A LEAST SQUARES METHOD FOR MIXED VARIATIONAL FORMULATIONS OF PARTIAL DIFFERENTIAL EQUATIONS

by

Jacob Jacavage

A dissertation submitted to the Faculty of the University of Delaware in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Mathematics

Summer 2019

A LEAST SQUARES METHOD FOR MIXED VARIATIONAL FORMULATIONS OF PARTIAL DIFFERENTIAL EQUATIONS

by

Jacob Jacavage

Approved: _____

Louis Rossi, Ph.D.
Chair of the Department of Mathematical Sciences

Approved: _____

John Pelesko, Ph.D.
Interim Dean of the College of Arts and Sciences

Approved: _____

Douglas J. Doren, Ph.D.
Interim Vice Provost for Graduate and Professional Education

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Constantin Bacuta, Ph.D.
Professor in charge of dissertation

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Peter Monk, Ph.D.
Member of dissertation committee

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Philippe Guyenne, Ph.D.
Member of dissertation committee

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Xiaoming Li, Ph.D.
Member of dissertation committee

ACKNOWLEDGEMENTS

First and foremost, I want to thank my advisor, Prof. Constantin Bacuta, for his patience and guidance throughout my research endeavours at the University of Delaware. This thesis would not be where it is without your constant encouragement and optimism. I want to also thank his wife, Prof. Cristina Bacuta, for her guidance and encouragement in other aspects of my professional development.

I would also like to thank the other members of my thesis committee: Prof. Peter Monk, Prof. Philippe Guyenne, and Prof. Xiaoming Li. All of you have had a positive impact on my studies whether it be through course work, conversations around the department, or conversations about my research through an outside perspective. It was Prof. Monk's course on the finite element method that introduced me to the topic. A special thank you to Prof. Francisco Sayas for helping me gain a much higher level of mathematical maturity. Through course work, he pushed me beyond my limits and greatly expanded my mathematical abilities. Prof. Sayas is sadly no longer with us, but will be remembered for all of his professional advice.

The Department of Mathematrical Sciences at University of Delaware has been very supportive throughout my time in Delaware. I would like to thank the graduate students in the department as a whole for their support and the friendships that I have made. In particular, I want to thank my office mates Ben and Sam who were always up for a conversation about mathematics or, most often, topics completely unrelated. Also, I want to thank Kris for taking me on a "victory lap" ride on his motorcycle after my thesis defense. A special thank you to Deborah See for answering all of my questions regarding forms to fill out and other administrative matters.

I want to thank my parents, Mick and Kim Jacavage, for their support along the way. I want to also thank my fiancée, Sarah, for her constant love and support, as well

as putting up with conversations about mathematics on a regular basis. Lastly, I want to thank the faculty at Bloomsburg University who saw the potential in my abilities and pushed me to pursue a graduate degree in mathematics. Their encouragement led me to where I am today.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

We present a general framework for solving mixed variational formulations of partial differential equations. The method relates the theories of least squares finite element methods, approximating solutions to elliptic boundary value problems, and approximating solutions to symmetric saddle point problems. A general preconditioning strategy for the proposed framework is also given that utilizes the theory of multilevel preconditioners. One of the main advantages of the method is that an inf $-$ sup condition is automatically satisfied at the discrete level for standard choices of test and trial spaces. Another benefit is that the method allows for the use of nonconforming trial spaces. In addition, the framework allows the freedom to choose the inner product on the test space, which is useful when solving PDEs, or first order systems of PDEs, with parameters and/or discontinuous coefficients. The proposed iterative solver does not require explicit bases for the trial spaces as well. Applications of the method to second order elliptic interface problems, reaction diffusion equations, and time-harmonic Maxwell's equations are presented. Numerical results in 2D and 3D, for both convex and non-convex domains, are given to support the methodology, including problems with discontinuous or highly oscillatory coefficients, low regularity of the solution, and boundary layers.

