VIRTUAL ELEMENTS AND BARYCENTRIC FINITE ELEMENTS IN COMPUTATIONAL MECHANICS

A Dissertation
Presented to
The Academic Faculty

By

Heng Chi

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Civil and Environmental Engineering

Georgia Institute of Technology

December 2018

VIRTUAL ELEMENTS AND BARYCENTRIC FINITE ELEMENTS IN COMPUTATIONAL MECHANICS

Approved by:

Dr. Glaucio H. Paulino, Advisor
School of Civil and Environmental Engineering
Georgia Institute of Technology

Dr. Arash Yavari
School of Civil and Environmental Engineering
Georgia Institute of Technology

Dr. Phanish Suryanarayana
School of Civil and Environmental Engineering
Georgia Institute of Technology

Dr. David L. McDowell
George W. Woodruff School of Mechanical Engineering
Georgia Institute of Technology

Dr. Lourenço Beirão da Veiga
Department of Mathematics and Applications
University of Milan-Bicocca

Dr. Cameron Talischi
Mckinsey & Company

Dr. Kyoungsoo Park
Department of Civil and Environmental Engineering
Yonsei University

Date Approved: Sept. 21, 2018

To my mother, Jing Zhao, and father, Renyong Chi.

ACKNOWLEDGEMENTS

First and foremost, I would like to sincerely thank my advisor, Professor Glaucio Paulino, for his continuous support and encouragement. His enthusiasm for research and innovation has always been a source of inspiration for me. I am also grateful for the numerous professional and academic opportunities he has provided me over the past six years. Without his guidance, I cannot imagine myself to be who I am today. I am also deeply grateful to Berta, Professor Paulino's wife, who offers me her warm hospitality, support and encouragement throughout my graduate studies.

This dissertation would not have been possible without the many wonderful people with whom I had the pleasure to collaborate. I am grateful to Professor Oscar Lopez-Pamies for his guidance and support during my studies at the University of Illinois at Urbana-Champaign, and for expanding my knowledge in the field of homogenization. I am thankful to Professor Lourenço Beirão da Veiga for many fruitful conversations on the numerical methods. Lourenço has a great sense of humor, and every discussion with him was enjoyable. I am indebted to Cameron Talischi for the many stimulating discussions and his mentorship during the early stage of my graduate study. Sincere thanks are due to Ivan Menezes for his constant encouragement and his hospitality and generosity during my visit to Rio de Janeiro. I would like to thank Anderson Peirera for the fruitful discussions and taking good care of me when I visited Rio de Janeiro. I would also like to thank Adeildo Ramos Jr. for the support and inspiring conversations. I extend my sincere thanks to the entire Ramos family for making my visit to Maceió so pleasant and fun. I gratefully thank Drs. David McDowell, Arash Yavari, Phanish Suryanarayana, Lourenço Beirão da Veiga, Cameron Talishi and Kyoungsoo Park for taking part in my Ph.D. thesis committee and providing me with invaluable suggestions.

I gratefully acknowledge the financial support of the US National Science Foundation through projects #1624232 (formerly #1437535) and #1663244. I am grateful for

the endowment provided by the Raymond Allen Jones Chair at the Georgia Institute of Technology. In addition, I am grateful to the financial support from the Corporate Technology of Siemens Corporation through the Deep Learning Enhanced Topology Optimization (DLETO) project.

I would like to thank my friends and colleagues, Xiaojia Shelly Zhang, Ke Liu, Daniel Spring, Evgueni Filipov, Cameron Talischi, Sofia Leon, Tomas Zegard, Tuo Zhao, Junho Chun, Arun Gain, Pradeep Phanisri, Kyoungsoo Park, Larissa Novelino, Fernando Senhora, Yang Jiang, Weichen Li, Emily Sanders, Erol Unal, Oliver Giraldo, Devyani Choudhary, Javier Vila Moran, Wei Peng, Dan Li, Xinjun Dong, Yang Zhang, Tao Zhu, Yuyu Zhang, Xi Liu, Chen Zhang, Hugo Bastos, and Guilherme Barros, who have provided me with constant support and brought tremendous joy to my graduate life. My special thanks are to Xiaojia Shelly Zhang, for her continuous kindness, support, and encouragement over the years; Daniel Spring, for his mentorship and for the many stimulating discussions we had; Evgueni Filipov, for his help and invaluable suggestions; and Ke Liu, for his support and friendship.

Lastly and most importantly, my deepest gratitude is to my mother, Jing Zhao, and my father, Renyong Chi, for their unconditional love, encouragement, and support. No words can describe how grateful I am for everything they have done for me. I am so proud to be their son, and it is with great pleasure that I dedicate this dissertation to them.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

α	Material parameter in Lopez-Pamies material
α_E	Stabilization parameter associated with E
β_j^F	The weight associated with $\mathbf{X}_j^F$ in defining $\mathbf{X}_s^F$
$\mathbf{\Pi}_E^0$	A tensor-valued L^2 projection which projects second order tensors onto its average over E
$\mathbf{\Pi}_k^0$	A vector-valued L^2 projection which projects vectors from $[\mathscr{V}_k(E)]^d$ onto $[\mathscr{P}_k(E)]^d$. When $k=1$ and $k=2$, $\mathbf{\Pi}_k^0 = \mathbf{\Pi}_E^\nabla$
$\mathbf{\Pi}_E^\nabla$	A vector-valued projection which projects vectors from $[\mathscr{V}(E)]^d$ onto $[\mathscr{P}_1(E)]^d$
Δ	Laplacian operator with respect to the undeformed configuration
$\ell(\cdot)$	Continous loading term for linear elasticity
η_h	Design function on Ω_h
$\Gamma^\mathbf{t}$	Part of the domain boundary where traction $\mathbf{t}$ is prescribed
$\Gamma_h^\mathbf{t}$	The boundary of the mesh where traction is applied
$\Gamma^\mathbf{X}$	Part of the domain boundary where displacement $\mathbf{u}^0$ is prescribed
$\Gamma_h^\mathbf{X}$	The boundary of the mesh where displacement is applied
κ	Bulk modulus
$\mathbb{C}$	Elasticity modulus tensor
$\mathscr{A}_h$	Space of admissible designs
$\mathscr{F}_h(\cdot)$	Discrete density filter map between ρ_h and η_h
$\mathscr{F}_i$	The set of faces that connect to $\mathbf{X}_i$ in P
$\mathscr{G}_F$	A map utilized to denote the relation between the global numbering and local numbering on F of element P
$\mathscr{K}$	Space of kinematically admissible displacmenets

$\mathscr{K}_{h,k}$ Discrete global VEM displacement space in a kth order approximation. The k is omitted when $k = 1$

$\mathscr{M}_k(E)$ Local finite element space of order k on element E.

$\mathscr{P}_k(E)$ Polynomial space of order k on element E

$\mathscr{Q}$ Space consists of square integrable functions

$\mathscr{Q}^C_{h,k-1}$ Discrete global pressure space consisting of continuous functions in a kth order mixed approximation (Talyor-Hood elements).

$\mathscr{Q}^D_{h,k-1}$ Discrete global pressure space consisting of discontinuous functions in a kth order mixed approximation (Crouzeix-Raviart elements). Also denoted as $\mathscr{Q}_h$ when $k = 1$

$\mathscr{V}_k(E)$ Local VEM space of order k on element E. k is omitted when $k = 1$.

μ Shear modulus

∇ Gradient operator with respect to the undeformed configuration

$\nabla_{E,k}$ Corrected gradient operator with respect to the undeformed configuration for a kth order finite element E

ν Poisson's ratio

$\oint$ A given numerical integration over ∂E that exactly integrate any polynomials of order $2k - 1$.

Ω Undeformed configuration

Ω_h A discretization of the domain Ω

ω_i A patch of elements associated with the ith node in the mesh

$\overline{\mathbf{F}}$ Deviatoric part of deformation gradient defined by $\overline{\mathbf{F}} = (\det \mathbf{F})^{-\frac{1}{3}}\mathbf{F}$

$\overline{V}$ Allowable volume fraction

$\phi(I_1, I_2, J)$ Another form of $\Psi(\mathbf{F})$ that depends solely on the invariants I_1 I_2 and J for isotropic solids

$\Pi(\mathbf{v})$ Potential energy in the displacement-based formulation

Π^0_E A vector-valued L^2 projection which projects a vector onto its average over E

Π^0_k A scalar-valued L^2 projection which projects vectors from $\mathscr{V}_k(E)$ onto $\mathscr{P}_k(E)$. When $k = 1$ and $k = 2$, $\Pi^0_k = \Pi^\nabla_E$

Π^∇_E A scalar-valued projection which projects vectors from $\mathscr{V}(E)$ onto $\mathscr{P}_1(E)$

$\Pi_h^{VEM}(\mathbf{v})$	VEM approximation of the potential energy in the displacement-based formulation
$\Psi(\mathbf{X},\mathbf{F})$	The remainder part of the stored-energy function
ρ_h	Material density function on Ω_h
$\sqint$	A given numerical integration over E that exactly integrate any polynomials of order $2k-2$.
$\mathbf{C}$	Right Cauchy-Green deformation tensor $\mathbf{C} = \mathbf{F}^T\mathbf{F}$
$\mathbf{F}$	Deformation gradient tensor
$\mathbf{f}$	Body force per unit undeformed volume
$\mathbf{F}_{\text{ext}}$	External force vector
$\mathbf{K}_\rho$	Global stiffness matrix incorporating spatially varying material densities
$\mathbf{M}_{\text{cov}}(E)$	The covariance matrix for element E
$\mathbf{n}^F$	The outward normal of face F in 3D
$\mathbf{n}_e$	The outward normal of edge e in 2D in the undeformed configuration
$\mathbf{P}$	First Piola-Kirchhoff stress tensor
$\mathbf{P}^{\mathscr{F}}$	Density filter matrix
$\mathbf{p}^{\omega_i}(\mathbf{X})$	A polynomial of order $k+1$ recovery on patch ω_i
$\mathbf{s}$	The vector of element stiffness interpreted from material densities
$\mathbf{t}$	Applied boundary traction per unit undeformed area
$\mathbf{U}$	Global displacement vector
$\mathbf{u}_h^{n-1}$	Equilibrium displacement field obtained in the previous Newton-Raphson step
$\mathbf{u}_h$	Equilibrium displacement field
$\mathbf{X}$	Position vector in the reference configuration
$\mathbf{X}_c^E$	The centroid of E
$\mathbf{X}^F$	A shifted position vector defined on face F as $\mathbf{X}^F \doteq \mathbf{X} - \mathbf{X}_c^F$.
$\mathbf{X}_c^F$	The centroid of F
$\mathbf{X}_j^F$	The jth vertex on face F of element P with local numbering
$\mathbf{X}_s^F$	The triangulation point on face F of element P

$\mathbf{X}^P$	A shifted position vector defined in P as $\mathbf{X}^P \doteq \mathbf{X} - \mathbf{X}_c^P$.
$\mathbf{X}_c^P$	The centroid of P
$\mathbf{X}_i$	The ith vertex of element E in the undeformed configuration
$\mathbf{y}$	The vector of material densities
$\mathbf{z}$	The vector of design variables
$\mathrm{adj}(\cdot)$	The adjugate operator
θ	A user-defined threshold for adaptive mesh refinement
$\varepsilon^s_{1,\mathbf{u}}$	The H^1-type skeletal error of the original displacement solution
$\varepsilon^s_{1,\widetilde{\mathbf{u}}}$	The H^1-type skeletal error of the recovered displacement solution
$\varepsilon^v_{0,\mathbf{u}}$	The L^2-type displacement error using only vertex values
$\varepsilon^v_{0,\widehat{p}}$	The L^2-type the pressure error using only vertex values
$\varepsilon^v_{1,\mathbf{u}}$	The H^1-type displacement error using only vertex values
$\varepsilon_{0,\mathbf{u}}$	The L^2-norm of the displacement error
$\varepsilon_{0,\widehat{p}}$	The L^2-norm of the pressure error
$\varepsilon_{1,\mathbf{u}}$	The H^1-seminorm of the displacement error
$\varphi_i^{(k)}$	The ith basis function of the local spaces $\mathscr{M}_k(E)$ or $\mathscr{V}_k(E)$. k is omitted when $k = 1$.
$\widehat{\Pi}\left(\mathbf{v},\widehat{p}\right)$	Potential energy in the mixed $\mathbf{F}$-formulation
$\widehat{\Pi}_h^{VEM}\left(\mathbf{v},\widehat{p}\right)$	VEM approximation of the potential energy in the mixed $\mathbf{F}$-formulation
$\widehat{U}^*\left(\mathbf{X},\widehat{q}\right)$	Legendre transformed function of $U(\mathbf{X},J)$ in J
$\widehat{W}(\mathbf{X},\mathbf{F},J)$	Function introduced such that $W(\mathbf{X},\mathbf{F}) = \widehat{W}(\mathbf{X},\mathbf{F},J)$ when $J = \det\mathbf{F}$
$\widehat{W}^*(\mathbf{X},\mathbf{F},\widehat{q})$	Function obtained by Legendre transformation of J in $\widehat{W}(\mathbf{X},\mathbf{F},J)$
$\widehat{X}_{ij}^{\ell}$	The ℓth internal (Gauss-Lobatto) point on the edge connecting vertices $\mathbf{X}_i$ and $\mathbf{X}_j$
$\widehat{X}_{ij}$	The middle node of the edge connecting vertices $\mathbf{X}_i$ and $\mathbf{X}_j$
$\widetilde{\mathscr{F}_i}$	The set of faces that connect to $\widetilde{\mathbf{X}}_i$ in $\widetilde{P}$

$\widetilde{\mathscr{V}}_k(E)$	Augmented local VEM space of order k on element E. k is omitted when $k=1$.
$\widetilde{\nabla}$	Gradient operator with respect to the deformed configuration
$\widetilde{\mathbf{n}}^e$	The outward normal of edge $\widetilde{e}$ in 2D
$\widetilde{\mathbf{n}}^F$	The outward normal of face $\widetilde{F}$ in 3D
$\widetilde{\mathbf{X}}_j^F$	The jth vertex on face $\widetilde{F}$ of polyhedron $\widetilde{P}$ with local numbering
$\widetilde{\mathbf{X}}_s^F$	The triangulation point on face $\widetilde{F}$ of element P
$\widetilde{\mathbf{X}}_i$	The ith vertex of element E in the deformed configuration
$\widetilde{\varepsilon}_{1,\mathbf{u}}^s$	The estimated H^1-type skeletal error of the original displacment solution
$\widetilde{E}$	A generic element (in 2D or 3D) in the deformed configuration
$\widetilde{e}$	A generic edge of element $\widetilde{E}$ in the deformed configuration
$\widetilde{F}$	A generic polygon or a generic face of a polyhedron in the deformed configuration
$\widetilde{P}$	A generic polyhedron in the deformed configuration
$\widetilde{T}_j^F$	The triangulated subfaces of $\widetilde{F}$
Ξ_i^e	The ith edge DOF of a given function in the patch ω
Ξ_i^F	The ith (internal) facial DOF of a given function in the patch ω
Ξ_i^I	The ith internal DOF of a given function in the patch ω
Ξ_i^v	The ith vertex DOF of a given function in the patch ω
$a(\cdot,\cdot)$	Continous bilinear form for linear elasticity
$a_h^E(\cdot,\cdot)$	Element-level discrete bilinear form for linear elasticity constructed using VEM
$a_h^\rho(\cdot,\cdot)$	Discrete bilinear form for linear elasticity constructed using VEM incorporating the spatially varying stiffness
$a_h(\cdot,\cdot)$	Discrete bilinear form for linear elasticity constructed using VEM
d	Dimension
E	A generic element (either in 2D or 3D) of the mesh Ω_h in the undeformed configuration
e	A generic edge of element E in the undeformed configuration

E_0	Young's modulus
F	A generic polygon or a generic face of a polyhedron in the undeformed configuration
$G_h\mathbf{u}_h$	A continuous displacement gradient reconstructed on the mesh skeleton
h_F	The diameter of polygon F
h_P	The diameter of polygon P
h_ω	A characteristic size of ω
I_1	First invariant of the right Cauchy-Green deformation tensor $\mathbf{C}$
I_2	Second invariant of the right Cauchy-Green deformation tensor $\mathbf{C}$
J	Determinant of deformation gradient $J = \det \mathbf{F}$
J_E	The area/volume average of $J = \det \mathbf{F}$ over E
M	Total number of elements in Ω_h
m_α^ω	The αth scaled basis function of polynomial space $\mathscr{P}_k(\omega)$
$m_S(\rho_h)$	Material interpolation function
$n\mathscr{M}_k$	The dimension of the local finite element space $\mathscr{M}_k(E)$.
$n\mathscr{P}_k$	The dimension of the polynomial space $\mathscr{P}_k(E)$.
N_ω^E	The total number of elements the patch ω contains
N_ω^e	The total number of edge DOFs of a given function in the patch ω
N_ω^I	The total number of (internal) facial DOFs of a given function in the patch ω
N_ω^I	The total number of internal DOFs of a given function in the patch ω
N_ω^v	The total number of vertex DOFs of a given function in the patch ω
N_D	The total number of design variables in the mesh Ω_h
N_E	Total number of DOFs in E
P	A generic polyhedron in the undeformed configuration
$S(i)$	The set of indices of design variables whose positions fall within a sphere of prescribed radius R centered at $\mathbf{X}_i$
$S_{h,E}(\cdot,\cdot)$	Bilinear form in the stabilization term associated with E
T_j^F	The triangulated subfaces of F

$U(\mathbf{X}, J)$ Volumetric part of the stored-energy function that depends only on $J = \det \mathbf{F}$

V The vector of element volumes

$W(\mathbf{X}, \mathbf{F})$ Stored-energy function

w_v^E The weight associated with vertex v to integration functions over E that is first order accurate

w_j^F The weight associated with $\mathbf{X}_j^F$ defined in a vertex-based quadrature rule to integration functions on F that is first order accurate

$w_i(\mathbf{X})$ The weight function associated with vertex i in the Mean Value Coordinates (MVC)

$\varepsilon_{\boldsymbol{\sigma}}$ The L^2 error of the stress solution

SUMMARY

Due to its unique and intriguing properties, polygonal and polyhedral discretization is an emerging field in computational mechanics. This thesis deals with developing finite element and virtual element formulations for computational mechanics problems on general polygonal and polyhedral discretizations.

The construction of finite element approximation on polygonal/polyhedral meshes relies on generalized barycentric coordinates, which are non-polynomial (e.g. rational) functions. Thus, the existing numerical integration schemes, typically designed to integrate polynomial functions, will lead to persistent consistency errors that do not vanish with mesh refinement. To overcome the limitation, this thesis presents a general gradient correction scheme, which restores the polynomial consistency by adding a minimal perturbation to the gradient of the displacement field, and applies it to formulate both lower- and higher-order polygonal finite elements for finite elasticity problems. With the gradient correction scheme, the optimal convergence is recovered in finite elasticity problems.

The Virtual Element Method (VEM) was recently proposed as an attractive framework to handle unstructured polygonal/polyhedral discretizations and beyond (e.g., arbitrary non-convex shapes). The VEM is inspired by the mimetic methods, which mimics fundamental properties of mathematical and physical systems (e.g., exact mathematical identities of tensor calculus). Unlike the Finite Element Method (FEM), there are no explicit shape functions in the VEM, which is a unique feature that leads to flexible definitions of the local VEM spaces. This thesis also develops novel VEM formulations for several classes of computational mechanics problems. First, to study soft materials, we present a general VEM framework for finite elasticity. The framework features a nonlinear stabilization scheme, which evolves with deformation; and a local mathematical displacement space, which can effectively handle any element shape, including concave elements or ones with non-planar faces. We verify the convergence and accuracy of the proposed virtual elements by means of examples using unique element shapes inspired by Escher (the Dutch artist famous for his so-called impossible constructions). Second, to fully realize the potential of VEM in mesh adaptation (i.e., refinement, coarsening and local re-meshing), we develop a gradient recovery scheme and *a posteriori* error estimator for VEM of arbitrary order for linear elasticity problems. The *a posteriori* error estimator is simple to implement yet has been shown to be effective through theoretical and numerical analyses. Finally, from a design viewpoint, we present an efficient topology optimization framework on general polyhedral discretizations by synergistically incorporating the VEM and its mathematical/numerical features in the underlining formulation. As a result, the tailored VEM space naturally leads to a continuous material density field interpolated from nodal design variables. This approach yields a mixed virtual element with an enhanced density field. We present examples that explore the aforementioned features of our VEM-based topology optimization framework and contrast our results with the traditional FEM-based approaches that dominate the technical literature.

CHAPTER 1

INTRODUCTION

Due to its unique and intriguing properties, polygonal/polyhedral discretization is an emerging area of research in the fields of engineering and mathematics. As more general discretizations, polygonal and polyhedral meshes have demonstrated several advantages over the standard triangular/tetrahedral and quadrilateral/hexahedral meshes numerical analysis. From a geometric point of view, polygonal and polyhedral elements offer more flexibility in discretization. For example, with the polyhedral elements, we can discretize any complex 3D geometry in an automated fashion using the idea illustrated in Fig. 1 (a). We can start with a structured mesh of regular hexahedral elements, and then trim the ones which intersect with the CAD surface to appropriately approximate the boundary [1]. In this approach, although the majority of elements in the mesh remain to be regular hexahedra, polyhedra are generated on the boundary of the mesh. Additionally, polygonal and polyhedral elements are well suited to modeling complex porous or particulate microstructures. For example, as illustrated in Fig. 1 (b), any inclusions, irrespectively of their shape, can be modeled as single polygons, and any mesh can be made periodic by locally inserting nodes at appropriate locations [2]. Moreover, polygonal and polyhedral elements are attractive in adaptive mesh refinement and coarsening. For instance, as shown in Fig. 1 (c), adaptivity using polygonal element can be used in simulating the cyclic competition model, in which three species competing in a common habitat for a finite set of resources, to greatly improved the computational efficiency without losing accuracy in capturing the interfaces [3]. From an analysis point of view, polygonal and polyhedral meshes exhibit improved performance in terms of inf-sup stability and accuracy, as well as in large deformation problems. For example, as shown in Fig. 1 (d) for fluid mechanics problems, polygonal meshes are shown in be unconditionally inf-sup stable without the need of any additional treatments with linear displacement and piece-wise constant pressure approximations, whereas the same approximations yield unphysical pressure distribution on quadrilateral meshes. Additionally, because of the enriched displacement gradient approximation, polygonal elements are shown to yield more accurate displacement solutions, as shown in Fig. 1 (e), and to be more tolerant to large localized deformations, as shown in Fig. 1 (f).

To formulate numerical methods on polygonal and polyhedral discretizations, conforming barycentric Finite Element Method (FEM) and the recent Virtual Element Method (VEM) are the two popular approaches used in the field of computational mechanics. Other approaches include the Voronoi Cell Finite Element Method (VCFEM) by Ghosh and coworkers [6]–[9], Discontinuous Galerkin (DG) method [10], [11] and Mimetic Finite Difference (MFD) [12]. This thesis aims to contribute to the developments of these two methods in finite elasticity and topology optimization problems. Moreover, to fully realize the geometrical advantages of polygonal and polyhedral discretizations, this thesis also contributes to the development of a simple and effective error estimator for adaptivity.

1.1 Research motivation

1.1.1 Polygonal and polyhedral finite element method

Despite their relatively short history of development, polygonal and polyhedral finite elements have been successfully applied to several fields of computational mechanics and have revealed several advantages over classical finite elements, i.e., triangular/tetrahedral and quadrilateral/brick elements. For instance, polygonal and polyhedral finite elements have been shown to better capture crack propagation and branching in computational fracture simulations [13]–[16], to produce numerically stable results in topology optimization [17]–[19] and fluid mechanics [4], and to better model contact [20] and arbitrary internal interface [21] in large deformation elasto-plasticity problems. Furthermore, recent studies have demonstrated that polygonal finite elements also possess great potential in the study of two-dimensional finite elasticity problems [2], [22]. More specifically, from a

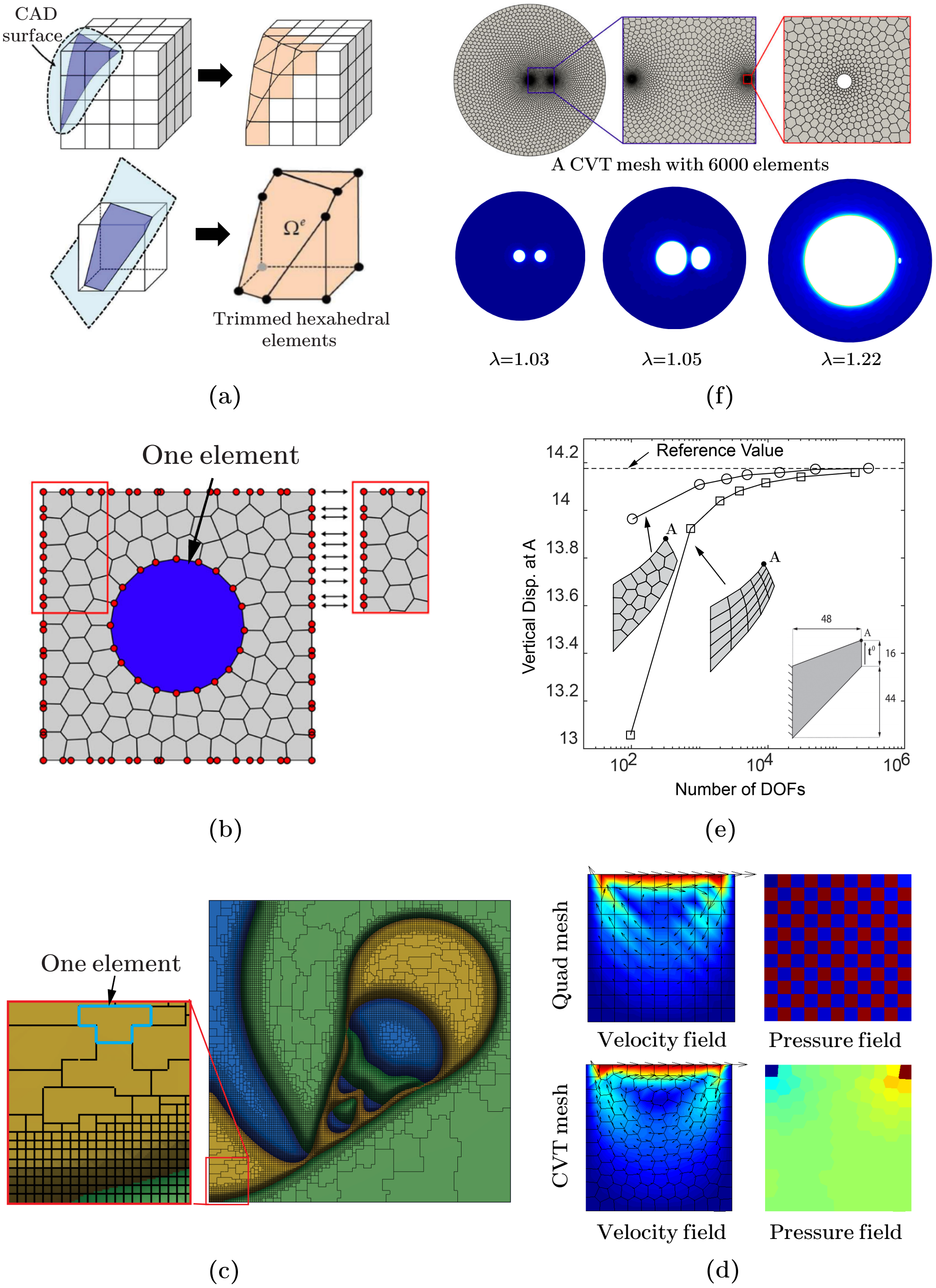


Figure: (Caption on the following page.)

Figure 1.1: Several examples show the advantages of polygonal and polyhedral discretizations from both geometry and analysis points of view. (a) An illustration of using trimmed hexahedral elements for the discretization of complex 3D geometry [1]. In the discretization, once an element intersects with the CAD surface, it is trimmed by the CAD surface to form a polyhedron. (b) A conceptual illustration of considering inclusions of any shape as one single polygon and the local insertion of nodes on the opposing edges of the unit cell to form a periodic mesh [2]. (c) An examples of the pattern formation of the cyclic competition model obtained using adaptive refinement and coarsening at a given time step [3]. (d) Comparisons of the velocity and pressure distributions in a lid-driven cavity problem obtained on a quadrilateral mesh and a polygonal one [4]. The pressure distribution obtained using a quadrilateral mesh is unphysical and forms the checkerboard pattern, while the one obtained using a polygonal mesh is physical. (e) The comparison of the accuracy of displacement solutions of the Cook's problem [5] under refinements of polygonal and quadrilateral meshes. It is shown that the solutions obtained by the polygonal meshes converge faster to the reference value than the ones obtained by the quadrilateral meshes [2]. (f) The discretization of a circular disk with two embedded defects using polygonal elements [2]. Several snapshots of the elastic growth of defects under different levels of hydrostatic stretch are plotted, demonstrating the capability of polygonal elements in capturing large and localized deformations [2].

geometrical point of view, polygonal finite elements are well suited to modeling complex microstructures, such as porous or particulate microstructures and microstructures involving different length scales, and to dealing with periodic boundary conditions (introducing hanging nodes). From an analysis point of view, two-field mixed polygonal finite elements are found to be numerically stable on Voronoi-type meshes, to produce more accurate results, and to be more tolerant to large localized deformations.

On the other hand, polygonal and polyhedral finite elements also suffer from several drawbacks. The construction of local finite element spaces on general polygon and polyhedral meshes relies on the generalized barycentric coordinates, which are typically non-polynomial (e.g., rational) functions. Thus, the existing quadratures schemes, typically designed for integration of polynomial functions, will lead to persistent consistency errors that do not vanish with mesh refinement [23], [24]. The persistence of consistency errors in turn renders the FEM suboptimally convergent or even non-convergent under mesh refinement.

Several attempts have been made in the literature to address this issue. For example, in the context of scalar diffusion problems, inspired by the virtual element method (VEM) [25]–[27], Talischi et al. [23] proposed a polynomial projection approach to ensure the polynomial consistency of the bilinear form, and thereby ensure the satisfaction of the patch test and optimal convergence for both linear and quadratic polygonal elements. A similar approach is also adopted by Manzini et al. [24] to solve Poisson problems on polyhedral meshes. However, those approaches require the existence of a bilinear form, and therefore extension to general nonlinear problem is non-trivial. From another perspective, borrowing the idea of pseudo-derivatives in the meshless literature [28], Bishop proposed an approach to correct the derivatives of the shape functions to enforce the linear consistency property on general polygonal and polyhedral meshes [13], [29]. Although being applicable for general nonlinear cases, extension to higher order cases(e.g. quadratic polygonal and polyhedral finite elements) is not readily implied. Thus, a general scheme that works for both higher-order polygonal and polyhedral finite elements in nonlinear problems is still needed.

To overcome the above limitations, Chapter 2 of this thesis adopts a general gradient correction scheme, which restores the polynomial consistency of the finite element approximation by adding a minimal perturbation to the gradient of the displacement field, and develops displacement-based and two-field mixed polygonal finite elements for finite elasticity problems. We demonstrate that the gradient correction scheme recovers the optimal convergence for solutions of both lower- and higher-order approximations. With the capability of accurately and efficiently handling higher-order mixed polygonal finite elements, Chapter 2 also discusses various choices of approximating the pressure fields for higher-order mixed polygonal finite elements and investigates their performance on inf-sup stability and accuracy.

1.1.2 Virtual element method

The performance of polygonal and polyhedral finite elements strongly depends on the quality of the basis functions. The use of finite elements with more general shapes, such as concave polyhedra, are limited by the availability of well-defined shape functions. Particularly in 3D, most of the existing shape functions for polyhedral finite elements are either restrictive in element geometries or computationally expensive to compute. For instance, although closed-form expressions are available, the Wachspress shape functions are restricted to strictly convex and simple polyhedrons (meaning the collection of faces that include each vertex consists of exactly three faces) [30], and the Mean Value coordinates are mainly applicable to polyhedrons with simplicial faces [31]. In contrast, while allowing for more general polyhedrons (e.g., concave ones), harmonic shape functions [29], [32], [33] and max-entropy shape functions [34]–[36] need to be computed numerically at the quadrature points element by element.

As an alternative approach, VEM is an emerging method first introduced in [37] as a generalization of FEM in the family of Galerkin methods. Unlike FEM, the basis functions of the local space in VEM are defined implicitly through a suite set of partial differential equations (PDEs), and this set of PDEs is never solved throughout the approximation. Instead, we apply integration by parts to compute suitable projections of the basis functions on to polynomials. [37], [38]. Those projections are then used in the VEM approximation to ensure its consistency together with a suitable stabilization term, which is needed in order to avoid the appearance of hourglass modes. As a result, only numerical integrations for polynomials (and not for more complex functions) are required in the VEM. These unique features allow VEM to handle any polygonal and polyhedral meshes (including the non-convex ones [39], [40]) and to construct various types of elements, including $H(\text{div})$ and $H(\text{curl})$ conforming elements [41]. The VEM has undergone substantial developments and has been successfully applied to a wide range of problems. In the field of computational mechanics, the VEM has been introduced for linear elasticity problems [26], [42], [43], small deformation nonlinear elastic and inelastic problems [44]–[47], plate bending problems [48]–[51] and contact problems [52].

Most work in the VEM literature considers either linear or nonlinear problems under small deformation (we also mention some more recent work on large deformation elasticity [53] and elasto-plasticity [54] problems). The goal of Chapter 3 is to put forward a general VEM framework for finite elasticity problems, which is capable of handling general polygonal and polyhedral elements, including concave ones. More specifically, Chapter 3 presents two VEM formulations including a two-field mixed VEM formulation, which involves an additional pressure field, and an equivalent displacement-based VEM formulation, which is numerically shown to be free of volumetric locking (as opposed to the standard displacement-based FEM). To achieve robust performance under large deformations, the VEM framework adopts an L^2 projection of the volumetric strain (i.e. determinant of the deformation gradient) computable through properly defined local VEM displacement spaces, and introduces a new stabilization scheme that consistently evolves through deformations.

Among various features of VEM, the flexibility in dealing with general polygonal and polyhedral meshes makes it appealing for adaptivity. For example, by introducing hanging nodes, adaptive mesh refinement can be made more efficient with polygonal and polyhedral elements because it only requires local modifications to the mesh [2]. On the other hand, the shape generality of polygonal and polyhedral elements (especially the concave ones) enables easier element agglomeration schemes for adaptive mesh coarsening [55], [56]. As a result, realizing the full potential of VEM in adaptivity urges the development of accurate *a posteriori* error estimators.

Comparing to FEM, *a posteriori* error estimations in the VEM framework is a more involved task because the basis functions of the local VEM space are unknown in the interior of elements. Nevertheless, there exist some error estimators in the literature for C^0 and C^1 VEMs [57]–[60], but all of them are of residual type. Comparing to the residual type error estimators, the recovery-based error estimators (although supported by a less extended theoretical background with respect to other methods) are often more favorable in practical applications because of their simple structure, easy implementation, and effectiveness in predicting errors. Thus, Chapter 4 of this thesis outlines a general recovery-based *a posteriori* error estimation framework for H^1 conforming VEM of arbitrary order on general polygonal/polyhedral meshes. The framework consists of a gradient recovery scheme and *a posteriori* error estimator based on the recovered displacement gradient. A skeletal error, which accurately mimics the behavior of the L^2 error of the displacement gradient by only sampling the displacement gradient on the mesh skeleton, is introduced.

1.1.3 Topology optimization on polyhedral discretization

Topology optimization is a powerful computational tool to design optimal structures under given loads and boundary conditions. Since the seminal work of Bendsøe and Kikuchi [61], the field of topology optimization has experienced tremendous growth and had a major impact in several areas of engineering and technology.

Among various topology optimization approaches, the density-based approach is commonly adopted on structured meshes (e.g., uniform grids) with a piece-wise constant density parametrization, where each element is assigned with a constant density, and a lower-order Lagrangian-type finite element approximation of the displacement field. However, this choice of approximation typically suffers from numerical instabilities such as checkerboard patterns and one node connections [62], [63], may lead to mesh-dependent solutions [64], and has limited ability to discretize complex design domains [17].

As a more general discretization, polygonal elements are shown to be effective in suppressing checkerboard patterns and reducing mesh dependency in the solutions of topology optimization [17]–[19], [64], [65]. In addition, the geometrical flexibility of polygonal elements also makes them more versatile in discretizing complex domains and attractive for mesh adaptations in topology optimization (see, e.g., [66], [67]). However, extending topology optimization to arbitrary 3D polyhedral meshes is a challenging task consisting of two major challenges: difficulty in solving the structural equations on general polyhedral meshes and the increased computational cost resulted from mixed approximations of displacement and design fields on polyhedral meshes. The latter challenge comes from the fact that, because a typical polyhedral (e.g., Voronoi) mesh contains significantly more vertices than elements, the choice of element-wise constant density and node-based displacement approximations becomes considerably less computationally economical on polyhedral meshes. In particular, when compared with uniform grids of a similar number of elements (thus similar numbers of densities and design variables and design resolution), we need to solve a much larger structural system in each optimization iteration if a polyhedral mesh is considered.

To overcome the above mentioned two challenges, Chapter 5 of this thesis makes full use of the features of VEM and propose a new VEM-based topology optimization framework on general polyhedral discretizations. More specifically, we adopt the VEM in both structural and optimization problems. Similarly to [65], we use VEM to solve the elasticity equation in the structural problem. The capability of VEM in handling arbitrary element geometry allows us to solve the structural problem more efficiently (i.e., no numerical integration is needed) and robustly (i.e., with respect to degenerated elements) on polyhedral meshes. Exploiting the flexibility of VEM in defining local spaces, this chapter also introduces an enriched continuous approximation of material densities using nodal VEM basis functions. As compared to the standard element-wise constant density approximation, the continuous approximation contains a greater number of degrees of freedom (DOFs) for any given polyhedral mesh and can thus improve the quality of structural topology parameterizations.

1.2 Thesis organization

The remainder of this thesis is organized as follows. Chapter 2 introduces lower- and higher-order polygonal finite element formulations for finite elasticity problems using both displacement-based and two-field mixed approximations. The convergence of the formulations are ensured by a general gradient correction scheme, whose basic idea is to correct the displacement gradient at the element level with a minimal perturbation so that the discrete divergence theorem is enforced against polynomial functions of suitable order. Chapter 2 also discusses various choices of approximating the pressure fields for higher-order mixed polygonal finite elements and investigates their performance on inf-sup stability and accuracy. Chapter 3 proposes a general VEM framework for finite elasticity which is capable of handling a more general class of element geometry in 2D and 3D. Two VEM formulations are presented: a two-field mixed and an equivalent displacement-based, which is free of volumetric locking. In addition to the standard ingredients in VEM, this framework also introduces a new L^2 projection of the volumetric strain, which is exactly computable through properly defined local displacement spaces, and a nonlinear stabilization technique, which evolves with the deformations. To fully realize its potential of VEM adaptivity, Chapter 4 focuses on the development of a simple and effective recovery-based *a posteriori* error estimator for VEM of arbitrary order of accuracy. To avoid the difficulty of not knowing the VEM basis functions in the interior of elements, we introduce a skeletal error measure, which accurately mimics the behavior of the L^2 error of the displacement gradient by only sampling its value on the mesh skeleton. Theoretical and numerical analyses are presented to demonstrate the accuracy of the *a posteriori* error estimator on a wide class of polygonal and polyhedral meshes. In Chapter 5, we present a VEM-based 3D topology optimization framework on general polyhedral meshes. The framework takes full advantage of the VEM properties by creating a unified approach in which the VEM is employed in both the structural and the optimization phases of the framework. Through two numerical examples with exact solutions, we verify the convergence and accuracy of both the VEM approximations of the displacement and material density fields. We also present several design examples involving non-Cartesian domains, demonstrating the main features of the proposed VEM-based topology optimization framework. Finally, some concluding remarks are provided in Chapter 6 along with several suggestions for potential extensions of the current work.

CHAPTER 2
A PARADIGM FOR HIGHER-ORDER POLYGONAL ELEMENTS IN FINITE ELASTICITY USING A GRADIENT CORRECTION SCHEME

Recent studies have demonstrated that polygonal elements possess great potential in the study of nonlinear elastic materials under finite deformations. On the one hand, these elements are well suited to model complex microstructures (e.g. particulate microstructures and microstructures involving different length scales) and incorporating periodic boundary conditions. On the other hand, polygonal elements are found to be more tolerant to large localized deformations than the standard finite elements, and to produce more accurate results in bending and shear than the standard finite elements. With mixed formulations, lower order mixed polygonal elements are also shown to be numerically stable on Voronoi-type meshes without any additional stabilization treatments. However, polygonal elements generally suffer from persistent consistency errors under mesh refinement with the commonly used numerical integration schemes. As a result, non-convergent finite element results typically occur, which severely limit their applications. In this work, a general gradient correction scheme is adopted that restores the polynomial consistency by adding a minimal perturbation to the gradient of the displacement field. With the correction scheme, the recovery of the optimal convergence for solutions of displacement-based and mixed formulations with both linear and quadratic displacement interpolants is confirmed by numerical studies of several boundary value problems in finite elasticity. In addition, for mixed polygonal elements, the various choices of the pressure field approximations are discussed, and their performance on stability and accuracy are numerically investigated. We present applications of those elements in physically-based examples including a study of filled elastomers with interphasial effect and a qualitative comparison with cavitation experiments for fiber-reinforced elastomers.

2.1 Introduction

The finite element space for polygonal elements contains non-polynomial (e.g. rational) functions and thereby the existing quadratures schemes, typically designed for integration of polynomial functions, will lead to persistent consistency errors that do not vanish with mesh refinement [23], [24]. As a direct consequence, the so-called patch test, which provides a measure of polynomial consistency of conforming discretizations, is not passed on general polygonal meshes, even in an asymptotic sense. Moreover, the persistence of the errors in turn renders the finite element method suboptimally convergent or even non-convergent under mesh refinement. In practice, using a sufficiently large number of integration points can lower the consistency error. For linear polygonal elements in two dimension (2D), a triangulation scheme with three integration points per triangle is shown to be sufficiently accurate for practical problems and mesh sizes [2], [23]. However, for higher-order polygonal elements, for instance, quadratic elements that will be discussed in this chapter, the number of integration points of such a scheme can become prohibitively large. This is also the case for polyhedral elements in three dimensions (3D). For example, maintaining optimal convergence rates with a linear polyhedral discretization for practical levels of mesh refinement may require several hundred integration points per element [24], [68]. As the number of elements increase, the associated computational cost on a polyhedral discretization can be too expensive for practical applications.

Several attempts have been made in the literature to address this issue. For example, in the context of scalar diffusion problems, inspired by the virtual element method (VEM) [25]–[27], Talischi et al. [23] has proposed a polynomial projection approach to ensure the polynomial consistency of the bilinear form, and thereby ensure the satisfaction of the patch test and optimal convergence for both linear and quadratic polygonal elements. A similar approach is also adopted by Manzini et al. [24] to solve Poisson problems on polyhedral meshes. However, those approaches require the existence of a bilinear form, and therefore extension to general nonlinear problem is non-trivial and is still an open question. Borrowing the idea of pseudo-derivatives in the meshless literature [28], Bishop has proposed an approach to correct the derivatives of the shape functions to enforce the linear consistency property on general polygonal and polyhedral meshes [13], [29]. With the correction, the linear patch test is passed and optimal convergence is achieved. Although being applicable for general nonlinear cases, extension to higher order cases(e.g. quadratic polygonal finite elements) is not trivially implied.

More recently, Talischi et al. [68] have proposed a general gradient correction scheme that is applicable to both linear and nonlinear problems on polygonal and polyhedral elements with arbitrary orders. In essence, the scheme corrects the gradient field at the element level with a minimal perturbation such that the discrete divergence theorem is satisfied against polynomial functions of suitable orders. With a minimum accuracy requirement of the numerical integration scheme, the correction scheme has been previously shown to restore optimal convergence for linear diffusion and nonlinear Forchheimer flow problems [68]. In this work, we adopt the gradient correction scheme in two dimensional finite elasticity problems and apply it to linear and quadratic polygonal elements. As we will see, the gradient correction scheme renders both linear and quadratic polygonal elements optimally convergent.

To enable the modeling of materials with a full range of compressibility, this work considers displacement-based as well as two-field mixed polygonal elements, the latter of which contains an additional discrete pressure field. For mixed finite elements, the numerical stability is a critical issue to ensure convergence and therefore has been subjected to extensive studies in the finite element literature. Generally, the stability condition is described by the well-known inf-sup condition [69]–[71], which, in essence, implies a balance between the discrete spaces for displacement field and pressure field [72]. Many of the classical mixed finite elements are known to be unstable (see, for instance, summaries in [73], [74]). As a result, some post-processing procedures or stabilization methods are needed for those elements (see, for instance, [75]–[78]). In contrast, some recent contributions have suggested that linear mixed polygonal elements coupled with element-wise constant pressure field are numerically stable on Voronoi-type meshes in both linear and nonlinear problems if every node/vertex in the mesh is connected to at most three edges [2], [4], [72]. Furthermore, with the availability of higher order displacement interpolants (see, e.g. [36], [79]), various mixed approximations for higher order polygonal elements featuring more enriched pressure spaces are made possible. For example, for mixed quadratic elements, together with quadratic interpolation of the displacement field, the pressure field can be approximated by either discontinuous piecewise-linear or continuous linearly complete functions. However, their stability, convergence and accuracy is still an open problem and has not been fully explored in the literature. In this chapter, the performance of mixed quadratic polygonal elements with different choices of pressure approximations is presented and studied for the first time with thorough numerical assessments. As a direct observation, the mixed quadratic polygonal elements also seem to be stable with both discontinuous piecewise-linear and continuous linearly complete interpolations of the pressure field on Voronoi-type meshes for linear elasticity problems. Intuitively, this stability results from the larger displacement space for polygonal finite elements when compared with the classical triangular and quadrilateral elements.

The remainder of the chapter is organized as follows. Displacement and mixed variational formulations of finite elasticity are briefly recalled in Section 2. In Section 3, finite element spaces and approximations are presented and numerical integration issues are discussed, including a review of the gradient correction scheme. Section 4 presents a numerical study of the convergence, accuracy, and stability for both linear and quadratic, displacement-based and mixed polygonal elements. In Section 5, two examples of practical relevance are studied with polygonal finite elements: (*i*) the nonlinear elastic response of a filled elastomer reinforced with a random isotropic distribution of circular particles bonded through finite-size interphases, and (*ii*) the onset of cavitation in a fiber-reinforced elastomer. Finally, some concluding remarks are recorded in Section 6.

We briefly and partially introduce the notation adopted in this chapter. For any subset E of the given domain Ω, $E \subset \Omega$, we denote by $|E|$ its area or volume and $\langle\cdot\rangle_E$ the average operator:

$$\langle\cdot\rangle_E \doteq \frac{1}{|E|}\int_E (\cdot)\,\mathrm{d}\mathbf{X}. \tag{2.1}$$

If the average is taken for the whole domain Ω, we denote the operator as $\langle\cdot\rangle$ with the subscript Ω omitted. We shall also $||\cdot||_{L^2(E)}$ to denote as the standard L^2-norm over E and $||\cdot||$ as the standard L^2-norm over Ω.

2.2 Finite elasticity formulations

In this section, for later use, two variational formulations of elastostatics are recalled, including the classical displacement-based formulation and a general two-field mixed formulation [80]–[83]. Throughout, a Lagrangian description of the fields is adopted.

Consider a body in its stress-free, undeformed configuration that occupies a domain Ω with boundary $\partial\Omega$. On its boundary, it is subjected to a prescribed displacement field $\mathbf{u}^0$ on $\Gamma^{\mathbf{X}}$ and prescribed surface traction $\mathbf{t}$ (per unit undeformed surface) on $\Gamma^{\mathbf{t}}$, such that $\Gamma^{\mathbf{X}} \cup \Gamma^{\mathbf{t}} = \partial\Omega$ and $\Gamma^{\mathbf{X}} \cap \Gamma^{\mathbf{t}} = \varnothing$. Moreover, it is also assumed to be subjected to a body-force $\mathbf{f}$ (per unit undeformed volume) in Ω. A stored-energy function W is used to characterize the constitutive behavior of the body, which is assumed to be an objective function of the deformation gradient $\mathbf{F}$. In terms of W, the first Piola-Kirchhoff stress tensor $\mathbf{P}$ at each material point $\mathbf{X} \in \Omega$ is given by the following relation:

$$\mathbf{P}(\mathbf{X}) = \frac{\partial W}{\partial \mathbf{F}}(\mathbf{X},\mathbf{F}), \tag{2.2}$$

which is used as the stress measure of choice in this chapter.

2.2.1 Displacement-based formulation

The displacement-based formulation considers the displacement field $\mathbf{u}$ as the only independent field. The deformation gradient $\mathbf{F}$ is then assumed to be a function of $\mathbf{u}$ given by $\mathbf{F}(\mathbf{u}) = \mathbf{I} + \nabla\mathbf{u}$, where ∇ denotes the gradient operator with respect to the undeformed configuration and $\mathbf{I}$ is the identity in the space of second order tensors. According to the principle of minimum potential energy, the unknown equilibrium displacement field $\mathbf{u}$ is the one that minimizes the potential energy Π among the set of all kinematically admissible displacements $\mathbf{v}$:

$$\Pi(\mathbf{u}) = \min_{\mathbf{v}\in\mathscr{K}} \Pi(\mathbf{v}), \tag{2.3}$$

with

$$\Pi(\mathbf{v}) = \int_{\Omega} (\mathbf{X}, \mathbf{F}(\mathbf{v}))\, \mathrm{d}\mathbf{X} - \int_{\Omega} \mathbf{f}\cdot\mathbf{v}\mathrm{d}\mathbf{X} - \int_{\Gamma^{\mathbf{t}}} \mathbf{t}\cdot\mathbf{v}\mathrm{d}S, \tag{2.4}$$

where $\mathscr{K}$ stands for the set of kinematically admissible displacements such that $\mathbf{v} = \mathbf{u}^0$ on $\Gamma^{\mathbf{X}}$.

The weak form of the Euler-Lagrange equations associated with the variational principle (2.3) is given by:

$$G(\mathbf{v}, \delta\mathbf{v}) = \int_{\Omega} \frac{\partial W}{\partial \mathbf{F}}(\mathbf{X}, \mathbf{F}(\mathbf{v})) : \nabla(\delta\mathbf{v})\mathrm{d}\mathbf{X} - \int_{\Omega} \mathbf{f}\cdot\delta\mathbf{v}\mathrm{d}\mathbf{X} - \int_{\Gamma^{\mathbf{t}}} \mathbf{t}\cdot\delta\mathbf{v}\mathrm{d}S = 0 \quad \forall\delta\mathbf{v}\in\mathscr{K}_0, \tag{2.5}$$

where the trial displacement field $\delta\mathbf{v}$ is the variation of $\mathbf{v}$, and $\mathscr{K}_0$ denotes the set of all the kinematically admissible displacement fields that vanish on $\Gamma^{\mathbf{X}}$.

2.2.2 A general two-field mixed variational formulation

The displacement-based formulation is known to perform poorly with standard finite elements when the material under consideration is nearly or purely incompressible. Mixed variational principles are typically adopted as the formulation of choice for such problems instead. Recently, a new two-field mixed variational principle has been put forth by Chi et al. [2], which, unlike the commonly used mixed formulations in the finite element literature, does not require the multiplicative decomposition of the deformation gradient tensor **F** [2]. For the reminder of the chapter, the formulation of Chi et al. [2] is referred to as the **F**-formulation, whereas the commonly used formulation making use of the multiplicative decomposition of **F** into a deviatoric part, $\overline{\mathbf{F}} = (\det \mathbf{F})^{-\frac{1}{3}}\mathbf{F}$, and a hydrostatic part, $(\det \mathbf{F})^{\frac{1}{3}}\mathbf{I}$, is referred to as the $\overline{\mathbf{F}}$-formulation. In this section, the **F**-formulation is briefly reviewed. For the $\overline{\mathbf{F}}$-formulation, the interested reader is referred to [2], [84]–[90] and references therein.

The **F**-formulation consists of finding the equilibrium $(\mathbf{u}, \widehat{p}) \in \mathscr{K} \times \mathscr{Q}$, such that,

$$\widehat{\Pi}(\mathbf{u}, \widehat{p}) = \min_{\mathbf{v} \in \mathscr{K}} \max_{\widehat{q} \in \mathscr{Q}} \widehat{\Pi}(\mathbf{v}, \widehat{q}), \tag{2.6}$$

with

$$\widehat{\Pi}(\mathbf{v}, \widehat{q}) = \int_{\Omega} \left\{ -\widehat{W}^*(\mathbf{X}, \mathbf{F}(\mathbf{v}), \widehat{q}) + \widehat{q}[\det \mathbf{F}(\mathbf{v}) - 1] \right\} \mathrm{d}\mathbf{X} - \int_{\Omega} \mathbf{f} \cdot \mathbf{v} \mathrm{d}\mathbf{X} - \int_{\Gamma^{\mathbf{t}}} \mathbf{t} \cdot \mathbf{v} \mathrm{d}S. \tag{2.7}$$

In the above variational principle, $\mathscr{Q}$ denotes the set of square-integrable scalar functions and the "complementary" stored-energy function $\widehat{W}^*(\mathbf{X}, \mathbf{F}, \widehat{q})$ is defined by partial Legendre transformation:

$$\widehat{W}^*(\mathbf{X}, \mathbf{F}, \widehat{q}) = \max_{J} \left\{ \widehat{q}(J-1) - \widehat{W}(\mathbf{X}, \mathbf{F}, J) \right\}, \tag{2.8}$$

where $\widehat{W}$ is defined such that $\widehat{W}(\mathbf{X}, \mathbf{F}, J) = W(\mathbf{X}, \mathbf{F})$ when $J = \det \mathbf{F}$.

Unlike the $\overline{\mathbf{F}}$-formulation, in which the second independent field agrees precisely with the Cauchy hydrostatic pressure field $p \doteq \operatorname{tr}\boldsymbol{\sigma}$ with $\boldsymbol{\sigma} = J^{-1}\mathbf{P}\mathbf{F}^T$, the maximizing scalar field $\widehat{p}$ is a pressure-like field, henceforth referred to as the pressure field, relates to the hydrostatic pressure field p via the relation

$$p = \widehat{p} - \frac{1}{3\det\mathbf{F}} \frac{\partial \widehat{W}^*}{\partial \mathbf{F}} (\mathbf{X}, \mathbf{F}, \widehat{p}) : \mathbf{F}. \tag{2.9}$$

The weak form of the Euler-Lagrange equations associated with (2.6)–(2.7) reads as

$$\begin{aligned} G^{\mathbf{v}}(\mathbf{v}, \widehat{q}, \delta\mathbf{v}) = \int_{\Omega} \left[-\frac{\partial \widehat{W}^*}{\partial \mathbf{F}} (\mathbf{X}, \mathbf{F}(\mathbf{v}), \widehat{q}) + \widehat{q}\,\mathrm{adj}\left(\mathbf{F}^T(\mathbf{v})\right) \right] : \nabla(\delta\mathbf{v})\mathrm{d}\mathbf{X} \\ - \int_{\Omega} \mathbf{f}\cdot\delta\mathbf{v}\mathrm{d}\mathbf{X} - \int_{\Gamma^{\mathbf{t}}} \mathbf{t}\cdot\delta\mathbf{v}\mathrm{d}S = 0 \quad \forall \delta\mathbf{v} \in \mathscr{K}^0, \end{aligned} \tag{2.10}$$

$$G^{\widehat{q}}(\mathbf{v}, \widehat{q}, \delta\widehat{q}) = \int_{\Omega} \left[\det\mathbf{F}(\mathbf{v}) - 1 - \frac{\partial \widehat{W}^*}{\partial \widehat{q}} (\mathbf{X}, \mathbf{F}(\mathbf{v}), \widehat{q}) \right] \delta\widehat{q}\mathrm{d}\mathbf{X} = 0 \quad \forall \delta\widehat{q} \in \mathscr{Q}, \tag{2.11}$$

where the trail pressure field $\delta\widehat{q}$ is the variation of $\widehat{q}$, and $\mathrm{adj}(\cdot)$ stands for the adjugate operator.

2.3 Polygonal finite elements approximations

This section addresses issues concerning polygonal finite element approximations. In particular, the constructions of conforming finite dimensional displacement space and pressure spaces, for both linear and quadratic polygonal finite elements, are presented. Furthermore, the gradient correction scheme of Talischi et al. [68] is reviewed and the numerical integration schemes used in this work are discussed. Finally, we record Galerkin approximations of the weak form of the Euler-Lagrange equations of the variational principles discussed in the preceding section including the incorporation of the gradient correction scheme for displacement-based and mixed formulations, and demonstrate the polynomial consistency of the proposed approximations.

2.3.1 Finite element spaces

Consider Ω_h to be a finite element decomposition of the domain Ω into non-overlapping polygons, where h is the maximum element size. The boundary of the mesh, denoted as Γ_h is assumed to be compatible with the applied boundary condition, that is, $\Gamma_h^{\mathbf{t}}$ and $\Gamma_h^{\mathbf{X}}$ are both unions of edges of the mesh. We also denote $E \in \Omega_h$ as the element of the mesh. The displacement space associated with the discretization is a conforming finite dimensional space $\mathscr{K}_{h,k}$ that defined as:

$$\mathscr{K}_{h,k} = \left\{ \mathbf{v}_h \in [C^0(\Omega)]^2 \cap \mathscr{K} : \mathbf{v}_h|_E \in [\mathscr{M}_k(E)]^2, \forall E \in \Omega_h \right\}, \tag{2.12}$$

where k is the order of the discretization. In the above definition, $\mathscr{M}_k(E)$ is a finite dimensional space defined over each element E whose basis functions are denoted as $\varphi_i^{(k)}$ henceforth. In this chapter, we consider linear and serendipity quadratic elements, corresponding to the cases of $k = 1$ and $k = 2$, respectively. For the case of $k = 1$, we will omit the superscript "(k)" in $\varphi_i^{(k)}$ and denote the basis functions as φ_i.

For a linear polygonal element E with n edges, the space $\mathscr{M}_1(E)$ is a n dimensional space, with the degrees of freedom at each vertex of E, as shown in Fig. 2.1 (a), which can be defined by a set of generalized barycentric coordinates φ_i. Quite a few barycentric coordinates can be found in the literature [32], [34], [91]–[96], among which the Mean Value coordinates [97] are adopted in this chapter. Over element E, the Mean Value coordinate associated with vertex i is defined as [97]:

$$\varphi_i(\mathbf{X}) = \frac{w_i(\mathbf{X})}{\sum_{j=1}^{n} w_j(\mathbf{X})}, \tag{2.13}$$

with w_i given by

$$w_i(\mathbf{X}) = \frac{\tan\left[\frac{\beta_{i-1}(\mathbf{X})}{2}\right] + \tan\left[\frac{\beta_i(\mathbf{X})}{2}\right]}{||\mathbf{X} - \mathbf{X}_i||}, \tag{2.14}$$

where $\mathbf{X}_i$ is the position vector of vertex i and t_i follow the definition $t_i = \tan(\beta_i/2)$ in

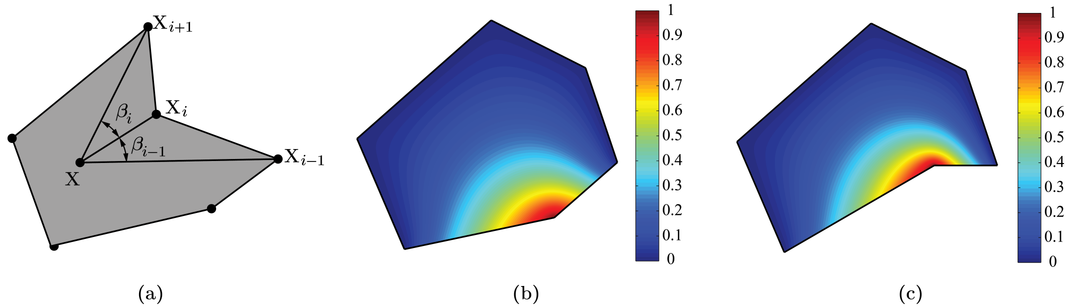


Figure 2.1: (a) Illustration of angles β_i defined in Mean Value Coordinates interpolant w_i. (b) Contour plot of a Mean Value basis vector φ_i over a convex polygon. (c) Contour plot of a Mean Value basis vector φ_i over a concave polygon.

which β_i is the angle defined in Fig. 2.1(a). By defining

$$\mathbf{c}_i = \frac{\mathbf{X}_i - \mathbf{X}}{||\mathbf{X}_i - \mathbf{X}||^2} - \frac{\mathbf{X}_{i+1} - \mathbf{X}}{||\mathbf{X}_{i+1} - \mathbf{X}||^2}, \tag{2.15}$$

the ratio $\mathbf{R}_i \doteq \nabla w_i / w_i$ is expressed as

$$\mathbf{R}_i = \left(\frac{t_{i-1}}{t_{i-1}+t_i}\right)\frac{\mathbf{c}_{i-1}^{\perp}}{\sin\beta_{i-1}} + \left(\frac{t_i}{t_{i-1}+t_i}\right)\frac{\mathbf{c}_i^{\perp}}{\sin\beta_i} + \frac{\mathbf{X}_i - \mathbf{X}}{||\mathbf{X}_i - \mathbf{X}||^2}. \tag{2.16}$$

In the above expression, we have made use of the notation $\mathbf{a}^{\perp} = [-a_y, a_x]^T$ to denote the 90° counterclockwise rotation of a given vector $\mathbf{a} = [a_x, a_y]^T \in \mathbb{R}^2$, and $||\mathbf{a}||$ to denote the its Euclidean norm. As a result, the gradients of the Mean Value coordinates are given by the following [30], [98]:

$$\nabla\varphi_i = \varphi_i\left(\mathbf{R}_i - \sum_{j=1}^{n}\varphi_j\mathbf{R}_j\right). \tag{2.17}$$

Unlike the Wachspress coordinates [30], the gradients of the Mean Value coordinates are shown to stay bounded as the interior angles approaching π [99], meaning that the Mean Value coordinates are able to handle polygons with collinear vertices. Furthermore, the Mean Value coordinates are shown to be well defined for concave polygons [100]. Examples of contour plots of the Mean Value coordinates on a convex and concave polygons are shown in Fig. 2.1 (b) and (c).

For a serendipity quadratic elements ($k = 2$), $\mathscr{M}_2(E)$ is a $2n$ dimensional space having the degrees of freedom at the each vertex of E, as well as the mid-point of each edge, as shown in Figs. 2.2 (a) and (d). According to Rand et al. [79], such a space can be constructed from linear combination of pairwise products of the barycentric coordinates φ_i. Its interpolants $\varphi_i^{(2)}$ are expressed as:

$$\varphi_i^{(2)}(\mathbf{X}) = \sum_{j=1}^{n}\sum_{l=1}^{n} c_{jl}^{i}\varphi_j(\mathbf{X})\varphi_l(\mathbf{X}), \quad i = 1, \cdots, 2n. \tag{2.18}$$

Here, φ_j are the barycentric coordinates for E, which are the Mean Value coordinates in this work. The coefficients c_{jl}^{i} are computed such that any quadratic functions can be interpolated exactly by $\varphi_i^{(2)}$:

$$p(\mathbf{X}) = \sum_{i=1}^{n}\left[p(\mathbf{X}_i)\varphi_i^{(2)}(\mathbf{X}) + p\left(\widehat{\mathbf{X}}_i\right)\varphi_{i+n}^{(2)}(\mathbf{X})\right], \quad \forall p \in \mathscr{P}_2(E), \tag{2.19}$$

where, $\widehat{\mathbf{X}}_i = (\mathbf{X}_i + \mathbf{X}_{i+1})/2$, are the positions of the mid-side nodes. By definition, the coefficients c_{jl}^{i} depend only on the coordinates of the vertices of E. Therefore, the gradients of the interpolants $\varphi_i^{(2)}$ are obtained as:

$$\nabla\varphi_i^{(2)} = \sum_{j=1}^{n}\sum_{l=1}^{n}\left(c_{jl}^{i} + c_{lj}^{i}\right)\nabla\varphi_j(\mathbf{X})\varphi_l(\mathbf{X}). \tag{2.20}$$

Although the construction above is derived assuming that the polygonal elements are strictly convex [79], we find it also seems to be valid for several cases of concave polygons provided that the barycentric coordinates φ_j in Eq.(2.18) are well defined over concave polygons, which holds for Mean Value coordinates. In fact, our numerical assessments in Section 4 suggest that the finite element solutions with certain non-convex polygonal elements indeed converge with their optimal rates. Examples of the basis function constructed using this approach are shown in Figs. 2.2 (b), (c), (e) and (f) on both convex and concave polygons.

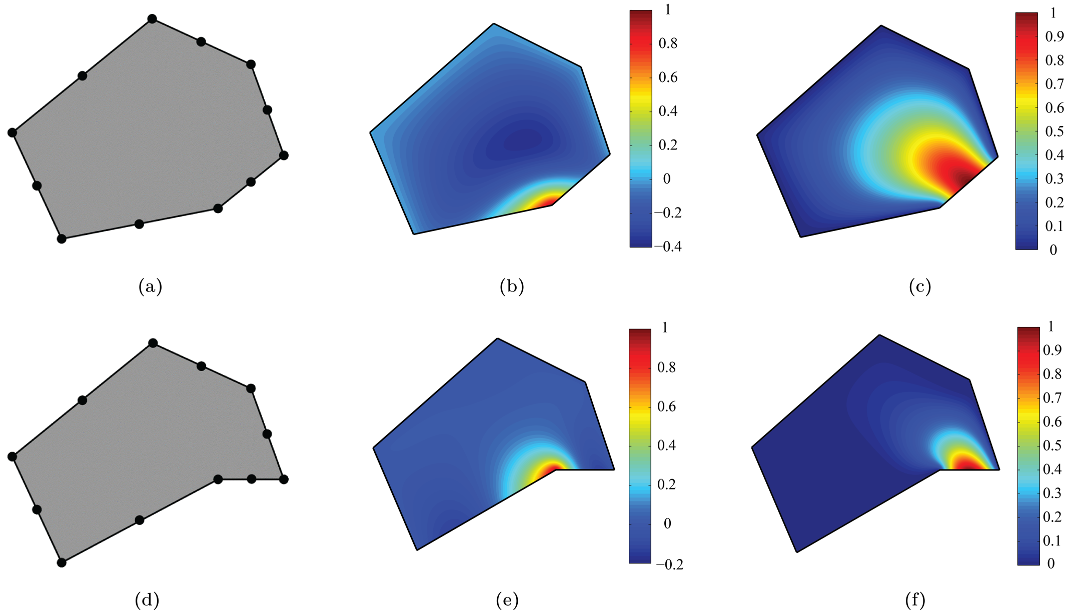


Figure 2.2: (a) Illustrations of the vertex and mid-edge degrees of freedom of a convex polygon. (b) Contour plot of a Mean Value basis $\varphi_i^{(2)}$ associated with a vertex over a convex polygon. (c) Contour plot of a Mean Value basis $\varphi_i^{(2)}$ associated with a mid-edge node over a convex polygon.(d) Illustration of the vertex and mid-edge degrees of freedom of a concave polygon. (e) Contour plot of a Mean Value basis $\varphi_i^{(2)}$ associated with a vertex over a concave polygon. (f) Contour plot of a Mean Value basis $\varphi_i^{(2)}$ associated with a mid-edge node over a concave polygon.

Following their definitions, the spaces $\mathscr{M}_1(E)$ and $\mathscr{M}_2(E)$ contain all the polynomial functions of order k over E, namely,

$$\mathscr{P}_k(E) \subseteq \mathscr{M}_k(E), \quad \forall E \in \Omega_h, k = 1,2, \tag{2.21}$$

where $\mathscr{P}_k(E)$ is the space of polynomial functions of order k. Their basis functions satisfy the Kronecker-delta property, that is $\varphi_i^{(k)}(\mathbf{X}_j) = \delta_{ij}$ with $k = 1,2$. In addition, any functions in $\mathscr{M}_k(E)$ posses kth order polynomial variations on ∂E.

Regarding the two-field mixed finite elements, an approximation of the additional pressure field is needed. As conforming approximations of $\mathscr{Q}$, either discontinuous or continuous approximations can be adopted. For discontinuous approximations, the discrete pressure space $\mathscr{Q}_{h,k-1}^{D}$ can be defined as:

$$\mathscr{Q}_{h,k-1}^{D} = \{q_h \in \mathscr{Q} : q_h|_E \in \mathscr{P}_{k-1}(E), \forall E \in \Omega_h\}, \tag{2.22}$$

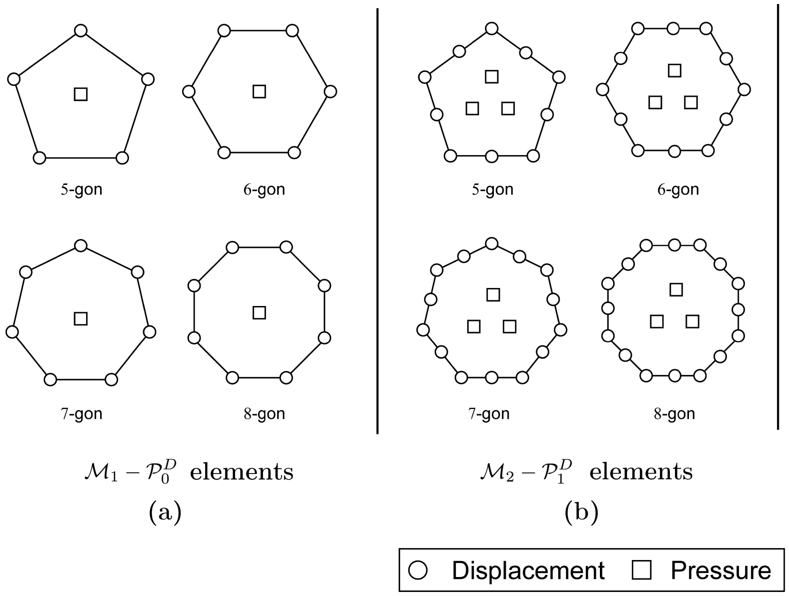
where k is the order of the element. As implied by the above definition, the approximated pressure field may be discontinuous across element boundaries. This type of mixed elements is similar to the Crouzeix-Raviart (C-R) elements in fluid problems[101]. For the remainder of the chapter, we denote this family of mixed element as $\mathscr{M}_k - \mathscr{P}_{k-1}^{D}$ elements, where "$\mathscr{M}$" denotes the Barycentric Mean Value spaces, "$\mathscr{P}^{D}$" stands for polynomial spaces which are discontinuous across element boundaries, and k is the order of the element. For instance, the pressure space of the $\mathscr{M}_1 - \mathscr{P}_0^{D}$ element consists of piecewise constant functions, which are constant over each element. Similarly, the pressure space of the $\mathscr{M}_2 - \mathscr{P}_1^{D}$ element contains piecewise linear functions that vary linearly over each element.

Alternatively, a continuous approximation of the pressure space can be defined in the following manner:

$$\mathscr{Q}_{h,k-1}^{C} = \left\{q_h \in C^0(\Omega) : q_h|_E \in \mathscr{M}_{k-1}(E), \forall E \in \Omega_h\right\}. \tag{2.23}$$

This class of elements resembles the Taylor-Hood (T-H) elements in fluid problems[102] and they are denoted as the $\mathscr{M}_k - \mathscr{M}_{k-1}$ elements for the remainder of the chapter. For convention, we define $\mathscr{M}_0 = \mathscr{P}_0$. As a result, the $\mathscr{M}_k - \mathscr{M}_{k-1}$ elements coincide with the $\mathscr{M}_k - \mathscr{P}_{k-1}^{D}$ elements for the linear case, i.e., $k = 1$.

In this chapter, we will consider both types of mixed polygonal finite elements up to quadratic order. As an illustration, the degrees of freedom (DOFs) of the displacement field and pressure field for those mixed polygonal elements are shown in Fig. 2.3.



$\mathcal{M}_1 - \mathcal{P}_0^D$ elements
(a)

$\mathcal{M}_2 - \mathcal{P}_1^D$ elements
(b)

$\mathcal{M}_2 - \mathcal{M}_1$ elements
(c)

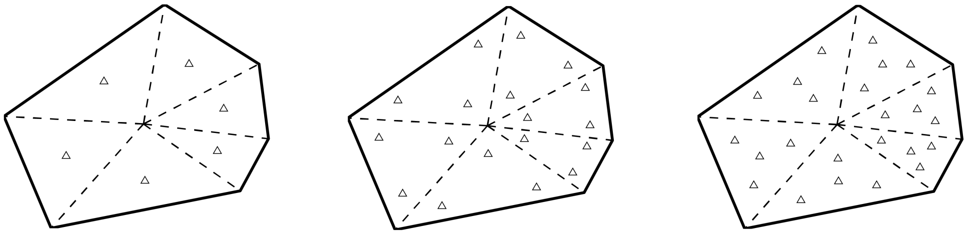
Figure 2.3: Illustration of the degrees of freedom of the displacement field and pressure field for different mixed polygonal element studied in this chapter : (a) $\mathscr{M}_1 - \mathscr{P}_0^D$ elements, (b) $\mathscr{M}_2 - \mathscr{P}_1^D$ elements, and (c) $\mathscr{M}_2 - \mathscr{M}_1$ elements.

2.3.2 Numerical integration and gradient correction scheme

Because of the non-polynomial nature of the space defined by the barycentric coordinates, commonly used quadrature rules for polynomial functions will introduce consistency errors that are persistent with mesh refinement and lead to non-convergent results in finite elasticity problem. Although higher order quadrature rules can reduce the consistency error, they may contain a large amount of integration points and consequently make it computationally expensive to iteratively evaluate stiffness and internal force vectors. To overcome the above-mentioned issues, we introduce the gradient correction scheme in this work to polygonal elements in the context of finite elasticity problems [68]. We note that although the gradient correction theory is applicable to both two dimensional (2D) and three dimensional (3D) problems, the following discussion is restricted to the 2D case.

Consider a general polygonal element E with ∂E denoted as its boundaries and h_E as its diameter. As defined in the previous subsection, $\mathscr{M}_k(E)$ and $\mathscr{P}_k(E)$ are the finite element space and polynomial space defined over E,respectively, which are of order k. In addition, we denote numerical integration schemes, $\int\!\!\!\!\square_E$, on E as an approximation of the area integral, $\int_E$. For the remainder of the chapter, the integration scheme is referred to as mth order if it can integrate polynomial functions of order m exactly. Regarding the boundary integral, one dimensional Gauss–Lobatto quadrature rule is adopted for the line integral $\int_{\partial E}$, which uses two integration points per edge for linear elements and three integration points per edge for quadratic elements.

Accuracy requirements on numerical integrations A minimum accuracy requirement is assumed for the candidate numerical integration schemes of $\int\!\!\!\!\square_E$ [68]. For a fixed element order k, hence the fixed order of space $\mathscr{M}_k(E)$ and $\mathscr{P}_k(E)$, the gradient correction scheme requires the available integration scheme $\int\!\!\!\!\square_E$ to be exact when integrating any polynomial functions of order at least $2k-2$. For instance, the selected scheme should integrate any constants (order 0) and quadratic (order 2) functions exactly for linear elements ($k=1$) and quadratic elements ($k=2$), respectively. Moreover, the integrations scheme $\int\!\!\!\!\square_E$ needs to be sufficiently rich enough to eliminate spurious energy modes. One example of such a scheme is the triangulation scheme [4], [103]. It divides each polygonal element into triangles by connecting the centroid to each vertex and applies available polynomially precise quadrature rules in each triangle. In this chapter, a triangulation scheme with the Dunavant rules [104] in each subdivided triangle is adopted for both linear and quadratic elements. According to the above stated requirements, instead of using a one-point rule that exactly integrates constant functions, the 1st order triangulation scheme containing one integration per subdivided triangle is used for linear element to avoid spurious energy modes. For a quadratic element, the 2nd order triangulation scheme is employed, which contains three integration points per subdivided triangle. Furthermore, the 3rd order triangulation scheme is also used to investigate the effect of increasing integration orders. Illustrations of those schemes are shown in Figs. 2.4 (a)-(c). As a side note, the triangulation scheme requires polygonal elements be star shaped with respect to their centroids, which is the case for all the examples presented in this chapter. However, since the gradient correction scheme is also applicable to other quadrature schemes, as long as they satisfy the accuracy requirement stated in the chapter, other more advance quadrature schemes available in the literature, e.g., [105]–[107], which are specifically designed for integrating polynomial functions over arbitrary polygonal domains, can also be used as $\fint_E$.



(a) (b) (c)

Figure 2.4: Illustration of the "triangulation" schemes for general polygons in physical domain: (a) 1st order triangulation scheme, (b) 2nd order triangulation scheme and (c) 3rd order triangulation scheme.

Under the above stated accuracy requirement, the gradient correction scheme corrects the exact gradient field by adding a small perturbation field to enforce the satisfaction of the discrete divergence theorem at the element level. In the sequel, we first define the gradient correction scheme for scalar problems, and then show its extension to vector problems.

Gradient correction for scalar problems For scalar problems, the corrected gradient, denoted as $\nabla_{E,k} v = \left[\left(\nabla_{E,k} v\right)_x, \left(\nabla_{E,k} v\right)_y \right]^T$, is taken to be closest vector field to $\nabla v = \left[(\nabla v)_x, (\nabla v)_y \right]^T$ that solves the following optimization problem:

$$\min_{\boldsymbol{\zeta}} \fint_E (\boldsymbol{\zeta} - \nabla v) \cdot (\boldsymbol{\zeta} - \nabla v) \, \mathrm{d}\mathbf{X} \tag{2.24}$$

subject to

$$\fint_E \mathbf{p} \cdot \boldsymbol{\zeta} \mathrm{d}\mathbf{X} = \int_{\partial E} (\mathbf{p} \cdot \mathbf{n}) \, v \mathrm{d}S - \fint_E v \, \mathrm{div} \mathbf{p} \mathrm{d}\mathbf{X}, \quad \forall \mathbf{p} \in [\mathscr{P}_{k-1}(E)]^2. \tag{2.25}$$

The above minimization is performed over all the sufficiently smooth functions such that the utilized quadrature makes sense. Since quadrature is used, we note that the above minimization problem only determines $\nabla_{E,k}v$ at the quadrature points and the following analysis shows that the difference of $\nabla_{E,k}v - \nabla v$ equal to an element of $[\mathscr{P}_{k-1}(E)]^2$ at those points. Consider a basis of $[\mathscr{P}_{k-1}(E)]^2$ denoted as $\left\{\boldsymbol{\xi}_1, \cdots, \boldsymbol{\xi}_{n\mathscr{P}_{k-1}}\right\}$ where $n\mathscr{P}_{k-1}$ is the dimension $[\mathscr{P}_{k-1}(E)]^2$. We replace the constraint (2.25) with an equivalent set of constraints:

$$\mathop{\int\!\!\!\!\!\square}_E \boldsymbol{\xi}_a \cdot \boldsymbol{\zeta} \mathrm{d}\mathbf{X} = \int_{\partial E} (\boldsymbol{\xi}_a \cdot \mathbf{n})\, v \mathrm{d}S - \mathop{\int\!\!\!\!\!\square}_E v \,\mathrm{div} \boldsymbol{\xi}_a \mathrm{d}\mathbf{X}, \quad a = 1, \cdots, n\mathscr{P}_{k-1}. \tag{2.26}$$

Introduce a set of Lagrange multipliers, $\lambda_1, \cdots, \lambda_{n\mathscr{P}_{k-1}}$, the Lagrangian of the constrained optimization problem (2.24)–(2.25) takes the form

$$\begin{aligned}\mathscr{L}\left(\boldsymbol{\zeta}, \lambda_1, \cdots, \lambda_{n\mathscr{P}_{k-1}}\right) = &\mathop{\int\!\!\!\!\!\square}_E (\boldsymbol{\zeta} - \nabla v) \cdot (\boldsymbol{\zeta} - \nabla v) \,\mathrm{d}\mathbf{X} \\ &+ \sum_{a=1}^{n\mathscr{P}_{k-1}} \lambda_a \left[\mathop{\int\!\!\!\!\!\square}_E \boldsymbol{\xi}_a \cdot \boldsymbol{\zeta} \mathrm{d}\mathbf{X} - \int_{\partial E} (\boldsymbol{\xi}_a \cdot \mathbf{n})\, v \mathrm{d}S + \mathop{\int\!\!\!\!\!\square}_E v \,\mathrm{div} \boldsymbol{\xi}_a \mathrm{d}\mathbf{X} \right].\end{aligned} \tag{2.27}$$

Taking variation of the Lagrangian with respect to $\boldsymbol{\eta}$, the optimality condition of $\nabla_{E,k}v$ gives

$$\mathrm{D}^{\boldsymbol{\zeta}} \mathscr{L}\left(\nabla_{E,k}v, \lambda_1, \cdots, \lambda_{n\mathscr{P}_{k-1}}\right) \cdot \boldsymbol{\eta} = \mathop{\int\!\!\!\!\!\square}_E \left(\nabla_{E,k}v - \nabla v + \sum_{a=1}^{n\mathscr{P}_{k-1}} \frac{1}{2} \lambda_a \mathbf{p}_a \right) \cdot \boldsymbol{\eta} = 0. \tag{2.28}$$

Therefore, motivated by above analysis, we formally defined $\nabla_{E,k}v$ as the vector field that satisfies the following two conditions:

$$\nabla_{E,k}v - \nabla v \in [\mathscr{P}_{k-1}(E)]^2, \text{ and} \tag{2.29}$$

$$\mathop{\int\!\!\!\!\!\square}_E \mathbf{p}\cdot\nabla_{E,k}v\mathrm{d}\mathbf{X} = \int_{\partial E}(\mathbf{p}\cdot\mathbf{n})\,v\mathrm{d}S - \mathop{\int\!\!\!\!\!\square}_E v\,\mathrm{div}\mathbf{p}\mathrm{d}\mathbf{X}, \quad \forall\mathbf{p}\in\left[\mathscr{P}_{k-1}(E)\right]^2. \tag{2.30}$$

Furthermore, notice that relation (2.30) holds for any functions in $\mathscr{P}_k(E)$. In such cases, as implied by the minimization problem (2.24) and (2.25), the correction function is zero and the corrected gradients coincide with the exact ones, implying

$$\nabla_{E,k}q = \nabla q, \quad \forall q\in\mathscr{P}_k(E). \tag{2.31}$$

By definition, we are able to show that for any sufficiently smooth vector fields $\boldsymbol{\psi}$, the element-level consistency error satisfies the following estimate

$$\mathop{\int\!\!\!\!\!\square}_E \boldsymbol{\psi}\cdot\nabla_{E,k}v\mathrm{d}\mathbf{X} - \int_E \boldsymbol{\psi}\cdot\nabla v\mathrm{d}\mathbf{X} = \mathscr{O}\left(h_E^k\right)||\nabla v||_{\mathscr{L}^2(E)}, \tag{2.32}$$

where h_E is the diameter of E [68].

From a computational perspective, since $\nabla_{E,k}$ is a linear map, only the gradient of each basis function in $\mathscr{M}_k(E)$ needs to be corrected in practice. Here, we present a procedure for computing the corrected gradient of each basis function. We denote $\left\{\varphi_1^{(k)},\cdots,\varphi_{n\mathscr{M}_k}^{(k)}\right\}$ as the basis for $\mathscr{M}_k(E)$, where $n\mathscr{M}_k$ is the dimensions of the space $\mathscr{M}_k(E)$. According to (2.29), we can find a coefficient matrix $\mathbf{S}$ such that

$$\nabla_{E,k}\varphi_i^{(k)} = \nabla\varphi_i^{(k)} + \sum_{a=1}^{n\mathscr{P}_{k-1}} S_{ia}\boldsymbol{\xi}_a, \quad \forall i = 1,...,n\mathscr{M}_k. \tag{2.33}$$

We further define matrices $\mathbf{R}$ of size $n\mathscr{M}_k\times n\mathscr{P}_{k-1}$, and $\mathbf{M}$ of size $n\mathscr{P}_{k-1}\times n\mathscr{P}_{k-1}$ with the following forms:

$$\begin{aligned} R_{ia} &= \int_{\partial E} \left(\boldsymbol{\xi}_a \cdot \mathbf{n}\right) \varphi_i^{(k)} \mathrm{d}S - \fint_E \varphi_i^{(k)} \mathrm{div} \boldsymbol{\xi}_a \mathrm{d}\mathbf{X} - \fint_E \boldsymbol{\xi}_a \cdot \nabla \varphi_i^{(k)} \mathrm{d}\mathbf{X}, \text{ and} \\ M_{ab} &= \fint_{\partial E} \boldsymbol{\xi}_a \cdot \boldsymbol{\xi}_b \mathrm{d}\mathbf{X}. \end{aligned} \tag{2.34}$$

Replacing v and $\mathbf{p}$ with φ_i^k and $\boldsymbol{\xi}_b$ in (2.30) yields the following linear system of equations

$$\sum_{a=1}^{n\mathscr{P}_{k-1}} S_{ia} M_{ab} = R_{ib}, \quad \forall i = 1, ..., n\mathscr{M}_k \text{ and } b = 1, ..., n\mathscr{P}_{k-1}. \tag{2.35}$$

Therefore, the coefficient matrix is obtained as $\mathbf{S} = \mathbf{R}\mathbf{M}^{-1}$.

Gradient correction for vectorial problems When extended to vector field $\mathbf{v} = [v_x, v_y]^T \in [\mathscr{M}_k(E)]^2$, the gradient correction scheme takes the form:

$$\nabla_{E,k} \otimes \mathbf{v} = \begin{bmatrix} \left(\nabla_{E,k} v_x\right)^T \\ \left(\nabla_{E,k} v_y\right)^T \end{bmatrix}. \tag{2.36}$$

Similar to the scalar case, the corrected gradient satisfies the discrete divergence theorem,

$$\fint_E \mathbf{p} : \nabla_{E,k} \otimes \mathbf{v} \mathrm{d}\mathbf{X} = \int_{\partial E} (\mathbf{p}\mathbf{n}) \cdot \mathbf{v} \mathrm{d}S - \fint_E \mathbf{v} \cdot \mathrm{div} \mathbf{p} \mathrm{d}\mathbf{X}, \quad \forall \mathbf{p} \in [\mathscr{P}_{k-1}(E)]^{2\times 2}, \tag{2.37}$$

and, moreover, for any sufficiently smooth 2nd order tensorial fields $\boldsymbol{\psi}$, the element-level consistency error satisfies

$$\fint_E \boldsymbol{\psi} : \nabla_{E,k} \otimes \mathbf{v} \mathrm{d}\mathbf{X} - \int_E \boldsymbol{\psi} : \nabla \mathbf{v} \mathrm{d}\mathbf{X} = \mathscr{O}\left(h_E^k\right) ||\nabla \mathbf{v}||_{\mathscr{L}^2(E)}. \tag{2.38}$$

In computational implementation, assuming the set of basis functions $\left\{\boldsymbol{\varphi}_1^{(k)}, ..., \boldsymbol{\varphi}_{2n\mathscr{M}_k}^{(k)}\right\}$ of $[\mathscr{M}_k(E)]^2$ are of the form

$$\boldsymbol{\varphi}_{2i-1}^{(k)} = \left\{\varphi_i^{(k)}, 0\right\}^T, \quad \boldsymbol{\varphi}_{2i}^{(k)} = \left\{0, \varphi_i^{(k)}\right\}^T, \quad i = 1, ..., n\mathscr{M}_k, \tag{2.39}$$

the correction scheme for vectorial problems in practice amounts to correcting each basis functions as follows

$$\nabla_{E,k}\otimes\boldsymbol{\varphi}^{(k)}_{2i-1} = \begin{bmatrix} \left(\nabla_{E,k}\varphi_i^{(k)}\right)^T \\ \mathbf{0} \end{bmatrix}, \quad \nabla_{E,k}\otimes\boldsymbol{\varphi}^{(k)}_{2i} = \begin{bmatrix} \mathbf{0} \\ \left(\nabla_{E,k}\varphi_i^{(k)}\right)^T \end{bmatrix} \quad i=1,...,n\mathscr{M}_k, \tag{2.40}$$

where $\nabla_{E,k}\varphi_i^k$ is computed according to the above-mentioned procedure described for scalar problems.

2.3.3 Conforming Galerkin approximations

Consider the given discretization Ω_h of the domain and Γ_h of its boundary, we define the numerical integration $\fint_{\Omega_h}$ on Ω_h, as the summation of the contributions from numerical integrals $\fint_E$ from element levels following typical assembly rules, namely, $\fint_{\Omega_h} = \sum_{E\in\Omega_h}\fint_E$, and $\int_{\Gamma_h^{\mathbf{t}}}$ as the numerical integration on $\Gamma_h^{\mathbf{t}}$ based on Gauss–Lobatto rule. In the same fashion, we define the discrete gradient map on the global level, $\nabla_{h,k} : \mathscr{K}_{h,k} \to \left[\mathscr{L}^2\left(\Omega_h\right)\right]^{2\times 2}$, such that it coincides with gradient correction map $\nabla_{E,k}$ at the element level,

$$\left(\nabla_{h,k}\mathbf{v}_h\right)|_E = \nabla_{E,k}\otimes\left(\mathbf{v}_h|_E\right), \quad \forall E\in\Omega_h \text{ and } \mathbf{v}_h\in\mathscr{K}_{h,k}. \tag{2.41}$$

The Galerkin approximation of the displacement-based formulation consists of finding $\mathbf{u}_h \in \mathscr{K}_{h,k}$, such that,

$$G_h\left(\mathbf{u}_h, \delta\mathbf{v}_h\right) = 0, \quad \forall\delta\mathbf{v}_h \in \mathscr{K}^0_{h,k}, \tag{2.42}$$

where $\mathscr{K}^0_{h,k} = \mathscr{K}_{h,k}\bigcap\mathscr{K}^0$ and $G_h\left(\mathbf{u}_h, \delta\mathbf{v}_h\right)$ is the quadrature evaluation of $G\left(\mathbf{u}, \delta\mathbf{v}\right)$ in Eq. (2.5) with the exact gradient operator ∇ replaced by $\nabla_{h,k}$, which takes the form:

$$G_h\left(\mathbf{u}_h, \delta\mathbf{v}_h\right) = \fint_{\Omega_h}\frac{\partial W}{\partial\mathbf{F}}\left(\mathbf{I}+\nabla_{h,k}\mathbf{u}_h\right) : \nabla_{h,k}(\delta\mathbf{v}_h\mathrm{d})\mathbf{X} - \fint_{\Omega_h}\mathbf{f}_h\cdot\delta\mathbf{v}_h\mathrm{d}\mathbf{X} - \int_{\Gamma_h^{\mathbf{t}}}\mathbf{t}_h\cdot\delta\mathbf{v}_h\mathrm{d}S, \tag{2.43}$$

and terms $\mathbf{f}_h$ and $\mathbf{t}_h$ are the approximated body force and boundary traction.

For the two-field mixed formulation, by introducing the additional finite element space $\mathscr{Q}^D_{h,k-1}$(or $\mathscr{Q}^C_{h,k-1}$)$\subseteq \mathscr{Q}$, the Galerkin approximation consist of finding $(\mathbf{u}_h, \widehat{p}_h) \in \mathscr{K}_{h,k} \times \mathscr{Q}^D_{h,k-1}$(or $\mathscr{Q}^C_{h,k-1}$), such that

$$\begin{aligned} G^{\mathbf{v}}_h(\mathbf{u}_h, \widehat{p}_h, \delta\mathbf{v}_h) = 0 \quad & \forall \delta\mathbf{v}_h \in \mathscr{K}^0_{h,k}, \\ G^{\widehat{q}}_h(\mathbf{u}_h, \widehat{p}_h, \delta\widehat{q}_h) = 0 \quad & \forall \delta\widehat{q}_h \in \mathscr{Q}^D_{h,k-1}(\text{or} \mathscr{Q}^C_{h,k-1}). \end{aligned} \tag{2.44}$$

with $G^{\mathbf{v}}_h(\mathbf{u}_h, \widehat{p}_h, \delta\mathbf{v}_h)$ and $G^{\widehat{q}}_h(\mathbf{u}_h, \widehat{p}_h, \delta\widehat{q}_h)$ being of the form

$$\begin{aligned} G^{\mathbf{v}}_h(\mathbf{u}_h, \widehat{p}_h, \delta\mathbf{v}_h) = \fint_{\Omega_h} \left[-\frac{\partial \widehat{W}^*}{\partial \mathbf{F}} \left(\mathbf{I} + \nabla_{h,k}\mathbf{u}_h, \widehat{p}_h \right) + \widehat{p}_h \, \text{adj} \left(\mathbf{I} + (\nabla_{h,k}\mathbf{u}_h)^T \right) \right] : \nabla_{h,k}(\delta\mathbf{v}_h) \text{d}\mathbf{X} \\ - \fint_{\Omega_h} \mathbf{f}_h \cdot \delta\mathbf{v}_h \text{d}\mathbf{X} - \int_{\Gamma^{\mathbf{t}}_h} \mathbf{t}_h \cdot \delta\mathbf{v}_h \text{d}S, \end{aligned} \tag{2.45}$$

$$G^{\widehat{q}}_h(\mathbf{u}_h, \widehat{p}_h, \delta\widehat{q}_h) = \fint_{\Omega_h} \left[\det \left(\mathbf{I} + \nabla_{h,k}\mathbf{u}_h \right) - 1 - \frac{\partial \widehat{W}^*}{\partial \widehat{p}} \left(\mathbf{I} + \nabla_{h,k}\mathbf{u}_h, \widehat{p}_h \right) \right] \delta\widehat{q}_h \text{d}\mathbf{X}. \tag{2.46}$$

Notice that since we replace the gradient operators ∇ of both $\mathbf{u}_h$ and $\mathbf{v}_h$ in Eqs. (2.43), (2.45) and (2.46) with $\nabla_{h,k}$, the resulting approximations yield symmetric linearizations.

We finalize this section with several remarks regarding the performance of the above approximations in patch tests, which are typically adopted to assess the level of consistency error. In the discussions that follow, we restrict our attention to case where the discretization exactly represents the domain and boundary conditions, namely, $\Omega_h = \Omega$, $\Gamma^{\mathbf{t}} = \Gamma^{\mathbf{t}}_h$ and $\Gamma^{\mathbf{X}} = \Gamma^{\mathbf{X}}_h$. As a result, the errors arising from the approximation of the geometry is neglected in the following discussions.

First, we consider the first-order patch test, in which the exact displacement field is a linear vector field, that is, $\mathbf{u} = \mathbf{p}_1 \in [\mathscr{P}_1(\Omega)]^2$. We note that by the polynomial completeness property of the element-level space $\mathscr{M}_1(E)$, the exact displacement field is also in $\mathscr{K}_h$. Accordingly, with any given stored-energy function $W(\mathbf{X},\mathbf{F})$, the body force $\mathbf{f}$ is zero everywhere in Ω and the boundary traction $\mathbf{t}$ on $\Gamma^{\mathbf{t}}$ is given by $\mathbf{t} = \mathbf{Pn}$, where $\mathbf{P} = \partial W(\mathbf{X},\mathbf{F}(\mathbf{p}_1))/\partial\mathbf{F}$. In addition, the associated exact pressure field is found as

$$\widehat{p}_0 = \begin{cases} \text{constant} & \det\mathbf{F}(\mathbf{p}_1) = 1 \\ & \\ \frac{\partial\widehat{W}(\mathbf{X},\mathbf{F}(\mathbf{p}_1),J)}{\partial J}\big|_{J=\det\mathbf{F}(\mathbf{p}_1)} & \text{otherwise} \end{cases}, \tag{2.47}$$

where the constant pressure for incompressible solids is determined by applied boundary traction $\mathbf{t}$. We proceed to verify the exact passage of first-order patch test by showing that $\mathbf{u}_h = \mathbf{p}_1$ and $\widehat{p}_h = \widehat{p}_0$ are solutions to Eqs. (2.42) and (2.44). For displacement-based approximation, upon recognizing

$$\nabla_{h,k}\mathbf{u}_h = \nabla_{h,k}\mathbf{p}_1 = \nabla\mathbf{p}_1 \in [\mathscr{P}_0(\Omega)]^{2\times 2}, \tag{2.48}$$

and

$$\int\!\!\!\!\!\!\square_E \nabla_{E,k}\otimes\mathbf{v}\mathrm{d}\mathbf{X} = \int_E \nabla\mathbf{v}\mathrm{d}\mathbf{X}, \quad \forall\mathbf{v}\in[\mathscr{M}_1(E)]^2 \tag{2.49}$$

we have for any trial displacement field $\delta\mathbf{v}_h \in \mathscr{K}^0_{h,k}$

$$\begin{aligned} G_h(\mathbf{p}_1,\delta\mathbf{v}_h) &= \sum_{E\in\Omega}\int\!\!\!\!\!\!\square_E \frac{\partial W}{\partial\mathbf{F}}(\mathbf{I}+\nabla\mathbf{p}_1) : \nabla_{E,k}(\delta\mathbf{v}_h)\mathrm{d}\mathbf{X} - \int_{\Gamma^{\mathbf{t}}}\mathbf{t}_h\cdot\delta\mathbf{v}_h\mathrm{d}S \\ &= \sum_{E\in\Omega}\int_E \frac{\partial W}{\partial\mathbf{F}}(\mathbf{I}+\nabla\mathbf{p}_1) : \nabla(\delta\mathbf{v}_h)\mathrm{d}\mathbf{X} - \int_{\Gamma^{\mathbf{t}}}\mathbf{t}\cdot\delta\mathbf{v}_h\mathrm{d}S \\ &= \int_\Omega \mathbf{P} : \nabla(\delta\mathbf{v}_h)\mathrm{d}\mathbf{X} - \int_{\Gamma^{\mathbf{t}}}\mathbf{t}\cdot\delta\mathbf{v}_h\mathrm{d}S = 0 \end{aligned} \tag{2.50}$$

In the similar manner, we are able to show that for any $\delta\mathbf{v}_h \in \mathscr{K}^0_{h,k}$,

$$G^{\mathbf{v}}_h(\mathbf{p}_1, \widehat{p}_0, \delta\mathbf{v}_h) = \sum_{E\in\Omega} \fint_E \left[-\frac{\partial \widehat{W}^*}{\partial \mathbf{F}}(\mathbf{I}+\nabla\mathbf{p}_1, \widehat{p}_0) + \widehat{p}_0 \mathrm{adj}\left(\det\left(\mathbf{I}+\nabla\mathbf{p}_1\right)\right)\right] : \nabla_{E,k}(\delta\mathbf{v}_h)\mathrm{d}\mathbf{X} \\ - \int_{\Gamma^{\mathbf{t}}} \mathbf{t}_h \cdot \delta\mathbf{v}_h \mathrm{d}S = 0, \quad (2.51)$$

and for any $\delta\widehat{q}_h \in \mathscr{Q}^D_{h,k-1}$(or $\mathscr{Q}^C_{h,k-1}$),

$$G^{\widehat{q}}_h(\mathbf{p}_1, \widehat{p}_0, \delta\widehat{q}_h) = \sum_{E\in\Omega} \fint_E \left[\det\left(\mathbf{I}+\nabla\mathbf{p}_1\right) - 1 - \frac{\partial \widehat{W}^*}{\partial \widehat{p}}(\mathbf{I}+\nabla\mathbf{p}_1, \widehat{p}_0)\right] \delta\widehat{q}_h \mathrm{d}\mathbf{X} = 0. \quad (2.52)$$

We note the equality (2.52) comes from the fact that

$$\det\left(\mathbf{I}+\nabla\mathbf{p}_1\right) - 1 - \frac{\partial \widehat{W}^*}{\partial \widehat{p}}(\mathbf{I}+\nabla\mathbf{p}_1, \widehat{p}_0) = 0, \quad (2.53)$$

according to (2.47) and the definition of $\widehat{W}^*$. As a result, both displacement-based and mixed approximations of order $k(k \geq 1)$ exactly pass the first-order patch test. In fact, our numerical studies in the subsequent section confirm that the first-order patch test is passed up to machine precision errors.

