

Accurate isogeometric methods for trimmed shell structures.

Présentée le 8 octobre 2021

Faculté des sciences de base
Chaire de modélisation numérique et simulation
Programme doctoral en mécanique

pour l'obtention du grade de Docteur ès Sciences

par

Luca CORADELLO

Acceptée sur proposition du jury

Prof. P. M. Nunes Pereira de Almeida Reis, président du jury
Prof. A. Buffa, Prof. A. Reali, directeurs de thèse
Prof. L. De Lorenzis, rapporteuse
Prof. B. Marussig, rapporteur
Prof. M. Picasso, rapporteur

*“Wir steigen nicht auf Berge, um Gipfel zu erreichen,
sondern heimzukehren in eine Welt, die uns als neue Chance,
als ein nochmals geschenktes Leben erscheint.”*

*“We do not climb mountains to reach summits,
but to return home to a world that appears to us as a new opportunity,
as a life given once again.”*

— Reinhold Messner

Abstract

SHELL elements derived from the Kirchhoff-Love theory have experienced a renaissance in the Finite Element community since the advent of isogeometric analysis. Despite numerous promising advancements in the field, several questions need to be addressed to establish isogeometric Kirchhoff-Love shells as an industry standard. In particular, it is known that the direct simulation of trimmed multi-patch shell models requires particular care, both from the point of view of the geometry as well as from the analysis side.

This thesis focuses on the latter. Specifically, our goal is to develop accurate and robust algorithms for the analysis of trimmed surfaces. To achieve this, we address several important aspects: (i) we systematically study the beneficial effect of local refinement for the proper resolution of localized features of the geometry/solution on complex trimming patterns, (ii) we devise a novel *a-posteriori* error estimator tailored to Kirchhoff plates and Kirchhoff-Love shells which allows us to develop a fully adaptive computational framework, (iii) we present a locking- and parameter-free coupling strategy adapted from the penalty method for achieving the required C^1 -continuity across patches and (iv) we provide a preliminary study on the impact of trimming on the critical time step in the scope of explicit Kirchhoff-Love shell dynamics.

We numerically verify and assess the performance of the aforementioned methods on an extensive selection of benchmarks defined on trimmed domains. We systematically observe an increase in accuracy compared to other established approaches. Specifically, our methods are constructed to attain the optimal accuracy of splines.

To conclude, we test the capabilities of our computational framework on several engineering structures by performing a static shell analysis on, e.g. the B-pillar of a car and the blade of a wind turbine.

Résumé

LES éléments coque dérivés de la théorie de Kirchhoff-Love ont connu une renaissance dans la communauté des éléments finis depuis l'avènement de l'analyse isogéométrique. Malgré de nombreuses avancées prometteuses dans ce domaine, plusieurs questions doivent être abordées pour que les coques isogéométriques de Kirchhoff-Love deviennent une norme industrielle. En particulier, il est connu que la simulation directe de modèles de coques multi-patch trimmées nécessite une attention particulière, tant du point de vue de la géométrie que de l'analyse.

Cette thèse se concentre sur ce dernier point. Plus précisément, notre objectif est de développer des algorithmes précis et robustes pour l'analyse des surfaces trimmées. A cette fin, nous abordons plusieurs aspects importants : (i) nous étudions systématiquement l'effet bénéfique du raffinement local pour la résolution correcte de caractéristiques localisées de la géométrie/solution sur des modèles trimmées complexes, (ii) nous concevons un nouvel estimateur d'erreur *a-posteriori* adapté aux plaques de Kirchhoff et aux coques de Kirchhoff-Love qui nous permet de développer une méthode de calcul entièrement adaptative, (iii) nous présentons une stratégie de couplage sans locking et sans paramètre, adaptée de la méthode penalty, pour obtenir la continuité C^1 requise à travers les patches, et (iv) nous fournissons une étude préliminaire de l'impact du trimming sur le pas de temps critique dans le cadre de la dynamique explicite des coques de Kirchhoff-Love.

Nous vérifions et évaluons numériquement les performances des méthodes susmentionnées sur une vaste sélection de tests benchmark définis sur des domaines trimmées. Nous observons systématiquement une augmentation de la précision par rapport aux autres approches établies. Plus précisément, nos méthodes sont construites pour atteindre la précision optimale des splines.

Pour conclure, nous testons les capacités de notre méthode de calcul sur plusieurs structures d'ingénierie en effectuant une analyse statique de coque sur, par exemple, le pilier B d'une voiture et la pale d'une éolienne.

Zusammenfassung

SCHALENELEMENTE, die von der Kirchhoff-Love-Theorie abgeleitet sind, haben seit dem Aufkommen der isogeometrischen Analyse eine Renaissance in der Finite-Elemente-Gemeinschaft erlebt. Trotz zahlreicher vielversprechender Fortschritte auf diesem Gebiet müssen noch einige Fragen geklärt werden, um isogeometrische Kirchhoff-Love-Schalen als Industriestandard zu etablieren. Insbesondere ist bekannt, dass die direkte Simulation von getrimmten mehrflächigen Modellen besondere Sorgfalt erfordert, sowohl aus Sicht der Geometrie als auch aus Sicht der Analyse.

Diese Dissertation konzentriert sich auf Letzteres. Unser Ziel ist es, genaue und zuverlässige Algorithmen für die Analyse von getrimmten Oberflächen zu entwickeln. Um dies zu erreichen, behandeln wir mehrere wichtige Aspekte: (i) wir untersuchen systematisch den Effekt der lokalen Verfeinerung für die korrekte Auflösung von lokalisierten Details der Geometrie/Lösung auf komplexen getrimmten Modellen, (ii) wir entwickeln einen neuartigen, auf Kirchhoff-Platten und Kirchhoff-Love-Schalen zugeschnittenen *a-posteriori*-Fehlerschätzer, der es uns erlaubt, ein vollständig adaptives Framework zu entwickeln, (iii) wir stellen eine locking- und parameterfreie Kopplungsstrategie vor, die an die Penalty-Methode angepasst ist, um die geforderte C^1 -Kontinuität über Patches hinweg zu erreichen, und (iv) wir liefern eine Studie über die Auswirkung des Trimmings auf den kritischen Zeitschritt im Rahmen der expliziten Kirchhoff-Love-Schalendynamik.

Durch eine umfassende Testreihe, die auf getrimmten Geometrien definiert ist, wird die Wirksamkeit der oben genannten Methoden numerisch verifiziert und bewertet. Wir beobachten systematisch eine Steigerung der Approximationsgenauigkeit im Vergleich zu anderen etablierten Ansätzen. Insbesondere sind unsere Methoden so entwickelt, dass sie die optimale Genauigkeit von Splines erreichen.

Abschließend testen wir die Anwendbarkeit unseres Berechnungsverfahrens an verschiedenen technischen Strukturen, indem wir eine statische Schalenanalyse durchführen, z. B. an der B-Säule eines Autos und dem Rotorblatt einer Windturbine.

Acknowledgements

First and foremost, I wish to thank my supervisor, Prof. Annalisa Buffa, for giving me the unique opportunity to pursue a PhD in her group. I am grateful for her constant support, her scientific advice, and the overall feeling of trust and personal freedom which led to the successful completion of this thesis. It is certain that without her guidance and the positive atmosphere she created this work would have not been possible. I am also grateful to my co-supervisor, Prof. Alessandro Reali, for his continuous interest in my research and personal development, showed since the very first time we met during my Master's thesis in Munich. His cheerful attitude and contagious enthusiasm inspired me to embark on this PhD journey.

I also wish to thank the members of the jury: Prof. Laura De Lorenzis, Prof. Benjamin Marussig and Prof. Marco Picasso, for kindly agreeing to review this work as well as for providing useful comments which improved the final version of this dissertation. Many thanks also to Prof. Pedro Reis, who kindly took on the role of president of the committee.

Furthermore, my genuine thanks goes to the great collaborators I have had the pleasure to work with during these years: Dr. Pablo Antolín, Prof. Josef Kiendl, Dr. Stefan Kollmannsberger, Prof. Ernst Rank and Dr. Rafael Vázquez. I have learned a lot from all of you and your valuable feedbacks (both on a scientific and human level) have definitely shaped my personal development. Additionally, I would like to acknowledge all the members of the MNS group for creating a positive work environment.

Like every good adventure, you need some great friends around you to be successful (and more importantly to have fun). I wish to thank my past and present roommates: Fanny, Marion, Thomas and Valérie, for making me feel at home from the get-go. Also, a big thanks to the wonderful people in Lausanne with whom I have shared many unforgettable memories, whether at work, hiking in the mountains, exploring new places around the world or at some dinner party: Alice, (Prof.) Ceci, Chiara, Edino, Fabian, (Stefano) Massei, Nicolino, (Sgt.) Statti, Riccardo. My gratitude also goes to my "international" friends. In particular, thanks to Gianna and Tito, for being able to always put a smile on my face and for their Swiss-like planning of all the weekends we have spent together in Copenhagen. Many thanks also to Alex, Davide and Massimo; I will never forget the time spent together in Munich and at conferences all around the world.

A very special thanks goes to Katy, Luca and Ondiña. It is impossible to put into words how much of this thesis is thanks to you. I cherish every moment we have spent together,

Acknowledgements

from the little coffee-and-cookie breaks, to the many (vertical) kilometers we walked and ran together, and the unforgettable adventures in the mountains.

Last, but clearly not least, I wish to thank my family. In particular, I am grateful to my parents and to my sister for their unconditional support and for always backing up any decision I have taken throughout these years. Quoting a younger me, I could not be standing here at this precise moment if it wasn't for you.

Lausanne, September 10, 2021

Luca Coradello

Contents

Abstract (English/Français/Deutsch)	i
Acknowledgements	vii
1 Introduction	1
1.1 Motivation	1
1.2 Outline	4
1.3 Implementation aspects	6
2 A review of the Finite Element Method	7
2.1 Mathematical foundations of the problem of elasticity	7
2.1.1 The strong formulation of elasticity	8
2.1.2 The weak formulation of elasticity	9
2.2 Some fundamentals of functional analysis	11
2.3 The Galerkin approximation	12
2.4 Characterization of the Finite Element space	14
2.5 Some results on the convergence of the Finite Element Method	17
3 The Kirchhoff plate and Kirchhoff-Love shell problems	19
3.1 The Kirchhoff plate problem	19
3.1.1 The strong formulation	19
3.1.2 The weak formulation	21
3.2 The Kirchhoff-Love shell problem	21
3.2.1 Some fundamentals of differential geometry	22
3.2.2 A review of continuum mechanics	24
3.2.3 The weak formulation	26
3.2.4 The strong formulation	27
3.2.5 Extension to laminate composites	28
3.2.6 Implementational details	30
4 A review of spline technologies in CAD and analysis	33
4.1 An introduction to B-splines	33
4.1.1 B-spline basis functions in one dimension	34
4.1.2 Non-Uniform Rational B-splines (NURBS) in one dimension	34

4.1.3	B-spline basis functions in multiple dimensions	35
4.1.4	B-spline curves	36
4.1.5	B-spline surfaces	37
4.1.6	Refinement strategies	37
4.2	Trimming	40
4.2.1	Boundary Representation (B-Rep)	40
4.2.2	Mathematical framework of trimming	42
4.2.3	Integration of trimmed elements	42
4.3	Local refinement of spline technologies	43
4.3.1	Hierarchical B-spline basis	44
4.3.2	Truncated hierarchical B-spline basis	46
4.3.3	Analysis-suitable Truncated Hierarchical B-splines on trimmed domains	48
4.4	Galerkin discretization of Kirchhoff plates and Kirchhoff-Love shells . .	50
4.5	Numerical results	51
4.5.1	On the imposition of weak constraints	51
4.5.2	Thin holes	53
4.5.3	Localized deformations	55
4.5.4	From CAD to analysis of an “engineering” structure	55
5	<i>A-posteriori</i> error estimation	63
5.1	Explicit error estimators	64
5.2	Implicit error estimators	66
5.2.1	Element residual method	66
5.3	Recovery-based error estimators	67
5.4	A novel error estimator for Kirchhoff plates and Kirchhoff-Love shells .	68
5.5	Adaptivity	72
5.5.1	Mark and refine	73
5.5.2	Admissible refinement of \mathcal{T}	75
5.6	Numerical results	77
5.6.1	The Poisson problem	78
5.6.2	Linear elasticity	79
5.6.3	Kirchhoff Plates	81
5.6.4	Kirchhoff-Love shell	90
6	Coupling of multi-patch Kirchhoff plates and Kirchhoff-Love shells	101
6.1	The projected super-penalty method	102
6.1.1	The multi-patch setting	103
6.1.2	The projected super-penalty formulation	105
6.2	A nested preconditioner based on the Schur Complement Reduction for non-trimmed Kirchhoff plates	114
6.2.1	The Schur Complement Reduction	114
6.2.2	Nested block preconditioner strategy based on SCR	116

6.3	Numerical examples	122
6.3.1	Coupling of Kirchhoff plates	122
6.3.2	Coupling of Kirchhoff-Love shells	135
7	A note on the critical time step in isogeometric trimmed explicit dynamics	151
7.1	Numerical stability of the time integration	151
7.2	A minimal stabilization technique for small cut elements	152
7.3	Numerical investigation	153
7.3.1	One-dimensional bar example	153
7.3.2	Two-dimensional rectangular Kirchhoff plate	155
7.3.3	Two-dimensional rotating plate	161
8	Concluding remarks and future outlook	163
8.1	Scientific contributions	163
8.1.1	Local refinement of trimmed shells	163
8.1.2	Adaptive simulation of Kirchhoff plates and Kirchhoff-Love shells	164
8.1.3	Coupling of Kirchhoff plates and Kirchhoff-Love shells	165
8.2	Future outlook	165
8.2.1	Automatic detection of refinement depth	165
8.2.2	Possible extensions of the bubble error estimator	166
8.2.3	Possible extensions of the coupling strategy	166
8.2.4	Future directions in trimmed explicit dynamics	167
	Curriculum Vitae	185

1 Introduction

1.1 Motivation

Shell structures are omnipresent in the natural realm and are widespread in many areas of engineering, see Figure 1.1 for an example taken from civil engineering. This is linked to the fact that shell structures are excellent at carrying weight through their curvature. Specifically, transversal loads can be withstand via in-plane tension and compression of the structure. Thanks to this feature, shells can be efficiently designed to maximize the so-called *stiffness-to-weight* ratio. This optimization process naturally causes one dimension of the body, typically labeled as *thickness*, to be much smaller in comparison to the others. Many shell formulations exploit this feature and are based on the dimensionality-reduction of the underlying full three-dimensional problem to a two-dimensional model. Inherently, the intrinsic slenderness of these structures yields an extremely sensitive mechanical response. This mechanical sensitivity is inherited by the corresponding computational models, rendering the devise of accurate and reliable algorithms particularly challenging. Quoting [Ramm and Wall, 2004], physics and numerics are deeply intertwined for shell structures. Therefore, it should come as no surprise that numerous different shell formulations have been developed over the years. Historically, the first shell formulation is attributed to [Love, 1888], based on the kinematic assumptions presented in [Kirchhoff, 1850]. Another fundamental theory can be traced back to the work in [Reissner, 1945; Mindlin, 1951], which, opposed to Kirchhoff-type theories, does not neglect the effects related to transverse shear deformations. Generally speaking, the Kirchhoff-Love formulation is considered accurate for thin shells. On the contrary, the Reissner-Mindlin theory is appropriate for the description of moderately thick structures. As a rule of thumb, engineers use a value of the slenderness equal to 20 to mark the limit between thick and thin shells. These two pioneering theories have given birth to a variety of formulations, where the interested reader is referred to [Bischoff et al., 2004; Reddy, 2006] and reference therein for a general overview.

The advent of the Finite Element Method (FEM) [Hrennikoff, 1941; Courant, 1943],



(a) Rolex Learning Center at EPFL, Lausanne.



(b) Olympiastadion in Munich.

Figure 1.1 – Examples of shell structures found in civil engineering. Pictures retrieved from <https://commons.wikimedia.org> under the Creative Commons license.

which gained momentum together with the advancement of computers, has revolutionized all fields of engineering. The core idea of FEM is to decompose the continuum into a *finite* number of *elements*. In two dimensions, structures are typically described by a collection of triangles and/or quadrilaterals. By employing the so-called *isoparametric* paradigm, the same functions used to describe the geometry are employed to approximate the solution field, where the typical choice consists of linear C^0 polynomials. Historically, this has hindered the development of shell elements of Kirchhoff type, since their formulations in primal form require global C^1 -continuity to be well-defined. This is the reason why Reissner-Mindlin elements and variants thereof are predominant in commercial FEM softwares. However, two important aspects need to be addressed:

- the description of curved surfaces by means of planar geometries introduces artificial geometrical imperfections. As mentioned above, the mechanical response of shells is particularly susceptible to small changes. This also introduces a mismatch between the Finite Element world and Computer-Aided Design, where the geometric modeling of free-forms is typically performed using smooth functions, namely B-splines and Non-Uniform Rational B-splines (NURBS) [Bézier, 1977; Piegl and Tiller, 1995].
- Notoriously, Reissner-Mindlin shell elements suffer from numerical locking in the thin regime. The literature on the subject is rich, e.g. we refer to [Belytschko et al., 1985; Babuška and Suri, 1992a,b; Arnold and Brezzi, 1997; Bletzinger et al., 2000]. However, a unified remedy to locking is still an active area of research, see for instance [Zou et al., 2021] and references therein for recent advancements in the field.

With the goal of mitigating the issues mentioned in the first point, isogeometric analysis (IGA) was introduced in the seminal paper [Hughes et al., 2005]. The main idea of IGA is to improve the interoperability between numerical simulations and CAD by employing the same mathematical objects used in the geometry description for the discretization of partial differential equations (PDEs). This shift in paradigm has paved the way for

an extensive amount of research, where the reader is referred to [Hughes et al., 2005; Cottrell et al., 2009; Hughes, 2017] for a detailed review of the method and its recent state-of-the-art, while its mathematical foundations can be found [Bazilevs et al., 2006; Beirão da Veiga et al., 2014]. In particular, IGA has created a major impact on shells research. Classical formulations of the Kirchhoff-Love type are governed by fourth-order PDEs. As mentioned above, this results in a global C^1 -continuity requirement which poses severe challenges to traditional finite element technologies. These obstacles are easily overcome within one isogeometric patch thanks to the higher continuity of B-splines, allowing to discretize higher-order PDEs directly in their primal form. We refer to [Kiendl et al., 2009, 2015, 2016] for a review of the method and several extensions in the scope of rotation-free isogeometric Kirchhoff-Love shells, whereas other applications to Kirchhoff plates can be found in [Reali and Gómez, 2015; Niiranen et al., 2017]. Moreover, several other spline technologies have been successfully applied to the analysis of Kirchhoff-Love shells, for instance T-splines [Bazilevs et al., 2012; Casquero et al., 2017, 2020], subdivision surfaces [Cirak et al., 2000], extended IGA [Nguyen-Thanh et al., 2015], PHT- and RHT-splines [Nguyen-Thanh et al., 2011, 2017], LR-splines [Proserpio et al., 2020], and recently extended B-splines [Schöllhammer et al., 2020].

Regarding the second point, the Kirchhoff-Love formulation has the clear advantage of avoiding shear-locking *a-priori* thanks to its kinematic assumptions. However, other sources of locking are still present, such as membrane and constraint-related locking [Bieber et al., 2018; Rafetseder and Zulehner, 2019; Guo et al., 2021], and they should be carefully considered by practitioners when interpreting the results of a numerical simulation.

The objective of this thesis is to deepen and improve the current understanding of isogeometric Kirchhoff-Love shells. Specifically, we want to develop accurate and reliable algorithms that enhance the applicability of Kirchhoff-Love shells to complex, industrial problems. To this end, we focus on several aspects which we briefly outline in the following.

- The proper treatment of trimmed surfaces needs careful consideration, see [Marussig and Hughes, 2018] for a review of the state-of-the-art and open challenges related to trimming. We recall that trimming is a standard operation for modeling complex shapes in commercial CAD softwares. Similarly to immersed methods, trimmed shape functions do not conform with the physical boundary. Consequently, a systematic way to treat trimming-related effects is pivotal to the development of any isogeometric framework. To tackle this, we study the benefits of local refinement in the analysis of trimmed shells.
- Additionally, complex geometries are typically described by multiple, non-conforming patches which demands a suitable coupling strategy to achieve the required C^1 -continuity in the context of Kirchhoff-Love shells. To achieve this, we

devise a penalty-like coupling strategy which avoids interface locking by construction and mitigates some well-known drawbacks of standard penalty methods.

- Lastly, we explore the realm of adaptivity on trimmed domains. Specifically, we devise a novel error estimator suitable for fourth-order PDEs which allows us to automatically steer an adaptive shell simulation on trimmed domains.

For each of our proposed methods, we thoroughly assess their performance on an extensive series of benchmarks. We systematically observe superior accuracy per-degree-of-freedom compared to other relevant approaches in the literature. Furthermore, we typically obtain considerable speed-ups when comparing the runtime. Finally, to test the potential of our framework in an industrial setting, we perform a static shell analysis of various models of engineering interest. These structures span several applications from civil engineering, where we consider a simplified model of the roof of the Rolex Learning Center, to mechanical engineering, where we study the B-pillar of a car and the blade of a wind turbine.

1.2 Outline

This thesis is structured as follows.

Chapter 2 provides a general overview of the Finite Element Method. In this chapter, we introduce the basic notation and mathematical foundations used throughout the rest of this manuscript. To this end, the problem of linear elasticity in its strong and weak formulations, respectively, is introduced. Then, we provide some fundamental results on the existence and uniqueness of the solution. We continue by outlining the Galerkin approximation, which constitutes the basis for the derivation of the Finite Element Method. We then rigorously describe several basic notions related to FEM. At the end of the chapter, we formalize the concept of convergence of the finite element approximation to the exact solution and we provide some classical *a-priori* estimates.

In Chapter 3, we provide an in-depth derivation of the Kirchhoff plate and Kirchhoff-Love shell formulations, respectively. Specifically, their weak and strong formulations are outlined. This allows us to introduce the terminology and framework related to plates and shells needed in the remainder of this thesis. Lastly, for ease of implementation, we summarize all the defining operators of the Kirchhoff-Love formulation in index notation.¹

Chapter 4 introduces the concepts of B-splines and their role in the geometric design of structures. Furthermore, we review their role as building blocks, i.e. the shape functions, employed in isogeometric analysis. Here, we also recall the notion of B-Representation and

¹In accordance with the Springer Copyright Transfer Statement, parts of this chapter are adapted from [Coradello et al., 2021b]. The main scientific research as well as the textual elaboration of the publication was performed by the author of this work.

trimming, two concepts that facilitate the description of complex (volumetric) geometries in modern CAD softwares. Building upon this, we briefly describe how trimming limits a direct simulation of trimmed surface in the scope of IGA. As a possible remedy, we review a solution based on the local reparametrization of the cut elements. Then, we introduce two variants of B-splines, namely hierarchical and truncated hierarchical B-splines, that enable the local refinement of the basis. This allows us to directly discretize the Kirchhoff plate and Kirchhoff-Love shell formulation, respectively, using a standard Galerkin procedure. Lastly, we present several numerical experiments that show the benefits of using local refinement for the isogeometric analysis of trimmed shells, especially when complex trimming patterns and small features are present in the design.²

In Chapter 5 we introduce the notion of *a-posteriori* error estimation in the finite element world. Specifically, we briefly review several main families of estimators present in the literature. Then, we focus on the development of a new error estimator readily applicable to Kirchhoff plates and Kirchhoff-Love shells and we discuss its extension to the trimmed case. This allows us to introduce the concept of error-driven adaptive simulation. We conclude the chapter by assessing the performance of the proposed method on an extensive series of benchmark problems. Lastly, we show the applicability of the estimator to industrial problems by performing an adaptive shell analysis of the B-pillar of a car.³

Chapter 6 presents a novel coupling strategy for enforcing displacement and rotational continuity, respectively, between non-conforming, trimmed isogeometric patches. Specifically, we formalize this penalty-like strategy by analyzing the underlying perturbed saddle-point problem. This allows us to devise a method that avoids interface locking *a-priori* and that attains the optimal rates of convergence achievable by B-splines. We then proceed to verify numerically the proposed coupling strategy on several benchmark problems described by multi-patch plates and shells. Lastly, we apply our framework to the static shell analysis of the DTU 10 MW Reference wind turbine blade.⁴

In Chapter 7 we provide some preliminary results on the behavior of the critical time step in the scope of explicit trimmed analysis of Kirchhoff-Love shells. Specifically, we shed some light on the appearance of spurious pairs of eigenvalues/eigenvectors that deteriorate the low frequency part of the spectrum. We also present a stabilization

²In accordance with the Springer Copyright Transfer Statement, parts of this chapter are adapted from [Coradello et al., 2020b]. The main scientific research and the textual elaboration of the publication have been equally developed in close collaboration between the first two authors Luca Coradello and Davide D’Angella. The co-authors of the first two authors confirm that the contributions of both Luca Coradello and Davide D’Angella were essential in this joint publication. Parts of [Coradello et al., 2020b] have been included in the dissertations of both first two authors with the approval of all co-authors.

³In accordance with the Elsevier publishing agreement, parts of this chapter are adapted from [Antolin et al., 2020; Coradello et al., 2020a]. The main scientific research as well as the textual elaboration of the publication was performed by the author of this work.

⁴In accordance with the Springer Copyright Transfer Statement, parts of this chapter are adapted from [Coradello et al., 2021b]. The main scientific research as well as the textual elaboration of the publication was performed by the author of this work.

technique that could potentially mitigate the issue when small trimmed elements are present.

Finally, in Chapter 8 we systematically draw some conclusions on the main contributions of this thesis. Then, several limitations are highlighted and possible future research directions are discussed.

1.3 Implementation aspects

Several different libraries and softwares are used in this thesis and, for the sake of completeness, are listed in the following.

The numerical examples in Chapter 4 are obtained with *AdHoC++*, a high-performance, object-oriented, high-order Finite Element library written in C++. This is a research code developed at the Chair for Computation in Engineering at the Technical University of Munich.

The methods described in Chapters 5 to 7 are implemented on top of the open-source and free OCTAVE/MATLAB package *GeoPDEs*⁵ [Vázquez, 2016]. Additionally, the reparametrization of trimmed elements for integration purposes is obtained with the in-house tool presented in [Antolin et al., 2019]. This tool makes use of the open-source mesh generator *Gmsh*⁶ [Geuzaine and Remacle, 2009] and its associated geometric kernel *OpenCASCADE*⁷ [OpenCASCADE, 2018].

Throughout this thesis, complex CAD models are handled in the commercial software *Rhinoceros*⁸ [McNeel et al., 2010]. The aforementioned models are exported in STEP format [ISO 10303-11, 1994] and are read into our framework via PYTHON wrappers built on top of *pyOCCT*⁹ [Laughlin, 2020], a collection of bindings to *OpenCASCADE*.

⁵<http://rafavzqz.github.io/geopdes/>

⁶<https://gmsh.info/>

⁷<https://www.opencascade.com/>

⁸<https://www.rhino3d.com/>

⁹<https://github.com/trelau/pyOCCT>

2 A review of the Finite Element Method

In this chapter we review some fundamentals of the Finite Element Method (FEM). In particular, we lie the mathematical foundations that describe numerous physical phenomena in their differential and variational form, respectively. To exemplify these concepts, we analyze in details the problem of linear elasticity. Then, we address the concept of existence and uniqueness of the solution of the corresponding boundary value problem. Next, we describe the standard finite element terminology and we summarize the key steps required to construct a finite element discretization. Finally, we provide some classical results of *a-priori* convergence of the method with respect to the analytical solution. The notation and derivation used in this chapter follow closely [Hughes, 2000; Holzapfel, 2001; Ciarlet, 2002; Brenner and Scott, 2008; Quarteroni and Valli, 2008].

2.1 Mathematical foundations of the problem of elasticity

In this work, we make the assumption that object can be modeled as a continuum. This macroscopic view yields highly accurate results for the typical length scales considered in many engineering applications. For a detail review of continuum mechanics, we refer the reader to [Holzapfel, 2001]. This mathematical framework allows for the description of physical phenomena by leveraging the notion of *field*, e.g. a quantity of interest associated with every point in space and time. Following standard nomenclature, we refer to a field as scalar, vector or tensor if it is represented by a zeroth, first or higher order tensor, respectively. In the following derivation, we will use the problem of linear elasticity as a recurring example, where the typical fields of interests are displacements and stresses. We remark that, although the terminology will be specific to linear elasticity, the abstract framework introduced in the following can be directly applied to a wide range of problems from physics and engineering.

2.1.1 The strong formulation of elasticity

Let us consider an arbitrary, bounded, open and connected body $\Omega \subset \mathbb{R}^3$ and its Lipschitz-continuous boundary $\partial\Omega$. As typically done, we can define a conservation or balance law over this volume as a function of time. In particular, let us associate to any spatial point \mathbf{x} at any given instant t a mass density $\rho = \rho(\mathbf{x}, t)$ and a velocity field $\dot{\mathbf{u}} = \dot{\mathbf{u}}(\mathbf{x}, t)$, where \mathbf{u} denotes the displacement of the body and the dot indicates differentiation with respect to time. We can now introduce the conservation of linear momentum \mathbf{L} in integral form as:

$$\dot{\mathbf{L}}(t) = \int_{\Omega} \rho \ddot{\mathbf{u}} \, d\Omega = \mathbf{F}(t), \quad (2.1)$$

where $\mathbf{F}(t)$ denotes the total external force applied to the body and $\ddot{\mathbf{u}}$ stands for the corresponding acceleration field. Let us also define a vector field $\mathbf{b} = \mathbf{b}(\mathbf{x}, t) \in [L^2(\Omega)]^3$ which represents the body force in Ω and a traction vector $\mathbf{t} = \mathbf{t}(\mathbf{x}, t) \in [L^2(\partial\Omega)]^3$ which denotes the force acting on the boundary surface $\partial\Omega$. With these definitions at hand, $\mathbf{F}(t)$ can be rewritten as:

$$\mathbf{F}(t) = \int_{\Omega} \mathbf{b} \, d\Omega + \int_{\partial\Omega} \mathbf{t} \, d\gamma. \quad (2.2)$$

Now, let us assume that there exists a tensor field $\boldsymbol{\sigma} = \boldsymbol{\sigma}(\mathbf{x}, t)$ such that:

$$\mathbf{t} = \boldsymbol{\sigma} \cdot \mathbf{n}, \quad (2.3)$$

where \mathbf{n} denotes the unit normal vector to $\partial\Omega$ and $\boldsymbol{\sigma}$ is the so-called Cauchy stress tensor. We highlight that $\boldsymbol{\sigma}$ is symmetric, where the symmetry of the latter can be shown from the balance of angular momentum, for further details we refer to [Holzapfel, 2001]. By leveraging the divergence theorem and the relation in Equation (2.3) we can convert the surface integral in Equation (2.2) to a volume integral as:

$$\int_{\partial\Omega} \mathbf{t} \, d\gamma = \int_{\partial\Omega} \boldsymbol{\sigma} \mathbf{n} \, d\gamma = \int_{\Omega} \nabla \cdot \boldsymbol{\sigma} \, d\Omega. \quad (2.4)$$

By substituting the result obtained in Equation (2.4) into Equation (2.1) we get:

$$\int_{\Omega} (\nabla \cdot \boldsymbol{\sigma} + \mathbf{b} - \rho \ddot{\mathbf{u}}) \, d\Omega = 0. \quad (2.5)$$

Given the arbitrariness of Ω we can write the equation of motion in its classical differential form:

$$\nabla \cdot \boldsymbol{\sigma} + \mathbf{b} = \rho \ddot{\mathbf{u}} \quad \forall \mathbf{x} \in \Omega, \forall t \in [0, +\infty]. \quad (2.6)$$

In this work, we assume that inertial effects are negligible and therefore we will work in the realm of elastostatics. Consequently, the result presented in Equation (2.6) can be

2.1. Mathematical foundations of the problem of elasticity

simplified to:

$$\nabla \cdot \boldsymbol{\sigma} + \mathbf{b} = 0 \quad \forall \mathbf{x} \in \Omega, \quad (2.7)$$

which describes the equilibrium of a body in the absence of accelerations acting on it. Equation (2.7) is commonly referred to as the *strong* form of the problem.

Now, let us assume a suitable decomposition of the boundary $\partial\Omega$ into two disjoint parts Γ_D and Γ_N , representing the Dirichlet and Neumann part of the boundary, respectively. Further, it holds that $\partial\Omega = \overline{\Gamma_D \cup \Gamma_N}$ and $\emptyset = \Gamma_D \cap \Gamma_N$. Given a prescribed traction force $\tilde{\mathbf{t}} \in [L^2(\Gamma_N)]^3$ on Γ_N and a prescribed displacement $\tilde{\mathbf{u}} \in [H^{\frac{1}{2}}(\Gamma_D)]^3$ on Γ_D , the strong form is complemented by the following set of boundary conditions:

$$\boldsymbol{\sigma} \cdot \mathbf{n} = \tilde{\mathbf{t}} \quad \forall \mathbf{x} \in \Gamma_N \quad (2.8a)$$

$$\mathbf{u} = \tilde{\mathbf{u}} \quad \forall \mathbf{x} \in \Gamma_D. \quad (2.8b)$$

2.1.2 The weak formulation of elasticity

Now, let us define the following space:

$$S = \left\{ \mathbf{u} \in [H^1(\Omega)]^3 \mid \mathbf{u} = \tilde{\mathbf{u}} \text{ on } \Gamma_D \right\} := [H_{\tilde{\mathbf{u}}}^1(\Omega)]^3, \quad (2.9)$$

where Γ_D is a subset of $\partial\Omega$ with strictly positive measure. The members of S are typically referred to as *trial* functions in the literature. Similarly, we can define the space of *test* functions as:

$$V = \left\{ \mathbf{v} \in [H^1(\Omega)]^3 \mid \mathbf{v} = \mathbf{0} \text{ on } \Gamma_D \right\} := [H_{\Gamma_D}^1(\Omega)]^3, \quad (2.10)$$

which represents the homogeneous counterpart of S , where the statement $\mathbf{v} = \mathbf{0}$ on Γ_D means that the trace of \mathbf{v} is vanishing on Γ_D . We also remark that if $\Gamma_D = \partial\Omega$, we will simply use the notation $[H_0^1(\Omega)]^3$. Now, we can multiply Equation (2.7) by the test functions \mathbf{v} and integrate over the domain Ω , which yields:

$$\int_{\Omega} \mathbf{v} \cdot (\nabla \cdot \boldsymbol{\sigma}) \, d\Omega + \int_{\Omega} \mathbf{v} \cdot \mathbf{b} \, d\Omega = 0. \quad (2.11)$$

Given that \mathbf{v} is square-integrable, the following holds true:

$$\nabla \cdot (\mathbf{v} \cdot \boldsymbol{\sigma}) = \mathbf{v} \cdot (\nabla \cdot \boldsymbol{\sigma}) + \nabla \mathbf{v} : \boldsymbol{\sigma}. \quad (2.12)$$

Consequently, Equation (2.11) can be rewritten as:

$$\int_{\Omega} \nabla \cdot (\mathbf{v} \cdot \boldsymbol{\sigma}) \, d\Omega - \int_{\Omega} \nabla \mathbf{v} : \boldsymbol{\sigma} \, d\Omega + \int_{\Omega} \mathbf{v} \cdot \mathbf{b} \, d\Omega = 0. \quad (2.13)$$

By leveraging the divergence theorem, the first integral can be reformulated as follows:

$$\int_{\Omega} \nabla \cdot (\mathbf{v} \cdot \boldsymbol{\sigma}) \, d\Omega = \int_{\partial\Gamma} \mathbf{v} \cdot \boldsymbol{\sigma} \cdot \mathbf{n} \, d\gamma = \underbrace{\int_{\Gamma_D} \mathbf{v} \cdot \boldsymbol{\sigma} \cdot \mathbf{n} \, d\gamma}_{=0} + \int_{\Gamma_N} \mathbf{v} \cdot \boldsymbol{\sigma} \cdot \mathbf{n} \, d\gamma, \quad (2.14)$$

where the integral on the Dirichlet portion of the boundary vanishes due to the choice of test functions. Then, by using the boundary condition in Equation (2.8a) and the result in Equation (2.14) we obtain:

$$\int_{\Omega} \nabla \mathbf{v} : \boldsymbol{\sigma} \, d\Omega = \int_{\Omega} \mathbf{v} \cdot \mathbf{b} \, d\Omega + \int_{\Gamma_N} \mathbf{v} \cdot \tilde{\mathbf{t}} \, d\gamma. \quad (2.15)$$

Now, let us introduce the linearized strain tensor $\boldsymbol{\varepsilon}$ as:

$$\boldsymbol{\varepsilon}(\mathbf{u}) = \frac{1}{2} \left(\nabla \mathbf{u} + \nabla \mathbf{u}^\top \right), \quad (2.16)$$

where higher-order terms have been neglected. Furthermore, we assume that the body deforms in the linear elastic regime, meaning that there exists a constitutive relation between the stress and strain tensors of the form:

$$\boldsymbol{\sigma}(\mathbf{u}) = \mathbb{C} : \boldsymbol{\varepsilon}(\mathbf{u}), \quad (2.17)$$

where \mathbb{C} denotes the fourth-order material tensor. If we consider an isotropic linear elastic material, its components read:

$$\mathbb{C}_{ijkl} = \lambda \delta_{ij} \delta_{kl} + \mu (\delta_{ik} \delta_{jl} + \delta_{il} \delta_{jk}), \quad (2.18)$$

where λ and μ denote the first and second Lamé parameters, respectively, and δ represents the standard Kronecker delta. We recall that \mathbb{C} can be equivalently defined in terms of the Young's modulus E and the Poisson's ratio ν by using the following relationships:

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)}, \quad (2.19a)$$

$$\mu = \frac{E}{2(1+\nu)}. \quad (2.19b)$$

With these definitions and by exploiting the symmetry of the stress tensor $\boldsymbol{\sigma}$, the integrand on the left-hand-side of Equation (2.15) can be rewritten in terms of the strain tensor:

$$\nabla \mathbf{v} : \boldsymbol{\sigma} = \frac{1}{2} \left(\nabla \mathbf{v} : \boldsymbol{\sigma} + \nabla \mathbf{v}^\top : \boldsymbol{\sigma}^\top \right) = \frac{1}{2} \left(\nabla \mathbf{v} : \boldsymbol{\sigma} + \nabla \mathbf{v}^\top : \boldsymbol{\sigma} \right) = \boldsymbol{\varepsilon}(\mathbf{v}) : \boldsymbol{\sigma}. \quad (2.20)$$

2.2. Some fundamentals of functional analysis

Finally, the *weak* formulation of the problem of linear elasticity can be stated as:

$$\begin{aligned} &\text{Find } \mathbf{u} \in H_{\tilde{\mathbf{u}}}^1(\Omega) \text{ such that:} \\ &a(\mathbf{u}, \mathbf{v}) = f(\mathbf{u}) \quad \forall \mathbf{v} \in \left[H_{\Gamma_D}^1(\Omega) \right]^3, \end{aligned} \quad (2.21)$$

where a is a symmetric, coercive and continuous bilinear form and f is a continuous linear form. They can be expanded, respectively, as follows:

$$a(\mathbf{u}, \mathbf{v}) = \int_{\Omega} \boldsymbol{\sigma}(\mathbf{u}) : \boldsymbol{\varepsilon}(\mathbf{v}) \, d\Omega \quad (2.22a)$$

$$f(\mathbf{v}) = \int_{\Omega} \mathbf{v} \cdot \mathbf{b} \, d\Omega + \int_{\Gamma_N} \mathbf{v} \cdot \tilde{\mathbf{t}} \, d\gamma. \quad (2.22b)$$

2.2 Some fundamentals of functional analysis

In the following, we summarize some classical results of functional analysis that show the existence and uniqueness of the solution \mathbf{u} . This abstract framework constitutes the mathematical foundations upon which numerical methods have been successfully devised. For the sake of simplicity, let us first consider the case $S = V = \left[H_{\Gamma_D}^1(\Omega) \right]^3$. We can now state the following fundamental result.

Theorem 2.1 (Lax-Milgram lemma [Quarteroni and Valli, 2008, Theorem 5.1.1])

Let V be a (real) Hilbert space, endowed with the norm $\|\cdot\|$, $a(\mathbf{u}, \mathbf{v}) : V \times V \rightarrow \mathbb{R}$ a bilinear form and $f(\mathbf{v}) : V \rightarrow \mathbb{R}$ a linear continuous functional, i.e., $f \in V'$, where V' denotes the dual space of V . Assume moreover that $a(\cdot, \cdot)$ is continuous, i.e.,

$$\exists \gamma > 0 : |a(\mathbf{w}, \mathbf{v})| \leq \gamma \|\mathbf{w}\| \|\mathbf{v}\| \quad \forall \mathbf{w}, \mathbf{v} \in V, \quad (2.23)$$

and coercive, i.e.,

$$\exists \alpha > 0 : |a(\mathbf{v}, \mathbf{v})| \geq \alpha \|\mathbf{v}\|^2 \quad \forall \mathbf{v} \in V. \quad (2.24)$$

Then, there exists a unique $\mathbf{u} \in V$ solution to Equation (2.21) and:

$$\|\mathbf{u}\| \leq \frac{1}{\alpha} \|f\|_{V'}. \quad (2.25)$$

The result in Equation (2.25) states that the solution \mathbf{u} *continuously* depends on the input data f , resulting in the well-posedness of the underlying problem. The Lax-Milgram lemma can be extended to the more general case $S \neq V$ as follows.

Theorem 2.2 ([Quarteroni and Valli, 2008, Theorem 5.1.2]) *Let S and V be two (real) Hilbert spaces, with norms $|||\cdot|||$ and $\|\cdot\|$, respectively. Assume that there exist two*

positive constants α and γ such that the bilinear form $a : S \times V \rightarrow \mathbb{R}$ satisfies:

$$|a(\mathbf{w}, \mathbf{v})| \leq \gamma |||\mathbf{w}||| \|\mathbf{v}\| \quad \forall \mathbf{w} \in S, \forall \mathbf{v} \in V, \quad (2.26)$$

$$\sup_{\mathbf{v} \in V, \mathbf{v} \neq \mathbf{0}} \frac{a(\mathbf{w}, \mathbf{v})}{\|\mathbf{v}\|} \geq \alpha |||\mathbf{w}||| \quad \forall \mathbf{w} \in S, \quad (2.27)$$

$$\sup_{\mathbf{w} \in S} a(\mathbf{w}, \mathbf{v}) > 0 \quad \forall \mathbf{v} \in V, \mathbf{v} \neq \mathbf{0}. \quad (2.28)$$

Then, for any $f \in V'$, there exists a unique solution $\mathbf{u} \in S$ of Equation (2.21) which satisfies:

$$|||\mathbf{u}||| \leq \frac{1}{\alpha} \|f\|_{V'}. \quad (2.29)$$

Assuming that the material tensor \mathbb{C} is bounded away from zero and infinity, the Lax-Milgram lemma guarantees the existence and uniqueness of the solution to the elasticity problem. For further details and a rigorous proof, the interested reader is referred to [Ciarlet, 2002; Brenner and Scott, 2008; Quarteroni and Valli, 2008].

2.3 The Galerkin approximation

It is well-known that finding a closed-form solution of Equation (2.21) can be achieved only in simplified scenarios. As a consequence, an enormous amount of effort has been put into the development of methods that suitably approximate the solution of Equation (2.21). These techniques are based on a discretization of the variational problem in a way that exploits the computational power of modern computers. In the scope of computational mechanics, one of the most prominent approaches is the so-called *(Bubnov-)Galerkin* approximation, which we summarize in the following.

Let us first define the following spaces:

$$S_h \subset S \quad (2.30a)$$

$$V_h \subset V, \quad (2.30b)$$

which denote the finite-dimensional counterpart of the previously introduced test and trial spaces. Before proceeding, we need to address the unbalance of these function spaces in case of inhomogeneous Dirichlet boundary conditions. Following the derivation in [Hughes, 2000], for every function $\mathbf{v}_h \in V_h$ we can construct a corresponding function $\mathbf{u}_h \in S_h$ as follows:

$$\mathbf{u}_h = \mathbf{v}_h + \mathbf{q}_h, \quad (2.31)$$

where \mathbf{q}_h satisfied the prescribed essential boundary conditions. With this definition, the

Galerkin approximation of Equation (2.21) reads:

$$\begin{aligned} &\text{Find } \mathbf{u}_h = \mathbf{v}_h + \mathbf{q}_h, \text{ where } \mathbf{v}_h \in V^h \text{ such that:} \\ &a(\mathbf{w}_h, \mathbf{v}_h) = f(\mathbf{w}_h) - a(\mathbf{w}_h, \mathbf{q}_h) \quad \forall \mathbf{w}_h \in V_h. \end{aligned} \quad (2.32)$$

Here, we report another important convergence result.

Theorem 2.3 ([Quarteroni and Valli, 2008, Theorem 5.2.1]) *Under the assumptions of Theorem 2.1 there exists a unique solution \mathbf{u}_h to Equation (2.32), which furthermore is stable since:*

$$\|\mathbf{u}_h\| \leq \frac{1}{\alpha} \|f\|_{V'} . \quad (2.33)$$

Moreover, if \mathbf{u} is the solution of Equation (2.21), it follows:

$$\|\mathbf{u} - \mathbf{u}_h\| \leq \frac{\gamma}{\alpha} \inf_{\mathbf{w}_h \in V_h} \|\mathbf{u} - \mathbf{w}_h\| . \quad (2.34)$$

This estimate shows the convergence of the Galerkin approximation to the true solution of the underlying problem and it is commonly referred to as *Céa's lemma* in the literature. Moreover, it states that the approximated solution \mathbf{u}_h is the best approximation of \mathbf{u} in V_h . Due to the symmetry of the bilinear form a , $a(\cdot, \cdot)$ represents an inner product on V which induces the following norm:

$$\|\mathbf{v}\|_{E(\Omega)} = \sqrt{a(\mathbf{v}, \mathbf{v})} , \quad (2.35)$$

defining the so-called *energy norm*, which stems from the underlying physical interpretation of a . This allows us to restate Céa's lemma in terms of the energy norm as follows:

$$\|\mathbf{u} - \mathbf{u}_h\|_{E(\Omega)} \leq \|\mathbf{u} - \mathbf{w}_h\|_{E(\Omega)} \quad \forall \mathbf{w}_h \in V_h . \quad (2.36)$$

This result can also be interpreted geometrically; indeed \mathbf{u}_h is the a -orthogonal projection of the exact solution \mathbf{u} onto the finite-dimensional subspace V_h , see e.g. [Ciarlet, 2002] for further details. Now, we need to further characterize the finite-dimensional space V_h . In particular, V_h is spanned by a finite number of basis $\mathbf{N} = [N_1, N_2, \dots, N_n]$, where n represents the dimension of V_h . This implies that every member \mathbf{w}_h of V_h can be written as a linear combination of the basis functions and their corresponding coefficients as follows:

$$\mathbf{w}_h = \sum_{i=1}^n N_i \tilde{c}_i = \mathbf{N} \tilde{\mathbf{c}} , \quad (2.37)$$

where $\tilde{\mathbf{c}} \in \mathbb{R}^n$ denotes the vector collecting the coefficients \tilde{c}_i of the linear combination.

Accordingly, Equation (2.32) can be rewritten in a discrete form as:

$$\begin{aligned} &\text{Find } \tilde{\mathbf{u}}_h \in \mathbb{R}^n \text{ such that:} \\ &a(N_i \tilde{c}_i, N_j \tilde{u}_j) = f(N_i \tilde{c}_i) \quad \forall \tilde{\mathbf{c}} \in \mathbb{R}^n, \end{aligned} \quad (2.38)$$

where we consider the homogeneous case $\mathbf{q}_h = 0$ for simplicity of notation and where the Einstein summation convention applies. If inhomogeneous essential conditions are prescribed, it suffices to recover the sought solution from Equation (2.31). By exploiting the linearity of a and f and by recalling that the coefficients c_i are arbitrary we get:

$$a(N_i, N_j) \tilde{u}_j = f(N_i). \quad (2.39)$$

This result can be expressed in matrix form:

$$\mathbf{K} \tilde{\mathbf{u}} = \mathbf{f}, \quad (2.40)$$

where \mathbf{K} commonly denotes the stiffness matrix, \mathbf{f} represents the load vector and $\tilde{\mathbf{u}}$ is the vector of solution coefficients.

2.4 Characterization of the Finite Element space

As highlighted in the previous section, the choice of V_h is pivotal to the devise of the Galerkin approximation. In particular, the choice of basis \mathbf{N} should allow to develop a systematic routine for tackling arbitrarily complex problem. Therefore, the following properties are essential:

- geometrical flexibility,
- suitable for a computer-based implementation,
- robust convergence based on rigorous error estimates.

The main ingredients needed to achieve these features are summarized in the following.

The Finite Element mesh

The first step in a traditional Finite Element routine deals with the construction of a suitable *mesh*. This procedure consists in sub-dividing the domain Ω into a finite number of polyhedra, where, following the definitions provided in [Quarteroni and Valli, 2008]:

$$\bar{\Omega} = \bigcup_{K \in \mathcal{T}} K, \quad (2.41)$$

where:

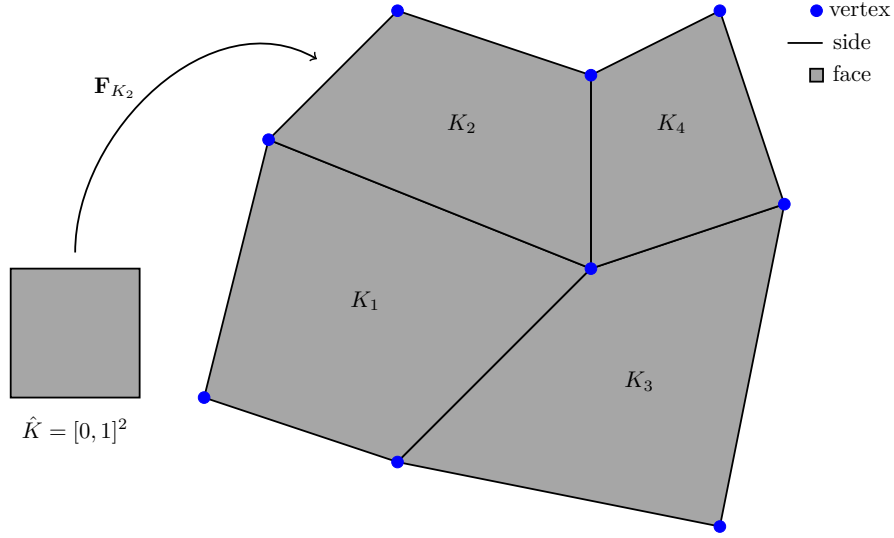


Figure 2.1 – Example of Finite Element mesh \mathcal{T} .

- each K is a polyhedron with $\mathring{K} \neq \emptyset$;
- $\mathring{K}_i \cap \mathring{K}_j = \emptyset$ for each distinct $\mathring{K}_i, \mathring{K}_j \in \mathcal{T}$;
- if $K_i \cap K_j \neq \emptyset$ (K_i and K_j distinct elements of \mathcal{T}) then F is a common face, side, or vertex of K_1 and K_2 .
- $\text{diam}(K) \leq h$ for each $K \in \mathcal{T}$, where:

$$h = \max_{K' \in \mathcal{T}} \text{diam}(K'). \quad (2.42)$$

\mathcal{T} is also commonly called a *triangulation* of $\bar{\Omega}$ in the literature. However, since this thesis focuses only on quadrilateral elements, we will use the term mesh in the remaining of this manuscript to avoid confusion. In order to automatize the procedure, we assume that each element K is obtained starting from a reference polyhedron \hat{K} , which is then mapped onto K via a bijective mapping $\mathbf{F}_K : \hat{K} \rightarrow K$. In particular, we restrict ourselves to the case where \hat{K} is the unit square $[0, 1]^2$. Finally, the key components of a Finite Element mesh are depicted in Figure 2.1.

The Finite Element space

Once we have constructed a suitable description of the geometry, we are ready to define the finite dimensional space $V_h \subset V$ needed in Equation (2.32). The fundamental observation is that functions $\mathbf{v}_h \in V_h$ are piecewise polynomials and therefore, for each element $K \in \mathcal{T}$, the space:

$$\mathcal{P} := \{\mathbf{v}_h|_K; \mathbf{v}_h \in V_h\} \quad (2.43)$$

consists of polynomials [Ciarlet, 2002; Quarteroni and Valli, 2008]. This property turns out to be crucial for the convergence of FEM. To develop a systematic strategy, let us first define the polynomial space of degree $p \in \mathbb{N}$ on the reference element \hat{K} as [Ainsworth and Oden, 1997]:

$$\hat{\mathcal{P}}(p) = \text{span}\{\hat{N}_1, \hat{N}_2, \dots, \hat{N}_{n(p)}\}, \quad (2.44)$$

where \hat{N}_i represent the standard basis functions, representing a collection of linearly independent polynomials up to degree p defined on the parametric element \hat{K} . Given the geometric mapping \mathbf{F}_K , the basis functions can be pushed forward to the physical element K :

$$N_i = \hat{N}_i \circ \mathbf{F}_K^{-1} \quad i = 1, \dots, n(p), \quad (2.45)$$

which allows us to define the space of polynomials over K as:

$$\mathcal{P} = \text{span}\{N_i\}. \quad (2.46)$$

Following [Ciarlet, 2002; Brenner and Scott, 2008], let us provide the definition of a finite element.

Definition 2.1 *Let*

- K be a bounded closed set with nonempty interior and piecewise smooth boundary.
- \mathcal{P} be a finite-dimensional space of functions on K (the space of shape functions) and
- $\mathcal{N} = \{\phi_1, \phi_2, \dots, \phi_{n(p)}\}$ be a basis for \mathcal{P}' (the set of nodal variables).

Then $(K, \mathcal{P}, \mathcal{N})$ is called a finite element.

Thanks to Definition 2.1, we can finally characterize V_h as:

$$V_h = \{\mathbf{v}_h \in C^0(\bar{\Omega}) : \mathbf{v}_h|_K \in \mathcal{P}(K), \forall K \in \mathcal{T}\}. \quad (2.47)$$

To fix ideas, an example is provided in Figure 2.2.

Remark 2.1 *We highlight that in many engineering applications of interest, we need additional continuity requirements on the space V_h . Of particular interest to this thesis are Kirchhoff plates and Kirchhoff-Love shells, which demand for global C^1 -continuity of the basis functions. This aspect will be addressed in details in a later chapter.*

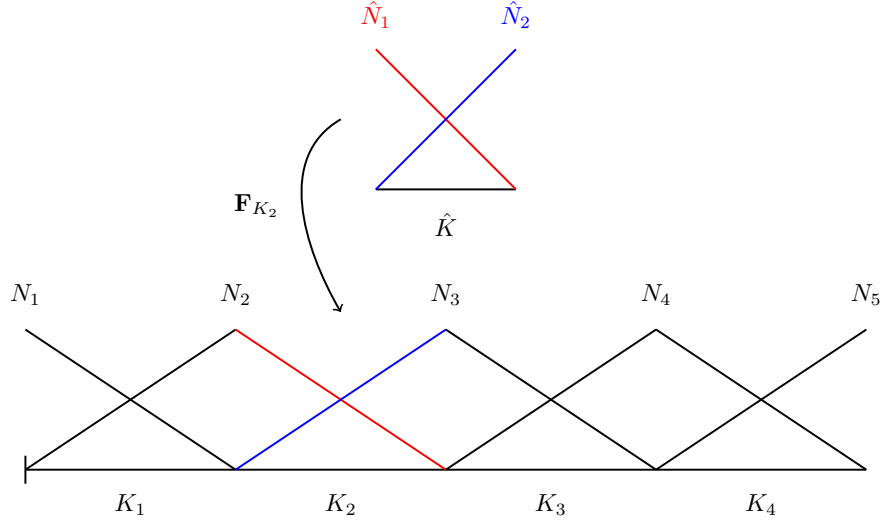


Figure 2.2 – Example of one-dimensional shape functions \hat{N}_i and N_i on the parametric element and corresponding physical elements, respectively, associated with a polynomial degree $p = 1$.

2.5 Some results on the convergence of the Finite Element Method

In the following, we summarize some classical *a priori* estimates on the convergence behavior of FEM. In particular, we provide an upper bound on the error $e_h = \mathbf{u} - \mathbf{u}_h$ between the true solution of the continuous problem and its discrete counterpart. First of all, we need to define the notion of convergence of the discrete solution, which can be expressed as:

$$\lim_{h \rightarrow 0} \|\mathbf{u} - \mathbf{u}_h\| = 0, \quad (2.48)$$

where $\|\cdot\|$ denotes the norm associated with the space V . In words, convergence holds if the error measured in the V -norm tends to zero under mesh refinement. Then, we note that Céa's lemma allows us to shift the problem of estimating the numerical error to a problem in approximation theory. Indeed, we want to measure the distance:

$$d(\mathbf{u}, V_h) = \inf_{\mathbf{w}_h \in V_h} \|\mathbf{u} - \mathbf{w}_h\|, \quad (2.49)$$

between a function $\mathbf{u} \in V$ and a finite-dimensional subspace $V_h \subset V$. Given suitable assumptions on the smoothness of \mathbf{u} , it can be proven that the *a priori* error bound is given as:

$$\|\mathbf{u} - \mathbf{u}_h\| \leq C(\mathbf{u})h^\beta, \quad (2.50)$$

where

$$h = \max_{K \in \mathcal{T}} h_K \quad (2.51)$$

denotes the maximum mesh size and C is a constant independent of h and for some exponent $\beta > 0$. Let us now further characterize this error bound with respect to the smoothness of the solution and to the norm chosen to measure the error. For the sake of brevity, here we provide only the most relevant results for this thesis, where for a rigorous mathematical derivation we refer to [Ciarlet, 2002].

Theorem 2.4 ([Ciarlet, 2002, Theorem 3.1.6]) *Let there be given a regular affine family of finite elements $(K, \mathcal{P}_K, \mathcal{N}_K)$ whose reference finite element $(\hat{K}, \hat{\mathcal{P}}, \hat{\mathcal{N}})$ satisfies the following inclusions for some integers $m \geq 0$ and $k \geq 0$ and for some numbers $p, q \in [1, \infty]$:*

$$W^{k+1,p}(\hat{K}) \hookrightarrow \mathcal{C}^s(\hat{K}), \quad (2.52a)$$

$$W^{k+1,p}(\hat{K}) \hookrightarrow W^{m,q}(\hat{K}), \quad (2.52b)$$

$$\mathcal{P}_k(\hat{K}) \subset \hat{\mathcal{P}} \subset W^{m,q}(\hat{K}), \quad (2.52c)$$

where we use the standard notation $W^{m,p}(\Omega)$ to denote the Sobolev space consisting of those functions $v \in L^p(\Omega)$ such that all their partial (distributional) derivatives $\partial^\beta v$ satisfying $|\beta| \leq m$ belong to space $L^p(\Omega)$. Under these assumptions, there exists a constant $C(\hat{K}, \hat{\mathcal{P}}, \hat{\mathcal{N}})$ such that, for all finite elements K in the family, and for all functions $v \in W^{k+1,p}(K)$,

$$\|v - \Pi_K v\|_{m,q,K} \leq C(\hat{K}, \hat{\mathcal{P}}, \hat{\mathcal{N}}) (\text{meas}(K))^{(1/q)-(1/p)} h_K^{k+1-m} |v|_{k+1,p,K}. \quad (2.53)$$

Building upon this result of approximation theory and extending it to the entire domain Ω , we can finally write a more compact and practical form of the estimate provided in Equation (2.53):

$$\|\mathbf{u} - \mathbf{u}_h\|_{H^s(\Omega)} \leq C h^{l-s} \|\mathbf{u}\|_{H^r(\Omega)} \quad r > s, \quad (2.54)$$

where r represents the regularity of the exact solution and l is defined as $l = \min(r, p+1)$. In our numerical results, we will often use this estimate to verify the optimal convergence behavior of a numerical scheme. We remark that the aforementioned result holds for $h \rightarrow 0$, meaning that in practice we need to select a fine enough mesh such that the error behaves asymptotically.

3 The Kirchhoff plate and Kirchhoff-Love shell problems

In this chapter we describe the governing equations associated with Kirchhoff plates and Kirchhoff-Love shells, respectively. At first, analogously to the derivation followed in Chapter 2, we provide the strong formulations of the respective problems. Then, we state the equations in their variational form, where the standard Galerkin discretization is employed in both cases. We recall that Kirchhoff-type theories are dimensionally-reduced approaches based on a two-dimensional description of the underlying physics. This simplification is justified when objects exhibit one dimension significantly smaller than the others, which is commonly denoted as *thickness* direction. From a practical point of view, this allows us to describe the mechanical response of the structure only in terms of quantities defined on its mid surface, thereby reducing the computational complexity. The full three-dimensional behavior is then recovered by leveraging suitable kinematic assumptions. We remark that the notation and derivation presented in this chapter follow [Kiendl et al., 2009; Kiendl, 2011; Reali and Gómez, 2015; Schillinger et al., 2016; Benzaken et al., 2021].

3.1 The Kirchhoff plate problem

Historically, the first plate theory is attributed to [Kirchhoff, 1850], which laid the theoretical foundations for many modern plate formulations. In the following, we review its main aspects and recast its derivation into the finite element framework.

3.1.1 The strong formulation

Let us introduce the bending-dominated Kirchhoff plate problem, governed by the fourth-order bi-Laplace differential operator. With this assumption, the sought solution field is a scalar value associated with the deflection of structure. Let us recall the definition of the computational domain as an open set $\Omega \subset \mathbb{R}^3$ with a sufficiently smooth boundary $\partial\Omega$, such that the normalized normal vector \mathbf{n} and the unit tangent vector \mathbf{t} are well-defined

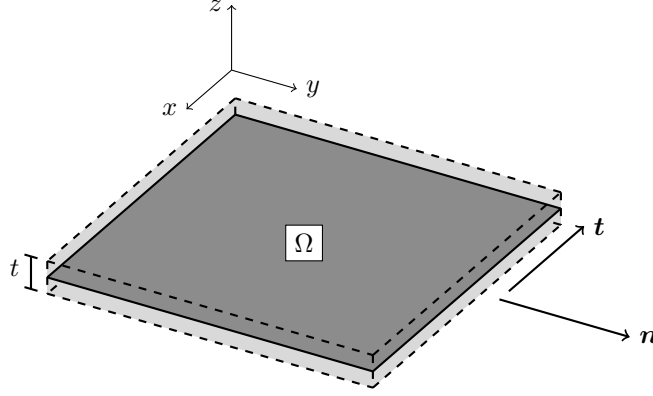


Figure 3.1 – Example of a plate geometry. The normal and tangent vectors \mathbf{n} and \mathbf{t} , respectively, have been separated for visualization purposes.

(almost) everywhere, see Figure 3.1 for an example. Additionally, we assume that the boundary $\Gamma = \partial\Omega$ can be decomposed into $\Gamma = \overline{\Gamma_u} \cup \overline{\Gamma_Q}$ and $\Gamma = \overline{\Gamma_\phi} \cup \overline{\Gamma_M}$, such that $\Gamma_u \cap \Gamma_Q = \emptyset$ and $\Gamma_\phi \cap \Gamma_M = \emptyset$, respectively. These conditions are related to the energetically conjugated nature of the deflection u and the transverse shear Q , and the rotations ϕ and bending moments M , respectively. Moreover, we introduce the set of corners $\chi \subset \Gamma$ along the domain boundary, where the normal vector presents a discontinuity. This set can be further split into a Neumann part $\chi_N = \chi \cap (\Gamma_Q \cup \Gamma_M)$ and a Dirichlet part $\chi_D = \chi \cap (\Gamma_u \cup \Gamma_\phi)$, respectively. We can formulate the strong form of the problem as follows:

$$D\Delta^2 u = g \quad \text{in } \Omega, \quad (3.1a)$$

$$u = \tilde{u} \quad \text{on } \Gamma_u, \quad (3.1b)$$

$$-\nabla u \cdot \mathbf{n} = \tilde{\phi} \quad \text{on } \Gamma_\phi, \quad (3.1c)$$

$$\nu D\Delta u + (1 - \nu)D\mathbf{n} \cdot (\nabla \nabla u)\mathbf{n} = \tilde{M} \quad \text{on } \Gamma_M, \quad (3.1d)$$

$$D(\nabla(\Delta u) + (1 - \nu)\boldsymbol{\Psi}(u)) \cdot \mathbf{n} = \tilde{Q} \quad \text{on } \Gamma_Q, \quad (3.1e)$$

$$[M_s(u)] = \tilde{S} \quad \text{on } \chi_N, \quad (3.1f)$$

where u represents the deflection of the plate, D its bending stiffness, ν is the Poisson's ratio and M_s denotes the twisting moment. Further, let us assume that $g \in L^2(\Omega)$ is the given load per unit area in the thickness direction, $\tilde{u} \in H^{3/2}(\Gamma_u)$, $\tilde{\phi} \in H^{1/2}(\Gamma_\phi)$, $\tilde{M} \in L^2(\Gamma_M)$ and $\tilde{Q} \in L^2(\Gamma_Q)$ are the prescribed deflection, rotation, bending moments and effective shear, respectively. Moreover, for every corner in χ_N , $\tilde{S} \in \mathbb{R}$ is the corresponding corner force, which is defined based on the associated applied twisting moment \tilde{M}_s as follows:

$$\tilde{S} = [[\tilde{M}_s]] \text{ where } [[\tilde{M}_s]] = \lim_{\varepsilon \rightarrow 0} \left(\tilde{M}_s(\mathbf{x} + \varepsilon \mathbf{t}) - \tilde{M}_s(\mathbf{x} - \varepsilon \mathbf{t}) \right). \quad (3.2)$$

3.2. The Kirchhoff-Love shell problem

Then, assuming an homogeneous and isotropic material, the bending stiffness D is defined as:

$$D = \frac{Et^3}{12(1 - \nu^2)}, \quad (3.3)$$

where t denotes the thickness of the plate, which without loss of generality we suppose to be a constant in Ω . Furthermore, the differential operator $\Psi(\cdot)$ reads:

$$\Psi(\cdot) = \left[\frac{\partial^3(\cdot)}{\partial x \partial^2 y}, \frac{\partial^3(\cdot)}{\partial^2 x \partial y} \right]^\top. \quad (3.4)$$

3.1.2 The weak formulation

Using standard variational calculus, the weak form of problem (3.1) reads: find $u \in V$ such that

$$a(u, v) = F(v) \quad \forall v \in V, \quad (3.5)$$

where $V \subset H^2(\Omega)$ denotes an infinite-dimensional space that depends in general on the boundary conditions of the problem at hand, for further details we refer to [Ciarlet, 2002]. Then, the bilinear form $a : V \times V \rightarrow \mathbb{R}$ can be expanded as follows:

$$a(u, v) = \int_{\Omega} D \left[(1 - \nu) \nabla(\nabla v) : \nabla(\nabla u) + \nu \Delta v \Delta u \right] d\Omega, \quad (3.6)$$

and similarly the linear functional $F : V \rightarrow \mathbb{R}$ reads:

$$F(v) = \int_{\Omega} g v d\Omega + \int_{\Gamma_M} \tilde{M} \frac{\partial v}{\partial \mathbf{n}} d\gamma + \int_{\Gamma_Q} \tilde{Q} v d\gamma + \sum_{e \in \chi_N} \left(\tilde{S} v \Big|_e \right). \quad (3.7)$$

It can be shown that (3.5) satisfies the conditions of the Lax-Milgram theorem, therefore $u \in V$ is the unique solution of the variational problem (3.5).

3.2 The Kirchhoff-Love shell problem

The first paper on a shell theory based on the kinematics assumptions of Kirchhoff can be traced back to [Love, 1888]. We highlight that these assumptions are generally valid for thin shells, where heuristically a structure is considered thin if its slenderness (defined as the ratio between the curvature radius and the thickness of the shell) is greater than 20. Similarly to the plate, here we review the Kirchhoff-Love formulation with a focus on its computer-based implementation.

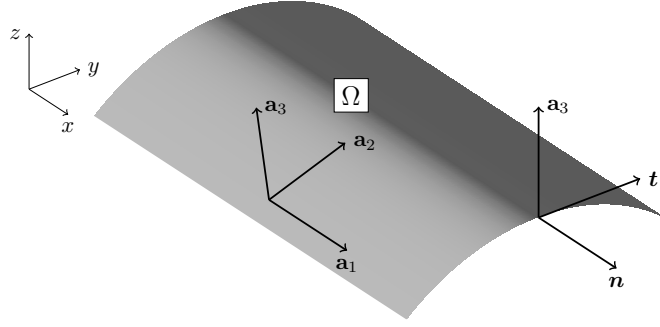


Figure 3.2 – Example of a shell mid-surface and its curvilinear coordinate frame.

