

Adaptive Refinement for Unstructured T-Splines with Linear Complexity

Roland Maier*, Philipp Morgenstern†, Thomas Takacs‡

May 3, 2022

Abstract. We present an adaptive refinement algorithm for T-splines on unstructured 2D meshes. While for structured 2D meshes, one can refine elements alternatingly in horizontal and vertical direction, such an approach cannot be generalized directly to unstructured meshes, where no two unique global mesh directions can be assigned. To resolve this issue, we introduce the concept of direction indices, i.e., integers associated to each edge, which are inspired by theory on higher-dimensional structured T-splines. Together with refinement levels of edges, these indices essentially drive the refinement scheme. We combine these ideas with an edge subdivision routine that allows for I-nodes, yielding a very flexible refinement scheme that nicely distributes the T-nodes, preserving global linear independence, analysis-suitability (local linear independence) except in the vicinity of extraordinary nodes, sparsity of the system matrix, and shape regularity of the mesh elements. Further, we show that the refinement procedure has linear complexity in the sense of guaranteed upper bounds on a) the distance between marked and additionally refined elements, and on b) the ratio of the numbers of generated and marked mesh elements.

Keywords. T-splines, unstructured meshes, adaptive refinement

1. Introduction

T-splines refer to a realization of B-splines on irregular meshes. They were introduced in [1] in the context of Computer-Aided Design and allow for local mesh refinement [2] without the requirement of a hierarchical basis. They were successfully applied in the context of Isogeometric Analysis [3, 4], but in the beginning also showed weaknesses such as linear dependencies [5] or even non-nestedness of approximation spaces [6] in certain cases. The problem of linear dependence was overcome by the introduction of analysis-suitability [7], which requires that T-junction extensions do not intersect, and the more abstract but equivalent concept of dual-compatibility [8]. Then again, the refinement algorithm in [9] yielded nested spline spaces and also preserved linear independence. Further works on the theoretical background considered arbitrary polynomial degrees [10] and the construction of T-spline meshes from boundary representations in 3D [11, 12]. At that time,

*Institute of Mathematics, Friedrich Schiller University Jena, Ernst-Abbe-Platz 2, 07743 Jena, Germany

†Institut für Angewandte Mathematik, Leibniz Universität Hannover, Welfengarten 1, 30167 Hannover, Germany

‡Johann Radon Institute for Computational and Applied Mathematics, Österreichische Akademie der Wissenschaften, Altenberger Straße 69, 4040 Linz, Austria

however, the linear independence of higher-dimensional T-splines was only characterized through the dual-compatibility criterion. This was resolved in [13], where a definition of T-junction extensions and analysis-suitability was presented in three dimensions, which was generalized to arbitrary dimensions in [14].

Simultaneously, the research on 2D T-splines continued, for instance in the context of analysis-suitable T-spline spaces with globally highest smoothness [15] or locally reduced smoothness [16]. Concerning unstructured meshes, T-splines were originally defined locally on structured, rectangular regions but already in the earliest works extended to more unstructured domains by combining them with subdivision surface constructions near so-called extraordinary nodes. However, to avoid the non-finite representation of subdivision surfaces, special geometrically continuous constructions based on local splits and/or locally higher polynomial degree were introduced in [17, 18, 19]. In [20], unstructured T-splines were constructed that are piecewise bicubic, globally C^1 -smooth and preserve linear independence also around extraordinary nodes. The construction is based on the concept of having a convenient design space and an analysis space for theoretical purposes. The construction was later extended in [21] to allow more general mesh configurations. While the constructions in [20, 19, 21] allow quite flexible mesh configurations and result in C^1 -smooth spaces, they are all based on specific splits of the elements near extraordinary nodes and are formulated for cubic T-splines only. In contrast, the construction we present in this paper requires a more restrictive separation of extraordinary nodes, but is stated for T-splines of general polynomial degree.

In this contribution, we consider unstructured spline spaces on two-dimensional unstructured meshes motivated by [22], which yield C^{p-1} -continuous splines except for the vicinity of extraordinary nodes, where the continuity is reduced to C^0 -continuity. These spaces are combined with the theory on higher-dimensional T-splines in [14]. More precisely, we show that unstructured meshes around an extraordinary node can be understood as traces of higher-dimensional structured meshes, for which the refinement routine of [14] is applicable. In practice, this ‘embedding’ of the mesh can be avoided by the idea of so-called direction indices, i.e., integers associated to each edge. These integers can be understood as the equivalent of a certain space dimension in higher-dimensional structured meshes and mark a crucial ingredient in the resulting refinement algorithm together with the refinement levels of edges.

Throughout this paper, we restrict ourselves to the case of odd-degree spline functions. For even polynomial degrees, we refer the reader to [10], where the additional concept of anchor elements is explained. Further note that the construction we present here yields analysis-suitable T-splines. This condition may be weakened by generating only so-called AS++ T-splines, as developed in [23, 24, 25].

This paper is organized as follows. Section 2 introduces the quadrilateral 2D meshes on which we then construct structured and unstructured T-spline spaces in Section 3. In Section 4, we present our new refinement algorithm and rigorously analyze properties of the generated meshes and spaces in Section 5 before we conclude in Section 6. Additionally, the connections to manifold splines [22] are outlined in the appendix, where also an extension to a larger class of unstructured meshes is presented. Further, some connections to Isogeometric Analysis are drawn.

2. Preliminaries

The goal of this paper is to define T-splines along with an appropriate adaptive refinement scheme on unstructured T-meshes over polygonal planar domains, planar domains with curved boundaries and surface domains. All domains may have holes. In case of surfaces, we assume that they are orientable. In principle, the concepts that we introduce can be generalized to higher-dimensional objects, such as volumes or space-time domains. We first introduce unstructured T-splines over

quadrilateral partitions of the plane. The polygonal partition serves as a template for the underlying topological structure of the mesh. The definition may then be generalized to manifold-like objects.

In the following subsections, we introduce necessary notation and state useful results. For convenience, the main notation is also summarized in Table 1.

$Q \in \mathcal{Q}$	mesh elements
$E \in \mathcal{E}$	mesh edges
$N = \{\mathbf{n}\} \in \mathcal{N}$	mesh nodes, $\mathbf{n} \in \mathbb{R}^d$
$\mathcal{M} = (\mathcal{Q}, \mathcal{E}, \mathcal{N})$	mesh (for higher dimensions in principle a longer vector)
$\mathcal{E}_{\text{int}}, \mathcal{E}_{\text{bd}}, \mathcal{N}_{\text{int}}, \mathcal{N}_{\text{bd}}$	interior/boundary edges/nodes
\mathcal{N}_A	anchors
\mathcal{N}_{EO}	extraordinary nodes
$\mathcal{E}(Q)$	edges of an element Q (contained in ∂Q), similarly $\mathcal{N}(Q)$, $\mathcal{N}(E)$
v_N	element valence of node N , $v_N = \mathcal{Q}(N) $
$\mathcal{D}(\mathcal{M}')$	(planar or surface) domain corresponding to a submesh \mathcal{M}'
$\mathcal{D}(Q')$	(planar or surface) domain corresponding to a set of elements Q'
\hat{A}	parameter domain of a submesh or mesh object (element, set of elements) A
$L(E)$	length of edge E
$\ell(E)$	refinement level of edge E
$\text{di}(E)$	direction index of edge E
$[i]$ (as subscript)	refers to refinement level i

Table 1: Summary of the notation

2.1. Partitions and meshes

Let $\Omega \subset \mathbb{R}^2$ be an open, bounded domain with a polygonal boundary $\partial\Omega$. On the domain we define a regular partition $(\mathcal{Q}, \mathcal{E}, \mathcal{N})$ into quadrilaterals \mathcal{Q} , edges \mathcal{E} , and nodes \mathcal{N} . Every node $N = \{\mathbf{n}\} \in \mathcal{N}$ corresponds to a point $\mathbf{n} \in \mathbb{R}^2$, every edge $E \in \mathcal{E}$ is a line segment between two nodes (excluding the endpoints) and every element $Q \in \mathcal{Q}$ is an open quadrilateral. All edges are either interior edges \mathcal{E}_{int} or boundary edges \mathcal{E}_{bd} , with $\mathcal{E} = \mathcal{E}_{\text{int}} \cup \mathcal{E}_{\text{bd}}$, similarly nodes are either interior nodes \mathcal{N}_{int} or boundary nodes \mathcal{N}_{bd} , $\mathcal{N} = \mathcal{N}_{\text{int}} \cup \mathcal{N}_{\text{bd}}$. All mesh objects are disjoint, i.e., we have for all $A, A' \in \mathcal{Q} \cup \mathcal{E} \cup \mathcal{N}$, that $A \cap A' = \emptyset$, and we have

$$\Omega = \bigcup_{A \in \mathcal{Q} \cup \mathcal{E}_{\text{int}} \cup \mathcal{N}_{\text{int}}} A \quad \text{and} \quad \partial\Omega = \bigcup_{A \in \mathcal{E}_{\text{bd}} \cup \mathcal{N}_{\text{bd}}} A.$$

The elements \mathcal{Q} , edges \mathcal{E} , and nodes \mathcal{N} , together with their connectivity structure, form a topological mesh \mathcal{M} . The connectivity relations between mesh objects are explained in more detail in the following definition.

Definition 2.1 (topological structure of the mesh). A mesh $\mathcal{M} = (\mathcal{Q}, \mathcal{E}, \mathcal{N})$ is given as a triple of sets of mesh objects (elements, edges, and nodes) together with their topological structure, i.e., the mesh objects satisfy the following connectivity relations:

- An edge or node $A \in \mathcal{E} \cup \mathcal{N}$ is connected to an element $Q \in \mathcal{Q}$, if $A \subset \partial Q$. We denote by $\mathcal{E}(Q)$ and $\mathcal{N}(Q)$ the sets of all edges and nodes, respectively, connected to the element Q .
- A node $N \in \mathcal{N}$ is connected to an edge $E \in \mathcal{E}$, if $N \subset \partial E$. We denote by $\mathcal{N}(E)$ the set of all nodes connected to the edge.
- We denote by $\mathcal{Q}(E)$ the set of those elements $Q \in \mathcal{Q}$, such that $E \in \mathcal{E}(Q)$.

- We denote by $\mathcal{Q}(\mathcal{N})$ and $\mathcal{E}(\mathcal{N})$ the sets of those elements and edges, such that $N \in \mathcal{N}(\mathcal{Q})$ and $N \in \mathcal{N}(\mathcal{E})$, respectively.
- Every node N has element valence $v_N \in \mathbb{N}^+$. That means, N is connected to v_N elements and v_N edges (if it is an interior node) or $v_N + 1$ edges (if it is a boundary node), i.e., $|\mathcal{Q}(\mathcal{N})| = v_N$, $|\mathcal{E}(\mathcal{N})| = v_N$ for $N \in \mathcal{N}_{\text{int}}$ and $|\mathcal{E}(\mathcal{N})| = v_N + 1$ for $N \in \mathcal{N}_{\text{bd}}$.

Proposition 2.2. *We have the following:*

- *By definition, the connectivity relations are symmetric, e.g., $N \in \mathcal{N}(\mathcal{Q})$ if and only if $\mathcal{Q} \in \mathcal{Q}(\mathcal{N})$.*
- *Every element is connected to at least four edges and nodes, i.e., $|\mathcal{E}(\mathcal{Q})| = |\mathcal{N}(\mathcal{Q})| \geq 4$.*
- *Every edge is connected to exactly two nodes, i.e., $|\mathcal{N}(\mathcal{E})| = 2$.*
- *Every interior edge is connected to two elements, $|\mathcal{Q}(\mathcal{E})| = 2$ for $\mathcal{E} \in \mathcal{E}_{\text{int}}$, and every boundary edge to one element, $|\mathcal{Q}(\mathcal{E})| = 1$ for $\mathcal{E} \in \mathcal{E}_{\text{bd}}$.*
- *Boundary edges are connected only to boundary nodes and interior nodes are connected only to interior edges.*

Remark 2.3. The concept of a partition of a planar, polygonal domain extends directly to more general meshes over bivariate domains $\Omega \subset \mathbb{R}^d$. The mesh is then composed of bivariate elements, which are curved quadrilateral subdomains without boundary, univariate edges, which are curve segments without endpoints, as well as nodes, which are singletons, each containing a point in \mathbb{R}^d . All mesh objects satisfy the relations described in Definition 2.1 and Proposition 2.2. Hence, from now on we do not distinguish between planar partitions and more general bivariate meshes.

Definition 2.4 (submesh and its domain). Let $\mathcal{M} = (\mathcal{Q}, \mathcal{E}, \mathcal{N})$ be a mesh and let $\mathcal{Q}' \subseteq \mathcal{Q}$ be a subset of elements of the mesh. Then the corresponding *submesh* $\mathcal{M}' = (\mathcal{Q}', \mathcal{E}', \mathcal{N}')$ contains all those mesh objects $A \in \mathcal{E} \cup \mathcal{N}$, such that

$$A \subset \overline{\mathcal{D}(\mathcal{M}')},$$

where the open set

$$\mathcal{D}(\mathcal{M}') = \left(\bigcup_{\mathcal{Q} \in \mathcal{Q}'} \overline{\mathcal{Q}} \right)^\circ$$

is the corresponding *domain* $\mathcal{D}(\mathcal{M}') \subseteq \Omega$ of the submesh. Since the domain depends only on the elements of the submesh, we also use the notation $\mathcal{D}(\mathcal{Q}') = \mathcal{D}(\mathcal{M}')$. The edges and nodes of the submesh are again split into interior ($A \subset \mathcal{D}(\mathcal{M}')$) and boundary ($A \subset \partial\mathcal{D}(\mathcal{M}')$) edges and nodes, respectively.

Definition 2.5 (regular mesh). We call a mesh *regular* if the closure of any element $\mathcal{Q} \in \mathcal{Q}$ contains exactly four edges and four nodes and the closures of any two elements $\mathcal{Q}, \mathcal{Q}' \in \mathcal{Q}$ a) are disjoint or b) equal or c) share exactly one edge and two nodes or d) share exactly one node.

Thus, a regular mesh (or a regular submesh) is a mesh without hanging nodes in its interior.

Definition 2.6 (extraordinary node). A node of a regular mesh is called *extraordinary* if a) it is a boundary node neighboring more than 2 quadrilaterals or b) it is an interior node which is not neighboring exactly 4 mesh elements.

2.2. Parameter domains of elements and structured submeshes

From now on, we consider to have an initial mesh $\mathcal{M}_{[0]} = (\mathcal{Q}_{[0]}, \mathcal{E}_{[0]}, \mathcal{N}_{[0]})$, which is regular. In the following, we introduce parameter domains to mesh elements of the initial mesh. These parameter domains can then be extended from single elements to larger structured submeshes and to refinements of the initial mesh. To this end, we assign a length to each edge.

Definition 2.7 (edge length). On the initial mesh $\mathcal{M}_{[0]}$ we assign the *length* $L(\mathbf{E}) = 1$ to each edge $\mathbf{E} \in \mathcal{E}_{[0]}$.

It is also possible to assign different lengths to different edges, with the restriction that the lengths must be consistent, i.e., for each element the same length is assigned to opposing edges. The choice $L(\mathbf{E}) = 1$ above is for simplicity.

Definition 2.8 (parameter domain). On the initial mesh $\mathcal{M}_{[0]}$ we assign the *parameter domain* $\widehat{Q} =]0, 1[^2$ to each element $Q \in \mathcal{Q}_{[0]}$.

The parameter domain is defined such that it is consistent with the edge lengths. If edge lengths different from $L(\mathbf{E}) = 1$ are assigned, then the parameter domains are changed accordingly.

Assumption 2.9. We assume for all $Q \in \mathcal{Q}_{[0]}$ that there exists a regular, positively-oriented C^∞ -mapping $\mathbf{F}_Q : \widehat{Q} \rightarrow Q$, i.e., there exists a constant $C > 0$ such that $\det \nabla \mathbf{F}_Q(\mathbf{x}) \geq C > 0$ for all $\mathbf{x} \in \widehat{Q}$. This implies that Q is not degenerate, e.g., no edge of Q is empty. Moreover, we assume that for any pair of elements Q, Q' sharing a common edge \mathbf{E} , the mappings \mathbf{F}_Q and $\mathbf{F}_{Q'}$ are continuous along \mathbf{E} , i.e., there exists a common mapping $\mathbf{F}_{Q,Q'} : (0, 1) \times (-1, 1) \rightarrow Q \cup \mathbf{E} \cup Q'$ and Euclidean motions \mathbf{R}_Q and $\mathbf{R}_{Q'}$, such that

$$\mathbf{F}_{Q,Q'}|_{(0,1) \times (0,1)} = \mathbf{F}_Q \circ \mathbf{R}_Q$$

and

$$\mathbf{F}_{Q,Q'}|_{(0,1) \times (-1,0)} = \mathbf{F}_{Q'} \circ \mathbf{R}_{Q'}.$$

By assumption, the Euclidean motions $\mathbf{R}_Q, \mathbf{R}_{Q'}$ must be combinations of a rotation by an angle which is an integer multiple of $\frac{\pi}{2}$ and a translation.

Now that we are given parameter domains \widehat{Q} and mappings \mathbf{F}_Q for each element Q , we can define (mapped) polynomial functions on each element. Moreover, the assumption that there exists a joint mapping for each edge allows us to define continuity of functions across edges. Consequently, we can define *splines* over the mesh \mathcal{M}_0 . Before we do that, we introduce a generalization of the mapping introduced in Assumption 2.9 and extend all definitions to T-meshes, which are obtained from an initial, regular mesh through refinement.

Definition 2.10 (structured mesh). Let \mathcal{M} be any submesh of the initial mesh $\mathcal{M}_{[0]}$, which has a simply connected domain $\mathcal{D}(\mathcal{M})$ and does not contain any extraordinary nodes. We call the mesh *structured*, if in addition there exists a continuous mapping

$$\mathbf{F}_{\mathcal{Q}} : \mathcal{D}(\widehat{\mathcal{M}}) \rightarrow \mathcal{D}(\mathcal{M}),$$

such that $\widehat{\mathcal{M}}$ is a mesh derived from a set of axis-aligned boxes $\widehat{\mathcal{Q}}$, with

$$\widehat{\mathcal{Q}} = \{\mathbf{R}_Q(\widehat{Q}), Q \in \mathcal{Q}\},$$

where \mathbf{R}_Q is a Euclidean motion, $\mathcal{D}(\widehat{\mathcal{M}})$ is simply connected and $\mathbf{F}_{\mathcal{Q}}$ satisfies

$$\mathbf{F}_{\mathcal{Q}} \circ \mathbf{R}_Q = \mathbf{F}_Q,$$

for all $Q \in \mathcal{Q}$. Here \mathbf{F}_Q is the mapping from Assumption 2.9.

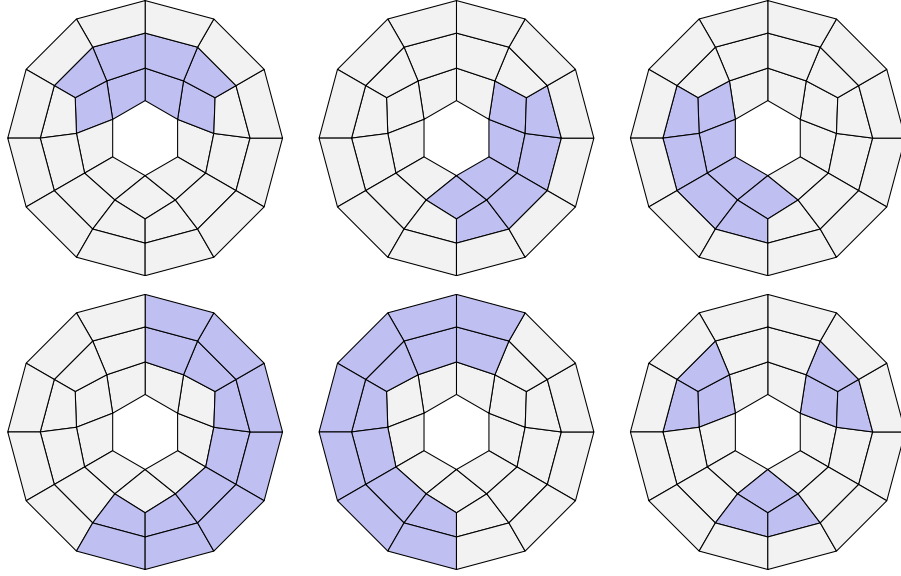


Figure 1: A regular mesh on a ring-shaped domain, which is covered by five structured submeshes (top line, bottom left and center) and three submeshes which are not structured (bottom right).

The definition of a structured mesh carries over directly to any regular mesh, not just submeshes of the initial mesh.

Proposition 2.11. *Let $\mathcal{M} = (\mathcal{Q}, \mathcal{E}, \mathcal{N})$ be a regular mesh. Each element $Q \in \mathcal{Q}$ and each interior edge $E \in \mathcal{E}_{\text{int}}$ is contained inside a structured submesh. Moreover, for each node $N \in \mathcal{N}$, which is not an extraordinary node, the neighborhood $N \cup \mathcal{E}(N) \cup \mathcal{Q}(N)$ is contained inside a structured submesh.*

A visualization of Proposition 2.11 is shown in Figure 1. For practical purposes, it is recommended to find a small number of structured submeshes to cover the domain.

