Pontifícia Universidade Católica
do Rio de Janeiro

Diego Santos Duarte

Topology optimization of plate and shell structures using polygonal finite elements

Tese de Doutorado

Thesis presented to the Programa de Pós–graduação em Engenharia Mecânica, do Departamento de Engenharia Mecânica da PUC-Rio in partial fulfillment of the requirements for the degree of Doutor em Engenharia Mecânica.

Advisor: Prof. Ivan Fábio Mota de Menezes

Rio de Janeiro
September 2025

Pontifícia Universidade Católica
do Rio de Janeiro

Diego Santos Duarte

Topology optimization of plate and shell structures using polygonal finite elements

Thesis presented to the Programa de Pós–graduação em Engenharia Mecânica da PUC-Rio in partial fulfillment of the requirements for the degree of Doutor em Engenharia Mecânica. Approved by the Examination Committee:

Prof. Ivan Fábio Mota de Menezes
Advisor
Departamento de Engenharia Mecânica – PUC-Rio

Profa. Maria Fernanda Figueiredo de Oliveira
Universidade do Estado do Rio de Janeiro – UERJ

Prof. Americo Barbosa da Cunha Junior
Laboratório Nacional de Computação Científica — LNCC

Prof. Rodrigo Bird Burgos
Universidade do Estado do Rio de Janeiro – UERJ

Prof. Anderson Pereira
Departamento de Engenharia Mecânica – PUC-Rio

Rio de Janeiro, September the 25th, 2025

Diego Santos Duarte

Majored in Mechanical Engineering from the Federal University of Rio de Janeiro (UFRJ) and obtained his master's degree in Mechanical Engineering from the Fluminense Federal University (UFF).

Bibliographic data

Duarte, Diego Santos

Topology optimization of plate and shell structures using polygonal finite elements / Diego Santos Duarte; advisor: Ivan Fábio Mota de Menezes. – 2025.

136 f: il. color. ; 30 cm

Tese (doutorado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Mecânica, 2025.

Inclui bibliografia

1. Engenharia Mecânica – Teses. 2. Otimização Topológica. 3. Placa e Casca. 4. Elementos Poligonais. 5. Travamento por Cisalhamento e Membrana. 6. Restrições de Tensão. I. Menezes, Ivan Fábio Mota de. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Mecânica. III. Título.

CDD: 621

To my mother, for her support
and endless love.

Acknowledgments

I would like to thank my advisor, Ivan Menezes, not only for supporting this research with his exceptional knowledge, but also for his empathy during every challenging moment over these four years. He will always be a role model to me of how to be a successful researcher and professor without losing kindness and joy along the way. I am also deeply grateful to Professor Anderson Pereira, who always encouraged me to go further and helped me with some of the codes used in this thesis. It was a true pleasure to have been their student.

I would also like to thank my beloved family: Mana, Sérgio, Gabriela, Talita, Manuela, and Bento. Their unwavering love and support have always fueled my strength to face life's challenges. A heartfelt thank you goes to my longtime friend João, who remained present despite the distance. Throughout this journey, he was almost like a lab mate, patiently listening to and discussing ideas that no one else would willingly do. To Rodrigo, my partner, I am grateful for his daily love and the moments of relief and joy he brought me. I extend special and sincere thanks to all my friends and family members, who surrounded me with enthusiasm and kindness. To my colleagues and students at UFBA, a thank you for the constant support and patience, especially during the times I was fully immersed in this thesis.

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

Abstract

Duarte, Diego Santos; Menezes, Ivan Fábio Mota de (Advisor). **Topology optimization of plate and shell structures using polygonal finite elements**. Rio de Janeiro, 2025. 136p. Tese de Doutorado – Departamento de Engenharia Mecânica, Pontifícia Universidade Católica do Rio de Janeiro.

Plates and shells are thin-walled structures with thicknesses that are small relative to their other dimensions. While plates primarily resist loads applied perpendicular to their mid-plane through bending, shells carry loads mainly via membrane forces along their curved surfaces. Their structural behavior is often studied using numerical methods, notably the Finite Element Method (FEM). Several types of finite elements have been developed, generally with triangular or quadrilateral geometries, for modeling these structures. More recently, polygonal elements have been proposed to provide greater geometric flexibility and mitigate numerical instabilities commonly encountered in topology optimization techniques, such as checkerboard patterns and one-point connection problems. This work presents a topology optimization framework for plate and shell structures using arbitrary polygonal finite elements. The primary motivation is to extend open-source educational software, such as `PolyMesher` and `PolyTop`, to include Reissner-Mindlin plate and shell formulations. The main numerical challenge lies in mitigating shear and/or membrane locking, which can lead to an overestimated stiff response in thin structures. For plates, a locking-free formulation is adopted that applies Timoshenko beam assumptions along element edges; for shells, a degenerated curved element with assumed shear and membrane strain fields is employed. The polygonal elements are validated using benchmark problems, and compliance-based topology optimization is then performed for both structural types. Additionally, this work presents local stress-constrained volume minimization results for plates via the Augmented Lagrangian method. The formulations are robust for both thick and thin structural regimes and also support structured quadrilateral meshes. A guideline for code modification is also provided to extend `PolyTop` for plate analysis, encouraging further research and educational applications of the proposed software.

Keywords

Topology Optimization; Plate and Shell; Polygonal Elements; Shear and Membrane Locking; Stress Constraint.

Resumo

Duarte, Diego Santos; Menezes, Ivan Fábio Mota de. **Otimização topológica de estruturas de placas e cascas usando elementos finitos poligonais**. Rio de Janeiro, 2025. 136p. Tese de Doutorado – Departamento de Engenharia Mecânica, Pontifícia Universidade Católica do Rio de Janeiro.

Placas e cascas são estruturas delgadas, cujas espessuras são pequenas em comparação com suas outras dimensões. Enquanto as placas trabalham predominantemente à flexão, quando submetidas a carregamentos perpendiculares ao seu plano médio, as cascas operam principalmente por meio de esforços de membrana ao longo de sua superfície curva. Frequentemente, o estudo do comportamento estrutural dessas estruturas é realizado por meio de métodos numéricos, notadamente o Método dos Elementos Finitos (*MEF*). Diversos tipos de elementos finitos foram desenvolvidos, geralmente com geometrias triangulares ou quadrilaterais, para a modelagem dessas estruturas. Mais recentemente, elementos poligonais foram propostos com o objetivo de oferecer maior flexibilidade geométrica e evitar instabilidades numéricas comumente observadas na aplicação de técnicas de otimização topológica, como o problema do "tabuleiro de xadrez" e o das conexões pontuais. Este trabalho apresenta uma metodologia para otimização topológica de estruturas de placas e cascas utilizando elementos finitos poligonais arbitrários. A principal motivação é a extensão de softwares educacionais de código aberto, como o `PolyMesher` e o `PolyTop`, para inclusão das formulações de placas e cascas de *Reissner-Mindlin*. O principal desafio numérico é a mitigação do travamento por cisalhamento e/ou por esforços de membrana, que pode levar à superestimação da rigidez em estruturas finas. Para placas, é utilizada uma formulação livre de travamento que aplica as hipóteses de viga de *Timoshenko* ao longo das bordas dos elementos; para cascas, emprega-se um elemento curvo degenerado com campos assumidos de deformação por cisalhamento e membrana. Os elementos poligonais são validados por meio de exemplos representativos da literatura e a otimização topológica visando à diminuição da complacência estrutural é realizada para ambas as estruturas. Além disso, são apresentados resultados de minimização de volume com restrições locais de tensão para placas, utilizando o método do Lagrangiano aumentado. As formulações são robustas tanto para estruturas espessas quanto finas e também suportam malhas quadrilaterais estruturadas. Um guia para modificação do código também é fornecido para estender o `PolyTop` para a análise de placas, incentivando pesquisas futuras e aplicações educacionais do *software* proposto.

Palavras-chave

Otimização Topológica; Placa e Casca; Elementos Poligonais; Travamento por Cisalhamento e Membrana; Restrições de Tensão.

Table of contents

List of figures

List of tables

List of Abreviations

AF – Amplification Factor

AL – Augmented Lagrangian

AMS – Assumed Membrane Strains

ANS – Assumed Natural Strains

ASS – Assumed Shear Strains

BESO – Bi-directional Evolutionary Structural Optimization

DOF – Degree of Freedom

DKQ – Discrete Kirchhoff Quadrilateral

ESIFA – Evolutionary Swarm Intelligence Firefly Algorithms

FE – Finite Element

FEM – Finite Element Method

FSDT – First-order Shear Deformation plate Theory

HDF – Hybrid Displacement Function

MITC – Mixed Interpolation of Tensorial Components

MMA – Method of Moving Asymptotes

NURBS – Non-Uniform Rational B-Splines

OC – Optimality Criteria

RAMP – Rational Approximation of Material Properties

SIMP – Solid Isotropic Material with Penalization

SRI – Selective Reduced Integration

SRI HC – Selective Reduced Integration with Hourglass Control

TO – Topology Optimization

VEM – Virtual Element Method

List of Symbols

ρ – density parameter

ε – Ersatz parameter

E – Young's modulus

w – transversal displacement

$\boldsymbol{\theta}$ – plate rotation vector

G – shear modulus

κ^s – shear correction factor

ν – Poisson ratio

$\mathbf{D}^b$ – plate bending constitutive tensor

$\mathbf{D}^s$ – plate shear constitutive tensor

h – plate/shell thickness

$\boldsymbol{\kappa}$ – plate bending strain

$\boldsymbol{\kappa}_V$ – plate bending strain vector in Voigt notation

$\boldsymbol{\gamma}$ – plate shear strain

$\mathbf{u}$ – displacement vector

ϕ_i – shape functions

$\mathbf{d}^e$ – nodal displacement vector of an element e

n – number of edges in a polygonal element

$\mathbf{B}$ – FE strain-displacement matrix

$\mathbf{K}$ – FE stiffness matrix

$\mathbf{U}$ – FE displacement vector

$\mathbf{F}$ – FE force vector

Π_{TPE} – total potential energy

$\boldsymbol{\sigma}$ – Cauchy stress vector

$\mathbf{D}$ – constitutive tensor

σ_e^v – *von Mises* stress at the centroid of an element e

$\mathbf{y}$ – filtered density vector

$\mathbf{z}$ – design variable vector

$\mathbf{P}$ – filtering matrix

p – SIMP penalization factor

$|\Omega_e|$ – element volume

$\overline{V}$ – volume constraint value

σ_{lim} – stress constraint value

g_j – constraint functions

$J_{\mu^{(k)}}$ – AL function at a kth step

$\mathbf{g}_i$ – covariant basis vectors

$\mathbf{g}^i$ – contravariant basis vectors

$\mathbf{T}_\varepsilon$ – stress-strain transformation matrix

$\mathbf{J}$ – Jacobian matrix

e_{ij} – Green-Lagrange covariant strain components

δ_j^i – Kronecker delta

$\mathbf{\Psi}$ – shell surface parameterization function

I like "OR"s. Doubt is reassuring.

Craig Thompson, *Blankets.*

1
Introduction

Thin-walled structures appear in numerous engineering applications, such as lightweight aircraft and vehicle fuselages, pressure vessels and pipelines, flat and curved roofs, vibration actuators, and so on. The main difference between these structures and a generic solid one is their relatively small thickness, when compared to other body dimensions. These structures are technically named plates if their geometries are flat or shells, in case of general three-dimensional configuration, comprising both flat and curved surfaces. Additionally, plates are restricted to transversal forces and/or bending loads, while shells may have loads associated with all degrees of freedom. In this work, our aim is to explore, contribute, and share some developments in the field of numerical simulation and design of plate and shell structures.

1.1
Structural Optimization

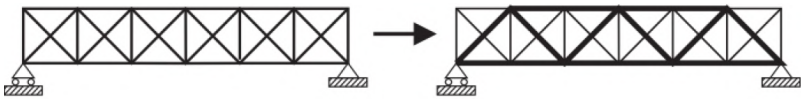
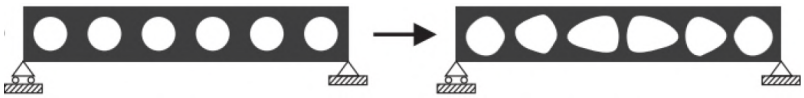
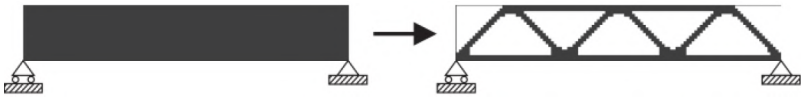
Optimization problems aim to typically minimize functions, called objective or cost functions, under possible constraints. Structural optimization uses parameters as the function domain in which at least one of these domain parameters will be related to the structure of the studied body. In cases where there is more than one objective, the optimization will have a multi-objective function that can translate the relevance of each parameter in the expected result, e.g., through weighted computations [1, 2, 3].

There are mainly three types of structural optimization [1]:

- Sizing Optimization: also known as parametric optimization, this type is limited to the optimization of body dimensions, such as the width or thickness of a beam with a predefined profile;
- Shape Optimization: this class of optimization aims to minimize the cost function through the shape of the body and, therefore, allowing changes on its boundaries;
- Topology Optimization: it can change the entire domain to obtain the optimum material distribution, that is, topological perturbations (material inclusions or removals) can be executed both on the boundary and inside the domain.

Optimization methodologies often combine the types mentioned simultaneously. Table 1.1 illustrates how the three classes of structural optimization would work for a simply-supported body example.

Table 1.1: Structural optimization types and examples. Adapted from [1].

Sizing	
Shape	
Topology	

1.1.1 Topology Optimization

Topology optimization is a widely known application that can use Finite Element Method (FEM) in order to compute objective functions, sensitivities and constraints. In general, a topology optimization problem aims to find a material distribution for a certain body minimizing an associated cost function. Hence, applications include the basic minimization of mass, while maintaining structural performance (the well-known structural compliance problem); the synthesis of complex compliant mechanisms; fluid-structure interaction; natural frequencies and modal analysis; and so on. There are many established and emerging methods in the topology optimization literature, but some of the most recurrent are the SIMP (Solid Isotropic Material with Penalization) [1], BESO (Bi-directional Evolutionary Structural Optimization) [4] and topological derivative [5, 3] methods.

In the paper disclosing the `PolyTop` software [6], the combination of the SIMP topology optimization method [1] with an arbitrary polygonal mesh from `PolyMesher` [7] is proposed for membrane finite elements through an educational code. The SIMP method maps the material domain through a density parameter $\rho(\mathbf{x})$ bounded by lower and upper limits ($\underline{\rho} \leq \rho(\mathbf{x}) \leq \overline{\rho}$). Density values mimic void to full-material regions, comprising intermediate regions (gray scales) – whose physical interpretation is a frequent subject of study [8]. In order to regularize the resultant topology into a well-defined black-and-white pattern, a penalty p is introduced in the SIMP method, in which

an interpolation is proposed based on the Ersatz material model as following

$$E(\mathbf{x}) = [\varepsilon + (1-\varepsilon)\rho(\mathbf{x})^p]\, E_0(\mathbf{x}) \,, \tag{1-1}$$

where E_0 is the solid material Young's modulus and $\varepsilon << 1$ is the Ersatz parameter. For plane stress bi-dimensional problems, the value $p = 3$ is usually adopted to achieve well-defined topologies, as shown in the following equation developed in [9]

$$p \geq \max\left\{\frac{2}{1-\nu_0}, \frac{4}{1+\nu_0}\right\} \,, \tag{1-2}$$

where ν_0 is the material Poisson ratio, which is often taken as 1/3 for engineering purposes (steel properties).

Finally, a sensitivity filtering scheme is an additional strategy to overcome the lack of consistent topology optimization solutions and numerical instabilities, such as the checkerboard pattern [1]. In this scheme, a filtering weight is computed based on the distances between element centroids and an input filtering radius.

1.2 Motivation and relevance

Linear elastic plate and shell structures are often described via Reissner–Mindlin, for thick structures, or Kirchhoff–Love theories, for the thin cases. There has been a trend in the literature to tackle both thickness cases in a finite-element robust algorithm based on a single theory. However, the well-known shear-locking phenomenon commonly appears in many Reissner-Mindlin plate and shell finite elements, overestimating their stiffness when the span-to-thickness ratio (L/h) increases. For traditional quadrilateral meshes, integration techniques are popular solutions. However, it is known that if one tries to apply these methods for arbitrary polygonal meshes, the locking prevention fails since the shear constraints will not vanish. These types of mesh, as the one disclosed by Talischi et al. [7] for the educational software `PolyMesher`, can provide an organic and enhanced distribution of elements even in complex real-world geometries, tracking a standardization – or even customization – of the element average sizes. In this work, we refer to arbitrary polygonal meshes as unstructured meshes composed of convex n-gons, which are not necessarily regular, with n varying throughout the domain. In other words, triangles, quadrilaterals, pentagons, hexagons, and so on are possible to be seen in such meshes.

These meshes have emerged in several contexts using numerical methods. For membrane elements, there exist uncountable applications and educational codes available for both arbitrary polygonal and regular meshes (linear quadri-

lateral and triangular). In the field of topology optimization, polygonal meshes generally prevent some numerical instabilities as the checkerboard pattern or hinged connections, once one-node element interfaces are rare in this kind of mesh – see the educational software `PolyTop` [6] that is built with `PolyMesher` [7]. After `PolyTop`, many other contributions were developed in related areas using this code as a basis, such as for fluid flow [10], multi-material bodies [11], dynamic loads [12], stress-constrained optimization [13], and plasticity [14], showing advantageous evidence in disclosing accessible codes. All of these studies utilize membrane or three-dimensional solid finite elements.

The present work is mainly motivated to add up to these contributions by expanding the coverage of the polygonal software for both plate and shell finite elements. The relevance of developing such simulation and topology optimization software is based on the proper approach to thin-walled structures, since plate and shell finite elements perform better, in some cases computationally cheaper, than the three-dimensional solid element [15]. Although polygonal plate finite elements have been launched, we could not find open codes available in the literature. Meanwhile, polygonal shell finite element should be treated as curved elements, whose development can be much more complex than flat elements. Hence, even though shear and membrane locking have already been treated for polygonal elements, open codes are still absent. Along the next Chapters of this thesis, literature reviews driven towards each specific field will shed a light on the state-of-the-art gaps, raising additional aspects regarding the novelty and relevance of the present work. In summary, the finite element and topology optimization developments of plates and shells using polygonal meshes are somehow recent and open in the literature.

1.3 Thesis contributions

There are two major contributions provided by this thesis: one regarding polygonal plate elements and the other, polygonal shell elements, both using the `PolyTop` software [6] as a base MATLAB® code. As already mentioned, although the Reissner-Mindlin theory (thick structures) is used to formulate plates and shell elements, the hereinafter contributions attempt to follow the literature trend in making codes robust to either thick or thin plates. In these major contributions, we list below related achievements in face of the state-of-the-art:

– `PolyTop` versions for both plates and shells are developed. Each software performs compliance minimization, properly circumventing shear and/or membrane locking, for any thickness case with no adjustments required.

– The `PolyTop` version for plates is shared with the academic community in a conference paper [16], while the shell version will be released in due course as the final derivations of this thesis. The `PolyTop` version for plates is achieved with a very simple sequence of steps to change the original `PolyTop` free software. These steps are described in Appendix A.

– A recent development for local stress-constrained topology optimization via the Augmented Lagrangian method [13], an aggregation-free process, is also adapted and tested for the first time in the literature to the context of plate structures – see [17]. This code is also suitable for both plate thickness cases and comprises either regular quadrilateral or polygonal meshes. This software can be generated following straightforward instructions to adjust the original `PolyStress` code [13]. The procedure is quite the same as described in Appendix A, but utilizing the `PolyStress` code.

– The `PolyTop` version for shells unveils the topology optimization made with a mesh fully composed of arbitrary polygonal elements – current literature only applies polygonal elements on the boundaries of the shell topology.

1.4 Thesis Outline

The core of this thesis is organized in the following five Chapters, divided in two main parts: plates and shells. For each of these parts, there will be a Chapter for the finite element and locking treatment methods and another for the topology optimization itself. The literature review is fragmented along the thesis, where each Chapter begins with the respective state-of-the-art discussion.

Therefore, in Chapter 2, the plate finite element and shear locking methods will be evaluated, modeled, and tested. The resulting element will be used in Chapter 3 for the topology optimization of plate structures. The same framework applies for shell structures, in which Chapter 4 stands for the finite element and locking treatment developments, while Chapter 5 proceeds with the topology optimization of shell structures. Then, we conclude by discussing the main contributions, advantages and drawbacks, and suggestions for future improvements of the present study in Chapter 6.

2
Finite Element Method for plate structures

2.1
Literature review

Considering the plate finite element, the aforementioned conventional quadrilaterals and triangles have been widely studied along with many finite element method applications – which includes the topology optimization. For any general plate finite element, the three nodal degrees of freedom are stated as the rotations around x and y axes plus a deflection in z axis. The phenomenon of shear locking is encountered in elements derived from the Reissner-Mindlin plate theory, also referred to as the "First-order Shear Deformation plate Theory" (FSDT). Although Reissner-Mindlin represents a thick model, there is an effort to build robust tools that encompasses both thickness cases. Such cases are generally categorized by a ratio between a characteristic length (L) of the plate domain and its thickness (h), for which a suggested practice is to take $L/h \geq 100$ for thin case and $L/h \leq 10$ for thick case [18]. Reissner-Mindlin theory is a widely used approach, also implemented in commercial finite element packages, such as ABAQUS® and ANSYS® [19]. This can be attributed to the fact the FSDT requires only $\mathcal{C}^0$ functions to represent the displacement and rotational fields, while Kirrchoff-Love models require the $\mathcal{C}^1$ type (i.e., besides fields, their first derivatives also need to be continuous).

Shear locking is an underestimated displacement response of the plate when decreasing its thickness, and it arises from the integration of the shear terms. In other words, as the plate becomes thinner, the plate elements based on the FSDT fails to satisfy the Kirchhoff constraint, i.e., $\boldsymbol{\nabla} w - \boldsymbol{\beta} = \mathbf{0}$ (where w is the transverse displacement and β is the rotation) [19]. That is, for thin plates, shear effects should be insignificant (Kirchhoff-Love assumption) and, therefore, shear locking is the abnormal behavior when these effects are overestimated.

A concern commonly addressed mutually with locking phenomenon is the occurrence of the so-called spurious zero-energy (hourglass) modes, which represent instabilities in the element to properly represent rigid-body motions.

In practice, the spurious modes can be detected with element tests. For example, the ellipticity condition test evaluates the zero eigenvalues of the stiffness matrix of one unsupported element model. If the number of eigenvalues exceeds the physical rigid-body modes, spurious modes are detected [20, 21].

2.1.1 Locking-Free Techniques for Quadrilateral Plate Elements

Figure 2.1 summarizes some classical locking-free techniques for quadrilateral elements. The selective integration technique (S) is a very common way to solve shear locking in linear quadrilateral meshes, due to its simplicity. It consists in reducing the order of the Gaussian quadrature numerical integration from 2×2 points to one point at the element center only in the shear terms, while maintaining the full integration in the bending terms of the equations. Reduced integration technique (R) reduces the integration order on both shear and bending terms, while the full integration (F) stands for the 2×2 points traditional method [22, 23]. It is evident from Fig. 2.1 that the selective integration presents the best performance comparing to the latter ones for quadrilateral elements. Heterosis [24] and second-order elements have effective locking-free performance, but requires more computational effort than the bilinear elements, due to their additional side nodes (strong drawback in nonlinear and dynamic applications). For a more detailed discussion about these techniques, see [15, 25].

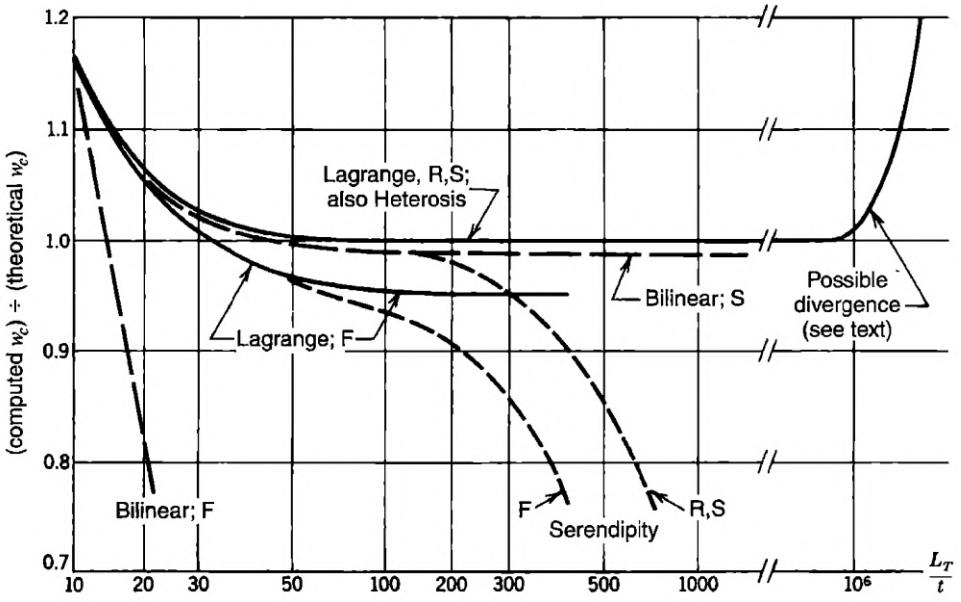


Figure 2.1: Center deflection w for shear locking evaluation – a uniformly loaded square plate of side length L, thickness t, with all edges clamped, modeled by an 8-by-8 mesh. Taken from [15].

Another particularly effective solution that has emerged within the finite element community to address the challenges associated with shear locking in Reissner-Mindlin plate elements is the MITC (Mixed Interpolation of Tensorial Components) technique. Introduced by Brezzi et al. [26], this methodology has enabled the construction of numerous plate element formulations, such as the well-known MITC3 and MITC4 elements, that successfully avoid locking-related issues.

2.1.2 Locking-Free Techniques for Arbitrary Polygonal Plate Elements

Although several contributions were proposed to deal with shear locking on conventional meshes, none of the simplest classical techniques, such as the reduced or selective integration, solve the issue on arbitrary polygonal meshes. If one tries to implement it for polygonal elements, this will not perform with the same effect as in quadrilateral element meshes. This is due to the fact that the intention of reducing the integration order when computing the shear terms is to also reduce the impact of imposed shear constraints in the stiffness matrix [15]. However, when dealing with polygonal elements, we have much more Gaussian points per element compared to the linear quadrilateral ones. Therefore, even when the integration order is reduced, the shear constraints persist.

Consequently, over the past years, some methods started emerging to circumvent locking in polygonal meshes. A pathbreaking technique was proposed by Nguyen-Xuan [27] based on a generalization of an assumed strain field imposed over the polygonal edges under the Timoshenko beam assumption [28], which naturally solves the shear locking drawback. In his work, many shape functions were tested, such as Wachspress [29], mean-value [30] and Laplace [31] functions, but a piecewise-linear function was proposed. This method has also been applied to laminated composite plates [32] and an improvement for geometrically nonlinear functionally-graded porous plates was proposed by Nguyen et al. [33].

Later, Videla et al. [19] utilized an approach based on the discrete Kirchhoff Mindlin theory and the assumed shear strain fields with Wachspress shape functions [29]. Meanwhile, Katili et al. [34] proposed a polygonal locking-free plate element for smoothed finite element method, while Wu et al. [35] addressed the problem for polygonal Hybrid Displacement-Function (HDF) element method. More recently, Nguyen et al. [36] adapted the first proposed technique [27] by imposing a factor α (whose appropriate value was chosen to be 0.5) onto the assumed rotations and shear strains. Right after, Nguyen and

Phan [37] proposed a readjustment on the methods by Nguyen-Xuan [27] and Nguyen et al. [33] to include a selective element domain interpolation.

Additionally, the Virtual Element Method (VEM) utilizing polygonal meshes has emerged as an approach for analyzing Reissner-Mindlin plates [38, 39, 40]. Inspired by the formulation of MITC elements, this method offers distinct advantages over the traditional FEM, such as its ability to manage hanging nodes in non-conforming meshes and its robustness to high mesh distortion. However, VEM is also pointed by some authors to present an increased mathematical complexity and higher computational expenses [41]. Table 2.1 summarizes the aforementioned techniques, briefly providing the distinguishing details.

Table 2.1: Locking-free techniques for arbitrary polygonal plate elements.

Designation by authors	**Approximation Method**	**Shape Functions**	**Year**
PRMn [27]	FEM	Piecewise-Linear	2017
(Not specified) [33]	FEM	Wachspress + Serendipity	2018
(Not specified) [38]	VEM	-	2018
DKM-ngon [19]	FEM	Wachspress	2019
SDKMn-gon [34]	Smoothed FEM	Wachspress + Serendipity	2019
HDF-PE [35]	HDF	Bilinear + Quadratic-Linear	2021
αARS-Poly [36]	FEM	Piecewise-Linear	2023
SI-ARS-Poly [37]	FEM	Piecewise-Linear	2023

In the following Sections, we will present the problem statement, the finite element formulation, and the locking-free treatment, which is inspired by Nguyen-Xuan [27], with the Wachspress shape functions as originally utilized in `PolyTop` [6].

2.2 Reissner-Mindlin plate modeling

Let $\Omega \subset \mathbb{R}^2$ be a bounded domain defined as the midplane of an isotropic Reissner-Mindlin plate. The governing equations for this model are (using Voigt notation) [27]

$$\begin{aligned} \boldsymbol{\nabla} \cdot \mathbf{D}^{\mathbf{b}} \boldsymbol{\kappa}(\boldsymbol{\theta}) + Gh\boldsymbol{\gamma} &= \mathbf{0} \qquad \text{in } \Omega, \\ Gh \boldsymbol{\nabla} \cdot \boldsymbol{\gamma} &= p \qquad \text{in } \Omega, \\ \mathrm{w} = \overline{\mathrm{w}}, \quad \boldsymbol{\theta} &= \overline{\boldsymbol{\theta}} \qquad \text{on } \partial\Omega, \end{aligned} \tag{2-1}$$

where w and $\boldsymbol{\theta}^T = [\theta_x, \theta_y]$ are the transversal displacement and the rotations about y and x axes, respectively (see Fig. 2.2). The plate is under the transversal load $p(x, y)$ per unit area, $G = \frac{\kappa^s E}{2(1+\nu)}$ is the shear modulus, with $\kappa^s = 5/6$ being the shear correction factor, E is the Young's modulus and ν is the Poisson ratio. The bending ($\mathbf{D}^{\mathbf{b}}$) and shear ($\mathbf{D}^{\mathbf{s}}$) matrices are defined as

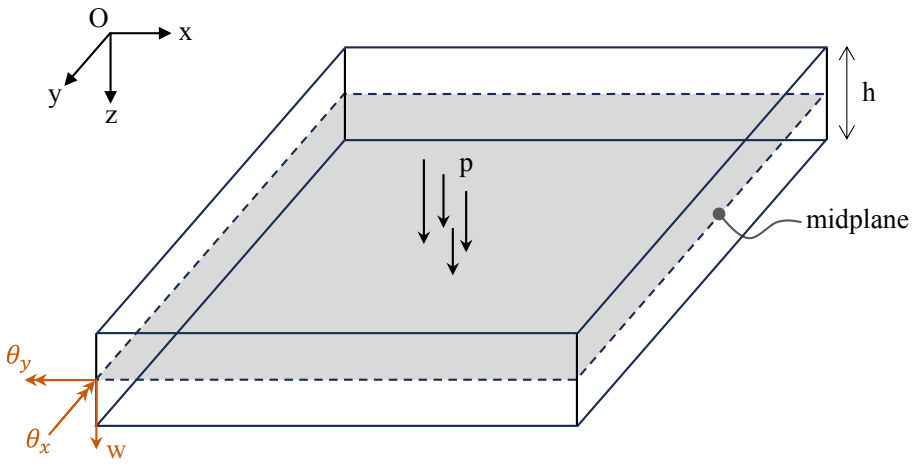


Figure 2.2: Reissner-Mindlin plate model.

$$\mathbf{D}^{\mathbf{b}} = \frac{Eh^3}{12(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & (1-\nu)/2 \end{bmatrix}; \quad \mathbf{D}^{\mathbf{s}} = Gh \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{2-2}$$

where h is the plate thickness. The bending ($\boldsymbol{\kappa}$) and shear ($\boldsymbol{\gamma}$) strains are

$$\boldsymbol{\kappa} = \frac{1}{2}\left(\boldsymbol{\nabla}\boldsymbol{\theta} + \boldsymbol{\nabla}\boldsymbol{\theta}^T\right); \qquad \boldsymbol{\gamma} = \boldsymbol{\nabla} w - \boldsymbol{\theta}, \tag{2-3}$$

where $\boldsymbol{\nabla} = [\partial/\partial x, \partial/\partial y]^T$ is the gradient operator. Let us define V and V_0 as

$$\begin{aligned} V &= \{[\mathrm{w}, \boldsymbol{\theta}] : \mathrm{w} \in H^1(\Omega), \boldsymbol{\theta} \in H^1(\Omega^2); \mathrm{w} = \overline{\mathrm{w}}, \boldsymbol{\theta} = \overline{\boldsymbol{\theta}} \text{ on } \partial\Omega\}, \\ V_0 &= \{[\delta\mathrm{w}, \delta\boldsymbol{\theta}] \in V; \delta\mathrm{w} = 0, \delta\boldsymbol{\theta} = \mathbf{0} \text{ on } \partial\Omega\}, \end{aligned} \tag{2-4}$$

where $H^1(\Omega)$ is a Hilbert space that is defined for a displacement field $[\delta \mathrm{w}, \delta\boldsymbol{\theta}]$. We also define $\mathbf{u}^T = (\mathrm{w}, \boldsymbol{\theta})$ as the displacement vector of any point in the domain.

2.2.1 Finite element discretization

The discrete weak form of Eq. 2-1 can be obtained as: find a discrete solution $(\mathrm{w}^h, \boldsymbol{\theta}^h) \in V^h$, such that

$$\int_\Omega \delta\boldsymbol{\kappa}^{h^T} \mathbf{D}^{\mathbf{b}} \boldsymbol{\kappa}^h \, d\Omega + \int_\Omega \delta\boldsymbol{\gamma}^{h^T} \mathbf{D}^{\mathbf{s}} \boldsymbol{\gamma}^h \, d\Omega = \int_\Omega \delta \mathrm{w} p \, d\Omega, \quad \forall (\delta \mathrm{w}^h, \delta\boldsymbol{\theta}^h) \in V_0^h, \tag{2-5}$$

in which $V^h \subset V$ and $V_0^h \subset V_0$ are the finite element approximation spaces.

The displacement field within each finite element can be stated as

$$\mathbf{u}^{\mathbf{e}}(x, y) = \sum_i \boldsymbol{\phi}_{\boldsymbol{i}}^{\boldsymbol{e}}(x, y) \mathbf{d}_{\mathbf{i}}^{\mathbf{e}}, \tag{2-6}$$

where i is the index for the number of polygonal vertices, $\mathbf{u}^{\mathbf{e}}(x, y)$ is the displacement vector at (x, y), $\boldsymbol{\phi}_{\boldsymbol{i}}^{\boldsymbol{e}}(x, y)$ are the shape functions and $\mathbf{d}_{\mathbf{i}}^{\mathbf{e}} = \left[\mathrm{w}_i^e, \theta_{xi}^e, \theta_{yi}^e\right]^T$ is the i^{th} nodal displacement vector of element e.[†] Therefore, we will be able to write

$$\boldsymbol{\kappa}^e = \sum_i \mathbf{B}_{\mathbf{i}}^{\mathbf{b,e}} \mathbf{d}_{\mathbf{i}}^{\mathbf{e}}; \qquad \boldsymbol{\gamma}^e = \sum_i \mathbf{B}_{\mathbf{i}}^{\mathbf{s,e}} \mathbf{d}_{\mathbf{i}}^{\mathbf{e}}, \tag{2-7}$$

where

$$\mathbf{B}_{\mathbf{i}}^{\mathbf{b,e}} = \begin{bmatrix} 0 & \frac{\partial \phi_i}{\partial x} & 0 \\ 0 & 0 & \frac{\partial \phi_i}{\partial y} \\ 0 & \frac{\partial \phi_i}{\partial y} & \frac{\partial \phi_i}{\partial x} \end{bmatrix}; \qquad \mathbf{B}_{\mathbf{i}}^{\mathbf{s,e}} = \begin{bmatrix} \frac{\partial \phi_i}{\partial x} & \phi_i & 0 \\ \frac{\partial \phi_i}{\partial y} & 0 & \phi_i \end{bmatrix}. \tag{2-8}$$

Finally, we have

$$\mathbf{K_0 U} = \mathbf{F}, \tag{2-9}$$

where

$$\begin{aligned} \mathbf{K_0} &= \sum_{e=1}^{n_e} \left[\int_{\Omega^e} \mathbf{B}^{\mathbf{b,e}^T} \mathbf{D}^{\mathbf{b}} \mathbf{B}^{\mathbf{b,e}} \, d\Omega^e + \int_{\Omega^e} \mathbf{B}^{\mathbf{s,e}^T} \mathbf{D}^{\mathbf{s}} \mathbf{B}^{\mathbf{s,e}} \, d\Omega^e \right], \\ \mathbf{F} &= \sum_{e=1}^{n_e} \int_{\Omega^e} p \boldsymbol{\phi}^{\boldsymbol{e}} \, d\Omega^e, \end{aligned} \tag{2-10}$$

where $\sum_{e=1}^{n_e}$ denotes the assembly procedure.

[†]The Finite Element Method for arbitrary polygonal elements, including the numerical integration strategy, shape functions, derivatives, and other relevant details, is described in the `PolyMesher` [7] and `PolyTop` [6] papers.

2.2.2 Locking-free technique

The locking-free technique proposed by Nguyen-Xuan [27] is utilized in this work, except for the shape function. As the `PolyTop` (or `PolyStress`) software is used as the basis code in this study, the shape functions remain unchanged for simplicity, i.e., the Wachspress functions are retained. Nguyen-Xuan [27] proposed a piecewise-linear shape function that led to the development of the so-called PRMn-PL element and evaluated its performance against various other functions for arbitrary polygonal elements, including Wachspress [29] (PRMn-W), mean-value [30] (PRMn-M), and Laplace [31] (PRMn-L) functions. Notably, he demonstrated that all PRMn elements successfully passed patch tests, with particular emphasis on the performances of PRMn-PL and PRMn-W. While PRMn-PL was reported to have certain advantages over PRMn-W, the differences were generally minor and are unlikely to consistently affect optimization results.

