Paulo Akira Figuti Enabe

Virtual Element Method Applied to the Linear Elastic Model

São Paulo
2021

Paulo Akira Figuti Enabe

Virtual Element Method Applied to the Linear Elastic Model

Revised Version

Dissertation submitted to the Polytechnic School of the University of São Paulo for the Master in Sciences Degree.

Structural Engineering

Advisor: Professor Rodrigo Provasi

São Paulo
2021

Este exemplar foi revisado e alterado em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, 03 de janeiro de 2022

Assinatura do autor Paulo Silva F. Enabe

Assinatura do orientador Rodrigo Provasi Correia

Este exemplar foi revisado e corrigido em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, _____ de _____ de _____

Assinatura do autor: _____

Assinatura do orientador: _____

Catalogação-na-publicação

Enabe, Paulo Akira Figuti
Virtual Element Method Applied to the Linear Elastic Model / P. A. F. Enabe -- versão corr. -- São Paulo, 2022.
123 p.

Dissertação (Mestrado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Estruturas e Geotécnica.

1.método dos elementos virtuais 2.método dos elementos finitos 3.modelo elástico linear 4.equação de Poisson I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Estruturas e Geotécnica II.t.

Acknowledgments

To Professor Rodrigo, for all the teaching and advice provided in the last four years that we have been working together and for all the support in difficult times.

To all my family and professors that were fundamental in this trajectory, providing me with all the support and resources necessary to have this work done.

To my friends André, Andrés and Caio for all this years of friendship that were crucial in my trajectory.

To Guilherme for all discussions, friendship and all work during the past years.

To Professor Alfredo for all constructive critics and advice.

To Marcelo for all his support and for providing me new opportunities in my career.

Resumo

Com a evolução dos computadores e da complexidade dos problemas em engenharia, é natural que também ocorra o surgimento de novos métodos numéricos que se insiriam nessa realidade. O Método dos Elementos Virtuais (MEV) se propõe a generalizar o clássico Método dos Elementos Finitos (MEF), sendo mais permissivo ao que se diz respeito aos elementos de discretização na malha, abrangendo qualquer polígono convexo e não convexo. Utilizar quaisquer polígonos traz como consequência a utilização de funções de forma não polinomiais. Para tanto, o método busca computar tais funções de forma implícita, sem a necessidade de fórmulas de quadratura. O método foi originalmente aplicado a Equação de Poisson e, por se tratar de um método relativamente recente, a gama de aplicações voltadas para problemas reais de engenharia de estruturas ainda não é tão vasta quando comparada com, por exemplo, o Método dos Elementos Finitos ou o Método dos Elementos Finitos Generalizados. Assim existem muitos caminhos possíveis para serem explorados com o intuito de expandir o estado da arte referente ao MEV. Neste projeto, desenvolve-se uma metodologia para aplicação do Métodos dos Elementos Virtuais no modelo reológico elástico linear. E, consequentemente, realizam-se comparações com o clássico Método dos Elementos Finitos ao que se diz respeito a desempenho em geometrias complexas, levando em consideração as particularidades e características de cada método.

Palavras-chave: método dos elementos virtuais, método dos elementos finitos, modelo elástico linear, equação de poisson

Abstract

Considering the evolution of computers in the recent years and the notorious increase in the complexity of engineering problems, it is natural that new numerical methods come up in order to take part in this reality. The Virtual Element Method (VEM) main proposal is to generalize the classical Finite Element Method (FEM), being more permissive regarding the mesh discretization, embracing every convex and non-convex polygon. Using this large variety of polygons types brings as consequence the necessity of working with non-polynomial functions. The method computes these functions implicitly, without the need of any quadrature formula. The Virtual Element Method was originally developed for the Poisson Equation and, for being relatively recent, the range of applications related to structural engineering is still very limited when compared to the Finite Element Method or the Generalized Finite Element Method. In this sense, there are a lot of possible paths that can be followed aiming to expand the state of art related to VEM. On this project, it is presented a methodology for the application of Virtual Element Method on the linear elastic rheological model. Consequently, comparisons with the classical FEM were made with respect the performance alongside simple and complex geometries, considering the particularities and characteristics of each method.

Keywords: virtual element method, finite element method, linear elastic model, poisson equation

Contents

List of Symbols

Δ	Laplace operator
Ω	Polygonal domain
$\partial\Omega$	Boundary of polygonal domain Ω
Π^{∇}	Projection operator for non-constant polynomials spaces
$\Pi^{\nabla}_{\mathcal{CS}_K}$	Projection operator related to the constant strain modes
$\Pi^{\nabla}_{\mathcal{R}_K}$	Projection operator related to the rigid body motions
$\Pi^{\nabla}_{\mathbb{E}_K}$	Projection operator related to the polynomial terms
$\underline{\mathbf{\Pi}}^{\nabla}$	Matrix representation of the projection operator
$\underline{\mathbf{\Pi}}^{\nabla}_{\dagger}$	Extended projection operator in matrix form
$\underline{\Pi}^{\nabla}_{\mathcal{CS}_K}$	Matrix form of the operator related to the constant strain modes
$\underline{\Pi}^{\nabla}_{\mathcal{CS}_{K,\dagger}}$	Extended matrix form of the operator related to the constant strain modes
$\underline{\Pi}^{\nabla}_{\mathcal{R}_K}$	Matrix form of the operator related to the rigid body motions
$\underline{\Pi}^{\nabla}_{\mathcal{R}_{K,\dagger}}$	Extended matrix form of the operator related to the rigid body motions
$\underline{\Pi}^{\nabla}_{\mathbb{E}_K}$	Matrix form of the operator related to the polynomial terms
$\underline{\Pi}^{\nabla}_{\mathbb{E}_{K,\dagger}}$	Extended matrix form of the operator related to the polynomial terms
Φ	Prandtl Stress Function
Ψ	Warping function
ε_q	Strain quadratic formula
$\varepsilon_x, \varepsilon_y, \varepsilon_z$	Normal strain components
$\hat{\varepsilon}$	Mean tensor associated to the strain modes
γ	Distortion between two unitary fibers
$\gamma_{xy}, \gamma_{xz}, \gamma_{yz}$	Shear strain components
λ	Stretch of a fiber
μ	Mass associated to a generic solid
ν	Poisson coefficient
σ	Normal stress component
σ_F	Stress field associated with the force F
$\sigma_{nominal}$	Nominal stress field
$\sigma_x, \sigma_y, \sigma_z$	Normal stress components
θ	Rotation angle related to torsion
θ'	Twist related to torsion

τ	Shear stress componenet
τ_h	Decomposition of a polygonal domain into simple polygons
$\tau_{xy}, \tau_{xz}, \tau_{yz}$	Shear stress components
$\hat{\chi}^{\mathbf{0}}$	Unitary vector
$\hat{\psi}$	Mean tensor associated to the rigid body motion
$\mathbf{a_c}$	Acceleration field
$a(\cdot,\cdot)$	Continuous bilinear form
$a_h(\cdot,\cdot)$	Discrete bilinear form
$a_{h,K}(\cdot,\cdot)$	Discrete bilinear form defined in a polygon K
$a_K(\cdot,\cdot)$	Continuous bilinear form defined in a polygon K
$\mathbf{b}$	Body force applied to a generic solid
e	Edge of a simple polygon
$\mathbf{e_1}, \mathbf{e_2}, \mathbf{e_3}$	Elements of the canonical basis of Euclidean space
f_b	Load term associated to the body forces
$f_{b,h}$	Global discrete load term associated to the body forces
$f_{b,K,h}$	Local discrete load term associated to the body forces
f_h	Discrete load term in V_h'
f_s	Load term associated to the external traction
$f_{s,h}$	Global discrete load term associated to the external traction
$f_{s,K,h}$	Local discrete load term associated to the external traction
$\mathbf{g}$	External traction (Neumann boundary condition)
h	Maximum polygonal diameter
h_e	Height of the cross-section
h_K	Polygonal diameter of a polygon K
$\hat{\mathbf{n}}$	Normal unitary vector
k	Order of accuracy
k_{CF}	Concentration factor constant
$l_{K,e}$	Length of an edge e
m_α	Scalad monomial
n_K	Number of vertices and edges of a polygon K
$\mathbf{p}(\cdot,\cdot)$	Generic stress term
u_π	Piecewise approximation of u in $\mathbb{P}_k(K)$
u	Analytical solution or displacement field
u_h	Discrete approximated solution
u_I	Approximation of u in V_h
$\mathbf{x_c}$	Centroid of a polygon
w_i	Width of the cross section
$\underline{\mathbf{B}}, \underline{\mathbf{D}}, \underline{\mathbf{G}}, \mathbb{G}$	Intermediary matrices for the construction of the stiffness matrix
$\underline{\mathbf{C}}, \underline{\mathbf{H}}$	Intermediary matrices for the construction of the load vector
$\mathcal{C}$	Inverse of the constitutive symmetric operator

CF	Concentration factor
$\mathcal{CS}_K$	Space of constant strain modes
$\mathcal{D}$	Constitutive symmetric operator
$\mathbb{D}_K$	Space of deformations associated to a simple polygon K
$\underline{\mathbf{D}}_{\mathcal{R}}, \underline{\mathbf{D}}_{\mathcal{CS}}$	Intermediary matrices in linear elasticity context
E	Elastic modulus
$\underline{\mathbf{E}}$	The Green-Lagrange strain tensor
$\mathbb{E}_k(\partial K)$	Space of continuous functions defined on the edges of polygon K
$\underline{\mathbf{F}}$	Deformation gradient
G	Shear modulus
$H^k(U)$	The Sobolev space for $p = 2$
$H_0^k(U)$	Closure of $C_c^\infty(U)$ in $W^{k,p}(U)$ for $p = 2$
$\underline{\mathbf{I}}$	Identity matrix
I_t	Analytical torsion constant
I_t^h	Approximated torsion constant
K	Simple polygon
∂K	Boundary of simple polygon K
$\underline{\mathbf{K}}_h$	Local stiffness matrix
L	Elliptic operator
$L^1_{loc}(U)$	Space of locally integrable functions
$L^p(U)$	Lesbegue space with $1 \leq p \leq \infty$
L_k^K	Projection operator regarding to the L^2-norm
$\underline{\mathbf{L}}$	L^2-projection operator matrix
$\mathcal{M}_k(K)$	Set of scaled monomials regarding to polygon K and order of accuracy k
N_{dof}	Total number of degrees of freedom in polygon K
N_{edge}	Total number of edges in a polygonal domain
N_{el}	Total number of elements in a polygonal domain
N_{vert}	Total number of vertices in a polygonal domain
P_0	Projection operator for constant polynomial space
$\mathbb{P}_k$	Polynomial space with degree k
$\mathcal{R}_K$	Space of rigid body motions
S_d	Set of points with prescribed displacement
S_f	Set of points where the external traction is applied
$S_K(\cdot,\cdot)$	Symmetric stability term
$\mathbf{S}_K$	Matrix form of the stability term
T	Constant torque
$\underline{\mathbf{T}}$	Cauchy Stress Tensor
U	Open subset in $\mathbb{R}^n$
V_h	Global virtual element space
V_h'	Dual space of global virtual element space

$V_{h,K,k}$	Local virtual element space
$\mathcal{V}_K$, $\mathcal{E}_K$, $\mathcal{P}_K$	Local sets of degrees of freedom
$\mathcal{V}$, $\mathcal{E}$, $\mathcal{P}$	Global sets of degrees of freedom
$W^{k,p}(U)$	The Sobolev space
$W_0^{k,p}(U)$	Closure of $C_c^\infty(U)$ in $W^{k,p}(U)$

List of Figures

Chapter 1

Introduction

A great variety of engineering problems does not possess analytical solutions or they are quite difficult to obtain in general contexts. In this sense, considering the evolution of computers, numerical methods became an indispensable tool for engineers. There is a entire research line dedicated to study this subject and it intersects different areas like mathematics, physics and engineering. This research line is very general and is focused on studying algorithms to solve mathematical problems numerically. Methods that are related to partial differential equations (in short, PDEs) and integral equations are particularly interesting to handle engineering problems. A classical method that is already consolidated both on industry and on academia is the Finite Element Method (FEM). This method first appearance dates from the 1940s and it is based on the work of Alexander Hrennikof and Richard Courant.

In a general way, the Finite Element Method has foundations on variational calculus and on analysis of partial differential equations. The method aims to solve PDEs approximately by dividing the domain on smaller pieces called elements. Figure 1.1 shows three dimensional images using tetrahedral elements. FEM has a sophisticated model but a relatively simple implementation, making it very popular among different fields of engineering and being applied not only to structural engineering but also to thermal, electromagnetic and fluid dynamics problems. There is also a very strong mathematical interest behind FEM because partial differential equations are a broad and fertile field.

While the complexity of engineering problems increases yearly, the computational power grows up in the same proportion. In this scenario, alternative methods started appearing to substitute, generalize or complement the classical Finite Element Method. The Generalized Finite Element Method (GFEM) is a method that uses the concept of partition of unity on the enrichment process to build the shape functions. As result, characteristics and information of the problem's original differential equation are inserted on each approximation space and more precise approximation functions can be found. More details about the GFEM can be found in Babuška and Melenk (1997) and Melenk and Babuška (1996). Other alternative method is the Finite Cell Method (FCM) that addresses the mesh discretization. Here, the

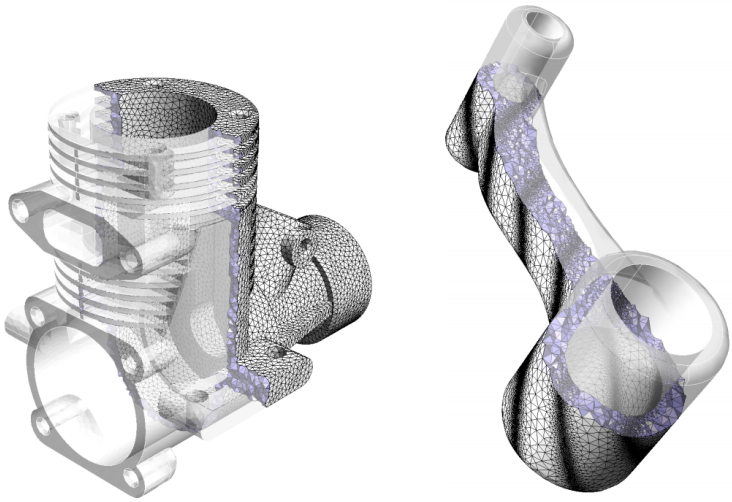


Figure 1.1: *Three dimensional mesh with tetrahedral elements.*
Source:(Geuzaine and Remacle, 2009)

mesh is not only made for the geometry but for the domain using squares and cubes as elements. Figure 1.2 shows the meshing process in steps (a) to (f) on FCM related to the modal analysis of a ship propeller. Step (a) shows the definition of geometry domains exported by a CAD software, step (b) shows the discretization of the domains using cubes, step (c) shows the finite cell mesh, step (d) shows the refinement of cells, step (e) shows the connection of finite cells and step (f) shows the result of the first eigenmode. Quadrature methods are use to determine what is inside and whats is out of the geometry. For further details about the method one can see Schillinger and Ruess (2015). The Smoothed Finite Element Method (SFEM) is a combination of the classical FEM with meshless methods properties. It aims to make mesh regularity less restrictive when compared to FEM and more details can be found in Zeng and Liu (2018) and Zhang *et al.* (2020). Finally, the Virtual Element Method (VEM) is the main study object of this project and it is described further.

It should be noted that working with numerical methods is a task that is not limited to the field of application. Often, the models of these methods derive from concepts of pure mathematics and must be adapted to an engineering solution. Therefore, it can be quite a challenging task and it is important for the researcher working in this area to posses domain from both ends. The VEM is a direct consequence of the Lax-Milgram Theorem and the Riesz Representation Theory that are classical results of functional analysis and partial differential equation analysis. The understanding of the details of the mathematical model allows the researcher to have plenty domain of all potentialities of the method, understand its limitations and find ways to enhance it.

The Virtual Element Method (VEM) is a relatively new model, first published in 2012 and developed by a group of mathematicians in Italy. The method has a rigorous mathematical formulation requiring a background that is not common in engineering courses. To understand the VEM and its potentialities it is necessary to be familiar with concepts such

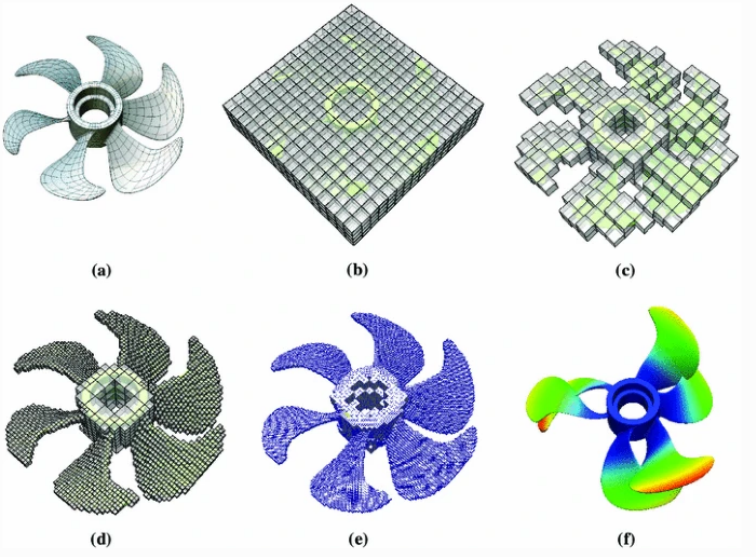


Figure 1.2: *Finite Cell Method mesh discretization process on a ship propeller.* Source:(Schillinger and Ruess, 2015)

as Functional Analysis, Measurement Theory and Analysis of Partial Differential Equations. The method aims to generalize the Finite Element Method with respect to the discretization of the mesh, by being less restrictive with respect to the elements, covering any convex and non-convex polygon. Figure 1.3 shows an illustrative example of non-convex mesh. Using any polygons results in the use of non-polynomial functions. Therefore, the method seeks to compute such functions implicitly, without properly knowing them. However, this makes the method highly dependent on the choice of degrees of freedom and, consequently, on the geometric input parameters.

Although recent, the VEM already has a certain projection in the scientific community due to its performance in complex geometries and versatility in discretization. The benchmark used for the method is the Poisson Equation and, even that, requires a considerable amount of work to be understood and implemented, as shown in next section. Even though the description of Poisson Equation is broadly available, the formulation is not extensively used in real structural engineering problems except for some particular problems and applications. This is due to its mathematical formulation that is not as intuitive as other numerical methods. Consequently, there is a lack of comparative analysis of the performance of the VEM to other methods in engineering problems. Thus, the central problem that the project intends to address is to analyze the performance of the VEM applied in a linear elastic rheology for a complex geometry.

1.1 Objectives

The main motivation for this project is the Virtual Element Method. Since VEM is very recent, it has many paths to be explored with great potential for engineering research. As stated in the previous section, the method generalizes the already consolidated FEM. In this

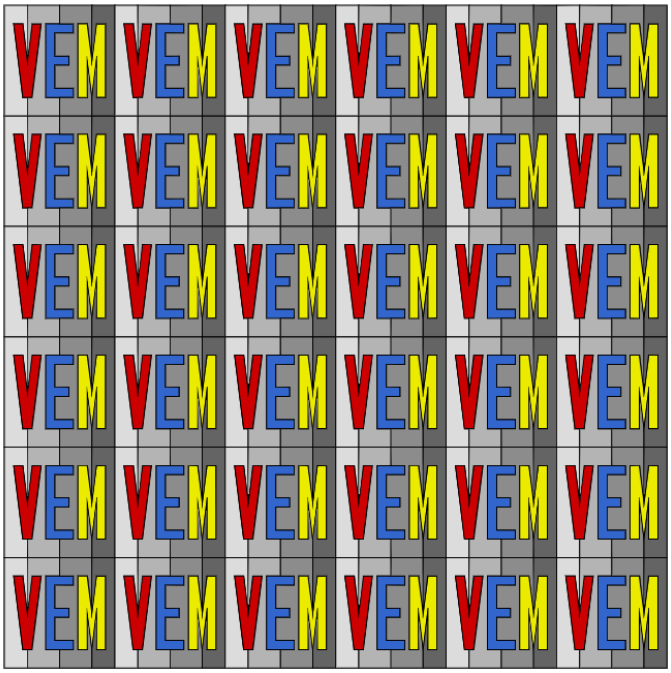


Figure 1.3: *Non-convex mesh using the word VEM and a background as element.*
Source:(Park *et al.*, 2019)

way, it improves aspects of the FEM, making it able to become a powerful tool in structural analysis. As this method is born from a research group in mathematics, at first, it is not concerned with engineering applications. Over the past few years some applications have emerged, showing the method a promising future. Thus, regarding to the Virtual Element Method aspects mentioned before, the following hypothesis are used:

- Due to the less demanding quality of the mesh thanks to the greater flexibility in the form of the discretizing elements, the Virtual Element Method is more accurate and more precise when working with complex geometries than the classic Finite Element Method.

- The Virtual Element Method is more robust about the mesh distortion. For example, when the nodes coalesce, the method continues to present trustful results.

Those hypothesis were formulated based on the works of da Veiga *et al.* (2013a), da Veiga *et al.* (2014), da Veiga *et al.* (2017b) and Mengolini *et al.* (2019).

The first part of the project aims to study the model of the method for the Poisson Equation in a very detailed way, considering all mathematical aspects. Then, apply the method to the linear elastic model. In this sense, the general objective of the project is to explore the Virtual Elements Method particularities and characteristics. And, consequently, to compare the results of VEM and FEM with complex geometries in structural engineering problems. Therefore, the mathematical formulation for linear elastic problems is developed based on the methodology used in the development of the method for the Poisson Equation presented in the first VEM model.

1.2 Outline

This work is divided in 9 chapters. The first chapter that was already presented is the introduction to contextualize the problem and present the objectives. Chapter 2 is dedicated to the literature review of the main works related to VEM. The next chapter is related to the formulation of the Virtual Element Method to Poisson Equation and the mathematical model of the method is presented. On Chapter 4, the implementation matrix framework is shown alongside an example of a problem with analytical solution.

The Theory of Elasticity is introduced in Chapter 5 and Chapter 6 is dedicated to the Saint-Venant Torsion problem, where the formulation using Pradtl's Function is presented. At the end, the results of VEM, FEM and FDM are compared. Next, the Virtual Element Method applied in the linear elasticity context is presented. Thus, use cases of the VEM are presented in Chapter 8, including the complex geometry case. Finally, Chapter 9 is dedicated to conclusions.

Chapter 2

Bibliographic Review

In this chapter the bibliographic review concerning to the Virtual Element Method is presented. The first section is dedicated to discuss the original mathematical model of VEM for Poisson Equation and for the differential equations of linear elasticity. Also in this section, papers about Virtual Element Method implementation are analyzed. The second section is focused on VEM usage in more specific applications in order to illustrate its versatility in different situations.

2.1 The Virtual Element Method

The paper da Veiga *et al.* (2013a) is a canonical paper of the Virtual Element Method. The authors presented in this work the mathematical formulation of the method for the Poisson Equation in two dimensions. It started with the continuous formulation of this equation in order to find a weak solution for it. For that, weak derivatives and Sobolev Spaces were introduced. With the weak formulation, the discrete problem was introduced. It was presented a set of hypotheses that establishes the main components of the method and a set of hypotheses that presents conditions for its operation. Using these hypotheses, a theorem was enunciated stating that the discrete problem has a unique solution and that the method converges. Using this theorem, the authors showed that the method derives from the classic result in Analysis of Partial Differential Equations, which is the Lax-Milgram Theorem. Also in this article, it was shown how the construction of the virtual element space is made, which can contain both polynomial and non-polynomial functions. It was also proven that the set of degrees of freedom chosen for a given geometry is unisolvent, that is, this set defines a single function in the space of virtual elements. In order to deal with non-polynomial functions, the projection operator Π^{∇} was introduced, which projects functions from the space of virtual elements onto the space of polynomials.

The work presented by da Veiga *et al.* (2014) gave a direction of how VEM can be applied to the Poisson Equation. This paper has more practical aspects than the previous one. Taking a random geometry, a discretization was made in simple polygons (convex or non-convex).

Here, the authors elucidated how the choice of degrees of freedom should be made. Taking any function in the virtual element space, the chosen degrees of freedom were: the value of the function at the vertices of each polygon, the value of the function at the midpoint of each edge and the value of the function at the interior points of the polygon. From there, a non-canonical basis was defined for the polynomial space. This basis takes into consideration a weighting with the geometric parameters (centroid, polygonal diameter and area) of each element of discretization. Then, the projection operator Π^{∇} was used to build the stiffness matrix using this non-canonical basis for the polynomial space and the chosen degrees of freedom. It was shown that the local stiffness matrix is computed directly from the degrees of freedom, reinforcing the importance of a robust input of geometry data. The loading vector was computed by introducing a L^2 operator that projects elements of the virtual element space into the polynomial space using the L^2-norm. Finally, some results are presented for elements of different shapes.

Exploring VEM properties, de Dios *et al.* (2016) presented a nonconforming formulation to the method regarding to the Poisson Equation. The nonconforming term here is related to the mesh. The paper focused on proposing a very general mathematical model for the nonconforming VEM in two and three dimensions. In this sense, a formulation for any order of accuracy and any polygon shape was presented. The order of accuracy refers to the dimension of the polynomial space in which the functions of the space of virtual elements are projected. The authors stated that for triangular elements and order of accuracy equal to one or two, the choice of degrees of freedom is the same for nonconforming VEM and the nonconforming Finite Element Method. Because of the generality of VEM formulation regarding to the mesh, the construction of the model is very similar to what is done in both of previous papers. No numerical analysis was done but some formulation comparison regarding the nonconforming FEM are made. The authors claim that the VEM formulation is more direct and complex meshes can be analyzed in simpler way using functional analysis tools. A extension of this work is presented on Cangiani *et al.* (2017b), where the authors proposed a unified framework for conforming and nonconforming Virtual Element Method. Also, numerical simulations were done for the Poisson Equation in a unitary square domain, showing that conforming VEM has very close results to nonconforming VEM.

The Virtual Element Method formulation was presented for general elliptic problems in da Veiga *et al.* (2016) and not just for the Laplace Operator as it has been done previously. Roughly speaking, to make this formulation possible, the projector operator that was originally defined for the classical H^1-norm is defined for the L^2-norm. The formulation of the projection operator with the L^2-norm was developed in Ahmad *et al.* (2013). The pipeline for the construction of the method is very similar to the papers presented earlier and a new formulation for the error estimation is shown. Numerical tests showed that, for low accuracy order, the method converged as expected.

In da Veiga *et al.* (2013b) a discussion was made about the application of the Virtual Element Method regarding the linear elastic model. The theorems presented da Veiga *et al.*

(2013a) are revisited and proved again. A set of hypotheses was made about the regularity of the space of virtual elements that, in classical works, was implicit. The construction of the method for the elastic rheology is very similar to what was presented in the classical articles, being done from a rigorous mathematical perspective and not focusing on implementation. Thus, the construction of the method components, like the stiffness matrix and the loading vector, were not very clear for immediate application. In Gain *et al.* (2014), a three-dimensional formulation of VEM was presented with a focus on its application to the linear elastic rheological model regarding to a set of differential equations to solve the linear elasticity for solids. The formulation was derived in detail and an implementation framework was presented. Both works presented some guidelines for error analysis and some numerical tests. An extension of these works was presented in da Veiga *et al.* (2015), which the focus is the formulation of VEM for non-linear elastic and inelastic problems for small deformations.

Artioli *et al.* (2017) proposed a matrix framework and explained in detail how to build each component of VEM formulation. The main focus of the paper was to present guidelines to implement Virtual Element Method applied to linear elastic model, very similar to what is done in da Veiga *et al.* (2014). This work is more general in many aspects than da Veiga *et al.* (2013b) because it can be extended to non-linear problems. In the first part of the work, general aspects of VEM, like the transition of the continuous problem to the discrete problem, the choice of degrees of freedom and construction of virtual element space, are presented. The second part is dedicated to discuss the construction of the bilinear form and the load vector using a matrix framework. In particular, the bilinear form is divided into consistent and stabilization term and their construction is discussed for linear and non-linear cases. The linearity of VEM model is related to the order of accuracy as was discussed before. Some tests on simple geometries were made and VEM was compared with FEM. The Virtual Element Method numerical solutions were very close to the finite element ones both for linear and non-linear model. Although, the author states that VEM was insensible with respect to mesh distortions.

Sutton (2016) and Ortiz-Bernardin *et al.* (2019) centered the discussion not only on the characteristics and particularities of the method implementation, but also proposed a solver for VEM. In Sutton (2016), it was presented a practical work but dealing only with the Poisson's Equation in two dimensions without any rheological model applied. The author's main goal was to present the implementation of a solver for the Virtual Element Method in MATLAB. The used model follows the same methodology presented in the last two previous articles. After presenting the model, the author detailed each part of the developed code. By presenting the code, the authors make the model less abstract and bring it more on par to the structural engineering context. This project used both a MATLAB native mesh generator and a generator developed by Talischi *et al.* (2012) (also written in MATLAB) called *Polymesher*. Examples of the mesh types can be seen in the figure 2.1. No comparisons were made between the Finite and Virtual Element Method.

Ortiz-Bernardin *et al.* (2019) proposed an implementation of the Virtual Element Method

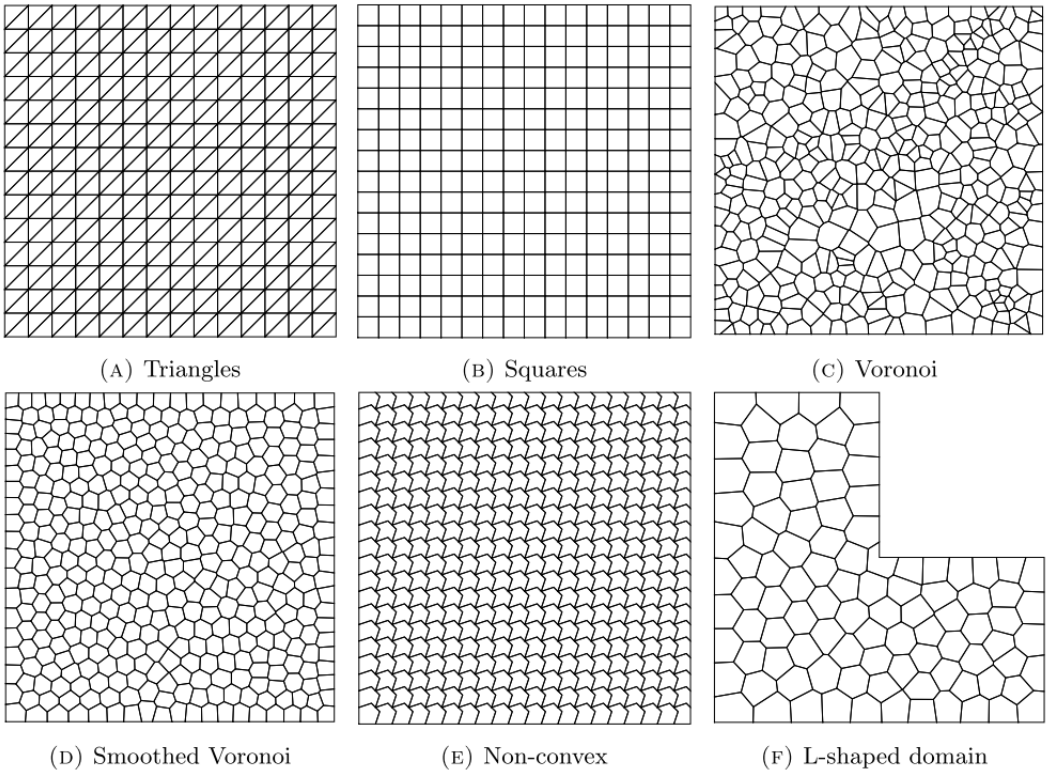


Figure 2.1: *Mesh examples available on Sutton (2016) solver.*
Source:(Sutton, 2016)

applied to the linear elastic model and the Poisson Equation in two dimensions using C ++. The approach to solve the Poisson equation vary from previous articles. While in other works the problem is dealt with a pure mathematical to numerical approach, this paper follows the numerical path from the start. This can be clearly seen by the preponderant use of matrix algebra in the model. Therefore, the implementation of the elliptical differential equation was restricted to a particular implementation of the linear elastic model. In this project, the authors took full advantage of the object orientation available in the C ++ language. In the project was available a built-in mesh generator and the *Polymesher* generator. The implementation for the linear elastic model was built upon the Weak Galerkin Method. The matrix framework was a particularization for what was done in Gain *et al.* (2014) for the case in two dimensions. Comparisons with FEM are allowed within the project due to an available functionality that allows the calculation of the method's convergence considering the L^2 norm and H^1 norm. A comparison was made between FEM and VEM implementations, concluding that for the same number of degrees of freedom and for the same number of elements of discretization, the precision and accuracy in the results was similar. A simulation of a problem with analytical solution in a unitary square domain was also carried out. The domain was discretized using Voronoi mesh and the Virtual Element Method was applied. This problem can be seen in Figure 2.2. It is obtained that in the standard L^2 the error is less than 3 %.

The work presented in Zhang *et al.* (2019) is an extension to the liner elasticity regarding

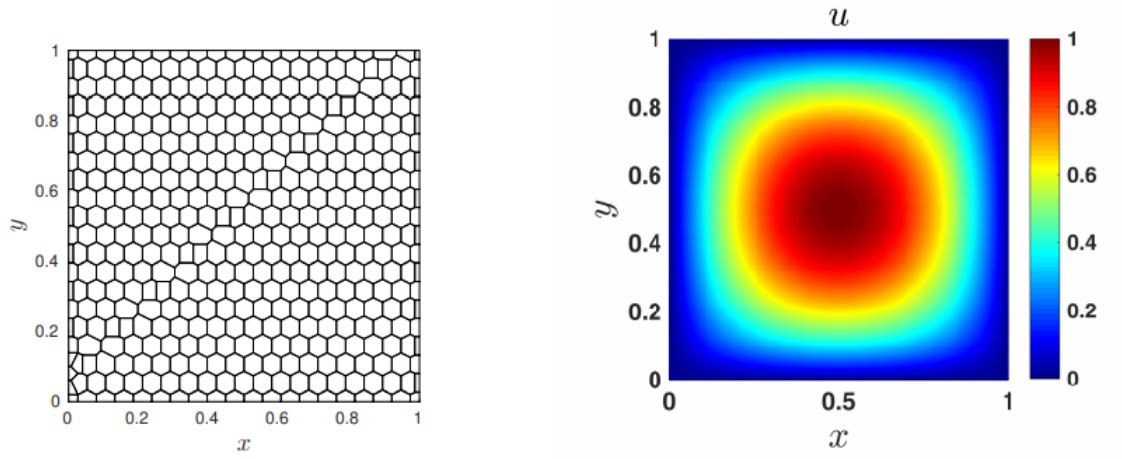


Figure 2.2: *Square plate domain for Poisson Equation and analytical solution.*
Source:(Ortiz-Bernardin *et al.*, 2019)

the mesh nonconforming VEM. The authors started with a quick review about nonconforming FEM, stating that high-order elements for classical nonconforming FEM are difficult to build. It was said that, for nearly incompressible materials, the lock phenomenon can happen. Therefore, the main proposal of the paper was to build a locking-free nonconforming VEM. The model was formulated based on the linear elasticity presented in da Veiga *et al.* (2013b) and on nonconforming VEM presented in de Dios *et al.* (2016). Numerical simulations were performed and a comparison to conforming VEM was made, concluding that the results were very close. Also, the authors conclude, as have been shown in the nonconforming formulation for the Poisson Equation, that for low accuracy orders and triangular mesh the nonconforming VEM coincides with nonconforming FEM.

Mengolini *et al.* (2019) focused on the comparison between Finite Element Method and Virtual Element Method by studying linear elastic models. In the first part of the article, the authors described the qualitative characteristics of the method and proposed a formulation directly from the perspective of numerical methods. A pseudo-code was also presented, highlighting the main points of its implementation, like computing the local stiffness matrix directly from the degrees of freedom. This pseudo-code summarized what was done in Gain *et al.* (2014), Sutton (2016) and Ortiz-Bernardin *et al.* (2019). In the second part of the article, a comparison was made between VEM and the FEM taking into account the number of degrees of freedom and the polygonal diameter h. Also, it was made a comparison of VEM's performance for different orders of accuracy k. For the evaluation of the error, the standard L^2 norm and the standard classic energy norm were used. The results were evaluated using a quadrilateral plate geometry. In Figure 2.3, it is possible to observe that the convergence towards higher orders of accuracy is faster, making the error smaller for both the number of degrees of freedom and the size of the polygonal diameter. The disadvantage of using higher accuracy orders is that more geometric parameters will have to be computed and more robust the data input must be, thus making the implementation of the method more complicated. Is noteworthy that no complex geometry was tested.