2.3. T-meshes

T-meshes are derived from an initial, regular mesh \mathcal{M}_0 by refinement. In the following, we characterize valid refinements on a mesh.

Definition 2.12 (T-mesh and splitting operations). A T-mesh is either a regular mesh or it is obtained by a splitting operation from a T-mesh, where we allow the following splitting operations:

- An edge $E \in \mathcal{E}$ can be split by bisection, creating two new edges E_1 and E_2 of length $L(E_i) = \frac{1}{2}L(E)$ and one new node $N = \overline{E_1} \cap \overline{E_2}$.
- An element $Q \in \mathcal{Q}$ can be split if the two opposite edges are bisected, creating two new elements Q_1, Q_2 and one new edge $E = \overline{Q_1} \cap \overline{Q_2}$. The length of the new edge is assigned such that it is consistent with the lengths of the existing edges of the element Q . Parameter domains are assigned to the new elements by bisecting the parameter domain \hat{Q} of the element Q . Mappings are then assigned trivially to the new elements by restricting \mathbf{F}_Q to the new parameter domains.

Thus, all parameter domains of elements of a T-mesh are axis aligned boxes with edge lengths in $\{1/2^i, i \in \mathbb{N}_0\}$. Since the parameter domains are inherited from the underlying regular mesh, many properties of the regular mesh can be generalized to T-meshes.

Definition 2.13 (nodes of a T-mesh). A node of a T-mesh is called an *extraordinary node* if and only if it is an extraordinary node of the underlying regular mesh \mathcal{M}_0 . An inner node N of a T-mesh is called an *I-node* if $|\mathcal{Q}(N)| = 2$, it is called a *T-node* if it is not an extraordinary node and $|\mathcal{Q}(N)| = 3$, it is called a *regular node* if $|\mathcal{Q}(N)| = 4$. Analogously, a boundary node is called an I-node if $|\mathcal{Q}(N)| = 1$ and a regular node if $|\mathcal{Q}(N)| = 2$. The set of extraordinary nodes is denoted \mathcal{N}_{EO} .

We can now introduce the concept of structured T-meshes and structured submeshes of the global T-mesh.

Definition 2.14 (structured T-mesh). A submesh of a T-mesh is *structured* if all conditions of Definition 2.10 are satisfied, where for each split element Q the parameter domain \widehat{Q} and mapping \mathbf{F}_Q are given as in Definition 2.12. Otherwise, the T-mesh is called *unstructured*.

By definition, a structured submesh of an unstructured T-mesh contains no extraordinary nodes in its interior. For each structured submesh \mathcal{M}^* we can define standard (structured) T-splines on $\widehat{\mathcal{M}^*}$, by defining them on $\mathcal{D}(\widehat{\mathcal{M}^*})$ and mapping them onto $\mathcal{D}(\mathcal{M}^*)$ using the mapping \mathbf{F}_{Q^*} .

We moreover propose the following for T-meshes which are obtained by refining a regular, structured mesh.

Proposition 2.15. *Let \mathcal{M}^* be a refinement of a mesh \mathcal{M}' , following Definition 2.12. Then \mathcal{M}^* is structured if and only if \mathcal{M}' is structured, with $\mathcal{D}(\mathcal{M}^*) = \mathcal{D}(\mathcal{M}')$ and $\mathbf{F}_{Q^*} = \mathbf{F}_Q$.*

From this it follows that Proposition 2.11 extends directly to T-meshes. Since the refinement of a T-mesh is based on bisections of edges, every edge in a T-mesh has length $|E| = 2^{-\ell(E)}$, where $\ell(E)$ is the refinement level of the edge E , i.e., the number of refinement steps that have been performed on it. The refinement level of an edge is defined properly in Definition 4.1 below.

2.4. The mesh metric and separation of extraordinary vertices

Definition 2.16 (mesh metric). We define a metric on Ω by the following steps:

1. Given an initial, regular mesh $\mathcal{M}_{[0]} = (\mathcal{Q}_{[0]}, \mathcal{E}_{[0]}, \mathcal{N}_{[0]})$, we understand the *distance* $\text{dist}_{[0]}(A, B)$ between two distinct nodes $A, B \in \mathcal{N}_{[0]}$ as the minimal number $n \in \mathbb{N}$ such that there is a sequence of n vertex-connected elements of $\mathcal{Q}_{[0]}$ with A being a vertex of the first and B a vertex of the last element. We consider two elements vertex-connected if they share a common vertex. For $A = B$, we set $\text{dist}_{[0]}(A, B) = 0$. This defines a metric on \mathcal{N}_0 .
2. For the uniform dyadic refinement $\mathcal{M}_{[j+1]}$ of $\mathcal{M}_{[j]}$, we observe for all nodes $A, B \in \mathcal{N}_{[j]}$ that

$$\text{dist}_{[j]}(A, B) = \frac{1}{2} \text{dist}_{[j+1]}(A, B) . \quad (1)$$

We extend the definition of $\text{dist}_{[j]}$ by requiring (1) for all nodes $A, B \in \mathcal{N}_{[j+1]}$.

3. The recursive application of step 2 yields for any $j \in \mathbb{N}$ and $A, B \in \mathcal{N}_{[j]}$ that

$$\text{dist}_{[0]}(A, B) = 2^{-j} \text{dist}_{[j]}(A, B) ,$$

which defines a metric on $\bigcup_{j \in \mathbb{N}_0} \mathcal{N}_{[j]}$, this is, on all nodes of all successive dyadic refinements of the initial mesh $\mathcal{M}_{[0]}$. This set of nodes is dense in Ω . The continuous extension defines a metric dist on Ω , which coincides with the piecewise bilinear interpolation of $\text{dist}_{[0]}$. We denote this metric by $\text{dist}(\mathbf{a}, \mathbf{b})$ for $\mathbf{a}, \mathbf{b} \in \Omega$.

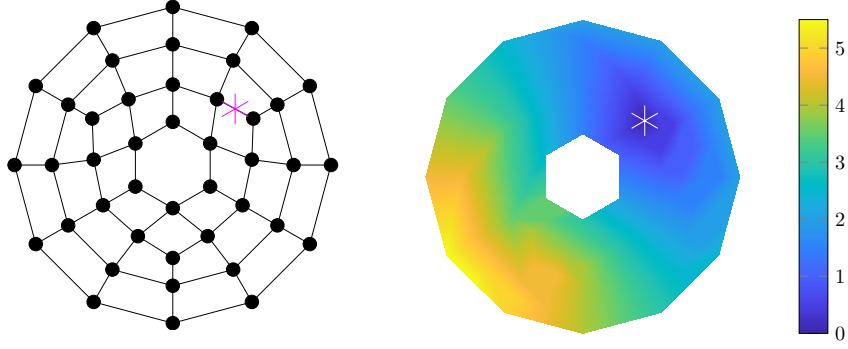


Figure 2: Example for the distance to an indicated point with respect to dist .

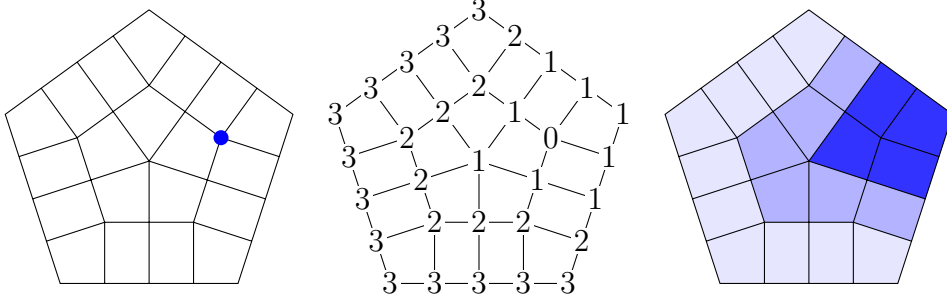


Figure 3: Left: example of a regular mesh with one node v marked. Center: distance of each node to v . Right: 1-disk (blue), 2-disk (blue and light blue), and 3-disk (blue, light blue and very light blue) of v .

On a uniform structured grid of unit squares, this metric corresponds to the ∞ -norm. However, dist is in general neither homogeneous nor invariant under translation and hence does not correspond to a norm, see Figure 2 for an illustration.

Definition 2.17 (k -disk). For $k \in \mathbb{N}$ and any node $N \in \mathcal{N}$ of a mesh \mathcal{M} , the k -disk $D_k(N)$ around N is defined recursively, where we have $D_1(N) = \mathcal{Q}(N)$ and define

$$\mathcal{N}(D_{k-1}(N)) = \bigcup_{Q' \in D_{k-1}(N)} \mathcal{N}(Q').$$

and

$$D_k(N) = \bigcup_{N' \in \mathcal{N}(D_{k-1}(N))} \mathcal{Q}(N'),$$

for $k \geq 2$.

Thus, on the initial, regular mesh \mathcal{M}_0 , the k -disk around N is the set of all elements within a distance to N smaller or equal to k .

3. T-spline spaces

We recall below the notion of splines over general T-meshes and related definitions. Section 3.2 recalls the construction of T-splines over structured meshes in arbitrary dimensions and Section 3.3 introduces a construction of T-splines over unstructured quadrilateral meshes.

3.1. Spline spaces over T-meshes

We consider a two-dimensional mesh $\mathcal{M} = (\mathcal{Q}, \mathcal{E}, \mathcal{N})$ or a more general mesh as in Remark 2.3. Over such a mesh we want to define splines, that is, piecewise polynomial functions with certain prescribed continuity. Having given parameter domains for all elements and mappings as introduced in Section 2.3 that relate parameter domains across edges, we can define polynomials on each element and continuity of functions across edges.

Using the mapping \mathbf{F}_Q over the parameter domain \widehat{Q} , we can define polynomials on the mesh element Q .

Definition 3.1 (polynomial on an element). Let $Q \in \mathcal{Q}$ and let $p \in \mathbb{N}_0$. A function $\varphi : Q \rightarrow \mathbb{R}$ is called *polynomial* of bi-degree (p, p) if $\varphi \circ \mathbf{F}_Q \in \mathbb{P}^{(p,p)}$. In short, we denote this by $\varphi \in \mathbb{P}^{(p,p)}(Q)$.

Definition 3.2 (spline on a mesh). A function $\varphi : \Omega \rightarrow \mathbb{R}$ is a *continuous, piecewise polynomial spline* of bi-degree (p, p) if φ is a polynomial of bi-degree (p, p) on all elements $Q \in \mathcal{Q}$ and $\varphi \in C^0(\Omega)$.

Continuity of a function on Ω can be defined directly on the domain itself. However, differentiability (and higher-order continuity) depends on the underlying mesh and embedding. Thus, we define it on the parameter domains of structured submeshes.

Definition 3.3 (continuity across an edge). Let $E \in \mathcal{E}_{\text{int}}$ be an interior edge of the mesh and let $\mathcal{Q}(E)$ contain the two neighboring elements. A function $\varphi : \mathcal{D}(\mathcal{Q}(E)) \rightarrow \mathbb{R}$ is C^k -smooth across the edge E , if $\varphi \circ \mathbf{F}_{\mathcal{Q}(E)} \in C^k(\mathcal{D}(\widehat{\mathcal{Q}(E)}))$. In short, we denote this by $\varphi \in C^k(E)$.

Thus, a function is C^k -smooth across an edge of the mesh if its pullback is C^k -smooth in the parameter domain. For each integer $k \geq 0$ and for each structured submesh \mathcal{M}^* , we can analogously define the space $C^k(\mathcal{D}(\mathcal{M}^*))$.

Definition 3.4 (spline space over a T-mesh). Let $\mathcal{M} = (\mathcal{Q}, \mathcal{E}, \mathcal{N})$ be a T-mesh over Ω , $p \in \mathbb{N}_0$ be a prescribed polynomial degree and $k : \mathcal{E} \rightarrow \mathbb{N}_0$ be a function associating an order of continuity to every edge of the mesh. The spline space $\mathcal{S}^p(\mathcal{M}, k)$ is defined as

$$\mathcal{S}^p(\mathcal{M}, k) = \left\{ \varphi : \Omega \rightarrow \mathbb{R} \mid \begin{array}{ll} \varphi \in \mathbb{P}^{(p,p)}(Q) & \text{for each } Q \in \mathcal{Q} \text{ and} \\ \varphi \in C^{k(E)}(E) & \text{for each } E \in \mathcal{E} \end{array} \right\}.$$

Certain constructions for bases on splines over T-meshes can be found in [22]. Note that the above construction can be further generalized using meshes defined on so-called parameter manifolds. This generalization is briefly covered in Section A in the appendix.

In the following, we introduce T-splines over T-meshes. First, we consider extensions to higher dimensions and, in a second step, unstructured T-splines accounting for the occurrence of extraordinary nodes. The T-mesh serves as a control structure of the T-spline space. The resulting T-splines are not piecewise polynomials over the control T-mesh, but over the so-called Bézier mesh, which is another T-mesh obtained from the control T-mesh through refinement. This is further explained below. We restrict ourselves to constructions of odd degree p , where, for structured T-meshes, the nodes of the mesh are anchors of the T-splines. Each anchor corresponds to one T-spline basis function. For even and mixed degree, see [10, 26].

Definition 3.5 (Bézier mesh). The mesh $\text{BEZIER}(\mathcal{M}) = (\text{BEZIER}(\mathcal{Q}), \text{BEZIER}(\mathcal{E}), \text{BEZIER}(\mathcal{N}))$ is the coarsest partition of the domain Ω such that all T-splines defined over the control T-mesh \mathcal{M} are piecewise polynomials on $\text{BEZIER}(\mathcal{Q})$.

The Bézier mesh $\text{BEZIER}(\mathcal{M})$ depends on the polynomial degree p of the spline space. The control mesh and Bézier mesh differ only near T-nodes and I-nodes. If \mathcal{M} is regular, then $\text{BEZIER}(\mathcal{M}) = \mathcal{M}$. For the class of meshes generated by our refinement scheme, the elements of the Bézier mesh can be computed with Algorithm 1.

Algorithm 1 Bézier mesh

```

1: procedure BEZIER( $\mathcal{M}$ )
2:   for  $j = 1, \dots, \lceil \frac{p}{2} \rceil$  do
3:      $\mathcal{Q}_T \leftarrow \{Q \in \mathcal{Q} \mid \#\mathcal{E}(Q) = \#\mathcal{N}(Q) = 5\}$      $\triangleright$  these are all elements that neighbor T- or
       I-nodes, see also Remark 5.7 for details
4:     for all  $Q \in \mathcal{Q}_T$  do
5:        $E \leftarrow$  unique edge in  $\mathcal{E}(Q)$  that has two opposite edges
6:        $\mathcal{M} \leftarrow$  SUBDIV( $\mathcal{M}, E$ )                                 $\triangleright$  see Algorithm 2
7:     end for
8:   end for
9:   return  $\mathcal{M}$ 
10: end procedure

```

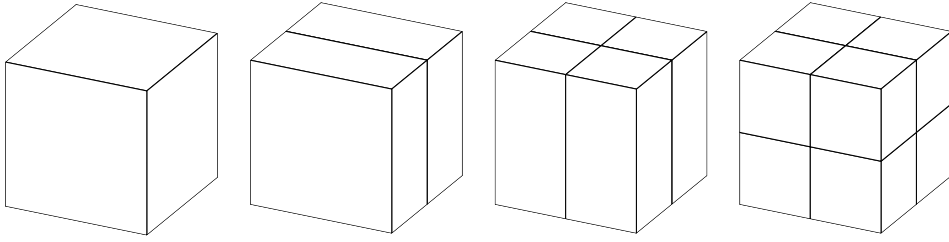


Figure 4: Example for uniform meshes $\mathcal{M}_{u[0]}^d$, $\mathcal{M}_{u[1/3]}^d$, $\mathcal{M}_{u[2/3]}^d$, and $\mathcal{M}_{u[1]}^d$, with $\mathcal{M}_{u[0]}^d$ consisting of a single element, $d = 3$.

3.2. Higher-dimensional structured T-spline basis

In this subsection, we explain the construction of T-splines in meshes that consist of axis-aligned d -dimensional boxes. For details, we refer the reader to [14, Section 5.3]. Note that such meshes are composed of d -dimensional elements, $(d - 1)$ -dimensional faces, \dots , 1-dimensional edges, and 0-dimensional nodes. While the elements, faces, and nodes play important roles for the definition of spline spaces of odd degree, the other mesh elements do not need to be defined rigorously. We emphasize that the definitions in Sections 2 and 3.1 can be naturally extended to dimensions $d > 2$ and we include a superscript d in such cases to distinguish higher-dimensional meshes and two-dimensional ones.

Definition 3.6 (uniform meshes). For each level $\ell = k + \frac{j}{d}$, with $k \in \mathbb{N}$ and $j \in \{0, \dots, d - 1\}$, we define the elements of the d -dimensional tensor-product mesh $\mathcal{M}_{u[\ell]}^d$ as

$$\begin{aligned}
 \mathcal{Q}_{u[\ell]}^d := & \left\{ (x_1 - 2^{-k-1}, x_1) \times \dots \times (x_j - 2^{-k-1}, x_j) \times (x_{j+1} - 2^{-k}, x_{j+1}) \times \dots \times (x_d - 2^{-k}, x_d) \right. \\
 & \left. \mid 2^{k+1}x_i \in \{-2^{k+1}p + 1, \dots, 2^{k+1}p\} \text{ for } i = 1, \dots, j \right. \\
 & \left. \text{and } 2^kx_i \in \{-2^k p + 1, \dots, 2^k p\} \quad \text{for } i = j + 1, \dots, d \right\} \quad (2)
 \end{aligned}$$

with $n_1, \dots, n_d \in \mathbb{N}$. We moreover denote by $\mathcal{E}_{u[\ell]}^d$ the $(d - 1)$ -dimensional faces between elements and by $\mathcal{N}_{u[\ell]}^d$ the vertices of all elements in $\mathcal{Q}_{u[\ell]}^d$; see also Figure 4 for an illustration. Note that we intentionally choose the domain $(-p, p)^d$ here since we explicitly require such a domain in Section 3.3.

The class of d -dimensional T-meshes \mathcal{M}^d considered below is the set of all meshes where \mathcal{Q}^d consists of finitely many elements from uniform meshes as above with possibly different levels such

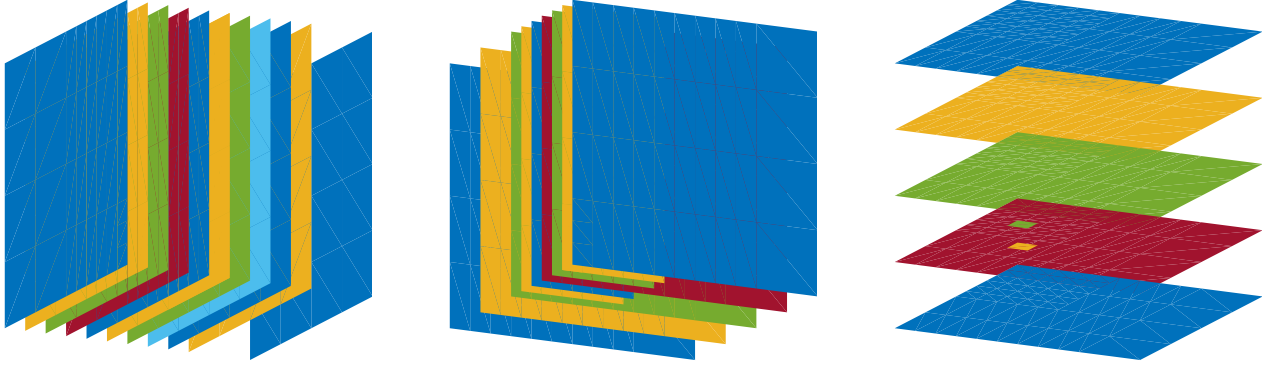


Figure 5: 1-orthogonal, 2-orthogonal and 3-orthogonal skeleton of a 3D cube refined in the front corner, taken from [13].

that any two elements of \mathcal{Q}^d are disjoint and the union of all elements of \mathcal{Q}^d and their corresponding edges and vertices is the same domain $(-p, p)^d$ that is covered by each uniform mesh.

Definition 3.7 (skeleton). Given a mesh \mathcal{M}^d with elements \mathcal{Q}^d , denote the union of all (closed) element faces that are orthogonal to the first dimension by $\text{Sk}_1(\mathcal{M}^d) := \bigcup_{Q \in \mathcal{Q}^d} \text{Sk}_1(Q)$, with

$$\begin{aligned} \text{Sk}_1(Q) &:= \{x_1, x_1 + \tilde{x}_1\} \times [x_2, x_2 + \tilde{x}_2] \times \cdots \times [x_d, x_d + \tilde{x}_d] \\ &\text{for any } Q = (x_1, x_1 + \tilde{x}_1) \times \cdots \times (x_d, x_d + \tilde{x}_d) \in \mathcal{Q}^d. \end{aligned}$$

We call $\text{Sk}_1(\mathcal{M}^d)$ the *1-orthogonal skeleton*. Analogously, we denote the j -orthogonal skeleton by $\text{Sk}_j(\mathcal{M}^d)$ for all $j = 1, \dots, d$. Similarly to the above Definition, we abbreviate $\text{Sk}_j := \text{Sk}_j(\mathcal{M}^d)$ if only one mesh \mathcal{M}^d is considered in the respective context. Note that $\text{Sk}_1 \cap \cdots \cap \text{Sk}_d = \bigcup \mathcal{N}^d$.