2.2.2.1 Original idea

The original idea applies Timoshenko beam formulation to each edge of a triangular element [28]. The formulation is summarized in Fig. 2.3. The deflection w(ξ), rotation $\beta(\xi)$ and shear $\gamma(\xi)$ strain fields are taken as the following cubic, quadratic and constant functions, respectively

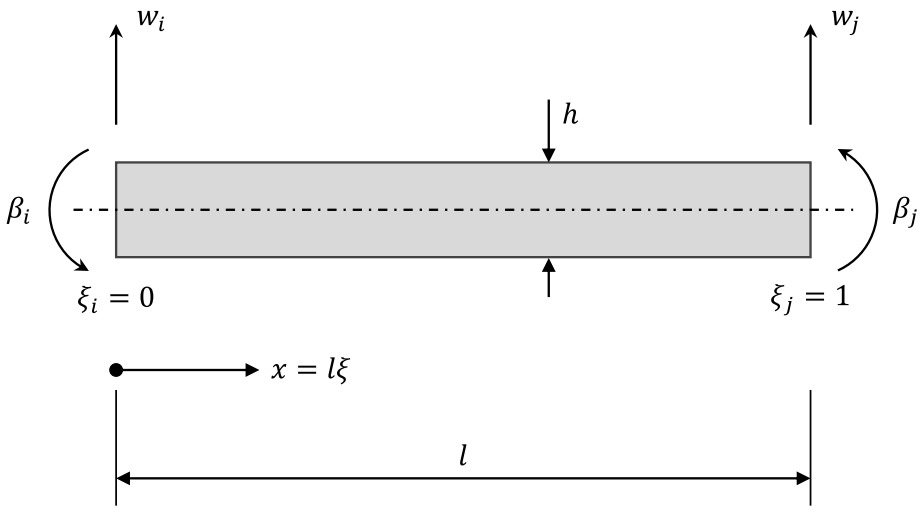


Figure 2.3: The Timoshenko's beam element and corresponding degrees of freedom.

$$\begin{cases} \mathrm{w}(\xi) = \mathrm{w}_i(1-\xi) + \mathrm{w}_j\xi + \alpha_1 l\xi(1-\xi) + \alpha_2 l\xi(1-\xi)(1-2\xi) & \text{on } \Gamma^\xi, \\ \beta(\xi) = \beta_i(1-\xi) + \beta_j\xi + \alpha_3\xi(1-\xi) & \text{on } \Gamma^\xi, \\ \gamma(\xi) = \gamma_0 & \text{on } \Gamma^\xi, \\ \mathrm{w} = \mathrm{w}_i \text{ and } \beta = \beta_i & \text{at } \xi = \xi_i = 0, \\ \mathrm{w} = \mathrm{w}_j \text{ and } \beta = \beta_j & \text{at } \xi = \xi_j = 1; \end{cases} \tag{2-11}$$

where $\alpha_1(\mathrm{w},\beta)$, $\alpha_2(\mathrm{w},\beta)$, $\alpha_3(\mathrm{w},\beta)$, and $\gamma_0(\mathrm{w},\beta)$ are unknown. Substituting Eq. 2-11 into the shear strain of the Timoshenko beam [28], we obtain

$$\begin{aligned} \gamma =& \gamma_0 = \frac{1}{l}\frac{d\mathrm{w}(\xi)}{d\xi} + \beta(\xi) \\ &= \left(\frac{-\mathrm{w}_i + \mathrm{w}_j}{l} + \alpha_1 + \alpha_2 + \beta_i\right) + \left(-2\alpha_1 - 6\alpha_2 - \beta_i + \beta_j + \alpha_3\right)\xi \\ &\quad + \left(6\alpha_2 - \alpha_3\right)\xi^2. \end{aligned} \tag{2-12}$$

Therefore

$$\alpha_1 = \frac{1}{2}(-\beta_i + \beta_j), \quad \alpha_2 = \gamma_0 - \frac{1}{2}\wp, \quad \alpha_3 = 6\left(\gamma_0 - \frac{1}{2}\wp\right) \tag{2-13}$$

where $l = \xi_j - \xi_i$ (beam length) and $\wp = 2\frac{(-\mathrm{w}_i + \mathrm{w}_j)}{l} + \beta_i + \beta_j$. Now, Nguyen-Xuan [27] considers the curvature of the Timoshenko beam by using constant α_3 as follows

$$\kappa = \frac{1}{l}\frac{d\beta(\xi)}{d\xi} = \kappa_0 + \frac{6\gamma_0}{l}(1-2\xi) \tag{2-14}$$

with $\kappa_0 = \beta_j - \beta_i - 3\wp(1-2\xi)$. Then, we need to find $\gamma_0(\mathrm{w},\beta)$ through the following minimization of strain energy

$$\min_{\gamma_0} U_{bs} \tag{2-15}$$

where U_{bs} is defined by

$$\begin{aligned} \Pi_{\mathrm{TPE}} &= \frac{l}{2}\int_{\Gamma^\xi} D^b\kappa^2\, d\xi + \frac{l}{2}\int_{\Gamma^\xi} D^s\gamma^2\, d\xi \\ &= \frac{l}{2}\int_0^1 D^b\kappa_0^2\, dx - \frac{6D^b}{l}\left(\gamma_0\wp - \gamma_0^2\right) + \frac{l}{2}D^s\gamma_0^2 \end{aligned} \tag{2-16}$$

with $D^b = Eh^3/12(1-\nu^2)$ and $D^s = Gh$ being bending and shear stiffness, respectively. The minimization generates

$$\gamma_0 = \frac{6D^b}{D^s l^2 + 12D^b}\wp = \chi\wp$$

where $\chi = \frac{(h/l)^2}{2(h/l)^2+\kappa^s(1-\nu)}$. Finally, substituting α_1, α_2, α_3, and γ_0 into Eq. 2-11, we have

$$\begin{cases} \mathrm{w}(\xi) & = \mathrm{w}_i(1-\xi) + \mathrm{w}_j\xi + \frac{l}{2}\left(-\beta_i + \beta_j\right)\xi(1-\xi) \\ & \qquad -\frac{l}{2}\wp(1-2\chi)\xi(1-\xi)(1-2\xi), \\ \beta(\xi) & = \beta_i(1-\xi) + \beta_j\xi - 3\wp(1-2\chi)\xi(1-\xi), \\ \gamma(\xi) & = \chi\wp, \end{cases} \tag{2-17}$$

When $h \to 0$, $\chi \to 0$ and the shear strain $\gamma \to 0$. Therefore, the shear locking is eliminated.

2.2.2.2 Generalization of the original idea for *n*-gons

The procedure to generalize the original idea for an arbitrary polygon (Fig. 2.4) is organized in the following steps according to the work of Nguyen-Xuan [27]. In the first 3 steps, we tackle the shear strains ($\boldsymbol{\gamma}$), while the remaining stands for the bending strains ($\boldsymbol{\kappa}$):

(1) Shear strains along each edge

Let us describe the normal and tangent directions for an edge $\hat{i}$ (see Fig. 2.4)

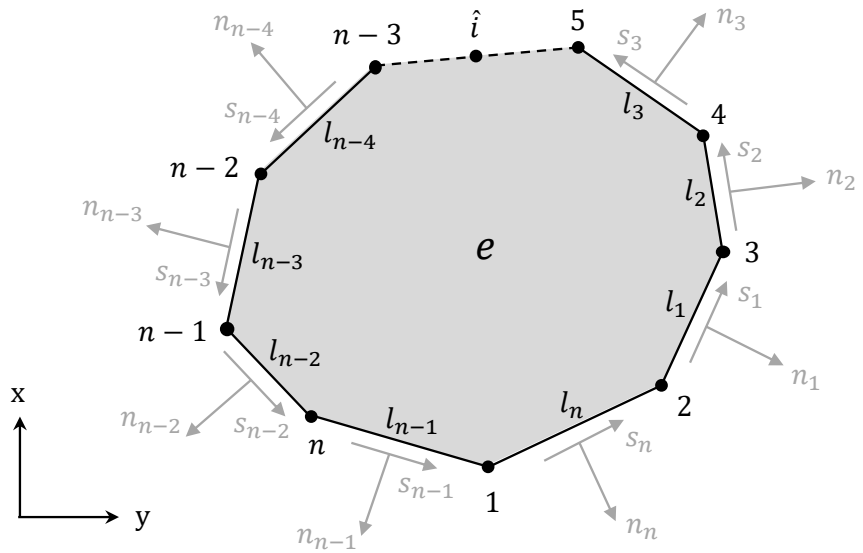


Figure 2.4: Arbitrary polygonal element: normal and tangential vectors on the edges.

$$\mathbf{n}_{\hat{i}} = \frac{1}{l_{\hat{i}}}\left[-b_{\hat{i}} \quad -c_{\hat{i}}\right]^T, \quad \mathbf{s}_{\hat{i}} = \frac{1}{l_{\hat{i}}}\left[c_{\hat{i}} \quad -b_{\hat{i}}\right]^T \tag{2-18}$$

with

$$\begin{cases} \hat{i} & = 1, 2, \ldots, n, \\ \hat{j} & = 2, 3, \ldots, n, 1, \\ \hat{k} & = 3, \ldots, n, 1, 2. \end{cases}$$

where $b_{\hat{i}} = y_{\hat{j}} - y_{\hat{k}}$, $c_{\hat{i}} = x_{\hat{k}} - x_{\hat{j}}$, $l_{\hat{i}} = ||\boldsymbol{x}_{\hat{j}} - \boldsymbol{x}_{\hat{k}}||$, with n being the number of nodes in an element e. From Eq. 2-17, the shear strains on the tangent direction of the $\hat{i}$th edge are

$$\bar{\gamma}_{s.\hat{j}\hat{k}}(\boldsymbol{x}) = \chi_{\hat{i}} \wp_{\hat{i}} \tag{2-19}$$

where

$$\begin{aligned} \wp_{\hat{i}} &= \frac{2}{l_{\hat{i}}} \left(-\mathrm{w}_{\hat{j}} + \mathrm{w}_{\hat{k}}\right) + \mathbf{s}_{\hat{i}} \cdot \left[\left(\beta_{x.\hat{j}} + \beta_{x.\hat{k}}\right) \quad \left(\beta_{y.\hat{j}} + \beta_{y.\hat{k}}\right)\right]^T \\ &= \frac{1}{l_{\hat{i}}} \left\{2\left(-\mathrm{w}_{\hat{j}} + \mathrm{w}_{\hat{k}}\right) - c_{\hat{i}}\left(\beta_{x.\hat{j}} + \beta_{x.\hat{k}}\right) + b_{\hat{i}}\left(\beta_{y.\hat{j}} + \beta_{y.\hat{k}}\right)\right\} \end{aligned} \tag{2-20}$$

The relation between $\bar{\gamma}_s$ and (γ_x, γ_y) along each edge gives

$$\begin{bmatrix} \bar{\gamma}_{n.\hat{j}\hat{k}} \\ \bar{\gamma}_{s.\hat{j}\hat{k}} \end{bmatrix} = \begin{bmatrix} \mathbf{n}_{\hat{i}}^T \\ \mathbf{s}_{\hat{i}}^T \end{bmatrix} \begin{bmatrix} \gamma_{x.\hat{j}\hat{k}} \\ \gamma_{y.\hat{j}\hat{k}} \end{bmatrix} \Rightarrow l_{\hat{i}} \bar{\gamma}_{s.\hat{j}\hat{k}} = c_{\hat{i}} \gamma_{x.\hat{j}\hat{k}} - b_{\hat{i}} \gamma_{y.\hat{j}\hat{k}} \tag{2-21}$$

(2) Nodal shear strains

There are two edges sharing the same node $\hat{i}$ ($\overline{\hat{m}\hat{i}}$ and $\overline{\hat{i}\hat{j}}$), as can be seen in Fig. 2.4. From Eqs. 2-19 and 2-20, the assumed nodal shear strain ($\tilde{\gamma}_{x.\hat{i}}$, $\tilde{\gamma}_{y.\hat{i}}$) can be represented by constant shear strains along $\overline{\hat{m}\hat{i}}$ and $\overline{\hat{i}\hat{j}}$, $\bar{\gamma}_{s.\hat{m}\hat{i}}$ and $\bar{\gamma}_{s.\hat{i}\hat{j}}$, as

$$\begin{bmatrix} \tilde{\gamma}_{x.\hat{i}} \\ \tilde{\gamma}_{y.\hat{i}} \end{bmatrix} = \frac{1}{c_{\hat{m}} b_{\hat{q}} - c_{\hat{q}} b_{\hat{m}}} \begin{bmatrix} -b_{\hat{m}} & b_{\hat{q}} \\ -c_{\hat{m}} & c_{\hat{q}} \end{bmatrix} \begin{bmatrix} l_{\hat{q}} \chi_{\hat{q}} \wp_{\hat{q}} \\ l_{\hat{m}} \chi_{\hat{m}} \wp_{\hat{m}} \end{bmatrix} \tag{2-22}$$

with

$$\begin{cases} \hat{q} = n-1, n, 1, 2, \ldots, n-2 \\ \hat{m} = n, 1, 2, \ldots, n-1 \end{cases}$$

(3) Approximation of shear strains

Now, shape functions are used to interpolate the assumed nodal shear strains ($\tilde{\gamma}_{x.\hat{i}}$, $\tilde{\gamma}_{y.\hat{i}}$) obtained in step *(2)* to form the element assumed shear strains $\tilde{\boldsymbol{\gamma}}^e \left(\tilde{\gamma}_x^e, \tilde{\gamma}_y^e\right)$

$$\begin{aligned} \tilde{\boldsymbol{\gamma}}^e &= \begin{bmatrix} \tilde{\gamma}_x^e \\ \tilde{\gamma}_y^e \end{bmatrix} = \begin{bmatrix} \sum_{\hat{i}} \phi_{\hat{i}}^e \tilde{\gamma}_{x.\hat{i}} \\ \sum_{\hat{i}} \phi_{\hat{i}}^e \tilde{\gamma}_{y.\hat{i}} \end{bmatrix} \\ &= \sum_{\hat{i},\hat{j},\hat{k},\hat{m}} \begin{bmatrix} \frac{b_{\hat{m}}\phi_{\hat{j}}}{c_{\hat{i}} b_{\hat{m}} - c_{\hat{m}} b_{\hat{i}}} - \frac{b_{\hat{j}}\phi_{\hat{k}}}{c_{\hat{j}} b_{\hat{i}} - c_{\hat{i}} b_{\hat{j}}} \\ \frac{c_{\hat{m}}\phi_{\hat{j}}}{c_{\hat{i}} b_{\hat{m}} - c_{\hat{m}} b_{\hat{i}}} - \frac{c_{\hat{j}}\phi_{\hat{k}}}{c_{\hat{j}} b_{\hat{i}} - c_{\hat{i}} b_{\hat{j}}} \end{bmatrix}_{(2\times n)} [l_{\hat{i}} \chi_{\hat{i}} \wp_{\hat{i}}]_{(n\times 1)} \end{aligned} \tag{2-23}$$

where $\phi_{\hat{i}}^{e}$ is the shape function evaluated at the node $\hat{i}$

(4) Edge normal and tangent rotations for bending strains

The normal rotation $\overline{\beta}_{n.\hat{j}\hat{k}}$ along an edge $\overline{\hat{j}\hat{k}}$ is approximated as

$$\bar{\beta}_{n.\hat{j}\hat{k}} = \left(\beta_{n.\hat{j}\hat{k}}\right)_{\hat{j}} \phi_{\hat{j}} + \left(\beta_{n.\hat{j}\hat{k}}\right)_{\hat{k}} \phi_{\hat{k}} \tag{2-24}$$

where $\left(\beta_{n.\hat{j}\hat{k}}\right)_{\hat{j}}$ and $\left(\beta_{n.\hat{j}\hat{k}}\right)_{\hat{k}}$ are the normal rotations along edge $\overline{\hat{j}\hat{k}}$ at nodes $\hat{j}$ and $\hat{k}$, respectively. The tangent rotation $\bar{\beta}_{s.\hat{j}\hat{k}}$ is determined by the second equation in Eq. 2-17, i.e.

$$\bar{\beta}_{s.\hat{j}\hat{k}} = \left(\beta_{s.\hat{j}\hat{k}}\right)_{\hat{j}} \phi_{\hat{j}} + \left(\beta_{s.\hat{j}\hat{k}}\right)_{\hat{k}} \phi_{\hat{k}} + 3\left(1 - 2\chi_{\hat{i}}\right) \wp_i \phi_{\hat{j}} \phi_{\hat{k}} \tag{2-25}$$

where $\left(\beta_{s.\hat{j}\hat{k}}\right)_{\hat{j}}$ and $\left(\beta_{s.\hat{j}\hat{k}}\right)_{\hat{k}}$ are the tangent rotation along edge $\overline{\hat{j}\hat{k}}$ at nodes $\hat{j}$ and $\hat{k}$, respectively. Analogously to shear strain process

$$\begin{bmatrix} \bar{\beta}_{n.\hat{j}\hat{k}} \\ \bar{\beta}_{s.\hat{j}\hat{k}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{n}_{\hat{i}}^{T} \\ \boldsymbol{s}_{\hat{i}}^{T} \end{bmatrix} \begin{bmatrix} \beta_{x.\hat{j}\hat{k}} \\ \beta_{y.\hat{j}\hat{k}} \end{bmatrix} \Rightarrow \begin{bmatrix} \beta_{x.\hat{j}\hat{k}} \\ \beta_{y.\hat{j}\hat{k}} \end{bmatrix} = \frac{1}{l_{\hat{i}}} \begin{bmatrix} -b_{\hat{i}} & c_{\hat{i}} \\ -c_{\hat{i}} & -b_{\hat{i}} \end{bmatrix} \begin{bmatrix} \bar{\beta}_{n.\hat{j}\hat{k}} \\ \bar{\beta}_{s.\hat{j}\hat{k}} \end{bmatrix} \tag{2-26}$$

.

Furthermore, (β_x, β_y) at nodes $\hat{j}$ and $\hat{k}$ are expressed as

$$\begin{aligned} \begin{bmatrix} \beta_{x.\hat{j}} \\ \beta_{y.\hat{j}} \end{bmatrix} &= \frac{1}{l_{\hat{i}}} \begin{bmatrix} -b_{\hat{i}} & c_{\hat{i}} \\ -c_{\hat{i}} & -b_{\hat{i}} \end{bmatrix} \begin{bmatrix} \left(\beta_{n.\hat{j}\hat{k}}\right)_{\hat{j}} \\ \left(\beta_{s.\hat{j}\hat{k}}\right)_{\hat{j}} \end{bmatrix}, \\ \begin{bmatrix} \beta_{x.\hat{k}} \\ \beta_{y.\hat{k}} \end{bmatrix} &= \frac{1}{l_{\hat{i}}} \begin{bmatrix} -b_{\hat{i}} & c_{\hat{i}} \\ -c_{\hat{i}} & -b_{\hat{i}} \end{bmatrix} \begin{bmatrix} \left(\beta_{n.\hat{j}\hat{k}}\right)_{\hat{k}} \\ \left(\beta_{s.\hat{j}\hat{k}}\right)_{\hat{k}} \end{bmatrix} \end{aligned} \tag{2-27}$$

Rearranging Eqs. 2-24, 2-25 and 2-26 into Eq. 2-27, we obtain

$$\begin{bmatrix} \beta_{x.\hat{j}\hat{k}} \\ \beta_{y.\hat{j}\hat{k}} \end{bmatrix} = \begin{bmatrix} \beta_{x.\hat{j}}\phi_{\hat{j}} + \beta_{x.\hat{k}}\phi_{\hat{k}} + \frac{3c_{\hat{i}}}{l_{\hat{i}}}\left(1 - 2\chi_{\hat{i}}\right) \wp_{\hat{i}} \phi_{\hat{j}} \phi_{\hat{k}} \\ \beta_{y.\hat{j}}\phi_{\hat{j}} + \beta_{y.\hat{k}}\phi_{\hat{k}} - \frac{3b_{\hat{i}}}{l_{\hat{i}}}\left(1 - 2\chi_{\hat{i}}\right) \wp_{\hat{i}} \phi_{\hat{j}} \phi_{\hat{k}} \end{bmatrix} \tag{2-28}$$

(5) Approximation of rotation fields in the element

Like the interpolation performed with the shape functions to obtain shear strains in step *(3)*, we approximate the assumed rotation field within the polygonal element from Eq. 2-28, leading to

$$\begin{bmatrix} \tilde{\beta}_x \\ \tilde{\beta}_y \end{bmatrix} = \begin{bmatrix} \sum_{\hat{i}} \beta_{x.\hat{i}}\phi_{\hat{i}} + \sum_{\hat{i},\hat{j},\hat{k}} \frac{3c_{\hat{i}}}{l_{\hat{i}}}\left(1 - 2\chi_{\hat{i}}\right) \wp_{\hat{i}} \phi_{\hat{j}} \phi_{\hat{k}} \\ \sum_{\hat{i}} \beta_{y.\hat{i}}\phi_{\hat{i}} - \sum_{\hat{i},\hat{j},\hat{k}} \frac{3b_{\hat{i}}}{l_{\hat{i}}}\left(1 - 2\chi_{\hat{i}}\right) \wp_{\hat{i}} \phi_{\hat{j}} \phi_{\hat{k}} \end{bmatrix}. \tag{2-29}$$

Finally, the curvature field of the polygonal element can be expressed as

$$\tilde{\boldsymbol{\kappa}}^e = \begin{bmatrix} \tilde{\beta}_{x,x} \\ \tilde{\beta}_{y,y} \\ \tilde{\beta}_{y,x} + \tilde{\beta}_{x,y} \end{bmatrix} \tag{2-30}$$

$$= \begin{bmatrix} \sum_{\hat{i}} \beta_{x.\hat{i}} \frac{\partial \phi_{\hat{i}}}{\partial x} + \sum_{\hat{i},\hat{j},\hat{k}} \frac{3c_{\hat{i}}}{l_{\hat{i}}} \left(\frac{\partial \phi_{\hat{j}}}{\partial x} \phi_{\hat{k}} + \frac{\partial \phi_{\hat{k}}}{\partial x} \phi_{\hat{j}} \right) (1 - 2\chi_{\hat{i}}) \wp_{\hat{i}} \\ \sum_{\hat{i}} \beta_{y.\hat{i}} \frac{\partial \phi_{\hat{i}}}{\partial y} - \sum_{\hat{i},\hat{j},\hat{k}} \frac{3b_{\hat{i}}}{l_{\hat{i}}} \left(\frac{\partial \phi_{\hat{j}}}{\partial y} \phi_{\hat{k}} + \frac{\partial \phi_{\hat{k}}}{\partial y} \phi_{\hat{j}} \right) (1 - 2\chi_{\hat{i}}) \wp_{\hat{i}} \\ \sum_{\hat{i}} \beta_{x.\hat{i}} \frac{\partial \phi_{\hat{i}}}{\partial y} + \sum_{\hat{i}} \beta_{y.\hat{i}} \frac{\partial \phi_{\hat{i}}}{\partial x} + \sum_{\hat{i},\hat{j},\hat{k}} \frac{3}{l_{\hat{i}}} \left(c_{\hat{i}} \frac{\partial \phi_{\hat{j}}}{\partial y} \phi_{\hat{k}} + c_{\hat{i}} \frac{\partial \phi_{\hat{k}}}{\partial y} \phi_{\hat{j}} \right) (1 - 2\chi_{\hat{i}}) \wp_{\hat{i}} \cdots \\ - \sum_{\hat{i},\hat{j},\hat{k}} \frac{3}{l_{\hat{i}}} \left(b_{\hat{i}} \frac{\partial \phi_{\hat{j}}}{\partial x} \phi_{\hat{k}} + b_{\hat{i}} \frac{\partial \phi_{\hat{k}}}{\partial x} \phi_{\hat{j}} \right) (1 - 2\chi_{\hat{i}}) \wp_{\hat{i}} \end{bmatrix}$$

(6) Matrix representation of the shear and bending strains

Eqs. 2-23 and 2-30 can be rewritten in matrix form as

$$\tilde{\boldsymbol{\gamma}}^e = \tilde{\mathbf{B}}^{\mathbf{s},\mathbf{e}} \mathbf{d}^{\mathbf{e}} = \mathbf{H}^{\mathbf{s}} \mathbf{I}^{\mathbf{s}} \mathbf{G}\, \mathbf{d}^{\mathbf{e}}, \tag{2-31}$$

$$\tilde{\boldsymbol{\kappa}}^e = \boldsymbol{\kappa}^e + \hat{\boldsymbol{\kappa}}^e = \left(\mathbf{B}^{\mathbf{b},\mathbf{e}} + \hat{\mathbf{B}}^{\mathbf{b},\mathbf{e}} \right) \mathbf{d}^{\mathbf{e}} = \left(\mathbf{B}^{\mathbf{b},\mathbf{e}} + \mathbf{H}^{\mathbf{b}} \mathbf{I}^{\mathbf{b}} \mathbf{G} \right) \mathbf{d}^{\mathbf{e}}, \tag{2-32}$$

where $\mathbf{d}^{\mathbf{e}} = \left[\mathbf{d}_{\mathbf{1}}^{\mathbf{e}T} \ \ldots \ \mathbf{d}_{\mathbf{n}}^{\mathbf{e}T} \right]^T$ and

$$\begin{cases} \mathbf{H}^{\mathbf{b}}{}_{(3\times n)} = \sum_{\hat{i},\hat{j},\hat{k}} \begin{bmatrix} \frac{3c_{\hat{i}}}{l_{\hat{i}}^2} \left(\frac{\partial \phi_{\hat{j}}}{\partial x} \phi_{\hat{k}} + \frac{\partial \phi_{\hat{k}}}{\partial x} \phi_{\hat{j}} \right) \\ -\frac{3b_{\hat{i}}}{l_{\hat{i}}^2} \left(\frac{\partial \phi_{\hat{j}}}{\partial y} \phi_{\hat{k}} + \frac{\partial \phi_{\hat{k}}}{\partial y} \phi_{\hat{j}} \right) \\ \frac{3}{l_{\hat{i}}^2} \left[\left(c_{\hat{i}} \frac{\partial \phi_{\hat{k}}}{\partial y} - b_{\hat{i}} \frac{\partial \phi_{\hat{k}}}{\partial x} \right) \phi_{\hat{j}} + \left(c_{\hat{i}} \frac{\partial \phi_{\hat{j}}}{\partial y} - b_{\hat{i}} \frac{\partial \phi_{\hat{j}}}{\partial x} \right) \phi_{\hat{k}} \right] \end{bmatrix}, \\ \mathbf{I}^{\mathbf{b}}{}_{(n\times n)} = \delta_{\hat{i}\hat{i}} (1 - 2\chi_{\hat{i}}), \\ \mathbf{H}^{\mathbf{s}}{}_{(2\times n)} = \sum_{\hat{i},\hat{j},\hat{k},\hat{m}} \begin{bmatrix} \frac{b_{\hat{m}} \phi_{\hat{j}}}{c_{\hat{i}} b_{\hat{m}} - c_{\hat{m}} b_{\hat{i}}} - \frac{b_{\hat{j}} \phi_{\hat{k}}}{c_{\hat{j}} b_{\hat{i}} - c_{\hat{i}} b_{\hat{j}}} \\ \frac{c_{\hat{m}} \phi_{\hat{j}}}{c_{\hat{i}} b_{\hat{m}} - c_{\hat{m}} b_{\hat{i}}} - \frac{c_{\hat{j}} \phi_{\hat{k}}}{c_{\hat{j}} b_{\hat{i}} - c_{\hat{i}} b_{\hat{j}}} \end{bmatrix}, \\ \mathbf{I}^{\mathbf{s}}{}_{(n\times n)} = \delta_{\hat{i}\hat{i}} \chi_{\hat{i}}, \\ \mathbf{G}_{(n\times 3n)} = \sum_{\hat{i},\hat{j}} \left[-2_{\hat{i},3\hat{j}-2} \ \ c_{\hat{i},3\hat{j}-1} \ \ - b_{\hat{i},3j} \right] + \sum_{\hat{i},\hat{k}} \left[2_{\hat{i},3\hat{k}-2} \ \ c_{\hat{i},3\hat{k}-1} \ \ - b_{\hat{i},3\hat{k}} \right], \end{cases} \tag{2-33}$$

with

$$\begin{cases} \hat{i} & = 1, 2, \ldots, n \\ \hat{j} & = 2, 3, \ldots, n, 1 \\ \hat{k} & = 3, 4, \ldots, n, 1, 2 \\ \hat{m} & = n, 1, 2, \ldots, n-1. \end{cases}$$

2.2.2.3 Locking-free stiffness matrix

The present locking-free technique turns the local stiffness matrix of Eq. 2-10 into the following expression

$$\mathbf{K}_0^e = \int_{\Omega^e} \left(\mathbf{B}^{\mathbf{b,e}} + \hat{\mathbf{B}}^{\mathbf{b,e}}\right)^T \mathbf{D}^{\mathbf{b}} \left(\mathbf{B}^{\mathbf{b,e}} + \hat{\mathbf{B}}^{\mathbf{b,e}}\right) d\Omega^e + \int_{\Omega^e} \tilde{\mathbf{B}}^{\mathbf{s,e}T} \mathbf{D}^{\mathbf{s}} \tilde{\mathbf{B}}^{\mathbf{s,e}} d\Omega^e, \quad (2\text{-}34)$$

where $\tilde{\mathbf{B}}^{\mathbf{s,e}}$ and $\hat{\mathbf{B}}^{\mathbf{b,e}}$ are defined in Eqs. 2-31 and 2-32, respectively.

2.2.3 Stress measure on plates

Since stresses will be evaluated for plate structures in this work, let us define the stress measure adopted. The Cauchy stress vector (using Voigt notation) at a given point can be expressed as

$$\begin{bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{zx} \end{bmatrix} = \begin{bmatrix} \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & (1-\nu)/2 \end{bmatrix} & \mathbf{0} \\ \mathbf{0} & \begin{bmatrix} G & 0 \\ 0 & G \end{bmatrix} \end{bmatrix} \begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{bmatrix}, \quad (2\text{-}35)$$

since σ_z is negligible over the other stress components in plate bending. Moreover, the transverse shear stresses τ_{yz} and τ_{zx} are also usually small comparing to σ_x, σ_y and τ_{xy} [15]. Therefore, in this work, we compute the stress vector as

$$\begin{bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{bmatrix} = \left(\frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & (1-\nu)/2 \end{bmatrix} \right) \begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{bmatrix} \quad (2\text{-}36)$$

or

$$\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon}. \quad (2\text{-}37)$$

Because we are interested in the maximum absolute stress values at a given element centroid, we shall consider $z = \pm h/2$ (at plate top or bottom surface), i.e.

$$\boldsymbol{\varepsilon}^{\boldsymbol{e}} = z|_{\pm h/2}\, \mathbf{B}^{\mathbf{b,e}} \mathbf{d}^{\mathbf{e}}. \quad (2\text{-}38)$$

Given the finite element solution, we note that the locking-free strain-displacement matrices are unnecessary once no integration is performed and no numerical instability is expected in the stress quantification.

Additionally, the von Mises stress will be calculated from now on at the centroid of an element e as

$$\sigma_e^v = \sqrt{\boldsymbol{\sigma}_e^T \mathbf{V}^v \boldsymbol{\sigma}_e}, \quad (2\text{-}39)$$

where $\boldsymbol{\sigma}_e = \mathbf{D}\boldsymbol{\varepsilon}^e$ and

$$\mathbf{V}^v = \begin{bmatrix} 1 & -1/2 & 0 \\ -1/2 & 1 & 0 \\ 0 & 0 & 3 \end{bmatrix}. \tag{2-40}$$

2.3 Validation

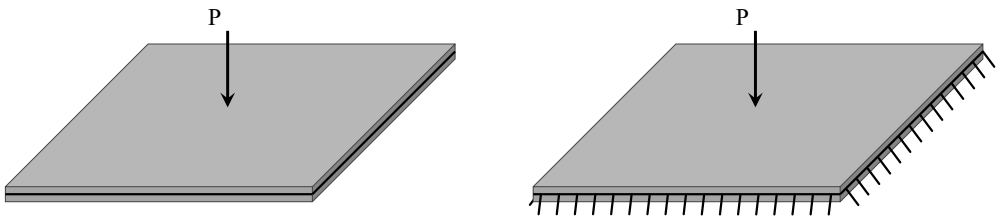
In this Section, we conduct several numerical experiments to validate the performance of the polygonal plate element when calculating displacements, moments, and stresses. In addition to these results, the reader is referred to the work of Nguyen-Xuan [27], who studied the locking-free performance of various elements, including the one used herein (PRMn-W), by evaluating finite element responses across different span-to-thickness ratios (L/h).

2.3.1 Plates under point load

Figures 2.5(a) and 2.5(b) illustrate a simply-supported plate and a clamped plate under a centered concentrated load, respectively. The normalized displacement at the center of an $L \times L$ plate, where a central load P is applied, can be determined as [25]

$$\overline{\mathrm{w}} = \frac{Eh^3}{12PL^2(1-\nu^2)}\mathrm{w}. \tag{2-41}$$

The mesh was discretized into 16,000 arbitrary polygonal elements. The parameters adopted were $L = 20$, $E = 30 \times 10^6$, $P = -10^4$, $h = 0.1$ (thin plate) and $\nu = 0.3$.



(a) Simply-supported (b) Clamped

Figure 2.5: Plate boundary conditions.

Table 2.2 presents the results, while Fig. 2.6 demonstrates the simulated plate deformations for each case. We included tests for the traditional selective integration method [22, 23] to show its inefficiency in solving the locking

anomaly on polygonal elements (Figs. 2.6(c) and 2.6(d)). However, by applying the technique proposed by Nguyen-Xuan [27], adopted in this study, we successfully avoided shear locking in the finite element module of the `PolyTop` code adapted for plates. It is important to note that quadrilateral or triangular meshes (called herein as conventional meshes) are also suitable to be used with the present technique.

Table 2.2: Finite Element normalized central deflections $\overline{\mathrm{w}}$ of square plates: shear locking effects.

Boundary condition	**No locking treatment**	**Selective integration**	**Locking treatment**	**Reference value [25]**
Simply supported	0.0082885	0.0087925	0.011618	0.011601
Clamped	0.0039949	0.0042308	0.0056304	0.0056012

2.3.2 Plates under uniformly distributed load

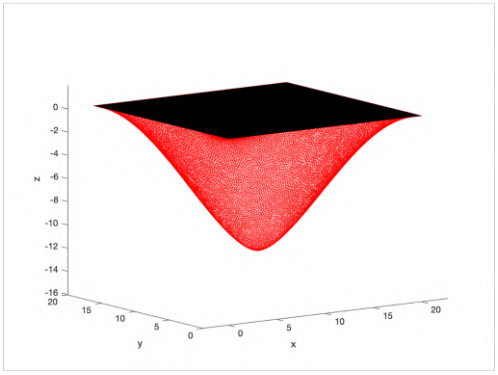
Other FE benchmarks available in the literature are the class of square plates under uniformly distributed load. These examples will be distinguished by four capital letters, where each letter represents the Dirichlet boundary condition on each plate's edge. For example, "SSFC" means that two parallel sides are simply supported (S), while the remaining are free (F) and clamped (C). With these examples, we will also evaluate moments and/or stresses. For all cases, the polygonal meshes are arbitrarily generated with 16,000 elements.

2.3.2.1 SSSS plate

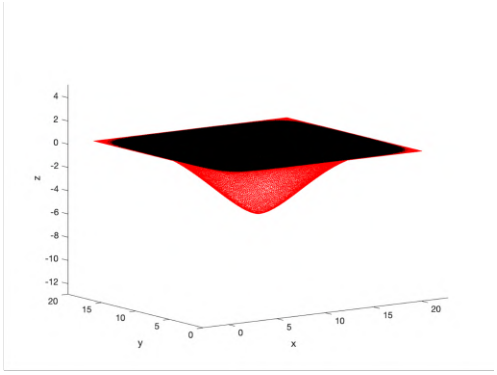
An $L \times L$ square plate with all sides simply supported is investigated again, but under a uniformly distributed load q. The Poisson ratio is 0.25. According to Reddy [42], reference values of normalized maximum deflection $\bar{\mathrm{w}}$ and normalized maximum stresses $\bar{\sigma}$ are obtained as [42]

$$\bar{\mathrm{w}} = \left(\frac{Eh^3}{qL^4}10^2\right)\mathrm{w}, \tag{2-42}$$

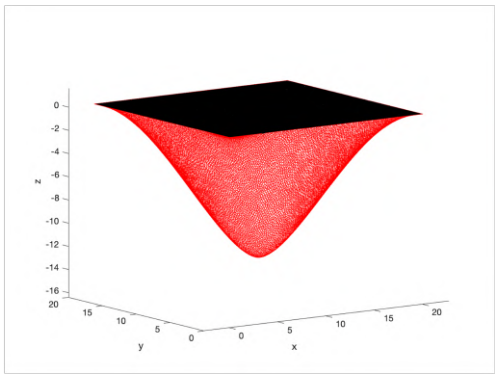
$$\bar{\sigma} = \left(\frac{h^2}{qL^2}\right)\sigma. \tag{2-43}$$



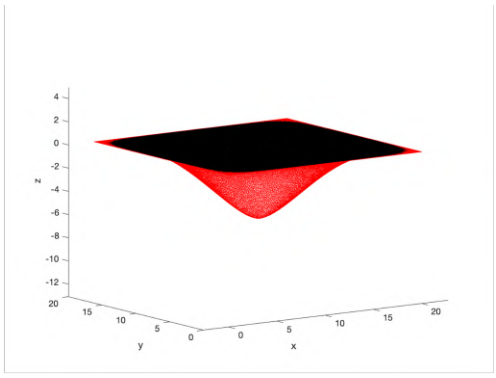
(a) Simply supported plate (no locking treatment)



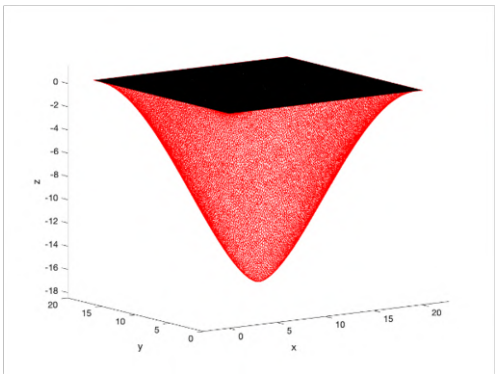
(b) Clamped plate (no locking treatment)



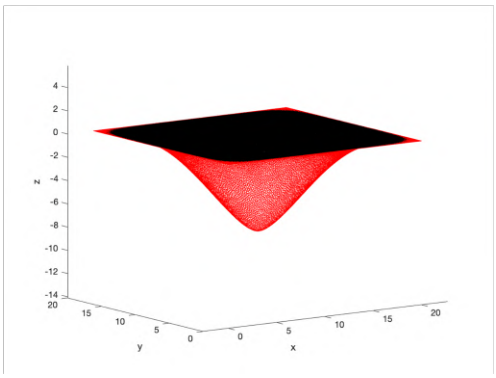
(c) Simply supported plate (Selective Integration) [22, 23]



(d) Clamped plate (Selective Integration) [22, 23]



(e) Simply supported plate (Locking-Free Technique) [27]



(f) Clamped plate (Locking-Free Technique) [27]

Figure 2.6: Deformations of square plates under a centered concentrated load: shear locking effects.