Chapter 1

INTRODUCTION

Mixed variational formulations for partial differential equations arise naturally when modeling physical systems. Some examples include diffusion through heterogeneous porous media, electromagnetism, elasticity, acoustics, and fluid flow. When approximating the physical quantities from these systems, such as the flux of the solution or the electric and magnetic fields, it is beneficial to obtain estimates that are robust with respect to the parameters associated with the system. Furthermore, it is important to obtain these robust estimates in the presence of numerical challenges, such as discontinuities in the material coefficients, low regularity data, or low regularity solutions (perhaps near the boundary). Recently, there has been a lot of research in the direction of applying least squares finite element methods to these problems [27, 28, 29, 41, 42, 43, 44, 65, 66, 73]. The methodology described in this thesis provides a unified theory of least squares methods for solving a large class of PDEs that can be written as mixed variational formulations. Furthermore, the theory will utilize efficient conforming and nonconforming approximation spaces, multilevel techniques, and residual error estimation.

We are interested in approximating the solution to PDEs which can be written as: Find $p \in Q$ such that

$$b(v,p) = \langle F, v\rangle \quad \text{for all } v \in V. \tag{1.0.1}$$

In the above equation, V and Q are infinite dimensional Hilbert spaces, F is a bounded, linear functional on V, and $b(\cdot,\cdot)$ is a bilinear form defined on $V \times \tilde{Q}$ (the space $\tilde{Q}$ is an infinite dimensional Hilbert space that contains Q). Instead of solving (1.0.1) directly,

we consider a "saddle point reformulation". More specifically, by denoting $a(\cdot,\cdot)$ as the inner product on V, we obtain the following saddle point reformulation of problem (1.0.1): Find $(w = 0, p) \in V \times Q$ such that

$$\begin{aligned} a(w,v) \quad + \quad b(v,p) \quad &= \langle F, v\rangle \qquad \text{for all } v \in V, \\ b(w,q) \qquad\qquad\quad &= 0 \qquad\qquad \text{for all } q \in Q. \end{aligned}$$

Under appropriate assumptions on $b(\cdot,\cdot)$ and the linear functional F, the p component of the solution to the saddle point reformulation solves the original problem, see Chapter 2. One advantage of solving the saddle point problem instead of (1.0.1) directly resides in the fact that we can construct and utilize discrete finite element trial spaces that have desirable approximability properties. Another advantage is that we are free to choose the inner product on V that we use in the saddle point reformulation. This fact is of particular importance when studying PDEs with parameters, and/or discontinuous coefficients. In addition, we can apply the classical approximation theory for symmetric saddle point systems.

The proposed method for solving problems of the form (1.0.1) is related to the Saddle Point Least Squares (SPLS) method introduced in [20]. The SPLS method combines the theory and discretization techniques for approximating solutions to elliptic boundary value problems with the theory of approximating solutions to symmetric saddle point problems [5, 11, 12, 25, 37, 39, 40, 54, 75, 80, 83]. The benefits of this approach are that a discrete $\inf - \sup$ condition is automatically satisfied for natural choices of test and trial spaces and the implentation of the proposed solver does not require explicit bases for the trial spaces. The SPLS framework has been applied to $\text{div} - \text{curl}$ systems [20], as well as second order problems [21], using conforming finite element spaces for both the test and trial spaces.

The approach taken in this thesis can be viewed as an extended version of the original SPLS method. The essential difference is that the new framework allows for the possibility of nonconforming trial spaces. This will allow for better approximation of the physical quantities associated with the PDE, especially in the case of discontinuous coefficients. The benefits from the original SPLS method carry over in that a discrete

inf − sup condition is automatically satisfied for particular choices of test and trial spaces, and explicit bases for the trial spaces are not needed. If the solution space Q is of L^2 type, the extended framework reduces to the original SPLS framework (Chapter 6 provides an application of the method for this case). We also combine the approach with multilevel preconditioning techniques in order to address particular challenges of the PDE to be solved due to discontinuous coefficients or multidimensional domains.