Definition 3.8 (global index sets). For any $N = (x_1, \dots, x_d) \in \mathbb{R}^d$ and $j \in \{1, \dots, d\}$, we define

$$\begin{aligned} \mathcal{K}_j(N) &:= \left\{ z \in [-p, p] \mid (x_1, \dots, x_{j-1}, z, x_{j+1}, \dots, x_d) \in \text{Sk}_j \right\} \\ &\quad \cup \left\{ -p - \left\lceil \frac{p}{2} \right\rceil, \dots, -p - 1, p + 1, \dots, p + \left\lceil \frac{p}{2} \right\rceil \right\}. \quad (3) \end{aligned}$$

Definition 3.9 (local index vectors). To each node $N \in \mathcal{N}^d$ and each dimension $j = 1, \dots, d$, we associate a local index vector $\mathbf{k}_j(N) \in \mathbb{R}^{p+2}$, which consists of the unique $p+2$ consecutive elements in $\mathcal{K}_j(N)$ having the j -th component of N as their $\frac{p+3}{2}$ -th (this is, the middle) entry.

Definition 3.10 (B-spline). Given a local index vector $\mathbf{k}_j(N) = (k_1, \dots, k_{p+2})$, we denote by $B_{\mathbf{k}_j(N)}$ the standard 1D B-spline of degree p that corresponds to the knots k_1, \dots, k_{p+2} .

Definition 3.11 (T-spline). We associate to each node $N \in \mathcal{N}^d$ a multivariate B-spline, referred to as T-spline, defined as the product of the B-splines on the corresponding local index vectors,

$$B_N(x_1, \dots, x_d) := B_{\mathbf{k}_1(N)}(x_1) \cdots B_{\mathbf{k}_d(N)}(x_d).$$

We emphasize that given a structured d -dimensional T-mesh \mathcal{M}^d with elements \mathcal{Q}^d , faces \mathcal{E}^d , and nodes \mathcal{N}^d , the functions B_N , for $N \in \mathcal{N}^d$, are piecewise polynomials of degree p in each direction over $\text{BEZIER}(\mathcal{Q}^d)$ and C^{p-1} -smooth across all element interfaces $E \in \text{BEZIER}(\mathcal{E}^d)$.

In the following, we use such (classical) T-splines based on tensor-product meshes to construct appropriate spline spaces over unstructured, two-dimensional T-meshes. While we use two-dimensional tensor-product splines as basis functions in structured regions, we use k -dimensional tensor-product splines to construct basis functions around extraordinary nodes of valence k .

3.3. Unstructured T-spline spaces

Throughout this subsection, we consider a two-dimensional T-mesh \mathcal{M} as a control structure together with its Bézier mesh $\text{BEZIER}(\mathcal{M})$. In order to simplify the construction, we suppose the following.

Assumption 3.12. Let $N \in \mathcal{N}$ be an extraordinary node of the mesh. We assume that all nodes $N' \in \mathcal{N}$, with $N' \neq N$, that lie inside the $(3p+1)/2$ -disk of N , i.e., $N' \in \mathcal{N}(D_{(3p-1)/2}(N))$, are regular, interior nodes. Moreover, we assume that the p -disks of two extraordinary nodes do not overlap, i.e., for all $N_1, N_2 \in \mathcal{N}$ with $N_1 \neq N_2$ we have $D_p(N_1) \cap D_p(N_2) = \emptyset$.

The first condition implies that no knot line extensions extend into the p -disk of an extraordinary node and that all extraordinary nodes are sufficiently far away from the boundary. Thus, all elements in the p -disk of N are also elements of the Bézier mesh $\text{BEZIER}(\mathcal{M})$, i.e., $D_p(N) \subseteq \text{BEZIER}(\mathcal{Q})$. The second condition guarantees, that the supports of functions corresponding to different extraordinary nodes do not overlap. These assumptions are not essential for the theory presented in this paper, but significantly facilitate technical details. It might be necessary to refine the initial mesh in order to guarantee that extraordinary nodes are sufficiently separated. A construction of analysis-suitable unstructured T-splines over more general meshes, which do not satisfy Assumption 3.12, is presented in [21]. However, there the construction of functions around extraordinary nodes differs from the approach presented here.

Recall the unstructured spline space of Definition 3.4, which is piecewise polynomial of a certain degree p over a given mesh and satisfies continuity conditions across edges as prescribed by a function κ defined on the edges of the mesh. To describe the desired spline space properly, we need the following definition.

Definition 3.13 (edge prolongation). For an edge $E \in \mathcal{E}$, we define the *edge prolongation* of E as the set of edges that share a node but not an element with E , plus E itself, i.e.,

$$\text{ep}(E) := \{E' \in \mathcal{E} \mid \mathcal{N}(E) \cap \mathcal{N}(E') \neq \emptyset \wedge \mathcal{Q}(E) \cap \mathcal{Q}(E') = \emptyset\} \cup \{E\}$$

and for a set of edges $\mathcal{S} \subseteq \mathcal{E}$

$$\text{ep}(\mathcal{S}) := \bigcup \{\text{ep}(E) \mid E \in \mathcal{S}\}.$$

Further, the *edge prolongation of order i* is defined as

$$\text{ep}^i(\mathcal{S}) := \text{ep}^{i-1}(\text{ep}(\mathcal{S})) \text{ for } i > 1, \quad \text{ep}^1(\mathcal{S}) := \text{ep}(\mathcal{S}) \text{ for } i = 1.$$

In the following, we aim to construct unstructured T-splines from the following space.

Definition 3.14 (unstructured spline space). The unstructured spline space of degree p over the mesh $\text{BEZIER}(\mathcal{M})$ is given as

$$\mathcal{S}^p(\text{BEZIER}(\mathcal{M}), \kappa_0),$$

where the function κ_0 prescribing the continuity across edges of $\text{BEZIER}(\mathcal{E})$ takes the form

$$\kappa_0(E) = \begin{cases} 0 & \text{if } E \text{ is in } \text{ep}^{p-1}(\mathcal{E}(N)) \text{ for an extraordinary node } N, \\ p-1 & \text{otherwise} \end{cases}$$

for all $E \in \text{BEZIER}(\mathcal{E})$.

Due to $\mathcal{E} \subseteq \text{BEZIER}(\mathcal{E})$, we have $\text{ep}^{p-1}(\mathcal{E}(N)) \subset \text{BEZIER}(\mathcal{E})$, so the edges are all edges of the Bézier mesh as well and κ_0 is thus well-defined for all edges. The choice of the continuity function κ_0 for the space $\mathcal{S}^p(\text{BEZIER}(\mathcal{M}), \kappa_0)$ corresponds to edges in $\text{ep}^{(p-1)/2}(\mathcal{E}(N))$ having multiplicity p ,

when interpreted in classical T-spline notation, while all other edges have multiplicity 1. Then the resulting Bézier mesh is C^0 across all edges in $\text{ep}^{p-1}(\mathcal{E}(\mathcal{N}))$ and C^{p-1} across all other edges.

Providing a basis for the spline space of Definition 3.14 may not be feasible for general meshes. Thus, we construct a basis for a subspace. We split the construction as follows. We define spline basis functions with support inside the p -disk of any extraordinary node via traces of multivariate tensor-product spline basis functions on a higher-dimensional structured grid (cf. Definition 3.11 with $d > 2$). Basis functions that do not have their support included in the p -disk of an extraordinary node (i.e., basis functions that are associated with nodes outside the $(p+1)/2$ -disk of any extraordinary node) are defined via standard bivariate T-splines (cf. Definition 3.11 with $d = 2$).

Definition 3.15 (T-splines over an unstructured T-mesh). Let $\mathcal{M} = (\mathcal{Q}, \mathcal{E}, \mathcal{N})$ be a T-mesh. Let $\mathcal{N}_{\text{EO}} \subset \mathcal{N}$ be the set of all extraordinary nodes. We define the set of anchors as

$$\mathcal{N}_A = \mathcal{N} \setminus \bigcup_{N \in \mathcal{N}_{\text{EO}} \cup \mathcal{N}_{\text{bd}}} \{N' \in \mathcal{N} : N' \subset \mathcal{D}(D_{(p+1)/2}(N))\}.$$

For each anchor $N \in \mathcal{N}_A$, we define a basis function B_N as in Definition 3.17. For each extraordinary node $N \in \mathcal{N}_{\text{EO}}$, we define a set of functions \mathcal{B}_N as in Definition 3.21. This yields a global set of functions

$$\mathcal{B} = \left(\bigcup_{N \in \mathcal{N}_{\text{EO}}} \mathcal{B}_N \right) \cup \{B_N : N \in \mathcal{N}_A\}.$$

The functions given in Definition 3.21 are well-defined due to Assumption 3.12. As a result of Definition 3.15, there is a layer of elements around the boundary $\partial\Omega$, where the spline space is not complete, cf. Theorem 5.1. The functions given in Definition 3.17 for all refinements of a regular mesh \mathcal{M}_0 are well-defined if the following is satisfied.

Assumption 3.16. Let $N \in \mathcal{N}_A$ be an anchor of the initial mesh \mathcal{M}_0 . Then the elements in the $\frac{p+1}{2}$ -disk $D_{(p+1)/2}(N)$ form a structured submesh \mathcal{M}_N , which possesses the parameter domain $\mathcal{D}(\widehat{\mathcal{M}}_N) = \left(-\frac{p+1}{2}, \frac{p+1}{2}\right)^2$, where $\widehat{N} = (0, 0)^T$. Moreover, we assume that for any two anchors $N, N' \in \mathcal{N}_A$ the union $D_{(p+1)/2}(N) \cup D_{(p+1)/2}(N')$ forms a structured submesh as well.

On the initial mesh the support of the function corresponding to $N \in \mathcal{N}_A$ is given by $D_{(p+1)/2}(N)$. The second part of the assumption guarantees that the union of supports of two functions is contained inside a structured submesh. Consequently, the same is true for any two functions on refined meshes.

The functions in \mathcal{B} are all linearly independent, see Theorem 5.8, and they satisfy

$$\text{span}(\mathcal{B}) \subseteq \mathcal{S}^p(\text{BEZIER}(\mathcal{M}), \kappa_0). \quad (4)$$

This construction is particularly useful in the context of refinement since refinement routines for structured grids are applicable. Note that we do not necessarily require equality in (4). Although we have equality when the spaces are restricted to functions with support either completely outside the p -disk of an extraordinary node (by classical T-spline arguments) or completely inside (cf. Proposition 3.22), this does not need to be true in the transition domain. However, we do not explicitly demand equality in our construction and instead profit from its simplicity.

We emphasize that one can extend the definition of T-splines from unstructured T-meshes over planar partitions to mapped isogeometric functions. This is explained in more detail in C.

For all anchors inside the structured parts of the T-mesh we have the following.

Definition 3.17 (structured T-spline). Let $N \in \mathcal{N}_A$ be an anchor of the mesh \mathcal{M} . Then the function

$$B_N : \Omega \rightarrow \mathbb{R}$$

has support in a structured submesh \mathcal{M}_N of \mathcal{M} and is on its support defined as

$$B_N \circ \mathbf{F}_{\mathcal{Q}_N} = B_{\widehat{N}},$$

where $\mathbf{F}_{\mathcal{Q}_N} : \mathcal{D}(\widehat{\mathcal{M}}_N) \rightarrow \mathcal{D}(\mathcal{M}_N)$. Moreover, we have $\widehat{\mathcal{M}}_N$ being a structured two-dimensional T-mesh and \widehat{N} being a node of that structured T-mesh.

Note that it follows from Assumption 3.12 and from the definition of the T-mesh refinement that such a submesh \mathcal{M}_N exists for all $N \in \mathcal{N}_A$.

Next we define the basis functions around extraordinary nodes. As a first step of the construction, we define a localized embedding of the unstructured mesh into a higher-dimensional structured grid, which is based on the following observation: the p -disk $D_p(N)$ consists of k non-overlapping structured submeshes $\{\mathcal{M}_i\}_{i=1}^k$, in counterclockwise enumeration around N , that are separated by the prolonged edge-neighborhood of N , given by $\text{ep}^{p-1}(\mathcal{E}(N))$. That is, for $i \neq j$, we have that

$$\mathcal{D}(\mathcal{M}_i) \cap \mathcal{D}(\mathcal{M}_j) = \emptyset \quad \text{and} \quad \overline{\mathcal{D}(\mathcal{M}_i)} \cap \overline{\mathcal{D}(\mathcal{M}_j)} \subset \overline{\text{ep}^{p-1}(\mathcal{E}(N))}. \quad (5)$$

Since by Assumption 3.12, all nodes in $\mathcal{N}_1, \dots, \mathcal{N}_k$ except N are regular, each of the submeshes $\mathcal{M}_1, \dots, \mathcal{M}_k$ is formed by a regular grid of $p \times p$ elements.

Definition 3.18 (higher-dimensional embedding). Let N be an extraordinary node with valence $v_N = k$, and suppose that Assumption 3.12 holds. Then there exists an embedding $\text{emb}^k : \mathcal{D}(D_p(N)) \rightarrow [0, p]^k \subset \mathbb{R}^k$ that maps the elements, edges, and nodes of $D_p(N)$ onto faces, edges and nodes $(\mathcal{Q}^k, \mathcal{E}^k, \mathcal{N}^k)$ of the k -dimensional structured mesh \mathcal{M}^d with nodes in \mathbb{Z}^k and elements in

$$\left\{ \prod_{j=1}^k (z_j, z_j + 1) \mid z_1, \dots, z_k \in \mathbb{Z} \right\}.$$

The embedding emb^k is characterized by the following choices.

- $\text{emb}^k(N) = (0, \dots, 0) \in \mathbb{R}^k$.
- For $j = 0, \dots, k-1$, \mathcal{N}_j is mapped to $\{ae_j + be_{j+1} = (0, \dots, 0, a, b, 0, \dots, 0) \mid a, b \in \{0, \dots, p\}\}$, with e_j the j -th unit vector, $\mathcal{N}_0 = \mathcal{N}_k$ and $e_0 = e_k$.
- For $j = 1, \dots, k$, the edge $E_j \in \mathcal{E}(N) \cap \mathcal{E}_j \cap \mathcal{E}_{j-1}$ is mapped to $\text{emb}^k(E_j) = \{ae_j \mid 0 < a < 1\}$, with $\mathcal{E}_0 = \mathcal{E}_k$.
- The remaining edges in the prolonged edge neighborhood $E_j \in \text{ep}^{p-1}(\mathcal{E}(N)) \cap \mathcal{E}_j \cap \mathcal{E}_{j-1}$, for $j = 1, \dots, k$ and $\mathcal{E}_0 = \mathcal{E}_k$, are mapped to $\text{emb}^k(E_j) = \{ae_j \mid i < a < i+1\}$, for $i \in \{1, \dots, p-1\}$ depending on E_j such that emb^k is continuous.
- The elements $Q_j \in \mathcal{Q}_j$ are mapped to $\text{emb}^k(Q_j) = \{ae_j + be_{j+1} \mid h < a < h+1, i < b < i+1\}$ for $h, i \in \{1, \dots, p-1\}$ depending on Q_j such that emb^k is continuous.

Together, this defines a unique map emb^k (up to rotated enumeration of the submeshes $\mathcal{M}_1, \dots, \mathcal{M}_k$) and we set $(\mathcal{Q}^k, \mathcal{E}^k, \mathcal{N}^k) = (\text{emb}^k(D_p(N)), \text{emb}^k(\mathcal{E}(D_p(N))), \text{emb}^k(\mathcal{N}(D_p(N))))$.

As a next step, we define a spline space on the p -disk $D_p(N)$ of an extraordinary node N by employing the higher-dimensional representation of the mesh. Note that, as explained in Section 3.2, the higher-dimensional mesh \mathcal{M}^k induced by \mathbb{Z}^k allows for a classical definition of T-splines by means of a tensor-product construction. In this case the space is actually a standard, k -variate tensor-product B-spline space. For a clear distinction between bivariate splines associated with the unstructured mesh and multivariate splines associated with \mathcal{M}^k , we denote the latter T-spline basis functions corresponding to a node $N = \{\mathbf{n}\}$, $\mathbf{n} \in \mathbb{Z}^k$, by $B_N^k : \mathbb{R}^k \rightarrow \mathbb{R}$. The spline space in the original two-dimensional unstructured mesh is then constructed as follows.

Definition 3.19 (local T-spline space for an extraordinary node). Let $r = (p - 1)/2$ and let N be an extraordinary node and \mathbf{emb}^k its embedding into the k -dimensional mesh \mathcal{M}^k . Then, we define the set of unstructured splines around N by

$$\mathcal{S}_u(N) := \text{span}\{B_{N'}^k \circ \mathbf{emb}^k : \mathcal{D}(D_p(N)) \rightarrow \mathbb{R}, \quad N' \in ([-r, r] \cap \mathbb{Z})^k\}. \quad (6)$$

That is, these functions are defined as traces of higher-dimensional structured spline functions.

Remark 3.20 (Choice of the local T-spline space). We emphasize that the choice of the local construction defined in Definition 3.19 is not unique and other choices can be considered as well, e.g., the cubic unstructured T-spline constructions presented in [17, 20, 19, 21]. However, the construction needs to be compatible with the refinement routine presented in Section 4 in the sense that spline spaces may be defined on any of the meshes generated with the refinement algorithm without losing linear independence. For the above construction, this is naturally fulfilled due to the fact that both the spline construction and the refinement routine are motivated by a higher-dimensional representation of the local mesh.

Due to the nature of the embedding defined in Definition 3.18, the spline space $\mathcal{S}_u(N)$ is exactly a space of spline functions supported in the p -disk of the extraordinary node N that

- are C^{p-1} -continuous except for the prolonged edge-neighborhood $\text{ep}^{p-1}(\mathcal{E}(N))$, where the functions are in general only C^0 -continuous, and
- span those functions in $\mathcal{S}^p(\text{BEZIER}(\mathcal{M}), \kappa_0)$ which have their support contained in the neighborhood $\mathcal{D}(D_p(N))$ of the node N . Due to the smoothness properties, the functions vanish and have vanishing derivatives up to order $p - 1$ at the boundary of $\mathcal{D}(D_p(N))$.

These properties are formalized in Proposition 3.22. Note that for nodes in the unstructured two-dimensional mesh that are outside the $(p + 1)/2$ -disk of an extraordinary node, the corresponding spline basis function can be constructed as described in Section 3.2 directly for $d = 2$. For nodes inside the $(p + 1)/2$ -disk of an extraordinary node, we do not associate functions directly but rely on a basis construction derived from the embedding \mathbf{emb}^k . There exist several possibilities for the choice of corresponding basis functions that yield the space $\mathcal{S}_u(N)$ defined in (6). It is easy to see that the functions $B_{N'}^k \circ \mathbf{emb}^k$ for the entire index space $N' \in ([-r, r] \cap \mathbb{Z})^k$ are linearly dependent. In the following, we present a choice of linearly independent basis functions of the space $\mathcal{S}_u(N)$.

Definition 3.21 (basis functions near an extraordinary node). Let $r = (p - 1)/2$. We consider the set of anchors

$$\begin{aligned} \mathcal{I}_r^k = & \{(i, j, -r, -r, \dots, -r) : -r \leq i, j \leq r\} \\ & \cup \{(-r, i, j, -r, \dots, -r) : -r \leq i, j \leq r\} \\ & \cup \{(-r, -r, i, j, \dots, -r) : -r \leq i, j \leq r\} \\ & \cup \dots \cup \{(j, -r, \dots, -r, i) : -r \leq i, j \leq r\} \subset ([-r, r] \cap \mathbb{Z})^k. \end{aligned}$$

Note that the cardinality of the set \mathcal{I}_r^k is $k(4r^2 + 2r) + 1 = kp(p - 1) + 1$. Every node N in the set \mathcal{I}_r^k is an anchor for a standard, k -variate basis function B_N^k . The set of corresponding basis functions on the two-dimensional unstructured mesh is then given by

$$\mathcal{B}_N := \{B_N^k \circ \mathbf{emb}^k : N \in \mathcal{I}_r^k\}$$

and the space of unstructured spline functions is defined by

$$\mathcal{S}_u^*(N) := \text{span}\{b : b \in \mathcal{B}_N\}.$$

We have that $\mathcal{S}_u^*(N)$ reproduces the classical C^0 -continuous T-spline space based on knot and edge multiplicities of valence p in the prolonged edge-neighborhood $\text{ep}^{(p-1)/2}(\mathcal{E}(N))$ of the extraordinary node N . Therefore, the set of basis functions as described in Definition 3.21 is a natural choice for the spline space near the extraordinary node.