The polygonal mesh generated for this example is illustrated in Fig. 2.7. Results are depicted in Table 2.3, where small relative errors are seen for maximum transverse deflection and stresses. Figure 2.8 displays the configuration of the deformed plate and the stress maps. From now on, we choose to hide elements' edges to make the plots cleaner.

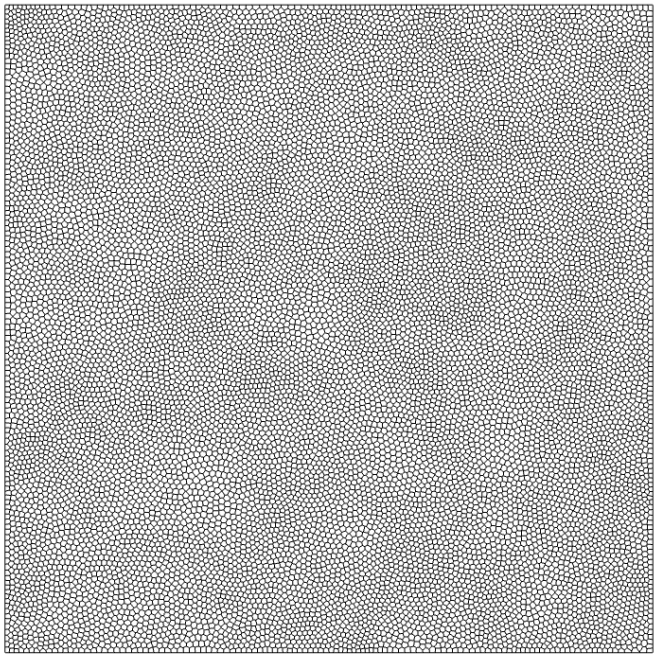


Figure 2.7: SSSS plate mesh.

Table 2.3: SSSS plate example – Normalized maximum deflection and stresses under uniformly distributed load.

	Present Method	**Reference Value [42]**	**Relative Error**
$\bar{\mathrm{w}}$	4.578	4.570	0.1731%
$\bar{\sigma}_x$	0.27617	0.27620	0.00995%
$\bar{\sigma}_y$	0.2771	0.2762	0.3249%
$\bar{\tau}_{xy}$	0.2099	0.2085	0.6694%

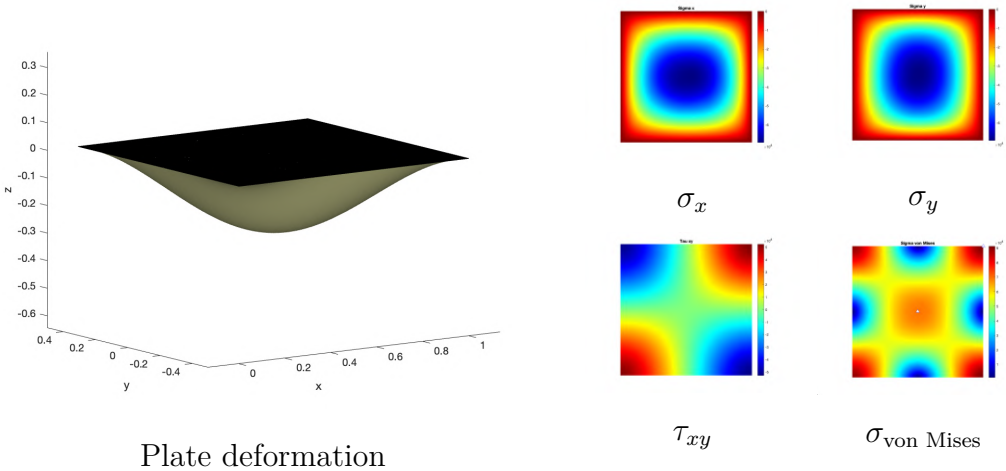


Figure 2.8: SSSS plate example – deformation and stress results.

2.3.2.2 CCCC plate

An $L \times L$ square plate with all sides clamped is investigated, this time under a uniformly distributed load ($q = 10.0$ kN/m^2). The Young's modulus is $E = 2.0 \times 10^8$ kN/m^2, the Poisson ratio is 0.3, $L = 1.0$ m, and the thickness is $h = 0.004$ m. Reference values for central deflection w and bending moments (m_x and m_y) are taken from Radwańska et al. [18], in which moments are given in Nm/m, and were determined herein from stresses calculated in the FE routine by [18]

$$m = \left(\frac{h^2}{6}\right)\sigma \ . \tag{2-44}$$

Table 2.4 demonstrates that the results are in good agreement with the reference values. Figure 2.9 shows a 50-times amplified configuration of the deformed plate and the stress maps.

Table 2.4: CCCC plate example – Deflection (m) and bending moments (Nm/m) at the center of the plate under uniformly distributed load.

	Present Method	**Reference Value [18]**	**Relative Error**
w	0.0108	0.0107	0.8116%
m_x	230.6	231.0	0.1603%
m_y	230.5	231.0	0.2058%

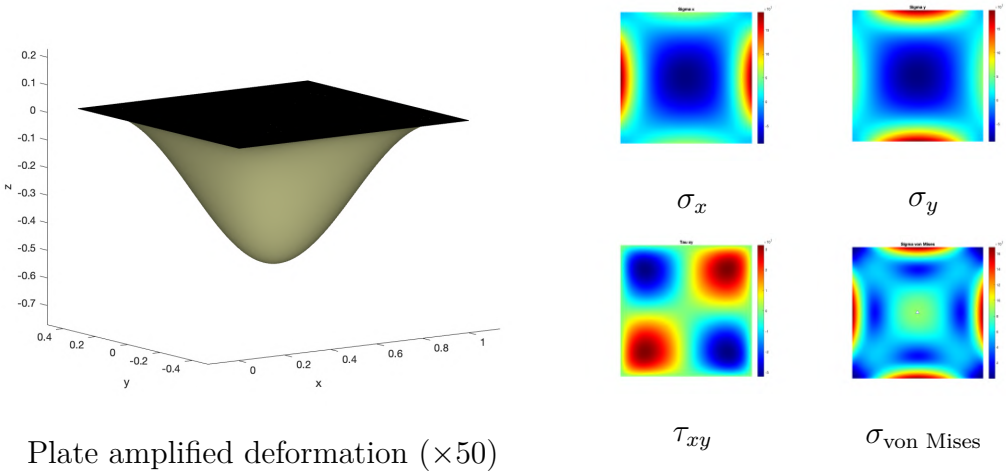


Figure 2.9: CCCC plate example – deformation and stress results.

2.3.2.3 SSFF plate

The last example is an $L \times L$ square plate under uniformly distributed load with two parallel sides simply supported and the others free. The Poisson ratio is equal to 0.3 again. Reference values of normalized maximum deflection $\bar{\text{w}}$ and normalized maximum moments $\overline{M}$ are obtained as [43]

$$\bar{\text{w}} = \left(\frac{Eh^3}{12qL^4(1-\nu^2)}\right)\text{w}, \tag{2-45}$$

$$\overline{M} = \left(\frac{h^2}{6qL^2}\right)\sigma. \tag{2-46}$$

Table 2.5 lists the results quite consonant to the literature, along with the plots exhibited in Fig. 2.10, where the deformed configuration of the plate was amplified by a factor of 500. It is still important to note that the reference values from [43] are not analytical, but numerically simulated with the namely "QUAD9* element", an assumed strain 9-node Lagrangian element.

Table 2.5: SSFF plate example – Normalized maximum deflection and moments of a square plate under uniformly distributed load.

	Present Method	**Reference Value [43]**	**Relative Error**
$\bar{w}$	0.0152	0.0151	0.8836%
$\overline{M}_x$	0.0274	0.0273	0.5216%
$\overline{M}_y$	0.1319	0.1317	0.1200%
$\overline{M}_{xy}$	0.0214	0.0218	1.7007%

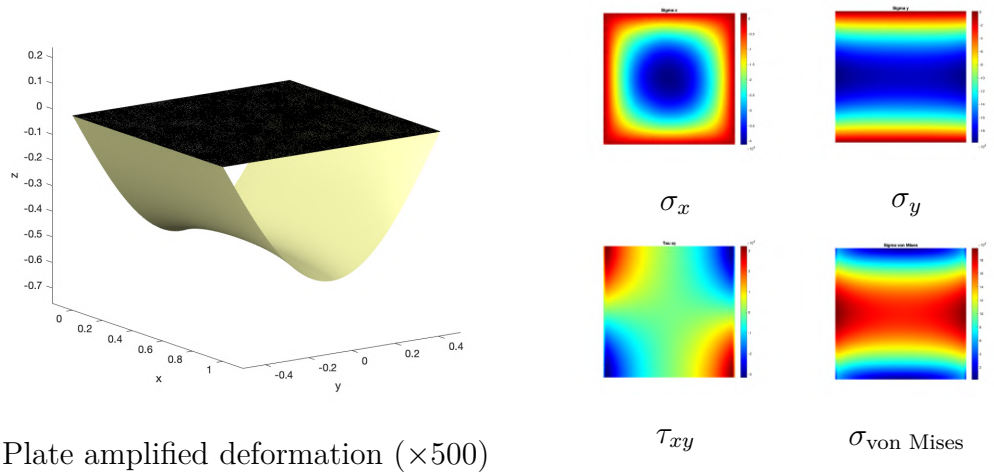


Figure 2.10: SSFF plate example – deformation and stress results.

3
Topology optimization applied to plate structures

Each Section of this Chapter will be divided into two parts: one for the compliance topology optimization and the other for the stress-constrained topology optimization.

3.1
Literature review

3.1.1
Compliance topology optimization of plates

As earlier discussed, polygonal finite element methods for thick-thin plates are quite recent or are still being studied. As a consequence, topology optimization of plates using arbitrary polygonal meshes is even more recent in the literature. In 2022, Pham and Phan [44] were the first to study topology optimization of thick-thin plate structures using the PRMn-PL element [27], where the classical compliance minimization was adopted. From the best of our knowledge, this was the sole topology optimization work available that uses polygonal plate FE formulation. In the same year but using a polygonal shell FE formulation, Ho-Nguyen-Tan and Kim [45] conducted a level-set based topology optimization for compliance and stress minimization of shell structures, which included some plate examples. Although shell approach is mostly able to simulate plates, we will discuss this latter methodology by Ho-Nguyen-Tan and Kim further in the Chapters for shells.

One may identify this scenario as an open field for investigations in topology optimization of plates using arbitrary polygonal meshes. In this work, we will first present the basic compliance optimization of plate structures, comparing some examples with the available references. However, as an opportunity to explore further gaps in the literature, we also carry out a topology optimization of plates considering local stress constraints.

3.1.2 Stress-constrained topology optimization of plates

The outcomes from regular topology optimization may not be adequate since sharp or re-entrant regions can emerge and cause undesired stress concentration. At the same time, topology optimization considering stress constraints involves classical numerical challenges, such as the singular optima [46, 47, 48], which appears as the stress measure is defined microscopically. Therefore, it does not vanish in regions without material (design variable near zero), which is physically incorrect. The global optimum is at a subspace disconnected from the solution space, so traditional gradient-based optimization techniques cannot achieve it. Some strategies emerged to include the degenerated subspace into the solution space, such as the ε relaxation [49] and the q-p relaxation [50], along with a vast variation thereof.

Another drawback is the high number of constraints since stresses are local measures generally linked to at least the number of elements, requiring expensive computational effort. Most works focus on global stress quantification strategies, both on membrane and plate elements. These strategies are known as aggregation techniques, wherein local stress measures are computed into a single and smooth quantity, such as the p-norm [51, 52, 53, 54], and the KS norm [55]. The first evidence in the literature addressing the stress-constrained topology optimization for plate structures was only in 2016 [56] with global aggregation, the same approach utilized by the subsequent works [57, 58, 59, 45]. The main concern of aggregation methods is that global quantification can still leave behind regions of highly concentrated stress (as evidenced by Guo et al. [60]). Le et al. [54] and Paris et al. [61] enhanced the aggregation techniques by clustering sets of elements into regions to evaluate. Even though the clustering improves the method, the decision on the number of clusters and their spatial delimitation turns the process problem- and mesh-dependent.

Silva et al. [62] performed a consistent comparison between local and global methodologies, concluding that the topic should be addressed with local strategies. Guo et al. [60], Duysinx and Bendsøe [63], and Bruggi and Duysinx [64] used a procedure only to consider some stress constraints when they are close to being violated, which they call the active-set method. Although it shows interesting outcomes, this strategy may still be expensive for large-scale problems or highly loaded structures. Pereira et al. [65], James et al. [66], Emmendoerfer Jr and Fancello [67, 68], and da Silva et al. [69] adopted the Augmented Lagrangian method [70, 71], in which the optimization problem becomes unconstrained by adding a penalization term to the objective function. Senhora et al. [72] first disclosed an aggregation-free local stress-constrained

topology optimization using an adapted Augmented Lagrangian method for arbitrary membrane and solid polygonal meshes. Then, an educational software developed by Giraldo-Londoño and Paulino [13], `PolyStress`, extended this approach for nonlinear materials with further adjustments. In the evaluation conducted by Silva et al. [62], the Augmented Lagrangian methods were shown to be promising and preferable to other techniques due to their superior efficiency in solving medium- and large-scale problems. To the best of our knowledge, the literature still lacks a local stress constraint approach for topology optimization of plate structures, disregarding the element geometry.

In the next Sections, we carry out two types of topology optimization. First, the compliance minimization, which is already established by [44], is validated. Then, for the first time in the literature, an aggregation-free topology optimization with local stress constraints is investigated for plate structures.

3.2 Compliance minimization

Before proceeding straight to the topology optimization problem, we introduce some topics, specifically those that will be set differently for the upcoming stress-constrained optimization.

3.2.1 Filter operator

To avoid numerical instabilities, a filter operator matrix is applied according to the linear hat filter [6]

$$P_{ij} = \frac{w_{ij}|\Omega_j|}{\sum_{e=1}^{n_e} w_{ie}|\Omega_e|}, \tag{3-1}$$

where

$$w_{ij} = \max\left(0, 1 - \frac{||\boldsymbol{x}_i^* - \boldsymbol{x}_j^*||_2}{R}\right), \tag{3-2}$$

$|\Omega_j|$ is the area of the jth element, $||\boldsymbol{x}_i^* - \boldsymbol{x}_j^*||_2$ is the distance between ith and jth element centroids ($\boldsymbol{x}_i^*$ and $\boldsymbol{x}_j^*$, respectively), and R is the filter radius.

Let us now define the vector of design variables $\mathbf{z} = \{z_e\}_{e=1}^{n_e}$, with $0 \leq z_e \leq 1$. For the interpolation functions, discussed in the following topic, `PolyTop` utilizes the vector $\mathbf{y}$ of filtered densities, which contains the values of ρ at each element centroid in the form

$$\mathbf{y} = \mathbf{P}\mathbf{z}, \tag{3-3}$$

where $\mathbf{P}$ is the filtering matrix defined in Eq. 3-1.

3.2.2
Volume and material interpolation functions

The volume interpolation function used in this work follows the same of the original `PolyTop` software, i.e., simply

$$\mathbf{V} = m_V(\mathbf{y}) = \mathbf{y}. \tag{3-4}$$

A material interpolation function according to the SIMP method used in `PolyTop` is also reproduced herein

$$\mathbf{E} = m_E(\mathbf{y}) = \epsilon + (1-\epsilon)\left[m_V(\mathbf{y})\right]^p, \tag{3-5}$$

where $\epsilon \ll 1$ is the Ersatz parameter and p is the SIMP penalization factor, usually taken around 3 for a well-defined output topology, depending on the material Poisson ratio [9]. Equation 3-5 may be adapted to incorporate other material representation methods, such as RAMP, as seen in `PolyTop` [6].

From Eqs. 2-34, 3-3 and 3-5, the final global stiffness matrix will be traditionally assembled as

$$\mathbf{K} = \sum_{e=1}^{n_e} m_E(\mathrm{y}_e)\mathbf{K}_0^e. \tag{3-6}$$

3.2.3
Problem statement

The topology optimization problem in the context of compliance minimization is stated as

$$\min_{\mathbf{z}} \quad \mathbf{F}^T\mathbf{U}, \tag{3-7}$$

$$\text{s.t.} \quad \begin{cases} \frac{\mathbf{A}^T m_V(\mathbf{y})}{\mathbf{A}^T\mathbf{1}} - \overline{V} \leq 0, \\ 0 \leq z_e \leq 1, \quad e = 1, \ldots, n_e \end{cases} \tag{3-8}$$

$$\text{with:} \quad \mathbf{K}\mathbf{U} = \mathbf{F},$$

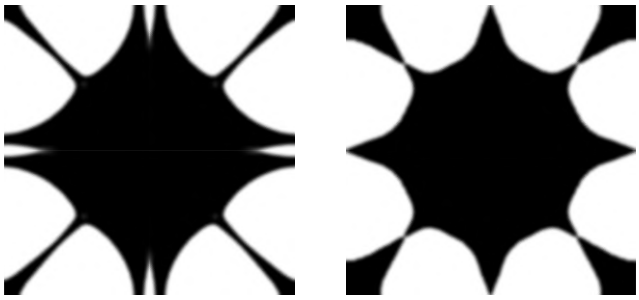
where $\mathbf{A} = \{|\Omega_e|\}_{e=1}^{n_e}$, $|\Omega_e|$ is the element volume and $\overline{V}$ is the volume constraint value. Other aspects, such as meshing, sensitivity analysis and optimization method, remain the same as in `PolyTop` [6], except for the continuation method, which will be replaced by a constant penalization value of $p = 3$.

3.2.4
Compliance minimization results

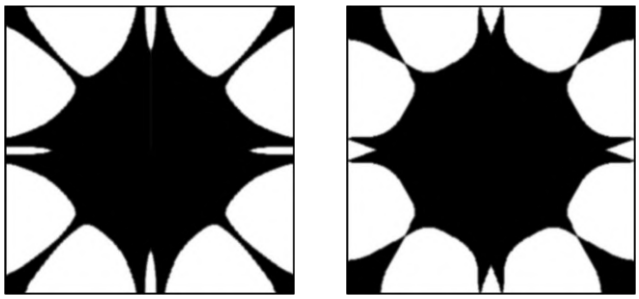
3.2.4.1
Square plates

Square plates are a classical benchmark problem in plate bending. In this Section, we will demonstrate the code efficiency by running topology optimization of simply supported and clamped plates, both on thick and thin cases. This example has been studied by [44] and [73]. The 1x1 plate has a unit load acting on its center – see Figs. 2.5(a) and 2.5(b). The Young's modulus is $E = 1{,}092{,}000$ and the Poisson ratio 0.3. The number of elements is set as 16,000 (same as in [44]), the filter radius is 0.0135, and the constraint volume fraction is 50% of the domain total volume. The Dirichlet boundary conditions apply for all plate sides, i.e., each side will be either simply supported or clamped, depending on each case. For thick approach, we set $h = 0.1$, and, for thin case, $h = 0.001$. Since the domain is symmetric, only one quarter of the plate will be considered in the optimization process.

Results are shown in Figs. 3.1(a) and 3.1(b), for simply-supported plates, and in Figs. 3.2(a) and 3.2(b), for clamped plates, where the whole plates are assembled with the four symmetric pieces. As we can see, the optimized topologies become very different when the thickness changes.

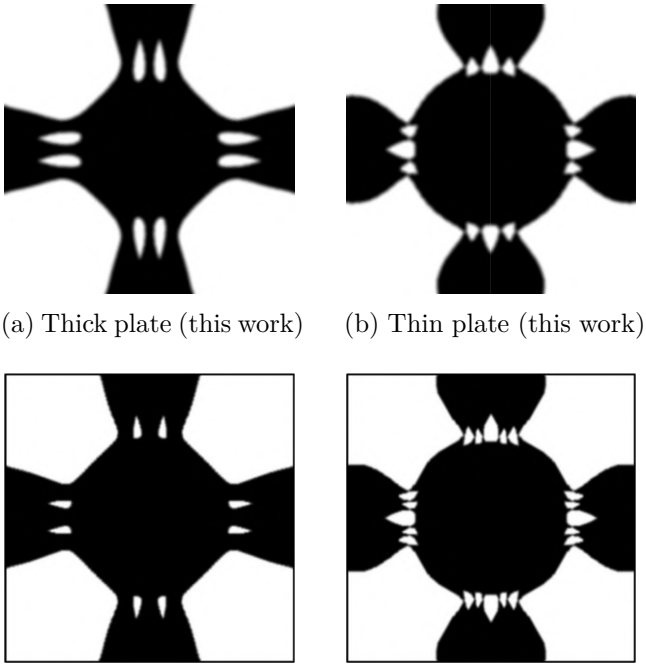


(a) Thick plate (this work) (b) Thin plate (this work)



(c) Thick plate from [44] (d) Thin plate from [44]

Figure 3.1: Simply-Supported Square Plate Topology Optimization Results.



(a) Thick plate (this work) (b) Thin plate (this work)

(c) Thick plate from [44] (d) Thin plate from [44]

Figure 3.2: Clamped Square Plate Topology Optimization Results.

Although there is a slight difference between the simply-supported thin plate topology of this work (Fig. 3.1(b)) and the one presented in [44] (Fig. 3.1(d)), we can see in Fig. 3.3 from [73] that, as the plate thickness decreases, the resulting topology approaches the topology of the present work. It is also noted that, for sufficiently thick plates, the results remain unaffected even without the application of any locking treatment.

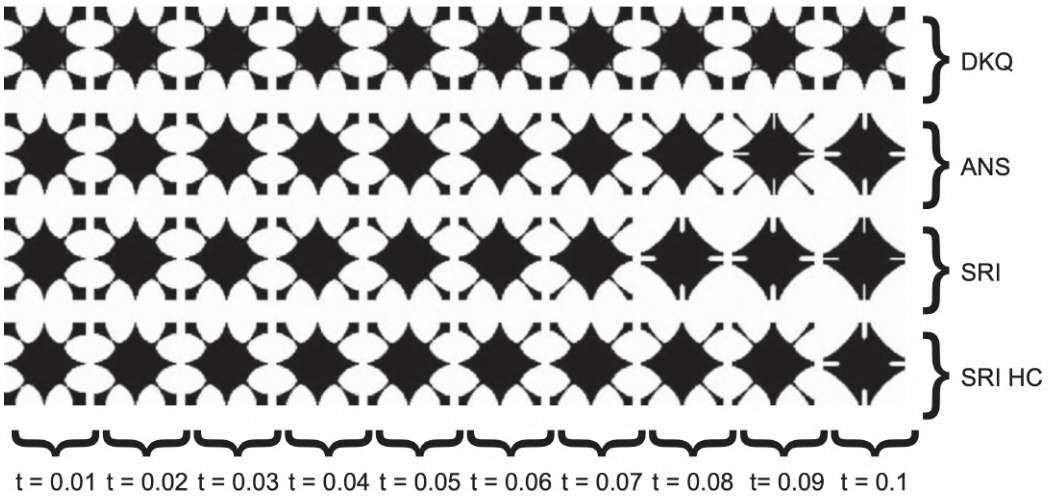
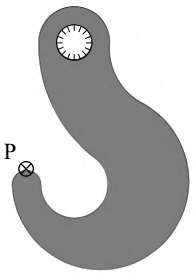


Figure 3.3: Optimal topologies for various simply-supported plate thicknesses, using different plate elements (Discrete Kirchhoff Quadrilateral - DKQ; Assumed Natural Strains - ANS; Selective Reduced Integration - SRI; SRI with Hourglass Control - SRI HC), taken from [73].

3.2.4.2 A Hook Plate

The hook example, illustrated in Fig. 3.4, was studied as a plate problem by Pham and Phan [44] and is an appropriate geometry to take advantage of the polygonal meshes' power. The domain is given in Fig. 3.4(a), where the eyelet circle is clamped and a transversal unit force acts on the hook end. The parameters remain the same as the previous examples, except for filter radius (2.0) and volume fraction (27.5%). The thickness for the thick case is $h = 25$, while $h = 0.01$ for the thin case.

Figures 3.4(b) and 3.4(c) present out results for thick and thin hook topology optimization, which are notably similar to the literature topologies (Figs. 3.4(d) and 3.4(e)). Again, the results differ from each other when distinct values of plate thickness are used.



(a) Hook plate model

(b) Thick hook plate (this work)

(c) Thin hook plate (this work)

(d) Thick hook plate from [44]

(e) Thin hook plate from [44]

Figure 3.4: Hook Plate Topology Optimization Results.

3.3 Stress-constrained volume minimization

As mentioned in Section 3.1, we propose an extension to linear elastic Reissner-Mindlin plates of the aggregation-free methodology for local stress-constrained topology optimization through the Augmented Lagrangian method by Giraldo-Londoño and Paulino [13]. The original algorithm's finite element method, optimization, and sensitivity analysis Sections are adapted to the context of plates. In this problem, the `PolyStress` [13] software is used as the basis code, where the locking-free technique previously described is implemented together with the modifications necessary to change the finite elements from membrane to plate version.

Any other detail not mentioned hereinafter is taken as unaltered from `PolyStress`. Several numerical examples are performed to investigate the methodology's efficiency and coverage. The approach is appropriate for any plate polygonal finite element in meshes made entirely of the same polygon (structured) or composed of random polygonal elements (unstructured).

3.3.1 Filter operator

As stated by Giraldo-Londoño and Paulino [13], to enforce the problem into a well-posed one, the filter operator matrix is applied according to the following [74, 75]

$$P_{ij} = \frac{w_{ij}|\Omega_j|}{\sum_{e=1}^{n_e} w_{ie}|\Omega_e|}, \tag{3-9}$$

where

$$w_{ij} = \max\left(0, 1 - \frac{||\boldsymbol{x}_i^* - \boldsymbol{x}_j^*||_2}{R}\right)^q \tag{3-10}$$

and q is a filter exponent. In case $q = 1$, the linear hat kernel will be invoked (as in Eq. 3-1), but a nonlinear kernel is preferred to stress-constrained topology optimization so that the topologies may have more precise boundaries.

3.3.2 Volume and material interpolation functions

The volume interpolation function, for a clear black-and-white definition, is the threshold projection function [76]

$$m_V(\mathrm{y}_e) = \frac{\tanh(\beta\eta) + \tanh(\beta(\mathrm{y}_e - \eta))}{\tanh(\beta\eta) + \tanh(\beta(1 - \eta))}, \tag{3-11}$$

with β and η being adjustable input parameters. The material interpolation function is according to the SIMP method, as in Eq. 3-5.

3.3.3 Polynomial vanishing constraints

The `PolyStress` code presented a variation of the traditional vanishing stress constraint [77], the so-called *polynomial vanishing constraints* [78], which takes the following form

$$g_j(\mathbf{z}, \mathbf{U}) = m_E(y_j)\Lambda_j(\Lambda_j^2 + 1) \leq 0, \tag{3-12}$$

where

$$\Lambda_j = \frac{\sigma_j^v}{\sigma_{\text{lim}}} - 1, \tag{3-13}$$

σ_j^v is accounted for according to Eq. 2-39, and σ_{lim} is the imposed stress limit. The polynomial vanishing constraints appropriately act like the traditional ones (Λ_j) when $\sigma_j^v/\sigma_{\text{lim}} \to 1$. However, when the constraint is violated ($\sigma_j^v/\sigma_{\text{lim}} \gg 1$), this function assumes a cubic behavior (Λ_j^3) that leads the optimizer to a lower overall stress solution. Giraldo-Londoño and Paulino [13] argued that the polynomial vanishing constraint drives the solution toward an overall lower stress state more quickly than the traditional one and that no struggles with the nonlinear behavior are seen. The number of stress constraints in this study is chosen to equal the number of elements in the discretized domain.

3.3.4 Topology optimization problem

The topology optimization problem considering local stress constraints can be originally stated as follows

$$\begin{aligned}
\min_{\mathbf{z}} \quad & f(\mathbf{z}) = \frac{\mathbf{A}^T m_V(\mathbf{y})}{\mathbf{A}^T \mathbf{1}}, \\
\text{s.t.} \quad & \begin{cases} g_j(\mathbf{z}, \mathbf{U}) = m_E(\mathrm{y}_j)\Lambda_j(\Lambda_j^2 + 1) \leq 0, \quad j = 1, \ldots, n_e \\ 0 \leq z_e \leq 1, \quad e = 1, \ldots, n_e \end{cases} \\
\text{with:} \quad & \mathbf{K}\mathbf{U} = \mathbf{F},
\end{aligned} \tag{3-14}$$

where $\mathbf{A} = \{|\Omega_e|\}_{e=1}^{n_e}$, m_V and m_E are respectively defined in Eqs. 3-11 and 3-5, $\mathbf{y}$ is obtained according to Eq. 3-3 (wherein $\mathbf{P}$ must now be taken from Eq. 3-9), $\mathbf{z}$ is the design variable vector, and Λ_j is defined in Eq. 3-13. The problem stated in Eq. 3-14 is valid for one load case only – for multiple load cases, refer to the work of Senhora et al. [72].

3.3.5 The Augmented Lagrangian method

As mentioned in Section 3.1.2, one of the main drawbacks of local stress constraints in topology optimization is the large number of constraints. The Augmented Lagrangian (AL) method [70, 71] is a promising method for tackling the problem. The AL method turns the original problem into a set of unconstrained optimization problems, each of them pursuing the minimization of its k^{th}-step AL function $J_{\mu^{(k)}}(\mathbf{z}, \boldsymbol{\lambda}^{(k)})$. This k^{th} sub-problem is stated as [13]

$$\min_{\mathbf{z}} \qquad J_{\mu^{(k)}}(\mathbf{z}, \boldsymbol{\lambda}^{(k)}) = f(\mathbf{z}) + \frac{1}{n_e} P^{(k)}(\mathbf{z}, \mathbf{U}), \tag{3-15}$$

where $P^{(k)}(\mathbf{z}, \mathbf{U})$ is the penalization term, expressed as

$$P^{(k)}(\mathbf{z}, \mathbf{U}) = \sum_{j=1}^{n_e} \left[\lambda_j^{(k)} h_j(\mathbf{z}, \mathbf{U}) + \frac{\mu^{(k)}}{2} h_j(\mathbf{z}, \mathbf{U})^2 \right], \tag{3-16}$$

$h_j(\mathbf{z}, \mathbf{U})$ are the equality constraints

$$h_j(\mathbf{z}, \mathbf{U}) = \max \left[g_j(\mathbf{z}, \mathbf{U}), -\frac{\lambda_j^{(k)}}{\mu^{(k)}} \right], \tag{3-17}$$

$\boldsymbol{\lambda}^{(k)} = \{\lambda_j^{(k)}\}_{j=1}^{n_e}$ is the Lagrange multiplier estimator vector, and $\mu^{(k)} > 0$ is a quadratic penalty factor, both updated as

$$\lambda_j^{(k+1)} = \lambda_j^{(k)} + \mu^{(k)} h_j(\mathbf{z}^{(k)}, \mathbf{U}), \tag{3-18}$$

$$\mu^{(k+1)} = \min \left[\alpha \mu^{(k)}, \mu_{\max} \right], \tag{3-19}$$

where $\alpha > 1$ and $\mu_{\max}$ are an update factor and an upper bound to avoid numerical instabilities.

The normalization factor $1/n_e$ in the second term of the right-hand side of Eq. 3-15 avoids numerical instabilities for large-scale problems, otherwise $P^{(k)}$ would govern over $f(\mathbf{z})$ as n_e becomes large. A pseudo-code is provided in Algorithm 1, where a version of the Method of Moving Asymptotes (MMA) [79] is employed to solve each AL sub-problem. In Algorithm 1, `Tol` and `TolS` are input tolerances for change in the design variables and stress constraints, respectively, while $\mathbf{z}_{i+1}^{(k)}$ and $\mathbf{z}_i^{(k)}$ are consecutive MMA solution vectors in a k^{th} AL sub-problem. The process runs over two main loops, both limited to prescribed maximum numbers of iterations, i.e., `MaxIter` for the outer and `MMA_Iter` for the inner loop. Moreover, `PolyStress` adopts a continuation of the penalization parameter β of the threshold projection function in Eq. 3-11. For further comments on the initial parameter calibration, the reader is referred to the original `PolyStress` paper [13].

Algorithm 1 AL-based stress-constrained topology optimization [13]

Read input parameters
Initialize $k = 0$, $\boldsymbol{\lambda}^{(0)}$, $\mu^{(0)}$
while $(k < \texttt{MaxIter})$ **and** $\left[\frac{1}{n_e}\text{sum}\left|\mathbf{z}_{i+1}^{(k)} - \mathbf{z}_i^{(k)}\right| > \texttt{Tol}\ \textbf{or}\ \max\left(\sigma_e^v/\sigma_{\text{lim}}\right) - 1 > \texttt{TolS}\right]$ **do**
    $k = k + 1$
    for $j = [1, \ldots, \texttt{MMA\_Iter}]$ **do**
        Compute AL function according to Eq. 3-15
        Use MMA to find the new $\mathbf{z}^{(k)}$
        Solve finite element equations based on new $\mathbf{z}^{(k)}$
        Compute σ_e^v based on new $\mathbf{z}^{(k)}$
        if $\left[\frac{1}{n_e}\text{sum}\left|\mathbf{z}_{i+1}^{(k)} - \mathbf{z}_i^{(k)}\right| \leq \texttt{Tol}\right]$ **and** $\left[\max\left(\frac{\sigma_e^v}{\sigma_{\text{lim}}}\right) - 1 \leq \texttt{TolS}\right]$ **then**
            break
        end if
    end for
    Update $\boldsymbol{\lambda}^{(k+1)}$ according to Eq. 3-18
    Update $\mu^{(k+1)}$ according to Eq. 3-19
    Run continuation of β in Eq. 3-11
end while
Output: final optimized topology Ω^*

3.3.6 Sensitivity analysis

Although the present sensitivity analysis for plates has only a tiny different detail (in Eq. 3-30) from the one in `PolyStress`, we will reproduce the whole process for a better understanding. Since our optimization algorithm is gradient-based, we need the sensitivity of the AL function in Eq. 3-15 to guide the problem in Eq. 3-14 towards minimization. Therefore, we use the chain rule

$$\begin{aligned}\frac{\mathrm{d}J^{(k)}}{\mathrm{d}\mathbf{z}} &= \frac{\partial \mathbf{E}}{\partial \mathbf{z}}\frac{\mathrm{d}J^{(k)}}{\mathrm{d}\mathbf{E}} + \frac{\partial \mathbf{V}}{\partial \mathbf{z}}\frac{\mathrm{d}J^{(k)}}{\mathrm{d}\mathbf{V}}, \\ &= \frac{\partial \mathbf{E}}{\partial \mathbf{z}}\left(\frac{\partial f}{\partial \mathbf{E}} + \frac{1}{n_e}\frac{\partial P^{(k)}}{\partial \mathbf{E}}\right) + \frac{\partial \mathbf{V}}{\partial \mathbf{z}}\left(\frac{\partial f}{\partial \mathbf{V}} + \frac{1}{n_e}\frac{\partial P^{(k)}}{\partial \mathbf{V}}\right),\end{aligned} \tag{3-20}$$

where $\mathbf{E} = m_E(\mathbf{y})$ and $\mathbf{V} = m_V(\mathbf{y})$. Then, depicting each term from the above equation

$$\frac{\partial \mathbf{E}}{\partial \mathbf{z}} = \mathbf{P}^T J_{m_E}(\mathbf{y}) \quad \text{and} \quad \frac{\partial \mathbf{V}}{\partial \mathbf{z}} = \mathbf{P}^T J_{m_V}(\mathbf{y}), \tag{3-21}$$

with $J_{m_E} = \text{diag}\left(m_E'(\mathrm{y}_1), \ldots, m_E'(\mathrm{y}_{n_e})\right)$ and $J_{m_V} = \text{diag}\left(m_V'(\mathrm{y}_1), \ldots, m_V'(\mathrm{y}_{n_e})\right)$. Based on Eqs. 3-14 and 3-16, we also verify that

$$\frac{\partial f}{\partial E_l} = 0 \quad \text{and} \quad \frac{\partial f}{\partial V_l} = \frac{A_l}{\mathbf{A}^T\mathbf{1}}, \tag{3-22}$$

$$\frac{\partial P^{(k)}}{\partial V_l} = 0. \tag{3-23}$$

The term $\partial P^{(k)}/\partial E_l$ is expressed as

$$\frac{\partial P^{(k)}}{\partial E_l} = \sum_{j=1}^{n_e} \left[\lambda_j^{(k)} + \mu^{(k)} h_j(\mathbf{z}, \mathbf{U})\right] \left[\frac{\partial h_j(\mathbf{z}, \mathbf{U})}{\partial E_l} + \frac{\partial h_j(\mathbf{z}, \mathbf{U})}{\partial \mathbf{U}} \frac{\partial \mathbf{U}}{\partial E_l}\right]. \tag{3-24}$$

We use the adjoint method to properly find $\partial \mathbf{U}/\partial E_l$ without high computational cost [80]. From Eq. 3-14

$$\frac{\mathrm{d}\mathbf{K}}{\mathrm{d}E_l}\mathbf{U} + \mathbf{K}\frac{\mathrm{d}\mathbf{U}}{\mathrm{d}E_l} = \mathbf{0}. \tag{3-25}$$

We may write

$$\boldsymbol{\xi}^T \left(\frac{\mathrm{d}\mathbf{K}}{\mathrm{d}E_l}\mathbf{U} + \mathbf{K}\frac{\mathrm{d}\mathbf{U}}{\mathrm{d}E_l}\right) = \mathbf{0}, \tag{3-26}$$

with $\boldsymbol{\xi}$ being the adjoint vector variable, so that we may add this zero term into Eq. 3-24 to obtain

$$\begin{aligned}\frac{\partial P^{(k)}}{\partial E_l} =& \sum_{j=1}^{n_e} \left[\lambda_j^{(k)} + \mu^{(k)} h_j(\mathbf{z}, \mathbf{U})\right] \left[\frac{\partial h_j(\mathbf{z}, \mathbf{U})}{\partial E_l} + \frac{\partial h_j(\mathbf{z}, \mathbf{U})}{\partial \mathbf{U}} \frac{\partial \mathbf{U}}{\partial E_l}\right] \\ &+ \boldsymbol{\xi}^T \left(\frac{\mathrm{d}\mathbf{K}}{\mathrm{d}E_l}\mathbf{U} + \mathbf{K}\frac{\mathrm{d}\mathbf{U}}{\mathrm{d}E_l}\right).\end{aligned} \tag{3-27}$$

Now, we choose $\boldsymbol{\xi}$ such that all terms multiplying $\partial \mathbf{U}/\partial E_l$ vanish from Eq. 3-27 and the following less-expensive adjoint problem emerges

$$\mathbf{K}\boldsymbol{\xi} = -\sum_{j=1}^{n_e} \left[\lambda_j^{(k)} + \mu^{(k)} h_j(\mathbf{z}, \mathbf{U})\right] \frac{\partial h_j(\mathbf{z}, \mathbf{U})}{\partial \mathbf{U}}. \tag{3-28}$$

According to Eq. 3-17, $\partial h_j(\mathbf{z}, \mathbf{U})/\partial \mathbf{U} = \mathbf{0}$ when $g_j(\mathbf{z}) < -\lambda_j^{(k)}/\mu^{(k)}$ and $\partial h_j(\mathbf{z}, \mathbf{U})/\partial \mathbf{U} = \partial g_j(\mathbf{z}, \mathbf{U})/\partial \mathbf{U}$ otherwise, i.e.

$$\frac{\partial h_j}{\partial \mathbf{U}} = \frac{\partial g_j}{\partial \sigma_j^v} \frac{\partial \sigma_j^v}{\partial \boldsymbol{\sigma}_j} \frac{\partial \boldsymbol{\sigma}_j}{\partial \mathbf{U}}. \tag{3-29}$$

The terms $\partial g_j/\partial \sigma_j^v$ and $\partial \sigma_j^v/\partial \boldsymbol{\sigma}_j$ are straightforwardly obtained from Eqs. 3-12 and 2-39, respectively. The last term is computed as

$$\frac{\partial \boldsymbol{\sigma}_j}{\partial \mathbf{U}} = \frac{\partial \boldsymbol{\sigma}_j}{\partial \boldsymbol{\varepsilon}^j} \frac{\partial \boldsymbol{\varepsilon}^j}{\partial \mathbf{U}} = z|_{h/2}\, \mathbf{D}\mathbf{B}^{\mathbf{b},\mathbf{e}}, \tag{3-30}$$

from which the adjoint vector can be found. Finally, the penalization term derivative becomes

$$\frac{\partial P^{(k)}}{\partial E_l} = \sum_{j=1}^{n_e} \left[\lambda_j^{(k)} + \mu^{(k)} h_j(\mathbf{z}, \mathbf{U})\right] \frac{\partial h_j(\mathbf{z}, \mathbf{U})}{\partial E_l} + \boldsymbol{\xi}^T \frac{\mathrm{d}\mathbf{K}}{\mathrm{d}E_l}\mathbf{U}, \tag{3-31}$$

where $\partial h_j(\mathbf{z}, \mathbf{U})/\partial E_l$ is obtained directly from Eqs. 3-17 and 3-12, while $\partial \mathbf{K}/\partial E_l$ from Eq. 3-6.

