Almost-C^1 splines

Biquadratic splines on unstructured quadrilateral meshes and their application to fourth order problems

Takacs, Thomas; Toshniwal, Deepesh

DOI
10.1016/j.cma.2022.115640

Publication date
2023

Document Version
Final published version

Published in
Computer Methods in Applied Mechanics and Engineering

Citation (APA)
Takacs, T., & Toshniwal, D. (2023). Almost-C^1 splines: Biquadratic splines on unstructured quadrilateral meshes and their application to fourth order problems. *Computer Methods in Applied Mechanics and Engineering*, *403*(Part A), Article 115640. https://doi.org/10.1016/j.cma.2022.115640

Available online at www.sciencedirect.com
ScienceDirect
Comput. Methods Appl. Mech. Engrg. 403 (2023) 115640
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

Almost-C^1 splines: Biquadratic splines on unstructured quadrilateral meshes and their application to fourth order problems

Thomas Takacs[a,b], Deepesh Toshniwal[c,*]

[a] *Johann Radon Institute for Computational and Applied Mathematics, Austrian Academy of Sciences, Austria*
[b] *Institute of Applied Geometry, Johannes Kepler University Linz, Austria*
[c] *Delft Institute of Applied Mathematics, Delft University of Technology, The Netherlands*

Received 27 January 2022; received in revised form 30 August 2022; accepted 4 September 2022
Available online xxxx

Abstract

Isogeometric Analysis generalizes classical finite element analysis and intends to integrate it with the field of Computer-Aided Design. A central problem in achieving this objective is the reconstruction of analysis-suitable models from Computer-Aided Design models, which is in general a non-trivial and time-consuming task. In this article, we present a novel spline construction, that enables model reconstruction as well as simulation of high-order PDEs on the reconstructed models. The proposed *almost-C^1 splines* are biquadratic splines on fully unstructured quadrilateral meshes (without restrictions on placements or number of extraordinary vertices). They are C^1 smooth at all regular and extraordinary vertices. Moreover, they are C^1 smooth across all edges between regular vertices and C^0 smooth across all edges that are adjacent to an extraordinary vertex. The splines thus form H^2-nonconforming analysis-suitable discretization spaces. This is the lowest-degree unstructured spline construction that can be used to solve fourth-order problems. The associated spline basis is non-singular and has several B-spline-like properties (e.g., partition of unity, non-negativity, local support), the almost-C^1 splines are described in an explicit Bézier-extraction-based framework that can be easily implemented. Numerical tests suggest that the basis is well-conditioned and exhibits optimal approximation behaviour.

Keywords: almost-C^1 splines; Isogeometric analysis; Unstructured quadrilateral meshes; Analysis-suitable splines; Optimal approximation

---

1. Introduction

In this article we present a new approach for building analysis-suitable biquadratic spline spaces on fully unstructured quadrilateral meshes, so-called almost-C^1 splines. In particular, with the recent mixed smoothness spline construction from [1] as the starting point, we build splines that are C^1 smooth at all mesh vertices and C^1 smooth across most edges, as long as the control points are in a generic position. Thus, the construction yields H^2-nonconforming spline spaces that can be used to solve fourth-order problems. We test this on several model problems, such as the biharmonic problem, Kirchhoff–Love thin shells, a surface Cahn–Hilliard model and the surface Laplace–Beltrami eigenvalue problem. We obtain almost-C^1 splines by employing approximate C^1

---

[*] Corresponding author.
E-mail addresses: thomas.takacs@ricam.oeaw.ac.at (T. Takacs), d.toshniwal@tudelft.nl (D. Toshniwal).

https://doi.org/10.1016/j.cma.2022.115640

(a) watertight quad mesh (b) reconstructed spline surface (c) underlying Bézier mesh

Fig. 1. The above figures show how almost-C^1 splines can enable analysis-suitable model reconstructions. For a crash simulation vehicle model of the Dodge Neon (not shown) [18], figure (a) shows a watertight bilinear quadrilateral mesh generated using Rhinoceros®. Figure (b) shows an analysis-suitable spline model reconstructed using almost-C^1 splines; figure (c) shows the underlying Bézier mesh. Note that watertight mesh generation for CAD models is itself an active area of research; for instance, see the recent article [19]. Note also that we used the quad-mesh in figure (a) to automatically compute control points used in figure (b); in practice this reconstruction step can be improved by using information from the original (CAD or finite element) model.

smoothness at a fixed number of mesh edges (depending only on mesh topology and independent of its refinement level) and classical, parametric C^1 smoothness at all other edges. The motivation for doing so is manifold.

1.1. Motivation

This spline construction is inspired by the general aim of Isogeometric Analysis (*IGA*), introduced in [2], which is the integration of a finite element-like analysis within Computer-Aided Design (*CAD*) [3], cf. [4]. Achieving this objective would lead to a uniform and significantly more efficient design-through-analysis workflow for many engineering applications. In contrast, in the current setup a significant portion of the time is spent neither on design nor on analysis but is dominated by generating analysis-suitable reconstructions of CAD models (studies suggest up to 80%, cf. [5]).

Thus, our focus lies on developing splines that enable model reconstruction as well as simulation of high-order PDEs on the reconstructed models (cf. Fig. 1). The goal is to achieve such analysis-suitable spline reconstructions for arbitrary topology geometries. While IGA was first introduced on single NURBS patches, that is, on structured quadrilateral meshes, to handle general bivariate geometries (planar domains and surfaces), one must be able to define analysis-suitable splines on general unstructured meshes.

To increase the geometric flexibility of the construction, the almost-C^1 splines are only approximately C^1 smooth near extraordinary vertices. Nonetheless, it is known that approximately C^1 smooth function spaces can be used to build optimally convergent finite element methods for partial differential equations (PDEs) in variational form requiring H^2 regularity; see, e.g., [6]. By relaxing the C^1 smoothness constraints in a specific manner, the singularities that appear for parametrically smooth spline constructions can be avoided (cf. [7]) and, potentially, better numerical convergence than non-singular geometrically smooth spline constructions (e.g. as in [8]) could be obtained.

Since we provide explicit descriptions of the almost-C^1 splines, they form a viable alternative to other approximately smooth constructions that are based on a weak imposition of smoothness, such as Nitsche's method [9,10], the mortar method [11–13] or a mixed approach [14]. A similar, explicit construction for approximately C^1 smooth isogeometric multi-patch spaces is presented in [15,16].

Finally, the proposed almost-C^1 splines allow a construction with a highly local footprint and fewer restrictions than the available parametrically or geometrically smooth alternatives, cf. [17].

With this motivation in mind, we begin by highlighting some prior work in these areas in Section 1.2, then we summarize the main properties of almost-C^1 splines in Section 1.3 and present a short outline of this article in Section 1.4.

1.2. Smooth splines over unstructured quadrilateral meshes

In the following we discuss other approaches that are related to the construction of almost-C^1 splines presented in this paper. Since almost-C^1 splines are smooth splines over quadrilateral meshes, we mostly focus the discussion on similar, quad-based spline constructions. Note that this is an extensive area of active research and we do not attempt to present an exhaustive overview; instead we point the readers to the comprehensive literature reviews recently presented in [17]. For instance, we focus here on constructions which build spline functions by smoothly joining polynomials on quadrilateral patches. This is in contrast to spline manifolds as in [20] where the functions are locally defined by composition of polynomials with suitably chosen blending functions, and which have been used to build bivariate [21–23] and trivariate splines [23] for IGA.

In the context of piecewise-polynomial splines over quadrilateral meshes, one can distinguish two types of smoothness — parametric and geometric. While parametrically smooth splines assume (at least locally) a joint parameter domain for neighbouring faces, in which the smoothness is prescribed, geometric smoothness is defined directly in physical space. As we will see in the following, many constructions rely on both parametric and geometric smoothness. While geometric smoothness between Bézier or B-spline patches is a well-known concept in CAD [3, Chapter 8], it has only recently been used for simulations.

One possibility to create parametrically smooth surfaces from unstructured meshes is to use subdivision, see [24–26]. While subdivision is a process that can be defined entirely on a mesh, its limit surface can be interpreted as a piecewise polynomial spline. This limit surface has a finite representation in regular regions of the mesh (e.g. equivalent to bicubic B-splines in case of Catmull–Clark subdivision) and is composed of an infinite number of spline rings around extraordinary vertices, cf. [27]. When using subdivision surfaces in IGA, this peculiar property must be taken into account, e.g. when performing numerical quadrature [28]. While the approximation properties are in general sub-optimal for both second- and fourth-order problems, see [8,29–34], recent work has shown that tuned subdivision schemes [35] can yield optimal approximation at least for second-order problems.

It is also possible to employ parametric smoothness over a quadrilateral mesh composed of finitely many faces. While such a construction yields B-splines (or locally refined splines, such as T-splines) in structured regions of the mesh, singularities are introduced at extraordinary vertices. This phenomenon was studied and used in [36,37] to create smooth but singular spline surfaces. In this setup, additional geometric smoothness has to be imposed at the extraordinary vertex. This results in smooth but degenerate Bézier patches, so-called D-patches. Such constructions were used for IGA in [7,38–40] by incorporating them within bicubic splines. Even though the spline geometries are singular, the spaces possess other favourable properties for both design and analysis. They can be used to discretize high-order PDEs, see e.g. [41,42], since they possess the required H^2-regularity properties, cf. [43]. The spaces constructed in [7,39] also demonstrate optimal convergence under mesh refinement when applied to fourth-order problems. Finally, while [7,39] assumed that the extraordinary vertices were separated by regular vertices, [40] showed that this assumption was not necessary for guaranteeing linear independence of the spline basis.

Alternatively, one can increase the flexibility of smooth splines over quadrilateral meshes by creating polar singularities, which are the result of edges mapped onto single points. General C^k smooth splines, for $k \geq 0$, were developed in [44] over singular Bézier patches and in [45] over general polar quad meshes. An explicit C^1 smooth construction was presented in detail in [46] and used for simulations on smooth, deforming surfaces in [47].

When constructing smooth splines over singular or polar configurations, additional geometric continuity has to be imposed at the singular or polar point to achieve the desired smoothness. Similar constructions can be employed, if parametric smoothness of higher order is enforced only for those edges that are away from extraordinary vertices, while at those edges near extraordinary vertices geometric smoothness is imposed, such as in [48,49]. The dimension of such locally defined, geometrically smooth spline spaces over topological, mixed quadrilateral–triangle meshes was studied in [50]. Later, approximation properties could be shown for G^1 smooth isogeometric elements over planar, quadrilateral [51] and mixed [52] meshes.

In the following we give an overview of constructions that rely on geometric smoothness not in such a local, but in a global sense. Geometrically C^1 smooth isogeometric discretizations can be defined over bilinear Bézier elements [53], over bilinear spline patches [54], or over more general planar multi-patch domains [55–57]. Recently, constructions for C^2 smoothness over multi-patch domains were developed in [58,59].

It has been shown in [55], that C^1 smooth spaces over multi-patch domains possess optimal approximation properties only in the case of so-called analysis-suitable G^1 parameterizations. While this condition can be fulfilled for planar domains following a reparameterization [60], for general planar multi-patch domains and surfaces it is not satisfied. Thus, to increase the geometric flexibility and allow for constructions over general domains, one may increase the polynomial degree locally [61,62] or reduce the smoothness requirements by replacing exact C^1 smoothness by approximate C^1 smoothness [15,16]. In this paper we follow a similar approach, but instead of enforcing approximate C^1 smoothness along entire interfaces between patches, we impose approximate C^1 smoothness only at mesh edges near extraordinary vertices. In the following we give an overview of the most important properties of the almost-C^1 splines that we propose in this paper.

1.3. Properties of almost-C^1 splines

Given an unstructured mesh $\mathscr{T}$ consisting of quadrilaterals (we allow both planar and non-planar meshes of arbitrary topology), almost-C^1 splines are biquadratic splines on $\mathscr{T}$ that extend the construction developed in [1]. The following briefly outlines some features of our construction; they are elaborated upon later in the article.

- **Definable on fully unstructured meshes**: We allow all types of manifold quadrilateral meshes with no restrictions on the numbers or placements of extraordinary vertices (e.g., multiple extraordinary vertices per quadrilateral are allowed, boundary extraordinary vertices are allowed).
- **Well-conditioned B-spline-like basis**: The spline basis functions have several useful B-spline-like properties: partition of unity, non-negativity, local support and linear independence, cf. Proposition 3.8. The spline degree-of-freedom structure is simple and allows simple control of the geometry and functions at the boundary. Furthermore, the spline basis is non-singular and well-conditioned.
- **Almost-C^1 smoothness**: If the control points are in a generic (non-degenerate) position, then the resulting splines are globally C^0 smooth and in addition C^1 smooth (in an isogeometric sense) at all vertices of the mesh, C^1 smooth across all edges between regular vertices, and approximately smooth across edges containing extraordinary vertices, cf. Proposition 3.9. This is in contrast to [1], where the splines are only C^0 at extraordinary vertices for almost all control point configurations.
- **Easy implementation**: We utilize and extend the non-nested refinement process from [1] which is convergent and nested, when restricted to the boundaries. As outlined in [1], this has two benefits which improve upon nested refinements of spline spaces of mixed smoothness (e.g., see [7,63]). The non-nestedness of the refinement allows us to "shrink" the neighbourhoods of approximate C^1 smoothness and leads to a very simplified computer implementation and, furthermore, in the limit of infinite refinements, converges to a smooth limit surface. At the same time, the refinement process leaves the spline invariant on the mesh boundary; this is especially useful if the boundaries of a spline geometry are composed of special curves such as conic sections.
  Numerical tests indicate that the spline spaces also demonstrate optimal approximation behaviour in the L^2, H^1 and H^2 norms for second- and fourth-order problems under mesh refinement. Conceptually, this approach can be seen as an amalgamation of the "design" and "analysis" philosophies from [7] – it offers the ease of working with the design space while also being suitable for analysis.
- **Lowest-order unstructured spline construction for fourth-order problems**: Our spline construction only uses biquadratic polynomial pieces and is thus the lowest-order unstructured spline construction suitable for fourth-order problems. Note that it is well-known that for certain problems (e.g., from structural mechanics) higher polynomial degrees might help alleviate *locked* or non-convergent approximations; similar ad-hoc solutions can also be derived for lower polynomial degrees (for instance, by utilizing reduced quadrature).

Note that, on planar geometries, the construction from [1] also yields optimal convergence rates, at least for the biharmonic problem, with the almost-C^1 splines offering smaller errors in the broken H^2 norm and in the corresponding DG error norm. However, on surface geometries, the construction from [1] is in general not smooth at extraordinary vertices and does not yield a smooth surface in the limit, unless special care is taken at the extraordinary vertices.

1.4. Outline of the paper

In Section 2 we introduce the relevant notation for the unstructured quadrilateral meshes that we focus on. Then, in Section 3 we present the construction of unstructured biquadratic splines over such meshes, culminating in the definition of almost-C^1 splines in Section 3.5. Their useful properties are presented in Propositions 3.8 and 3.9. Next, in Section 4, we discuss the (non-nested) refinement of almost-C^1 spline spaces and geometries. Proposition 4.3 characterizes our refinement rules. Finally, in Section 5 we present some numerical tests focusing on the analysis-suitability of the proposed B-splines, including the Scordelis–Lo thin shell benchmark, a Cahn–Hilliard problem on a closed surface and the analysis of a Laplace–Beltrami eigenvalue problem.

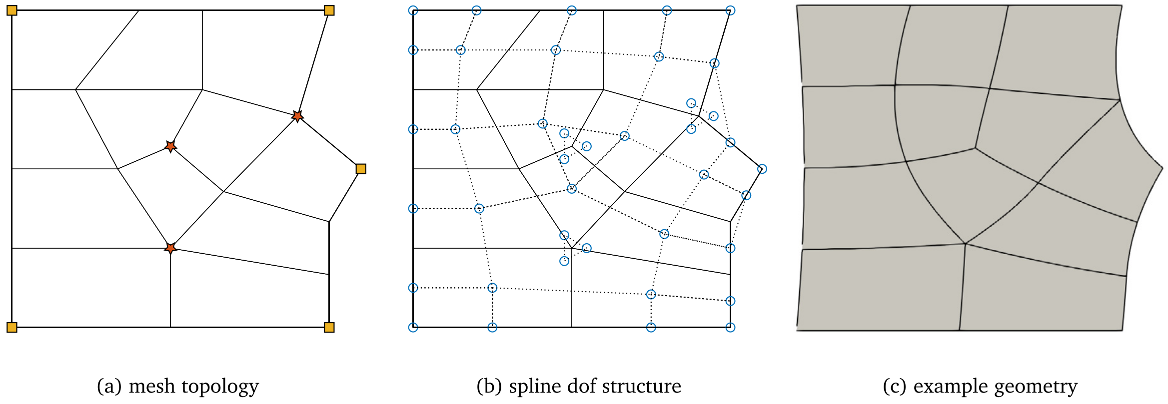

(a) mesh topology (b) spline dof structure (c) example geometry

Fig. 2. An example of almost-C^1 splines as defined in Section 3, with the underlying mesh in (a), the dof-structure in (b) and an example geometry in (c).

2. Unstructured quadrilateral meshes

We are interested in modelling complex, smooth, 2-dimensional geometries of arbitrary topology and in solving scalar and vector-valued PDEs on such geometries. For instance, planar geometries in $\mathbb{R}^2$ or surfaces in $\mathbb{R}^3$. Then, splines defined on unstructured meshes can help create such complex geometries, and can thereafter be used for numerically solving PDEs on them. We focus here on unstructured quadrilateral meshes. In this section, we define some relevant notation for such meshes, which are the basis for defining almost-C^1 splines; see Fig. 2(a) for reference.

Before we begin, we would like to point out that we consider $\mathscr{T}$ as a topological construct only — in general, the quadrilaterals in $\mathscr{T}$ will not be assumed to occupy a common parametric domain. Similarly, the meshes are not restricted to be planar or of trivial topology either. Representations as in Fig. 2(a) will only be for the purpose of specifying the connectivity of the different quadrilaterals with each other. As such, spline geometries and spline functions on those geometries will be built by appropriately selecting the degrees of freedom for splines on $\mathscr{T}$. An example corresponding to the mesh in Fig. 2(a) is shown in Fig. 2(c); see Section 3.5 for details on the construction of almost-C^1 splines. Note that the spline construction presented there is based on local geometric data around extraordinary vertices. However, the construction can easily be generalized to a purely topological one, as explained in Remark 3.4 and Appendix B.

We will denote all *quadrilateral meshes* with $\mathscr{T}$. We assume that $\mathscr{T}$ is without any hanging nodes and that the interiors of all quadrilaterals are disjoint. Each quadrilateral in $\mathscr{T}$ will be called a *face of* $\mathscr{T}$, or simply a *face*. More generally, for $k \leq 2$, the k-dimensional geometric components of the mesh will be collected in sets $\mathscr{T}_k$. That is, vertices in $\mathscr{T}_0$, edges in $\mathscr{T}_1$, and faces in $\mathscr{T}_2$.