Proposition 3.22. *Let N be an extraordinary node. Then, the set of functions as given in Definition 3.21 forms a basis of the space $\mathcal{S}_u(N)$ as in Definition 3.19. We moreover have*

$$\mathcal{S}_u^*(N) = \mathcal{S}_u(N) = \{f \in \mathcal{S}^p(\text{BEZIER}(\mathcal{M}), \kappa_0) : \text{supp}(f) \subseteq \mathcal{D}(\mathcal{D}_p(N))\}.$$

Proof. We trivially have

$$\mathcal{S}_u^*(N) \subseteq \mathcal{S}_u(N) \subseteq \{f \in \mathcal{S}^p(\text{BEZIER}(\mathcal{M}), \kappa_0) : \text{supp}(f) \subseteq \mathcal{D}(\mathcal{D}_p(N))\},$$

since $\mathcal{I}_r^k \subset ([-r, r] \cap \mathbb{Z})^k$ and all functions B_N^k are C^{p-1} smooth piecewise polynomials and emb^k is C^∞ except for $\text{ep}^{p-1}(\mathcal{E}(N))$, where it is only C^0 . Moreover, the condition $\text{supp}(B_N^k \circ \text{emb}^k) \subseteq \mathcal{D}(\mathcal{D}_p(N))$ follows from $B_N^k \subset [-p, p]^k$ for all $N \in ([-r, r] \cap \mathbb{Z})^k$ and from the definition of emb^k .

As described before in Definition 3.18, let $\{\mathcal{M}_i\}_{i=1}^k$ be non-overlapping structured submeshes around the extraordinary node N . Each of these submeshes also defines a subdomain $\mathcal{D}(\mathcal{M}_i) \subset \Omega$. As before, $k = v_N$ is the valence of the extraordinary node.

The dimension of the space

$$\{f \in \mathcal{S}^p(\text{BEZIER}(\mathcal{M}), \kappa_0) : \text{supp}(f) \subseteq \mathcal{D}(\mathcal{D}_p(N))\} \quad (7)$$

equals $kp(p-1)+1$, which can be checked using a simple counting argument following the definition of standard C^0 multi-patch B-splines as, e.g., in [27, Section 3.3].

The last step consists in showing that the functions in \mathcal{B}_N are linearly independent. Due to $|\mathcal{B}_N| = kp(p-1)+1$, these functions then form a basis of the space (7) above.

To prove this, let $\mathcal{B}_N = \{b_\ell\}_{\ell=1}^{kp(p-1)+1}$ be the set of functions that are constructed as described in Definition 3.21. Each of these functions has support

- on $i \times j$ elements ($1 \leq i, j \leq p$) on one of the submeshes and whose boundary includes N and
- on $i \times 1$ or $1 \times j$ or 1×1 elements on all the remaining submeshes.

Based on the functions \mathcal{B}_N , we construct a set of functions $\tilde{\mathcal{B}}_N = \{\tilde{b}_\ell\}_{\ell=1}^{kp(p-1)+1}$ with the same cardinality, for which the linear independence is more apparent due to support arguments. To this end, we sketch the construction considering three different cases below.

Case 1: function centered at N . The function $\tilde{b}_1 = b_1$, which has support on exactly one element on each of the submeshes and corresponds to the index $(-r, -r, \dots, -r) \in \mathcal{I}_r^k$, is the only function in the modified set of functions that is supported on more than two submeshes and therefore linearly independent to the functions of Case 2 and 3.

Case 2: functions associated to nodes on $\text{ep}^r(\mathcal{E}(N))$. As a second case, we consider the functions $\{b_\ell\}_{\ell=2}^{kp(p-1)+1}$ that have support on exactly one element on all but two submeshes and have support on $i \times 1$ and $1 \times i$ elements ($2 \leq i \leq p$) on the remaining two submeshes, respectively. In the index notation, these functions are characterized by the indices $(i, -r, -r, \dots, -r)$, $(-r, i, -r, \dots, -r)$, \dots , $(-r, -r, -r, \dots, i) \in \mathcal{I}_r^k$ for $-r < i \leq r$.

Due to the definition of these functions and with the function from Case 1, we can subtract an appropriate multiple of b_1 to obtain functions that are only supported on the two submeshes with support on $i \times 1$ and $1 \times i$ elements, respectively. We denote them with $\{\tilde{b}_\ell\}_{\ell=2}^{kp(p-1)+1}$. These functions are linearly independent to each other since they all have different supports that cannot be obtained by the combination of the supports of other functions. Further, the functions from Case 2 are all linearly independent to the functions from Case 3 since all functions from Case 3 are only supported on one submesh.

Case 3: remaining functions. The remaining functions $\{b_\ell\}_{\ell=k(p-1)+2}^{kp(p-1)+1}$ have support on $i \times j$ elements ($2 \leq i, j \leq p$) on one (main) submesh, on $i \times 1$ and $1 \times j$ elements on the two edge-neighboring submeshes, respectively, and on one element on the remaining submeshes. These functions correspond to the remaining indices $(i, j, -r, \dots, -r)$, $(-r, i, j, \dots, -r)$, $\dots \in \mathcal{I}_r^k$ with $-r < i, j \leq r$ and can be modified to functions $\{\tilde{b}_\ell\}_{\ell=k(p-1)+2}^{kp(p-1)+1}$ whose supports lie within the $i \times j$ elements of only one submesh.

For a particular functions $b \in \{b_\ell\}_{\ell=k(p-1)+2}^{kp(p-1)+1}$, we achieve this by subtracting appropriate multiples of two functions in $\{\tilde{b}_\ell\}_{\ell=2}^{k(p-1)+1}$, namely those with overlap of $1 \times i$ and $j \times 1$ elements with the main submesh, respectively. To compensate for the overlap of these two functions, a multiple of b_1 has to be added to obtain functions that are only supported within one submesh. As above, the resulting functions all have different supports that cannot be obtained by an appropriate combination of the supports of the remaining functions.

In summary, the functions in $\tilde{\mathcal{B}}_N$ are linearly independent. Since they are constructed from the set \mathcal{B}_N with equal size, also the functions in \mathcal{B}_N are linearly independent, which completes the proof. \square

4. Refinement

The refinement algorithm makes use of the following key features:

1. an edge subdivision routine that allows for I-nodes, see Algorithm 2,
2. refinement levels of edges,
3. an edge neighborhood based on the mesh metric from Definition 2.16,
4. direction indices, i.e., integers associated to each edge as explained in Definition 4.3 below.

Algorithm 2 Subdivision of an edge

```

1: procedure SUBDIV( $\mathcal{M}, E = [a, b]$ ) ▷ the mesh and the edge to be subdivided
2:    $\text{mid}(E) \leftarrow \frac{a+b}{2}$ ;  $E_1 \leftarrow [a, \text{mid}(E)]$ ;  $E_2 \leftarrow [\text{mid}(E), b]$ 
3:    $\mathcal{N} \leftarrow \mathcal{N} \cup \{\text{mid}(E)\}$  ▷ update nodes
4:    $\mathcal{E} \leftarrow \mathcal{E} \cup \{E_1, E_2\}$  ▷ update edges
5:   mark  $E$  as inactive/refined
6:   for all  $Q \in \mathcal{Q}$  neighboring  $E$  do ▷ addresses at most two elements
7:      $E' \leftarrow$  opposite edge of  $E$  in  $Q$ 
8:     if  $E'$  has been subdivided then
9:        $\tilde{E} \leftarrow$  new edge connecting the midpoints of  $E$  and  $E'$ 
10:       $\{Q_1, Q_2\} \leftarrow$  children of  $Q$  being subdivided by  $\tilde{E}$ 
11:       $\mathcal{E} \leftarrow \mathcal{E} \cup \{\tilde{E}\}$ 
12:       $\mathcal{Q} \leftarrow \mathcal{Q} \cup \{Q_1, Q_2\}$ 
13:      mark  $Q$  as inactive/refined
14:     end if
15:   end for
16:   return  $\mathcal{M}$ 
17: end procedure

```

Definition 4.1 (refinement level). All edges of the initial mesh have the refinement level zero, written $\ell(E) = 0$ for all $E \in \mathcal{E}$. The levels of new edges created during refinement obey the following rules.

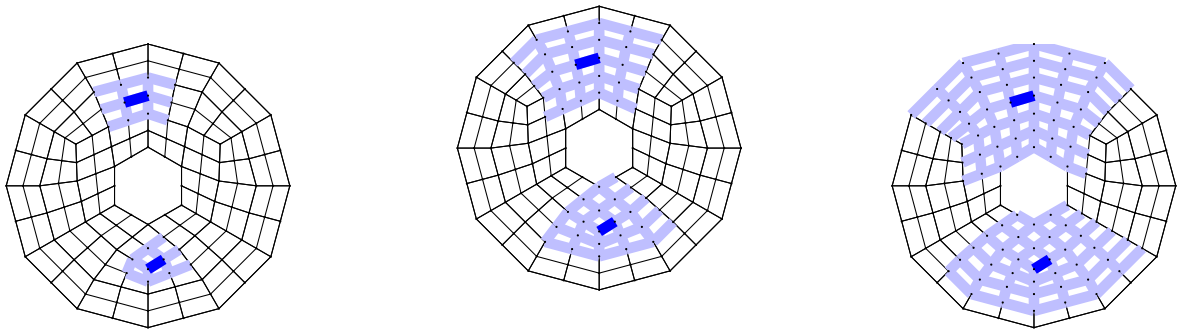


Figure 6: Examples for edge neighborhoods for $p = 1$ (left) and $p = 3$ (center) and $p = 5$ (right).

1. The children E_1, E_2 of an edge E as in Algorithm 2, line 2, have the level of their parent, increased by 1, i.e., $\ell(E_1) = \ell(E_2) = \ell(E) + 1$.
2. Other new edges such as \tilde{E} in Algorithm 2, line 9, have the same level as their opposite edges.

Definition 4.2 (edge neighborhood). Given a mesh \mathcal{M} , we define for each edge $E \in \mathcal{E}$ the neighborhood

$$\text{neighb}(\mathcal{M}, E) = \left\{ E' \in \mathcal{E} \mid \text{dist}(E, E') \leq \frac{p+1}{2} \cdot 2^{-\ell(E)} \right\},$$

where $\text{dist}(E, E')$ is the abbreviation for $\text{dist}(\text{mid}(E), \text{mid}(E'))$.

The main difficulty for T-splines on unstructured meshes is that there is no global notion of directions in the sense of horizontal and vertical edges. However, we can assign edge directions on each element and require consistency of these directions across the mesh. This is realized by integer-valued direction indices, which are defined below.

Definition 4.3 (direction index). Given a T-mesh \mathcal{M} , the *direction index* $\text{di}(E)$ of any edge $E \in \mathcal{E}$ is defined by a mapping $\text{di} : \mathcal{E} \rightarrow \mathbb{N}$ such that

$$\forall Q \in \mathcal{Q}, E_1, E_2 \in \mathcal{E}(Q) : \quad \text{di}(E_1) = \text{di}(E_2) \Leftrightarrow \overline{E_1} \cap \overline{E_2} = \emptyset. \quad (8)$$

This is, edges opposite to each other have identical direction indices, whereas neighboring edges in any element have distinct direction indices. Note, however, that $\text{di}(E_1) = \text{di}(E_2)$ and $\overline{E_1} \cap \overline{E_2} \neq \emptyset$ is allowed if there is no common element $Q \in \mathcal{Q}$ such that $\mathcal{E}(Q)$ contains both E_1 and E_2 , see Figure 7 for an illustration. We emphasize that the concept of direction indices is motivated by the different coordinate directions in higher dimensions. That is, if the unstructured mesh is obtained from traces of a higher-dimensional mesh, a direction index of an edge (locally) corresponds to the coordinate direction to which the corresponding edge in the higher-dimensional mesh is parallel.

Any map $\text{di} : \mathcal{E} \rightarrow \mathbb{N}$ satisfying (8) is called *direction labeling*, and its value $\text{di}(E)$ at $E \in \mathcal{E}$ is called direction index of E .

The direction indices of new edges created during refinement are defined through the rules

1. The children E_1, E_2 of an edge E as in Algorithm 2, line 2, have the same direction index as their parent.
2. Other new edges such as \tilde{E} in Algorithm 2, line 9, have the same direction index as their opposite edges.

For practical purposes, it is important to find a configuration of direction indices with a small number of distinct indices for a given unstructured mesh. We suggest the following approach: for any two edges E and E' , we write $E \sim E'$ if E and E' are opposite edges of a common neighboring cell.

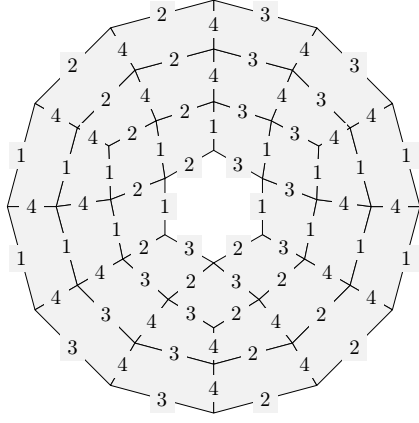


Figure 7: A direction labeling for the mesh from Figure 1.

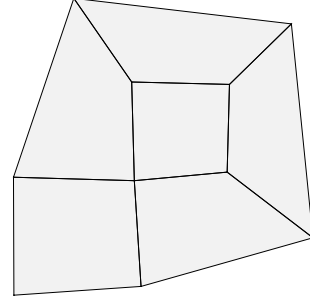


Figure 8: Example for a totally unstructured mesh.

In addition, we require that the relation \sim is reflexive and transitive, such that $E \sim E'$ is equivalent to the fact that the rule (8) enforces $\text{di}(E) = \text{di}(E')$. We draw an undirected graph, where each node represents an equivalence class $[E]$ (with $[E] = [E'] \Leftrightarrow E \sim E'$), and two nodes $[E]$, $[E']$ are connected if E and E' are adjacent edges of a common neighboring cell. This construction reduces our problem to a standard graph coloring problem, for which we can apply existing approaches, see, e.g., [28]. In the context of totally unstructured meshes, this approach is presented in more detail in B.

In the following, we assume that there exists a direction labeling. This is an important assumption, as there exist meshes for which no direction labeling is possible, since the graph constructed above may have self-connected nodes. According to [14, Definition 7.2.2], these meshes are called *totally unstructured*, see Figure 8 for an illustration and B for generalized direction indices on these meshes.

Our refinement algorithm is described in Algorithm 3 and example outputs are shown in Figure 9.

Algorithm 3 Refinement algorithm

```

1: procedure REFINE( $\mathcal{M}, E$ )
2:   while  $\exists E' \in \text{neighb}(\mathcal{M}, E) : \ell(E') < \ell(E)$ 
3: or  $\exists E' \in \text{neighb}(\mathcal{M}, E) : \ell(E') = \ell(E) \wedge \text{di}(E') < \text{di}(E)$  do
4:   for all  $E' \in \text{neighb}(\mathcal{M}, E)$  do
5:     if  $\ell(E') < \ell(E)$  then
6:        $\mathcal{M} \leftarrow \text{REFINE}(\mathcal{M}, E')$ 
7:     else if  $\ell(E') = \ell(E) \wedge \text{di}(E') < \text{di}(E)$  then
8:        $\mathcal{M} \leftarrow \text{REFINE}(\mathcal{M}, E')$ 
9:     end if
10:  end for
11:  end while
12:   $\mathcal{M} \leftarrow \text{SUBDIV}(\mathcal{M}, E)$ 
13:  return  $\mathcal{M}$ 
14: end procedure

```

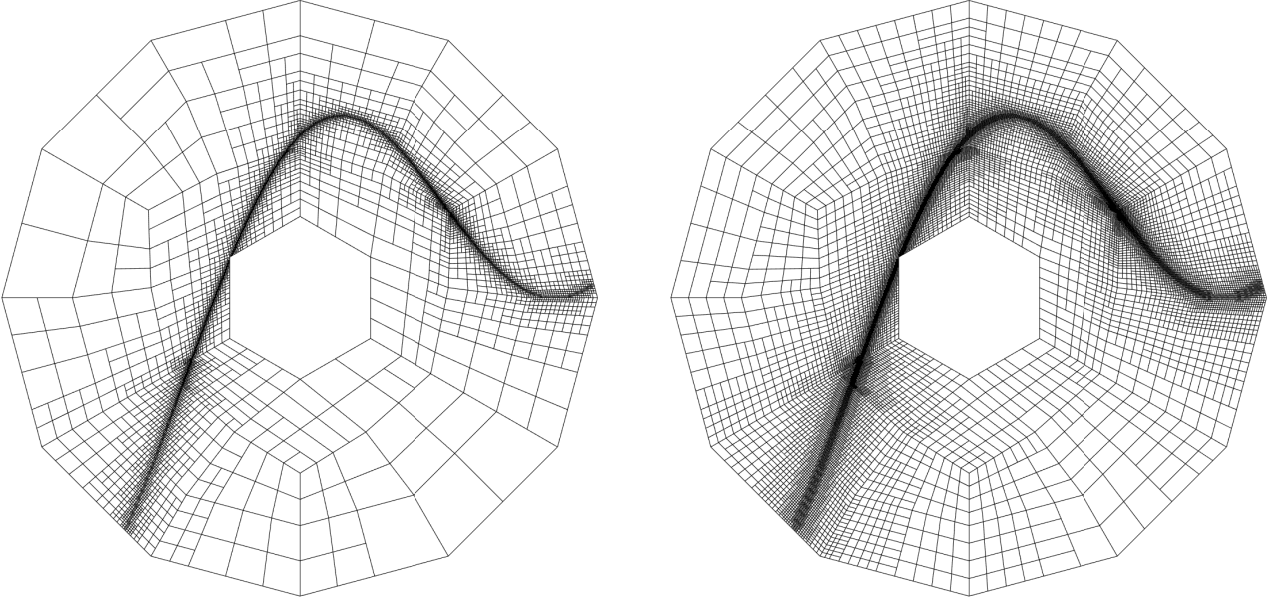


Figure 9: Refinement examples for $p = 1$ (left) and $p = 3$ (right), where the mesh is refined towards a given curve.

5. Properties of T-meshes and T-splines

By construction, the basis functions in \mathcal{B} , as given in Definition 3.15, have local support and satisfy (4), i.e.,

$$\text{span}(\mathcal{B}) \subseteq \mathcal{S}^p(\text{BEZIER}(\mathcal{M}), \kappa_0).$$

Moreover, we have the following.

Theorem 5.1. *For all elements $Q \in \mathcal{Q}$ that are sufficiently far away from the boundary of the mesh, we have that*

$$\text{span}(\mathcal{B})|_Q = \mathbb{P}^{(p,p)}(Q).$$

Proof. Recall the definition of \mathcal{B} as

$$\mathcal{B} = \left(\bigcup_{N \in \mathcal{N}_{\text{EO}}} \mathcal{B}_N \right) \cup \{B_N : N \in \mathcal{N}_A\}.$$

For each extraordinary node $N \in \mathcal{N}_{\text{EO}}$, the support of \mathcal{B}_N is given by $\mathcal{D}(\mathcal{D}_p(N))$. Due to Assumption 3.12, the $(3p - 1)/2$ -disk $\mathcal{D}_{(3p-1)/2}(N)$ around an extraordinary node N is composed of k non-overlapping structured submeshes $\{\mathcal{M}_i^*\}_{i=1}^k$. Note that the meshes \mathcal{M}_i^* contain the submeshes \mathcal{M}_i in (5).

We assume that $Q \in \mathcal{Q}$ is sufficiently far away from the boundary $\partial\Omega$, i.e., $Q \notin \mathcal{D}_p(N)$ for all boundary nodes $N \in \mathcal{N}_{\text{bd}}$. We distinguish two different cases. We either have

1. $Q \in \mathcal{D}_p(N)$ for exactly one extraordinary node $N \in \mathcal{N}_{\text{EO}}$, or
2. $Q \notin \mathcal{D}_p(N)$ for all extraordinary nodes $N \in \mathcal{N}_{\text{EO}}$.

Since the p -disks $\mathcal{D}_p(N)$ of all extraordinary nodes $N \in \mathcal{N}_{\text{EO}}$ are disjoint, these are the only possible cases.

Case 1: Let $Q \in \mathcal{D}_p(N)$ for an extraordinary node N . Let \mathcal{M}_j^* be the submesh containing Q . On \mathcal{M}_j^* the functions in

$$\mathcal{B}_N \cup \{B_{N'} : N' \in \mathcal{N}(\mathcal{D}_{(3p-1)/2}(N))\}$$

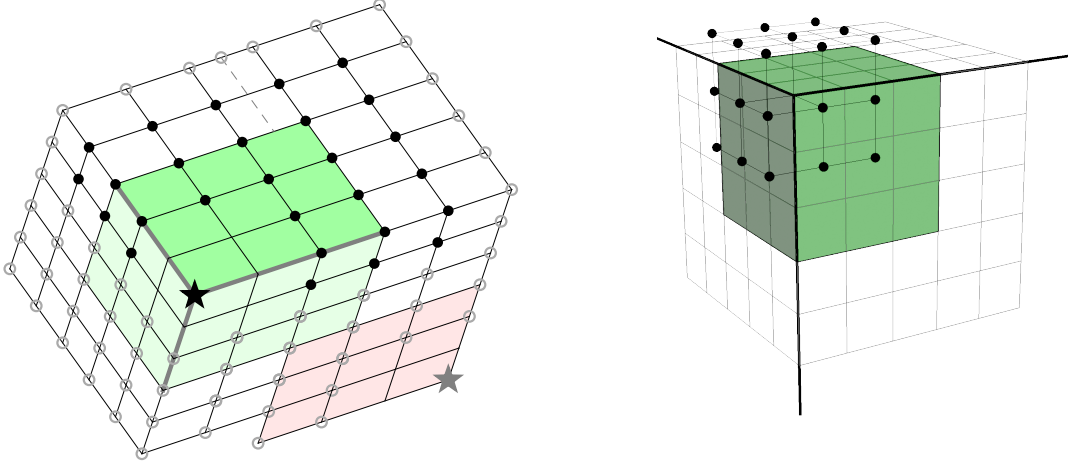


Figure 10: Left: An example of a neighborhood of an extraordinary node N for degree $p = 3$. The green region marks the support of \mathcal{B}_N , while the red region marks the support of $\mathcal{B}_{N'}$ for another extraordinary node N' . The anchors are marked with circles, where the full, black circles are denoting those anchors that correspond to functions having support in the darker green region. The thick lines mark the edges of C^0 -smoothness for functions in \mathcal{B}_N . Right: The higher-dimensional embedding and relevant basis functions, as in Definition 3.21. The anchors of basis functions are marked with black bullets. Their joint support, intersected with the coordinate planes, is marked in green.

form a standard tensor-product basis, since all nodes in $\mathcal{N}(\mathcal{D}_{(3p-1)/2}(N))$ are regular. Thus, they span all tensor-product polynomials of degree p on Q .