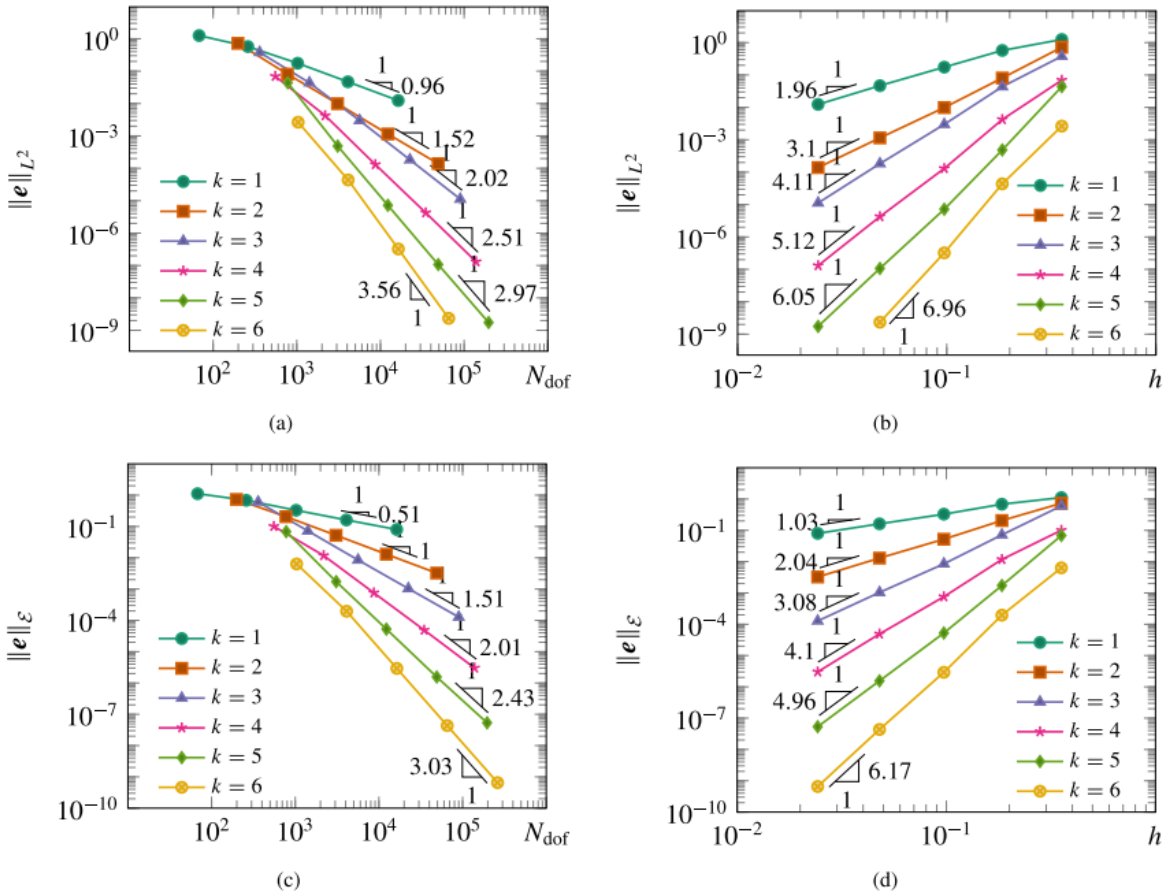


Figure 2.3: *VEM convergence for different accuracy order with respect to the degrees of freedom and polygonal diameter. The L^2 norm is presented on* ***(a)*** *and* ***(b)*** *and the energetic norm is presented on* ***(c)*** *and* ***(d)***

Source:(Mengolini *et al.*, 2019)

From the papers reviewed, independently of the chosen approach (numerical method framework or pure mathematics abstraction), it is clear that there is a methodology for the implementation of Virtual Element Method concerning to the Poisson Equation as it is described below:

1. Domain discretization.
2. Construction of virtual element space.
3. Introduction of projection operator Π^{∇}.
4. Construction of the local stiffness matrix.
5. Construction of the local load vector.
6. Assembly the local components and solve the global problem as in the Finite Element Method.

This project aims to develop a VEM formulation for linear elastic rheology problems, similar to what is done using Finite Element MEthod.

2.2 Applications of the Virtual Element Method

Some applications of the Virtual Element Method in more specific areas are presented in this section. In Wriggers *et al.* (2016), the method was applied to the problem of structural contact, using Lagrange multipliers and penalty method, as in classical Finite Element Method approach. According to the authors, the application of the virtual elements in the contact formulation makes it possible to build a node-to-node contact approach. The contact meshes were transformed into coincident meshes at the interfaces that were not necessarily coincident. In general, VEM allows the addition of nodes in the discretization and, consequently, the calculations for each element remains unchanged. A non-matching mesh and a mesh with additional nodes considering VEM mesh discretization are shown in Figure 2.4. In this article, only the linear case of VEM was used and the approach was very close to Gain *et al.* (2014). A new way to build the bilinear form was introduced by Wriggers *et al.* (2016). In Aldakheel *et al.* (2020), the contact formulation was extended to curved edges that are related to modeling complex geometries. The authors concluded that there were no major complications regarding the implementation of VEM.

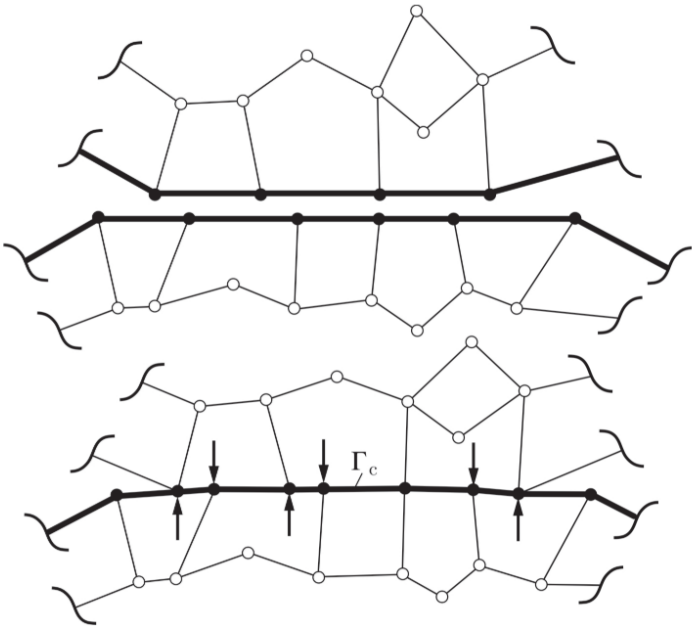
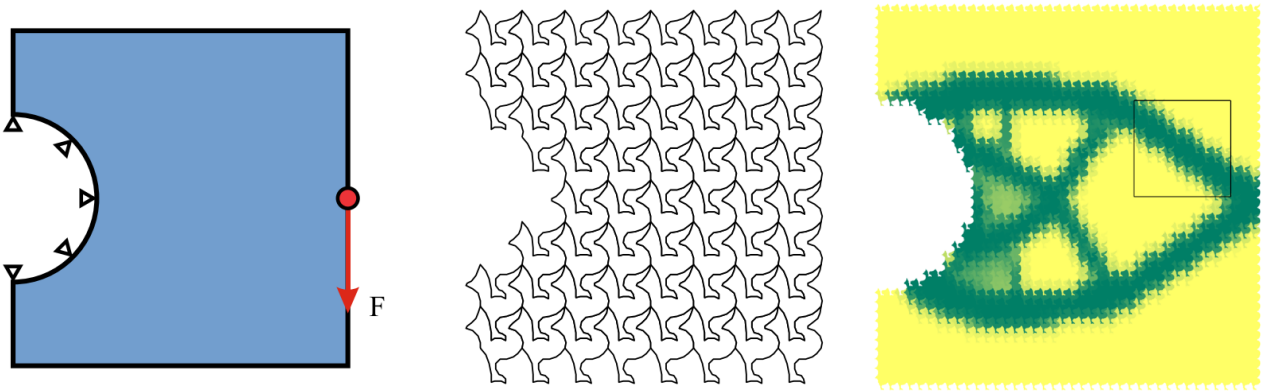


Figure 2.4: *Non-matching mesh and a mesh with additional nodes considering VEM mesh discretization*

Source:(Wriggers *et al.*, 2016)

Paulino and Gain (2015) presented an application of VEM for topology optimization using tessellation. According to the authors, topology optimization main goal is to optimize the material distribution according to design requirements. The Virtual Element Method was used to solve the elasticity state equations concerning to the optimization. The authors stated that tessellation is up to the next stage of element shape evolutionary line and in the first part of the work a discussion about that was presented, starting with tessellation of simple polygons and then presenting the ideas behind M.C. Escher's Tessellations. The

basic idea was to use the fact that VEM can work with any polygon and, consequently, the shape functions can be computed implicitly. Some tests were performed using meshes of different shapes and some comparisons with FEM were made. The results with VEM were consistent and near to FEM solution. Figure 2.5 shows an example of topology optimization: a) cantilever beam problem, b) mesh with bird elements, c) converged topology.



(a) *Cantilever beam problem* **(b)** *Mesh with bird figures* **(c)** *Converged topology*

Figure 2.5: *Example of topology optimization.*
Source:(Paulino and Gain, 2015)

The Virtual Element Method can also be applied to fluid dynamics problems. Considering the two dimensional case, according to the authors in da Veiga *et al.* (2017a), the non-linearity of the Navier-Stokes Equation led to the introduction of new projections not included in the original formulation of the method. The work proposed a rigorous error analysis development, taking into account the characteristics of VEM. Simulations were also carried out to test the numerical performance of the Virtual Element Method. They concluded that the method is a valid approach, since considerably small errors were obtained.

The literature referring to VEM has been expanding in recent years, focusing not only on the model itself but on contributions related to the particularities and potentialities of the method. da Veiga *et al.* (2017b) focused on the stability analysis of the term that makes up the bilinear form of the Virtual Element Method related to the treatment of functions in a non-polynomial way (stability term). The authors intended to prove that the method is robust by considering more general meshes. To analyze the stability, the Poisson equation in two dimensions was considered. The article developed an approach to prove the convergence of VEM using weaker stability conditions than in the classic formulation. Tests were carried out in different situations to validate this approach. An analysis of the classic stability term (presented in da Veiga *et al.* (2013a)) is also made for more general meshes. It was proved that this term is equivalent to semi-norm in the Sobolev space H^1. The article is essentially mathematical, with numerical results that attest to the robustness of the method.

Cangiani *et al.* (2017a) implemented the Virtual Element Method for a quasilinear problem in which the projection operator of the method was used to treat non-linearity. In the case of a non-linear problem, an iterative method was used, more specifically the fixed point

method. Also, it was proved that the problem is well posed with VEM.

Wriggers *et al.* (2020) proposed a greater generalization for the Virtual Element Method discretization elements. The authors used an isoparametric and *NURBS* approach so that the discretization elements do not necessarily need to have straight edges, allowing VEM to adapt even more to complex geometries and to have even greater flexibility regarding the quality of the mesh. Although, the proposal was restricted to small orders of accuracy, it was mentioned by the authors that it can be extended to high orders.

These applications are useful in this project as they show the versatile characteristics of the Virtual Element Method. Also, the papers present properties of the method that can be explored and can be particularized for the linear elastic model. For example, stability of the method concerning to less restrictive mesh requirements is particularly interesting to enhance VEM applications in structural engineering problems.

Chapter 3

The Virtual Element Method and the Poisson Equation

The focus of this chapter is to present the VEM formulation for the Poisson Equation as it was originally conceived. But first, some results in mathematics will be presented and discussed. The continuous Poisson Problem will be shown and from it the weak form will be constructed. Following the weak form, the discrete problem will be derived and the VEM formulation for that problem will be presented. The chosen approach is to give a detailed formulation of the method in order to show all particularities and characteristics of the method. Finally, a brief discussion about mesh regularity concerning to the usage of considerable small edges will be made in the last section.

3.1 The weak form of Poisson Equation

In this section, the main results of mathematics were based on the works of Evans (2010) and Isnard (2013). Also, the monograph written by Professor Marcelo Furtado (Furtado, 2012) has great influence on this work. The inequalities and function space notations used in this text can be found in Appendix A.

Let U be an open set of $\mathbb{R}^n$ and $n \geq 1$. The Lebesgue Space is given by

$$L^p(U) = \left\{ f(x) : \int_U |f(x)|^p dx < \infty \right\}, \tag{3.1}$$

where $p \in [1, \infty]$. The norm associated to that space is $\|u\|_{L^p(U)} = (\int_U |u(x)|^p dx)^{1/p}$. In this work there is a particular interest in the $p = 2$ case that can be defined as

$$L^2(U) = \left\{ f(x) : \int_U |f(x)|^2 dx < \infty \right\}. \tag{3.2}$$

The Lesbegue Space is important because the weak formulation will be built from it. In a general way, the main goal is to make less restrictions as possible for functions. Basically, it will not be necessary that a function is differentiable everywhere, only integrable in some points. In this way, it is natural to define what does it mean to be locally integrable as presented in Strichartz (2003):

Definition 1. *Let $f : U \longrightarrow \mathbb{C}$ be a measurable function. Then, f is locally integrable if $\int_U |f(x)\phi(x)|dx < \infty$ (absolutely convergent) for every function $\phi \in C_c^\infty(U)$. The space of locally integrable functions is denoted by $L^1_{loc}(U)$.*

For the weak formulation, the week derivative concept is fundamental. It is enough that functions are only differenciable locally as stated by definition below.

Definition 2. *Assume $u, v \in L^1_{loc}(U)$ and α is a multi-index such that $|\alpha| = \alpha_1 + \alpha_2 + ... + \alpha_n = k$. Then, v is the $\alpha - th$ weak partial derivative of u if*

$$\int_U uD^\alpha \varphi dx = (-1)^{|\alpha|} \int_U v\varphi dx$$

for all functions $\varphi \in C_c^\infty(U)$. The $\varphi \in C_c^\infty(U)$ are called test functions.

The term "weak" comes from the fact that instead of a k times derivative function u, it is only required an integrable function v. The weak derivative will be denoted by $D^\alpha u = v$. Now it is possible to define the space of weak derivatives, called Sobolev Space.

Definition 3. *The Sobolev Space is given by:*

$$W^{k,p}(U) = \{u \in L^p(U) | D^\alpha u = v \in L^p(U), \quad \text{for all } \alpha \text{ such that } |\alpha| \leq k\}. \tag{3.3}$$

For $p = 2$ the following notation will be used: $H^k(U) = W^{k,2}(U)$, for k non-negative integer. Accordingly to Furtado (2012), the Sobolev Space norm is given by:

$$\|u\|_{W^{k,p}(U)} = \begin{cases} (\sum\limits_{|\alpha|\leq k} \int |D^\alpha u|^p dx)^{1/p}, & \text{se} \quad p \in [1, \infty) \\ \sum\limits_{|\alpha|\leq k} \|D^\alpha u\|_{L^\infty(U)}, & \text{se} \quad p = \infty \end{cases} \tag{3.4}$$

and the semi-norm is given by

$$|u|_{W^{k,p}(U)} = \begin{cases} (\sum\limits_{|\alpha|=k} \int |D^\alpha u|^p dx)^{1/p}, & \text{se} \quad p \in [1, \infty) \\ \sum\limits_{|\alpha|=k} \|D^\alpha u\|_{L^\infty(U)}, & \text{se} \quad p = \infty \end{cases} \tag{3.5}$$

Let $\{u_i\}_{i\in\mathbb{N}}$ be a sequence in $W^{k,p}(U)$. Thus, the sequence $\{u_i\}_{i\in\mathbb{N}}$ converges to u in $W^{k,p}(U)$ if $\lim\limits_{i\to\infty} \|u - u_i\|_{W^{k,p}(U)} = 0$. In this way, it is possible to prove that every Sobolev space is

a complete space (Banach Space). This result is relevant in order to build Hilbert spaces from Sobolev spaces. The definition of Hilbert spaces can be found in Appendix A. Also, it is necessary to define the closure of $C_c^\infty(U)$ in order to apply Dirichlet boundary conditions.

Definition 4. *For $1 \leq p \leq \infty$ and k a non-negative integer, the closure of $C_c^\infty(U)$ in $W^{k,p}(U)$ with respect to the norm $\|\cdot\|_{W^{k,p}}$ is denoted by $W_0^{k,p}(U)$. For $p = 2$ it will be denoted $H_0^k(U) = W_0^{k,2}(U)$.*

The Virtual Element Method can be seen as a consequence of the next result as will be shown further. The Lax-Milgram Theorem will guarantee under certain conditions the existence and uniqueness of solution for the second order differential problem. But before enunciate the theorem, the following definition is necessary:

Definition 5. *For $u, v \in H_0^1(U)$, the bilinear form is defined by:*

$$a(u,v) = \int_U \nabla u \cdot \nabla v dx$$

From that definition it is possible to enunciate the Lax-Milgram Theorem:

Theorem 1 (Lax-Milgram)**.** *Let H be a real Hilbert space. Suppose that $a : H \times H \mapsto \mathbb{R}$ is a bilinear form as defined above and there exists constants $\alpha, \beta > 0$ such that:*

1. $|a(u,v)| \leq \alpha \|u\| \|v\|$,
2. $\beta \|u\|^2 \leq a(u,u)$,

with $u, v \in H$. Given $F \in H'$ there exists a unique $u \in H$ such that

$$a(u,v) = F(v)$$

for all $v \in H$. The solution u is called weak solution of $a(u,v) = F(v)$.

It is important to recall that H' denotes the dual space of H and $\|\cdot\|$ is the associated norm of the inner product of H.

A more detailed presentation of the Sobolev spaces and a deeper discussion of the Lax-Milgram Theorem can be found in Chapter 5 and 6 of Evans (2010), respectively. These concepts and definitions will be crucial for the construction of the virtual element space and the error analysis concerning to the convergence of VEM. Before delve into the Poisson equation analysis, a brief discussion about the general form of elliptic equations is given as the Poisson Equation is a particular case of second order elliptic equation defined by:

$$\begin{cases} Lu = f \quad \text{em} \quad U \\ u = 0 \quad \text{em} \quad \partial U \end{cases},$$

where $u \in C^2(U) \cap C(\overline{U})$, $f \in L^2(U)$ and L is a second order differential operator. This operator is particularly important because with VEM formulation we will be able to choose different types of it considering some restrictions related to the degrees of freedom and the virtual element space. The second order operator form is equally presented in Evans (2010) and Furtado (2012) as:

$$Lu = -\sum_{i,j=1}^{n} a^{ij}(x)u_{x_i x_j} + \sum_{i=1}^{n} b^i(x)u_{x_i} + c(x)u, \tag{3.6}$$

where $a^{ij}, b^i, c \in L^\infty(U)$ and $x \in U$. The "elliptic" term mentioned above comes from the following definition taken from Evans (2010):

Definition 6. *A second order operator L is elliptic if there exists a positive constant η such that:*

$$\xi A(x)\xi = \sum_{i,j=1}^{n} a^{ij}(x)\xi_i\xi_j \geq \eta|\xi|^2, \tag{3.7}$$

for almost everywhere $x \in U$ and for all $\xi \in R^n - \{0\}$, where

$$A(x) = \begin{bmatrix} a^{11}(x) & a^{12}(x) & \cdots & a^{1n}(x) \\ a^{21}(x) & a^{22}(x) & \cdots & a^{2n}(x) \\ \vdots & \vdots & \ddots & \vdots \\ a^{n1}(x) & a^{n2}(x) & \cdots & a^{nn}(x) \end{bmatrix}$$

is a symmetric matrix for each $x \in U$. This is the same as to require A to be positively defined.

Take $a^{ij}(x) = 1$ if $i = j$, $a^{ij}(x) = 0$ if $i \neq j$, $b^i(x) = 0$ and $c(x) = 0$, for $x \in U$ and $i, j \in [1, n]$. As result we obtain the Laplacian Operator, denoted by Δ.

Now, consider the Poisson Equation in a polygonal domain Ω. The Poisson Equation with Dirichlet boundary condition is given by:

$$\begin{cases} -\Delta u = f & \text{in} \quad \Omega \\ u = 0 & \text{on} \quad \partial\Omega \end{cases}, \tag{3.8}$$

where $u \in C^2(U) \cap C(\overline{\Omega})$ and $f \in L^2(\Omega)$. It is possible to find a analytical solution applying the Green's Function. Although the solution is purely mathematical, the interest lays on the difficulty to numerically compute the solution, especially for general domains. In this sense, the strategy is to weaken the original problem, demanding less from the solution and it is only natural to use the weak derivatives and the Sobolev space.

Let $v \in H_0^1(\Omega)$ be a test function. Multiplying equation (3.8) by v and integrating by

parts:

$$\int_{\Omega}(-\Delta u - f)v dx = \int_{\Omega}\nabla u \cdot \nabla v dx - \int_{\partial\Omega} v\nabla u \cdot \eta dS(x) - \int_{\Omega} f v dx = 0 \tag{3.9}$$

From equation (3.9) it is possible to conclude:

$$\int_{\Omega}\nabla u \cdot \nabla v dx = \int_{\Omega} f v dx. \tag{3.10}$$

Equation (3.10) is weaker than equation (3.8) because it is only needed the first derivative of u that does not need to be continuous, only integrable. Thus, it is enough that $u \in H_0^1(\Omega)$. Now, will be proved the uniqueness of the solution using the Lax-Milgram Theorem. The inner product can be associated to the space $H_0^1(\Omega)$ by:

$$(u, v)_{H_0^1(\Omega)} = \int_{\Omega}\nabla u \cdot \nabla v dx. \tag{3.11}$$

This inner product induces the norm of $H_0^1(\Omega)$:

$$\|u\|_{H_0^1(\Omega)} = \|u\|_{W^{1,2}(\Omega)} = \left(\int_{\Omega}|\nabla u|^2 dx\right)^{1/2} = (u, u)^{1/2}_{H_0^1(\Omega)}. \tag{3.12}$$

As mentioned before, the Sobolev space is complete and, with the defined inner product, it is possible to conclude that $H_0^1(\Omega)$ is a Hilbert space. Using Definition 5,

$$a(u, v) = \int_{\Omega}\nabla u \cdot \nabla v dx = (u, v)_{H_0^1(\Omega)}. \tag{3.13}$$

By Hölder and Cauchy-Schwarz inequalities presented in Appendix A, it is possible to conclude that

$$\begin{aligned} |a(u, v)| = \left|\int_{\Omega}\nabla u \cdot \nabla v dx\right| \leq \int_{\Omega}|\nabla u||\nabla v| dx \leq \|\nabla u\|_{L^2(\Omega)}\|\nabla v\|_{L^2(\Omega)} = \\ = \|\nabla u\|_{H_0^1(\Omega)}\|\nabla v\|_{H_0^1(\Omega)} \end{aligned} \tag{3.14}$$

As $|a(u, v)| \leq \|\nabla u\|_{H_0^1(\Omega)}\|\nabla v\|_{H_0^1(\Omega)}$, then the bilinear form a is continuous. It is also possible to observe that

$$a(u, u) = \int_{\Omega}|\nabla u|^2 dx = \|u\|^2_{H_0^1(\Omega)}, \tag{3.15}$$

guaranteeing the coercivity of the bilinear form. Let $F(v) = \int_{\Omega} f v dx$ be a linear functional.

Using again the Cauchy-Schwarz and Hölder inequality:

$$\left| \int_\Omega f v dx \right| \leq \int_\Omega |fv| dx \leq \int_\Omega |f||v| dx \leq \|f\|_{L^2(\Omega)} \|v\|_{L^2(\Omega)}. \tag{3.16}$$

By Poincaré Inequality, there exists $C > 0$ such that

$$\|f\|_{L^2(\Omega)} \|v\|_{L^2(\Omega)} \leq \|f\|_{L^2(\Omega)} C \|\nabla v\|_{L^2(\Omega)}. \tag{3.17}$$

Thus, the linear functional is continuous. By applying the Lax-Milgram Theorem, we conclude the uniqueness of the solution for equation (3.10).

3.2 The discrete problem

After transforming the continuous problem into the weak counterpart and proving that it has a solution and it is unique, the next step is to setup the discrete problem in order to use the numerical methods. This setup is done for the Virtual Element Method, thus it is proved that the discrete problem has solution and it is unique for VEM formulation. The following sections are based on the canonical work of da Veiga *et al.* (2013a) and in the work presented on Savarè and Chanon (2016). The construction pipeline of the method used here is very similar to both works and complementary commentaries were added in order to make text simpler to read and the model easier to understand.

Considering a decomposition τ_h of Ω in polygons K. The h subscript refers to the maximum polygonal diameter, defined below:

Definition 7. *Given a polygon K, the polygonal diameter, denoted by h_K, is the largest distance between two non consecutive vertices. The maximum polygonal diameter is $h = \max_{K \in \tau_h} h_K$.*

The bilinear form can be written as $a(u, v) = \sum_{K \in \tau_h} a_K(u, v)$, with $u, v \in H_0^1(\Omega)$ and where $a_K(u, v) = \int_K \nabla u \cdot \nabla v dx$ following definition 5, for each $K \in \tau_h$. The polygon K is a simple polygon, thus it can be convex or non-convex. The general shape is one of the main characteristics of Virtual Element Method, making it more general than FEM in terms of discretization. The definition of simple polygons is given below:

Definition 8. *Simple polygons are simply connected sets in which the boundary is formed by straight line segments that do not intersect except at their ends.*

Accordingly to definition in Savarè and Chanon (2016), for each polygon K the following semi-norm is given:

$$|v|_{H_0^1(K)} = a_K(v, v)^{1/2} \tag{3.18}$$

and the following norm is given:

$$\|v\|_{H_0^1(\Omega)} = \left(\sum_{K \in \tau_h} |v|^2_{H_0^1(K)} \right)^{1/2}, \tag{3.19}$$

for all $v \in H_0^1(\Omega)$. It is important to mention that the norm is originated from the Sobolev space norm.

From the continuous problem it is possible to build the discrete version of it, given by: find $u_h \in V_h$ such that $a_h(u_h, v_h) = \langle f_h, v_h \rangle$, for all $v_h \in V_h$. Here, the goal is to build the virtual element space V_h, the bilinear form $a_h(u, v)$ and the load term $\langle f_h, v_h \rangle$. In da Veiga *et al.* (2013a) the following hypothesis are taken:

Hypothesis 1. *For each h, we have:*

1. $V_h \subset H_0^1(\Omega)$,
2. *a symmetric bilinear form $a_h : V_h \times V_h \longrightarrow \mathbb{R}$ and a bilinear form $a_{h,K} : V_{h,K,k} \times V_{h,K,k} \longrightarrow \mathbb{R}$ such that $a_h(u, v) = \sum_{K \in \tau_h} a_{h,K}(u, v)$, where $V_{h,K,k}$ is the local virtual element space, and*
3. *a load term $f_h \in V_h'$.*

Hypothesis 2. *Let $k \geq 1$ be an integer called order of accuracy such that, for all $K \in \tau_h$:*

1. *$\mathbb{P}_k(K) \subset V_{h,K,k}$, where $\mathbb{P}_k(K)$ is the polynomial space of degree k in K, $\mathbb{P}_{-1}(K) = \{0\}$ and $V_{h,K,k}$ is the local virtual element space,*
2. *k-consistency: it is true that $a_{h,K}(q, v) = a_K(q, v)$, for all $q \in \mathbb{P}_k(K)$ and for all $v_h \in V_{h,K,k}$,*
3. *stability: exists constants $C_1, C_2 \in \mathbb{R}_+$ that are independent of the polygonal diameter h and the polygon K such that $C_1 a_K(v, v) \leq a_{h,K}(v, v) \leq C_2 a_K(v, v)$, for all $v \in V_{h,K,k}$.*

It is worth mentioning that the first set of hypothesis gives the components necessary of the method and the second set of hypothesis gives the conditions for the method to work. The consistency and stability criterion have a fundamental part in the construction of the bilinear form. The stability criterion is be responsible to treat the non-polynomial functions inside the virtual element space. For the polynomial space, a scaled monomial basis, instead of the canonical one, is chosen and given by:

$$m_\alpha = \begin{cases} 1, & if \quad \alpha = 1 \\ \left(\frac{\mathbf{x} - \mathbf{x_c}}{h} \right)^\alpha, & if \quad \alpha > 1 \end{cases} \tag{3.20}$$

such that $\mathcal{M}_k(K) = \{m_\alpha : \ 0 \leq |\alpha| \leq k\}$ where $\mathbf{x} = (x, y)$, $\mathbf{x_c}$ is the centroid and h is the polygonal diameter. This choice will be important for the construction of the bilinear form. Also, the following hypothesis are made concerning to the mesh regularity:

Hypothesis 3. *Denoting by $l_{K,e}$ the length of an edge $e \in \partial K$:*

1. *there exits a real number $\gamma > 0$ such that all elements $K \in \tau_h$ are star-shaped with respect to a ball B_K with radius $R_K \geq \gamma h_K$ and center $\mathbf{x}_K$,*

2. *there exits a real number $\eta > 0$ such that for all elements $K \in \tau_h$ and all edges ∂K it is true that $l_{K,e} \geq \eta h_K$.*

The set of hypothesis 3 are later discussed in the Appendix B about mesh regularity. It is shown that is possible to make weaker restrictions concerning to mesh regularity and proving the robustness of the method before mesh distortions.

The next theorem is a consequence of Lax-Milgram Theorem as shown. The theorem that guarantees the uniqueness of the solution and its convergence is the following:

Theorem 2. *Under the set of hypothesis 1 and 2 mentioned above, it is true that:*

1. *the discrete problem has a unique solution,*

2. *with respect to the convergence, let u_h be the solution for the discrete problem, for all u_π that is piecewise in $\mathbb{P}_k(K)$ and for all $u_I \in V_h$ that is an approximation of u, $\|u - u_h\|_{H_0^1(\Omega)} \leq \tilde{C}(\tilde{F}_h + \|u - u_\pi\|_{H_0^1(\Omega)} + \|u - u_I\|_{H_0^1(\Omega)})$, where $\tilde{C}(C_1, C_2) \in \mathbb{R}$ and $\tilde{F}_h$ is the smallest constant such that $F(v) - \langle f_h, v\rangle \leq \tilde{F}_h \|v\|_{H_0^1(\Omega)}$, for all $v \in V_h$.*

Proof. The continuity of the discrete bilinear form a_h comes from the stability hypothesis. Using the Cauchy-Schwarz Inequality and the stability criterion, for all $u_h, v_h \in V_h$:

$$a_h(u_h, v_h) = \sum_{K\in\tau_h} a_{h,K}(u_h, v_h) \leq C_2 \sum_{K\in\tau_h} |u_h|_{H_0^1(K)} |v_h|_{H_0^1(K)} \leq$$

$$\leq C_2 \left(\sum_{K\in\tau_h} |u_h|^2_{H_0^1(K)} \right)^{1/2} \left(\sum_{K\in\tau_h} |v_h|^2_{H_0^1(K)} \right)^{1/2} = C_2 \|u_h\|_{H_0^1(\Omega)} \|v_h\|_{H_0^1(\Omega)}.$$

Then,

$$a_{h,K}(u_h, v_h) \leq C_2 \|u_h\|_{H_0^1(\Omega)} \|v_h\|_{H_0^1(\Omega)}. \tag{3.21}$$

Thus, from equation (3.21) it is possible to conclude that the bilinear form is a continuous operator. It also can be observed that:

$$a_h(v_h, v_h) = \sum_{K\in\tau_h} a_{h,K}(v_h, v_h) \geq C_1 \sum_{K\in\tau_h} a_K(v_h, v_h) = C_1 a(v_h, v_h) = C_1 \|v_h\|^2_{H_0^1(\Omega)} \tag{3.22}$$

From equation (3.22) it is possible to conclude that the bilinear form a_h is also coercive. As by definition, f_h is a continuous functional, then by the Lax-Milgram Theorem the discrete problem has a unique solution.

Defining $\xi_h = u_h - u_I$. Then, by the linearity of the operator:

$$a_h(u_h - u_I, \xi_h) = a_h(u_h, \xi_h) - a_h(u_I, \xi_h) \tag{3.23}$$

Using equation (3.23) in (3.22):

$$C_1\|\xi_h\|^2_{H_0^1(\Omega)} \leq a_h(u_h,\xi_h) - a_h(u_I,\xi_h) = \langle f_h, v_h\rangle - \sum_{K\in\tau_h} a_{h,K}(u_I,\xi_h) \tag{3.24}$$

Again, using the linearity of $a_{h,K}$:

$$a_{h,K}(u_I - u_\pi + u_\pi, \xi_h) = a_{h,K}(u_I - u_\pi, \xi_h) + a_{h,K}(u_\pi, \xi_h) \tag{3.25}$$

From (3.25) in (3.24) and using consistency criterion:

$$C_1\|\xi_h\|^2_{H_0^1(\Omega)} \leq \langle f_h, v_h\rangle - \sum_{K\in\tau_h} [a_{h,K}(u_I - u_\pi, \xi_h) + a_h(u_\pi, \xi_h)] = \tag{3.26}$$

$$= \langle f_h, v_h\rangle - \sum_{K\in\tau_h} [a_{h,K}(u_I - u_\pi, \xi_h) + a_K(u_\pi - u + u, \xi_h)] = \tag{3.27}$$

$$= \langle f_h, v_h\rangle - \sum_{K\in\tau_h} [a_{h,K}(u_I - u_\pi, \xi_h) + a_K(u_\pi - u, \xi_h)] - \sum_{K\in\tau_h} a_K(u, \xi_h). \tag{3.28}$$

Using $a_K(u,v) = \int_K \nabla u \cdot \nabla v dx$ and equation (3.10) in (3.28):

$$C_1\|\xi_h\|^2_{H_0^1(\Omega)} \leq \langle f_h, v_h\rangle - \sum_{K\in\tau_h} [a_{h,K}(u_I - u_\pi, \xi_h) + a_K(u_\pi - u, \xi_h)] - F(\xi_h) \leq \tag{3.29}$$

$$\leq |F(\xi_h) - \langle f_h, v_h\rangle| + \sum_{K\in\tau_h} [a_{h,K}(u_I - u_\pi, \xi_h) + a_K(u_\pi - u, \xi_h)] \leq \tag{3.30}$$

$$\leq \tilde{F}_h\|\xi_h\|_{H_0^1(\Omega)} - a_h(u_I - u_\pi, \xi_h) - a(u_\pi - u, \xi_h) \tag{3.31}$$

Due to the continuity of a_h and a:

$$C_1\|\xi_h\|^2_{H_0^1(\Omega)} \leq \|\xi_h\|_{H_0^1(\Omega)} \left(\tilde{F}_h + C_2\|u_I - u_\pi\|_{H_0^1(\Omega)} + \|u_\pi - u\|_{H_0^1(\Omega)}\right) \leq \tag{3.32}$$

$$\leq \|\xi_h\|_{H_0^1(\Omega)} \max\{C_2, 1\} \left(\tilde{F}_h + \|u_I - u_\pi\|_{H_0^1(\Omega)} + \|u_\pi - u\|_{H_0^1(\Omega)}\right) \tag{3.33}$$

Using the Triangular Inequality:

$$\|u - u_h\|_{H_0^1(\Omega)} \leq \left(\frac{\max\{C_2, 1\}}{C_1} + 1\right) \left(\tilde{F}_h + \|u_I - u_\pi\|_{H_0^1(\Omega)} + \|u_\pi - u\|_{H_0^1(\Omega)}\right). \tag{3.34}$$

□

Now, the virtual element space V_h shall be constructed and the degrees of freedom shall be chosen.

3.3 The virtual element space

To define the virtual element space, an adequate choice of the degrees of freedom must be done, as they define unique approximation functions. First, the definition below will provide

some notation.

