakpoints* is C^{p-m} with m denoting the multiplicity of the corresponding knot value, i.e., $\xi_j = \xi_{j+1} = \cdots = \xi_{j+m-1}$. If successive knots do not share the same value, i.e., $\xi_s < \xi_{s+1}$, they describe a (non-zero) *knot span* and the half open interval $[\xi_s, \xi_{s+1})$ marks the valid region of $\mathcal{B}_i^s$. These knot spans partition the domain $[a, b]$ described by the knot vector into a set of elements. Figure 2 illustrates polynomial segments of a quadratic B-spline. Note that each $\mathcal{B}^s$ is restricted to its knot span for the representation of the B-spline function, but they could be evaluated outside of their limits as well.

The knot vector Ξ defines not only a single function, but an entire set of linearly independent B-splines $\{B_{i,p}\}_{i=0}^n$. The linear span of B-splines – the space of all possible spline functions on $[a, b]$ that are C^{p-m} at the breakpoints – is completely defined upon the choice of polynomial degree p and knot vector Ξ. That is, the space of splines on $[a, b]$ is given by

$$\mathbb{S}_{\Xi,p}([a, b]) := \left\{ \sum_{i=0}^{n} B_{i,p}(\xi) c_i \;\middle|\; \xi \in [a, b],\ c_i \in \mathbb{R}, \quad i = 0, \ldots, n \right\}. \tag{1}$$

A B-spline basis forms a positive partition of unity

$$\begin{cases} B_{i,p}(\xi) \geq 0, \\ \sum_{i=0}^{n} B_{i,p}(\xi) = 1, \quad \forall \xi \in [a, b]. \end{cases} \tag{2}$$

Each $B_{i,p}$ has local support, $\operatorname{supp}\{B_{i,p}\}$, specified by the knots $\{\xi_i, \ldots, \xi_{i+p+1}\}$ and each knot span s contains $p + 1$ non-zero B-splines represented by $\mathcal{B}_i^s$ with $i = s - p, \ldots, s$. These functions locally span the full space of degree p polynomials

$$\mathcal{P}^p([\xi_s, \xi_{s+1}]) \equiv \operatorname{span}\{B_{s-p,p}(\xi), ..., B_{s,p}(\xi)\}. \tag{3}$$

The multiplicity of the first and last knot values is often set to $p+1$, resulting in interpolatory basis functions at the beginning and the end of the basis. In this case the knot vector is called *open*.

Basis functions of B-spline surfaces are obtained by computing the tensor product of univariate B-splines which are given by separate knot vectors Ξ_1 and Ξ_2. In general, multivariate basis functions of any dimension δ are determined by

$$B_{\mathbf{i},\mathbf{p}}(\boldsymbol{\xi}) = \prod_{j=1}^{\delta} B_{i_j,p_j}(\xi_j) \tag{4}$$

with multi-indices $\mathbf{i} = \{i_1, \ldots, i_\delta\}$ and $\mathbf{p} = \{p_1, \ldots, p_\delta\}$. The former defines the position of the basis function in the tensor product structure and the latter represents the degree in each parametric direction. The related univariate B-spline in the i-direction is denoted by B_{i_j,p_j}.

B-splines and their derivatives are evaluated recursively by convex combinations of B-splines of lower degree. For details on efficient evaluation algorithms the interested reader is referred to [34]. It is pointed out that there is usually no explicit representation of the polynomial segments $\mathcal{B}^s$ required.

A parametric representation of a curve in the model space $\mathbb{R}^d$ is given by

$$\boldsymbol{C}(\xi) = \sum_{i=0}^{n} \boldsymbol{c}_i B_{i,p}(\xi) \tag{5}$$

where $\boldsymbol{c}_i \in \mathbb{R}^d,\ i = 0, \ldots, n$ are the *control points* of the B-spline curve. Their piecewise linear interpolant is called *control polygon*. B-spline surfaces are specified in an analogous manner using bivariate B-splines and a regular grid of control points as illustrated in Figure 1.

2.2. B-splines of a trimmed basis

Trimming is a procedure that allows the visualization of arbitrary regions $\mathcal{A}^{\mathrm{v}}$ over a B-spline object. It is usually applied to overcome some limitations of tensor product surfaces. Without loss of generality, we are going to focus on the univariate setting to clarify the effect of trimming on a B-spline basis.

Suppose a parameter space is trimmed at a position t, three different types of basis functions arise. In order to classify these various types, each B-spline $B_{i,p}$ is associate to an *anchor* $\bar{\xi}_i$, i.e., a specific location in the parameter space. These will be defined by the Greville abscissae

$$\bar{\xi}_i = \frac{\xi_{i+1} + \xi_{i+2} + \cdots + \xi_{i+p}}{p}. \tag{6}$$

Based on the anchors $\bar{\xi}_i$ and the support of the basis functions $\operatorname{supp}\{B_{i,p}\}$, B-splines of a trimmed basis are classified as:

- *Stable* if $\bar{\xi}_i \in \mathcal{A}^{\mathrm{v}}$
- *Degenerate* if $\bar{\xi}_i \notin \mathcal{A}^{\mathrm{v}}$ and $\operatorname{supp}\{B_{i,p}\} \cap \mathcal{A}^{\mathrm{v}} \neq \emptyset$
- *Exterior* if $\operatorname{supp}\{B_{i,p}\} \cap \mathcal{A}^{\mathrm{v}} = \emptyset$

An example of these different types is shown in Figure 3. Note that the given specification is by no means restricted to the univariate case; it can be easily applied to trimmed surfaces as well. Some authors identify the different B-splines of a trimmed parameter space based on the size of their support within $\mathcal{A}^{\mathrm{v}}$, see e.g., [17, 35]. The proposed scheme, however, provides a simple and fast separation of stable, degenerate, and exterior functions and it guarantees that anchors of stable B-splines are within the domain of interest. The latter is essential for isogeometric methods that use the B-spline anchors for collocation.

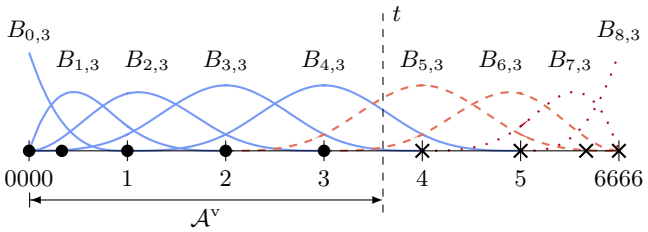
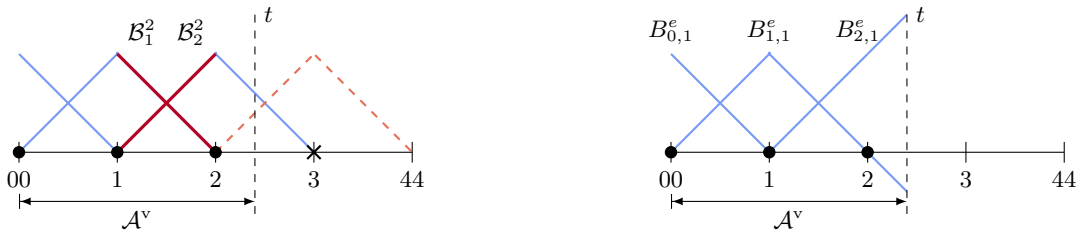


Figure 3: Types of basis functions of a trimmed cubic B-spline basis: stable (solid), degenerate (dashed), and exterior (dotted). Circles mark anchors of basis functions that are within the valid domain $\mathcal{A}^{\mathrm{v}}$, while crosses indicate those outside the valid domain $\mathcal{A}^{\mathrm{v}}$.



(a) Initial B-splines

(b) Extended B-splines

Figure 4: Conversion of a potentially unstable B-spline basis to a stable extended B-spline basis: (a) initial stable and degenerate basis functions and (b) the resulting extended B-splines. In (a), the polynomial segments $\mathcal{B}_1^2$ and $\mathcal{B}_2^2$ that provide the extensions for the stabilization are highlighted by thick red lines.

The main purpose of the type classification is to detect degenerate B-splines. They are of particular interest, because their support may become arbitrary small due to the trimming procedure. In the example depicted in Figure 3, this would be the case for $B_{6,3}$ as the trimming parameter t approaches the knot value 3. Trimmed basis functions with arbitrary small support can yield serious stability issues, when they are incorporated in an analysis as highlighted in [31]. Extended B-splines presented in the following section are an elegant method to resolve this problem.

2.3. Definition of extended B-splines

The fundamental idea of extended B-splines $B_{i,p}^e$ is to substitute the polynomial segments of knot spans that contain degenerate functions by *extensions* of stable ones. As a result, degenerate B-splines are eliminated from the basis and thus, the domain of interest $\mathcal{A}^{\mathrm{v}}$ is described by stable basis functions only.

Considering univariate B-splines, the extensions are provided by the polynomial segments $\mathcal{B}_i^s$ where s denotes the closest knot span with respect to the boundary of $\mathcal{A}^{\mathrm{v}}$ which contains only stable B-splines. Figure 4 illustrates the extension concept and the components involved for a set of linear basis functions; exterior B-splines are omitted since they do not make any contribution to the basis. It should be noted that the supports of the resulting extended B-splines $B_{i,p}^e$ shown in Figure 4(b) can never become arbitrarily small. Thus, an extended B-spline basis is stable.

Extended B-splines can be defined as a linear combination of the original B-splines. To be precise, the extended polynomial segments $\mathcal{B}_i^s$ of the non-trimmed knot span s are expressed by B-splines of the trimmed knot span t such that

$$\mathcal{B}_i^s(\xi) = \sum_{j=t-p}^{t} B_{j,p}(\xi)\, e_{i,j} \qquad\qquad \xi \in [\xi_t, \xi_{t+1})\,. \tag{7}$$

This representation is exact since all $\mathcal{B}_i^s \in \mathbb{S}_{\Xi,p}$. The main task is to determine the values of the *extrapolation weights* $e_{i,j}$. If the index j corresponds to a stable B-spline, the weights are trivial, i.e.,

$$e_{i,i} = 1 \qquad \Leftrightarrow \qquad \mathcal{B}_i^s(\xi) \equiv B_{i,p}(\xi), \quad \xi \in [\xi_s, \xi_{s+1})\,, \quad \forall i \in \{s-p, \dots, s\}, \tag{8}$$

$$e_{i,j} = 0 \qquad \Leftrightarrow \qquad \mathcal{B}_i^s(\xi) \neq B_{j,p}(\xi), \quad \xi \in [\xi_s, \xi_{s+1})\,, \quad \forall j \in \{s-p, \dots, s\}\backslash i. \tag{9}$$

The remaining extrapolation weights related to degenerate B-splines can be determined by

$$e_{i,j} = \frac{1}{p!} \sum_{k=0}^{p} (-1)^k\, (p-k)!\, \beta_{p-k}\, k!\, \alpha_k \tag{10}$$

with the coefficients α and β denoting the constants of the polynomials

$$\mathcal{B}_i^s(\xi) = \sum_{k=0}^{p} \alpha_k\, \xi^k \qquad \text{and} \qquad \psi_{j,p}(\xi) = \prod_{m=1}^{p} (\xi - \xi_{j+m}) = \sum_{k=0}^{p} \beta_k\, \xi^k. \tag{11}$$

The former is the power basis form of the extended polynomial segment which may be derived by Taylor expansion, and the Newton polynomials $\psi_{j,p}$ result from a quasi interpolation procedure, i.e., the *de Boor–Fix* functional [9, 10]. A detailed discussion about the conversion of the polynomials to power basis form and the role of quasi interpolation is given in [31]. By taking the trivial extrapolations weights (8) and (9) into account, an extended B-spline is defined as

$$B^e_{i,p} = B_{i,p} + \sum_{j \in \mathbb{J}_i} e_{i,j} B_{j,p} \tag{12}$$

where $B_{i,p}$ is the stable B-spline from which the extension originates, and $\mathbb{J}_i$ is the index set of all degenerate B-splines related to $B^e_{i,p}$. Definition (12) also applies to bivariate basis functions of B-spline surfaces. The corresponding extrapolation weights are determined by the tensor product of the univariate counterparts. For more information regarding the properties of extended B-splines, the interested reader is referred to [17, 31].

2.4. Application to a system of equations

A very convenient feature of extended B-splines is that they do *not* need to be explicitly evaluated during an analysis. Suppose we have a linear system of n equations, one for each stable B-spline, set up by all basis functions m which are at least partially inside $\mathcal{A}^{\mathrm{v}}$. This yields a system of equations

$$\mathbf{K}\mathbf{u} = \mathbf{f} \qquad \text{where} \qquad \mathbf{u} \in \mathbb{R}^m, \mathbf{f} \in \mathbb{R}^n \qquad \text{and} \qquad \mathbf{K} \in \mathbb{R}^{n \times m} \qquad \text{with} \qquad m > n. \tag{13}$$

Up to this point, only conventional B-splines have been used to compute the system matrix $\mathbf{K}$. In order to get a stabilized square system matrix, $\mathbf{K}$ is multiplied by an *extension matrix* $\mathbf{E} \in \mathbb{R}^{m \times n}$ [17]. The entries of $\mathbf{E}$ are the extrapolation weights $e_{i,j}$ of all extended B-splines. The trivial weights $e_{i,i} = 1$ are stored as well, even if, a stable B-spline has no related degenerate ones. The matrix entries of $\mathbf{E}$ are assembled such that columns of the ith row of $\mathbf{K}$ are distributed according to the definition (12) of the associated extended B-spline $B^e_{i,p}$. In case of multi-patch models, the indices refer to the global degree of freedom. The stable system of equations due to the extended B-spline basis is determined by

$$\mathbf{K}_{st}\mathbf{u}_{st} = \mathbf{f} \qquad \text{with} \qquad \mathbf{K}_{st} = \mathbf{K}\mathbf{E}, \qquad \mathbf{K}_{st} \in \mathbb{R}^{n \times n}. \tag{14}$$

The solution vector $\mathbf{u}_{st} \in \mathbb{R}^n$ corresponds to the extended B-spline basis. Its relation to the original basis can also be expressed by the extension matrix as $\mathbf{u} = \mathbf{E}\mathbf{u}_{st}$.

2.5. Application to NURBS models

The stabilization described is tailored to B-spline functions where it is exploited that the extensions of any polynomial segment $\mathcal{B}^s_i$ can be exactly represented by a linear combination of basis functions of the trimmed knot span. In case of non-uniform rational B-splines (NURBS), this property is not guaranteed due to the local influence of the NURBS weights which are associated to the basis functions. In order to apply extended B-splines to a trimmed NURBS basis, we propose the application of an independent field approximation [28, 30, 32]. The basic idea of independent field approximation is to use different basis functions for the representation of the geometry and the approximations of the physical fields. That is, conventional B-splines are used for the discretization of the field variables over NURBS patches. To be precise, the B-spline knot vectors introduced are constructed as a superset of the knot vector of the geometry in order to take the object's continuity into account. This allows a straightforward application of extended B-splines. In addition, the combination of NURBS for the geometry description and B-splines for the field approximations has been shown to be more efficient [28] and does not lead to a loss of accuracy [8, 25, 30]. There is one caveat: independent fields are inconsistent with the isoparametric concept in mechanics [21].

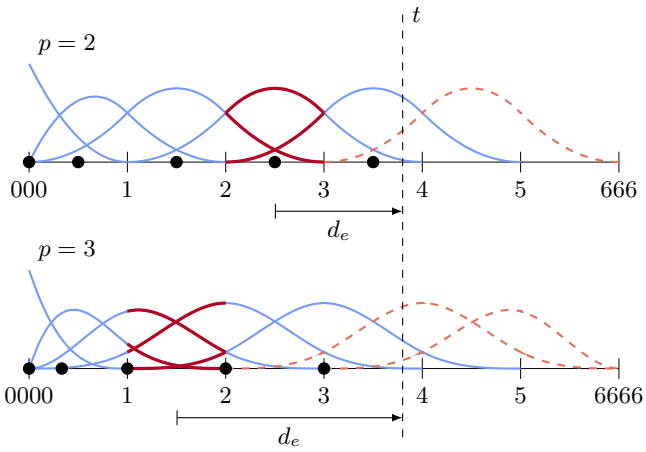


Figure 5: Correlation between the polynomial degree p and the extrapolation length d_e by comparing a quadratic basis (top) with its cubic counterpart (bottom).

2.6. Extended B-splines for higher degree

The definition of extended B-splines (12) is applicable to any degree p. However, the degree has an impact on the extrapolation process. That is, the area represented by extensions, rather than original B-splines, increases with p due to the fact that supports of degenerate B-splines propagate further into $\mathcal{A}^{\mathrm{v}}$ as indicated in Figure 5. We specify the *extrapolation length*, d_e, by the shortest distance of the boundary of $\mathcal{A}^{\mathrm{v}}$ to the center of the knot span that provides the extensions.