3.3.7
Stress-constrained volume minimization results

Several numerical experiments are conducted to investigate different aspects of the proposed methodology. In addition to demonstrating the algorithm's efficiency and scalability, we employ arbitrary polygonal and rectangular meshes, analyze both thick and thin plate scenarios, compare results from local stress-constrained topology optimization with compliance optimization, and utilize complex geometries to emphasize the capabilities of `PolyMesher` [7].

A set of relevant optimization parameters is outlined in `PolyStress` [13], with minor adjustments made in this study — see Table 3.1. These parameters are maintained across all examples unless otherwise indicated. Some of the following examples are roughly replicated from existing literature; however, they may not correspond to the exact same cases due to variations in mesh types and sizes.

Table 3.1: Input optimization parameters from `PolyStress` [13]

Parameter	**Value**
Initial Lagrange multiplier estimators - $\boldsymbol{\lambda}^{(0)}$	$\mathbf{0}$
Initial penalty factor - $\mu^{(0)}$	10
Maximum penalty factor - $\mu_{\max}$	10,000
Penalty factor update parameter - α	1.10
SIMP penalization factor - p	3.5
Nonlinear filter exponent - q	3
Ersatz parameter - ϵ	10^{-8}
MMA iterations per AL step - `MMA_Iter`	5
Initial threshold projection penalization factor[a] - β	1
Maximum threshold projection penalization factor[a] - $\beta_{\max}$	10
Threshold projection density - η	0.5
Initial guess - $\mathbf{z}^{(0)}$	0.5
Convergence tolerance on design variables - `Tol`	0.0002
Convergence tolerance on stress constraints - `TolS`	0.003
Maximum number of AL steps - `MaxIter`	150

[a] β starts as 1 and increases by 1 every 5 AL iterations up to its maximum value $\beta_{\max}$.

Additionally, the von Mises stress maps are processed using a standard normalization technique, i.e.

$$\tilde{\sigma}_e^v = E_e \frac{\sigma_e^v}{\sigma_{\text{lim}}}. \tag{3-32}$$

3.3.7.1 Cantilever plates

The initial examples consist of rectangular cantilever plates fixed along one edge and subjected to a uniformly distributed transverse load on the opposite edge. Two cases are replicated from the literature, differing only in their aspect ratio.

***(i)* A long-and-narrow cantilever plate**

Figure 3.5 summarizes the example of Goo et al. [56], where a force of 0.1 N is uniformly distributed across three central nodes to mitigate stress concentration. The boundary conditions restrict all degrees of freedom at the clamped edge. To replicate the original experiment, Young's modulus is set at 1 GPa, and the Poisson's ratio is 0.3. The plate dimensions are 60×20 mm with a thickness of 0.5 mm, qualifying it as a thin plate. A mesh consisting of 10,092 traditional Q4 elements[1] is generated to implement the proposed methodology. In this study, we will vary the thickness of the cantilever plate examples. To obtain significant stress values, we propose an increased force of 10 N for a thickness of 10 mm (thick case). A stress limit of 16 MPa is imposed for both thickness configurations, and a filter radius of 5 mm is employed.

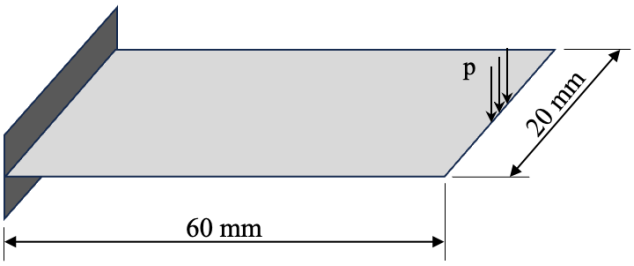
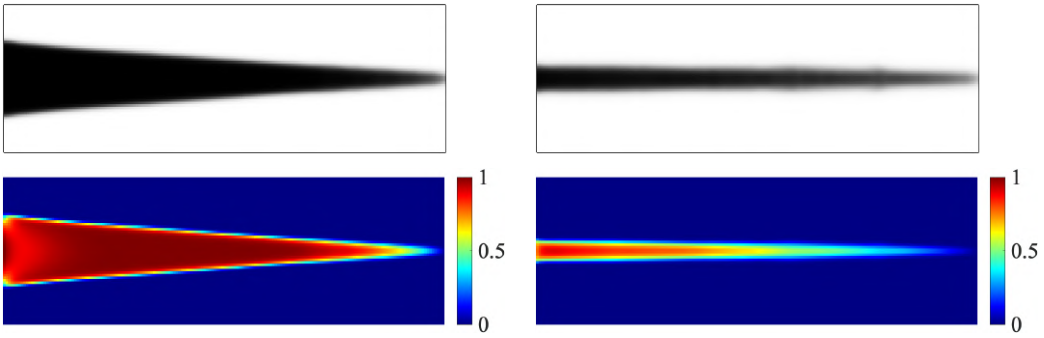


Figure 3.5: Long-and-narrow cantilever plate example as presented by Goo et al. [56].

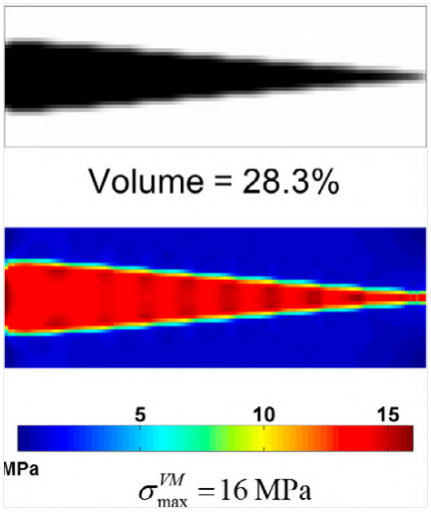
Figure 3.6(a) displays the resulting topology for the thin case, consistent with the results of Goo et al. [56] (see Fig. 3.6(c)). Aggregation methods, particularly those utilizing the p norm, require careful selection of appropriate p values, as variations can significantly impact the final topology. The present method yields a unique output, although certain key factors, such as mesh size, may influence the results from other methods. The normalized von Mises

[1]Although the quadrilateral element by the polygonal framework still has four nodes (a linear element), the `PolyTop` algorithm will numerically integrate the element stiffness matrix with 3×4 Gauss points per element.

stress map indicates that the stress constraints are satisfied throughout the domain. A final material volume of approximately 26.5% of the total domain is achieved. For the thick case, illustrated in Figure 3.6(b), a similar topology is obtained, albeit with a reduced material volume of about 11.4% of the domain total volume. Thanks to the locking-free technique, no additional modifications are required to execute thin or thick cases. It is important to note that Goo et al. [56] conducted this experiment solely for a thin plate. For both thickness configurations, the objective function (material volume fraction) and the maximum normalized von Mises stress are plotted across each AL iteration, as shown in Figure 3.7. The results indicate that the initial topology (linked to the initial guess) already satisfies the stress criterion in both cases. However, optimization may still be advantageous for conducting a trade-off analysis between minimizing material volume and stress constraint conformity.

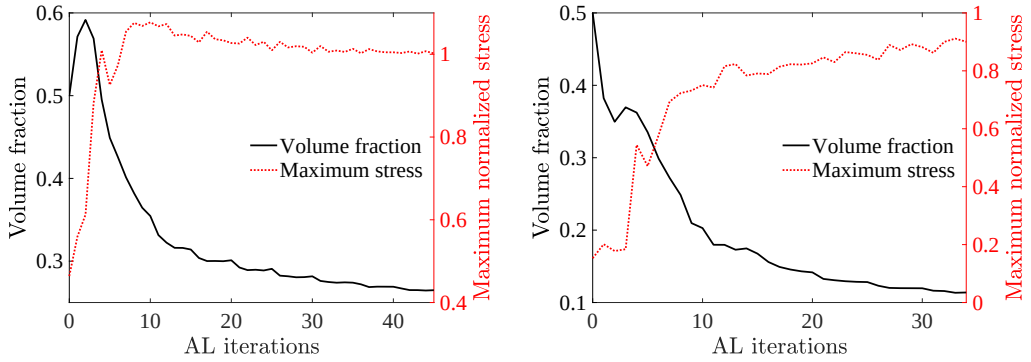


(a) Thin case (b) Thick case



(c) Reference for thin case [56]

Figure 3.6: Long-and-narrow cantilever plate – Final topologies for thin (a) and thick (b) cases, with the corresponding normalized von Mises stress maps; Goo et al. [56] results for the thin case in (c), with stress scale not normalized.



(a) Thin case (b) Thick case

Figure 3.7: Long-and-narrow cantilever plate – Evolution of material volume fraction and maximum normalized von Mises stress for both thickness cases.

(ii) A short-and-wide cantilever plate

The second cantilever plate example, proposed by Liu et al. [57], is similar to the previous one but features dimensions of 30×90 mm and a thickness of 0.5 mm (thin case) — refer to Fig. 3.8. A force of 25 N is uniformly distributed across six neighboring elements. The material properties are defined as $E = 210$ GPa and $\nu = 0.3$. Additionally, we present a thicker plate variation for this example, with a thickness of 5 mm and a total applied force of 250 N. For both cases, the stress limit is set at 358 MPa, and the filter radius at 5 mm. A mesh of 10,092 Q4 elements was used once again.

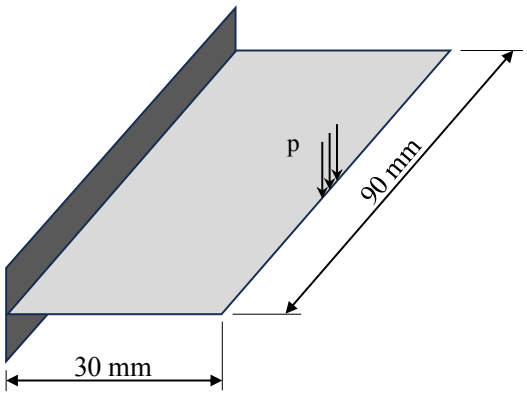
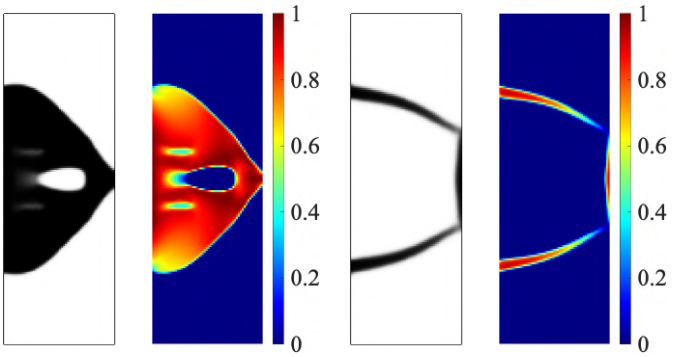
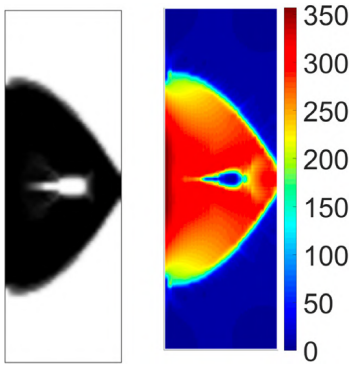


Figure 3.8: Short-and-wide cantilever plate example as proposed by Liu et al. [57].

The results depicted in Figs. 3.9(a) and 3.9(b) demonstrate clear topologies. In the thin case, the results are in good agreement with those reported by Liu et al. [57] (Fig. 3.9(c)). Figure 3.10 illustrates the evolution of material volume fraction and maximum normalized von Mises stress throughout the optimization process. Final material volumes of approximately 36.5% and 6.8% of the domain total values are achieved for thin and thick cases, respectively.

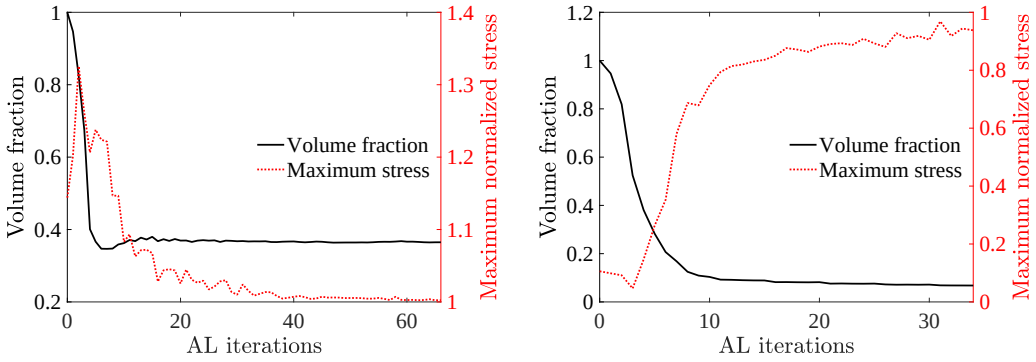


(a) Thin case (b) Thick case



(c) Reference for thin case [57]

Figure 3.9: Short-and-wide cantilever plate – Final topologies for thin (a) and thick (b) cases, with the corresponding normalized von Mises stress maps; Liu et al. [57] results for the thin case in (c), with stress scale not normalized.



(a) Thin case (b) Thick case

Figure 3.10: Short-and-wide cantilever plate – Evolution of material volume fraction and maximum normalized von Mises stress for both thickness.

3.3.7.2 T-shaped plate

Similar to the L-shaped beam or Corbel design benchmarks for membrane stress-constrained topology optimization, the T-shaped plate example features a re-entrant corner geometry that induces stress concentrations, which must be mitigated. This example was investigated by Goo et al. [56] and Ho-Nguyen-Tan and Kim [45] in the context of thin plates. We adopted the parameters used by Goo et al. [56], namely Young's modulus of 1 GPa, Poisson's ratio of 0.3, and thickness of 0.5 mm. A force of 0.1 N was uniformly distributed across three central nodes to minimize stress concentration effects. The boundary conditions, dimensions, and force location are shown in Fig. 3.11. A mesh with 40,000 Q4 elements, a filter radius of 4.0 mm, and an initial guess of $z_e^{(0)} = 0.25$ were employed. We replicated the study by Goo et al. [56], applying stress limits of 12, 20, and 30 GPa.

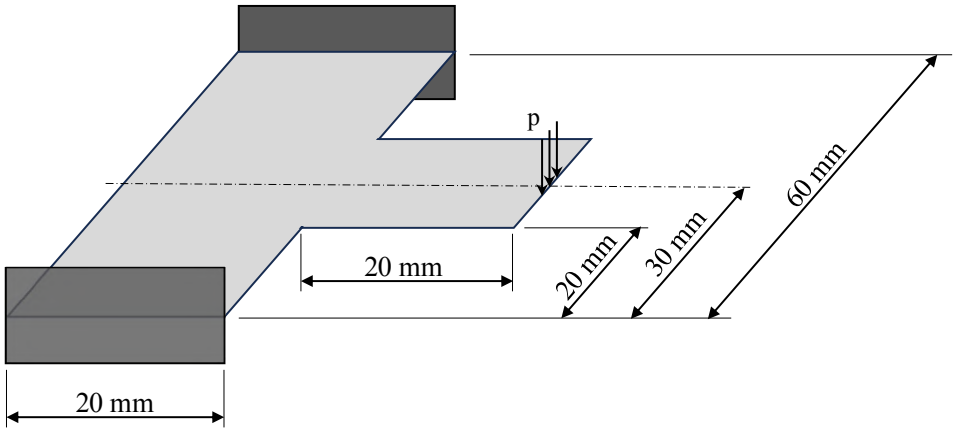
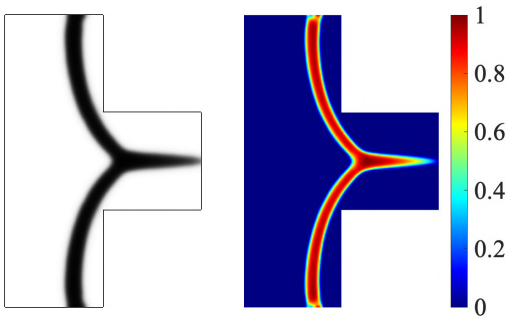
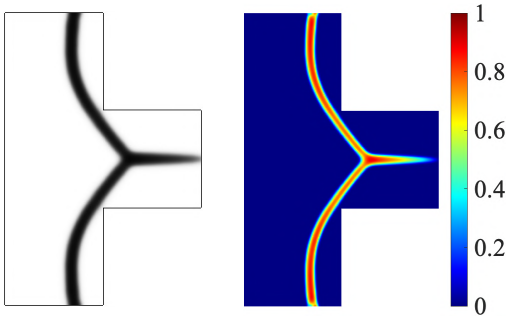


Figure 3.11: T-shaped plate example by Goo et al. [56].

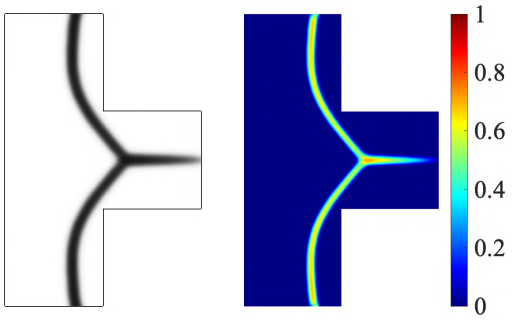
Figure 3.12 demonstrates that the final topologies successfully avoid replicating the re-entrant corner present in the initial domain. These results, which are consistent with those from the literature [56, 45] (Fig. 3.13), reveal the expected trend of decreasing the characteristic width of the final plate topologies as the stress limit increases. Specifically, the final material volume fractions are 15.8% (Fig. 3.12(a)), 11.3% (Fig. 3.12(b)) and 9.9% (Fig. 3.12(c)).



(a) $\sigma_{\text{lim}} = 12\,MPa$



(b) $\sigma_{\text{lim}} = 20\,MPa$



(c) $\sigma_{\text{lim}} = 30\,MPa$

Figure 3.12: Final topologies of the T-shaped plate for various stress limits, along with the corresponding normalized von Mises stress maps.

Algorithm's scalability assessment

We use the T-shaped plate example with $\sigma_{\text{lim}} = 12\,MPa$ to evaluate the scalability of the proposed algorithm.[2] In this evaluation, we consider five different meshes, all configured with the same parameters as previously described for this example, except for the distribution of the transversal load, the convergence tolerance for the design variables (`Tol` $= 10^{-15}$), and the maximum number of Augmented Lagrangian (AL) steps (`MaxIter` $= 55$). The load is applied to nodes within a fixed distance of 0.8 mm from the edge-midpoint neighborhood. We significantly reduce the `Tol` parameter and adjust

[2]This experiment was conducted using Matlab 2024a (Linux) on a computer with a Dual Xeon E5 2640 V4, 256 GB RAM, and an onboard video running on a CentOS 7 64-bit operating system.

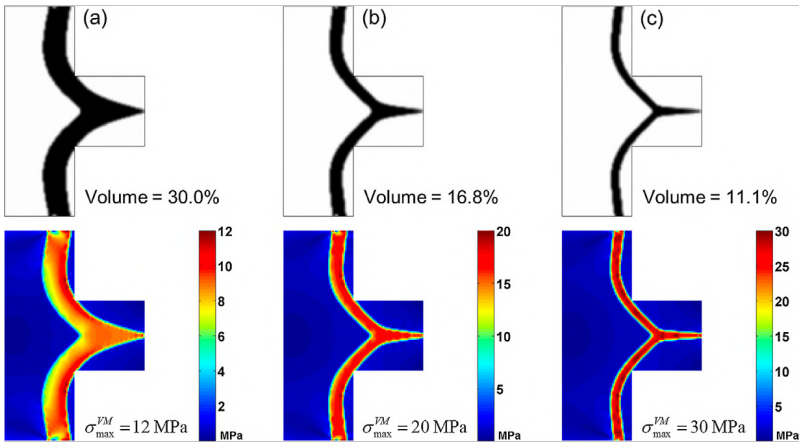


Figure 3.13: Results of Goo et al. [56] for the T-shaped plate example (stress scale is not normalized).

the `MaxIter` parameter to force the algorithm to terminate after a maximum number of iterations instead of based on the tolerance criterion. The maximum number of AL iterations was previously determined to ensure that each mesh converges at the usual tolerance value, i.e., `Tol` $= 0.0002$, as shown in Table 3.1. This approach allows us to evaluate different mesh sizes using a standardized comparison method. The meshes consist of 10,000; 50,176; 99,856; 200,704; and 300,304 Q4 elements, which correspond to the number of local stress constraints imposed.

The normalized von Mises stress maps for each mesh are depicted in Fig. 3.14, along with a plot of degrees of freedom (DOF) versus time. A brief time breakdown of the time required is provided in Table 3.2, where times are rounded to one decimal place and presented in seconds. The *Precomputations* time includes mesh generation, computation of the filter matrix, initial finite element calculation, assembly procedures, and other preliminary steps before running Algorithm 1. *Optimization* time refers to the duration Algorithm 1 takes to complete 55 AL iterations (or 275 MMA iterations). The *Others* time accounts for additional minor tasks, such as data recording and plotting. As shown, the contribution of *Precomputations* time (as a percentage of the *Total* time) increases with mesh size, while the *Others* time decreases, as the *Optimization* time remains dominant for all mesh sizes. Similar to how the AL algorithm for membrane and solid finite elements has been shown to be scalable by Senhora et al. and Giraldo-Londoño and Paulino [72, 13], this study demonstrates that the proposed methodology for plate elements is also computationally feasible. However, it is important to note that the times presented in Table 3.2 could be further reduced. One way to achieve this would be by using a filter radius that adapts to the mesh size so that each element considers a fixed number of neighboring elements at each iteration. To obtain

similar topologies, we kept the filter radius constant at 4 mm for all mesh sizes, which led to an increasing number of neighboring elements being considered, thus resulting in longer computational times. It is also worth noting that we intentionally exceeded the usual convergence number of iterations associated with the standard `Tol` parameter (as shown in Table 3.1), which would have resulted in significantly faster runtimes (e.g., the total time for the largest mesh with the standard tolerance was 2983.4 seconds).

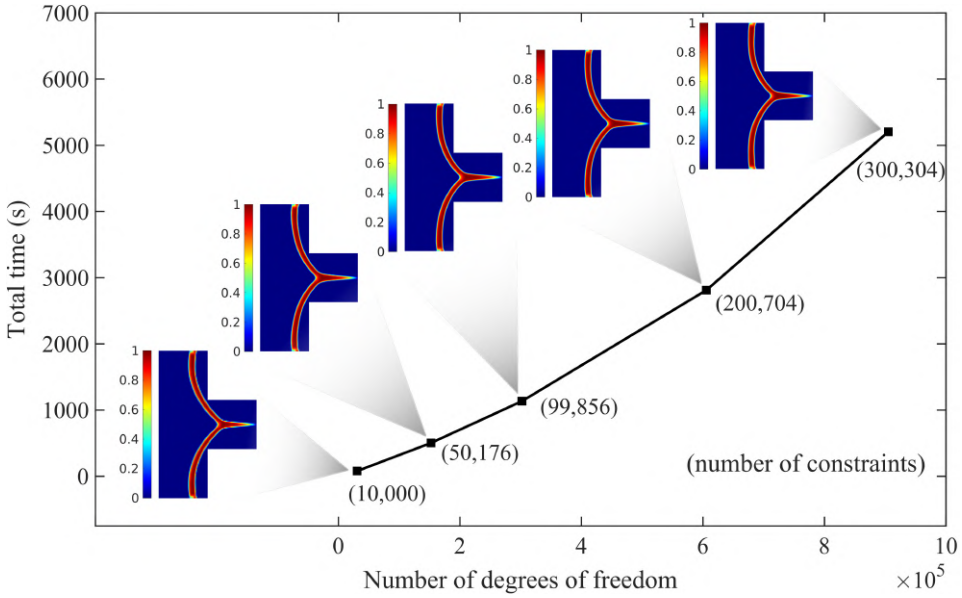


Figure 3.14: Scalability assessment – resulting normalized von Mises stress maps for the T-shaped plate example with $\sigma_{\text{lim}} = 12\,MPa$ considering different mesh sizes (number of constraints in parentheses).

Table 3.2: Scalability assessment – time breakdown (in seconds) for the T-shaped plate example with $\sigma_{\text{lim}} = 12\,MPa$ considering different mesh sizes.

Number of DOF	**30,753**	**152,211**	**301,941**	**605,475**	**905,025**
Precomputations	9.2 (11.5%)	60.4 (12.0%)	109.2 (9.6%)	296.6 (10.5%)	652.0 (12.5%)
Optimization	63.0 (79.2%)	435.3 (86.5%)	1014.0 (89.4%)	2492.5 (88.7%)	4520.6 (86.9%)
Others	7.4 (9.3%)	7.5 (1.5%)	11.5 (1.0%)	22.3 (0.8%)	31.1 (0.6%)
Total	79.5	503.2	1134.7	2811.4	5203.6

3.3.7.3 Hook plate domain

This example is again an adaptation of the hook domain, initially introduced by Talichi et al. [6] for plane stress application. Pham and Phan [44] modified this numerical experiment for plate applications, where the hook's eyelet is fully clamped, and a transverse concentrated load is applied. This plate version of the hook example was previously only considered for compliance minimization, as validated herein in Section 3.2.4.2. Then, we extend the experiment to a stress-constrained approach. The only minor modification is the transverse load redistribution along three neighboring nodes centered at the original location shown in Fig. 3.15 to avoid stress concentration singularities. To replicate the examples provided by Pham and Phan [44], we use the same parameters: Young's modulus of 1,092,000 and Poisson's ratio of 0.3. The total applied force is 1, and the thicknesses are 0.01 for the thin case and 25 for the thick case. The filter exponent is set to $q = 1$, and the SIMP penalization factor is $p = 3$. The dimensions are based on the work by Talischi et al. [6]. Stress limits are chosen to ensure that the final volume fractions obtained will generate similar topologies to those by Pham and Phan [44], where these fractions will be used as input parameters for the compliance minimization problem. The selected stress limits are 250,000 for the thin case and 0.045 for the thick case. A polygonal mesh with 16,000 elements was generated using `PolyMesher` [7], and a filter radius of 2.0 is applied.

The compliance minimization problem is conducted to replicate the results from Pham and Phan [44] and to compare them with the outcomes from the present method. The settings for the compliance problem differ from Section 3.2.4.2 and are established as follows. The volume constraints are set at approximately 0.281 for the thin case and 0.262 for the thick case, corresponding to the final volume fractions obtained from the stress-constrained approach. The implementation of the volume and material interpolation functions follows Eqs. 3-11 and 3-5, respectively, with a maximum of 150 MMA iterations. The threshold projection continuation on β follows the same framework used in this study, except for the update frequency, which is adjusted to 25 MMA iterations. This adjustment is based on the 5 MMA iterations per 5 AL-iteration frequency specified in Table 3.1. All other parameters were kept consistent with those used in the stress-constrained problem.

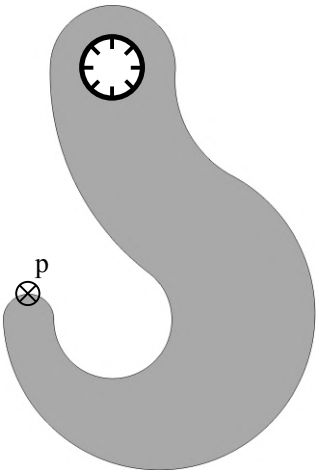
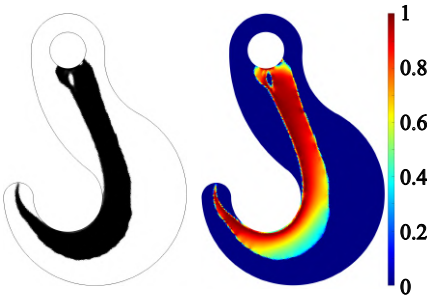
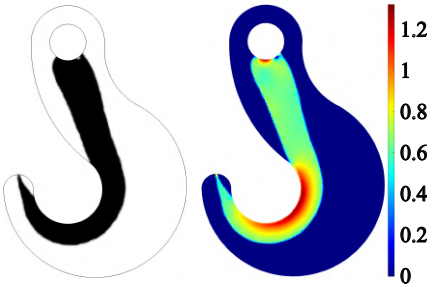


Figure 3.15: Hook plate example.

Figure 3.16 depicts the results for the thin hook plate under stress-constrained optimization (Fig. 3.16(a)) and compliance minimization (Fig. 3.16(b)), with the latter showing again good agreement with the results from Pham and Phan [44] (Fig. 3.4). Both topologies are similar, with final volume fractions approximately equal at 28.1%; however, even a minor disparity can result in stresses above the threshold. The final topologies for the thick hook plate are shown in Fig. 3.17, where the compliance result (Fig. 3.17(b)) also aligns with those from Pham and Phan [44] (Fig. 3.4). In contrast to the thin case, noticeable shape differences appear between the two methods, with the compliance output exceeding the stress limit by approximately a factor of two. It can be suggested that the topologies from the stress-constrained approach could be enhanced through the calibration of certain optimization parameters. However, we limited the parameters to ensure the reproduction of topologies similar to those in the compliance counterpart from the literature.

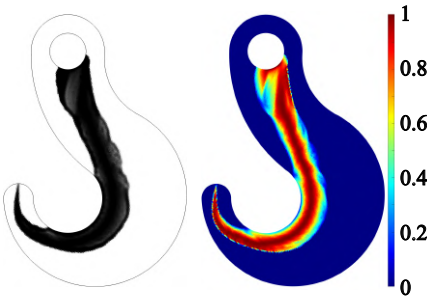


(a) Stress-constrained TO

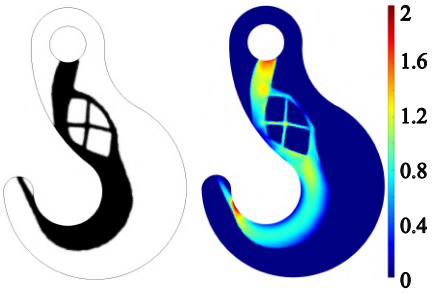


(b) Compliance TO

Figure 3.16: Final topologies of the *thin* hook plate example along with the corresponding normalized von Mises stress maps.



(a) Stress-constrained TO



(b) Compliance TO

Figure 3.17: Final topologies of the *thick* hook plate example along with the corresponding normalized von Mises stress maps.

3.3.7.4
Plate with curved boundaries

The final example consists of a flat, thin plate with curved boundaries, as proposed by Liu et al. [57], in which bilinear and biquadratic NURBS isogeometric elements were employed. The shape and dimensions of the plate are illustrated in Fig. 3.18, which considers two separated transversal load locations, A and B, while the opposite straight edge is fully clamped. Although the initial design used Non-Uniform Rational B-Splines (NURBS), we approximate the curved boundaries through image processing and point acquisition techniques. The parameters adopted are consistent with those from the literature: a Young's modulus of 1 GPa, a Poisson ratio of 0.3, a plate thickness of 0.5 mm, and a stress limit of 90 MPa. A total transverse force of 1 N is distributed over six neighborhood elements (regions A or B) to reduce stress concentrations, as highlighted in Fig. 3.18. The convergence tolerance for the design variables is set to $\texttt{Tol} = 0.002$; the filter radius is 10 mm, and the initial guess is $z_e^{(0)} = 0.7$. The mesh was generated using `PolyMesher`, consisting of 16,000 arbitrary polygonal elements.

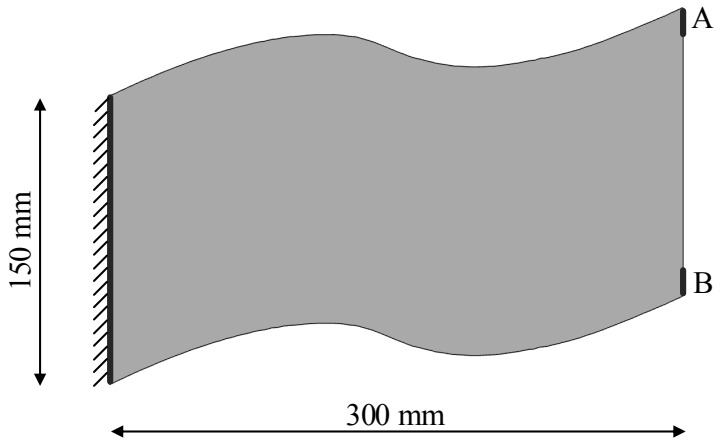
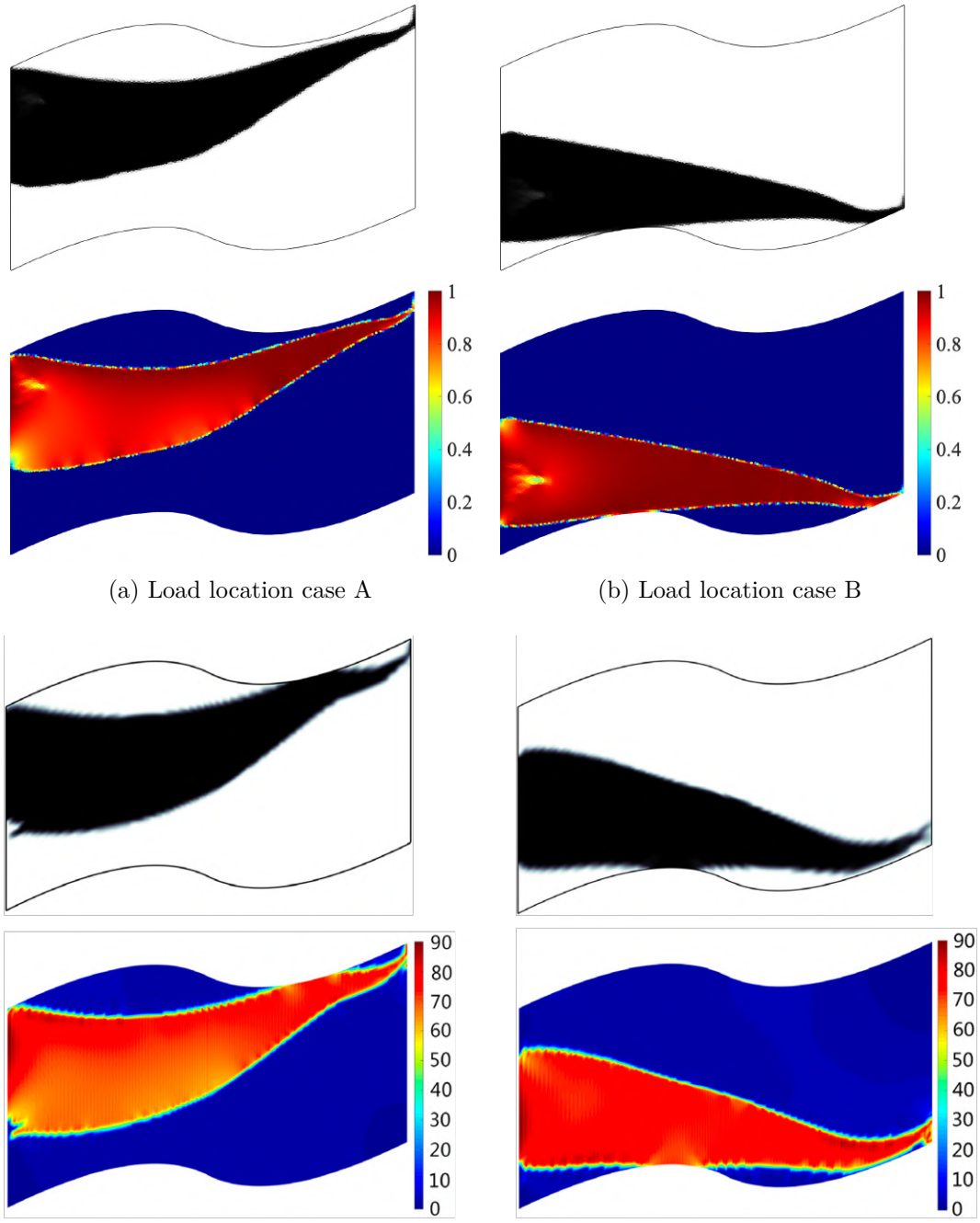


Figure 3.18: Plate with curved boundaries example.

Figures 3.19(a) (load location case A) and 3.19(b) (load location case B) present results in good agreement with those of Liu et al. [57] (Fig. 3.19(c)), despite the geometry being designed using approximation methods. Case B appears to be more sensitive to minor variations in parameters. The final material volume fractions are approximately 32.8% for case A and 28.3% for case B, with maximum von Mises stresses remaining within the acceptable range for both cases.



(a) Load location case A

(b) Load location case B

(c) Reference results [57] for load locations A (left) and B (right)

Figure 3.19: Final topologies of the plate with curved boundaries by Liu et al. [57] for load location A (a) and B (b), with respective normalized von Mises stress maps; Liu et al. [57] results for both load cases in (c), with stress scale not normalized.