Definition 9. *Let K be a simple polygon. Then, the number of vertices and edges are the same and it is denoted by n_K.*

In Virtual Element Method, the usage of any simple polygons leads to an approximation space that might contain functions that are not necessarily polynomials. Also, the behavior of this functions shall be specific in some parts of the domain. Thus, in the case of edges that composes the boundary ∂K of each polygon K, the following space of continuous function is defined:

Definition 10. *For each $k \geq 1$, $\mathbb{E}_k(\partial K) = \{v \in C^0(\partial K) : v|_e \in \mathbb{P}_k(e), \forall e \in \partial K\}$.*

Definition 10 states that the functions in the space $\mathbb{E}_k(\partial K)$ behaves like polynomials in the edges. Due to the choice of degrees of freedom that will be presented further, a polynomial function in ∂K will be determined by its values in the vertices and, for $k > 1$ also by its $k-1$ points in each edge. As result, the dimension o the space is given by:

$$\dim \mathbb{E}_k(\partial K) = n_K + n_K(k-1) = n_K k. \tag{3.35}$$

Take $u, v \in \mathbb{E}_k(\partial K)$ and a number $\beta \in \mathbb{R}$. By definition, u and v are continuous in the boundary implying that $u + \beta v$ is continuous in the boundary ∂K. For all $e \in \partial K$, $u|_e$ and $v|_e$ are polynomials of degree k. As result, $(u + \beta v)|_e = u|_e + \beta v|_e$ is a polynomial in the boundary ∂K. This shows that $\mathbb{E}_k(\partial K)$ is also linear.

Considering $k \geq 1$, the definition of the local virtual element space is given below.

Definition 11. *For $k \geq 1$, the local virtual element space regarding to polygon K is*

$$V_{h,K,k} = \{v \in H_0^1(K) : v|_{\partial K} \in \mathbb{E}_k(\partial K), \quad \Delta v|_K \in \mathbb{P}_{k-2}(K)\}. \tag{3.36}$$

By definition $\mathbb{P}_{-1}(K) = \{0\}$ (see Hypothesis 1). Then, if $k = 1$, $\Delta v|_K \equiv 0$ and $V_{h,K,1}$ is a space of harmonic functions that are linear on each edge of polygon K. In this sense, these functions are uniquely defined by its values in the vertices and, consequently, $\dim V_{h,K,1} = n_K$. In turn, if $k = 2$, the Laplacian for the functions in $V_{h,K,2}$ is constant and the polynomials in ∂K have degree less or equal than 2. For each constant $\omega \in \mathbb{R}$ and for all $t \in \mathbb{E}_2(\partial K)$, it is possible to setup the following problem:

$$\begin{cases} \Delta v = \omega, \quad in \quad K, \\ v = t, \quad in \quad \partial K \end{cases}. \tag{3.37}$$

Applying the Lax-Milgram Theorem it is possible to guarantee that equation (3.37) has a unique solution. Therefore, $v \in V_{k,K,2}$ is determined by it values in the vertices, in the middle point of edges and by an internal value that is consequence of equation (3.37). As result, it is possible to conclude that $\dim V_{h,K,2} = 2n_K + 1$.

In a more general way it is possible to determine the dimension of $V_{h,K,k}$ for every k. By Lax-Milgram Theorem, there exists a unique function $v \in H^1(K)$ such that

$$\begin{cases} \Delta v = q, \quad in \quad K, \\ v = t, \quad in \quad \partial K \end{cases}, \tag{3.38}$$

for all $q \in \mathbb{P}_{k-2}(K)$ and $t \in \mathbb{E}_k(\partial K)$. Thus,

$$\dim V_{h,K,k} = \dim \mathbb{E}_k(\partial K) + \dim \mathbb{P}_{k-2}(K) = n_K k + \binom{k}{k-2} = n_K k + \frac{k!}{(k-2)!2} \tag{3.39}$$

Resulting in:

$$\dim V_{h,K,k} = n_K k + \frac{k(k-1)}{2} \tag{3.40}$$

The total number of degrees of freedom that must be chosen in a way to represent all functions on that space is equal to the dimension of the local virtual element space $V_{h,K,k}$. For all, $v \in V_{h,K,k}$, the chosen degrees of freedom are:

- $\mathcal{V}_K$ = the values of v in each vertex of K,
- $\mathcal{E}_K$ = the values of v in the $k-1$ middle points of each edge of K and for $k > 1$,
- $\mathcal{P}_K$ = the values of v internal points with order up to $k-2$ of K and for $k > 1$.

The internal points values are called moments and are given by:

$$i_K(v) = \frac{1}{|K|} \int_K m(\mathbf{x}) v(\mathbf{x}) dx, \quad \forall m \in \mathcal{M}_{k-2}(K), \tag{3.41}$$

where $|K|$ is the area of polygon K. As it was discussed before it is observable that:

$$\dim V_{h,K,k} = N_{dof} = n_K k + \frac{k(k-1)}{2}, \tag{3.42}$$

where N_{dof} is the total number of degrees of freedom regarding to polygon K. This is justified once the number of degrees of freedom in the set $\mathcal{V}_K$ is equal to the number of vertices n_K, the number of degrees of freedom in $\mathcal{E}_K$ is equal to $n_K(k-1)$ and, as $\mathcal{P}_K$ is directly related with $\mathbb{P}_{k-2}(K)$, the number of degrees in it is $\binom{k}{k-2}$.

The next necessary step is to show that a set of degrees of freedom $\mathcal{V}_K \cup \mathcal{E}_K \cup \mathcal{P}_K$ determines a unique function $v \in V_{h,K,k}$. Regarding to the choice of the degrees of freedom and the definition of the virtual element space, the set $\mathcal{V}_K \cup \mathcal{E}_K$ are related to a polynomial of degree less or equal to k in boundary of the polygonal element. In turn, the set of degrees of freedom $\mathcal{P}_K$ determines the projection of v in $\mathbb{P}_{k-2}(K)$ through L^2-norm. This projection will be denoted by $L^K_{k-2}v$, for each $v \in V_{h,K,k}$.

Theorem 3. *The set of degrees of freedom $\mathcal{V}_K \cup \mathcal{E}_K \cup \mathcal{P}_K$ is unisolvent.*

Proof. Given a function $v \in V_{h,K,k}$, the goal is to prove for all $K \in \tau_h$ that:

$$\begin{cases} v = 0, \quad in \quad \partial K, \\ L^K_{k-2} v = 0, \quad in \quad K \end{cases} . \tag{3.43}$$

In other words, the objective is to show that the operator that associates the degrees of freedom with the function v is injective. To prove that $v = 0$ in ∂K it is enough to show that $\Delta v = 0$ in K. After solving

$$\begin{cases} \Delta v = 0, \quad in \quad K, \\ v = 0, \quad in \quad \partial K \end{cases} , \tag{3.44}$$

it is possible to conclude that $v \equiv 0$ is the unique solution.

The second part can be proved by solving the following problem: for all $p \in \mathbb{P}_{k-2}(K)$, find $u \in H^1_0(K)$ such that

$$\begin{cases} -\Delta u = p, \quad in \quad K, \\ u = 0, \quad in \quad \partial K \end{cases} . \tag{3.45}$$

By Lax-Milgram Theorem, this problem has solution and its unique. This solution can be written as $u = \Delta^{-1} p$. Considering $T : \mathbb{P}_{k-2}(K) \longrightarrow \mathbb{P}_{k-2}(K)$ such that

$$T(p) = L^K_{k-2}(\Delta^{-1} p) = L^K_{k-2} u, \tag{3.46}$$

for all $p \in \mathbb{P}_{k-2}(K)$, it is true that:

$$\int_K p \cdot T(p) dK = \int_K p \cdot L^K_{k-2} u dK = \int_K p \cdot u dK = a_K(u, u) \tag{3.47}$$

As $u \in H^1_0(K)$:

$$T(p) = 0 \Leftrightarrow p = 0 \tag{3.48}$$

It is true that:

$$L^K_{k-2} v = L^K_{k-2} \left[-\Delta^{-1}(-\Delta v) \right] = T(-\Delta v), \tag{3.49}$$

because $\Delta v \in \mathbb{P}_{k-2}(K)$ and if $v = 0$ in ∂K, it implies that $v \in H^1_0(K)$. Finally,

$$L^K_{k-2} v = 0 \Rightarrow T(-\Delta v) = 0. \tag{3.50}$$

Then, $\Delta v = 0$ and, consequently, the mapping is injective. □

By the construction of the local virtual element space, the global one is given as a union of each local space.

Definition 12. *For $k \geq 1$ and for all $K \in \tau_h$*

$$V_h = \bigcup_{K \in \tau_h} V_{h,K,k} = \left\{ v \in H_0^1(\Omega) : v|_{\partial K} \in \mathbb{E}_k(\partial K), \quad \Delta v_K \in P_{k-2}(K), \quad \forall K \in \tau_h \right\} \tag{3.51}$$

Using the same arguments used to deduct the dimension of local space, it results that the dimension of the global space is given by:

$$\dim V_h = N_{vert} + N_{edge}(k-1) + N_{el}\frac{k(k-1)}{2}, \tag{3.52}$$

where N_{vert} is the total number of vertices, N_{edge} is the total number of edges and N_{el} is the total number of elements. The sets of degrees of freedom are also very similar to what was done for the local spaces, for $v \in V_h$:

- $\mathcal{V}$ = the values of v in each vertex,
- $\mathcal{E}$ = the values of v in the $k-1$ middle points of each edge for $k > 1$,
- $\mathcal{P}$ = the values of v moments with order up to $k-2$ for $k > 1$.

Analogously to the local case, the number of degrees of freedom coincides with the dimension of V_h. Also, Theorem 3 can be extended to the global case. Thus, the degrees of freedom are unisolvent in V_h. The next step is the construction of the bilinear form.

3.4 The bilinear form

Prior to the bilinear form presentation, the projection operator must be introduced. The projection operator is responsible for projecting components of the virtual element space into the the polynomial space and directly treating the non-polynomial functions. It is important to recall that the polynomial space is contained in the virtual element space (see set of Hypothesis 2). Considering the operator Π^∇, the virtual element space can be seen as:

$$V_{h,K,k} = [polynomial] + [non-polynomial] = \Pi^\nabla(V_{h,K,k}) + (1 - \Pi^\nabla)(V_{h,K,k}).$$

In this sense, the shape functions of Virtual Element Method can be computed implicitly and direct from the degrees of freedom.

The classical choice for the projection operator Π^∇ is:

Definition 13. *Define $\Pi^\nabla : V_{h,K,k} \longrightarrow \mathbb{P}_k(K)$ such that*

$$\int_K \nabla p \cdot \nabla(\Pi^\nabla v - v) dK = 0, \quad \forall p \in \mathbb{P}_k(K) \tag{3.53}$$

and $P_0 : \Pi^{\nabla} : V_{h,K,k} \longrightarrow \mathbb{P}_0(K)$ *such that*

$$\begin{cases} P_0 v = \frac{1}{n_K} \sum_{i=1}^{n_K} v(V_i) = \frac{1}{n_K} \sum_{i=1}^{n_K} \Pi^{\nabla} v(V_i), & k = 1; \\ P_0 v = \frac{1}{|K|} \int_K v dK = \frac{1}{|K|} \int_K \Pi^{\nabla} v dK, & k \geq 2, \end{cases} \tag{3.54}$$

where V_i *is the i-th vertex of polygon* K *and for all* $v \in V_{h,K,k}$. P_0 *is the projection operator in the constant polynomial space* $\mathbb{P}_0(K)$.

Here, it is worth mentioning that other choice for the constant projection operator can be made. For example, in da Veiga *et al.* (2017b) the constant projection operator is defined as:

$$P_0 v = \frac{1}{|\partial K|} \int_{\partial K} v dS, \tag{3.55}$$

where $|\partial K|$ is the perimeter of K. In this work, definition 13 is used. It is important to notice that if $p \in \mathbb{P}_k(K)$ then it is natural that $\Pi^{\nabla} p = p$.

Fixing $k \geq 1$, for all $K \in \tau_h$, for all $v \in V_{h,K,k}$ and for all $q \in \mathbb{P}_k(K)$, using integration by parts:

$$a_K(q, v) = \int_K \nabla q \cdot \nabla v dK = -\int_K \Delta q v dK + \int_{\partial K} \frac{\partial q}{\partial \eta} v dS. \tag{3.56}$$

As stated in Savarè and Chanon (2016), analyzing the first term of equation (3.56), Δq can be written in terms of $\mathcal{M}_{k-2}(K)$ that is a basis of scaled monomials for $\mathbb{P}_{k-2}(K)$ (see equation (3.20) regarding to monomial basis) since $\Delta q \in \mathbb{P}_{k-2}(K)$. Thus, this integral is a linear combination of polynomials with the chosen degrees of freedom and, consequently, it can be computed exactly. The second term of the integral is composed by $\frac{\partial q}{\partial \eta} \in \mathbb{P}_{k-1}(e)$ and $v \in \mathbb{P}_k(e)$, with the edge $e \in \partial K$, that are all polynomials and the values of v are known in the edges. As result, the second term of the integral can also be computed exactly. Finally, it is possible to compute $a_K(q, v)$ exactly for any $K \in \tau_h$, $q \in \mathbb{P}_k(K)$ and $v \in V_{h,K,k}$.

Then, the obvious choice for the bilinear form would be $a_{h,K}(u, v) = a_K(\Pi^{\nabla}, \Pi^{\nabla} v)$, for all $u, v \in V_{h,K,k}$. Although, this choice would only satisfy the consistency criterion presented in Hypothesis 2 and not the stability criterion because, due to a deviation included by the projection, it would not be possible find the stability constants C_1 and C_2. Therefore, a term must be added to guarantee stability. Defining the symmetrical bilinear form $S_K(u, v)$ such that

$$C_3 a_K(v, v) \leq S_K(u, v) \leq C_4 a_K(v, v), \tag{3.57}$$

where $C_3, C_4 \geq 0$ are independent of K and h_k, it is possible to enunciate the following theorem:

Theorem 4. *Given the stability term as in equation (3.57), if for all* $u, v \in V_{h,K,k}$, *the*

discrete bilinear form is defined by

$$a_{h,K}(u,v) = a_K(\Pi^\nabla u, \Pi^\nabla v) + S_K(u - \Pi^\nabla u, v - \Pi^\nabla v), \tag{3.58}$$

then $a_{h,K}$ satisfies the consistency and the stability criteria.

Proof. For all $q \in \mathbb{P}_k(K)$ and by the definition of the projection operator Π^∇, it is true that:

$$S_K(q - \Pi^\nabla q, v - \Pi^\nabla v) = 0, \tag{3.59}$$

for all $v \in V_{h,K,k}$. Then consistency is given by:

$$a_{h,K}(q,v) = a_K(\Pi^\nabla q, \Pi^\nabla v) = a_K(q, \Pi^\nabla v) = a_K(q,v). \tag{3.60}$$

For all $v \in V_{h,K,k}$, it is true that $\Pi^\nabla \left(\Pi^\nabla v - v\right) = 0$. Thus,

$$\begin{aligned} a_{h,K}(v,v) &\leq a_K(\Pi^\nabla v, \Pi^\nabla v) + C_4 a_K(v - \Pi^\nabla v, v - \Pi^\nabla v) \leq \\ &\leq \max\{1, C_4\} \left[a_K(\Pi^\nabla v, \Pi^\nabla v) + a_K(v - \Pi^\nabla v, v - \Pi^\nabla v)\right]. \end{aligned} \tag{3.61}$$

Analogously,

$$a_{h,K}(v,v) \geq \min\{1, C_3\} \left[a_K(\Pi^\nabla v, \Pi^\nabla v) + a_K(v - \Pi^\nabla v, v - \Pi^\nabla v)\right]. \tag{3.62}$$

As result, stability criterion is satisfied. □

The bilinear form $S_K : V_{h,K,k} \times V_{h,K,k} \longrightarrow \mathbb{R}$ must be asymptotic in order to guarantee its behavior when problems, like mesh distortion, occurs. In this sense, the stiffness matrix must continue stable even when parameters are changed. Some choices for S_K can be made, like in Wriggers *et al.* (2016), the stability term is given by:

$$S_K(u,v) = h_K \int_{\partial K} \frac{\partial u}{\partial s} \frac{\partial v}{\partial s} ds, \tag{3.63}$$

for all $u, v \in V_{h,K,k}$. However, in this work the classical choice presented in da Veiga *et al.* (2013a) and da Veiga *et al.* (2014) is used:

$$S_K(u,v) = \sum_{p=1}^{N_{dof}} dof_p(u) dof_p(v), \tag{3.64}$$

where $dof_i : V_{h,K,k} \longrightarrow \mathbb{R}$ is the application that gives the value of function v in the i-th degree of freedom and N_{dof} is the total number of degrees of freedom. Defining the basis $(\phi_i)_{i \in [1, N_{dof}]}$ for $V_{h,K,k}$, the application dof has the Kronecker property: $dof_i(\phi_j) = \delta_{ij}$. Therefore, given $v \in V_{h,K,k}$ at this point it is possible to write $v = \sum_{i=1}^{N_{dof}} dof_i(v)\phi_i$.

It is possible to observe that equation (3.64) satisfy the stability criterion. Rewriting the stability term as $S_K(v,v) = \sum_{i=1}^{N_{dof}} dof_i(v)dof_i(v) = \sum_{i=1}^{N_{dof}} dof_i(v)^2$. Also, the consistency term can be written as $a_K(v,v) = \sum_{i=1}^{N_{dof}} v_i^2 a_K(\phi_i,\phi_i)$. Therefore,

$$a_K(v,v) \leq \max_i\{a_K(\phi_i,\phi_i)\}\sum_{i=1}^{N_{dof}} dof_i(v)^2 = \frac{1}{C_4}S_K(v,v) \tag{3.65}$$

and

$$a_K(v,v) \geq \min_i\{a_K(\phi_i,\phi_i)\}\sum_{i=1}^{N_{dof}} dof_i(v)^2 = \frac{1}{C_3}S_K(v,v), \tag{3.66}$$

where $C_3 = \frac{1}{\min_i\{a_K(\phi_i,\phi_i)\}}$ and $C_4 = \frac{1}{\max_i\{a_K(\phi_i,\phi_i)\}}$.

3.5 Construction of the load term

To obtain the load term as given in equation (3.10), one can use the L^2-norm to the polynomial space. This construction can be divided in two cases. The first case regards to $k = 1$ which f_h is piecewise constant. Recalling that the operator L_0^K projects functions using $L^2(K)$-norm into the constant polynomial space $\mathbb{P}_0(K)$. Thus, it possible to give the following definition:

Definition 14. *For $k = 1$,*

$$\langle f_h, v\rangle = \sum_{K\in\tau_h} L_0^K f \frac{1}{n_K}\sum_{i=1}^{n_K} v(V_i), \tag{3.67}$$

where $\{V_i\}_{i\in[0,n_K]}$ is the set of vertices of polygon K.

An analogous idea is used for $k \geq 2$ case. However, here the projection operator L_{k-2}^K maps to $\mathbb{P}_{k-2}(K)$.

Definition 15. *For $k \geq 2$,*

$$\langle f_h, v\rangle = \sum_{K\in\tau_h}\int_K f_h v dK = \sum_{K\in\tau_h}\int_K (L_{k-2}^K f)v dK. \tag{3.68}$$

It can be seen that the load term for $k \geq 2$ can be written as a linear combination of the moments given by equation 3.41. As result, it can be computed from the chosen degrees of freedom.

This concludes the construction of the VEM for the Poisson equation. Implementation aspects will be discussed in Chapter 4, in which a methodology about the construction of the method for the Poisson Equation is shown in more details. First the virtual element

space is constructed and then the degrees of freedom are chosen. With the adequate degrees of freedom, the bilinear form is constructed taking in consideration the consistency and stability criteria and choosing an adequate basis for the virtual element space. Finally, the load term is constructed considering the idea of projecting L^2 functions into polynomial spaces.

Chapter 4

Implementation of Virtual Element Method for Poisson Equation

This chapter is dedicated to present the implementation framework for the Poisson Equation. The first section presents the construction of the stiffness matrix using the projection operator and the scaled monomial basis. Next, the construction of the load term is presented using the L^2-projection operator. Finally, some numerical results are presented regarding to a problem with analytical solution using two types of meshes and different number of elements.

4.1 Construction of the stiffness matrix

The next two sections are based on the works of da Veiga *et al.* (2014) and Sutton (2016). The same matrix framework is used here aiming to write the stiffness matrix showing some intermediary matrices that are easier to compute directly. For that, the projection operator and the load vector will be used.

Recalling that the bilinear form in definition 5 can be written in terms of the inner product as in equation (3.11). The projection operator in definition 13 can be written as:

$$\left(\nabla q, \nabla(\Pi^{\nabla} v - v)\right)_K = \int_K \nabla q \cdot \nabla(\Pi^{\nabla} v - v) dK = 0, \tag{4.1}$$

for all $q \in \mathbb{P}_k(K)$. Using the scaled monomial basis $\mathcal{M}_k(K)$ of $\mathbb{P}_k(K)$ and denoting by n_p the dimension of the polynomial space $\mathbb{P}_k(K)$, it is possible to write the projection of v in terms of scaled monomials:

$$\Pi^{\nabla} v = \sum_{\beta=1}^{n_p} r_\beta m_\beta. \tag{4.2}$$

As $q \in \mathbb{P}_k(K)$, it also can be written in terms of scaled monomials. Thus,

$$\begin{aligned}(\nabla m_\alpha, \nabla(\Pi^\nabla v - v))_K = 0 &\Rightarrow (\nabla m_\alpha, \nabla\Pi^\nabla v - \nabla v)_K = 0 \\ &\Rightarrow (\nabla m_\alpha, \nabla\Pi^\nabla v)_K - (\nabla m_\alpha, \nabla v)_K = 0 \\ &\Rightarrow \sum_{\beta=1}^{n_p} r_\beta (\nabla m_\alpha, \nabla m_\beta)_K = (\nabla m_\alpha, \nabla v)_K.\end{aligned} \tag{4.3}$$

In the case that $q \in \mathbb{P}_0(K)$, the procedure is similar:

$$P_0(\Pi^\nabla v - v) = 0 \Rightarrow P_0\Pi^\nabla v = P_0 v \Rightarrow \sum_{\beta=1}^{n_p} P_0 r_\beta m_\beta = P_0 v. \tag{4.4}$$

Rewriting equations (4.3) and (4.4) as a system of equations:

$$\underline{\mathbf{G}}\mathbf{r} = \mathbf{b}, \tag{4.5}$$

where

$$\underline{\mathbf{G}} = \begin{bmatrix} P_0 m_1 & P_0 m_2 & \dots & P_0 m_{n_p} \\ 0 & (\nabla m_2, \nabla m_2)_K & \dots & (\nabla m_2, \nabla m_{n_p})_K \\ \vdots & \vdots & \ddots & \vdots \\ 0 & (\nabla m_{n_p}, \nabla m_2)_K & \dots & (\nabla m_{n_p}, \nabla m_{n_p})_K \end{bmatrix}, \tag{4.6}$$

$$\mathbf{r} = \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_{n_p} \end{bmatrix} \quad \textit{and} \quad \mathbf{b} = \begin{bmatrix} P_0 v \\ (\nabla m_2, \nabla v)_K \\ \vdots \\ (\nabla m_{n_p}, \nabla v)_K \end{bmatrix}. \tag{4.7}$$

Using the basis $(\phi_i)_{i\in[1,N_{dof}]}$ for $V_{h,K,k}$ and writing $v = \sum_{i=1}^{N_{dof}} dof_i(v)\phi_i$ as in Chapter 3. It is possible to write:

$$\mathbf{r} = \begin{bmatrix} r_{1,i} \\ r_{2,i} \\ \vdots \\ r_{n_p,i} \end{bmatrix} \quad \textit{and} \quad \mathbf{b} = \begin{bmatrix} P_0 v \\ (\nabla m_2, \nabla \phi_i)_K \\ \vdots \\ (\nabla m_{n_p}, \nabla \phi_i)_K \end{bmatrix}, \tag{4.8}$$

for $i = 1, 2, ..., N_{dof}$. In order to consider and compute all degrees of freedom at once, the matrix $\underline{\mathbf{B}}$ is given by:

$$\underline{\mathbf{B}} = \begin{bmatrix} P_0\phi_1 & P_0\phi_1 & \dots & P_0\phi_{N_{dof}} \\ (\nabla m_2, \nabla\phi_1)_K & (\nabla m_2, \nabla\phi_2)_K & \dots & (\nabla m_2, \nabla\phi_{N_{dof}})_K \\ \vdots & \vdots & \ddots & \vdots \\ (\nabla m_{n_p}, \nabla\phi_1)_K & (\nabla m_{n_p}, \nabla\phi_1)_K & \dots & (\nabla m_{n_p}, \nabla\phi_{N_{dof}})_K \end{bmatrix}. \tag{4.9}$$

The matrix $\underline{\mathbf{B}}$ has dimension $n_p \times N_{dof}$. In this sense, the matrix representation of the projection operator presented in definition 13 is given by:

$$\underline{\mathbf{\Pi}}^{\nabla} = \underline{\mathbf{G}}^{-1}\underline{\mathbf{B}}. \tag{4.10}$$

This representation is directly related to handle with the polynomial function in virtual element space, more specifically it is related to the consistency term. Now, it is also necessary to define the matrix form of the projection operator in order to handle the stability term. Recalling from the first item in the set of hypothesis 2, the polynomial space is contained in $V_{h,K,k}$. The idea is to construct a extended projection operator from $V_{h,K,k}$ to $V_{h,K,k}$. Using the basis choices discussed earlier, it possible to define:

$$\Pi^{\nabla}\phi_i = \sum_{\beta=1}^{n_p} r_{\beta,i} \sum_{j=1}^{N_{dof}} dof_j(m_\beta)\phi_j = \sum_{j=1}^{N_{dof}} d_{ij}\phi_j, \tag{4.11}$$

where

$$d_{ij} = \sum_{\beta=1}^{n_p} r_{\beta,i} dof_j(m_\beta). \tag{4.12}$$

Thus, it is possible to define the matrix $\underline{\mathbf{D}}$ as:

$$\underline{\mathbf{D}} = \begin{bmatrix} dof_1(m_1) & dof_1(m_2) & \cdots & dof_1(m_{n_p}) \\ dof_2(m_1) & dof_2(m_2) & \cdots & dof_2(m_{n_p}) \\ \vdots & \vdots & \ddots & \vdots \\ dof_{N_{dof}}(m_1) & dof_{N_{dof}}(m_2) & \cdots & dof_{N_{dof}}(m_{n_p}) \end{bmatrix}. \tag{4.13}$$

The matrix $\underline{\mathbf{D}}$ has dimension $N_{dof} \times n_p$. From equation (4.13) in (4.12), the extended projection operator in matrix form can be defined by:

$$\underline{\mathbf{\Pi}}^{\nabla}_{\dagger} = \underline{\mathbf{D}}\ \underline{\mathbf{G}}^{-1}\underline{\mathbf{B}} = \underline{\mathbf{D}}\ \underline{\mathbf{\Pi}}^{\nabla}. \tag{4.14}$$

Finally, with all this intermediary matrices defined, it is possible to assembly into the stiffness matrix. Writing the $u, v \in V_{h,K,k}$ in terms of the basis $(\phi_i)_{i\in[1,N_{dof}]}$ and substituting in the bilinear form presented in theorem 4:

$$a_{h,K}(\phi_i, \phi_j) = a_K(\Pi^{\nabla}\phi_i, \Pi^{\nabla}\phi_j) + S_K(\phi_i - \Pi^{\nabla}\phi_i, \phi_j - \Pi^{\nabla}\phi_j). \tag{4.15}$$

using the classical choice for the stability term presented in equation (3.64) the stiffness matrix is given by:

$$\underline{\mathbf{K}}_h = (\underline{\mathbf{\Pi}}^{\nabla})^t \mathbb{G} (\underline{\mathbf{\Pi}}^{\nabla}) + (\mathbf{I} - \underline{\mathbf{\Pi}}^{\nabla}_{\dagger})^t (\mathbf{I} - \underline{\mathbf{\Pi}}^{\nabla}_{\dagger}), \tag{4.16}$$

where $\mathbb{G}$ is the $\underline{\mathbf{G}}$ with the first row completed with zeros.

The construction of the load term is very similar to what was done to construct the

stiffness matrix. Instead of using the Π^{∇} operator the L^2 projection operator denoted by L_k^K will be used. As before, the following orthogonality property is used:

$$\left(\nabla q, \nabla(L_k^K v - v)\right)_K = 0. \tag{4.17}$$

Writing the projection in terms of scaled monomials as below:

$$L_k^K v = \sum_{\beta=1}^{n_p} w_\beta m_\beta, \tag{4.18}$$

as in equation 4.5, the follwoing system is obtained:

$$\underline{\mathbf{H}}\mathbf{w} = \mathbf{c}. \tag{4.19}$$

The $\mathbf{c}$ vector can be associated to a matrix $\underline{\mathbf{C}}$. If $\alpha \in [1, n_{p-2}]$, then

$$C_{\alpha,i} = (m_\alpha, \phi_i)_K, \tag{4.20}$$

for $i = 1, 2, ..., N_{dof}$. Otherwise, if $\alpha \in (n_{p-2}, n_p]$, then

$$\underline{\mathbf{C}} = \underline{\mathbf{H}}\ \underline{\mathbf{G}}^{-1}\underline{\mathbf{B}}. \tag{4.21}$$

Thus, the L^2-projection operator matrix is given by:

$$\underline{\mathbf{L}} = \underline{\mathbf{D}}\ \underline{\mathbf{H}}^{-1}\underline{\mathbf{C}} \tag{4.22}$$

With equation (4.22), the load term can be computed using the formulation presented in Chapter 3. A detailed approach to the linear case is introduced by Sutton (2016).

4.2 Analytical solution for the Poisson Equation

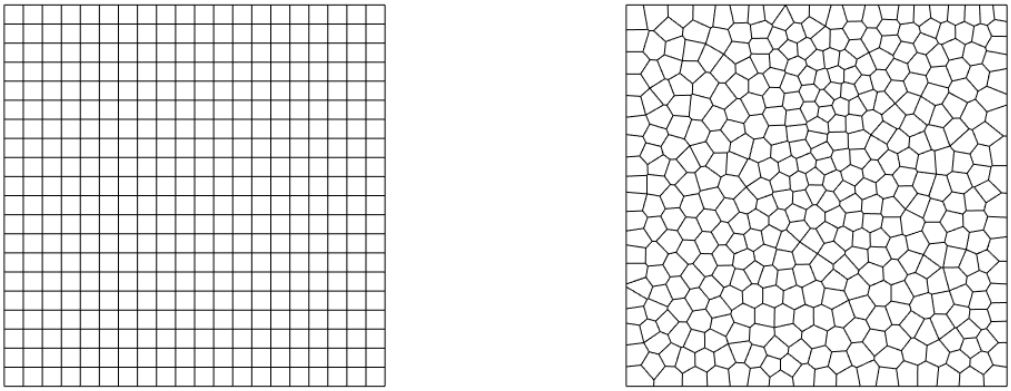
In this section, the Virtual Element Method is implemented for the Poisson Equation with a known analytical solution in a unitary square domain $\Omega = [0, 1] \times [0, 1]$. In this sense, the equation is given by:

$$\begin{cases} -\Delta u = \sin(\pi x_c)\sin(\pi y_c) \quad \text{in} \quad \Omega \\ u = 0 \quad \text{in} \quad \partial\Omega \end{cases}. \tag{4.23}$$

The analytical result for this equation:

$$u(\mathbf{x}) = -\frac{\sin(\pi x)\sin(\pi y)}{2\pi^2}, \tag{4.24}$$

where $\mathbf{x} = (x, y)$. To perform the simulations, the *Polymesher* was used to generate the Voronoi mesh. Also, a square uniform mesh was used. Both of them can be seen in Figure 4.1. A detailed discussion about the mesh generator *Polymehser* can be found in Talischi *et al.* (2012). It is important to mention that only the linear case of VEM will be implemented to this work. As can be oserved in Sutton (2016), Ortiz-Bernardin *et al.* (2019) and da Veiga *et al.* (2017b), the linear case is enough to make a very complete analysis of the method characteristics.



(a) *Uniform mesh* **(b)** *Voronoi mesh*

Figure 4.1: *Example of meshes used to perform simulations regarding to the Poisson Equation with analytical solution*

Source: Author

To evaluate the error, the L^2-norm, denoted by $\|\cdot\|_{L^2}$, was used. Thus,

$$e(u_h, u) = \|u - u_h\|_{L^2}, \tag{4.25}$$

where u is the analytical solution and u_h is the numerical solution. Table 4.1 shows the results obtained from the simulations for different number of elements. Figure 4.3 shows a

Poisson Equation with Analytical Soluition		
Elements	$e(u_h, u)$**-Uniform**	$e(u_h, u)$**-Voronoi**
16	3.10E-3	4.50E-3
36	1.90E-3	3.80E-3
64	1.40E-3	2.50E-3
144	8.84E-3	1.80E-3
256	6.58E-4	1.50E-3
400	5.24E-4	1.30E-3
1024	3.26E-4	7.94E-4
2704	2.01E-4	5.31E-4
4096	1.63E-4	4.71E-4

Table 4.1: *Associated errors for different number of elements.*

graphical representation of data in Table 4.1. It is possible to observe a faster convergence

of the method when more elements are used, as expected. The idea here is not to compare performance between the meshes but to illustrate VEM main characteristics. For different meshes, even with the same number of elements, performance cannot be compared once the number of degrees of freedom is different. The analytical solution plot can be seen in figure 4.3 and the plot of numerical solution for 16, 400 and 4096 elements regarding to the uniform mesh is presented in Figures 4.4, 4.5 and 4.6, respectively.

