Moving Mesh Virtual Element Methods

Harry Wells

A thesis submitted to the University of Nottingham for the degree of Doctor of Philosophy

June 2023

Preface

The past few years have unfolded differently than I anticipated. When I started my PhD in October of 2019, no one could have predicted the onset of the COVID-19 pandemic and the profound effect it would have on people's lives. This unforeseen event steered me towards an unconventional PhD journey, filled with unique personal and academic challenges. Nonetheless, four years on, I write this preface at the successful close of my doctoral studies. The emotional and professional support required to complete a PhD often remains under-acknowledged. Hence, I've crafted this preface to extensively recognize everyone who contributed, in both large and small ways, to my current position. I intended to balance succinctness with gratitude, resulting in numerous lists spanning all forms of support that culminated in the success of this PhD.

Foremost, I'm indebted to my parents and my brother, for their steadfast emotional and financial support and encouragement throughout this journey.

My gratitude goes to my supervisors, Andrea Cangiani and Matthew Hubbard. Their unwavering patience and guidance were invaluable. I also want to acknowledge Andreas Dedner for his indispensable contributions to the project.

I've been fortunate to work within a supportive research group and office. My sincere thanks to Thomas Radley, Richard Widdowson, Adam Blakey, Anne Boschman, Zak Crowson, Oliver Sutton, Kris Van der Zee, Tristan Lawrie, James Branch, and Fabian Cejka.

During the COVID pandemic years, I was fortunate to have the company of Robert Barnett, Alistair Delboyer, and Fabian Cejka. Their presence helped me cope with the loneliness and isolation of the lockdowns.

For those late nights in the office that tested my sanity, my appreciation extends to Akash Sharma, Michael Yiasemides, and Yiyun Fan for being my anchors.

I'd like to thank Thomas Radley, Thomas Hall, Fabian Cejka, Niren Bhoja,

and Ahmed Sherif. They always found ways to put a smile on my face and brought out the inner child in me, even during the toughest times of this PhD.

I owe a deep debt of gratitude to Thomas Radley, Nikita Kathuria, and Ruth Penny. They stood by me as my emotional pillars during some of the most challenging personal moments.

To my housemates over the past two years: Thomas Radley, Thomas Hall, Suleiman Mousa, Ruth Penny, Isabell Ogland, Jessica Hughes, Katherine Hamilton, Katie Bravey, Christinia Stuart, Ahmed Sherif, Nikita Kathuria, and Demi Karamansoylu - thank you for the countless memories and experiences.

I will cherish the endless socials, trips and sporting events with Valentin Breaz, Celene Lee, Akash Sharma, Riccardo Corrardin, Federico Girotti, Thomas Laird, Thomas Radley, Thomas Hall, Sophie Wartnaby, Aaron Walker, Isabelle Sims, Robert Allen, Adam Wilkinson, Piers Hinds, Niren Bhoja, Josh Pimm, James Branch, Charles Valdez, Alexander Lewis, Yiyun Fan, Yajie Guo, Wasuir KhudaBukhsh, and Yordan Raykov.

Lastly, to all those not named here but who have been integral in this chapter of my life, my deepest gratitude.

I will conclude this preface on a personal note. Despite the difficulties and challenges of the past few years, I am very proud of what I have been able to achieve over the course of this PhD. The personal and professional growth from the naïve 22-year-old to where I am today has been astonishing, and I look forward to what the future has in store for me with great optimism and excitement. I hope to look back in the future on this era of my life with a great sense of pride in my time at Nottingham.

Harry Wells, October 13, 2023

Abstract

This thesis explores the development and analysis of moving mesh Virtual Element Methods for partial differential equations on time-dependent domains. This thesis presents the first moving mesh method to purely use the Virtual Element Method, an isoparametric Virtual Element Method for approximating partial differential equations on curved domains and a high-order Arbitrary Lagrangian-Eulerian Virtual Element Method for problems on time-dependent domains with moving boundaries. Each contribution successfully demonstrates the applicability and accuracy of Virtual Element Methods in existing moving mesh algorithms, achieving similar orders of accuracy compared to classical Finite Element Method approaches. The results suggest that the flexibility of moving mesh methods can be greatly improved by incorporating more general mesh structures, including polygons and curved-edged polygons, proving the Virtual Element Method offers an effective extension to classical approaches. This work provides a foundation for future research in Virtual Element Methods for more complex problems on time-dependent domains and developing the analysis to support proposed moving mesh methods.

Contents

Contents **5**

1 Introduction **8**

1.1 Motivation 8

1.2 Thesis Overview 10

1.3 Notation & Definitions 11

2 Preliminaries **14**

2.1 The Virtual Element Method 14

2.2 Moving Mesh Methods 24

2.3 Literature Review 31

I A Velocity-based Moving Mesh Virtual Element Method **39**

3 A Virtual Element discretisation of a Moving Mesh Algorithm **42**

3.1 The Porous Medium Equation 43

3.2 Algorithm Review 44

3.3 A Moving Virtual Element Space 51

3.4 A Virtual Element Method for the Velocity Field 52

3.5 A Virtual Element Method for the Solution 54

3.6 A Conservation Law 56

3.7 Implementation Details 57

4 Numerical Investigations **59**

4.1 Convergence for the Porous Medium Equation 59
4.2 A Contact Algorithm 64
4.3 A Fourth-order Diffusion Problem 75

II Isoparametric Virtual Element Methods **81**

5 Method I **85**
5.1 Problem Statement 86
5.2 The Virtual Element Mapping 88
5.3 Extending the PDE to the Virtual Domain 89
5.4 Approximating the Jacobian 91
5.5 Method I Formulation 97
5.6 Well-posedness 98
5.7 Consistency Errors 102
5.8 A H^1 Estimate 108

6 Method II **111**
6.1 Projected Elements 111
6.2 A Virtual Element Space on Curved Edge Polygons 119
6.3 Computable Projection Operators 121
6.4 Method II Formulation 128
6.5 Consistency Error 129
6.6 Well-posedness 132
6.7 A H^1 Estimate 133

7 Numerical Investigations **141**
7.1 Example Mappings 141
7.2 Convergence Results in Two Dimensions 142

III A Virtual Element ALE Method **158**

8 A Virtual Element Arbitrary Lagrangian-Eulerian Scheme **161**

8.1 Problem Definition 161

8.2 A Moving Virtual Element Space 163

8.3 A Semi-discrete Scheme 165

8.4 A fully Discrete Scheme 166

9 Numerical Investigation **168**

9.1 Time-independent ALE Maps 169

9.2 A Pure Diffusion Problem 170

9.3 A General Advection-diffusion Problem 171

9.4 The Velocity-based Moving Mesh Method 174

10 Conclusion **183**

10.1 Introduction 183

10.2 Summary of Part I 184

10.3 Summary of Part II 184

10.4 Summary of Part III 185

10.5 Discussion, Interpretation, and Contribution to Knowledge 185

10.6 Recommendations for Future Research 185

10.7 Conclusion 186

Bibliography **188**

Chapter 1

Introduction

This thesis is devoted to the development and analysis of Virtual Element Methods for moving mesh methods for partial differential equations on time-dependent domains.

In Section 1.1, the motivation for developing polygonal moving mesh methods is discussed. The structure and major contributions of this thesis are highlighted in Section 1.2. Standard notation and conventions adopted in this thesis are discussed in Section 1.3.

1.1 Motivation

Finite Element Methods (FEMs) have been successfully employed for decades in the simulation of partial differential equations (PDEs). The theory and implementation of FEMs for a variety of problems are well-documented [41, 29, 74]. Among the benefits of FEMs is the adaptivity of the mesh (h-adaptivity) and the polynomial degree (p-adaptivity), which allow for optimal mesh refinement and increased accuracy of the FEM. In particular, these forms of adaptivity are effectively employed in discontinuous Galerkin FEMs which allow for more general polygonal/polyhedral domain discretisations [37]. Another form of adaptivity, known as mesh movement or r-adaptivity, is less well known.

The benefit of a moving mesh method is that the accuracy of a numerical

simulation can be improved by relocating mesh vertices, without incurring the computational expense of *hp*-refinement strategies. Moving mesh methods also enable better resolution of time-dependent features of a given PDE, such as moving boundaries.

A common setback of moving mesh methods in certain applications is the need to remesh during simulations to prevent the degradation of the moving mesh [89, 14].

Using a polygonal discretisation for a moving mesh method offers two benefits: remeshing can be performed locally with greater efficiency and minimal changes to the numerical method, and complex moving features, such as moving boundaries, can be simulated with fewer degrees of freedom while achieving the same level of accuracy [37, 38]. Additionally, the framework of a moving polygonal mesh provides the foundation for developing even more general moving mesh methods, including those with curved faces.

In developing a moving polygonal mesh method, the following conditions were desired: (a) the discretisation had to reduce to classical moving mesh FEMs when triangular meshes are employed, (b) the discrete space used had to be, at least, continuous across the entire moving mesh, and (c) the analysis of such a discretisation needed to be an extension of typical FEM analysis.

The Virtual Element Method (VEM) [18, 2] satisfies all three conditions. It can be interpreted as an extension of classical FEMs to polygonal/polyhedral meshes by introducing "virtual" shape functions that are implicitly understood through a finite set of degrees of freedom. These degrees of freedom are used to compute accurate and stable approximations of the weak formulation of a given PDE.

Other well-developed polygonal discretisation techniques, such as the discontinuous Galerkin method [37], the Hybrid High-Order (HHO) method [49], and the Mimetic Finite Difference (MFD) method [77], exist. However, at the time of writing, only the discontinuous Galerkin method has been successfully

applied to moving mesh problems, as shown in recent examples [78, 80, 59, 58]. Notably, the VEM has only recently (in 2019) been applied to moving mesh methods when coupled with a discontinuous Galerkin method [78].

The motivation of this thesis is to propose moving mesh methods using exclusively a VEM. The first challenge is to test whether the VEM can be successfully applied to an existing moving mesh algorithm. After this, a foundation to analyse a moving mesh VEM is required in order to develop methods for more complex problems. The objectives of this project can be summarised as follows:

> *"To propose, analyse, and implement robust and effective moving mesh Virtual Element Methods."*

1.2 Thesis Overview

The thesis is structured into three distinct parts, each of which can be considered as a self-contained piece of work. Results from previous parts will be referenced appropriately and utilised in subsequent sections. The structure of each part is broken down as follows: an introduction that outlines the key contributions and provides an overview of the layout, chapters that present the main contributions, and a discussion that critically reviews the work covered in that specific part.

Part I

In Part I, a velocity-based moving mesh VEM is proposed. A moving mesh algorithm is extended to polygonal discretisations by proposing a linear virtual element approximation of the integral equations. The new method is numerically tested against non-linear diffusion problems and a mesh contact algorithm is presented that exploits the generality of the VEM to simulate contact be-

tween moving boundaries and fixed obstacles. The work presented in Part I is currently under review [99].

Part II

In Part II an isoparametric VEM is proposed for second-order elliptic partial differential equations on transformed domains. Two discretisation approaches are formulated and analysed with H^1 *a priori* error estimates being presented in both cases. Numerical results are presented that validate the analysis. This research was conducted in collaboration with Andreas Dedner (University of Warwick). At the time of writing, a publication is being drafted which presents the error estimates and a sample of the numerical results in 2D.

Part III

In Part III the isoparametric VEM is employed to formulate a high-order conservative Arbitrary Lagrangian-Eulerian scheme for solving advection-diffusion partial differential equations on time-dependent domains. Numerical experiments are presented which obtain higher-order convergence rates.

1.3 Notation & Definitions

For a bounded set $\omega \subset \mathbb{R}^2$, the L^p norm for a $p \in [1, \infty]$ is defined for a function f by

$$\|f\|_{L^p(\omega)} = \left(\int_\omega |f|^p \ d\mathbf{x} \right)^{\frac{1}{p}}, \qquad p \in \mathbb{N},$$

$$\|f\|_{L^\infty(\omega)} = \sup_{\mathbf{x} \in \omega} |f|, \qquad p = \infty.$$

The well known Lebesgue space is defined by

$$L^p(\omega) = \left\{ f : \omega \to \mathbb{R} \ \middle| \ \|f\|_{L^p(\omega)} < \infty \right\}.$$

For a positive integer k and an indexing set α the weak derivative of order $|\alpha|$ of f is denoted by $D^\alpha f$ [1]. The Sobolev norm is defined by

$$\|f\|_{W_k^p(\omega)} = \left(\sum_{|\alpha| \leq k} \|D^\alpha f\|_{L^p(\omega)}^p \right)^{\frac{1}{p}}, \qquad p \in \mathbb{N},$$

$$\|f\|_{W_k^\infty(\omega)} = \max_{|\alpha| \leq k} \|D^\alpha f\|_{L^\infty(\omega)}, \qquad p = \infty.$$

The Sobolev space is denoted by $W_k^p(\omega)$ and defined by

$$W_k^p(\omega) = \left\{ f : \omega \to \mathbb{R} \;\middle|\; \|f\|_{W_k^p(\omega)} < \infty \right\}.$$

To simplify the presentation, the norm and semi-norms are denoted by $\|\cdot\|_{k,p,\omega}$ and $|\cdot|_{k,p,\omega}$ respectively. In the case of $p = 2$, the corresponding Hilbert space is denoted by $H^k(\omega)$ with the norm and semi-norm denoted by $\|\cdot\|_{k,\omega}$ and $|\cdot|_{k,\omega}$ respectively.

The well known inequalities of Hölderand Minkowski will be used without proof [1]: for $p, q \in [1, \infty]$ with $1/p + 1/q = 1$

$$\|fg\|_{0,1,\omega} \leq \|f\|_{0,p,\omega} \|g\|_{0,q,\omega},$$

$$\|f + g\|_{0,p,\omega} \leq \|f\|_{0,p,\omega} + \|g\|_{0,p,\omega}.$$

The Cauchy-Schwarz inequality will also be used without proof [1]. Additional inequalities will be used within this thesis with appropriate citation to detailed proofs.

The notation $\lesssim$ will be adopted in this thesis for ease of reading. The statement $X \lesssim Y$ implies there exists a constant $C > 0$ such that $X \leq CY$. This constant will be referred to as a "hidden constant" and the dependencies of this constant will be stated in results that use this notation. The use of $\lesssim$ between two results does not imply that hidden constant is the same.

For functions, sets and variables f, ω, x, discrete counterparts will be denoted using f_h, ω_h and x_h. Time-dependent terms will include an additional subscript term of $f_{h,t}$, $\omega_{h,t}$ and $x_{h,t}$. If a discrete time level t^n is considered then the t subscript will be replaced with an n. Solutions to PDEs will be denoted using

ρ throughout the thesis. This is done to avoid confusion with velocity fields that are denoted by $\mathbf{u}$.

Chapter 2

Preliminaries

In this chapter, the prerequisite material for the Virtual Element Method and moving mesh methods is presented. For the Virtual Element Method, the focus is on the two-dimensional C^0-conforming Virtual Element Method originally proposed in [18] and enhanced in [2]. The concept of a moving domain and the Arbitrary Lagrangian-Eulerian method are introduced, along with a general outline of moving mesh algorithms.

In Section 2.1, the Virtual Element Method is introduced, including a review of the key components such as the construction of the discrete spaces, polynomial projection operators, and discrete bilinear forms. The framework of a moving mesh method and the Arbitrary Lagrangian-Eulerian method are presented in Section 2.2. A brief literature review, highlighting significant contributions towards moving polygonal mesh methods, is provided at the conclusion of this chapter in Section 2.3.

2.1 The Virtual Element Method

In this section the fundamentals of the Virtual Element Method (VEM) are discussed. Primarily, this thesis will focus on two-dimensional problems using conforming methods hence, only the details of this discretisation are presented.

The VEM was first presented in [18] as a generalisation of the Finite Ele-

ment Method for a two-dimensional Poisson problem to unstructured polygonal meshes. Subsequently, the VEM was applied to linear elasticity [43] and plate bending problems [33] but has since been adapted to tackle more general elliptic [20], parabolic [96], hyperbolic [95] and quasi-linear [36] PDEs. Additionally, VEMs have been presented in a variety of disciplines including elasticity, fracture modelling and computational fluid dynamics [5, 24, 102, 61].

The novelty of the VEM method is that the basis/shape functions are only implicitly understood through a finite set of degrees of freedom (DoFs). Approximations to a weak formulation of a given PDE can then computed, locally on a given polygon/polyhedron, using only the DoFs via computable polynomial projections of the virtual basis functions along with suitable stabilisation terms.

Within the literature, there exist several works on the implementation details of the VEM in 2D and 3D [93, 88, 44, 39, 36, 48]. As such these details are not discussed in this chapter.

Polygonal Mesh Structures

The main advantage of the VEM is the flexibility in mesh structures that are permitted compared to classical Finite Element Methods. The most common mesh structure considered is a polygonal/polyhedral mesh of elements that are simple, connected, non-overlapping polygons/polyhedra that form a partition of a given domain $\Omega \subset \mathbb{R}^d$ $(d = 2, 3)$.

For $\Omega \subset \mathbb{R}^2$, the mesh is denoted by $\mathcal{T}_h$ containing polygonal elements $E \in \mathcal{T}_h$. The properties of a given $E \in \mathcal{T}_h$ are defined as the diameter h_E, barycentric coordinate $\mathbf{x}_c^E = (x_c^E, y_c^E)$ and area $|E|$, boundary edges of E are denoted by $e \subset \partial E$. The global mesh size is defined by

$$h := \max_{E \in \mathcal{T}_h} h_E.$$

The shape regularity of a given element is defined as a constant ϱ_E such that

$$\varrho_E \leq \frac{\gamma_E}{h_E},$$

where γ_E is the maximum radius of a ball contained in E. The mesh is said to be shape regular if the mesh regularity parameter

$$\varrho = \max_{E \in \mathcal{T}_h} \varrho_E, \tag{2.1}$$

is contained in $(0, 1)$. To ensure that optimal approximation results can be obtained, the following mesh regularity assumption is required [18].

Assumption 1 (Mesh Regularity)**.**
Every $E \in \mathcal{T}_h$ is a star-shaped domain or a finite union of star shaped domains with respect to a ball of radius greater than γh_E for some uniform $\gamma > 0$. Additionally, for all edges $e \in \partial E$, the length of e is greater than δh_E for some uniform $\delta > 0$.

Remark 1. *It has been observed in the literature that the VEM can remain robust even when Assumption 1 does not hold [30]. This is observed in the numerical experiments of Part I of this thesis where the VEM remains robust even when arbitrarily small mesh edges are produced.*

Two commonly used polygonal meshes are the Voronoi Tessellation [91, 10, 51] and agglomerated triangulations of Ω: both are demonstrated in Figure 2.1. These mesh structures can be generated by using PolyMesher [94], Gmsh [63] or METIS [69], to name a few of the publicly available packages.

Polynomial Projection Operators

In order to attain sufficient accuracy in the VEM, polynomial projection operators are employed to approximate integral equations. For a given bounded set $\omega \subset \mathbb{R}^2$ The space of polynomials of degree $k \geq 0$ is denoted by $\mathbb{P}_k(\omega)$. The most common polynomial projection operators in the VEM literature are denoted by Π^∇, Π^0 and Π^1 [18, 2, 20] and are defined respectively as follows.

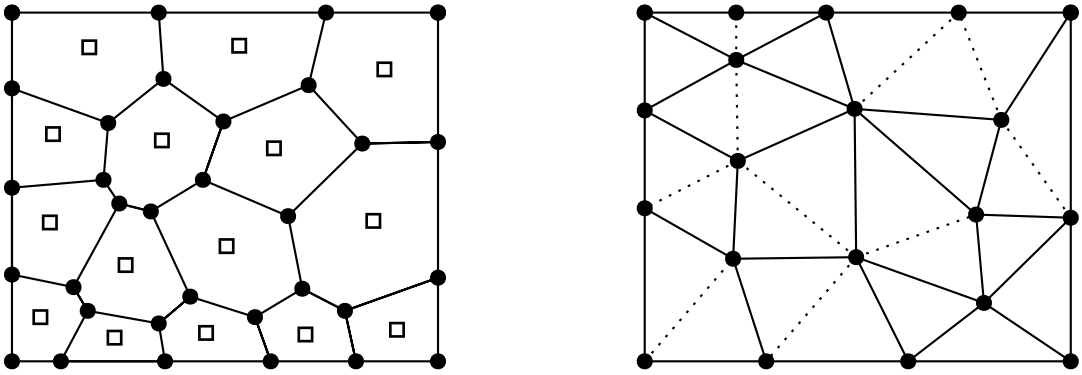

Figure 2.1: Two different methods to discretise a rectangular domain. The CVT mesh (left) shows the the Voronoi seeds used to generate the mesh by squares. The agglomerated mesh (right) has dashed lines to represent the removed edges from the triangulation.

Definition 2.1.1 (The Π^{∇} Operator)**.**

The operator $\Pi_k^{\nabla} : H^1(\omega) \to \mathbb{P}_k(\omega)$ *is defined for all* $v \in H^1(\omega)$ *by*

$$\int_{\omega} \nabla \Pi_k^{\nabla} v \cdot \nabla p \; d\mathbf{x} = \int_{\omega} \nabla v \cdot \nabla p \; d\mathbf{x} \qquad \forall p \in \mathbb{P}_k(\omega)$$
$$\int_{\omega} v - \Pi_k^{\nabla} v \; d\mathbf{x} = 0.$$

Definition 2.1.2 (The Π^0 Operator)**.**

The operator $\Pi_k^0 : L^2(\omega) \to \mathbb{P}_k(\omega)$ *is defined for all* $v \in L^2(\omega)$ *by*

$$\int_{\omega} \Pi_k^0 v \; p \; d\mathbf{x} = \int_{\omega} v \; p \; d\mathbf{x} \qquad \forall p \in \mathbb{P}_k(\omega).$$

Definition 2.1.3 (The Π^1 Operator)**.**

The operator $\Pi_k^1 : H^1(\omega) \to \mathbb{P}_k(\omega)$ *is defined for all* $v \in H^1(\omega)$ *by* $\Pi_k^1 v := \Pi_k^0 \nabla v$ *or equivalently*

$$\int_{\omega} \Pi_k^1 v \cdot \mathbf{p} \; d\mathbf{x} = \int_{\omega} \nabla v \cdot \mathbf{p} \; d\mathbf{x} \qquad \forall \mathbf{p} \in [\mathbb{P}_k(\omega)]^2 .$$

For each of these projection operators there exists well established stability and accuracy results under the condition that Assumption 1 is satisfied. This is summarised in the following theorem.

Theorem 2.1.1 (Approximation of Π^{∇}, Π^0)**.**

Let $\omega \subset \mathbb{R}^2$ *with diameter* h_ω *satisfy the conditions of Assumption 1. Let a real number* $p \in [1, \infty]$ *and* $k \geq 0$ *be an integer,* $s \in \{1, ..., k+1\}$ *and* $m \in \{0, ..., s\}$.

Then for all $v \in W_s^p(\omega)$ *it holds that*

$$\left|v - \Pi_k^\nabla v\right|_{m,p,\omega} \lesssim h_\omega^{s-m} \left|v\right|_{s,p,\omega}.$$

Additionally, for $s \in \{0, ..., k+1\}$ *and* $m \in \{0, ..., s\}$ *it holds that*

$$\left|v - \Pi_k^0 v\right|_{m,p,\omega} \lesssim h_\omega^{s-m} \left|v\right|_{s,p,\omega},$$

where the hidden constants depend only on the shape regularity parameter (2.1) ϱ*,* k*,* p *and* s*.*

Proof. A statement and proof of the estimates are given in Theorem 1.45 and Theorem 1.48 of [49]. □

Local Virtual Element Spaces

The classical VEM space is built on a given polygon $E\mathcal{T}_h$ by considering a local boundary value problem. Originally this was done by considering a Poisson problem subject to piecewise polynomial boundary conditions [18]. The boundary space is defined as

$$\mathbb{B}_k(\partial E) = \{v_h \in C^0(\partial E) \ : \ v_h|_e \in \mathbb{P}_k(e) \quad \forall e \subset \partial E\},$$

from which the original local VEM space of degree k can then be defined as

$$W_k(E) = \{v_h \in H^1(E) \ : \ v_h|_{\partial E} \in \mathbb{B}_k(\partial E), \ \Delta v_h|_E \in \mathbb{P}_{k-2}(E)\}. \tag{2.2}$$

Degrees of Freedom

The construction of a VEM relies on choosing a set of degrees of freedom (DoFs). These DoFs serve two purposes in the method: they provide the necessary information to construct computable polynomial approximation operators, and they uniquely identify a virtual element function on any given polygon, which would otherwise be an unknown solution to a local boundary value problem.

To define the DoFs of the classical VEM, the space of scaled monomials of degree k on a given element $E \in \mathcal{T}_h$ is defined as

$$\mathcal{M}_k^*(E) = \left\{ \left(\frac{x - x_c}{h_E}\right)^\alpha \left(\frac{y - y_c}{h_E}\right)^\beta \ : \ \alpha, \beta \in \mathbb{Z}\geq 0, \ \alpha + \beta = k \right\},$$

where the union of scaled monomials forms a basis for $\mathbb{P}_k(E)$:

$$\mathcal{M}_k(E) = \bigcup_{l \leq k} \mathcal{M}_l^*(E).$$

These polynomials are called "scaled monomials" because each polynomial scales as one when defined on any given polygon:

$$\|m_\alpha\|_{0,\infty,E} \sim 1 \qquad \forall m_\alpha \in \mathcal{M}_k(E).$$

The use of this space is crucial in defining the degrees of freedom, which will be discussed in further detail in the subsequent sections. In $\mathbb{R}^2$, the classical VEM DoFs are presented in the following definition.

Definition 2.1.4 (Degrees of Freedom for the Virtual Element Method)**.** *Let $E \in \mathcal{T}_h$ and $W_k(E)$ be the local VEM space defined in Equation* (2.2)*. The degrees of freedom of a given function $v_h \in W_k(E)$ are defined as follows:*

- *The point values of v at each vertex of E.*
- *The point values of v at the Gauss-Lobatto quadrature points of order $k-1$ on each edge $e \subset \partial E$.*
- *The internal moments of $\frac{1}{|E|}\int_E v_h q \, dx$ for all $q \in \mathcal{M}_{k-2}(E)$.*

An illustrative diagram of these DoFs is presented in Figure 2.2. A proof that these constitute a unisolvent set of DoFs is provided in [18]. The H^1 projection Π_k^∇ is computable using only these degrees of freedom, as is the L^2 projection Π_{k-2}^0.

Enhanced Virtual Element Spaces

In order to develop the VEM for problems beyond Poisson's Equation, the local VEM space had to be modified such that the full L^2 projection was computable.

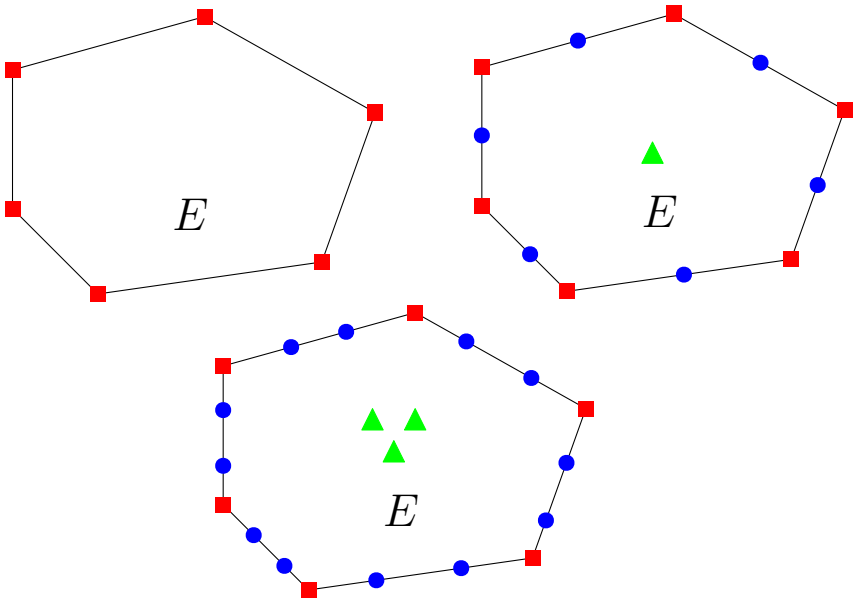

Figure 2.2: Degrees of freedom for a polygon E in the classical VEM for $k = 1$ (left), $k = 2$ (center), and $k = 3$ (right). Vertex values, edge quadrature point values, and internal moments are denoted by squares, circles, and triangles, respectively.

A solution was presented in [2] and is commonly used as the default VEM space in most recent discretisations. This enhanced space is referred to as the classical VEM space in this thesis.

The approach to enhance the VEM space is to define an auxiliary projection operator $\Pi_k^* : W_k(E) \to \mathbb{P}_k(E)$ to supplement the remaining internal moments of

$$\frac{1}{|E|}\int_E v_h q \; dx \qquad q \in \mathcal{M}_k^*(E) \cup \mathcal{M}_{k-1}^*(E).$$

The original local VEM space is enlarged via,

$$\tilde{W}_k(E) = \{v_h \in H^1(E) \;:\; v_h|_{\partial E} \in \mathbb{B}_k(\partial E),\; \Delta v_h \in \mathbb{P}_k(E)\},$$

and then restricted to define the classical VEM space,

$$V_k(E) = \left\{ v_h \in \tilde{W}_k(E) \;:\; \int_E (v_h - \Pi_k^* v_h) q \; d\mathbf{x} = 0 \quad \forall q \in \mathbb{P}_k(E) \backslash \mathbb{P}_{k-2}(E) \right\}. \tag{2.3}$$

The degrees of freedom given in Definition 2.1.4 also form a unisolvent set of DoFs for the classical VEM space defined in Equation (2.3) [2]. This auxiliary projection is taken as the Π_k^∇ projection in the classical VEM [2, 44].

The global VEM space of degree k is then defined as

$$V_h = \left\{ v_h \in H^1(\Omega) \;\; v_h|_E \in V_k(E) \; \forall E \in \mathcal{T}_h \right\}. \tag{2.4}$$

Using the VEM to solve PDEs with homogeneous Dirichlet boundary conditions requires a restricted global VEM space with zero trace. This is denoted as $V_{h,0} = V_h \cap H_0^1(\Omega)$.

Canonical Basis Functions & Interpolation

For a local VEM function $v_h \in V_k(E)$, the DoFs on a local VEM space are denoted by $\mathrm{dof}_i(v_h)$ for $i = 1, \ldots, N^{\mathrm{dofs}}$, with N^{dofs} being the number of DoFs on E. The Lagrangian VEM basis function is introduced as

$$\{\varphi_i\}_{i=1}^{N^{\mathrm{dofs}}} \subset V_k(E), \qquad \mathrm{dof}_i(\varphi_j) = \delta_{i,j} \text{ for } i,j = 1, \ldots, N^{\mathrm{dofs}}. \tag{2.5}$$

From this definition, any $v_h \in V_k(E)$ can be written as

$$v_h = \sum_{i=1}^{N^{\mathrm{dofs}}} \mathrm{dof}_i(v_h)\varphi_i. \tag{2.6}$$

For any $v \in H^2(E)$, the VEM Lagrange interpolant can be defined using Equation (2.6):

$$v_I = \sum_{i=1}^{N^{\mathrm{dofs}}} \mathrm{dof}_i(v)\varphi_i. \tag{2.7}$$

From this, the following interpolation error estimate holds.

Theorem 2.1.2 (VEM Interpolation)**.**
Let a polynomial degree $k \in \mathbb{N}$ and an integer $s \in \{2, \ldots, k+1\}$ be given. Then for any element $E \in \mathcal{T}_h$ and all $v \in H^s(E)$, the VEM interpolant $V_I \in V_k(E)$, defined by Equation (2.7)*, satisfies*

$$\|v - v_I\|_{0,E} + h_E \, |v - v_I|_{1,E} \lesssim h_E^s \, |v|_{s,E},$$

where the hidden constant is dependent on ϱ and k.

Proof. The inequality is given (see Proposition 4.3 in [18]) by classical Scott-Dupont estimates [29]. □

Remark 2. *In the case of $v \notin H^2(E)$, Theorem 2.1.2 can be generalised to infer the existence of a quasi-interpolant $v_I \in V_k(E)$ that satisfies the same approximation properties [29].*

Approximation of Linear and Bilinear Forms

Let $\mathcal{A}(\cdot,\cdot) : H_0^1(\Omega) \times H_0^1(\Omega) \to \mathbb{R}$ and $l(\cdot) : H_0^1(\Omega) \to \mathbb{R}$ be a bilinear and linear form respectively. Consider the general variational problem: find $\rho \in H_0^1(\Omega)$ such that $\mathcal{A}(\rho, v) = l(v)$ for all $v \in H_0^1(\Omega)$. The bilinear and linear forms can be decomposed into contributions over polygonal elements

$$\mathcal{A}(\rho, v) = \sum_{E \in \mathcal{T}_h} \mathcal{A}^E(\rho, v), \qquad l(v) = \sum_{E \in \mathcal{T}_h} l^E(v).$$

The novelty of the VEM is the introduction of discrete approximations $\mathcal{A}_h(\cdot,\cdot)$ and $l_h(\cdot)$ that use the DoFs to compute the projection operators Π^*, Π^0 and Π^1 and use these projections to approximate the integral equations to a sufficient degree of accuracy. These approximations are also constructed via local element contributions

$$\mathcal{A}_h(\rho_h, v_h) = \sum_{E \in \mathcal{T}_h} \mathcal{A}_h^E(\rho_h, v_h), \qquad l_h(v_h) = \sum_{E \in \mathcal{T}_h} l_h^E(v_h) \qquad \forall \rho_h, v_h \in V_{h,0}.$$

To enforce the coercivity of the discrete bilinear form, a stabilisation term is introduced $S^E(\cdot,\cdot)$ to ensure that the kernel of $\mathcal{A}_h^E(\cdot,\cdot)$ scales like the kernel of $\mathcal{A}^E(\cdot,\cdot)$. There exists VEMs within the literature for which stabilisation of specific bilinear forms are not required [2, 44, 39].

An example construction of the bilinear forms is presented through the simplest case of Poisson's Equation [18]

$$\begin{aligned} -\Delta\rho &= f \qquad \mathbf{x} \in \Omega, \\ \rho &= 0 \qquad \mathbf{x} \in \partial\Omega. \end{aligned}$$

The continuous Galerkin formulation is well known [29, 74]

$$\mathcal{A}(\rho, v) := \int_\Omega \nabla\rho \cdot \nabla v \; d\mathbf{x} = \int_\Omega f v \; d\mathbf{x} := l(v),$$

and can be decomposed into a summation over local elements $E \in \mathcal{T}_h$

$$\mathcal{A}(\rho, v) = \sum_{E \in \mathcal{T}_h} \mathcal{A}^E(\rho, v), \qquad l(v) = \sum_{E \in \mathcal{T}_h} l^E(v),$$

where

$$\mathcal{A}^E(\rho, v) = \int_E \nabla\rho \cdot \nabla v \; d\mathbf{x}, \qquad l^E(v) = \int_E fv \; d\mathbf{x}.$$

To derive an appropriate VEM, the following assumptions on the discrete bilinear form are sufficient. In [18] these conditions are utilised to prove optimal order H^1 error bounds for a VEM approximation to Poisson's Equation.

Assumption 2 (Polynomial k-consistency)**.**
Let $k \in \mathbb{N}$ and $\mathcal{T}_h$ be a polygonal mesh that satisfies Assumption 1 with $E \in \mathcal{T}_h$. For a VEM approximation of $\mathcal{A}_h^E(\cdot,\cdot)$ of a bilinear form $\mathcal{A}^E(\cdot,\cdot)$ it holds that

$$\mathcal{A}_h(p, v_h) = \mathcal{A}(p, v_h) \qquad \forall p \in \mathbb{P}_k(E), v_h \in V_k(E).$$

Assumption 3 (Stability)**.**
Let $k \in \mathbb{N}$ and $\mathcal{T}_h$ be a polygonal mesh that satisfies Assumption 1 with $E \in \mathcal{T}_h$. For a VEM approximation of $\mathcal{A}_h^E(\cdot,\cdot)$ of a bilinear form $\mathcal{A}^E(\cdot,\cdot)$ there exists positive constants $\alpha^*, \alpha_* > 0$ independent of h_E such that

$$\alpha_* \mathcal{A}(\rho_h, v_h) \leq \mathcal{A}_h(\rho_h, v_h) \leq \alpha^* \mathcal{A}(\rho_h, v_h) \qquad \forall \rho_h, v_h \in V_k(E).$$

Assumptions 2 and 3 are satisfied by defining the local approximation by

$$\mathcal{A}_h^E(\rho_h, v_h) = \mathcal{A}^E(\Pi_k^\nabla \rho_h, \Pi_k^\nabla v_h) + \mathcal{S}^E(\rho_h - \Pi_k^\nabla \rho_h, v_h - \Pi_k^\nabla v_h),$$

where the stabilisation term is given by a Euclidean inner product of the DoFs

$$S^E(u_h, w_h) = \sum_{i=1}^{N^{\text{dofs}}} \text{dof}_i(u_h) \cdot \text{dof}_i(w_h) \qquad \forall u_h, w_h \in V_k(E), \tag{2.8}$$

and is commonly referred to as"dofi-dofi" stabilisation within the VEM literature [18, 2, 44]. For the discrete linear form, it is sufficient to use the Π_{k-2}^0 projection of f, under the assumption that $f \in H^{k-1}(E)$, to obtain an optimal order accuracy VEM

$$l_h^E(v_h) = \int_E \Pi_{k-2}^0 f \; v_h \; d\mathbf{x}.$$

Remark 3. *For approximating the Laplacian operator, the stabilisation does not require any scaling, as $\mathcal{A}(\cdot,\cdot)$ scales as one on a given $E \in \mathcal{T}_h$ [18]. However, in general, the VEM stabilisation will have to be scaled according to the dimension, PDE coefficient terms, and the measure of E [39, 36, 48]. For example, in the two-dimensional case of the L^2 inner product, the dofi-dofi stabilisation is scaled by h_E^2.*

Extensions of the Virtual Element Method

This section concludes with a mention of extensions and variations of the VEM. The classical VEM formulation and analysis in three dimensions have been well-established [2, 44, 39]. Additionally, the VEM has been extended to non-conforming [39, 11], higher regularity [33, 6], and H(div), H(curl) spaces [45, 22]. Mixed VEMs have also been introduced [31, 19, 47, 46]. These discretisations are not considered in this thesis. In Parts II and III, the curved VEM proposed in [23] will be employed, which is reviewed in Chapter 6.

2.2 Moving Mesh Methods

In this section, an overview of moving mesh methods is presented for time-dependent problems on moving domains. Moving mesh methods are part of a large class of adaptive mesh refinement techniques, alongside h- and p-refinement strategies. Moving mesh methods also have a close relation to the Arbitrary Lagrangian-Eulerian (ALE) method in computational fluid dynamics [50]. The fundamentals of the ALE method are also reviewed in this section.

The primary advantage of moving mesh methods is the ability to optimise mesh structures without requiring any change in the mesh connectivity, which can cause computational challenges, particularly when parallel implementations are considered. They also provide a natural framework for tracking physical features of a time-dependent PDE, such as blow-up problems [34], phase

change modeling [17], fluid-structure interaction problems [89], and more general time-dependent PDEs [13, 85, 62].

The primary goal of a moving mesh method is to evolve the coordinates of a mesh over time in an optimal manner to maintain the accuracy of the numerical simulation and provide computationally efficient resolution of moving solution features. There are several approaches within the FEM literature, including the moving FEM [13], the MMPDE method [65, 66], the Geometric Conservation Law (GCL) method [40], and a velocity-based moving mesh algorithm [14, 15].

Moving Coordinate Systems

A moving domain is denoted by Ω_t and within this thesis it is assumed that Ω_t is an open, bounded and simply connected subset of $\mathbb{R}^2$ at any given time. Frequently, Ω_t will be referred to as the "physical domain". A time-independent "reference domain" is denoted by $\hat{\Omega}$ and is also assumed to be an open, bounded and simply connected subset of $\mathbb{R}^2$.

An invertible mapping between $\hat{\Omega}$, and Ω_t is defined for a finite time $0 \leq t \leq T < \infty$ by

$$\mathbf{x}(\boldsymbol{\xi}, t) : \hat{\Omega} \times [0, T] \rightarrow \Omega_t.$$

where $\boldsymbol{\xi}$ is a coordinate system of the reference domain. Typically the reference domain is chosen to be the initial domain $\hat{\Omega} = \Omega_0$ or a discrete approximation of it. An example of such a mapping is given in Figure 2.3.

In the literature of computational fluid-dynamics $\mathbf{x}(\boldsymbol{\xi}, t)$ is often referred to as the Lagrangian coordinate system, owing to the fact that this typical represents the moving coordinates of particles in a body of fluid. Following the same reasoning $\boldsymbol{\xi}$ is referred to as the Eulerian coordinate system. The Lagrangian velocity field is also defined as the time derivative of $\mathbf{x}(\boldsymbol{\xi}, t)$ and is denoted by $\mathbf{u} = \dot{\mathbf{x}}(\boldsymbol{\xi}, t)$.

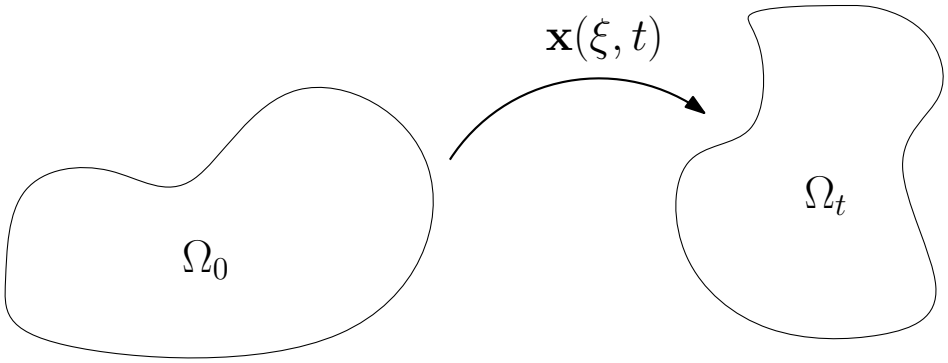

Figure 2.3: A translation between two domains with respect to time. The function $\mathbf{x}(\boldsymbol{\xi}, t)$ maps the coordinate system from the reference domain to the physical one.

The Arbitrary Lagrangian-Eulerian Framework

The ALE method was motivated by the inefficiency of Lagrangian and Eulerian approaches to problems in computational fluid dynamics. The Lagrangian approach allows for simple tracking of solution features and moving boundaries. However, the robustness of this approach depends on the fluid flow, and frequently, the moving mesh degrades, resulting in poor-quality meshes. On the other hand, the Eulerian approach fixes the computational mesh, preventing mesh quality issues but at the cost of poor resolution of flow details and moving boundaries [50]. The ALE method allows for an arbitrary transformation between $\hat{\Omega}$ and Ω_t that exhibits the benefits of both the Eulerian and Lagrangian approaches. The ALE mapping is defined by

$$\mathscr{A}_t : \hat{\Omega} \times [0, T] \to \Omega_t,$$

with a corresponding ALE velocity field denoted by $\mathbf{w} = \dot{\mathscr{A}}_t$.

ALE methods allow for the evaluation of terms between the three frames of reference: the Lagrangian, Eulerian and ALE reference frames. In practise only transformations between the ALE and Eulerian coordinates are required [50]. A function $f : \Omega_t \to \mathbb{R}$ can be represented in the ALE and Eulerian coordinates via

$$\hat{f} = f \circ \mathscr{A}_t, \qquad f = \hat{f} \circ \mathscr{A}_t^{-1}. \tag{2.9}$$

Consistent with the literature, functions and operators defined on the reference domain will represented using the hat notation [50, 89, 60]. Additionally, the Jacobian operator of the transformation is defined on the reference domain via

$$\mathbf{J}_{\mathscr{A}_t} = \frac{\partial}{\partial \boldsymbol{\xi}} \mathscr{A}_t = \hat{\nabla} \mathscr{A}_t, \tag{2.10}$$

with the corresponding time-dependent determinant defined by $j = \det(\mathbf{J}_{\mathscr{A}_t})$. In order to perform the analysis of these ALE methods, regularity conditions on the ALE mapping must be imposed [54, 60]. These conditions are discussed in Parts II and III.

A Conservative Arbitrary Lagrangian-Eulerian Formulation

In this thesis, only the conservative ALE method is presented. Examples of non-conservative formulations can be found in [54, 27, 28, 81]. The material derivative operator plays an important role in the formulation of an ALE method. This is defined as the time evolution of a function along the velocity field trajectory

$$\mathcal{D}_t f = \frac{\partial f}{\partial t} + \mathbf{w} \cdot \nabla f.$$

Important to the formulation and analysis of ALE methods is the well-known Reynolds Transport Theorem.

Theorem 2.2.1 (Reynolds Transport Theorem)**.**
Let Ω_t be a time-dependent volume for $0 \leq t \leq T < \infty$ with velocity $\mathbf{w}$. For $t \in [0, T]$, it holds for a scalar function f defined on Ω_t that

$$\frac{d}{dt} \int_{\Omega_t} f \, d\mathbf{x} = \int_{\Omega_t} \frac{\partial f}{\partial t} + \nabla \cdot (f \mathbf{w}) \, d\mathbf{x}.$$

Proof. Proofs can be found in [100, 86]. □

In the presentation of the material derivative and the Reynolds Transport Theorem, The velocity field is given by $\mathbf{w}$ to denote the time derivative of the domain transformation. This will be used in this thesis to denote an ALE velocity field. Particle velocity fields or physical derived domain transformations

will be denoted by $\mathbf{u}$, noting that $\mathbf{w} = \mathbf{u}$ is valid in Reynolds Transport Theorem. In the derivation of the Navier-Stokes equations, $\mathbf{u}$ is commonly used to denote the velocity field of fluid particles [100].

For a given PDE $\partial\rho/\partial t = \mathcal{L}\rho$, where $\mathcal{L}$ is some differential operator, the Reynolds Transport Theorem is used to derive a Galerkin formulation of the PDE by considering a test function v

$$\frac{d}{dt}\int_{\Omega_t} \rho v \; d\mathbf{x} = \int_{\Omega_t} \frac{\partial(\rho v)}{\partial t} + \nabla\cdot(\mathbf{w}\rho v) \; d\mathbf{x}.$$

Applying the product rule and rearranging terms leads to

$$\frac{d}{dt}\int_{\Omega_t} \rho v \; d\mathbf{x} = \int_{\Omega_t} \rho\mathcal{D}_t v + v\left\{\mathcal{L}\rho + \nabla\cdot(\mathbf{w}\rho)\right\} \; d\mathbf{x}. \tag{2.11}$$

In discretising this equation it is assumed that the test function moves with the mesh velocity, i.e. $\mathcal{D}_t v = 0$ so that the right hand side of Equation (2.11) can be computed using only the PDE information and the mesh velocity field

$$\frac{d}{dt}\int_{\Omega_t} \rho v \; d\mathbf{x} = \int_{\Omega_t} v\mathcal{L}\rho + v\nabla\cdot(\mathbf{w}\rho) \; d\mathbf{x}. \tag{2.12}$$

Indeed, when the velocity field is represented discretely in the Lagrangian Finite Element space, this property is well-known [68]. However, non-Lagrangian Finite Element representations require an additional integral term for computation [68, 65]. In the FEM, an approximation of the right-hand side of Equation (2.12) is computed, and integration over time is performed using a standard time-stepping scheme, such as an implicit Euler scheme [54]. The solution is then reconstructed on a new mesh using a standard global L^2 projection [14, 65]. In computational fluid dynamics, it is common to transform the integral equations to the reference domain using Equations (2.9) and (2.10). Details of this are deferred until Parts II and III of this thesis.

Mesh Updates

In general, a moving mesh method requires either the computation of a mesh velocity or mapping between the current discrete time level and the next one.

In practical applications it is required that the trace of $\mathcal{A}_t$ and consequently the trace of $\mathbf{w}$ corresponds to the Lagrangian domain transformation over time (or a sufficiently accurate approximation of it).

A mesh movement strategy can be described as *location-based* if the ALE map is solved for by a numerical scheme or *velocity-based* if the ALE velocity field is approximated and subsequently integrated with respect to time to compute the ALE map [65].

Transferring Functions Between Meshes

In general, there is no hierarchical structure between meshes at different time levels, which raises the question: How can a solution be transferred between one mesh and the next? The choice of transferring a numerical solution between meshes falls into one of two categories [65].

A *rezoning* approach seeks to simulate the PDE independently of the mesh movement. To achieve this a transfer operator is required to map a numerical solution between unrelated meshes at a fixed discrete time level. Once the transfer is computed the numerical method is computed with the mesh positions fixed.

A *quasi-Lagrange* approach considers the mesh vertices as moving continuously with respect to time. Here the evolution of the PDE along the mesh trajectories are approximated. The mesh and solution are therefore updated in time simultaneously. The conservative ALE update given by Equation (2.11) is an example of a quasi-Lagrange approach.

Choices of ALE Mappings

The best choice of ALE mapping often depends on the specific problem being solved. A degree of choice is also afforded to the user as to how much computational effort is required to maintain an optimal moving mesh. The choice of ALE mapping can be driven by the desire to improve the quality of the

numerical simulation. This is achieved by, for example, introducing a function that monitors the distribution of solution quantities and seeks to maintain alignment and equidistribution properties of this monitor over time [65].

In most cases the goal of the ALE mapping is to improve the robustness of the moving mesh method and avoid issues such as node tangling and element degeneration which eventually lead to numerical instabilities in the method [50, 14, 89]. Two commonly used approaches are described below. Depending on whether a location-based or velocity-based mesh movement strategy is employed, the ALE mapping or the ALE velocity field can be solved for respectively using these approaches. For brevity, the location-based strategy case is presented. The velocity-based counterparts simply involve the substitution of $\mathscr{A}_t$ and $\mathbf{x}$ with $\mathbf{w}$ and $\mathbf{u}$ respectively in Equations (2.13), (2.14), (2.15), and (2.16) [50].

The flexibility of the ALE method allows, generally, for the interior mesh elements to move arbitrarily. Consequently, there is no requirement to compute the interior value of the ALE map or velocity field to high precision. However, it has been shown that the ALE method is more robust under complex domain transformations when variational approaches are taken that improve the regularity of the ALE map [60, 89].

Laplacian smoothing seeks to regularise the ALE mapping separately in both coordinate directions. As such the method is given by a simple Poisson's Equation in which the Lagrangian map is taken as Dirichlet boundary conditions [50]

$$\Delta \mathscr{A}_t = 0 \qquad \boldsymbol{\xi} \in \hat{\Omega}, \tag{2.13}$$

$$\mathscr{A}_t = \mathbf{x}(\boldsymbol{\xi}, t) \qquad \boldsymbol{\xi} \in \partial\hat{\Omega}. \tag{2.14}$$

A more complex approach is to treat the moving mesh as an elastic body subject to a Navier-Lamé equation [60, 55]. For a scalar parameter $\lambda \geq 0$, the

ALE map is given by

$$-\Delta \mathscr{A}_t + \lambda \nabla(\nabla \cdot \mathscr{A}_t) = \mathbf{0} \qquad \boldsymbol{\xi} \in \hat{\Omega}, \tag{2.15}$$

$$\mathscr{A}_t = \mathbf{x}(\boldsymbol{\xi}, t) \qquad \boldsymbol{\xi} \in \partial\hat{\Omega}. \tag{2.16}$$

The special case of $\lambda = 0$ is commonly referred to as harmonic extension.

2.3 Literature Review

To conclude this chapter, a review is provided of the recent works that investigate polygonal moving mesh methods. The mathematical details of these studies are omitted here, but the required preliminary material can be found in Sections 2.1 and 2.2, and further information can be obtained from the respective publications. Finally, a brief mention is made of additional research indirectly related to moving meshes and polygonal discretisations.

A Conservative ALE Remapping Method

Lipnikov and Morgan's work [78, 80] presented a moving polygonal mesh scheme using a discontinuous Galerkin method with a conservative ALE update. They mapped a discrete function between two curvilinear polygonal meshes via a pseudo PDE problem on a moving mesh between the original and final meshes. They employed the VEM to approximate integrals that would otherwise be uncomputable.

They constructed the moving coordinates system by linearising the difference between the two meshes by a pseudo-time parameter. By defining the moving geometry and time derivative of the function, which is zero under movement, they could simplify the Reynolds transport theorem into a computable integral form.

A semi-discrete method subdivided the pseudo-time interval into segments, each with a corresponding sequence of geometries and meshes. A discrete

Jacobian was approximated via a virtual element space to approximate any integration back to the initial discrete space.

They defined a fully discrete system with mass term and right-hand side, and the system could be computed solely on the initial mesh with the final mesh only required at the end of the pseudo-time stepping scheme. At the end of the algorithm, the H^1 projection was applied to the solution interpolated onto a virtual element space to ensure the local weight of the reconstructed polynomial was consistent with that of the solution of the pseudo-time problem solved on the initial mesh.

They demonstrated the volume of the solution over the domain was preserved during the pseudo-time stepping. Numerical results suggested optimal accuracy in L^2 and L^∞ norms for polynomial degrees $k = 0, 1, 2$. The method proved a moving polygonal mesh scheme was possible, with flexibility in defining the mesh movement, allowing for more generalised representations of the velocity field.

However, limitations of this method included its restriction, to a fixed function on a fixed domain without a moving boundary. Further research would be needed for extension to time-dependent PDE problems or domains with moving boundaries. The necessity to use a $k+1$ order polynomial projection of the velocity field requires examination, as it's uncommon in typical VEMs.

Mappings between moving polygons

In the paper of Huang and Wang [66], the quality measures of polygonal meshes, specifically focusing on anisotropic mesh refinement, are studied. They proposed a set of quality measures for Voronoi meshes and utilised the MMPDE scheme [65] for the refinement of adaptive moving meshes. Their study also included numerical experiments that employed the Wachspress finite element method [98] for the anisotropic solution of Laplace's equation.

The paper considered two different types of meshes: computational and

physical. For every physical mesh, they assumed a corresponding reference polygon in the computational mesh. They connected these polygons through a bijection, ensuring only mappings that preserve the polygon's connectivity and map nodes to nodes.

The quality of the polygonal mesh was assessed based on the integral of a metric tensor over a particular domain [65]. They established that a high-quality physical mesh should maintain a constant ratio between this metric integral and the area of the domain. Two conditions are utilised, alignment and equidistribution, to define the mesh quality. When adapting mesh movement, an ideal mesh aligns and equitably distributes, such that the quality functions equal one.

Three approximations of the mapping are suggested in this paper. A linear least squares fitting method provides an approximation for all possible maps that satisfy a specific condition. It employs the least squares approach to compute the map, taking into account the nodal coordinates. This approach is applicable for convex, non-degenerate polygons. This method provides an approximate measure of the mesh quality based on the Jacobian. Generalised barycentric mappings is another approach that constructs a specific mapping using generalised barycentric coordinate systems. These coordinates are functions that satisfy specific conditions and are particularly applicable for a given N-gon. The mapping can then be defined using these coordinates. There are two possible coordinate systems for convex polygons: the Wachspress coordinates and the piecewise linear coordinates. Each system has its own way of defining the coordinates and the mesh quality measures. The third approach utilises affine mapping by extending to higher dimensions. Here, a set of infinitely many reference polygons are used to ensure the mapping is affine. The polygon in question is assumed to be translated such that the polygon centroid is at the origin. A mapping can then be defined through a hyperplane translation. The quality of the mesh is then defined using a specific Jacobian

equation.

The study applies the moving mesh PDE method (MMPDE) [65] to adaptively refine the positioning of mesh vertices based on a metric tensor dependent on the Hessian of the solution. The movement of mesh nodes is determined using piecewise barycentric mappings on sub-triangles. The method demonstrates comparable convergence rates to uniform mesh refinement, while achieving smaller numerical errors in solution accuracy.

Preliminary results compare three mesh quality measures for generating a centroidal Voronoi tessellation (CVT) using Lloyd's iteration [51]. The results show convergence of the quality measures to 1, with the piecewise linear measure exhibiting more volatile behavior. Two test problems involving Poisson's equation on a unit square domain are considered, showcasing strong anisotropic behavior in the solution gradient and Hessian. The MMPDE method, using the defined metric tensor, achieves similar convergence rates to uniform mesh refinement, but with smaller numerical errors in both test problems. The mesh quality measures also converge to 1 for each moving mesh refinement, including cases with modified metric tensors and an infinite solution Hessian at the origin.

The paper's contribution lies in the methodology of representing mesh transformations, allowing for flexibility in approximating translations between elements in a moving polygonal mesh method. Different mapping techniques demonstrate the potential for robust and computable mesh quality and alignment measures during r-refinement. However, the study does not explore the coupling of this adaptive algorithm with time-dependent PDEs or the integration of translation choices within the computational mesh. The use of subtessellations combined with polygonal discretisation schemes for PDEs presents a promising avenue for future moving mesh methods.

PDE based reconstruction on non-hierarchical meshes

In a paper by Cangiani, Georgoulis, and Sutton [38], an alternative approach to moving mesh methods is proposed. The method utilises discrete elliptic operators to transfer the numerical solution of a PDE between meshes. Residual-based a posteriori error estimates are derived, and the method is tested on a moving mesh method for a parabolic PDE.

The paper considers parabolic diffusion problems defined on a fixed bounded domain. The PDEs are solved numerically using finite element or virtual element frameworks. Bilinear forms are defined, along with corresponding norm and projection operators. The numerical method involves solving a discrete scheme for the PDE, with a discrete derivative and elliptic reconstruction of the solution. Error estimates are provided for the $L^2(H^1)$ and $L^\infty(L^2)$ norms.

Computable transfer operators, such as the coarsening operator and refinement operator, are introduced in the VEM. These operators interpolate degrees of freedom and compute H^1 projections, respectively. Estimators for the elliptic reconstruction, including L^2 and H^1 estimators, are derived based on jump residuals and element residuals. A computable estimator for the temporal component is not presented in the paper but can be adapted from the finite element counterpart.