The method can also be viewed as a new Discontinuous Petrov-Galerkin (DPG) method, which was introduced by Demkowicz and Gopalakrishnan and is currently undergoing an intensive study, see e.g., [49, 50, 51]. While both methods have strong connections with least squares and minimum residual techniques, the proposed discretization process stands apart from the DPG approach due to the different ways in which the trial and test spaces are chosen. In the approach presented in this thesis, a discrete test space is chosen first. The trial space is then built from the test space using the action of the continuous differential operator associated with the problem in order to satisfy a discrete inf − sup condition. In the DPG method, the order of building the spaces is reversed. In addition, we focus on the discrete inf − sup condition first and the approximability properties second, while the DPG method reverses the focus. For these reasons, this approach can be thought of as a dual to the DPG method.

This thesis is organized as follows. Chapter 2 describes the extended SPLS framework and its connection with the original SPLS method. In Chapter 3, a variant of the framework that utilizes the theory of elliptic preconditioning is introduced and analyzed. Chapters 4 and 5 involve the application of the framework to second order elliptic interface problems and reaction diffusion equations, respectively. The application of the framework to the time-harmonic Maxwell equations is presented in Chapter 6. A discussion of future work, including other possible applications for the method, is presented in Chapter 7.

Chapter 2

THE SADDLE POINT LEAST SQUARES THEORY

In this chapter, we will introduce and analyze the extended SPLS framework. Recall the abstract variational problems of interest: Find $p \in Q$ such that

$$b(v, p) = \langle F, v \rangle \quad \text{for all } v \in V. \tag{2.0.1}$$

In the original SPLS framework [20], the form $b(\cdot, \cdot)$ is defined on $V \times Q$. In constrast, we will assume that $b(\cdot, \cdot)$ is defined on $V \times \tilde{Q}$, where $Q \subset \tilde{Q}$ is a closed subspace. This extension of the form $b(\cdot, \cdot)$ is essential for the discretization of the problem, and it will be the main assumption that allows for the possibility of a nonconforming trial space. While the general structure of the approach presented in this chapter and the original SPLS approach share their similarities, different techniques are required to analyze the approximability properties of the chosen discrete trial spaces due to the definition and assumptions on the form $b(\cdot, \cdot)$. In addition, there are subtle differences between the proposed iterative solver and the analysis of its convergence properties. For simplicity, we will refer to the methodology presented as the SPLS method instead of the extended SPLS method for the remainder of the thesis. This chapter is published in [15].

This chapter is organized as follows. In Section 2.1, we will introduce the abstract theory for the SPLS method and discuss the solvability of problem (2.0.1). The abstract discretization theory is presented in Section 2.2. Several choices for the discrete trial spaces are analyzed in Section 2.3. In Section 2.4, an Uzawa Conjugate Gradient algorithm is outlined to solve the corresponding discrete formulation. The convergence properties of the algorithm are also analyzed. Section 2.5 briefly discusses the special case in which the extended framework coincides with the original SPLS method.

2.1 The Saddle Point Least Squares Approach

We let V and $\tilde{Q}$ be two infinite dimensional Hilbert spaces equipped with inner products $a(\cdot,\cdot)$ and $(\cdot,\cdot)_{\tilde{Q}}$, respectively, and assume that Q is a closed subspace of $\tilde{Q}$ equipped with the induced inner product. We further assume that the inner products induce the norms $|\cdot|_V := a(\cdot,\cdot)^{1/2}$ and $\|\cdot\|_{\tilde{Q}} := (\cdot,\cdot)_{\tilde{Q}}^{1/2}$. The dual spaces of V and $\tilde{Q}$ will be denoted by V^* and $\tilde{Q}^*$, respectively. The dual pairings on $V^* \times V$ and $\tilde{Q}^* \times \tilde{Q}$ will both be denoted by $\langle\cdot,\cdot\rangle$. With the inner product $a(\cdot,\cdot)$, we associate the operator $\mathcal{A} : V \to V^*$ defined by