A large extrapolation length may increase the approximation error in the vicinity of the trimming curve. For instance, it has been demonstrated in [31] that lower degree extended B-splines perform almost like discretizations based on regular B-spline patches, whereas noticeable accuracy differences occur in higher degree cases. This approximation error is only introduced in the representation of the physical fields of the problem considered and does not affect the geometry description. Recall that the initial system matrix $\mathbf{K}$ is set up using conventional B-splines before it is stabilized by the extension matrix. Hence, all geometry evaluations are automatically performed with respect to the original object.

Local refinement provides a means to gain control over the extrapolation length. In the context of tensor product surfaces, however, this topic requires further considerations which are outlined in the subsequent section. It may be argued that the extrapolation length could also be controlled if stable and degenerate basis functions are classified by the size of their support rather than the position of their anchors. For instance, a basis function could be marked as stable if at least one entire knot span of its support is within $\mathcal{A}^{\mathrm{v}}$. Consequently, the degree does not affect the extrapolation length anymore. On the other hand, this would necessarily entail that the valid ratio of the interior and exterior support of a stable B-spline evolves with the degree, which may lead to conditioning issues for higher degree. In other words, such a strategy would just replace the issue of the extrapolation length with a different problem. Furthermore, the resulting stable basis could not be applied to collocation schemes.

3. Local refinement technique

Local refinement of multivariate splines is an active area of research and several techniques have been developed, such as T-splines [6, 41], LR-B-splines [12, 22], and hierarchical B-splines [7, 23, 43]. In this work local refinement by means of truncated hierarchical B-splines is considered [14]. Besides its relative ease of implementation, hierarchical splines have been shown to work well with extended B-splines [17], which is of importance in our application. This section starts with the key concepts that are essential to all refinement techniques. Subsequently, we focus on hierarchical B-splines and discuss its extension to the truncated basis.

3.1. Function based refinement

In contrast to traditional mesh refinement in finite element analysis, refinement of splines is based on the concept of *function refinement*. Refinement of the underlying mesh is implicit in its construction. This alternate viewpoint has lead to relatively simple refinement procedures that achieve high levels of inter-element continuity with automatic treatment of hanging nodes. The fundamental technique used is called *knot insertion*. A local multivariate interpretation of knot insertion is used in the context of T-splines and LR-B-splines. Other local refinement techniques, such as hierarchical B-splines, are based on *subdivision*, a technique that can be derived from knot insertion.

3.1.1. Knot insertion

Consider a parametric B-spline curve defined by a knot vector Ξ and polynomial degree p,

$$\boldsymbol{C}(\xi) = \sum_{j=0}^{n} \boldsymbol{c}_j B_{j,p}(\xi). \tag{15}$$

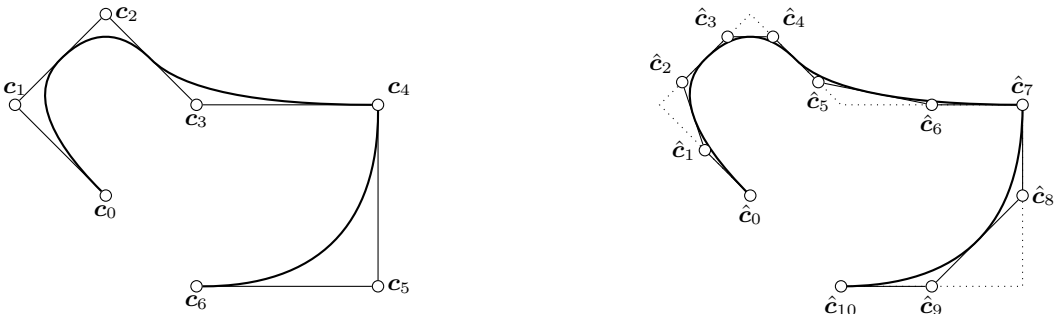
Let $\Xi \subset \hat{\Xi}$, that is, $\hat{\Xi}$ can be thought of the knot vector that is obtained by inserting one or more knots into Ξ. This leads to nested spline spaces $\mathbb{S}_{\Xi,p} \subset \mathbb{S}_{\hat{\Xi},p}$. Hence, we can write,

$$\boldsymbol{C}(\xi) = \sum_{j=0}^{n} \boldsymbol{c}_j B_{j,p}(\xi) = \sum_{i=0}^{\hat{n}} \sum_{j=0}^{n} \left(\mathbf{S}_{ij}\boldsymbol{c}_j\right) \hat{B}_{i,p}(\xi) = \sum_{i=0}^{\hat{n}} \hat{\boldsymbol{c}}_i \hat{B}_{i,p}(\xi) = \hat{\boldsymbol{C}}(\xi) \tag{16}$$

where $\mathbf{S} : \mathbb{R}^{n+1} \mapsto \mathbb{R}^{\hat{n}+1}$ is a sparse matrix, denoted as the *subdivision matrix*. An example is shown in Figure 6. It is important to realize that the geometry and parametric representation do not change under refinement, that is, $\boldsymbol{C}(\xi) \equiv \hat{\boldsymbol{C}}(\xi)$.

The subdivision matrix encodes the relation between degrees of freedom of the initial coarse geometry description and the refined one by means of the equation,

$$\hat{\boldsymbol{c}}_i = \sum_{j=0}^{n} \mathbf{S}_{ij}\ \boldsymbol{c}_j \qquad \text{for } i = 0, \ldots, \hat{n}. \tag{17}$$



(a) Original curve

(b) Multiple knot insertion

(c) Original basis functions

(d) Basis functions after inserting multiple knots

Figure 6: Left: original B-spline curve and basis specified by $\Xi = \{0,0,0,1,2,3,3,4,4,4\}$ and the control points $\boldsymbol{c}_i$. Right: refined version of the B-spline based on knot insertion at $\hat{\xi}_j = \{0.5, 1.5, 2.5, 3.5\}$. The coordinates of the new control points $\hat{\boldsymbol{c}}_i$ are determined by the knot insertion rule (18) and the overall matrix $\mathbf{S}$ is given in equation (19).

The entries of $\mathbf{S}$ are computed as follows. First consider inserting a single knot into Ξ. In other words, $\hat{\Xi} = \Xi \cup \hat{\xi}$, where $\hat{\xi} \in [\xi_s, \xi_{s+1})$ initially has multiplicity m. We define a sparse matrix $\mathbf{A} : \mathbb{R}^{n+1} \mapsto \mathbb{R}^{n+2}$, with non-zero entries,

$$\begin{cases} \mathbf{A}(k, k-1) & = 1 - \alpha_k \\ \mathbf{A}(k, k) & = \alpha_k \end{cases} \qquad \alpha_k = \begin{cases} 1 & k \leqslant s - p \\ \frac{\hat{\xi} - \xi_k}{\xi_{k+p} - \xi_k} & s - p + 1 \leqslant k \leqslant s - m \\ 0 & k \geqslant s - m + 1 \end{cases} \tag{18}$$

Finally set $\mathbf{S} = \mathbf{A}$. Multiple knots can be inserted by repeating this process. In general, if r knots are inserted into Ξ, then the matrix $\mathbf{S}$ can be factored as $\mathbf{S} = \mathbf{A}_r \, \mathbf{A}_{r-1} \cdots \mathbf{A}_1$, where each $\mathbf{A}_i$ is a sparse matrix with a structure as in (18). Subdivision matrices $\mathbf{S}$ have positive entries only and have row-sum equal to one; properties that are directly inherited from the matrices $\mathbf{A}_i$. The subdivision matrix corresponding to the refinement example in Figure 6 is given below. Its columns relate to a process called subdivision, hence the name subdivision matrix.

$$\mathbf{S} = \begin{pmatrix} 1 & & & & & & & & \\ 1/2 & 1/2 & & & & & & & \\ & 3/4 & 1/4 & & & & & & \\ & 1/4 & 3/4 & 1/4 & & & & & \\ & & 3/4 & 3/4 & & & & & \\ & & 1/4 & 1/4 & & & & & \\ & & & 1/2 & 1/2 & & & & \\ & & & & 1 & 1/2 & & & \\ & & & & 1/2 & 1/2 & 1/2 & & \\ & & & & & & 1 & & \end{pmatrix}. \tag{19}$$

3.1.2. Subdivision

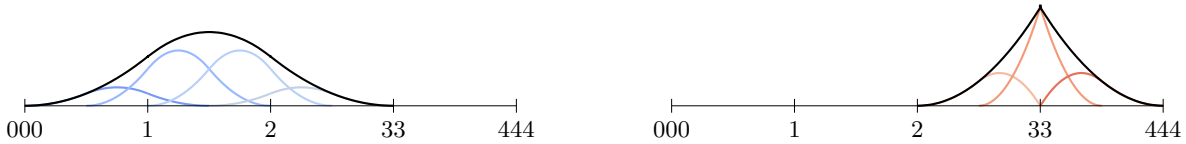
A dual perspective of knot insertion can be obtained by reordering the terms in (16),

$$\boldsymbol{C}(\xi) = \sum_{j=0}^{n} \boldsymbol{c}_j B_{j,p}(\xi) = \sum_{i=0}^{\hat{n}} \sum_{j=0}^{n} (\mathbf{S}_{ij} \boldsymbol{c}_j) \, \hat{B}_{i,p}(\xi) = \sum_{j=0}^{n} \boldsymbol{c}_j \sum_{i=0}^{\hat{n}} \left(\mathbf{S}_{ij} \hat{B}_{i,p}(\xi) \right) = \hat{\boldsymbol{C}}(\xi). \tag{20}$$

Hence, a coarse scale function can be exactly represented in terms of a linear combination of fine scale functions,

$$B_{j,p}(\xi) = \sum_{i=0}^{\hat{n}} \mathbf{S}_{ij} \; \hat{B}_{i,p}(\xi) \tag{21}$$

This is called a *two-scale relation*. We shall refer to $B_{j,p}(\xi)$ as the *parent* function and to $\{\hat{B}_{i,p}(\xi), \; \mathbf{S}_{ij} \neq 0\}$ as its *children*. Examples of two-scale relations are shown in Figure 7. These two examples correspond to the highlighted columns in the subdivision matrix $\mathbf{S}$ in (19).



(a) Two-scale relation for $B_{2,2}(\xi)$: $\{1/4, 3/4, 3/4, 1/4\}$

(b) Two-scale relation for $B_{4,2}(\xi)$: $\{1/2, 1, 1/2\}$

Figure 7: Examples of two-scale relations for non-uniform quadratic B-splines. The parent function is shown in black and its children in color.

3.2. Hierarchical B-splines

Multilevel hierarchical B-splines [23] provide local refinement through a hierarchy of nested spline spaces. Initial work includes its application in the WEB-spline method [17]. More recent work constitutes its use in isogeometric analysis [7, 16, 26, 37, 43] and its extension to the so-called truncated bases [14]. The theory is well established and several references exist detailing its implementation [7, 26, 40, 43]. We shall give a concise introduction to multilevel hierarchical splines, discuss its main attributes, and elaborate on the similarities and differences with other implementations.

3.2.1. Definition

Let Ω denote the n_{sd}-variate parametric domain and consider a sequence of $\ell_{\max}+1$ nested tensor product spline spaces,

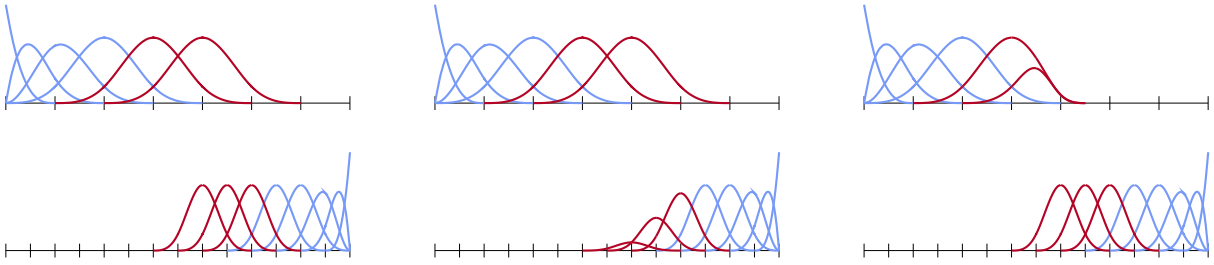
$$\mathbb{S}^{(0)}(\Omega) \subset \mathbb{S}^{(1)}(\Omega) \subset \mathbb{S}^{(2)}(\Omega) \subset \cdots \subset \mathbb{S}^{(\ell_{\max})}(\Omega) \tag{22}$$

where $\mathbb{S}^{(\ell)} := \text{span}\left\{B_i^{(\ell)} \text{ for } i \in \eta^{(\ell)} \equiv 0, 1, \ldots, n^{(\ell)}\right\}$. The subscripts of the spaces are neglected for the sake of brevity. However, we assume these spaces have the same polynomial degree p_I corresponding to each parametric direction and are defined by a sequence of nested knot vectors $\Xi_I^{(0)} \subset \Xi_I^{(1)} \subset \cdots \subset \Xi_I^{(\ell_{\max})}$, $I = 1, \ldots, n_{\text{sd}}$.

A multilevel spline space is defined as follows,

$$\mathcal{MS}(\Omega) := \text{span}\left\{B_i^{(\ell)} \text{ for } i \in \eta_A^{(l)},\ \ell = 0, 1, ..., \ell_{\max}\right\}. \tag{23}$$

Here $\eta_A^{(\ell)}$ is an index set that refers to *active* basis functions on level ℓ. Its complement, $\eta_D^{(\ell)}$, such that $\eta_A^{(\ell)} \cap \eta_D^{(\ell)} = \emptyset$ and $\eta_A^{(\ell)} \cup \eta_D^{(\ell)} = \eta^{(\ell)}$, refers to all B-splines that are *inactive* on level ℓ. Starting from an initial coarse spline space, $\mathbb{S}^{(0)}(\Omega)$, hierarchical refinement proceeds by deactivating parent functions on level ℓ and activating their children on level $\ell+1$. This operation is based on the two-scale relations presented in the previous section. Different approaches have been proposed in the literature, see e.g., [14, 23, 43], as illustrated in Figure 8. However, the fundamental properties to be maintained throughout the refinement process are global linear independence and nestedness of the spaces. This poses constraints on the index sets $\{\eta_A^{(0)}, \ldots, \eta_A^{(\ell_{\max})}\}$. There are other aspects to consider, such as properly balancing the refinements across the different levels. We shall discuss each of these topics below and describe their implementation.



(a) Classical (b) Scaled (c) Truncated

Figure 8: Different types of hierarchical B-splines basis. Basis functions corresponding to the overlapping area of two levels are highlighted in red. The basis in (b) and (c) form a partition of unity. In (b), this is achieved by scaling on the finer level, whereas truncation on the coarser level is used in (c).

Remark: Examples will be limited to the univariate setting. Furthermore, two-scale relations in the multivariate setting are obtained by tensor products of the univariate two-scale relations. In other words, two-scale relations are encoded in a subdivision matrix $\mathbf{S} := \mathbf{S}_1 \otimes \cdots \otimes \mathbf{S}_{n_{\text{sd}}}$, where $\mathbf{S}_i$, $i = 1, \ldots, n_{\text{sd}}$ denote the univariate subdivision matrices in each of the parametric directions. With these preliminaries, the presented material is independent of spatial dimension. We refer to [7] for additional information.

3.2.2. Maintaining nested spaces

Nested refinements imply polynomial reproducibility, which is important for accuracy of the method. A necessary and sufficient condition to maintain nested spaces is to make sure that,

$$\text{span}\left\{B_i^{(\ell)},\ i \in \eta_D^{(\ell)}\right\} \subset \text{span}\left\{B_i^{(k)},\ \text{for } i \in \eta_A^{(k)},\ k = \ell+1, ..., \ell_{\max}\right\}. \tag{24}$$

In words: functions that are *deactivated* on level ℓ should be representable by a linear combination of active functions on levels $\ell+1, \ldots, \ell_{\max}$. Since polynomials are in the space $\mathbb{S}^{(0)}(\Omega)$, it is easy to check by induction that this requirement maintains polynomial reproducibility at all stages of refinement.

Nested refinements are obtained in practice by studying the two-scale relation,

$$B_j^{(\ell)}(\xi) = \sum_{i \in \eta^{(\ell+1)}} \mathbf{S}_{ij}^{(\ell)} B_i^{(\ell+1)}(\xi). \tag{25}$$

Suppose $B_j^{(\ell)}(\xi)$ is deactivated. A sufficient condition to satisfy (24) is to activate all its children,

$$\{B_i^{(\ell+1)}(\xi),\ \mathbf{S}_{ij}^{(\ell)} \neq 0\}. \tag{26}$$

The subdivision matrices give all the information required to successfully perform this operation. Due to their sparsity and the fact that they are univariate operators, one may choose to explicitly store these matrices in a sparse format. This allows efficient access to the parent-child relation at any time.