Empirical results demonstrate the convergence and accuracy of the method. The moving mesh method using classical finite element methods and VEM both exhibit optimal convergence rates on various test cases. The method performs well on parabolic problems with rapidly moving solution features, with a refined mesh strategy based on error estimators.

The a posteriori error estimates provided in the paper offer a solid foundation for further analysis of moving polygonal mesh methods. The numerical results support the effectiveness of the virtual element method in handling rapidly changing mesh geometries. The $L^2(H^1)$ norm estimator is found to be particularly reliable for solutions with sharp features. However, the paper does

not address time-dependent domains, leaving room for future exploration of the impact of inconsistent meshes with the true domain boundary.

Space-time Elements on Moving Voronoi Meshes

In the work of Gaburro et al. [59, 58] a moving mesh method is proposed that is based on Voronoi tessellations to construct space-time elements for simulating hyperbolic PDEs while satisfying the geometric conservation law. The method combines an ALE Finite Volume method with a Discontinuous Galerkin method to achieve high-order accuracy and robustness.

The mesh is generated using Voronoi tessellations, which are regenerated at each discrete time level. The movement of the mesh is determined by an ALE velocity field that aims to maintain quasi-uniformity and emulate Lagrangian behavior. The positioning of mesh generator points is computed based on averaging the extrapolated Lagrangian velocity and an optimal seed position derived from Lloyd's algorithm [51]. Higher order representation of the mesh movement is achieved by considering a Taylor expansion of the generator points.

Space-time elements are constructed by connecting the edges of the polygons between consecutive time states. For standard elements, quadrilateral faces connect the edges, while additional considerations are made for degenerate edges in sliver elements. The method includes a predictor step, which is performed locally on each space-time element, and a corrector step that integrates the weak form of the PDE globally using the predictor solutions.

Extensive numerical results demonstrate the method's effectiveness and accuracy for a range of test cases, including vortex flows, explosion problems, the Sedov problem, and magnetohydrodynamics. The method exhibits high-order convergence, robustness for long-time simulations, and the ability to handle highly rotational flows without suffering from mesh tangling.

While the method shows promising results, some limitations and areas for further investigation are mentioned. The explicit knowledge of the domain ge-

ometry, particularly the rectangular region, is required, raising questions about extending the approach to moving boundary problems. The impact of using a low-order Finite Volume limiter has not been fully explored, and its frequent use may result in a loss of convergence. The reconstruction-based nature of the method eliminates the need for knowledge of the material derivative operator, distinguishing it from other moving mesh methods.

Additional Works

To achieve higher-order accuracy in simulating moving boundary problems, accurate representation of the boundary is crucial. Curved polytopic elements offer the ability to achieve higher accuracy for PDEs on domains that deviate from typical rectangular benchmarks. In the virtual element literature, there have been significant research efforts to enhance the flexibility of this method for curved elements. Promising results have been obtained through various approaches [23, 26, 21, 9, 46], demonstrating optimal convergence rates for benchmark elliptic problems. To attain convergence rates higher than quadratic order, it is essential to improve the representation of the moving boundary by incorporating curved edge elements and assessing their impact on moving mesh methods.

Moving surface and surface PDE problems are related to coupled moving mesh and PDE models. Previous studies on moving surface FEMs include [52, 53, 73, 70, 72, 71]. The analysis approaches used in these works can potentially be extended to a virtual element discretisation for similar problems. Additionally, there are works in the VEM literature focusing on surface PDEs [57, 56, 12, 76], which closely relate to moving mesh methods and provide theoretical foundations to analyse each component of the method separately.

To conclude this review, two additional relevant papers are mentioned. In [83], a linear VEM is coupled with a mesh agglomeration algorithm for long-term simulations of moving landforms. In [7], a cut-based discontinuous

Galerkin scheme for fluid-structure interaction is introduced and tested. Both papers demonstrate that changes in mesh connectivity can be successfully coupled with polygonal discretisation techniques and produce accurate solutions.

Part I

A Velocity-based Moving Mesh Virtual Element Method

Introduction

Within the literature there did not exist an implementation and/or analysis of a moving mesh method that fully depended on the VEM. In order to motivate the development of moving mesh VEMs, it is imperative to demonstrate that the VEM can be effectively applied to an elementary instance of a moving mesh problem.

In Part I a VEM discretisation of a moving mesh method is proposed and benchmarked against a series of numerical experiments. Here a *velocity-based* moving mesh algorithm [14, 16, 17, 67, 81, 15] is considered for the numerical solution of free boundary problems. The method is closely related to the Geometric Conservation Law method (GCL) [40] and also forms part of a larger family of adaptive moving mesh methods discussed in Section 2.2. It uses a Lagrangian formulation of the given PDE to solve directly for the mesh velocities which are integrated over time to evolve the mesh. The solution computed on any given time-step is then transferred to the following time-step using an ALE scheme based on a weak distribution of a given monitor function. The lowest order VEM discretisation is formulated for this moving mesh method. This new method is presented focusing on the solution of the free boundary problem for the Porous Medium Equation (PME) as a model problem. This method can also be regarded as an extension of the linear moving mesh FEM to polygonal meshes that reduces to the original algorithm [14] when the mesh consists of only triangles.

In Chapter 3 the moving mesh method is reviewed and a VEM formulation

of the mesh velocity and ALE update are presented. The discrete equations are derived using the classical VEM spaces defined in Section 2.1. A conservation law is then presented for the linear moving mesh VEM.

In Chapter 4 numerical results are presented for a similarity solution of the PME. To showcase the methods generality, an extension of the method to a nonlinear fourth-order problem is formulated and numerically tested. To demonstrate the advantages of using a polygonal discretisation in moving mesh methods, a simple node insertion/removal algorithm for problems involving collision between a moving boundary and fixed obstacles is outlined and tested. The benefit of this algorithm is that typical issues such as node tangling, contact with obstacles, and topological changes of the moving domain [65, 15] can be dealt with fully by local changes in the mesh topology.

A selection of results from Part I has been submitted for publication [99].

Chapter 3

A Virtual Element discretisation of a Moving Mesh Algorithm

The velocity-based moving mesh method was first proposed in [14] for non-linear diffusion problems (such as the PME). The solution and velocity spaces were discretised using a linear Lagrange finite element space on the moving mesh. In this chapter we present the formulation of the two-dimensional moving mesh method for the PME using the VEM. Whilst the method has been successfully applied to a variety of parabolic problems with different solution features, our presentation is restricted to the PME with an extension demonstrated in Chapter 4. A summary review of the method is provided in [15] and provides a framework in which the moving mesh VEM could be extended to these problems.

In Section 3.1 we introduce the PME problem and the associated free-boundary problem which the moving mesh method will be derived from. The velocity-based moving mesh method is reviewed in Section 3.2. The VEM space employed in the proposed method is outlined in Section 3.3. The discretisations of the velocity and solution problems are presented in Sections 3.4 and 3.5 respectively. The chapter is concluded with a conservation law result presented in Section 3.6 and an overview of the algorithm in Section 3.7.

3.1 The Porous Medium Equation

On an open, bounded domain $\tilde{\Omega} \subset \mathbb{R}^2$, we consider the following initial-boundary value problem for the PME: find $\rho : \tilde{\Omega} \times \mathbb{R}^+ \to \mathbb{R}$ such that

$$\frac{\partial \rho}{\partial t}(\mathbf{x}, t) = \Delta \Phi(\rho(\mathbf{x}, t)) \qquad (\mathbf{x}, t) \in \tilde{\Omega} \times (0, \infty], \tag{3.1}$$

$$\rho(\mathbf{x}, t) = 0 \qquad (\mathbf{x}, t) \in \partial\tilde{\Omega} \times (0, \infty],$$

$$\rho(\mathbf{x}, 0) = \rho_0(\mathbf{x}) \qquad \mathbf{x} \in \tilde{\Omega}, \tag{3.2}$$

where $\Phi = \rho^{m+1}/(m+1)$, for some $m > 0$, and $\rho_0 \geq 0$ having compact support in $\tilde{\Omega}$. The PME belongs to the broader class of Generalised Porous Medium Equations (GPME), also known as filtration equations, obtained with $\Phi : \mathbb{R}^+ \to \mathbb{R}^+$ any increasing function. The mathematical analysis of the GPME is well developed; see the monograph [97] and the references therein. In particular, the notion of appropriate weak solutions is discussed in [97] where it is shown that, for non-negative $\rho_0 \in L^1(\tilde{\Omega})$ and for $\Psi(\rho_0) \in L^1(\tilde{\Omega})$, where Ψ is the anti-derivative of Φ, if $\Psi(\rho) > 0$ for $\rho > 0$, there exists a unique non-negative weak solution to the GPME globally in time.

The PME models a number of physical processes such as fluid flow, heat transfer, and diffusion. It exhibits several interesting properties, including the existence of a family of radially symmetric similarity solutions [86], which are used to test the numerical method in Chapter 4. Other properties of the PME are discussed in [97, 15, 87].

It is a fundamental example of a degenerate parabolic equation, stemming from the condition that Ψ is non-negative, rather than simply positive.

Solutions that exhibit an evolving compact support are ideal for the class of moving mesh methods considered herein because considering as unknown the support of the solution leads to a moving boundary problem that can be simulated over time without having to discretise the entire geometry of $\tilde{\Omega}$.

Introducing the time-dependent support of ρ as Ω_t, we define the time-dependent coordinate system $\mathbf{x}(\boldsymbol{\xi}, t)$ with a velocity field $\dot{\mathbf{x}} = \mathbf{u}(\mathbf{x}, t)$ that

corresponds, at $\partial\Omega_t$, with the movement of the free boundary of Ω_t for all $t \in [0, \infty]$. Additionally, we consider the free boundary problem for the PME in which part of the boundary may be obstructed by a fixed object. To this end, the boundary is divided into a moving part $\partial\Omega_t^M$ and a fixed part $\partial\Omega_t^F$, such that $\partial\Omega = \partial\Omega_t^M \cup \partial\Omega_t^F$ and $\partial\Omega_t^M \cap \partial\Omega_t^F = \emptyset$. Thus, we arrive to the following classical free boundary problem for the PME, whose smooth solutions are weak solutions of (3.1)-(3.2), cf. [97].

Problem 3.1.1 (The Porous Medium Equation (PME))**.**
Let $T > 0$ and $m > 0$. Find $\rho = \rho(\mathbf{x}, t)$ such that $\rho(\mathbf{x}, 0) = \rho_0(\mathbf{x})$ for $\mathbf{x} \in \Omega_0$ and, for all $t \in (0, T]$,

$$\begin{aligned} \frac{\partial \rho}{\partial t} &= \nabla \cdot (\rho^m \nabla \rho) && \mathbf{x} \in \Omega_t, \\ \rho &= 0 && \mathbf{x} \in \partial\Omega_t^M, \\ \rho^m \nabla \rho \cdot \mathbf{n} &= 0 && \mathbf{x} \in \partial\Omega_t^F. \end{aligned}$$

Here, $\mathbf{n}$ is the outward pointing unit normal to the boundary $\partial\Omega$. Note that an additional (kinematic) boundary condition,

$$\rho \mathbf{u} \cdot \mathbf{n} = -\rho^m \nabla \rho \cdot \mathbf{n} \qquad \mathbf{x} \in \partial\Omega_t^M,$$

which imposes zero flux through the moving boundary, is required to determine the boundary velocity $\mathbf{u}$. On $\partial\Omega_t^F$, the boundary velocity is specified and $\rho\mathbf{u}\cdot\mathbf{n} = 0$.

3.2 Algorithm Review

We present the derivation of this method for a general time-dependent PDE of the form

$$\frac{\partial \rho}{\partial t} = \mathcal{L}\rho, \tag{3.3}$$

with $\mathcal{L}\rho$ being a generic spatial differential operator in $\mathbb{R}^2$. In the case of the PME in Section 3.1 we have $\mathcal{L}\rho = \nabla \cdot (\rho^m \nabla \rho)$.

Monitor Distribution

The movement of the mesh is derived by specifying the time evolution of the distribution of the spatial integral of some solution-dependent monitor function, $\mathbb{M}(\rho)$. Specifically, the velocity field $\mathbf{u}$ is determined by requiring that the initial distribution of the monitor function $\mathbb{M}(\rho)$ is conserved as time progresses.

We define the space-time domain as

$$\mathcal{Q}_T = \{(\mathbf{x}, t) \,:\, t \in [0, T], \mathbf{x} = \mathbf{x}(\boldsymbol{\xi}, t), \boldsymbol{\xi} \in \Omega_0\}.$$

The goal of this moving mesh method is to seek a $\mathbf{u}$ such that, for all $v \in L^2(\mathcal{Q}_T)$, the coordinates $\mathbf{x}(\boldsymbol{\xi}, t)$ satisfy

$$\frac{\int_{\Omega_t} v(\mathbf{x}, t)\, \mathbb{M}(\rho(\mathbf{x}, t))\; d\mathbf{x}}{\int_{\Omega_t} \mathbb{M}(\rho(\mathbf{x}, t))\; d\mathbf{x}} = \frac{\int_{\Omega_0} v(\boldsymbol{\xi}, 0)\, \mathbb{M}(\rho(\boldsymbol{\xi}, 0))\; d\boldsymbol{\xi}}{\int_{\Omega_0} \mathbb{M}(\rho(\boldsymbol{\xi}, 0))\; d\boldsymbol{\xi}} \qquad \forall\, t \in [0, T]. \quad (3.4)$$

The derivation of the moving mesh method is based on application of the Reynolds Transport Theorem 2.2.1 and on the assumption that the material derivatives of the weighting functions $v(\mathbf{x}, t)$ are zero with respect to the velocity field $\mathbf{u}$. Namely,

$$\mathcal{D}_t v = \frac{\partial v}{\partial t} + \mathbf{u} \cdot \nabla v = 0. \quad (3.5)$$

This is a common assumption made in finite element approaches to moving mesh algorithms as discussed in Section 2.2 and is equivalent to assuming that $v(\mathbf{x}(\boldsymbol{\xi}, \mathbf{t}), t) = v(\boldsymbol{\xi}, 0)$ in Equation (3.4).

Equidistribution-based mesh movement algorithms typically attempt to reduce the global approximation error without changing the number of degrees of freedom by choosing a finite set of weighting functions $v_i(\mathbf{x}, t)$, so that each one has a direct association with a mesh node or element. The monitor function $\mathbb{M}(\rho)$ is then selected to act as a local error indicator, and the mesh is adjusted in an attempt to equidistribute the values of the weighted monitor integrals

$$\mu_t(v_i) = \int_{\Omega_t} v_i\, \mathbb{M}(\rho)\; d\mathbf{x}, \quad (3.6)$$

and hence to equidistribute the local error across the mesh. Such an approach could be followed within our framework [81, 15]. However, in this method we

adopt the following approach which is more akin to Lagrangian mesh movement algorithms, which attempt to move the mesh with the 'flow' velocity.

Choosing $\mathbb{M}(\rho) = \rho$ in equation (3.6) naturally leads to a weak approximation of the Lagrangian 'flow' velocity of the PDE when $\mathcal{L}\rho = \nabla \cdot \mathbf{f}$ for some flux $\mathbf{f}$ in Equation (3.3). This has two benefits: (a) it allows us to predict the movement of free boundaries; (b) it reduces interpolation error of the computed mesh and solution between time-steps because the mesh (and hence the solution) is transported with the velocity field inherent to the PDE.

In many PDEs (including the PME defined in Section 3.1), ρ represents a density, *i.e.* its integral in space is a mass, so we will refer to $\mathbb{M}(\rho) = \rho$ as the mass monitor. For clarity of presentation, we will assume use of this mass monitor from now on. Hence, each weight function $v(\mathbf{x}, t)$ is assigned its own 'mass':

$$\mu_t(v) = \int_{\Omega_t} v\rho \; d\mathbf{x} = c(v)\theta_t \qquad \text{where} \qquad \theta_t = \int_{\Omega_t} \rho \; d\mathbf{x}. \tag{3.7}$$

Equation (3.4) then provides $c(v) = \mu_0(v)/\theta_0$ from the initial conditions of the PDE, and we assume that these distribution coefficients remain constant in time. The evolution of ρ is governed by Equation (3.3), so Equation (3.7) provides us with a way to prescribe the evolution of the coordinate system in a way which retains the initial 'mass' distribution.

The original moving mesh finite element method [14, 15] chose the weight functions to be the standard linear Lagrange finite element test functions on meshes of simplices. This associates a fixed proportion of the total mass of the system with each node of the mesh, so the values of $\mu_t(v)$ depend not only on ρ, but also on the mesh node positions. In this situation, Equation (3.7) provides a way to compute mesh node velocities using standard finite element techniques, in a way which is consistent with local mass conservation when this is a property of the underlying PDE. In this chapter we demonstrate that the same principle can be applied on polygonal meshes within a virtual element

framework.

The Velocity Field

Let $t \in [0,T]$ and consider all $v \in H^1(\mathcal{Q}_T)$ such that $\mathcal{D}_t v = 0$, cf. Equation (3.5). Differentiating the first equality in (3.7) with respect to time and applying the Reynolds Transport Theorem 2.2.1 leads to

$$\dot{\mu}_t(v) = \int_{\Omega_t} v \left\{ \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) \right\} \, d\mathbf{x}. \tag{3.8}$$

Noting that (3.8) does not fully determine the velocity, we further require that each weighting function retains a fixed proportion of θ_t as the mesh evolves. Hence we impose the constraints

$$\dot{\mu}_t(v) = c(v)\dot{\theta}_t. \tag{3.9}$$

Inserting (3.9) in (3.8), using (3.3), and integrating by parts results in

$$c(v)\dot{\theta}_t \; + \; \int_{\Omega_t} \rho \nabla v \cdot \mathbf{u} \; d\mathbf{x} \;\; = \;\; \int_{\Omega_t} v \mathcal{L}\rho \; d\mathbf{x} \; + \; \int_{\partial\Omega_t} v\rho \mathbf{u} \cdot \mathbf{n} \; ds, \tag{3.10}$$

with $\mathbf{n}$ the outward unit normal on $\partial\Omega_t$. Similarly, a direct application of the Reynolds Transport Theorem 2.2.1 to the second equality in (3.7), yields

$$\dot{\theta}_t \;\; = \;\; \int_{\Omega_t} \mathcal{L}\rho \; d\mathbf{x} \; + \; \int_{\partial\Omega_t} \rho \mathbf{u} \cdot \mathbf{n} \; ds. \tag{3.11}$$

In fact, $\dot{\theta}_t$ is typically known explicitly because $\rho\mathbf{u} \cdot \mathbf{n}$ is known on the whole of $\partial\Omega_t$, and the boundary conditions provided with the PDE will enable evaluation of the integral of $\mathcal{L}\rho$. Furthermore, for mass-conservative problems, $\dot{\theta}_t = 0$. Note also that assuming that $c(v)$ remains constant preserves the initial distribution of the mass (or a more general monitor integral) so a monitor which is initially equidistributed between a set of weighting functions should remain equidistributed as the coordinate system evolves with this velocity field.

Equations (3.10) and (3.11) are used to compute an instantaneous velocity $\mathbf{u}$ which is consistent with conserving the proportion of mass associated with each weighting function. This provides a form of local mass conservation when $\dot{\theta}_t =$

0. However, it still does not uniquely define $\mathbf{u}$ in multiple space dimensions. To overcome this issue, the velocity field is written in terms of its Helmholtz decomposition

$$\mathbf{u} = \mathbf{q} + \nabla\phi, \tag{3.12}$$

where ϕ is a scalar potential and $\mathbf{q}$ must be specified. This constraint is equivalent to imposing the curl of the velocity field because $\nabla\times\mathbf{q} = \nabla\times\mathbf{u}$. Moreover, for simplicity, we may further assume that $\mathbf{q} = \mathbf{0}$. This is the natural choice for the porous medium equation, which is derived under the assumption of a curl-free flow velocity field. An example of the method applied with a rotational velocity field is presented in [15]. The problem for determining the velocity potential is therefore: find $\phi \in H^1(\Omega_t)$ such that

$$\int_{\Omega_t} \rho\nabla v\cdot\nabla\phi \; d\mathbf{x} = \int_{\Omega_t} v\mathcal{L}\rho \; d\mathbf{x} \; + \; \int_{\partial\Omega_t} v\rho\mathbf{u}\cdot\mathbf{n} \; ds \; - \; c(v)\dot{\theta}_t \qquad \forall v \in H^1(\Omega_t), \tag{3.13}$$

where $\phi = 0$ is specified at one point in Ω_t to ensure uniqueness.

The velocity field is finally obtained as the solution of equation (3.12) written in weak form with $\mathbf{q} = \mathbf{0}$ and ϕ given by (3.13). That is, we find $\mathbf{u} \in [H^1(\Omega_t)]^2$ such that

$$\int_{\Omega_t} z\mathbf{u} \; d\mathbf{x} = \int_{\Omega_t} z\nabla\phi \; d\mathbf{x} \qquad \forall\, z \in H^1(\Omega_t), \tag{3.14}$$

with $\mathbf{u}\cdot\mathbf{n}$ imposed on any part of the boundary where it is known. For instance, in case of contact with an obstacle, we would impose $\mathbf{u}\cdot\mathbf{n} = 0$ on the contact boundary. Note that (3.14) is just the component-wise L^2-projection.

In the case of the PME, under the assumption that $\dot{\theta}_t = 0$, Equation (3.13) can be simplified to: find $\phi \in H^1(\Omega_t)$ such that

$$A_t(\phi, v) = d_t(v) \qquad \forall v \in H^1(\Omega_t),$$

where

$$A_t(\phi, v) = \int_{\Omega_t} \rho\nabla\phi\cdot\nabla v \; d\mathbf{x}, \tag{3.15}$$

$$d_t(v) = -\int_{\Omega_t} \rho^m\nabla\rho\cdot\nabla v \; d\mathbf{x}. \tag{3.16}$$

Remark 4. *We note the continuity of the velocity potential problem is guaranteed by the conditions on the PME solution of* $\rho_t \in L^\infty(\Omega_t) \cap H^1(\Omega_t)$ *for all* $t \in [0, T]$ *[29]. Coercivity of* $A_t(\cdot, \cdot)$ *is provided under the condition that* $\rho_t \geq 0$ *on* Ω_t *and that the integral of* ρ_t *is uniformly bounded below for all* $t \in [0, t]$. *Proofs of well-posedness are given for the PME in [97].*

Similarly, the velocity reconstruction Equation (3.14) can be written as: find $\mathbf{u} \in [H^1(\Omega_t)]^2$ such that

$$M_t(\mathbf{u}, v) = b_t(v) \qquad \forall v \in H^1(\Omega_t),$$

where

$$M_t(\mathbf{u}, v) = \int_{\Omega_t} \mathbf{u} v \; d\mathbf{x}, \tag{3.17}$$

$$b_t(v) = \int_{\Omega_t} \nabla\phi \; v \; d\mathbf{x}. \tag{3.18}$$

The velocity field in the interior of Ω_t could be discarded at this stage, and replaced by one derived using a different approach which is constrained by the boundary velocity derived above. Any of the ALE velocity schemes presented in Section 2.2 would be appropriate. Here we introduce the ALE velocity field given through a Laplacian smoothing of the velocity field. From Equations (2.13) and (2.14), the velocity field is defined by

$$\begin{aligned} \Delta \mathbf{w} &= 0 \qquad \mathbf{x} \in \Omega_t, \\ \mathbf{w} &= \mathbf{u} \qquad \mathbf{x} \in \partial\Omega_t. \end{aligned}$$

For completeness, we define the bilinear form for this Poisson's Equation by

$$K_t(\mathbf{w}, v) = \int_{\Omega_t} \nabla \mathbf{w} \cdot \nabla v \; d\mathbf{x} \qquad \forall v \in H_0^1(\Omega_t). \tag{3.19}$$

The Solution

The ALE velocity field $\mathbf{w}$ derived using Equations (3.13), (3.14) and (3.19) can now be used in the update of the solution. Integration of Equation (3.8) by

parts and substitution of Equation (3.3) results in

$$\dot{\mu}_t(v) = \int_{\Omega_t} v\mathcal{L}\rho \; d\mathbf{x} \; - \; \int_{\Omega_t} \rho\nabla v \cdot \mathbf{w} \; d\mathbf{x} \; + \; \int_{\partial\Omega_t} v\rho\mathbf{u}\cdot\mathbf{n} \; ds \qquad \forall\, v \in H^1(\Omega_t). \tag{3.20}$$

This is a standard conservative ALE update reviewed in Section 2.2. Inserting the PME into Equation (3.20) leads to

$$\dot{\mu}_t(v) = -\int_{\Omega_t} \rho^m\nabla\rho\cdot\nabla v \; d\mathbf{x} \; - \; \int_{\Omega_t} \rho\nabla v\cdot\mathbf{w} \; d\mathbf{x} \qquad \forall\, v \in H^1(\Omega_t).$$

Along with the condition $\dot{\mathbf{x}} = \mathbf{w}$, it gives a system of ODEs governing the evolution of the coordinate system and the distribution of the mass monitor which can be approximated using standard solvers such as explicit Runge-Kutta methods [15, 67]. With these at hand, the solution can be recovered by solving the problem: find $\rho \in H^1(\Omega_t)$ such that

$$m_t(\rho, v) := \int_{\Omega_t} \rho v \; d\mathbf{x} = \mu_t(v) \qquad \forall\, v \in H^1(\Omega_t). \tag{3.21}$$

In the special case where $\dot{\theta} = 0$ and $c(v)$ is assumed constant in time, Equation (3.20) is redundant because $\dot{\mu}_t(v) \approx 0$ by design. In fact, *without* the velocity recovery from the potential through (3.13) and (3.14), we would have $\dot{\mu}_t(v) \equiv 0$. However, the practical steps of the recovery procedure may introduce small perturbations. Alternatively, one may use the knowledge that $\dot{\mu}_t(v) \equiv 0$ and hence resort directly to the initial values $\mu_0(v)$, avoiding the need to calculate the ALE update (3.20) altogether: we refer to this as direct recovery. However, direct recovery can only be used with the specific choice of the mass monitor and with mass-conservative PDEs. In other situations the interior velocity field will not correspond to $\dot{\mu}_t(v) = 0$ and the ALE update is essential.

Remark 5. *The alternative, non-conservative, ALE formulation discussed in Section 2.2 can be derived which would give an equation for the material derivative $\mathcal{D}_t\rho$ instead of $\dot{\mu}$ (see [81]). This derivation does not require that the weight functions evolve with zero material derivative, but conservation of mass is lost as a result.*

3.3 A Moving Virtual Element Space

Similar to the original method, we consider a semi-discretisation approach for the moving mesh method VEM. In the following we let $0 = t_0 < t_1 < \dots t_{N_t} = T$ denote a given sequence of discrete time levels at which the PME is discretised.

We consider the set of moving mesh vertices defined at a time level t_n as $\{(\mathbf{x}_{h,n})_i\}_{i=1}^{N^{\text{dofs}}} \subset \mathbb{R}^2$. The time-dependent mesh $\mathcal{T}_{h,n}$ is defined to be the polygonal mesh given by transforming the initial mesh vertices according to the nodal displacements given by $\mathbf{x}_{h,n}$. A polygonal element of $\mathcal{T}_{h,n}$ is denoted by $E_{h,n}$.

The mesh connectivity is maintained for all time levels. Assumption 1 can be imposed on the initial domain but there is no guarantee this regularity will be maintained, especially if the mesh velocity approximates a Lagrangian velocity field [50]. The conditions of Assumption 1 can be maintained by introducing an ALE velocity field on the interior of $\Omega_{h,n}$ such as the Laplacian smoothing defined in Equation (3.19). Other choices of ALE velocity include the examples given in Section 2.2. Alternatively, we could formulate this method using a different monitor function [81].

The classical linear VEM space constructed on $\mathcal{T}_{h,n}$ is using Equations (2.3) and (2.4)

$$V_{h,n} = \left\{ v_h \in H^1(\Omega_{h,n}) \ : \ v_h \in V_1(E_{h,n}) \ \forall E_{h,n} \in \mathcal{T}_{h,n} \right\}. \tag{3.22}$$

When required, homogeneous Dirichlet boundary conditions can be embedded in the virtual element space by fixing the relevant boundary nodes. Hence, given a $\Gamma \subseteq \partial\Omega_{h,n}$, we denote the constrained space by

$$V_{h,n}^{\Gamma} = \{ v_h \in V_{h,n} \ : \ v_h|_{\Gamma} = 0 \}.$$

For ease of reading the second subscript will be omitted on the understanding that the VEM space of Equation (3.22) is only constructed on discrete time levels. At a fixed time, the moving mesh method considers approximations of the solution $\rho_h \in V_h$ and Lagrangian velocity $\mathbf{u}_h \in [V_h]^2$.

Remark 6. *As this moving mesh method is only considered for the lowest order discretisation, we only need to consider the vertex coordinates in the moving mesh at discrete time levels. In general the mapping from $\Omega_{h,0}$ to $\Omega_{h,n}$ would be represented by a discrete VEM function $\mathscr{A}_{h,n}(\boldsymbol{\xi})$ of degree $k \in \mathbb{N}$. This generalisation is explored in Part II for time-independent transformations with Chapter 6 being a direct extension of this idea to higher-order methods.*

3.4 A Virtual Element Method for the Velocity Field

We present the VEM for the solution of the velocity Equations (3.13) and (3.14).

It is assumed that a discrete solution at time level t_n has been computed as $\rho_h \in V_{h,n}$ on the current mesh $\mathcal{T}_{h,n}$. When $n = 0$, ρ_h is defined as the virtual element interpolation of the initial condition of the PME Problem 3.1.1. Otherwise, ρ_h is the current time discrete solution.

The VEM discretisation of the velocity potential Equation (3.13) reads: given $\rho_h \in V_{h,n}$, find $\phi_h \in V_{h,n}$ such that

$$A_{h,n}(\phi_h, v_h) = d_{h,n}(v_h) \qquad \forall\, v_h \in V_{h,n} \tag{3.23}$$

with the approximate bilinear form $A_{h,n}(\cdot,\cdot)$ and linear form $d_{h,n}(\cdot)$ built by summing element-wise contributions

$$A_{h,n}(\phi_h, v_h) = \sum_{E_h \in \mathcal{T}_{h,n}} A_{h,n}^{E_h}(\phi_h, v_h) \qquad \text{and} \qquad d_{h,n}(v_h) = \sum_{E_h \in \mathcal{T}_{h,n}} d_{h,n}^{E_h}(v_h).$$

As anticipated above however, the preceding definition of the local VEM forms necessitates the use of projections as follows:

$$\begin{aligned} A_{h,n}^{E_h}(\phi_h, v_h) = &\int_{E_h} \bar{\rho}_h \; \Pi_0^1 \phi_h \cdot \Pi_0^1 v_h \; d\mathbf{x} \\ &+ \bar{\rho}_h S^{E_h}(\phi_h - \Pi_1^\nabla \phi_h, v_h - \Pi_1^\nabla v_h), \end{aligned} \tag{3.24}$$

$$d_{h,n}^{E_h}(v_h) = -\int_{E_h} (\bar{\rho}_h)^m \; \Pi_0^1 \rho_h \cdot \Pi_0^1 v_h \; d\mathbf{x}, \tag{3.25}$$

where $\bar{\rho}_h$ is the nodal average of ρ_h computed via

$$\bar{\rho}_h = \frac{1}{N^{\text{dofs}}} \sum_{i=1}^{N^{\text{dofs}}} \text{dof}_i(\rho_h).$$

Following standard VEMs [18], we employ the dofi-dofi stabilisation form

$$S^{E_h}(w_h, v_h) = \sum_{i=1}^{N^{\text{dofs}}} \text{dof}_i(w_h) \cdot \text{dof}_i(v_h) \qquad \forall w_h, v_h \in V_1(E_h). \tag{3.26}$$

The integration constant is fixed by constraining a single vertex value of ϕ_h to zero. A variety of suitable stabilisation choices are admissible [20, 39] but we adopt the simplest choice in this chapter. In particular, in the case when arbitrarily small edges appear in the mesh we refer to [30] for more appropriate stabilisation terms.

The velocity reconstruction Equation (3.14) is a global L^2 projection, which has a well documented VEM discretisation [2, 44]. Its VEM discretisation reads: given $\phi_h \in V_{h,n}$, the solution of (3.23), find $\mathbf{u}_h \in [V_{h,n}]^2$ such that $\mathbf{u}_h \cdot \mathbf{n} = 0$ on the portion of $\Omega_{h,n}$ approximating $\partial\Omega_n^F$ and

$$M_{h,n}(\mathbf{u}_h, v_h) = b_{h,n}(v_h) \qquad \forall\, v_h \in V_{h,n}.$$

As before, the forms $M_{h,n}(\cdot,\cdot)$ and $b_{h,n}(\cdot,\cdot)$ are obtained summing the respective elemental forms

$$\begin{aligned} M_{h,n}^{E_h}(\mathbf{u}_h, v_h) &= \int_{E_h} \Pi_1^0 \mathbf{u}_h \; \Pi_1^0 v_h \; d\mathbf{x} \\ &\quad + \; |E_h| \, S^{E_h}(\mathbf{u}_h - \Pi_1^0 \mathbf{u}_h, v_h - \Pi_1^0 v_h), \end{aligned} \tag{3.27}$$

$$b_{h,n}^{E_h}(v_h) = \int_{E_h} \Pi_1^0 v_h \; \Pi_0^1 \phi_h \; d\mathbf{x}, \tag{3.28}$$

with the stabilisation term given by Equation (3.26). We remark that the computation of Equations (3.27) and (3.28) are done component-wise using the standard VEM.

In a VEM framework the interior ALE velocity mesh velocity is computed by solving a standard Poisson's equation with Dirichlet boundary conditions taken as the approximation of the Lagrangian velocity field [44]. We denote

this ALE velocity by $\mathbf{w}_h \in [V_{h,n}]^2$ and define the discrete approximation of Equation (3.19) by

$$K_{h,n}^{E_h}(\mathbf{w}_h, v_h) = \int_{E_h} \Pi_0^1 \mathbf{w}_h \cdot \Pi_0^1 v_h \, d\mathbf{x} + S^{E_h}(\mathbf{w}_h - \Pi_1^\nabla \mathbf{w}_h, v_h - \Pi_1^\nabla v_h), \quad (3.29)$$

where the Dirichlet boundary condition of $\mathbf{w}_h = \mathbf{u}_h$ is imposed. We present the remaining formulation using this velocity field.

The mesh is transferred between discrete time levels by displacing the nodes of the mesh and maintaining the mesh connectivity between $t = t_n$ and $t = t_{n+1}$. For the set of mesh nodes $\{(\mathbf{x}_{h,n})_i\}_{i=1}^{N^{\text{dofs}}}$, the new positions are obtained by the forward Euler method applied to $\dot{\mathbf{x}}_{h,n} = \mathbf{w}_{h,n}$, yielding

$$\mathbf{x}_{h,n+1} = \mathbf{x}_{h,n} + (t_{n+1} - t_n)\mathbf{w}_{h,n}.$$

Thus, consistent with the VEM philosophy, only the values of $\mathbf{w}_h$ at the nodes, that is the degrees of freedom of $\mathbf{w}_h$, are required to compute the mesh movement.

Remark 7. *We note that the mesh velocities computed in this way do not guarantee a priori that Assumption 1 holds true. For instance, when performing the numerical experiments for contact problems of Section 4.2, we observed a degradation of mesh quality near the contact boundary. The Laplacian smoothing improves the robustness of this method but not indefinitely. This is also observed in FEM ALE schemes [89].*

3.5 A Virtual Element Method for the Solution

Once the new mesh node positions have been computed, we consider the process of updating the solution. This is performed in two steps corresponding, respectively, to the conservative ALE update of the mass monitor Equation (3.20) and the actual solution update Equation (3.21).

The original moving mesh method in [15] was based on the linear FEM for which the validity at the discrete level of the material derivatives assump-

tion (3.5) leading to the ALE update (3.8) has been proven in [68]. In the VEM setting, instead, we exploit the fact that virtual element functions are *only accessed through their nodal values*: in close alignment with the work of Lipnikov & Morgan reviewed in Section 2.3, only the mesh skeleton velocity is known and used. Hence, in view of the solution update through the time step $[t_n, t_{n+1}]$, we can assume that (3.5) is satisfied by the space-time discrete basis functions, which are then interpolated at the new time level, again, just by accessing the nodal values of the solution.

The initial condition $\rho_{h,0}$ is approximated by interpolating the degrees of freedom of ρ_0 into the VEM space $V_{h,0}$. Then, the initial mass monitor distribution is computed via

$$\mu_{h,0}(v_h) = \sum_{E_h \in \mathcal{T}_{h,0}} \int_{E_h} \Pi_1^0 \rho_{h,0} \; \Pi_1^0 v_{h,0} \; d\mathbf{x}. \tag{3.30}$$

The next task is the update of the mass monitor over time levels. Once again, this is performed via the forward Euler method: the new monitor is thus given by

$$\mu_{h,n+1}(v_h) = \mu_{h,n}(v_h) + (t_{n+1} - t_n)\dot{\mu}_{h,n}(v_h).$$

This, in turn, requires the approximation of the ALE Equation (3.20) which is performed still on the old time level (temporal subscript omitted) by

$$\dot{\mu}_h(v_h) = -\sum_{E_h \in \mathcal{T}_{h,n}} \int_{E_h} \Pi_0^1 \rho_h \; \Pi_0^1 v_h \cdot \left\{ (\bar{\rho}_h)^{m-1} \Pi_0^1 \rho_h + \Pi_1^0 \mathbf{w}_h \right\} \; d\mathbf{x}, \tag{3.31}$$

where again $\bar{\rho}_h$ denotes the nodal average of ρ_h on E_h. Once the monitor is updated, we set the time level to the new time and update the solution using a VEM discretisation of Equation (3.21): find $\rho_{h,n+1} \in V_{h,n+1}$ such that

$$m_{h,n+1}(\rho_h, v_h) = \mu_{h,n+1}(v_h) \qquad \forall\, v_{h,n+1} \in V_{h,n+1}.$$

Similarly to the velocity reconstruction, the discrete form $m_{h,n+1}(\cdot,\cdot)$ is computed on the new time level by summing over element contributions

$$m_{h,n+1}(\rho_h, v_h) = \sum_{E_h \in \mathcal{T}_{h,n+1}} m_{h,n+1}^{E_h}(\rho_h, v_h),$$

where

$$m_{h,n+1}^{E_h}(\rho_h, v_h) = \int_{E_h} \Pi_1^0 \rho_h \; \Pi_1^0 v_h \; d\mathbf{x} \; + |E_h| \, S^{E_h}(\rho_h - \Pi_0^1 \rho_h, v_h - \Pi_0^1 v_h), \quad (3.32)$$

with the stabilisation term given by Equation (3.26).

3.6 A Conservation Law

A beneficial property of the linear virtual element method is that the basis functions form a partition of unity on $\Omega_{h,n}$ at any discrete time level, i.e.

$$\sum_{i=1}^{N^{\text{dofs}}} \varphi_i = 1, \quad (3.33)$$

where the set $\{\varphi_i\}_{i=1}^{N^{\text{dofs}}}$ refers to the set of canonical VEM basis functions associated to the vertices of the mesh (see Section 2.1); that is $\varphi_i((\mathbf{x}_{h,n})_j) = \delta_{ij}$ for $i = 1, ..., N^{\text{dofs}}$ where $(\mathbf{x}_{h,n})_j$ is the j-th node in the mesh.

For a given $\rho_h \in V_{h,n}$ the monitor integral θ_n reads

$$\theta_{h,n} = \int_{\Omega_{h,n}} \rho_h \; d\mathbf{x},$$

from which the polynomial consistency and partition of unity property of the VEM gives

$$\begin{aligned}
\theta_{h,n} &= \sum_{E_h \in \mathcal{T}_{h,n}} \int_{E_h} \Pi_1^0 \rho_h \; d\mathbf{x} \\
&= \sum_{E_h \in \mathcal{T}_{h,n}} \int_{E_h} \Pi_1^0 \rho_h \sum_{i=1}^{N^{\text{dofs}}} \varphi_i \; d\mathbf{x} \\
&= \sum_{i=1}^{N^{\text{dofs}}} \sum_{E_h \in \mathcal{T}_{h,n}} \int_{E_h} \Pi_1^0 \rho_h \; \Pi_1^0 \varphi_i \; d\mathbf{x}.
\end{aligned}$$

Therefore the global conservation of the mass monitor is only dependent on the polynomial component of the discrete solution and weighting functions. Further, the exact value of the discrete monitor integral can also be recovered via

$$\theta_{h,n} = \sum_{i=1}^{N^{\text{dofs}}} \mu_{h,n}(\varphi_i).$$

Following a similar argument we obtain using Equation (3.31) that

$$\dot{\theta}_{h,n} = \sum_{i=1}^{N^{\text{dofs}}} \dot{\mu}_{h,n}(\varphi_i) = 0,$$

which implies that the VEM conserves the global mass of the solution at every time level and agrees with the conservation of mass principle for this particular PDE. In fact, virtual elements preserving relevant global conservation laws in different contexts can be constructed, an example of which is given in [96] for the heat equation.

3.7 Implementation Details

This section presents a complete overview of the moving mesh virtual element method. The construction of the required algebraic equations and imposition of boundary conditions are reviewed along with some practical remarks regarding the implementation of this method. The initial weak distribution of the monitor is stored in the vector $\boldsymbol{\mu}_0$ and can be computed using equation (3.30) whilst the mass matrix $\mathbf{M}_n$ is computed by assembling the contributions from equation (3.32).

In order to compute the discrete potential $\phi_h \in V_{h,n}$ from equation (3.23), we solve the linear system $\mathbf{A}_n\boldsymbol{\phi} = \mathbf{d}_n$ with $\mathbf{A}_n$ and $\mathbf{d}_n$ computed using equations (3.24) and (3.25), respectively. The solution of the resulting linear system determines ϕ_h up to an additive constant which is inherited from the continuous formulation of the method; here we impose $\text{dof}_1(\phi_h) = 0$.

Once ϕ_h is recovered, the Lagrangian velocity field is reconstructed solving $\mathbf{M}_n^R\mathbf{u} = \mathbf{b}_n$, with $\mathbf{M}_n^R$ and $\mathbf{b}_n$ given by equations (3.27) and (3.28), respectively.

The discrete ALE velocity is computed by solving $\mathbf{K}_n\mathbf{w} = \mathbf{0}$ using Equation (3.29).

The ALE update of vector $\dot{\boldsymbol{\mu}}_n$ is then obtained using equation (3.31) and, along with the mesh nodal velocities $\dot{\mathbf{x}}_{h,n} = \mathbf{w}_{h,n}$, provides a system of ODEs

which can be approximated using the forward Euler method. The main method is outlined in algorithm 1.

Algorithm 1: Moving mesh VEM

input : The initial condition $\rho_{h,0} \in V_{h,0}$ and mesh $\mathcal{T}_{h,0}$, the final time T.

Set $n = 0$;

Compute $\boldsymbol{\mu}_0$ according to equation (3.30);

while $t_n < T$ **do**

- Construct and solve $\mathbf{A}_n\boldsymbol{\phi} = \mathbf{d}_n$ using equations (3.24) and (3.25) for $\phi_h \in V_{h,n}$;
- Reconstruct the velocity via $\mathbf{M}_n^R\mathbf{u} = \mathbf{b}_n$;
- Compute the ALE velocity via $\mathbf{K}_n\mathbf{w}$;
- Compute the ALE update $\dot{\boldsymbol{\mu}}_n$ from equation (3.31);
- Select Δt and set $t_{n+1} = t_n + \Delta t$;
- Update the mesh node by $\mathbf{x}_{h,n+1} = \mathbf{x}_{h,n} + \Delta t\mathbf{w}$;
- Update the monitor distribution by $\boldsymbol{\mu}_{n+1} = \boldsymbol{\mu}_n + \Delta t\dot{\boldsymbol{\mu}}_n$;
- Reconstruct and solve $\mathbf{M}_{n+1}\boldsymbol{\rho}_{n+1} = \boldsymbol{\mu}_{n+1}$ for $\rho_{h,n+1} \in W_{h,n+1}$;
- Update $n = n + 1$;

end

output: The final solution $\rho_{h,T} \in V_{h,T}$, the final mesh $\mathcal{T}_{h,T}$

.

If required, we can strongly impose any Dirichlet boundary conditions on the solution over time whilst simultaneously conserving the monitor integral, the test space used in recovering the solution is augmented to preserve the partition of unity property (3.33). In the solution updates, Dirichlet boundary conditions can be enforced using the methodology presented in [67] which extends to a near identical virtual element approach when considering polygonal elements. This is not a necessary step to attain the accuracy results given in Chapter 4 and so the implementation details are not discussed in this thesis.

Chapter 4

Numerical Investigations

Numerical experiments are presented in this chapter for the moving mesh VEM. The method is tested against known similarity solutions of the PME in line with the preliminary tests conducted for the FEM discretisation of this method [14, 15]. The numerical errors are assessed for both the solution DoFs and coordinates of the moving boundary. To exploit the generality in mesh structure granted by the VEM, we implement a contact algorithm to handle local changes in mesh topology without requiring a global remeshing or significant changes to the method's structure.

In Section 4.1, the PME is used as a test PDE on a set of varied polygonal mesh structures. The contact algorithm is presented and tested in Section 4.2. The moving mesh VEM for a fourth-order diffusion problem is presented in Section 4.3.

4.1 Convergence for the Porous Medium Equation

In our first set of tests we consider a family of radially symmetric similarity solutions on a given initial circular domain of radius r_0 for the PME (Problem

3.1.1) derived in [86] and given by

$$\rho(r,t) = \begin{cases} \frac{1}{\lambda(t)^d}\left(1 - \left(\frac{r}{r_0\lambda(t)}\right)^2\right)^{\frac{1}{m}} & |r| \leq r_0\lambda(t) \\ 0 & \text{otherwise} \end{cases}, \tag{4.1}$$

where d is the spatial dimension, r_0 is the initial radius, and

$$\lambda(t) = \left(\frac{t}{t_0}\right)^{\frac{1}{2+dm}}, \quad t_0 = \frac{r_0^2 m}{2(2+dm)}.$$

Because of the nature of Equation (4.1), the solution is expected to have finite slope normal to the moving boundary for $m \leq 1$ whilst, for $m > 1$, the solution presents an infinite slope normal to the boundary. Further properties of this analytical solution are discussed in [87, 15].

Sample meshes

We have tested four different mesh types used to subdivide the initial domain; representative examples of each are shown in Figure 4.1. The first mesh is the Voronoi Tessellation produced by randomly sampling mesh seeds in the domain [91, 10]. The second mesh is a Centroidal Voronoi Tessellation (CVT) produced by the Lloyd algorithm which smooths a given Voronoi tessellation such that the generator points are the barycentric coordinates for each polygon [51]. The MATLAB package PolyMesher [94] was used to produce these two mesh types. The third mesh is constructed by overlaying the domain with a grid of uniform squares and cutting the mesh along the boundary. The last mesh type is a mixture of uniform Cartesian and polar tessellations. Note that the first three initial mesh types may present arbitrarily small edges and, moreover, arbitrarily small elements may appear near the boundary in the grid mesh type, as such potentially contradicting the mesh regularity Assumption 1. In this respect, we note that the VEM is known to be quite robust, as we have also witnessed. We refer to the mesh size in each case as the largest element diameter in $\mathcal{T}_{h,0}$.

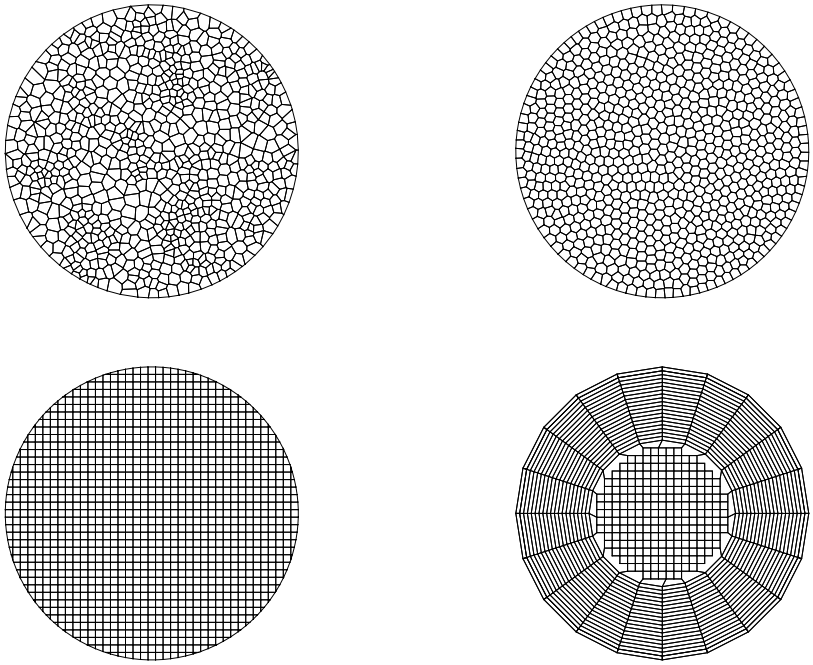

Figure 4.1: Examples of each of the four mesh types used in numerical tests for a circular domain: the Voronoi Tessellation (top left), the CVT (top right), the grid mesh (bottom left) and the mixed mesh (bottom right).

Error computation

The numerical error is computed for both the solution and mesh by generalising the discrete approximations from [15]. An l^1 solution error is given by,

$$\|\rho_n - \rho_{h,n}\|_{sol} = \frac{1}{N^{\text{dofs}}} \sum_{i=1}^{N^{\text{dofs}}} \left|\text{dof}_i(\rho_n) - \text{dof}_i(\rho_{h,n})\right|, \tag{4.2}$$

while, for the mesh error, an l^1 norm is considered for the radial distance $r(t)$ from the boundary of the mesh to the origin; thus

$$\|r_n - r_{h,n}\|_{mesh} = \frac{1}{N^B} \sum_{i=1}^{N^B} \left|R_{i,n} - r_0\lambda(t_n)\right|. \tag{4.3}$$

Here, N^B denotes the number of boundary nodes in the mesh and $R_{i,n}$ denotes the radial distance from the origin of the i-th boundary node at time t_n. A uniform Δt that is small enough to ensure numerical stability is set for each initial sample mesh. Note that the meshes used in these numerical tests are not hierarchical. Furthermore, each Voronoi mesh is generated independently from

randomly generated seeds. For each reduction of initial mesh size by a factor of 2, the time-step Δt is reduced by a factor of 4 to ensure numerical stability. By reducing the time-step size by this factor we also expect that the temporal error to be $O(h^2)$ when using the Forward Euler method for the refinement path of the four mesh types.

Convergence Tests

In this first convergence test the solution of the similarity solution for $m = 1, d = 2$, and $r_0 = 0.5$ is compared against the numerical solution for $t = t_0 + T$. The method is tested on each of the four mesh types for a circular domain. The time step sizes for the coarsest meshes are chosen according to

$$\Delta t = \frac{1}{250} h_{mean}^2,$$

where h_{mean} is the average element diameter of the initial mesh $\mathcal{T}_{h,0}$. In each mesh case we observe that the initial time-step size is approximately 10^{-4}. From the coarse-mesh time-step sizes we reduce Δt by a factor of 4 each time the mesh is refined, which corresponds to the mesh size approximately halving with each refinement. In choosing the time step sizes we made conservative choices such that the numerical method was stable and presented the expected orders of convergence. A more robust approach would be to use adaptive time-stepping schemes but in this work we present convergence results for a uniform reduction in the time-step.

A selection of solution snapshots are given in Figure 4.2 using the CVT-type mesh.

As shown in Figure 4.3, second order accuracy is observed for the solution error for all mesh cases when $T = 0.01$. In the case of the Voronoi and Grid meshes, the empirical order of convergence (EOC) is less smooth compared to the CVT and mixed mesh types. This is most likely due to the weaker shape regularity of elements in these mesh types, but further studies are required.

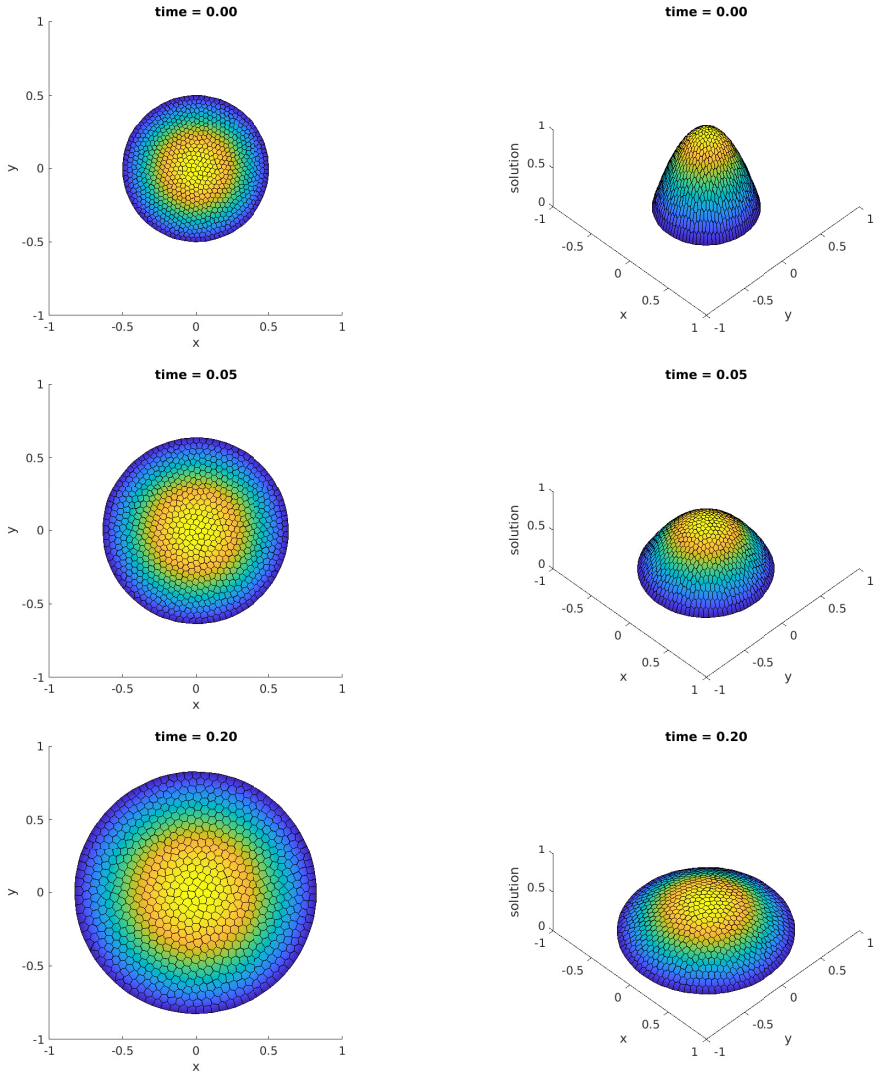

Figure 4.2: Snapshots of the moving mesh VEM solution for the PME with $m = 1$ at $t = 0$ (top), $t = 0.05$ (middle), $t = 0.2$ (bottom). A CVT-type mesh with 800 elements was used to initialise the mesh at $t = 0$.

The mesh error EOC appears to have a long pre-asymptotic regime: the EOC grows monotonically towards the expected rate in all cases with the final computed values ranging between 1.55 (Voronoi) and 1.83 (mixed). This is consistent with finite element approximations of Darcy flow which observed the order of convergence of the velocity field to be lower than that of the pressure field [32, 82].

Conservation of mass in the numerical solution is observed up to machine precision in all test cases.

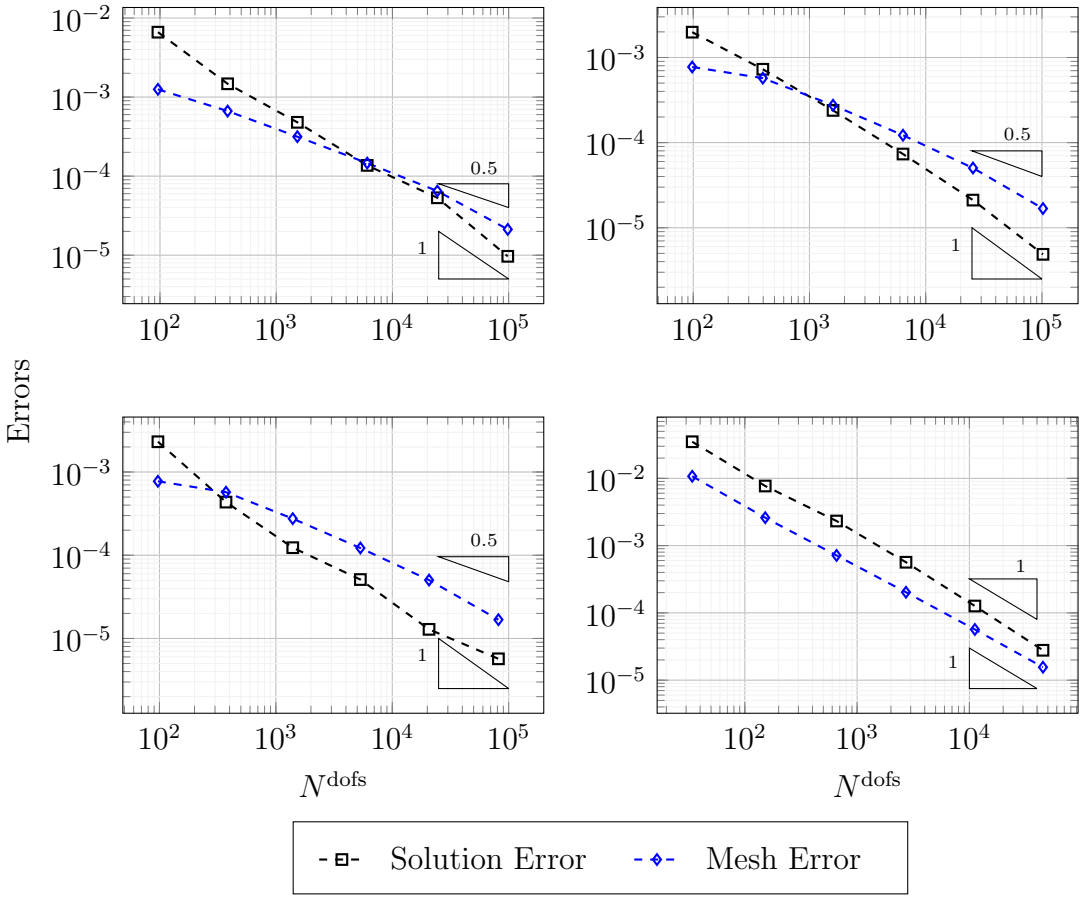

Figure 4.3: PME similarity solution with $m = 1$: the l^1 solution and mesh errors (4.2) and (4.3), respectively, at time $T = 0.01$ for each mesh type: Voronoi (top left), CVT (top right), Grid (bottom left), Mixed (bottom right).