3.2.1 Some fundamentals of differential geometry

Let us review some fundamentals of differential geometry needed to describe the Kirchhoff-Love equations defined on a manifold, following closely the notation in [Benzaken et al., 2021]. Let us consider the mid-surface $\Omega \in \mathbb{R}^3$ of our shell structure, which represents a manifold immersed in Euclidean space, and its Lipschitz-continuous boundary $\partial\Omega$. Recalling the spline geometric mapping of the mid surface $\mathbf{F} : \hat{\Omega} \rightarrow \Omega$, the in-plane covariant basis vectors \mathbf{a}_α are defined as follows:

$$\mathbf{a}_\alpha(\xi, \eta) = \mathbf{F}_{,\alpha}(\xi, \eta), \quad (3.8)$$

where the comma is used to indicate differentiation with respect to the corresponding curvilinear coordinate. Hereafter, we also use the convention that Greek indices (e.g. α, β, γ) take the values $[1, 2]$ while Latin indices (e.g. i, j, k) represent the values $[1, 2, 3]$. Now, the unit normal vector to the mid surface of the shell \mathbf{a}_3 is computed as the normalized cross-product of the in-plane vectors \mathbf{a}_α :

$$\mathbf{a}_3 = \frac{\mathbf{a}_1 \times \mathbf{a}_2}{\|\mathbf{a}_1 \times \mathbf{a}_2\|}, \quad (3.9)$$

where an example is depicted in Figure 3.2. Then, let us introduce the covariant metric coefficients as:

$$a_{\alpha\beta} = \mathbf{a}_\alpha \cdot \mathbf{a}_\beta, \quad (3.10)$$

where it is straightforward to verify the symmetry of a , i.e. $a_{\alpha\beta} = a_{\beta\alpha}$. Now, the contravariant basis vectors are defined via the following algebraic relationship:

$$\mathbf{a}^\alpha \cdot \mathbf{a}_\beta = \delta^\alpha_\beta, \quad (3.11)$$

3.2. The Kirchhoff-Love shell problem

where the corresponding covariant and contravariant metric coefficients are linked by the inverse operator:

$$\left[a^{\alpha\beta} \right] = \left[a_{\alpha\beta} \right]^{-1}. \quad (3.12)$$

With these coefficients the contravariant basis vectors can be obtained as:

$$\mathbf{a}^\alpha = a^{\alpha\gamma} \mathbf{a}_\gamma. \quad (3.13)$$

We now define the in-plane normal vector $\mathbf{n} = n^\alpha \mathbf{a}_\alpha$ to the boundary $\partial\Omega$, where n^α denotes its contravariant components. We highlight that \mathbf{n} is contained in the tangent plane to the shell. Similarly, we can define the tangent vector $\mathbf{t} = t^\alpha \mathbf{a}_\alpha$ where it holds $\mathbf{n} = \mathbf{t} \times \mathbf{a}_3$, see Figure 3.2 for an example. Then, we denote by b the second fundamental form of the surface, whose components are given as:

$$b_{\alpha\beta} = \mathbf{a}_3 \cdot \mathbf{a}_{\alpha,\beta}, \quad (3.14)$$

where

$$\mathbf{a}_{\alpha,\beta} = \mathbf{F}_{,\alpha\beta}, \quad (3.15)$$

is well-defined due to the smoothness assumption on the geometric mapping. Furthermore, c indicates the third fundamental form of the manifold and reads:

$$c_{\alpha\beta} = b_\alpha^\gamma b_{\gamma\beta}, \quad (3.16)$$

where the index of the curvature tensor can be raised via the following formula:

$$b_\alpha^\beta = a^{\alpha\gamma} b_{\gamma\beta}. \quad (3.17)$$

Under our assumptions on the differentiability of \mathbf{F} , it is easy to verify that also the second and third fundamental forms of the surface are symmetric. Then, we define the Christoffel symbols of the second kind:

$$\Gamma_{\alpha\beta}^\gamma = \mathbf{a}^\gamma \cdot \mathbf{a}_{\alpha,\beta}, \quad (3.18)$$

which express the coefficients of the Levi-Civita connection in a coordinate basis associated with the manifold. This quantity is a pseudo third-order tensor whose lower indices are symmetric, i.e. $\Gamma_{\alpha\beta}^\gamma = \Gamma_{\beta\alpha}^\gamma$, due to the torsion-free property of the Levi-Civita connection. In the following derivation, we will also make use of the first derivatives of the Christoffel symbols, which read:

$$\Gamma_{\alpha\beta,\mu}^\lambda = -\Gamma_{\nu\mu}^\lambda \Gamma_{\alpha\beta}^\nu + b_{\alpha\beta} b_\mu^\lambda + \mathbf{a}^\lambda \cdot \mathbf{a}_{\alpha,\beta\mu}. \quad (3.19)$$

Finally, we define the transformation which maps Cartesian components to curvilinear ones as:

$$\mathcal{Q}_\beta^i = \mathbf{e}^i \cdot \mathbf{a}_\beta \quad \text{and} \quad \mathcal{Q}_3^i = \mathbf{e}^i \cdot \mathbf{a}_3, \quad (3.20)$$

where \mathbf{e}^i represent the standard Euclidean basis.

3.2.2 A review of continuum mechanics

In order to describe quantities of interest in the full three-dimensional body, the definitions presented above must be suitably extended in the thickness direction. This can be readily achieved as follows:

$$\mathbf{X}(\xi, \eta, \zeta) = \mathbf{F}(\xi, \eta) + \zeta \mathbf{a}_3(\xi, \eta), \quad (3.21)$$

which provides a parametrized representation of the shell. In the following, this represents the so-called *reference* or *undeformed* configuration of the continuum. The out-of-plane coordinate ζ in the thickness direction takes values in the range $[-t/2, t/2]$, where we recall that t denotes the thickness of the structure. Consequently, the curvilinear coordinates frame attached to the shell reads:

$$\mathbf{G}_\alpha = \mathbf{X}_{,\alpha} = \left(\delta_\beta^\alpha - \zeta b_\alpha^\beta \right) \mathbf{a}_\beta \quad \text{and} \quad \mathbf{G}_3 = \mathbf{X}_{,3} = \mathbf{a}_3. \quad (3.22)$$

At this point, we introduce the *actual* or *deformed* configuration of the body and we denote it by \mathcal{X} . Furthermore, we assume there exists a smooth bijective function which maps the reference configuration into the deformed one. An important quantity in mechanics, the deformation gradient, is defined as the tensor describing the latter mapping and reads:

$$\mathcal{F} := \frac{\partial \mathcal{X}}{\partial \mathbf{X}} = \frac{\partial \mathcal{X}}{\partial \xi^i} \frac{\xi^i}{\partial \mathbf{X}} = \mathbf{G}_i \otimes \mathbf{G}^i, \quad (3.23)$$

where \mathbf{G} denotes the curvilinear coordinate frame in the deformed configuration. In our derivation we employ the Green-Lagrange strain tensor, which is given by:

$$\mathbf{E} = \frac{1}{2} \left((\mathbf{G}^i \otimes \mathbf{G}_i) \cdot (\mathbf{G}_j \otimes \mathbf{G}^j) - \mathbf{I} \right) = \frac{1}{2} (G_{ij} - G_{ij}) = E_{ij} \mathbf{G}^i \otimes \mathbf{G}^j, \quad (3.24)$$

where \mathbf{I} represents the second order identity tensor and G_{ij} and G_{ij} denote the covariant metric coefficients in the current and reference configurations, respectively. Let us explicitly state the dependency of the coefficients E_{ij} on the full displacement field \mathbf{U} . Indeed, we can write:

$$\mathbf{G}_i = \frac{\partial \mathcal{X}}{\partial \xi^i} = \frac{(\mathbf{X} + \mathbf{U})}{\partial \xi^i} = \mathbf{G}_i + \mathbf{U}_{,i}. \quad (3.25)$$

3.2. The Kirchhoff-Love shell problem

By substituting this result into Equation (3.24), we obtain the following definition of the components of the Green-Lagrange strain tensor:

$$E_{ij} = \frac{1}{2} (\mathbf{U}_{,i} \cdot \mathbf{G}_j + \mathbf{U}_{,j} \cdot \mathbf{G}_i + \mathbf{U}_{,i} \cdot \mathbf{U}_{,j}) . \quad (3.26)$$

In the scope of this work we use the linearized version of Equation (3.26), where the higher order displacement contribution is neglected. Therefore, Equation (3.26) can be simplified to:

$$\varepsilon_{ij} = \frac{1}{2} (\mathbf{U}_{,i} \cdot \mathbf{G}_j + \mathbf{U}_{,j} \cdot \mathbf{G}_i) , \quad (3.27)$$

which describes the components of the standard strain tensor of linear elasticity introduced in Equation (2.16), associated with a curvilinear reference frame. We are now ready to introduce the Kirchhoff-Love assumptions on the kinematics of the shell. In particular, the normals to the structure should remain straight and normal during deformation. Consequently, transversal shear strains are neglected. In particular, let us describe the displacement field by quantity defined on the mid-surface as:

$$\mathbf{U}(\xi, \eta, \zeta) = \mathbf{u}(\xi, \eta) + \zeta \theta(\mathbf{u}(\xi, \eta)) , \quad (3.28)$$

where θ represents the rotation of the normal vector \mathbf{a}_3 . By employing the Kirchhoff-Love assumption that transverse shearing is disregarded ab-initio, we can introduce the following constraint on the rotation field:

$$\theta(\mathbf{u}) = -\mathbf{a}_3 \cdot \nabla^*(\mathbf{u}) , \quad (3.29)$$

where $\nabla^*(-)$ represents the surface gradient, defined as [Brand, 1947]:

$$\nabla^*(-) = \frac{\partial(-)}{\partial \xi^\alpha} \otimes \mathbf{a}^\alpha . \quad (3.30)$$

Similarly, we introduce the surface divergence operator $\nabla^* \cdot (-)$:

$$\nabla^* \cdot (-) = \frac{\partial(-)}{\partial \xi^\alpha} \cdot \mathbf{a}^\alpha . \quad (3.31)$$

With these definitions, we can define the strain operators related to the membrane and bending action, respectively, as:

$$\alpha(\mathbf{u}) = P \cdot \text{sym}(\nabla^*(\mathbf{u})) \cdot P , \quad (3.32)$$

and

$$\beta(\mathbf{u}) = -P \cdot \text{sym}(\mathbf{a}_3 \cdot \nabla^* \nabla^*(\mathbf{u})) \cdot P , \quad (3.33)$$

where $P = \mathbf{I} - \mathbf{a}_3 \otimes \mathbf{a}_3$ denotes the in-plane projector and the operator $\text{sym}(\cdot)$ returns the symmetric part of the input tensor. Consequently, the linearized Green-Lagrange strain tensor defined in Equation (3.27) can be decomposed into a membrane and bending part, respectively, as:

$$\varepsilon(\mathbf{u}) = \alpha(\mathbf{u}) + \zeta \beta(\mathbf{u}). \quad (3.34)$$

Similarly to linear elasticity, let us now introduce a constitutive law between strains and stresses. In particular, we can write the energetically conjugate stress measure to the Green-Lagrange strain, the Cauchy stress tensor, via:

$$\sigma(\mathbf{u}) = \mathbb{C} : \alpha(\mathbf{u}) + \zeta \mathbb{C} : \beta(\mathbf{u}), \quad (3.35)$$

where the fourth-order tensor \mathbb{C} for homogeneous materials can be expressed in curvilinear coordinates as:

$$\mathbb{C} = \mathbb{C}^{\alpha\beta\lambda\mu} \mathbf{a}_\alpha \otimes \mathbf{a}_\beta \otimes \mathbf{a}_\lambda \otimes \mathbf{a}_\mu \text{ with } \mathbb{C}^{\alpha\beta\lambda\mu} = \frac{E}{2(1+v)} \left(a^{\alpha\lambda} a^{\beta\mu} + a^{\alpha\mu} a^{\beta\lambda} + \frac{2v}{1-v} a^{\alpha\beta} a^{\lambda\mu} \right). \quad (3.36)$$

Now, if we analytically integrate through the thickness, we can define the membrane stress resultant:

$$A(\mathbf{u}) = \int_{-t/2}^{t/2} \sigma(\mathbf{u}) d\zeta = t \mathbb{C} : \alpha(\mathbf{u}), \quad (3.37)$$

and the bending stress resultant:

$$B(\mathbf{u}) = \int_{-t/2}^{t/2} \zeta \sigma(\mathbf{u}) d\zeta = \frac{t^3}{12} \mathbb{C} : \beta(\mathbf{u}), \quad (3.38)$$

respectively. This results follow from our assumption of a constant thickness and a linear stress distribution through the thickness of the shell. Lastly, let us introduce the bending moment $B_{nn}(\mathbf{u}) = \mathbf{n} \cdot B(\mathbf{u}) \cdot \mathbf{n}$ and the twisting moment $B_{nt}(\mathbf{u}) = \mathbf{n} \cdot B(\mathbf{u}) \cdot \mathbf{t}$.

3.2.3 The weak formulation

Let us now introduce the weak formulation of the Kirchhoff-Love problem. First, let us define the space:

$$\mathcal{V} = \{\mathbf{v} = v_\alpha \mathbf{a}^\alpha + v_3 \mathbf{a}^3 \mid v_\alpha \in [H^1(\Omega)]^2, v_3 \in H^2(\Omega)\}. \quad (3.39)$$

Analogously to the case of plates, let us split the boundary $\Gamma = \partial\Omega$ into a part associated with Dirichlet-type boundary conditions $\Gamma_D = \Gamma_{\mathbf{u}} \cup \Gamma_\theta$ and a part corresponding to Neumann-type boundary conditions $\Gamma_N = \Gamma_{\mathbf{T}} \cup \Gamma_{B_{nn}}$ such that $\Gamma = \overline{\Gamma_D} \cup \overline{\Gamma_N}$. It also

3.2. The Kirchhoff-Love shell problem

holds that $\Gamma_{\mathbf{u}} \cap \Gamma_{\mathbf{T}} = \emptyset$ and $\Gamma_{\theta} \cap \Gamma_{B_{nn}} = \emptyset$ due to the energetically conjugate nature of applied displacements and transverse shear, and applied rotations and bending moments, respectively. Let us also assume an applied body force $\tilde{\mathbf{f}} = (\tilde{f}_\alpha \mathbf{a}^\alpha + \tilde{f}_3 \mathbf{a}^3) \in [L^2(\Omega)]^d$, a prescribed bending moment $\tilde{B}_{nn} \in L^2(\Gamma_{B_{nn}})$, a prescribed Ersatz force $\tilde{\mathbf{T}} \in L^2(\Gamma_{\mathbf{T}})$ and a given corner force $\tilde{S} \in \mathbb{R}$ for all corners in χ_N . We recall that the corner forces are linked to the jump of the prescribed twisting moments \tilde{B}_{nt} as follows:

$$\tilde{S} = \llbracket \tilde{B}_{nt} \rrbracket \text{ where } \llbracket \tilde{B}_{nt} \rrbracket = \lim_{\varepsilon \rightarrow 0} \left(\tilde{B}_{nt}(\mathbf{x} + \varepsilon \mathbf{t}) - \tilde{B}_{nt}(\mathbf{x} - \varepsilon \mathbf{t}) \right), \quad (3.40)$$

where \mathbf{x} denotes the location of the corresponding corner. By leveraging the notation previously introduced and variational calculus, the weak formulation of the Kirchhoff-Love shell reads, find $\mathbf{u} \in V$ such that:

$$a(\mathbf{u}, \mathbf{v}) = f(\mathbf{v}) \quad \forall \mathbf{v} \in V, \quad (3.41)$$

where the choice of infinite-dimensional space $V \subset \mathcal{V}$ depends in general on the boundary conditions of the problem at hand. The bilinear form a can be expanded as follows:

$$a(\mathbf{u}, \mathbf{v}) = \int_{\Omega} A(\mathbf{u}) : \alpha(\mathbf{v}) \, d\Omega + \int_{\Omega} B(\mathbf{u}) : \beta(\mathbf{v}) \, d\Omega. \quad (3.42)$$

Similarly, the linear functional f reads:

$$f(\mathbf{v}) = \int_{\Omega} \tilde{\mathbf{f}} \cdot \mathbf{v} \, d\Omega + \int_{\Gamma_{\mathbf{T}}} \tilde{\mathbf{T}} \cdot \mathbf{v} \, d\Gamma + \int_{\Gamma_{B_{nn}}} \tilde{B}_{nn} \theta_n(\mathbf{v}) \, d\Gamma + \sum_{e \in \chi_N} \left(\tilde{S} v_3 \Big|_e \right), \quad (3.43)$$

where θ_n denotes the normal rotation.

3.2.4 The strong formulation

For the sake of completeness, let us also introduce the strong form of the Kirchhoff-Love problem. Given a prescribed displacement $\tilde{\mathbf{u}} = \tilde{u}_\alpha \mathbf{a}^\alpha + \tilde{u}_3 \mathbf{a}^3$ such that $\tilde{u}_\alpha \in H^{\frac{1}{2}}(\Gamma_{\mathbf{u}})$ and $\tilde{u}_3 \in H^{\frac{3}{2}}(\Gamma_{\mathbf{u}})$ and a prescribed normal rotation $\tilde{\theta}_n \in H^{\frac{1}{2}}(\Gamma_{\theta})$, we seek the displacement field $\mathbf{u} : \bar{\Omega} \rightarrow \mathbb{R}^3$ such that:

$$P \cdot [\nabla^\star \cdot (b \cdot B(\mathbf{u})) \cdot b - \nabla^\star \cdot A(\mathbf{u})] = \tilde{f}_\alpha \quad \text{in } \Omega \quad (3.44a)$$

$$B(\mathbf{u}) : c - \nabla^\star \cdot (P \cdot (\nabla^\star \cdot B(\mathbf{u}))) - A(\mathbf{u}) : b = \tilde{f}_3 \quad \text{in } \Omega \quad (3.44b)$$

$$\mathbf{u} = \tilde{\mathbf{u}} \quad \text{on } \Gamma_{\mathbf{u}} \quad (3.44c)$$

$$\theta_n(\mathbf{u}) = \tilde{\theta}_n \quad \text{on } \Gamma_{\theta} \quad (3.44d)$$

$$\mathbf{T}(\mathbf{u}) = \tilde{\mathbf{T}} \quad \text{on } \Gamma_{\mathbf{T}} \quad (3.44e)$$

$$B_{nn}(\mathbf{u}) = \tilde{B}_{nn} \quad \text{on } \Gamma_{B_{nn}} \quad (3.44f)$$

$$\llbracket B_{nt}(\mathbf{u}) \rrbracket = \tilde{S} \quad \text{on } \chi_N, \quad (3.44g)$$

where b and c denote the second and third differential forms given in Equations (3.14) and (3.16), respectively. We recall that the Ersatz force is defined as [Benzaken et al., 2021]:

$$\mathbf{T}(\mathbf{u}) = A(\mathbf{u}) \cdot \mathbf{n} - b \cdot \left(B(\mathbf{u}) \cdot \mathbf{n} + \mathbf{t} B_{nt}(\mathbf{u}) \right) + \left(\left(\nabla^* \cdot B(\mathbf{u}) \right) \cdot \mathbf{n} + \frac{\partial B_{nt}(\mathbf{u})}{\partial \mathbf{t}} \right) \mathbf{a}_3. \quad (3.45)$$

3.2.5 Extension to laminate composites

Let us consider a thin composite shell defined as a sequence of orthotropic plies, which can be described by the classical theory of laminated plates [Reddy, 1999; Kiendl, 2011]. Consequently, the bilinear form in (3.41) must be modified as explained in the following. Let us consider a stacking of plies, numbered by an index $n = 1, \dots, P$, where P denotes the total number of plies. For each ply we can define the material tensor \mathbb{C}_n , obtained by transforming the corresponding orthotropic ply tensor from the local ply coordinates to the shell curvilinear reference frame, as:

$$\mathbb{C}_n = \mathcal{T}^\top(\phi) \mathbb{C}_n^{\text{ort}} \mathcal{T}(\phi), \quad (3.46)$$

where $\mathbb{C}_n^{\text{ort}}$ depends on the Young's moduli E_1 and E_2 , the corresponding Poisson's ratios ν_{12} and ν_{21} and the shear modulus G_{12} . Namely, it can be expressed in matrix form as:

$$\mathbb{C}_n^{\text{ort}} = \begin{bmatrix} \frac{E_1}{(1 - \nu_{12}\nu_{21})} & \frac{\nu_{21}E_1}{(1 - \nu_{12}\nu_{21})} & 0 \\ \frac{\nu_{12}E_2}{(1 - \nu_{12}\nu_{21})} & \frac{E_1}{(1 - \nu_{12}\nu_{21})} & 0 \\ 0 & 0 & G_{12} \end{bmatrix}, \quad (3.47)$$

where it holds $\nu_{21}E_1 = \nu_{12}E_2$ due to the symmetry of the resulting material tensor. Furthermore, the transformation matrix $\mathcal{T}(\phi)$ depends on the fiber angle ϕ and reads:

$$\mathcal{T}(\phi) = \begin{bmatrix} \cos^2(\phi) & \sin^2(\phi) & \sin(\phi) \cos(\phi) \\ \sin^2(\phi) & \cos^2(\phi) & -\sin(\phi) \cos(\phi) \\ -2 \sin(\phi) \cos(\phi) & 2 \sin(\phi) \cos(\phi) & \cos^2(\phi) - \sin^2(\phi) \end{bmatrix}, \quad (3.48)$$

which maps quantities defined in the ply local coordinate system to the shell reference frame, see Figure 3.3 for an example. Now, following the classical theory of laminates, the homogenized extensional stiffness \mathbb{A} , the coupling stiffness \mathbb{B} and the bending stiffness

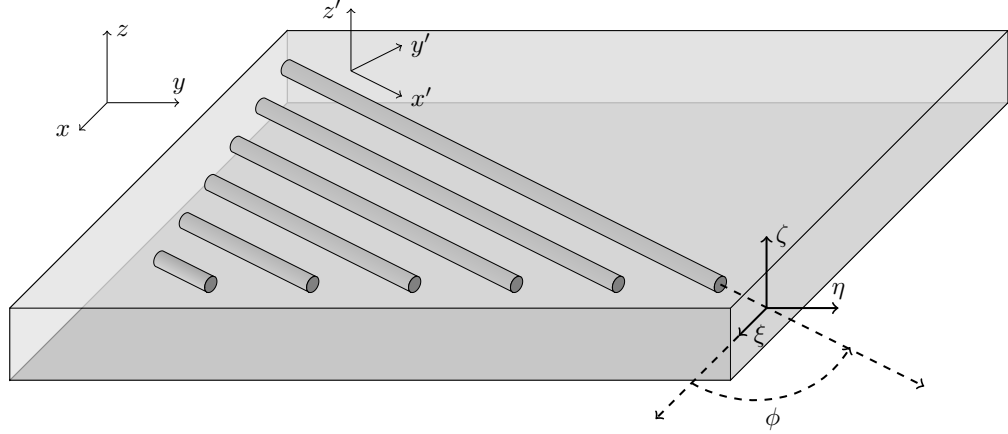


Figure 3.3 – Example of a laminate with its corresponding ply, shell and Cartesian coordinate systems, respectively.

\mathbb{D} are computed, respectively, as:

$$\begin{aligned} \mathbb{A} &= \int_{-t/2}^{t/2} \mathbb{C} d\zeta = \sum_{n=1}^P \mathbb{C}_n t_n, \\ \mathbb{B} &= \int_{-t/2}^{t/2} \zeta \mathbb{C} d\zeta = \sum_{n=1}^P \mathbb{C}_n t_n z_n, \\ \mathbb{D} &= \int_{-t/2}^{t/2} \zeta^2 \mathbb{C} d\zeta = \sum_{n=1}^P \mathbb{C}_n \left(t_n z_n^2 + \frac{t_n^3}{12} \right), \end{aligned} \quad (3.49)$$

where t_n indicates the thickness of the n -th ply and z_n denotes the distance between the centroid of the n -th ply and the mid-plane of the shell, where an example is depicted in Figure 3.4. Then, the bilinear form associated with a thin laminate shell reads:

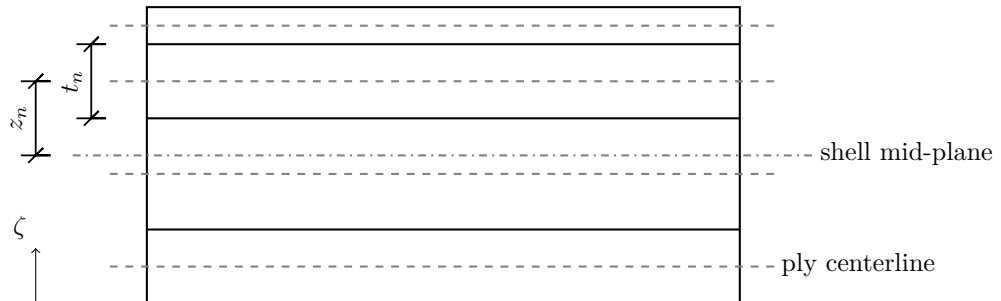


Figure 3.4 – Example of a laminate along the thickness direction ζ formed by a non-uniform and non-symmetric ply sequence.

$$\begin{aligned} a(\mathbf{u}, \mathbf{v}) = & \int_{\Omega} \left(\mathbb{A} : \alpha(\mathbf{u}) + \mathbb{B} : \beta(\mathbf{u}) \right) : \alpha(\mathbf{v}) d\Omega \\ & + \int_{\Omega} \left(\mathbb{B} : \alpha(\mathbf{u}) + \mathbb{D} : \beta(\mathbf{u}) \right) : \beta(\mathbf{v}) d\Omega, \end{aligned} \quad (3.50)$$

where for further details we refer to [Kiendl, 2011] and references therein.

3.2.6 Implementational details

The derivation presented thus far employs tensorial quantities. In order to implement the latter within a computer routine, we need to explicitly express the components of these quantities. In the notation presented below, we follow closely the derivation in [Benzaken et al., 2021, Appendix C]. Recalling the transformation map in Equation (3.20), the components of the membrane strain read:

$$\alpha_{\beta\gamma} = \frac{1}{2} \left[\mathcal{Q}_{\beta}^i u_{i,\gamma} + \mathcal{Q}_{\gamma}^i u_{i,\beta} \right], \quad (3.51)$$

where we have expressed the displacement field in the Cartesian reference frame as $\mathbf{u} = u_i \mathbf{e}^i$. Using the constitutive relationship, the components of the membrane stress can be written as:

$$A^{\alpha\beta}(\mathbf{u}) = \mathbb{C}^{\alpha\beta\gamma\delta} \alpha_{\gamma\delta}(\mathbf{u}). \quad (3.52)$$

Similarly, the component of the bending strain are given as:

$$\beta_{\alpha\gamma} = -\frac{1}{2} \mathcal{Q}_3^i \left[u_{i|\alpha\gamma} + u_{i|\gamma\alpha} \right], \quad (3.53)$$

where we make use of the following definition of the second covariant derivatives of the displacements:

$$u_{i|\alpha\beta} = u_{i,\alpha\beta} - \Gamma_{\alpha\beta}^{\gamma} u_{i,\gamma}, \quad (3.54)$$

to compact the notation. The components of the bending stress can be again obtained from the constitutive law:

$$B^{\alpha\beta}(\mathbf{u}) = \mathbb{C}^{\alpha\beta\gamma\delta} \beta_{\gamma\delta}(\mathbf{u}). \quad (3.55)$$

Then, given the rotation vector $\theta(\mathbf{u}) = -\mathcal{Q}_3^i u_{i,\alpha} \mathbf{a}^{\alpha}$, the normal and twisting rotations can be expressed as:

$$\theta_n(\mathbf{u}) = -\mathcal{Q}_3^i u_{i,\alpha} n^{\alpha} \quad \text{and} \quad \theta_t(\mathbf{u}) = -\mathcal{Q}_3^i u_{i,\alpha} t^{\alpha}, \quad (3.56)$$

respectively. We recall that here n^{α} and t^{α} denote the contravariant components of the normal and tangent vectors, respectively. Analogously, the bending and twisting

3.2. The Kirchhoff-Love shell problem

moments read:

$$B_{nn}(\mathbf{u}) = n_\alpha n_\beta B^{\alpha\beta}(\mathbf{u}) \quad \text{and} \quad B_{nt}(\mathbf{u}) = n_\alpha t_\beta B^{\alpha\beta}(\mathbf{u}), \quad (3.57)$$

respectively. Finally, we introduce the components of the Ersatz force, which can be split into its in-plane part:

$$\mathbf{T}^\alpha(\mathbf{u}) = A^{\alpha\beta}(\mathbf{u})n_\beta - b_{\cdot\lambda}^\alpha B^{\lambda\beta}(\mathbf{u})n_\beta - B_{nt}(\mathbf{u})b_{\cdot\lambda}^\alpha t^\lambda, \quad (3.58)$$

and its out-of-plane components:

$$\mathbf{T}^3(\mathbf{u}) = n_\alpha B_{\cdot\cdot\beta}^{\alpha\beta}(\mathbf{u}) + n_{\alpha|\lambda} B^{\alpha\beta}(\mathbf{u})t_\beta t^\lambda + n_\alpha B_{\cdot\cdot|\lambda}^{\alpha\beta}(\mathbf{u})t_\beta t^\lambda + n_\alpha B^{\alpha\beta}(\mathbf{u})t_{\beta|\lambda} t^\lambda, \quad (3.59)$$

respectively, where we have introduced the \cdot notation for tensors to specify the correct index ordering. Furthermore, the covariant derivative of the bending stress reads:

$$B_{\cdot\cdot|\nu}^{\alpha\beta}(\mathbf{u}) = \mathbb{C}^{\alpha\beta\lambda\mu} \beta_{\lambda\mu|\nu}(\mathbf{u}), \quad (3.60)$$

where the covariant derivative of the bending strain is given by:

$$\beta_{\alpha\beta|\nu}(\mathbf{u}) = \frac{1}{2} \left[\Lambda_\lambda^i \left(b_{\cdot\nu}^\lambda u_{i|\alpha\beta} + b_{\cdot\nu}^\lambda u_{i|\beta\alpha} \right) - \Lambda_3^i \left(u_{i|\alpha\beta\mu} + u_{i|\beta\alpha\mu} \right) \right]. \quad (3.61)$$

Lastly, the third-order covariant derivative of the displacements can be written as:

$$u_{i|\alpha\beta\mu}(\mathbf{u}) = u_{i,\alpha\beta\mu} - u_{i,\lambda} \Gamma_{\alpha\beta,\mu}^\lambda - \Gamma_{\alpha\beta}^\lambda u_{i,\lambda\mu} - \Gamma_{\alpha\mu}^\lambda u_{i|\lambda\beta} - \Gamma_{\beta\mu}^\lambda u_{i|\alpha\lambda}, \quad (3.62)$$

where the derivatives of the Christoffel symbols are given in Equation (3.19). With these definitions at hand, the Kirchhoff-Love shell problem can be readily implemented into a finite element code.

4 A review of spline technologies in CAD and analysis

In this chapter, the main features of B-splines and their role in isogeometric analysis are summarized, following closely the derivation in [Höllig, 2003; Cottrell et al., 2009]. We recall that B-splines are ubiquitous in Computer Aided Design (CAD), where the interested reader is referred to the seminal works [de Boor, 1978; Piegl and Tiller, 1995; Rogers, 2001]. We also introduce the basic concepts underlying the use of B-splines in CAD, namely the Boundary Representation (B-Rep) of objects and trimming. Moreover, we address another issue of B-splines which concerns the local refinement capabilities of the basis. We remark that this limitation stems from the tensor-product nature of classical B-splines. To overcome this and building upon standard splines, two variants thereof that allow for local refinement are introduced, namely \mathcal{HB} - and \mathcal{THB} -splines, respectively, where we follow the derivation in [Kraft, 1997; Vuong et al., 2011; Giannelli et al., 2012, 2013]. We then extend their definition to trimmed surfaces. Lastly, we show the advantages of employing spline functions for the Galerkin discretization of the Kirchhoff plate and Kirchhoff-Love shell problems, respectively. In particular, we perform several numerical experiments to verify the benefits of local refinement for the analysis of complex trimmed geometries with details at different scales.

4.1 An introduction to B-splines

Starting from two integers $p, n \in \mathbb{N}$, let us introduce a non-decreasing sequence of real values referred to as *knot vector*, denoted in the following as $\Xi = \{\xi_1, \dots, \xi_{n+p+1}\}$. In the latter, $\xi_i \in \mathbb{R}, i = 1, \dots, n + p + 1$ represent the knots which partition the parameter space of a patch into elements. Let us also introduce the concept of multiplicity of a knot, which expresses whether a knot is repeated in the knot vector. We highlight that the multiplicity is directly linked to the continuity achievable within one patch.

Remark 4.1 *Following the common practice in CAD design, in this thesis we only consider so-called open knot vectors, where the first and last knots are repeated $p+1$ times.*

In 1D, this entails that spline functions formed by open knot vectors are interpolatory at the beginning and at the end of their corresponding parameter space.

4.1.1 B-spline basis functions in one dimension

Starting from a knot vector Ξ and a degree p , the corresponding univariate B-spline basis function $b_{i,p}(\xi)$ are defined recursively via the Cox-de Boor formula [de Boor, 1977] starting from piecewise constants:

$$b_{i,0}(\xi) = \begin{cases} 1 & \text{if } \xi_i \leq \xi < \xi_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (4.1)$$

For $p > 0$, they read as follows:

$$b_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} b_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} b_{i+1,p-1}(\xi). \quad (4.2)$$

We remark that for $p = 0, 1$ this formula provides the same piecewise constant and linear polynomials, respectively, as those used in standard finite elements. Let us summarize here the most important properties of the B-spline basis:

- They satisfy the partition of unity, i.e.

$$\sum_{i=1}^{n+p+1} b_{i,p}(\xi) = 1 \quad \forall \xi \in [\xi_1, \xi_{n+p+1}]. \quad (4.3)$$

- The smoothness of each B-spline basis is C^{p-k} at every knot, where k denotes the multiplicity of the considered knot, while it is C^∞ elsewhere. This feature has paramount implications in the discretization of higher-order PDEs, e.g. Kirchhoff plates and Kirchhoff-Love shells.
- Each basis function is non-negative within one patch, i.e., $b_{i,p}(\xi) \geq 0, \forall \xi \in [\xi_1, \xi_{n+p+1}]$.
- Each basis function has local support, i.e. over $p + 1$ knot spans.

To exemplify these concepts, the quadratic B-spline basis associated with a one-dimensional non-uniform open vector $\Xi = [0 \ 0 \ 0 \ 0.2 \ 0.2 \ 0.4 \ 0.6 \ 0.8 \ 1 \ 1 \ 1]$ is depicted in Figure 4.1.

4.1.2 Non-Uniform Rational B-splines (NURBS) in one dimension

Although B-splines allow for an easy description of many free-form shapes, a wide variety of important objects in engineering, such as conical sections, cannot be exactly

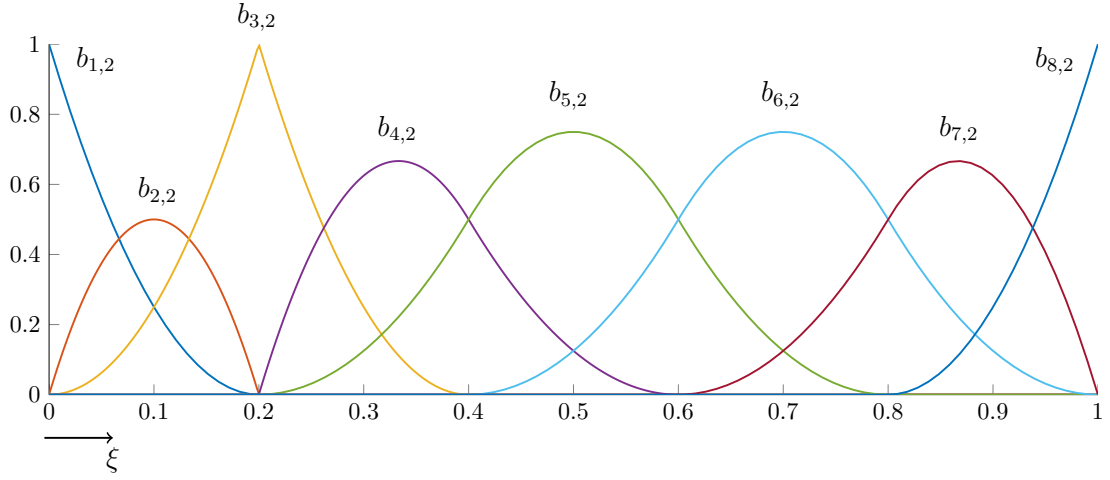


Figure 4.1 – Example of a quadratic B-spline basis associated with the knot vector $\Xi = [0 \ 0 \ 0 \ 0.2 \ 0.2 \ 0.4 \ 0.6 \ 0.8 \ 1 \ 1 \ 1]$. Note the reduction of continuity at the repeated knot.

represented. To overcome this issue, let us introduce the univariate NURBS basis function $r_{i,p}(\xi)$ as follows:

$$r_{i,p}(\xi) = \frac{b_{i,p}(\xi)w_i}{\sum_j b_{j,p}(\xi)w_j}, \quad (4.4)$$

where w_i represent the weights associated with the rational function. We highlight that NURBS inherit must of the favorable properties of their building blocks, i.e. B-splines. Furthermore, it is straightforward to verify that if all weights are the same, the corresponding NURBS is equivalent to the underlying B-spline.

Remark 4.2 *From a geometrical standpoint, NURBS entities in \mathbb{R}^d are often characterized by employing a projective transformation of B-splines defined in \mathbb{R}^{d+1} . Although this construction provides a direct interpretation of NURBS and their (projection) weights, we prefer the more abstract definition of basis functions given in Equation (4.4), which fits into the general FEM framework.*

4.1.3 B-spline basis functions in multiple dimensions

The definition of multivariate B-splines $\mathcal{B}_{\mathbf{i},\mathbf{p}}(\boldsymbol{\eta})$ is achieved in a straight-forward manner using the tensor product of univariate B-splines as:

$$\mathcal{B}_{\mathbf{i},\mathbf{p}}(\boldsymbol{\eta}) = \prod_{j=1}^d b_{i_j,p_j}^j(\eta_j), \quad (4.5)$$

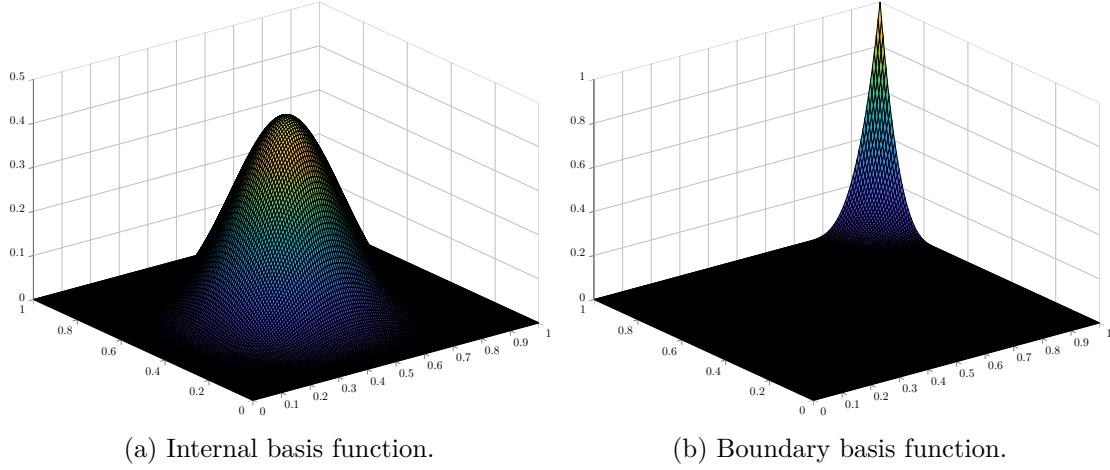


Figure 4.2 – Example of multivariate cubic B-splines defined in the parameter space of dimension $\hat{d} = 2$.

where \hat{d} denotes the dimension of the parameter space. Additionally, the multi-index $\mathbf{i} = \{i_1, \dots, i_{\hat{d}}\}$ denotes the position in the tensor product structure and $\mathbf{p} = \{p_1, \dots, p_{\hat{d}}\}$ indicates the vector of polynomial degrees, associated with the corresponding parametric dimension $\boldsymbol{\eta} = (\eta_1, \dots, \eta_{\hat{d}})$, respectively. Of particular relevance in this work is the case $\hat{d} = 2$ corresponding to the parametric dimensions $\boldsymbol{\eta} = (\eta_1, \eta_2) = (\xi, \eta)$, where an example of multivariate B-spline shape functions is provided in Figure 4.2. For the sake of clarity in our notation, we drop the multi-index \mathbf{i} unless explicitly needed. Furthermore, without loss of generality, we assume that the degree vector \mathbf{p} is identical in each parametric direction and therefore it can be simplified to a single scalar value p . The latter will be also omitted unless explicitly needed in our exposition. We are now ready to introduce the following discrete space:

$$X_h = \text{span} \left\{ b \circ \mathbf{F}^{-1} \mid b \in \mathcal{B} \right\},$$

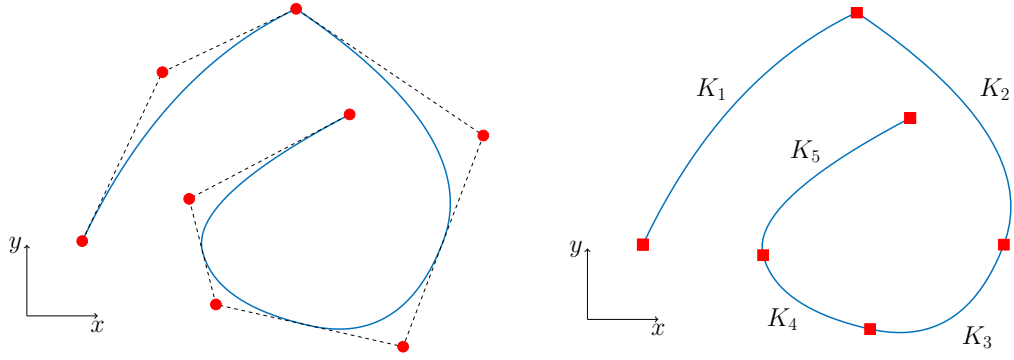
which can be exploited for the Galerkin approximation of PDEs.

4.1.4 B-spline curves

Once we have defined the basis functions, B-spline curves are constructed as the linear combination of the basis and their associated coefficients:

$$\mathbf{C}(\xi) = \sum_i b_{i,p}(\xi) \mathbf{P}_i, \quad (4.6)$$

where the $\mathbf{P}_i \in \mathbb{R}^d$ are commonly referred to as *control points*, and d represents the dimensionality of the physical space. An example of B-spline curve defined with the same knot vector used in Figure 4.1 is given in Figure 4.3, where we also plot the corresponding



(a) B-spline curve, control points and corresponding control polygon. (b) B-spline curve and corresponding mesh.

Figure 4.3 – Example of a B-spline curve in the physical space \mathbb{R}^2 associated with the knot vector $\Xi = [0 \ 0 \ 0 \ 0.2 \ 0.2 \ 0.4 \ 0.6 \ 0.8 \ 1 \ 1 \ 1]$.

control points, their control polygon and the associated isogeometric mesh. Furthermore, we highlight the non-interpolatory nature of the control points, which is a distinguishing feature compared to standard FEM nodes. We remark that many properties of the B-spline basis are directly inherited by B-spline curves.

4.1.5 B-spline surfaces

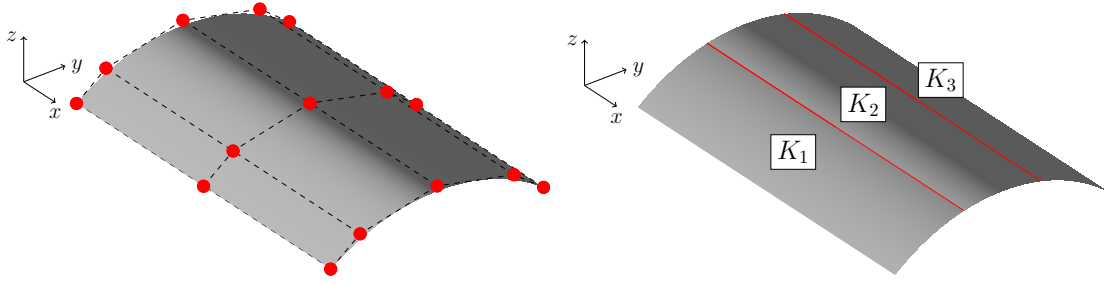
Given a bivariate B-spline basis $\mathcal{B}_{i,j}(\xi, \eta)$ (note that we have expanded the multi-index \mathbf{i} for clarity) and a net of control points $\mathbf{P}_{i,j} \in \mathbb{R}^d$, a B-spline surface is defined as the following linear combination:

$$\mathbf{S}(\xi, \eta) = \sum_i \sum_j \mathcal{B}_{i,j}(\xi, \eta) \mathbf{P}_{i,j}. \quad (4.7)$$

Many properties of B-spline surfaces follow directly from the basis and its tensor product nature. For instance, as we are using open knot vectors in each parametric direction, the vertices of the surface are interpolated. This is clear from the example given in Figure 4.4, where a surface of degree $p = 2$ associated with the knot vectors $\Xi = [0 \ 0 \ 0 \ 1/3 \ 2/3 \ 1 \ 1 \ 1]$ and $\Psi = [0 \ 0 \ 0 \ 1 \ 1 \ 1]$ in the parametric directions ξ and η , respectively, is depicted. For the sake of conciseness, we skip the extension of the notation to NURBS curves and surfaces; the interested reader is referred to [Cottrell et al., 2009] for a detailed derivation.

4.1.6 Refinement strategies

The successful application of splines in computations relies on the ability of enriching their approximation power without changing the underlying geometrical description. This can be achieved by three different refinement strategies, which we briefly summarize



(a) B-spline surface, control points net and corresponding control polygon. (b) B-spline surface and corresponding mesh.

Figure 4.4 – Example of a B-spline surface in the physical space \mathbb{R}^3 associated with the knot vectors $\Xi = [0 \ 0 \ 0 \ 1/3 \ 2/3 \ 1 \ 1 \ 1]$ and $\Psi = [0 \ 0 \ 0 \ 1 \ 1 \ 1]$.

in the following.

Knot insertion

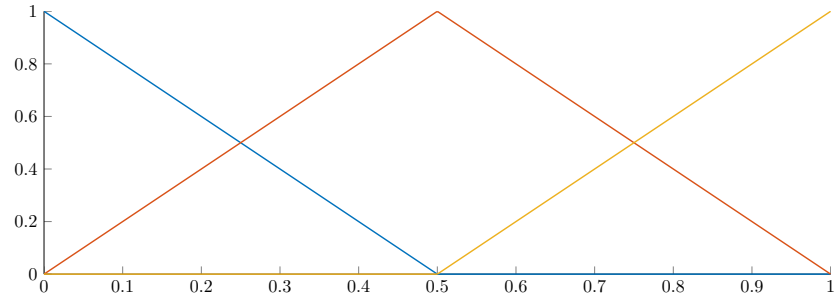
Given a knot vector Ξ , this strategy entails the construction of a new, augmented knot vector $\tilde{\Xi}$, such that $\Xi \subset \tilde{\Xi}$. In 1D, every new knot introduces a new basis function. Clearly, this strategy entails similarities with the classical h -refinement in FEM. However, it is worth noting that knots can also be inserted to reduce the continuity of the basis for specific applications, which shows the greater flexibility of B-splines compared to standard C^0 finite elements.

Degree elevation

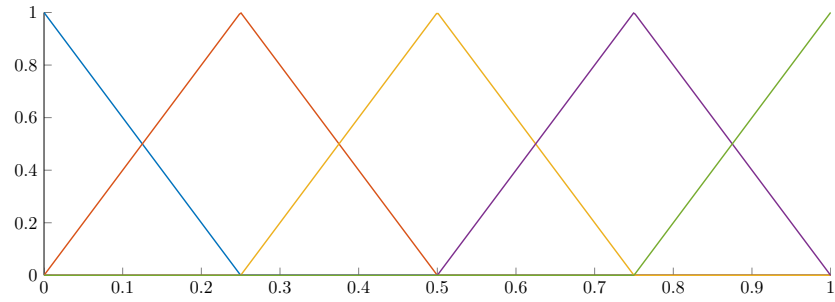
Analogously to p -refinement in FEM, this strategy aims at increasing the polynomial order of the basis. We remark that the operation of order elevation keeps the existing continuity of the underlying B-splines untouched. Consequently, the multiplicity of every knot is increased by one to preserve the original $p - m_i$ continuous derivatives. Again, the difference between p -refinement and degree elevation lies in the flexible inter-element continuity achievable by B-splines.

k -refinement

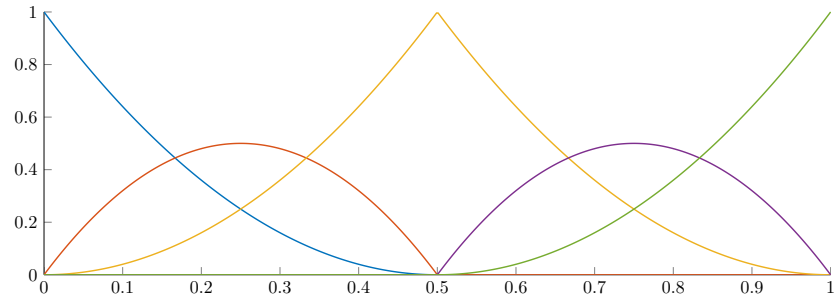
The procedure of k -refinement consists in performing first order elevation of the basis from degree p to \tilde{p} , followed by the insertion of a new knot $\tilde{\xi}_i$. It is easy to see that now the basis attains $C^{\tilde{p}-1}$ -continuity at the newly inserted knot $\tilde{\xi}_i$. A comparison of these refinement strategies is provided in Figure 4.5, starting from the linear basis associated with the open knot vector $\Xi = [0 \ 0 \ 0.5 \ 1 \ 1]$.



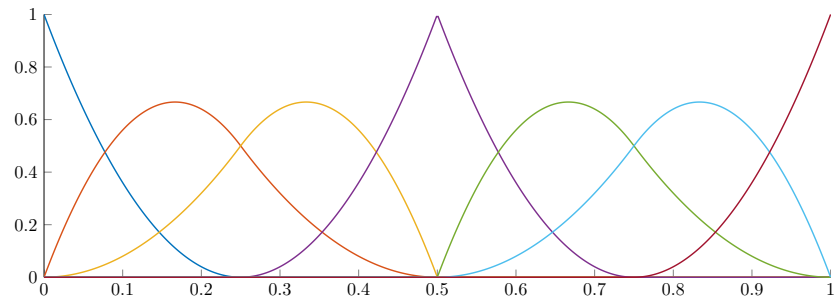
(a) Unrefined basis, $\Xi = [0 \ 0 \ 0.5 \ 1 \ 1]$.



(b) h -refinement, $\tilde{\Xi} = [0 \ 0 \ 0.25 \ 0.5 \ 0.75 \ 1 \ 1]$.



(c) p -refinement, $\tilde{\Xi} = [0 \ 0 \ 0 \ 0.5 \ 0.5 \ 1 \ 1 \ 1]$.



(d) k -refinement, $\tilde{\Xi} = [0 \ 0 \ 0 \ 0.25 \ 0.5 \ 0.5 \ 0.75 \ 1 \ 1 \ 1]$.

Figure 4.5 – Comparison of different refinement strategies.

4.2 Trimming

Trimming is a basic mathematical Boolean operation, which allows for an easy description of complex geometries and it is regarded as standard in most commercial CAD softwares. A trimmed surface $\mathbf{S} \in \mathbb{R}^3$ is composed of two parts, an underlying geometry, that describes the geometric shape, and a set of properly ordered trimming curves $\mathbf{C}^{\text{trim}} = \{\mathbf{C}_i\}$, that delimit the regions that are clipped. We refer to the latter as *trimming loop* in the following. Then, for the analysis, we need the description of these curves in the parameter space of the corresponding surface, where a typical instance of this framework is depicted in Figure 4.6. This step is needed in order to build suitable integrations rule for those elements that are cut. We remark that, since generally the inverse mapping of \mathbf{C}^{trim} is not known analytically, its parametric representation $\hat{\mathbf{C}}^{\text{trim}} \in \mathbb{R}^2$ is only an approximation of the true trimming curves which depends on the chosen geometric tolerance. Even for those cases where an analytical solution is available, the resulting order of $\hat{\mathbf{C}}^{\text{trim}}$ is usually intractable from a computational standpoint [Renner and Weiß, 2004]. Therefore, an approximation is usually found by employing a surface-to-surface intersection algorithm, for further details we refer to [Hohmeyer, 1992; Farin et al., 2002]. Despite its simple definition, trimming yields to severe challenges in the context

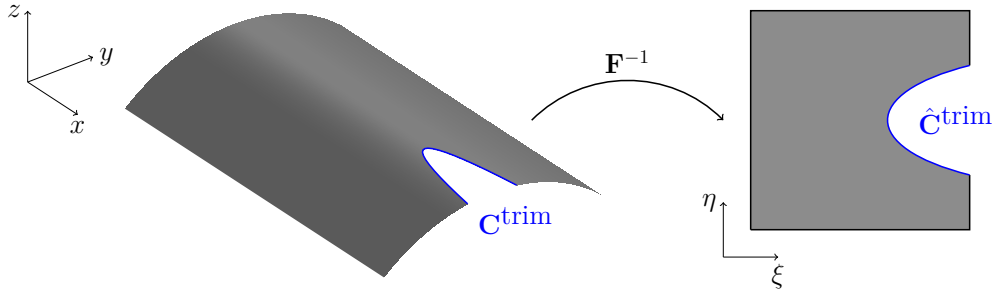


Figure 4.6 – Example of trimmed surface and its associated trimming loop.

of a smooth Design-Through-Analysis workflow. Indeed, when a trimming operation is performed within a CAD software, the visualization of the resulting surface is modified but its underlying mathematical description remains unchanged. Moreover, in general, trimming operations yield overlapping and/or non-watertight designs, commonly known as dirty geometries. For a detailed review of trimming and related open challenges in IGA we refer to [Marussig and Hughes, 2018] and references therein.

4.2.1 Boundary Representation (B-Rep)

B-Rep is the most common representation technique of free-form solid objects in CAD. The key idea of B-Rep consists in describing a volume by its outer skin, i.e. its boundary [Mäntylä, 1987]. More precisely, we can define a B-Rep as an entity formed by a geometry and endowed by its associated topological information. In particular, the

geometry is described by (trimmed) B-spline/NURBS curves and surfaces, whereas the topology expresses the relationships existing between these geometrical entities. A brief summary of the topological information is provided in the following.

Remark 4.3 *Although many engineering applications, such as additive manufacturing, require a full volumetric discretization of the object, B-Reps are clearly suited to be used in conjunction with dimensionally-reduced models, e.g. plates and shells, for the analysis. Indeed, these theories require only a description of the mid-surface of the three-dimensional object, which can be easily obtained in modern CAD softwares.*

Topological description

In this work, we make use of the so-called *face-edge-vertex* B-Rep representation. Analogously to the building blocks forming a finite element mesh, we can define the following topological entities:

$$V = \{v_i \mid i = 1, \dots, n_v\} \quad (4.8a)$$

$$E = \{e_j \mid j = 1, \dots, n_e\} \quad (4.8b)$$

$$F = \{f_k \mid k = 1, \dots, n_f\}, \quad (4.8c)$$

where V, E and F denote the set of vertices, edges and faces, respectively, and n_v, n_e and n_f are the corresponding number of vertices, edges and faces in the B-Rep. Each topological component has a geometric counterpart, namely surfaces \mathbf{S} , curves \mathbf{C} and points \mathbf{P} , respectively. To fix ideas, an example of a B-Rep structure is provided in Figure 4.7.

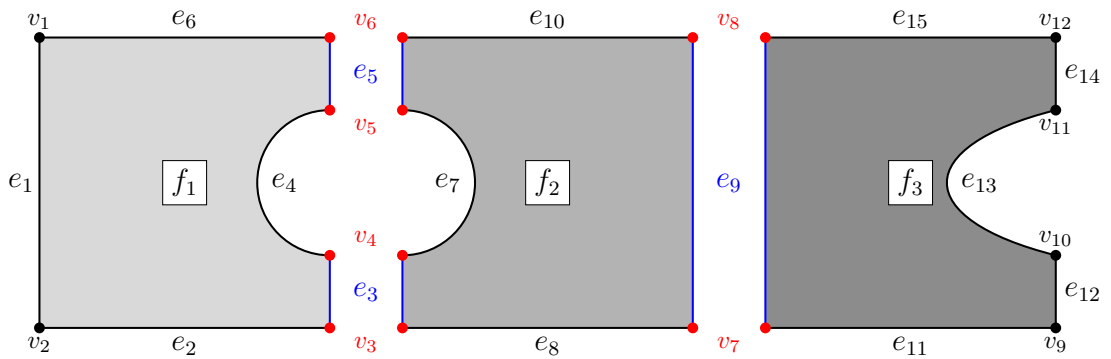


Figure 4.7 – Example of a B-Rep and its topological entities. Shared edges and vertices are colored in blue and red, respectively. Note that we have separated the faces for visualization purposes.

4.2.2 Mathematical framework of trimming

Let us now introduce the basic mathematical setting in the scope of isogeometric methods defined on trimmed domains, following closely the notation used in [Antolin et al., 2019; Buffa et al., 2020]. Let us define the untrimmed domain $\Omega_0 \subset \mathbb{R}^d$, described by a spline map $\mathbf{F} : \hat{\Omega}_0 = [0, 1]^d \rightarrow \Omega_0$, where $\hat{\Omega}_0$ denotes the corresponding parametric domain. Let us also introduce $\Omega_1, \dots, \Omega_N \subset \mathbb{R}^d$ Lipschitz-regular domains that define regions to be trimmed away from Ω_0 . We have seen that their boundaries are typically associated with trimming loops. Then, the physical domain can be written as follows:

$$\Omega = \Omega_0 \setminus \bigcup_{i=1}^N \bar{\Omega}_i, \quad (4.9)$$

where an example is depicted in Figure 4.8. It is worth highlighting that the trimming

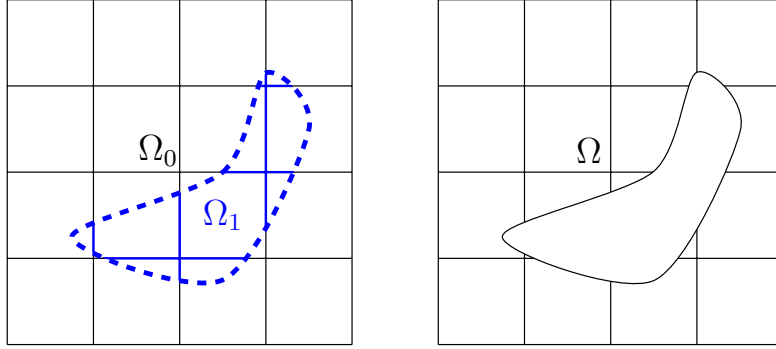


Figure 4.8 – Example of a trimmed domain. From the original rectangular domain Ω_0 the red domain Ω_1 is trimmed away to produce the final domain Ω .

operation does not change the underlying mathematical description of the original domain. Hence, elements and basis functions are constructed with respect to the non-trimmed domain Ω_0 and the correct physics is handled at the integration level. Given the definition of our computational domain in (4.9), we can now split its boundary in a part which coincides with the boundary of the original domain $\partial\Omega \cap \partial\Omega_0$ and a trimmed part $\partial\Omega \setminus \partial\Omega_0$. Contrary to conforming standard element techniques, the imposition of essential boundary conditions requires additional care in the trimmed case and it is usually performed in a weak sense. We will clarify this aspect in more details in a later section.

4.2.3 Integration of trimmed elements

The next point to be addressed concerns the integration of those element that are cut by a trimming curve. In particular, this introduces a discontinuity in the integrand, which needs to be properly handled by the integration scheme. Several methods have been proposed in the literature, where the most prominent can be categorized into:

- adaptive integration schemes, typically based on recursive algorithms, e.g. quadtree partitioning [Parvizia et al., 2007],
- moment fitting schemes [Müller et al., 2013; Joulaian et al., 2016],
- methods based on a local re-parametrization of the trimmed elements [Kudela et al., 2015, 2016; Breitenberger et al., 2015; Guo et al., 2018].

In this work we leverage the re-parametrization tool presented in [Antolin et al., 2019] to properly distribute the integration points. This family of methods relies on the automatic creation of a high-order integration mesh for those cells that are cut. We highlight that this mesh is used solely for integration and therefore badly shaped elements are still valid as long as a non-negative determinant of their corresponding Jacobian is ensured. We will refer to the latter elements as *tiles* in the remainder of this work. An example of the general integration workflow is given in Figure 4.9.

Remark 4.4 *For computational efficiency, it is important to find a good balance between the accuracy of the numerical integration and the number of integration elements that are generated.*

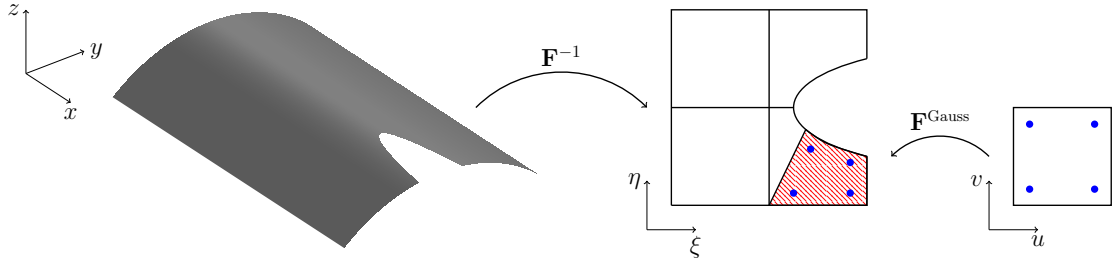


Figure 4.9 – Example of the local re-parametrization algorithm. From the physical domain Ω , the trimming curves are mapped back into the parameter space of the surface. Then, a high-order mesh is created for integration. The red surface represents one integration tile and the blue dots constitute the quadrature nodes.

4.3 Local refinement of spline technologies

Although the refinement strategies introduced for B-splines are paramount for the success of splines in CAD and analysis, they are not localized, meaning that the refinement propagates throughout the entire patch. A wide variety of technologies have been developed that allow for the local refinement of the basis and therefore allow to overcome the limitations intrinsic to the tensor-product nature of B-splines and NURBS. The most prominent in the literature are:

- hierarchical B-splines (\mathcal{HB}) [Forsey and Bartels, 1988; Greiner and Hormann, 1997; Kraft, 1997],
- truncated hierarchical B-splines (\mathcal{THB}) [Giannelli et al., 2012, 2016],
- T-splines [Bazilevs et al., 2010; Scott et al., 2012; Beirão da Veiga et al., 2013],
- subdivision surfaces [Peters and Reif, 2008],
- polynomial splines over T-meshes (PHT-splines) [Deng et al., 2008],
- LR-splines [Dokken et al., 2013; Bressan, 2013],
- ...

In this work, we make use of \mathcal{HB} - and \mathcal{THB} -splines, due to their conceptual simplicity, ease of implementation and since they are well-understood from a mathematical standpoint. Indeed, it can be shown that \mathcal{HB} and \mathcal{THB} yield a linearly independent basis suitable for the analysis also in the presence of trimming, where we refer to [Höllig, 2003, Section 4.5] for a proof in the context of hierarchical B-splines.

4.3.1 Hierarchical B-spline basis

Let us now review the fundamental concepts behind the hierarchical B-spline basis. Here, we summarize the definition of \mathcal{HB} given in [Vuong et al., 2011; Kraft, 1997]. Let $V^0 \subset V^1 \subset \dots \subset V^N$ be a sequence of $N + 1$ nested multivariate splines spaces, determined by their corresponding degree and knot vectors. Then, the B-spline basis associated with the space V^ℓ of level ℓ is denoted by $\mathcal{B}^\ell := \{b_i^\ell \mid i = 0, \dots, N^\ell - 1\}$, where N^ℓ represents the dimension of V^ℓ . Additionally, let us denote by \mathcal{T}_0^ℓ the mesh associated with \mathcal{B}^ℓ , where $K \in \mathcal{T}_0^\ell$ represents a cell of level ℓ .

Let us also define the set $\mathbf{\Omega}_0^N := \{\Omega_0^0, \Omega_0^1, \dots, \Omega_0^N\}$ as a hierarchy of subdomains of depth N if the following holds:

$$\hat{\Omega}_0 = \Omega_0^0 \supset \Omega_0^1 \supset \dots \supset \Omega_0^{N-1} \supset \Omega_0^N = \emptyset, \quad (4.10)$$

and each subdomain Ω_0^ℓ is the union of closed cells of level $\ell - 1$, where Ω_0^0 coincides with the initial parametric domain $\hat{\Omega}_0$. The subscript 0 is used to remark that up to now we refer *only* to the non-trimmed domain Ω_0 . We can now define the hierarchical B-spline basis \mathcal{HB} . Given a sequence of spaces $\{V^\ell\}_{\ell=0,\dots,N}$ (as defined above) with the corresponding B-spline bases $\{\mathcal{B}^\ell\}_{\ell=0,\dots,N}$ and a hierarchy of subdomains $\mathbf{\Omega}_0^N$ of depth

N , we define \mathcal{HB} as the $(N - 1)$ -th step of the following recursive definition:

$$\mathcal{HB}^0 = \mathcal{B}^0 \quad (4.11)$$

$$\mathcal{HB}^{\ell+1} = \left\{ b \in \mathcal{HB}^\ell \mid \text{supp } b \not\subset \Omega_0^{\ell+1} \right\} \cup \quad (4.12)$$

$$\left\{ b \in \mathcal{B}^{\ell+1} \mid \text{supp } b \subset \Omega_0^{\ell+1} \right\}, \quad \ell = 0, \dots, N - 2, \quad (4.13)$$

where an example is depicted in Figure 4.10. Consequently, we can introduce the

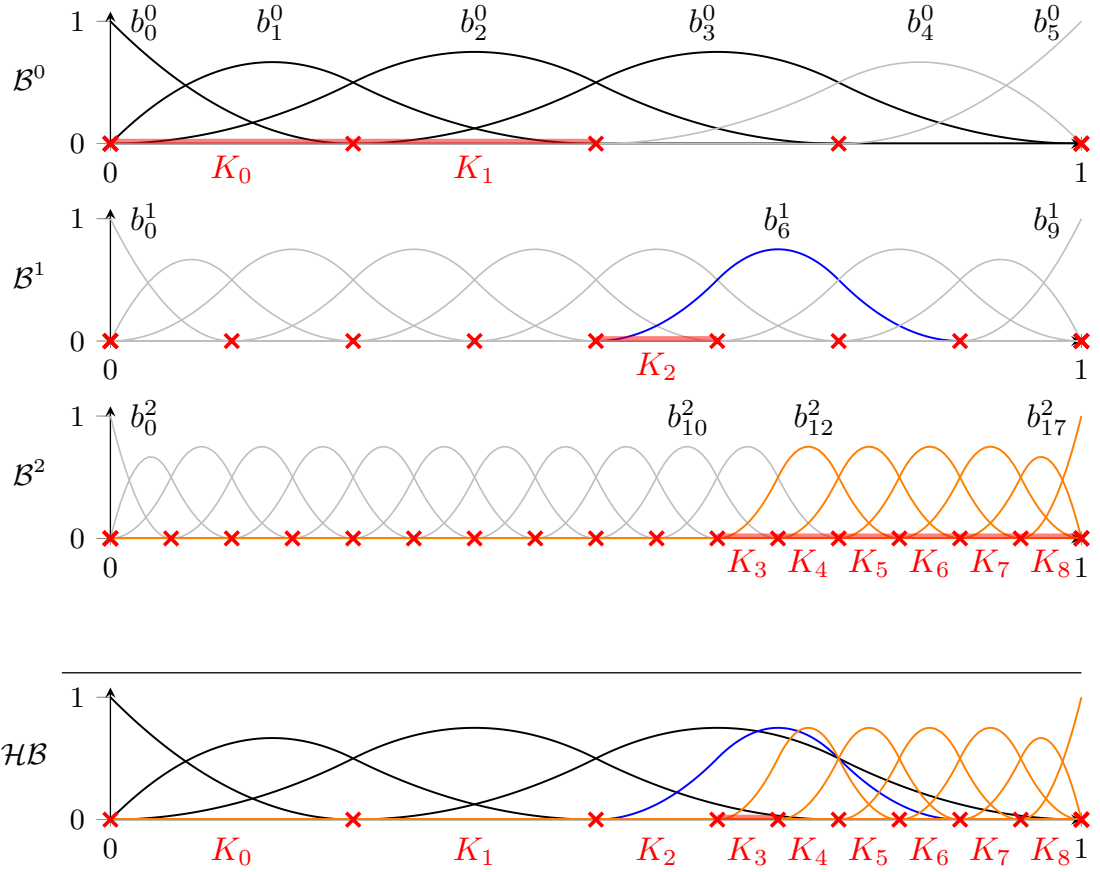


Figure 4.10 – Example of \mathcal{HB} basis on three levels. Active elements are highlighted in red, whereas red crosses indicate the location of the knots.

(parametric) mesh of level $\hat{\mathcal{T}}_A^\ell$ and the resulting hierarchical mesh $\hat{\mathcal{T}}_0$ associated with \mathcal{HB} as:

$$\hat{\mathcal{T}}_{0,A}^\ell = \left\{ K \in \mathcal{T}_0^\ell \mid K \subset \Omega_0^\ell \wedge K \not\subset \Omega_0^{\ell+1} \right\} \quad (4.14)$$

$$\hat{\mathcal{T}}_0 = \bigcup_{\ell=0}^{N-1} \hat{\mathcal{T}}_{0,A}^\ell. \quad (4.15)$$

Finally, in the following, we will refer to b as an active function if $b \in \mathcal{HB}$ and as an active

function of level ℓ if $b \in \mathcal{B}_A^\ell := \mathcal{HB} \cap \mathcal{B}^\ell$. An analogous terminology can be introduced for elements, where we denote K as an active cell if $K \in \widehat{\mathcal{T}}_0$ and as an active cell of level ℓ if $K \in \widehat{\mathcal{T}}_{0,A}^\ell$. For further details we refer to [Garau and Vázquez, 2018].