We assume that $\mathscr{T}$ is such that each edge is shared by at most two faces of the mesh. If any edge is contained in only one face then it is called a *boundary edge*, otherwise it is called an *interior edge*; boundary edges have been displayed with slightly thicker lines in Fig. 2(a). Any vertices that lie on a boundary edge are called *boundary vertices*, otherwise they are called *interior vertices*. We denote the sets of boundary edges and vertices with $\mathscr{T}_1^B$ and $\mathscr{T}_0^B$, respectively. The set of interior vertices and edges will be denoted by $\mathring{\mathscr{T}}_k$ for $k = 0$ and 1, respectively. We also assume that there are no 'kissing vertices' in the mesh. That is, for any two faces σ, σ' that share a common vertex γ, there is a sequence of faces $\sigma_0, \ldots, \sigma_\ell$ that all contain γ such that $\sigma_0 = \sigma$, $\sigma_i = \sigma'$, and $\sigma_i \cap \sigma_{i-1} \in \mathring{\mathscr{T}}_1$ for $i = 1, \ldots, \ell$.

We always assume faces and edges to be closed sets. Thus, we define the *valence* of a vertex, edge or face of $\mathscr{T}$ to be the number of faces that contain it. For $\phi \in \mathscr{T}_k$, $k \leq 2$, we will denote the valence of ϕ with μ_ϕ. In particular,

- the valence μ_σ of any face σ is exactly 1 since each face contains itself;
- the valence μ_τ of a boundary or interior edge τ is 1 or 2, respectively, by the above definitions.

Vertices γ of $\mathscr{T}$ will be called *extraordinary vertices* if they are interior vertices with valences $\mu_\gamma \neq 4$, or if they are boundary vertices with valences $\mu_\gamma > 2$. Vertices of valence 1 will be called *corner vertices* and will be collected

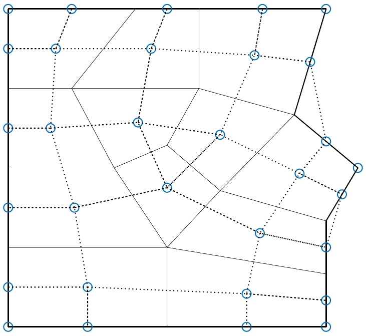


Fig. 3. The degree of freedom structure for the spline space $\mathscr{B}^*$ on the mesh $\mathscr{T}$ from Fig. 2(a). There is one dof associated to each face, boundary edge and corner vertex of the mesh $\mathscr{T}$.

in the set $\mathscr{T}_0^C \subseteq \mathscr{T}_0^B = \mathscr{T}_0 \backslash \mathring{\mathscr{T}}_0$. For visual consistency, we will always denote the extraordinary and corner vertices of a mesh by respectively placing red stars and yellow squares on them; see Fig. 2(a). Moreover, we call all edges that contain one or more extraordinary vertices as *spoke edges* and all faces that contain one or more extraordinary vertices as *extraordinary faces*. Otherwise, we call them regular edges and regular faces, respectively. We denote the set of all extraordinary vertices and faces with $\mathscr{T}_0^E$ and $\mathscr{T}_2^E$, respectively.

3. Unstructured biquadratic splines

In this section we present the construction of splines over unstructured quadrilateral meshes $\mathscr{T}$ as defined above. The construction is based on two steps. First, we summarize a biquadratic spline basis construction from [1] which depends purely on topological properties of the mesh (Sections 3.1–3.3). These functions span the space $\mathscr{B}^*$, which is biquadratic on every face, C^0 at all extraordinary points and across all spoke edges and C^1 across all other edges of the mesh. In the second step the basis functions of $\mathscr{B}^*$, which have support on extraordinary faces, are modified such that they have vanishing value and gradient at each extraordinary point. Consequently, three additional functions for each extraordinary vertex are introduced that locally span all linear functions. These functions can be defined using some geometric information of the mesh. The resulting functions constitute the almost-C^1 splines over the mesh $\mathscr{T}$, spanning the space $\mathscr{B}$ (see Section 3.5). The almost-C^1 splines are then C^1 smooth at all vertices and composed of modified functions from $\mathscr{B}^*$ as well as three new functions associated to each extraordinary vertex.

3.1. Degree-of-freedom structure for $\mathscr{B}^*$

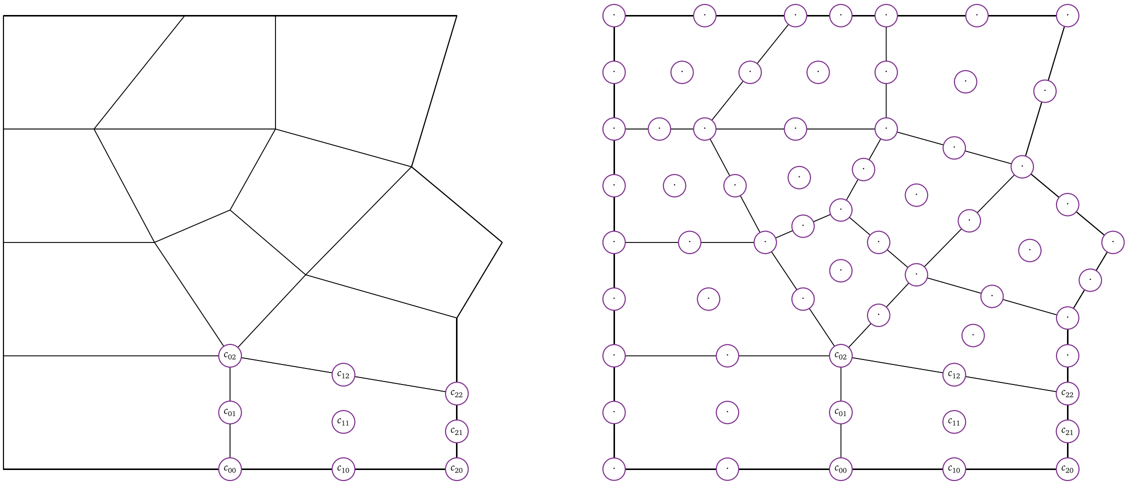
The degrees of freedom, or *dofs* in short, corresponding to $\mathscr{B}^*$ are divided into three categories; the spline construction will be specified for each category separately.

- Face dofs: We associate one degree of freedom to each face of $\mathscr{T}$, i.e., one degree of freedom for each member of $\mathscr{T}_2$.
- Boundary edge dofs: We associate one degree of freedom to each boundary edge of $\mathscr{T}$, i.e., one degree of freedom for each member of $\mathscr{T}_1^B$.
- Corner vertex dofs: We associate one degree of freedom to each corner vertex of $\mathscr{T}$, i.e., one degree of freedom for each member of $\mathscr{T}_0^C$.

We create a basis for $\mathscr{B}^*$ by associating one function to each dof, which we summarize using the index set

$$\mathscr{I}^* := \mathscr{T}_2 \cup \mathscr{T}_1^B \cup \mathscr{T}_0^C \ . \tag{1}$$

For visual consistency, in the topological description (as in Fig. 2(b)), we will denote each dof by placing an unfilled blue circle on the associated face/boundary edge/corner vertex of $\mathscr{T}$, and the connectivity of the dofs will be denoted with thin dotted lines; see Fig. 3.



(a) face-local coefficients

(b) extension to neighbours by 0

Fig. 4. Here we take a submesh of the one from Fig. 2(a) to illustrate how the extraction coefficients will be graphically denoted. Given a face $\sigma \in \mathscr{T}$ and ϕ such that $\phi \cap \sigma = \phi$, B^*_ϕ will be defined graphically by specifying its Bernstein–Bézier coefficients on σ; see figure (a). This description will be extended to neighbouring faces of $\mathscr{T}$ with zero coefficients denoted here with a $\cdot$ in figure (b). See Section 3.2 for further elaboration.

Following this categorization of the dofs, we will also call the associated splines *face*, *boundary edge* and *corner vertex splines*, respectively. Denote these spline functions as B^*_ϕ, $\phi \in \mathscr{I}^*$. The associated spline space on $\mathscr{T}$ is then going to be defined as

$$\mathscr{B}^* := \operatorname{span}\left(B^*_\phi \;:\; \phi \in \mathscr{I}^*\right) \;,$$

where $\mathscr{I}^*$ is defined as in (1).

In Section 3.3, we define the B-splines B^*_ϕ, $\phi \in \mathscr{I}^*$. This will be done by specifying their local polynomial descriptions in terms of Bernstein–Bézier polynomials on quadrilaterals via so-called *extraction matrices*. Extraction matrices specify how the face-local Bernstein–Bézier polynomials can be linearly combined to yield the face-local description of a spline basis function; these were introduced in [64,65] and have been used, for instance, for defining splines on unstructured quadrilateral meshes in [7,45,63] and for multi-degree splines in [66,67]. We explain our extraction matrix convention in Section 3.2, and the B-spline definitions are subsequently presented in Section 3.3.

3.2. *Extraction matrix convention*

We will graphically denote the extraction matrix for each B-spline. For all B-splines, this will be done by specifying their Bernstein–Bézier coefficients on a single face of $\mathscr{T}$; the face-local description will be extended to the neighbours. We elaborate upon this convention here and use Fig. 4 for reference. For visual consistency and to differentiate local Bernstein–Bézier coefficients from spline dofs, all local polynomial coefficients will be displayed inside magenta coloured disks.

Consider a face $\sigma \in \mathscr{T}_2$ and let B^*_ϕ be a spline associated either to σ or to a boundary edge/corner vertex that belongs to σ, i.e., $\phi \cap \sigma = \phi$. Then, we will present the definition of B^*_ϕ graphically as on the left in Fig. 4(a) by specifying 9 coefficients $\{c_{jk}[B^*_\phi;\sigma]\}^2_{j,k=0}$. The shown coefficients are to be interpreted as defining the following local spline description,

$$B_\phi{}^*\big|_\sigma = \sum_{j,k=0}^{2} c_{jk}[B^*_\phi;\sigma]\, b^0_{jk,\square}\;, \tag{2}$$

(a) face-local mesh labels

(b) face spline associated to σ

(c) boundary-edge spline associated to τ_1

(d) corner-vertex spline associated to γ_1

Fig. 5. This figure defines all B-splines that may be associated to a face, boundary edge or a corner vertex. In (a), the local mesh neighbourhood of a face of $\mathscr{T}$ are shown with vertices γ_i and edges τ_i. In (b), the face-local coefficients of a B-spline associated to σ are presented. In (c), assuming that τ_1 is a boundary edge, the face-local coefficients of a B-spline associated to τ_1 are presented. Finally, in (d), assuming that γ_1 is a corner vertex, the face-local coefficients of a B-spline associated to γ_1 are presented.

where $b^0_{jk,\square}$ is the (j,k)th biquadratic Bernstein–Bézier polynomial defined on σ by interpreting it as the unit square $[0,1]^2$ (the origin is placed at the corner with the coefficient c_{00}) and local coordinates $\boldsymbol{\xi} := (u, v)$,

$$b^0_{jk,\square}(\boldsymbol{\xi}) := \binom{2}{j}\binom{2}{k}(1-u)^j(1-v)^k u^j v^k \; . \tag{3}$$

Thus, $c_{jk}[f;\sigma]$ denotes the (j,k)th coefficient of the function f restricted to the face σ.

Finally, the face-local descriptions specified as above on any face σ are extended to any neighbouring face σ' with the help of zero coefficients; see Fig. 4(b). Note that on a planar quadrilateral mesh such a representation corresponds to representing the dof-structure of the C^0-smooth piecewise biquadratic space using domain points, as e.g. in [68]. With this graphical convention in place, let us now define the three different types of splines B^*_ϕ, $\phi \in \mathscr{I}^*$, in the following section.

3.3. The face-local spline representations

Face splines

Consider a face $\sigma \in \mathscr{T}_2$ and let B^*_σ be the B-spline associated to its dof. Let the edges and vertices of σ be numbered as shown at the top in Fig. 5(a). Then, the face-local coefficients for B^*_σ are defined as in Fig. 5(b). The figure uses a *boundary flag* for vertices and edges which is a characteristic function defined as follows for $\phi \in \mathscr{T}_k$, $k = 0, 1$,

$$\chi_{\partial,\phi} = \begin{cases} 1\,, & \phi \in \mathring{\mathscr{T}}_k\,, \\ 0\,, & \text{otherwise}\,. \end{cases} \tag{4}$$

Boundary edge splines

Consider a face $\sigma \in \mathscr{T}_2$. Let the edges and vertices of σ be numbered as shown at the top in Fig. 5(a), and let τ_1 be a boundary edge and $B^*_{\tau_1}$ the B-spline associated to its dof. Then, the face-local coefficients for $B^*_{\tau_1}$ are defined as in Fig. 5(c). The figure uses a *corner flag* for vertices which is a characteristic function defined as follows for $\gamma \in \mathscr{T}_0$,

$$\chi_{\wedge,\gamma} = \begin{cases} 0\,, & \gamma \in \mathscr{T}_0^C\,, \\ 1\,, & \text{otherwise}\,. \end{cases} \tag{5}$$

Corner vertex splines

Consider a face $\sigma \in \mathscr{T}_2$. Let the edges and vertices of σ be numbered as shown at the top in Fig. 5(a). Let γ_1 be a boundary vertex that has been chosen to be a corner vertex, and let $B^*_{\gamma_1}$ the B-spline associated to its dof. Then, the face-local coefficients for $B^*_{\gamma_1}$ are defined as in Fig. 5(d).

3.4. Spline functions

Given the above B-spline definitions, we can create a spline function $f^* \in \mathscr{B}^*$ by linearly combining the B-splines with coefficients $f^*_\phi \in \mathbb{R}$,

$$f^* := \sum_{\phi\in\mathscr{I}^*} f^*_\phi B^*_\phi\,. \tag{6}$$

Following the piecewise definition of B-splines, the above definition is also interpreted in a piecewise manner. That is, for any $\sigma \in \mathscr{T}_2$ and local coordinates $\boldsymbol{\xi}$ as in Eq. (3),

$$f^*\big|_\sigma(\boldsymbol{\xi}) = \sum_{\phi\in\mathscr{I}^*} f^*_\phi B^*_\phi\big|_\sigma(\boldsymbol{\xi})\,, \tag{7}$$

where $B_\phi{}^*\big|_\sigma$ follows from Section 3.3. Similar to (2), we thus have

$$f^*\big|_\sigma(\boldsymbol{\xi}) = \sum_{j,k=0}^{2} c_{jk}[f^*;\sigma]\, b^0_{jk,\square}(\boldsymbol{\xi})\,,$$

where

$$c_{jk}[f^*;\sigma] = \sum_{\phi\in\mathscr{I}^*} f^*_\phi c_{jk}[B^*_\phi;\sigma]\,.$$

These kinds of linear combinations can be used, for instance, to create a bivariate spline geometry $\boldsymbol{x}^* \in (\mathscr{B}^*)^d$. This can be done by choosing appropriate control points $\boldsymbol{x}^*_\phi \in \mathbb{R}^d$, $\phi \in \mathscr{I}^*$, and defining

$$\boldsymbol{x}^* := \sum_{\phi\in\mathscr{I}^*} \boldsymbol{x}^*_\phi B^*_\phi\,. \tag{8}$$

We also have a local representation for the geometry $\boldsymbol{x}^*$,

$$\boldsymbol{c}_{jk}[\boldsymbol{x}^*;\sigma] = \sum_{\phi\in\mathscr{I}^*} \boldsymbol{x}^*_\phi c_{jk}[B^*_\phi;\sigma].$$

The functions $\{B^*_\phi\}_{\phi\in\mathscr{I}^*}$ are linearly independent. They also form a non-negative, local partition of unity, thus the coefficients $\boldsymbol{x}^*_\phi$ can be seen as classical spline control points. See [1] for a discussion of other properties of $\mathscr{B}^*$.

3.5. The almost-C^1 splines $\mathscr{B}$

In the following we discuss how $\mathscr{B}^*$ and its basis can be modified to build almost-C^1 splines spanning the space $\mathscr{B}$. We focus on one specific construction which uses geometric data, i.e., which is based on an underlying geometry mapping $\boldsymbol{x}^* \in (\mathscr{B}^*)^d$, or equivalently on underlying control points $\boldsymbol{x}^*_\phi \in \mathbb{R}^d$, $\phi \in \mathscr{I}^*$, as well as on extraordinary-vertex normals $\boldsymbol{n}_\gamma$, for each extraordinary vertex $\gamma \in \mathscr{T}_0^E$. Note that the construction depends only on a small neighbourhood of each extraordinary vertex, that is, on the control points corresponding to extraordinary faces $\phi \in \mathscr{T}_2^E \subset \mathscr{I}^*$. A completely geometry independent construction is also possible, as developed in Appendix B.

First, the set of degrees of freedom $\mathscr{I}$ for $\mathscr{B}$ is given by the dofs from $\mathscr{I}^*$ enriched by three additional dofs for each extraordinary vertex $\gamma \in \mathscr{T}_0^E$. More precisely, we set

$$\begin{aligned} \mathscr{I} = &\left\{\sigma \in \mathscr{T}_2 \ : \ \exists \gamma \in \mathscr{T}_0 \setminus \mathscr{T}_0^E \text{ s.t. } \gamma \in \sigma\right\} \\ &\bigcup \left\{\tau \in \mathscr{T}_1^B \ : \ \exists \gamma \in \mathscr{T}_0 \setminus \mathscr{T}_0^E \text{ s.t. } \gamma \in \tau\right\} \\ &\bigcup \mathscr{T}_0^C \bigcup \left(\mathscr{T}_0^E \times \{1, 2, 3\}\right) , \end{aligned} \tag{9}$$

and describe a basis B_ϕ, $\phi \in \mathscr{I}$, in the following. Note that there are no splines in $\mathscr{B}$ associated to those faces and boundary edges where all incident vertices are extraordinary. The space $\mathscr{B}$ is defined as the span of B_ϕ,

$$\mathscr{B} := \operatorname{span}\left(B_\phi \ : \ \phi \in \mathscr{I}\right) .$$

B-splines on regular faces are identical

Let $\phi \in \mathscr{I}^*$ be a dof of the spline space $\mathscr{B}^*$. Then we define the corresponding basis function B_ϕ of $\mathscr{B}$ such that

$$B_\phi|_\sigma := B_\phi^*|_\sigma , \quad \forall \sigma \in \mathscr{T}_2 \backslash \mathscr{T}_2^E .$$

B-splines on extraordinary faces are subdivided and truncated

For $\phi \in \mathscr{I}^*$ and $\sigma \in \mathscr{T}_2^E$, we will define $B_\phi|_\sigma$ by modifying (i.e., subdividing and truncating) the local representations $B_\phi^*|_\sigma$; the modification will impose vanishing values and derivatives for B_ϕ at all $\gamma \in \mathscr{T}_0^E$. If the modifications imply that $B_\phi \equiv 0$, then ϕ will not be a dof for the spline space $\mathscr{B}$; otherwise, B_ϕ will be the basis function corresponding to the dof ϕ. In particular, the former situation will arise only if all vertices incident upon ϕ are extraordinary.