In contrast, higher order patch test may not be exactly passed in general for finite elasticity problems because of the general forms that $\partial W/\partial \mathbf{F}$, $\partial \widehat{W}^*/\partial \mathbf{F}$ and $\partial \widehat{W}^*/\partial \widehat{p}$ may take. However, the following analysis demonstrates that the associated consistency errors converge to zero with the same rate as the finite element approximation errors (we recall that the approximation errors are typical order k for the kth order element) [108], implying the higher order patch test will be passed asymptotically with mesh refinement. For instance, in the kth order patch test, the exact displacement is taken as a kth order polynomial field, i.e., $\mathbf{u} = \mathbf{p}_k \in [\mathscr{P}_k(\Omega)]^2$, and the body force $\mathbf{f}$, boundary traction $\mathbf{t}$, and pressure field $\widehat{p}$ can be computed accordingly through constitutive and equilibrium equations. For displacement-based finite element approximation, we have for any $\delta\mathbf{v}_h \in \mathscr{K}^0_{h,k}$,

$$
\begin{aligned}
&G_h(\mathbf{p}_k, \delta\mathbf{v}_h) \\
&= \sum_{E\in\Omega} \fint_E \frac{\partial W}{\partial \mathbf{F}}(\mathbf{I}+\nabla\mathbf{p}_k) : \nabla_{E,k}(\delta\mathbf{v}_h)\mathrm{d}\mathbf{X} - \fint_\Omega \mathbf{f}_h \cdot \delta\mathbf{v}_h \mathrm{d}\mathbf{X} - \int_{\Gamma^\mathbf{t}} \mathbf{t}_h \cdot \delta\mathbf{v}_h \mathrm{d}S \\
&= \int_\Omega \frac{\partial W}{\partial \mathbf{F}}(\mathbf{I}+\nabla\mathbf{p}_k) : \nabla(\delta\mathbf{v}_h)\mathrm{d}\mathbf{X} + \mathscr{O}\left(h^k\right)||\nabla(\delta\mathbf{v}_h)|| - \fint_\Omega \mathbf{f}_h \cdot \delta\mathbf{v}_h \mathrm{d}\mathbf{X} - \int_{\Gamma^\mathbf{t}} \mathbf{t}_h \cdot \delta\mathbf{v}_h \mathrm{d}S \\
&= \left(\int_\Omega \mathbf{f} \cdot \delta\mathbf{v}_h \mathrm{d}\mathbf{X} - \fint_\Omega \mathbf{f}_h \cdot \delta\mathbf{v}_h \mathrm{d}\mathbf{X}\right) + \left[\int_{\Gamma^\mathbf{t}} (\mathbf{t}-\mathbf{t}_h) \cdot \delta\mathbf{v}_h \mathrm{d}S\right] + \mathscr{O}\left(h^k\right)||\nabla(\delta\mathbf{v}_h)|| \\
&= \mathscr{O}\left(h^k\right)||\nabla(\delta\mathbf{v}_h)||,
\end{aligned}
\tag{2.54}
$$

where the second equality comes from the estimate (2.38) and the second to last equality is a consequence of the assumed exactness requirements of the volume and boundary integration scheme [42], [109]. For mixed approximation, we can also show in the similar manner that,

$$G^{\mathbf{v}}_h(\mathbf{p}_k, \widehat{p}, \delta\mathbf{v}_h) = \mathscr{O}\left(h^k\right)||\nabla(\delta\mathbf{v}_h)||, \tag{2.55}$$

for any $\delta\mathbf{v}_h \in \mathscr{K}_{h,k}$, and, based on the assumed exactness of volumetric integral $\fint_E$,

$$G^{\widehat{q}}_h(\mathbf{p}_k, \widehat{p}, \delta\widehat{q}_h) = \mathscr{O}\left(h^k\right)||\delta\widehat{q}_h||, \tag{2.56}$$

for any $\delta\widehat{q}_h \in \mathscr{Q}^D_{h,k-1}$(or $\mathscr{Q}^C_{h,k-1}$). Although not presented in this work, our numerical studies indicate that both displacement and mixed approximations indeed asymptotically pass the higher order patch test with their respective optimal convergence rates.

2.4 Numerical assessment

In this section, we present a series of numerical tests to assess the performance of the displacement-based and two-field mixed polygonal elements. Both linear and quadratic elements are considered and investigated. Through the patch test and convergence studies, the effectiveness of the gradient correction scheme in ensuring the convergence of polygonal finite element solutions is demonstrated. Moreover, for mixed polygonal elements, numerical evaluations and discussions on the numerical stability and accuracy for different choice of pressure approximations are also provided.

Throughout the section, plane strain conditions are assumed and material behavior is considered to be Neo-Hookean as characterized by the following stored-energy function:

$$W(\mathbf{F}) = \frac{\mu}{2}[\mathbf{F}:\mathbf{F}-3] - \mu(\det\mathbf{F}-1) + \frac{3\kappa+\mu}{6}(\det\mathbf{F}-1)^2, \tag{2.57}$$

where μ and κ denote the initial shear and bulk moduli of the material response. The corresponding Legendre transformation (2.8) is given by

$$\widehat{W}^*(\mathbf{F},\widehat{q}) = -\frac{\mu}{2}[\mathbf{F}:\mathbf{F}-3] + \frac{3(\mu+\widehat{q})^2}{2(3\kappa+\mu)}. \tag{2.58}$$

Unless otherwise stated, triangulation rules with minimally required orders of accuracy are adopted, which we recall are 1st and 2nd order for linear and quadratic polygonal elements, respectively. In terms of the technique for solving the nonlinear system of equations, the standard Newton-Raphson method is employed and each loading step is regarded as converged once the norm of the residual reduces below 10^{-8} times that of the initial residual. The polygonal meshes used in this section are generated by the general purpose mesh generator for polygonal elements, `PolyMesher` [110].

2.4.1 Displacement-based polygonal finite elements

In this subsection, we provide numerical experiments assessing the performance of the displacement-based polygonal elements with the gradient correction scheme. For comparison purposes, results from triangulation rules but without the correction of the gradients are also provided. In all of the examples considered we use $\mu = \kappa = 1$. Two global error measures are adopted, the L^2-norms and H^1-seminorms of the displacement field errors, which are defined as

$$\varepsilon_{0,\mathbf{u}} = ||\mathbf{u} - \mathbf{u}_h|| \text{ and } \varepsilon_{1,\mathbf{u}} = ||\nabla\mathbf{u} - \nabla\mathbf{u}_h||, \tag{2.59}$$

and evaluated with an 8th order triangulation rule in all the remaining numerical examples.

Patch test We begin with the standard patch test on a unit square domain $\Omega = (0,1)^2$, as depicted in Fig. 2.5 (a). On the boundary of the unit square $\partial\Omega$, an exact displacement field is applied, which is linear in both X and Y directions:

$$u_x(\mathbf{X}) = 2X, \qquad u_y(\mathbf{X}) = -0.5Y. \tag{2.60}$$

Structured polygonal meshes and the centroid Voronoi Tessellation (CVT) meshes are considered, as shown in Fig. 2.5 (b) and (c). Each of the structured polygonal mesh consists of hexagons in the interior and pentagons and quadrilaterals on the boundary. In order to take into account irregular element shapes, the elements in the mesh are slightly elongated in X direction.

We summarize the numerical results of the patch test in Tables 2.1 and 2.2. In the tables, only the more representative H^1-seminorm of the displacement error is presented, and each data for the CVT mesh is obtained by taking an average of errors from a set of three meshes. Without applying the gradient correction scheme, the errors for both linear and quadratic polygonal elements stay constant over the mesh refinement, indicating that the patch test is not passed. Although not presented here, we have observed the same non-vanishing consistency errors with triangulation rules of higher order. In contrast, the errors remain close to machine precision levels for both linear and quadratic polygonal elements when the gradients are corrected, indicating that the patch test is passed. For quadratic elements, an evident accumulation of numerical errors under mesh refined, which is possibly due to the accumulation of numerical errors in computing the shape functions and their gradients.

We note that the similar behavior has also been observed in references [23], [68], where the same construction of quadratic shape function is adopted as this chapter.

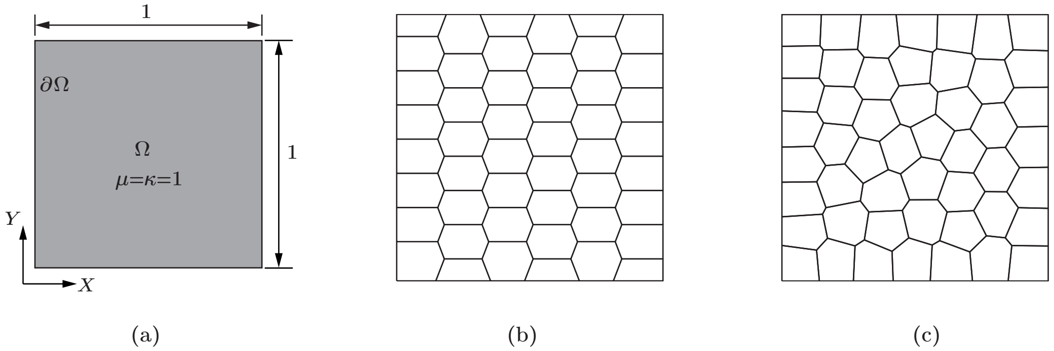


Figure 2.5: (a) Problem setting for the patch test. (b) An illustration of the structured hexagonal-dominant mesh with 48 elements. (c) An illustration of the CVT mesh with 50 elements.

Table 2.1: Results of $\varepsilon_{1,\mathbf{u}}$ for the patch test for both linear and quadratic polygonal elements on structured hexagonal-dominant meshes.

# element	h	Linear polygonal elements		Quadratic polygonal elements	
		Uncorrected	Corrected	Uncorrected	Corrected
130	0.088	0.1168	$5.16E-14$	$1.15E-2$	$4.77E-13$
520	0.044	0.1282	$1.03E-13$	$1.08E-2$	$3.88E-12$
2,080	0.022	0.1339	$2.15E-13$	$1.03E-2$	$3.38E-11$
18,720	0.0073	0.1377	$5.99E-13$	$9.98E-3$	$9.10E-10$

Table 2.2: Results of $\varepsilon_{1,\mathbf{u}}$ for the patch test for both linear and quadratic polygonal elements on CVT meshes.

# element	h	Linear polygonal elements		Quadratic polygonal elements	
		Uncorrected	Corrected	Uncorrected	Corrected
100	0.1	0.141	$4.16E-14$	$2.27E-2$	$2.99E-13$
500	0.045	0.118	$9.25E-14$	$1.18E-2$	$3.35E-12$
2,000	0.022	0.119	$1.86E-13$	$1.24E-2$	$2.64E-11$
20,000	0.0071	0.119	$5.92E-13$	$1.16E-2$	$8.56E-10$

Convergence study A convergence study is performed in which we consider a boundary value problem where a rectangular block of size $\frac{\pi}{3} \times \pi$ is subjected to a certain distribution of body forces [111] so as to be bent into a semicircle; see Fig. 2.6 (a). More precisely, the displacement field is given by

$$u_x(\mathbf{X}) = -1 + (1+X)\cos Y - X, \qquad u_y(\mathbf{X}) = (1+X)\sin Y - Y, \tag{2.61}$$

and the body force by

$$f_x(\mathbf{X}) = -\frac{\cos Y\,(X+1)\,(3\kappa - 2\mu)}{3}, \qquad f_y(\mathbf{X}) = -\frac{\sin Y\,(X+1)\,(3\kappa - 2\mu)}{3}. \tag{2.62}$$

Similar to the patch test discussed above, we first make use of hexagonal-dominated meshes, an example of which is displayed in Fig. 2.6 (b). In order to investigate the effect of the integration order on the convergence and accuracy of the results, we also consider the integration rules that are one order higher than the minimally required one, namely 2nd order for linear elements and 3rd order for quadratic elements. The convergence results are summarized in Figs. 2.7 (a)-(d). For linear polygonal elements, it is clear from the figures that the 1st order integration is not a sufficient scheme to ensure optimal convergence of the finite element solutions without the gradient correction scheme. The L^2-norm of the displacement error shows severely deteriorated convergence and a lack of convergence is observed for the H^1-seminorm of the displacement error. This is due to the dominance of consistency errors observed in the patch test, which do not vanish under mesh refinement. The 2nd order triangulation rule, on the other hand, seems sufficient to ensure enough accuracy and optimal convergence for the range of mesh sizes considered even without the gradient correction scheme. We should note, however, that with further refinement of the mesh, the consistency error will gradually become dominant and the convergence rates of the error norms are expected to decrease accordingly. Unfortunately, it is not the case for quadratic polygonal elements. Without the correction of gradients, both 2nd and 3rd order triangulation rules show severely deteriorated convergence in the L^2-norm of the displacement error and non-convergence in the H^1-seminorm of the displacement error, which indicates that the consistency error plays a dominant role on the convergence of the finite element solutions for quadratic polygonal elements.

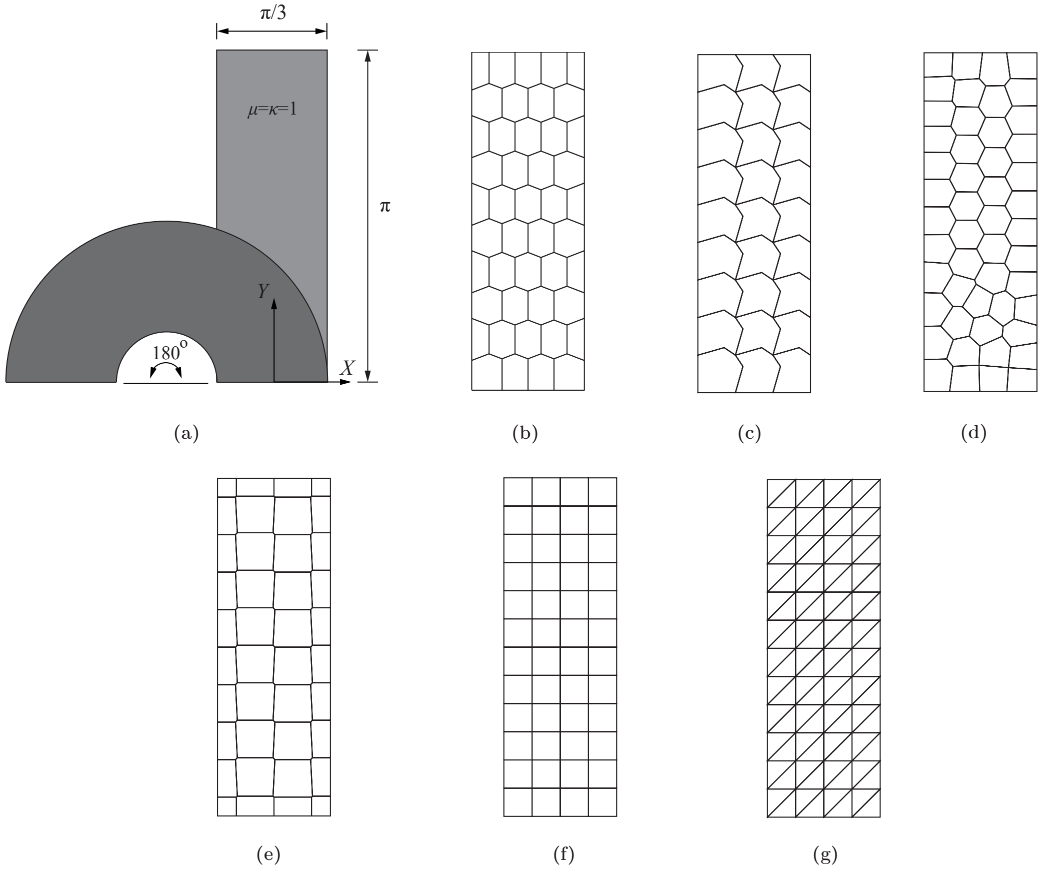


Figure 2.6: (a) Schematic of the bending of a compressible ($\mu = \kappa = 1$) rectangular block into semicircular shape. (b) An example of a structured hexagonal-dominant mesh consisting of 45 elements. (c) An example of the concave octagonal mesh consisting of 27 elements. (d) An example of a CVT mesh consisting of 50 elements. (e) An example of a degenerated Voronoi mesh with small edges consisting of 40 elements. (f) An example of a structured quadrilateral mesh consisting of 48 elements. (g) An example of a triangular mesh consisting of 96 elements.

With the application of the gradient correction scheme, on the contrary, we recover the optimal convergence rates for both linear and quadratic polygonal elements, namely 2 and 1 for the L^2-norm and H^1-seminorm of the displacement errors, respectively, for linear elements, and 3 and 2 for those of the quadratic elements, respectively. Another key observation is that the gradient correction scheme allows the usage of the minimally required order of integration to achieve the same level of accuracy as with higher order integrations. As shown in Figs. 2.7 (a) and (b), the error norms are almost identical for 1st and 2nd order triangulations rules for linear polygonal elements when the gradients are corrected. This suggests that a 1st order integration rule with gradient correction can be used in practice without sacrificing accuracy, which leads to more efficient implementations. Typically, a triangulation rule of order 1 contains n integration points for a n-gon. Compared to the commonly used 2nd order triangulation rule for linear polygons in the literature, which contains $3n$ integration points per n-gon instead, the 1st order triangulation rule can roughly reduce two thirds of the computational cost in forming the stiffness matrices and internal force vectors. The same observations are also made for the quadratic polygonal elements, i.e., the solutions errors are almost identical for 2nd and 3rd integration rules. This indicates that, when the gradient correction scheme is applied, the minimally required 2nd order integration is also sufficient in practice for quadratic polygonal elements.

We also consider a set of concave meshes, an example of which is shown in Fig. 2.6 (c) and use the minimally required integration orders, which are 1st and 2nd orders for linear and quadratic polygonal elements respectively. The numerical results are shown in Fig. 2.8 (a) and (b), which confirm the optimal convergence rates when the gradients are corrected. In fact, the optimal convergence implies the applicability of the quadratic shape functions and the gradient correction scheme adopted in this work to certain concave polygonal elements.

We conclude this subsection with a brief study on the accuracy of the polygonal elements. The polygonal meshes adopted are the CVT meshes and degenerated Voronoi meshes with small edges, as shown in Figs. 2.6 (d) and (e) respectively. Both linear and quadratic polygonal elements are considered. For comparison purpose, we also include the quadrilateral and triangular meshes, examples of which are shown in Figs. 2.6 (f) and (g). Similarly, the triangular and quadrilateral finite elements use the standard iso-parametric construction and are up to quadratic order (for quadratic quadrilateral elements, we use the 8-node serendipity elements). For each type of element, we plot in Figs. 2.9 (a) and (b) the error norms $\varepsilon_{0,\mathbf{u}}$ and $\varepsilon_{1,\mathbf{u}}$ against the total number of nodes under the refinement of mesh,

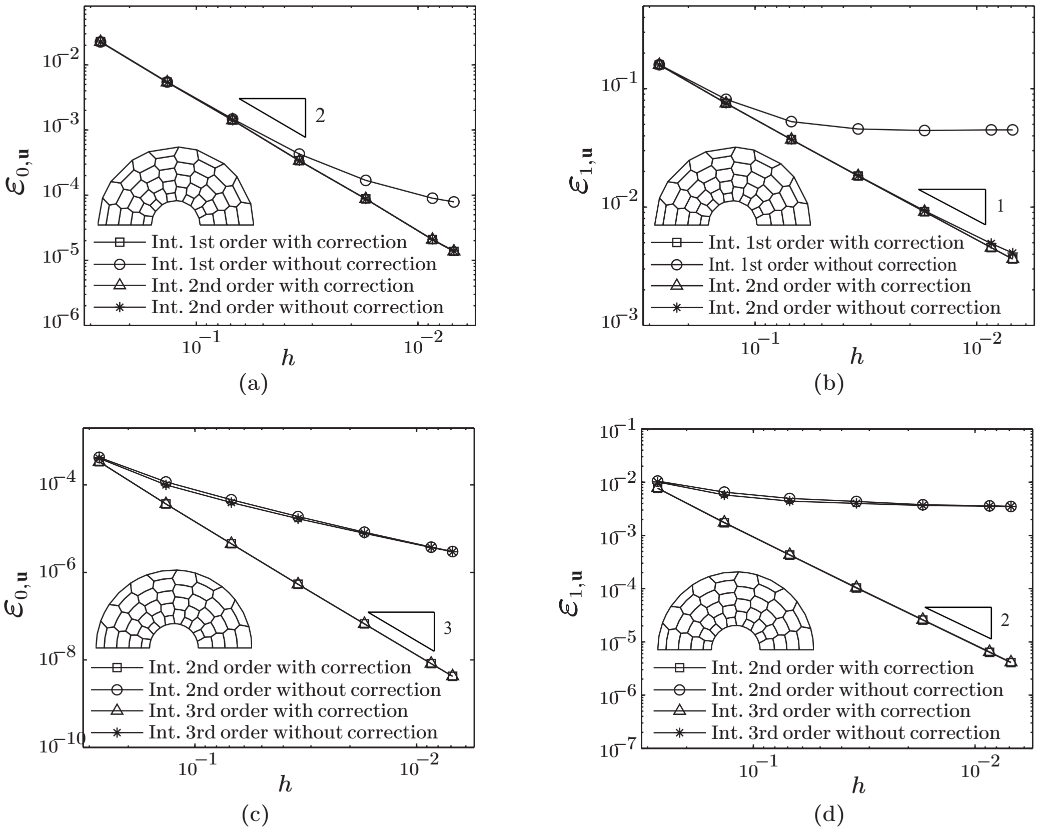
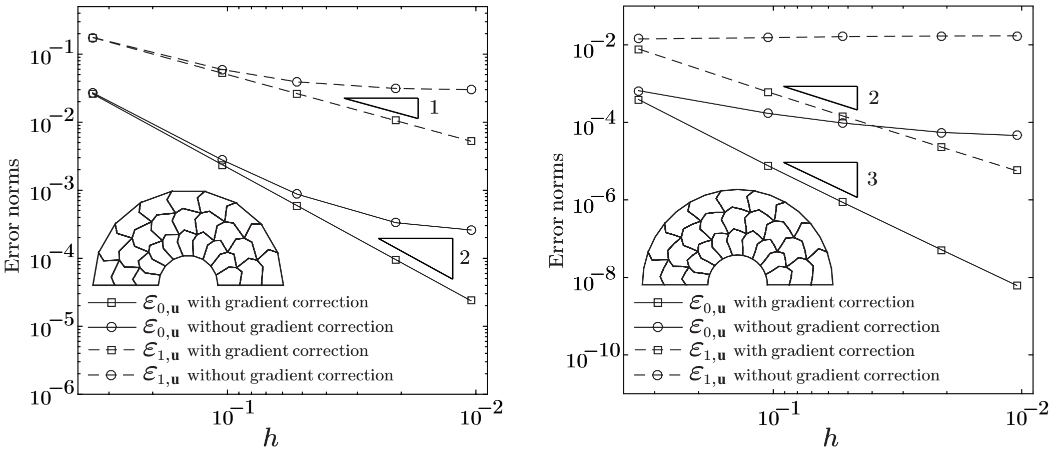


Figure 2.7: Plots of the error norms against the average mesh size h for the structured hexagonal-dominant meshes: (a) the L^2-norm of the error in the displacement field for linear polygonal elements, (b) the H^1-seminorm of the error in the displacement field for linear polygonal elements, (c) the L^2-norm of the error in the displacement field for quadratic polygonal elements and (d) The H^1-seminorm of the error in the displacement field for quadratic polygonal elements.

which reflects the size of the global system of equations and thus correlates with the cost of solving them. Each data point for the CVT meshes is obtained from an average of the errors in three meshes. As we can see from the results in the Figs. 2.9 that the structured quadrilateral meshes provide the most accurate solutions for a given number of nodes in both linear and quadratic cases. This may be attributed to the fact that the exact displacement field for this problem is multiplicatively separable in X and Y and is thus particularly well-suited for approximation by the tensor product in structured quadrilateral meshes. In terms of the L^2-norm of the displacement error, the polygonal meshes (the CVT mesh and degenerated Voronoi meshes with small edges) yields similar accuracy to the triangular



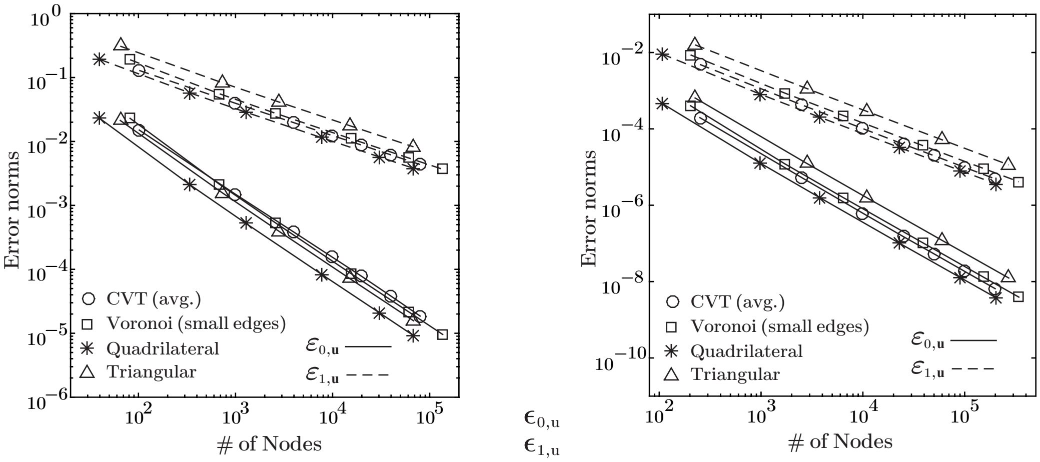
(a) (b)

Figure 2.8: Plots of the error norms against the average mesh size h for the concave octagonal meshes: (a) the L^2-norm and H^1-seminorm of the displacement errors for linear polygonal elements, and (b) the L^2-norm and H^1-seminorm of the displacement errors for quadratic polygonal elements. Only the minimally required orders of integration are used here, i.e., 1st order for the linear polygonal elements and 2nd order for the quadratic ones.

meshes in the linear case, where as in the quadratic case, they are more accurate than the triangular meshes. One the other hand, in terms of the H^1-seminorm of the displacement error, the polygonal meshes yields more accurate results than the triangular meshes in both linear and quadratic cases, meaning that the polygonal meshes are able to approximate the gradient of the displacement field more accurately. Moreover, the comparison between the results from the CVT and degenerated Voronoi meshes with small edges shows that the effect of small edges in the accuracy of polygonal elements is small, indicating the polygonal meshes are tolerant to the presence of small edges.

2.4.2 Two-field mixed polygonal element

In this subsection, together with the gradient correction scheme, the performance of two-field mixed polygonal elements on stability, accuracy and convergence are numerically evaluated. Three types of mixed polygonal finite elements are considered here, namely, $\mathscr{M}_1 - \mathscr{P}_0^D$ elements, $\mathscr{M}_2 - \mathscr{P}_1^D$ elements, and $\mathscr{M}_2 - \mathscr{M}_1$ elements. In this subsection, the material is considered to be incompressible with $\mu = 1$ and $\kappa = \infty$ and thus characterized



(a)

(b)

Figure 2.9: Comparisons of the error norms against the average mesh size h between polygonal meshes (CVT meshes and Voronoi meshes with small edges) and meshes with standard triangular and quadrilateral finite elements: (a) the L^2-norm and H^1-seminorm of the displacement errors for linear elements, and (b) the L^2-norm and H^1-seminorm of the displacement errors for polygonal elements. For the polygonal meshes the results are obtained width corrected gradients.

by the standard Neo-Hookean stored-energy function

$$W(\mathbf{X},\mathbf{F}) = \begin{cases} \dfrac{\mu}{2}\left[\mathbf{F}:\mathbf{F}-3\right] & \text{if} \quad \det\mathbf{F}=1 \\ +\infty & \text{otherwise} \end{cases} . \tag{2.63}$$

In addition to the measure of displacement errors defined in (2.59), we also consider the $\mathscr{L}^2$-norm of the errors in the pressure field,

$$\varepsilon_{0,\widehat{p}} = ||\widehat{p} - \widehat{p}_h||. \tag{2.64}$$

Numerical stability for linear elasticity For mixed finite element methods involving approximations of the displacement and pressure fields, the satisfaction of the stability condition is crucial to guarantee convergence [69]–[71]. In the context of finite elasticity, the stability condition is formally defined by the generalized inf-sup condition [70], [71]. It states that for a given $\mathbf{u}_h \in \mathscr{K}_{h,k}$, there exist a strictly positive, size-independent constant

C_0, such that

$$\beta_h(\mathbf{u}_h) = \inf_{q_h \in \mathscr{Q}^{D \text{ or } C}_{h,k-1}} \sup_{\mathbf{v}_h \in \mathscr{K}_{h,k}} \frac{\int_\Omega q_h \operatorname{adj}\left(\mathbf{I} + (\nabla \mathbf{u}_h)^T\right) : \nabla \mathbf{v}_h \mathrm{d}\mathbf{X}}{||\nabla \mathbf{v}_h|| \, ||q_h||} \geq C_0. \tag{2.65}$$

Note that the above condition depends on the deformation state $\mathbf{u}_h$ in addition to variations $\mathbf{v}_h$ and $\widehat{q}_h$, which is nontrivial to verify. Instead, we only verify the inf-sup condition for linear elasticity in this work, that is

$$\beta^0_h = \inf_{q_h \in \mathscr{Q}^{D \text{ or } C}_{h,k-1}} \sup_{\mathbf{v}_h \in \mathscr{K}_{h,k}} \frac{\int_\Omega q_h \operatorname{div} \mathbf{v}_h \mathrm{d}\mathbf{X}}{||\nabla \mathbf{v}_h|| \, ||q_h||} \geq C_0, \tag{2.66}$$

which can be viewed as a special case of (2.65) when $\mathbf{u}_h = \mathbf{0}$. For meshes consisting of lower order polygonal mixed elements, Beirãoda Veiga et al. [72] have derived a geometrical condition to guarantee the satisfaction of (2.66) if every internal node/vertex in the mesh is connected to at most three edges. For higher order mixed elements, however, the analogous condition is still an open question and is subjected to future research. Here, we numerically evaluate the stability of mixed linear and quadratic polygonal elements. To that end, we adopt the so-called inf-sup test proposed by Chapelle and Bathe [74]. We note that while passing the inf-sup test only constitutes a necessary condition for the satisfaction of the inf-sup condition (2.66), its predictions are shown to reliably match the analytical results [74], [110]. In the test, we consider a unit square domain with imposed boundary conditions as shown in Fig. 2.10 (a). Three families of Voronoi-type meshes are considered here, namely the structured hexagonal, CVT and random Voronoi Meshes, as shown in Fig. 2.10 (b)–(d). In general, the Voronoi-type meshes satisfy the geometrical condition of Beirão da Veiga et al. [72], including the random Voronoi mesh considered here (although they may contain very small edges). However, we note that in several cases, Voronoi meshes from degenerated seeds alignments may be in violation of the geometrical condition and lead to the failure of the inf-sup condition, for instance, the case where the Voronoi seeds are aligned in a Cartesian grid, forming a Cartesian mesh. For all types of meshes considered, we compute the stability index β_h^0 of linear and both types of mixed polygonal elements and plot them as functions of the average mesh size h in Figs. 2.10 (e)–(g). Each point in the plot for CVT and random Voronoi meshes represents an average of the results from a set of three meshes. As suggested by the results of the test, all three types of mixed polygonal finite elements are numerically stable on all families of meshes considered. For comparison purposes, the test results for most of the classical triangular and quadrilateral elements can be found in [73], [74] and hence are not listed here for the sake of conciseness. We note that while the classical mixed linear and quadratic elements with continuous pressure approximations (the T-H family) are unconditionally stable [69], most of those with discontinuous pressure approximations (the C-R family), such as lower order mixed triangular and quadrilateral elements, are numerically unstable [69], [73], [74].

Patch test A patch test study is performed on a unit square domain $\Omega = (0,1)^2$, which is subjected to a uniaxial displacement loading on its right edge, as shown in Fig. 2.11 (a). The analytical displacement is a linear field of the form:

$$u_x(\mathbf{X}) = 2X, \qquad u_y(\mathbf{X}) = -\frac{2}{3}Y, \tag{2.67}$$

and the pressure type field $\widehat{p}$ defined in the **F**-Formulation is a constant over the domain with a value of $\widehat{p} = -\frac{1}{9}$.

The same sets of structured polygonal meshes and CVT meshes are used as that in the patch test study of the displacement-based elements, samples of which are shown in Fig. 2.11 (b) and (c). The patch test results are provided in Tables 2.3–2.6 for mixed linear and quadratic polygonal elements. Again, with the gradient correction scheme, all the error norms stay close to machine precisions for all three type of mixed elements on both sets of meshes, which are not the case for those with uncorrected gradients.

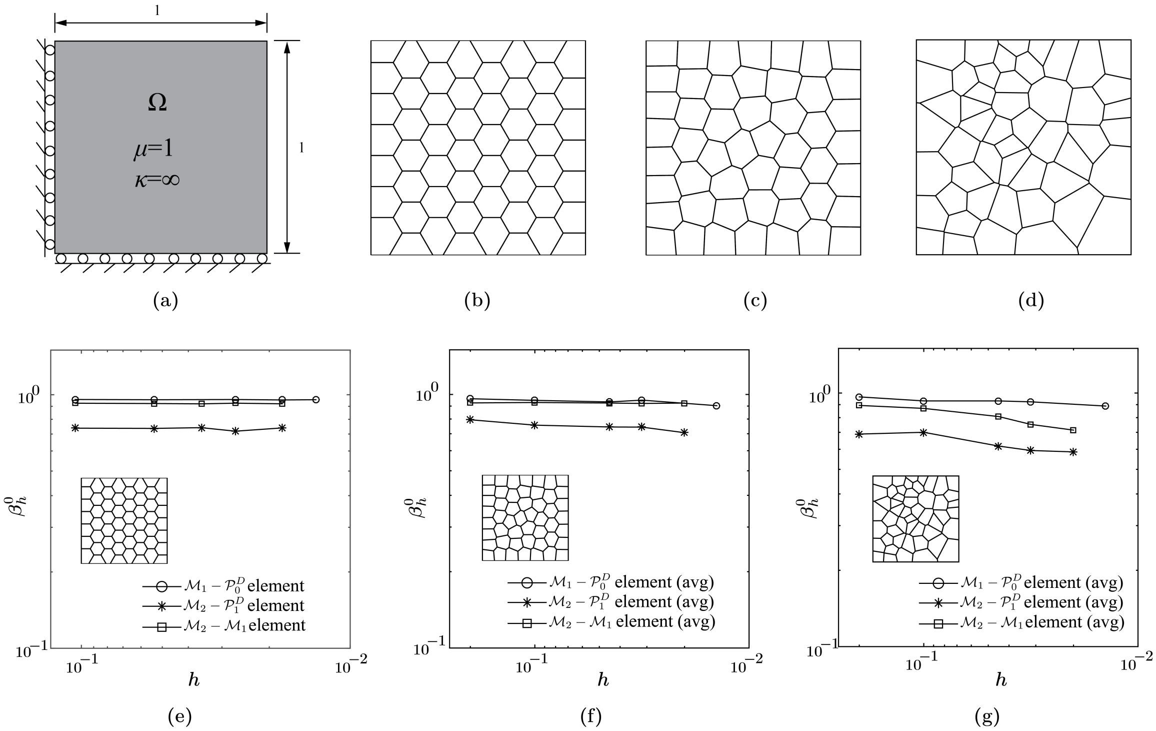


Figure 2.10: (a) Dimensions and boundary conditions for the Inf-Sup test. (b) An example of the structured hexagonal dominant mesh with 56 elements. (c) An example of the randomly generated CVT mesh with 50 elements. (d) An example of the random Voronoi mesh with 50 elements. (e) Plot of the computed value of the stability index as a function of the average mesh size *h* for structured hexagonal dominant meshes. (f) Plot of the computed value of the stability index as a function of the average mesh size *h* for CVT meshes. (g) Plot of the computed value of the stability index as a function of the average mesh size *h* for random Voronoi meshes.

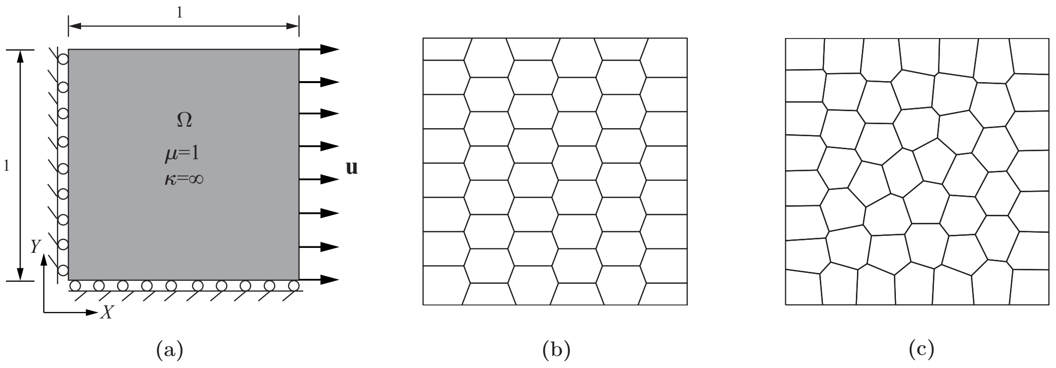


Figure 2.11: (a) Problem setting for first-order patch test; (b) an example of the structured hexagonal-dominated mesh used in the patch test with 48 elements. (c) an example of the CVT mesh used in the patch test with 50 elements.

Table 2.3: Results of patch test with gradient correction for mixed linear polygonal elements with structured hexagonal-dominant meshes.

# element	h	$\varepsilon_{1,\mathbf{u}}$		$\varepsilon_{0,\widehat{p}}$	
$\mathscr{M}_1-\mathscr{P}_0^D$ elements		Uncorrected	Corrected	Uncorrected	Corrected
130	0.088	$1.87E-01$	$5.17E-14$	$1.08E-02$	$3.22E-15$
520	0.044	$1.80E-01$	$1.05E-13$	$1.09E-02$	$7.07E-15$
2,080	0.022	$1.74E-01$	$2.20E-13$	$8.36E-03$	$1.34E-14$
18,720	0.0073	$1.70E-01$	$6.16E-13$	$4.55E-03$	$4.56E-14$

Table 2.4: Results of patch test with gradient correction for mixed linear polygonal elements with CVT meshes.

# element	h	$\varepsilon_{1,\mathbf{u}}$		$\varepsilon_{0,\widehat{p}}$	
$\mathscr{M}_1-\mathscr{P}_0^D$ elements		Uncorrected	Corrected	Uncorrected	Corrected
100	0.088	$1.56E-01$	$4.16E-14$	$4.95E-03$	$2.16E-15$
500	0.044	$1.41E-01$	$9.71E-14$	$5.65E-03$	$5.46E-15$
2,000	0.022	$1.36E-01$	$1.91E-13$	$6.33E-03$	$1.30E-14$
20,000	0.0073	$1.46E-01$	$6.09E-13$	$6.47E-03$	$4.31E-14$

Table 2.5: Results of patch test with gradient correction for mixed quadratic polygonal elements with structured hexagonal-dominant meshes.

# element	h	$\varepsilon_{1,\mathbf{u}}$		$\varepsilon_{0,\widehat{p}}$	
$\mathscr{M}_2-\mathscr{P}_1^D$ elements		Uncorrected	Corrected	Uncorrected	Corrected
130	0.088	$1.37E-02$	$5.59E-13$	$1.08E-03$	$1.10E-13$
520	0.044	$1.26E-02$	$3.95E-12$	$8.61E-04$	$2.84E-13$
2,080	0.022	$1.20E-02$	$3.45E-11$	$6.41E-04$	$1.81E-12$
18,720	0.0073	$1.17E-02$	$9.26E-10$	$3.75E-04$	$4.05E-11$
$\mathscr{M}_2-\mathscr{M}_1$ elements		Uncorrected	Corrected	Uncorrected	Corrected
130	0.088	$1.37E-02$	$4.89E-13$	$9.68E-04$	$5.111E-14$
520	0.044	$1.26E-02$	$3.93E-12$	$8.85E-04$	$2.02E-13$
2,080	0.022	$1.20E-02$	$3.43E-11$	$6.97E-04$	$1.00E-12$
18,720	0.0073	$1.17E-02$	$9.28E-10$	$4.27E-04$	$1.87E-11$

Convergence study We proceed to evaluate the accuracy and convergence of the mixed polygonal finite elements with a boundary value problem where an incompressible rectangular block of dimensions $\frac{\pi}{3}\times\pi$ is bent into semicircular shape. For this boundary value problem, it is possible to work out an analytical solution [80], [112]. Specifically, the

Table 2.6: Results of patch test with gradient correction for mixed quadratic polygonal elements with CVT meshes.

# element	h	$\varepsilon_{1,\mathbf{u}}$		$\varepsilon_{0,\widehat{p}}$	
$\mathscr{M}_2-\mathscr{P}_1^D$ elements		Uncorrected	Corrected	Uncorrected	Corrected
100	0.088	$1.97E-02$	$3.75E-13$	$2.26E-03$	$1.08E-13$
500	0.044	$1.65E-02$	$3.15E-12$	$1.68E-03$	$4.88E-13$
2,000	0.022	$1.43E-02$	$2.84E-11$	$1.48E-03$	$3.34E-12$
20,000	0.0073	$1.40E-02$	$8.72E-10$	$1.37E-03$	$9.90E-11$
$\mathscr{M}_2-\mathscr{M}_1$ elements		Uncorrected	Corrected	Uncorrected	Corrected
100	0.088	$1.56E-02$	$3.14E-13$	$1.90E-04$	$4.08E-14$
500	0.044	$1.48E-02$	$3.40E-12$	$7.38E-05$	$1.68E-13$
2,000	0.022	$1.47E-02$	$2.68E-11$	$3.60E-05$	$8.15E-13$
20,000	0.0073	$1.44E-02$	$8.74E-10$	$1.14E-05$	$1.94E-11$

displacement field and its gradient read as

$$\begin{aligned} u_x &= r(X)\cos Y - r\left(-\frac{\pi}{6}\right) - \frac{\pi}{6} - X, \quad u_y = r(X)\sin Y - Y \\ u_{x,x} &= \frac{\cos Y}{r(X)} - 1, \qquad u_{x,y} = -r(X)\sin Y, \\ u_{y,x} &= \frac{\sin Y}{r(X)}, \qquad u_{y,y} = r(X)\cos Y - 1, \end{aligned} \tag{2.68}$$

where the function $r(X)$ is given by

$$r(X) = \sqrt{2X+\beta} \qquad \text{with} \qquad \beta = \sqrt{\frac{4\pi^2}{9}+4}. \tag{2.69}$$

Additionally, one can also obtain the pressure field $\widehat{p}$:

$$\widehat{p} = -\frac{\mu}{2}\left[\frac{1}{r(X)^2} - r(X)^2 + \mu\beta\right]. \tag{2.70}$$

Figure 2.12 (a) shows a schematic of the problem. In order to avoid the development of surface instabilities [113] and guarantee the uniqueness of the finite element solutions, the displacement is prescribed on the left side of the block.

In addition to the optimal convergence rate for the error measures of the displacement

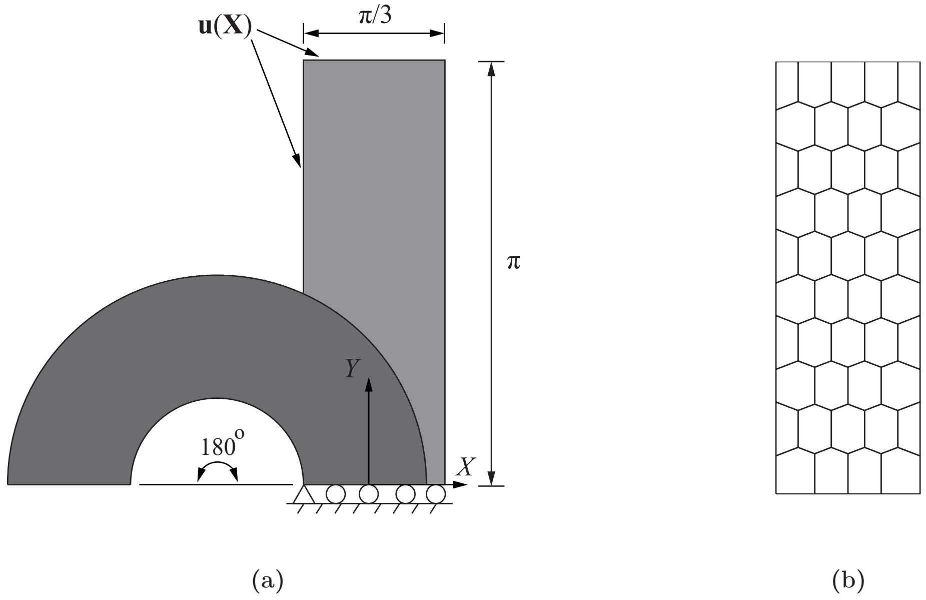
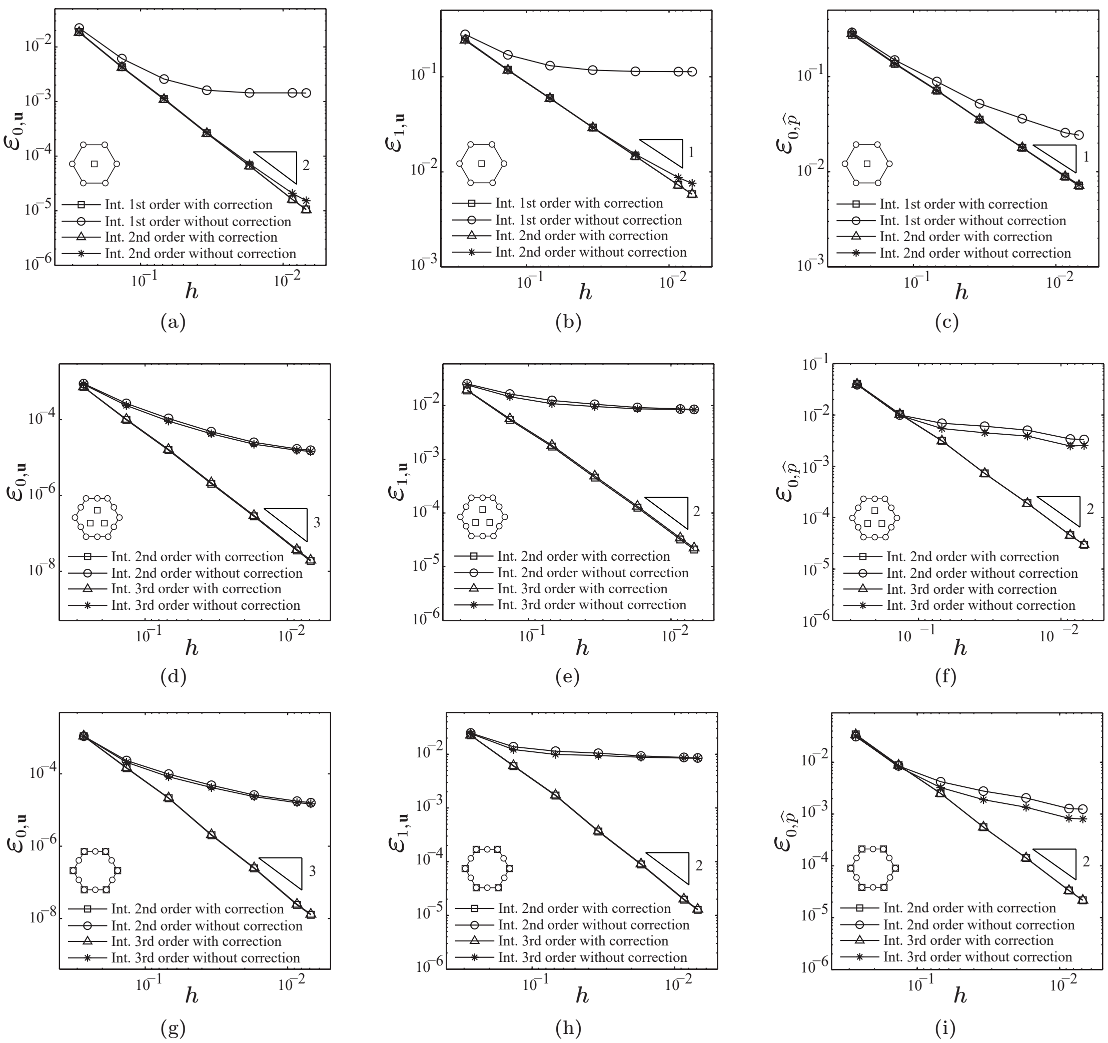


Figure 2.12: (a) Schematic of the bending of an incompressible ($\mu = 1$ and $\kappa = \infty$) rectangular block into semicircular shape. (b) An example mesh utilized in the convergence tests with 45 elements [110].

field discussed in previous subsections with displacement-based elements, we expect the optimal convergence rate for its L^2 pressure error to be 1 for linear mixed elements and 2 for mixed quadratic elements with either $\mathscr{M}_2 - \mathscr{P}_1^D$ or $\mathscr{M}_2 - \mathscr{M}_1$ elements. The results of the convergence test are shown in Figs. 2.13 (a)-(i) for all three types of mixed finite elements considered. When no gradient correction scheme is used, we observe the same behavior regarding the convergence of displacement field solution as those of the displacement-based elements. In terms of the pressure field, although the 1st order integration leads to a degenerated convergence rate, the optimal convergence rate is obtained with the 2nd order integration scheme for linear mixed elements. For mixed quadratic elements, however, the degeneration of the convergence rate becomes more severe and increasing the integration order again proves to be not as helpful as in the linear case. In contrast, with the correction scheme, optimal convergence in both displacement field and pressure field are recovered for both mixed linear and quadratic elements. Again, the same level of accuracy for both displacement and pressure field is achieved with a minimally required order of triangulation



(a) (b) (c) (d) (e) (f) (g) (h) (i)

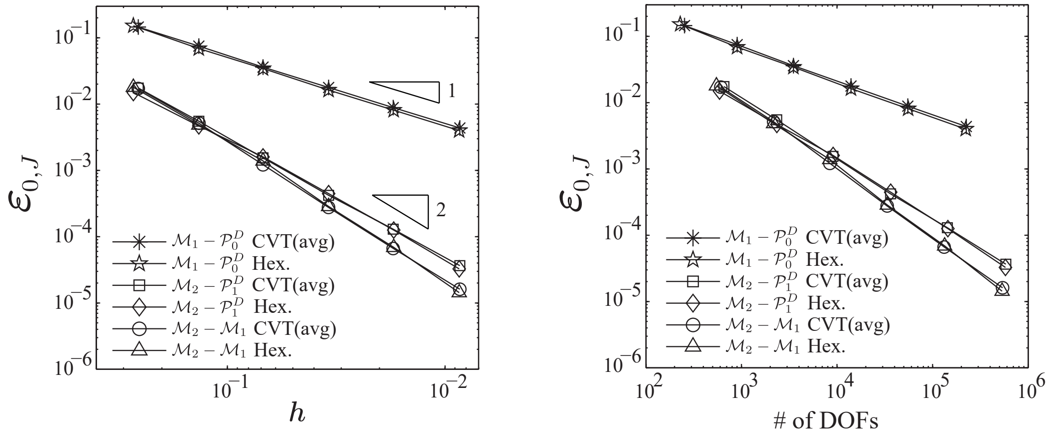
Figure 2.13: Plots of the error norms against the average mesh size h: (a) the L^2-norm of the displacement error, (b) the H^1-seminorm of the displacement error, and (c) the L^2-norm of pressure error for $\mathscr{M}_1 - \mathscr{P}_0^D$ elements; (d) the L^2-norm of the displacement error, (e) the H^1-seminorm of the displacement error, and (f) the L^2-norm of the pressure error for $\mathscr{M}_2 - \mathscr{P}_1^D$ elements; (g) the L^2-norm of the displacement error, (h) the L^1-seminorm of the displacement error, and (i) the L^2-norm of the pressure error for $\mathscr{M}_2 - \mathscr{M}_1$ elements.

rule.

We conclude this section by examining the accuracy of different types of mixed elements in enforcing the incompressibility constraint for the displacement field. To quantify the accuracy, we define $\varepsilon_{0,J}$ as the L^2-norm of the error between $\det \mathbf{F}$ and 1,

$$\varepsilon_{0,J} = ||\det \mathbf{F}(\mathbf{u}_h) - 1||. \tag{2.71}$$

Figures 2.14 (a) and (b) show the convergence of $\varepsilon_{0,J}$ as a function of mesh size and total number of DOFs, respectively, for both hexagonal and CVT meshes [110]. For the CVT meshes, each data point is obtained by averaging values from a set of three meshes with the same number of elements. As we can see, the rate of convergence of $\varepsilon_{0,J}$ to 0 is 1 for mixed linear polygonal elements and roughly 2 for both types of mixed quadratic polygonal elements. In addition, with either similar mesh size or total number of DOFs, $\mathscr{M}_2 - \mathscr{M}_1$ elements seem to better enforce the incompressibility constraint on both hexagonal and CVT meshes for this problem.



(a)

(b)

Figure 2.14: Plots of L^2 error in satisfying the incompressibility constraint versus (a) the average mesh size h and (b) the total number of DOFs.

2.5 Applications

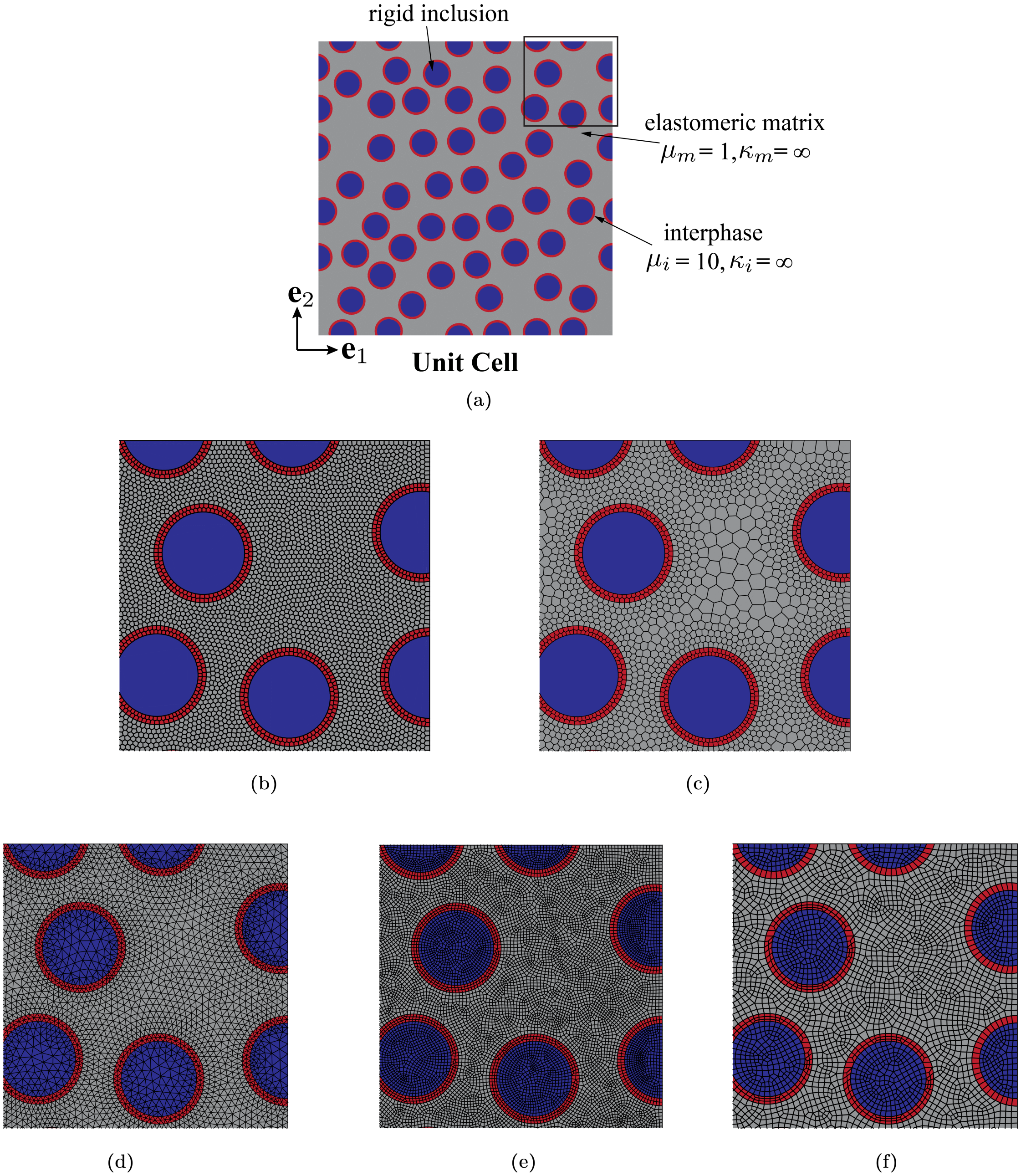
This section presents physically-based applications of the above-developed polygonal finite elements. Two problems are studied: (*i*) the nonlinear elastic response of an incompressible elastomer reinforced with a random isotropic distribution of rigid circular particles bonded through finite-size interphases, and (*ii*) the onset of cavitation in a fiber-reinforced elastomer. These example problems are aimed at demonstrating the ability of polygonal elements to model the complex behavior of nonlinear elastic materials over a wide range of length scales and deformations. For both examples, the inclusions are considered to be infinitely rigid and the variational formulation proposed by Chi et al. [114] is adopted, which treats the presence of rigid inclusions as a set of kinematic constraints on the displacement DOFs of their boundaries. Again, the polygonal meshes considered in this section are generated by `PolyMesher` [110].

2.5.1 Elastomers reinforced with particles bonded through interphases

Experimental evidence has by now established that filled elastomers contain stiff "interphases" or layers of stiff "bound rubber" around its inclusions [115], and that these can significantly affect their macroscopic response when the fillers are submicron in size [116], [117]. In most cases of practical interest, filler particles happen to be indeed submicron in size. In the sequel, motivated by the recent work of Goudarzi et al. [117], we investigate the nonlinear elastic response of an elastomer filled with a random isotropic distribution of circular particles bonded through finite-size stiff interphases by means of the polygonal finite element framework presented in this chapter.

Figure 2.15 shows the unit cell that we consider for the problem at hand, which is assumed to be repeated periodically in the $\mathbf{e}_1$ and $\mathbf{e}_2$ directions so as to approximate a truly random and isotropic distribution of particles. The unit cell contains a total of 50 monodisperse rigid particles at an area fraction of $c_p = 25\%$. Each particle is bonded to the matrix through an interphase of constant thickness t, which is taken to be 20% of the particle radius, resulting in a total area fraction $c_i = 11\%$ of interphases. The matrix phase is modeled as an incompressible Neo-Hookean solid with stored-energy function given by (2.63) with $\mu_m = 1$. Further, the constitutive behavior of the interphases is also taken as incompressible Neo-Hookean, but ten times stiffer than the matrix phase, namely, $\mu_i = 10\mu_m = 10$.

We employ the commonly used mixed formulation in this example (the $\overline{\mathbf{F}}$-formulation) and study the nonlinear elastic behavior of such material by polygonal meshes with mixed



(a)

(b)

(c)

(d)

(e)

(f)

Figure 2.15: (a) The unit cell considered in the problem. (b) The linear polygonal mesh comprised of 39,738 elements and 76,020 nodes in total, with 66,095 nodes in the matrix phase. (c) The quadratic polygonal mesh comprised of 20,205 elements and 95,233 nodes in total, with 67,325 nodes in the matrix phase. (d) The quadratic triangular mesh comprised of 58,814 CPE6MH elements and 118,141 nodes in total, with 71,179 nodes in the matrix phase. (e) The linear quadrilateral mesh comprised of 108,742 CPE4H elements and 109,323 nodes in total, with 68,819 nodes in the matrix phase. (f) The quadratic quadrilateral mesh comprised of 36,444 CPE8H elements and 109,986 nodes in total, with 66,502 nodes in the matrix phase.

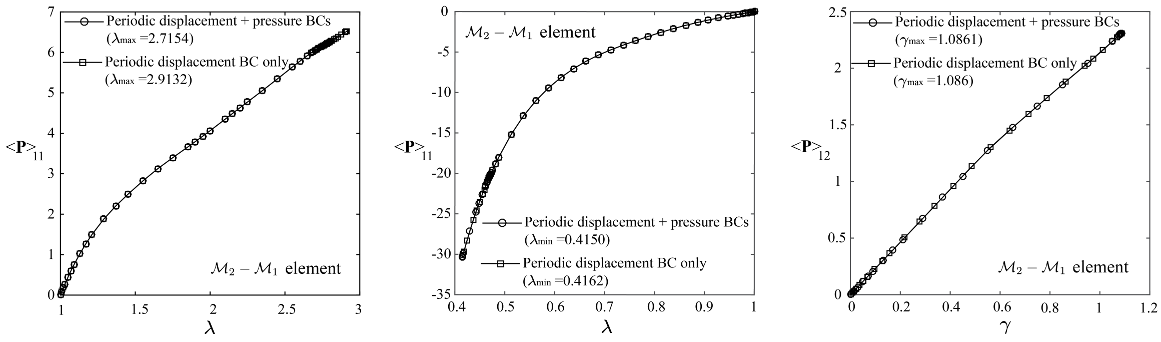
linear and quadratic elements [110]; see Figs. 2.15 (b) and (c). For the quadratic polygonal mesh, elements with both discontinuous and continuous pressure approximations are considered, namely, the $\mathscr{M}_2 - \mathscr{M}_1$ and $\mathscr{M}_2 - \mathscr{P}_1^D$ elements. For comparison purposes, finite element meshes of 4-node hybrid linear quadrilateral elements (termed CPE4H), 6-node hybrid quadratic triangular elements (termed CPE6MH) and 8-node hybrid quadratic quadrilateral elements (termed CPE8H), are also included, the depictions of which can be found in Fig. 2.15 (d)–(f). For a fair comparison, all the meshes are chosen such that they have a similar number of nodes in the matrix phase. For the polygonal mesh, we use the same convergence criterion as the other examples, that is, each loading step is regarded as converged once the norm of the residual reduces below 10^{-8} times that of the initial residual. Whereas for the results by ABAQUS, we use its default set of convergence criteria, among which the permissible ratio of the largest solution correction to the largest corresponding incremental solution is $|\Delta\mathbf{u}|/|\mathbf{u}_{max}| = 10^{-2}$, the largest residual to the corresponding average force norm is 5×10^{-3}. In addition, the largest permissible absolute error in satisfying the incompressibility constraint is10^{-5} in the ABAQUS default convergence criteria.

The boundary conditions are as follows [116]

$$
\begin{aligned}
&\mathbf{u}(1,Y) - \mathbf{u}(0,Y) = \langle\mathbf{F}\rangle - \mathbf{I},\\
&\mathbf{u}(X,1) - \mathbf{u}(X,0) = \langle\mathbf{F}\rangle - \mathbf{I},\\
&p(1,Y) = p(0,Y) \quad \text{and}\\
&p(X,1) = p(X,0).
\end{aligned} \tag{2.72}
$$

and We consider two loading conditions: (*i*) pure shear where $\langle\mathbf{F}\rangle = \lambda\mathbf{e}_1\otimes\mathbf{e}_1 + \lambda^{-1}\mathbf{e}_2\otimes\mathbf{e}_2$ and (*ii*) simple shear where $\langle\mathbf{F}\rangle = \mathbf{I} + \gamma\mathbf{e}_1\otimes\mathbf{e}_2$ with $\lambda > 0$ and $\gamma \geq 0$ denoting the applied stretch and amount of shear, respectively.

Here, it is important to remark that the periodic boundary condition of pressure in (2.72)



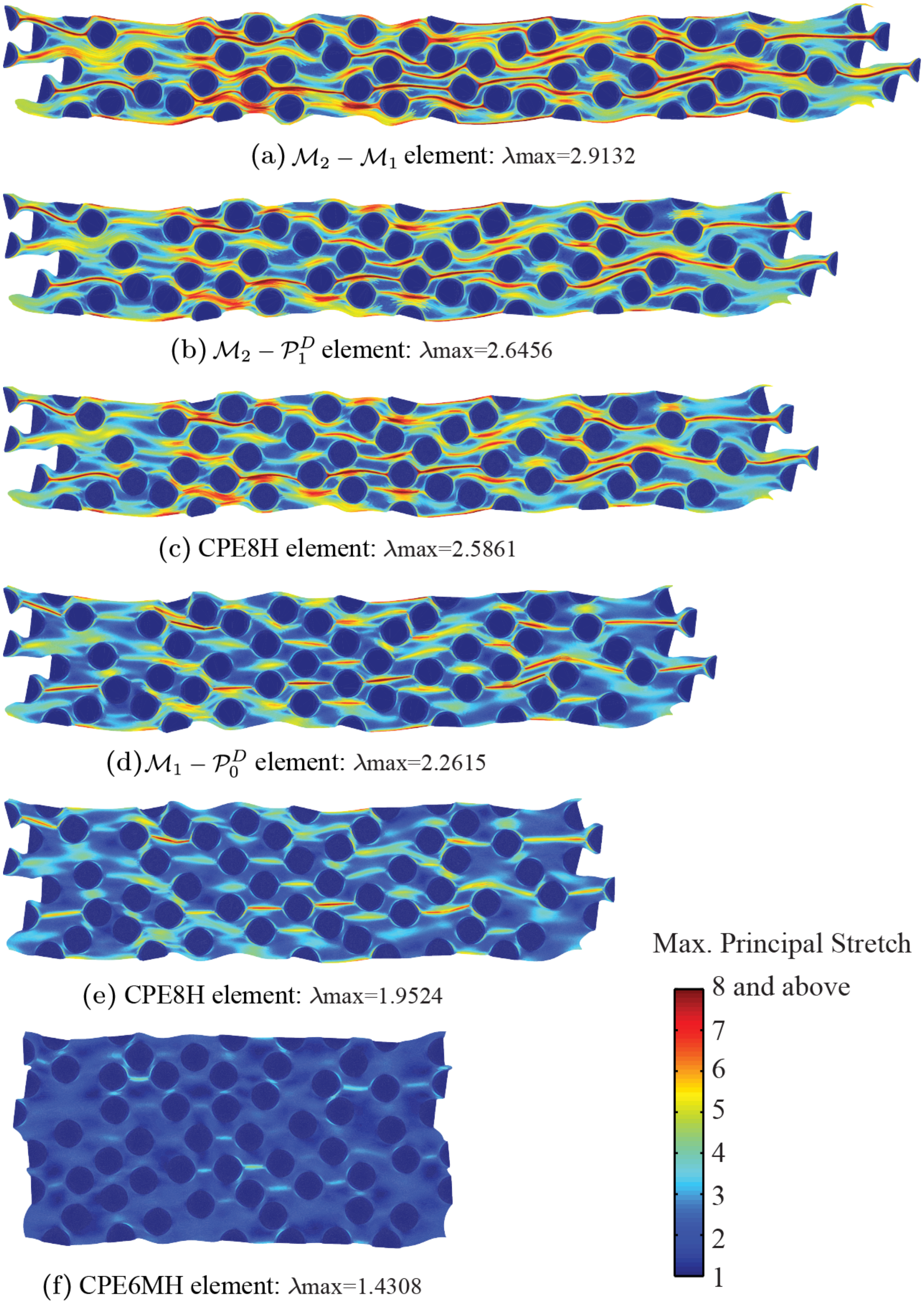
(a) (a) (a)

Figure 2.16: Plots of the macroscopic first Piola-Kirchhoff stress as functions as the applied stretch/shear for the cases of applying both periodic displacement and pressure boundary conditions and applying only periodic displacement boundary condition under (a)-(b) pure and (c) simple shear. The plots are obtained using $\mathscr{M}_2-\mathscr{M}_1$ elements.

cannot be enforced, in general, when using elements with discontinuous pressure fields, such as the $\mathscr{M}_2-\mathscr{P}_1^D$, $\mathscr{M}_1-\mathscr{P}_0^D$, CPE4H, CPE6HM, and CPE8H elements. On the other hand, it can be enforced when using elements with continuous pressure fields, such as the $\mathscr{M}_2-\mathscr{M}_1$ elements. To gain insight into the effects of not enforcing the periodicity of pressure, Figs. 2.16 (a)-(c) show plots of the macroscopic stress response for loading cases (*i*)-(*ii*) against the applied stretch/shear making use of $\mathscr{M}_2-\mathscr{M}_1$ elements with and without the enforcement of periodicity of pressure in (2.72). As it can be seen from the plots, the macroscopic stress from applying both periodic displacement and hydrostatic pressure boundary conditions are, for all practical purposes, identical to the ones from only applying the periodic displacement boundary conditions. Moreover, the unit cell considering only periodicity in the displacement field reaches a larger global stretch. Based on these observations, for the results presented in the remainder of this example, only periodic displacement boundary conditions are applied.

Figures 2.17 (a)-(d) depict the deformed configurations of the unit cell obtained with $\mathscr{M}_2-\mathscr{M}_1$, $\mathscr{M}_2-\mathscr{P}_1^D$, $\mathscr{M}_1-\mathscr{P}_0^D$, CPE4H, CPE6MH, and CPE8H elements, at their respective maximum global stretches, $\lambda_{\max}=2.9132, 2.6456, 2.2515, 1.9524, 1.4308,$ and 2.5861. The fringe scales correspond to the maximum principal stretches of each element, with those having a value of 8 and above being plotted as red. Similarly, Figs. 2.18 (a)-(c) depict the deformed configurations of the unit cell at the same level of global stretch $\lambda = 2$ obtained with $\mathscr{M}_2 - \mathscr{M}_1$, $\mathscr{M}_2 - \mathscr{P}_1^D$, and $\mathscr{M}_1 - \mathscr{P}_0^D$ elements, with the elements whose maximum principal stretches are larger than or equal to 5 plotted red. In addition, the relevant components of the macroscopic first Piola-Kirchhoff stresses $\langle \mathbf{P} \rangle$ as functions of the global stretch λ or shear γ are shown in Figs. 2.19 (a)-(c) for loadings (*i*)-(*ii*), and compared with the available analytical solution in the literature for elastomers reinforced by a random and isotropic distribution of polydisperse circular particles [117], [118].

Several observations can be made from Figs. 2.17, 2.18, and 2.19. For starters, the macroscopic responses computed from all types of elements agree reasonably well with the analytical solution for all loading cases. Yet, the ones from quadrilateral and polygonal discretizations reach significantly larger global deformations than the ones from the triangular mesh. We note that, because of the high effective area fraction considered in this example ($c_p + c_i = 36\%$) and the resulting high level of heterogeneity and localization in the deformation field, remeshing remedies would be of little help to make the unit cell with the triangular meshes (CPE6MH elements) stretch farther, as demonstrated in [119]. Comparing the polygonal meshes and quadrilateral meshes, except for the case of pure shear ($\lambda < 1$), where the quadrilateral mesh with CPE8H elements stretch farther, polygonal meshes can research larger deformation states than the quadrilateral ones, especially under pure shear ($\lambda > 1$) with $\mathscr{M}_2 - \mathscr{M}_1$ and $\mathscr{M}_1 - \mathscr{P}_0^D$ elements. Moreover, although all the polygonal elements produce almost identical deformed configurations at $\lambda = 2$, as shown in Figs. 2.18 (a)-(c), the quadratic polygonal elements exhibit better performance than the linear ones in reaching a larger deformation state. As shown by Figs. 2.17 and 2.19, the unit cell discretized with the $\mathscr{M}_2 - \mathscr{M}_1$ and $\mathscr{M}_2 - \mathscr{P}_1^D$ elements reach significantly larger global deformations than the one with the $\mathscr{M}_1 - \mathscr{P}_0^D$ elements under pure shear. In addition, the linear polygonal elements produce slightly stiffer macroscopic behavior than quadratic polygonal elements. This is because the quadratic elements provide



(a) $\mathcal{M}_2 - \mathcal{M}_1$ element: λmax=2.9132

(b) $\mathcal{M}_2 - \mathcal{P}_1^D$ element: λmax=2.6456

(c) CPE8H element: λmax=2.5861

(d)$\mathcal{M}_1 - \mathcal{P}_0^D$ element: λmax=2.2615

(e) CPE8H element: λmax=1.9524

(f) CPE6MH element: λmax=1.4308

Figure 2.17: The final deformed configuration reached by (a) $\mathscr{M}_2 - \mathscr{M}_1$ elements, (b) $\mathscr{M}_2 - \mathscr{P}_1$ elements, (c) CPE8H elements (solved in ABAQUS), (d) $\mathscr{M}_1 - \mathscr{P}_0$ elements, (e) CPE4H elements (solved in ABAQUS), and (f) CPE6HM elements (solved in ABAQUS).

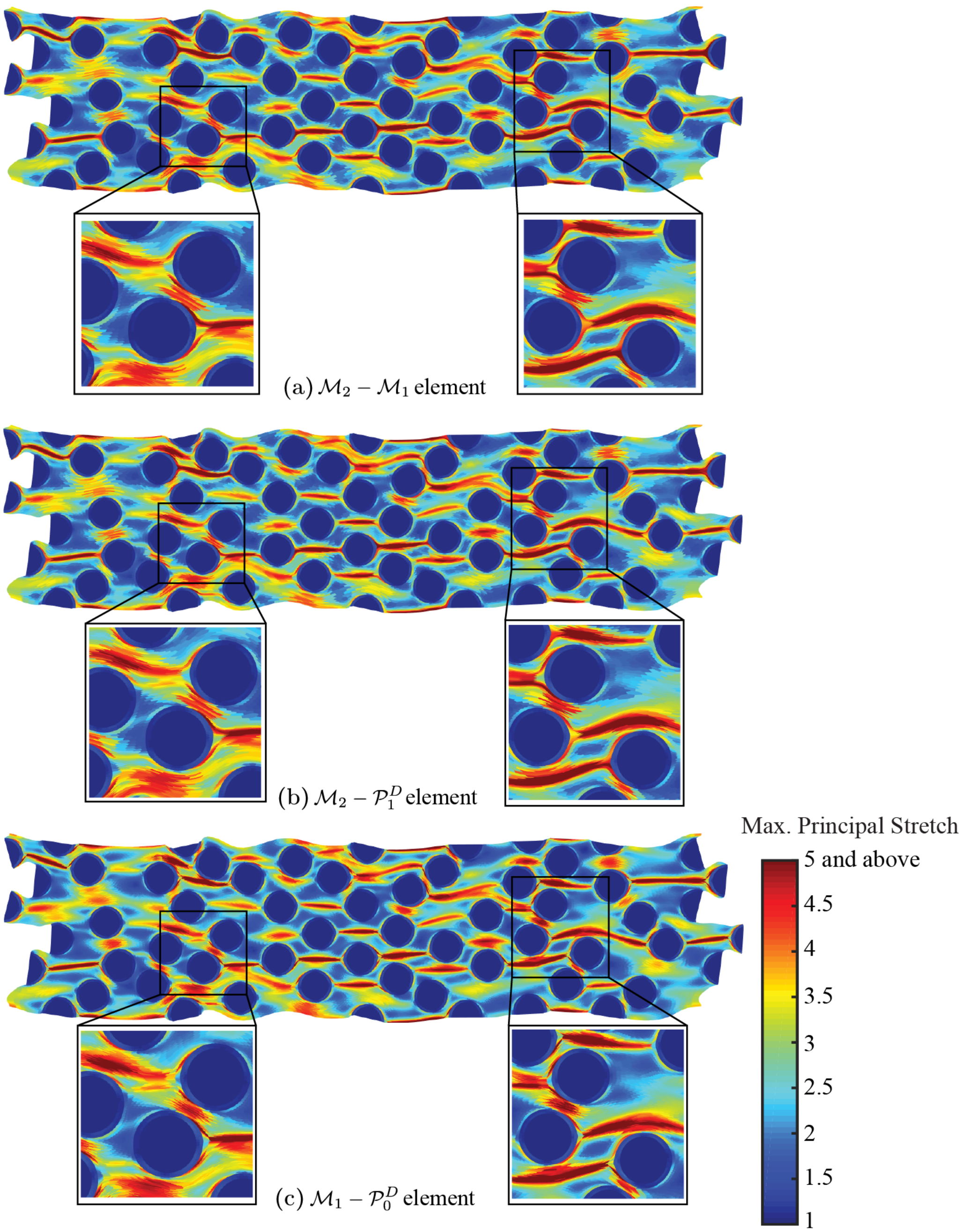
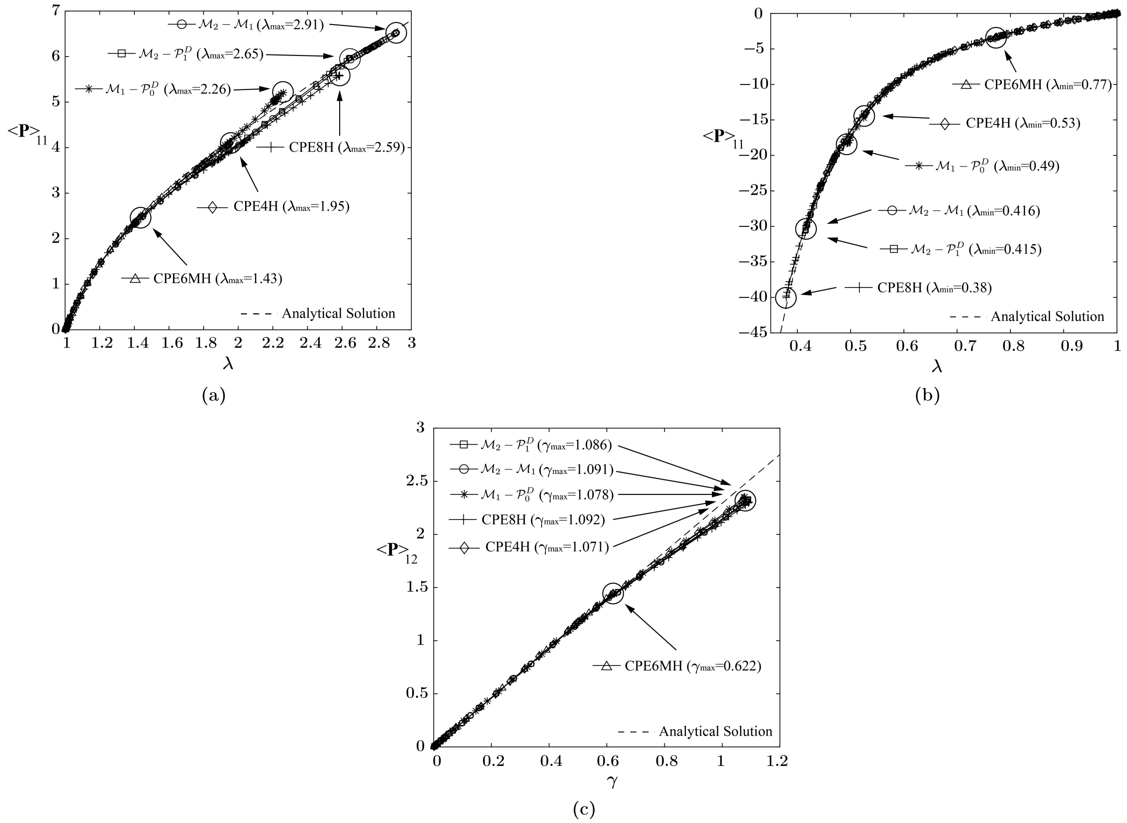


Figure 2.18: The deformed configuration at $\lambda = 2$ obtained by (a) $\mathscr{M}_2 - \mathscr{M}_1$ element, (b) $\mathscr{M}_2 - \mathscr{P}_1^D$ element, and (c) $\mathscr{M}_1 - \mathscr{P}_0^D$ element elements.



(a)

(b)

(c)

Figure 2.19: Plots of the macroscopic first Piola-Kirchhoff stress as functions as the applied stretch/shear for different type of elements under (a) pure shear ($\lambda \geq 1$), (b) pure shear ($0 < \lambda < 1$) and (c) simple shear.

a richer functional approximation of displacements and so they can better represent the curvatures in the highly stretched regions (see, e.g., the red regions in Figs. 2.17 (a)-(c) and Figs. 2.18 (a)-(c)) than the linear ones. We also notice that although yielding almost identical macroscopic responses as the $\mathscr{M}_2 - \mathscr{P}_1^D$ elements, the $\mathscr{M}_2 - \mathscr{M}_1$ elements seem to be able to reach a slightly larger global stretch than $\mathscr{M}_2 - \mathscr{P}_1^D$ elements under pure shear ($\lambda = 2.91$ as compared to $\lambda = 2.65$).

To further understand the performance of each of the four types of elements, we compute the error norm $\varepsilon_{0,J}$ associated with the satisfaction of the incompressibility constraint. The results are summarized in Table 2.7. In essence, the linear mixed polygonal elements are seen to deliver results with larger errors than the quadratic ones. Furthermore, the results from the $\mathscr{M}_2 - \mathscr{P}_1^D$ elements are more accurate in satisfying the incompressibility constraint than the $\mathscr{M}_2 - \mathscr{M}_1$ elements on the same mesh, which is a consequence of the larger number of global pressure DOFs that the $\mathscr{M}_2 - \mathscr{P}_1^D$ mesh contains. As a result, the $\mathscr{M}_2 - \mathscr{M}_1$ elements are able to reach a larger global stretch than the $\mathscr{M}_2 - \mathscr{P}_1^D$ elements under pure shear.

Table 2.7: Error in the approximation of the incompressibility constraint under different loading conditions.

	# of Pressure DOFs	$\varepsilon_{0,J}$ ($\lambda = 2$)	$\varepsilon_{0,J}$ ($\lambda = 0.6$)	$\varepsilon_{0,J}$ ($\gamma = 0.6$)
$\mathscr{M}_1 - \mathscr{P}_0^D$ elements	39,738	0.1180	0.0757	0.0355
$\mathscr{M}_2 - \mathscr{P}_1^D$ elements	60,615	0.0295	0.0163	0.0071
$\mathscr{M}_2 - \mathscr{M}_1$ elements	37,495	0.0508	0.0341	0.0177

2.5.2 The onset of cavitation in fiber-reinforced elastomers

An interesting phenomenon of re-emerging interest that occurs in elastomers is the so-called phenomenon of cavitation, namely, the sudden growth of pre-existing defects into largely enclosed cavities under sufficiently large loads [120]. Cho and Gent [121] studied this phenomenon in fiber-reinforced elastomers by means of some ingenious experiments. As shown in Fig. 2.20 (a), they fabricated specimens consisting of two steel tubes of length

Table 2.8: Summary of the experimental results by Cho and Gent [121] with different initial distance between two steel tubes.

Specimen No.	$L\,(\text{mm})$	$D\,(\text{mm})$	$d\,(\text{mm})$	d/D	$\delta\,(\text{mm})$	$\overline{\varepsilon}_e$
# 1	12.5	9.5	0.43	0.045	0.034	0.08
# 2	12.5	9.5	0.91	0.095	0.255	0.28

L and diameter D, parallel aligned at an initial distance d, bonded by a layer of transparent silicone rubber. The surfaces of the steel tubes were polished and treated with a primer so as to prevent debonding. The specimens were subjected to uniaxial tension transverse to the tubes. The onset of cavitation could then be monitored visually through the transparent rubber.

In this subsection, motivated by the work of Lefèvre et al. [122], we reproduce theoretically (via a 2D plane-strain model) two of the experiments of Cho and Gent [121] by means of the polygonal finite element framework presented in this chapter. Among the several sets of experiments that they reported, we choose two with dimensions given in Table 2.8. In the table, we also include the values of the measured "macroscopic" strain $\overline{\varepsilon}_e$ at which cavitation occurred. In addition, Fig. 2.20 (d) shows a photograph of the cavity formed between the two steel tubes in the specimen with initial distance $d = 0.43$mm.

Given that the length of the steel tubes $L = 12.5$ mm is more than one order of magnitude larger than their initial distance d for both cases ($d = 0.43$ mm and 0.91 mm), and that the cavities observed in the experiments are somewhat cylindrical in shape, as shown by Fig. 2.20 (d), we idealize the experiment as a two-dimensional plane-strain problem and restrict attention to the growth of cylindrical defects. More specifically, we consider a square block of side $H = 300$ mm in which two cylindrical fibers are embedded adjacent to each other. The size of the block is large enough to avoid the interaction between the fibers and the outer boundary. The block is subjected to uniaxial tension in the direction of alignment of the fibers. As in the experiments, the "macroscopic" strain is defined as $\overline{\varepsilon} \doteq h/H - 1$, where h is the deformed length of the block in the direction of applied tension. Figures 2.20 (b) and (c) show schematic descriptions of the model in its undeformed

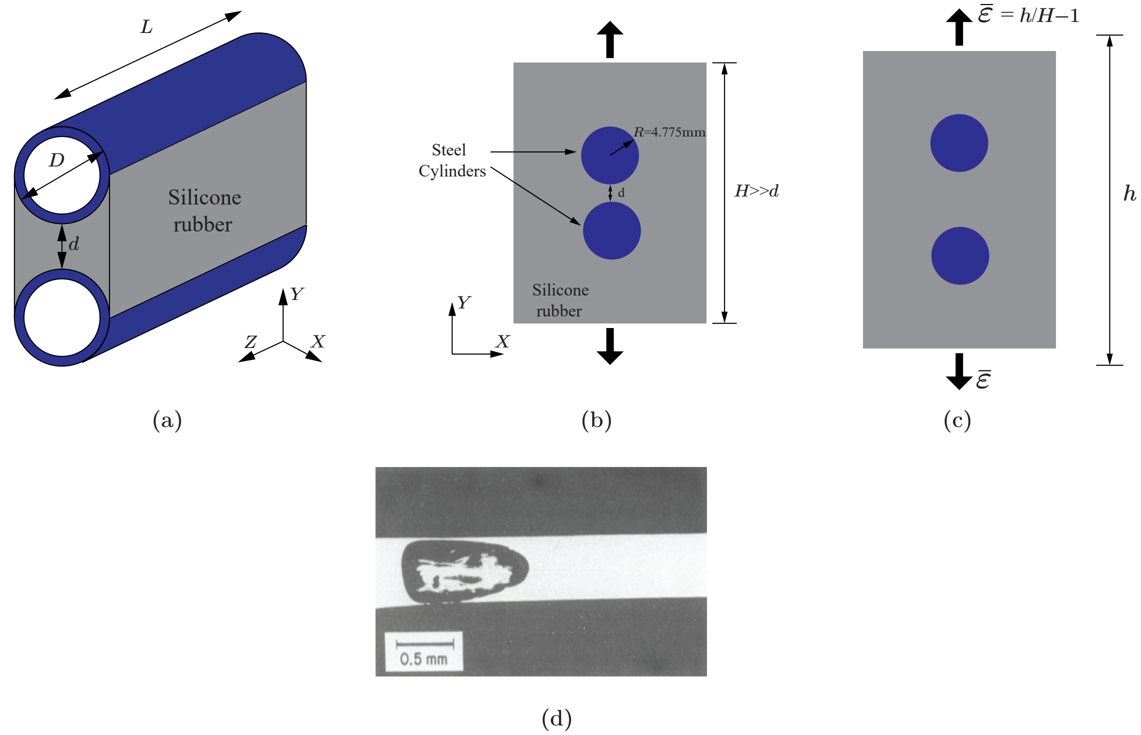


Figure 2.20: (a) Schematic of the experimental setup of Cho and Gent [121]. (b) A two-dimensional plane-strain model of the experiment in the undeformed configuration. (c)A two-dimensional plane-strain model of the experiment in the deformed configuration. (d) A photograph of the lateral view (X direction) of the cavity formed between the two steel tubes for the case of $d = 0.43$mm [121].

and deformed configurations. The silicone rubber is modeled as a one-term compressible Lopez-Pamies material [123]:

$$W(\mathbf{F}) = \frac{3^{1-\alpha}}{2\alpha}\mu\left[(\mathbf{F}:\mathbf{F})^{\alpha} - 3^{\alpha}\right] - \mu(\det\mathbf{F}-1) + \left(\frac{\mu}{2}+\frac{\kappa}{2}-\frac{\alpha\mu}{3}\right)(\det\mathbf{F}-1)^2. \quad (2.73)$$

Its Legendre transformation $\widehat{W}^*(\mathbf{F},\widehat{q})$ is given by:

$$\widehat{W}^*(\mathbf{F},\widehat{q}) = -\frac{3^{1-\alpha}}{2\alpha}\mu\left[(\mathbf{F}:\mathbf{F})^{\alpha} - 3^{\alpha}\right] + \frac{(\mu+\widehat{q})^2}{2\left(\mu+\kappa-\frac{2}{3}\alpha\mu\right)}. \quad (2.74)$$

The shear modulus μ of the rubber is taken to be 1 and, to reasonably model the nearly incompressible behavior of the silicone rubber, its bulk modulus κ is taken to be 1000. The material parameter α is taken to be 0.6 here in order to ensure appropriate growth conditions that allow for cavitation in 2D [124]. Moreover, again, the fibers are treated as infinitely rigid bodies (since the shear modulus of the steel is in the order of 80 GPa) that are perfectly bonded to the rubber. In all of the calculations for this example problem, we employ the **F**-formulation of Chi et al. [2].

From a computational point of view, the modeling of cavitation is challenging with standard finite elements because of the extreme local deformations that take place around the boundaries of the growing defects [125], [126]. In the literature, use has been made of structured quadrilateral discretizations which have proved successful in a variety of problems (see, e.g., [122], [126]). However, such an approach is not deemed viable to deal with material systems with realistically complex microstructures. Unstructured polygonal discretizations, on the other hand, have recently shown potential as a viable approach [2]. To further examine the performance of different types of mixed polygonal elements for cavitation problems, we consider unstructured polygonal discretizations and consider all three types of mixed polygonal elements in this example.

Following the analysis of Lefèvre et al. [122], we begin by monitoring the onset of cavitation for the two cases with initial fiber distances $d = 0.41$ and 0.91 mm. Although the actual growth of defects in rubber is a complex process that may involve irreversible inelastic deformations (e.g., fracture), cavitation criteria based on purely elastic deformations have been shown to provide accurate qualitative agreement with experimental observations, suggesting that the elastic behavior of rubber plays a key role in the occurrence of the phenomenon [122]. For a review of those criteria, the interested reader is referred to the papers by Lopez-Pamies et al. [127], [128]. For the problem under investigation here, we make use of the cavitation criterion derived by Lopez-Pamies [129] for compressible isotropic solids under general plane-strain loading conditions. The criterion states that, under plane-strain conditions, defects inside a compressible isotropic solid may suddenly grow into cylindrical cavities whenever the principal stretches (λ_1, λ_2) satisfy the following condition:

$$\frac{\dfrac{\partial\Phi}{\partial\lambda_1}(\lambda_1,\lambda_2)-\dfrac{\partial\Phi}{\partial\lambda_2}(\lambda_1,\lambda_2)}{\lambda_1-\lambda_2}=0, \tag{2.75}$$

where $\Phi(\lambda_1,\lambda_2)$ stands for the stored-energy function, written in terms of the in-plane principal stretches, characterizing the elastic response of the rubber under plane-strain conditions. Upon recognizing that $\mathbf{F}:\mathbf{F}=\lambda_1^2+\lambda_2^2+1$ and $\det\mathbf{F}=\lambda_1\lambda_2$, the criterion (2.75) reduces to

$$\kappa+2\mu-\frac{2\alpha\mu}{3}+3^{1-\alpha}\mu\left(\lambda_1^2+\lambda_2^2+1\right)^{(\alpha-1)}-\lambda_1\lambda_2\left(\kappa+\mu-\frac{2\alpha\mu}{3}\right)=0 \tag{2.76}$$

when specialized to the stored-energy function (2.73).

To monitor the onset of cavitation criterion (2.76), we make use of a polygonal mesh with 30000 $\mathscr{M}_1-\mathscr{P}_0^D$ elements, which are graded so that smaller elements are placed in between the fibers [110]. Figures 2.21 shows the mesh utilized for the case with $d=0.43$ mm. Figures 2.22 (a) and (b) depict the deformed states of the region between the two rigid fibers with initial distances $d=0.41$ mm and 0.91 mm at four levels of the applied macroscopic strain. In the figures, those elements whose average in-plane principal stretches satisfy the criterion (2.76) are plotted in red, indicating the possible onset of cavitation at the material points that they sit on. We emphasize that except for the elements plotted, no other elements are found to satisfy the cavitation criterion during the entire loading process. For

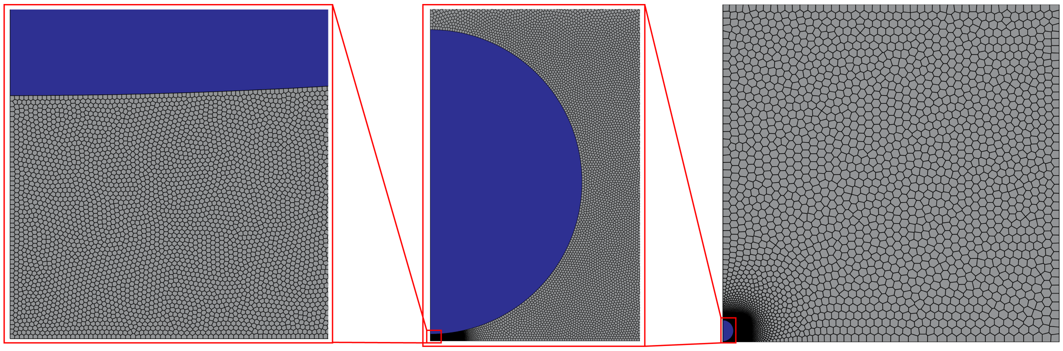


Figure 2.21: The polygonal mesh used of the model of initial distance $d=0.41$mm, which consists of 30000 polygonal elements.

both cases of $d = 0.41$ mm and 0.91 mm, the criterion is first satisfied at the rubber/fiber interfaces. As the macroscopic strain increases, the satisfaction of the criterion quickly propagates to the midpoint between the fibers and then continues to grow outward. For the case of larger interfiber distance, $d = 0.91$ mm, the onset of cavitation at the rubber/fiber interfaces occurs at a significantly larger macroscopic strain, $\overline{\varepsilon} \approx 8.835\%$, as compared to $\overline{\varepsilon} \approx 5.66\%$. Similarly, the onset of cavitation at the midpoint between the fiber also occurs at a considerably larger macroscopic strain, $\overline{\varepsilon} \approx 9.635\%$, for the case of $d = 0.91$ mm, as compared to $\overline{\varepsilon} \approx 5.86\%$ for the case of $d = 0.43$ mm. Qualitatively, these relative values are in fair agreement with the reported experimental data summarized in Table 2.8.

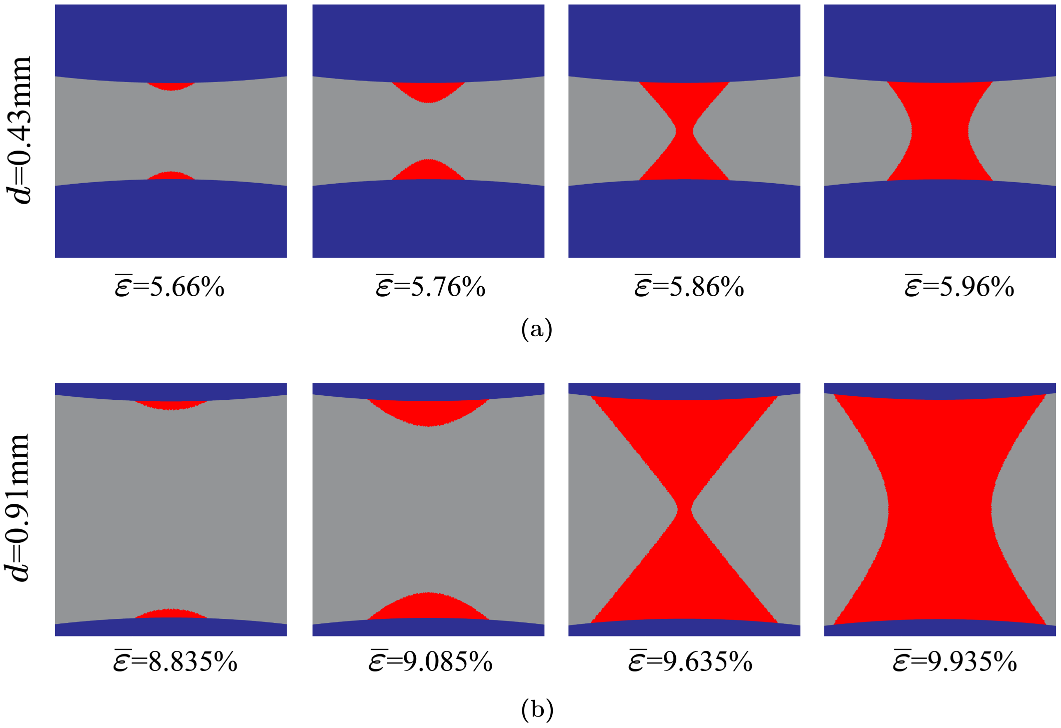
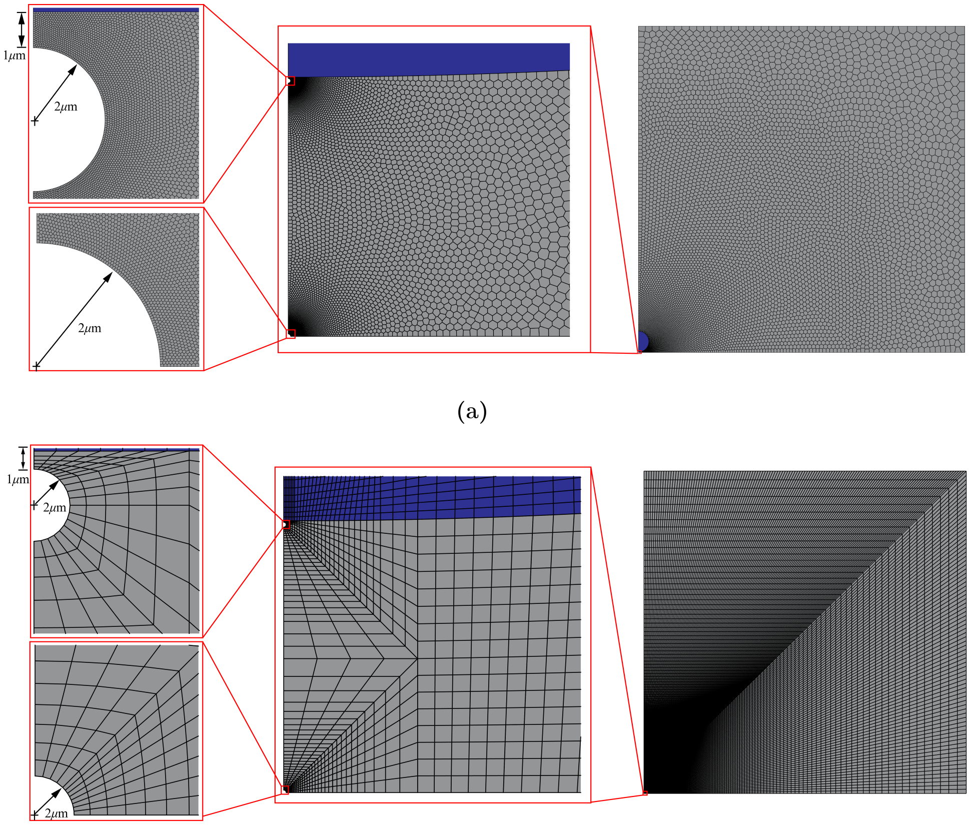

d=0.43mm
$\overline{\varepsilon}$=5.66%
$\overline{\varepsilon}$=5.76%
$\overline{\varepsilon}$=5.86%
$\overline{\varepsilon}$=5.96%
(a)
d=0.91mm
$\overline{\varepsilon}$=8.835%
$\overline{\varepsilon}$=9.085%
$\overline{\varepsilon}$=9.635%
$\overline{\varepsilon}$=9.935%
(b)

Figure 2.22: The deformed configurations of the region between two fibers under four levels of macroscopic strains for cases (a) $d = 0.43$mm and (b) $d = 0.91$mm. The material points at which the cavitation criterion (2.76) are satisfied are plotted in red.

With the knowledge that the onset of cavitation might first occur at the rubber/fiber interfaces on the inner sides of the fibers and then propagates to the midpoint between the fibers, we proceed to introduce pre-existing defects at those locations in the finite element models to study their actual growth and interaction. Thus, three vacuous defects that are circular in shape are introduced in each finite element model with an initial radius $R_c = 2\mu$m. Two of them are placed 1μm away from the inner side of each fiber at the rubber/fiber interfaces while the third one is placed at the midpoint between the fibers. Both linear and quadratic polygonal meshes are considered, containing 25,000 and 10,000 elements respectively. As an example, Fig. 2.23 (a) shows the linear polygonal mesh utilized in the case with the initial distance $d = 0.43$ mm [110]. We also consider structured quadrilateral meshes of ABAQUS CPE4H elements to serve as reference solutions [122], which are structured radially along each defect to accommodate its growth. Figure 2.23 (b) shows the structured quadrilateral mesh utilized in the case with the initial distance $d = 0.43$ mm

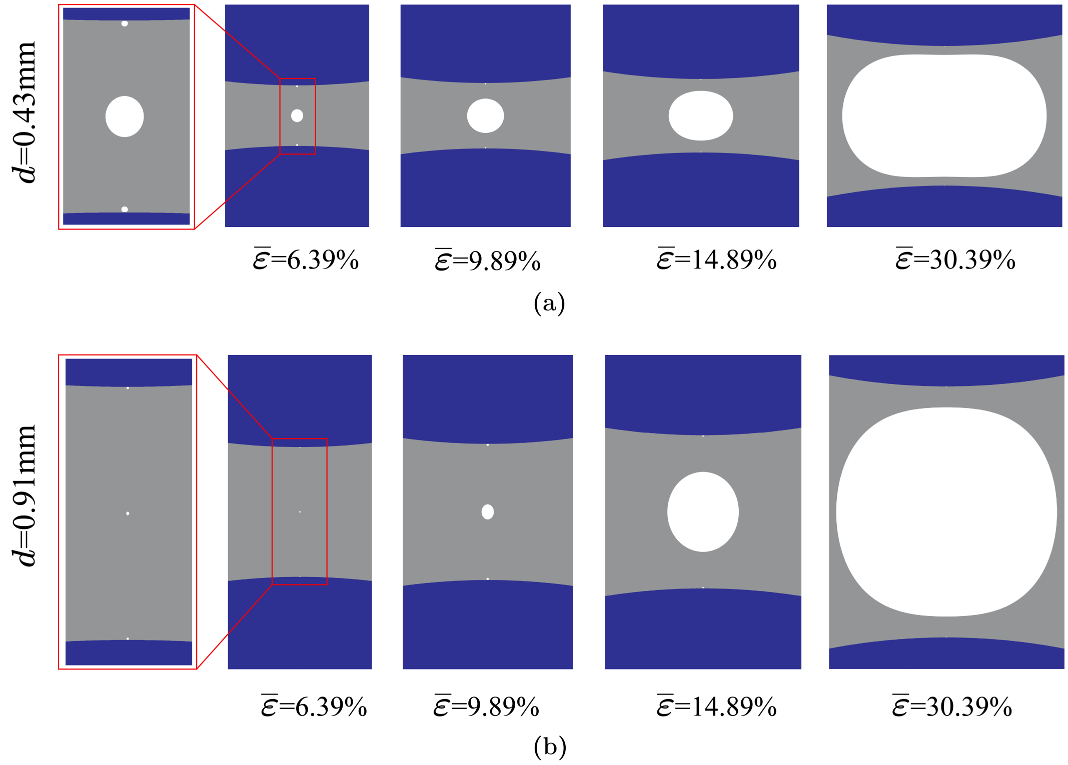


(a)

(b)

Figure 2.23: (a) The linear polygonal mesh used for the model of initial distance $d = 0.41$mm contains three defects. The mesh consists of 25,000 elements. (b) The quadrilateral mesh used of the model of initial distance $d = 0.41$mm, which consists of 40,843 linear quadrilateral elements.