4.3.2 Truncated hierarchical B-spline basis

We now define the truncated hierarchical B-spline basis \mathcal{THB} , as introduced in [Giannelli et al., 2012]. The only difference between \mathcal{HB} and \mathcal{THB} is that in the latter the basis functions whose support overlaps finer elements are truncated as described in the following. This generates a basis that spans the same space as \mathcal{HB} , but has in general better properties from a numerical standpoint. Indeed, \mathcal{THB} :

- have smaller support,
- form a partition of unity,
- possess strongly stable stability constants,

where we refer to [Giannelli et al., 2013] for further details. Let us first introduce the truncation operator. Namely, exploiting the fact that B-splines of level ℓ can be written as a linear combinations of B-splines of level $\ell + 1$ with non-negative coefficients, the truncation operator with respect to level $\ell + 1$ reads:

$$\text{trunc}^{\ell+1}(b_i^\ell) = \sum_{k=1}^{N^{\ell+1}} \tilde{c}_k^{\ell+1}(b_i^\ell) b_k^{\ell+1}, \quad (4.16)$$

where the coefficients are defined by:

$$\tilde{c}_k^{\ell+1}(b_i^\ell) = \begin{cases} 0 & \text{if } b_k^{\ell+1} \in \mathcal{HB}^{\ell+1} \cap \mathcal{B}^{\ell+1}, \\ c_k^{\ell+1} & \text{otherwise,} \end{cases} \quad (4.17)$$

where $c_k^{\ell+1}$ are the standard non-negative coefficients of the two-scale relation. They allow to write a B-spline of level ℓ as a linear combination of splines of level $\ell + 1$ as follows:

$$b_i^\ell = \sum_k c_k^{\ell+1}(b_i^\ell) b_k^{\ell+1} \quad \forall b_i^\ell \in \mathcal{B}^\ell. \quad (4.18)$$

Now, the truncated hierarchical basis \mathcal{THB} is defined as the $(N - 1)$ -th step of the following recursive algorithm:

$$\mathcal{THB}^0 = \mathcal{B}^0 \quad (4.19a)$$

$$\mathcal{THB}^{\ell+1} = \left\{ \text{trunc}^{\ell+1}(b) \mid b \in \mathcal{THB}^{\ell} \wedge \text{supp } b \not\subset \Omega_0^{\ell+1} \right\} \cup \left\{ b \in \mathcal{B}^{\ell+1} \mid \text{supp } b \subset \Omega_0^{\ell+1} \right\}, \quad \ell = 0, \dots, N - 2, \quad (4.19b)$$

where an example defined on the same hierarchical mesh of Figure 4.10 is given in Figure 4.11. Finally, we can introduce the following discrete space:

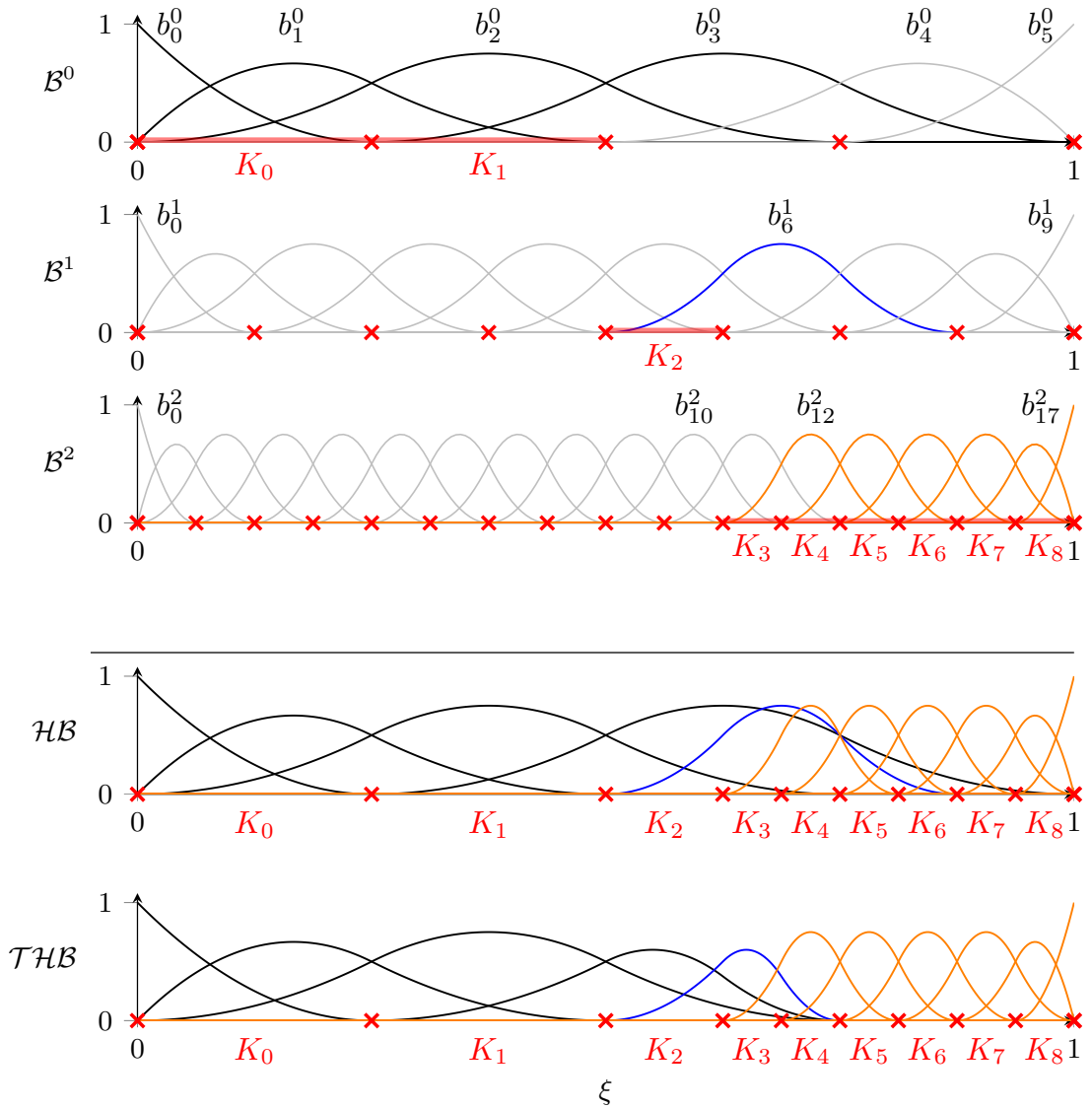


Figure 4.11 – Example of \mathcal{THB} basis on three levels. Active elements are highlighted in red, whereas red crosses indicate the location of the knots.

$$X_h = \text{span} \left\{ b \circ \mathbf{F}^{-1} \mid b \in \mathcal{HB} \right\} = \text{span} \left\{ b \circ \mathbf{F}^{-1} \mid b \in \mathcal{THB} \right\}, \quad (4.20)$$

which can be directly employed in the Galerkin approximation of PDEs for non-trimmed domains.

4.3.3 Analysis-suitable Truncated Hierarchical B-splines on trimmed domains

Let us now trim the aforementioned domain Ω_0 as defined in (4.9) and let us make the assumption that $\Omega \subset \mathbb{R}^d$ is regular enough. In order to define a (truncated) hierarchical basis suitable for the analysis on Ω , we follow the construction provided in [Höllig, 2003, Section 4.5], which guarantees the linear independence of the basis. Recalling the definition provided in (4.10) for the non-trimmed case, we introduce a hierarchy of trimmed subdomains $\Omega^N := \{\Omega^0, \Omega^1, \dots, \Omega^N\}$ such that:

$$\widehat{\Omega} = \Omega^0 \supset \Omega^1 \supset \dots \supset \Omega^{N-1} \supset \Omega^N = \emptyset, \quad (4.21)$$

where $\widehat{\Omega}$ denotes the trimmed parametric domain and N represents the depth of the hierarchy. Notice that we dropped the subscript in our notation consistently with our definition of trimmed domain (4.9). Then, let us define the support of a spline function b restricted to the trimmed domain as:

$$\text{supp}_{\widehat{\Omega}}(b) := \widehat{\Omega} \cap \text{supp}(b). \quad (4.22)$$

Now, for every level ℓ , we introduce the corresponding B-spline basis restricted to $\widehat{\Omega}$ as:

$$\mathcal{B}_{\widehat{\Omega}}^{\ell} := \left\{ b|_{\widehat{\Omega}} : b \in \mathcal{B}^{\ell} \wedge \text{meas}(\text{supp}_{\widehat{\Omega}}(b)) \neq 0 \right\}. \quad (4.23)$$

With these definitions, the recursive algorithms for the construction of the \mathcal{HB} and \mathcal{THB} bases on trimmed domains are analogous to those introduced in Sections 4.3.1 and 4.3.2 in the non-trimmed case, where we need to replace the hierarchy of subdomains with its trimmed counterpart as defined in (4.21), the support of functions with the definition in (4.22) and the B-spline basis of level ℓ , \mathcal{B}^{ℓ} , with its restriction to $\widehat{\Omega}$ as provided in (4.23). At this point, we highlight that it was proved in [Höllig, 2003, Section 4.5] that this construction guarantees the linear independence of the basis in the scope of hierarchical B-splines. We remark that the truncation operator does not affect this property (as it only acts reducing the original support) and therefore the same rationale can be applied to the \mathcal{THB} basis, yielding again a basis that is suitable for the analysis in the context of trimmed domains. Then, we also modify the definition of parametric

mesh of level and hierarchical mesh as follows:

$$\widehat{\mathcal{T}}_A^\ell = \left\{ K \in \widehat{\mathcal{T}}_{0,A}^\ell : \text{meas}(K \cap \widehat{\Omega}) \neq 0 \right\} \quad (4.24a)$$

$$\widehat{\mathcal{T}} = \left\{ K \in \widehat{\mathcal{T}}_0 : \text{meas}(K \cap \widehat{\Omega}) \neq 0 \right\}, \quad (4.24b)$$

from which the physical mesh reads:

$$\mathcal{T} = \left\{ \mathbf{F}(K) : K \in \widehat{\mathcal{T}} \right\}. \quad (4.25)$$

Remark 4.5 *We remark that the construction of (hierarchical) B-splines can be performed directly on the physical domain, replacing b by $b \circ \mathbf{F}^{-1}$, and the subdomains Ω_0^ℓ by $\mathbf{F}(\Omega_0^\ell)$. The same holds for the trimmed case. When the interpretation is clear from the context, we will use either one or the other notation interchangeably.*

Remark 4.6 *Here we provide an algorithm that identifies the active basis functions and retains the linear independence of the trimmed basis (as previously defined) based on the standard implementation of (truncated) hierarchical B-splines. This allows for an easier implementation of the trimmed basis on top of existing isogeometric codes. In particular, when a trimmed element K is marked for refinement, we also refine all those elements outside Ω (but inside Ω_0) contained in the support of basis functions whose support incorporate K . We refer to these elements as ‘ghost’ cells. This procedure guarantees that when all the elements in the support of a function that intersects Ω are refined, those ones outside Ω will also be refined, and therefore the function will be deactivated. We summarize a possible implementation in Algorithm 1.*

Algorithm 1 Algorithm for avoiding linear dependence of the \mathcal{HB} or \mathcal{THB} basis in the standard implementation.

```

1: procedure AVOID_LINEAR_DEPENDENCE(hierarchical mesh  $\mathcal{T}$ )
2:   for each level  $\ell$  of  $\mathcal{T}$  do
3:     for each  $K \in \mathcal{T}^\ell$  marked for refinement do
4:       Get all functions with support on  $K$ 
5:       Get the supports of these functions
6:       Get all ghost cells within these supports
7:       Mark the ghost cells for refinement
8:     end for
9:   end for
10:  Refine marked elements
11: end procedure

```

Lastly, we introduce the approximation space defined on the trimmed domain Ω as follows:

$$\tilde{X}_h = \text{span} \left\{ b \circ \mathbf{F}^{-1} \mid b \in \mathcal{HB} \right\} = \text{span} \left\{ b \circ \mathbf{F}^{-1} \mid b \in \mathcal{THB} \right\}, \quad (4.26)$$

where the construction of \mathcal{HB} and \mathcal{THB} takes trimming into account as explained in details above. Analogously to the non-trimmed case, this space can be used for the Galerkin discretization of PDEs.

4.4 Galerkin discretization of Kirchhoff plates and Kirchhoff-Love shells

Historically, plate and shells formulations based on Reissner-Mindlin type theories have been predominant in the finite element world. This can be easily explained by the fact that these formulations only require C^0 -continuity between elements, since displacements and rotations are discretized separately. Therefore, standard linear shape functions are suffice to fulfill the continuity requirement. On the other hand, we have seen that Kirchhoff-based formulations require at least C^1 inter-element continuity to yield a well-defined bending operator. This constraint poses severe challenges for their finite element implementation. Several solutions can be found in the literature, where we summarize the most relevant in the following. For plates, the C^1 -continuity requirement can be relaxed by employing the so-called *discrete Kirchhoff* concept, see the pioneering works [Stricklin et al., 1969; Dhett, 1970] and [Batoz et al., 1980] for its mathematical foundations and computer-oriented implementation. In these elements, the Kirchhoff constraint is enforced only at the discrete level, but convergence to the limit Kirchhoff plate solution is obtained under mesh refinement. Concerning Kirchhoff-Love shells, given the vast amount of possible simplifications and corresponding formulations, we refer to the survey [Bischoff et al., 2004]. A conforming discretization of the Kirchhoff-Love problem in its primal form, meaning based only on displacement degrees-of-freedom, has been studied in [Cirak et al., 2000] in the context of subdivision surfaces. The first work using B-splines has been introduced in [Kiendl et al., 2009], where the rotation-free isogeometric shell element employed in this thesis was outlined. At this point, it should be clear that B-splines and variants thereof are an ideal candidate for the discretization of Kirchhoff-Love shells, since the continuity constraint is satisfied automatically within one patch for splines of degree $p \geq 2$ and maximum smoothness. In particular, following the Galerkin rationale, we can state the discretized counterpart of the weak formulation in Equation (3.5) associated with the Kirchhoff problem, respectively, as follows. Find $u_h \in V_h$ such that:

$$a(u_h, v_h) = f(v_h) \quad \forall v_h \in V_h, \quad (4.27)$$

where the choice of the discrete space $V_h \subset X_h$ or $V_h \subset \tilde{X}_h$ depends in general on the boundary conditions of the problem at hand. We remark that here the superscript (\cdot) is used to distinguish between the non-trimmed and trimmed case. Similarly for the Kirchhoff-Love problem, we can write the discrete counterpart of Equation (3.41) as: find $\mathbf{u}_h \in V_h$ such that:

$$a(\mathbf{u}_h, \mathbf{v}_h) = f(\mathbf{v}_h) \quad \forall \mathbf{v}_h \in V_h, \quad (4.28)$$

where again the choice $V_h \subset [X_h]^3$ or $V_h \subset [\tilde{X}_h]^3$ depends on the boundary conditions. Finally, in both cases, the Galerkin approximation can be conveniently expressed in matrix form as described in Equation (2.40):

$$\mathbf{K}\tilde{\mathbf{u}} = \mathbf{f}, \quad (4.29)$$

from which it is clear to see that the unknowns are only related to the deflection or displacement field, respectively.

4.5 Numerical results

In the following numerical experiments, we show the potential of local refinement in efficiently capture localized traits of the solution. We remark that refinement is achieved by employing \mathcal{HB} -splines as a basis for the analysis. Furthermore, we numerically verify that local refinement improves the accuracy of the imposition of weak boundary conditions (e.g. by means of the penalty method) on complex trimmed boundaries and mitigates the effect of over-constraining of the solution space in those elements close to trimming curves where boundary conditions are enforced weakly. Analogous locking effects will also be studied in depth in the context of multi-patch coupling in a later chapter. Moreover, we shed some light on the issue of spurious coupling of basis functions arising when “small” (compared to the element size h) features are present in the geometry (e.g. thin holes), where again the use of local refinement proves to be an effective remedy. All results have been obtained by extending the Finite Cell Method (FCM) to the analysis of thin structures, developing the ideas first introduced in [Rank et al., 2011] and applying the boundary-conforming adaptive integration scheme described in [Kudela et al., 2015, 2016] to trimmed surfaces, see also [Coradello, 2016]. For the sake of conciseness, we do not review the FCM here but the interested reader is referred to [Düster et al., 2008; Parvizian et al., 2007; Schillinger et al., 2012] and references therein.

4.5.1 On the imposition of weak constraints

In this work, we consider weak constraints applied by means of the penalty method [Babuška, 1973]. For Kirchhoff-Love shells, a weak displacement and rotational boundary

condition terms are added to the weak formulation to enforce the corresponding boundary conditions. Due to their conceptual simplicity and straightforward implementation, penalty-type strategies are extensively used for the imposition of boundary conditions in trimmed shells, e.g. see [Kiendl, 2011; Breitenberger, 2016; Lei et al., 2015; Pasch et al., 2021] and references therein. Other methods have been successfully applied, e.g. extensions of the original Nitsche's method [Nitsche, 1971] have been studied in [Apostolatos et al., 2015; Guo and Ruess, 2015; Kamensky et al., 2015; Benzaken et al., 2021]. However, generally speaking, these methods are computationally more involved since they require the computation of higher-order covariant derivatives. Another family of strategies is based on the introduction of Lagrange multipliers associated with the boundary constraints, see e.g. [Apostolatos et al., 2015; Apostolatos, 2019; Hirschler et al., 2019; Schuß et al., 2019]. It is well-known that mortar methods introduce additional artificial unknowns into the underlying system of equations to enforce the corresponding constraints, where the choice of discretization space for these Lagrange multipliers plays a pivotal role for the robustness of the method. In particular, the inf-sup stability is a crucial property, see [Boffi et al., 2013] for further details. In our derivation, the discrete Kirchhoff-Love problem in Equation (4.28) is augmented by suitable penalty terms as:

$$\begin{aligned} &\text{Find } \mathbf{u}_h \in V_h \text{ such that} \\ &a(\mathbf{u}_h, \mathbf{v}_h) + b^{\text{disp}}(\mathbf{u}_h, \mathbf{v}_h) + b^{\text{rot}}(\mathbf{u}_h, \mathbf{v}_h) = f(\mathbf{v}_h) \quad \forall \mathbf{v}_h \in V_h, \end{aligned} \quad (4.30)$$

where:

$$b^{\text{disp}}(\mathbf{u}_h, \mathbf{v}_h) = \int_{\Gamma_u} \alpha^{\text{disp}}(\mathbf{u}_h - \tilde{\mathbf{u}}) \cdot \mathbf{v}_h \, \mathbf{d}\gamma, \quad (4.31a)$$

$$b^{\text{rot}}(\mathbf{u}_h, \mathbf{v}_h) = \int_{\Gamma_\theta} \alpha^{\text{rot}}[\theta_n(\mathbf{u}_h) - \tilde{\theta}_n]\theta_n(\mathbf{v}_h) \, \mathbf{d}\gamma, \quad (4.31b)$$

where α^{disp} and $\alpha^{\text{rot}} \in \mathbb{R}^+$ are the displacement and rotational penalty parameters, respectively. Typically, they are scaled with the material properties and element size [Utku and Carey, 1982], where following [Herrema et al., 2019] we set:

$$\alpha^{\text{disp}} = \delta \frac{Et}{h(1 - \nu^2)}, \quad (4.32a)$$

$$\alpha^{\text{rot}} = \delta \frac{Et^3}{12h(1 - \nu^2)}, \quad (4.32b)$$

where δ is a problem independent, user-defined parameter. Furthermore, for non-uniform refinement, h is heuristically chosen as the smallest element size of the mesh \mathcal{T} which intersects the boundary where the corresponding boundary condition is enforced. Again, we will analyze this choice more in details in a later chapter in the scope of coupling multi-patch plates and shells. At this point we highlight that, when imposing boundary conditions weakly on arbitrarily-shaped trimming curve, it can happen that the finite element space is not capable of both accurately satisfy the boundary conditions

and approximate the numerical solution in the proximity of the trimming boundary. For complicated geometry and/or high penalty parameters α^{disp} and α^{rot} (without excessively deteriorating the conditioning of the system matrix), some elements will be over-constrained. To illustrate this effect we consider the setup depicted in Figure 4.12a, where a uniformly distributed vertical load $\tilde{\mathbf{f}} = (0, 0, -100)^\top$ is applied on the top surface of a violin, while clamped conditions are enforced with a penalty term ($\delta = 10^6$) on the internal curves, which feature kinks and areas of high curvature. In Figures 4.12b and 4.12c it can be observed that the displacement and stress are artificially low on the elements cut by the trimming curve, and these fields spuriously follow the element boundaries. We remark that this effect is more pronounced in the proximity of complex trimming patterns. Given a fixed mesh, it is in general desirable to balance the error in the application of boundary conditions and in the solution approximation. We refer to [de Prenter et al., 2018] for a discussion in the context of Nietsche’s method. However, in case it is not practical, or it is not known *a priori* how to optimally choose the penalty factors, mesh refinement can be employed to potentially improve accuracy in both interior and boundary terms by enlarging the finite element space and intersecting the curve with smaller elements, generally yielding simpler intersections. By doing this, small geometric features induced by the trimming curves can be selectively resolved. For instance, in Figures 4.12d and 4.12e the solution and Von Mises stress distribution obtained with \mathcal{HB} -splines of degree $p = 2$ is shown. This example qualitatively demonstrates the capability of local refinement to reduce the over-constraining effects linked to the weak imposition of Dirichlet-type boundary conditions. We compare the resultant Von Mises stresses for different meshes in Figure 4.13. Figure 4.13b shows the solution obtained with a uniform mesh of 38 607 dofs. The mesh is chosen to have a number of dofs similar to the locally refined mesh in Figures 4.12e and 4.13a, which consists of 36 225 dofs. As a reference, Figure 4.13c shows the solution obtained by an overkill mesh with 4026 378 dofs. Note that the locally refined mesh does not yield spurious low stress on the constrained curve, similarly to the overkill solution.

4.5.2 Thin holes

In case trimming curves define holes that are “thin” compared to the geometry knot spans, it can happen that the support of a basis function intersected with the computational domain Ω is disconnected and composed of several disjoint sub-domains. As an example, the support of a function with two disjoint physical sub-domains is depicted in red in Figure 4.15b. This creates an unphysical coupling between the two sides of the hole, that generally results in an inaccurate mechanical response. These spurious effects become particularly severe when the local behavior of the structure is strongly determined by the geometry of the hole, as it is often the case for complex models. For instance, consider the setup illustrated in Figure 4.15a, where the geometry is modeled via a trimmed NURBS surface of degree $p = 2$, which is exported from *Rhinoceros* together with a set of 32 trimming curves. It is worth remarking the complexity of the geometry at

hand, which presents several thin sharp features and kinks in its description. For the analysis, homogeneous Dirichlet boundary conditions on the displacement are applied to the outer boundary of the violin via the penalty method, as described above, and a line load $\tilde{\mathbf{f}} = (0, 0, -100)^\top [N/m]$ is applied to the reentrant tip of the f-hole, such that the tip is pulled upwards. A direct analysis on the coarse geometry yields an unphysical response, where the maximum displacement is attained towards the center of the geometry (see Figure 4.15c), instead of the reentrant tip, as expected from engineering intuition. In order to mitigate this issue, local refinement can be used to substitute functions with large support by functions with smaller support. In particular, the removal of the unphysical coupling is obtained when no support is composed of disjoint physical parts. To this end, let us consider the refinement procedure presented in Algorithm 2. Here, we introduce a parameter γ , which represents the extension of the refinement area in the proximity of a given trimming curve, where an example of the effects of different choices $\gamma = 0, 1, 2$ is depicted in Figure 4.14 for a single level ℓ . Furthermore, the definition of distance used in the algorithm is based on their position in the tensor product structure. In particular, if the vectors $\mathbf{i} = [i_\xi, i_\eta]$ and $\mathbf{i}' = [i'_\xi, i'_\eta]$ collect the positional indexes of elements K and K' , respectively, in the tensor product grid of level, we define:

$$\text{dist}(K, K') = \|\mathbf{i} - \mathbf{i}'\|_\infty, \quad (4.33)$$

as the distance between the two. Note that for $\gamma = p$ or higher, only functions of the

Algorithm 2 Refinement towards trimming curves

```

1: procedure REFINE (mesh  $\mathcal{T}$ , max level  $\ell_{\max}$ ,  $\Gamma_{\mathbf{u}}$  and  $\Gamma_\theta$ ,  $\gamma$ )
2:   for  $\ell = 1 \dots \ell_{\max}$  do
3:     for each active element  $K \in \mathcal{T}_A^\ell$  of level  $\ell$  cut by  $\Gamma_{\mathbf{u}}$  or  $\Gamma_\theta$  do
4:       for each active element  $K' \in \mathcal{T}_A^\ell$  of level  $\ell$  do
5:         if  $\text{dist}(K, K') \leq \gamma$  then
6:           mark  $K'$  for refinement
7:         end if
8:       end for
9:     end for
10:    refine marked elements
11:    update  $\mathcal{T}$ 
12:  end for
13: end procedure

```

finest level have support on the trimming curve. By direct comparison between the size of the finest knot spans and the size of the hole, one can obtain the value of ℓ_{\max} that removes the unphysical coupling. For the current example we obtain a decoupling value $\ell_{\max} = 5$ and a qualitatively correct response is observed in Figure 4.15d for $\gamma = p = 2$, where the maximum deflection occurs at the reentrant tip. In Figure 4.16a, the displacement at point A (marked with a blue dot in Figure 4.15b) is shown, where the latter point of interest is located at one of the geometrical kinks of the trimmed boundary.

Note the sudden improvement in accuracy of the solution obtained for $\ell_{\max} = 5$, while smaller choices of the maximum refinement level yield inaccurate results, since the spurious coupling is still present in the corresponding \mathcal{HB} bases. Then, in Figure 4.16a, we compare the convergence behavior of the solution obtained by local refinement to the tensor product refinement constructed by Algorithm 2 with $\gamma = 0$ and to a standard uniform tensor-product refinement, obtained by recursively bisecting every knot span. The meshes corresponding to local and tensor-product refinements for $\ell_{\max} = 5$ are depicted in Figures 4.17a and 4.17b, respectively. In Figure 4.16b we observe that all strategies converge towards a reference value for the displacement obtained from an overkill solution, where local refinement achieves a level of accuracy comparable to uniform refinement with substantially less degrees-of-freedom.

4.5.3 Localized deformations

The example given in Figure 4.15 produces a localized deformation (see Figure 4.15d) which needs to be properly resolved. We highlight once more that the local refinement strategy employed to remove the unphysical coupling already gives an approximation that is comparable to uniform refinement, but with a considerable reduction in dofs, where we refer to Figures 4.16a and 4.16b, respectively. In Table 4.1, the energy and z -displacement (at point A) errors are shown with respect to the following reference values $\tilde{E} \approx 0.2211$, $\tilde{u}_z \approx 7.268 \cdot 10^{-3} [m]$. These values have been obtained with an overkill solution on a uniformly refined mesh of level $\ell_{\max} = 5$. This numerical experiment confirms that local refinement accurately and efficiently captures both local quantities (such as the solution at point A) and global quantities (the energy of the system) compared to tensor product refinement, where in this particular example approximately 5 times less dofs are required to achieve a comparable level of accuracy.

	dofs	energy error $(1 - \frac{E^h}{E})\%$	z -displ. error $(1 - \frac{u_z^h}{\tilde{u}_z})\%$
tensor product ref.	193 359	2.07%	1.72%
local ref.	37 215	2.88%	2.29%

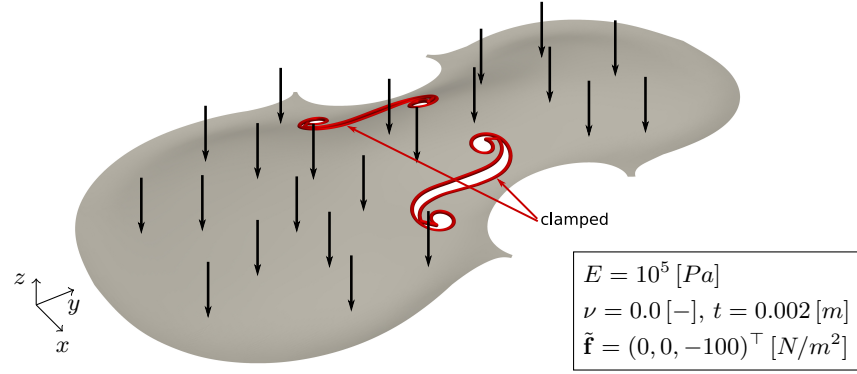
Table 4.1 – Comparison of the error in the energy norm and error in the z -displacement at point A against the number of dofs for tensor product and local refinements, respectively.

4.5.4 From CAD to analysis of an “engineering” structure

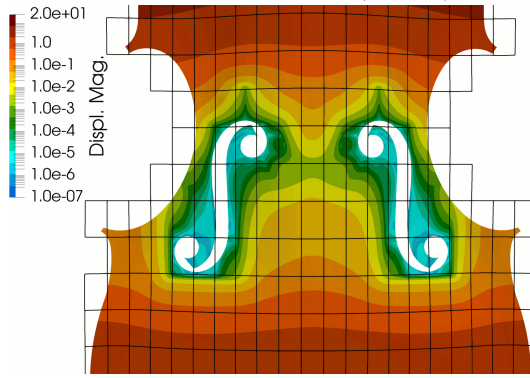
This example concerns the simulation of a simplified model, depicted in Figure 4.18a, of the rooftop of the Rolex Learning Center, the campus library at the École Polytechnique Fédérale de Lausanne (see Figure 4.18b). Here, we aim at showing the capabilities of the proposed numerical framework for the analysis of complex structures of architectural relevance. The geometry is modeled as a trimmed NURBS surface of degree $p = 3$

composed of 20×20 knot spans. A planar map of the building has been used to trace all the major structural holes. Then, 150 cylinders representing the pillars have been placed in a 15×10 Cartesian fashion into the model (those falling outside of the physical domain have been discarded). Their intersection with the surface is used to impose homogeneous Dirichlet boundary conditions, therefore modeling the static behavior of the supports. Let us remark the fact that this simplification in the design does not correspond to the actual placement of the pillars. Additionally, we remark that we are aware of the limitation of this model, but in the scope of this dissertation it is used as an illustrative example for the range of applicability of the proposed simulation tool, spanning from complex geometrical models to architectural designs. Lastly, the material parameters are chosen as $E = 40 \cdot 10^9 [Pa]$, $\nu = 0.15 [-]$ and $t = 0.2 [m]$ for the Young's modulus, Poisson's ratio, and thickness of the shell, respectively. The roof is subjected to its self-weight, set to $\tilde{\mathbf{f}} = (0, 0, -2 \cdot 10^3)^\top [N/m^2]$, and in Figure 4.18c the displacement magnitude field is depicted, where $k = 5$ levels of hierarchical refinement are used to resolve the boundary conditions given by the pillars and the trimming curves. A close-up of the solution in the vicinity of a trimming curve for a coarse uniform tensor-product mesh and a locally refined mesh is depicted in Figure 4.19. We note a substantial reduction of over-constraining of the solution field close to the pillars in the locally refined case. On one hand, this example clearly motivates local refinement, which is needed to resolve the small scale of the problem (the pillars) while considerably reducing the computational cost. On the other hand, it shows how all the relevant information needed to impose boundary conditions can be taken directly from the CAD model, in the spirit of reducing the gap between analysis and design.

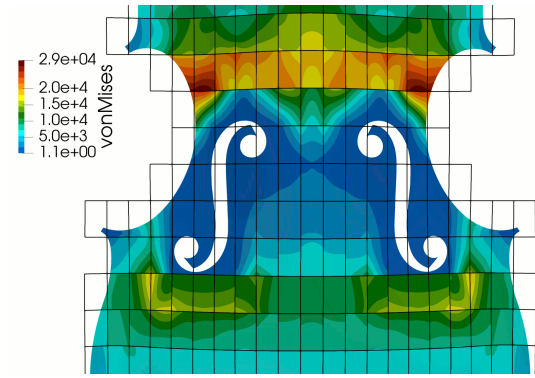
¹Pictures retrieved from <https://commons.wikimedia.org> under the Creative Commons license.



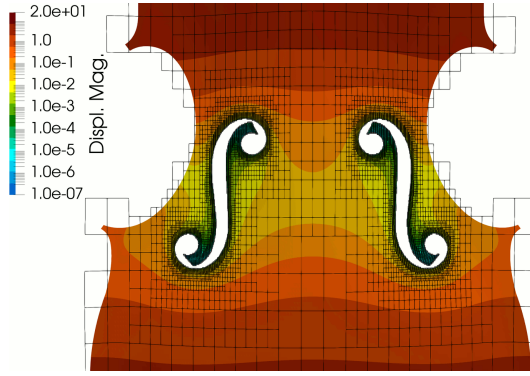
(a) Boundary conditions: clamped (red wire) and displacement load (arrows).



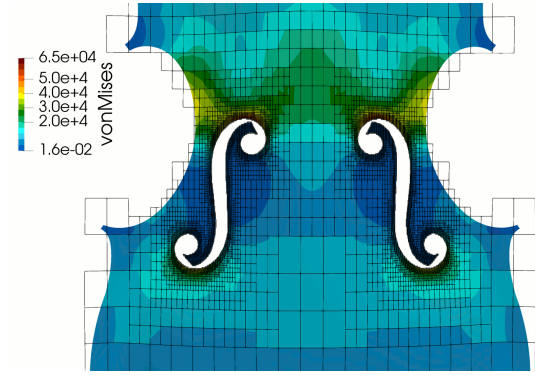
(b) Displacement magnitude (logarithmic scale) on unrefined mesh.



(c) Von Mises stress on unrefined mesh.



(d) Displacement magnitude (logarithmic scale) on a locally refined mesh.



(e) Von Mises stress on a locally refined mesh.

Figure 4.12 – Example of a violin subjected to gravity load. \mathcal{HB} -splines of degree $p = 2$ are used to achieve refinement.

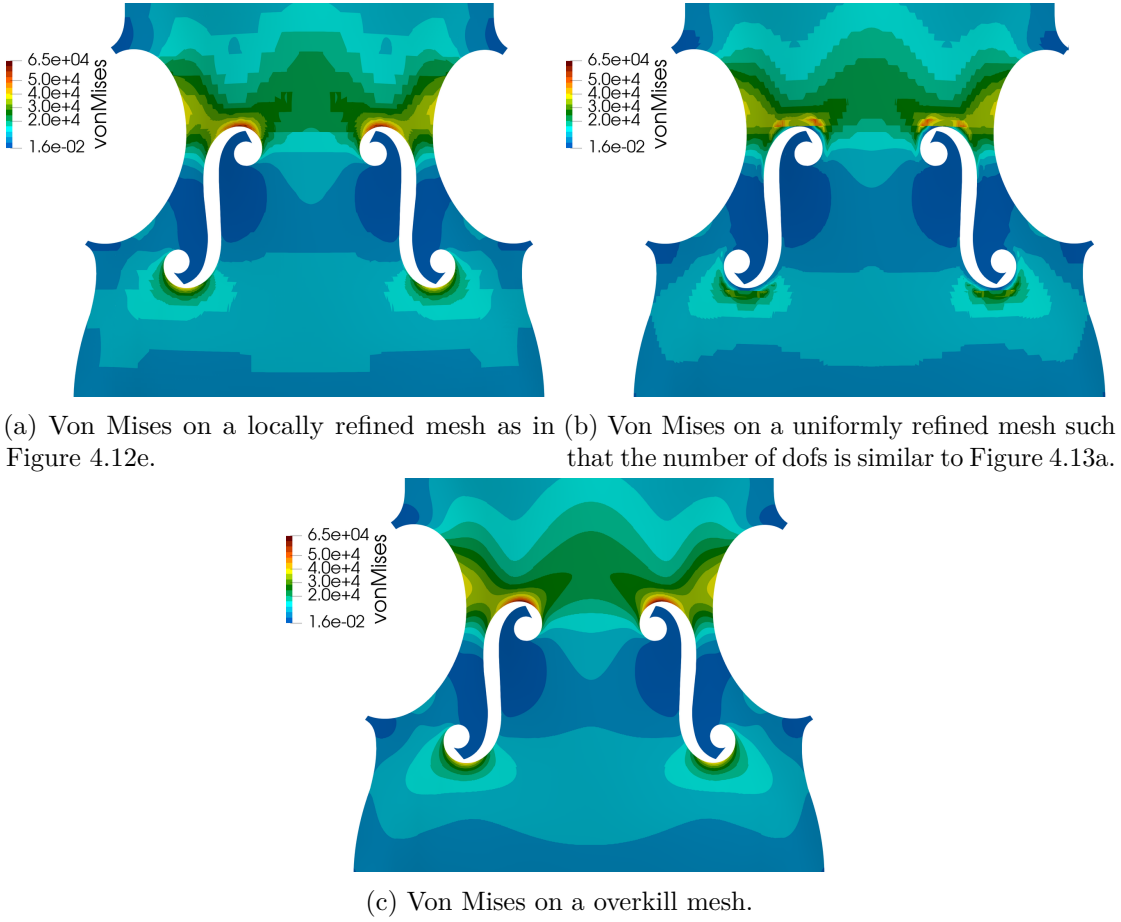


Figure 4.13 – Comparison of Von Mises stresses around the f-holes for different meshes.

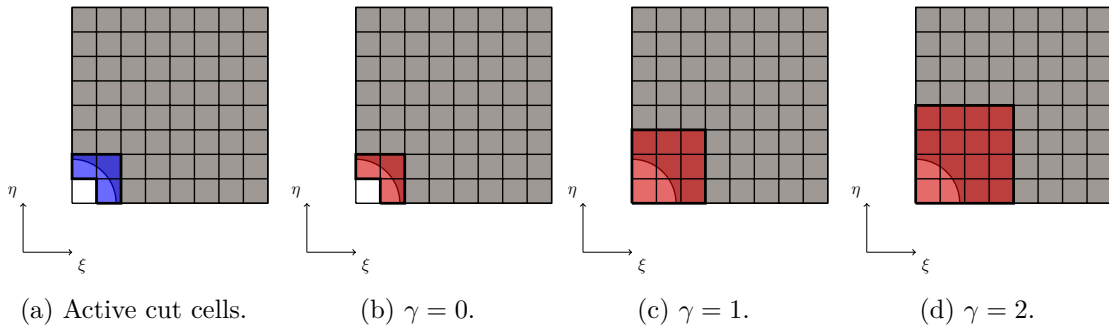


Figure 4.14 – Illustration of the influence of marking parameter γ on the refinement. The active cut cells of level are highlighted in blue, whereas the elements marked for refinement after performing Algorithm 2 with $\gamma = 0, 1, 2$, respectively, are colored in red.

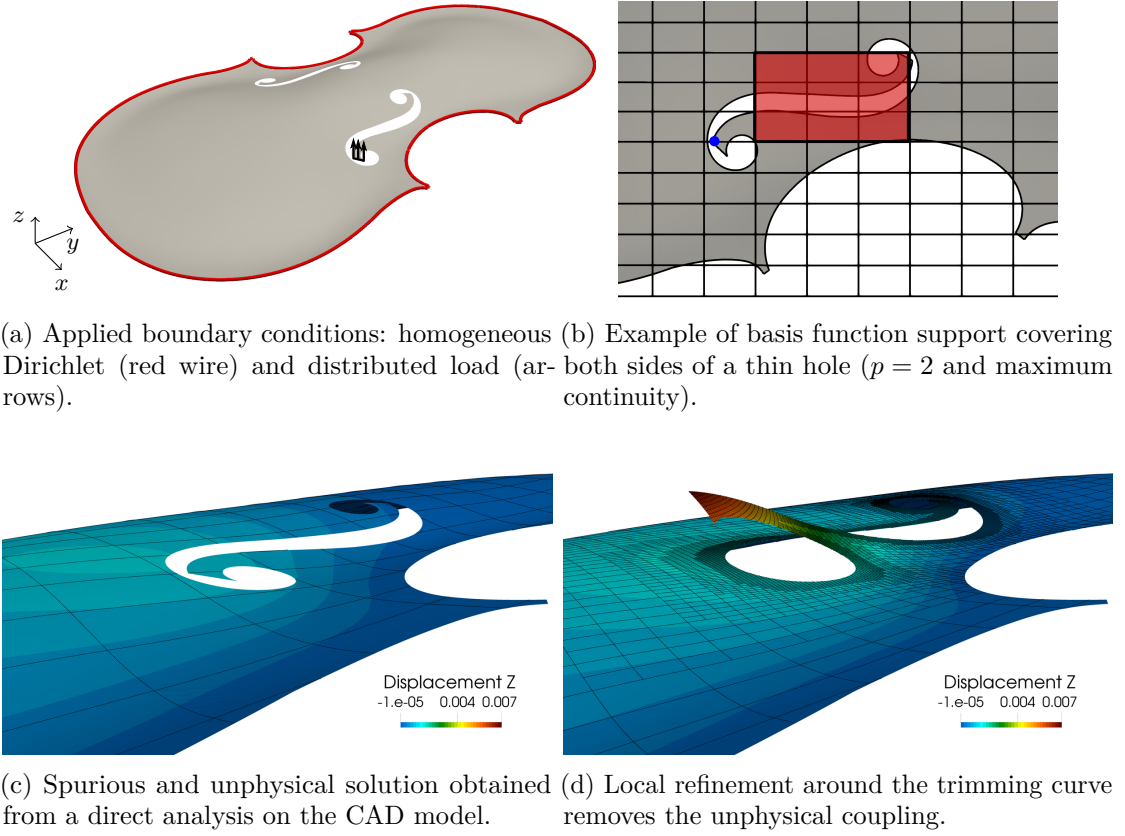


Figure 4.15 – Example problem on a complex trimmed geometry that shows the unphysical coupling occurring at the sides of thin holes.

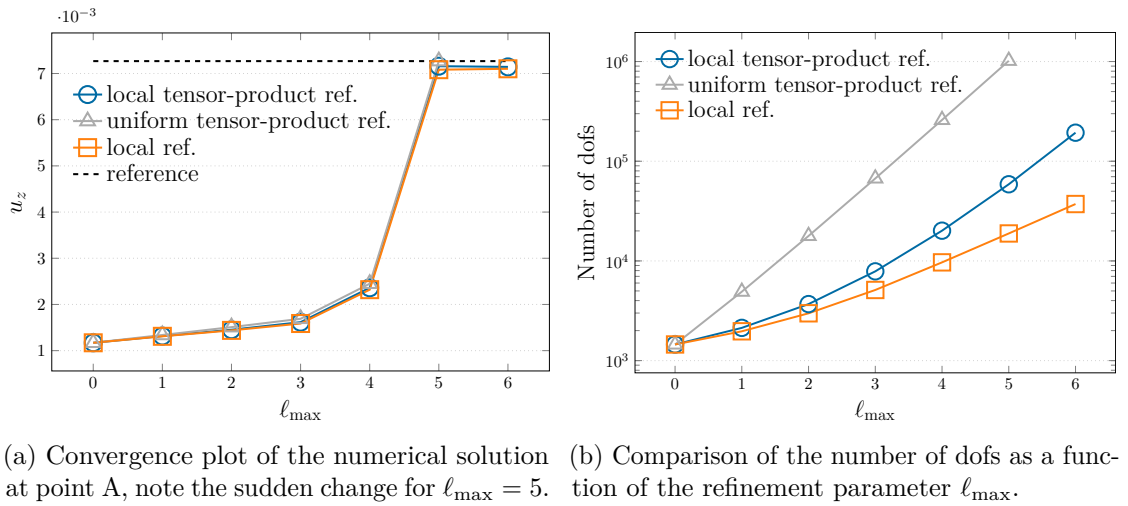


Figure 4.16 – Convergence graphs for the violin example subjected to a line load.

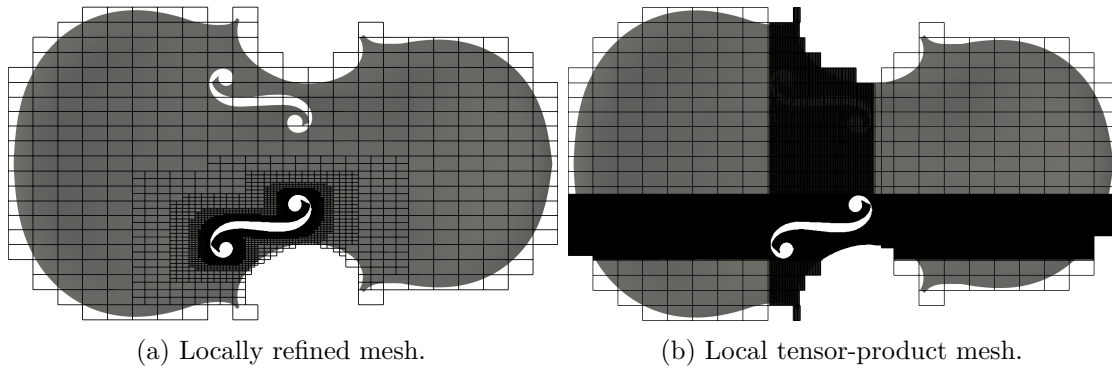


Figure 4.17 – Meshes obtained by local and local tensor-product refinements by setting $\ell_{\max} = 5$.

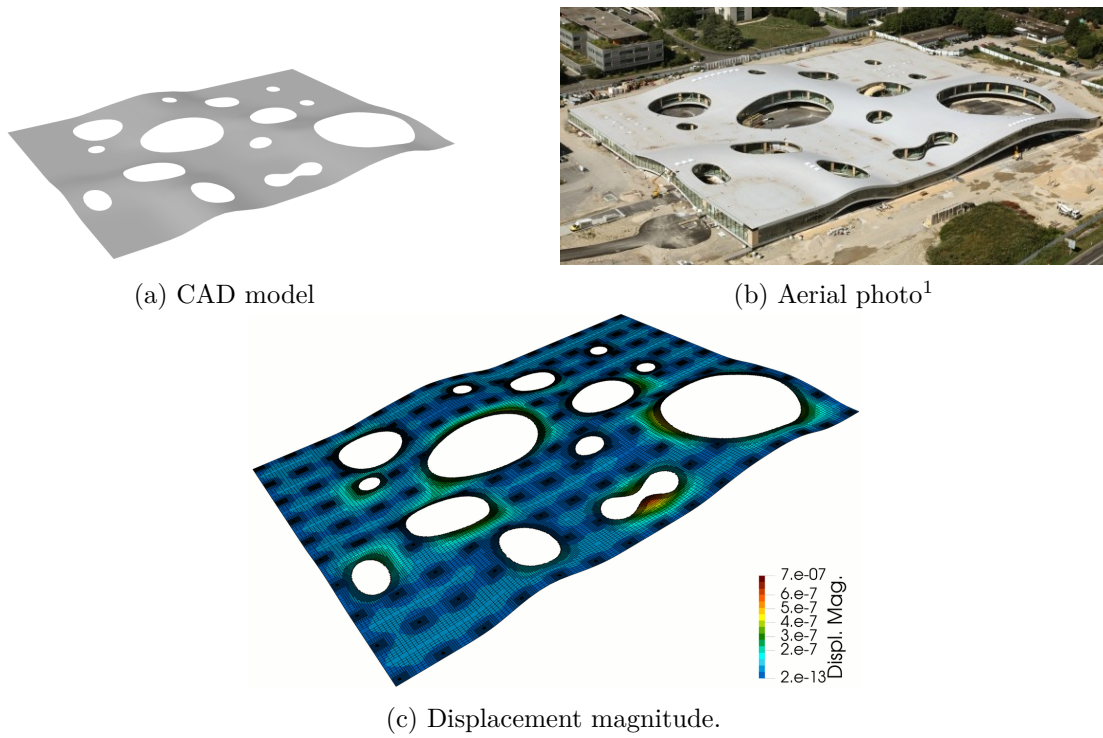
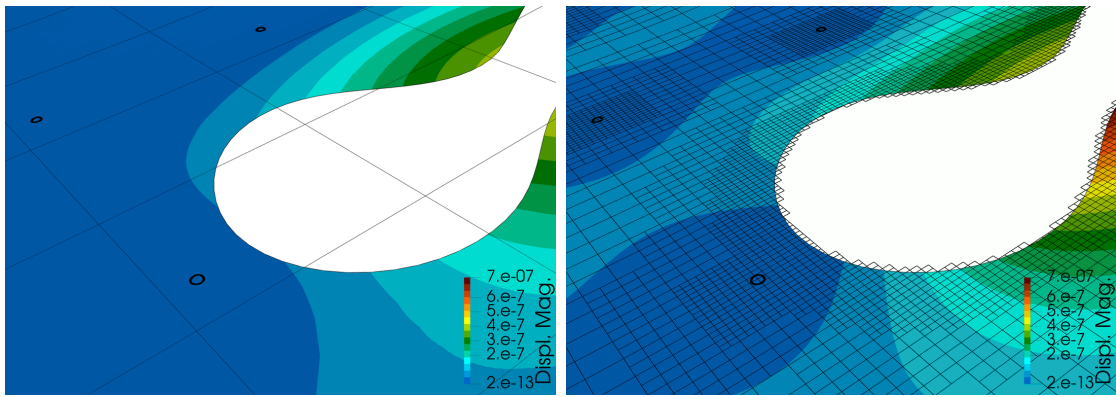
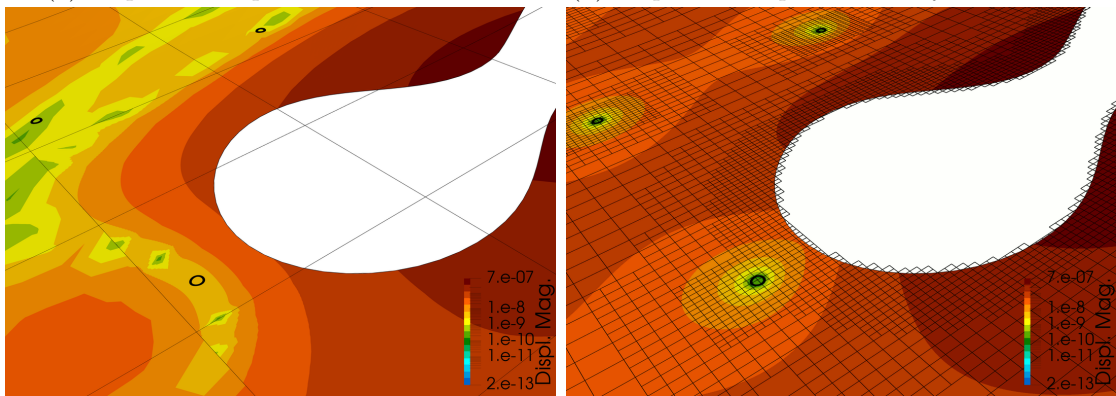


Figure 4.18 – Geometric model, actual building and solution of the Rolex Learning Center subject to its self-weight. For the solution hierarchical B-splines of degree $p = 3$ and $k = 5$ levels of refinement have been used. The geometry contains approximately 150 pillars modeled as the intersection between the untrimmed surface and cylinders directly in Rhinoceros.



(a) Displacement plot on a coarse mesh.

(b) Displacement plot on locally refined mesh.



(c) Displacement plot on a coarse mesh (logarithmic scale).

(d) Displacement plot on a locally refined mesh (logarithmic scale).

Figure 4.19 – Zoom on the solution in the proximity of a trimming curve for the Rolex Learning Center example. Note how weakly imposed boundary conditions and geometric features are efficiently resolved by local refinement.

5 *A-posteriori* error estimation

We have seen in Section 2.5 that there exist rigorous *a priori* error estimates for the finite element method. Although they constitute the mathematical foundations of FEM and are paramount for its success, they cannot be used in practical applications since the exact unknown solution \mathbf{u} appears on both side of the inequality in Equation (2.54). Finding a computable bound on the error falls into the realm of *a posteriori* error estimation. Indeed, the main objective of these techniques is to estimate the discretization error without having any knowledge of the exact solution. This question is clearly crucial in engineering applications, since for a given mesh we want to (approximately) know the level of accuracy of the corresponding FE analysis. In this chapter, we review the most common types of error estimators following the work in [Babuška, 1981; Ainsworth and Oden, 1997; Grätsch and Bathe, 2005; Verfürth, 2013] and using the equations of linear elasticity as model problem. Then, we propose a novel error estimator particularly suited for Kirchhoff plates and Kirchhoff-Love shells based on the evaluation of the residual equation in a weak sense. Indeed, the proposed method is based on the solution of an additional residual-like problem, formulated onto a so-called *bubble space*. We then proceed to summarize the concept of adaptivity in the scope of FEM. Here, we employ the procedure introduced in [Babuška and Vogelius, 1984; Babuška and Miller, 1987], where a *feedback* loop is described which allows to automatically refine the mesh until a convergence criterion is reached. In particular, starting from the initial discretization \mathcal{T}^1 and corresponding finite element solution \mathbf{u}_h^1 we can define the mesh update as:

$$\mathcal{T}^1 = \mathcal{A}(\mathcal{T}^1, \dots, \mathcal{T}^{i-1}, \mathbf{u}_h^1, \dots, \mathbf{u}_h^{i-1}), \quad (5.1)$$

where the *transition operator* \mathcal{A} formally describe the mesh refinement step. We can now state that an adaptive scheme is called convergent if its associated transition operator yields a sequence of numerical solutions such that:

$$\left\| \mathbf{u} - \mathbf{u}_h^i \right\|_{E(\Omega)} \rightarrow 0 \quad \text{for} \quad i \rightarrow \infty. \quad (5.2)$$

Lastly, a wide range of numerical experiments both on non-trimmed and trimmed geometries are presented to assess the performance of the proposed estimator in steering an adaptive loop.

5.1 Explicit error estimators

The starting point to derive this family of error estimators is the weak formulation of the error equation:

$$a(e_h, \mathbf{v}) = a(\mathbf{u}, \mathbf{v}) - a(\mathbf{u}_h, \mathbf{v}) = f(\mathbf{v}) - a(\mathbf{u}_h, \mathbf{v}) = \mathcal{R}_h(\mathbf{v}) \quad \forall \mathbf{v} \in V, \quad (5.3)$$

where $\mathcal{R}_h(\cdot)$ denotes the residual in a weak sense. We can now rewrite the error equation in an element-wise fashion as:

$$a(e_h, \mathbf{v}) = \sum_{K \in \mathcal{T}} \left(\int_K \mathbf{v} \cdot \mathbf{b} \, d\Omega + \int_{\partial K \cap \Gamma_N} \mathbf{v} \cdot \tilde{\mathbf{t}} \, d\gamma - \int_K \boldsymbol{\sigma}(\mathbf{u}) : \boldsymbol{\varepsilon}(\mathbf{v}) \, d\Omega \right) \quad \forall \mathbf{v} \in V, \quad (5.4)$$

where ∂K denotes the boundary of element K and we have expanded all the integrals involved in the right-hand-side of Equation (5.3). We can now perform integration by parts of the internal energy and rewrite Equation (5.4) as:

$$a(e_h, \mathbf{v}) = \sum_{K \in \mathcal{T}} \int_K R \mathbf{v} \, d\Omega + \sum_{e \in \partial \mathcal{T}} \int_e J \mathbf{v} \, d\gamma \quad \forall \mathbf{v} \in V, \quad (5.5)$$

where R denotes the residual in the interior of an element:

$$R = \nabla \cdot \boldsymbol{\sigma}(\mathbf{u}_h) + \mathbf{b}, \quad (5.6)$$

and J represents the jump of the stress across an element edge e :

$$J = \begin{cases} \llbracket \boldsymbol{\sigma}(\mathbf{u}_h) \cdot \mathbf{n} \rrbracket & \text{if } e \not\subset \Gamma \\ \tilde{\mathbf{t}} - \boldsymbol{\sigma}(\mathbf{u}_h) \cdot \mathbf{n} & \text{if } e \subset \Gamma_N \\ 0 & \text{if } e \subset \Gamma_D \end{cases}, \quad (5.7)$$

where with the notation $e \not\subset \Gamma$ we refer to those edges that are shared between elements K and K' . At this point we recall the Galerkin orthogonality:

$$a(e_h, \mathbf{v}_h) = \mathcal{R}_h(\mathbf{v}_h) = 0 \quad \forall \mathbf{v}_h \in V_h, \quad (5.8)$$

which allows us to rewrite Equation (5.5) as follows:

$$a(e_h, \mathbf{v}) = \sum_{K \in \mathcal{T}} \int_K R(\mathbf{v} - \mathcal{I}_h(\mathbf{v})) \, d\Omega + \sum_{e \in \partial \mathcal{T}} \int_e J(\mathbf{v} - \mathcal{I}_h(\mathbf{v})) \, d\gamma \quad \forall \mathbf{v} \in V, \quad (5.9)$$

where we have employed the Clément interpolant $\mathcal{J}_h(\cdot) : H^1 \rightarrow V_h$ [Clément, 1975]. Applying the Cauchy-Schwarz inequality element by element yields:

$$a(e_h, \mathbf{v}) \leq \sum_{K \in \mathcal{T}} \int_K \|R\|_{L^2(K)} \|\mathbf{v} - \mathcal{J}_h(\mathbf{v})\|_{L^2(K)} \, d\Omega + \sum_{e \in \partial \mathcal{T}} \int_e \|J\|_{L^2(e)} \|\mathbf{v} - \mathcal{J}_h(\mathbf{v})\|_{L^2(e)} \, d\gamma. \quad (5.10)$$

We can now introduce the following upper bounds on the error associated with the Clément interpolant:

$$\|\mathbf{v} - \mathcal{J}_h(\mathbf{v})\|_{L^2(K)} \leq C_1 h_K \|\mathbf{v}\|_{H^1(\tilde{K})}, \quad (5.11a)$$

$$\|\mathbf{v} - \mathcal{J}_h(\mathbf{v})\|_{L^2(\partial K)} \leq C_2 \sqrt{h_K} \|\mathbf{v}\|_{H^1(\tilde{K})}, \quad (5.11b)$$

where \tilde{K} denotes the patch associated with element K , formed by all neighboring elements with a common edge with K , and C_1 and C_2 are again constants independent of h_K . Substituting Equation (5.11) into Equation (5.10) yields:

$$a(e_h, \mathbf{v}) \leq \|\mathbf{v}\|_{H^1(\tilde{K})} \left[\sum_{K \in \mathcal{T}} C_1 h_K^2 \|R\|_{L^2(K)}^2 + \sum_{e \in \partial \mathcal{T}} C_2 h_K \|J\|_{L^2(e)}^2 \right]^{\frac{1}{2}}. \quad (5.12)$$

Using the equivalence of the H^1 and energy norms, and plugging e_h instead of \mathbf{v} , we obtain:

$$\|e_h\|_{E(\Omega)}^2 \leq \sum_{K \in \mathcal{T}} \left[C_1 h_K^2 \|R\|_{L^2(K)}^2 + C_2 h_K \|J\|_{L^2(\partial K)}^2 \right], \quad (5.13)$$

where the two constants C_1 and C_2 correspond to the element and jump residuals, respectively. Finally, we can introduce an element-wise error indicator as:

$$\|e_h\|_{E(\Omega)}^2 \leq \eta^2 = \sum_{K \in \mathcal{T}} \eta_K^2 \quad \text{where} \quad \eta_K^2 = C_1 h_K^2 \|R\|_{L^2(K)}^2 + C_2 h_K \|J\|_{L^2(\partial K)}^2, \quad (5.14)$$

which, except for constants C_1 and C_2 , provides a computable bound on the error. In the literature, some effort has been put into finding a way to estimate these constants, e.g. we refer to [Johnson and Hansbo, 1992; Stein et al., 1994]. However, while providing a sharp bound is an important property of an error estimator, we will see in the following that it is not crucial to steer the feedback loop. We can now introduce some desirable feature of the estimator η . In particular, we say that an estimator is *reliable* if there exists a constant C_1 such that:

$$\|e_h\|_{E(\Omega)} \leq \frac{1}{C_1} \eta. \quad (5.15)$$

Conversely, we characterize an estimator as *efficient* if there exists another constant C_2 such that:

$$\eta \leq C_2 \|e_h\|_{E(\Omega)} . \quad (5.16)$$

If η is both reliable and efficient then it holds:

$$C_1 \|e_h\|_{E(\Omega)} \leq \eta \leq C_2 \|e_h\|_{E(\Omega)} . \quad (5.17)$$

It is straightforward to see that in this case the estimator behaves asymptotically as the true error. Lastly, we introduce the so-called *effectivity index* as follows:

$$\theta = \frac{\sqrt{\sum_{K \in \mathcal{T}} \eta_K^2}}{\|e_h\|_{E(\Omega)}} , \quad (5.18)$$

which measures how well the estimator approximates the true error, where the optimal value of θ is one.

5.2 Implicit error estimators

Contrary to explicit estimators, implicit error estimators require the solution of an additional problem to provide an approximation on the error. To limit the computational burden of these schemes, the auxiliary problem is typically defined either at element level or over small patches to preserve the locality and increase the level of parallelism of the algorithm.

5.2.1 Element residual method

This family of estimators was first studied in [Demkowicz et al., 1984; Bank and Weiser, 1985]. The starting point in the derivation is again the weak formulation of the error, this time restricted to an element K :

$$a(e_h, \mathbf{v})|_K = \int_K R\mathbf{v} \, d\Omega + \int_{\partial K} \mathbf{v} \cdot [\boldsymbol{\sigma}(\mathbf{u}) \cdot \mathbf{n} - \boldsymbol{\sigma}(\mathbf{u}_h) \cdot \mathbf{n}] \, d\gamma \quad \forall \mathbf{v} \in V_K , \quad (5.19)$$

where the discrete space V_K over an element is defined as:

$$V_K = \{\mathbf{v} \in [H^1(K)]^3 : \mathbf{v} = \mathbf{0} \text{ on } \partial K \cap \Gamma_D\} . \quad (5.20)$$

Next, we need to distinguish three different cases:

- If $\partial K \cap \Gamma_D \neq \emptyset$ then we have a zero contribution on the error from the corresponding Dirichlet boundary condition.

- If $\partial K \cap \Gamma_N \neq \emptyset$ then we know the exact prescribed traction on the boundary from the corresponding Neumann boundary condition.
- Else, if ∂K is an interelement boundary, we approximate the exact traction as follows:

$$\boldsymbol{\sigma}(\mathbf{u}) \cdot \mathbf{n} \approx \bar{\boldsymbol{\sigma}}(\mathbf{u}_h) \cdot \mathbf{n} = \frac{1}{2} (\boldsymbol{\sigma}(\mathbf{u}_h)|_K + \boldsymbol{\sigma}(\mathbf{u}_h)|_{K'}) \cdot \mathbf{n}, \quad (5.21)$$

where K' denotes the neighbor of K sharing the edge under consideration.

Substituting this result into Equation (5.19) yields the following local problem: find $\psi_K \in V_K$ such that:

$$a(\psi_K, \mathbf{v})|_K = \int_K R\mathbf{v} \, d\Omega + \int_{\partial K \setminus \Gamma_N} \mathbf{v} \cdot [\bar{\boldsymbol{\sigma}}(\mathbf{u}_h) \cdot \mathbf{n} - \boldsymbol{\sigma}(\mathbf{u}_h) \cdot \mathbf{n}] \, d\gamma + \int_{\partial K \cap \Gamma_N} \mathbf{v} \cdot \tilde{\mathbf{t}} \, d\gamma \quad \forall \mathbf{v} \in V_K, \quad (5.22)$$

which can be solved for the local unknown ψ_K . With the latter, the error estimator is defined as:

$$\|e_h\|_{E(\Omega)}^2 \leq \sum_{K \in \mathcal{T}} \eta_K^2 \quad \text{where} \quad \eta_K^2 = \|\psi_K\|_{E(K)}^2. \quad (5.23)$$

The major drawback of this family of estimators is that the well-posedness of Equation (5.22) cannot be guaranteed due to the potential incompatibility of the Neumann data, which poses severe limitations to its applicability. Some remedies have been studied, where either the space V_K is suitably modified or the boundary data are carefully chosen to ensure compatibility, see for instance [Ainsworth and Oden, 1997] and references therein.

5.3 Recovery-based error estimators

The next family of error estimators is based on a post-processing step for the gradient of the numerical solution. The underlying idea is to obtain an enhanced approximation of the gradient, denoted in the following by $\nabla \mathbf{u}_h^*$, such that the true error can be approximated as:

$$\|e_h\|_{E(\Omega)}^2 \approx \int_{\Omega} [\nabla \mathbf{u}_h^* - \nabla \mathbf{u}_h]^2 \, d\Omega. \quad (5.24)$$

In the following we briefly present a popular scheme to obtain a better approximation of the numerical gradient, firstly introduced in [Zienkiewicz and Zhu, 1987]. Let us write the enhanced gradient as a linear combination of coefficients and basis functions $\phi \in V_h$

such that:

$$\nabla \mathbf{u}_h^* = \sum_{i=1}^{\dim(V_h)} (\nabla \mathbf{u}_h^*)_i \phi_i, \quad (5.25)$$

where the unknown coefficients $(\nabla \mathbf{u}_h^*)_i$ are given by:

$$\int_{\Omega} \phi_j (\nabla \mathbf{u}_h^* - \nabla \mathbf{u}_h) \, d\Omega = 0 \quad j = 1, \dots, \dim(V_h). \quad (5.26)$$

Expanding the integrand on the left-hand-side and rearranging the terms yields the standard linear system associated with the L^2 -projection:

$$\sum_{i=1}^{\dim(V_h)} \int_{\Omega} \phi_i \phi_j \, d\Omega (\nabla \mathbf{u}_h^*)_i = \int_{\Omega} \phi_j \nabla \mathbf{u}_h \, d\Omega \quad j = 1, \dots, \dim(V_h). \quad (5.27)$$

Once the unknown coefficients are found, the error estimator is defined element-wise as:

$$\|e_h\|_{E(\Omega)}^2 \approx \sum_{K \in \mathcal{T}} \eta_K^2 \quad \text{where} \quad \eta_K^2 = \|\nabla \mathbf{u}_h^* - \nabla \mathbf{u}_h\|_{L^2(K)}^2. \quad (5.28)$$

Clearly, the performance of this family of error estimators relies heavily on the assumption that $\nabla \mathbf{u}_h^*$ provides a better representation of the true gradient $\nabla \mathbf{u}$. For further details on how to achieve this property, we refer to [Zienkiewicz and Zhu, 1992a,b; Carstensen and Funken, 2001].

Remark 5.1 *A vast amount of literature is dedicated to the study of error estimators. In the presentation above, only a brief overview of the most relevant families of global a posteriori error estimators is provided. We highlight that often in engineering applications, one is interested in local quantities rather than global errors in the energy norm. This is the realm of research of the so-called goal-oriented estimators, where for an in-depth discussion on the subject the reader is referred to [Oden and Prudhomme, 2001; Grätsch and Bathe, 2005, 2006] and references therein.*

5.4 A novel error estimator for Kirchhoff plates and Kirchhoff-Love shells

In the following, we introduce a variant of the family of implicit error estimators studied in [Bank and Smith, 1993] in the context of the p -version of FEM and we extend its isogeometric version, successfully applied to T-splines in [Dörfel et al., 2010] and later to hierarchical B-splines in [Vuong et al., 2011], to elliptic second- and fourth-order PDEs defined on trimmed domains. Then, we present a possible implementation of the proposed indicator which makes use of truncated hierarchical B-splines. Let us now review the method and formulate it into the framework of trimmed hierarchical IGA. Let

5.4. A novel error estimator for Kirchhoff plates and Kirchhoff-Love shells

us define the finite dimensional solution space \tilde{V}_h^p over a trimmed domain as the span of \mathcal{THB} -splines basis functions of order p , as defined in Section 4.3. Then, let us denote by $u_h \in \tilde{V}_h^p$ the discrete solution of the problem at hand. Now, starting from the degree and continuity of the aforementioned space \tilde{V}_h^p and for every level ℓ , let us introduce the B-splines of degree $p+1$ and of reduced continuity, denoted as $\mathcal{B}_{red}^{p+1,\ell}$, defined on \mathcal{Q}_0^ℓ (the rectangular grid of level ℓ introduced in Section 4.3), where the chosen continuity depends on the problem at hand (see Remark 5.3). From an algorithmic point of view, this space is obtained by performing first degree elevation followed by increasing the multiplicity of the original knots. Now, from all the functions in $\mathcal{B}_{red}^{p+1,\ell}$, we select only a suitable subset $\tilde{\mathcal{B}}_{act}^{p+1,\ell}$, which we characterize as:

$$\tilde{\mathcal{B}}_{act}^{p+1,\ell} = \{b \in \mathcal{B}_{red}^{p+1,\ell} \mid \text{supp}(b) \subset K \text{ for some } K \in \hat{\mathcal{T}}_A^\ell\}. \quad (5.29)$$

In the following, we refer to these splines as *bubble functions*, where an example defined on a single level is depicted in Figure 5.1. We remark that by definition all functions in $\tilde{\mathcal{B}}_{act}^{p+1,\ell}$ have support on a single element. In fact, the bubble functions on each element always coincide with a subset of (scaled) Bernstein polynomials, and as a consequence they can be defined on a single reference element, which is then mapped to the elements of the mesh, analogously to basis functions in finite elements. We remark that this reference element is independent of the level in the hierarchy. Then, let assume there exists a larger space $\tilde{V}_h^p \subset \tilde{W}_h^p \subset V$ such that the following decomposition holds:

$$\tilde{W}_h^p = \tilde{V}_h^p \oplus \tilde{Z}_h^{p+1}, \quad (5.30)$$

where \tilde{Z}_h^{p+1} is the space (defined over a trimmed geometry) in which we seek a good estimation of the error $e_h \approx e = \|u - u_h\|$, in a suitable norm $\|\cdot\|$. In our case, we use the energy norm associated with the problem. In particular, we can characterize \tilde{Z}_h^{p+1} in a multi-level fashion as follows:

$$\tilde{Z}_h^{p+1} = \bigcup_{\ell=0}^N \tilde{Z}_{h,\ell}^{p+1}, \quad (5.31)$$

where

$$\tilde{Z}_{h,\ell}^{p+1} = \text{span} \{b \circ \mathbf{F}^{-1} \mid b \in \tilde{\mathcal{B}}_{act}^{p+1,\ell}\}. \quad (5.32)$$

Namely, $\tilde{Z}_{h,\ell}^{p+1}$ is the space spanned by active B-splines of level ℓ obtained by degree elevation and knot insertion as discussed above, such that their support is compact and overlaps exactly with one active element (trimmed or non-trimmed) K of level ℓ , where we postpone the discussion on the required continuity of b to Remark 5.3. We highlight that, in the trimmed case, we must replace the standard definition of support of a function with its trimmed counterpart, which was previously denoted as $\text{supp}_{\hat{\Omega}}(\cdot)$. We are now ready to define the a posteriori error estimate e_h as the solution to the following

problem: find $e_h \in \tilde{Z}_h^{p+1}$ such that

$$a(e_h, b_h) = F(b_h) - a(u_h, b_h) \quad \forall b_h \in \tilde{Z}_h^{p+1}. \quad (5.33)$$

Notice that due to the compact support property of b_h , the linear system corresponding to the discrete error weak form in Equation (5.33) is block diagonal, where each block corresponds to a single element, and its size is given by the number of bubble functions. Lastly, we compute the element-wise error estimator η_K as:

$$\eta_K = C_a \|e_h\|_{E(K)} \quad \forall K \in \mathcal{T}, \quad (5.34)$$

where $\|\cdot\|_{E(K)}$ denotes the energy norm restricted to an element K of the hierarchical mesh \mathcal{T} .