First, for any $\sigma \in \mathscr{T}_2^E$, consider the local representation of $B_\phi^*|_\sigma$,

$$B_\phi^*|_\sigma = \sum_{j,k=0}^{2} c_{jk}[B_\phi^*; \sigma] b_{jk,\square}^0 ,$$

and define the matrix of coefficients $c[f; \sigma] := \left(c_{jk}[f; \sigma]\right)_{jk}$. Let the four vertices of σ be ordered 1 through 4 in counter-clockwise manner starting from the bottom-left, as in Fig. 5(a). Then, we locally define B_ϕ to be a modified representation of B_ϕ^* as

$$B_\phi|_\sigma := \sum_{j,k=0}^{3} \hat{c}_{jk}[B_\phi; \sigma] b_{jk,\square}^1 , \tag{10}$$

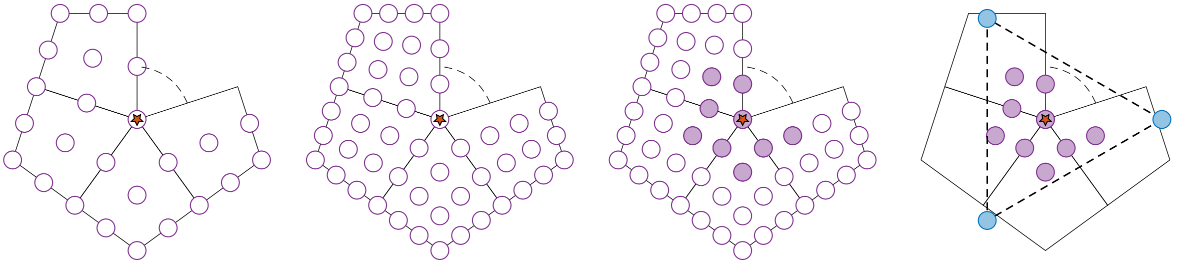
where $b_{jk,\square}^1$ is the (j, k)th C^1 tensor-product B-spline corresponding to the knot vector $(0, 0, 0, \frac{1}{2}, 1, 1, 1)$ in both parametric directions, and

$$\hat{c}[B_\phi; \sigma] := \left(\mathrm{K} c[B_\phi^*; \sigma] \mathrm{K}^T\right) \bigodot_{i=1}^{4} \mathrm{T}_i ,$$

with $\odot$ signifying the Hadamard product (or, element-wise product) of matrices. In the above, K is the univariate B-spline knot-insertion matrix that takes a quadratic Bézier element [0, 1] and inserts a single knot at 0.5,

$$\mathrm{K} = \begin{bmatrix} 1 & & \\ \frac{1}{2} & \frac{1}{2} & \\ & \frac{1}{2} & \frac{1}{2} \\ & & 1 \end{bmatrix} , \tag{11}$$

and T_i is the truncation matrix associated to the ith vertex of σ as ordered above. If the ith vertex is not extraordinary, T_i is defined to be a 4×4 matrix with all entries equal to 1. Else, if the ith vertex is extraordinary, $i = 1, \ldots, 4$,



(a) coefficients $c_{jk}[\boldsymbol{x}^*;\sigma_i]$ (b) coefficients $\hat{c}_{jk}[\boldsymbol{x}^*;\sigma_i]$ (c) projected coefficients (d) control triangle

Fig. 6. An extraordinary vertex $\gamma \in \mathscr{T}_0^E$ surrounded by its 1-ring, composed of the extraordinary faces $\sigma_1, \sigma_2, \dots, \sigma_\mu$. The coefficients $c_{jk}[\boldsymbol{x}^*;\sigma_i]$ as well as the once refined coefficients $\hat{c}_{jk}[\boldsymbol{x}^*;\sigma_i]$ of $\boldsymbol{x}^*$ within the 1-ring neighbourhood of γ are depicted in (a) and (b), respectively. In (c) the relevant coefficients needed to define $B_{(\gamma,\nu)}$, as in (13), are highlighted in purple. The control triangle covering the relevant coefficients is visualized in (d). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

we respectively define

$$\mathrm{T}_1 = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}, \quad \mathrm{T}_2 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \quad \mathrm{T}_3 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}, \quad \mathrm{T}_4 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}. \tag{12}$$

Note that if all vertices of σ are extraordinary, $\hat{c}[B_\phi;\sigma] = \mathbf{0}$ since $\odot_{i=1}^4 \mathrm{T}_i = \mathbf{0}$. In particular, if $\hat{c}[B_\phi;\sigma] = \mathbf{0}$ for σ such that $\sigma \cap \phi = \phi$, then B_ϕ will be globally zero since the corresponding B_ϕ^* is supported only on extraordinary faces and is entirely truncated on each face in its support. Furthermore, for all $\phi \in \mathscr{I}^*$ such that $\phi \cap \gamma = \emptyset$ for all $\gamma \in \mathscr{T}_0^E$, we have $B_\phi = B_\phi^*$. That is, all regular (face, boundary edge and corner) B-splines are unchanged by the subdivision and truncation.

Remark 3.1. A similar construction can be achieved if the functions $b^1_{jk,\Box}$ in (10) are replaced by bicubic polynomials and if K is replaced by a degree elevation matrix instead of the knot insertion matrix, leading to almost-C^1 splines being bicubic polynomials on extraordinary faces. Note that both constructions are computationally equivalent and yield spaces of the same dimension. Moreover, the numerical performance of the two approaches is indistinguishable. Here we present only the construction based on knot insertion since we consider in the following a global mesh refinement scheme based on bisection of all edges and faces. Thus, the knot insertion based construction facilitates this refinement scheme as it yields a purely piecewise-biquadratic spline space. Details on the construction based on degree elevation can be found in Appendix A.

Extraordinary vertex splines $B_{(\gamma,i)}$ are added

Let $\gamma \in \mathscr{T}_0^E$ be an extraordinary vertex of valence μ and let the faces around it be denoted by $\sigma_1, \sigma_2, \dots, \sigma_\mu$. Given a prescribed normal direction $\boldsymbol{n}_\gamma$ at the vertex, cf. Remark 3.3, we introduce a tangent plane through $\boldsymbol{x}^*|_\gamma$ which is orthogonal to the vector $\boldsymbol{n}_\gamma$. Then, the 1-ring $\boldsymbol{x}^*|_{\bigcup_{i=1}^\mu \sigma_i}$ is projected orthogonally onto the tangent plane. We denote this projection by P_γ. Since the construction is affine invariant, this is equivalent to projecting the Bernstein–Bézier coefficients of the function $\boldsymbol{x}^*|_{\sigma_i}$ for each face σ_i, i.e.,

$$P_\gamma(\boldsymbol{x}^*)|_{\sigma_j} = \sum_{j,k=0}^{3} P_\gamma(\hat{\boldsymbol{c}}_{jk}[\boldsymbol{x}^*;\sigma_j])b^1_{jk,\Box}.$$

Remark 3.2. Note that $P_\gamma(\boldsymbol{x}^*)|_{\sigma_j}$ could have self-intersections but that is irrelevant for our construction where we only need the projections of 4 specific face-local coefficients in the vicinity of γ to be regular; see Fig. 6, Eq. (13) and Remark 3.3.

Thus, P_γ results in a planar configuration of faces around the extraordinary vertex, see Fig. 6. We select the local control points

$$\boldsymbol{c}^i_{jk} := P_\gamma(\hat{\boldsymbol{c}}_{jk}[\boldsymbol{x}^*; \sigma_i]), \quad \text{with } i \in \{1, \ldots, \mu\},\ (j,k) \in \{0,1\}^2 \ , \tag{13}$$

which are relevant for the basis construction. These local control points $\boldsymbol{c}^i_{jk}$, as highlighted in Fig. 6(c), are then covered by a triangle $(\boldsymbol{x}_{(\gamma,1)}, \boldsymbol{x}_{(\gamma,2)}, \boldsymbol{x}_{(\gamma,3)})$, shown in Fig. 6(d). The triangle is selected such that it is the smallest triangle that contains all points $\boldsymbol{c}^i_{jk}$ in its interior (or of similar size to the smallest triangle, cf. Remark 3.5). We denote the barycentric coordinates of a point $\boldsymbol{c}$ with respect to the triangle $(\boldsymbol{x}_{(\gamma,1)}, \boldsymbol{x}_{(\gamma,2)}, \boldsymbol{x}_{(\gamma,3)})$ by $\lambda_1(\boldsymbol{c})$, $\lambda_2(\boldsymbol{c})$ and $\lambda_3(\boldsymbol{c})$, i.e., we have

$$\boldsymbol{c} = \lambda_1(\boldsymbol{c})\boldsymbol{x}_{(\gamma,1)} + \lambda_2(\boldsymbol{c})\boldsymbol{x}_{(\gamma,2)} + \lambda_3(\boldsymbol{c})\boldsymbol{x}_{(\gamma,3)} \ ,$$

with $\lambda_1(\boldsymbol{c}) + \lambda_2(\boldsymbol{c}) + \lambda_3(\boldsymbol{c}) = 1$. By construction, we have $\lambda_\nu(\boldsymbol{c}^i_{jk}) \geq 0$ for all $\nu \in \{1,2,3\}$, $(j,k) \in \{0,1\}^2$ and $i \in \{1, \ldots, \mu\}$.

The triangle $(\boldsymbol{x}_{(\gamma,1)}, \boldsymbol{x}_{(\gamma,2)}, \boldsymbol{x}_{(\gamma,3)})$ serves as a control triangle to determine the coefficients of the three new basis functions $B_{(\gamma,\nu)}$, with $\nu \in \{1,2,3\}$. Such control triangles are commonly used for defining smooth splines on triangulations and go back to [69]. The coefficients of $B_{(\gamma,\nu)}|_{\sigma_i}$ are given as the barycentric coordinates corresponding to the local control point $\boldsymbol{x}_{(\gamma,\nu)}$, i.e., the function $B_{(\gamma,\nu)}$ is defined to be

$$B_{(\gamma,\nu)}|_{\sigma_i} = \sum_{j,k=0}^{3} \hat{c}_{jk}[B_{(\gamma,\nu)}; \sigma_i] b^1_{jk,\square} \ , \tag{14}$$

with

$$\hat{c}_{jk}[B_{(\gamma,\nu)}; \sigma_i] = \begin{cases} \lambda_\nu(\boldsymbol{c}^i_{jk}) \ , & (j,k) \in \{0,1\}^2 \ , \\ 0 \ , & \text{otherwise} \ . \end{cases}$$

Moreover, $B_{(\gamma,\nu)}|_\sigma = 0$ for all $\sigma \notin \{\sigma_1, \sigma_2, \ldots, \sigma_\mu\}$.

Once the above process is repeated for all extraordinary points, the geometry description is updated from $\boldsymbol{x}^*$ to $\boldsymbol{x} \in \mathscr{B}^d$, with the latter defined to be

$$\boldsymbol{x} = \sum_{\phi \in \mathscr{I}} \boldsymbol{x}_\phi B_\phi \ . \tag{15}$$

Remark 3.3. The prescribed normal vector $\boldsymbol{n}_\gamma$ must be given such that the projection is well-defined in a neighbourhood of the extraordinary vertex, i.e., $\boldsymbol{n}_\gamma \neq \boldsymbol{0}$, and the projected surface $P_\gamma(\boldsymbol{x}^*)$ is regular *in a neighbourhood* of γ. This implies a mild regularity assumption on the geometry $\boldsymbol{x}^*$. If no normal vector is prescribed, a suitable normal vector can be constructed e.g. through averaging of local normals in a neighbourhood of γ.

Remark 3.4. The projection onto the tangent plane prescribed by $\boldsymbol{n}_\gamma$ introduces a dependence of the almost-C^1 splines on the geometry $\boldsymbol{x}^*$. This is not necessary. Instead, for all valences that appear in the mesh $\mathscr{T}$, one can also prescribe regular templates that enforce coplanarity in a geometry independent manner. The resulting coefficients for such templates are given in Appendix B for some common valences.

Remark 3.5. The choice of the control triangle has no effect on the resulting space but only on the properties of the constructed basis. Thus, the properties desired from the basis can also inform the choice of the control triangle. We briefly mention two cases here: one concerning the conditioning of the basis, and another concerning basis functions at the boundary. First, regarding conditioning, the control triangle should be chosen to be sufficiently regular (such that the smallest angle is bounded away from zero) and not too large. See [70] where control triangles were used to specify values and derivatives for splines over triangulations. Second, in case of an extraordinary vertex γ at the boundary, one can select the control triangle to obtain a basis that behaves like the usual C^1 quadratic B-spline basis when restricted to the boundary. Let $\tau_1, \tau_2 \in \mathscr{T}_1^B$ be the two spoke edges containing γ, and let the projected control points that correspond to one or both of these boundary edges be $\boldsymbol{c}_1^1$, $\boldsymbol{c}_1^2$ and $\boldsymbol{c}_0 := \frac{\boldsymbol{c}_1^1+\boldsymbol{c}_1^2}{2}$. Then all other projected control points in (13) lie on one side of the line through $\boldsymbol{c}_1^1$ and $\boldsymbol{c}_1^2$. Thus, choosing a control triangle such that this line contains one of its edges, only two of the basis functions $B_{(\gamma,\nu)}$ will be non-zero restricted to the mesh boundary; the effect of the corresponding control points on the boundary will be analogous to the effect of the control points of a univariate C^1 quadratic spline curve.

3.6. Properties of almost-C^1 splines

We collect the properties of $\boldsymbol{x}$ and $\mathscr{B}$ in the following results. In particular, the definitions of spline functions outlined in Section 3.5 immediately imply properties that are useful in numerical simulations.

Lemma 3.6. *For $\gamma \in \mathscr{T}_0^E$, consider the set of local control points $\{\boldsymbol{x}_\phi \;:\; \gamma \in \phi \in \mathscr{I}^*\}$. Let all control points in this set be coplanar, with the common plane defined by the normal $\boldsymbol{n}_\gamma$. Then, in Eq. (13), $\boldsymbol{c}^i_{jk} = \hat{\boldsymbol{c}}_{jk}[\boldsymbol{x}^*;\sigma_i]$.*

Corollary 3.7. *For all $\gamma \in \mathscr{T}_0^E$, let the set of local control points $\{\boldsymbol{x}_\phi \;:\; \gamma \in \phi \in \mathscr{I}^*\}$ be coplanar with the common plane defined by the normal $\boldsymbol{n}_\gamma$. Then, $\boldsymbol{x} = \boldsymbol{x}^*$.*

Proposition 3.8.

(a) The total number of dofs satisfies

$$\mathfrak{n} := |\mathscr{I}| \leq |\mathscr{T}_2| + |\mathscr{T}_1^B| + |\mathscr{T}_0^C| + 3|\mathscr{T}_0^E| \,. \tag{16}$$

If no face and no boundary edge of the mesh contains only extraordinary vertices, then the above relation becomes an equality.

(b) Non-negativity: On any $\sigma \in \mathscr{T}_2$ and any $\phi \in \mathscr{I}$, $B_\phi\big|_\sigma \geq 0$.

(c) Partition of unity: On any $\sigma \in \mathscr{T}_2$, $\sum_{\phi\in\mathscr{I}} B_\phi\big|_\sigma \equiv 1$.

(d) Local support: If B_ϕ is such that $\phi \in \mathscr{T}_k$, $k = 0, 1, 2$, then $B_\phi\big|_\sigma = 0$ for any $\sigma \in \mathscr{T}_2$ such that $\sigma \cap \phi = \emptyset$. Similarly, if $\gamma \in \mathscr{T}_0^E$ with neighbouring faces $\{\sigma_1, \ldots, \sigma_\mu\}$, then $B_{(\gamma,k)}|_\sigma = 0$ for any $\sigma \in \mathscr{T}_2 \setminus \{\sigma_1, \ldots, \sigma_\mu\}$.

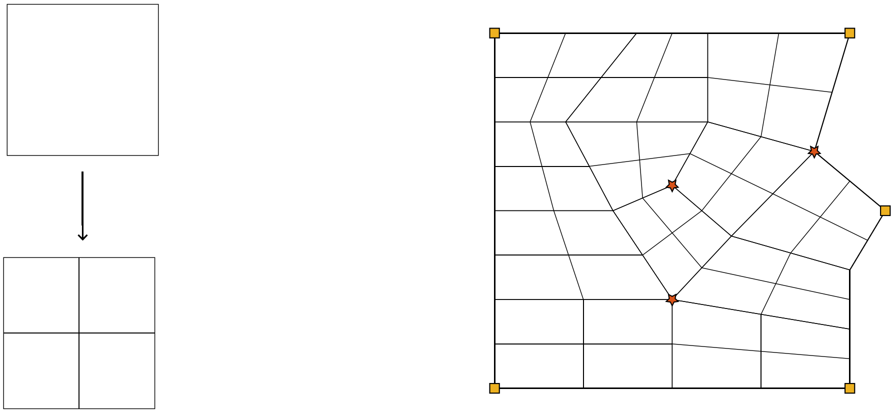
(e) Boundary Kronecker–Delta: All B_ϕ with $\phi \in \mathscr{T}_2$ are identically zero on the boundary of $\mathscr{T}$.

(f) Linear independence: $\{B_\phi : \phi \in \mathscr{I}\}$ form a basis for $\mathscr{B}$.

Proof. Property (a) can be shown by a simple counting exercise. Properties (b)–(d) can be deduced easily from the representation of the splines in Bernstein–Bézier form, or in local B-spline form as in (10), on each face. One can see directly that all Bernstein–Bézier (and B-spline) coefficients are non-negative (in case of the extraordinary vertex splines this is due to the control triangle being large enough), thus the basis functions are non-negative. Due to the truncation and the control triangle construction, all Bernstein–Bézier (and B-spline) coefficients of functions sum to one, thus all functions sum to one. The local support property can also be deduced directly from the Bernstein–Bézier form. Property (e) follows by definition, cf. Fig. 5(b). The linear independence stated in property (f) can be derived from the linear independence of the basis functions $\{B^*_\phi \;:\; \phi \in \mathscr{I}^*\}$ for $\mathscr{B}^*$, which was shown in [1]. After the local modifications, the truncated and non-truncated functions stay linearly independent and the new functions that are introduced for each extraordinary vertex are linearly independent with respect to the others as well as between themselves, due to the control triangle being non-degenerate. This concludes the proof. ■

Proposition 3.9. *The almost-C^1 splines spanning $\mathscr{B}$ are C^1 at all vertices, that is, any surface $\boldsymbol{x} \in \mathscr{B}^d$ possesses a well-defined tangent plane in every vertex, if the parameterization is regular. Moreover, the splines are C^1 across all edges between regular vertices (i.e. inner vertices of valency four and boundary vertices of valency two or three) and C^0 across all other edges.*

Proof. The C^1 smoothness at all regular vertices and across all edges between regular vertices follows from the fact that the almost-C^1 splines are locally equivalent to biquadratic tensor-product B-splines, cf. [1]. All functions associated to faces, boundary edges and corner vertices are C^1 at all extraordinary vertices due to the truncation. All extraordinary vertex functions are C^1 smooth in an isogeometric sense at the associated vertex due to the local B-spline control points being coplanar. Finally, the global C^0 smoothness follows by construction. ■



(a) refining a quadrilateral face

(b) refining the mesh from Figure 2(a)

Fig. 7. The above figure shows how the mesh faces are split when performing refinement. Each quadrilateral face is split into 4 new faces as shown in (a). Figure (b) shows how the mesh topology changes after refinement for the mesh previously shown in Fig. 2(a); it is also shown how the corner/extraordinary vertices retain their labels after refinement. See Section 4 for details.