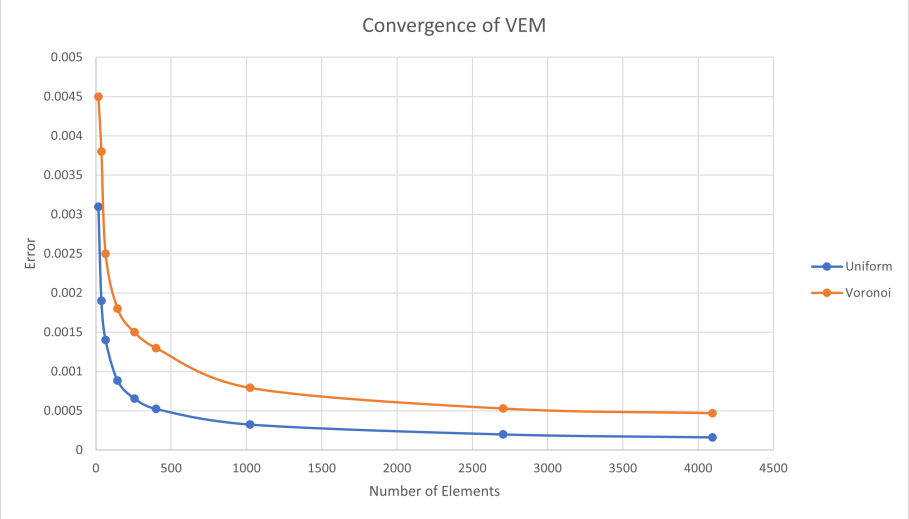


Figure 4.2: *Convergence of VEM for Poisson Equation with analytical solution.*
Source:Author

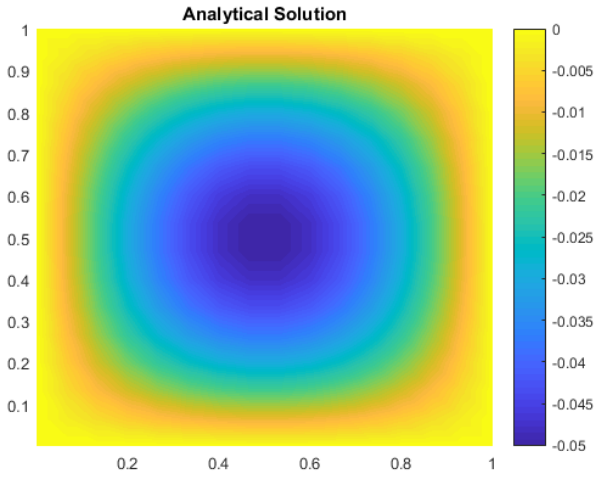


Figure 4.3: *Analytical solution of Poisson Equation 4.23.*
Source:Author

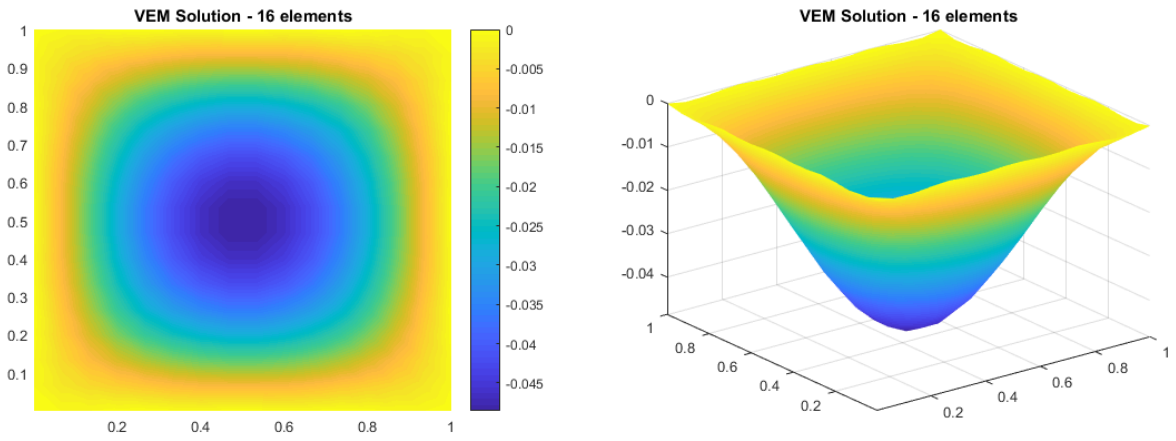


Figure 4.4: *Numerical solution using 16 elements*
Source: Author

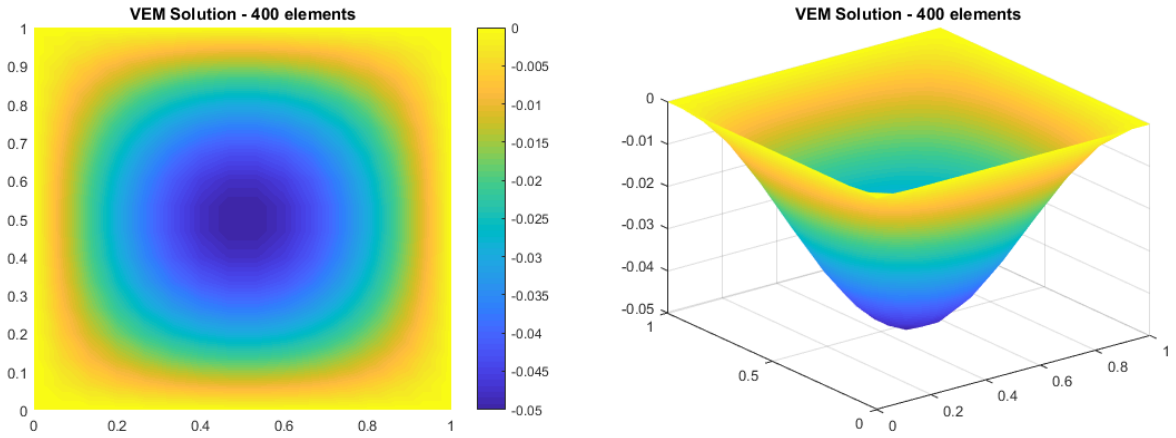


Figure 4.5: *Numerical solution using 400 elements*
Source: Author

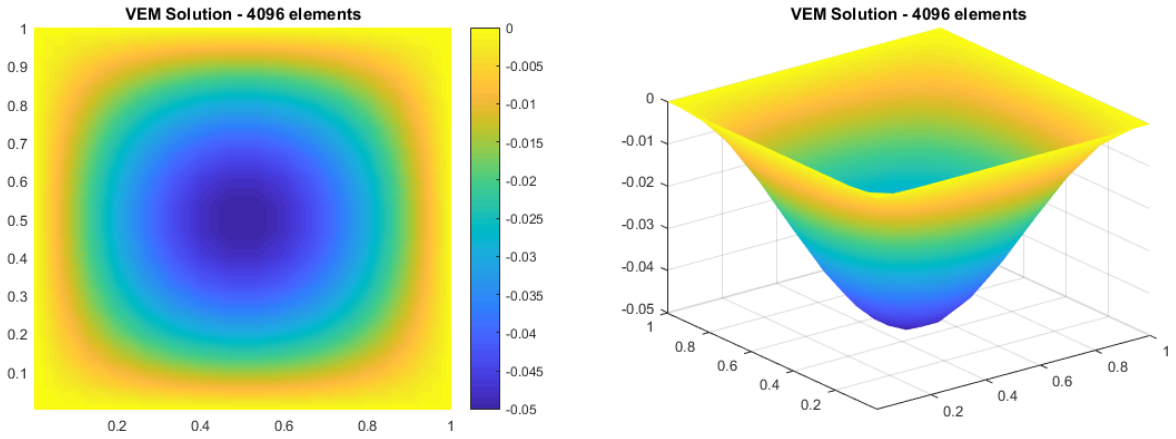


Figure 4.6: *Numerical solution using 4096 elements*
Source: Author

It is possible to conclude that even with few elements, VEM shows convergence with order of magnitude 10^{-3}. One advantage of VEM before FEM regarding to the Voronoi mesh is that no isoparametric elements are needed when five or six sided convex polygons are present in the mesh. Thus, no transformation and no Jacobian matrix calculation is needed, once Virtual Element Method can compute these elements directly as showed in Chapter 3. In this section, the Poisson Equation with analytical solution was presented to verify the convergence of the method.

Chapter 5

Theory of Elasticity

On this chapter, the Theory of Elasticity is presented. The main goal here is to show the mathematical formulation regarding to finite elasticity. Then, this formulation is particularized to the linear elasticity theory of which Virtual Element Method will be applied. The basic references for this chapter are the canonical work of Timoshenko and Goodier (1951) and the work of Young and Budynas (2002). Also, it is used as support text the work of Bucalem and Bathe (2011).

Accordingly to Timoshenko and Goodier (1951), in general the materials have the elastic property. That means, there are external forces acting on a solid body causing deformation and, if the force does no exceed a established limit when it is removed the deformation disappears in part. In this sense, the first part is dedicated to study the finite elasticity formulation, discussing the kinematics and stress analysis. The next step refers to make hypothesis to introduce the linear elastic theory using the Generalized Hooke's Law as constitutive equation.

5.1 Displacement and Deformation in Elasticity

The configuration of a solid body refers to the portion occupied by this solid in an instant of time t_0. The main goal here is to write the deformed configuration $\mathbb{V}$ using a reference configuration $\mathbb{V}_0$ in an instant t. The formulation is developed using the classical Euclidean space $\mathbb{R}^3$ and the canonical basis related to this space is $(\mathbf{e_1}, \mathbf{e_2}, \mathbf{e_3})$. Figure 5.1 shows a generic solid body in the reference configuration and in the deformed configuration after suffering deformation.

Considering a vector $\mathbf{x}$ in the Euclidean space and the canonical basis defined earlier, the following notation will be used:

$$\mathbf{x} = x_1\mathbf{e_1} + x_2\mathbf{e_2} + x_3\mathbf{e_3} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \tag{5.1}$$

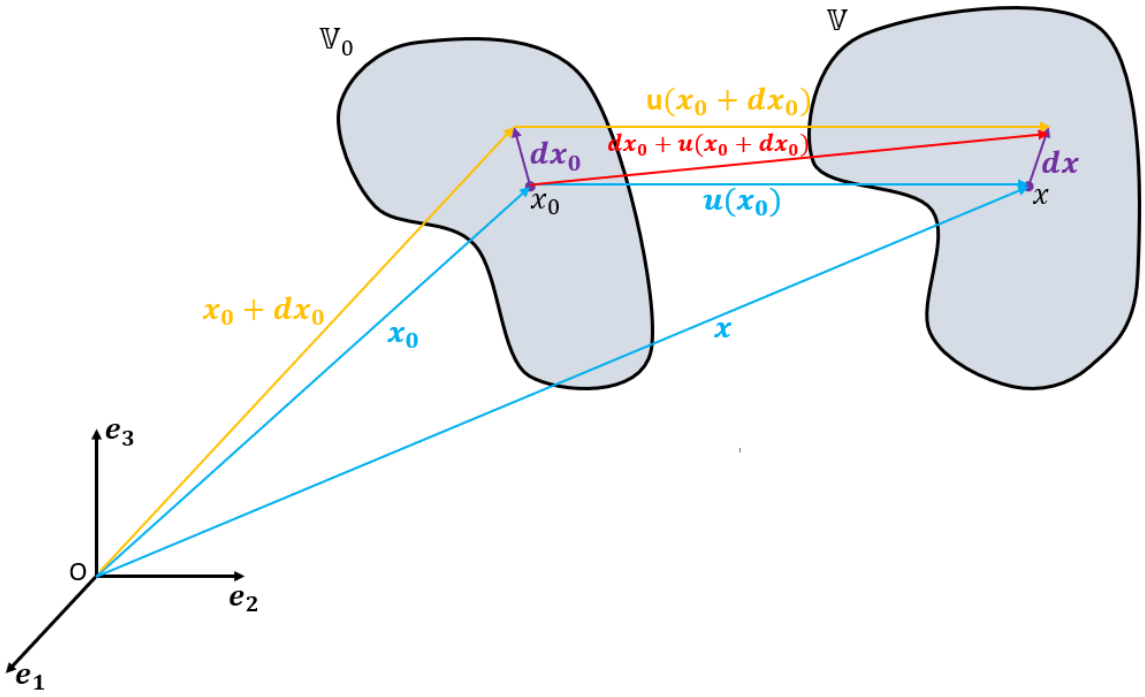


Figure 5.1: *Generic solid body in reference configuration and deformed configuration.*
Source: Author

where $x_i \in \mathbb{R}$ with $i = 1, 2, 3$ are called components. To keep the notation as clean as possible, the components concerning to the reference configuration is denoted by x_i^0, for each $i = 1, 2, 3$. The associated norm to the Euclidean space is given by:

$$\|\mathbf{x}\| = \sqrt{x_1^2 + x_2^2 + x_3^2}. \tag{5.2}$$

According to Bucalem and Bathe (2011), the normal strain depends on the fiber direction and location. Considering the fiber $\mathbf{dx_0}$ in $\mathbb{V}_0$ and the fiber $\mathbf{dx}$ in $\mathbb{V}$, it is possible to write:

$$\mathbf{dx} = \mathbf{dx_0} + \mathbf{u}(\mathbf{x_0} + \mathbf{dx_0}) - \mathbf{u}(\mathbf{x_0}). \tag{5.3}$$

Expanding $u_i(x_1 + dx_1, x_2 + dx_2, x_3 + dx_3)$ in terms of its derivatives (first order Taylor series), it holds true for each $i = 1, 2, 3$ that:

$$\begin{aligned} u_i(x_1^0 + dx_1, x_2^0 + dx_2, x_3^0 + dx_3) - u_i(x_1^0, x_2^0, x_3^0) = \\ = dx_1 \frac{\partial u_i}{\partial x_1^0}(x_1^0, x_2^0, x_3^0) + dx_1 \frac{\partial u_i}{\partial x_2^0}(x_1^0, x_2^0, x_3^0) + dx_3 \frac{\partial u_i}{\partial x_3^0}(x_1^0, x_2^0, x_3^0). \end{aligned} \tag{5.4}$$

Substituting equation (5.4) in (5.3):

$$\begin{bmatrix} dx_1 \\ dx_2 \\ dx_3 \end{bmatrix} = \begin{bmatrix} dx_1^0 \\ dx_2^0 \\ dx_3^0 \end{bmatrix} + \begin{bmatrix} \frac{\partial u_1}{\partial x_1^0} & \frac{\partial u_1}{\partial x_2^0} & \frac{\partial u_1}{\partial x_3^0} \\ \frac{\partial u_2}{\partial x_1^0} & \frac{\partial u_2}{\partial x_2^0} & \frac{\partial u_2}{\partial x_3^0} \\ \frac{\partial u_3}{\partial x_1^0} & \frac{\partial u_3}{\partial x_2^0} & \frac{\partial u_3}{\partial x_3^0} \end{bmatrix} \begin{bmatrix} dx_1^0 \\ dx_2^0 \\ dx_3^0 \end{bmatrix} \tag{5.5}$$

with

$$\nabla \mathbf{u} = \begin{bmatrix} \frac{\partial u_1}{\partial x_1^0} & \frac{\partial u_1}{\partial x_2^0} & \frac{\partial u_1}{\partial x_3^0} \\ \frac{\partial u_2}{\partial x_1^0} & \frac{\partial u_2}{\partial x_2^0} & \frac{\partial u_2}{\partial x_3^0} \\ \frac{\partial u_3}{\partial x_1^0} & \frac{\partial u_3}{\partial x_2^0} & \frac{\partial u_3}{\partial x_3^0} \end{bmatrix} \tag{5.6}$$

is called displacement gradient. Thus, it is possible to define the deformation gradient as:

$$\underline{\mathbf{F}} = \mathbf{I} + \nabla \mathbf{u} = \begin{bmatrix} \frac{\partial x_1}{\partial x_1^0} & \frac{\partial x_1}{\partial x_2^0} & \frac{\partial x_1}{\partial x_3^0} \\ \frac{\partial x_2}{\partial x_1^0} & \frac{\partial x_2}{\partial x_2^0} & \frac{\partial x_2}{\partial x_3^0} \\ \frac{\partial x_3}{\partial x_1^0} & \frac{\partial x_3}{\partial x_2^0} & \frac{\partial x_3}{\partial x_3^0} \end{bmatrix}. \tag{5.7}$$

The length of the reference fiber and the length of the fiber in current configuration are defined, respectively, by:

$$dS^0 = \|\mathbf{dx_0}\| \quad and \quad dS = \|\mathbf{dx}\|. \tag{5.8}$$

The strain formula can be calculated using the quadratic formula given by:

$$\varepsilon_q = \frac{1}{2}\frac{(dS)^2 - (dS^0)^2}{(dS^0)^2} = \frac{1}{2}(\lambda^2 - 1), \tag{5.9}$$

where $\lambda = \frac{dS}{dS^0}$ is the stretch. Substituting equation (5.8) in (5.9):

$$\varepsilon_q = \frac{1}{2}\frac{\mathbf{dx_0} \cdot \underline{\mathbf{F}}^T\underline{\mathbf{F}}\mathbf{dx_0} - \mathbf{dx_0} \cdot \mathbf{dx_0}}{\mathbf{dx_0} \cdot \mathbf{dx_0}} = \frac{1}{2}\hat{\chi}^{\mathbf{0}} \cdot \left(\underline{\mathbf{F}}^T\underline{\mathbf{F}} - \mathbf{I}\right)\hat{\chi}^{\mathbf{0}} = \hat{\chi}^{\mathbf{0}} \cdot \underline{\mathbf{E}}\hat{\chi}^{\mathbf{0}}, \tag{5.10}$$

where $\hat{\chi}^{\mathbf{0}}$ is the unitary vector, $\underline{\mathbf{F}}^T$ is the transpose of the deformation gradient and

$$\underline{\mathbf{E}} = \frac{1}{2}\left(\underline{\mathbf{F}}^T\underline{\mathbf{F}} - \mathbf{I}\right) = \frac{1}{2}(\nabla \mathbf{u} + \nabla \mathbf{u}^T + \nabla \mathbf{u}^T\nabla \mathbf{u}) \tag{5.11}$$

is the Green-Lagrange strain tensor. An important characteristic of this tensor is that it is symmetric.

From the Green-Lagrange strain tensor it is possible to define the distortion γ between two unitary orthogonal fibers $\mathbf{z_1}$ and $\mathbf{z_2}$ as:

$$\sin\gamma(\mathbf{z_1}, \mathbf{z_2}) = \frac{2\mathbf{z_1} \cdot \underline{\mathbf{E}}\mathbf{z_2}}{\sqrt{(2\mathbf{z_1} \cdot \underline{\mathbf{E}}\mathbf{z_1} + 1)(2\mathbf{z_2} \cdot \underline{\mathbf{E}}\mathbf{z_2} + 1)}}. \tag{5.12}$$

The diagonal of $\underline{\mathbf{E}}$ regard to the normal strain and the other terms regards to the distortion.

5.2 The Stress Tensor and Motion Equations

After discussing strain and deformation, it is natural to introduce a stress analysis. A solid body subjected to forces implies internal forces acting to maintain the equilibrium. The internal forces are called stresses and they are related to an specific area. Figure 5.2 shows

a half of a solid body subjected to forces and the stress $\mathbf{p}$ associated to the cut surface, the area dA, the point x and the unitary normal vector $\hat{\mathbf{n}}$. It is known that the stress only depends on the associated normal vector related to the cut surface.The generic stress $\mathbf{p}$ can be divided into a normal component and a shear component as shown below:

$$\mathbf{p}(x, \hat{\mathbf{n}}) = \sigma(x, \hat{\mathbf{n}}) + \tau(x, \hat{\mathbf{n}}), \tag{5.13}$$

where σ is the normal stress and τ is the shear stress.

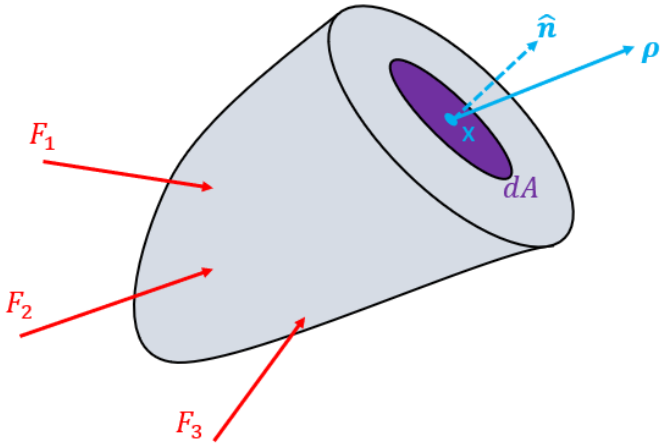


Figure 5.2: *Half of a solid body subjected to forces and the stress* $\mathbf{p}$.
Source: Author

Before introducing the stress tensor, it is relevant to present the principle of linear momentum (PLM).

Principle 1 (Linear Momentum)**.** *For any given volume V of a generic solid body and for any time instant t, it holds true that:*

$$\int_V \mathbf{b} dV + \int_{\mathcal{S}} \mathbf{p} d\mathcal{S} = \int_V \mu \mathbf{a_c} dV, \tag{5.14}$$

where $\mathbf{b}$ *is the body force applied to the solid,* μ *is the specific mass,* $\mathcal{S}$ *is the surface area and* $\mathbf{a_c}$ *is the acceleration field.*

From this principle, it can be concluded that:

$$\mathbf{p}(x, \hat{\mathbf{n}}) = -\mathbf{p}(x, -\hat{\mathbf{n}}). \tag{5.15}$$

Also, it possible to present the Cauchy stress tensor $\underline{\mathbf{T}} : \mathbb{R}^3 \longrightarrow \mathbb{R}^3$ that maps a normal vector to a stress vector, such that:

$$\underline{\mathbf{T}}\hat{\mathbf{n}} = \mathbf{p}(x, \hat{\mathbf{n}}). \tag{5.16}$$

The tensor can be described by:

$$\underline{\mathbf{T}} = \begin{bmatrix} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \\ T_{31} & T_{32} & T_{33} \end{bmatrix}, \tag{5.17}$$

where the terms of the diagonal refer to normal stress and the other terms refer to shear stress. It is important to mention that this matrix is symmetric. Considering the canonical basis $(\mathbf{e_1}, \mathbf{e_2}, \mathbf{e_3})$ and the notation T_{ij}, i indicates the direction and j the plane with normal $\mathbf{e_j}$. Now, with the stress tensor defined it is possible to deduce the motion differential equations:

$$\int_V \mathbf{b}dV + \int_{\mathcal{S}} \mathbf{p}d\mathcal{S} = \int_V \mu\mathbf{a_c}dV \Rightarrow \int_{\mathcal{S}} \underline{\mathbf{T}}\hat{\mathbf{n}}d\mathcal{S} = \int_V \mu\mathbf{a_c} - \mathbf{b}dV. \tag{5.18}$$

Applying the Divergence Theorem to the term on the left-hand side of the equation such that:

$$\int_{\mathcal{S}} \underline{\mathbf{T}}\hat{\mathbf{n}}d\mathcal{S} = \int_V div\underline{\mathbf{T}}dV, \tag{5.19}$$

then it holds true that:

$$\int_V div\underline{\mathbf{T}}dV = \int_V \mu\mathbf{a_c} - \mathbf{b}dV \Rightarrow \int_V (div\underline{\mathbf{T}} - \mu\mathbf{a_c} + \mathbf{b})dV = \mathbf{0}. \tag{5.20}$$

As equation (5.20) is true for any arbitrary V and considering static equilibrium:

$$div\underline{\mathbf{T}} + \mathbf{b} = \mathbf{0}. \tag{5.21}$$

Equation (5.21) can be written as a system of equations:

$$\begin{cases} \frac{\partial T_{11}}{\partial x_1} + \frac{\partial T_{12}}{\partial x_2} + \frac{\partial T_{13}}{\partial x_3} + b_1 = 0 \\ \frac{\partial T_{21}}{\partial x_1} + \frac{\partial T_{22}}{\partial x_2} + \frac{\partial T_{23}}{\partial x_3} + b_2 = 0 \\ \frac{\partial T_{31}}{\partial x_1} + \frac{\partial T_{32}}{\partial x_2} + \frac{\partial T_{33}}{\partial x_3} + b_3 = 0 \end{cases}. \tag{5.22}$$

5.3 Theory of Linear Elasticity

Everything presented until now considers the finite elasticity not restricted to small displacements and rotations. Consequently, the resultant equation system contains non-linear terms, which, in general, are difficult to handle. For a great range of problems a simpler formulation can be considered, by adopting the linear hypothesis. The geometric linearity hypothesis concerns to the assumption of infinitesimal displacements. In this sense, the reference configuration and the deformed configuration can be considered as equal and the non-linear term in the Green-Lagrange strain tensor, presented in equation (5.11) can

be considered zero:

$$\underline{\mathbf{E}} = \frac{1}{2}(\nabla \mathbf{u} + \nabla \mathbf{u}^T). \tag{5.23}$$

In linear elasticity, the stress components can be represented by the Cauchy stress tensor given in equation (5.17). As both $\underline{\mathbf{E}}$ and $\underline{\mathbf{T}}$ are symmetric, there are six independent components from each tensor. Thus, using the Voigt's notation, it is possible to represent them as:

$$\sigma = \begin{bmatrix} T_{11} \\ T_{22} \\ T_{33} \\ T_{12} \\ T_{13} \\ T_{23} \end{bmatrix}; \quad \varepsilon = \begin{bmatrix} E_{11} \\ E_{22} \\ E_{33} \\ 2E_{12} \\ 2E_{13} \\ 2E_{23} \end{bmatrix} = \begin{bmatrix} E_{11} \\ E_{22} \\ E_{33} \\ \gamma_{12} \\ \gamma_{13} \\ \gamma_{23} \end{bmatrix}. \tag{5.24}$$

The physical linearity hypothesis is related to the behavior of the adopted material. Therefore, it is possible to introduce a linear symmetric operator $\mathcal{D}$ which associates stresses and strains, implying in a linear constitutive equation:

$$\sigma = \mathcal{D}\varepsilon \Leftrightarrow \varepsilon = \mathcal{C}\sigma, \tag{5.25}$$

where $\mathcal{C} = \mathcal{D}^{-1}$. Considering a linear elastic isotropic material, it is possible to write:

$$\mathcal{D} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} (1-\nu) & \nu & \nu & 0 & 0 & 0 \\ \nu & (1-\nu) & \nu & 0 & 0 & 0 \\ \nu & \nu & (1-\nu) & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} \tag{5.26}$$

and, consequently,

$$\mathcal{C} = \frac{1}{E} \begin{bmatrix} 1 & -\nu & -\nu & 0 & 0 & 0 \\ -\nu & 1 & -\nu & 0 & 0 & 0 \\ -\nu & -\nu & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2(1+\nu) & 0 & 0 \\ 0 & 0 & 0 & 0 & 2(1+\nu) & 0 \\ 0 & 0 & 0 & 0 & 0 & 2(1+\nu) \end{bmatrix}. \tag{5.27}$$

The principle of linear momentum still holds true and the balance equation is given by 5.21. In this sense, considering the set of prescribed displacements $\mathbb{S}_u$ and the set of prescribed forces $\mathbb{S}_t$, the differential formulation of the linear elasticity problem is complete.

A particular case of the Theory of Elasticity is the plane strain formulation in which the displacement field in one direction is equal to zero. In this work, the plane strain is considered

in the context of linear elasticity. With this simplification, three dimensional problems can be solved with a two dimensional formulation. Considering that $u_3 \equiv 0 \Rightarrow \varepsilon_3 = E_{33} = 0$. Thus, the stress and strain are reduced to:

$$\sigma = \begin{bmatrix} T_{11} \\ T_{22} \\ T_{12} \end{bmatrix}; \quad \varepsilon = \begin{bmatrix} E_{11} \\ E_{22} \\ 2E_{12} \end{bmatrix} = \begin{bmatrix} E_{11} \\ E_{22} \\ \gamma_{12} \end{bmatrix}. \tag{5.28}$$

The constitutive operator becomes:

$$\mathcal{D} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} (1-\nu) & \nu & 0 \\ \nu & (1-\nu) & 0 \\ 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} \tag{5.29}$$

and,

$$\mathcal{C} = \frac{1+\nu}{E} \begin{bmatrix} (1-\nu) & -\nu & 0 \\ -\nu & (1-\nu) & 0 \\ 0 & 0 & 2 \end{bmatrix}. \tag{5.30}$$

The balance equations, considering static equilibrium are given by:

$$\begin{cases} \frac{\partial T_{11}}{\partial x_1} + \frac{\partial T_{12}}{\partial x_2} + b_1 = 0 \\ \frac{\partial T_{21}}{\partial x_1} + \frac{\partial T_{22}}{\partial x_2} + b_2 = 0 \end{cases}. \tag{5.31}$$

Again, considering the prescribed displacements and forces, the formulation for the plane strain is complete.

Chapter 6

Saint-Venant Torsion Problem

This chapter was inspired in the work presented in Moherdaui and Neto (2019) and the main goal is to show an application of Virtual Element Method in a engineering problem regarding to the Poisson Equation. Also, a comparison between Virtual Element Method, Finite Element Method and Finite Difference Method is made. FDM is one of the classical approaches for solving the Poisson Equation. Since the FDM method, which is based on Taylor series approximations, requires a space of continuous function, an iterative approach is used. As support text to build the formulation, the works in Timoshenko and Goodier (1951) and Bucalem and Bathe (2011) were used.

The first section is dedicated to the formulation of the Saint-Venant torsion problem using the Prandtl Stress Function. This function is the responsible for the construction of Poisson Equation in the torsion problem. Then, using the implementation pipeline presented in Chapter 4, the Virtual Element Method is used in order to find the torsion constant. The results will be compared with a FEM and a FDM approach.

6.1 Formulation of Saint-Venant Torsion Problem

According to Timoshenko and Goodier (1951), Coloumb stated that for bars with circular cross-section no warping occurs during torsion. Here, warping can be understood as the movement in the direction of the bar's axis during torsion phenomenon. The authors say that Navier assumed this hypothesis for prismatic bars with non-circular cross-section and erroneously conclude that the torsion angle was inversely proportional to the cross-section polar moment of inertia and that the maximum shear stress tend to occur at the point farther of the centroid. This conclusion causes a contradiction between boundary and equilibrium conditions. The correct formulation was given by Saint-Venant by considering the warping. Thus, both boundary and equilibrium conditions were satisfied for an uniform torsion problem. Under these circumstances, the following hypothesis are made:

Hypothesis 4. *For a prismatic bar it is assumed that:*

- *the cross-sections experience rigid body rotation,*

- *all cross-sections warp in the same way,*
- *the formulation is restricted for small rotations (*$\sin\theta \approx \theta$ *and* $\cos\theta \approx 1$*).*

The linear elastic model is used in this formulation. Figure 6 shows the top view of the cross-section where $\theta(x)$ is the rotation angle that occurs when a constant torque T is applied and O is the rotation center. As the torsion is uniform, then the twist rate $\theta'(x) = \frac{\partial\theta}{\partial x}(x)$ is assumed constant and will be denoted just by θ'. Thus, by Saint-Venant's formulation it can be written as:

$$T = GI_t\theta' \Leftrightarrow \theta' = \frac{T}{GI_t}, \tag{6.1}$$

where G is the shear modulus and I_t is the torsion constant. The shear modulus can be computed as $G = \frac{E}{2(1+\nu)}$, where E is the elastic modulus and ν is the Poisson coefficient. It also can be assumed that:

$$\theta(0) = 0 \quad and \quad \theta(x) = x\theta'. \tag{6.2}$$

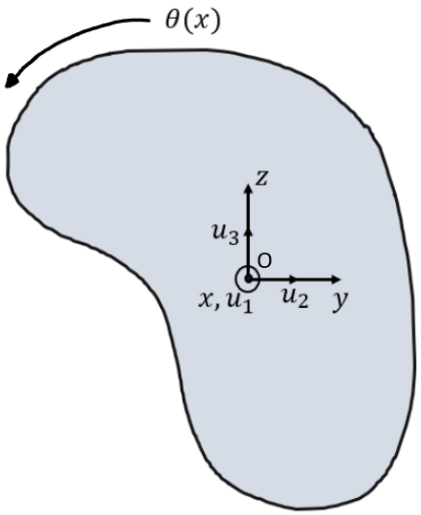


Figure 6.1: *Top view of the cross-section of a prismatic bar.*
Source: Author

It is important to mention that the small rotation hypothesis implies that $\sin\theta \approx 0$ and $\cos\theta \approx 1$. Thus, as in Timoshenko and Goodier (1951), the displacement field $u = (u_1, u_2, u_3)$ is given by:

$$u_1 = \theta'\Psi(y,z), \tag{6.3}$$
$$u_2 = -zx\theta', \tag{6.4}$$
$$u_3 = yx\theta' \tag{6.5}$$

where $\Psi(y,z)$ is the warping function. Consequently, the strain components are:

$$\varepsilon_x = \frac{\partial u_1}{\partial x} = 0, \quad \varepsilon_y = \frac{\partial u_2}{\partial y} = 0, \quad \varepsilon_z = \frac{\partial u_3}{\partial z} = 0, \tag{6.6}$$

$$\gamma_{xz} = \frac{\partial u_1}{\partial z} + \frac{\partial u_3}{\partial x} = \theta' \left(\frac{\partial \Psi}{\partial z} + y \right), \tag{6.7}$$

$$\gamma_{yz} = \frac{\partial u_2}{\partial z} + \frac{\partial u_3}{\partial x} = -x\theta' + x\theta = 0, \tag{6.8}$$

$$\gamma_{xy} = \frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x} = \theta' \left(\frac{\partial \Psi}{\partial y} - z \right). \tag{6.9}$$

By the Generalized Hooke's Law:

$$\sigma_x = \sigma_y = \sigma_z = \tau_{yz} = 0, \tag{6.10}$$

$$\tau_{xz} = G\gamma_{xz} = G\theta' \left(\frac{\partial \Psi}{\partial z} + y \right), \tag{6.11}$$

$$\tau_{xy} = G\gamma_{xy} = G\theta' \left(\frac{\partial \Psi}{\partial y} - z \right). \tag{6.12}$$

Considering static equilibrium and no body force, the equilibrium equation can be written as

$$div\underline{\mathbf{T}} = \mathbf{0}, \tag{6.13}$$

where $\underline{\mathbf{T}}$ is the Cauchy Stress Tensor. Substituting equation (6.10) in equation (6.13):

$$\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{xz}}{\partial z} = 0 \Leftrightarrow \frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{xz}}{\partial z} = 0. \tag{6.14}$$

Now, substituting equations (6.10) and (6.11) in (6.14):

$$\frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{xz}}{\tau_{xz}} = 0 \Rightarrow G\theta' \left(\frac{\partial^2 \Psi}{\partial y^2} + \frac{\partial^2 \Psi}{\partial z^2} \right) = 0. \tag{6.15}$$

As $G\theta' \neq 0$, equation (6.15) can be written as:

$$\frac{\partial^2 \Psi}{\partial y^2} + \frac{\partial^2 \Psi}{\partial z^2} = 0 \Leftrightarrow \Delta\Psi = 0, \forall (y,z) \in \Omega, \tag{6.16}$$

where Ω is the geometric domain of the cross-section. Equation (6.16) is knows as Laplace Equation and can be seen as an particular case of Poisson Equation when $f \equiv 0$. The boundary conditions also need to be defined. Figure 6.2 shows the domain Ω, the boundary $\partial\Omega$, the normal vector $\mathbf{n}$ to the boundary and the tangent vector $\mathbf{t}$ to the boundary. Considering the parameterization curve of the boundary given by $r(s) = (0, y(s), z(s))$ and the canonical basis $(\mathbf{e_1}, \mathbf{e_2}, \mathbf{e_3})$ for the euclidean space, the tangent vector can be written as:

$$\mathbf{t} = \left(0, \frac{\partial y}{\partial s}(s), \frac{\partial z}{\partial s}(s) \right) \tag{6.17}$$

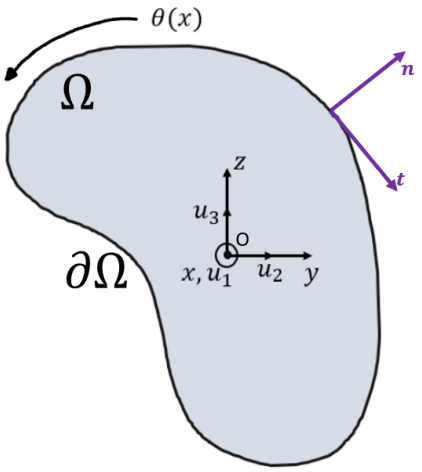


Figure 6.2: *Cross-section domain* Ω*, the boundary* $\partial\Omega$*, the normal vector* $\mathbf{n}$ *to the boundary and the tangent vector* $\mathbf{t}$ *to the boundary.*

Source: Author

and the normal vector, using the cross product,

$$\mathbf{n} = \mathbf{t} \times \mathbf{e_1} = \left(0, -\frac{\partial z}{\partial s}, \frac{\partial y}{\partial s}\right). \tag{6.18}$$

Since the stress on $\partial\Omega$ is zero, it is possible to write:

$$\underline{\mathbf{T}}\mathbf{n} = \mathbf{0} \Leftrightarrow G\theta'\left(\frac{\partial \Psi}{\partial y} - z\right)\left(-\frac{\partial z}{\partial s}\right) + G\theta'\left(\frac{\partial \Psi}{\partial z} + y\right)\left(\frac{\partial y}{\partial s}\right) = 0. \tag{6.19}$$

Equation (6.19) represents the boundary condition for the torsion problem.

The next step is define the Prandtl Stress Function in order to setup the Poisson Equation as showed in Chapter 3.

Definition 16. *Let* $\Phi \in C^2(\Omega)$ *such that*

$$\tau_{xy} = \frac{\partial \Phi}{\partial z}(y, z), \quad \textit{and} \quad \tau_{xz} = -\frac{\partial \Phi}{\partial y}(y, z). \tag{6.20}$$

This function is called Prandtl Stress Function.