4
Finite Element Method for shell structures

4.1
Literature review

Shell methodologies are quite as recent as the plate proposals for arbitrary polygonal elements, but the modeling of three-dimensional shell structures may be more involved depending on the approach. Two alternative ways to model shell structures prevail in the finite element literature: the flat and curved approaches. The flat shell element simply consists of superposing the degrees of freedom of the classic membrane and plate elements, plus an optional drilling (normal) rotation. Once there are properly established locking-free polygonal membrane and plate elements, the flat shell approach shall be straightforward. However, according to Zienkiewicz et al. [25], only flat quadrilateral and triangular meshes can model cylindrical shapes, while spherical or any other doubly-curved shells are only feasible to be discretized by flat triangles. Then, an alternative solution for polygonal meshes can be the curved element approach, which is indeed more complex than the flat option. These curved shell elements are based on a degeneration of the 3D solid finite element and must also be adjusted to circumvent shear and membrane locking behaviors.

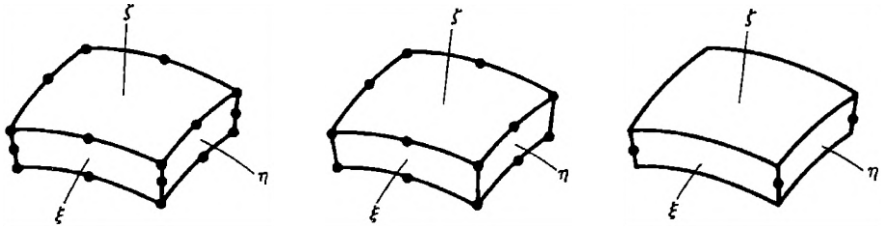
If one tries to transform an arbitrary two-dimensional polygonal mesh, like `PolyMesher` [7], into a shell surface, it cannot be guaranteed that the elements will remain flat – in fact, they will certainly be warped. Although recognizing this warping effect, Wu et al. [81] proposed a polygonal flat shell element based on the hybrid stress/displacement-function finite element method for linear and nonlinear analyses by merging membrane and plate elements with a drilling degree of freedom. Indeed, it is demonstrated that warping effect did not render significant errors, but only one type of example (a single-curved cylindrical surface) was evaluated, with variations of quadrilateral-pentagonal and hexagonal meshes, all somehow structured. Then, there is still a persistent issue to ensure that the element's flat geometry approximation would succeed for wider variations of surface curvature and polygonal mesh arbitrariness. For that reason, we believe that polygonal shell elements should be modeled

without neglecting their spatial distortion, and the curved shell finite element approach should be preferred.

4.1.1 Conventional curved shell finite elements

Since the approach further used in polygonal curved elements is inspired by the conventional solutions (for triangular and quadrilateral meshes), a brief discussion is first reviewed herein.

Shells with general geometries can actually be modeled using tree-dimensional solid elements that typically have a thickness much smaller than its other dimensions (as illustrated in Fig. 4.1(a)). However, if the shell is not thick enough to be treated as a solid, assigning three nodes along the thickness direction at each element corner introduces too many unnecessary degrees of freedom. By removing the midsurface nodes, one obtains the configuration in Fig. 4.1(b). In this setup, lines along the thickness direction remain straight but are allowed to rotate relative to the shell midsurface, which aligns with the Reissner-Mindlin theory for plates and shells.



(a) 20-node solid element (b) 16-node element (c) degenerated shell element

Figure 4.1: Degenerated shell element idea: the 20-node solid element (a) generates a 16-node element after elimination of midpoints in thickness direction (b), which finally converges to the linear degenerated shell element (c). Adapted from Cook et al. [15].

As the element becomes thinner in Figs. 4.1(a) and 4.1(b), the stiffness components in the thickness direction increase significantly, leading to poor conditioning of the stiffness matrix [15]. To address this, one can enforce that each pair of nodes along a thickness line shares the same displacement in that direction, resulting in five DOF per node pair. Eventually, this leads to the simplified configuration shown in Fig. 4.1(c), where surface nodes are replaced by midsurface nodes, each having three translational and two rotational degrees of freedom. We adapted the original figure in Cook et al. [15] to purposely show a linear degenerated shell element (Fig. 4.1(c)), as studied in the present

work (i.e., the merged pair of nodes at the edge midpoints were removed). As with other elements based on the Reissner-Mindlin theory, challenges such as spurious modes or shear locking can emerge.

The MITC (Mixed Interpolation of Tensorial Components) approach was originally introduced to mitigate shear locking in the 3- and 4-node degenerated shell elements based on continuum mechanics, known as MITC3 and MITC4 [82, 83, 20]. Instead of interpolating the transverse shear strains (γ_{xz} and γ_{yz}) directly from the displacement field, they enforce constant covariant transverse shear strains along element edges by assuming that these constant values are the ones at the tying points (barycenter for MITC3 and midside points for MITC4). Commercial finite element packages commonly use the Assumed Shear Strains (ASS) by the MITC elements [84]. However, when the original MITC elements are applied to curved geometries using unstructured meshes, membrane locking may also occur. Then, higher-order triangular and quadrilateral elements were developed to reduce these anomalies (the MITC9, MITC16, and MITC6 shell elements) [85, 86, 87, 88]. Hereinafter, we will focus on the MITC4 variants, once the polygonal techniques will take advantage especially of these quadrilateral elements.

Alternatively to higher-order elements, linear options have been proposed to alleviate membrane locking in the original MITC4 element. Ko et al. [89] proposed the so-called "MITC4+" element, in which membrane locking is especially mitigated when elements are geometrically distorted in curved geometries. By using the concept of the MITC method, the tying membrane strains are obtained from four triangular domains which subdivide the shell mid-surface of the 4-node quadrilateral element. Sequentially, the same authors proposed a "new MITC4+" element [90] to improve the membrane efficiency of the previous "MITC4+" element, again in the context of distorted elements. In this new version, they represented strains in terms of characteristic geometry and displacement vectors, with the membrane strain field designed upon ideas by Choi and Paik [91] and Kulikov and Plotnikova [92]. Right after, the same authors launched one more option, namely the "improved MITC4+" element [93], where they further improved the last variant by utilizing the assumed strain field of a 2D solid element ("2D-MITC4" element) for the membrane behavior (maintaining the performance in bending and transverse shearing).

In 2024, Cui et al. [94] decreased the computational cost of the "MITC4+" element by proposing a stabilization technique that allowed a reduced numerical integration, from 2×2 to 1 Gauss point. Right after, Choi and Lee [84] disclosed a "simplified MITC4+" element, an updated version of the "improved MITC4+" approach, where a simplified assumed strain field

is obtained by merging the tying points into the element center, as well as the adoption of a geometry-dependent Gauss integration scheme [95]. Lastly, in 2025, Ko et al. [96] transformed the MITC4 and the "improved MITC4+" elements into 6-DOF versions with an extra drilling rotation at the nodes (the "MITC4/D" and "MITC4+/D" elements). The inclusion of this degree of freedom enables the connection of shells to beam elements and further increases the element membrane efficiency.

4.1.2 Polygonal curved shell finite elements

In 2018, Ho-Nguyen-Tan and Kim [97] opened a pathway developing a pentagonal curved shell element with Assumed Natural Strains (ANS) to tackle shear locking inspired by the same idea of the MITC elements. It is well known that shear locking governs the element error over the membrane locking, and that is the reason why some elements with no membrane treatment still reverberate in the literature. The overall motivation of Ho-Nguyen-Tan and Kim [97] was to transform curved boundaries of quadrilateral meshes, which are originally pixelated, into more rounded, smoother curved boundaries. The idea consists of trimming each quadrilateral element on a curved boundary, from which pentagons may emerge. Another motivation of this methodology regards the topology optimization, in which new body boundaries will arise throughout the process, allowing well-defined topologies.

An year later, the same authors extended the pentagonal approach for any arbitrary polygonal shell element, with the addition of a membrane locking-free technique [98]. A strategy of splitting the polygonal elements into quadrilateral subdomains was used to relieve the membrane locking, where each quadrilateral subdomain was treated as a "new MITC4+" element [90] to build the Assumed Membrane Strains (AMS). The same motivation of trimming quadrilaterals on boundaries remains, but this time any polygon may be addressed, not only pentagons. More recently, an enriched virtual element (VEM) was proposed by Yang et al. [99] for a membrane shell model, whose approach disregards bending moments. From the best of our knowledge, no updates following the constant development of the MITC4 variations are found in the literature, and no other polygonal curved finite element has been detected since then. Therefore, the present study will employ the most up-to-date methodology [98] to handle arbitrary polygonal curved shell finite elements.

4.2 Continuum-based degenerated shell elements

The Reissner-Mindlin theory for shells and plates is governed by similar constitutive equations, as both account for transverse shear deformation and employ independent rotational degrees of freedom. However, the shell formulation generalizes the plate theory to curved surfaces by introducing curvilinear coordinates, covariant derivatives, and additional in-plane displacements. While the stress relations maintain the same physical structure, their mathematical expressions in shell theory are adapted to the geometry of the shell midsurface [42, 43, 100, 101]. The difference, therefore, lies not in the constitutive behavior, but in the geometric framework and associated degrees of freedom, as will be discussed below.

Figure 4.2 shows a continuum-based degenerated polygonal shell element. The geometry of an n-gon element is described by

$$\boldsymbol{x}(\xi,\eta,\zeta) = \sum_{i=1}^{n} \phi^i(\xi,\eta)\boldsymbol{x}^i + \frac{\zeta}{2} h \sum_{i=1}^{n} \phi^i(\xi,\eta)\boldsymbol{V}_3^i, \tag{4-1}$$

where $\phi^i(\xi,\eta)$ are the two-dimensional shape functions at each node i of the element (Wachspress type [29]), $\boldsymbol{x}^i$ are the position vectors of the shell midsurface ($\zeta = 0$) at each node i in the global Cartesian coordinates (x, y, z), h is the shell thickness, and $\boldsymbol{V}_3^i$ are the unit direction vectors at each node i. Vectors $\boldsymbol{V}_3^i$ are not necessarily normal to the shell midsurface, but in this work we will in general model surface geometry with normal $\boldsymbol{V}_3^i$ vectors.

The displacement vector field is then

$$\boldsymbol{u}(\xi,\eta,\zeta) = \sum_{i=1}^{n} \phi^i(\xi,\eta)\boldsymbol{u}^i + \frac{\zeta}{2} h \sum_{i=1}^{n} \phi^i(\xi,\eta)\left(-\boldsymbol{V}_2^i\alpha^i + \boldsymbol{V}_1^i\beta^i\right), \tag{4-2}$$

where $\boldsymbol{u}^i = [u^i \;\; v^i \;\; w^i]^T$ are the nodal translational displacement vectors at the midsurface, $\boldsymbol{V}_1^i$ and $\boldsymbol{V}_2^i$ are the other unit direction vectors, from which α^i and β^i are defined as rotations of $\boldsymbol{V}_3^i$ about $\boldsymbol{V}_1^i$ and $\boldsymbol{V}_2^i$, respectively. The nodal displacement vector is then written as $\mathbf{d}^i = [u^i \;\; v^i \;\; w^i \;\; \alpha^i \;\; \beta^i]^T$. One common way to determine vectors $\boldsymbol{V}_1^i$ and $\boldsymbol{V}_2^i$ is by the following cross product

$$\begin{cases} \boldsymbol{V}_1^i = \frac{\boldsymbol{e}_2 \times \boldsymbol{V}_3^i}{\left\|\boldsymbol{e}_2 \times \boldsymbol{V}_3^i\right\|} \text{ and } \boldsymbol{V}_2^i = \boldsymbol{V}_3^i \times \boldsymbol{V}_1^i, & \text{if } \boldsymbol{V}_3^i \nparallel \boldsymbol{e}_2; \\ \boldsymbol{V}_2^i = \frac{\boldsymbol{V}_3^i \times \boldsymbol{e}_1}{\left\|\boldsymbol{V}_3^i \times \boldsymbol{e}_1\right\|} \text{ and } \boldsymbol{V}_1^i = \boldsymbol{V}_2^i \times \boldsymbol{V}_3^i, & \text{if } \boldsymbol{V}_3^i \parallel \boldsymbol{e}_2. \end{cases} \tag{4-3}$$

The thickness-direction vector $\boldsymbol{V}_3^i$ can also be represented as the coordinates of the direction cosines, i.e.,

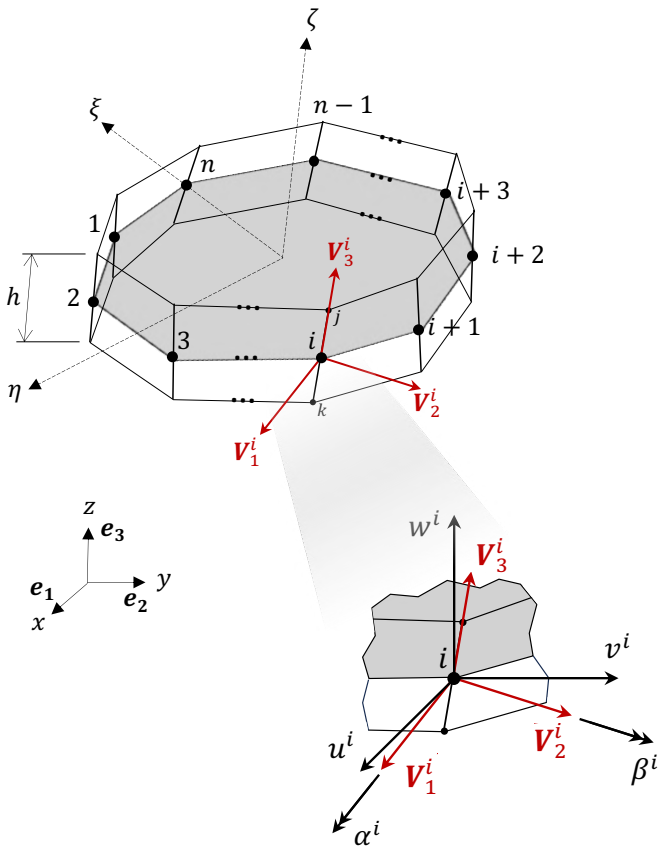


Figure 4.2: The continuum-based degenerated shell element with n sides.

$$\boldsymbol{V}_3^i = \left\{ \begin{array}{c} l_3^i \\ m_3^i \\ n_3^i \end{array} \right\} = \frac{1}{h} \left\{ \begin{array}{c} x_j - x_k \\ y_j - y_k \\ z_j - z_k \end{array} \right\}, \tag{4-4}$$

where (x_j, y_j, z_j) and (x_k, y_k, z_k) represent the coordinates of the top ($\zeta = 1$) and bottom ($\zeta = -1$) surface vertices at node i, respectively (see Fig. 4.2). We may also write $\boldsymbol{V}_1^i = [l_1^i, m_1^i, n_1^i]$ and $\boldsymbol{V}_2^i = [l_2^i, m_2^i, n_2^i]$. These vectors can be interpolated to any point inside the element (e.g., a Gauss point) as

$$\boldsymbol{V}_1 = \sum_{i=1}^{n} \phi^i \boldsymbol{V}_1^i; \qquad \boldsymbol{V}_2 = \sum_{i=1}^{n} \phi^i \boldsymbol{V}_2^i; \qquad \boldsymbol{V}_3 = \sum_{i=1}^{n} \phi^i \boldsymbol{V}_3^i. \tag{4-5}$$

The linear, Green-Lagrange version of the covariant[1] strain components, defined with respect to the element natural coordinate system, are [102]

$$e_{ij} = \frac{1}{2} \left(\mathbf{g}_i \cdot \boldsymbol{u}_{,j} + \mathbf{g}_j \cdot \boldsymbol{u}_{,i} \right), \tag{4-6}$$

where

$$\mathbf{g}_i = \frac{\partial \boldsymbol{x}}{\partial \xi_i} \quad \text{and} \quad \boldsymbol{u}_{,i} = \frac{\partial \boldsymbol{u}}{\partial \xi_i}, \qquad \text{with } \xi_1 = \xi, \xi_2 = \eta, \xi_3 = \zeta. \tag{4-7}$$

[1]Covariant coordinates refer to a system of curvilinear coordinates defined on a surface, where the basis vectors vary from point to point and are tangent to the geometry, enabling the formulation of strain and stress components relative to this surface shape [100, 101, 102].

In other words,

$$\mathbf{g}_i = \begin{cases} \sum_{j=1}^{n} \phi_{,i}^{j} \left(\boldsymbol{x}^j + \frac{\zeta}{2} h \boldsymbol{V}_3^j \right), & \text{if } i = 1, 2; \\ \frac{h}{2} \sum_{j=1}^{n} \phi^j \boldsymbol{V}_3^j, & \text{if } i = 3, \end{cases} \tag{4-8}$$

and

$$\boldsymbol{u}_{,i} = \begin{cases} \sum_{j=1}^{n} \phi_{,i}^{j} \left[\boldsymbol{u}^j + \frac{\zeta}{2} h \left(-\boldsymbol{V}_2^j \alpha^j + \boldsymbol{V}_1^j \beta^j \right) \right], & \text{if } i = 1, 2; \\ \frac{h}{2} \sum_{j=1}^{n} \phi^j \left(-\boldsymbol{V}_2^j \alpha^j + \boldsymbol{V}_1^j \beta^j \right), & \text{if } i = 3. \end{cases} \tag{4-9}$$

Vectors $\mathbf{g}_i$ are also called the covariant basis vectors. The displacement-based covariant strain components may be written as

$$e_{ij} = \mathbf{B}_{ij} \mathbf{U}. \tag{4-10}$$

We can transfer the covariant strain components directly to the global Cartesian coordinate system simply by

$$\boldsymbol{\varepsilon} = e_{ij} \left(\mathbf{g}^i \otimes \mathbf{g}^j \right), \tag{4-11}$$

where $\mathbf{g}^i = \partial \xi_i / \partial \boldsymbol{x}$, or the contravariant basis vectors (Einstein summation notation is valid). All strains in the following Sections are covariant strain components, unless otherwise stated.

The local constitutive tensor ($\mathbf{D}'$) for linear elastic, isotropic material is obtained from

$$\begin{bmatrix} \sigma_\xi \\ \sigma_\eta \\ \sigma_\zeta \\ \tau_{\xi\eta} \\ \tau_{\eta\zeta} \\ \tau_{\zeta\xi} \end{bmatrix} = \begin{bmatrix} \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 & 0 \\ \nu & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & (1-\nu)/2 \end{bmatrix} & \mathbf{0} \\ \mathbf{0} & \begin{bmatrix} G & 0 \\ 0 & G \end{bmatrix} \end{bmatrix} \begin{bmatrix} \varepsilon_\xi \\ \varepsilon_\eta \\ \varepsilon_\zeta \\ \gamma_{\xi\eta} \\ \gamma_{\eta\zeta} \\ \gamma_{\zeta\xi} \end{bmatrix}, \tag{4-12}$$

or

$$\boldsymbol{\sigma}' = \mathbf{D}' \boldsymbol{\varepsilon}', \tag{4-13}$$

where $G = \frac{\kappa^s E}{2(1+\nu)}$ is the shear modulus with $\kappa^s = 5/6$ being the shear correction factor. Note that we enforce $\sigma_\zeta = 0$ in Eq. 4-12 to have plane stress conditions in each thickness-direction layer. We introduce the stress-strain transformation matrix ($\mathbf{T}_\varepsilon$) as

$$\mathbf{T}_\varepsilon = \begin{bmatrix} l_1^2 & m_1^2 & n_1^2 & l_1m_1 & m_1n_1 & n_1l_1 \\ l_2^2 & m_2^2 & n_2^2 & l_2m_2 & m_2n_2 & n_2l_2 \\ l_3^2 & m_3^2 & n_3^2 & l_3m_3 & m_3n_3 & n_3l_3 \\ 2l_1l_2 & 2m_1m_2 & 2n_1n_2 & l_1m_2+l_2m_1 & m_1n_2+m_2n_1 & n_1l_2+n_2l_1 \\ 2l_2l_3 & 2m_2m_3 & 2n_2n_3 & l_2m_3+l_3m_2 & m_2n_3+m_3n_2 & n_2l_3+n_3l_2 \\ 2l_3l_1 & 2m_3m_1 & 2n_3n_1 & l_3m_1+l_1m_3 & m_3n_1+m_1n_3 & n_3l_1+n_1l_3 \end{bmatrix} \tag{4-14}$$

where $[l_*, m_*, n_*]$ are the direction-cosines components of the vector $\boldsymbol{V}_*$ as per Eq. 4-5. The transformation of the constitutive tensor from local ($\mathbf{D}'$) to global ($\mathbf{D}$) reference is performed through

$$\mathbf{D} = \mathbf{T}_\varepsilon^T \mathbf{D}' \mathbf{T}_\varepsilon \tag{4-15}$$

Finally, the element stiffness matrix of the polygonal degenerated shell element is stated as

$$\begin{aligned} \mathbf{K}_0^e &= \int_{\Omega^e} \mathbf{B}^T\mathbf{D}\mathbf{B} d\Omega^e \\ &\approx \sum_{i=1}^{Ngp_\xi}\sum_{j=1}^{Ngp_\eta}\sum_{k=1}^{Ngp_\zeta} \mathbf{B}^T\mathbf{D}\mathbf{B}\Big|_{(\xi_i,\eta_j,\zeta_k)} \det\left(\mathbf{J}\Big|_{(\xi_i,\eta_j,\zeta_k)}\right) W_iW_jW_k, \end{aligned} \tag{4-16}$$

where W are the Gauss weights and Ngp_ξ is the number of Gauss points used for ξ (analogous to η and ζ). Ngp_ξ and Ngp_η follows the same triangularization framework as in `PolyTop` (or quadrangularization, as further discussed), and Ngp_ζ is integrated in this work with two Gauss points in the thickness direction. With Eq. 4-7, the Jacobian matrix $\mathbf{J}$ can be computed as follows

$$\mathbf{J} = \begin{bmatrix} — & \mathbf{g}_1^T & — \\ — & \mathbf{g}_2^T & — \\ — & \mathbf{g}_3^T & — \end{bmatrix}, \tag{4-17}$$

while its inverse can be built with the contravariant basis vectors as

$$\mathbf{J}^{-1} = \begin{bmatrix} | & | & | \\ \mathbf{g}^1 & \mathbf{g}^2 & \mathbf{g}^3 \\ | & | & | \end{bmatrix}. \tag{4-18}$$

Whenever necessary, the above approach requires a proper definition of $\boldsymbol{V}_3^i$ at nodes where the shell surface is sharp to avoid ambiguity.

4.2.1 Shear locking treatment

In this study, the technique to alleviate transverse shear locking in the polygonal shell is based on Ho-Nguyen-Tan and Kim [98]. These authors claim that it is possible to find some points at which the transverse shear strains

may represent the average distribution of the transverse shear strain fields at a particular region. Also, they state that, for thin shells, even though the transverse shear strains do not vanish at all points, the average value of the transverse shear strains over the element tends to zero. Therefore, an Assumed Shear Strain (ASS) field, constructed from the transverse shear strains evaluated at specific locations, referred to as "tying points", that represent the average values can be used to vanish spurious transverse shear strains within the element [103]. In the MITC method [82, 20], this ASS field can be defined as

$$\tilde{e}_{ij} = \sum_{k=1}^{n_{tp}} \psi^k e_{ij}\left(\xi_{tp}^k, \eta_{tp}^k\right) \text{ with } i = 1, 2 \text{ and } j = 3, \tag{4-19}$$

where $\left(\xi_{tp}^k, \eta_{tp}^k\right)$ is the location of a tying point k, n_{tp} is the number of tying points within an element, and ψ^k is the interpolation function of the tying point k. As proposed by Ho-Nguyen-Tan and Kim [98], the interpolation utilized is a simple polynomial, such that

$$\begin{aligned} \tilde{e}_{\xi\zeta} &= \boldsymbol{f}^{\langle p \rangle}(\xi, \zeta) \cdot \boldsymbol{a} \\ \tilde{e}_{\eta\zeta} &= \boldsymbol{f}^{\langle p \rangle}(\xi, \zeta) \cdot \boldsymbol{b} \end{aligned} \tag{4-20}$$

where

$$\boldsymbol{f}^{\langle p \rangle}(\xi, \eta) = [\underbrace{1, \xi, \eta, \ldots}_{m}] \tag{4-21}$$

is the isotropic basis vector of degree p on the Pascal's triangle in (ξ, η) coordinates (see Fig. 4.3), and $p = \lceil n/2 \rceil - 1$ and $m = \sum_{i=1}^{p+1} i$. The utilization of these isotropic polynomials of Pascal's triangle guarantees geometric isotropy of the element. The unknown coefficient vectors ($\boldsymbol{a}$ and $\boldsymbol{b}$) are defined as

$$\begin{aligned} \boldsymbol{a} &= [a_1, a_2, \ldots, a_m]^T \\ \boldsymbol{b} &= [b_1, b_2, \ldots, b_m]^T . \end{aligned} \tag{4-22}$$

The tying point $\left(\xi_{tp}^i, \eta_{tp}^i\right)$ is placed at the midpoint of the polygonal element edge i, as in the original MITC4 element [82]. The transverse shear strains at the tying points on a general inclined edge, $e_{q\zeta}^{(i)}\left(\xi_{tp}^i, \eta_{tp}^i\right)$, are

$$e_{q\zeta}^{(i)}\left(\xi_{tp}^i, \eta_{tp}^i\right) = e_{\xi\zeta}^{(i)}\left(\xi_{tp}^i, \eta_{tp}^i\right)\cos\varphi^{(i)} + e_{\eta\zeta}^{(i)}\left(\xi_{tp}^i, \eta_{tp}^i\right)\cos\rho^{(i)} \tag{4-23}$$

where $\varphi^{(i)}$ and $\rho^{(i)}$ are the angles of the inclined edge i with respect to ξ and η axes, respectively, and q indicates the direction of the inclined edge tangent axis (see Fig. 4.4). Spurious transverse shear strains can be avoided by assuming constant transverse shear strain conditions along the edges of the element. To ensure these conditions are met along edge i, the assumed shear strains $\tilde{e}_{q\zeta}^{(i)}$ along the element edge must match the transverse shear strain $e_{qt}^{(i)}$ evaluated

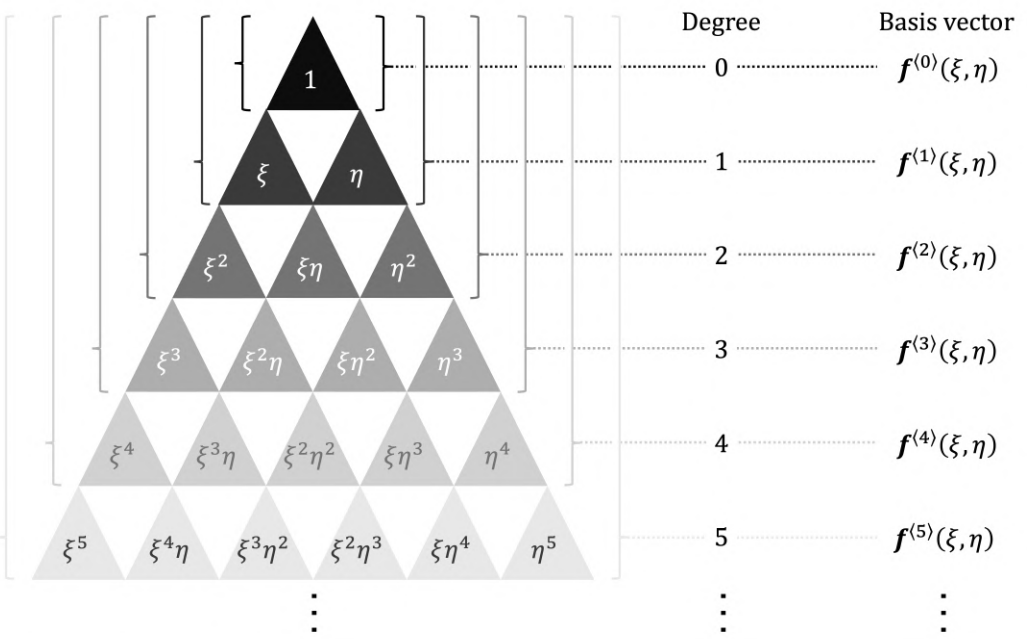


Figure 4.3: The isotropic polynomial basis vectors of Pascal's triangle in (ξ, η) coordinate system (adapted from [98]).

at the corresponding tying points, i.e.

$$\tilde{e}^{(i)}_{q\zeta}\left(\xi^i_h, \eta^i_h\right) = e^{(i)}_{q\zeta}\left(\xi^i_{tp}, \eta^i_{tp}\right), h = 1, 2, \ldots, n_c, \tag{4-24}$$

where (ξ^i_h, η^i_h) are equally spaced positions, starting from the vertex, along the element edge i and $n_c = \lceil n/2 \rceil$ is the required number of these positions to satisfy the constant transverse shear strain conditions (see Fig. 4.4). Combining Eqs. 4-20, 4-23, and 4-24, we obtain

$$\left[\boldsymbol{f}^{\langle p\rangle}\left(\xi^i_h, \eta^i_h\right)\cos\varphi^{(i)}\right]\cdot\boldsymbol{a} + \left[\boldsymbol{f}^{\langle p\rangle}\left(\xi^i_h, \eta^i_h\right)\cos\rho^{(i)}\right]\cdot\boldsymbol{b} = e^{(i)}_{q\zeta}\left(\xi^i_{tp}, \eta^i_{tp}\right) \tag{4-25}$$

or

$$\begin{bmatrix}
\boldsymbol{f}^{\langle p\rangle}\left(\xi^1_1, \eta^1_1\right)\cos\varphi^{(1)} & \boldsymbol{f}^{\langle p\rangle}\left(\xi^1_1, \eta^1_1\right)\cos\rho^{(1)} \\
\boldsymbol{f}^{\langle p\rangle}\left(\xi^1_2, \eta^1_2\right)\cos\varphi^{(1)} & \boldsymbol{f}^{\langle p\rangle}\left(\xi^1_2, \eta^1_2\right)\cos\rho^{(1)} \\
\cdots & \cdots \\
\boldsymbol{f}^{\langle p\rangle}\left(\xi^1_{n_c}, \eta^1_{n_c}\right)\cos\varphi^{(1)} & \boldsymbol{f}^{\langle p\rangle}\left(\xi^1_{n_c}, \eta^1_{n_c}\right)\cos\rho^{(1)} \\
\boldsymbol{f}^{\langle p\rangle}\left(\xi^2_1, \eta^2_1\right)\cos\varphi^{(2)} & \boldsymbol{f}^{\langle p\rangle}\left(\xi^2_1, \eta^2_1\right)\cos\rho^{(2)} \\
\boldsymbol{f}^{\langle p\rangle}\left(\xi^2_2, \eta^2_2\right)\cos\varphi^{(2)} & \boldsymbol{f}^{\langle p\rangle}\left(\xi^2_2, \eta^2_2\right)\cos\rho^{(2)} \\
\cdots & \cdots \\
\boldsymbol{f}^{\langle p\rangle}\left(\xi^2_{n_c}, \eta^2_{n_c}\right)\cos\varphi^{(2)} & \boldsymbol{f}^{\langle p\rangle}\left(\xi^2_{n_c}, \eta^2_{n_c}\right)\cos\rho^{(2)} \\
\cdots & \cdots \\
\cdots & \cdots \\
\boldsymbol{f}^{\langle p\rangle}\left(\xi^n_{n_c}, \eta^n_{n_c}\right)\cos\varphi^{(n)} & \boldsymbol{f}^{\langle p\rangle}\left(\xi^n_{n_c}, \eta^n_{n_c}\right)\cos\rho^{(n)}
\end{bmatrix}
\begin{bmatrix} \boldsymbol{a} \\ \boldsymbol{b} \end{bmatrix} =
\begin{bmatrix}
e^{(1)}_{q\zeta}\left(\xi^1_{tp}, \eta^1_{tp}\right) \\
e^{(1)}_{q\zeta}\left(\xi^1_{tp}, \eta^1_{tp}\right) \\
\cdots \\
e^{(1)}_{q\zeta}\left(\xi^1_{tp}, \eta^1_{tp}\right) \\
e^{(2)}_{q\zeta}\left(\xi^2_{tp}, \eta^2_{tp}\right) \\
e^{(2)}_{q\zeta}\left(\xi^2_{tp}, \eta^2_{tp}\right) \\
\cdots \\
e^{(2)}_{q\zeta}\left(\xi^2_{tp}, \eta^2_{tp}\right) \\
\cdots \\
\cdots \\
e^{(n)}_{q\zeta}\left(\xi^n_{tp}, \eta^n_{tp}\right)
\end{bmatrix}. \tag{4-26}$$

For $n > 3$, the number of unknowns $(2\times m)$ is always less than the number

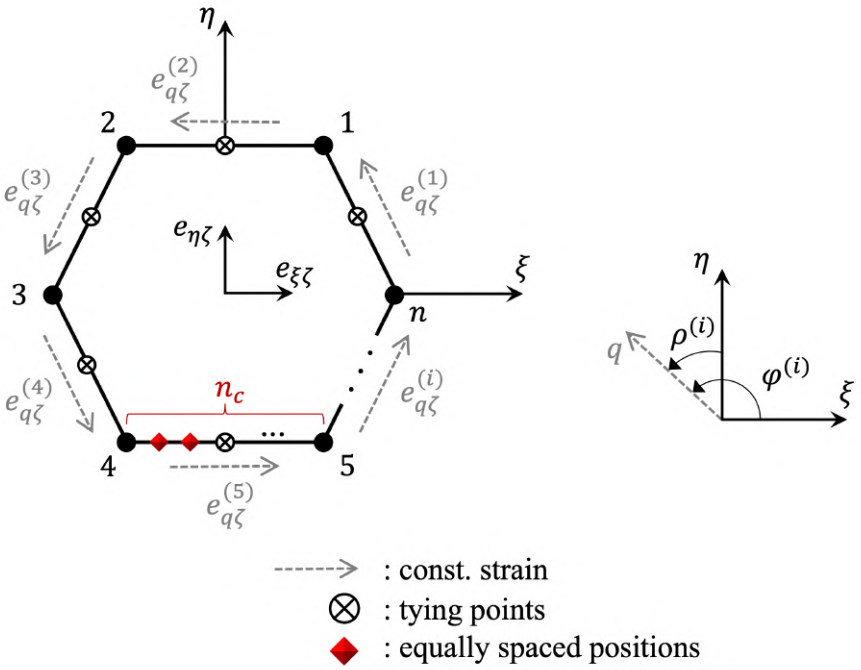


Figure 4.4: Tying points, angles and equally spaced positions for the ASS field of polygonal shell elements (adapted from [98]).

of equations ($n \times n_c$). Therefore, a pseudoinverse is applied through the least square method to find an optimal solution to the constant transverse shear strain conditions. Consequently, the ASS field is a least square fitting of the constant transverse shear strains on element edges to the isotropic polynomials of Pascal's triangle.

4.2.2 Membrane locking treatment

From equation 4-6, the in-plane strains is rewritten as a combination of the membrane strains at the shell midsurface ($\zeta = 0$) with terms depending on ζ due to bending deformations [89, 90], i.e.

$$e_{ij} = e^m_{ij} + \zeta e^{b1}_{ij} + \zeta^2 e^{b2}_{ij}, \text{ with } i,j = 1,2 \tag{4-27}$$

where

$$\begin{aligned} e^m_{ij} &= \frac{1}{2}\left(\frac{\partial \boldsymbol{x}_m}{\partial \xi_i}\cdot\frac{\partial \boldsymbol{u}_m}{\partial \xi_j} + \frac{\partial \boldsymbol{x}_m}{\partial \xi_j}\cdot\frac{\partial \boldsymbol{u}_m}{\partial \xi_i}\right) \\ e^{b1}_{ij} &= \frac{1}{2}\left(\frac{\partial \boldsymbol{x}_m}{\partial \xi_i}\cdot\frac{\partial \boldsymbol{u}_b}{\partial \xi_j} + \frac{\partial \boldsymbol{x}_m}{\partial \xi_j}\cdot\frac{\partial \boldsymbol{u}_b}{\partial \xi_i} + \frac{\partial \boldsymbol{x}_b}{\partial \xi_i}\cdot\frac{\partial \boldsymbol{u}_m}{\partial \xi_j} + \frac{\partial \boldsymbol{x}_b}{\partial \xi_j}\cdot\frac{\partial \boldsymbol{u}_m}{\partial \xi_i}\right) \\ e^{b2}_{ij} &= \frac{1}{2}\left(\frac{\partial \boldsymbol{x}_b}{\partial \xi_i}\cdot\frac{\partial \boldsymbol{u}_b}{\partial \xi_j} + \frac{\partial \boldsymbol{x}_b}{\partial \xi_j}\cdot\frac{\partial \boldsymbol{u}_b}{\partial \xi_i}\right) \end{aligned} \tag{4-28}$$

with

$$\boldsymbol{x}_m = \sum_{i=1}^{n} \phi^i(\xi,\eta)\boldsymbol{x}^i \quad , \qquad \boldsymbol{x}_b = \frac{1}{2}\sum_{i=1}^{n} h\phi^i(\xi,\eta)\boldsymbol{V}_3^i \tag{4-29}$$

$$\boldsymbol{u}_m = \sum_{i=1}^{n} \phi^i(\xi,\eta)\boldsymbol{u}^i \quad , \qquad \boldsymbol{u}_b = \frac{1}{2}\sum_{i=1}^{n} h\phi^i(\xi,\eta)\left(-\boldsymbol{V}_2^i\alpha^i + \boldsymbol{V}_1^i\beta^i\right). \tag{4-30}$$

It is well-established that the degenerated shell elements on curved geometries may exhibit membrane locking. This issue arises because such elements fail to reproduce pure bending modes or inextensional motions of the shell's midsurface without inadvertently generating spurious membrane strains [55]. To mitigate this undesired behavior, Ho-Nguyen-Tan and Kim [98] introduced an Assumed Membrane Strain (AMS) field by partitioning the polygonal shell element into a set of non-overlapping quadrilateral subdomains. These subdomains are constructed using the centroid of the polygonal element and the midpoints of its edges, as illustrated in Fig. 4.5. The centroid of the polygonal shell element, considering only $n > 4$, is defined as

$$\overline{\boldsymbol{x}} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}^i, \tag{4-31}$$

and the displacement vector at the centroid is

$$\overline{\boldsymbol{u}} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{u}^i. \tag{4-32}$$

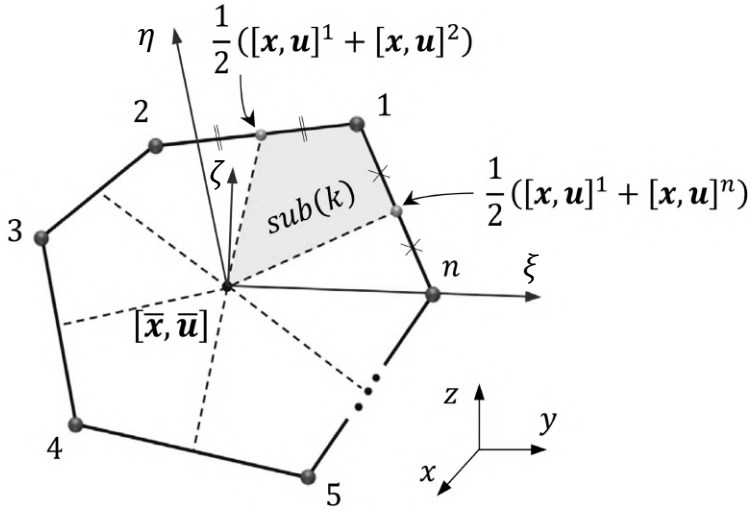


Figure 4.5: Subdivision of the polygonal shell element into quadrilateral subdomains to build the AMS field (adapted from [98]).