In contrast to $\mathscr{B}^*$, which is only C^0 smooth at extraordinary vertices, the almost-C^1 splines spanning $\mathscr{B}$ are C^1 at all vertices, assuming that the spline control points are in a non-degenerate position. The tangent plane is orthogonal to the prescribed normal vector $\boldsymbol{n}_\gamma$ for each extraordinary vertex $\gamma \in \mathscr{T}_0^E$. Furthermore, for any spline surface $\boldsymbol{x} \in \mathscr{B}^d$ or $\boldsymbol{x}^* \in (\mathscr{B}^*)^d$ the boundary of the domain can be interpreted as a collection of quadratic B-spline curve segments with uniform (open) knot vectors. Similar to tensor-product B-splines, rational representations derived from $\mathscr{B}$ yield boundary curves that are quadratic NURBS. If the control points also satisfy $\boldsymbol{x}_\gamma = \boldsymbol{x}^*_\gamma$ for all $\gamma \in \mathscr{T}_0^C$, $\boldsymbol{x}_\tau = \boldsymbol{x}^*_\tau$ for all $\tau \in \mathscr{T}_1^B$, then the control points corresponding to boundary extraordinary vertices $\gamma \in \mathscr{T}_0^E \cap \mathscr{T}_0^B$ can be chosen such that the boundary curves of $\boldsymbol{x}$ and $\boldsymbol{x}^*$ coincide.

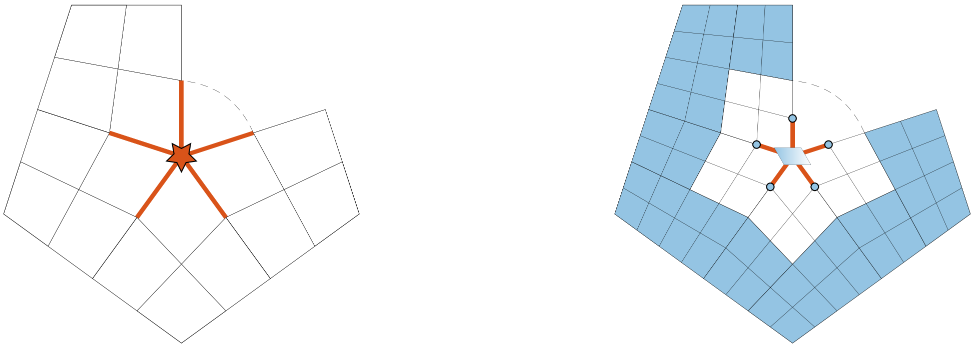
In Appendix C we present an alternative construction of almost-C^1 splines, which does not rely on subdividing the faces near extraordinary vertices. This results in a space that possesses no dofs corresponding to extraordinary faces, while three dofs per extraordinary vertex remain. The functions are biquadratic polynomials on all faces.

4. Refining almost-C^1 splines

Refinement of the mesh can help improve the resolving power of splines for the purpose of, for instance, obtaining a better approximation to the solution of a PDE. Moreover, in our setting, the spline function and the surface will also become smoother by refining as the region of reduced smoothness will shrink. Assume that we are given a mesh $\mathscr{T}$, the associated almost-C^1 spline space $\mathscr{B}$, and control points $\boldsymbol{x}_\phi$, $\phi \in \mathscr{I}$, that define a spline geometry $\boldsymbol{x}$. In this section, we outline precisely how they can be refined. We start by describing the refinement of $\mathscr{T}$ in Section 4.1; next, in Section 4.2 we explain the motivation behind the refinement scheme for $\boldsymbol{x}$ and $\mathscr{B}$, the latter is described in Sections 4.3–4.5. Finally, we discuss some properties of the refinement scheme in Section 4.6. Our refinement scheme is closely related to the one presented recently in [1].

4.1. Refining $\mathscr{T}$

To refine $\mathscr{T}$, which only contains topological information, we only need to specify how the connectivity and quadrilateral-composition of $\mathscr{T}$ are to be updated. In this document, we adopt a simple global refinement approach whereby all quadrilaterals are split into 2 × 2 quadrilaterals; see Fig. 7. We will denote refined quantities with a "hat" — for instance, the refined mesh will be denoted as $\widehat{\mathscr{T}}$. During this process, no hanging nodes are introduced, i.e., $|\mathscr{T}_1| + |\mathscr{T}_2|$ new mesh vertices are added. We assume that the old mesh vertices retain their labels as per Section 2. Moreover, since the old vertices retain their labels, the corner and extraordinary vertices of $\widehat{\mathscr{T}}$ are



(a) coarse neighborhood (b) refined neighborhood

Fig. 8. The above schematic depicts the motivation for our non-nested refinement scheme (c.f. Section 4.2 and Proposition 4.3). In (a) is a 2-neighbourhood of an extraordinary vertex; it is assumed that except the central vertex, all other vertices in this neighbourhood are regular. In (b) is the refinement of the coarse neighbourhood from (a). In both (a) and (b), splines (in $\mathscr{B}$ and $\widehat{\mathscr{B}}$, respectively) are not C^1 smooth across the *interiors* of the bold red edges. In (b), the flat shaded faces correspond to those where the refined spline geometry is identical to the coarse geometry; the filled disks correspond to points where the refined spline geometry interpolates the coarse geometry; and the central shaded parallelogram indicates that the coarse and refined spline geometries have identical normals at the extraordinary vertex. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

respectively chosen to be identical to the corner and extraordinary vertices of $\mathscr{T}$, i.e., $\widehat{\mathscr{T}}_0^C = \mathscr{T}_0^C$ and $\widehat{\mathscr{T}}_0^E = \mathscr{T}_0^E$. Finally, the refined dof index sets $\widehat{\mathscr{I}}^*$ and $\widehat{\mathscr{I}}$ are defined for $\widehat{\mathscr{T}}$ exactly as outlined in Section 3. We directly obtain

$$\widehat{\mathscr{I}} = \widehat{\mathscr{T}_2} \bigcup \widehat{\mathscr{T}_1^B} \bigcup \widehat{\mathscr{T}_0^C} \bigcup (\widehat{\mathscr{T}_0^E} \times \{1, 2, 3\}).$$

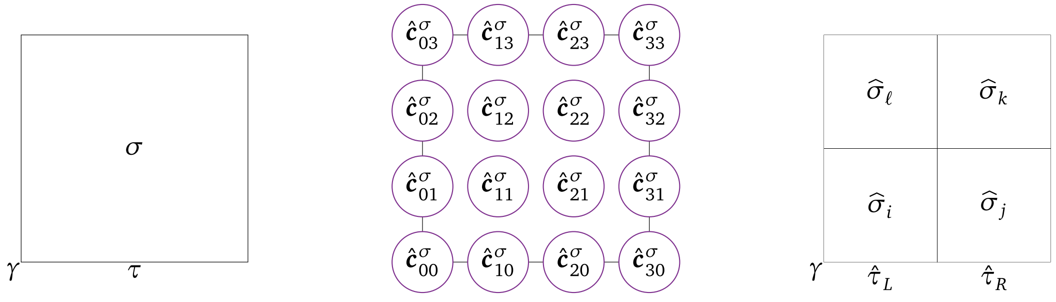
4.2. Motivation behind refinement of $\boldsymbol{x}$ and $\mathscr{B}$

In the next three subsections, we describe the refinement scheme in three steps. First, in Section 4.3, we use B-spline knot insertion to refine the face-local restrictions $\boldsymbol{x}|_\sigma$ on all faces σ of $\mathscr{T}$. Let the thus refined face-local control points for σ be indexed as $\widehat{\boldsymbol{c}}^{\sigma}_{jk}$, $0 \le j, k \le 3$. Next, in Section 4.4, we obtain the refined spline control points $\widehat{\boldsymbol{x}}^*_\phi$, $\phi \in \mathscr{I}^*$, as linear combinations of the control points $\widehat{\boldsymbol{c}}_{jk,\sigma}$. These define a geometry $\widehat{\boldsymbol{x}}^* \in (\widehat{\mathscr{B}}^*)^d$ on $\widehat{\mathscr{T}}$. Finally, in Section 4.5, we use $\widehat{\boldsymbol{x}}^*$ and the extraordinary-vertex normals $\boldsymbol{n}_\gamma$, $\gamma \in \widehat{\mathscr{T}}_0^E$, to define both $\widehat{\mathscr{B}}$ and $\widehat{\boldsymbol{x}}$. It is worth mentioning that, as shown in Section 4.6, $\widehat{\boldsymbol{x}}$ will be identical to $\widehat{\boldsymbol{x}}^*$.

Thus, the refinement scheme as outlined above will only ensure that $\widehat{\boldsymbol{x}} = \widehat{\boldsymbol{x}}^* \approx \boldsymbol{x}$ and, in general, the spline spaces will be non-nested, i.e., $\mathscr{B} \not\subset \widehat{\mathscr{B}}^*$ and $\mathscr{B} \not\subset \widehat{\mathscr{B}}$. We opt for such non-nested refinements because nested refinements would necessarily require involved bookkeeping (e.g., introduction of additional dofs with a refinement-level-dependent structure [7,63]) which we seek to avoid.

The non-nested refinement scheme implies that the map $\boldsymbol{x} \mapsto \widehat{\boldsymbol{x}}$ can be specified in different ways. For instance, if there is a 'true geometry' $\boldsymbol{X}$ (e.g., the geometry at the coarsest refinement level or an underlying smooth surface), then at any given refinement level $\widehat{\boldsymbol{x}}$ can be computed so as to minimize $\|\widehat{\boldsymbol{x}} - \boldsymbol{X}\|^2$ in a suitable norm. Alternatively, the refined geometry $\widehat{\boldsymbol{x}}$ can be computed so that certain desirable properties of $\boldsymbol{x}$ are preserved — this is the approach we adopt. We formulate a refinement scheme that only uses local information and achieves two objectives. Firstly, it ensures that $\boldsymbol{x}$ and $\widehat{\boldsymbol{x}}$ are equal in the structured parts of the mesh $\mathscr{T}$ — in these parts $\boldsymbol{x}$ coincides with a tensor-product biquadratic spline, and thus B-spline knot insertion is sufficient for achieving this objective. Secondly, certain values and derivatives of $\boldsymbol{x}$ and $\widehat{\boldsymbol{x}}$ are equal in the unstructured parts of $\mathscr{T}$. See Fig. 8 for a more precise visual overview of the scheme.

The refinement scheme will be depicted graphically in the following sections. The old control points will not be shown in the figures and, following our earlier convention, the new control points will be shown as filled blue circles. To further declutter the figures, only the indices of the new control points will be annotated.



(a) coarse mesh face σ (b) refined face-local control points (c) mesh faces after refining σ

Fig. 9. The above figures show (a) a face σ that belongs to the coarse mesh $\mathscr{T}$, (b) the refined face-local control points corresponding to the coarse spline geometry $\boldsymbol{x}|_\sigma$ (see Section 4.3), and (c) the faces of $\widehat{\mathscr{T}}$ obtained by refinements of σ. In particular, these figures are used as reference to explain the refinement scheme in Section 4.4.

4.3. *Refining face-local representations $\boldsymbol{x}|_\sigma$*

On face σ of $\mathscr{T}$, consider the restriction $\boldsymbol{x}|_\sigma$. If $\sigma \in \mathscr{T}_2 \backslash \mathscr{T}_2^E$, then $\boldsymbol{x}|_\sigma$ is a linear combination of the Bernstein–Bézier polynomials $b^0_{jk,\square}$ for some face-local control points $\boldsymbol{c}_{jk}[\boldsymbol{x};\sigma]$,

$$\boldsymbol{x}|_\sigma =: \sum_{j,k=0}^{2} \boldsymbol{c}_{jk}[\boldsymbol{x};\sigma] b^0_{jk,\square}\ . \tag{17}$$

Similarly, if $\sigma \in \mathscr{T}_2^E$, then $\boldsymbol{x}|_\sigma$ is a linear combination of the C^1 biquadratic B-splines $b^1_{jk,\square}$ for some face-local control points $\boldsymbol{c}_{jk}[\boldsymbol{x};\sigma]$,

$$\boldsymbol{x}|_\sigma =: \sum_{j,k=0}^{3} \boldsymbol{c}_{jk}[\boldsymbol{x};\sigma] b^1_{jk,\square}\ . \tag{18}$$

Then, for any $\sigma \in \mathscr{T}_2$ and with the matrix $\boldsymbol{c}[\boldsymbol{x};\sigma] := \big(\boldsymbol{c}_{jk}[\boldsymbol{x};\sigma]\big)_{jk}$, we define the face-local refined control points $\hat{\boldsymbol{c}}_{jk}[\boldsymbol{x};\sigma]$, $0 \le j,k \le 3$, as

$$\hat{\boldsymbol{c}}[\boldsymbol{x};\sigma] = \begin{cases} K\boldsymbol{c}[\boldsymbol{x};\sigma]K^T\ , & \sigma \in \mathscr{T}_2 \backslash \mathscr{T}_2^E\ , \\ \boldsymbol{c}[\boldsymbol{x};\sigma]\ , & \sigma \in \mathscr{T}_2^E\ , \end{cases} \tag{19}$$

where K is the univariate B-spline knot-insertion matrix as in (11). Note that for all faces σ in $\mathscr{T}_2$, we have

$$\boldsymbol{x}|_\sigma = \sum_{j,k=0}^{3} \hat{\boldsymbol{c}}_{jk}[\boldsymbol{x};\sigma] b^1_{jk,\square}\ . \tag{20}$$

For later reference, we will use the schematic shown in Fig. 9 where we use the shorthand $\hat{\boldsymbol{c}}^{\sigma}_{jk} := \hat{\boldsymbol{c}}_{jk}[\boldsymbol{x};\sigma]$ for convenience.

4.4. *Defining $\widehat{\boldsymbol{x}}^*$ and $\widehat{\mathscr{B}}^*$*

The spline space $\widehat{\mathscr{B}}^*$ and the associated dof index set $\widehat{\mathscr{I}}^*$ are defined on $\widehat{\mathscr{T}}$ following the approach in Section 3.3. Then, we define a geometry $\widehat{\boldsymbol{x}}^* \in (\widehat{\mathscr{B}}^*)^d$ by computing the associated control points $\widehat{\boldsymbol{x}}^*_\phi$, $\phi \in \widehat{\mathscr{I}}^*$, as linear combinations of the face-local control points $\hat{\boldsymbol{c}}[\boldsymbol{x};\sigma]$, $\sigma \in \mathscr{T}_2$, from Section 4.3. We split this computation in four parts: corner vertex control points, boundary-edge control points, face control points in locally structured regions, and face control points in locally unstructured regions.

Corner vertex control points

As mentioned earlier, the number of corner vertices remains fixed during refinement (see Fig. 7 for an example) and these vertices retain their labels. With reference to Fig. 9(a) and (c), let $\gamma \in \widehat{\mathscr{T}}_0^C = \mathscr{T}_0^C$ and let $\widehat{\boldsymbol{x}}^*_\gamma$ be the refined spline control point associated to it. Then, we set

$$\widehat{\boldsymbol{x}}^*_\gamma := \hat{\boldsymbol{c}}_{00}[\boldsymbol{x};\sigma]\ . \tag{21}$$

Boundary-edge control points

With reference to Fig. 9(a), let $\tau \in \mathscr{T}_1^B$ be refined into two new boundary edges $\widehat{\tau}_L, \widehat{\tau}_R \in \widehat{\mathscr{T}}_1^B$, as shown in Fig. 9(c). Then, the control points corresponding to $\widehat{\tau}_L$ and $\widehat{\tau}_R$, respectively denoted as $\widehat{\boldsymbol{x}}^*_L$ and $\widehat{\boldsymbol{x}}^*_R$ here, are defined as,

$$\widehat{\boldsymbol{x}}^*_L := \hat{\boldsymbol{c}}_{10}[\boldsymbol{x};\sigma]\ ,\quad \widehat{\boldsymbol{x}}^*_R := \hat{\boldsymbol{c}}_{20}[\boldsymbol{x};\sigma]\ .$$

Face control points: locally structured regions

A locally structured part of $\mathscr{T}$ is composed of boundary edges and faces that do not contain any extraordinary vertices. With reference to Fig. 9(a), let $\sigma \in \mathscr{T}_2$ be a face containing no extraordinary vertices. Let σ be refined into four new faces $\widehat{\sigma}_i, \widehat{\sigma}_j, \widehat{\sigma}_k$ and $\widehat{\sigma}_\ell$, as shown in Fig. 9(c). Then, the corresponding control points, respectively denoted as $\widehat{\boldsymbol{x}}^*_i, \widehat{\boldsymbol{x}}^*_j, \widehat{\boldsymbol{x}}^*_k$ and $\widehat{\boldsymbol{x}}^*_\ell$, are defined as,

$$\left(\widehat{\boldsymbol{x}}^*_i, \widehat{\boldsymbol{x}}^*_j, \widehat{\boldsymbol{x}}^*_k, \widehat{\boldsymbol{x}}^*_\ell\right) := \left(\hat{\boldsymbol{c}}_{11}[\boldsymbol{x};\sigma],\ \hat{\boldsymbol{c}}_{21}[\boldsymbol{x};\sigma],\ \hat{\boldsymbol{c}}_{22}[\boldsymbol{x};\sigma],\ \hat{\boldsymbol{c}}_{12}[\boldsymbol{x};\sigma]\right)\ .$$

Face control points: locally unstructured regions

An unstructured part of $\mathscr{T}$ is a face that contains one or more extraordinary vertices. Since we opt for non-nested refinements, it is not possible to exactly preserve the geometry in the unstructured parts of $\mathscr{T}$ during refinement. Nevertheless, the following approach ensures that $\widehat{\boldsymbol{x}}^*$ interpolates the midpoints of the coarse spoke edges when all faces of $\mathscr{T}$ contain at most one extraordinary vertex.

First, with reference to Fig. 9(a), let $\sigma \in \mathscr{T}_2$ be a face containing either more than one extraordinary vertices, or containing an extraordinary vertex that is in the 1-ring of another extraordinary vertex. This σ is being refined into four new faces $\widehat{\sigma}_i, \widehat{\sigma}_j, \widehat{\sigma}_k$ and $\widehat{\sigma}_\ell$, as shown in Fig. 9(c). Then, similarly to the locally structured case, the corresponding control points, respectively denoted as $\widehat{\boldsymbol{x}}^*_i, \widehat{\boldsymbol{x}}^*_j, \widehat{\boldsymbol{x}}^*_k$ and $\widehat{\boldsymbol{x}}^*_\ell$, are defined as

$$\left(\widehat{\boldsymbol{x}}^*_i, \widehat{\boldsymbol{x}}^*_j, \widehat{\boldsymbol{x}}^*_k, \widehat{\boldsymbol{x}}^*_\ell\right) := \left(\hat{\boldsymbol{c}}_{11}[\boldsymbol{x};\sigma],\ \hat{\boldsymbol{c}}_{21}[\boldsymbol{x};\sigma],\ \hat{\boldsymbol{c}}_{22}[\boldsymbol{x};\sigma],\ \hat{\boldsymbol{c}}_{12}[\boldsymbol{x};\sigma]\right)\ .$$

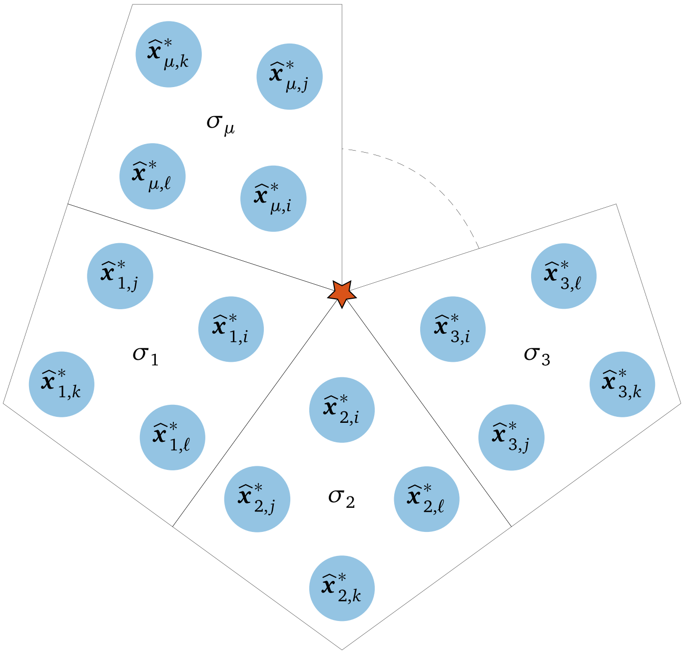
Now we tackle the final remaining case: refinement of faces around an extraordinary vertex that is not contained in the 1-ring of any other extraordinary vertex. We first consider the case where this extraordinary vertex is an interior vertex and later when it is a boundary vertex.