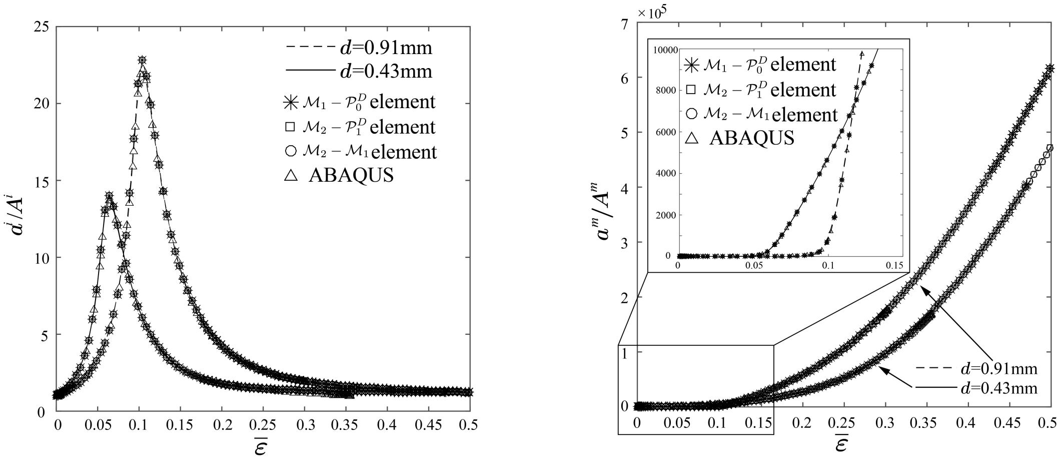
Figures 2.24 (a) and (b) show snapshots of deformed states of the regions between the fibers obtained by $\mathscr{M}_1-\mathscr{P}_0^D$ elements for cases with initial distances $d=0.43$ mm and 0.91 mm, respectively, at four stages of macroscopic strains, $\overline{\varepsilon}=6.39, 9.89, 14.89,$ and 30.39%. The results obtained by mixed quadratic polygonal elements are identical and therefore are not shown here. To gain more quantitative insight, we also quantify the current-to-undeformed area ratios a^i/A^i and a^m/A^m of the defects at the interfaces and midpoints. The results are displayed in Figs. 2.25 (a) and (b) as functions of the applied macroscopic strain $\overline{\varepsilon}$. For comparison purposes, these figures include results computed with a structured quadrilateral mesh of ABAQUS CPE4H elements [122]. An immediate observation is that all the mixed polygonal elements, namely, the $\mathscr{M}_1-\mathscr{P}_0^D$, $\mathscr{M}_2-\mathscr{P}_1^D$, and $\mathscr{M}_2-\mathscr{M}_1$ elements are able to capture the entire growth history and interactions between the defects and agree with the ABAQUS results. Physically, the results indicate that the defects at the interfaces grow indeed first but that, upon further loading, the defects at the midpoint start to grow as well. Further loading results in the contraction of the interfacial defects and the concentration of all of the growth in the midpoint defects, which eventually become visible cavities. This intricate behavior appears to be in qualitative agreement with the experiments of Cho and Gent, at least for the specimen with initial interfiber separation $d=0.43$ mm. The reason behind such a behavior is that the midpoint is the farthest point from the stiff rubber/fiber interface and therefore is the location where the growth of a cavity is the least energetically expensive [122]. Furthermore, if we heuristically define that cavitation occurs whenever a defect grows 1000 times in area say, the specimen with initial fiber distance of $d=0.43$ mm cavitates at a macroscopic strain around $\overline{\varepsilon}\approx 6\%$ while the one with initial fiber distance of $d=0.91$mm cavitates at around $\overline{\varepsilon}\approx 10\%$. These values are in good agreement with the prediction given by the criterion (2.76) as well as with the experimental observations. We emphasize that such an agreement is remarkable given that the computations carried out here account only for the elastic deformation of the defects, whereas in the experiments inelastic growth of defects by fracture does occur.


d=0.43mm
$\overline{\varepsilon}$=6.39%
$\overline{\varepsilon}$=9.89%
$\overline{\varepsilon}$=14.89%
$\overline{\varepsilon}$=30.39%
d=0.91mm
$\overline{\varepsilon}$=6.39%
$\overline{\varepsilon}$=9.89%
$\overline{\varepsilon}$=14.89%
$\overline{\varepsilon}$=30.39%

(a)

(b)

Figure 2.24: With three defects introduced in the model, the deformed configurations of the region between two fibers under four levels of macroscopic strains for cases (a) $d =$ 0.43mm and (b) $d = 0.91$mm.



(a)

(b)

Figure 2.25: The growth histories of the three defects quantified by the ratio of their deformed areas to their initial areas as functions of the applied macroscopic strain ε for both cases of $d = 0.43$ mm and $d = 0.91$ mm: (a) the defects on the matrix/fiber interface, and (b) the defect in the middle of the two fibers.

2.6 Concluding remarks

The modeling of nonlinear elastic materials with standard finite elements under finite deformations is a challenging task due to, by and large, the invariably large local deformations induced by their heterogeneity. Recently, lower order polygonal elements have shown potential in modeling such a class of materials. To further examine the capabilities of polygonal elements to model nonlinear elastic materials, this chapter has generalized existing displacement-based and mixed polygonal elements to higher order and has provided a study of their performance under finite deformations. One critical aspect of the generalization is the numerical integration scheme. Because polygonal elements contain functions of non-polynomial nature in their finite element spaces, they typically suffer from consistent errors with the existing integrations schemes. As shown by numerical tests here, the consistent errors do not vanish under mesh refinement when a standard integration scheme is used, which lead to non-convergent finite element results. This problem becomes even more severe when using higher order elements. To address this issue, this chapter has proposed to employ a gradient correction scheme which can be applied to both displacement-based and mixed polygonal elements of arbitrary order with any numerical integration scheme that satisfy the corresponding minimal order requirement. With the correction scheme, both linear and quadratic polygonal elements have been shown to yield optimally convergent results with displacement-based and mixed approximations. Furthermore, two choices of the pressure field approximations have been discussed for the mixed polygonal elements, namely, the discontinuous approximation and continuous approximation, and their performances on numerical stability and accuracy have been investigated. Our numerical studies indicate that both families of mixed polygonal elements are numerically stable in linear elasticity problems without any additional stabilization treatments up to quadratic order. The comparison of performance in the accuracy (for instance, errors in satisfaction of incompressibility constraint) between the different types of mixed polygonal element, on the other hand, appears to be problem specific (c.f., the discussions in the examples of bending of a rectangular block into a semicircle and particle reinforced elastomer), and thus needs further study. For demonstration purposes, we have employed the proposed polygonal elements to model the nonlinear response of particle reinforced elastomers with interphases and the onset of cavitation in fiber-reinforced elastomers. These challenging physically-based applications have indicated that the proposed polygonal elements are well suited to deal with complex microstructures including particulate microstructures and those involving different length scales, and appears to be more tolerant to large local deformation than the standard finite elements (triangles and quads).

CHAPTER 3
SOME BASIC FORMULATIONS OF THE VIRTUAL ELEMENT METHOD (VEM) FOR FINITE DEFORMATIONS

We present a general Virtual Element Method (VEM) framework for finite elasticity, which emphasizes two issues: element-level volume change (volume average of the determinant of the deformation gradient) and stabilization. To address the former issue, we provide and exact evaluation of the average volume change in both 2D and 3D on properly constructed local displacement spaces. For the later issue, we provide a new stabilization scheme that is based on the trace of the material tangent modulus tensor, which captures highly heterogeneous and localized deformations. Two VEM formulations are presented: a two-field mixed and an equivalent displacement-based, which is free of volumetric locking. Convergence and accuracy of the VEM are verified by means of numerical examples, and engineering applications are demonstrated.

3.1 Introduction

In this chapter, we put forth a Virtual Element Method (VEM) framework for finite elasticity problems. The introduced VEM framework allows for general two-dimensional polygonal and three-dimensional polyhedral meshes. Two VEM formulations are presented: a two-field mixed VEM formulation and an equivalent displacement-based VEM formulation, which is free of volumetric locking. We highlight that both VEM formulations utilize the average volume change (the volume average of the determinant of the deformation gradient) in the element level, which can be computed exactly for both polygonal and polyhedral virtual elements under any given deformation fields. For polyhedral virtual elements, the local displacement space is properly constructed such that the average volume change can be exactly computed. We study various choices of constructing the loading terms and the stabilization terms for the VEM. A new stabilization scheme is further proposed for isotropic materials, which is based on the trace of the material tangent modulus tensor. We conduct thorough numerical assessments to evaluate the convergence of the proposed VEM. Through an application example, the performance of the proposed VEM is demonstrated in a problem that involves heterogeneous and localized deformations.

The remainder of this chapter is organized as follows. Section 2 provides motivations of this chapter and summarizes the related chapter in the VEM literature. Section 3 reviews the displacement-based and two-field mixed variational principles for finite elasticity. Section 4 introduces the constructions of displacement and pressure VEM spaces, together with derivations of the exact average volume changes of virtual elements under given displacement fields. In Section 5, a two-field mixed VEM approximation and its equivalent displacement-based counterpart are presented together with detailed discussions on the construction of loading and stability terms. A new stabilization scheme is also proposed in this section. Several numerical assessments are presented in Section 6 to evaluate the convergence as well as the accuracy of the proposed VEM approximations. In Section 7, a two dimensional application example is presented, which studies the nonlinear elastic response of a filled elastomer, to showcase and evaluate the performance of the VEM formulations in problems involving large heterogeneous and localized deformations. Section 8 contains several concluding remarks and future research directions.

3.2 Motivation and related work

Due to their unique and intriguing properties, polygonal and polyhedral elements have been attracting high level of attention in the computational mechanics community. Despite their relatively short history of development, polygonal and polyhedral finite elements have been successfully applied to several fields of computational mechanics and have revealed several advantages over classical finite elements, i.e., triangular/tetrahedral and quadrilateral/brick elements. For instance, polygonal and polyhedral finite elements have been shown to better capture crack propagation and branching in computational fracture simulations [13]–[16], to produce numerically stable results in topology optimization [17]–[19] and fluid mechanics [4], and to better model contact [20] and arbitrary internal interface [21] in large deformation elasto-plasticity problems. Furthermore, recent studies have demonstrated that polygonal finite elements also possess great potential in the study of two-dimensional finite elasticity problems [2], [22]. More specifically, from a geometrical point of view, polygonal finite elements are well suited to modeling complex microstructures, such as porous or particulate microstructures and microstructures involving different length scales, and to dealing with periodic boundary conditions (introducing hanging nodes). From an analysis point of view, two-field mixed polygonal finite elements are found to be numerically stable on Voronoi-type meshes, to produce more accurate results, and to be more tolerant to large localized deformations.

One the other hand, polygonal and polyhedral finite elements also suffer from drawbacks. First, while the performance of the finite element results strongly depends on the quality of the shape functions, the use of polygonal and polyhedral elements with general shapes, such as concave polyhedrons, are limited by the availability of well-defined shape functions. Particularly in 3D, most of the existing shape functions for polyhedral finite elements are either restrictive in element geometries or computationally expensive to compute. For instance, although closed-form expressions are available, the Wachspress shape functions are restricted to strictly convex and simple polyhedrons (meaning the collection of faces that include each vertex consists of exactly three faces) [30], and the Mean Value coordinates are mainly applicable to polyhedrons with simplicial faces [31]. In contrast, while allowing for more general polyhedrons (e.g., concave ones), harmonic shape functions [29], [32], [33] and max-entropy shape functions [34]–[36] need to be computed numerically at the quadrature points element by element. Moreover, especially for nonlinear and three-dimensional problems, efficient yet consistent numerical quadrature rule on general polygons and polyhedrons are difficult to construct and typically contain numerous quadrature points (see [68], [130], [131] for some recent papers to alleviate this issue), which is also undesirable from a computational perspective.

The Virtual Element Method (VEM) has been recently introduced as a generalization of the finite element method (FEM). The VEM is able to handle general polygonal and polyhedral meshes [37], [38] as it abstracts from the shape functions, which are essential to any FEM approximation. Unlike FEM, the shape functions in VEM are constructed implicitly, and the approximations of the weak forms are decomposed into consistency and stability terms, both of which are directly computed from the degrees of freedoms (DOFs) of the unknown fields. By doing this, only numerical quadratures of polynomials (and not of more complex functions) is needed in VEM. These favorable features make the VEM an attractive framework to efficiently deal with general meshes in 2D and 3D. While most of the studies in the VEM literature consider linear problems in 2D (see, e.g. [25]–[27], [37], [38], [42], [48], [132]–[134]), some attempts have been made for nonlinear [44], [135] and 3D [26] problems. In the context of structural mechanics, VEM has been introduced in [42] for (possibly incompressible) two dimensional linear elasticity, in [26] for three dimensional linear elasticity, in [44] for general two dimensional elastic and inelastic problems under small deformations, and in [52] for simple contact problems.

The goal of this chapter is to put forward a VEM framework for finite elasticity problems, which is capable of handling general polygonal and polyhedral elements, including concave ones, in both 2D and 3D. Two formulations are presented: a two-field mixed VEM formulation, which involves an additional pressure field, and an equivalent displacement-based VEM formulation, which is numerically shown to be free of volumetric locking on a wide class of meshes (as opposed to the standard displacement-based FEM). Both formulations make use of the exact average volume change (the volume average of the determinant of the deformation gradient) of each element, the analytical expressions of which are derived in both 2D and 3D case under given displacement fields. For polyhedral elements in 3D, the analytical expression for this average volume change is obtained through a proper definition of the local displacement VEM space. In both 2D and 3D, the VEM formulations are shown to deliver optimally convergent results, even for meshes with non-convex elements. Moreover, since nonlinear elastic materials are characterized by non-convex stored-energy functions as a result of the large reversible deformation, the stabilization term plays a crucial role in the performance of the VEM. *We study various choices of the stabilization term and discuss their influence on the performance in finite elasticity VEM problems. We also propose a new stabilization scheme for isotropic materials, which is based on the trace of the material tangent modulus tensor*. A two dimensional practical application example is presented, which addresses the nonlinear elastic response of a filled elastomer, and demonstrates the performance proposed VEM formulations in a problem that involves large heterogeneous and localized deformations.

3.3 Theoretical background

This section revisits the variational principles for finite elastostatics [80]. Displacement-based and two-field mixed variational principles are presented. Throughout, Lagrangian description of the fields is adopted.

Consider an elastic solid that occupies a domain $\Omega \in \mathbb{R}^d$ with $\partial\Omega$ being its boundary in its stress-free, undeformed configuration and d being the dimension. It is subjected to a prescribed displacement field $\mathbf{u}^0$ on $\Gamma^{\mathbf{X}}$ and a prescribed surface traction $\mathbf{t}$ (per unit undeformed surface) on $\Gamma^{\mathbf{t}}$, such that $\Gamma^{\mathbf{X}} \cup \Gamma^{\mathbf{t}} = \partial\Omega$ and $\Gamma^{\mathbf{X}} \cap \Gamma^{\mathbf{t}} = \varnothing$ for a well-posed problem. Additionally, we also prescribe a body-force $\mathbf{f}$ (per unit undeformed volume) in the solid. A stored-energy function W is used to characterize the constitutive behavior of the body, which is assumed to be an objective function of the deformation gradient tensor $\mathbf{F}$. Throughout this chapter, we further assume that the stored-energy function is composed of two terms, namely,

$$W(\mathbf{X},\mathbf{F}) = \Psi(\mathbf{X},\mathbf{F}) + U(\mathbf{X},J). \tag{3.1}$$

U is the volumetric part of the stored-energy function that depends only on $J = \det \mathbf{F}$, such that i) U is strictly convex (at least in the neighborhood of $J = 1$); ii) U achieves minimum 0 at $J = 1$, and iii) $U \to +\infty$ as $J \to +\infty$. For instance, in the case of purely incompressible solids, U takes the form

$$U(\mathbf{X}, J) = \begin{cases} 0 & \text{if } J = 1 \\ +\infty & \text{otherwise} \end{cases} . \tag{3.2}$$

Once $\mathbf{U}(\mathbf{X}, J)$ is identified, $\Psi(\mathbf{F})$ is the remaining term of the stored-energy function $W(\mathbf{F})$. The first Piola-Kirchhoff stress tensor $\mathbf{P}$ at each material point $\mathbf{X} \in \Omega$ is given by the following relation:

$$\mathbf{P}(\mathbf{X}) = \frac{\partial W}{\partial \mathbf{F}}(\mathbf{X}, \mathbf{F}) = \frac{\partial \Psi}{\partial \mathbf{F}}(\mathbf{X}, \mathbf{F}) + \frac{\partial U}{\partial \mathbf{F}}(\mathbf{X}, J). \tag{3.3}$$

which is used as the stress measure of choice in this chapter.

3.3.1 Displacement-based variational principle

In the displacement-based formulation, the displacement field $\mathbf{u}$ is taken to be the only independent variable. The deformation gradient $\mathbf{F}$ is then assumed to be dependent on $\mathbf{u}$ through $\mathbf{F}(\mathbf{u}) = \mathbf{I} + \nabla \mathbf{u}$, where ∇ denotes the gradient operator with respect to the undeformed configuration and $\mathbf{I}$ stands for identity in the space of second-order tensors. Similarly, J depends on $\mathbf{u}$ through $J(\mathbf{u}) = \det \mathbf{F}(\mathbf{u})$. Based on the principle of minimum potential energy, the displacement-based formulation consists of seeking the unknown displacement $\mathbf{u}$ which minimize the potential energy Π among all the kinematically admissible displacements $\mathbf{v}$

$$\Pi(\mathbf{u}) = \min_{\mathbf{v} \in \mathscr{K}} \Pi(\mathbf{v}), \tag{3.4}$$

with

$$\Pi(\mathbf{v}) = \int_{\Omega} \Big[\Psi(\mathbf{X}, \mathbf{F}(\mathbf{v})) + U(\mathbf{X}, J(\mathbf{v})) \Big] \mathrm{d}\mathbf{X} - \int_{\Omega} \mathbf{f} \cdot \mathbf{v} \mathrm{d}\mathbf{X} - \int_{\Gamma^{\mathbf{t}}} \mathbf{t} \cdot \mathbf{v} \mathrm{d}S, \tag{3.5}$$

where $\mathscr{K}$ stands for the set of kinematically admissible displacements such that $\mathbf{v} = \mathbf{u}^0$ on $\Gamma^{\mathbf{X}}$.

The weak form of the Euler-Lagrange equations of the minimization problem is given by

$$G(\mathbf{v}, \delta\mathbf{v}) = \int_{\Omega} \left[\frac{\partial \Psi}{\partial \mathbf{F}} (\mathbf{X}, \mathbf{F}(\mathbf{v})) + \frac{\partial U}{\partial \mathbf{F}} (\mathbf{X}, J(\mathbf{v})) \right] : \nabla(\delta\mathbf{v}) \mathrm{d}\mathbf{X} \\ - \int_{\Omega} \mathbf{f} \cdot \delta\mathbf{v} \mathrm{d}\mathbf{X} - \int_{\Gamma^{\mathbf{t}}} \mathbf{t} \cdot \delta\mathbf{v} \mathrm{d}S = 0 \quad \forall \delta\mathbf{v} \in \mathscr{K}^0, \quad (3.6)$$

where the trial displacement field $\delta\mathbf{v}$ is the variation of $\mathbf{v}$, and $\mathscr{K}^0$ denotes the set of all the kinematically admissible displacement fields that vanish on $\Gamma^{\mathbf{X}}$.

3.3.2 A general two-field mixed variational principle

For nearly and purely incompressible materials, the two-field mixed variational principle is typically utilized. Under the assumption of (3.1), different types of two-field mixed variational principles can be derived based on different forms of the decompositions. In the $\mathbf{F}$-formulation, Ψ is assumed to be a general function on $\mathbf{F}$, i.e., no multiplicative decomposition of the deformation gradient is used, whereas, in the commonly used $\overline{\mathbf{F}}$-formulation, Ψ is assumed to solely depend on the deviatoric part of the deformation gradient, that is, $\overline{\mathbf{F}} = J^{-1/3}\mathbf{F}$. In the latter case, Ψ is the stored-energy induced by the deviatoric component of the deformation.

This section briefly reviews the $\mathbf{F}$-formulation proposed in [2]. For the $\overline{\mathbf{F}}$-formulation, the derivation follows similar procedures and therefore is not presented here. The interested readers are referred to [80], [86], [88], [136], [137] and references therein.

The basic idea is to introduce a Legendre transformation of the term $U(J)$ in the stored-energy function as follows [2]

$$\widehat{U}^*(\mathbf{X}, \widehat{q}) = \max_{J} \widehat{q}(J-1) - U(\mathbf{X}, J). \quad (3.7)$$

Since $U(J)$ is assumed to be convex over J, then the duality of the transformation yields

$$U(\mathbf{X},J) = \max_{\widehat{q}} \widehat{q}\,(J-1) - \widehat{U}^{*}(\mathbf{X},\widehat{q}). \tag{3.8}$$

Direct substitution of (3.8) into the displacement-based formulation (3.4) and (3.5) renders the following two-field mixed variational principle, which seeks the unknown displacement field $\mathbf{u}$ and pressure-like field $\widehat{p}$ such that

$$\widehat{\Pi}(\mathbf{u},\widehat{p}) = \min_{\mathbf{v}\in\mathscr{K}} \max_{\widehat{q}\in\mathscr{Q}} \widehat{\Pi}(\mathbf{v},\widehat{q}), \tag{3.9}$$

where $\widehat{\Pi}(\mathbf{v},\widehat{q})$ is of the form

$$\widehat{\Pi}(\mathbf{v},\widehat{q}) = \int_{\Omega} \left\{ \Psi(\mathbf{X},\mathbf{F}(\mathbf{v})) + \widehat{q}[J(\mathbf{v})-1] - \widehat{U}^{*}(\mathbf{X},\widehat{q}) \right\} \mathrm{d}\mathbf{X} - \int_{\Omega} \mathbf{f}\cdot\mathbf{v}\mathrm{d}\mathbf{X} - \int_{\Gamma^{\mathbf{t}}} \mathbf{t}\cdot\mathbf{v}\mathrm{d}S, \tag{3.10}$$

and $\mathscr{Q}$ denotes the set of square-integrable functions. Unlike the commonly adopted $\overline{\mathbf{F}}$-formulation, whose unknown pressure field is the equilibrium Cauchy hydrostatic pressure field $p \doteq \mathrm{tr}\boldsymbol{\sigma}$, the additional unknown field $\widehat{p}$ in the $\mathbf{F}$-formulation is found to be a pressure-like scalar field which relates to p through

$$p = \widehat{p} + \frac{1}{3J}\frac{\partial\Psi}{\partial\mathbf{F}}(\mathbf{X},\mathbf{F}) : \mathbf{F}. \tag{3.11}$$

The weak forms of the Euler-Lagrange equations from (3.9) and (3.10) are obtained as

$$\begin{aligned} G^{\mathbf{v}}(\mathbf{v},\widehat{q},\delta\mathbf{v}) = \int_{\Omega} \left[\frac{\partial\Psi}{\partial\mathbf{F}}(\mathbf{X},\mathbf{F}(\mathbf{v})) + \widehat{q}\frac{\partial J}{\partial\mathbf{F}}(\mathbf{F}(\mathbf{v})) \right] &: \nabla(\delta\mathbf{v})\mathrm{d}\mathbf{X} \\ - \int_{\Omega} \mathbf{f}\cdot\delta\mathbf{v}\mathrm{d}\mathbf{X} - \int_{\Gamma^{\mathbf{t}}} \mathbf{t}\cdot\delta\mathbf{v}\mathrm{d}S &= 0 \quad \forall\delta\mathbf{v}\in\mathscr{K}^{0}, \end{aligned} \tag{3.12}$$

$$G^{\widehat{q}}(\mathbf{v},\widehat{q},\delta\widehat{q}) = \int_{\Omega} \left[J(\mathbf{v}) - 1 - \frac{\partial\widehat{U}^{*}}{\partial\widehat{q}}(\mathbf{X},\widehat{q}) \right] \delta\widehat{q}\mathrm{d}\mathbf{X} = 0 \quad \forall\delta\widehat{q}\in\mathscr{Q}, \tag{3.13}$$

where the trial pressure field $\delta\widehat{q}$ is the variation of $\widehat{q}$.

3.4 Virtual element spaces and projection operators

Consider Ω_h to be a tessellation of the domain Ω into non-overlapping polygon/polyhedral elements with h being the maximum element size. The boundary of the mesh, denoted as Γ_h is assumed to be compatible with the applied boundary condition, that is, $\Gamma_h^{\mathbf{t}}$ and $\Gamma_h^{\mathbf{X}}$ are both unions of edges/faces of the mesh. In the following presentation, we use F to denote a generic polygonal (an element in 2D or a face in 3D) and P to denote a generic polyhedron. Moreover, we use e to represent a generic edge. When the definitions are independent of dimensions, we denote E as a generic element with $|E|$ being its area/volume.

3.4.1 Displacement VEM space and projection operators

The global virtual displacement space $\mathscr{K}_h$ associated with the mesh Ω_h is a conforming finite dimensional space that defined as:

$$\mathscr{K}_h = \{\mathbf{v}_h \in \mathscr{K} : \mathbf{v}_h|_E \in \mathscr{V}(E), \forall E \in \Omega_h\}, \tag{3.14}$$

where $\mathscr{V}(E)$ is a local VEM space defined on $E \in \Omega_h$.

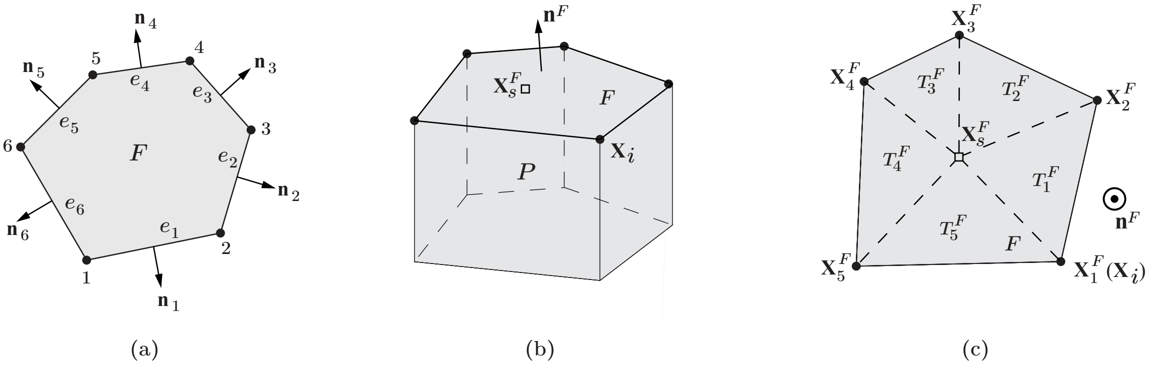


Figure 3.1: Illustrations of (a) a two dimensional polygonal element F in its undeformed configuration, (b) a three dimensional polyhedral element P in its undeformed configuration, and (c) a generic face F of the polyhedral element P.

Local VEM space in 2D In two dimensions (2D), let us consider a given element F with m vertices, denoted as $\mathbf{X}_i = [X_i, Y_i]^T$, $i = 1, ..., m$, which are numbered counterclockwise as shown in Fig. 3.1 (a). We define the local displacement VEM space as [37], [42]

$$\mathscr{V}(F) \doteq \left\{ \mathbf{v} \in \left[H^1(F)\right]^2 : \Delta\mathbf{v} = 0 \text{ in } F, \mathbf{v}|_e \in [\mathscr{P}_1(e)]^2 \, \forall e \in \partial F \right\}, \tag{3.15}$$

where Δ stands for the Laplacian operator. As inferred by its definition, the VEM space contains harmonic functions which are implicitly known in the interior of F and explicitly known to possess linear variations on the edges of F. We note that enforcing the functions within the local VEM space to be harmonic is a simple choice instead of a requirement. Other choices can also be made. For instance, another option is to enforce the functions to have linearly varying Laplacian together with additional constraints to fix the total number of DOFs. An example of under this philosophy can be found in (4.9). One can also use other types of PDEs than the Laplace equation. For example, the Stokes equation is used in [138], [139] to construct the local VEM in order solve the Stokes or Navier-Stokes flow problems. By using the Stokes equation, one can enforce point-wise divergence-free property of the local VEM space, which is shown to produce solutions with improved accuracy and robustness for Stokes and Navier-Stokes flow problems [138], [139]. Moreover, the VEM space $\mathscr{V}(F)$ is linearly complete, namely, $[\mathscr{P}_1(F)]^2 \subseteq \mathscr{V}(F)$. In order to guarantee the continuity of the global displacement space $\mathscr{K}_h$, the degrees of freedom (DOFs) of the $\mathscr{V}(F)$ are taken at the vertices of F, which are identical to those in the first-order finite element spaces on general polygons.

By construction, we are able to compute the area average of the gradient of any given function $\mathbf{v} \in \mathscr{V}(F)$ using only the DOFs of $\mathbf{v}$ and geometrical information of F. Indeed, using integration by parts, for any given function $\mathbf{v}$ in $\mathscr{V}(F)$ it holds:

$$\frac{1}{|F|} \int_F \nabla \mathbf{v} \mathrm{d}\mathbf{X} = \frac{1}{|F|} \sum_{e \in \partial F} \int_e \mathbf{v} \otimes \mathbf{n}_e \mathrm{d}S, \tag{3.16}$$

where $\mathbf{n}_e$ is the outward norm vector associated with edge e. Realizing that the functions in $\mathscr{V}(F)$ possess linear variations on each edge of F, the edge integral on the right-hand side of the above relation can be expressed as

$$\sum_{e\in\partial F}\int_e \mathbf{v}\otimes\mathbf{n}_e \mathrm{d}S = \frac{1}{2}\sum_{i=1}^{m}\mathbf{v}(\mathbf{X}_i)\cdot(|e_i|\mathbf{n}_i+|e_{i-1}|\mathbf{n}_{i-1}), \tag{3.17}$$

yielding

$$\frac{1}{|F|}\int_F \nabla\mathbf{v}\mathrm{d}\mathbf{X} = \frac{1}{2|F|}\sum_{i=1}^{m}\mathbf{v}(\mathbf{X}_i)\cdot(|e_i|\mathbf{n}_i+|e_{i-1}|\mathbf{n}_{i-1}). \tag{3.18}$$

where $|e_i|$ is the length of the ith edge, as shown in Fig. 3.1 (a). For convenience, in the sums we follow the standard convention that $i+1=1$ whenever $i=m$ and $i-1=m$ whenever $i=1$.

Local VEM space in 3D We proceed to define the displacement VEM space in 3D. To begin with, let us consider a polyhedron P whose boundary consists of planar faces as shown in Fig. 3.1 (b). Suppose that P contains m vertices, which are numbered as $\mathbf{X}_i=[X_i,Y_i,Z_i]^T$, $i=1,...,m$. For the ith vertex $\mathbf{X}_i$, we denote $\mathscr{F}_i$ as the set of faces that are connected to it. Moreover, for any given face $F\in\partial P$ with m^F vertices, we assume that the vertices are renumbered locally as $\mathbf{X}^F_j$, $j=1,...,m^F$, such that the numbering is in a counterclockwise fashion with respect to the outward normal $\mathbf{n}^F$ pointing out of the element, as shown in Fig. 3.1 (c). A map $\mathscr{G}_F$ is utilized to denote the relation between the global numbering and the local numbering on face F. If the ith vertex of P (with global numbering) becomes the jth vertex of F (with local numbering), we write $\mathbf{X}^F_j=\mathbf{X}^F_{\mathscr{G}_F(i)}$ or $\mathbf{X}_i=\mathbf{X}_{\mathscr{G}_F^{-1}(j)}$. Notice that for any given $F\notin\mathscr{F}_i$, $\mathscr{G}_F(i)=\emptyset$. Additionally, any given face $F\in\partial P$ is assumed to be star-convex [29] with respect to a point $\mathbf{X}^F_s$ given by

$$\mathbf{X}^F_s=\sum_{j=1}^{m^F}\beta^f_j\mathbf{X}^F_j, \tag{3.19}$$

where the weights β_j^F are chosen weights that satisfy $\beta_j^F \geq 0$, $j = 1, ..., m^F$ and $\sum_{j=1}^{m^F} \beta_j^F = 1$. It follows that for any linear function $\mathbf{p} \in [\mathscr{P}_1(\mathbb{R})]^3$,

$$\mathbf{p}\left(\mathbf{X}_s^F\right) = \sum_{j=1}^{m^F} \beta_j^F \mathbf{p}\left(\mathbf{X}_j^F\right). \tag{3.20}$$

By assuming star-convexity of each face, we can introduce triangulations of the all the faces $F \in \partial P$ by connecting all the vertices to $\mathbf{X}_s^F$, as shown in Fig. 3.1 (c). We use T_j^F, $j = 1, ..., m^F$ to denote the triangulated subfaces of F, such that T_j^F and T_{j-1}^F are connected to vertex $\mathbf{X}_j^F$, as shown in Fig. 3.1 (c).

The virtual space on P is defined as

$$\begin{aligned} \mathscr{V}(P) \doteq \Bigg\{ \mathbf{v} \in \left[H^1(P)\right]^3 : \mathbf{v}|_{\partial P} \in \left[C^0(\partial P)\right]^3, \mathbf{v}\left(\mathbf{X}_s^F\right) = \sum_{j=1}^{m^F} \beta_j^F \mathbf{v}\left(\mathbf{X}_j^F\right) \quad \text{and} \\ \mathbf{v}|_{T_j^F} \in \left[\mathscr{P}_1\left(T_j^F\right)\right]^3, \quad j = 1, ..., m^F, \forall F \in \partial P \quad \text{and} \quad \Delta \mathbf{v} = 0, \, \text{in} \, P \Bigg\}. \end{aligned} \tag{3.21}$$

By the above definition, the above space contains harmonic functions which are continuous and piecewise linear on each face of the element P. Moreover, due to (3.20), we have $[\mathscr{P}_1(P)]^3 \subseteq \mathscr{V}(P)$. As for the degrees of freedom for $\mathscr{V}(P)$, we take the pointwise values at the vertices of P; these are a valid set of degrees of freedom for the local VEM space.

Under the definition above, the volume average of the gradient for any function $\mathbf{v}$ in $\mathscr{V}(P)$ can be computed exactly taking the steps from the 3D analog of Eq. (3.16):

$$\frac{1}{|P|} \int_P \nabla \mathbf{v} \mathrm{d}\mathbf{X} = \frac{1}{|P|} \sum_{F \in \partial P} \int_F \mathbf{v} \otimes \mathbf{n}^F \mathrm{d}S. \tag{3.22}$$

Notice that since F is assumed to be planar, its outward unit normal vector $\mathbf{n}^F$ is a constant vector. Upon realizing that $\mathbf{v}$ is piecewise linear on F, the face integral $\int_F \mathbf{v} \otimes \mathbf{n}^F \mathrm{d}S$ can be explicitly carried out by using a vertex-based quadrature rule over each triangulated face:

$$\int_F \mathbf{v}\otimes\mathbf{n}^F \mathrm{d}S = \sum_{j=1}^{m^F}\left\{\frac{|T_j^F|}{3}\left[\mathbf{v}\left(\mathbf{X}_j^F\right)+\mathbf{v}\left(\mathbf{X}_{j+1}^F\right)+\mathbf{v}\left(\mathbf{X}_s^F\right)\right]\right\}\otimes\mathbf{n}^F, \tag{3.23}$$

where $|T_j^F|$ is the (signed) area of T_j^F defined as

$$|T_j^F| = \frac{1}{2}\mathbf{n}^F\cdot\left(\mathbf{X}_j^F-\mathbf{X}_s^F\right)\wedge\left(\mathbf{X}_{j+1}^F-\mathbf{X}_s^F\right). \tag{3.24}$$

Again, analogous summing conventions as in (3.17)-(3.18) are used. The above expression (3.24) can be further recast into a vertex based quadrature rule over F as follows

$$\int_F \mathbf{v}\otimes\mathbf{n}^F \mathrm{d}S = \sum_{j=1}^{m^F}\left\{\left[\frac{\beta_j^F}{3}|F|+\frac{1}{3}\left(|T_j^F|+|T_{j-1}^F|\right)\right]\mathbf{v}\left(\mathbf{X}_j^F\right)\right\}\otimes\mathbf{n}^F = \sum_{j=1}^{m^F} w_j^F\mathbf{v}\left(\mathbf{X}_j^F\right)\otimes\mathbf{n}^F \tag{3.25}$$

by assigning each vertex $\mathbf{X}_j^F$ with a weight w_j^F of the form $w_j^F \doteq \frac{\beta_j^F}{3}|F|+\frac{1}{3}\left(|T_j^F|+|T_{j-1}^F|\right)$, where $|F|$ is the (absolute) area of F. Several remarks are worthwhile making on the above vertex based quadrature rule. First, notice that with the assumption that each F is star-convex with respect to $\mathbf{X}_s^F$, the weight w_j^F associated with each vertex is guaranteed to be strictly positive. On the other hand, for cases where F is not star-convex with respect to $\mathbf{X}_s^F$ (including $\mathbf{X}_s^F$ lies outside of F), the weights w_j^F may take negative values. However, we observe that the resulting vertex based quadrature rule on F still enjoy first-order accuracy, as will be discussed in detail in Appendix B. Note that, in such non star convex case, definition (3.21) makes no sense anymore; anyway (thanks to the above observations) it turns out that one can still implement the above formulas and obtain a convergent scheme.

As a result, the volume average of $\nabla\mathbf{v}$ takes the final form

$$\frac{1}{|P|}\int_P \nabla\mathbf{v}\mathrm{d}\mathbf{X} = \frac{1}{|P|}\sum_{F\in\partial P}\sum_{j=1}^{m^F}\left\{w_j^F\mathbf{v}\left(\mathbf{X}_j^F\right)\otimes\mathbf{n}^F\right\}. \tag{3.26}$$

Observe that, as in the 2D case, the above expression only makes use of the DOFs of $\mathbf{v}$ and the geometric information of P.

In the proposed VEM approximation, we make use of two projection operators [44]. The first projection operator, denoted as $\mathbf{\Pi}_E^0$, is a tensor-valued L^2 projection operator which projects any given second-order tensor $\mathbf{G} \in [L^2(E)]^{d\times d}$ onto its average value over E, that is,

$$\mathbf{\Pi}_E^0 \mathbf{G} = \frac{1}{|E|}\int_E \mathbf{G} \mathrm{d}\mathbf{X}. \tag{3.27}$$

We remark that for any given element $E \in \Omega_h$ and $\mathbf{v} \in \mathscr{V}(E)$, the projection $\mathbf{\Pi}_E^0(\nabla \mathbf{v})$ is explicitly computable in both 2D and 3D using the DOFs of $\mathbf{v}$ and geometric information of E, which is a direct consequence of (3.16) and (3.26).

On the other hand, the second projection operator, denoted as $\mathbf{\Pi}_E^\nabla$, is a vector-valued projection operator, which projects from the VEM space $\mathscr{V}(E)$ onto $[\mathscr{P}_1(E)]^d$. It is defined such that

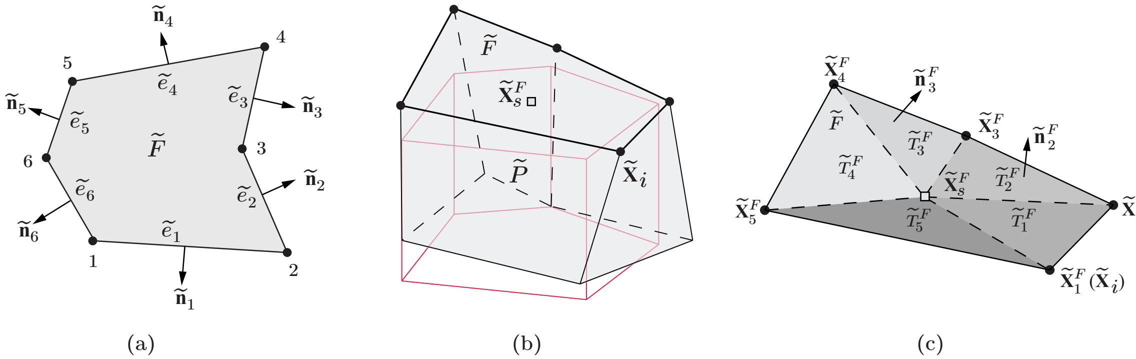
$$\begin{cases} \nabla\left(\mathbf{\Pi}_E^\nabla \mathbf{v}\right) = \mathbf{\Pi}_E^0\left(\nabla \mathbf{v}\right) \\ \sum_{i=1}^m \left(\mathbf{\Pi}_E^\nabla \mathbf{v}\right)(\mathbf{X}_i) = \sum_{i=1}^m \mathbf{v}\left(\mathbf{X}_i\right) \end{cases}. \tag{3.28}$$

As we can see, for any given $\mathbf{v} \in \mathscr{V}(E)$, $\mathbf{\Pi}^\nabla \mathbf{v}$ becomes a linear function. The first condition in the above definition ensures that the gradient of the linear function equals to the average gradient of $\mathbf{v}$ over E and the second condition determines the constant component of the linear function by ensuring that the average value $\mathbf{\Pi}_E^\nabla \mathbf{v}$ of vertices are equal to the average value of $\mathbf{v}$ at vertices. With the two aforementioned conditions, the linear function $\mathbf{\Pi}_E^\nabla \mathbf{v}$ is uniquely determined and depends only on the DOF values of $\mathscr{V}(E)$ and on geometric information of E.

We close this section with an important observation. Our choice (3.21) for the definition of $\mathscr{V}(E)$ on faces is different from the ones adopted in Chapters 4 and 5, as well as the ones proposed in [26], [44], [140]. The reason is that, differently from those space definitions, a displacement in space (3.21) guarantees that the faces of the deformed element are piecewise planar. Our choice is therefore more suitable for large deformation problems and the associated computations.

3.4.2 Area/volume average of J on general polygonal and polyhedral elements

For a given displacement field $\mathbf{v} \in \mathscr{V}(E)$, let us denote $\widetilde{E}$ as the deformed configuration of element E and $\widetilde{\mathbf{X}}_i$ (where $\widetilde{\mathbf{X}}_i = [\widetilde{X}_i, \widetilde{Y}_i]^T$ in 2D or $\widetilde{\mathbf{X}}_i = [\widetilde{X}_i, \widetilde{Y}_i, \widetilde{Z}_i]^T$ in 3D) as the position vector of vertex i of $\widetilde{E}$ such that $\widetilde{\mathbf{X}}_i = \mathbf{X}_i + \mathbf{v}(\mathbf{X}_i)$, as shown in Figs. 3.2 (a) and (b). Similarly, we also utilize $\widetilde{e}$, $\widetilde{F}$, $\widetilde{P}$ to denote the deformed edge, deformed polygon/face and deformed polyhedron, respectively. In addition, we denote by $\widetilde{\mathscr{F}^i}$ the set of deformed faces that are connected to the ith vertex of $\widetilde{P}$. Notice that the faces of a deformed polyhedron may not be planar, but the deformation changes neither the vertex-face connectivity nor the relation between the global and local numbering of the vertices that polyhedron. In other words, if a given F belongs to $\mathscr{F}_i$ in the undeformed configuration, we have $\widetilde{F} \in \widetilde{\mathscr{F}_i}$. Additionally, if the ith vertex of P is renumbered as the jth vertex on face F in the undeformed configuration, the relations $\widetilde{\mathbf{X}}_j^F = \widetilde{\mathbf{X}}^F_{\mathscr{G}_F(i)}$ and $\widetilde{\mathbf{X}}_i = \widetilde{\mathbf{X}}_{\mathscr{G}_F^{-1}(j)}$ still hold in the deformed configuration.


$\widetilde{\mathbf{n}}_4$
4
5
$\widetilde{e}_4$
$\widetilde{e}_3$
$\widetilde{\mathbf{n}}_5$
$\widetilde{\mathbf{n}}_3$
$\widetilde{e}_5$
$\widetilde{F}$
3
6
$\widetilde{\mathbf{n}}_2$
$\widetilde{e}_6$
$\widetilde{e}_2$
$\widetilde{\mathbf{n}}_6$
$\widetilde{e}_1$
1
2
$\widetilde{\mathbf{n}}_1$
$\widetilde{F}$
$\widetilde{\mathbf{X}}_s^F$
$\widetilde{P}$
$\widetilde{\mathbf{X}}_i$
$\widetilde{\mathbf{X}}_4^F$
$\widetilde{\mathbf{n}}_3^F$
$\widetilde{F}$
$\widetilde{T}_3^F$
$\widetilde{\mathbf{X}}_3^F$
$\widetilde{\mathbf{n}}_2^F$
$\widetilde{T}_4^F$
$\widetilde{\mathbf{X}}_s^F$
$\widetilde{T}_2^F$
$\widetilde{\mathbf{X}}_5^F$
$\widetilde{T}_5^F$
$\widetilde{T}_1^F$
$\widetilde{\mathbf{X}}_2^F$
$\widetilde{\mathbf{X}}_1^F\,(\widetilde{\mathbf{X}}_i)$

(a) (b) (c)

Figure 3.2: Illustrations of (a) a 2D polygonal element $\widetilde{F}$ in its deformed configuration, (b) a 3D polyhedral element $\widetilde{P}$ in its deformed configuration, and (c) a generic face $\widetilde{F}$ (not necessarily planar) of the polyhedral element $\widetilde{P}$.

The area/volume average of J over element E under $\mathbf{v}$, denoted as J_E henceforth, is defined as

$$J_E(\mathbf{v}) = \frac{1}{|E|} \int_E J(\mathbf{v})\, \mathrm{d}\mathbf{X}. \tag{3.29}$$

Pushing the integral forward into the deformed configuration and applying the divergence theorem, we end up with the following expression

$$J_E\left(\mathbf{v}\right) = \frac{1}{|E|}\int_{\widetilde{E}} \mathrm{d}\widetilde{\mathbf{X}} = \frac{1}{d|E|}\int_{\widetilde{E}} \widetilde{\nabla}\cdot\widetilde{\mathbf{X}}\mathrm{d}\widetilde{\mathbf{X}} = \frac{1}{d|E|}\int_{\partial\widetilde{E}} \widetilde{\mathbf{X}}\cdot\widetilde{\mathbf{n}}\mathrm{d}\widetilde{S}, \tag{3.30}$$

where $\widetilde{\nabla}$ stands for the gradient operator with respect to the deformed configuration and $\widetilde{\mathbf{n}}$ is the unit normal vector in the deformed configuration.

Computing J_E in the 2D case In the 2D case, because any displacement field $\mathbf{v} = \{v_x, v_y\}^T \in \mathscr{V}(F)$ varies linearly on each edge e, then $\widetilde{e}$ remains straight in the deformed configuration, as shown in Fig. 3.2 (a). Thus, we can expand (3.30) as

$$J_E(\mathbf{v}) = \frac{1}{2|F|}\int_{\partial\widetilde{F}} \widetilde{\mathbf{X}}\cdot\widetilde{\mathbf{n}}\mathrm{d}\widetilde{S} = \frac{1}{4|F|}\sum_{i=1}^{m}\widetilde{\mathbf{X}}_i\cdot\left(|\widetilde{e}_i|\widetilde{\mathbf{n}}_i + |\widetilde{e}_{i-1}|\widetilde{\mathbf{n}}_{i-1}\right), \tag{3.31}$$

where $|\widetilde{e}_i|$ and $\widetilde{\mathbf{n}}_i$ denote the length and outward unit normal vector of the ith deformed edge $\widetilde{e}_i$.

By introducing a matrix $\mathbf{R} \in \mathbb{R}^{2\times 2}$ representing a 90° counterclockwise rotation, which is of the form

$$\mathbf{R} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \tag{3.32}$$

and satisfies $|\widetilde{e}_i|\widetilde{\mathbf{n}}_i = \mathbf{R}(\widetilde{\mathbf{X}}_i - \widetilde{\mathbf{X}}_{i+1})$, we simplify the above relation as

$$J_E(\mathbf{v}) = \frac{1}{4|F|}\sum_{i=1}^{m}\widetilde{\mathbf{X}}_i\cdot\mathbf{R}\left(\widetilde{\mathbf{X}}_{i-1} - \widetilde{\mathbf{X}}_{i+1}\right). \tag{3.33}$$

By direct derivation of (3.33), the first variation of J_E with respect to $\delta\mathbf{v} = [\delta v_x, \delta v_y]^T \in \mathscr{V}(F)$ can be obtained as

$$DJ_E\left(\mathbf{v}\right)\cdot\delta\mathbf{v} = \frac{1}{2|F|}\sum_{i=1}^{m}\delta\mathbf{v}\left(\mathbf{X}_i\right)\cdot\mathbf{R}\left(\widetilde{\mathbf{X}}_{i-1} - \widetilde{\mathbf{X}}_{i+1}\right). \tag{3.34}$$

For practical computations, it is convenient to recast the above expressions into matrix representation. By introducing $\mathbf{H} \in \mathbb{R}^{2m\times 2m}$ of the form

$$\mathbf{H} = \begin{bmatrix} \mathbf{0} & -\mathbf{R} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{R} \\ \mathbf{R} & \mathbf{0} & -\mathbf{R} & \cdots & \mathbf{0} & \mathbf{0} \\ & & \vdots & & & \\ -\mathbf{R} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{R} & \mathbf{0} \end{bmatrix}, \tag{3.35}$$

which is a symmetric matrix (since $\mathbf{R}$ is skew-symmetric), we can rewrite Eqs. (3.33) and (3.34) as

$$J_E(\mathbf{v}) = \frac{1}{4|F|}\widetilde{\mathbf{N}}^T\mathbf{H}\widetilde{\mathbf{N}} \quad \text{and} \quad DJ_E(\mathbf{v})\cdot\delta\mathbf{v} = \frac{1}{2|F|}\delta\mathbf{V}^T\mathbf{H}\widetilde{\mathbf{N}}, \tag{3.36}$$

where $\widetilde{\mathbf{N}}$ and $\delta\mathbf{V} \in \mathbb{R}^{2m}$ are given by

$$\begin{aligned} \widetilde{\mathbf{N}} &= \begin{bmatrix} \widetilde{X}_1 & \widetilde{Y}_1 & \cdots & \widetilde{X}_m & \widetilde{Y}_m \end{bmatrix}^T \text{ and} \\ \delta\mathbf{V} &= \begin{bmatrix} \delta v_x(\mathbf{X}_1) & \delta v_y(\mathbf{X}_1) & \cdots & \delta v_x(\mathbf{X}_m) & \delta v_y(\mathbf{X}_m) \end{bmatrix}^T, \end{aligned} \tag{3.37}$$

respectively.

The second variation of J_E can be obtained directly in matrix representation by a plain derivation of (3.36) and recalling the relation between $\widetilde{\mathbf{N}}$ and $\mathbf{v}$. We obtain the following result

$$D\left(DJ_E(\mathbf{v})\cdot\delta\mathbf{v}\right)\cdot\mathbf{w} = \frac{1}{|F|}\int_F \left(\nabla(\delta\mathbf{v}) : \frac{\partial^2 J}{\partial\mathbf{F}\partial\mathbf{F}}(\mathbf{v})\right) : \nabla\mathbf{w}\mathrm{d}\mathbf{X} = \frac{1}{2|F|}\delta\mathbf{V}^T\mathbf{H}\mathbf{W}, \tag{3.38}$$

where $\mathbf{w} = [w_x, w_y]^T \in \mathscr{V}(F)$ is the incremental displacement field and $\mathbf{W} \in \mathbb{R}^{2m}$ is a vector of the form

$$\mathbf{W} = \begin{bmatrix} w_x(\mathbf{X}_1) & w_y(\mathbf{X}_1) & \cdots & w_x(\mathbf{X}_m) & w_y(\mathbf{X}_m) \end{bmatrix}^T. \tag{3.39}$$

Computing J_E in the 3D case The calculation of J_E in the 3D case is more complicated because the planar face F typically deforms into a non-planar $\widetilde{F}$ in its deformed configuration, as shown in Fig. 3.2 (c). As a result, the associated deformed normal vector $\widetilde{\mathbf{n}}_F$ become non-constant over $\widetilde{F}$. Yet, each deformed triangulated subface $\widetilde{T}_i^F$ of the T_i^F still remains planar. If we denote the deformed normal of $\widetilde{T}_i^F$ as $\widetilde{\mathbf{n}}_i^F$ and employ a vertex-based integration rule, we have

$$\begin{aligned} J_E(\mathbf{v}) &= \frac{1}{3|P|}\int_{\partial\widetilde{P}} \widetilde{\mathbf{X}}\cdot\widetilde{\mathbf{n}}\mathrm{d}\widetilde{S} = \frac{1}{3|P|}\sum_{\widetilde{F}\in\partial\widetilde{P}}\sum_{j=1}^{m^F}\left[|\widetilde{T}_j^F|\widetilde{\mathbf{n}}_j^F\cdot\frac{1}{3}\left(\widetilde{\mathbf{X}}_j^F+\widetilde{\mathbf{X}}_{j+1}^F+\widetilde{\mathbf{X}}_s^F\right)\right] \\ &= \frac{1}{3|P|}\sum_{\widetilde{F}\in\partial\widetilde{P}}\sum_{j=1}^{m^F}\left\{\frac{1}{6}\left(\widetilde{\mathbf{X}}_j^F+\widetilde{\mathbf{X}}_{j+1}^F+\widetilde{\mathbf{X}}_s^F\right)\cdot\left(\widetilde{\mathbf{X}}_j^F-\widetilde{\mathbf{X}}_s^F\right)\wedge\left(\widetilde{\mathbf{X}}_{j+1}^F-\widetilde{\mathbf{X}}_s^F\right)\right\} \end{aligned}$$

Further simplification of the above equation yields, after some calculations,

$$J_E(\mathbf{v}) = \frac{1}{6|P|}\sum_{\widetilde{F}\in\partial\widetilde{P}}\sum_{j=1}^{m^F}\left\{\widetilde{\mathbf{X}}_s^F\cdot\widetilde{\mathbf{X}}_j^F\wedge\widetilde{\mathbf{X}}_{j+1}^F\right\}. \tag{3.40}$$

Notice that, since J_E is an algebraic function of the DOFs of $\mathbf{v}$, then its first variation with respect to any given $\delta\mathbf{v}\in\mathscr{V}(P)$ is obtained as

$$\begin{aligned} DJ_E(\mathbf{v})\cdot\delta\mathbf{v} = \frac{1}{6|P|}\sum_{i=1}^{m}\Bigg\{\delta\mathbf{v}(\mathbf{X}_i)\cdot\Bigg[\sum_{\widetilde{F}\in\widetilde{\mathscr{F}}^i}\Bigg(\beta_i^F\sum_{j=1}^{m^F F}\widetilde{\mathbf{X}}_{j-1}^F\wedge\widetilde{\mathbf{X}}_j^F \\ +\widetilde{\mathbf{X}}_s^F\wedge\left(\widetilde{\mathbf{X}}_{\mathscr{G}_F(i)-1}^F-\widetilde{\mathbf{X}}_{\mathscr{G}_F(i)+1}^F\right)\Bigg)\Bigg]\Bigg\}. \end{aligned} \tag{3.41}$$

Following the same procedure above, the second variation of J_E is given by

$$D\left(DJ_E(\mathbf{v})\cdot\delta\mathbf{v}\right)\cdot\mathbf{w} = \sum_{i=1}^{m}\sum_{j=1}^{m}\delta\mathbf{v}(\mathbf{X}_i)\cdot\frac{\partial^2 J_E}{\partial\widetilde{\mathbf{X}}_i\partial\widetilde{\mathbf{X}}_j}(\mathbf{v})\cdot\mathbf{w}\left(\mathbf{X}_j\right), \tag{3.42}$$

where the 3×3 matrixes $\partial^2 J_E / \partial \widetilde{\mathbf{X}}_i \partial \widetilde{\mathbf{X}}_j$ can be expressed in the form

$$\frac{\partial^2 J_E}{\partial \widetilde{\mathbf{X}}_i \partial \widetilde{\mathbf{X}}_j}(\mathbf{v}) = \frac{1}{6|P|} \sum_{\widetilde{F} \in \widetilde{\mathscr{F}}_{ij}} \Bigg\{ \beta_i^F \left[\widetilde{\mathbf{X}}^F_{\mathscr{G}_F(j)-1} - \widetilde{\mathbf{X}}^F_{\mathscr{G}_F(j)+1} \right]_\wedge$$
$$+ \beta_j^F \left[\widetilde{\mathbf{X}}^F_{\mathscr{G}_F(i)+1} - \widetilde{\mathbf{X}}^F_{\mathscr{G}_F(i)-1} \right]_\wedge + \left(\delta_{\mathscr{G}_F(j),\mathscr{G}_F(i)-1} - \delta_{\mathscr{G}_F(j),\mathscr{G}_F(i)+1} \right) \left[\widetilde{\mathbf{X}}^F_s \right]_\wedge \Bigg\} \quad (3.43)$$

and $\widetilde{\mathscr{F}}_{ij}$ denotes the set of faces that are connected to both the ith and the jth vertices (with global numbering) of P. In the above expression, $\delta_{i,j}$ denotes the Kronecker delta function and we have made use of the notation

$$[\mathbf{a}]_\wedge = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & a_x \\ -a_y & -a_x & 0 \end{bmatrix} \quad (3.44)$$

for a given vector $\mathbf{a} = [a_x, a_y, a_z]^T$.

We remark that in both 2D and 3D cases, J_E and its first and second variations are all algebraic functions that can be exactly computed using only the coordinates of the vertices in the deformed configuration and the DOFs of fields $\delta\mathbf{v}$ and $\mathbf{w}$.

We also remark that another option of approximating J_E is to simply use the projected gradient. For any given $\delta\mathbf{v}$, $\mathbf{w} \in \mathscr{V}(E)$, we obtain

$$J_E(\mathbf{v}) = \det\left(\mathbf{I} + \mathbf{\Pi}^0_E(\nabla\mathbf{v})\right), \quad (3.45)$$
$$DJ_E(\mathbf{v}) \cdot \delta\mathbf{v} = \frac{\partial J}{\partial \mathbf{F}}\left(\mathbf{I} + \mathbf{\Pi}^0_E(\nabla\mathbf{v})\right) : \mathbf{\Pi}^0_E(\nabla(\delta\mathbf{v})), \quad (3.46)$$
$$D(DJ_E(\mathbf{v}) \cdot \delta\mathbf{v}) \cdot \mathbf{w} = \mathbf{\Pi}^0_E(\nabla(\delta\mathbf{v})) : \frac{\partial^2 J}{\partial \mathbf{F} \partial \mathbf{F}}\left(\mathbf{I} + \mathbf{\Pi}^0_E(\nabla\mathbf{v})\right) : \mathbf{\Pi}^0_E(\nabla\mathbf{w}), \quad (3.47)$$

As compared to the previous formula of J_E derived in this subsection, the above approximations take simpler forms and are easier for computational implementations, especially in the 3D case. Moreover, we observe that the approximated J_E and its first variation become exact for any $\mathbf{u}$ and $\mathbf{v} \in [\mathscr{P}_1(E)]^d$, meaning that the approximation (3.45) is also first-order consistent. Therefore, as demonstrated by numerical examples, the above approximations in the VEM construction can also lead to optimally convergent results. However, we remark that (3.45) is less robust than (3.29) in the sense that, when the level of deformation increases and the shape of the element becomes more irregular, this approach is more prone to experience non-convergence in the Newton-Raphson algorithm and may provide less accurate solutions. We will present both approaches of computing J_E and its variations in the numerical examples of this chapter. For convenience, we refer to the VEM following (3.29), i.e. using expressions (3.36), (3.38) and (3.40)–(3.42), as to the "Robust" approach and to the VEM using (3.45)–(3.47) as to the "Simple" approach. This is illustrated by Table 3.1.

Table 3.1: Approaches for computing area or volume average of $J = \det \mathbf{F}$ over element E.

Approaches	Associated equations
Robust	(3.36), (3.38), (3.40), (3.41), (3.42)
Simple	(3.45), (3.46), (3.47)

3.4.3 Pressure VEM spaces

For the two-field mixed virtual elements used in this chapter, we assume a piecewise constant approximation of the pressure field, which takes constant value over each element E. This approximation has been shown to lead to numerically stable approximations on several polygonal meshes in 2D, such as the Centroidal Voronoi Tessellation (CVT) and structured hexagonal meshes [2], [4], [72]. Accordingly, the finite dimensional pressure space $\mathscr{Q}_h$ is then identified with the following definition:

$$\mathscr{Q}_h \doteq \{\widehat{q}_h \in \mathscr{Q} : \widehat{q}|_E = \text{constant} \forall E \in \Omega_h\} . \tag{3.48}$$

3.5 Variational approximations

Having defined the displacement and pressure VEM spaces, we proceed to propose the displacement-based and two-field mixed VEM approximations on the polygonal/polyhedral mesh Ω_h. In particular, we first introduce the two-field mixed Galerkin approximation assuming the piecewise constant pressure field and exact integration, as well as an equivalent displacement-based Galerkin approximation, which also assume exact integration. Subsequently, decomposition of the exact integral into consistency and stability components is introduced for both approximations, which leads to the final form of the displacement-based and mixed VEM approximations. Moreover, we discuss different constructions of stability and loading terms and propose a new stabilization scheme for isotropic solids. Finally, the polynomial consistency of the VEM approximations is demonstrated.

3.5.1 Two-field mixed and equivalent displacement-based Galerkin approximations with exact integration

For lower-order mixed elements, whose pressure field $\widehat{q}_h$ is assumed to be constant over each element, the Galerkin approximation consists of finding the equilibrating displacement $\mathbf{u}_h$ and pressure $\widehat{p}_h$, such that

$$\widehat{\Pi}_h\left(\mathbf{u}_h, \widehat{p}_h\right) = \min_{\mathbf{v}_h \in \mathscr{K}_h} \max_{\widehat{q}_h \in \mathscr{Q}_h} \widehat{\Pi}\left(\mathbf{v}_h, \widehat{q}_h\right), \tag{3.49}$$

where

$$\widehat{\Pi}_h\left(\mathbf{v}_h, \widehat{q}_h\right) = \sum_E \int_E \left\{\Psi(\mathbf{F}\left(\mathbf{v}_h\right)) + \widehat{q}_h\left[J\left(\mathbf{v}_h\right) - 1\right] - \widehat{U}^*\left(\widehat{q}_h\right)\right\} \mathrm{d}\mathbf{X} - \sum_E \int_E \mathbf{f}\cdot\mathbf{v}_h \mathrm{d}\mathbf{X} - \int_{\Gamma_h^{\mathbf{t}}} \mathbf{t}\cdot\mathbf{v}_h \mathrm{d}S. \tag{3.50}$$

For the time being, we assume exact integration over E in our Galerkin approximations.

Since $\widehat{q}_h$ takes constant values over E the above formulation can be rewritten as

$$\widehat{\Pi}_h(\mathbf{v}_h,\widehat{q}_h) = \sum_E \int_E \Psi(\mathbf{F}(\mathbf{v}_h))\mathrm{d}\mathbf{X} + \sum_E \left\{|E| \left[\widehat{q}_h|_E \left(J_E(\mathbf{v}_h) - 1\right) - \widehat{U}^*(\widehat{q}_h|_E)\right]\right\}$$
$$- \sum_E \int_E \mathbf{f}\cdot\mathbf{v}_h \mathrm{d}\mathbf{X} - \int_{\Gamma_h^{\mathbf{t}}} \mathbf{t}\cdot\mathbf{v}_h \mathrm{d}S. \quad (3.51)$$

Observe that J_E appears in the second term on the right-hand side of the above expression, which can evaluated either exactly (3.29) or approximately (3.45) using the formulas provided in the preceding section.

We also introduce a displacement-based Galerkin approximation that consists of seeking the unknown displacement field such that

$$\Pi_h(\mathbf{u}_h) = \min_{\mathbf{v}_h\in\mathscr{K}_h} \Pi_h(\mathbf{v}_h), \quad (3.52)$$

where

$$\Pi_h(\mathbf{v}_h) = \sum_E \int_E \Psi(\mathbf{F}(\mathbf{v}_h))\mathrm{d}\mathbf{X} + \sum_E |E|\,U(J_E(\mathbf{v}_h)) - \sum_E \int_E \mathbf{f}\cdot\mathbf{v}_h\mathrm{d}\mathbf{X} - \int_{\Gamma_h^{\mathbf{t}}} \mathbf{t}\cdot\mathbf{v}_h\mathrm{d}S. \quad (3.53)$$

By realizing that

$$\sum_E |E|U(J_E(\mathbf{v}_h)) = \max_{\widehat{q}_h\in\mathscr{Q}_h} \left\{\sum_E |E| \left[\widehat{q}_h|_E \left(J_E(\mathbf{v}_h) - 1\right) - \widehat{U}^*(\widehat{q}_h|_E)\right]\right\} \quad (3.54)$$

as a consequence of (3.8), we note that the displacement-based approximation is equivalent to the two-field mixed one except for the case of purely incompressible materials, where the displacement approximation fails because of (3.2). In the finite element method (FEM) literature, this equivalence is also explored in [141]. Moreover, since $\mathrm{d}^2U/(dJ)^2 \to +\infty$ as the material approaches incompressibility, employing the above displacement-based approximation for nearly incompressible materials will lead to finite element systems with high condition numbers in the stiffness matrices, possibly resulting in poor convergence performance of the Newton-Raphson method. By contrast, the two-field mixed approximation is free of the above shortcomings and is valid for materials with any level of incompressibility, including those that are purely incompressible.

This chapter also considers the standard displacement-based and two-field mixed FEM approximations on polygonal meshes [2]. The Mean Value coordinates are used to construct the displacement spaces [97] and the gradient correction scheme [68] is also adopted to ensure the polynomial consistency of the finite element approximations.

3.5.2 Displacement-based and two-field mixed VEM approximations

This subsection proceeds to propose the displacement-based and two-field mixed VEM approximations. Following the original idea of the VEM, we decompose the exact integrals associated with Ψ in (3.51) and (3.53) into "consistency" and "stability" terms [37].

The displacement-based VEM approximation then consists of seeking the unknown displacement field $\mathbf{u}_h$ such that it satisfies Eq. (3.52) with

$$\Pi_h^{VEM}(\mathbf{v}_h) = \sum_E |E| \Psi\left(\mathbf{I} + \mathbf{\Pi}_E^0 \nabla \mathbf{v}_h\right) + \frac{1}{2} \sum_E \alpha_E(\mathbf{s}_h) S_{h,E}\left(\mathbf{v}_h - \mathbf{\Pi}_E^{\nabla}\mathbf{v}_h, \mathbf{v}_h - \mathbf{\Pi}_E^{\nabla}\mathbf{v}_h\right) \\ + \sum_E |E| U\Big(J_E(\mathbf{v}_h)\Big) - \langle \mathbf{f}, \mathbf{v}_h \rangle_h - \langle \mathbf{t}, \mathbf{v}_h \rangle_h. \quad (3.55)$$

In the above expression, $\langle \mathbf{f}, \mathbf{v}_h \rangle_h$ and $\langle \mathbf{t}, \mathbf{v}_h \rangle_h$ stand for the approximated loading terms associated with body force and surface traction respectively, whose forms will be discussed in the next subsection. Moreover, notice that the first term on the right-hand side of (3.55) is first-order consistent, and is thus called the "consistency" term, meaning that if $\mathbf{v}_h$ is a first-order polynomial function, this term recovers the first term in (3.53). However, unless E is a linear triangular/tetrahedral element, the consistency term contains non-physical kernel that may lead to spurious modes in the displacement solution. A stabilization term (the second term on the right-hand side of (3.55)) is thus needed to penalize those non-physical kernels. The stabilization term is made up of two components, $\alpha_E(\mathbf{s}_h)$ and

$S_{h,E}(\mathbf{v}_h - \mathbf{\Pi}_E^{\nabla}\mathbf{v}_h, \mathbf{v}_h - \mathbf{\Pi}_E^{\nabla}\mathbf{v}_h)$. In fact, $S_{h,E}(\cdot,\cdot)$ is a bilinear form given by [44]

$$S_{h,E}\left(\mathbf{v}_h, \mathbf{w}_h\right) = h_E^{d-2} \sum_{v \in E} \mathbf{v}_h\left(\mathbf{X}_v\right) \cdot \mathbf{w}_h\left(\mathbf{X}_v\right) \quad \forall \mathbf{v}_h, \mathbf{w}_h \in \mathscr{K}_h \quad \forall E \in \Omega_h \tag{3.56}$$

with $h_E = |E|^{1/d}$ measuring the size of E, whereas $\alpha_E(\mathbf{s}_h)$ is a scalar-valued function of $\mathbf{s}_h \in \mathscr{K}_h$, a detailed discussion on which will be provided on the next subsection.

In the same manner, we also state the final form of the two-field mixed VEM approximation, which consists of finding the unknown displacement field $\mathbf{u}_h$ and pressure field $\widehat{p}_h$ such that it satisfies Eq. (3.49) with

$$\begin{aligned}\widehat{\Pi}_h^{VEM}\left(\mathbf{v}_h, \widehat{q}_h\right) = \sum_E |E| \Psi\left(\mathbf{I} + \mathbf{\Pi}_E^0 \nabla \mathbf{v}_h\right) + \frac{1}{2} \sum_E \alpha_E\left(\mathbf{s}_h\right) S_{h,E}\left(\mathbf{v}_h - \mathbf{\Pi}_E^{\nabla}\mathbf{v}_h, \mathbf{v}_h - \mathbf{\Pi}_E^{\nabla}\mathbf{v}_h\right) \\ + \sum_E |E| \left[\widehat{q}_h|_E \left(J_E(\mathbf{v}_h) - 1\right) - \widehat{U}^*\left(\widehat{q}_h|_E\right)\right] - \langle \mathbf{f}, \mathbf{v}_h \rangle_h - \langle \mathbf{t}, \mathbf{v}_h \rangle_h.\end{aligned} \tag{3.57}$$

While the focus of this chapter is on developing lower order VEM approximations, a few remarks are worthwhile making on their possible extension to the higher-order VEM approximations. For higher order VEM approximations, the local VEM spaces are well-defined and the associated projections are computable [140]. A possible extension using the “Simple” approach can follow the same methodology presented in this section: one can decompose the stored-energy function into a consistency part which utilizes higher-order projections and then stabilize the remaining part. On the other hand, the extension to higher order VEM considering the “Robust” approach is a more challenging part, which deserves further theoretical developments.

3.5.3 Discussion on the stabilization parameter $\alpha_E(\mathbf{s}_h)$

In the sequel, we review and discuss the stabilization parameter α_E proposed in [44] for small deformation nonlinear elastic and inelastic problems. Motivated by the discussions, we hereby propose a new expression for the stabilization parameter α_E, which is based on the trace of the Hessian of Ψ.

Norm-based stabilization In the chapter by Beirão da Veiga et. al. [44], the stabilization parameter α_E is given by the following expression:

$$\alpha_E(\mathbf{s}_h) = |||\frac{\partial^2\Psi}{\partial\mathbf{F}\partial\mathbf{F}}\left(\mathbf{I}+\mathbf{\Pi}^0_E\nabla\mathbf{s}_h\right)|||, \quad \forall\mathbf{s}_h\in\mathscr{K}_h \tag{3.58}$$

where $|||\cdot|||$ denotes any norm of a fourth order tensor. In the remainder of the chapter, we refer this choice as the norm-based stabilization. As a common choice, the norm induced by the Euclidean vector norm is considered in this chapter. In this case, the stabilization parameter α_E equals to the absolute value of the largest eigenvalue of the Hessian of Ψ. Intuitively, this amounts to assigning the moduli of a material in its stiffest direction to all its directions. As a result, as will be shown in the filled elastomer example, the VEM approximations adopting this stabilization parameter typically yield over stiff responses in large deformation analysis. Notice that in this norm-based stabilization, unlike the original form of α_E used in [44], which essentially uses the Hessian of the entire stored-energy function W, we only use that of the first component of the stored-energy function Ψ (similar to the selective integration concept in the FEM literature).

The choice of $\mathbf{s}_h\in\mathscr{K}_h$ can be any kinematically admissible displacement. Two choices of $\mathbf{s}_h$ are discussed in [44], which are $\mathbf{s}_h=\mathbf{0}$ and $\mathbf{s}_h=\mathbf{u}_h^{n-1}$, where $\mathbf{u}_h^{n-1}$ is the equilibrium displacement field obtained in the last Newton-Raphson step. The former choice amounts to evaluating α_E in the undeformed configuration. For this choice, α_E is a constant scalar that is independent of the deformation state and, as shown in [44], can easily lead to unsatisfactory results also in small deformation regimes. By contrast, the latter choice evaluates α_E in the last deformed configuration that is solved by the Newton-Raphson method, implying that α_E keeps updating according to the deformation states. The motivation for the choice $\mathbf{s}_h=\mathbf{u}_h^{n-1}$ in [44] (instead of the more natural $\mathbf{s}_h=\mathbf{u}_h$ with $\mathbf{u}_h$ the displacement field that we are seeking for) is to simplify the Newton-Raphson iterations by avoiding to compute the derivatives of α_E in the associated tangent matrix. On the other hand, when applied to finite elasticity problems, especially to those involving heterogeneous and large localized deformations, also this second choice has its own shortcomings. As will be demonstrated by the filled elastomer example in Section 6, the choice of $\mathbf{s}_h = \mathbf{0}$ in α_E tends to under stabilize some elements at high deformation levels (which leads to noticeable hourglass-type deformations in those elements), while choosing $\mathbf{s}_h = \mathbf{u}_h^{n-1}$ renders the approximations dependent on the loading histories, possibly making it unsuited for problems involving very large and heterogeneous deformation fields. It is worthwhile noting that, although $\mathbf{s}_h = \mathbf{u}_h$ is a more intuitive choice, it is also more computationally demanding than $\mathbf{s}_h = \mathbf{u}_h^{n-1}$ for the reason mentioned above.

Trace-based stabilization The expression (3.58) was introduced in [44] in order to obtain a (strictly) positive stabilization factor, that is a reasonable condition for small deformation regimes. On the other hand, this may become an unreliable choice in large deformation analysis, where issues related to the physical stability of the material appear. Motivated by the above discussion, we propose a new stabilization parameter α_E which is based on the trace of the Hessian of Ψ given by

$$\alpha_E\left(\mathbf{s}_h\right) = \frac{1}{d^2}\mathrm{tr}\left(\frac{\partial^2\Psi}{\partial\mathbf{F}\partial\mathbf{F}}\left(\mathbf{I}+\mathbf{\Pi}_E^0\nabla\mathbf{s}_h\right)\right), \quad \forall\mathbf{s}_h\in\mathscr{K}_h, \tag{3.59}$$

where d is the dimension $(d = 2,3)$. We refer to it as the trace-based stabilization. Instead of taking into account only the largest eigenvalue of the Hessian of Ψ as in the norm-based stabilization using the Euclidean induced norm, α_E in the trace-based stabilization takes into account all the eigenvalues of the Hessian of Ψ (α_E equals the algebraic average of all the eigenvalues of the Hessian of Ψ). In addition, whereas the norm-based stabilization is always positive, the trace-based stabilization may take negative values, for instance, when the eigenvalues of the Hessian are dominantly negative. This issue is closely related to the physical stability of the material, a detailed analysis of which is beyond the scope of this chapter and is subjected to future chapter [142].

When specialized to isotropic materials, explicit expressions can be obtained for α_E in the trace-based stabilization. For isotropic material, $\Psi(\mathbf{F})$ is a function $\Phi(I_1, I_2, J)$ which solely depends on the three invariants I_1, I_2 and J of the right Cauchy-Green deformation tensor $\mathbf{C} = \mathbf{F}^T\mathbf{F}$, where $I_1 = \mathrm{tr}\mathbf{C}$ and $I_2 = 1/2[(\mathrm{tr}\mathbf{C})^2 - \mathrm{tr}(\mathbf{C}^2)]$. Under this condition, the proposed stabilization parameter α_E is given by:

$$\alpha_E\left(\mathbf{s}_h\right) = \frac{1}{d^2}\mathrm{tr}\left[\frac{\partial^2\Phi}{\partial\mathbf{F}\partial\mathbf{F}}\left(I_1\left(\mathbf{I}+\mathbf{\Pi}_E^0\nabla\mathbf{s}_h\right), I_2\left(\mathbf{I}+\mathbf{\Pi}_E^0\nabla\mathbf{s}_h\right), J\left(\mathbf{I}+\mathbf{\Pi}_E^0\nabla\mathbf{s}_h\right)\right)\right]. \qquad (3.60)$$

As will be shown in detail in Appendix A, making use of the Cayley-Hamilton theorem, the above expression can be further simplified as

$$\begin{aligned}\alpha_E\left(\mathbf{s}_h\right) = \frac{1}{4}\Bigg[&4I_1\frac{\partial^2\Phi}{\partial I_1\partial I_1} + \left(8I_1I_2 - 4J^2I_1\right)\frac{\partial^2\Phi}{\partial I_2\partial I_2} + I_1\frac{\partial^2\Phi}{\partial J\partial J} + 16I_2\frac{\partial^2\Phi}{\partial I_1\partial I_2} + 8J\frac{\partial^2\Phi}{\partial I_1\partial J} \\ &+2JI_1\frac{\partial^2\Phi}{\partial I_2\partial J} + 8\frac{\partial\Phi}{\partial I_1} + 2I_1\frac{\partial\Phi}{\partial I_2}\Bigg]. \qquad (3.61)\end{aligned}$$

in 2D and

$$\begin{aligned}\alpha_E\left(\mathbf{s}_h\right) = \frac{1}{9}\Bigg[&4I_1\frac{\partial^2\Phi}{\partial I_1\partial I_1} + \left(4I_1I_2 + 12J^2\right)\frac{\partial^2\Phi}{\partial I_2\partial I_2} + I_2\frac{\partial^2\Phi}{\partial J\partial J} + 16I_2\frac{\partial^2\Phi}{\partial I_1\partial I_2} + 12J\frac{\partial^2\Phi}{\partial I_1\partial J} \\ &+4JI_1\frac{\partial^2\Phi}{\partial I_2\partial J} + 18\frac{\partial\Phi}{\partial I_1} + 8I_1\frac{\partial\Phi}{\partial I_2}\Bigg]. \qquad (3.62)\end{aligned}$$

in 3D, where implicit dependences of I_1, I_2 and J on $\mathbf{\Pi}_E^0\nabla\mathbf{s}_h$ are assumed. The above explicit expressions are helpful to enable the more intuitive choice of $\mathbf{s}_h = \mathbf{u}_h$ in the implementation, which is found to render the VEM approximations independent of the loading history, as well as more tolerant to heterogeneous and large localized deformations than the choice of $\mathbf{s}_h = \mathbf{u}_h^{n-1}$. On the other hand, unlike choosing $\mathbf{s}_h = \mathbf{0}$ and $\mathbf{u}_h^{n-1}$, $\alpha_E(\mathbf{u}_h)$ becomes a nonlinear function of $\mathbf{u}_h$, resulting in additional terms in the weak forms of the VEM approximations in practice.

Table 3.2: Summary of stabilization schemes and choices of $\mathbf{s}_h$. A symbol "×" means that such choice has not been tested numerically–the reason can be found in the corresponding footnote.

	$\mathbf{s}_h = \mathbf{0}$	$\mathbf{s}_h = \mathbf{u}_h^{n-1}$	$\mathbf{s}_h = \mathbf{u}_h$
norm-based stabilization	✓	✓	×[1]
trace-based stabilization	×[2]	✓	✓

3.5.4 Construction of the loading terms

We present the constructions of first-order accurate loading terms which, according to [42], guarantee the optimal convergence of the displacement and its gradient in the L^2 error.

To construct the loading term associated with surface traction $\langle \mathbf{t}, \mathbf{v}_h \rangle_h$, a first-order Gauss-Lobatto quadrature rule is used in 2D with the quadrature points being the vertices v on $\Gamma_h^{\mathbf{t}}$,

$$\langle \mathbf{t}, \mathbf{v}_h \rangle_h = \sum_{v \in \Gamma_h^{\mathbf{t}}} w_v \mathbf{t}\left(\mathbf{X}_v\right) \cdot \mathbf{v}_h\left(\mathbf{X}_v\right), \quad \forall \mathbf{v}_h \in \mathscr{K}_h, \tag{3.63}$$

where w_v is the weight associated with v defined by a first-order Gauss-Lobatto rule. One the other hand, the vertex-based quadrature defined by (3.25) over F with m^F vertices is adopted in 3D as follows:

$$\langle \mathbf{t}, \mathbf{v}_h \rangle_h = \sum_{F \in \Gamma_h^{\mathbf{t}}} \sum_{j=1}^{m^F} w_j^F \mathbf{t}\left(\mathbf{X}_j^F\right) \cdot \mathbf{v}_h\left(\mathbf{X}_j^F\right), \quad \forall \mathbf{v}_h \in \mathscr{K}_h. \tag{3.64}$$

Notice that in both 2D and 3D cases, the loading term $\langle \mathbf{t}, \mathbf{v}_h \rangle_h$ yields exact integral when $\mathbf{t}$ is constant on each edge/face of $\Gamma_h^{\mathbf{t}}$, i.e.

$$\langle \mathbf{t}, \mathbf{v}_h \rangle_h = \int_{\Gamma_h^{\mathbf{t}}} \mathbf{t} \cdot \mathbf{v}_h \mathrm{d}S, \quad \forall \mathbf{v}_h \in \mathscr{K}_h. \tag{3.65}$$

Regarding the loading term associated with the body force $\langle \mathbf{f}, \mathbf{v}_h \rangle_h$, various approaches

[1] This choice in the norm-based stabilization is significantly more computationally demanding than other choices and is therefore not considered in this chapter.

[2] Although this choice can be easily considered, we prefer to disregard it because it shares the same drawback as choosing $\mathbf{s}_h = \mathbf{0}$ in the norm-based stabilization.

can be utilized. We consider two approaches in this chapter, neither of which needs explicit knowledge of $\mathbf{v}_h$ in the interior of each element E. The first approach utilizes a (first-order) vertex-based quadrature over each element in 2D and 3D,

$$\text{Approach \#1: } \langle \mathbf{f}, \mathbf{v}_h \rangle_h = \sum_E \sum_{v \in E} w_v^E \mathbf{f}(\mathbf{X}_v) \cdot \mathbf{v}_h(\mathbf{X}_v), \quad \forall \mathbf{v}_h \in \mathscr{K}_h, \tag{3.66}$$

where $\mathbf{X}_v$ denotes the location of vertex v of E and w_v^E is the associated weight defined in [26], [42]. Notice that the first-order accuracy of the nodal quadrature rule in [26], [42] is only valid for convex elements and certain concave elements in 2D and 3D. For instance, with the presence of non star-convex elements, certain weights in the nodal become negative and moreover the first-order accuracy may be no longer satisfied.

We also propose another approach for the construction of the loading term $\langle \mathbf{f}, \mathbf{v}_h \rangle_h$, which is of the form

$$\text{Approach \#2: } \langle \mathbf{f}, \mathbf{v}_h \rangle_h = \sum_E \mathbf{f}\left(\mathbf{X}_c^E\right) \cdot \left(\mathbf{\Pi}_E^\nabla \mathbf{v}_h\right)\left(\mathbf{X}_c^E\right), \quad \forall \mathbf{v}_h \in \mathscr{K}_h, \tag{3.67}$$

where $\mathbf{X}_c^E \doteq \int_E \mathbf{X} \mathrm{d}\mathbf{X} / |E|$ stands for the centroid of element E, which can be calculated for any given elements with arbitrary shapes in 2D and 3D. Although the centroid $\mathbf{X}_c^E$ may not lie in the interior of E, we can show that this construction of the body force term is first-order accurate for elements with arbitrary shapes, such as non star-shaped elements, in both 2D and 3D as long as $\mathbf{\Pi}_E^\nabla$ is well defined. In fact, a two dimensional example will be presented in a subsequent section which confirms that VEM with the second load construction produces optimally convergent results also for non star-shaped elements. The only limitation of this second choice is that, whenever $\mathbf{X}_c^E \notin E$, loadings that are discontinuous across mesh edges may yield less accurate results.

3.5.5 Weak forms of the VEM approximations

With the proposed stabilization parameter and loading terms, the weak form for the displacement-based VEM approximation (3.52) and (3.55) is given by (assuming the trace-based stabilization scheme in which $\mathbf{s}_h$ is taken as the current displacement):

$$\begin{aligned}
&G_h(\mathbf{v}_h,\delta\mathbf{v}_h)\\
&=\sum_E |E|\left[\frac{\partial\Psi}{\partial\mathbf{F}}\left(\mathbf{I}+\mathbf{\Pi}^0_E\nabla\mathbf{v}_h\right)+\frac{1}{2}S_{h,E}\left(\mathbf{v}_h-\mathbf{\Pi}^\nabla_E\mathbf{v}_h,\mathbf{v}_h-\mathbf{\Pi}^\nabla_E\mathbf{v}_h\right)\frac{\partial\alpha_E}{\partial\mathbf{F}}(\mathbf{v}_h)\right]:\mathbf{\Pi}^0_E\nabla(\delta\mathbf{v}_h)\\
&\quad+\sum_E\alpha_E(\mathbf{v}_h)S_{h,E}\left(\mathbf{v}_h-\mathbf{\Pi}^\nabla_E\mathbf{v}_h,\delta\mathbf{v}_h-\mathbf{\Pi}^\nabla_E(\delta\mathbf{v}_h)\right)+\sum_E|E|\frac{\mathrm{d}U}{\mathrm{d}J}\left(J_E(\mathbf{v}_h)\right)DJ_E(\mathbf{v}_h)\cdot\delta\mathbf{v}_h|_E\\
&\quad-\langle\mathbf{f},\delta\mathbf{v}_h\rangle_h-\langle\mathbf{t},\delta\mathbf{v}_h\rangle_h=0\quad\forall\delta\mathbf{v}_h\in\mathscr{K}^0_h.
\end{aligned}\tag{3.68}$$

On the other hand, the weak forms of the first variation of the two-field mixed VEM approximation (3.49) and (3.57) take the following form:

$$\begin{aligned}
&G^{\mathbf{v}}_h(\mathbf{v}_h,\widehat{q}_h,\delta\mathbf{v}_h)\\
&=\sum_E |E|\left[\frac{\partial\Psi}{\partial\mathbf{F}}\left(\mathbf{I}+\mathbf{\Pi}^0_E\nabla\mathbf{v}_h\right)+\frac{1}{2}S_{h,E}\left(\mathbf{v}_h-\mathbf{\Pi}^\nabla_E\mathbf{v}_h,\mathbf{v}_h-\mathbf{\Pi}^\nabla_E\mathbf{v}_h\right)\frac{\partial\alpha_E}{\partial\mathbf{F}}(\mathbf{v}_h)\right]:\mathbf{\Pi}^0_E\nabla(\delta\mathbf{v}_h)\\
&\quad+\sum_E\alpha_E(\mathbf{v}_h)S_{h,E}\left(\mathbf{v}_h-\mathbf{\Pi}^\nabla_E\mathbf{v}_h,\delta\mathbf{v}_h-\mathbf{\Pi}^\nabla_E(\delta\mathbf{v}_h)\right)+\sum_E|E|\widehat{q}_h|_E\left(DJ_E(\mathbf{v}_h)\cdot\delta\mathbf{v}_h|_E\right)\\
&\quad-\langle\mathbf{f},\delta\mathbf{v}_h\rangle_h-\langle\mathbf{t},\delta\mathbf{v}_h\rangle_h=0\quad\forall\delta\mathbf{v}_h\in\mathscr{K}^0_h,
\end{aligned}\tag{3.69}$$

and

$$G^{\widehat{q}}_h(\mathbf{v}_h,\widehat{q}_h,\delta\widehat{q}_h)=\sum_E|E|\left\{\widehat{q}_h|_E\left[J_E(\mathbf{v}_h)-1\right]-\frac{\mathrm{d}\widehat{U}^*}{\mathrm{d}q}\left(\widehat{q}_h|_E\right)\delta\widehat{q}_h|_E\right\}=0\quad\forall\delta\widehat{q}_h\in\mathscr{Q}_h.\tag{3.70}$$

We underline that, whenever $\mathbf{s}_h$ is taken as the displacement at the previous load increment iteration step $\mathbf{u}^{n-1}_h$ (see Section 3.5.3), the addendum involving $\partial\alpha_E/\partial\mathbf{F}$ in both the right-hand sides of (3.68) and (3.69) vanish, and $\alpha_E(\mathbf{v}_h)$ becomes $\alpha_E(\mathbf{u}^{n-1}_h)$.

3.5.6 On the VEM patch test

We provide a brief discussion on the performance of VEM approximations in the patch test. In the discussion that follows, we restrict our attention to cases where the geometry and boundary conditions are exactly represented by the mesh, namely, no error arises from the approximation of geometry in the following discussion.

In the patch test, the unknown displacement field is taken to be a linear vector field, i.e., $\mathbf{u} = \mathbf{p}_1 \in [\mathscr{P}_1(\Omega)]^d$. Accordingly, the first Piola-Kirchhoff stress $\mathbf{P}$ is a constant tensor given by $\mathbf{P} = \partial\Psi(\mathbf{F}(\mathbf{p}_1))/\partial\mathbf{F} + \partial U(J(\mathbf{p}_1))/\partial\mathbf{F}$. The body force $\mathbf{f}$ is $\mathbf{0}$ everywhere and the boundary traction $\mathbf{t}$ is given by $\mathbf{t} = \mathbf{P}\mathbf{n}$ on $\Gamma^{\mathbf{X}}$ where $\mathbf{n}$ is the outward unit normal vector on the boundary. Additionally, the known pressure field $\widehat{p}_0$ is a constant field taking the value $\frac{\partial U}{\partial J}(J(\mathbf{p}_1))$ unless in the presence of a purely incompressible solid, in which case $\widehat{p}_0$ is determined by the applied traction $\mathbf{t}$. Furthermore, for any linear function $\mathbf{p}_1 \in [\mathscr{P}_1(\Omega)]^d$, $\nabla\mathbf{p}_1$ and $\mathbf{F}(\mathbf{p}_1)$ are constant tensors. It follows that for any $E \in \Omega_h$,

$$\mathbf{\Pi}_E^0(\nabla\mathbf{p}_1|_E) = \nabla\mathbf{p}_1 \tag{3.71}$$

$$\mathbf{\Pi}_E^\nabla(\mathbf{p}_1|_E) = \mathbf{p}_1|_E \quad \text{and} \tag{3.72}$$

$$J_E(\mathbf{p}_1) = J(\mathbf{p}_1) = \det\Big(\mathbf{F}(\mathbf{p}_1)\Big). \tag{3.73}$$

We proceed to verify the exact satisfaction of the patch test with the proposed displacement-based and two-field mixed VEM approximations by demonstrating that $\mathbf{v}_h = \mathbf{p}_1$ and $\widehat{q}_h = \widehat{p}_0$ are solutions to Eqs. (3.68), (3.69) and (3.70). Let us first consider the displacement-based VEM approximation. Realizing that exact integration is recovered in the following two terms as a result of (3.71)–(3.73)

$$|E|\frac{\partial\Psi}{\partial\mathbf{F}}\Big(\mathbf{F}(\mathbf{p}_1)\Big) : \mathbf{\Pi}_E^0\Big(\nabla(\delta\mathbf{v}_h)\Big) = \int_E \frac{\partial\Psi}{\partial\mathbf{F}}\Big(\mathbf{F}(\mathbf{p}_1)\Big) : \nabla(\delta\mathbf{v}_h)\mathrm{d}\mathbf{X} \tag{3.74}$$

and

$$|E|\frac{\mathrm{d}U}{\mathrm{d}J}\Big(J_E\left(\mathbf{p}_1\right)\Big)\,DJ_E\left(\mathbf{p}_1\right)\cdot\delta\mathbf{v}_h|_E=\int_E\frac{\mathrm{d}U}{\mathrm{d}J}\Big(J\left(\mathbf{p}_1\right)\Big)\frac{\partial J}{\partial\mathbf{F}}\Big(\mathbf{F}\left(\mathbf{p}_1\right)\Big):\nabla(\delta\mathbf{v}_h)\mathrm{d}\mathbf{X},\quad(3.75)$$

we have that for any trial field $\delta\mathbf{v}_h\in\mathscr{K}_h^0$

$$\begin{aligned}&G_h\left(\mathbf{p}_1,\delta\mathbf{v}_h\right)\\&=\sum_E\int_E\left[\frac{\partial\Psi}{\partial\mathbf{F}}\left(\mathbf{F}(\mathbf{p}_1)\right)+\frac{\mathrm{d}U}{\mathrm{d}J}\left(J\left(\mathbf{p}_1\right)\right)\frac{\partial J}{\partial\mathbf{F}}\left(\mathbf{F}(\mathbf{p}_1)\right)\right]:\nabla(\delta\mathbf{v}_h)\mathrm{d}\mathbf{X}-\langle\mathbf{t},\delta\mathbf{v}_h\rangle_h\\&=\int_\Omega\left[\frac{\partial\Psi}{\partial\mathbf{F}}\left(\mathbf{F}(\mathbf{p}_1)\right)+\frac{\mathrm{d}U}{\mathrm{d}J}\left(J\left(\mathbf{p}_1\right)\right)\frac{\partial J}{\partial\mathbf{F}}\left(\mathbf{F}(\mathbf{p}_1)\right)\right]:\nabla(\delta\mathbf{v}_h)\mathrm{d}\mathbf{X}-\int_{\Gamma_h^{\mathbf{t}}}\mathbf{t}\cdot\delta\mathbf{v}_h\mathrm{d}S=0,\end{aligned}\quad(3.76)$$

where the second equality in the above relation follows from the fact that **t** is constant on each edge/face together with relation (3.65).

For the two-field mixed VEM approximation, we are able to show in a similar manner that for any $\delta\mathbf{v}_h\in\mathscr{K}_h^0$,

$$G_h^{\mathbf{v}}(\mathbf{p}_1,\widehat{p}_0,\delta\mathbf{v}_h)=\sum_E\int_E\left[\frac{\partial\Psi}{\partial\mathbf{F}}\left(\mathbf{F}(\mathbf{p}_1)\right)+\widehat{p}_0\frac{\partial J}{\partial\mathbf{F}}\left(\mathbf{F}(\mathbf{p}_1)\right)\right]:\nabla(\delta\mathbf{v}_h)\mathrm{d}\mathbf{X}-\langle\mathbf{t},\delta\mathbf{v}_h\rangle_h=0,\quad(3.77)$$

and for any $\delta\widehat{q}_h\in\mathscr{Q}_h$

$$\begin{aligned}G_h^{\widehat{q}}(\mathbf{p}_1,\widehat{p}_0,\delta\widehat{q}_h)&=\sum_E|E|\left\{\left[J_E\left(\mathbf{p}_1\right)-1\right]\delta\widehat{q}_h|_E-\frac{\mathrm{d}\widehat{U}^*}{\mathrm{d}q}\left(\widehat{p}_0\right)\delta\widehat{q}_h|_E\right\}\\&=\sum_E\int_E\left[J(\mathbf{p}_1)-1-\frac{\mathrm{d}\widehat{U}^*}{\mathrm{d}\widehat{q}}\left(\widehat{p}_0\right)\right]\delta\widehat{q}_h\mathrm{d}\mathbf{X}=0.\end{aligned}\quad(3.78)$$

According to the above analysis, both displacement-based and two-field mixed VEM approximations pass the patch test exactly in 2D and 3D. In fact, although not presented in this chapter, our numerical studies confirm that the patch tests are passed with errors up to machine precisions for both displacement-based and mixed VEM approximations.

3.5.7 A note on the performance of polyhedral elements with non star-convex faces in patch tests

The discussion in the last subsection is applicable to meshes consisting of polyhedral elements with faces that are star-convex with respect to $\mathbf{X}_s^F$. In the case where the element P contains any face F that is not star-convex with respect to $\mathbf{X}_s^F$, the local displacement VEM space $\mathscr{V}(P)$ is not well defined according to its formal definition (3.21) because certain regions of the triangulated subdomains may lie outside of F. However, our numerical studies demonstrate that convergence (in terms of DOFs) is still achieved with the VEM formulations even for meshes with elements containing non star-convex faces. While a more rigorous theoretical analysis is beyond the scope of this chapter, we demonstrate that the patch test is strictly passed with our mixed VEM formulation even when the meshes contain elements with non star-convex faces. A similar analysis can also be applied to the displacement-based VEM formulation.

To begin with, although the element space $\mathscr{V}(P)$ is not well defined when P contains non star-convex faces, we remark that the projection $\mathbf{\Pi}_E^0 \nabla \mathbf{v}$, is still computable numerically using the nodal values of $\mathbf{v}$ and the vertex based quadrature rule defined in (3.25):

$$\mathbf{\Pi}_E^0 \nabla \mathbf{v} = \frac{1}{|P|} \sum_{F \in \partial P} \sum_{j=1}^{m^F} \left\{ w_j^F \mathbf{v}\left(\mathbf{X}_j^F\right) \otimes \mathbf{n}^F \right\}, \tag{3.79}$$

where the weights w_j^F may take negative values. The same also applies to the other projection $\mathbf{\Pi}_E^\nabla \mathbf{v}$.

As we will show in detail in Appendix B, with the definition of the vertex-based rule on F in (3.25) and (3.79), the following expression holds

$$\mathbf{\Pi}_E^0 \nabla \mathbf{p}_1 = \frac{1}{|P|} \sum_{F \in \partial P} \sum_{j=1}^{m^F} \left\{ w_j^F \mathbf{p}_1\left(\mathbf{X}_j^F\right) \otimes \mathbf{n}^F \right\} = \frac{1}{|P|} \int_P \nabla \mathbf{p}_1 \mathrm{d}\mathbf{X}, \quad \forall \mathbf{p}_1 \in [\mathscr{P}_1(P)]^3 \tag{3.80}$$

even if P contains non star-convex faces.

Furthermore, if P contains non star-convex faces, we are able to show that both the "Robust" and "Simple" approaches of Table 3.1 give exact J_E for any linear displacement field $\mathbf{u} = \mathbf{p}_1 \in [\mathscr{P}_1(P)]^3$, namely,

$$\frac{1}{6|P|}\sum_{\widetilde{F}}\sum_{j=1}^{m^F}\left\{\widetilde{\mathbf{X}}_s^F \cdot \widetilde{\mathbf{X}}_j^F \wedge \widetilde{\mathbf{X}}_{j+1}^F\right\} = J_E\left(\mathbf{p}_1\right). \tag{3.81}$$

Analogous calculations for the first variation of J_E show that expressions (3.41) (the "Robust" approach) and (3.46) (the "Simple" approach) are also valid for the non star-convex case, i.e.,

$$DJ_E(\mathbf{p}_1)\cdot\delta\mathbf{v} = |P|\frac{\partial J}{\partial \mathbf{F}}\Big(\mathbf{F}(\mathbf{p}_1)\Big) : \mathbf{\Pi}_0^E\Big(\nabla(\delta\mathbf{v})\Big). \tag{3.82}$$

Based on (3.79)–(3.82) and realizing that relations (3.71)–(3.73) still hold when P contains non star-convex faces, we are able to show that for any trial fields $\delta\mathbf{v}_h \in \mathscr{K}_h^0$ and $\delta\widehat{q}_h \in \mathscr{Q}_h$, both the "Robust" and "Simple" approaches give

$$G_h^{\mathbf{v}}(\mathbf{p}_1, \widehat{p}_0, \delta\mathbf{v}_h) = 0 \text{ and } G_h^{\widehat{q}}(\mathbf{p}_1, \widehat{p}_0, \delta\widehat{q}_h) = 0, \tag{3.83}$$

indicating that the patch test is strictly passed for mixed VEM even when E contains non star-convex faces. Thorough derivations of expressions (3.80)–(3.83) are provided in Appendix B.

3.6 Numerical assessment

This section presents a series of numerical studies to assess the performance of the displacement-based and mixed VEM approximations in two- and three- dimensional finite elasticity problems. In particular, studies on families of subsequently refined meshes are conducted, through which we demonstrate the performance of VEM formulations on convergence and accuracy.

Unless otherwise stated, the VEM approximations adopt the trace-based stabilization with the choice of $\mathbf{s}_h = \mathbf{u}_h$ throughout this section. For any polyhedral element P, a uniform β_i^F on each face F with m^F vertices is assumed for the remainder of the chapter, namely, $\beta_i^F = 1/m^F$, $i = 1, ..., m^F$. For any given convex polyhedron this choice leads to well-defined local VEM space $\mathscr{V}(P)$ because every face F is guaranteed to be star-shaped with respect to the resulting $\mathbf{X}_s^F$. To solve the nonlinear system of equations, the standard Newton-Raphson method is utilized; as a stopping criterion, we check at each iteration if the Euclidean norm of the residual vector reduces below 10^{-8} times that of the initial residual. To generate 2D polygonal meshes, the general purpose mesh generator `PolyMesher` [110] is used, while for generating 3D polyhedral meshes, the methodology and algorithm provided in [143] are adopted.