3.2.3. Balancing the refinements

With each level we can associate a region corresponding to its active functions, that is,

$$\Omega_\ell := \bigcup_{i \in \eta_A^{(\ell)}} \text{supp}\left\{B_i^{(\ell)}(\xi)\right\}. \tag{27}$$

Figure 9 depicts an example of a three level hierarchical spline space and illustrates the regions $\Omega_0, \Omega_1, \Omega_2$. The scaled approach to hierarchical B-splines is shown because the scaling factors are of particular interest in what follows. Although not thoroughly discussed in the literature, it is important to control the size and arrangement of overlap regions $\Omega_\ell \cap \Omega_k \neq \emptyset,\ \ell \neq k$ [26, 40]. In other words, besides maintaining nested spaces and linear independence, one needs to appropriately *balance* the refinements across the separate levels. This benefits the conditioning of the final system and provides control over the footprint of the refinement [26, 40].

We present a very simple approach to this issue. Consider a vector of weights $\mathbf{w}^{(\ell)} = \{w_0^{(\ell)}, w_1^{(\ell)}, \ldots, w_{n^\ell}^{(\ell)}\}$, where,

$$w_i^{(0)} = \begin{cases} 1 & i \in \eta_A^{(0)} \\ 0 & i \in \eta_D^{(0)} \end{cases} \tag{28}$$

and when $\ell > 0$,

$$w_i^{(\ell)} = \begin{cases} \sum_{j \in \eta_D^{(l-1)}} \mathbf{S}_{ij}^{(\ell-1)} & i \in \eta_A^{(\ell)} \\ 0 & i \in \eta_D^{(\ell)} \end{cases} \tag{29}$$

We shall refer to the entries in $\mathbf{w}^{(\ell)}$ as the *subdivision weights*. Those corresponding to active functions on level ℓ are computed as the row-sum of the columns of $\mathbf{S}^{(\ell-1)}$ corresponding to deactivated functions on level $\ell - 1$. Those corresponding to inactive functions on level ℓ are zero. This is best explained by an example.

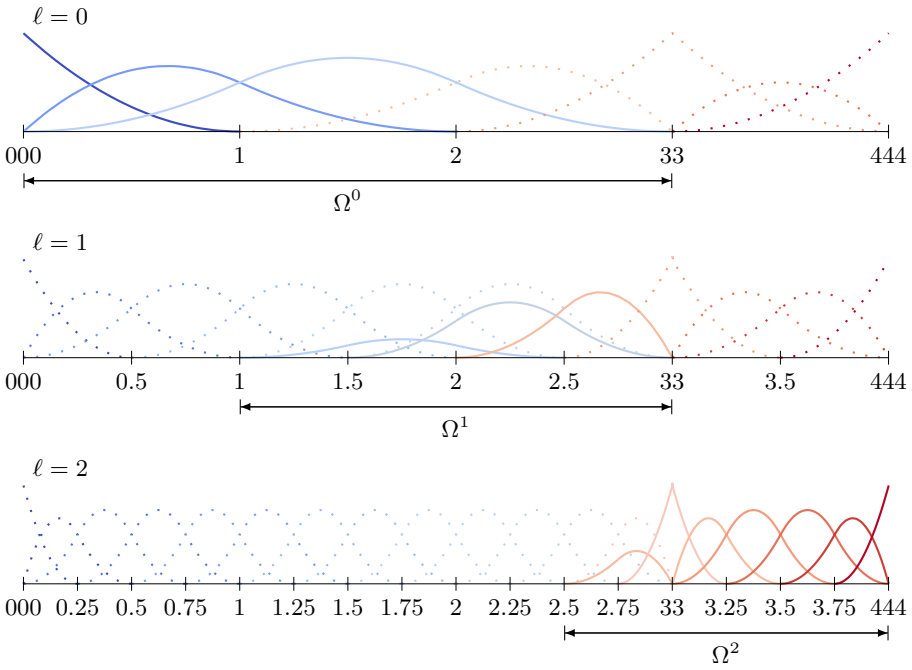


Figure 9: Three-level hierarchical B-spline basis. The functions are scaled using the subdivision weights to form a partition of unity. Active functions are represented by solid lines, whereas dotted lines represent the inactive functions.

Consider Figure 9. The coefficients of $\mathbf{w}^{(0)}$ are $\{1, 1, 1, 0, 0, 0, 0\}$. Those of $\mathbf{w}^{(1)}$ are computed as follows,

$$\mathbf{w}^{(1)} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1/4 \\ 3/4 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad \text{corresponding to} \quad \mathbf{S}^{(0)} = \left(\begin{array}{cccccc} 1 & & & & & \\ 1/2 & 1/2 & & & & \\ & 3/4 & 1/4 & & & \\ & 1/4 & 3/4 & & & \\ \hline & & 3/4 & 1/4 & & \\ & & 1/4 & 3/4 & & \\ & & & 1/2 & 1/2 & \\ \hline & & & & 1 & \\ & & & & 1/2 & 1/2 \\ & & & & & 1/2 & 1/2 \\ & & & & & & 1 \end{array}\right). \tag{30}$$

Only entries in $\mathbf{w}^{(1)}$ corresponding to active functions, highlighted in red, need to be computed. Their value corresponds to the row sum of columns $3-6$ of $\mathbf{S}^{(0)}$. The remaining values in $\mathbf{w}^{(1)}$ are zero by definition.

We classify the subdivision weights and divide them into three classes,

$$\begin{cases} w_i = 1 & \text{(active, refineable)} \\ 0 < w_i < 1 & \text{(active, not refineable)} \\ w_i = 0 & \text{(inactive)} \end{cases}. \tag{31}$$

Zero entries in $\mathbf{w}^{(\ell)}$ correspond to inactive functions on level ℓ, while non-zero entries correspond to activated ones. In practice, each $\mathbf{w}^{(\ell)}$ is stored in sparse vector format. Besides only having to store the non-zero values, this allows efficient access into active basis functions ($0 < w_i \leq 1$) and their complement ($w_i = 0$). Proper balancing of the refinement across levels is achieved by allowing functions to be refined only when their subdivision weight is equal to one. Consider again Figure 9 and the flow-diagram below. Suppose functions

3-6 corresponding to level 0 are refined, that is, they are deactivated and their children are activated. This is encoded in the subdivision weights as follows: $\{w_3^{(0)}, w_4^{(0)}, w_5^{(0)}, w_6^{(0)}\}$ are set to zero and the entries of $\mathbf{w}^{(1)}$ are computed as in given in (29). Entries $6-10$ become equal to 1 and are allowed to be refined. Hence, in a following step, we could for instance refine functions $7-10$ leading to the three-level hierarchical spline space depicted in Figure 9.

$$\mathbf{w}^{(0)} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \xrightarrow{\text{deactivate}} \mathbf{w}^{(0)} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \xrightarrow{\text{activate}} \mathbf{w}^{(1)} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1/4 \\ 3/4 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \xrightarrow{\text{deactivate}} \mathbf{w}^{(1)} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1/4 \\ 3/4 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \xrightarrow{\text{activate}} \mathbf{w}^{(2)} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ \vdots \\ \vdots \\ 1/2 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}.$$

To summarize, sparse representation of the vectors $\mathbf{w}^{(\ell)}$, $\ell = 0, 1, \ldots \ell_{\max}$, encode the sets of active and inactive functions, $\eta_A^{(\ell)}, \eta_D^{(\ell)}$, and the non-zero values, $w_i^{(\ell)}$, can be used effectively in order to:

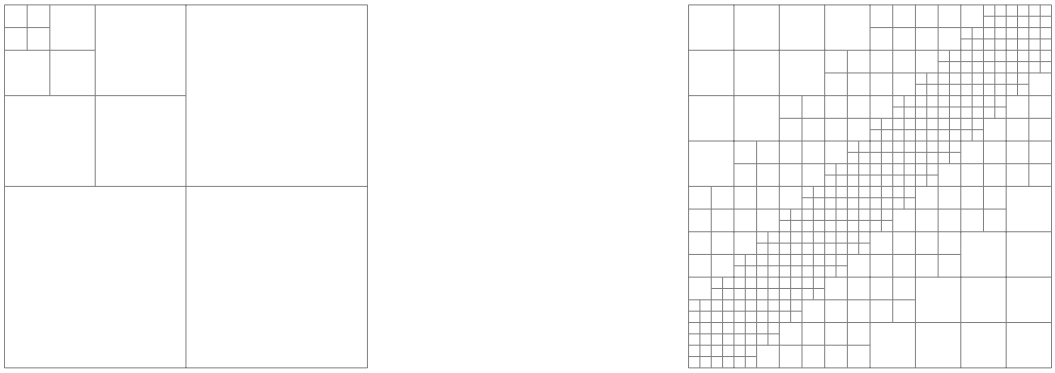
1. balance the refinement to minimize the overlap of coarse and fine scale functions thereby minimizing the detrimental effect on conditioning of the multilevel spline basis.

2. maintain a partition of unity,

$$\sum_{\ell=0}^{\ell_{\max}} \sum_{i \in \eta_A^{(\ell)}} w_i^{(\ell)} \cdot B_i^{(\ell)} = 1 \tag{32}$$

3. extend hierarchical B-splines to the truncated basis.

We shall elaborate on this last point in the next section. Finally, a drawback of the proposed balancing is that the refinement footprint could be large, especially in the setting of maximal continuity spline spaces. Two examples of the refinement footprint obtained are displayed in Figure 10.



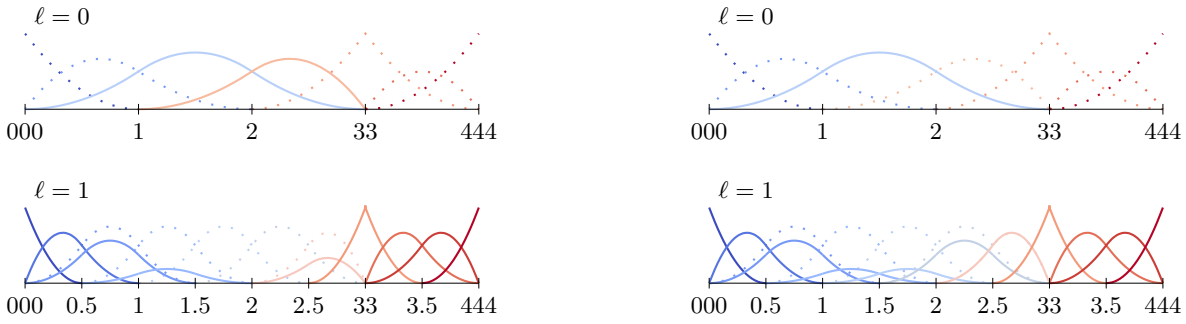
(a) Refinement of a corner

(b) Refinement along the diagonal

Figure 10: Illustration of the refinement footprint of the balancing scheme based on equation (31). Both cases show local refinement of maximal continuity spline spaces defined by open knot vectors and degree $p = 2$.

3.2.4. Maintaining global linear independence

Linear independence is lost when an active basis function on level ℓ can be represented as a linear combination of active functions on levels $\ell+1,\ldots,\ell_{\max}$. Although local refinement is driven by deactivating a parent function and activating only its children, linear dependencies can still occur, as is illustrated in Figure 11. To avoid this from happening the linear dependency needs to be explicitly removed. This is done as follows. First a separate check for linear independence is performed after all proposed refinements have been applied. This is done by checking for each active parent function on level $\ell < \ell_{\max}$ if all children are active as well. If this is the case, then the parent function is deactivated, thereby removing the linear dependency. In practice, we refine a parent function if fewer than two children are deactivated. This does not increase the number of active basis functions and improves the balancing of the levels.



(a) Linear independent basis

(b) Linear dependent basis

Figure 11: The two-level spline basis in (a) is linear independent. Refining function 3 on level 0 leads to a linear dependency (b), since function 2 on level 0 can be represented in terms of functions 2,3,4 and 5 on level 1.

3.2.5. Matrix formation and assembly

In the matrix formation and assembly one loops over each Beziér element of a particular level and adds contributions to the global system matrix. In parts of Ω in which only one level is active there is no difference with standard finite element formation and assembly in isogeometric analysis. However, local hierarchical refinement may lead to two or more levels that overlap. In these zones special treatment of the element integrals is necessary. In [7] a clever way is introduced using the subdivision matrices. In overlapping regions $\Omega_\ell \cap \Omega_k \neq \emptyset$ $(\ell \neq k)$ element integrals are computed using tensor product splines corresponding to the finest level. The contributions of functions on the coarse levels are then computed using the two-scale relations. For more details we refer to [7].

3.3. Extension to the truncated basis

The truncated basis was introduced in [14] to improve the numerical properties of the multilevel spline space. Truncated basis functions are defined as,

$$\text{trunc } B_j^{(\ell)}(\xi) = \sum_{i \in \eta_D^{(\ell+1)}} \mathbf{S}_{ij}^{(\ell)} B_i^{(\ell+1)}(\xi). \tag{33}$$

It is shown in [14] that the collection of active functions forms a partition of unity, is linearly independent and spans the multilevel spline space $\mathcal{MS}(\Omega)$. The main improvement of the truncated basis, with respect to hierarchical B-splines, is that the overlap of consecutive levels is decreased resulting in improved stability properties and conditioning. We refer the reader to [14] for more information. Figure 12 illustrates the truncated version of the basis shown in Figure 9. Note that the size of Ω^0 is reduced compared to the non-truncated case, while the size of Ω^1 remains the same due to the double knot at 3. Furthermore, it is pointed out that the basis functions on level 2 are not affected by the truncation procedure.

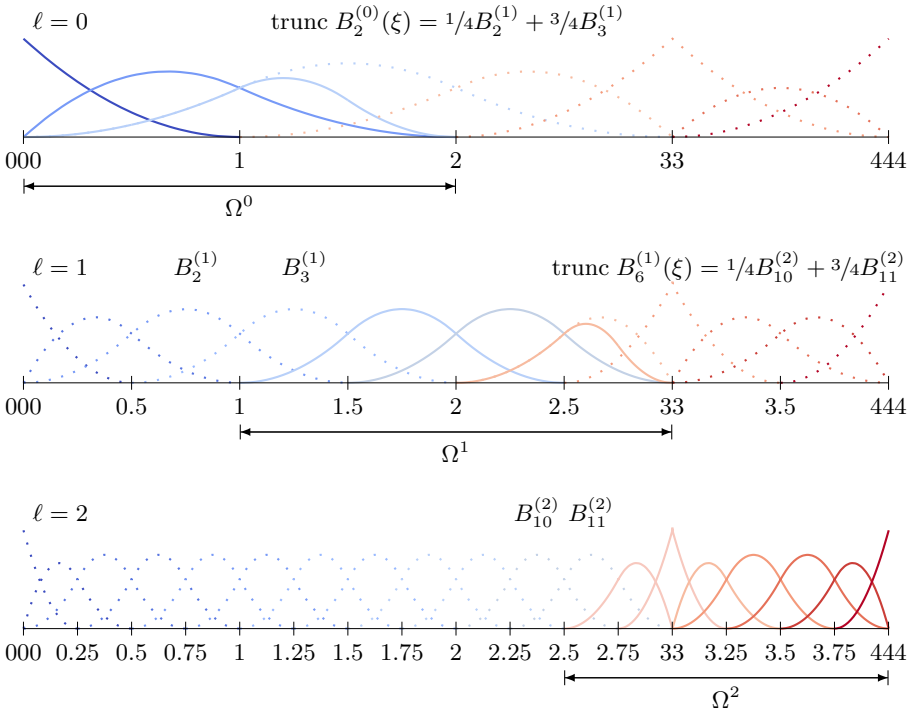
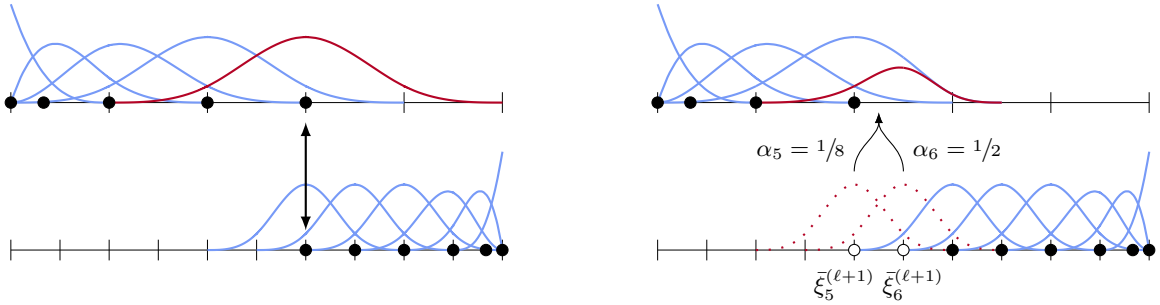


Figure 12: Three-level THB-spline basis. Active functions are represented by solid lines and basis functions involved in the truncation are labeled.