Let $\gamma \in \mathscr{T}_0^E \cap \mathring{\mathscr{T}}_0$ be an interior extraordinary vertex of valence μ and let the labelling of the coarse faces and their refined face-local control points in the neighbourhood of γ be as in Fig. 10(a). Then, the corresponding refined control points $\widehat{\boldsymbol{x}}^*_{r,i}, \widehat{\boldsymbol{x}}^*_{r,j}, \widehat{\boldsymbol{x}}^*_{r,k}$ and $\widehat{\boldsymbol{x}}^*_{r,\ell}$, $r = 1, \ldots, \mu$, are computed as below, where we employ the shorthand $\widehat{\boldsymbol{c}}^r_{jk} := \hat{\boldsymbol{c}}_{jk}[\boldsymbol{x};\sigma_r]$,

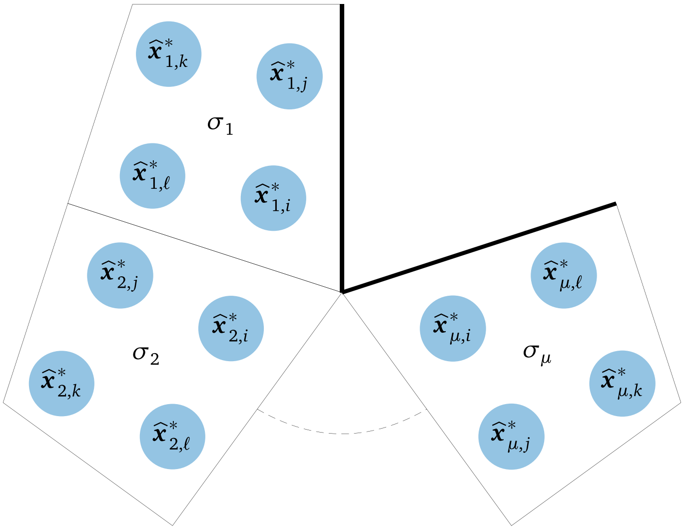
$$\begin{aligned}
&\begin{bmatrix}\widehat{\boldsymbol{x}}^*_{r,j}\\ \widehat{\boldsymbol{x}}^*_{r,k}\\ \widehat{\boldsymbol{x}}^*_{r,\ell}\end{bmatrix} = \begin{bmatrix}\widehat{\boldsymbol{c}}^r_{21}\\ \widehat{\boldsymbol{c}}^r_{22}\\ \widehat{\boldsymbol{c}}^r_{12}\end{bmatrix}\ ,\qquad r = 1, \ldots, \mu\ ,\\
&\begin{bmatrix}\widehat{\boldsymbol{x}}^*_{1,i}\\ \widehat{\boldsymbol{x}}^*_{2,i}\\ \vdots\\ \widehat{\boldsymbol{x}}^*_{\mu,i}\end{bmatrix} = \mathring{\mathrm{S}}_\mu \begin{bmatrix}\widehat{\boldsymbol{c}}^1_{01}\\ \widehat{\boldsymbol{c}}^2_{01}\\ \vdots\\ \widehat{\boldsymbol{c}}^\mu_{01}\end{bmatrix} + \mathring{\mathrm{Q}}_\mu \begin{bmatrix}\widehat{\boldsymbol{c}}^1_{11}\\ \widehat{\boldsymbol{c}}^2_{11}\\ \vdots\\ \widehat{\boldsymbol{c}}^\mu_{11}\end{bmatrix}\ ,
\end{aligned} \tag{22}$$

where $\mathring{\mathrm{S}}_\mu$ and $\mathring{\mathrm{Q}}_\mu$ are circulant matrices. If μ is odd, they are defined to be

$$\mathring{\mathrm{S}}_\mu = \mathrm{circulant}\big(1,\ -1,\ 1,\ -1, \ldots,\ -1,\ 1\big)\ ,\quad \mathring{\mathrm{Q}}_\mu = \boldsymbol{0}\ , \tag{23}$$



(a) interior extraordinary vertex



(b) boundary extraordinary vertex

Fig. 10. The above shows the refinement of face control points in the locally unstructured regions of the mesh. Let $\sigma_1, \ldots, \sigma_\mu \in \mathscr{T}_2$ be quadrilaterals that share an extraordinary vertex and, moreover, this is the only extraordinary vertex that each face contains; figure (a) shows the case for an interior extraordinary vertex and figure (b) for a boundary extraordinary vertex. Then, the refined control points $\widehat{\boldsymbol{x}}^*_{m,i}$, $\widehat{\boldsymbol{x}}^*_{m,j}$, $\widehat{\boldsymbol{x}}^*_{m,k}$ and $\widehat{\boldsymbol{x}}^*_{m,\ell}$ are obtained using Eqs. (22), (23) and (24) for figure (a), and Eqs. (25) and (26) for figure (b). Note that the labelling of face control points with the subscripts i, j, k, ℓ corresponds to the refined face labelling shown in Fig. 9(c), assuming that the extraordinary vertex in the above figures coincides with γ in Fig. 9(a).

and if μ is even, they are defined to be

$$\begin{aligned} \mathring{\mathrm{S}}_\mu &= \text{circulant}\left(2-\frac{2}{\mu},\ -\left(2-\frac{4}{\mu}\right),\ 2-\frac{6}{\mu},\ -\left(2-\frac{8}{\mu}\right),\dots,\ 0\right)\ , \\ \mathring{\mathrm{Q}}_\mu &= \frac{1}{\mu}\text{circulant}\left(1,\ -1,\ 1,\ -1,\dots,\ -1\right)\ . \end{aligned} \tag{24}$$

Next, let $\gamma \in \mathscr{T}_0^E \cap \mathscr{T}_0^B$ be a boundary extraordinary vertex of valence μ, and let the labelling of the coarse faces and their refined face-local control points in the neighbourhood of γ be as in Fig. 10(b). Then, the corresponding refined control points $\widehat{\boldsymbol{x}}^*_{r,i}, \widehat{\boldsymbol{x}}^*_{r,j}, \widehat{\boldsymbol{x}}^*_{r,k}$ and $\widehat{\boldsymbol{x}}^*_{r,\ell}$, $r = 1,\dots,\mu$, are computed as below, where we again employ the shorthand $\widehat{\boldsymbol{c}}^r_{jk} := \hat{\boldsymbol{c}}_{jk}[\boldsymbol{x};\sigma_r]$,

$$\begin{aligned} &\begin{bmatrix} \widehat{\boldsymbol{x}}^*_{r,j} \\ \widehat{\boldsymbol{x}}^*_{r,k} \\ \widehat{\boldsymbol{x}}^*_{r,\ell} \end{bmatrix} = \begin{bmatrix} \widehat{\boldsymbol{c}}^r_{21} \\ \widehat{\boldsymbol{c}}^r_{22} \\ \widehat{\boldsymbol{c}}^r_{12} \end{bmatrix}\ ,\quad r = 1,\dots,\mu\ , \\ &\begin{bmatrix} \widehat{\boldsymbol{x}}^*_{1,i} \\ \widehat{\boldsymbol{x}}^*_{2,i} \\ \vdots \\ \widehat{\boldsymbol{x}}^*_{\mu,i} \end{bmatrix} = \mathrm{S}^\partial_\mu \begin{bmatrix} \widehat{\boldsymbol{c}}^1_{01} \\ \widehat{\boldsymbol{c}}^2_{01} \\ \vdots \\ \widehat{\boldsymbol{c}}^{\mu-1}_{01} \end{bmatrix} + \mathrm{Q}^\partial_\mu \begin{bmatrix} \widehat{\boldsymbol{c}}^1_{11} \\ \widehat{\boldsymbol{c}}^2_{11} \\ \vdots \\ \widehat{\boldsymbol{c}}^{\mu}_{11} \end{bmatrix}\ , \end{aligned} \tag{25}$$

where S^∂_μ is defined to be

$$\mathrm{S}^\partial_\mu = \frac{\mathrm{R}_\mu + \mathrm{J}_\mu \mathrm{R}_\mu \mathrm{J}_{\mu-1}}{2}\ , \tag{26}$$

and Q^∂_μ, R_μ and J_k are defined to be the following matrices of sizes $\mu \times \mu$, $\mu \times (\mu - 1)$ and $k \times k$, $k \geq 1$, respectively,

$$\mathrm{Q}^\partial_\mu = \frac{1}{\mu}\left((-1)^{j+k}\right)_{jk}\ ,\quad \mathrm{R}_\mu = \begin{bmatrix} 4-\frac{4}{\mu} & -\left(4-\frac{8}{\mu}\right) & \left(4-\frac{12}{\mu}\right) & \cdots & (-1)^\mu\frac{4}{\mu} \\ & \left(4-\frac{8}{\mu}\right) & -\left(4-\frac{12}{\mu}\right) & \cdots & (-1)^{\mu+1}\frac{4}{\mu} \\ & & \left(4-\frac{12}{\mu}\right) & \cdots & (-1)^{\mu+2}\frac{4}{\mu} \\ & & & \ddots & \vdots \\ & & & & (-1)^{2\mu-2}\frac{4}{\mu} \\ & & & & 0 \end{bmatrix}\ ,$$

$$\mathrm{J}_k = \begin{bmatrix} & & & 1 \\ & & \iddots & \\ & 1 & & \\ 1 & & & \end{bmatrix}\ .$$

4.5. *Defining the refined almost-C^1 spline geometry $\widehat{\boldsymbol{x}}$ and $\widehat{\mathscr{B}}$*

Finally, starting from $\widehat{\boldsymbol{x}}^*$ and the extraordinary-vertex normals $\boldsymbol{n}_\gamma$, $\gamma \in \widehat{\mathscr{T}}_0^E = \mathscr{T}_0^E$, we define both $\widehat{\mathscr{B}}$ and $\widehat{\boldsymbol{x}}$. This is done exactly as outlined in Section 3.5.

4.6. *Properties of the refinement scheme*

The following results outline the properties of the refinement scheme explained in the previous sections. These properties were previously visually depicted in Fig. 8.

Lemma 4.1. *The control points in the set* $\left\{\widehat{\boldsymbol{x}}^*_{r,i} \ : \ r = 1, \dots, \mu\right\}$ *defined by Eqs. (22) and (25) are coplanar, with the common plane defined by the normal vector* $\boldsymbol{n}_\gamma$.

Proof. By definition of $\boldsymbol{x}$ and $\hat{\boldsymbol{c}}[\boldsymbol{x};\sigma]$, $\sigma \in \mathscr{T}_2$, the face-local control points

$$\left\{\hat{\boldsymbol{c}}_{jk}[\boldsymbol{x};\sigma_r] \ : \ (j,k) \in \{(0,0),(1,0),(0,1),(1,1)\}\ , \quad r = 1,\dots,\mu\right\}\ ,$$

are coplanar. More precisely, the plane in which they lie is orthogonal to the normal vector $\boldsymbol{n}_\gamma$ specified for the extraordinary vertex in Fig. 10.

Then, first considering the interior vertex case, and observing that each row of $\mathring{\mathrm{S}}_\mu + \mathring{\mathrm{Q}}_\mu$ sums to 1, we have

$$\begin{bmatrix} \widehat{\boldsymbol{x}}^*_{1,i} - \widehat{\boldsymbol{c}}^1_{00} \\ \widehat{\boldsymbol{x}}^*_{2,i} - \widehat{\boldsymbol{c}}^1_{00} \\ \vdots \\ \widehat{\boldsymbol{x}}^*_{\mu,i} - \widehat{\boldsymbol{c}}^1_{00} \end{bmatrix} = \mathring{\mathrm{S}}_\mu \begin{bmatrix} \widehat{\boldsymbol{c}}^1_{01} - \widehat{\boldsymbol{c}}^1_{00} \\ \widehat{\boldsymbol{c}}^2_{01} - \widehat{\boldsymbol{c}}^1_{00} \\ \vdots \\ \widehat{\boldsymbol{c}}^\mu_{01} - \widehat{\boldsymbol{c}}^1_{00} \end{bmatrix} + \mathring{\mathrm{Q}}_\mu \begin{bmatrix} \widehat{\boldsymbol{c}}^1_{11} - \widehat{\boldsymbol{c}}^1_{00} \\ \widehat{\boldsymbol{c}}^2_{11} - \widehat{\boldsymbol{c}}^1_{00} \\ \vdots \\ \widehat{\boldsymbol{c}}^\mu_{11} - \widehat{\boldsymbol{c}}^1_{00} \end{bmatrix}\ .$$

Thus the vectors on the left hand side and the vectors on the right hand side are coplanar and the claim follows. The boundary vertex case can be similarly shown since each row of $\mathrm{S}^\partial_\mu + \mathrm{Q}^\partial_\mu$ also sums to 1. ■

Corollary 4.2. *The geometry* $\widehat{\boldsymbol{x}}$ *is identical to* $\widehat{\boldsymbol{x}}^*$, *i.e.*, $\widehat{\boldsymbol{x}} = \widehat{\boldsymbol{x}}^* \in (\widehat{\mathscr{B}} \cap \widehat{\mathscr{B}}^*)^d$.

Proof. The claim follows from Corollary 3.7 and Lemma 4.1. ■

Proposition 4.3. *Let* $\widehat{\boldsymbol{x}}$ *be obtained from* $\boldsymbol{x}$ *via the refinement process outlined in Sections 4.3–4.5. Then the following hold true.*

(a) *Translation and rotation invariance: Let* A *be the matrix mapping the coarse control points* $\boldsymbol{x}_\phi$ *to the refined control points* $\widehat{\boldsymbol{x}}_\phi$. *If* T *and* R *denote a translation and a rotation, then*

$$\mathrm{A} \circ \mathrm{T} = \mathrm{T} \circ \mathrm{A}\ , \quad \mathrm{A} \circ \mathrm{R} = \mathrm{R} \circ \mathrm{A}\ .$$

In particular, the rows of A *sum to* 1.

(b) *Boundary preservation: Let* τ *be a boundary edge of* $\mathscr{T}$ *and* $\boldsymbol{x}_\tau$ *the local representation of* $\boldsymbol{x}$ *restricted to* τ. *With reference to Fig. 9, let the origin of local coordinates* $[0,1]$ *on* τ *be at the left end. Then,*

$$\widehat{\boldsymbol{x}}|_{\widehat{\tau}_L} = \boldsymbol{x}_\tau|_{[0,0.5]}\ , \quad \widehat{\boldsymbol{x}}|_{\widehat{\tau}_R} = \boldsymbol{x}_\tau|_{[0.5,1]}\ .$$

(c) *Structured quadrilateral preservation: Let* σ *be a quadrilateral of* $\mathscr{T}$ *that contains no extraordinary vertices and no corner vertices of valence >1, and* $\boldsymbol{x}_\sigma$ *the local polynomial representation of* $\boldsymbol{x}$ *restricted to* σ. *With reference to Fig. 9, let the origin of local coordinates* $[0,1]^2$ *on* σ *be at the bottom-left vertex* γ. *Then,*

$$\widehat{\boldsymbol{x}}|_{\widehat{\sigma}_i} = \boldsymbol{x}_\sigma|_{[0,0.5]^2}\ , \quad \widehat{\boldsymbol{x}}|_{\widehat{\sigma}_j} = \boldsymbol{x}_\sigma|_{[0.5,1]\times[0,0.5]}\ , \quad \widehat{\boldsymbol{x}}|_{\widehat{\sigma}_k} = \boldsymbol{x}_\sigma|_{[0.5,1]^2}\ , \quad \widehat{\boldsymbol{x}}|_{\widehat{\sigma}_l} = \boldsymbol{x}_\sigma|_{[0,0.5]\times[0.5,1]}\ .$$

(d) *Midpoint interpolation on interior spoke edges: Consider the settings shown in Fig. 10(a) and (b) with the extraordinary vertex* γ. *Let* τ *be an interior spoke edge in the figures such that* γ *corresponds to its first endpoint, and let it be refined into edges* $\widehat{\tau}_a$ *and* $\widehat{\tau}_b$ *such that* γ *now corresponds to the first endpoint of* $\widehat{\tau}_a$. *Denote as* $\boldsymbol{x}_\tau$ *the local representation of* $\boldsymbol{x}$ *restricted to* τ. *Then,*

$$\widehat{\boldsymbol{x}}|_{\widehat{\tau}_a}(1) = \boldsymbol{x}_\tau(0.5)\ .$$

(e) *Normal vector preservation: The refined geometry* $\widehat{\boldsymbol{x}}$ *has well-defined normal vectors at all vertices, and these are identical to the normal vectors for the coarse geometry* $\boldsymbol{x}$.

(f) *Dimension of the refined spline space: We have* $\widehat{\mathscr{I}} = \mathscr{I}^* \cup (\widehat{\mathscr{T}}_0^E \times \{1,2,3\})$ *and*

$$\widehat{\mathfrak{n}} := |\widehat{\mathscr{I}}| = |\widehat{\mathscr{T}}_2| + |\widehat{\mathscr{T}}_1^B| + |\widehat{\mathscr{T}}_0^C| + 3|\widehat{\mathscr{T}}_0^E|\ . \tag{27}$$

(g) *Convergent refinement scheme: The limit surface exists and has a well-defined tangent plane at all points.*

Proof. It is easy to see that (a), (e) and (f) hold by construction. Following the same line of reasoning adopted in [1, Proposition 4.2], it can be seen that properties (b)–(d) hold for the geometry $\widehat{\boldsymbol{x}}^*$. Then, from Corollary 4.2, properties (b)–(d) also hold for $\widehat{\boldsymbol{x}}$. Finally, from [1, Appendix], the limit surface exists when the refinement scheme is applied to geometries in $(\mathscr{B}^*)^d$. Thus, from Corollary 4.2, the same is true for geometries in $\mathscr{B}^d$. While this limit surface may not have a well-defined tangent plane at extraordinary points for all geometries in $(\mathscr{B}^*)^d$, it will be smooth there for geometries in $\mathscr{B}^d$ by virtue of property (e). This implies property (g) and completes the proof. ■

5. Numerical tests

In this section we test the approximation properties and conditioning of almost-C^1 splines for several second and fourth order PDE model problems of practical relevance. We start out by summarizing the implementation-related considerations in Section 5.1. Convergence tests and condition number growths for the Poisson and Biharmonic problems are presented in Section 5.2, the Scordelis–Lo thin shell benchmark in Section 5.3, the surface Cahn–Hilliard problem in Section 5.4 and the surface Laplace–Beltrami eigenvalue problem in Section 5.5.