Case 2: The polynomial reproduction on Q follows from the standard T-spline construction, since all functions which have support on Q are standard T-splines from $\{B_N : N \in \mathcal{N}_A\}$.

This concludes the proof. \square

In Figure 10, we visualize the construction of the spaces around an extraordinary node. In this example, we have $p = 3$. In the left part, we show an example of a mesh around an extraordinary node N (marked with a black star). We assume that the depicted part of a mesh is sufficiently far away from the boundary. The green region marks the p -disk around the node, which corresponds to the union of supports of functions in \mathcal{B}_N . Similarly, the region marked in red represents the support of $\mathcal{B}_{N'}$ for another extraordinary node N' (marked with a gray star). The two extraordinary nodes satisfy that their corresponding p -disks (marked in green and red, respectively) do not overlap. Moreover, all nodes inside the $(3p + 1)/2$ -disk are regular. Thus, Assumption 3.12 is satisfied. However, nodes on the boundary of the $(3p + 1)/2$ -disk may be T-nodes, as indicated in the figure. The black and gray nodes represent the anchors of classical tensor-product T-spline basis functions which can be defined on a regular two-dimensional submesh. The construction of functions in \mathcal{B}_N that have support completely contained in the green region is visualized in the right part of Figure 10. More precisely, each black node represents an anchor from \mathcal{I}_r^k as in Definition 3.21, where $r = (p - 1)/2$ and $k = 3$. The supports of the corresponding k -dimensional basis functions are all contained in the green region, when intersected with the coordinate planes.

Note that polynomial reproduction is only guaranteed for elements that are sufficiently far away from the boundary. With minor modifications, generalizing B-splines on open knot vectors, the space can be adapted to obtain interpolatory B-splines at the boundary, which reproduce polynomials on all elements.





In the following subsections, we study properties of the proposed refinement algorithm. First, we show that the effect of the refinement algorithm is always local and of linear complexity and

after that we show that the resulting functions remain linearly independent.

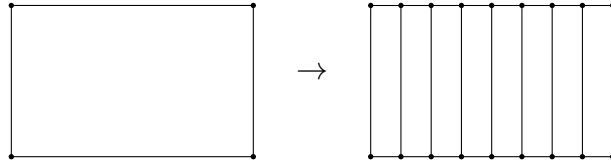
5.1. Linear complexity

In this subsection, we present two theorems that address the complexity of the refinement procedure. Theorem 5.2 states the existence of a uniform upper bound C on the distance between an edge called for refinement and any edge that is produced by the recursive refinement procedure. Theorem 5.3 states that there is a uniform upper bound C on the ratio between the accumulated numbers of produced and marked edges over the whole refinement history.

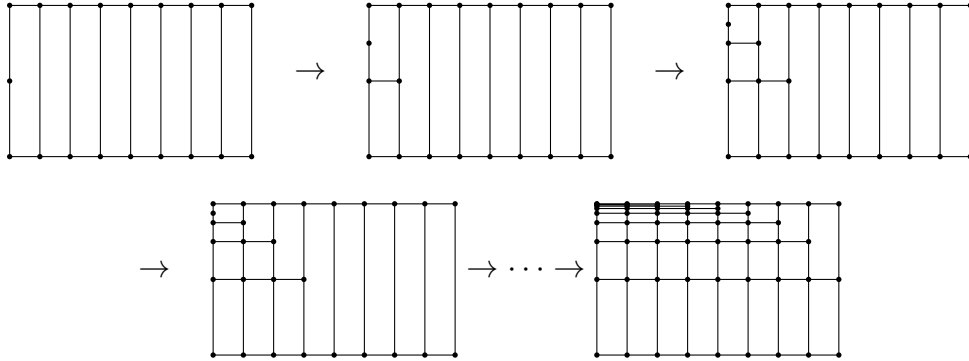
In order to indicate that this is not always given, we present here an example of a refinement strategy that does not have linear complexity, a scenario which – for computational reasons – should be avoided. Consider the following intuitive set of refinement rules for bilinear T-splines:

1. We use edge bisections as in Algorithm 2.
2. Any cell may contain at most one T-junction ( is ok,  is not ok).
3. If a refinement of an edge causes more than one T-junction on a neighbor element Q , then Q is refined in direction of the previously existing T-junction (this is, the opposite edge of the old T-junction is bisected such that  becomes .

We start with an initial mesh that is structured and contains a single cell, and we apply $2n \in \mathbb{N}$ bisections on horizontal edges, obtaining $n + 1$ cells and $3n + 4$ edges.



For $k = 1, \dots, n + 2$, we mark the vertical edge in the upper left corner for bisection, increasing the number of edges by $2k - 1$ in each step.



In total we have marked $\#\mathcal{E}_M = 3n + 2$ edges for refinement, increasing the number of edges by

$$\#\mathcal{E}_{2n+1} - \#\mathcal{E}_0 = 3n + \sum_{k=1}^{n+2} (2k - 1) = 3n + (n + 2)^2 = n^2 + 7n + 4. \quad (9)$$

For any given constant $C > 0$, we can choose $n = \lceil 3C \rceil$ in order to generate a sequence of refinements such that the ratio of generated and marked edges is

$$\frac{\#\mathcal{E}_{2n+1} - \#\mathcal{E}_0}{\#\mathcal{E}_M} = \frac{n^2 + 7n + 4}{3n + 2} > \frac{n}{3} \geq C. \quad (10)$$

This refinement strategy thus does not have linear complexity in the sense of Theorem 5.3.

Theorem 5.2 (locality of the refinement). *Let \mathbf{E} be an edge in the mesh \mathcal{M} , then any new edge \mathbf{E}' in $\text{REFINE}(\mathcal{M}, \mathbf{E}) \setminus \mathcal{M}$ satisfies*

$$\text{dist}(\mathbf{E}, \mathbf{E}') < C_{5.2} 2^{-\ell(\mathbf{E}')} \quad \text{and} \quad \ell(\mathbf{E}') \leq \ell(\mathbf{E}) + 1,$$

where $\text{dist}(\mathbf{E}, \mathbf{E}')$ is the abbreviation for $\text{dist}(\text{mid}(\mathbf{E}), \text{mid}(\mathbf{E}'))$, $K = \#\text{di}(\{\mathbf{E}_0, \dots, \mathbf{E}_J\})$ is the number of distinct direction indices occuring in the neighborhoods between \mathbf{E} and \mathbf{E}' , and

$$C_{5.2} = \frac{1}{2} + \frac{2}{2^{1/K}(2^{1/K}-1)} \cdot (p+1) < c_{5.2} K(p+1),$$

where $c_{5.2}$ does not depend on the mesh \mathcal{M} , K , or p .

Theorem 5.2 is a very common result in the theory of mesh-adaptive Galerkin schemes, see, e.g., [14, Lemmas 3.1.11 and 5.4.14] for Hierarchical B-splines and T-splines on structured meshes, or [29, Lemma 18] for the Newest Vertex Bisection on triangular meshes.

Proof. The existence of the edge \mathbf{E}' in $\text{REFINE}(\mathcal{M}, \mathbf{E}) \setminus \mathcal{M}$ means that Algorithm 3 has subdivided a sequence of edges $\mathbf{E} = \mathbf{E}_J, \mathbf{E}_{J-1}, \dots, \mathbf{E}_0$ with \mathbf{E}' being a child of \mathbf{E}_0 , thus $\ell(\mathbf{E}') = \ell(\mathbf{E}_0) + 1$, and with $\mathbf{E}_{j-1} \in \text{neighb}(\mathcal{M}, \mathbf{E}_j)$ and either $\ell(\mathbf{E}_{j-1}) < \ell(\mathbf{E}_j)$ or $\ell(\mathbf{E}_{j-1}) = \ell(\mathbf{E}_j) \wedge \text{di}(\mathbf{E}_{j-1}) < \text{di}(\mathbf{E}_j)$ for $j = 1, \dots, J$.

Let $K = \#\text{di}(\{\mathbf{E}_0, \dots, \mathbf{E}_J\})$ be the number of distinct direction indices in this sequence. Then, we have $\ell(\mathbf{E}_{j-K}) < \ell(\mathbf{E}_j)$, or equivalently, $\ell(\mathbf{E}_{j-K}) + 1 \leq \ell(\mathbf{E}_j)$, for any $j \in \{K, \dots, J\}$. Repeated application yields for any $j \in \{K, \dots, J\}$

$$\ell(\mathbf{E}_j) \geq \ell(\mathbf{E}_{j-K}) + 1 \geq \dots \geq \ell(\mathbf{E}_{j-\lfloor j/K \rfloor \cdot K}) + \lfloor \frac{j}{K} \rfloor \geq \ell(\mathbf{E}_0) + \frac{j-K+1}{K}. \quad (11)$$

Since dist is a metric, we have the triangle inequality

$$\text{dist}(\mathbf{E}', \mathbf{E}) \leq \text{dist}(\mathbf{E}', \mathbf{E}_0) + \text{dist}(\mathbf{E}_0, \mathbf{E}) \quad (12)$$

with

$$\text{dist}(\mathbf{E}', \mathbf{E}_0) = \frac{1}{2} \text{diam}(\mathbf{E}') = 2^{-\ell(\mathbf{E}')-1}, \quad \text{dist}(\mathbf{E}_0, \mathbf{E}) \leq \sum_{j=1}^J \text{dist}(\mathbf{E}_{j-1}, \mathbf{E}_j). \quad (13)$$

Since $\mathbf{E}_{j-1} \in \text{neighb}(\mathcal{M}, \mathbf{E}_j)$, we get by construction that $\text{dist}(\mathbf{E}_{j-1}, \mathbf{E}_j) \leq p \cdot 2^{-\ell(\mathbf{E}_j)}$ and hence

$$\begin{aligned} \sum_{j=1}^J \text{dist}(\mathbf{E}_{j-1}, \mathbf{E}_j) &\leq \sum_{j=1}^J \frac{p+1}{2} \cdot 2^{-\ell(\mathbf{E}_j)} \stackrel{(11)}{\leq} \sum_{j=1}^J (p+1) \cdot 2^{-1-\ell(\mathbf{E}_0)-(j-K+1)/K} \\ &= (p+1) \cdot 2^{-1-\ell(\mathbf{E}_0)+(K-1)/K} \sum_{j=1}^J (2^{-1/K})^j < (p+1) \cdot 2^{-(\ell(\mathbf{E}_0)+1)+(K-1)/K} \frac{2^{-1/K}}{1-2^{-1/K}} \\ &= (p+1) \cdot 2^{-\ell(\mathbf{E}')+(K-1)/K} \frac{2^{1/K}}{2^{2/K}-2^{1/K}} = (p+1) \cdot 2^{-\ell(\mathbf{E}')} \frac{2}{2^{1/K}(2^{1/K}-1)} \end{aligned}$$

The combination with (12) and (13) concludes the distance estimate. The application of (11) to the case $j = J \geq 0$, together with $\ell(\mathbf{E}_0) = \ell(\mathbf{E}') - 1$ from above, yields

$$\ell(\mathbf{E}) \geq \ell(\mathbf{E}_0) + \frac{J-K+1}{K} = \ell(\mathbf{E}') - 2 + \frac{J+1}{K},$$

hence $\ell(\mathbf{E}') \leq \ell(\mathbf{E}) + 2 - \frac{J+1}{K}$. Since $\frac{J+1}{K} > 0$ and $\ell(\mathbf{E}'), \ell(\mathbf{E})$ are integer numbers, we get $\ell(\mathbf{E}') \leq \ell(\mathbf{E}) + 1$. Thus, we obtain

$$\text{dist}(\mathbf{E}, \mathbf{E}') < C_{5.2} 2^{-\ell(\mathbf{E}')}$$

with

$$C_{5.2} = \frac{1}{2} + \frac{2}{2^{1/K}(2^{1/K}-1)} \cdot (p+1).$$

To conclude the proof, we show that

$$\frac{C_{5.2}-1/2}{K(p+1)} = \frac{2}{K2^{1/K}(2^{1/K}-1)} \rightarrow \frac{2}{\ln(2)}$$

for $K \rightarrow \infty$. To do so, we replace K by $x = 1/K$ and obtain

$$\lim_{x \rightarrow 0} \frac{2x}{2^x(2^x-1)} = \lim_{x \rightarrow 0} \frac{2}{2^{2x} \ln(2) + 2^{2x}(2^x-1) \ln(2)} = \frac{2}{\ln(2)},$$

using de L'Hôpital's rule. Hence, $C_{5.2} \rightarrow_{K \rightarrow \infty} \frac{1}{2} + \frac{2}{\ln(2)} K(p+1)$ and the statement follows. \square

Theorem 5.3. *Let $\mathcal{E}_M = \{E_0, \dots, E_{J-1}\}$ be the set of marked edges used to generate the sequence of T -meshes $\mathcal{M}_0, \mathcal{M}_1, \dots, \mathcal{M}_J$ with $\mathcal{M}_j = (\mathcal{Q}_j, \mathcal{E}_j, \mathcal{N}_j)$ starting from \mathcal{M}_0 , namely*

$$\mathcal{M}_j = \text{REFINE}(\mathcal{M}_{j-1}, E_{j-1}), \quad E_{j-1} \in \mathcal{E}_{j-1} \quad \text{for } j \in \{1, \dots, J\}.$$

Then, there exists a positive constant $C_{5.3}$ such that

$$\#\mathcal{E}_J - \#\mathcal{E}_0 \leq C_{5.3} \#\mathcal{E}_M.$$

Proof. We denote by $\tilde{\mathcal{E}} = \bigcup_{\mathcal{M} \text{ refinement of } \mathcal{M}_0} \mathcal{E}$ the set of all edges in the initial mesh \mathcal{M}_0 and in all of its possible refinements. For any $E' \in \tilde{\mathcal{E}}$ and $E \in \mathcal{E}_M$, let

$$\lambda(E', E) := \begin{cases} 2^{\ell(E')-\ell(E)} & \text{if } \ell(E') \leq \ell(E) + 1 \text{ and } \text{dist}(E', E) < 2^{1-\ell(E')} C_{5.2}, \\ 0 & \text{otherwise.} \end{cases}$$

The proof consists of two main steps devoted to identify

(i) a lower bound for the sum of the λ function as E varies in \mathcal{E}_M so that each $E' \in \mathcal{E}_J \setminus \mathcal{E}_0$ satisfies

$$\sum_{E \in \mathcal{E}_M} \lambda(E', E) \geq 1, \tag{14}$$

(ii) an upper bound for the sum of the λ function as the refined element E' varies in $\mathcal{E}_J \setminus \mathcal{E}_0$ so that for any $j = 0, \dots, J-1$

$$\sum_{E' \in \mathcal{E}_J \setminus \mathcal{E}_0} \lambda(E', E_j) \leq C_{5.3} \tag{15}$$

holds.

If inequalities (14) and (15) hold for a certain constant $C_{5.3}$, we have

$$\begin{aligned} \#\mathcal{E}_J - \#\mathcal{E}_0 &\leq \#(\mathcal{E}_J \setminus \mathcal{E}_0) = \sum_{E' \in \mathcal{E}_J \setminus \mathcal{E}_0} 1 \\ &\leq \sum_{E' \in \mathcal{E}_J \setminus \mathcal{E}_0} \sum_{E \in \mathcal{E}_M} \lambda(E', E) \leq \sum_{E \in \mathcal{E}_M} C_{5.3} = C_{5.3} \#\mathcal{E}_M, \end{aligned}$$

and the proof of the theorem is complete. We detail below the proofs of (i) and (ii).

(i) Let $E' \in \mathcal{E}_J \setminus \mathcal{E}_0$ be an edge generated in the refinement process from \mathcal{M}_0 to \mathcal{M}_J , and let $j_1 < J$ be the unique index such that $E' \in \mathcal{E}_{j_1+1} \setminus \mathcal{E}_{j_1}$. Theorem 5.2 states that

$$\text{dist}(E', E_{j_1}) \leq 2^{-\ell(E')} C_{5.2} \quad \text{and} \quad \ell(E') \leq \ell(E_{j_1}) + 1,$$

and consequently $\lambda(\mathbf{E}', \mathbf{E}_{j_1}) = 2^{\ell(\mathbf{E}') - \ell(\mathbf{E}_{j_1})} > 0$. The repeated use of Lemma 5.2 yields a sequence $\{\mathbf{E}_{j_2}, \mathbf{E}_{j_3}, \dots\}$ with $\mathbf{E}_{j_{i-1}} \in \mathcal{E}_{j_{i+1}} \setminus \mathcal{E}_{j_i}$, for $j_1 > j_2 > j_3 > \dots$, such that

$$\text{dist}(\mathbf{E}_{j_{i-1}}, \mathbf{E}_{j_i}) \leq 2^{-\ell(\mathbf{E}_{j_{i-1}})} C_{5.2} \quad \text{and} \quad \ell(\mathbf{E}_{j_{i-1}}) \leq \ell(\mathbf{E}_{j_i}) + 1. \quad (16)$$

We iteratively apply Theorem 5.2 as long as

$$\lambda(\mathbf{E}', \mathbf{E}_{j_i}) > 0 \quad \text{and} \quad \ell(\mathbf{E}_{j_i}) > 0,$$

until we reach the first index L with $\lambda(\mathbf{E}', \mathbf{E}_{j_L}) = 0$ or $\ell(\mathbf{E}_{j_L}) = 0$. By considering the three possible cases below, inequality (14) may be derived as follows.

- If $\ell(\mathbf{E}_{j_L}) = 0$ and $\lambda(\mathbf{E}', \mathbf{E}_{j_L}) > 0$, then

$$\sum_{\mathbf{E} \in \mathcal{E}_M} \lambda(\mathbf{E}', \mathbf{E}) \geq \lambda(\mathbf{E}', \mathbf{E}_{j_L}) = 2^{\ell(\mathbf{E}') - \ell(\mathbf{E}_{j_L})} > 1$$

since $\ell(\mathbf{E}') > \ell(\mathbf{E}_{j_L}) = 0$.

- If $\lambda(\mathbf{E}', \mathbf{E}_{j_L}) = 0$ because $\ell(\mathbf{E}') > \ell(\mathbf{E}_{j_L}) + 1$, then (16) yields $\ell(\mathbf{E}_{j_{L-1}}) \leq \ell(\mathbf{E}_{j_L}) + 1 < \ell(\mathbf{E}')$ and hence

$$\sum_{\mathbf{E} \in \mathcal{E}_M} \lambda(\mathbf{E}', \mathbf{E}) \geq \lambda(\mathbf{E}', \mathbf{E}_{j_{L-1}}) = 2^{\ell(\mathbf{E}') - \ell(\mathbf{E}_{j_{L-1}})} > 1.$$

- If $\lambda(\mathbf{E}', \mathbf{E}_{j_L}) = 0$ because $\text{dist}(\mathbf{E}', \mathbf{E}_{j_L}) \geq 2^{1-\ell(\mathbf{E}')} C_{5.2}$, then a triangle inequality combined with Theorem 5.2 leads to

$$2^{1-\ell(\mathbf{E}')} C_{5.2} \leq \text{dist}(\mathbf{E}', \mathbf{E}_{j_1}) + \sum_{i=1}^{L-1} \text{dist}(\mathbf{E}_{j_i}, \mathbf{E}_{j_{i+1}}) \leq 2^{-\ell(\mathbf{E}')} C_{5.2} + \sum_{i=1}^{L-1} 2^{-\ell(\mathbf{E}_{j_i})} C_{5.2}.$$

Consequently, $2^{-\ell(\mathbf{E}')} \leq \sum_{i=1}^{L-1} 2^{-\ell(\mathbf{E}_{j_i})}$ and we obtain

$$1 \leq \sum_{i=1}^{L-1} 2^{\ell(\mathbf{E}') - \ell(\mathbf{E}_{j_i})} = \sum_{i=1}^{L-1} \lambda(\mathbf{E}', \mathbf{E}_{j_i}) \leq \sum_{\mathbf{E} \in \mathcal{E}_M} \lambda(\mathbf{E}', \mathbf{E}).$$

(ii) Inequality (15) can be derived as follows. For any $0 \leq j \leq J-1$, we consider the set of edges of level j whose distance from \mathbf{E} is less than $2^{1-j} C_{5.2}$, defined as

$$A(\mathbf{E}, j) := \{\mathbf{E}' \in \mathcal{E}(\mathcal{M}_{u[j]}^d) \mid \text{dist}(\mathbf{E}', \mathbf{E}) < 2^{1-j} C_{5.2}\}.$$

According to the definition of λ , the set $A(\mathbf{E}, j)$ contains all elements at level j that satisfy $\lambda(\mathbf{E}', \mathbf{E}) > 0$. We have

$$\sum_{\mathbf{E}' \in \mathcal{E}_J \setminus \mathcal{E}_0} \lambda(\mathbf{E}', \mathbf{E}) \leq \sum_{\mathbf{E}' \in \bigcup_{j \in \mathbb{N}} \mathcal{E}(\mathcal{M}_{u[j]}^d) \setminus \mathcal{E}_0} \lambda(\mathbf{E}', \mathbf{E}) = \sum_{j=1}^{\ell(\mathbf{E})+1} 2^{j-\ell(\mathbf{E})} \#A(\mathbf{E}, j). \quad (17)$$

If $j < \log_2(2C_{5.2}/p)$, then $\#A(\mathbf{E}, j)$ is bounded by $\#A(\mathbf{E}, j) \leq 4^j \#A(\mathbf{E}, 0)$, where $\#A(\mathbf{E}, 0)$ depends only on the initial mesh, in particular on the number of distinct direction indices, and on the number, locations, and valences of extraordinary nodes.