In definition 16, since $\Phi \in C^2(\Omega)$ by applying the Schwarz Theorem it is possible to conclude that:

$$\frac{\partial^2 \Phi}{\partial y \partial z} = \frac{\partial^2 \Phi}{\partial z \partial y}. \tag{6.21}$$

Substituting equations (6.11) and (6.12) in definition 16:

$$\tau_{xz} = G\theta'\left(\frac{\partial \Psi}{\partial z} + y\right) = -\frac{\partial \Phi}{\partial y}, \tag{6.22}$$

$$\tau_{xy} = G\theta'\left(\frac{\partial \Psi}{\partial y} - z\right) = \frac{\partial \Phi}{\partial z}. \tag{6.23}$$

The derivatives of equations (6.22) and (6.23) with respect to y and z respectively are given by:

$$G\theta'\left(\frac{\partial^2\Psi}{\partial z\partial y}+1\right) = -\frac{\partial^2\Phi}{\partial y^2}, \tag{6.24}$$

$$G\theta'\left(\frac{\partial^2\Psi}{\partial z\partial y}-1\right) = \frac{\partial^2\Phi}{\partial z^2}. \tag{6.25}$$

Subtracting equation (6.25) from equation (6.24):

$$\frac{\partial^2\Phi}{\partial y^2}+\frac{\partial^2\Phi}{\partial z^2} = -2G\theta'. \tag{6.26}$$

Considering the total derivative of Φ denoted by $D\Phi$:

$$D\Phi = \frac{\partial\Phi}{\partial y}\frac{\partial y}{\partial s}ds+\frac{\partial\Phi}{\partial z}\frac{\partial z}{\partial s}ds = \left(\frac{\partial\Phi}{\partial y}\frac{\partial y}{\partial s}+\frac{\partial\Phi}{\partial z}\frac{\partial z}{\partial s}\right)ds = \frac{d\Phi}{ds}ds, \tag{6.27}$$

for all $y, z \in \partial\Omega$. Rewriting equation (6.19) as:

$$-\frac{\partial\Phi}{\partial z}\frac{\partial z}{\partial s}-\frac{\partial\Phi}{\partial y}\frac{\partial y}{\partial s} = 0. \tag{6.28}$$

Then,

$$\frac{d\Phi}{ds} = 0, \quad \forall y, z \in \partial\Omega. \tag{6.29}$$

Arbitrating that $\Phi \equiv 0$ on $\partial\Omega$, thus the Poisson Equation regarding to Saint-Venant formulation for torsion is:

$$\begin{cases} \Delta\Phi = \frac{\partial^2\Phi}{\partial y^2}+\frac{\partial^2\Phi}{\partial z^2} = -2G\theta' & \text{in} \quad \Omega \\ \Phi = 0 \quad \text{on} \quad \partial\Omega \end{cases}. \tag{6.30}$$

Finally, the torque can be defined as the total moment in each cross-section:

$$\begin{aligned} T = \int_\Omega \left(y\frac{\partial\Phi}{\partial y}-z\frac{\partial\Phi}{\partial z}\right)d\Omega = -\int_\Omega \left(\frac{\partial(y\Phi)}{\partial y}+\frac{\partial(z\Phi)}{\partial z}-2\Phi\right)d\Omega = \\ = \int_\Omega \left(\frac{\partial(y\Phi)}{\partial y}+\frac{\partial(z\Phi)}{\partial z}\right)d\Omega+2\int_\Omega \Phi d\Omega. \end{aligned} \tag{6.31}$$

Applying the Divergence Theorem in the first term of equation (6.31):

$$T = \int_{\partial\Omega}\Phi\left(-y\frac{\partial z}{\partial s}+z\frac{\partial y}{\partial s}\right)d\partial\Omega+2\int_\Omega \Phi d\Omega. \tag{6.32}$$

Since $\Phi \equiv 0$ on $\partial\Omega$,

$$T = 2 \int_{\Omega} \Phi d\Omega. \tag{6.33}$$

Substituting equation (6.33) in equation (6.1), the torsion constant is:

$$I_t = \frac{2 \int_{\Omega} \Phi d\Omega}{G\theta'} = \frac{2}{G\theta'} \int_{\Omega} \Phi d\Omega. \tag{6.34}$$

6.2 VEM applied to torsion

This section is dedicated to present the results obtained by applying the Virtual Element Method to the Saint-Venant's torsion formulation showed earlier. Since the goal is to solve Poisson Equation (6.30) and obtain the torsion constant by applying equation (6.34), the implementation framework of VEM is the same as presented in Chapter 4. As mentioned before, a comparison with FDM and FEM are also made and a discussion regarding the performance of the methods are discussed. It is important to recall that for VEM and FEM only the linear case is considered. The discussions about FEM and FDM implementation were suppressed, since it is out of the scope of the present work. For further information about the other methods, a detailed explanation about the Finite Difference Method and Finite Element Method can be found in LeVeque (2007) and Alberty *et al.* (1999), respectively.

As in Chapter 4, the domain is a unitary square $\Omega = [0, 1] \times [0, 1]$ representing the cross section of the prismatic bar. In Timoshenko and Goodier (1951), an analytical value for I_t concerning to rectangular cross sections is presented. Considering the width w_i of the cross section , the height h_e of the cross section and that $G\theta' = 1$, the analytical value is given by a trigonometric series:

$$I_t = \frac{1}{3} w_i^3 h_e \left(1 - \frac{192 w_i}{h_e \pi^5} \sum_{i=0}^{\infty} \frac{1}{(2i+1)^5} \tanh \frac{(2i+1) h_e \pi}{2 w_i} \right). \tag{6.35}$$

In this case, $h_e = w_i = 1$, thus:

$$I_t = \frac{1}{3} (1 - 0.63) = 0.1406. \tag{6.36}$$

The numerical value of the torsion constant will be denoted by I_t^h and the error will be evaluated by:

$$Error(\%) = \frac{|I_t^h - I_t|}{I_t} 100. \tag{6.37}$$

The meshes used for VEM are the same shown in Figure 4.1. For FEM, only the uniform square mesh was used. Table 6.1 shows the numerical values obtained using VEM, FEM and FDM alongside the associated error for different number of elements. It is important to mention that for FDM elements are not properly used, instead it is used a discretization

composed by points equivalent to the nodes of VEM and FEM meshes. Figures 6.3 is graphi-

Numerical solution for Saint-Venant torsion problem								
Elements	**VEM**		**VEM Voronoi**		**FEM**		**FDM**	
	$\mathbf{I_t^h}$	**Error**	$\mathbf{I_t^h}$	**Error**	$\mathbf{I_t^h}$	**Error**	$\mathbf{I_t^h}$	**Error**
16	0.1354	3.6866	0.1450	3.0993	0.1279	9.0314	0.1152	18.0410
36	0.1382	1.6874	0.1444	2.7277	0.1348	4.0901	0.1286	8.5202
64	0.1392	0.9628	0.1427	1.5288	0.1373	2.3226	0.1337	4.9131
100	0.1397	0.6239	0.1421	1.0755	0.1385	1.4971	0.1361	3.1873
144	0.1400	0.4389	0.1418	0.8776	0.1391	1.0465	0.1375	2.2334
256	0.1402	0.2544	0.1413	0.5178	0.1398	0.5970	0.1388	1.2724
400	0.1404	0.1688	0.1411	0.3342	0.1401	0.3883	0.1394	0.8231
676	0.1405	0.1066	0.1409	0.1939	0.1403	0.2366	0.1399	0.4950
1024	0.1405	0.0759	0.1408	0.1249	0.1404	0.1618	0.1401	0.3328
2304	0.1405	0.0428	0.1407	0.0390	0.1405	0.0810	0.1404	0.1572
2704	0.1405	0.0389	0.1406	0.0269	0.1405	0.0715	0.1404	0.1364
4096	0.1406	0.0313	0.1406	0.0037	0.1405	0.0527	0.1405	0.0956

Table 6.1: *Numerical results obtained regarding to the torsion constant.*

cal representations of Table 6.1, showing the relation between number of elements and the torsion constant. Figures 6.4 and 6.5 show the error in different graphical scales. From the table and the figures it is possible to observe the convergence for all methods. The Virtual Element Method using uniform and Voronoi mesh has a very similar performance when compared to Finite Element Method. It is possible to note that VEM is slightly faster than FEM for a small number of elements. On the other hand, comparing Virtual Element Method outperforms Finite-Difference Method as one can be seen.

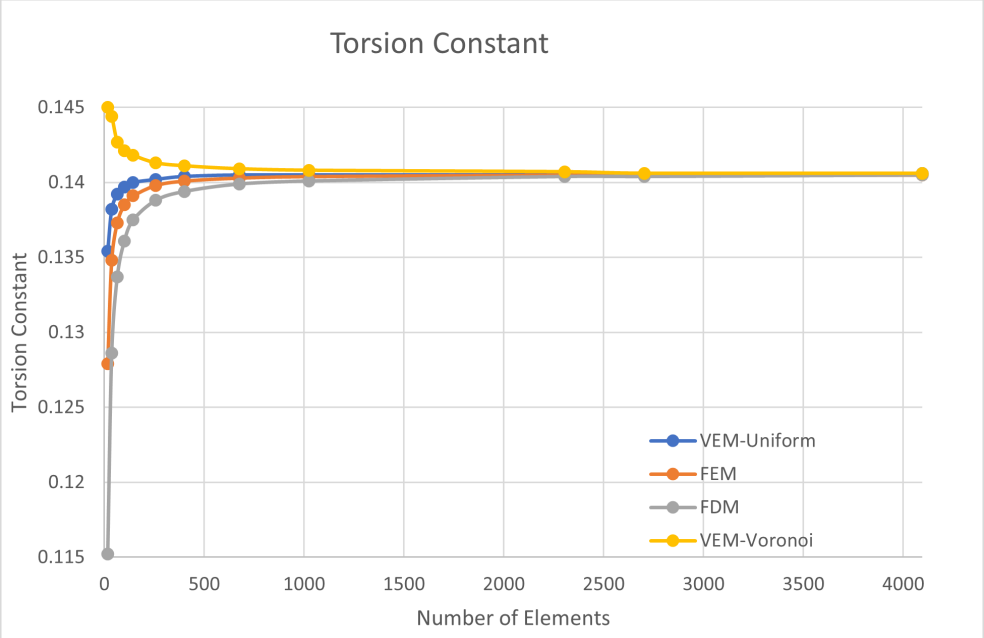


Figure 6.3: *Numerical value of torsion constant and number of elements.*
Source:Author

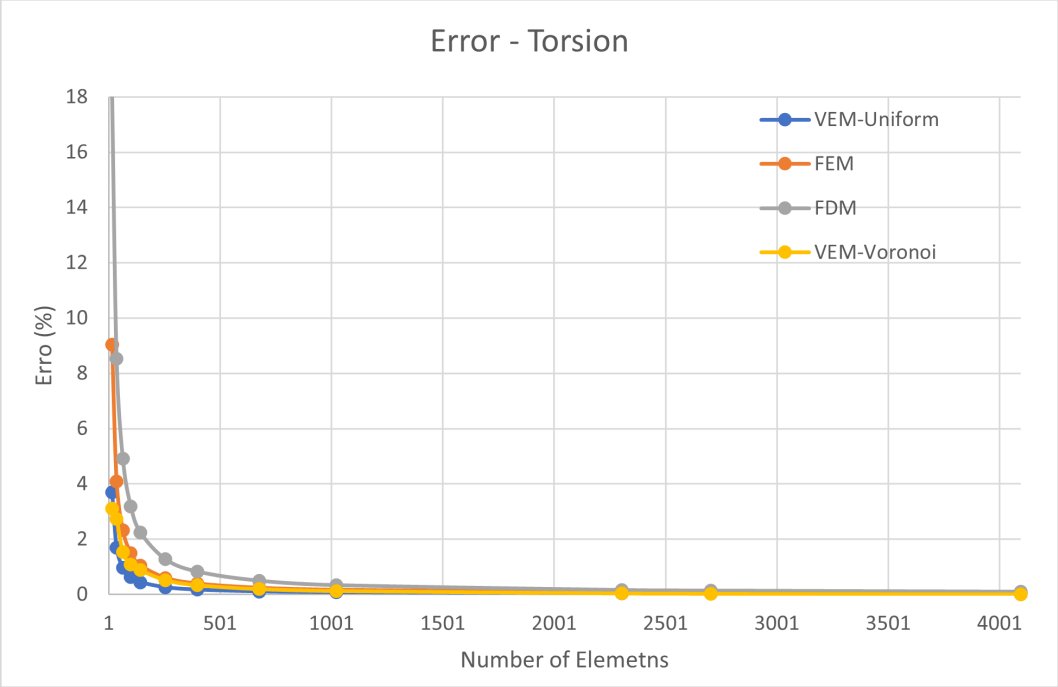


Figure 6.4: *Error related to torsion constant and number of elements.*
Source:Author

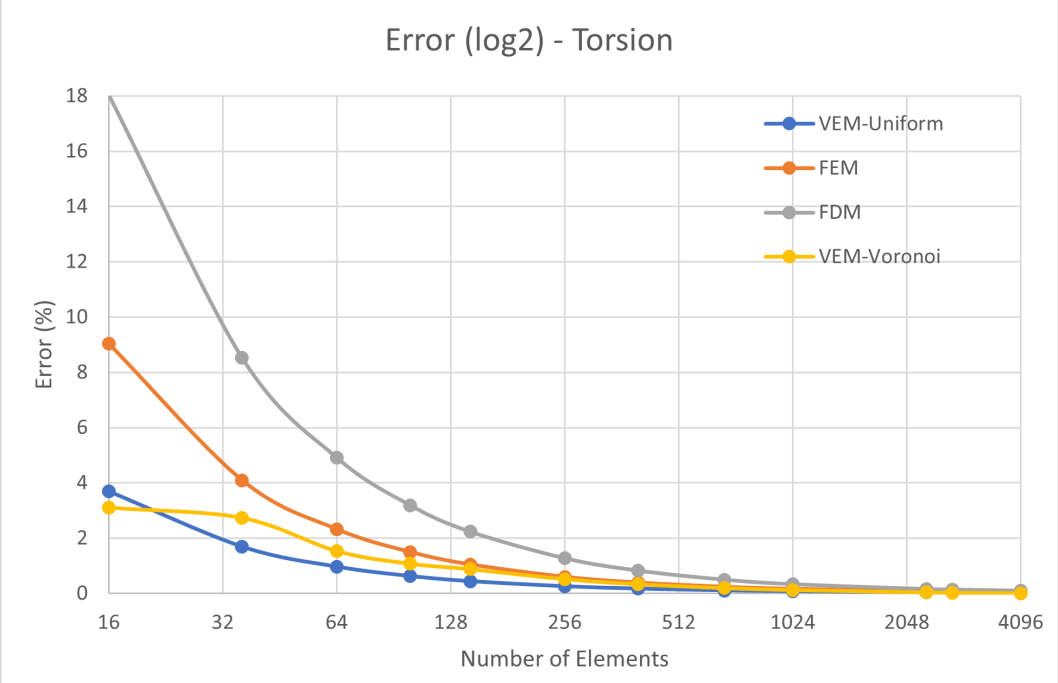
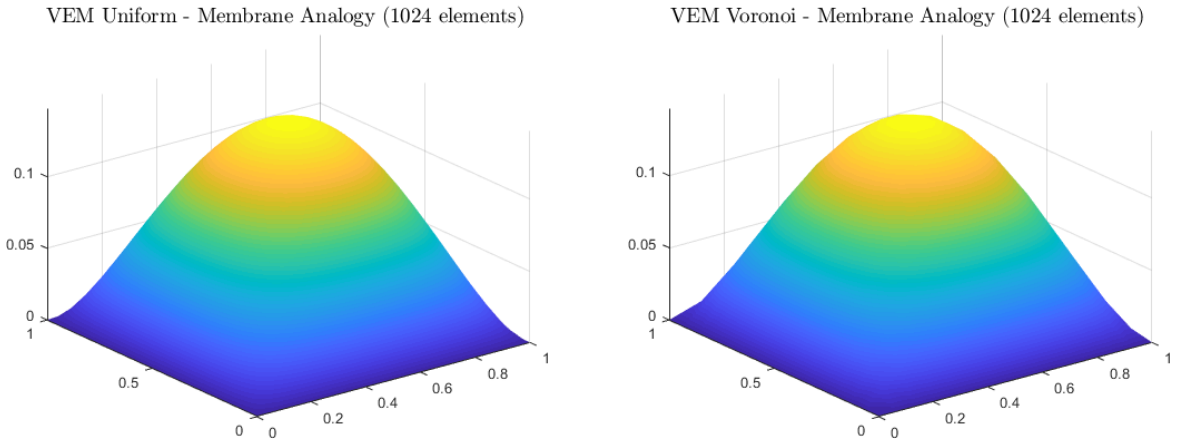


Figure 6.5: *Error related to torsion constant and number of elements (logarithm scale).*
Source:Author

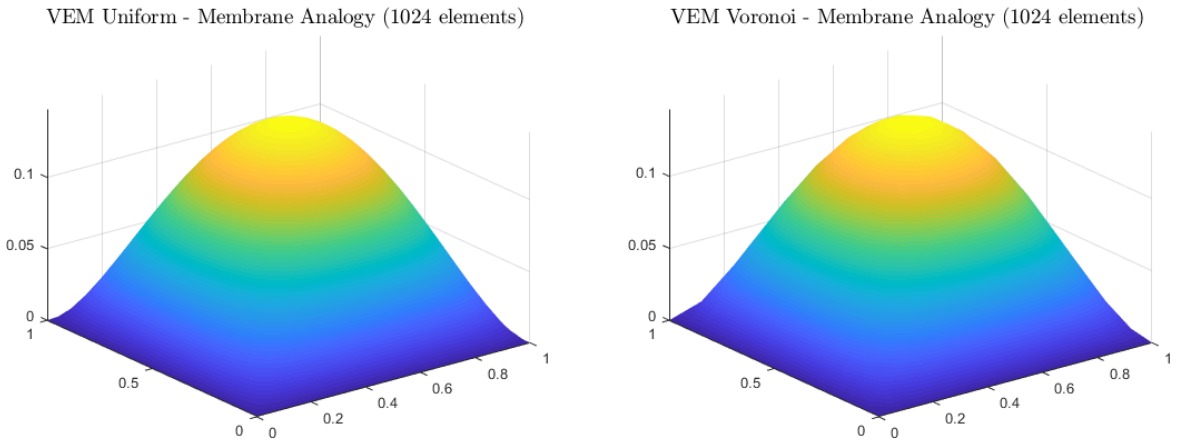
To evaluate the Prandtl Stress Function Φ, the membrane analogy can be used. Accordingly to Timoshenko and Goodier (1951), this analogy was introduced by Pradtl and it is used when Φ cannot be determined explicitly. The membrane analogy can be seen as a membrane above a empty structure. In the present case the structure is a square. A pressure is applied to the membrane and it inflates. In this sense, by the analogy, the stress is tangential

to the contour lines that emerges in the membrane as result of the pressure application and the boundary, the module of the tangential stress in each cross section is proportional to the membrane's slope and the volume of the deformed membrane is proportional to the torque. Thus, it is possible to observer that for the square cross section the Φ should be zero at the boundaries and its maximum value must be in the center of the cross section. Figure 6.6 shows the membrane analogy regarding to Virtual Element Method using uniform and Voronoi meshes with 1024 elements. And, Figure 6.7 shows the membrane analogy for FEM and FDM with 1024 elements. It is possible to see from these figures the correspondence to the membrane analogy assumptions.



(a) *Uniform mesh* **(b)** *Voronoi mesh*

Figure 6.6: *Membrane analogy using the Virtual Element Method*
Source: Author



(a) *FDM* **(b)** *FEM*

Figure 6.7: *Membrane analogy using Finite-Difference Method and Finite Element Method*
Source: Author

As expected, for simple geometries Virtual Element Method and Finite Element Method has very similar performance. The VEM with Voronoi mesh showed a better performance

than with the uniform mesh. Thus, it can be seen as an advantage of using VEM since the use of Voronoi mesh with FEM requires the usage of isoparametric elements (as discussed in Chapter 4) and with VEM the elements are computed directly from the degrees of freedom. Also, it is important to mention that VEM performance for Poisson Equation implementation is in accordance with the literature presented in Chapter 2.

Chapter 7

Virtual Element Method Applied to Linear Elasticity

In this chapter, Virtual Element Method is applied to solve the differential equations in linear elasticity context considering the plane state hypothesis. As can be seen, the formulation of VEM in this case is an extension of what was shown in the Poisson Equation case. This chapter is mostly based on the model presented in Gain *et al.* (2014) and in Ortiz-Bernardin *et al.* (2019). This model is restricted to the linear case ($k = 1$) but as it is stated in Ortiz-Bernardin *et al.* (2019), due to the matrix framework the formulation is more familiar to engineers.

The first part of this chapter is dedicated to present the weak formulation. Then, the Virtual Element Formulation is shown keeping the notation presented in Chapter 3. Finally, the implementation framework is displayed similarly to Chapter 4.

7.1 The weak formulation

Let $\Omega \subset \mathbb{R}^2$ be a generic elastic solid domain. This solid is subjected to a body force $\mathbf{b}$ in Ω and in $\partial\Omega$ it is subjected to an external traction $\mathbf{g}$ (Neumann's boundary condition) and prescribed displacements $\hat{\mathbf{u}}$ (Dirichlet's boundary condition) as shown in Figure 7.1. S_f denotes the set of points where the external traction is applied and S_d denotes the set of the points with prescribed displacement.

As described in Chapter 5, the differential formulation for the problem mentioned above is given by:

$$\begin{cases} div\underline{\mathbf{T}} + \mathbf{b} = \mathbf{0}, \\ \underline{\mathbf{T}}\hat{\mathbf{n}} = \mathbf{g}, \quad \mathbf{x} \in S_f, \\ \mathbf{u} = \hat{\mathbf{u}}, \quad \mathbf{x} \in S_d, \end{cases} \tag{7.1}$$

where $\mathbf{u} \in C^2(\Omega) \cap C(\overline{\Omega})$, $\mathbf{b} \in L^2(\Omega)$ and $\mathbf{g} \in L^2(\partial\Omega)$. As in the Poisson Equation, to apply the Virtual Element Method, the weak formulation is necessary.

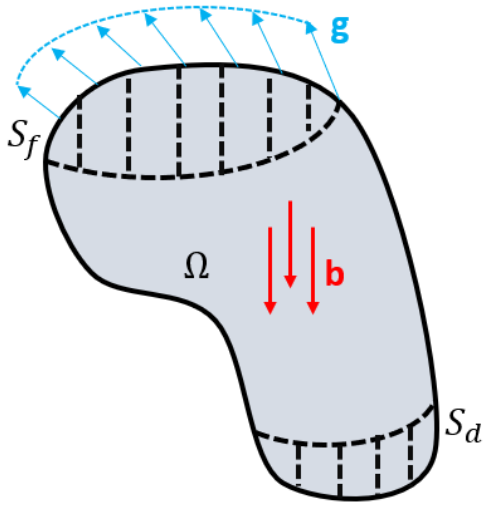


Figure 7.1: *Generic elastic solid domain with Dirichlet and Neumann boundary conditions.*
Source: Author

Let $\mathbf{v} \in H_0^1(\Omega)$ be a test function. Multiplying equation (7.1) by the test function, one shall obtain:

$$\int_\Omega div\underline{\mathbf{T}} \cdot \mathbf{v} d\Omega + \int_\Omega \mathbf{b} \cdot \mathbf{v} d\Omega = 0. \tag{7.2}$$

Integrating by parts the first term of the equation above:

$$\int_\Omega div\underline{\mathbf{T}} \cdot \mathbf{v} d\Omega = -\int_\Omega \underline{\mathbf{T}} : \nabla \mathbf{v} d\Omega + \int_{\partial\Omega} \underline{\mathbf{T}}\hat{\mathbf{n}} \cdot \mathbf{v} dS. \tag{7.3}$$

Using the Neumann boundary condition:

$$\int_\Omega div\underline{\mathbf{T}} \cdot \mathbf{v} d\Omega = -\int_\Omega \underline{\mathbf{T}} : \nabla \mathbf{v} d\Omega + \int_{\partial\Omega} \mathbf{g} \cdot \mathbf{v} dS. \tag{7.4}$$

Substituting equation (7.4) in 7.2, the weak formulation is given by:

$$\int_\Omega \underline{\mathbf{T}} : \nabla \mathbf{v} d\Omega = \int_{\partial\Omega} \mathbf{g} \cdot \mathbf{v} dS + \int_\Omega \mathbf{b} \cdot \mathbf{v} d\Omega, \tag{7.5}$$

where $\mathbf{u}, \mathbf{v} \in H_0^1(\Omega)$. The operation $\underline{\mathbf{T}} : \nabla \mathbf{v} = tr(\underline{\mathbf{T}}^T \nabla \mathbf{v})$ refers to the scalar product between two tensors, where $tr(\cdot)$ is the trace.

The approach shown above is also known as the Principle of Virtual Work and $\mathbf{v}$ is also called virtual displacement field. As $\nabla \mathbf{v}$ is a second order tensor it can be decomposed in a symmetric and a skew-symmetric component. Thus, as only the symmetric component has influence and considering the Voigt notation it is possible to write:

$$\int_\Omega \sigma(\mathbf{u}) \cdot \varepsilon(\mathbf{v}) d\Omega = \int_{\partial\Omega} \mathbf{g} \cdot \mathbf{v} dS + \int_\Omega \mathbf{b} \cdot \mathbf{v} d\Omega. \tag{7.6}$$

It is worth mentioning that with the Voigt notation the scalar product between two vector is considered. Writing the problem in terms of the bilinear form, one shall have:

$$a(\mathbf{u}, \mathbf{v}) = f_s(\mathbf{v}) + f_b(\mathbf{v}), \tag{7.7}$$

where $a(\mathbf{u}, \mathbf{v}) = \int_\Omega \sigma(\mathbf{u}) \cdot \varepsilon(\mathbf{v}) d\Omega$, $f_s(\mathbf{u}) = \int_{\partial\Omega} \mathbf{g} \cdot \mathbf{v} dS$ and $f_b(\mathbf{v}) = \int_\Omega \mathbf{b} \cdot \mathbf{v} d\Omega$. With the weak form constructed, the Virtual Element Method can be formulated.

7.2 The Virtual Element Method

Considering a decomposition τ_h of Ω into simple polygons K. Similarly to what what done before, the idea is to construct each term of the discrete problem given by:

$$a_h(\mathbf{u}, \mathbf{v}) = f_{b,h}(\mathbf{v}) + f_{s,h}(\mathbf{v}), \tag{7.8}$$

for all $\mathbf{u}, \mathbf{v} \in V_h$ and where $a_h(\mathbf{u}, \mathbf{v}) = \sum_{K \in \tau_h} a_{h,K}(\mathbf{u}, \mathbf{v})$, $f_{b,h}(\mathbf{v}) = \sum_{K \in \tau_h} f_{b,K,h}(\mathbf{v})$ and $f_{s,h}(\mathbf{v}) = \sum_{K \in \tau_h} f_{s,K,h}(\mathbf{v})$. It is important to mention that hypothesis 1, 2, 3 are still assumed and Theorem 2 is still valid. Thus, the uniqueness and convergence of the solution for the discrete problem are guaranteed.

As stated in Gain *et al.* (2014) this formulation is restricted to the linear case (k=1). In this way, the set of degrees of freedom are the values of $\mathbf{v}$ on the vertices of each simple polygon K (see Chapter 3). Setting up $\mathcal{E}_K = \emptyset$ and $\mathcal{P}_K = \emptyset$, then $\mathcal{V}_K \cup \mathcal{E}_K \cup \mathcal{P}_K$ is unisolvent as shown in Theorem 3.

The authors in Ortiz-Bernardin *et al.* (2019) state that the convergence of the discrete solution is associated with the characteristic of the approximated displacement field that can be decomposed into a rigid body component and constant strain component. In this sense, the continuous space given in Definition 10, now refers to the space of linear displacements, characterized by a linear polynomial as presented in definition below.

Definition 17. *For each $K \in \tau_h$, the linear space is given by:*

$$\mathbb{E}_K = \{\mathbf{a} + \underline{\mathbf{M}}(\mathbf{x} - \mathbf{x_c}) : \mathbf{a} \in \mathbb{R}^2, \quad \underline{\mathbf{M}} \in \mathbb{R}^{2\times 2}\}. \tag{7.9}$$

With the linear space, it is possible to write the definition below.

Definition 18. *The virtual element space is defined as:*

$$V_h = \{\mathbf{v} \in C^0(\overline{\Omega}) : \mathbf{v}|_K \in \mathbb{D}_K, \forall K \in \tau_h\}, \tag{7.10}$$

where $\mathbb{D}_K$ is the space of deformations associated to each polygon K and $\mathbb{E}_K \subseteq \mathbb{D}_K$.

One can observe that $\mathbb{D}_K \subseteq H_0^1(K)$, for all $K \in \tau_h$, then $V_h \subseteq H_0^1(\Omega)$. From the definition of virtual element space, it can be verified that $\mathbb{D}_K$ has a similar role to the local virtual element space in the formulation of VEM regarding to the Poisson Equation. Recalling that a second order tensor can be decomposed into a symmetric and a skew-symmetric tensor, $\underline{\mathbf{M}}$ is written as:

$$\underline{\mathbf{M}} = \underline{\mathbf{M}}^* + \underline{\mathbf{M}}_*, \tag{7.11}$$

where $\underline{\mathbf{M}}^* = sym(\underline{\mathbf{M}})$ and $\underline{\mathbf{M}}_* = skw(\underline{\mathbf{M}})$. The linear space $\mathbb{E}_K$ can be kinetically decomposed into a space of rigid body motion $\mathcal{R}_K$ and a space of constant strain modes $\mathcal{CS}_K$, such that:

$$\mathcal{R}_K = \{\mathbf{a} + \underline{\mathbf{M}}_*(\mathbf{x} - \mathbf{x_c}) : \mathbf{a} \in \mathbb{R}^2, \quad \underline{\mathbf{M}}_* \in \mathbb{R}^{2\times 2}\}, \tag{7.12}$$

$$\mathcal{CS}_K = \{\underline{\mathbf{M}}^*(\mathbf{x} - \mathbf{x_c}) : \underline{\mathbf{M}}^* \in \mathbb{R}^{2\times 2}\} \tag{7.13}$$

Then, it is possible to to write $\mathbb{E}_K = \mathcal{R}_K + \mathcal{CS}_K$, where $\mathbb{E}_K \subseteq \mathbb{D}_K$

The projection operator Π^∇ is divided in three parts:

- $\Pi^\nabla_{\mathcal{R}_K} : \mathbb{D}_K \longrightarrow \mathcal{R}_K$ that is responsible to extract the rigid body motions such that

$$\Pi^\nabla_{\mathcal{R}_K}\mathbf{r} = \mathbf{r}, \quad \forall \mathbf{r} \in \mathcal{R}_K; \tag{7.14}$$

- $\Pi^\nabla_{\mathcal{CS}_K} : \mathbb{D}_K \longrightarrow \mathcal{CS}_K$ that is responsible to extract the constant strain modes, such that

$$\Pi^\nabla_{\mathcal{CS}_K}\mathbf{c} = \mathbf{c}, \quad \forall \mathbf{c} \in \mathcal{CS}_K; \tag{7.15}$$

- $\Pi^\nabla_{\mathbb{E}_K} : \mathbb{D}_K \longrightarrow \mathbb{E}_K$ that is responsible to extract the polynomial terms, such that

$$\Pi^\nabla_{\mathbb{E}_K} = \Pi^\nabla_{\mathcal{R}_K} + \Pi^\nabla_{\mathcal{CS}_K} \quad and \quad \Pi^\nabla_{\mathbb{E}_K}\mathbf{q} = \mathbf{q}, \quad \forall \mathbf{q} \in \mathbb{E}_K. \tag{7.16}$$

This approach is much similar to what was done in the VEM formulation for the Poisson Equation, once, functions that are not known in the first moment are being projected in subspaces of polynomial spaces. The definition below gives the explicit formula for the $\Pi^\nabla_{\mathcal{R}_K}$ operator.

Definition 19. *For all $\mathbf{v} \in \mathbb{D}_K$, it holds true that:*

$$\Pi^\nabla_{\mathcal{R}_K} = \overline{\mathbf{v}} + \hat{\psi}(\mathbf{v})(\mathbf{x} - \mathbf{x_c}), \tag{7.17}$$

where the mean tensor associated to the rigid body motion is given by:

$$\hat{\psi}(\mathbf{v}) = \frac{1}{|K|}\int_K \psi(\mathbf{v})dK = \frac{1}{2|K|}\int_{\partial K}(\mathbf{v}\times\hat{\mathbf{n}} - \hat{\mathbf{n}}\times\mathbf{v})dS, \tag{7.18}$$

with $\psi(\mathbf{v}) = \frac{1}{2}(\nabla\mathbf{v} - \nabla\mathbf{v}^T)$, $\hat{\mathbf{n}}$ *is the normal vector to* ∂K *and* $\overline{\mathbf{v}} = \frac{1}{n_K}\sum_{i=1}^{n_K}\mathbf{v}(\mathbf{x}^i)$ *is the mean of* $\mathbf{v}$ *in the vertices.*

From Definition 19, it can be observed that the term $\hat{\psi}(\mathbf{v})(\mathbf{x} - \mathbf{x_c})$ is related to the rotations and $\overline{\mathbf{v}}$ is related to translations. In the same way, the definition below presents a explicit formula for the $\Pi^{\nabla}_{\mathcal{CS}}$ operator.

Definition 20. *For all* $\mathbf{v} \in \mathbb{D}_K$*, it holds true that:*

$$\Pi^{\nabla}_{\mathcal{CS}} = \hat{\varepsilon}(\mathbf{v})(\mathbf{x} - \mathbf{x_c}), \tag{7.19}$$

where the mean tensor associated to the strain modes is given by:

$$\hat{\varepsilon}(\mathbf{v}) = \frac{1}{|K|}\int_K \varepsilon(\mathbf{v})dK = \frac{1}{2|K|}\int_{\partial K}(\mathbf{v}\times\hat{\mathbf{n}} + \hat{\mathbf{n}}\times\mathbf{v})dS. \tag{7.20}$$

Due to the orthogonality of the projection operators given by $\Pi^{\nabla}_{\mathcal{R}_K}\mathbf{c} = \mathbf{0}$, for all $\mathbf{c} \in \mathcal{CS}_K$ and $\Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{r} = \mathbf{0}$, for all $\mathbf{r} \in \mathcal{R}_K$, it is possible to conclude that $\psi(\Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v}) = \mathbf{0}$ and $\varepsilon(\Pi^{\nabla}_{\mathcal{R}_K}\mathbf{v}) = \mathbf{0}$, for all $\mathbf{v} \in \mathbb{D}_K$. There are other ways to define the projection operator that may lead to differences in the results. For example, Artioli *et al.* (2017) define a single operator that directly maps the displacement field to the strain field. On the other hand, da Veiga *et al.* (2013b) and Mengolini *et al.* (2019) introduce a projection operator that maps a displacement field directly into the polynomial space. Also, it is worth mentioning that the projection operator $\Pi^{\nabla}_{\mathbb{E}_K}$ can be easily obtained by a composition of $\Pi^{\nabla}_{\mathcal{R}_K}$ and $\Pi^{\nabla}_{\mathcal{CS}_K}$ as shown before.

The next step refers to build the discrete bilinear term. But first, in order to guarantee the consistency property it is necessary to show that the residual term $\mathbf{v} - \Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v}$ is orthogonal to every $\mathbf{c} \in \mathcal{CS}_K$. By the bilinear form definition:

$$a_K(\mathbf{c}, \mathbf{v} - \Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v}) = \int_K \sigma(\mathbf{c})\cdot\varepsilon(\mathbf{v} - \Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v})dK = \int_K \sigma(\mathbf{c})\cdot[\varepsilon(\mathbf{v}) - \varepsilon(\Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v})]dK. \tag{7.21}$$

As $\sigma(\mathbf{c})$ is constant for every $\mathbf{c} \in \mathcal{CS}_K$:

$$a_K(\mathbf{c}, \mathbf{v} - \Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v}) = \sigma(\mathbf{c})\cdot\left[\int_K \varepsilon(\mathbf{v})dK - |K|\varepsilon(\Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v})\right] \tag{7.22}$$

By Definition 20, it is possible to see that $\varepsilon(\Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v}) = \hat{\varepsilon}(\mathbf{v})$. Thus,

$$a_K(\mathbf{c}, \mathbf{v} - \Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v}) = \sigma(\mathbf{c})\cdot\left[\int_K \varepsilon(\mathbf{v})dK - |K|\hat{\varepsilon}(\mathbf{v})\right] = 0, \tag{7.23}$$

proving the orthogonality.