$$\langle \mathcal{A}u, v\rangle = a(u,v) \quad \text{for all } u, v \in V.$$

We assume that $b(\cdot,\cdot) : V \times \tilde{Q} \to \mathbb{R}$ is a bilinear form satisfying

$$\sup_{p\in\tilde{Q}} \sup_{v\in V} \frac{b(v,p)}{|v|_V \|p\|_{\tilde{Q}}} = M < \infty, \tag{2.1.1}$$

and the $\inf - \sup$ condition

$$\inf_{p\in Q} \sup_{v\in V} \frac{b(v,p)}{|v|_V \|p\|_{\tilde{Q}}} = m > 0. \tag{2.1.2}$$

Note that while we assume that $b(\cdot,\cdot)$ is continuous on $V \times \tilde{Q}$, the $\inf - \sup$ condition is assumed on $V \times Q$. This assumption is essential to discuss the solvability of problem (2.0.1). With the form $b(\cdot,\cdot)$, we associate the linear operators $B : V \to \tilde{Q}$ and $B^* : \tilde{Q} \to V^*$ defined by

$$(Bv, q)_{\tilde{Q}} = b(v,q) = \langle B^* q, v\rangle \quad \text{for all } v \in V,\ q \in \tilde{Q}.$$

Here, the operator B is defined through the inner product on $\tilde{Q}$, while the operator B^* is defined through the duality pairing on $V^* \times V$. We also define

$$V_0 := \{v \in V \,|\, b(v,q) = 0, \text{ for all } q \in \tilde{Q}\} = \mathrm{Ker}(B).$$

The solvability of problem (2.0.1) is well known and was first studied by Aziz and Babuška. The following Lemma can be found in [5].

Lemma 2.1.1. *(Babuška) Let $b(\cdot,\cdot)$ be a bilinear form satisfying (2.1.1) and (2.1.2) and $F \in V^*$. Then problem (2.0.1) has a unique solution $p \in Q$ if and only if F satisfies the compatibility condition*

$$\langle F, v\rangle = 0 \quad \text{for all } v \in V_0. \tag{2.1.3}$$

Instead of solving problem (2.0.1) directly, we adopt a "saddle point reformulation". More specifically, with problem (2.0.1) we associate the following symmetric saddle point problem by introducing an auxiliary variable w: Find $(w,p) \in V \times Q$ such that

$$\begin{aligned} a(w,v) \;+\; b(v,p) &= \langle F, v\rangle \qquad \text{for all } v \in V, \\ b(w,q) \qquad\quad\; &= 0 \qquad\quad\; \text{for all } q \in Q. \end{aligned} \tag{2.1.4}$$

Throughout this thesis, the variable w will always refer to the variable introduced for the purposes of the saddle point reformulation. The following Proposition can be found in [17, 48] and is essential to the SPLS approach.

Proposition 2.1.2. *In the presence of the continuous* inf − sup *condition (2.1.2) and the compatibility condition (2.1.3), we have that p is the unique solution of (2.0.1) if and only if $(w = 0, p)$ is the unique solution of (2.1.4).*

For the rest of this chapter, we assume that the compatibility condition (2.1.3) holds. Consequently, problem (2.0.1) has a unique solution.

2.2 Saddle Point Least Squares Discretization

In this section, we discuss the abstract theory for the discretization of the SPLS approach. Let $V_h \subset V$ and $\mathcal{M}_h \subset \tilde{Q}$ be finite dimensional approximation spaces, and consider the restrictions of the forms $a(\cdot,\cdot)$ and $b(\cdot,\cdot)$ to $V_h \times V_h$ and $V_h \times \mathcal{M}_h$, respectively. We define the operator A_h to be the discrete analog of the operator $\mathcal{A}$ from the previous section, i.e., A_h satisfies

$$\langle A_h u_h, v_h\rangle = a(u_h, v_h) \quad \text{for all } u_h, v_h \in V_h.$$