Remark 5.2 *At this stage, we introduce the heuristic constant C_a which, considering a wide variety of numerical experiments, seems to be independent from the chosen degree and from the problem at hand. In the scope of this thesis, we set the value $C_a = 3$. This ensures that the true error is not underestimated by our indicator η .*

It is worth highlighting that up to this point the derivation is completely independent from the problem we are solving, in the sense that once bilinear and linear forms and a suitable bubble space are chosen, the methodology follows the same aforementioned steps. We will demonstrate this over a variety of elliptic PDEs in our numerical examples. Moreover, trimming can be treated naturally in this framework at integration level. This confirms the simplicity of the proposed estimator on trimmed domains and that the method can be implemented on top of existing isogeometric codes in a straightforward manner, e.g. see the pseudo-code summarized in Algorithm 3. Furthermore, our technique is computationally cheap and embarrassingly parallelizable due to the definition of the disjoint bubble space \tilde{Z}_h^{p+1} and the choice of solving Equation (5.33) level-wise over the hierarchical mesh \mathcal{T} . Indeed, we can summarize the properties of the error estimator as follows:

1. the associated linear system possesses a block-diagonal structure due to the single element support of bubble functions,
2. small dimension of each block, that are defined element-wise,
3. computation is easily parallelizable.

All the aforementioned features make the proposed method appealing from a computational point of view.

5.4. A novel error estimator for Kirchhoff plates and Kirchhoff-Love shells

Algorithm 3 Bubble error estimator algorithm

```

1: procedure ESTIMATE_ERROR(numerical solution  $u_h$ , hierarchical mesh  $\mathcal{T}$ )
2:   Initialize vector  $\eta$ 
3:   for each level  $\ell$  of  $\mathcal{T}$  do
4:     Build bubble space  $\tilde{Z}_{h,\ell}^{p+1}$ 
5:     for each  $K \in \mathcal{T}_A^\ell$  do
6:       Given  $u_h$ , solve locally on element  $K$  the additional system in Equation (5.33)
7:       Compute the element-wise indicator  $\eta_K$  from Equation (5.34)
8:       Store  $\eta_K$  in  $\eta$ 
9:     end for
10:  end for
11:  Return  $\eta$ 
12: end procedure

```

A known drawback of this family of estimators is that their reliability and efficiency is subjected to the saturation assumption onto the underlying augmented space. As noted in [Bank and Smith, 1993], this assumption is, in general, problem dependent and it can, potentially, become crucial for the performance of the estimator. However, we highlight that our focus is to find a good indicator for steering adaptive simulations in the scope of structural mechanics and all our numerical experiments confirm that the proposed estimator performs well in this task. Finally, we conclude our discussion on the proposed error estimator with a series of important remarks.

Remark 5.3 (On the choice of the bubble space) *We note that the choice of the bubble space \tilde{Z}_h^{p+1} has to be compatible with the underlying bilinear form. Therefore, for every level ℓ , we select the continuity of $\mathcal{B}_{red}^{p+1,\ell}$ such that all functions are at least C^0 -continuous globally for second order problems (such as linear elasticity and the Poisson problem) and at least C^1 -continuous for fourth order problems (e.g. Kirchhoff plates and Kirchhoff-Love shell). This ensures that the bilinear form in Equation (5.33) is well-defined.*

Remark 5.4 (Inhomogeneous boundary conditions) *On one hand, we highlight that whenever inhomogeneous boundary conditions of Neumann-type are applied to the problem at hand, Equation (5.33) must also contain the corresponding additional terms in the right-hand-side. This is also true if we apply Dirichlet-type boundary conditions weakly. In these cases, we augment the bubble space \tilde{Z}_h^{p+1} with suitable boundary bubbles to properly capture the error on the imposition of the corresponding boundary conditions. These functions must either have non-zero value or non-zero derivatives at the boundary. An example is depicted in Figure 5.2 for C^1 -continuous bubbles of degree $p = 4, 5$, where the resulting space captures the contribution of inhomogeneous boundary conditions on*

the bending moments and/or weakly enforced rotations. On the other hand, we make the assumption that the error on the strong imposition of inhomogeneous Dirichlet-type boundary conditions is negligible and therefore no additional shape functions are needed for these cases.

Remark 5.5 (The case $p = 2$ for C^1 -continuous bubble spaces) *For the sake of completeness, we should highlight that the construction of the bubble space \tilde{Z}_h^{p+1} is not straightforward in the case where the underlying B-splines are quadratic. We propose as possible remedies:*

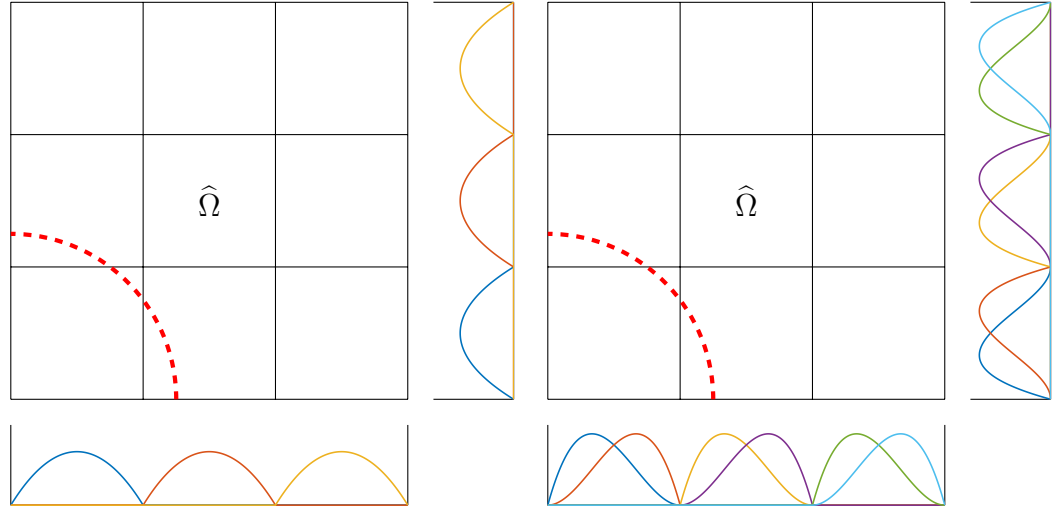
1. *Construction of an additional sub-grid obtained as dyadic refinement of the original hierarchical mesh and definition of the bubble functions on the corresponding space defined by the new grid.*
2. *Introduction of suitable edge functions in the bubble space.*

Notice that, although both approaches could be feasible, they add a non-negligible complexity to the method. In the first case, a new, finer hierarchical grid has to be created at every iteration of the adaptive loop and numerical integration of (5.33) has to be performed over it. In the second case, the definition of which edge bubbles have to be considered in the hierarchical case is not unique and considerably worsen the simplicity of the method. Indeed, the corresponding linear system would become more difficult to solve since the element-wise locality of the bubble functions would be lost. Therefore, in light of these considerations, we will not address further bubble functions of degree $p = 2$ and C^1 -continuity in the scope of this chapter.

Remark 5.6 (Jumps in the residual for fourth-order PDEs) *In the last remark, we notice that, although the residual exhibits jumps across element edges for the discretization of degree $p = 3$, in all our computations the estimator behaves still optimally even if the corresponding jump terms are not accounted for in (5.33). Therefore, from a practical standpoint, the estimator retains its favorable properties.*

5.5 Adaptivity

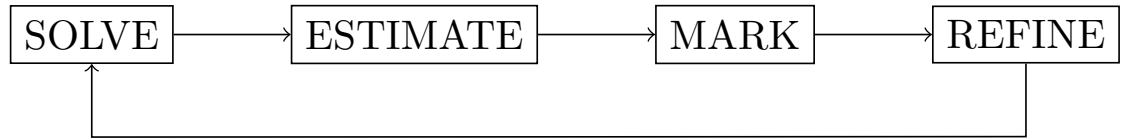
We have seen that, once an element-wise estimation of the error is available, there exists a feedback loop that refines the mesh locally towards the regions where the error is bigger. Here, we summarize the basic concepts needed in our derivation, where for a recent review of the fundamental concepts underlying adaptive finite element methods we refer to [Nochetto et al., 2009; Nochetto and Veiser, 2012] and reference therein. Following the



(a) Bubble basis $p + 1 = 2$ associated with an univariate knot vector $\Xi = [0001122333]$ in each parametric direction. (b) Bubble basis $p + 1 = 3$ associated with an univariate knot vector $\Xi = [00001112223333]$ in each parametric direction.

Figure 5.1 – Example of construction of C^0 bubble functions on a trimmed parametric domain as tensor product of univariate functions for the cases $p + 1 = 2, 3$. The corresponding space is suitable for the error estimation of the solution of second-order PDEs. We remark that we tackle trimming at integration level by reparametrizing those elements that are cut.

notation in [Buffa and Giannelli, 2017], we split an adaptive algorithm into the following main four steps:



We have already reviewed how to obtain an approximate solution and a suitable error estimator for the problem at hand. Let us briefly characterize the remaining steps in the following.

5.5.1 Mark and refine

Following the standard procedure in adaptivity, once the element-wise error η_K has been computed for all elements of the (trimmed) hierarchical mesh \mathcal{T} , we mark elements for refinement. Several strategies have been proposed in the literature, e.g. see [Babuška and Rheinboldt, 1978; Johnson, 1990], where the predominant ones are:

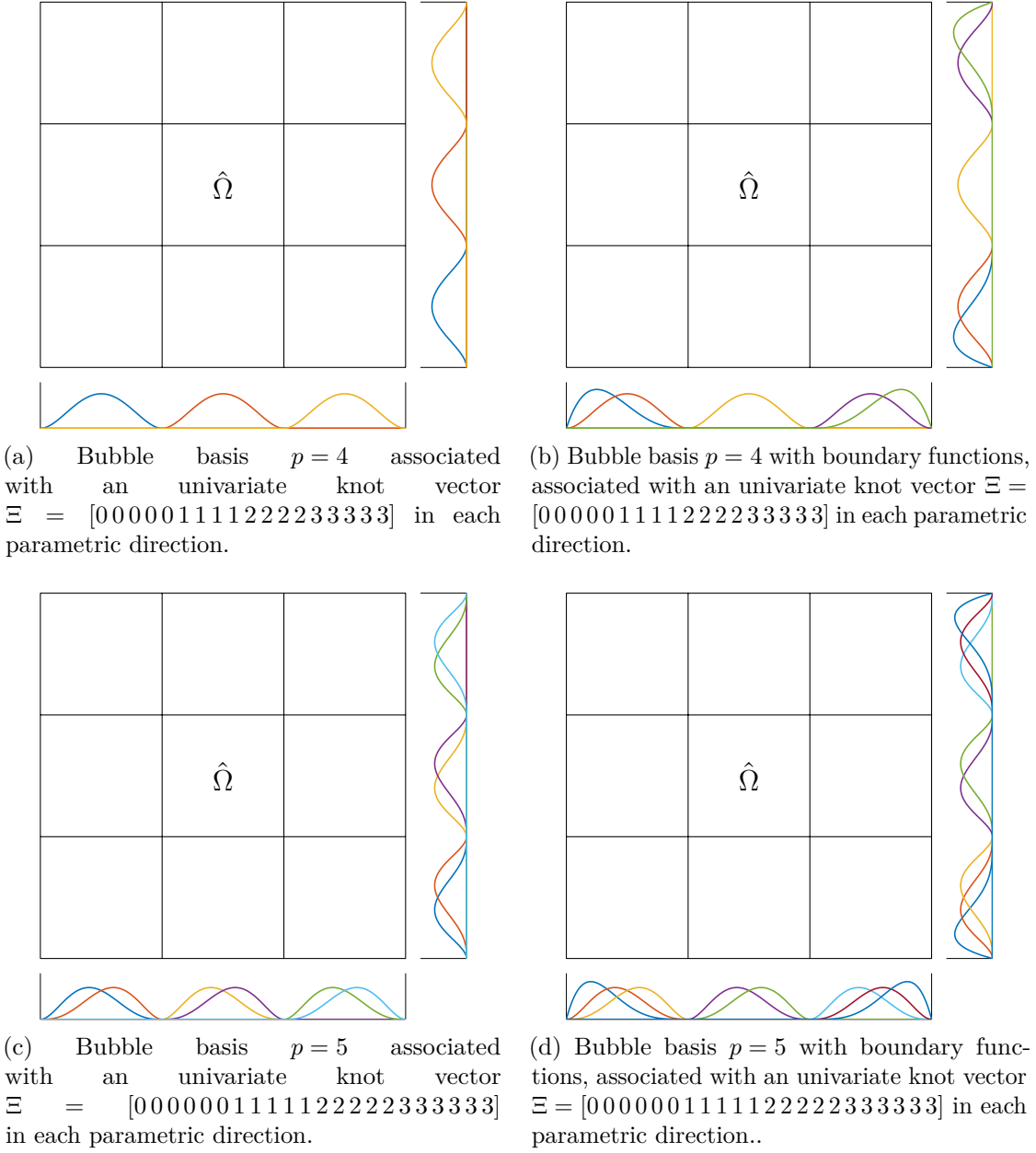


Figure 5.2 – Example of construction of C^1 bubble functions as tensor product of univariate functions for the case $p + 1 = 3, 4$, where on the left-side only internal functions are depicted whereas on the right-side also boundary functions are plotted. The corresponding space is suitable for the error estimation of the solution of fourth-order PDEs.

1. Maximum strategy, which given a user-defined threshold $\gamma \in (0, 1)$, marks for refinement all elements such that:

$$\eta_K > \gamma \tilde{\eta}, \quad \text{where} \quad \tilde{\eta} = \max_{K' \in \mathcal{T}} \eta_{K'}. \quad (5.35)$$

2. Dörfler strategy [Dörfler, 1996], which given a user-defined parameter $\gamma \in (0, 1)$, selects for refinement a set of elements $\mathcal{M} \subseteq \mathcal{T}$ such that:

$$\eta_{\mathcal{M}} \geq (1 - \gamma)\eta, \quad (5.36)$$

where the set \mathcal{M} is constructed by taking sequentially the elements with the largest error contribution until the criterion above is met. This strategy guarantees error reduction during the adaptive loop.

Unless stated otherwise, in our numerical experiments we use the maximum strategy and we set $\gamma = 0.5$. Once the set of marked elements \mathcal{M} is available, the mesh \mathcal{T} needs to be refined accordingly, see for instance [Morin et al., 2002]. In case of \mathcal{HB} and \mathcal{THB} splines, this step will be discussed further in the following.

5.5.2 Admissible refinement of \mathcal{T}

Before actually performing refinement of the corresponding basis, our refine module is designed to preserve the so-called *admissibility* (as defined in [Buffa and Giannelli, 2017]) of the hierarchical mesh \mathcal{T} between consecutive iterations of the adaptive procedure. Let us introduce the following definition:

Definition 5.1 ([Buffa et al., 2016]) *A mesh \mathcal{T} is \mathcal{HB} -, respectively \mathcal{THB} -, admissible of class m if the (truncated) hierarchical basis functions in $\mathcal{HB}(\mathcal{T})$ (respectively $\mathcal{THB}(\mathcal{T})$) which take non-zero value over any element $K \in \mathcal{T}$ belong to at most m consecutive levels.*

This step guarantees to keep the number of basis functions acting on any element of the mesh bounded. Indeed, this bound states that the number of non-zero splines acting on any element is smaller than $m \prod_{i=1}^d (p_i + 1)$. From a practical point of view, this means that we prevent the interaction between functions belonging to very fine and very coarse levels. From an algorithmic standpoint, this property is guaranteed by the following refinement module. Let us first define the notion of support extension for hierarchical meshes.

Definition 5.2 ([Bracco et al., 2018]) *The multilevel support extension $S(\mathcal{T}, k)$ of an element $K \in \mathcal{T}_A^\ell$ with respect to level k , with $0 \leq k \leq \ell$, is defined as:*

$$S(\mathcal{T}, k) = \{K' \in \mathcal{T}_A^k : \exists b \in \mathcal{B}^k, \text{supp}(b) \cap K' \neq \emptyset \wedge \text{supp}(b) \cap K \neq \emptyset\}. \quad (5.37)$$

Remark 5.7 *We highlight that in the trimmed case, this definition should be modified by taking $\text{supp}_\Omega(\cdot)$ instead of $\text{supp}(\cdot)$. For the sake of simplicity, we derive the algorithm*

using the non-trimmed notation, but all results readily apply to the trimmed case by employing the corresponding proper modifications discussed in Section 4.3.3.

Then, let us also introduce, for $\ell = 0, \dots, N - 1$, the following sub-domains:

$$\omega_{\mathcal{HB}}^\ell = \bigcup \left\{ \bar{K} : K \in \mathcal{T}_A^\ell \wedge S(\mathcal{T}, \ell - 1) \subseteq \Omega^\ell \right\} \quad (5.38a)$$

$$\omega_{\mathcal{THB}}^\ell = \bigcup \left\{ \bar{K} : K \in \mathcal{T}_A^\ell \wedge S(\mathcal{T}, \ell) \subseteq \Omega^\ell \right\}. \quad (5.38b)$$

With these additional sub-domains we can introduce a new definition of admissibility.

Definition 5.3 ([Bracco et al., 2018]) *A mesh \mathcal{T} is strictly \mathcal{HB} -, respectively \mathcal{THB} -, admissible of class m if it holds that:*

$$\Omega^\ell \subseteq \omega_{\mathcal{HB}}^{\ell-m+1}, \quad \Omega^\ell \subseteq \omega_{\mathcal{THB}}^{\ell-m+1}, \quad \text{respectively,} \quad (5.39)$$

for $\ell = m, m + 1, \dots, N - 1$.

Lastly, we need a definition of neighborhood tailored to hierarchical splines, which reads:

Definition 5.4 ([Bracco et al., 2018]) *Given an element $K \in \mathcal{T} \cap \mathcal{T}_A^\ell$, its \mathcal{HB} - and \mathcal{THB} -neighborhoods with respect to m are respectively defined as:*

$$\mathcal{N}_{\mathcal{HB}}(\mathcal{T}, K, m) = \left\{ K' \in \mathcal{T}_A^{\ell-m+1} : K' \in S(K, \ell - m + 1) \right\}, \quad (5.40a)$$

$$\mathcal{N}_{\mathcal{THB}}(\mathcal{T}, K, m) = \left\{ K' \in \mathcal{T}_A^{\ell-m+1} : \exists K'' \in S(K, \ell - m + 2), K'' \subseteq K' \right\}, \quad (5.40b)$$

when $\ell - m + 1 \geq 0$, and $\mathcal{N}_{\mathcal{HB}}(\mathcal{T}, K, m) = \mathcal{N}_{\mathcal{THB}}(\mathcal{T}, K, m) = \emptyset$ for $\ell - m + 1 < 0$.

With these definitions at hand, we summarize in Algorithms 4 and 5 a procedure that, given as input a strictly admissible hierarchical mesh of class m and the set of marked elements \mathcal{M} , preserves the (strict) admissibility of the latter during refinement.

Algorithm 4

$\mathcal{T} = \text{REFINE}(\mathcal{T}, \mathcal{M}, m)$

```

1: for each element  $K \in \mathcal{T} \cap \mathcal{M}$  do
2:    $\mathcal{T} = \text{REFINE\_RECURSIVE}(\mathcal{T}, K, m)$ 
3: end for
4: Return  $\mathcal{T}$ 

```

Algorithm 5

$\mathcal{T} = \text{REFINE_RECURSIVE}(\mathcal{T}, K, m)$

```

1: for each element  $K' \in \mathcal{N}(\mathcal{T}, K, m)$  do
2:    $\mathcal{T} = \text{REFINE\_RECURSIVE}(\mathcal{T}, K', m)$ 
3: end for
4: if  $K$  has not been subdivided then
5:   Subdivide  $K$  and update  $\mathcal{T}$  by replacing
      $K$  with its children
6: end if

```

In our numerical examples, we will set the class of admissibility m to be $m = p$ and $m = p - 1$ for second-order and fourth-order PDEs, respectively. We refer the reader

to [Buffa et al., 2016; Buffa and Giannelli, 2017; Bracco et al., 2018, 2019] for a detailed mathematical derivation of the concept of admissibility and its application to adaptive refinement.

The treatment of isolated elements

Finally, we highlight that if we employ a standard mark-by-element strategy, we cannot ensure an improvement of the solution space for every step of the feedback loop since, to achieve that, we should either mark p neighbors in each parametric direction or employ a mark-by-function strategy [Buffa and Garau, 2018]. In particular, isolated refined elements do not improve the solution space \tilde{Z}_h^{p+1} (they do not add any additional degree of freedom) but only the associated numerical integration is refined. To mitigate this, we slightly modify the maximum strategy marking algorithm such that once an element K of level ℓ is marked for refinement, the algorithm also marks all the direct neighbors of $K \in \mathcal{T}_A^\ell$. This approach alleviates the issue while maintaining the locality of the refinement and providing a classical element-wise point of view.

5.6 Numerical results

We have highlighted in Section 4.5 some of the benefits of using local refinement in the analysis. Here, we show the advantage of using local refinement in combination with an error indicator to achieve a fully automatic adaptive loop. Indeed, in the following, we assess the performance of the proposed bubble error estimator in steering an adaptive simulation on a wide range of second- and fourth-order elliptic PDEs, namely the Poisson equation, linear elasticity and Kirchhoff plates and Kirchhoff-Love shells, respectively. For our benchmarks, we consider both non-trimmed and trimmed geometries. Additionally, we demonstrate the applicability of our method both for problems which exhibit smooth and singular solutions. In all cases presented here, the proposed error estimator shows excellent performance in steering the adaptive simulation, yielding the expected optimal rates of convergence in the asymptotic regime. Furthermore, the estimator provides an excellent approximation of the true error.

Remark 5.8 *In the trimmed case, to reduce the detrimental effects of trimming on the conditioning number, we apply a simple diagonal scaling, which can be seen as a Jacobi preconditioner. Specifically, we apply the preconditioner symmetrically:*

$$DKD\tilde{\mathbf{x}} = D\mathbf{f} \quad \text{with} \quad \tilde{\mathbf{u}} = D\tilde{\mathbf{x}}, \quad (5.41)$$

where D denotes the diagonal scaling preconditioner given by:

$$D = \text{diag} \left(\frac{1}{\sqrt{K_{11}}}, \frac{1}{\sqrt{K_{22}}}, \dots, \frac{1}{\sqrt{K_{nn}}} \right), \quad (5.42)$$

where $K_{ii}, i = 1, \dots, n$ represent the diagonal entries of the stiffness matrix. This seems to suffice in all the following numerical experiments. However, we are aware of the potential limitations of this choice of preconditioner, especially when extremely small and slender trimmed elements are present. Several possible remedies have been proposed in the literature, for instance the ghost penalty method [Burman and Hansbo, 2012; Burman et al., 2015], the extended B-spline method [Höllig et al., 2012; Marussig et al., 2017, 2018], CutIGA [Elfverson et al., 2018] and preconditioners tailored to trimmed isogeometric analysis, e.g. [de Prenter et al., 2017; de Prenter et al., 2020a]. For a more detailed discussion of the conditioning issues stemming from trimming we refer to [Marussig and Hughes, 2018; de Prenter, 2019] and references therein.

5.6.1 The Poisson problem

We start our numerical investigation from the well-known Poisson equation. Let us briefly recall the strong form of the problem at hand, which describes the equilibrium of a temperature-like scalar field u as:

$$\begin{aligned} -\Delta u &= \tilde{f} \quad \text{in } \Omega \\ u &= \tilde{u} \quad \text{on } \Gamma_D \\ \frac{\partial u}{\partial n} &= \tilde{g} \quad \text{on } \Gamma_N, \end{aligned} \tag{5.43}$$

where $\Gamma_D \subset \partial\Omega \cap \partial\Omega_0$ and Γ_N denote the Dirichlet and Neumann part of the boundary $\partial\Omega$, respectively, and it holds $\bar{\Gamma}_D \cup \bar{\Gamma}_N = \partial\Omega$ and $\Gamma_D \cap \Gamma_N = \emptyset$. Additionally, $\tilde{f} \in L^2(\Omega)$ represents the given source term, and $\tilde{u} \in H^{\frac{1}{2}}(\Gamma_D)$ and $\tilde{g} \in H^{-\frac{1}{2}}(\Gamma_N)$ the prescribed Dirichlet and Neumann data, respectively. Following the standard Galerkin derivation, we can write the discrete weak formulation of Equation (5.43) as follows: find $u_h \in \tilde{V}_h$ such that

$$a(u_h, v_h) = F(v_h) \quad \forall v_h \in \tilde{V}_h. \tag{5.44}$$

The bilinear form $a(\cdot, \cdot)$ and the linear functional $F(\cdot)$ can be expanded as:

$$a(u_h, v_h) = \int_{\Omega} \nabla v_h \cdot \nabla u_h \, d\Omega \tag{5.45a}$$

$$F(v_h) = \int_{\Omega} \tilde{f} v_h \, d\Omega + \int_{\Gamma_N} \tilde{g} v_h \, d\Gamma, \tag{5.45b}$$

respectively. We remark that this problem requires C^0 -continuity across elements for its well-posedness.

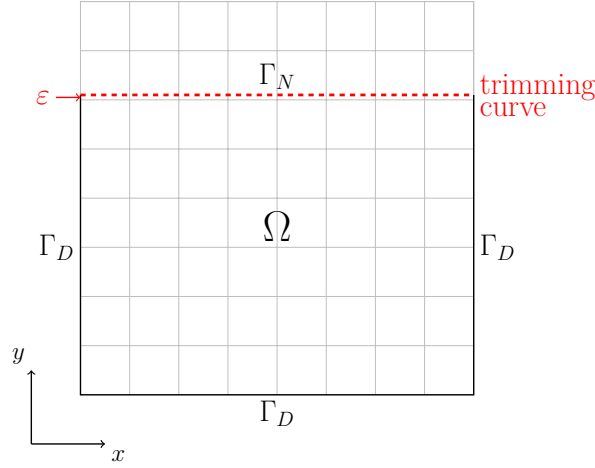


Figure 5.3 – Problem description for the Poisson singular problem.

Singular problem, line singularity

In this problem, we consider a trimmed computational domain defined by $\Omega = [0, 1] \times [0, \frac{3}{4} + \varepsilon]$, as depicted in Figure 5.3, where $\varepsilon = 10^{-5}$ is chosen. As exact solution, we take a singular function of the form $u = x^\alpha y^\alpha$ with $\alpha = 2.4$. The imposed boundary conditions are also given in Figure 5.3 and are computed such that they fulfill the exact solution, as well as the given source term \tilde{f} . We remark that with this choice of parameter it holds $u \in H^2(\Omega) \setminus H^3(\Omega)$. From the classical *a priori* error analysis results presented in Section 2.5, it is known that this reduction in regularity hinders the rate of convergence achievable with uniform refinement of the mesh. For our choice of α , optimal rates of convergence are achieved in case of uniform refinement only for bi-quadratic B-splines whereas higher degree discretizations suffer from the lack of regularity, see Figure 5.4a. As shown in Figure 5.4b, it is clear that optimal rates of convergence are recovered for all degrees $p = 2, 3, 4$ with an adaptive simulation driven by the proposed estimator. Moreover, this example highlights the benefits of using an adaptive strategy in terms of accuracy per degree-of-freedom.

5.6.2 Linear elasticity

Next, we apply the error estimator to the problem of linear elasticity, which we reviewed in Section 2.1. Analogously to the Poisson problem, global C^0 -continuity of the basis is required and, consequently, this constraint is also built into the bubble space \tilde{Z}_h^{p+1} .

Infinite plate with a hole

In this example, we analyze the well-known benchmark of an infinite plate with a circular hole. The geometry and problem setup are depicted in Figure 5.5, where thanks to

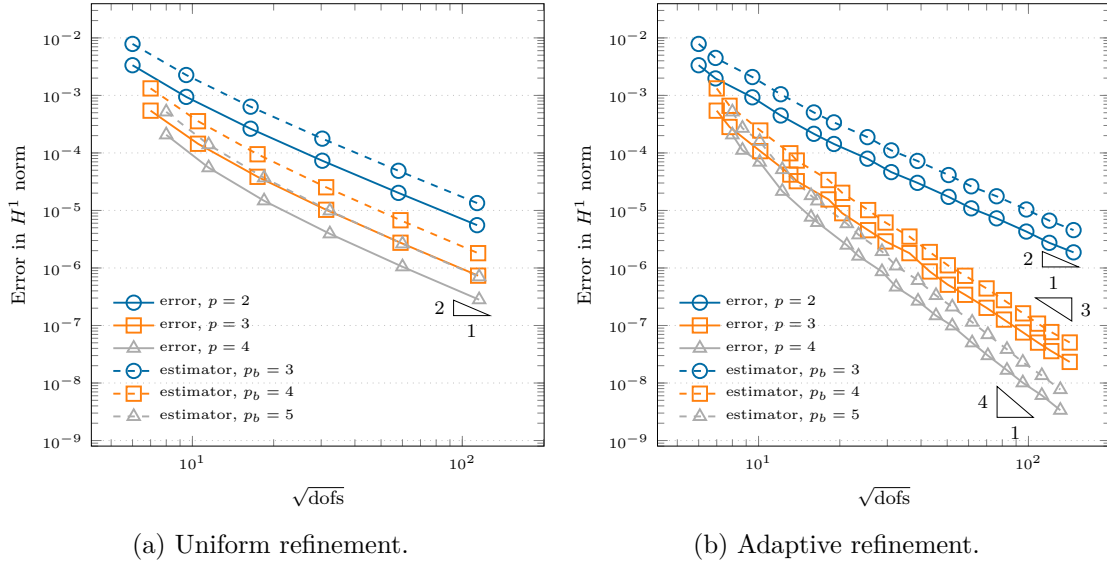


Figure 5.4 – Study of the convergence of the error measured in the H^1 -norm and the bubble estimator for the Poisson singular problem. Comparison of uniform and adaptive refinements.

symmetry only one quarter of the plate is modeled. In particular, we use a Cartesian mesh trimmed by a curve corresponding to a quarter of a circle (light grey grid and dashed, red curve in Figure 5.5, respectively). We recall that the displacement \mathbf{u} is given as a function of the polar coordinates (r, θ) and reads [Gould, 1999]:

$$\mathbf{u}(r, \theta) = \begin{pmatrix} u_x \\ u_y \end{pmatrix} = \frac{T_x R}{8\mu} \begin{pmatrix} \left(\frac{r}{R}(\kappa + 1) \cos(\theta) + 2 \frac{R}{r}((1 + \kappa) \cos(\theta) + \cos(3\theta)) - 2 \frac{R^3}{r^3} \cos(3\theta) \right) \\ \left(\frac{r}{R}(\kappa - 3) \sin(\theta) + 2 \frac{R}{r}((1 - \kappa) \sin(\theta) + \sin(3\theta)) - 2 \frac{R^3}{r^3} \sin(3\theta) \right) \end{pmatrix}, \quad (5.46)$$

where $\kappa = 3 - 4\nu$ is the so-called Kolosov constant for the plane strain case, μ represents the second Lamè parameter, T_x is the exact traction applied at infinity and R denotes the radius of the hole. The solution \mathbf{u} is smooth and therefore we expect optimal rates of convergence for both uniform and adaptive refinements. However, due to the presence of a hole, a stress concentration appears in the proximity of the bottom left corner. This feature is correctly detected and resolved by the error estimator, which yields a faster convergence in the pre-asymptotic regime of the adaptive simulation, as shown in Figure 5.6. Finally, in Figure 5.7, the hierarchical meshes obtained at various iterations $k = 4, 8, 11, 14$ of the adaptive loop are presented, where we remark that, in the visualization, the triangular elements close to the trimming curve are those provided by the re-parametrization tool for integration [Antolin et al., 2019]. Here, we notice how the refinement is at first steered around the hole where the stress concentration is located and then it gradually propagates into the rest of the computational domain.

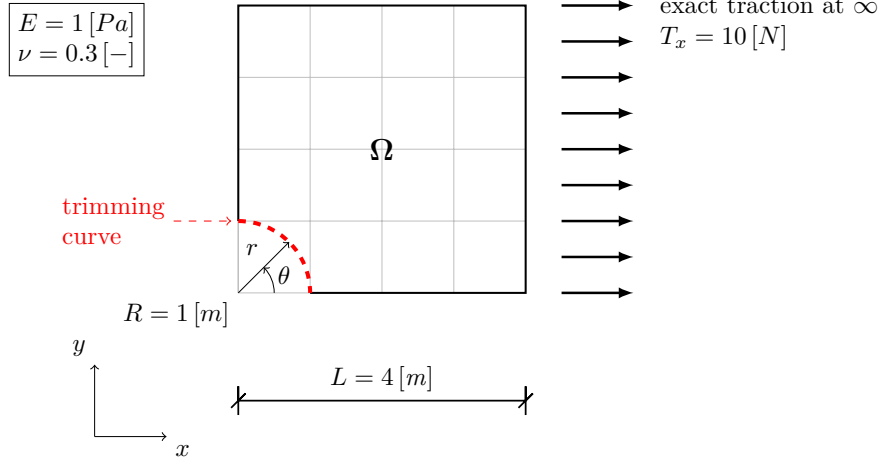


Figure 5.5 – Problem setup and boundary conditions for the plate with a hole benchmark.

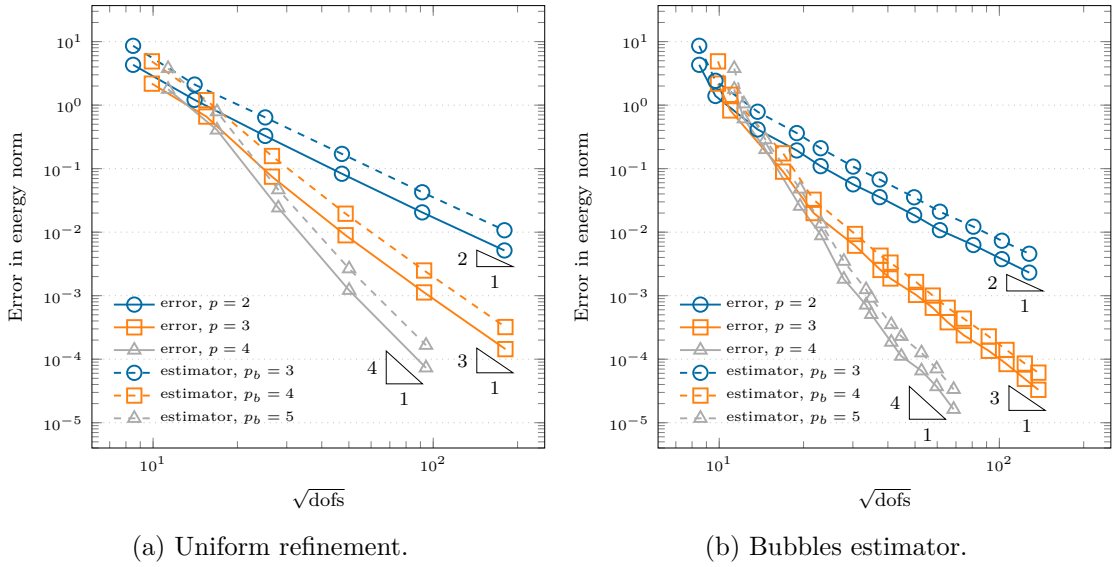


Figure 5.6 – Study of the convergence of the error measured in the energy norm and the bubble estimator for the plate with a hole. Comparison of uniform and adaptive refinements.

5.6.3 Kirchhoff Plates

In the next examples, we assess the behavior of the bubble estimator in the scope of adaptivity for Kirchhoff plates. For this problem, the bubble space \tilde{Z}_h^{p+1} is built such that the C^1 -continuity requirement is met.

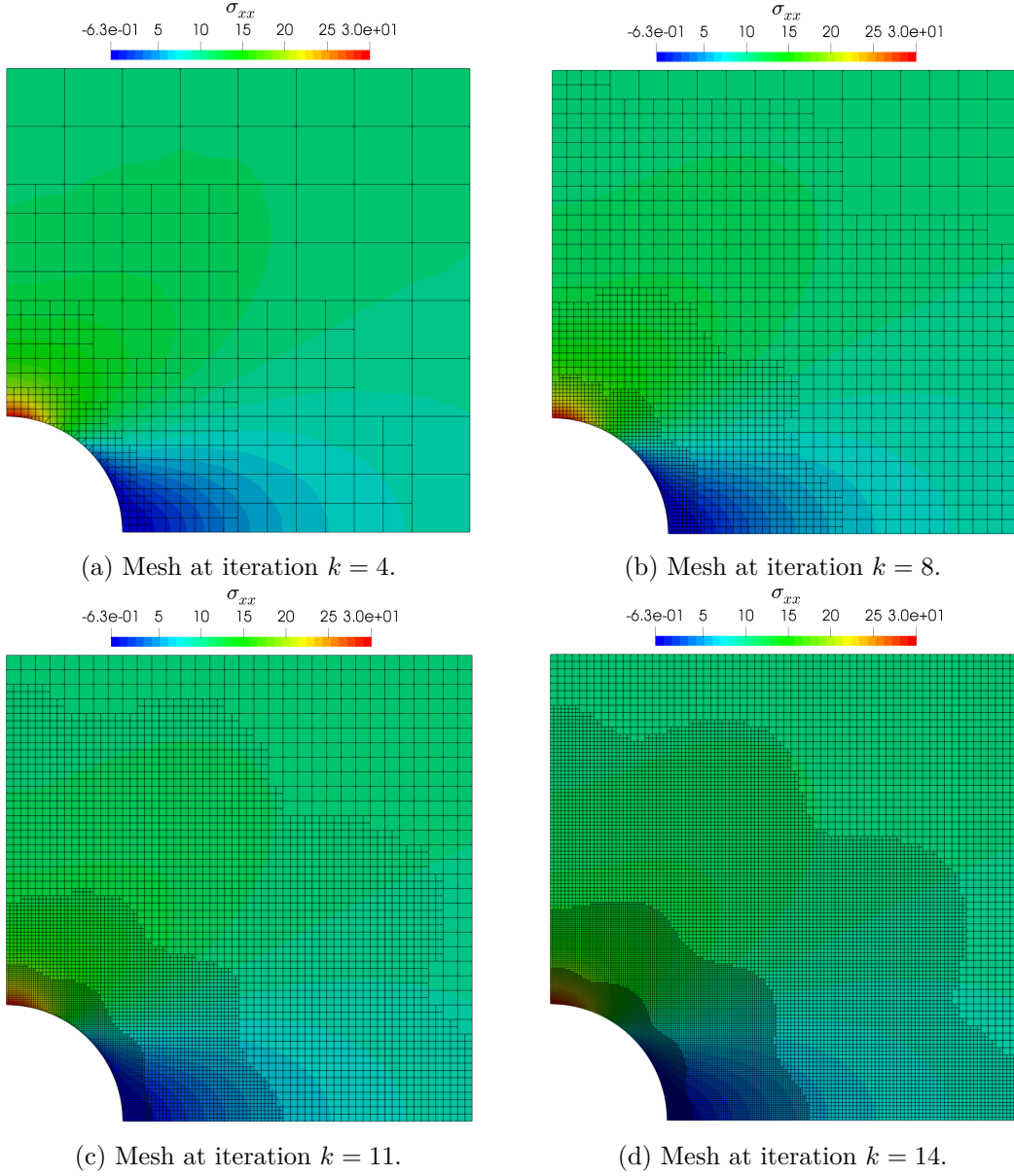


Figure 5.7 – Mesh and stress σ_{xx} at different steps of the adaptive loop driven by the bubble error estimator for the plate with a hole benchmark, solution obtained using \mathcal{THB} -splines of degree $p = 2$. Results obtained by setting the maximum strategy marking parameter to $\gamma = 0.55$.

Smooth solution on a square plate

In the first plate example, we analyzed the behavior of the bubble error estimator compared to a classical residual-based type error estimator. We define the computational domain to be the unit square $\Omega = [0, 1]^2$ and we impose the following homogeneous

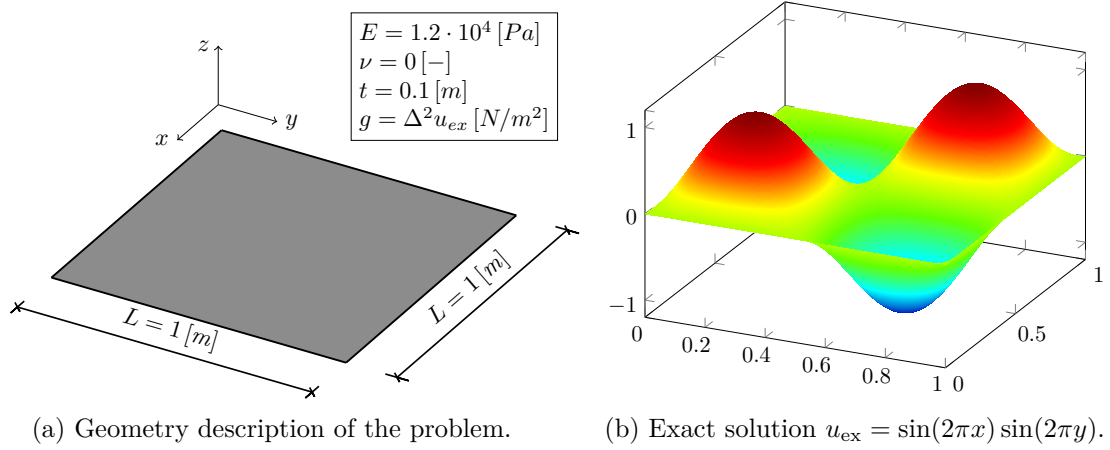


Figure 5.8 – Geometry, physical parameters and exact solution contour of the square plate example.

boundary conditions:

$$u = 0 \quad \text{on} \quad \partial\Omega, \quad (5.47a)$$

$$\nu D\Delta u + (1 - \nu)D\mathbf{n} \cdot (\nabla\nabla u)\mathbf{n} = 0 \quad \text{on} \quad \partial\Omega. \quad (5.47b)$$

Additionally, the applied load g is constructed such that it fulfills the following manufactured solution $u_{\text{ex}} = \sin(2\pi x) \sin(2\pi y)$. Namely, g is given by:

$$g = 64\pi^4 \sin(2\pi x) \sin(2\pi y). \quad (5.48)$$

The geometry of the structure, physical parameters and the exact solution are depicted in Figure 5.8, where the physical constants are chosen such that the flexural rigidity D is unitary. For the sake of comparison, in the case of Kirchhoff plates only, we drive the adaptive method by means of both the residual-based and the bubble-based estimators. To test our implementation, we perform at first uniform refinement for different spline degrees $p = 3, 4, 5$ and check the convergence rate of the error in the energy norm and both estimators against the element size h , as depicted in Figure 5.9 for the bubble and residual-based, respectively. In all the presented cases the optimal asymptotic rates of convergence are observed, both for the error and the estimator. Furthermore, we remark that the bubble estimator provides a better approximation of the real error compared to the residual-based. Finally, we run the same example letting the bubble and residual-based estimators drive automatically the adaptive simulation. The results are reported in Figure 5.10, where the error in energy norm and the estimator are plotted against the square root of the number of dofs. We observe analogous results to the uniform refinement case, where for $p = 3, 4, 5$ the expected asymptotic rates of convergence are achieved. Also in the adaptive example, the bubble estimator provides a better estimate of the real error compared to a classical residual-based estimator, yielding an effectivity index much closer to the optimal value, as shown in Figure 5.11. This statement holds in

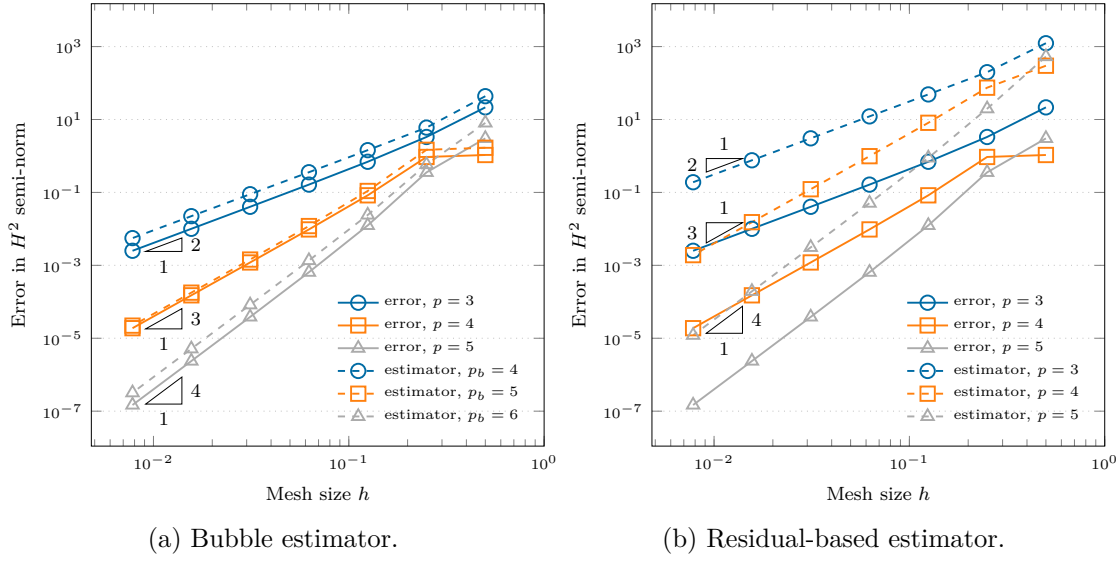


Figure 5.9 – Study of the convergence of the error measured in the H^2 semi-norm, bubble and residual-based estimators against the mesh size h for different p on a smooth plate problem ($u_{\text{ex}} = \sin(2\pi x) \sin(2\pi y)$). Uniform refinement.

an analogous way for all the presented examples and verifies numerically the efficiency of the proposed method.

Smooth solution on a rhomboid plate

In the following example we take as computational domain Ω the rhomboid plate described in Figure 5.12a. The boundary conditions on the deflections and moments (defined on the entire boundary $\partial\Omega$) and the applied load are manufactured such that the exact solution u_{ex} reads $u_{\text{ex}} = \sin(\pi x) \sin(\pi y)$. In Figure 5.12b a comparison of the total CPU time needed to compute the bubble and residual estimators, respectively, is presented. We note that with our approach we generally gain around one order of magnitude compared to a classical residual-based estimator, independently of the degree p . The obtained results in the adaptive case are depicted in Figure 5.13, where we compare the convergence of the true error and the estimator against the square root of the number of dofs for the bubble and the residual-based estimators, respectively. Similarly to the previous example, for different degrees of the discretization $p = 3, 4, 5$ the expected asymptotic rates of convergence are attained by both estimators. However, we highlight again that the bubble indicator performs much better in predicting the true error. This example is meant to test the robustness of the proposed estimator to element distortion.

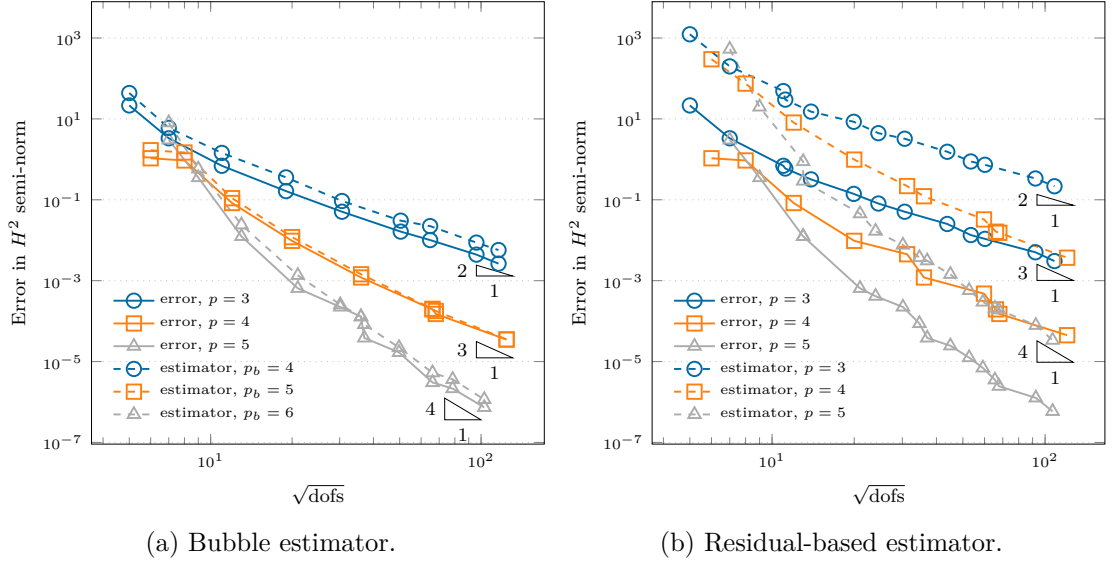


Figure 5.10 – Study of the convergence of the error measured in the H^2 semi-norm, bubble and residual-based estimators against the square root of the number of dofs for different p on a smooth plate problem ($u_{\text{ex}} = \sin(2\pi x) \sin(2\pi y)$). Adaptive refinement.

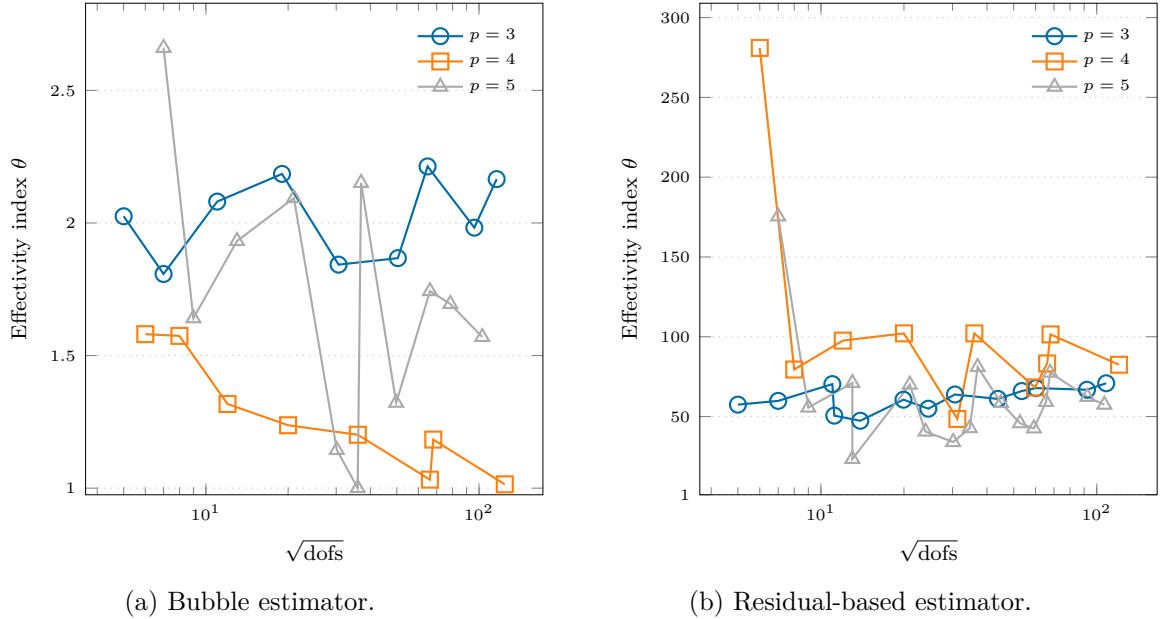


Figure 5.11 – Effectivity index θ for the bubble and residual-based estimators against the square root of the dofs for different $p = 3, 4, 5$ on the plate problem with smooth solution. Notice the difference of two orders of magnitude in the scale used for the y-axis.

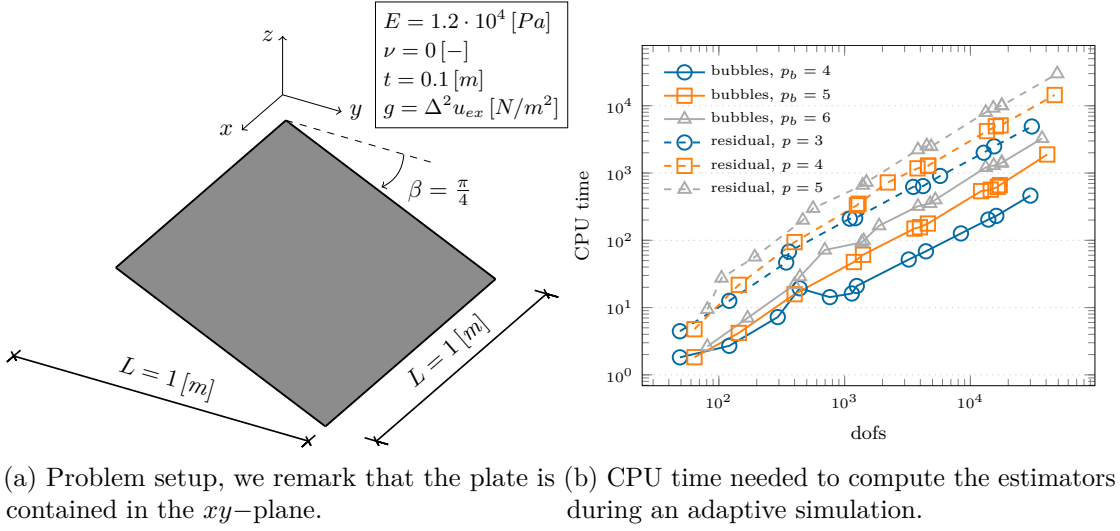


Figure 5.12 – Geometry and physical parameters of the rhomboid plate example and CPU time spent computing the bubble and residual estimators for different discretization degrees $p = 3, 4, 5$. We measure the time by using the MATLAB built-in commands *tic* and *toc*.

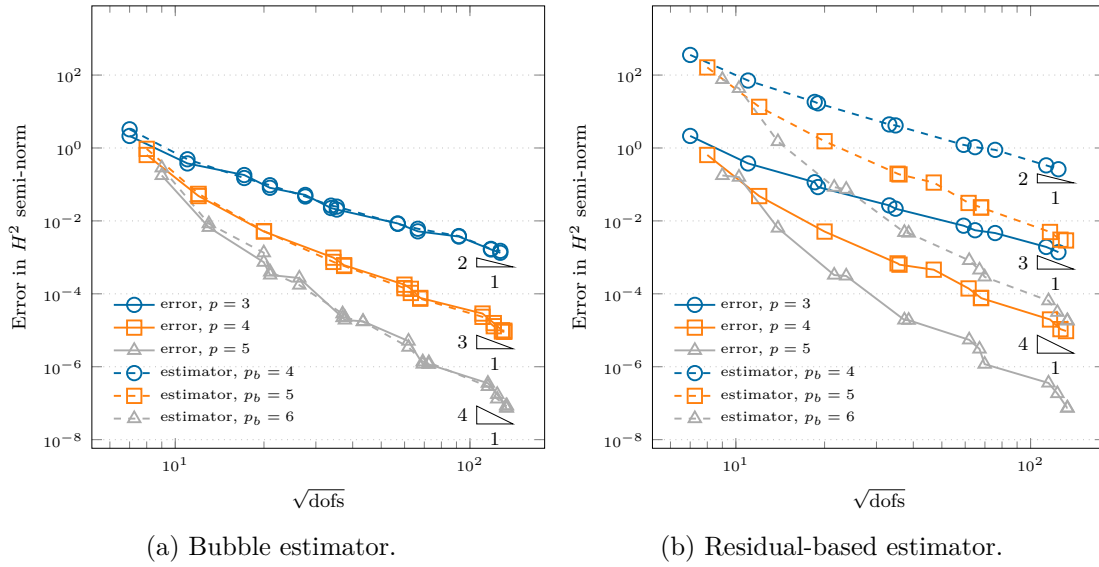


Figure 5.13 – Study of the convergence of the error measured in the H^2 semi-norm, bubble and residual-based estimators against the square root of the number of dofs for different p on the rhomboid plate problem ($u_{ex} = \sin(\pi x) \sin(\pi y)$). Adaptive refinement based on the maximum strategy ($\gamma = 0.5$).

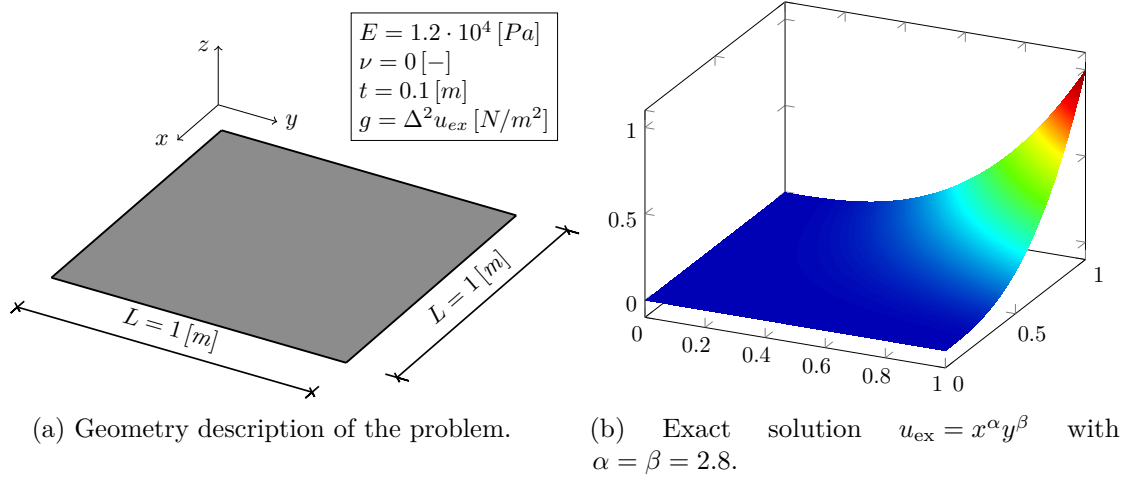


Figure 5.14 – Geometry, physical parameters and exact solution contour of the singular square plate example.

Singular solution on a square plate

In the next example, we consider again the computational domain to be a unit square $\Omega = [0, 1]^2$, see Figure 5.14. We set again the physical parameters and the thickness of the plate such that $D = 1 [Pa \cdot m^3]$. However, this time the manufactured solution $u_{\text{ex}} = x^\alpha y^\beta$ with $\alpha = \beta = 2.8$ is constructed such that a singularity is present along the bottom and the left edges of the plate. In particular, it holds $u_{\text{ex}} \in H^3(\Omega) \setminus H^4(\Omega)$. The applied load g is again constructed such that it fulfills the manufactured solution u_{ex} and it is given as follows:

$$g = \alpha(\alpha - 1)(\alpha - 2)(\alpha - 3)x^{(\alpha-4)}y^\beta + \beta(\beta - 1)(\beta - 2)(\beta - 3)x^\alpha y^{(\beta-4)} + 2 \left[\alpha\beta(\alpha - 1)(\beta - 1)x^{(\alpha-2)}y^{(\beta-2)} \right]. \quad (5.49)$$

The boundary conditions are also constructed from the exact solution as:

$$u = u_{\text{ex}} \quad \text{on} \quad \partial\Omega, \quad (5.50a)$$

$$\nu D \Delta u + (1 - \nu) D \mathbf{n} \cdot (\nabla \nabla u) \mathbf{n} = M_{\text{ex}} \quad \text{on} \quad \partial\Omega, \quad (5.50b)$$

where u_{ex} and M_{ex} denote the exact deflection and bending moment, respectively. Similarly to the singular example presented in the scope of the Poisson problem, the reduction in regularity of the solution limits the rate of convergence in case of uniform refinement, even for increasing p . This effect can be clearly seen in Figure 5.15 for $p = 3, 4$. Figure 5.15 also shows the results for an adaptive simulation driven by the bubble estimator, where optimal rates of convergence are recovered by our method and a significant increase in accuracy per degree-of-freedom is achieved. Additionally, the obtained mesh at different iterations of the adaptive algorithm are depicted in Figure 5.16, where it can be seen that the singularities are accurately detected and resolved by the

estimator.

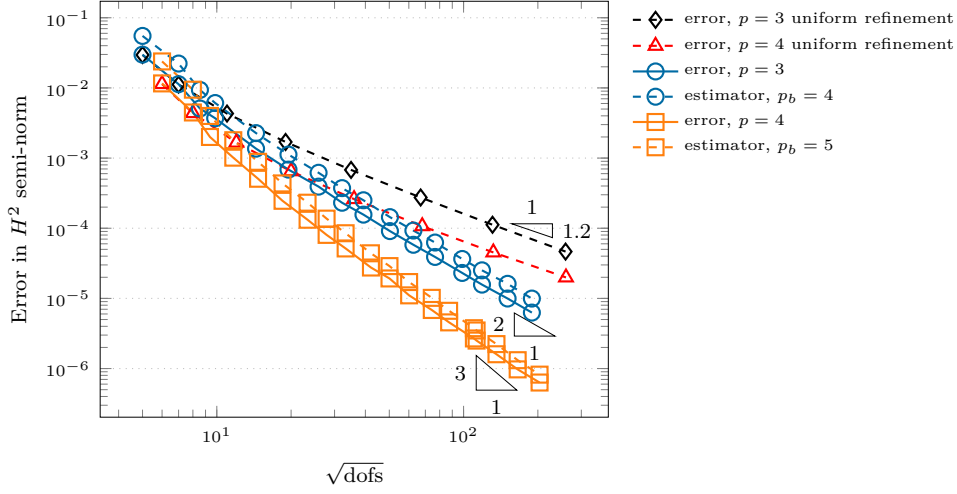


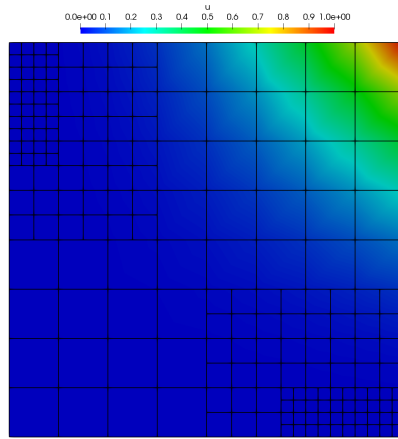
Figure 5.15 – Study of the convergence of the error measured in the H^2 semi-norm and bubbles estimator against the square root of the number of dofs for different p (and associated p_b for the bubble space) on the square plate problem with singularity ($u_{\text{ex}} = x^\alpha y^\beta$ with $\alpha = \beta = 2.8$).

Point load on a rectangular plate

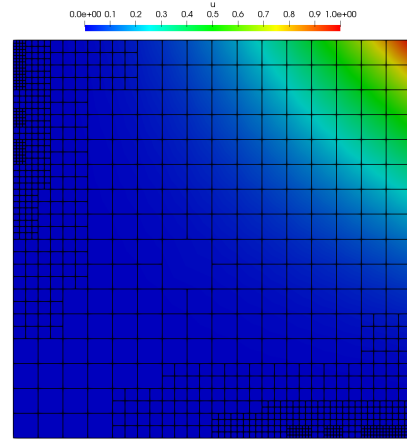
In this example, we test the performance of the proposed error estimator when a point load is applied as an external force. We perform again our computation on a unit square $\Omega = [0, 1]^2$ and we suppose that the plate is simply supported on the entire boundary $\partial\Omega$, meaning that the deflection and bending moment vanish. Additionally, we apply the external point load at the center of the structure. There exists an analytical solution for the deflection under the load, given as an infinite series [Reddy, 2006]:

$$u_{\text{ex}} = \frac{4gL^2}{D\pi^4} \sum_{n=1,3,\dots}^{\infty} \sum_{m=1,3,\dots}^{\infty} \frac{1}{(m^2 + n^2)^2}, \quad (5.51)$$

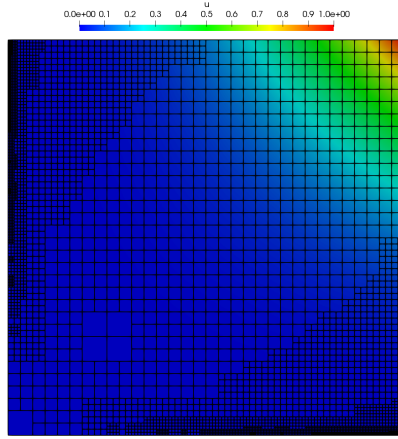
where g represents the applied external force, L is the length of the plate and D denotes its flexural stiffness. We set $g = -1 [N]$, $L = 1 [m]$ and the physical parameters such that $D = 1 [Pa \cdot m^3]$. Computing (5.51) with an adequate number of terms for the double Fourier series yields a reference value of $u_{\text{ex}} = -0.011600839735872 \dots [m]$ for the deflection. In Figure 5.17a the convergence of the normalized deflection is plotted against the number of degrees-of-freedom, where the former is defined as $|1 - u_h/u_{\text{ex}}|$. Moreover, the obtained solution and hierarchical mesh obtained after $k = 9$ steps of the adaptive algorithm are depicted in Figure 5.17b. Here, the advantages of using an adaptive scheme are clearly highlighted, in terms of efficiency and accuracy per degree-of-freedom. For instance, at around 2500 dofs the adaptive strategy yields a deflection which is already



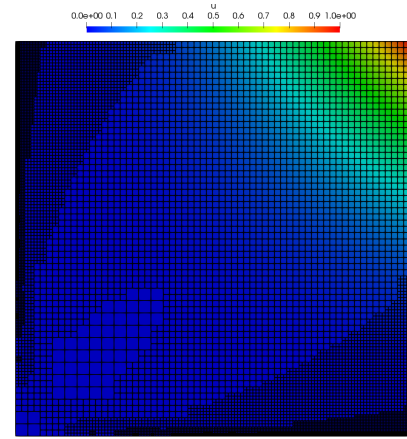
(a) Deflection u_h and mesh at iteration $k = 3$.



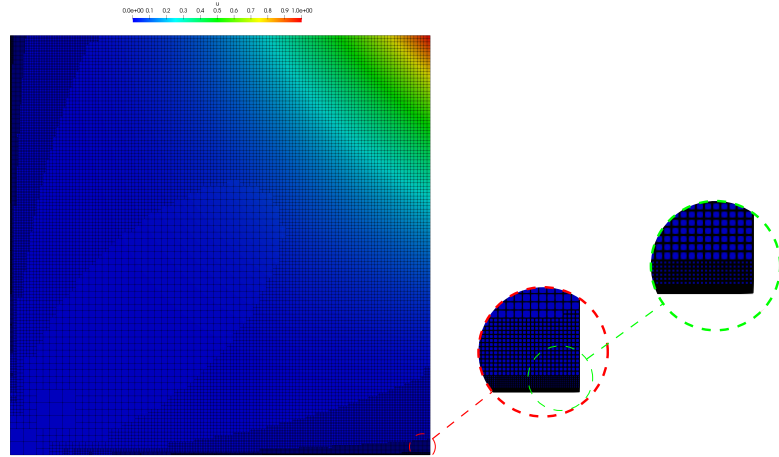
(b) Deflection u_h and mesh at iteration $k = 5$.



(c) Deflection u_h and mesh at iteration $k = 8$.



(d) Deflection u_h and mesh at iteration $k = 10$.