Using the proposed data structure, truncated B-splines can be easily identified. Suppose an element of a THB-basis shall be evaluated, where the element's level $\ell + 1$ is the finest level with active basis functions at the evaluation point ξ. In general, the non-zero B-splines of the element are $B_i^{(\ell+1)}$ with $i = s^{(\ell+1)} - p, \ldots, s^{(\ell+1)}$, based on the knot span index $s^{(\ell+1)}$ of the element. This element contains truncated B-splines on the previous level only if one of its $B_i^{(\ell+1)}$ is *inactive*, that is, the related subdivision weight $w_i^{(\ell+1)}$ is equal to zero. In order to determine if more than two levels overlap, the entries of the subdivision matrix $\mathbf{S}_{jk}^{(\ell-1)}$ are examined with $\ell - 1 > 0$, $j \in \{s^{(\ell)} - p, \ldots, s^{(\ell)}\} \cap \eta_D^{(\ell)}$, and $k = s^{(\ell-1)} - p, \ldots, s^{(\ell-1)}$. Here, $s^{(\ell)}$ and $s^{(\ell-1)}$ refer to the knot spans on level ℓ and $\ell - 1$ that contain ξ, respectively. If these entries do not form a null matrix, there is an additional overlapping level $\ell - 1$ and related truncated basis functions. This check is repeated for prior levels until a null matrix is obtained or the initial level 0 is reached.

3.4. Collocation of THB-splines

The purpose of this section is to maintain the applicability of the extended B-spline stabilization to isogeometric collocation schemes [2, 31] when hierarchical refinement is applied. It has been demonstrated that collocation at the Greville abscissae (6) is in general a good option in various contexts such as interpolation problems [9], isogeometric BEM formulations [25, 28], and isogeometric FEM collocation [4, 5]. Nevertheless, it is worth noting that FEM simulations based on the Greville abscissae perform suboptimally for odd degree. Recently, several investigations have started to address this issue for conventional B-splines, see e.g., [1, 15, 33]. These schemes have not been applied to hierarchical B-splines yet and this topic is also beyond the scope of this paper. As a matter of fact, even the application of the Greville absissae to a hierarchical basis requires special considerations since collocation points of different levels may coincide as illustrated in Figure 13(a). To overcome this problem, adaptive isogeometric collocation has been proposed



(a) Coinciding Greville abscissae

(b) Adaptive isogeometric collocation

Figure 13: Greville abscissae in the context of hierarchical B-splines: (a) potential problems at the transition from one level to another and (b) substitution of the troublesome point by a set of fine-scale abscissae $\bar{\xi}_j^{(\ell+1)}$ weighted by corresponding coefficients α_j of the related two-scale relation. In (b), the new collocation points are marked by white circles and the corresponding weighted fine-scale B-splines are illustrated by dashed lines. The accumulation of their values is indicted by arrows.

[38]: collocate at the Greville abscissae if they do not lie within the support of an active basis function of a finer level; apply a weighted collocation scheme otherwise. In the latter case, the B-spline of interest, $B_i^{(\ell)}$, is associated with a set of collocation points, derived from the Greville abscissae $\bar{\xi}_j^{(\ell+1)}$ of its children on the subsequent level. Regarding THB-splines, these $\bar{\xi}_j^{(\ell+1)}$ are the anchors of the basis functions that specify the truncated function on the coarser level. Figure 13(b) shows the $\bar{\xi}_j^{(\ell+1)}$ of a univariate cubic THB-spline. Note that a $\bar{\xi}^{(\ell+1)}$ may coincide with anchors of the coarser level, but there will always be at least one independent point in the overall set of collocation points associated to $B_i^{(\ell)}$. The results of the distinct $\bar{\xi}_j^{(\ell+1)}$ are weighted by the corresponding coefficients α_j – the same parameters used for the additive definition (33) of the truncated basis function, i.e., $\alpha_j = \mathbf{S}_{ji}^{(\ell)}$ with $j \in \eta_D^{(\ell+1)}$. The accumulation of these weighted values yields the final contribution of $B_i^{(\ell)}$. This concept is very general, but it also increases the number of collocations points.

In this work, we consider alternative collocation approaches for THB-splines that introduce only as many collocation points as basis functions. In particular, collocation at (i) the maximal value of truncated B-splines or (ii) the Demko abscissae is investigated. The former approach follows the adaptive collocation idea in that the Greville abscissae are used if possible, but instead of weighted collocation, the maximal value of a troublesome basis function on the coarse level determines the position of its collocation point. Thus, a single point is associated to the spline rather than a set of points. This can be employed since maximal values of THB-splines do not coincide due to the truncation procedure. However, a THB-spline may have multiple peaks in the bivariate case. In such a situation, one of the possible locations is chosen. The other collocation approach is inspired by the abscissae introduced by Demko [11]. In general, these abscissae seek

to minimize the max-norm of the inverse spline collocation matrix given by

$$\|\mathbf{A}^{-1}\|_\infty = \max_{k=0,\dots,I-1} \sum_{l=0}^{I-1} |\mathbf{A}^{-1}(k,l)| \qquad \text{with} \qquad \mathbf{A}(i,j) = B_{j,p}(\bar{\xi}_i), \qquad i,j = 0,\dots,I-1 \tag{34}$$

where I is the total number of basis functions. The measure $\|\mathbf{A}^{-1}\|_\infty$ bounds the approximation error of a spline interpolation problem [9]. Its minimization is achieved using the Demko abscissae $\bar{\xi}^d$, which are determined by extrema of Chebyshev splines. These splines oscillate constantly between their extrema which are alternately 1 and -1. To be precise, a function f of degree p given by $f(\bar{\xi}_i^d) = (-1)^{I-i}$ is represented by the B-splines $B_{i,p}$ of $\Xi = \{\xi_0, \dots, \xi_{I+p}\}$ such that it has max-norm 1 on $[\xi_0, \xi_{I+p}]$. The determination of the extrema involves a Newton-Raphson scheme which optimizes the entire set of collocation points. The Greville abscissae provide a good initial guess for the procedure. For multivariate basis functions, the optimization is applied to each parametric direction independently and the tensor product of the univariate results provides the final abscissae. Here, we are going to apply Demko's approach to THB-splines. The initial guess for basis functions within the transition zone of two levels is defined by points equally distributed between the first Greville abscissa on the finer level and the last valid Greville abscissa on the coarse one.

3.4.1. Assessment of the collocation schemes

In order to compare the adaptive collocation approach, the maximal value collocation, and the application of the Demko abscissae to THB-splines, the following examples are investigated: firstly, the univariate case is considered. The initial level 0 of the THB-basis is defined by an open knot vector with uniform knot spacing, i.e., $\Xi^{(0)} = \{0, \dots, 0, 1, \dots, 14, 15, \dots, 15\}$, for various polynomial degrees, $p = \{3, \dots, 8\}$. The second level is introduced by refining all basis functions, $B_{i,p}^{(0)}$ with $i < I^{(0)}/2$, where $I^{(0)}$ is the total number of B-splines on level 0. Secondly, the univariate example is extended to the bivariate setting. The initial level is determined by the tensor product $\Xi^{(0)} \times \Xi^{(0)}$ and hierarchical refinement is performed for all basis functions $B_{\mathbf{i},\mathbf{p}}^{(0)}$ where the index in the ξ_1-direction is greater or equal to the index in the ξ_2-direction, i.e., $i_1 \geq i_2$. Figure 14 illustrates the resulting parameter space for $p = 4$ and the corresponding collocation points due to the different collocation strategies. It is apparent that the number of points is reduced if the Demko abscissae or the maximal value collocation is used. Furthermore, it is emphasized that the Demko procedure is only applied to anchors of basis functions associated to the initial level 0. We have studied

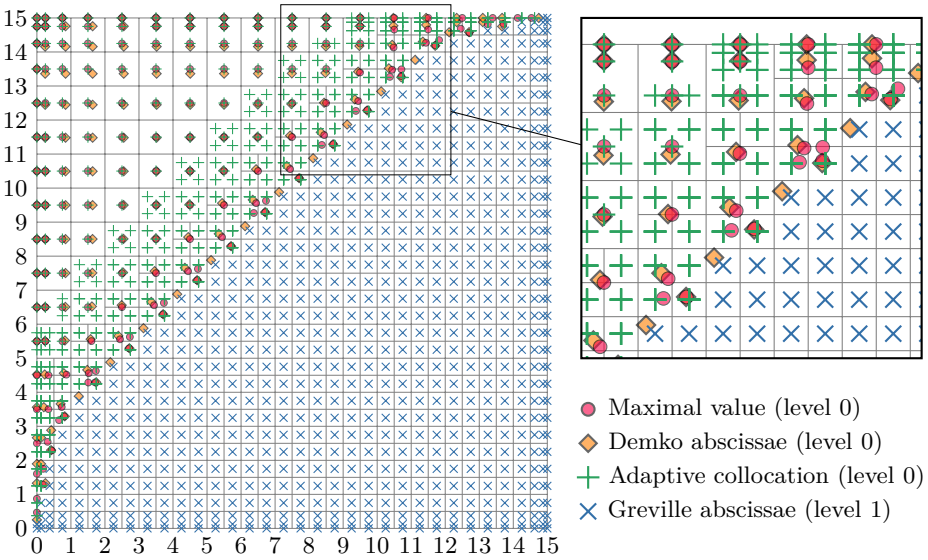


Figure 14: Collocation points of the bivariate example for degree $p = 4$.

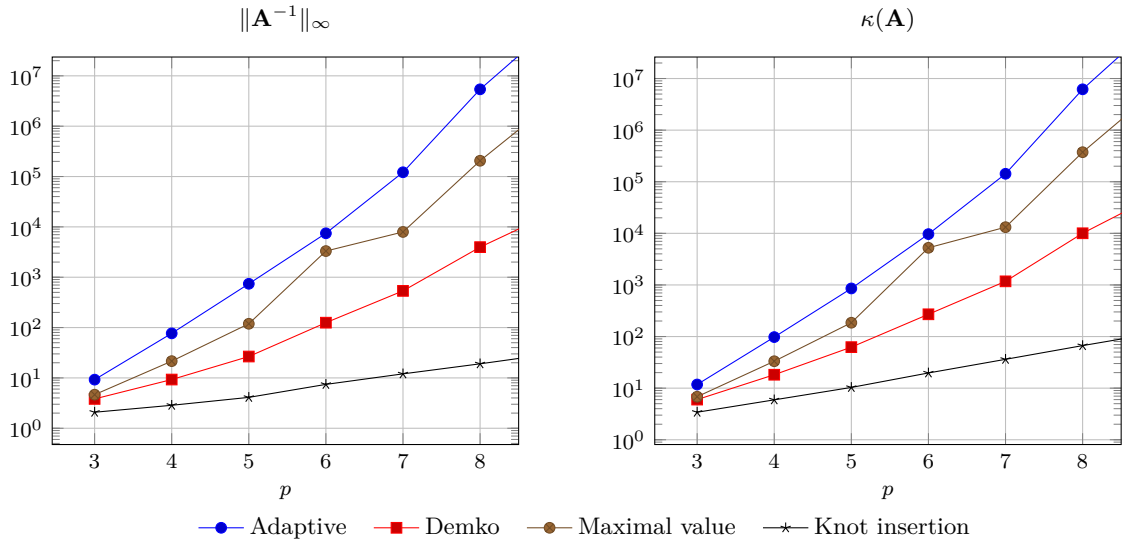


Figure 15: Univariate THB-spline example: max-norm $\|\mathbf{A}^{-1}\|_\infty$ and condition number $\kappa(\mathbf{A})$ of the spline collocation matrix $\mathbf{A}$ obtained by adaptive isogeometric collocation (Adaptive), the Demko abscissae (Demko), or maximal value collocation (Maximal value). In addition, the results of an equivalent regular B-spline basis refined by knot insertion (Knot insertion) are provided for comparison.

the effect of modifying the fine scale anchors as well, but hardly any changes have been observed for the example considered. Hence, these results are omitted.

For all examples, the corresponding spline collocation matrix $\mathbf{A}$ is set up using each collocation scheme. This matrix represents the system matrix of an interpolation problem. The quality of the corresponding approximation is estimated by the max-norm $\|\mathbf{A}^{-1}\|_\infty$ defined by equation (34). Further, the condition number of the matrix

$$\kappa(\mathbf{A}) = \|\mathbf{A}\|_2 \cdot \|\mathbf{A}^{-1}\|_2 \tag{35}$$

is computed with the spectral norm

$$\|\mathbf{M}\|_2 = \sqrt{\lambda_{max}\left(\mathbf{M}^\intercal \mathbf{M}\right)} \tag{36}$$

where λ_{max} denotes the largest eigenvalue. In the univariate case, an equivalent conventional B-spline basis refined by knot insertion is utilized as reference solution; the related matrix $\mathbf{A}$ is set up using the Greville abscissae. The one-dimensional results are reported in Figure 15 and the results of the bivariate example are summarized in Figure 16.

It is evident that the adaptive collocation scheme performs suboptimally with respect to $\|\mathbf{A}^{-1}\|_\infty$ and $\kappa(\mathbf{A})$ in all cases. The application of the Demko abscissae leads to a significant improvement that is more pronounced with increasing degree p. At the same time, all univariate THB-spline results show a clear shortfall compared to the conventional B-spline basis. This indicates that there is still potential for an enhanced treatment along the transition of two levels of a THB-basis. Nevertheless, the Demko approach and the maximal value collocation seem to be good choices when it comes to collocating THB-splines. The main issue of the Demko scheme is that it has some limitations regarding its practicality. As was pointed out previously, the related minimization process is always applied to a set of basis functions and usually the Demko abscissae of bivariate basis functions are computed by the tensor product of univariate results. The present bivariate example allows the application of this tensor product construction in a straightforward manner, but the proper choice of B-splines for the minimization process is much more complicated for more general refinement cases. With this in mind, the maximal value collocation seems to be a useful compromise: (i) the performance related to $\|\mathbf{A}^{-1}\|_\infty$ and $\kappa(\mathbf{A})$ is in between the two other schemes, (ii) it introduces

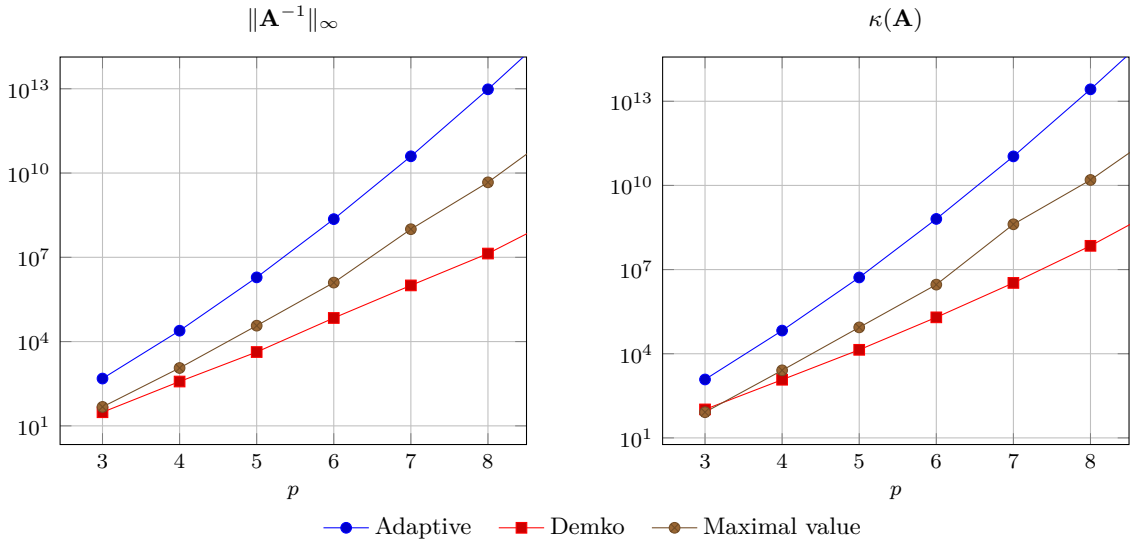


Figure 16: Bivariate THB-spline example: max-norm $\|\mathbf{A}^{-1}\|_\infty$ and condition number $\kappa(\mathbf{A})$ of the spline collocation matrix $\mathbf{A}$ obtained by adaptive isogeometric collocation (Adaptive), the Demko abscissae (Demko), or maximal value collocation (Maximal value).

merely one collocation point per basis function just like the Demko approach, and (iii) the construction of these points involves only the affected THB-splines similar to the adaptive concept. Due to the local nature, the maximal value collocation can be easily applied to arbitrary complex situations. In addition, the determination of the location of the maximal value of a THB-spline is a relatively small overhead that has to be applied solely for the troublesome basis functions at the overlap of two hierarchical levels. Based on this discussion, the maximal value collocation is preferred and used in all subsequent examples.