To evaluate the accuracy of the numerical solutions, two global error measures of the displacement field are used, the L^2-norm and H^1-seminorm of the displacement error, which are defined as

$$\varepsilon_{0,\mathbf{u}} = ||\mathbf{u} - \mathbf{\Pi}_E^{\nabla} \mathbf{u}_h|| \text{ and } \varepsilon_{1,\mathbf{u}} = ||\nabla \mathbf{u} - \mathbf{\Pi}_E^0 \left(\nabla \mathbf{u}_h\right)|| \tag{3.84}$$

where the $||\cdot||$ (when applied to functions) stands for the L^2 norm over Ω that are evaluated using a fifth order triangulation quadrature scheme. For the meshes with non star-shaped elements, the above-defined displacement error measures become difficult to evaluate due to the failure of the triangulation quadrature scheme. Instead, we adopt the following L^2-type and H^1-type displacement error measures, which utilize only the displacement errors at the vertices of the mesh

$$\varepsilon_{0,\mathbf{u}}^{v} = \left[\sum_E \frac{|E|}{m} \sum_{v \in E} \mathbf{e}\left(\mathbf{X}_v\right) \cdot \mathbf{e}\left(\mathbf{X}_v\right)\right]^{1/2}, \tag{3.85}$$

$$\varepsilon_{1,\mathbf{u}}^{v} = \left\{ \sum_{F} h_F \sum_{e \in F} \left[\mathbf{e}\left(\mathbf{X}_{v_1^e}\right) - \mathbf{e}\left(\mathbf{X}_{v_2^e}\right) \right] \cdot \left[\mathbf{e}\left(\mathbf{X}_{v_1^e}\right) - \mathbf{e}\left(\mathbf{X}_{v_2^e}\right) \right] / |e| \right\}^{1/2} \quad \text{in 2D} \qquad (3.86)$$

and

$$\varepsilon_{1,\mathbf{u}}^{v} = \left\{ \sum_{P} h_P \sum_{F \in \partial P} h_F \sum_{e \in \partial F} \left[\mathbf{e}\left(\mathbf{X}_{v_1^e}\right) - \mathbf{e}\left(\mathbf{X}_{v_2^e}\right) \right] \cdot \left[\mathbf{e}\left(\mathbf{X}_{v_1^e}\right) - \mathbf{e}\left(\mathbf{X}_{v_2^e}\right) \right] / |e| \right\}^{1/2} \quad \text{in 3D} \qquad (3.87)$$

where $\mathbf{e} = \mathbf{u} - \mathbf{u}_h$ is the displacement error, m is the number of vertices in E, h_P and h_F denote the diameters of polyhedron P and face F, respectively, and v_1^e and v_2^e are the two endpoints of edge e. We remark that expressions (3.86) and (3.87) mimic the H^1-seminorms of the displacement in the following sense: they take differences of the displacement on the skeleton of the mesh, and those differences are then scaled in order to achieve the same behavior (with respect to element contractions/expansions) as the L^2-norm of the displacement gradient.

For the mixed formulations in addition to the error measures for displacements defined in (3.84)–(3.87), we consider the L^2 pressure errors

$$\varepsilon_{0,\widehat{p}} = ||\widehat{p} - \widehat{p}_h||, \qquad (3.88)$$

where the norms are evaluated using a fifth order triangulation quadrature rule. For meshes with non star-shaped elements, an L^2-type pressure error is defined as

$$\varepsilon_{0,\widehat{p}}^{v} = \left\{ \sum_{E} |E| \sum_{v \in E} \left[\widehat{p}(\mathbf{X}_v) - \widehat{p}_h(\mathbf{X}_v) \right]^2 / m \right\}^{1/2}, \qquad (3.89)$$

which utilizes the pressure errors at the vertices of the mesh.

3.6.1 Displacement-based VEM

This subsection considers the numerical assessments on the performance of both two- and three-dimensional displacement-based VEM. Throughout this subsection, material behavior is considered to be neo-Hookean with the following stored-energy function [80]:

$$W(\mathbf{F}) = \frac{\mu}{2}\left(I_1(\mathbf{F}) - 3\right) - \mu\left(\det\mathbf{F} - 1\right) + \frac{3\kappa + \mu}{6}\left(\det\mathbf{F} - 1\right)^2, \tag{3.90}$$

where μ and κ are the initial shear and bulk moduli, which are taken to be $\mu = 1$ and $\kappa = 10$ in the remainder of this subsection. Realizing that $\Psi(\mathbf{F}) = \mu/2(\mathbf{F} : \mathbf{F} - 3)$ in the above stored-energy function, we obtain the trace-based stabilization term $\alpha_E = \mu$ for both 2D and 3D cases according to (3.61) and (3.62), which is independent of the deformation state $\mathbf{u}$. Additionally, J_E is computed exactly using the "Robust" approach (Table 3.1) leading to (3.33) and (3.40) for 2D and 3D cases respectively. In general, we have found that computing J_E using the "Simple" approach also produces results with similar accuracy for the meshes with regular elements, e.g., convex elements.

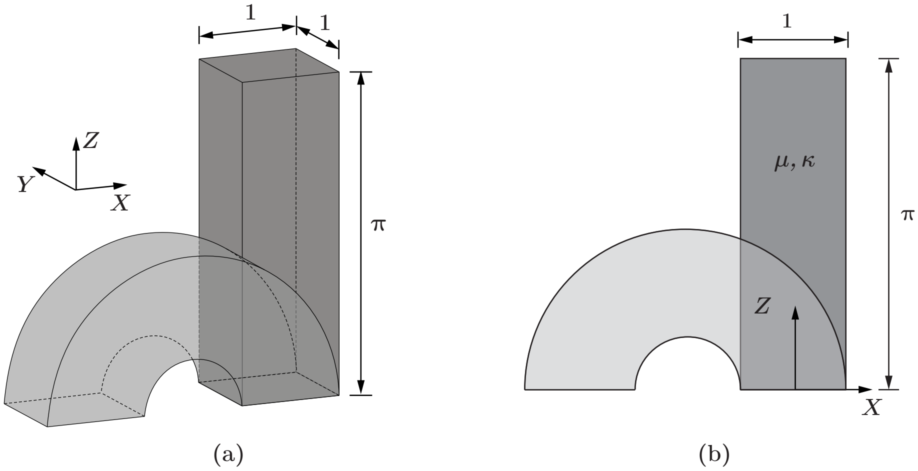


Figure 3.3: (a) Illustration of the 3D boundary value problems where a rectangular block is bent into semicircular shapes. (b) Illustration of the simplified 2D plane strain problems. In the problem for the displacement-based VEM, the initial bulk modulus are set as $\kappa = 10$ and, in the one for the mixed VEM, κ is taken to be $\kappa \to \infty$. In all cases, $\mu = 1$.

This subsection considers a boundary value problem where a rectangular block of dimensions $1 \times 1 \times \pi$ is bent into a semi-circular shape [111], as illustrated in Fig. 3.3 (a). The rectangular block is defined by $-0.5 < X < 0.5$, $0.5 < Y < 0.5$ and $0 < Z < \pi$ in a Cartesian coordinate system (X,Y,Z), and the analytical displacement describing the deformation is given by

$$u_x(\mathbf{X}) = -1 + (1+X)\cos Z - X, \quad u_y(\mathbf{X}) = 0, \quad u_z(\mathbf{X}) = (1+X)\sin Z - Z. \tag{3.91}$$

Accordingly, the body force $\mathbf{f} = [f_x, f_y, f_z]^T$ is computed as

$$f_x(\mathbf{X}) = -\frac{(1+X)\cos Z\,(3\kappa - 2\mu)}{3}, \quad f_y(\mathbf{X}) = 0, \quad f_z(\mathbf{X}) = -\frac{(1+X)\sin Z\,(3\kappa - 2\mu)}{3}. \tag{3.92}$$

Since no deformation occurs in the Y direction, this problem simplifies into a plane strain problem in the $X-Z$ plane, as shown in Fig. 3.3 (b). In the following studies, displacement boundary conditions are applied everywhere on the boundary in both the original 3D problem and its 2D simplification.

2D displacement-based VEM We first consider the VEM in 2D and study its performance in the boundary value problems under plane strain conditions. To that end, we consider the simplified plane strain problem setup illustrated in Fig. 3.3 (b). Two sets of convex meshes are considered, the structured hexagonal meshes and the CVT meshes, examples of which are provided in Fig. 3.4 (a) and (b) respectively. The VEM results include both Approaches #1 and #2 for treating the body force term, i.e. expressions (3.66) and (3.67). For comparison purpose, we also conduct an analysis using the standard displacement-based FEM with the gradient correction scheme [22], [68] on the same sets of polygonal meshes. Figure 3.5 plots the displacement errors as functions of the average mesh size h. Each data point for the CVT mesh represents an average of the results from five meshes. It is observed that VEM with both Approaches #1 and #2 for treating the body force term delivers optimal convergence rates (2 for the L^2-norm of the displacement error and 1 for the H^1-seminorm of the displacement errors), whose results have almost identical accuracy. When compared to the FEM, the VEM also gives displacement fields with similar accuracy (although the VEM results are slightly less accurate in term of the H^1-seminorm than the FEM results). However, the VEM is more efficient than the FEM in the sense that the VEM uses only one integration point per element whereas the FEM uses n integration points for n-sided elements.

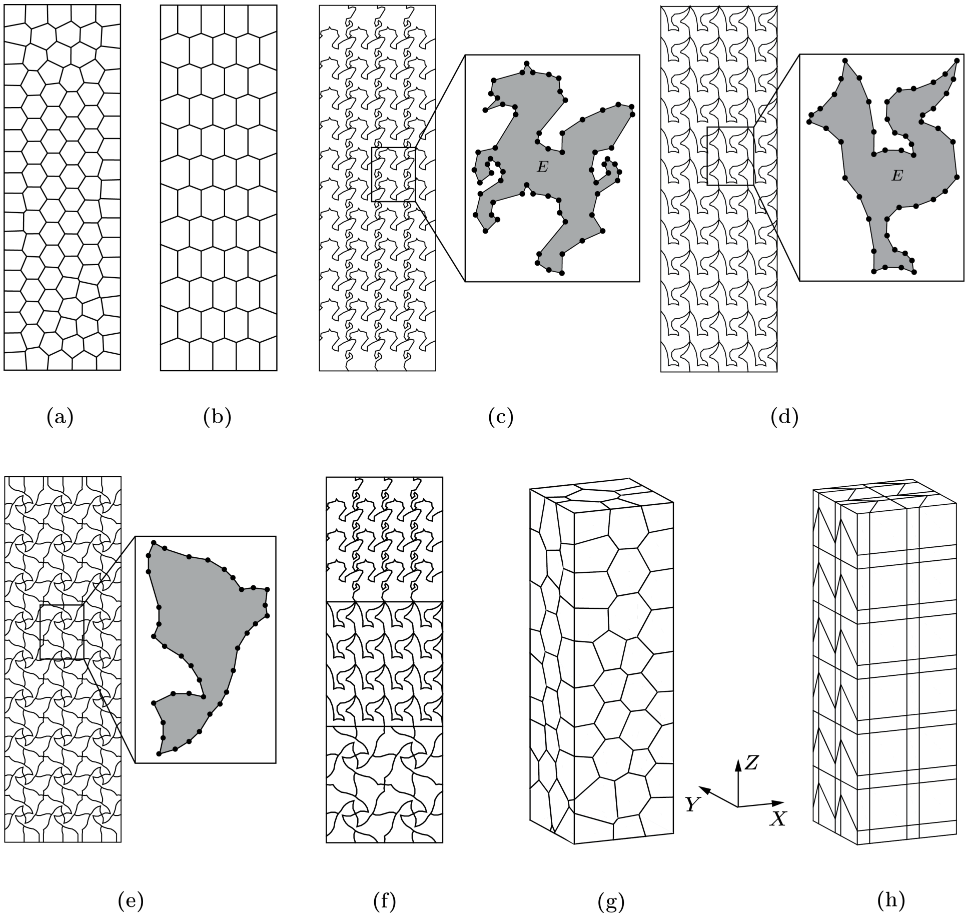


Figure 3.4: Examples of (a) a CVT mesh consisting of 100 2D elements; (b) a structured hexagonal mesh consisting of 60 2D elements; (c) a pegasus mesh containing 48 non star-shaped 2D elements; (d) a bird mesh with 64 non star-shaped 2D elements; (e) a fish mesh consisting of 132 non star-shaped 2D elements; (f) a “zoo” mesh containing a total number of 64 non star-shape 2D elements, including 16 pegasus elements, 20 bird elements and 28 fish elements; (g) a CVT mesh consisting of 100 3D elements; and (h) a distorted hexahedral mesh consisting of 96 3D elements.

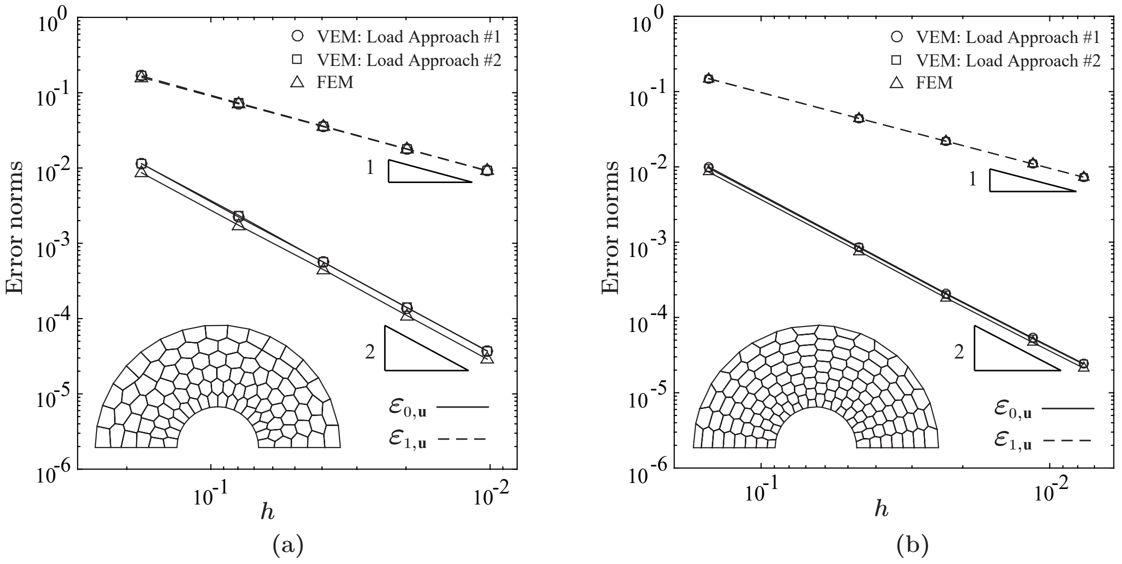


Figure 3.5: The convergence plots of the error norms as functions of the average mesh size h for the (a) the CVT meshes and (b) the structured hexagonal meshes.

One major advantage of the VEM is its ability to deal with more general meshes in an efficient way. To demonstrate this advantage, we consider three sets of Escher-based meshes [40]: the pegasus mesh as shown in Fig. 3.4 (c), the bird mesh as shown in Fig. 3.4 (d), and the fish mesh as shown in Fig. 3.4 (e). Following the discussion of Section 4, because the Approach #1 in (3.66) fails with the non star-shaped elements, the body force term here is constructed using the Approach #2 given by (3.67). Figures 3.6 (a) and (b) show the two error measures (3.85) and (3.86) as functions of the average mesh size h. In addition, examples of the deformed shapes of all sets of meshes are shown in Figs 3.7 – 3.9. Our numerical results confirm that optimal converges are obtained in both the L^2-type and H^1-type norms of the displacement error. As an additional remark, we point out that the VEM using the "Simple" approach (see Table 3.1) to compute J_E in this case is found to lead to non-convergent results due to the failure of convergence in the Newton-Raphson method.

3D displacement-based VEM We proceed to perform numerical assessments of the displacement-based VEM in 3D. In this case, we consider original 3D problem illustrated

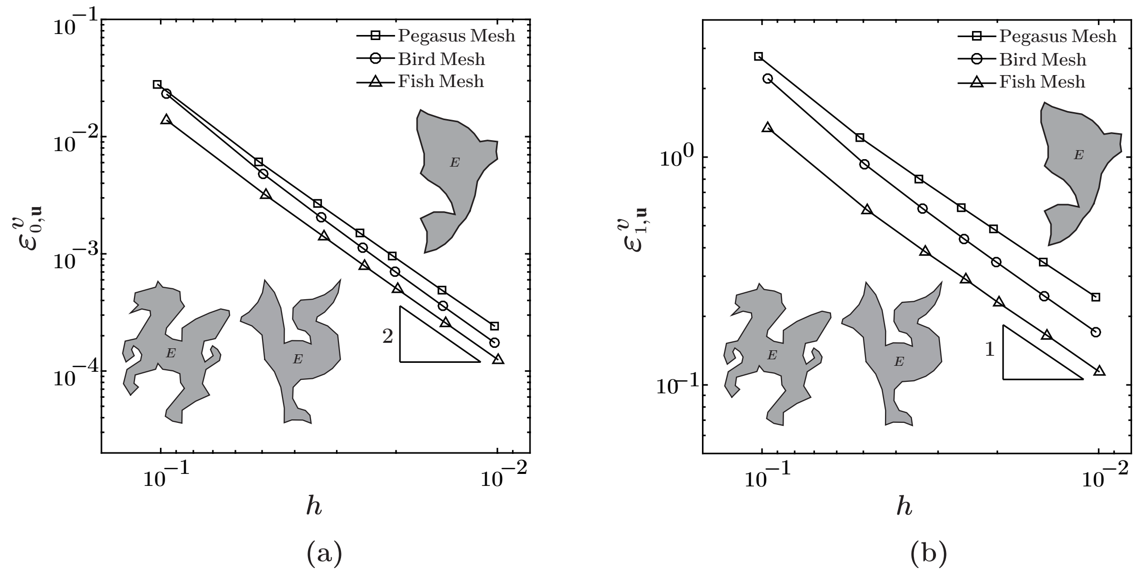


Figure 3.6: The convergence plots of the displacement error norms against average mesh size h for both the bird and pegasus meshes: (a) the L^2-type displacement error and (b) the H^1-type displacement error.

in Fig. 3.3 (a) with the CVT and the distorted hexahedral meshes, examples of which are shown in Fig. 3.4 (g) and (h), respectively. We also consider both approaches of treating the body force term given by (3.66) and (3.67), and the displacement error measures defined in (3.84) are used here. Figures 3.10 (a) and (b) show plots of the displacement error norms as functions of the average mesh sizes h. Each data point in the plots for the CVT mesh is obtained by averaging the results from five meshes. Again, VEM using both Approaches #1 and #2 of treating the body force term yields an optimally convergent displacement field with similar accuracy.

3.6.2 Mixed VEM

In this subsection, the performance of the mixed VEM is numerically evaluated. Throughout the subsection, the material is considered to be purely incompressible which is charac-

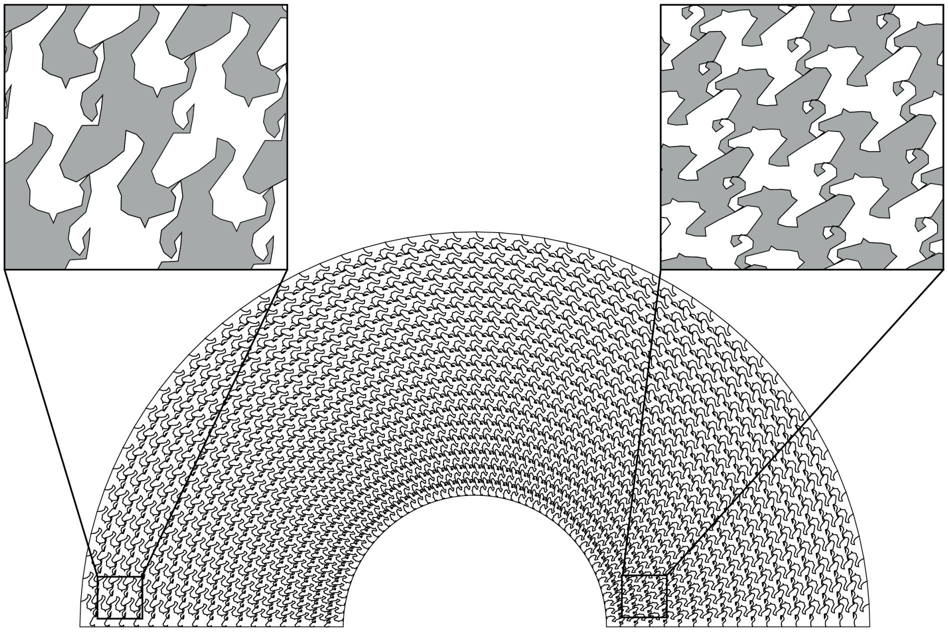


Figure 3.7: The final deformed configuration is shown for the pegasus mesh with 972 elements. This deformed shape is obtained by the displacement-based VEM.

terized by the following stored-energy function [123]:

$$W(\mathbf{F}) = \begin{cases} \frac{3^{1-\alpha}}{2\alpha}\mu\left[(I_1(\mathbf{F}))^{\alpha} - 3^{\alpha}\right] & \text{if } \det\mathbf{F} = 1 \\ +\infty & \text{otherwise} \end{cases}, \tag{3.93}$$

with the initial shear modulus μ being 1 and the material parameter α being 3. For this material model, the trace-based stabilization gives

$$\alpha_E = \frac{3^{1-\alpha}(\alpha+1)\mu}{2} I_1^{\alpha-1} \text{ in 2D and } \alpha_E = \frac{(2\alpha+7)\mu}{3^{\alpha+1}} I_1^{\alpha-1} \text{ in 3D}, \tag{3.94}$$

both of which are nonlinear functions in $\mathbf{u}$.

We consider a similar boundary value problem in which an incompressible rectangular block of dimension $1 \times 1 \times \pi$ is bent in to semi-circle shape. As depicted in Fig. 3.3 (a), the rectangular block is defined by $-0.5 < X < 0.5$, $-0.5 < Y < 0.5$ and $0 < Z < \pi$ in a Cartesian coordinate system (X,Y,Z), with both surfaces $X = \pm 0.5$ being traction free. It is possible to obtain a closed-form solution for this problem [80], [112]. In particular, the

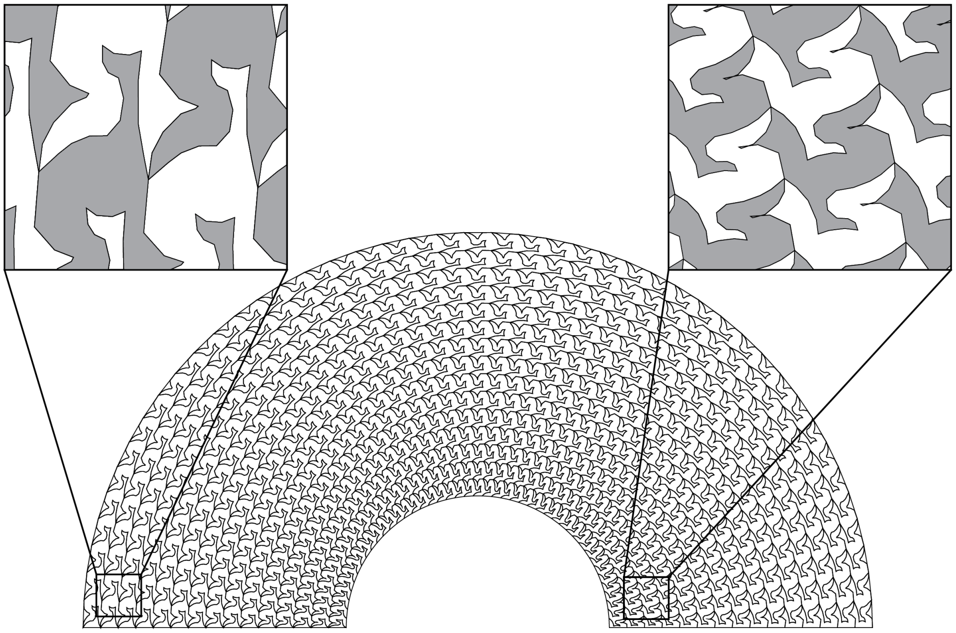


Figure 3.8: The final deformed configuration is shown for the bird mesh with 735 elements. This deformed shape is obtained by the displacement-based VEM.

analytical displacement field takes the form [80], [112]:

$$u_x(\mathbf{X}) = -r(-0.5) + r(X)\cos Z - 0.5 - X, \quad u_y(\mathbf{X}) = 0, \quad u_z(\mathbf{X}) = r(X)\sin Z - Z, \tag{3.95}$$

where the function $r(X)$ is given by

$$r(X) = \sqrt{2X + \sqrt{2}}. \tag{3.96}$$

Additionally, one can obtain the analytical pressure field $\widehat{p}$ as

$$\widehat{p} = \frac{\left(22\sqrt{2}+25\right)\mu}{54} - \frac{\mu\left(r^2(X) + \frac{1}{r^2(X)} + 1\right)^3}{54} + \frac{\mu\left(r^2(X) + \frac{1}{r^2(X)} + 1\right)^2}{9r^2(X)}. \tag{3.97}$$

Unlike the problem in the preceding subsection, the body force **f** for this problem is **0**. Again, this 3D problem can be simplified into a 2D plane strain problem, as illustrated in Fig. 3.3 (b). In our following studies of both the 3D problem and its 2D simplification,

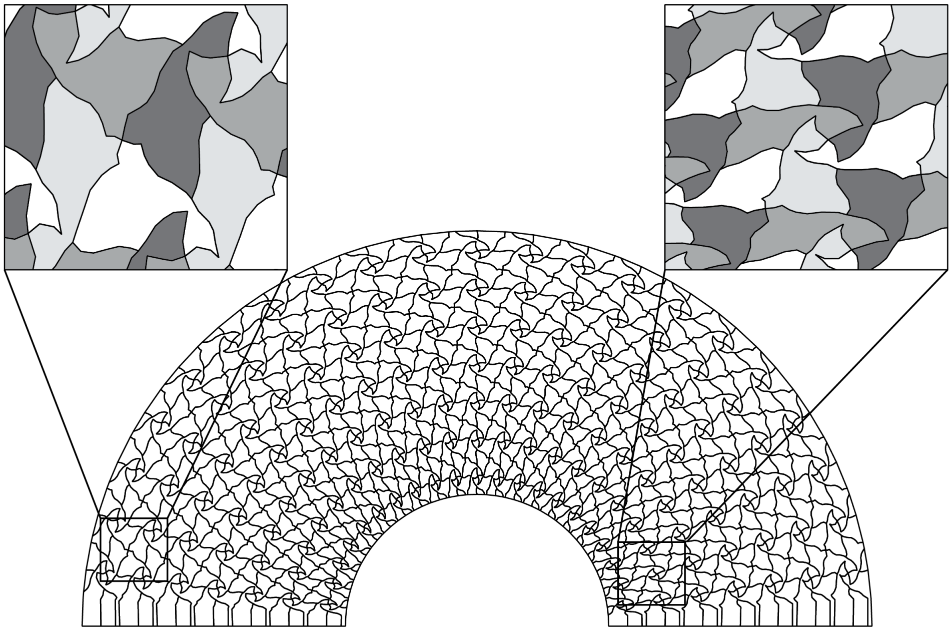


Figure 3.9: The final deformed configuration is shown for the fish mesh with 644 elements. This deformed shape is obtained by the displacement-based VEM.

in order to avoid the potential development of free surface instability (the performance of VEM on capturing the physical instability is beyond the scope of this chapter and is a subject for future investigation), displacement is applied to all the surfaces except the surface $X = 0.5$.

2D mixed VEM We first consider the 2D mixed VEM and study its performance on the accuracy and convergence. We remind that the proposed approximation has been shown to lead to numerically stable schemes (in the linear regime) on several polygonal meshes in 2D, such as the CVT and structured hexagonal meshes [2], [4], [72]. The 2D plane strain problem shown in Fig. 3.3 (b) is considered with the following sets of meshes: structured hexagonal, CVT, and non star-shaped "zoo" meshes made up of pegasus, bird and fish elements, as depicted in Figs. 3.4 (a), (b) and (f) respectively. In Figure 3.11 we plot the displacement and pressure error norms as functions of the average mesh sizes h for the VEM results using both "Robust" and "Simple" approaches to compute J_E. The results obtained form the mixed FEM with the gradient correction scheme for the CVT meshes and

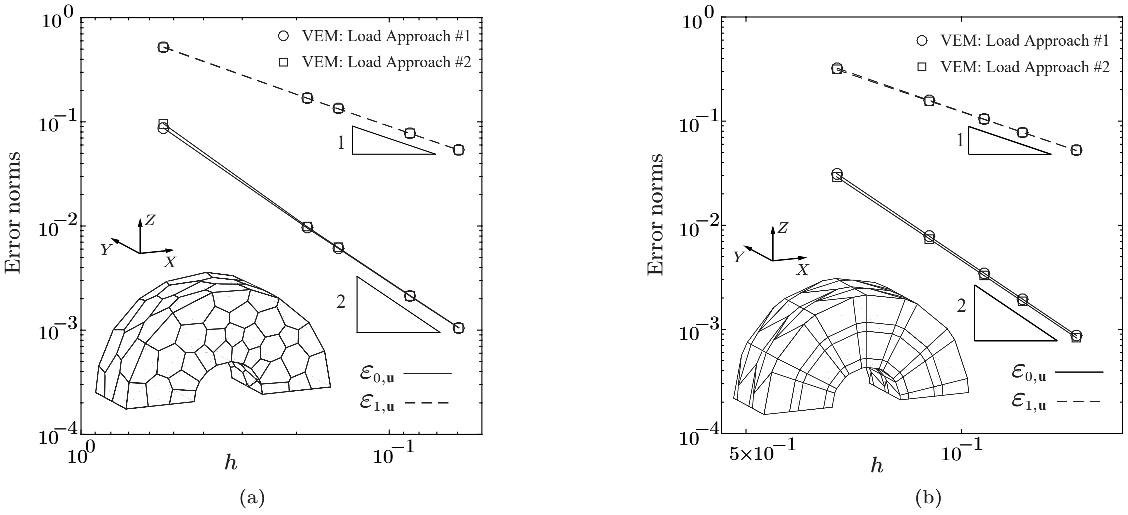


Figure 3.10: The convergence plots of the error norms as functions of the average mesh size are shown in (a) for the CVT meshes and (b) for the distorted hexahedral meshes.

the structured hexagonal meshes are also included for comparison purposes. In addition, an example of the deformed shape of the "zoo" mesh is shown in Fig. 3.12. We remark that since the "zoo" meshes contain non star-shaped elements, it is difficult to perform mixed FEM analysis on them, however, the VEM analysis can be naturally conducted. Again, each data point in the plots for the CVT mesh is an average of five results. It is observed from the figures that mixed VEM using both "Robust" and "Simple" approaches to compute J_E delivers results that are optimally convergent in terms of both displacement and pressure errors. While the level of accuracy for the results obtained using the "Robust" and "Simple" approaches are almost identical with the CVT and structure hexagonal meshes, the results obtained using the former approach are slightly more accurate than the ones obtained by the latter approach with the non star-shaped "zoo" meshes, especially in terms of the $\mathscr{H}^1$-type norm of the displacement errors. This demonstrates the advantage of the "Robust" approach with respect to the "Simple" approach when dealing with highly irregular and non-convex meshes. Moreover, the comparison between the VEM and FEM results suggests that they have a similar level of accuracy. Again, the mixed VEM use only one integration point per element and thus is more computationally efficient than the mixed FEM.

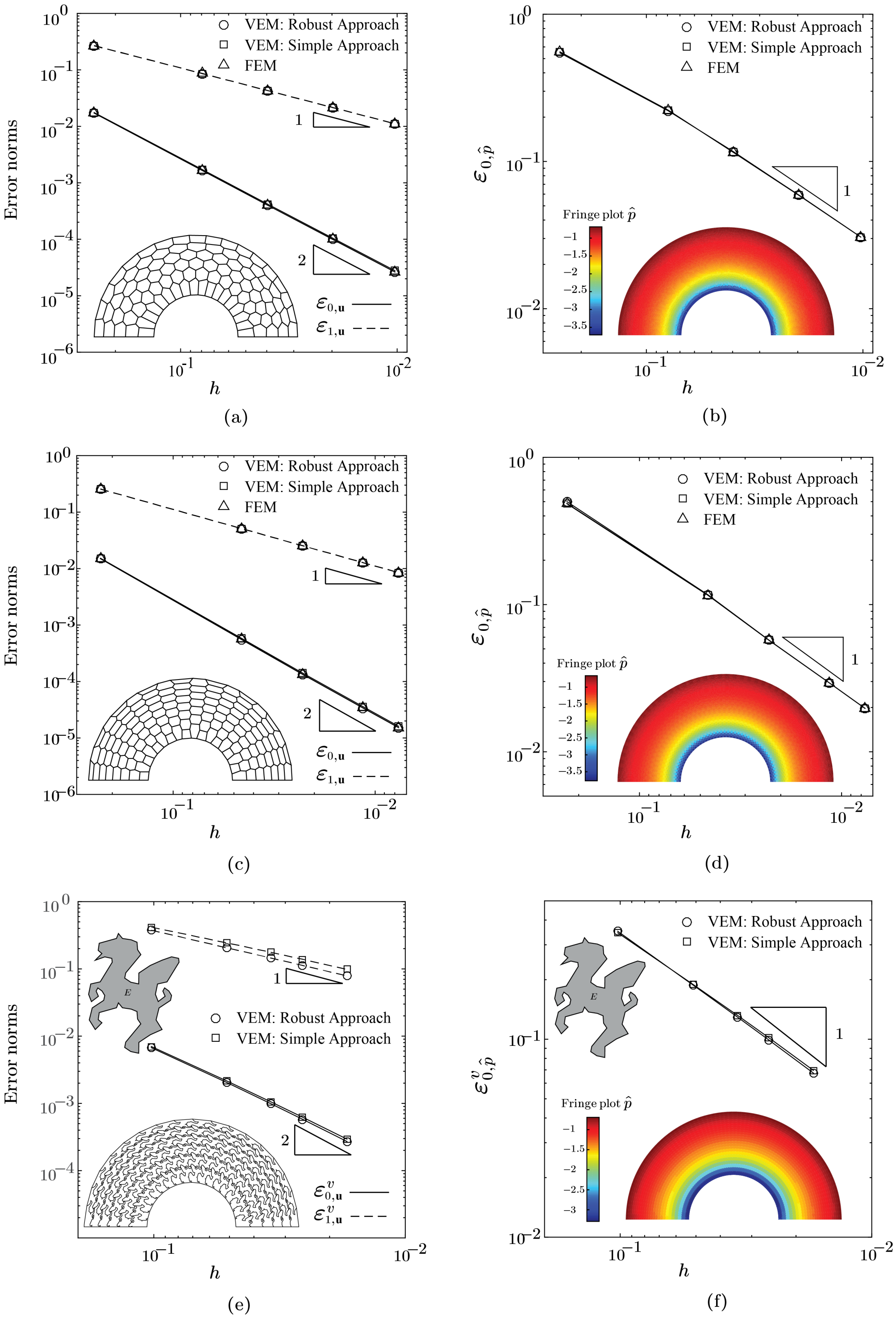


Figure 3.11: The convergence plots of the error norms as functions of the average mesh size for (a) displacement errors and (b) the pressure error in the CVT meshes, (c) the displacement errors and (d) the pressure error in the structured hexagonal meshes, and (e) the displacement errors and (f) the pressure error in the non star-shaped "zoo" meshes.

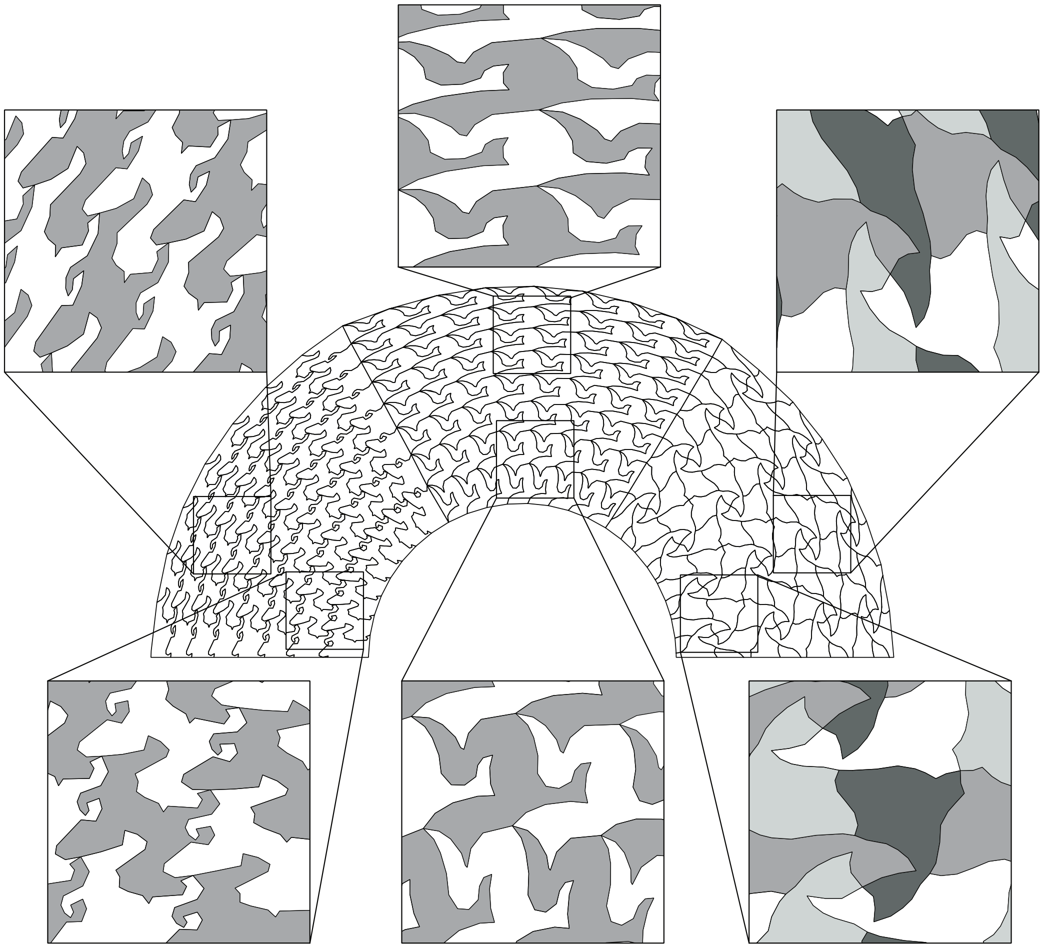


Figure 3.12: The final deformed configuration is shown for the “zoo” mesh. The “zoo” mesh contains 224 elements with 64 pegasus, 72 bird, and 88 fish elements. This deformed shape is obtained by the mixed VEM.

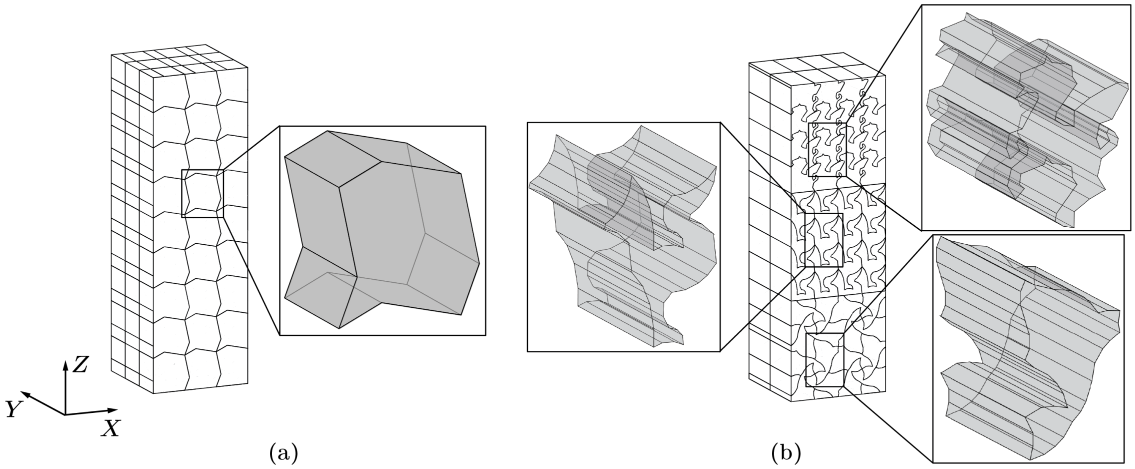


Figure 3.13: Examples for (a) an extruded octagonal meshes with 81 elements and (b) a non star-shaped "zoo" meshes with 128 elements. This "zoo" mesh is made up of 32 extruded pegasus elements, 48 extruded bird elements, and 48 extruded fish elements.

3D mixed VEM Our next step consists of performing numerical studies of the 3D mixed VEM with the original 3D problem shown in Fig. 3.3 (a). We note that, differently from the 2D case [72], the inf-sup condition in the 3D case has not been proven, but it can be numerical investigated as has been done in the 2D (polygonal) FEM context [2], [4]. Similarly to the displacement-based VEM case, CVT and distorted hexahedral meshes are considered, as shown in Figs. 3.4 (g) and (h), respectively. Notice that both types of meshes only contain convex elements, hence it is possible to treat them using the standard FEM with the Wachspress shape functions [30]. On the other hand, we also consider two sets of concave meshes, the extruded octagonal meshes and the extruded version of the non star-shaped "zoo" meshes consist of pegasus, bird and fish elements, as shown in Figs 3.13 (a) and (b) respectively. We note that while all the elements in the extruded octagonal meshes satisfy the assumption in the definition of the 3D local VEM space, i.e., all the faces of each element are star convex with respect to $\mathbf{X}_s^F = 1/m^F \sum_{j=1}^{m^F} \mathbf{X}_j^F$, all the elements in the extruded "zoo" meshes contain non star-shaped faces, which violates this assumption. Figures 3.14 and 3.15 show the displacement and pressure errors of the results obtained by the VEM with both the "Robust" and "Simple" approaches to compute J_E, together with illustrations of the deformed shapes and the fringe plots of the pressure field $\widehat{p}_h$. For the hexahedral meshes, VEM with both “Robust” and “Simple” approaches to compute J_E offers optimal convergence in the displacement errors. Yet, a slight degeneracy in the convergence of the pressure error occurs and their pressure fringe plots exhibit checkerboard modes. These indicate the occurrence of a mild numerical instability. For all the other meshes, optimal convergence in both displacement and pressure errors are observed for VEM with both approaches and the pressure fields are more smoothly distributed, suggesting numerical stability for those meshes. In particular, through the convergence observed in the results of the extruded “zoo” meshes, we highlight that our VEM formulation ensures convergence even though the meshes contain elements with non star-shaped faces (which violates our assumption in defining the local VEM space). Moreover, by comparing the results obtained from the “Robust” and “Simple” approaches to compute J_E, we remark that although both approaches appear to provide similarly accurate results for regular meshes (the hexahedral and CVT meshes), the “Robust” approach seems to produce more accurate results when the meshes become more irregular, e.g. the extruded “zoo” meshes.

3.7 Application example: elastomer filled with rigid inclusions

In this section, we deploy the proposed VEM framework to the study of the nonlinear elastic response of an elastomer filled with an isotropic distribution of rigid circular particles in 2D. Through this application example, we aim to demonstrate the performance of the proposed displacement-based and mixed VEM formulations in problems which involve heterogeneous and large localized deformation fields. In the meantime, the influences of various choices of stabilization techniques and different approaches to compute J_E on the performance of the VEM in such problems are investigated.

We consider the filled elastomer to be a periodic repetition of a unit cell that contains a random distribution of a large number of particles constructed by means of a random sequential adsorption algorithm [116], [144]. Motivated by the polydispersity in size of

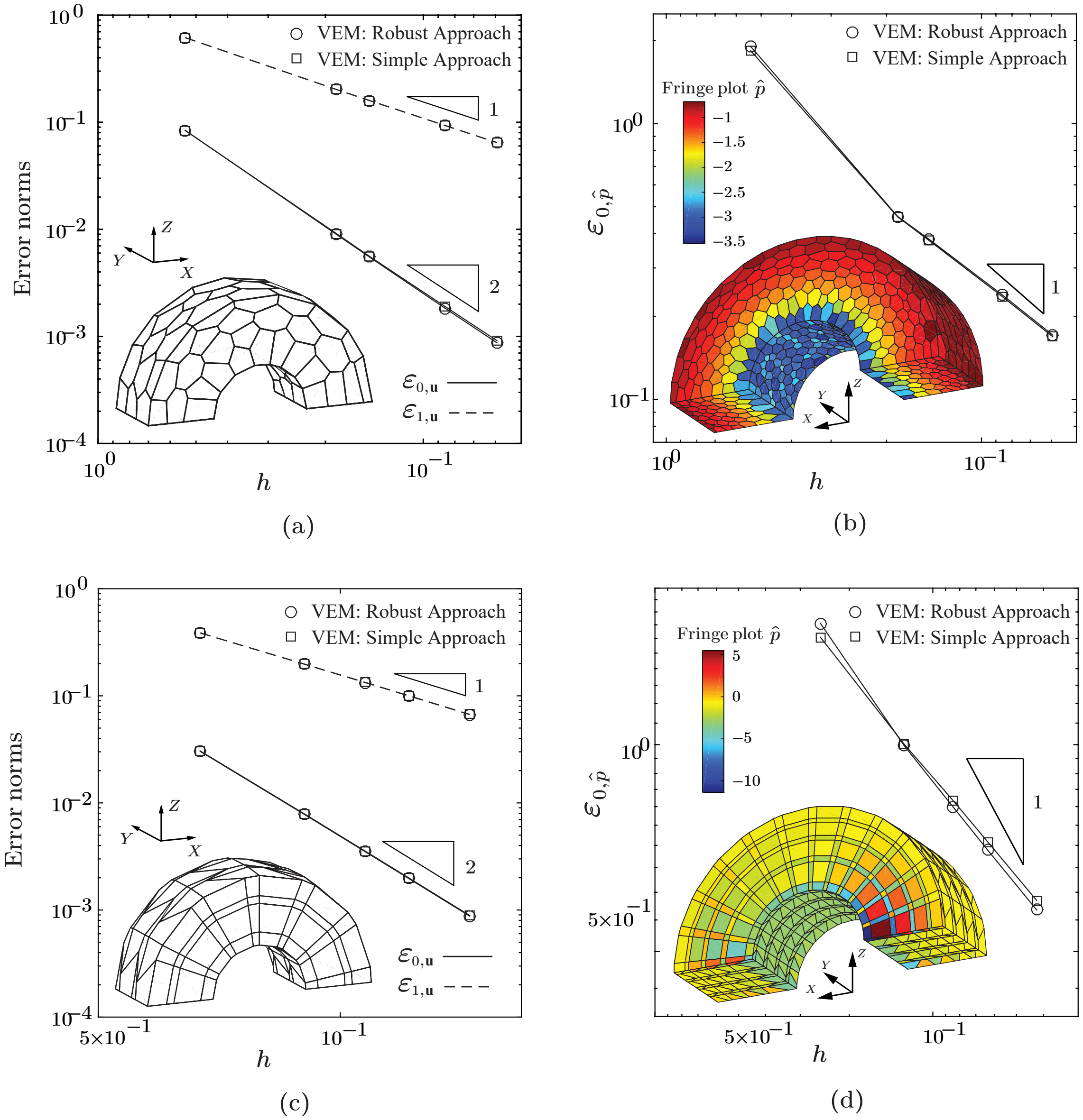
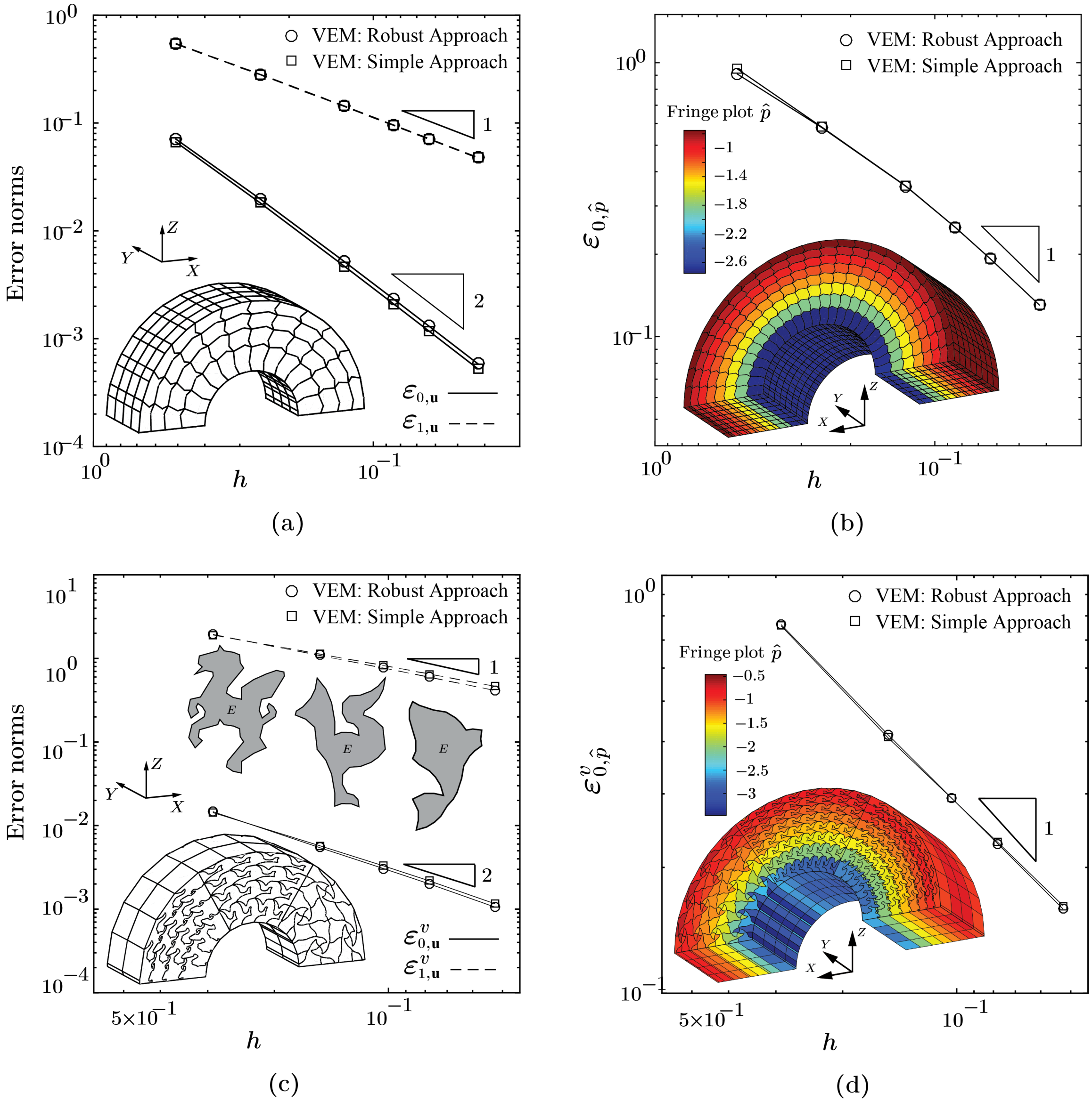


Figure 3.14: The convergence plots of the error norms as functions of the average mesh size for (a) the displacement errors and (b) the pressure error in the CVT meshes, (c) the displacement errors and (d) the pressure error in the hexahedral meshes.



(a)

(b)

(c)

(d)

Figure 3.15: The convergence plots of the error norms as functions of the average mesh size for (a) the displacement errors and (b) the pressure error in the extruded octagonal meshes, (c) the displacement errors and (d) the pressure error in the extruded "zoo" meshes.

typical fillers, we consider a particular case with three families of particles with radii

$$\left\{r^{(1)}, r^{(2)}, r^{3}\right\} = \{r, 0.75r, 0.5r\} \quad \text{with} \quad r = \left(\frac{c^{(1)}}{N^{(1)}\pi}\right)^{(1/2)}, \text{ and} \tag{3.98}$$

concentrations

$$\left\{c^{(1)}, c^{(2)}, c^{(3)}\right\} = \{0.5c, 0.3c, 0, 2c\}, \tag{3.99}$$

where $N^{(1)}$ is the number of particles with the largest radius $r^{(1)}$, and c stands for the total area fraction of particles. In the present example, we take $N^{(1)} = 20$ and $c = 0.35$. A realization of such a unit cell containing a total number of 75 particles at area fraction $c =$ 35% is shown in Fig. 3.16 (a). In addition, the Displacement periodic boundary condition in (2.72) is applied.

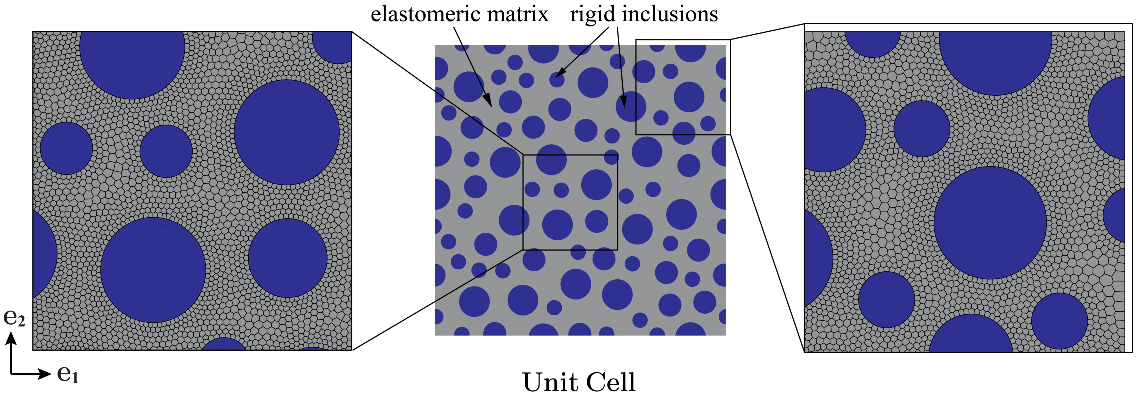


Figure 3.16: Illustrations of the unit cell, which consists of 75 polydispersed particles at a total area fraction of $c = 35\%$, and the details of the CVT mesh, which consists of 20,000 elements and 40,196 nodes.

Throughout this section, we make use of a CVT mesh as depicted in Fig. 3.16 (b), which is generated by `PolyMesher` [110]. The discretization contains a total of 20,000 elements and 40,196 nodes. To apply periodic displacement boundary conditions to this CVT mesh, we adopt the concept introduced in [2] of locally inserting additional nodes to achieve periodic nodal distributions on opposite boundaries of the mesh (a procedure that can be easily applied thanks to the flexibility of polygonal meshes). On the other hand, each particle in the CVT mesh is considered to be infinitely rigid by adopting the variational formulation proposed by Chi et al. [114], in which its presence is treated as a set of kinematic constraints on the displacement DOFs.

In the subsequent VEM and polygonal FEM simulations on the CVT mesh, the macroscopic deformation gradient is evaluated as

$$< \mathbf{F} > \doteq \frac{\sum_E |E| \left(\mathbf{I} + \mathbf{\Pi}_E^0 \nabla \mathbf{u}_h \right)}{|\Omega_h|}. \tag{3.100}$$

In addition, to quantitatively evaluate the performance of those simulations, we define the macroscopic energy of the filled elastomer in the following manner

$$< W > \doteq \frac{1}{|\Omega_h|} \sum_E \left[\Psi \left(\mathbf{I} + \mathbf{\Pi}_E^0 \nabla \mathbf{u}_h \right) + U \left(J_E(\mathbf{u}_h) \right) \right], \tag{3.101}$$

where J_E is evaluated using the “Robust” approach with expression (3.33). When the matrix is purely incompressible, the second term in the above expression is simply zero for all the elements in the mesh. We also monitor the relation between the macroscopic deformation gradient $< \mathbf{F} >$ and macroscopic first Piola-Kirchoff stress $< \mathbf{P} >$ in the VEM and polygonal FEM simulations by equivalently monitoring the displacement-traction relationship at the lower-right corner of the unit cell (the location where we apply displacement in our simulations).

3.7.1 Filled elastomers with neo-Hookean matrix

In this subsection, the matrix is assumed to be compressible neo-Hookean material, which is described by the stored-energy function

$$W(\mathbf{F}) = \frac{\mu}{2} \left(\overline{\mathbf{F}} : \overline{\mathbf{F}} - 3 \right) + \frac{\kappa}{2} (\det \mathbf{F} - 1)^2, \tag{3.102}$$

where μ and κ are the initial shear and bulk moduli. Throughout this subsection, the initial shear modulus is taken to be $\mu = 1$, and the filled elastomer is assumed to be subjected to uniaxial tension, whose macroscopic deformation $< \mathbf{F} >$ gradient has the form $< \mathbf{F} >= \lambda_1 \mathbf{e}_1 \otimes \mathbf{e}_1 + \lambda_2 \mathbf{e}_2 \otimes \mathbf{e}_2$ where λ_1 and λ_2 are the macroscopic stretches with λ_1 in the tensile direction.

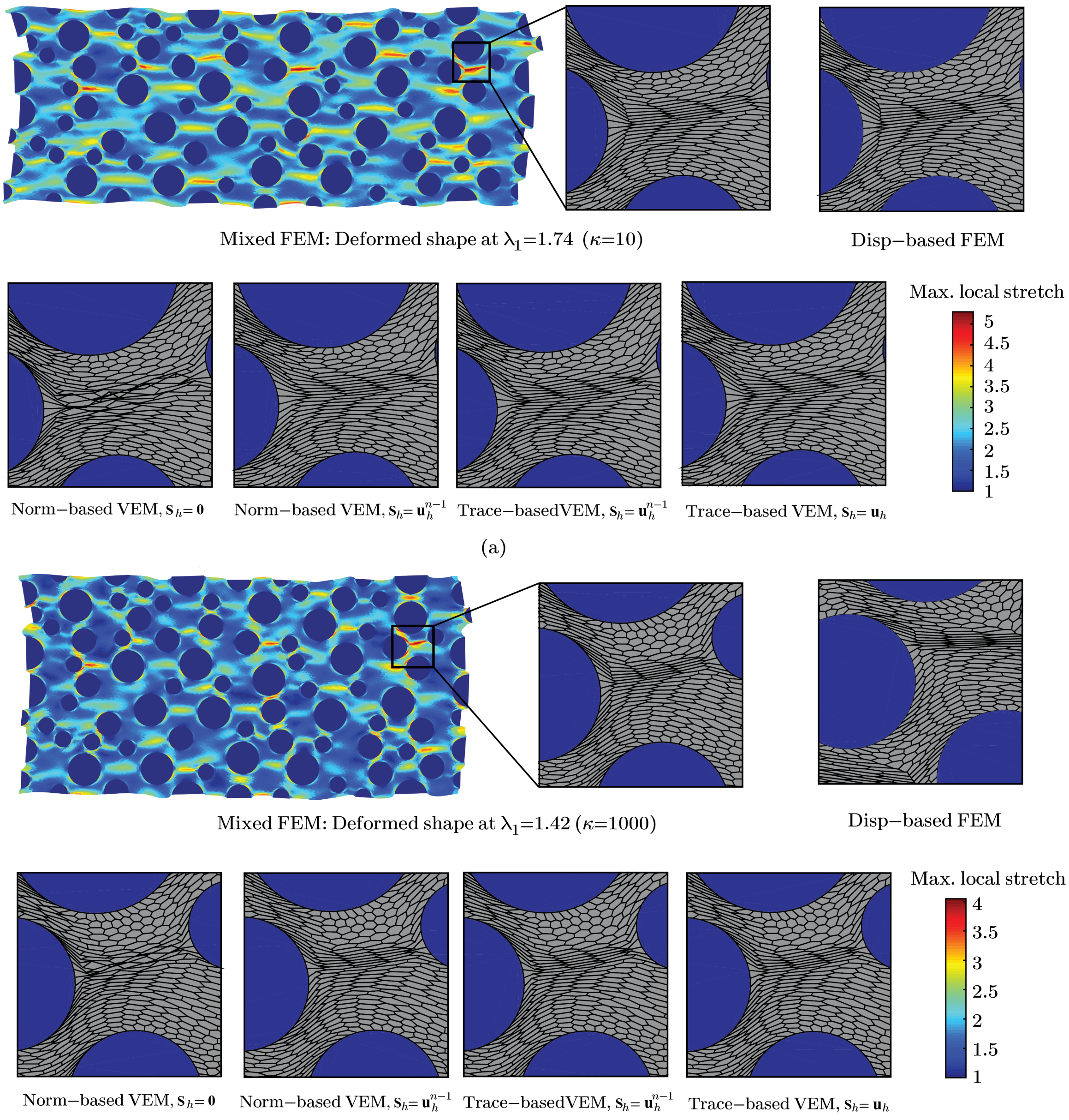
We first study the influence of various choices of the stability parameter $\alpha_E(\mathbf{s}_h)$ on the performance of the displacement-based VEM. More specifically, we investigate the four choices of $\alpha_E(\mathbf{s}_h)$ as shown in Table 3.3. We assume two values of the initial bulk moduli, $\kappa = 10$ and $\kappa = 1000$, which represent compressible and nearly incompressible materials respectively. For all the cases, the target macroscopic stretch λ_1 is set as 2.

Table 3.3: VEM stability parameter $\alpha_E(\mathbf{s}_h)$.

Operator	Norm: Eq. (3.58)		Trace: Eq. (3.59)	
$\mathbf{s}_h$	$\mathbf{0}$	$\mathbf{u}_h^{n-1}$	$\mathbf{u}_h^{n-1}$	$\mathbf{u}_h$

As a qualitative comparison, Figs 3.17 (a)–(b) depict the detailed views of the unit cell at the same applied macroscopic stretches using different choices of $\alpha_E(\mathbf{s}_h)$ in Table 3.3. These detailed views are taken from the deformed configurations at $\lambda_1 = 1.74$ for $\kappa = 10$ and from those at $\lambda_1 = 1.42$ for $\kappa = 1000$. As a reference, the detailed views obtained from the standard displacement-based and mixed FEM are also included. Additionally, Figs. 3.18 (a)-(d) show the comparison of macroscopic responses (energy and relevant component of stress) as functions of the applied stretch λ_1 between the various choices of $\alpha_E(\mathbf{s}_h)$, with the ones obtained by the displacement-based FEM represented by the dashed lines.

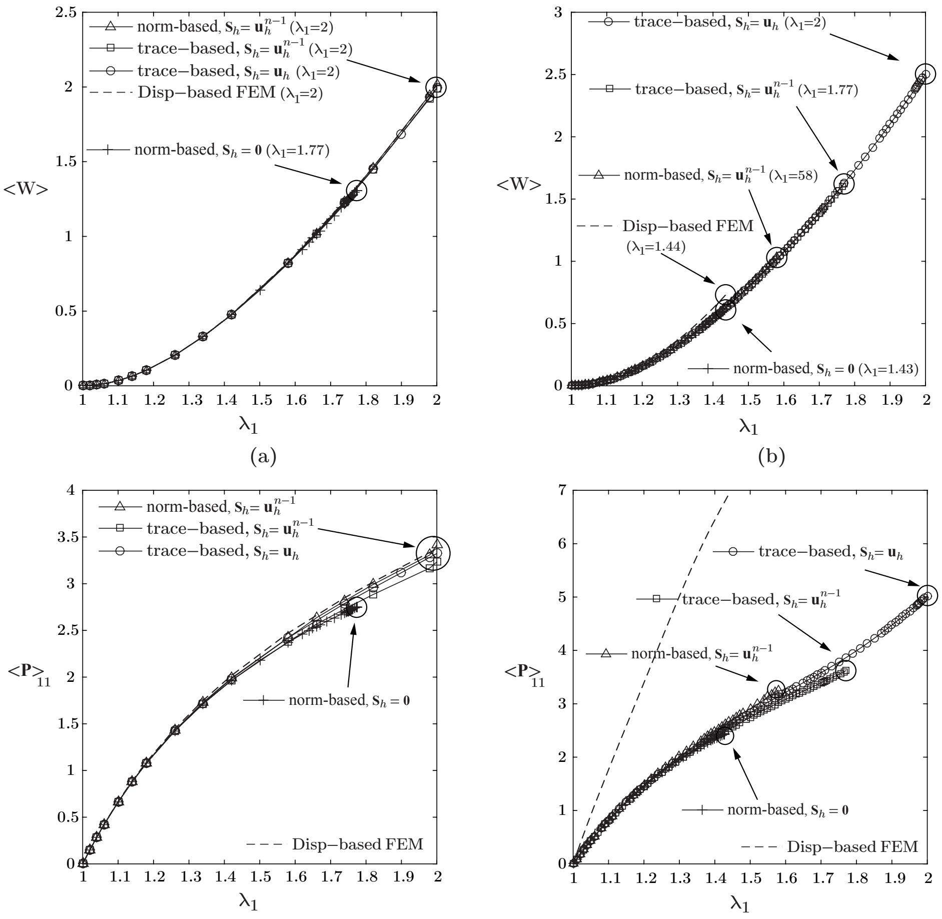
Several immediate observations can be made. First, the displacement-based FEM apparently suffers from volumetric locking when the matrix becomes nearly incompressible, i.e., $\kappa = 1000$, as it produces over stiff macroscopic energy and stress. On the other hand, the proposed displacement-based VEM formulation appears free of volumetric locking and produces more reasonable macroscopic energy and stress. The displacement-based VEM



(a)

(b)

Figure 3.17: Detailed views of the unit cell at the same applied macroscopic stretches obtained using the displacement-based and mixed FEM, and the displacement-based VEM using different choices of $\alpha_E(\mathbf{s}_h)$ in Table 3.3: (a) the case of $\kappa = 10$ at $\lambda_1 = 1.74$ and (b) the case of $\kappa = 1000$ at $\lambda_1 = 1.42$.



(a)

(b)

(c)

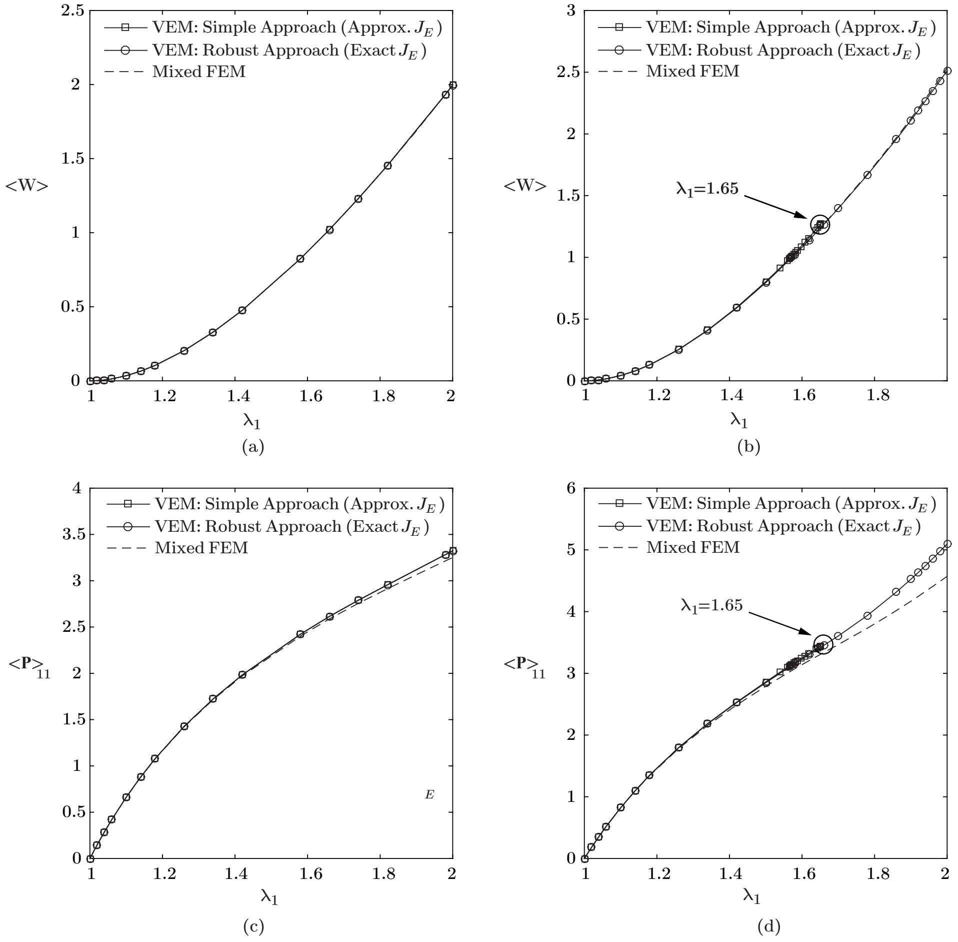
(d)

Figure 3.18: Comparison of the macroscopic responses obtained by the displacement-based VEM with various stabilization techniques: (a) Macroscopic energy as a function of the applied stretch λ_1 for the case of $\kappa = 10$. (b) Macroscopic energy as a function of the applied stretch λ_1 for the case of $\kappa = 1000$. (c) Macroscopic stress as a function of the applied stretch λ_1 for the case of $\kappa = 10$. (d) Macroscopic stress as a function of the applied stretch λ_1 for the case of $\kappa = 1000$.

with all the stabilization choices predicts almost identical macroscopic energies, although we note that the norm-based stabilization with $\mathbf{s}_h = \mathbf{u}_h^{n-1}$ and trace-based stabilization with $\mathbf{s}_h = \mathbf{u}_h$ produce a slightly stiffer macroscopic stress than the other two. Moreover, the detailed views in Figs. 3.17 indicate that the norm-based stabilization with $\mathbf{s}_h = \mathbf{0}$ leads to hourglass modes in the element at high deformation levels (the elements that are plotted in red). As we have discussed in Section 4, this is a consequence of the under-stabilizations of $\alpha_E(\mathbf{0})$ in those elements. The detailed views of other choices, on the other hand, do not contain hourglass modes in those elements and all are qualitatively similar to the one obtained with the mixed FEM.

Another major interest is the ability of the VEM to model large localized and heterogeneous deformations. To that end, we quantify the maximum global stretch λ_1 that the displacement-based VEM (with each stabilization choice) can achieve before non-converge occurs in the Newton-Raplson process; in general, the larger global stretch the unit cell reaches, the larger localized deformation it induces. Based on this measure, it is apparent that the norm-based stabilization with $\mathbf{s}_h = \mathbf{0}$ is the worst choice for both cases of $\kappa = 10$ and $\kappa = 1000$. With this choice, the unit cell reaches significantly less maximum global stretch than the other choices. The same happens with the displacement-based FEM due to the volumetric locking. On the other hand, as the incompressibility level of the matrix increases from $\kappa = 10$ to $\kappa = 1000$, we notice the advantage of using the trace-based stabilization for VEM, as well as choosing $\mathbf{s}_h = \mathbf{u}_h$ over $\mathbf{s}_h = \mathbf{u}_h^{n-1}$, in achieving larger global stretches. We note that since the displacement-based VEM is equivalent to the mixed VEM as discussed in Section 4, the above observations and discussions also apply to the mixed VEM formulation.

Our second study investigates the performance of the mixed VEM using different approaches to compute J_E. Both the "Robust" approach (i.e., Eqs. (3.36) and (3.38)) and the "Simple" approach (i.e., Eqs. (3.45)–(3.47)) are considered. Similarly to the preceding study, we choose two values of initial bulk moduli, $\kappa = 10$ and $\kappa = \infty$, representing compressible and purely incompressible matrices. In this study, we utilize the trace-based stabilization with $\mathbf{s}_h = \mathbf{u}_h$ and the target global stretch is set as $\lambda_1 = 2$.



(a)

(b)

(c)

(d)

Figure 3.19: Comparison of the macroscopic responses obtained by the mixed VEM with the two approaches to compute J_E: (a) Macroscopic energy as a function of the applied stretch λ_1 for the case of $\kappa = 10$. (b) Macroscopic energy as a function of the applied stretch λ_1 for the case of $\kappa = \infty$. (c) Macroscopic stress as a function of the applied stretch λ_1 for the case of $\kappa = 10$. (d) Macroscopic stress as a function of the applied stretch λ_1 for the case of $\kappa = \infty$.

We plot in in Figs. 3.19 (a)–(d) the macroscopic responses (macroscopic energy and relevant component of stress) as functions of the applied stretch λ_1 for the cases of $\kappa = 10$ and $\kappa = \infty$. It is noted from those figures that although the two approaches yield similar macroscopic responses, using the exact expression for J_E (the “Robust” approach) helps the unit cell to reach a significantly larger global stretch than using the approximation of

J_E (the "Simple" approach) when the material is nearly or purely incompressible.

We conclude from the above studies that the trace-based stabilizations generally yield more accurate macroscopic responses than the norm-based ones (when compared with the results obtained from the mixed FEM). Moreover, choosing $\mathbf{s}_h = \mathbf{u}_h$ instead of $\mathbf{s}_h = \mathbf{u}_h^{n-1}$ (although it seems to produce slightly stiffer macroscopic stress), as well as using the "Robust" approach to compute J_E, are more computationally demanding but typically help the unit cell reach larger global stretches, especially as the matrix is approaching the incompressible limit. Consequently, for the remainder of this Section, the trace-based stabilization with $\mathbf{s}_h = \mathbf{u}_h$ and the "Robust" approach to compute J_E are adopted. We finally remark that, for such problems, linear and quadratic triangular and quadrilateral (conforming) elements are found to lead to a loss of convergence at much smaller global stretches when compared to the polygonal methods studied above, especially when the matrix phase is nearly or purely incompressible (see, e.g., Section 5.1 of [22]).

3.7.2 Filled elastomer with matrix described by other constitutive models

In the sequel, we adopt the mixed VEM formulation to study the nonlinear elastic responses of the filled elastomer when its matrix is characterized by other material models. The main purpose of this study is to thoroughly compare the performance of the mixed VEM with the mixed FEM for various material models. In particular, we consider the incompressible Mooney-Rivlin model

$$W(\mathbf{F}) = \begin{cases} C_1\,[I_1(\mathbf{F}) - 3] + C_2\,[I_2(\mathbf{F}) - 3] & \text{if } \det\mathbf{F} = 1 \\ +\infty & \text{otherwise} \end{cases}, \tag{3.103}$$

and the incompressible model utilized in [116], [117] to describe a typical silicone rubber

$$W(\mathbf{F}) = \begin{cases} \frac{3^{1-\alpha_1}}{2\alpha_1}\mu_1\left[(I_1(\mathbf{F}))^{\alpha_1} - 3^{\alpha_1}\right] + \frac{3^{1-\alpha_2}}{2\alpha_2}\mu_2\left[(I_1(\mathbf{F}))^{\alpha_2} - 3^{\alpha_2}\right] & \text{if } \det\mathbf{F} = 1 \\ +\infty & \text{otherwise} \end{cases} . \tag{3.104}$$

In the Mooney-Rivlin model, we choose the material parameters to be $C_1 = 0.3$ and $C_2 = 0.2$, resulting in an initial shear modulus $\mu = 2(C_1 + C_2) = 1$. The material parameters for the typical silicone rubber are taken to be $\alpha_1 = 3.837$, $\alpha_2 = 0.559$, $\mu_1 = 0.032$ and $\mu_2 = 0.3$ with the initial shear modulus being $\mu = \mu_1 + \mu_2 = 0.332$. Two loading conditions are considered: i) uniaxial tension whose macroscopic deformation gradient $< \mathbf{F} >$ is of the form $< \mathbf{F} >= \lambda \mathbf{e}_1 \otimes \mathbf{e}_1 + \lambda^{-1}\mathbf{e}_2 \otimes \mathbf{e}_2$ and ii) simple shear whose macroscopic deformation gradient $< \mathbf{F} >$ is of the form $< \mathbf{F} >= \mathbf{I} + \gamma \mathbf{e}_1 \otimes \mathbf{e}_2$, where λ and γ are the applied global stretch and shear.

Under both loading conditions, Fig. 3.20 depicts the final deformed configurations of the unit cells with Mooney-Rivlin matrix obtained by the mixed VEM and mixed FEM at their respective maximum global deformation states. Similarly, Fig 3.21 shows the deformed configurations of the unit cell with typical silicone rubber matrix obtained by the mixed VEM and mixed FEM at their respective maximum global deformation states. The color scale in each configuration corresponds to the maximum principal stretch of each element, with those having a value of 5 and above plotted red. Additionally, the macroscopic responses predicted by the VEM and FEM (macroscopic energies and relevant components of macroscopic stress) as functions of the applied stretch λ or shear γ are shown in Figs 3.22 and Figs 3.23, respectively for unit cells with Mooney-Rivlin matrix and typical silicone rubber matrix. In the plots, we also show the deformed configurations of the unit cell at the same levels of global deformations obtained by VEM and FEM, with the elements whose maximum principal stretches greater than 5 plotted read.

Several observations can be made from Figs 3.20–3.23. First, for both material models considered, the results obtained from the mixed VEM and mixed FEM are in good

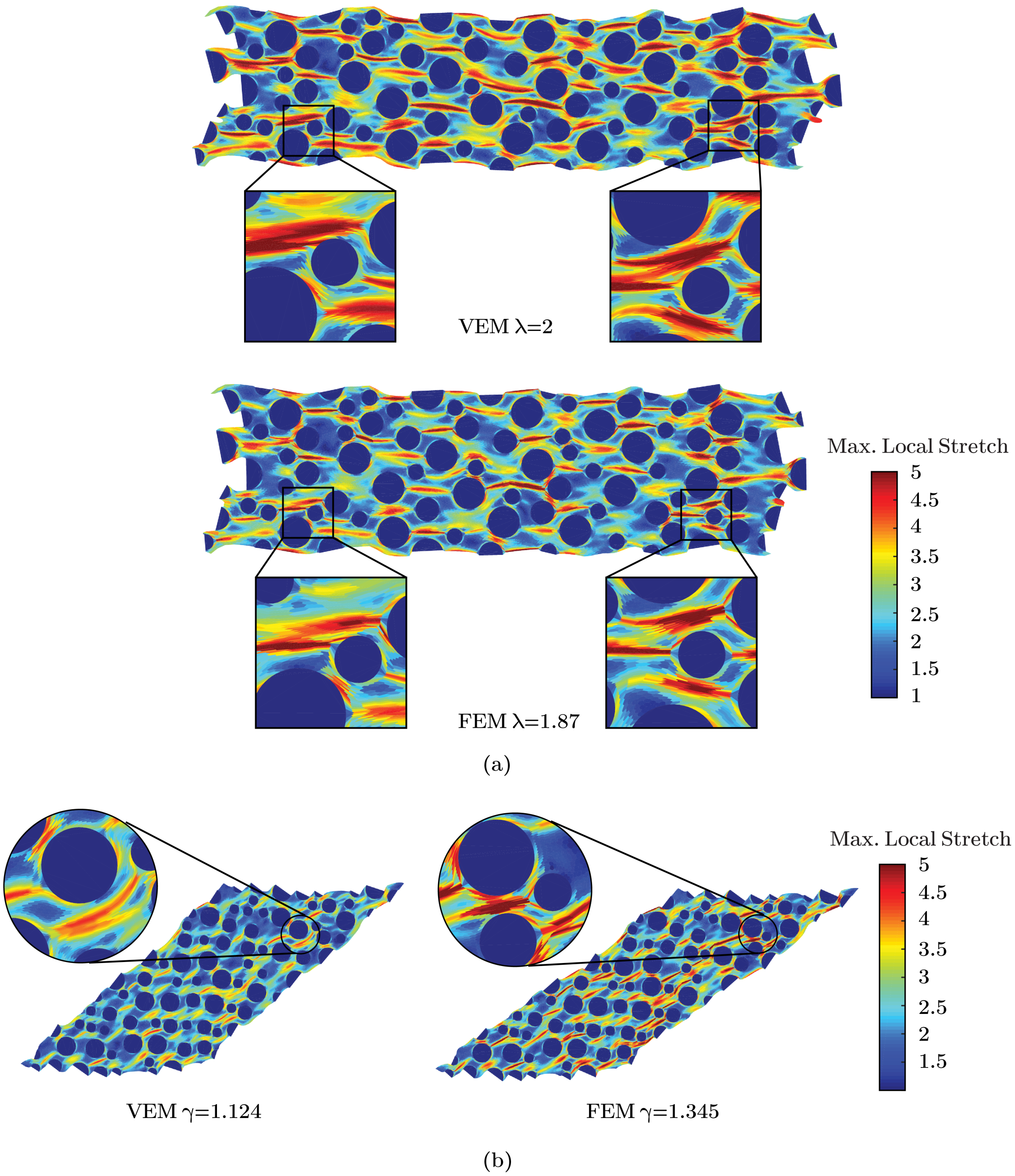


Figure 3.20: The maximum deformed configurations of the unit cell obtained by the mixed VEM and mixed FEM for the Mooney-Rivlin matrix under (a) uniaxial tension and (b) simple shear.

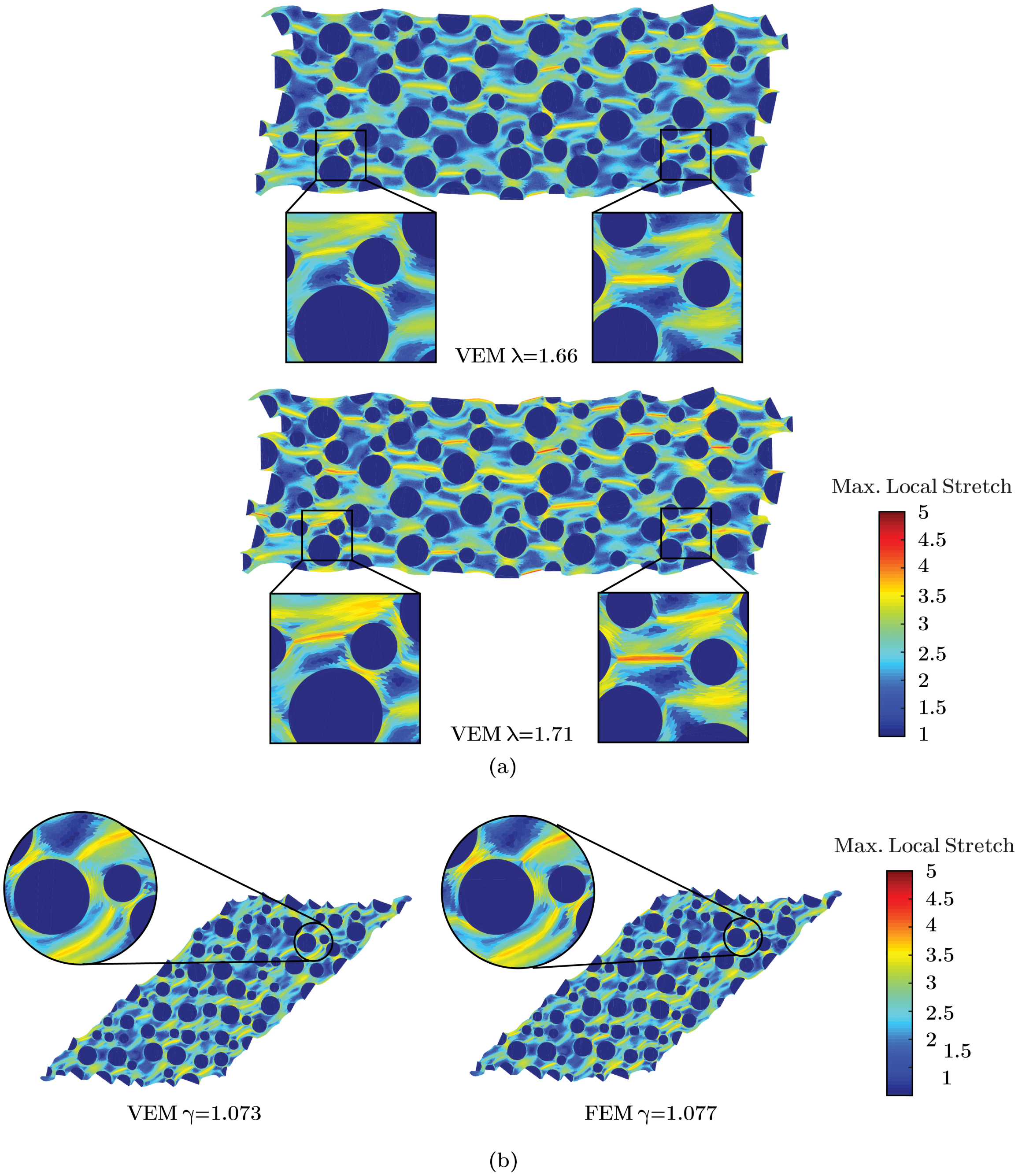
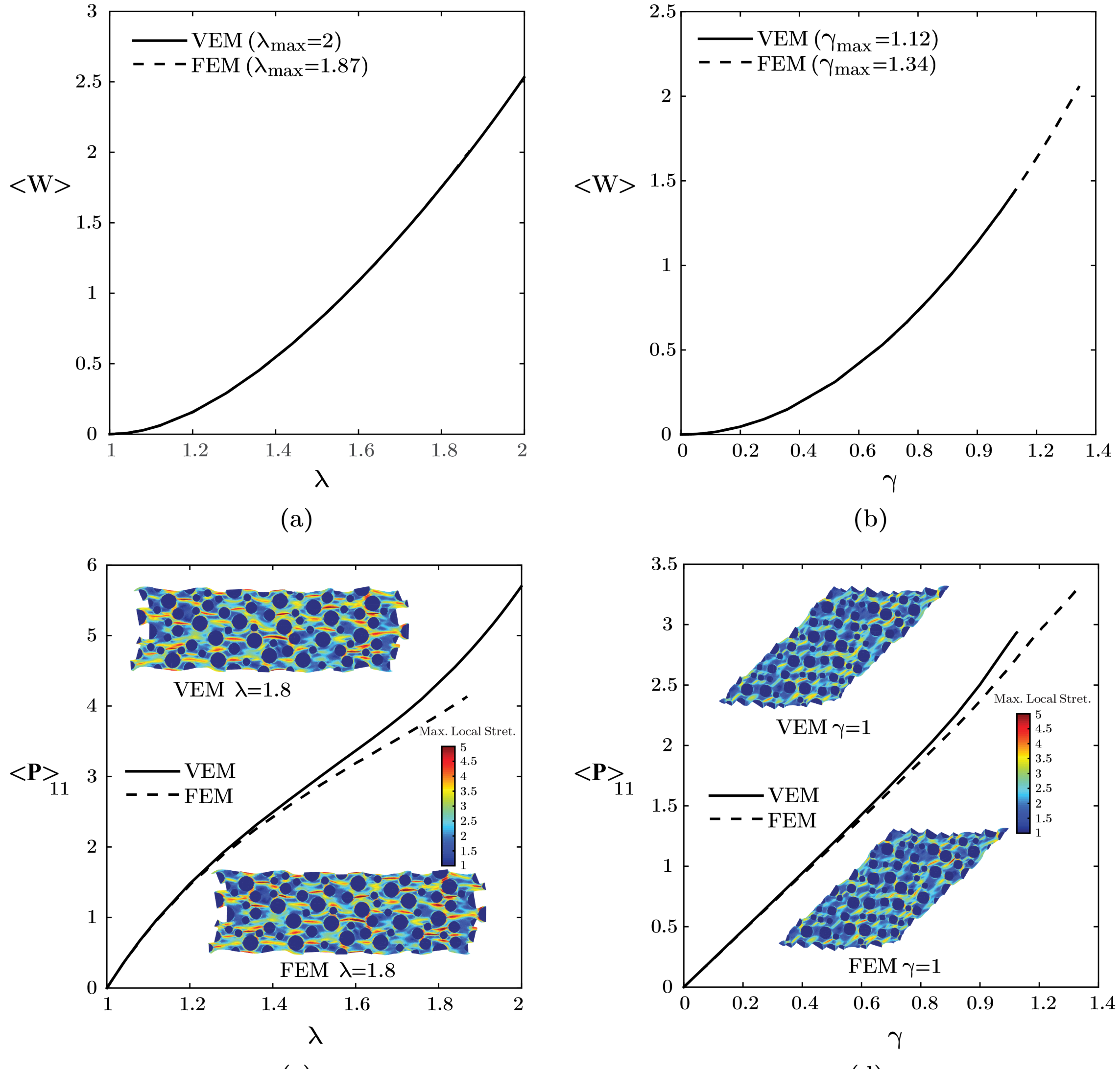


Figure 3.21: The maximum deformed configurations of the unit cell obtained by the mixed VEM and mixed FEM for the typical silicone rubber matrix under (a) uniaxial tension and (b) simple shear.



(a)

(b)

(c)

(d)

Figure 3.22: Comparison of the macroscopic responses obtained by the mixed VEM and mixed FEM for Mooney-Rivlin matrix: (a) macroscopic energy as a function of the applied stretch λ under uniaxial tension; (b) macroscopic energy as a function of the applied shear γ under simple shear; (c) macroscopic stress as a function of the applied stretch λ under uniaxial tension; (d) macroscopic stress as a function of the applied shear γ under simple shear.

qualitative and quantitative agreements. As shown in Figs 3.22 and 3.23, the deformed configurations obtained by the VEM and FEM share similar patterns at the same global deformations levels. The macroscopic responses predicted by the VEM and FEM also match reasonably well, especially for the macroscopic energy. However, it is worthwhile noting that, similarly to the case of neo-Hookean matrix presented in the previous subsection, we also observe stiffer responses in the macroscopic stresses obtained by VEM than the ones by FEM for the unit cell with Mooney-Rivlin matrix. Furthermore, by comparing the maximum deformation levels reached by the unit cells modeled respectively by the VEM and FEM, it appears that the VEM and FEM have similar capabilities of modeling large localized deformations. As a further investigation, we summarize in Tables 3.4 and 3.5 the maximum global stretch/shear reached by the VEM and FEM for both material models and the corresponding maximum local principal stretches. It is interesting to note that for the unit cell with Mooney-Rivlin matrix, although the VEM yields a larger global stretch in uniaxial tension, the corresponding maximum local principal stretches in the FEM mesh is higher. We again underline that, on this same test, standard triangular and quadrilateral FEM would reach a much smaller global stretch when compared to the polygonal methods under study [2], [22].

Table 3.4: Summary of maximum principal stretches of among all the elements in the mesh for the unit cell considering Mooney-Rivlin material and different loading conditions.

		Mooney-Rivlin matrix	
		Uniaxial tension	Simple shear
Max. global stretch/shear	VEM	2	1.124
	FEM	1.870	1.345
Max. principal stretch	VEM	6.559	4.650
	FEM	7.015	8.022

3.8 Concluding remarks

This chapter introduces a VEM framework for two and three dimensional finite elasticity problems. Two VEM formulations are presented, which adopt a displacement-based

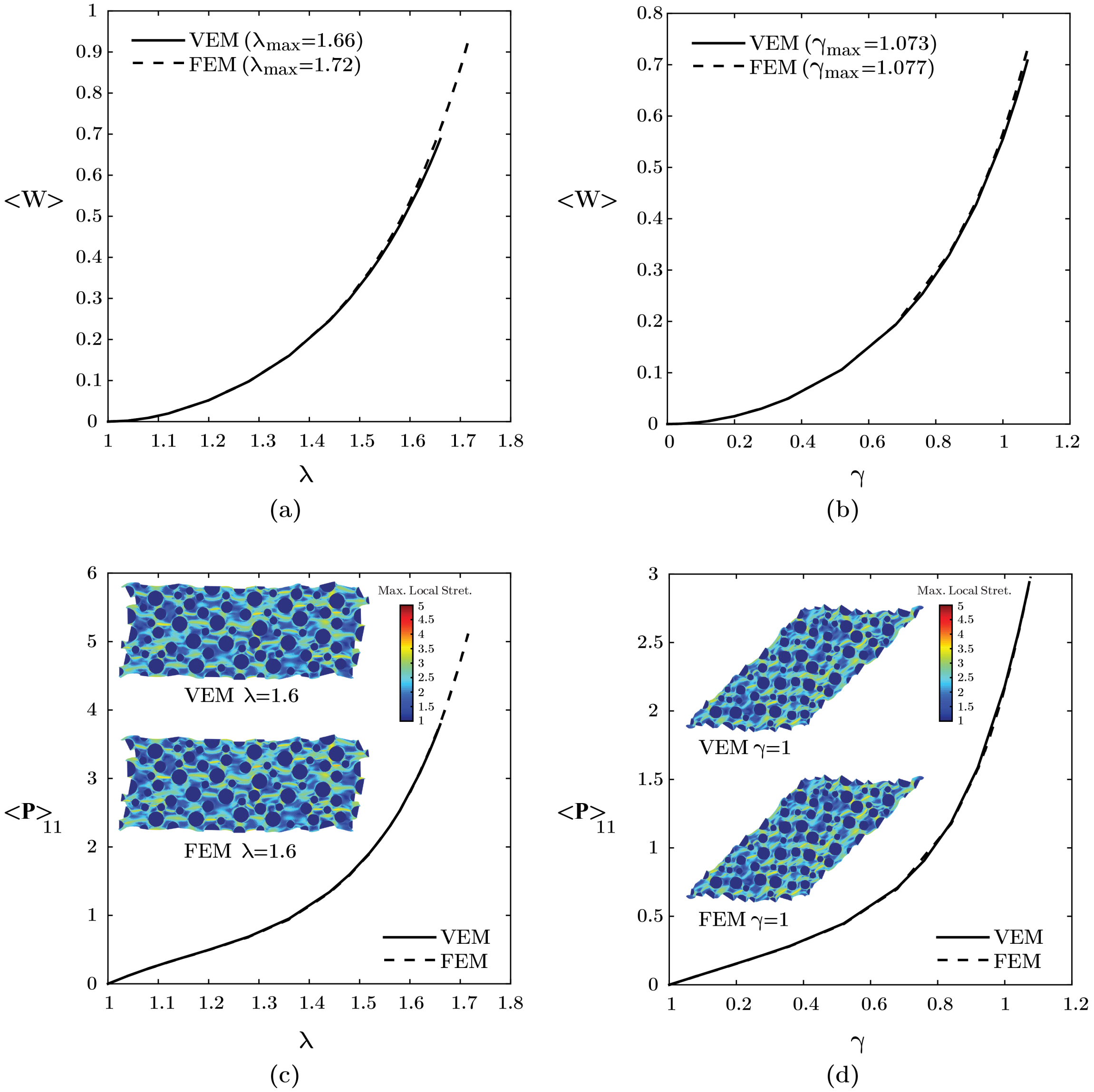


Figure 3.23: Comparison of the macroscopic responses obtained by the mixed VEM and mixed FEM for typical silicone rubber matrix: (a) macroscopic energy as a function of the applied stretch λ under uniaxial tension; (b) macroscopic energy as a function of the applied shear γ under simple shear; (c) macroscopic stress as a function of the applied stretch λ under uniaxial tension; (d) macroscopic stress as a function of the applied shear γ under simple shear.

Table 3.5: Summary of maximum principal stretches of among all the elements in the mesh for the unit cell considering a typical silicone rubber and different loading conditions.

		Typical silicone rubber matrix	
		Uniaxial tension	Simple shear
Max. global stretch/shear	VEM	1.660	1.073
	FEM	1.716	1.077
Max. principal stretch	VEM	3.943	3.737
	FEM	4.384	3.903

and a two-field mixed variational principles respectively. The displacement-based VEM formulation appears to be free of volumetric locking as the material becomes nearly incompressible, at least for the proposed set of meshes. By construction, the proposed VEM formulations are able to efficiently handle a more general class of polygonal and polyhedral meshes than the standard FEM, including the ones with non-convex elements. Several numerical studies are presented, which confirm the convergence and accuracy of the VEM formulations. In particular, for three-dimensional problems, our numerical studies have further shown that the VEM formulations appear to produce convergent results even for meshes containing non-star shaped elements, which makes the VEM formulation even more forgiving with respect to the quality of the mesh. We also show that, according to the definition of VEM spaces, closed-form expressions for the exact average volume changes over each polygonal/polyhedral elements can be derived. Those closed-form expressions render the VEM more accurate and robust, especially for irregular meshes, such as those containing elements with non star-shaped faces and in problems that involve large heterogeneous and localized deformations. Furthermore, different constructions of the loading terms are discussed and various stabilization strategies are studied, which are shown to have a significant influence on the performance of the VEM formulations in finite elasticity problems, especially those involving large localized and heterogeneous deformations fields. A stabilization scheme is further proposed in this chapter for isotropic materials, which is based on the trace of the material tangent modulus tensor. We deploy the proposed VEM formulations to the study of the nonlinear elastic response of a filled elastomer in 2D and demonstrate that they are able to capture large localized deformation fields in such problems.

The aforementioned chapter indicates that the VEM offers room for novel developments in nonlinear mechanics. We remark that several extensions are of interest, for instance, studies of the performance of VEM in capturing physical instabilities and development of more advanced stabilization schemes for finite elasticity problems.

CHAPTER 4
A SIMPLE AND EFFECTIVE GRADIENT RECOVERY SCHEME AND A POSTERIORI ERROR ESTIMATOR FOR THE VIRTUAL ELEMENT METHOD (VEM)

This chapter introduces a general recovery-based *a posteriori* error estimation framework for the Virtual Element Method (VEM) of arbitrary order on general polygonal/polyhedral meshes. The framework consists of a gradient recovery scheme and *a posteriori* error estimator based on the recovered displacement gradient. A skeletal error, which accurately mimics the behavior of the L^2 error of the displacement gradient by only sampling the displacement gradient on the mesh skeleton, is introduced. Through numerical studies on various polygonal/polyhedral meshes, we demonstrate that the proposed gradient recovery scheme can produce considerably more accurate displacement gradient than the original VEM solutions, and that the *a posteriori* error estimator is able to accurately capture both local and global errors without the knowledge of exact solutions.

4.1 Introduction

The VEM is a recent generalization of the finite element method (FEM) that is capable of efficiently handling general polygonal/polyhedral meshes [37]. This feature makes VEM a suitable framework for mesh adaptations (e.g. adaptive refinement and coarsening). To realize this potential, we propose a general recovery-based *a posteriori* error estimation for VEM of arbitrary order on general polygonal/polyhedral meshes, and demonstrate the idea in the context of linear elasticity. More specifically, we first introduce an efficient patch-based gradient recovery scheme for VEM, which reconstructs a more accurate displacement gradient field by least square fitting the displacement degrees of freedom (DOFs) over each selected patch of elements in the mesh. Based on the recovered gradient, we further introduce a simple yet effective recovery-based a posteriori error estimator. *To avoid explicit constructions of the VEM basis functions, this error estimator evaluates the displacement error on the skeleton of the mesh*, which mimics the behavior of the L^2 norm of the displacement gradient error. We conduct thorough numerical studies to assess the performance of the proposed gradient recovery scheme and error estimator. Though the numerical examples, the proposed gradient recovery scheme and error estimator are shown to be accurate on various polygonal/polyhedral meshes and with different types of displacement solutions.

The remainder of this chapter is organized as follows. Section 2 provides motivations for the chapter and summarizes related work in both VEM and FEM literature. Section 3 reviews the VEM framework for 2D and 3D linear elasticity problems. In Section 4, we introduce the gradient recovery scheme for lower- and higher-order VEM, and *a posteriori* error estimator based on the recovered displacement gradient. In Section 5, some theoretical estimates of the recovered displacement gradient, which provide insights into the proposed gradient recovery scheme, are provided. Several numerical assessments are presented in Section 5 to demonstrate the accuracy of the gradient recovery scheme and *a posteriori* error estimator on various polygonal/polyhedral meshes and with different displacement solution types. Section 6 contains several concluding remarks and future research directions.

4.2 Motivation and related work

Interest in numerical methods that can handle polygonal/polyhedral meshes has been growing in fields of mathematics and engineering, see [2], [4], [13], [15], [17], [20]–[22], [96], [103], [110], [145], [146] for a minimal sample of references. Among them, VEM is an emerging method, first introduced in [37], as a generalization of FEM in the family of Galerkin methods. In the VEM, the basis functions of the local space are defined implicitly through a suite set of partial differential equations (PDEs). Unlike FEM, this set of PDEs is never solved throughout the approximation. Instead, we apply integration by parts to compute suitable projections of the basis functions on to polynomials. [37], [38]. Those projections are then used in the VEM approximation to ensure its consistency together with a suitable stabilization term, which is needed in order to avoid the appearance of hourglass modes. As a result, only numerical quadratures for polynomials (and not for more complex functions) are required in the VEM. These unique features allow the VEM to handle general polygonal/polyhedral meshes (including the non-convex ones [39], [40]) and to construct various types of elements, including $H(\mathrm{div})$ and $H(\mathrm{curl})$ conforming elements [41]. The VEM has undergone substantial developments and has been successfully applied to a wide range of problems. For conciseness, we only focus on the literature of VEM for structural mechanics here. In structural mechanics, the VEM has been introduced for linear elasticity problems [26], [42], [43], small deformation nonlinear elastic and inelastic problems [44]–[47], finite deformation elasticity [39], [53] and elasto-plasticity problems [54], plate bending problems [48]–[51] and contact problems [52].

Among various features of VEM, the flexibility in dealing with general polygonal and polyhedral meshes makes it appealing for adaptivity. For example, by introducing hanging nodes, adaptive mesh refinement strategies can be made more efficient with polygonal/polyhedral elements because it only requires local modifications to the mesh [2]. On the other hand, the shape generality of polygonal/polyhedral elements (especially the concave ones) enables easier element agglomeration schemes for mesh coarsening [55], [56]. To realize the full potential of VEM in adaptivity, development of *a posteriori* error estimator is essential – this is the focus of this chapter.

A posteriori error estimation is a classic topic in FEM with a vast literature, and it is typically categorized into many types: here we consider the recovery-based error estimation type, see e.g. [147]–[154]. For more details about the error estimation in FEM, the interested readers are referred to references [153], [155] and the references within. Recovery-based error estimators (although supported by a less extended theoretical background with respect to other methods) are often preferable in practical applications because of its simple structure, easy implementation, and effectiveness in predicting errors. The recovery-based error estimators require an additional post-processing procedure to obtain recovered solutions (typically of gradient type), which are more accurate than the original ones. Among various recovery techniques in the literature, the most notable one is the SPR developed by Zienkiewicz and Zhu [147], [148], [156]. The SPR is adopted in [157] to develop error estimators for polygonal and polyhedral FEM. Other techniques also include the recovery by equilibrium in patches (REP) [152], [158] and the polynomial preserving recovery (PPR) [150], [159]–[161]. Comparing to FEM, developing error estimations in the VEM framework is a more involved task because the basis functions of the local VEM space are unknown in the interior of elements. Nevertheless, there exist some error estimators in the literature for C^0 and C^1 VEMs [57]–[60], but all of them are of the residual type.