Remark 8. *Setting* $\dot{\boldsymbol{\mu}} = \mathbf{0}$ *produces similar results to those reported in Figure 4.3. This is referred to as a "direct recovery" approach in the literature, see [15]. For brevity, the corresponding results are not presented here.*

4.2 A Contact Algorithm

In this section we discuss two occurrences of contact and present corresponding basic node insertion algorithms that allow for localised and minimal changes to the mesh structure. In all mesh refinements considered, the change in mesh topology is only performed at the discrete time-levels. Hence, to ease notation, the subscript used to denote time-steps is omitted. Modified discrete functions, operators, and vectors are denoted using a superscript of $\mathcal{R}$.

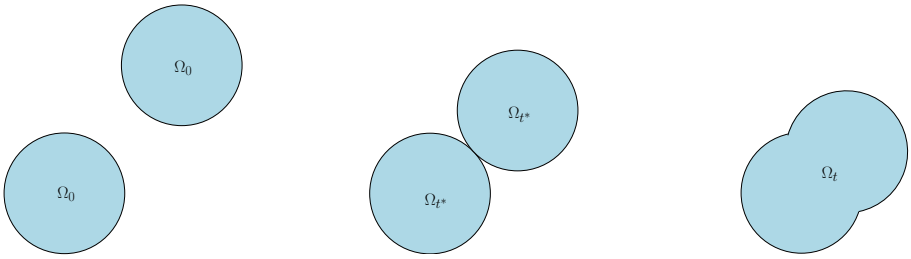

Figure 4.4: A demonstration of the mesh self-intersection problem. The initial condition has disconnected support in $\tilde{\Omega}$ (left). The disconnected $\partial\Omega_t$ continues to move until some time t^* where the boundary collides with itself (centre). A new connected boundary is formed over time and the topology of Ω_t is now connected.

Contact scenarios

Here we present two contact scenarios that are numerically investigated in this section. The first scenario concerns the collision of the moving boundary with itself whereas the second situation involves collision with fixed geometric obstacles.

Self-intersection handles the situation where two parts of the moving boundary collide with each other. Typically, a remeshing is required in this instance. By using a VEM, the remeshing can be kept local and simple for colliding elements. A motivational case for a disconnected initial condition of the PME is given in Figure 4.4. Moving mesh finite element simulations of this type of problem are presented and discussed in [87].

Obstacle contact is encountered when the evolution of Ω_t is obstructed by external obstacles. An example of this is the presence of a solid phase in porous media. Figure 4.5 presents an example of collision with impermeable obstacles. By using a collision detection and node insertion algorithm, the moving mesh is capable of simulating the contact and the interaction between the moving mesh and a set of obstacles. As with the self-intersection problem, the VEM allows for this with minimal changes to the mesh topology. Additionally, the VEM is capable of performing local changes to polygonal elements such that the mesh boundary can move around the object boundaries without requiring additional mesh refinements.

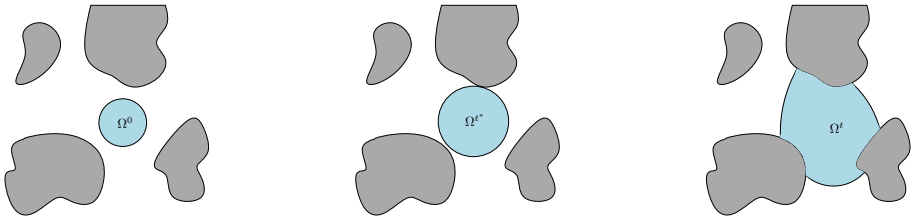

Figure 4.5: A demonstration of the obstacle contact problem. The initial condition is contained in a region of obstacles (left). After some time t^* the moving boundary collides with the first obstacle (centre). A time-dependent interface now forms between the obstacles and $\partial\Omega_t$ (right).

Collision detection

For detecting mesh contact we use an adaptation of the classical *node-to-segment collision detection algorithm* [64, 101, 102]. We only consider boundary mesh and obstacle edges and nodes, thus ensuring that the additional computational cost is $O(N_B^2)$, where N_B is the number of boundary nodes. We consider triplets of points $(\mathbf{x}_{1,t}, \mathbf{x}_{2,t}, \mathbf{x}_{3,t})$ where $\mathbf{x}_{1,t}$ and $\mathbf{x}_{2,t}$ form a time-dependent edge e_t and $\mathbf{x}_{3,t}$ is boundary node disconnected from e_t. This triplet is referred to as a node-to-edge pair.

Given a set of linear velocities $(\dot{\mathbf{x}}_1, \dot{\mathbf{x}}_2, \dot{\mathbf{x}}_3)$, by defining the vectors

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}, \quad \dot{\mathbf{x}} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix}, \quad \dot{\mathbf{y}} = \begin{bmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \dot{y}_3 \end{bmatrix},$$

the contact time Δt^* between the line of e_t and $\mathbf{x}_{3,t}$ is given as the minimum positive root of the quadratic equation

$$0 = a(\Delta t^*)^2 + b(\Delta t^*) + c, \tag{4.4}$$

where the coefficients are defined by

$$a = \sum_{i=1}^{3} (\dot{\mathbf{y}} \times \dot{\mathbf{x}})_i \qquad b = \sum_{i=1}^{3} (\dot{\mathbf{y}} \times \mathbf{x} + \mathbf{y} \times \dot{\mathbf{x}})_i \qquad c = \sum_{i=1}^{3} (\mathbf{y} \times \mathbf{x})_i .$$

In practise, we choose the contact time that makes physical sense (e.g. we discard negative roots as infeasible contact time) and only admit a single contact time value for Δt^*. In the case of no feasible contact times we set $\Delta t^* = \infty$ and when two feasible contact times are given by equation (4.4) we choose

the minimum of the two. By solving equation (4.4) for a set of node-to-edge pairings, a set of contact times can be computed. In the context of the moving mesh method, each node-to-edge pair on the boundary of the mesh and (when given) obstacle mesh is considered. If any cases indicate contact, the time step is scaled down to the minimum contact time and the corresponding node-to-edge pair is marked for contact. The detection method is outlined in Algorithm 2 for a single node-to-edge pairing.

Algorithm 2: Contact Detection

input : A node-to-edge pairing $(\mathbf{x}_{1,t}, \mathbf{x}_{2,t}, \mathbf{x}_{3,t})$, a set of nodal velocities $(\dot{\mathbf{x}}_{1,t}, \dot{\mathbf{x}}_{2,t}, \dot{\mathbf{x}}_{3,t})$, the current time step Δt

Solve equation (4.4) and set Δt^* to the minimum positive root;

for *each value of* $\Delta t^* \in \mathbb{R}$ **do**

  Compute $\mathbf{x}_{3,t+\Delta t^*}$ and $e_{t+\Delta t^*}$;

  if $\Delta t^* \in [0, \Delta t]$ *and* $\mathbf{x}_{3,t+\Delta t^*} \in e_{t+\Delta t^*}$ **then**

    Mark the node-to-edge pair for contact;

  else

    Set the contact pair to no contact;

  end

end

output: the node-to-edge contact pair, the contact time step Δt^*

Node Insertion algorithm

Since the mesh allows for general polygonal element shapes, the insertion of a new node into a mesh edge can simply be performed by adding a vertex to the polygons sharing that edge. Then, a solution value associated with the new vertex must be introduced which requires an interpolation technique between the old and new global discrete spaces. In contrast to the approaches reviewed in Section 2.3, we choose to preserve the polynomial component of the solution $\Pi_0^1 \rho_h$ between refinements through a redistribution of $\boldsymbol{\mu}_n$. The reason for this choice is that, by preserving the polynomial component of ρ_h, the global mass conservation of $\theta_{h,n}$ is maintained. This would *not* be the case if interpolation of the degrees of freedom was employed instead.

On a given element $E_h \in \mathcal{T}_{h,n}$ with an inserted node on the boundary, we

impose that the polynomial component of the solution is preserved so that the reconstruction $\rho_h^{\mathcal{R}} \in V_1^{\mathcal{R}}(E_h)$ satisfies

$$\Pi_1^0 \rho_h^{\mathcal{R}} = \Pi_1^0 \rho_h, \tag{4.5}$$

where $V_1^{\mathcal{R}}(E_h)$ denotes refined VEM space.

The arguments in Section 3.6 can be easily modified to show that this approach conserves both locally and globally the mass of the discrete solution. When introducing a new node onto a mesh edge of a given element E under the assumption of Equation (4.5), the local contribution to $\boldsymbol{\mu}^{\mathcal{R}}$ is given by,

$$\mu_{h,n}^{\mathcal{R},E_h}(\varphi_i^{\mathcal{R}}) = \int_{E_h} \Pi_1^0 \rho_h \;\; \Pi_1^0 \varphi_i^{\mathcal{R}} \; d\mathbf{x} \quad i = 1, ..., N^{\text{dofs}} + 1. \tag{4.6}$$

The algorithm for node insertion is given in Algorithm 3.

Algorithm 3: Node Insertion

input : An element E_h, a position $\mathbf{x} \in \partial E_h$ to insert a node, the monitor distribution $\boldsymbol{\mu}$, the solution $\rho_h \in V_{h,n}$, the mesh $\mathcal{T}_{h,n}$
Compute $\Pi_1^0 \rho_h$ on E_h;
Compute the new mesh $\mathcal{T}_{h,n}^{\mathcal{R}}$ by inserting the node;
Compute the new monitor distribution $\boldsymbol{\mu}^{\mathcal{R}}$ using equation (4.6);
Reconstruct the new solution $\rho_h^{\mathcal{R}} \in V_{h,n}^{\mathcal{R}}$ by solving $\mathbf{M}^{\mathcal{R}} \boldsymbol{\rho}^{\mathcal{R}} = \boldsymbol{\mu}^{\mathcal{R}}$;
output: the new solution $\rho_h^{\mathcal{R}}$, the new monitor distribution $\boldsymbol{\mu}^{\mathcal{R}}$, the new mesh $\mathcal{T}_{h,n}^{\mathcal{R}}$

Self-intersection algorithm

Due once more to the VEM flexibility in element geometries, the self-intersection problem does not require any introduction of additional degrees of freedom. Instead, the local connectivity of the disconnected mesh is updated to include the new node-to-edge pairing. Then, the node insertion algorithm is applied to recompute the solution and monitor distribution. As the boundary node velocities are not arbitrarily set, small edges are likely to appear during node insertion. This has not presented any stability issues within the numerical experiments presented later in this section and we expect that the method remains robust in the presence of degenerate edges [30]. The subsequent mesh

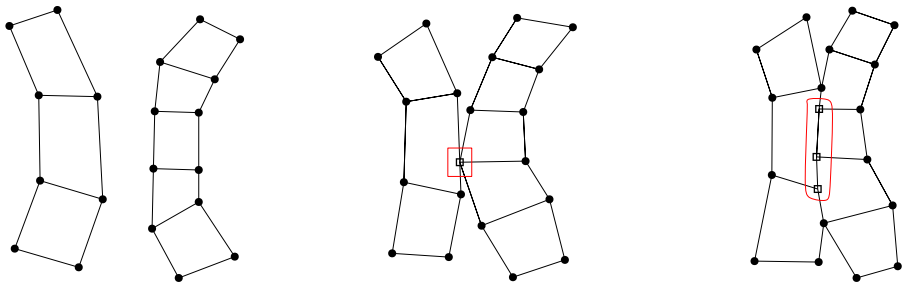

Figure 4.6: A demonstration of the self intersection algorithm. A sample of elements is shown (left) where a collision is expected. Algorithm 4 is applied with contact nodes marked by a square (centre). Following subsequent mesh updates certain degrees of freedom no longer lie on the boundary (right). The highlighted degrees of freedom are now treated as internal degrees of freedom.

velocity problem is then solved based on the updated mesh and corresponding virtual element space. A simple demonstration is provided in Figure 4.6; the method at a given time step t^n is presented in Algorithm 4.

Algorithm 4: Self-intersection

input : A mesh $\mathcal{T}_{h,n}$, a solution $\rho_{h,n}$, a velocity field $\mathbf{w}$, a time step size Δt.
Apply Algorithm 2 for boundary node-to-edge pairs in $\mathcal{T}_{h,n}$;
Update Δt;
Compute $\mathcal{T}_{h,n+1}$, $\boldsymbol{\mu}_{n+1}$, $\rho_{h,n+1}$;
if *any contact pairs are marked* **then**
  find the element $E_h \in \mathcal{T}_{h,n+1}$ which contains the marked edge;
  Apply Algorithm 3;
end
Update the boundary conditions of $\rho_{h,n+1}$;

Obstacle contact and pivot node algorithm

We denote a time-independent polygonal discretisation of the set of obstacles by $\mathcal{O}_h$ and consider the mesh node-to-edge pairings of boundary nodes from $\mathcal{T}_{h,n}$ and edges of $\mathcal{O}_h$ and vice versa. In the case of obstacle-node to mesh-edge contact, the node insertion Algorithm 3 is applied to introduce an additional degree of freedom to the system; we refer to this new node, which is a fixed point on the obstacle geometry, as a "pivot node".

For contact between $\mathcal{T}_{h,n}$ and $\mathcal{O}_h$ a no-penetration condition on the nodal velocities is strongly imposed on the formulation of the potential problem

(see Equations (3.15) and (3.16)) and the velocity reconstruction problem (see Equations (3.17) and (3.18)); namely,

$$\mathbf{u} \cdot \mathbf{n} = \nabla\phi \cdot \mathbf{n} = 0. \tag{4.7}$$

Hence, movement tangential to the obstacle's boundary is allowed. This is except when a pivot node is introduced, in which case we constrain its velocity to zero to preserve the geometry of the interface between the domain and the obstacle. The obstacle contact algorithm is outlined in Algorithm 5.

Given that the pivot node mesh velocity is constrained to zero, it is possible for the other boundary nodes laying on the obstacle (which have experienced mesh-node to obstacle-edge contact) to pass through the pivot node. When this occurs, the connectivity of the mesh is updated to transfer the pivot node from one mesh edge to another as well as swapping the boundary node from one obstacle face to another.

Detection for pivot node collision is performed using Algorithm 2 for connected mesh boundary nodes moving from one obstacle edge to another.

A node is considered to be on $\partial\Omega_{h,n}^F$ only if both boundary edges sharing that node are in contact with the obstacle. We define a node to be "connected" if it lies on an edge of an obstacle. Degrees of freedom associated to connected nodes are constrained by equation (4.7) in the mesh velocity computations. If a node is connected by mesh edges to other connected boundary nodes we consider this to be an "interface" node and consequently change the boundary conditions from Dirichlet to Neumann defined in problem 3.1.1 (i.e. we change the degree of freedom from $\partial\Omega_{h,n}^M$ to $\partial\Omega_{h,n}^F$). If a connected node is not also an interface node, the homogeneous boundary condition and no-penetration condition are maintained. The structure of the boundary conditions are updated once every time step.

When a mesh node $\mathbf{x}_i$ and a pivot node coincide (while the mesh node is moving along the obstacle boundary) the test function associated to a pivot

Algorithm 5: Obstacle contact

input : A mesh $\mathcal{T}_{h,n}$, a solution $\rho_{h,n}$, a velocity field $\mathbf{u}$, a time step size Δt, an obstacle mesh $\mathcal{O}_h$.

Apply Algorithm 2 for boundary nodes in $\mathcal{T}_{h,n}$ and boundary edges in $\mathcal{O}_h$;

Apply Algorithm 2 for boundary nodes in $\mathcal{O}_h$ and boundary edges in $\mathcal{T}_{h,n}$;

Select the contact pair with the smallest Δt^* and set $\Delta t = \Delta t^*$;

Compute $\mathcal{T}_{h,n+1}$, $\boldsymbol{\mu}_{n+1}$, $\rho_{h,n+1}$;

if *obstacle node to mesh edge is marked* **then**

  find the element $E \in \mathcal{T}_{h,n+1}$ which contains the marked edge;

  Apply Algorithm 3 to introduce a pivot node $\mathbf{x}_{pivot}$;

  set $\mathbf{u} = \mathbf{0}$ at $\mathbf{x}_{pivot}$;

end

if *Mesh node to obstacle edge is marked* **then**

  Set $\mathbf{u} \cdot \mathbf{n} = 0$ at mesh node;

  Update Neumann conditions for $\rho_{h,n+1}$;

end

node is chosen to satisfy $\varphi_{pivot} \equiv \varphi_i$. In other words, we duplicate the original VEM basis function and add it to the new discrete space.

Node insertion convergence test

Our next numerical experiment considers the case of the one-dimensional PME extended in the x direction to a two-dimensional problem. This experiment has two interesting features. Firstly, the initial domain is geometrically exact ($\Omega_0 \equiv \Omega_{h,0}$) unlike the circular meshes. Secondly, it allows us to test the obstacle contact and node insertion algorithm numerically against a known analytical solution derived from the one-dimensional case of equation (4.1).

This is obtained by considering once again Equation (4.1) with the values $m = 1, d = 1, r_0 = 0.5$, and $r = y$ on the initial domain given by $\Omega^0 = [-0.5, 0.5]^2$ with initial condition

$$\rho(\mathbf{x}, 0) = 1 - 4y^2.$$

The mesh is connected to two vertical planes at $x = -0.5$ and $x = 0.5$ with a

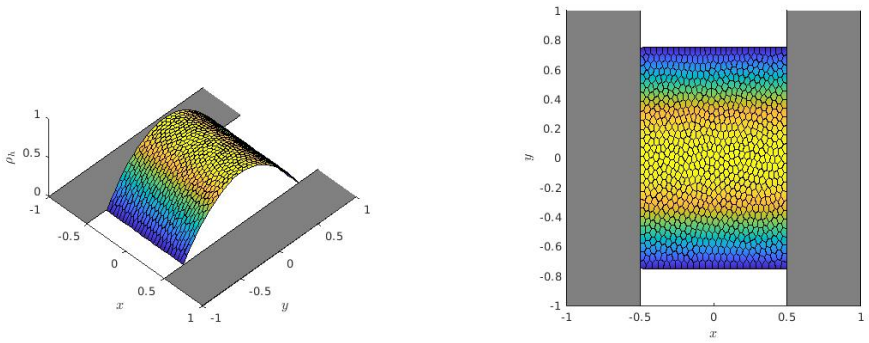

Figure 4.7: A solution snapshot at time $T = 0.1$ for a CVT mesh with 800 elements (right).

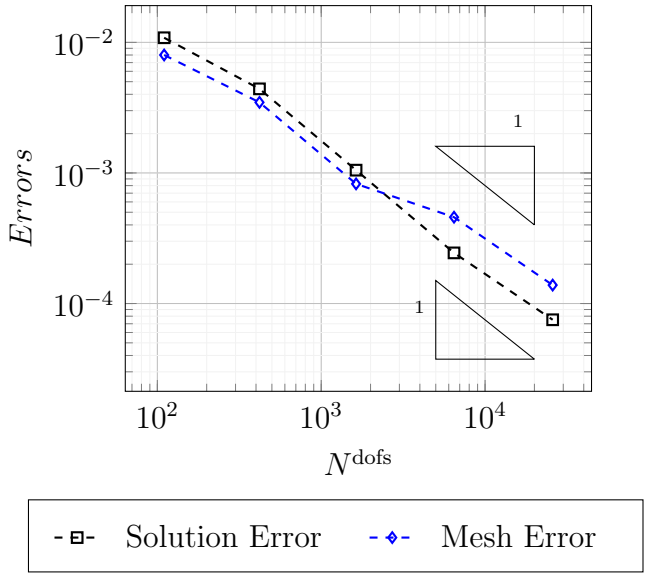

Figure 4.8: Node insertion convergence test on a 1D-type PME similarity solution with $m = 1$: the l^1 solution and mesh errors (4.2) and (4.3), respectively, at time $T = 0.1$.

no-penetration condition strongly imposed in the x direction; namely,

$$\dot{x} = 0 \quad \text{when } |x| = 1/2.$$

Solution snapshots at time $T = 0.1$ are shown in Figure 4.7 for the CVT mesh type. The mesh error is exclusively computed on the top and bottom faces of the rectangular domain.

In this test we only focus our attention on the CVT mesh type. We test the accuracy of the node insertion algorithm by including a discretisation of the two planes into intervals in the y direction. In reference to the fixed domain PME (3.1) we define $\tilde{\Omega} := [-1, 1]^2$ and discretise the boundary $\partial\tilde{\Omega}$ into N uniformly spaced intervals to construct an N-gon. Vertices are then removed

that intersect Ω^0. This results in uniformly discretised intervals along both contact planes.

For the discretisation of the contact planes N is set to an initial value of 32 and is doubled with each mesh refinement. Convergence results are reported in Figure 4.8. Here we observe second order accuracy in both the solution and mesh error. Also the mass is conserved, by design, up to machine precision throughout each test.

Contact demonstrations

We finally present two demonstrations of the node insertion algorithms for Problem 3.1.1 in challenging scenarios without known analytical solution.

To ensure the quality of the mesh is preserved we have to use the Laplacian smoothing ALE velocity defined using Equation (3.19). We remark that the quality of the mesh will still deteriorate over time. The purpose of these examples is to demonstrate the application of the node insertion algorithms. Optimising the choice of ALE velocity is left for future investigation.

In the first demonstration, we consider an initial condition of the PME that has a disconnected support such that self-intersection is expected to occur. The initial condition is given by

$$\rho(\mathbf{x},0) = \begin{cases} 1-4r_1^2 & r_1 = |\mathbf{x}-(-0.8,0)|,\ r_1 \leq 1/2, \\ 1-4r_2^2 & r_2 = |\mathbf{x}-(0.8,0)|,\ r_2 \leq 1/2, \\ 0 & \text{otherwise.} \end{cases}$$

An illustrative example of such initial condition is given in Figure 4.9 (top-left plot). The standard method is applied to simulate the PME for $m = 1$ with the contact detection Algorithm 2 applied at every time level to check for collision between elements. When contact occurs, Algorithm 4 is used to update the monitor distribution whilst the Dirichlet boundary degrees of freedom are flagged as interior degrees of freedom as the mesh connectivity

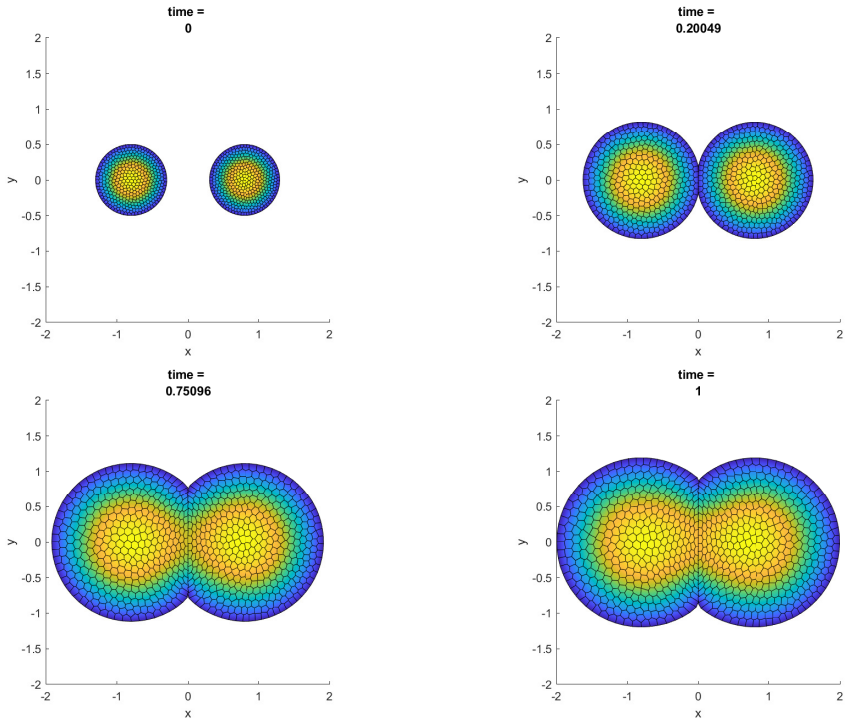

Figure 4.9: Self-intersection demonstration: snapshots of the moving mesh VEM solution at $t = 0$ (top left), $t = 0.20049$ (top right), $t = 0.75096$ (bottom left), and $t = 1$ (bottom right). A CVT type mesh with 800 elements was used to initialise the mesh at $t = 0$.

evolves. Snapshots of the solution evolving over time are reported in Figure 4.9. The behaviour of the PME solution over time is in agreement with fixed mesh finite element approximations of this problem and similar benchmark tests performed for the PME in [87].

To demonstrate the obstacle contact algorithm we consider again the initial condition given by Equation (4.1). A set of obstacles are added to the computational domain in the shape of uniform polygonal approximations of circles with random radii and centres. Each circle is approximated as a uniform polygon of comparable accuracy to the initial mesh. We tested the moving mesh VEM starting with a CVT type mesh made of 800 elements discretising the support of the initial solution. A few snapshots of the numerical solution are shown in Figure 4.10. Pivot nodes are inserted and removed along the contact. Mesh degeneration occurs after $T = 0.2$ and thus the simulation had to be terminated. In this case, and similar to finite element methods, a remeshing

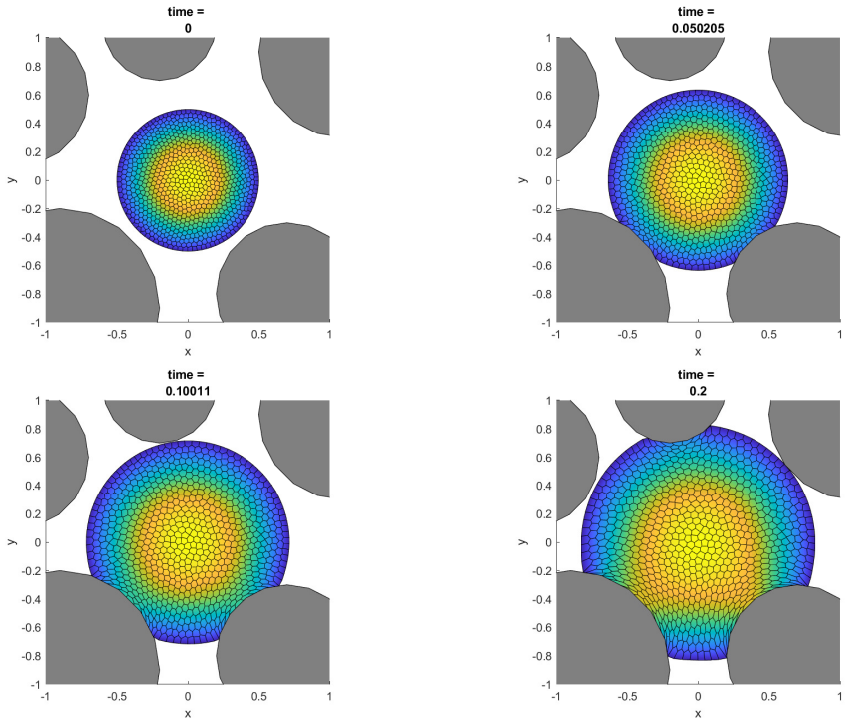

Figure 4.10: Obstacle contact demonstration: snapshots of the moving mesh VEM solution at $T = 0$ (top left), $T = 0.050205$ (top right), $T = 0.10011$ (bottom left) and $T = 0.2$ (bottom right). A CVT type mesh with 800 elements was used to initialise the mesh at $t = 0$.

approach would rectify this issue.

4.3 A Fourth-order Diffusion Problem

To demonstrate the extensibility of the moving mesh VEM we consider the following fourth-order nonlinear diffusion problem used as a benchmark for the original moving mesh method [14, 15], for which the whole of the boundary $\partial\Omega_t$ is free to move, and the differential operator is given by $\mathcal{L}\rho = -\nabla \cdot (\rho^m \nabla \Delta \rho)$. In this chapter we choose $m = 1$, for which there is a simple similarity solution, defined below. The resulting time-dependent equation $\partial\rho/\partial t = \mathcal{L}\rho$ is complemented, at the free boundary, with two conditions on ρ, namely $\rho = \nabla\rho \cdot \mathbf{n} = 0$ plus the kinematic condition $\rho\mathbf{u} \cdot \mathbf{n} = \rho\nabla\Delta\rho \cdot \mathbf{n}$, which is used to determine the boundary velocity $\mathbf{u}$.

In view of its numerical solution, we re-write the fourth-order problem as a coupled system of second-order PDEs by introducing a pressure term $p = -\Delta\rho$.

The problem then reads as: *find* $\rho = \rho(\mathbf{x}, t)$ *such that* $\rho(\mathbf{x}, 0) = \rho_0(\mathbf{x})$ *for* $\mathbf{x} \in \Omega_0$ *and, for all* $t \in (0, T]$,

$$\begin{aligned} \frac{\partial \rho}{\partial t} &= \nabla \cdot (\rho \nabla p) && \mathbf{x} \in \Omega_t, \\ p &= -\Delta \rho && \mathbf{x} \in \Omega_t, \\ \rho = \nabla \rho \cdot \mathbf{n} &= 0 && \mathbf{x} \in \partial\Omega_t, \\ \rho \mathbf{u} \cdot \mathbf{n} &= -\rho \nabla p \cdot \mathbf{n} && \mathbf{x} \in \partial\Omega_t. \end{aligned} \tag{4.8}$$

This problem is structurally very similar to the porous medium equation problem and the moving mesh algorithm remains mostly unchanged. The main addition is an intermediate step which provides the pressure by discretising the weak form of (4.8), namely: *given* $\rho \in H^1(\Omega_t)$, *find* $p \in H^1(\Omega_t)$ *such that*

$$\int_{\Omega_t} pv \; d\mathbf{x} = \int_{\Omega_t} \nabla \rho \cdot \nabla v \; d\mathbf{x} \qquad \forall v \in H^1(\Omega_t).$$

This is discretised using once again the VEM applied to the problem within each time-step t_n: given $\rho_h \in V_{h,n}$ find $p_h \in V_{h,n}$ such that

$$m_{h,n}(p_h, v_h) = \sum_{E_h \in \mathcal{T}_h^n} \int_{E_h} \Pi_0^1 \rho_h \cdot \Pi_0^1 v_h \; d\mathbf{x} \qquad \forall v_h \in V_{h,n},$$

where $m_{h,n}(\cdot, \cdot)$ is defined by Equation (3.32).

Equations (3.16) and (3.20) are also modified for this problem to give, respectively,

$$\begin{aligned} d_n(v) &= -\int_{\Omega_t} \rho \nabla p \cdot \nabla v \; d\mathbf{x}, \\ \dot{\mu}_n(v) &= \int_{\Omega_t} -\rho \nabla v \cdot \{\nabla p + \mathbf{w}\} \; d\mathbf{x} \qquad \forall \, v \in H^1(\Omega_t). \end{aligned}$$

These equations are approximated using the VEM discretisations (3.25) and (3.31), as described in Sections 3.4 and 3.5 for the approximation of the corresponding integrals for the porous medium equation. These are computed at

time t_n as follows:

$$d_{h,n}(v_h) = -\sum_{E_h \in \mathcal{T}_{h,n}} \int_{E_h} \bar{\rho}_h \; \Pi_0^1 p_h \cdot \Pi_0^1 v_h \; d\mathbf{x},$$

$$\dot{\mu}_{h,n}(v_h) = -\sum_{E_h \in \mathcal{T}_{h,n}} \int_{E_h} \Pi_1^0 \rho_h \Pi_0^1 v_h \cdot \left\{ \Pi_0^1 p_h + \Pi_1^0 \mathbf{w}_h \right\} \; d\mathbf{x},$$

$\forall v_h \in V_{h,n}$.

In view of assessing numerically the resulting moving mesh VEM, we recall that this fourth-order nonlinear diffusion problem has a radially symmetric similarity solution given by

$$\rho(r,t) = \begin{cases} At^\beta U_0 (1-\eta^2)^2 & |r| \le A^{\frac{1}{4}} t^\delta \\ 0 & \text{otherwise} \end{cases},$$

where

$$\eta = \frac{r}{A^{\frac{1}{4}} t^\delta}, \qquad \delta = \frac{1}{4+d}, \qquad \beta = 4\delta - 1, \qquad A = U_0^{-4\delta}.$$

Setting $d = 2$, we fix $U_0 = 1/192$ so that $\rho(0, t_0) = 1$ and $t_0 = 1/192$ is specified so that the initial radius is equal to 1.

The VEM is tested on the same sequence of CVT-type meshes used in Section 4.1, with the same coarse-mesh time-step size of 10^{-4} and a reduction by a factor of 4 each time the mesh is refined. Figure 4.11 shows that second-order accuracy is again attained for both the solution and mesh errors. Similar to the PME, the mass is conserved exactly at each time step.

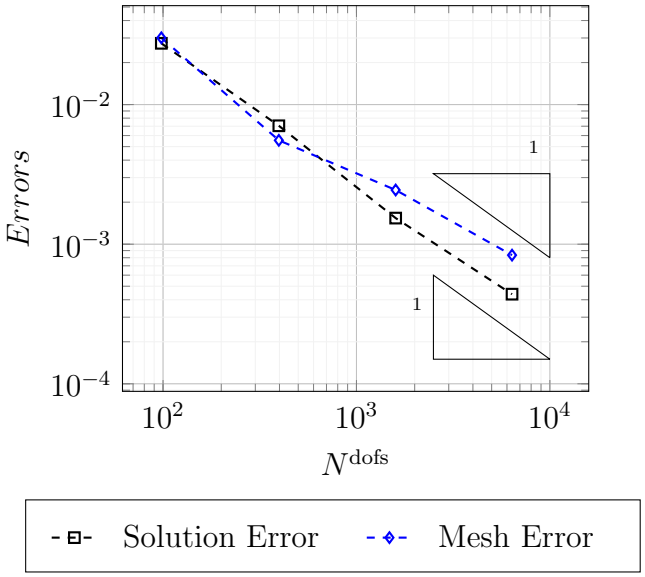

Figure 4.11: Convergence test for the fourth-order diffusion problem: the l^1 solution and mesh errors (4.2) and (4.3), respectively, at time $T = 0.01$ using the CVT mesh type.

Discussion

In Part I a velocity-based moving mesh FEM is extended to polygonal discretisations using the lowest order VEM. The classical VEM reviewed in Section 2.1 was directly employed onto physically reconstructed polygonal meshes at each time step using the mesh connectivity and coordinates of the moving mesh vertices. Numerical tests confirmed that this moving mesh VEM obtains the same orders of accuracy as the original FEM discretisation and satisfies the same conservation properties for specific non-linear diffusion problems with associated conservation laws. Demonstration of node insertion algorithms suggest that this moving mesh method offers practical extensions to more complex problems.

Extensions to this method would include the construction of a linear three-dimensional VEM to test the robustness of this method. Moving polyhedral elements also present the issue of degenerate faces that would be worthy of investigation. Further investigation into optimising the choice of ALE velocity is also required to improve the robustness of this method under complex domain transformations. It is important to note that these are also open problems within the finite element implementation of this algorithm [15]. Extensions of the algorithm to non-conservative ALE updates with more general monitor functions [81] has also been implemented in a VEM framework but these results are omitted from this thesis.

A major set back of this new method is the computational cost associated with computing the discrete bilinear forms. Unlike the FEM case, each local

VEM space and the corresponding projection operators must be recomputed at each discrete time level. Considering this cost with the constraint that the time integration scheme must be explicit, and consequently subject to a stability condition on the time step sizes, results in a numerical method that might not be feasible compared to other numerical schemes. A beneficial property of the VEM is that it can be coupled with FEM discretisations between neighbouring elements. Taking the contact examples of Section 4.2 as motivation, the moving mesh could be initially constructed using only triangular elements and the solution could be computed using a FEM. The VEM would only then be employed when changes in mesh connectivity were required. This would improve the overall efficiency of the proposed moving mesh VEM.

To overcome the restriction to a linear VEM, it is expected that a higher degree of accuracy is required in simulating the ALE map $\mathscr{A}_t$ over time. Numerical experiments which coupled higher-order ($k > 1$) VEMs with this mesh movement strategy resulted in poor quality numerical solutions. The results of [79, 80] suggest that higher orders of convergence should be possible when moving mesh are permitted to contain curvilinear polygons. To address this issue, a greater understanding of VEMs on transformed domains is required, such as VEMs on domains with curved edges [23]. This is the motivation for the work of Part II.

Part II

Isoparametric Virtual Element Methods

Introduction

In Part II the groundwork for the analysis of a moving VEM is presented. Following the discussion of Part I, an understanding of how the VEM behaves under domain transformations is required. A natural consequence of non-linear domain transformations is the presence of curved elements in the physical coordinate system. This motivates the study of the VEM on curved domain transformations.

Within the literature, curved edge VEMs have already been proposed [23, 26, 21, 3, 8, 46]. The limitation of these works for moving mesh methods is that they each depend on explicit *a priori* knowledge of the domain transformation and its regularity. In practise these methods usually only consider transformations local to the domain boundary and of sufficient "closeness" to the identity mapping.

Moving mesh methods require a method that only depends on the degrees of freedom of a given transformation. This is achieved by designing, analysing and implementing isoparametric VEMs. Isoparametric finite elements are characterised by the use of the same finite element space for the solution as that used to approximate the domain geometry [41]. This is also an ideal choice of approach for moving meshes as the domain transformation and/or velocity field is given as a virtual element approximation, such as the method proposed in Part I. In principle isoparametric finite elements extends to using VEM spaces for the domain approximation and solution. A C^0 domain transformation can be defined using a standard VEM space however, the "virtual" nature of this trans-

formation means integration on these isoparametric elements is not necessarily computable. Instead two approaches are proposed to construct a computable VEM, these are referred to as Methods I & II in this thesis. These methods are designed in such a way that they reduce to well established isoparametric finite element methods on triangular/tetrahedral meshes [42, 75].

In Chapter 5 we consider the problem transformed to a computational reference domain by a discrete VEM mapping. The VEM is then further used to approximate the terms resulting from this transformation such as the Jacobian operator, its determinant and inverse operator. Using these approximations we propose a computable VEM on the computational reference domain. This framework closely aligns with and is motivated by the approaches taken by Lipnikov & Morgan [78, 80] discussed in the literature review of Section 2.3.

In Chapter 6 we consider the problem on the domain generated by the VEM approximation of the transformation. On this "virtual domain", we define on local elements a variation of the curved VEM space presented in [23]. To ensure the method is computatable we introduce bespoke polynomial approximation operators such that the consistency and stability terms can be computed to a sufficient degree of accuracy.

Numerical results are presented in Chapter 7 for a selection of problems in two-dimensions. These results verify the expected orders of accuracy in the H^1 norm and suggest optimal orders of accuracy can be obtained in the L^2 norm as well.

The results of Part II are presented for two-dimensional problems. Both methods extent to a conforming three-dimensional method and this will be reviewed at the end of Part II.

This work was conducted in collaboration with Andreas Dedner, *University of Warwick* in addition to the project supervisors. A summary of the research for the two-dimensional case is being drafted for publication at the time of writing this thesis.

The PDE is considered on a bounded Lipschitz domain $\Omega \subset \mathbb{R}^2$ with a corresponding polygonal/polyhedral computational reference domain $\hat{\Omega} \subset \mathbb{R}^2$. To assist the reader in Part II, the following notation is adopted in the proofs of Chapters 5 and 6: constants will be denoted by $C_{i,\alpha,\beta,\gamma}$ to represent hidden constants dependent on α, β and γ with the index i used to differentiate between constants with the same dependencies.

Chapter 5

Method I

The first isoparametric approach is to consider a discretization of the transformed problem onto a mesh of the reference domain. The VEM is then employed to approximate the components of the transformed weak formulation such as the Jacobian operator through the standard polynomial projection operators.

In Section 5.1 The elliptic PDE and domain transformation under consideration for **both methods** is presented. The virtual element approximation of the domain transformation is outlined in Section 5.2. A discussion on the extension of the PDE from the physical domain to the VEM approximation is provided in Section 5.3. The approximation of the Jacobian operator is given in Section 5.4 along with stability and accuracy estimates. The VEM for Method I is formulated in Section 5.5 and well-posedness is verified in Section 5.6. Technical lemmas are presented in Section 5.7 that aid in the derivation of the Strang-type H^1 error estimate of Section 5.8.

5.1 Problem Statement

For both Method I and II, a general elliptic PDE is considered. The PDE is defined as follows: find ρ such that

$$\begin{aligned} -\nabla \cdot (a\nabla\rho) + \mathbf{b} \cdot \nabla\rho + c\rho = f \quad \mathbf{x} \in \Omega, \\ \rho = 0 \quad \mathbf{x} \in \partial\Omega. \end{aligned} \tag{5.1}$$

Standard assumptions are made regarding the PDE coefficients; $a \in L^\infty(\Omega)$, $\mathbf{b} \in [L^\infty(\Omega)]^2, c \in L^\infty(\Omega)$ where there exists an $a_0 > 0$ such that

$$a(\mathbf{x}) \geq a_0 \quad \forall \mathbf{x} \in \Omega,$$

Furthermore, we suppose that there exists a constant $\mu_0 \geq 0$ such that

$$\mu := c - \frac{1}{2}\nabla \cdot \mathbf{b} \geq \mu_0,$$

for almost every $\mathbf{x} \in \Omega$, and assume that $\nabla \cdot \mathbf{b} \in L^\infty(\Omega)$. The resulting weak formulation with convective term in skew-symmetric form: find $\rho \in H_0^1(\Omega)$ such that

$$\mathcal{A}(\rho, v) := A(\rho, v) + B(\rho, v) + C(\rho, v) = l(v) \quad \forall v \in H_0^1(\Omega),$$

where

$$A(\rho, v) = \int_\Omega a\nabla\rho \cdot \nabla v \; d\mathbf{x}, \tag{5.2}$$

$$B(\rho, v) = \frac{1}{2}\int_\Omega \mathbf{b} \cdot (v\nabla\rho - \rho\nabla v) \; d\mathbf{x},$$

$$C(\rho, v) = \int_\Omega \mu \; \rho v \; d\mathbf{x},$$

$$l(v) = \int_\Omega f v \; d\mathbf{x}. \tag{5.3}$$

It is well known that this weak formulation has a unique solution in $H_0^1(\Omega)$ by the Lax-Milgram Theorem [74].

The domain transformation is introduced as an invertible, bijective mapping between the reference and physical domains $\mathscr{A} : \hat{\Omega} \to \Omega$. Additional conditions

are imposed on the mapping inline with assumption made within ALE analysis [27, 28, 60, 55]. We impose that $\mathscr{A}$ is bi-Lipschitz i.e. there exists $\lambda_1, \lambda_2 > 0$ such that

$$\|\mathscr{A}(\boldsymbol{\xi}_1) - \mathscr{A}(\boldsymbol{\xi}_2)\| \leq \lambda_1 \|\boldsymbol{\xi}_1 - \boldsymbol{\xi}_2\| \qquad \forall \boldsymbol{\xi}_1, \boldsymbol{\xi}_2 \in \hat{\Omega},$$
$$\left\|\mathscr{A}^{-1}(\mathbf{x}_1) - \mathscr{A}^{-1}(\mathbf{x}_2)\right\| \leq \lambda_2 \|\mathbf{x}_1 - \mathbf{x}_2\| \qquad \forall \mathbf{x}_1, \mathbf{x}_2 \in \Omega.$$

The Jacobian of $\mathscr{A}$ and its determinant are denoted, consistent with Equation (2.10), by

$$\mathbf{J}_{\mathscr{A}} := \hat{\nabla}\mathscr{A}, \qquad j = \det\left(\mathbf{J}_{\mathscr{A}}\right).$$

Following [27, 28], as a consequence of the assumptions on $\mathscr{A}$, we have that the Jacobian matrix is invertible. We also have that the determinant of the Jacobian is uniformly bounded

$$j \in L^{\infty}(\hat{\Omega}),$$

and strictly positive such that there exits a $j_0 \in (0, 1)$ such that

$$j_0 < j.$$

We note the equivalence of H^1 norms between the physical and reference configurations is a well known classical result [41].

To simplify the presentation of the analysis, we restrict the regularity of the domain transformation to

$$\mathscr{A} \in \left[W^{\infty}_{m+1}(\hat{\Omega})\right]^2,$$

where $m \in \mathbb{N}$ satisfies $m \geq \max\{l, k\}$ with l and k being the domain and solution discretisation degrees respectively.

Remark 9. *The domain $\hat{\Omega}$ or, rather, its polygonal mesh defined below, will serve as a collection of reference polygonal elements for the isoparametric VEM scaled with the physical frame. For instance $\hat{\Omega}$ could just be defined as a polygon whose boundary interpolates $\partial\Omega$. As such, the VEMs introduced provides a direct generalisation of the classical isoparametric finite elements on general meshes.*

5.2 The Virtual Element Mapping

The classical VEM space of degree $l \in \mathbb{N}$ defined by Equation (2.4) is taken as the discretization space on $\hat{\Omega}$. In both Method I and II we denote this space by $\left[\hat{V}_h\right]^2$. The mesh of $\hat{\Omega}$ is denoted by $\hat{\mathcal{T}}_h$ and we assume that this mesh satisfies the standard regularity conditions of Assumption 1. For the purposes of analysing the accuracy of Methods I and II we impose the following assumption on the approximation of the mapping $\mathscr{A}_h \in \left[\hat{V}_h\right]^2$.

Assumption 4.
Let $\hat{\mathcal{T}}_h$ be a polygonal mesh of $\hat{\Omega}$ satisfying the shape regularity Assumption 1 and let $m, l \in \mathbb{N}$. A mapping $\mathscr{A} \in \left[W^{\infty}_{m+1}(\hat{\Omega})\right]^2$ has a VEM approximation $\mathscr{A}_h \in \left[\hat{V}_h\right]^2$ of degree l such that for all $s \in \{1, ..., l\}$

$$\|\mathscr{A} - \mathscr{A}_h\|_{0,\hat{\Omega}} + h\,|\mathscr{A} - \mathscr{A}_h|_{1,\hat{\Omega}} \lesssim h^{s+1}\,|\mathscr{A}|_{s+1,\hat{\Omega}}\,,$$

where the hidden constant depends only on the mesh regularity parameter (2.1) *ϱ and l.*

If we consider the case of $\mathscr{A}_h$ being the interpolant of the degrees of freedom of $\mathscr{A}$, Assumption 4 holds by Theorem 2.1.2. For the remainder of Part II we consider $\mathscr{A}_h$ to be the VEM interpolant of $\mathscr{A}$. This transformation approximation follows the ideas of [79, 80] in that we use the VEM approximation $\mathscr{A}_h$ of $\mathscr{A}$ to define a transformation to a "virtual mesh", which is piecewise polynomial on the edges of $\hat{\mathcal{T}}_h$. The "virtual polygon" E_h is then defined by the image of the approximated mapping

$$E_h := \mathscr{A}_h(\hat{E}),$$

a simple example of this for $l = 1, 2, 3$ is given in Figure 5.1. We then define the virtual mesh $\mathcal{T}_h$ as the collection of virtual polygons given by $\mathscr{A}_h$. We also define the virtual domain by

$$\Omega_h := \bigcup_{\hat{E} \in \hat{\mathcal{T}}_h} E_h.$$

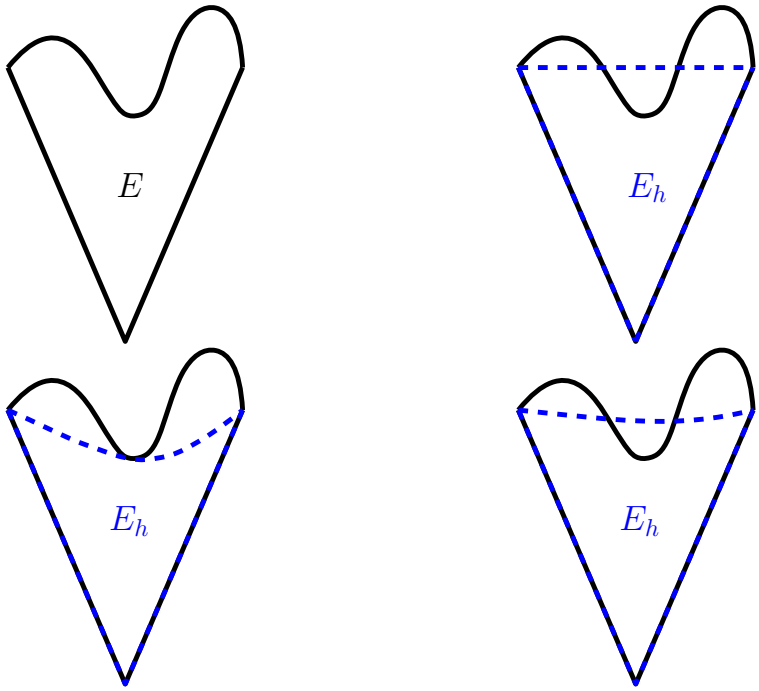

Figure 5.1: An example of the virtual polygon E_h compared to a true mapped element E (top left). The curved boundary is interpolated from $\mathscr{A}$ using degrees of $l = 1$ (top right), $l = 2$ (bottom left) and $l = 3$ (bottom right).

5.3 Extending the PDE to the Virtual Domain

Important to the implementation and analysis of the isoparametric VEM is the availability of extension operators, which is guaranteed by the well-known Stein's Extension Theorem [92, 1].

Theorem 5.3.1 (The Stein Extension Theorem)**.**
Let $\omega \subset \mathbb{R}^2$ be a bounded Lipschitz domain, then for all $p \in [1, \infty]$ and integers $s \geq 0$ there exists an extension operator $\mathfrak{E} : W_s^p(\omega) \to W_s^p(\mathbb{R}^2)$ such that for all $v \in W_s^p(\omega)$ the following holds:

$$\mathfrak{E}v = v \qquad a.e.\ in\ \omega$$

$$\|\mathfrak{E}v\|_{s,p,\mathbb{R}^2} \lesssim \|v\|_{s,p,\omega}\,,$$

where the hidden constant depends only on s and p.

In the following, extensions will be required for the PDE data and exact solutions; for short we shall denote these by $\tilde{v} := \mathfrak{E}v$ and so on.

The weak formulation of the PDE (5.1) has an intuitive extension to the virtual domain Ω_h given by the perturbed bilinear form $\mathcal{B}(\cdot,\cdot) : H^1(\Omega_h) \times H^1(\Omega_h) \to \mathbb{R}$

$$\mathcal{B}(\rho, v) = \int_{\Omega_h} \tilde{a}\nabla\rho \cdot \nabla v + \frac{1}{2}\tilde{\mathbf{b}} \cdot (v\nabla\rho - \rho\nabla v) + \tilde{\mu}\rho v \; d\mathbf{x} \qquad \forall \rho, v \in H^1(\Omega_h),$$

along with an approximation of degree k of

$$\mathcal{B}_h(\rho, v) = \sum_{E_h \in \mathcal{T}_h} \int_{E_h} \tilde{a}\Pi^1_{k-1}\rho \cdot \Pi^1_{k-1}v + \frac{1}{2}\tilde{\mathbf{b}} \cdot (\Pi^0_k v \Pi^1_{k-1}\rho - \Pi^0_k \rho \Pi^1_{k-1} v) + \tilde{\mu}\Pi^0_k\rho \; \Pi^0_k v \; d\mathbf{x}$$
$$+ \mathcal{S}^{E_h}(\rho - \Pi^*_k\rho, v - \Pi^*_k v), \tag{5.4}$$

with the stabilisation term $\mathcal{S}^{E_h}(\cdot,\cdot)$ being defined by the dofi-dofi stability Equation (2.8) scaled appropriately. Following the works of [39] or [20], we can prove that this VEM is well-posed and accurate. The setback in the case of an isoparametric VEM is that Equation (5.4) is not generally computable.

To remedy this, the two methods presented in Part II approximate the physical domain by two separate approaches. Method I, presented in the remainder of this chapter, approximates the Jacobian operator and computes an approximation to the problem on the computational reference domain using the classical VEM outlined in Section 2.1. For Method II of Chapter 6, the DoFs of the domain transformation are used to compute projected elements on which the curved VEM of [23] and bespoke polynomial projection operators are employed to approximate Equation (5.4).

In Part II, the extension of the PDE data, $\tilde{a}$, $\tilde{\mathbf{b}}$, $\tilde{c}$ and $\tilde{f}$ is assumed to be prescribed and computable. In Method I we presume that the PDE data can be represented and integrated on the computational reference domain with sufficient accuracy. In Method II, we presume that the extension of the PDE data can be integrated over each projected element with sufficient accuracy. In the numerical experiments of Chapter 7, the PDE data is well defined over $\mathbb{R}^2$ and the DUNE implementation computes the necessary integration using symbolic UFL expressions of each term [48]. Further studies are required to quantify the

effects of approximating the PDE data extensions for both methods.

5.4 Approximating the Jacobian

Method I utilizes the VEM as a mapping scheme in order to approximate the continuous problem posed on the reference domain. The bilinear forms defined in equations (5.2)-(5.3) can be written in the reference configuration by

$$\begin{aligned} A(\rho, v) &= \int_{\hat{\Omega}} \hat{a}\ \mathbf{J}_{\mathscr{A}}^{-T}\hat{\nabla}\hat{\rho}\cdot\mathbf{J}_{\mathscr{A}}^{-T}\hat{\nabla}\hat{v}\ \det\left(\mathbf{J}_{\mathscr{A}}\right)\ d\boldsymbol{\xi}, \\ B(\rho, v) &= \frac{1}{2}\int_{\hat{\Omega}} \hat{\mathbf{b}}\cdot(\hat{v}\mathbf{J}_{\mathscr{A}}^{-T}\hat{\nabla}\hat{\rho} - \hat{\rho}\mathbf{J}_{\mathscr{A}}^{-T}\hat{\nabla}\hat{v})\ \det\left(\mathbf{J}_{\mathscr{A}}\right)\ d\boldsymbol{\xi}, \\ C(\rho, v) &= \int_{\hat{\Omega}} \hat{\mu}\ \hat{\rho}\hat{v}\ \det\left(\mathbf{J}_{\mathscr{A}}\right)\ d\boldsymbol{\xi}, \\ l(v) &= \int_{\hat{\Omega}} \hat{f}\hat{v}\ \det\left(\mathbf{J}_{\mathscr{A}}\right)\ d\boldsymbol{\xi}. \end{aligned}$$

The novelty of Method I lies in the approximation of the Jacobian matrix, the inverse and the determinant. Using the gradient projection Π^{∇} to approximate these was first introduced in [79], in this work a more standard projection operator is considered. The Jacobian will be approximated for an $l \in \mathbb{N}$ and a reference element $\hat{E}$ via

$$\mathbf{J}_{\mathscr{A},h} := \Pi_{l-1}^{1}\mathscr{A}_h. \tag{5.5}$$

The determinant $j_h := \det\left(\mathbf{J}_{\mathscr{A},h}\right)$ is computable and is also a polynomial of degree $2(l-1)$. The matrix of cofactors of $\mathbf{J}_{\mathscr{A},h}$ is denoted by $\mathbf{C}_{\mathscr{A},h}$.

In quantifying the Jacobian error we make use of the operator norm on the Jacobian and it's inverse transpose. On an open and bounded set $\omega \subseteq \mathbb{R}^d$, for a $d \times d$ matrix operator $\mathbf{A} : [L^2(\omega)]^d \to [L^2(\omega)]^d$ the operator norm is defined by

$$\|\mathbf{A}\|_{op} := \sup_{\|v\|_{L^2(\omega)}=1} \|\mathbf{A}v\|_{0,\omega}.$$

To utilize the projection estimates in bounding the operator norms we use the following bound

$$\|\mathbf{A}\|_{op} \leq d\,\|\mathbf{A}\|_{0,\infty,\omega}. \tag{5.6}$$

Further we consider the inverse of a matrix $\mathbf{A}$ in the form

$$\mathbf{A}^{-1} = \frac{1}{\det(\mathbf{A})}\mathbf{C}_{\mathbf{A}}^{T}, \tag{5.7}$$

where $\mathbf{C}_{\mathbf{A}}$ is the matrix of cofactors of $\mathbf{A}$.

The norm of the determinant and inverse of $\mathbf{A}$ can be bounded using the following lemma.

Lemma 5.4.1.

Let $\mathbf{A}$ *be* 2×2 *matrix operator* $\mathbf{A} : [L^2(\omega)]^2 \to [L^2(\omega)]^2$ *that satisfies* $A_{ij} \in L^\infty(\omega)$ *for* $i, j = 1, 2$. *If there exists a* a_0 *such that* $det(\mathbf{A}) > a_0 > 0$ *for all* $\mathbf{x} \in \omega$ *then it holds that*

$$\begin{aligned} \|det(\mathbf{A})\|_{0,\infty,\omega} &\leq 2\|\mathbf{A}\|_{0,\infty,\omega}^2 \\ \left\|\mathbf{A}^{-1}\right\|_{0,\infty,\omega} &\leq \frac{1}{a_0}\left\|\mathbf{C}_{\mathbf{A}}^{\top}\right\|_{0,\infty,\omega} \leq \frac{1}{a_0}\|\mathbf{A}\|_{0,\infty,\omega}. \end{aligned}$$

Proof. The first inequality is derived from the formula for the determinant and bounding each term by the L^∞ norm of $\mathbf{A}$. Similarly, the second bound is derived from Equation (5.7) using the lower bound on $\det(\mathbf{A})$ and bounding the entries of the matrix of cofactors of $\mathbf{A}$. □

Error & Stability Estimates

For ease of reading, the error terms used in this analysis are defined via

$$\begin{aligned} \mathcal{E}_{\mathbf{J}} &:= \mathbf{J}_{\mathscr{A},h} - \mathbf{J}_{\mathscr{A}}, \\ \mathcal{E}_{\det} &:= j_h - j, \\ \mathcal{E}_{\mathrm{cof}} &:= \mathbf{C}_{\mathscr{A},h} - \mathbf{C}_{\mathscr{A}}, \\ \mathcal{E}_{\mathrm{inv}} &:= \mathbf{J}_{\mathscr{A},h}^{-1} - \mathbf{J}_{\mathscr{A}}^{-1}. \end{aligned}$$

For this method, estimates on an element $\hat{E} \in \hat{\mathcal{T}}_h$ for the approximations of the Jacobian, the determinant and the inverse are given in the following lemmas.