Now, looking at a quadrilateral subdomain k, the membrane geometry is interpolated as

$$\hat{\boldsymbol{x}}_m^{(k)}(\hat{\xi},\hat{\eta}) = \sum_{I=1}^{4} \hat{\phi}^I(\hat{\xi},\hat{\eta})\boldsymbol{x}^{I(k)}, \tag{4-33}$$

where $\boldsymbol{x}^{I(k)}$ is the position vector of the point I of the kth quadrilateral subdomain, $\hat{\phi}^I(\hat{\xi},\hat{\eta}) = 1/4(1+\hat{\xi}^I\hat{\xi})(1+\hat{\eta}^I\hat{\eta})$ is the interpolation function of point I in $\hat{\xi}$ and $\hat{\eta}$ coordinates, with $\hat{\xi}^I = (-1,1,1,-1)$ and $\hat{\eta}^I = (-1,-1,1,1)$, and $(\hat{\xi},\hat{\eta})$ is the element natural coordinates for the quadrilateral subdomains. Then, we can write

$$\hat{\boldsymbol{u}}_m^{(k)}(\hat{\xi},\hat{\eta}) = \sum_{I=1}^{4} \hat{\phi}^I(\hat{\xi},\hat{\eta})\boldsymbol{u}^{I(k)}. \tag{4-34}$$

As shown in Fig. 4.5, the displacement vectors at the edge midpoints are computed by a linear interpolation of the displacements of the two neighboring nodes. Therefore, the membrane strains in a quadrilateral subdomain k can be expressed as

$$\hat{e}_{ij}^{m(k)} = \frac{1}{2}\left(\hat{\mathbf{g}}_i^{(k)} \cdot \hat{\boldsymbol{u}}_{,j}^{(k)} + \hat{\mathbf{g}}_j^{(k)} \cdot \hat{\boldsymbol{u}}_{,i}^{(k)}\right), \tag{4-35}$$

where

$$\hat{\mathbf{g}}_i^{(k)} = \frac{\partial \hat{\boldsymbol{x}}_m^{(k)}}{\partial \hat{\xi}_i} \quad \text{and} \quad \hat{\boldsymbol{u}}_{,i}^{(k)} = \frac{\partial \hat{\boldsymbol{u}}_m^{(k)}}{\partial \hat{\xi}_i}, \quad \text{with } \hat{\xi}_1 = \hat{\xi},\ \hat{\xi}_2 = \hat{\eta}, \quad \text{and } \hat{\mathbf{g}}_3^{(k)} = \mathbf{g}_3. \tag{4-36}$$

From Eq. 4-33, we obtain

$$\frac{\partial \hat{\boldsymbol{x}}_m^{(k)}}{\partial \hat{\xi}} = \hat{\boldsymbol{x}}_{\hat{\xi}}^{(k)} + \hat{\eta}\hat{\boldsymbol{x}}_d^{(k)}, \quad \frac{\partial \hat{\boldsymbol{x}}_m^{(k)}}{\partial \hat{\eta}} = \hat{\boldsymbol{x}}_{\hat{\eta}}^{(k)} + \hat{\xi}\hat{\boldsymbol{x}}_d^{(k)} \tag{4-37}$$

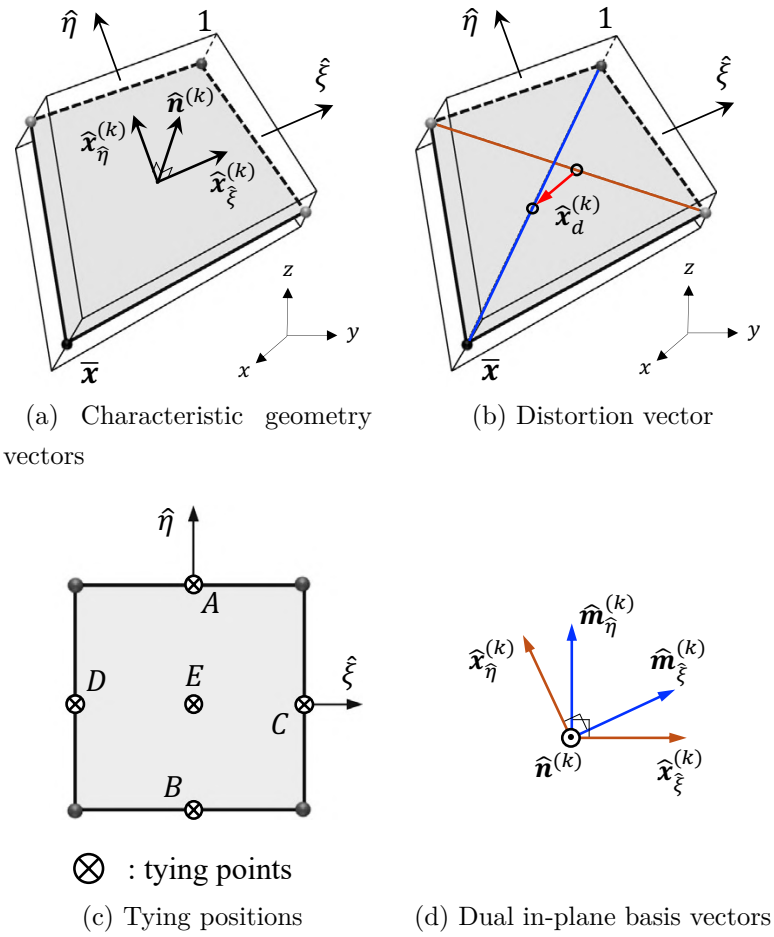
$$\begin{aligned}
\hat{\boldsymbol{x}}_{\hat{\xi}}^{(k)} &= \frac{1}{4}\sum_{I=1}^{4} \hat{\xi}^I \boldsymbol{x}^{I(k)}, \\
\hat{\boldsymbol{x}}_{\hat{\eta}}^{(k)} &= \frac{1}{4}\sum_{I=1}^{4} \hat{\eta}^I \boldsymbol{x}^{I(k)}, \\
\hat{\boldsymbol{x}}_{d}^{(k)} &= \frac{1}{4}\sum_{I=1}^{4} \hat{\xi}^I\hat{\eta}^I \boldsymbol{x}^{I(k)},
\end{aligned} \tag{4-38}$$

where $\hat{\boldsymbol{x}}_{\hat{\xi}}^{(k)}$ and $\hat{\boldsymbol{x}}_{\hat{\eta}}^{(k)}$ are the characteristic geometry vectors and $\hat{\boldsymbol{x}}_d^{(k)}$ is the distortion vector of the quadrilateral subdomain k (see Figs. 4.6(a) and 4.6(b)). Analogously, from Eq. 4-34, we have

$$\frac{\partial \hat{\boldsymbol{u}}_m^{(k)}}{\partial \hat{\xi}} = \hat{\boldsymbol{u}}_{\hat{\xi}}^{(k)} + \hat{\eta}\hat{\boldsymbol{u}}_d^{(k)}, \quad \frac{\partial \hat{\boldsymbol{u}}_m^{(k)}}{\partial \hat{\eta}} = \hat{\boldsymbol{u}}_{\hat{\eta}}^{(k)} + \hat{\xi}\hat{\boldsymbol{u}}_d^{(k)} \tag{4-39}$$

$$\begin{aligned}
\hat{\boldsymbol{u}}_{\hat{\xi}}^{(k)} &= \frac{1}{4}\sum_{I=1}^{4} \hat{\xi}^I \boldsymbol{u}^{I(k)}, \\
\hat{\boldsymbol{u}}_{\hat{\eta}}^{(k)} &= \frac{1}{4}\sum_{I=1}^{4} \hat{\eta}^I \boldsymbol{u}^{I(k)}, \\
\hat{\boldsymbol{u}}_{d}^{(k)} &= \frac{1}{4}\sum_{I=1}^{4} \hat{\xi}^I\hat{\eta}^I \boldsymbol{u}^{I(k)},
\end{aligned} \tag{4-40}$$

where $\hat{\boldsymbol{x}}_{\hat{\xi}}^{(k)}$, $\hat{\boldsymbol{x}}_{\hat{\eta}}^{(k)}$, and $\hat{\boldsymbol{x}}_d^{(k)}$ are the characteristic displacement vectors.



(a) Characteristic geometry vectors

(b) Distortion vector

(c) Tying positions

(d) Dual in-plane basis vectors

Figure 4.6: The characteristic geometry vectors on a quadrilateral subdomain k of the polygonal shell element: two in-plane vectors $\hat{\boldsymbol{x}}_{\hat{\xi}}^{(k)}$, $\hat{\boldsymbol{x}}_{\hat{\eta}}^{(k)}$ and normal vector $\hat{\boldsymbol{n}}^{(k)}$ (a), the distortion vector $\hat{\boldsymbol{x}}_d^{(k)}$ (b), tying positions A, B, C, D and E of a quadrilateral subdomain (c), and the dual in-plane basis vectors $\hat{\boldsymbol{m}}_{\hat{\xi}}^{(k)}$ and $\hat{\boldsymbol{m}}_{\hat{\eta}}^{(k)}$ (d). (Adapted from [98]).

An AMS field can be formulated using the characteristic geometry and displacement vectors to diminish spurious membrane strains that arise in curved shell elements subjected to pure bending deformation [90]. In this methodology, linear terms are incorporated into the in-plane shear strain expressions to ensure the patch test is satisfied. Furthermore, the bending performance is enhanced by adopting the strategy proposed by Kulikov and Plotnikova [92]. Ho-Nguyen-Tan and Kim [98] utilized this same approach within each quadrilateral subdomain to formulate the AMS field for the polygonal shell elements. Specifically, the membrane strains calculated at the five tying points in the quadrilateral reference element [90] are used to establish the AMS field within each subdomain k, such that

$$\begin{aligned}\tilde{e}^{m(k)}_{\hat{\xi}\hat{\xi}} = \quad & \frac{1}{2}\left(1-2\varphi_A+\hat{\eta}+2\varphi_A\hat{\eta}^2\right)\hat{e}^{m(A)}_{\hat{\xi}\hat{\xi}} \\ & +\frac{1}{2}\left(1-2\varphi_B-\hat{\eta}+2\varphi_B\hat{\eta}^2\right)\hat{e}^{m(B)}_{\hat{\xi}\hat{\xi}} \\ & +\varphi_C\left(-1+\hat{\eta}^2\right)\hat{e}^{m(C)}_{\hat{\eta}\hat{\eta}} \\ & +\varphi_D\left(-1+\hat{\eta}^2\right)\hat{e}^{m(D)}_{\hat{\eta}\hat{\eta}} \\ & +\varphi_E\left(-1+\hat{\eta}^2\right)\hat{e}^{m(E)}_{\hat{\xi}\hat{\eta}},\end{aligned} \tag{4-41}$$

$$\begin{aligned}\tilde{e}^{m(k)}_{\hat{\eta}\hat{\eta}} = \quad & \varphi_A\left(-1+\hat{\xi}^2\right)\hat{e}^{m(A)}_{\hat{\xi}\hat{\xi}} \\ & +\varphi_B\left(-1+\hat{\xi}^2\right)\hat{e}^{m(B)}_{\hat{\xi}\hat{\xi}} \\ & +\frac{1}{2}\left(1-2\varphi_C+\hat{\xi}+2\varphi_C\hat{\xi}^2\right)\hat{e}^{m(C)}_{\hat{\eta}\hat{\eta}} \\ & +\frac{1}{2}\left(1-2\varphi_D-\hat{\xi}+2\varphi_D\hat{\xi}^2\right)\hat{e}^{m(D)}_{\hat{\eta}\hat{\eta}} \\ & +\varphi_E\left(-1+\hat{\xi}^2\right)\hat{e}^{m(E)}_{\hat{\xi}\hat{\eta}},\end{aligned} \tag{4-42}$$

$$\begin{aligned}\tilde{e}^{m(k)}_{\hat{\xi}\hat{\eta}} = \quad & \frac{1}{4}\left(\hat{\xi}+4\varphi_A\hat{\xi}\hat{\eta}\right)\hat{e}^{m(A)}_{\hat{\xi}\hat{\xi}} \\ & +\frac{1}{4}\left(-\hat{\xi}+4\varphi_B\hat{\xi}\hat{\eta}\right)\hat{e}^{m(B)}_{\hat{\xi}\hat{\xi}} \\ & +\frac{1}{4}\left(\hat{\eta}+4\varphi_C\hat{\xi}\hat{\eta}\right)\hat{e}^{m(C)}_{\hat{\eta}\hat{\eta}} \\ & +\frac{1}{4}\left(-\hat{\eta}+4\varphi_D\hat{\xi}\hat{\eta}\right)\hat{e}^{m(D)}_{\hat{\eta}\hat{\eta}} \\ & +\left(1+\varphi_E\hat{\xi}\hat{\eta}\right)\hat{e}^{m(E)}_{\hat{\xi}\hat{\eta}},\end{aligned} \tag{4-43}$$

where the tying points of the quadrilateral subdomain (A, B, C, D and E) can be seen in Fig. 4.6(c), while the geometric coefficients are given as

$$\begin{aligned}\varphi_A &= \frac{c_{\hat{\xi}}\left(c_{\hat{\xi}}-1\right)}{2d}, \\ \varphi_B &= \frac{c_{\hat{\xi}}\left(c_{\hat{\xi}}+1\right)}{2d}, \\ \varphi_C &= \frac{c_{\hat{\eta}}\left(c_{\hat{\eta}}-1\right)}{2d}, \\ \varphi_D &= \frac{c_{\hat{\eta}}\left(c_{\hat{\eta}}+1\right)}{2d}, \\ \varphi_E &= \frac{2c_{\hat{\xi}}c_{\hat{\eta}}}{d},\end{aligned} \tag{4-44}$$

with

$$\begin{aligned}c_{\hat{\xi}} &= \hat{\boldsymbol{m}}^{(k)}_{\hat{\xi}}\cdot\hat{\boldsymbol{x}}^{(k)}_{d}, \\ c_{\hat{\eta}} &= \hat{\boldsymbol{m}}^{(k)}_{\hat{\eta}}\cdot\hat{\boldsymbol{x}}^{(k)}_{d}, \\ d &= c^2_{\hat{\xi}}+c^2_{\hat{\eta}}-1.\end{aligned} \tag{4-45}$$

The dual basis vectors $\hat{\boldsymbol{m}}^{(k)}_{\hat{\xi}_i}$ on the plane formed by two vectors $\hat{\boldsymbol{x}}^{(k)}_{\hat{\xi}}$ and $\hat{\boldsymbol{x}}^{(k)}_{\hat{\eta}}$

(see Fig. 4.6(d)) are defined as

$$\hat{\boldsymbol{m}}^{(k)}_{\hat{\xi}_i} \cdot \hat{\boldsymbol{x}}^{(k)}_{\hat{\xi}_j} = \delta^i_j, \quad \hat{\boldsymbol{m}}^{(k)}_{\hat{\xi}_i} \cdot \hat{\boldsymbol{n}}^{(k)} = 0 \text{ with } i,j = 1,2 \tag{4-46}$$

and $\hat{\boldsymbol{n}}^{(k)} = \hat{\boldsymbol{x}}^{(k)}_{\hat{\xi}} \times \hat{\boldsymbol{x}}^{(k)}_{\hat{\eta}} / \left\| \hat{\boldsymbol{x}}^{(k)}_{\hat{\xi}} \times \hat{\boldsymbol{x}}^{(k)}_{\hat{\eta}} \right\|$.

To transfer the membrane strains from the quadrilateral subdomains to the polygonal shell element (in the final element natural coordinate system), we calculate

$$\tilde{e}^{m(k)}_{ij} = \tilde{e}^{m(k)}_{\hat{n}\hat{l}} \left(\mathbf{g}_i \cdot \hat{\mathbf{g}}^{\hat{n}} \right) \left(\mathbf{g}_j \cdot \hat{\mathbf{g}}^{\hat{l}} \right), \quad i,j = 1,2 \text{ and } \hat{n},\hat{l} = 1,2 \tag{4-47}$$

where $\hat{\mathbf{g}}^{\hat{n}} \cdot \hat{\mathbf{g}}_{\hat{l}} = \delta_{\hat{n}\hat{l}}$. When the geometry of the quadrilateral subdomains is flat, the AMS field described in Eqs. 4-41, 4-42, and 4-43 becomes equivalent to that of standard two-dimensional plane elements [90]. It should be noted that Ho-Nguyen-Tan and Kim [98] emphasize that the linearized covariant membrane strains in each quadrilateral subdomain cannot be exactly transformed into the linearized covariant strains defined in the polygonal shell elements by a simple coordinate transformation. As a result, the proposed polygonal shell element may not pass membrane patch tests due to the use of different basis vectors in the polygonal domain and the quadrilateral subdomains. However, these authors also state that the numerical results indicate that the AMS field defined in Eq. 4-47 remains effective in practice for significantly reducing membrane locking in polygonal shell elements.

Finally, enforcing assumed membrane strains in Eq. 4-27 yields

$$\tilde{e}_{ij} = \tilde{e}^m_{ij} + \zeta e^{b1}_{ij} + \zeta^2 e^{b2}_{ij}, \text{ with } i,j = 1,2. \tag{4-48}$$

Polygons with 3 or 4 edges are not subdivided into quadrilateral subdomains and are equal to the "MITC3" [20] and "new MITC4+" [90] elements. Triangular elements also do not suffer from membrane locking due to its flat geometry [98]. For other polygonal elements ($n > 4$), the quadrilateral subdomains are used as the integration domains, once the AMS field is built by using these subdomains. The ASS field is also continuous in each quadrilateral subdomain. Although this polygonal shell element does not pass all patch tests unrestrainedly, four-point integration for each quadrilateral subdomain demonstrated good performance in numerical experiments [98].

4.3 Surface mesh generation

It is well known that mesh properties influence the convergence and accuracy of numerical approximation methods such as FEM. This is valid for any type of mesh, either 2D or 3D cases. Speaking of which, this work is even

inspired by the fact that different meshes may present improvements in relation to others – in particular, we are motivated by the expansion in the coverage of the optimized 2D polygonal mesh `PolyMesher` by Talischi et al. [7], once it has several advantages as discussed in the previous Chapters. However, it is seen that meshes modeling three-dimensional curved surfaces deserve even deeper investigation than the in-plane meshes already established. The creation of an arbitrary polygonal mesh with standardized element size and aspect ratio may be a complex task. Recalling that mesh generation is not the main focus of this work and considering the significance of this topic, which we believe warrants dedicated attention, a simple mesh generation approach is employed in this work. This approach consists of taking advantage of the consolidated 2D mesh generator `PolyMesher` and post-processing its planar result to transform it into a three-dimensional mesh. It should be noted that a mesh generated directly on the three-dimensional domain was not investigated in this study.

The first idea that arises is to orthogonally project the nodes on a parameterized surface. However, elements may suffer from stretching in steep regions, as demonstrated in Table 4.1, which yields poor finite element accuracy. An alternative is to perform a pre-processing step to adapt the node locations in order to concentrate them in the 2D domain according to surface slopes and then proceed to the orthogonal projection. However, it is still a challenge to propose a generic adaptation scheme that works for a variety of surface shapes. Therefore, we adopt the following surface parameterization to post-process the 2D mesh.

Consider a node position vector $\boldsymbol{x} = [x, y, z]^T$ that belongs to the shell surface. Let $\boldsymbol{\Psi} : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ be a parameterization function defined by

$$\boldsymbol{\Psi}(u, v) = [x, y, z]^T, \tag{4-49}$$

where (u, v) is the parameterization position vector in the 2D domain. The approach consists of first generating the 2D mesh in `PolyMesher` and then parameterizing $\boldsymbol{x} = [x, y, z]^T$ with (u, v) to the desired surface. We note that some parameterizations may cause FE node-numbering inversion and need to be corrected (here, we artificially multiply the FE force vector by -1). Indeed, this approach is only suitable for a set of parameterizable surfaces and can be involved depending on the shell shape, but when practicable, the mesh uniformity obtained is quite satisfactory (see Tab. 4.1). At the same time, orthogonal projection also cannot model any type of surface, especially those with multiple nodes on the same vertical direction. The last two rows of Tab. 4.1 show examples where it is not adequate to generate surface meshes using each of the approaches discussed. The parameterization functions and

2D domains will be specified for each shell example in the next Section.

Finally, the $\boldsymbol{V}_3^i$ vectors in Eq. 4-4 can be easily obtained by

$$\boldsymbol{V}_3^i = \frac{\boldsymbol{\Psi}_{,u}^i \times \boldsymbol{\Psi}_{,v}^i}{\left\| \boldsymbol{\Psi}_{,u}^i \times \boldsymbol{\Psi}_{,v}^i \right\|}. \tag{4-50}$$

Figure 4.7 illustrates some examples of shell meshes with the plotting of $\boldsymbol{V}_3^i$ vectors.

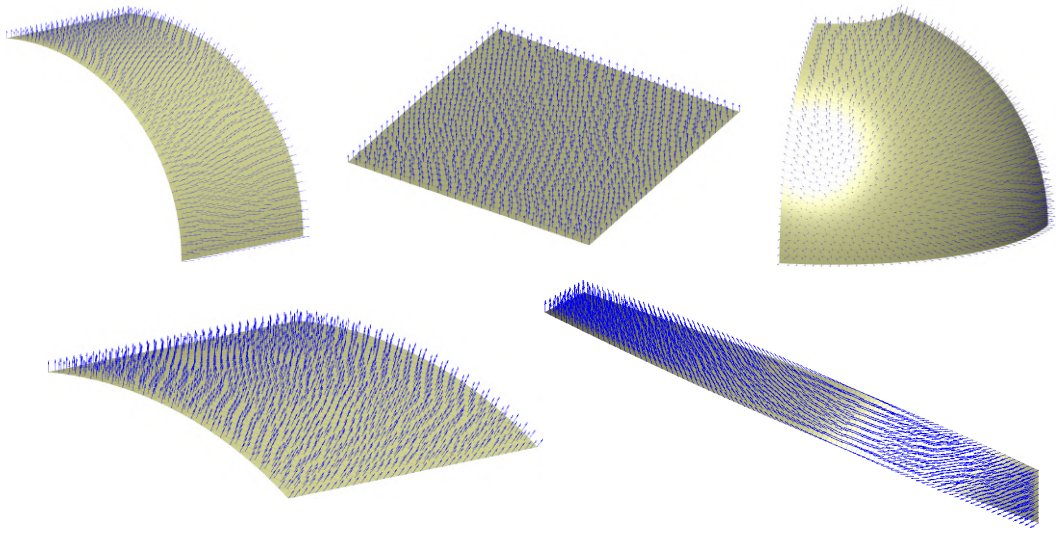
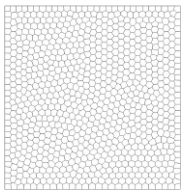
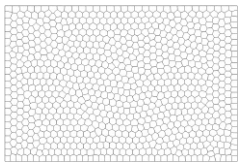
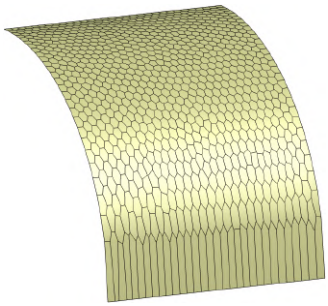
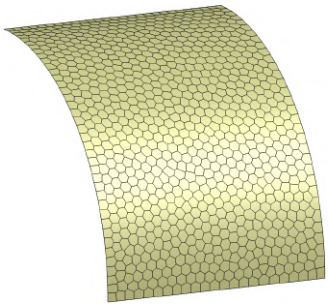
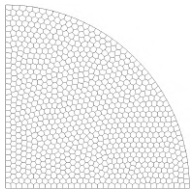
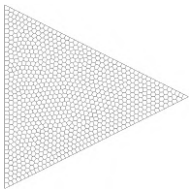
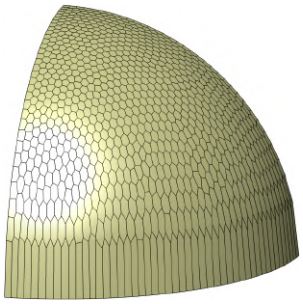
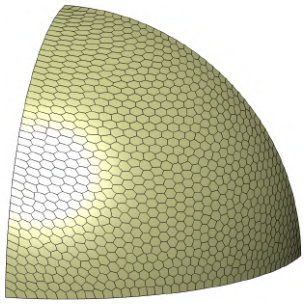
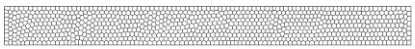
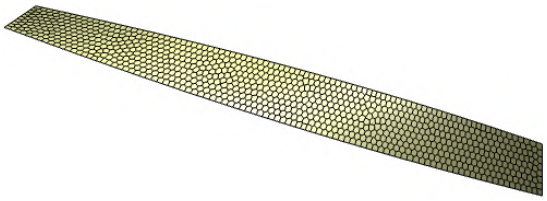
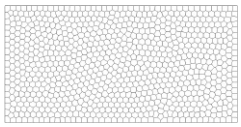
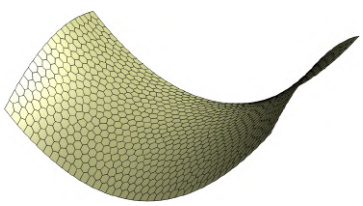


Figure 4.7: Examples of shell meshes with $\boldsymbol{V}_3^i$ vectors as per Eq. 4-50.

Table 4.1: Surface mesh generation – each mesh presents 900 elements.

Mesh domain	**Orthogonal projection**	**Surface parameterization**
2D		
3D		
2D		
3D		
2D	–	
3D	–	
2D		–
3D		–

4.4 Validation

In this Section, we showcase the efficiency of the polygonal degenerated shell elements and the proposed meshing approach. Both thick and thin cases are addressed, as well as shells with point and distributed loads, single- and doubly-curved surfaces, and flat shells modeling plate cases. For all examples, we utilize a 2D mesh generated by `PolyMesher` [7] with the same parameter setting as originally published, except for the maximum number of Lloyd iterations, which we increase to 500 to guarantee good mesh uniformity. Further discussion can be found in [98], including an analysis of the element's locking-free performance across multiple span-to-thickness ratios (L/h).

4.4.1 Pinched cylinders

The pinched cylinder examples comprise a cylindrical body with diametrically opposite compressive loads, as illustrated in Fig. 4.8. The softer version of this example has free ends, while the most common version has diaphragm restrictions on both cylinder ends.

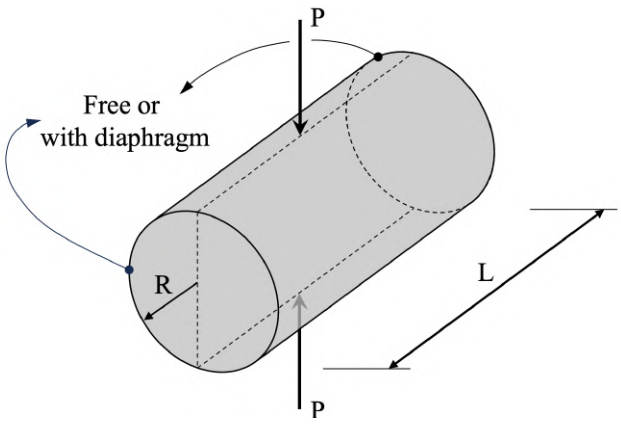


Figure 4.8: Pinched cylinder model.

Each mesh corresponds to one symmetric octant of the cylinder and has 900 elements with the following parameterization function (as per Eq. 4-49)

$$\mathbf{\Psi} = \begin{bmatrix} R\cos\left(\frac{u}{R}\right) \\ v \\ R\sin\left(\frac{u}{R}\right) \end{bmatrix}, \quad \text{for all cases,} \tag{4-51}$$

where $u \in \left[0, \frac{R\pi}{2}\right]$ and $v \in \left[0, \frac{L}{2}\right]$, i.e., the 2D mesh domain is a $\frac{R\pi}{2} \times \frac{L}{2}$ rectangle. Examples of these 2D mesh domains can be seen in the first row of Tab. 4.1.

4.4.1.1 Pinched cylinder with free ends

In this example, we will analyze the behavior of the element with variation in thickness. Ashwell and Sabir [104] performed both thick and thin cases for this example. Material and geometric parameters are summarized in Tab. 4.2.

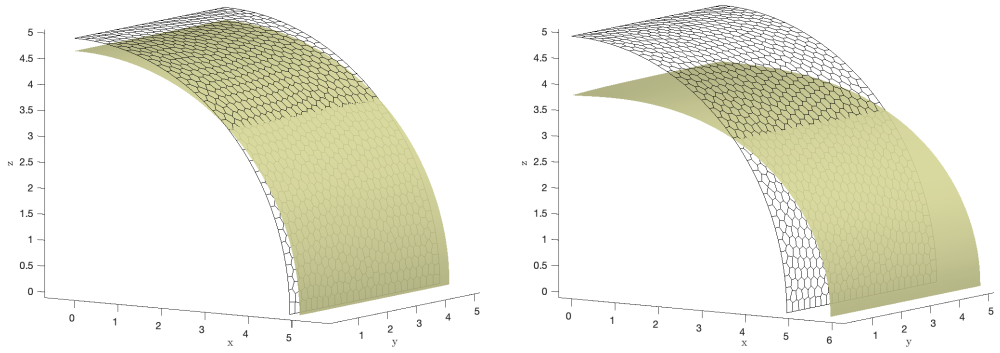
Table 4.2: Pinched cylinder with free ends – Material and geometric parameters by [104] (Note: AF is an amplification factor to apply on the deformation results).

Parameter	**Thin Case**	**Thick Case**
E (lbf/in^2)	10.5×10^6	10.5×10^6
ν	0.3125	0.3125
L (in)	10.35	10.35
R (in)	4.953	4.953
h (in)	0.01548	0.094
P (lbf)	0.1	100
AF	10	10

Table 4.3 displays the maximum deflection in the direction of the loads along with the reference values and relative errors. As can be seen, both thickness cases demonstrate satisfactory results. Figure 4.9 shows the amplified deformation behavior of the shells.

Table 4.3: Pinched cylinder with free ends – Maximum deflection results (in).

Case	$\mathbf{w_{max}}$	**Reference value [104]**	**Relative error**
Thin	0.02459	0.02439	0.8127%
Thick	0.1138	0.1139	0.1303%



(a) Thin case (b) Thick case

Figure 4.9: Pinched cylinder with free ends – Amplified deformation results on one symmetric octant of the cylinder.

4.4.1.2 Pinched cylinder with diaphragm ends

This version of the cylinder is most commonly seen in the literature, where rigid diaphragms restrict both cylinder ends [105, 106]. The restriction fixes all translations in the plane of the circular ends and the rotation about the cylinder axial direction, but the translational DOF normal to this plane and other rotational DOFs are set free. Table 4.4 gives the parameters utilized. The results are displayed in Tab. 4.5 and in Fig. 4.10.

Table 4.4: Pinched cylinder with diaphragm ends – Material and geometric parameters by [105, 106] (Note: AF is an amplification factor to apply on the deformation results).

Parameter	**Value**
E	3.0×10^6
ν	0.3
L	600
R	300
h	3
P	1
AF	5.0×10^6

Table 4.5: Pinched cylinder with diaphragm ends – Maximum deflection results and reference values from [105, 106].

$\mathbf{w_{max}}$	**Reference value**	**Relative error**
1.838×10^{-5}	1.825×10^{-5}	0.7354%

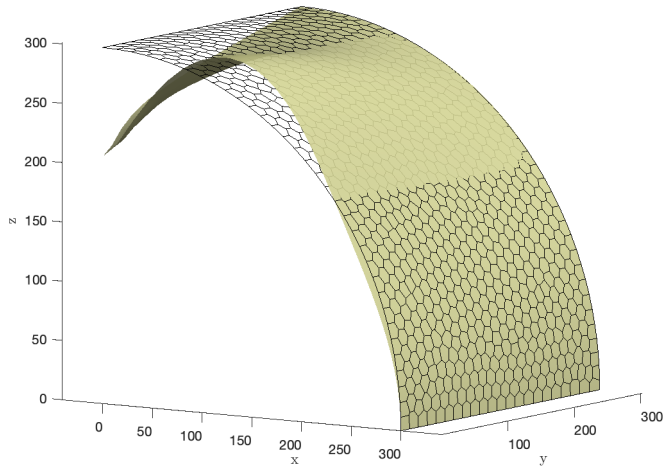


Figure 4.10: Pinched cylinder with diaphragm ends – Amplified deformation results on one symmetric octant of the cylinder.

4.4.2 Scordelis-Lo roof

The Scordelis-Lo roof is a well-known established benchmark with distributed load, recently encouraged by Krysl and Chen [107], who criticized the use of FE shell benchmarks with point loads. This example comprises a roof in the form of a cylindrical section, with both ends restricted by diaphragms, and a uniformly distributed load (q) applied over the shell surface (refer to Fig. 4.11). The parameters are given in Tab. 4.6. The simulation domain is the one-quarter symmetric segment of the roof, with a mesh of 3,600 elements. The parameterization function remains the same as in Eq. 4-51, but with $u \in \left[0, \frac{R\theta}{2}\right]$ and $v \in \left[0, \frac{L}{2}\right]$, i.e., the 2D mesh domain is a $\frac{R\theta}{2} \times \frac{L}{2}$ rectangle.

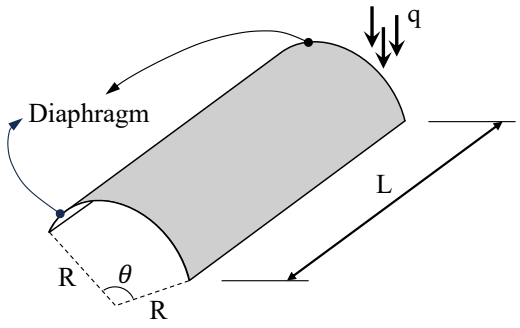


Figure 4.11: Scordelis-Lo roof – uniformly distributed load over the roof.

Table 4.6: Scordelis-Lo roof – Material and geometric parameters by [107] (Note: AF is an amplification factor to apply on the deformation results).

Parameter	**Value**
E (lbf/ft^2)	4.32×10^8
ν	0.0
L (ft)	50
R (ft)	25
h (ft)	0.25
θ	80°
q (lbf/ft^2)	90
AF	10

The results are shown in Tab. 4.7 and Fig. 4.12, where displacements in x and z directions are measured at the midpoint of the lateral edge of the roof. Although good correlation can be verified for this case where distributed load is employed, errors typically range from 0.1% to 2.1%. The reason is that we approximate the distributed load by simply dividing the total load by the number of nodes, instead of conducting the numerical integration and assembling of the FE force vector. Therefore, this approximation depends on the number of nodes and nodal distribution over the surface. The Scordelis-Lo roof has not yet been tested with arbitrary polygonal elements in the literature, but here we demonstrate that these elements are also capable of effectively addressing this type of shell benchmark.

Table 4.7: Scordelis-Lo roof – Displacement results (ft) at the midpoint of the lateral edge and reference values from [107].

	This work	**Reference value**	**Relative error**
u	-0.160	-0.159	0.433%
w	-0.3028	-0.3024	0.1466%

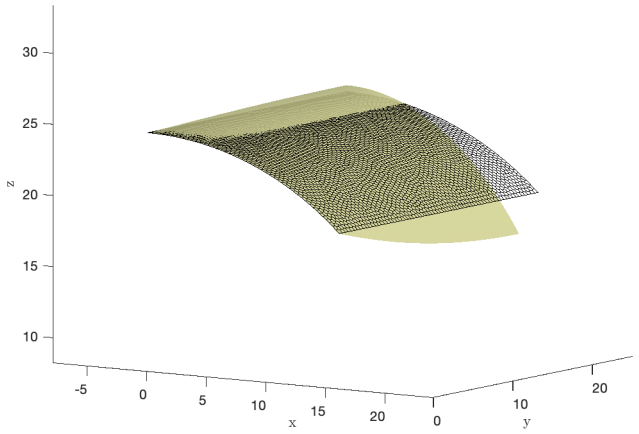


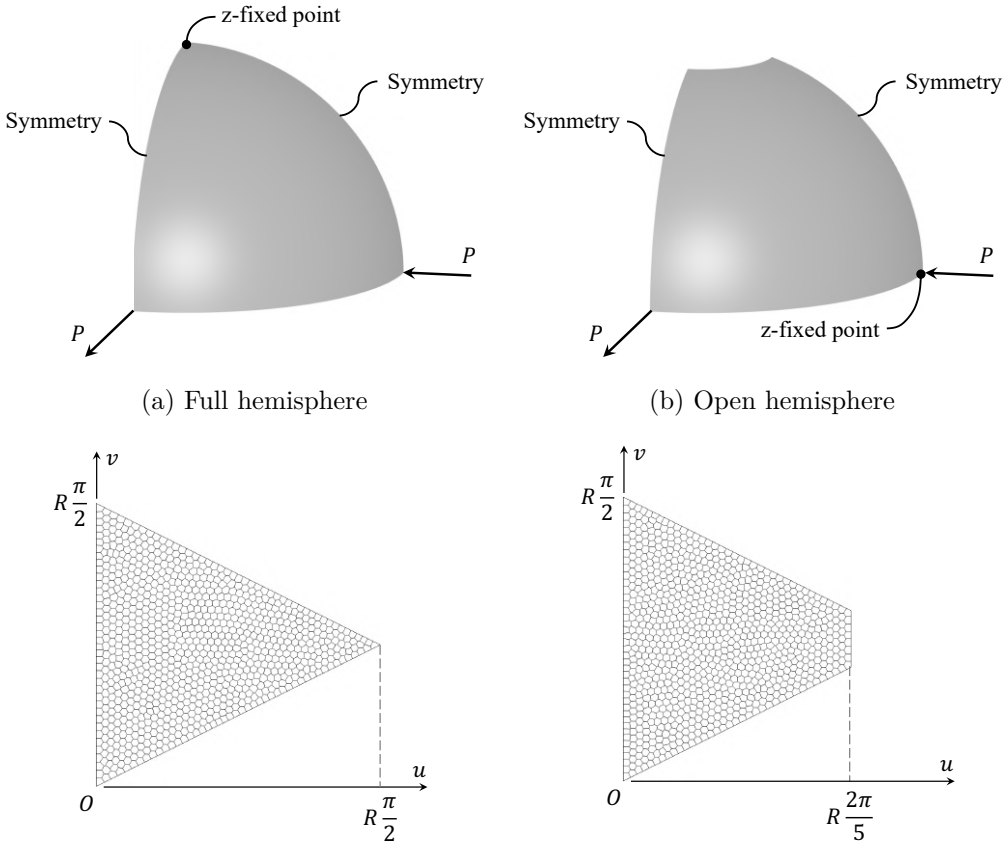
Figure 4.12: Scordelis-Lo roof – Amplified deformation results on a symmetric one-quarter segment of the roof.