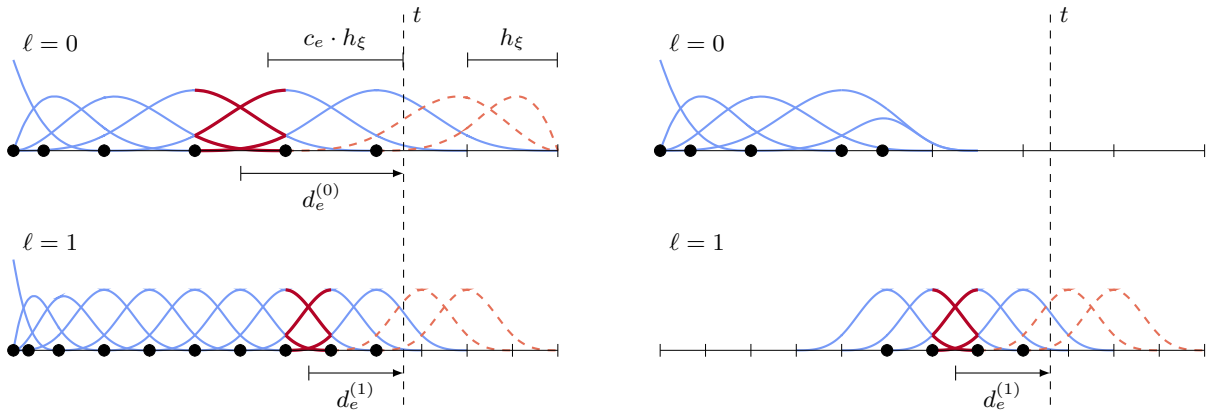
The assessment of the present approaches focuses on approximation properties in the context of interpolation problems. In our case, these factors are sufficient, because collocation points which are good for interpolation problems have been shown to be well suited for collocated isogeometric BEM simulations [25, 28]. At this point, it is emphasized again that choosing proper points for isogeometric FEM collocation requires further attention, especially for odd degree.

4. Extended THB-splines

The complementary features of extended B-splines and THB-splines are combined in order to obtain a more powerful stabilization technique for isogeometric analysis of trimmed geometries. While extended B-splines resolve the stability issue introduced by degenerate B-splines of a trimmed parameter space, truncated hierarchical refinement provides control over the extrapolation length d_e, thereby making the stabilization independent from the degree of the basis. We propose to unite these two components based on an admissibility criterion for the extrapolation length specified by

$$d_e < c_e \cdot h_\xi \tag{37}$$

where h_ξ refers to the average knot span size of the initial level 0 of the basis and c_e is a user-defined constant. Refinement levels are added until the related extrapolation length $d_e^{(\ell)}$ complies with condition (37). Figure 17(a) illustrates the identification of the required level. It should be noted that $d_e^{(\ell)}$ is checked for each level separately. Once the number of hierarchical levels is determined, the THB-basis is initiated by activating those basis functions of the finest level that are involved in the stabilization. The rest of the space is spanned by basis functions of the previous levels as indicated in Figure 17(b). From an algorithmic



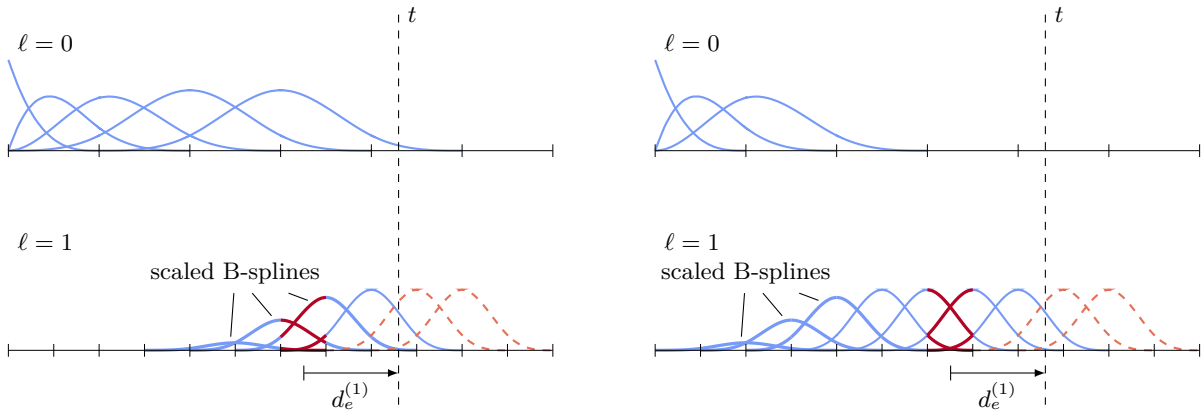
(a) Detection of the refinement level required

(b) Resulting THB-splines

Figure 17: Setting up of an extended THB-spline basis: (a) determination of the refinement level ℓ required to fulfill the criterion $d_e < c_e \cdot h_\xi$, and (b) activation of the basis function for the extrapolation on the finest level.

point of view the THB-basis is set up as follows: the degenerate and stable B-splines that contribute to the extended B-spline construction have already been revealed during the determination of the required refinement level $\ell_{\max}$. These basis functions are sorted such that the degenerate ones come first and the stable ones follow, ordered by the distance of their anchors to the trimming curve. To be precise, the B-spline that is furthest away from the trimmed curve is listed last. Next, the sorted basis functions are successively activated. Activating a B-spline $B_i^{(\ell_{\max})}$ is performed by refining *all* of its parents on the previous level $\ell_{\max} - 1$. Following the refinement rule (31), parents are only refineable if their subdivision weight is equal to 1. Thus, basis functions of the levels $\ell < (\ell_{\max} - 1)$ may have to be refined preliminarily. This leads to a recursive refinement process which evolves through the hierarchical structure. Since all parents of $B_i^{(\ell_{\max})}$ are refined, the function itself becomes refineable as well, i.e., $w_i^{(\ell_{\max})} = 1$. In addition, B-splines $B_j^{(\ell_{\max})}$ adjacent to $B_i^{(\ell_{\max})}$ are activated indirectly, if they share a parent function. These B-splines are active, but not refineable, i.e., $0 < w_j^{(\ell_{\max})} < 1$. The overall refinement process stops once all basis functions of the sorted set are either directly or indirectly activated. Due to the initial sorting of the functions, the evolution of the THB-basis begins outside of $\mathcal{A}^{\mathrm{v}}$ and gradually propagates into it. Hence, the refinement ends before it affects basis functions on level $\ell_{\max}$ that are not related to the extended B-splines. In the following, the term *extended THB-splines* will be used to refer to the overall stabilization procedure described.

It is underscored that extended B-splines are restricted to the finest level of the hierarchical basis; all other levels consists of conventional THB-splines only. Thus, the actual stabilization, i.e., the determination of the extrapolation weights, remains unchanged and no additional effort is introduced. Moreover, no superfluous basis functions are included to the THB-basis since all stable B-splines of the finest level have at least one degenerate B-spline associated to them. These benefits – decoupling the local refinement from the stabilization procedure and minimizing the number of degrees of freedom – can be carried out at the same time because basis functions on the finest level of a THB-basis are standard B-splines and the truncation only alters B-splines on coarser levels. Suppose scaled hierarchical B-splines would be used for the local refinement, the scaling factors would either affect the extended polynomial segments, and therefore the computation of the extension matrix, or additional basis functions would have to be added to the finest level. These two cases are illustrated in Figure 18. A guideline on how to chose c_e is given later on during the discussion of the numerical results.



(a) Coupling of scaling factors with the stabilization

(b) Increasing degrees of freedom

Figure 18: Disadvantage of scaled hierarchical B-spline refinement regarding the stabilization: (a) either the scaling factors have to be taken into account during the construction of the extended B-splines, or (b) additional basis functions on the finest level have to be activated. These issues do not occur when THB-splines are used (see Figure 17(b)).

5. Numerical results

Extended THB-splines are applied to various numerical examples of scalar Laplace and linear elasticity problems. The analysis is performed by an isogeometric BEM formulation which utilizes the collocation scheme derived in Section 3.4. Using BEM allows us to draw our attention to the performance of the proposed combination of extended B-splines with local refinement by THB-splines since issues which would arise in a FEM context, e.g., the enforcement of essential boundary conditions, do not have to be addressed. The following subsection (5.1) gives a brief introduction to the isogeometric BEM formulation used. For detailed information on the related implementation, the interested reader is referred to [28]. The numerical experiments are presented and discussed in the subsequent parts of this section.

5.1. Description of the analysis setting

Suppose we have a Neumann problem in which the following *boundary value problem* is considered: find $u(\boldsymbol{x})$ such that

$$\begin{aligned} \mathcal{L}u(\boldsymbol{x}) &= 0 && \forall \boldsymbol{x} \in \Omega \\ \mathcal{T}u(\boldsymbol{x}) &= t(\boldsymbol{y}) = g_N(\boldsymbol{y}) && \forall \boldsymbol{y} \in \Gamma_N \end{aligned} \tag{38}$$

for the domain of interest Ω and the prescribed Neumann data g_N on its boundary $\Gamma_N \equiv \Gamma := \partial\Omega$. In case of scalar Laplace problems, the elliptic partial differential operators $\mathcal{L}$ is given by

$$\mathcal{L}u(\boldsymbol{x}) := -k\Delta u(\boldsymbol{x}) \tag{39}$$

where the coefficient k denotes the conductivity of the material and Δ is the Laplace operator. For elasticity, on the other hand, $\mathcal{L}$ is the Lamé-Navier operator, defined as

$$\mathcal{L}u(\boldsymbol{x}) := -\mu\Delta u(\boldsymbol{x}) - (\lambda + \mu)\,\mathrm{grad}\,\mathrm{div}\,u(\boldsymbol{x}) \tag{40}$$

with the Lamé constants λ and μ [24]. The conormal derivative or traction operator $\mathcal{T}$ of the Laplace operator transforms the potential $u(\boldsymbol{x})$ to the flux along the boundary $t(\boldsymbol{y})$ by

$$\mathcal{T}u(\boldsymbol{x}) = k\nabla u(\boldsymbol{y}) \cdot \mathbf{n}(\boldsymbol{y}) = t(\boldsymbol{y}) \tag{41}$$

where $\mathbf{n}$ denotes the outward normal vector of Γ. For the Lamé-Navier operator (40), it is defined by

$$\mathcal{T}u(\boldsymbol{x}) = \lambda\nabla \cdot u(\boldsymbol{y})\mathbf{n}(\boldsymbol{y}) + 2\mu\nabla u(\boldsymbol{y}) \cdot \mathbf{n}(\boldsymbol{y}) + \mu\,\mathbf{n}(\boldsymbol{y}) \times (\nabla \times u(\boldsymbol{y})) = t(\boldsymbol{y}) \tag{42}$$

which maps displacements $u(\boldsymbol{x})$ to surface tractions $t(\boldsymbol{y})$.

By using the *fundamental solution* U of the operator $\mathcal{L}$, the solution u of any point $\boldsymbol{x} \in \Omega$ can be calculated by the representation formula

$$u(\boldsymbol{x}) = \int_\Gamma \mathsf{U}(\boldsymbol{x},\boldsymbol{y})\, t(\boldsymbol{y})\, \mathrm{ds}_{\boldsymbol{y}} - \int_\Gamma \mathsf{T}(\boldsymbol{x},\boldsymbol{y})\, u(\boldsymbol{y})\, \mathrm{ds}_{\boldsymbol{y}} \qquad \forall \boldsymbol{x} \in \Omega,\ \forall \boldsymbol{y} \in \Gamma \tag{43}$$

once the entire boundary data $u(\boldsymbol{y})$ and $t(\boldsymbol{y})$ are known [42]. The kernel T is determined by the conormal derivative with respect to the $\boldsymbol{y}$ variable, i.e., $\mathsf{T}(\boldsymbol{x},\boldsymbol{y}) = \mathcal{T}_{\boldsymbol{y}}\mathsf{U}(\boldsymbol{x},\boldsymbol{y})$. The fundamental solutions U and T for potential and elasticity problems considered in this paper are well known and can be found in various textbooks, see e.g., [13, 36, 42]. In general, a fundamental solution $\mathsf{U}(\boldsymbol{x},\boldsymbol{y})$ provides the response at a *field point* $\boldsymbol{y}$ due to a unit point source applied at $\boldsymbol{x}$, which is called the *source point*.

The boundary trace $\Omega \ni \boldsymbol{x} \to \boldsymbol{y} \in \Gamma$ of the representation formula (43) results in the *boundary integral equation*

$$c(\boldsymbol{x})u(\boldsymbol{x}) = \int_\Gamma \mathsf{U}(\boldsymbol{x},\boldsymbol{y})\, t(\boldsymbol{y})\, \mathrm{ds}_{\boldsymbol{y}} - \int_\Gamma \mathsf{T}(\boldsymbol{x},\boldsymbol{y})\, u(\boldsymbol{y})\, \mathrm{ds}_{\boldsymbol{y}} \qquad \forall \boldsymbol{x},\boldsymbol{y} \in \Gamma \tag{44}$$

which provides the basis for the collocated BEM formulation used. Note that equation (44) consists merely of variables defined along Γ. The direction of the outward normal $\mathbf{n}$ determines the corresponding Ω by pointing into the void. The jump term $c(\boldsymbol{x})$ is 0.5 if Γ is smooth; it generally depends on the geometrical angle at $\boldsymbol{x}$ and in case of elasticity also on the Poisson's ratio [13, 27].

In order to determine the unknown data $u(\boldsymbol{y})$, the geometry and the boundary data are discretized by splines, and equation (44) is enforced at a set of *collocation points* $\boldsymbol{x}^c$. Their position is specified by the anchors of the basis functions that discretize the unknown field. In order to allow discontinuities between patches, collocation points at the boundary of a patch are slightly shifted into its domain $\mathcal{A}^{\mathrm{v}}$, see [30]. The resulting discretized boundary integral equation yields a *system of equations* for the Neumann problem

$$\mathbf{K}\tilde{\mathbf{u}} = \mathbf{V}\tilde{\mathbf{g}}_N \tag{45}$$

with vectors $\tilde{\mathbf{u}}$ and $\tilde{\mathbf{g}}_N$ of the unknown control variables of the Dirichlet field and the known Neumann coefficients, respectively. The matrix entries are determined by

$$\mathbf{V}\,(i,j) = \int_\Gamma \mathsf{U}(\boldsymbol{x}_i^c,\boldsymbol{y}) B_j^t(\boldsymbol{y})\, \mathrm{ds}_{\boldsymbol{y}} \qquad \text{and} \qquad \mathbf{K}\,(i,j) = c_{ij} + \int_\Gamma \mathsf{T}(\boldsymbol{x}_i^c,\boldsymbol{y}) B_j^u(\boldsymbol{y})\, \mathrm{ds}_{\boldsymbol{y}} \tag{46}$$

where B_j^t and B_j^u are the basis functions related to t and u. Control variables for the jump term c_{ij} are calculated such that

$$\sum_{j=0}^{n} c_{ij}\, B_j^u(\boldsymbol{x}_i^c) = c(\boldsymbol{x}_i^c). \tag{47}$$

The system matrix $\mathbf{K}$ is invertible for an exterior problem [36], but would be singular in the interior case. The direction of $\mathbf{n}$ is the essential component of a BEM discretization to distinguish between these different problem types.

If the domain of interest is represented by trimmed patches, collocation points are only defined for stable B-splines as described in Section 2.4. Furthermore, collocation points of a locally refined basis are determined as discussed in Section 3.4. In order to incorporate the *stabilization* by extended THB-splines, the left hand side matrix $\mathbf{K}$ and the right hand side matrix $\mathbf{V}$ of system of equations (45) are multiplied by corresponding extension matrices

$$\mathbf{K}\mathbf{E}_u\tilde{\mathbf{u}}_{st} = \mathbf{V}\mathbf{E}_t\tilde{\mathbf{g}}_N \tag{48}$$

$$\mathbf{K}_{st}\tilde{\mathbf{u}}_{st} = \mathbf{f} \tag{49}$$

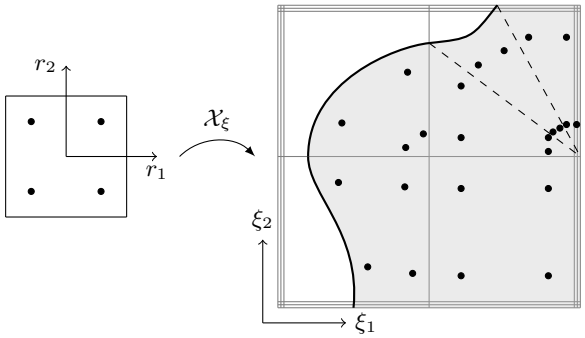


Figure 19: Partitioning of trimmed elements into integration regions. Dashed lines indicate if an element is represented by more than one integration region. Quadrature points are distributed based on a local ruled surface mapping $\mathcal{X}_r$.

with $\tilde{\mathbf{g}}_N \in \mathbb{R}^n, \tilde{\mathbf{u}}_{st} \in \mathbb{R}^n$, $\mathbf{f} \in \mathbb{R}^n$ and $\mathbf{K}_{st} \in \mathbb{R}^{n\times n}$. The subscripts of the extension matrices emphasize that they correspond to the basis of the Dirichlet and Neumann field, respectively.