Lemma 5.4.2 (Jacobian Error & Stability Estimate)**.**
Let the degree of the VEM approximation $\mathscr{A}$ be $l \in \mathbb{N}$ and $\mathscr{A} \in W^{\infty}_{l+1}(\hat{E})$. For the approximate Jacobian defined in Equation (5.5) *and a fixed integer $s \in \{0, 1, ..., l\}$ it holds that*

$$\begin{aligned} \|\mathcal{E}_{\mathbf{J}}\|_{0,\infty,\hat{E}} &\lesssim h^{s}_{\hat{E}} \left|\mathscr{A}\right|_{s+1,\infty,\hat{E}}, \\ \|\mathbf{J}_{\mathscr{A},h}\|_{0,\infty,\hat{E}} &\lesssim \left|\mathscr{A}\right|_{1,\infty,\hat{E}}, \end{aligned}$$

where the hidden constants depends only on ϱ and l.

Proof. The stability is a direct consequence of the definition of $\mathbf{J}_{\mathscr{A},h}$. By definition of the approximate Jacobian (5.5) and discrete mapping $\mathscr{A}_h$ and using the triangle inequality it holds that

$$\begin{aligned} \|\mathcal{E}_{\mathbf{J}}\|_{0,\infty,\hat{E}} &= \left\|\Pi^1_{l-1}\mathscr{A}_h - \hat{\nabla}\mathscr{A}\right\|_{0,\infty,\hat{E}} \\ &\leq \left\|\Pi^1_{l-1}\mathscr{A}_h - \hat{\nabla}\Pi^0_l\mathscr{A}\right\|_{0,\infty,\hat{E}} + \left\|\hat{\nabla}\Pi^0_l\mathscr{A} - \hat{\nabla}\mathscr{A}\right\|_{0,\infty,\hat{E}}. \end{aligned} \tag{5.8}$$

we apply the inverse estimate found in [49] (Lemma 1.25) of

$$\|p\|_{0,\infty,\hat{E}} \leq C_{1,\varrho,l} h^{-1}_{\hat{E}} \|p\|_{0,\hat{E}} \qquad \forall p \in \mathbb{P}_{l-1}(\hat{E}).$$

Applying this inverse estimate gives

$$\left\|\Pi^1_{l-1}\mathscr{A}_h - \hat{\nabla}\Pi^0_l\mathscr{A}\right\|_{0,\infty,\hat{E}} \leq C_{1,\varrho,l} h^{-1}_{\hat{E}} \left\|\Pi^1_{l-1}\mathscr{A}_h - \hat{\nabla}\Pi^0_l\mathscr{A}\right\|_{0,\hat{E}}.$$

Further application of the triangle inequality leads to

$$\begin{aligned} \left\|\Pi^1_{l-1}\mathscr{A}_h - \hat{\nabla}\Pi^0_l\mathscr{A}\right\|_{0,\infty,\hat{E}} \leq C_{\varrho,l} h^{-1}_{\hat{E}} \Big(&\left\|\Pi^1_{l-1}\mathscr{A}_h - \Pi^1_{l-1}\mathscr{A}\right\|_{0,\hat{E}} \\ &+ \left\|\Pi^1_{l-1}\mathscr{A} - \hat{\nabla}\mathscr{A}\right\|_{0,\hat{E}} \\ &+ \left\|\hat{\nabla}\mathscr{A} - \hat{\nabla}\Pi^0_l\mathscr{A}\right\|_{0,\hat{E}} \Big). \end{aligned}$$

Application of the stability of Π^1 and Assumption 4 and Theorem 2.1.1 results in

$$\begin{aligned} \left\|\Pi^1_{l-1}\mathscr{A}_h - \hat{\nabla}\Pi^0_l\mathscr{A}\right\|_{0,\infty,\hat{E}} &\leq C_{1,\varrho,l} h^{-1}_{\hat{E}} \left\{ C_{2,\varrho,l} h^{s}_{\hat{E}} \left|\mathscr{A}\right|_{s+1,\hat{E}} \right\} \\ &\leq C_{3,\varrho,l} h^{s-1}_{\hat{E}} \left|\mathscr{A}\right|_{s+1,\hat{E}}. \end{aligned}$$

Inserting this bound into Equation (5.8) and directly applying Theorem 2.1.1 provides

$$\|\mathcal{E}_{\mathbf{J}}\|_{0,\infty,\hat{E}} \leq C_{4,\varrho,l} h_{\hat{E}}^{s} \left|\mathscr{A}\right|_{s+1,\infty,\hat{E}} + C_{3,\varrho,l} h_{\hat{E}}^{s-1} \left|\mathscr{A}\right|_{s+1,\hat{E}} .$$

Noting that

$$\left|\mathscr{A}\right|_{s+1,\hat{E}} \leq h_{\hat{E}} \left|\mathscr{A}\right|_{s+1,\infty,\hat{E}} ,$$

in the second bounding term leads to

$$\|\mathcal{E}_{\mathbf{J}}\|_{0,\infty,\hat{E}} \leq C_{5,\varrho,l} h_{\hat{E}}^{s} \left|\mathscr{A}\right|_{s+1,\infty,\hat{E}} .$$

Stability is given taking $s = 0$ and applying the triangle inequality

$$\|\mathbf{J}_{\mathscr{A},h}\|_{0,\infty,\hat{E}} \leq (C_{5,\varrho,l} + 1) \left|\mathscr{A}\right|_{1,\infty,\hat{E}} .$$

Further bounding the constant term completes the proof. □

Lemma 5.4.3 (Matrix of Cofactors Error & Stability Estimate)**.**
Let the degree of the VEM approximation $\mathscr{A}$ be $l \in \mathbb{N}$ and $\mathbf{C}_{\mathscr{A},h}$ be the matrix of cofactors of $\mathbf{J}_{\mathscr{A},h}$. If $\mathscr{A} \in W_{l+1}^{\infty}(\hat{E})$ then for a fixed integer $s \in \{0, 1, ..., l\}$ it holds that

$$\|\mathbf{C}_{\mathscr{A},h}\|_{0,\infty,\hat{E}} \lesssim \left|\mathscr{A}\right|_{1,\infty,\hat{E}} ,$$
$$\|\mathcal{E}_{cof}\|_{0,\infty,\hat{E}} \lesssim h_{\hat{E}}^{s} \left|\mathscr{A}\right|_{s+1,\infty,\hat{E}} ,$$

where the hidden constants depend only on ϱ and l.

Proof. Stability is a direct consequence of Lemmas 5.4.1 and 5.4.2. The error estimate is trivial, given the definition of minors of $\mathbf{J}_{\mathscr{A},h}$ in the two-dimensional case. □

Lemma 5.4.4 (Determinant Error & Stability)**.**
Let the degree of the VEM approximation $\mathscr{A}$ be $l \in \mathbb{N}$ and $\mathscr{A} \in W_{l+1}^{\infty}(\hat{E})$. For a fixed integer $s \in \{0, 1, ..., l\}$ it holds that

$$\|j_h\|_{0,\infty,\hat{E}} \lesssim \left|\mathscr{A}\right|_{1,\infty,\hat{E}}^{2} ,$$
$$\|\mathcal{E}_{det}\|_{0,\infty,\hat{E}} \lesssim h_{\hat{E}}^{s} \left\|\mathscr{A}\right\|_{s+1,\infty,\hat{E}}^{2} ,$$

where the hidden constants depend only on ϱ and l.

Proof. Stability is given by noting that j_h is a linear combination of products of terms from $\mathbf{J}_{\mathscr{A},h}$. To demonstrate the proof, we denote the terms of the discrete Jacobian matrix terms by $J_{h,i,j} = (\mathbf{J}_{\mathscr{A},h})_{i,j}$ for $i,j = 1,2$. Applying the triangle inequality and bounding L^∞ norms leads to

$$\begin{aligned}\|j_h\|_{0,\infty,\hat{E}} &= \|J_{h,0,0}J_{h,1,1} - J_{h,1,0}J_{h,0,1}\|_{0,\infty,\hat{E}} \\ &\leq \|J_{h,0,0}\|_{0,\infty,\hat{E}} \|J_{h,1,1}\|_{0,\infty,\hat{E}} + \|J_{h,1,0}\|_{0,\infty,\hat{E}} \|J_{h,0,1}\|_{0,\infty,\hat{E}} \\ &\leq 2\|\mathbf{J}_{\mathscr{A},h}\|^2_{0,\infty,\hat{E}},\end{aligned}$$

then applying Lemma 5.4.2 completes the stability estimate. The error estimate is proven with a similar argument. By considering the entries of $\mathbf{J}_{\mathscr{A}}$, we have the error term of

$$\mathcal{E}_{\text{det}} = J_{h,0,0}J_{h,1,1} - J_{0,0}J_{1,1} + J_{1,0}J_{0,1} - J_{h,1,0}J_{h,0,1}.$$

Considering the first difference term, we can rewrite this as

$$J_{h,0,0}J_{h,1,1} - J_{0,0}J_{1,1} = J_{h,0,0}(J_{h,1,1} - J_{1,1}) + J_{11}(J_{h,0,0} - J_{0,0}).$$

bounding this term using the triangle inequality and Lemma 5.4.2 gives

$$\begin{aligned}\|J_{h,0,0}J_{h,1,1} - J_{0,0}J_{1,1}\|_{0,\infty,\hat{E}} &\leq \|\mathbf{J}_{\mathscr{A},h}\|_{0,\infty,\hat{E}} \|\mathcal{E}_{\mathbf{J}}\|_{0,\infty,\hat{E}} + \|\mathbf{J}_{\mathscr{A}}\|_{0,\infty,\hat{E}} \|\mathcal{E}_{\mathbf{J}}\|_{0,\infty,\hat{E}} \\ &\leq (C_{1,\varrho,l}+1)\,|\mathscr{A}|_{1,\infty,\hat{E}} \|\mathcal{E}_{\mathbf{J}}\|_{0,\infty,\hat{E}} \\ &\leq C_{2,\varrho,l}(C_{1,\varrho,l}+1)h^s_{\hat{E}}\,|\mathscr{A}|_{1,\infty,\hat{E}}\,|\mathscr{A}|_{s+1,\infty,\hat{E}}\,.\end{aligned}$$

By repeating this argument for the remaining terms in $\mathcal{E}_{\text{det}}$ and applying the triangle inequality we get

$$\|\mathcal{E}_{\text{det}}\|_{0,\infty,\hat{E}} \leq 2C_{2,\varrho,l}(C_{1,\varrho,l}+1)h^s_{\hat{E}}\,|\mathscr{A}|_{1,\infty,\hat{E}}\,|\mathscr{A}|_{s+1,\infty,\hat{E}}\,.$$

Bounding the semi-norms by the W^∞_{s+1} norm completes the proof. □

Lemma 5.4.5 (Inverse Error & Stability Estimate)**.**
Let the degree of the VEM approximation $\mathscr{A}$ be $l \in \mathbb{N}$ and $\mathscr{A} \in W^\infty_{l+1}(\hat{E})$. It

holds for sufficiently small $h_{\hat{E}}$ and a fixed integer $s \in \{0, 1, ..., l\}$ that

$$j_h > j_0, \tag{5.9}$$

$$\left\|\mathbf{J}_{\mathscr{A},h}^{-1}\right\|_{0,\infty,\hat{E}} \lesssim |\mathscr{A}|_{1,\infty,\hat{E}}\,, \tag{5.10}$$

$$\|\mathcal{E}_{inv}\|_{0,\infty,\hat{E}} \lesssim h_{\hat{E}}^{s} \|\mathscr{A}\|_{s+1,\infty,\hat{E}}^{3}\,,$$

where the hidden constants only depend on j_0, ϱ and l.

Proof. From the estimate of Lemma 5.4.4, we have a.e. on $\hat{E}$ that

$$|j - j_h| \leq C_{1,\varrho,l} h_{\hat{E}} \|\mathscr{A}\|_{2,\infty,\hat{E}}^{2}\,,$$

then we define

$$h^* = \frac{\delta}{C_{1,\varrho,l} \|\mathscr{A}\|_{2,\infty,\hat{E}}^{2}}, \qquad \delta = \|j\|_{0,\infty,\hat{E}} - j_0,$$

such that for $h_E \leq h^*$ we have $\|\mathcal{E}_{\det}\|_{0,\infty,\hat{E}} \leq \delta$. Equation (5.9) is then deduced by using a continuity argument.

Equation (5.10) is proven by using Equation (5.9) and Lemma 5.4.1

$$\begin{aligned}\left\|\mathbf{J}_{\mathscr{A},h}^{-1}\right\|_{0,\infty,\hat{E}} &\leq \left\|\frac{1}{j_h}\right\|_{0,\infty,\hat{E}} \left\|\mathbf{C}_{\mathscr{A},h}^{\top}\right\|_{0,\infty,\hat{E}} \\ &\leq \frac{1}{j_0} \|\mathbf{J}_{\mathscr{A},h}\|_{0,\infty,\hat{E}}\,.\end{aligned}$$

The bound is then concluded by applying Lemma 5.4.2.

Using the definition of $\mathcal{E}_{\mathbf{J}}$ and substitution of $\mathbf{I} = \mathbf{J}_{\mathscr{A}}\mathbf{J}_{\mathscr{A}}^{-1}$ results in

$$\mathcal{E}_{\text{inv}} = -\mathbf{J}_{\mathscr{A},h}^{-1}\, \mathcal{E}_{\mathbf{J}}\, \mathbf{J}_{\mathscr{A}}^{-1}.$$

Expanding and bounding norms gives

$$\|\mathcal{E}_{\text{inv}}\|_{0,\infty,\hat{E}} \leq \left\|\mathbf{J}_{\mathscr{A},h}^{-1}\right\|_{0,\infty,\hat{E}} \|\mathcal{E}_{\mathbf{J}}\|_{0,\infty,\hat{E}} \left\|\mathbf{J}_{\mathscr{A}}^{-1}\right\|_{0,\infty,\hat{E}}\,.$$

Lemmas 5.4.1 and 5.4.2 and Equation (5.10) provides

$$\begin{aligned}\|\mathcal{E}_{\text{inv}}\|_{0,\infty,\hat{E}} &\leq \left\|\mathbf{J}_{\mathscr{A},h}^{-1}\right\|_{0,\infty,\hat{E}} \|\mathcal{E}_{\mathbf{J}}\|_{0,\infty,\hat{E}} \frac{1}{j_0} |\mathscr{A}|_{1,\infty,\hat{E}} \\ &\leq \left\|\mathbf{J}_{\mathscr{A},h}^{-1}\right\|_{0,\infty,\hat{E}} C_{2,\varrho,l} h_{\hat{E}}^{s} |\mathscr{A}|_{s+1,\infty,\hat{E}} \frac{1}{j_0} |\mathscr{A}|_{1,\infty,\hat{E}} \\ &\leq C_{3,j_0,\varrho,l} |\mathscr{A}|_{1,\infty,\hat{E}} C_{2,\varrho,l} h_{\hat{E}}^{s} |\mathscr{A}|_{s+1,\infty,\hat{E}} \frac{1}{j_0} |\mathscr{A}|_{1,\infty,\hat{E}}\,.\end{aligned}$$

Collecting terms gives

$$\|\mathcal{E}_{\text{inv}}\|_{0,\infty,\hat{E}} \leq \frac{1}{j_0} C_{2,\varrho,l} C_{3,j_0,\varrho,l} h_{\hat{E}}^s \, |\mathscr{A}|_{1,\infty,\hat{E}}^2 \, |\mathscr{A}|_{s+1,\infty,\hat{E}} \, .$$

We complete the proof by bounding the semi-norms by the W_{s+1}^{∞} norm. □

Remark 10. *In practise, we do not require the invertibility of* $\mathbf{J}_{\mathscr{A},h}$ *provided that the determinant is non-zero at the quadrature points used in numerical integration.*

5.5 Method I Formulation

In Method I we seek a solution to the PDE (5.1) in the reference configuration. The solution space of degree k is defined on the reference mesh $\hat{\mathcal{T}}_h$ using the classical VEM space given by Equation (2.4) denoted in this chapter by $\hat{V}_h$. The restriction of this space to VEM functions with zero trace on $\partial\hat{\Omega}$ is defined as $\hat{V}_{h,0} := \hat{V}_h \cap H_0^1(\hat{\Omega})$. Here we seek a VEM approximation $\hat{\rho}_h \in \hat{V}_{h,0}$ of degree $k \in \mathbb{N}$. In the analysis of Method I we assume that the PDE data is sufficiently regular on $\hat{\Omega}$, this can be proven by following the analysis of [42]. The method formulation is given by summing up local contributions from elements on the computational mesh $\hat{\mathcal{T}}_h$, hence

$$\begin{aligned}
\mathcal{A}_h(\rho_h, v_h) &= \sum_{\hat{E}\in\hat{\mathcal{T}}_h} A_h^{\hat{E}}(\rho_h, v_h) + B_h^{\hat{E}}(\rho_h, v_h) + C_h^{\hat{E}}(\rho_h, v_h) \\
&\quad + \sum_{\hat{E}\in\hat{\mathcal{T}}_h} \mathcal{S}^{\hat{E}}(\hat{\rho}_h - \Pi_k^{\nabla}\hat{\rho}_h, \hat{v}_h - \Pi_k^{\nabla}\hat{v}_h), \\
l_h(v_h) &= \sum_{\hat{E}\in\hat{\mathcal{T}}_h} l_h^{\hat{E}}(v_h),
\end{aligned}$$

with the local forms defined by

$$A_h^{\hat{E}}(\rho_h, v_h) = \int_{\hat{E}} \hat{a}\ \mathbf{J}_{\mathscr{A},h}^{-T}\Pi_{k-1}^1\hat{\rho}_h \cdot \mathbf{J}_{\mathscr{A},h}^{-T}\Pi_{k-1}^1\hat{v}_h\ j_h\ d\boldsymbol{\xi} \tag{5.11}$$

$$B_h^{\hat{E}}(\rho_h, v_h) = \frac{1}{2}\int_{\hat{E}} \hat{\mathbf{b}} \cdot (\Pi_k^0\hat{v}_h\ \mathbf{J}_{\mathscr{A},h}^{-T}\Pi_{k-1}^1\hat{\rho}_h - \Pi_k^0\hat{\rho}_h\ \mathbf{J}_{\mathscr{A},h}^{-T}\Pi_{k-1}^1\hat{v}_h)\ j_h\ d\boldsymbol{\xi},$$

$$C_h^{\hat{E}}(\rho_h, v_h) = \int_{\hat{E}} \hat{\mu}\ \Pi_k^0\hat{\rho}_h\ \Pi_k^0\hat{v}_h\ j_h\ d\boldsymbol{\xi},$$

$$l_h^{\hat{E}}(v_h) = \int_{\hat{E}} \hat{f}_h\ \Pi_k^0\ \hat{v}_h\ j_h\ d\boldsymbol{\xi}.$$

For the stabilization term $\mathcal{S}^{\hat{E}}(\cdot,\cdot)$ we use the dofi-dofi stabilisation defined in Equation (2.8) scaled appropriately by the PDE coefficients [39]

$$\mathcal{S}^{\hat{E}}(\rho_h, v_h) = (\bar{a} + h_{\hat{E}}^2\bar{\mu})\sum_{i=1}^{N^{\text{dofs}}} \text{dof}_i(\hat{\rho}_h)\cdot\text{dof}_i(\hat{v}_h), \tag{5.12}$$

where $\bar{\cdot}$ denotes the nodal average over $\hat{E}$. For the forcing term approximation $\hat{f}_h \in \hat{V}_h$, we consider the Π^0 projection of $\hat{f}_h := \Pi_{k-2}^0\hat{f}$ for $k \geq 2$ and $\hat{f}_h := \Pi_0^0\hat{f}$ for $k = 1$.

The VEM is then defined as follows: find $\hat{\rho}_h \in \hat{V}_{h,0}$ such that

$$\mathcal{A}_h(\rho_h, v_h) = l_h(v_h) \qquad \forall \hat{v}_h \in \hat{V}_{h,0}.$$

Remark 11. *We note that the choice of stabilization scaling is given as an example. For brevity, we do not present the analysis for other choices of stabilization. In these approaches we can follow the proof structure outlined in [20] to obtain the same optimal bounds.*

5.6 Well-posedness

In order to prove this method satisfies the conditions of the Lax-Milgram theorem, we require a series of technical lemmas that quantifies the error in approximating the bilinear form. The continuity and coercivity constants obtained in this section will also influence the computation of the H^1 estimate in the subsequent sections.

Lemma 5.6.1 (Continuity)**.**

For a sufficiently small mesh size h of $\hat{\mathcal{T}}_h$, The discrete bilinear and linear forms defined in this method are continuous and satisfy for all $\hat{\rho}_h, \hat{v}_h \in \hat{V}_{h,0}$

$$\mathcal{A}_h(\rho_h, v_h) \lesssim \|\hat{\rho}_h\|_{1,\hat{\Omega}} \|\hat{v}_h\|_{1,\hat{\Omega}},$$

where the hidden constant depends on ϱ, l, the L^∞ norms of a, $\mathbf{b}$ and μ, the stability constant given in Assumption 3 and the W_1^∞ semi-norm of $\mathscr{A}$. It also holds that

$$l_h(v_h) \lesssim \left\|\hat{f}\right\|_{0,\hat{E}} \|\hat{v}_h\|_{1,\hat{E}},$$

with the hidden constant in this case only dependent on ϱ, l and the W_1^∞ semi-norm of $\mathscr{A}$.

Proof. Through applying Hőlder's, Cauchy-Schwarz and the operator norm inequality of Equation (5.6) we have on a local reference element $\hat{E} \in \hat{\mathcal{T}}_h$ that

$$\begin{aligned}
A_h^{\hat{E}}(\rho_h, v_h) &\leq 4 \|\hat{a}\|_{0,\infty,\hat{E}} \|\mathbf{C}_{\mathscr{A},h}\|_{0,\infty,\hat{E}} \left\|\mathbf{J}_{\mathscr{A},h}^{-T}\right\|_{0,\infty,\hat{E}} \left\|\Pi_{k-1}^1 \hat{\rho}_h\right\|_{0,\hat{E}} \left\|\Pi_{k-1}^1 \hat{v}_h\right\|_{0,\hat{E}}, \\
B_h^{\hat{E}}(\rho_h, v_h) &\leq 2 \left\|\hat{\mathbf{b}}\right\|_{0,\infty,\hat{E}} \|\mathbf{C}_{\mathscr{A},h}\|_{0,\infty,\hat{E}} \left(\left\|\Pi_{k-1}^1 \hat{\rho}_h\right\|_{0,\hat{E}} \left\|\Pi_k^0 \hat{v}_h\right\|_{1,\hat{E}} + \left\|\Pi_k^0 \hat{\rho}_h\right\|_{1,\hat{E}} \left\|\Pi_{k-1}^1 \hat{v}_h\right\|_{1,\hat{E}}\right), \\
C_h^{\hat{E}}(\rho_h, v_h) &\leq \|\hat{\mu}\|_{0,\infty,\hat{E}} \|j_h\|_{0,\infty,\hat{E}} \left\|\Pi_k^0 \hat{\rho}_h\right\|_{0,\hat{E}} \left\|\Pi_k^0 \hat{v}_h\right\|_{0,\hat{E}}.
\end{aligned}$$

Applying the stability of the projection operators and bounding these norms by the H^1 norm leads to

$$\begin{aligned}
A_h^{\hat{E}}(\rho_h, v_h) &\leq 4 \|\hat{a}\|_{0,\infty,\hat{E}} \|\mathbf{C}_{\mathscr{A},h}\|_{0,\infty,\hat{E}} \left\|\mathbf{J}_{\mathscr{A},h}^{-T}\right\|_{0,\infty,\hat{E}} \|\hat{\rho}_h\|_{1,\hat{E}} \|\hat{v}_h\|_{1,\hat{E}}, \\
B_h^{\hat{E}}(\rho_h, v_h) &\leq 4 \left\|\hat{\mathbf{b}}\right\|_{0,\infty,\hat{E}} \|\mathbf{C}_{\mathscr{A},h}\|_{0,\infty,\hat{E}} \|\hat{\rho}_h\|_{1,\hat{E}} \|\hat{v}_h\|_{1,\hat{E}}, \\
C_h^{\hat{E}}(\rho_h, v_h) &\leq \|\hat{\mu}\|_{0,\infty,\hat{E}} \|j_h\|_{0,\infty,\hat{E}} \|\hat{\rho}_h\|_{1,\hat{E}} \|\hat{v}_h\|_{1,\hat{E}}.
\end{aligned}$$

Next we apply Lemmas 5.4.3, 5.4.5 and 5.4.4 to get

$$\begin{aligned}
A_h^{\hat{E}}(\rho_h, v_h) &\leq C_{1,j_0,\varrho,l} \|\hat{a}\|_{0,\infty,\hat{E}} |\mathscr{A}|_{1,\infty,\hat{E}}^2 \|\hat{\rho}_h\|_{1,\hat{E}} \|\hat{v}_h\|_{1,\hat{E}}, \\
B_h^{\hat{E}}(\rho_h, v_h) &\leq C_{2,\varrho,l} \left\|\hat{\mathbf{b}}\right\|_{0,\infty,\hat{E}} |\mathscr{A}|_{1,\infty,\hat{E}} \|\hat{\rho}_h\|_{1,\hat{E}} \|\hat{v}_h\|_{1,\hat{E}}, \\
C_h^{\hat{E}}(\rho_h, v_h) &\leq C_{3,\varrho,l} \|\hat{\mu}\|_{0,\infty,\hat{E}} |\mathscr{A}|_{1,\infty,\hat{E}}^2 \|\hat{\rho}_h\|_{1,\hat{E}} \|\hat{v}_h\|_{1,\hat{E}}.
\end{aligned}$$

For the stabilization term, we employ the stability Assumption 3: there exists an $\alpha^* > 0$ such that

$$\mathcal{S}^{\hat{E}}(\hat{\rho}_h - \Pi_k^\nabla \hat{\rho}_h, \hat{v}_h - \Pi_k^\nabla \hat{v}_h) \leq \alpha^* \left\|\hat{\rho}_h - \Pi_k^\nabla \hat{\rho}_h\right\|_{1,\hat{E}} \left\|\hat{v}_h - \Pi_k^\nabla \hat{v}_h\right\|_{1,\hat{E}},$$

from this we apply Theorem 2.1.1 to get

$$\mathcal{S}^{\hat{E}}(\hat{\rho}_h - \Pi_k^\nabla \hat{\rho}_h, \hat{v}_h - \Pi_k^\nabla \hat{v}_h) \leq C_{4,\varrho,k,\alpha^*} \|\hat{\rho}_h\|_{1,\hat{E}} \|\hat{v}_h\|_{1,\hat{E}}.$$

Summing these bounds over $\hat{\mathcal{T}}_h$, globally bounding the W_1^∞ semi-norm and introducing the constant

$$C_{5,j_0,a,\mathbf{b},\mu,\varrho,k,l,\alpha^*} = \max_{\hat{E}\in\hat{\mathcal{T}}_h} \left\{ C_{1,j_0,\varrho,l} \|\hat{a}\|_{0,\infty,\hat{E}}, \; C_{2,\varrho,l} \left\|\hat{\mathbf{b}}\right\|_{0,\infty,\hat{E}}, \; C_{3,\varrho,l} \|\hat{\mu}\|_{0,\infty,\hat{E}}, \; C_{4,\varrho,k,\alpha^*} \right\},$$

results in

$$\mathcal{A}_h(\rho_h, v_h) \leq C_{5,j_0,a,\mathbf{b},\mu,\varrho,k,l,\alpha^*} \left(|\mathscr{A}|_{1,\infty,\hat{\Omega}} + |\mathscr{A}|_{1,\infty,\hat{\Omega}}^2 + 1 \right) \|\hat{\rho}_h\|_{1,\hat{\Omega}} \|\hat{v}_h\|_{1,\hat{\Omega}}.$$

For the linear form $l_h(\cdot)$, we prove the continuity bound by noting the similarity to $C_h(\cdot,\cdot)$ with $\hat{v}_h = \hat{f}_h$ and $\hat{\mu} = 1$, hence

$$l_h^{\hat{E}}(v_h) \leq C_{3,\varrho,l} |\mathscr{A}|_{1,\infty,\hat{E}}^2 \left\|\hat{f}_h\right\|_{0,\hat{E}} \|\hat{v}_h\|_{0,\hat{E}}.$$

Since we have chosen $\hat{f}_h$ as the Π^0 projection, we apply the stability of Π^0 and bound the norm of $\hat{v}_h$ to get

$$l_h^{\hat{E}}(v_h) \leq C_{3,\varrho,l} |\mathscr{A}|_{1,\infty,\hat{E}}^2 \left\|\hat{f}\right\|_{0,\hat{E}} \|\hat{v}_h\|_{1,\hat{E}}.$$

The proof is concluded by summing over elements and bounding the W_1^∞ semi-norm. □

Lemma 5.6.2 (Coercivity)**.**

For all $\hat{v}_h \in \hat{V}_{h,0}$ it holds for sufficiently small mesh size h that

$$\|\hat{v}_h\|_{1,\hat{\Omega}}^2 \lesssim \mathcal{A}_h(v_h, v_h),$$

where the hidden constant is dependent on j_0, the PDE coefficient lower bounds of a_0 and μ_0, the stability constant of Assumption 3, ϱ, l and the W_1^∞ semi-norm of $\mathscr{A}$.

Proof. Consider the L^2 norm of the Π^1_{k-1} projection on a element $\hat{E} \in \hat{\mathcal{T}}_h$ and assume a sufficiently small h such that $j_0 \leq j_h$ (see Lemma 5.4.4). This gives

$$\begin{aligned} a_0 j_0 \left\| \Pi^1_{k-1} \hat{v}_h \right\|^2_{0,\hat{E}} &\leq \int_{\hat{E}} \hat{a} \left| \Pi^1_{k-1} \hat{v}_h \right|^2 j_h \, d\boldsymbol{\xi} \\ &\leq \int_{\hat{E}} \hat{a} \left| \mathbf{J}^T_{\mathscr{A},h} \mathbf{J}^{-T}_{\mathscr{A},h} \Pi^1_{k-1} \hat{v}_h \right|^2 j_h \, d\boldsymbol{\xi} \\ &\leq \left\| \mathbf{J}^T_{\mathscr{A},h} \right\|^2_{0,\infty,\hat{E}} \int_{\hat{E}} \hat{a} \left| \mathbf{J}^{-T}_{\mathscr{A},h} \Pi^1_{k-1} \hat{v}_h \right|^2 j_h \, d\boldsymbol{\xi} \\ &\leq \left\| \mathbf{J}^T_{\mathscr{A},h} \right\|^2_{0,\infty,\hat{E}} A^{\hat{E}}_h(v_h, v_h). \end{aligned}$$

where in the last line we have applied the definition of $A_h(\cdot,\cdot)$ in Equation (5.11). Applying Lemma 5.4.2 gives a bound of

$$a_0 j_0 \left\| \Pi^1_{k-1} \hat{v}_h \right\|^2_{0,\hat{E}} \leq C_{1,\varrho,l,\mathscr{A}} A^{\hat{E}}_h(v_h, v_h). \tag{5.13}$$

Similarly, we have

$$C^{\hat{E}}_h(\hat{v}_h, v_h) \geq \mu_0 j_0 \left\| \Pi^0_k \hat{v}_h \right\|^2_{0,\hat{E}},$$

and we note the definition of $B^{\hat{E}}_h(\cdot,\cdot)$ implies that

$$B^{\hat{E}}_h(\hat{v}_h, v_h) = 0.$$

By adding

$$C^{\hat{E}}_h(\hat{v}_h, v_h) + B^{\hat{E}}_h(\hat{v}_h, v_h) + S^{\hat{E}}(\hat{v}_h - \Pi^\nabla_k \hat{v}_h, \hat{v}_h - \Pi^\nabla_k v_h),$$

to Equation (5.13) we get

$$\begin{aligned} (C_{1,\varrho,l,\mathscr{A}} + 1) \, \mathcal{A}^{\hat{E}}_h(v_h, v_h) \geq \mu_0 j_0 \left\| \Pi^0_k \hat{v}_h \right\|^2_{0,\hat{E}} + a_0 j_0 \left\| \Pi^1_{k-1} \hat{v}_h \right\|^2_{0,\hat{E}} \\ + S^{\hat{E}}(\hat{v}_h - \Pi^\nabla_k \hat{v}_h, \hat{v}_h - \Pi^\nabla_k v_h). \end{aligned} \tag{5.14}$$

Following standard VEM analysis [20, 39], we apply the definition of the stabilization term in Equation (5.12) and Assumption 3: there exists an $\alpha_* > 0$ such that

$$S^{\hat{E}}(\hat{v}_h - \Pi^\nabla_k \hat{v}_h, \hat{v}_h - \Pi^\nabla_k v_h) \geq \alpha_* a_0 \left\| \hat{\nabla} \hat{v}_h - \Pi^1_{k-1} \hat{v}_h \right\|_{0,\hat{E}} + \alpha_* \mu_0 \left\| \hat{v}_h - \Pi^0_k \hat{v}_h \right\|_{0,\hat{E}},$$

from which substitution into Equation (5.14) obtains

$$(C_{1,\varrho,l,\mathscr{A}}+1)\,\mathcal{A}_h^{\hat{E}}(v_h,v_h) \geq \min\{\alpha_*,j_0\}\mu_0\,\|\hat{v}_h\|_{0,\hat{E}}^2 + \min\{\alpha_*,j_0\}a_0\left\|\hat{\nabla}\hat{v}_h\right\|_{0,\hat{E}}^2.$$

We introduce the constant $C_{2,\alpha_*,j_0,\mu_0,a_0}=\min\{\alpha_*,j_0,\mu_0,a_0\}$ which gives us

$$(C_{1,\varrho,l,\mathscr{A}}+1)\,\mathcal{A}_h^{\hat{E}}(v_h,v_h) \geq C_{2,\alpha_*,j_0,\mu_0,a_0}\,\|\hat{v}_h\|_{1,\hat{E}}^2.$$

Summing over elements and bounding norms completes the proof. □

We now prove that Method I is well posed through the following theorem.

Theorem 5.6.3 (Well-posedness)**.**
For sufficiently small h, the VEM: find $\hat{\rho}_h \in \hat{V}_{h,0}$ such that

$$\mathcal{A}_h(\rho_h,v_h)=l_h(v_h) \qquad \forall \hat{v}_h \in \hat{V}_{h,0},$$

it holds that $\mathcal{A}_h(\cdot,\cdot)$ is coercive and continuous and $l_h(\cdot)$ is continous. Consequently, the solution $\hat{\rho}_h \in \hat{V}_{h,0}$ exists and is unique.

Proof. Continuity of $\mathcal{A}_h(\cdot,\cdot)$ and $l_h(\cdot)$ is given as a consequence of Lemma 5.6.1. Coercivity is given by Lemma 5.6.2. The proof is concluded by application of the Lax-Milgram Theorem. □

5.7 Consistency Errors

We follow the approaches of [20, 39] in constructing the Strang-type error bound. Specifically, we are required to quantify the error terms on a given element $\hat{E} \in \hat{\mathcal{T}}_h$ for a polynomial $\hat{p} \in \mathbb{P}_k(\hat{E})$ and all $\hat{v}_h \in V_k(\hat{E})$

$$\mathcal{A}_h^{\hat{E}}(\hat{p},v_h) - \mathcal{A}^{\hat{E}}(\hat{p},v_h),$$
$$l_h^{\hat{E}}(v_h) - l^{\hat{E}}(v_h).$$

These bounds are proven in Theorem 5.7.3.

In order to compute these bounds we require product rules for Sobolev norms given by the following Lemmas 5.7.1 and 5.7.2.

Lemma 5.7.1.

Let $\omega \subset \mathbb{R}^2$ be a bounded domain and $s \in \mathbb{N}$. For $f \in W_s^\infty(\omega)$ and $g \in H^s(\omega)$ it holds that

$$\|fg\|_{s,\omega} \lesssim \|f\|_{s,\infty,\omega} \|g\|_{s,\omega},$$

where the hidden constant depends only on s. For a matrix $\mathbf{A}$ with entries $A_{i,j} \in W_s^\infty(\omega)$ for $i,j = 1,2$ and a vector $\mathbf{v} \in [H^s(\omega)]^2$ it holds that

$$\|\mathbf{Av}\|_{s,\omega} \lesssim \|\mathbf{A}\|_{s,\infty,\omega} \|\mathbf{v}\|_{s,\omega},$$

where the hidden constant depends only on s.

Proof. We prove the first bound by considering the semi-norm for some integer $0 \leq l \leq s$. For indexing sets α, β with $|\alpha| = l$ it holds a.e. that

$$D^\alpha(fg) = \sum_{|\beta| \leq |\alpha|} \binom{|\alpha|}{|\beta|} D^\beta f \; D^{\alpha-\beta} g.$$

We then take the L^2 norm, bound the derivative of f and the binomial coefficients and apply the triangle inequality to get

$$\|D^\alpha(fg)\|_{0,\omega} \leq s! \, \|f\|_{s,\infty,\omega} \sum_{|\beta| \leq |\alpha|} \left\|D^{\alpha-\beta} g\right\|_{0,\omega},$$

$$\leq C_{1,s} \, \|f\|_{s,\infty,\omega} \, \|g\|_{s,\omega}.$$

Considering the definition of the $\|fg\|_{s,\omega}^2$ and bounding terms in the summation completes the first bound. For the second bound we have using standard arguments

$$\|\mathbf{Av}\|_{s,\omega}^2 = \sum_{i=1}^{2} \|(\mathbf{Av})_i\|_{s,\omega}^2 \leq 4 \max_{1 \leq i,j \leq 2} \|A_{ij} v_i\|_{s,\omega}^2.$$

Applying the first bound and the definition of the matrix and vector Sobolev norms gives

$$\|\mathbf{Av}\|_{s,\omega}^2 \leq C_{2,s} \, \|\mathbf{A}\|_{s,\infty,\omega}^2 \, \|\mathbf{v}\|_{s,\omega}^2,$$

taking square roots completes this proof. □

Lemma 5.7.2.

Let $\omega \subset \mathbb{R}^2$ be a bounded domain and $f \in W_s^\infty(\omega)$ for some $s \in \mathbb{N}$. If there exists an $f_0 \in (0,1)$ such that $f > f_0$ a.e. in ω then it also holds that

$$\left\|f^{-1}\right\|_{s,\infty,\omega} \lesssim \|f\|_{s,\infty,\omega},$$

where the hidden constant depends on s and f_0.

Proof. The chain rule and quotient rule give, for any indexing set $|\alpha| \leq s$, that

$$\begin{aligned}
\left|D^\alpha f^{-1}\right| &= \left|(-1)^{|\alpha|}\frac{\alpha!}{f^{|\alpha|+1}}D^\alpha f\right| \\
&\leq \frac{s!}{|f|^{|\alpha|+1}}\left|D^\alpha f\right| \\
&\leq \frac{s!}{f_0^{|\alpha|+1}}\left|D^\alpha f\right| \\
&\leq s!\left(\frac{1}{f_0}\right)^{s+1}\left|D^\alpha f\right| \\
&\leq s!\left(\frac{1}{f_0}\right)^{s+1}\|f\|_{s,\infty,\omega}.
\end{aligned}$$

The proof is concluded by noting this hold for any indexing set α. □

Theorem 5.7.3 (Consistency Error)**.**

Let $\hat{E}$ be a polygonal element of $\mathcal{T}_h$ that satisfies the mesh regularity Assumption 1 with $h_{\hat{E}}$ sufficiently small. Let $\hat{a}, \hat{\mu} \in W_{m+1}^\infty(\hat{E})$ and $\mathscr{A}, \hat{\mathbf{b}} \in \left[W_{m+1}^\infty(\hat{E})\right]^2$. The VEM bilinear forms defined in Section 5.5 satisfy for a fixed integer $s \in \{1, ..., \min\{k,l\}\}$ that

$$\begin{aligned}
\left|\mathcal{A}_h^{\hat{E}}(\hat{p}, v_h) - \mathcal{A}^{\hat{E}}(\hat{p}, v_h)\right| &\lesssim h_{\hat{E}}^s \|\hat{p}\|_{s+1,\hat{E}} \|\hat{v}_h\|_{1,\hat{E}}, \\
\left|l_h^{\hat{E}}(v_h) - l^{\hat{E}}(v_h)\right| &\lesssim h_{\hat{E}}^s \left\|\hat{f}\right\|_{s-1,\hat{E}} \|\hat{v}_h\|_{1,\hat{E}},
\end{aligned}$$

where The first hidden constant is dependent on ϱ, s, j_0, the W_{s+1}^∞ norms of $\hat{a}$, $\hat{\mathbf{b}}$ and $\hat{\mu}$ and the W_{s+1}^∞ norm of $\mathscr{A}$. The second hidden constant is dependent on ϱ, s and the W_{s+1}^∞ norm of $\mathscr{A}$.

Proof. For brevity, we copy the presentation in [39] and only present the derivation of the estimate for $A_h^{\hat{E}}(\hat{p}, v_h) - A^{\hat{E}}(\hat{p}, v_h)$ and state the remaining estimates, in each case the same line of reasoning can be pursued to attain the bounds.

By adding and subtracting terms we have for $\hat{v}_h = \hat{p}$ that the stability term and projection difference terms vanish to give

$$A_h^{\hat{E}}(\hat{p}, v_h) - A^{\hat{E}}(\hat{p}, v_h) = E1 + E2 + E3,$$

where

$$\begin{aligned} E1 &= \int_{\hat{E}} \hat{a}\mathcal{E}_{\text{cof}}\hat{\nabla}\hat{p} \cdot \mathbf{J}_{\mathscr{A}}^{-T}\hat{\nabla}\hat{v}_h \; d\boldsymbol{\xi} \\ E2 &= \int_{\hat{E}} \hat{a}\mathbf{C}_{\mathscr{A},h}\hat{\nabla}\hat{p} \cdot \mathcal{E}_{\text{inv}}^{T}\hat{\nabla}\hat{v}_h \; d\boldsymbol{\xi} \\ E3 &= \int_{\hat{E}} \hat{a}\mathbf{J}_{\mathscr{A},h}^{-T}\hat{\nabla}\hat{p} \cdot \mathbf{C}_{\mathscr{A},h}(\hat{\nabla} - \Pi_{k-1}^{1})\hat{v}_h \; d\boldsymbol{\xi}. \end{aligned}$$

By considering $E1$ and applying the Hőlder and Cauchy-Schwarz inequalities then Lemma 5.4.1 we get

$$\begin{aligned} |E1| &\leq \|\hat{a}\|_{0,\infty,\hat{E}} \|\mathcal{E}_{\text{cof}}\|_{0,\infty,\hat{E}} \left\|\mathbf{J}_{\mathscr{A}}^{-T}\right\|_{0,\infty,\hat{E}} |\hat{p}|_{1,\hat{E}} |\hat{v}_h|_{1,\hat{E}} \\ &\leq \|\hat{a}\|_{0,\infty,\hat{E}} \|\mathcal{E}_{\text{cof}}\|_{0,\infty,\hat{E}} \frac{1}{j_0} \|\mathbf{C}_{\mathscr{A}}\|_{0,\infty,\hat{E}} |\hat{p}|_{1,\hat{E}} |\hat{v}_h|_{1,\hat{E}} \,. \end{aligned}$$

Next we apply Lemma 5.4.3 to bound $|E1|$ by

$$\begin{aligned} |E1| &\leq \|\hat{a}\|_{0,\infty,\hat{E}} \left\{ C_{1,\varrho,s} h_{\hat{E}}^{s} |\mathscr{A}|_{s+1,\infty,\hat{E}} \right\} \frac{1}{j_0} \|\mathbf{C}_{\mathscr{A}}\|_{0,\infty,\hat{E}} |\hat{p}|_{1,\hat{E}} |\hat{v}_h|_{1,\hat{E}} \\ &\leq \|\hat{a}\|_{0,\infty,\hat{E}} \frac{C_{1,\varrho,s}}{j_0} h_{\hat{E}}^{s} |\mathscr{A}|_{s+1,\infty,\hat{E}} |\mathscr{A}|_{1,\infty,\hat{E}} |\hat{p}|_{1,\hat{E}} |\hat{v}_h|_{1,\hat{E}} \\ &\leq \|\hat{a}\|_{0,\infty,\hat{E}} \frac{C_{1,\varrho,s}}{j_0} h_{\hat{E}}^{s} \|\mathscr{A}\|_{s+1,\infty,\hat{E}}^{2} |\hat{p}|_{1,\hat{E}} |\hat{v}_h|_{1,\hat{E}} \,. \end{aligned}$$

We simplify this bound by introducing the constant $C_{2,\varrho,s,j_0,a,\mathscr{A}}$ as

$$C_{2,\varrho,s,j_0,a,\mathscr{A}} = \|\hat{a}\|_{0,\infty,\hat{E}} \frac{C_{1,\varrho,s}}{j_0} \|\mathscr{A}\|_{s+1,\infty,\hat{E}}^{2} \,,$$

so that $|E1|$ is bounded by

$$|E1| \leq C_{2,\varrho,s,j_0,a,\mathscr{A}} h_{\hat{E}}^{s} |\hat{p}|_{1,\hat{E}} |\hat{v}_h|_{1,\hat{E}} \,.$$

A similar argument using instead Lemma 5.4.5 provides

$$\begin{aligned}|E2| &\leq \|\hat{a}\|_{0,\infty,\hat{E}} \frac{C_{3,\varrho,j_0,s}}{j_0} h_{\hat{E}}^s \|\mathscr{A}\|_{s+1,\infty,\hat{E}}^3 |\mathscr{A}|_{1,\infty,\hat{E}} |\hat{p}|_{1,\hat{E}} |\hat{v}_h|_{1,\hat{E}} \\ &\leq \left\{ \|\hat{a}\|_{0,\infty,\hat{E}} \frac{C_{3,\varrho,j_0,s}}{j_0} \|\mathscr{A}\|_{s+1,\infty,\hat{E}}^4 \right\} h_{\hat{E}}^s |\hat{p}|_{1,\hat{E}} |\hat{v}_h|_{1,\hat{E}} \\ &\leq C_{4,\varrho,s,j_0,a,\mathscr{A}} h_{\hat{E}}^s |\hat{p}|_{1,\hat{E}} |\hat{v}_h|_{1,\hat{E}}\end{aligned}$$

Combining these bounds leads to

$$|E1| + |E2| \leq (C_{2,\varrho,s,j_0,a,\mathscr{A}} + C_{4,\varrho,s,j_0,a,\mathscr{A}}) h_{\hat{E}}^s \|\hat{p}\|_{1,\hat{E}} \|\hat{v}_h\|_{1,\hat{E}}.$$

To bound $E3$ we emulate the steps taken in [39], $E3$ can be written as

$$\begin{aligned}E3 &= \int_{\hat{E}} \hat{a} \mathbf{J}_{\mathscr{A},h}^{-T} \hat{\nabla} \hat{p} \cdot \mathbf{C}_{\mathscr{A},h} (\hat{\nabla} - \Pi_{k-1}^1) \hat{v}_h \, d\boldsymbol{\xi} \\ &= \int_{\hat{E}} \boldsymbol{\gamma} \cdot (\hat{\nabla} - \Pi_{k-1}^1) \hat{v}_h \, d\boldsymbol{\xi}\end{aligned}$$

where $\mathbf{C}_{\mathscr{A},h}$ has polynomial entries of degree $(l-1)$ and $\boldsymbol{\gamma} := (\hat{a} \mathbf{J}_{\mathscr{A},h}^{-T} \hat{\nabla} \hat{p})^T \mathbf{C}_{\mathscr{A},h}$. By orthogonality of the Π^0 projection we have

$$E3 = \int_{\hat{E}} (\boldsymbol{\gamma} - \Pi_{k-1}^0 \boldsymbol{\gamma}) \cdot (\hat{\nabla} - \Pi_{k-1}^1) \hat{v}_h \, d\boldsymbol{\xi}.$$

Then applying Cauchy-Schwarz and Theorem 2.1.1 gives

$$\begin{aligned}|E3| &\leq \left\| \boldsymbol{\gamma} - \Pi_{k-1}^0 \boldsymbol{\gamma} \right\|_{0,\hat{E}} \left\| (\hat{\nabla} - \Pi_{k-1}^1) \hat{v}_h \right\|_{0,\hat{E}} \\ &\leq C_{5,\varrho,s} h_{\hat{E}}^s \|\boldsymbol{\gamma}\|_{s,\hat{E}} \|\hat{v}_h\|_{1,\hat{E}}. \end{aligned} \tag{5.15}$$

We bound the norm $\|\boldsymbol{\gamma}\|_{s,\hat{E}}$ making use of Lemma 5.7.1. Firstly, we note that $\boldsymbol{\gamma}$ can be written as

$$\boldsymbol{\gamma} = \hat{a} \frac{1}{j_h} (\mathbf{C}_{\mathscr{A},h} \hat{\nabla} \hat{p})^\top \mathbf{C}_{\mathscr{A},h}.$$

From repeated applications of Lemma 5.7.1 and Lemma 5.7.2 it holds that

$$\begin{aligned}\|\boldsymbol{\gamma}\|_{s,\hat{E}} &\leq C_{6,s} \|\hat{a}\|_{s,\infty,\hat{E}} \left\| \frac{1}{j_h} \right\|_{s,\infty,\hat{E}} \|\mathbf{C}_{\mathscr{A},h}\|_{s,\infty,\hat{E}}^2 \left\| \hat{\nabla} \hat{p} \right\|_{s,\hat{E}} \\ &\leq C_{6,s} C_{7,s,j_0} \|\hat{a}\|_{s,\infty,\hat{E}} \|j_h\|_{s,\infty,\hat{E}} \|\mathbf{C}_{\mathscr{A},h}\|_{s,\infty,\hat{E}}^2 \left\| \hat{\nabla} \hat{p} \right\|_{s,\hat{E}}. \end{aligned} \tag{5.16}$$

To bound the norms of j_h and $\mathbf{C}_{\mathscr{A},h}$ we apply an inverse Sobolev embedding, noting that both terms are polynomials of degree $2l-2$ and $l-1$ respectively.

Here we present the first case with j_h with the norm $\mathbf{C}_{\mathscr{A},h}$ following in a similar fashion. By adding and subtracting $\Pi^0_{2l-2}j$ and applying the triangle inequality we have

$$\begin{aligned}\|j_h\|_{s,\infty,\hat{E}} &\leq \left\|j_h - \Pi^0_{2l-2}j\right\|_{s,\infty,\hat{E}} + \left\|\Pi^0_{2l-2}j\right\|_{s,\infty,\hat{E}} \\ &= \left\|\Pi^0_{2l-2}\mathcal{E}_{\text{det}}\right\|_{s,\infty,\hat{E}} + \left\|\Pi^0_{2l-2}j\right\|_{s,\infty,\hat{E}}.\end{aligned}$$

From [49] (Corollary 1.29) it holds that

$$\|j_h\|_{s,\infty,\hat{E}} \leq C_{8,s}h^{-s}_{\hat{E}}\left\|\Pi^0_{2l-2}\mathcal{E}_{\text{det}}\right\|_{0,\infty,\hat{E}} + \left\|\Pi^0_{2l-2}j\right\|_{s,\infty,\hat{E}}. \tag{5.17}$$

The stability of the Π^0 projection is then applied to both terms in Equation (5.17) to give

$$\|j_h\|_{s,\infty,\hat{E}} \leq C_{8,s}h^{-s}_{\hat{E}}\left\|\mathcal{E}_{\text{det}}\right\|_{0,\infty,\hat{E}} + \|j\|_{s,\infty,\hat{E}},$$

Lemma 5.4.4 is then applied to the first term and the second term is bounded using Lemma 5.7.1

$$\|j_h\|_{s,\infty,\hat{E}} \leq \left(C_{8,s}C_{9,\varrho,s}+2\right)\|\mathscr{A}\|^2_{s+1,\infty,\hat{E}}. \tag{5.18}$$

Using again Corollary 1.29 from [49] and Lemma 5.4.3 gives a positive constant C_2 dependent on the shape regularity of $\hat{E}$ and s such that

$$\|\mathbf{C}_{\mathscr{A},h}\|_{s,\infty,\hat{E}} \leq C_{10,\varrho,s}\|\mathscr{A}\|_{s+1,\infty,\hat{E}}. \tag{5.19}$$

Inserting Equations (5.18) and (5.19) into Equation (5.16) gives

$$\begin{aligned}\|\boldsymbol{\gamma}\|_{s,\hat{E}} &\leq C_{6,s}C_{7,s,j_0}\|\hat{a}\|_{s,\infty,\hat{E}}\left(C_{8,s}C_{9,\varrho,s}+2\right)C^2_{10,\varrho,s}\|\mathscr{A}\|^4_{s+1,\infty,\hat{E}}\left\|\hat{\nabla}\hat{p}\right\|_{s,\hat{E}} \\ &\leq \left\{C_{6,s}C_{7,s,j_0}\|\hat{a}\|_{s,\infty,\hat{E}}\left(C_{8,s}C_{9,\varrho,s}+2\right)C^2_{10,\varrho,s}\|\mathscr{A}\|^4_{s+1,\infty,\hat{E}}\right\}\|\hat{p}\|_{s+1,\hat{E}} \\ &\leq C_{11,\varrho,s,j_0,a,\mathscr{A}}\|\hat{p}\|_{s+1,\hat{E}}.\end{aligned}$$

Inserting this bound into Equation (5.15) and combining the bounds for $E1$,

$E2$ and $E3$ we have that

$$\begin{aligned}\left|A_h^{\hat{E}}(\hat{p},v_h)-A^{\hat{E}}(\hat{p},v_h)\right| &\leq (C_{2,\varrho,s,j_0,a,\mathscr{A}}+C_{4,\varrho,s,j_0,a,\mathscr{A}})h_{\hat{E}}^s\left\|\hat{p}\right\|_{1,\hat{E}}\left\|\hat{v}_h\right\|_{1,\hat{E}}\\ &\quad+C_{5,\varrho,s}h_{\hat{E}}^s\left\|\boldsymbol{\gamma}\right\|_{s,\hat{E}}\left\|\hat{v}_h\right\|_{1,\hat{E}}\\ &\leq (C_{2,\varrho,s,j_0,a,\mathscr{A}}+C_{4,\varrho,s,j_0,a,\mathscr{A}})h_{\hat{E}}^s\left\|\hat{p}\right\|_{1,\hat{E}}\left\|\hat{v}_h\right\|_{1,\hat{E}}\\ &\quad+(C_{5,\varrho,s}C_{11,\varrho,s,j_0,a,\mathscr{A}})h_{\hat{E}}^s\left\|\hat{p}\right\|_{s+1,\hat{E}}\left\|\hat{v}_h\right\|_{1,\hat{E}}\\ &\leq C_{12,\varrho,s,j_0,a,\mathscr{A}}h_{\hat{E}}^s\left\|\hat{p}\right\|_{s+1,\hat{E}}\left\|\hat{v}_h\right\|_{1,\hat{E}}\end{aligned}$$

A similar line of reasoning provides the additional constants with the similar dependencies such that

$$\begin{aligned}\left|B_h^{\hat{E}}(\hat{p},v_h)-B^{\hat{E}}(\hat{p},v_h)\right| &\leq C_{13,\varrho,s,j_0,\mathbf{b},\mathscr{A}}h_{\hat{E}}^s\left\|\hat{p}\right\|_{s+1,\hat{E}}\left\|\hat{v}_h\right\|_{1,\hat{E}},\\ \left|C_h^{\hat{E}}(\hat{p},v_h)-C^{\hat{E}}(\hat{p},v_h)\right| &\leq C_{14,\varrho,s,j_0,\mu,\mathscr{A}}h_{\hat{E}}^s\left\|\hat{p}\right\|_{s+1,\hat{E}}\left\|\hat{v}_h\right\|_{1,\hat{E}},\\ \left|l_h^{\hat{E}}(v_h)-l^{\hat{E}}(v_h)\right| &\leq C_{15,\varrho,s,j_0,\mathscr{A}}h_{\hat{E}}^s\left\|\hat{f}\right\|_{s-1,\hat{E}}\left\|\hat{v}_h\right\|_{1,\hat{E}},\end{aligned}$$

where in the final bound the estimate of approximating $\hat{f}$ is used. Applying the triangle inequality and combining the bounds completes the proof. □

5.8 A H^1 Estimate

We now prove the H^1 estimate for Method I using the framework outlined in [39]. Let $C_{1,\varrho,a_0,\mu_0,l,\mathscr{A}}, C_{2,\varrho,l,a,\mathbf{b},\mu,\mathscr{A}} > 0$ denote the coercivity and continuity constants of this method derived in Lemmas 5.6.2 and 5.6.1 respectively. We consider a polynomial $\hat{p} \in \mathbb{P}_k(\hat{\mathcal{T}}_h)$, where $\mathbb{P}_k(\hat{\mathcal{T}}_h)$ is the space of polynomials of degree k that are discontinuous across element faces and $\hat{w}_h = \hat{\rho}_h - \hat{v}_h$. The Strang-type bound is given by

$$\begin{aligned}\frac{1}{C_{1,\varrho,a_0,\mu_0,l,\mathscr{A}}}\left\|\hat{\rho}_h-\hat{v}_h\right\|_{1,\hat{\Omega}}^2 \leq [l_h(\hat{w}_h)-l(\hat{w}_h)]+\sum_{\hat{E}\in\hat{\mathcal{T}}_h}\left(\mathcal{A}_h^{\hat{E}}(\hat{p},\hat{w}_h)-\mathcal{A}^{\hat{E}}(\hat{p},\hat{w}_h)\right)\\ +\mathcal{A}_h(\hat{p}-\hat{v}_h,\hat{w}_h)+\mathcal{A}(\hat{\rho}-\hat{p},\hat{w}_h).\end{aligned}$$

Applying the continuity of $\mathcal{A}(\cdot,\cdot)$ on the reference domain and the continuity of $\mathcal{A}_h(\cdot,\cdot)$ from Lemma 5.6.1 provides

$$\frac{1}{C_{1,\varrho,a_0,\mu_0,l,\mathscr{A}}}\left\|\hat{\rho}_h-\hat{v}_h\right\|_{1,\hat{\Omega}}^2 \leq \left[l_h(\hat{w}_h)-l(\hat{w}_h)\right]+\sum_{\hat{E}\in\hat{\mathcal{T}}_h}\left(\mathcal{A}_h^{\hat{E}}(\hat{p},\hat{w}_h)-\mathcal{A}^{\hat{E}}(\hat{p},\hat{w}_h)\right) + \left\|\hat{w}_h\right\|_{1,\hat{\Omega}}\left(C_{2,\varrho,l,a,\mathbf{b},\mu,\mathscr{A}}\left\|\hat{p}-\hat{v}_h\right\|_{1,\hat{\Omega}}+C_{3,\mathscr{A}}\left\|\hat{\rho}-\hat{p}\right\|_{1,\hat{\Omega}}\right).$$

We introduce the constant $C_{coer,\varrho,a_0,\mu_0,l,a,\mathbf{b},\mu,\mathscr{A}}$ by

$$C_{coer,\varrho,a_0,\mu_0,l,a,\mathbf{b},\mu,\mathscr{A}}=\max\left\{C_{1,\varrho,a_0,\mu_0,l,\mathscr{A}},C_{2,\varrho,l,a,\mathbf{b},\mu,\mathscr{A}},C_{3,\mathscr{A}},1\right\}. \tag{5.20}$$

As a result we have

$$\frac{1}{C_{coer,\varrho,a_0,\mu_0,l,a,\mathbf{b},\mu,\mathscr{A}}}\left\|\hat{\rho}_h-\hat{v}_h\right\|_{1,\hat{\Omega}}^2 \leq \left[l_h(\hat{w}_h)-l(\hat{w}_h)\right]+\sum_{\hat{E}\in\hat{\mathcal{T}}_h}\left(\mathcal{A}_h^{\hat{E}}(\hat{p},\hat{w}_h)-\mathcal{A}^{\hat{E}}(\hat{p},\hat{w}_h)\right) + \left\|\hat{w}_h\right\|_{1,\hat{\Omega}}\left(\left\|\hat{p}-\hat{v}_h\right\|_{1,\hat{\Omega}}+\left\|\hat{\rho}-\hat{p}\right\|_{1,\hat{\Omega}}\right).$$

Dividing through by $\left\|\hat{w}_h\right\|_{1,\hat{\Omega}}$ and applying the triangle inequality gives

$$\begin{aligned}\frac{1}{C_{coer,\varrho,a_0,\mu_0,l,a,\mathbf{b},\mu,\mathscr{A}}}\left\|\rho-\hat{\rho}_h\right\|_{1,\hat{\Omega}} \leq & \inf_{\hat{w}_h\in\hat{V}_{h,0}}\left\|\hat{\rho}-\hat{w}_h\right\|_{1,\hat{\Omega}}+\inf_{\hat{p}\in\mathbb{P}_k(\hat{\mathcal{T}}_h)}\left\|\hat{\rho}-\hat{p}\right\|_{1,\hat{\Omega}} \\ & +\inf_{\hat{p}\in\mathbb{P}_k(\hat{\mathcal{T}}_h)}\sum_{\hat{E}\in\hat{\mathcal{T}}_h}\sup_{\hat{w}_h\in\hat{V}_{h,0}\setminus\{0\}}\frac{\left|\mathcal{A}_h^{\hat{E}}(\hat{p},\hat{w}_h)-\mathcal{A}^{\hat{E}}(\hat{p},\hat{w}_h)\right|}{\left\|\hat{w}_h\right\|_{1,\hat{E}}} \\ & +\sup_{\hat{w}_h\in\hat{V}_{h,0}\setminus\{0\}}\frac{\left|l_h(\hat{w}_h)-l(\hat{w}_h)\right|}{\left\|\hat{w}_h\right\|_{1,\hat{\Omega}}}.\end{aligned} \tag{5.21}$$

Theorem 5.8.1 (H^1 Error Estimate for Method I)**.**

Let $\mathscr{A}\in W_{m+1}^{\infty}(\hat{\Omega})$ be a mapping from a polygonal reference domain $\hat{\Omega}$ to Ω. Additionally, suppose the coefficients satisfy the regularity conditions of $\hat{a},\hat{\mathbf{b}},\hat{\mu}\in W_{m+1}^{\infty}(\Omega)$. For the corresponding solution to the PDE (5.1) on the reference domain $\hat{\rho}\in H^{m+1}(\hat{\Omega})$ and the VEM solution $\rho_h\in\hat{V}_{h,0}$ of degree l and k outlined in Section 5.5, we have for sufficiently small mesh size h and $s\in\{0,1,\ldots\min\{k,l\}\}$ that

$$\left\|\hat{\rho}-\hat{\rho}_h\right\|_{1,\hat{\Omega}}\lesssim h^s\left(\left\|\hat{\rho}\right\|_{s+1,\hat{\Omega}}+\left\|\hat{f}\right\|_{s-1,\hat{\Omega}}\right),$$

where the hidden constant is dependent on ϱ, s, l, the lower bounds j_0, a_0, μ_0 and the W_{s+1}^{∞} norms of $\hat{a}$, $\hat{\mathbf{b}}$, $\hat{\mu}$ and $\mathscr{A}$.