4.4.3 Hemispheres

In order to evaluate the finite element behavior on a doubly-curved shell, the hemisphere examples are usually considered. The first case was introduced by Belytschko et al. [105] with a full hemispherical shell only restrained from presenting rigid body motions and subjected to compressive and tractive loads shifted by 90°. Then, a variation of this case was proposed by Simo et al. [106], with the same configuration, but with an 18° opening at the top of the hemisphere. Figure 4.13 shows both cases in detail and the parameters listed in Tab. 4.8 are the same for the two examples. In Fig. 4.13, the indicated "z-fixed" points are imposed to prohibit rigid body motions [98]. The mesh of each hemisphere case represents one symmetric quarter of the entire domain and has 900 elements with the parameterization function for both cases given by

$$\boldsymbol{\Psi} = \begin{bmatrix} R\cos\left(\frac{u}{R}\right)\cos\left(\frac{v-u/2}{R\pi/2-u}\frac{\pi}{2}\right) \\ R\cos\left(\frac{u}{R}\right)\sin\left(\frac{v-u/2}{R\pi/2-u}\frac{\pi}{2}\right) \\ R\sin\left(\frac{u}{R}\right) \end{bmatrix}, \tag{4-52}$$

where (u, v) is delimited by the 2D mesh domain represented in Fig. 4.13 for each case.



(a) Full hemisphere

(b) Open hemisphere

(c) Full hemisphere - 2D mesh domain

(d) Open hemisphere - 2D mesh domain

Figure 4.13: Hemisphere models and respective 2D mesh domains.

Table 4.8: Hemispheres – Material and geometric parameters by [105, 106].

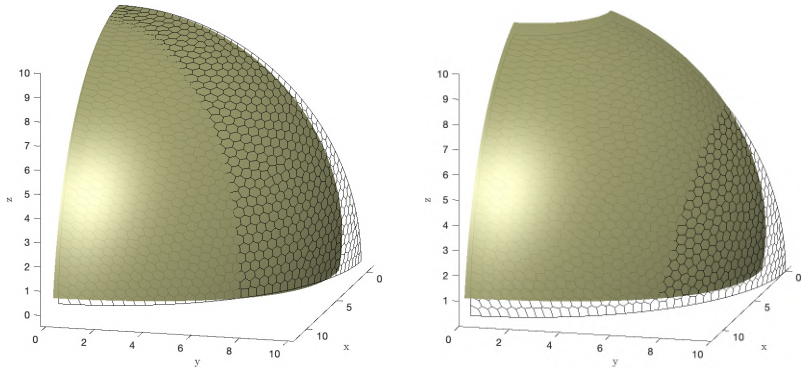
Parameter	**Value**
E	6.825×10^7
ν	0.3
R	10
h	0.04
P	2
AF	10

The measurement analyzed for both cases is the displacement at any load point in the direction of the applied force at this point, whose result and reference values are summarized in Tab. 4.9. Figure 4.14 exhibits the

deformation behavior of each shell example. Results emphasize the polygonal shell element efficiency in modeling doubly-curved surfaces.

Table 4.9: Hemispheres – Displacement results at a load point and respective load direction, with reference values from [105, 106].

Hemisphere case	$\mathbf{U_{load}}$	**Reference value**	**Relative error**
Full	0.0920	0.0924	0.492%
Open	0.093	0.094	1.0%



(a) Full hemisphere (b) Open hemisphere

Figure 4.14: Hemispheres – Amplified deformation results on one symmetric quarter of the hemisphere.

4.4.4 Twisted beam

Another benchmark is the twisted beam problem, where elements are tested in handling the warping of their own surfaces to accommodate the pretwist of the structure. Krysl and Chen [107], inspired by Belytschko et al. [105] and Simo et al. [106], also recommended this example in the sense of stimulating the use of distributed loads. As shown in Fig. 4.15, the body is clamped on one of its ends and is continuously twisted until its opposite end is shifted by 90°, where two load cases can be applied separately. The difference between the two load cases regards vertical versus horizontal force direction, both uniformly distributed along the final edge. The literature also encompasses two thickness cases for this shell example, whose parameters are found in Tab. 4.10. Again, a mesh with 900 elements is utilized, with the parameterization function, also applied to all cases, given as

$$\mathbf{\Psi} = \begin{bmatrix} u \\ v\cos\left(\frac{u}{L}\frac{\pi}{2}\right) \\ v\sin\left(\frac{u}{L}\frac{\pi}{2}\right) \end{bmatrix}, \tag{4-53}$$

where $u \in [0, L_x]$ and $v \in \left[-\frac{L_y}{2}, \frac{L_y}{2}\right]$, i.e., the 2D mesh domain is a $L_x \times L_y$ rectangle (see third row of Tab. 4.1).

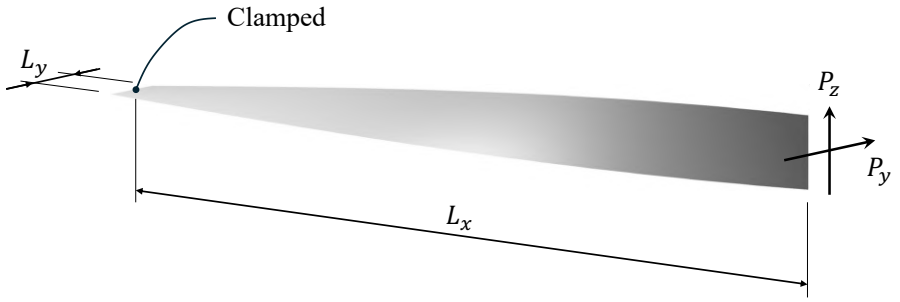


Figure 4.15: Twisted beam model.

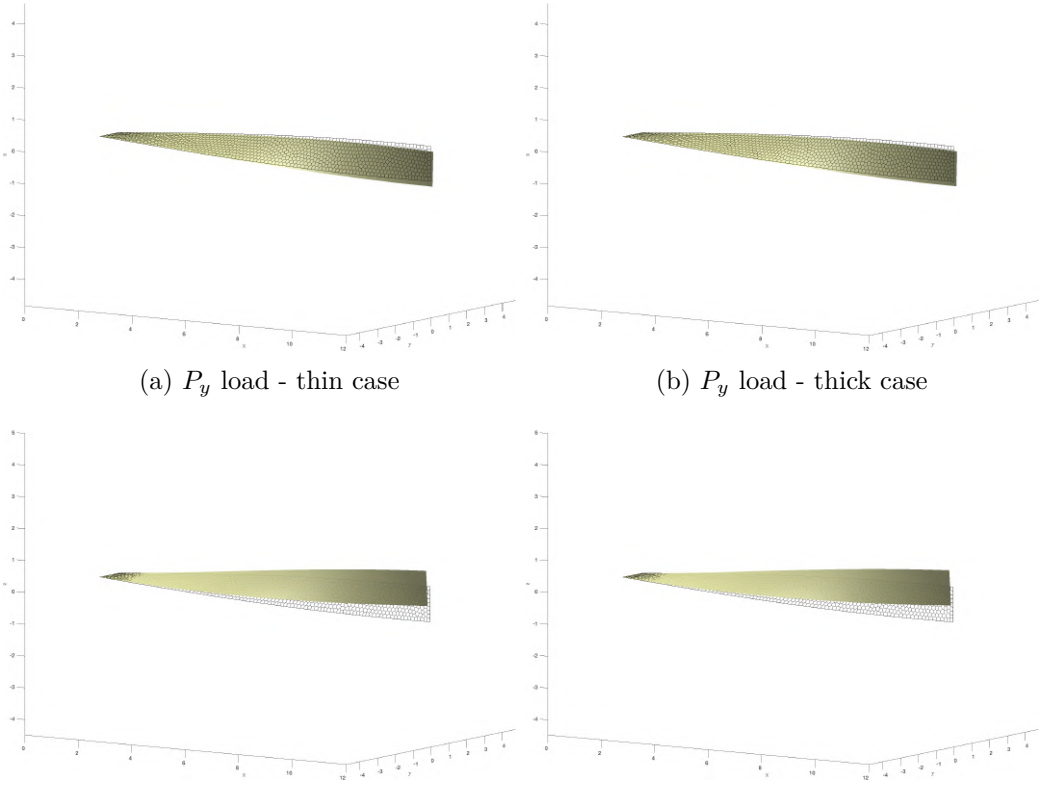
Table 4.10: Twisted beam – Material and geometric parameters by [107] (Note: AF is an amplification factor to apply on the deformation results).

Parameter	**Thin Case**	**Thick Case**
E (psi)	29×10^6	29×10^6
ν	0.22	0.22
L_x (in)	12	12
L_y (in)	1.1	1.1
h (in)	0.0032	0.32
$P_z = P_y$ (lbf)	1.0	1.0×10^{-6}
AF	100	100

The displacement is evaluated at the midpoint of the loaded edge, in the direction of the load case. As mentioned for the Scordelis-Lo roof problem, the twisted beams have also not been tested for arbitrary polygonal elements before, therefore, after investigation, we showcase the element satisfactory performance in another challenging benchmark. Table 4.11 and Fig. 4.16 display the results with relative errors and deformed configurations, respectively.

Table 4.11: Twisted beam – Displacement results (in) at the midpoint of the loaded edge in the direction of the load case, with reference values from [107].

Load case	**Thickness case**	$\mathbf{U_{load}}$ (10^{-3})	**Reference value**	**Relative error**
P_y	Thin	1.291	1.294	0.2465%
	Thick	1.748	1.754	0.3546%
P_z	Thin	5.244	5.256	0.2347%
	Thick	5.4146	5.424	0.1726%



(a) P_y load - thin case

(b) P_y load - thick case

(c) P_z load - thin case

(d) P_z load - thick case

Figure 4.16: Twisted beam – Amplified deformation results.

4.4.5 Shell as a plate example

Finally, we take the opportunity to show that the code also appropriately works for flat surfaces, i.e., plate structures. The example reproduced is the clamped square plate with a concentrated central load as described in

Section 2.3.1 and Fig. 2.5(b). The mesh is generated with 900 polygonal shell elements and no post-processing is necessary, once the structure is two-dimensional $\left(\mathbf{\Psi} = [u, v, 0]^T\right)$. Table 4.12 and Fig. 4.17 confirm that the degenerated polygonal shell elements succeed at addressing flat structures. Even though the error is greater than the one in Section 2.3.1 (0.52132%), the shell elements show rapid convergence performance, once the error difference is minor considering the disparity in the number of elements (16,000 plate elements were used in Section 2.3.1).

Table 4.12: Shell as a plate example – Normalized central deflection $\overline{\mathrm{w}}$, with reference value from [25].

$\overline{\mathbf{w}}$	**Reference value**	**Relative error**
0.0056437	0.0056012	0.75943%

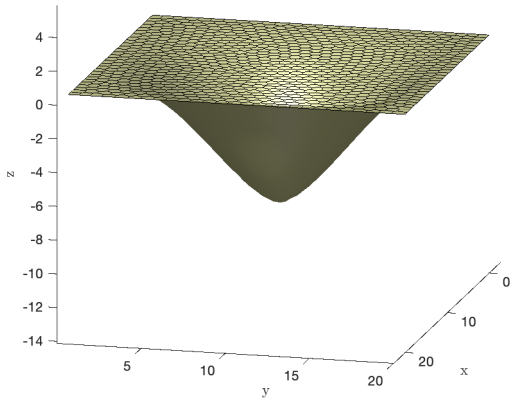


Figure 4.17: Shell as a plate example – Amplified deformation results.

5
Topology optimization applied to shell structures

In this Chapter, we first discuss the state-of-the-art scenario and then apply the validated polygonal shell element to the compliance topology optimization.

5.1
Literature review

The first evidence of topology optimization for shell structures with a polygonal element was found in 2020 by Chandrasekhar et al. [108]. In their work, a mesh of structured hexagonal elements was employed, where quadrilateral and triangular elements were used to allow straight boundaries not practical with structured hexagons. Although degenerated curved shell elements were used, with rational fraction shape functions, no locking treatments were discussed. Compliance topology optimization was performed therein with the Evolutionary Swarm Intelligence Firefly Algorithms (ESIFA) using optimality criteria, but only plates and a single-curved cylindrical shell roof were simulated.

Ho-Nguyen-Tan and Kim [45], the same authors who created the polygonal finite element utilized in the previous Chapter [98], conducted in 2022 a level-set topology optimization for compliance and stress minimization of shell structures using trimmed quadrilateral shell meshes. These meshes are obtained during the optimization process by intersecting a background quadrilateral shell mesh with the zero-isolines of a level set function (see Fig. 5.1). Along the boundaries of the shell structures, where the trimmed quadrilateral elements are formed, the polygonal shell elements with assumed strains of the last Chapter were employed. In this study, a p-norm aggregation method is applied to globally handle the stress constraints. In the same year, these authors disclosed a method for shape and topology optimization of shell structures [109] utilizing the trimmed quadrilateral meshing approach (same as in Fig. 5.1). Lastly, a year later, the same authors performed a level-set stress-constrained concurrent two-scale topology optimization of functionally graded cellular structures [110], where the trimmed quadrilateral mesh was employed again.

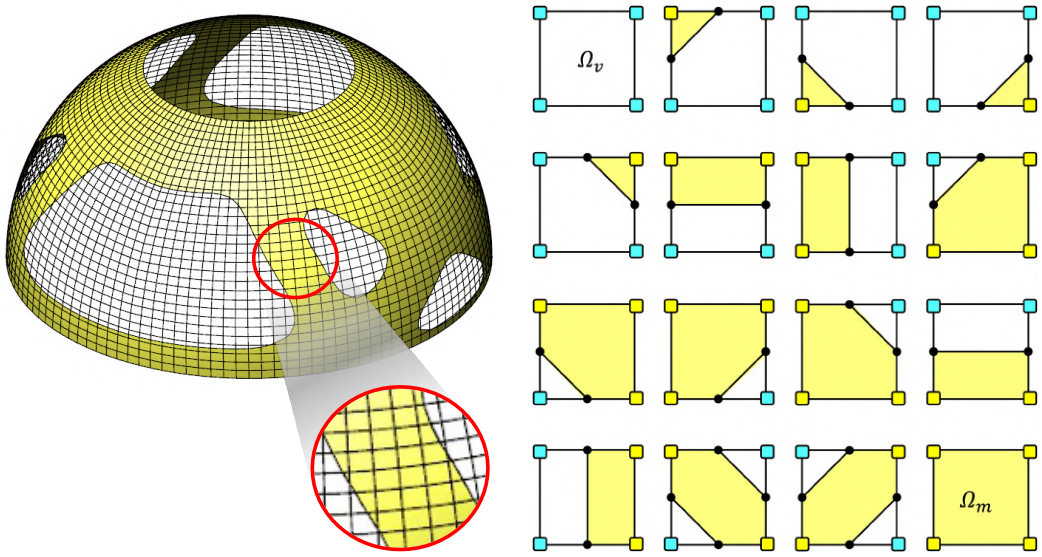


Figure 5.1: Trimmed quadrilateral meshes by Ho-Nguyen-Tan and Kim [45] – the way polygonal elements have been used for topology optimization in the literature (adapted from [109]).

From the best of our knowledge, there is no evidence of any sort of topology optimization applied to shell structures using arbitrary polygonal elements throughout the entire design domain. Although the trimmed quadrilateral mesh is a prospective alternative to provide clear boundaries, the trimming methodology and mesh adaptivity may be involved. In addition, as discussed in the previous Chapters, arbitrary polygonal meshes avoid the checkerboard pattern and hinged-connection instabilities, often occurring in quadrilateral meshes. Therefore, in this Chapter, we present topology optimization results for shell structures using only arbitrary polygonal meshes, which we believe deserves dedicated consideration as available for membrane [6], plate [44] and 3D solid [72, 111] elements.

5.2 Compliance minimization

In this Section, we first introduce the optimization problem and then we emphasize some small but important modifications necessary to adjust the algorithms.

5.2.1 Problem statement

The topology optimization problem to minimize the structural compliance in shells is formulated as

$$\min_{\mathbf{z}} \quad \mathbf{F}^T\mathbf{U}, \tag{5-1}$$

$$\text{s.t.} \quad \begin{cases} \frac{\mathbf{A}^T m_V(\mathbf{y})}{\mathbf{A}^T\mathbf{1}} - \overline{V} \leq 0, \\ 0 \leq z_e \leq 1, \quad e = 1, \ldots, n_e \end{cases} \tag{5-2}$$

$$\text{with:} \quad \mathbf{KU} = \mathbf{F},$$

where $\mathbf{A} = \{|\Omega_e|\}_{e=1}^{n_e}$, V^e is the element volume and $\overline{V}$ is the volume constraint value.

5.2.2 Minor modifications in `PolyTop` and `PolyFilter` codes

The compliance minimization problem is configured in the same overall form as done for plate structures in Sec. 3.2, which in turn follows almost the same pattern of `PolyTop` [6]. In other words, despite the finite element and meshing procedure defined in the previous Chapter, the filter matrix, sensitivity analysis, interpolation functions, elimination of the SIMP penalization factor continuation, and any other aspect not mentioned hereinafter are maintained the same. However, we shall emphasize some minor extra modifications necessary to adjust the codes for a three-dimensional curved surface domain.

The first regards the three-dimensionality of the mesh when computing the element centroids (`ElemCtrd` in `PolyFilter` code). We point out that one should not forget to update the number of centroid coordinates from two to three, as well as the calculation of these coordinates, which we propose as simply as follows

$$\overline{\boldsymbol{x}} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i \tag{5-3}$$

Although this may seem obvious, the way it was originally computed in `PolyTop` is only suitable for two-dimensional domains. Still in `PolyFilter` code, the original search for element centroids inside a circle with the input filter radius is no longer valid, since the circle is a two-dimensional locus. Then, we propose that a sphere of input filter radius is used to determine the element centroids affected by the filtering scheme. In other words, $\texttt{dist}_i^e$ in the `DistPntSets` function, which accounts for the distance between the element e and i centroids, is calculated as

$$\texttt{dist}_i^e = \sqrt{\left(\overline{x}^e - \overline{x}_i\right)^2 + \left(\overline{y}^e - \overline{y}_i\right)^2 + \left(\overline{z}^e - \overline{z}_i\right)^2}. \tag{5-4}$$

Lastly, in `PolyTop` code, each element area $|\Omega_e|$ must be computed to be used in Eq. 5-2. Again, the original form of this calculation is only valid for 2D domains. Therefore, we propose an element area approximation in the FE

numerical integration routine by including

$$|\Omega_e| = \int_{\Omega^e} \left\| \frac{\partial \boldsymbol{x}}{\partial \xi} \times \frac{\partial \boldsymbol{x}}{\partial \eta} \right\| d\xi d\eta, \tag{5-5}$$

where $\partial \boldsymbol{x}/\partial \xi$ and $\partial \boldsymbol{x}/\partial \eta$ can be determined by the first and second rows of the Jacobian matrix (Eq. 4-17) evaluated at $\zeta = 0$, i.e., the shell element midsurface area.

5.3 Validation and case studies

Several examples are proposed to illustrate the efficiency of arbitrary polygonal meshes in topology optimization applied to shell structures. The goal of conducting the following case studies is to validate and exploit the advantages of such meshes. For all experiments, the SIMP penalization factor is $p = 3$, the convergence tolerance on design variables is 0.01, and the maximum number of optimization iterations is 150. The MMA optimization parameters remain unchanged from `PolyTop` [6], i.e., the allowable move step as 0.2 and the exponent $\eta_{OC} = 0.5$ in the Optimality Criteria (OC) update scheme.

5.3.1 Shell roof

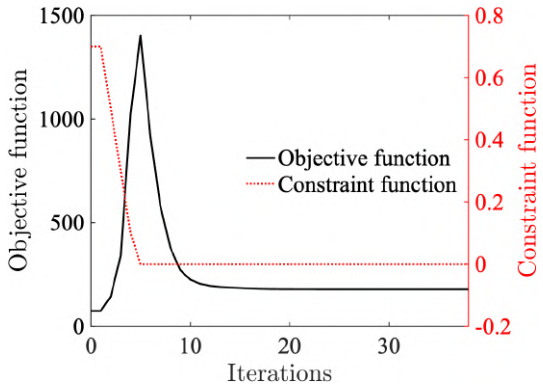
The first example consists of a single-curved cylindrical roof recently proposed by Wen et al. [112]. The geometry and load information are described in Fig. 5.2(a), where all five loads have the same value P and the roof is simply supported at its four corners. Other material, optimization and mesh parameters are given in Tab. 5.1. Due to symmetry, only one quarter of the structure is optimized. The reference result is available in Fig. 5.2(b) for comparison with the resulting topology from the present work in Fig. 5.2(c), which shows a similar result. Figure 5.2(d) shows the process evolution, which took 38 iterations to achieve the convergence tolerance in 27 seconds[1], rendering a final volume fraction of 0.3 of the domain total volume.

[1]Using MATLAB® 2023a on a MacBook Air M1 2020, 8 Core CPU and GPU, 8GB RAM and 512 SSD, with an operating system macOS Sequoia 15.5.

Table 5.1: Shell roof – Material, optimization and mesh parameters by Wen et. al [112].

Parameter	**Value**
E	2100
ν	0.3
h	5
P	100
$\overline{V}$	0.3
$N_{elements}$	5,000
R_{filter}	0.3
$\boldsymbol{z}_{init}$	1

(a) Boundary conditions (b) Reference (c) Present work



(d) Convergence plot

Figure 5.2: Shell roof topology optimization – boundary and loading conditions (a), reference results (b), topology from the present work (c) and the evolution of compliance objective function and material volume constraint (d). Figs. (a) and (b) were taken from Wen et al. [112].

5.3.2 Curved square shell

Curved square shell examples are one of the most prevalent benchmark for topology optimization of shells [113, 114, 115, 116]. Sometimes this surface is parameterized as an elliptic paraboloid, a portion of the two-sheeted hyperboloid or even spherical surfaces. We reproduce in this work the hyperbolic shell example by Pan et al. [117], where all shell edges are simply supported and five equal loads are applied (see Fig. 5.3(a)). However, the exact geometric characteristics are not made available by the authors, so we herein utilize a surface approximation guess as follows

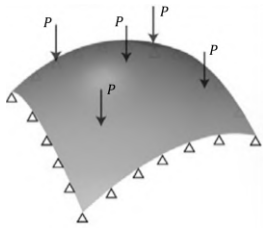
$$\mathbf{\Psi} = \begin{bmatrix} u \\ v \\ -(\frac{u}{10})^2 - (\frac{v}{10})^2 + 12.5 \end{bmatrix}, \tag{5-6}$$

where $u, v \in [-25, 0]$, i.e., the 2D mesh domain is a 25×25 square. Indeed, this three-dimensional mesh generation is an orthogonal projection, not a surface parameterization as discussed before. However, since there are no highly steep regions on the surface in Fig. 5.3(a), the qualitative results of the topology optimization should not be compromised.

The mesh generated respresents one symmetric quarter of the shell. Table 5.2 summarizes the information utilized to run this case. As Figs. 5.3(b) and 5.3(c) demonstrate, the algorithm also accomplishes desired topologies for doubly-curved shells. The values of the objective and constraint functions during the optimization process is plotted in Fig. 5.3(d).

Table 5.2: Curved square shell – Material, optimization and mesh parameters by Pan et al. [117].

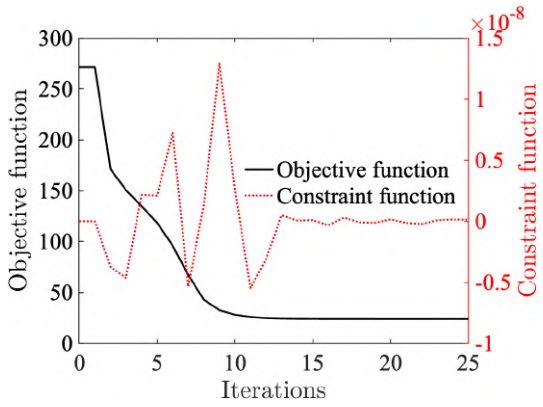
Parameter	**Value**
E	2100
ν	0.3
h	5
P	100
$\overline{V}$	0.3
$N_{elements}$	5,000
R_{filter}	0.01
$\boldsymbol{z}_{init}$	0.3



(a) Boundary conditions

(b) Reference

(c) Present work



(d) Convergence plot

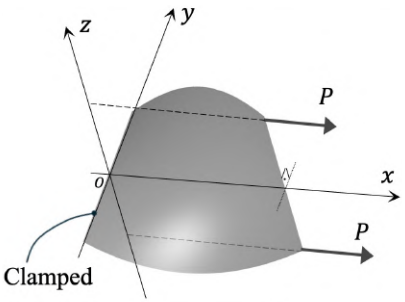
Figure 5.3: Curved square shell topology optimization – boundary and loading conditions (a), reference results (b), topology from the present work (c) and the evolution of compliance objective function and material volume constraint (d). Figs. (a) and (b) were taken from Pan et al. [117].

5.3.3 Twisted beam

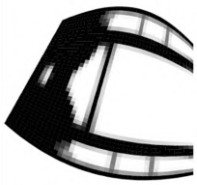
The twisted beam, as studied in the previous Chapter, has also been optimized by Long et al. [73]. The differences from the FE example are the load and the beam aspect ratio (now, it is 1×1) – refer to Fig. 5.4(a). Table 5.3 shows the input parameters, while Figs. 5.4(c) and 5.4(d) show the topology similarity with the reference result (5.4(b)) and the optimization evolution, respectively. Note that we purposely utilize a coarse mesh in this example to reproduce the same resolution from the reference article (same number of elements are employed).

Table 5.3: Twisted beam – Material, optimization and mesh parameters by Long et al. [73].

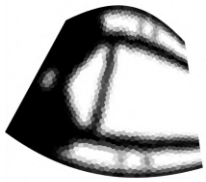
Parameter	**Value**
E	2.1×10^8
ν	0.3
h	0.01
P	1
$\overline{V}$	0.5
$N_{elements}$	1,600
R_{filter}	0.05
$\boldsymbol{z}_{init}$	1



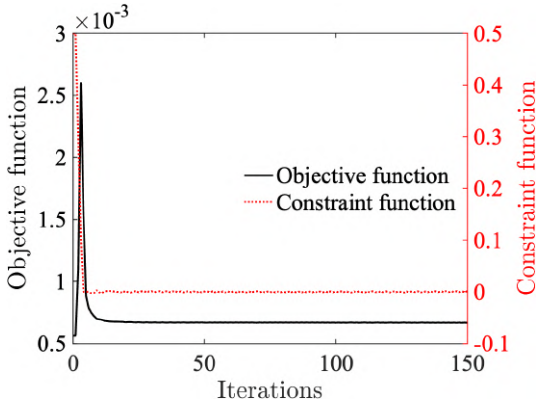
(a) Boundary conditions



(b) Reference



(c) Present work



(d) Convergence plot

Figure 5.4: Twisted beam topology optimization – boundary and loading conditions (a), reference results (b), topology from the present work (c) and the evolution of compliance objective function and material volume constraint (d). Fig. (b) was taken from Long et al. [73].

5.3.4 Saddle shell

Another common benchmark is the saddle shell [118, 117, 112], defined by a hyperbolic paraboloid surface. We attempt to replicate the example by Pan et al. [117], as in Fig. 5.5(a), with the parameters stated as in Tab. 5.4. All edges of the saddle shell are clamped and the optimization domain is one symmetric quarter of the structure. Again, the geometry information is not available in the referred article, so the surface was approximated here by

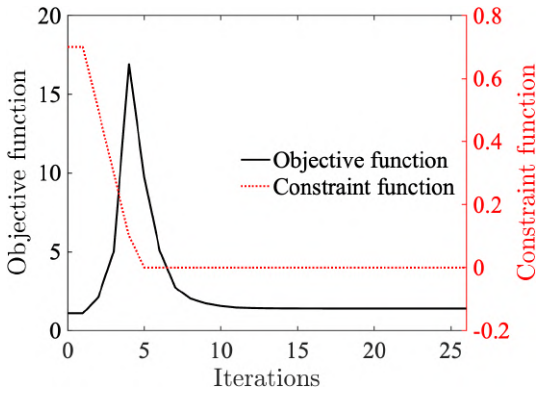
$$\mathbf{\Psi} = \begin{bmatrix} u \\ v \\ (\frac{u}{7.5})^2 - (\frac{v}{5})^2 \end{bmatrix}, \tag{5-7}$$

where $u, v \in [-25, 0]$, that is, the 2D mesh domain is a 25×25 square. Although orthogonal projection is used again, and now with a surface having quite steep regions, Figs. 5.5(b) and 5.5(c) still present consonant results.

Table 5.4: Saddle shell – Material, optimization and mesh parameters by Pan et al. [117].

Parameter	**Value**
E	2100
ν	0.3
h	5
P	100
$\overline{V}$	0.3
$N_{elements}$	10,000
R_{filter}	0.05
$\boldsymbol{z}_{init}$	1

(a) Boundary conditions (b) Reference (c) Present work



(d) Convergence plot

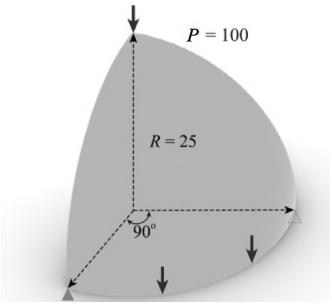
Figure 5.5: Saddle shell topology optimization – boundary and loading conditions (a), reference results (b), topology from the present work (c) and the evolution of compliance objective function and material volume constraint (d). Figs. (a) and (b) were taken from Pan et al. [117].

5.3.5 Hemisphere

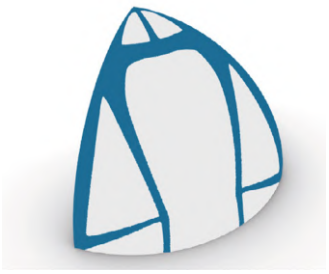
Wen et al. [112] studied the topology optimization on one quarter of the hemisphere shell, as shown in Fig. 5.6(a). The hemisphere corners are fixed (all DOFs restrained) and three concentrated loads are applied. Other information is outlined in Tab. 5.5. This is a particular result that the topology outcome from the present work (Fig. 5.6(c)) is not much similar to the reference result (Fig. 5.6(b)). This may occur because the reference article utilizes meshes that are adaptively and locally refined along the density boundaries, enabling distinct topological paths. However, we still consider the results to be somehow comparable and a strict comparison should only be made upon same techniques being employed.

Table 5.5: Hemisphere – Material, optimization and mesh parameters by Wen et. al [112].

Parameter	**Value**
E	2100
ν	0.3
h	5
P	100
$\overline{V}$	0.3
$N_{elements}$	10,000
R_{filter}	0.5
$\boldsymbol{z}_{init}$	1

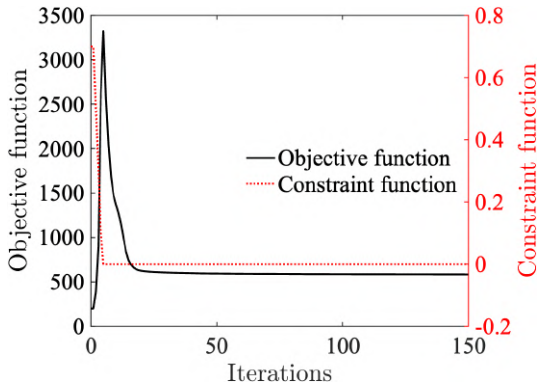


(a) Boundary conditions



(b) Reference

(c) Present work

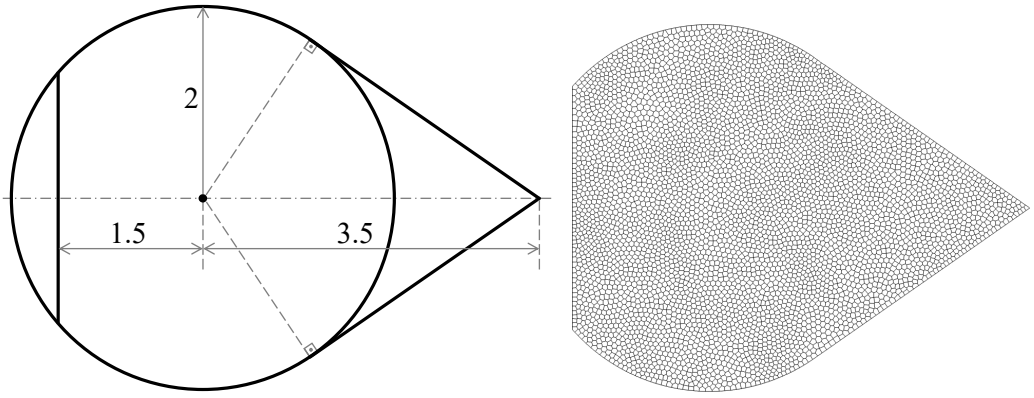


(d) Convergence plot

Figure 5.6: Hemisphere topology optimization – boundary and loading conditions (a), reference results (b), topology from the present work (c) and the evolution of compliance objective function and material volume constraint (d). Figs. (a) and (b) were taken from Wen et al. [112].

5.3.6
Leaf-shaped shell

The last example is a case study proposed in this work to demonstrate the advantages of `PolyMesher` [7] in modeling non-Cartesian domains, such as the curved boundaries of a leaf-inspired shell. First, a two-dimensional (2D) domain is generated with the dimensions shown in Fig. 5.7(a), resulting in the 2D mesh presented in Fig. 5.7(b).



(a) Dimensions (b) 2D mesh

Figure 5.7: Leaf-shaped shell – 2D domain.

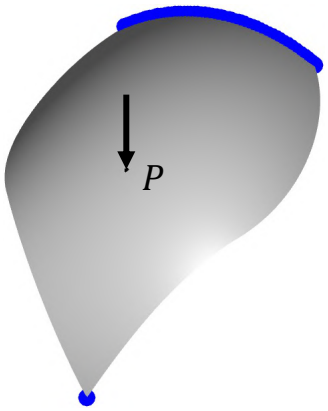
Then, the shell surface is parameterized using the following function

$$\mathbf{\Psi} = \begin{bmatrix} R_2 \sin\left(\frac{x_1\alpha_2}{R_2}\right) \\ y_1 \\ R_2 \cos\left(\frac{x_1\alpha_2}{R_2}\right) \frac{z_1}{\max(z_1)} \end{bmatrix}, \tag{5-8}$$

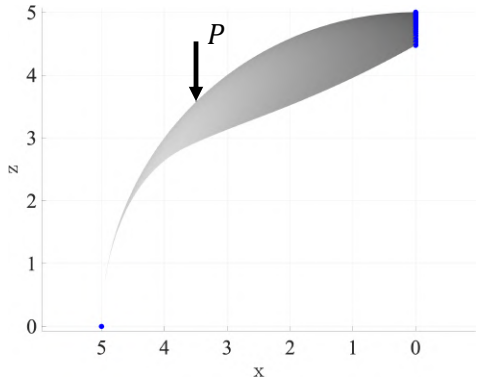
with

$$\begin{cases} x_1 = u, \\ y_1 = R_1 \sin\left(\frac{v\alpha_1}{R_1}\right), \\ z_1 = R_1 \cos\left(\frac{v\alpha_1}{R_1}\right), \end{cases} \tag{5-9}$$

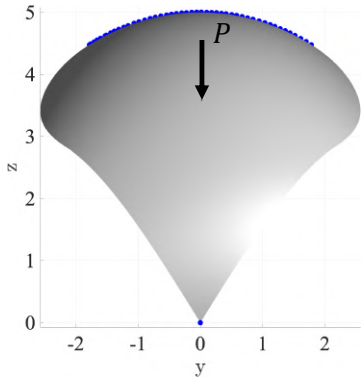
where u and v belong to the 2D domain shown in Fig. 5.7, with $R_1 = 4$, $R_2 = 5$, $\alpha_1 = 4\pi/9$, and $\alpha_2 = \pi/2$. The resulting surface is illustrated in Fig. 5.8(a), along with the boundary conditions: the blue nodes are fixed, and a point load P is applied along the symmetry axis at a horizontal distance of 3.5 from the upper clamped edge of the shell. Auxiliary views are provided in Figs. 5.8(b) and 5.8(c) for better visualization. The parameters used in this example are summarized in Table 5.6.



(a) Generated surface



(b) Side view



(c) Front view

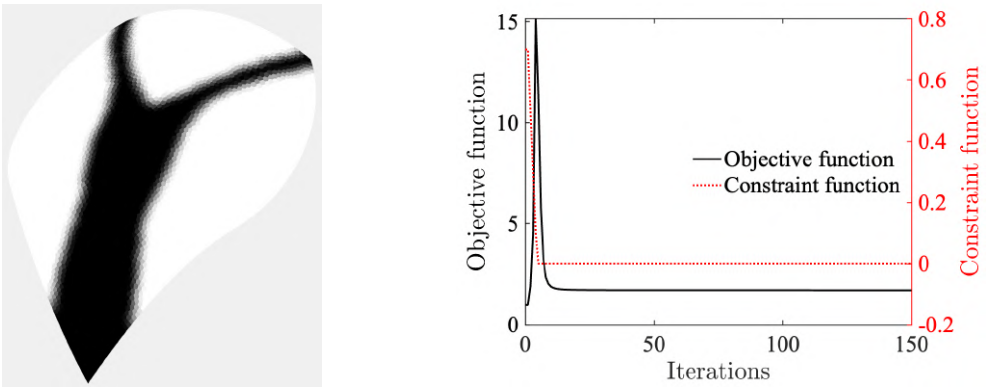
Figure 5.8: Leaf-shaped shell – surface with the boundary and loading conditions (a) and auxiliary views (b) and (c).

Table 5.6: Leaf-shaped shell – Material, optimization and mesh parameters.

Parameter	**Value**
E	2100
ν	0.3
h	0.5
P	10
$\overline{V}$	0.3
$N_{elements}$	5,000
R_{filter}	0.2
$\boldsymbol{z}_{init}$	1

The resulting topology is shown in Fig. 5.9(a). It can be observed that a larger branch develops to stiffen the structure in the region where the

membrane effects of the shell are likely to be more pronounced, due to the surface slope and the load direction. The plot in Fig. 5.9(b) shows that the process converges rapidly to the final topology but continues until the maximum iteration limit because of small oscillations around the tolerance.



(a) Resulting topology

(b) Convergence plot

Figure 5.9: Leaf-shaped shell topology optimization – final topology obtained in the present work (a) and the evolution of compliance objective function and material volume constraint (b).