We also define the discrete operators $B_h : V_h \to \mathcal{M}_h$ and $B_h^* : \mathcal{M}_h \to V_h^*$ by

$$(B_h v_h, q_h)_{\tilde{Q}} = b(v_h, q_h) = \langle B_h^* q_h, v_h \rangle \quad \text{for all } v_h \in V_h, q_h \in \mathcal{M}_h.$$

Similar to the way the operator B is defined at the continuous level in Section 2.1, the operator B_h is defined using the inner product on $\mathcal{M}_h$ and not with the duality on $\mathcal{M}_h^* \times \mathcal{M}_h$. We assume the discrete $\inf - \sup$ condition

$$\inf_{p_h \in \mathcal{M}_h} \sup_{v_h \in V_h} \frac{b(v_h, p_h)}{|v_h|_V \|p_h\|_{\tilde{Q}}} = m_h > 0, \tag{2.2.1}$$

holds for the pair of spaces $(V_h, \mathcal{M}_h)$ and define

$$M_h := \sup_{p_h \in \mathcal{M}_h} \sup_{v_h \in V_h} \frac{b(v_h, p_h)}{|v_h|_V \|p_h\|_{\tilde{Q}}} \leq M < \infty. \tag{2.2.2}$$

With the above setting, we can define the discrete Schur complement $S_h : \mathcal{M}_h \to \mathcal{M}_h$ as

$$S_h := B_h A_h^{-1} B_h^*.$$

The following lemma can be found in [7].

Lemma 2.2.1. *The operator S_h is a symmetric and positive definite operator satisfying*

$$(S_h p_h, p_h)_{\tilde{Q}} = \sup_{v_h \in V_h} \frac{b(v_h, p_h)^2}{|v_h|_V^2}.$$

Consequently, $m_h^2, M_h^2 \in \sigma(S_h)$ and

$$\sigma(S_h) \subset [m_h^2, M_h^2].$$

We define

$$V_{h,0} := \{v_h \in V_h \,|\, b(v_h, q_h) = 0 \quad \text{for all } q_h \in \mathcal{M}_h\},$$

to be the kernel of the discrete operator B_h and $f_h \in V_h^*$ to be the restriction of F to V_h, i.e.,

$$\langle f_h, v_h \rangle := \langle F, v_h \rangle \quad \text{for all } v_h \in V_h.$$

Lemma 2.2.2. *If $b(\cdot,\cdot)$ satisfies (2.2.1) and $V_{h,0} \subset V_0$, then the compatibility condition (2.1.3) implies the discrete compatibility condition*

$$\langle f_h, v_h \rangle = 0 \quad \text{for all } v_h \in V_{h,0},$$

and the discrete problem of finding $p_h \in \mathcal{M}_h$ such that

$$b(v_h, p_h) = \langle f_h, v_h \rangle \quad \text{for all } v_h \in V_h, \tag{2.2.3}$$

has a unique solution.

Similar to the formulation at the continuous level, instead of working with problem (2.2.3) directly, we associate (2.2.3) with the following discrete saddle point problem: Find $(w_h, p_h) \in V_h \times \mathcal{M}_h$ such that

$$\begin{array}{lcll} a(w_h, v_h) & + & b(v_h, p_h) & = \langle f_h, v_h \rangle \qquad \text{for all } v_h \in V_h, \\ b(w_h, q_h) & & & = 0 \qquad\quad \text{for all } q_h \in \mathcal{M}_h. \end{array} \tag{2.2.4}$$

Remark 2.2.3. *In general, (2.1.3) may not hold on $V_{h,0}$ and problem (2.2.3) may not be well-posed. However, if the bilinear form $b(\cdot,\cdot)$ satisfies (2.2.1) then the problem of finding $(w_h, p_h) \in V_h \times \mathcal{M}_h$ satisfying (2.2.4) does have a unique solution. Solving for p_h from (2.2.4), we obtain*

$$S_h p_h = B_h (A_h^{-1} B_h^*) p_h = B_h A_h^{-1} f_h. \tag{2.2.5}$$

Since the Hilbert transpose of B_h is $A_h^{-1} B_h^*$, the component p_h of the solution to (2.2.4) is the least squares solution of problem (2.2.3).