Proof. For ease of reading, we introduce the constants C, dependent on the mesh regularity and s, that implicitly bounds all constants independent of $\mathscr{A}$ in this proof. We bound each line of Equation (5.21) in order. By considering $\hat{w}_h = \hat{\rho}_I$ and $\hat{p} = \Pi_k^0 \hat{\rho}$ from Theorems 2.1.2 and 2.1.1 we have that

$$\inf_{\hat{w}_h \in \hat{V}_{h,0}} \|\hat{\rho} - \hat{w}_h\|_{1,\hat{\Omega}} + \inf_{\hat{p} \in \mathbb{P}_k(\hat{\mathcal{T}}_h)} \left\|\hat{\rho} - \Pi_k^0 \hat{\rho}\right\|_{1,\hat{\Omega}} \leq C_{1,\varrho,s} h^s \|\hat{\rho}\|_{s+1,\hat{\Omega}},$$

For the second line we consider $\hat{p} = \Pi_k^0 \hat{\rho}$ and apply Theorem 5.7.3 to get

$$\begin{aligned}
&\inf_{\hat{p} \in \mathbb{P}_k(\hat{\mathcal{T}}_h)} \sum_{\hat{E} \in \hat{\mathcal{T}}_h} \sup_{\hat{w}_h \in \hat{V}_{h,0} \setminus \{0\}} \frac{\left|\mathcal{A}_h^{\hat{E}}(\hat{p}, \hat{w}_h) - \mathcal{A}^{\hat{E}}(\hat{p}, \hat{w}_h)\right|}{\|\hat{w}_h\|_{1,\hat{E}}} \\
&\leq C_{2,\varrho,s,a,\mathbf{b},\mu,j_0,\mathscr{A}} \sum_{\hat{E} \in \hat{\mathcal{T}}_h} h_{\hat{E}}^s \left\|\Pi_k^0 \hat{\rho}\right\|_{s+1,\hat{E}} \\
&\leq C_{2,\varrho,s,a,\mathbf{b},\mu,j_0,\mathscr{A}} h^s \|\hat{\rho}\|_{s+1,\hat{\Omega}},
\end{aligned}$$

where in the final line we applied the stability of the Π^0 projection and bounded norms. Next, we have from Theorem 5.7.3 that

$$\begin{aligned}
\frac{|l_h(\hat{w}_h) - l(\hat{w}_h)|}{\|\hat{w}_h\|_{1,\hat{\Omega}}} &\leq \sum_{\hat{E} \in \hat{\mathcal{T}}_h} \frac{\left|l_h^{\hat{E}}(\hat{w}_h) - l^{\hat{E}}(\hat{w}_h)\right|}{\|\hat{w}_h\|_{1,\hat{E}}} \\
&\leq C_{3,\varrho,s,\mathscr{A}} \sum_{\hat{E} \in \hat{\mathcal{T}}_h} h_{\hat{E}}^s \left\|\hat{f}\right\|_{s-1,\hat{E}} \\
&\leq C_{3,\varrho,s,\mathscr{A}} h^s \left\|\hat{f}\right\|_{s-1,\hat{\Omega}}.
\end{aligned}$$

Combining the three bounds and the bounding constants and inserting these into Equation (5.21) along with the constant defined in Equation (5.20) we have that

$$\begin{aligned}
\|\hat{\rho} - \hat{\rho}_h\|_{1,\hat{\Omega}} \leq C_{coer,\varrho,a_0,\mu_0,l,a,\mathbf{b},\mu,\mathscr{A}} \big(&C_{1,\varrho,s} + C_{2,\varrho,s,a,\mathbf{b},\mu,j_0,\mathscr{A}} \\
&+ C_{3,\varrho,s,\mathscr{A}}\big) h^s \left(\|\hat{\rho}\|_{s+1,\hat{\Omega}} + \left\|\hat{f}\right\|_{s-1,\hat{\Omega}}\right).
\end{aligned}$$

We conclude the proof by further bounding the constant terms. □

Chapter 6

Method II

The second of our proposed isoparametric methods is to construct virtual, physical curved edge polygons. This is achieved by constructing a curved VEM space on each virtual polygon using the framework of curved virtual elements proposed in [23] and perform the necessary integration by considering a set of projected elements using the Π^0 operator on computational reference domain $\hat{\Omega}$.

In Section 6.1, the projected element used in the computation of Method II is introduced. The curved VEM space used in Method II is reviewed in Section 6.2. The discrete projection operators are introduced and analysed in Section 6.3. The complete formulation of Method II is given in Section 6.4. The consistency error and well-posedness of Method II are proven in Sections 6.5 and 6.6 respectively. The proof of the H^1 *a priori* estimate is given in Section 6.7.

6.1 Projected Elements

In Method II, the integration over the virtual domain is decomposed into contributions over projected elements. By using a projection of an element, we have a computable geometry on which we can integrate polynomials exactly. In keeping with the VEM philosophy, we decompose $\mathscr{A}_h$ into a polynomial and

non-polynomial component, namely

$$\mathscr{A}_h = \Pi_l^0 \mathscr{A}_h + (\mathscr{A}_h - \Pi_l^0 \mathscr{A}_h),$$

The projected element $E_h^\pi := \Pi_l^0 \mathscr{A}_h(\hat{E})$ will be used to construct an explicitly computable method.

Remark 12. *In constructing the projected elements we require the use of a computational reference domain. However, the method is independent of* $\hat{\Omega}$ *and the resulting linear systems of equations are purely computed without coordinate transformations.*

We utilise the projected elements to provide a computable method. Local VEM spaces will be constructed on E_h and not E_h^π with E_h^π only being used to perform quadrature. We quantify the effects of using the projected elements for integration in the following technical results.

Lemma 6.1.1 (Area Approximation for the Projected Element)**.**
Let $\hat{\mathcal{T}}_h$ *be the computational reference domain that satisfies Assumption 1. For a reference element* $\hat{E} \in \hat{\mathcal{T}}_h$ *and the corresponding element on the virtual domain* $E_h \in \mathcal{T}_h$ *and a projected element* E_h^π *of degree* l*, it holds for a fixed integer* $s \in \{0, 1, ..., l\}$ *and for a sufficiently small mesh size that*

$$\begin{aligned} |meas(E_h) - meas(E_h^\pi)| &\lesssim h_{\hat{E}}^{s+2} \left\| \mathscr{A} \right\|_{s+1,\infty,\hat{E}}^2, \\ meas(E_h \backslash E_h^\pi) &\lesssim h_{\hat{E}}^{s+2} \left\| \mathscr{A} \right\|_{s+1,\infty,\hat{E}}^3, \\ meas(E_h^\pi \backslash E_h) &\lesssim h_{\hat{E}}^{s+2} \left\| \mathscr{A} \right\|_{s+1,\infty,\hat{E}}^3, \end{aligned}$$

where the hidden constants depend on ϱ *and* l*.*

Proof. By definition we have that

$$\text{meas}(E_h) - \text{meas}(E_h^\pi) = \left| \int_{\hat{E}} \det \left(\hat{\nabla} \mathscr{A}_h \right) \, d\boldsymbol{\xi} \right| - \left| \int_{\hat{E}} \det \left(\hat{\nabla} \Pi_l^0 \mathscr{A}_h \right) \, d\boldsymbol{\xi} \right|.$$

We apply the arguments of Lemma 5.4.5 to imply that for a sufficiently small $h_{\hat{E}}$, it holds that both determinants are strictly positive

$$\text{meas}(E_h) - \text{meas}(E_h^\pi) = \int_{\hat{E}} \det \left(\hat{\nabla} \mathscr{A}_h \right) - \det \left(\hat{\nabla} \Pi_l^0 \mathscr{A}_h \right) \, d\boldsymbol{\xi}.$$

Hölder's inequality gives

$$|\text{meas}(E_h) - \text{meas}(E_h^\pi)| \leq \left|\hat{E}\right| \left\|\det\left(\hat{\nabla}\mathscr{A}_h\right) - \det\left(\hat{\nabla}\Pi_l^0\mathscr{A}_h\right)\right\|_{0,\infty,\hat{E}}$$
$$\leq h_{\hat{E}}^2 \left\|\det\left(\hat{\nabla}\mathscr{A}_h\right) - \det\left(\hat{\nabla}\Pi_l^0\mathscr{A}_h\right)\right\|_{0,\infty,\hat{E}}. \quad (6.1)$$

Next we add and subtract terms and apply the triangle inequality to get

$$\left\|\det\left(\hat{\nabla}\mathscr{A}_h\right) - \det\left(\hat{\nabla}\Pi_l^0\mathscr{A}_h\right)\right\|_{0,\infty,\hat{E}} \leq \left\|\det\left(\hat{\nabla}\mathscr{A}_h\right) - \det\left(\hat{\nabla}\mathscr{A}\right)\right\|_{0,\infty,\hat{E}}$$
$$+ \left\|\det\left(\hat{\nabla}\mathscr{A}\right) - \det\left(\hat{\nabla}\Pi_l^0\mathscr{A}\right)\right\|_{0,\infty,\hat{E}}$$
$$+ \left\|\det\left(\hat{\nabla}\Pi_l^0\mathscr{A}\right) - \det\left(\hat{\nabla}\Pi_l^0\mathscr{A}_h\right)\right\|_{0,\infty,\hat{E}}.$$

Following the same steps as in Lemma 5.4.4 for each of the three terms in the above bound we have

$$\left\|\det\left(\hat{\nabla}\mathscr{A}_h\right) - \det\left(\hat{\nabla}\Pi_l^0\mathscr{A}_h\right)\right\|_{0,\infty,\hat{E}} \leq C_{1,\varrho,l} h_{\hat{E}}^s \left\|\mathscr{A}\right\|_{s+1,\infty,\hat{E}}^2,$$

where substitution of this into Equation (6.1) completes the first estimate.

The second estimate is proved by introducing the indicator function $\chi_{E_h^\pi} : \mathbb{R}^2 \to \{0,1\}$

$$\chi_{E_h^\pi}(\mathbf{x}) = \begin{cases} 1 & \mathbf{x} \in E_h^\pi, \\ 0 & \text{otherwise} \end{cases},$$

in which we note that $\left\|\chi_{E_h^\pi}\right\|_{op} \leq 1$. We write $\text{meas}(E_h \backslash E_h^\pi)$ as

$$\text{meas}(E_h \backslash E_h^\pi) = \int_{E_h} \left(1 - \chi_{E_h^\pi}\right) \, d\mathbf{x}$$
$$= \int_{\hat{E}} \left(1 - \chi_{E_h^\pi} \circ \mathscr{A}_h(\boldsymbol{\xi})\right) \det\left(\mathbf{J}_{\mathscr{A}_h}\right) \, d\boldsymbol{\xi}.$$

Applying the Cauchy-Schwarz inequality and the operator norm inequality for $\chi_{E_h^\pi}$ gives

$$\text{meas}(E_h \backslash E_h^\pi) \leq \left\|\det\left(\mathbf{J}_{\mathscr{A}_h}\right)\right\|_{0,\hat{E}} \left\|1 - \chi_{E_h^\pi} \circ \mathscr{A}_h\right\|_{0,\hat{E}}$$
$$\leq \left\|\det\left(\mathbf{J}_{\mathscr{A}_h}\right)\right\|_{0,\hat{E}} \left\|\chi_{E_h^\pi} \circ \Pi_l^0\mathscr{A}_h - \chi_{E_h^\pi} \circ \mathscr{A}_h\right\|_{0,\hat{E}}$$
$$\leq \left\|\det\left(\mathbf{J}_{\mathscr{A}_h}\right)\right\|_{0,\hat{E}} \left\|\chi_{E_h^\pi}\right\|_{op} \left\|\Pi_l^0\mathscr{A}_h - \mathscr{A}_h\right\|_{0,\hat{E}}$$
$$\leq \left\|\det\left(\mathbf{J}_{\mathscr{A}_h}\right)\right\|_{0,\hat{E}} \left\|\Pi_l^0\mathscr{A}_h - \mathscr{A}_h\right\|_{0,\hat{E}}$$

Next we apply Hölder's inequality with Lemma 5.4.4 to get

$$\begin{aligned}\operatorname{meas}(E_h \backslash E_h^\pi) &\leq \left|\hat{E}\right|^{1/2} \left\|\det\left(\mathbf{J}_{\mathscr{A}_h}\right)\right\|_{0,\infty,\hat{E}} \left\|\Pi_l^0 \mathscr{A}_h - \mathscr{A}_h\right\|_{0,\hat{E}} \\ &\leq C_{2,\varrho,l} h_{\hat{E}} \left\|\mathscr{A}\right\|_{1,\infty,\hat{E}}^2 \left\|\Pi_l^0 \mathscr{A}_h - \mathscr{A}_h\right\|_{0,\hat{E}}. \end{aligned} \tag{6.2}$$

The projection error is bounded by using the triangle inequality, the stability of Π^0, Assumption 4 and Theorem 2.1.1

$$\begin{aligned}\left\|\Pi_l^0 \mathscr{A}_h - \mathscr{A}_h\right\|_{0,\hat{E}} &\leq \left\|\Pi_l^0 \mathscr{A}_h - \Pi_l^0 \mathscr{A}\right\|_{0,\hat{E}} + \left\|\Pi_l^0 \mathscr{A} - \mathscr{A}\right\|_{0,\hat{E}} \\ &\leq \left\|\mathscr{A}_h - \mathscr{A}\right\|_{0,\hat{E}} + \left\|\Pi_l^0 \mathscr{A} - \mathscr{A}\right\|_{0,\hat{E}} \\ &\leq C_{3,\varrho,l} h_{\hat{E}}^s \left|\mathscr{A}\right|_{s+1,\hat{E}} \\ &\leq C_{3,\varrho,l} h_{\hat{E}}^{s+1} \left|\mathscr{A}\right|_{s+1,\infty,\hat{E}}. \end{aligned}$$

Inserting this bound into Equation (6.2) and bounding norms gives

$$\operatorname{meas}(E_h \backslash E_h^\pi) \leq C_{2,\varrho,l} C_{3,\varrho,l} h_{\hat{E}}^{s+2} \left|\mathscr{A}\right|_{s+1,\infty,\hat{E}}^3.$$

The final estimate follows by the same arguments. □

When quantifying the error of this method, we have to consider the scaling of the mesh size terms h_ω for a choice of $\omega = \hat{E}, E_h, E_h^\pi$. This is presented in the following theorem.

Lemma 6.1.2 (Scaling of Mesh Sizes)**.**
Let $\hat{\mathcal{T}}_h$ be a mesh of $\hat{\Omega}$ of sufficiently small mesh size h satisfying the shape regularity Assumption 1. We have

$$h_{\hat{E}} \lesssim h_{E_h}, \; h_{\hat{E}} \gtrsim h_{E_h}$$

where the first hidden constant depends on j_0 and $\varrho_{\hat{E}}$ and the second depends on the W_1^∞ norm of $\mathscr{A}$ and ϱ_{E_h}.

Proof. From [49] (Lemma 1.12) it holds for any shape regular element E that

$$\varrho_E^2 \pi h_E^2 \leq |E| \leq h_E^2. \tag{6.3}$$

Using Equation (6.3) and Lemma 5.4.5 we have

$$\begin{aligned} j_0 \varrho_E^2 \pi h_{\hat{E}}^2 &\leq j_0 \left|\hat{E}\right| \\ &\leq |E_h| \\ &\leq h_{E_h}^2. \end{aligned}$$

Taking square roots completes the lower bound. The upper bound is given by computing the area of E_h using $\hat{E}$ and applying Hölder's inequality

$$\begin{aligned} |E_h| &= \left\| \det\left(\hat{\nabla} \mathscr{A}_h\right) \right\|_{0,1,\hat{E}} \\ &\leq \left\| \det\left(\hat{\nabla} \mathscr{A}_h\right) \right\|_{0,\infty,\hat{E}} \left|\hat{E}\right| \\ &\leq 2 \left|\mathscr{A}_h\right|_{1,\infty,\hat{E}}^2 \left|\hat{E}\right|. \end{aligned}$$

From this we use Equation (6.3) to get

$$\varrho_{E_h}^2 \pi h_{E_h}^2 \leq 2 \left|\mathscr{A}_h\right|_{1,\infty,\hat{E}}^2 h_{\hat{E}}^2.$$

The proof is concluded by using a stability argument to bound $|\mathscr{A}_h|_{1,\infty,\hat{E}}^2$. □

Remark 13. *We note that if the computational reference domain $\hat{\Omega}$ is defined as being "close to" Ω then the element sizing scales as one. An example of this is when $\hat{\Omega}$ is a polygonal interpolant of Ω. The generality provided by Lemma 6.1.2 allows us to extend the analysis of Part II to moving domains, where in general the mesh size varies over time.*

Extension of Polynomials

In the case of polynomials on E_h, the extension $\tilde{p}$ for some $p \in \mathbb{P}_k(E_h)$ is defined as the *natural* extension obtained by preserving the polynomial coefficients of p and evaluating p over $\mathbb{R}^2$, in these cases we drop the extension notation. We emphasise that this extension does not satisfy Stein's Extension Theorem 5.3.1. Instead we have the following polynomial extension bound.

Lemma 6.1.3 (Polynomial Extension)**.**

Let $\hat{\mathcal{T}}_h$ be a computational reference domain satisfying Assumption 1, $E_h \in$

$\mathcal{T}_h$ be shape regular and E_h^π be the projected element of degree $l \in \mathbb{N}$. For a polynomial of any degree $p \in \mathbb{P}(E_h)$ and $h_{\hat{E}}$ sufficiently small, it holds that

$$\|p\|_{0,\infty,E_h^\pi} \lesssim \|p\|_{0,\infty,E_h},$$

where the hidden constant depends on the shape regularity parameters ϱ, ϱ_{E_h}, l and the W_1^∞ norm of $\mathscr{A}$.

Proof. In this proof we consider the ratio between the norms and prove that this scales as one. Let R be given as

$$R = \frac{\|p\|_{0,\infty,E_h^\pi}}{\|p\|_{0,\infty,E_h}}.$$

Applying an inverse estimate [49] to the numerator and a Sobolev inequality to the denominator of R gives

$$R \leq C_{1,\varrho,l} \frac{h_{E_h^\pi}^{-1} \|p\|_{0,E_h^\pi}}{h_{E_h}^{-1} \|p\|_{0,E_h}} \leq C_{2,\varrho,l} \frac{\|p\|_{0,E_h^\pi}}{\|p\|_{0,E_h}},$$

where we applied the property of $h_{E_h^\pi}, h_{E_h} \sim h_{\hat{E}}$ from Lemma 6.1.2. We consider the bound of R^2 and expand the numerator to get

$$\begin{aligned} R^2 &\leq C_{2,\varrho,l}^2 \frac{\|p\|_{0,E_h^\pi}^2}{\|p\|_{0,E_h}^2} = C_{2,\varrho,l}^2 \left(\frac{\|p\|_{0,E_h}^2 - \|p\|_{0,E_h \setminus E_h^\pi}^2 + \|p\|_{0,E_h^\pi \setminus E_h}^2}{\|p\|_{0,E_h}^2} \right) \\ &\leq C_{2,\varrho,l}^2 \left(1 + \frac{\|p\|_{0,E_h^\pi \setminus E_h}^2}{\|p\|_{0,E_h}^2} \right). \end{aligned} \tag{6.4}$$

The ratio of norms in the bound above can be further bounded by applying Hölder's inequality on the numerator and an inverse estimate [49] on the denominator to give

$$\frac{\|p\|_{0,E_h^\pi \setminus E_h}^2}{\|p\|_{0,E_h}^2} \leq C_{3,\varrho,l} R^2 \frac{|E_h^\pi \setminus E_h|}{|E_h|}.$$

Using Lemmas 6.1.1 and 6.1.2 we get

$$\frac{\|p\|_{0,E_h^\pi \setminus E_h}^2}{\|p\|_{0,E_h}^2} \leq C_{4,\varrho,\varrho_{E_h},\mathscr{A},l} R^2 h_{\hat{E}}^s.$$

Substitution into Equation (6.4) provides

$$R^2 \leq C_{2,\varrho,l}^2 + C_{2,\varrho,l}^2 C_{4,\varrho,\varrho_{E_h},\mathscr{A},l} R^2 h_{\hat{E}},$$

therefore taking a $h_{\hat{E}}$ such that

$$h_{\hat{E}} \leq \left(C_{2,\varrho,l}^2 C_{4,\varrho,\varrho_{E_h},\mathscr{A},l} \right)^{-1},$$

results in

$$\frac{1}{2} R^2 \leq C_{2,\varrho,l}.$$

Bounding constant terms and taking square roots completes the proof. □

Quadrature Error

In Method II we choose to use the projected elements to approximate the integration of polynomials over E_h. In two dimensions integration could be performed over an isoparametric triangulation of E_h. However, in three dimensions the projected elements become a necessity and the following presentation easily extends to three-dimensional problems. The accuracy of performing quadrature over these projected elements is given in the following lemma for two dimensions.

Lemma 6.1.4 (Integration Error Over a Projected Element)**.**
Let E be a physical element given by the mapping $E = \mathscr{A}(\hat{E})$, with its corresponding interpolated element E_h and the projected element E_h^π. For $g \in L^\infty(E)$ and p, q being polynomials of any degree, the following holds for a fixed integer $s \in \{0, 1, ..., l\}$ and sufficiently small $h_{\hat{E}}$

$$\left| \int_{E_h} \tilde{g} p q \, d\mathbf{x} - \int_{E_h^\pi} \tilde{g} p q \, d\mathbf{x} \right| \lesssim h_{\hat{E}}^s \left\| p \right\|_{0,E_h} \left\| q \right\|_{0,E_h},$$

where the hidden constant depends on ϱ, l, the L^∞ norm of g, the W_{s+1}^∞ norm $\mathscr{A}$ and j_0.

Proof. We decompose the integral difference into

$$\int_{E_h} \tilde{g} p q \, d\mathbf{x} - \int_{E_h^\pi} \tilde{g} p q \, d\mathbf{x} = \int_{E_h \setminus E_h^\pi} \tilde{g} p q \, d\mathbf{x} - \int_{E_h^\pi \setminus E_h} \tilde{g} p q \, d\mathbf{x},$$

from which we have

$$\left|\int_{E_h} \tilde{g}pq \; d\mathbf{x} - \int_{E_h^\pi} \tilde{g}pq \; d\mathbf{x}\right| \leq \|\tilde{g}\|_{0,\infty,E_h\triangle E_h^\pi} \|p\|_{0,\infty,E_h\triangle E_h^\pi} \|q\|_{0,\infty,E_h\triangle E_h^\pi} \operatorname{meas}(E_h\triangle E_h^\pi), \tag{6.5}$$

where we denote by $E_h\triangle E_h^\pi = (E_h\backslash E_h^\pi) \cup (E_h^\pi\backslash E_h)$ the symmetric difference between the two sets. We bound the measure of this set using Lemma 6.1.1

$$\begin{aligned}\operatorname{meas}(E_h\triangle E_h^\pi) &\leq \operatorname{meas}(E_h\backslash E_h^\pi) + \operatorname{meas}(E_h^\pi\backslash E_h) \\ &\leq C_{1,\varrho,l} h_{\hat{E}}^{s+2} \|\mathscr{A}\|_{s+1,\infty,\hat{E}}^3 .\end{aligned}$$

Substitution back into Equation (6.5) and Applying Stein's Extension Theorem 5.3.1 for $\tilde{g}$ gives

$$\begin{aligned}\left|\int_{E_h} \tilde{g}pq \; d\mathbf{x} - \int_{E_h^\pi} \tilde{g}pq \; d\mathbf{x}\right| &\leq \|\tilde{g}\|_{0,\infty,E_h\triangle E_h^\pi} \|p\|_{0,\infty,E_h\triangle E_h^\pi} \\ &\qquad \|q\|_{0,\infty,E_h\triangle E_h^\pi} C_{1,\varrho,l} h_{\hat{E}}^{s+2} \|\mathscr{A}\|_{s+1,\infty,\hat{E}}^3 \\ &\leq C_2 \|g\|_{0,\infty,E} \|p\|_{0,\infty,E_h\triangle E_h^\pi} \\ &\qquad \|q\|_{0,\infty,E_h\triangle E_h^\pi} C_{1,\varrho,l} h_{\hat{E}}^{s+2} \|\mathscr{A}\|_{s+1,\infty,\hat{E}}^3 .\end{aligned} \tag{6.6}$$

Next we consider the L^∞ norm of p and bound this via

$$\begin{aligned}\|p\|_{0,\infty,E_h\triangle E_h^\pi} &\leq \|p\|_{0,\infty,E_h\backslash E_h^\pi} + \|p\|_{0,\infty,E_h^\pi\backslash E_h} \\ &\leq \|p\|_{0,\infty,E_h} + \|p\|_{0,\infty,E_h^\pi} .\end{aligned}$$

From Lemma 6.1.3 it holds that

$$\|p\|_{0,\infty,E_h\triangle E_h^\pi} \leq (1 + C_3) \|p\|_{0,\infty,E_h} ,$$

and similarly we have

$$\|q\|_{0,\infty,E_h\triangle E_h^\pi} \leq (1 + C_3) \|q\|_{0,\infty,E_h} .$$

Substitution of these bounds into Equation (6.6) results in

$$\begin{aligned}\left|\int_{E_h} \tilde{g}pq \; d\mathbf{x} - \int_{E_h^\pi} \tilde{g}pq \; d\mathbf{x}\right| &\leq C_{1,\varrho,l} C_2 (1 + C_3)^2 h_{\hat{E}}^{s+2} \|g\|_{0,\infty,E} \\ &\qquad \|p\|_{0,\infty,E_h} \|q\|_{0,\infty,E_h} \|\mathscr{A}\|_{s+1,\infty,\hat{E}}^3 .\end{aligned} \tag{6.7}$$

We simplify Equation (6.7) by introducing the constant

$$C_{4,\varrho,l,g,\mathscr{A}} := C_{1,\varrho,l} C_2 (1 + C_3)^2 \left\| g \right\|_{0,\infty,E} \left\| \mathscr{A} \right\|^3_{s+1,\infty,\hat{E}},$$

such that

$$\left| \int_{E_h} \tilde{g} p q \; d\mathbf{x} - \int_{E_h^\pi} \tilde{g} p q \; d\mathbf{x} \right| \leq C_{4,\varrho,l,g,\mathscr{A}} h_{\hat{E}}^{s+2} \left\| p \right\|_{0,\infty,E_h} \left\| q \right\|_{0,\infty,E_h}. \tag{6.8}$$

Finally, we apply the inverse estimate found in [49] (Lemma 1.25) of

$$\left\| p \right\|_{0,\infty,E_h} \leq |E_h|^{-1/2} \; \left\| p \right\|_{0,E_h} \qquad \forall p \in \mathbb{P}(E_h),$$

Then combining the inverse estimates with Equation (6.8) obtains

$$\left| \int_{E_h} \tilde{g} p q \; d\mathbf{x} - \int_{E_h^\pi} \tilde{g} p q \; d\mathbf{x} \right| \leq C_{4,\varrho,l,g,\mathscr{A}} \left| E_h \right|^{-1} h_{\hat{E}}^{s+2} \left\| p \right\|_{0,E_h} \left\| q \right\|_{0,E_h}.$$

Application of Lemma 6.1.2 concludes the proof. □

6.2 A Virtual Element Space on Curved Edge Polygons

Here we review the VEM on elements with curved edges proposed in [23]. This space will be used to compute the discrete solution of the PDE. Importantly in our method, any curved edges of ∂E_h can be represented as piecewise polynomial maps from $\partial \hat{E}$ or more simply a collection of maps from the reference interval $I = [0, 1]$.

We denote a curved edge of E_h by e_h with its corresponding reference straight edge $\hat{e} \subset \partial \hat{E}$ and define an arc-length parameterisation by $\gamma_{e_h} : I_{e_h} \to e_h$, where $I_{e_h} = [0, l_{e_h}]$ and l_{e_h} is the length of e_h. From this we introduce the mapped polynomial space of degree k as

$$\mathbb{P}_k^\gamma(e_h) = \{ q \in L^2(e_h) \; : \; q = \hat{q} \circ \gamma_{e_h}^{-1}, \; \hat{q} \in \mathbb{P}_k(I_{e_h}) \}.$$

The local VEM space on E_h is given in [23] as

$$W_k(E_h) = \{ v_h \in H^1(E_h) \cap C^0(E_h) \; : \; \Delta v_h \in \mathbb{P}_{k-2}(E_h), \; v_h|_{e_h} \in \mathbb{P}_k^\gamma(e_h) \; \forall e_h \subset \partial E_h \}.$$

Following the remarks of [23], we enhance this space under the same approach as the polygonal element case [2], taking Π^* to be the least square projection [48], to

$$V_k(E_h) = \{v_h \in H^1(E_h) \cap C^0(E_h) \ : \Delta v_h \in \mathbb{P}_k(E_h),\ v_h|_{e_h} \in \mathbb{P}_k^\gamma(e_h)\ \forall e_h \subset \partial E_h, \int_{E_h} (\Pi_k^* v_h - v_h) q \ d\mathbf{x} = 0\ \forall q \in \mathbb{P}_k(E_h) \backslash \mathbb{P}_{k-2}(E_h)\}. \tag{6.9}$$

The DoFs for this space are similar to the original VEM, we define them for a $v_h \in V_k(E_h)$ as,

- The point values of v_h at the vertices of E_h.
- The point values of v_h for $k-1$ internal points of e_h that are the images through γ_{e_h} of the $k-1$ internal points of the $(k+1)$-point Gauss-Lobatto quadrature on $[0, l_{e_h}]$.
- The internal moments of $\frac{1}{|E_h|}\int_{E_h} v_h q \ dx$ for all $q \in \mathcal{M}_{k-2}(E_h)$.

A proof that these constitute a unisolvent set of DoFs is given in [23]. The global VEM space is then defined as

$$V_h = \{v_h \in H^1(\Omega_h) \ : \ v_h|_{E_h} \in V_k(E_h) \quad \forall E_h \in \mathcal{T}_h\}, \tag{6.10}$$

with the restricted space of VEM functions with zero trace on $\partial\Omega_h$ defined by $V_{h,0} := V_h \cap H_0^1(\Omega_h)$.

Remark 14. *In defining our VEM space we have chosen to use an arc-length parametrisation of each curved edge in the mesh. By doing this we have an interpolation estimate for V_h in which the unknown constant is only dependent on k [23]. In the implementation we use a parametrisation of $\gamma_{e_h} : [0,1] \to e_h$ and still attain optimal orders of accuracy in the numerical experiments of Chapter 7.*

We provide a global interpolation estimate for this virtual element space in the following theorem.

Theorem 6.2.1 (Curved VEM Interpolation Estimate)**.**
Let $V_{h,0}$ be the virtual element space of degree k defined using Equation (6.10) *on a virtual domain Ω_h. For any $v \in H_0^{k+1}(\Omega_h)$ it holds for a fixed integer $s \in \{0, 1, ..., k\}$ that*

$$\|v - v_I\|_{1,\Omega_h} \lesssim h^s \|v\|_{s+1,\Omega_h},$$

where the hidden constant depends on k, the Poincaré-Friedrichs constant of Ω_h, the W_1^∞ norm of $\mathscr{A}$ and the shape regularity parameter $\varrho_{\mathscr{A}}$ of $\mathcal{T}_h$.

Proof. This result is a corollary of Theorem 3.1 from [23]. It holds that

$$\sum_{E_h \in \mathcal{T}_h} |v - v_I|_{1,E_h} \leq C_{1,k,\varrho_{\mathscr{A}}} \max_{E_h \in \mathcal{T}_h} h_{E_h}^s \|v\|_{s+1,\Omega_h}.$$

We bound the error on a local element and scale the mesh size via Lemma 6.1.2

$$\begin{aligned} |v - v_I|_{1,E_h} &\leq C_{1,k,\varrho_{\mathscr{A}}} \max_{\hat{E} \in \hat{\mathcal{T}}_h} C_{2,\mathscr{A}} h_{\hat{E}}^s \|\mathscr{A}\|_{1,\infty,\hat{E}}^s \|v\|_{s+1,\Omega_h} \\ &\leq C_{3,k,\varrho_{\mathscr{A}}} h^s \|\mathscr{A}\|_{1,\infty,\hat{\Omega}}^s \|v\|_{s+1,\Omega_h}. \end{aligned}$$

Summing over elements of the virtual domain and bounding terms gives

$$|v - v_I|_{1,\Omega_h} \leq C_{4,k,\varrho_{\mathscr{A}},\mathscr{A}} h^s \|v\|_{s+1,\Omega_h}.$$

The proof is concluded by applying the Poincaré-Friedrichs inequality. □

6.3 Computable Projection Operators

Unlike Method I proposed in Chapter 5, we do not directly use the H^1 and L^2 projection operators Π^0 and Π^1. In [23] it is argued that a sufficient amount of quadrature points can be used to compute the required integrals over curved edges and curvilinear polygons. Additionally, this works under the assumption that the properties of the mapping $\mathscr{A}$ are known *a priori*. In our work we have made the assumption that only a set of DoFs of $\mathscr{A}_h$ are known and that these may be an approximation themselves. Hence, we introduce two new computable operators that serve to approximate Π^0 and Π^1, namely Π_h^0 and

Π_h^1 respectively. The benefit to this is that we can compute these new operators exactly using only the projected elements and the DoFs with minimal quadrature rules required.

We describe these new operators as projections but note that they do not generally satisfy the definition of a projection as they do not reproduce polynomials exactly on E_h. In general $\Pi_h^0 p \neq p$, $\Pi_h^1 p \neq \nabla p$ for $p \in \mathbb{P}(E_h)$. The error estimates of Theorem 6.3.1 and 6.3.2 suggests that this inconsistency introduces errors terms that are appropriately controlled by the mesh size.

We approximate the Π^0 projection using the projected element and quantify the error of this approximation below. Similarly, we introduce the operator $\Pi_h^1 : V_k(E_h) \to [\mathbb{P}_{k-1}(E_h)]^2$ to approximate Π^1. The accuracy of these approximations are given in Theorems 6.3.1 and 6.3.2. Without loss of generality, we order the DoFs such that $\text{dof}_i(v_h)$ corresponds to the internal moment with respect to the *ith* scaled monomial.

Definition 6.3.1 (The Π_h^0 Projection)**.**
Let $E_h \in \mathcal{T}_h$ and $v_h \in V_k(E_h)$ for some $k \in \mathbb{N}$. The projection $\Pi_{h,k}^0 v_h$ satisfies for all $m_\alpha \in \mathcal{M}_k(E_h)$

$$\int_{E_h^\pi} \Pi_{h,k}^0 v_h \; m_\alpha \; d\mathbf{x} = \begin{cases} |E_h^\pi| \; dof_\alpha(v_h) & m_\alpha \in \mathcal{M}_{k-2}(E_h) \\ \int_{E_h^\pi} \Pi_k^* v_h \; m_\alpha \; d\mathbf{x} & m_\alpha \in \mathcal{M}_k(E_h) \backslash \mathcal{M}_{k-2}(E_h). \end{cases}$$

Definition 6.3.2 (The Π_h^1 Projection)**.**
Let $E_h \in \mathcal{T}_h$ and $v_h \in V_k(E_h)$ for some $k \in \mathbb{N}$. The projection $\Pi_{h,k-1}^1 v_h$ satisfies

$$\int_{E_h^\pi} \Pi_{h,k-1}^1 v_h \cdot \mathbf{m}_\alpha \; d\mathbf{x} = -\frac{|E_h^\pi|}{|E_h|} \int_{E_h} v_h \nabla \cdot \mathbf{m}_\alpha \; d\mathbf{x} + \int_{\partial E_h} v_h \mathbf{m}_\alpha \cdot \mathbf{n} \; dS \qquad \forall \mathbf{m}_\alpha \in [\mathcal{M}_{k-1}(E_h)]^2.$$

Theorem 6.3.1.
Let $\hat{\mathcal{T}}_h$ be the computational reference mesh that satisfies Assumption 1 and $E_h \in \mathcal{T}_h$ be a shape regular element given by $\mathscr{A}_h(\hat{E})$ for some $\hat{E} \in \hat{\mathcal{T}}_h$ with $h_{\hat{E}}$ sufficiently small. For the projection operator $\Pi_{h,k}^0 : V_k(E_h) \to \mathbb{P}_k(E_h)$ the

following bound holds for all $k \in \mathbb{N}$, $v_h \in V_k(E_h)$ *and* $s \in \{0, 1, ..., l\}$

$$\left\|\Pi_k^0 v_h - \Pi_{h,k}^0 v_h\right\|_{0,E_h} \lesssim h_{\hat{E}}^s \left\|v_h\right\|_{0,E_h}, \tag{6.11}$$

where the hidden constant depends on N^{dofs}, ϱ, l, j_0 *and the* W_{s+1}^∞ *norm of* $\mathscr{A}$. *Furthermore, the following stability estimate holds*

$$\left\|\Pi_{h,k}^0 v_h\right\|_{0,E_h} \lesssim \left\|v_h\right\|_{0,E_h},$$

with the same hidden constant dependency as Equation (6.11).

Proof. In this proof we adopt the notation used in [44] in defining the matrix equations to compute the L^2 projection. We consider the error for some $v_h \in V_k(E_h)$ as a vector-matrix product

$$\left\|\Pi_k^0 v_h - \Pi_{h,k}^0 v_h\right\|_{0,E_h}^2 = \left(\mathbf{t} - \mathbf{t}_h\right)^T \mathbf{M} \left(\mathbf{t} - \mathbf{t}_h\right),$$

where we define the mass matrix on E_h by

$$M_{\alpha,\beta} = \int_{E_h} m_\alpha m_\beta \, d\mathbf{x},$$

for $m_\alpha, m_\beta \in \mathcal{M}_k(E_h)$ and the vectors of coefficients of Π^0 and Π_h^0 are given by $\mathbf{t}$ and $\mathbf{t}_h$ respectively. By using the l_2 norm $\|\cdot\|_2$ and the matrix norm induced by $\|\cdot\|_2$ and analysing individual matrix terms we can bound the error noting that the mass matrix terms scale like $|E_h|$

$$\begin{aligned}\left\|\Pi_k^0 v_h - \Pi_{h,k}^0 v_h\right\|_{0,E_h}^2 &\leq \left\|\mathbf{M}\right\|_2 \left\|\mathbf{t} - \mathbf{t}_h\right\|_2^2 \\ &\leq C_{1,N^{\text{dofs}}} h_{E_h}^2 \left\|\mathbf{t} - \mathbf{t}_h\right\|_2^2, \end{aligned} \tag{6.12}$$

and it remains to bound the error in the projection coefficients. The Π_k^0 projection is determined by

$$\mathbf{t} = \mathbf{M}^{-1}\mathbf{c} \qquad \text{where } c_\alpha = \begin{cases} |E_h| \operatorname{dof}_\alpha(v_h) & m_\alpha \in \mathcal{M}_{k-2}(E_h) \\ \int_{E_h} \Pi_k^* \, v_h \, m_\alpha \, d\mathbf{x} & m_\alpha \in \mathcal{M}_k(E_h) \backslash \mathcal{M}_{k-2}(E_h) \end{cases},$$

and, similarly, the Π_h^0 projection is determined by

$$\mathbf{t}_h = \mathbf{M}_h^{-1}\mathbf{c}_h \qquad \text{where } c_{\alpha,h} = \begin{cases} |E_h^\pi| \operatorname{dof}_\alpha(v_h) & m_\alpha \in \mathcal{M}_{k-2}(E_h) \\ \int_{E_h^\pi} \Pi_k^* v_h \, m_\alpha \, d\mathbf{x} & m_\alpha \in \mathcal{M}_k(E_h) \backslash \mathcal{M}_{k-2}(E_h). \end{cases}$$

We then consider the difference of these coefficients, noting that

$$\mathbf{t} - \mathbf{t}_h = \left(\mathbf{M}^{-1} - \mathbf{M}_h^{-1}\right)\mathbf{c} \; + \; \mathbf{M}_h^{-1}\left(\mathbf{c} - \mathbf{c}_h\right).$$

Taking the norm and applying the triangle and matrix norm inequalities results in

$$\left\|\mathbf{t} - \mathbf{t}_h\right\|_2 \leq \left\|\mathbf{M}^{-1} - \mathbf{M}_h^{-1}\right\|_2 \left\|\mathbf{c}\right\|_2 + \left\|\mathbf{M}_h^{-1}\right\|_2 \left\|\mathbf{c} - \mathbf{c}_h\right\|_2.$$

The first term can be expanded by

$$\mathbf{M}^{-1} - \mathbf{M}_h^{-1} = \mathbf{M}_h^{-1}\left(\mathbf{M}_h - \mathbf{M}\right)\mathbf{M}^{-1}. \tag{6.13}$$

We note that matrix inverses scale like

$$\mathbf{M}^{-1}, \mathbf{M}_h^{-1} \sim h_{E_h}^{-2}.$$

We bound Equation (6.13) using matrix norms, the scaling of the mass matrix and applying the result of Lemma 6.1.4

$$\begin{aligned}
\left\|\mathbf{M}^{-1} - \mathbf{M}_h^{-1}\right\|_2 &\leq \left\|\mathbf{M}_h^{-1}\right\|_2 \left\|\mathbf{M}_h - \mathbf{M}\right\|_2 \left\|\mathbf{M}^{-1}\right\|_2 \\
&\leq C_{2,N^{\text{dofs}}} h_{E_h}^{-4} \left\|\mathbf{M}_h - \mathbf{M}\right\|_2 \\
&\leq C_{3,N^{\text{dofs}}} h_{E_h}^{-4} \max_{\alpha,\beta} \left| \int_{E_h} m_\alpha m_\beta \; d\mathbf{x} - \int_{E_h^\pi} m_\alpha m_\beta \; d\mathbf{x} \right| \\
&\leq C_{3,N^{\text{dofs}}} h_{E_h}^{-4} C_{4,\varrho,l,j_0,\mathscr{A}} h_{\hat{E}}^s \left\|m_\alpha\right\|_{0,E_h} \left\|m_\beta\right\|_{0,E_h} \\
&\leq C_{5,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A}} h_{E_h}^{-2} h_{\hat{E}}^s,
\end{aligned}$$

where in the last bound we applied the property that $\left\|m_\alpha\right\|_{0,E_h} \sim h_{E_h}$ for all m_α. Using this bound gives us

$$\left\|\mathbf{t} - \mathbf{t}_h\right\|_2 \leq C_{6,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A}} h_{E_h}^{-2} h_{\hat{E}}^s \left\|\mathbf{c}\right\|_2 + C_{7,N^{\text{dofs}}} h_{E_h}^{-2} \left\|\mathbf{c} - \mathbf{c}_h\right\|_2. \tag{6.14}$$

The difference of $\mathbf{c} - \mathbf{c}_h$ requires a case-by-case analysis.

Case I: $m_\alpha \in \mathcal{M}_{k-2}(E_h)$

In this case the integral is provided by the DoFs of v_h. Without loss of generality

we refer to these as the α-th degree of freedom

$$c_\alpha = \int_{E_h} v_h m_\alpha \, d\mathbf{x} = |E_h| \operatorname{dof}_\alpha(v_h).$$

In this case only the measure of E_h is unknown and approximated using the projected element. Taking the difference and applying Lemma 6.1.4 (for $p = q = 1$) gives

$$\begin{aligned} |c_\alpha - c_{\alpha,h}| &\leq |\operatorname{dof}_\alpha(v_h)| \, ||E_h| - |E_h^\pi|| \\ &\leq |\operatorname{dof}_\alpha(v_h)| \, C_{8,\varrho,l,\mathscr{A},j_0} h_{\hat{E}}^s \, |E_h| \, . \end{aligned}$$

Here we recall the "scaling as one" property of the scaled monomials [44]

$$\operatorname{dof}_\alpha(v_h) \sim h_{E_h}^{-1} \, \|v_h\|_{0,E_h} \, .$$

Using this property we get

$$|c_\alpha - c_{\alpha,h}| \leq C_{9,\varrho,l,\mathscr{A},j_0} h_{E_h} h_{\hat{E}}^s \, \|v_h\|_{0,E_h} \, .$$

Case II: $m_\alpha \in \mathcal{M}_k(E_h) \backslash \mathcal{M}_{k-2}(E_h)$

In this case we use the orthogonality condition imposed in the definition of the VEM space in Equation (6.9). Here we have

$$c_\alpha = \int_{E_h} \Pi_k^* v_h \; m_\alpha \; d\mathbf{x}, \qquad c_{\alpha,h} = \int_{E_h^\pi} \Pi_k^* v_h \; m_\alpha \; d\mathbf{x}.$$

Then applying Lemma 6.1.4 and the Cauchy-Schwarz inequality we get

$$|c_\alpha - c_{\alpha,h}| \leq C_{10,\varrho,l,\mathscr{A},j_0} h_{\hat{E}}^s \, \|\Pi_k^* v_h\|_{0,E_h} \, \|m_\alpha\|_{0,E_h} \, .$$

Next we apply the stability of the Π_k^* projection and bound the norms of the scaled monomials to get

$$|c_\alpha - c_{\alpha,h}| \leq C_{11,\varrho,l,\mathscr{A},j_0} h_{E_h} h_{\hat{E}}^s \, \|v_h\|_{0,E_h} \, .$$

Having bounded both cases we can now bound $\mathbf{c} - \mathbf{c}_h$ via

$$\|\mathbf{c} - \mathbf{c}_h\|_2 \leq C_{12,\varrho,l,\mathscr{A},j_0} h_{E_h} h_{\hat{E}}^s \, \|v_h\|_{0,E_h} \, . \tag{6.15}$$

The norm of $\mathbf{c}$ can also be bounded by a similar argument

$$\|\mathbf{c}\| \leq C_{13,N^{\text{dofs}}} h_{E_h} \|v_h\|_{0,E_h} . \tag{6.16}$$

Now we bound the term $\mathbf{t} - \mathbf{t}_h$ using Equations (6.14), (6.15) and (6.16) and the scaling of $\mathbf{M}_h^{-1}$ to get

$$\|\mathbf{t} - \mathbf{t}_h\|_2 \leq C_{14,N^{\text{dofs}},\varrho,l,\mathscr{A},j_0} h_{\hat{E}}^s h_{E_h}^{-1} \|v_h\|_{0,E_h} .$$

To conclude the proof we substitute this bound into the original error Equation (6.12)

$$\begin{aligned}
\left\|\Pi_k^0 v_h - \Pi_{h,k}^0 v_h\right\|_{0,E_h}^2 &\leq C_{1,N^{\text{dofs}}} h_{E_h}^2 \|\mathbf{t} - \mathbf{t}_h\|_2^2 \\
&\leq C_{1,N^{\text{dofs}}} h_{E_h}^2 \left(C_{14,N^{\text{dofs}},\varrho,l,\mathscr{A},j_0} h_{\hat{E}}^s h_{E_h}^{-1} \|v_h\|_{0,E_h}\right)^2 \\
&\leq C_{15,N^{\text{dofs}},\varrho,l,\mathscr{A},j_0} h_{\hat{E}}^{2s} \|v_h\|_{0,E_h}^2 .
\end{aligned}$$

Taking the square root completes the proof. The stability bound is a consequence of setting $s = 0$ and applying the triangle inequality. □

Theorem 6.3.2.

Let $\hat{\mathcal{T}}_h$ be the computational reference mesh that satisfies Assumption 1 and $E_h \in \mathcal{T}_h$ be a shape regular element by $\mathscr{A}_h(\hat{E})$ for some $\hat{E} \in \hat{\mathcal{T}}_h$ with $h_{\hat{E}}$ sufficiently small. For the projection operator $\Pi_{h,k-1}^1 : V_k(E_h) \to \mathbb{P}_k(E_h)$ the following bound holds for all $k \in \mathbb{N}$, $v_h \in V_k(E_h)$ and $s \in \{0, 1, ..., l\}$

$$\left\|\Pi_{k-1}^1 v_h - \Pi_{h,k-1}^1 v_h\right\|_{0,E_h} \lesssim h_{\hat{E}}^s \|v_h\|_{1,E_h} ,$$

where the hidden constant depends on N^{dofs}, ϱ, l, j_0 and the W_{s+1}^∞ norm of $\mathscr{A}$. Furthermore, the following stability estimate holds

$$\left\|\Pi_{h,k-1}^1 v_h\right\|_{0,E_h} \lesssim \|v_h\|_{1,E_h} ,$$

Proof. The proof of the estimate follows the same arguments as Theorem 6.3.1. We have

$$\begin{aligned}
\left\|\Pi_{k-1}^1 v_h - \Pi_{h,k-1}^1 v_h\right\|_{0,E_h}^2 &\leq \|\mathbf{M}\|_2 \|\mathbf{t} - \mathbf{t}_h\|_2^2 \\
&\leq C_{1,N^{\text{dofs}}} h_{E_h}^2 \|\mathbf{t} - \mathbf{t}_h\|_2^2 ,
\end{aligned} \tag{6.17}$$

where $\mathbf{M}$ and the corresponding approximation $\mathbf{M}_h$ are the vector equivalent of $\mathbf{M}$ and $\mathbf{M}_h$ defined in the proof of Theorem 6.3.1. We recall the Π^1 projection is determined by

$$\mathbf{t} = \mathbf{M}^{-1}\mathbf{c},$$

where

$$c_\alpha = -\int_{E_h} v_h \nabla \cdot \mathbf{m}_\alpha \, d\mathbf{x} + \int_{\partial E_h} v_h \mathbf{m}_\alpha \cdot \mathbf{n} \, dS \qquad \mathbf{m}_\alpha \in \left[\mathcal{M}_{k-1}(E_h)\right]^2. \quad (6.18)$$

Similarly, the Π^1_h projection is determined by

$$\mathbf{t}_h = \mathbf{M}_h^{-1}\mathbf{c}_h,$$

where

$$c_{\alpha,h} = -\frac{|E_h^\pi|}{|E_h|}\int_{E_h} v_h \nabla \cdot \mathbf{m}_\alpha \, d\mathbf{x} + \int_{\partial E_h} v_h \mathbf{m}_\alpha \cdot \mathbf{n} \, dS \qquad \mathbf{m}_\alpha \in \left[\mathcal{M}_{k-1}(E_h)\right]^2. \quad (6.19)$$

Following the proof of Theorem 6.3.1 up to Equation (6.14) gives

$$\|\mathbf{t} - \mathbf{t}_h\|_2 \leq C_{2,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A}} h_{E_h}^{-2} h_{\hat{E}}^s \|\mathbf{c}\|_2 + C_{3,N^{\text{dofs}}} h_{E_h}^{-2} \|\mathbf{c} - \mathbf{c}_h\|_2. \quad (6.20)$$

Applying integration by parts to Equation (6.18) gives

$$c_\alpha = \int_{E_h} \mathbf{m}_\alpha \cdot \nabla v_h \, d\mathbf{x},$$

from which we bound $\mathbf{c}$ using the Cauchy-Schwarz inequality and the scaling of $\mathbf{m}_\alpha$

$$\|\mathbf{c}\|_2 \leq C_{3,N^{\text{dofs}}} h_{E_h} \|\nabla v_h\|_{0,E_h}. \quad (6.21)$$

It remains to estimate the error term of $\|\mathbf{c} - \mathbf{c}_h\|_2$. Since $\nabla \cdot \mathbf{m}_\alpha \in [\mathbb{P}_{k-2}(E_h)]^2$ we can evaluate

$$\frac{1}{|E_h|}\int_{E_h} v_h \nabla \cdot \mathbf{m}_\alpha \, d\mathbf{x},$$

exactly using only the DoFs. As a result the difference term is given, using Equations (6.18) and (6.19), by

$$c_\alpha - c_{\alpha,h} = (|E_h| - |E_h^\pi|)\frac{1}{|E_h|}\int_{E_h} v_h \nabla \cdot \mathbf{m}_\alpha \, d\mathbf{x}, \quad (6.22)$$

noting that the integration over the element boundary is computed exactly on the reference element. The integral in the difference term is computable using the internal DoFs. The difference in volume is bounded as done in Theorem 6.3.1 by applying Lemma 6.1.4

$$||E_h^\pi| - |E_h|| \leq C_{1,\varrho,l,\mathscr{A},j_0} h_{\hat{E}}^s \, |E_h| \, .$$

Noting that the weighted integral is a linear combination of DoFs of v_h, that scale as one by definition [44], we have

$$\frac{1}{|E_h|} \int_{E_h} v_h \nabla \cdot \mathbf{m}_\alpha \, d\mathbf{x} \sim h_{E_h}^{-1} \left\| v_h \right\|_{0,E_h} .$$

Inserting these bounds into Equation (6.22) gives

$$\left\| \mathbf{c} - \mathbf{c}_h \right\|_2 \leq C_{4,N^{\text{dofs}},\varrho,l,\mathscr{A},j_0} h_{\hat{E}}^s h_{E_h} \left\| v_h \right\|_{0,E_h} .$$

Next we insert this bound into Equation (6.20) along with Equation (6.21) gives

$$\begin{aligned} \left\| \mathbf{t} - \mathbf{t}_h \right\|_2 &\leq C_{5,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A}} h_{\hat{E}}^s h_{E_h}^{-1} \left(\left\| \nabla v_h \right\|_{0,E_h} + \left\| v_h \right\|_{0,E_h} \right) \\ &\leq 2C_{5,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A}} h_{\hat{E}}^s h_{E_h}^{-1} \left\| v_h \right\|_{1,E_h} . \end{aligned}$$

Finally, we have from Equation (6.17) that

$$\begin{aligned} \left\| \Pi_{h,k-1}^1 v_h - \Pi_{k-1}^1 v_h \right\|_{0,E_h}^2 &\leq C_{1,N^{\text{dofs}}} h_{E_h}^2 \left(2C_{5,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A}} h_{\hat{E}}^s h_{E_h}^{-1} \left\| v_h \right\|_{1,E_h} \right)^2 \\ &\leq C_{6,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A}} h_{\hat{E}}^{2s} \left\| v_h \right\|_{1,E_h}^2 . \end{aligned}$$

Taking the square root completes the proof. The stability bound is a consequence of setting $s = 0$ and applying the triangle inequality. □

6.4 Method II Formulation

This method is formulated in the standard VEM fashion locally on each projected element [20, 39]. In this method we divide the bilinear/linear forms into

contributions over the virtual mesh $\mathcal{T}_h$

$$\begin{aligned}\mathcal{A}_h(\rho_h, v_h) &= \sum_{E_h \in \mathcal{T}_h} A_h^{E_h}(\rho_h, v_h) + B_h^{E_h}(\rho_h, v_h) + C_h^{E_h}(\rho_h, v_h) \\ &\quad + \sum_{E_h \in \mathcal{T}_h} \mathcal{S}^{E_h}(\rho_h - \Pi_k^* \rho_h, v_h - \Pi_k^* v_h), \\ l_h(v_h) &= \sum_{E_h \in \mathcal{T}_h} l_h^{E_h}(v_h).\end{aligned}$$

The local integral equations are defined by

$$\begin{aligned}A_h^{E_h}(\rho_h, v_h) &= \int_{E_h^\pi} \tilde{a}\ \Pi^1_{h,k-1}\rho_h \cdot \Pi^1_{h,k-1} v_h\ d\mathbf{x}, \\ B_h^{E_h}(\rho_h, v_h) &= \frac{1}{2}\int_{E_h^\pi} \tilde{\mathbf{b}} \cdot (\Pi^0_{h,k} v_h \Pi^1_{h,k-1}\rho_h - \Pi^0_{h,k}\rho_h \Pi^1_{h,k-1} v_h)\ d\mathbf{x}, \\ C_h^{E_h}(\rho_h, v_h) &= \int_{E_h^\pi} \tilde{\mu}\ \Pi^0_{h,k}\rho_h\ \Pi^0_{h,k} v_h\ d\mathbf{x}, \\ l_h^{E_h}(v_h) &= \int_{E_h^\pi} \tilde{f}_h\ \Pi^0_{h,k} v_h\ d\mathbf{x},\end{aligned}$$

and the stabilisation term is given by the dofi-dofi formula (see Equation (2.8))

$$\mathcal{S}^{E_h}(\rho_h, v_h) = (\bar{a} + h_{E_h}^2 \bar{\mu}) \sum_{i=1}^{N^{\text{dofs}}} \text{dof}_i(\rho_h) \cdot \text{dof}_i(v_h), \tag{6.23}$$

with $\bar{a}$ and $\bar{\mu}$ being nodal averages on E_h of the extensions of a and μ respectively. In this method $\tilde{f}_h := \Pi^0_{h,k-2}\tilde{f}$ for $k > 2$ and $\tilde{f}_h := \Pi^0_0 \tilde{f}$ for $k = 1, 2$. To simplify the analysis we assume the scaling constants in the stabilisation are equivalent to those used in the definition of the perturbed bilinear form.