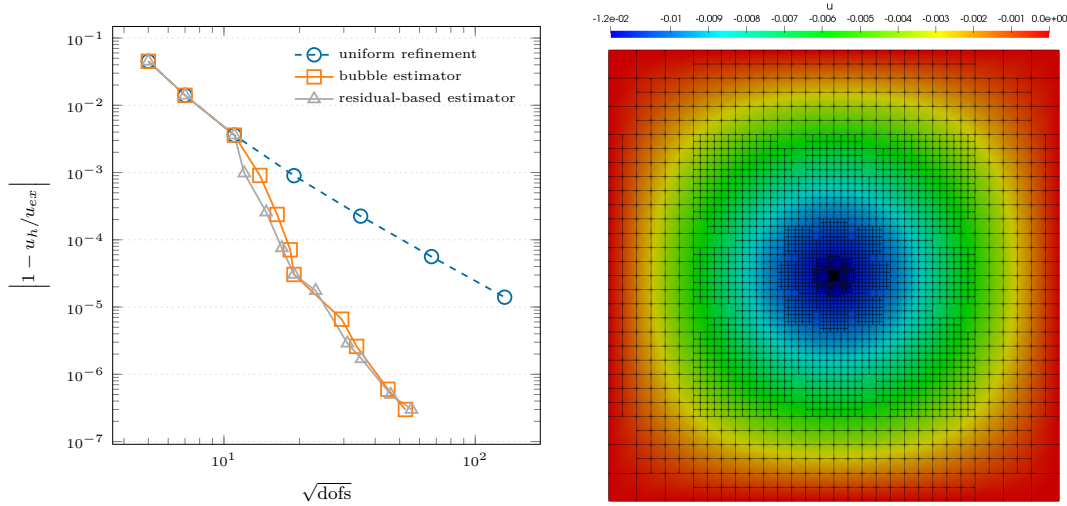


(e) Deflection u_h and mesh at iteration $k = 14$ and zoom on the bottom right corner.

Figure 5.16 – Mesh and solution contours at different steps $k = 3, 5, 8, 10, 14$ of the adaptive loop driven by the bubble error estimator for the square plate with singular solution. \mathcal{HB} -splines of degree $p = 3$.

two orders of magnitude more accurate compare to the results obtained by uniform refinement.

Remark 5.9 (Regularization of the Dirac delta) *It is worth noting that a point load is modeled as a Dirac delta, which rigorously speaking is a distribution. Therefore, for the evaluation of the residual in a strong sense, needed in the residual-based estimator, we cannot directly use it inside our computations but instead we must regularize it. In our example, this is achieved with a steep Gaussian function. A considerable amount of literature has been written on how to regularize and correctly integrate the Dirac delta according to the corresponding application (e.g. we refer to [Waldén, 1999; Tornberg and Engquist, 2003, 2004; Engquist et al., 2005] and references therein) and although our approach might not be optimal we feel it suits the purpose of our test case while at the same time it proves to be easily implementable.*



(a) Study of the convergence of the displacement under a point load on the square plate example, solution obtained by adaptive refinement vs. uniform refinement. (b) Mesh at iteration $k = 9$ and corresponding adaptive refinement driven by the bubble estimator.

Figure 5.17 – Convergence study and numerical solution for the square plate problem subjected to a point load applied at the center of the plate. \mathcal{HB} -splines of degree $p = 3$.

5.6.4 Kirchhoff-Love shell

In the next series of benchmarks, we test the performance of the proposed method in the context of trimmed Kirchhoff-Love shells. For this problem, the bubble space \tilde{Z}_h^{p+1} must comply to the C^1 requirement. We remark that by solving the residual equation in a weak sense in a suitable space we bypass completely the computational burden stemming from the evaluation of higher order derivatives and their jumps across edges, which would be required in classical residual-based error indicators. This is particular advantageous

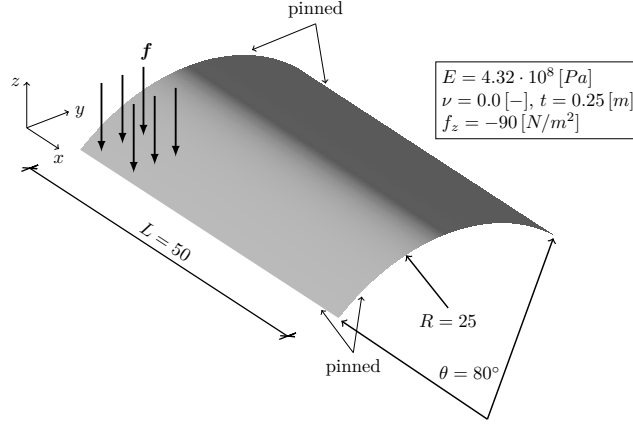


Figure 5.18 – Geometry description and physical parameters of the pinned roof example (only part of the load is depicted for visualization purposes).

for Kirchhoff-Love shells, where the terms associated with higher derivatives involve the cumbersome and error-prone task of taking (nested) covariant derivatives of quantities of interest, see for instance [Maurin et al., 2018; Benzaken et al., 2021].

Gravity load on a pinned roof

In this example, we study the behavior of a structure subjected to its self-weight where pinned supports are applied to the entire boundary. Specifically, we fix the z -displacement on the entire $\partial\Omega$ and the x - and y -displacement on the curved and straight edges, respectively. The geometry, boundary conditions and physical parameters are given in Figure 5.18, where the problem setup (except for the boundary conditions) is taken from the classical Scordelis-Lo benchmark, e.g. see [Timoshenko and Woinowsky-Krieger, 1959; Belytschko et al., 1985]. The boundary conditions are modified in order to mitigate the numerical issues stemming from the computation of the reference solution and corresponding error, since the original problem definition of the Scordelis-Lo roof is not well-posed, see for instance [Kiendl et al., 2009]. Moreover, the conditioning of the corresponding linear system causes additional numerical instabilities and loss of significant digits [Benzaken et al., 2021]. To the best of the authors' knowledge, there is no closed-form global solution available in the literature for the problem at hand. Therefore, all the convergence studies presented in the following assess the behavior against a reference solution $\mathbf{u}_h^{\text{ref}}$, which was computed using B-splines of degree $p = 8$ on a fine uniform mesh with approximately 200 000 degrees-of-freedom. We define this quantity as:

$$\|\tilde{e}\|_{E(\Omega)} = \sqrt{a(\mathbf{u}_h^{\text{ref}} - \mathbf{u}_h, \mathbf{u}_h^{\text{ref}} - \mathbf{u}_h)} \sim \|\mathbf{u} - \mathbf{u}_h\|_{E(\Omega)}, \quad (5.52)$$

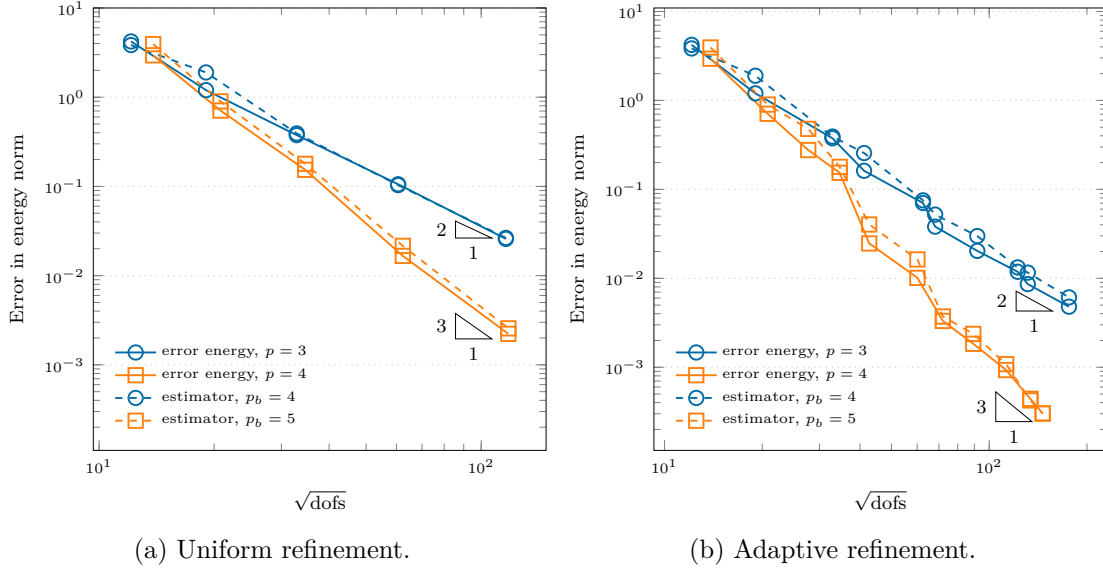


Figure 5.19 – Study of the convergence of the error in the energy norm for the bubble estimator employing hierarchical B-splines of degree $p = 3, 4$, pinned roof example.

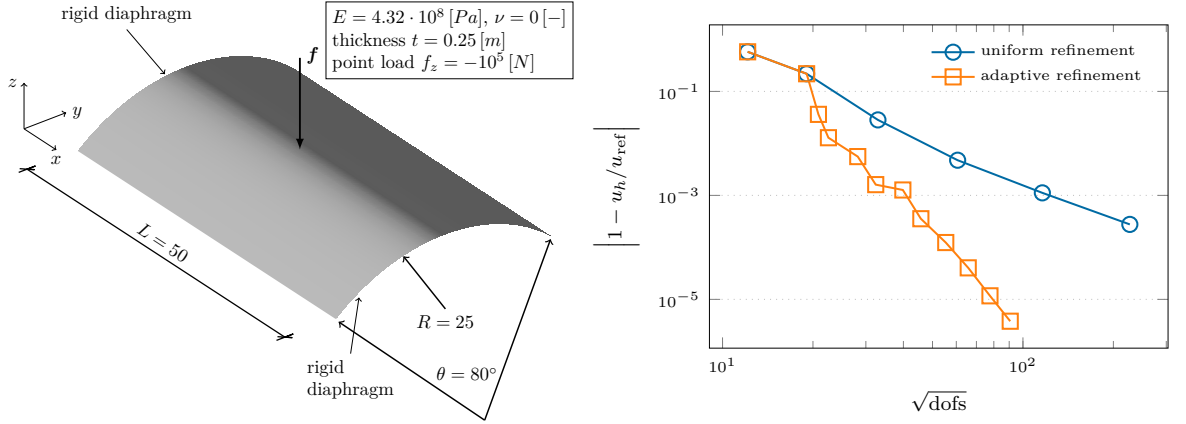
where $\mathbf{u}_h^{\text{ref}}$ denotes the reference solution. We highlight that the numerical integration of \tilde{e} is performed on the finer mesh to properly capture the error. In Figure 5.19 the convergence behavior of $\|\tilde{e}\|$ is depicted for the case of uniform refinement and adaptive refinement driven by the bubble estimator, against the number of dofs. We notice that since the true solution of the problem is regular enough, we obtained the optimal asymptotic rate of convergence, both for the error in energy norm and the estimator. This example is thought as a first assessment of the performance of the proposed estimator on shell geometries.

Remark 5.10 *In standard shell benchmarks adopted in the literature, convergence is usually tested against a reference value for the displacement in some points of interest of the structure. While this information is relevant in many engineering applications, we feel that a deeper look into the behavior of derived global quantities, like for instance the error in energy norm used here, can be useful and mathematically more rigorous.*

Point load on the Scordelis-Lo roof

The next example is meant to demonstrate once again the higher accuracy per-degree-of-freedom achievable using local refinement. The geometrical setup, physical parameters and boundary conditions are taken as defined in the Scordelis-Lo benchmark and are given in Figure 5.20a, where the rigid diaphragms at both ends of the cylinder hinder the displacement in the yz -plane. However, we change the external prescribed loading. Specifically, we apply a point load of magnitude 10^5 [N] in the middle of the structure,

directed in the negative vertical direction. The convergence behavior of the displacement



(a) Geometry description and physical parameters of the Scordelis-Lo roof subjected to a point load.

(b) Study of the convergence of the displacement on the shell under a point load, uniform refinement against adaptive refinement using hierarchical B-splines of degree $p = 3$.

Figure 5.20 – Problem setup and convergence plot for the Scordelis-Lo roof example under a point load.

under the point load is depicted in Figure 5.20b, where a converged value of $\mathbf{u}_h^{\text{ref}} = [0, 0, -0.206794699852788 \dots]^\top [m]$ has been obtained with an overkill solution, which was computed using B-splines of degree $p = 4$ on a fine uniform mesh with approximately 200 000 degrees-of-freedom. It can be seen that the solution obtained with an adaptive simulation based on the bubble estimator is several orders of magnitude more accurate compared to performing uniform refinement of the patch, for an equal number of dofs. Finally, in Figure 5.21, the obtained displacement in the z -direction, hierarchical mesh and Von Mises stress distribution are presented at different steps of the adaptive loop where we remark that once again the estimator properly captures and resolves the sharp features of the solution.

Trimmed Scordelis-Lo roof with an elliptic hole

In this numerical example, we take again the geometry and material properties as defined in the well-known Scordelis-Lo roof benchmark. Again, we impose rigid diaphragm boundary conditions at both curved ends of the cylindrical structure, where we hinder the displacement in the xz -plane as defined by the coordinate system of Figure 5.22. Then, we trim out a circular hole in the parameter space of the surface, as depicted in Figure 5.22. We recall that, analogously to the original setup, the structure is subjected to its self-weight and we set its value to $f_z = -90 [N/m^2]$. To the best of the authors' knowledge there is no analytical solution in closed form for this problem, therefore we perform a convergence study of the error measured in the energy norm against a reference solution obtained with bi-quintic B-splines defined on a uniform mesh with 62 456 active

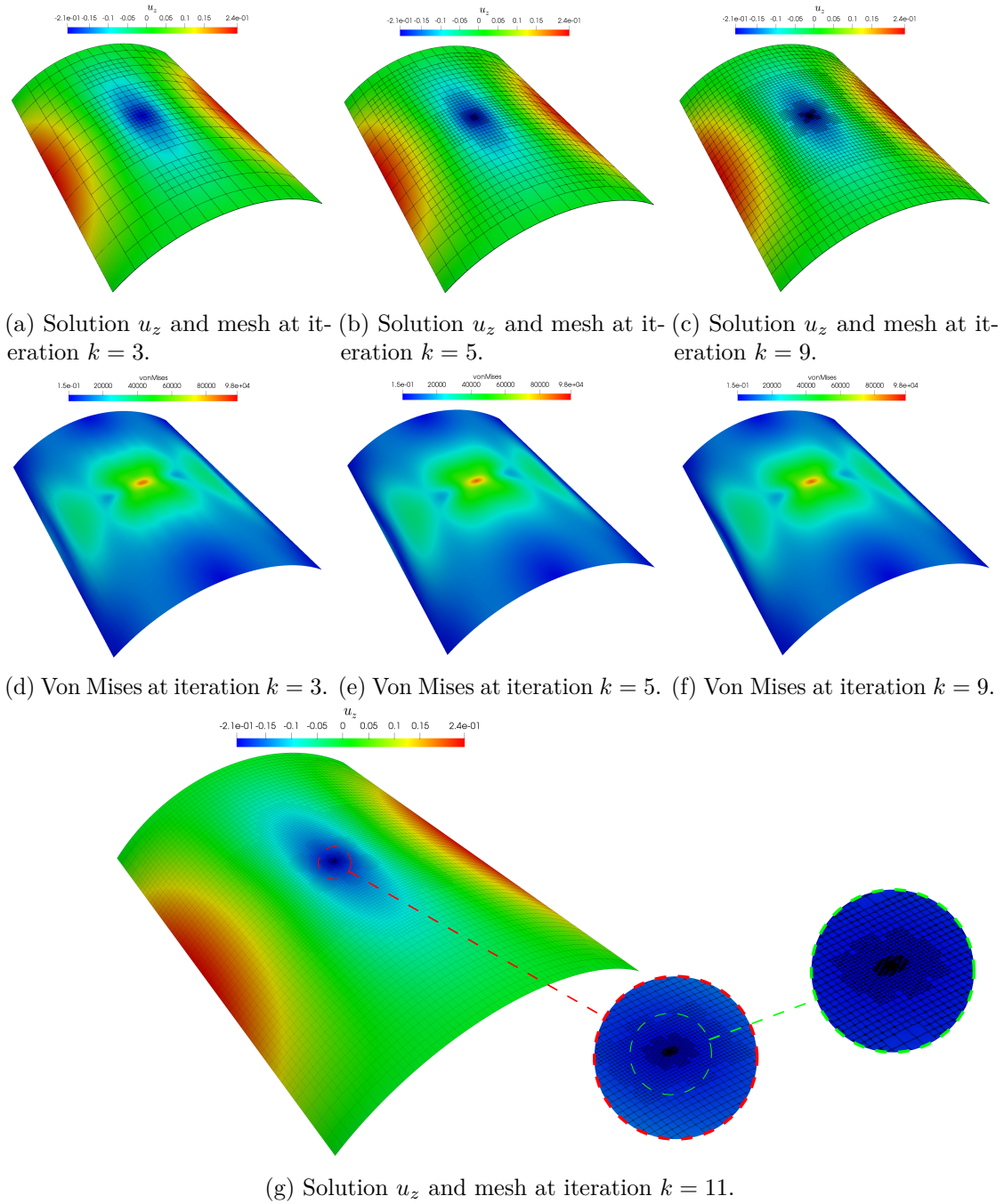


Figure 5.21 – Mesh, solution u_z and Von Mises stress at different steps of the adaptive loop driven by the bubble error estimator for the Scordelis-Lo roof subjected to a point load, solution obtained employing hierarchical B-splines of degree $p = 3$.

elements. The results are depicted in Figure 5.23 for THB-splines of degree $p = 3, 4$, where uniform and adaptive refinement are analyzed. In both cases, we observe optimal rates of convergence of the error and a similar level of accuracy between the uniform and the adaptive strategy. This is due to the fact that this problem exhibits a smooth solution.

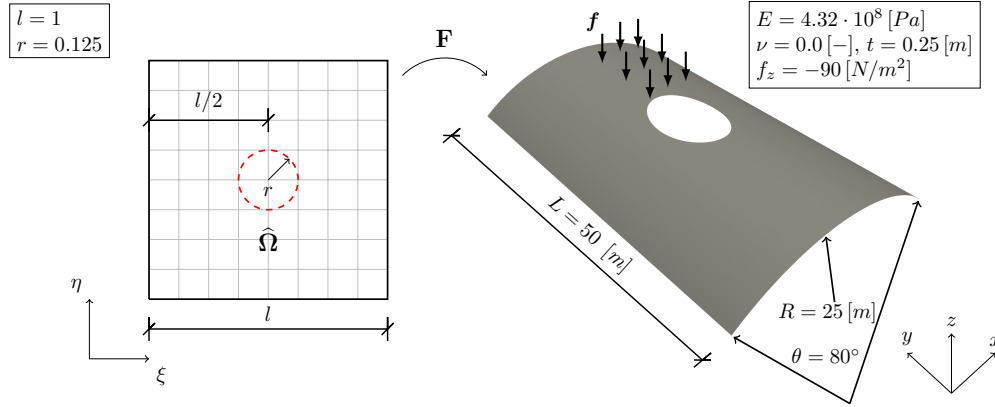


Figure 5.22 – Geometry and physical parameters of the Scordelis-Lo roof with an elliptic hole (only part of the gravity load is shown for visualization purposes).

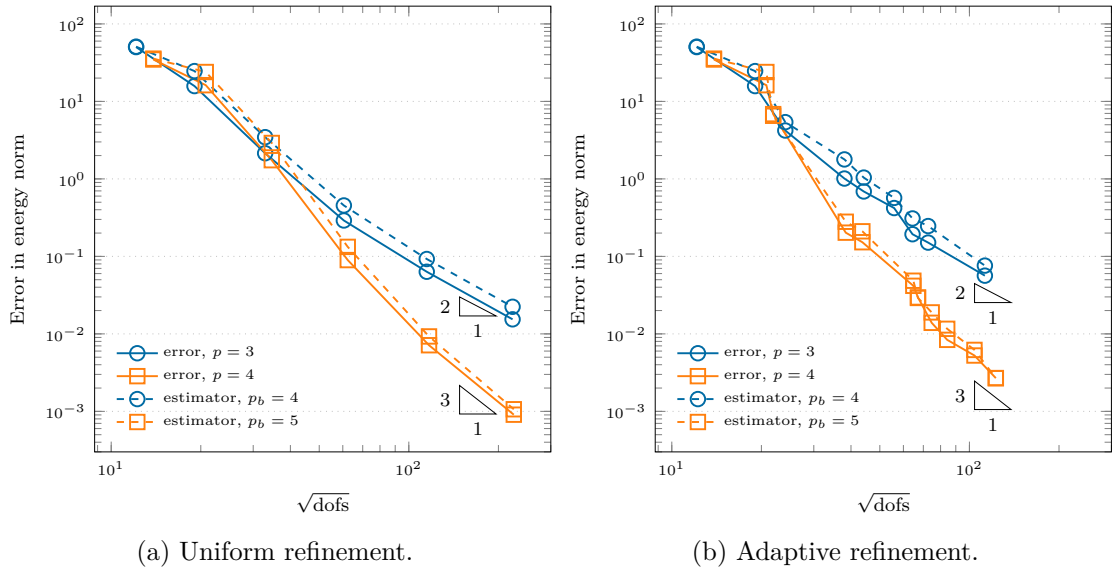


Figure 5.23 – Study of the convergence of the (approximated) error measured in the energy norm and the bubble estimator for the Scordelis-Lo roof with a hole. Comparison of uniform and adaptive refinements.

Trimmed Scordelis-Lo roof with holes under a point load

In this numerical test, we use again the same geometrical and material properties as defined in the Scordelis-Lo roof benchmark. This time, we trim the structure with four holes defined in the parametric space of the surface as circles, as depicted in Figure 5.24. As in the example above, both curved ends are supported by rigid diaphragms boundary conditions (the displacement in the xz -plane is set to zero, with respect to the coordinate system defined in Figure 5.24). Additionally, the roof is subjected to a point load in the vertical direction, defined as $f_z = -10^5$ [N], applied at the center of the structure. A reference displacement u_z^{ref} at the location under the point load has been obtained with a fine, uniformly refined mesh of bi-cubic B-splines with 53 216 active elements. In Figure 5.25g the convergence results for the uniform and the adaptive refinement are compared, where on the y -axis we plot the quantity $|1 - u_z/u_z^{\text{ref}}|$, evaluated at the center of the structure. We notice that the adaptive strategy achieves a solution that is several order of magnitudes more accurate than uniform refinement for the same number of degrees of freedom. This result confirms the superior efficiency of our error-driven strategy in the presence of singularities. Lastly, in Figures 5.25a to 5.25f we present the numerical solution u_z and corresponding Von Mises stress at different iterations $k = 3, 6, 8$ of the adaptive simulation. Remarkably, the refinement is driven not only in the region where the load is applied but also in those areas around the holes where stress concentrations are expected.

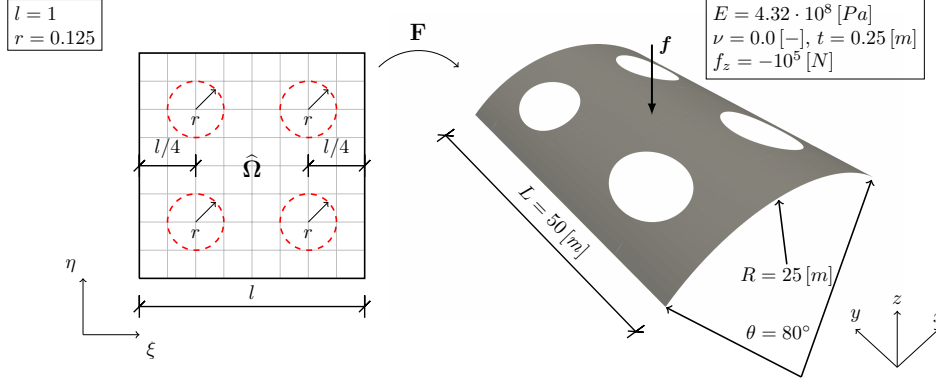


Figure 5.24 – Geometry and physical parameters of the Scordelis-Lo roof with four holes.

B-pillar of a car

In the last example of this chapter, we model the B-pillar of a car in the commercial CAD software *Rhinoceros*. The geometry (courtesy of *Honda R&D Co.*¹) is defined as a trimmed, single-patch B-spline surface of degree $p = 3$ composed of 30 and 180 knot spans in the two parametric directions, respectively. The structure is considered

¹<https://global.honda>

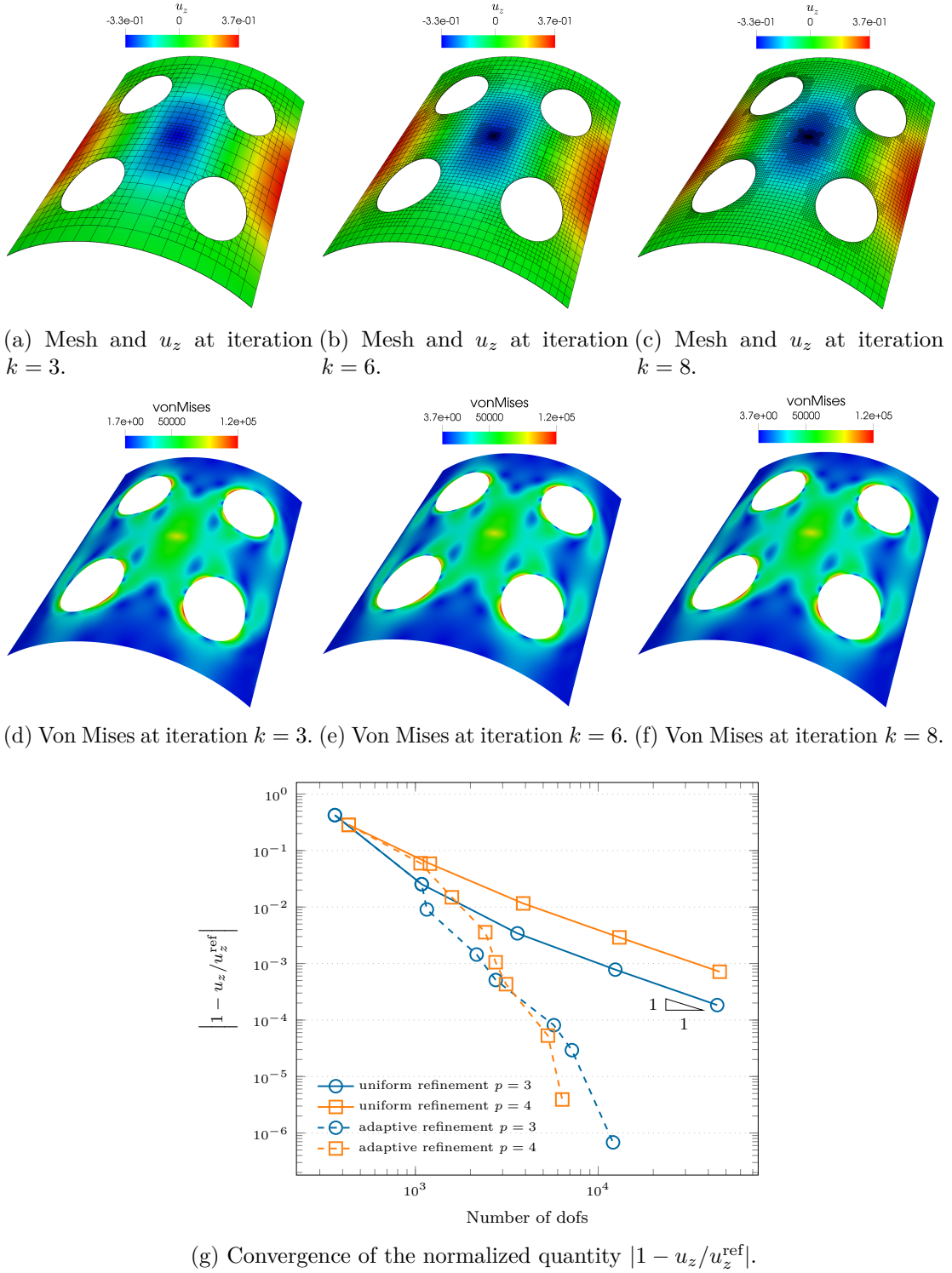


Figure 5.25 – Mesh, solution and Von Mises stress at different steps $k = 3, 6, 8$ of the adaptive loop driven by the bubble error estimator for the Scordelis-Lo roof with holes subjected to a point load, solution obtained employing \mathcal{THB} -splines of degree $p = 3$ (a-f). Convergence plot of the normalized displacement at the center of the structure. Comparison of uniform and adaptive refinements (g).

fixed on the entire external boundary (all displacement components are set to zero) and is loaded with a circular, uniformly distributed force of radius $r = 0.01 [m]$ in the y -direction, see Figure 5.26a. This example demonstrates the applicability of our approach to the adaptive analysis of complex shell structures of industrial relevance, modeled and exported directly from a commercial CAD software. In Figures 5.26b to 5.26f the numerical solution at several steps $k = 5, 7, 9, 11$ of the adaptive loop and the Von Mises stress obtained with truncated hierarchical B-splines of degree $p = 3$ are depicted. It is worth noting how the estimator correctly refines the circular area around the load, particularly at the boundary of the circle where a sharp change in boundary conditions is present. Furthermore, it also detects and refines stress concentrations in the proximity of the trimmed holes and regions of high curvature of the geometry, where higher bending stresses are expected.

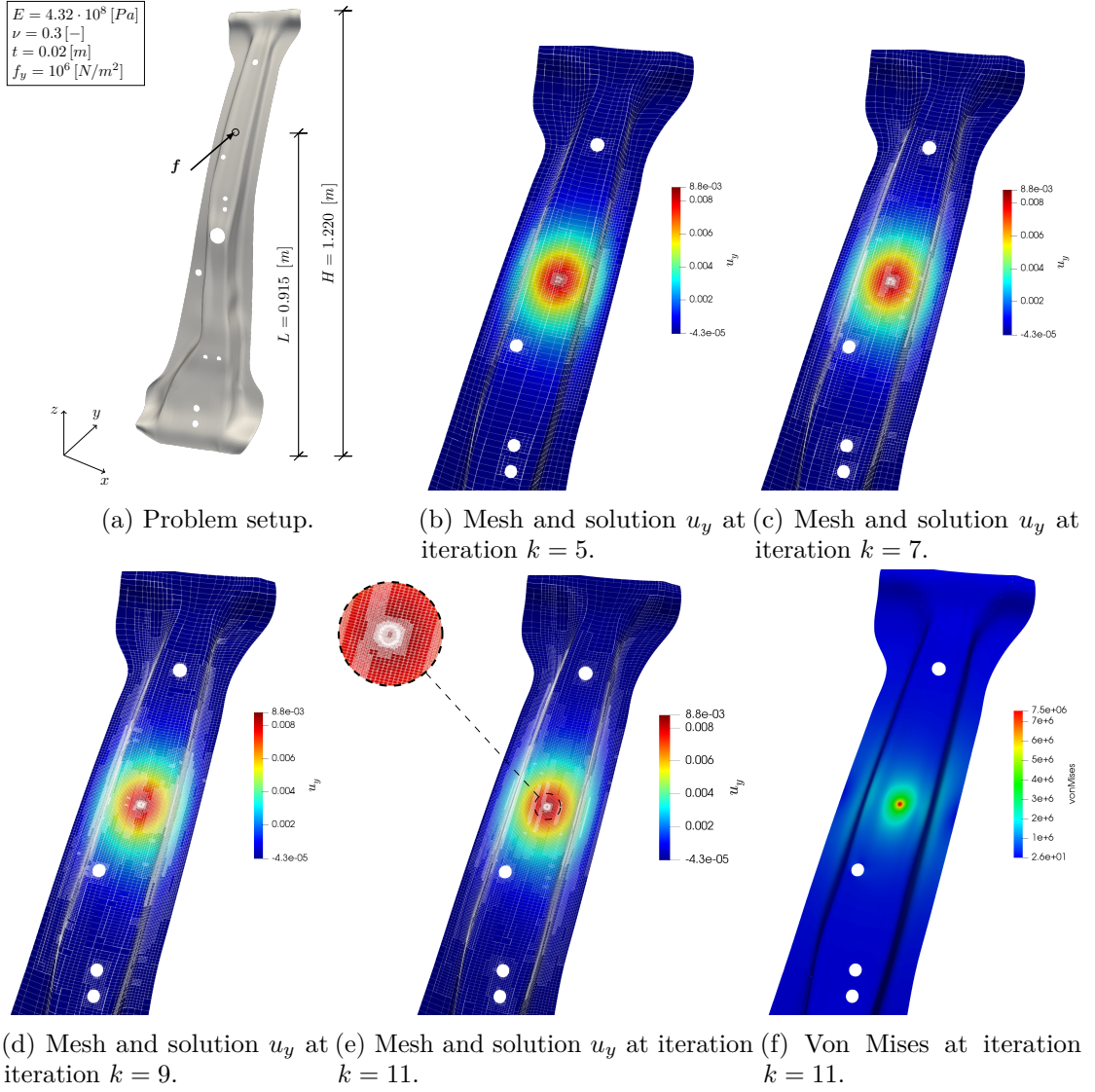


Figure 5.26 – Geometry of the B-pillar and mesh, solution and Von Mises stress at different steps of the adaptive loop driven by the bubble error estimator for the B-pillar example, solution obtained employing truncated hierarchical B-splines of degree $p = 3$.

6 Coupling of multi-patch Kirchhoff plates and Kirchhoff-Love shells

In the previous chapters we have studied a variety of numerical examples defined on a single isogeometric patch. Although the flexibility of B-splines combined with trimming allows for the geometrical description and analysis of relatively complex geometries, this framework is not powerful enough to handle arbitrary structures of engineering relevance. Specifically, structures used in the industry are typically described by multiple, non-conforming, trimmed patches which, in turn, calls for a suitable coupling strategy to achieve the required C^1 -continuity in the context of Kirchhoff plates and Kirchhoff-Love shells. Similarly to the nomenclature introduced in [Herrema et al., 2019], we distinguish between C^0 - or displacement continuity and C^1 - or rotational continuity. The latter is not restricted to smooth interfaces but will also refer in the following to patches meeting at an arbitrary angle, which is preserved during deformation. This scenario is commonly encountered in complex engineering applications. Analogously to the case of boundary conditions presented in Section 4.5, three methods are predominantly employed to achieve displacement and rotational continuity in a weak sense and they are briefly outlined in the following. Mortar-type methods have been presented for patch coupling in [Hörger et al., 2019; Hirschler et al., 2019] in the scope of Kirchhoff plates and Kirchhoff-Love shells, respectively, and have been generalized to arbitrary smoothness in [Dittmann et al., 2019]. It is well-known that mortar methods introduce additional artificial unknowns into the underlying system of equations to enforce the corresponding constraints, where the choice of discretization space for these Lagrange multipliers plays a pivotal role for the robustness of the method. Specifically, the inf-sup condition turns out to be crucial for developing stable algorithms, see [Boffi et al., 2013] for further details. Another widespread coupling technique relies on Nitsche’s method, where the reader is referred to [Guo and Ruess, 2015; Nguyen-Thanh et al., 2017; Guo et al., 2018]. This family of methods is variationally consistent and, generally speaking, it is more robust with respect to the choice of parameters compared to classical penalty approaches. However, Nitsche-type algorithms are computationally less favorable as their implementation is

problem-dependent and requires the computation of higher-order derivatives. In the case of shells, we highlight that these derivatives are defined on a manifold, which significantly increases the complexity. Lastly, penalty-like methods are widely spread throughout numerous areas of engineering due to their versatility and ease of implementation. A variant thereof named bending strip was firstly studied in [Kiendl et al., 2010] in the scope of Kirchhoff-Love shells coupled along matching interfaces. Several other penalty-like approaches able to treat non-conforming discretization have followed, where the reader is referred to [Breitenberger et al., 2015; Duong et al., 2017]. For a comprehensive comparison of the performance of these three methodologies we refer to [Apostolatos, 2019]. In this chapter, we extend our notation to the case of multiple isogeometric patches. Then, we introduce a novel penalty-like coupling strategy for Kirchhoff plates and Kirchhoff-Love shells based on the L^2 -projection of the coupling terms onto a suitable degree-reduces space defined on the corresponding interface. This choice yields an inherently locking-free strategy. Moreover, the penalty parameters are automatically defined by the problem setup, namely material properties, geometry of the structure and underlying discretization and they are suitably built to retain the higher-order accuracy of B-splines. Then, for conforming Kirchhoff plates, we address the ill-conditioning issues stemming from our choice of super-penalty parameters. Specifically, we adapt the block preconditioner based on an inexact Schur Complement Reduction (SCR) studied in [Liu and Marsden, 2019; Liu et al., 2020; Pegolotti et al., 2021] and we combine it with a preconditioner tailored to the isogeometric discretization of Kirchhoff plates, where we exploit the tensor product structure of B-splines and an efficient algorithm for the solution of the arising Sylvester-like system; for a detailed derivation we refer to [Sangalli and Tani, 2016; Montardini et al., 2018b; Loli et al., 2019]. Lastly, we verify numerically the robustness of the proposed methodology on a series of non-trimmed and trimmed examples and we compare it with other penalty-like methods.

6.1 The projected super-penalty method

In this section, motivated by the work presented in [Brivadis et al., 2015] in the context of isogeometric mortar methods, we devise a coupling strategy based on the L^2 -projection of the displacement and rotational coupling terms at the interface. Typically, they are defined in terms of the degree p of the solution space related to the corresponding patch, onto a reduced space of B-splines of degree $p - 2$ defined on the so-called *active* side of the interface. This procedure mitigates the locking phenomena due to the over-constraint of the solution space in the proximity of the corresponding coupling interface. This type of locking is analogous to what we observed for the weak imposition of boundary conditions. We remark that our method shares some similarities with the penalty coupling proposed in [Leonetti et al., 2020].

6.1.1 The multi-patch setting

Following closely the notation introduced in [Brivadis et al., 2015], let us split the computational domain Ω into N non-overlapping subdomains Ω^i such that:

$$\overline{\Omega} = \bigcup_{i=1}^N \overline{\Omega^i}, \quad \text{where} \quad \Omega^i \cap \Omega^j = \emptyset \quad \text{for} \quad i \neq j. \quad (6.1)$$

In CAD terminology, Ω is a B-Rep, i.e. a collection of trimmed surfaces endowed with their topological information. By leveraging the topology, we can then define the interface γ^ℓ between two adjacent trimmed patches $\Omega^m, \Omega^n, 1 \leq m, n \leq N$ as a common edge between their faces, see Figure 6.1 for an example on shell structure composed of four trimmed patches. Note that two surfaces can share more than one edge. For completeness, in Figure 6.2 we depict an example in the case of two conforming plates. Then, the skeleton Γ is defined as the union of all common interfaces and reads:

$$\Gamma = \bigcup_{\ell=1}^L \gamma^\ell, \quad (6.2)$$

where L denotes the total number of interfaces and ℓ is an ordered index such that $1 \leq \ell \leq L$.

Remark 6.1 *By a slight abuse of notation, γ^ℓ can represent both a trimmed or a non-trimmed coupling interface.*

Further, let us introduce the cross-points c^s as the intersection of at least three shared edges and let us label them with an ordered index $c^s, s = 1, \dots, S$, see again Figure 6.1 for an illustration.

Remark 6.2 *It is well-known that CAD softwares provide only an approximation of the true common edge γ^ℓ which depends on the chosen tolerance. For the sake of simplicity, in our derivation we assume exactness, or equivalently watertightness, of the geometric representation. From a computational standpoint, if the B-Rep is not watertight we perform a closest point projection of the relevant quantities, such as quadrature points and interface knots, onto the coupling edge. For further details we refer to [Bazilevs et al., 2012].*

For Kirchhoff plates, let us define u^m as the value of the deflection restricted to Ω^m , and similarly u^n the value of the primary field on the neighboring subdomain Ω^n . Consequently, for each interface γ^ℓ shared by Ω^m and Ω^n we can write the following

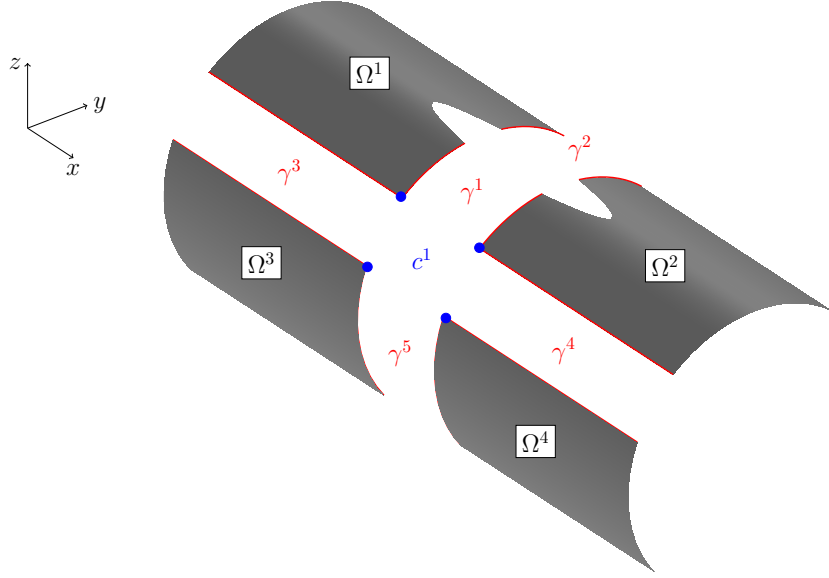


Figure 6.1 – Example of four subdomains $\Omega^i, i = 1, \dots, 4$ with their coupling interfaces $\gamma^\ell, \ell = 1, \dots, 5$, highlighted in red, and one corresponding cross-point $c^s, s = 1$, represented by blue dots. Note that we have separated the subdomains for visualization purposes.

coupling conditions:

$$u^m - u^n = 0 \quad \text{on} \quad \gamma^\ell, \quad (6.3a)$$

$$\nabla u^m \cdot \mathbf{n}^m + \nabla u^n \cdot \mathbf{n}^n = 0 \quad \text{on} \quad \gamma^\ell, \quad (6.3b)$$

which can be rewritten using the standard jump and normal jump operators, respectively, as:

$$[[u]] = 0 \quad \text{on} \quad \gamma^\ell, \quad (6.4a)$$

$$[[\nabla u]]_n = 0 \quad \text{on} \quad \gamma^\ell. \quad (6.4b)$$

Similarly for Kirchhoff-Love shells, let us denote by \mathbf{u}^m the value of the displacement field restricted to Ω^m , and similarly \mathbf{u}^n the value of the primary field on the neighboring subdomain Ω^n . Then, for each interface γ^ℓ the following coupling conditions must be satisfied:

$$\mathbf{u}^m - \mathbf{u}^n = 0 \quad \text{on} \quad \gamma^\ell, \quad (6.5a)$$

$$\theta_n(\mathbf{u}^m) + \theta_n(\mathbf{u}^n) = 0 \quad \text{on} \quad \gamma^\ell. \quad (6.5b)$$

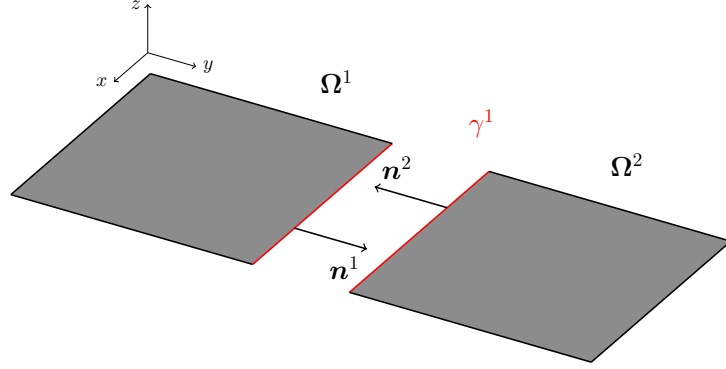


Figure 6.2 – Example of two subdomains $\Omega^i, i = 1, 2$ with their coupling interface $\gamma^\ell, \ell = 1$, highlighted in red, and their corresponding normal vectors $\mathbf{n}^i, i = 1, 2$. Note that we have separated the subdomains for visualization purposes.

Again, these equations can be rewritten by leveraging the jump operator as:

$$\llbracket \mathbf{u} \rrbracket = 0 \quad \text{on} \quad \gamma^\ell, \quad (6.6a)$$

$$\llbracket \theta_n(\mathbf{u}) \rrbracket = 0 \quad \text{on} \quad \gamma^\ell. \quad (6.6b)$$

6.1.2 The projected super-penalty formulation

In the following, we directly formulate our coupling strategy within the scope of trimming. We remark that the conforming case can be viewed as a subset of the trimmed case and therefore all concepts can be readily transferred. Let us now introduce for each patch Ω^i the following space:

$$Z_{i,h} = \text{span}\{b \in [\mathcal{S}_h^p(\Omega^i)]^d\}. \quad (6.7)$$

Additionally, we denote by $V_{i,h} \subset Z_{i,h}$ the finite-dimensional space given by the span of splines associated with subdomain Ω^i , where the exact definition of $V_{i,h}$ depends on the set of boundary conditions of the problem at hand. This allows us to introduce the following finite-dimensional space,

$$V_h = \left\{ v \in L^2(\Omega) \mid v \in V_{i,h} \quad \forall i = 1, \dots, N \text{ and } v \text{ is continuous in } c^s, s = 1, \dots, S \right\}, \quad (6.8)$$

where we highlight the C^0 -continuity requirement at the cross-points c^s . Furthermore, for each interface γ^ℓ , we introduce the associated knot vector Ξ^ℓ . The latter is constructed as follows. First, we arbitrarily choose one of the neighboring patches as active. Then, we build Ξ^ℓ by intersecting the knot lines of the active patch and γ^ℓ . We highlight that this operation can be performed directly in the parameter space of the active surface, since the B-Rep structure provides a representation of γ^ℓ in the parameter space of both

Chapter 6. Coupling of multi-patch Kirchhoff plates and Kirchhoff-Love shells

surfaces. For each surface, we denote the latter representation by $\hat{\gamma}^\ell(\hat{\Omega}^i)$, see Figure 6.4 for an example.

Remark 6.3 *At this stage, in the spirit of developing a simple and efficient method, we disregard the internal knots of the coupling curve for the construction of Ξ^ℓ . We highlight that the number of these knots depends on the chosen tolerance in the CAD model, with this number being potentially large. We are aware that this choice could potentially yield a loss of optimality of the method, but for smooth interfaces this effect is negligible. We verify this numerically on two trimmed patches in Figure 6.3, coupled along a C^1 -continuous quadratic B-spline curve. Indeed, for $p = 2, 3$ the results are practically indistinguishable, whereas only minor differences are present for the case $p = 4$. Although outside the scope of this work, finding a simple way to remove this source of sub-optimality constitutes a future research direction.*

Then, we build the isogeometric space $\mathcal{S}_h^{p-2}(\gamma^\ell)$ leveraging the $p/p-2$ pairing. Assuming B-splines of maximum smoothness, this space is obtained by removing from Ξ^ℓ the first and last two knots, where an illustrative example is given in Figure 6.4 for bivariate B-splines of degree $p = 2$ and corresponding $p-2 = 0$ degree-reduced splines defined on the interface knot vector $\Xi^\ell = [0 \ 1/3 \ 2/3 \ 1]$.

Remark 6.4 *The $p/p-2$ pairing has been proven to be inf-sup stable in the context of isogeometric mortar methods in [Brivadis et al., 2015]. Although its stability for trimmed geometries has not been rigorously studied, we verify numerically its applicability to the coupling of trimmed Kirchhoff plate and Kirchhoff-Love shells.*

Lastly, let us define the following space:

$$Q_h = \left\{ \mu \in L^2(\Gamma) \mid \mu \in \mathcal{S}_h^{p-2}(\gamma^\ell) \ \forall \ell = 1, \dots, L \right\}, \quad (6.9)$$

which is used to characterize the Lagrange multipliers associated with the coupling conditions.

The Kirchhoff-Love shell saddle point problem

We are now ready to define the discretized version of the singularly-perturbed saddle point problem associated with the Kirchhoff-Love shell. Without loss of generality, let us consider homogeneous Neumann-type boundary conditions. Then, the saddle problem

6.1. The projected super-penalty method

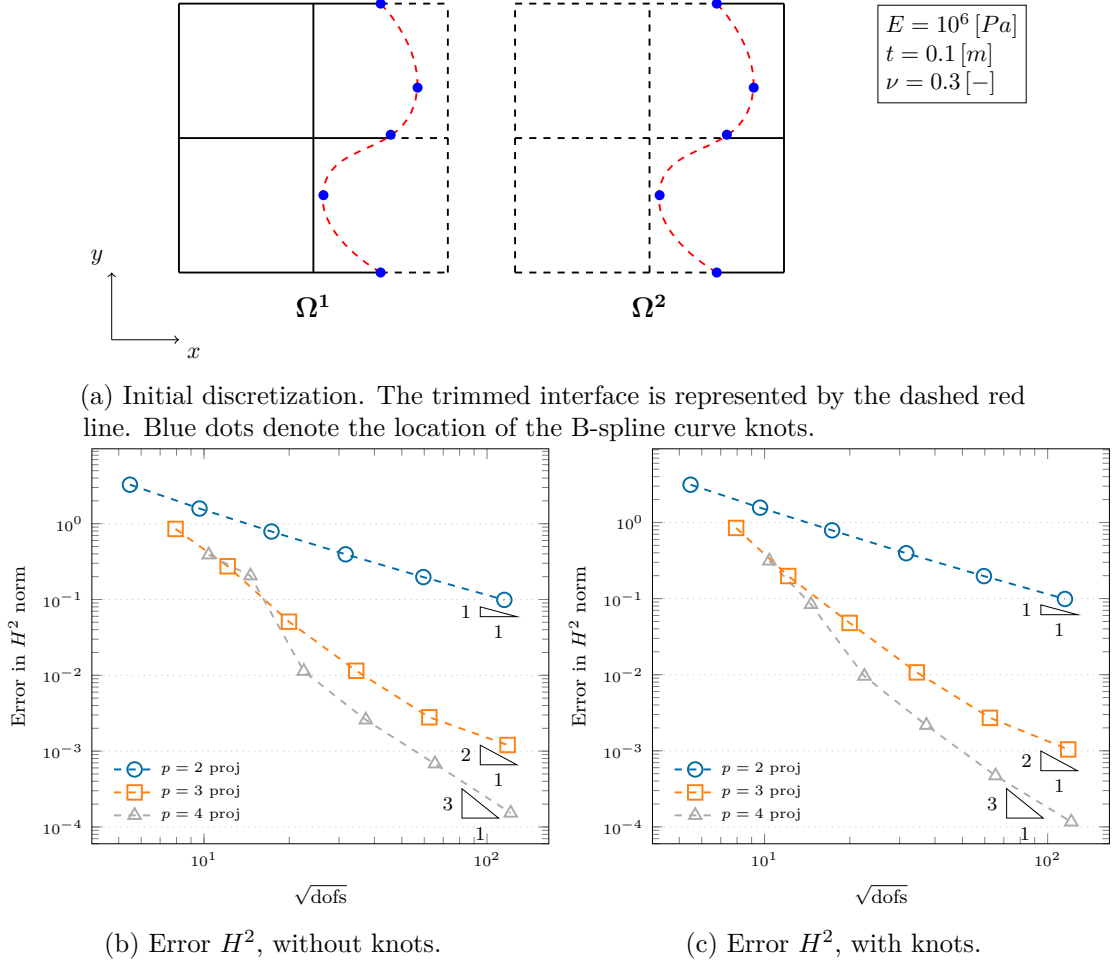


Figure 6.3 – Influence of internal knots of the coupling curve on a two trimmed plates example.

reads: find $(\mathbf{u}_h, \boldsymbol{\lambda}_{1,h}, \lambda_{2,h}) \in V_h \times [Q_h]^d \times Q_h$ such that:

$$\sum_{i=1}^N \left(\int_{\Omega^i} A(\mathbf{u}_h) : \alpha(\mathbf{v}_h) + B(\mathbf{u}_h) : \beta(\mathbf{v}_h) \right) + \sum_{\ell=1}^L \left(\int_{\gamma^\ell} \llbracket \mathbf{v}_h \rrbracket \cdot \boldsymbol{\lambda}_{1,h} + \int_{\gamma^\ell} \llbracket \theta_n(\mathbf{v}_h) \rrbracket \lambda_{2,h} \right) = (\mathbf{f}, \mathbf{v}_h) \quad \forall \mathbf{v}_h \in V_h, \quad (6.10a)$$

$$\sum_{\ell=1}^L \left(\int_{\gamma^\ell} \llbracket \mathbf{u}_h \rrbracket \cdot \boldsymbol{\mu}_{1,h} - \frac{1}{\alpha_{\text{disp}}^\ell} \int_{\gamma^\ell} \boldsymbol{\lambda}_{1,h} \cdot \boldsymbol{\mu}_{1,h} \right) = 0 \quad \forall \boldsymbol{\mu}_{1,h} \in [Q_h]^d, \quad (6.10b)$$

$$\sum_{\ell=1}^L \left(\int_{\gamma^\ell} \llbracket \theta_n(\mathbf{u}_h) \rrbracket \mu_{2,h} - \frac{1}{\alpha_{\text{rot}}^\ell} \int_{\gamma^\ell} \lambda_{2,h} \mu_{2,h} \right) = 0 \quad \forall \mu_{2,h} \in Q_h, \quad (6.10c)$$

where we have introduced the parameters $\alpha_{\text{disp}}^\ell$ and α_{rot}^ℓ corresponding to the displacements and normal rotations, respectively. As highlighted in [Herrema et al., 2019; Pasch

Chapter 6. Coupling of multi-patch Kirchhoff plates and Kirchhoff-Love shells

et al., 2021], these coefficients depend in general on the problem definition, e.g. the material parameters, the thickness of the shell, the applied boundary conditions, the mesh size and discretization degree, where a precise definition of our parameters will be provided in a later section. Let us now eliminate the Lagrange multipliers and rewrite (6.10a) only in terms of the displacement field. In particular, rearranging the second and third equations we obtain:

$$\lambda_{1,h}|\gamma^\ell = \alpha_{\text{disp}}^\ell \Pi^\ell \llbracket \mathbf{u}_h \rrbracket, \quad (6.11a)$$

$$\lambda_{2,h}|\gamma^\ell = \alpha_{\text{rot}}^\ell \Pi^\ell \llbracket \theta_n(\mathbf{u}_h) \rrbracket, \quad (6.11b)$$

where, with a slight abuse of notation, Π^ℓ stands for the L^2 -projection, defined on the interface γ^ℓ , onto the degree-reduced space $\left[\mathcal{S}_h^{p-2}(\gamma^\ell)\right]^d$ related to the displacements and onto the space $\mathcal{S}_h^{p-2}(\gamma^\ell)$ associated with the normal rotations, respectively. By substituting Equation (6.11) into the first line of (6.10) and leveraging the properties of the L^2 -projection, we obtain:

$$\begin{aligned} a_p(\mathbf{u}_h, \mathbf{v}_h) &= \sum_{i=1}^N a^{(i)}(\mathbf{u}_h, \mathbf{v}_h) + \\ &+ \sum_{\ell=1}^L \left(\int_{\gamma^\ell} \alpha_{\text{disp}}^{(\ell)} \Pi^\ell \llbracket \mathbf{v}_h \rrbracket \cdot \Pi^\ell \llbracket \mathbf{u}_h \rrbracket + \int_{\gamma^\ell} \alpha_{\text{rot}}^{(\ell)} \Pi^\ell \llbracket \theta_n(\mathbf{v}_h) \rrbracket \Pi^\ell \llbracket \theta_n(\mathbf{u}_h) \rrbracket \right). \end{aligned} \quad (6.12)$$

These coupling terms weakly impose the transmission conditions in (6.6) on the displacements and normal rotations, respectively.

Remark 6.5 *From a computational standpoint, we rewrite the coupling term associated with the rotations in (6.12) as defined in [Herrema et al., 2019], where the constraint is recast into two complementary terms. This ensures a non-zero penalty contribution for patches meeting at an arbitrary angle. Then, the L^2 -projection of these terms is performed. For further details, we refer to [Herrema et al., 2019] and references therein.*

Now, let us further characterize the aforementioned projection from a computational viewpoint. Let us consider a generic function $u \in V_h$ defined as the linear combination of basis functions and their corresponding coefficients $\hat{\mathbf{u}}$ as:

$$u = \sum_i \mathcal{B}_i \hat{u}_i \quad i = 1, \dots, \dim(V_h). \quad (6.13)$$

Similarly, its projection $\Pi^\ell(u)$ onto the space $\mathcal{S}_h^{p-2}(\gamma^\ell)$ can be written as another linear combination of spline functions and their associated coefficients $\tilde{\mathbf{u}}$:

$$\Pi^\ell(u) = \sum_j b_j \tilde{u}_j \quad j = 1, \dots, \dim(\mathcal{S}_h^{p-2}(\gamma^\ell)). \quad (6.14)$$

6.1. The projected super-penalty method

The orthogonality of the projection can now be expressed as:

$$\int_{\gamma^\ell} \Pi^\ell(u)b = \int_{\gamma^\ell} ub \quad \forall b \in \mathcal{S}_h^{p-2}(\gamma^\ell), \quad (6.15)$$

which can be rewritten in matrix form by substituting Equations (6.13) and (6.14) into Equation (6.15) as follows:

$$\mathcal{M}\tilde{\mathbf{u}} = \mathcal{F}\hat{\mathbf{u}}, \quad (6.16)$$

where \mathcal{M} denotes the mass matrix associated with the degree-reduced basis and \mathcal{F} represents the right-hand-side matrix corresponding to the inner product between the basis functions in $\mathcal{S}_h^{p-2}(\gamma^\ell)$ and V_h , respectively. In particular, for the projection of the displacement term introduced in Equation (6.11), $\mathcal{F}_{\text{disp}}$ is defined as the inner product between the splines in $[\mathcal{S}_h^{p-2}(\gamma^\ell)]^d$ and the jump of the basis functions in V_h . Analogously for the rotational term, \mathcal{F}_{rot} is assembled as the inner product between the basis functions in $\mathcal{S}_h^{p-2}(\gamma^\ell)$ and the jump of discrete normal rotations in V_h . Similarly, we distinguish between the mass matrix \mathcal{M} associated with the splines in $\mathcal{S}_h^{p-2}(\gamma^\ell)$ and its vectorial counterpart \mathbf{M} corresponding to the functions in $[\mathcal{S}_h^{p-2}(\gamma^\ell)]^d$. With these definitions at hand, we summarize the computation of the projected terms in Algorithm 6.

Algorithm 6 Computation of the penalty terms in Equation (6.12).

```

1: procedure COMPUTATION OF THE PENALTY TERMS
2:   for each interface  $\gamma^\ell$  in  $\Gamma$  do
3:     Build the spaces  $\mathcal{S}_h^{p-2}(\gamma^\ell)$  and  $[\mathcal{S}_h^{p-2}(\gamma^\ell)]^d$ 
4:     Build the intersection mesh for integration
5:      $\tilde{\mathbf{u}}_{\text{disp}} \leftarrow$  solve Equation (6.16) with  $\mathbf{M}$  and  $\mathcal{F}_{\text{disp}}$ 
6:      $\tilde{\mathbf{u}}_{\text{rot}} \leftarrow$  solve Equation (6.16) with  $\mathcal{M}$  and  $\mathcal{F}_{\text{rot}}$ 
7:      $\mathbf{K} = \mathbf{K} + \alpha_{\text{disp}}^{(\ell)} \tilde{\mathbf{u}}_{\text{disp}}^\top \mathbf{M} \tilde{\mathbf{u}}_{\text{disp}}$ 
8:      $\mathbf{K} = \mathbf{K} + \alpha_{\text{rot}}^{(\ell)} \tilde{\mathbf{u}}_{\text{rot}}^\top \mathcal{M} \tilde{\mathbf{u}}_{\text{rot}}$ 
9:   end for
10: end procedure

```

Lastly, we remark that the solution of Equation (6.16) is computationally inexpensive for B-splines of degree $p = 2, 3$ associated with a reduced space of degree $p - 2 = 0, 1$, respectively, for which the mass matrix is either diagonal or can be lumped.

The Kirchhoff plate saddle point problem

The same procedure used above can be applied to Kirchhoff plates. First, let us introduce the space:

$$X_{i,h} = \text{span}\{b \in \mathcal{S}_h^p(\Omega^i)\}, \quad (6.17)$$

where, with a slight abuse of notation, we denote by $V_{i,h} \subset X_{i,h}$ the finite-dimensional space given by the span of splines associated with subdomain Ω^i . Again, the exact definition of $V_{i,h}$ depends on the prescribed boundary conditions. Consequently, let us to introduce the following finite-dimensional space,

$$V_h = \left\{ v \in L^2(\Omega) \mid v \in V_{i,h} \forall i = 1, \dots, N \text{ and } v \text{ is continuous in } c^s, s = 1, \dots, S \right\}. \quad (6.18)$$

Then, the discrete singularly-perturbed Kirchhoff plate problem in mixed form reads: find $(u_h, \lambda_{1,h}, \lambda_{2,h}) \in V_h \times Q_h \times Q_h$ such that:

$$\begin{aligned} \sum_{i=1}^N \int_{\Omega^i} D \left[(1 - \nu) \nabla(\nabla v_h) : \nabla(\nabla u_h) + \nu \Delta v_h \Delta u_h \right] + \\ + \sum_{\ell=1}^L \left(\int_{\gamma^\ell} \llbracket v_h \rrbracket \lambda_{1,h} + \int_{\gamma^\ell} \llbracket \nabla v_h \rrbracket_n \lambda_{2,h} \right) = (f, v_h) \quad \forall v_h \in V_h, \end{aligned} \quad (6.19a)$$

$$\sum_{\ell=1}^L \left(\int_{\gamma^\ell} \llbracket u_h \rrbracket \mu_{1,h} - \frac{1}{\alpha_{\text{def}}^\ell} \int_{\gamma^\ell} \lambda_{1,h} \mu_{1,h} \right) = 0 \quad \forall \mu_{1,h} \in Q_h, \quad (6.19b)$$

$$\sum_{\ell=1}^L \left(\int_{\gamma^\ell} \llbracket \nabla u_h \rrbracket_n \mu_{2,h} - \frac{1}{\alpha_{\text{rot}}^\ell} \int_{\gamma^\ell} \lambda_{2,h} \mu_{2,h} \right) = 0 \quad \forall \mu_{2,h} \in Q_h, \quad (6.19c)$$

where α_{def}^ℓ and α_{rot}^ℓ are again “large” parameters related to the deflections and rotations, respectively. Similarly as before, we can formally eliminate the Lagrange multipliers and recast (6.19) into its primal form. Specifically, we can write:

$$\begin{aligned} \lambda_{1,h}|_{\gamma^\ell} &= \alpha_{\text{def}}^\ell \Pi^\ell \llbracket u_h \rrbracket \\ \lambda_{2,h}|_{\gamma^\ell} &= \alpha_{\text{rot}}^\ell \Pi^\ell \llbracket \nabla u_h \rrbracket_n, \end{aligned} \quad (6.20a)$$

where $\Pi^\ell : L^2(\gamma^\ell) \rightarrow \mathcal{S}_h^{p-2}(\gamma^\ell)$ denotes the L^2 -projection, associated with the interface γ^ℓ , onto the reduced space $\mathcal{S}_h^{p-2}(\gamma^\ell)$. Finally, employing the previous results and the properties of the L^2 -projection, the resulting discretized bilinear form, augmented by suitable penalty terms that weakly enforce the coupling conditions (6.4), reads: find

$u_h \in V_h$ such that:

$$\begin{aligned} & \sum_{i=1}^N \int_{\Omega^i} D \left[(1 - \nu) \nabla(\nabla v_h) : \nabla(\nabla u_h) + \nu \Delta v_h \Delta u_h \right] + \\ & + \sum_{\ell=1}^L \left(\alpha_{\text{def}}^\ell \int_{\gamma^\ell} \Pi^\ell \llbracket u_h \rrbracket \Pi^\ell \llbracket v_h \rrbracket + \alpha_{\text{rot}}^\ell \int_{\gamma^\ell} \Pi^\ell \llbracket \nabla u_h \rrbracket_n \Pi^\ell \llbracket \nabla v_h \rrbracket_n \right) = (f, v_h) \quad \forall v_h \in V_h. \end{aligned} \quad (6.21a)$$

Selection of penalty parameters

It is well-known that the perturbed problem (6.10) is variationally consistent only if we select $\alpha_{\text{disp}}^\ell = \alpha_{\text{rot}}^\ell \rightarrow \infty$ $\ell = 1, \dots, L$. However, the well-posedness of the underlying problem is insensitive to the choice of the parameters $\alpha_{\text{disp}}^\ell$ and α_{rot}^ℓ . Therefore, our method is inherently free from boundary locking, independently of the choice of penalty values. This allows us to select $\alpha_{\text{disp}}^\ell$ and α_{rot}^ℓ to guarantee the high-order convergence rates achievable by B-splines. Furthermore, in the spirit of developing a parameter-free penalty method, we modify the choice proposed in [Herrema et al., 2019], scaling the displacement and rotation penalty parameters by the physical constants of the underlying problem, the local mesh size, the spline degree and the geometry. For homogeneous isotropic materials they read:

$$\alpha_{\text{disp}}^\ell = (|\gamma^\ell|)^{\beta-1} \frac{Et}{(h_\ell)^\beta (1 - \nu^2)}, \quad (6.22a)$$

$$\alpha_{\text{rot}}^\ell = (|\gamma^\ell|)^{\beta-1} \frac{Et^3}{12(h_\ell)^\beta (1 - \nu^2)}, \quad (6.22b)$$

where the measure of γ^ℓ serves as a characteristic length and the exponent β is chosen solely to ensure the optimal convergence of the method. Therefore, it must be a function of the degree p of the underlying discretization. Numerically we have observed that the scaling factor $\beta = p - 1$ in (6.22) is necessary to attain optimal convergence of the method in the H^2 norm, whereas for a scaling of $\beta = p$ we noticed optimality in the H^2 and H^1 norms. Finally, a factor of $\beta = p + 1$ provides optimality in the H^2 , H^1 and L^2 norms. If not stated otherwise, we will use $\beta = p + 1$ in all our numerical examples. In case of orthotropic laminates, we adapt the minimum strategy presented in [Herrema et al., 2019], where the minimum local stiffness between adjacent patches Ω^m and Ω^n is used. Consequently, the penalty parameters are defined as:

$$\begin{aligned} \alpha_{\text{disp}}^\ell &= (|\gamma^\ell|)^{\beta-1} \frac{\min(\max_{i,j}(\mathbb{A}_{ij}^{(m)}), \max_{i,j}(\mathbb{A}_{ij}^{(n)}))}{(h_\ell)^\beta} \\ \alpha_{\text{rot}}^\ell &= (|\gamma^\ell|)^{\beta-1} \frac{\min(\max_{i,j}(\mathbb{D}_{ij}^{(m)}), \max_{i,j}(\mathbb{D}_{ij}^{(n)}))}{(h_\ell)^\beta}. \end{aligned} \quad (6.23)$$

Note that all of these parameters are known and depend only on the problem definition, meaning that no user-defined factor is required. Moreover, it is straightforward to check that the penalty terms are dimensionally consistent with respect to their corresponding energy contribution in the weak forms (6.12) and (6.21), respectively.

Remark 6.6 *Clearly, the choice of β influences the condition number of the associated system matrix. This, together with small trimmed elements, can potentially yield ill-conditioned systems of equations and, consequently, loss of accuracy due to numerical round-off errors. In the context of trimmed single-patch shells, a possible remedy based on extended B-splines has been studied in [Schöllhammer et al., 2020]. Furthermore, in the scope of immersed methods, an ad-hoc multigrid preconditioner has been developed in [de Preter et al., 2020b]. In all trimmed examples presented in this chapter, we employ again a direct solver where the stiffness matrix is preconditioned by a simple diagonal scaling. This seems to suffice for the level of accuracy reached in our numerical experiments. We remark that a thorough study of the condition number in the context of trimmed multi-patch Kirchhoff-Love shells constitutes a future research direction. Conversely, for conforming Kirchhoff plates, we present an ad-hoc preconditioner that mitigates the issue.*

Cross-points modification

In the literature of mortar methods, it is well-known that the treatment of cross-points requires extra considerations, see [Dittmann et al., 2020] and references therein for a discussion in the context of mortar coupling of isogeometric multi-patches. Analogously, our method also inherits the need for a cross-points modification. Indeed, in order to retain optimality of the method, a linear constraint must be imposed to the control variables meeting at the cross-point to ensure C^0 -continuity. An example with four patches is depicted in Figure 6.5, where in Figure 6.5a we depict the dofs associated with each coupling interface and in Figure 6.5b we visualize the imposition of the constraint. To explain the procedure, let us start from the following unconstrained system of equations:

$$\mathcal{A}\mathbf{u}_h = \mathbf{f}. \quad (6.24)$$

Now, the constraint can be incorporated easily into the standard linear system in a fully algebraic fashion, where a possible implementation is presented in Algorithm 7.

6.1. The projected super-penalty method

Algorithm 7 Algorithm for applying a C^0 constraint at a cross-point.