For every $\mathbf{v} \in \mathbb{D}_K$ it is possible to write:

$$\mathbf{v} = \Pi^{\nabla}_{\mathcal{R}_K}\mathbf{v} + \Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v} + (\mathbf{v} - \Pi^{\nabla}_{\mathbb{E}_K}\mathbf{v}). \tag{7.24}$$

The last term of this equation refers to the non-polynomial functions residues. In this sense, one obtains:

$$a_K(\mathbf{v},\mathbf{v}) = a_K(\Pi^{\nabla}_{\mathcal{R}_K}\mathbf{v} + \Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v} + (\mathbf{v} - \Pi^{\nabla}_{\mathbb{E}_K}\mathbf{v}), \Pi^{\nabla}_{\mathcal{R}_K}\mathbf{v} + \Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v} + (\mathbf{v} - \Pi^{\nabla}_{\mathbb{E}_K}\mathbf{v})). \tag{7.25}$$

As mentioned before $\varepsilon(\Pi^{\nabla}_{\mathcal{R}_K}\mathbf{v}) = \mathbf{0}$, i.e., no strain energy is associated to the rigid body motions, as expected. Consequently,

$$a_K(\mathbf{v},\mathbf{v}) = a_K(\Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v} + (\mathbf{v} - \Pi^{\nabla}_{\mathbb{E}_K}\mathbf{v}), \Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v} + (\mathbf{v} - \Pi^{\nabla}_{\mathbb{E}_K}\mathbf{v})). \tag{7.26}$$

Thus,

$$a_K(\mathbf{v},\mathbf{v}) = a_K(\Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v}, \Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v}) + 2a_K(\Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v}, (\mathbf{v} - \Pi^{\nabla}_{\mathbb{E}_K}\mathbf{v})) + a_K(\mathbf{v} - \Pi^{\nabla}_{\mathbb{E}_K}\mathbf{v}, \mathbf{v} - \Pi^{\nabla}_{\mathbb{E}_K}\mathbf{v}). \tag{7.27}$$

From equation (7.23), one can observe that:

$$a_K(\mathbf{v},\mathbf{v}) = a_K(\Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v}, \Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v}) + a_K(\mathbf{v} - \Pi^{\nabla}_{\mathbb{E}_K}\mathbf{v}, \mathbf{v} - \Pi^{\nabla}_{\mathbb{E}_K}\mathbf{v}). \tag{7.28}$$

The bilinear form in equation (7.28) presents the consistency and the stability term. In this way, by choosing an adequate stability term, without loss of generality, it is possible to write:

$$a_{h,K}(\mathbf{v},\mathbf{v}) = a_K(\Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v}, \Pi^{\nabla}_{\mathcal{CS}_K}\mathbf{v}) + S_K(\mathbf{v} - \Pi^{\nabla}_{\mathbb{E}_K}\mathbf{v}, \mathbf{v} - \Pi^{\nabla}_{\mathbb{E}_K}\mathbf{v}), \tag{7.29}$$

where S_K is the symmetric bilinear form defined as in Gain *et al.* (2014) and it is detailed in the next section.

Finally the load terms are piecewise constant for $k = 1$ and can be computed similarly to the load term in the VEM formulation for the Poisson Equation. Thus, the body force is given by:

$$f_{b,K,h}(\mathbf{v}) = |K|\hat{\mathbf{b}} \cdot \overline{\mathbf{v}}, \tag{7.30}$$

where $\hat{\mathbf{b}} = \frac{1}{|K|}\int_K \mathbf{b}dK$. In turn, the external traction can be calculated as:

$$f_{s,K,h}(\mathbf{v}) = l_{K,e}\hat{\mathbf{g}} \cdot \overline{\mathbf{v}}, \tag{7.31}$$

where $l_{K,e}$ is the length of edge e and $\hat{\mathbf{g}} = \frac{1}{|K|}\int_K \mathbf{g}dK$. With the formulation of the method presented, the next section will be dedicated to present the implementation.

7.3 Implementation

The implementation strategy is similar to what is done in da Veiga *et al.* (2014). Defining $(\phi_i)_{i\in[1,n_K]}$ as the basis for the space $\mathbb{D}_K$. The choice of the basis here is analogous to the choice for the local virtual element space made in Chapters 3 and 4. For the space of rigid body motion $\mathcal{R}_K$, the basis is given by $\mathbf{r}^1 = (1,0)$, $\mathbf{r}^2 = (0,1)$ and $\mathbf{r}^3 = (x_2 - x_{c,2}, -x_1 + x_{c,1})$ such that

$$\beta_{r,j}(\mathbf{x}) = \sum_{i=1}^{n_K} \phi_i(\mathbf{x})\mathbf{r}^j(\mathbf{x}^i), \tag{7.32}$$

with $j = 1,2,3$. Recalling that $\overline{v_1}$ and $\overline{v_2}$ are the translations on the vertices and $\hat{\psi}_{12}$ is associated with the rotations on the vertices. Thus, this values are directly related to the choice of degrees of freedom. It is possible to write:

$$\Pi^{\nabla}_{\mathcal{R}_K}\mathbf{v} = \beta_{r,1}\overline{v_1} + \beta_{r,2}\overline{v_2} + \beta_{r,3}\hat{\psi}_{12}. \tag{7.33}$$

Also, it is possible to write $\mathbf{v}$ in terms of $(\phi_i)_{i\in[1,n_K]}$ and its values on vertices as:

$$\mathbf{v} = \sum_{i=1}^{n_K} \phi_i(\mathbf{x})\mathbf{v}^i, \tag{7.34}$$

where $\mathbf{v}^i = (v_{i,1}, v_{i,2})$ are the values on the vertices. In matrix form, one shall have:

$$\Pi^{\nabla}_{\mathcal{R}_K}\mathbf{v} = \underline{\mathbf{B}}\ \underline{\Pi}^{\nabla}_{\mathcal{R}_{K,\dagger}}\mathbf{d}, \tag{7.35}$$

where

$$\mathbf{d} = \begin{bmatrix} v_1^1 & v_2^1 & v_1^2 & v_2^2 & \cdots & v_1^{n_K} & v_2^{n_K} \end{bmatrix} \tag{7.36}$$

and $\underline{\Pi}^{\nabla}_{\mathcal{R}_{K,\dagger}} = \underline{\mathbf{D}}_R\ \underline{\Pi}^{\nabla}_{\mathcal{R}_K}$ with

$$\underline{\Pi}^{\nabla}_{\mathcal{R}_K} = \begin{bmatrix} \frac{1}{n_K} & 0 & \frac{1}{n_K} & 0 & \cdots & \frac{1}{n_K} & 0 \\ 0 & \frac{1}{n_K} & 0 & \frac{1}{n_K} & \cdots & 0 & \frac{1}{n_K} \\ \eta_{1,2} & -\eta_{1,1} & \eta_{2,2} & -\eta_{2,1} & \cdots & \eta_{n_K,2} & -\eta_{n_K,1} \end{bmatrix} \tag{7.37}$$

and $\eta_{i,j} = \frac{1}{4|K|}(l_{K,e_{i-1}}\hat{n}_{i-1,j} + l_{K,e_i}\hat{n}_{i,j})$, for $j = 1,2$, is the component associated with the normal vectors. As before, $\underline{\mathbf{B}}$ and $\underline{\mathbf{D}}_R$ are intermediary matrices given by:

$$\underline{\mathbf{B}} = \begin{bmatrix} \phi_1 & 0 & \phi_2 & 0 & \cdots & \phi_{n_K} & 0 \\ 0 & \phi_1 & 0 & \phi_2 & \cdots & 0 & \phi_{n_K} \end{bmatrix} \tag{7.38}$$

and

$$\underline{\mathbf{D}}_R = \begin{bmatrix} 1 & 0 & x_2^1 - x_{c,2} \\ 0 & 1 & -x_1^1 + x_{c,1} \\ 1 & 0 & x_2^2 - x_{c,2} \\ 0 & 1 & -x_1^2 + x_{c,1} \\ \vdots & \vdots & \vdots \\ 1 & 0 & x_2^{n_K} - x_{c,2} \\ 0 & 1 & -x_1^{n_K} + x_{c,1} \end{bmatrix}. \tag{7.39}$$

Analogously, defining a basis for the space of constant strain modes $\mathcal{CS}_K$ as $\mathbf{c}^1 = (x_1 - x_{c,1}, 0)$, $\mathbf{c}^2 = (0, x_2 - x_{c,2})$ and $\mathbf{c}^3 = (x_2 - x_{c,2}, x_1 - x_{c,1})$ such that

$$\beta_{c,j}(\mathbf{x}) = \sum_{i=1}^{n_K} \phi_i(\mathbf{x}) \mathbf{c}^j(\mathbf{x}^i), \tag{7.40}$$

with $j = 1, 2, 3$. Again, considering the choice of basis, it is possible to write:

$$\Pi^{\nabla}_{\mathcal{CS}_K} \mathbf{v} = \beta_{c,1}\hat{\varepsilon}_1 + \beta_{c,2}\hat{\varepsilon}_2 + \beta_{c,3}\hat{\varepsilon}_{12}. \tag{7.41}$$

The matrix format is given by:

$$\Pi^{\nabla}_{\mathcal{CS}_K} \mathbf{v} = \underline{\mathbf{B}}\ \underline{\Pi}^{\nabla}_{\mathcal{CS}_{K,\dagger}}\ \mathbf{d}, \tag{7.42}$$

where $\underline{\Pi}^{\nabla}_{\mathcal{CS}_{K,\dagger}} = \underline{\mathbf{D}}_{CS}\ \underline{\Pi}^{\nabla}_{\mathcal{CS}_K}$, with

$$\underline{\mathbf{D}}_{CS} = \begin{bmatrix} x_1^1 - x_{c,1} & 0 & x_2^1 - x_{c,2} \\ 0 & x_2^1 - x_{c,2} & x_1^1 - x_{c,1} \\ x_1^2 - x_{c,1} & 0 & x_2^2 - x_{c,2} \\ 0 & x_2^2 - x_{c,2} & x_1^2 - x_{c,1} \\ \vdots & \vdots & \vdots \\ x_1^{n_K} - x_{c,1} & 0 & x_2^{n_K} - x_{c,2} \\ 0 & x_2^{n_K} - x_{c,2} & x_1^{n_K} - x_{c,1} \end{bmatrix} \tag{7.43}$$

and

$$\underline{\Pi}^{\nabla}_{\mathcal{CS}_K} = \begin{bmatrix} 2\eta_{1,1} & 0 & 2\eta_{2,1} & 0 & \cdots & 2\eta_{n_K,1} & 0 \\ 0 & 2\eta_{1,2} & 0 & 2\eta_{2,2} & \cdots & 0 & 2\eta_{n_K,2} \\ \eta_{1,2} & \eta_{1,1} & \eta_{2,2} & \eta_{2,1} & \cdots & \eta_{n_K,2} & \eta_{n_K,1} \end{bmatrix}. \tag{7.44}$$

The matrix form of the projection operator $\Pi^{\nabla}_{\mathbb{E}_K}$ is obtained by:

$$\underline{\Pi}^{\nabla}_{\mathbb{E}_{K,\dagger}} = \underline{\Pi}^{\nabla}_{\mathcal{R}_{K,\dagger}} + \underline{\Pi}^{\nabla}_{\mathcal{CS}_{K,\dagger}}. \tag{7.45}$$

By the definition of the intermediary matrix $\underline{\mathbf{B}}$, it is possible to write:

$$\mathbf{u} = \underline{\mathbf{B}}\mathbf{d} \quad \textit{and} \quad \mathbf{v} = \underline{\mathbf{B}}\mathbf{d}. \tag{7.46}$$

Substituting equations (7.42), (7.45) and (7.46) in equation (7.29), the local stiffness matrix is given by:

$$\mathbb{K}_h = |K|\underline{\Pi}^{\nabla}_{\mathcal{CS}_{K,\dagger}}\mathcal{D}(\underline{\Pi}^{\nabla}_{\mathcal{CS}_{K,\dagger}})^T + (\mathbf{I} - \underline{\Pi}^{\nabla}_{\mathbb{E}_{K,\dagger}})^T\mathbf{S}_K(\mathbf{I} - \underline{\Pi}^{\nabla}_{\mathbb{E}_{K,\dagger}}), \tag{7.47}$$

where

$$\mathbf{S}_K = |K|\frac{tr(\mathcal{D})}{tr(\underline{\mathbf{D}}^T_{CS}\underline{\mathbf{D}}_{CS})} \tag{7.48}$$

as defined in Gain *et al.* (2014) and $\mathcal{D}$ is the constitutive operator defined in equation (5.29).

The body force vector and the external traction vector can be written directly as follow:

$$\mathbf{f}_{b,K,h} = |K|\begin{bmatrix} \frac{1}{n_K} & 0 & \frac{1}{n_K} & 0 & \cdots & \frac{1}{n_K} & 0 \\ 0 & \frac{1}{n_K} & 0 & \frac{1}{n_K} & \cdots & 0 & \frac{1}{n_K} \end{bmatrix}\hat{\mathbf{b}} \tag{7.49}$$

and

$$\mathbf{f}_{s,K,h} = |K|\begin{bmatrix} \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} \end{bmatrix}\hat{\mathbf{g}}. \tag{7.50}$$

Chapter 8

Applications

This chapter is dedicated to present some examples regarding the Virtual Element Method applied to linear elasticity. The first application is a square plate under traction which possess and analytical solution. The second application, a non-convex polygon, does not have an analytical solution and, thus, its results are compared against *Ansys*. The third application is the problem of a rectangular plate with a hole. The formulation and analytical results regarding to stress concentration factor are presented. The last application concerns to the chosen complex geometry that is a zeta-shaped pressure armor.

The meshes used in this chapter are quadrilateral and square meshes and, to generate these meshes, the Gmsh software presented in Geuzaine and Remacle (2009) is used. The presented method implementation has influence from various works. The integral computations of normal vectors and vectorization of the process were based on the works of Sutton (2016) and Chen (2018). The method implementation itself was inspired and based on VEM-LAB library by Professor Ortiz, specially regarding to the post-processing, and the *Veamy* software discussed in Chapter 2.

8.1 Unitary Square Plate

The first application is a unitary square plate with the movement restricted in horizontal direction on the left edge and in vertical direction in the bottom edge. A distributed load $t = 1kN/m$ is applied on the right edge and the material parameters are $E = 1MPa$ and $\nu = 0.3$. Accordingly to Artioli *et al.* (2017), the analytical solution for this problem is given by:

$$\begin{aligned} u(x,y) &= \frac{g}{E}x, \\ v(x,y) &= -\frac{\nu g}{E}y. \end{aligned} \tag{8.1}$$

The error is given by the author as:

$$e(u_h, u) = \sqrt{\frac{\sum\limits_{\mathbf{x}\in\tau_h} \|\mathbf{u}(\mathbf{x}) - \mathbf{u_h}(\mathbf{x})\|^2}{\sum\limits_{\mathbf{x}\in\tau_h} \|\mathbf{u}(\mathbf{x})\|^2}}, \tag{8.2}$$

where $\mathbf{x}$ represents a node in the decomposition τ_h. This problem is inspired on Mengolini *et al.* (2019) patch test and Figure 8.1 shows its illustration.

Figure 8.2 shows examples of used quadrilateral and triangular meshes. As mentioned before the meshes were generated using *Gmsh*.

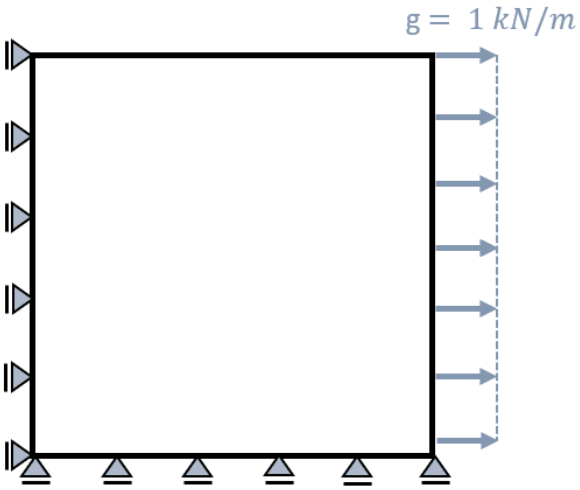
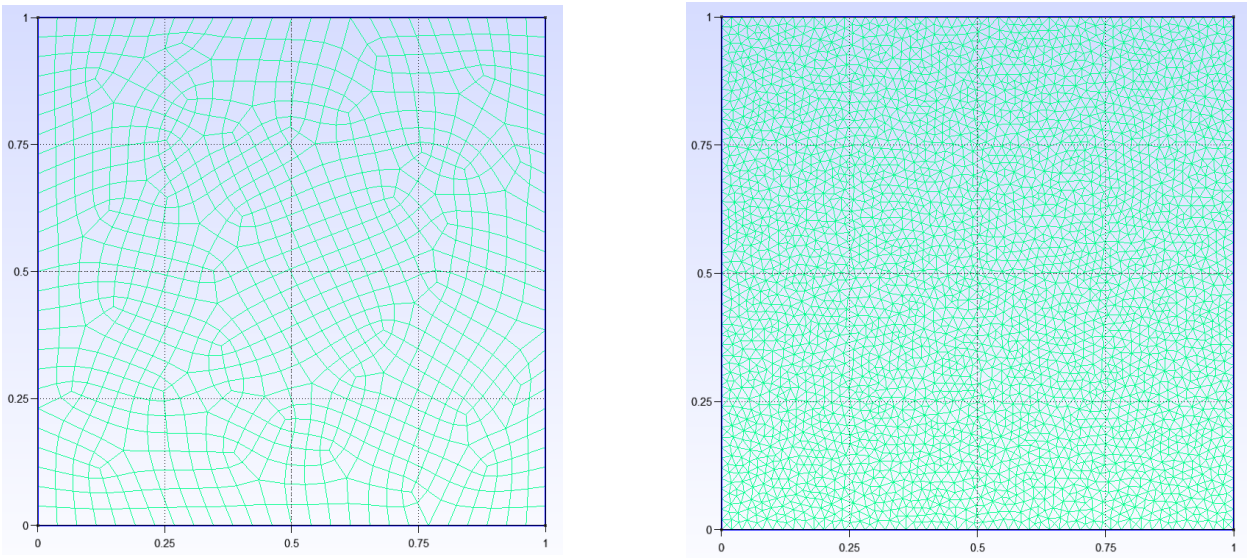


Figure 8.1: *Unitary square plate with distributed load.*
Source: Author



(a) *Quadrilateral mesh*

(b) *Triangular mesh*

Figure 8.2: *Example of meshes, generated with Gmsh, used to perform simulations.*
Source: Author

Table 8.1 shows the error associated to the performed simulations for different element

sizes. Figure 8.3 shows a graphical representation of this table. It is possible to observe that the error values are very close to each other independent of the different element sizes and different element shapes. This fact may indicate the robustness of Virtual Element Method regarding the generalization of meshes. On the other hand, the errors obtained are not close to one obtained in Artioli *et al.* (2017). This may occur because of the difference in the formulations, especially regarding to the choice of the stabilization term. Also, it is possible to see in table 8.1, that the convergence rate of the error is very small. A justification for this occurrence is related to the choice of the stabilization term.

Patch test		
Elements size	$e(u_h, u)$**-Quadrilateral**	$e(u_h, u)$**-Triangle**
0.4	0.0770	0.0767
0.2	0.0743	0.0753
0.1	0.0726	0.0733
0.08	0.0720	0.0718
0.04	0.0718	0.0716
0.02	0.0716	0.0715
0.01	0.0713	0.0712

Table 8.1: *Associated errors for different size of elements regarding the quadrilateral and triangular mesh.*

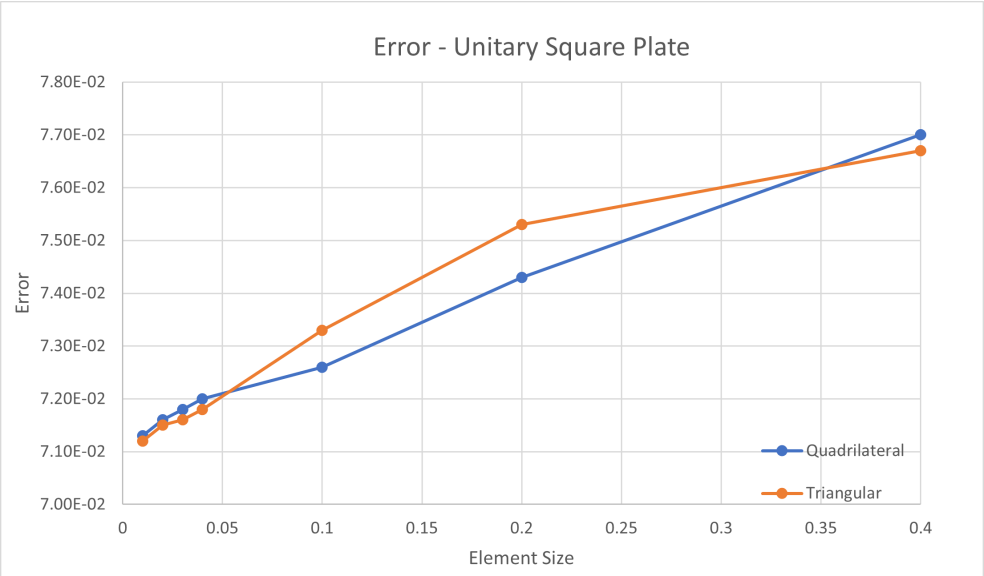
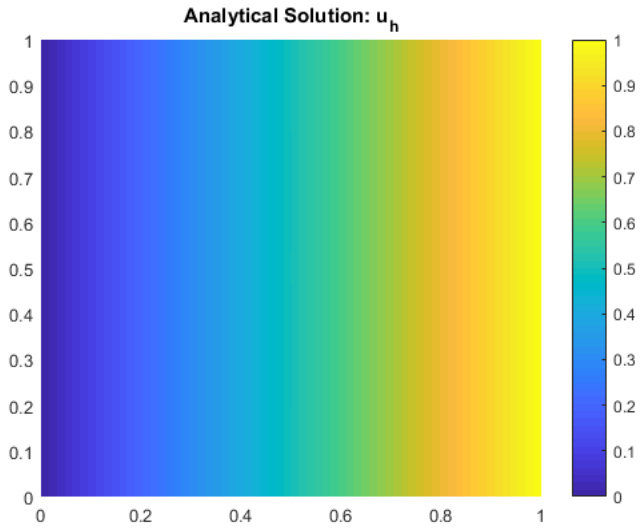
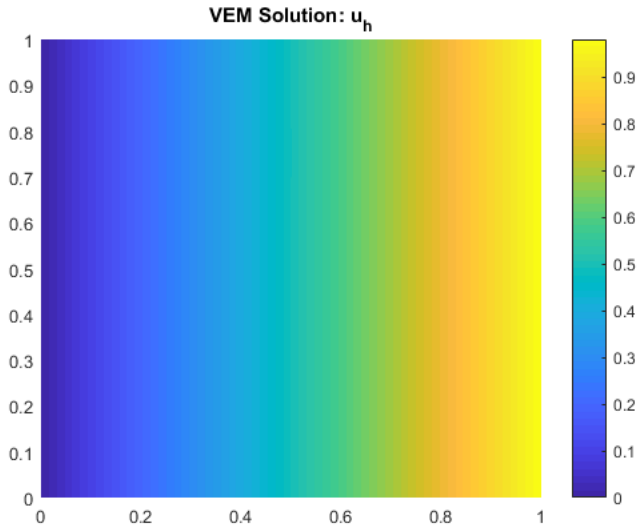


Figure 8.3: *Convergence of VEM for the patch test.*
Source: Author

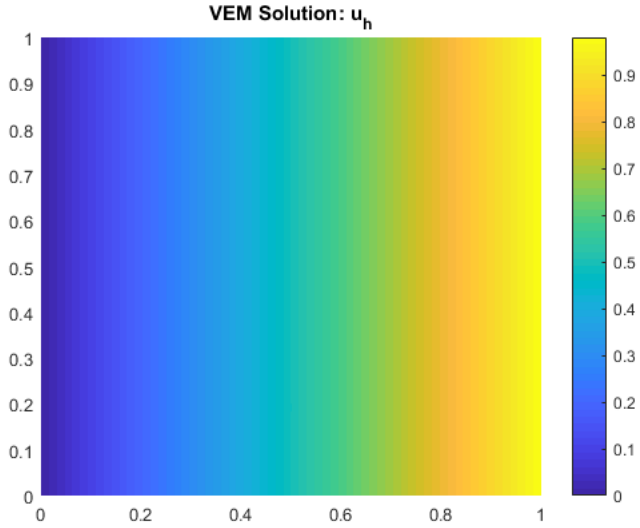
Figures 8.4, 8.5 and 8.6 show the analytical and numerical solution considering a quadrilateral mesh and a triangular mesh with element size of 0.04, respectively. The notation $\|U_h\|$ represents the total displacement. From the presented results it is possible to conclude that VEM converges to the patch test with a satisfactory error order.



(a) *Analytical solution u*

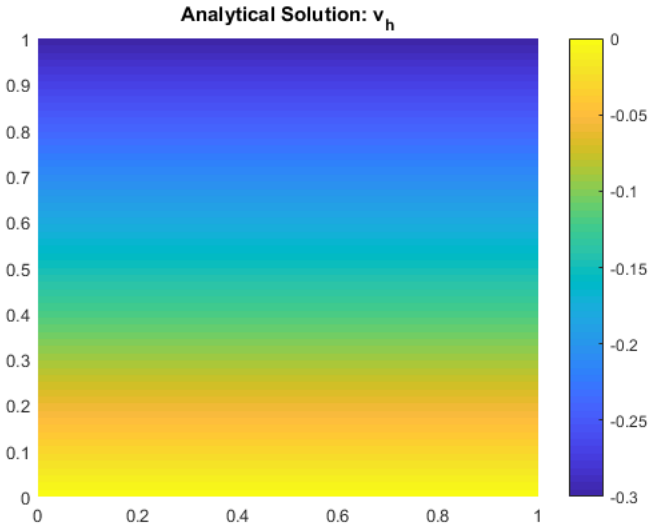


(b) *Quadrilateral Mesh - u_h*

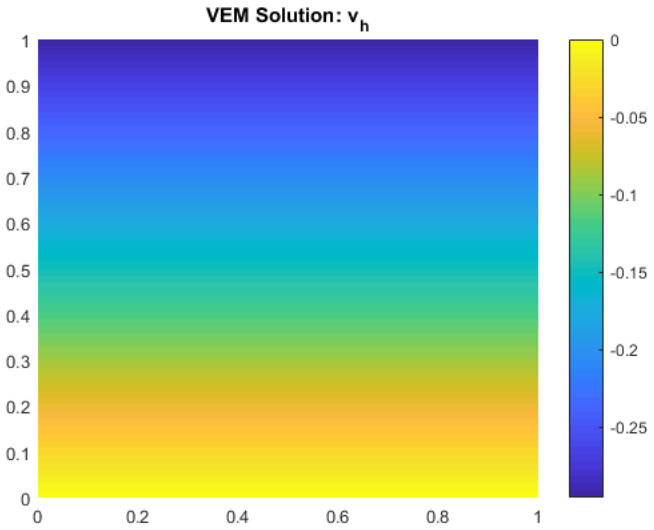


(c) *Triangular Mesh - u_h*

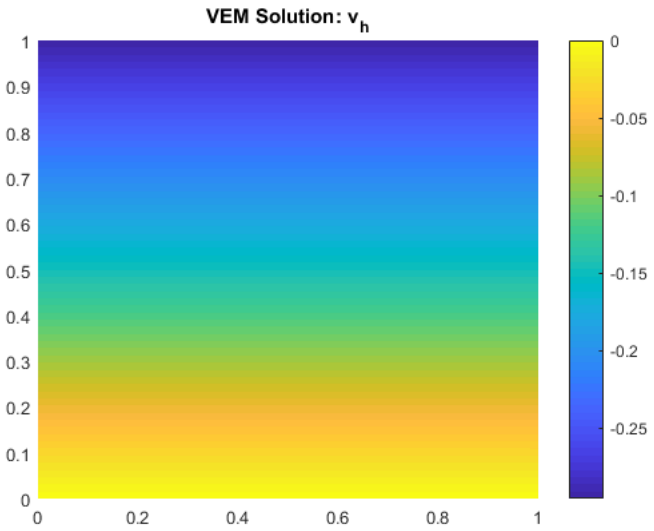
Figure 8.4: *Solution for u component.*
Source: Author



(a) *Analytical solution v*

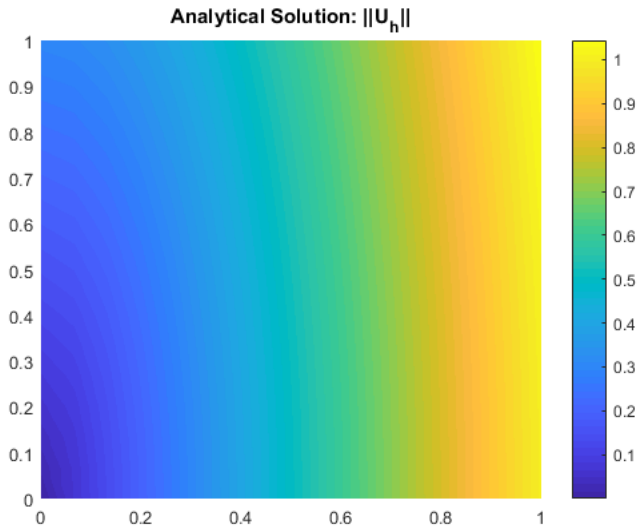


(b) *Quadrilateral Mesh - v_h*

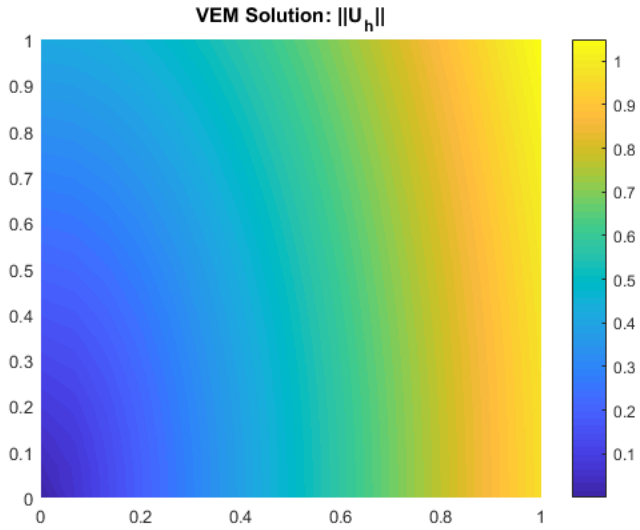


(c) *Triangular Mesh - v_h*

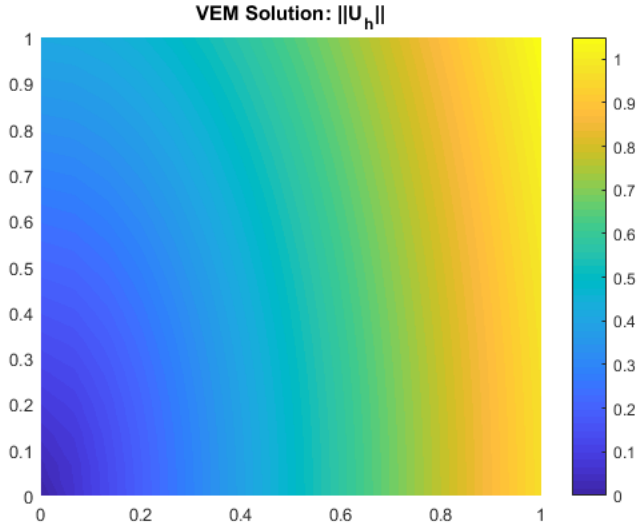
Figure 8.5: *Solution for v component.*
Source: Author



(a) *Analytical solution U*



(b) *Quadrilateral Mesh - U_h*



(c) *Triangular Mesh - U_h*

Figure 8.6: *Total sum solution.*
Source: Author

8.2 Non-convex pentagon

In this application, a direct comparison between the Virtual Element Method and Ansys is made. For each specified element sizes, the maximum absolute values of the horizontal displacement u_h, the vertical displacement v_h and the total displacement U_h are compared. This test uses the same material parameters of the patch test showed in the previous section. The pentagon orthogonal edges are unitary and the inclined edges length is $\frac{\sqrt{2}}{2}$. The geometry with an unitary uniform load is showed in Figure 8.7. For this application, only quadrilateral meshes generated using *Gmsh* were used, as shown in Figure 8.8.

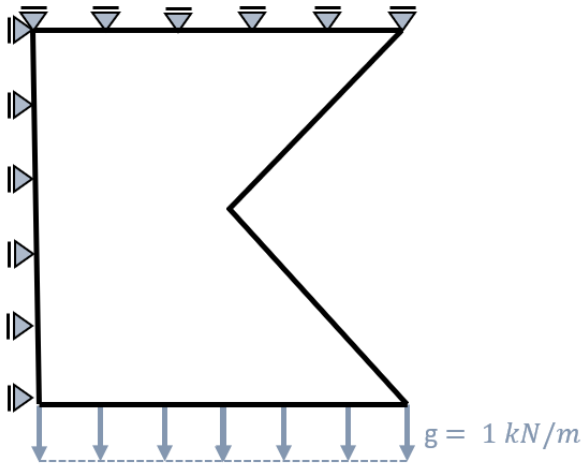


Figure 8.7: *Pentagon with unitary distributed load.*
Source: Author

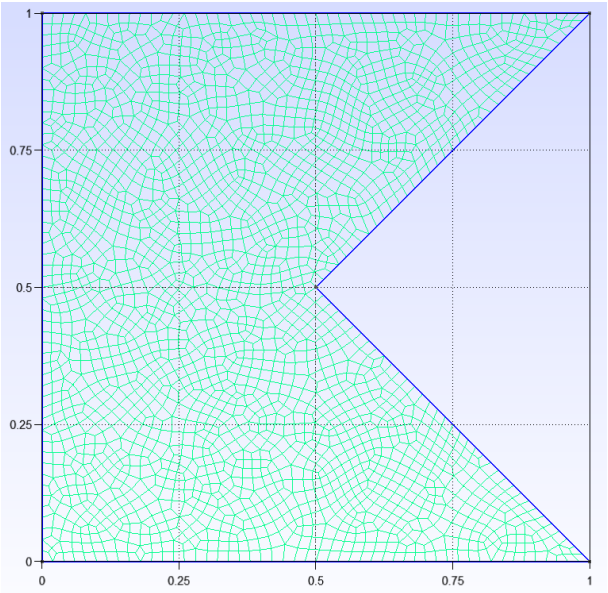


Figure 8.8: *Pentagon quadrilateral mesh using Gmsh.*
Source: Author

Table 8.2 shows the maximum absolute values obtained with VEM while Table 8.3 shows the maximum absolute values obtained with Ansys. And Table 8.4 shows the deviation

between the VEM and Ansys. It can be observed that the convergence rate of the Virtual Element Method is slower than Ansys. Again, this might occur because of the choice of the stability term. To enhance the VEM solution the high-order implementations ($k \geq 2$) should be a solution.

VEM - Pentagon			
Elements Size	$\max(\lvert u_h \rvert)$	$\max(\lvert v_h \rvert)$	$\max(U_h)$
0.1	3.3506	9.0506	9.6509
0.05	3.4148	9.2143	9.8267
0.01	3.4854	9.3904	10.0164
0.008	3.4942	9.4075	10.0354
0.004	3.5045	9.4360	10.0658

Table 8.2: *Maximum absolute values for the pentagon obtained with the Virtual Element Method.*

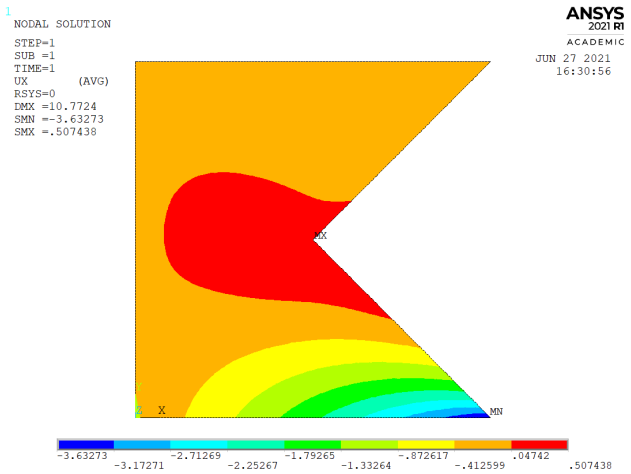
Ansys - Pentagon			
Elements Size	$\max(\lvert u_h \rvert)$	$\max(\lvert v_h \rvert)$	$\max(U_h)$
0.1	3.0875	8.7535	9.2820
0.05	3.3574	9.4226	10.0030
0.01	3.6305	10.1330	10.7640
0.008	3.6327	10.1410	10.7720
0.004	3.6653	10.2210	10.8580

Table 8.3: *Maximum absolute values for the pentagon obtained with Ansys.*

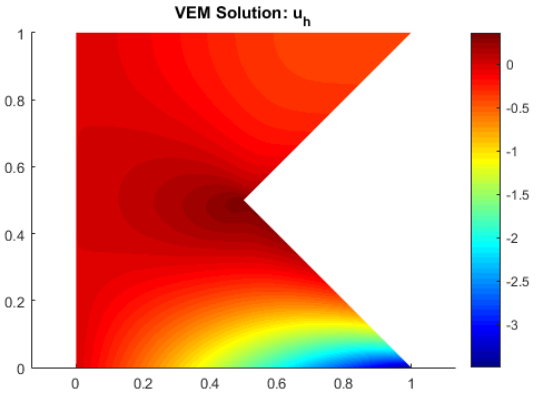
Deviation - Pentagon			
Elements Size	$\max(\lvert u_h \rvert)$	$\max(\lvert v_h \rvert)$	$\max(U_h)$
0.1	0.0852	0.0339	0.0397
0.05	0.0171	0.0221	0.0176
0.01	0.0400	0.0733	0.0695
0.008	0.0381	0.0723	0.0684
0.004	0.0439	0.0768	0.0730

Table 8.4: *Deviation regarding maximum absolute values for the pentagon.*

Figures 8.9, 8.10 and 8.11 shows the numerical solution of VEM and Ansys. It is possible to conclude that the Virtual Element Method presents a satisfactory behavior when compared to a commercial finite element software. Although, for simple geometries with triangular or quadrilateral meshes the advantage of using VEM instead of FEM is not clear.