In this work, we outline a general recovery-based *a posteriori* error estimation framework for H^1 conforming VEM of arbitrary order on general polygonal/polyhedral meshes. For the kth order VEM, a polynomial of order $k+1$ is obtained by a least square fit of the displacement DOFs over each patch. The recovered displacement gradients on the sample points (i.e. vertices and edge nodes) in that patch are then taken as the gradient of that polynomial evaluated at those points. Based on the recovered displacement gradient, an error estimator is obtained through a skeletal error, which evaluates the displacement error on the skeleton of the mesh. In the proposed framework, because only displacement DOFs are used in the fitting process and the errors are only evaluated on the mesh skeleton, then we avoid the difficulty of ignoring the explicit values of the VEM basis functions in the interior of elements. Through numerical examples, the proposed error estimation scheme is shown to be accurate for lower- and higher-order VEMs on various polygonal/polyhedral meshes and with different types of displacement solutions (e.g. smooth displacement fields, displacements with sharp gradients, and ones containing singularities). For linear VEM, the accuracy and effectiveness of the proposed error estimation framework are further demonstrated by comparing it with an SPR-type error estimation as outlined in Appendix C.

4.3 VEM for linear elasticity

In this section, the VEM framework for linear elasticity problem is reviewed. Consider a solid in its stress-free state that occupies a domain $\Omega \subset \mathbb{R}^d$ with d being the dimension. The solid is subjected to a prescribed displacement field, $\mathbf{u}^0$, on one portion of the solid boundary $\Gamma^{\mathbf{X}}$ and a traction $\mathbf{t}$ on the other portion of the solid boundary $\Gamma^{\mathbf{t}}$, such that $\Gamma^{\mathbf{X}} \cup \Gamma^{\mathbf{t}} = \partial\Omega$ and $\Gamma^{\mathbf{X}} \cap \Gamma^{\mathbf{t}} = \emptyset$. In addition, a body force $\mathbf{f}$ is applied in the interior of Ω.

The governing equations of linear elasticity are:

$$\begin{aligned} \operatorname{div}(\mathbb{C}\boldsymbol{\varepsilon}(\mathbf{u})) + \mathbf{f} = \mathbf{0} \text{ in } \Omega \\ \mathbf{u} = \mathbf{u}^0 \text{ on } \Gamma^{\mathbf{X}} \\ \mathbb{C}\boldsymbol{\varepsilon}(\mathbf{u}) \cdot \mathbf{n} = \mathbf{t} \text{ on } \Gamma^{\mathbf{t}}, \end{aligned} \tag{4.1}$$

where $\mathbb{C}$ is the elasticity modulus tensor, which possesses major and minor symmetries, i.e. $C_{ijkl} = C_{klij} = C_{jikl} = C_{ijlk}$. Additionally, $\boldsymbol{\varepsilon}(\mathbf{u})$ is the linearized strain tensor:

$$\boldsymbol{\varepsilon}(\mathbf{u}) = \frac{1}{2}\left[(\nabla\mathbf{u})^T + \nabla\mathbf{u}\right], \tag{4.2}$$

where ∇ stands for the gradient operator. The above governing equations for linear elasticity can be recast in variational form, which consists of finding $\mathbf{u}$ among the space of kinematically admissible displacements such that

$$\int_\Omega \boldsymbol{\varepsilon}(\delta\mathbf{u}) : \left[\mathbb{C}\boldsymbol{\varepsilon}(\mathbf{u})\right] \mathrm{d}\mathbf{X} = \int_\Omega \mathbf{f} \cdot \delta\mathbf{u} \mathrm{d}\mathbf{X} + \int_{\Gamma^{\mathbf{t}}} \mathbf{t} \cdot \delta\mathbf{u} \mathrm{d}S \quad \forall \delta\mathbf{u} \in \mathscr{K}^0, \tag{4.3}$$

where the space $\mathscr{K}^0$ denotes the kinematically admissible displacements that vanish on $\Gamma^{\mathbf{X}}$.

4.3.1 Virtual elements on arbitrary meshes

In this subsection, we review the definition of the local spaces of virtual elements in 2D and 3D [42], [140], [162]. As a declaration of notations, in the following presentation we use F and P to denote polygons (an element in 2D or a face in 3D) and polyhedrons, respectively. Moreover, we use e to represent a generic edge in the mesh. Whenever we have definitions that are independent of dimensions, we use E as a generic element (i.e. generic polytope).

2D virtual elements

Before stating the definition of the local virtual space $\mathscr{V}_k(F)$ for 2D virtual elements, we first introduce a preliminary virtual space denoted by $\widetilde{\mathscr{V}}_k(F)$ as follows

$$\widetilde{\mathscr{V}}_k(F) \doteq \left\{ v \in H^1(F) : v|_e \in \mathscr{P}_k(e) \forall e \in \partial F, \Delta v \in \mathscr{P}_k(F) \right\}, \tag{4.4}$$

where Δ stands for the Laplacian operator and $\mathscr{P}_k(\cdot)$ is the space of polynomial functions whose orders are less or equal to k. By definition, the space $\widetilde{\mathscr{V}}_k(F)$ contains functions whose Laplacians in the interior of F and boundary variations are both kth order polynomials. It is immediate that $\mathscr{P}_k(F) \subseteq \widetilde{\mathscr{V}}_k(F)$. Furthermore, for any v in $\widetilde{\mathscr{V}}_k(F)$, by knowing the following three sets of information:

- the values of v at vertices $\mathbf{X}_v$: $v(\mathbf{X}_v), \quad \forall \mathbf{X}_v \in F;$ (4.5)
- the values of v at $\mathbf{X}_e^\ell$: $v(\mathbf{X}_e^\ell), \quad \forall e \in \partial F, \ell = 1, ..., k-1;$ (4.6)
- the moments of v up to order $k-2$: $\displaystyle\int_F v p_{k-2} \mathrm{d}\mathbf{X} \quad \forall p_{k-2} \in \mathscr{P}_{k-2}(F),$ (4.7)

where $\mathbf{X}_e^\ell$ is the ℓth internal integration point of the Gauss-Lobatto rule of order $2k-1$ on edge e, one can compute a projection $\Pi_F^\nabla v \in \mathscr{P}_k(F)$ such that

$$\begin{cases} \int_F \nabla(\Pi_F^\nabla v)\cdot\nabla p_k \mathrm{d}\mathbf{X} = \int_F \nabla v\cdot\nabla p_k \mathrm{d}\mathbf{X} \quad \forall p_k \in \mathscr{P}_k(F) & \\ \sum_{\mathbf{X}_v \in F} \Pi_F^\nabla v(\mathbf{X}_v) = \sum_{\mathbf{X}_v \in E} v(\mathbf{X}_v) & \text{if } k=1 \\ \int_F \Pi_F^\nabla v \mathrm{d}\mathbf{X} = \int_F v \mathrm{d}\mathbf{X} & \text{if } k \geq 2 \end{cases} . \tag{4.8}$$

In fact, the computability of the projection $\Pi_F^\nabla v$ when (4.5)–(4.7) are given becomes more apparent by applying an integration by parts to the first condition of Eq. (4.8).

Having introduced the preliminary virtual space $\widetilde{\mathscr{V}}_k(F)$, we formally define the 2D local virtual space $\mathscr{V}_k(F)$ as

$$\mathscr{V}_k(F) \doteq \left\{ v \in \widetilde{\mathscr{V}}_k(F) : \int_F vq\mathrm{d}\mathbf{X} = \int_F (\Pi_F^\nabla v) q \mathrm{d}\mathbf{X} \quad \forall q \in \Big(\mathscr{P}_k / \mathscr{P}_{k-2}(F) \Big) \right\}, \tag{4.9}$$

where $\Big(\mathscr{P}_k / \mathscr{P}_{k-2}(F) \Big)$ stands for the polynomial space in $\mathscr{P}_k(F)$ that are L^2 orthogonal to $\mathscr{P}_{k-2}(F)$. By definition, we have $\mathscr{P}_k(F) \subseteq \mathscr{V}_k(F) \subseteq \widetilde{\mathscr{V}}_k(F)$ and, moreover, the sets of information of v in (4.5)–(4.7) constitute a complete set of DOFs for $\mathscr{V}_k(F)$.

Remark 4.3.1 Using the given set of DOFs of v in (4.5)–(4.7) and, according to (4.9), we can exactly compute $\int_F v p_k \mathrm{d}\mathbf{X}$ for any $p_k \in \mathscr{P}_k(F)$. As will be discussed later, this is an important property of the local virtual space $\mathscr{V}_k(F)$ that allows us to compute a L^2 projection of $\mathscr{V}_k(F)$ onto $\mathscr{P}_k(F)$ [140].

3D virtual elements

For the 3D case, we restrict our attention to polyhedrons with planar faces. Again, as a preliminary space, $\widetilde{\mathscr{V}}_k(P)$ is introduced as

$$\widetilde{\mathscr{V}}_k(P) \doteq \left\{ v \in H^1(P) : v|_F \in \mathscr{V}_k(F) \; \forall F \in \partial P, \; \Delta v \in \mathscr{P}_k(P) \right\}. \tag{4.10}$$

In the above definition, we specify that the functions in $\widetilde{\mathscr{V}}_k(P)$ belong to the 2D virtual space $\mathscr{V}_k(F)$ on each face F of polyhedron P. For any v in $\widetilde{\mathscr{V}}_k(P)$, the obvious 3D counterpart of the projection $\Pi_P^\nabla v$ defined in (4.8) of v onto $\mathscr{P}_k(P)$ can be uniquely determined and computed provided that we have the following information on v (see [162]):

- the values of v at vertices $\mathbf{X}_v$: $v(\mathbf{X}_v), \quad \forall \mathbf{X}_v \in P$; (4.11)
- the values of v at $\mathbf{X}_e^\ell$: $v(\mathbf{X}_e^\ell), \quad \forall e \in \partial P, \ell = 1, ..., k-1$; (4.12)
- the moments of v up to order $k-2$ on face $F \in \partial P$: $\displaystyle\int_F v p_{k-2} \mathrm{d}S \quad \forall p_{k-2} \in \mathscr{P}_{k-2}(F)$; (4.13)
- the moments of v up to order $k-2$ in P: $\displaystyle\int_P v p_{k-2} \mathrm{d}\mathbf{X} \quad \forall p_{k-2} \in \mathscr{P}_{k-2}(P)$. (4.14)

As compared to the 2D case, an additional set of information of v is needed in 3D, i.e. (4.13), which includes the moments of v up to order $k-2$ on every face of the element. In particular, by examining the first condition of (4.8), also in the 3D counterpart we can apply integration by parts to the term on its right-hand side, which yields

$$\int_P \nabla v \cdot \nabla p_k \mathrm{d}\mathbf{X} = \int_{\partial P} v\,(\nabla p_k \cdot \mathbf{n}) \mathrm{d}s - \int_P v \Delta p_k \mathrm{d}\mathbf{X}. \tag{4.15}$$

The terms in the above relation can be exactly computed. On one hand, the second term on the right-hand side of the above relation is known due to (4.14). On the other hand, because v belongs to $\mathscr{V}_k(F)$ on each face $F \in P$, then the first term on the right-hand side of the above relation is also computable by combining (4.13) and (4.9) (also see **Remark 4.3.1**).

With $\widetilde{\mathscr{V}}_k(P)$, we are ready to define the virtual space $\mathscr{V}_k(P)$ for 3D virtual elements as

$$\mathscr{V}_k(P) \doteq \left\{ v \in \widetilde{\mathscr{V}}_k(P) : \int_P vq \mathrm{d}\mathbf{X} = \int_P (\Pi_P^\nabla v) q \mathrm{d}\mathbf{X} \quad \forall q \in \left(\mathscr{P}_k / \mathscr{P}_{k-2}(P) \right) \right\}. \tag{4.16}$$

Again, it can be shown that (4.11)–(4.14) constitute a complete set of DOFs for $\mathscr{V}_k(P)$

[162].

Remark 4.3.2 For a given element E in 2D or 3D, although the projection operator Π_E^∇ is defined as from $\widetilde{\mathscr{V}}_k(E)$ onto $\mathscr{P}_k(E)$, we can also define the projection Π_E^∇ from $\mathscr{V}_k(E)$ onto $\mathscr{P}_k(E)$ using the same set of definitions (4.8). For any v in $\mathscr{V}_k(E)$, this projection is computable using only the DOFs of v, i.e., (4.5)–(4.7) in 2D or (4.11)–(4.14) in 3D.

Two projection operators

In VEM approximations, two L^2 projections are utilized, which respectively project a given function $v \in \mathscr{V}_k(E)$ and its gradient onto polynomial functions. The first projection, denoted as $\Pi_k^0 : \mathscr{V}_k(E) \to \mathscr{P}_k(E)$, is defined for any given v in $\mathscr{V}_k(E)$ as

$$\int_E \Pi_k^0 v\, p_k \mathrm{d}\mathbf{X} = \int_E v\, p_k \mathrm{d}\mathbf{X} \quad \forall p_k \in \mathscr{P}_k(E). \tag{4.17}$$

Because any $p_k \in \mathscr{P}_k(E)$ can always be decomposed as $p_k = p_{k-2} + q_k$ with $p_{k-2} \in \mathscr{P}_{k-2}(E)$ and $q_k \in (\mathscr{P}_k/\mathscr{P}_{k-2}(E))$, then we can express the right-hand side of above expression as

$$\int_E v\, p_k \mathrm{d}\mathbf{X} = \int_E v(p_{k-2} + q_k)\mathrm{d}\mathbf{X} = \int_E v p_{k-2} \mathrm{d}\mathbf{X} + \int_E (\Pi_E^\nabla v) q_k \mathrm{d}\mathbf{X}. \tag{4.18}$$

This shows that the term $\int_E v\, p_k \mathrm{d}\mathbf{X}$ can be exactly computed using only the DOFs of v in both 2D and 3D and, thus, the same holds for the projection $\Pi_k^0 v$.

Remark 4.3.3 For linear and quadratic virtual elements, i.e. $k = 1$ and $k = 2$, it is shown in [140] that the projection $\Pi_k^0 v$ coincides with $\Pi_k^\nabla v$ for any $v \in \mathscr{V}_1(E)$. For virtual elements of order $k \geq 3$, however, projections $\Pi_k^0 v$ and $\Pi_E^\nabla v$ are different.

The second projection, which is utilized to project ∇v, is denoted as $\Pi_{k-1}^0 : [L^2(E)]^d \to [\mathscr{P}_{k-1}(E)]^d$ for a kth order virtual elements. For a given v in $\mathscr{V}_k(E)$, the projection $\Pi_{k-1}^0 \nabla v$

is defined as

$$\int_E \Pi^0_{k-1}\nabla v\cdot \mathbf{p}_{k-1}\mathrm{d}\mathbf{X} = \int_E \nabla v\cdot \mathbf{p}_{k-1}\mathrm{d}\mathbf{X} \quad \forall \mathbf{p}_{k-1}\in[\mathscr{P}_{k-1}(E)]^d. \tag{4.19}$$

Applying integration by part to the right-hand side of the above expression gives

$$\int_E \nabla v\cdot \mathbf{p}_{k-1}\mathrm{d}\mathbf{X} = \int_{\partial E} v\,\mathbf{p}_{k-1}\cdot \mathbf{n}\mathrm{d}s - \int_E v\mathrm{div}\mathbf{p}_{k-1}\mathrm{d}\mathbf{X}, \tag{4.20}$$

which can is also computable in both 2D and 3D using the DOFs of v. More specifically, the second term on the right-hand side of the above expression is immediately given by DOFs (4.7) and (4.14). Moreover, in the 2D case, the first term on the right-hand side of (4.20) can be exactly integrated using the Gauss-Lobatto rule of order $2k-1$ on each edge using the sets of DOFs (4.5) and (4.6). For the 3D case, on the other hand, by decomposing $\mathbf{p}_{k-1}$ as $\mathbf{p}_{k-1} = \mathbf{p}_{k-2}+\mathbf{q}_{k-1}$ with $\mathbf{p}_{k-2}\in[\mathscr{P}_{k-2}(E)]^3$ and $\mathbf{q}_{k-1}\in[\mathscr{P}_{k-1}/\mathscr{P}_{k-2}(E)]^3$, we can express the first term on the right-hand side of (4.20) as

$$\int_{\partial P} v\,\mathbf{p}_{k-1}\cdot\mathbf{n}\mathrm{d}S = \sum_{F\in\partial P}\int_F v(\mathbf{p}_{k-2}+\mathbf{q}_{k-1})\cdot\mathbf{n}\mathrm{d}S = \sum_{F\in\partial P}\left[\int_F v\mathbf{p}_{k-2}\cdot\mathbf{n}\mathrm{d}S + \int_F \Pi^{\nabla}_P v\mathbf{q}_{k-1}\cdot\mathbf{n}\mathrm{d}S\right], \tag{4.21}$$

which is computable using only the DOFs of v.

4.3.2 Virtual element approximation for linear elasticity

Let us consider a discretization denoted by Ω_h of the domain Ω into non-overlapping polygons or polyhedrons, where h stands for the average element size. We also assume that the boundary of the mesh, denoted by Γ_h,is compatible with the applied displacement and traction boundary conditions, namely, both $\Gamma^{\mathbf{X}}_h$ and $\Gamma^{\mathbf{t}}_h$ consist of unions of edges and faces of the mesh. In 3D, we also use F to denote a generic face in the mesh. For a kth order discretization, the global displacement space $\mathscr{K}_{h,k}$ is a conforming finite dimensional space defined as

$$\mathscr{K}_{h,k} \doteq \left\{ \mathbf{v}_h \in \mathscr{K} : \mathbf{v}_h|_E \in [\mathscr{V}_k(E)]^d, \forall E \in \Omega_h \right\}, \tag{4.22}$$

where $\mathscr{V}_k(E)$ is a kth order local virtual space defined in the preceding subsections. In each element E, each component of the local displacement $\mathbf{v}$ ($\mathbf{v} = [v_x, v_y]^T$ in 2D or $\mathbf{v} = [v_x, v_y, v_z]^T$ in 3D) has the set of DOFs specified in (4.5)–(4.7) and (4.11)–(4.14) respectively for 2D and 3D cases. In the following discussions, we introduce $\mathbf{\Pi}_k^0 : [\mathscr{V}_k(E)]^d \to [\mathscr{P}_k(E)]^d$ as the action of Π_k^0 on each component of the vector field, i.e., $\mathbf{\Pi}_k^0\mathbf{v} = [\Pi_k^0 v_x, \Pi_k^0 v_y]^T$ in 2D and $\mathbf{\Pi}_k^0\mathbf{v} = [\Pi_k^0 v_x, \Pi_k^0 v_y, \Pi_k^0 v_z]^T$ in 3D. Similarly, we also introduce the projection $\mathbf{\Pi}_{k-1}^0 : [L^2(E)]^{d\times d} \to [\mathscr{P}_{k-1}(E)]^{d\times d}$ for second order tensors as

$$\mathbf{\Pi}_{k-1}^0 \nabla \mathbf{v} = \begin{bmatrix} (\Pi_{k-1}^0 \nabla v_x)^T \\ (\Pi_{k-1}^0 \nabla v_y)^T \end{bmatrix} \text{ in 2D or } \mathbf{\Pi}_{k-1}^0 \nabla \mathbf{v} = \begin{bmatrix} (\Pi_{k-1}^0 \nabla v_x)^T \\ (\Pi_{k-1}^0 \nabla v_y)^T \\ (\Pi_{k-1}^0 \nabla v_z)^T \end{bmatrix} \text{ in 3D.} \tag{4.23}$$

When applied to the strain tensor $\boldsymbol{\varepsilon}(\mathbf{v})$, the projection $\mathbf{\Pi}_{k-1}^0 \boldsymbol{\varepsilon}(\mathbf{v})$ stands for

$$\mathbf{\Pi}_{k-1}^0 \boldsymbol{\varepsilon}(\mathbf{v}) = \frac{1}{2} \left[(\mathbf{\Pi}_{k-1}^0 \nabla \mathbf{v})^T + \mathbf{\Pi}_{k-1}^0 \nabla \mathbf{v} \right] \tag{4.24}$$

Our next step is to introduce the VEM approximation to the continuous problem (4.3), see for instance [26], [42], [45]. To that end, for any element E, we need to first approximate the following local bilinear form on the left-hand side of (4.3):

$$a^E(\mathbf{u}_h, \mathbf{v}_h) = \int_E \boldsymbol{\varepsilon}(\mathbf{u}_h) : [\mathbb{C}\boldsymbol{\varepsilon}(\mathbf{v}_h)]\, \mathrm{d}\mathbf{X}. \tag{4.25}$$

The VEM approximation $a_h^E(\mathbf{u}_h, \mathbf{v}_h)$ of $a^E(\mathbf{u}_h, \mathbf{v}_h)$ is composed of the following two terms:

$$a_h^E(\mathbf{u}_h, \mathbf{v}_h) = \int_E \mathbf{\Pi}_{k-1}^0 \boldsymbol{\varepsilon}(\mathbf{v}_h) : \mathbb{C} : \mathbf{\Pi}_{k-1}^0 \boldsymbol{\varepsilon}(\mathbf{u}_h) \mathrm{d}\mathbf{X} + \alpha_E S_{h,E}(\mathbf{u}_h - \mathbf{\Pi}_k^0 \mathbf{u}_h, \mathbf{v}_h - \mathbf{\Pi}_k^0 \mathbf{v}_h), \tag{4.26}$$

where the first integral on the right-hand side is evaluated (exactly) using a numerical integration of order $2k-2$; $S_{h,E}(\cdot,\cdot)$ is a bilinear form that is computationally inexpensive to compute and satisfies the coercivity condition; and α_E is a scaling parameter that scales $S_{h,E}(\cdot,\cdot)$ to the same order of magnitude as $a^E(\cdot,\cdot)$. Typical choices of $S_{h,E}(\cdot,\cdot)$ and α_E are:

$$S_{h,E}(\mathbf{u}_h,\mathbf{v}_h) = h_E^{d-2}\sum_{i=1}^{N_E}\boldsymbol{\Xi}_i(\mathbf{u}_h)\cdot\boldsymbol{\Xi}_i(\mathbf{v}_h) \text{ and } \alpha_E = \text{trace}\mathbf{C} = C_{ijij} \text{ (in indical notation)} \tag{4.27}$$

where $h_E \doteq |E|^{1/d}$ represents the size of element E, $\boldsymbol{\Xi}_i(\mathbf{v})$ stands for the ith DOF of $\mathbf{v}$ in E, and N_E stands for the total number of such local DOFs in E. In the VEM literature, the first and second terms of $a_h^E(\mathbf{u}_h,\mathbf{v}_h)$ are respectively known as the consistency and stability terms, and they are respectively responsible for the satisfaction of consistency and stability conditions, which are the two key conditions to ensure the convergence of the VEM approximation [37], [42].

On the other hand, we approximate the loading term on the right-hand side of (4.3) as

$$\langle\mathbf{f},\mathbf{v}_h\rangle_h + \langle\mathbf{t},\mathbf{v}_h\rangle_h = \sum_F \int\!\!\!\!\!\square_F \mathbf{f}\cdot\boldsymbol{\Pi}_k^0\mathbf{v}_h\mathrm{d}\mathbf{X} + \sum_{e\in\Gamma_h^t}\oint_e \mathbf{t}\cdot\mathbf{v}_h\mathrm{d}\mathbf{X} \qquad \text{in 2D,} \tag{4.28}$$

and

$$\langle\mathbf{f},\mathbf{v}_h\rangle_h + \langle\mathbf{t},\mathbf{v}_h\rangle_h = \sum_P \int\!\!\!\!\!\square_P \mathbf{f}\cdot\boldsymbol{\Pi}_k^0\mathbf{v}_h\mathrm{d}\mathbf{X} + \sum_{F\in\Gamma_h^t}\oint_F \mathbf{t}\cdot\boldsymbol{\Pi}_k^0\mathbf{v}_h\mathrm{d}\mathbf{X} \qquad \text{in 3D,} \tag{4.29}$$

where $\int\!\!\!\!\!\square$ and $\oint$ denote any numerical integrations that are exact for polynomials of order $2k-2$ and $2k-1$, respectively. We note that, in the 2D case, $\mathbf{v}_h$ is by definition known as a polynomial function of order k on each edge. In addition, in the special case of $k=1$, instead of using a integration rule that is exact of order 0 for $\int\!\!\!\!\!\square$ (i.e. is able to integrate any constant function exactly), we use a one-point integration rule with the quadrature point and weight being the centroid and either the area or volume of each element, respectively.

This integration rule is exact when the integrand is a linear function. For the higher order VEM in this chapter, we utilize the commonly used triangulation scheme with the above-stated order of accuracy (more sophisticated choices, for example [107], [163], could be taken).

We are now ready to state the final form of the VEM approximation for linear elasticity problems, which consists of finding $\mathbf{u}_h \in \mathscr{K}_h$ such that

$$\sum_E \left[\int_E \mathbf{\Pi}^0_{k-1} \boldsymbol{\varepsilon}(\delta \mathbf{u}_h) : \mathbb{C} : \mathbf{\Pi}^0_{k-1} \boldsymbol{\varepsilon}(\mathbf{u}_h) \mathrm{d}\mathbf{X} + \alpha_E S_{h,E} \Big(\mathbf{u}_h - \mathbf{\Pi}^0_k \mathbf{u}_h, \delta \mathbf{u}_h - \mathbf{\Pi}^0_k (\delta \mathbf{u}_h) \Big) \right]$$
$$= \langle \mathbf{f}, \delta \mathbf{u}_h \rangle_h + \langle \mathbf{t}, \delta \mathbf{u}_h \rangle_h, \quad \forall \delta \mathbf{u}_h \in \mathscr{K}_h^0, \quad (4.30)$$

where $\mathscr{K}_h^0$ is a subspace of $\mathscr{K}_h$ with functions that vanish on $\Gamma_h^{\mathbf{X}}$.

4.4 A recovery-based *a posteriori* error estimation for VEM

This section introduces a gradient recovery scheme that reconstructs a more accurate displacement gradient based on the VEM solutions. Making use of the reconstructed displacement gradient, this section further propose an *a posteriori* error estimator for the VEM, which estimates the error of the displacement gradient. According to the VEM philosophy, the gradient recovery scheme presented here will focus on reconstructing displacement gradients on the skeleton (i.e. union of edges) of the mesh. We also employ an H^1-type skeletal norm for the displacement gradient error estimators. For a given discretization Ω_h and a function $\mathbf{v}$, the H^1-type skeletal norm, denoted by $\varepsilon^s_{1,\mathbf{v}}$, is defined as

$$\varepsilon^s_{1,\mathbf{v}} = \left[\sum_{F \in \Omega_h} h_F \sum_{e \in \partial F} \int_e (\nabla \mathbf{v} \cdot \boldsymbol{\tau}_e) \cdot (\nabla \mathbf{v} \cdot \boldsymbol{\tau}_e) \mathrm{d}e \right]^{\frac{1}{2}} \qquad \text{in 2D,} \qquad (4.31)$$

and

$$\varepsilon^s_{1,\mathbf{v}} = \left[\sum_{P\in\Omega_h} h_P \sum_{F\in\partial P} h_F \sum_{e\in\partial F} \int_e (\nabla\mathbf{v}\cdot\boldsymbol{\tau}_e)\cdot(\nabla\mathbf{v}\cdot\boldsymbol{\tau}_e)\mathrm{d}e\right]^{\frac{1}{2}} \qquad \text{in 3D}, \tag{4.32}$$

where h_P and h_F are the diameter of polyhedron P and polygon F; and $\boldsymbol{\tau}_e$ denotes the unit tangential vector of edge e. We remark that, unlike the regular H^1 semi-norm that integrates over each element in the mesh, the H^1-type skeletal norms defined in (4.31) and (4.32) only sample function gradients along the skeleton of the mesh. These skeletal norms mimic the regular H^1 semi-norms in the following sense: they take H^1 semi-norms on the skeleton of the mesh (so that the norm for a constant function vanishes), and those differences are then scaled in order to achieve the same behavior (with respect to element contractions/expansions) as the regular H^1 semi-norm. We also remark that, similar mesh-dependent norms are considered in the Mimetic Finite Difference (MFD) literature [164], [165].

4.4.1 A gradient recovery scheme for the VEM

We present a gradient recovery scheme for VEM of arbitrary order. This subsection is outlined as follows. We will first describe in detail the displacement gradient reconstructions for linear and quadratic virtual elements, respectively. Afterward, a general gradient recovery framework will be outlined for VEM of arbitrary order k. Note that, in the present section, we assume to have a patch of elements associated with each vertex of the mesh; the selection of such patches will be discussed later.

Before proceeding, we first introduce the notation utilized in this gradient recovery scheme. For the ith node in the mesh expressed by $\mathbf{X}_i = [X_i, Y_i]^T$ in 2D (and $\mathbf{X}_i = [X_i, Y_i, Z_i]^T$ in 3D) , we denote by ω_i a patch of elements associated with it. We will describe the criteria for how to choose ω_i at the end of this subsection. For now, let us assume that a patch ω is given. We denote by h_ω a characteristic size of the patch ω, which is taken as the maximum distance between the nodes in the patch, and by N^E_ω the total number of elements it contains. For the 2D cases, we use $\Xi^v_i(v)$, $\Xi^e_i(v)$ and $\Xi^I_i(v)$ to denote the ith vertex, edge and internal DOFs of a given function v in the patch ω, respectively, and N^v_ω, N^e_ω and N^I_ω to denote the total numbers of vertex, edge and internal DOFs in this patch, respectively. For the 3D case, we additionally denote by $\Xi^F_i(v)$ and N^F_ω the ith (internal) face DOFs and the total number of (internal) face DOFs in the patch ω, respectively. For instance, if a given patch consists of 2D linear virtual elements, the vertex DOFs of v are its values at the vertices of this patch, and there are no edge and internal DOFs. Alternatively, if a given patch consists of 2D quadratic virtual elements, in addition to the vertex DOFs (as in linear virtual elements), we have one edge DOF at the mid-node on every edge in this patch, and each element in the patch has one internal DOF, which is the zero order moment of v over this element.

Moreover, for a given patch ω, we denote by $\mathscr{P}_k(\omega)$ the set of polynomial functions of degree less than or equal to k with dimension $n\mathscr{P}_k$. For a given order k, we have $n\mathscr{P}_k = (k+1)(k+2)/2$ in 2D and $n\mathscr{P}_k = (k+1)(k+2)(k+3)/6$ in 3D. If we introduce a multi-index $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \alpha_3)$, we can define the set of basis functions of $\mathscr{P}_k(\omega)$ as

$$m^\omega_{\boldsymbol{\alpha}}(\mathbf{X}) = \Big(\frac{X - X_\omega}{h_\omega}\Big)^{\alpha_1}\Big(\frac{Y - Y_\omega}{h_\omega}\Big)^{\alpha_2}\Big(\frac{Z - Z_\omega}{h_\omega}\Big)^{\alpha_3}, \quad |\boldsymbol{\alpha}| \leq k, \tag{4.33}$$

where $|\boldsymbol{\alpha}| = \alpha_1 + \alpha_2 + \alpha_3$ and $\mathbf{X}_\omega = [X_\omega, Y_\omega, Z_\omega]^T$ is the centroid of ω (alternatively, $\mathbf{X}_\omega$ can also be any point in ω, e.g., the mean of all nodes in ω). We note that while the above notation is introduced for the 3D case, it also applies to the 2D case with α_3 being zero. In the following discussions, we will reshape the multi-index $\boldsymbol{\alpha}$, $|\boldsymbol{\alpha}| \leq k$ into a scalar index α, $\alpha = 1, ..., n_{\mathscr{P}_k}$ and use $m^\omega_\alpha(\mathbf{X})$ to denote the αth basis functions for $\mathscr{P}_k(\omega)$.

Displacement gradient recovery for linear elements

We address the displacement gradient recovery of linear virtual elements in both 2D and 3D cases. Since the gradient recoveries in 2D and 3D follow similar concept and procedures, we will thoroughly describe the recovery in 2D and then comment on suitable modifications in the 3D case. For both cases, we attempt to reconstruct a displacement gradient at every vertex of the mesh, and, for each vertex, the reconstruction is performed in its associated patches. Once the nodal gradients of all the vertices are obtained, a continuous displacement gradient on mesh skeleton, denoted by $G_h\mathbf{u}_h$, can then be obtained by interpolating those nodal displacement gradients on the mesh skeleton.

For a given patch ω_i (associated with $\mathbf{X}_i$) in 2D, the basic idea of the gradient recovery scheme is to seek a quadratic vector field, denoted by $\mathbf{p}^{\omega_i}(\mathbf{X}) = [p_x^{\omega_i}, p_y^{\omega_i}]^T \in [\mathscr{P}_2(\omega_i)]^2$, such that its X and Y components satisfy

$$p_x^{\omega_i} = \underset{\xi\in\mathscr{P}_2(\omega_i)}{\text{argmin}} \sum_{j=1}^{N_{\omega_i}^v} \left[\Xi_j^v(\xi) - \Xi_j^v(u_{h,x})\right]^2 \text{ and } p_y^{\omega_i} = \underset{\xi\in\mathscr{P}_2(\omega_i)}{\text{argmin}} \sum_{j=1}^{N_{\omega_i}^v} \left[\Xi_j^v(\xi) - \Xi_j^v(u_{h,y})\right]^2, \tag{4.34}$$

respectively, where $\mathbf{u}_h = [u_{h,x}, u_{h,y}]^T$ is the VEM displacement solution.

We can express the quadratic functions $p_x^{\omega_i}$ and $p_y^{\omega_i}$ as linear combinations of the basis functions of $\mathscr{P}_2(\omega_i)$, $m_\alpha^{\omega_i}$, $\alpha = 1,...,6$, as

$$p_x^{\omega_i}(\mathbf{X}) = \sum_{\alpha=1}^{6} m_\alpha^{\omega_i}(\mathbf{X}) q_\alpha^{\omega_i,x} \text{ and } p_y^{\omega_i} = \sum_{\alpha=1}^{6} m_\alpha^{\omega_i}(\mathbf{X}) q_\alpha^{\omega_i,y}, \tag{4.35}$$

where $q_\alpha^{\omega_i,x}$ and $q_\alpha^{\omega_i,y}$ are the coefficients. By further considering the DOFs of $p_x^{\omega_i}$ and $p_y^{\omega_i}$, we arrive at the following matrix expressions:

$$\begin{aligned} \mathbf{P}\mathbf{q}^{\omega_i,x} &= \begin{bmatrix} \Xi_1^v(p_x^{\omega_i}) & \Xi_2^v(p_x^{\omega_i}) & \cdots & \Xi_{N_{\omega_i}^v}^v(p_x^{\omega_i}) \end{bmatrix}^T \\ \mathbf{P}\mathbf{q}^{\omega_i,y} &= \begin{bmatrix} \Xi_1^v(p_y^{\omega_i}) & \Xi_2^v(p_y^{\omega_i}) & \cdots & \Xi_{N_{\omega_i}^v}^v(p_y^{\omega_i}) \end{bmatrix}^T, \end{aligned} \tag{4.36}$$

where $\mathbf{P} \in \mathbb{R}^{N^v_{\omega_i} \times 6}$ and $\mathbf{q}^{\omega_i,x}, \mathbf{q}^{\omega_i,y} \in \mathbb{R}^{6\times 1}$ are of the forms

$$\mathbf{P} = \begin{bmatrix} \Xi^v_1(m^{\omega_i}_1) & \Xi^v_1(m^{\omega_i}_2) & \Xi^v_1(m^{\omega_i}_3) & \Xi^v_1(m^{\omega_i}_4) & \Xi^v_1(m^{\omega_i}_5) & \Xi^v_1(m^{\omega_i}_6) \\ \Xi^v_2(m^{\omega_i}_1) & \Xi^v_2(m^{\omega_i}_2) & \Xi^v_2(m^{\omega_i}_3) & \Xi^v_2(m^{\omega_i}_4) & \Xi^v_2(m^{\omega_i}_5) & \Xi^v_2(m^{\omega_i}_6) \\ & & \vdots & & & \\ \Xi^v_{N^v_{\omega_i}}(m^{\omega_i}_1) & \Xi^v_{N^v_{\omega_i}}(m^{\omega_i}_2) & \Xi^v_{N^v_{\omega_i}}(m^{\omega_i}_3) & \Xi^v_{N^v_{\omega_i}}(m^{\omega_i}_4) & \Xi^v_{N^v_{\omega_i}}(m^{\omega_i}_5) & \Xi^v_{N^v_{\omega_i}}(m^{\omega_i}_6) \end{bmatrix} \tag{4.37}$$

and

$$\mathbf{q}^{\omega_i,x} = \begin{bmatrix} q^{\omega_i,x}_1 & q^{\omega_i,x}_2 & \cdots & q^{\omega_i,x}_6 \end{bmatrix}^T \text{ and } \mathbf{q}^{\omega_i,y} = \begin{bmatrix} q^{\omega_i,y}_1 & q^{\omega_i,y}_2 & \cdots & q^{\omega_i,y}_6 \end{bmatrix}^T. \tag{4.38}$$

With the introduction of above matrices, we can equivalently rewrite (4.34) as seeking $\mathbf{q}^{\omega_i,x}$ and $\mathbf{q}^{\omega_i,y} \in \mathbb{R}^{6\times 1}$ such that

$$\mathbf{q}^{\omega_i,x} = \underset{\mathbf{a}\in\mathbb{R}^{6\times 1}}{\operatorname{argmin}} \left[\mathbf{Pa} - \mathbf{b}^{\omega_i,x}\right]^T \left[\mathbf{Pa} - \mathbf{b}^{\omega_i,x}\right] \text{ and } \mathbf{q}^{\omega_i,y} = \underset{\mathbf{a}\in\mathbb{R}^{6\times 1}}{\operatorname{argmin}} \left[\mathbf{Pa} - \mathbf{b}^{\omega_i,y}\right]^T \left[\mathbf{Pa} - \mathbf{b}^{\omega_i,y}\right], \tag{4.39}$$

where $\mathbf{b}^{\omega_i,x}$ and $\mathbf{b}^{\omega_i}$ are vectors that collects the DOFs of VEM solution $\mathbf{u}_h$ in ω_i, namely, $\mathbf{b}^{\omega_i,x} = [\Xi^v_1(u_{h,x}), ..., \Xi^v_{N^v_{\omega_i}}(u_{h,x})]^T$ and $\mathbf{b}^{\omega_i,y} = [\Xi^v_1(u_{h,y}), ..., \Xi^v_{N^v_{\omega_i}}(u_{h,y})]^T$. Examining the optimality conditions of the above minimization problems, we obtain the following expressions for $\mathbf{q}^{\omega_i,x}$ and $\mathbf{q}^{\omega_i,y}$:

$$\mathbf{q}^{\omega_i,x} = (\mathbf{P}^T\mathbf{P})^{-1}\mathbf{P}^T\mathbf{b}^{\omega_i,x} \text{ and } \mathbf{q}^{\omega_i,y} = (\mathbf{P}^T\mathbf{P})^{-1}\mathbf{P}^T\mathbf{b}^{\omega_i,y}. \tag{4.40}$$

Once the vectors of coefficients $\mathbf{q}^{\omega_i,x}$ and $\mathbf{q}^{\omega_i,y}$ are computed, we obtain the polynomial functions $p^{\omega_i}_x(\mathbf{X})$ and $p^{\omega_i}_y(\mathbf{X})$, and the value of the reconstructed displacement gradient $G_h\mathbf{u}_h(\mathbf{X}_i)$ at $\mathbf{X}_i$ is taken as

$$G_h\mathbf{u}_h(\mathbf{X}_i) = \nabla\mathbf{p}^{\omega_i}(\mathbf{X}_i) = \begin{bmatrix} \frac{\partial p^{\omega_i}_x}{\partial X}(\mathbf{X}_i) & \frac{\partial p^{\omega_i}_x}{\partial Y}(\mathbf{X}_i) \\ \frac{\partial p^{\omega_i}_y}{\partial X}(\mathbf{X}_i) & \frac{\partial p^{\omega_i}_y}{\partial Y}(\mathbf{X}_i) \end{bmatrix}. \tag{4.41}$$

In 3D, the recovery scheme follows a procedure similar to the one outlined above except for two modifications. First, in order to account for the expanding dimension of $\mathscr{P}_2(\omega_i)$ in 3D, the matrix $\mathbf{P}$ defined in (4.37) is modified as

$$\mathbf{P} = \begin{bmatrix} \Xi_1^v(m_1^{\omega_i}) & \Xi_1^v(m_2^{\omega_i}) & \Xi_1^v(m_3^{\omega_i}) & \cdots & \Xi_1^v(m_{10}^{\omega_i}) \\ \Xi_2^v(m_1^{\omega_i}) & \Xi_2^v(m_2^{\omega_i}) & \Xi_2^v(m_3^{\omega_i}) & \cdots & \Xi_2^v(m_{10}^{\omega_i}) \\ & & \ddots & & \\ \Xi_{N_{\omega_i}^v}^v(m_1^{\omega_i}) & \Xi_{N_{\omega_i}^v}^v(m_2^{\omega_i}) & \Xi_{N_{\omega_i}^v}^v(m_3^{\omega_i}) & \cdots & \Xi_{N_{\omega_i}^v}^v(m_{10}^{\omega_i}) \end{bmatrix}. \tag{4.42}$$

Moreover, the recovery scheme is expanded to include the Z component of the displacement vector $\mathbf{u}_h = [u_{h,x}, u_{h,y}, u_{h,z}]^T$ in 3D. With the modified matrix $\mathbf{P}$, the displacement gradient recovery scheme in 3D consists of seeking $\mathbf{q}^{\omega_i,x}$, $\mathbf{q}^{\omega_i,y}$ and $\mathbf{q}^{\omega_i,z} \in \mathbb{R}^{10\times 1}$ such that

$$\begin{aligned} \mathbf{q}^{\omega_i,x} &= \underset{\mathbf{a}\in\mathbb{R}^{10\times 1}}{\operatorname{argmin}} \left[\mathbf{Pa} - \mathbf{b}^{\omega_i,x}\right]^T \left[\mathbf{Pa} - \mathbf{b}^{\omega_i,x}\right] \\ \mathbf{q}^{\omega_i,y} &= \underset{\mathbf{a}\in\mathbb{R}^{10\times 1}}{\operatorname{argmin}} \left[\mathbf{Pa} - \mathbf{b}^{\omega_i,y}\right]^T \left[\mathbf{Pa} - \mathbf{b}^{\omega_i,y}\right] \\ \mathbf{q}^{\omega_i,z} &= \underset{\mathbf{a}\in\mathbb{R}^{10\times 1}}{\operatorname{argmin}} \left[\mathbf{Pa} - \mathbf{b}^{\omega_i,z}\right]^T \left[\mathbf{Pa} - \mathbf{b}^{\omega_i,z}\right], \end{aligned} \tag{4.43}$$

where, similarly to $\mathbf{b}^{\omega_i,x}$ and $\mathbf{b}^{\omega_i,y}$, $\mathbf{b}^{\omega_i,z}$ is a vector collecting the DOFs of the Z component of $\mathbf{u}_h$ as $\mathbf{b}^{\omega_i,z} = [\Xi_1^v(u_{h,z}), ..., \Xi_{N_{\omega_i}^v}^v(u_{h,z})]^T$. Once the coefficient vectors $\mathbf{q}^{\omega_i,x}$, $\mathbf{q}^{\omega_i,y}$ and $\mathbf{q}^{\omega_i,z}$ are obtained, the values of the reconstructed displacement gradient field $G_h\mathbf{u}_h$ at node $\mathbf{X}_i$ are naturally given by

$$G_h\mathbf{u}_h(\mathbf{X}_i) = \nabla\mathbf{p}^{\omega_i}(\mathbf{X}_i) = \begin{bmatrix} \frac{\partial p_x^{\omega_i}}{\partial X}(\mathbf{X}_i) & \frac{\partial p_x^{\omega_i}}{\partial Y}(\mathbf{X}_i) & \frac{\partial p_x^{\omega_i}}{\partial Z}(\mathbf{X}_i) \\ \frac{\partial p_y^{\omega_i}}{\partial X}(\mathbf{X}_i) & \frac{\partial p_y^{\omega_i}}{\partial Y}(\mathbf{X}_i) & \frac{\partial p_y^{\omega_i}}{\partial Z}(\mathbf{X}_i) \\ \frac{\partial p_z^{\omega_i}}{\partial X}(\mathbf{X}_i) & \frac{\partial p_z^{\omega_i}}{\partial Y}(\mathbf{X}_i) & \frac{\partial p_z^{\omega_i}}{\partial Z}(\mathbf{X}_i) \end{bmatrix}. \tag{4.44}$$

Displacement gradient recovery for 2D quadratic elements

We present a gradient reconstruction scheme for quadratic elements in 2D, in which we attempt to reconstruct the displacement gradient at every vertex of the mesh, as well as the mid-side node of every edge. In particular, the displacement gradient for every vertex is reconstructed within the associated patch, and the one for each mid-side node is obtained by averaging the reconstructed gradients within both patches associated with the two vertices of its edge.

Let us first look at the gradient recovery on the vertices. Similar to the linear case, for a given vertex $\mathbf{X}_i$ and its associated patch ω_i, we aim to seek a cubic vector field, denoted by $\mathbf{p}^{\omega_i} = [p_x^{\omega_i}, p_y^{\omega_i}]^T \in [\mathscr{P}_3(\omega_i)]^2$, such that its components satisfy

$$p_x^{\omega_i} = \underset{\xi\in\mathscr{P}_3(\omega_i)}{\operatorname{argmin}} \left\{ \sum_{j=1}^{N_{\omega_i}^v} \left[\Xi_j^v(\xi) - \Xi_j^v(u_{h,x})\right]^2 + \sum_{j=1}^{N_{\omega_i}^e} \left[\Xi_j^e(\xi) - \Xi_j^e(u_{h,x})\right]^2 \right.$$
$$\left. + \sum_{j=1}^{N_{\omega_i}^I} \left[\Xi_j^I(\xi) - \Xi_j^I(u_{h,x})\right]^2 \right\} \quad (4.45)$$

$$p_y^{\omega_i} = \underset{\xi\in\mathscr{P}_3(\omega_i)}{\operatorname{argmin}} \left\{ \sum_{j=1}^{N_{\omega_i}^v} \left[\Xi_j^v(\xi) - \Xi_j^v(u_{h,y})\right]^2 + \sum_{j=1}^{N_{\omega_i}^e} \left[\Xi_j^e(\xi) - \Xi_j^e(u_{h,y})\right]^2 \right.$$
$$\left. + \sum_{j=1}^{N_{\omega_i}^I} \left[\Xi_j^I(\xi) - \Xi_j^I(u_{h,y})\right]^2 \right\}, \quad (4.46)$$

where we recall that the operator $\Xi_j^I(\cdot) = 1/|E_j| \int_{E_j} (\cdot) \mathrm{d}\mathbf{X}$ represents the internal DOF of the jth element in the patch.

Expanding $p_x^{\omega_i}(\mathbf{X})$ and $p_y^{\omega_i}(\mathbf{X})$ in terms of basis functions $m_\alpha^{\omega_i}(\mathbf{X})$ of $\mathscr{P}_3(\omega_i)$, $\alpha = 1,\ldots,10$, as

$$p_x^{\omega_i}(\mathbf{X}) = \sum_{\alpha=1}^{10} m_\alpha^{\omega_i}(\mathbf{X}) q_\alpha^{\omega_i,x} \quad \text{and} \quad p_y^{\omega_i}(\mathbf{X}) = \sum_{\alpha=1}^{10} m_\alpha^{\omega_i}(\mathbf{X}) q_\alpha^{\omega_i,y}, \quad (4.47)$$

we obtain the following relations in matrix forms as

$$\mathbf{P}\mathbf{q}^{\omega_i,x} = \begin{bmatrix} \Xi_1^v(p_x^{\omega_i}) & \cdots & \Xi_{N_{\omega_i}^v}^v(p_x^{\omega_i}) & \Xi_1^e(p_x^{\omega_i}) & \cdots & \Xi_{N_{\omega_i}^e}^e(p_x^{\omega_i}) & \Xi_1^I(p_x^{\omega_i}) & \cdots & \Xi_{N_{\omega_i}^I}^I(p_x^{\omega_i}) \end{bmatrix}^T, \tag{4.48}$$

and

$$\mathbf{P}\mathbf{q}^{\omega_i,y} = \begin{bmatrix} \Xi_1^v(p_y^{\omega_i}) & \cdots & \Xi_{N_{\omega_i}^v}^v(p_y^{\omega_i}) & \Xi_1^e(p_y^{\omega_i}) & \cdots & \Xi_{N_{\omega_i}^e}^e(p_y^{\omega_i}) & \Xi_1^I(p_y^{\omega_i}) & \cdots & \Xi_{N_{\omega_i}^I}^I(p_y^{\omega_i}) \end{bmatrix}^T, \tag{4.49}$$

where matrices $\mathbf{P} \in \mathbb{R}^{(N_{\omega_i}^v + N_{\omega_i}^e + N_{\omega_i}^I)\times 10}$ and $\mathbf{q}^{\omega_i,x}$, $\mathbf{q}^{\omega_i,y} \in \mathbb{R}^{10\times 1}$ are given by

$$\mathbf{P} = \begin{bmatrix} \Xi_1^v(m_1^{\omega_i}) & \Xi_1^v(m_2^{\omega_i}) & \Xi_1^v(m_3^{\omega_i}) & \cdots & \Xi_1^v(m_{10}^{\omega_i}) \\ & & \ddots & & \\ \Xi_{N_{\omega_i}^v}^v(m_1^{\omega_i}) & \Xi_{N_{\omega_i}^v}^v(m_2^{\omega_i}) & \Xi_{N_{\omega_i}^v}^v(m_3^{\omega_i}) & \cdots & \Xi_{N_{\omega_i}^v}^v(m_{10}^{\omega_i}) \\ \Xi_1^e(m_1^{\omega_i}) & \Xi_1^e(m_2^{\omega_i}) & \Xi_1^e(m_3^{\omega_i}) & \cdots & \Xi_1^e(m_{10}^{\omega_i}) \\ & & \ddots & & \\ \Xi_{N_{\omega_i}^e}^e(m_1^{\omega_i}) & \Xi_{N_{\omega_i}^e}^e(m_2^{\omega_i}) & \Xi_{N_{\omega_i}^e}^e(m_3^{\omega_i}) & \cdots & \Xi_{N_{\omega_i}^e}^e(m_{10}^{\omega_i}) \\ \Xi_1^I(m_1^{\omega_i}) & \Xi_1^I(m_2^{\omega_i}) & \Xi_1^I(m_3^{\omega_i}) & \cdots & \Xi_1^I(m_{10}^{\omega_i}) \\ & & \ddots & & \\ \Xi_{N_{\omega_i}^I}^I(m_1^{\omega_i}) & \Xi_{N_{\omega_i}^I}^I(m_2^{\omega_i}) & \Xi_{N_{\omega_i}^I}^I(m_3^{\omega_i}) & \cdots & \Xi_{N_{\omega_i}^I}^I(m_{10}^{\omega_i}) \end{bmatrix}, \tag{4.50}$$

and

$$\mathbf{q}^{\omega_i,x} = \begin{bmatrix} q_1^{\omega_i,x} & q_2^{\omega_i,x} & \cdots & q_{10}^{\omega_i,x} \end{bmatrix}^T \text{ and } \mathbf{q}^{\omega_i,y} = \begin{bmatrix} q_1^{\omega_i,y} & q_2^{\omega_i,y} & \cdots & q_{10}^{\omega_i,y} \end{bmatrix}^T. \tag{4.51}$$

We can then rewrite Eqs. (4.45) and (4.46) as finding $\mathbf{q}^{\omega_i,x}$ and $\mathbf{q}^{\omega_i,y} \in \mathbb{R}^{10\times 1}$ such that

$$\begin{aligned} \mathbf{q}^{\omega_i,x} &= \underset{\mathbf{a}\in\mathbb{R}^{10\times 1}}{\operatorname{argmin}} \left[\mathbf{P}\mathbf{a} - \mathbf{b}^{\omega_i,x}\right]^T \left[\mathbf{P}\mathbf{a} - \mathbf{b}^{\omega_i,x}\right] \\ \mathbf{q}^{\omega_i,y} &= \underset{\mathbf{a}\in\mathbb{R}^{10\times 1}}{\operatorname{argmin}} \left[\mathbf{P}\mathbf{a} - \mathbf{b}^{\omega_i,y}\right]^T \left[\mathbf{P}\mathbf{a} - \mathbf{b}^{\omega_i,y}\right], \end{aligned} \tag{4.52}$$

where $\mathbf{b}^{\omega_i,x}$ and $\mathbf{b}^{\omega_i,y}$ are vectors consisting of both edge and internal DOFs of $u_{h,x}$ and $u_{h,y}$ in the patch, i.e.

$$\mathbf{b}^{\omega_i,x} = \begin{bmatrix} \Xi_1^v(u_{h,x}) & \cdots & \Xi_{N_{\omega_i}^v}^v(u_{h,x}) & \Xi_1^e(u_{h,x}) & \cdots & \Xi_{N_{\omega_i}^e}^e(u_{h,x}) & \Xi_1^I(u_{h,x}) & \cdots & \Xi_{N_{\omega_i}^I}^I(u_{h,x}) \end{bmatrix}^T . \tag{4.53}$$

and

$$\mathbf{b}^{\omega_i,y} = \begin{bmatrix} \Xi_1^v(u_{h,y}) & \cdots & \Xi_{N_{\omega_i}^v}^v(u_{h,y}) & \Xi_1^e(u_{h,y}) & \cdots & \Xi_{N_{\omega_i}^e}^e(u_{h,y}) & \Xi_1^I(u_{h,y}) & \cdots & \Xi_{N_{\omega_i}^I}^I(u_{h,y}) \end{bmatrix}^T . \tag{4.54}$$

The above equations yield

$$\mathbf{q}^{\omega_i,x} = (\mathbf{P}^T\mathbf{P})^{-1}\mathbf{P}^T\mathbf{b}^{\omega_i,x} \text{ and } \mathbf{q}^{\omega_i,y} = (\mathbf{P}^T\mathbf{P})^{-1}\mathbf{P}^T\mathbf{b}^{\omega_i,y}. \tag{4.55}$$

Having obtained coefficient vectors $\mathbf{q}^{\omega_i,x}$ and $\mathbf{q}^{\omega_i,y}$, we can reconstruct the displacement gradient $G_h\mathbf{u}_h$ at vertex $\mathbf{X}_i$ using Eq. (4.41).

Let us then look at the reconstruction of the mid-edge nodes. If we denote by $\widehat{\mathbf{X}}_{ij}$ the middle node of the edge connecting vertices $\mathbf{X}_i$ and $\mathbf{X}_j$, i.e. $\widehat{\mathbf{X}}_{ij} = 1/2(\mathbf{X}_i + \mathbf{X}_j)$, we define the recovered values of the displacement gradient at $\widehat{\mathbf{X}}_{ij}$ as the average value of the ones obtained from both patches associated with the two end vertices, namely,

$$\begin{aligned} G_h\mathbf{u}_h(\widehat{\mathbf{X}}_{ij}) &= \frac{1}{2}\left[\nabla\mathbf{p}^{\omega_i}(\widehat{\mathbf{X}}_{ij}) + \nabla\mathbf{p}^{\omega_j}(\widehat{\mathbf{X}}_{ij})\right] \\ &= \frac{1}{2}\begin{bmatrix} \frac{\partial p_x^{\omega_i}}{\partial X}(\widehat{\mathbf{X}}_{ij}) + \frac{\partial p_x^{\omega_j}}{\partial X}(\widehat{\mathbf{X}}_{ij}) & \frac{\partial p_x^{\omega_i}}{\partial Y}(\widehat{\mathbf{X}}_{ij}) + \frac{\partial p_x^{\omega_j}}{\partial Y}(\widehat{\mathbf{X}}_{ij}) \\ \frac{\partial p_y^{\omega_i}}{\partial X}(\widehat{\mathbf{X}}_{ij}) + \frac{\partial p_y^{\omega_j}}{\partial X}(\widehat{\mathbf{X}}_{ij}) & \frac{\partial p_y^{\omega_i}}{\partial Y}(\widehat{\mathbf{X}}_{ij}) + \frac{\partial p_y^{\omega_j}}{\partial Y}(\widehat{\mathbf{X}}_{ij}) \end{bmatrix}, \end{aligned} \tag{4.56}$$

where we recall that $\mathbf{p}^{\omega_i}$ and $\mathbf{p}^{\omega_j}$ are the reconstructed cubic vectorial functions for patch ω_i and ω_j, respectively. Note that the above construction is different than taking the average of the reconstructed gradient at the two vertices (a choice that would lead to a loss of accuracy).

Remark 4.4.1 By comparing with the definition (4.34) for linear elements, the ones (4.45) and (4.46) for quadratic elements also considers internal DOFs of $\mathbf{u}_h$ for each patch. This requires evaluating not only the edge DOFs of the basis functions $m_\alpha^{\omega_i}(\mathbf{X})$ but also their internal DOFs when forming the matrix $\mathbf{P}$ defined in (4.50). While the edge DOFs of $m_\alpha^{\omega_i}(\mathbf{X})$ are their values at the selected points of edges, the evaluation of their internal DOFs (the moments of $m_\alpha^{\omega_i}$) requires the use of at least a 3rd-order numerical integration rule, which imposes addition computational expense on the simulation (recall that only a 2nd order integration is needed to exactly compute the stiffness matrix). As we will demonstrate in the numerical examples, it is possible to neglect the consideration of internal DOFs in the definitions (4.45) and (4.46) without sacrificing too much accuracy in reconstructions. Doing this, however, leads to a more efficient gradient recovery scheme, as it avoids the integration of basis functions $m_\alpha^{\omega_i}(\mathbf{X})$ to obtain their internal DOFs as we form the matrix $\mathbf{P}$.

A displacement gradient recovery for virtual elements of arbitrary order

Although not numerically evaluated in this chapter, this subsection outlines an extension of the displacement gradient recovery scheme for virtual elements of arbitrary order k in both 2D and 3D. Similarly to before, this subsection first describes the reconstructions at the vertices, and then states the reconstructions at the edge nodes of the mesh.

For the vertex $\mathbf{X}_i$ with its associated patch ω_i, the first step of the gradient recovery scheme for kth order virtual elements is to reconstruct a polynomial function $\mathbf{p}^{\omega_i} \in [\mathscr{P}_{k+1}(\omega_i)]^d$ via a least square fitting of all the DOFs of $\mathbf{u}_h$ in ω_i. More specifically, in the 2D case, the polynomial function $\mathbf{p}^{\omega_i} = [p_x^{\omega_i}, p_y^{\omega_i}]^T$ is obtained following a similar definition as for quadratic virtual elements. In the component form, we define

$$p_x^{\omega_i} = \underset{\xi \in \mathscr{P}_{k+1}(\omega_i)}{\operatorname{argmin}} \left\{ \sum_{j=1}^{N_{\omega_i}^v} \left[\Xi_j^v(\xi) - \Xi_j^v(u_{h,x})\right]^2 + \sum_{j=1}^{N_{\omega_i}^e} \left[\Xi_j^e(\xi) - \Xi_j^e(u_{h,x})\right]^2 \right.$$
$$\left. + \sum_{j=1}^{N_{\omega_i}^I} \left[\Xi_j^I(\xi) - \Xi_j^I(u_{h,x})\right]^2 \right\} \tag{4.57}$$

$$p_y^{\omega_i} = \underset{\xi \in \mathscr{P}_{k+1}(\omega_i)}{\operatorname{argmin}} \left\{ \sum_{j=1}^{N_{\omega_i}^v} \left[\Xi_j^v(\xi) - \Xi_j^v(u_{h,y})\right]^2 + \sum_{j=1}^{N_{\omega_i}^e} \left[\Xi_j^e(\xi) - \Xi_j^e(u_{h,y})\right]^2 \right.$$
$$\left. + \sum_{j=1}^{N_{\omega_i}^I} \left[\Xi_j^I(\xi) - \Xi_j^I(u_{h,y})\right]^2 \right\}. \tag{4.58}$$

For the 3D case, because of the presence of face DOFs, the above component-wise definition of $\mathbf{p}^{\omega_i} = [p_x^{\omega_i}, p_y^{\omega_i}, p_z^{\omega_i}]^T$ is expanded to also include those face DOFs. In particular, we seek

$$p_x^{\omega_i} = \underset{\xi \in \mathscr{P}_{k+1}(\omega_i)}{\operatorname{argmin}} \left\{ \sum_{j=1}^{N_{\omega_i}^v} \left[\Xi_j^v(\xi) - \Xi_j^v(u_{h,x})\right]^2 + \sum_{j=1}^{N_{\omega_i}^e} \left[\Xi_j^e(\xi) - \Xi_j^e(u_{h,x})\right]^2 \right.$$
$$\left. + \sum_{j=1}^{N_{\omega_i}^I} \left[\Xi_j^I(\xi) - \Xi_j^I(u_{h,x})\right]^2 + \sum_{j=1}^{N_{\omega_i}^F} \left[\Xi_j^F(\xi) - \Xi_j^F(u_{h,x})\right]^2 \right\} \tag{4.59}$$

(and the analogous one for the other two components $p_y^{\omega_i}, p_z^{\omega_i}$) where we recall that the face DOF operator $\Xi_i^F(\cdot)$, $i = 1, ..., N_{\omega_i}^F$, takes the moments of its argument up to order $k-2$ on the faces in patch ω_i. In practice, the polynomial function $\mathbf{p}^{\omega_i}$ is calculated following identical procedures described in the preceding subsections for linear and quadratic virtual elements, namely, using the matrix $\mathbf{P}$, which contains the edge, internal (and face, in 3D) DOFs of the basis functions $m_\alpha^{\omega_i}$ of $\mathscr{P}_{k+1}(\omega_i)$, and vectors $\mathbf{b}^{\omega_i,x}$, $\mathbf{b}^{\omega_i,y}$ and $\mathbf{b}^{\omega_i,z}$, which consist of those DOFs of the X, Y and Z components of $\mathbf{u}_h$, respectively. For the sake of brevity, those procedures are not repeated here.

Once we obtain the polynomial function $\mathbf{p}^{\omega_i}(\mathbf{X})$ for each patch ω_i, the recovered displacement gradient at vertex $\mathbf{X}^i$ is computed by evaluating $\mathbf{p}^{\omega_i}$ at this vertex, which is identical to the linear and quadratic cases, i.e.

$$G_h\mathbf{u}_h(\mathbf{X}_i) = (\nabla\mathbf{p}^{\omega_i})(\mathbf{X}_i). \tag{4.60}$$

For higher-order virtual elements, i.e. $k \geq 2$, the gradient recovery scheme also seeks to reconstruct $G_h\mathbf{u}_h$ at $k-1$ internal points on each edge, which coincides with the internal integration points of the Gauss-Lobatto rule of order $k+1$ on that edge. Assuming that $\widehat{\mathbf{X}}_{ij}^{\ell}$, $\ell = 1,...,k-1$, is the ℓth internal point of the edge that connects vertices $\mathbf{X}_i$ and $\mathbf{X}_j$, we generalize a suitable definition of the values of $G_h\mathbf{u}_h$ at this point from (4.56) as follows [159]

$$G_h\mathbf{u}_h(\mathbf{X}_i) = \alpha_{ij}^{\ell}(\nabla\mathbf{p}^{\omega_i})(\widehat{\mathbf{X}}_{ij}^{\ell}) + (1-\alpha_{ij}^{\ell})(\nabla\mathbf{p}^{\omega_j})(\widehat{\mathbf{X}}_{ij}^{\ell}), \tag{4.61}$$

where α_{ij}^{ℓ} is the ratio of the distance between $\widehat{\mathbf{X}}_{ij}^{\ell}$ and $\mathbf{X}_i$, and the length of the edge, namely, $\alpha_{ij}^{\ell} = ||\widehat{\mathbf{X}}_{ij}^{\ell} - \mathbf{X}_i||/||\mathbf{X}_i - \mathbf{X}_j||$.

Remark 4.4.2 The displacement gradient recovery scheme is tailored for the *a posteriori* error estimator using the H^1-type semi-norms (4.31)–(4.32) defined on the skeleton of the mesh. To that end, we only reconstruct the displacement gradient on edges of the mesh, that leads to a computationally cheaper procedure. Nevertheless, by following the same lines, our recovery scheme could be easily extended in order to yield also the internal degree of freedom values of the reconstructed gradient.

Remark 4.4.3 According to **Remark 4.4.1**, it is possible to neglect the internal and facial DOFs of $\mathbf{u}_h$ in the recovery process. As suggested by the numerical examples for quadratic virtual elements, this may lead to almost identically reconstructed displacement gradients, and render the recovery scheme more efficient. For virtual elements of order $k \geq 3$, however, more investigations are needed.

Reconstruction of the displacement gradient $G_h\mathbf{u}_h$

In this subsection, we present the procedures to reconstruct a continuous displacement gradient on the mesh skeleton using the recovered values of $G_h\mathbf{u}_h$ at $k+1$ equally-spaced points on every edge in the mesh skeleton. In the remaining, for a given generic edge e in the skeleton, we denote $\mathbf{G}_e^\ell$ by the recovered values of the displacement gradient on the ℓth point on this edge.

The basic idea of the reconstruction is to interpolate the recovered values of $G_h\mathbf{u}_h$ at the $k+1$ edge nodes using C^0 continuous shape functions on each edge. More specifically, for a generic edge e, we interpolate $G_h\mathbf{u}_h$ as:

$$G_h\mathbf{u}_h(\xi) = \sum_{\ell=1}^{k+1} N_\ell^{(k)}(\xi)\mathbf{G}_e^\ell, \tag{4.62}$$

where $N_\ell^{(k)}(\xi)$ is the 1D Lagrangian shape function (of order k) associated with the ℓ edge node on e, with ξ varying from 0 to L_e, in which L_e is the length of edge e.

In Fig. 4.1, we provide an illustration of the displacement gradient reconstruction for linear virtual elements, together with the adopted 1D shape functions on a generic edge e.

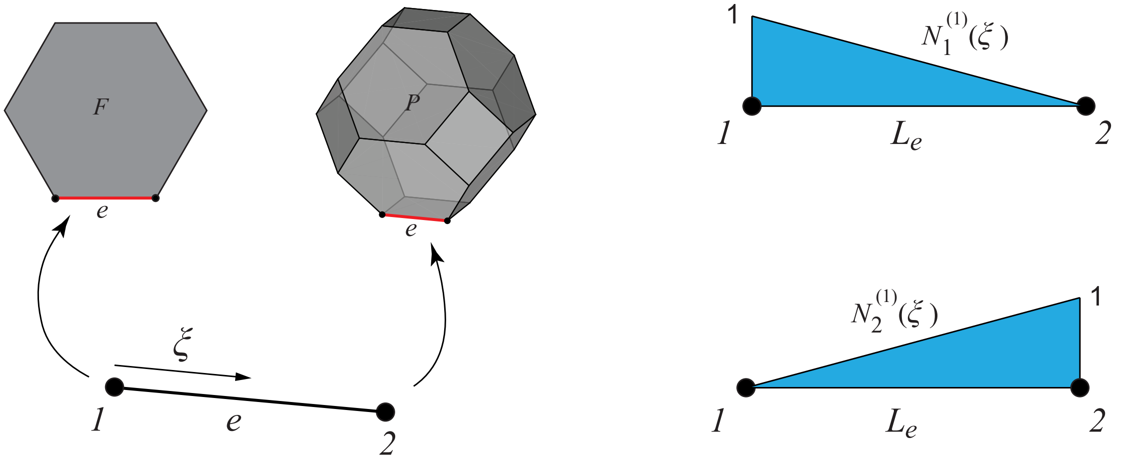


Figure 4.1: An illustration of the displacement gradient reconstruction for linear virtual elements in 2D and 3D. The displacement gradient on each edge e is interpolated from the reconstructed nodal values using 1D linear shape functions.

As a side note, in addition to reconstructing a displacement gradient only on the mesh skeleton, one can also choose to reconstruct a displacement gradient over the entire Ω using suitable basis functions. For linear virtual elements in 2D, for instance, a continuous displacement gradient over Ω can be reconstructed by interpolating the values at vertices obtained from the recovery procedure using nodal vector-valued basis functions of degree 1. Alternatively, one can use both the recovered displacement[1] and displacement gradient on the vertices and perform a C^1 reconstruction of the displacement field using C^1 basis functions [48], [135]. The reconstructed displacement gradient is then taken as the gradient of the reconstructed displacement. The advantage of this alternative approach over the former one is that, in addition to giving a continuous displacement gradient, the alternative approach also provides a displacement with higher continuity (i.e. C^1) and potentially higher accuracy over the original displacement solutions (which is only C^0 continuous). Aside from the above comments, however, we stick to recovering a C^0 displacement gradient on the mesh skeleton in the remainder of the chapter.

4.4.2 Suitable choices of patches

In the gradient recovery scheme, the selection of patches is a crucial component. In the sequel, we describe the criteria of how to select the patches. For a vertex $\mathbf{X}_i$, we choose its associated patch in the same fashion as in the finite element literature. As illustrated in Fig. 4.2 (a) and Fig. 4.3 (a) for 2D and 3D cases respectively, the patch ω_i is defined to be the union of all the elements that connect to $\mathbf{X}_i$. In 3D, this patch definition also applies to the boundary facial vertices, as shown in Fig. 4.3 (b).

Similar to the finite element literature, special attention needs to be paid when selecting boundary patches as they are likely to contain insufficient DOFs to uniquely determine the polynomial functions in the recovery process (in this case, the matrix $\mathbf{P}^T\mathbf{P}$ associated with that patch is singular). For instance, if chosen based on the criterion stated above,

[1] Instead of evaluating the gradient of the reconstructed polynomial function at vertices as in (4.41), a recovered displacement is obtained by evaluating the reconstructed polynomial at the vertices

the patches associated with boundary vertices in 2D (cf. Fig. 4.2 (b)) and boundary edge vertices in 3D (cf. Fig. 4.3 (c)) will typically contain only two elements. Those patches are most likely to contain insufficient DOFs, especially for higher order elements. In addition, for general polygonal and polyhedral discretization, similar scenarios may occur even for patches that belong to internal vertices. Therefore, to ensure the robustness of the recovery scheme, we propose to monitor N_{ω}^{E}, which is the number of elements the patch contains. If N_{ω}^{E} is less than three, we propose to enlarge the vertex patch by including another layer of elements. An enlarged patch is defined to be the union of patches associated with all the vertices that the original patch contains. This criterion will automatically include the boundary patches (i.e. the patches associated with boundary vertices in 2D (cf. Fig. 4.2 (b)) and boundary edge vertices in 3D (cf. Fig. 4.3 (c))) that were discussed previously.[2]

4.4.3 Error evaluation and *a posteriori* error estimation for the VEM

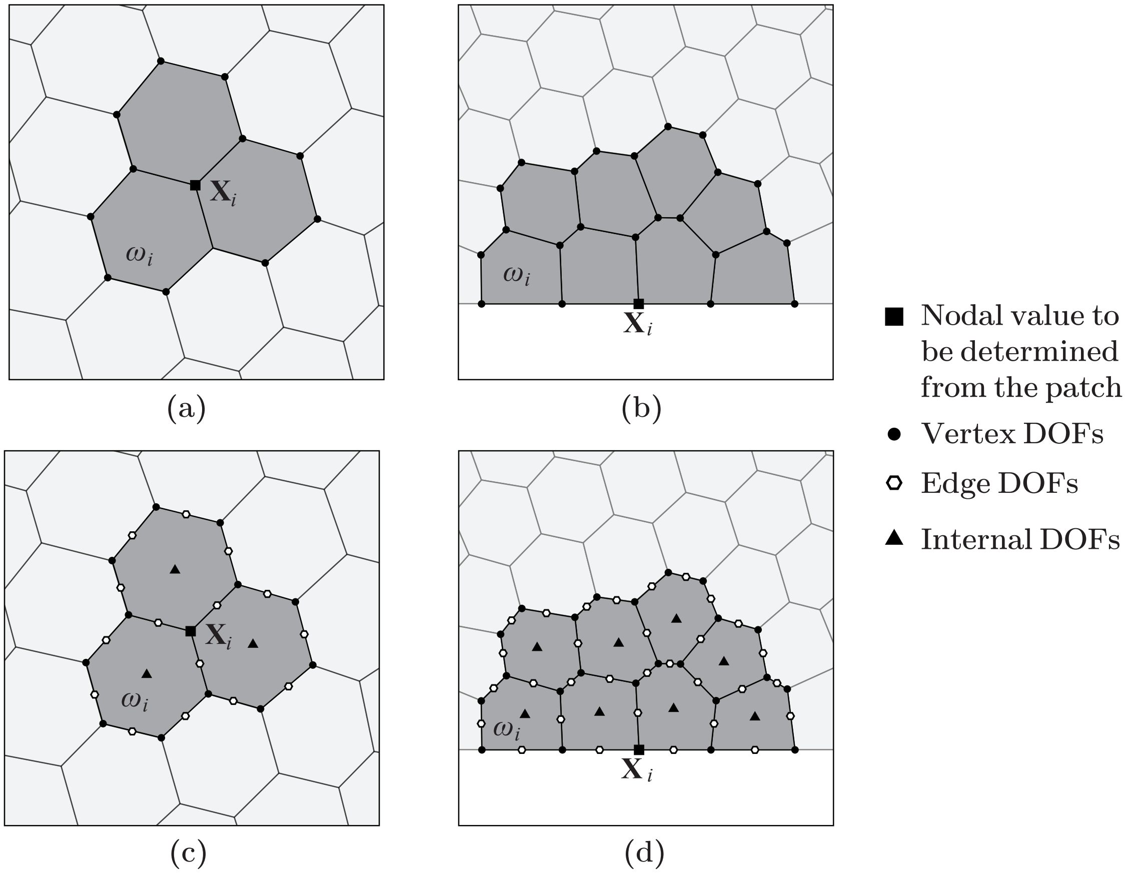
We discuss approaches to evaluate the errors of both the original and recovered displacement using the H^1-type skeletal norm defined in (4.31) and (4.32). On top of the discussions on error evaluations, we further present *a posteriori* error estimation based on the recovered displacement gradient $G_h \mathbf{u}_h$, which also makes use of the H^1-type skeletal norm defined in (4.31) and (4.32).

Error evaluation

According to (4.31) and (4.32), the original displacement error from VEM simulations using the H^1-type skeletal norm is given by

$$\varepsilon_{1,\mathbf{u}}^{s} = \left[\sum_{F \in \Omega_h} h_F \sum_{e \in \partial F} \int_e (\nabla \mathbf{u}_h \cdot \boldsymbol{\tau}_e - \nabla \mathbf{u} \cdot \boldsymbol{\tau}_e) \cdot (\nabla \mathbf{u}_h \boldsymbol{\tau}_e - \nabla \mathbf{u} \cdot \boldsymbol{\tau}_e) \mathrm{d}e \right]^{\frac{1}{2}} \qquad \text{in 2D,} \quad (4.63)$$

[2] An even safer procedure to select whenever to enlarge the vertex patches would be a direct check on the non-singularity of the matrix $\mathbf{P}^T \mathbf{P}$. Nevertheless, the procedure proposed here is cheaper and turns out to be robust in all the numerical experiments.



(a) (b)

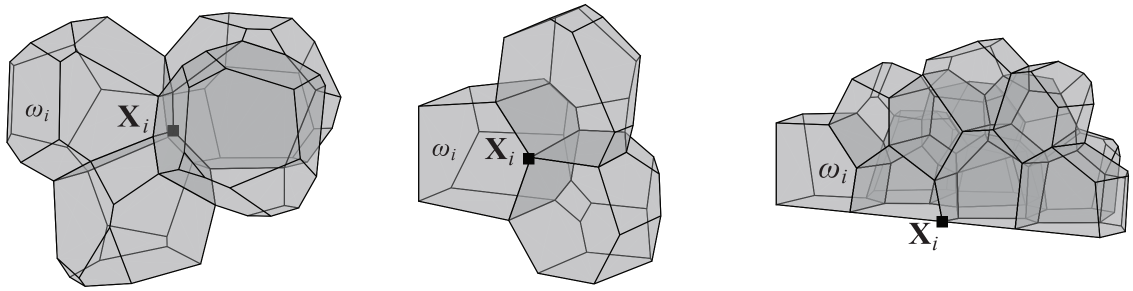
(c) (d)

Figure 4.2: Examples of (a) internal and (b) boundary patches in 2D for linear virtual elements. Examples of (c) internal and (d) boundary patches in 2D for quadratic virtual elements.

and

$$\varepsilon^s_{1,\mathbf{u}} = \left[\sum_{P \in \Omega_h} h_P \sum_{F \in \partial P} h_F \sum_{e \in \partial F} \int_e (\nabla \mathbf{u}_h \cdot \boldsymbol{\tau}_e - \nabla \mathbf{u} \cdot \boldsymbol{\tau}_e) \cdot (\nabla \mathbf{u}_h \boldsymbol{\tau}_e - \nabla \mathbf{u} \cdot \boldsymbol{\tau}_e) \mathrm{d}e \right]^{\frac{1}{2}} \qquad \text{in 3D,} \tag{4.64}$$

where $\mathbf{u}$ is the exact displacement. For the VEM of order k, the displacement solution $\mathbf{u}_h$ possesses kth order variation on a generic edge e, which can be interpolated from the edge DOFs of $\mathbf{u}_h$ on the $k+1$ equally spaced points of this edge (including the two end points). For this generic edge e, let us denote $\mathbf{U}^\ell_e$ as ℓth DOF of $\mathbf{u}_h$ on e. We can interpolate $\mathbf{u}_h$ and



Internal vertex patch
(a)

Boundary facial vertex patch
(b)

Boundary edge vertex patch
(c)

Figure 4.3: Examples of patches in 3D a polyhedral mesh: (a) an internal patch, (b) a patch on a face of the boundary, and (c) a patch on an edge of the boundary.

$\nabla\mathbf{u}_h \cdot \boldsymbol{\tau}_e$ along this edge as

$$\mathbf{u}_h(\xi) = \sum_{\ell=1}^{k+1} N_\ell^{(k)}(\xi)\mathbf{U}_e^\ell \text{ and } (\nabla\mathbf{u}_h \cdot \boldsymbol{\tau}_e)(\xi) = \sum_{\ell=1}^{k+1} \frac{\mathrm{d}N_\ell^{(k)}(\xi)}{\mathrm{d}\xi}\mathbf{U}_e^\ell, \tag{4.65}$$

where $N_\ell^{(k)}(\xi)$ is the 1D shape function (of order k) associated with $\mathbf{U}_e^\ell$ on e, with ξ varying from 0 to L_e, and L_e is the length of edge e. In practice, based on the above parametrization, the original displacement error is evaluated using a Gauss-Lobatto rule of a suitable order on each edge.

For the error evaluation of the recovered fields, the reconstruction $G_h\mathbf{u}_h$ in (4.62) is used, replacing $\nabla\mathbf{u}_h$ in the original displacement error (4.63) – (4.64). In practice, similar to the original displacement error discussed above, the skeletal error of the recovered displacement gradient is evaluated as

$$\varepsilon_{1,\tilde{\mathbf{u}}}^s = \left[\sum_{F\in\Omega_h} h_F \sum_{e\in\partial F} \int_e (G_h\mathbf{u}_h \cdot \boldsymbol{\tau}_e - \nabla\mathbf{u}\cdot\boldsymbol{\tau}_e)\cdot(G_h\mathbf{u}_h\boldsymbol{\tau}_e - \nabla\mathbf{u}\cdot\boldsymbol{\tau}_e)\mathrm{d}e\right]^{\frac{1}{2}} \text{ in 2D,} \tag{4.66}$$

and

$$\varepsilon_{1,\tilde{\mathbf{u}}}^s = \left[\sum_{P\in\Omega_h} h_P \sum_{F\in\partial P} h_F \sum_{e\in\partial F} \int_e (G_h\mathbf{u}_h \cdot \boldsymbol{\tau}_e - \nabla\mathbf{u}\cdot\boldsymbol{\tau}_e)\cdot(G_h\mathbf{u}_h\boldsymbol{\tau}_e - \nabla\mathbf{u}\cdot\boldsymbol{\tau}_e)\mathrm{d}e\right]^{\frac{1}{2}} \text{ in 3D,} \tag{4.67}$$

where we interpolate $G_h\mathbf{u}_h$ along edge e as

$$(G_h\mathbf{u}_h)(\xi) = \sum_{\ell=1}^{k+1} N_\ell^{(k)}(\xi)\mathbf{G}_e^\ell \tag{4.68}$$

with $N_\ell^{(k)}(\xi)$ being the 1D Lagrangian shape function (of order k) associated with the ℓ edge node on e, ξ varying from 0 to L_e; and $\mathbf{G}_e^\ell$ being the recovered values of $G_h\mathbf{u}_h$ on the ℓth point on edge e. In practice, the above integrals can be evaluated using a Gauss-Lobatto rule of suitable order on each edge.

A recovery-based a posteriori error estimator

In the proceeding subsection, the error evaluations make use of the exact displacement gradients $\nabla\mathbf{u}$, which are typically unknown in practice. Thus, this subsection proposes a recovery-based *a posteriori* error estimator, which is based on the reconstructed displacement gradient, to estimate the original skeletal errors (4.63) – (4.64) without knowing $\nabla\mathbf{u}$. The main idea behind this estimator is to replace the exact displacement gradient with the reconstructed one in the original error evaluations (4.63) – (4.64). Provided that the reconstructed displacement gradient is more accurate than the original ones, as expected, doing so should yield a reasonable estimation of the original error [148], [153].

The error estimator, denoted by $\widetilde{\varepsilon}^s_{1,\mathbf{u}}$, makes use of the reconstructed displacement gradient $G_h\mathbf{u}_h$ on the mesh skeleton. Over a generic element E, we can express the estimated error using as

$$\widetilde{\varepsilon}^s_{1,\mathbf{u}}|_E = \left\{ h_F \sum_{e\in\partial F} \int_e (G_h\mathbf{u}_h \cdot \boldsymbol{\tau}_e - \nabla\mathbf{u}_h \cdot \boldsymbol{\tau}_e) \cdot (G_h\mathbf{u}_h\boldsymbol{\tau}_e - \nabla\mathbf{u}_h \cdot \boldsymbol{\tau}_e)\mathrm{d}e \right\}^{\frac{1}{2}} \quad \text{in 2D,} \tag{4.69}$$

and

$$\widetilde{\varepsilon}^{s}_{1,\mathbf{u}}|_E = \left\{ h_P \sum_{F\in\partial P} h_F \sum_{e\in\partial F} \int_e (G_h\mathbf{u}_h \cdot \boldsymbol{\tau}_e - \nabla\mathbf{u}\cdot\boldsymbol{\tau}_e)\cdot(G_h\mathbf{u}_h\boldsymbol{\tau}_e - \nabla\mathbf{u}\cdot\boldsymbol{\tau}_e)\mathrm{d}e \right\}^{\frac{1}{2}} \qquad \text{in 3D,} \tag{4.70}$$

where the $G_h\mathbf{u}_u$ and $\nabla\mathbf{u}\cdot\boldsymbol{\tau}_e)$ are interpolated using (4.68) and (4.65) on every edge e of element E, respectively. In practice, the above integrals can be evaluated exactly using Gauss-Lobatto rules of at least order $2k$ on edge e. We note that the above local error estimations are useful in the adaptive refinement and coarsening analysis to pinpoint which regions to refine and which regions to coarsen [156]. Once the estimated error is known for each element, the estimated global error is computed by summing the local error from every element as:

$$\widetilde{\varepsilon}^{s}_{1,\mathbf{u}} = \left[\sum_{E\in\Omega_h} \left(\widetilde{\varepsilon}^{s}_{1,\mathbf{u}}|_E\right)^2 \right]^{\frac{1}{2}}. \tag{4.71}$$

In the subsequent numerical studies, we will verify the skeletal errors by comparing them with the standard error measures. To that end, we also consider the more standard L^2 errors of both the original and recovered displacement gradients given as

$$\varepsilon_{1,\mathbf{u}} = \left[\sum_{E\in\Omega_h} \int_E (\nabla\mathbf{u} - \nabla\mathbf{u}_h)\cdot(\nabla\mathbf{u} - \nabla\mathbf{u}_h)\mathrm{d}\mathbf{X} \right]^{\frac{1}{2}}, \tag{4.72}$$

and

$$\varepsilon_{1,\widetilde{\mathbf{u}}} = \left[\sum_{E\in\Omega_h} \int_E (\nabla\mathbf{u} - G_h\mathbf{u}_h)\cdot(\nabla\mathbf{u} - G_h\mathbf{u}_h)\mathrm{d}\mathbf{X} \right]^{\frac{1}{2}}, \tag{4.73}$$

respectively. In the L^2 errors above, both the original displacement $\mathbf{u}_h$ and recovered displacement gradients $G_h\mathbf{u}_h$ are interpolated using the barycentric coordinates – we use the mean value shape functions in 2D [97] and Wachspress shape functions in 3D [30] for the linear VEM, and the serendipity shape functions constructed from mean value shape functions based on [79] for the quadratic VEM. The Wachspress shape functions are only applicable to convex elements, whereas the mean value shape functions, as well as the serendipity shape functions constructed from them, are applicable to both convex and non-convex elements [22]. Regarding numerical integration, all the integrals are evaluated using a fifth order quadrature rule.

To conclude this section, we summarize in Table 4.1 the notation of all the error measures we have presented. The subsequent numerical examples will follow those notations.

Table 4.1: Summary of the notations of the errors used in the numerical studies.

	skeletal norm	L^2 norm
Err. of $\nabla\mathbf{u}_h$	$\varepsilon^s_{1,\mathbf{u}}$ (by Eqs. (4.63)–(4.64))	$\varepsilon_{1,\mathbf{u}}$ (by Eq. (4.72))
Err. of $G_h\mathbf{u}_h$	$\varepsilon^s_{1,\widetilde{\mathbf{u}}}$ (by Eqs. (4.66)–(4.67))	$\varepsilon_{1,\widetilde{\mathbf{u}}}$ (by Eqs. (4.73))
Estimated err. of $\nabla\mathbf{u}_h$	$\widetilde{\varepsilon}^s_{1,\mathbf{u}}$ (by Eqs. (4.69)–(4.70))	n/a

4.5 Some theoretical estimates

In the present section we develop some theoretical estimates for the post-processed error $\varepsilon^s_{1,\widetilde{\mathbf{u}}}$ in (4.66). Note that the present derivations should not be intended as a proof, as they hinge on some reasonable but not demonstrated assumption. Instead, the purpose of this section is to give some theoretical backbone to our construction.

In the following, we will use the standard notation for L^p spaces and H^k Sobolev spaces. In order to keep the derivations simple, we will restrict our attention to the case of $k = 1$ in 2D. The generalization to 3D follows almost identically while higher order cases can be tackled following the same lines. In the following, we assume that the (family of) meshes are shape regular, in the sense [37] that every element is star shaped with respect to a ball of the uniformly comparable radius and every edge has a length that is uniformly comparable to the diameter of the parent elements. Under such assumptions, one can derive that (see for instance [166]) the original error (4.63) satisfies

$$\varepsilon^s_{1,\mathbf{u}} \leq Ch\|\mathbf{u}\|_{H^2(\Omega)}, \tag{4.74}$$

where here and in the sequel C will denote a generic constant, possibly different at each occurrence, independent of the mesh size h.

For any sufficiently regular tensor valued function $\mathbf{v}$, let us now define the operator $|||\cdot|||$ as

$$|||\mathbf{v}|||^2 = \sum_{F\in\Omega_h} |||\mathbf{v}|||_F^2 \quad \text{where} \quad |||\mathbf{v}|||_F^2 = h_F \sum_{e\in\partial F} \int_e ||\mathbf{v}\cdot\boldsymbol{\tau}_e||^2 \mathrm{d}e. \tag{4.75}$$

Note that, by definition, the error in the skeleton norm is given by:

$$\varepsilon^s_{1,\widetilde{\mathbf{u}}} = |||\nabla\mathbf{u} - G_h\mathbf{u}_h|||. \tag{4.76}$$

We recall that, given any element $F\in\Omega_h$, each of its vertices $\mathbf{X}_i$ (with i running in some integer set $\mathscr{I}$) is associated with a patch of elements ω_i (see Section 4.4.2). Moreover, we here denote by $\omega_F = \cup_{i\in\mathscr{I}}\omega_i$ the union of such patches. Note that the operator G_h depends only on the pointwise values of $\mathbf{u}_h$ at the vertices of Ω_h, and thus it can be obviously extended to any function space for which pointwise values make sense, e.g. $[H^2(\Omega)]^2$. Given any $\mathbf{w}\in[H^2(\Omega)]^2$ and any vertex $\mathbf{X}_i$ of the mesh, we denote by $\mathbf{p}_{\mathbf{w}}^{\omega_i}\in[\mathscr{P}_2(\omega_i)]^2$ the polynomial built by our least square procedure with input values given by evaluating $\mathbf{w}$ at the vertices of ω_i. The VEM function $G_h\mathbf{w}$ at $\mathbf{X}_i$ is then computed, as usual, as $G_h\mathbf{w}(\mathbf{X}_i) = \nabla\mathbf{p}_{\mathbf{w}}^{\omega_i}(\mathbf{X}_i)$. We moreover observe that our post-processing construction is $\mathscr{P}_2$ preserving, in the sense that for all $F\in\Omega_h$ it holds

$$\nabla\mathbf{p}|_F = G_h\mathbf{p}|_F \qquad \forall\mathbf{p}\in[\mathscr{P}_2(\omega_F)]^2. \tag{4.77}$$

We start by showing a stability result for the operator G_h. In order to do so, we need the following reasonable assumption (see **Remark 4.5.1**):

(A1) There exists a constant $C^\star$ such that, for all $\mathbf{w}\in[H^2(\Omega)]^2$, $F\in\Omega_h$ and all vertices $\mathbf{X}_i$ of F, it holds

$$||\mathbf{p}_{\mathbf{w}}^{\omega_i}||_{L^\infty(F)} \leq C^\star \max_{\mathbf{X}_j \text{ vertex in } \omega^i} ||\mathbf{w}(\mathbf{X}_j)||.$$

Let us now take any $\mathbf{w} \in [H^2(\Omega)]^2$. Given the definition of $|||\cdot|||_F$, first by a Holder inequality and then recalling that $G_h\mathbf{w}$ is linear on edges, we have

$$|||G_h\mathbf{w}|||_F^2 \leq Ch_F^2 \|G_h\mathbf{w}\|^2_{L^\infty(\partial F)} \leq Ch_F^2 \max_{\mathbf{X}_i \text{ vertex of } F} \|G_h\mathbf{w}(\mathbf{X}_i)\|^2 = Ch_F^2 \max_{\mathbf{X}_i \text{ vertex of } F} \|\nabla \mathbf{p}_{\mathbf{w}}^{\omega_i}(\mathbf{X}_i)\|^2,$$

where the last identity follows from the definition of G_h. From the above bound, first by a trivial inequality, then with a standard inverse estimate for polynomials (on star-shaped domains), we get

$$|||G_h\mathbf{w}|||_F^2 \leq Ch_F^2 \max_{\mathbf{X}_i \text{ vertex of } F} \|\nabla \mathbf{p}_{\mathbf{w}}^{\omega_i}\|^2_{L^\infty(F)} \leq C \max_{\mathbf{X}_i \text{ vertex of } F} \|\mathbf{p}_{\mathbf{w}}^{\omega_i}\|^2_{L^\infty(F)}.$$

We now conclude our stability estimate for G_h by applying assumption (**A1**)

$$|||G_h\mathbf{w}|||_F^2 \leq C \max_{\mathbf{X}_i \text{ vertex of } F} \left(\max_{\mathbf{X}_j \text{ vertex in } \omega^i} \|\mathbf{w}(\mathbf{X}_j)\|^2 \right) = C \max_{\mathbf{X}_j \text{ vertex in } \omega_F} \|\mathbf{w}(\mathbf{X}_j)\|^2 \qquad (4.78)$$

Let now a generic $F \in \Omega_h$. Using the $\mathscr{P}_2$ preserving property (4.77), adding/subtracting terms and the triangle inequality easily yield

$$|||\nabla\mathbf{u} - G_h\mathbf{u}_h|||_F \leq |||\nabla(\mathbf{u}-\mathbf{p})|||_F + |||G_h(\mathbf{p}-\mathbf{u})|||_F + |||G_h(\mathbf{u}-\mathbf{u}_h)|||_F =: T_1 + T_2 + T_3, \quad (4.79)$$

for any polynomial $\mathbf{p} \in [\mathscr{P}_2(\omega_F)]^2$, with the obvious meaning of the three terms T_1, T_2, T_3. The first term is bounded by a standard scaled trace inequality and approximation results for polynomials on star-shaped domains, yielding

$$T_1 \leq C\|\nabla(\mathbf{u}-\mathbf{p})\|_{L^2(F)} + h_F|\nabla(\mathbf{u}-\mathbf{p})|_{H^1(F)} \leq Ch_F^2|\mathbf{u}|_{H^3(F)}.$$

The second term is bounded using (4.78) and again standard approximation results for polynomials

$$T_2 \leq \max_{\mathbf{X}_j \text{ vertex in } \omega_F} \|\mathbf{p}(\mathbf{X}_j) - \mathbf{u}(\mathbf{X}_j)\| \leq Ch_{\omega_F}^2|\mathbf{u}|_{H^3(\omega_F)},$$

while again (4.78) yields

$$T_3 \leq C \max_{\mathbf{X}_j \text{ vertex in } \omega_F} \|\mathbf{u}(\mathbf{X}_j) - \mathbf{u}_h(\mathbf{X}_j)\|^2.$$

Now, by definition of $|||\cdot|||$ and bounding all diameters with h, it is easy to check that equation (4.79) and the bounds above for T_1, T_2, T_3 finally give

$$\varepsilon^s_{1,\widetilde{\mathbf{u}}} = |||\nabla\mathbf{u} - G_h\mathbf{u}_h||| \leq Ch^2|\mathbf{u}|_{H^3(\Omega)} + C\Big(\sum_{F\in\Omega_h} \max_{\mathbf{X}_j \text{ vertex in } \omega_F} \|\mathbf{u}(\mathbf{X}_j) - \mathbf{u}_h(\mathbf{X}_j)\|^2\Big)^{1/2}. \tag{4.80}$$

The above bound is to be compared with (4.74). It shows that the accuracy of the post-processed gradient $G_h\mathbf{u}_h$ depends directly on the values of $\mathbf{u} - \mathbf{u}_h$ at the *vertices* of the mesh. Therefore if such values are particularly accurate (as is typically the case), the post-processed gradient will strongly benefit. In the extremely positive case that $\mathbf{u}_h = \mathbf{u}_I$ (that is the interpolant of $\mathbf{u}$) the second term vanishes and one obtains an $O(h^2)$ convergence rate. The same happens in the presence of some super-convergence phenomena of $\mathbf{u}_h$ to $\mathbf{u}$ at vertices. In many situations, $G_h\mathbf{u}_h$ will be more accurate than $\nabla\mathbf{u}_h$ but still of the same order in h.

One can get a better understanding of the last term by assuming that all elements in the mesh have a comparable diameter (which is just to make the argument simple). Indeed, in such case, the number of elements in the mesh behaves as h^{-2} and thus we get immediately

$$\sum_{F\in\Omega_h} \max_{\mathbf{X}_j \text{ vertex in } \omega_F} \|\mathbf{u}(\mathbf{X}_j) - \mathbf{u}_h(\mathbf{X}_j)\|^2 \leq Ch^{-2} \max_{\mathbf{X}_j \text{ vertex in } \Omega_h} \|\mathbf{u}(\mathbf{X}_j) - \mathbf{u}_h(\mathbf{X}_j)\|^2.$$

Therefore the last term in (4.80) (considering also the square root) behaves as

$$h^{-1} \max_{\mathbf{X}_j \text{ vertex in } \Omega_h} \|\mathbf{u}(\mathbf{X}_j) - \mathbf{u}_h(\mathbf{X}_j)\|,$$

which (by standard approximation estimates for $(\mathbf{u} - \mathbf{u}_h)$ in the L^∞ norm) is guaranteed to be $O(h)$ for regular enough $\mathbf{u}$, but could be far better in the presence of a super convergence property at vertices.

Remark 4.5.1 Assumption (A1) represents the (uniform in the mesh family) stability of the polynomial least-square procedure, and depends on the position of the vertices in each patch ω_i. Roughly, it is sufficient that the set of vertices in each patch ω_i uniquely determines a least square $\mathscr{P}_2$ interpolant (that is, any $\mathscr{P}_2$ polynomial that takes value zero at all such points must vanish) and "stays far from" degenerate cases in which such condition is lost. Proving such an assumption would be possible focusing on given element patterns.

4.6 Numerical examples

In this section, both 2D and 3D numerical studies are presented to show the accuracy and effectiveness of the recovered displacement gradient and the (both local and global) recovery-based *a posteriori* error estimators. Various types of displacement solutions are considered here, including ones with both small and large (but finite) gradients, and additional one with singularity. Throughout, we set Young's modulus and Poisson's ratio of the solids to be $E_0 = 10$ and $\nu = 0.35$, respectively, and consider plane strain condition for the 2D examples. Consistent units are adopted throughout the chapter.

4.6.1 Numerical investigation in a 2D unit square

In the 2D numerical studies, we test the proposed scheme on a unit square domain $\Omega = (0,1)^2$, as depicted in Fig. 4.4 (a). We assume two exact displacement fields. The first displacement field $\mathbf{u} = [u_x, u_y]^T$ is given by

$$u_x = \sin(X)e^Y \text{ and } u_y = Y^2 - 2X, \tag{4.81}$$

which is referred to as the "smooth solution" due to its smoothness and regularity. The second displacement field $\mathbf{u}$ takes the form of [57]

$$u_x = 16X(1-X)Y(1-Y)\text{atan}(\frac{25X-100Y+50}{4}) \text{ and } u_y = 2X^2+4Y. \tag{4.82}$$

We refer to this displacement field as the "steep solution" as its X component has a sharp (but finite) gradient along the line $X-4Y+2=0$. In the numerical studies, the assumed exact displacements $\mathbf{u}$ are applied on the entire boundary of the unit square $\partial\Omega$. We consider various types of polygonal discretizations: random Voronoi, centroid Voronoi Tessellation (CVT), and structured hexagonal meshes, as shown in Figs. 4.4 (b)–(d), respectively. We test both linear and quadratic VEMs.

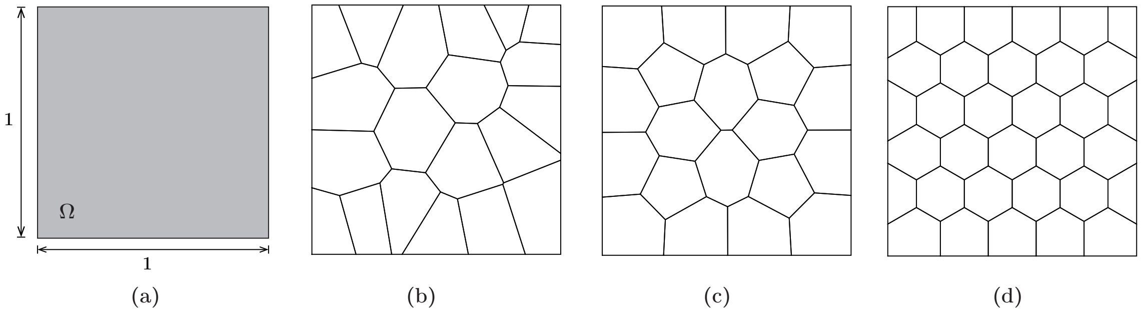


Figure 4.4: (a) Problem setup and dimensions of the unit square. (b) An example of the random Voronoi mesh. (c) An example of the CVT mesh. (d) An example of the structured hexagonal mesh.