5.1. A summary of the implementation

In the following we present the basic structure of the implementation. First we describe how an almost-C^1 spline surface can be constructed from (the dual of) a quadrilateral mesh. Then we explain the ingredients needed to set up a finite element implementation.

Construction of almost-C^1 splines surface

Given a quadrilateral mesh (e.g., as in Fig. 2(a)) where each face, boundary edge and corner vertex has a control point associated to it (e.g., as visualized in Fig. 3), the following steps help build an almost-C^1 spline geometry (control points, patch-local coefficients and patch parameterizations).

1. *Initialize spline coefficients:*
   Associate to each non-boundary edge the average of the two adjacent face control points, then associate to each non-corner vertex the average of the adjacent edge control points. This yields face local coefficients as in Fig. 4(b).
2. *Create initial spline surface:*
   For each face, the 3×3 control points associated to the vertices, edges and face can be interpreted as the Bernstein–Bézier coefficients of a biquadratic Bézier patch. The resulting spline, given by the collection of all Bézier patches is C^1 across all edges that do not contain an extraordinary vertex.
3. *Refine extraordinary faces:*
   For each face that contains an extraordinary node perform one refinement step by bisection in both directions, as in (18), cf. Fig. 9.
4. *Perform local averaging around extraordinary vertices:*
   For each extraordinary vertex, collect all (refined) control points within its one ring, i.e., the nearest 2×2 control points on each face adjacent to the vertex (those highlighted in purple in Fig. 6(c)). Find the best approximating plane through these control points and project all of them onto the plane.
5. *Create final spline surface:*
   For each extraordinary face, the new (refined and projected) 4×4 control points can be interpreted as the spline coefficients of a biquadratic B-spline patch with one inner knot in each direction. The resulting almost-C^1 spline, given by the collection of all Bézier patches for regular faces and all B-spline patches for extraordinary faces is C^1 across all edges that do not contain an extraordinary vertex and G^1 at all vertices.

Finite element implementation

On the almost-C^1 spline geometry built as above, one can now set up all ingredients needed for a finite element implementation using almost-C^1 splines, i.e., the ingredients needed for assembling mass and stiffness matrices by looping over all faces (or, alternatively, over all basis functions).

1. *Build dof structure:*
   The almost-C^1 spline dofs are implied by the given mesh structure, i.e., one per face, one per boundary edge, one per corner vertex and three per extraordinary vertex, cf. Fig. 2(b).

2. *Initialize face-local tensor-product basis:*
   Loop over all faces and initialize the corresponding local tensor-product basis at the face-local quadrature points. Any spline function supported on a mesh face can be represented as a linear combination of the corresponding face-local basis. There are two relevant cases to consider.
   - Regular faces: Here the local basis is formed by tensor-products of $b^0_{jk,\square}$ and standard Gauss quadrature points can be distributed on the face.
   - Extraordinary faces: Here the local basis is formed by tensor-products of $b^1_{jk,\square}$ and the local quadrature rule needs to account for the face-internal split. (For instance, standard Gauss rules can be used on each of the 2 × 2 sub-faces.)
3. *Loop over dofs and store face-local extraction operators:*
   The basis function associated to a given dof is supported on only a few mesh faces. On these faces, each basis function is a linear combination of the face-local tensor-product basis, and so we only need to store the corresponding extraction coefficients. The extraction coefficients are presented in Sections 3.3–3.5; 3 × 3 extraction coefficients for regular faces and 4 × 4 for extraordinary faces. The following are the relevant cases.
   - Corner dofs and regular face dofs: The corresponding splines are simply C^1 biquadratic tensor-product B-splines. They can be supported on both regular and extraordinary faces. The local representations on extraordinary faces are subdivided; truncation is unnecessary as it leaves the face-local representation invariant.
   - Boundary edge dofs and extraordinary face dofs: The corresponding splines can be supported on both regular and extraordinary faces; the local representations on extraordinary faces are subdivided and truncated.
   - Extraordinary vertex dofs: The corresponding splines are supported only on extraordinary faces; the local representations on extraordinary faces are obtained from Eq. (14).

5.2. Poisson and Biharmonic problems

We start by investigating the convergence behaviour of almost-C^1 splines under mesh refinements. For this purpose, we solve H^1 and H^2 projection problems on meshes containing a single extraordinary vertex. The exact solution of the problem for all Ω is chosen to be

$$f_{\text{exact}}(x,y) = \sin\left(\pi x + \frac{\pi}{3}\right)\sin\left(\pi y + \frac{\pi}{5}\right)\ . \tag{28}$$

The following trial and test function spaces were used for the problem,

$$\begin{aligned}\mathcal{S}_0 &:= \left\{f \in \mathcal{B} : f = f_0 \text{ on } \Gamma\right\}\ , \qquad & \mathcal{V}_0 &:= \left\{w \in \mathcal{B} : w = 0 \text{ on } \Gamma\right\}\ ,\\ \mathcal{S}_1 &:= \left\{f \in \mathcal{S}_0 : f_{,\boldsymbol{n}} = f_{1,\boldsymbol{n}} \text{ on } \Gamma\right\}\ , \qquad & \mathcal{V}_1 &:= \left\{w \in \mathcal{V}_0 : w_{,\boldsymbol{n}} = 0 \text{ on } \Gamma\right\}\ ,\end{aligned}$$

where $\Gamma = \partial\Omega$ is the boundary of Ω, $\boldsymbol{n}$ is the unit normal to Γ, $a_{,\boldsymbol{n}} := \nabla a \cdot \boldsymbol{n}$, and $f_0 \in \mathcal{B}$ and $f_1 \in \mathcal{S}_0$ are defined such that,

$$\begin{aligned}\int_\Gamma w f_0\, d\Gamma &= \int_\Gamma w f_{\text{exact}}\, d\Gamma\ , \qquad && \forall w \in \mathcal{B},\ w \neq 0 \text{ on } \Gamma\ ,\\ \int_\Gamma w_{,\boldsymbol{n}} f_{1,\boldsymbol{n}}\, d\Gamma &= \int_\Gamma w_{,\boldsymbol{n}} f_{\text{exact},\boldsymbol{n}}\, d\Gamma\ , \qquad && \forall w \in \mathcal{B},\ w_{,\boldsymbol{n}} \neq 0 \text{ on } \Gamma\ .\end{aligned}$$

The weak form of the problems using the above trial and test spaces is

$$\mathrm{P}_1 : \text{Find } f \in \mathcal{S}_0 : \qquad \int_\Omega \nabla w \cdot \nabla f\, d\Omega = -\int_\Omega w \Delta f_{\text{exact}}\, d\Omega\ , \qquad \forall w \in \mathcal{V}_0\ , \tag{29}$$

$$\mathrm{P}_2 : \text{Find } f \in \mathcal{S}_1 : \qquad \int_\Omega \Delta w \Delta f\, d\Omega = \int_\Omega w \Delta^2 f_{\text{exact}}\, d\Omega\ , \qquad \forall w \in \mathcal{V}_1\ . \tag{30}$$

For different valent extraordinary points, the spline geometries Ω at the coarsest refinement level are shown on the left in Fig. 11. The error convergence with mesh refinement is shown in the plots on the right, with the error norms

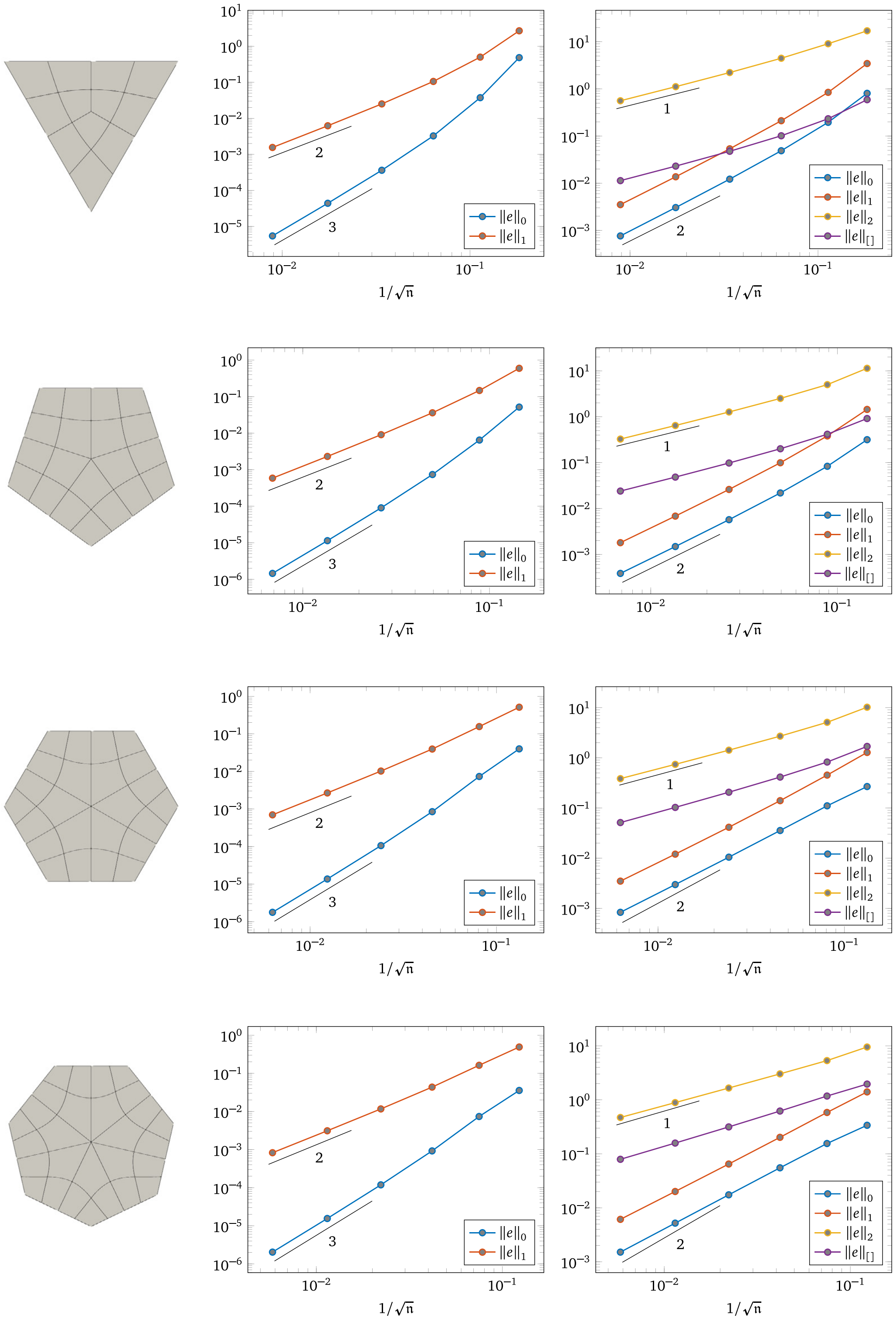


Fig. 11. The above plots show the error convergence with mesh refinement when solving the Poisson (middle) and Biharmonic (right) problems in Eqs. (29) and (30), respectively, on the different spline geometries (left).

Table 1
The above table shows the condition numbers (κ) of the system matrices corresponding to problems P_1 and P_2 from Eqs. (29) and (30), respectively. The condition number at the ith refinement level is denoted as κ_i. The increase in condition numbers is as expected from the nature of the two problems — proportional to $\mathfrak{n}$ and $\mathfrak{n}^2$ for P_1 and P_2, respectively, with $\sqrt{\mathfrak{n}}$ as a stand-in for the mesh size. Note that, from Proposition 3.8 and Fig. 11, the number of dofs at the ith refinement level is $\mathfrak{n}_i = \mu(2^{i+1}+1)^2+3$.

Problem	μ	κ_0	κ_1	κ_2	κ_3	κ_4	κ_5	
P_1	3	5.10e+0	1.56e+1	5.51e+1	2.16e+2	8.78e+2	3.55e+3	$\propto \mathfrak{n}$
	5	1.19e+1	2.30e+1	7.99e+1	3.04e+2	1.20e+3	4.85e+3	
	6	1.20e+1	2.25e+1	8.77e+1	3.50e+2	1.40e+3	5.69e+3	
	7	1.35e+1	2.89e+1	1.12e+2	4.51e+2	1.82e+3	7.44e+3	
P_2	3	2.94e+1	6.23e+2	9.64e+3	1.55e+5	2.51e+6	4.09e+7	$\propto \mathfrak{n}^2$
	5	3.63e+1	5.64e+2	8.46e+3	1.32e+5	2.10e+6	3.35e+7	
	6	3.39e+1	5.44e+2	8.27e+3	1.29e+5	2.07e+6	3.37e+7	
	7	4.32e+1	7.30e+2	1.13e+4	1.80e+5	2.95e+6	4.80e+7	

plotted against the inverse of the square root of the number of dofs. With the error $e := f - f_{\text{exact}}$, we plot the following four error norms to investigate the approximation behaviour of the almost-C^1 spline space,

$$\begin{aligned} \|e\|_0^2 &:= \int_\Omega e^2\, d\Omega\,, \qquad \|e\|_1^2 := \int_\Omega |\nabla e|^2\, d\Omega\,, \\ \|e\|_2^2 &:= \sum_{i=1}^{|\mathscr{T}_2|} \int_{\Omega_i} (\Delta e)^2\, d\Omega\,, \qquad \|e\|_{[]}^2 := \mathfrak{n}^{\frac{1}{2}} \sum_{i=1}^{|\mathscr{T}_2|} \int_{\partial\Omega_i\setminus\Gamma} \left[\frac{de}{dn}\right]^2 d\Gamma\,, \end{aligned} \tag{31}$$

where Ω_i is the geometric image of the ith mesh face, and $\left[\frac{de}{dn}\right]$ is the jump in the normal derivative of e across the interior mesh edges. For a biquadratic spline space demonstrating optimal approximation behaviour, one would expect $\|e\|_0$ and $\|e\|_1$ to converge with orders 3 and 2 for problem P_1, and $\|e\|_0$, $\|e\|_1$, $\|e\|_2$ and $\|e\|_{[]}$ to converge with orders 2, 2, 1 and 1 for problem P_2, respectively. As is clear from Fig. 11, the spline spaces demonstrate optimal convergence rates for all error norms. (Note that $\|e\|_{[]}$ is relevant only for the problem P_2 and therefore only plotted in the rightmost column of plots in Fig. 11.) Only a slight deterioration is visible in the L^2 norm (≈ 1.8) for the Biharmonic problem at high refinement levels; this is similar to the results for bicubic splines in [7]. Finally, for both problems we show the condition numbers of the system matrices in Table 1, as can be seen, the conditioning is as expected from a well-conditioned basis without singularities.

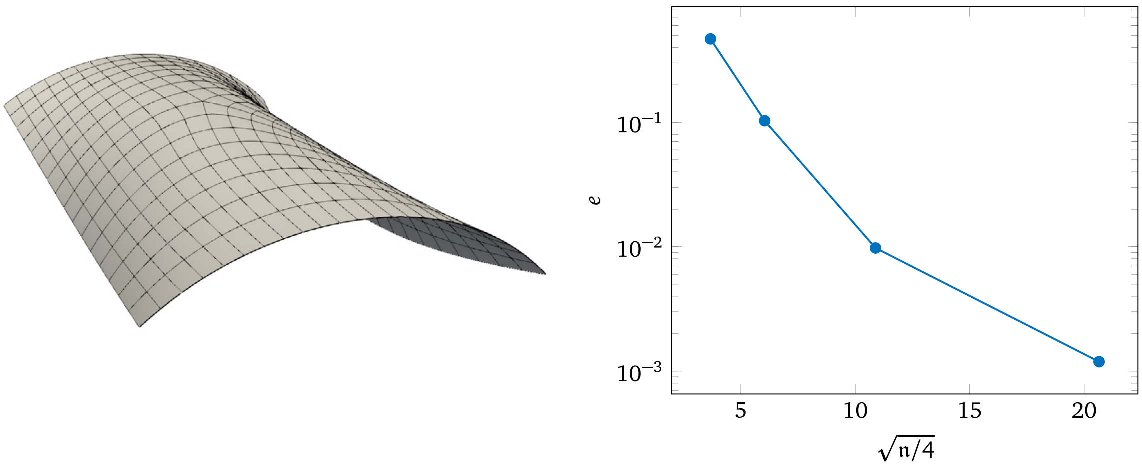
Remark 5.1. Imposition of boundary conditions for values of unstructured splines is very simple here, given the boundary Kronecker–Delta property from Proposition 3.8. Boundary conditions for normal derivatives are also not problematic as they only involve the basis functions for the boundary dofs and the layer of dofs adjacent to boundary dofs (i.e., the adjacent face dofs and dofs for boundary extraordinary vertices).

5.3. Kirchhoff–Love shells: The Scordelis–Lo benchmark problem

We now solve a Kirchhoff–Love benchmark problem — the Scordelis–Lo test case. A curved cylinder with dimensions $(r,\ L,\ \theta_{\text{sector}}) = (25,\ 50,\ 2\pi/3)$ is loaded under gravity and has the following material parameters:

$$\begin{aligned} \text{Young's modulus,} \quad & E = 4.32 \times 10^8\,, \\ \text{Poisson's ratio,} \quad & \nu = 0.0\,, \\ \text{Thickness,} \quad & t = 0.25\,. \end{aligned}$$

The shell formulation used is based on the Kirchhoff–Love thin shell theory in which transverse shear strains are zero. The end result is a rotation-free formulation requiring C^1 continuous trial functions. Necessarily, since transverse shear strains are suppressed, one would anticipate the theory would result in smaller deformations than for Reissner–Mindlin shell theory, which accounts for transverse shear strains. As observed in [71] and elsewhere,



(a) deformed geometry, $\mathfrak{n} = 474$

(b) error convergence for midpoint displacement

Fig. 12. For the Scordelis–Lo benchmark problem (Section 5.3), Figure (a) shows the solution at one of the refinement levels, while Figure (b) shows the convergence of the error $e := |1 - \frac{w}{w_{\text{ref}}}|$ under mesh refinement, where w is the vertical displacement computed at the midpoint of one of the non-curved edges of the cylindrical roof and w_{ref} is a reference solution based on Kirchhoff–Love theory [71]. Our results are based on Kirchhoff–Love thin shell theory in which transverse shear deformations are suppressed, and as one would expect, the Kirchhoff–Love converged displacement is somewhat less than that for Reissner–Mindlin theory.

the Kirchhoff–Love theory leads to a converged maximum downward vertical displacement $w_{\text{ref}} = 0.3006$, while the Reissner–Mindlin theory yields 0.3024; in the following we use the former as the reference solution.