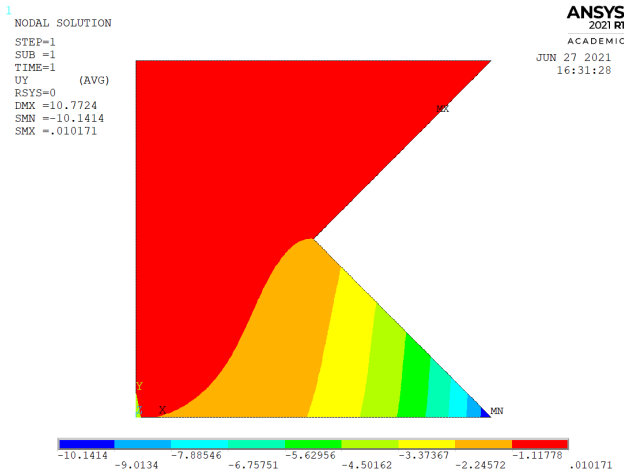


(a) *Ansys solution* u_h

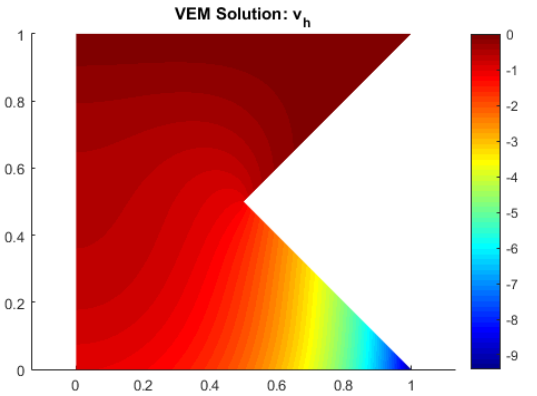


(b) *VEM Solution* u_h

Figure 8.9: *Numerical solution* u_h *for non-convex pentagon with element size equals to 0.008.*
Source: Author

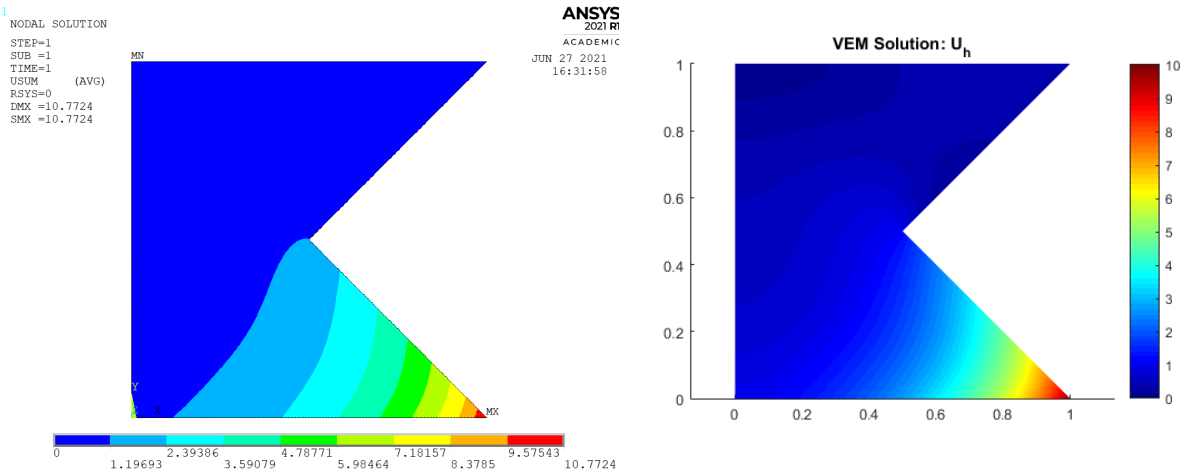


(a) *Ansys solution* v_h



(b) *VEM Solution* v_h

Figure 8.10: *Numerical solution* v_h *for non-convex pentagon with element size equals to 0.008.*
Source: Author



(a) *Ansys solution* U_h **(b)** *VEM Solution* U_h

Figure 8.11: *Numerical solution* U_h *for non-convex pentagon with element size equals to 0.008.*
Source: Author

8.3 Thin plate with a hole

This example refers to a thin rectangular plate with a circular hole in the middle. The plate has width W of $60cm$, height H of $10cm$ and a central hole of diameter D of $1cm$. It is assumed that the thickness is much smaller than the width with value of $t = 1cm$. Thus, a plane state can be considered. Also, a stress field of $\sigma_F = 1000kNm^{-2}$ is applied on the edges as shown in Figure 8.12. It is important to mention that σ_F is associated to a force F and can be analytically calculated as

$$\sigma_F = \frac{F}{tH}. \tag{8.3}$$

The meshes used are quadrilateral with a refinement near the hole generated by *Gmsh*. Figure 8.13 shows a example of the mesh considering the double symmetry conditions.

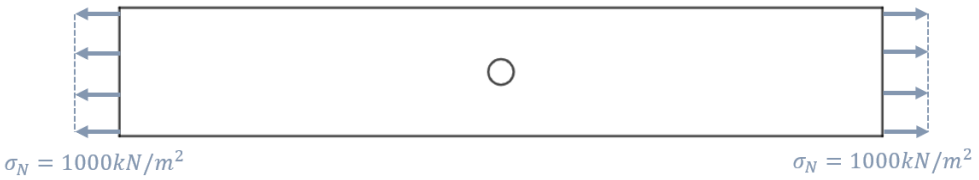


Figure 8.12: *Geometry of a thin rectangular plate with a hole geometry.*
Source: Author

The main goal of this example is to evaluate the stress concentration in the rectangular plate with a central hole for that the concentration factor (CF) is used. Accordingly to Young and Budynas (2002), the CF can be calculated as:

$$CF = \frac{k_{CF}\sigma_{nominal}}{\sigma_F}, \tag{8.4}$$

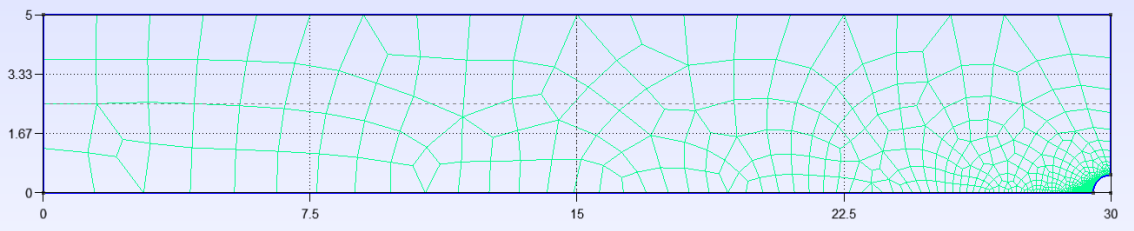


Figure 8.13: *Quadrilateral mesh with refinement near to the hole.*
Source: Author

where

$$k_{CF} = 3 - 3.13\frac{D}{H} + 3.66\left(\frac{D}{H}\right)^2 - 1.53\left(\frac{D}{H}\right)^3 \tag{8.5}$$

and

$$\sigma_{nominal} = \frac{F}{t(H-D)}. \tag{8.6}$$

Equation 8.4 can just be applied to $\frac{W}{H} \geq 5$ and $D < H$ because under these condition the edges will not influence in the analytical result. For the geometrical configuration of the plate described earlier, the analytical value is $CF = 3.0340$. The error is calculated as:

$$Error(\%) = \frac{|CF - CF_{numerical}|}{CF}100. \tag{8.7}$$

Table 8.5 shows the concentration factor numerical values calculated with VEM for different numbers of elements and the associated error with respect the analytical value. Figures 8.14 and 8.15 show the graphical representation of this table.

Concentration Factor		
Number of Elements	**VEM - CF**	**Error (%)**
1995	2.9953	1.2761
6204	3.0183	0.5164
8001	3.0238	0.3361
14790	3.0.283	0.1866
17352	3.0.284	0.1843
22383	3.0293	0.1559
38922	3.0303	0.1206

Table 8.5: *Concentration factor and error for the rectangular plate with a central hole.*

It is possible to observe from the results that the associated error is satisfactory small even for a few elements (near 1% for 1995 elements). Although, as it was seen earlier, the results rate tend to stabilize when the number of elements increases. Again, that might be associated to the choice of the stabilization term or even to the definition of the projector operator.

The Virtual Element Method showed satisfactory results for simpler geometries. The next step is to apply the method for a complex geometry.

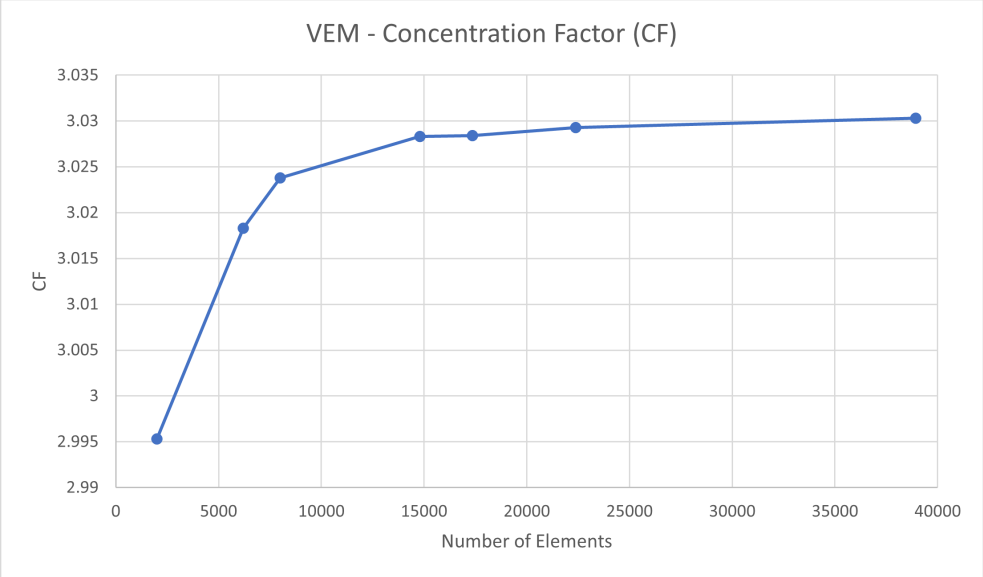


Figure 8.14: *Concentration factor calculated with VEM.*
Source: Author

8.4 Complex geometry: zeta-shaped pressure armor

A simplified model of a zeta-shaped pressure armor presented in Mendonça (2016) is considered as a two dimensional model with left and right sides with movement restricted and a distributed load applied on the top as shown in Figure 8.16. Also, it is important to mention that the problem was modeled considering the plain strain state. The elastic module is equal to $E = 207GPa$ and the Poisson coefficient is $\nu = 0.3$. For more details and a contextualization abou the zeta-shaped pressure armor refer to Appendix C.

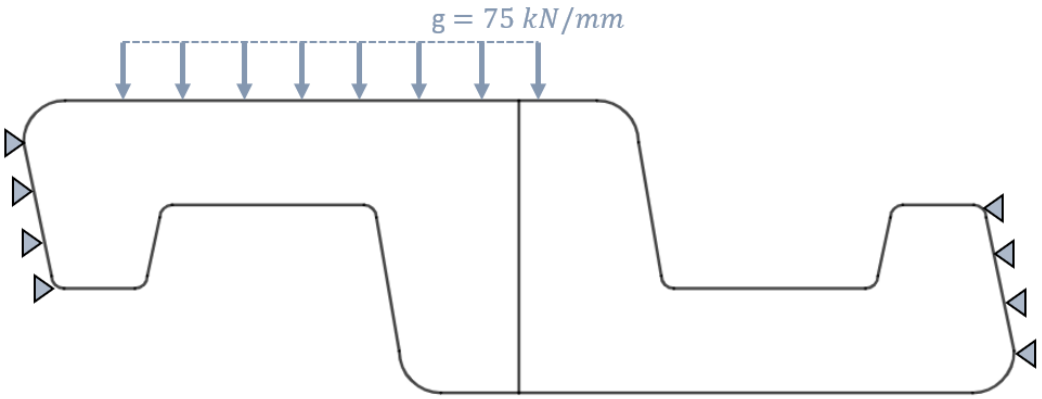


Figure 8.16: *Simplified model of a zeta-shape profile.*
Source: Author

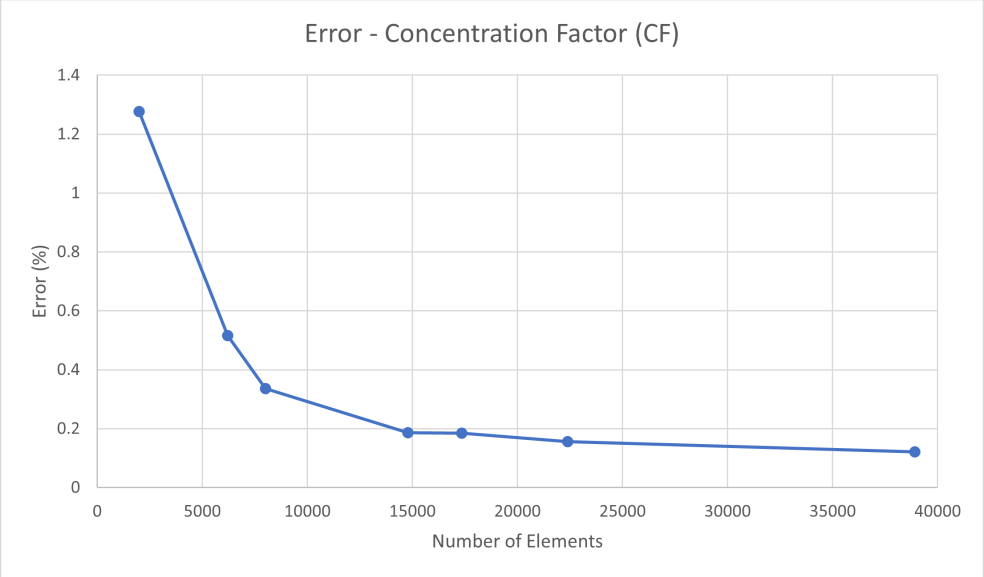


Figure 8.15: *Error between the numerical and analytical values of the concentration factor calculated with VEM.*

Source: Author

The approach here considers a model with 120227 nodes and a quadrilateral mesh in Ansys as reference solution and compare the Virtual Element Method numerical solution with it. Figure 8.17 shows an example of a mesh generated with *Gmsh*. The reference solution is shown in Figure 8.18. The chosen metric to compare the VEM with Ansys is the maximum absolute value for u_h, v_h and U_h. Regarding to the VEM, only quadrilateral meshes are considered.

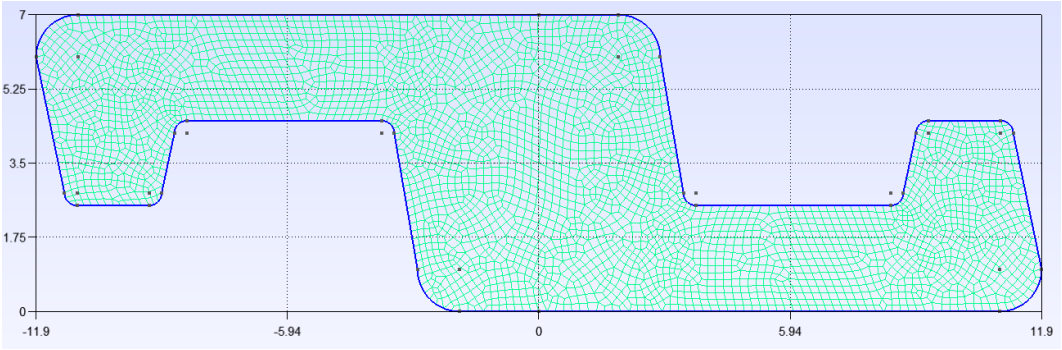
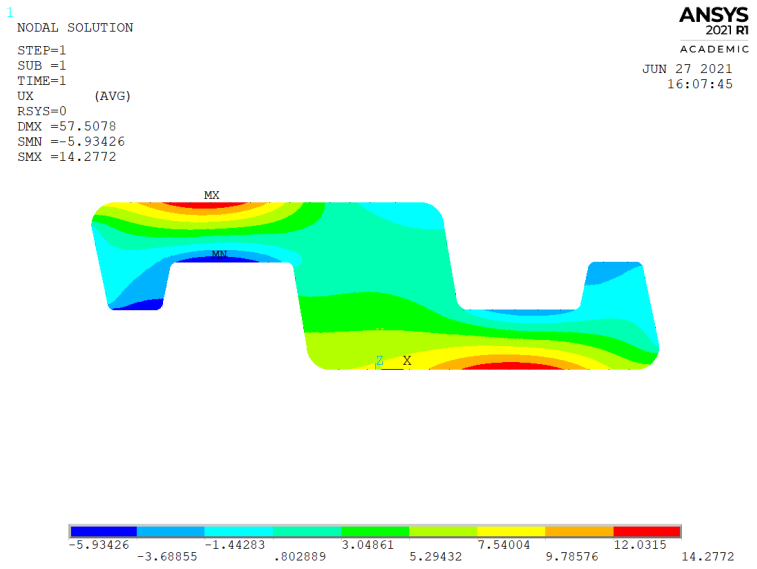
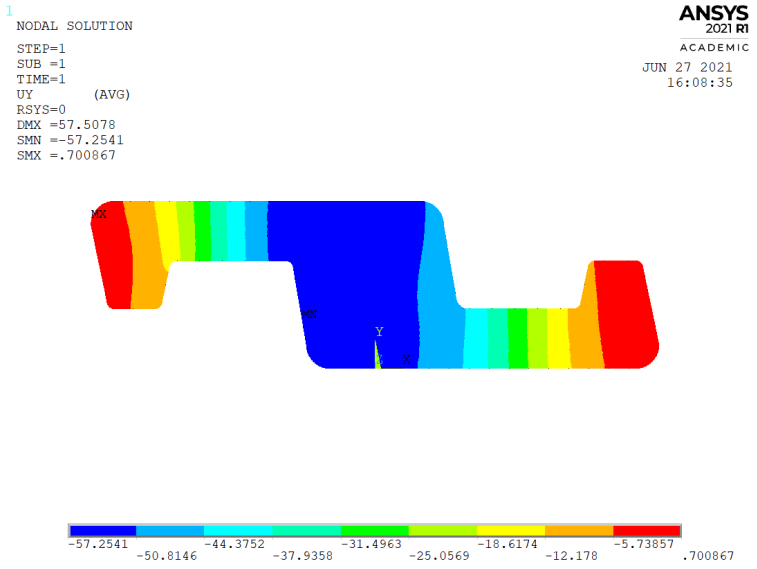


Figure 8.17: *Zeta-shape profile with quadrilateral mesh generated with Gmsh.*

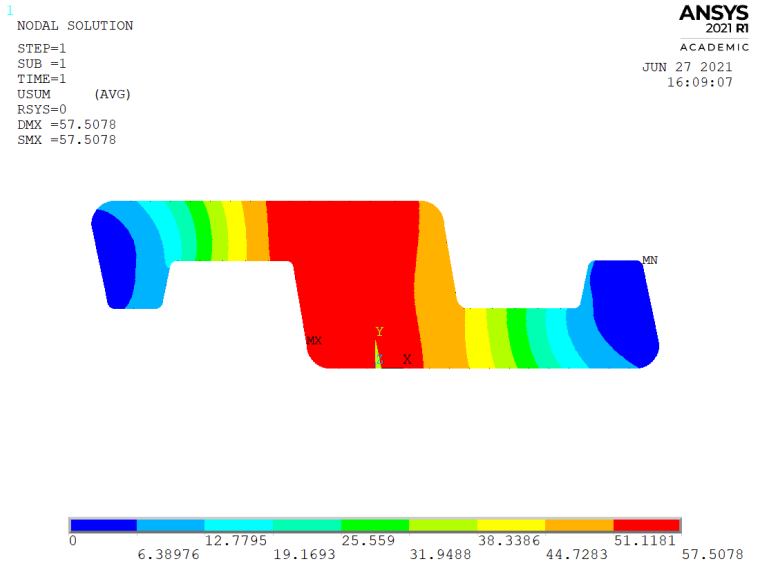
Source: Author



(a) *Reference solution* u_h



(b) *Reference solution* v_h



(c) *Reference solution* U_h

Figure 8.18: *Reference solution generated with Ansys for 120227 nodes.*

Source: Author

Table 8.6, shows the deviation between the maximum value obtained with the VEM and the maximum values in the reference solution. Figures 8.19 and 8.20 show the graphical representation of this table. The convergence rate might be related to the choice of the stabilization term as discussed in Wriggers *et al.* (2016). This is justified once the stabilization term is responsible to handle the non-polynomial terms related to the projection operator. Thus, the projection of the displacement field onto the polynomial space shall have a large residual associated. It is worth mentioning that for the Poisson Equation formulation presented in Chapters 3 and 4 the choice of the stabilization term does not have significant influence on the final results as proved in da Veiga *et al.* (2017b). Yet there is no clear methodology for choosing the adequate stabilization term.

In the case of the complex geometry, another factor that may interfere in the error convergence rate is the choice of the basis for the virtual element space. Probably the chosen basis is a rough representation of the true behavior of the geometry. Changing the basis would clearly imply in the alteration of the intermediary matrices and might also imply in a greater computational cost. Different choices for the basis are not yet largely explored in the literature.

Deviation - Zeta-shape profile			
Number of Nodes	$\max(\|u_h\|)$	$\max(\|v_h\|)$	$\max(U_h)$
333	16.4371	14.7079	14.7725
475	14.6123	13.6303	13.6381
897	10.6689	10.3526	10.3600
2365	8.0977	8.0521	8.0364
4237	7.9137	7.7766	7.7637
15577	6.6954	6.9201	6.8971
19314	6.5645	6.823	6.7970
27901	6.3525	6.6498	6.6234
51541	6.1760	6.4995	6.4718

Table 8.6: *Deviation between the VEM and the reference solution in Ansys for the zeta-shape profile.*

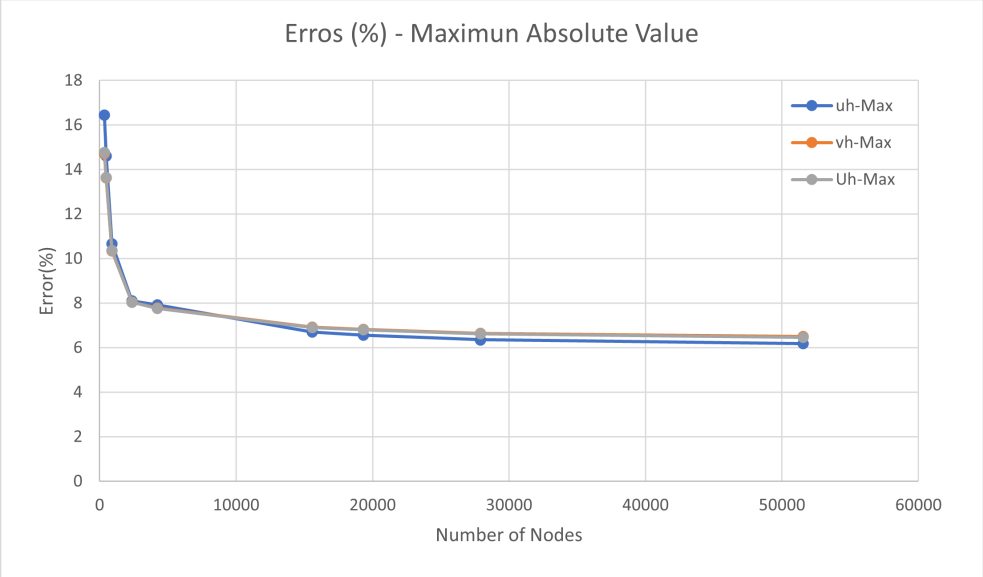


Figure 8.19: *Deviation in percentage between the reference model and the VEM.*
Source: Author

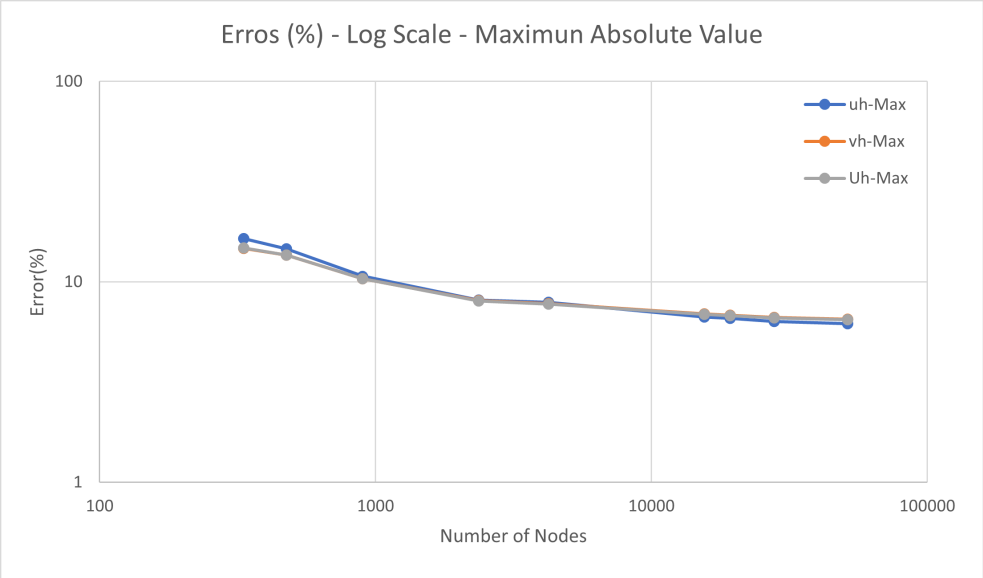
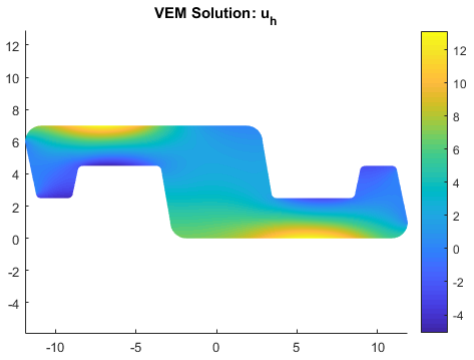
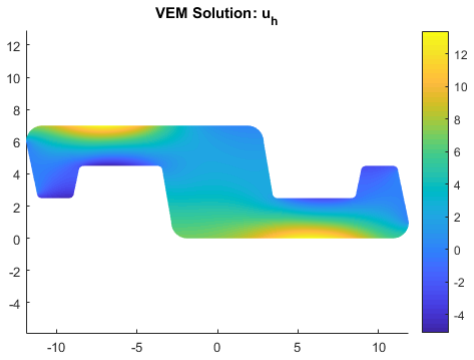


Figure 8.20: *Deviation in percentage between the reference model and the VEM in logarithm scale.*
Source: Author

It also can be seen in Table 8.6, that for 51541 nodes, less than a half of the number of nodes of the Ansys model, the associated deviation is around 6%. This result is in accordance to the literature regarding to the Virtual Element Method performance regarding to complex geometries. This results are related to the fact that the VEM is more flexible regarding to the mesh quality, being able to achieve better results with a similar mesh in the parts of the domain where the geometry is, for example, related to regions where the curvature changes abruptly or with many curved components. Figures 8.21, 8.22 and 8.23, shows the results obtained with the VEM of u_h, v_h and U_h.

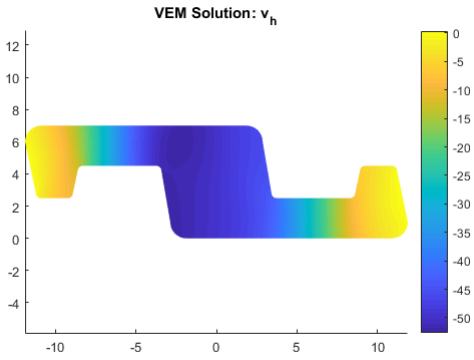


(a) *u_h for 2365 Nodes*

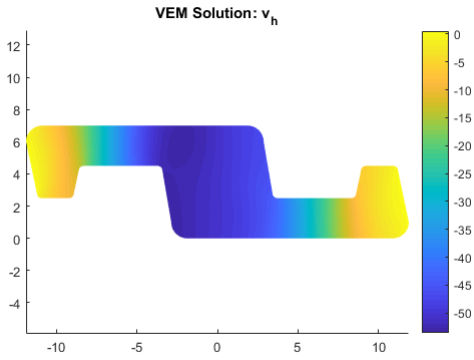


(b) *u_h for 51541 Nodes*

Figure 8.21: *VEM solution u_h for the zeta-shape profile.*
Source: Author

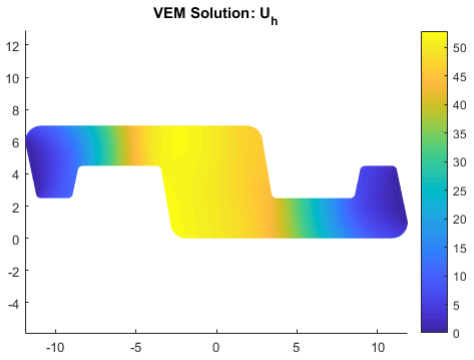


(a) *v_h for 2365 Nodes*

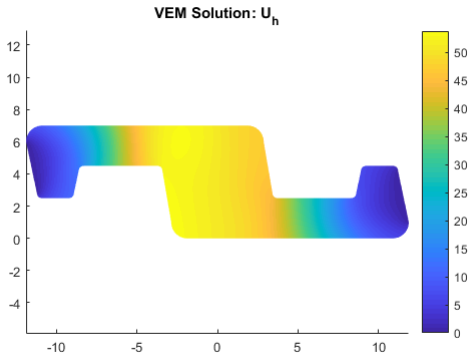


(b) *u_h for 51541 Nodes*

Figure 8.22: *VEM solution v_h for the zeta-shape profile.*
Source: Author



(a) *U_h for 2365 Nodes*



(b) *U_h for 51541 Nodes*

Figure 8.23: *VEM solution U_h for the zeta-shape profile.*
Source: Author

Chapter 9

Conclusions

The first part of this work consists on the formulation of Virtual Element Method for the Poisson Equation. The details of the model are deeply discussed and it is possible to observe that the main idea of the method is to work with any simple polygon as discretization element and compute the functions implicitly using projection opertors. Inspired on da Veiga *et al.* (2014), a matrix framework is presented in order to provide a guideline for the implementation of VEM, showing how to compute each matrix. Then, the method is implemented for a problem with a analytical solution for a square uniform mesh and the Voronoi mesh. The results show the convergence of VEM and that the implementation using the uniform mesh outperformed the implementation for the Voronoi mesh.

In order to apply Virtual Element Method to an engineering problem, the Saint Venant torsion formulation is presented. This formulation is transformed into the Poisson Equation using the Prandtl Stress Function Φ. The equation is solved numerically using the Virtual Element Method, the Finite Element Method and the Finite-Difference Method. The torsion constant I_t is calculated and it is possible to observe that VEM and FEM has a similar performance but FDM presents a slower convergence. In this case, VEM is also implemented for the uniform and the Voronoi mesh but here the Voronoi mesh implementation shows better results, indicating the influence of the right-hand side term f. It is important to mention that FEM is just implemented for the square uniform mesh. In the end, the membrane analogy is briefly discussed.

Virtual Element Method provides expected results for simple geometry domains (unitary square) as can be verified in the literature. One advantage of VEM when compared to FEM is the fact that VEM can use any simple polygon as element of discretization. Thus, the stiffness matrix and the load vector can be computed directly and no isoparametric elements are needed, although the Virtual Element Method formulation is more complicated than the classic Finite Element Method.

The second part of this work is related to the formulation and application of the Virtual Element Method in the linear elasticity context. In the literature there are different variations for the formulation of VEM to this case. However, it is possible to observe a general

pipeline for the VEM formulation and the case of linear elasticity context can be seen as an extension of the Poisson Equation case. First the partial differential equations are arranged into the weak form. This step is general for the Finite Element Method and for the Virtual Element Method. The next step consists on discretizing this weak formulation by choosing and adequate domain decomposition. In the VEM case, any simple polygon can be chosen as discretization element. Then, the virtual element space is constructed upon the definition of the degrees of freedom. It is important to mention that the choice of degrees of freedom is the same for both the Poisson Equation and for the linear elasticity context. In order to define the discrete bilinear form. the projection operator is introduced. The idea is to project function that are not known in a first moment from the local virtual element space to a subspace of a polynomial space. The bilinear form is defined to satisfy both the consistency and stability criteria. To ensure stability, a symmetric bilinear term S_K is introduced. The load term can be constructed analogously to the bilinear form.

In this work, the formulation presented on Gain *et al.* (2014) and Ortiz-Bernardin *et al.* (2019) are chosen, once the implementation framework is more familiar for engineers. Although, the model is restricted to the linear case ($k = 1$). Three examples of applications are presented regarding to a problem with analytical solution, a non-convex pentagon geometry and a plate with a hole in plain state context. Then a complex geometry related to a pressure armor is presented and simulations with VEM and FEM are compared.

In all use cases, it is possible to observe a slow convergence rate. This may occur due to the choice of the stability term. Once this term is related to the non-polynomial terms projection residues, some choices may not lead to an adequate representation of the non-polynomial functions behavior. Other aspect that may have influence in the convergence rate is the choice of the basis for the virtual element space. Other basis shall represent the displacement field with bigger accuracy and precision.

Regarding to the complex geometry, it is possible to see that the Virtual Element Method presented satisfactory results, once with almost half of the number of nodes used in *Ansys*, the VEM provided a deviation of around 6%. It is expected that with different choices of stability term, this deviation decreases. Thus, as future work one shall try different stability term to investigate their impact on complex geometries. Also, the Virtual Element should be implemented to higher orders ($k > 1$) and tested with the complex geometry.

Appendix A

Mathematical Tools

In order to fill the gaps of pure mathematics, some support material concerning to some classical works were used. Before going through more advanced topics, a review of main results and concepts of real analysis was made, based mostly on Bartle (1982) and Rudin (1987). Specifically for Measure Theory, the work of Isnard (2013) was used as the book is very illustrative with examples and introduce topics using a clear language. For Functional Analysis, the work of Botelho *et al.* (2015) and Lax (2002) were mainly used. The book of Brezis (2010) was particularly interesting because it builds the partial differential equation theory upon the optics of functional analysis. The main reference for partial differential equation theory was Evans (2010). This chapter is dedicated to present some mathematical results used in this work. Even though proofs are suppressed in the text, the references to find them are indicated.

First, the inequalities used in work are presented. The proof for each inequality can be found in Evans (2010) and Furtado (2012).