Linear VEM ($k=1$)

Let us first verify the accuracy of the skeletal error by comparing it with the standard error measures. To that end, we plot in Figs. 4.5 (a)–(f) the convergence of the skeletal errors of both the original and reconstructed displacement gradients (solid lines) for the smooth and steep solutions. In the plots, those skeletal errors are compared with the standard L^2 error of the original and reconstructed displacement gradients (dashed lines). It is immediate from the comparisons that, for both the smooth and steep solutions, the proposed skeletal

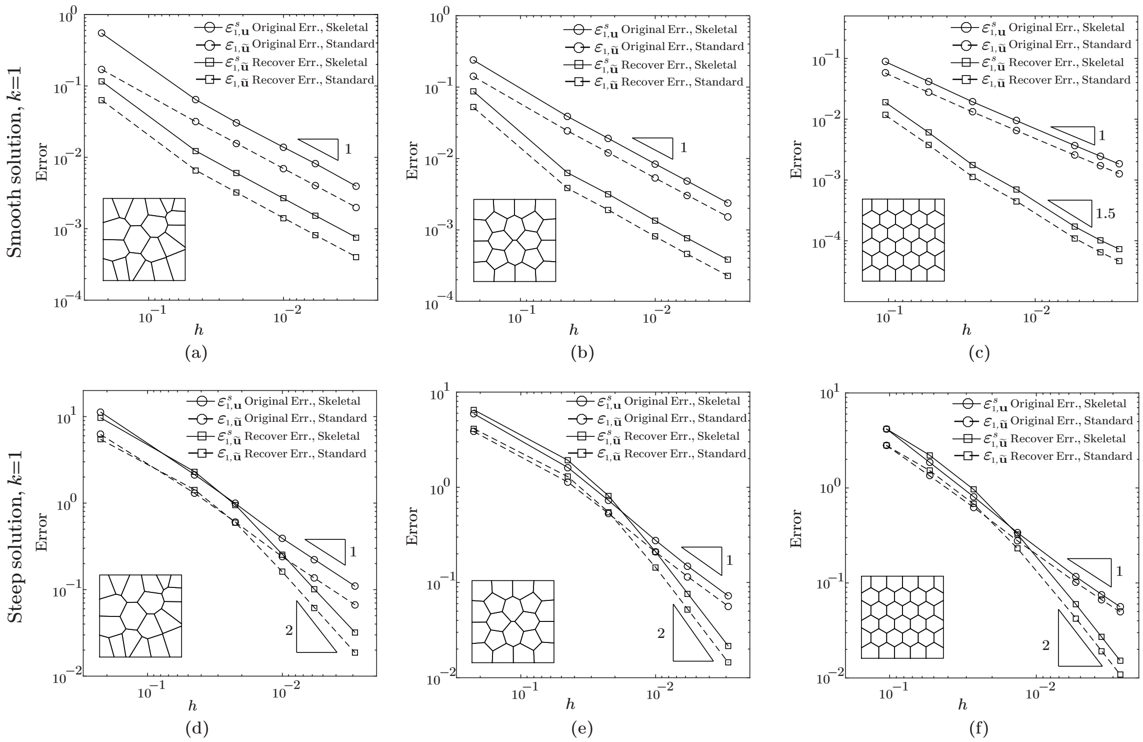
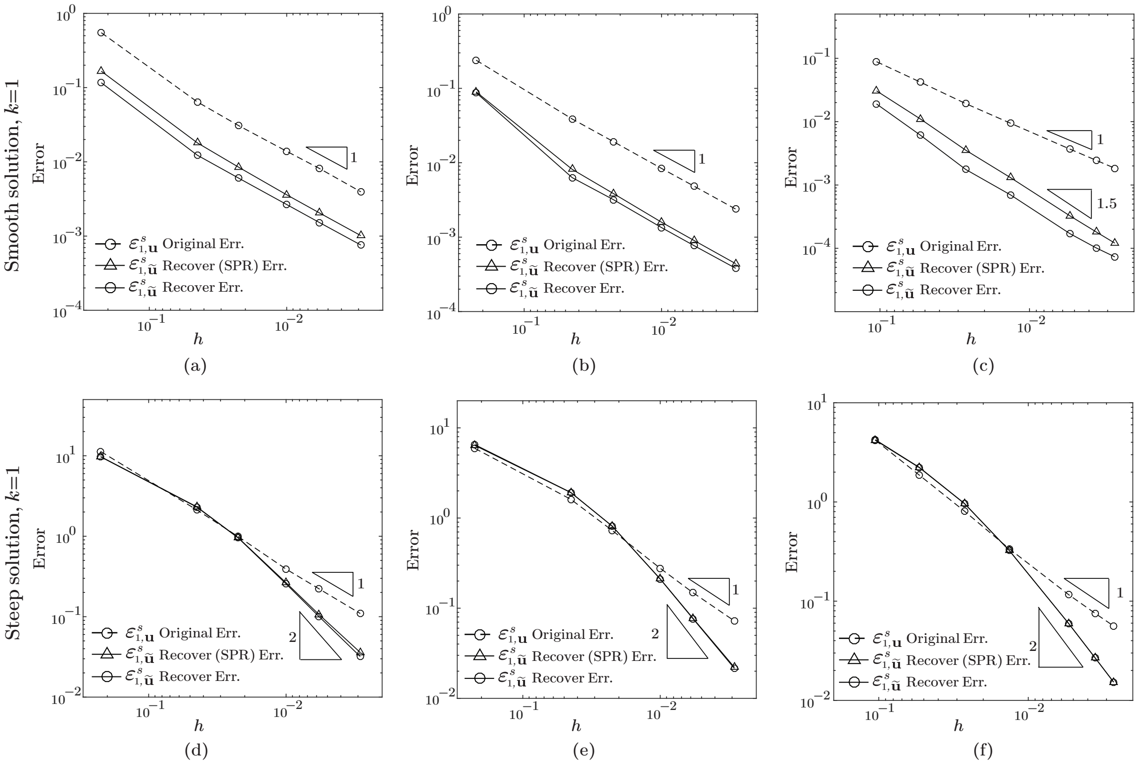


Figure 4.5: Comparing the convergence behaviors between the skeletal errors ($\varepsilon^s_{1,\mathbf{u}}$ and $\varepsilon^s_{1,\tilde{\mathbf{u}}}$) and the standard ones ($\varepsilon_{1,\mathbf{u}}$ and $\varepsilon_{1,\tilde{\mathbf{u}}}$) for 2D linear VEM. Smooth solution: (a) random Voronoi meshes, (b) CVT meshes, and (c) structured hexagonal meshes. Steep solution: (d) random Voronoi meshes, (e) CVT meshes, and (f) structured hexagonal meshes.

norm is able to capture the correct convergence behavior of the L^2 error of both the original and reconstructed displacement gradients. Thus, for the remainder of this study, only the skeletal errors will be utilized.

We then study the accuracy of the recovered displacement gradient $G_h\mathbf{u}_h$. To do so, we depict in Figs. 4.6 (a)–(f) the comparisons between the (global) errors of $G_h\mathbf{u}_h$ and those of the original displacement gradient $\nabla\mathbf{u}_h$ for the smooth and steep solutions. In the case of the smooth solution, we observe that $G_h\mathbf{u}_h$ is far more accurate than $\nabla\mathbf{u}_h$ for all families of meshes considered. Only for the structured hexagonal meshes, $G_h\mathbf{u}_h$ exhibits superconvergent behavior – the convergence rate being roughly 1.5 as compared to 1 for those of the original displacement gradient. On the other hand, in case of the steep solution, $G_h\mathbf{u}_h$ exhibits superconverent behaviors for all families of meshes – its errors start to converge at a higher convergence rate (i.e. rate of 2) with respect to the original displacement gradient



(a) (b) (c)

(d) (e) (f)

Figure 4.6: Comparing the accuracy between the original errors ($\varepsilon^s_{1,\mathbf{u}}$) and the recovered ones ($\varepsilon^s_{1,\tilde{\mathbf{u}}}$) for 2D linear VEM. The recovered errors include ones obtained from the proposed scheme and an SPR-type scheme introduced in Appendix C. Smooth solution: (a) random Voronoi meshes, (b) CVT meshes, and (c) structured hexagonal meshes. Steep solution: (d) random Voronoi meshes, (e) CVT meshes, and (f) structured hexagonal meshes.

(i.e. rate of 1) as the meshes are refined. Finally, we also perform an SPR-type recovery scheme (see the detailed description in Appendix C) and plot the errors of the reconstructed $G_h\mathbf{u}_h$ in Figs. 4.6 (a)–(f). By comparing the accuracy of $G_h\mathbf{u}_h$ obtained from the proposed recovery scheme and the SPR-type recovery scheme, we conclude that the proposed recovery scheme seems to yield more accurate reconstructions than the SPR-type recovery schemes.

Lastly, the accuracy of both global and local error estimators is studied. In Figs. 4.7 (a)–(f), we plot the convergence of the global error estimator computed from $G_h\mathbf{u}_h$ for the smooth and steep solutions on all families of meshes. For the local error estimators, we show in Figs. 4.8–4.10 fringe plots of the element-wise error estimators and exact errors for random Voronoi, CVT and structured hexagonal meshes, respectively. The compar-

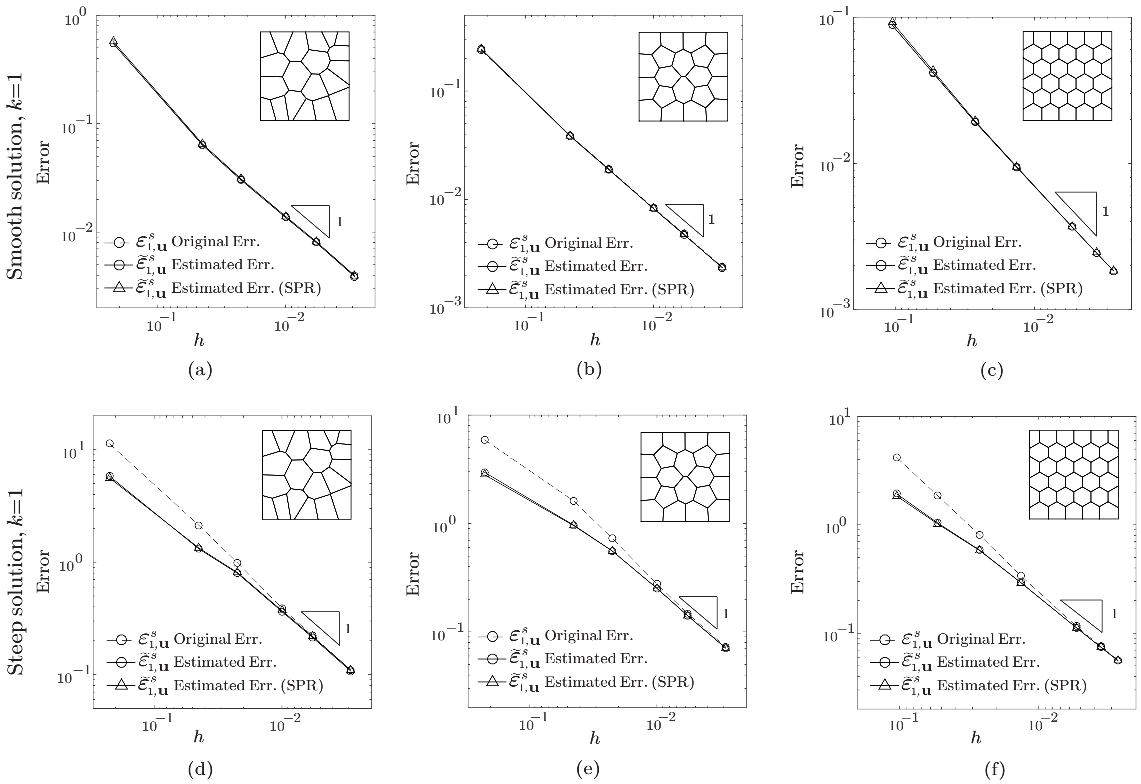


Figure 4.7: Comparing the accuracy between the original errors ($\varepsilon^s_{1,\mathbf{u}}$) and the estimated ones ($\widetilde{\varepsilon}^s_{1,\mathbf{u}}$) for 2D linear VEM. The estimated errors include ones obtained from the proposed scheme and an SPR-type scheme introduced in Appendix C. Smooth solution: (a) random Voronoi meshes, (b) CVT meshes, and (c) structured hexagonal meshes. Steep solution: (d) random Voronoi meshes, (e) CVT meshes, and (f) structured hexagonal meshes.

isons demonstrate that the error estimator introduced in this work is effective for various polygonal discretizations, and is capable of accurately estimating both the global and local displacement errors, especially for the smooth solution case. For the steep solution, although the error estimators are less accurate on coarse meshes, they quickly converge to the exact errors as the meshes are refined.

Quadratic VEM ($k = 2$)

For quadratic VEM, a similar set of numerical tests are conducted for both smooth and steep solutions on the three families of meshes shown in Fig. 4.4 (b)–(d). To begin with, the use of skeletal errors is verified by comparing them with the standard L^2 errors of the

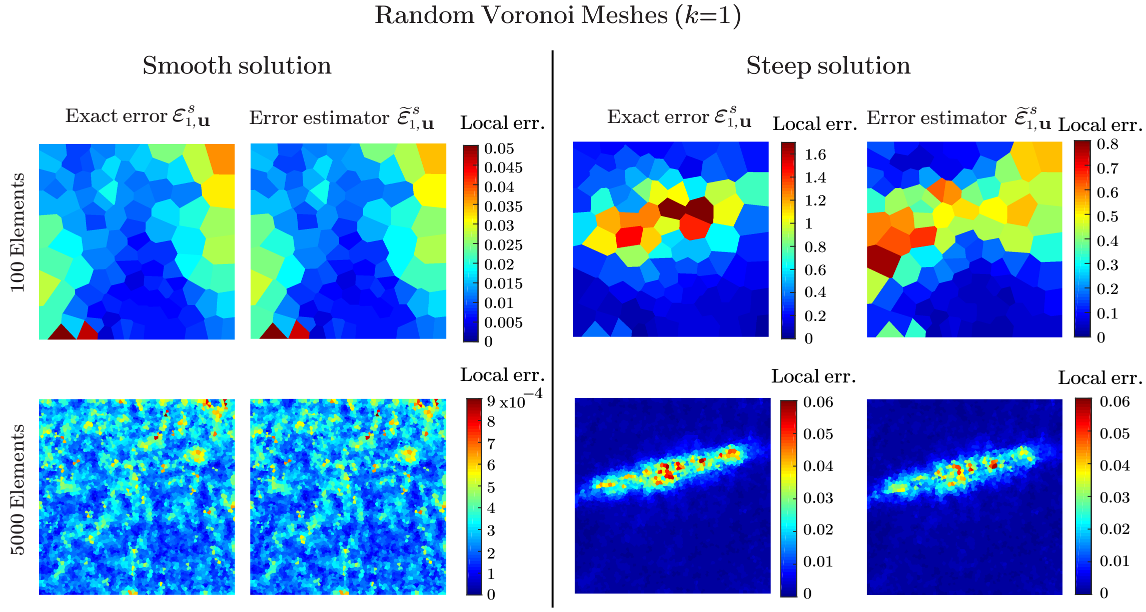


Figure 4.8: Fringe plots of the element-level exact errors and estimated ones for 2D linear VEM on random Voronoi meshes.

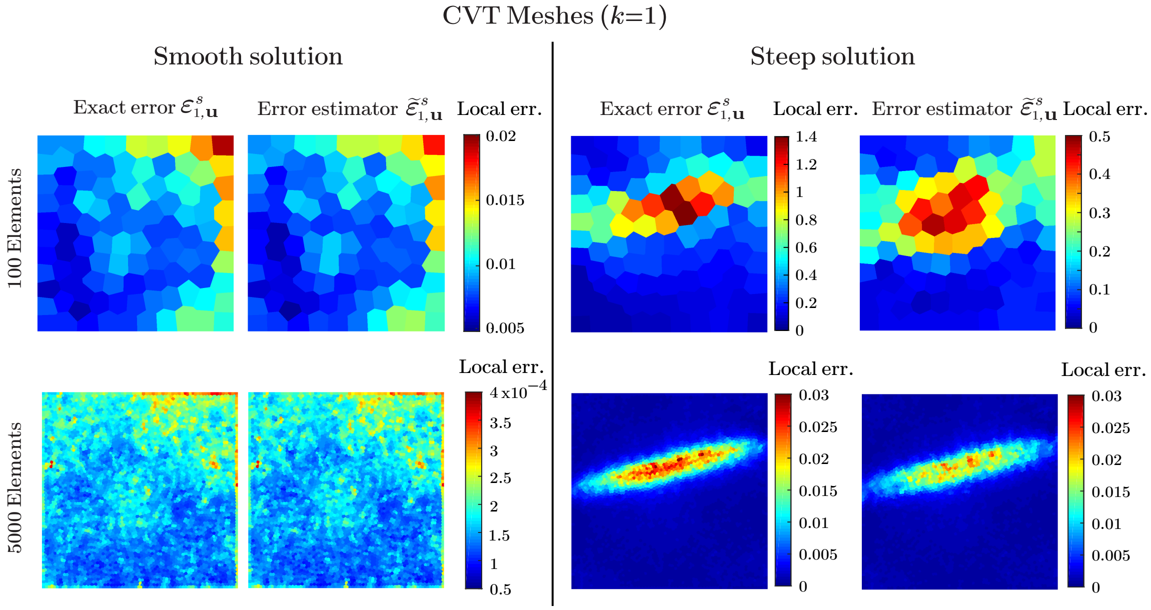


Figure 4.9: Fringes plots of the element-level exact errors and estimated ones for 2D linear VEM on CVT meshes.

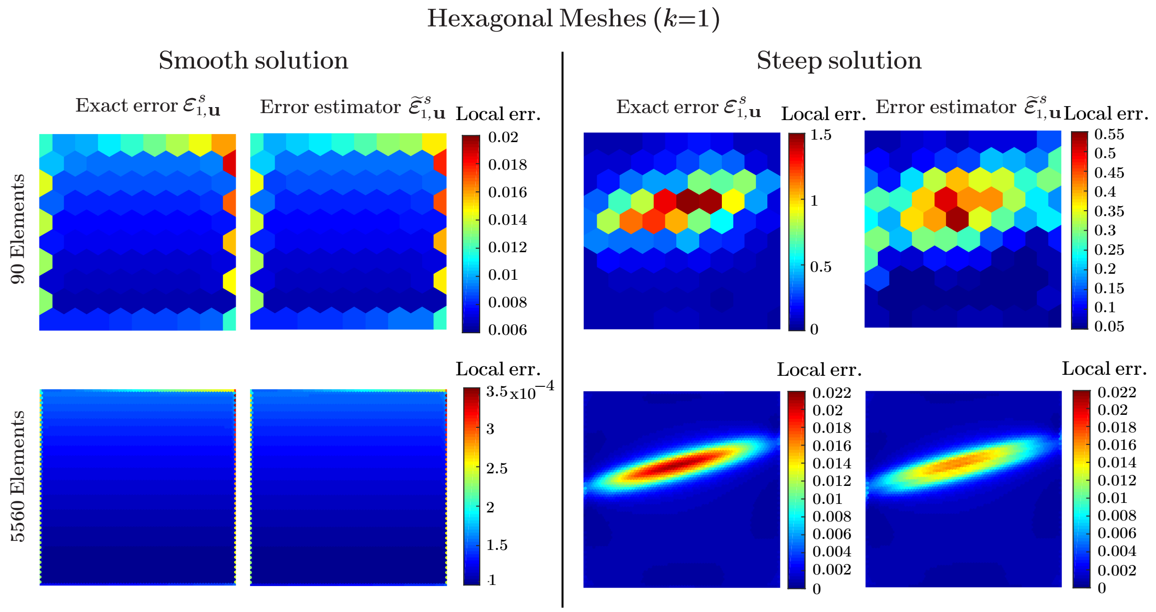


Figure 4.10: Fringes plots of the element-level exact errors and estimated ones for 2D linear VEM on hexagonal meshes.

displacement gradients. Figs. 4.11 (a)–(f) plot the skeletal errors of both original and the recovered displacement gradients and compare their convergence to the corresponding L^2 errors. From the comparisons, we observe that, for both steep and steep solutions, the skeletal errors of both the original and recovered displacement gradients agree well (in terms of rate and trend of convergence) with the standard L^2 errors. This observation provides us confidence in using the skeletal error in the following tests.

Furthermore, we plot in Figs. 4.12 (a)–(f) errors of the recovered displacement gradients obtained with and without the internal DOFs (see **Remark 4.4.1**) as functions of average mesh size h. The errors of the original displacement gradient are also included in the plot as references. We can see from the figures that neglecting the internal DOFs in the recovery scheme almost does not affect the accuracy of recovered displacement gradient. Thus, for the sake of computational efficiency, we conclude that it is favorable to neglect the internal DOFs in the recovery scheme. Additionally, similar to the observations in linear VEM, the recovered displacement gradients are more accurate than the original ones and, in case of the steep solution, they exhibit super-convergent behaviors – the rate

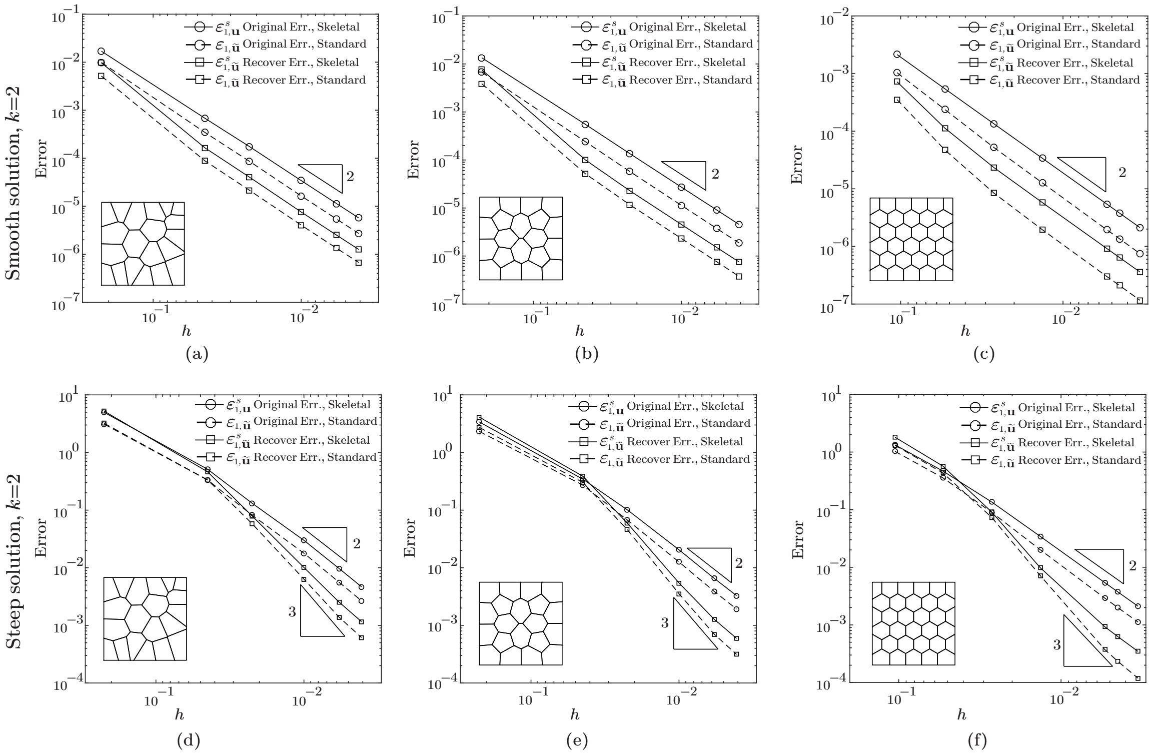


Figure 4.11: Comparing the convergence behaviors between the skeletal errors ($\varepsilon^s_{1,\mathbf{u}}$ and $\varepsilon^s_{1,\widetilde{\mathbf{u}}}$) and the more standard ones ($\varepsilon_{1,\mathbf{u}}$ and $\varepsilon_{1,\widetilde{\mathbf{u}}}$) for 2D quadratic VEM. Smooth solution: (a) random Voronoi meshes, (b) CVT meshes, and (c) structured hexagonal meshes. Steep solution: (d) random Voronoi meshes, (e) CVT meshes, and (f) structured hexagonal meshes.

of convergence is roughly 3.

Finally, we investigate the accuracy of both global and local error estimators by comparing them with the exact errors in cases both of the smooth and steep solutions. For the global error estimator, we depict in Figs. 4.13 (a)–(f) the convergence of both the estimated and exact errors as the meshes are refined and, for the local error estimator, Figs. 4.14–4.16 show the fringe plots of the element-wise distributions of the estimated and exact errors for random Voronoi, CVT and hexagonal meshes, respectively. It is apparent from both comparisons that the global and local error estimators can effectively and accurately predict the exact errors and, as the meshes are refined, the predictions become more and more accurate. Moreover, aligned with previous conclusions, the recovered displacement gradients obtained without considering the internal DOFs yield almost identical estimated errors to

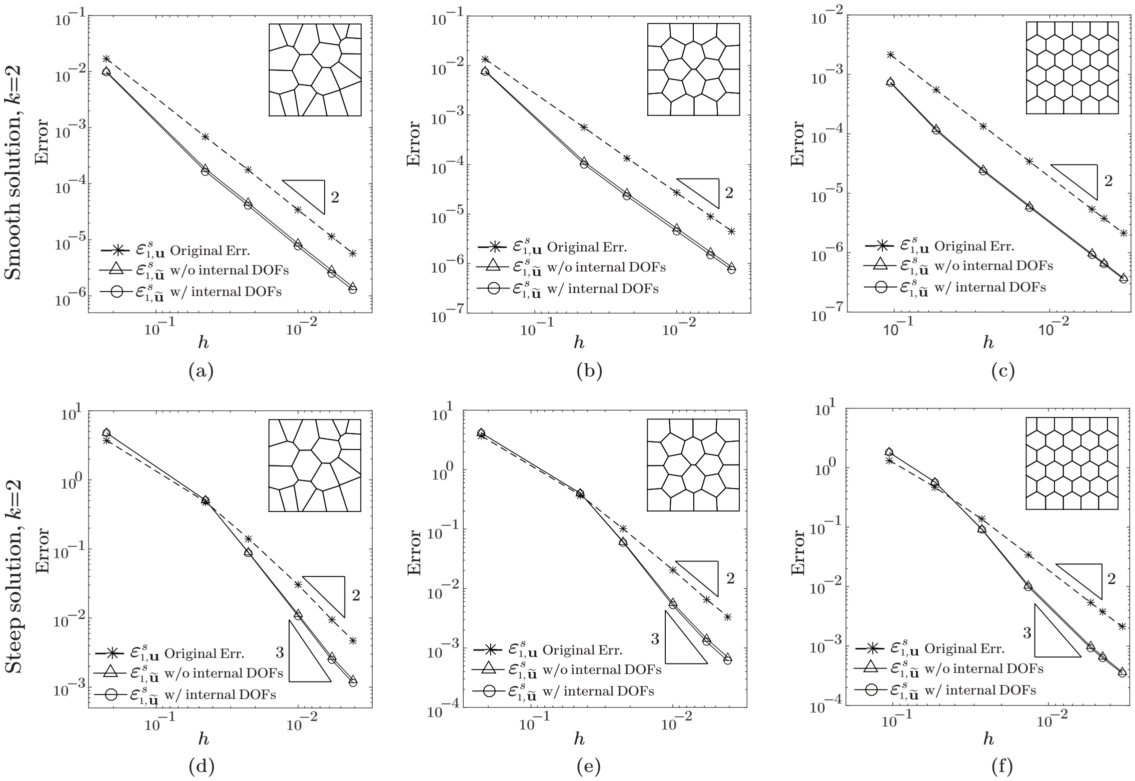


Figure 4.12: Comparing the accuracy between the original errors ($\varepsilon^s_{1,\mathbf{u}}$) and the recovered ones ($\varepsilon^s_{1,\tilde{\mathbf{u}}}$) for 2D quadratic VEM. The recovered errors include ones obtained from the proposed scheme and an SPR-type scheme introduced in Appendix C. Smooth solution: (a) random Voronoi meshes, (b) CVT meshes, and (c) structured hexagonal meshes. Steep solution: (d) random Voronoi meshes, (e) CVT meshes, and (f) structured hexagonal meshes.

the ones computed including the internal DOFs, which confirms again that we can neglect the internal DOFs in the recovery procedure (see **Remark 4.4.1**).

4.6.2 Numerical investigation in a 2D “L”-shaped domain

In this numerical example, we consider an L-shaped domain, whose dimensions are given in Fig. 4.17 (a). The domain is clamped on its top edge and subjected to a constant shear $\tau = 1$ pointing downward on its right edge. In this subsection, we will consider three families of polygonal meshes, i.e. random Voronoi, CVT and concave octagonal meshes, as shown in Figs. 4.17 (b)–(d), respectively. We will then assess the performance of the proposed gradient recover scheme and the global and local error estimators under uniform

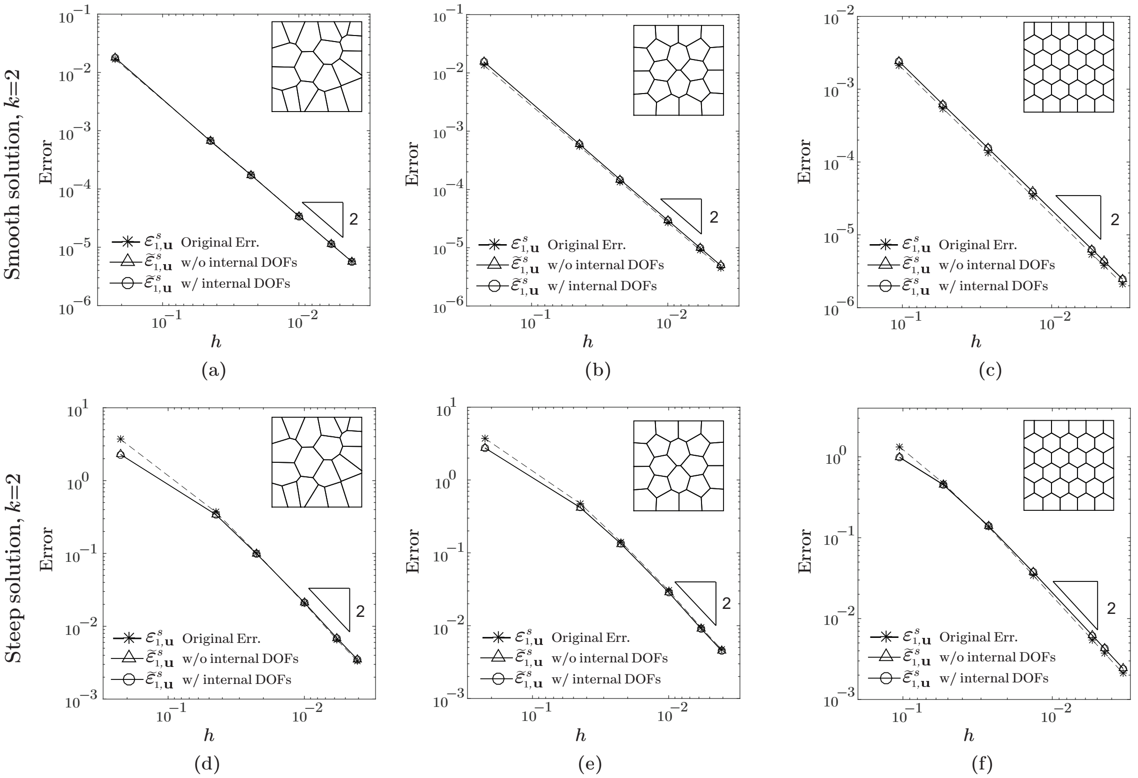


Figure 4.13: Comparing the accuracy between the original errors ($\varepsilon^s_{1,\mathbf{u}}$) and the estimated ones ($\widetilde{\varepsilon}^s_{1,\mathbf{u}}$) for 2D quadratic VEM. The estimated errors include ones obtained from the proposed scheme and an SPR-type scheme introduced in Appendix C. Smooth solution: (a) random Voronoi meshes, (b) CVT meshes, and (c) structured hexagonal meshes. Steep solution: (d) random Voronoi meshes, (e) CVT meshes, and (f) structured hexagonal meshes.

mesh refinement. In Section 7, we will restrict our attention to a CVT mesh and evaluate the effectiveness of the proposed error estimator under various adaptive mesh refinement strategies. For this boundary value problem, singularity exists in the displacement and stress solutions at the reentrant corner of the domain. Thus, one goal of this example is to showcase that the proposed error estimator is able to capture this singularity in the solution and can effectively drive the adaptive mesh refinement.

We first study the accuracy of the proposed gradient recovery scheme and the error estimators under uniform mesh refinement. Because we do not know the exact solution for this boundary value problem, only the estimated errors when linear ($k = 1$) and quadratic ($k = 2$) virtual elements are considered. For quadratic VEM, the gradient recovery is con-

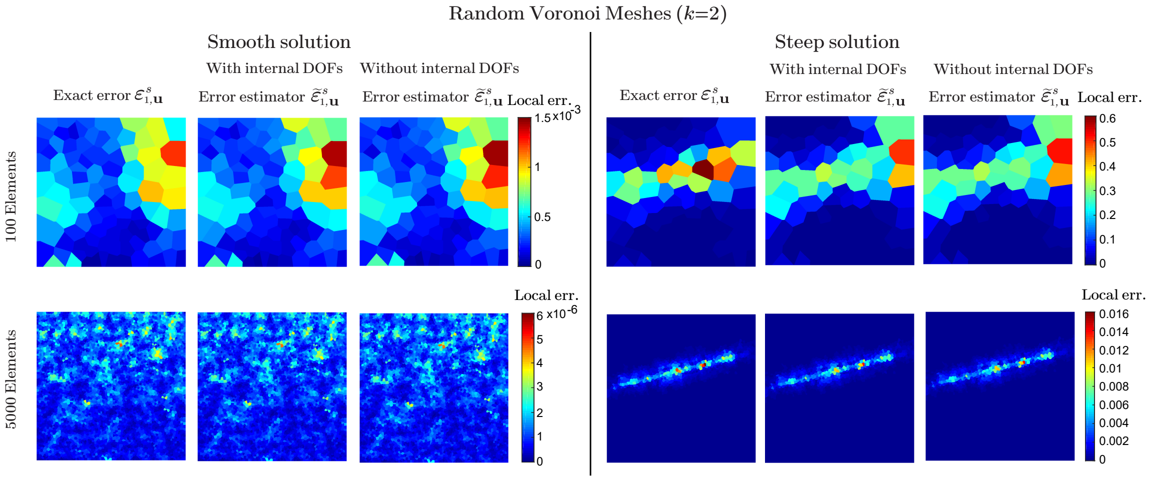


Figure 4.14: Fringes plots of the element-level exact errors and estimated ones for 2D quadratic VEM on random Voronoi meshes.

ducted excluding the internal DOFs. In Fig. 4.18 (a)–(c), we plot the convergence of estimated global errors for both linear and quadratic virtual elements on Random Voronoi, CVT, and octagonal meshes, respectively. We can see that, for all families of meshes, the estimated errors all converge at a degenerated rate of 0.5 irrespective of the order of the virtual element. This behavior agrees with theoretical results in the literature [167], indicating the effectiveness of the proposed error estimators. Moreover, we also depict the distributions of element-wise error estimators for linear and quadratic VEMs in fringe plots 4.19 and 4.20, respectively. It is immediate to appreciate that the proposed error estimators are able to capture the singularity of the displacement at the reentrant corner for both linear and quadratic meshes.

4.6.3 Numerical investigation in a 3D unit cube

In the last example, we present a 3D study on a unit cube $\Omega = (0,1)^3$, as depicted in Fig. 4.21. The exact displacement solution $\mathbf{u} = [u_x, u_y, u_z]^T$ is given by

$$u_x = \sin(X)e^{Y+2Z}, \quad u_y = \cos Z \quad \text{and} \quad u_z = X^3 - 2Y^2, \tag{4.83}$$

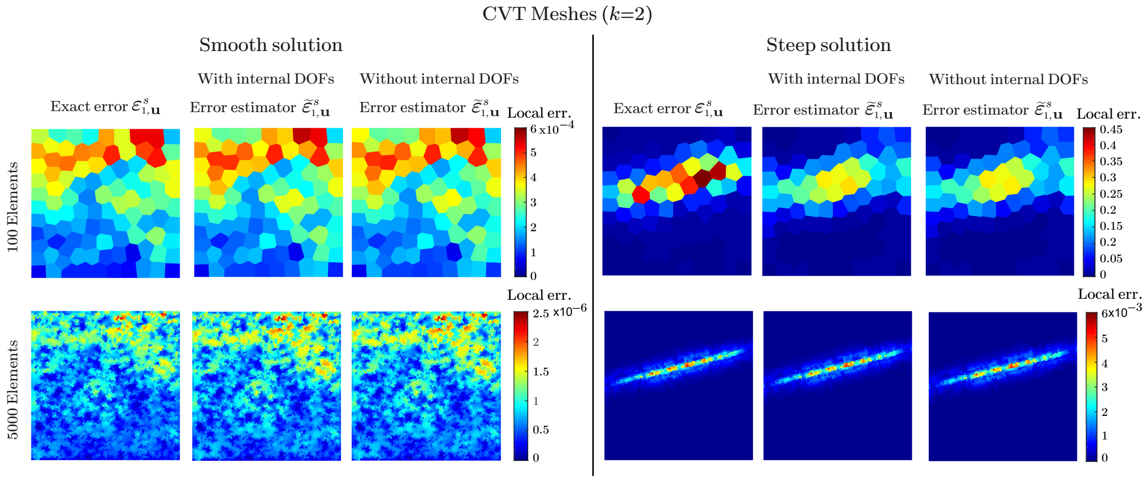


Figure 4.15: Fringes plots of the element-level exact errors and estimated ones for 2D linear VEM on CVT meshes.

which is applied to the entire boundary of the unit cube $\partial\Omega$. In this study, we consider various types of polyhedral discretizations: truncated octahedral, CVT, distorted hexahedral, and extruded octagonal meshes, as shown in Figs. 4.21 (b)–(e), respectively. Only linear virtual elements are considered here.

Similar to the 2D studies, we first verify the use of skeletal errors in 3D by comparing them with the standard L^2 errors of the displacement gradients. Figures 4.22 (a)–(d) show the comparison of these two types of errors for both original and recovered displacements on truncated octahedral, CVT, distorted hexahedral, and extruded octagonal meshes, respectively. The comparisons indicate that, for these mesh types, the skeletal errors agree well with the standard L^2 errors of the displacement gradients. Moreover, Figs. 4.22 (a)–(d) compare the accuracy between the original and recovered displacement gradients and showcase that, for all types of meshes, the recovered displacement gradients are more accurate than the original ones. It is also interesting to note that, for the extruded octagonal meshes, the recovered displacement gradient exhibits superconvergence – the rate of converge is 2. In addition, comparing with the SPR-type scheme introduced in Appendix C, the proposed recovery scheme consistently gives more accurate recovered displacement gradients on all types of meshes.

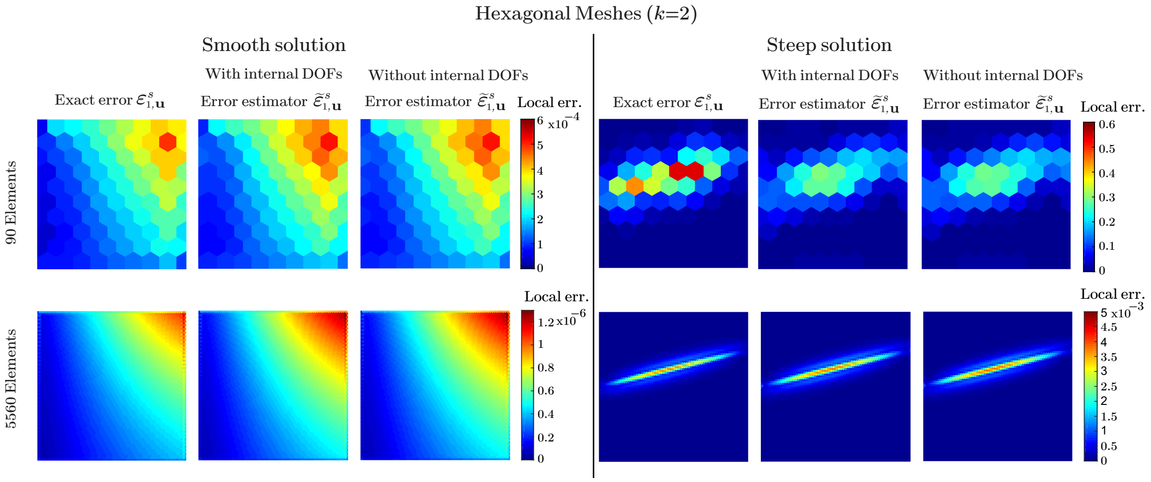


Figure 4.16: Fringes plots of the element-level exact errors and estimated ones for 2D linear VEM on hexagonal meshes.

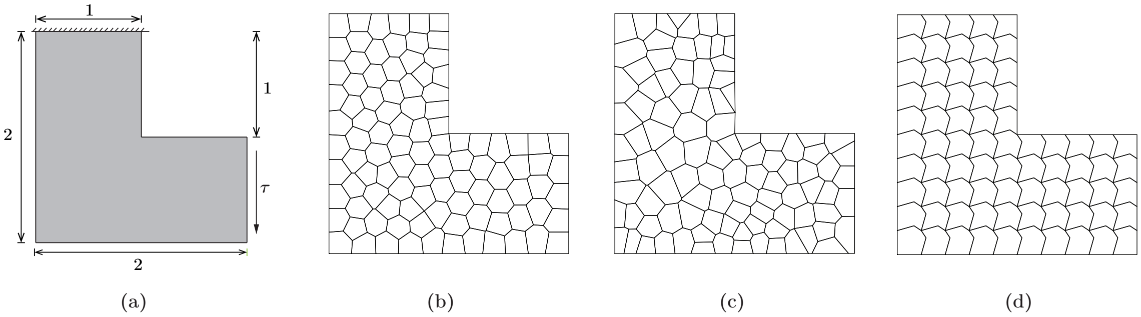


Figure 4.17: (a) Dimensions, load and boundary conditions of the L-shape beam example. (b) An example of the random Voronoi mesh. (c) An example of the CVT mesh. (d) An example of the concave octagonal mesh.

Lastly, we investigate the accuracy of error estimators by comparing them with the exact errors. To that end, Figs. 4.23 (a)–(d) show the convergence of both the estimated and exact global errors under mesh refinements for the truncated octahedral, CVT, distorted hexahedral, and extruded octagonal meshes, respectively. For the local error estimator, on the other hand, Figs. 4.24–4.27 show the fringe plots of the element-wise estimated and exact errors for those meshes. From the comparisons, we conclude that the error estimators (using both the proposed recovery scheme and the SPR-type scheme) can accurately predict the exact errors on both global and local levels.

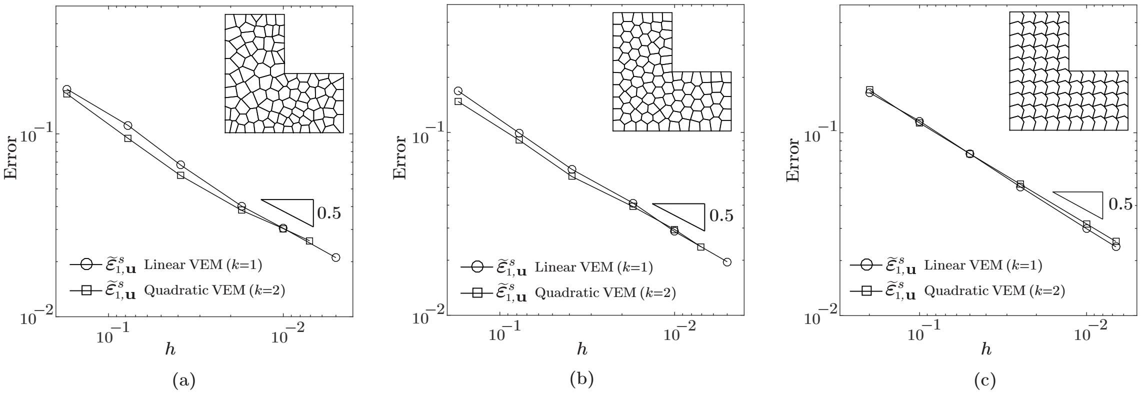


Figure 4.18: Convergence of the estimated errors as functions of the average mesh size h for linear and quadratic VEMs on (a) random Voronoi meshes, (b) CVT meshes, and (c) concave octagonal meshes.

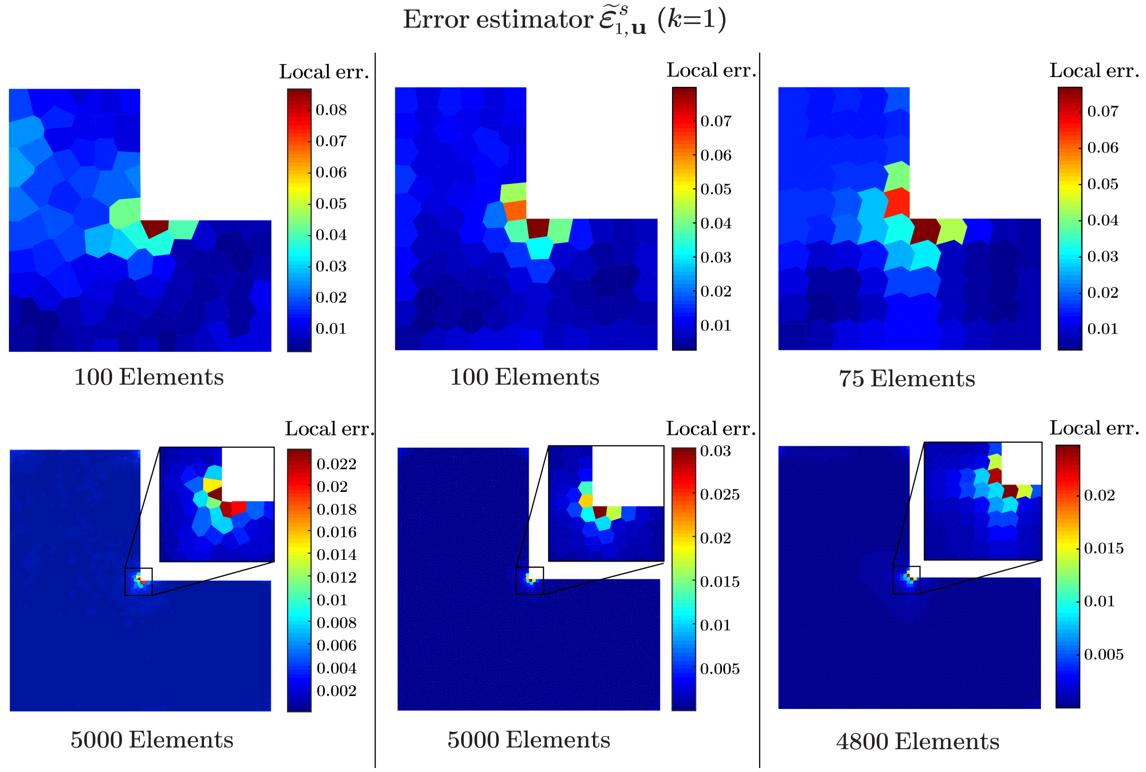


Figure 4.19: Fringe plots of element-level estimated errors for linear VEM on random Voronoi meshes (left column), CVT meshes (middle column) and concave octagonal meshes (right column).

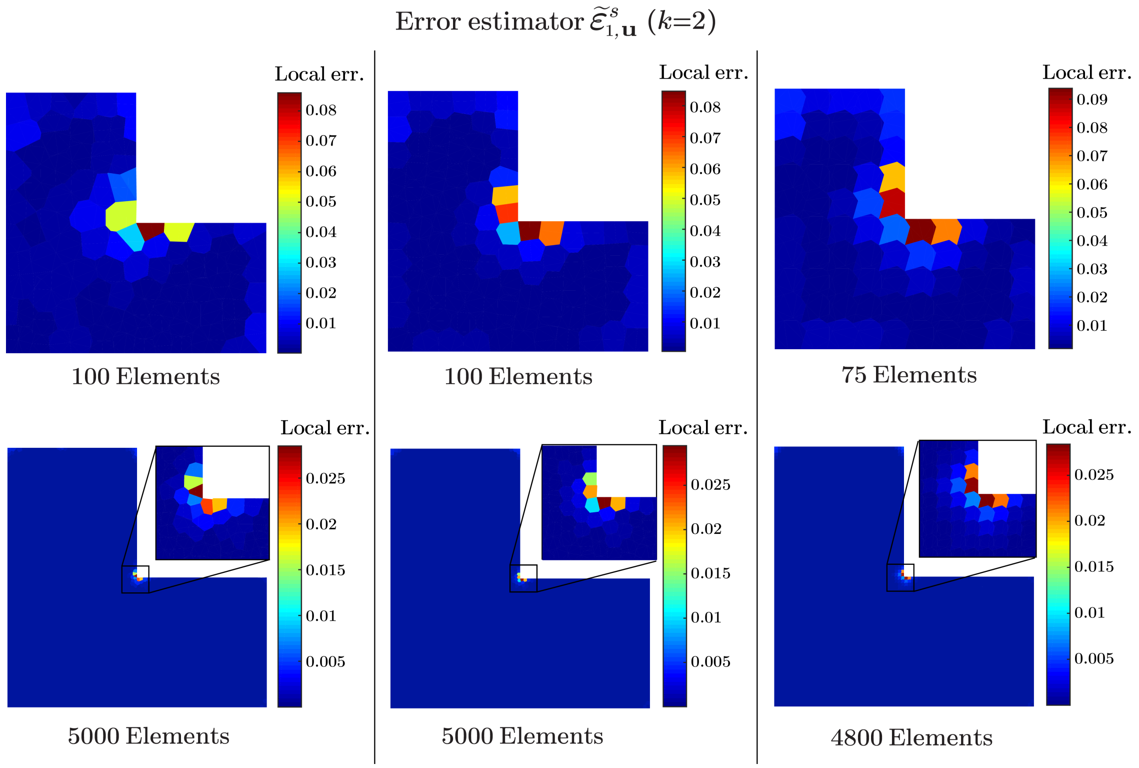


Figure 4.20: Fringe plots of element-level estimated errors for quadratic VEM on random Voronoi meshes (left column), CVT meshes (middle column) and concave octagonal meshes (right column).

4.7 On the use of the error estimators for adaptive mesh refinement

We study the effectiveness of the gradient recovery scheme and error estimators under adaptive mesh refinements. In this section, we restrict our attention to the CVT mesh and consider both linear and quadratic virtual elements. We consider the L-shaped problem in Section 6.2 and take the initial mesh to be the CVT one shown in Fig. 4.17 (c) with 100 elements. We then progressively refine this CVT mesh following a solve → estimate → mark → refine procedure: once the estimated error for each element is computed based on the VEM solution, we first mark those elements whose estimated errors are above $\theta \max_{E\in\Omega_h}(\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}|_E)$, where θ is a user-defined threshold set to be 0.2 throughout this example, and then perform refinement on those marked elements. In order to demonstrate the effectiveness of the proposed gradient recovery scheme and error estimators for a wide va-

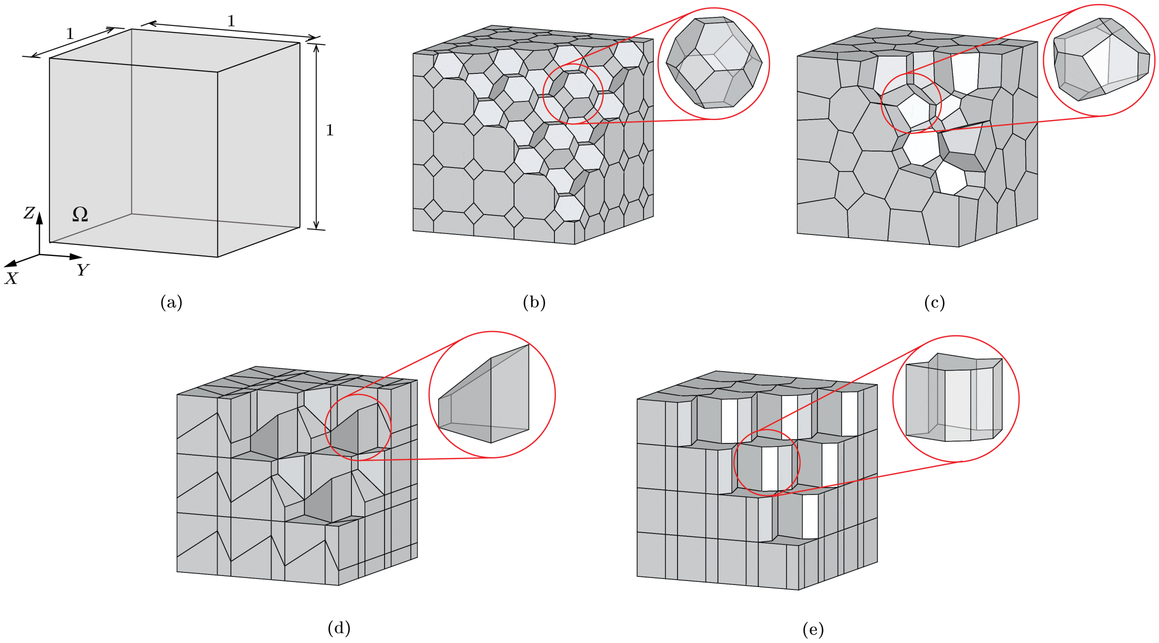


Figure 4.21: (a) Problem setup and dimensions of the unit cube. (b) An example of the truncated octahedral meshes. (c) An example of the CVT meshes. (d) An example of the distorted hexahedral meshes. (e) An example of the extruded octagonal meshes.

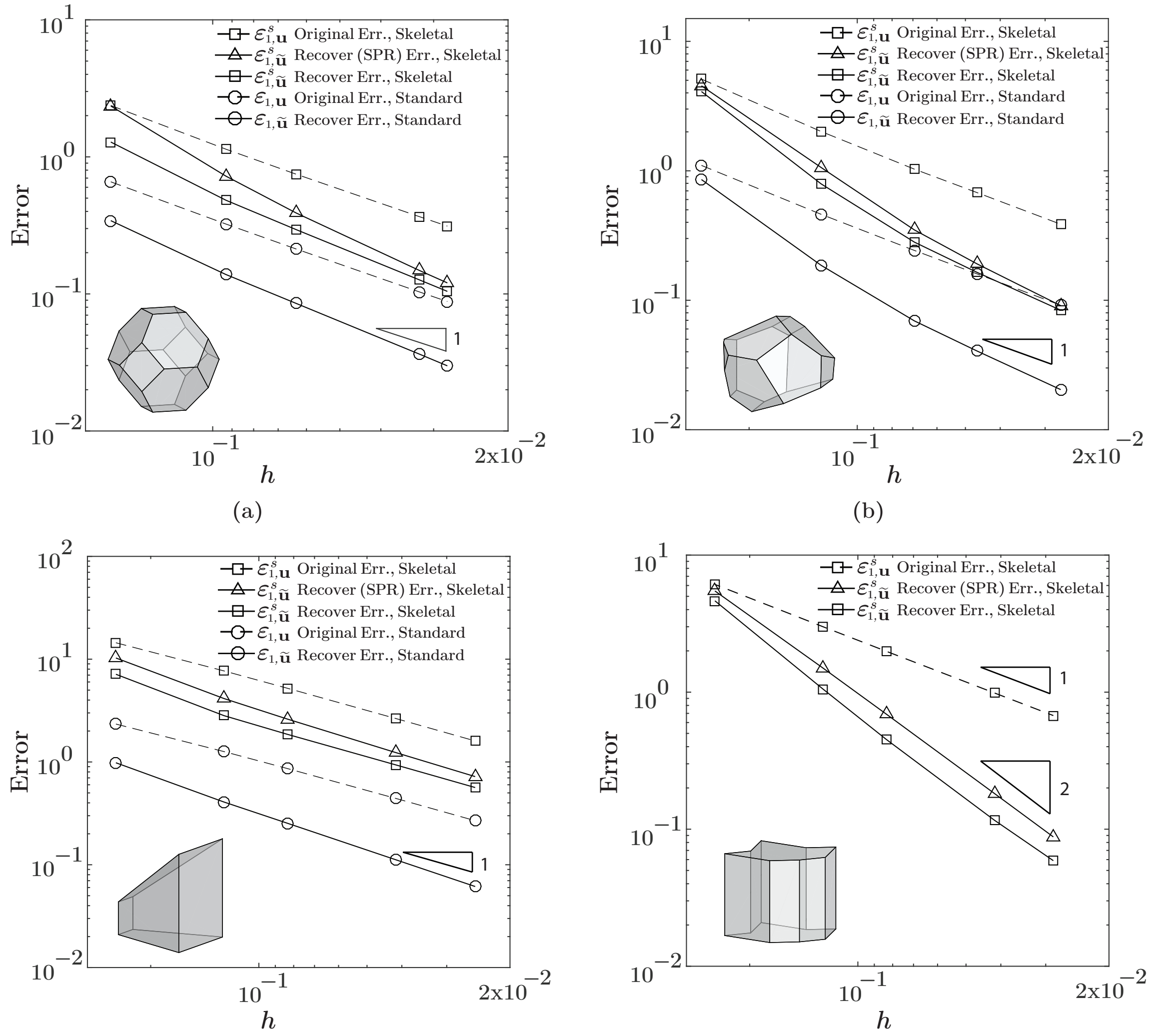
riety of adaptive refinement strategies, we consider the following three adaptive refinement strategies in the literature:

Strategy 1: In this strategy, each marked element is bisected through its centroid along the eigenvector corresponding to the smaller eigenvalue of the covariance matrix $\mathbf{M}_{\text{cov}}$ [168]. For element E, the covariance matrix $\mathbf{M}_{\text{cov}}(E)$ is defined as

$$\mathbf{M}_{\text{cov}}(E) = \frac{1}{|E|}\int_E (\mathbf{X}-\mathbf{X}_c^E)\cdot(\mathbf{X}-\mathbf{X}_c^E)\mathrm{d}\mathbf{X}, \tag{4.84}$$

where $\mathbf{X}_c^E$ is the centroid of E. A schematic illustration of this strategy is shown in Fig. 4.28 (b).

Strategy 2: In this strategy, each marked element with n vertices is subdivided into n quadrilateral elements by connecting the midpoint of each edge to its centroid [58], [169]. A schematic illustration of this strategy is shown in Fig. 4.28 (c).



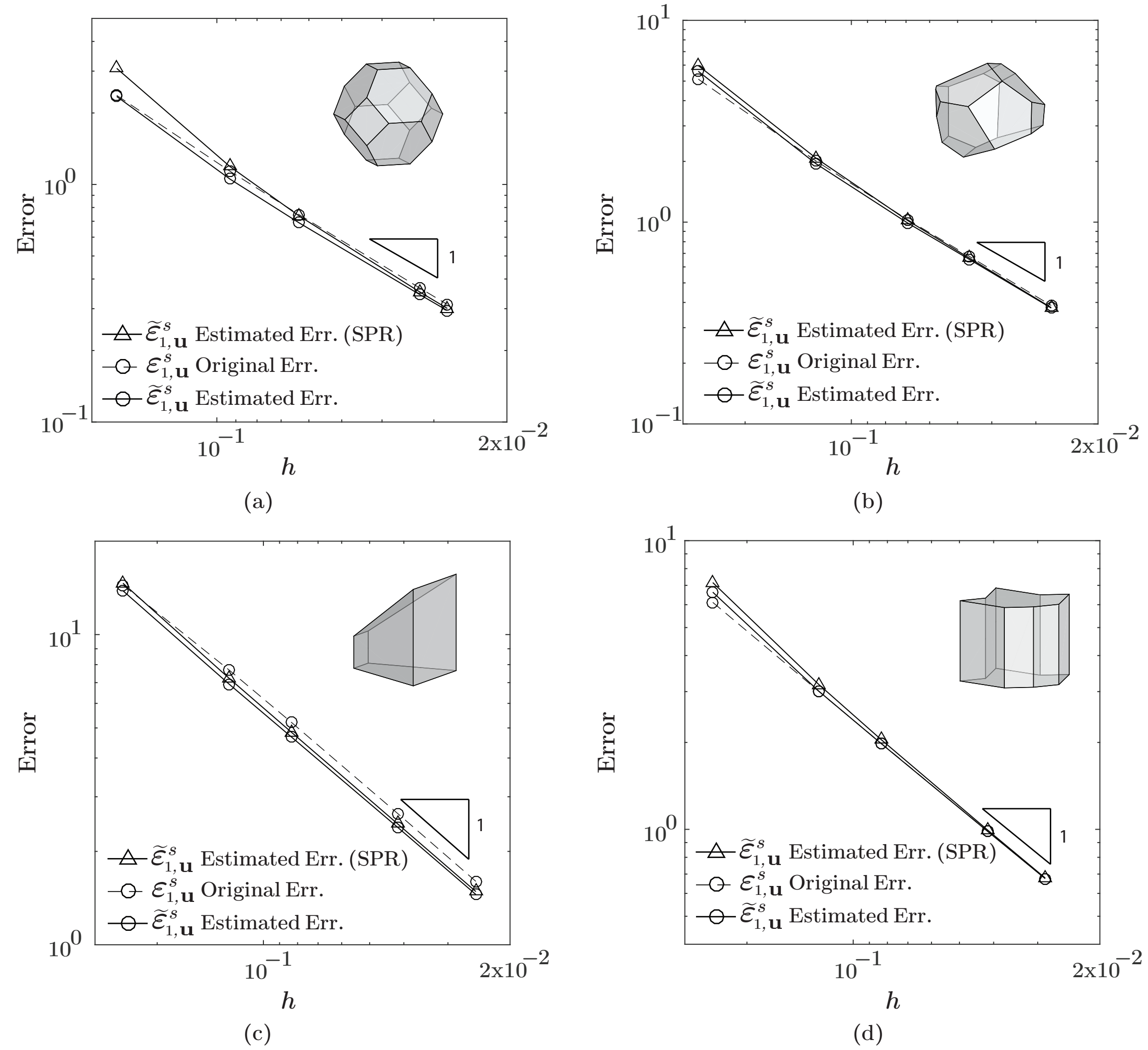
(a)

(b)

(c)

(d)

Figure 4.22: Comparing the convergence behaviors between the skeletal errors ($\varepsilon^s_{1,\mathbf{u}}$ and $\varepsilon^s_{1,\widetilde{\mathbf{u}}}$) and the standard ones ($\varepsilon_{1,\mathbf{u}}$ and $\varepsilon_{1,\widetilde{\mathbf{u}}}$) for 3D linear VEM on: (a) truncated octahedral meshes, (b) CVT meshes, (c) distorted hexahedral meshes, and (d) extruded octagonal meshes. Because the extruded octagonal meshes contain concave elements, the 3D Wachspress shape functions are not applicable [30]. Thus, we do not consider the standard error for the extruded octagonal meshes.



(a)

(b)

(c)

(d)

Figure 4.23: Comparing the accuracy between the original errors ($\varepsilon^s_{1,\mathbf{u}}$) and the estimated ones ($\widetilde{\varepsilon}^s_{1,\mathbf{u}}$) for 3D linear VEM on: (a) truncated octahedral meshes, (b) CVT meshes, (c) distorted hexahedral meshes, and (d) extruded octagonal meshes. The estimated errors include ones obtained from the proposed scheme and an SPR-type scheme introduced in Appendix C.

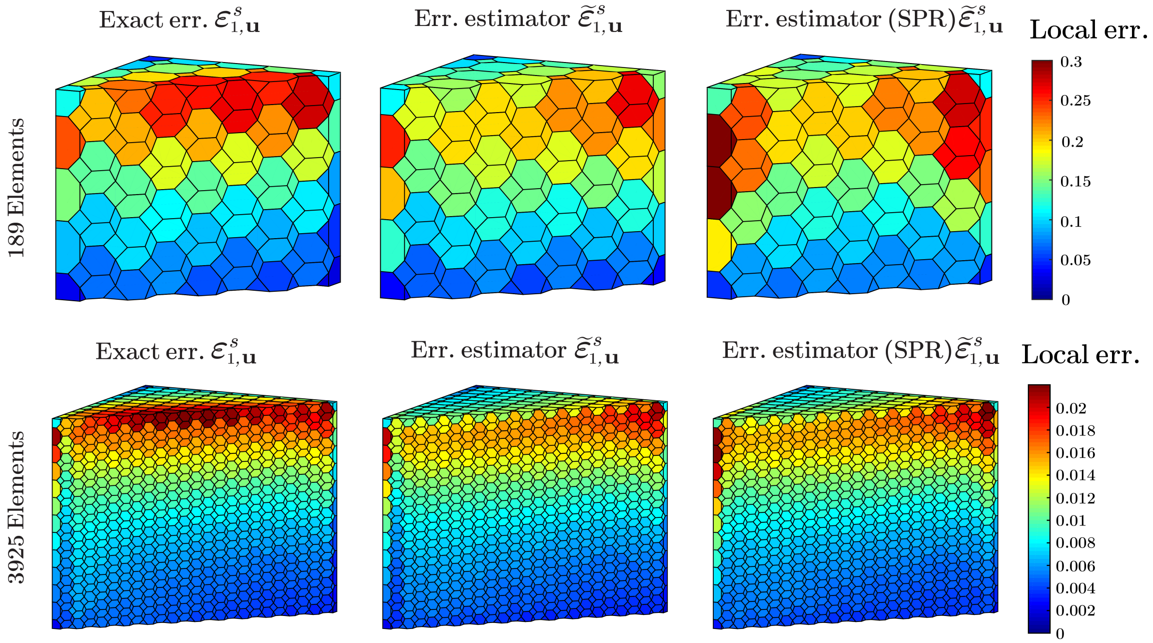


Figure 4.24: Fringes plots of the element-level exact errors and estimated ones for 3D linear VEM on truncated octahedral meshes.

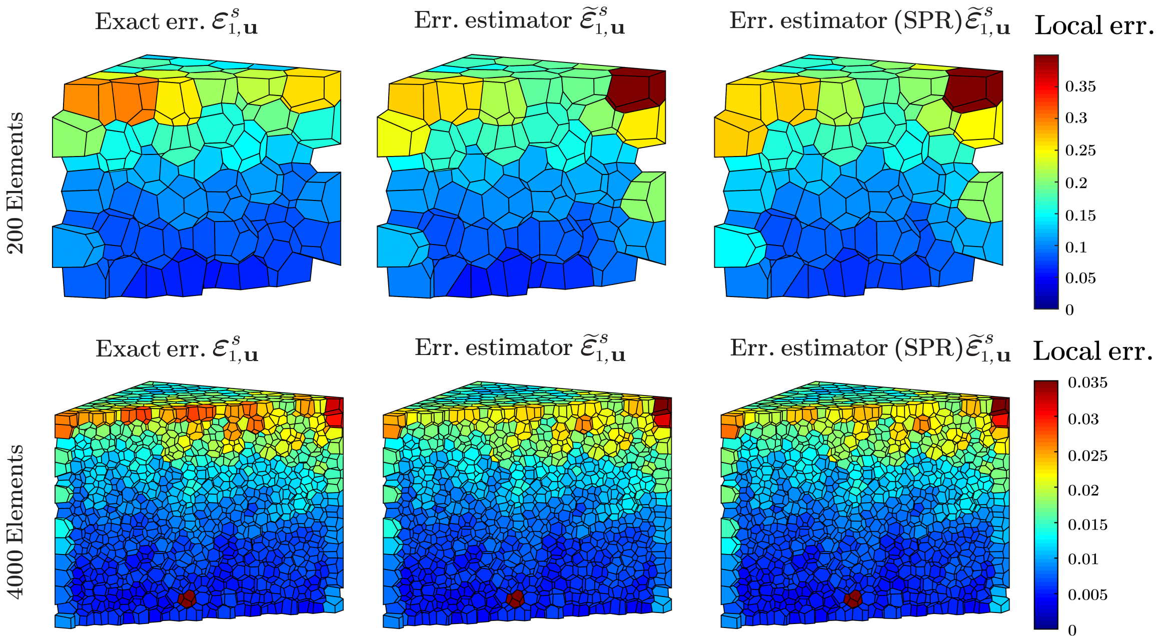


Figure 4.25: Fringes plots of the element-level exact errors and estimated ones for 3D linear VEM on CVT meshes.

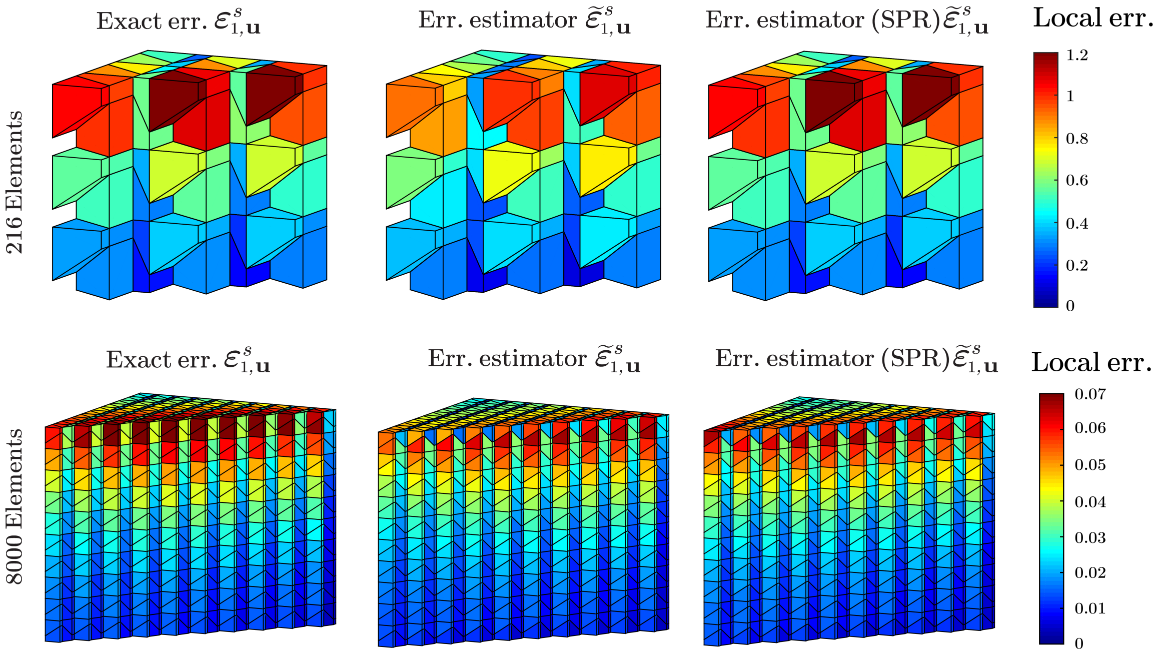


Figure 4.26: Fringes plots of the element-level exact errors and estimated ones for 3D linear VEM on distorted hexahedral meshes.

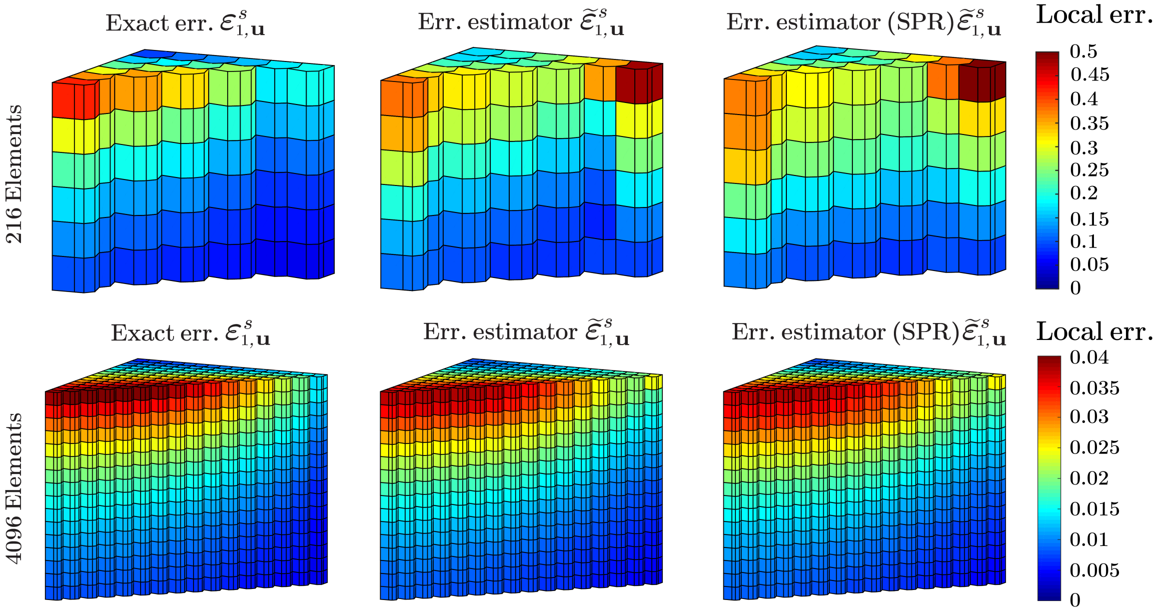
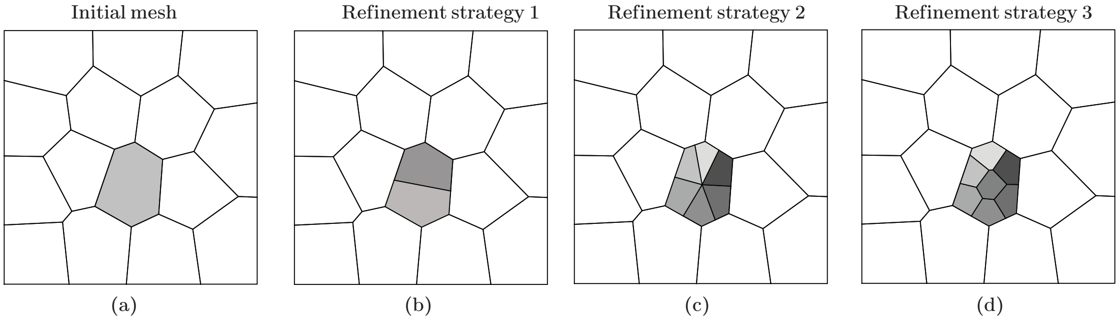


Figure 4.27: Fringes plots of the element-level exact errors and estimated ones for 3D linear VEM on extruded octagonal meshes.

Strategy 3: In this strategy, each marked element with n vertices is subdivided into $n+1$ CVT elements using Lloyd's algorithm. The initial seeds of Lloyd's algorithm are placed at the centroid this element as well as the midpoints of the lines connecting the centroid and the vertices [66], [170]. A schematic illustration of this strategy is shown in Fig. 4.28 (d).

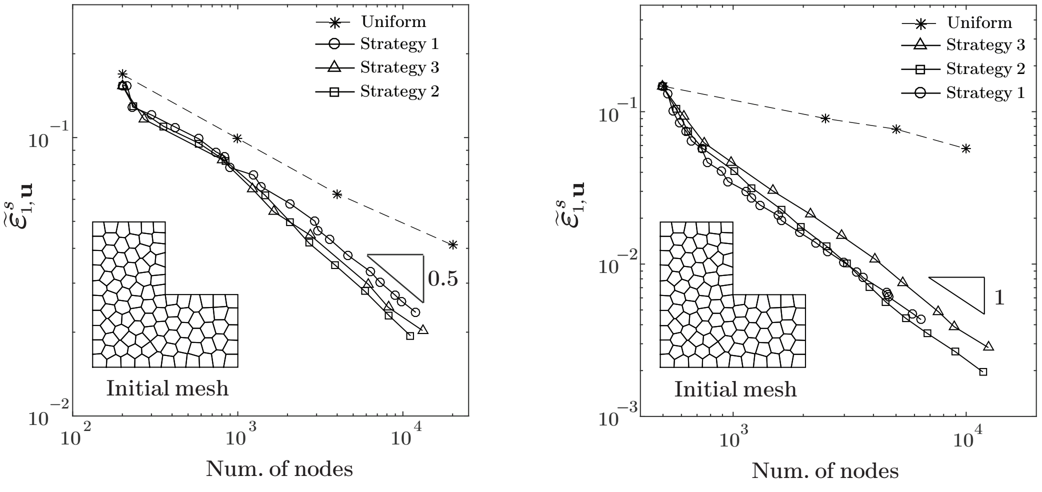
We note that the above adaptive refinement strategies will naturally introduce hanging nodes to the elements adjacent to the ones that are refined. In our implementation of both Strategies 2 and 3, in order to save degrees of freedom, those hanging nodes are not regarded as regular vertices. For instance, in Strategy 2, let us assume that a quadrilateral element with one hanging nodes needs to be refined. If the hanging node is considered as a regular vertex of that element, then the element will be subdivided into five elements. Instead, our implementation does not consider this hanging node as a regular vertex and, thus, will split this element into only four elements.

Figures 4.29 (a) and (b) show the convergence of the estimated global errors $\widetilde{\varepsilon}_{\mathbf{u},s}$ as functions of the total number of nodes with the above-mentioned adaptive mesh refinement strategies using linear and quadratic virtual elements, respectively. It is observed from both figures that, unlike the cases of uniform refinement shown in Figs. 4.18 (a)–(c), all the three adaptive mesh refine strategies driven by the proposed error estimator are able to restore the optimal convergence rates of the H^1-type displacement error (i.e. 0.5 for linear and 1 for quadratic VEM) with respect to the total numbers of nodes, demonstrating the effectiveness as well as the flexibility of the proposed gradient recovery scheme and error estimator. Moreover, three representative meshes are selected for each strategy at three similar global error levels and are plotted in Figs. 4.30 and 4.31 for linear and quadratic VEMs, respectively. We conclude from both figures that all the three adaptive refinement strategies effectively capture the problematic regions of the domain: the reentrant corner and the two ends of the clamped top edge. We also observe that, compared to the refinements with linear virtual elements, the ones with quadratic virtual elements are more localized to those problematic regions because of the higher potential local accuracy in the approximation. Finally, we comment that, for both linear and quadratic VEMs, Strategy 2 seems to produce the most accurate result under a similar number of nodes (although this may be problem-dependent), and Strategy 3 is more aggressive than the other ones in the sense that it achieves similar levels of global errors using fewer numbers of refinements.



(a) (b) (c) (d)

Figure 4.28: (a) An illustration of an initial CVT mesh with the gray element being marked for refinement. (b) A schematic illustration of the adaptive refinement strategy 1, where the marked element is bisected into two elements. (b) A schematic illustration of the adaptive refinement strategy 2, where the marked element is subdivided into 6 quadrilateral elements. (d) A schematic illustration of the adaptive refinement strategy 3, where the marked element tessellated into 7 CVT elements.



(a) (b)

Figure 4.29: Convergence of the estimated errors as functions of the total numbers of nodes with the three adaptive mesh refinement strategies for (a) linear VEM, and (b) quadratic VEM.

4.8 Concluding remarks

This chapter introduces a general recovery-based *a posteriori* error estimation framework for lower and higher order VEM on polygonal and polyhedral meshes, and demonstrates the theory in the context of linear elasticity. Within the error estimation framework, we first recover a more accurate displacement gradient through least square fitting the displacement DOFs over each patch in the mesh. Based on the recovered gradient, an error estimator is obtained by evaluating the difference between the recovered and original displacement gradients on the skeleton of the mesh. This skeletal error is shown in the numerical studies to capture the standard L^2 norm of the displacement gradient error extremely well. With thorough numerical studies, the introduced error estimation is shown to be accurate for both linear and quadratic virtual elements on various polygonal/polyhedral meshes and with various types of displacement solutions (e.g. smooth ones, ones with sharp gradients, and ones containing singularities). For linear VEM, the accuracy and effectiveness of the proposed error estimation framework are further demonstrated by comparing it with an SPR-type error estimation as outlined in Appendix C. For higher-order VEM, the numerical studies also suggest that we can neglect the internal displacement DOFs (which are in the form of function moments over the elements) in the gradient recovery procedure without sacrificing accuracy. We note that, although being proposed for VEM and linear elasticity problems, the error estimation framework can also be readily employed for polygonal and polyhedral FEM and nonlinear elasticity problems.

In terms of future work, we remark that the proposed recovery-based *a posteriori* error estimation offers an effective tool for adaptive VEM analysis. Future extensions of this research include developing novel and efficient mesh adaption strategies for VEM on polygonal/polyhedral discretization. Another promising area of research consists of extending the ideas presented here to nonlinear fracture mechanics problems [171].

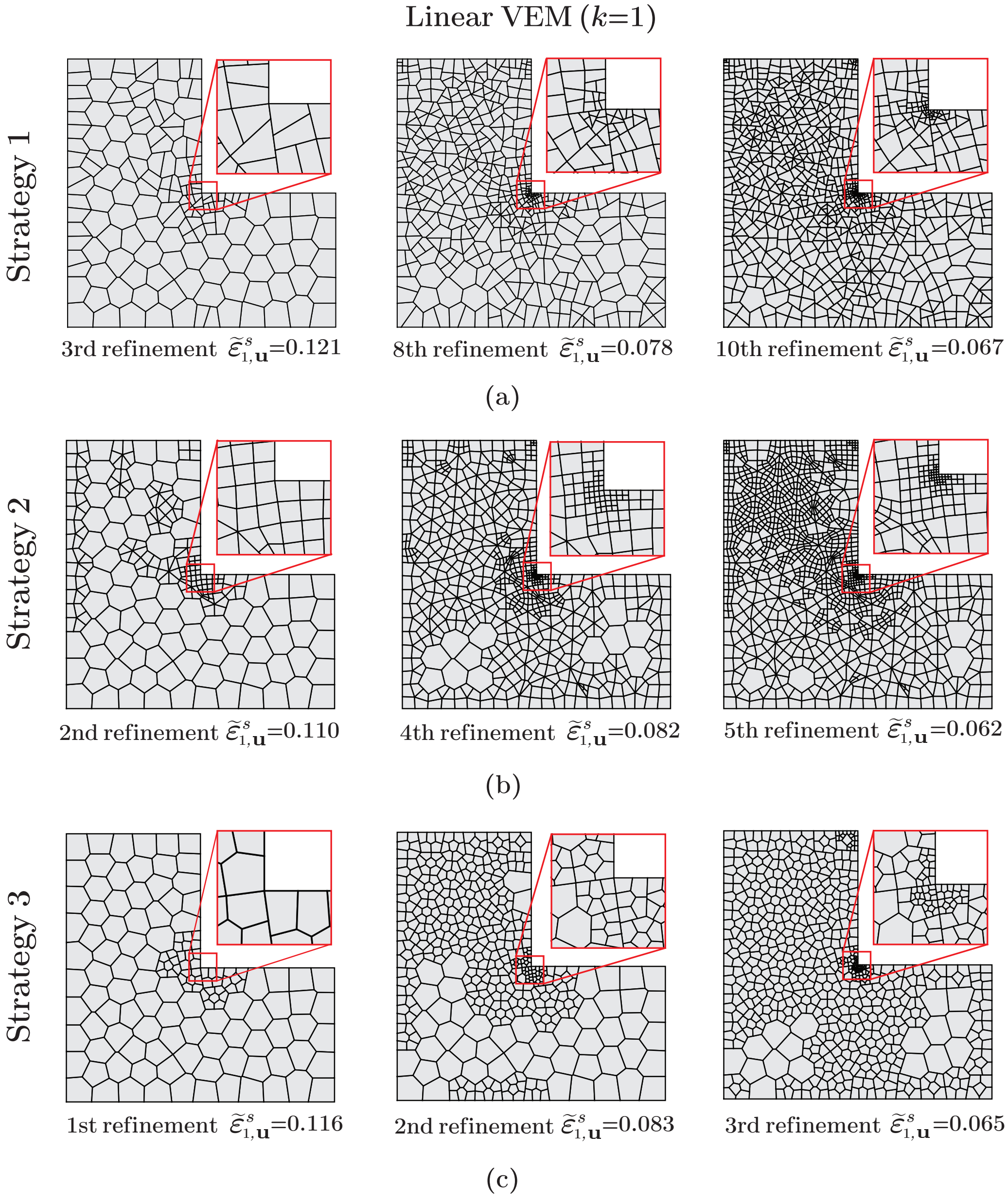
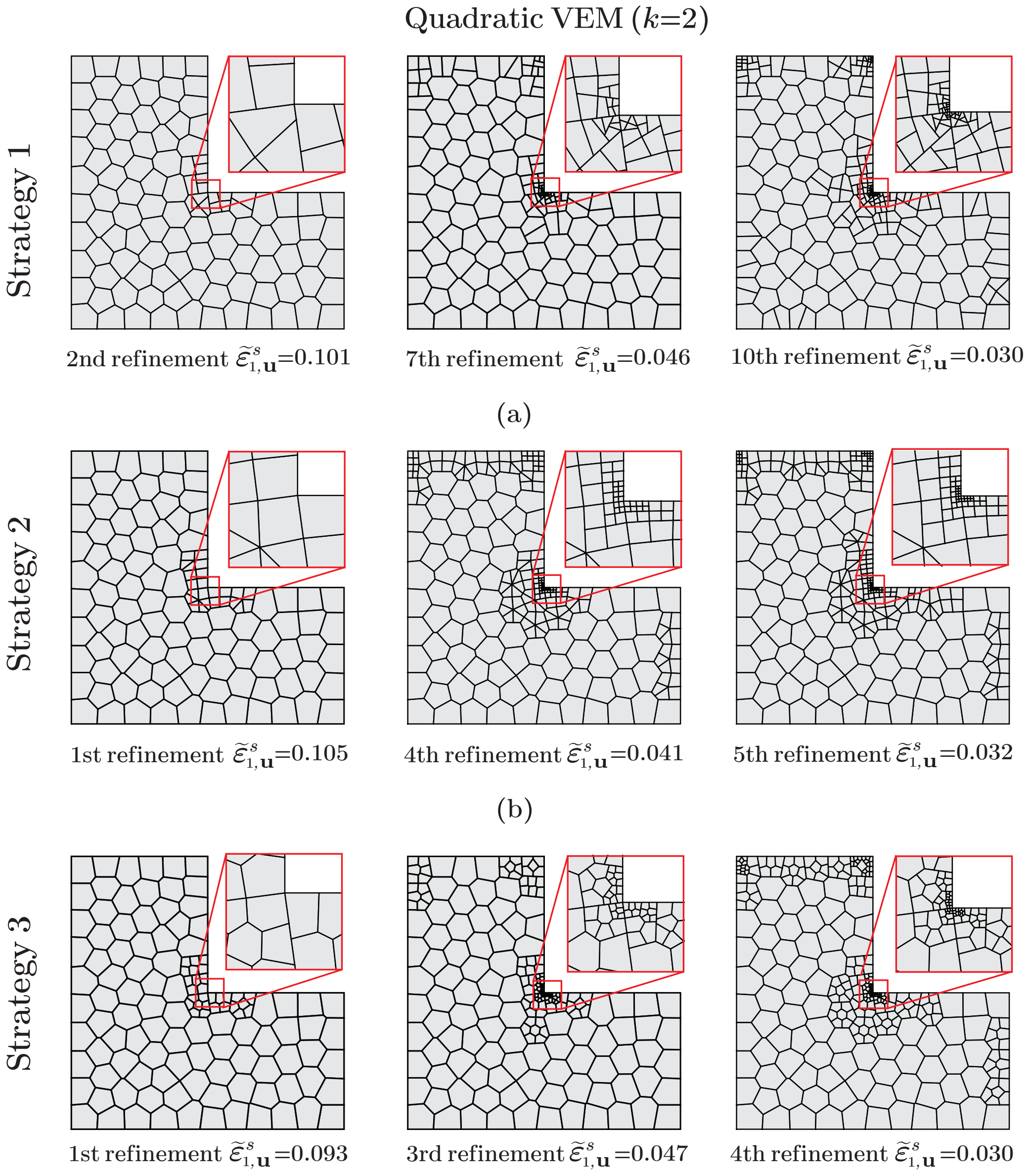

Linear VEM (k=1)
Strategy 1
3rd refinement $\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}$=0.121
8th refinement $\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}$=0.078
10th refinement $\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}$=0.067
(a)
Strategy 2
2nd refinement $\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}$=0.110
4th refinement $\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}$=0.082
5th refinement $\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}$=0.062
(b)
Strategy 3
1st refinement $\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}$=0.116
2nd refinement $\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}$=0.083
3rd refinement $\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}$=0.065
(c)

Figure 4.30: When linear virtual elements are considered, some representative meshes using (a) adaptive refinement strategy 1, (b) adaptive refinement strategy 2, and (c) adaptive refinement strategy 3. The representative meshes are selected so that the three meshes in each column of this figure have a similar level of estimated global error.


Quadratic VEM (k=2)
Strategy 1
2nd refinement $\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}$=0.101
7th refinement $\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}$=0.046
10th refinement $\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}$=0.030
Strategy 2
1st refinement $\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}$=0.105
4th refinement $\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}$=0.041
5th refinement $\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}$=0.032
Strategy 3
1st refinement $\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}$=0.093
3rd refinement $\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}$=0.047
4th refinement $\widetilde{\mathcal{E}}^s_{1,\mathbf{u}}$=0.030

(a)

(b)

(c)

Figure 4.31: When quadratic virtual elements are considered, some representative meshes using (a) adaptive refinement strategy 1, (b) adaptive refinement strategy 2, and (c) adaptive refinement strategy 3. The representative meshes are selected so that the three meshes in each column of this figure have a similar level of estimated global error.

CHAPTER 5
VIRTUAL ELEMENT METHOD (VEM)-BASED TOPOLOGY OPTIMIZATION: AN INTEGRATED FRAMEWORK

We present a Virtual Element Method (VEM)-based topology optimization framework using polyhedral elements, which allows for easy handling of non-Cartesian design domains in three dimensions. We take full advantage of the VEM properties by creating a unified approach in which the VEM is employed in both the structural and optimization phases of the framework. In the structural problem, the VEM is adopted to solve the three-dimensional elasticity equation. Compared to the finite element method (FEM), the VEM does not require numerical integration and is less sensitive to degenerated elements (e.g. ones with skinny faces or small edges). In the optimization problem, we introduce a continuous approximation of material densities using VEM basis functions. As compared to the standard element-wise constant one, the continuous approximation enriches geometrical representations of structural topologies. Through two numerical examples with exact solutions, we verify the convergence and accuracy of both the VEM approximations of the displacement and material density fields. We also present several design examples involving non-Cartesian domains, demonstrating the main features of the proposed VEM-based topology optimization framework.

5.1 Introduction

Topology optimization is a powerful computational tool to design optimal structures under given loads and boundary conditions. Since the seminal work of Bendsøe and Kikuchi [61], the field of topology optimization has experienced tremendous growth and had a major impact in several areas of engineering and technology. For an overview of this field, we refer the interested readers to textbooks [172]–[174] and review papers [175], [176]. Among various topology optimization approaches, the density-based approach is commonly adopted, in which the geometry is parametrized by a material density function and the displacement field is approximated by finite elements. Because of its simplicity and efficiency, the choice of piece-wise constant density parametrization, where each finite element is assigned with a constant density, is typically employed in conjunction with lower-order Lagrangian-type displacement approximation. However, this choice of density parametrization suffers from numerical instabilities such as checkerboard patterns and one node connections [62], [63]. In a general setting, Jog and Haber [177] formulated the topology optimization as a mixed variational problem and demonstrated how different choices of displacement and density interpolations affect the appearance of numerical instabilities. Along this direction, different choices of displacement and density approximations were investigated. For instance, Matsui and Terada [178] and Rahmatalla and Swan [179] employed continuous approximations of the density fields interpolated by Lagrangian finite element basis functions. This approach, sometimes known as Continuous Approximation of Material Distribution (CAMD), is effective in preventing the checkerboard patterns but may lead to other forms of numerical instabilities such as “islanding” and “layering” with linear finite elements [179]. Alternatively, Guest et al. [180] introduced nodal design variables and employed a projection map with an embedded length scale to construct an element-wise constant density field. Other related efforts also include the use of higher-order displacement approximations [62], [178], [181], non-conforming elements [182], [183], and approximating density and displacement fields on different discretizations [184], [185]. The multi-resolution topology optimization approaches (see, e.g. [170], [181], [186], [187]) belong to the last family. In addition to various types of numerical instabilities, topology optimization is typically performed on structured meshes (e.g. uniform grids), which may lead to mesh-dependent solutions [64] and limited ability to discretize complex design domains.

In recent studies, polygonal finite elements have been shown to be effective in suppressing checkerboard patterns and reducing mesh dependency in the solutions of topology optimization [17]–[19], [64], [65]. Moreover, as compared to standard FEM uniform grids, polygonal elements are more versatile in discretizing complex domains. To this effect, a robust mesh generator, named `PolyMesher`, able to discretize arbitrary 2D domains with polygonal elements, has been developed [110]. Other efforts in developing polygonal and polyhedral meshers also include [188] and [189]. In addition, the geometrical flexibility of polygonal finite elements also makes them attractive for mesh adaptations in topology optimization (see, e.g. [66], [67]). However, most of the above-mentioned investigations are in 2D and efficient extensions to 3D problems possess several challenges.

The first challenge comes from the difficulties of polyhedral FEM [190]. For polyhedral finite elements, one major difficulty is associated with obtaining the shape functions and their gradients. Although several shape functions exist in the literature with closed-form expressions, most of them are limited to certain classes of element geometry. For example, the Wachspress shape functions only work with strictly convex and simple polyhedra (meaning the collection of faces that include each vertex consists of exactly three faces) [30], and the Mean Value coordinates are mainly applicable to polyhedra with simplicial faces [31]. Those limitations in the element geometry could potentially affect the accuracy of the polyhedral FEM when dealing with degenerated elements, such as the ones with skinny faces or small edges. Other types of shape functions, such as harmonic [29], [33] and max-entropy [34], [35], allow for more general polyhedra (e.g. concave ones). However, their values and gradients at integration points can only be computed numerically, which could be undesirable from a computational perspective for large-scale problems. Another difficulty of the polyhedral FEM is associated with numerical integration [29], [130], [131]. Because there is no iso-parametric mapping for polyhedral finite elements, numerical integration needs to be performed in the physical domain. Due to the non-polynomial nature of their shape functions, efficient yet consistent numerical integration rules are difficult to construct on general polyhedral finite elements. Ensuring convergence of the numerical solution typically requires a prohibitively large number of integration points in each element (see [22],

[68], [107] for some recent work that attempts to overcome this difficulty).

In this work, we also identify another major challenge to use polyhedral meshes in 3D topology optimization, which is related to the computational efficiency. As mentioned in the preceding paragraphs, typically topology optimization adopts an element-wise constant density approximation and a lower-order displacement approximation with the degrees of freedom (DOFs) located at the vertices of the mesh. As a polyhedral mesh (e.g. Centroidal Voronoi Tessellation (CVT)) typically contains significantly more vertices than elements, this typical choice of density and displacement approximations becomes considerably less computationally economical on polyhedral meshes. In particular, when compared with uniform grids of a similar number of elements (thus similar numbers of densities and design variables), we need to solve a much larger structural system in each optimization iteration if a polyhedral mesh is considered.

The Virtual Element Method (VEM) [37] is a recently proposed approach that has the potential to overcome the difficulties of the polyhedral FEM. *The VEM can be viewed as a generalization of the FEM and is able to effectively handle arbitrary polygonal and polyhedral meshes.* One main feature of the VEM is that its shape functions are defined implicitly through a suitable set of Partial Differential Equations (PDEs). Instead of solving the PDEs for the values of shape functions and their gradients at the integration points, the VEM constructs a set of projection operators which project the shape functions and their gradients onto polynomial functions of suitable orders [37], [38]. By construction, these projections can be exactly computed using only the DOFs of the unknown fields. Another major feature of the VEM is that it decomposes the weak forms into consistency and stability terms, both of which can be directly formed using the projections of the shape functions and their gradients as well as the DOFs of the unknown fields [37]. As a result, for any element geometry, the VEM only needs to integrate polynomials (and not non-polynomials as in the polyhedral FEM). *For lower-order VEM, which is the focus of this chapter, no numerical integration is needed.* Because of such attractive features, the VEM has gained significant visibility in the computational mechanics community. For instance, the VEM has been developed for linear elasticity [26], [42], [43], small deformation nonlinear elastic and inelastic [44]–[47], finite deformation elasticity and elasto-plasticity [39], [53], [54], [191], plate bending [48]–[51], and damage and fracture problems [192], [193], to name a few. We also mention that the VEM has been adopted to solve the state equations in topology optimization on unstructured polygonal [64] and polyhedral [65] meshes as well as on Escher-based tessellations [40].

In this work, we propose a new VEM-based topology optimization framework on general polyhedral discretizations. To address the above-mentioned challenges of efficiently formulating topology optimization on polyhedral meshes, we adopt the VEM in both structural and optimization problems. Similarly to [65], we use VEM to solve the elasticity equation in the structural problem. The capability of VEM in handling arbitrary element geometry allows us to solve the structural problem more efficiently (i.e. no numerical integration is needed) and robustly (i.e. with respect to degenerated elements) on polyhedral meshes. Exploiting the flexibility of VEM in defining local spaces, this work also introduces an enriched continuous approximation of material densities using nodal VEM basis functions. As compared to the standard element-wise constant density approximation, the continuous approximation contains a greater number of DOFs for any given polyhedral mesh and can thus improve the quality of structural topology parameterizations. Moreover, this work explores various approaches of discretizing complex domains in 3D, including regular polyhedra-dominated and unstructured polyhedral meshes, and investigate their influences on the quality and the numerical stability of solutions in the topology optimization. Several design examples are presented on non-Cartesian domains to demonstrate the main features of the VEM-based topology optimization framework. To complement the library of educational codes (e.g. [3], [17], [194]–[198]) in the topology optimization and VEM literature, the source code for a MATLAB implementation, named `PolyTop3D`, is developed accompanying this chapter.

The remainder of this chapter is organized as follows. Section 2 provides an overview of the theory and implementation of VEM for 3D linear elasticity problems. Section 3 introduces the proposed VEM-based topology optimization together with a simple numerical example comparing the performance of the proposed continuous density parametrization with the standard element-wise constant one. In Section 4, we present a set of design examples featuring non-Cartesian domains to highlight the main features of the proposed VEM-based topology optimization framework. Section 5 contains several concluding remarks and future research directions. In Appendix D, the implementation of the `PolyTop3D` is presented and the computational efficiency of the code is demonstrated.

5.2 VEM Basics: theory and implementation

We consider an elastic solid $\Omega \in \mathbb{R}^3$ with its boundary denoted by $\partial\Omega$. The solid is subjected to a prescribed displacement $\mathbf{u}^0$ on one portion of the boundary $\Gamma^{\mathbf{X}}$ and a traction $\mathbf{t}$ on the other portion $\Gamma^{\mathbf{t}}$, such that $\Gamma^{\mathbf{X}} \cup \Gamma^{\mathbf{t}} = \partial\Omega$ and $\Gamma^{\mathbf{X}} \cap \Gamma^{\mathbf{t}} = \emptyset$. In the interior of Ω, the solid is subjected to a body force $\mathbf{f}$. For a given displacement field $\mathbf{u}$, the linearized strain tensor $\boldsymbol{\varepsilon}(\mathbf{u})$ is obtained as $\boldsymbol{\varepsilon}(\mathbf{u}) = 1/2[\nabla\mathbf{u} + (\nabla\mathbf{u})^T]$, where ∇ stands for the gradient operator. The stress tensor is given by the constitutive relationship

$$\boldsymbol{\sigma} = \mathbf{C}\boldsymbol{\varepsilon}, \tag{5.1}$$

where $\mathbf{C}$ is the elasticity tensor.

The weak form of the elasticity problem consists of finding the displacement $\mathbf{u}$ among the space $\mathscr{K}$ of kinematically admissible displacements,

$$a(\mathbf{u}, \delta\mathbf{u}) = \ell(\delta\mathbf{u}), \quad \forall \delta\mathbf{u} \in \mathscr{K}^0, \tag{5.2}$$

where $\delta\mathbf{u}$ is the variation of u, $\mathscr{K}^0$ stands for the space of kinematically admissible displacements that vanish on $\Gamma^{\mathbf{X}}$, and

$$a(\mathbf{u},\mathbf{v}) = \int_{\Omega} [\mathbb{C}\boldsymbol{\varepsilon}(\mathbf{u})] : \boldsymbol{\varepsilon}(\mathbf{v}) \mathrm{d}\mathbf{X} \qquad \ell(\mathbf{v}) = \int_{\Omega} \mathbf{f} \cdot \mathbf{v} \mathrm{d}\mathbf{X} + \int_{\Gamma^{\mathbf{t}}} \mathbf{t} \cdot \mathbf{v} \mathrm{d}S. \tag{5.3}$$

5.2.1 Virtual spaces on polygonal and polyhedral elements

For a general polyhedral element P consisting of planar faces, this subsection describes the construction of the local virtual space $\mathscr{V}(P)$ following the technique introduced in [140], [199]. In this technique, the construction of the virtual space for P depends on the virtual spaces on the faces of P. Therefore, before introducing the construction of the virtual space on P, we will first describe the construction of virtual spaces on faces of P. The reason why we choose this construction is that we find it yields functions which are positive in the interior and on the faces of P. We also note that, in the literature, there exist different approaches to construct local virtual spaces on general polyhedral elements. The interested readers are referred to [26], [39], [140], [162] for further information.

Virtual space on polygons

Here we introduce the definition of the virtual space $\mathscr{V}(F)$ on a generic face F of P, which is assumed to be a planar polygon. The basic idea is to first introduce a preliminary space denoted by $\widetilde{\mathscr{V}}(F)$ as

$$\widetilde{\mathscr{V}}(F) = \left\{ v \in H^1(F) : v|_e \in \mathscr{P}_1(e) \; \forall e \in \partial F, \, \Delta v \in \mathscr{P}_0(F) \right\}, \tag{5.4}$$

where e denotes a generic edge of face F and $\mathscr{P}_k(\cdot)$ is the polynomial space of order k. For the preliminary space $\widetilde{\mathscr{V}}(F)$, a set of DOFs consists of [37], [140]

- the values of v at vertices of F, (5.5)
- the mean value of v over F, i.e. $\dfrac{1}{|F|}\displaystyle\int_F v \mathrm{d}\mathbf{X}$, (5.6)

where $|F|$ stands for the area of face F.

By identity

$$2\int_F v\mathrm{d}\mathbf{X} = \int_F v\mathrm{div}\,\mathbf{X}_F\mathrm{d}\mathbf{X} = -\int_F \nabla v\cdot\mathbf{X}_F\mathrm{d}\mathbf{X} + \int_{\partial F} v\,\mathbf{X}_F\cdot\mathbf{n}\mathrm{d}s, \tag{5.7}$$

we can equivalently replace the DOF (5.6) by the following integral [200]

$$\int_F \nabla v\cdot\mathbf{X}^F\mathrm{d}\mathbf{X}, \tag{5.8}$$

where $\mathbf{X}^F \doteq \mathbf{X}-\mathbf{X}_c^F$ with $\mathbf{X}_c^F$ being the centroid of face F. In fact, once we know the above integral, we can compute the mean value of v over F using the DOFs (5.5) and the identity (5.7).