We start from a rectangular planar geometry defined using a mesh $\mathscr{T}$ with two interior extraordinary points (valences 3 and 5); the planar control points are chosen such that the planar geometry coincides with the projection of the cylindrical roof on the xy-plane. To build the cylindrical roof, we first refine the planar geometry a desired number of times and then perform an L^2-projection to find the height of the control points. Note that since we are trying to match a given target geometry here, it is reasonable to perform this fitting for each new refinement.

Solving the Kirchhoff–Love problem on the geometry thus obtained, the solution is as shown in Fig. 12(a); the deformations have been scaled up by a factor of 15 for the purpose of visualization. The vertical displacement at the mid-point of the free edges converges toward the reference Kirchhoff–Love reference solution of w_{ref}; the normalized error in the computed solution is shown in Fig. 12(b). Note that since we did not exploit any symmetry conditions in the simulation, we display the error against the square root of a quarter of the number of degrees of freedom $\mathfrak{n}$ for the spline space defined on $\mathscr{T}$. This brings the results in line with those of [71] which assumed four-fold symmetry and plotted the error versus the number of control points per edge of their rectangular mesh. The performance of our construction is indistinguishable from the results of [71] which used standard tensor-product B-splines.

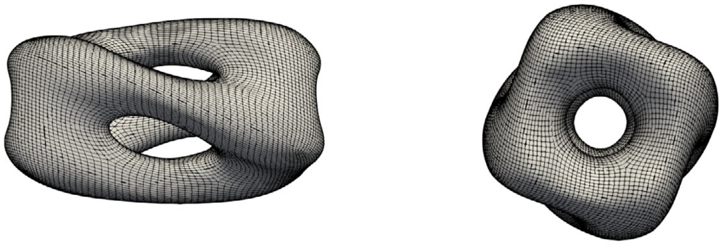
5.4. Surface Cahn–Hilliard problem

We now solve the fourth-order non-linear Cahn–Hilliard problem on the topologically complex surface Ω shown in Fig. 13. The non-dimensional strong form of the problem is as below (see [72] for the associated weak form):

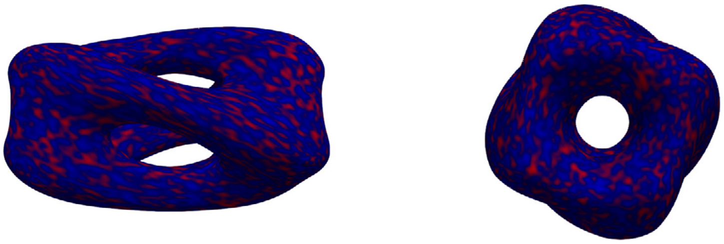
$$\frac{\partial c}{\partial t} = \nabla_\Omega \cdot (c(1-c)\nabla_\Omega(\mathbb{N}_2\mu_c - \Delta_\Omega c)) \quad \text{on } \Omega \times [0,\ T]\ ,$$

$$c(\boldsymbol{x}, 0) = c_0(\boldsymbol{x}) \quad \text{on } \Omega\ ,$$

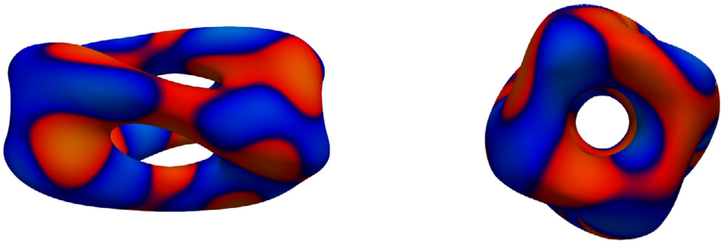
where ∇_Ω and Δ_Ω are the surface gradient and Laplace–Beltrami operators, respectively, and $\mu_c := \frac{1}{3}\log\left(\frac{c}{1-c}\right) + 1 - 2c$. The unstructured mesh had 18,432 quadrilateral faces and 18,552 degrees of freedom, and we solved the equations for initial volume fraction $\bar{c} = 0.5$ and the corresponding value of $\mathbb{N}_2$ was 41.7313. The initial value of c, namely c_0, was determined by randomly perturbing $\bar{c}$, as described in [73,74]. The results are shown in



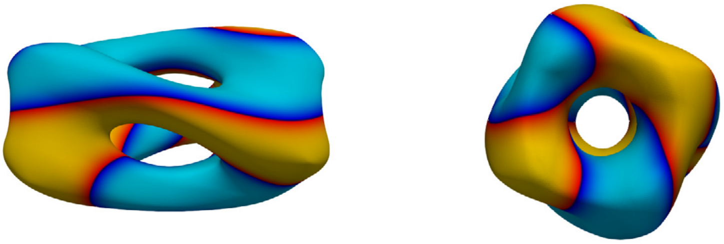
spline geometry, $\mathfrak{n} = 18,552$



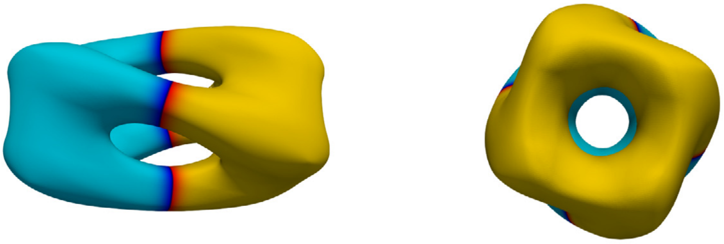
$t = 0$



$t = 0.089765$



$t = 0.636560$



$t = 1128.56$

Fig. 13. These figures show the initial volume fraction distribution ($t = 0$) over a surface of non-trivial topology (top row), which is the domain of interest for the surface Cahn–Hilliard problem (Section 5.4), and the rows below show its time-evolution. The meshes used for the computation contained 18,552 degrees of freedom. (The left and right columns show different views of the geometry and solution.)

Fig. 13. Steady state was reached for the configuration in 600 time-steps with the aid of an adaptive time-stepping scheme [74]. At all times, the solution coefficients were strictly between 0 and 1. Then, since the spline basis functions are non-negative and form a partition of a unity, it easily follows that the computed solution is pointwise between 0 and 1 at all times.

5.5. Laplace–Beltrami eigenvalue problem

For the final set of tests, we showcase the application of our splines for reconstructing a complex CAD geometry and solving an eigenvalue problem on it. The geometry, a portion of a BMW car, is freely available as a Blender® model. This model was imported into Rhinoceros® where a quadrilateral mesh consisting of 4482 faces was created; this mesh contained both boundary and interior extraordinary points. Using our splines built on this mesh (5330 dofs), we reconstructed the geometry and solved a Laplace–Beltrami eigenvalue problem [75] on the spline geometry. This problem is defined as: find $f \in \mathscr{S}$ and $\lambda \in \mathbb{R}$ such that

$$\int_{\Omega} \nabla_{\Omega} g \cdot \nabla_{\Omega} f \, d\Omega = \int_{\Omega} g f \, d\Omega \; , \quad \forall g \in \mathscr{S} \; , \tag{32}$$

where $\mathscr{S} := \{f \in \mathscr{B} \; : \; f|_{\Gamma_b} = 0\}$ and Γ_b is the union of the two bottom edges of the car hull. The spline geometry and select eigenmodes for the problem are shown in Fig. 14.

6. Conclusions

We have presented the construction and refinement of almost-C^1 splines, that is, analysis-suitable biquadratic spline spaces on fully unstructured quadrilateral meshes, for building smooth spline surfaces of arbitrary topology as well as for solving fourth-order problems on them. Several numerical examples of challenging fourth-order problems have been presented to exemplify this. The corresponding almost-C^1 spline basis functions are well-conditioned and have several B-spline-like properties such as partition of unity, non-negativity, local support and linear independence. Furthermore, we have described the construction explicitly and in a self-contained manner using Bézier extractions for enabling immediate implementation.

We use approximate smoothness in an explicit manner for our construction and strongly believe that this is a powerful approach for arriving at spline constructions that can circumvent many of the obstacles faced by strongly smooth unstructured splines, while at the same time retaining many of the latter's advantages. In this first paper we focus on the construction itself and while the numerical results are highly encouraging, a theoretical analysis of convergence is an interesting topic for future research. It is our opinion that such a theoretical analysis may be simpler for almost-C^1 splines than for constructions that employ singularities in the definitions of the spline basis functions (although they are strongly smooth, nested and demonstrate optimal convergence in numerical tests; the singularities usually pose significant difficulties for any standard approximation proofs). A possible way to prove convergence for almost-C^1 splines may be to follow strategies for non-conforming finite elements, such as in [6].

Furthermore, within this new framework, there are many extensions possible which will be the focus of our future work. Some of these are the formulation of similar constructions for higher polynomial degrees and higher orders of approximate smoothness, as well as incorporation of local refinement. For the latter, we note that since our construction is highly local, it can be readily embedded within locally refined spline constructions (e.g., similarly to [39]) by assuming sufficient separation between hanging nodes and extraordinary vertices — for biquadratic splines, a 2-ring distance between extraordinary vertices and hanging nodes is expected to be sufficient. As an alternative approach for local refinement, a generalization similar to hierarchical B-splines may also be possible, even though the spaces on different refinement levels are not nested. Finally, other minor extensions include generalizing the set of corner vertices to include boundary vertices of valence higher than 1, as well as reducing the smoothness across select interior edges to create geometries with creased features.

(a) spline geometry

(b) mode 1

(c) mode 11

(d) mode 12

(e) mode 15

Fig. 14. The figure in the top row shows a part of a BMW car — the model is a Blender® demofile, a coarse quadrilateral mesh for it was generated using Rhinoceros® and the spline geometry was constructed on that mesh using our unstructured splines. The bottom rows show select eigenmodes corresponding to the Laplace–Beltrami eigenvalue problem (Section 5.5) solved on this geometry. The mesh contains 4482 faces and 5330 dofs.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgements

The research of Deepesh Toshniwal is supported by project number 212.150 awarded through the Veni research programme by the Dutch Research Council (NWO), Netherlands. 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".

Appendix A. Alternative construction 1: Truncation using degree elevation

A construction similar to the one explained in Section 3.5 can be achieved if the functions $B^*_\phi \in \mathscr{B}$ are degree elevated on every extraordinary face, i.e., $B^*_\phi|_\sigma$ is represented as a bicubic polynomial for each $\sigma \in \mathscr{T}_2^E$. Then $B_\phi|_\sigma$ is given as the truncated, degree elevated version of $B^*_\phi|_\sigma$. Algorithmically, the only change is in the matrix K, which has to be replaced by the degree elevation matrix

$$K_{d.e.} = \begin{bmatrix} 1 & & \\ \frac{1}{3} & \frac{2}{3} & \\ & \frac{2}{3} & \frac{1}{3} \\ & & 1 \end{bmatrix} , \tag{A.1}$$

and in the local basis $b^1_{jk,\square}$ in (10), which is replaced by bicubic Bernstein polynomials. The truncation step as well as the basis construction for the extraordinary vertex splines remain the same.

Appendix B. Alternative construction 2: Geometry-independent templates

In the following we discuss how the basis functions $B_{(\gamma,\nu)}$, $\nu \in \{1, 2, 3\}$, can be defined independently of the geometry. This is achieved by replacing the projection onto a prescribed tangent plane, as in (13), by a template configuration depending only on the valence. We assume that the control triangle is always given as

$$(\boldsymbol{a}_1, \boldsymbol{a}_2, \boldsymbol{a}_3) = \left((0,1)^T, \left(-\frac{\sqrt{3}}{2}, -\frac{1}{2}\right)^T, \left(\frac{\sqrt{3}}{2}, -\frac{1}{2}\right)^T \right) .$$

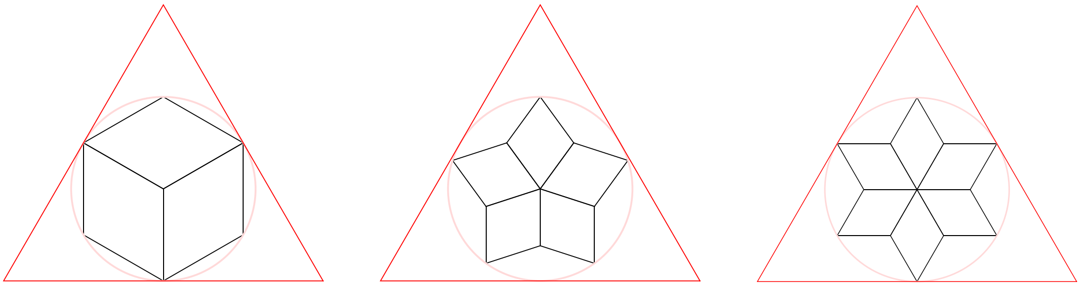
Then we define points $\boldsymbol{c}^i_{jk}$, with $(j,k) \in \{0,1\}^2$, $i \in \{1,\ldots,\mu\}$ to be

$$\begin{aligned} \boldsymbol{c}^i_{00} &= (0,0)^T \\ \boldsymbol{c}^i_{11} &= \frac{1}{2}(-\sin(2\pi(i-1)/\mu), \cos(2\pi(i-1)/\mu))^T \\ \boldsymbol{c}^i_{10} = \boldsymbol{c}^{i+1}_{01} &= \frac{1}{4\cos(\pi/\mu)}(-\sin(2\pi(i-1/2)/\mu), \cos(2\pi(i-1/2)/\mu))^T , \end{aligned}$$

see Fig. B.15.

Consequently, the coefficients of the basis functions $B_{(\gamma,\nu)}$ are again given as the barycentric coordinates with respect to the vertices of the triangle, i.e.,

$$B_{(\gamma,\nu)}|_{\sigma_i} = \sum_{jk=0}^{3} \hat{c}_{jk}[B_{(\gamma,\nu)}; \sigma_j] b^1_{jk,\square} ,$$



(a) valence 3. (b) valence 5. (c) valence 6.

Fig. B.15. Templates for valencies 3, 5 and 6.

with

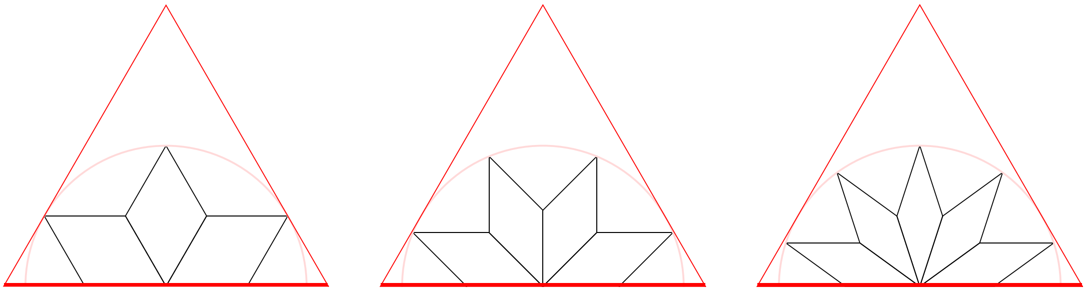
$$\hat{c}_{jk}[B_{(\gamma,\nu)};\sigma_i] = \begin{cases} \lambda_\nu(\boldsymbol{c}^i_{jk})\ , & (j,k) \in \{0,1\}^2\ , \\ 0\ , & \text{otherwise}\ . \end{cases}$$

Independent of the valence, we always have $\hat{c}_{00}[B_{(\gamma,\nu)};\sigma_i] = \frac{1}{3}$. For $\mu = 3$ we have

$$\begin{aligned} \hat{c}_{11}[B_{(\gamma,1)};\sigma_1] &= \frac{2}{3} & \hat{c}_{10}[B_{(\gamma,1)};\sigma_1] &= \frac{1}{2} \\ \hat{c}_{11}[B_{(\gamma,1)};\sigma_2] &= \frac{1}{6} & \hat{c}_{10}[B_{(\gamma,1)};\sigma_2] &= 0 \\ \hat{c}_{11}[B_{(\gamma,1)};\sigma_3] &= \frac{1}{6} & \hat{c}_{10}[B_{(\gamma,1)};\sigma_3] &= \frac{1}{2}\ , \end{aligned}$$

for $\mu = 5$ we have

$$\begin{aligned} \hat{c}_{11}[B_{(\gamma,1)};\sigma_1] &= \frac{2}{3} & \hat{c}_{11}[B_{(\gamma,2)};\sigma_1] &= \frac{1}{6} \approx 0.166667 \\ \hat{c}_{10}[B_{(\gamma,1)};\sigma_1] &= \frac{1}{2} & \hat{c}_{10}[B_{(\gamma,2)};\sigma_1] &= \frac{3+\sqrt{15-6\sqrt{5}}}{12} \approx 0.354867 \\ \hat{c}_{11}[B_{(\gamma,1)};\sigma_2] &= \frac{3+\sqrt{5}}{12} & \hat{c}_{11}[B_{(\gamma,2)};\sigma_2] &= \frac{9-\sqrt{5}+\sqrt{30+6\sqrt{5}}}{24} \approx 0.556377 \\ \hat{c}_{10}[B_{(\gamma,1)};\sigma_2] &= \frac{1+\sqrt{5}}{12} & \hat{c}_{10}[B_{(\gamma,2)};\sigma_2] &= \frac{11-\sqrt{5}+\sqrt{30-6\sqrt{5}}}{24} \approx 0.534843 \\ \hat{c}_{11}[B_{(\gamma,1)};\sigma_3] &= \frac{3-\sqrt{5}}{12} & \hat{c}_{11}[B_{(\gamma,2)};\sigma_3] &= \frac{9+\sqrt{5}+\sqrt{30-6\sqrt{5}}}{24} \approx 0.637848 \\ \hat{c}_{10}[B_{(\gamma,1)};\sigma_3] &= \frac{3-\sqrt{5}}{6} & \hat{c}_{10}[B_{(\gamma,2)};\sigma_3] &= \frac{3+\sqrt{5}}{12} \approx 0.436339 \\ \hat{c}_{11}[B_{(\gamma,1)};\sigma_4] &= \frac{3-\sqrt{5}}{12} & \hat{c}_{11}[B_{(\gamma,2)};\sigma_4] &= \frac{9+\sqrt{5}-\sqrt{30-6\sqrt{5}}}{24} \approx 0.298491 \\ \hat{c}_{10}[B_{(\gamma,1)};\sigma_4] &= \frac{1+\sqrt{5}}{12} & \hat{c}_{10}[B_{(\gamma,2)};\sigma_4] &= \frac{11-\sqrt{5}-\sqrt{30-6\sqrt{5}}}{24} \approx 0.195485 \\ \hat{c}_{11}[B_{(\gamma,1)};\sigma_5] &= \frac{3+\sqrt{5}}{12} & \hat{c}_{11}[B_{(\gamma,2)};\sigma_5] &= \frac{9-\sqrt{5}-\sqrt{30+6\sqrt{5}}}{24} \approx 0.00728413 \\ \hat{c}_{10}[B_{(\gamma,1)};\sigma_5] &= \frac{1}{2} & \hat{c}_{10}[B_{(\gamma,2)};\sigma_5] &= \frac{3-\sqrt{15-6\sqrt{5}}}{12} \approx 0.145133\ , \end{aligned}$$



(a) valence 3. (b) valence 4. (c) valence 5.