The isoparametric VEM is then defined, recalling the definition of V_h and $V_{h,0}$ in Equation (6.10), as follows: find $\rho_h \in V_{h,0}$ such that

$$\mathcal{A}_h(\rho_h, v_h) = l_h(v_h) \qquad \forall v_h \in V_{h,0}.$$

6.5 Consistency Error

Here we introduce a bound on the inconsistency by the use of the projected elements that will be used in the H^1 estimate as well as the coercivity proof

for this method. For the analysis we recall the non-computable VEM bilinear form $\mathcal{B}_h(\cdot,\cdot)$ defined in Equation (5.4)

$$\mathcal{B}_h(\rho,v) = \sum_{E_h\in\mathcal{T}_h}\int_{E_h} \tilde{a}\Pi^1_{k-1}\rho\cdot\Pi^1_{k-1}v + \frac{1}{2}\tilde{\mathbf{b}}\cdot(\Pi^0_k v\Pi^1_{k-1}\rho - \Pi^0_k\rho\Pi^1_{k-1}v) + \tilde{\mu}\Pi^0_k\rho\ \Pi^0_k v\ d\mathbf{x} + \mathcal{S}^{E_h}(\rho-\Pi^*_k\rho, v-\Pi^*_k v),$$

with the stabilisation term taken exactly as in Equation (6.23). The isoparametric VEM is directly compared against $\mathcal{B}_h^{E_h}(\cdot,\cdot)$ in the following Lemma.

Lemma 6.5.1.

For $k,l\in\mathbb{N}$, let $E_h\in\mathcal{T}_h$ be a curved element of degree l given by a shape regular reference element $\hat{E}\in\hat{\mathcal{T}}_h$, with $h_{\hat{E}}$ sufficiently small, and $E_h=\mathscr{A}(\hat{E})$. For $\rho_h, v_h\in V_k(E_h)$ and a fixed integer $s\in\{0,1,...,\min\{k,l\}\}$, the following bound holds

$$\left|\mathcal{B}_h^{E_h}(\rho_h,v_h) - \mathcal{A}_h^{E_h}(\rho_h,v_h)\right| \lesssim h_{\hat{E}}^s \left\|\rho_h\right\|_{1,E_h}\left\|v_h\right\|_{1,E_h},$$

where the hidden constant depends on N^{dofs}, ϱ, l, j_0, the W^∞_{s+1} norm of $\mathscr{A}$ and the L^∞ norms of a, $\mathbf{b}$ and μ on the element $E=\mathscr{A}(\hat{E})$.

Proof. Firstly we split the difference term via

$$\mathcal{B}_h^{E_h}(\rho_h,v_h) - \mathcal{A}_h^{E_h}(\rho_h,v_h) = EQ + E1 + E2 + E3 + ES,$$

where through adding and subtracting terms and noting that $ES = 0$ we have,

$$
\begin{aligned}
EQ = &\int_{E_h} \tilde{a}\Pi^1_{h,k-1}\rho_h \cdot \Pi^1_{h,k-1} v_h \; d\mathbf{x} - \int_{E_h^\pi} \tilde{a}\Pi^1_{h,k-1}\rho_h \cdot \Pi^1_{h,k-1} v_h \; d\mathbf{x} \\
&+ \frac{1}{2}\int_{E_h} \tilde{\mathbf{b}} \cdot (\Pi^0_{h,k} v_h \Pi^1_{h,k-1}\rho_h - \Pi^0_{h,k}\rho_h \Pi^1_{h,k-1} v_h) \; d\mathbf{x} \\
&- \frac{1}{2}\int_{E_h^\pi} \tilde{\mathbf{b}} \cdot (\Pi^0_{h,k} v_h \Pi^1_{h,k-1}\rho_h - \Pi^0_{h,k}\rho_h \Pi^1_{h,k-1} v_h) \; d\mathbf{x} \\
&+ \int_{E_h} \tilde{\mu}\Pi^0_{h,k} p \Pi^0_{h,k} v_h \; d\mathbf{x} - \int_{E_h^\pi} \tilde{\mu}\Pi^0_{h,k} p \Pi^0_{h,k} v_h \; d\mathbf{x}, \\
E1 = &\int_{E_h} \big(\tilde{a}(\Pi^1_{k-1} - \Pi^1_{h,k-1})\rho_h \cdot \Pi^1_{k-1} v_h + \tilde{a}\Pi^1_{h,k-1}\rho_h \cdot (\Pi^1_{k-1} - \Pi^1_{h,k-1}) v_h \; \big) \, d\mathbf{x}, \\
E2 = &\frac{1}{2}\int_{E_h} \tilde{\mathbf{b}} \cdot \big(\Pi^0_k v_h \Pi^1_{k-1}\rho_h - \Pi^0_k \rho_h \Pi^1_{k-1} v_h + \Pi^0_{h,k} v_h \Pi^1_{h,k-1}\rho_h + \Pi^0_{h,k}\rho_h \Pi^1_{h,k-1} v_h\big) \, d\mathbf{x}, \\
E3 = &\int_{E_h} \big(\tilde{\mu}(\Pi^0_k - \Pi^0_{h,k})\rho_h \Pi^0_k v_h + \tilde{\mu}\Pi^0_{h,k}\rho_h(\Pi^0_k - \Pi^0_{h,k}) v_h \; \big) \, d\mathbf{x}.
\end{aligned}
$$

The quadrature error EQ is bounded by directly applying Lemma 6.1.4 to give

$$
\begin{aligned}
|EQ| \leq C_{1,\varrho,l,\mathscr{A},j_0} h_{\hat{E}}^s \Big(&\left\|\tilde{a}\right\|_{0,\infty,E_h} \left\|\Pi^1_{h,k-1}\rho_h\right\|_{0,E_h} \left\|\Pi^1_{h,k-1} v_h\right\|_{0,E_h} \\
&+ \frac{1}{2}\left\|\tilde{\mathbf{b}}\right\|_{0,\infty,E_h} \left\|\Pi^1_{h,k-1}\rho_h\right\|_{0,E_h} \left\|\Pi^0_{h,k} v_h\right\|_{0,E_h} \\
&+ \frac{1}{2}\left\|\tilde{\mathbf{b}}\right\|_{0,\infty,E_h} \left\|\Pi^1_{h,k-1} v_h\right\|_{0,E_h} \left\|\Pi^0_{h,k}\rho_h\right\|_{0,E_h} \\
&+ \left\|\tilde{\mu}\right\|_{0,\infty,E_h} \left\|\Pi^0_{h,k}\rho_h\right\|_{0,E_h} \left\|\Pi^0_{h,k} v_h\right\|_{0,E_h} \Big).
\end{aligned}
$$

Then using the stability bounds in Theorems 6.3.1 and 6.3.2 provides

$$
\begin{aligned}
|EQ| \leq C_{2,N^{\text{dofs}},\varrho,l,\mathscr{A},j_0} h_{\hat{E}}^s \Big(&\left\|\tilde{a}\right\|_{0,\infty,E_h} \left\|\rho_h\right\|_{1,E_h} \left\|v_h\right\|_{1,E_h} \\
&+ \frac{1}{2}\left\|\tilde{\mathbf{b}}\right\|_{0,\infty,E_h} \left\|\rho_h\right\|_{1,E_h} \left\|v_h\right\|_{0,E_h} \\
&+ \frac{1}{2}\left\|\tilde{\mathbf{b}}\right\|_{0,\infty,E_h} \left\|v_h\right\|_{1,E_h} \left\|\rho_h\right\|_{0,E_h} \\
&+ \left\|\tilde{\mu}\right\|_{0,\infty,E_h} \left\|\rho_h\right\|_{0,E_h} \left\|v_h\right\|_{0,E_h} \Big).
\end{aligned}
$$

Bounding the L^2 norm by the H^1 norm and Applying Stein's Extension Theorem 5.3.1 gives

$$
\begin{aligned}
|EQ| &\leq C_{3,N^{\text{dofs}},\varrho,l,\mathscr{A},j_0} h_{\hat{E}}^s \max\left(\left\|a\right\|_{0,\infty,E}, \left\|\mathbf{b}\right\|_{0,\infty,E}, \left\|\mu\right\|_{0,\infty,E}\right) \left\|\rho_h\right\|_{1,E_h} \left\|v_h\right\|_{1,E_h} \\
&\leq C_{4,N^{\text{dofs}},\varrho,l,\mathscr{A},j_0,a,\mathbf{b},\mu} h_{\hat{E}}^s \left\|\rho_h\right\|_{1,E_h} \left\|v_h\right\|_{1,E_h}. \qquad (6.24)
\end{aligned}
$$

We bound $E1$ by using Hölder's inequality and the Cauchy-Schwarz inequality along with Theorem 6.3.2 to get

$$\begin{aligned}|E1| &\leq \|\tilde{a}\|_{0,\infty,E_h} \left\|\Pi^1_{k-1}\rho_h - \Pi^1_{h,k-1}\rho_h\right\|_{0,E_h} \left\|\Pi^1_{k-1}v_h\right\|_{0,E_h} \\ &\quad + \|\tilde{a}\|_{0,\infty,E_h} \left\|\Pi^1_{h,k-1}\rho_h\right\|_{0,E_h} \left\|\Pi^1_{k-1}v_h - \Pi^1_{h,k-1}v_h\right\|_{0,E_h} \\ &\leq C_{5,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A},a} h^s_{\hat{E}} \|\rho_h\|_{1,E_h} \|v_h\|_{1,E_h},\end{aligned}$$

where in the final line we applied Stein's Extension Theorem 5.3.1 to bound the norm of $\tilde{a}$ by a constant. Following similar steps for $E2$ and $E3$ and utilising Theorems 6.3.1 and 6.3.2 gives

$$|E1| + |E2| + |E3| \leq C_{6,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A},a,\mathbf{b},\mu} h^s_{\hat{E}} \|\rho_h\|_{1,E_h} \|v_h\|_{1,E_h}.$$

We conclude the proof by combining the above bound with Equation (6.24). □

6.6 Well-posedness

Theorem 6.6.1.

For sufficiently small h, $\mathcal{A}_h(\cdot,\cdot)$ is coercive and continuous and $l_h(\cdot)$ is bounded such that the VEM: Find $\rho_h \in V_{h,0}$ such that

$$\mathcal{A}_h(\rho_h, v_h) = l_h(v_h) \qquad \forall v_h \in V_{h,0},$$

has a unique solution via the Lax-Milgram Theorem.

Proof. Here we proof local continuity and infer the global property using standard arguments [20]. Through the triangle inequality we have that

$$\begin{aligned}|\mathcal{A}_h(\rho_h, v_h)| &\leq |\mathcal{A}_h(\rho_h, v_h) - \mathcal{B}_h(\rho_h, v_h)| + |\mathcal{B}_h(\rho_h, v_h)| \\ &\leq \sum_{E_h \in \mathcal{T}_h} \left|\mathcal{A}^{E_h}_h(\rho_h, v_h) - \mathcal{B}^{E_h}_h(\rho_h, v_h)\right| + |\mathcal{B}_h(\rho_h, v_h)|.\end{aligned}$$

We apply the local estimate of Lemma 6.5.1 and bound terms to get

$$\begin{aligned}|\mathcal{A}_h(\rho_h, v_h)| &\leq \sum_{E_h \in \mathcal{T}_h} C_{1,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A},a,\mathbf{b},\mu} h^s_{\hat{E}} \|\rho_h\|_{1,E_h} \|v_h\|_{1,E_h} + |\mathcal{B}_h(\rho_h, v_h)| \\ &\leq C_{2,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A},a,\mathbf{b},\mu} h^s \|\rho_h\|_{1,\Omega_h} \|v_h\|_{1,\Omega_h} + |\mathcal{B}_h(\rho_h, v_h)|.\end{aligned}$$

To complete the continuity proof, we introduce the continuity constant of $\mathcal{B}_h(\cdot,\cdot)$ given by [39] to give

$$|\mathcal{A}_h(\rho_h, v_h)| \leq C_{3,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A},a,\mathbf{b},\mu} \|\rho_h\|_{1,\Omega_h} \|v_h\|_{1,\Omega_h} . \tag{6.25}$$

Continuity of $l_h(\cdot)$ follows from a similar argument

$$|l_h(v_h)| \leq C_{4,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A},f} \|v_h\|_{1,\Omega_h} .$$

To prove the coercivity of $\mathcal{A}_h(\cdot,\cdot)$, we consider the bilinear form $\mathcal{B}_h^{E_h}(\cdot,\cdot)$ defined in Equation (5.4) and consider the coercivity constant $C_{5,a_0,\mu_0} > 0$ given in [39] and apply Lemma 6.5.1 to get

$$\begin{aligned} C_{5,a_0,\mu_0} \|v_h\|_{1,\Omega_h}^2 &\leq \mathcal{B}_h(v_h, v_h) \\ &\leq \mathcal{A}_h(v_h, v_h) + \sum_{E_h \in \mathcal{T}_h} \left| \mathcal{A}_h^{E_h}(v_h, v_h) - \mathcal{B}_h^{E_h}(v_h, v_h) \right| \\ &\leq \mathcal{A}_h(v_h, v_h) + C_{2,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A},a,\mathbf{b},\mu} h^s \|v_h\|_{1,\Omega_h}^2 , \end{aligned}$$

hence

$$\left(C_{5,a_0,\mu_0} - C_{2,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A},a,\mathbf{b},\mu} h^s\right) \|v_h\|_{1,\Omega_h}^2 \leq \mathcal{A}_h(v_h, v_h).$$

Local coercivity is then obtained by, for example, imposing that h is sufficiently small such that

$$C_{2,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A},a,\mathbf{b},\mu} h^s < \frac{C_{5,a_0,\mu_0}}{2}.$$

This provides the coercivity condition with a constant independent of the mesh size and the mapping $\mathscr{A}$

$$C_{6,a_0,\mu_0} \|v_h\|_{1,\Omega_h}^2 \leq \mathcal{A}_h(v_h, v_h). \tag{6.26}$$

□

6.7 A H^1 Estimate

We derive a H^1 estimate through a Strang-type error bound quantification of the quadrature error of using projected elements and bounds for local bilinear form differences.

Theorem 6.7.1 (Strang-type Bound)**.**

Let $\rho_h, v_h \in V_{h,0}$ be the VEM solution to the PDE 5.1 using Method II outlined in section 6.4. Furthermore, let $\tilde{\rho}$ be the extension of the solution to the PDE (5.1) *from Ω to Ω_h and let f^* be the extension of the forcing term in Equation* (5.1) *defined as in [41] by*

$$-\nabla \cdot (\tilde{a}\nabla\tilde{\rho}) + \tilde{\mathbf{b}} \cdot \nabla\tilde{\rho} + \tilde{c}\tilde{\rho} = f^* \qquad \mathbf{x} \in \Omega_h, \tag{6.27}$$

along with the following linear operator

$$l_*(v_h) = \int_{\Omega_h} f^* v_h \, d\mathbf{x}.$$

It holds on Ω_h, with a corresponding reference mesh $\hat{\mathcal{T}}_h$ satisfying Assumption 1 and of sufficently small mesh size h, that for all $v_h \in V_{h,0}$

$$\begin{aligned}
\|\rho_h - v_h\|_{1,\Omega_h} \lesssim & \left(\inf_{v_h \in V_{h,0}} \|\tilde{\rho} - v_h\|_{1,\Omega_h} + \inf_{p \in \mathbb{P}_k(\mathcal{T}_h)} \|\tilde{\rho} - p\|_{1,\Omega_h} \right) \\
& + \inf_{p \in \mathbb{P}_k(\mathcal{T}_h)} \left[\sum_{E_h \in \mathcal{T}_h} \sup_{w_h \in V_k(E_h)\setminus\{0\}} \frac{\left| \mathcal{B}^{E_h}(p, w_h) - \mathcal{B}_h^{E_h}(p, w_h) \right|}{\|w_h\|_{1,E_h}} \right] \\
& + \inf_{p \in \mathbb{P}_k(\mathcal{T}_h)} \left[\sum_{E_h \in \mathcal{T}_h} \sup_{w_h \in V_k(E_h)\setminus\{0\}} \frac{\left| \mathcal{B}_h^{E_h}(p, w_h) - \mathcal{A}_h^{E_h}(p, w_h) \right|}{\|w_h\|_{1,E_h}} \right] \\
& + \sup_{w_h \in V_{h,0}\setminus\{0\}} \frac{|l_h(w_h) - l_*(w_h)|}{\|w_h\|_{1,\Omega_h}},
\end{aligned}$$

where the hidden constant depends on N^{dofs}, ϱ, l, the lower bounds a_0, μ_0, j_0, the L^∞ norms of a, $\mathbf{b}$ and μ and the W^∞_{s+1} norm of $\mathscr{A}$.

Proof. From Equation (6.26) of Theorem 6.6.1 there exists a coercivity of $\mathcal{A}_h(\cdot,\cdot)$ such that

$$C_{1,a_0,\mu_0} \|\rho_h - v_h\|^2_{1,\Omega_h} \leq \mathcal{A}_h(\rho_h - v_h, \rho_h - v_h).$$

Then by setting $w_h = \rho_h - v_h$ and expanding linear terms and introducing the numerical method we get

$$C_{1,a_0,\mu_0} \|\rho_h - v_h\|^2_{1,\Omega_h} \leq l_h(w_h) + \mathcal{A}_h(p - v_h, w_h) - \mathcal{A}_h(p, w_h),$$

where $p \in \mathbb{P}_k(\mathcal{T}_h)$ is piecewise polynomial. Then adding and subtracting $\mathcal{B}(p, w_h)$ and $\mathcal{B}_h(p, w_h)$ provides

$$C_{1,a_0,\mu_0} \|\rho_h - v_h\|^2_{1,\Omega_h} \leq l_h(w_h) - \mathcal{B}(p, w_h) + \mathcal{A}_h(p - v_h, w_h) + [\mathcal{B}_h(p, w_h) - \mathcal{A}_h(p, w_h)] + [\mathcal{B}(p, w_h) - \mathcal{B}_h(p, w_h)].$$

Next we add and subtract $\mathcal{B}(\tilde{\rho}, w_h)$ to get

$$C_{1,a_0,\mu_0} \|\rho_h - v_h\|^2_{1,\Omega_h} \leq [l_h(w_h) - \mathcal{B}(\tilde{\rho}, w_h)] + \mathcal{B}(\tilde{\rho} - p, w_h) + \mathcal{A}_h(p - v_h, w_h) + [\mathcal{B}_h(p, w_h) - \mathcal{A}_h(p, w_h)] + [\mathcal{B}(p, w_h) - \mathcal{B}_h(p, w_h)].$$

We apply the continuity of the bilinear forms using Equation (6.25) of Theorem 6.6.1 and [39] to get,

$$\begin{aligned} C_{1,a_0,\mu_0} \|\rho_h - v_h\|^2_{1,\Omega_h} \leq\ & C_{2,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A},a,\mathbf{b},\mu} \left(\|\tilde{\rho} - p\|_{1,\Omega_h} \|w_h\|_{1,\Omega_h} + \|p - v_h\|_{1,\Omega_h} \|w_h\|_{1,\Omega_h} \right) \\ & + [l_h(w_h) - \mathcal{B}(\tilde{\rho}, w_h)] \\ & + [\mathcal{B}_h(p, w_h) - \mathcal{A}_h(p, w_h)] + [\mathcal{B}(p, w_h) - \mathcal{B}_h(p, w_h)] \end{aligned}$$

Dividing through by $\|w_h\|_{1,\Omega_h}$ and applying the triangle inequality gives

$$\begin{aligned} C_{1,a_0,\mu_0} \|\rho_h - v_h\|_{1,\Omega_h} \leq\ & C_{2,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A},a,\mathbf{b},\mu} \left(2 \|\tilde{\rho} - p\|_{1,\Omega_h} + \|\tilde{\rho} - v_h\|_{1,\Omega_h} \right) \\ & + \frac{|l_h(w_h) - \mathcal{B}(\tilde{\rho}, w_h)|}{\|w_h\|_{1,\Omega_h}} \\ & + \frac{|\mathcal{B}_h(p, w_h) - \mathcal{A}_h(p, w_h)|}{\|w_h\|_{1,\Omega_h}} + \frac{|\mathcal{B}(p, w_h) - \mathcal{B}_h(p, w_h)|}{\|w_h\|_{1,\Omega_h}}. \end{aligned}$$

Through integration by parts and applying the homogeneous Dirichlet boundary conditions we have

$$\mathcal{B}(\tilde{\rho}, w_h) = \int_{\Omega_h} f^* w_h \, d\mathbf{x}.$$

Substitution into the previous bound and further application of the triangle inequality by expanding terms over $\mathcal{T}_h$ completes the proof. □

Strang Error Terms

Here we break down the bounds of Theorem 6.7.1 into four components. The first term is bounded by standard projection and interpolation estimates. The

second term can be bounded by applying exactly the results of [39] to give a local bound. The third bound is given by the consistency estimate of Lemma 6.5.1. The fourth term is bounded in this section by Lemma 6.7.3.

Lemma 6.7.2.

Let $k, m \in \mathbb{N}$ and $\rho \in H^{m+1}(\Omega)$ be the solution to the PDE (5.1) with coefficients satisfying the regularity conditions of $a, \mathbf{b}, c \in W^{\infty}_{m+1}(\Omega)$. For the extension $\tilde{\rho} \in H^{m+1}(\Omega_h)$ from Ω to Ω_h, we have for all $E_h \in \mathcal{T}_h$ and a fixed integer $s \in \{0, 1, ..., k\}$ the following bound

$$\sup_{w_h \in V_k(E_h)\setminus\{0\}} \frac{\left|\mathcal{B}^{E_h}(\Pi^0_k \tilde{\rho}, w_h) - \mathcal{B}^{E_h}_h(\Pi^0_k \tilde{\rho}, w_h)\right|}{\|w_h\|_{1,E_h}} \lesssim h^s_{E_h} \|\tilde{\rho}\|_{s+1,E_h},$$

where the hidden constant depends on the shape regularity constant of E_h denoted by ϱ_{E_h}, k and the W^{∞}_{s+1} norms of $\tilde{a}$, $\tilde{\mathbf{b}}$ and $\tilde{\mu}$.

Proof. See Theorem 6.2 in [39]. □

Lemma 6.7.3.

Let $l, k \in \mathbb{N}$ and $\rho \in H^{m+1}(\Omega)$ be the solution to the PDE (5.1) with coefficients satisfying the regularity conditions of $a, \mathbf{b}, c \in W^{\infty}_{m+1}(\Omega)$. For the extensions $\tilde{\rho}, \tilde{f} \in H^{m+1}(\Omega_h)$, we have for all $E_h \in \mathcal{T}_h$, $w_h \in V_{h,0}$ and a fixed integer $s \in \{1, ..., \min\{k, l\}\}$, the following bound

$$\frac{|l_h(w_h) - l_*(w_h)|}{\|w_h\|_{1,\Omega_h}} \lesssim h^s \left(\|\tilde{\rho}\|_{s+1,\Omega_h} + \left\|\tilde{f}\right\|_{s-1,\Omega_h} \right),$$

where the hidden constant is dependent on N^{dofs}, the shape regularity parameters of $\mathcal{T}_h$ and $\hat{\mathcal{T}}_h$ denoted by ϱ and $\varrho_{\mathscr{A}}$, k, l, j_0 and the W^{∞}_{s+1} norms of $\mathscr{A}$, a, $\mathbf{b}$ and μ.

Proof. We prove the $k \geq 2$ case noting that the $k = 1$ case follows a near identical presentation. We rewrite the difference as

$$l_h(w_h) - l_*(w_h) = \sum_{E_h \in \mathcal{T}_h} \int_{E^{\pi}_h} \tilde{f}_h \, \Pi^0_{h,k} w_h \, d\mathbf{x} - \int_{E_h} f^* w_h \, d\mathbf{x}.$$

We can expand the second integral by adding and subtracting terms and using the orthogonality condition of the Π^0 projection

$$\int_{E_h} f^* w_h \, d\mathbf{x} = \int_{E_h} f_h^* \, \Pi_k^0 w_h \, d\mathbf{x} + \int_{E_h} (f^* - f_h^*) w_h \, d\mathbf{x},$$

where $f_h^* = \Pi_{h,k-2}^0 f^*$ for $k \geq 2$ and $f_h^* = \Pi_0^0 f^*$ for $k = 1$. Via the above expansion we can define the local difference terms by

$$l_h(w_h) - l_*(w_h) = \sum_{E_h \in \mathcal{T}_h} E1 + E2 + E3,$$

where,

$$\begin{aligned}
E1 &= \int_{E_h^\pi} \tilde{f}_h \Pi_{h,k}^0 w_h \, d\mathbf{x} - \int_{E_h} \tilde{f}_h \Pi_{h,k}^0 w_h \, d\mathbf{x}, \\
E2 &= \int_{E_h} (f_h^* - f^*) w_h \, d\mathbf{x}, \\
E3 &= \int_{E_h} \tilde{f}_h \Pi_{h,k}^0 w_h \, d\mathbf{x} - \int_{E_h} f_h^* \Pi_k^0 w_h \, d\mathbf{x}
\end{aligned}$$

$E1$ is bounded by applying Lemma 6.1.4 and the stability estimate of Theorem 6.3.1 to get

$$\begin{aligned}
|E1| &\leq C_{1,N^{\text{dofs}},\varrho,l,\mathscr{A},j_0} h_{\hat{E}}^s \left\| \tilde{f} \right\|_{0,E_h} \|w_h\|_{0,E_h} \\
&\leq C_{1,N^{\text{dofs}},\varrho,l,\mathscr{A},j_0} h_{\hat{E}}^s \left\| \tilde{f} \right\|_{0,E_h} \|w_h\|_{1,E_h} .
\end{aligned}$$

$E2$ is bounded by applying the Cauchy-Schwarz inequality, the triangle inequality and Theorems 2.1.1 and 6.3.1

$$\begin{aligned}
|E2| &\leq \left(\left\| f^* - \Pi_{k-2}^0 f^* \right\|_{0,E_h} + \left\| \Pi_{h,k-2}^0 f^* - \Pi_{k-2}^0 f^* \right\|_{0,E_h} \right) \|w_h\|_{0,E_h} \\
&\leq C_{2,N^{\text{dofs}},\varrho_{E_h},k,l,\mathscr{A},j_0} (h_{E_h}^s + h_{\hat{E}}^s) \|f^*\|_{s-1,E_h} \|w_h\|_{0,E_h} .
\end{aligned}$$

Applying Lemma 6.1.2 gives

$$|E2| \leq C_{3,N^{\text{dofs}},\varrho_{E_h},k,l,\mathscr{A},j_0,a,\mathbf{b},\mu} h_{\hat{E}}^s \|f^*\|_{s-1,E_h} \|w_h\|_{0,E_h} .$$

By the definition of f^* in Equation (6.27) we can further bound E2 via

$$|E2| \leq C_{3,N^{\text{dofs}},\varrho_{E_h},k,l,\mathscr{A},j_0,a,\mathbf{b},\mu} h_{\hat{E}}^s \|\tilde{\rho}\|_{s+1,E_h} \|w_h\|_{0,E_h} .$$

$E3$ is further split into two differences of

$$E3 = \int_{E_h} \tilde{f}_h(\Pi^0_{h,k} w_h - \Pi^0_k w_h)\ d\mathbf{x} - \int_{E_h} (\tilde{f}_h - f^*_h)\Pi^0_k w_h\ d\mathbf{x}.$$

The first integral is bounded by Applying the Cauchy-Schwarz inequality, Theorem 6.3.1 and the stability of Π^0 which results in

$$\begin{aligned}\int_{E_h} \tilde{f}_h(\Pi^0_{h,k} w_h - \Pi^0_k w_h)\ d\mathbf{x} &\leq C_{4,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A}} h^s_{\hat{E}} \left\|\tilde{f}\right\|_{0,E_h} \|w_h\|_{0,E_h} \\ &\leq C_{4,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A}} h^s_{\hat{E}} \left\|\tilde{f}\right\|_{0,E_h} \|w_h\|_{1,E_h}.\end{aligned}$$

The second integral is decomposed into

$$\int_{E_h \cap E} (\tilde{f}_h - f^*_h)\Pi^0_k w_h\ d\mathbf{x} + \int_{E_h \backslash E} (\tilde{f}_h - f^*_h)\Pi^0_k w_h\ d\mathbf{x}.$$

From Steins Extension theorem we have that $f = f^* = \tilde{f}$ a.e. on the set $E_h \cap E$. The above equation is bounded by Applying Hölder's inequality and following the proof of Lemma 6.1.1 to get

$$\begin{aligned}\left|\int_{E_h \backslash E} (\tilde{f}_h - f^*_h)\Pi^0_k w_h\ d\mathbf{x}\right| &\leq |E_h \backslash E| \left\|\tilde{f}_h - f^*_h\right\|_{0,\infty,E_h\backslash E} \left\|\Pi^0_k w_h\right\|_{0,\infty,E_h\backslash E} \\ &\leq C_{5,\varrho,l,\mathscr{A}} h^{s+2}_{\hat{E}} \left\|\tilde{f}_h - f^*_h\right\|_{0,\infty,E_h\backslash E} \left\|\Pi^0_k w_h\right\|_{0,\infty,E_h\backslash E} \\ &\leq C_{5,\varrho,l,\mathscr{A}} h^{s+2}_{\hat{E}} \left\|\tilde{f}_h - f^*_h\right\|_{0,\infty,E_h} \left\|\Pi^0_k w_h\right\|_{0,\infty,E_h}.\end{aligned}$$

Applying an inverse estimate on E_h and scaling h_{E_h} according to Lemma 6.1.2 gives

$$\left|\int_{E_h \backslash E} (\tilde{f}_h - f^*_h)\Pi^0_k w_h\ d\mathbf{x}\right| \leq C_{6,\varrho,l,\mathscr{A}} h^s_{\hat{E}} \left\|\tilde{f}_h - f^*_h\right\|_{0,E_h} \left\|\Pi^0_k w_h\right\|_{0,E_h}.$$

We then bound E3 using this inequality and applying the triangle inequality and the stability of Π^0 to get

$$\begin{aligned}|E3| &\leq C_{4,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A}} h^s_{\hat{E}} \left\|\tilde{f}\right\|_{0,E_h} \|w_h\|_{1,E_h} + C_{6,\varrho,l,\mathscr{A}} h^s_{\hat{E}} \left\|\tilde{f}_h - f^*_h\right\|_{0,E_h} \left\|\Pi^0_k w_h\right\|_{0,E_h} \\ &\leq C_{4,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A}} h^s_{\hat{E}} \left\|\tilde{f}\right\|_{0,E_h} \|w_h\|_{1,E_h} + C_{6,\varrho,l,\mathscr{A}} h^s_{\hat{E}} \left(\left\|\tilde{f}_h\right\|_{0,E_h} + \|f^*_h\|_{0,E_h}\right) \left\|\Pi^0_k w_h\right\|_{0,E_h} \\ &\leq C_{4,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A}} h^s_{\hat{E}} \left\|\tilde{f}\right\|_{0,E_h} \|w_h\|_{1,E_h} + C_{6,\varrho,l,\mathscr{A}} h^s_{\hat{E}} \left(\left\|\tilde{f}\right\|_{0,E_h} + \|f^*\|_{0,E_h}\right) \|w_h\|_{1,E_h} \\ &\leq C_{7,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A}} h^s_{\hat{E}} \left(\left\|\tilde{f}\right\|_{0,E_h} + \|f^*\|_{0,E_h}\right) \|w_h\|_{1,E_h}\end{aligned}$$

We conclude the bounding of $E3$ by applying the definition of f^* in Equation (6.27)

$$|E3| \leq C_{8,N^{\text{dofs}},\varrho,l,j_0,\mathscr{A},a,\mathbf{b},\mu} h_{\hat{E}}^s \left(\left\| \tilde{f} \right\|_{0,E_h} + \|\tilde{\rho}\|_{2,E_h} \right) \|w_h\|_{1,E_h} .$$

Combining the bounds for $E1$, $E2$ and $E3$ and gives

$$|l_h(w_h) - l_*(w_h)| \leq \sum_{E_h \in \mathcal{T}_h} C_{9,N^{\text{dofs}},\varrho,\varrho_{E_h},k,l,j_0,\mathscr{A},a,\mathbf{b},\mu} h_{\hat{E}}^s \left(\|\tilde{\rho}\|_{s+1,E_h} + \left\| \tilde{f} \right\|_{s-1,E_h} \right) \|w_h\|_{1,E_h} ,$$

From which the proof is concluded by applying triangle inequality, bounding norms and dividing by $\|w_h\|_{1,\Omega_h}$. □

The H^1 Error estimate

Theorem 6.7.4 (H^1 Error Estimate for Method II)**.**
Let $l, k \in \mathbb{N}$ and $\rho \in H^{m+1}(\Omega)$ be the solution to the PDE (5.1) with coefficients satisfying the regularity conditions of $a, \mathbf{b}, c \in W^{\infty}_{m+1}(\Omega)$. For the extension $\tilde{\rho} \in H^{m+1}(\Omega_h)$, and the VEM of degree l and k outlined in section 6.4 we have the following error bound for a fixed integer $s \in \{1, ..., \min\{l, k\}\}$ and sufficiently small h

$$\|\tilde{\rho} - \rho_h\|_{1,\Omega_h} \lesssim h^s \left(\|\rho\|_{s+1,\Omega} + \|f\|_{s-1,\Omega} \right).$$

Proof. Firstly, we add and subtract the global VEM interpolant of the extension $\tilde{\rho}_I$. Using the triangle inequality and applying Theorem 6.2.1 provides,

$$\begin{aligned} \|\tilde{\rho} - \rho_h\|_{1,\Omega_h} &\leq \|\tilde{\rho} - \tilde{\rho}_I\|_{1,\Omega_h} + \|\tilde{\rho}_I - \rho_h\|_{1,\Omega_h} \\ &\leq C_{1,k,\varrho_{\mathscr{A}},\mathscr{A}} h^s \|\tilde{\rho}\|_{s+1,\Omega_h} + \|\tilde{\rho}_I - \rho_h\|_{1,\Omega_h} . \end{aligned} \tag{6.28}$$

The second term is bounded by applying Theorem 6.7.1 and considering $p = \Pi^0_k \tilde{\rho}$ and $v_h = \tilde{\rho}_I$ with the constant of Theorem 6.7.1 denoted by

$$C_{strang} := C_{strang,N^{\text{dofs}},\varrho,l,a_0,\mu_0,j_0,a,\mathbf{b},\mu,\mathscr{A}},$$

$$\begin{aligned}\frac{1}{C_{strang}}\left\|\rho_h-\tilde{\rho}_I\right\|_{1,\Omega_h} \leq \left\|\tilde{\rho}-\tilde{\rho}_I\right\|_{1,\Omega_h} + \left\|\tilde{\rho}-\Pi_k^0\tilde{\rho}\right\|_{1,\Omega_h} \\ + \sum_{E_h\in\mathcal{T}_h}\sup_{w_h\in V_k(E_h)\setminus\{0\}}\frac{\left|\mathcal{B}^{E_h}(\Pi_k^0\tilde{\rho},w_h)-\mathcal{B}_h^{E_h}(\Pi_k^0\tilde{\rho},w_h)\right|}{\left\|w_h\right\|_{1,E_h}} \\ + \sum_{E_h\in\mathcal{T}_h}\sup_{w_h\in V_k(E_h)\setminus\{0\}}\frac{\left|\mathcal{B}_h^{E_h}(\Pi_k^0\tilde{\rho},w_h)-\mathcal{A}_h^{E_h}(\Pi_k^0\tilde{\rho},w_h)\right|}{\left\|w_h\right\|_{1,E_h}} \\ + \sup_{w_h\in V_h\setminus\{0\}}\frac{\left|l_h(w_h)-l_*(w_h)\right|}{\left\|w_h\right\|_{1,\Omega_h}}.\end{aligned}$$

From this we again apply Theorems 6.2.1, 2.1.1 and scale the mesh size according to Lemma 6.1.2 to bound the first two terms. Lemmas 6.7.2, 6.5.1 and 6.7.3 are applied for the third, fourth and fifth terms respectively. This results in

$$\begin{aligned}\frac{1}{C_{strang}}\left\|\rho_h-\tilde{\rho}_I\right\|_{1,\Omega_h} \leq C_{2,l,k,\varrho,\varrho_{\mathscr{A}},\mathscr{A}}h^s\left\|\tilde{\rho}\right\|_{s+1} + \sum_{E_h\in\mathcal{T}_h}C_{3,\varrho_{E_h},k,a,\mathbf{b},\mu}h_{E_h}^s\left\|\tilde{\rho}\right\|_{s+1,E_h} \\ + \sum_{E_h\in\mathcal{T}_h}C_{4,N^{\text{dofs}},\varrho,l,j_0,a,\mathbf{b},\mu,\mathscr{A}}h_{\hat{E}}^s\left\|\Pi_k^0\tilde{\rho}\right\|_{s+1,E_h} \\ + C_{5,\varrho,\varrho_{\mathscr{A}},l,k,j_0,\mathscr{A},a,\mathbf{b},\mu}h^s\left(\left\|\tilde{\rho}\right\|_{s+1,\Omega_h}+\left\|\tilde{f}\right\|_{s-1,\Omega_h}\right).\end{aligned}$$

Applying Lemma 6.1.2, the stability of Π^0 and bounding local terms gives

$$\frac{1}{C_{strang}}\left\|\rho_h-\tilde{\rho}_I\right\|_{1,\Omega_h} \leq C_{6,N^{\text{dofs}},\varrho,\varrho_{\mathscr{A}},l,k,j_0,\mathscr{A},a,\mathbf{b},\mu}h^s\left(\left\|\tilde{\rho}\right\|_{s+1,\Omega_h}+\left\|\tilde{f}\right\|_{s-1,\Omega_h}\right).$$

Substitution into Equation (6.28) gives

$$\begin{aligned}\left\|\rho_h-\tilde{\rho}\right\|_{1,\Omega_h} \leq C_{1,k,\varrho_{\mathscr{A}},\mathscr{A}}h^s\left\|\tilde{\rho}\right\|_{s+1,\Omega_h} \\ + C_{strang}C_{6,N^{\text{dofs}},\varrho,\varrho_{\mathscr{A}},l,k,j_0,\mathscr{A},a,\mathbf{b},\mu}h^s\left(\left\|\tilde{\rho}\right\|_{s+1,\Omega_h}+\left\|\tilde{f}\right\|_{s-1,\Omega_h}\right)\end{aligned}$$

The proof is concluded by applying Stein's Extension Theorem 5.3.1. □

Chapter 7

Numerical Investigations

In this chapter we present numerical experiments to validate the theory of Chapters 5 and 6. A series of domain transformations are defined in Section 7.1. These are tested for the interpolation and projection of the solution data onto the computation and virtual meshes $\hat{\mathcal{T}}_h$, $\mathcal{T}_h$ for Methods I and II respectively. Both methods are then tested against two second-order elliptic PDEs. Convergence results are presented in Section 7.2. The numerical experiments of this chapter may involve PDEs with non-zero boundary conditions. In these instances, Dirichlet boundary conditions are enforced using the interpolant of the DoFs of the true solution. Furthermore, the forcing data is prescribed by applying symbolic differentiation to the known solution using the DUNE-UFL library [48], as such explicit details of the forcing data is not provided in this chapter.

7.1 Example Mappings

In this series of numerical experiments we use three different domain transformations to validate the estimates of Chapters 5 and 6. In this Chapter we refer to these mappings as the "Diamond", "CE" and "Warped Square" maps. In all tests the reference domain is taken as $\hat{\Omega} = [0, 1]^2$.

Diamond Map

The Diamond mapping serves as a patch test to validate the numerical methods perform as expected under linear transformations. The mapping is defined by

$$\mathscr{A}(\boldsymbol{\xi}) = \left[\frac{\xi+\eta}{2},\ \frac{-\xi+\eta}{2}\right].$$

The Jacobian and projection approximations of this mapping are exact for $l \geq 1$. As seen in Section 7.2 the expectation is that both numerical methods will exhibit optimal orders of accuracy with respect to the solution degree k and remain independent of the transformation degree l.

The CE Map

The CE mapping is taken from the works of Lipnikov & Morgan [78, 80]. This is a cubic polynomial mapping defined as

$$\mathscr{A}(\boldsymbol{\xi}) = [\xi + \xi\eta(1-\xi)/2,\ \eta + \xi\eta(1-\eta)/2]\,. \tag{7.1}$$

Additionally, this mapping is the identity on the boundary of $\hat{\Omega}$ and only transforms internal edges and vertices.

The Warped Square Map

The final mapping considered is the Warped Square defined by

$$\mathscr{A}(\boldsymbol{\xi}) = \left[\sin\frac{\xi\pi}{3},\ e^{\eta}\right]. \tag{7.2}$$

This choice of mapping is distinct from the previous two choices as Equation (7.2) is a non-polynomial and non-linear translation between $\hat{\Omega}$ and Ω.

7.2 Convergence Results in Two Dimensions

For each convergence test we consider a sequence of 5 Centroidal Voronoi Tessellations [91] of $\hat{\Omega} = [0,1]^2$. The global mesh size roughly halves each time the number of polygonal elements is quadrupled in the mesh sequence. The numerical experiments are performed within the DUNE software environment. The

mesh files are generated using PolyMesher within MATLAB [94] and imported into DUNE.

The forcing term f is chosen in each test such that the exact solution, in the physical coordinate system, is given by

$$\rho(\mathbf{x}) = cos(xy)sin(x). \tag{7.3}$$

These tests go beyond the scope of the theory on two accounts: we assess both the H^1 and L^2 error norms for Methods I and II, all problems considered have non-homogeneous Dirichlet boundary conditions.

To compare the accuracy of the two methods, we consider discrete H^1 and L^2 semi-norms that approximate the solution error on Ω_h. These are defined as the H^1 and L^2 norms respectively as

$$\|\rho_h - \rho\|^2_{h,1} := \sum_{\hat{E}\in\hat{\mathcal{T}}_h} \int_{\hat{E}} \left|\mathbf{J}^{-\top}_{\mathscr{A},h}\ \Pi^1_{k-1}(\hat{\rho}_h - \hat{\rho})\right|^2\ j_{h,n}\ d\boldsymbol{\xi}, \tag{7.4}$$

$$\|\rho_h - \rho\|^2_{h,0} := \sum_{\hat{E}\in\hat{\mathcal{T}}_h} \int_{\hat{E}} \left|\ \Pi^0_{k}(\hat{\rho}_h - \hat{\rho})\right|^2\ j_{h,n}\ d\boldsymbol{\xi}, \tag{7.5}$$

for Method I. Similarly, for Method II we estimate the errors via

$$\|\rho_h - \rho\|^2_{h,1} := \sum_{E_h\in\mathcal{T}_h} \int_{E^\pi_h} \left|\Pi^1_{h,k-1}(\rho_h - \tilde{\rho})\right|^2\ d\mathbf{x},$$

$$\|\rho_h - \rho\|^2_{h,0} := \sum_{E_h\in\mathcal{T}_h} \int_{E^\pi_h} \left|\ \Pi^0_{h,k}(\rho_h - \tilde{\rho})\right|^2\ d\mathbf{x}.$$

Interpolation & Projection

The first sequence of tests assess the accuracy of Interpolating the solution data onto the computational domains. We consider a Lagrange-type interpolation by assessing the DoFs of $\mathscr{A}$ and a global VEM L^2 projection approach.

In the case of the Diamond map, we observe in Figure 7.1 that optimal orders of accuracy of $O(h^k)$ in H^1 and $O(h^{k+1})$ in L^2 are achieved for both Methods I and II. This is expected for the reasons outlined in Section 7.1 as the mapping requires no approximation. Indeed we obtain identical numerical results when approximating $\mathscr{A}$ with a linear VEM space. As such we omit

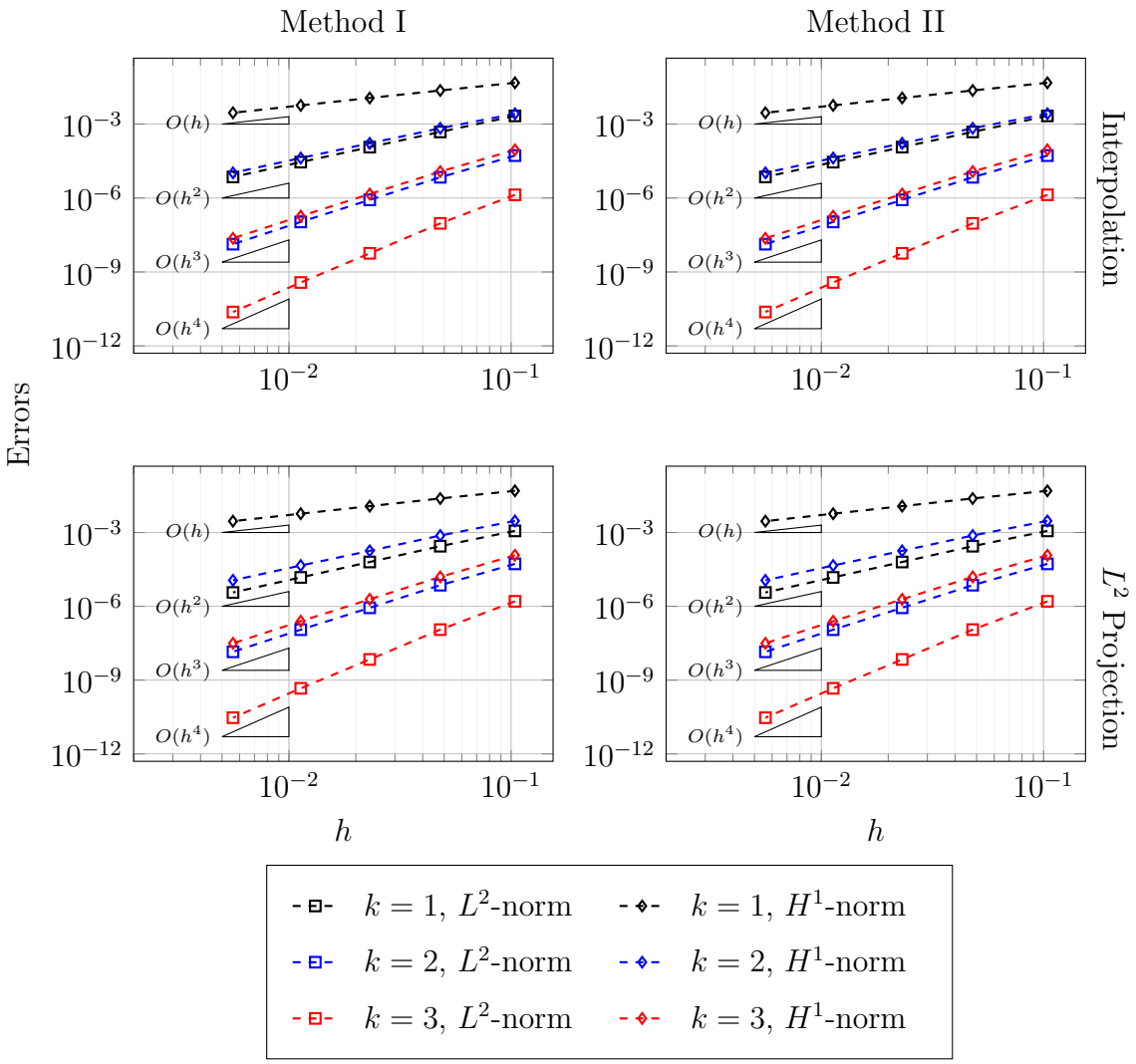

Figure 7.1: Interpolation and Projection error plots using the Diamond mapping approximated using the linear VEM ($l = 1$).

the data for $l = 2, 3$ from the presentation, this will also be the case for the remainder of this section for the Diamond mapping.

The CE mapping presents the first instance in which the convergence rates are limited by the minimum of l and k, as expected by Theorems 5.8.1 and 6.7.4 for Method I and Method II respectively. This behaviour is observed only for Method II in which the data approximation is limited to $O(h^{l+1})$ in the L^2 norm and $O(h^l)$ in the H^1 norm. From Figures 7.2 and 7.3 we observe this effect for the $l = 1, 2$ cases, taking note that the CE map is exact for $l = 3$, in particular we observe sub-optimal convergence rates for the case in

which $k = 3, l = 2$. In contrast Method I maintains $O(h^k)$ and $O(h^{k+1})$ H^1 and L^2 errors independent of l. This difference between Methods I and II is explained through the implementation within DUNE. In Method I, the solution data is given as a continuous UFL [4] expression transformed to the reference configuration and consequently is independent of the mapping discretization degree l. For Method II, the DoFs of the solution data are actually interpolated onto the set projected elements $\{E_h^\pi\}$ which is an approximation of degree l of the physical domain Ω. The reductions in convergence rates seen in method II for $k = 3, l = 2$ requires further investigation. The same conclusions are drawn for the Warped Square mapping for both the interpolation and L^2 projection. The numerical results are presented in Figures 7.4 and 7.5 respectively.

A Reaction-diffusion Problem

In this experiment we consider the simplified PDE defined when $a = c = 1$ and $\mathbf{b} = \mathbf{0}$. This is referred to as the Reaction-diffusion problem as can be expressed as

$$\begin{aligned} -\Delta\rho + \rho &= f \qquad \mathbf{x} \in \Omega, \\ \rho &= g \qquad \mathbf{x} \in \partial\Omega, \end{aligned}$$

for some function $g \in H^{1/2}(\partial\Omega)$. In each numerical test f, g are chosen such that the true solution is given by Equation (7.3).

As presented in Figure 7.6, the Diamond transformation produces optimal order $O(h^k)$ and $O(h^{k+1})$ results in the H^1 and L^2 error norms for $l = 1$ with identical results for $l = 2, 3$. A similar outcome is noted between the CE and Warped Square mappings in Figures 7.7 and 7.8 respectively. We notice that the convergence rate for both Method I and Method II is now similarly limited to $\min(l, k)$ in the H^1 norm, in agreement with the estimates of Theorems 5.8.1 and 6.7.4. A distinction between Method I and Method II is that in Method I we see that for $k = 3, l = 2$ that we maintain second and third order accuracy

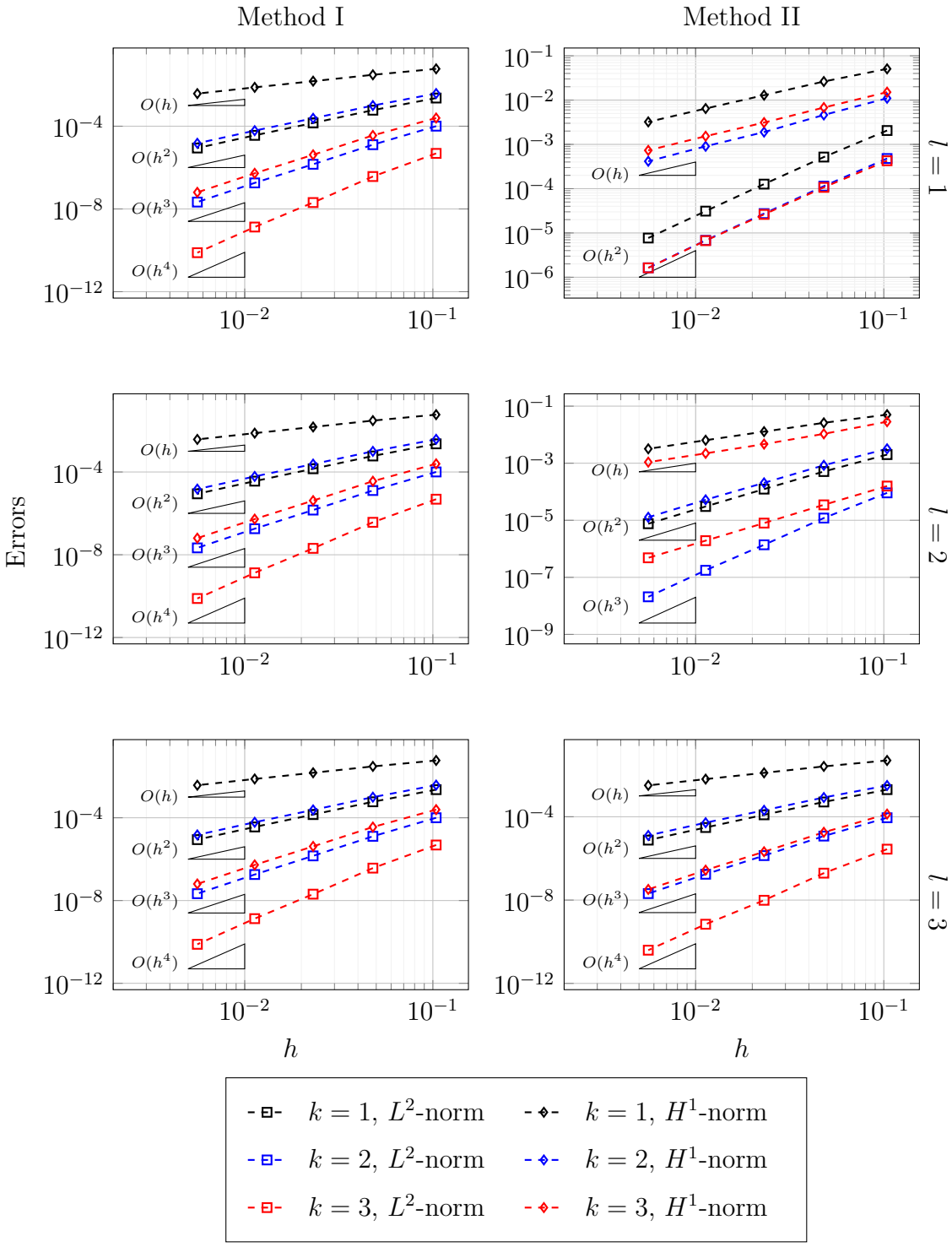

Figure 7.2: Interpolation error plots using CE mapping approximations of degree $l = 1, 2, 3$.