Having defined the preliminary virtual space $\widetilde{\mathscr{V}}(F)$, we can define the formal virtual space $\mathscr{V}(F)\subset\widetilde{\mathscr{V}}(F)$ on face F such that

$$\mathscr{V}(F) = \left\{ v\in H^1(F) : v|_e\in\mathscr{P}_1(e)\;\forall e\in F,\; \Delta v\in\mathscr{P}_0(F),\text{ and } \int_F \nabla v\cdot\mathbf{X}^F\mathrm{d}\mathbf{X}=0 \right\}. \tag{5.9}$$

By definition, we can show that $\mathscr{P}_1\subseteq\mathscr{V}(F)$ and (5.5) constitutes a complete set of DOFs of $\mathscr{V}(F)$. Using this set of DOFs, we can exactly compute moment of v on F according to the identity (5.7) as

$$\int_F v\mathrm{d}\mathbf{X} = \frac{1}{2}\int_{\partial F} v\mathbf{X}^F\cdot\mathbf{n}\mathrm{d}S = \frac{1}{2}\sum_{e\in\partial F}\int_e v\mathbf{X}^F\cdot\mathbf{n}^e\mathrm{d}S, \tag{5.10}$$

where $\mathbf{n}^e$ denotes the outward norm vector of edge e. Noticing that $\mathbf{X}^F\cdot\mathbf{n}^e = (\mathbf{X}-\mathbf{X}_c^F)\cdot\mathbf{n}^e$ takes constant value for any points on edge e (which is assumed to be straight), we can simply evaluate it at any point $\mathbf{a}^e$ on e, i.e. $\mathbf{X}^F\cdot\mathbf{n}^e = (\mathbf{a}^e-\mathbf{X}_c^F)\cdot\mathbf{n}^e, \forall\mathbf{a}^e\in e$.

Virtual space on polyhedrons

Once we know the virtual space on each face F, we are ready to define the virtual element space $\mathscr{V}(P)$ on P. Following the same concept, we define the final form of the virtual space $\mathscr{V}(P)$ as

$$\mathscr{V}(P) = \left\{ v \in H^1(P) : v|_F \in \mathscr{V}(F)\ \forall F \in \partial P,\ \Delta v \in \mathscr{P}_0(P),\ \int_P \nabla v \cdot \mathbf{X}^P \mathrm{d}\mathbf{X} = 0 \right\}, \quad (5.11)$$

where $\mathbf{X}^P \doteq \mathbf{X} - \mathbf{X}_c^P$ with $\mathbf{X}_c^P$ being the centroid of P. Similarly to $\mathscr{V}(F)$, we can define the set of DOFs of $\mathscr{V}(P)$ as the values of its functions on the vertices of P. Because $\mathscr{V}(P)$ includes the polynomial space $\mathscr{P}_1(P)$, i.e. $\mathscr{P}_1(P) \subseteq \mathscr{V}(P)$, we can define a projection operator $\Pi_P^\nabla : \mathscr{V}(P) \to \mathscr{P}_1(P)$ such that for any $v \in \mathscr{V}(P)$,

$$\begin{aligned} &\int_P \nabla(\Pi_P^\nabla v) \cdot \nabla p_1 \mathrm{d}\mathbf{X} = \int_P \nabla v : \nabla p_1 \mathrm{d}\mathbf{X} \qquad \forall p_1 \in \mathscr{P}_1(P) \\ &\sum_{\mathbf{X}_v \in P} v(\mathbf{X}_v) = \sum_{\mathbf{X}_v \in P} \Pi_P^\nabla v(\mathbf{X}_v). \end{aligned} \quad (5.12)$$

Because both $\nabla \Pi_P^\nabla v$ and ∇p_1 are constant vectors, the first condition in (5.12) can be simplified as

$$\nabla(\Pi_P^\nabla v) = \frac{1}{|P|} \int_P \nabla v \mathrm{d}\mathbf{X} = \frac{1}{|P|} \sum_{F \in \partial P} \int_F v \mathbf{n}^F \mathrm{d}F, \quad (5.13)$$

where $|P|$ denotes the volume of element P and $\mathbf{n}^F$ is the (constant) outward unit normal vector of face F. This condition ensures that the gradient of the projection $\Pi_P^\nabla v$ equals the average gradient of v over P. Recalling from the definition of $\mathscr{V}(F)$ that, given the value of v at the vertices of F, we can compute the moment of v over F (see Eq. (5.10)), and consequently, we can explicitly compute the boundary integral on the right-hand side of (5.13) using the DOFs of v and geometric information of P. On the other hand, the second condition in (5.12) determines the constant component of the projection by ensuring that, when evaluated at the vertices of P, the average value of $\Pi_P^\nabla v$ is equal to the average value of v. Furthermore, making use of the following identity

$$3\int_P v\mathrm{d}\mathbf{X} = \int_P v\,\mathrm{div}\,\mathbf{X}^P \mathrm{d}\mathbf{X} = -\int_P \nabla v\cdot\mathbf{X}^P \mathrm{d}\mathbf{X} + \int_{\partial P} v\,\mathbf{X}^P\cdot\mathbf{n}\mathrm{d}S, \tag{5.14}$$

and the definition of $\mathscr{V}(P)$, we can express the moment of v over P as

$$\int_P v\mathrm{d}\mathbf{X} = \frac{1}{3}\sum_{F\in\partial P}\int_F v\mathbf{X}^P\cdot\mathbf{n}^F\mathrm{d}F. \tag{5.15}$$

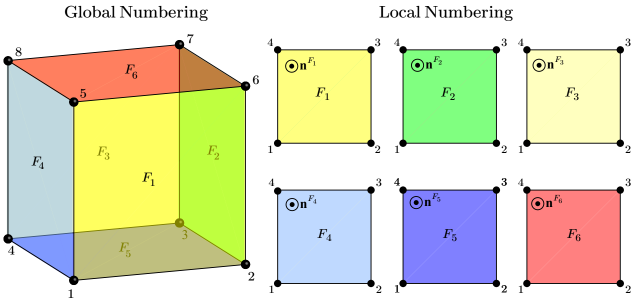
Realizing that $\mathbf{X}^P\cdot\mathbf{n}^F = (\mathbf{X}-\mathbf{X}_c^P)\cdot\mathbf{n}^F$ is constant for any points $\mathbf{X}$ on planar F, we can simply evaluate this quantity at any point $\mathbf{a}^F$ on face F, i.e. $\mathbf{X}^P = (\mathbf{a}^F-\mathbf{X}_c^P)\cdot\mathbf{n}^F$, $\forall\mathbf{a}^F\in F$. Thus, we can compute the moment of v over P exactly as

$$\int_P v\mathrm{d}\mathbf{X} = \frac{1}{3}\sum_{F\in\partial P}(\mathbf{X}^P\cdot\mathbf{n}^F)\int_F v\mathrm{d}F, \tag{5.16}$$

using only the DOFs of v (recalling Eq. (5.10)).

Some implementation details

Consider a polyhedron P consisting of m vertices numbered as $\mathbf{X}_1,...,\mathbf{X}_m$ and denote $\mathscr{F}_i$ as the set of faces that are connected to the ith vertex. Suppose that a face $F\in\mathscr{F}_i\in\mathbb{R}^3$ has m^F vertices $\mathbf{X}_j^F$, $j=1,...,m^F$, we locally renumber those vertices in a counterclockwise fashion with respect to the outward normal $\mathbf{n}^F$ which points out of the element. We also utilize a map $\mathscr{G}_F$ to denote the relation between the global numbering and local numbering of the vertices on face F. If the ith vertex of P (under global numbering) is numbered as the jth vertex of F (under local numbering), we write $j=\mathscr{G}_F(i)$. Figure 5.1 shows an illustration of the above notation for a hexahedral element. Following this notation, this subsection provides the implementational details to construct the virtual space $\mathscr{V}(P)$. In particular, we focus on the procedures of calculating the projection $\Pi_P^\nabla\varphi_i$ and the moment $\int_P\varphi_i$, where φ_i is the basis function of $\mathscr{V}(P)$ associated with the ith vertex.



		Global Numbering							
	i	1	2	3	4	5	6	7	8
Local Numbering	$\mathscr{G}_{F_1}(i)$	1	2			4	3		
	$\mathscr{G}_{F_2}(i)$		1	2			4	3	
	$\mathscr{G}_{F_3}(i)$			1	2			4	3
	$\mathscr{G}_{F_4}(i)$	1			4	2			3
	$\mathscr{G}_{F_5}(i)$	1	4	3	2				
	$\mathscr{G}_{F_6}(i)$					1	2	3	4

Figure 5.1: Illustration of the global and local numbering conventions and the mapping, $\mathscr{G}_F(i)$, between them on a regular hexahedral elements.

Based on the definition of projection operator Π_P^∇ in (5.12), we can express $\Pi_P^\nabla \varphi_i$ as

$$\Pi_P^\nabla \varphi_i = (\nabla \Pi_P^\nabla \varphi_i) \cdot (\mathbf{X} - \widehat{\mathbf{X}}^P) + \frac{1}{m}, \tag{5.17}$$

where $\widehat{\mathbf{X}}^P \doteq 1/m \sum_{j=1}^m \mathbf{X}_i$ is the algorithmic mean of the position vectors of the vertices of P. To compute this projection, we first need to know the moment of φ_i on F, $\forall F \in \mathscr{F}_i$. Using relation (5.10) and realizing that φ_i varies linear on ∂F, we can compute the moment of φ_i on F as

$$\begin{aligned}
\int_F \varphi_i \mathrm{d}F &= \frac{1}{2} \sum_{e \in \partial F} (\mathbf{X}^F \cdot \mathbf{n})^e \int_e \varphi_i \mathrm{d}S \\
&= \frac{1}{2} \sum_{e \in \partial F} (\mathbf{X}_i - \mathbf{X}_c^f) \cdot (\mathbf{n}^e \int_e \varphi_i \mathrm{d}S) \quad \text{(evaluate } \mathbf{X}^F = \mathbf{X} - \mathbf{X}_c^F \text{ at vertex } \mathbf{X}_i) \\
&= \frac{1}{4} (\mathbf{X}_{\mathscr{G}_F(i)+1} - \mathbf{X}_{\mathscr{G}_F(i)-1}) \wedge \mathbf{n}^F \cdot (\mathbf{X}_i - \mathbf{X}_c^f),
\end{aligned} \tag{5.18}$$

where $\wedge$ stands for the cross product and the convention is used: $\mathscr{G}_F(\cdot) + 1 = 1$ whenever $\mathscr{G}_F(\cdot) = m^F$, and $\mathscr{G}_F(\cdot) - 1 = m^F$ whenever $\mathscr{G}_F(\cdot) = 1$. Substituting the above relation into (5.13), we can show that

$$\nabla \Pi_P^\nabla \varphi_i = \frac{1}{|P|} \sum_{F \in \mathscr{F}_i} \int_F \varphi_i \mathbf{n}^F \mathrm{d}F = \frac{1}{4|P|} \sum_{F \in \mathscr{F}_i} (\mathbf{X}_i - \mathbf{X}_c^F) \wedge (\mathbf{X}_{\mathscr{G}_F(i)+1} - \mathbf{X}_{\mathscr{G}_F(i)-1}) \tag{5.19}$$

and, by (5.17), we arrive at the following expression for $\Pi_F^\nabla \varphi_i$:

$$\Pi_P^\nabla \varphi_i = \frac{1}{4|P|} \sum_{F \in \mathscr{F}_i} (\mathbf{X}_i - \mathbf{X}_c^F) \wedge (\mathbf{X}_{\mathscr{G}_F(i)+1} - \mathbf{X}_{\mathscr{G}_F(i)-1}) \cdot (\mathbf{X} - \widehat{\mathbf{X}}^P) + \frac{1}{m}. \tag{5.20}$$

Moreover, since $\Pi_P^\nabla \varphi_i \in \mathscr{P}_1(P) \subseteq \mathscr{V}(P)$, we can alternatively express $\Pi_E^\nabla \varphi_i$ as a linear combination of the set of basis functions $\varphi_1, ..., \varphi_m$ as

$$\Pi_P^\nabla \varphi_i = \sum_{j=1}^{m} \mathbf{P}_{(ij)}^P \varphi_j, \tag{5.21}$$

where $\mathbf{P}_{(ij)}^P$ is the (i,j)th components of a matrix $\mathbf{P}^P \in \mathbb{R}^{m \times m}$ of the form

$$\mathbf{P}_{(ij)}^P = \frac{1}{4|P|} \sum_{F \in \mathscr{F}_i} (\mathbf{X}_i - \mathbf{X}_c^F) \wedge (\mathbf{X}_{\mathscr{G}_F(i)+1} - \mathbf{X}_{\mathscr{G}_F(i)-1}) \cdot (\mathbf{X}_j - \widehat{\mathbf{X}}^P) + \frac{1}{m}. \tag{5.22}$$

In terms of implementation, we form $\mathbf{P}^P$ using matrix multiplication as follows. We first define $\mathbf{G}^P \in \mathbb{R}^{m \times 3}$ as a matrix collecting the information of $\nabla \Pi_P^\nabla \varphi_i$, $i = 1, ..., m$:

$$\mathbf{G}^P = \begin{bmatrix} \frac{\partial \Pi_P^\nabla \varphi_1}{\partial X} & \frac{\partial \Pi_P^\nabla \varphi_1}{\partial Y} & \frac{\partial \Pi_P^\nabla \varphi_1}{\partial z} \\ \vdots & \vdots & \vdots \\ \frac{\partial \Pi_P^\nabla \varphi_m}{\partial X} & \frac{\partial \Pi_P^\nabla \varphi_m}{\partial Y} & \frac{\partial \Pi_P^\nabla \varphi_m}{\partial z} \end{bmatrix}, \tag{5.23}$$

and compute $\mathbf{P}^P$ as

$$\mathbf{P}^P = \begin{bmatrix} x_1 - \frac{1}{m}\sum_{i=1}^m x_i & y_1 - \frac{1}{m}\sum_{i=1}^m y_i & z_1 - \frac{1}{m}\sum_{i=1}^m z_i & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_m - \frac{1}{m}\sum_{i=1}^m x_i & y_m - \frac{1}{m}\sum_{i=1}^m y_i & z_m - \frac{1}{m}\sum_{i=1}^m z_i & 1 \end{bmatrix} \begin{bmatrix} \mathbf{G}^P & \frac{1}{m}\mathbf{1} \end{bmatrix}^T, \tag{5.24}$$

where $\mathbf{1} \in \mathbb{R}^{m \times 1}$ is a column vector with all components being 1.

For later use, we also provide the expression to compute the moment of φ_i over P, which, according to expressions (5.16) and (5.18), takes the form

$$\begin{aligned}\int_P \varphi_i \mathrm{d}\mathbf{X} &= \frac{1}{3}\sum_{F\in\mathscr{F}_i} (\mathbf{X}^P\cdot\mathbf{n}^F)\int_F \varphi_i \mathrm{d}F \\ &= \frac{1}{3}\sum_{F\in\mathscr{F}_i}\left[(\mathbf{X}_i-\mathbf{X}_c^P)\cdot\mathbf{n}^F\right]\int_F \varphi_i\mathrm{d}F \quad \text{(evaluate } \mathbf{X}^P=\mathbf{X}-\mathbf{X}_c^P \text{ at vertex } \mathbf{X}_i\text{)} \\ &= \frac{1}{12}\sum_{F\in\mathscr{F}_i}\left[(\mathbf{X}_i-\mathbf{X}_c^P)\cdot\mathbf{n}^F\right]\left[(\mathbf{X}_{\mathscr{G}_F(i)+1}-\mathbf{X}_{\mathscr{G}_F(i)-1})\wedge\mathbf{n}^F\cdot(\mathbf{X}_i-\mathbf{X}_c^F)\right] \ \text{(using (5.18))} \\ &= \frac{1}{12}\sum_{F\in\mathscr{F}_i}(\mathbf{X}_{\mathscr{G}_F(i)+1}-\mathbf{X}_{\mathscr{G}_F(i)-1})\wedge(\mathbf{X}_i-\mathbf{X}_c^P)\cdot(\mathbf{X}_i-\mathbf{X}_c^F).\end{aligned} \tag{5.25}$$

5.2.2 VEM approximations for 3D linear elasticity

Theoretical background

We consider a discretization, denoted by Ω_h, of the solid Ω into non-overlapping polyhedra consisting of planar faces, where h denotes the average element size. We denote $\Gamma_h^{\mathbf{t}}$ and $\Gamma_h^{\mathbf{X}}$ as the portions of the mesh boundary where the traction and displacement boundary conditions are applied, respectively. We define the discrete global displacement space $\mathscr{K}_h \subset \mathscr{K}$ as

$$\mathscr{K}_h \doteq \left\{\mathbf{v}_h \in \mathscr{K} : \mathbf{v}_h|_P \in [\mathscr{V}(P)]^3\right\}. \tag{5.26}$$

Over each element P, the local displacement $\mathbf{v}=[v_x,v_y,v_z]^T$ belongs to the vectorial space $[\mathscr{V}(P)]^3$, having three displacement DOFs for each vertex of P. In the following discussion, we define the projection operator for vector fields $\boldsymbol{\Pi}_P^\nabla : [\mathscr{V}(P)]^3 \to [\mathscr{P}_1(P)]^3$ as the action of Π_P^∇ on every component of the vector field, e.g., $\boldsymbol{\Pi}_P^\nabla\mathbf{v} = [\Pi_P^\nabla v_x, \Pi_P^\nabla v_y, \Pi_P^\nabla v_z]^T$.

Similarly to Chapter 4, the element-level discrete bilinear form is decomposed into the consistency term and the stability term, and they are responsible for the satisfaction of the two key conditions, namely consistency and stability, respectively, to ensure the convergence of the VEM approximation [37]. In the stability term, we define the bilinear term $S_{h,P}(\cdot,\cdot)$ and parameter α_P as:

$$S_{h,P}(\mathbf{u}_h,\mathbf{v}_h) = h_P \sum_{\mathbf{X}_v \in P} \mathbf{u}_h(\mathbf{X}_v)\cdot\mathbf{v}_h(\mathbf{X}_v) \text{ and } \alpha_P = \operatorname{trace}\mathbb{C} = C_{ijij} \text{ (in indical notation)}, \tag{5.27}$$

where $h_P \doteq |P|^{1/3}$ represents the size of element P and $\mathbf{X}_v$ stands for the vertices that belong to P. This gives the final form of the element-level discrete bilinear form as

$$a_h^P(\mathbf{u}_h,\mathbf{v}_h) = |P|\boldsymbol{\varepsilon}\Big(\boldsymbol{\Pi}_P^{\nabla}\mathbf{u}_h\Big) : \mathbb{C} : \boldsymbol{\varepsilon}\big(\boldsymbol{\Pi}_P^{\nabla}\mathbf{v}_h\big)\Big) + \alpha_P S_{h,P}\Big(\mathbf{u}_h - \boldsymbol{\Pi}_P^{\nabla}\mathbf{u}_h, \mathbf{v}_h - \boldsymbol{\Pi}_P^{\nabla}\mathbf{v}_h\Big). \tag{5.28}$$

On the other hand, we define the discrete loading term $\ell_h(\mathbf{v}_h)$ as [39]

$$\ell_h(\mathbf{v}_h) = \sum_{F\in\Gamma_h^{\mathbf{t}}} |F|\mathbf{t}(\mathbf{X}_c^F)(\boldsymbol{\Pi}_F^{\nabla}\mathbf{v}_h)(\mathbf{X}_c^F) + \sum_{P\in\Omega_h} \mathbf{F}(\mathbf{X}_c^P)(\boldsymbol{\Pi}_E^{\nabla}\mathbf{v}_h)(\mathbf{X}_c^P), \tag{5.29}$$

where $\boldsymbol{\Pi}_F^{\nabla}\mathbf{v}_h$ is the projection of $\mathbf{v}_h|_F$ onto $[\mathscr{P}_1(F)]^3$ defined in the same way as $\boldsymbol{\Pi}_P^{\nabla}\mathbf{v}_h$; and $\mathbf{t}(\mathbf{X}_c^F)$ and $\mathbf{F}(\mathbf{X}_c^P)$ are the values of traction and body forces evaluated at $\mathbf{X} = \mathbf{X}_c^F$ and $\mathbf{X} = \mathbf{X}_c^P$, respectively (we recall that $\mathbf{X}_c^F$ and $\mathbf{X}_c^P$ are the centroids of face F and element P, respectively). The above approximation essentially utilizes one-point rules on face F and element P, both of which are exact for integrating any linear function.

We are now ready to state the final form of the VEM approximation for 3D linear elasticity problems, which consists of finding $\mathbf{u}_h \in \mathscr{K}_h$ such that

$$a_h(\mathbf{u}_h,\delta\mathbf{u}_h) = \sum_P a_h^P(\mathbf{u}_h,\delta\mathbf{u}_h) = \ell_h(\delta\mathbf{u}_h) \quad \forall\delta\mathbf{u}_h \in \mathscr{K}_h^0, \tag{5.30}$$

where $\mathscr{K}_h^0$ is a subspace of $\mathscr{K}_h$ with functions that vanish on $\Gamma_h^{\mathbf{X}}$.

Some implementation details

For a given element P, we consider a set of basis functions, $\boldsymbol{\varphi}_1, ..., \boldsymbol{\varphi}_{3m}$, for the local displacement space $[\mathscr{V}(P)]^3$ of the form

$$\boldsymbol{\varphi}_{3i-2} = [\varphi_i, 0, 0]^T, \quad \boldsymbol{\varphi}_{3i-1} = [0, \varphi_i, 0]^T, \quad \boldsymbol{\varphi}_{3i} = [0, 0, \varphi_i]^T, \qquad i = 1, ..., m \tag{5.31}$$

where we recall $\varphi_1, ..., \varphi_m$ is the set of basis functions for $\mathscr{V}(P)$. With the set of basis functions, any displacement field $\mathbf{v} = [v_x, v_y, v_z]^T \in [\mathscr{V}(P)]^3$ can be interpolated using its DOFs (the values at the vertices) as

$$\mathbf{v} = \sum_{i=1}^{m} \left[\boldsymbol{\varphi}_{3i-2} v_x(\mathbf{X}_i) + \boldsymbol{\varphi}_{3i-1} v_y(\mathbf{X}_i) + \boldsymbol{\varphi}_{3i} v_z(\mathbf{X}_i)\right]. \tag{5.32}$$

Moreover, by definition, the projection $\boldsymbol{\Pi}_P^\nabla \boldsymbol{\varphi}_j$, $j = 1, ..., 3m$ is given by:

$$\boldsymbol{\Pi}_P^\nabla \boldsymbol{\varphi}_{3i-2} = [\Pi_P^\nabla \varphi_i, 0, 0]^T, \quad \boldsymbol{\Pi}_P^\nabla \boldsymbol{\varphi}_{3i-1} = [0, \Pi_P^\nabla \varphi_i, 0]^T, \quad \boldsymbol{\Pi}_P^\nabla \boldsymbol{\varphi}_{3i} = [0, 0, \Pi_P^\nabla \varphi_i]^T, i = 1, ..., m. \tag{5.33}$$

According to (5.28), we evaluate the (j,k)th component of the element stiffness matrix $\mathbf{k}^P \in \mathbb{R}^{3m \times 3m}$ as

$$\mathbf{k}^P_{(jk)} = a_h^P(\boldsymbol{\varphi}_j, \boldsymbol{\varphi}_k) = |P| \boldsymbol{\varepsilon}(\boldsymbol{\Pi}_P^\nabla \boldsymbol{\varphi}_j) : \mathbb{C} : \boldsymbol{\varepsilon}(\boldsymbol{\Pi}_P^\nabla \boldsymbol{\varphi}_k) + \alpha_P S^P(\boldsymbol{\varphi}_j - \boldsymbol{\Pi}_P^\nabla \boldsymbol{\varphi}_j, \boldsymbol{\varphi}_k - \boldsymbol{\Pi}_P^\nabla \boldsymbol{\varphi}_k). \tag{5.34}$$

More specifically, we can define matrices $\mathbf{B}^P \in \mathbb{R}^{6\times 3m}$ and $\mathbf{D} \in \mathbb{R}^{6\times 6}$ of the form

$$\mathbf{B}^P = \begin{bmatrix} \frac{\partial \Pi_P^\nabla \varphi_1}{\partial X} & 0 & 0 & \dots & \frac{\partial \Pi_P^\nabla \varphi_m}{\partial X} & 0 & 0 \\ 0 & \frac{\partial \Pi_P^\nabla \varphi_1}{\partial Y} & 0 & \dots & 0 & \frac{\partial \Pi_P^\nabla \varphi_m}{\partial Y} & 0 \\ 0 & 0 & \frac{\partial \Pi_P^\nabla \varphi_1}{\partial z} & \dots & 0 & 0\frac{\partial \Pi_P^\nabla \varphi_m}{\partial z} & \\ \frac{\partial \Pi_P^\nabla \varphi_1}{\partial Y} & \frac{\partial \Pi_P^\nabla \varphi_1}{\partial X} & 0 & \dots & \frac{\partial \Pi_P^\nabla \varphi_m}{\partial Y} & \frac{\partial \Pi_P^\nabla \varphi_m}{\partial X} & 0 \\ 0 & \frac{\partial \Pi_P^\nabla \varphi_1}{\partial z} & \frac{\partial \Pi_P^\nabla \varphi_1}{\partial Y} & \dots & 0 & \frac{\partial \Pi_P^\nabla \varphi_m}{\partial z} & \frac{\partial \Pi_P^\nabla \varphi_m}{\partial Y} \\ \frac{\partial \Pi_P^\nabla \varphi_1}{\partial z} & 0 & \frac{\partial \Pi_P^\nabla \varphi_1}{\partial X} & \dots & \frac{\partial \Pi_P^\nabla \varphi_m}{\partial z} & 0 & \frac{\partial \Pi_P^\nabla \varphi_m}{\partial X} \end{bmatrix} \tag{5.35}$$

$$\mathbf{D} = \begin{bmatrix} C_{1111} & C_{1122} & C_{1133} & C_{1112} & C_{1123} & C_{1113} \\ & C_{2222} & C_{2233} & C_{2212} & C_{2223} & C_{2213} \\ & & C_{3333} & C_{3312} & C_{3323} & C_{3313} \\ & & & C_{1212} & C_{1223} & C_{1213} \\ & \text{symm.} & & & C_{2323} & C_{2313} \\ & & & & & C_{1313} \end{bmatrix}, \tag{5.36}$$

and rewrite the first term of the above equation in matrix form as

$$|P|\boldsymbol{\varepsilon}(\boldsymbol{\Pi}_P^\nabla \boldsymbol{\varphi}_j) : \mathbb{C} : \boldsymbol{\varepsilon}(\boldsymbol{\Pi}_P^\nabla \boldsymbol{\varphi}_k) = |P|(\mathbf{B}^P)^T\,\mathbf{D}\mathbf{B}^P. \tag{5.37}$$

For the second term of (5.34), we can express it in matrix notation as

$$\alpha_P S_{h,P}(\boldsymbol{\varphi}_j - \Pi_P^\nabla \boldsymbol{\varphi}_j, \boldsymbol{\varphi}_k - \Pi_P^\nabla \boldsymbol{\varphi}_k) = \alpha_P \left(\mathbf{I} - \mathbf{S}^P\right)^T \left(\mathbf{I} - \mathbf{S}^P\right), \tag{5.38}$$

where $\mathbf{S}^P \in \mathbb{R}^{3m\times 3m}$ is given by

$$\mathbf{S}^P = \begin{bmatrix} \mathbf{P}^P_{(11)} & 0 & 0 & \dots & \mathbf{P}^P_{(1m)} & 0 & 0 \\ 0 & \mathbf{P}^P_{(11)} & 0 & \dots & 0 & \mathbf{P}^P_{(1m)} & 0 \\ 0 & 0 & \mathbf{P}^P_{(11)} & \dots & 0 & 0 & \mathbf{P}^P_{(1m)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{P}^P_{(m1)} & 0 & 0 & \dots & \mathbf{P}^P_{(mm)} & 0 & 0 \\ 0 & \mathbf{P}^P_{(m1)} & 0 & \dots & 0 & \mathbf{P}^P_{(mm)} & 0 \\ 0 & 0 & \mathbf{P}^P_{(m1)} & \dots & 0 & 0 & \mathbf{P}^P_{(mm)} \end{bmatrix} \tag{5.39}$$

with $\mathbf{P}^P$ defined in (5.24).

Having computed the local stiffness matrix for each element, we can obtain the global stiffness matrix $\mathbf{K} = \sum_P \mathbf{k}^P$ through the standard assembly procedure and solve the linear system of equation

$$\mathbf{K}\mathbf{U} = \mathbf{F}_{\text{ext}} \tag{5.40}$$

for the nodal displacement vector $\mathbf{U}$, where $\mathbf{F}_{\text{ext}}$ is the external force vector. For a given discretization, the external force vector $\mathbf{F}_{\text{ext}}$ contains the nodal loads computed from applied traction $\mathbf{t}$ and body force $\mathbf{f}$ based on (5.29).

5.2.3 Numerical assessment of the VEM approximation for linear elasticity

In this subsection, the performance of the VEM approximations on convergence and accuracy is assessed through a benchmark problem. In the assessment, we evaluate two error measures of the displacement and stress solutions. The measure of error in the displacement solution is defined as:

$$\varepsilon_{0,\mathbf{u}} = \sqrt{\sum_{P\in\Omega_h} \int_P (\widetilde{\mathbf{u}}_h - \mathbf{u}) \cdot (\widetilde{\mathbf{u}}_h - \mathbf{u}) \mathrm{d}\mathbf{X}}, \tag{5.41}$$

where $\mathbf{u}$ is the exact displacement solution and $\widetilde{\mathbf{u}}_h$ denotes the displacement field obtained by interpolating the VEM DOFs using the 3D Wachspress shape functions [30]. On the other hand, we also define the L^2 norm of the stress error as

$$\varepsilon_{\boldsymbol{\sigma}} = \sqrt{\sum_{P \in \Omega_h} \int_P (\boldsymbol{\sigma}_h - \boldsymbol{\sigma}) \cdot (\boldsymbol{\sigma}_h - \boldsymbol{\sigma}) \mathrm{d}\mathbf{X}}, \tag{5.42}$$

where $\boldsymbol{\sigma}$ is the exact stress solution and $\boldsymbol{\sigma}_h$ is a piecewise constant stress field defined such that

$$\boldsymbol{\sigma}_h|_P = \mathbb{C} : \boldsymbol{\varepsilon}(\boldsymbol{\Pi}_P^{\nabla} \mathbf{u}_h). \tag{5.43}$$

For both error measures, the integrals are evaluated using a 4th order integration rule on each tetrahedral subdivision of P.

We consider a boundary value problem in which a cantilever beam is loaded by end shear. As described in Fig. 5.2 (a), the beam occupies domain $\Omega = (-1,1) \times (-1,1) \times (0,10)$ and is subjected to a constant traction $\mathbf{t} = [0, -\tau, 0]^T$ on its top surface. According to [201], the stress solution of such a problem is given by the following expressions:

$$\begin{aligned}
&\sigma_{xx} = \sigma_{yy} = \sigma_{xy} = 0, \qquad \sigma_{zz} = \frac{3\tau}{4} YZ \\
&\sigma_{xz} = \frac{3\tau\nu}{2\pi^2(1+\nu)} \sum_{n=1}^{\infty} \frac{(-1)^n}{n^2 \cosh(n\pi)} \sin(n\pi X) \sinh(n\pi Y) \\
&\sigma_{yz} = \frac{3\tau(1-Y^2)}{8} + \frac{\tau\nu(3X^2-1)}{8(1+\nu)} - \frac{3\tau\nu}{2\pi^2(1+\nu)} \sum_{n=1}^{\infty} \frac{(-1)^n}{n^2 \cosh(n\pi)} \cos(n\pi X) \cosh(n\pi Y),
\end{aligned} \tag{5.44}$$

where ν is Poisson's ratio. For the stress solution, we can see that, the beam is traction-free on its four lateral surfaces. Additionally, the displacement solution that corresponds to the above stress distributions, up to the addition of a rigid body motion, is given by:

$$\begin{aligned}
u_x &= -\frac{3\tau\nu}{4E_0}XYZ \\
u_y &= \frac{\tau}{8E_0}(3\nu Z(X^2 - Y^2) - Z^3) \\
u_z &= \frac{\tau}{8E_0}\left[3YZ^2 + \nu Y(Y^2 - 3X^2)\right] + \frac{2(1+\nu)}{E_0}w(\mathbf{X}),
\end{aligned} \tag{5.45}$$

where E_0 is Young's modulus, $w(\mathbf{X})$ is the anti-derivative of σ_{yz} with respect to Y, and L is the length of the beam, which is 10 in the present case. In our numerical study, the material properties of the solid are taken to be $E_0 = 25$ and $\nu = 0.3$, and the magnitude of the shear load is chosen to be $\tau = -0.1$. Consistent units are used throughout this chapter. We apply the analytical displacement field given in (5.45) on the bottom surface and analytical traction (calculated from (5.44)) on the top surface. Three families of structured meshes made up of regular space-filling polyhedra (i.e. regular hexahedra, truncated octahedra and rhombic dodecahedra) and an additional family of CVT meshes are considered, as shown in Figs. 5.2 (b)–(e).

The convergence of both displacement error, $\varepsilon_{0,\mathbf{u}}$, and stress error, $\varepsilon_{\boldsymbol{\sigma}}$, as functions of the average mesh size h are depicted in Figs. 5.3 (a) and (b), respectively. The results confirm that optimal convergence rates are obtained for both displacement and stress errors – the rate of convergence for $\varepsilon_{0,\mathbf{u}}$ is 2 and that for $\varepsilon_{\boldsymbol{\sigma}}$ is 1.

5.3 A new VEM-based topology optimization using polytopes

This section introduces a new VEM-based topology optimization framework on polyhedral meshes, which features continuous design and material density functions. We shall focus on the classic compliance minimization problems and remark that the proposed framework is readily applicable to other optimization formations.

For a given discretization Ω_h consisting of non-overlapping polyhedra, the topology

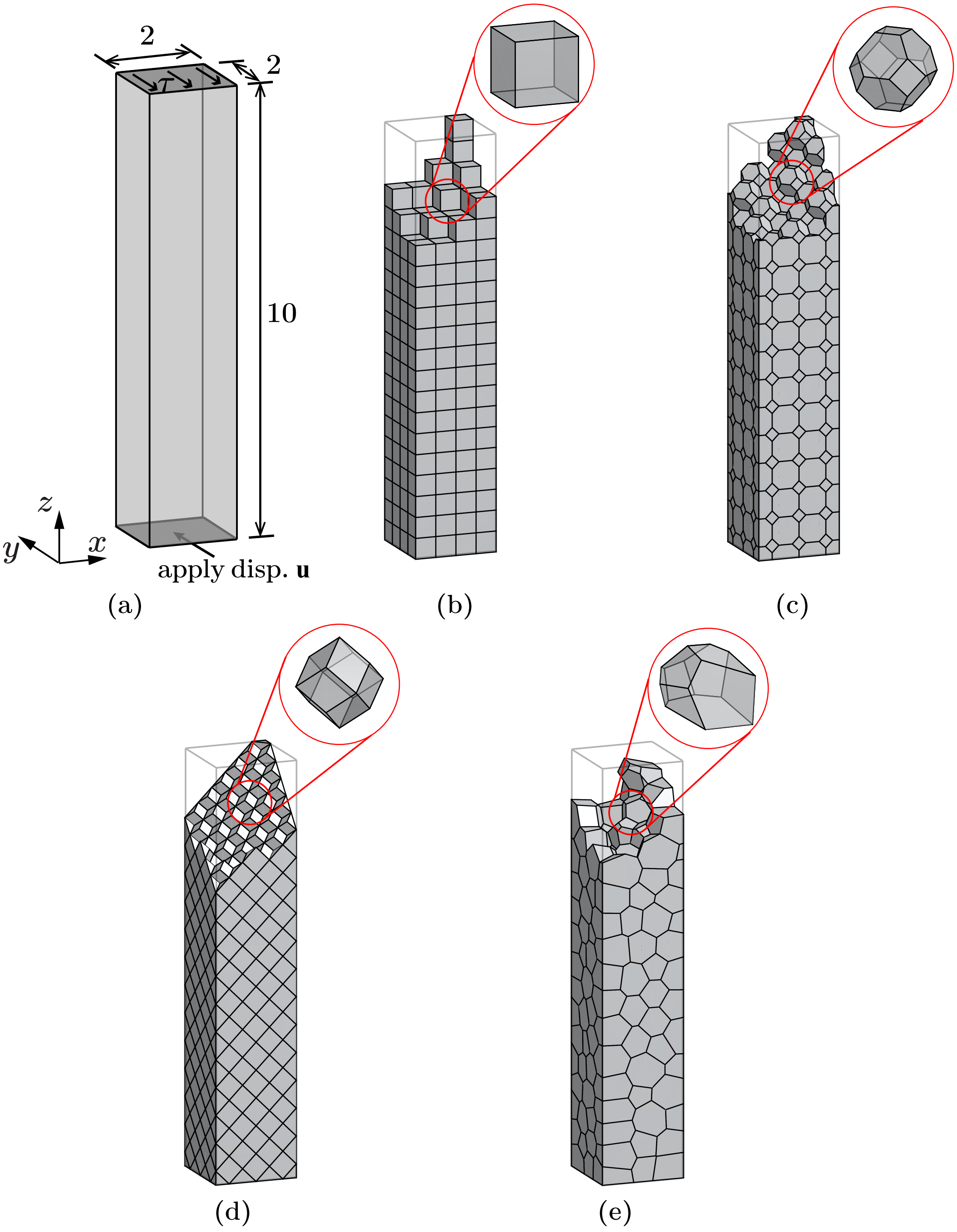


Figure 5.2: (a) Problem description of a cantilever beam loaded by end shear. (b) An example of the mesh consists of regular hexahedra. (c) An example of the mesh consists of truncated octahedra. (d) An example of the mesh consists of Rhombic dodecahedra. (e) An example of the CVT mesh.

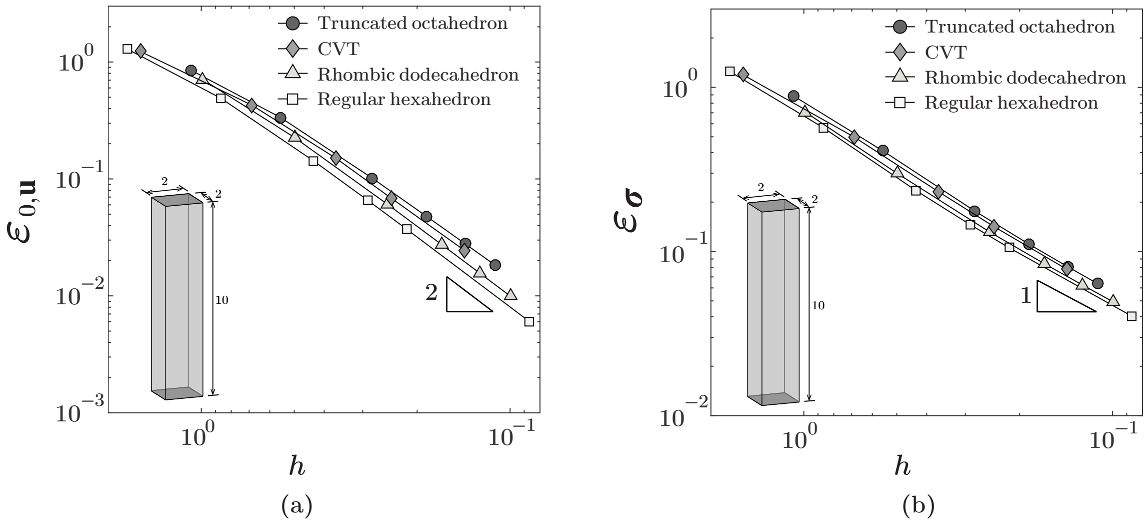


Figure 5.3: Convergence of the $\mathscr{L}^2$ norms of the (c) displacement error $\varepsilon_{\mathbf{u}}$ and (d) stress error $\varepsilon_{\boldsymbol{\sigma}}$ when traction is applied on the top boundary of the block.

optimization formulation for the minimum compliance problems is stated as:

$$\begin{aligned} &\inf_{\rho_h \in \mathscr{A}_h} \int_{\Gamma_h^{\mathbf{t}}} \mathbf{t}\cdot\mathbf{u}_h \\ &\text{s.t.} \quad \frac{1}{|\Omega_h|}\int_{\Omega_h} \rho_h - \overline{V} \leq 0 \\ &\text{with} \quad a_h^{\rho}(\mathbf{u}_h, \delta\mathbf{u}_h) = \ell_h(\delta\mathbf{u}_h) \quad \forall \delta\mathbf{u}_h \in \mathscr{K}_h^0, \end{aligned} \tag{5.46}$$

where ρ_h is a material density function, $\mathscr{A}_h$ is the space of admissible designs and $\overline{V}$ is the allowable volume fraction. To regularized the formulation, the material density function ρ_h is defined as the image of a design function η_h under a map $\mathscr{F}_h$ (e.g. the density filter) and the DOFs of the design function η_h are the design variables (DVs). Moreover, a material interpolation function $m_S(\rho_h)$ is employed to relate the material stiffness to the value of ρ_h at any given point. For instance, if the SIMP model is used [202]–[204], we have $m_S(\rho_h) = \varepsilon + (1-\varepsilon)\rho_h^p$, where ε is the Ersatz parameter and p is the penalization parameter. Incorporating the spatially varying stiffness $m_S(\rho_h)$, $a_h^{\rho}(\mathbf{u}_h, \mathbf{v}_h)$ is the discrete bilinear form constructed using VEM.

The topology optimization formulation (5.46) can be viewed as a two-field mixed ap-

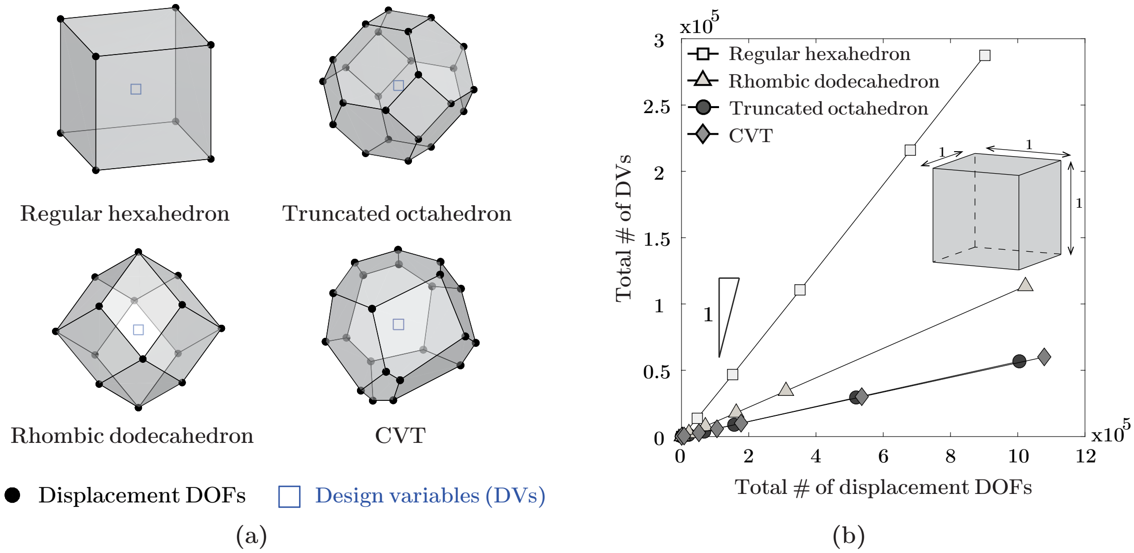


Figure 5.4: (a) Illustrations of the mixed elements adopted in the standard topology optimization framework. The displacement DOFs are located at the vertices and each element contains one DV representing the constant design field. (b) The total numbers of DVs as functions of the total numbers of displacement DOFs for various discretizations of a unit cube consisting of the mixed elements shown in (a).

proximation problem involving a discrete displacement space $\mathscr{K}_h$ and a discrete design space $\mathscr{A}_h$ [177]. The standard density-based topology optimization framework in the literature typically employs a continuous displacement field, whose DOFs are the displacements at the vertices of the mesh, and a piecewise-constant design function, whose value in each element is the associated DV. The material density within each element takes a constant value as well. Several examples of mixed elements of this type is shown in Fig. 5.4 (a). Roughly speaking, for a given discretization, the number of DVs governs the “resolution” of the topology whereas the number of the displacement DOFs (the size of the state equation) dictates the computational cost. If we consider a unit cube and discretize it with the mixed elements shown in Fig. 5.4 (a), we can then plot in Fig 5.4 (b) the total numbers of DVs as functions of the total numbers of displacement DOFs when those meshes are refined. We observe that those functions are close to linear and their slopes can be used to quantify the computational efficiency of the topology optimization framework on various discretizations: the larger slope is the more computationally efficient the discretization is.

As shown from the slopes in Fig. 5.4 (b), the mixed approximation in the standard topology optimization framework leads to considerably more displacement DOFs than of DVs on various discretizations, which is undesirable from a computational efficiency perspective. Moreover, Fig. 5.4 (b) also suggests that polyhedral discretizations yield smaller slopes as compared to the hexahedral ones. This observation indicates that, although polyhedral discretizations exhibits several geometric advantages in topology optimization [65], they are less computationally efficient than hexahedral ones in the standard topology optimization framework.

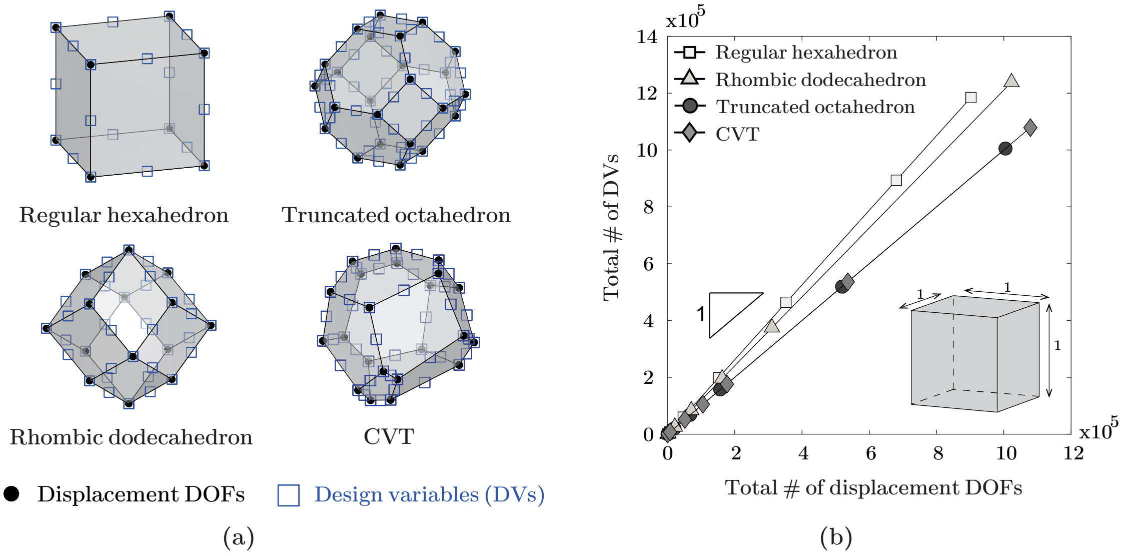


Figure 5.5: (a) Illustrations of new mixed elements proposed in this work. The displacement DOFs are located at the vertices and the DVs are assigned to the vertices as well as mid-edge nodes of each element. (b) The total numbers of DVs as functions of the total numbers of displacement DOFs for various discretizations of a unit cube consisting of the mixed elements shown in (a).

Motivated by the above observations and discussions, we propose a new approximation of the design function (as well as the material density function) for topology optimization on general polyhedral meshes. The basic idea is to consider a more enriched local design space with continuous design functions in each element. In terms of DVs, they are placed at the vertices as well as the mid-edge nodes of the meshes. On the other hand, the displacement approximation is kept the same as in the standard case. This leads to a new mixed approximation for topology optimization on general polyhedral meshes. We note that the idea of enriching the design space is conceptually similar to the multi-resolution methodologies [186] in topology optimization. An illustration of several new mixed elements of this type are shown in Fig. 5.5 (a). Again, consider a unit cube discretized with those mixed elements, we plot in Fig. 5.5 (b) the total numbers of DVs as functions of the total numbers of displacement DOFs with mesh refinement. By comparing the slopes of those functions in Figs. 5.4 (b) and Fig. 5.5 (b), we conclude that the proposed mixed approximation improves the computational efficiency as compared to the standard topology optimization framework. More specifically, for a given discretization with a fixed number of displacement DOFs (thus with roughly fixed computational cost), the proposed mixed approximation yields a more enriched design field and, thus, can produce topologies with improved resolutions.

5.3.1 The new VEM-based topology optimization framework

This subsection formalizes the proposed topology optimization framework. In this framework, both the discrete displacement and design fields are constructed using the VEM. As we can see, because of its flexibility in handling any element geometry as well as local spaces, the VEM provides an efficient platform to formulate the proposed topology optimization framework.

To introduce the new topology optimization framework, we define the space $\mathscr{A}_h$ for admissible density function ρ_h as:

$$\mathscr{A}_h = \{\rho_h = \mathscr{F}_h(\eta_h) : 0 \leq \rho_h \leq 1,\ \rho_h|_E \in \mathscr{V}(P),\ \text{and}\ 0 \leq \eta_h \leq 1,\ \eta_h|_E \in \mathscr{V}(P),\ \forall P \in \Omega_h\}\,. \tag{5.47}$$

In the above definition, we assume that both density function ρ_h and design function η_h are continuous functions with their DOFs being their values at vertices as well as mid-edge nodes of the mesh. The density function ρ_h is defined as the map of a design function η_h

by the density filter $\mathscr{F}_h(\cdot)$. For a discretization Ω_h consisting of M elements and N_D DVs, we henceforth introduce two vectors, $\mathbf{z} = [z_1, z_2, ..., z_{N_D}]^T$ and $\mathbf{y} = [y_1, y_2, ..., y_{N_D}]^T$, where z_i and y_i are values of η_h and ρ_h at the ith DOF of η_h and ρ_h, respectively.

Since both functions ρ_h and η_h can be characterized by their DOFs, we adopt the following approach to construct the density filter map $\mathscr{F}_h(\cdot)$ between ρ_h and η_h, which is based on their DOFs and the associated position vectors. If we denote by $S(i)$ the set of the indices of DOFs whose positions fall within a sphere of prescribed radius R centered at $\mathbf{X}_i$ (the position vector associated with DOF i), y_i (the ith DOF of the density field ρ_h) is computed as

$$y_i = \frac{\sum_{j \in S(i)} z_j (1 - ||\mathbf{X}_i - \mathbf{X}_j||/R)^q}{\sum_{k \in S(i)} (1 - ||\mathbf{X}_i - \mathbf{X}_k||/R)^q}, \tag{5.48}$$

where q is the order of the density filter [205], [206]. Figure 5.6 illustrates this density filter on a 2D mesh patch. For easier implementation, we express the density filter mapping between ρ_h and η_h in a matrix form as

$$\mathbf{y} = \mathbf{P}^{\mathscr{F}} \mathbf{z}, \tag{5.49}$$

where $\mathbf{P}^{\mathscr{F}}$ is a constant and sparse matrix with its (i, j)th component given by

$$\mathbf{P}^{\mathscr{F}}_{(ij)} = \frac{\max(0, (1 - ||\mathbf{X}_i - \mathbf{X}_j||/R)^q)}{\sum_{k \in S(i)} (1 - ||\mathbf{X}_i - \mathbf{X}_k||/R)^q} \tag{5.50}$$

Moreover, within each element P (assuming P has m vertices), the density function ρ_h belongs to the VEM space $\mathscr{V}(P)$ defined in (5.11). Likewise, if we use y_i^P to denote the ith DOFs of ρ_h in P, we can express $\rho_h|_P$ in terms of the set of basis functions for $\mathscr{V}(P)$, $\varphi_1, ..., \varphi_m$, as

$$\rho_h|_P = \sum_{i=1}^{m} \varphi_i y_i^P. \tag{5.51}$$

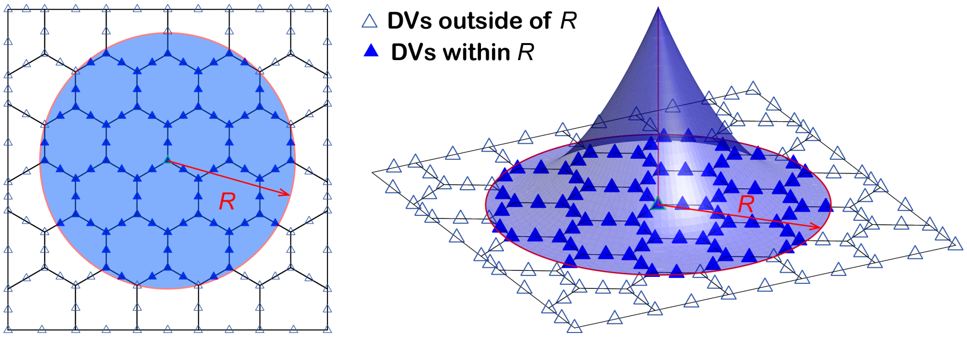


Figure 5.6: An illustration of a quadratic (i.e., $q = 2$) density filter mapping for a 2D mesh patch. The DVs with solid markers are within the filter radius R and thus have non-zero weights.

The volume constraint function in (5.46) can then be recast as

$$\frac{\int_{\Omega_h} \rho_h}{|\Omega_h|} - \overline{V} = \frac{\sum_{P\in\Omega_h}\sum_{i=1}^{m}(\int_P \varphi_i) y_i^P}{\sum_{P\in\Omega_h}|P|} - \overline{V}, \tag{5.52}$$

where we recall from the (5.25) that $\int_P \varphi_i$, $i = 1,...,m$ can be exactly computed by definition of $\mathscr{V}(P)$. To assist easier implementation, we also express the volume constraint function in matrix notation as

$$\frac{\int_{\Omega_h} \rho_h}{|\Omega_h|} - \overline{V} = \frac{\mathbf{V}^T\mathbf{P}^{\mathscr{V}}\mathbf{y}}{\mathbf{V}^T\mathbf{1}} - \overline{V} = \frac{\mathbf{V}^T\mathbf{P}^{\mathscr{V}}\mathbf{P}^{\mathscr{F}}\mathbf{z}}{\mathbf{V}^T\mathbf{1}} - \overline{V}, \tag{5.53}$$

where $\mathbf{V} = [|P_1|, |P_2|, ..., |P_M|]^T$ is a vector collecting element volumes and $\mathbf{P}^{\mathscr{V}}$ is a constant matrix with its (i, j)th component being

$$\begin{aligned}\mathbf{P}^{\mathscr{V}}_{(ij)} &= \begin{cases} \frac{1}{|P_i|}\int_{P_i}\varphi_j & \text{if node } j \in \text{ element } P_i \\ 0 & \text{otherwise} \end{cases} \\ &= \begin{cases} \frac{1}{12}\sum_{F\in\mathscr{F}_j\subset P_i}(\mathbf{X}_{\mathscr{G}_F(j)+1} - \mathbf{X}_{\mathscr{G}_F(j)-1}) \wedge (\mathbf{X}_j - \mathbf{X}_c^P)\cdot(\mathbf{X}_j - \mathbf{X}_c^F) & \text{if } \mathbf{X}_j \in P_i \\ 0 & \text{otherwise} \end{cases}.\end{aligned} \tag{5.54}$$

We also note from the above definition that matrix $\mathbf{P}^{\mathscr{V}}$ is sparse, and thus it is formed and stored as a sparse matrix in our implementation.

In the state equation, the interpolated stiffness function $m_S(\rho_h)$ is utilized, which is assumed to be an element-wise constant functions, such that within element P, $m_S(\rho_h)|_P \doteq m_S(<\rho_h>_P)$, where $<\rho_h>_P$ denotes the volume average of ρ_h over P:

$$<\rho_h>_P = \frac{1}{|P|}\sum_{i=1}^{m}(\int_P \varphi_i)y_i^P. \tag{5.55}$$

Having defined the form of the stiffness interpolation function, we propose to incorporate it in the element-level discrete bilinear form (5.28) as follows

$$\begin{aligned} a_h^{\rho,P}(\mathbf{u}_h,\mathbf{v}_h) = {} & |P| m_S(<\rho_h>_P)\boldsymbol{\varepsilon}(\boldsymbol{\Pi}_P^{\nabla}\mathbf{u}_h) : \mathbf{C} : \boldsymbol{\varepsilon}(\boldsymbol{\Pi}_P^{\nabla}\mathbf{v}_h) \\ & + m_S(<\rho_h>_P)\alpha_P S_{h,P}(\mathbf{u}_h - \boldsymbol{\Pi}_P^{\nabla}\mathbf{u}_h, \mathbf{v}_h - \boldsymbol{\Pi}_P^{\nabla}\mathbf{v}_h). \end{aligned} \tag{5.56}$$

Furthermore, we can pull $m_S(<\rho_h>_P)$ out of the above element-level bilinear form and get the global discrete bilinear form $a_h^{\rho}(\mathbf{u}_h,\mathbf{v}_h)$ as

$$a_h^{\rho}(\mathbf{u}_h,\mathbf{v}_h) = \sum_{i=1}^{M} m_S(<\rho_h>_{P_i}) a_h^{P_i}(\mathbf{u}_h,\mathbf{v}_h). \tag{5.57}$$

We note that, although the L^2 projection of ρ_h onto a linear function is computable over each element [140], we only utilize the L^2 projection of ρ_h onto a constant in each element to construct discrete bilinear form. From an implementation perspective, using an L^2 projection of ρ_h onto a constant instead of a linear function in each element leads to a significantly simplified computational implementation because, by doing so, no numerical integration is needed.

Similarly, we introduce a vector $\mathbf{s}$ in the implementation whose ith component s_i is the value of $m_S(<\rho_h>_P)$ for element P_i. Utilizing the matrix $\mathbf{P}^{\mathscr{V}}$, the vector $\mathbf{s}$ can be expressed as

$$\mathbf{s} = m_S(\mathbf{P}^{\mathscr{V}}\mathbf{y}) = m_S(\mathbf{P}^{\mathscr{V}}\mathbf{P}^{\mathscr{F}}\mathbf{z}). \tag{5.58}$$

We can then compute the global stiffness matrix for $a_h^\rho(m_S(\rho_h), \mathbf{u}_h, \mathbf{v}_h)$ as

$$\mathbf{K}_\rho = \sum_{i=1}^{M} s_i \mathbf{k}^{P_i}. \tag{5.59}$$

Finally, we arrive at the topology optimization formulation considering nodal densities as

$$\begin{aligned} \min_{\mathbf{z}\in[0,1]^{N_D}} \int_{\Gamma_h^{\mathbf{t}}} \mathbf{t}\cdot\mathbf{u}_h &= \min_{\mathbf{z}\in[0,1]^{N_D}} \mathbf{F}_{\text{ext}}^T\mathbf{U} \\ \text{s.t.} \quad \frac{\mathbf{V}^T\mathbf{P}^{\mathscr{V}}\mathbf{P}^{\mathscr{F}}\mathbf{z}}{\mathbf{V}^T\mathbf{1}} &- \overline{V} \leq 0 \\ \text{with } \mathbf{K}_\rho\mathbf{U} &= \mathbf{F}_{\text{ext}}. \end{aligned} \tag{5.60}$$

We next describe how to compute the gradients of the objective and volume constraint functions with respect to the design variable $\mathbf{z}$. For the objective function, we first compute its gradient with respect to the vector $\mathbf{s}$ as

$$\frac{\partial C}{\partial s_j} = -\mathbf{U}^T\frac{\partial \mathbf{K}_\rho}{\partial s_j}\mathbf{U} = -\mathbf{U}^T\mathbf{k}^{P_j}\mathbf{U}, \quad j = 1, ..., M, \tag{5.61}$$

and then, using the chain rule, we arrive at

$$\frac{\partial C}{\partial \mathbf{z}} = (\mathbf{P}^{\mathscr{V}}\mathbf{P}^{\mathscr{F}})^T J_{m_S}(\mathbf{P}^{\mathscr{V}}\mathbf{P}^{\mathscr{F}}\mathbf{z})\frac{\partial C}{\partial \mathbf{s}}, \tag{5.62}$$

where $J_{m_S}(\mathbf{y}) \doteq \text{diag}(m_S'(y_1), ..., m_S'(y_{N_D}))$ is the Jacobian matrix of the stiffness interpolation function m_S. The gradient of the volume constraint function can be simply obtained as

$$\frac{\partial g}{\partial \mathbf{z}} = \frac{(\mathbf{P}^{\mathscr{F}})^T(\mathbf{P}^{\mathscr{V}})^T\mathbf{V}}{\mathbf{V}^T\mathbf{1}}. \tag{5.63}$$

We conclude this subsection by noting that, even with a more enriched space for design and material density functions, the proposed formulation (5.60), which is formulated in the VEM context using local projections, can be implemented in a similar code structure to the `PolyTop` software [110].

5.3.2 A verification example

In this subsection, we verify the proposed continuous density approximation through a simple numerical example and compare its performance with the standard element-wise density approximation. In particular, we want to compare how these two approximations perform in terms of capturing the varying density in a simple boundary value problem.

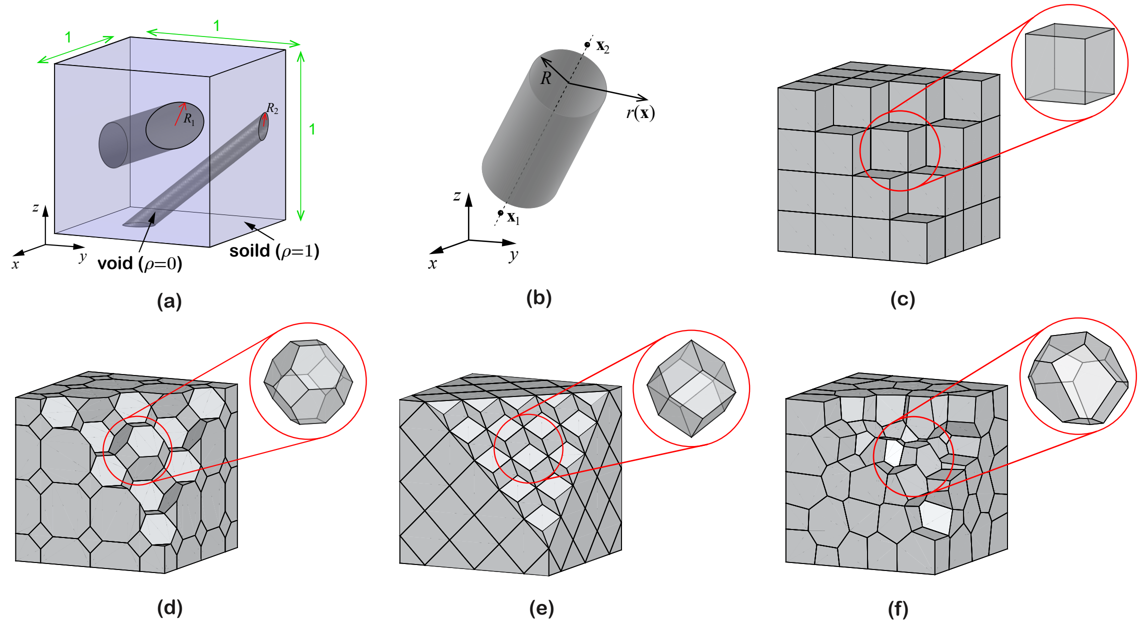


Figure 5.7: (a) Problem description of a unit cube embedded with two hollow tubes. (b) Geometrical parametrization of a tube. (c) An example of the mesh consists of regular hexahedra. (e) An example of the mesh consists of truncated octahedra. (e) An example of the mesh consists of Rhombic dodecahedra. (F) An example of the CVT mesh.

The setup of the boundary value problem is as follows. As shown in Fig. 5.7 (a), we consider a unit cube Ω and place two hollow ($\rho = 0$) tubes in it. The rest of the cube is solid ($\rho = 1$). The tube whose central axis passing through points $[0.22, 0.4, 0]^T$ and $[1, 0.7, 0.78]^T$ has a radius of $R_1 = 0.13$, and the other one whose central axis passing through $[0.5, 0, 0.3]^T$ and $[0.3, 0.6, 1]^T$ has a radius of $R_2 = 0.06$. For this setup, we can introduce a continuous parametrization $\rho(\mathbf{X})$ of the material density over Ω using smooth heaviside functions as

$$\rho(\mathbf{X}) = \frac{1}{1+e^{-200(r_1(\mathbf{X})-R_1)}} + \frac{1}{1+e^{-200(r_2(\mathbf{X})-R_2)}} + \varepsilon - 1, \tag{5.64}$$

where ε is a small positive number assigned to ensure the positivity of $\rho(\mathbf{X})$, and $r_1(\mathbf{X})$ and $r_2(\mathbf{X})$ are the distances of a given point $\mathbf{X}$ to the central axes of the first and second tubes, respectively. As illustrated in Fig. 5.7 (b), for a tube whose central axis passing through two given points $\mathbf{X}_1$ and $\mathbf{X}_2$, $r(\mathbf{X})$ is given by

$$r(\mathbf{X}) = \frac{||(\mathbf{X}-\mathbf{X}_1)\wedge(\mathbf{X}-\mathbf{X}_2)||}{||\mathbf{X}_1-\mathbf{X}_2||}, \tag{5.65}$$

where $||\cdot||$ stands for the Euclidean norm. Making use of $\rho(\mathbf{X})$, the (continuous) distribution of the elasticity modulus in Ω is given by $\mathbb{C}(\mathbf{X}) = \rho(\mathbf{X})\mathbb{C}_0$, where $\mathbb{C}_0$ is the elasticity modulus tensor of the solid material. In the present study, the solid material is considered to be isotropic with Young's modulus being $E_0 = 25$ and Poisson's ratio being $\nu = 0.3$.

Adopting the method of manufactured solutions, we assume an exact displacement solution $\mathbf{u}$ of the form

$$u_x = Z^2 e^{XY} \quad u_y = 2Y^2Z^3 + ZX^4 \text{ and } u_z = Z\sin(2\pi X)e^Y, \tag{5.66}$$

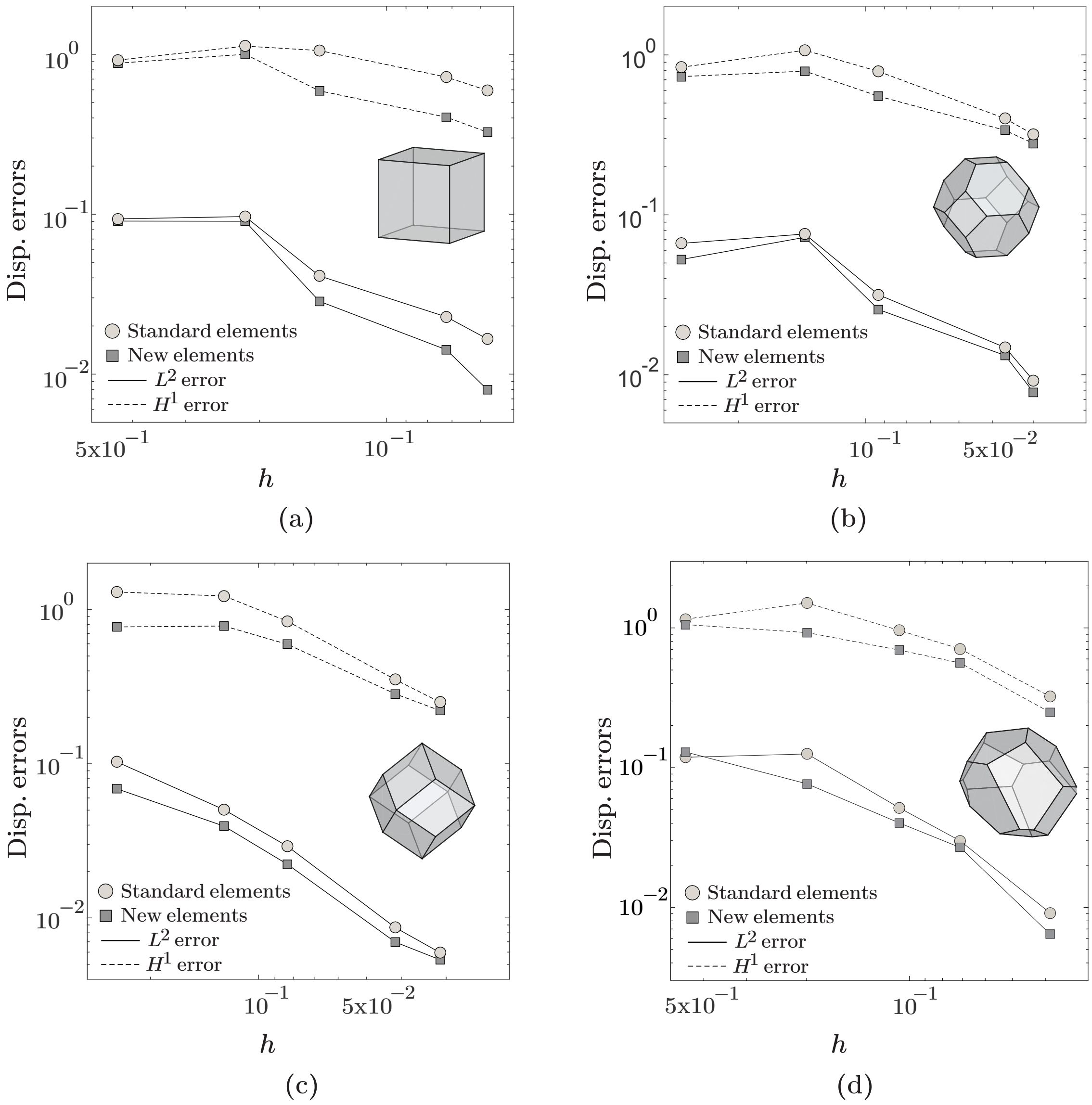
and prescribe it on the entire boundary of the cube. Accordingly, a body force, which is computed using the exact displacement solution $\mathbf{u}$ and stiffness distribution $\mathbb{C}(\mathbf{X})$, is prescribed in the interior of Ω. Four families of polyhedral meshes are considered: hexahedral, truncated octahedral, rhombic dodecahedral, and CVT; and their examples are shown in Figs. 5.7 (c)–(F), respectively. To assess the accuracy of the solution, we make use of the L^2 error of the displacement defined in (5.41) and an H^1 displacement error, which is defined as

$$\varepsilon_{1,\mathbf{u}} = \sqrt{\sum_{P\in\Omega_h}\int_P (\nabla(\mathbf{\Pi}_E^{\nabla}\mathbf{u}_h) - \nabla\mathbf{u})\cdot(\nabla(\mathbf{\Pi}_E^{\nabla}\mathbf{u}_h) - \nabla\mathbf{u})\mathrm{d}\mathbf{X}}, \tag{5.67}$$

where the integral is evaluated using the same 4th order integration rule on each tetrahedral subdivision of P.

In the numerical simulations, the material distribution $\rho(\mathbf{X})$ needs to be approximated in order to compute the stiffness matrix. We consider two approaches here. The first approach assigns a constant density to each element with the value being $\rho(\mathbf{X})$ evaluated at the centroid of that element. This approach resembles the situation in the standard density-based topology optimization where element-wise constant densities are used. On the other hand, the second approach evaluates $\rho(\mathbf{X})$ at the vertices and mid-edge nodes, and interpolates them using VEM basis functions. This approach resembles the situation in the proposed topology optimization framework where the DVs are located at the vertices as well as mid-edge nodes of the mesh. Once the material distribution is approximated, the stiffness matrix of the discretized system for both approaches can be constructed. For the former approach, the stiffness matrix is constructed using the standard procedure whereas, in the latter approach, the stiffness matrix is formed following (5.57).

Figures 5.8 (a)–(d) show the convergence of both L^2 and H^1 displacement errors as functions of mesh size h for both approaches on the four families of meshes. The standard elements represent the first approach and the new elements stand for the second approach. As we can see from the comparisons, the second approach (corresponding to the proposed topology optimization framework) always gives more accurate displacement solutions than the first approach (corresponding to the standard density-based framework) on a given mesh. This in turn implies that the material densities interpolated from values on both vertices and mid-edge nodes provide better approximations of the exact density distribution $\rho(\mathbf{X})$ than the one constructed from element-wise constant values. Based on this observation, we argue that, in topology optimization, it is also more favorable to use the proposed material density approximation because it gives a better parametrization of the density distribution of the varying topologies.



(a)

(b)

(c)

(d)

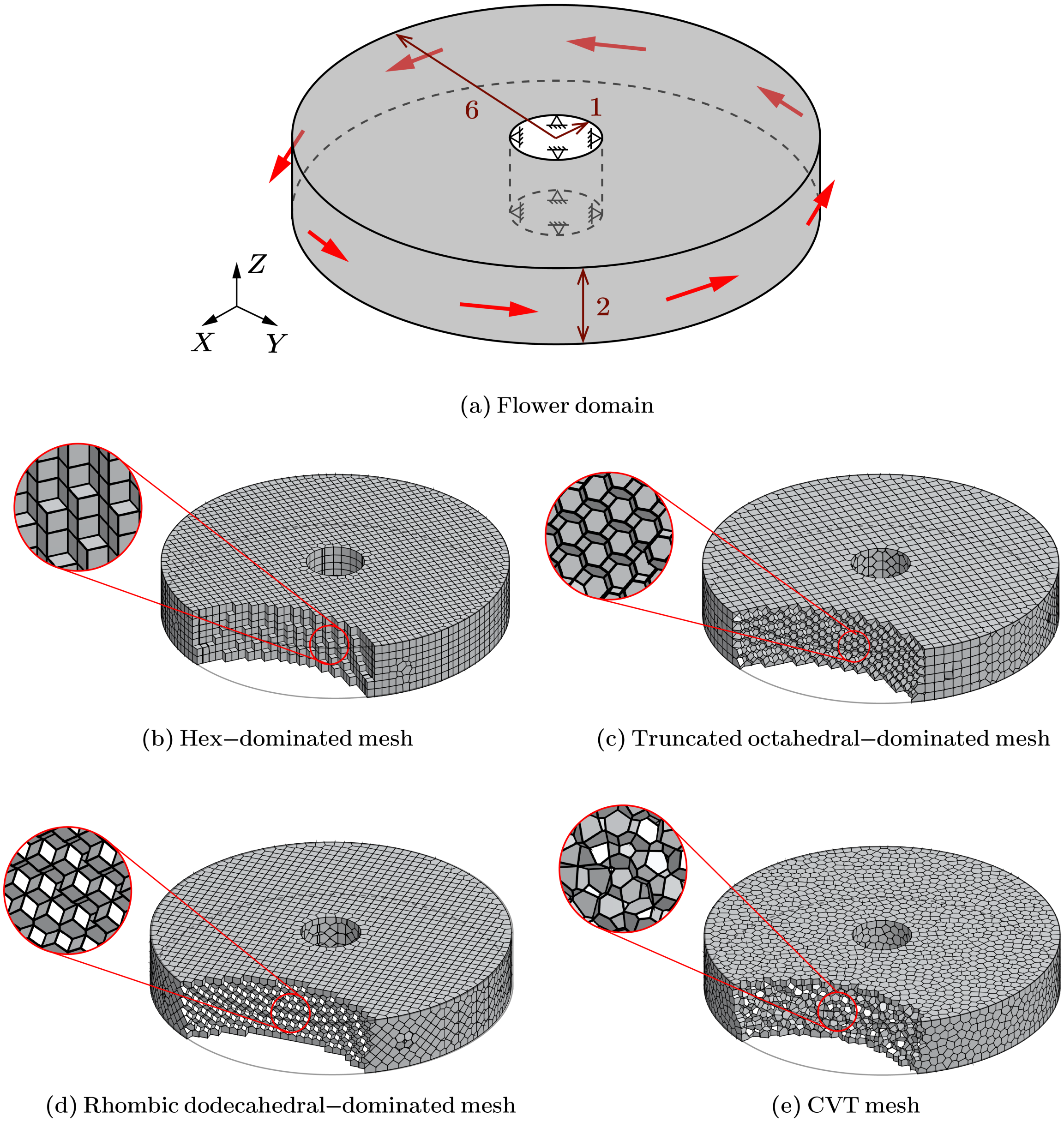
Figure 5.8: Convergence of both L^2 and H^1 displacement errors as functions of the average mesh sizes h when the standard and proposed mixed elements are used for: (a) regular hexahedral meshes; (b) truncated octahedral meshes; (c) rhombic dodecahedral meshes; and (d) CVT meshes.

5.4 Numerical examples

In this section, we present several examples involving non-Cartesian design domains to demonstrate the effectiveness and versatility of the proposed VEM-based topology optimization framework. For all the design examples, we select the Ersatz parameter as $\varepsilon = 10^{-9}$ and take Young's modulus and Poisson's ratio of the solid phase to be $E_0 = 100$ and $\nu = 0.3$, respectively. During optimization, the Optimality Criteria (OC) [172] is chosen as the design-variable update scheme with the damping parameter and move limit being $\eta = 0.5$ and $move = 0.3$. The maximum tolerance for the change of design variables in the convergence criterion is taken to be 0.1%. Moreover, a continuation scheme of the penalization parameter p is adopted here. We initialize p as $p = 1$ and increase it every (maximum) 20 optimization iterations by 1 until $p = 3$. When p reaches 3, we then set the maximum allowable optimization iteration number to be 150. For comparison purposes, the topologies obtained by the standard topology optimization framework [65] with element-wise constant densities are also provided. The same VEM formulation is used to solve the state equation in the standard framework as in the proposed framework. Unless otherwise stated, the final topologies are plotted using iso-surface with the cut-off value being 0.5. To distinguish the topologies obtained using the standard framework and the proposed one, we plot the results obtained using the standard framework in blue and those obtained using the proposed framework in red. We also point out that, for a fixed mesh and fixed set of parameters, the computational time of both proposed and standard frameworks are closely identical because the sizes of state equations are the same.

5.4.1 Shear loaded disc problem

The first design example is the shear loaded disc problem. As shown in Fig. 5.9, the design domain is a disc with an outer radius of 6 and an inner radius of 1. The thickness of the disc is taken to be 2. Fixed in the inner surface, the circumference of this disc is



(a) Flower domain

(b) Hex–dominated mesh

(c) Truncated octahedral–dominated mesh

(d) Rhombic dodecahedral–dominated mesh

(e) CVT mesh

Figure 5.9: (a) The geometry, load and boundary conditions of the shear loaded disc problem. (b) A hexahedral-dominated mesh consisting of $31,791$ nodes and $12,180$ elements. (c) A truncated octahedral-dominated mesh consisting of $65,418$ nodes and $10,808$ elements. (d) A rhombic dodecahedral-dominated mesh consisting of $52,606$ nodes and $12,264$ elements. (e) A CVT mesh consists of $64,097$ nodes and $10,000$ elements.

subjected to eight equidistant shear load of uniform magnitude 1. A volume fraction of 10% is prescribed.

In order to provide a thorough assessment of the performance of the proposed framework on various types of meshes, we consider four meshes in this design example: a hexahedral-dominated mesh, a truncated octahedral-dominated mesh, a rhombic dodecahedral-dominated mesh, and a CVT mesh, as shown in Fig. 5.9 (b)–(e), respectively. The first three meshes consist of regular space-filling polyhedra in the interior of the design domain and unstructured polyhedra in the boundary regions, while the CVT mesh is made up of unstructured polyhedra inside of the entire design domain. The statistics of this mesh is provided in Table 5.1. Notice that the four meshes have similar numbers of elements.

Table 5.1: Statistics of the meshes for the shear loaded disc problem.

Mesh	Node #	EL. #	DV # (Proposed)	DV # (Standard)
Hexahedral	31,791	12,180	104,166	12,180
CVT	64,097	10,000	191,256	10,000
Trun. octahedral	65,418	10,808	196,329	10,808
Rhom. dodecahedral	52,606	12,264	171,275	12,264

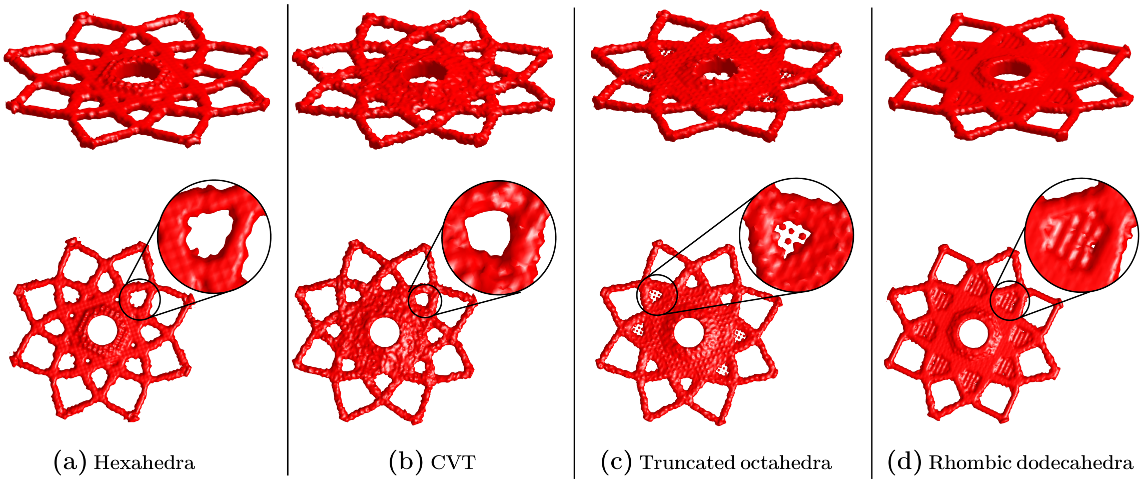
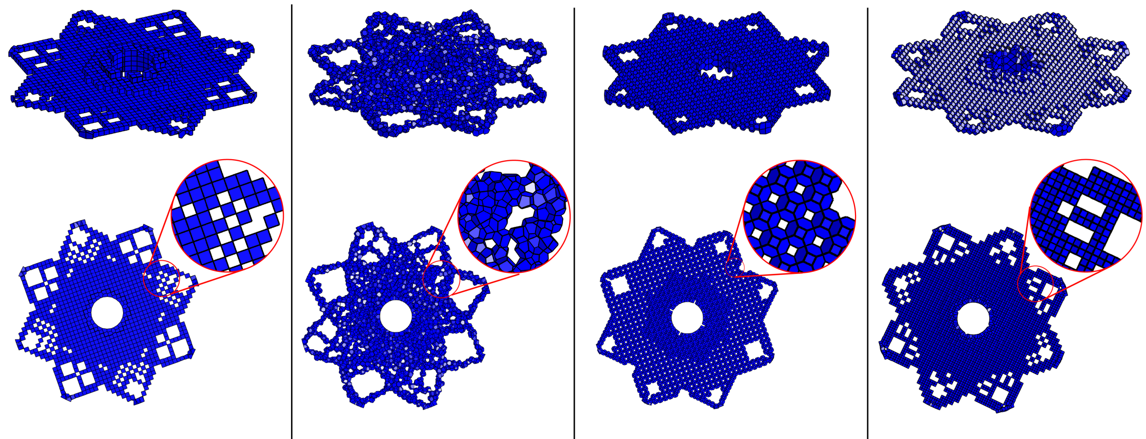

(a) Hexahedra
(b) CVT
(c) Truncated octahedra
(d) Rhombic dodecahedra

Figure 5.10: The final topologies obtained from the proposed topology optimization without the density filter on (a) the hexahedral-dominated mesh; (b) the truncated octahedral-dominated mesh; (c) the rhombic dodecahedral-dominated mesh; and (e) the CVT mesh.

We first design the shear loaded disc problem without applying the density filter. Figures 5.10 and 5.11 show the final topologies obtained from the proposed and the standard



(a) Hexahedra (b) CVT (c) Truncated octahedra (d) Rhombic dodecahedra

Figure 5.11: The final topologies obtained from the standard topology optimization without the density filter on (a) the hexahedral-dominated mesh; (b) the truncated octahedral-dominated mesh; (c) the rhombic dodecahedral-dominated mesh; and (d) the CVT mesh. In the designs, those elements whose densities are above 0.01 are plotted.

frameworks, respectively, on the four meshes. Notice that, for the results obtained with the standard topology optimization framework, we get almost black and white designs for all the four meshes. Thus, for those results, instead of showing the iso-surface plots of the final topology, we simply plot those elements whose densities are above 0.01. In the topology optimization literature, it is well-known that, without density or sensitive filters, quadrilateral or hexahedral meshes will produce checkerboard patterns in the standard framework, e.g., see Fig. 5.11 (a). For general discretizations, previous work has demonstrated that 2D regular hexagonal [18] and 2D and 3D CVT meshes [17], [19], [65] are free of checkerboard patterns. From our numerical results in Fig. 5.11 (c) for truncated octahedral and Fig. 5.11 (d) for rhombic dodecahedra, *we observe that, unlike the regular polygonal discretization in 2D, regular polyhedral discretization in 3D tends to generate patterns indicating numerical instability*. Those patterns provide artificial stiffness and are impractical from a manufacturing perspective. On the contrary, with proposed topology optimization framework, we obtain physical designs on all four meshes considered, which resemble a flower, although their iso-surface plots exhibit rough boundaries because of the absence of regularizations from the density filter.

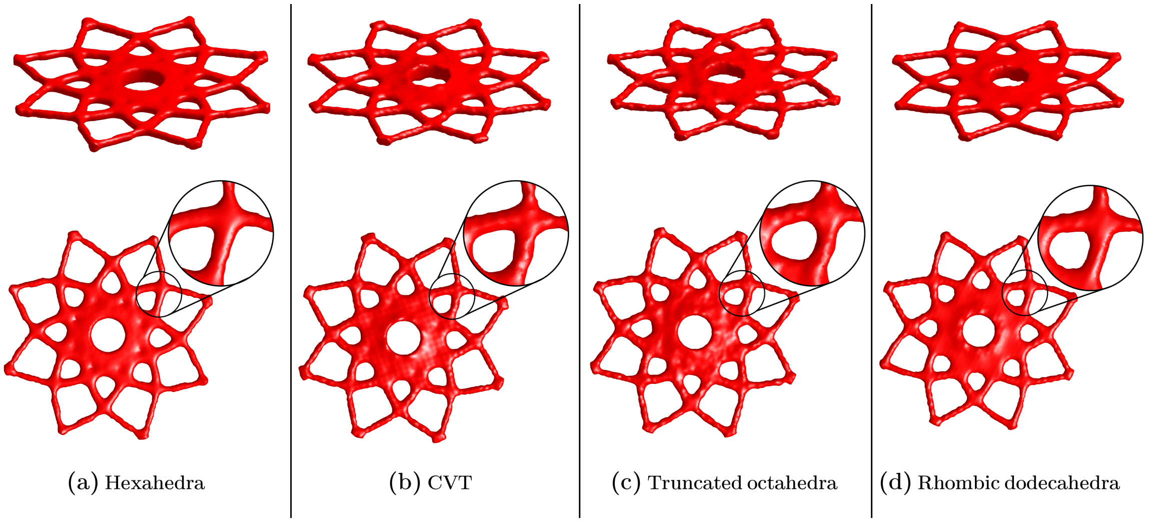


Figure 5.12: The final topologies obtained from the proposed topology optimization with a quadratic density filter of radius $R = 0.45$ on (a) the hexahedral-dominated mesh; (b) the truncated octahedral-dominated mesh; (c) the rhombic dodecahedral-dominated mesh; and (e) the CVT mesh.

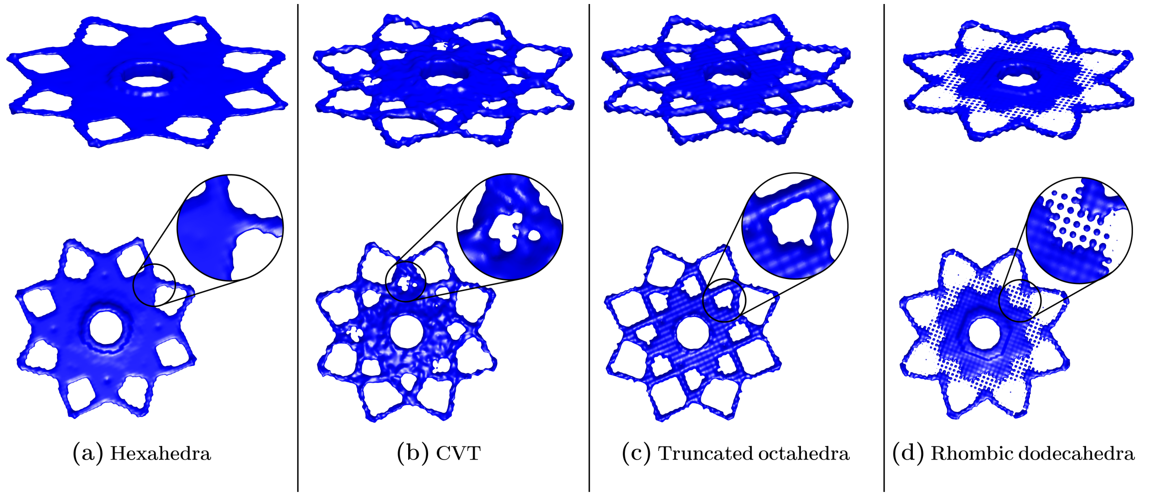


Figure 5.13: The final topologies obtained from the standard topology optimization with a quadratic density filter of radius $R = 0.45$ on (a) the hexahedral-dominated mesh; (b) the truncated octahedral-dominated mesh; (c) the rhombic dodecahedral-dominated mesh; and (e) the CVT mesh.

We then apply a quadratic density filter (i.e. $q = 2$) with a radius of $R = 0.45$, and keep the other parameter settings unchanged. The results obtained are shown in Figs. 5.12 and 5.13 using the proposed and standard framework, respectively. Several observations can be made. First, compared to the ones obtained from the standard framework, the designs obtained from the proposed framework have iso-surface presentations with greatly improved smoothness. This is because, with the proposed framework, we have more DVs as compared to the standard framework on the same discretization, as shown in Table 5.1. Second, unlike the ones obtained using the standard framework, the optimal topology obtained using the proposed framework exhibit similar topologies on all meshes, suggesting that the proposed framework is less sensitive to mesh types than the standard framework.

5.4.2 Hook design

Having investigated the performance of the proposed topology optimization framework through the last design example, we now apply it to several problems involving non-Cartesian design domains. Let us first look at the hook problem. As shown in Fig. 5.14 (a), the design domain in this problem is fixed in the upper half of the circle and is subjected to a uniformly distributed line load in the negative z direction. The volume fraction of this problem is set as $\overline{V} = 15\%$ and the radius of the quadratic density filter is $R = 2.8$. The polyhedral mesh considered in this problem is made up of regular hexahedra in the interior and unstructured polyhedral in the regions near the boundary, as shown in Fig. 5.14 (b). The statistics of this mesh is provided in Table 5.2.

Table 5.2: Statistics of the mesh for the hook problem.

Mesh	Node #	EL. #	DV # (Proposed)	DV # (Standard)
Hexahedral	$83,235$	$33,928$	$275,293$	$33,928$

In Figs. 5.14 (c) and (d), we show the converged designs from both the proposed and standard topology optimization frameworks. We can see that the design obtained from the proposed framework (Fig. 5.14 (c)) possesses a smoother iso-surface representation than the one obtained from the standard framework (Fig. 5.14 (d)). The main difference of the two designs is in the fan region of the hook: the one obtained from the proposed framework has a nice and clear fan region similarly to the design reported in the literature for a 2D version of this problem [110]; whereas the members in the fan region of the design obtained from the standard framework seem to be affected by the structure of the mesh considered.

5.4.3 Wrench design

We also investigate the design of a wrench problem. The design domain of this problem is depicted in Fig. 5.15 (a). In the domain, the bigger circle is fixed and a half of the other circle is subjected to a distributed line load along the negative Y direction. Here, we consider two CVT meshes, a coarse one and a refined one, whose statistics are given in Table 5.3. Again, a quadratic density filter is applied with a radius of $R = 0.05$ and the volume fraction is prescribed as $\overline{V} = 15\%$. In this design example, we prescribe another requirement that the final topology has to be symmetric. To achieve this, we use a matrix $\mathbf{P}^{\mathscr{S}}$ introduced in [110], which is given by

$$\mathbf{P}^{\mathscr{S}}_{(\ell k)} = \begin{cases} 1 & \text{if the } \ell\text{th and } k\text{th DOFs of the density are symmetric in } Y \text{ direction} \\ 0 & \text{otherwise} \end{cases} \tag{5.68}$$

and the vectors $\mathbf{Y}$ and $\mathbf{s}$ are then given by

$$\mathbf{Y} = \mathbf{P}^{\mathscr{S}}\mathbf{P}^{\mathscr{F}}\mathbf{z} \text{ and } \mathbf{s} = m_S(\mathbf{P}^{\mathscr{V}}\mathbf{P}^{\mathscr{S}}\mathbf{P}^{\mathscr{F}}\mathbf{z}) \tag{5.69}$$

respectively. As we can see, the matrix $\mathbf{P}^{\mathscr{S}}$ enforces symmetry through mapping the admissible topologies in the design space to symmetric configurations.

Figures 5.16 (a) and (b) depict the final topologies obtained using both the proposed and standard frameworks for the coarse and refined CVT meshes, respectively. As an

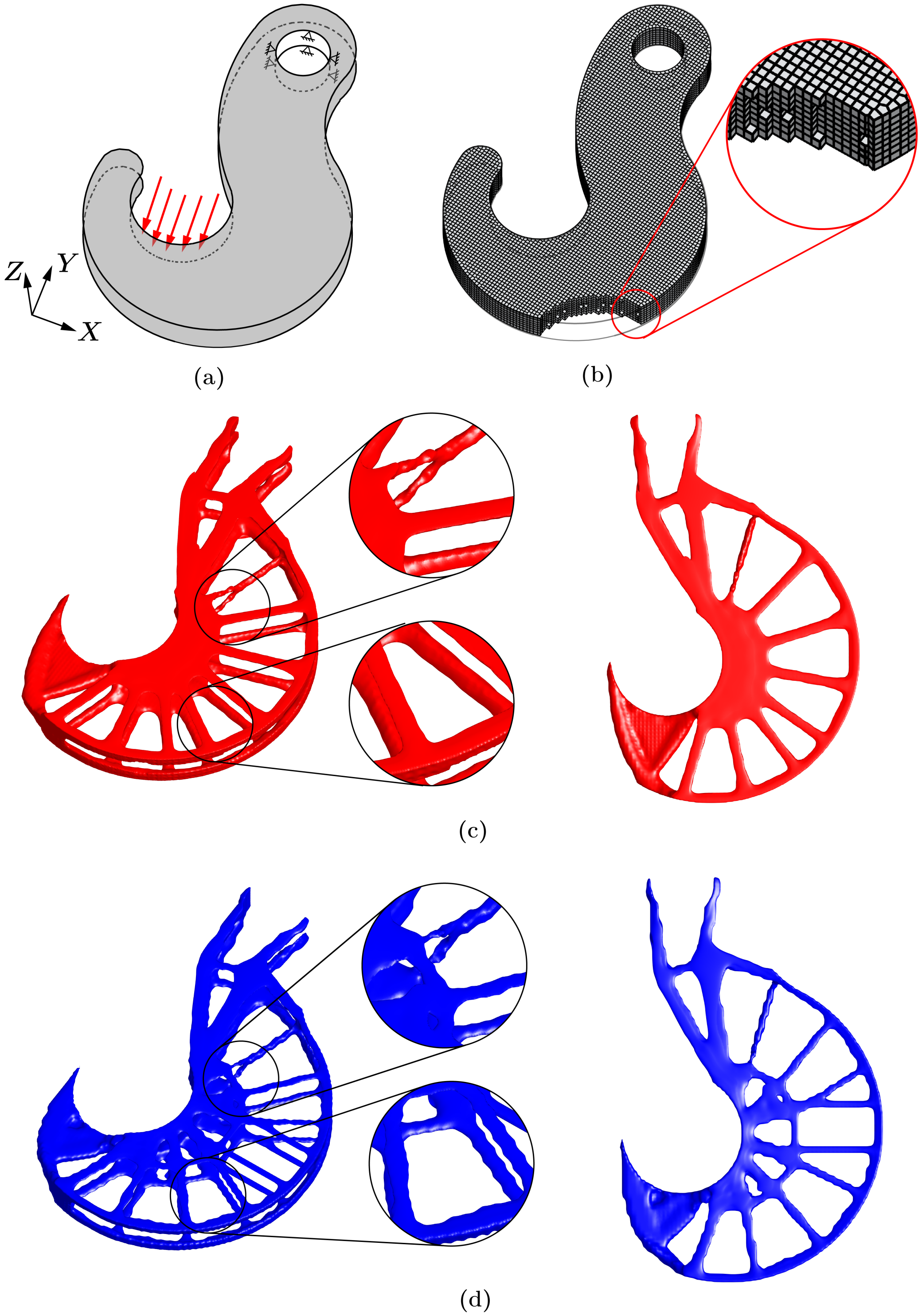
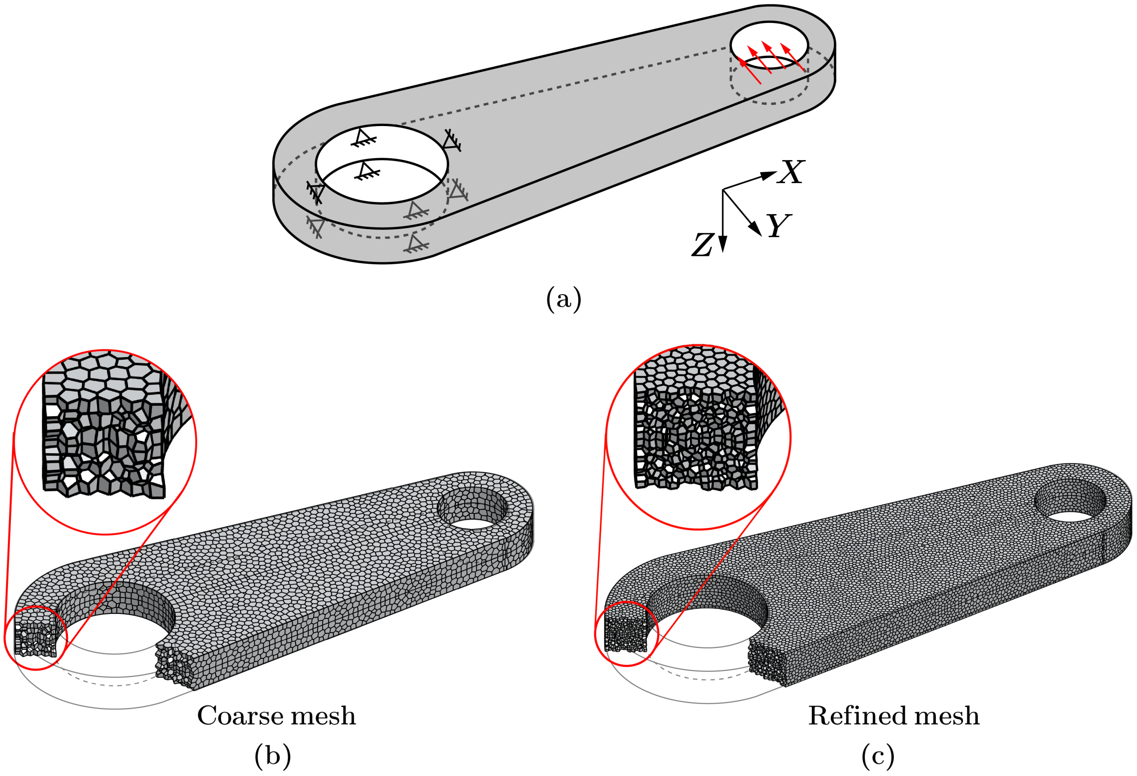


Figure 5.14: (a) The geometry, load and boundary conditions of the hook problem. (b) A hexahedral-dominated mesh consisting of 83,235 nodes and 33,928 elements. The final topologies for the hook problem obtained from the (c) proposed and (d) standard frameworks. Both topologies are obtained with a quadratic density filter of radius $R = 2.8$ and prescribed volume fraction of $\overline{V} = 15\%$.



(a)

Coarse mesh

(b)

Refined mesh

(c)

Figure 5.15: (a) The geometry, load and boundary conditions of the wrench problem. (b) A relatively coarse CVT mesh consisting of $68,339$ nodes and $12,000$ elements. (b) A relatively refined CVT mesh consisting of $349,748$ nodes and $60,000$ elements. Both meshes are symmetric with respect to Y axis and obtained by reflecting the Voronoi seeds along the Y axis.

Table 5.3: Statistics of the meshes for the wrench problem.

Mesh	Node #	EL. #	DV # (Proposed)	DV # (Standard)
CVT (Coarse)	$68,339$	$12,000$	$205,724$	$12,000$
CVT (Refined)	$349,748$	$60,000$	$1,052,109$	$60,000$

immediate observation, the proposed and standard frameworks yield symmetric designs similar to each other for both the coarse and refined meshes. With the coarse mesh, the proposed framework yields a clearer and manufacturable design, which resembles the 2D optimization result in [110], than the standard framework, indicating that the proposed framework is more effective on relatively coarser meshes.