Fig. B.16. Templates for boundary vertices with valencies 3, 4 and 5. The boundary is assumed to be at the bottom edge of the control triangle.

and for $\mu = 6$ we have

$$\begin{aligned}
\hat{c}_{11}[B_{(\gamma,1)};\sigma_1] &= \frac{2}{3} & \hat{c}_{10}[B_{(\gamma,1)};\sigma_1] &= \frac{1}{2}\\
\hat{c}_{11}[B_{(\gamma,1)};\sigma_2] &= \frac{1}{2} & \hat{c}_{10}[B_{(\gamma,1)};\sigma_2] &= \frac{1}{3}\\
\hat{c}_{11}[B_{(\gamma,1)};\sigma_3] &= \frac{1}{6} & \hat{c}_{10}[B_{(\gamma,1)};\sigma_3] &= \frac{1}{6}\\
\hat{c}_{11}[B_{(\gamma,1)};\sigma_4] &= 0 & \hat{c}_{10}[B_{(\gamma,1)};\sigma_4] &= \frac{1}{6}\\
\hat{c}_{11}[B_{(\gamma,1)};\sigma_5] &= \frac{1}{6} & \hat{c}_{10}[B_{(\gamma,1)};\sigma_5] &= \frac{1}{3}\\
\hat{c}_{11}[B_{(\gamma,1)};\sigma_6] &= \frac{1}{2} & \hat{c}_{10}[B_{(\gamma,1)};\sigma_6] &= \frac{1}{2}\,.
\end{aligned}$$

The other coefficients are determined via the symmetry of the configuration. While valencies 3 and 6 allow rotationally symmetric configurations, where all three basis functions look alike, valence 5 yields two different types of functions ($B_{(\gamma,2)}$ and $B_{(\gamma,3)}$ are equivalent up to reflection). Similar templates can be derived for extraordinary vertices at the boundary. We show three such configurations in Fig. B.16 and leave the computation of coefficients as an exercise to the readers.

Appendix C. Alternative construction 3: A smooth subspace without local subdivison

In the following we define an approximate C^1 subspace $\mathscr{B}^\dagger$ of the B-spline space $\mathscr{B}^*$ defined in Section 3.1. For this we need to reduce the number of degrees of freedom. Instead of taking all faces in $\mathscr{T}_2$ as degrees of freedom, we mark faces in such a way, that for each extraordinary vertex exactly three faces in its 1-ring are marked. In addition, all faces that are not contained in the 1-ring of an extraordinary vertex are marked. Such a marking may not always be possible. However, by bisecting the mesh once, all 1-rings of extraordinary vertices are disjoint and a valid marking exists. We collect all marked faces in ${\mathscr{T}_2}^\dagger$.

Thus, we end up with the following degrees of freedom:

- Face dofs: We associate one degree of freedom to each marked face $\sigma \in {\mathscr{T}_2}^\dagger$.
- Boundary edge dofs: We associate one degree of freedom to each boundary edge $\tau \in {\mathscr{T}_1}^B$.
- Corner vertex dofs: We associate one degree of freedom to each corner vertex $\gamma \in \mathscr{T}_0^C$.

Given an extraordinary vertex $\gamma \in \mathscr{T}_0^E$ of valence μ and let $\sigma_1, \sigma_2, \ldots, \sigma_\mu$ be the faces surrounding it. Moreover, let $\{i_1, i_2, i_3\} \subset \{1, 2, \ldots, \mu\}$ denote the marked faces σ_{i_1}, σ_{i_2} and σ_{i_3}. Given a spline geometry $\boldsymbol{x}^*$, let $\boldsymbol{x}_i^*$ be the control points corresponding to the faces σ_i.

Let again P be the orthogonal projection onto the tangent plane given by a prescribed normal $\boldsymbol{n}_\gamma$. This yields new control points $P(\boldsymbol{x}_i^*)$. New basis functions $B^\dagger_{\sigma_{i_\nu}}$ can now be constructed for σ_{i_1}, σ_{i_2} and σ_{i_3}, via

$$B^\dagger_{\sigma_{i_\nu}} = \sum_{i=1}^{\mu} \lambda_\nu(P(\boldsymbol{x}_i^*)) B^*_{\sigma_i} \ ,$$

where $\lambda_\nu(\boldsymbol{c})$ denotes the νth component of the barycentric coordinates of $\boldsymbol{c}$ with respect to the triangle

$$(P(\boldsymbol{x}_{i_1}^*), P(\boldsymbol{x}_{i_2}^*), P(\boldsymbol{x}_{i_3}^*)) \ .$$

All functions corresponding to regular faces, boundary edges and corner vertices remain unchanged, i.e., $B^\dagger_\phi = B^*_\phi$ for all $\phi \in \mathscr{T}_2 \setminus \mathscr{T}_2^E \cup \mathscr{T}_1^B \cup \mathscr{T}_0^C$.

Similar to the construction in Section 3.5 one can describe a basis with respect to a control triangle which is different from the control points of marked faces. Such a construction based on a control triangle is only feasible if no face contains more than one extraordinary vertex. Alternatively, a construction based on a template as in Appendix B can be applied as well.

References

[1] D. Toshniwal, Quadratic splines on quad-tri meshes: Construction and an application to simulations on watertight reconstructions of trimmed surfaces, Comput. Methods Appl. Mech. Engrg. 388 (2022) 114174.

[2] T.J.R. Hughes, J.A. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, Comput. Methods Appl. Mech. Engrg. 194 (2005) 4135–4195.

[3] G.E. Farin, J. Hoschek, M.-S. Kim, Handbook of Computer Aided Geometric Design, Elsevier, 2002.

[4] T.J.R. Hughes, The Finite Element Method: Linear Static and Dynamic Finite Element Analysis, Courier Corporation, 2012.

[5] P.T. Boggs, A. Althsuler, A.R. Larzelere, E.J. Walsh, R.L. Clay, M.F. Hardwick, DART System Analysis, Technical report, Sandia National Laboratories, 2005.

[6] Z.C. Shi, The FEM test for convergence of nonconforming finite elements, Math. Comp. 49 (180) (1987) 391–405.

[7] 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, Comput. Methods Appl. Mech. Engrg. 327 (2017) 411–458.

[8] T. Nguyen, K. Karčiauskas, J. Peters, A comparative study of several classical, discrete differential and isogeometric methods for solving Poisson's equation on the disk, Axioms 3 (2014) 280–299.

[9] Y. Guo, M. Ruess, Nitsche's method for a coupling of isogeometric thin shells and blended shell structures, Comput. Methods Appl. Mech. Engrg. 284 (2015) 881–905.

[10] S.E. Moore, Discontinuous Galerkin isogeometric analysis for the biharmonic equation, Comput. Math. Appl. 76 (4) (2018) 673–685.

[11] A. Benvenuti, G. Sangalli, Isogeometric Analysis for C^1-continuous Mortar Method (Ph.D. thesis), Ph.D. thesis, Corso di Dottorato in Matematica e Statistica, Università degli studi di Pavia, 2017.

[12] T. Horger, A. Reali, B. Wohlmuth, L. Wunderlich, A hybrid isogeometric approach on multi-patches with applications to Kirchhoff plates and eigenvalue problems, Comput. Methods Appl. Mech. Engrg. 348 (2019) 396–408.

[13] D. Miao, Z. Zou, M.A. Scott, M.J. Borden, D.C. Thomas, Isogeometric Bézier dual mortaring: The Kirchhoff–Love shell problem, Comput. Methods Appl. Mech. Engrg. 382 (2021) 113873.

[14] K. Rafetseder, W. Zulehner, A new mixed approach to Kirchhoff–Love shells, Comput. Methods Appl. Mech. Engrg. 346 (2019) 440–455.

[15] P. Weinmüller, T. Takacs, Construction of approximate C^1 bases for isogeometric analysis on two-patch domains, Comput. Methods Appl. Mech. Engrg. 385 (2021) 114017.

[16] P. Weinmüller, T. Takacs, An approximate C^1 multi-patch space for isogeometric analysis with a comparison to Nitsche's method, Comput. Methods Appl. Mech. Engrg. 401 (2022) 115592.

[17] T.J.R. Hughes, G. Sangalli, T. Takacs, D. Toshniwal, Smooth multi-patch discretizations in isogeometric analysis, in: Handbook of Numerical Analysis, Elsevier, 2020.

[18] United States Department of Transportation, National highway traffic safety administration, 2021, https://www.nhtsa.gov/crash-simulation-vehicle-models. [Online; accessed 01-December-2021].

[19] K.M. Shepherd, X.D. Gu, T.J.R. Hughes, Isogeometric model reconstruction of open shells via Ricci flow and quadrilateral layout-inducing energies, Eng. Struct. 252 (2022) 113602.

[20] 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, ACM Press, 1995, pp. 359–368.

[21] M. Majeed, F. Cirak, Isogeometric analysis using manifold-based smooth basis functions, Comput. Methods Appl. Mech. Engrg. 316 (2017) 547–567.

[22] Q. Zhang, T. Takacs, F. Cirak, Manifold-based B-splines on unstructured meshes, in: Conference on Isogeometric Analysis and Applications, Springer, 2018, pp. 243–262.

[23] K.J. Koh, D. Toshniwal, F. Cirak, An optimally convergent smooth blended B-spline construction for unstructured quadrilateral and hexahedral meshes, 2021, arXiv preprint arXiv:2111.04401.
[24] D. Doo, M. Sabin, Behaviour of recursive division surfaces near extraordinary points, Comput. Aided Des. 10 (6) (1978) 356–360.
[25] E. Catmull, J. Clark, Recursively generated B-spline surfaces on arbitrary topological meshes, Comput. Aided Des. 10 (6) (1978) 350–355.
[26] J. Stam, Exact evaluation of Catmull–Clark subdivision surfaces at arbitrary parameter values, in: Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques, ACM Press, 1998, pp. 395–404.
[27] J. Peters, U. Reif, Subdivision Surfaces, Springer-Verlag, 2008.
[28] P.J. Barendrecht, M. Bartoň, J. Kosinka, Efficient quadrature rules for subdivision surfaces in isogeometric analysis, Comput. Methods Appl. Mech. Engrg. 340 (2018) 1–23.
[29] D. Burkhart, B. Hamann, G. Umlauf, Iso-geometric finite element analysis based on Catmull-Clark subdivision solids, 29 (5) (2010) 1575–1584.
[30] P.J. Barendrecht, Isogeometric Analysis for Subdivision Surfaces, Eindhoven University of Technology, Eindhoven, the Netherlands, 2013.
[31] X. Wei, Y. Zhang, T.J.R. Hughes, M.A. Scott, Truncated hierarchical Catmull–Clark subdivision with local refinement, Comput. Methods Appl. Mech. Engrg. 291 (2015) 1–20.
[32] A. Riffnaller-Schiefer, U.H. Augsdörfer, D.W. Fellner, Isogeometric shell analysis with NURBS compatible subdivision surfaces, Appl. Math. Comput. 272 (2016) 139–147.
[33] Q. Zhang, M. Sabin, F. Cirak, Subdivision surfaces with isogeometric analysis adapted refinement weights, Comput. Aided Des. 102 (2018) 104–114.
[34] X. Li, X. Wei, Y.J. Zhang, Hybrid non-uniform recursive subdivision with improved convergence rates, Comput. Methods Appl. Mech. Engrg. 352 (2019) 606–624.
[35] X. Wei, X. Li, Y.J. Zhang, T.J.R. Hughes, Tuned hybrid nonuniform subdivision surfaces with optimal convergence rates, Internat. J. Numer. Methods Engrg. 122 (9) (2021) 2117–2144.
[36] U. Reif, A refineable space of smooth spline surfaces of arbitrary topological genus, J. Approx. Theory 90 (1997) 174–199.
[37] U. Reif, TURBS-topologically unrestricted rational B-splines, Constr. Approx. 14 (1998) 57–77.
[38] T. Nguyen, J. Peters, Refinable C^1 spline elements for irregular quad layout, Comput. Aided Geom. Design 43 (2016) 123–130.
[39] 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, Comput. Methods Appl. Mech. Engrg. 360 (2020) 112765.
[40] X. Wei, X. Li, K. Qian, T.J.R. Hughes, Yongjie Jessica Zhang, Hugo Casquero, Analysis-suitable unstructured T-splines: multiple extraordinary points per face, Comput. Methods Appl. Mech. Engrg. 391 (2022) 114494.
[41] C. Zimmermann, D. Toshniwal, C.M. Landis, T.J.R. Hughes, K.K. Mandadapu, R.A. Sauer, An isogeometric finite element formulation for phase transitions on deforming surfaces, 2019.
[42] H. Casquero, C. Bona-Casas, D. Toshniwal, T.J.R. Hughes, H. Gomez, Y.J. Zhang, The divergence-conforming immersed boundary method: Application to vesicle and capsule dynamics, J. Comput. Phys. 425 (2021) 109872.
[43] T. Takacs, B. Jüttler, H^2 regularity properties of singular parameterizations in isogeometric analysis, Graph. Models 74 (6) (2012) 361–372.
[44] T. Takacs, Construction of smooth isogeometric function spaces on singularly parameterized domains, in: International Conference on Curves and Surfaces, Springer, 2014, pp. 433–451.
[45] D. Toshniwal, H. Speleers, R.R. Hiemstra, T.J.R. Hughes, Multi-degree smooth polar splines: A framework for geometric modeling and isogeometric analysis, Comput. Methods Appl. Mech. Engrg. 316 (2017) 1005–1061.
[46] H. Speleers, D. Toshniwal, A general class of C1 smooth rational splines: Application to construction of exact ellipses and ellipsoids, Comput. Aided Des. 132 (2021) 102982.
[47] D. Toshniwal, T.J.R. Hughes, Isogeometric discrete differential forms: Non-uniform degrees, Bézier extraction, polar splines and flows on surfaces, Comput. Methods Appl. Mech. Engrg. 376 (2021) 113576.
[48] U. Reif, Biquadratic G-spline surfaces, Comput. Aided Geom. Design 12 (2) (1995) 193–205.
[49] 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, Comput. Methods Appl. Mech. Engrg. 254 (2013) 197–221.
[50] B. Mourrain, R. Vidunas, N. Villamizar, Dimension and bases for geometrically continuous splines on surfaces of arbitrary topology, Comput. Aided Geom. Design 45 (2016) 108–133.
[51] M. Kapl, G. Sangalli, T. Takacs, A family of C^1 quadrilateral finite elements, Adv. Comput. Math. 47 (6) (2021) 1–38.
[52] J. Grošelj, M. Kapl, M. Knez, T. Takacs, V. Vitrih, A super-smooth C1 spline space over planar mixed triangle and quadrilateral meshes, Comput. Math. Appl. 80 (12) (2020) 2623–2643.
[53] M. Bercovier, T. Matskewich, Smooth Bézier Surfaces over Unstructured Quadrilateral Meshes, in: Lecture Notes of the Unione Matematica Italiana, Springer International Publishing, 2017.
[54] M. Kapl, V. Vitrih, B. Jüttler, K. Birner, Isogeometric analysis with geometrically continuous functions on two-patch geometries, Comput. Math. Appl. 70 (2015) 1518–1538.
[55] A. Collin, G. Sangalli, T. Takacs, Analysis-suitable G^1 multi-patch parametrizations for C^1 isogeometric spaces, Comput. Aided Geom. Design 47 (2016) 93–113.
[56] M. Kapl, G. Sangalli, T. Takacs, Dimension and basis construction for analysis-suitable G^1 two-patch parameterizations, Comput. Aided Geom. Design 52–53 (2017) 75–89.
[57] M. Kapl, G. Sangalli, T. Takacs, An isogeometric C^1 subspace on unstructured multi-patch planar domains, Comput. Aided Geom. Design 69 (2019) 55–75.

[58] M. Kapl, V. Vitrih, Space of C^2-smooth geometrically continuous isogeometric functions on two-patch geometries, Comput. Math. Appl. 73 (2017) 37–59.
[59] M. Kapl, V. Vitrih, Dimension and basis construction for C^2-smooth isogeometric spline spaces over bilinear-like G^2 two-patch parameterizations, J. Comput. Appl. Math. 335 (2018) 289–311.
[60] M. Kapl, G. Sangalli, T. Takacs, Construction of analysis-suitable G^1 planar multi-patch parameterizations, Comput. Aided Des. 97 (2018) 41–55.
[61] C.L. Chan, C. Anitescu, T. Rabczuk, Isogeometric analysis with strong multipatch C^1-coupling, Comput. Aided Geom. Design 62 (2018) 294–310.
[62] C.L. Chan, C. Anitescu, T. Rabczuk, Strong multipatch C^1-coupling for isogeometric analysis on 2D and 3D domains, Comput. Methods Appl. Mech. Engrg. 357 (2019) 112599.
[63] X. Wei, Y.J. Zhang, D. Toshniwal, H. Speleers, X. Li, C. Manni, J.A Evans, T.J.R. Hughes, Blended B-spline construction on unstructured quadrilateral and hexahedral meshes with optimal convergence rates in isogeometric analysis, Comput. Methods Appl. Mech. Engrg. 341 (2018) 609–639.
[64] M.J. Borden, M.A. Scott, J.A. Evans, T.J.R. Hughes, Isogeometric finite element data structures based on Bézier extraction of NURBS, Internat. J. Numer. Methods Engrg. 87 (2011) 15–47.
[65] M.A. Scott, M.J. Borden, C.V. Verhoosel, T.W. Sederberg, T.J.R. Hughes, Isogeometric finite element data structures based on Bézier extraction of T-splines, Internat. J. Numer. Methods Engrg. 88 (2011) 126–156.
[66] D. Toshniwal, H. Speleers, R.R. Hiemstra, C. Manni, T.J.R. Hughes, Multi-degree B-splines: Algorithmic computation and properties, Comput. Aided Geom. Design 76 (2020) 101792.
[67] H. Speleers, Algorithm 999: Computation of multi-degree B-splines, ACM Trans. Math. Software 45 (2019) 43.
[68] M.-J. Lai, L.L. Schumaker, Spline Functions on Triangulations, Vol. 110, Cambridge University Press, 2007.
[69] P. Dierckx, On calculating normalized Powell–Sabin B-splines, Comput. Aided Geom. Design 15 (1) (1997) 61–78.
[70] H. Speleers, A normalized basis for reduced Clough–Tocher splines, Comput. Aided Geom. Design 27 (9) (2010) 700–712.
[71] J. Kiendl, K.-U. Bletzinger, J. Linhard, R. Wüchner, Isogeometric shell analysis with Kirchhoff–Love elements, Comput. Methods Appl. Mech. Engrg. 198 (49–52) (2009) 3902–3914.
[72] A. Bartezzaghi, L. Dedè, A. Quarteroni, Isogeometric analysis of high order partial differential equations on surfaces, Comput. Methods Appl. Mech. Engrg. 295 (2015) 446–469.
[73] H. Gómez, V.M. Calo, Y. Bazilevs, T.J.R. Hughes, Isogeometric analysis of the Cahn–Hilliard phase-field model, Comput. Methods Appl. Mech. Engrg. 197 (2008) 4333–4352.
[74] J. Liu, L. Dedè, J.A. Evans, M.J. Borden, T.J.R. Hughes, Isogeometric analysis of the advective Cahn–Hilliard equation: Spinodal decomposition under shear flow, J. Comput. Phys. 242 (2013) 321–350.
[75] L. Dedè, A. Quarteroni, Isogeometric analysis for second order partial differential equations on surfaces, Comput. Methods Appl. Mech. Engrg. 284 (2015) 807–834.