- 1: **procedure** APPLY_ C^0 _CONSTRAINT(vector of dofs at cross-points \mathbf{u}_{cp})
 - 2: Label one dof in \mathbf{u}_{cp} as active
 - 3: Label the remaining dofs in \mathbf{u}_{cp} as inactive
 - 4: Build the rectangular matrix \mathcal{C} representing the linear active-inactive constraints (see (6.25))
 - 5: Solve the reduced system $\hat{\mathcal{A}}\hat{\mathbf{u}}_h = \hat{\mathbf{f}}$, where $\hat{\mathcal{A}} = \mathcal{C}^\top \mathcal{A} \mathcal{C}$ and $\hat{\mathbf{f}} = \mathcal{C}^\top \mathbf{f}$
 - 6: Recover the solution \mathbf{u}_h from $\mathbf{u}_h = \mathcal{C}\hat{\mathbf{u}}_h$
 - 7: **end procedure**
-

The construction of the rectangular matrix \mathcal{C} is best explained with an example. Let us assume that the dofs at the cross-point are numbered as $\mathbf{u}_{\text{cp}} = [u_{\text{cp}1} \ u_{\text{cp}2} \ u_{\text{cp}3} \ u_{\text{cp}4}]$. Now, without loss of generality, we pick $u_{\text{cp}1}$ as the active control point and the rest as inactive nodes. Then, the constraint can be expressed via the matrix \mathcal{C} as follows:

$$\mathbf{u}_h = \begin{bmatrix} u_1 \\ \vdots \\ u_{\text{cp}1} \\ \vdots \\ u_{\text{cp}2} \\ \vdots \\ u_{\text{cp}3} \\ \vdots \\ u_{\text{cp}4} \\ \vdots \\ u_{\text{ndof}} \end{bmatrix} = \begin{matrix} u_1 \\ \vdots \\ u_{\text{cp}1} \\ \vdots \\ u_{\text{cp}2} \\ \vdots \\ u_{\text{cp}3} \\ \vdots \\ u_{\text{cp}4} \\ \vdots \\ u_{\text{ndof}} \end{matrix} \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ & \ddots & & & \\ 0 & 0 \dots & 1 & 0 \dots & 0 \\ & \ddots & & & \\ 0 & 0 \dots & 1 & 0 \dots & 0 \\ & \ddots & & & \\ 0 & 0 \dots & 1 & 0 \dots & 0 \\ & \ddots & & & \\ 0 & 0 \dots & 1 & 0 \dots & 0 \\ & \ddots & & & \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{bmatrix} u_1 \\ \vdots \\ u_{\text{cp}1} \\ \vdots \\ u_{\text{ndof}} \end{bmatrix} = \mathcal{C}\hat{\mathbf{u}}_h, \quad (6.25)$$

where ndof denotes the total number of degrees-of-freedom in the system. This procedure eliminates the unknowns associated with the inactive nodes from the system.

Remark 6.7 *Note that, as we only require C^0 -continuity at the cross-point, the valence of the point does not pose any additional conceptual challenge to the method.*

In case of geometrically non-conforming¹ or trimmed coupling interfaces, it is not possible to strongly impose the constraint with the method presented above. Possible remedies include enforcing the constraint weakly via the penalty method or via Lagrange

¹Similarly to [Brivadis et al., 2015], we define an interface as geometrically conforming if the pull-back with respect to both active and inactive domains is an entire edge of each parametric domain $\hat{\Omega}^i$.

multipliers [Felippa, 2003]. For robustness and ease of implementation, we employ the penalty method. Specifically, the unconstrained system in Equation (6.24) is modified as explained in the following. For simplicity, let us consider the constraint $u_{cp1} = u_{cp2}$, the generalization to more dofs is straightforward. Now, we can write the penalty matrix associated with the constraint as:

$$\alpha \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} u_{cp1} \\ u_{cp2} \end{bmatrix} = \mathbf{0}, \quad (6.26)$$

where α denotes the penalty parameter. This contribution can be then assembled into the global stiffness matrix. In our case, each dof involved in the definition of the constraint is scaled by its corresponding basis functions as:

$$\mathcal{B}_1 u_{cp1} + \mathcal{B}_2 u_{cp2} + \dots + \mathcal{B}_n u_{cpn} = 0. \quad (6.27)$$

Consequently, the penalty stiffness in Equation (6.26) is modified as follows:

$$\alpha \begin{bmatrix} \mathcal{B}_1 & \mathcal{B}_2 & \dots & \mathcal{B}_n \end{bmatrix} \begin{bmatrix} \mathcal{B}_1 \\ \mathcal{B}_2 \\ \dots \\ \mathcal{B}_n \end{bmatrix} \begin{bmatrix} u_{cp1} \\ u_{cp2} \\ \dots \\ u_{cpn} \end{bmatrix} = \mathbf{0}. \quad (6.28)$$

6.2 A nested preconditioner based on the Schur Complement Reduction for non-trimmed Kirchhoff plates

In this section, following the notation introduced in [Quarteroni et al., 2000] and building upon the work presented in [Liu and Marsden, 2019; Liu et al., 2020; Pegolotti et al., 2021] in the context of elastodynamics and hemodynamics, we present an efficient way to mitigate the detrimental effects on the condition number stemming from our choice of super-penalty parameters. This preconditioner is based on the approximate solution of the block factorization of the system matrix known as Schur Complement Reduction (SCR). We remind the reader that before performing the algorithm described in the following, we apply a symmetric diagonal scaling to the system matrix.

6.2.1 The Schur Complement Reduction

We begin by reordering the matrix $\mathcal{A} \in \mathbb{R}^{\text{ndof} \times \text{ndof}}$ stemming from (6.21) in blocks as follows:

$$\mathcal{A} = \begin{bmatrix} \mathbf{A}_{i,i} & \mathbf{B}_{i,\Gamma} \\ \mathbf{B}_{i,\Gamma}^\top & \mathbf{C}_{\Gamma,\Gamma} \end{bmatrix}, \quad (6.29)$$

6.2. A nested preconditioner based on the Schur Complement Reduction for non-trimmed Kirchhoff plates

where the subscripts i and Γ refer to internal and interface dofs, respectively, where an example is depicted in Figure 6.6. Let us remark that $\mathbf{A}_{i,i}$ is a block-diagonal matrix where every block is the matrix associated with an homogeneous Dirichlet problem (fully clamped) on the corresponding patch Ω^i . Moreover, with a slight abuse of notation, we assume that, if needed, \mathcal{A} has already been modified to account for the C^0 -constraint related to the cross-points. Now, we can perform the following block factorization of \mathcal{A} :

$$\mathcal{A} = \mathcal{L}\mathcal{D}\mathcal{U} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{B}_{i,\Gamma}^\top \mathbf{A}_{i,i}^{-1} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{A}_{i,i} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_{\Gamma,\Gamma} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{A}_{i,i}^{-1} \mathbf{B}_{i,\Gamma} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}, \quad (6.30)$$

where we have introduced the Schur complement $\mathbf{S}_{\Gamma,\Gamma} := \mathbf{C}_{\Gamma,\Gamma} - \mathbf{B}_{i,\Gamma}^\top \mathbf{A}_{i,i}^{-1} \mathbf{B}_{i,\Gamma}$. Multiplying by \mathcal{L} on both sides we get:

$$\begin{bmatrix} \mathbf{A}_{i,i} & \mathbf{B}_{i,\Gamma} \\ \mathbf{0} & \mathbf{S}_{\Gamma,\Gamma} \end{bmatrix} \begin{bmatrix} \mathbf{x}_i \\ \mathbf{x}_\Gamma \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{B}_{i,\Gamma}^\top \mathbf{A}_{i,i}^{-1} & \mathbf{I} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{r}_i \\ \mathbf{r}_\Gamma \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{B}_{i,\Gamma}^\top \mathbf{A}_{i,i}^{-1} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{r}_i \\ \mathbf{r}_\Gamma \end{bmatrix} = \begin{bmatrix} \mathbf{r}_i \\ \mathbf{r}_\Gamma - \mathbf{B}_{i,\Gamma}^\top \mathbf{A}_{i,i}^{-1} \mathbf{r}_i \end{bmatrix}. \quad (6.31)$$

We highlight that, up to this point, this factorization is performed in exact algebra. Then, from (6.31), we can solve for \mathbf{x} in a segregated fashion by exploiting Algorithm 8.

Algorithm 8 SCR algorithm

1: **procedure** SOLUTION OF $\mathcal{A}\mathbf{x} = \mathbf{r}$ BASED ON SCR

2: Solve for an intermediate solution $\hat{\mathbf{x}}_i$

$$\mathbf{A}_{i,i} \hat{\mathbf{x}}_i = \mathbf{r}_i \quad (6.32)$$

3: Update the interface residual $\mathbf{r}_\Gamma = \mathbf{r}_\Gamma - \mathbf{B}_{i,\Gamma}^\top \hat{\mathbf{x}}_i$

4: Solve for the interface solution \mathbf{x}_Γ from the Schur equation

$$\mathbf{S}_{\Gamma,\Gamma} \mathbf{x}_\Gamma = \mathbf{r}_\Gamma \quad (6.33)$$

5: Update the internal residual $\mathbf{r}_i = \mathbf{r}_i - \mathbf{B}_{i,\Gamma} \mathbf{x}_\Gamma$

6: Solve for the internal solution \mathbf{x}_i from

$$\mathbf{A}_{i,i} \mathbf{x}_i = \mathbf{r}_i \quad (6.34)$$

7: **end procedure**

In practice, the Schur complement $\mathbf{S}_{\Gamma,\Gamma}$ is expensive and often infeasible to compute explicitly. A way around this issue is given in Algorithm 9, where we summarize a matrix-free procedure to apply the Schur complement to a vector.

Algorithm 9 Algorithm for applying the Schur complement to a vector

- 1: **procedure** APPLICATION OF $\mathbf{S}_{\Gamma,\Gamma}$ TO A VECTOR \mathbf{x}_Γ
- 2: Compute the matrix-vector multiplication $\hat{\mathbf{x}}_\Gamma = \mathbf{C}_{\Gamma,\Gamma}\mathbf{x}_\Gamma$
- 3: Compute the matrix-vector multiplication $\bar{\mathbf{x}}_\Gamma = \mathbf{B}_{i,\Gamma}\mathbf{x}_\Gamma$
- 4: Solve for an intermediate solution $\tilde{\mathbf{x}}_\Gamma$ from

$$\mathbf{A}_{i,i}\tilde{\mathbf{x}}_\Gamma = \bar{\mathbf{x}}_\Gamma \quad (6.35)$$

- 5: Compute the matrix-vector multiplication $\bar{\mathbf{x}}_\Gamma = \mathbf{B}_{i,\Gamma}^\top \tilde{\mathbf{x}}_\Gamma$
 - 6: Return $\hat{\mathbf{x}}_\Gamma - \bar{\mathbf{x}}_\Gamma$
 - 7: **end procedure**
-

Remark 6.8 As noted in [Liu and Marsden, 2019], the cost of the preconditioner is often dominated by the solution of the Schur system (6.33). To reduce the computational burden of this step, we use as preconditioner a coarse approximation of the Schur complement obtained by applying only a few iterations of GMRES to $\mathbf{A}_{i,i}$ for assembling $\tilde{\mathbf{S}}_{\Gamma,\Gamma} = \mathbf{C}_{\Gamma,\Gamma} - \mathbf{B}_{i,\Gamma}^\top \tilde{\mathbf{A}}_{i,i}^{-1} \mathbf{B}_{i,\Gamma}$. Although this choice works reasonably well for our numerical examples, we remark that more research is needed to find a robust (both in h and p) and scalable preconditioner for the Schur complement and, more in general, for fourth-order PDEs.

6.2.2 Nested block preconditioner strategy based on SCR

The main idea presented in [Liu and Marsden, 2019] is to combine the robustness of the SCR factorization with the ease of application of a block preconditioners (such as SIMPLE or variants thereof [Quarteroni et al., 2000]). Indeed, we can build a preconditioner \mathcal{P}_{SCR} based on an approximate factorization of (6.31), where Equations (6.32) to (6.34) are solved within a prescribed tolerance. Given that \mathcal{P}_{SCR} changes its algebraic definition at every iteration, following [Liu and Marsden, 2019], we employ a flexible GMRES algorithm (FGMRES) [Saad, 1993] as the iterative method for the most outer solve $\mathcal{A}\mathbf{x} = \mathbf{r}$. At each iteration of FGMRES, we can apply the preconditioner \mathcal{P}_{SCR} via Algorithm 8, where this entails the solution of the blocks $\mathbf{A}_{i,i}$ and $\mathbf{S}_{\Gamma,\Gamma}$, respectively. This part of the algorithm is denoted as intermediate solver. Last, since we do not assemble the Schur complement explicitly, but we apply its action on a vector through Algorithm 9, we perform a final solve for $\mathbf{A}_{i,i}$ in (6.35), denoted as inner solver. The final performance of the preconditioner is therefore determined by the prescribed tolerances for the outer, intermediate and inner layers, respectively, where the objective is finding a good balance between the computational cost and the robustness of the method. In the following, we denote the aforementioned tolerances by η_o, η_t and η_n for the outer, intermediate and inner layers, respectively.

6.2. A nested preconditioner based on the Schur Complement Reduction for non-trimmed Kirchhoff plates

A preconditioner based of the Fast Diagonalization (FD) algorithm

Since each outer iteration of the nested preconditioner is based on the solution of three systems involving the block $\mathbf{A}_{i,i}$, an efficient and robust preconditioner for this block is required. In this work we extend the isogeometric preconditioner studied in [Sangalli and Tani, 2016; Montardini et al., 2018b], based on the Fast Diagonalization algorithm, to the Kirchhoff plate problem. In the following, we focus our derivation on the single-patch case. By construction, the extension to the multi-patch case is straightforward since the block $\mathbf{A}_{i,i}$ is formed by disjoint sub-blocks associated with each patch Ω^i . Now, exploiting the tensor product structure of the B-spline basis at the patch level, let us introduce the preconditioner \mathcal{P}_{FD} in Kronecker form as:

$$\mathcal{P}_{\text{FD}} = M_1 \otimes K_2 + K_1 \otimes M_2, \quad (6.36)$$

where M_k and K_k with $k = 1, 2$ refer to the one-dimensional, parametric mass and hessian matrices associated with the k -th parametric dimension, respectively. They can be expanded as follows:

$$\begin{aligned} [M_k]_{i,j} &= \int_0^1 b_{i,p}(\eta_k) b_{j,p}(\eta_k) \, d\eta_k \\ [K_k]_{i,j} &= \int_0^1 b''_{i,p}(\eta_k) b''_{j,p}(\eta_k) \, d\eta_k, \end{aligned} \quad (6.37)$$

where b indicates the univariate B-spline basis functions. Then, analogously to [Montardini et al., 2018a], we partially include the geometry and physical coefficients inside the preconditioner. In particular, let us denote by \mathfrak{C} the following function:

$$\mathfrak{C}(\boldsymbol{\eta}) = D \left(\left\| J_{\mathbf{F}}^{-1} \right\|_2 \right)^4 |\det (J_{\mathbf{F}})|, \quad (6.38)$$

where we recall that $J_{\mathbf{F}}$ represents the jacobian of the B-spline parametrization \mathbf{F} and D is the flexural stiffness of the plate. Now, as explained in [Montardini et al., 2018a, Appendix A.3], we perform a separation of variables on \mathfrak{C} such that we can write:

$$\mathfrak{C}(\boldsymbol{\eta}) \approx \tilde{\mathfrak{C}}(\boldsymbol{\eta}) = \begin{bmatrix} \omega_1(\eta_1)\tau_2(\eta_2) & 0 \\ 0 & \tau_1(\eta_1)\omega_2(\eta_2) \end{bmatrix}, \quad (6.39)$$

where this matrix is evaluated at each quadrature point. Details on the algorithm used to perform this separation can be found in [Diliberto and Straus, 1951; Wachspress, 2013]. With this, we can modify the preconditioner given in (6.36) to partially account for the geometry and coefficients information as follows:

$$\mathcal{P}_{\text{FD}}^{\mathbf{F}} = \widetilde{M}_1 \otimes \widetilde{K}_2 + \widetilde{K}_1 \otimes \widetilde{M}_2, \quad (6.40)$$

where

$$\begin{aligned}\left[\widetilde{M}_k\right]_{i,j} &= \int_0^1 \omega_k(\eta_k) b_{i,p}(\eta_k) b_{j,p}(\eta_k) \, d\eta_k \\ \left[\widetilde{K}_k\right]_{i,j} &= \int_0^1 \tau_k(\eta_k) b''_{i,p}(\eta_k) b''_{j,p}(\eta_k) \, d\eta_k.\end{aligned}\tag{6.41}$$

Finally, each iteration of the iterative solver requires the solution of the following system:

$$\mathcal{P}_{\text{FD}}^{\mathbf{F}} s = r, \tag{6.42}$$

where r denotes the current residual. Due to the tensor structure of the preconditioner, we can rewrite (6.42) as a Sylvester matrix equation [Simoncini, 2016]:

$$\widetilde{M}_2 S \widetilde{K}_1 + \widetilde{K}_2 S \widetilde{M}_1 = R, \tag{6.43}$$

where $s = \text{vec}(S)$ and $r = \text{vec}(R)$.

Remark 6.9 *Let us recall that for any matrix $Z \in \mathbb{R}^{r \times c}$ the operator $\text{vec}(Z)$ gives as output the vector $z \in \mathbb{R}^{rc}$ formed by stacking the columns of Z .*

Let us now consider the generalized eigendecomposition of the matrix pencils $(\widetilde{K}_1, \widetilde{M}_1)$ and $(\widetilde{K}_2, \widetilde{M}_2)$, respectively, as:

$$\begin{aligned}\widetilde{K}_1 U_1 &= \widetilde{M}_1 U_1 D_1 \\ \widetilde{K}_2 U_2 &= \widetilde{M}_2 U_2 D_2.\end{aligned}\tag{6.44}$$

Here, D_1 and D_2 are diagonal matrices containing the eigenvalues of $\widetilde{M}_1^{-1} \widetilde{K}_1$ and $\widetilde{M}_2^{-1} \widetilde{K}_2$, respectively. Further, U_1 and U_2 are defined as:

$$\begin{aligned}U_1^\top \widetilde{M}_1 U_1 &= \mathbf{I} \\ U_2^\top \widetilde{M}_2 U_2 &= \mathbf{I}.\end{aligned}\tag{6.45}$$

With these definitions at hand, we can rewrite (6.40) in Kronecker form as:

$$(U_1 \otimes U_2)^{-\top} (D_1 \otimes \mathbf{I} + \mathbf{I} \otimes D_2) (U_1 \otimes U_2)^{-1} s = r, \tag{6.46}$$

where the preconditioner can be efficiently applied via Algorithm 10.

6.2. A nested preconditioner based on the Schur Complement Reduction for non-trimmed Kirchhoff plates

Algorithm 10 FD method for applying $\mathcal{P}_{\text{FD}}^{\mathbf{F}}$

- 1: **procedure** UPDATE OF THE ITERATION RESIDUAL VIA THE FD METHOD
 - 2: Compute the generalized eigendecomposition in (6.44)
 - 3: Compute the intermediate result $\tilde{r} = (U_1 \otimes U_2)^\top r$
 - 4: Compute the intermediate residual $\tilde{s} = (D_1 \otimes \mathbf{I} + \mathbf{I} \otimes D_2)^{-1} \tilde{r}$
 - 5: Return $s = (U_1 \otimes U_2) \tilde{s}$
 - 6: **end procedure**
-

Remark 6.10 *We remark that the application of the nested preconditioner \mathcal{P}_{SCR} combined with $\mathcal{P}_{\text{FD}}^{\mathbf{F}}$ can be implemented in a fully matrix-free framework. Furthermore, although not investigated in this work, the patch-wise block structure of $\mathbf{A}_{i,i}$ could be further exploited for parallelization.*

For the sake of conciseness, we do not provide here further details of the FD algorithm, but we refer to [Sangalli and Tani, 2016; Montardini et al., 2018b; Loli et al., 2020, 2021] for a thorough theoretical and numerical investigation of the method in the scope of isogeometric analysis.

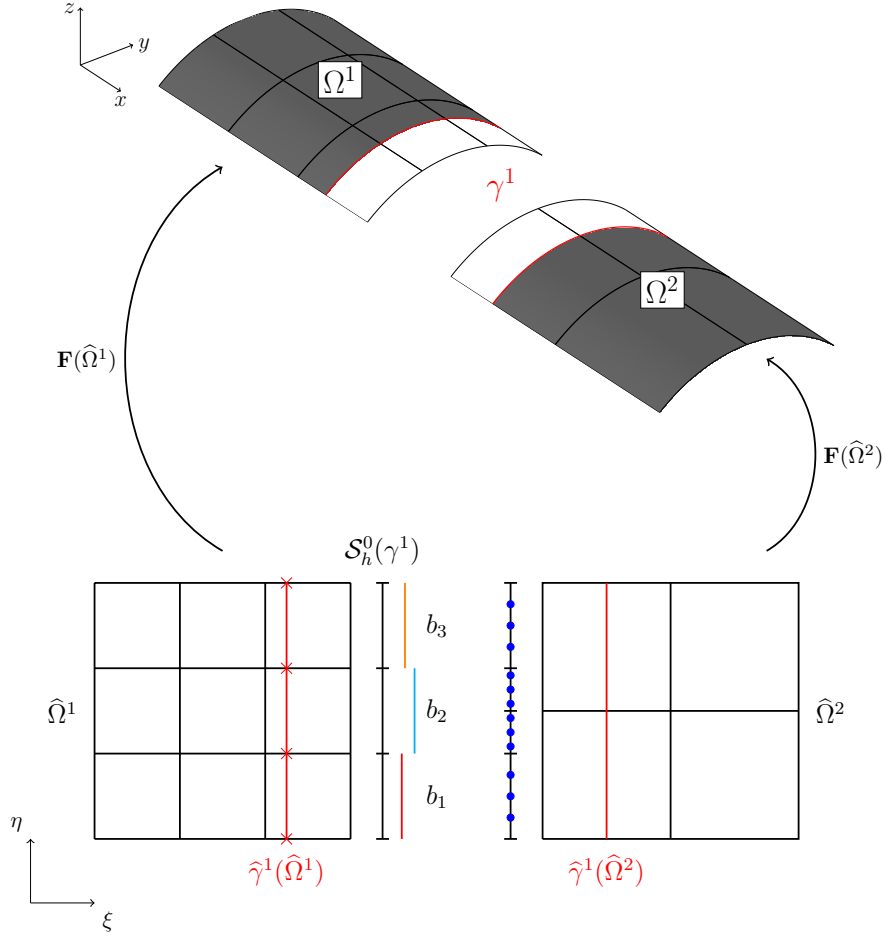


Figure 6.4 – Example of the projection setup on a coupling interface for B-splines of degree $p = 2$. We arbitrarily select the finer mesh (on Ω^1 in this example) to define the projection space $\mathcal{S}_h^{p-2}(\gamma^\ell)$, where the intersections between the parametric coupling curve $\hat{\gamma}^1(\hat{\Omega}^1)$ and the knot lines of Ω^1 are represented by red crosses. Additionally, an intersection mesh at the interface is created only for integration purposes to properly compute the projected penalty terms in Equation (6.12). The $p + 1$ integration points are schematically represented by blue dots. Note that we have separated the subdomains for visualization purposes.

6.2. A nested preconditioner based on the Schur Complement Reduction for non-trimmed Kirchhoff plates

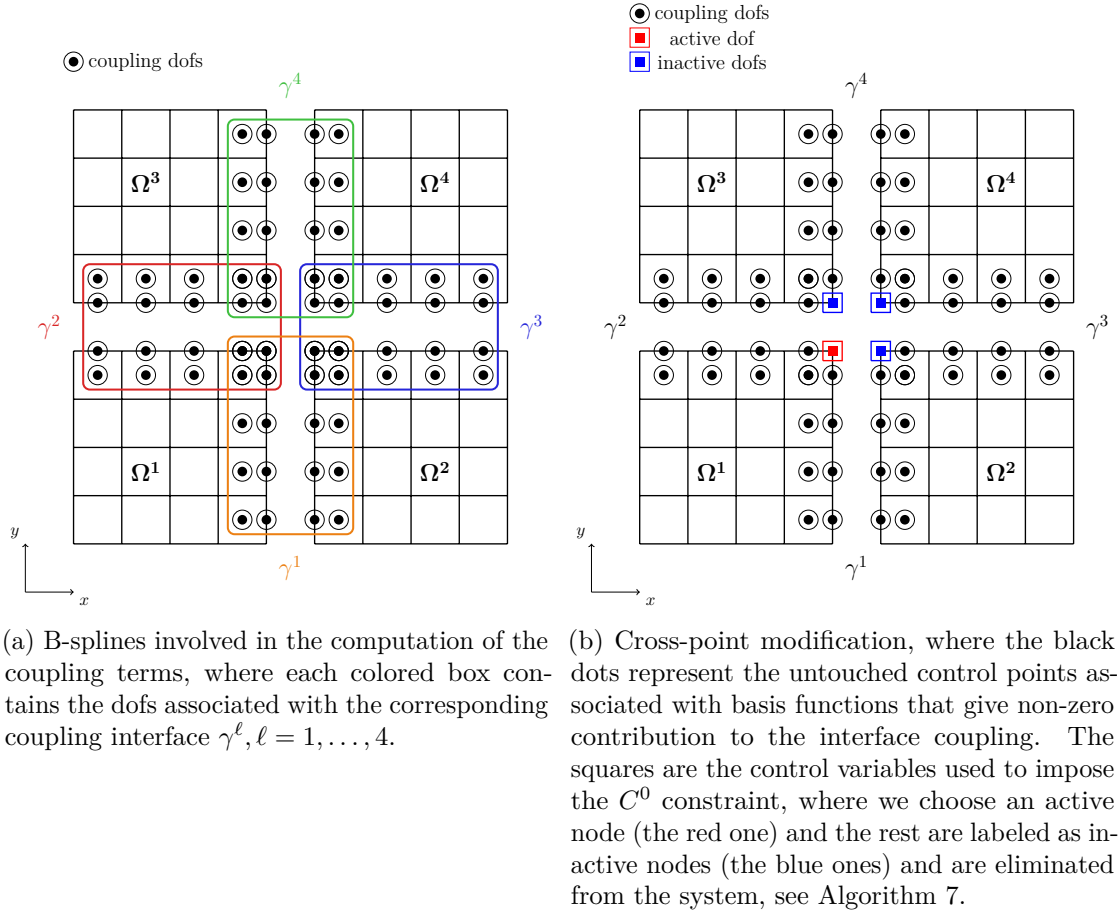


Figure 6.5 – Example of the dofs involved in the computation of the coupling integrals and cross-point modification in a four patches setup.

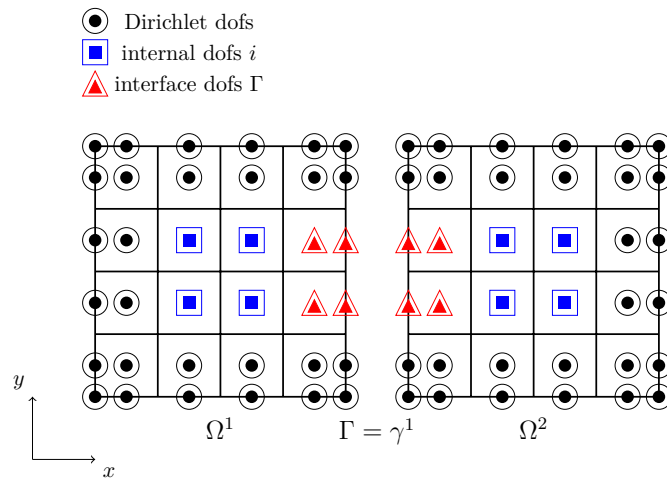


Figure 6.6 – Example of reordering of the dofs in a two patches setup, discretized by B-splines of degree $p = 2$, associated with the block system matrix \mathcal{A} .

6.3 Numerical examples

In this section we assess the performance of the proposed coupling technique with several numerical examples defined both on trimmed and untrimmed, non-conforming, multi-patch geometries. The analytical shell solutions are taken from the new shell obstacle course studied in [Benzaken et al., 2021], where the exact manufactured functions are evaluated in the freely-available Mathematica notebook² with 100 digits of precision. Moreover, similarly to [Benzaken et al., 2021], for every element we employ 25×25 quadrature points to properly capture the highly non-linearity of the quantities of interest. The results of these computations are then imported into *GeoPDEs*. Also, in all examples taken from [Benzaken et al., 2021], we derive from the manufactured solution and apply on the entire boundary $\partial\Omega$ non-homogeneous Dirichlet boundary conditions for the displacements and non-homogeneous Neumann boundary conditions for the bending moments, respectively. Finally, throughout this section, we compare our choice of penalty factors to a classical approach where the parameters are kept constant and are scaled by the Young's modulus:

$$\begin{aligned}\alpha_{\text{disp}}^\ell &= 10^3 E \\ \alpha_{\text{rot}}^\ell &= 10^3 E,\end{aligned}\tag{6.47}$$

and to the method proposed in [Herrema et al., 2019]:

$$\begin{aligned}\alpha_{\text{disp}}^\ell &= \delta \frac{Et}{(h_\ell)(1 - \nu^2)} \\ \alpha_{\text{rot}}^\ell &= \delta \frac{Et^3}{12(h_\ell)(1 - \nu^2)},\end{aligned}\tag{6.48}$$

where the problem-independent, user-defined parameter $\delta = 10^3$ has been numerically validated on an extensive series of benchmarks.

6.3.1 Coupling of Kirchhoff plates

We start by analyzing the coupling of planar surfaces where the underlying physics is described by the Kirchhoff plate problem.

A non-trimmed four-patches plate example

In this example we consider the computational domain $\Omega = [0, 2] \times [0, 2]$ depicted in Figure 6.7, split into four subdomains $\Omega^i, i = 1, \dots, 4$. We remark that all meshes are non-conforming at every coupling interface, as the irrational factor $\sqrt{2}/100$ has been used to shift the interface knots. The body load and boundary data are computed such

²<https://github.com/wdas/shell-obstacle-course>

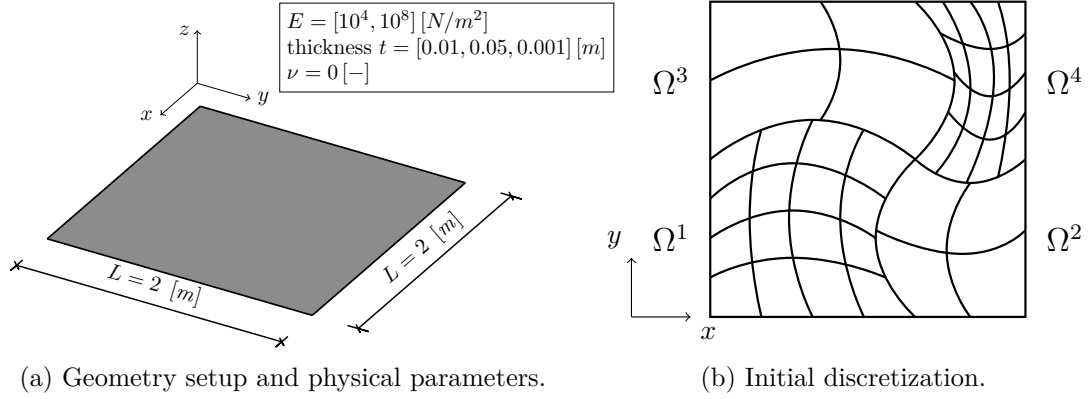


Figure 6.7 – Problem setup and initial non-conforming, multi-patch discretization for the curved four patches example.

that the exact deflection is smooth and it reads:

$$u_{\text{ex}} = \sin(\pi x) \cos(\pi y). \quad (6.49)$$

This setup is used to test the robustness of our method in the case of severe non-matching discretizations and with respect to the problem parameters. To this end, we present the convergence results for all combinations of Young's moduli $E = [10^4, 10^8] [Pa]$ and thickness of the plate $t = [0.05, 0.01, 0.005] [m]$, where we set the Poisson's ratio $\nu = 0 [-]$. From the results in Figure 6.8, we observe that the projection strategy shows robustness with respect to the input parameters and allows for an easy treatment of locking phenomena, where optimal convergence rates are attained starting from very coarse meshes. In Figure 6.9 the convergence behavior of the error measured in the H^2 norm with and without the imposition of the C^0 constraint at the cross-point is plotted. We observe that the loss of accuracy hinders the convergence for $p = 3, 4$, whereas the expected optimal rates of convergence are recovered in all cases when the linear constraint is imposed to the system. This is further highlighted in Figure 6.10, where the element-wise H^2 error is depicted for a discretization of degree $p = 4$, without and with the constraint, respectively. On one hand, we remark how the error is concentrated and much higher in the elements around the cross-point, spoiling the optimal convergence when the constraint is not imposed. On the other hand, with the linear constraint, we recover optimal convergence properties of the method. Finally, for this example we also analyze the performance of the nested preconditioner. In Table 6.1 we report the iterations needed by the external solver and in brackets the average number of intermediate iterations, for several degrees of the discretization $p = 2, 3$, and we compare it with a classical diagonally preconditioned conjugate gradient (PCG), a PCG where an incomplete LU (ILU) is used as preconditioner and a GMRES preconditioned with ILU. All the results refer to a global tolerance η_o of 10^{-10} and, for the nested SCR-FGMRES strategy, the intermediate and inner tolerances η_t and η_n are set to 10^{-6} . Further, the Schur complement is preconditioned by an approximation $\tilde{\mathbf{S}}_{\Gamma, \Gamma}$ obtained with a maximum

Chapter 6. Coupling of multi-patch Kirchhoff plates and Kirchhoff-Love shells

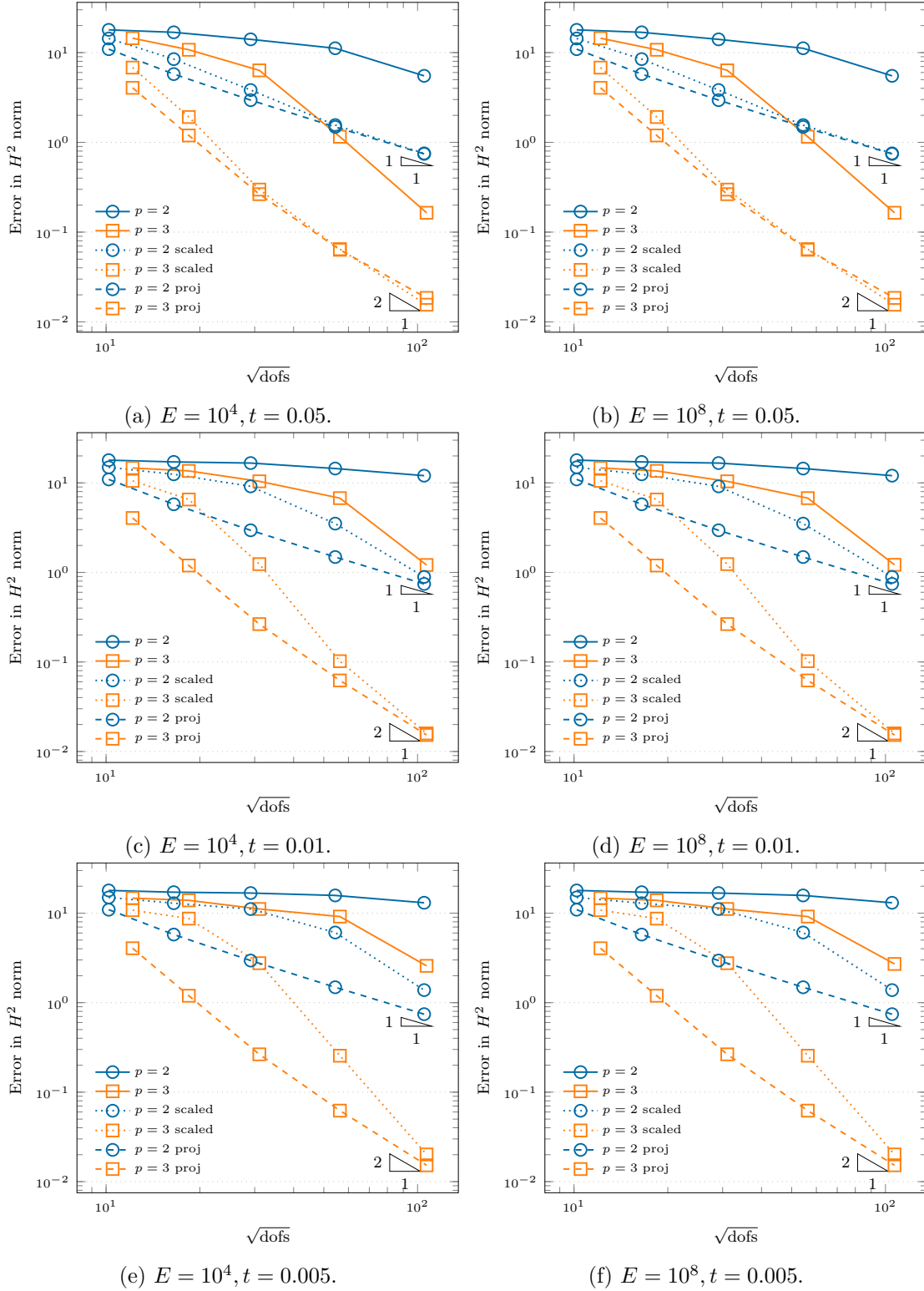


Figure 6.8 – Convergence of the H^2 norm of the error on the four patches with curved interface example, B-splines of degree $p = 2, 3$. Comparison of a classic penalty method, the scaled version with respect to the problem parameters proposed in [Herrema et al., 2019] (*scaled*) and our projection approach (*proj*).

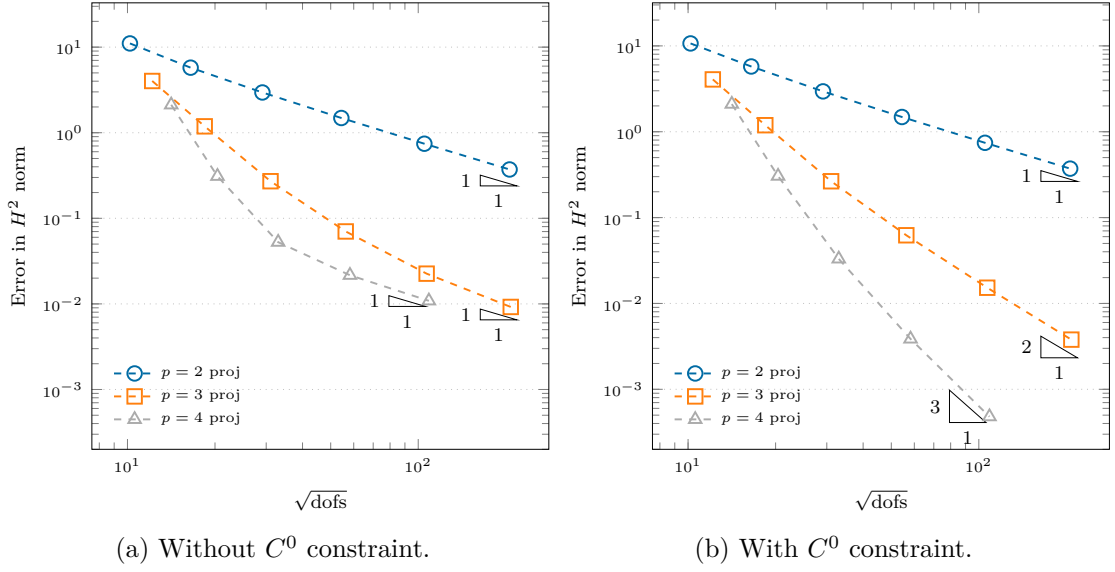


Figure 6.9 – Convergence study of the error in the H^2 norm in the non-matching case for the curved four patches example. Influence of imposing a C^0 constraint at the cross-point.

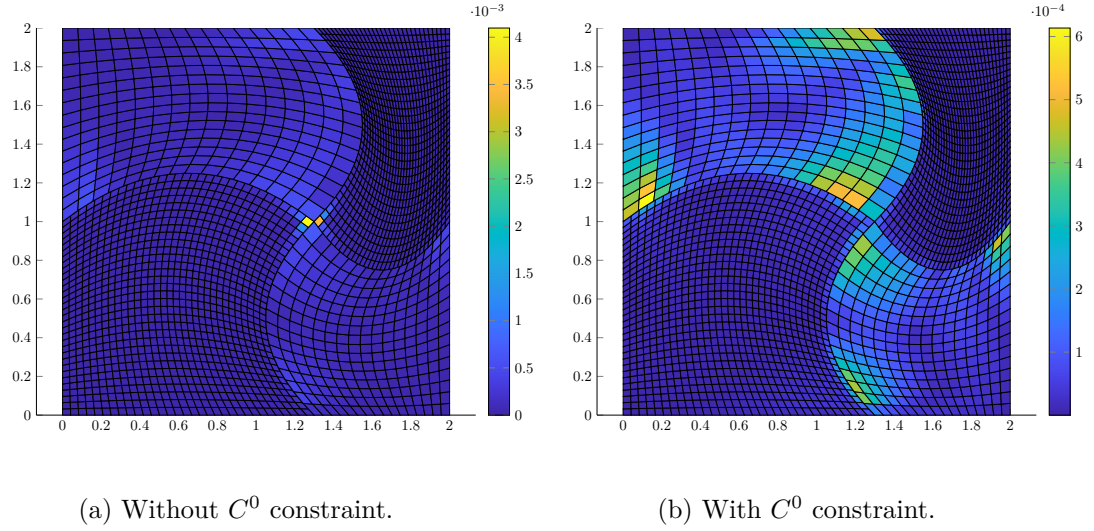


Figure 6.10 – Element-wise plot of the error in the H^2 norm in the non-matching case for the curved four patches example, B-splines of degree $p = 4$. Influence of imposing a C^0 constraint at the cross-point, notice the difference of one order of magnitude used in the two error colorbars.

Chapter 6. Coupling of multi-patch Kirchhoff plates and Kirchhoff-Love shells

of 6 iterations of GMRES. For the sake of completeness, we perform the same test with the choice of penalty parameters studied in [Herrema et al., 2019]. The results are summarized in Table 6.2, where we observe no substantial difference regarding the iterations needed to solve the system compared to the case where our choice of parameters is employed. This suggests that the proposed preconditioner is robust with respect to the penalty factors and it is also suitable to precondition systems stemming from other penalty approaches. In Table 6.3 we study the influence of the intermediate and inner

	256el.	1024el.	4096el.	16384el.
Condition number	$1.80 \cdot 10^7$	$2.03 \cdot 10^8$	$2.61 \cdot 10^9$	$3.82 \cdot 10^{10}$
Diagonally scaled PCG	792	953	--	--
PCG with ILU	111	980	--	--
GMRES with ILU	63	174	402	--
Nested SCR-FGMRES	3(21.6/3.3/20.3)	3(36/6/30.3)	4(51/17.5/40.7)	6(66.5/51/51.3)

(a) $p = 2$.

	256el.	1024el.	4096el.	16384el.
Condition number	$8.76 \cdot 10^7$	$1.77 \cdot 10^9$	$4.09 \cdot 10^{10}$	$1.60 \cdot 10^{12}$
Diagonally scaled PCG	921	--	--	--
PCG with ILU	53	221	--	--
GMRES with ILU	35	73	--	--
Nested SCR-FGMRES	3(26.6/4/24)	3(41.6/9/35)	4(58/25.5/45.7)	6(76.3/68.3/55.3)

(b) $p = 3$.

Table 6.1 – Condition number of \mathcal{A} and number of iterations needed by different iterative methods, $p = 2, 3$, as a function of the elements (el.). For the nested SCR-FGMRES, the numbers in brackets indicate the average number of intermediate iterations needed to solve Equations (6.32) to (6.34) in Algorithm 8, respectively. Setups marked with – did not reached convergence within the prescribed 1000 maximum number of iterations.

tolerances on the number of outer iterations required by the FGMRES solver, on a fixed mesh of 4096 elements, for B-splines of degree $p = 2, 3$. We note that as the chosen tolerances become smaller and smaller, we recover the algebraically exact SCR method, where in the limit the algorithm converges in one iteration. We also remark that finding an optimal choice for these parameters is, to the best of the authors' knowledge, still an open question in the community.

A non-trimmed nine-patches plate geometry

In this example we consider the computational domain $\Omega = [0, 3] \times [0, 3]$ depicted in Figure 6.11, divided into nine subdomains $\Omega^i, i = 1 \dots, 9$. Similarly to the previous example, all meshes are non-conforming at every coupling interface, where again an irrational factor of $\sqrt{2}/100$ has been used to shift the interface knots. Again, the body

6.3. Numerical examples

	256el.	1024el.	4096el.	16384el.
Condition number	$1.21 \cdot 10^9$	$3.82 \cdot 10^9$	$1.33 \cdot 10^{10}$	$5.01 \cdot 10^{10}$
Diagonally scaled PCG	992	--	--	--
PCG with ILU	175	--	--	--
GMRES with ILU	79	209	478	--
Nested SCR-FGMRES	3(22.3/4/21.6)	3(32/7.3/28.3)	4(44.5/20.2/34.7)	5(63.2/55/50.2)

(a) $p = 2$.

	256el.	1024el.	4096el.	16384el.
Condition number	$1.67 \cdot 10^9$	$4.79 \cdot 10^9$	$1.48 \cdot 10^{10}$	$5.25 \cdot 10^{10}$
Diagonally scaled PCG	778	--	--	--
PCG with ILU	697	--	--	--
GMRES with ILU	90	209	--	--
Nested SCR-FGMRES	3(28.6/5.3/27.3)	3(39.3/13/34.6)	5(55.8/29.4/41)	6(74/73.3/48.3)

(b) $p = 3$.

Table 6.2 – Condition number of \mathcal{A} and number of iterations needed by different iterative methods, $p = 2, 3$, as a function of the elements (el.) for the parameters proposed in [Herrema et al., 2019]. For the nested SCR-FGMRES, the numbers in brackets indicate the average number of intermediate iterations needed to solve Equations (6.32) to (6.34) in Algorithm 8, respectively. Setups marked with – did not reached convergence within the prescribed 1000 maximum number of iterations.

	$\eta_t = \eta_n = 10^{-4}$	$\eta_t = \eta_n = 10^{-5}$	$\eta_t = \eta_n = 10^{-6}$	$\eta_t = \eta_n = 10^{-8}$	$\eta_t = \eta_n = 10^{-10}$
$p = 2$	11	5	4	3	2
$p = 3$	13	7	4	3	2

Table 6.3 – Influence of the intermediate and inner tolerances η_t and η_n (where we always set $\eta_t = \eta_n$) on the number of outer iterations needed by the FGMRES solver, $p = 2, 3$, on a fixed mesh with 4096 elements.

load and boundary data are derived from the following analytical exact solution:

$$u_{\text{ex}} = \sin(\pi x) \cos(\pi y). \quad (6.50)$$

Further, we set the Young's modulus to $E = 10^6 [Pa]$, the thickness of the plate to $t = 0.01 [m]$ and the Poisson's ratio to $\nu = 0 [-]$. The convergence results of the error measured in the L^2 , H^1 and H^2 are presented in Figure 6.12, for splines of degree $p = 2, 3$. In this example we test the robustness of the method with respect to:

- floating patches;
- the presence of multiple cross-points where a constraint must be applied.

We again observe the expected asymptotic convergence rates of the error for all norms,

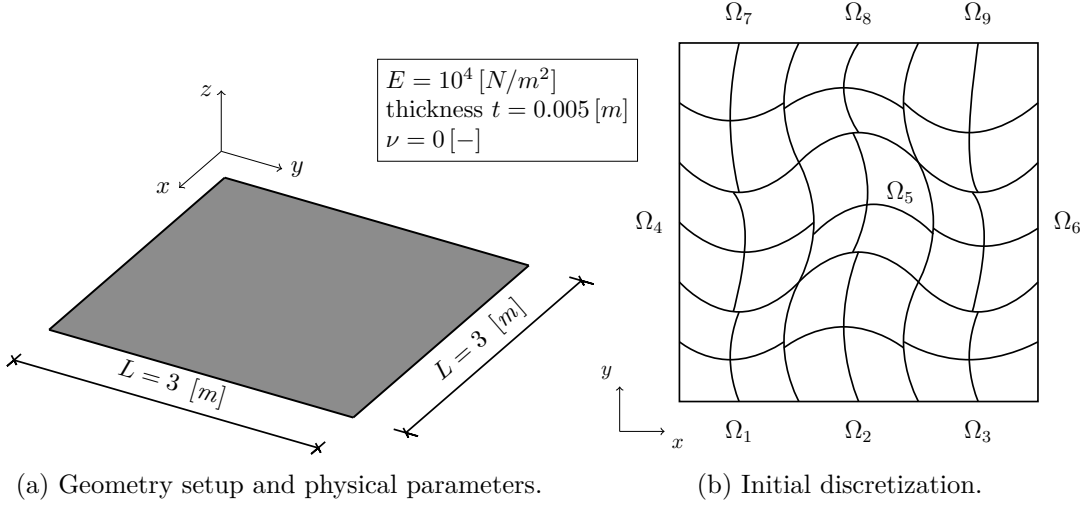


Figure 6.11 – Problem setup and initial non-conforming, multi-patch discretization for the nine patches example.

where we remark that the method behaves optimally, particularly for very coarse meshes, where locking phenomena are avoided. Indeed, on one hand, we again notice that a classical “vanilla” choice of the penalty parameters yield a severe overconstraint of the solution space, resulting in a loss of accuracy of several order of magnitudes compared to the projection method. On the other hand, the scaling studied in [Herrema et al., 2019] leads to better results especially in the energy norm. However, for coarse meshes, we note that the method still suffers from locking, thus hindering the accuracy achievable by B-splines.

A non-trimmed three-patches plate example

In this example we consider the computational domain $\Omega = [0, 2] \times [0, 2]$, split into three subdomains $\Omega^i, i = 1, 2, 3$, see Figure 6.13a. The initial non-conforming discretization used in the following is depicted in Figure 6.13b, where the interface knots are again shifted by a factor of $\sqrt{2}/100$ to induce the non-conformity. The peculiarity of this example is the presence of a geometrically non-conforming interface between the patches, which is further used to assess the robustness of our method. Similarly to the previous examples, the exact solution reads:

$$u_{\text{ex}} = \sin(\pi x) \cos(\pi y), \quad (6.51)$$

from which the applied body load and imposed boundary conditions are derived. Regarding the problem parameters, we set the Young’s modulus to $E = 10^6$ [Pa], the thickness of the plate to $t = 0.01$ [m] and the Poisson’s ratio to $\nu = 0$ [-]. The convergence results of the error measured in the L^2 , H^1 and H^2 are presented in Figure 6.14, for splines of

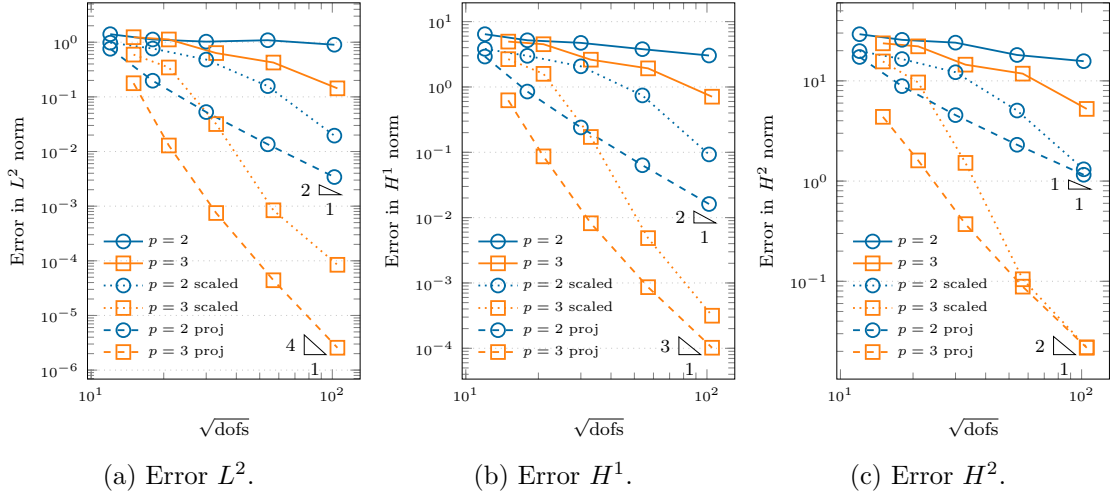


Figure 6.12 – Convergence of the error measured in the L^2 , H^1 and H^2 norms in the non-matching case for nine patches example, B-splines of degree $p = 2, 3$. Comparison of a classic penalty method, the scaled version with respect to the problem parameters proposed in [Herrema et al., 2019] (*scaled*) and our projection approach (*proj*).

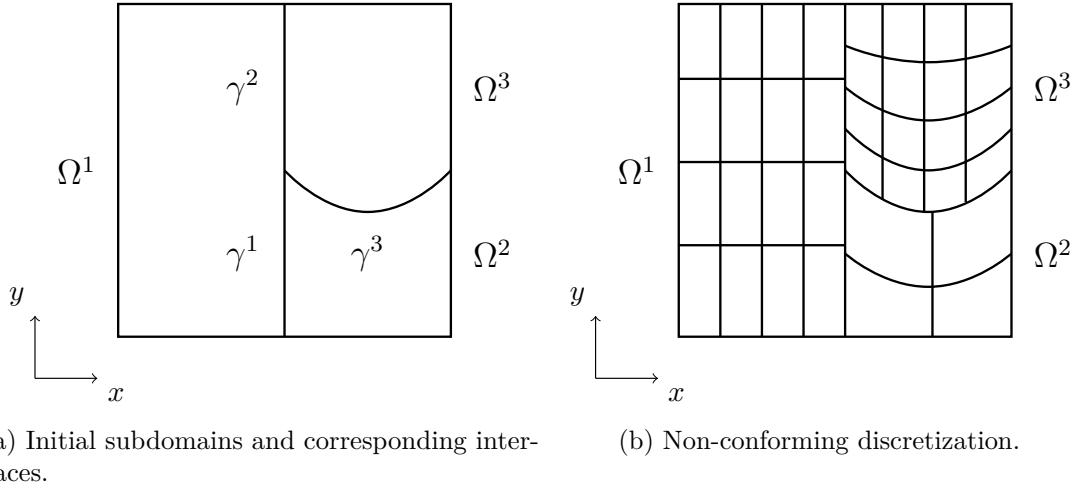


Figure 6.13 – Initial configuration and non-conforming discretization for the three patches example.

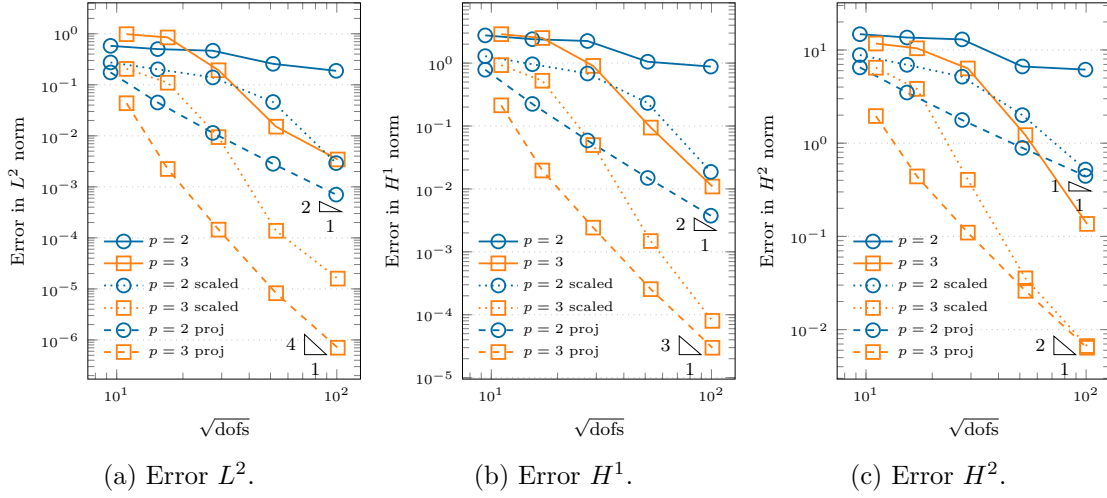


Figure 6.14 – Convergence study of the error measured in the L^2 , H^1 and H^2 norms in the non-matching case for the three patches example, B-splines of degree $p = 2, 3$. Comparison of a classic penalty method, the scaled version with respect to the problem parameters proposed in [Herrema et al., 2019] (*scaled*) and our projection approach (*proj*).

degree $p = 2, 3$. Analogously to our previous results, our method attains optimal rates of convergence, even in the presence of a geometrically non-conforming interface. Once again, this numerical experiment confirms that our method is insensitive to boundary locking, starting from very coarse discretizations, where a substantial gain in accuracy per degree-of-freedom is observed.

A flat L-bracket

The last example we present is meant to show the applicability of the method to more complex multi-patch geometries. Analogously to the example studied in [Benzaken et al., 2017], we modeled a flat L-bracket with 28 patches coupled along 34 interfaces, as depicted in Figure 6.15. We apply a constant line load of $100 [N/m]$ in the negative z -direction on the upper right edge and we impose clamped boundary conditions on the entire boundary of the upper left and lower left holes, respectively. Further, we set the Young's modulus to $E = 200 \cdot 10^9 [Pa]$, the thickness of the plate to $t = 0.01 [m]$ and the Poisson's ratio to $\nu = 0 [-]$. The solution field obtained with B-splines of degree $p = 2, 3$ is depicted in Figure 6.16, where we remark the smoothness of the obtained solution, especially across the coupling interfaces. In Figure 6.17 we also plot the bending stress tensor \mathbf{m} , where its components are defined as:

$$m_{ij} = D(\nu \delta_{ij} u_{kk} + (1 - \nu) u_{ij}) , \quad (6.52)$$

and where δ_{ij} denotes the standard Kronecker delta. We again obtain a smooth stress field, where no visible spurious oscillations are introduced by the proposed coupling strategy. Finally, in Figure 6.18, we plot the convergence results of the stress component m_{11} , evaluated at point A marked in Figure 6.15a, as a function of the number of dofs on a series of uniformly refined meshes. We note that for the classical penalty approach, and only for this example, we have tuned the penalty parameters to converge towards the reference value, where we have set $\alpha_{\text{def}}^\ell = 10^4 E$, $\alpha_{\text{rot}}^\ell = E$, $\ell = 1, \dots, L$. This example highlights once again the gain in accuracy achieved on coarse meshes by the proposed method, also for point-wise quantities of interest.

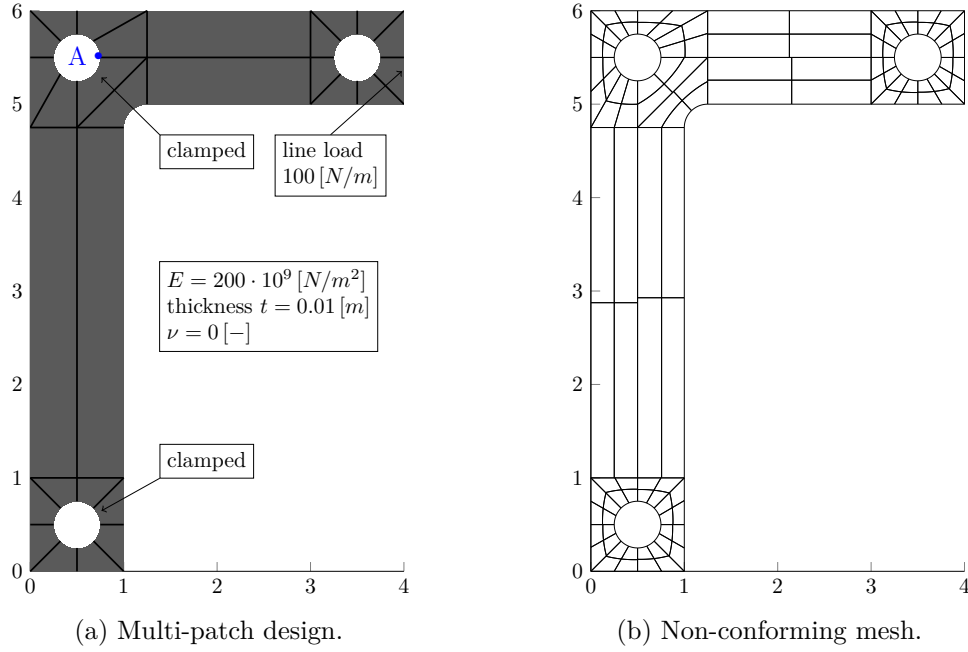


Figure 6.15 – Geometry setup and non-conforming discretization for the flat L-bracket example.

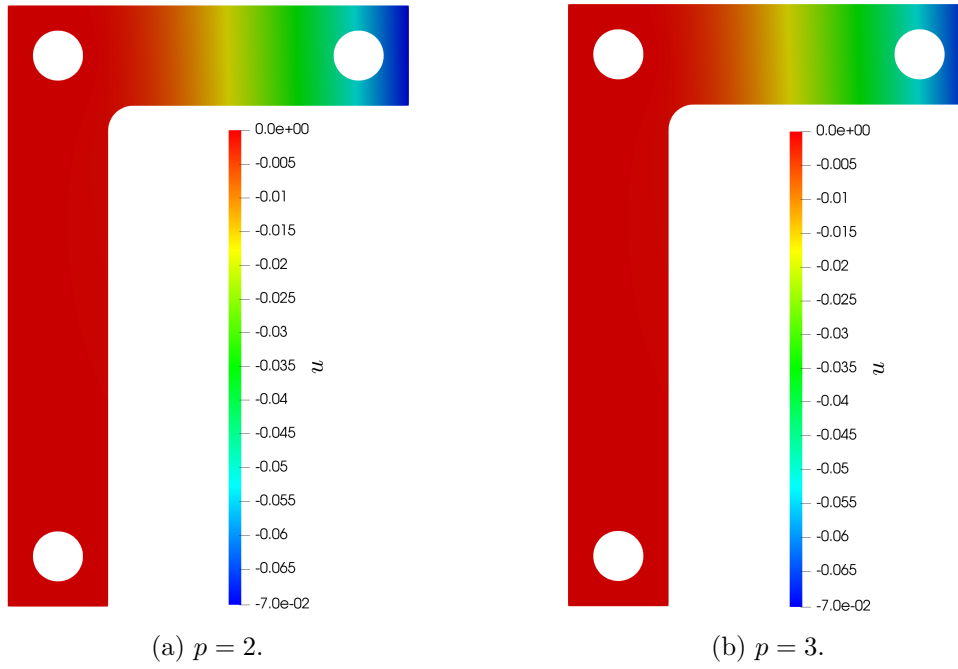


Figure 6.16 – Solution contour for the flat L-bracket example, B-splines of degree $p = 2, 3$.

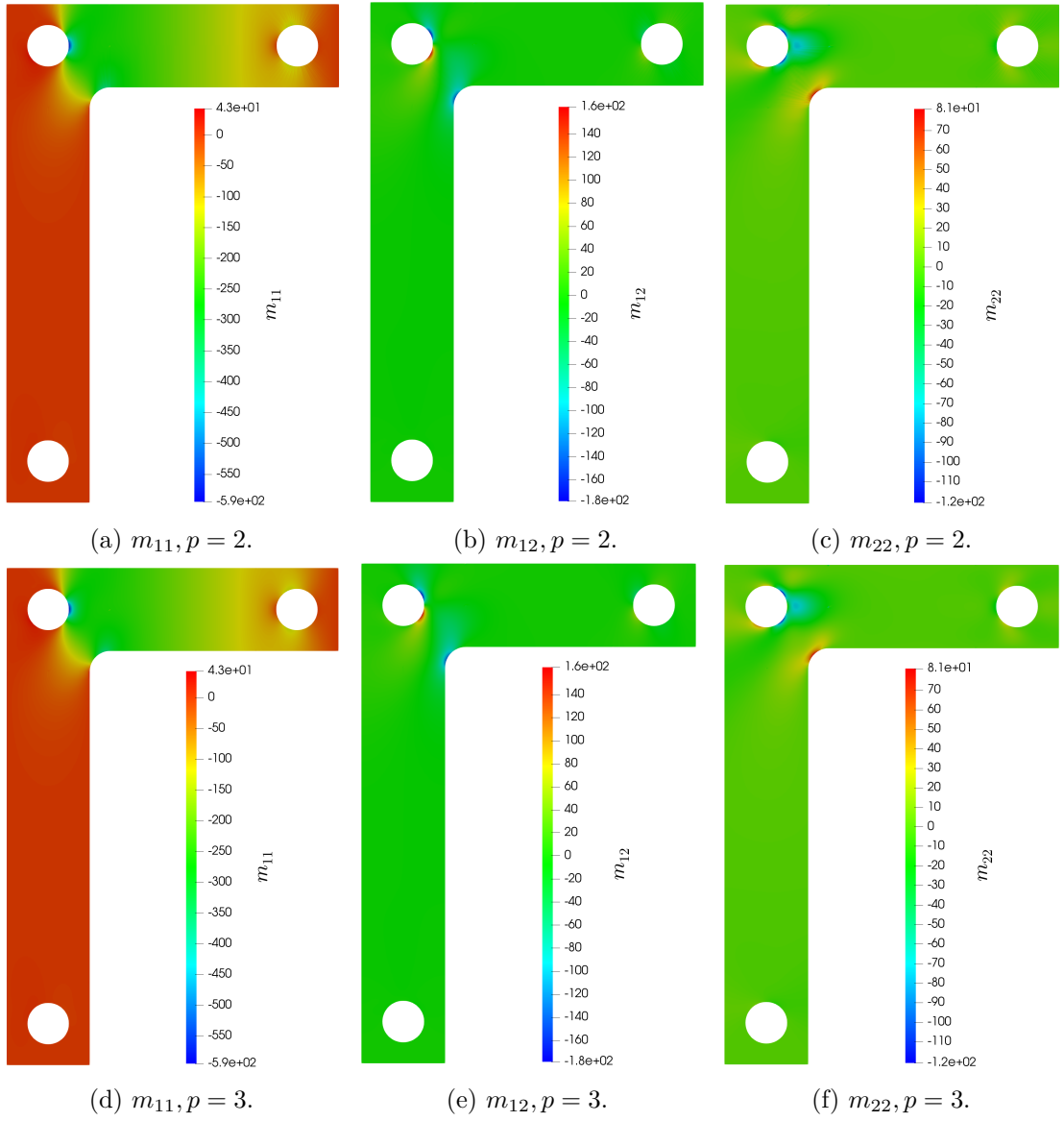


Figure 6.17 – Components of the bending stress tensor \mathbf{m} for the flat L-bracket example, B-splines of degree $p = 2, 3$.

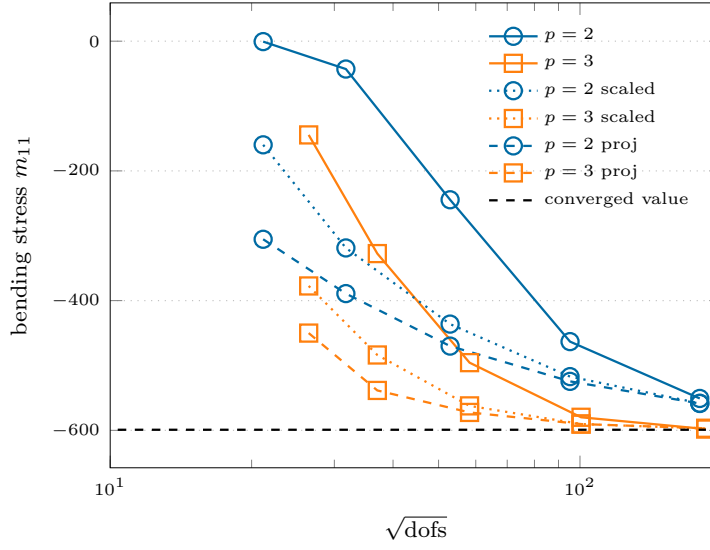


Figure 6.18 – Convergence study of the stress component m_{11} , evaluated at point A in Figure 6.13a, for the flat L-bracket example for different B-splines of degree $p = 2, 3$. Comparison of a classic penalty method, the scaled version with respect to the problem parameters proposed in [Herrema et al., 2019] (*scaled*) and our projection approach (*proj*).

6.3.2 Coupling of Kirchhoff-Love shells

Next, we verify the performance of the coupling strategy on a series of benchmarks in the scope of Kirchhoff-Love shells.

Four non-trimmed planar patches

The first example is meant to test and verify the implementation of our strategy in a non-trimmed planar setting. The geometrical setup is the domain $\Omega = [0, 2] \times [0, 2]$, subdivided into four non-conforming patches $\Omega^i, i = 1, \dots, 4$ coupled along curved interfaces, see Figure 6.19. To enforce the non-conformity of the latter, the initial interface knots have been shifted by the irrational factor $\sqrt{2}/100$. In our problem definition, we set the Young's modulus $E = 10^6 [Pa]$, the thickness of the plate $t = 0.005 [m]$ and the Poisson's ratio $\nu = 0.3 [-]$, respectively. Then, to verify the theoretical orders of convergence, we compute the approximation error in the L^2 and H^2 norms with respect to a manufactured smooth solution of the form:

$$\mathbf{u}_{\text{ex}}(x, y, z) = \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix} = \begin{pmatrix} \sin(\pi x) \sin(\pi y) \\ \sin(\pi x) \sin(\pi y) \\ \sin(\pi x) \sin(\pi y) \end{pmatrix}. \quad (6.53)$$

The results are summarized in Figure 6.20, where the convergence of the error measured in the L^2 and H^2 norms, respectively, is plotted against the square root of the number of dofs. We observe that our proposed method attains the expected order of convergence starting from very coarse meshes, whereas interface locking hinders the convergence rates of other penalty methods in the pre-asymptotic regime. As a consequence, we observe a substantial gain of accuracy per degree-of-freedom of the projection strategy, particularly in the L^2 norm. Additionally, we highlight the suboptimal convergence rates achieved by the method proposed in [Herrema et al., 2019], noticeable in the asymptotic regime for $p = 4$. Further, also for $p = 4$ and the L^2 norm, we observe the detrimental impact of our choice of penalty parameters on the conditioning of the stiffness matrix and, consequently, on the solution accuracy. This effect can be mitigated by reducing the exponent β in Equation (6.22), knowing that the method will converge sub-optimally, as depicted in Figure 6.21. For this reason, we will focus solely on moderate spline degrees $p = 2, 3$ in the following numerical experiments.