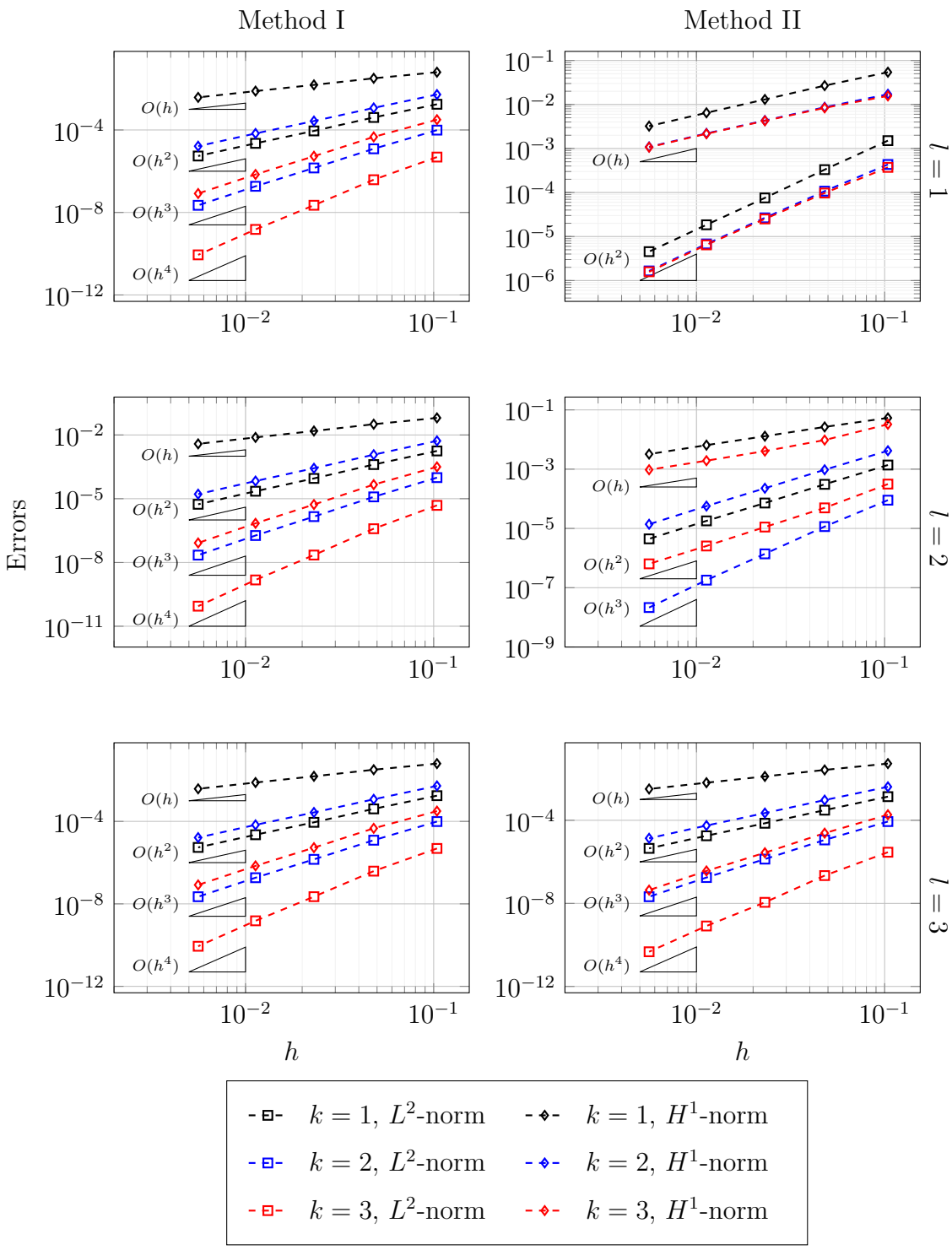

Figure 7.3: L^2 projection error plots using CE mapping approximations of degree $l = 1, 2, 3$.

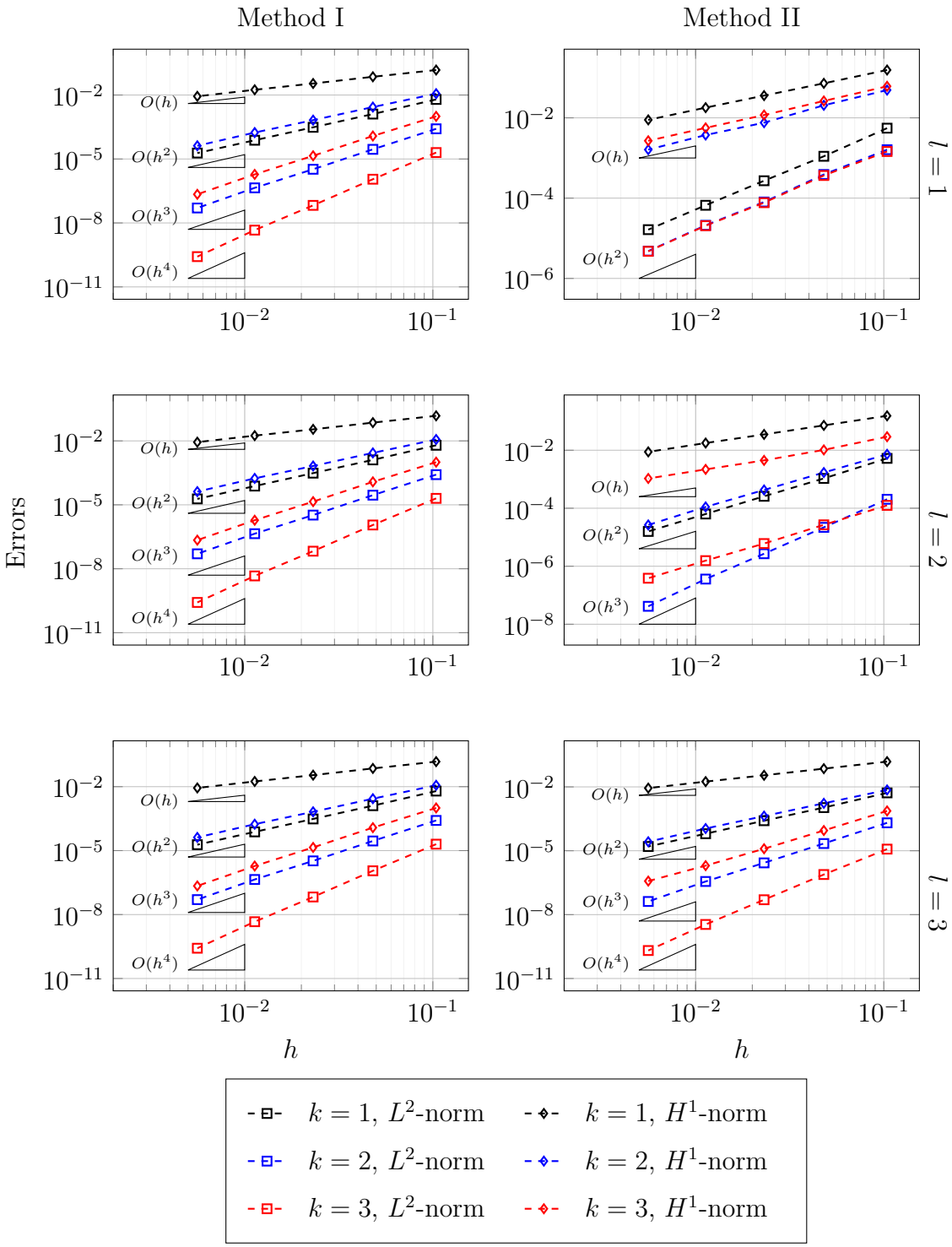

Figure 7.4: Interpolation error plots using Warped Square mapping approximations of degree $l = 1, 2, 3$.

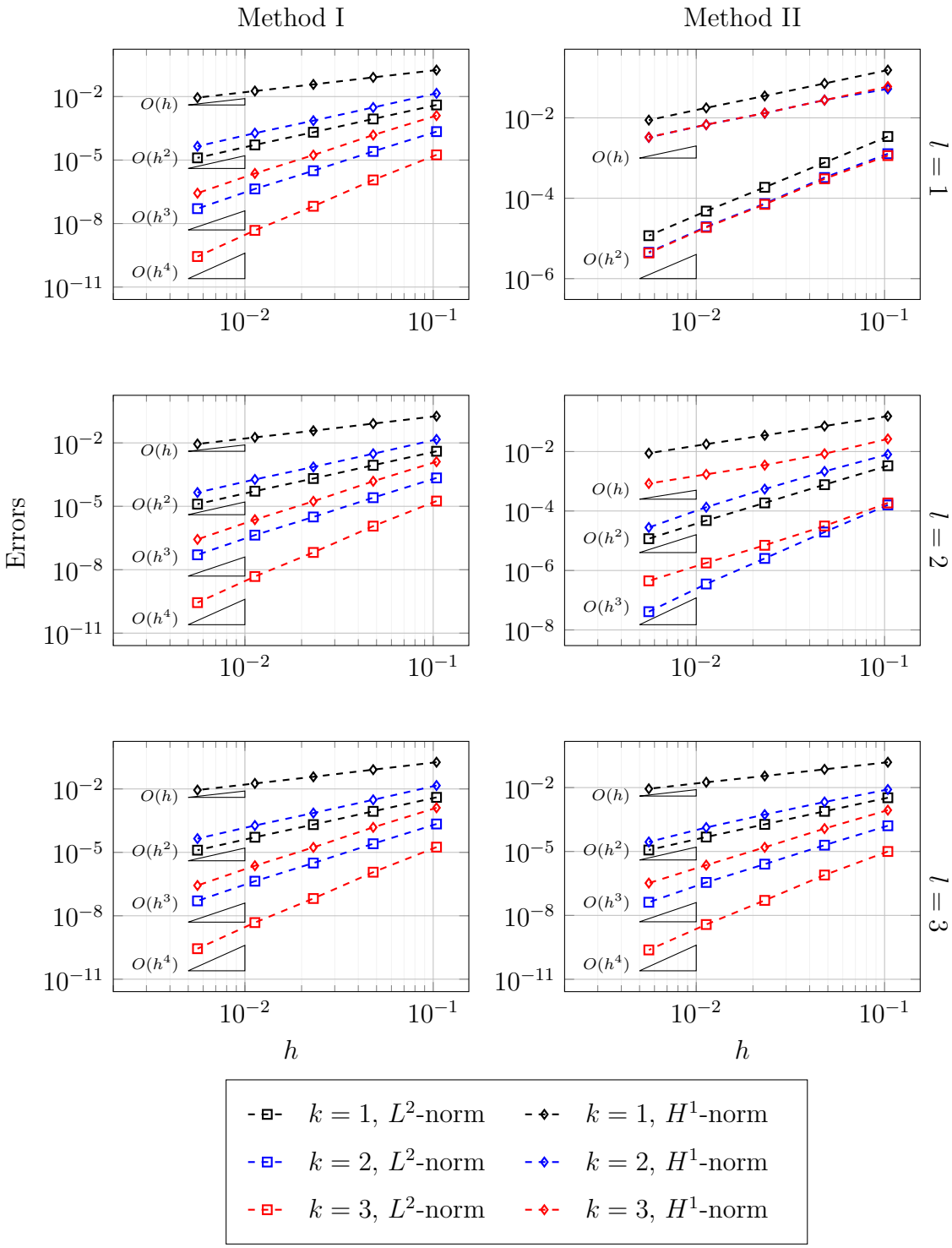

Figure 7.5: L^2 projection error plots using Warped Square mapping approximations of degree $l = 1, 2, 3$.

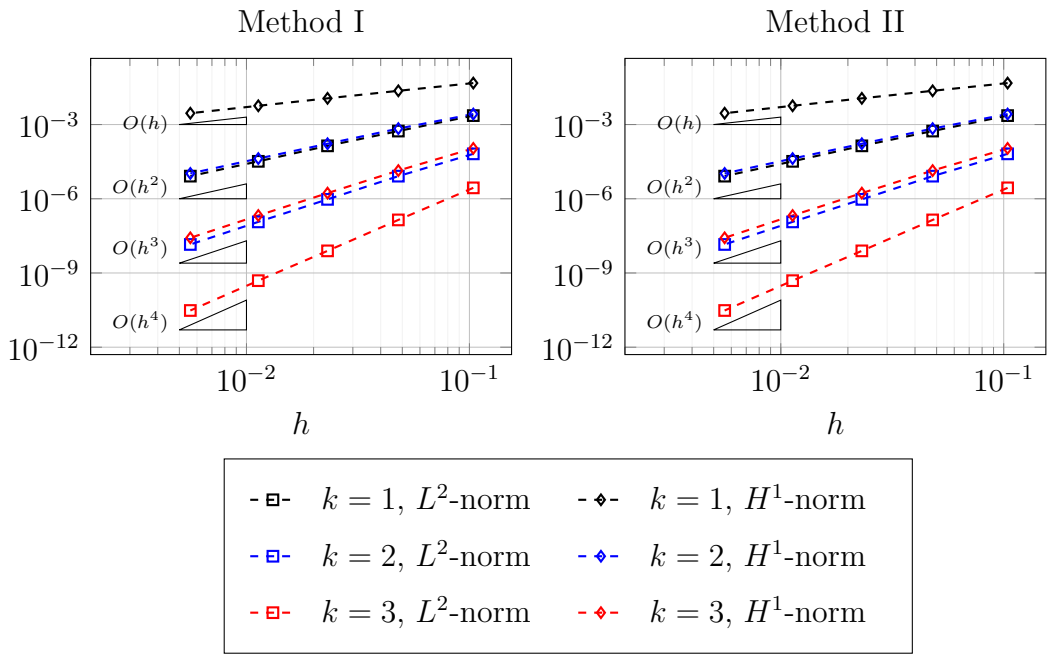

Figure 7.6: Error plots for the Reaction-diffusion problem using a Diamond mapping approximation of degree $l = 1$.

for the H^1 and L^2 errors whereas Method II still has sub-optimal convergence rates as seen in the interpolation and projection experiments.

A General Elliptic PDE

In the final experiment we consider a second-order elliptic PDE with inhomogenous Dirichlet boundary conditions. Following the numerical example from [20], the PDE (5.1) is defined by

$$a = \begin{pmatrix} y^2 + 1 & -xy \\ -xy & x^2 + 1 \end{pmatrix}, \qquad \mathbf{b} = \mathbf{x}, \qquad c = x^2 + y^3,$$

with f and the Dirichlet boundary conditions given such that the solution in the physical coordinates satisfies Equation (7.3). The same convergence behaviour discussed in the Reaction-diffusion problem is observed in Figures 7.9, 7.10 and 7.11 for the Diamond, CE and Warped Square mappings respectively.

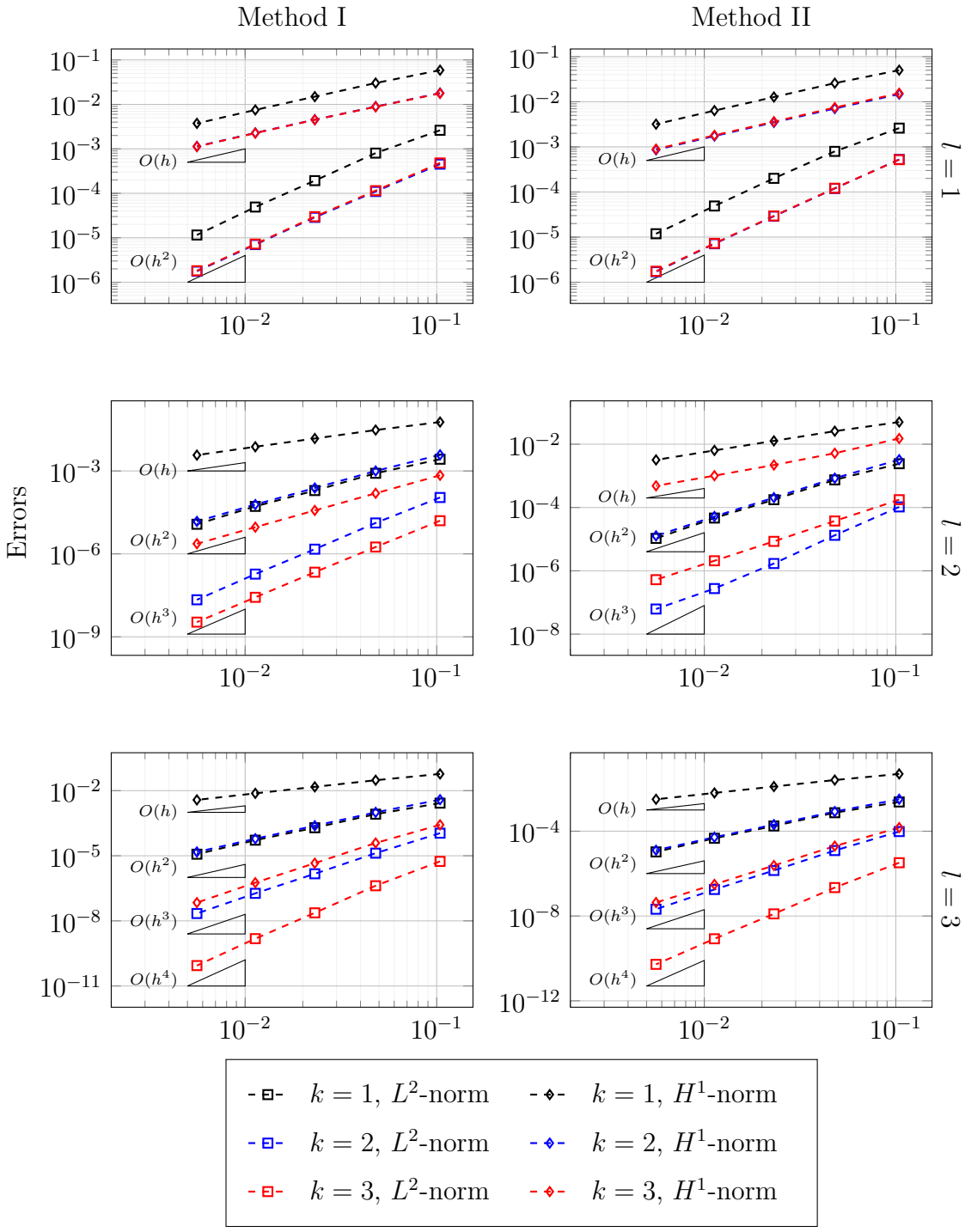

Figure 7.7: Error plots for the Reaction-diffusion problem using CE mapping approximations of degree $l = 1, 2, 3$.

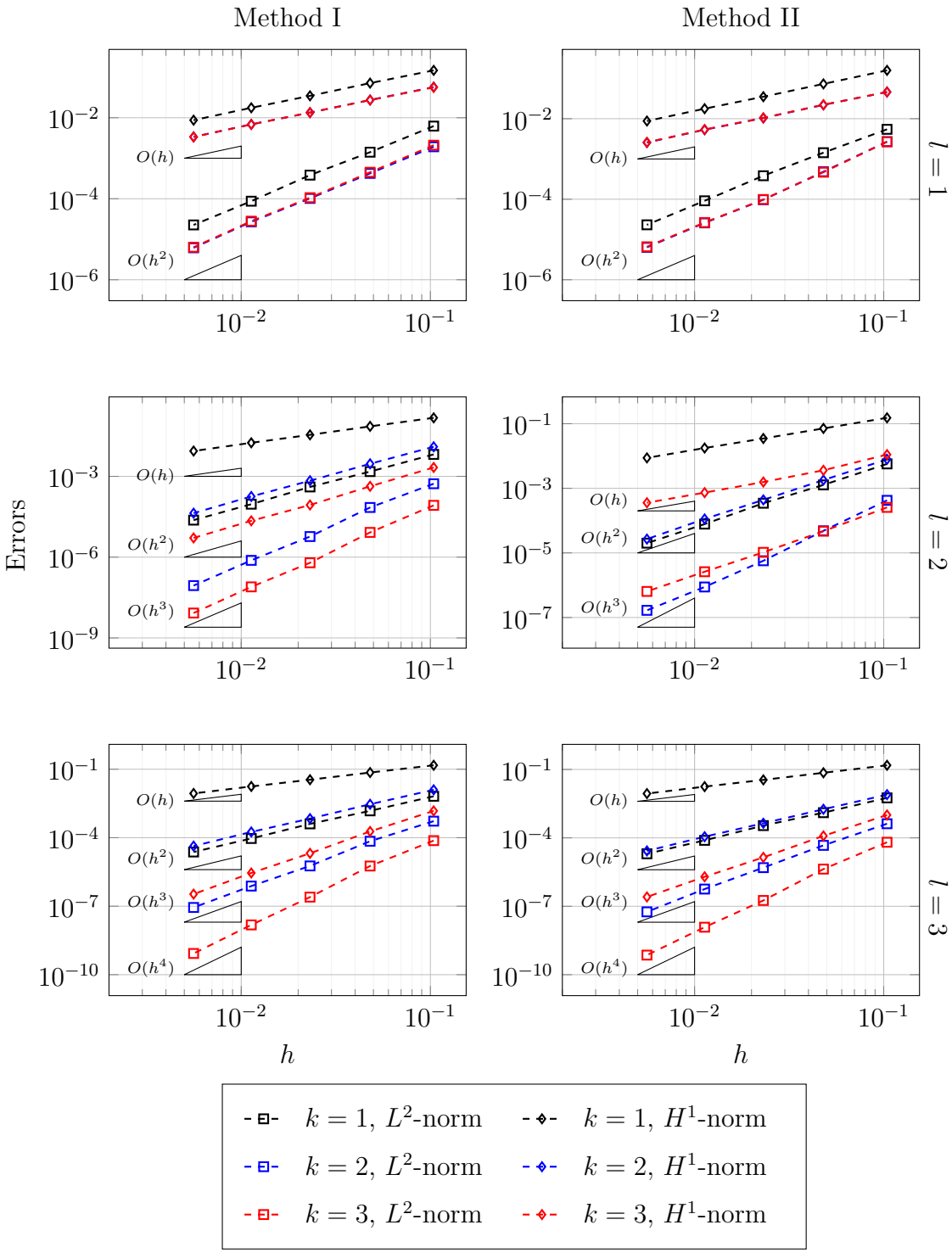

Figure 7.8: Error plots for the Reaction-diffusion problem using Warped Square mapping approximations of degree $l = 1, 2, 3$.

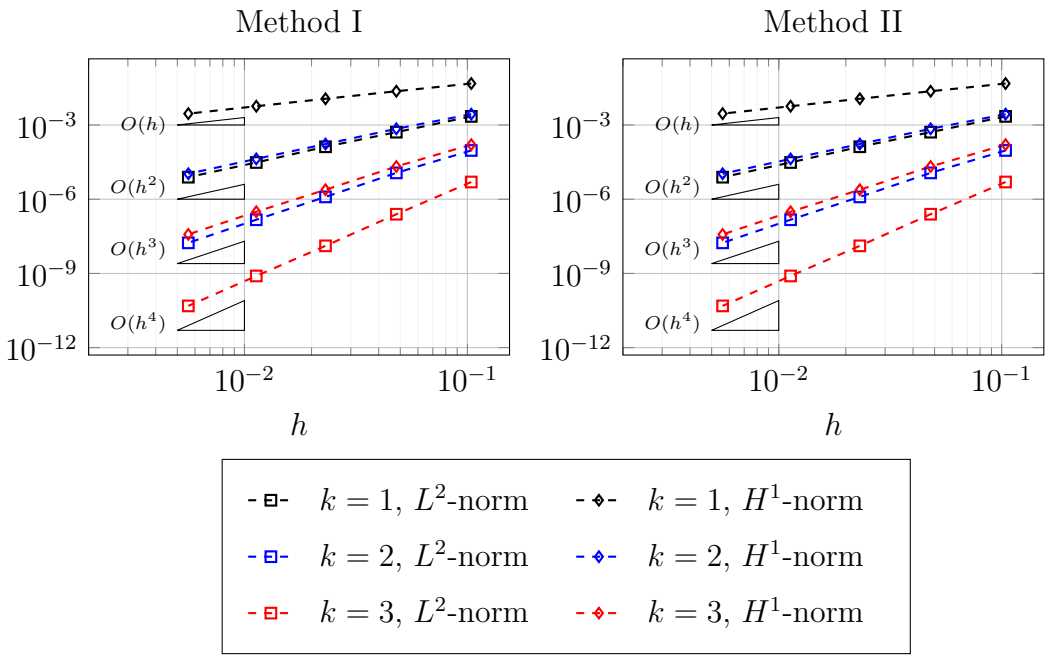

Figure 7.9: Error plots for the general elliptic PDE using a Diamond mapping approximation of degree $l = 1$.

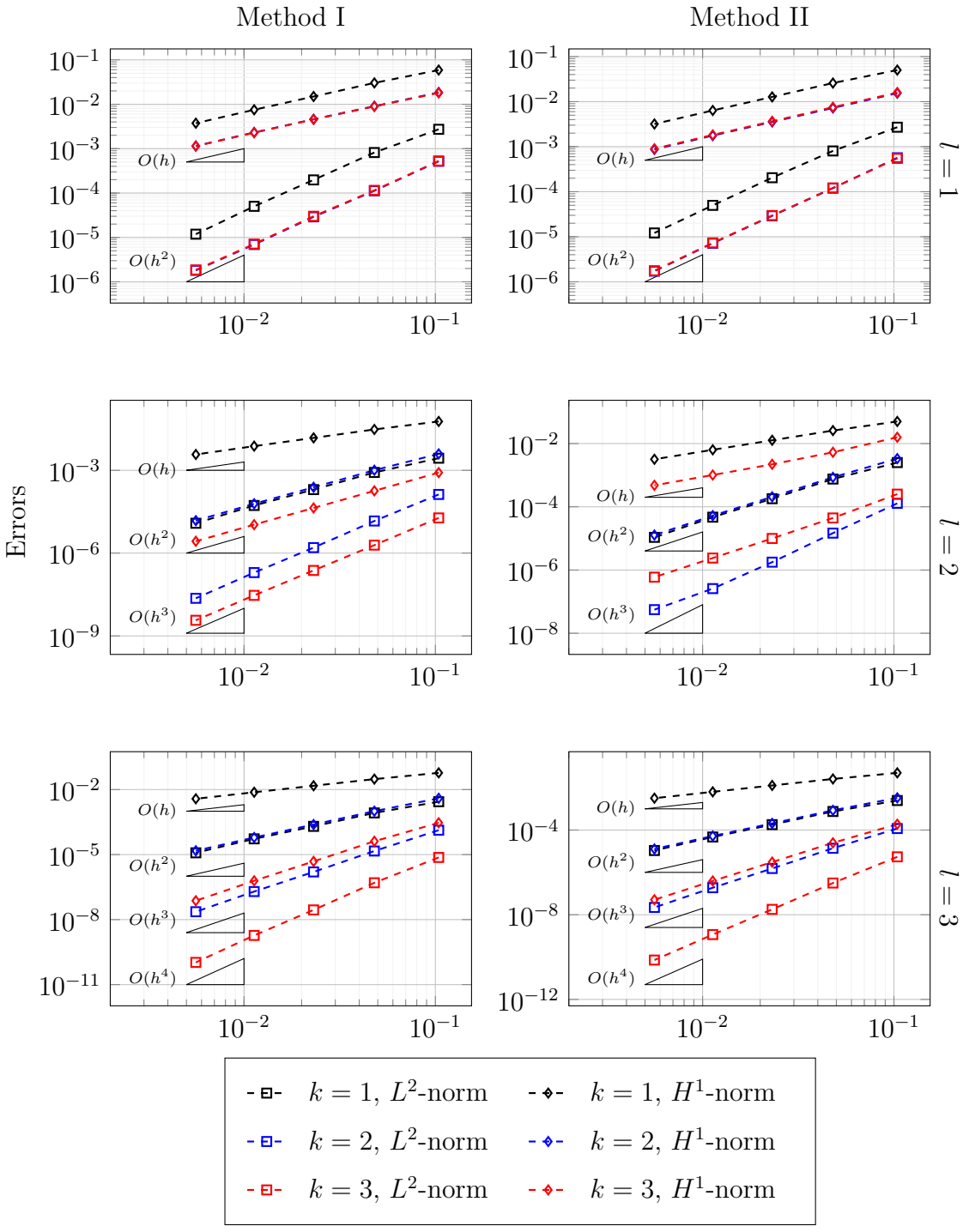

Figure 7.10: Error plots for the general elliptic PDE using CE mapping approximations of degree $l = 1, 2, 3$.

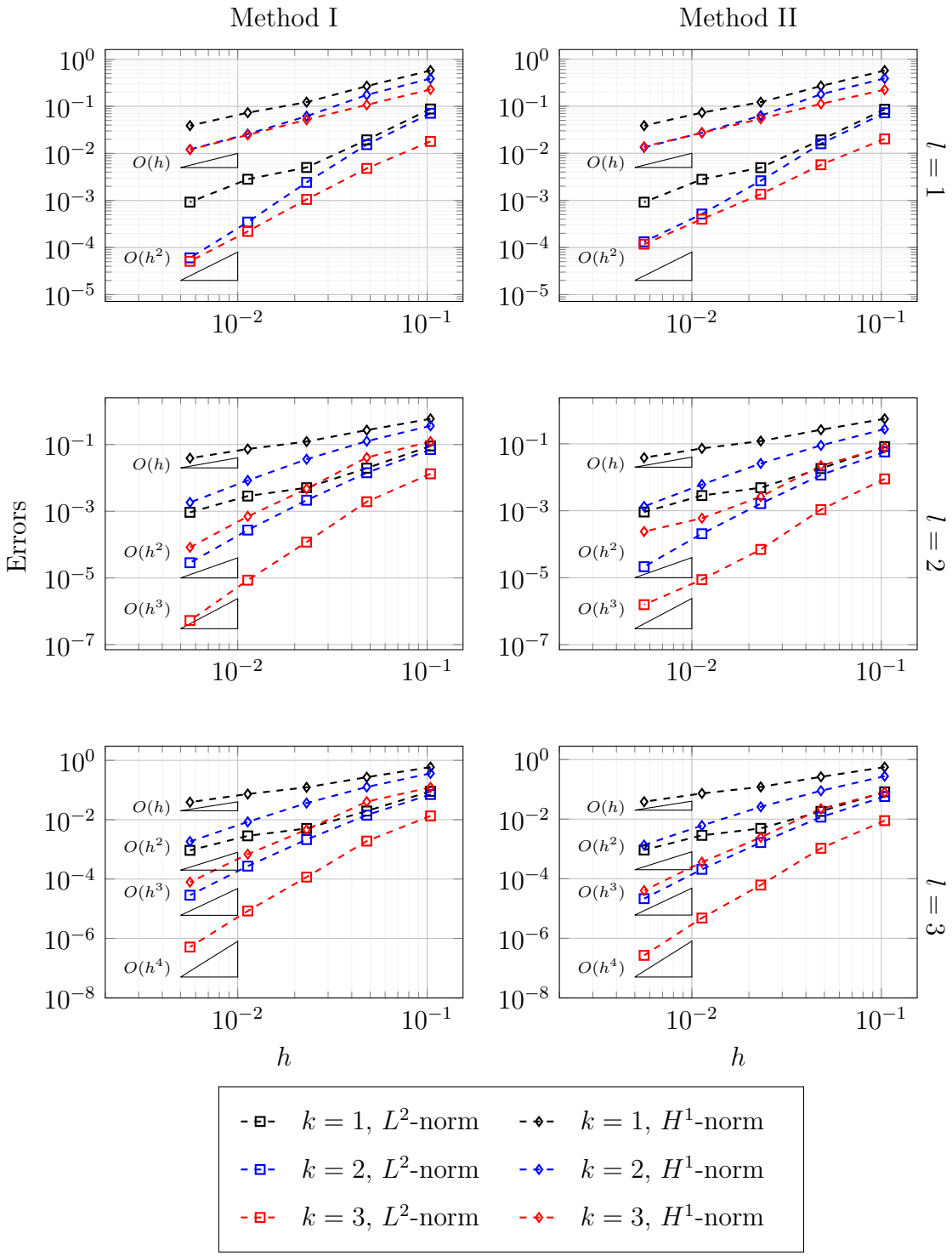

Figure 7.11: Error plots for the general elliptic PDE using Warped Square mapping approximations of degree $l = 1, 2, 3$.

Discussion

Two approaches to isoparametric VEMs have been presented in Part II. In each case an optimal order H^1 *a priori* estimate is derived and these have been verified, along with optimal L^2 orders of convergence, on a series of numerical tests in two dimensions.

Both methods provide the groundwork to formally develop moving mesh VEMs. In particular the formulation of Method II can be seen as extension of the approach taken in Part I. The VEM of Part I can be viewed as the $k = l = 1$ case of Method II where the projected elements are defined by an interpolation of the vertices of a moving polygon. Method I also offers a formal analysis which in combination with the reviewed works of Lipnikov & Morgan in Section 2.3 can provide a moving mesh method. This is studied in Part III.

The analysis of these isoparametric methods can include three-dimensional elliptic problems by extending the technical results of Chapters 5 and 6 to three-dimensional domains. In the case of Method I, the results of Section 5.4 can be extended to the three-dimensional setting through simple inductive arguments. The H^1 estimate can then be deduced following the remaining arguments of Chapter 5. In method II, the definition of the discrete projections extend to three-dimensional curved polytopes. The use of curved three-dimensional VEMs is not well developed in the literature and extending the analysis of Method II would be dependent on ongoing works in the curved VEM research community. For example, an extension of the interpolation estimate of Theorem 6.2.1 and the projection approximation results of Theorems 6.3.1 and 6.3.2

would require extension to curved polyhedral elements.

The construction of optimal L^2 error estimates for both Methods remains an open problem. The use of a classical Aubin-Nitsche duality arguments used to derive L^2 estimates for VEMs [20, 39] does not easily extend to these methods.

In the construction of both methods, a significant restriction is placed on the regularity of the domain transformation through the Lipschitz continuity condition. Whilst this is a standard assumption made in ALE analysis, and has the additional benefit of simplifying the analysis, the choices of domain transformations are more restricted than in isoparametric FEMs [42, 75]. Preliminary numerical investigations found that in the case of a simple non-smooth transformation $\mathscr{A} \in C^0(\hat{\Omega}) \backslash C^1(\hat{\Omega})$ that the orders of convergence are limited to $O(h^{3/2})$ for both Methods I and II. Additional analysis is required to explain this observed behaviour.

Part III

A Virtual Element ALE Method

Introduction

In the final Part of this thesis a Virtual Element Method is formulated, analysed and implemented for a conservative ALE scheme. Motivated by the successful implementation of a linear VEM for a moving mesh algorithm in Part I, the isoparametric VEMs studied in Part II are utilised in the computation and analysis of this ALE scheme.

The analysis of FEM-ALE schemes is well developed. Early works provide a formulation and analysis of a linear and quadratic finite element approaches [54, 84, 60]. In more recent works a discontinuous-Galerkin method was proposed and analysed [27, 28]. These papers form the template upon which the Virtual Element Method Arbitrary Lagrangian-Eulerian (VEM-ALE) scheme is designed and analysed. As this is the first instance of a VEM-ALE scheme, the scope of Part III is limited to problems of reduced complexity. These include: only considering prescribed fluid velocity fields, problems with a prescribed ALE velocity and domain transformation and a conservative ALE weak formulation only. Following the results of Part II, Method I of Chapter 5 will be used in the formulation with the domain and solution polynomial degrees being equal ($l = k$).

In Chapter 8, the definition of the time-dependent ALE mapping and velocity field are presented and the advection-diffusion problem is defined. Then a VEM using the isoparametric VEM of Chapter 5 is employed to derive a moving mesh VEM-ALE scheme.

In Chapter 9 the VEM-ALE scheme is tested against a series of advection-

diffusion problems and domain transformations. Numerical results suggest the accuracy of the isoparametric VEM from Part II extends to moving mesh problems. In addition to this, numerical experiments are conducted to test the extension of the isoparametric VEM to the velocity-based moving mesh VEM of Part I.

Chapter 8

A Virtual Element Arbitrary Lagrangian-Eulerian Scheme

In this chapter we propose an VEM-ALE scheme for an advection-diffusion problem with a prescribed convective fluid velocity field. Using the framework of Method I from Chapter 5 all components of the method: the solution, the domain transformation and the ALE velocity field are considered on a VEM space of degree $k \in \mathbb{N}$. Following elementary approaches in the literature, we consider first a semi-discretization of this method then present a fully discrete numerical method using a standard θ-scheme to integrate with respect to time [54, 60, 89].

In Section 8.1 we define the advection-diffusion problem and a corresponding ALE formulation. The moving VEM space is defined in Section 8.2 from which a semi-discretisation using the isoparametric VEM of Chapter 5 is formulated within Section 8.3. The fully discrete method is presented in Section 8.4.

8.1 Problem Definition

In alignment with Part II, we consider a polygonal reference domain $\hat{\Omega} \subset \mathbb{R}^2$ with a time-dependent domain $\Omega_t \subset \mathbb{R}^2$ for $t \in [0, T]$ where $T > 0$. We denote time-dependent ALE map by $\mathscr{A}_t : \hat{\Omega} \to \Omega_t$. The ALE velocity is denoted in

both the reference and physical domain by $\hat{\mathbf{w}}$ and $\mathbf{w}$ respectively. In this ALE formulation, we assumed that the ALE mapping is prescribed. In alignment with [27, 28], we consider an ALE mapping that is Lipschitz with a continuous Lipschitz inverse $\mathscr{A}_t^{-1} : \Omega_t \to \hat{\Omega}$. Under this condition, we have that the determinant of the Jacobian of the ALE mapping is uniformly bounded below by a time-independent constant j_0 such that

$$0 < j_0 < \det\left(\mathbf{J}_{\mathscr{A}_t}\right) \qquad \forall t \in [0, T].$$

The Advection-diffusion Problem

The PDE considered in Part III is a linear advection-diffusion problem given as: for $t \in (0, T]$, find $\rho(\mathbf{x}, t)$ such that

$$\begin{aligned} \frac{\partial \rho}{\partial t} - \mu\Delta\rho + \nabla\cdot(\mathbf{b}\rho) &= f && \mathbf{x} \in \Omega_t,\ t \in (0, T], \\ \rho &= \rho_0(\mathbf{x}) && \mathbf{x} \in \Omega_0,\ t = 0, \\ \rho &= 0 && \mathbf{x} \in \partial\Omega_t,\ t \in [0, T], \end{aligned} \tag{8.1}$$

where $\mathbf{b}$ is a convection velocity and $\mu \geq 0$ is a constant diffusivity parameter.

The conservative ALE formulation for this advection diffusion problem is well-documented [50]. We defined the space of functions with zero material derivative by

$$\mathcal{X}(\Omega_t) = \left\{ v : \Omega_t \to \mathbb{R} \ :\ v = \hat{v} \circ \mathscr{A}_t^{-1},\ \hat{v} \in H_0^1(\hat{\Omega}) \right\}. \tag{8.2}$$

Reynolds Transport Theorem 2.2.1 gives for some test function $v \in \mathcal{X}(\Omega_t)$ that

$$\frac{d}{dt}\int_{\Omega_t} \rho v \ d\mathbf{x} = \int_{\Omega_t} v \left\{ \frac{\partial \rho}{\partial t} + \nabla\cdot(\mathbf{w}\rho) \right\} \ d\mathbf{x}.$$

Substitution of the PDE (8.1) and integration by parts leads to

$$\begin{aligned} \frac{d}{dt}\int_{\Omega_t} \rho v \ d\mathbf{x} &= \int_{\Omega_t} v \left\{ f + \mu\Delta\rho + \nabla\cdot(\mathbf{w}\rho - \mathbf{b}\rho) \right\} \ d\mathbf{x} \\ &= \int_{\Omega_t} fv - \mu\nabla\rho\cdot\nabla v - (\mathbf{w} - \mathbf{b})\cdot\rho\nabla v \ d\mathbf{x}. \end{aligned}$$

The above equation can be written as follows

$$\frac{d}{dt}M_t(\rho, v) + \mu A_t(\rho, v) + B_t(\rho, v; \mathbf{w}) = l_t(v), \tag{8.3}$$

where,

$$M_t(\rho, v) = \int_{\Omega_t} \rho v \; d\mathbf{x}, \tag{8.4}$$

$$A_t(\rho, v) = \int_{\Omega_t} \nabla\rho \cdot \nabla v \; d\mathbf{x}, \tag{8.5}$$

$$B_t(\rho, v; \mathbf{w}) = \int_{\Omega_t} (\mathbf{w} - \mathbf{b}) \cdot \rho\nabla v \; d\mathbf{x}, \tag{8.6}$$

$$l_t(v) = \int_{\Omega_t} f v \; d\mathbf{x}. \tag{8.7}$$

The stability of this formulation is well known under the assumption that $\nabla \cdot \mathbf{b} \in L^\infty(\Omega_t)$ and $f \in H^{-1}(\Omega_t)$ for all $t \in [0, T]$, where $H^{-1}(\Omega_t)$ denotes the dual space of $H^1(\Omega)$ [54]. The stability of the continuous formulation is independent on the choice of ALE mapping. Similar estimates have been presented for a FEM discretisation of the ALE method in which the stability is dependent on the choice of ALE mapping [54, 55, 28]. In Part III we do not present any analysis on the VEM-ALE scheme, instead we assume that the choices of $\mathbf{b}$ and f satisfy the aforementioned regularity conditions and that this is sufficient to propose a numerically stable method.

At this point we remark that the continuous formulation given by Equation (8.3) is independent of the choice of a conservative or non-conservative ALE approach in the continuous framework [28]. As with FEMs, this is not the case for VEM discretisation of Equation (8.3) and the VEM formulation in the next section is only applicable to a conservative ALE formulation of the advection-diffusion problem.

8.2 A Moving Virtual Element Space

Following Chapter 5, we denote the reference mesh, constructed on $\hat{\Omega}$ and satisfying Assumption 1, by $\hat{\mathcal{T}}_h$. The classical VEM space of degree k on $\hat{\Omega}$ is

defined as

$$\hat{V}_h = \left\{ \hat{v}_h \in H^1(\hat{\Omega}) \ : \ \hat{v}_h|_{\hat{E}} \in V_k(\hat{E}) \quad \forall \hat{E} \in \hat{\mathcal{T}}_h \right\},$$

with $V_k(\hat{E})$ defined in Equation (2.3), and the corresponding restricted space with homogeneous boundary conditions defined by

$$\hat{V}_{h,0} = \left\{ v_h \in \hat{V}_h \cap H_0^1(\hat{\Omega}) \right\}.$$

The discrete ALE mapping $\mathscr{A}_{h,t} \in \left[\hat{V}_h\right]^2$ is defined for all time $t \in [0,T]$ using the VEM interpolation of the true ALE mapping. The corresponding ALE velocity field $\hat{\mathbf{w}}_{h,t} \in \left[\hat{V}_h\right]^2$ is given by the interpolation of $\hat{\mathbf{w}}_t$. As the ALE mapping and velocity are defined on the reference domain, these functions are only spatially dependent on $\boldsymbol{\xi}$ with a time-dependent set of DoFs. We define the discrete Jacobian matrix by

$$\mathbf{J}_{\mathscr{A}_{h,t}} = \Pi^1_{k-1} \mathscr{A}_{h,t} \qquad \forall t \in [0,T],$$

with the corresponding determinant denoted by $j_{h,t}$. The virtual domain generated by this VEM mapping is denoted by $\Omega_{h,t}$ and the corresponding virtual mesh is denoted by $\mathcal{T}_{h,t}$. We refer to Section 5.2 for additional details.

The discrete counterpart to Equation (8.2) is defined using the discrete ALE mapping

$$\mathcal{X}_h(\Omega_{h,t}) = \left\{ v : \Omega_t \to \mathbb{R} \ : \ v = \hat{v} \circ \mathscr{A}_{h,t}^{-1}, \ \hat{v} \in \hat{V}_h \right\}.$$

For the VEM we choose a basis to be a subset of $\mathcal{X}_h$

$$\{\varphi_i(\mathbf{x},t)\}_{i=1}^{N^{\text{dofs}}} \subset \mathcal{X}_h(\Omega_{h,t}),$$

where each basis function satisfies for all $t \in [0,T]$

$$\varphi_i(\mathbf{x},t) = \hat{\varphi}_i(\boldsymbol{\xi}) \circ \mathscr{A}_{h,t}^{-1}, \tag{8.8}$$

with $\hat{\varphi}_i$ being the canonical VEM basis function of $\hat{V}_h$ defined using Equation (2.5). Using this basis, we define a time dependent VEM function by

$$v_{h,t} = v_h(\mathbf{x},t) = \sum_{i=1}^{N^{\text{dofs}}} \text{dof}_i(v_h(\mathbf{x},t)) \varphi_i(\mathbf{x},t), \tag{8.9}$$

where φ_i is defined by Equation (8.8). We emphasis that the DoFs of the discrete function $v_{h,t}$ in Equation (8.8) are time-dependent so in general $v_{h,t} \notin \mathcal{X}_h(\Omega_{h,t})$, instead $v_{h,t}$ is a time-dependent linear combination of elements of $\mathcal{X}_h(\Omega_{h,t})$. The moving VEM space can then be defined, using Equations (8.8) and (8.9), as

$$V_{h,t} = \left\{ v_{h,t} \in H_0^1(\Omega_{h,t}) \ : \ v_{h,t} = \hat{v}_{h,t} \circ \mathscr{A}_{h,t}^{-1}, \ \hat{v}_{h,t} \in \hat{V}_h \right\}.$$

To impose the homogeneous Dirichlet boundary conditions, the restriction of these VEM spaces to zero boundary conditions are denoted by

$$V_{h,t,0} = \left\{ v_{h,t} \in V_{h,t} \ : \ \hat{v}_{h,t} = v_{h,t} \circ \mathscr{A}_{h,t} \in \hat{V}_{h,0} \right\},$$
$$\mathcal{X}_{h,0} = \left\{ v_h \in \mathcal{X}_h(\Omega_{h,t}) \ : \ \hat{v}_h = v_h \circ \mathscr{A}_{h,t} \in \hat{V}_{h,0} \right\}.$$

8.3 A Semi-discrete Scheme

In this VEM the numerical solution of the advection-diffusion equation is given by $\rho_{h,t} \in V_{h,t,0}$. This solution is only implicitly known in the ALE coordinates as the VEM-ALE scheme is computed using only the representation of $\rho_{h,t}$ and the test functions $v_{h,t} \in \mathcal{X}_h(\Omega_{h,t})$ in the reference coordinates.

For ease of reading, we drop the temporal subscript for the solution and test functions in the following formulations, instead the time-dependency will be described by the temporal subscript in the bilinear forms of the method.

The VEM-ALE semi-discrete formulation is given as follows: For a given $t \in (0, T]$ find $\rho_{h,t} \in V_{h,t,0}$ such that

$$\frac{d}{dt} M_{h,t}(\rho_h, v_h) + \mu A_{h,t}(\rho_h, v_h) + B_{h,t}(\rho_h, v_h; \mathbf{w}_h) = l_{h,t}(v_h) \qquad \forall v_h \in \mathcal{X}_{h,0}(\Omega_{h,t}).$$

The VEM discretisations of Equations (8.4), (8.5), (8.6) and (8.7) are given

by element-wise contributions on the reference mesh

$$M_{h,t}(\rho_h, v_h) = \sum_{\hat{E}\in\hat{\mathcal{T}}_h} M_{h,t}^{\hat{E}}(\rho_h, v_h), \qquad A_{h,t}(\rho_h, v_h) = \sum_{\hat{E}\in\hat{\mathcal{T}}_h} A_{h,t}^{\hat{E}}(\rho_h, v_h),$$
$$B_{h,t}(\rho_h, v_h; \mathbf{w}_h) = \sum_{\hat{E}\in\hat{\mathcal{T}}_h} B_{h,t}^{\hat{E}}(\rho_h, v_h; \mathbf{w}_h), \qquad l_{h,t}(v_h) = \sum_{\hat{E}\in\hat{\mathcal{T}}_h} l_{h,t}^{\hat{E}}(v_h).$$

The local element contributions are defined in a similar fashion to those given in Section 5.5 as

$$\begin{aligned}
M_{h,t}^{\hat{E}}(\rho_h, v_h) &= \int_{\hat{E}} \Pi_k^0\hat{\rho}_h \ \Pi_k^0\hat{v}_h \ j_{h,t} \ d\boldsymbol{\xi} + h_{\hat{E}}^2 S^{\hat{E}}(\hat{\rho}_h - \Pi_k^0\hat{\rho}_k, \hat{v}_h - \Pi_k^0\hat{v}_k),\\
A_{h,t}^{\hat{E}}(\rho_h, v_h) &= \int_{\hat{E}} \mathbf{J}_{\mathscr{A}_{h,t}}^{-\top}\Pi_{k-1}^1\hat{\rho}_h \ \mathbf{J}_{\mathscr{A}_{h,t}}^{-\top}\Pi_{k-1}^1\hat{v}_h \ j_{h,t} \ d\boldsymbol{\xi} + S^{\hat{E}}(\hat{\rho}_h - \Pi_k^\nabla\hat{\rho}_k, \hat{v}_h - \Pi_k^\nabla\hat{v}_k),\\
B_{h,t}^{\hat{E}}(\rho_h, v_h; \mathbf{w}_h) &= \int_{\hat{E}} \Pi_k^0(\hat{\mathbf{w}}_h - \hat{\mathbf{b}}_h)\cdot\Pi_k^0\rho_h\mathbf{J}_{\mathscr{A}_{h,t}}^{-\top}\Pi_{k-1}^1\hat{v}_h \ j_{h,t} \ d\boldsymbol{\xi},\\
l_{h,t}^{\hat{E}}(v_h) &= \int_{\hat{E}} \hat{f}_h \ \Pi_k^0\hat{v}_h \ j_{h,t} \ d\boldsymbol{\xi},
\end{aligned}$$

where $S^{\hat{E}}(\cdot,\cdot)$ is the standard dofi-dofi stabilisation term defined in Equation (2.8), $\hat{f}_h$ is a discrete approximation of the forcing data defined as $\hat{f}_h := \Pi_{k-2}^0(f(\mathbf{x},t)\circ\mathscr{A}_{h,t})$ for $k \geq 2$ and $\hat{f}_h := \Pi_0^0 f(\mathbf{x},t)$ for $k = 1$ and $\hat{\mathbf{b}}_h := \mathbf{b}\circ\mathscr{A}_{h,t}$

8.4 A fully Discrete Scheme

A method of lines approach is taken to perform the integration of the weak formulation with respect to time. The θ-scheme is well known [74, 35] for an ODE $dy/dt = f(\mathbf{x},t)$ over an interval $[t_n, t_{n+1}]$ as

$$\frac{dy}{dt}\Big|_{t_n} \approx \theta f(\mathbf{x}, t_{n+1}) + (1-\theta)f(\mathbf{x}, t_n) \qquad \theta \in [0,1].$$

We define the fully discrete VEM-ALE scheme using the θ-scheme. For $\theta \in [0,1]$ and $0 \leq t_n < t_{n+1} \leq T$ with $\Delta t_{n+1} = t_{n+1} - t_n$, the time derivative of

$M_{h,t}(\rho_h, v_h)$ is approximated via

$$\begin{aligned}\frac{M_{h,n+1}(\rho_h, v_h) - M_{h,n}(\rho_h, v_h)}{\Delta t} &= \theta l_{h,n+1}(\rho_h, v_h) + (1-\theta) l_{h,n}(\rho_h, v_h) \\ &\quad - \mu \left(\theta A_{h,n+1}(\rho_h, v_h) + (1-\theta) A_{h,n}(\rho_h, v_h)\right) \\ &\quad - \left(\theta B_{h,n+1}(\rho_h, v_h; \mathbf{w}_h) + (1-\theta) B_{h,n}(\rho_h, v_h; \mathbf{w}_h)\right).\end{aligned} \tag{8.10}$$

By rearranging the terms of Equation (8.10) we can define an algebraic system of equations. We define the matrices $\boldsymbol{\mathcal{G}}_{n+1}$ and $\boldsymbol{\mathcal{H}}_n$ by

$$\begin{aligned}(\mathcal{G}_{n+1})_{i,j} &= M_{h,n+1}(\varphi_i, \varphi_j) + \Delta t\theta[A_{h,n+1}(\varphi_i, \varphi_j) + B_{h,n+1}(\varphi_i, \varphi_j; \mathbf{w}_h)] \\ (\mathcal{H}_n)_{i,j} &= M_{h,n}(\varphi_i, \varphi_j) + \Delta t(\theta - 1)[A_{h,n}(\varphi_i, \varphi_j) + B_{h,n}(\varphi_i, \varphi_j; \mathbf{w}_h)],\end{aligned}$$

and the vector $\boldsymbol{\mathcal{F}}_n$ is defined as

$$(\mathcal{F}_n)_i = \Delta t\theta l_{h,n+1}(\varphi_i) + \Delta t(1-\theta) l_{h,n}(\varphi_i).$$

The fully discrete VEM-ALE scheme is the defined as follows: given $\boldsymbol{\rho}_n$, $\mathbf{w}_{n+1}$ and $\mathbf{w}_n$, find $\boldsymbol{\rho}_{n+1}$ such that

$$\boldsymbol{\mathcal{G}}_{n+1}\boldsymbol{\rho}_{n+1} = \boldsymbol{\mathcal{H}}_n\boldsymbol{\rho}_n + \boldsymbol{\mathcal{F}}_n.$$

To conclude this chapter, we note that if the ALE velocity field $\mathbf{w}_h$ is not computable, but the ALE mapping is known, we can approximate this via

$$\mathbf{w}_{h,n} = \frac{\mathscr{A}_{h,n+1} - \mathscr{A}_{h,n}}{\Delta t_{n+1}}. \tag{8.11}$$

If the boundary transformation is known, the ALE mapping can be chosen such that it is indeed given as a piecewise linear displacement in time meaning Equation (8.11) introduces no additional discretisation error.

Chapter 9

Numerical Investigation

In the final chapter of this thesis, the VEM-ALE scheme is benchmarked against a series of advection-diffusion problems on moving meshes within DUNE, directly utilising the isoparamteric VEM software developed in Part II. Third and Fourth order convergence results are obtained in all test cases in the L^2 norm for $k = 2$ and $k = 3$ respectively. The formulation is then extended to attain higher orders of accuracy for the moving mesh method of Part I.

A sequence of CVT-type meshes is utilised in all numerical experiments of this chapter, an example of a CVT-type mesh was given in Figure 2.1. The VEM-ALE is implemented, as in Chapter 7, within the DUNE environment [90, 48]. Unless stated otherwise, the Crank-Nicolson method ($\theta = 0.5$) is used to approximate integration between discrete time levels. This choice provides unconditional stability on the numerical solution and second-order accuracy in time [35]. The time step size Δt is reduced in all advection-diffusion simulations according to $\Delta t^2 \sim h^{k+1}$ such that the expected orders of convergence in the spatial discretisation are produced. All numerical experiments in this chapter are run to a final time of $T = 0.01$. The H^1 and L^2 errors are computed at the final time T using the error norms used for Method I in Equations (7.4) and (7.5).

To ensure that the accuracy of the VEM-ALE scheme can be assessed, prob-

lems with inhomogenous boundary conditions have to be considered. These conditions are enforced by applying a Dirichlet boundary condition on the method at each time step using the interpolant of the DoFs of the true solution on the moving boundary. Similar to the experiments of Chapter 7, the DUNE-UFL library is used to symbolically compute the forcing data and the explicit expressions of these terms are omitted in this chapter.

In Section 9.1, the VEM is verified by considering an advection-diffusion problem on a pair of time-independent ALE mappings. A pure diffusion problem is tested in Section 9.2. A general advection-diffusion problem on a time-dependent domain is considered in Section 9.3. In Section 9.4, the chapter is concluded with a high-order discretisation of the velocity-based moving mesh VEM of Part I.

9.1 Time-independent ALE Maps

In the first test we validate the numerical method by considering a time-independent domain transformation. The CE and Warped Square mappings are reused. These are given by Equations (7.1) and (7.2) of Section 7.1 as

$$\mathscr{A}(\boldsymbol{\xi}) = [\xi + \xi\eta(1-\xi)/2,\ \eta + \xi\eta(1-\eta)/2]\,,$$
$$\mathscr{A}(\boldsymbol{\xi}) = \left[\sin\frac{\xi\pi}{3},\ e^{\eta}\right].$$

We choose the diffusivity parameter to be $\mu = 1$ and the convective velocity to be $\mathbf{b} = \mathbf{x}$. The forcing data f and Dirichlet boundary conditions are chosen such that the true solution of the advection-diffusion problem in the physical domain is given as

$$\rho(x, y, t) = \exp(-\pi^2 t)\sin(\pi x)\sin(\pi y).$$

Numerical results for both mappings are given in Figure 9.1. Here we observe the expected convergence rates in the H^1 and L^2 norms of $O(h^k)$ and $O(h^{k+1})$ respectively when run to a final time of $T = 0.01$.