If on the other hand $j \geq \log_2(2C_{5.2}/p)$, we observe that the neighborhood of any edge $\mathbf{E} \in \mathcal{E}(\mathcal{M}_{u[j]}^d)$ contains at most one extraordinary node v_E . We set $\text{val}(\mathbf{E}) = \max(4, \text{valence of } v_E)$ and observe that $\#A(\mathbf{E}, j) \leq 2 \cdot (2C_{5.2} + 1)^2 \cdot \text{val}(\mathbf{E})$.

Together, we find that

$$\#A(\mathbf{E}, j) \leq \max\left\{4^{\log_2(2C_{5.2}/p)} \#A(\mathbf{E}, 0), 2 \cdot (2C_{5.2} + 1)^2 \cdot \text{val}(\mathbf{E})\right\} =: C(\mathbf{E}).$$

Finally, the index substitution $k := 1 - j + \ell(\mathbf{E})$ reduces (17) to

$$\begin{aligned} \sum_{\mathbf{E}' \in \mathcal{E}_j \setminus \mathcal{E}_0} \lambda(\mathbf{E}', \mathbf{E}) &\leq \sum_{j=1}^{\ell(\mathbf{E})+1} 2^{j-\ell(\mathbf{E})} \#A(\mathbf{E}, j) = \sum_{k=0}^{\ell(\mathbf{E})} 2^{1-k} \#A(\mathbf{E}, j) \\ &< 2 \sum_{k=0}^{\infty} 2^{-k} \#A(\mathbf{E}, j) \leq 4 \max_{\mathbf{E} \in \bigcup_{j \in \mathbb{N}} \mathcal{E}(\mathcal{M}_{u[j]}^d) \setminus \mathcal{E}_0} C(\mathbf{E}) =: C_{5.3}, \end{aligned}$$

which completes the proof. \square

With Theorems 5.2 and 5.3, we have that the refinement obtained by Algorithm 3 remains local and exhibits linear complexity.

5.2. Linear independence

In this subsection, we show that the T-spline basis functions defined over a refined T-mesh are linearly independent.

Theorem 5.4 (quasi-uniformity of generated meshes). *For any T-mesh $\mathcal{M} = (\mathcal{N}, \mathcal{E}, \mathcal{Q})$ generated by REFINE (see Algorithm 3), the edge levels in any edge neighborhood are bounded from above and below by*

$$\forall \mathbf{E} \in \mathcal{E}, \mathbf{E}' \in \text{neighb}(\mathcal{M}, \mathbf{E}) : \begin{cases} \ell(\mathbf{E}) \leq \ell(\mathbf{E}') \leq \ell(\mathbf{E}) + 1 & \text{if } \text{di}(\mathbf{E}') < \text{di}(\mathbf{E}) \\ \ell(\mathbf{E}) - 1 \leq \ell(\mathbf{E}') \leq \ell(\mathbf{E}) + 1 & \text{if } \text{di}(\mathbf{E}') = \text{di}(\mathbf{E}) \\ \ell(\mathbf{E}) - 1 \leq \ell(\mathbf{E}') \leq \ell(\mathbf{E}) & \text{if } \text{di}(\mathbf{E}') > \text{di}(\mathbf{E}). \end{cases} \quad (18)$$

Proof. We prove the theorem by induction over admissible subdivisions. We call $\check{\mathcal{M}} = \text{SUBDIV}(\mathcal{M}, \mathbf{E})$ an *admissible subdivision* if

$$\forall \mathbf{E}' \in \text{neighb}(\mathcal{M}, \mathbf{E}) : \begin{cases} \ell(\mathbf{E}') > \ell(\mathbf{E}) & \text{if } \text{di}(\mathbf{E}') < \text{di}(\mathbf{E}) \\ \ell(\mathbf{E}') \geq \ell(\mathbf{E}) & \text{if } \text{di}(\mathbf{E}') \geq \text{di}(\mathbf{E}). \end{cases} \quad (19)$$

Note that REFINE always performs a sequence of admissible subdivisions. Our approach is therefore appropriate for proving the theorem for all T-meshes.

The claim is true for the initial mesh \mathcal{M}_0 since all edges $\mathbf{E}, \mathbf{E}' \in \mathcal{E}_0$ satisfy $\ell(\mathbf{E}) = \ell(\mathbf{E}') = 0$. For the induction proof, we consider a T-mesh \mathcal{M} that satisfies (18) and $\mathbf{E} \in \mathcal{E}$ satisfying (19). Together, this yields

$$\forall \mathbf{E}' \in \text{neighb}(\mathcal{M}, \mathbf{E}) : \begin{cases} \ell(\mathbf{E}') = \ell(\mathbf{E}) + 1 & \text{if } \text{di}(\mathbf{E}') < \text{di}(\mathbf{E}) \\ \ell(\mathbf{E}) \leq \ell(\mathbf{E}') \leq \ell(\mathbf{E}) + 1 & \text{if } \text{di}(\mathbf{E}') = \text{di}(\mathbf{E}) \\ \ell(\mathbf{E}') = \ell(\mathbf{E}) & \text{if } \text{di}(\mathbf{E}') > \text{di}(\mathbf{E}). \end{cases} \quad (20)$$

We aim to show that $\check{\mathcal{M}} = \text{SUBDIV}(\mathcal{M}, \mathbf{E})$ also fulfills the claim. The subdivision of \mathbf{E} removes the edge \mathbf{E} and adds two children $\mathbf{E}_1, \mathbf{E}_2$ to the mesh, and maybe $\tilde{\mathbf{E}}$ or $\tilde{\mathbf{E}}_1, \tilde{\mathbf{E}}_2$ if one or two neighboring quadrilaterals are subdivided. We will below show the claim for

- (i) $\mathbf{E}_1, \mathbf{E}_2$,
- (ii) $\mathbf{E}' \in \check{\mathcal{E}} \cap \mathcal{E}$ with $\mathbf{E} \in \text{neighb}(\mathcal{M}, \mathbf{E}')$,
- (iii) $\tilde{\mathbf{E}}_1, \tilde{\mathbf{E}}_2$, and
- (iv) $\mathbf{E}' \in \check{\mathcal{E}} \cap \mathcal{E}$ with $\mathbf{E} \notin \text{neighb}(\mathcal{M}, \mathbf{E}')$.

(i) Any edge $\mathbf{E}' \in \text{neighb}(\check{\mathcal{M}}, \mathbf{E}_1)$ satisfies

$$\text{dist}(\mathbf{E}', \mathbf{E}) \leq \text{dist}(\mathbf{E}', \mathbf{E}_1) + \text{dist}(\mathbf{E}_1, \mathbf{E}) \leq \frac{p+1}{2} \cdot 2^{-\ell(\mathbf{E}_1)} + 2^{-1-\ell(\mathbf{E}_1)} \leq \frac{p+1}{2} \cdot 2^{-\ell(\mathbf{E})}$$

since $\frac{p+1}{2} \geq \frac{1}{2}$. Hence all edges $\mathbf{E}' \in \text{neighb}(\check{\mathcal{M}}, \mathbf{E}_1) \cap \mathcal{E}$ are in $\text{neighb}(\mathcal{M}, \mathbf{E})$ and satisfy (20). With $\ell(\mathbf{E}_1) = \ell(\mathbf{E}) + 1$ and $\text{di}(\mathbf{E}_1) = \text{di}(\mathbf{E})$, we can rewrite this as

$$\forall \mathbf{E}' \in \text{neighb}(\check{\mathcal{M}}, \mathbf{E}_1) \cap \mathcal{E} : \begin{cases} \ell(\mathbf{E}') = \ell(\mathbf{E}_1) & \text{if } \text{di}(\mathbf{E}') < \text{di}(\mathbf{E}_1) \\ \ell(\mathbf{E}_1) - 1 \leq \ell(\mathbf{E}') \leq \ell(\mathbf{E}_1) & \text{if } \text{di}(\mathbf{E}') = \text{di}(\mathbf{E}_1) \\ \ell(\mathbf{E}') = \ell(\mathbf{E}_1) - 1 & \text{if } \text{di}(\mathbf{E}') > \text{di}(\mathbf{E}_1). \end{cases} \quad (21)$$

For the remaining edges $\mathbf{E}' \in \text{neighb}(\check{\mathcal{M}}, \mathbf{E}_1) \setminus \mathcal{E}$, which are $\mathbf{E}_1, \mathbf{E}_2, \tilde{\mathbf{E}}_1, \tilde{\mathbf{E}}_2$ from above, we know that $\ell(\mathbf{E}_2) = \ell(\mathbf{E}_1) = \ell(\mathbf{E}) + 1$ and that $\tilde{\mathbf{E}}_1, \tilde{\mathbf{E}}_2$ have the same level and direction index as their opposite edges (see Definitions 4.3 and 4.1), at least one of which is in $\text{neighb}(\check{\mathcal{M}}, \mathbf{E}_1) \cap \mathcal{E}$ and satisfies (21). Hence all new edges also satisfy (21) and the claim is fulfilled by \mathbf{E}_1 , and similarly by \mathbf{E}_2 .

(ii) Let $\mathbf{E}' \in \check{\mathcal{E}} \cap \mathcal{E}$ with $\mathbf{E} \in \text{neighb}(\mathcal{M}, \mathbf{E}')$. The induction hypothesis yields that

$$\forall \mathbf{E}'' \in \text{neighb}(\mathcal{M}, \mathbf{E}') : \begin{cases} \ell(\mathbf{E}') \leq \ell(\mathbf{E}'') \leq \ell(\mathbf{E}') + 1 & \text{if } \text{di}(\mathbf{E}'') < \text{di}(\mathbf{E}') \\ \ell(\mathbf{E}') - 1 \leq \ell(\mathbf{E}'') \leq \ell(\mathbf{E}') + 1 & \text{if } \text{di}(\mathbf{E}'') = \text{di}(\mathbf{E}') \\ \ell(\mathbf{E}') - 1 \leq \ell(\mathbf{E}'') \leq \ell(\mathbf{E}') & \text{if } \text{di}(\mathbf{E}'') > \text{di}(\mathbf{E}'). \end{cases} \quad (22)$$

Since \mathbf{E}' is unchanged, all old neighbor edges $\mathbf{E}'' \in \text{neighb}(\check{\mathcal{M}}, \mathbf{E}') \cap \text{neighb}(\mathcal{M}, \mathbf{E}')$ satisfy (22) also in the new mesh \mathcal{M} . The set of new neighbor edges $\text{neighb}(\check{\mathcal{M}}, \mathbf{E}') \setminus \text{neighb}(\mathcal{M}, \mathbf{E}')$ may contain at most $\mathbf{E}_1, \mathbf{E}_2, \tilde{\mathbf{E}}_1$ and $\tilde{\mathbf{E}}_2$, and it contains at least one of the children $\mathbf{E}_1, \mathbf{E}_2$. Since (22) also holds for $\mathbf{E}'' = \mathbf{E}$ with $\ell(\mathbf{E}_1) = \ell(\mathbf{E}) + 1$ and $\text{di}(\mathbf{E}_1) = \text{di}(\mathbf{E})$, we have

$$\begin{aligned} \ell(\mathbf{E}') + 1 &\leq \ell(\mathbf{E}_1) \leq \ell(\mathbf{E}') + 2 & \text{if } \text{di}(\mathbf{E}_1) < \text{di}(\mathbf{E}') \\ \ell(\mathbf{E}') &\leq \ell(\mathbf{E}_1) \leq \ell(\mathbf{E}') + 2 & \text{if } \text{di}(\mathbf{E}_1) = \text{di}(\mathbf{E}') \\ \ell(\mathbf{E}') &\leq \ell(\mathbf{E}_1) \leq \ell(\mathbf{E}') + 1 & \text{if } \text{di}(\mathbf{E}_1) > \text{di}(\mathbf{E}'). \end{aligned}$$

Assume for contradiction that $\text{di}(\mathbf{E}_1) \leq \text{di}(\mathbf{E}')$ and $\ell(\mathbf{E}_1) = \ell(\mathbf{E}') + 2$ or that $\ell(\mathbf{E}_1) = \ell(\mathbf{E}') + 1$ and $\text{di}(\mathbf{E}_1) > \text{di}(\mathbf{E}')$. We conclude that $\ell(\mathbf{E}) = \ell(\mathbf{E}') + 1$ or $\ell(\mathbf{E}) = \ell(\mathbf{E}')$, respectively, and together with $\mathbf{E} \in \text{neighb}(\mathcal{M}, \mathbf{E}')$ that $\text{dist}(\mathbf{E}, \mathbf{E}') \leq \frac{p+1}{2} \cdot 2^{-\ell(\mathbf{E}')} \leq \frac{p+1}{2} \cdot 2^{-\ell(\mathbf{E})}$ and hence $\mathbf{E}' \in \text{neighb}(\mathcal{M}, \mathbf{E})$. Thus, \mathbf{E}' satisfies (20) which leads to contradiction in both cases. This shows the claim for $\mathbf{E}_1 \in \text{neighb}(\check{\mathcal{M}}, \mathbf{E}')$, and similarly for \mathbf{E}_2 .

Consider $\tilde{\mathbf{E}} \in \{\tilde{\mathbf{E}}_1, \tilde{\mathbf{E}}_2\}$, which has two opposite edges with the same refinement level and direction index. At least one of these opposite edges is in $\text{neighb}(\check{\mathcal{M}}, \mathbf{E}') \cap \text{neighb}(\mathcal{M}, \mathbf{E}')$ and satisfies (22), and so does $\tilde{\mathbf{E}}$.

(iii) Consider again $\tilde{\mathbf{E}} \in \{\tilde{\mathbf{E}}_1, \tilde{\mathbf{E}}_2\}$, which has two opposite edges $\tilde{\mathbf{E}}_1, \tilde{\mathbf{E}}_2$ with the same refinement level and direction index. This and the construction of the metric dist yield that

$$\text{neighb}(\check{\mathcal{M}}, \tilde{\mathbf{E}}) \subseteq \text{neighb}(\check{\mathcal{M}}, \tilde{\mathbf{E}}_1) \cup \text{neighb}(\check{\mathcal{M}}, \tilde{\mathbf{E}}_2).$$

The claim has been shown for $\tilde{\mathbf{E}}_1, \tilde{\mathbf{E}}_2$ in (ii), and it also holds for $\tilde{\mathbf{E}}$ due to the equality of levels and direction indices.

(iv) We consider $\mathbf{E}' \in \check{\mathcal{E}} \cap \mathcal{E}$ with $\mathbf{E} \notin \text{neighb}(\mathcal{M}, \mathbf{E}')$. In order to prove the claim, we need to show that any new edges $\mathbf{E}'' \in \text{neighb}(\check{\mathcal{M}}, \mathbf{E}') \setminus \text{neighb}(\mathcal{M}, \mathbf{E}')$ satisfy (18). We therefore distinguish three cases.

Case 1: $\tilde{\mathbf{E}} \in \text{neighb}(\check{\mathcal{M}}, \mathbf{E}') \setminus \text{neighb}(\mathcal{M}, \mathbf{E}')$ from the subdivision of a neighbor element of \mathbf{E} . The edge $\tilde{\mathbf{E}}$ has two opposite edges $\tilde{\mathbf{E}}_1, \tilde{\mathbf{E}}_2$ with the same refinement level and direction index. Due to the

construction of the metric dist, at least one edge $\tilde{E} \in \{\tilde{E}_1, \tilde{E}_2\}$ is in $\text{neighb}(\check{\mathcal{M}}, \mathbf{E}') \cap \text{neighb}(\mathcal{M}, \mathbf{E})$ and satisfies (22), and \tilde{E} satisfies the same due to the equality of levels and direction indices.

Case 2: $E_1 \in \text{neighb}(\check{\mathcal{M}}, \mathbf{E}') \setminus \text{neighb}(\mathcal{M}, \mathbf{E}')$, with E_1 being a child of \mathbf{E} and $\ell(\mathbf{E}') \geq \ell(\mathbf{E}) + 1$. In this case we get

$$\begin{aligned} \text{dist}(\mathbf{E}', \mathbf{E}) &\leq \text{dist}(\mathbf{E}', E_1) + \text{dist}(E_1, \mathbf{E}) \leq \frac{p+1}{2} \cdot 2^{-\ell(\mathbf{E}')} + 2^{-2-\ell(\mathbf{E})} \\ &\leq \left(\frac{p+1}{2} + \frac{1}{2}\right) \cdot 2^{-1-\ell(\mathbf{E})} \leq \frac{p+1}{2} \cdot 2^{-\ell(\mathbf{E})} \\ \Rightarrow \mathbf{E}' \in \text{neighb}(\mathcal{M}, \mathbf{E}) &\stackrel{(20)}{\Rightarrow} \ell(\mathbf{E}') = \ell(\mathbf{E}) + 1 = \ell(E_1). \end{aligned}$$

Case 3: $E_1 \in \text{neighb}(\check{\mathcal{M}}, \mathbf{E}') \setminus \text{neighb}(\mathcal{M}, \mathbf{E}')$, with E_1 being a child of \mathbf{E} and $\ell(\mathbf{E}') \leq \ell(\mathbf{E})$. The edge midpoints of both \mathbf{E}' and \mathbf{E} are nodes in the uniform mesh $\mathcal{M}_{u[L]}^d$ of level $L = \max(\ell(\mathbf{E}'), \ell(\mathbf{E})) + 1 = \ell(\mathbf{E}) + 1 = \ell(E_1)$. By definition of the mesh metric dist, we have $\text{dist}(\mathbf{E}', \mathbf{E}) = \text{dist}(\text{mid}(\mathbf{E}'), \text{mid}(\mathbf{E})) = n \cdot 2^{-\ell(E_1)}$ with $n \in \mathbb{N}$ being the minimal number of elements connecting $\text{mid}(\mathbf{E}')$ and $\text{mid}(\mathbf{E})$ in the mesh $\mathcal{M}_{u[L]}^d$. A triangle inequality and $E_1 \in \text{neighb}(\check{\mathcal{M}}, \mathbf{E}')$ yield

$$\text{dist}(\mathbf{E}', \mathbf{E}) \leq \text{dist}(\mathbf{E}', E_1) + \text{dist}(E_1, \mathbf{E}) \leq \frac{p+1}{2} \cdot 2^{-\ell(\mathbf{E}')} + 2^{-1-\ell(E_1)} = \left(\frac{p+1}{2} \cdot 2^{\ell(E_1)-\ell(\mathbf{E}')} + \frac{1}{2}\right) \cdot 2^{-\ell(E_1)}.$$

Hence, we expect n to be bounded by $n \leq \frac{p+1}{2} \cdot 2^{\ell(E_1)-\ell(\mathbf{E}')} + \frac{1}{2}$. Since both n and $\frac{p+1}{2} \cdot 2^{\ell(E_1)-\ell(\mathbf{E}')}$ are integer numbers, we conclude that $n \leq \frac{p+1}{2} \cdot 2^{\ell(E_1)-\ell(\mathbf{E}')}$, hence $\text{dist}(\mathbf{E}', \mathbf{E}) \leq \frac{p+1}{2} \cdot 2^{-\ell(\mathbf{E}')}$ and finally $\mathbf{E} \in \text{neighb}(\mathcal{M}, \mathbf{E}')$ in contradiction to the assumption above. \square

Definition 5.5 (extension and skeleton of T- and I-nodes). Given a T-node or boundary I-node $N \in \mathcal{N}$ (see Definition 2.13), there is a unique element $Q \in \mathcal{Q}$ with two common edges of equal direction index, $\mathcal{E}(N) \cap \mathcal{E}(Q) = \{E_1, E_2\}$, $E_1 \neq E_2$, $\text{di}(E_1) = \text{di}(E_2)$, that have a common opposite edge $E_o \in \mathcal{E}(Q)$. The *first-order extension* $\text{ext}^1(N)$ is a singleton set containing the edge that connects N with the midpoint of E_o . For an interior I-node N , there are two such neighboring elements Q_1, Q_2 , and $\text{ext}^1(N)$ is a set of two edges related to these two elements. In both cases, we have $\text{ext}^1(N) \subset \text{BEZIER}(\mathcal{E}) \setminus \mathcal{E}$.