Multi-patch design of the Scordelis-Lo roof

In this example we assess the performance of our method on the well-known Scordelis-Lo roof. The geometrical setup, the chosen parameters and the initial non-conforming multi-patch design, where the roof is split into six subdomains $\Omega^i, i = 1, \dots, 6$, are summarized in Figure 6.22. The structure is supported at both ends of the cylindrical

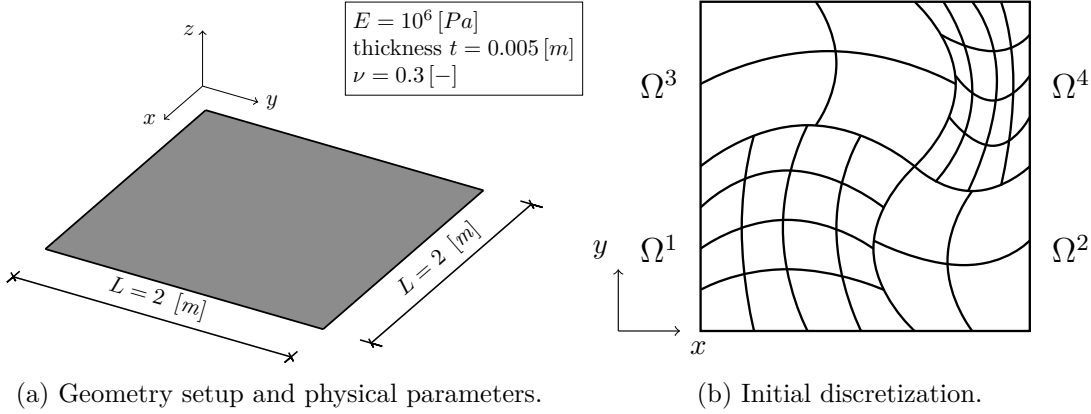


Figure 6.19 – Problem setup and initial non-trimmed, non-conforming, multi-patch discretization for the four planar patches example.

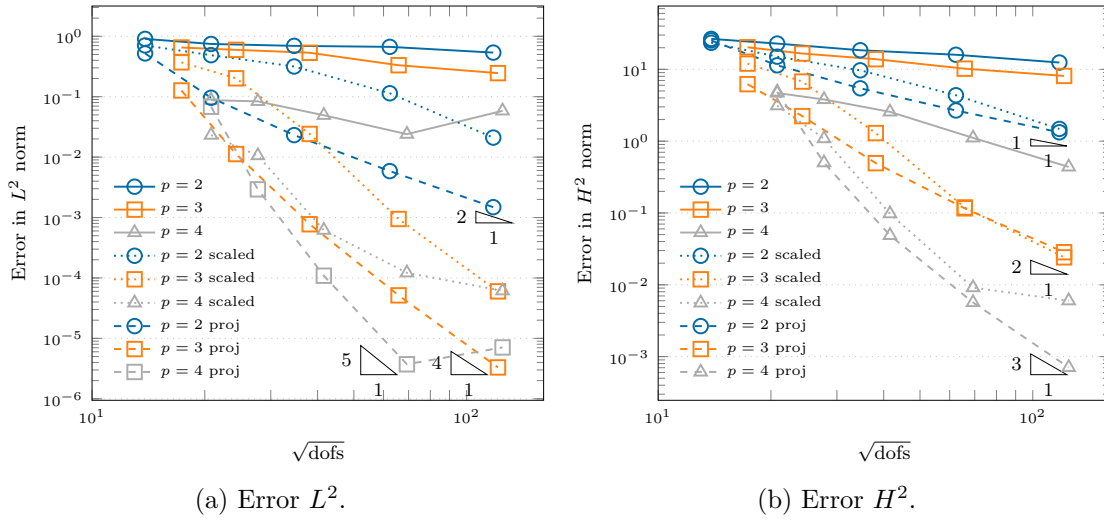


Figure 6.20 – Convergence of the error measured in the L^2 and H^2 norms in the non-trimmed, non-matching four patches example for different B-splines of degree $p = 2, 3, 4$. Comparison of a classic penalty method, the scaled version with respect to the problem parameters proposed in [Herrema et al., 2019] (*scaled*) and our projection approach (*proj*).

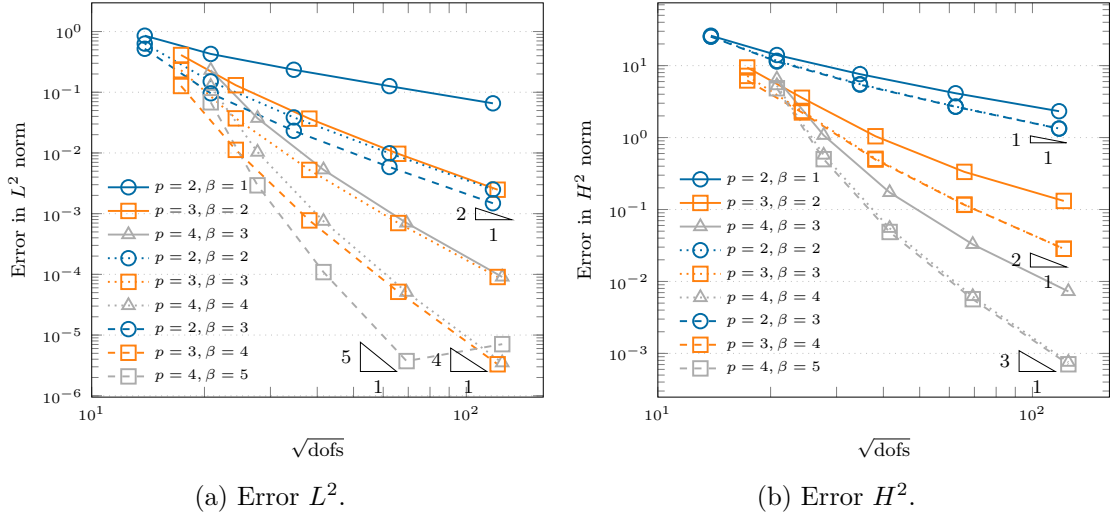


Figure 6.21 – Convergence of the error measured in the L^2 and H^2 norms for the projection method in the non-trimmed, non-matching four patches example for different B-splines of degree $p = 2, 3, 4$. Comparison of various scaling exponent β of the penalty parameters in Equation (6.22).

roof by so-called rigid diaphragms, which fix the displacement in the y - and z -directions, respectively. Moreover, the roof is subjected to a uniform gravity load, directed in the negative z -direction. As studied in [Herrema et al., 2019], we modify the original thickness of the benchmark problem. In particular, we set the Young’s modulus, the Poisson’s ratio and the thickness of the structure to $4.32 \cdot 10^8 [Pa]$, $0.0 [-]$ and $0.025 [m]$, respectively. As typically done for this problem, we study the convergence of the displacement in the z -direction at the center of the free edge, where the reference value $u_z^{\text{ref}} = -32.01045$ is used for normalization. The results are presented in Figure 6.23b for different penalty methods and also for the single-patch case. We observe that our approach, the method presented in [Herrema et al., 2019] and the single patch case show a similar convergence behavior. However, in case when the penalty parameter is only scaled by the Young’s modulus, interface locking phenomena arise, and they are particularly severe for quadratic B-splines. Moreover, in Figure 6.23a, we compare the time needed to compute and assemble the penalty terms for the aforementioned approaches, where the projection method shows its computational efficiency. This is linked to the fact that, although the projection algorithm requires the solution of an additional system, the corresponding coupling terms involve significantly fewer dofs compared to standard penalty-like methods.

Remark 6.11 *Although the Scordelis-Lo roof is a classical benchmark for shell analysis, it only provides a reference value for the displacement in a point. Therefore, it is not suited to quantify the order of convergence of a method, but it only serves as verification of the latter.*

Chapter 6. Coupling of multi-patch Kirchhoff plates and Kirchhoff-Love shells

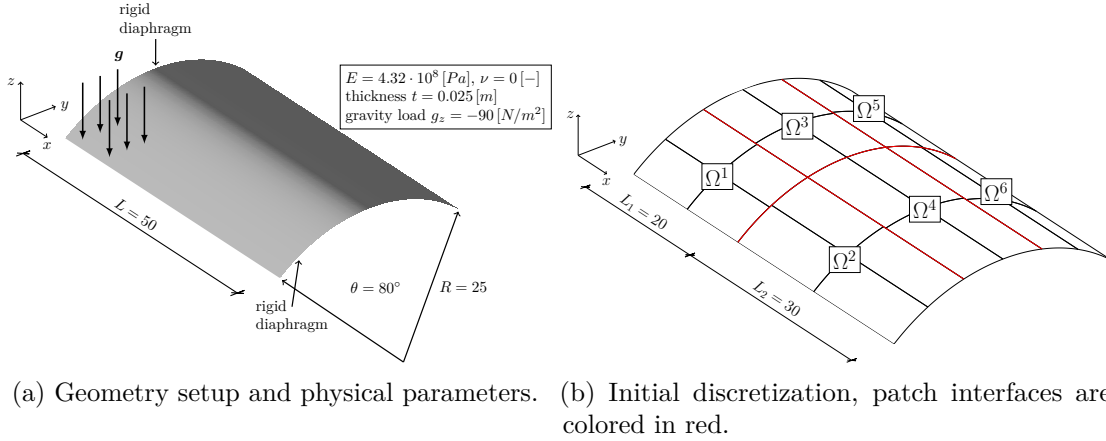


Figure 6.22 – Problem setup for the Scordelis-Lo roof example.

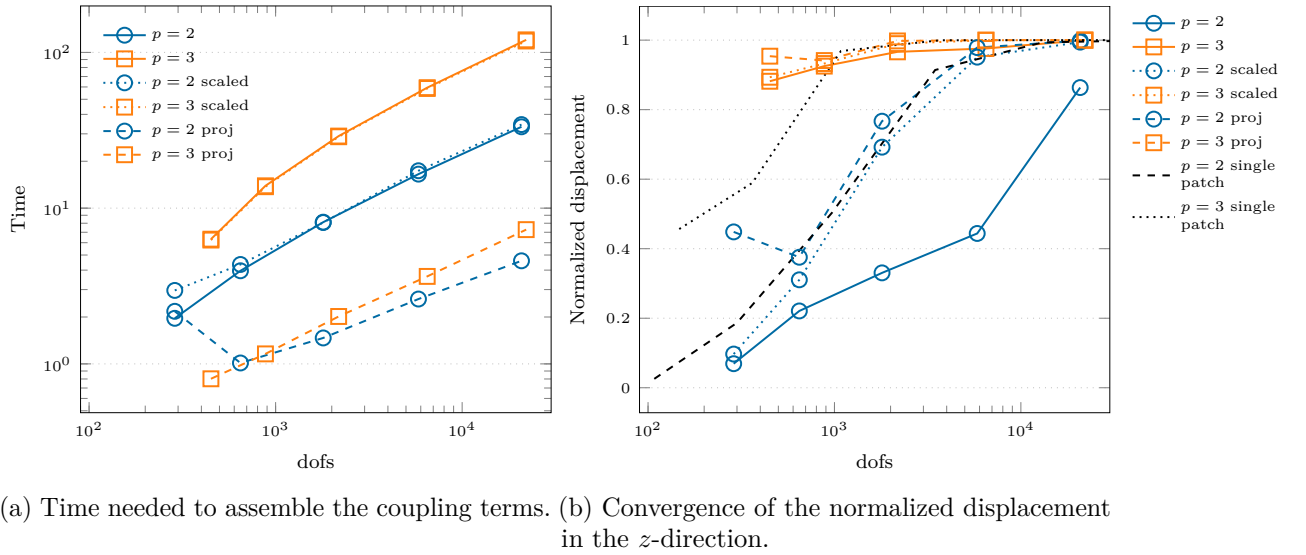


Figure 6.23 – Convergence of the normalized displacement in the z -direction at the middle of the free edge and time needed to assemble the penalty contributions, Scordelis-Lo roof example.

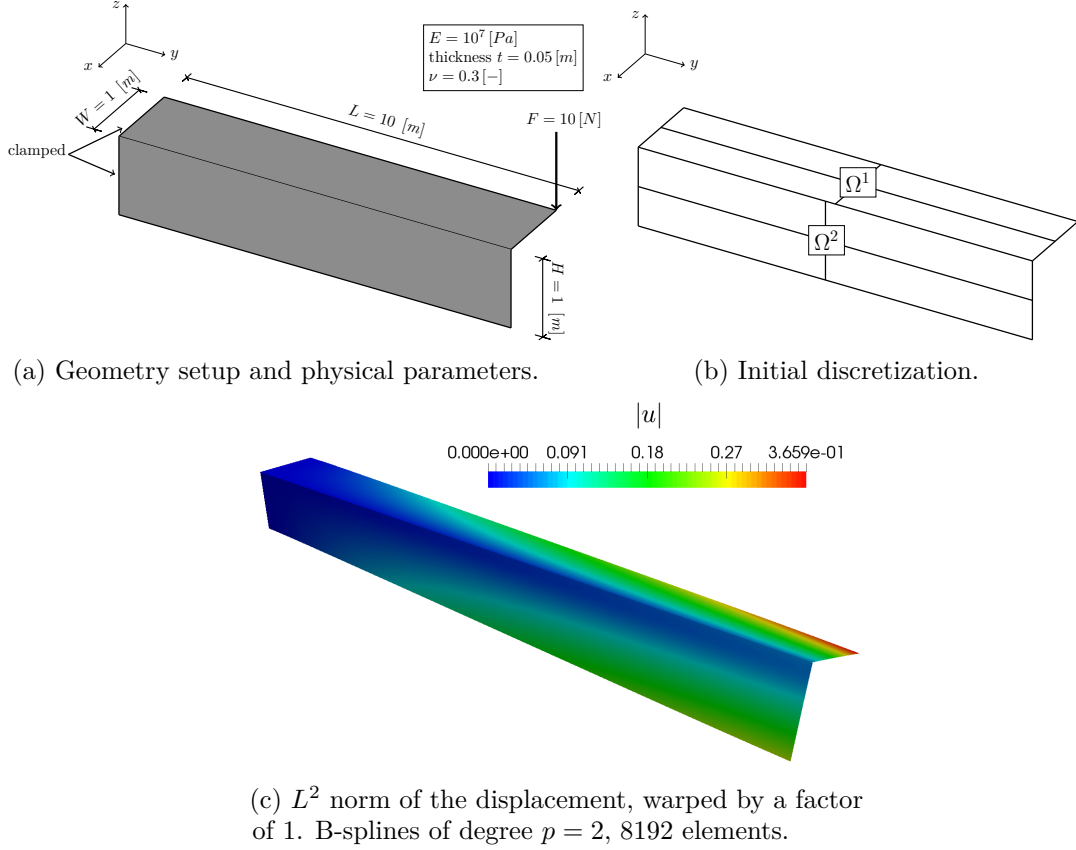


Figure 6.24 – Problem setup for the L-beam example and L^2 norm of the displacement field obtained with the proposed coupling strategy.

L-beam

This example is meant to demonstrate the applicability of our approach to couple patches at an arbitrary angle, where the corresponding rotational constraint keeps the angle fix during deformation. We consider a beam with an L-section discretized by two non-conforming patches $\Omega^i, i = 1, 2$, as depicted in Figure 6.24. The beam is clamped on one side and it is subjected to a point load of 10 [N], directed in the negative z -direction. Further, we set the Young's modulus, the Poisson's ratio and the thickness of the structure to 10^7 [Pa], 0.3 [-] and 0.05 [m], respectively. To check the correct imposition of the rotational constraint, we compute the angle formed by the two patches at the free corner on a series of uniformly refined meshes. The corresponding results are presented in Figure 6.25a. We remark that on coarse meshes and for the projection method, the rotational constraint is imposed in a less “rigid” way compared to other penalty approaches. This allows to mitigate the effects related to interface locking starting from coarse meshes. Similarly to the previous example, we observe a faster convergence behavior of the vertical displacement under the point load when our approach is employed, see Figure 6.25b, especially compared to a classical penalty method.

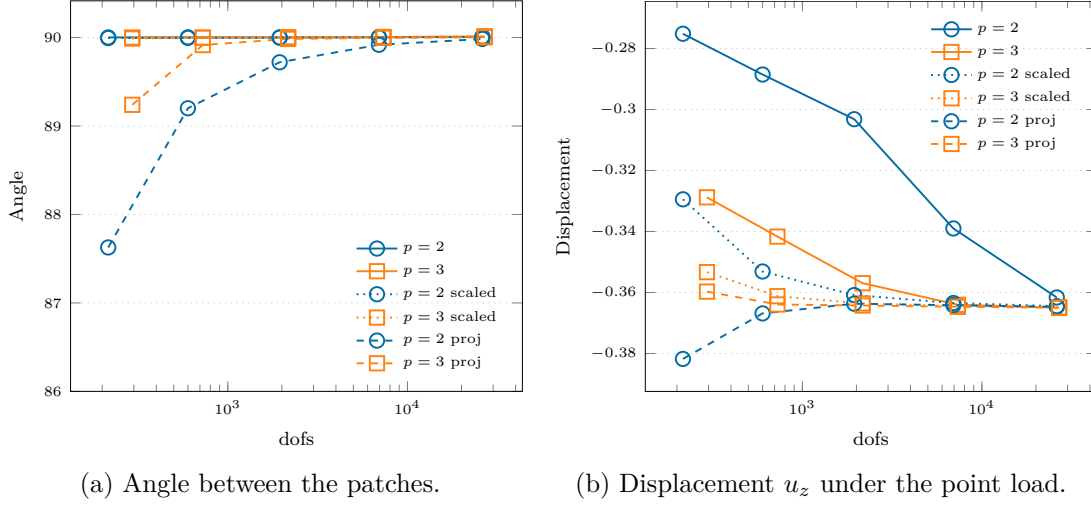


Figure 6.25 – Convergence study of the angle between the patches and the displacement u_z under the point load in the L-beam example, B-splines of degree $p = 2, 3$. Comparison of a classic penalty method, the scaled version with respect to the problem parameters proposed in [Herrema et al., 2019] (*scaled*) and our projection approach (*proj*).

Pure bending of three trimmed planar patches

In this example we consider the computational domain $\Omega = [0, 2] \times [0, 1]$ split into three trimmed subdomains $\Omega^i, i = 1, 2, 3$ as depicted in Figure 6.26. We remark that in this particular setup, the middle patch is coupled on both sides along trimming interfaces, defined by quadratic spline curves. The applied boundary conditions and loading function are again derived from a smooth solution of the form:

$$\mathbf{u}_{\text{ex}}(x, y, z) = \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \sin(\pi x) \sin(\pi y) \end{pmatrix}. \quad (6.54)$$

Then, we fix the Young's modulus and the Poisson's ratio of the structure to $10^6 [Pa]$ and $0.3 [-]$, respectively. This example confirms the severity of locking interface phenomena, especially as the shell gets progressively slender. Indeed, we vary the thickness in the range $[0.5, 0.05, 0.01] [m]$, where the results are reported in Figure 6.27. Moreover, we observe that in the trimmed case these detrimental effects are even more pronounced, since more basis functions are involved in the imposition of the constraints compared to the non-trimmed case.

Remark 6.12 *In the trimmed case, at any point of a coupling interface γ^ℓ and for each neighboring patch, we have $(p+1) \times (p+1)$ shape functions providing a non-zero contribution to the penalty matrices. This is in contrast to the non-trimmed case, where at any point of γ^ℓ and for each neighboring patch we have at most $(p+1)$, respectively, $2(p+1)$ B-splines involved in the computation of the displacement and rotational coupling*

\mathbf{P}_{ij}	11	12	13	21	22	23	31	32	33
x	0	1/3	0	1/2	1/2	1/2	1	2/3	1
y	0	1/2	1	1/3	1/2	2/3	0	1/2	1
z	0	0	0	0	0	0	0	0	0

Table 6.4 – Coordinates of the control points \mathbf{P}_{ij} , $i, j = 1, \dots, 3$ associated with the astroid domain.

terms.

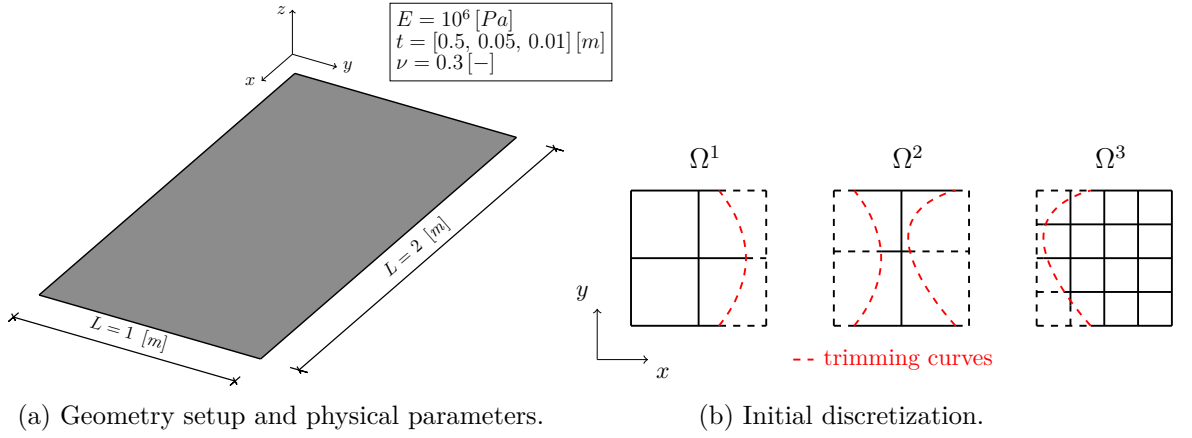


Figure 6.26 – Problem setup for the trimmed three planar patches example. For the sake of visualization, the patches have been separated.

Trimmed astroid

This example is adapted from the shell obstacle course presented in [Benzaken et al., 2021]. We consider the computational domain Ω split into three trimmed subdomains Ω^i , $i = 1, 2, 3$ as depicted in Figure 6.28. In the same figure, the two trimmed interfaces γ^ℓ , $\ell = 1, 2$ are defined as quadratic B-spline curves. The domain Ω is characterized by the control points \mathbf{P}_{ij} , $i, j = 1, \dots, 3$ as summarized in Table 6.4, where the indices i, j are ordered as the parametric coordinates ξ, η represented in Figure 6.28. The load function and boundary data are computed from the following manufactured solution:

$$\mathbf{u}_{\text{ex}}(\xi, \eta) = \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix} = \begin{pmatrix} (\frac{1}{2} - \eta)\xi^2(\xi - 1)^2\eta(1 - \eta) \\ (\xi - \frac{1}{2})\eta^2(\eta - 1)^2\xi(1 - \xi) \\ \xi(1 - \xi)\sin(\pi\xi)\sin(\pi\eta) \end{pmatrix}, \quad (6.55)$$

where we impose inhomogeneous Dirichlet and Neumann type boundary conditions on the displacements and on the bending moments, respectively, on the entire boundary $\partial\Omega$. Even though this problem is defined on a planar geometry, meaning that bending and membrane responses are decoupled, its investigation is still worthwhile since the

solution \mathbf{u} is defined as a function of the parametrization. This drastically complicates the derivation of the exact quantities and their stable computation. The convergence results for the error in the L^2 and energy norms for several values of the thickness $t = [0.1, 0.01, 0.005] [m]$ are depicted in Figure 6.29. This example confirms that our projection method mitigates the detrimental effects linked to interface locking, yielding a significant gain of accuracy per-degrees-of-freedom. This is particularly noticeable as the thickness of the structure becomes smaller, where, for other penalty techniques, locking phenomena hinder the optimal convergence in the pre-asymptotic regime.

Trimmed cylinder

This example is again adapted from the shell obstacle course presented in [Benzaken et al., 2021]. We consider the computational domain Ω split into four trimmed subdomains $\Omega^i, i = 1, \dots, 4$ as depicted in Figure 6.30. The corresponding trimmed interfaces $\gamma^\ell, \ell = 1, \dots, 4$ are defined as quadratic B-spline curves. This numerical experiment tests the applicability of the proposed methodology to the coupling of trimmed multi-patch surfaces in the presence of cross-points. Similarly to previous examples, the initial internal knots of patches Ω^2 and Ω^3 have been shifted by a factor $\sqrt{2}/100$ to achieve non-conforming discretization at the corresponding trimmed interfaces. Then, we set the Young's modulus, the Poisson's ratio and the thickness of the cylinder to $10^7 [Pa]$, $0.3 [-]$ and $0.001 [m]$, respectively. The load function and boundary data are computed from the following manufactured solution:

$$\mathbf{u}_{\text{ex}}(\xi, \eta) = -(\xi - 1)^2 \xi^2 \eta(\eta - 1) \mathbf{a}_3, \quad (6.56)$$

where \mathbf{a}_3 denotes the covariant vector in the thickness direction. The convergence results for the error measured in the L^2 and energy norms, respectively, are depicted in Figure 6.31. Similarly to our previous findings, we observe a faster convergence behavior of the projection method in the pre-asymptotic regime, where interface locking is avoided on very coarse meshes. This results in a substantial gain of accuracy per-degree-of-freedom, which is particularly noticeable for quadratic B-splines.

The DTU 10 MW Reference wind turbine blade

In our last example, we perform an isogeometric shell analysis of the DTU 10 MW Reference wind turbine blade [Bak et al., 2013], whose design was inspired by the NREL 5 MW reference wind turbine [Jonkman et al., 2009]. The blade is modeled by 20 non-conforming cubic spline surfaces. As noted in [Herrema et al., 2019], a multi-patch design allows to accurately resolve material discontinuities along the patch interfaces. The outer shell of the blade and the internal shear webs are depicted in Figure 6.32. In the same figure, colored regions are used to define the corresponding composite layup, where each region has a different multi-directional ply stacking sequence and a varying

thickness distribution along the spanwise direction. We summarize the most relevant mechanical properties in Table 6.5. Moreover, in Figure 6.33, we show the composite layup of the leading panels through the thickness as a function of the spanwise coordinate. For further details on material properties and thickness profiles we refer to [Bak et al., 2013]. For the analysis, we consider the response of the blade under gravity load, where

Multi-directional ply	Uniax	Biax	Triax	Balsa	
Young's modulus E_1	41.63	13.92	21.79	0.050	[GPa]
Young's modulus E_2	14.93	13.92	14.67	0.050	[GPa]
Shear modulus G_{12}	5.047	11.50	9.413	0.01667	[GPa]
Poisson's ratio ν_{12}	0.241	0.533	0.478	0.5	[—]
Shear modulus $G_{13} = G_{23}$	5.04698	4.53864	4.53864	0.150	[GPa]
Mass density ρ	1915.5	1845.0	1845.0	110	[kg/m ³]

Table 6.5 – Mechanical properties of the multi-directional plies.

the blade is modeled as clamped on the rotor side. The L^2 norm of the displacement field and the corresponding deflection of the blade are depicted in Figure 6.34. The results are obtained by employing a discretization of quadratic B-splines defined on 89528 elements. Note that we directly import the geometry used in [Bak et al., 2013] for the structural analysis.

Remark 6.13 *The latter is true for every patch except for webs A, B and C, which are obtained by linear extrusion of a generating spline, meaning that one linear element suffices to exactly describe the surface along the corresponding parametric direction. Therefore, h -refinement is performed along the latter direction by introducing 50 equidistributed knots.*

We remark that all the results on the blade have been obtained by setting the scaling factor $\beta = p$ in the penalty terms to limit the impact of the latter on the condition number of the stiffness matrix.

Simplified topology optimization of webs A and B

This example is meant to show the applicability of the proposed methodology to trimmed geometries obtained by a simplified topology optimization. Note that this numerical test is just a showcase of the flexibility of our computational framework and a realistic topology optimization of the webs is beyond the scope of this work. Furthermore, the geometric operation described in this section are based on an heuristic engineering approach. We trimmed away from the original geometry two holes, one close to the center of the structure and another at the end the web. This design is obtained by adapting the optimized solution presented in [Albanesi et al., 2020], where the final geometry is depicted in Figure 6.35. This design results in a reduction of $\approx 20.6\%$ of the original total

mass of web A. Similarly, we perform the same operations on web B. The L^2 norm of the displacement field and the corresponding deflection of the blade with trimmed webs are depicted in Figure 6.36, where we observe a reduction of $\approx 2.0\%$ in tip displacement related to the loss of total mass of the structure. The results are obtained with quadratic B-splines defined on a total of 84250 active elements.

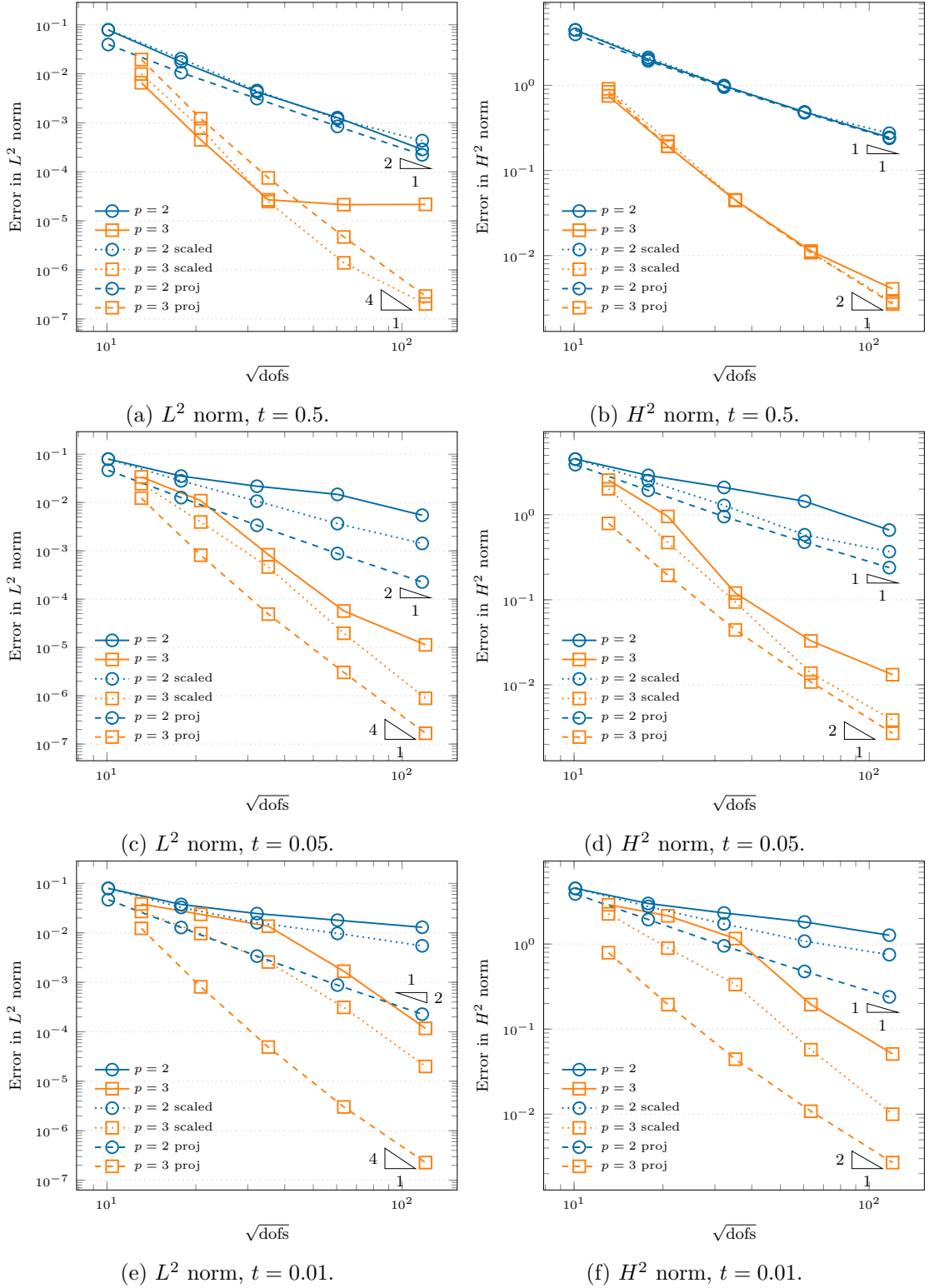
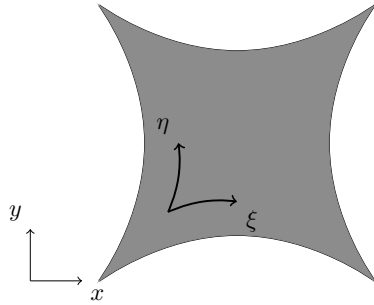
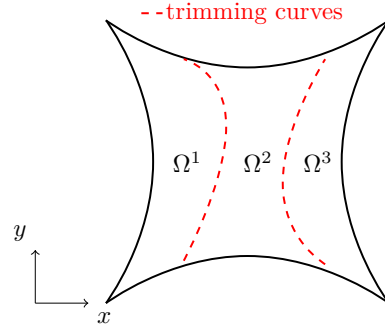


Figure 6.27 – Convergence study of the error measured in the H^2 and L^2 norms in the trimmed three patches example, B-splines of degree $p = 2, 3$, thickness $t = [0.5, 0.05, 0.01]$ [m]. Comparison of a classic penalty method, the scaled version with respect to the problem parameters proposed in [Herrema et al., 2019] (*scaled*) and our projection approach (*proj*).

$E = 10^7 [N/m^2]$
 thickness $t = [0.1, 0.01, 0.005] [m]$
 $\nu = 0.3 [-]$



(a) Geometry setup and physical parameters.



(b) Trimmed multi-patch geometry.

Figure 6.28 – Problem setup and initial multi-patch geometry for the trimmed astroid example.

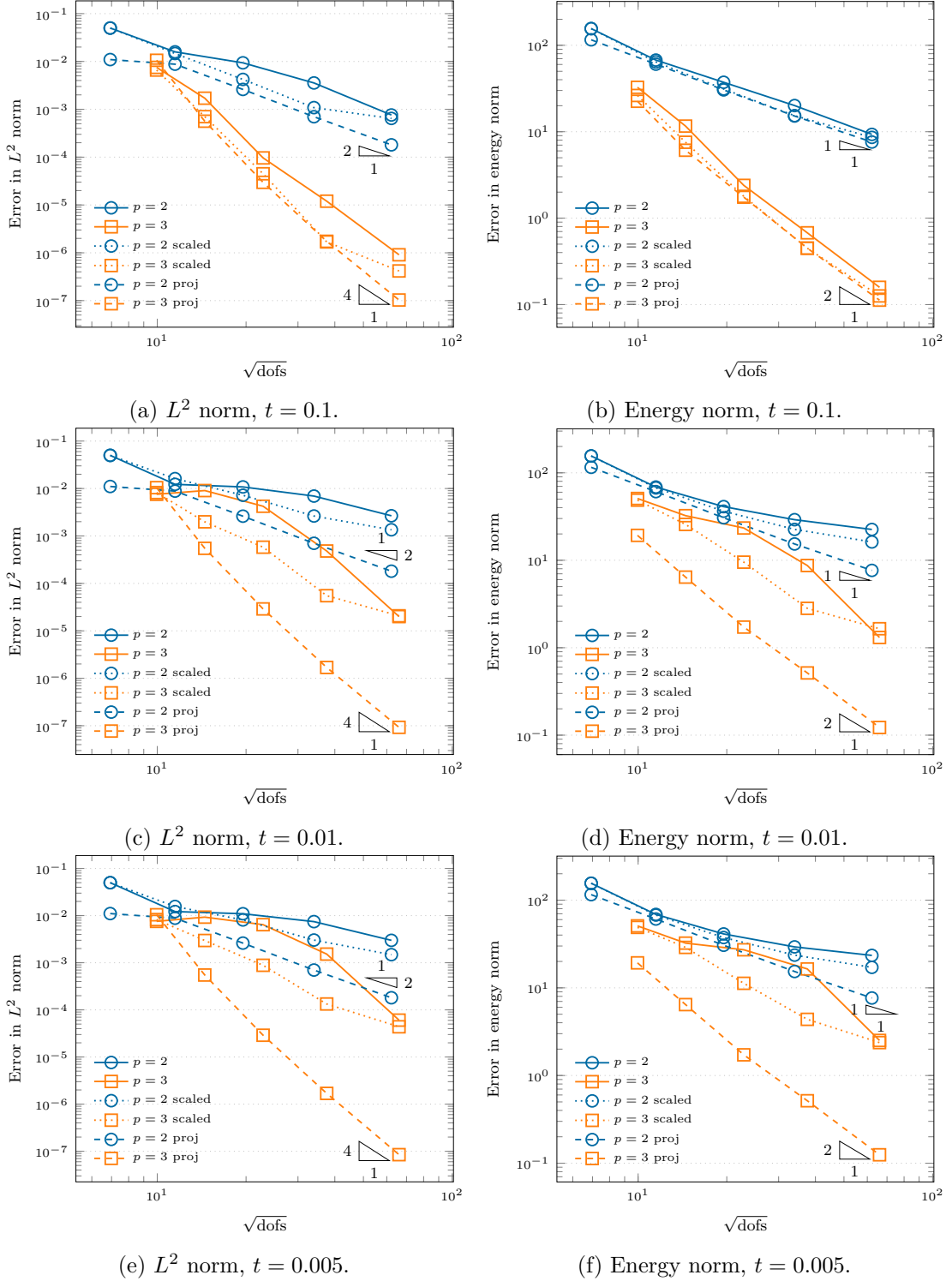
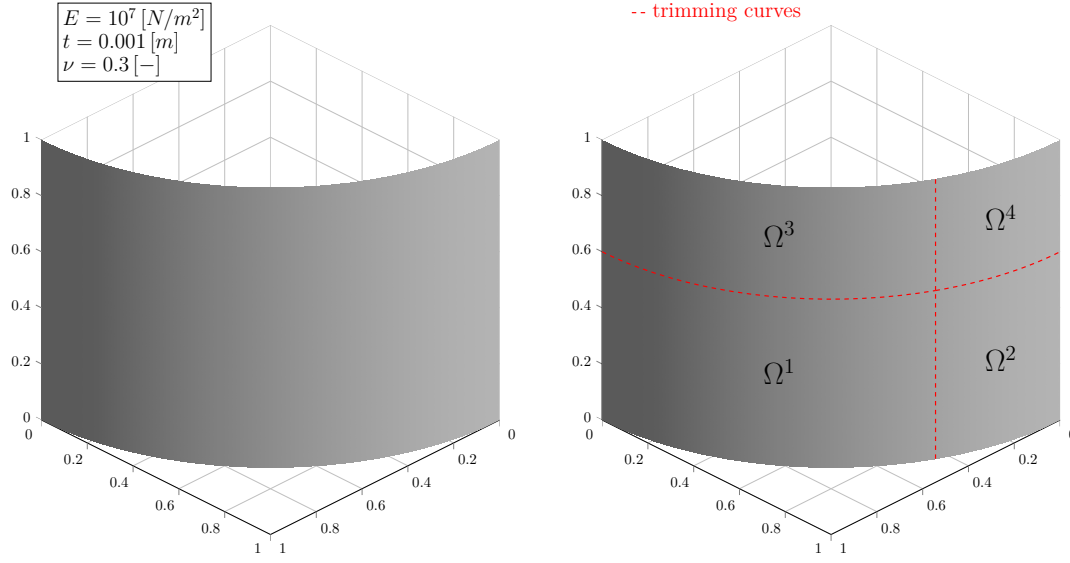


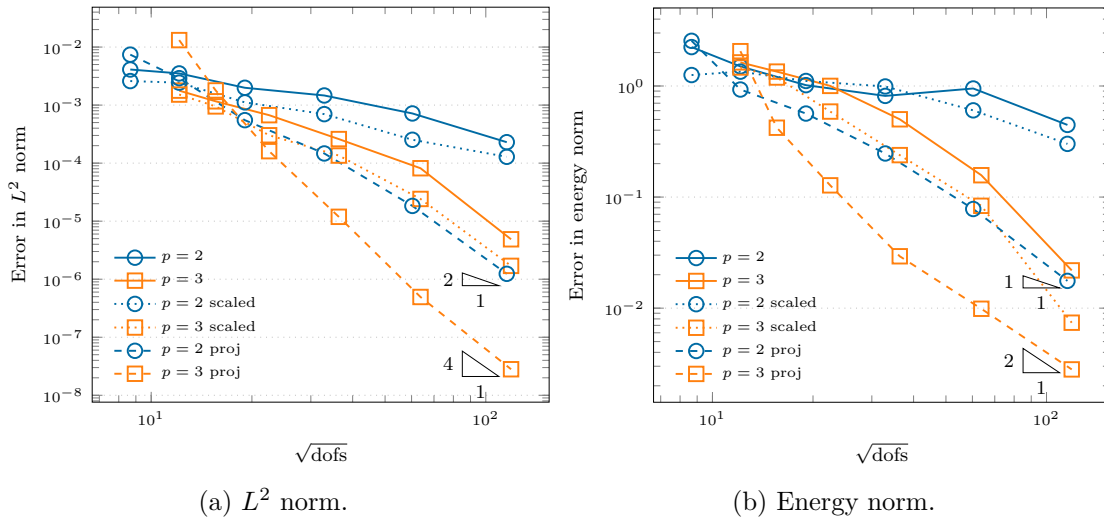
Figure 6.29 – Convergence study of the error measured in the energy and L^2 norms in the trimmed astroid example, B-splines of degree $p = 2, 3$, thickness $t = [0.1, 0.01, 0.005]$ [m]. Comparison of a classic penalty method, the scaled version with respect to the problem parameters proposed in [Herrema et al., 2019] (*scaled*) and our projection approach (*proj*).



(a) Geometry setup and physical parameters.

(b) Trimmed multi-patch geometry.

Figure 6.30 – Problem setup and initial multi-patch geometry for the trimmed cylinder example.



(a) L^2 norm.

(b) Energy norm.

Figure 6.31 – Convergence study of the error measured in the L^2 and energy norms for the trimmed cylinder example, B-splines of degree $p = 2, 3$. Comparison of a classic penalty method, the scaled version with respect to the problem parameters proposed in [Herrema et al., 2019] (*scaled*) and our projection approach (*proj*).

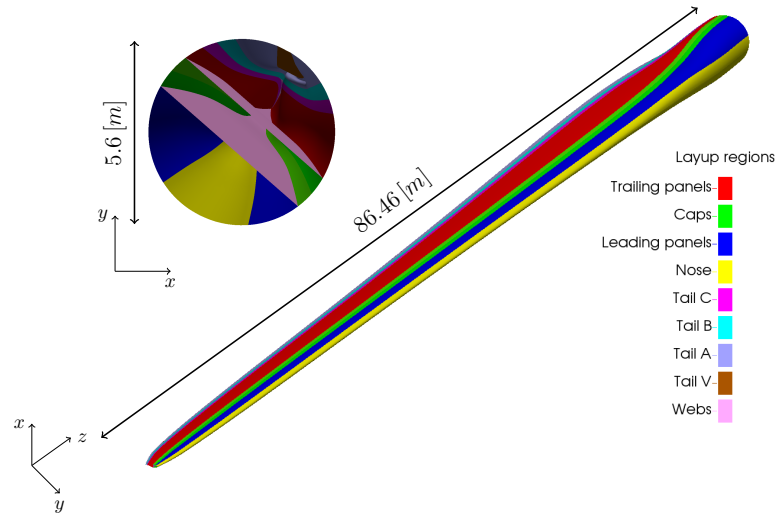


Figure 6.32 – DTU 10 MW blade geometry and depiction of the circumferential regions used for the composite materials definition.

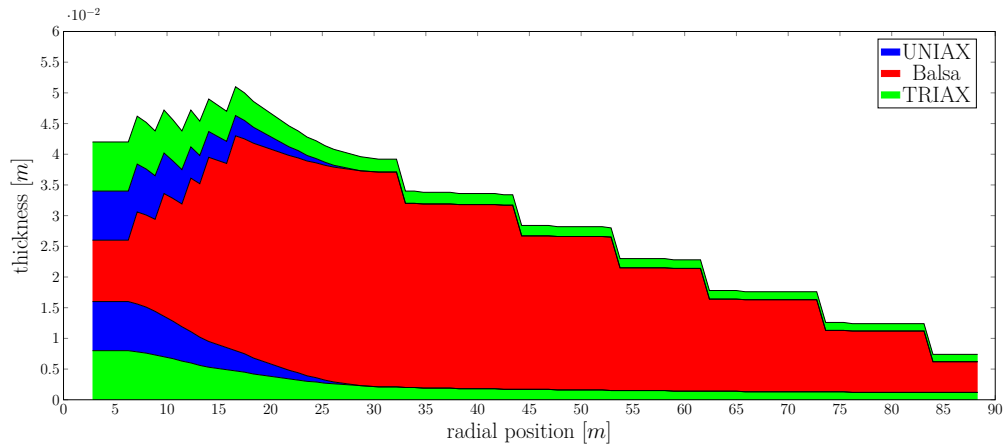


Figure 6.33 – Composite layup along the radial direction of the leading panels.

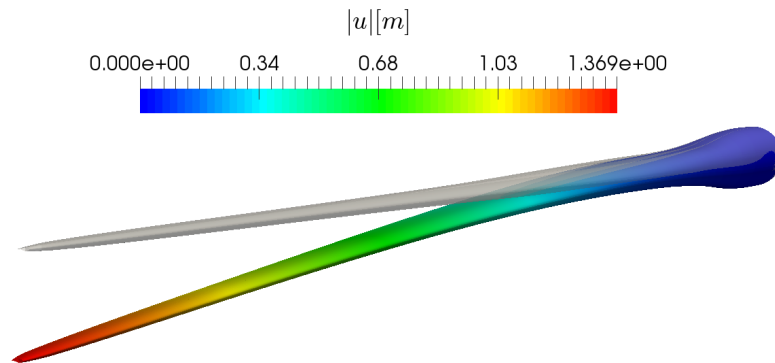


Figure 6.34 – L^2 norm of the displacement field on the DTU 10 MW blade subject to gravitational load, B-splines of degree $p = 2$, 89528 elements, results warped by a factor 5.



Figure 6.35 – Design of web A after simplified topology optimization. Zoom on the detail at the end of the blade.

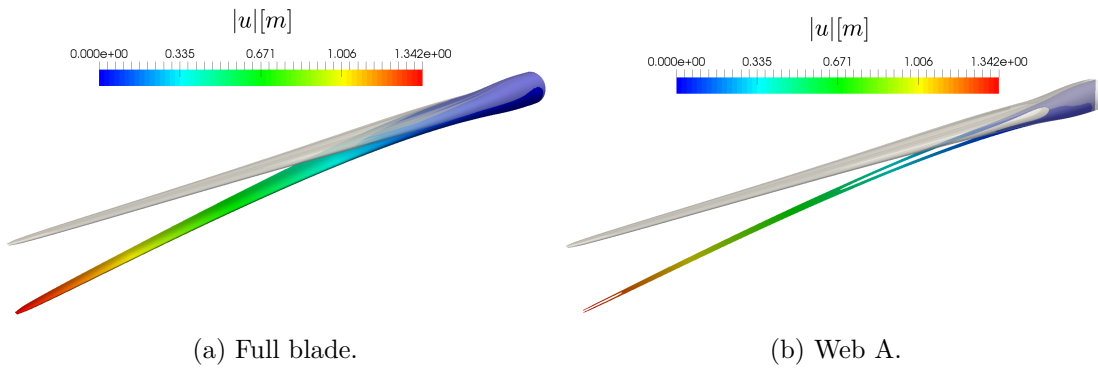


Figure 6.36 – L^2 norm of the displacement field on the topologically optimized blade subject to gravitational load, B-splines of degree $p = 2$, 84250 elements, results warped by a factor 5.

7 A note on the critical time step in isogeometric trimmed explicit dynamics¹

This chapter focuses on the estimation of the critical time step for the explicit dynamics of trimmed structures. Specifically, starting our investigation from a simple one-dimensional bar, we shed some lights on the spurious eigenvalues and eigenvectors stemming from:

1. the use of a lumped mass matrix instead of the consistent one,
2. trimming.

We then extend our findings to Kirchhoff plates, where similar results are observed. We remark that these are preliminary findings and further studies are needed to draw some definitive conclusions on the impact of trimming in dynamics. Indeed, it is likely that these spurious modes are not activated by common loading scenarios in many engineering applications. However, understanding the origin and the role of this mechanism is important to estimate the resulting accuracy in an explicit simulation. The notation and the examples presented in this chapter follow closely [Leidinger et al., 2019; Leidinger, 2020].

7.1 Numerical stability of the time integration

Let us start our derivation from the problem of elastodynamics given in Equation (2.6). If we apply the standard Galerkin procedure, the (linearized) semi-discrete equations of motions can be written in algebraic form as:

$$M\ddot{u} + Ku = f, \quad (7.1)$$

¹Some results presented in this chapter have been obtained in collaboration with Mr. Davide D'Angella and Dr. Lukas Leidinger. Their contributions are gratefully acknowledged.

where \mathbf{M} denotes the mass matrix of the underlying system and $\ddot{\mathbf{u}}$ represents the acceleration. Note that we have dropped the superscript tilde for clarity of the notation. Moreover, for simplicity, damping effects have been neglected. Now, following [Belytschko et al., 2000], the system of equations given above is diagonalized by exploiting the modal decomposition. By assuming a separation of variables of the form:

$$\mathbf{u}(t) = \boldsymbol{\varphi}_\ell e^{i\sqrt{\lambda_\ell^h} t}, \quad (7.2)$$

and substituting this ansatz into the homogeneous counterpart of Equation (7.1), the associated eigenvalue problem which governs the free vibrations of a structure reads:

$$(\mathbf{K} - \lambda_\ell^h \mathbf{M}) \boldsymbol{\varphi}_\ell = \mathbf{0} \quad \ell = 1, 2, \dots, n_{\text{dof}}, \quad (7.3)$$

where we assume that all eigenvalues are non-negative and are sorted as:

$$0 \leq \lambda_1^h \leq \lambda_2^h \leq \dots \leq \lambda_{\max}^h, \quad (7.4)$$

and the eigenvectors verify the following orthonormality condition:

$$\boldsymbol{\varphi}_\ell^\top \mathbf{M} \boldsymbol{\varphi}_m = \delta_{\ell m}. \quad (7.5)$$

Moreover, the set of eigenvectors $\{\boldsymbol{\varphi}_\ell\}$ forms a basis of $\mathbb{R}^{n_{\text{dof}}}$. Consequently, we can write the solution \mathbf{u} as a linear combination of coefficients and eigenvectors as:

$$\mathbf{u}(t) = \sum_{\ell=1}^{n_{\text{dof}}} \tilde{u}_\ell(t) \boldsymbol{\varphi}_\ell. \quad (7.6)$$

This allows us to reduce Equation (7.1) to a scalar single-dof initial-value problem, formulated for every dof in the system, e.g. see [Hughes, 2000] for a detailed derivation. Assuming a central difference scheme for the time discretization [Sheppard, 1899], we can now state the following estimation of the critical time step size:

$$\Delta t_{\text{crit}} = \min_{\ell} \left(\frac{2}{\sqrt{\lambda_\ell^h}} \right) = \frac{2}{\sqrt{\lambda_{\max}^h}}, \quad (7.7)$$

where the last equality is straightforward to verify. A detailed derivation of the stability analysis leading to the previous result can be found in [Belytschko et al., 2000].

7.2 A minimal stabilization technique for small cut elements

In the following, we adapt the minimal stabilization studied in [Buffa et al., 2020] in the context of Dirichlet boundary conditions imposed via Nitsche's method to the stiffness

and mass matrices appearing in Equation (7.3). Let us briefly outline the method here. Given an user-defined parameter $\vartheta \in (0, 1]$, let us define the element $K \in \mathcal{T}$ as a *good* element if:

$$\frac{|\Omega \cap K|}{|K|} \leq \vartheta, \quad (7.8)$$

else K is considered *bad*. Consequently, we denote by \mathcal{T}^b and \mathcal{T}^g the collections of all bad and good elements in the mesh, respectively. Assuming that for each bad element K there exists a good neighbor K' , the stabilization is constructed as follows. For each bad element $K \in \mathcal{T}^b$, the spline functions belonging to the good neighbor K' are L^2 -projected onto the polynomial space $\mathbb{Q}_p(K')$. The latter denotes the space of polynomials on K' of degree p in each parametric direction. Then, their polynomial extension is used to evaluate the mass and stiffness matrices, respectively, in K . We remark that this operation is performed locally. For the sake of conciseness, we do not report the full derivation here but we refer to [Haslinger and Renard, 2009; Buffa et al., 2020; Antolin et al., 2021] for further details.

7.3 Numerical investigation

In the following numerical examples, we study the behavior of critical time step Δt_{crit} on a series of benchmark problems. Furthermore, we also focus on the appearance of spurious eigenvalues and eigenvectors in the discrete spectrum. In our investigation, we distinguish between the *consistent* mass matrix \mathbf{M}^c , whose entries are defined as:

$$M_{ij}^c = \int_{\Omega} \rho \mathcal{B}_i \mathcal{B}_j \, d\Omega, \quad (7.9)$$

and the so-called (row-sum) *lumped* mass matrix \mathbf{M}^ℓ , whose elements read:

$$M_{jj}^\ell = \int_{\Omega} \rho \sum_i \mathcal{B}_i \mathcal{B}_j \, d\Omega = \int_{\Omega} \rho \mathcal{B}_j \, d\Omega. \quad (7.10)$$

\downarrow from the partition of unity.

For engineering applications, the latter is often preferred from a computational standpoint since its diagonal structure allows for an easy matrix inversion, where this step turns out to be the major bottleneck in the central difference scheme.

7.3.1 One-dimensional bar example

In this first example, we reproduce the results presented in [Leidinger, 2020] on a one-dimensional bar. Similar observations in the non-trimmed case can be found in [Cottrell et al., 2006; Hughes et al., 2008]. We consider the trimmed computational domain

Chapter 7. A note on the critical time step in isogeometric trimmed explicit dynamics

$\Omega = [0, 0.75 + \varepsilon]$, where ε represents the trimming parameter measuring the distance from the last physical knot, see Figure 7.1. Without loss of generality, we set the Young's modulus $E = 1$, the density of the bar $\rho = 1$ and the area of the cross section of the structure $A = 1$. Then, we apply homogeneous Dirichlet boundary conditions on one end

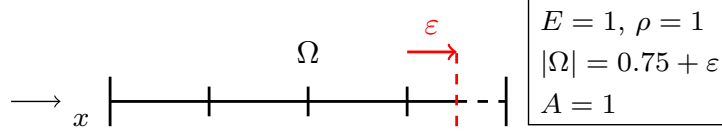


Figure 7.1 – Setup of the one-dimensional bar example.

of the bar and homogeneous Neumann on the trimmed one. For a fixed-free ends bar the analytical eigenvalues read:

$$\lambda_\ell^{\text{ex}} = \frac{2\ell - 1}{2} \frac{\pi}{|\Omega|} \quad \ell = 1, 2, 3, \dots, +\infty. \quad (7.11)$$

Now, we solve the generalized eigenvalue problem given in Equation (7.3) and we compute the estimated critical time step from Equation (7.7) for B-splines of degree $p = 1, \dots, 4$ and the consistent mass matrix and lumped mass matrix, respectively. The results on a fixed mesh with 49 active elements are reported in Figure 7.2 for splines of maximum and C^0 -continuity, respectively. Analogously to [Leidinger, 2020], we observe that B-splines of degree $p \geq 2$ and maximum continuity yield a critical time step independent from the trimming parameter ε when the lumped mass matrix is used. Furthermore, their

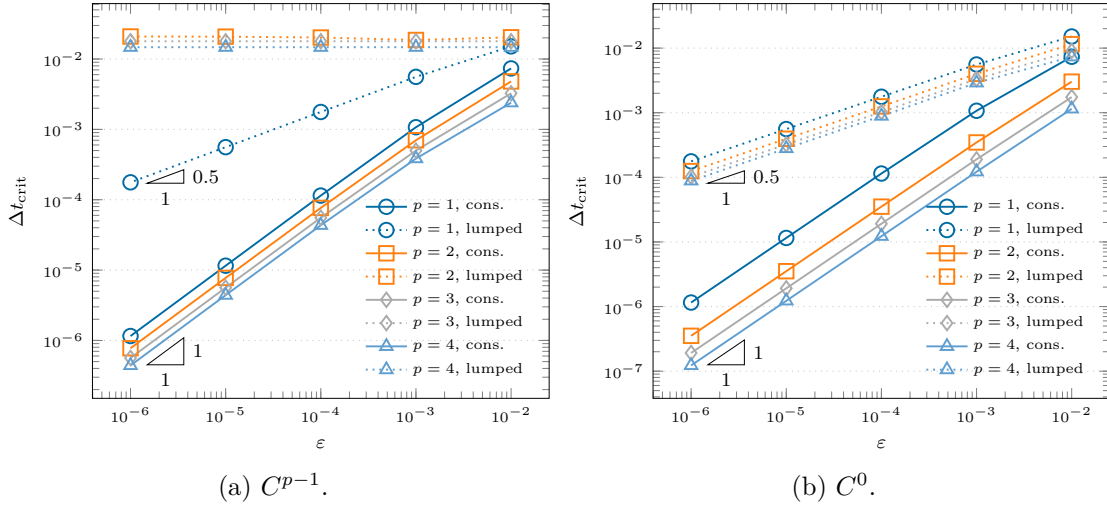


Figure 7.2 – Convergence study of the critical time step for C^{p-1} and C^0 splines of different degrees $p = 1, \dots, 4$, consistent and lumped mass matrices, trimmed one-dimensional bar.

corresponding normalized spectra are depicted in Figure 7.3. We observe that, in the case of mass lumping, the approximation of the lower frequencies deteriorates as the

trimmed element gets smaller. This is caused by spurious eigenvalues (and corresponding eigenvectors) appearing in the lower part of the spectrum which cause a shift in the frequency domain, where an example for splines of degree $p = 3$ and $\varepsilon = 10^{-6}$ is provided in Figure 7.4. In general, their position depend on the trimming parameter, the mesh size h and the degree of the discretization p , making the development of an algorithmic way to detect them difficult. Moreover, the impact of these spurious eigenvalues on the accuracy of a dynamic simulation is not yet fully understood. Lastly, the critical time step estimation obtained with the stabilization is reported in Figure 7.5 for B-splines of degree $p = 1, \dots, 4$ and maximum and C^0 -continuity, respectively. In this example, we set θ such that the trimmed element is always stabilized. For all cases, the stabilized generalized eigenvalue problem yields a critical time step that is independent from the trimming parameter, where the use of a lumped mass matrix allows for larger time steps.

Remark 7.1 *Although the stabilization solves the dependency of the critical time step on the trimming parameter, we highlight that spurious eigenvalues and corresponding eigenfrequencies are still present in the stabilized spectrum.*

7.3.2 Two-dimensional rectangular Kirchhoff plate

In the following example, we study the eigenvalues and corresponding eigenvectors on a trimmed, rectangular, Kirchhoff plate. For the interested reader, the unforced vibration of thin plates have been thoroughly studied in [Cottrell et al., 2006; Shojaee et al., 2012] in the context of non-trimmed IGA. We consider the trimmed computational domain $\Omega = [0.3 - \varepsilon, 0.7 + \varepsilon] \times [0.0, 0.7 + \varepsilon]$, where again ε represents the trimming parameter. Without loss of generality, we set the physical parameters of the structure such that the flexural rigidity D is unitary. As boundary condition, the plate is clamped at the untrimmed edge corresponding to $y = 0$. The setup of the problem is depicted in Figure 7.6. Similarly to the previous example, we compute the estimated critical time step for B-splines of degree $p = 2, \dots, 5$ and the consistent mass matrix and lumped mass matrix, respectively. The results on a fixed mesh with 49 active elements are reported in Figure 7.7 for splines of maximum and C^1 -continuity, respectively. Interestingly, we observe that for B-splines of degree $p \geq 4$ and maximum continuity we obtain a critical time step independent from the trimming parameter ε when the lumped mass matrix is used. For cubic C^2 B-splines, we observe a dependency of the critical time step with an order of convergence proportional to square root of ε . This dependency is less severe compared to their reduced-continuity counterpart. Also in Figure 7.7, we report the results for splines of maximum smoothness where the stabilization is used to mitigate the effects related to trimming and the stabilization parameter $\theta = 1$ is set. Analogously to the one-dimensional case, we observe that the critical time step becomes independent of the trimming parameter for all degrees and for both consistent and lumped mass matrices, respectively. Furthermore, larger time steps can be used when the lumped mass

Chapter 7. A note on the critical time step in isogeometric trimmed explicit dynamics

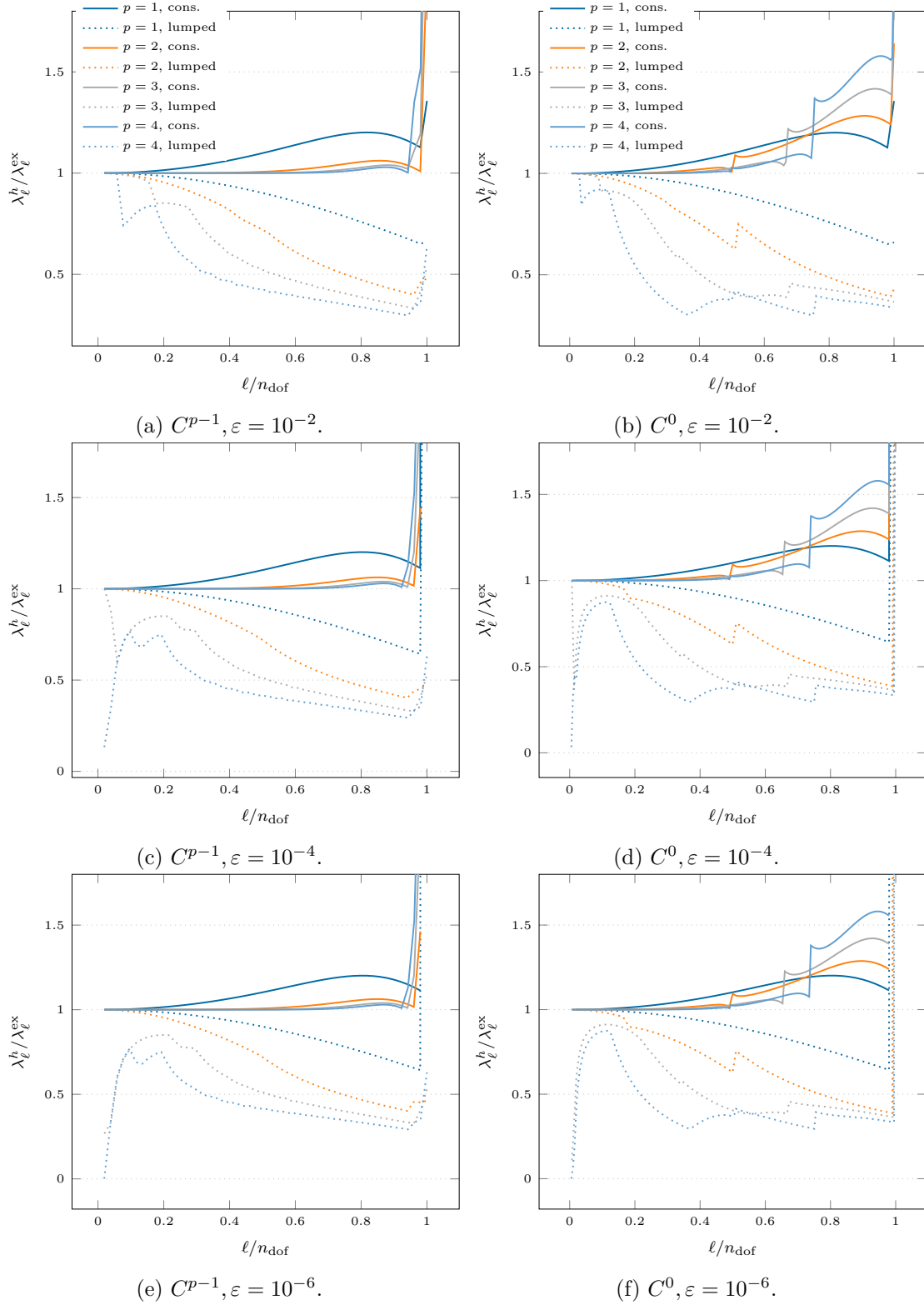


Figure 7.3 – Spectra for C^{p-1} - and C^0 -continuous splines of different degrees $p = 1, \dots, 4$ and various trimming parameters $\varepsilon = [10^{-2}, 10^{-4}, 10^{-6}]$, trimmed one-dimensional bar.

7.3. Numerical investigation

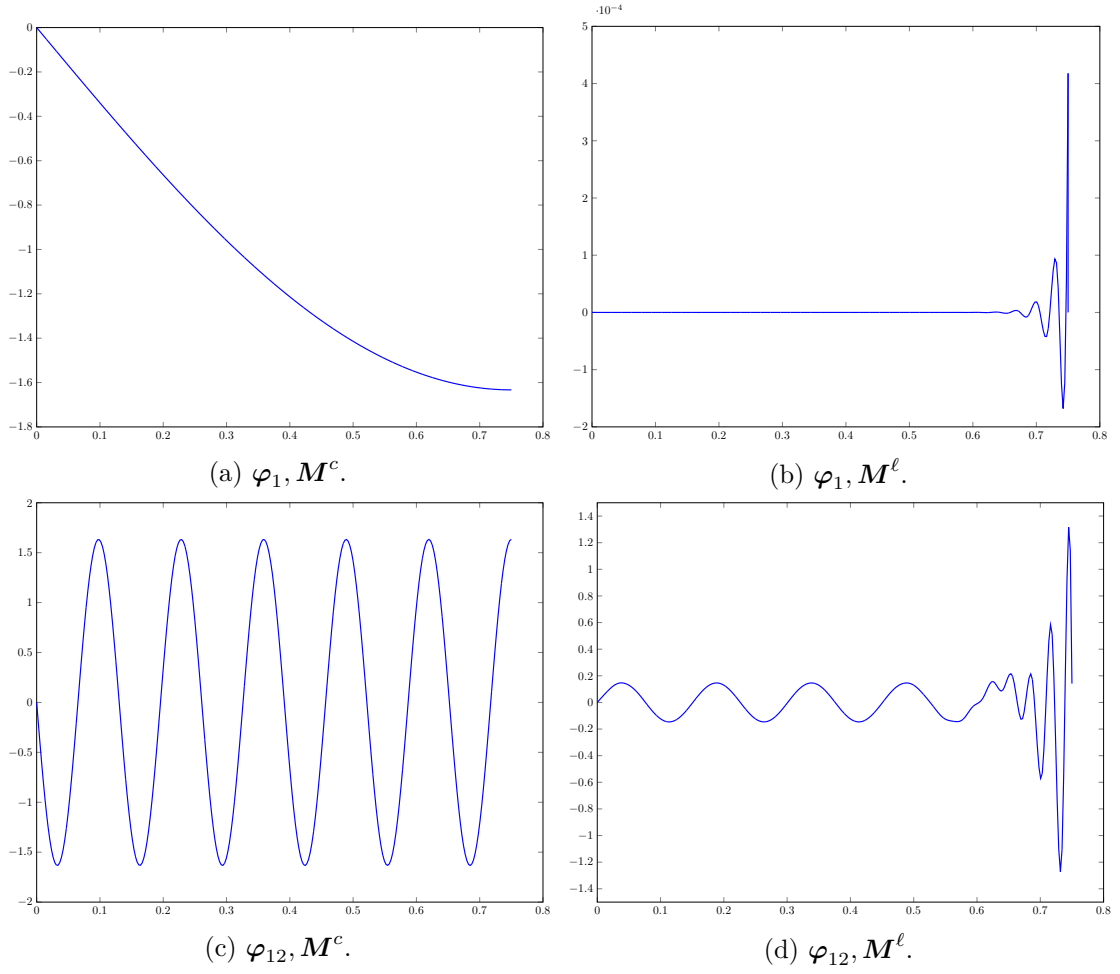


Figure 7.4 – Numerical eigenvectors φ_1 and φ_{12} corresponding to splines of degree $p = 3$ and trimming parameter $\varepsilon = 10^{-6}$. Note the spurious, localized eigenvectors stemming from trimming and the use of the lumped mass matrix.

Chapter 7. A note on the critical time step in isogeometric trimmed explicit dynamics

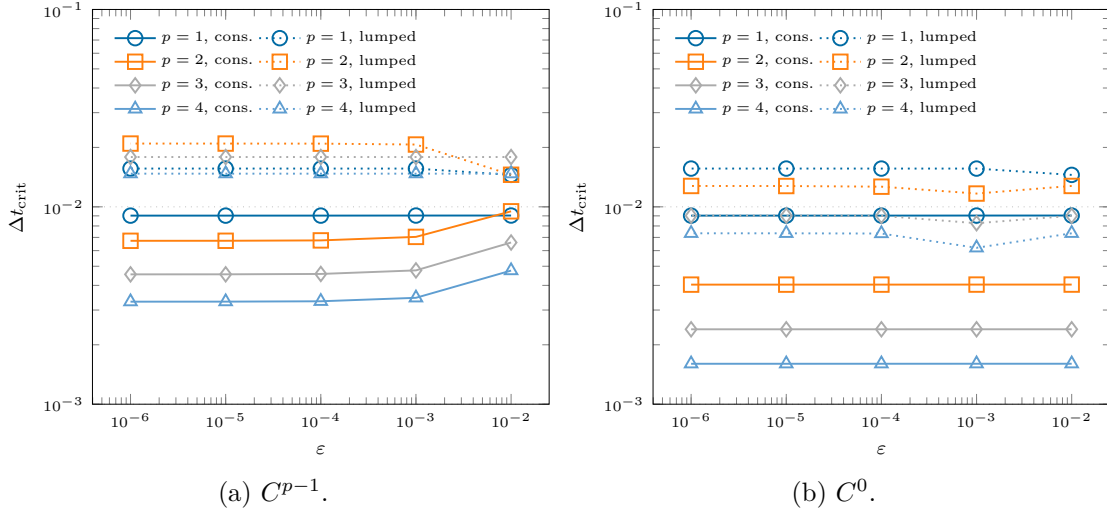


Figure 7.5 – Convergence study of the critical time step for C^{p-1} and C^0 splines of different degrees $p = 1, \dots, 4$, stabilized consistent and lumped mass matrices, trimmed one-dimensional bar.

matrix is employed. Lastly, an example of the spurious modes is depicted in Figure 7.8 for cubic B-splines and $\varepsilon = 10^{-2}$. We remark that the trimming pattern used in this numerical experiments is pathological, since it creates extremely small elements of size ε^2 at the top corners. These findings could potentially impact the use of Kirchhoff plates and Kirchhoff-Love shells in explicit dynamics, however further numerical and theoretical investigations are needed to shed some light on the impact of trimming for the explicit dynamics simulation of isogeometric Kirchhoff-Love shells. In particular, the effects of stabilization techniques and weakly-imposed boundary/coupling conditions must be analyzed in depth.