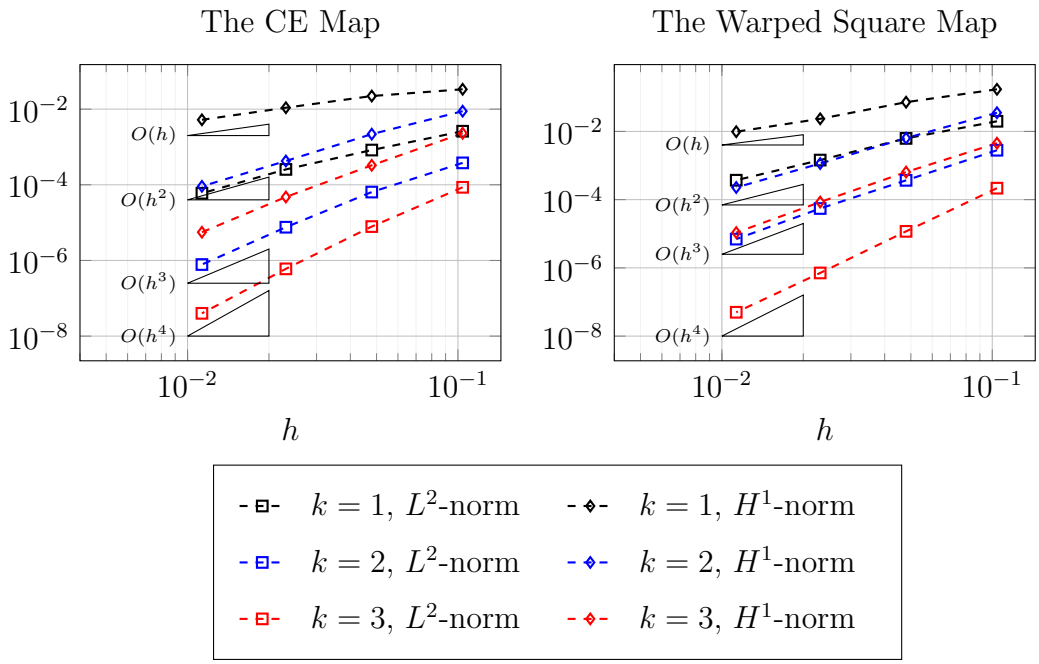

Figure 9.1: Error plots for the advection-diffusion equation on the time-independent CE ALE mapping (left) and the Warped Square ALE mapping (right) for $k = 1, 2, 3$.

9.2 A Pure Diffusion Problem

For this experiment a heat equation problem is considered in which there is a zero convection term $\mathbf{b} = 0$, leading to a linear diffusion problem on a moving domain. We set $\mu = 1$ and $f = 0$ and choose an initial conditions such that the solution is given by

$$\rho(x, y, t) = \exp(-2\pi^2 t)\sin(\pi x)\sin(\pi y).$$

The domain is transformed by the linearised CE mapping from Equation (7.1)

$$\mathscr{A}_t(\boldsymbol{\xi}) = \boldsymbol{\xi} + \frac{t}{2T}\xi\eta\left[1-\xi, 1-\eta\right],$$

where $T = 0.01$ is the final time in the simulation. The vorticial motion (VM) map [78] is defined as the solution of the system of ODEs:

$$\dot{x} = 2\sin(\pi x)\cos(\pi y), \tag{9.1}$$

$$\dot{y} = -2\cos(\pi x)\sin(\pi y). \tag{9.2}$$

In the case of the VM mapping defined by Equations (9.1) and (9.2), only the velocity-field of the ALE mapping is provided and the non-linear nature of

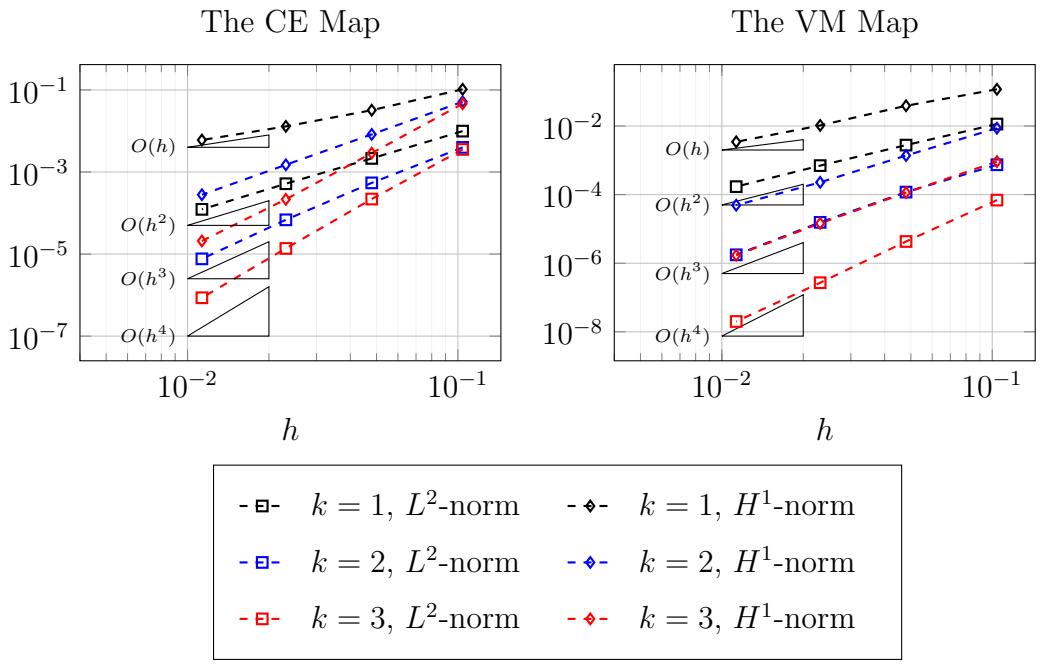

Figure 9.2: Error plots for the pure diffusion problem on the linearised CE ALE mapping (left) and the VM ALE mapping (right) for $k = 1, 2, 3$.

the velocity field requires an explicit time integration scheme. To match the second-order accuracy of the Crank-Nicolson time-stepping scheme, we employ Heun's method (modified Euler [35]) to integrate Equations (9.1) and (9.2) over time. The numerical results are presented in Figure 9.2 for both the CE and VM ALE mappings, again we observe $O(h^{k+1})$ and $O(h^k)$ orders of convergence in the L^2 and H^1 norms for both mappings.

9.3 A General Advection-diffusion Problem

In this final experiment we consider the numerical example of a solution to the advection-diffusion equation with a travelling feature. The domain is transformed from $\hat{\Omega} = [0, 1]^2$ to a rectangular domain $[0, 2] \times [0, 1]$ which oscillates in the y direction. This transformation is defined as

$$\mathscr{A}_t(\boldsymbol{\xi}) = \boldsymbol{\xi} + [\xi,\ A\sin(\pi u_y t)],$$

where $A > 0$ and u_y are user specified parameters which control the amplitude and frequency of the oscillations respectively. The ALE velocity field of this

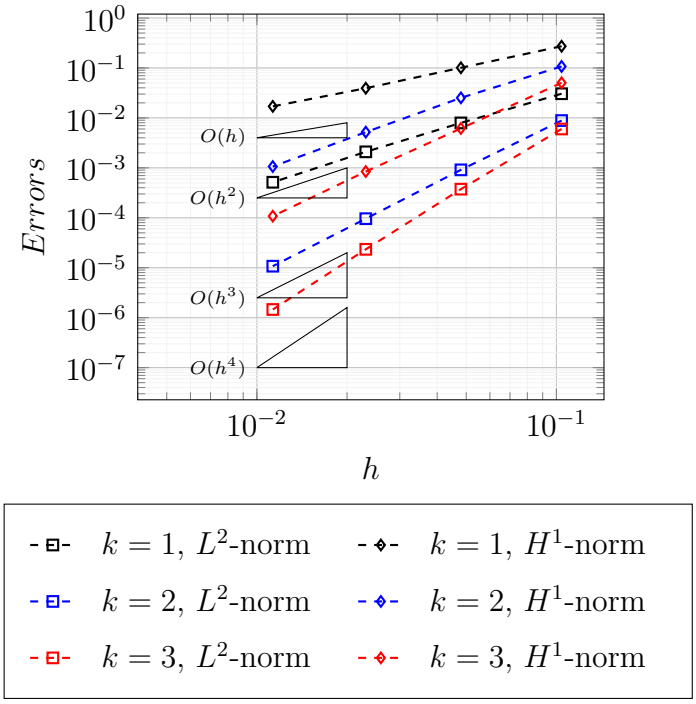

Figure 9.3: Error plot for the general advection-diffusion problem using the oscillating ALE map for $k = 1, 2, 3$.

transformation is given as

$$\mathbf{w} = [0, Au_y\pi\cos(\pi u_y t)].$$

The parameters of the advection-diffusion Equation (8.1) are set as $\mu = 1$, $\mathbf{b} = \mathbf{x}$ and f is chosen such that the true solution of the PDE is

$$\rho(\mathbf{x}, t) = \exp(-\pi^2 t)\sin(\pi(y - Asin(\pi u_y t)))\sin(\pi(x - u_x t)),$$

where u_x is another user specified parameter that controls the speed of travel of the solution in the positive x direction. In our experiment we consider the ALE mapping given by $A = 1/10$ and $u_x = u_y = 20$ and run the simulation to a final time of $T = 0.01$. From Figure 9.3, we observe the expected orders of convergence in the L^2 and H^1 norms. Moving mesh and solution profile snapshots are provided in Figures 9.4 and 9.5 respectively.

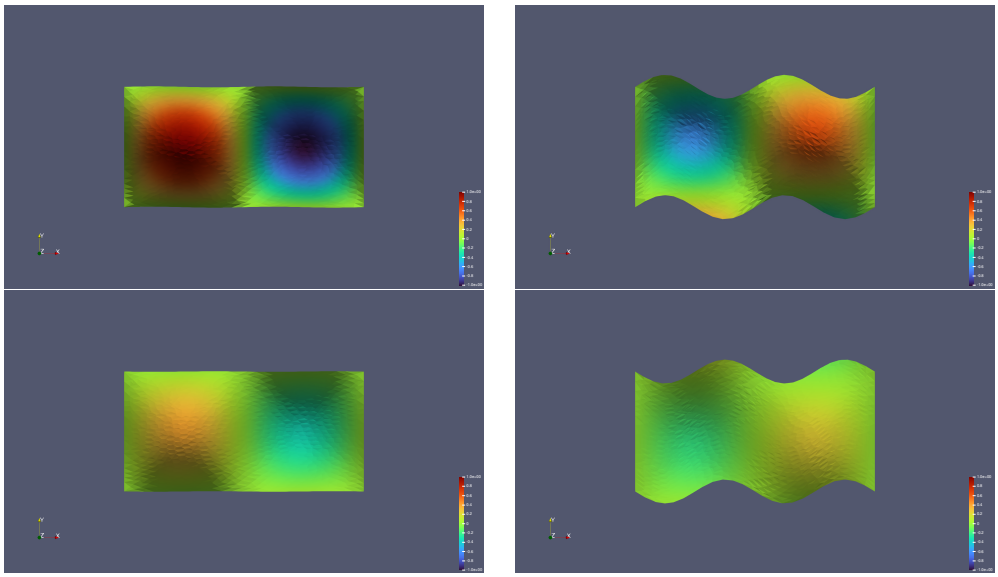

Figure 9.4: Solution snapshots of the advection-diffusion equation using a quadratic VEM and a reference mesh of 800 elements. The snapshots are taken at times $t = 0$ (top left), $t = 0.025$ (top right), $t = 0.05$ (bottom left) and $t = 0.075$ (bottom right).

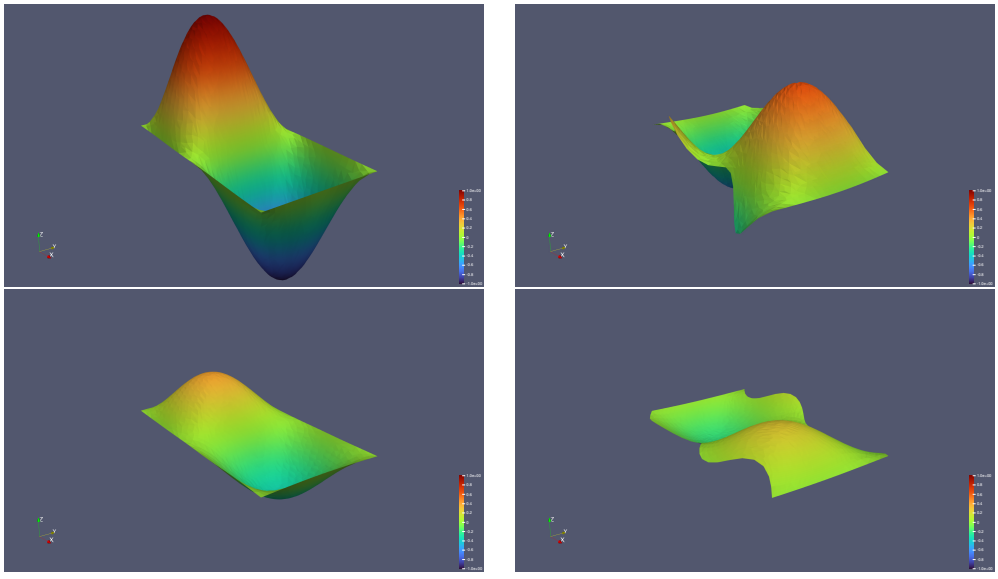

Figure 9.5: Solution snapshots of the advection-diffusion equation using a quadratic VEM and a reference mesh of 800 elements. The snapshots are taken at times $t = 0$ (top left), $t = 0.025$ (top right), $t = 0.05$ (bottom left) and $t = 0.075$ (bottom right).

9.4 The Velocity-based Moving Mesh Method

To conclude this chapter, we revisit the moving mesh VEM proposed in Part I and consider the formulation for the PME presented in in Chapter 3. By applying the isoparametric VEM discretisation of Method I, the velocity field and the ALE update can be computed to a higher-degree of precision than before.

We consider the $m = 1$ case of the PME

$$\frac{\partial \rho}{\partial t} = \nabla \cdot (\rho \nabla \rho),$$

and measure the numerical error, as done in Section 4.1, against the similarity solution of the PME defined by Equation (4.1). In the interest of brevity, we only present the local bilinear form equations for the velocity-based moving mesh method. The construction of the global system of equations and the structure of Algorithm 1 remains unchanged. The same dofi-dofi stabilisation terms are also used (see Equation (2.8)) computed in this case using the projection operators defined on the reference domain.

The Velocity Problem

Consistent with the approach in Section 3.4, we discretise the velocity potential problem as: find $\hat{\phi}_{h,n} \in V_{h,n}$ such that

$$A_{h,n}(\phi_h, v_h) = d_{h,n}(v_h) \qquad \forall v_{h,n} \in V_{h,n},$$

where the local element contributions From Equations (3.24) and (3.25) are now given by

$$\begin{aligned} A^{\hat{E}}_{h,n}(\phi_h, v_h) &= \int_{\hat{E}} \Pi^0_k \hat{\rho}_h \ \mathbf{J}^{-\top}_{\mathscr{A}_{h,n}} \Pi^1_{k-1} \hat{\phi}_h \cdot \mathbf{J}^{-\top}_{\mathscr{A}_{h,n}} \Pi^1_{k-1} \hat{v}_h \ j_{h,n} \ d\boldsymbol{\xi} \\ &\quad + \bar{\rho}_h S^{\hat{E}}(\hat{\phi}_h - \Pi^\nabla_k \hat{\phi}_h, \hat{v}_h - \Pi^\nabla_k \hat{v}_h), \\ d^{\hat{E}}_{h,n}(v_h) &= -\int_{\hat{E}} \Pi^0_k \hat{\rho}_h \ \mathbf{J}^{-\top}_{\mathscr{A}_{h,n}} \Pi^1_{k-1} \hat{\rho}_h \cdot \mathbf{J}^{-\top}_{\mathscr{A}_{h,n}} \Pi^1_{k-1} \hat{v}_h \ j_{h,n} \ d\boldsymbol{\xi}, \end{aligned}$$

and we impose that $\text{dof}_1(\hat{\phi}_h) = 0$. Likewise the velocity reconstruction problem is discretised as: find $\mathbf{u}_h \in [V_h]^2$ such that

$$M_{h,n}(\mathbf{u}_h, v_h) = b_{h,n}(v_h) \qquad \forall v_{h,n} \in V_{h,n},$$

where $M_{h,n}(\cdot,\cdot)$ and $b_{h,n}(\cdot)$ are modified from Equations (3.27) and (3.28) to local contributions of

$$\begin{aligned} M_{h,n}^{\hat{E}}(\mathbf{u}_h, v_h) &= \int_{\hat{E}} \Pi_k^0 \hat{\mathbf{u}}_h \; \Pi_k^0 \hat{v}_h \; j_{h,n} \; d\boldsymbol{\xi} \\ &\quad + h_{\hat{E}}^2 S^{\hat{E}}(\hat{\mathbf{u}}_h - \Pi_k^0 \hat{\mathbf{u}}_h, \hat{v}_h - \Pi_k^0 \hat{v}_h), \\ b_{h,n}^{\hat{E}}(v_h) &= \int_{\hat{E}} \mathbf{J}_{\mathscr{A}_{h,n}}^{-\top} \Pi_{k-1}^1 \hat{\phi}_h \; \Pi_k^0 \hat{v}_h \; j_{h,n} \; d\boldsymbol{\xi}. \end{aligned}$$

The ALE Problem

In this experiment we only consider the case of $\mathbf{w}_h = \mathbf{u}_h$. The distribution of the initial mass monitor defined in Equation (3.30) is now computed locally via

$$\mu_{h,0}(v_h) = \sum_{\hat{E} \in \hat{\mathcal{T}}_h} \int_{\hat{E}} \Pi_k^0 \hat{\rho}_h \; \Pi_k^0 \hat{v}_h \; j_{h,0} \; d\boldsymbol{\xi}.$$

The ALE update Equation (3.31) is redefined locally by

$$\dot{\mu}_{h,n}^{\hat{E}}(v_h) = -\int_{\hat{E}} \left\{ \mathbf{J}_{\mathscr{A}_{h,n}}^{-\top} \Pi_{k-1}^1 \hat{\rho}_h - \Pi_k^0 \hat{\mathbf{w}}_h \right\} \cdot \Pi_k^0 \hat{\rho}_h \; \mathbf{J}_{\mathscr{A}_{h,n}}^{-\top} \Pi_{k-1}^1 \hat{v}_h \; j_{h,n} \; d\boldsymbol{\xi}, \qquad \forall v_{h,n} \in V_{h,n}.$$

Integration of the above equation is performed as done in Section 3.5, using a forward Euler time stepping scheme. The solution is reconstructed at a new time level by the following problem: find $\rho_{h,n} \in V_{h,n}$ such that

$$m_{h,n}(\rho_h, v_h) = \mu_{h,n}(v_h) \qquad \forall v_h \in V_{h,n},$$

where the local element contributions of $m_{h,n}(\rho_h, v_h)$ are given by

$$m_{h,n}^{\hat{E}}(\rho_h, v_h) = \int_{\hat{E}} \Pi_k^0 \hat{\rho}_h \; \Pi_k^0 \hat{v}_h \; j_{h,n} \; d\boldsymbol{\xi} + h_{\hat{E}}^2 S^{\hat{E}}(\hat{\rho}_h - \Pi_k^0 \hat{\rho}_h, \hat{v}_h - \Pi_k^0 \hat{v}_h).$$

Approximating Ω_0

Since the initial domain is a circle of radius $r_0 = 1/2$, we take the reference domain to be the polygonal interpolation of Ω_0. In the linear VEM this was a

suitable choice as the interpolant was a second order accurate approximation of Ω_0. From Assumption 4, we require that

$$|\mathscr{A}_{h,0} - \mathscr{A}_0|_{1,\hat{\Omega}} = O(h^k),$$

which is achieved by modifying the boundary DoFs (point values in two-dimensional VEM) of the polygonal approximation to interpolate $\partial\Omega_0$. An alternative would be to apply the harmonic extension operator defined by Equations (2.15) and (2.16). This has also been implemented and we discuss the impacts of choosing the approximation of Ω_0 at the end of this section.

Time-stepping

As the Lagrangian velocity field is required before the ALE update can be performed this experiment is restricted to using the Forward Euler method only ($\theta = 0$). In our presentation we use the original Forward Euler approach and scale the time-step sizes appropriately such that the numerical method is stable and produces the expected orders of accuracy. The VEM-ALE mapping can be written as

$$\mathscr{A}_{h,n+1} = \mathscr{A}_{h,0} + \sum_{i=0}^{n} (t_{i+1} - t_i)\hat{\mathbf{w}}_{h,i}.$$

In the FEM implementation, Heun's method has been successfully implemented to improve the efficiency of the method [67].

Results

Numerical experiments were conducted for the moving mesh method with $k = 2, 3$ ran for $T = 0.01$. At first, we tested the moving mesh method using the interpolation of the initial domain. This resulted in poor convergence rates limited to $O(h^{k-1/2})$ in the H^1 norm and $O(h^{k+1/2})$ in the L^2 norm. By using the harmonic extension of Equations (2.15) and (2.16) to construct $\Omega_{h,0}$ we observe improved rates of convergence. These results are presented in Figure 9.6. For the quadratic case we observe the expected convergence rates of $O(h^2)$

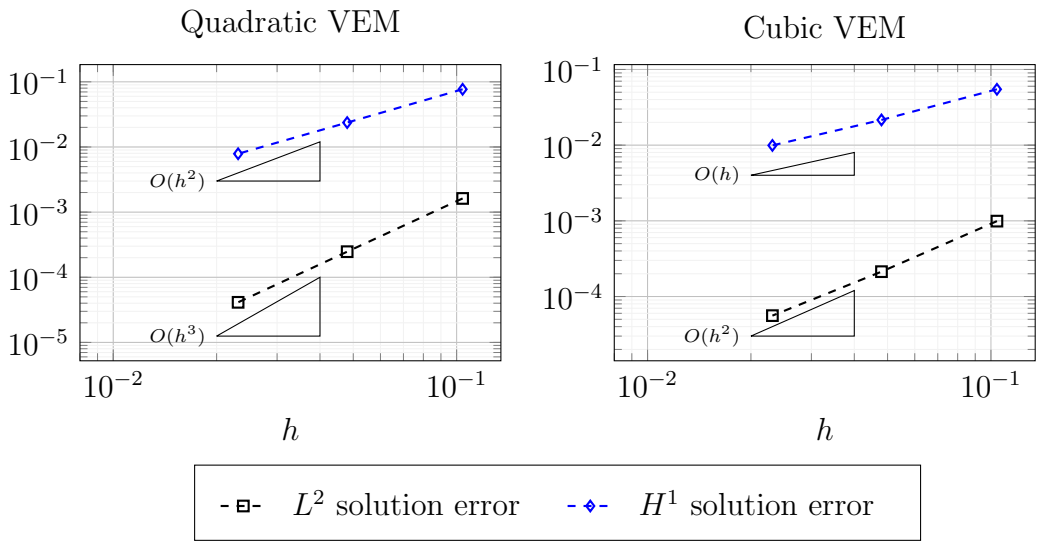

Figure 9.6: Error plots for the velocity-based moving mesh VEM of degree $k = 2$ (left) and $k = 3$ (right).

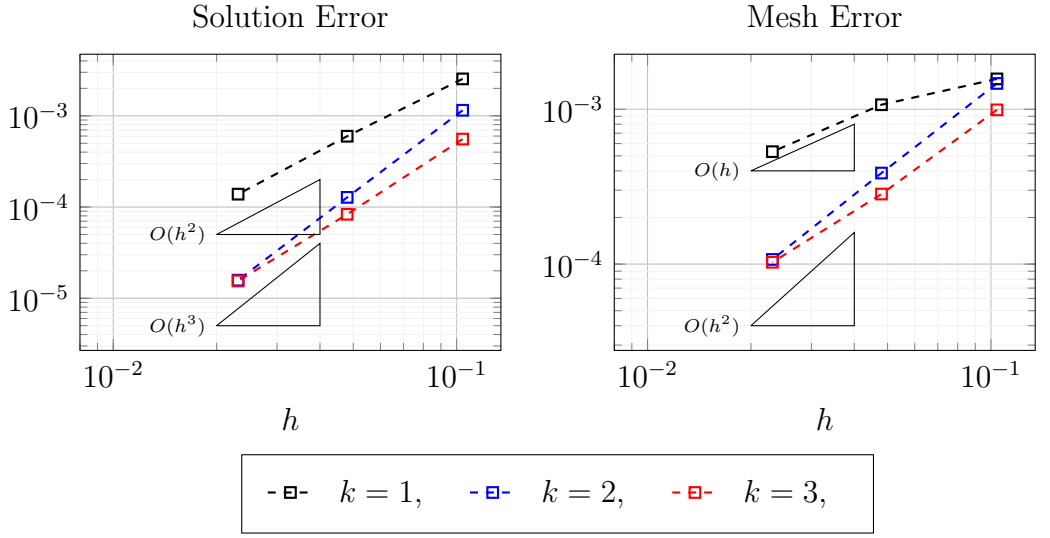

Figure 9.7: Discrete l^1 solution and mesh errors for the velocity-based moving mesh method of degree $k = 1, 2, 3$.

and $O(h^3)$ in the H^1 and L^2 error norms respectively. In the cubic case we observe a reduction in convergence rates from the quadratic VEM to $O(h)$ and $O(h^2)$ in the H^1 and L^2 norms respectively. This observation is further supported in Figure 9.7 with a comparison of the discrete l^1 errors compared to the linear moving mesh VEM presented in Part I.

To offer a possible explanation of this behaviour, we considered taking $\Omega_{h,0} = \hat{\Omega}$ and weakly imposing the boundary conditions on the solution. In this case

the initial ALE map is the identity map, and consequently $\mathscr{A}_{h,0} \in \left[C^\infty(\hat{\Omega})\right]^2$. As the initial condition is a quadratic polynomial, we expect from the polynomial consistency property of the VEM that the only source of numerical error in this simulation comes from integration with respect to time and the moving boundary being restricted to $O(h^2)$ accuracy. From figure 9.8 we observe that the order of convergence for the numerical solution is $O(h^{k+1})$ in both the H^1 and L^2 norms. In figure 9.9 we also observe against the original linear moving mesh VEM of Part I that this new scheme provides higher order accuracy in the discrete l^1 norm for $k = 2, 3$. The mesh error convergence rates do improve to second-order for $k = 2$ but does not achieve a higher order of accuracy for $k = 3$, with reduced mesh error but not an additional degree of accuracy.

We hypothesise from these results that the choice of the initial ALE map is important in obtaining expected convergence rates. This is possibly due to a regularity condition on the $\mathscr{A}_{h,0}$ that a VEM interpolation of a circle does not satisfy. As discussed at the conclusion of Part II, we have observed that non-smooth domain transformations lead to a reduction in the orders of accuracy of the isoparametric VEM. One remedy could be to employ a higher regularity VEM space for $\mathscr{A}_{h,0}$ such as a C^1 conforming VEM discretisation [48]. This remains an open problem that will be the subject of future works.

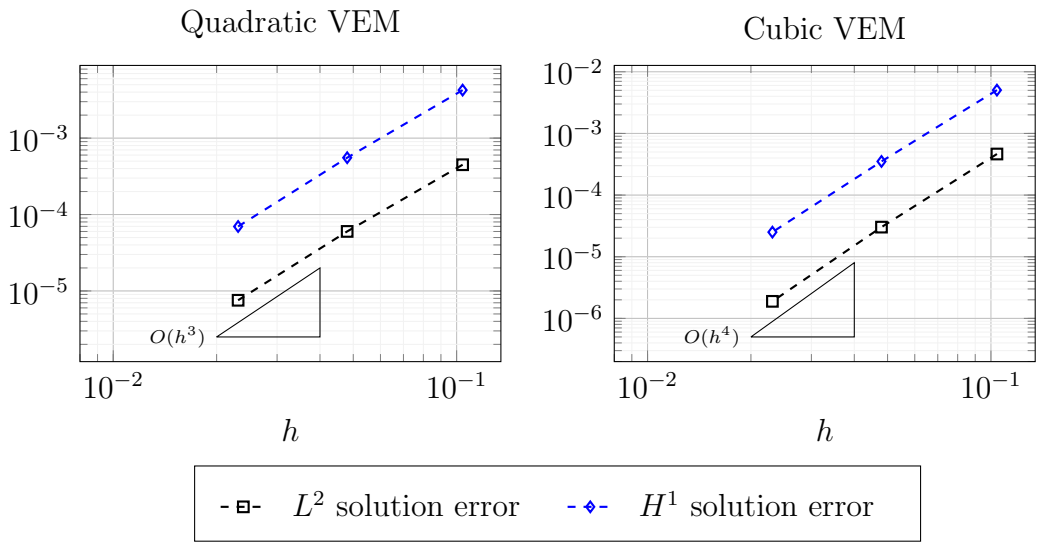

Figure 9.8: Error plots for the velocity-based moving mesh VEM of degree $k = 2$ (left) and $k = 3$ (right).

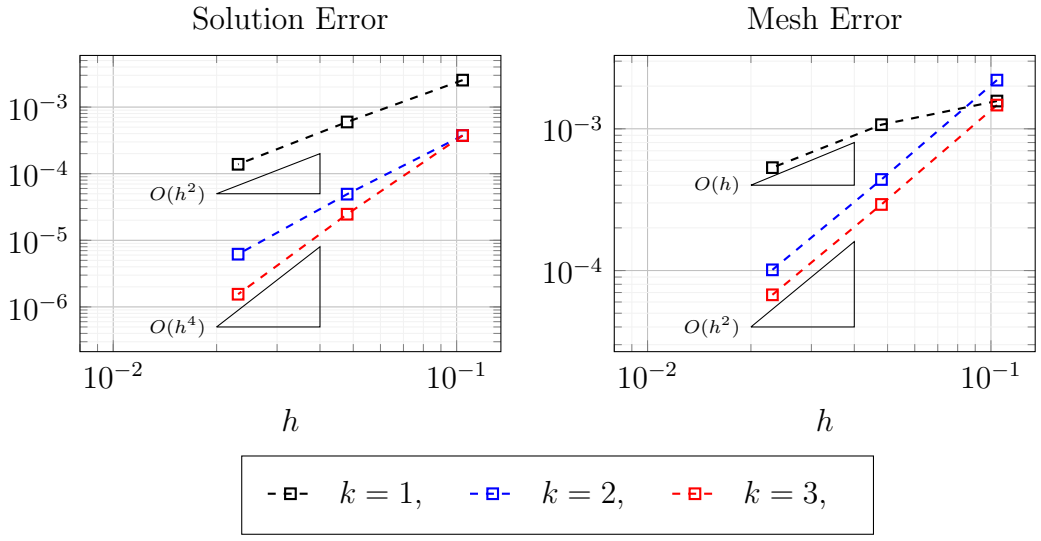

Figure 9.9: Discrete l^1 solution and mesh errors for the velocity-based moving mesh method of degree $k = 1, 2, 3$.

Discussion

In Part III a VEM discretisation is formulated and analysed for a conservative ALE formulation of an advection-diffusion problem. This VEM-ALE scheme is then successfully tested against a series of problems in which high orders of convergence of $O(h^k)$ and $O(h^{k+1})$ are observed in the H^1 and L^2 norms respectively. The framework is then extended to the velocity-based moving mesh method proposed in Part I which provides improved rates of convergence, leading to a substantial improvement in the accuracy of the moving mesh method.

To support the results of Part III, semi-discrete *a priori* stability and error estimate need to be developed. The framework for analysing FEM-ALE schemes from [54, 84, 60] can be extended to the VEM-ALE scheme using the isoparametric VEM analysis of Part II. A study is also required to verify the numerical time-stepping schemes and whether in the virtual element context these still possess satisfactory conservation properties, such as the Geometric Conservation Law, that finite element approaches can provide [54, 27].

It is also worth noting that only the first isoparametric VEM (Method I) of Chapter 5 was considered in the formulation and experiments. Method II could also be formulated by changing the semi-discrete equations of Section 8.3 accordingly. Similarly, a non-conservative ALE formulation could also be presented and discretised using both isoparametric methods. The implementation of these is ongoing work.

Within these experiments, there was an attempt to reproduce the results of the conservative remapping scheme [78] reviewed in Section 2.1. The VEM-

ALE did not produce a converging numerical approximation to this problem. In certain circumstance, such as when the ALE mapping was polynomial, the numerical scheme converged with $O(h^{k+1})$ accuracy in the L^2 norm. The hypothesis for this issue is that the psuedo-time PDE is effectively a transport equation, thus a conforming VEM would not be expected to produce numerically stable results. Instead a non-conforming Petrov-Galerkin type VEM would be more appropriate [25], this remains an open problem.

Having successfully implemented a high-order VEM-ALE scheme, the scope for expansion is very promising. The advection-diffusion equation has typically been used as a preliminary problem to benchmark and analyse before a numerical method is derived for the Navier-Stokes equations on moving domains. To develop a moving mesh method for the Navier-Stokes equations, a H(div) VEM, such as the one found in [45], would be required. In fact this has very recently been implemented for simulating Navier-Stokes on a fixed domain with a divergence-free VEM where the VEM produced better resolution of flow features compared to a standard Taylor-Hood finite element discretisation [48].

Chapter 10

Conclusion

10.1 Introduction

The objective of this project was stated as: To propose, analyse, and implement robust and effective moving mesh Virtual Element Methods.

This was pursued by proposing and implementing several moving mesh Virtual Element Method schemes for non-linear diffusion and advection-diffusion problems on time-dependent domains. Isoparametric Virtual Element Methods were proposed and analysed to support the development of higher-order accurate moving mesh methods using the Virtual Element Method.

The results presented in this thesis demonstrate that Virtual Element Methods can be successfully applied to existing moving mesh algorithms and achieve the same order of accuracy compared to classical Finite Element Method approaches.

The important contributions of this thesis include: the first moving mesh Virtual Element Method that solely utilises the Virtual Element Method in the numerical computations, an isoparametric Virtual Element Method that allows for the approximation of partial differential equations when only degrees of freedom of the domain transformation are known, and the first high-order Arbitrary Lagrangian-Eulerian Virtual Element Method for partial differential

equations on time-dependent domains with a moving boundary.

10.2 Summary of Part I

In Part I, a velocity-based moving mesh Finite Element Method is extended to a polygonal discretisation using the lowest order Virtual Element Method. The new method is benchmarked against non-linear diffusion problems by modeling the support of the solution on a moving boundary. Numerical results demonstrate that the Virtual Element Method achieves similar orders of accuracy compared to the Finite Element Method. The additional flexibility of the Virtual Element Method is then exploited to propose and implement a more complex algorithm to simulate contact between a moving boundary and fixed obstacles. Numerical simulations suggest that the Virtual Element Method remains robust in the presence of degenerating mesh edges and coalescing mesh vertices.

10.3 Summary of Part II

In Part II, an isoparametric Virtual Element Method is proposed, analysed, and benchmarked for second-order elliptic partial differential equations on transformed domains. Two discretisation schemes are proposed to provide a computable and accurate isoparametric method. In both methods, H^1 estimates are proven, suggesting that the orders of accuracy are limited to the minimum discretisation degree of the domain transformation and the partial differential equation solution. Numerical experiments validate the H^1 estimates and indicate that an additional degree of accuracy is obtained in the L^2 error norm as well.

10.4 Summary of Part III

In Part III, the isoparametric Virtual Element Method from Part II is employed to propose and implement a conservative Arbitrary Lagrangian-Eulerian Virtual Element Method for advection-diffusion problems on time-dependent domains with prescribed boundary motion. Numerical experiments achieve high orders of convergence in the H^1 and L^2 error norms for a series of advection-diffusion problems. The formulation is then extended to the moving mesh method from Part I, resulting in improved orders of accuracy over the lowest order Virtual Element Method.

10.5 Discussion, Interpretation, and Contribution to Knowledge

At the time of writing, this thesis provides the first successful implementations of the Virtual Element Method for moving mesh algorithms in the literature. This demonstrates that moving mesh methods can be extended to incorporate more general mesh structures, including polygons and curved-edged elements. This greatly improves the flexibility of moving mesh methods and allows for the future development of optimal mesh movement strategies that exploit this flexibility. In the area of Virtual Element Methods, this thesis contributes to the rapidly growing list of problems to which the Virtual Element Method can be successfully and efficiently applied, setting itself apart from other directions in this research community. The successful application of the Virtual Element Method in this context serves as a motivation for its continued development for applications in mathematics and engineering.

10.6 Recommendations for Future Research

The scope for future research is very broad, falling into two categories.

The first direction is to further develop the mathematical analysis to support the numerical methods presented in this thesis. The analysis of the moving mesh algorithm from Part I remains an open problem in both the finite element and virtual element implementations. Additional analysis is required to support the isoparametric Virtual Element Method, specifically in deriving L^2 estimates and studying the use of domain transformations with limited regularity. The stability and error estimates necessary to support the numerical results presented in the Arbitrary Lagrangian-Eulerian scheme from Part III are currently being investigated.

The second direction is to implement Virtual Element Methods for more complex problems. The results of Part III suggest that the next step would be to develop an Arbitrary Lagrangian-Eulerian Virtual Element Method for the Navier-Stokes equations. This would involve utilising $H(\text{div})$ Virtual Element Methods and further examining the choice of mesh transformations and time-stepping schemes. If successful, the method can be further developed for fluid-structure interaction problems, which commonly employ Arbitrary Lagrangian-Eulerian schemes. Additionally, the isoparametric Virtual Element Method from Part II can be extended to simulate surface partial differential equations and moving surface problems, which are ongoing popular areas of research as discussed in Section 2.3. The numerical method from Part II may provide higher-order methods for some of the problems in this research area.

10.7 Conclusion

In conclusion, this thesis has made contributions to the field of moving mesh methods by proposing and analysing robust and effective moving mesh Virtual Element Method schemes. The results demonstrate the applicability and accuracy of Virtual Element Methods in existing moving mesh algorithms, achieving similar orders of accuracy compared to classical Finite Element Method

approaches.

The contributions of this research include the development of novel moving mesh Virtual Element Method formulations, such as the first purely Virtual Element Method-based moving mesh scheme and the isoparametric Virtual Element Method that allows for the approximation of partial differential equations using only known degrees of freedom of the domain transformation. Additionally, the introduction of a high-order Arbitrary Lagrangian-Eulerian Virtual Element Method for problems on time-dependent domains with moving boundaries provides a stepping stone to develop more complex methods to simulate problems in computational fluid dynamics.

The significance of this work lies in its ability to enhance the flexibility and applicability of moving mesh algorithms by employing polygonal and curved edge polygonal meshes. The findings and methodologies presented contribute to the existing knowledge by providing alternatives to classical Finite Element Method approaches.

The future research directions identified in this thesis offer exciting opportunities to further advance the mathematical analysis and explore the application of Virtual Element Methods in more complex problems. By pursuing these avenues, researchers can continue to expand upon the techniques developed in this thesis. If these future developments prove fruitful, then moving mesh Virtual Element Methods will become viable for industrial applications and commercial software development.

Bibliography

[1] R. A. Adams and J. J. Fournier. *Sobolev Spaces*. Elsevier, 2003.

[2] B. Ahmad, A. Alsaedi, F. Brezzi, L. Marini, and A. Russo. Equivalent projectors for virtual element methods. *Computers & Mathematics with Applications*, 66(3):376–391, 9 2013.

[3] F. Aldakheel, B. Hudobivnik, E. Artioli, L. Beirão da Veiga, and P. Wriggers. Curvilinear virtual elements for contact mechanics. *Computer Methods in Applied Mechanics and Engineering*, 372:113394, 12 2020.

[4] M. S. Alnæs, A. Logg, K. B. Ølgaard, M. E. Rognes, and G. N. Wells. Unified form language. *ACM Transactions on Mathematical Software*, 40(2):1–37, 2 2014.

[5] P. F. Antonietti, L. Beirão Da Veiga, S. Scacchi, and M. Verani. A C1 virtual element method for the Cahn-Hilliard equation with polygonal meshes. *SIAM Journal on Numerical Analysis*, 54(1):34–56, 1 2016.

[6] P. F. Antonietti, G. Manzini, and M. Verani. The conforming virtual element method for polyharmonic problems. *Computers & Mathematics with Applications*, 79(7):2021–2034, 4 2020.

[7] P. F. Antonietti, M. Verani, C. Vergara, and S. Zonca. Numerical solution of fluid-structure interaction problems by means of a high order Discontinuous Galerkin method on polygonal grids. *Finite Elements in Analysis and Design*, 159:1–14, 2019.

[8] E. Artioli, L. Beirão da Veiga, and F. Dassi. Curvilinear Virtual Elements for 2D solid mechanics applications. *Computer Methods in Applied Mechanics and Engineering*, 359:112667, 2 2020.

[9] E. Artioli, L. da Veiga, and M. Verani. An adaptive curved virtual element method for the statistical homogenization of random fibre-reinforced composites. *Finite Elem. Anal. Des.*, 177:12,103418, 2020.

[10] F. Aurenhammer. Voronoi diagrams—a survey of a fundamental geometric data structure. *ACM Computing Surveys (CSUR)*, 23(3):345–405, 1991.

[11] B. Ayuso De Dios, K. Lipnikov, and G. Manzini. The nonconforming virtual element method. *ESAIM: Mathematical Modelling and Numerical Analysis*, 50(3):879–904, 5 2016.

[12] E. Bachini, G. Manzini, and M. Putti. Arbitrary-order intrinsic virtual element method for elliptic equations on surfaces. *Calcolo*, 58(3):30, 9 2021.

[13] M. J. Baines. *Moving finite elements*. Oxford University Press, Inc., 1994.

[14] M. J. Baines, M. E. Hubbard, and P. K. Jimack. A moving mesh finite element algorithm for the adaptive solution of time-dependent partial differential equations with moving boundaries. *Applied Numerical Mathematics*, 54(3-4):450–469, 8 2005.

[15] M. J. Baines, M. E. Hubbard, and P. K. Jimack. Velocity-Based Moving Mesh Methods for Nonlinear Partial Differential Equations. *Communications in Computational Physics*, 10(3):509–576, 9 2011.

[16] M. J. Baines, M. E. Hubbard, P. K. Jimack, and A. C. Jones. Scale-invariant moving finite elements for nonlinear partial differential equations in two dimensions. *Applied Numerical Mathematics*, 56(2):230–252, 2 2006.

[17] M. J. Baines, M. E. Hubbard, P. K. Jimack, and R. Mahmood. A moving-mesh finite element method and its application to the numerical solution of phase-change problems. *Communications in Computational Physics*, 6(3):595–624, 2009.

[18] 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(01):199–214, 1 2013.

[19] L. Beirão Da Veiga, F. Brezzi, L. D. Marini, and A. Russo. Mixed virtual element methods for general second order elliptic problems on polygonal meshes. *ESAIM: Mathematical Modelling and Numerical Analysis*, 50(3), 2016.

[20] 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, 2016.

[21] L. Beirão da Veiga, F. Brezzi, L. D. Marini, and A. Russo. Polynomial preserving virtual elements with curved edges. *Mathematical Models and Methods in Applied Sciences*, 30(08):1555–1590, 7 2020.

[22] L. Beirão da Veiga, D. Mora, G. Rivera, and R. Rodríguez. A virtual element method for the acoustic vibration problem. *Numerische Mathematik*, 136(3):725–763, 7 2017.

[23] L. Beirão da Veiga, A. Russo, and G. Vacca. The Virtual Element Method with curved edges. *ESAIM: Mathematical Modelling and Numerical Analysis*, 53(2):375–404, 3 2019.

[24] M. F. Benedetto, S. Berrone, S. Pieraccini, and S. Scialò. The virtual element method for discrete fracture network simulations. *Computer Methods in Applied Mechanics and Engineering*, 280:135–156, 10 2014.

[25] S. Berrone, A. Borio, and G. Manzini. SUPG stabilization for the nonconforming virtual element method for advection–diffusion–reaction equations. *Computer Methods in Applied Mechanics and Engineering*, 340, 2018.

[26] S. Bertoluzza, M. Pennacchio, and D. Prada. High order VEM on curved domains. *Atti della Accademia Nazionale dei Lincei, Classe di Scienze Fisiche, Matematiche e Naturali, Rendiconti Lincei Matematica e Applicazioni*, 30(2):391–412, 2019.

[27] A. Bonito, I. Kyza, and R. H. Nochetto. Time-discrete higher order ALE formulations: A priori error analysis. *Numerische Mathematik*, 125(2):225–257, 2013.

[28] A. Bonito, I. Kyza, and R. H. Nochetto. Time-Discrete Higher-Order ALE Formulations: Stability. *SIAM Journal on Numerical Analysis*, 51(1):577–604, 1 2013.

[29] S. C. Brenner and L. R. Scott. *The Mathematical Theory of Finite Element Methods*, volume 15 of *Texts in Applied Mathematics*. Springer New York, New York, NY, 2008.

[30] S. C. Brenner and L. Y. Sung. Virtual element methods on meshes with small edges or faces. *Mathematical Models and Methods in Applied Sciences*, 28(7):1291–1336, 2018.

[31] F. Brezzi, R. S. Falk, and L. Donatella Marini. Basic principles of mixed virtual element methods. *ESAIM: Mathematical Modelling and Numerical Analysis*, 48(4), 2014.

[32] F. Brezzi, T. J. R. Hughes, L. D. Marini, and A. Masud. Mixed Discontinuous Galerkin Methods for Darcy Flow. *Journal of Scientific Computing*, 22-23(1-3):119–145, 6 2005.

[33] F. Brezzi and L. D. Marini. Virtual Element Methods for plate bending problems. *Computer Methods in Applied Mechanics and Engineering*, 253:455–462, 1 2013.

[34] C. J. Budd, W. Huang, and R. D. Russell. Moving Mesh Methods for Problems with Blow-Up. *SIAM Journal on Scientific Computing*, 17(2):305–327, 3 1996.

[35] A. Burden, R. L. Burden, and J. Douglas Faires. *Numerical Analysis, 10th ed.*, volume 10. 2016.

[36] A. Cangiani, P. Chatzipantelidis, G. Diwan, and E. H. Georgoulis. Virtual element method for quasilinear elliptic problems. *IMA Journal of Numerical Analysis*, 40(4):2450–2472, 2021.

[37] A. Cangiani, Z. Dong, E. H. Georgoulis, and P. Houston. *hp-Version Discontinuous Galerkin Methods on Polygonal and Polyhedral Meshes*. Springer Briefs in Mathematics. Springer International Publishing, Cham, 2017.

[38] A. Cangiani, E. H. Georgoulis, and O. J. Sutton. Adaptive non-hierarchical Galerkin methods for parabolic problems with application to moving mesh and virtual element methods. *Mathematical Models and Methods in Applied Sciences*, 31(4):711–751, 2021.

[39] A. Cangiani, G. Manzini, and O. J. Sutton. Conforming and nonconforming virtual element methods for elliptic problems. *IMA Journal of Numerical Analysis*, page drw036, 8 2016.

[40] W. Cao, W. Huang, and R. D. Russell. A Moving Mesh Method Based on the Geometric Conservation Law. *SIAM Journal on Scientific Computing*, 24(1):118–142, 1 2002.

[41] P. G. Ciarlet. *The Finite Element Method for Elliptic Problems*. Society for Industrial and Applied Mathematics, 1 2002.

[42] P. G. Ciarlet and P. A. Raviart. Interpolation theory over curved elements, with applications to finite element methods. *Computer Methods in Applied Mechanics and Engineering*, 1(2):217–249, 8 1972.

[43] L. B. da Veiga, F. Brezzi, and L. D. Marini. Virtual Elements for Linear Elasticity Problems. *SIAM Journal on Numerical Analysis*, 51(2):794–812, 1 2013.

[44] L. B. Da Veiga, F. Brezzi, L. D. Marini, A. Russo, F. Brezzi, and G. Manzini. The Hitchhiker's guide to the virtual element method. *Mathematical Models and Methods in Applied Sciences*, 24(8):1541–1573, 2014.

[45] L. B. da Veiga, C. Lovadina, and G. Vacca. Divergence free virtual elements for the stokes problem on polygonal meshes. *ESAIM: Mathematical Modelling and Numerical Analysis*, 51(2):509–535, 3 2017.

[46] F. Dassi, A. Fumagalli, D. Losapio, S. Scialò, A. Scotti, and G. Vacca. The mixed virtual element method on curved edges in two dimensions. *Computer Methods in Applied Mechanics and Engineering*, 386, 12 2021.

[47] F. Dassi, A. Fumagalli, A. Scotti, and G. Vacca. Bend 3d mixed virtual element method for Darcy problems. *Computers & Mathematics with Applications*, 119:1–12, 8 2022.

[48] A. Dedner and A. Hodson. A framework for implementing general virtual element spaces. *arXiv preprint arXiv:2208.08978.*, 8 2022.

[49] D. A. Di Pietro and J. Droniou. *The Hybrid High-Order Method for Polytopal Meshes*, volume 19 of *MS&A*. Springer International Publishing, Cham, 2020.

[50] J. Donea, A. Huerta, J.-P. Ponthot, and A. Rodríguez-Ferran. Arbitrary Lagrangian-Eulerian Methods. In *Encyclopedia of Computational Mechanics*, pages 1–25. John Wiley & Sons, Ltd, Chichester, UK, 11 2004.

[51] Q. Du, M. Emelianenko, and L. Ju. Convergence of the Lloyd Algorithm for Computing Centroidal Voronoi Tessellations. *SIAM Journal on Numerical Analysis*, 44(1):102–119, 1 2006.

[52] G. Dziuk and C. M. Elliott. Finite elements on evolving surfaces. *IMA Journal of Numerical Analysis*, 27(2):262–292, 2007.

[53] G. Dziuk and C. M. Elliott. Finite element methods for surface PDEs. *Acta Numerica*, 22:289–396, 2013.

[54] L. Formaggia and F. Nobile. Stability analysis for the arbitrary Lagrangian Eulerian formulation with finite elements. *East-West Journal of Numerical Mathematics*, 7(2):105–131, 1999.

[55] L. Formaggia and F. Nobile. Stability analysis of second-order time accurate schemes for ALE–FEM. *Computer Methods in Applied Mechanics and Engineering*, 193(39-41):4097–4116, 10 2004.

[56] M. Frittelli, A. Madzvamuse, and I. Sgura. Bulk-surface virtual element method for systems of PDEs in two-space dimensions. *Numerische Mathematik*, 147(2):305–348, 2 2021.

[57] M. Frittelli and I. Sgura. Virtual Element Method for the Laplace-Beltrami equation on surfaces. *ESAIM: Mathematical Modelling and Numerical Analysis*, 52(3):965–993, 5 2018.

[58] E. Gaburro. A Unified Framework for the Solution of Hyperbolic PDE Systems Using High Order Direct Arbitrary-Lagrangian–Eulerian Schemes on Moving Unstructured Meshes with Topology Change. *Archives of Computational Methods in Engineering*, 28(3):1249–1321, 2021.

[59] E. Gaburro, W. Boscheri, S. Chiocchetti, C. Klingenberg, V. Springel, and M. Dumbser. High order direct Arbitrary-Lagrangian-Eulerian

schemes on moving Voronoi meshes with topology changes. *Journal of Computational Physics*, 407, 2020.

[60] L. Gastaldi. A priori error estimates for the Arbitrary Lagrangian Eulerian formulation with finite elements. *Journal of Numerical Mathematics*, 9(2):123–156, 1 2001.

[61] G. N. Gatica, M. Munar, and F. A. Sequeira. A mixed virtual element method for the Navier–Stokes equations. *Mathematical Models and Methods in Applied Sciences*, 28(14):2719–2762, 12 2018.

[62] R. J. Gelinas, S. K. Doss, and K. Miller. The moving finite element method: applications to general partial differential equations with multiple large gradients. *Journal of Computational Physics*, 40(1):202–249, 1981.

[63] C. Geuzaine and J.-F. 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(11):1309–1331, 9 2009.

[64] J. O. Hallquist, G. L. Goudreau, and D. J. Benson. Sliding interfaces with contact-impact in large-scale Lagrangian computations. *Computer methods in applied mechanics and engineering*, 51(1-3):107–137, 1985.

[65] W. Huang and R. D. Russell. *Adaptive moving mesh methods*, volume 174. Springer Science & Business Media, 2011.

[66] W. Huang and Y. Wang. Anisotropic mesh quality measures and adaptation for polygonal meshes. *Journal of Computational Physics*, 410:109368, 6 2020.

[67] M. E. Hubbard, M. J. Baines, and P. K. Jimack. Consistent Dirichlet boundary conditions for numerical solution of moving boundary problems. *Applied Numerical Mathematics*, 59(6), 2009.

[68] P. K. Jimack and A. J. Wathen. Temporal Derivatives in the Finite-Element Method on Continuously Deforming Grids. *SIAM Journal on Numerical Analysis*, 28(4):990–1003, 8 1991.

[69] G. Karypis and V. Kumar. A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs. *SIAM Journal on Scientific Computing*, 20(1):359–392, 1 1998.

[70] B. Kovács. High-order evolving surface finite element method for parabolic problems on evolving surfaces. *IMA Journal of Numerical Analysis*, 38(1):430–459, 2018.

[71] B. Kovács. Computing arbitrary Lagrangian Eulerian maps for evolving surfaces. *Numerical Methods for Partial Differential Equations*, 35(3):1093–1112, 2019.

[72] B. Kovács, B. Li, and C. Lubich. A convergent evolving finite element algorithm for mean curvature flow of closed surfaces. *Numerische Mathematik*, 143(4):797–853, 2019.

[73] B. Kovács, B. Li, C. Lubich, and C. A. Power Guerra. Convergence of finite elements on an evolving surface driven by diffusion on the surface. *Numerische Mathematik*, 137(3):643–689, 2017.

[74] M. G. Larson and F. Bengzon. *The Finite Element Method: Theory, Implementation, and Applications*, volume 10 of *Texts in Computational Science and Engineering*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[75] M. Lenoir. Optimal Isoparametric Finite Elements and Error Estimates for Domains Involving Curved Boundaries. *SIAM Journal on Numerical Analysis*, 23(3):562–580, 6 1986.

[76] J. Li, X. Feng, and Y. He. Local tangential lifting virtual element method for the diffusion–reaction equation on the non-flat Voronoi discretized surface. *Engineering with Computers*, 1 2022.

[77] K. Lipnikov, G. Manzini, and M. Shashkov. Mimetic finite difference method. *Journal of Computational Physics*, 257(PB):1163–1227, 1 2014.

[78] K. Lipnikov and N. Morgan. A high-order conservative remap for discontinuous Galerkin schemes on curvilinear polygonal meshes. *Journal of Computational Physics*, 399:108931, 2019.

[79] K. Lipnikov and N. Morgan. A high-order discontinuous Galerkin method for level set problems on polygonal meshes. *Journal of Computational Physics*, 397:108834, 11 2019.

[80] K. Lipnikov and N. Morgan. Conservative high-order discontinuous Galerkin remap scheme on curvilinear polyhedral meshes. *Journal of Computational Physics*, 420:109712, 11 2020.

[81] R. Marlow, M. E. Hubbard, and P. K. Jimack. Moving mesh methods for solving parabolic partial differential equations. *Computers & Fluids*, 46(1):353–361, 7 2011.

[82] A. Masud and T. J. Hughes. A stabilized mixed finite element method for Darcy flow. *Computer Methods in Applied Mechanics and Engineering*, 191(39-40):4341–4370, 8 2002.

[83] A. Mazzia, M. Ferronato, P. Teatini, and C. Zoccarato. Virtual element method for the numerical simulation of long-term dynamics of transitional environments. *Journal of Computational Physics*, 407, 2020.

[84] P. A. Mendes and F. A. Branco. Analysis of fluid-structure interaction by an arbitrary Lagrangian-Eulerian finite element formulation. *International Journal for Numerical Methods in Fluids*, 30(7):897–919, 8 1999.

[85] K. Miller and R. N. Miller. Moving finite elements. I. *SIAM Journal on Numerical Analysis*, 18(6):1019–1032, 1981.

[86] J. D. Murray. *Mathematical Biology: 1. An Introduction (3rd edition)*. Springer-Verlag New York, 2002.

[87] C. Ngo and W. Huang. A study on moving mesh finite element solution of the porous medium equation. *Journal of Computational Physics*, 331:357–380, 2 2017.

[88] A. Ortiz-Bernardin, C. Alvarez, N. Hitschfeld-Kahler, A. Russo, R. Silva-Valenzuela, and E. Olate-Sanzana. Veamy: an extensible object-oriented C++ library for the virtual element method. *Numerical Algorithms*, 82(4):1189–1220, 12 2019.

[89] T. Richter. *Fluid-structure Interactions*, volume 118 of *Lecture Notes in Computational Science and Engineering*. Springer International Publishing, Cham, 2017.

[90] O. Sander. *DUNE – The Distributed and Unified Numerics Environment*, volume 140 of *Lecture Notes in Computational Science and Engineering*. Springer International Publishing, Cham, 2020.

[91] M. Senechal, A. Okabe, B. Boots, and K. Sugihara. Spatial Tessellations: Concepts and Applications of Voronoi Diagrams. *The College Mathematics Journal*, 26(1):79, 1995.

[92] E. M. Stein. *Singular Integrals and Differentiability Properties of Functions (PMS-30)*. Princeton University Press, 12 1971.

[93] O. J. Sutton. The virtual element method in 50 lines of MATLAB. *Numerical Algorithms*, 75(4):1141–1159, 8 2017.

[94] C. Talischi, G. H. Paulino, A. Pereira, and I. F. M. Menezes. PolyMesher: a general-purpose mesh generator for polygonal elements written

in Matlab. *Structural and Multidisciplinary Optimization*, 45(3):309–328, 3 2012.

[95] G. Vacca. Virtual Element Methods for hyperbolic problems on polygonal meshes. *Computers and Mathematics with Applications*, 74(5):882–898, 2017.

[96] G. Vacca and L. Beirão Da Veiga. Virtual element methods for parabolic problems on polygonal meshes. *Numerical Methods for Partial Differential Equations*, 31(6):2110–2134, 2015.

[97] J. L. Vazquez. *The Porous Medium Equation*. Oxford University Press, 10 2006.

[98] E. L. Wachspress and S. M. Rohde. A Rational Finite Element Basis. *Journal of Lubrication Technology*, 98(4):635–635, 1976.

[99] H. Wells, M. E. Hubbard, and A. Cangiani. A Velocity-based Moving Mesh Virtual Element Method. *arXiv preprint arXiv:2211.13521*, 11 2022.

[100] P. Wesseling. *Principles of Computational Fluid Dynamics*, volume 29 of *Springer Series in Computational Mathematics*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001.

[101] P. Wriggers. *Computational Contact Mechanics*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2 edition, 2006.

[102] P. Wriggers, W. T. Rust, and B. D. Reddy. A virtual element method for contact. *Computational Mechanics*, 58(6):1039–1050, 12 2016.