The *integration* of trimmed geometries is performed according to [31]. Regarding the integration of trimmed surfaces, the main task is to find a mapping $\mathcal{X}_r(\boldsymbol{r})$ from the reference element $\boldsymbol{r} \in [-1,1]^{d-1}$ to the elements that are cut by the trimming curve. We define $\mathcal{X}_r$ by a planar and local ruled surface mapping, i.e., quadrature points are distributed by a linear interpolation between the portion of a trimming curve within an element and its opposing interior edge of the element. Figure 19 shows an example of the considered trimming cases and the associated distribution of quadrature points due to $\mathcal{X}_r$. Note that the opposing edge may collapse to a point and furthermore, a trimmed element may be represented by a set of integration regions. For each collocation point $\boldsymbol{x}^c$ the integral is performed over the whole boundary, i.e., $\forall \boldsymbol{y} \in \Gamma$. When an integration region contains the current $\boldsymbol{x}^c$, regularization of the integrals is required since the fundamental solutions become singular as $\boldsymbol{y}$ approaches $\boldsymbol{x}^c$. Details on the regularization techniques employed are given in [30]. It is emphasized that neither the stabilization by extended B-splines nor the utilization of THB-splines introduce any additional considerations for the integration procedure.

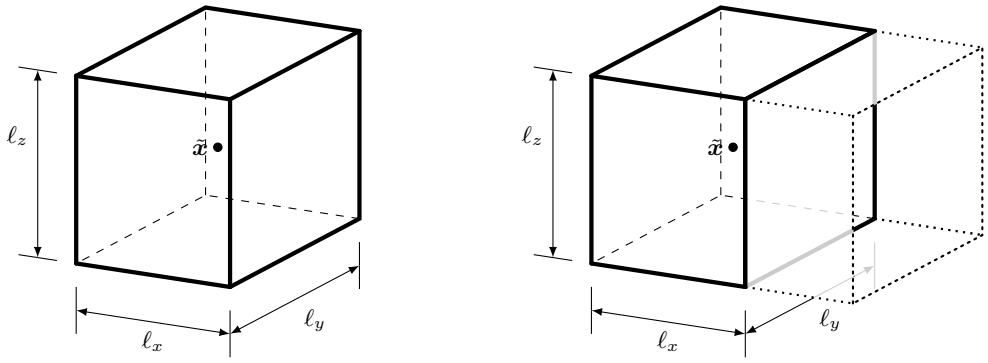
BEM has two properties that are very beneficial in our context: (i) BEM formulations allow discontinuities between adjacent elements, and (ii) *all* boundary conditions are incorporated into the system of equations in an integral sense. Consequently, we do not have to include additional techniques for enforcing essential boundary conditions or coupling of trimmed patches, which may alter the extended THB-basis we want to examine. Furthermore, approximation errors along trimming curves become more apparent due to the discontinuous fields.

5.2. Trimmed cube

The first numerical example compares simulations of trimmed objects stabilized by extended THB-splines with isogeometric BEM analyses of regular geometries. Exterior Neumann problems for the Laplace equation and elasticity are considered. In particular, a unit cube ($\ell_x = \ell_y = \ell_z = 1.0$) specifies the boundary Γ of the domain of interest $\Omega = \mathbb{R}^3 \setminus \Omega^-$ where Ω^- refers to the void. Two different geometry models are used to represent Γ. They are depicted in Figure 20: (a) six regular patches with matching parameterization provide the basis of the reference solutions, and (b) a model with two regular and four trimmed sides is used to investigate the performance of the proposed stabilization procedure. The misfit of the trimmed patches is defined by $\ell_x = 1.70$. The prescribed Neumann data along Γ are given by

$$t(\boldsymbol{y}) \coloneqq \mathsf{T}(\tilde{\boldsymbol{x}}, \boldsymbol{y}) \qquad\qquad \boldsymbol{y} \in \Gamma,\, \tilde{\boldsymbol{x}} \in \Omega^- \tag{50}$$

where $\tilde{\boldsymbol{x}}$ is a source point located in the center of the cube. The exact solution of the primary variable $u(\boldsymbol{x})$ is defined by the fundamental solution $\mathsf{U}(\tilde{\boldsymbol{x}}, \boldsymbol{x})$. Thus, the relative approximation error of a simulation can



(a) Regular (b) Trimmed

Figure 20: The unit cube of the exterior Neumann problem defined by (a) regular patches and (b) trimmed patches.

be calculated by

$$\epsilon_{rel} = \frac{u_h(\boldsymbol{y}) - \mathsf{U}(\tilde{\boldsymbol{x}}, \boldsymbol{y})}{\mathsf{U}(\tilde{\boldsymbol{x}}, \boldsymbol{y})} \qquad \boldsymbol{y} \in \Gamma,\, \tilde{\boldsymbol{x}} \in \Omega^- \tag{51}$$

and measured with respect to the L_2-norm $\|\epsilon_{rel}\|_{L_2}$, where $u_h(\boldsymbol{y})$ is the numerical solution of the boundary data. Uniform knot refinement is used to improve $u_h(\boldsymbol{y})$; an additional knot insertion step is applied in the x-direction of the trimmed patches to get similar element sizes on all faces of the cube. The final THB-basis of the trimmed surfaces is determined by the admissibility criterion (37) with the knot span size h_ξ measured after the global refinement. The influence of this criterion is investigated by using different constants $c_e = \{p/2, 0.5, 10\}$. In the latter case, the allowed extrapolation length is very large and hence, no local refinement is performed. On the other hand, $c_e = 0.5$ is a very strict condition yielding a fine resolution along the trimmed edge. Finally, moderate local refinement is established by $c_e = p/2$. The study is repeated for various polynomial degrees $p = \{2, 3, 4\}$.

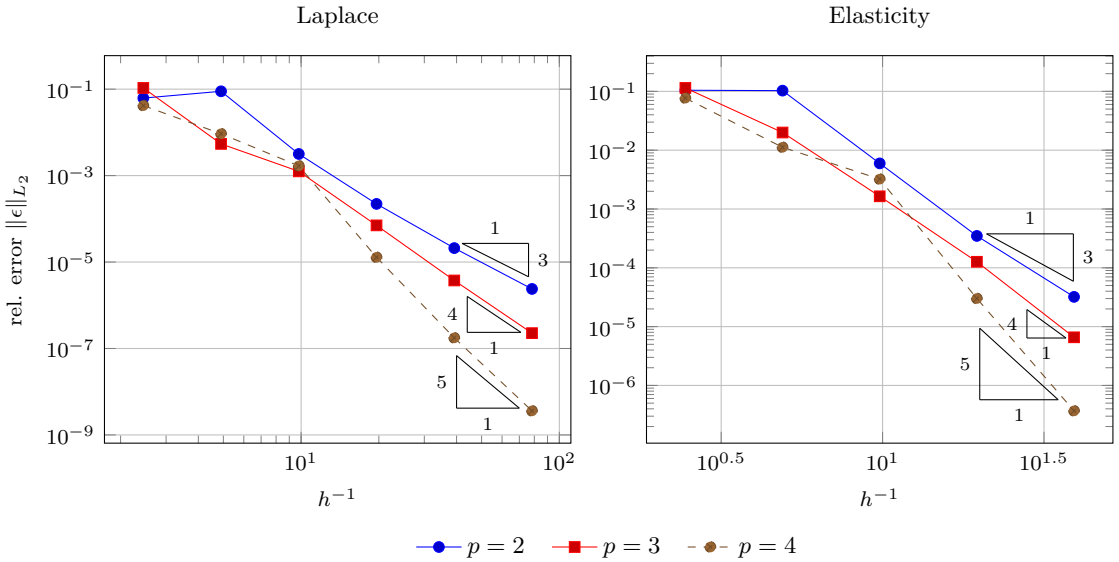


Figure 21: Convergence behavior of the regular model with respect to the mesh parameter h under uniform refinement. These graphs are used later on in Figure 22 as reference values (Regular).

First, the applied isogeometric BEM implementation as well as the reference solution of the regular B-spline model are verified. The corresponding convergence plots in Figure 21 are illustrated with respect to a mesh parameter

$$h = \left(\frac{A_{max}}{A_\Gamma} \right)^{1/2} \tag{52}$$

which relates the maximal area of all elements A_{max} to the complete surface area of the boundary A_Γ. As pointed out by Sauter and Schwab [36], there are still open questions regarding convergence proofs of collocation methods in the BEM context. However, the rate of convergence of a direct formulation – as used in our study – is expected to be $\mathcal{O}\left(h^{-p-1}\right)$ [3]. The present numerical experiments comply with these rates.

Figure 22 provides an overview of all results obtained. For a better comparison, the convergence is plotted versus the number of degrees of freedom. In addition, the distribution of the relative error of the Laplace problem discretized by $p = 3$ and two knot insertion steps is shown for the simulation based on extended B-splines without local refinement ($c_e = 10$) and the proposed extended THB-spline approach ($c_e = p/2$) in Figure 23.

Earlier studies presented in [31] showed that conventional extended B-splines ($c_e = 10$) are able to achieve the same accuracy as regular B-spline discretizations for linear splines. In this work, we sought to derive a methodology that establishes this behavior for higher degree as well. As can be seen in Figure 22, the graphs related to $c_e = 10$ are in general in good agreement with the reference solution (Regular). Yet, they contain certain outliers where the error is significantly larger. Figure 23(a) illustrates the error distribution of such an outlier. Apparently, the inaccuracy along the trimmed edges diminishes the overall quality of the numerical solution. This is a manifestation of the potentially negative effect of an uncontrolled extrapolation length as discussed in Section 2.6. The outliers do not correlate to a specific refinement step; they rather depend on the trimming situation and the corresponding set of degenerate B-splines. Nevertheless, the amplitude of the outliers tend to decrease with the fineness of a discretization since the contribution of the results along trimming curves becomes smaller with respect to the overall solution.

With the proposed extended THB-spline concept, the extrapolation length, and therefore the error along the trimming curve, can be controlled by performing local refinement. However, the results related to $c_e = 0.5$ clearly demonstrate that local refinement can become counterproductive if the admissibility condition is too strict. That is, the number of degrees of freedom concentrated in the vicinity of the trimmed edges is much higher than needed. Based on this study, a good balance between accuracy and number of degrees of freedom is obtained when the admissibility constant is set to $c_e = p/2$. Using this criterion, only a few refinement levels are introduced and in those cases where conventional extended B-splines already yield sufficient accuracy, no local refinement is applied at all. The results of $c_e = p/2$ are in an excellent agreement with the regular reference solution for all degrees. Still, there is an offset between the graphs related to $p = 4$. This deviation may occur due to the fact that the local refinement procedure introduces basis functions also in the direction parallel to the trimmed edge. These B-splines are indeed superfluous for the given trimming situation which is aligned with one parametric direction, and their number increases with the degree. Overall, it can be concluded that extended THB-splines make the stabilization of trimmed parameter spaces more robust and the related numerical results more accurate.

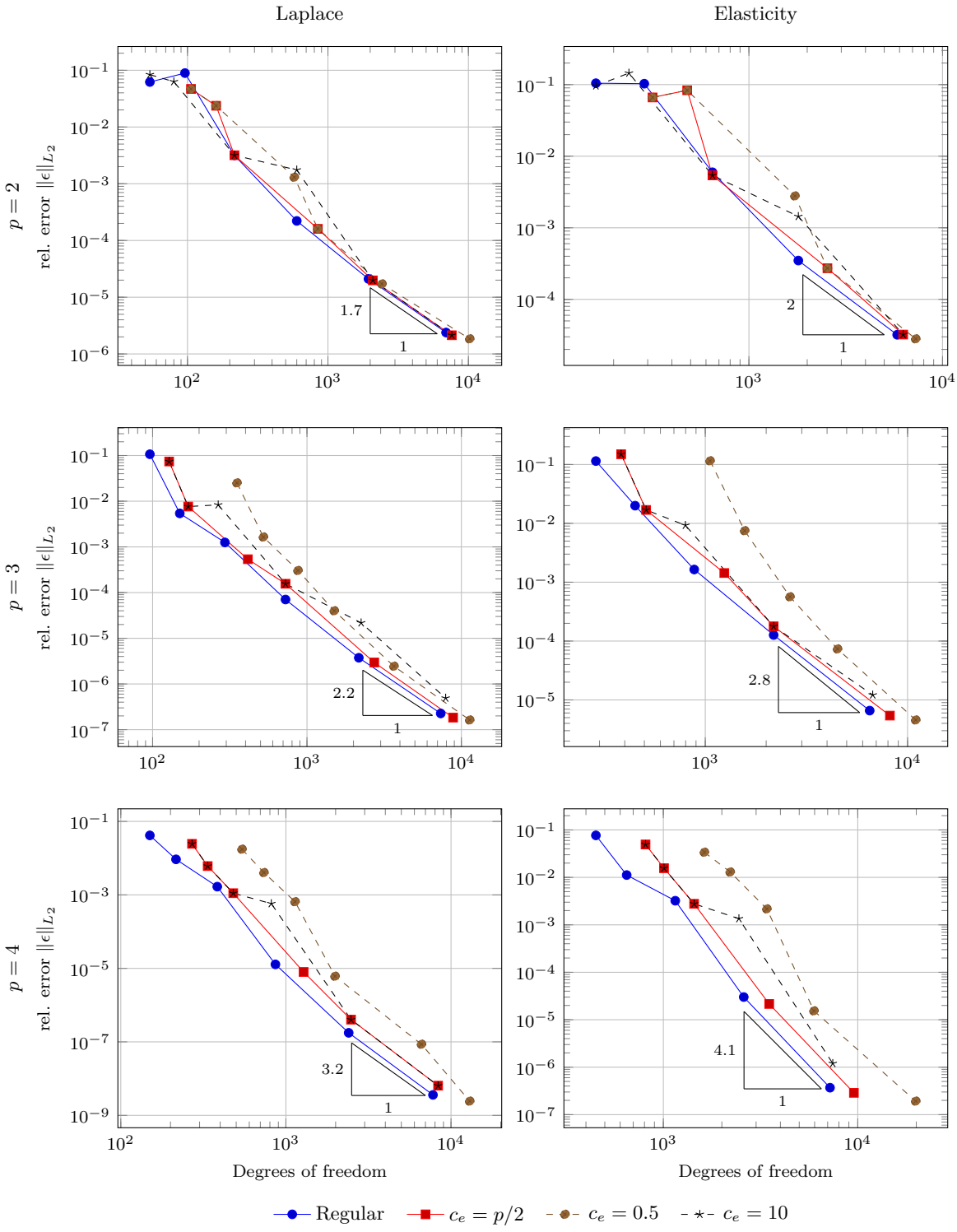
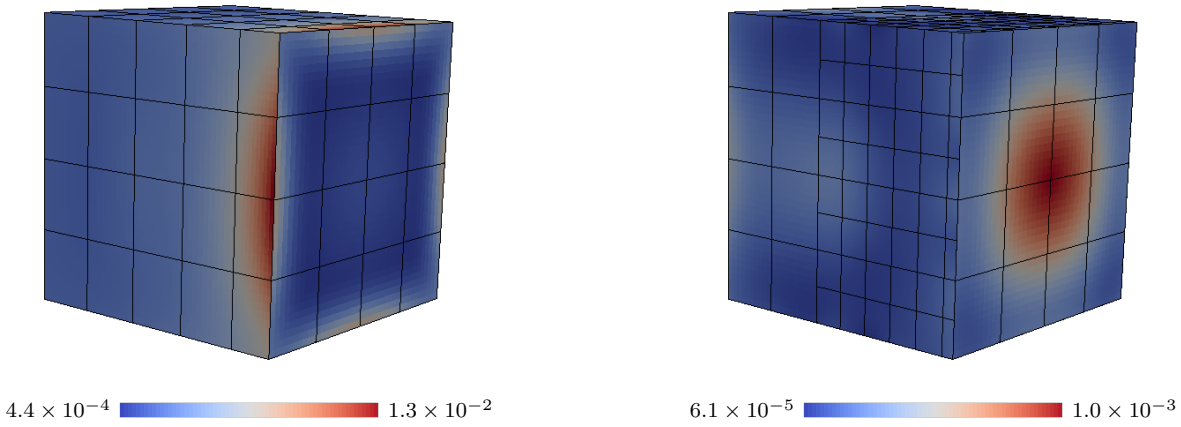
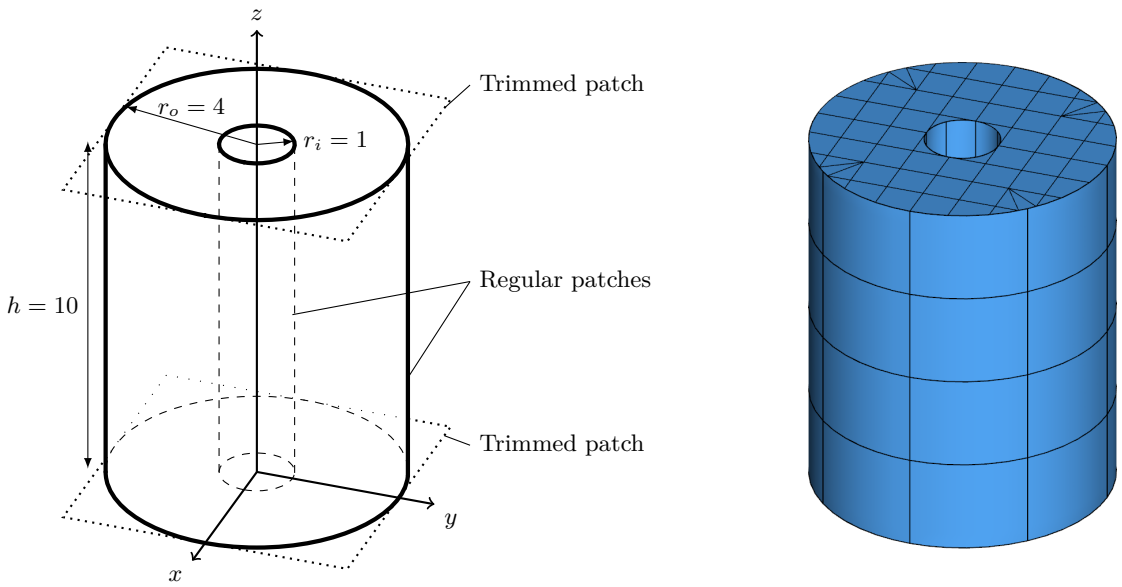


Figure 22: Relative L_2-error of the exterior Neumann problem of the cube example with respect to the number of degrees of freedom. The left column refers to the Laplace problem, whereas the Elasticity results are shown on the right. The rows correspond to the degrees $p = 2$, 3, and 4, respectively. Each diagram contains results related to discretizations using regular patches (Regular), and discretization based on trimmed ones which employ different refinement criteria: $c_e = \{p/2, 0.5, 10\}$.