The *T-node extension* (also for I-nodes) is the $\frac{p-1}{2}$ -th prolongation of $\text{ext}^1(N)$,

$$\text{ext}(N) = \text{ep}^{(p-1)/2}(\text{ext}^1(N)) \subset \text{BEZIER}(\mathcal{E}).$$

For T-nodes and boundary I-nodes, we define the *first-order skeleton* of N as $\text{Sk}(\text{ext}^1(N)) = \{E_1, E_2, E_o\}$, and for interior I-nodes, $\text{Sk}(\text{ext}^1(N)) = \{E_1, E_2, E_{o,1}, E_{o,2}\}$, with $E_{o,1} \in \mathcal{E}(Q_1)$ and $E_{o,2} \in \mathcal{E}(Q_2)$ the common opposite edges of E_1 and E_2 . The *n-th-order skeleton* is the union of the $(n-1)$ -th-order skeleton and their opposite edges, for $n = 1, \dots, \frac{p+1}{2}$. The *extension skeleton* $\text{Sk}(\text{ext}(N))$ of N is defined as its $\frac{p+1}{2}$ -th-order skeleton.

Note that all edges in $\text{Sk}(\text{ext}(N))$ have the same direction index due to the criterion (8). Hence, we can associate to each T-node or I-node the unique direction index of its extension skeleton, and we call N with $\text{di}(\text{Sk}(\text{ext}(N))) = \{j\}$ a *j-orthogonal* T-node (or I-node).

Theorem 5.6. *For any T- or I-nodes N, N' in a mesh \mathcal{M} generated by REFINE, where N is i-orthogonal, N' is j-orthogonal, and $i \neq j$, it holds $\text{ext}(N) \cap \text{ext}(N') = \emptyset$, i.e., the T-node extensions do not intersect.*

Proof. Analogously to the definition above, there is $Q \in \mathcal{Q}(N')$ with common edges $E_1, E_2 \in \mathcal{E}(N') \cap \mathcal{E}(Q)$ with $\text{di}(E_1) = \text{di}(E_2) = j$, and an opposite edge $E_o \in \mathcal{E}(Q)$ with $\text{di}(E_o) = j$. Since the mesh \mathcal{M} is generated by REFINE, the subdivision of E_1 's parent edge E_p was admissible in the sense of (19) and we conclude, since the mesh may have been refined further, that

$$\forall \mathbf{E}' \in \text{neighb}(\mathcal{M}, E_p) : \begin{cases} \ell(\mathbf{E}') > \ell(E_p) & \text{if } \text{di}(\mathbf{E}') < j \\ \ell(\mathbf{E}') \geq \ell(E_p) & \text{if } \text{di}(\mathbf{E}') \geq j. \end{cases}$$

For the opposite edge $E_o \in \mathcal{E}$, which is not refined, Theorem 5.4 yields

$$\forall E' \in \text{neighb}(\mathcal{M}, E_o) : \begin{cases} \ell(E_o) \leq \ell(E') \leq \ell(E_o) + 1 & \text{if } \text{di}(E') < j \\ \ell(E_o) - 1 \leq \ell(E') \leq \ell(E_o) + 1 & \text{if } \text{di}(E') = j \\ \ell(E_o) - 1 \leq \ell(E') \leq \ell(E_o) & \text{if } \text{di}(E') > j. \end{cases}$$

Together with $\ell(E_p) = \ell(E_o)$, we have

$$\forall E' \in \text{neighb}(\mathcal{M}, E_o) \cap \text{neighb}(\mathcal{M}, E_p) : \begin{cases} \ell(E') = \ell(E_o) + 1 & \text{if } \text{di}(E') < j \\ \ell(E_o) \leq \ell(E') \leq \ell(E_o) + 1 & \text{if } \text{di}(E') = j \\ \ell(E') = \ell(E_o) & \text{if } \text{di}(E') > j. \end{cases}$$

The size of the joint neighborhood $\text{neighb}(\mathcal{M}, E_o) \cap \text{neighb}(\mathcal{M}, E_p)$ is hence p elements in j -orthogonal direction, centered at Q , times $p + 2$ elements in the other direction(s), centered at Q . Elements in this neighborhood may have been subdivided in j -th direction, but in no other direction. Hence any i -orthogonal T-node N , $i \neq j$, is far away in the sense that its extension cannot intersect with the extension of N' . \square

Remark 5.7. The theorem above implies that for each iteration in Algorithm 1 for the Bézier mesh, the computed mesh contains only elements $Q \in \mathcal{Q}$ with either $\#\mathcal{N}(Q) = \#\mathcal{E}(Q) = 4$ or $\#\mathcal{N}(Q) = \#\mathcal{E}(Q) = 5$, where the latter are exactly those elements that neighbor a T- or I-node.

Theorem 5.6 states that the T-splines defined over the T-mesh \mathcal{M} are analysis-suitable. This implies that the functions are linearly independent. Moreover, on all elements $Q \in \mathcal{Q} \setminus \bigcup_{N \in \mathcal{N}_{EO}} D_p(N)$ the functions are locally linearly independent.

Theorem 5.8 (linear independence). *If the mesh \mathcal{M} is generated by REFINE, then the functions in \mathcal{B} as given in Definition 3.15 are linearly independent.*

Proof. The proof consists of three steps:

- a) For each extraordinary node $N \in \mathcal{N}_{EO}$, the functions in \mathcal{B}_N , together with the anchors $\mathcal{N}_A[N] = \{N' \in \mathcal{N}_A : \text{dist}(N, N') = (p + 1)/2\}$, span all functions in the 1-disk $D_1(N)$, i.e.,

$$\text{span}(\mathcal{B}_N \cup \mathcal{N}_A[N])|_{D_1(N)} = \mathcal{S}^p(\text{BEZIER}(\mathcal{M}), \kappa_0)|_{D_1(N)}.$$

Moreover, they are linearly independent by construction.

- b) For all other anchors $N' \in \mathcal{N}_A$, that are not contained in $\mathcal{N}_A[N]$ for any extraordinary node N , we have

$$B_{N'}|_{D_1(N)} \equiv 0$$

for all $N \in \mathcal{N}_{EO}$. Thus, they are linearly independent to the functions in (a).

- c) For all functions B_N with $N \in \mathcal{N}_A$, one can define functionals λ_N on the support of B_N , which form a dual basis, see, e.g., [22]. Since for any pair of anchors $N, N' \in \mathcal{N}_A$ the supports of B_N and $\lambda_{N'}$ are contained inside a structured submesh, due to Assumption 3.16, the analysis-suitability condition in Theorem 5.6 yields dual compatibility, i.e., $\lambda_{N'}(B_N) = \delta_{N, N'}$.

Consequently, all functions are linearly independent. \square

A similar statement is given for specific C^1/C^2 -smooth cubic T-splines in [20, Proposition 3.2], where the proof follows a similar reasoning. See also [17, 30] for related approaches.

6. Conclusions & Outlook

We have introduced an adaptive refinement algorithm for T-splines on unstructured meshes in two dimensions. Motivated by ideas for higher-dimensional structured T-meshes, the refinement routine is based on the concept of so-called direction indices that are involved in a recursive refinement procedure. We have proved that the refinement algorithm has linear complexity and preserves analysis-suitability away from extraordinary nodes. Additionally, we sketched a generalization to more general unstructured meshes and drew the connections to manifold splines and Isogeometric Analysis.

For the preservation of global analysis-suitability, we require uniform refinement in the p -disk of extraordinary nodes at the current stage, while the treatment of T-nodes in these regions is left to future work. Moreover, a combination of our adaptive refinement scheme with special constructions near extraordinary nodes, based on special splits and/or the imposition of smoothness of higher order as, e.g., in [17, 18, 19, 20, 21], is of great relevance for practical applications. Further, a natural extension of our work is the application to physical problems in the setting of Isogeometric Analysis, along with numerical experiments investigating the condition numbers and sparsity patterns of the system matrices.

Acknowledgments

Roland Maier acknowledges support by the German Research Foundation (DFG) in the Priority Program 1748 *Reliable simulation techniques in solid mechanics* (PE2143/2-2) and by the Göran Gustafsson Foundation for Research in Natural Sciences and Medicine. The research of Thomas Takacs is partially supported by the Austrian Science Fund (FWF) and the government of Upper Austria through the project P 30926-NBL entitled “Weak and approximate C^1 -smoothness in isogeometric analysis”.

References

- [1] T. Sederberg, J. Zheng, A. Bakenov, A. Nasri, *T-Splines and T-NURCCs*, ACM Transactions on Graphics **22**(3):477–484, 2003.
- [2] T. Sederberg, D. Cardon, G. Finnigan, N. North, J. Zheng, T. Lyche, *T-spline simplification and local refinement*, ACM Transactions on Graphics **23**(3):276–283, 2004.
- [3] Y. Bazilevs, V. Calo, J. Cottrell, J. Evans, T. Hughes, S. Lipton, M. Scott, T. Sederberg, *Isogeometric Analysis using T-splines*, Computer Methods in Applied Mechanics and Engineering **199**(5–8):229–263, 2010.
- [4] M. Dörfel, B. Jüttler, B. Simeon, *Adaptive Isogeometric Analysis by local h-refinement with T-splines*, Computer Methods in Applied Mechanics and Engineering **199**(5–8):264–275, 2010.
- [5] A. Buffa, D. Cho, G. Sangalli, *Linear independence of the T-spline blending functions associated with some particular T-meshes*, Computer Methods in Applied Mechanics and Engineering **199**(23–24):1437–1445, 2010.
- [6] X. Li, M. Scott, *On the nesting behavior of T-splines*, ICES Report 11-13, University of Texas at Austin, 2011.
- [7] X. Li, J. Zheng, T. Sederberg, T. J. R. Hughes, M. Scott, *On linear independence of T-spline blending functions*, Computer Aided Geometric Design **29**(1):63–76, 2012.

- [8] L. Beirão da Veiga, A. Buffa, D. Cho, G. Sangalli, *Analysis-suitable T -splines are dual-compatible*, Computer Methods in Applied Mechanics and Engineering **249/252**:42–51, 2012.
- [9] M. Scott, X. Li, T. Sederberg, T. Hughes, *Local refinement of analysis-suitable T -splines*, Computer Methods in Applied Mechanics and Engineering **213–216**:206–222, 2012.
- [10] L. Beirão da Veiga, A. Buffa, G. Sangalli, R. Vázquez, *Analysis-suitable T -splines of arbitrary degree: definition, linear independence and approximation properties*, Mathematical Models and Methods in Applied Sciences **23**(11):1979–2003, 2013.
- [11] Y. Zhang, W. Wang, T. J. R. Hughes, *Solid T -spline construction from boundary representations for genus-zero geometry*, Computer Methods in Applied Mechanics and Engineering **249/252**:185–197, 2012.
- [12] W. Wang, Y. Zhang, L. Liu, T. J. R. Hughes, *Trivariate solid T -spline construction from boundary triangulations with arbitrary genus topology*, Computer-Aided Design **45**(2):351–360, 2013.
- [13] P. Morgenstern, *Globally structured three-dimensional analysis-suitable T -splines: Definition, linear independence and m -graded local refinement*, SIAM Journal on Numerical Analysis **54**(4):2163–2186, 2016.
- [14] P. Morgenstern, *Mesh refinement strategies for the adaptive isogeometric method*, Ph.D. thesis, Institut für Numerische Simulation, Rheinische Friedrich-Wilhelms-Universität Bonn, 2017.
- [15] X. Li, M. Scott, *Analysis-suitable T -splines: Characterization, refineability, and approximation*, Mathematical Models and Methods in Applied Sciences **24**(06):1141–1164, 2014.
- [16] A. Bressan, A. Buffa, G. Sangalli, *Characterization of analysis-suitable T -splines*, Computer Aided Geometric Design **39**:17–49, 2015.
- [17] M. A. Scott, R. N. Simpson, J. A. Evans, S. Lipton, S. P. A. Bordas, T. J. R. Hughes, T. W. Sederberg, *Isogeometric boundary element analysis using unstructured T -splines*, Computer Methods in Applied Mechanics and Engineering **254**:197–221, 2013.
- [18] H. Casquero, L. Liu, C. Bona-Casas, Y. Zhang, H. Gomez, *A hybrid variational-collocation immersed method for fluid-structure interaction using unstructured T -splines*, International Journal for Numerical Methods in Engineering **105**(11):855–880, 2016.
- [19] H. Casquero, X. Wei, D. Toshniwal, A. Li, T. J. R. Hughes, J. Kiendl, Y. J. Zhang, *Seamless integration of design and Kirchhoff–Love shell analysis using analysis-suitable unstructured T -splines*, Computer Methods in Applied Mechanics and Engineering **360**:112765, 2020.
- [20] D. Toshniwal, H. Speleers, T. J. R. Hughes, *Smooth cubic spline spaces on unstructured quadrilateral meshes with particular emphasis on extraordinary points: geometric design and isogeometric analysis considerations*, Computer Methods in Applied Mechanics and Engineering **327**:411–458, 2017.
- [21] X. Wei, X. Li, K. Qian, T. J. R. Hughes, Y. Zhang, H. Casquero, *Analysis-suitable unstructured T -splines: Multiple extraordinary points per face*, Computer Methods in Applied Mechanics and Engineering **391**:114494, 2022.
- [22] G. Sangalli, T. Takacs, R. Vázquez, *Unstructured spline spaces for isogeometric analysis based on spline manifolds*, Computer Aided Geometric Design **47**:61–82, 2016.

- [23] X. Li, J. Zhang, *AS++ T-splines: Linear independence and approximation*, Computer Methods in Applied Mechanics and Engineering **333**:462–474, 2018.
- [24] J. Zhang, X. Li, *Local refinement for analysis-suitable++ T-splines*, Computer Methods in Applied Mechanics and Engineering **342**:32–45, 2018.
- [25] X. Li, X. Li, *AS++ T-splines: arbitrary degree, nestedness and approximation*, Numerische Mathematik **148**(4):795–816, 2021.
- [26] R. Görmer, P. Morgenstern, *Analysis-suitable T-splines of arbitrary degree and dimension*. Submitted, 2021.
- [27] A. Buffa, G. Sangalli, *Isogeometric analysis: a new paradigm in the numerical approximation of PDEs: Cetraro, Italy 2012*, Vol. 2161, Springer, Cham; Centro Internazionale Matematico Estivo (C.I.M.E.), Florence, 2016.
- [28] M. Farzaneh, *Graph coloring by genetic algorithm*, MATLAB Central File Exchange. Retrieved February 3, 2022.
URL <https://de.mathworks.com/matlabcentral/fileexchange/74118-graph-coloring-by-genetic-algorithm/>
- [29] R. H. Nochetto, A. Veese, *Primer of adaptive finite element methods*, in: Multiscale and adaptivity: modeling, numerics and applications, Vol. 2040 of Lecture Notes in Mathematics, Springer, Heidelberg, 2012, pp. 125–225.
- [30] X. Li, *Some properties for analysis-suitable T-splines*, Journal of Computational Mathematics **33**(4):428–442, 2015.
- [31] C. M. Grimm, J. F. Hughes, *Modeling surfaces of arbitrary topology using manifolds*, in: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques, 1995, pp. 359–368.
- [32] M. Siqueira, D. Xu, J. Gallier, L. G. Nonato, D. M. Morera, L. Velho, *A new construction of smooth surfaces from triangle meshes using parametric pseudo-manifolds*, Computers & Graphics **33**(3):331–340, 2009.
- [33] X. Yuan, K. Tang, *Rectified unstructured T-splines with dynamic weighted refinement for improvement in geometric consistency and approximation convergence*, Computer Methods in Applied Mechanics and Engineering **316**:373–399, 2017.

A. Manifold interpretation of unstructured meshes

In this section, we generalize the setting introduced in Section 2 from planar partitions to manifolds. The notion of an unstructured T-mesh extends quite directly to surfaces and manifold-like domains. In Section 2.1, we defined meshes via partitions of planar, polygonal domains. However, the functions on such unstructured T-meshes are defined locally, either on structured submeshes or on specific submeshes around extraordinary nodes. The entire polygonal domain is thus covered by overlapping submeshes and more precisely, every point of the partition, except for extraordinary nodes, is in the interior of some structured submesh. All structured submeshes \mathcal{M}' possess a corresponding parameter domain $\mathcal{D}(\widehat{\mathcal{M}}')$, as introduced in Definition 2.10. In that definition, $\widehat{\mathcal{M}}'$ is a mesh composed entirely of axis-aligned rectangles. Any function $\varphi \in \mathcal{B}$ defined on the mesh

\mathcal{M} has a spline representation on the parameter domain of each submesh, i.e., for the submesh \mathcal{M}' we have the spline

$$\widehat{\varphi}' : \mathcal{D}(\widehat{\mathcal{M}}') \rightarrow \mathbb{R}.$$

The spline representations on different submeshes \mathcal{M}' and \mathcal{M}'' are related through the mappings $\mathbf{F}_{\mathcal{Q}'}$ and $\mathbf{F}_{\mathcal{Q}''}$ introduced in Definition 2.10, via $\widehat{\varphi}' \circ \mathbf{F}_{\mathcal{Q}'}^{-1} \circ \mathbf{F}_{\mathcal{Q}''} = \widehat{\varphi}''$, which is defined only on the parameter domain of the intersection $\mathcal{D}(\mathcal{M}') \cap \mathcal{D}(\mathcal{M}'')$.

This underlying structure suggests that the submeshes and parameter domains, together with the mappings $\mathbf{F}_{\mathcal{Q}'}^{-1} \circ \mathbf{F}_{\mathcal{Q}''}$, define a manifold. Splines can then be defined on this manifold, following the approach in [22]. The following subsections are devoted to defining T-meshes and T-splines based on such a manifold interpretation.

A.1. Meshes on parameter manifolds

We follow [22], which is based on [31], to introduce an abstract representation of the parameter domain, the so-called *parameter manifold*. For simplicity, we provide definitions only for two-dimensional manifolds. Extensions to general dimensions $d \geq 2$ can be found in [22].

Definition A.1 (proto-manifold). A proto-manifold of dimension two consists of

- a finite set $\{\omega_i\}_{i=1,\dots,N}$ (named *proto-atlas*) of *charts* ω_i , that are open polytopes $\omega_i \subset \mathbb{R}^2$;
- a set of open *transition domains* $\{\omega_{i,j}\}_{i,j=1,\dots,N}$ that are polytopes such that $\omega_{i,j} \subset \omega_i$, $\omega_{i,i} = \omega_i$, and each $\omega_{i,j}$ is the interior of its closure;
- a set of *transition functions* $\{\psi_{i,j}\}_{i,j=1,\dots,N}$, that are homeomorphisms $\psi_{i,j} : \omega_{i,j} \rightarrow \omega_{j,i}$ fulfilling the *cocycle condition* $\psi_{j,k} \circ \psi_{i,j} = \psi_{i,k}$ in $\omega_{i,j} \cap \omega_{i,k}$ for all $i, j, k = 1, \dots, N$, where $\psi_{ii} = \text{id}$;
- For every i, j , with $i \neq j$, for every $\zeta_i \in \partial\omega_{i,j} \cap \omega_i$ and $\zeta_j \in \partial\omega_{j,i} \cap \omega_j$, there are open balls, V_{ζ_i} and V_{ζ_j} , centered at ζ_i and ζ_j , such that no point of $V_{\zeta_j} \cap \omega_{j,i}$ is the image of any point of $V_{\zeta_i} \cap \omega_{i,j}$ by $\psi_{i,j}$.

Note that the last condition is taken from [32]. As pointed out in [22], it prevents bifurcations of the domain and guarantees that the resulting object is indeed a manifold.

It is easy to see that on a planar partition of Ω , which is segmented into overlapping, open subdomains such that $\Omega = \bigcup_i \Omega_i$, each Ω_i is an open polytope. Thus, the choice $\omega_i = \Omega_i$ as charts, $\omega_{i,j} = \Omega_{i,j} = \Omega_i \cap \Omega_j$ as transition domains with transition functions $\psi_{i,j}$ being the identity mapping on $\omega_{i,j}$ forms a proto-manifold. However, such a structure complicates the definition of smooth spline functions over the mesh. Alternatively, we can define for every structured submesh \mathcal{M}_i over the domain $\Omega_i = \mathcal{D}(\mathcal{M}_i)$ the corresponding chart as $\omega_i = \mathcal{D}(\widehat{\mathcal{M}}_i)$. In addition to such structured charts, one can introduce for each extraordinary node $N \in \mathcal{N}_{\text{EO}}$ a specific unstructured chart, which corresponds to the domain $\mathcal{D}(\mathbf{D}_p(N))$ of the p -disk around the node.

By merging and identifying the charts of the proto-manifold, we obtain a manifold, which can serve as the parameter domain, thus the name *parameter manifold*.