7.3. Numerical investigation

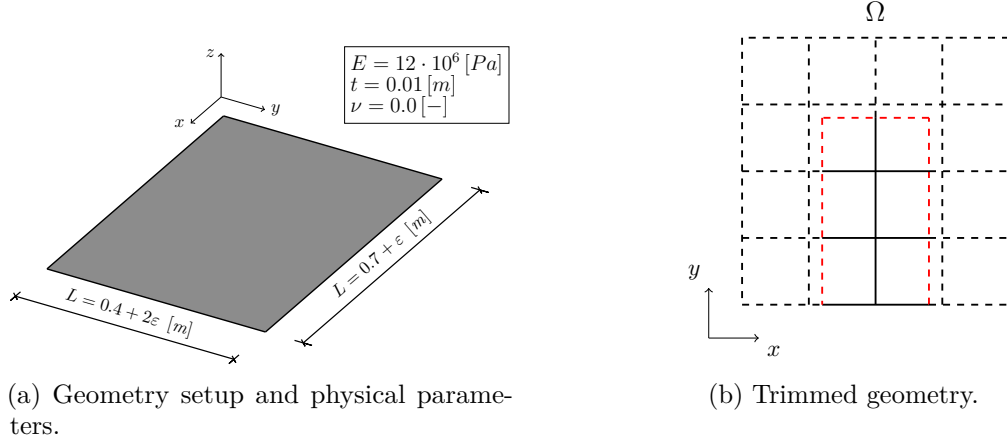


Figure 7.6 – Problem setup and trimmed geometry for the rectangular plate example.

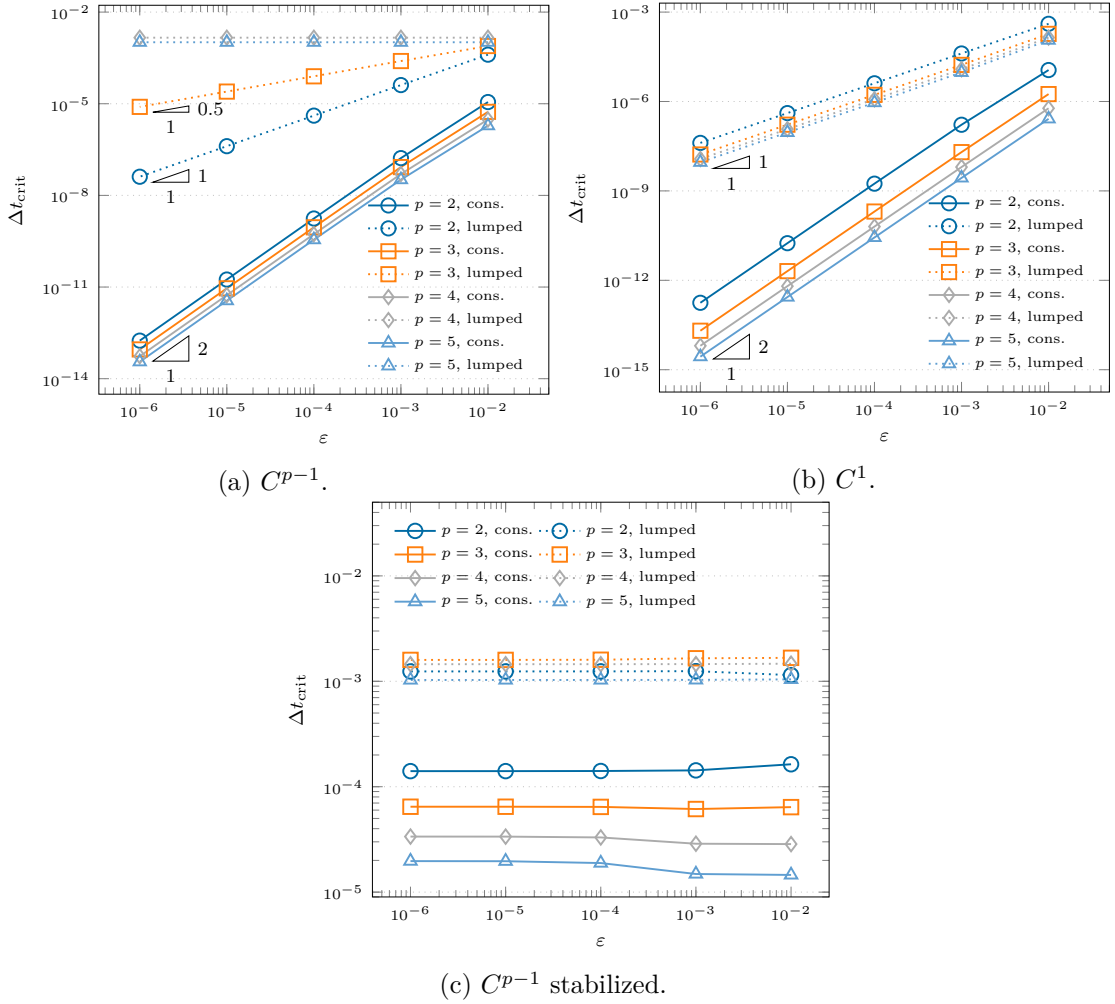


Figure 7.7 – Convergence study of the critical time step for C^{p-1} - and C^1 -continuous splines of different degrees $p = 2, \dots, 5$, consistent and lumped mass matrices, trimmed rectangular plate.

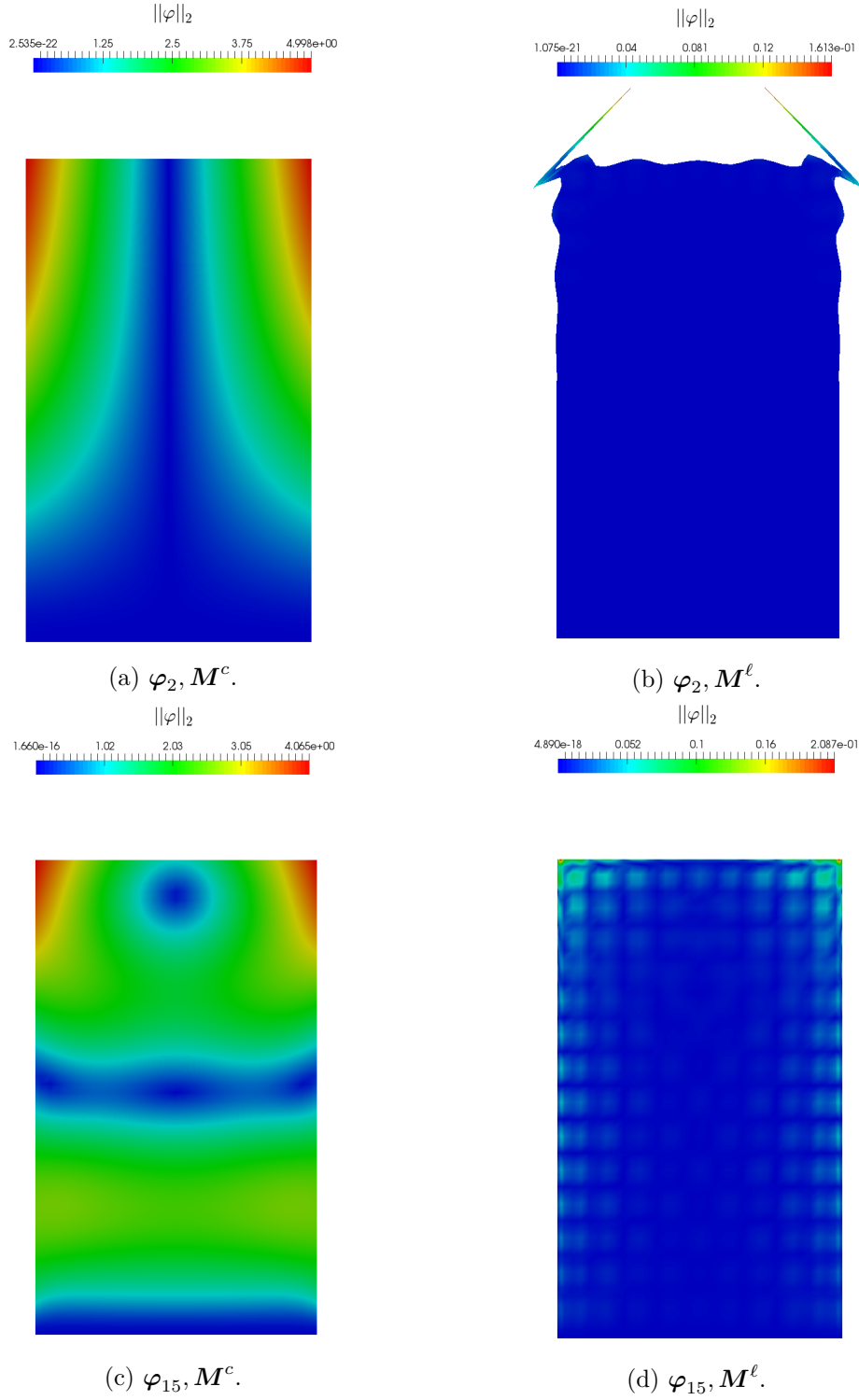


Figure 7.8 – Numerical eigenvectors φ_2 and φ_{15} corresponding to splines of degree $p = 3$ and trimming parameter $\varepsilon = 10^{-2}$. Note the spurious, localized eigenvectors stemming from trimming and the use of the lumped mass matrix.

7.3.3 Two-dimensional rotating plate

In the last example, we consider the trimmed computational domain $\Omega = [0.19, 0.78] \times [0.22, 0.78]$ embedded into the unit square $\Omega_0 = [0, 1]^2$, where the untrimmed domain Ω_0 is discretized by a fixed mesh of 20 elements in each parametric direction. This setup takes inspiration from [de Prenter et al., 2017; Buffa et al., 2020]. The plate is then rotated around its center of mass for a series of angles $\alpha_i = i\pi/200, i = 0, \dots, 100$, see Figure 7.9 for an example. This setup creates a variety of trimming patterns, where small cut elements may arise. To quantify how badly elements are trimmed, let us introduce the smallest volume fraction, which is defined as $\eta = \min_{K \in \mathcal{T}} |\Omega \cap K|$. Then, we set the stabilization parameter $\theta = 0.5$. In Figure 7.10, we plot the critical time step for splines of degree $p = 2, 3$ in the non-stabilized and stabilized cases, respectively, against the volume fraction η , where again a consistent and a lumped mass matrices are used. Similarly to the examples studied above, the stabilization mitigates the detrimental effects of badly cut elements. This is particularly apparent when a consistent mass matrix is used. Indeed, in this case, it is clear that the stabilized critical time step becomes independent from the volume fraction, allowing for selecting larger time steps in dynamics.

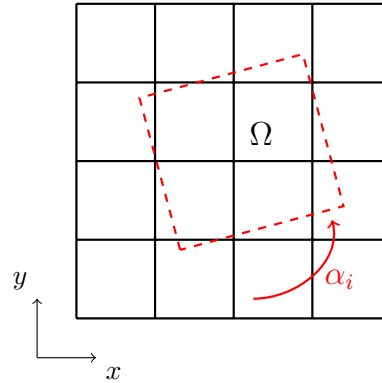


Figure 7.9 – Problem setup for the trimmed rotating plate example.

Remark 7.2 *Although not reported here, our preliminary observations appear to hold true also for (trimmed) trivariate B-splines.*

Chapter 7. A note on the critical time step in isogeometric trimmed explicit dynamics

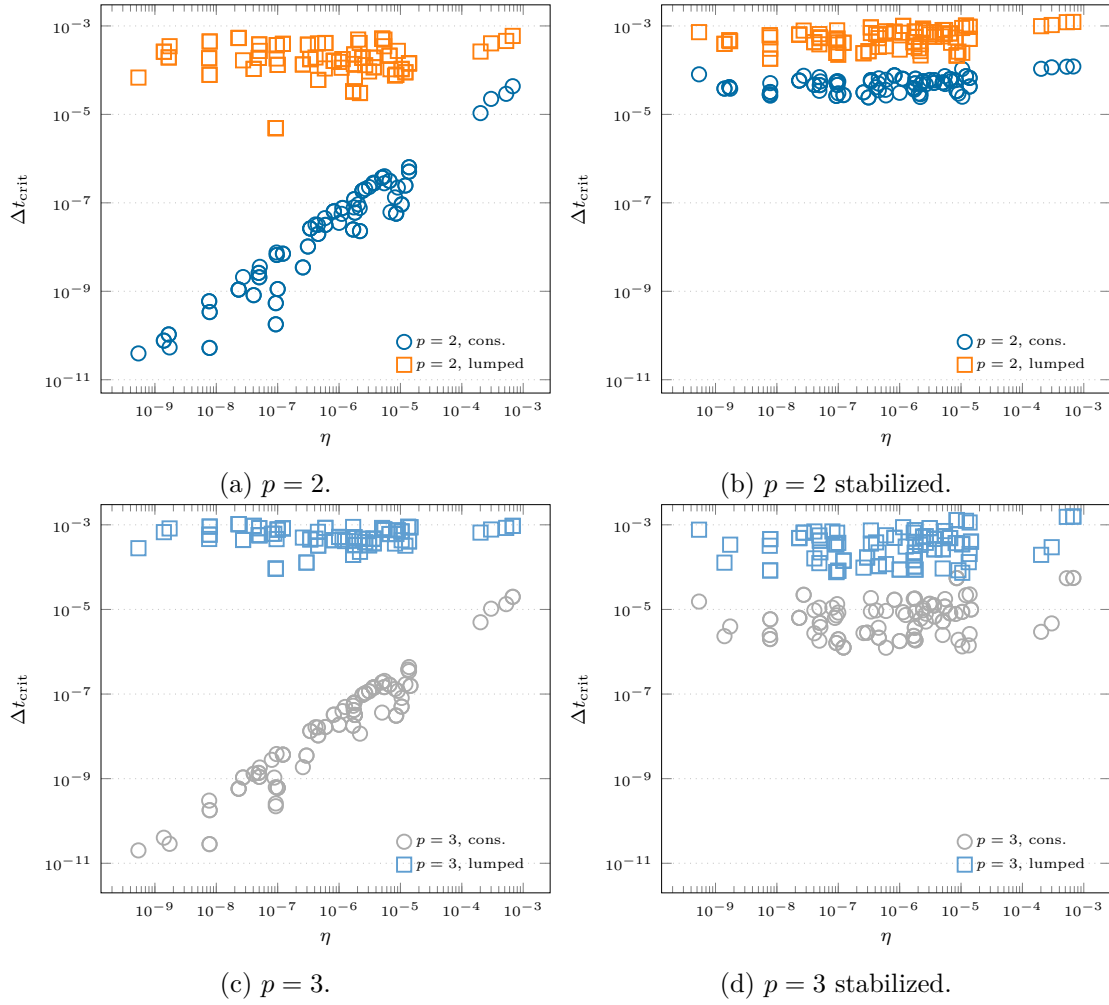


Figure 7.10 – Convergence study of the critical time step for different degrees $p = 2, 3$, consistent and lumped mass matrices, without and with stabilization, trimmed rotating plate.

8 Concluding remarks and future outlook

This thesis mainly focuses on increasing and guaranteeing the accuracy of isogeometric methods for the analysis of trimmed Kirchhoff plates and Kirchhoff-Love shells. In particular, our aim is to develop an efficient computational framework able to tackle industrial-like applications. This goal has been pursued by employing several different methodologies outlined in the following.

8.1 Scientific contributions

Here, we systematically summarize the main scientific contributions of this work. These findings are based on the following published papers [Antolin et al., 2020; Coradello et al., 2020a,b, 2021b] and a manuscript currently under review for publication [Coradello et al., 2021a].

8.1.1 Local refinement of trimmed shells

In the first part of our numerical investigation, we have shown the beneficial effects of employing local refinement for the analysis of trimmed, complex B-Rep surfaces. In particular, we have highlighted how the \mathcal{HB} and \mathcal{THB} bases mitigate the issues related to the weak imposition of Dirichlet-type boundary conditions and to the presence of thin holes. Moreover, we have numerically verified how local refinement can efficiently reduce and eventually eliminate the spurious effects of over-constraining of the solution space in the proximity of weakly constrained trimming curves. Furthermore, we have presented a simple algorithmic way to remove the unphysical coupling of basis functions across “small” geometrical features. This is an essential starting point for the development of an efficient framework capable of handling trimmed surfaces with complex features and details defined at different scales.

8.1.2 Adaptive simulation of Kirchhoff plates and Kirchhoff-Love shells

Next, we have addressed the issues of:

- efficiently compute an error estimator for fourth-order PDEs,
- steer an adaptive simulation via the aforementioned estimator,
- extend the framework to suitably account for trimming.

Specifically, a novel *a posteriori* implicit error estimator has been developed and its performance has been assessed. The evaluation of this estimator is based on the solution of an additional, residual-like variational problem formulated in the so-called bubble space. The latter is composed of Bernstein polynomials defined locally on active elements, constructed in a way such that the C^1 -continuity constraint is directly built into the space. We remark that, thanks to this choice of the aforementioned space, the resulting linear system is in general small, block-diagonal and easily-invertible. Moreover, this method is suitable for parallelization and straightforward to implement on top of existing isogeometric codes. More importantly, it is computationally cheap compared to classical residual-type estimators since it avoids the computation of the residual in a strong sense. On one hand, this is a major advantage particularly for Kirchhoff-Love shells, since the evaluation of covariant derivatives is a tedious task, which becomes very quickly computationally expensive and, for all practical purposes, almost intractable from a numerical standpoint. On the other hand, with this technique we also avoid the computation of integral terms involving the evaluation of the jump of the derivatives across element boundaries. For Kirchhoff plates, a typical speed up of one order of magnitude has been observed.

This method can be readily extended to incorporate trimming. In particular, all the aforementioned favorable features carry over to the trimmed case. This allows us to take another step towards the integration of design and automated adaptive isogeometric analysis, where we have validated the proposed framework on an extensive series of benchmark problems. With these investigations, we have numerically proved the reliability and robustness of the proposed error-driven adaptive framework for the simulation of various mechanically-relevant PDEs defined on trimmed domains. In particular, the proposed method systematically achieves superior efficiency and accuracy per-degree-of-freedom in problems showing sharp features of the solution and/or singularities. Lastly, its applicability to industrial-like geometries has been demonstrated on the B-pillar of a car.

8.1.3 Coupling of Kirchhoff plates and Kirchhoff-Love shells

A further key step to reach the goal of this thesis consists in the proper treatment and coupling of multi-patch designs. Typically, these geometric models are represented by a collection of trimmed B-Reps. Therefore, a reliable and accurate coupling strategy capable of handling trimmed, non-conforming patches is required. To achieve this, we have developed a method based on the L^2 -projection of suitable penalty terms at the corresponding coupling interface onto a degree-reduced space. The choice of this space is motivated by the stable $p/p - 2$ pairing studied in the scope of mortar methods. On one hand, the projection mitigates the detrimental effects related to interface locking starting from very coarse discretization. On the other hand, it gives us insights into the proper scaling of the penalty parameters based on the underlying discretization. Consequently, the proposed coupling method retains the optimal rates of convergence achievable by B-splines, as demonstrated by our findings on an extensive series of benchmark problems. Moreover, our approach is fully parameter-free, since the penalty coefficients are completely determined by the problem setup. The proposed strategy is particularly suited for spline spaces of moderate degrees $p = 2, 3$, where the projection turns out to be computationally efficient and the condition number stemming from the super-penalty does not yield a significant deterioration of the solution accuracy. To conclude this part, the applicability of our method to tackle complex structure of engineering relevance has been studied. In particular, we have performed a static shell analysis of the DTU 10MW Reference wind turbine blade. Furthermore, since trimming is naturally incorporated into our methodology, we have carried out a simplified topology optimization of the internal shear webs. This example demonstrates that the proposed computational framework is able to efficiently and accurately handle industrial optimization loops.

8.2 Future outlook

Clearly, this thesis provides some initial insights and a proof-of-concept for the topics presented above. However, many questions remain unanswered and further investigations are needed to improve the proposed computational framework. We list some potential future research directions in the following.

8.2.1 Automatic detection of refinement depth

In the first part of this thesis, a user-defined parameter has been used to determine the maximum depth of the refinement. In principle, one could algorithmically find a priori the suitable depth based on the smallest feature of the model. However, this entails performing a search over the set of trimming loops in order to define a “characteristic length” of the detail. While this seems a reasonable approach that has to be performed

only once as a pre-processing step, the difficulty resides in performing this task efficiently. Moreover, many complex CAD designs present physical and/or artificial small features that might or might not contribute significantly to the mechanical response. In order to avoid an unreasonable refinement depth, an automatic de-feature strategy based on the work in [Buffa et al., 2021] could be employed to determine which features should be resolved by local refinement.

8.2.2 Possible extensions of the bubble error estimator

In its current implementation, the error estimator does not take into account the residual caused by the weak imposition of boundary conditions and coupling constraints. To perform this task, the method should be extended by using the same idea employed for the imposition of in-homogeneous Neumann boundary conditions, where suitable boundary bubble functions are added to the space. Furthermore, although some preliminary results confirm the performance in the presence of small holes, its behavior should be systematically assessed on a series of geometries with small features.

From a computational standpoint, the current code should be parallelized to take full advantage of the structure of the linear system stemming from the weak residual equation. This would increase even further the typical speed-ups observed when comparing the run-time to a classical residual-type estimator.

8.2.3 Possible extensions of the coupling strategy

For the time being, the coupling strategy has been implemented and tested for geometrically linear Kirchhoff-Love shells. The extension to non-linearities entails the projection of additional terms which vanish when the reference and deformed configurations coincide. Clearly, this constitutes an exciting development of the method which would allow the simulation of more complex mechanical responses.

Additionally, in our numerical experiments we have not thoroughly studied the accuracy of the stress field along and in the proximity of trimmed coupling interfaces. This aspect should be further assessed in the future.

Furthermore, to scale the current simulation capabilities beyond a few million degrees of freedom, a suitable preconditioner for shells is necessary. Based on the work we have presented for Kirchhoff plates, the nested preconditioner could be formulated in a fully algebraic fashion without the need of introducing the fast diagonalization algorithm. Instead, the method could rely on an algebraic multigrid preconditioner, which is typically regarded as standard in linear algebra packages. Alternatively, one could change approach and try to adapt the FETI method [Farhat and Roux, 1991] and variants thereof to the problem at hand.

8.2.4 Future directions in trimmed explicit dynamics

Clearly, the preliminary results we have presented in the scope of trimmed explicit dynamics can be expanded in numerous directions. A few interesting questions can be summarized as follows:

- perform an in-depth study of the effects of the stabilization on the critical time step for Kirchhoff plates and Kirchhoff-Love shells,
- deeper theoretical understanding of the origin of spurious eigenvalues/eigenvectors and the role they play in a dynamical simulation,
- study of the impact of the proposed coupling strategy on the critical time step,
- systematically investigate the volumetric case.

Bibliography

- Ainsworth, M. and Oden, J. T. (1997). A posteriori error estimation in finite element analysis. *Computer Methods in Applied Mechanics and Engineering*, 142(1):1 – 88.
- Albanesi, A. E., Peralta, I., Bre, F., Storti, B. A., and Fachinotti, V. D. (2020). An optimization method based on the evolutionary and topology approaches to reduce the mass of composite wind turbine blades. *Structural and Multidisciplinary Optimization*, 62(2):619–643.
- Antolin, P., Buffa, A., and Coradello, L. (2020). A hierarchical approach to the a posteriori error estimation of isogeometric Kirchhoff plates and Kirchhoff–Love shells. *Computer Methods in Applied Mechanics and Engineering*, 363:112919.
- Antolin, P., Buffa, A., and Martinelli, M. (2019). Isogeometric Analysis on V-reps: First results. *Computer Methods in Applied Mechanics and Engineering*, 355:976 – 1002.
- Antolin, P., Buffa, A., Puppi, R., and Wei, X. (2021). Overlapping Multipatch Isogeometric Method with Minimal Stabilization. *SIAM Journal on Scientific Computing*, 43(1):A330–A354.
- Apostolatos, A. (2019). *Isogeometric Analysis of Thin-Walled Structures on Multipatch Surfaces in Fluid-Structure Interaction*. Dissertation, Technische Universität München, München.
- Apostolatos, A., Breitenberger, M., Wüchner, R., and Bletzinger, K.-U. (2015). *Isogeometric Analysis and Applications 2014*, chapter Domain Decomposition Methods and Kirchhoff-Love Shell Multipatch Coupling in Isogeometric Analysis, pages 73–101. Springer International Publishing, Cham.
- Arnold, D. N. and Brezzi, F. (1997). Locking-Free Finite Element Methods for Shells. *Mathematics of Computation*, 66(217):1–14.
- Babuška, I. (1973). The Finite Element Method with Penalty. *Mathematics of Computation*, 27(122):221–228.

Bibliography

- Babuška, I. (1981). A-posteriori error estimation for the finite element method. In Wunderlich, W., Stein, E., and Bathe, K.-J., editors, *Nonlinear Finite Element Analysis in Structural Mechanics*, pages 3–10, Berlin, Heidelberg. Springer Berlin Heidelberg.
- Babuška, I. and Miller, A. (1987). A feedback finite element method with a posteriori error estimation: Part I. The finite element method and some basic properties of the a posteriori error estimator. *Computer Methods in Applied Mechanics and Engineering*, 61(1):1–40.
- Babuška, I. and Rheinboldt, W. C. (1978). Error Estimates for Adaptive Finite Element Computations. *SIAM Journal on Numerical Analysis*, 15(4):736–754.
- Babuška, I. and Suri, M. (1992a). Locking effects in the finite element approximation of elasticity problems. *Numerische Mathematik*, 62(1):439–463.
- Babuška, I. and Suri, M. (1992b). On Locking and Robustness in the Finite Element Method. *SIAM Journal on Numerical Analysis*, 29(5):1261–1293.
- Babuška, I. and Vogelius, M. (1984). Feedback and adaptive finite element solution of one-dimensional boundary value problems. *Numerische Mathematik*, 44(1):75–102.
- Bak, C., Zahle, F., Bitsche, R., Kim, T., Yde, A., Henriksen, L., Andersen, P., Natarajan, A., and Hansen, M. (2013). Design and performance of a 10 MW wind turbine. Technical report, DTU.
- Bank, R. E. and Smith, R. K. (1993). A posteriori error estimates based on hierarchical bases. *SIAM Journal on Numerical Analysis*, 30(4):921–935.
- Bank, R. E. and Weiser, A. (1985). Some A Posteriori Error Estimators for Elliptic Partial Differential Equations. *Mathematics of Computation*, 44(170):283–301.
- Batoz, J.-L., Bathe, K.-J., and Ho, L.-W. (1980). A study of three-node triangular plate bending elements. *International Journal for Numerical Methods in Engineering*, 15(12):1771–1812.
- Bazilevs, Y., Beirão Da Veiga, L., Cottrell, J. A., Hughes, T. J. R., and Sangalli, G. (2006). Isogeometric analysis: Approximation, stability and error estimates for h-refined meshes. *Mathematical Models and Methods in Applied Sciences*, 16(07):1031–1090.
- Bazilevs, Y., Calo, V., Cottrell, J., Evans, J., Hughes, T. J. R., Lipton, S., Scott, M., and Sederberg, T. (2010). Isogeometric analysis using T-splines. *Computer Methods in Applied Mechanics and Engineering*, 199(5):229 – 263. Computational Geometry and Analysis.
- Bazilevs, Y., Hsu, M.-C., and Scott, M. (2012). Isogeometric fluid–structure interaction analysis with emphasis on non-matching discretizations, and with application to wind turbines. *Computer Methods in Applied Mechanics and Engineering*, 249-252:28–41. Higher Order Finite Element and Isogeometric Methods.

- Beirão da Veiga, L., Buffa, A., Sangalli, G., and Vázquez, R. (2014). Mathematical analysis of variational isogeometric methods. *Acta Numerica*, 23:157–287.
- Beirão da Veiga, L., Buffa, A., Sangalli, G., and Vázquez, R. (2013). Analysis-suitable T-splines of arbitrary degree: definition, linear independence and approximation properties. *Mathematical Models and Methods in Applied Sciences*, 23(11):1979–2003.
- Belytschko, T., Liu, W., and Moran, B. (2000). *Nonlinear Finite Elements for Continua and Structures*. John Wiley & Sons, Ltd.
- Belytschko, T., Stolarski, H., Liu, W. K., Carpenter, N., and Ong, J. S. (1985). Stress projection for membrane and shear locking in shell finite elements. *Computer Methods in Applied Mechanics and Engineering*, 51(1):221 – 258.
- Benzaken, J., Evans, J. A., McCormick, S. F., and Tamstorf, R. (2021). Nitsche’s method for linear Kirchhoff–Love shells: Formulation, error analysis, and verification. *Computer Methods in Applied Mechanics and Engineering*, 374:113544.
- Benzaken, J., Herrema, A., Hsu, M.-C., and Evans, J. (2017). A rapid and efficient isogeometric design space exploration framework with application to structural mechanics. *Computer Methods in Applied Mechanics and Engineering*, 316:1215 – 1256. Special Issue on Isogeometric Analysis: Progress and Challenges.
- Bézier, P. (1977). *Essai de définition numérique des courbes et des surfaces expérimentales: contribution à l’étude des propriétés des courbes et des surfaces paramétriques polynomiales à coefficients vectoriels*. Dissertation, Université Pierre-et-Marie-Curie.
- Bieber, S., Oesterle, B., Ramm, E., and Bischoff, M. (2018). A variational method to avoid locking–independent of the discretization scheme. *International Journal for Numerical Methods in Engineering*, 114(8):801–827.
- Bischoff, M., Bletzinger, K.-U., Wall, W. A., and Ramm, E. (2004). *Models and Finite Elements for Thin-Walled Structures*, chapter 3. American Cancer Society.
- Bletzinger, K. U., Bischoff, M., and Ramm, E. (2000). A unified approach for shear-locking-free triangular and rectangular shell finite elements. *Computers & Structures*, 75:321–334.
- Boffi, D., Brezzi, F., and Fortin, M. (2013). *Mixed Finite Element Methods and Applications*. Springer Series in Computational Mathematics. Springer Berlin Heidelberg.
- Bracco, C., Buffa, A., Giannelli, C., and Vázquez, R. (2019). Adaptive isogeometric methods with hierarchical splines: An overview. *Discrete & Continuous Dynamical Systems*, 39(1):241–261.
- Bracco, C., Giannelli, C., and Vázquez, R. (2018). Refinement algorithms for adaptive isogeometric methods with hierarchical splines. *Axioms*, 7(3).

Bibliography

- Brand, L. (1947). *Vector and tensor analysis / by Louis Brand*. John Wiley & sons, New York London.
- Breitenberger, M. (2016). *CAD-Integrated Design and Analysis of Shell Structures*. Dissertation, Technische Universität München, München.
- Breitenberger, M., Apostolatos, A., Philipp, B., Wüchner, R., and Bletzinger, K.-U. (2015). Analysis in computer aided design: Nonlinear isogeometric B-Rep analysis of shell structures. *Computer Methods in Applied Mechanics and Engineering*, 284:401 – 457. Isogeometric Analysis Special Issue.
- Brenner, S. C. and Scott, L. R. (2008). *The Mathematical Theory of Finite Element Methods*, volume 15 of *Texts in Applied Mathematics*. Springer.
- Bressan, A. (2013). Some properties of LR-splines. *Computer Aided Geometric Design*, 30(8):778 – 794.
- Brivadis, E., Buffa, A., Wohlmuth, B., and Wunderlich, L. (2015). Isogeometric mortar methods. *Computer Methods in Applied Mechanics and Engineering*, 284:292 – 319. Isogeometric Analysis Special Issue.
- Buffa, A., Chanon, O., and Vázquez, R. (2021). Analysis-aware defeaturing: problem setting and a posteriori estimation.
- Buffa, A. and Garau, E. M. (2018). A posteriori error estimators for hierarchical B-spline discretizations. *Mathematical Models and Methods in Applied Sciences*, 28(08):1453–1480.
- Buffa, A. and Giannelli, C. (2017). Adaptive isogeometric methods with hierarchical splines: Optimality and convergence rates. *Mathematical Models and Methods in Applied Sciences*, 27(14):2781–2802.
- Buffa, A., Giannelli, C., Morgenstern, P., and Peterseim, D. (2016). Complexity of hierarchical refinement for a class of admissible mesh configurations. *Computer Aided Geometric Design*, 47:83 – 92. SI: New Developments Geometry.
- Buffa, A., Puppi, R., and Vázquez, R. (2020). A Minimal Stabilization Procedure for Isogeometric Methods on Trimmed Geometries. *SIAM Journal on Numerical Analysis*, 58(5):2711–2735.
- Burman, E., Claus, S., Hansbo, P., Larson, M. G., and Massing, A. (2015). CutFEM: Discretizing geometry and partial differential equations. *International Journal for Numerical Methods in Engineering*, 104(7):472–501.
- Burman, E. and Hansbo, P. (2012). Fictitious domain finite element methods using cut elements: II. A stabilized Nitsche method. *Applied Numerical Mathematics*, 62(4):328–341. Third Chilean Workshop on Numerical Analysis of Partial Differential Equations (WONAPDE 2010).

- Carstensen, C. and Funken, S. A. (2001). Averaging technique for FE – a posteriori error control in elasticity. Part I: Conforming FEM. *Computer Methods in Applied Mechanics and Engineering*, 190(18):2483–2498.
- Casquero, H., Liu, L., Zhang, Y., Reali, A., Kiendl, J., and Gómez, H. (2017). Arbitrary-degree T-splines for isogeometric analysis of fully nonlinear Kirchhoff-Love shells. *Computer-Aided Design*, 82:140 – 153. Isogeometric Design and Analysis.
- Casquero, H., Wei, X., Toshniwal, D., Li, A., Hughes, T. J., Kiendl, J., and Zhang, Y. J. (2020). Seamless integration of design and Kirchhoff–Love shell analysis using analysis-suitable unstructured T-splines. *Computer Methods in Applied Mechanics and Engineering*, 360:112765.
- Ciarlet, P. (2002). *The Finite Element Method for Elliptic Problems*. Society for Industrial and Applied Mathematics.
- Cirak, F., Ortiz, M., and Schröder, P. (2000). Subdivision surfaces: a new paradigm for thin-shell finite-element analysis. *International Journal for Numerical Methods in Engineering*, 47(12):2039–2072.
- Clément, P. (1975). Approximation by finite element functions using local regularization. *ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique*, 9(R2):77–84.
- Coradello, L. (2016). Implementation of a high-order Kirchhoff-Love shell: a comparison of IGA and p -FEM. Master’s thesis.
- Coradello, L., Antolin, P., Vázquez, R., and Buffa, A. (2020a). Adaptive isogeometric analysis on two-dimensional trimmed domains based on a hierarchical approach. *Computer Methods in Applied Mechanics and Engineering*, 364:112925.
- Coradello, L., D’Angella, D., Carraturo, M., Kiendl, J., Kollmannsberger, S., Rank, E., and Reali, A. (2020b). Hierarchically refined isogeometric analysis of trimmed shells. *Computational Mechanics*, 66(2):431–447.
- Coradello, L., Kiendl, J., and Buffa, A. (2021a). Coupling of non-conforming trimmed isogeometric Kirchhoff-Love shells via a projected super-penalty approach.
- Coradello, L., Loli, G., and Buffa, A. (2021b). A projected super-penalty method for the C^1 -coupling of multi-patch isogeometric Kirchhoff plates. *Computational Mechanics*.
- Cottrell, J., Reali, A., Bazilevs, Y., and Hughes, T. J. R. (2006). Isogeometric analysis of structural vibrations. *Computer Methods in Applied Mechanics and Engineering*, 195(41):5257–5296. John H. Argyris Memorial Issue. Part II.
- Cottrell, J. A., Hughes, T. J. R., and Bazilevs, Y. (2009). *Isogeometric Analysis*. John Wiley & Sons, Ltd, Chichester, UK.

Bibliography

- Courant, R. (1943). Variational methods for the solution of problems of equilibrium and vibrations. *Bulletin of the American Mathematical Society*, 49(1):1 – 23.
- de Boor, C. (1977). Package for Calculating with B-Splines. *SIAM Journal on Numerical Analysis*, 14(3):441–472.
- de Boor, C. (1978). *A Practical Guide to Splines*. Springer Verlag, New York.
- de Prenter, F. (2019). *Preconditioned iterative solution techniques for immersed finite element methods : with applications in immersed isogeometric analysis for solid and fluid mechanics*. Dissertation, Technische Universiteit Eindhoven.
- de Prenter, F., Lehrenfeld, C., and Massing, A. (2018). A note on the stability parameter in Nitsche’s method for unfitted boundary value problems. *Computers & Mathematics with Applications*, 75(12):4322 – 4336.
- de Prenter, F., Verhoosel, C., van Zwieten, G., and van Brummelen, E. (2017). Condition number analysis and preconditioning of the finite cell method. *Computer Methods in Applied Mechanics and Engineering*, 316:297–327. Special Issue on Isogeometric Analysis: Progress and Challenges.
- de Prenter, F., Verhoosel, C. V., van Brummelen, E. H., Evans, J. A., Messe, C., Benzaken, J., and Maute, K. (2020a). Multigrid solvers for immersed finite element methods and immersed isogeometric analysis. *Computational Mechanics*, 65(3):807–838.
- de Prenter, F., Verhoosel, C. V., van Brummelen, E. H., Evans, J. A., Messe, C., Benzaken, J., and Maute, K. (2020b). Multigrid solvers for immersed finite element methods and immersed isogeometric analysis. *Computational Mechanics*, 65(3):807–838.
- Demkowicz, L., Oden, J., and Strouboulis, T. (1984). Adaptive finite elements for flow problems with moving boundaries. part I: Variational principles and a posteriori estimates. *Computer Methods in Applied Mechanics and Engineering*, 46(2):217–251.
- Deng, J., Chen, F., Li, X., Hu, C., Tong, W., Yang, Z., and Feng, Y. (2008). Polynomial splines over hierarchical T-meshes. *Graphical Models*, 70(4):76–86.
- Dhatt, G. S. (1970). An efficient triangular shell element. *AIAA Journal*, 8(11):2100–2102.
- Diliberto, S. P. and Straus, E. G. (1951). On the approximation of a function of several variables by the sum of functions of fewer variables. *Pacific Journal of Mathematics*, 1(2):195 – 210.
- Dittmann, M., Schuß, S., Wohlmuth, B., and Hesch, C. (2019). Weak C^n coupling for multipatch isogeometric analysis in solid mechanics. *International Journal for Numerical Methods in Engineering*, 118(11):678–699.

- Dittmann, M., Schuß, S., Wohlmuth, B., and Hesch, C. (2020). Crosspoint modification for multi-patch isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 360:112768.
- Dokken, T., Lyche, T., and Pettersen, K. F. (2013). Polynomial splines over locally refined box-partitions. *Computer Aided Geometric Design*, 30(3):331 – 356.
- Dörfler, M. R., Jüttler, B., and Simeon, B. (2010). Adaptive isogeometric analysis by local h-refinement with T-splines. *Computer Methods in Applied Mechanics and Engineering*, 199(5-8):264 – 275. Computational Geometry and Analysis.
- Duong, T. X., Roohbakhshan, F., and Sauer, R. A. (2017). A new rotation-free isogeometric thin shell formulation and a corresponding continuity constraint for patch boundaries. *Computer Methods in Applied Mechanics and Engineering*, 316:43 – 83. Special Issue on Isogeometric Analysis: Progress and Challenges.
- Düster, A., Parvizian, J., Yang, Z., and Rank, E. (2008). The finite cell method for three-dimensional problems of solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, 197(45-48):3768–3782.
- Dörfler, W. (1996). A Convergent Adaptive Algorithm for Poisson’s Equation. *SIAM Journal on Numerical Analysis*, 33(3):1106–1124.
- Elfverson, D., Larson, M. G., and Larsson, K. (2018). CutIGA with basis function removal. *Advanced Modeling and Simulation in Engineering Sciences*, 5(1):6.
- Engquist, B., Tornberg, A.-K., and Tsai, R. (2005). Discretization of Dirac delta functions in level set methods. *Journal of Computational Physics*, 207(1):28–51.
- Farhat, C. and Roux, F.-X. (1991). A method of finite element tearing and interconnecting and its parallel solution algorithm. *International Journal for Numerical Methods in Engineering*, 32(6):1205–1227.
- Farin, G., Hoschek, J., and Kim, M.-S. (2002). *Handbook of Computer Aided Geometric Design*. North Holland & IFIP, 1st edition.
- Felippa, C. (2003). *Introduction to Finite Element Methods*.
- Forsey, D. R. and Bartels, R. H. (1988). Hierarchical B-spline Refinement. In *Proceedings of the 15th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH ’88, pages 205–212, New York, NY, USA. ACM.
- Garau, E. M. and Vázquez, R. (2018). Algorithms for the implementation of adaptive isogeometric methods using hierarchical B-splines. *Applied Numerical Mathematics*, 123:58 – 87.

Bibliography

- Geuzaine, C. and Remacle, J.-F. (2009). Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79(11):1309–1331.
- Giannelli, C., Jüttler, B., Kleiss, S. K., Mantzaflaris, A., Simeon, B., and Špeh, J. (2016). THB-splines: An effective mathematical technology for adaptive refinement in geometric design and isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 299:337–365.
- Giannelli, C., Jüttler, B., and Speleers, H. (2012). THB-splines: The truncated basis for hierarchical splines. *Computer Aided Geometric Design*, 29(7):485–498.
- Giannelli, C., Jüttler, B., and Speleers, H. (2013). Strongly stable bases for adaptively refined multilevel spline spaces. *Adv Comput Math*, 40(2):459–490.
- Gould, P. L. (1999). *Introduction to Linear Elasticity*. Springer-Verlag.
- Greiner, G. and Hormann, K. (1997). Interpolating and Approximating Scattered 3D-data with Hierarchical Tensor Product B-Splines. In *In Surface Fitting and Multiresolution Methods*, pages 163–172. Vanderbilt University Press.
- Grätsch, T. and Bathe, K.-J. (2005). A posteriori error estimation techniques in practical finite element analysis. *Computers & Structures*, 83(4):235–265.
- Grätsch, T. and Bathe, K.-J. (2006). Goal-oriented error estimation in the analysis of fluid flows with structural interactions. *Computer Methods in Applied Mechanics and Engineering*, 195(41):5673–5684. John H. Argyris Memorial Issue. Part II.
- Guo, Y., Heller, J., Hughes, T. J. R., Ruess, M., and Schillinger, D. (2018). Variationally consistent isogeometric analysis of trimmed thin shells at finite deformations, based on the step exchange format. *Computer Methods in Applied Mechanics and Engineering*, 336:39 – 79.
- Guo, Y. and Ruess, M. (2015). Nitsche’s method for a coupling of isogeometric thin shells and blended shell structures. *Computer Methods in Applied Mechanics and Engineering*, 284:881–905. Isogeometric Analysis Special Issue.
- Guo, Y., Zou, Z., and Ruess, M. (2021). Isogeometric multi-patch analyses for mixed thin shells in the framework of non-linear elasticity. *Computer Methods in Applied Mechanics and Engineering*, 380:113771.
- Haslinger, J. and Renard, Y. (2009). A New Fictitious Domain Approach Inspired by the Extended Finite Element Method. *SIAM Journal on Numerical Analysis*, (2):1474–1499.
- Herrema, A. J., Johnson, E. L., Proserpio, D., Wu, M. C., Kiendl, J., and Hsu, M.-C. (2019). Penalty coupling of non-matching isogeometric Kirchhoff-Love shell patches

- with application to composite wind turbine blades. *Computer Methods in Applied Mechanics and Engineering*, 346:810 – 840.
- Hirschler, T., Bouclier, R., Dureisseix, D., Duval, A., Elguedj, T., and Morlier, J. (2019). A dual domain decomposition algorithm for the analysis of non-conforming isogeometric Kirchhoff-Love shells. *Computer Methods in Applied Mechanics and Engineering*, 357:112578.
- Hohmeyer, M. E. (1992). *Robust and Efficient Surface Intersection for Solid Modeling*. PhD thesis, EECS Department, University of California, Berkeley.
- Höllig, K. (2003). *Finite element methods with B-splines*, volume 26 of *Frontiers in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA.
- Höllig, K., Hörner, J., and Hoffacker, A. (2012). Finite element analysis with b-splines: Weighted and isogeometric methods. In Boissonnat, J.-D., Chenin, P., Cohen, A., Gout, C., Lyche, T., Masure, M.-L., and Schumaker, L., editors, *Curves and Surfaces*, pages 330–350, Berlin, Heidelberg. Springer Berlin Heidelberg.
- Holzappel, G. (2001). *Nonlinear Solid Mechanics: A Continuum Approach for Engineering*. John Wiley & Sons, second print edition.
- Horger, T., Reali, A., Wohlmuth, B., and Wunderlich, L. (2019). A hybrid isogeometric approach on multi-patches with applications to Kirchhoff plates and eigenvalue problems. *Computer Methods in Applied Mechanics and Engineering*, 348:396 – 408.
- Hrennikoff, A. (1941). Solution of Problems of Elasticity by the Framework Method. *Journal of Applied Mechanics*, 8(4):A169–A175.
- Hughes, T. J. R. (2000). *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Dover Publications.
- Hughes, T. J. R. (2017). Isogeometric analysis: Progress and challenges. *Computer Methods in Applied Mechanics and Engineering*, 316:1. Special Issue on Isogeometric Analysis: Progress and Challenges.
- Hughes, T. J. R., Cottrell, J. A., and Bazilevs, Y. (2005). Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, 194(39–41):4135–4195.
- Hughes, T. J. R., Reali, A., and Sangalli, G. (2008). Duality and unified analysis of discrete approximations in structural dynamics and wave propagation: Comparison of p-method finite elements with k-method NURBS. *Computer Methods in Applied Mechanics and Engineering*, 197(49):4104–4124.

Bibliography

- ISO 10303-11 (1994). Industrial automation systems and integration – Product data representation and exchange. Standard, International Organization for Standardization, Geneva, CH.
- Johnson, C. (1990). Adaptive finite element methods for diffusion and convection problems. *Computer Methods in Applied Mechanics and Engineering*, 82(1):301–322. Proceedings of the Workshop on Reliability in Computational Mechanics.
- Johnson, C. and Hansbo, P. (1992). Adaptive finite element methods in computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 101(1):143–181.
- Jonkman, J., Butterfield, S., Musial, W., and Scott, G. (2009). Definition of a 5-MW Reference Wind Turbine for Offshore System Development. Technical report.
- Joulaian, M., Hubrich, S., and Düster, A. (2016). Numerical integration of discontinuities on arbitrary domains based on moment fitting. *Computational Mechanics*, 57(6):979–999.
- Kamensky, D., Hsu, M.-C., Schillinger, D., Evans, J. A., Aggarwal, A., Bazilevs, Y., Sacks, M. S., and Hughes, T. J. (2015). An immersogeometric variational framework for fluid–structure interaction: Application to bioprosthetic heart valves. *Computer Methods in Applied Mechanics and Engineering*, 284:1005–1053. Isogeometric Analysis Special Issue.
- Kiendl, J. (2011). *Isogeometric Analysis and Shape Optimal Design of Shell Structures*. Dissertation, Technische Universität München, München.
- Kiendl, J., Ambati, M., Lorenzis, L. D., Gomez, H., and Reali, A. (2016). Phase-field description of brittle fracture in plates and shells. *Computer Methods in Applied Mechanics and Engineering*, 312:374 – 394. Phase Field Approaches to Fracture.
- Kiendl, J., Bazilevs, Y., Hsu, M.-C., Wüchner, R., and Bletzinger, K.-U. (2010). The bending strip method for isogeometric analysis of Kirchhoff-Love shell structures comprised of multiple patches. *Computer Methods in Applied Mechanics and Engineering*, 199(37):2403 – 2416.
- Kiendl, J., Bletzinger, K.-U., Linhard, J., and Wüchner, R. (2009). Isogeometric shell analysis with Kirchhoff-Love elements. *Computer Methods in Applied Mechanics and Engineering*, 198(49):3902 – 3914.
- Kiendl, J., Hsu, M.-C., Wu, M. C., and Reali, A. (2015). Isogeometric Kirchhoff-Love shell formulations for general hyperelastic materials. *Computer Methods in Applied Mechanics and Engineering*, 291:280 – 303.
- Kirchhoff, G. (1850). Über das Gleichgewicht und die Bewegung einer elastischen Scheibe. *Journal für die reine und angewandte Mathematik*, 40:51–88.

- Kraft, R. (1997). Adaptive and linearly independent multilevel B-splines. In *Surface Fitting and Multiresolution Methods*. Vanderbilt University Press.
- Kudela, L., Zander, N., Bog, T., Kollmannsberger, S., and Rank, E. (2015). Efficient and accurate numerical quadrature for immersed boundary methods. *Advanced Modeling and Simulation in Engineering Sciences*, 2(1).
- Kudela, L., Zander, N., Kollmannsberger, S., and Rank, E. (2016). Smart octrees: Accurately integrating discontinuous functions in 3D. *Computer Methods in Applied Mechanics and Engineering*, 306:406–426.
- Laughlin, T. (2020). pyOCCT – Python bindings for OpenCASCADE via pybind11. <https://github.com/trelau/pyOCCT>.
- Lei, Z., Gillot, F., and Jezequel, L. (2015). A C0/G1 multiple patches connection method in isogeometric analysis. *Applied Mathematical Modelling*, 39(15):4405–4420.
- Leidinger, L. (2020). *Explicit Isogeometric B-Rep Analysis for Nonlinear Dynamic Crash Simulations*. Dissertation, Technische Universität München, München.
- Leidinger, L., Breitenberger, M., Bauer, A., Hartmann, S., Wüchner, R., Bletzinger, K.-U., Duddeck, F., and Song, L. (2019). Explicit dynamic isogeometric B-Rep analysis of penalty-coupled trimmed NURBS shells. *Computer Methods in Applied Mechanics and Engineering*, 351:891–927.
- Leonetti, L., Liguori, F. S., Magisano, D., Kiendl, J., Reali, A., and Garcea, G. (2020). A robust penalty coupling of non-matching isogeometric Kirchhoff–Love shell patches in large deformations. *Computer Methods in Applied Mechanics and Engineering*, 371:113289.
- Liu, J. and Marsden, A. L. (2019). A robust and efficient iterative method for hyperelastodynamics with nested block preconditioning. *Journal of Computational Physics*, 383:72 – 93.
- Liu, J., Yang, W., Dong, M., and Marsden, A. L. (2020). The nested block preconditioning technique for the incompressible navier-stokes equations with emphasis on hemodynamic simulations. *Computer Methods in Applied Mechanics and Engineering*, 367:113122.
- Loli, G., Montardini, M., Sangalli, G., and Tani, M. (2019). Space-time Galerkin isogeometric method and efficient solver for parabolic problems.
- Loli, G., Montardini, M., Sangalli, G., and Tani, M. (2020). An efficient solver for space–time isogeometric Galerkin methods for parabolic problems. *Computers & Mathematics with Applications*, 80(11):2586–2603. High-Order Finite Element and Isogeometric Methods 2019.

Bibliography

- Loli, G., Sangalli, G., and Tani, M. (2021). Easy and efficient preconditioning of the isogeometric mass matrix. *Computers & Mathematics with Applications*.
- Love, A. E. H. (1888). The Small Free Vibrations and Deformation of a Thin Elastic Shell. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 179:491–546.
- Mäntylä, M. (1987). *An Introduction to Solid Modeling*. Computer Science Press, Inc., USA.
- Marussig, B., Hiemstra, R., and Hughes, T. J. R. (2018). Improved conditioning of isogeometric analysis matrices for trimmed geometries. *Computer Methods in Applied Mechanics and Engineering*, 334:79–110.
- Marussig, B. and Hughes, T. J. R. (2018). A review of trimming in isogeometric analysis: Challenges, data exchange and simulation aspects. *Archives of Computational Methods in Engineering*, 25(4):1059–1127.
- Marussig, B., Zechner, J., Beer, G., and Fries, T.-P. (2017). Stable isogeometric analysis of trimmed geometries. *Computer Methods in Applied Mechanics and Engineering*, 316:497–521. Special Issue on Isogeometric Analysis: Progress and Challenges.
- Maurin, F., Greco, F., Coox, L., Vandepitte, D., and Desmet, W. (2018). Isogeometric collocation for Kirchhoff-Love plates and shells. *Computer Methods in Applied Mechanics and Engineering*, 329:396 – 420.
- McNeel, R. et al. (2010). Rhinoceros 3D, Version 6.0. *Robert McNeel & Associates, Seattle, WA*.
- Mindlin, R. D. (1951). Influence of Rotatory Inertia and Shear on Flexural Motions of Isotropic, Elastic Plates. *Journal of Applied Mechanics*, 18(1):31–38.
- Montardini, M., Negri, M., Sangalli, G., and Tani, M. (2018a). Space-time least-squares isogeometric method and efficient solver for parabolic problems.
- Montardini, M., Sangalli, G., and Tani, M. (2018b). Robust isogeometric preconditioners for the Stokes system based on the Fast Diagonalization method. *Computer Methods in Applied Mechanics and Engineering*, 338:162 – 185.
- Morin, P., Nochetto, R. H., and Siebert, K. G. (2002). Convergence of Adaptive Finite Element Methods. *SIAM Review*, 44(4):631–658.
- Müller, B., Kummer, F., and Oberlack, M. (2013). Highly accurate surface and volume integration on implicit domains by means of moment-fitting. *International Journal for Numerical Methods in Engineering*, 96(8):512–528.

- Nguyen-Thanh, N., Kiendl, J., Nguyen-Xuan, H., Wüchner, R., Bletzinger, K., Bazilevs, Y., and Rabczuk, T. (2011). Rotation free isogeometric thin shell analysis using PHT-splines. *Computer Methods in Applied Mechanics and Engineering*, 200(47):3410–3424.
- Nguyen-Thanh, N., Valizadeh, N., Nguyen, M., Nguyen-Xuan, H., Zhuang, X., Areias, P., Zi, G., Bazilevs, Y., De Lorenzis, L., and Rabczuk, T. (2015). An extended isogeometric thin shell analysis based on Kirchhoff–Love theory. *Computer Methods in Applied Mechanics and Engineering*, 284:265–291. Isogeometric Analysis Special Issue.
- Nguyen-Thanh, N., Zhou, K., Zhuang, X., Areias, P., Nguyen-Xuan, H., Bazilevs, Y., and Rabczuk, T. (2017). Isogeometric analysis of large-deformation thin shells using RHT-splines for multiple-patch coupling. *Computer Methods in Applied Mechanics and Engineering*, 316:1157–1178. Special Issue on Isogeometric Analysis: Progress and Challenges.
- Niiranen, J., Kiendl, J., Niemi, A. H., and Reali, A. (2017). Isogeometric analysis for sixth-order boundary value problems of gradient-elastic Kirchhoff plates. *Computer Methods in Applied Mechanics and Engineering*, 316:328 – 348. Special Issue on Isogeometric Analysis: Progress and Challenges.
- Nitsche, J. (1971). Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 36(1):9–15.
- Nochetto, R. H., Siebert, K. G., and Veiser, A. (2009). Theory of adaptive finite element methods: An introduction. In DeVore, R. and Kunoth, A., editors, *Multiscale, Nonlinear and Adaptive Approximation*, pages 409–542, Berlin, Heidelberg. Springer Berlin Heidelberg.
- Nochetto, R. H. and Veiser, A. (2012). *Primer of Adaptive Finite Element Methods*, pages 125–225. Springer Berlin Heidelberg, Berlin, Heidelberg.
- Oden, J. and Prudhomme, S. (2001). Goal-oriented error estimation and adaptivity for the finite element method. *Computers & Mathematics with Applications*, 41(5):735–756.
- OpenCASCADE (2018). Open CASCADE technology (version 7.3.0). <http://www.opencascade.com>.
- Parvizian, J., Düster, A., and Rank, E. (2007). Finite cell method. *Computational Mechanics*, 41(1):121–133.
- Pasch, T., Leidinger, L., Apostolatos, A., Wüchner, R., Bletzinger, K.-U., and Duddeck, F. (2021). A priori penalty factor determination for (trimmed) NURBS-based shells with Dirichlet and coupling constraints in isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 377:113688.

Bibliography

- Pegolotti, L., Pfaller, M. R., Marsden, A. L., and Deparis, S. (2021). Model order reduction of flow based on a modular geometrical approximation of blood vessels. *Computer Methods in Applied Mechanics and Engineering*, 380:113762.
- Peters, J. and Reif, U. (2008). *Subdivision Surfaces*. Springer Publishing Company, Incorporated, 1st edition.
- Piegl, L. and Tiller, W. (1995). *The NURBS Book*. Monographs in Visual Communications. Springer Berlin Heidelberg, Berlin, Heidelberg.
- Proserpio, D., Ambati, M., De Lorenzis, L., and Kiendl, J. (2020). A framework for efficient isogeometric computations of phase-field brittle fracture in multipatch shell structures. *Computer Methods in Applied Mechanics and Engineering*, 372:113363.
- Quarteroni, A., Saleri, F., and Veneziani, A. (2000). Factorization methods for the numerical approximation of Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 188(1):505 – 526.
- Quarteroni, A. M. and Valli, A. (2008). *Numerical Approximation of Partial Differential Equations*. Springer Publishing Company, Incorporated, 1st ed. 1994. 2nd printing edition.
- Rafetseder, K. and Zulehner, W. (2019). A new mixed approach to Kirchhoff–Love shells. *Computer Methods in Applied Mechanics and Engineering*, 346:440–455.
- Ramm, E. and Wall, W. A. (2004). Shell structures—a sensitive interrelation between physics and numerics. *International Journal for Numerical Methods in Engineering*, 60(1):381–427.
- Rank, E., Kollmannsberger, S., Sorger, C., and Düster, A. (2011). Shell Finite Cell Method: A high order fictitious domain approach for thin-walled structures. *Computer Methods in Applied Mechanics and Engineering*, 200(45-46):3200–3209.
- Realí, A. and Gómez, H. (2015). An isogeometric collocation approach for Bernoulli-Euler beams and Kirchhoff plates. *Computer Methods in Applied Mechanics and Engineering*, 284:623 – 636. Isogeometric Analysis Special Issue.
- Reddy, J. N. (1999). *Theory and Analysis of Laminated Composite Plates*, pages 1–79. Springer Netherlands, Dordrecht.
- Reddy, J. N. (2006). *Theory and Analysis of Elastic Plates and Shells*. Series in Systems and Control. CRC Press.
- Reissner, E. (1945). The Effect of Transverse Shear Deformation on the Bending of Elastic Plates. *Journal of Applied Mechanics*, 12(2):A69–A77.
- Renner, G. and Weiß, V. (2004). Exact and approximate computation of B-spline curves on surfaces. *Computer-Aided Design*, 36(4):351–362.

- Rogers, D. F. (2001). *An Introduction to NURBS: With Historical Perspective*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
- Saad, Y. (1993). A Flexible Inner-Outer Preconditioned GMRES Algorithm. *SIAM Journal on Scientific Computing*, 14(2):461–469.
- Sangalli, G. and Tani, M. (2016). Isogeometric Preconditioners Based on Fast Solvers for the Sylvester Equation. *SIAM Journal on Scientific Computing*, 38(6):A3644–A3671.
- Schillinger, D., Harari, I., Hsu, M.-C., Kamensky, D., Stoter, S. K., Yu, Y., and Zhao, Y. (2016). The non-symmetric Nitsche method for the parameter-free imposition of weak boundary and coupling conditions in immersed finite elements. *Computer Methods in Applied Mechanics and Engineering*, 309:625 – 652.
- Schillinger, D., Ruess, M., Zander, N., Bazilevs, Y., Düster, A., and Rank, E. (2012). Small and large deformation analysis with the p- and B-spline versions of the Finite Cell Method. *Computational Mechanics*, 50(4):445–478.
- Schuß, S., Dittmann, M., Wohlmuth, B., Klinkel, S., and Hesch, C. (2019). Multi-patch isogeometric analysis for Kirchhoff–Love shell elements. *Computer Methods in Applied Mechanics and Engineering*, 349:91–116.
- Schöllhammer, D., Marussig, B., and Fries, T.-P. (2020). A Consistent Higher-Order Isogeometric Shell Formulation.
- Scott, M., Li, X., Sederberg, T., and Hughes, T. J. R. (2012). Local refinement of analysis-suitable T-splines. *Computer Methods in Applied Mechanics and Engineering*, 213-216:206 – 222.
- Sheppard, W. F. (1899). Central-Difference Formulæ. *Proceedings of the London Mathematical Society*, s1-31(1):449–488.
- Shojaee, S., Izadpanah, E., Valizadeh, N., and Kiendl, J. (2012). Free vibration analysis of thin plates by using a NURBS-based isogeometric approach. *Finite Elements in Analysis and Design*, 61:23–34.
- Simoncini, V. (2016). Computational methods for linear matrix equations. *SIAM Review*, 58(3):377–441.
- Stein, E., Seifert, B., Ohnimus, S., and Carstensen, C. (1994). Adaptive finite element analysis of geometrically non-linear plates and shells, especially buckling. *International Journal for Numerical Methods in Engineering*, 37(15):2631–2655.
- Stricklin, J. A., Haisler, W. E., Tisdale, P. R., and Gunderson, R. (1969). A rapidly converging triangular plate element. *AIAA Journal*, 7(1):180–181.
- Timoshenko, S. and Woinowsky-Krieger, S. (1959). *Theory of plates and shells*. Engineering societies monographs. McGraw-Hill.

Bibliography

- Tornberg, A.-K. and Engquist, B. (2003). Regularization Techniques for Numerical Approximation of PDEs with Singularities. *Journal of Scientific Computing*, 19(1):527–552.
- Tornberg, A.-K. and Engquist, B. (2004). Numerical approximations of singular source terms in differential equations. *Journal of Computational Physics*, 200(2):462 – 488.
- Utku, M. and Carey, G. (1982). Boundary penalty techniques. *Computer Methods in Applied Mechanics and Engineering*, 30(1):103 – 118.
- Vázquez, R. (2016). A new design for the implementation of isogeometric analysis in Octave and Matlab: GeoPDEs 3.0. *Computers and Mathematics with Applications*, 72(3):523 – 554.
- Verfürth, R. (2013). *A Posteriori Error Estimation Techniques for Finite Element Methods*. A Posteriori Error Estimation Techniques for Finite Element Methods. OUP Oxford.
- Vuong, A. V., Giannelli, C., Jüttler, B., and Simeon, B. (2011). A hierarchical approach to adaptive local refinement in isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 200(49–52):3554–3567.
- Wachspress, E. (2013). *The ADI Model Problem*. Springer New York, New York, NY, 1st ed. 2013. edition.
- Waldén, J. (1999). On the approximation of singular source terms in differential equations. *Numerical Methods for Partial Differential Equations*, 15(4):503–520.
- Zienkiewicz, O. C. and Zhu, J. Z. (1987). A simple error estimator and adaptive procedure for practical engineering analysis. *International Journal for Numerical Methods in Engineering*, 24(2):337–357.
- Zienkiewicz, O. C. and Zhu, J. Z. (1992a). The superconvergent patch recovery and a posteriori error estimates. Part 1: The recovery technique. *International Journal for Numerical Methods in Engineering*, 33(7):1331–1364.
- Zienkiewicz, O. C. and Zhu, J. Z. (1992b). The superconvergent patch recovery and a posteriori error estimates. Part 2: Error estimates and adaptivity. *International Journal for Numerical Methods in Engineering*, 33(7):1365–1382.
- Zou, Z., Hughes, T. J. R., Scott, M. A., Sauer, R. A., and Savitha, E. J. (2021). Galerkin formulations of isogeometric shell analysis: Alleviating locking with Greville quadratures and higher-order elements. *Computer Methods in Applied Mechanics and Engineering*, 380:113757.

LUCA CORADELLO

Email: l.coradello@gmail.com

LinkedIn: luca-coradello

Skype: lcoradello



EDUCATION

École Polytechnique Fédérale de Lausanne

Lausanne, CH

Ph.D. in Mechanical Engineering

February 2017–Ongoing (August 2021)

- Thesis: “Accurate isogeometric methods for trimmed shell structures.”

Technical University of Munich

Munich, DE

M.Sc. (honors) in Computational Mechanics, GPA: 1.5 (passed with distinction)

September 2013–March 2016

- Thesis: “Implementation of a high-order Kirchhoff-Love shell: a comparison of IGA and p -FEM.”
- Member of the Bavarian Graduate School of Computational Engineering, part of an initiative of the state of Bavaria to support the education and advancement of highly talented students. Successfully completed the extra 30 ECTS which comprise additional scientific courses, soft skills and a final research project.
- Honors project: “A fictitious material approach to contact mechanics using the Finite Cell Method.”

University of Brescia

Brescia, IT

B.Sc. in Mechanical Engineering, Final Grade: 102/110

September 2009–November 2012

- Thesis: “The Halpin-Tsai equation towards the analytical description of the mechanical behavior of composite materials.”

PROFESSIONAL EXPERIENCE

École Polytechnique Fédérale de Lausanne

Lausanne, CH

PhD Candidate at the chair for Numerical Modelling and Simulation

February 2017–August 2021

- Investigated the beneficial effects of local refinement in the scope of trimmed geometries.
- Developed a reliable and robust *a-posteriori* error estimator tailored to isogeometric Kirchhoff plates and Kirchhoff-Love shells.
- Devised an accurate penalty-like coupling strategies to achieve the required continuity across non-conforming interfaces in the context of Kirchhoff plates and Kirchhoff-Love shells.
- Applied the computational framework to the static shell analysis of several engineering structures, spanning from the B-pillar of a car to a wind turbine blade.
- All methods and functionalities have been implemented and validated on top of the open-source OCTAVE/MATLAB package *GeoPDEs*¹.

Technical University of Munich

Munich, DE

Graduate Research Assistant at the chair for Computation in Engineering

May–December 2016

- Main focus on plasticity (Hencky theory and flow theory) and refinement strategies within the context of high-order discretization and the Finite Cell Method. These functionalities have been implemented into an in-house, object-oriented, high-order Finite Element code written in C++.

¹<http://rafavzqz.github.io/geopdes/>

Technical University of Munich

Student Research Assistant at the chair for Computation in Engineering

Munich, DE

November 2014–April 2016

- Main focus on contact mechanics, non-linear material modeling and shell analysis within the context of p -FEM and isogeometric analysis, combined to the Finite Cell Method. These functionalities have been implemented into an in-house, object-oriented, high-order Finite Element code written in C++.

PUBLICATIONS

1. Antolín P., Buffa A. and **Coradello L.**, “A hierarchical approach to the *a-posteriori* error estimation of isogeometric Kirchhoff plates and Kirchhoff-Love shells”, *Computer Methods in Applied Mechanics and Engineering*, 2020
2. **Coradello L.**, Antolín P., Vázquez R. and Buffa A., “Adaptive isogeometric analysis on two-dimensional trimmed domains based on a hierarchical approach”, *Computer Methods in Applied Mechanics and Engineering*, 2020
3. **Coradello L.**, D’Angella D., Carraturo M., Kiendl J., Kollmannsberger S., Rank E. and Reali A., “Hierarchically refined isogeometric analysis of trimmed shells”, *Computational Mechanics*, 2020
4. **Coradello L.**, Loli G. and Buffa A., “A projected super-penalty method for the C^1 -coupling of multi-patch isogeometric Kirchhoff plates”, *Computational Mechanics*, 2021
5. **Coradello L.**, Kiendl J. and Buffa A., “Coupling of non-conforming trimmed isogeometric Kirchhoff-Love shells via a projected super-penalty approach”, Submitted to *Computer Methods in Applied Mechanics and Engineering*

TEACHING & SUPERVISING

- **Student Teaching Assistant** at TUM
Helped coaching the graduate course Computation in Engineering.
- **Teaching Assistant** at EPFL
Supervised and co-organized several undergraduate and graduate exercise sessions (Analyse Numérique, Numerical Analysis, Numerical Methods for Saddle Point Problems, Numerical Analysis and Computational Mathematics). Additionally, co-supervised a research project on numerical methods for the Stokes equations.

SKILLS

- **Programming:** Matlab, Python, C/C++
- **Machine learning:** TensorFlow
- **Scientific Computing:** Finite Element Method, Isogeometric Analysis, p -FEM, Finite Cell Method
- **Tools/Techs:** LaTeX, Git, ParaView, Rhinoceros
- **Operating systems:** Linux (Ubuntu), macOS

LANGUAGES

- **English:** full professional proficiency
- **Italian:** native speaker
- **German:** working proficiency
- **French:** working proficiency

VOLUNTEERING

- President and Vice-President of the EPFL Chapter of SIAM 2019–Ongoing
Promoting amongst Masters and Ph.D. students the role of mathematics in the industrial world. This includes organizing seminars and *get-together* events with internationally acclaimed speakers.

EXTRACURRICULAR ACTIVITIES

I spend most of my leisure time seeking adventures and exploring different places, generally while running on the trails, hiking in the mountains, back-country skiing or backpacking in a new country.