Theorem 5 (Hölder Inequality)**.** *Let $f \in L^p(U)$ and $g \in L^q(U)$ with $1 \leq p, q \leq \infty$ and $\frac{1}{p} + \frac{1}{q} = 1$. Then, it holds*

$$\|fg\|_{L^1(U)} \leq \|f\|_{L^p(U)} \|g\|_{L^q(U)}. \tag{A.1}$$

The Cauchy-Schwarz Inequality is particular case of Hölder Inequality when $p = q = 2$. Thus, it can be written as:

$$\|fg\|_{L^1(U)} \leq \|f\|_{L^2(U)} \|g\|_{L^2(U)}. \tag{A.2}$$

The Poincaré Inequality relates a function of Sobolev Space to its gradient by a constant M. The following theorem is an adaptation to whats is presented in Furtado (2012):

Theorem 6 (Poincaré Inequality)**.** *Let U be limited and $1 \leq p \leq n$. There exists a constant M dependent of p and U, such that*

$$\|f\|_{L^p(U)} \leq M \|\nabla f\|_{L^p(U)}, \tag{A.3}$$

for all $f \in W_0^{1,p}(U)$.

Following the notation of Evans (2010) and Furtado (2012), considering k a positive integer and α a mulit-index, function spaces are denoted by:

- $C(U) = \{f : U \longrightarrow \mathbb{R} \mid f \text{ is continuous}\}$,
- $C(\overline{U}) = \{f \in C(U) \mid f \text{ is uniformly continuous in limited subsets of } U\}$,
- $C^k(U) = \{f : U \longrightarrow \mathbb{R} \mid D^\alpha f \text{ exists and is continuous in } U, \forall |\alpha| \leq k\}$,
- $C^k(\overline{U}) = \{f \in C^k(U) \mid D^\alpha u$
  $\text{exists and is uniformly continuous in limited subsets of } U, \forall |\alpha| \leq k\}$,
- $C^\infty(U) = \{f : U \longrightarrow \mathbb{R} \mid f \text{ is infinitely differentiable}, \forall |\alpha| \leq k\}$.

The subscript c that may come with the notation above (e.g $C_c^k(U)$, $C_c^\infty(U)$) refers to functions with compact support. Recalling that the support of a function is the smallest closed subset of the domain where the function is not zero.

The definition of Banach space is related to the Cauchy Sequence convergence in normed spaces. Recalling that in Cauchy Sequences the terms start to get really close when the sequence tends to infinity.

Definition 21. *Let* $\{u_r\}_{r\in\mathbb{N}}$ *be a sequence in a real linear normed space* B *with a norm* $\|\cdot\|_B$. *The sequence* $\{u_r\}_{r\in\mathbb{N}}$ *is called Cauchy Sequence if, given* $\varepsilon > 0$, *there exists* $N > 0$, *such that*

$$\|u_i - u_j\|_B < \varepsilon \quad \forall i, j \geq N. \tag{A.4}$$

Definition 22 (Banach Space)**.** *A Banach space* B *is a normed space where all Cauchy Sequence converge.*

Next, the definition of inner product and Hilbert space are given following Evans (2010).

Definition 23. *Let* H *be a linear space,* $f, g, h \in H$ *and* $\alpha \in \mathbb{R}$. *The mapping* $(\cdot,\cdot)_H : H \times H \longrightarrow \mathbb{R}$ *is a inner product if it satisfies:*

1. $(f, g + h)_H = (f, g)_H + (f, h)_H$,
2. $(f, g)_H = (g, f)_H$,
3. $(f, \alpha g)_H = \alpha(f, g)_H$,
4. $(f, f)_H \geq 0$, *for all* $f \in H$,
5. $(f, f)_H = 0 \Leftrightarrow f = 0$.

Definition 24 (Hilbert Space)**.** *Given the inner product* $(\cdot,\cdot)_H$ *and* $f \in H$ *the associated norm is* $\|f\| = (f, f)_H$. *The Hilbert Space is a Banach Space with a inner product that induces the norm.*

Appendix B

A brief discussion about mesh regularity

This chapter is dedicated to briefly discuss the error analysis regarding some assumptions that are made in da Veiga *et al.* (2013a) and da Veiga *et al.* (2017b). The theorems are presented and are followed by some intuitive discussion. The main motivation for this chapter is the work of da Veiga *et al.* (2017b). An analysis of the stability term given by (3.64) and (3.63) is presented in paper but in this project the analysis will be restricted to the term given by (3.64).

First, the following definition shall be made:

Definition 25. *Given the Sobolev Space $H_0^1(\Omega)$ and a polygon $K \in \tau_h$ $V_{h,K,k}$, the subspace of sufficiently regular function that the stability term S_K exists, is defined by $\mathcal{H}_K \subseteq H_0^1(\Omega)|_K$. Also, the semi-norm induced by the stability term is given by*

$$|v|^2_{\mathcal{H}_K} = a_K(\Pi^\nabla v, \Pi^\nabla v) + S_K((\mathbf{I} - P_0)v, (\mathbf{I} - P_0)v), \tag{B.1}$$

for all $v \in V_{h,K,k} + \mathcal{H}_K$. Thus, the global semi-norm is given by

$$|v|^2_{\mathcal{H}} = \sum_{K \in \tau_h} |v|^2_{\mathcal{H}_K}. \tag{B.2}$$

By this definition, the following set of hypothesis can be made:

Hypothesis 5. *For all $v \in V_{h,K,k}$ and for all $q \in \mathbb{P}_k(K)$, it is true that*

$$a_K(v, v) \leq C_5(K)|v|^2_{\mathcal{H}_K} \tag{B.3}$$

and

$$|q|^2_{\mathcal{S},K} \leq C_6(K)a_K(q, q). \tag{B.4}$$

It is important to mention that hypothesis (B.4) is weaker than the stability criterion presented in the set of hypothesis 2 because there an estimate for all $v \in V_{h,K,k}$ was required. And now, it is sufficient to analyze the polynomials $q \in \mathbb{P}_k(K)$. Therefore, a more general

convergence result can be presented when compared to convergence in Theorem 2. The following theorem is also more general because it retrieves the semi-norm information.

Theorem 7. *Assuming that (B.3) and (B.4) are true, given that the solution of the continuous problem (3.10) satisfies $u|_K \in \mathcal{H}_K$, for all $K \in \tau_h$. Then, for all $u_i \in V_h$ and for all approximation of u_π that is piecewise in $\mathbb{P}_k(K)$, it holds that for the discrete solution u_h of the discrete problem:*

$$|u - u_h|_{H^1(\Omega)} \leq C_e(h) \left(\tilde{F}_h + |u - u_I|_{\mathcal{H}} + |u - u_I|_{H^1(\Omega)} + |u - u_\pi|_{\mathcal{H}} + \|u - u_\pi\|_{H^1(\Omega)} \right) \quad \text{(B.5)}$$

Defining $\tilde{C}_e(h) = \max_{K \in \tau_h}\{1, C_6(K)\}$, $C_5(h) = \max_{K \in \tau_h}\{C_1(K)\}$, $C_\alpha(K) = \max\{1, C_5(K)C_6(K)\}$ and $C_\alpha(h) = \max_{K \in \tau_h}\{C_\alpha(K)\}$, the estimate for constant $C_e(h)$ is

$$C_e(h) = \max\{1, \tilde{C}_e(h)C_5(h), \tilde{C}_e(h)^{3/2}\sqrt{C_\alpha(h)C_5(h)}\}. \quad \text{(B.6)}$$

In Section 3.2 the set of hypothesis 3 was presented regarding to mesh regularity. Using these hypothesis, it is shown in the classical works da Veiga *et al.* (2013a) and da Veiga *et al.* (2014) a theorem concerning to the projection error.

Theorem 8. *Let the set of hypothesis 3 holds true. Let $u \in H^s(\Omega)$, with $s > 1$, be the solution for the continuous problem in (3.10). And, let $u_h \in V_h$ be the solution for the discrete problem. Then,*

$$\|u - u_h\|_{H^s(\Omega)} \leq h^{s-1}\|u\|_{H^s(\Omega)}, \quad \text{(B.7)}$$

with $1 < s \leq k + 1$.

Although, in da Veiga *et al.* (2017b) some less restrictive hypothesis regarding to mesh regularity are made as shown below.

Hypothesis 6. *Denoting by $l_{K,e}$ the length of an edge $e \in \partial K$:*

1. *there exits a real number $\gamma > 0$ such that all elements $K \in \tau_h$ are star-shaped with respect to a ball B_K with radius $R_K \geq \gamma h_K$ and center $\mathbf{x}_K$,*
2. *there exits $\sigma \in \mathbb{N}$ such that $n_K \leq \sigma$, for all $K \in \tau_h$.*

The first hypothesis in the set 3 and 6 are the same. However, the second hypothesis in 6 is much weaker than in 3 because it does not consider the polygonal diameter, just the number of edges, thus it allows arbitrarily small edges. In this sense, the following theorem can be written:

Theorem 9. *Let the set of hypothesis 3 holds true. Let $u \in H^s(\Omega)$, with $s > 1$, be the solution for the continuous problem in (3.10).And, let $u_h \in V_h$ be the solution for the discrete problem. There exists a constant*

$$\kappa(h) = \max_{K \in \tau_h} \left[\ln\left(1 + \frac{h_k}{l_{K,e}}\right) \right] \quad \text{(B.8)}$$

such that

$$\|u - u_h\|_{H^s(\Omega)} \leq \kappa(h) h^{s-1} \|u\|_{H^s(\Omega)}, \tag{B.9}$$

with $1 < s \leq k + 1$.

Theorem 9 guarantees the robustness of the method regarding to mesh quality. it is important to mention that before the work presented in da Veiga *et al.* (2017b), this result was empirically observed but no proof was shown. Also, the demonstration of theorem 9 can be found in da Veiga *et al.* (2017b) with a detailed discussion about how to deduce the term κ. Regarding to the classical result presented in theorem 8, a more complete discussion can be found in da Veiga *et al.* (2013a).

Appendix C

Risers and pressure armors

In this appendix a contextualization about risers and zeta pressure armor is presented. Tubes are very common structures used by offshore industry to transport oil. They can be composed by metallic and polymeric layers (flexible tubes) or exclusively by steel (rigid tubes). Accordingly to Mendonça (2016), the installation of the flexible tubes are inserted in an adverse context in which the most significant loads are the pipe own weight, the radial compression due to the shoes of the tensioner and the squeezing load applied to the pipe due to the pressure armor. Also, the author states that main advantages of flexible tubes are the robust behavior in extreme dynamical situations, the consistent isolation and the compatibility with the environment chemical properties (the operation environment is filled with corrosive gases).

Regarding to the configuration of pipes in an offshore oil exploitation system, if the pipe is suspended it is called riser and if the pipe is touching the ground, it is called flowline. Basically, risers are used to connect the platform to the flowline, while jumpers are pipes with smaller length that are used to connect different equipment. Figure C.1 shows a schematic about the mentioned configuration. An alternative to tubes in riser structures are the umbilical cables. Accordingly to Provasi (2013), umbilical cables are responsible for hydraulic and electrical controls, to pump fluids into the oil well and to transmit electrical energy and signals.

Risers can present different sets of configuration. A very common riser configuration is the free-hanging catenary where it is fixed on top, subjected to its own weight and suffers traction. Accordingly to Gay Neto (2012), catenary may not be adequate for deepwater when the movement of the platform is significantly big. In this situation the lazy-wave, in which buoyancy modules are distributed in the middle of the tube, is better. There are some other configurations like the steep wave where one of the extreme points is vertically connected to the seabed and bend stiffeners are used. In Figure C.2, it possible to see the free-hanging catenary, lazy wave, steep wave and other configurations of risers.

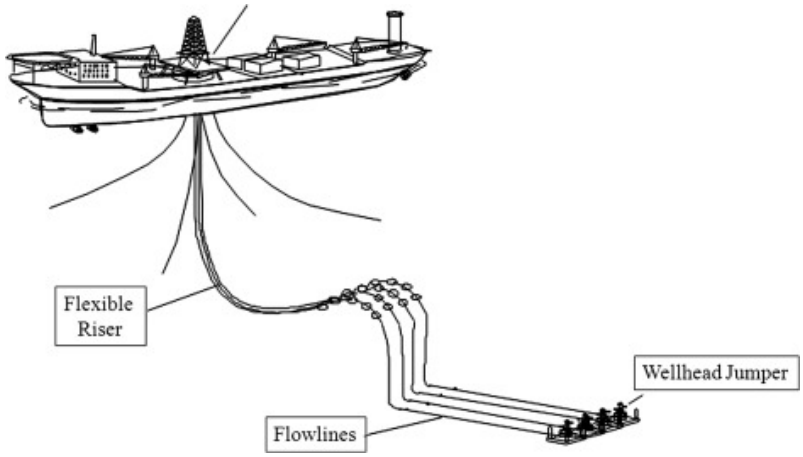


Figure C.1: *Schematic representation of a platform with riser, flowline and jumper configuration.*
Source: (Bai and Bai, 2019)

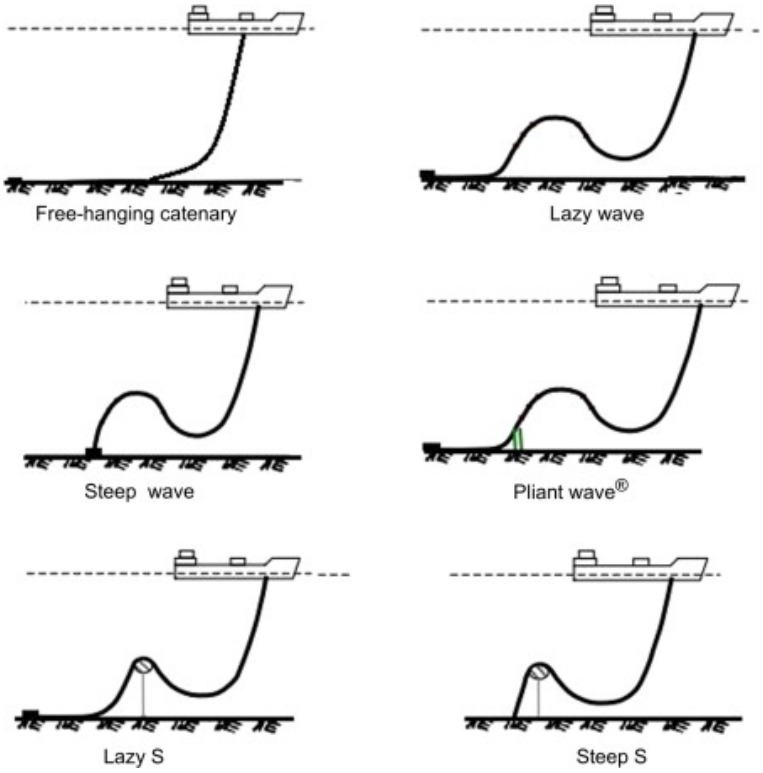


Figure C.2: *Examples of possible risers configurations.*
Source: (Bai and Bai, 2019)

It is important to know the different types of platform once its dynamics has direct influence on risers. Provasi (2013) and Gay Neto (2012) briefly explain the main types of platforms used in the offshore industry:

- **Fixed Platform**: they are built upon a metallic structure and fixed on the seabed with stakes.
- **Semi-submersible Platform**: its construction is based on columns attached to a buoyancy system and an anchor system is used to restrict the platform movement.

- **Compliant Platform**: it has a similar construction when compared to the Fixed Platforms but it has a better performance with respect to marine load due its flexibility.
- **Tension Leg Platform (TLP)**: they are floating platform with a tension mooring system that is responsible to keep the platform stable.
- **SPAR**: this kind of platform is connected to the seabed by an anchor line and its movement is related to the environmental dynamics. The construction of a SPAR makes the use of rigid tubes instead of the flexible ones.
- **Floating Production, Storage and Offload (FPSO)**: they were first conceived as an adaptation of oil ships but today they are built with this design. The FPSO has a large capacity of storage but they are very susceptible to the ship hydrodynamics. The turret configuration is adopted to overcome this problem.

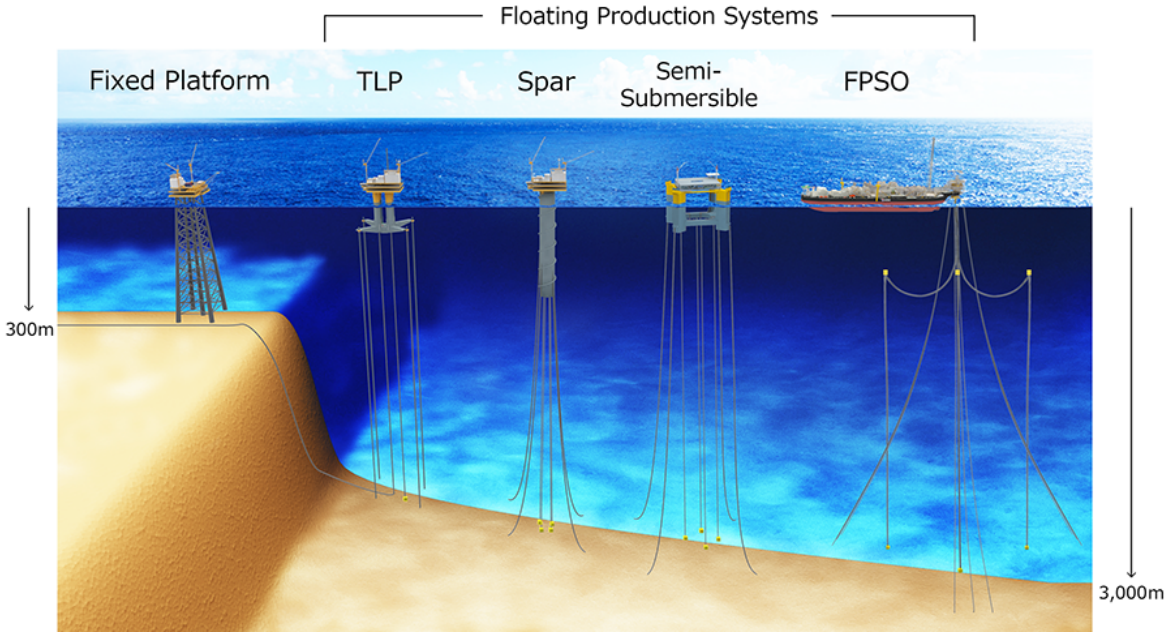


Figure C.3: *Illustration of platforms: Fixed Platform, Tension Leg Platform, SPAR, Semi-submersible Platform and Floating Production, Storage and Offload*

Source: <https://www.modec.com/>

This work focuses on flexible tubes used as risers. For this kind of application, the tubes are classified as unbonded once the layers are independent and can move freely relatively to another. This category of tubes are commonly used in deepwater exploitation but they suffer with the wear of the structure. Flexible tubes can also be classified as bonded where the layers have no relative movement and are mostly used in jumpers.

Accordingly to Mendonça (2016), the bonded flexible tubes are constructed with polymeric and metallic concentric layers to have low flexural stiffness and high axial stiffness. The author says that the polymeric layers have are responsible to seal the anchor line and the

metallic layers have a structural function. The main layers of a flexible tube are mentioned bellow and Figure C.4 shows the layers and respective profiles with more detail:

- Interlocked carcass (metallic),
- Internal pressure sheath (polymeric),
- Pressure armor (metallic),
- Anti-wear layer (polymeric),
- Tensile armor (metallic),
- Outer sheath (polymeric).

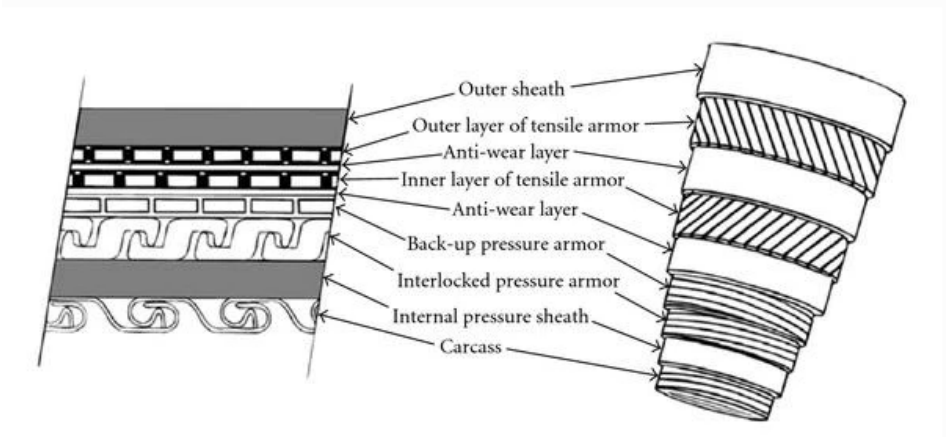


Figure C.4: *Bonded flexible tube structure*
Source: (Pipa *et al.*, 2010)

The chosen complex geometry to apply the Virtual Element Method is the pressure armor. This layer is built with wire in a helix geometry with the main purpose of resisting high pressure workload. Mendonça (2016) states that pressure armor are designed to resist the crushing load, the squeezing load and hydrostatic pressure. In this way the armor has a structural role in tube construction, elevating the tube resistance. Normally the pressure armor are made of carbon steel and the most common profiles are the Zeta-shape, T-shape and C-shape. Figure C.5 shows the zeta-shape that is the one chosen for this work.

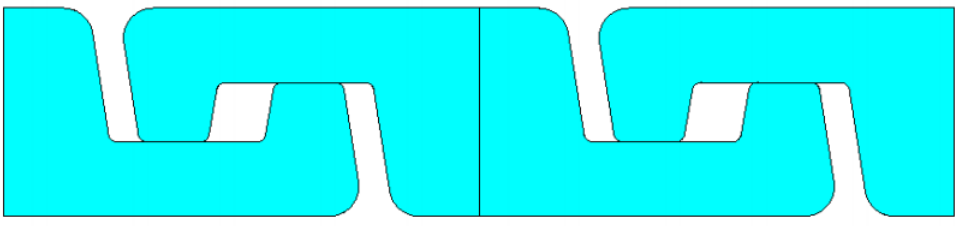


Figure C.5: *Zeta-shaped pressure armor with two steps*
Source: (Mendonça, 2016)

Bibliography

Ahmad *et al.***(2013)** B. Ahmad, A. Alsaedi, F. Brezzi, L. D. Marini and A. Russo. Equivalent projectors for virtual element methods. *Computers and Mathematics with Applications*, 66(3):376–391. ISSN 08981221. doi: 10.1016/j.camwa.2013.05.015. URL http://dx.doi.org/10.1016/j.camwa.2013.05.015. Cited on page 8

Alberty *et al.***(1999)** Jochen Alberty, Carsten Carstensen and Stefan A. Funken. ARemarks around 50 lines of Matlab: short finite element implementation. *Numerical Algorithms*, 20:117–137. ISSN 1572-9265. doi: https://doi.org/10.1023/A:1019155918070. Cited on page 56

Aldakheel *et al.***(2020)** Fadi Aldakheel, Blaž Hudobivnik, Edoardo Artioli, Lourenço Beirão da Veiga and Peter Wriggers. Curvilinear virtual elements for contact mechanics. *Computer Methods in Applied Mechanics and Engineering*, 372:113394. ISSN 00457825. doi: 10.1016/j.cma.2020.113394. URL https://doi.org/10.1016/j.cma.2020.113394. Cited on page 13

Artioli *et al.***(2017)** 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*, 60(3):355–377. ISSN 01787675. doi: 10.1007/s00466-017-1404-5. Cited on page 9, 65, 71, 73

Babuška and Melenk(1997) I. Babuška and J. M. Melenk. The partition of unity method. *International Journal for Numerical Methods in Engineering*, 40(4):727–758. ISSN 00295981. doi: 10.1002/(SICI)1097-0207(19970228)40:4<727::AID-NME86>3.0.CO;2-N. Cited on page 1

Bai and Bai(2019) Yong Bai and Qiang Bai, editors. *Front Matter*. Gulf Professional Publishing, Boston, second edition edição. ISBN 978-0-12-812622-6. doi: https://doi.org/10.1016/B978-0-12-812622-6.01001-0. URL https://www.sciencedirect.com/science/article/pii/B9780128126226010010. Cited on page 98

Bartle(1982) R.G. Bartle. *Elements of Real Analysis*. A Wiley Arabook. John Wiley & Sons Incorporated. ISBN 9780471063919. Cited on page 91

Botelho *et al.***(2015)** G.M.A. Botelho, D.M. Pellegrino and E.V. Teixeira. *Fundamentos de Análise Funcional*. SBM, 5th edição. ISBN 9788583370680. Cited on page 91

Brezis(2010) H. Brezis. *Functional Analysis, Sobolev Spaces and Partial Differential Equations*. Universitext. Springer New York. ISBN 9780387709130. Cited on page 91

Bucalem and Bathe(2011) M. L. Bucalem and K. J. Bathe. *The Mechanics of Solids and Structures - Hierarchical Modeling and the Finite Element Solution*. Springer-Verlag, Berlin Heidelberg. Cited on page 43, 44, 51

Cangiani *et al.***(2017a)** Aandrea Cangiani, Panagiotis Chatzipantelidis, Ganesh Diwan and Emmnuil H. Georgoulis. A virtual element method for quasilinear elliptic problems. *IMA Journal ofNumerical Analysis*, (October 2019):2450–2472. ISSN 23318422. doi: 10.1093/imanum/drz035. Cited on page 14

Cangiani *et al.***(2017b)** Andrea Cangiani, Gianmarco Manzini and Oliver J. Sutton. Conforming and nonconforming virtual element methods for elliptic problems. *IMA Journal of Numerical Analysis*, 37(3):1317–1354. ISSN 14643642. doi: 10.1093/imanum/drw036. Cited on page 8

Chen(2018) Long Chen. Programming of Finite Element Methods in Matlab. *arXiv*. Cited on page 71

da Veiga *et al.***(2013a)** 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*, 23(1):199–214. doi: 10.1142/S0218202512500492. Cited on page 4, 7, 8, 14, 22, 23, 31, 93, 94, 95

da Veiga *et al.***(2013b)** L. Beirão da Veiga, F. Brezzi and L. D. Marini. Virtual elements for linear elasticity problems. *SIAM Journal on Numerical Analysis*, 51(2):794–812. ISSN 00361429. doi: 10.1137/120874746. Cited on page 8, 9, 11, 65

da Veiga *et al.***(2014)** L. Beirão da Veiga, F. Brezzi, L. D. Marini and A. Russo. The hitchhiker's guide to the virtual element method. *Mathematical Models and Methods in Applied Sciences*, 2(8):1541–1573. doi: 10.1142/s021820251440003x. Cited on page 4, 7, 9, 31, 35, 67, 89, 94

da Veiga *et al.***(2015)** 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*, 295:327–346. ISSN 00457825. doi: 10.1016/j.cma.2015.07.013. Cited on page 9

da Veiga *et al.***(2016)** L. Beirão da Veiga, F. Brezzi, L. D. Marini and A. Russo. Virtual Element Method for general second-order elliptic problems on polygonal meshes. *Mathematical Models and Methods in Applied Sciences*, 26(4):729–750. ISSN 02182025. doi: 10.1142/S0218202516500160. Cited on page 8

da Veiga *et al.***(2017a)** L. Beirão da Veiga, C. Lovadina and G. Vacca. Virtual Elements for the Navier-Stokes problem on polygonal meshes. *SIAM Journal on Numerical Analysis*, 56(3):1210–1242. ISSN 23318422. Cited on page 14

da Veiga *et al.***(2017b)** L. Beirão da Veiga, Carlo Lovadina and Alessandro Russo. Stability analysis for the virtual element method. *Mathematical Models and Methods in Applied Sciences*, 27(13):2557–2594. ISSN 02182025. doi: 10.1142/S021820251750052X. Cited on page 4, 14, 30, 39, 85, 93, 94, 95

de Dios *et al.***(2016)** Blanca Ayuso de Dios, Konstantin Lipnikov and Gianmarco Manzini. The nonconforming virtual element method. *ESAIM: M2AN*, 50(3):879–904. doi: 10.1051/m2an/2015090. URL https://doi.org/10.1051/m2an/2015090. Cited on page 8, 11

Evans(2010) Lawrence C. Evans. *Partial differential equations*, volume 19. American Mathematical Society. ISBN 9780821849743. Cited on page 17, 19, 20, 91, 92

Furtado(2012) Marcelo Furtado. Notas de edp. https://www.mat.unb.br/~furtado/homepage/notas-edp2.pdf, 2012. Last access in 06/03/2021. Cited on page 17, 18, 20, 91, 92

Gain *et al.***(2014)** Arun L. Gain, Cameron Talischi and Glaucio H. Paulino. On the virtual element method for three-dimensional linear elasticity problems on arbitrary polyhedral meshes. *Computer Methods in Applied Mechanics and Engineering*, 282:132–160. ISSN 00457825. doi: 10.1016/j.cma.2014.05.005. Cited on page 9, 10, 11, 13, 61, 63, 66, 69, 90

Gay Neto(2012) Alfredo Gay Neto. *Estabilidade estrutural da configuração estática de risers em catenária*. Tese de Doutorado, Escola Politécnica, Universidade de São Paulo, Brazil. Cited on page 97, 98

Geuzaine and Remacle(2009) Christophe Geuzaine and Jean-François Remacle. Gmsh: A 3-d finite element mesh generator with built-in pre and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79:1309 – 1331. doi: 10.1002/nme.2579. Cited on page 2, 71

Isnard(2013) Carlos Isnard. *Introdução à Medida e Integração*. IMPA, 3rd edição. ISBN 9788524402708. Cited on page 17, 91

Lax(2002) P.D. Lax. *Functional Analysis*. Pure and Applied Mathematics: A Wiley Series of Texts, Monographs and Tracts. Wiley. ISBN 9780471556046. Cited on page 91

LeVeque(2007) Randall J. LeVeque. *Numerical Analysis*. Society for Industrial and Applied Mathematics (SIAM). ISBN 78-0-898716-29-0. Cited on page 56

Melenk and Babuška(1996) J. M. Melenk and I. Babuška. The partition of unity finite element method: Basic theory and applications. *Computer Methods in Applied Mechanics and Engineering*, 139(1-4):289–314. ISSN 00457825. doi: 10.1016/S0045-7825(96)01087-0. Cited on page 1

Mendonça(2016) Heloisa Guedes Mendonça. Análise paramétrica do efeito de cargas de esmagamento e estrangulamento sobre um tubo flexível na operação de lançamento, 2016. Cited on page 82, 97, 99, 100

Mengolini *et al.***(2019)** Michael Mengolini, Matías F. Benedetto and Alejandro M. Aragón. An engineering perspective to the virtual element method and its interplay with the standard finite element method. *Computer Methods in Applied Mechanics and Engineering*, 350:995 – 1023. ISSN 0045-7825. doi: https://doi.org/10.1016/j.cma.2019.02.043. Cited on page 4, 11, 12, 65, 72

Moherdaui and Neto(2019) Tiago Moherdaui and Alfredo Neto. Virtual Element Method vs Finite Element Method for Prandtl's Solution of St. Venant Torsion Problem. *Proceedings XL Ibero-Latin-American Congress on Computational Methods in Engineering (CILAMCE 2019)*, 1. ISSN 2675-6269. Cited on page 51

Ortiz-Bernardin *et al.***(2019)** Alejandro Ortiz-Bernardin, Catalina Alvarez, Nancy Hitschfeld-Kahler, Alessandro Russo, Rodrigo Silva-Valenzuela and Edgardo Olate-Sanzana. Veamy: an extensible object-oriented c++ library for the virtual element method. *Numerical Algorithms*. doi: 10.1007/s11075-018-00651-0. Cited on page 9, 11, 39, 61, 63, 90

Park *et al.***(2019)** Kyoungsoo Park, Heng Chi and Glaucio H. Paulino. On nonconvex meshes for elastodynamics using virtual element methods with explicit time integration. *Computer Methods in Applied Mechanics and Engineering*, 356:669–684. ISSN 00457825. doi: 10.1016/j.cma.2019.06.031. Cited on page 4

Paulino and Gain(2015) Glaucio H. Paulino and Arun L. Gain. Bridging art and engineering using Escher-based virtual elements. *Structural and Multidisciplinary Optimization*, 51(4):867–883. ISSN 16151488. doi: 10.1007/s00158-014-1179-7. Cited on page 13, 14

Pipa *et al.***(2010)** Daniel Pipa, Sérgio Morikawa, Gustavo Pires, Claudio Camerini and JoãoMárcio Santos. Flexible riser monitoring using hybrid magnetic/optical strain gage techniques through rls adaptive filtering. *EURASIP Journal on Advances in Signal Processing*, 2010(1):176203. ISSN 1687-6180. doi: 10.1155/2010/176203. URL https://doi.org/10.1155/2010/176203. Cited on page 100

Provasi(2013) Rodrigo Provasi. *Contribuição ao projeto de cabos umbilicais e tubos flexíveis : ferramentas de CAD e modelo de macro elementos*. Tese de Doutorado, Escola Politécnica, Universidade de São Paulo, Brazil. Cited on page 97, 98

Rudin(1987) W. Rudin. *Real and Complex Analysis*. Higher Mathematics Series. McGraw-Hill Education. ISBN 9780070542341. Cited on page 91

Savarè and Chanon(2016) Stefano Savarè and Ondine Chanon. Virtual element method. http://www1.mate.polimi.it/~forma/Didattica/ProgettiPacs/SavareChanon15-16/main.pdf, 2016. Last access in 06/03/2021. Cited on page 22, 30

Schillinger and Ruess(2015) Dominik Schillinger and Martin Ruess. The Finite Cell Method: A Review in the Context of Higher-Order Structural Analysis of CAD and Image-Based Geometric Models. *Archives of Computational Methods in Engineering*, 22(3):391 – 455. ISSN 18861784. doi: 10.1007/s11831-014-9115-y. Cited on page 2, 3

Strichartz(2003) Robert S Strichartz. *A Guide to Distribution Theory and Fourier Transforms*. WORLD SCIENTIFIC. doi: 10.1142/5314. URL https://www.worldscientific.com/doi/abs/10.1142/5314. Cited on page 18

Sutton(2016) Oliver Sutton. The virtual element method in 50 lines of matlab. *Numerical Algorithms*, 75:1141–1159. doi: 10.1007/s11075-016-0235-3. Cited on page xiii, 9, 10, 11, 35, 38, 39, 71

Talischi *et al.***(2012)** Cameron Talischi, Glaucio Paulino, Anderson Pereira and Ivan Menezes. Polymesher: a general-purpose mesh generator for polygonal elements written in matlab. *Structural and Multidisciplinary Optimization*, 45:309–328. doi: 10.1007/s00158-011-0706-z. Cited on page 9, 39

Timoshenko and Goodier(1951) Stephen Timoshenko and James Norman Goodier. *Theory of Elasticity*. McGraw-Hill, 2 edição. Cited on page 43, 51, 52, 56, 58

Wriggers *et al.***(2016)** P. Wriggers, W. T. Rust and B. D. Reddy. A virtual element method for contact. *Computational Mechanics*, 58(6):1039–1050. ISSN 01787675. doi: 10.1007/s00466-016-1331-x. Cited on page 13, 31, 85

Wriggers *et al.***(2020)** P. Wriggers, B. Hudobivnik and F. Aldakheel. A virtual element formulation for general element shapes. *Computational Mechanics*, 66:963–977. doi: 10.1007/s00466-020-01891-5. Cited on page 15

Young and Budynas(2002) Warren C. Young and Richard G. Budynas. *Roark's Formulas for Stress and Strain*. McGraw-Hill, 7 edição. ISBN 0-07-072542-X. Cited on page 43, 80

Zeng and Liu(2018) W. Zeng and G. R. Liu. Smoothed Finite Element Methods (S-FEM): An Overview and Recent Developments. *Archives of Computational Methods in Engineering*, 25(2):397–435. ISSN 18861784. doi: 10.1007/s11831-016-9202-3. Cited on page 2

Zhang *et al.***(2019)** Bei Zhang, Jikun Zhao, Yongqin Yang and Shaochun Chen. The nonconforming virtual element method for elasticity problems. *Journal of Computational Physics*, 378:394–410. ISSN 10902716. doi: 10.1016/j.jcp.2018.11.004. URL https://doi.org/10.1016/j.jcp.2018.11.004. Cited on page 10

Zhang *et al.***(2020)** J. F. Zhang, R. P. Niu, Y. F. Zhang, C. Q. Wang, M. Li and G. R. Liu. Development of sfem-pre: A novel preprocessor for model creation for the smoothed finite element method. *International Journal of Computational Methods*, 17(02):1845002. doi: 10.1142/S0219876218450020. URL https://doi.org/10.1142/S0219876218450020. Cited on page 2