Definition A.2 (parameter manifold). Given a proto-manifold, the set

$$\Omega^P = \left(\bigsqcup_{i=1,\dots,N} \omega_i \right) / \sim \quad (23)$$

is called a *parameter manifold*. Here \bigsqcup denotes the disjoint union, i.e.,

$$\bigsqcup_{i=1,\dots,N} \omega_i = \{[\zeta_i, i], \zeta_i \in \omega_i, i = 1, \dots, N\}$$

and the equivalence relation \sim is defined for all $\zeta_i \in \omega_i$ and $\zeta_j \in \omega_j$, as

$$[\zeta_i, i] \sim [\zeta_j, j] \Leftrightarrow \psi_{i,j}(\zeta_i) = \zeta_j.$$

We denote by $\pi_i(\zeta_i) \in \Omega^P$ the equivalence class corresponding to $\zeta_i \in \omega_i$.

In the following, we use the notation $\Omega_i^P = \pi_i(\omega_i)$ and $\Omega_{i,j}^P = \pi_i(\omega_{i,j})$, where $\Omega_i^P, \Omega_{i,j}^P \subset \Omega^P$. Hence, we have $\Omega_{i,j}^P = \Omega_{j,i}^P = \Omega_i^P \cap \Omega_j^P$. Note that π_i is a one-to-one correspondence between each ω_i and Ω_i^P and plays the role of a local representation. Next, we define a proto-mesh on the charts.

Definition A.3 (proto-mesh). A *proto-mesh* on a proto-manifold is a collection of conforming meshes, i.e., it is a set

$$\{\tau_i\}_{i=1,\dots,N} \quad \text{with } \tau_i = \{\mathbf{q} \subset \omega_i\},$$

where

(P1) each set τ_i is composed of axis-parallel, open rectangles \mathbf{q} , called *elements*, and each set τ_i is a mesh on ω_i , i.e. the elements are disjoint and the union of the closures of the elements is the closure of ω_i ,

(P2) for every i, j , $\omega_{i,j}$ is the interior of the union of the closure of elements of τ_i , and

(P3) the transition functions $\psi_{i,j}$ map elements onto elements, i.e.,

$$\forall \mathbf{q} \in \tau_i, \psi_{i,j}(\mathbf{q}) \in \tau_j.$$

Similarly, the transition functions map vertices of \mathbf{q} onto vertices of $\psi_{i,j}(\mathbf{q})$ and edges of \mathbf{q} onto edges of $\psi_{i,j}(\mathbf{q})$.

Note that in addition to mesh elements one can also define the edges and nodes of the mesh derived from the elements in τ_i . To simplify the notation, we omit those definitions.

The proto-mesh naturally defines a mesh on Ω^P through the mappings π_i .

Definition A.4 (mesh on Ω^P). We define the mesh on the parameter manifold Ω^P as $\mathcal{M} = (\mathcal{Q}, \mathcal{E}, \mathcal{N})$, with

$$\mathcal{Q} = \{\mathbf{Q} \subset \Omega^P \mid \mathbf{Q} = \pi_i(\mathbf{q}), \mathbf{q} \in \tau_i \text{ for some } i = 1, \dots, N\}.$$

The property (P3) allows analogous definitions of the edges

$$\mathcal{E} = \{\mathbf{E} \subset \Omega^P \mid \mathbf{E} = \pi_i(\mathbf{e}), \mathbf{Q} = \pi_i(\mathbf{q}), \mathbf{e} \in \mathcal{E}(\mathbf{q}), \mathbf{q} \in \tau_i \text{ for some } i = 1, \dots, N\}$$

and nodes

$$\mathcal{N} = \{\mathbf{N} \subset \Omega^P \mid \mathbf{N} = \pi_i(\mathbf{n}), \mathbf{Q} = \pi_i(\mathbf{q}), \mathbf{n} \in \mathcal{N}(\mathbf{q}), \mathbf{q} \in \tau_i \text{ for some } i = 1, \dots, N\}$$

of the mesh \mathcal{M} .

Each chart ω_i defines a submesh \mathcal{M}_i with elements $\mathcal{Q}_i = \{\pi_i(\mathbf{q}), \mathbf{q} \in \tau_i\}$. Given this alternative definition of a mesh and submesh, we can also extend the definition of a structured (sub)mesh to manifolds. A chart ω_i and local mesh τ_i are called *structured* if τ_i is a structured mesh according to Definition 2.10. The definition can be extended to larger submeshes composed of several charts and is directly applicable to T-meshes as in Section 2.3.

If we are given a partition of a polygonal domain Ω , then $\Omega_i^P = \Omega_i$ for all subdomains and we can associate Ω^P with Ω . Each element of the equivalence class Ω^P corresponds to a point in Ω . For general parameter manifolds Ω^P , there exists a domain $\Omega \subset \mathbb{R}^n$ and a bijective embedding $\mathbf{G} : \Omega^P \rightarrow \Omega$. This yields a mapping $\mathbf{F}_{\mathcal{Q}_i} = \mathbf{G} \circ \pi_i : \omega_i \rightarrow \Omega_i \subseteq \Omega$. We can thus define T-splines over the manifold domain Ω .

A.2. Splines over meshes on parameter manifolds

To define splines over T-meshes on a parameter manifold, in analogy to Section 2.3, the functions need to be defined consistently as piecewise polynomials and the present setting requires a clear definition of continuity over edges.

Each spline function $\varphi : \Omega \rightarrow \mathbb{R}$ is defined as a piecewise polynomial over structured charts. Thus, for each structured chart ω_i the functions

$$\hat{\varphi}_i : \omega_i \rightarrow \mathbb{R},$$

with $\hat{\varphi}_i = \varphi \circ \mathbf{F}_{\mathcal{Q}_i}$, are polynomials on each element in τ_i . Since the functions need to be consistent for all elements shared by several structured charts, we need for all structured charts ω_i and ω_j that

$$\hat{\varphi}_i \circ \mathbf{F}_{\mathcal{Q}_i}^{-1} \circ \mathbf{F}_{\mathcal{Q}_j} = \hat{\varphi}_i \circ \pi_i^{-1} \circ \mathbf{G}^{-1} \circ \mathbf{G} \circ \pi_j = \hat{\varphi}_i \circ \pi_i^{-1} \circ \pi_j = \hat{\varphi}_i \circ \psi_{j,i} = \hat{\varphi}_j.$$

For both functions $\hat{\varphi}_i$ and $\hat{\varphi}_j$ to be polynomials of a fixed bi-degree (p, p) on each element, we need the transition function $\psi_{j,i}$ to be an Euclidean motion. In general, following [22], transition functions are assumed to be piecewise bilinear mappings with respect to the mesh. To obtain a unique definition of edge lengths, we assume that all transition functions between structured charts are global Euclidean motions. Moreover, we assume that every element and every edge of the mesh is contained inside the domain $\mathcal{D}(\Omega_i^P)$ of some structured chart ω_i .

The concept of continuity of functions $\varphi : \Omega = \mathbf{G}(\Omega^P) \rightarrow \mathbb{R}$, defined over a parameter manifold Ω^P , can now be introduced consistently. A function φ is C^k -continuous across an edge $\mathbf{E} \in \mathcal{E}$, if $\hat{\varphi}_i$ is C^k -continuous across $\pi_i^{-1}(\mathbf{E})$ for all structured charts ω_i that contain the edge (cf. Definition 3.3). This definition is consistent since all transition functions between structured charts are assumed to be C^∞ . This generalization allows one to define a spline space as in Definition 3.4, which is solely characterized by underlying structured charts, and extends the construction in Section 3. For more rigorous definitions of structured/unstructured meshes and spline spaces on parameter manifolds, we refer to [22].

B. Refining totally unstructured meshes

In this section, we investigate a variant of the proposed refinement scheme with rare violations of the condition (8) that is based on the manifold definition introduced in the previous section. This will enable us to use the refinement scheme on totally unstructured meshes (see Fig. 8) and when obeying a set of rules for these violations, we preserve the properties of the refinement algorithm presented above.

In order to identify violations of (8), we generalize the definition of direction indices and replace Definition 4.3, which works on the T-mesh \mathcal{M} on the parameter manifold Ω by the following construction. Recall Definition A.3 for the proto-mesh. On each submesh τ_i , $i = 1, \dots, N$, the edges are assumed to be axis-parallel. By setting the direction index of an edge \mathbf{e}_i in τ_i to the actual direction of \mathbf{e}_i , we find a direction labeling di_i that locally satisfies (8). The condition (P3) says that the transition functions map edges onto edges and each edge \mathbf{E} in the T-mesh \mathcal{M} represents an equivalence class $\mathbf{E} = \pi_i(\mathbf{e}_i)$ of edges \mathbf{e}_i in the submeshes τ_i . We get a set-valued direction labeling $\text{di} : \mathcal{E} \rightarrow 2^{\mathbb{N}}$, with $\text{di}(\pi_i(\mathbf{e}_i)) = \{\text{di}_j(\mathbf{e}_j) \mid \mathbf{e}_j \sim \mathbf{e}_i\}$.

Definition B.1 (admissible direction labeling). We call a direction labeling *admissible* if for any edge \mathbf{E} , the following conditions are fulfilled:

- Any adjacent edge $\mathbf{E}' \in \mathcal{E}(\mathcal{N}(\mathbf{E}))$ of \mathbf{E} satisfies $\max \text{di}(\mathbf{E}') < \min \text{di}(\mathbf{E})$ or $\min \text{di}(\mathbf{E}') > \max \text{di}(\mathbf{E})$.

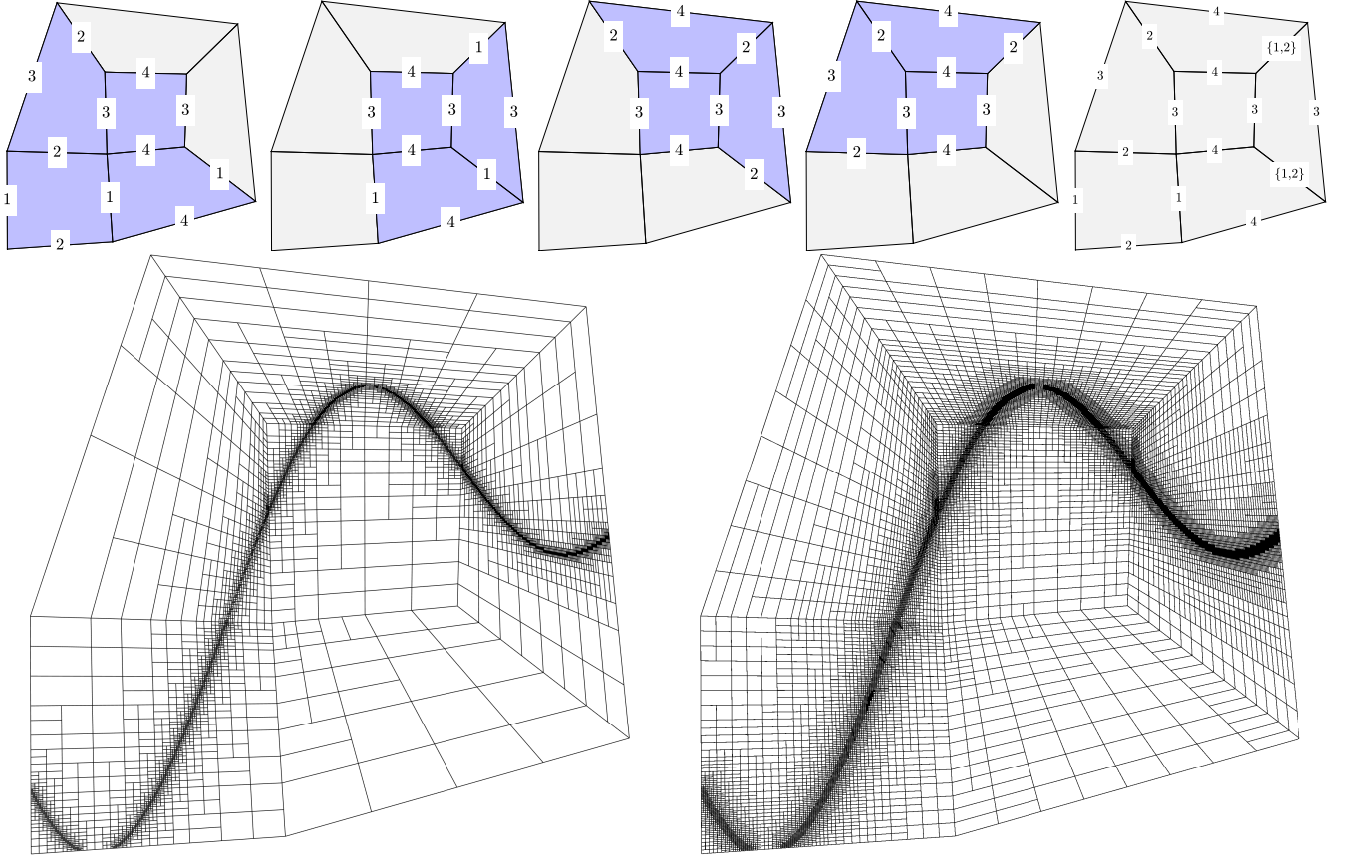


Figure 11: Although the mesh is totally unstructured, we can find a direction labeling that satisfies (8) locally and the associated generalized direction labeling satisfies Definition B.1 (top row). This can be applied for refinement, here for $p = 1$ (bottom left) and $p = 3$ (bottom right).

- In any neighbor element $Q \in \mathcal{Q}(E)$, the opposite edge $E' \in \mathcal{E}(Q) \setminus \mathcal{E}(\mathcal{N}(E))$ of E satisfies $\max \text{di}(E') \geq \min \text{di}(E)$ and $\min \text{di}(E') \leq \max \text{di}(E)$.

If these conditions are met, Algorithm 3 can be applied, with “ $\text{di}(E') < \text{di}(E)$ ” understood as $\max \text{di}(E') < \min \text{di}(E)$. The results from section 5 are still valid in this case, since the same interpretation is applicable in the proofs. Note, however, that the relation “ $\max \text{di}(E') \geq \min \text{di}(E)$ and $\min \text{di}(E') \leq \max \text{di}(E)$ ” instead of $\text{di}(E') = \text{di}(E)$ in the second bullet point above is not a transitive relation, but only applicable elementwise. We refer to Fig. 11 for an example with the totally unstructured mesh from Fig. 8.

For the deduction of direction indices, we further elaborate on the approach from Definition 4.3. Regarding the mesh as a plane graph $G_{\mathcal{M}}$, the above-mentioned approach can be explained via the dual graph G^* , which contains a node for each element in $G_{\mathcal{M}}$, including a node for the exterior element which neighbors all boundary edges. Two nodes in G^* are connected if the corresponding elements in $G_{\mathcal{M}}$ have a common edge. Hence there is a one-to-one correspondence between the edges of $G_{\mathcal{M}}$ and the edges of G^* . The condensation of edges into equivalence classes $[E]$ corresponds to a circuit decomposition of G^* , namely the unique decomposition into smallest circuits that traverse all nodes in a way, such that subsequent edges have no common element. Circuits are defined as follows.

- A walk is a sequence $(v_1, e_1, \dots, e_n, v_{n+1})$ such that v_i and v_{i+1} are boundary nodes of the edge e_i for all $i = 1, \dots, n$.
- A trail is a walk without edge repetitions.

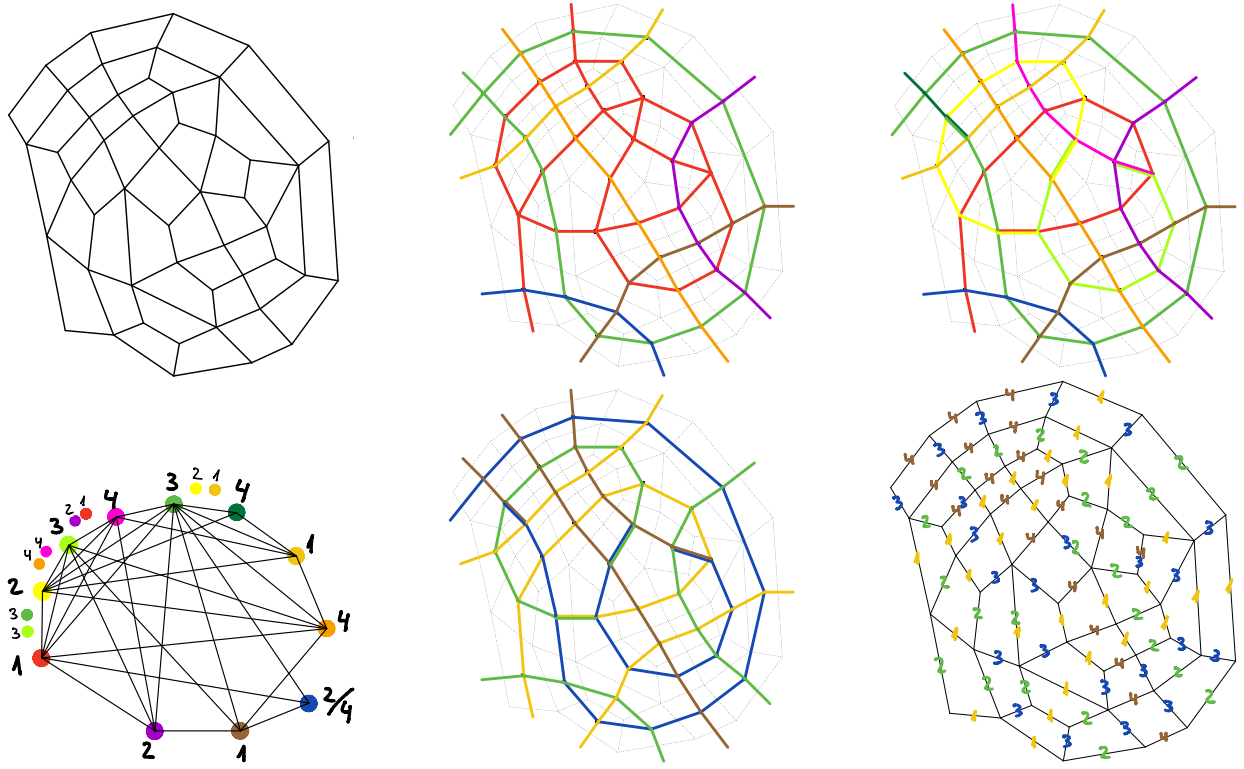


Figure 12: For the deduction of direction indices in totally unstructured meshes, we subdivide the circuits of the dual graph into overlapping paths. Then, we solve a constrained coloring problem on the touch graph of the paths. The touch graph contains a node for each path, and two nodes are connected if the corresponding paths have a common node except for the exterior node. The constrained coloring problem seeks a labeling $c : \text{paths} \rightarrow \mathbb{N}$ such that any two paths p, q with a common non-exterior node satisfy $c(p) \neq c(q)$, and that for any edge e that connects vertices v, w and that is in two distinct, overlapping paths p, q , any other path r containing one of the nodes v, w satisfies $c(r) < \min(c(p), c(q))$ or $c(r) > \max(c(p), c(q))$. This constraint corresponds to the second bullet point in Definition B.1.

- A path is a trail without node repetitions.
- A circuit is a closed trail, i.e. a trail $(v_1, e_1, \dots, e_n, v_{n+1})$ with $v_1 = v_{n+1}$.

In order to eliminate self-crossings of the circuits in the totally unstructured case, we subdivide the circuits into overlapping paths that are not self crossing, see Figure 12. Afterwards, the corresponding touch graph is colored with the additional constraint that for any edge with multi-valued direction with minimal value a and maximal value b must not have a direction index in the range $[a, b]$, all adjacent edges must not have direction indices in the range $[a, b]$.

C. Geometry mapping and isogeometric functions

As T-splines are used in isogeometric analysis, one also has to take into account a geometry mapping which maps T-splines from the (manifold) domain Ω onto a physical domain of interest D .

We consider a T-mesh \mathcal{M} and a corresponding T-spline basis \mathcal{B} . We define a mapping

$$\mathbf{G} : \Omega \rightarrow D \subset \mathbb{R}^n,$$

where n is the spatial dimension of the physical domain of interest such that

$$\mathbf{G} \in (\text{span}(\mathcal{B}))^n.$$

We can now define mapped elements of the T-mesh and mapped (isogeometric) T-splines over D as

$$\mathcal{T} = \{\mathbf{G}(\mathcal{Q}) : \mathcal{Q} \in \mathcal{Q}\}$$

and

$$\mathcal{B}(\mathcal{T}) = \{B \circ \mathbf{G}^{-1} : B \in \mathcal{B}\},$$

respectively.

Let \mathcal{M}' be a refinement of \mathcal{M} and let $\mathcal{B}(\mathcal{M}')$ be the refined T-spline basis. For many configurations, e.g., near extraordinary nodes, it may not be feasible to assume nestedness of the spaces, i.e.,

$$\text{span}(\mathcal{B}(\mathcal{M})) \not\subseteq \text{span}(\mathcal{B}(\mathcal{M}')). \quad (24)$$

Consequently, the mapping $\mathbf{G}': \Omega \rightarrow D'$ may not be equal to \mathbf{G} but only an approximation. If the dimensions of Ω and D are equal ($d = n$), it is reasonable to assume that the physical domain remains unchanged, i.e., $D = D'$. This is usually not possible if the domain is a surface in space, for instance. In both cases, the non-nestedness of the underlying spline spaces has to be taken into account when refining, cf. [33].