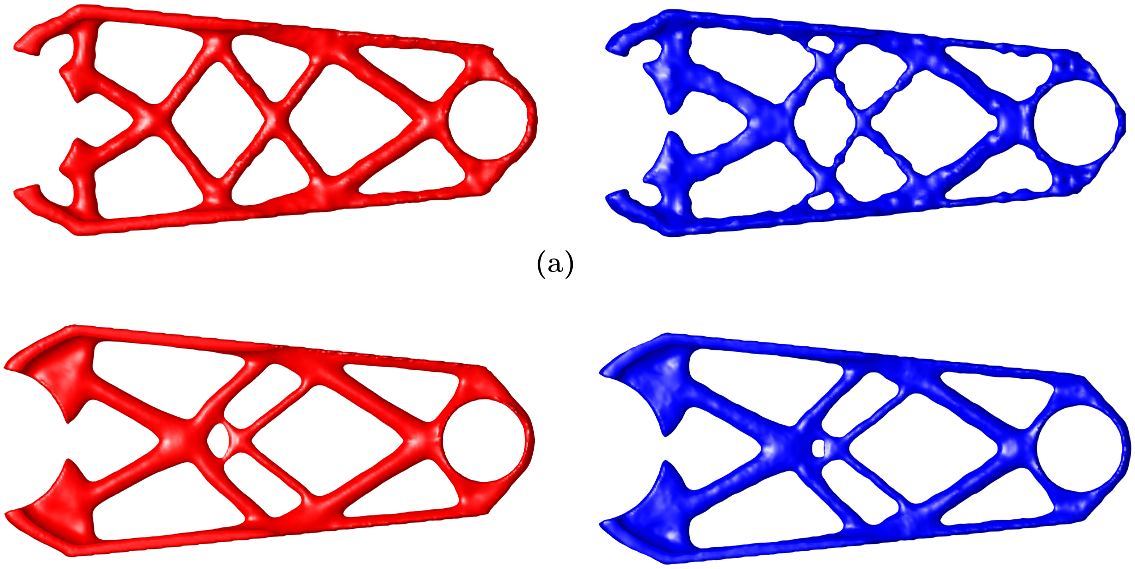


Figure 5.16: The final topologies for the wrench problem obtained from (a) the coarse mesh and (b) the refined mesh. The left and right columns of the figure show the topologies obtained using the proposed and standard frameworks, respectively. All topologies are obtained with a quadratic density filter of radius $R = 0.05$ and prescribed volume fraction of $\overline{V} = 15\%$.

5.4.4 Serpentine design

In the final design example, we perform topology optimization in a serpentine domain. As shown in Fig. 5.17 (a), the serpentine domain is fixed on its left face and is subjected to a point load along the negative Z direction in the middle of the lower edge of the right face. A volume fraction of $\overline{V} = 10\%$ is prescribed and a quadratic density filter of radius $R = 0.4$ is used. As in the hook example, we consider a mesh composed of regular hexahedra in the interior and unstructured polyhedra in regions near the boundary. The mesh is plotted in Fig. 5.17 (b) and its statistics is presented in Table 5.4. In Fig. 5.17 (c) and Figs. 5.18, we show both the final design produced by the proposed framework and the manufactured design using the FDM 3D printing, respectively. This example demonstrates that the proposed topology optimization framework can lead to designs that are directly manufacturable [206].

Table 5.4: Statistics of the mesh for the serpentine problem.

Mesh	Node #	EL. #	DV # (Proposed)
Hexahedral	$75,624$	$33,520$	$253,675$

5.5 Concluding remarks

In this work, we present a 3D VEM-based topology optimization framework on general polyhedral discretizations. The unique feature of this work is that it takes full advantage of the VEM and applies it to both structural and optimization problems. In terms of structural problems, the VEM is adopted to solve the state equation efficiently and effectively. Because VEM does not require explicit computations of the shape functions and their gradients, it does not need numerical integration and is less sensitive to degenerated polyhedra (e.g. ones with skinny faces or small edges) as compared to the FEM. In terms of optimization problems, exploiting the great flexibility of VEM in element geometries and local space definitions, we introduce an enhanced VEM space for the continuous design and material densities functions, which contain DOFs at the vertices as well as the mid-edge nodes of the mesh. The total volume of any topologies in this design space can be computed exactly using a properly defined VEM projection operator. As a result, for a given mesh and under similar computational cost, the proposed VEM based topology optimization is shown to produce designs with improved geometrical resolutions as compared to the

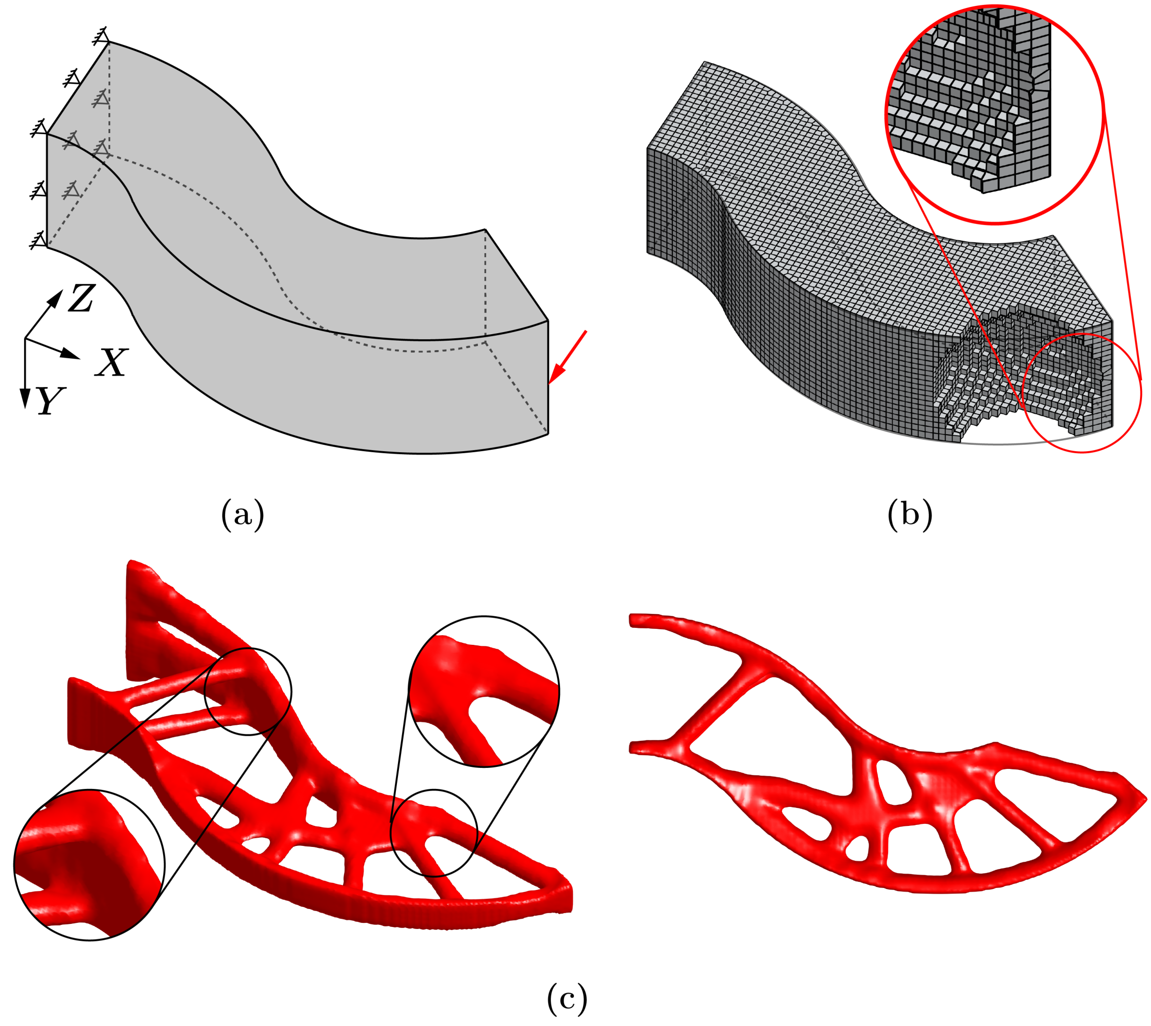


Figure 5.17: (a) The geometry, load and boundary conditions of the serpentine problem. (b) A regular hexahedral-dominated mesh consisting of 75,624 nodes and 33,520 elements. (c)The final topologies for the hook problem obtained from the proposed framework with a quadratic density filter of radius $R = 0.4$ and prescribed volume fraction of $\overline{V} = 10\%$.

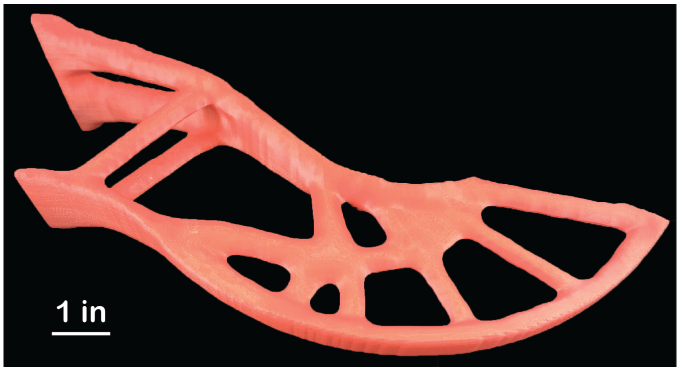


Figure 5.18: A printed model of the serpentine design using the FDM 3D printing.

standard topology optimization framework with element-wise constant DVs and material densities. We also demonstrate that the VEM-based topology optimization framework can be implemented in a similar code structure to the `PolyTop` software [17]. In terms of discretizating complex domains in 3D, this work explores two approaches: unstructured polyhedral (i.e. CVT) meshes and regular polyhedra (i.e. hexahedra, truncated octahedra and rhombic dodecahedra) dominated meshes.

Both of the VEM approximations for the displacement field and material density function are verified through numerical examples. The convergence of the VEM in solving the state equations is verified via a benchmark problem involving beam bending. Through a simple example, we also compare the performance of the proposed continuous density parametrization with the commonly used element-wise constant one in terms of approximating varying material density distributions. Moreover, several design examples involving non-Cartesian domains are presented, showcasing that the VEM-based topology optimization framework can produce optimal designs which have smoother boundaries and are less biased to the initial meshes.

Finally, we remark that the proposed VEM-based topology optimization offers an effective tool for mesh adaptation in topology optimization. The potential of this research includes developing efficient mesh adaption strategies for topology optimization by exploiting the advantages of polyhedral elements.

CHAPTER 6
CONCLUSIONS AND FUTURE DIRECTIONS

This thesis aims to develop novel numerical approaches to more effectively solve computational mechanics problems on general discretizations with emphases on finite elasticity and topology optimization problems. Additionally, to fully realize the potential of polygonal and polyhedral elements in adaptivity, this thesis also introduces a simple yet effective recovery-based *a posteriori* error estimator that is able to accurately capture both local and global errors on general discretizations without the knowledge of exact solutions. In this chapter, the contributions of this thesis are summarized and suggestions for future work are provided.

6.1 Concluding remarks

In Chapter 2, lower- and higher-order polygonal finite element formulations using generalized barycentric coordinates are developed for finite elasticity problems. One critical aspect in the development is related to the numerical integration. Because the generalized barycentric coordinates are typically non-polynomial in nature, the existing integration schemes used in the standard FEM typically suffer from consistency errors that do not vanish with mesh refinement. As shown by the numerical studies in Chapter 2, the consistency errors do not vanish under mesh refinement when a standard integration scheme is used, which lead to non-convergent finite element results. This problem becomes even more severe in higher-order cases. To address this issue, Chapter 2 proposes to employ a general gradient correction scheme which can be applied to the displacement approximation of arbitrary order with any numerical integration scheme that satisfies the corresponding minimal order requirement. With the correction scheme, polygonal finite elements with both linear and quadratic displacement approximations have been shown to yield optimally convergent results in both displacement-based and mixed formulations. Furthermore, the gradient correction scheme enables us to confidently study various choices of pressure approximations. In this thesis, two choices of the pressure approximations have been discussed for the mixed polygonal finite elements, namely, the discontinuous and continuous ones, and their performance on inf-sup stability and accuracy is investigated. Our numerical studies indicate that both families of mixed polygonal finite elements seem to be inf-sup stable in linear and finite elasticity problems without any additional stabilization treatments up to the quadratic order. For demonstration purposes, we also employed the proposed polygonal finite element formulations to model the nonlinear response of particle reinforced elastomers with interphases and the onset of cavitation in fiber reinforced elastomers. These challenging physically-based applications indicate that the proposed polygonal finite element formulations are well suited to deal with complex microstructures including particulate microstructures and those involving different length scales, and appear to be more tolerant to large local deformation than the standard finite elements (triangles and quads).

The polygonal and polyhedral finite elements are limited by the availabilities of well-defined shape functions and efficient yet consistent numerical integration schemes. Especially in 3D, those restrictions considerably limit the element geometry that the method can handle. As an alternative approach, the VEM is able to handle a significantly more general class of discretizations in both 2D and 3D than the FEM as it abstracts from the shape functions. In Chapter 3, we introduce a VEM framework for 2D and 3D finite elasticity problems. Two VEM formulations are presented, which adopt a displacement-based and a two-field mixed variational principles respectively. The displacement-based VEM formulation appears to be free of volumetric locking as the material becomes nearly incompressible, at least for the proposed set of meshes. By construction, the proposed VEM formulations are able to efficiently handle a more general class of polygonal and polyhedral meshes, including the ones with non-convex elements. Several numerical studies are presented, which confirm the convergence and accuracy of the VEM formulations. In particular, for 3D problems, our numerical studies have further shown that the VEM formulations appear to produce convergent results even for meshes containing non-star shaped elements, which makes the VEM formulation even more forgiving with respect to the quality of the mesh. We also show that, with a proper definition of local VEM spaces, an L^2 projection of the volumetric strain can be exactly computable in addition to the standard VEM projections of the displacement and its gradient. The L^2 projection of the volumetric strain renders the VEM more accurate and robust, especially for irregular meshes, such as those containing elements with non star-shaped faces and in problems that involve large heterogeneous and localized deformations. Furthermore, different constructions of the loading terms are discussed and various stabilization strategies are studied, which are shown to have significant influence on the performance of the VEM formulations in finite elasticity problems. A stabilization scheme is further proposed in this chapter for isotropic materials, which is based on the trace of the material tangent modulus tensor. We deploy the proposed VEM formulations to the study of the nonlinear elastic response of a filled elastomer in 2D and demonstrate that they are able to capture large localized deformation fields in such problems.

Because of their flexible geometry, polygonal and polyhedral elements hold great potential in the adaptive analysis. To realize this potential, Chapter 4 introduces a simple yet effective recovery-based *a posteriori* error estimation framework for lower- and higher-order VEM on polygonal and polyhedral meshes, and demonstrates the theory in the context of linear elasticity. The basic idea of this error estimation framework is to first recover a more accurate displacement gradient through a least square fitting of the displacement DOFs over each patch in the mesh. Based on the recovered gradient, an error estimator is then obtained by evaluating the difference between the recovered and original displacement gradients on the skeleton of the mesh. This skeletal error is shown in the numerical studies to capture the standard L^2 norm of the displacement gradient error extremely well. With thorough numerical studies, the introduced error estimation is shown to be accurate for both linear and quadratic virtual elements on various polygonal/polyhedral meshes and with various types of displacement solutions (e.g. smooth ones, ones with sharp gradients, and ones containing singularities). For linear VEM, the accuracy and effectiveness of the proposed error estimation framework are further demonstrated by comparing it with an SPR-type error estimation as outlined in Appendix C. For higher-order VEM, the numerical studies also suggest that we can neglect the internal displacement DOFs (which are in the form of function moments over the elements) in the gradient recovery procedure without sacrificing accuracy.

Chapter 5 presents a 3D VEM-based topology optimization framework on general polyhedral discretizations. The unique feature of this framework is that it takes full advantage of the VEM and applies it to both structural and optimization problems. In terms of structural problems, the VEM is adopted to solve the state equation more efficiently and robustly. In terms of optimization problems, exploiting the great flexibility of VEM in element geometries and local space definitions, we introduce an enhanced VEM space for the continuous design and material densities functions, which contain DOFs at the vertices as well as the mid-edge nodes of the mesh. The total volume of any topologies in this design space can be computed exactly using a properly defined VEM projection operator. As a result, for a given mesh and under similar computational cost, the proposed VEM based topology optimization is shown to produce designs with improved geometrical resolutions as compared to the standard topology optimization framework with element-wise constant DVs and material densities. We verify the convergence both of the VEM approximations for the displacement field and material density function through numerical examples. Additionally, we also present several design examples involving non-Cartesian domains, showcasing that the VEM-based topology optimization framework can produce optimal designs which have smoother boundaries and are less biased to the initial meshes.

6.2 Suggestions for future work

Through this thesis, we have demonstrated that the VEM and barycentric FEM hold great promises in the field of computational mechanics. Some potential extensions of the current work are suggested below.

6.2.1 VEM formulations for other computational mechanics problems

Throughout this thesis, we focus on the development of VEM formulations in linear and finite elasticity problems considering isotropic materials. In those problems, we have demonstrated that the VEM framework works for more general discretizations, enables more flexible mesh adaptivity, and better models the large and heterogeneous deformation fields.

The VEM formulation developed in this thesis for finite elasticity problems considers only lower-order elements. In the future, one could extend this formulation to higher-order virtual elements. Moreover, several extensions of the VEM formulations developed in this thesis to other computational mechanics problems are of interest. For instance, one could incorporate material anisotropy into the VEM formulations, and to apply the formulations to plasticity, strain gradient elasticity and plasticity problems, as well as problems involving localization and discontinuities, such as fracture mechanics.

Incorporating material anisotropy Several directions are worthwhile exploring to incorporate material anisotropy (e.g. fiber reinforcements) into the VEM framework for finite elasticity presented in Chapter 3. First, the stabilization scheme used in Chapter 3 is derived under the assumption that materials are isotropic. The basic idea is to construct the stabilization energy using a scalar-valued function (i.e. α_E) of the invariants of the right Cauchy-Green deformation tensor. Thus, the stabilization energy is also isotropic and does not have preferred directions. To account for material anisotropy, one could employ the representation theorem for isotropic functions [207], [208] and define the stabilization energy and α_E as a scalar-valued function of the irreducible integrity basis associated with the anisotropic materials. Second, the local VEM space is defined based on the Laplacian operator, which is an isotropic differential operator. One could explore encoding the material anisotropy into the local VEM space using anisotropic differential operators.

Strain gradient elasticity and plasticity The classical elasticity and plasticity theories fail to explain the size effects in materials because they do not include any parameter related to material length scales. To overcome this limitation, strain gradient elasticity and plasticity theories, which includes higher-order derivatives of the displacement field, are proposed, see, e.g., [209]–[217]. The numerical solution of boundary value problems using strain gradient theories is a challenging task because the underlying partial differential equations are fourth order, which requires C^1 interpolations [218], [219]. To avoid this, mixed finite element approximations are developed as an alternative [220], which treats both displacement and strain fields as independent variables. This is undesirable from an efficiency perspective because it introduces additional DOFs and Lagrange multipliers.

The VEM can naturally handle numerical approximations of arbitrary continuity and element geometry. Thus, it holds great potential in the numerical solution of boundary value problems governed by strain gradient elasticity and plasticity theories. Interesting future work along this direction includes the development C^1 VEM formulations [25], [48], [221] for strain gradient elasticity and plasticity problems on general polygonal and polyhedral discretizations, and comparing the performance of the C^1 displacement-based VEM approximations with the typically adopted mixed approximations.

Local enrichment in VEM for problems involving discontinuities Problems with discontinuities and localized features in their solutions are common in the field of computational mechanics. A representative example is fracture in solids. For such problems, the Extended/Generalized Finite Element Method (XFEM/GFEM) [222]–[228] is a popular method which is capable of capturing the discontinuities without local mesh refinement or remeshing. The basic idea of XFEM/GFEM is to enrich the local function space with discontinuity (e.g. Heaviside function) so that the discontinuity in solutions can be captured even with structured and regular discretization. However, with the enrichment, the numerical integration at the element level becomes more difficult, and advanced numerical integration schemes are needed even for quadrilateral elements [107], [229]. The VEM decomposes the approximation at the element-level into two parts: a consistent term that preserves suitable polynomial accuracy and a stability term that ensures the stability of the approximation. By doing so, numerical integration can be simplified and only integrations of polynomial functions instead of complex ones are needed. Thus, one could integrate the VEM methodology with the enrichment concept in XFEM/GFEM so that the resulting framework preserves the nice features of both methods and overcomes their respective limitations. Some preliminary investigations of along this direction are presented in [230]. Moreover, since the local function space in VEM is defined implicitly through a suitable set of PDE, its definition can be quite flexible. Thus, another area of future research includes exploring ways to embed discontinuity into the definition and governing PDEs of the local VEM space.

6.2.2 Recovery-based error estimator for finite elasticity problems

Although proposed for linear elasticity problems, the recovery-based error estimator can be naturally extended for any nonlinear problems with H^1 conforming approximations, such as the finite elasticity problems. However, more numerical investigations in different types of boundary value problems are needed to evaluate the accuracy of that error estimator for finite elasticity problems.

6.2.3 Novel adaptive mesh refinement and coarsening schemes using VEM

Chapter 4 shows that the proposed recovery-based errors estimator provides an effective indicator to drive the adaptive mesh refinement in 2D linear elasticity problems. Future extensions of this research includes developing novel and efficient mesh adaption strategies, which take full advantage of the geometric flexibility of polygonal and polyhedral elements to achieve adaptivity (i.e. refinement, coarsening and re-meshing) with only local modifications. The adaptive coarsening is particularly interesting. For example, with the capability of VEM in handling any concave element geometry, one can simply agglomerate all the elements with a relatively small level of errors together to form a coarsened element without any local remeshing. In addition, to effectively deal with adaptive refinement and coarsening on polygonal and polyhedral meshes, especially in 3D and for large-scale simulations, efficient mesh data structure is essential. The Topological Data Structure (TopS) [231], which is a compact adjacency-based data structure, potentially provides an efficient platform for adaptivity on polygonal and polyhedral meshes. Thus, extending the TopS to include polygonal and polyhedral mesh representations serves as a promising area of future research. Another promising area of research consists of applying adaptivity refinement and coarsening with polygonal and polyhedral elements to structural dynamics and nonlinear fracture mechanics problems [171].

6.2.4 Incorporating manufacturing process and requirements into the VEM-based topology optimization

It is crucial to take into account in topology optimization whether the final designs can be manufactured and how they are going to be fabricated. Thus, one potential extension of the VEM-based topology optimization framework developed in this thesis includes the incorporation of manufacturing constraints [232] (e.g. overhanging angle [233], minimum [234] and maximum [235] length scales, hole size [236], and geometric complexity [237]) and manufacturing process (e.g. printing direction and the dependence of material property on printing direction).

6.2.5 Adaptive topology optimization using VEM

The topology optimization is an iterative process which involves redistribution of material densities. With the mesh adaptivity, the efficiency of topology optimization can be significantly improved, see e.g. [66]. However, most work in topology optimization only considers adaptivity refinement, whereas the coarsening aspect of the adaptivity receives relatively less attention. Being able to handle arbitrary concave elements, the VEM offers an attractive path to develop novel adaptive refinement and coarsening schemes in topology optimization. Thus, the potential research includes developing tailored mesh adaption, especially coarsening, strategies for the VEM-based topology optimization framework introduced in Chapter 5 by exploiting the advantages of the geometric flexibility of virtual elements.

Appendices

APPENDIX A
A TRACE-BASED STABILIZATION PARAMETER α_E FOR VEM UNDER FINITE DEFORMATIONS

We present a detailed derivation of the trace-based stabilization parameter α_E in (3.61) and (3.62). Considering a general function $\Phi(I_1, I_2, J)$, we have α_E be expanded as

$$
\begin{aligned}
\alpha_E = \frac{1}{d^2}\mathrm{tr}\left(\frac{\partial^2\Phi}{\partial\mathbf{F}\partial\mathbf{F}}\right) = \frac{1}{d^2}\mathrm{tr}\Bigg[&\frac{\partial^2\Phi}{\partial I_1\partial I_1}\frac{\partial I_1}{\partial\mathbf{F}}:\frac{\partial I_1}{\partial\mathbf{F}} + \frac{\partial^2\Phi}{\partial I_2\partial I_2}\frac{\partial I_2}{\partial\mathbf{F}}:\frac{\partial I_2}{\partial\mathbf{F}} + \frac{\partial^2\Phi}{\partial J\partial J}\frac{\partial J}{\partial\mathbf{F}}:\frac{\partial J}{\partial\mathbf{F}} \\
&+2\frac{\partial^2\Phi}{\partial I_1\partial I_2}\frac{\partial I_1}{\partial\mathbf{F}}:\frac{\partial I_2}{\partial\mathbf{F}} + 2\frac{\partial^2\Phi}{\partial I_1\partial J}\frac{\partial I_1}{\partial\mathbf{F}}:\frac{\partial J}{\partial\mathbf{F}} + 2\frac{\partial^2\Phi}{\partial I_2\partial J}\frac{\partial I_2}{\partial\mathbf{F}}:\frac{\partial J}{\partial\mathbf{F}} + \frac{\partial\Phi}{\partial I_1}\mathrm{tr}\left(\frac{\partial^2 I_1}{\partial\mathbf{F}\partial\mathbf{F}}\right) \\
&+\frac{\partial\Phi}{\partial I_2}\mathrm{tr}\left(\frac{\partial^2 I_2}{\partial\mathbf{F}\partial\mathbf{F}}\right) + \frac{\partial\Phi}{\partial J}\mathrm{tr}\left(\frac{\partial^2 J}{\partial\mathbf{F}\partial\mathbf{F}}\right)\Bigg],
\end{aligned}
\tag{A.1}
$$

where the dependences of α_E, Φ and its derivatives on I_1, I_2 and J are assumed. In the above equation, each component can be explicitly expressed as

$$
\begin{aligned}
&\frac{\partial I_1}{\partial\mathbf{F}}:\frac{\partial I_1}{\partial\mathbf{F}} = 4I_1, \quad \frac{\partial I_2}{\partial\mathbf{F}}:\frac{\partial I_2}{\partial\mathbf{F}} = 4\left[2I_1I_2 - I_1\mathrm{tr}\left(\mathbf{C}^2\right) + \mathrm{tr}\left(\mathbf{C}^3\right)\right], \quad \frac{\partial J}{\partial\mathbf{F}}:\frac{\partial J}{\partial\mathbf{F}} = J^2\mathrm{tr}\left(\mathbf{C}^{-1}\right). \\
&\frac{\partial I_1}{\partial\mathbf{F}}:\frac{\partial I_2}{\partial\mathbf{F}} = 4\left[I_1^2 - \mathrm{tr}\left(\mathbf{C}^2\right)\right] = 8I_2, \quad \frac{\partial I_1}{\partial\mathbf{F}}:\frac{\partial J}{\partial\mathbf{F}} = 4dJ, \quad \frac{\partial I_2}{\partial\mathbf{F}}:\frac{\partial J}{\partial\mathbf{F}} = 2\left(d-1\right)JI_1, \\
&\mathrm{tr}\left(\frac{\partial^2 I_1}{\partial\mathbf{F}\partial\mathbf{F}}\right) = 2d^2, \quad \mathrm{tr}\left(\frac{\partial^2 I_2}{\partial\mathbf{F}\partial\mathbf{F}}\right) = 2\left(d-1\right)^2 I_1, \quad \mathrm{tr}\left(\frac{\partial^2 J}{\partial\mathbf{F}\partial\mathbf{F}}\right) = 0.
\end{aligned}
\tag{A.2}
$$

In the steps that follow, the Cayley-Hamilton theorem (see, for example, [238]) is adopted to further simplify the following two terms

$$
\frac{\partial I_2}{\partial\mathbf{F}}:\frac{\partial I_2}{\partial\mathbf{F}} = 4\left[2I_1I_2 - I_1\mathrm{tr}\left(\mathbf{C}^2\right) + \mathrm{tr}\left(\mathbf{C}^3\right)\right], \text{ and } \frac{\partial J}{\partial\mathbf{F}}:\frac{\partial J}{\partial\mathbf{F}} = J^2\mathrm{tr}\left(\mathbf{C}^{-1}\right). \tag{A.3}
$$

In 2D, the Cayley-Hamilton theorem [238] states that $\mathbf{C} \in \mathbb{R}^{2\times 2}$ satisfies the following characteristic equation

$$\mathbf{C}^2 - I_1\mathbf{C} + \det(\mathbf{C})\mathbf{I} = \mathbf{0}. \tag{A.4}$$

The following relation can be extracted from the above characteristic equation:

$$\operatorname{tr}\left(\mathbf{C}^3\right) - I_1 \operatorname{tr}\left(\mathbf{C}^2\right) = J^2 I_1 \text{ and } J^2 \operatorname{tr}\left(\mathbf{C}^{-1}\right) = I_1, \tag{A.5}$$

and consequently,

$$\frac{\partial I_2}{\partial \mathbf{F}} : \frac{\partial I_2}{\partial \mathbf{F}} = \left(8 I_1 I_2 - 4 J^2 I_1\right) \text{ and } \frac{\partial J}{\partial \mathbf{F}} : \frac{\partial J}{\partial \mathbf{F}} = I_1. \tag{A.6}$$

Combining (A.1), (A.2) and (A.6), we obtain the final expression for α_E in the 2D case

$$\begin{aligned}\alpha_E\left(I_1, I_2, J\right) = \frac{1}{4}\Bigg[& 4 I_1 \frac{\partial^2 \Phi}{\partial I_1 \partial I_1} + \left(8 I_1 I_2 - 4 J^2 I_1\right) \frac{\partial^2 \Phi}{\partial I_2 \partial I_2} + I_1 \frac{\partial^2 \Phi}{\partial J \partial J} + 16 I_2 \frac{\partial^2 \Phi}{\partial I_1 \partial I_2} \\ & + 8 J \frac{\partial^2 \Phi}{\partial I_1 \partial J} + 2 J I_1 \frac{\partial^2 \Phi}{\partial I_2 \partial J} + 8 \frac{\partial \Phi}{\partial I_1} + 2 I_1 \frac{\partial \Phi}{\partial I_2}\Bigg]. \end{aligned} \tag{A.7}$$

Similarly, the Cayley-Hamilton [238] in the 3D case states that $\mathbf{C} \in \mathbb{R}^{3\times 3}$ satisfies the following characteristic equation

$$\mathbf{C}^3 - I_1 \mathbf{C}^2 + I_2 \mathbf{C} - \det(\mathbf{C})\mathbf{I} = \mathbf{0}. \tag{A.8}$$

As a result, the following expressions can be obtained

$$\operatorname{tr}\left(\mathbf{C}^3\right) - I_1 \operatorname{tr}\left(\mathbf{C}^2\right) = 3J^2 - I_1 I_2 \text{ and } J^2 \operatorname{tr}\left(\mathbf{C}^{-1}\right) = I_2, \tag{A.9}$$

and, thus,

$$\frac{\partial I_2}{\partial \mathbf{F}} : \frac{\partial I_2}{\partial \mathbf{F}} = \left(4 I_1 I_2 + 12 J^2 I_1\right) \text{ and } \frac{\partial J}{\partial \mathbf{F}} : \frac{\partial J}{\partial \mathbf{F}} = I_2. \tag{A.10}$$

By combining (A.1), (A.2) and (A.10), we arrive at the final expression for α_E for the

3D case

$$\alpha_E\left(I_1, I_2, J\right) = \frac{1}{9}\left[4I_1\frac{\partial^2\Phi}{\partial I_1\partial I_1} + \left(4I_1I_2 + 12J^2\right)\frac{\partial^2\Phi}{\partial I_2\partial I_2} + I_2\frac{\partial^2\Phi}{\partial J\partial J} + 16I_2\frac{\partial^2\Phi}{\partial I_1\partial I_2}\right.$$
$$\left. + 12J\frac{\partial^2\Phi}{\partial I_1\partial J} + 4JI_1\frac{\partial^2\Phi}{\partial I_2\partial J} + 18\frac{\partial\Phi}{\partial I_1} + 8I_1\frac{\partial\Phi}{\partial I_2}\right]. \quad \text{(A.11)}$$

APPENDIX B

THE PROOF OF RELATIONS (3.80) – (3.83)

We provide detailed derivations of relations (3.80)–(3.83) in Section 4.7 of Chapter 3 when the polyhedral element P contains non star-convex faces. First, we are able to show that the utilized vertex-based quadrature rule on each face F can integrate any linear functions exactly on F, even if F is not star-convex. Given any linear functions $\mathbf{p}_1$ in $[\mathscr{P}_1(P)]^3$, we can express it as $\mathbf{p}_1 = \mathbf{a} + \mathbf{B}\mathbf{X}$, where $\mathbf{a}$ and $\mathbf{B}$ are an arbitrary constant vector and a second order tensor, respectively. The exact face integral $\int_F \mathbf{p}_1 \otimes \mathbf{n}^F \mathrm{d}S$ is obtained as

$$\int_F \mathbf{p}_1 \otimes \mathbf{n}^F \mathrm{d}S = \int_F (\mathbf{a} + \mathbf{B}\mathbf{X}) \otimes \mathbf{n}^F \mathrm{d}S = |F| \left(\mathbf{a} + \mathbf{B}\mathbf{X}_c^F\right) \otimes \mathbf{n}^F, \tag{B.1}$$

where $\mathbf{X}_c^F$ is the centroid of face F. Applying the vertex based rule defined in (3.23) and (3.25) to the function $\mathbf{v} = \mathbf{p}_1$, we obtain

$$\begin{aligned}\int_f \mathbf{p}_1 \otimes \mathbf{n}^F \mathrm{d}S &= \sum_{j=1}^{m^F} \frac{|T_j^F|}{3} \left[\mathbf{p}_1\left(\mathbf{X}_j^F\right) + \mathbf{p}_1\left(\mathbf{X}_{j+1}^F\right) + \mathbf{p}_1\left(\mathbf{X}_s^F\right)\right] \otimes \mathbf{n}^F \\ &= \sum_{j=1}^{m^F} \frac{|T_j^F|}{3} \left[3\mathbf{a} + \mathbf{B}\left(\mathbf{X}_j^F + \mathbf{X}_{j+1}^F + \mathbf{X}_s^F\right)\right] \otimes \mathbf{n}^F \\ &= \left[\sum_{j=1}^{m^F} |T_j^F| \mathbf{a} + \mathbf{B}\left(\sum_{j=1}^{m^F} |T_j^F| \frac{\mathbf{X}_j^F + \mathbf{X}_{j+1}^F + \mathbf{X}_s^F}{3}\right)\right] \otimes \mathbf{n}^F.\end{aligned} \tag{B.2}$$

Moreover, if we recall the definition of the signed area $|T_j^F|$ defined in (3.24), then we observe that for non star-convex faces

$$\sum_{j=1}^{m^F} |T_j^F| = |F|\,, \qquad \sum_{j=1}^{m^F} |T_j^F| \frac{\mathbf{X}_j^F + \mathbf{X}_{j+1}^F + \mathbf{X}_s^F}{3} = |F|\mathbf{X}_c^F, \tag{B.3}$$

and thus

$$\left[\sum_{j=1}^{m^F} |T_j^F|\mathbf{a} + \mathbf{B}\left(\sum_{j=1}^{m^F} |T_j^F|\frac{\mathbf{X}_j^F + \mathbf{X}_{j+1}^F + \mathbf{X}_s^F}{3}\right)\right] \otimes \mathbf{n}^F = |F|\left(\mathbf{a} + \mathbf{B}\mathbf{X}_c^F\right) \otimes \mathbf{n}^F = \int_F \mathbf{p}_1 \otimes \mathbf{n}^F \, \mathrm{d}S. \tag{B.4}$$

Notice that the above analysis also holds if F is not star-convex, implying that the vertex based quadrature rule is able to integrate any linear functions exactly even in those cases. This implies that (3.80) holds even if P contains non star-shaped faces.

Second, if the displacement field is linear, i.e. $\mathbf{u} = \mathbf{p}_1$, each deformed face $\widetilde{F}$ remains planar and the outward normal vector $\widetilde{\mathbf{n}}^F$ in the deformed configuration stays constant over each $\widetilde{F}$. Since the proofs of (3.81) and (3.82) for the “Simple” approach are more straightforward, we hereby prove the two expressions for the “Robust” approach. By applying $\mathbf{u} = \mathbf{p}_1$ to (3.40) and using (B.3), we obtain

$$\begin{aligned}
\frac{1}{6|P|}\sum_{\widetilde{F}}\sum_{j=1}^{m^F}\left\{\widetilde{\mathbf{X}}_s^F \cdot \widetilde{\mathbf{X}}_j^F \wedge \widetilde{\mathbf{X}}_{j+1}^F\right\} &= \frac{1}{3|P|}\sum_{\widetilde{F}}\sum_{j=1}^{m^F}\left(|\widetilde{T}_j^F|\frac{\widetilde{\mathbf{X}}_j^F + \widetilde{\mathbf{X}}_{j+1}^F + \widetilde{\mathbf{X}}_s^F}{3}\right) \cdot \widetilde{\mathbf{n}}^F \\
&= \frac{1}{3|P|}\sum_{\widetilde{F}}\sum_{j=1}^{m^F} |\widetilde{F}|\widetilde{\mathbf{X}}_c^F \cdot \widetilde{\mathbf{n}}^F \\
&= \frac{1}{3|P|}\sum_{\widetilde{F}}\int_{\widetilde{F}} \widetilde{\mathbf{X}} \cdot \widetilde{\mathbf{n}}^F \, \mathrm{d}S = J_P(\mathbf{p}_1),
\end{aligned} \tag{B.5}$$

where $|\widetilde{F}|$ is the (absolute) area of deformed face $\widetilde{F}$, and $\widetilde{\mathbf{X}}_c^F$ stands for the centroid of the deformed face $\widetilde{F}$. Again, no assumption is made whether F is star-convex with respect to $\mathbf{X}_s^F$, which proves relation (3.81) in Section 4.7 of Chapter 3.

Additionally, the first variation of $J_P(\mathbf{p}_1)$ with respect to any $\delta\mathbf{v}$ obtained in (3.41) can be recast as

$$\begin{aligned}
DJ_P\left(\mathbf{u}\right)\cdot\delta\mathbf{v} &= \frac{1}{6|P|}\sum_{i=1}^{m}\left\{\delta\mathbf{v}\left(\mathbf{X}_i\right)\cdot\left[\sum_{\widetilde{F}\in\widetilde{\mathscr{F}}^i}\left(\beta_i^F\sum_{j=1}^{m_F}\widetilde{\mathbf{X}}_{j-1}^F\wedge\widetilde{\mathbf{X}}_j^F+\widetilde{\mathbf{X}}_s^F\wedge\left(\widetilde{\mathbf{X}}_{\mathscr{G}_F(i)-1}^F-\widetilde{\mathbf{X}}_{\mathscr{G}_F(i)+1}^F\right)\right)\right]\right\} \\
&= \frac{1}{6|P|}\sum_{\widetilde{F}\in\partial\widetilde{P}}\sum_{i=1}^{m^F}\delta\mathbf{v}\left(\mathbf{X}_i^f\right)\cdot\left[\beta^F\sum_{j=1}^{m^F}\widetilde{\mathbf{X}}_{j-1}^F\wedge\widetilde{\mathbf{X}}_j^F+\widetilde{\mathbf{X}}_s^F\wedge\left(\widetilde{\mathbf{X}}_{i-1}^F-\widetilde{\mathbf{X}}_{i-1}^F\right)\right] \\
&= \frac{1}{3|P|}\sum_{F\in\partial P}\sum_{i=1}^{m^F}\delta\mathbf{v}\left(\mathbf{X}_i^F\right)\cdot\left(\beta^F|\widetilde{F}|+|\widetilde{T}_j^F|+|\widetilde{T}_{j-1}^F|\right)\widetilde{\mathbf{n}}^F.
\end{aligned} \tag{B.6}$$

Pushing back the above expression to the undeformed configuration using the Nanson's formula [80] yields

$$\begin{aligned}
DJ_E\left(\mathbf{u}\right)\cdot\delta\mathbf{v} &= \frac{1}{3|P|}\sum_{F\in\partial P}\sum_{i=1}^{m^F}\delta\mathbf{v}\left(\mathbf{X}_i^F\right)\cdot\left(\beta^F|\widetilde{F}|+|\widetilde{T}_j^F|+|\widetilde{T}_{j-1}^F|\right)\widetilde{\mathbf{n}}^F \\
&= \frac{1}{3|P|}\sum_{F\in\partial P}\sum_{i=1}^{m^F}\delta\mathbf{v}\left(\mathbf{X}_i^F\right)\cdot\left[\left(\beta^F|F|+|T_j^F|+|T_{j-1}^F|\right)\frac{\partial J}{\partial\mathbf{F}}\left(\mathbf{F}(\mathbf{p}_1)\right)\mathbf{n}^F\right] \\
&= \frac{\partial J}{\partial\mathbf{F}}\left(\mathbf{F}(\mathbf{p}_1)\right):\sum_{F\in\partial P}\sum_{i=1}^{m^F}w_i^F\,\delta\mathbf{v}\left(\mathbf{X}_i^F\right)\otimes\mathbf{n}^F \\
&= |P|\frac{\partial J}{\partial\mathbf{F}}\left(\mathbf{F}(\mathbf{p}_1)\right):\Pi_0^E\left(\nabla(\delta\mathbf{v})\right),
\end{aligned} \tag{B.7}$$

and therefore we prove (3.82).

Finally, relation (3.83) in Section 4.7 of Chapter 3 can be derived as follows:

$$
\begin{aligned}
&G_h^{\mathbf{v}}(\mathbf{p}_1, \widehat{p}_0, \delta\mathbf{v}_h) \\
&= \sum_P |P| \left[\frac{\partial \Psi}{\partial \mathbf{F}} (\mathbf{F}(\mathbf{p}_1)) : \Pi_E^0 \nabla(\delta\mathbf{v}_h) + \widehat{q}_0|_E \, DJ_E(\mathbf{p}_1) \cdot \delta\mathbf{v}_h|_E \right] - \langle \mathbf{t}, \delta\mathbf{v}_h \rangle_h \\
&= \left[\frac{\partial \Psi}{\partial \mathbf{F}} (\mathbf{F}(\mathbf{p}_1)) + \frac{\partial U}{\partial J} (\mathbf{p}_1) \frac{\partial J}{\partial \mathbf{F}} (\mathbf{F}(\mathbf{p}_1)) \right] : \sum_P |P| \Pi_E^0 \nabla(\delta\mathbf{v}_h) - \sum_{F \in \Gamma_h^{\mathbf{t}}} \sum_{v \in F} w_v^F \, \mathbf{t}(\mathbf{X}_u) \cdot \delta\mathbf{v}_h(\mathbf{X}_v) \\
&= \mathbf{P} : \sum_E \sum_{F \in \partial P} \sum_{j=1}^{m^F} \left[w_j^F \delta\mathbf{v}_h \left(\mathbf{X}_j^F \right) \otimes \mathbf{n}^F \right] - \sum_{F \in \Gamma_h^{\mathbf{t}}} \sum_{j=1}^{m^F} w_j^F \mathbf{P}\mathbf{n}^F \cdot \delta\mathbf{v}_h \left(\mathbf{X}_j^F \right) \\
&= \mathbf{P} : \sum_{F \in \Gamma_h^{\mathbf{t}}} \sum_{j=1}^{m^F} \left[w_j^F \delta\mathbf{v}_h \left(\mathbf{X}_j^F \right) \otimes \mathbf{n}^F \right] - \sum_{F \in \Gamma_h^{\mathbf{t}}} \sum_{j=1}^{m^F} w_j^F \mathbf{P}\mathbf{n}^F \cdot \delta\mathbf{v}_h \left(\mathbf{X}_j^F \right) = 0 \quad \forall \delta\mathbf{v}_h \in \mathscr{K}_h^0.
\end{aligned}
\tag{B.8}
$$

APPENDIX C

AN SPR-TYPE RECOVERY SCHEME OF LINEAR VEM

To complement the discussions in Section 5 of Chapter 4, this appendix presents an SPR-type scheme, which is conceptually similar to the SPR in the FEM literature [147], [148], [156], for 2D and 3D linear VEM. We will particularly focus on the reconstructing displacement gradient at the vertices of the mesh. A similar scheme is used in [239] to recover the stress at the vertices of the mesh for 2D linear elastic fracture mechanics problems. We note that, unlike the gradient recovery scheme proposed in Section 4 of Chapter 4, the extension of this SPR-type scheme to higher order VEMs is not straightforward.

Following the notations introduced in Section 4 of Chapter 4 and for a given patch ω_i, the basic idea of this SPR-type recovery scheme is to seek a linear tensorial field, denoted by $\mathbf{g}^{\omega_i}(\mathbf{X}) \in [\mathscr{P}_1(\omega_i)]^{d\times d}$, such that

$$\mathbf{g}^{\omega_i} = \underset{\boldsymbol{\xi}\in[\mathscr{P}_1(\omega_i)]^{d\times d}}{\operatorname{argmin}} \sum_{j=1}^{N^E_{\omega_i}} \left[\boldsymbol{\xi}(\mathbf{X}_c^{E_j}) - \mathbf{\Pi}_0^0(\nabla\mathbf{u}_h)|_{E_j}\right] : \left[\boldsymbol{\xi}(\mathbf{X}_c^{E_j}) - \mathbf{\Pi}_0^0(\nabla\mathbf{u}_h)|_{E_j}\right], \tag{C.1}$$

where $N^E_{\omega_i}$ is the total number of elements in patch ω_i, E_j is the jth element in ω_i with its centroid being $\mathbf{X}_c^{E_j}$, and we recall from (4.23) that $\mathbf{\Pi}_0^0$ is the L^2 projection of the displacement gradient onto constant functions. With $\mathbf{g}^{\omega_i}$ obtained, the value of the reconstructed gradient $G_h\mathbf{u}_h$ at $\mathbf{X}_i$ is given by

$$G_h\mathbf{u}_h(\mathbf{X}_i) = \mathbf{g}^{\omega_i}(\mathbf{X}_i). \tag{C.2}$$

Having computed $G_h\mathbf{u}_h$ at every vertex of the mesh, the remaining steps of the SPR-type scheme are identical to the ones outlined in Section 4.4.1 of Chapter 4.

Finally, we remark that the rules for choosing patches in this SPR-type scheme are the same as the ones described in Section 4.4.2 of Chapter 4 except for the threshold on the number of elements which decides whether to enlarge the patch. In this SPR-type scheme, a necessary condition for the recovery (C.1) to have a unique solution is $N^E_{\omega_i} \geq 3$ in 2D and $N^E_{\omega_i} \geq 4$ in 3D. Thus, the threshold for the SPR-type scheme is set to be 3 in 2D and 4 in 3D. For any patch that contains less number of elements than the threshold, the patch will be enlarged in the recovery process.

APPENDIX D

POLYTOP3D: AN EFFICIENT MATLAB IMPLEMENTATION OF THE PROPOSED VEM-BASED TOPOLOGY OPTIMIZATION FRAMEWORK

An implementation of the proposed VEM-based topology optimization framework into a modular Matlab code named `PolyTop3D`, which can handle any non-Cartesian design domains specified by the users on general polyhedral discretizations (both structured and unstructured), is developed in this thesis. The `PolyTop3D` is modularized in a similar manner to the `PolyTop` code presented in [110] together with a similar naming convention for its variables. Thus, we refer the readers to [110] for a thorough introduction of the structure of the code. We hope that the modularity and flexibility offered by `PolyTop3D` will motivate the community to explore the proposed VEM-based framework in other topology optimization problems.

In the sequel, we demonstrate the efficiency of the code `PolyTop3D` by benchmarking it with the `Top3D` code by Liu and Tovar [196]. For comparison purpose, the cantilever example presented in Table 4 of [196] is solved on a set of three regular hexahedral meshes whose statistics are shown in Table D.1. Each element in those meshes is a unit cube. Throughout this study, the filter radius is set as $R = 1.5$ and the volume constraint is taken to be $\overline{V} = 15\%$. For both codes, a constant penalty parameter of $p = 3$ is used and 200 optimization iterations are carried out on a desktop computer with an Intel(R) Xeon(R), 3.00 GHz processor and 256 GB of RAM running Matlab R2016a. For all the meshes, the two codes produce almost identical final topologies and are not shown here for the sake of conciseness.

In Table D.2, we present a comparison of the total runtimes of `PolyTop3D` and `Top3D` for the three meshes. In addition, Table D.3 shows the breakdown of the total runtime of the `PolyTop3D` code into major steps. One immediate conclusion from Tables D.2 and

D.3 is that the PolyTop3D code is able to achieve similar efficiency to the Top3D code using more than four times of DVs. The major runtime difference of the two codes comes from the steps of forming projection matrices, $\mathbf{P}^{\mathscr{F}}$ and $\mathbf{P}^{\mathscr{V}}$ (c.f. Eqs. (5.50) and (5.54)), and VEM shape functions φ_i.

Table D.1: Statistics of three meshes for the cantilever problem

Meshes	Dimensions	Node #	EL. #	DV # (PolyTop3D)	DV # (Top3D)
Mesh 1	$48 \times 16 \times 12$	$10,829$	$9,216$	$41,625$	$9,216$
Mesh 2	$72 \times 24 \times 18$	$34,675$	$31,104$	$135,013$	$31,104$
Mesh 3	$96 \times 32 \times 24$	$80,025$	$73,728$	$313,649$	$73,728$

Table D.2: Total runtime (sec.) comparison of PolyTop3D with the Top3D code. The times are reported in seconds for 200 optimization iterations.

	Mesh 1	Mesh 2	Mesh 3
PolyTop3D	340.48	2275.20	13517.09
TOP3D	296.22	2085.20	12797.00

Table D.3: Breakdown of the PolyTop3D runtime (sec.) fro 200 optimization iterations. The times are in seconds with the percentage of total runtime provided in parentheses.

	Mesh 1	Mesh 2	Mesh 3
Forming $\mathbf{P}^{\mathscr{F}}$ and $\mathbf{P}^{\mathscr{V}}$	7.66 (2.25%)	144.02 (6.33%)	766.34 (5.67%)
Forming VEM shape func.	23.31 (6.85%)	82.53 (3.63%)	199.47 (1.48%)
Assemble $\mathbf{K}_\rho$	124.13 (36.46%)	434.55 (19.10%)	1174.41 (8.69%)
Solving $\mathbf{K}_\rho \mathbf{U} = \mathbf{F}$	156.20 (45.88%)	1508.18 (66.29%)	11097.56 (82.10%)
Compliance sensitivity	22.73 (6.68%)	79.034 (3.47%)	210.53 (1.56%)
OC update	2.15 (0.63%)	12.61 (0.55%)	31.46 (0.23%)

REFERENCES

[1] H. G. Kim and D. Sohn, “A new finite element approach for solving three-dimensional problems using trimmed hexahedral elements,” *International Journal for Numerical Methods in Engineering*, vol. 102, no. 9, pp. 1527–1553, 2015.

[2] H. Chi, C. Talischi, O. Lopez-Pamies, and G. H. Paulino, “Polygonal finite elements for finite elasticity,” *International Journal for Numerical Methods in Engineering*, vol. 101, pp. 305–328, 2015.

[3] O. J. Sutton, “The virtual element method in 50 lines of Matlab,” *Numerical Algorithms*, vol. 75, no. 4, pp. 1141–1159, 2017.

[4] C. Talischi, A. Pereira, G. H. Paulino, I. F. M. Menezes, and M. S. Carvalho, “Polygonal finite elements for incompressible fluid flow,” *International Journal for Numerical Methods in Fluids*, vol. 74, pp. 134–151, 2014.

[5] R. D. Cook, D. S. Malkus, M. E. Plesha, and R. J. Witt, *Concepts and Applications of Finite Element Analysis, 4th Edition*. John Wiley & Sons, Ltd., 2002.

[6] S. Ghosh and S. Mukhopadhyay, “A two-dimensional automatic mesh generator for finite element analysis for random composites,” *Computers & Structures*, vol. 41, no. 2, pp. 245–256, 1991.

[7] S. Ghosh and S. N. Mukhopadhyay, “A material based finite element analysis of heterogeneous media involving Dirichlet tessellations,” *Computer Methods in Applied Mechanics and Engineering*, vol. 104, no. 2, pp. 211–247, 1993.

[8] S. Ghosh and R. Mallett, “Voronoi cell finite elements,” *Computers & Structures*, vol. 50, no. 1, pp. 33–46, 1994.

[9] S. Ghosh, *Micromechanical analysis and multi-scale modeling using the Voronoi cell finite element method*. CRC Press, 2011.

[10] D. N. Arnold, F. Brezzi, B. Cockburn, and L. D. Marini, “Unified analysis of discontinuous Galerkin methods for elliptic problems,” *SIAM Journal on Numerical Analysis*, vol. 39, no. 5, pp. 1749–1779, 2002.

[11] E. H. Cangiani A.and Georgoulis and P. Houston, “Hp-version discontinuous Galerkin methods on polygonal and polyhedral meshes,” *Mathematical Models and Methods in Applied Sciences*, vol. 24, no. 10, pp. 2009–2041, 2014.

[12] K. Lipnikov, G. Manzini, and M. Shashkov, “Mimetic finite difference method,” *Journal of Computational Physics*, vol. 257, pp. 1163–1227, 2014.

[13] J. E. Bishop, “Simulating the pervasive fracture of materials and structures using randomly close packed Voronoi tessellations.,” *Computational Mechanics*, vol. 44, no. 4, pp. 455–471, 2009.

[14] N. Sukumar and J. E. Bolander, “Voronoi-based interpolants for fracture modelling,” in *Tessellations in the Sciences: Virtues, Techniques and Applications of Geometric Tilings*, Springer Verlag, 2009.

[15] D. W. Spring, S. E. Leon, and G. H. Paulino, “Unstructured polygonal meshes with adaptive refinement for the numerical simulation of dynamic cohesive fracture,” *International Journal of Fracture*, vol. 189, no. 1, pp. 33–57, 2014.

[16] S. E. Leon, D. W. Spring, and G. H. Paulino, “Reduction in mesh bias for dynamic fracture using adaptive splitting of polygonal finite elements,” *International Journal for Numerical Methods in Engineering*, vol. 100, no. 8, pp. 555–576, 2014.

[17] C. Talischi, G. H. Paulino, A. Pereira, and I. F. M. Menezes, “PolyTop: A Matlab implementation of a general topology optimization framework using unstructured polygonal finite element meshes,” *Structural and Multidisciplinary Optimization*, vol. 45, no. 3, pp. 329–357, 2012.

[18] C. Talischi, G. H. Paulino, and C. H. Le, “Honeycomb Wachspress finite elements for structural topology optimization,” *Structural and Multidisciplinary Optimization*, vol. 37, no. 6, pp. 569–583, 2009.

[19] C. Talischi, G. H. Paulino, A. Pereira, and I. F. M. Menezes, “Polygonal finite elements for topology optimization: A unifying paradigm.,” *International Journal for Numerical Methods in Engineering*, vol. 82, pp. 671–698, 2010.

[20] S. O. R. Biabanaki, A. R. Khoei, and P. Wriggers, “Polygonal finite element methods for contact-impact problems on non-conformal meshes,” *Computer Methods in Applied Mechanics and Engineering*, vol. 269, pp. 198–221, 2014.

[21] S. O. R. Biabanaki and A. R. Khoei, “A polygonal finite element method for modeling arbitrary interfaces in large deformation problems,” *Computational Mechanics*, vol. 50, no. 1, pp. 19–33, 2012.

[22] H. Chi, C. Talischi, O. Lopez-Pamies, and G. H. Paulino, “A paradigm for higher-order polygonal elements in finite elasticity,” *Computational Methods in Applied Mechanics and Engineering*, vol. 306, pp. 216–251, 2016.

[23] C. Talischi and G. H. Paulino, "Addressing integration error for polygonal finite elements through polynomial projections: A patch test connection," *Mathematical Models and Methods in Applied Sciences*, vol. 24, no. 08, pp. 1701–1727, 2014.

[24] G.Manzini, A.Russo, and N.Sukumar, "New perspective on polygonal and polyhedral finite element method," *Mathematical Models and Methods in Applied Sciences*, vol. 24, no. 08, pp. 1665–1699, 2014.

[25] L. Beirão da Veiga and G. Manzini, "A virtual element method with arbitrary regularity," *SIMA Journal of Numerical Analysis*, vol. 34, no. 2, pp. 759–781, 2014.

[26] A. L. Gain, C. Talischi, and G. H. Paulino, "On the virtual element method for three-dimensional linear elasticity problems on arbitrary polyhedral meshes," *Computer Methods in Applied Mechanics and Engineering*, vol. 282, pp. 132–160, 2014.

[27] F. Brezzi, R. Falk, and L. Donatella M., "Basic principles of mixed virtual element methods," *ESAIM: Mathematical Modelling and Numerical Analysis*, vol. 48, no. 04, pp. 1227–1240, 2014.

[28] Y. Krongauz and T. Belytschko, "Consistent pseudo-derivatives in meshless methods," *Computer Methods in Applied Mechanics and Engineering*, vol. 146, no. 3, pp. 371–386, 1997.

[29] J. Bishop, "A displacement-based finite element formulation for general polyhedra using harmonic shape functions.," *International Journal for Numerical Methods in Engineering*, vol. 97, no. 1, pp. 1–31, 2014.

[30] M. Floater, A. Gillette, and N. Sukumar, "Gradient bounds for Wachspress coordinates on polytopes," *SIAM Journal on Numerical Analysis*, vol. 52, no. 1, pp. 515–532, 2014.

[31] M. Floater, G. Kós, and M. Reimers, "Mean value coordinates in 3D," *Computer Aided Geometric Design*, vol. 22, no. 7, pp. 623–631, 2005.

[32] P. Joshi, M. Meyer, T. DeRose, B. Green, and T. Sanocki, "Harmonic coordinates for character articulation," in *ACM Transactions on Graphics*, ACM, vol. 26, 2007, p. 71.

[33] S. Martin, P. Kaufmann, M. Botsch, M. Wicke, and M. Gross, "Polyhedral finite elements using harmonic basis functions," in *Computer Graphics Forum*, Wiley Online Library, vol. 27, 2008, pp. 1521–1529.

[34] M. Arroyo and M. Ortiz, "Local maximum-entropy approximation schemes: A seamless bridge between finite elements and meshfree methods.," *International*

Journal for Numerical Methods in Engineering, vol. 65, no. 13, pp. 2167–2202, 2006.

[35] K. Hormann and N. Sukumar, "Maximum entropy coordinates for arbitrary polytopes.," in *Eurographics Symposium on Geometry Processing*, vol. 27, 2008, pp. 1513–1520.

[36] N Sukumar, "Quadratic maximum-entropy serendipity shape functions for arbitrary planar polygons," *Computer Methods in Applied Mechanics and Engineering*, vol. 263, pp. 27–41, 2013.

[37] L. Beirão da Veiga, F. Brezzi, A. Cangiani, G. Manzini, L. D. Marini, and A. Russo, "Basic principles of virtual element methods," *Mathematical Models and Methods in Applied Sciences*, vol. 23, no. 1, pp. 199–214, 2013.

[38] L. Beirão da Veiga, F. Brezzi, L. Marini, and A. Russo, "The hitchhiker's guide to the virtual element method," *Mathematical Models and Methods in Applied Sciences*, vol. 24, no. 08, pp. 1541–1573, 2014.

[39] H. Chi, L. Beirão da Veiga, and G. H. Paulino, "Some basic formulations of the virtual element method (VEM) for finite deformations," *Computer Methods in Applied Mechanics and Engineering*, vol. 318, pp. 148–192, 2017.

[40] G. H. Paulino and A. L. Gain, "Bridging art and engineering using Escher-based virtual elements," *Structural and Multidisciplinary Optimization*, vol. 51, no. 4, pp. 867–883, 2015.

[41] L. Beirão da Veiga, F. Brezzi, L. D. Marini, and A. Russo, "H(div) and H(curl)-conforming virtual element methods," *Numerische Mathematik*, vol. 133, no. 2, pp. 303–332, 2016.

[42] L. Beirão da Veiga, F. Brezzi, and L. D. Marini, "Virtual elements for linear elasticity problems," *SIAM Journal on Numerical Analysis*, vol. 51, no. 2, pp. 794–812, 2013.

[43] E. Artioli, S. DeMiranda, C. Lovadina, and L. Patruno, "A stress/displacement virtual element method for plane elasticity problems," *Computer Methods in Applied Mechanics and Engineering*, vol. 325, pp. 155 –174, 2017.

[44] L. Beirão da Veiga, C. Lovadina, and D. Mora, "A virtual element method for elastic and inelastic problems on polytope meshes," *Computer Methods in Applied Mechanics and Engineering*, vol. 295, pp. 327–346, Oct. 2015.

[45] E. Artioli, L. Beirão da Veiga, C. Lovadina, and E. Sacco, “Arbitrary order 2D virtual elements for polygonal meshes: Part I, elastic problem,” *Computational Mechanics*, vol. 60, pp. 355–377, 3 2017.

[46] E. Artioli, L. Beirão da Veiga, C. Lovadina, and E. Sacco, “Arbitrary order 2D virtual elements for polygonal meshes: Part II, inelastic problem,” *Computational Mechanics*, vol. 60, no. 4, pp. 643–657, 2017.

[47] R. L. Taylor and E. Artioli, “VEM for inelastic solids,” in *Advances in Computational Plasticity: A Book in Honour of D. Roger J. Owen*, E. Oñate, D. Peric, E. de Souza Neto, and M. Chiumenti, Eds., Cham: Springer, 2018, pp. 381–394.

[48] F. Brezzi and L. D. Marini, “Virtual Element Methods for plate bending problems,” *Computer Methods in Applied Mechanics and Engineering*, vol. 253, pp. 455–462, 2013.

[49] P. Antonietti, G. Manzini, and M. Verani, “The fully nonconforming virtual element method for biharmonic problems,” *arXiv preprint arXiv:1611.08736*, 2016.

[50] D. Mora, G. Rivera, and I. Velásquez, “A virtual element method for the vibration problem of Kirchhoff plates,” *arXiv preprint arXiv:1703.04187*, 2017.

[51] J. Zhao, S. Chen, and B. Zhang, “The nonconforming virtual element method for plate bending problems,” *Mathematical Models and Methods in Applied Sciences*, vol. 26, no. 09, pp. 1671–1687, 2016.

[52] P. Wriggers, W. Rust, and B. Reddy, “A virtual element method for contact,” *Computational Mechanics*, vol. 58, no. 6, pp. 1039–1050, Dec. 2016.

[53] P. Wriggers, B. Reddy, W. Rust, and B. Hudobivnik, “Efficient virtual element formulations for compressible and incompressible finite deformations,” *Computational Mechanics*, vol. 60, pp. 253–268, 2 2017.

[54] P. Wriggers and B. Hudobivnik, “A low order virtual element formulation for finite elasto-plastic deformations,” *Computer Methods in Applied Mechanics and Engineering*, vol. 327, pp. 459–477, 2017.

[55] F. Bassi, L. Botti, and A. Colombo, “Agglomeration-based physical frame DG discretizations: An attempt to be mesh free,” *Mathematical Models and Methods in Applied Sciences*, vol. 24, no. 08, pp. 1495–1539, 2014.

[56] F. Bassi, L. Botti, A. Colombo, D. A. Di Pietro, and P. Tesini, “On the flexibility of agglomeration based physical space discontinuous Galerkin discretizations,” *Journal of Computational Physics*, vol. 231, no. 1, pp. 45–65, 2012.

[57] L. Beirão da Veiga and G. Manzini, “Residual a posteriori error estimation for the virtual element method for elliptic problems,” *ESAIM: Mathematical Modelling and Numerical Analysis*, vol. 49, no. 2, pp. 577–599, 2015.

[58] A. Cangiani, E. H. Georgoulis, T. Pryer, and O. J. Sutton, “A posteriori error estimates for the virtual element method,” *Numerische Mathematik*, vol. 137, no. 4, pp. 857–893, 2017.

[59] D. Mora, G. Rivera, and R. Rodríguez, “A posteriori error estimates for a virtual element method for the Steklov eigenvalue problem,” *Computers & Mathematics with Applications*, vol. 74, pp. 2172–2190, 9 2017.

[60] S. Berrone and A. Borio, “A residual a posteriori error estimate for the virtual element method,” *Mathematical Models and Methods in Applied Sciences*, vol. 27, 14231458, 08.

[61] M. P. Bendsøe and N. Kikuchi, “Generating optimal topologies in structural design using a homogenization method,” *Computer Methods in Applied Mechanics and Engineering*, vol. 71, no. 2, pp. 197–224, 1988.

[62] A. Diaz and O. Sigmund, “Checkerboard patterns in layout optimization,” *Structural Optimization*, vol. 10, no. 1, pp. 40–45, 1995.

[63] O. Sigmund, S. Torquato, and I. Aksay, “On the design of 1–3 piezocomposites using topology optimization,” *Journal of materials research*, vol. 13, no. 04, pp. 1038–1048, 1998.

[64] P. F. Antonietti, M. Bruggi, S. Scacchi, and M. Verani, “On the virtual element method for topology optimization on polygonal meshes: A numerical study,” *Computers & Mathematics with Applications*, vol. 74, no. 5, pp. 1091–1109, 2017.

[65] A. L. Gain, G. H. Paulino, L. S. Duarte, and I. F. Menezes, “Topology optimization using polytopes,” *Computer Methods in Applied Mechanics and Engineering*, vol. 293, pp. 411–430, 2015.

[66] H. Nguyen-Xuan, “A polytree-based adaptive polygonal finite element method for topology optimization,” *International Journal for Numerical Methods in Engineering*, vol. 110, no. 10, pp. 972–1000, 2017.

[67] T. Y. S. Hoshina, I. F. M. Menezes, and A. Pereira, “A simple adaptive mesh refinement scheme for topology optimization using polygonal meshes,” *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 40, no. 7, p. 348, 2018.

[68] C. Talischi, A. Pereira, I. Menezes, and G. H. Paulino, “Gradient correction for polygonal and polyhedral finite elements,” *International Journal for Numerical Methods in Engineering*, vol. 102, no. 3-4, pp. 728–747, 2015.

[69] D. Boffi, F. Brezzi, and M. Fortin, *Mixed finite element methods and applications*. Springer, 2013.

[70] J. Oden, “Recent developments in the theory of finite element approximations of boundary value problems in nonlinear elasticity,” *Computer Methods in Applied Mechanics and Engineering*, vol. 17, pp. 183–202, 1979.

[71] P. Le Tallec, “Existence and approximation results for nonlinear mixed problems: Application to incompressible finite elasticity,” *Numerische Mathematik*, vol. 38, no. 3, pp. 365–382, 1982.

[72] L. Beirão da Veiga and K. Lipnikov, “A mimetic discretization of the Stokes problem with selected edge bubbles,” *SIAM Journal on Scientific Computing*, vol. 32, no. 2, pp. 875–893, Jan. 2010.

[73] T. J. R. Hughes, *The finite element method: linear static and dynamic finite element analysis*. Courier Dover Publications, 2012.

[74] D Chapelle and K. J. Bathe, “The inf-sup test,” *Computers & Structures*, vol. 47, no. 4, pp. 537–545, 1993.

[75] J. T. Oden and N Kikuchi, “Finite element methods for constrained problems in elasticity,” *International Journal for Numerical Methods in Engineering*, vol. 18, no. 5, pp. 701–725, 1982.

[76] J. Chen, C. Pan, and T. Chang, “On the control of pressure oscillation in bilinear-displacement constant-pressure element,” *Computer Methods in Applied Mechanics and Engineering*, vol. 128, no. 1, pp. 137–152, 1995.

[77] T. Hughes, L. Franca, and M. Balestra, “A new finite element formulation for computational fluid dynamics: V. circumventing the Babuška-Brezzi condition: A stable Petrov-Galerkin formulation of the Stokes problem accommodating equal-order interpolations,” *Computer Methods in Applied Mechanics and Engineering*, vol. 59, no. 1, pp. 85–99, 1986.

[78] F. Brezzi and J. Douglas Jr, “Stabilized mixed methods for the stokes problem,” *Numerische Mathematik*, vol. 53, no. 1-2, pp. 225–235, 1988.

[79] A. Rand, A. Gillette, and C. Bajaj, “Quadratic serendipity finite elements on polygons using generalized barycentric coordinates,” *Mathematics of Computation*, vol. 83, no. 290, pp. 2691–2716, 2014.

[80] R. W. Ogden, *Nonlinear elastic deformations*. Courier Dover Publications, 1997.

[81] J. E. Marsden and T. Hughes, *Mathematical foundations of elasticity*. Courier Corporation, 1994.

[82] P. Wriggers, *Nonlinear finite element methods*. Springer, 2008, vol. 4.

[83] J. Bonet and R. D. Wood, *Nonlinear continuum mechanics for finite element analysis*. Cambridge university press, 2008.

[84] L. R. Herrmann, "Elasticity equations for incompressible and nearly incompressible materials by a variational theorem.," *AIAA Journal*, vol. 3, no. 10, pp. 1896–1900, 1965.

[85] S. N. Atluri and E. Reissner, "On the formulation of variational theorems involving volume constraints," *Computational Mechanics*, vol. 5, no. 5, pp. 337–344, 1989.

[86] J. C. Simo, R. L. Taylor, and K. S. Pister, "Variational and projection methods for the volume constraint in finite deformation elasto-plasticity," *Computer Methods in Applied Mechanics and Engineering*, vol. 51, no. 1, pp. 177–208, 1985.

[87] T. Sussman and K. J. Bathe, "A finite element formulation for nonlinear incompressible elastic and inelastic analysis," *Computers & Structures*, vol. 26, no. 1, pp. 357–409, 1987.

[88] U. Brink and E. Stein, "On some mixed finite element methods for incompressible and nearly incompressible finite elasticity," *Computational Mechanics*, vol. 19, no. 1, pp. 105–119, 1996.

[89] T. Y. P. Chang, A. F. Saleeb, and G. Li, "Large strain analysis of rubber-like materials based on a perturbed lagrangian variational principle," *Computational Mechanics*, vol. 8, no. 4, pp. 221–233, 1991.

[90] J. S. Chen, W. Han, C. T. Wu, and W. Duan, "On the perturbed Lagrangian formulation for nearly incompressible and incompressible hyperelasticity," *Computer Methods in Applied Mechanics and Engineering*, vol. 142, no. 3, pp. 335–351, 1997.

[91] V. V. Belikov, V. D. Ivanov, V. K. Kontorovich, S. A. Korytnik, and A. Y. Semenov, "The non-Sibsonian interpolation: A new method of interpolation of the values of a function on an arbitrary set of points.," *Computational Mathematics and Mathematical Physics*, vol. 37, no. 1, pp. 9–15, 1997.

[92] M. S. Floater, K. Hormann, and G. Kòs, "A general construction of barycentric coordinates over convex polygons.," *Advances in Computational Mathematics*, vol. 24, no. 1-4, pp. 311–331, 2004.

[93] H. Hiyoshi and K. Sugihara, "Two generalizations of an interpolant based on Voronoi diagrams.," *International Journal of Shape Modeling*, vol. 5, no. 2, pp. 219–231, 1999.

[94] E. A. Malsch, J. J. Lin, and G. Dasgupta, "Smooth two dimensional interpolants: A recipe for all polygons," *Journal of Graphics Tools*, vol. 10, p. 2, 2005.

[95] M. Meyer, A. Barr, H. Lee, and M. Desbrun, "Generalized barycentric coordinates on irregular polygons," *J Graphics Tools*, vol. 7, pp. 13–22, 2002.

[96] N. Sukumar, "Construction of polygonal interpolants: A maximum entropy approach.," *International Journal of Numerical Methods in Engineering*, vol. 61, no. 12, pp. 2159–2181, 2004.

[97] M. S. Floater, "Mean value coordinates.," *Computer Aided Geometric Design*, vol. 20, no. 1, pp. 19–27, 2003.

[98] M. Floater, "Generalized barycentric coordinates and applications," *Acta Numerica*, vol. 24, pp. 161–214, 2015.

[99] A. Rand, A. Gillette, and C. Bajaj, "Interpolation error estimates for mean value coordinates over convex polygons," *Advances in Computational Mathematics*, vol. 39, no. 2, pp. 327–347, 2013.

[100] K. Hormann and M. Floater, "Mean value coordinates for arbitrary planar polygons," *ACM Transactions on Graphics (TOG)*, vol. 25, no. 4, pp. 1424–1441, 2006.

[101] M. Crouzeix and P. Raviart, "Conforming and nonconforming finite element methods for solving the stationary Stokes equation," *Revue française d'automatique informatique recherche opérationnelle. Mathématique*, vol. 7, pp. 33–76, 1973.

[102] C Taylor and P Hood, "A numerical solution of the Navier-Stokes equations using the finite element technique," *Computers & Fluids*, vol. 1, no. 1, pp. 73–100, 1973.

[103] N Sukumar and A Tabarraei, "Conforming polygonal finite elements.," *International Journal of Numerical Methods in Engineering*, vol. 61, no. 12, pp. 2045–2066, 2004.

[104] D. Dunavant, "High degree efficient symmetrical Gaussian quadrature rules for the triangle," *International Journal for Numerical Methods in Engineering*, vol. 21, no. 6, pp. 1129–1148, 1985.

[105] S. E. Mousavi, H. Xiao, and N. Sukumar, “Generalized Gaussian quadrature rules on arbitrary polygons.,” *International Journal for Numerical Methods in Engineering*, vol. 82, no. 1, pp. 99–113, 2010.

[106] S. Mousavi and N. Sukumar, “Numerical integration of polynomials and discontinuous functions on irregular convex polygons and polyhedrons,” *Computational Mechanics*, vol. 47, no. 5, pp. 535–554, 2011.

[107] E. B. Chin, J. B. Lasserre, and N. Sukumar, “Numerical integration of homogeneous functions on convex and nonconvex polygons and polyhedra,” *Computational Mechanics*, vol. 56, no. 6, pp. 967–981, 2015.

[108] G. Strang and G. Fix, *An Analysis of the Finite Element Method*, 2nd. Wellesley-Cambridge, 2008.

[109] P. G. Ciarlet, *The Finite Element Method for Elliptic Problems*, 2nd. Society for Industrial and Applied Mathematics, 2002.

[110] C. Talischi, G. H. Paulino, A. Pereira, and I. F. M. Menezes, “PolyMesher: A general-purpose mesh generator for polygonal elements written in Matlab.,” *Structural and Multidisciplinary Optimization*, vol. 45, no. 3, pp. 309–328, 2012.

[111] J. L. Ericksen, “Deformations possible in every compressible, isotropic, perfectly elastic material,” *Journal of Mathematics and Physics*, vol. 34, no. 1-4, pp. 126–128, 1955.

[112] J. L. Ericksen, “Deformations possible in every isotropic, incompressible, perfectly elastic body,” *Zeitschrift für Angewandte Mathematik und Physik (ZAMP)*, vol. 5, no. 6, pp. 466–489, 1954.

[113] M. Biot, “Surface instability of rubber in compression,” *Applied Scientific Research, Section A*, vol. 12, no. 2, pp. 168–182, 1963.

[114] H. Chi, O. Lopez-Pamies, and G. Paulino, “A variational formulation with rigid-body constraints for finite elasticity: Theory, finite element implementation, and applications,” *Computational Mechanics*, vol. 57, pp. 325–338, 2016.

[115] J. Leblanc, *Filled Polymers: Science and Industrial Applications*. CRC Press, 2010.

[116] O. Lopez-Pamies, T. Goudarzi, and K. Danas, “The nonlinear elastic response of suspensions of rigid inclusions in rubber: II — a simple explicit approximation for finite-concentration suspensions,” *Journal of the Mechanics and Physics of Solids*, vol. 61, pp. 19–37, 2013.

[117] T. Goudarzi, D. Spring, G. Paulino, and O. Lopez-Pamies, “Filled elastomers: A theory of filler reinforcement based on hydrodynamic and interphasial effects,” *Journal of the Mechanics and Physics of Solids*, vol. 80, pp. 37–67, 2015.

[118] O. Lopez-Pamies, “An exact result for the macroscopic response of particle-reinforced Neo-Hookean solids,” *Journal of Applied Mechanics*, vol. 77, no. 2, p. 021 016, 2010.

[119] J Moraleda, J Segurado, and J LLorca, “Finite deformation of incompressible fiber-reinforced elastomers: A computational micromechanics approach,” *Journal of the Mechanics and Physics of Solids*, vol. 57, no. 9, pp. 1596–1613, 2009.

[120] A. Gent, “Cavitation in rubber: A cautionary tale,” *Rubber Chemistry and Technology*, vol. 63, no. 3, pp. 49–53, 1990.

[121] K. Cho and A. N. Gent, “Cavitation in model elastomeric composites,” *Journal of Materials Science*, vol. 23, no. 1, pp. 141–144, 1988.

[122] V. Lefevre, K. Ravi-Chandar, and O. Lopez-Pamies, “Cavitation in rubber: An elastic instability or a fracture phenomenon?” *International Journal of Fracture*, pp. 1–23, 2014.

[123] O Lopez-Pamies, “A new I_1-based hyperelastic model for rubber elastic materials,” *Comptes Rendus Mecanique*, vol. 338, no. 1, pp. 3–11, 2010.

[124] J. M. Ball, “Discontinuous equilibrium solutions and cavitation in nonlinear elasticity,” *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, vol. 306, no. 1496, pp. 557–611, 1982.

[125] X. Xu and D. Henao, “An efficient numerical method for cavitation in nonlinear elasticity,” *Mathematical Models and Methods in Applied Sciences*, vol. 21, no. 08, pp. 1733–1760, 2011.

[126] T. Nakamura and O. Lopez-Pamies, “A finite element approach to study cavitation instabilities in non-linear elastic solids under general loading conditions,” *International Journal of Non-Linear Mechanics*, vol. 47, no. 2, pp. 331–340, 2012.

[127] O. Lopez-Pamies, T. Nakamura, and M. I. Idiart, “Cavitation in elastomeric solids: I — a defect-growth theory,” *Journal of the Mechanics and Physics of Solids*, vol. 59, pp. 1464–1487, 2011.

[128] O. Lopez-Pamies, T. Nakamura, and M. I. Idiart, “Cavitation in elastomeric solids: II — onset-of-cavitation surfaces for Neo-Hookean materials,” *Journal of the Mechanics and Physics of Solids*, vol. 59, pp. 1488–1505, 2011.

[129] O. Lopez-Pamies, "Onset of cavitation in compressible, isotropic, hyperelastic solids," *Journal of Elasticity*, vol. 94, no. 2, pp. 115–145, 2009.

[130] C. Talischi and G. H. Paulino, "Addressing integration error for polygonal finite elements through polynomial projections: A patch test connection," *Mathematical Models and Methods in Applied Sciences*, vol. 24, no. 08, pp. 1701–1727, 2014.

[131] G. Manzini, A. Russo, and N. Sukumar, "New perspective on polygonal and polyhedral finite element method," *Mathematical Models and Methods in Applied Sciences*, vol. 24, no. 08, pp. 1665–1699, 2014.

[132] L. Beirão da Veiga, F. Brezzi, L. Marini, and A. Russo, "Virtual element method for general second-order elliptic problems on polygonal meshes," *Mathematical Models and Methods in Applied Sciences*, vol. 26, pp. 729–750, 2014.

[133] D. Mora, G. Rivera, and R. Rodríguez, "A virtual element method for the Steklov eigenvalue problem," *Mathematical Models and Methods in Applied Sciences*, vol. 25, no. 8, pp. 1421–1445, 2015.

[134] M. F. Benedetto, S. Berrone, S. Pieraccini, and S. Scialò, "The virtual element method for discrete fracture network simulations," *Comput. Methods Appl. Mech. Engrg.*, vol. 280, pp. 135–156, 2014.

[135] P. Antonietti, L. Beirão da Veiga, S. Scacchi, and M. Verani, "A C^1 virtual element method for the Cahn–Hilliard equation with polygonal meshes," *SIAM Journal on Numerical Analysis*, vol. 54, no. 1, pp. 34–56, 2016.

[136] R. W. Ogden, "Volume changes associated with the deformation of rubber-like solids," *Journal of the Mechanics and Physics of Solids*, vol. 24, no. 6, pp. 323–338, 1976.

[137] R. W. Ogden, "Nearly isochoric elastic deformations: Application to rubberlike solids," *Journal of the Mechanics and Physics of Solids*, vol. 26, no. 1, pp. 37–57, 1978.

[138] L. Beirão da Veiga, C. Lovadina, and G. Vacca, "Divergence free virtual elements for the Stokes problem on polygonal meshes," *ESAIM: Mathematical Modelling and Numerical Analysis*, vol. 51, no. 2, pp. 509–535, 2017.

[139] L. Beirão da Veiga, C. Lovadina, and G. Vacca, "Virtual elements for the Navier–Stokes problem on polygonal meshes," *SIAM Journal on Numerical Analysis*, vol. 56, no. 3, pp. 1210–1242, 2018.

[140] B. Ahmad, A. Alsaedi, F. Brezzi, L. D. Marini, and A. Russo, "Equivalent projectors for virtual element methods.," *Computers and Mathematics with Applications*, vol. 66, no. 3, pp. 376–391, 2013.

[141] D. S. Malkus and T. Hughes, "Mixed finite element methods–reduced and selective integration techniques: A unification of concepts," *Computer Methods in Applied Mechanics and Engineering*, vol. 15, no. 1, pp. 63–81, 1978.

[142] M. Crisfield, *Non-Linear Finite Element Analysis of Solids and Structures: Advanced Topics*. John Wiley & Sons, Inc., 1997.

[143] R. S. Thedin, A. Pereira, I. F. Menezes, and G. H. Paulino, "Polyhedral mesh generation and optimization for finite element computations," in *Proceedings of the XXXV Ibero-Latin American Congress on Computational Methods in Engineering*, 2014.

[144] J. Segurado and J. Llorca, "A numerical approximation to the elastic properties of sphere-reinforced composites," *Journal of the Mechanics and Physics of Solids*, vol. 50, no. 10, pp. 2107–2121, 2002.

[145] N. Sukumar and E. A. Malsch, "Recent advances in the construction of polygonal finite element interpolations.," *Archives of Computational Methods in Engineering*, vol. 13, no. 1, pp. 129–163, 2006.

[146] S. Rjasanow and S. Weißer, "Higher-order BEM-based FEM on polygonal meshes," *SIAM Journal on Numerical Analysis*, vol. 50, no. 5, pp. 2357–2378, 2012.

[147] O. C. Zienkiewicz and J. Zhu, "The superconvergent patch recovery and a posteriori error estimates. part 1: The recovery technique," *International Journal for Numerical Methods in Engineering*, vol. 33, no. 7, pp. 1331–1364, 1992.

[148] O. C. Zienkiewicz and J. Z. Zhu, "The superconvergent patch recovery and a posteriori error estimates. part 2: Error estimates and adaptivity," *International Journal for Numerical Methods in Engineering*, vol. 33, no. 7, pp. 1365–1382, 1992.

[149] G. Maisano, S. Micheletti, S. Perotto, and C. Bottasso, "On some new recovery-based a posteriori error estimators," *Computer Methods in Applied Mechanics and Engineering*, vol. 195, no. 37, pp. 4794–4815, 2006.

[150] A. Naga and Z. Zhang, "A posteriori error estimates based on the polynomial preserving recovery," *SIAM Journal on Numerical Analysis*, vol. 42, no. 4, pp. 1780–1800, 2004.

[151] M. Ainsworth and A. Craig, "A posteriori error estimators in the finite element method," *Numerische Mathematik*, vol. 60, no. 1, pp. 429–463, 1991.

[152] B. Boroomand and O. Zienkiewicz, "Recovery by equilibrium in patches (REP)," *International Journal for Numerical Methods in Engineering*, vol. 40, no. 1, pp. 137–164, 1997.

[153] M. Ainsworth and J. T. Oden, *A posteriori error estimation in finite element analysis*. John Wiley & Sons, 2011, vol. 37.

[154] T. Blacker and T. Belytschko, "Superconvergent patch recovery with equilibrium and conjoint interpolant enhancements," *International Journal for Numerical Methods in Engineering*, vol. 37, no. 3, pp. 517–536, 1994.

[155] R. Verfürth, *A posteriori error estimation techniques for finite element methods*. OUP Oxford, 2013.

[156] O. Zienkiewicz and J. Zhu, "The superconvergent patch recovery (SPR) and adaptive finite element refinement," *Computer Methods in Applied Mechanics and Engineering*, vol. 101, no. 1-3, pp. 207–224, 1992.

[157] K. Balaji, R. Amirtham, and S. Paul, "Adaptive Poly-FEM for the analysis of plane elasticity problems," *International Journal for Computational Methods in Engineering Science and Mechanics*, vol. 18, no. 2-3, pp. 146–165, 2017.

[158] B. Boroomand and O. Zienkiewicz, "An improved REP recovery and the effectivity robustness test," *International Journal for Numerical Methods in Engineering*, vol. 40, no. 17, pp. 3247–3277, 1997.

[159] Z. Zhang and A. Naga, "A new finite element gradient recovery method: Superconvergence property," *SIAM Journal on Scientific Computing*, vol. 26, no. 4, pp. 1192–1213, 2005.

[160] A. Naga and Z. Zhang, "The polynomial-preserving recovery for higher-order finite element methods in 2d and 3d," *Discrete and Continuous Dynamical Systems Series B*, vol. 5, no. 3, p. 769, 2005.

[161] Z. Zhang, "Polynomial preserving gradient recovery and a posteriori estimate for bilinear element on irregular quadrilaterals," *Int. J. Numer. Anal. Model*, vol. 1, no. 1, pp. 1–24, 2004.

[162] L. Beirão da Veiga, F. Dassi, and A. Russo, "High-order virtual element method on polyhedral meshes," *Computers & Mathematics with Applications*, vol. 74, pp. 1110–1122, 5 2017.

[163] A. Sommariva and M. Vianello, "Gauss–green cubature and moment computation over arbitrary geometries," *Journal of Computational and Applied Mathematics*, vol. 231, no. 2, pp. 886–896, 2009.

[164] F. Brezzi, A. Buffa, and K. Lipnikov, "Mimetic finite differences for elliptic problems.," *Mathematical Modelling and Numerical Analysis*, vol. 43, no. 2, pp. 277–295, 2009.

[165] L. Beirão da Veiga, K. Lipnikov, and G. Manzini, "Arbitrary-order nodal mimetic discretizations of elliptic problems on polygonal meshes," *SIAM Journal of Numerical Analysis*, vol. 49, no. 5, pp. 1737–1760, 2011.

[166] L. Beirão da Veiga, C. Lovadina, and A. Russo, "Stability analysis for the virtual element method," *Mathematical Models and Methods in Applied Sciences*, vol. 27, no. 13, pp. 2557–2594, 2017.

[167] G. Strang and G. J. Fix, *An analysis of the finite element method*. Prentice-Hall Englewood Cliffs, NJ, 1973, vol. 212.

[168] S. Weißer, "Residual error estimate for BEM-based FEM on polygonal meshes," *Numerische Mathematik*, vol. 118, no. 4, pp. 765–788, 2011.

[169] A. V. Astaneh, F. Fuentes, J. Mora, and L. Demkowicz, "High-order polygonal discontinuous Petrov–Galerkin (PolyDPG) methods using ultraweak formulations," *Computer Methods in Applied Mechanics and Engineering*, vol. 332, pp. 686–711, 2018.

[170] E. T. Filipov, J. Chun, G. H. Paulino, and J. Song, "Polygonal multiresolution topology optimization (PolyMTOP) for structural dynamics," *Structural and Multidisciplinary Optimization*, vol. 53, no. 4, pp. 673–694, 2016.

[171] K. Park, G. H. Paulino, W. Celes, and R. Espinha, "Adaptive mesh refinement and coarsening for cohesive zone modeling of dynamic fracture," *International Journal for Numerical Methods in Engineering*, vol. 92, no. 1, pp. 1–35, 2012.

[172] P. Christensen and A. Klarbring, *An introduction to structural optimization*. Linköping: Springer Science & Business Media, 2009.

[173] R. T. Haftka and Z. Gürdal, *Elements of structural optimization*. Springer Science & Business Media, 2012, vol. 11.

[174] M. P. Bendsoe and O. Sigmund, *Topology optimization: theory, methods, and applications*. Springer Science & Business Media, 2013.

[175] G. Rozvany, "A critical review of established methods of structural topology optimization," *Structural and Multidisciplinary Optimization*, vol. 37, no. 3, pp. 217–237, 2009.

[176] O. Sigmund and K. Maute, "Topology optimization approaches," *Structural and Multidisciplinary Optimization*, vol. 48, no. 6, pp. 1031–1055, 2013.

[177] C. S. Jog and R. B. Haber, "Stability of finite element models for distributed-parameter optimization and topology design," *Computer Methods in Applied Mechanics and Engineering*, vol. 130, no. 3-4, pp. 203–226, 1996.

[178] K. Matsui and K. Terada, "Continuous approximation of material distribution for topology optimization," *International Journal for Numerical Methods in Engineering*, vol. 59, no. 14, pp. 1925–1944, 2004.

[179] S. F. Rahmatalla and C. C. Swan, "A q4/q4 continuum structural topology optimization implementation," *Structural and Multidisciplinary Optimization*, vol. 27, no. 1-2, pp. 130–135, 2004.

[180] J. K. Guest, J. H. Prévost, and T. Belytschko, "Achieving minimum length scale in topology optimization using nodal design variables and projection functions," *International Journal for Numerical Methods in Engineering*, vol. 61, no. 2, pp. 238–254, 2004.

[181] J. P. Groen, M. Langelaar, O. Sigmund, and M. Ruess, "Higher-order multi-resolution topology optimization using the finite cell method," *International Journal for Numerical Methods in Engineering*, vol. 110, no. 10, pp. 903–920, 2017.

[182] G. W. Jang, J. H. Jeong, Y. Y. Kim, D. Sheen, C. Park, and M. N. Kim, "Checkerboard-free topology optimization using non-conforming finite elements," *International Journal for Numerical Methods in Engineering*, vol. 57, no. 12, pp. 1717–1735, 2003.

[183] G. W. Jang, S. Lee, Y. Y. Kim, and D. Sheen, "Topology optimization using non-conforming finite elements: Three-dimensional case," *International Journal for Numerical Methods in Engineering*, vol. 63, no. 6, pp. 859–875, 2005.

[184] G. H. Paulino and C. H. Le, "A modified q4/q4 element for topology optimization," *Structural and Multidisciplinary Optimization*, vol. 37, no. 3, pp. 255–264, 2009.

[185] J. K. Guest and L. C. Smith G., "Reducing dimensionality in topology optimization using adaptive design variable fields," *International Journal for Numerical Methods in Engineering*, vol. 81, no. 8, pp. 1019–1045, 2010.

[186] T. H. Nguyen, G. H. Paulino, J. Song, and C. H. Le, "A computational paradigm for multiresolution topology optimization (mtop)," *Structural and Multidisciplinary Optimization*, vol. 41, no. 4, pp. 525–539, 2010.

[187] T. H. Nguyen, G. H. Paulino, J. Song, and C. H. Le, "Improving multiresolution topology optimization via multiple discretizations," *International Journal for Numerical Methods in Engineering*, vol. 92, no. 6, pp. 507–530, 2012.

[188] A. Abdelkader, C. L. Bajaj, M. S. Ebeida, A. H. Mahmoud, S. A. Mitchell, J. D. Owens, and A. A. Rushdi, "Sampling conditions for conforming Voronoi meshing by the VoroCrust algorithm," *CoRR*, vol. abs/1803.06078, 2018. arXiv: `1803.06078`.

[189] J. Pouderoux, M. Charest, M. Kenamond, and M. Shashkov, "2d & 3d Voronoi meshes generation with shapo," in *The 8th International Conference on Numerical Methods for Multi-Material Fluid Flow (MULTIMAT 2017)*, Santa-Fe, NM, USA, 2017.

[190] K. Hormann and N. Sukumar, *Generalized Barycentric Coordinates in Computer Graphics and Computational Mechanics*, K. Hormann and N. Sukumar, Eds. CRC Press, 2018.

[191] P. Wriggers, B. Hudobivnik, and J. Korelc, "Efficient low order virtual elements for anisotropic materials at finite strains," in *Advances in Computational Plasticity: A Book in Honour of D. Roger J. Owen*, E. Oñate, D. Peric, E. de Souza Neto, and M Chiumenti, Eds., vol. 46, Cham: Springer, 2018, pp. 417–434.

[192] M. L. De Bellis, P. Wriggers, B. Hudobivnik, and G. Zavarise, "Virtual element formulation for isotropic damage," *Finite Elements in Analysis and Design*, vol. 144, pp. 38–48, 2018.

[193] M. F. Benedetto, A. Caggiano, and G. Etse, "Virtual elements and zero thickness interface-based approach for fracture analysis of heterogeneous materials," *Computer Methods in Applied Mechanics and Engineering*, vol. 338, pp. 41–67, 2018.

[194] O. Sigmund, "A 99 line topology optimization code written in Matlab," *Structural and Multidisciplinary Optimization*, vol. 21, no. 2, pp. 120–127, 2001.

[195] E. Andreassen, A. Clausen, M. Schevenels, B. S. Lazarov, and O. Sigmund, "Efficient topology optimization in Matlab using 88 lines of code," *Structural and Multidisciplinary Optimization*, vol. 43, no. 1, pp. 1–16, 2011.

[196] K. Liu and A. Tovar, "An efficient 3d topology optimization code written in Matlab," *Structural and Multidisciplinary Optimization*, vol. 50, no. 6, pp. 1175–1196, 2014.

[197] A. Pereira, C. Talischi, G. H. Paulino, I. F. Menezes, and M. S. Carvalho, "Fluid flow topology optimization in PolyTop: Stability and computational implementa-

tion," *Structural and Multidisciplinary Optimization*, vol. 54, no. 5, pp. 1345–1364, 2016.

[198] P. Wei, Z. Li, X. Li, and M. Y. Wang, "An 88-line Matlab code for the parameterized level set method based topology optimization using radial basis functions," *Structural and Multidisciplinary Optimization*, pp. 1–19, 2018.

[199] L. Beirão da Veiga, F. Brezzi, F. Dassi, L. Marini, and A. Russo, "Lowest order virtual element approximation of magnetostatic problems," *Computer Methods in Applied Mechanics and Engineering*, 2018.

[200] L. Beirão da Veiga, F. Brezzi, F. Dassi, L. Marini, and A. Russo, "Virtual element approximation of 2d magnetostatic problems," *Computer Methods in Applied Mechanics and Engineering*, vol. 327, pp. 173–195, 2017.

[201] J. R. Barber, *Elasticity.* 3rd. Springer, 2010.

[202] M. P. Bendsøe, "Optimal shape design as a material distribution problem," *Structural Optimization*, vol. 1, no. 4, pp. 193–202, 1989.

[203] G. I. N. Rozvany, M. Zhou, and T. Birker, "Generalized shape optimization without homogenization," *Structural optimization*, vol. 4, no. 3-4, pp. 250–252, 1992.

[204] M. P. Bendsøe and O. Sigmund, "Material interpolation schemes in topology optimization," *Archive of applied mechanics*, vol. 69, no. 9-10, pp. 635–654, 1999.

[205] B. Bourdin, "Filters in topology optimization," *International Journal for Numerical Methods in Engineering*, vol. 50, no. 9, pp. 2143–2158, 2001.

[206] T. Zegard and G. H. Paulino, "Bridging topology optimization and additive manufacturing," *Structural and Multidisciplinary Optimization*, vol. 53, no. 1, pp. 175–192, 2016.

[207] C. C. Wang, "A new representation theorem for isotropic functions: An answer to professor gf smith's criticism of my papers on representations for isotropic functions," *Archive for Rational Mechanics and Analysis*, vol. 36, no. 3, pp. 166–197, 1970.

[208] G. Smith, "On isotropic functions of symmetric tensors, skew-symmetric tensors and vectors," *International Journal of Engineering Science*, vol. 9, no. 10, pp. 899–916, 1971.

[209] R. D. Mindlin, "Micro-structure in linear elasticity," *Archive for Rational Mechanics and Analysis*, vol. 16, no. 1, pp. 51–78, 1964.

[210] W. Koiter, “Couple-stress in the theory of elasticity, I & II,” in *Proc. K. Ned. Akad. Wet*, North Holland Pub, vol. 67, 1964, pp. 17–44.

[211] R. A. Toupin, “Elastic materials with couple-stresses,” *Archive for Rational Mechanics and Analysis*, vol. 11, no. 1, pp. 385–414, 1962.

[212] N. Fleck and J. Hutchinson, “A phenomenological theory for strain gradient effects in plasticity,” *Journal of the Mechanics and Physics of Solids*, vol. 41, no. 12, pp. 1825–1857, 1993.

[213] N. Fleck, G. Muller, M. Ashby, and J. Hutchinson, “Strain gradient plasticity: Theory and experiment,” *Acta Metallurgica et Materialia*, vol. 42, no. 2, pp. 475–487, 1994.

[214] N. Fleck and J. Hutchinson, “Strain gradient plasticity,” *Advances in Applied Mechanics*, vol. 33, pp. 296–361, 1997.

[215] E. C. Aifantis, “The physics of plastic deformation,” *International Journal of Plasticity*, vol. 3, no. 3, pp. 211–247, 1987.

[216] H. Gao, Y. Huang, W. D. Nix, and J. W. Hutchinson, “Mechanism-based strain gradient plasticityI. Theory,” *Journal of the Mechanics and Physics of Solids*, vol. 47, no. 6, pp. 1239–1263, 1999.

[217] A. C. Fannjiang, G. H. Paulino, and Y. Chan, “Strain gradient elasticity for antiplane shear cracks: A hypersingular integrodifferential equation approach,” *SIAM Journal on Applied Mathematics*, vol. 62, no. 3, pp. 1066–1091, 2002.

[218] R. De Borst and H. B. Mühlhaus, “Gradient-dependent plasticity: Formulation and algorithmic aspects,” *International Journal for Numerical Methods in Engineering*, vol. 35, no. 3, pp. 521–539, 1992.

[219] R. De Borst and J. Pamin, “Some novel developments in finite element procedures for gradient-dependent plasticity,” *International Journal for Numerical Methods in Engineering*, vol. 39, no. 14, pp. 2477–2505, 1996.

[220] E. Amanatidou and N. Aravas, “Mixed finite element formulations of strain-gradient elasticity problems,” *Computer Methods in Applied Mechanics and Engineering*, vol. 191, no. 15-16, pp. 1723–1751, 2002.

[221] L. Beirão da Veiga, D. Mora, and G. Rivera, “Virtual elements for a shear-deflection formulation of Reissner–Mindlin plates,” *Mathematics of Computation*, vol. 88, no. 315, pp. 149–178, 2019.

[222] I. Babuška, T. Strouboulis, C. Upadhyay, S. Gangaraj, and K. Copps, "Validation of a posteriori error estimators by numerical approach," *International journal for numerical methods in engineering*, vol. 37, no. 7, pp. 1073–1123, 1994.

[223] J. M. Melenk and I. Babuška, "The partition of unity finite element method: Basic theory and applications," *Computer Methods in Applied Mechanics and Engineering*, vol. 139, no. 1-4, pp. 289–314, 1996.

[224] C. A. Duarte and J. T. Oden, "An hp-adaptive method using clouds," *Computer Methods in Applied Mechanics and Engineering*, vol. 139, no. 1-4, pp. 237–262, 1996.

[225] J. T. Oden, C. A. Duarte, and O. C. Zienkiewicz, "A new cloud-based hp finite element method," *Computer Methods in Applied Mechanics and Engineering*, vol. 153, no. 1-2, pp. 117–126, 1998.

[226] N. Moës, J. Dolbow, and T. Belytschko, "A finite element method for crack growth without remeshing," *International Journal for Numerical Methods in Engineering*, vol. 46, no. 1, pp. 131–150, 1999.

[227] N. Sukumar, N. Moës, B. Moran, and T. Belytschko, "Extended finite element method for three-dimensional crack modelling," *International Journal for Numerical Methods in Engineering*, vol. 48, no. 11, pp. 1549–1570, 2000.

[228] T. Belytschko, R. Gracie, and G. Ventura, "A review of extended/generalized finite element methods for material modeling," *Modelling and Simulation in Materials Science and Engineering*, vol. 17, no. 4, p. 043 001, 2009.

[229] E. B. Chin, J. Lasserre, and N. Sukumar, "Modeling crack discontinuities without element-partitioning in the extended finite element method," *International Journal for Numerical Methods in Engineering*, vol. 110, no. 11, pp. 1021–1048, 2017.

[230] G. Manzini, E. Benvenuti, A. Chiozzi, and N. Sukumar, "Numerical experiments with the extended virtual element method for the Laplace problem with strong discontinuities," Los Alamos National Laboratory, 2018.

[231] W. Celes, G. H. Paulino, and R. Espinha, "A compact adjacency-based topological data structure for finite element mesh representation," *International Journal for Numerical Methods in Engineering*, vol. 64, no. 11, pp. 1529–1556, 2005.

[232] S. L. Vatanabe, T. N. Lippi, R. Cícero de Lima, G. H. Paulino, and E. C. N. Silva, "Topology optimization with manufacturing constraints: A unified projection-based approach," *Advances in Engineering Software*, vol. 100, pp. 97–112, 2016.

[233] A. T. Gaynor and J. K. Guest, “Topology optimization considering overhang constraints: Eliminating sacrificial support material in additive manufacturing through design,” *Structural and Multidisciplinary Optimization*, vol. 54, no. 5, pp. 1157–1172, 2016.

[234] M. Zhou, B. S. Lazarov, F. Wang, and O. Sigmund, “Minimum length scale in topology optimization by geometric constraints,” *Computer Methods in Applied Mechanics and Engineering*, vol. 293, pp. 266–282, 2015.

[235] B. S. Lazarov and F. Wang, “Maximum length scale in density based topology optimization,” *Computer Methods in Applied Mechanics and Engineering*, vol. 318, pp. 826–844, 2017.

[236] S. R. M. Almeida, G. H. Paulino, and E. C. N. Silva, “A simple and effective inverse projection scheme for void distribution control in topology optimization,” *Structural and Multidisciplinary Optimization*, vol. 39, no. 4, pp. 359–371, 2009.

[237] L. Stromberg, A. Beghini, W. Baker, and G. Paulino, “Application of layout and topology optimization using pattern gradation for the conceptual design of buildings,” *Structural and Multidisciplinary Optimization*, vol. 43, no. 2, pp. 165–180, 2011.

[238] M. E. Gurtin, *An Introduction to Continuum Mechanics*. Academic press, 1982, vol. 158.

[239] V. M. Nguyen-Thanh, X. Zhuang, H. Nguyen-Xuan, T. Rabczuk, and P. Wriggers, “A virtual element method for 2D linear elastic fracture analysis,” *Computer Methods in Applied Mechanics and Engineering*, 2018.