(a) $c_e = 10$ (b) $c_e = p/2$

Figure 23: Distribution of the relative error along the surface of the trimmed unit cube for the Laplace problem discretized by (a) extended B-splines and (b) extended THB-splines of degree $p = 3$ after two refinement steps. Black lines indicate the resulting elements of the patches. Note that the right face is a regular surface and the edge in the front represents a trimming curve in both cases.



(a) Geometry specification (b) Initial discretization

Figure 24: Model of the hollow cylinder: (a) the interior and outer sides of the cylinder are described by regular surfaces, whereas trimmed patches represent its top and bottom; (b) the corresponding integration regions before local refinement is applied.

5.3. Hollow cylinder

The interior side of a solid hollow cylinder is subjected to an internal constant pressure p_c. The analytic solution of the displacement u_r in radial direction is given by

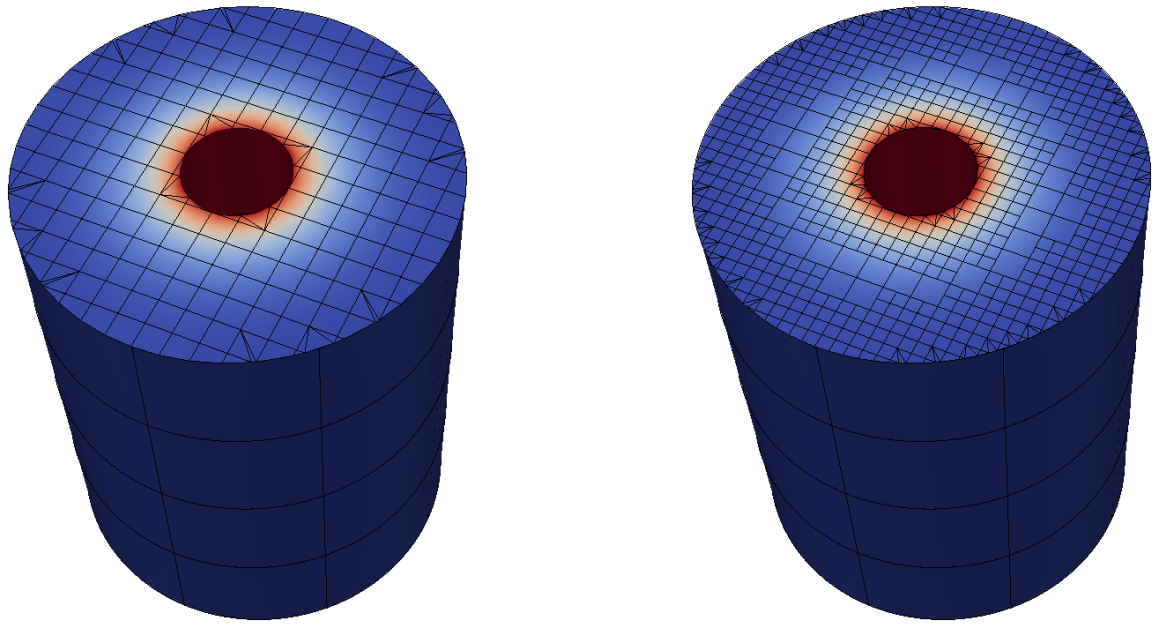
$$u_r(r) = \frac{p_c}{E} \frac{r_i^2}{r_o^2 - r_i^2} \left((1-\nu)\, r + (1+\nu)\, \frac{r_o^2}{r} \right) \tag{53}$$

with r_i and r_o denoting the inner and outer radius of the cylinder, respectively [39]. The material properties are defined by the Young's modulus E and the Poisson ratio ν. These parameters correspond to the Lamé constants of equation (40) by

$$\lambda = \frac{E\,\nu}{(1-2\,\nu)(1+\nu)} \qquad \text{and} \qquad \mu = \frac{E}{2\,(1+\nu)}. \tag{54}$$

Figure 24 shows the geometry of the problem and its discretization before local refinement is applied. Quadratic B-splines are used to define all patches of the model including the trimmed planar surfaces on the top and bottom of the cylinder. The material parameters are set to $\nu = 0$ and $E = 1 \times 10^5\,\text{MPa}$. The interior side of the cylinder is subjected to a constant pressure of $p_c = -100\,\text{MPa}$ which is applied to the model by prescribing the related displacement $u_r(r_i = 1)$ multiplied by the outward normal $\mathbf{n}$. The boundary tractions of all other surfaces are set to zero. Two simulations with different constants for the admissibility criterion (37) are performed, i.e., $c_e = \{1.0, 0.8\}$. The former uses the factor $p/2$ based on the discussion of the previous numerical example, and the latter introduces an additional refinement level. The number of degrees of freedom of these discretizations are 1800 and 4056, respectively.

The resulting integration regions and the displacements obtained are depicted in Figure 25. In addition, Figure 26 illustrates a comparison of the radial displacement along the x-axis of each model with the reference solution (53).



(a) $c_e = 1.0$ (b) $c_e = 0.8$

Figure 25: Displacement of the hollow cylinder discretized by quadratic extended THB-splines. The local refinement is determined by (a) the proposed factor $c_e = p/2$ and (b) a stricter admissibility criterion. The same parameter range is used for both figures.

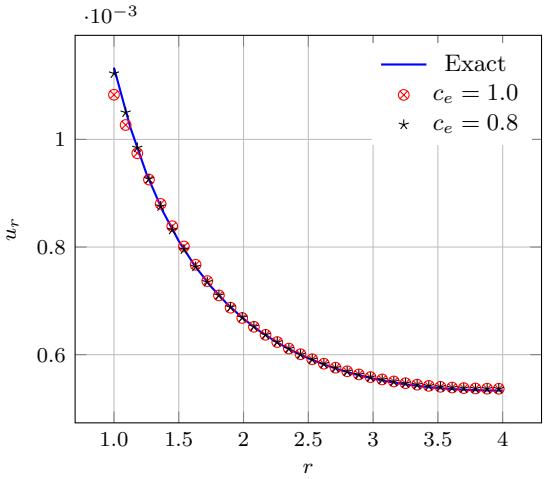


Figure 26: Displacement of the hollow cylinder along the x-axis.

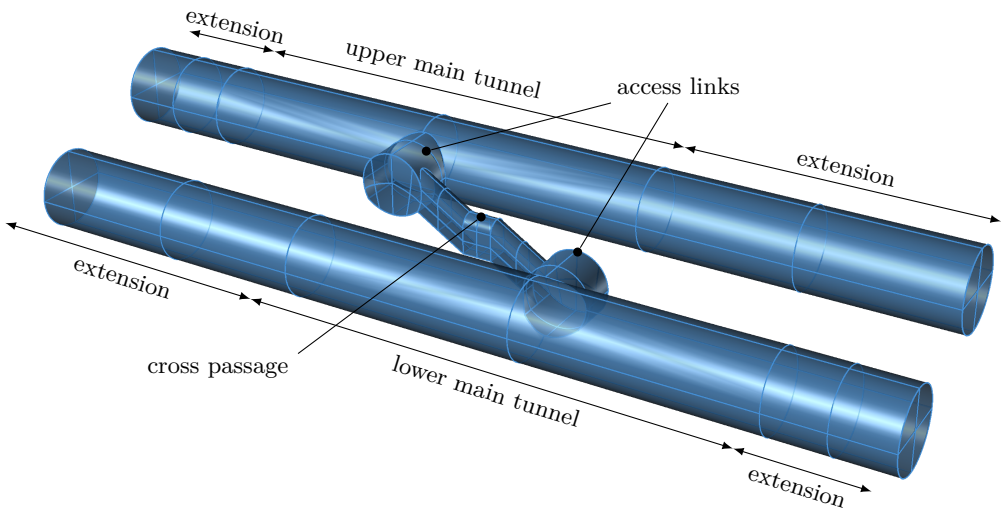
It is apparent that the axisymmetric behavior of the solution is resolved very well in both cases, despite the fact that the trimmed parameter spaces are not aligned with the radial direction. The comparison with the reference solution verifies the quality of the numerical solutions and the choice of $c_e = p/2$ for a good balance between accuracy and numerical effort. It is worth noting that the coarser model ($c_e = 1.0$) also addresses the event of two refinement fronts merging. Such situations may lead to linearly dependent basis functions if not properly dealt with.

5.4. Tunnel cross passage

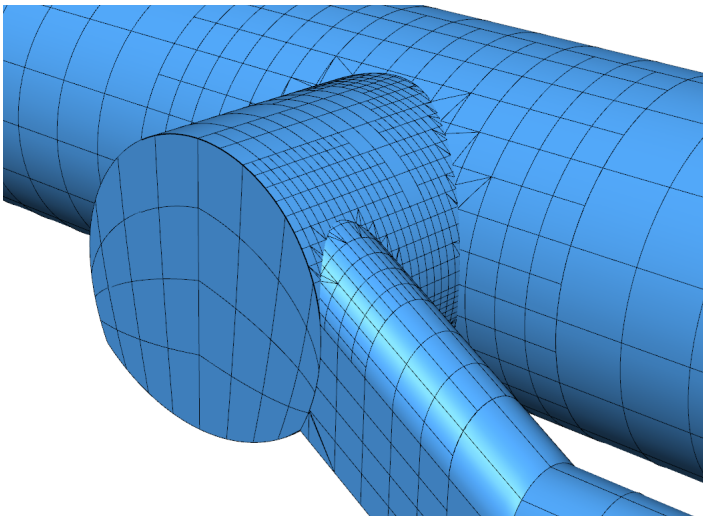
The excavation of a tunnel is simulated. Two parallel tunnel tubes connected by a cross passage describe the domain of interest. The corresponding CAD model of the boundary Γ is depicted in Figure 27(a). Each main tube consists of three sections: the main tunnel in the middle and extensions at both of its ends. The closed cross-sections of the main tubes locally increase the stiffness of the model and the extensions assure that this effect does not influence the results of the main tunnel. The connection of the main tubes is also divided into three parts: the cross passage and two access links. The intersections of these parts yield various complex trimming cases. In total, the geometry is represented by 9 trimmed patches and 6 regular ones (i.e., the extensions and the closed cross-sections of the access links).

The analysis model is obtained by knot insertion and the degree of linear patches is elevated to $p = 2$. In particular, the main tunnels and all parts of their connection are refined. Two stabilization strategies are employed: (i) extended THB-splines using $c_e = p/2$ and (ii) extended B-splines without local refinement. Figure 27(b) illustrates the resulting integration regions of the former approach at the connection of the upper tunnel tube with the cross passage. Regarding the degrees of freedom, the total number increases about 10 % from 11868 to 12993 due to the local refinement. The excavation process is simulated by subjecting the boundary Γ to excavation forces that are determined by the virgin stress field of the soil given by $\sigma_{xx} = \sigma_{yy} = 1.375\,\text{MPa}$ and $\sigma_{zz} = 2.75\,\text{MPa}$. To be precise, the applied tractions $t(\boldsymbol{y})\ \forall \boldsymbol{y} \in \Gamma$ are computed by multiplying the virgin stress tensor with the outward normal $\mathbf{n}$. The material properties are specified by the Poisson ratio $\nu = 0.2$ and Young's modulus $E = 313\,\text{MPa}$.

The displacements due to the excavation process of the extended THB-spline discretization is shown in Figure 28. Furthermore, contour-plots of the displacements obtained by the extended THB-spline scheme and the conventional one are compared in Figure 29. Recall that the isogeometric BEM formulation applied allows discontinuities between individual patches, hence jumps of the displacement field along an intersection can be seen as a kind of error indicator. The largest displacements occur at the intersections of the access links with the main tunnels. Hence, this region is of particular interest for this example. Note the abrupt change of the contour-plots along the intersection of the upper main tube depicted in Figure 29(a) and the



(a) CAD model



(b) Integration regions

Figure 27: Tunnel example with trimmed patches: (a) the original CAD model and (b) a detail showing the resulting integration regions along the intersection of the cross passage and the upper main tunnel.

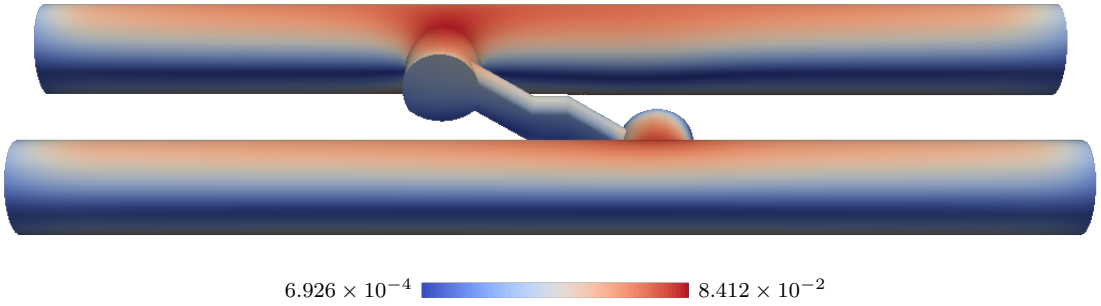
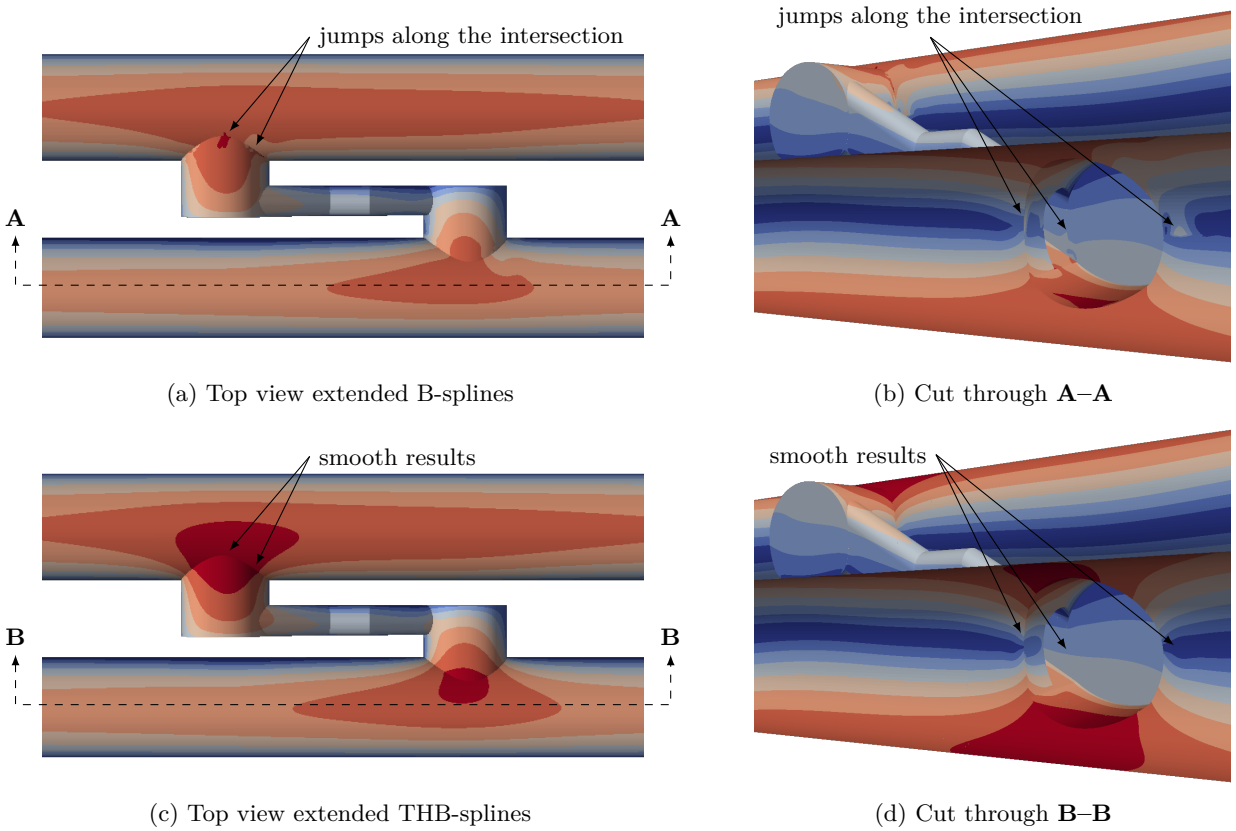


Figure 28: Displacements of the tunnel excavation example using extended THB-splines.