6
Conclusions

In this work, we aimed at the topology optimization for plate and shell structures utilizing arbitrary polygonal finite elements, while the main motivation was to empower easy-to-use, open-source educational codes developed from the well-known `PolyTop` software [6]. To do so, the main obstacle arises from the finite element perspective, since shear and/or membrane locking overestimate the element stiffness when the span-to-thickness ratio (L/h) increases, artificially stiffening the structure. The arbitrary polygonal mesh from the open-source `PolyMesher` code [7] has the advantages of shaping complex geometries and avoiding numerical instabilities in topology optimization, such as the checkerboard pattern and hinged connections. To systematically address this subject, we divided the study into plate and shell developments.

6.1
Remarks on plate structures

To circumvent shear locking in the polygonal plate finite elements, many solutions have been proposed in the literature over the past few years. We utilized the technique proposed by Nguyen-Xuan [27], in which the Timoshenko beam assumptions are enforced on each edge of a polygonal element. Nguyen-Xuan [27] originally proposed an element with the so-called piecewise-linear shape functions (the PRMn-PL element), but also investigated many other shape functions, such as the Wachspress utilized in `PolyTop` (the PRMn-W element). Although this author demonstrated that PRMn-PL slightly performs better than PRMn-W, the latter still passes all patch tests and shows a good convergence behavior. The PRMn-W element was successfully tested in this work in providing the displacement, moment and stress measurements for linear elastic isotropic plates.

The topology optimization of plates using the PRMn-W elements was then conducted first for the compliance minimization. Although this type of topology optimization of plates was already done in the literature with PRMn-PL element, the computational implementation was not available yet. Therefore, we disclosed simple instructions to modify the `PolyTop` code in order to enable the proper utilization in both thick and thin plate arrangements.

Then, we proceeded to investigate and fulfill a gap in the literature regarding the stress-constrained topology optimization of plates considering local stress constraints. The local nature of stress definition and existing literature suggest that a local methodology is the most effective approach to addressing the stress-constrained topology optimization problem.

By taking advantage of open-source codes once again, we proposed an extension of the Augmented Lagrangian method, previously demonstrated to be efficient for membrane and solid elements in the `PolyStress` code [13], to accommodate local stress-constrained topology optimization for plate elements. Any convex-polygon plate mesh can be considered using the proposed approach, including the quadrilateral and arbitrary polygonal meshes, which yielded consistent results and demonstrated viable scalability. Furthermore, the locking-free technique facilitated the evaluation of both thick and thin plate cases using the same algorithm without requiring additional adjustments.

Indeed, the Augmented Lagrangian (AL) method for local stress-constrained topology optimization applied to plates is computationally more expensive than its application to two-dimensional membrane problems in `PolyStress`, primarily due to the additional degree of freedom per node. However, global techniques yield competitive results only when the p values (from p-mean aggregation in a p-continuation scheme) exceed 100. They become comparable to the AL method only when the p value reaches 300, which demands significantly greater computational effort compared to the AL method (see [62] for further discussion). Although these values originally pertain to membrane elements, they reasonably suggest a similar discrepancy in efficiency for plate applications.

6.2 Remarks on shell structures

The literature still lacks more options for addressing polygonal shell elements, which must alleviate both shear and membrane locking. Considering the three-dimensional distortion of a polygonal element lied on a surface, we adopted a degenerated curved shell element with assumed shear and membrane strain fields, as proposed by Ho-Nguyen-Tan and Kim [98] for polygonal elements. The numerical implementation is more involved than for plate elements and an open-source code would probably be welcomed by the academic and scientific community. Another obstacle was to keep the mesh generation as simple as possible by using available and easy-to-use software. We proposed the utilization of `PolyMesher` to generate a 2D mesh and, then, post-process it into a 3D mesh through a surface parameterization that avoids

the stretching of elements over steep regions. We first validated the code by efficiently tackling cylindrical and doubly-curved shells.

Subsequently, the topology optimization for shell structures was examined in the context of compliance minimization. In addition to the modifications made in the FE Section of `PolyTop`, we provided some extra, but small adjustments relative to the filter scheme and the computation of the curved element areas. We emphasize, from the best of our knowledge, that there is no investigation reported in the literature regarding the topology optimization of shells fully meshed with arbitrary polygonal elements. Therefore, several shell arrangements were topologically optimized, either validated with the state-of-the-art results or evaluated as a case study. Although the code for the shell version of `PolyTop` is not available in this text, it will be released to the academic and scientific community in due course as the final derivations of this study.

6.3 Prospective research

Throughout the present study, many challenges and drawbacks emerged. We conclude this work suggesting further improvements and developments for each of the studied structures. First, a framework to enrich the methodology for material and geometric nonlinear behaviors is hugely relevant, as well as encompassing composite structures. With the validated codes provided by this work, the development of further investigations of topology optimization for plates and shells is simplified. A few specific suggestions are separately organized in the following Sections.

6.4 Plate developments

- Implement and assess all available locking-free methods for polygonal plate elements;
- Apply the approach for dynamic topology optimization based on the `PolyDyna` [12], envisioning the design of plate damping layers;
- Tackle multi-material topology optimization of plates; and
- Evaluate the compliance minimization of plates under both local-stress and volume constraints.

6.5 Shell developments

- Develop strategies to reduce the computational cost of numerical integration, considering that each polygon is subdivided into n quadrilateral subdomains, with 2×2 Gauss points per subdomain, resulting in a total of $4 \times n$ integration points per polygon. This is higher than the $3 \times n$ integration points originally proposed in PolyTop, which uses a triangulation-based approach;
- Formulate a generic and easy-to-implement method for generating a regular polygonal mesh;
- Test the "graded seeds" methodology utilized in `PolyMesher` and `PolyDyna` to concentrate seeds by a mathematical procedure that takes the surface gradient into account for detecting slopes;
- Incorporate the most up-to-date MITC4 alternatives in the quadrilateral subdomains of the polygonal shell elements to enhance membrane behavior; and
- Include the drilling degree of freedom to enable connections with beams, applied moments or normal rotational boundary conditions.

Bibliography

[1] BENDSØE, M. P.; SIGMUND, O.. **Topology Optimization - Theory, Methods, and Applications**. Springer Berlin, Heidelberg, 2002.

[2] HUANG, X.; XIE, M.. **Evolutionary Topology Optimization of Continuum Structures: Methods and Applications**. John Wiley Sons, 2010.

[3] NOVOTNY, A. A.; SOKOŁOWSKI, J.. **An Introduction to the Topological Derivative Method**. SpringerBriefs in Mathematics. Springer International Publishing, 2020.

[4] HUANG, X.; XIE, Y. M.. **Bi-directional evolutionary topology optimization of continuum structures with one or multiple materials**. Comput Mech, 43:393–401, 2009.

[5] NOVOTNY, A. A.; SOKOŁOWSKI, J.. **Topological derivatives in shape optimization**. Interaction of Mechanics and Mathematics. Springer-Verlag, Berlin, Heidelberg, 2013.

[6] TALISCHI, C.; PAULINO, G. H.; PEREIRA, A. ; MENEZES, I. F. M.. **`PolyTop`: a Matlab implementation of a general topology optimization framework using unstructured polygonal finite element meshes**. Struct Multidisc Optim, 45:329–357, 2012.

[7] TALISCHI, C.; PAULINO, G. H.; PEREIRA, A. ; MENEZES, I. F. M.. **`PolyMesher`: a general-purpose mesh generator for polygonal elements written in Matlab**. Struct Multidisc Optim, 45:309–328, 2012.

[8] FERRER, A.. **SIMP-ALL: A generalized SIMP method based on the topological derivative concept**. International Journal for Numerical Methods in Engineering, 120:361–381, 2019.

[9] BENDSØE, M. P.; SIGMUND, O.. **Material interpolation schemes in topology optimization**. Archive of Applied Mechanics, 69:635–654, 1999.

[10] PEREIRA, A.; TALISCHI, C.; PAULINO, G. H.; MENEZES, I. ; CARVALHO, M. S.. **Fluid flow topology optimization in `PolyTop`: stability and computational implementation**. Struct Multidisc Optim, 54:1345–1364, 2016.

[11] SANDERS, E. D.; PEREIRA, A.; AGUILÓ, M. A. ; PAULINO, G. H.. **`PolyMat`: an efficient matlab code for multi-material topology optimization**. Proceedings of the Royal Society A, 58:2727–2759, 2018.

[12] GIRALDO-LONDOÑO, O.; PAULINO, G. H.. **`PolyDyna`: a Matlab implementation for topology optimization of structures subjected to dynamic loads**. Struct Multidisc Optim, 64:957–990, 2021.

[13] GIRALDO-LONDOÑO, O.; PAULINO, G. H.. **`PolyStress`: a Matlab implementation for local stress-constrained topology optimization using the augmented Lagrangian method**. Struct Multidisc Optim, 63:2065–2097, 2021.

[14] ALCAZAR, E.; RUSS, J. B. ; PAULINO, G. H.. **`PolyPlas`: a Python implementation of a topology optimization framework for plasticity with unstructured polygonal finite elements**. Struct Multidisc Optim, 68(153), 2025.

[15] COOK, R. D.; MALKUS, D. S.; PLESHA, M. E. ; WITT, R. J.. **Concepts and Applications of Finite Element Analysis**. Wiley, 4 edition, 2001.

[16] DUARTE, D. S.; MENEZES, I. F. M.. **An educational MATLAB code for topology optimization of thick-thin plates using arbitrary polygonal meshes**, 2023. Paper presented at the 27th COBEM - International Congress of Mechanical Engineering, Florianópolis, 4–8 December 2023.

[17] DUARTE, D. S.; MENEZES, I. F. M.. **Topology optimization of thick–thin plate structures considering local stress constraints**. Engineering with Computers, 2025.

[18] RADWAŃSKA, M.; STANKIEWICZ, A.; WOSATKO, A. ; PAMIN, J.. **Plate and Shell Structures: Selected Analytical and Finite Element Solutions**. John Wiley Sons Ltd, 2017.

[19] VIDELA, J.; NATARAJAN, S. ; BORDAS, S. P. A.. **A new locking-free polygonal plate element for thin and thick plates based on**

Reissner-Mindlin plate theory and assumed shear strain fields. Computers & Structures, 220:32–42, 2019.

[20] LEE, P.-S.; BATHE, K.-J.. **Development of MITC isotropic triangular shell finite elements**. Computers & Structures, 82(11):945–962, 2004.

[21] BATHE, K.-J.; LEE, P.-S.. **Measuring the convergence behavior of shell analysis schemes**. Comput Struct, 89:285—301, 2011.

[22] ZIENKIEWICZ, O. C.; TAYLOR, R. L. ; TOO, J. M.. **Reduced integration technique in general analysis of plates and shells**. Int J Num Methods Eng, 3(2):275–290, 1971.

[23] HUGHES, T. J. R.; COHEN, M. ; HAROUN, M.. **Reduced and selective integration techniques in the finite element analysis of plates**. Nucl Eng Des, 46(1):203–222, 1978.

[24] HUGHES, T. J. R.; COHEN, M.. **The "heterosis" finite element for plate bending**. Computers & Structures, 9(5):445–450, 1978.

[25] ZIENKIEWICZ, O. C.; TAYLOR, R. L. ; ZHU, J. Z.. **The Finite Element Method: Its Basis and Fundamentals**. Butterworth-Heinemann, 2013.

[26] BREZZI, F.; BATHE, K.-J. ; FORTIN, M.. **Mixed-interpolated elements for Reissner-Mindlin plates**. Int. J. Numer. Meth. Eng., 28:1787—-1801, 1989.

[27] NGUYEN-XUAN, H.. **A polygonal finite element method for plate analysis**. Computers & Structures, 188:45–62, 2017.

[28] SOH, A.; LONG, Z. ; CEN, S.. **A new nine DOF triangular element for analysis of thick and thin plates**. Comput Mech, 24:408–417, 1999.

[29] FLOATER, M.; GILLETTE, A. ; SUKUMAR, N.. **Gradient bounds for wachspress coordinates on polytopes**. SIAM J Numer Anal, 52(1):515–532, 2014.

[30] FLOATER, M. S.. **Mean value coordinates**. Comp Aided Geom Des, 20(1):19–27, 2003.

[31] SUKUMAR, N.; MORAN, B. ; BELYTSCHKO, T.. **The natural element method in solid mechanics**. Int J Numer Meth Eng, 43(5):839–887, 1998.

[32] NGUYEN, N. V.; NGUYEN, H. X.; PHAN, D.-H. ; NGUYEN-XUAN, H.. **A polygonal finite element method for laminated composite plates**. Int J Mech Sci, 133:863–882, 2017.

[33] NGUYEN, N. V.; NGUYEN, H. X.; L, S. ; NGUYEN-XUAN, H.. **Geometrically nonlinear polygonal finite element analysis of functionally graded porous plates**. Advances in Engineering Software, 126:110–126, 2018.

[34] KATILI, I.; MAKNUN, I. J.; KATILI, A. M.; BORDAS, S. P. A. ; NATARAJAN, S.. **A unified polygonal locking-free thin/thick smoothed plate element**. Composite Structures, 219:147–157, 2019.

[35] WU, C.; CEN, S. ; SHANG, Y.. **Shape-free polygonal hybrid displacement-function element method for analyses of Mindlin–Reissner plates**. Engineering with Computers, 37:1975–1998, 2021.

[36] NGUYEN, S. H.; NAM, N. N.; HOANG, T.-D.; NGUYEN, T. N. ; NGUYEN-THOI, T.. **Alpha (α) assumed rotations and shear strains for spatially isotropic polygonal Reissner-Mindlin plate elements (αARS-Poly)**. Computers & Structures, 274, 2023.

[37] NGUYEN, S. H.; PHAN, D.-H.. **Selective element domain interpolation technique for assumed rotations and shear strains in polygonal finite element thick/thin plate analysis**. Thin-Walled Structures, 186, 2023.

[38] CHINOSI, C.. **Virtual elements for the Reissner-Mindlin plate problem**. Numerical Methods for Partial Differential Equations, 34(4):1117–1144, 2018.

[39] DA VEIGA, L. B.; MORA, D. ; RIVERA, G.. **Virtual elements for a shear-deflection formulation of Reissner-Mindlin plates**. Mathematics of Computation, 88(315):149–178, 2019.

[40] D'ALTRI, A. M.; PATRUNO, L.; DE MIRANDA, S. ; SACCO, E.. **First-order VEM for Reissner-Mindlin plates**. Computational Mechanics, 69:315–333, 2022.

[41] MENGOLINI, M.; BENEDETTO, M. F. ; ARAGÓN, A. M.. **An engineering perspective to the virtual element method and its interplay with the standard finite element method**. Computer Methods in Applied Mechanics and Engineering, 350:995–1023, 2019.

[42] REDDY, J. N.. **Theory and Analysis of Elastic Plates and Shells**. CRC Press, Boca Raton, 2nd edition, 2006. eBook published on 19 November 2006.

[43] HUANG, H.-C.. **Static and Dynamic Analyses of Plates and Shells: Theory, Software and Applications**. Springer Science & Business Media, illustrated edition, 2012.

[44] PHAM, Q. H.; PHAN, D. H.. **Polygonal topology optimization for Reissner–Mindlin plates**. Engineering with Computers, 38, 2022.

[45] HO-NGUYEN-TAN, T.; KIM, H.-G.. **Level set-based topology optimization for compliance and stress minimization of shell structures using trimmed quadrilateral shell meshes**. Computers and Structures, 259, 2022.

[46] SVED, G.; GINOS, Z.. **Structural optimization under multiple loading**. International Journal of Mechanical Sciences, 10(10):803–805, 1968.

[47] KIRSCH, U.. **Optimal topologies of truss structures**. Comput Methods Appl Mech Eng, 72(1):15–28, 1989.

[48] KIRSCH, U.. **On singular topologies in optimum structural design**. Struct Optim, 2:133–142, 1990.

[49] CHENG, G. D.; GUO, X.. **ε-relaxed approach in structural topology optimization**. Struct Optim, 13:258–266, 1997.

[50] BRUGGI, M.. **On an alternative approach to stress constraints relaxation in topology optimization**. Struct Multidisc Optim, 36(2):125–141, 2008.

[51] DUYSINX, P.; SIGMUND, O.. **New developments in handling optimal stress constraints in optimal material distributions**. In: PROCEEDINGS OF THE 7TH AIAA/USAF/NASA/ISSMO SYMPOSIUM ON MULTIDISCIPLINARY ANALYSIS AND OPTIMIZATION, p. 1501–1509. 1998.

[52] HOLMBERG, E.; TORSTENFELT, B. ; KLARBRING, A.. **Stress constrained topology optimization**. Struct Multidisc Optim, 48:33–47, 2013.

[53] KIYONO, C. Y.; VATANABE, S. L.; SILVA, E. C. N. ; REDDY, J. N.. **A new multi-p-norm formulation approach for stress-based topology optimization design**. Composite Structures, 156:10–19, 2016.

[54] LE, C.; NORATO, J.; BRUNS, T.; HA, C. ; TORTORELLI, D.. **Stress-based topology optimization for continua**. Struct Multidisc Optim, 41:605–620, 2010.

[55] YANG, R. J.; CHEN, C. J.. **Stress-based topology optimization**. Struct Optim, 12:98–105, 1996.

[56] GOO, S.; WANG, S.; HYUN, J. ; JUNG, J.. **Topology optimization of thin plate structures with bending stress constraints**. Computers & Structures, 175:134–143, 2016.

[57] LIU, H.; YANG, D.; HAO, P. ; ZHU, Z.. **Isogeometric analysis based topology optimization design with global stress constraint**. Comput Methods Appl Mech Eng, 342:625–652, 2018.

[58] ZHANG, W.; JIANG, S.; LIU, C.; LI, D.; KANG, P.; YOUN, S.-K. ; GUO, X.. **Stress-related topology optimization of shell structures using IGA/TSA-based Moving Morphable Void (MMV) approach**. Comput Methods Appl Mech Eng, 366, 2020.

[59] NGUYEN, S. H.; NGUYEN, T. N. ; NGUYEN-THOI, T.. **A finite element level-set method for stress-based topology optimization of plate structures**. Computers & Mathematics with Applications, 115:26–40, 2022.

[60] GUO, X.; ZHANG, W. S.; WANG, M. Y. ; WEI, P.. **Stress-related topology optimization via level set approach**. Comput Methods Appl Mech Eng, 200(47–48):3439–3452, 2011.

[61] PARÍS, J.; NAVARRINA, F.; COLOMINAS, I. ; CASTELEIRO, M.. **Block aggregation of stress constraints in topology optimization of structures**. Advances in Engineering Software, 41(3):433–441, 2010.

[62] DA SILVA, G. A.; AAGE, N.; BECK, A. T. ; SIGMUND, O.. **Local versus global stress constraint strategies in topology optimization: A comparative study**. Int J Numer Methods Eng, 122(21):6003–6036, 2021.

[63] DUYSINX, P.; BENDSØE, M. P.. **Topology optimization of continuum structures with local stress constraints**. Int J Numer Methods Eng, 43(8):1453–1478, 1998.

[64] BRUGGI, M.; DUYSINX, P.. **Topology optimization for minimum weight with compliance and stress constraints**. Struct Multidisc Optim, 46(3):369–384, 2012.

[65] PEREIRA, J.; FANCELLO, E. ; BARCELLOS, C.. **Topology optimization of continuum structures with material failure constraints**. Struct Multidisc Optim, 26:50–66, 2004.

[66] JAMES, K. A.; LEE, E. ; MARTINS, J. R. R. A.. **Stress-based topology optimization using an isoparametric level set method**. Finite Elem Anal Des, 58:20–30, 2012.

[67] EMMENDOERFER JR, H.; FANCELLO, E. A.. **A level set approach for topology optimization with local stress constraints**. Int J Numer Methods Eng, 99(2):129–156, 2014.

[68] EMMENDOERFER JR, H.; FANCELLO, E. A.. **Topology optimization with local stress constraint based on level set evolution via reaction-diffusion**. Comput Methods Appl Mech Eng, 305:62–88, 2016.

[69] DA SILVA, G. A.; BECK, A. T. ; SIGMUND, O.. **Stress-constrained topology optimization considering uniform manufacturing uncertainties**. Comput Methods Appl Mech Eng, 344:512–537, 2019.

[70] BERTSEKAS, D. P.. **Nonlinear programming**. Athena Scientific, Nashua, 2 edition, 1999.

[71] NOCEDAL, J.; WRIGHT, S. J.. **Numerical optimization**. Springer, Berlin, 2 edition, 2006.

[72] SENHORA, F. V.; GIRALDO-LONDOÑO, O.; MENEZES, I. F. M. ; PAULINO, G. H.. **Topology optimization with local stress constraints: a stress aggregation-free approach**. Struct Multidisc Optim, 62:1639–1668, 2020.

[73] LONG, C. S.; LOVEDAY, P. W. ; GROENWOLD, A. A.. **Effects of finite element formulation on optimal plate and shell structural topologies**. Finite Elem Anal Des, 45:817–825, 2009.

[74] BOURDIN, B.. **Filters in topology optimization**. Int J Numer Methods Eng, 50(9):2143–2158, 2001.

[75] BORRVALL, T.; PETERSSON, J.. **Topology optimization using regularized intermediate density control**. Comput Methods Appl Mech Eng, 190(37–38):4911–4928, 2001.

[76] WANG, F.; LAZAROV, B. S. ; SIGMUND, O.. **On projection methods, convergence and robust formulations in topology optimization**. Struct Multidisc Optim, 43:767–784, 2011.

[77] CHENG, G.; JIANG, Z.. **Study on topology optimization with stress constraints**. Eng Optim, 20(2):129–148, 1992.

[78] GIRALDO-LONDOÑO, O.; PAULINO, G. H.. **A unified approach for topology optimization with local stress constraints considering various failure criteria: von Mises, Drucker–Prager, Tresca, Mohr–Coulomb, Bresler–Pister, and Willam–Warnke**. Proc. R. Soc. A, 476:20190861, 2020.

[79] SVANBERG, K.. **The method of moving asymptotes–a new method for structural optimization**. Int J Numer Methods Eng, 24(2):359–373, 1987.

[80] BENDSØE, M. P.; SIGMUND, O.. **Topology Optimization - Theory, Methods, and Applications**. Springer Berlin, Heidelberg, 2003.

[81] WU, S.; CEN, S.; MA, R. ; LI, C. F.. **Shape-free arbitrary polygonal hybrid stress/displacement-function flat shell element for linear and geometrically nonlinear analyses**. International Journal for Numerical Methods in Engineering, 122:4172–4218, 2021.

[82] DVORKIN, E. N.; BATHE, K.-J.. **A continuum mechanics based four-node shell element for general nonlinear analysis**. Engineering Computations, 1(1):77–88, 1984.

[83] BATHE, K.-J.; DVORKIN, E. N.. **A formulation of general shell elements — The use of mixed interpolation of tensorial components**. International Journal for Numerical Methods in Engineering, 22(3):697–722, 1986.

[84] CHOI, H.-G.; LEE, P.-S.. **The simplified MITC4+ shell element and its performance in linear and nonlinear analysis**. Computers & Structures, 290:107177, 2024.

[85] BATHE, K.-J.; LEE, P.-S. ; HILLER, J. F.. **Towards improving the MITC9 shell element**. Computers & Structures, 81(8):477–489, 2003.

[86] BUCALEM, M. L.; BATHE, K.-J.. **Higher-order MITC general shell elements**. International Journal for Numerical Methods in Engineering, 36(21):3729–3749, 1993.

[87] DA VEIGA, L. B.; CHAPELLE, D. ; SUÁREZ, I. P.. **Towards improving the MITC6 triangular shell element**. Computers & Structures, 85(21–22):1589–1610, 2007.

[88] KIM, D. N.; BATHE, K.-J.. **A triangular six-node shell element**. Computers & Structures, 87(23–24):1451–1460, 2009.

[89] KO, Y.; LEE, P.-S. ; BATHE, K.-J.. **The MITC4+ shell element and its performance**. Computers & Structures, 169:57–68, 2016.

[90] KO, Y.; LEE, P.-S. ; BATHE, K.-J.. **A new MITC4+ shell element**. Computers & Structures, 182:404–418, 2017.

[91] CHOI, C. K.; PAIK, J. G.. **An efficient four node degenerated shell element based on the assumed covariant strain**. Structural Engineering and Mechanics, 2(1):17–34, 1994.

[92] KULIKOV, G. M.; PLOTNIKOVA, S. V.. **A family of ANS four-node exact geometry shell elements in general convected curvilinear coordinates**. International Journal for Numerical Methods in Engineering, 83(10):1376–1406, 2010.

[93] KO, Y.; LEE, P.-S. ; BATHE, K.-J.. **A new 4-node MITC element for analysis of two-dimensional solids and its formulation in a shell element**. Computers & Structures, 192:34–49, 2017.

[94] CUI, X.; PENG, G.; RAN, Q.; ZHANG, H. ; LI, S.. **Derivation and implementation of one-point quadrature quadrilateral shell element with MITC4+ method (MITC4+R)**. Computers & Structures, 291:107207, 2024.

[95] CHOI, H.-G.; LEE, P.-S.. **Towards improving the 2D-MITC4 element for analysis of plane stress and strain problems**. Computers & Structures, 275:106933, 2023.

[96] KO, Y.; BATHE, K. ; ZHANG, X.. **Continuum mechanics-based shell elements with six degrees of freedom at each node the MITC4/D and MITC4+/D elements**. Computers & Structures, 308:107622, 2025.

[97] HO-NGUYEN-TAN, T.; KIM, H.-G.. **A new strategy for finite-element analysis of shell structures using trimmed quadrilateral shell meshes: a paving and cutting algorithm and a pentagonal**

shell element. International Journal for Numerical Methods in Engineering, 114:1–27, 2018.

[98] HO-NGUYEN-TAN, T.; KIM, H.-G.. **Polygonal shell elements with assumed transverse shear and membrane strains**. Comput Methods Appl Mech Eng, 349:595–627, 2019.

[99] YANG, Q.; SHEN, X.; ZHAO, J. ; CHENG, Y.. **An enriched virtual element method for 2D-3C generalized membrane shell model on surface**. International Journal for Numerical Methods in Engineering, 125(8), 2024.

[100] CHAPELLE, D.; BATHE, K.-J.. **The Finite Element Analysis of Shells - Fundamentals**. Springer, Berlin; Heidelberg, second edition, 2011.

[101] OÑATE, E.. **Structural Analysis with the Finite Element Method. Linear Statics. Volume 2: Beams, Plates and Shells**. Lecture Notes on Numerical Methods in Engineering and Sciences (LNNMES). Springer, Dordrecht, 2013.

[102] BATHE, K.-J.. **Finite Element Procedures**. Prentice Hall, second edition, 1996.

[103] BUCALEM, M. L.; BATHE, K.-J.. **Finite element analysis of shell structures**. Arch. Comput. Methods Eng., 4:3–61, 1997.

[104] ASHWELL, D. G.; SABIR, A. B.. **A new cylindrical shell finite element based on simple independent strain functions**. Int J of Mech Sci, 14, 1972.

[105] BELYTSCHKO, T.; STOLARSKI, H.; LIU, W. K.; CARPENTER, N. ; ONG, J. S. J.. **Stress projection for membrane and shear locking in shell finite elements**. Computer Methods in Applied Mechanics and Engineering, 51:221–258, 1985.

[106] SIMO, J. C.; FOX, D. D. ; RIFAI, M. S.. **On a stress resultant geometrically exact shell model. Part II: The linear theory; Computational aspects**. Computer Methods in Applied Mechanics and Engineering, 73:53–92, 1989.

[107] KRYSL, P.; CHEN, J. S.. **Benchmarking computational shell models**. Archives of Computational Methods in Engineering, 30:301–315, 2023.

[108] CHANDRASEKHAR, K. N. V.; BHIKSHMA, V. ; MOHI, S. A.. **On the six node hexagon elements for continuum topology optimization of plates carrying in plane loading and shell structures carrying out of plane loading**. J Appl Comput Mech, 6:617–639, 2020.

[109] HO-NGUYEN-TAN, T.; KIM, H.-G.. **An efficient method for shape and topology optimization of shell structures**. Struct and Mult Opt, 65, 2022.

[110] HO-NGUYEN-TAN, T.; KIM, H.-G.. **Stress-constrained concurrent two-scale topology optimization of functionally graded cellular structures using level set-based trimmed quadrilateral meshes**. Structural and Multidisciplinary Optimization, 66:123, 2023.

[111] CHI, H.; PEREIRA, A.; MENEZES, I. F. M. ; PAULINO, G. H.. **Virtual element method (VEM)-based topology optimization: an integrated framework**. Struct Multidisc Optim, 62:1089–1114, 2020.

[112] WEN, Z.; PAN, Q.; ZHAI, X.; KANG, H. ; CHEN, F.. **Adaptive isogeometric topology optimization of shell structures based on PHT-splines**. Computers & Structures, 305, 2024.

[113] BELBLIDIA, F.; BULMAN, S.. **A hybrid topology optimization algorithm for static and vibrating shell structures**. International Journal for Numerical Methods in Engineering, 54:835–852, 2002.

[114] CAI, S.; ZHANG, H. ; ZHANG, W.. **An integrated design approach for simultaneous shape and topology optimization of shell structures**. Computer Methods in Applied Mechanics and Engineering, 415, 2023.

[115] ZHANG, X.; XIAO, M.; GAO, L. ; GAO, J.. **A T-splines-oriented isogeometric topology optimization for plate and shell structures with arbitrary geometries using Bézier extraction**. Computer Methods in Applied Mechanics and Engineering, 425, 2024.

[116] PAN, Q.; ZHAI, X.; KANG, H.; DU, X. ; CHEN, F.. **Isogeometric topology optimization of multi-patch shell structures**. Computer-Aided Design, 174, 2024.

[117] PAN, Q.; ZHAI, X. ; CHEN, F.. **Density-based isogeometric topology optimization of shell structures**. Computer-Aided Design, 176, 2024.

[118] HUO, W.; LIU, C.; DU, Z.; JIANG, X.; LIU, Z. ; GUO, X.. **Topology optimization on complex surfaces based on the moving morphable component method and computational conformal mapping**. Journal of Applied Mechanics (ASME), 89(5), 2022.

[119] PAULINO, G. H.. **Glaucio H. Paulino's website, Software**. Glaucio H. Paulino's website, Princeton University, http://paulino.princeton.edu/software.html. Accessed 20 May 2023, 2023.

A
PolyTop version for plates

In this appendix, we present the main code modifications on the original PolyTop code provided by Talischi et al. [6] for compliance minimization. As the MATLAB® software has been updated several times since the publication of the original PolyTop article, an updated version of this code has been made available on the website of one of the authors [119], under the name of PolyTop v1.1, which will be the software considered hereinafter. We note that, since the following modifications will constantly change the line numbers, these numbers will be always updated accordingly to the previous changes already made. Therefore, the reader must strictly follow the sequence proposed below.

A.1
PolyScript.m

Replace line 19 by the following lines:

```
'h',0.1,...                 % Plate Thickness
'kappa',5/6,...               % Shear correction factor (rectang
   . section)
'Reg',0 ...                   % Tag for regular meshes
```

Delete lines 44 and 47, and replace line 43 by:

```
penal = 3;
```

A.2
PolyTop.m

Replace line 63 by:

```
fem.ElemNDof = 3*cellfun(@length,fem.Element); % # of DOFs per
   element
```

Replace line 71 by:

```
  eDof = reshape([3*fem.Element{el}-2;3*fem.Element{el}-1;3*fem
     .Element{el}],NDof,1);
```

Replace lines 80 to 82 by the following lines:

```
  fem.F = zeros(3*fem.NNode,1);  %external load vector
  fem.F(3*fem.Load(1:NLoad,1)-2) = fem.Load(1:NLoad,2);  %w-crdnt
  fem.F(3*fem.Load(1:NLoad,1)-1)   = fem.Load(1:NLoad,3);  %
     thetax-crdnt
  fem.F(3*fem.Load(1:NLoad,1))   = fem.Load(1:NLoad,4);  %thetay-
     crdnt
```

Replace lines 85 and 86 by:

```
  FixedDofs = [fem.Supp(1:NSupp,2).*(3*fem.Supp(1:NSupp,1)-2);
               fem.Supp(1:NSupp,3).*(3*fem.Supp(1:NSupp,1)-1);
               fem.Supp(1:NSupp,4).*(3*fem.Supp(1:NSupp,1))];
```

Replace lines 89 and 94 respectively by:

```
  AllDofs   = 1:3*fem.NNode;
U = zeros(3*fem.NNode,1);
```

Replace line 99 by:

```
G = fem.E0/(2*(1+fem.Nu0)); Ds = [G 0;0 G];
nn=length(eNode); Ke=zeros(3*nn,3*nn);
```

Insert the following line after line 104:

```
  N = fem.ShapeFnc{nn}.N(:,:,q);
```

Replace lines 107 to 112 by:

```
  Bb = zeros(3,3*nn);
  Bb(1,2:3:3*nn) = dNdx(:,1)';
  Bb(2,3:3:3*nn) = dNdx(:,2)';
  Bb(3,2:3:3*nn) = dNdx(:,2)';
  Bb(3,3:3:3*nn) = dNdx(:,1)';
  [BbT,BsT] = TimoAssumed(fem,nn,N,dNdx,fem.Node(eNode,:));
  Ke = Ke + fem.h^3/12*(Bb+BbT)'*D*(Bb+BbT)*W(q)*det(J0) + ...
       fem.kappa*fem.h*BsT'*Ds*BsT*W(q)*det(J0);
```

After line 116, insert the following lines to add the locking-free technique by Nguyen-Xuan [27]:

```
%------------------------------------------ TIMOSHENKO LOCKING-
   FREE MATRICES
function [BbT,BsT] = TimoAssumed(fem,nn,N,dNdx,xy)
jkm = [2:nn 1; 3:nn 1 2; nn 1:nn-1]';
```

```
c = xy(jkm(:,2),1)-xy(jkm(:,1),1); b = xy(jkm(:,1),2)-xy(jkm(:,2)
    ,2);
l = sqrt(c.^2 + b.^2);
chi = (fem.h./l).^2./(2*(fem.h./l).^2+fem.kappa*(1-fem.Nu0));
Ib = zeros(nn); Is = Ib; G = zeros(nn,3*nn);
Hb = zeros(3,nn); Hs = zeros(2,nn);
for i = 1:nn
    j = jkm(i,1); k = jkm(i,2); m = jkm(i,3);
    Ib(i,i) = 1-2*chi(i);
    Is(i,i) = chi(i);
    G(i,3*j-2:3*j) = [-2 c(i) -b(i)];
    G(i,3*k-2:3*k) = [2 c(i) -b(i)];
    Hb(:,i) = 3/l(i)^2*[c(i)*(dNdx(j,1)*N(k)+dNdx(k,1)*N(j));
                        -b(i)*(dNdx(j,2)*N(k)+dNdx(k,2)*N(j));
                        (c(i)*dNdx(k,2)-b(i)*dNdx(k,1))*N(j)+(c(i
                            )*dNdx(j,2)-b(i)*dNdx(j,1))*N(k)];
    Hs(:,i) = [b(m)/(c(i)*b(m)-c(m)*b(i))*N(j) - b(j)/(c(j)*b(i)-
        c(i)*b(j))*N(k);
               c(m)/(c(i)*b(m)-c(m)*b(i))*N(j) - c(j)/(c(j)*b(i)-
                   c(i)*b(j))*N(k)];
end
BbT = Hb*Ib*G; BsT = Hs*Is*G;
```

A.3
New domain examples

The domains for the numerical examples in Section 3.2.4 are defined herein.

A.3.1
`SquarePlateDomain.m`

Copy the original `MbbDomain.m` content, paste into the new `SquarePlateDomain.m` file and adapt according to the following instructions.

Replace lines 6 and 7 by:

```
function [x] = SquarePlateDomain(Demand,Arg)
  BdBox = [0 0.5 0 0.5];
```

For a simply supported plate, replace lines 20 to 28 by:

```
LeftEdgeNodes = find(abs(Node(:,1)-BdBox(1))<eps);   %x=0 (y-
    parallel)
RightEdgeNodes = find(abs(Node(:,1)-BdBox(2))<eps);  %x=Lx (y-
    parallel)
BottomEdgeNodes = find(abs(Node(:,2)-BdBox(3))<eps); %y=0 (x-
    parallel)
UpperEdgeNodes = find(abs(Node(:,2)-BdBox(4))<eps);  %y=Ly (x-
    parallel)
FixedNodes = [LeftEdgeNodes;BottomEdgeNodes;RightEdgeNodes;
    UpperEdgeNodes];
Supp = zeros(length(FixedNodes),4); Supp(:,1)=FixedNodes;
div = length([LeftEdgeNodes;BottomEdgeNodes]);
Supp(1:div,2)=1;                              % w = 0 at left &
    bottom edges
Supp(1:length(LeftEdgeNodes),4)=1;            % at left edge,
    theta_y = 0
Supp(length(LeftEdgeNodes)+1:div,3)=1;        % at bottom edge,
    theta_x = 0
Supp(div+1:div+length(RightEdgeNodes),3)=1; % in y-parallel sym
    ., theta_x = 0
Supp(div+length(RightEdgeNodes)+1:end,4)=1; % in x-parallel sym
    ., theta_y = 0
UpperRightNode = find(abs(Node(:,1)-BdBox(2))<eps & abs(Node
    (:,2)-BdBox(4))<eps);
Load = [UpperRightNode,-1/4,0,0];
```

After implementing for simply supported plates, the reader may adapt the domain for a clamped plate by replacing lines 26 to 31 by:

```
div1 = length([LeftEdgeNodes;BottomEdgeNodes]);
div2 = length([LeftEdgeNodes;BottomEdgeNodes;RightEdgeNodes]);
Supp(1:div1,2:4)=1;    % all dof fixed at left & bottom edges
Supp(div1+1:div2,3)=1; % in y-parallel sym., theta_x = 0
Supp(div2+1:end,4)=1;  % in x-parallel sym., theta_y = 0
```

A.3.2 `HookPlateDomain.m`

Similarly, copy the original `HookDomain.m` content, paste into `HookPlateDomain.m` file and then replace lines 33 to 40 by:

```
UpperCircleNodes = find(abs(sqrt(Node(:,1).^2+(Node(:,2)...
                                -80.6226).^2)-10)<eps);
Supp = ones(size(UpperCircleNodes,1),4);
Supp(:,1) = UpperCircleNodes;
EndHookNode = find(abs(Node(:,1)+27.0406)<eps & abs(Node(:,2)
   -8.0406)<eps);
while isempty(EndHookNode)
    eps=eps*1.1;
    EndHookNode = find(abs(Node(:,1)+27.0406)<5*eps & abs(Node
       (:,2)-8.0406)<5*eps);
end
EndHookNode = EndHookNode(find(Node(EndHookNode,2)==max(Node(
   EndHookNode,2))));
Load = [EndHookNode,-1,0,0];
```