(a) Top view extended B-splines

(b) Cut through **A**–**A**

(c) Top view extended THB-splines

(d) Cut through **B**–**B**

Figure 29: Comparison of displacement contour-plots of the tunnel example solved by (a,b) an extended B-spline discretization without local refinement and (c,d) a discretization using the proposed extended THB-spline approach. In each subfigure, arrows point to the crucial areas of the example. The data range of all plots is the same as in Figure 28.

jumps in the vicinity of the circular hole in the lower tunnel tube shown in Figure 29(b). When the proposed local refinement scheme is applied, a smooth displacement field is obtained also at these critical regions as illustrated in Figure 29(c) and Figure 29(d).

The shown results demonstrate that the extended THB-spline approach is capable of dealing with complex trimming cases. Despite the fact that all trimmed patches are treated individually, the displacements along intersections match very well. It should, however, be pointed out that the improvement of the results from the proposed approach compared to conventional extended B-splines occurs due to the reduction of the extrapolation length as well as the increased number of degrees of freedom along and in the vicinity of the intersections.

6. Conclusion

Extended B-splines are an effective stabilization technique for isogeometric analysis of trimmed geometries. Trimmed basis functions that may compromise the conditioning of system matrices are substituted by extrapolations of B-splines that are well defined within the active domain $\mathcal{A}^{\mathrm{v}}$ of the parameter space. The set of stable basis functions is determined by the location of their anchors. To be precise, it contains all B-splines where the anchors are inside of $\mathcal{A}^{\mathrm{v}}$. We chose this classification for two reasons: first, it is simple and more importantly, it allows the application of extended B-splines to collocation methods. There is a correlation between the stabilization and the polynomial degree p of the basis. In case of the classification used, the length of the extrapolations d_e of stable B-splines increases with p, which may affect the approximation quality in the vicinity of trimming curves.

We enhance the extended B-spline approach by utilizing truncated hierarchical B-splines (THB-splines) as a local refinement technique in order to control d_e. At the same time, the proposed extended THB-spline approach treats the stabilization and the local refinement independently from each other, which simplifies the implementation. This is accomplished by restricting the extended B-spline construction to the finest refinement level of the THB-basis. Several properties of THB-splines are beneficial for the stabilization scheme derived: (i) they are based on the same spline technology as extended B-splines, (ii) they form a partition of unity, (iii) the maximal values of the functions within a THB-basis are well separated, and (iv) all basis functions on the finest refinement level are regular B-splines. The latter property guarantees that no superfluous basis functions are introduced, that is, only B-splines needed for the stabilization are active on the finest level. The third feature is employed to define distinct anchors of THB-splines in the overlapping region of two refinement levels. These points are utilized for an alternative collocation strategy for THB-splines that introduces only one point per basis function in contrast to previous schemes for hierarchical B-splines [38].

A main part of the paper is concerned with the implementation of THB-splines and hierarchical spline data structures in general. Subdivision matrices play an essential role in the present concept. They define the refinement process between the levels of a hierarchical basis and we utilize their components related to active basis functions to provide a very simple balancing criterion. This criterion assures a proper transition between refinement levels so that no hierarchical level is omitted during the local refinement. In other words, B-splines on level ℓ are merely surrounded by basis functions related to $\ell-1$, ℓ, or $\ell+1$. A downside of the simple criterion is that the footprint of the refinement is relatively large. However, the resulting advantages are several and include (i) the stability of the basis, and (ii) ease of implementation and data management of active and deactivated basis functions.

Linear elasticity and scalar Laplace problems are investigated to assess the performance of extended THB-splines. A collocated isogeometric boundary element method (BEM) is used for the analysis. It is shown that the error along trimming curves may dominate the overall quality of the results if the extrapolation length is not controlled by local refinement. On the other hand, excessive local refinement is counterproductive. Based on our study, a good balance between numerical effort and accuracy is attained, if the extrapolation length d_e of the stabilization is bounded by $p/2 \cdot h_\xi$, where h_ξ refers to the average knot span size of the basis before local refinement is applied. The results show excellent agreement with analytical solutions and reference solutions based on non-trimmed models. This criterion introduces only a few refinement levels and it even identifies cases where no local refinement is required. On the whole, the conducted numerical simulations verify that the combination of extended B-splines with THB-splines yields a robust stabilization technique for isogeometric analysis of trimmed geometries.

The introduced local refinement criterion provides an indicator for the required resolution along trimming curves. Alternatively, the numerical method could be enhanced with an error estimator. We want to emphasize that extended THB-splines are by no means restricted to BEM, but can also be applied to a finite element framework. The main difference is that additional aspects, such as the application of essential boundary conditions and weak coupling of adjacent patches have to be considered as well.

7. Acknowledgment

This research was supported by the Austrian Science Fund (FWF): J 3884-N32. T.J.R. Hughes was partially supported by the Office of Naval Research (Grant Nos. N00014-17-1-2119 and N00014-13-1-0500), and by the Army Research Office (Grant No. W911NF-13-1-0220). The support is gratefully acknowledged.

References

[1] Cosmin Anitescu, Yue Jia, Yongjie Jessica Zhang, and Timon Rabczuk. An isogeometric collocation method using superconvergent points. *Computer Methods in Applied Mechanics and Engineering*, 284:1073–1097, 2015. ISSN 0045-7825. doi: 10.1016/j.cma.2014.11.038. URL http://www.sciencedirect.com/science/article/pii/S004578251400471X.

[2] C. Apprich, K. Höllig, J. Hörner, and U. Reif. Collocation with WEB–Splines. *Advances in Computational Mathematics*, 42(4):823–842, 2016. doi: 10.1007/s10444-015-9444-x.

[3] Kendall E. Atkinson. *The Numerical Solution of Integral Equations of the Second Kind*. Cambridge University Press, 1997. URL http://dx.doi.org/10.1017/CBO9780511626340.

[4] F. Auricchio, L. Beirão Da Veiga, T. J. R. Hughes, A. Reali, and G. Sangalli. Isogeometric collocation methods. *Mathematical Models and Methods in Applied Sciences*, 20(11):2075–2107, 2010.

[5] F. Auricchio, L. Beirão da Veiga, T. J. R. Hughes, A. Reali, and G. Sangalli. Isogeometric collocation for elastostatics and explicit dynamics. *Computer Methods in Applied Mechanics and Engineering*, 249–252:2–14, 2012. ISSN 0045-7825. doi: 10.1016/j.cma.2012.03.026. URL http://www.sciencedirect.com/science/article/pii/S0045782512001028.

[6] 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(5–8):229–263, 2010. ISSN 0045-7825.

[7] P. B. Bornemann and F. Cirak. A subdivision-based implementation of the hierarchical b-spline finite element method. *Computer Methods in Applied Mechanics and Engineering*, 253:584–598, January 2013. ISSN 0045-7825. doi: 10.1016/j.cma.2012.06.023. URL http://www.sciencedirect.com/science/article/pii/S0045782512002204.

[8] L. Beirão da Veiga, A. Buffa, G. Sangalli, and R. Vázquez. Mathematical analysis of variational isogeometric methods. *Acta Numerica*, 23:157–287, 2014.

[9] Carl de Boor. *A practical guide to splines*, volume 27 of *Applied Mathematical Sciences*. Springer, New York, 2001. ISBN 0387953663; 9780387953663.

[10] Carl de Boor and G. J. Fix. Spline approximation by quasiinterpolants. *Journal of Approximation Theory*, 8(1):19–45, 1973. ISSN 0021-9045.

[11] S. Demko. On the existence of interpolating projections onto spline spaces. *Journal of Approximation Theory*, 43(2): 151–156, 1985.

[12] Tor Dokken, Tom Lyche, and Kjell Fredrik Pettersen. Polynomial splines over locally refined box-partitions. *Computer Aided Geometric Design*, 30(3):331–356, 2013. ISSN 0167-8396. doi: 10.1016/j.cagd.2012.12.005. URL http://www.sciencedirect.com/science/article/pii/S0167839613000113.

[13] L. Gaul, M. Kögl, and M. Wagner. *Boundary Element Methods for Engineers and Scientists: An Introductory Course with Advanced Topics*. Springer-Verlag, Berlin-Heidelberg-New York, 2003.

[14] Carlotta Giannelli, Bert Jüttler, and Hendrik Speleers. THB-splines: The truncated basis for hierarchical splines. *Computer Aided Geometric Design*, 29(7):485–498, October 2012. ISSN 0167-8396. doi: 10.1016/j.cagd.2012.03.025. URL http://www.sciencedirect.com/science/article/pii/S0167839612000519.

[15] Hector Gomez and Laura De Lorenzis. The variational collocation method. *Computer Methods in Applied Mechanics and Engineering*, 309:152–181, 2016. ISSN 0045-7825. doi: 10.1016/j.cma.2016.06.003. URL http://www.sciencedirect.com/science/article/pii/S0045782516305035.

[16] René R. Hiemstra, Francesco Calabrò, Dominik Schillinger, and Thomas 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*, 316:966–1004, April 2017. ISSN 0045-7825. doi: 10.1016/j.cma.2016.10.049. URL http://www.sciencedirect.com/science/article/pii/S004578251631489X.

[17] K. Höllig. *Finite Element Methods with B-Splines*, volume 26 of *Frontiers in Applied Mathematics*. SIAM, 2003.

[18] K. Höllig and U. Reif. Nonuniform web-splines. *Computer Aided Geometric Design*, 20(5):277–294, 2003.

[19] K. Höllig, U. Reif, and J. Wipper. B-spline approximation of Neumann problems. *Mathematisches Institut A, University of Stuttgart*, 2001.

[20] K. Höllig, U. Reif, and J. Wipper. Weighted extended B-spline approximation of Dirichlet problems. *SIAM Journal on Numerical Analysis*, 39(2):442–462, 2002.

[21] Thomas J. R. Hughes. *The finite element method: linear static and dynamic finite element analysis*. Courier Corporation, 2000.

[22] Kjetil André Johannessen, Trond Kvamsdal, and Tor Dokken. Isogeometric analysis using LR B-splines. *Computer Methods in Applied Mechanics and Engineering*, 269:471–514, 2014. ISSN 0045-7825. doi: 10.1016/j.cma.2013.09.014. URL http://www.sciencedirect.com/science/article/pii/S0045782513002417.

[23] Rainer Kraft. *Adaptive and linearly independent multilevel B-splines*. SFB 404, Geschäftsstelle, 1997.

[24] V. D. Kupradze, T. G. Gegelia, M. O. Basheleishvili, and T. V. Burchuladze. *Three-Dimensional Problems of the Mathematical Theory of Elasticity and Thermoelasticity*, volume 25 of *Applied Mathematics and Mechanics*. North-Holland Publishing Company, 1979.

[25] Kang Li and Xiaoping Qian. Isogeometric analysis and shape optimization via boundary integral. *Computer-Aided Design*, 43(11):1427–1437, 2011. ISSN 0010-4485. URL http://www.sciencedirect.com/science/article/pii/S0010448511002302.

[26] 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*, 319:515–548, June 2017. ISSN 0045-7825. doi: 10.1016/j.cma.2017.03.009. URL http://www.sciencedirect.com/science/article/pii/S0045782516318254.

[27] V. Mantic. A new formula for the c-matrix in the somigliana identity. *Journal of Elasticity*, 33:191–201, 1993.

[28] Benjamin Marussig. *Seamless Integration of Design and Analysis through Boundary Integral Equations*. Monographic Series TU Graz: Structural Analysis. Verlag der Technischen Universität Graz, 2016. doi: 10.3217/978-3-85125-440-2.

[29] Benjamin Marussig and Thomas J. R. Hughes. A review of trimming in isogeometric analysis: Challenges, data exchange and simulation aspects. *Archives of Computational Methods in Engineering*, pages 1–69, 2017. ISSN 1886-1784. doi: 10.1007/s11831-017-9220-9. URL http://dx.doi.org/10.1007/s11831-017-9220-9.

[30] Benjamin Marussig, Jürgen Zechner, Gernot Beer, and Thomas-Peter Fries. Fast isogeometric boundary element method based on independent field approximation. *Computer Methods in Applied Mechanics and Engineering*, 284:458–488, 2015. ISSN 0045-7825. doi: 10.1016/j.cma.2014.09.035. URL http://www.sciencedirect.com/science/article/pii/S0045782514003582.

[31] Benjamin Marussig, Jürgen Zechner, Gernot Beer, and Thomas-Peter Fries. Stable isogeometric analysis of trimmed geometries. *Computer Methods in Applied Mechanics and Engineering*, 316:497–521, 2016. doi: 10.1016/j.cma.2016.07.040.

[32] Benjamin Marussig, Jürgen Zechner, Gernot Beer, and Thomas-Peter Fries. Integration of design and analysis through boundary integral equations. In *VII European Congress on Computational Methods in Applied Sciences and Engineering, ECCOMAS*, 2016. URL https://eccomas2016.org/proceedings/pdf/5812.pdf.

[33] M. Montardini, G. Sangalli, and L. Tamellini. Optimal-order isogeometric collocation at Galerkin superconvergent points. *Computer Methods in Applied Mechanics and Engineering*, 316:741–757, 2017. ISSN 0045-7825. doi: https://doi.org/10.1016/j.cma.2016.09.043. URL http://www.sciencedirect.com/science/article/pii/S0045782516312750.

[34] Les A. Piegl and Wayne Tiller. *The NURBS book*. Springer, New York, NY, USA, 2nd edition, 1997. ISBN 3-540-61545-8.

[35] Thomas Rüberg and F. Cirak. Subdivision-stabilised immersed b-spline finite elements for moving boundary flows. *Computer Methods in Applied Mechanics and Engineering*, 209–212:266–283, 2012. ISSN 0045-7825.

[36] S. A. Sauter and C. Schwab. *Boundary Element Methods*, volume 39. Springer, Heidelberg Dordrecht London New York, 2011.

[37] Dominik Schillinger, Luca Dedè, Michael A. Scott, John A. Evans, Michael J. Borden, Ernst Rank, and Thomas 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*, 249:116–150, December 2012. ISSN 0045-7825. doi: 10.1016/j.cma.2012.03.017. URL http://www.sciencedirect.com/science/article/pii/S004578251200093X.

[38] Dominik Schillinger, John A. Evans, Alessandro Reali, Michael A. Scott, and Thomas J. R. Hughes. Isogeometric collocation: Cost comparison with galerkin methods and extension to adaptive hierarchical NURBS discretizations. *Computer Methods in Applied Mechanics and Engineering*, 267:170–232, 2013. ISSN 0045-7825. doi: doi:10.1016/j.cma.2013.07.017.

[39] 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: 197–221, 2013.

[40] M. A. Scott, D. C. Thomas, and E. J. Evans. Isogeometric spline forests. *Computer Methods in Applied Mechanics and Engineering*, 269:222–264, February 2014. ISSN 0045-7825. doi: 10.1016/j.cma.2013.10.024. URL http://www.sciencedirect.com/science/article/pii/S0045782513002764.

[41] Thomas W. Sederberg, Jianmin Zheng, Almaz Bakenov, and Ahmad Nasri. T-splines and T-NURCCs. *ACM Transactions on Graphics*, 22(3):477–484, #jul# 2003. ISSN 0730-0301.

[42] Olaf Steinbach. *Numerical approximation methods for elliptic boundary value problems: finite and boundary elements*. Springer, New York, NY, 2008. ISBN 0387313125, 9780387313122.

[43] 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(49):3554–3567, December 2011. ISSN 0045-7825. doi: 10.1016/j.cma.2011.09.004. URL http://www.sciencedirect.com/science/article/pii/S0045782511002933.