The above remark justifies the least squares terminology in the approach. We call the component p_h of the solution (w_h, p_h) of (2.2.4) the saddle point least squares approximation to the solution p of the original problem (2.0.1).

2.3 Choices of Discrete Spaces

In this section, we describe two pairs of discrete spaces, based on the same constructions introduced in [20], which satisfy the discrete $\inf - \sup$ condition (2.2.1).

For both pairs of spaces, we let $V_h \subset V$ be a finite element approximation space, and assume the action of B, defined in Section 2.1, is easy to obtain at the continuous level. The trial space $\mathcal{M}_h$ in both cases will be constructed from the test space V_h.

2.3.1 No Projection Trial Space

We first consider the case when $\mathcal{M}_h$ is given by

$$\mathcal{M}_h := BV_h,$$

where the inner product on $\mathcal{M}_h$ is chosen to coincide with the inner product on $\tilde{Q}$. Note that for $v_h \in V_h$

$$b(v_h, q_h) = (Bv_h, q_h)_{\tilde{Q}} \quad \text{for all } q_h \in \mathcal{M}_h.$$

This implies $V_{h,0} \subset V_0$. Also, a discrete $\inf - \sup$ condition holds. Indeed, for a generic $p_h = Bw_h \in \mathcal{M}_h$, where $w_h \in V_{h,0}^{\perp}$, we have

$$\begin{aligned}
\inf_{p_h \in \mathcal{M}_h} \sup_{v_h \in V_h} \frac{b(v_h, p_h)}{|v_h|_V \, \|p_h\|_{\tilde{Q}}} &= \inf_{w_h \in V_{h,0}^{\perp}} \sup_{v_h \in V_h} \frac{(Bv_h, Bw_h)_{\tilde{Q}}}{|v_h|_V \, \|Bw_h\|_{\tilde{Q}}} \\
&\geq \inf_{w_h \in V_{h,0}^{\perp}} \frac{\|Bw_h\|_{\tilde{Q}}^2}{|w_h|_V \, \|Bw_h\|_{\tilde{Q}}} \\
&= \inf_{w_h \in V_{h,0}^{\perp}} \frac{\|Bw_h\|_{\tilde{Q}}}{|w_h|_V} \\
&:= m_{h,0}. \qquad (2.3.1)
\end{aligned}$$

Thus, both variational formulations (2.2.3) and (2.2.4) have a unique solution $p_h \in \mathcal{M}_h$. Furthermore, using Proposition 2.1.2 for the discrete pair of spaces $(V_h, \mathcal{M}_h)$, the pair $(w_h = 0, p_h)$ is the solution of (2.2.4).

If p is the solution of (2.0.1) and p_h is the solution of (2.2.3) (or $(0, p_h)$ is the solution of (2.2.4)), then from (2.0.1) and (2.2.3) we obtain

$$0 = b(v_h, p - p_h) = (Bv_h, p - p_h)_{\tilde{Q}} \quad \text{for all } v_h \in V_h.$$

Thus, p_h is the orthogonal projection of p onto $\mathcal{M}_h$ which gives us

$$\|p - p_h\|_{\tilde{Q}} = \inf_{q_h \in \mathcal{M}_h} \|p - q_h\|_{\tilde{Q}}. \qquad (2.3.2)$$

This result is optimal, and in contrast with the standard approximation estimates for saddle point problems, it does not depend on $m_{h,0}$.

2.3.2 Projection Type Trial Space

For the second type of trial space, we first let $\tilde{\mathcal{M}}_h \subset \tilde{Q}$ be a finite dimensional subspace equipped with the inner product $(\cdot,\cdot)_h$, and define the representation operator $R_h : \tilde{Q} \to \tilde{\mathcal{M}}_h$ by

$$(R_h p, q_h)_h := (p, q_h)_{\tilde{Q}} \quad \text{for all } q_h \in \tilde{\mathcal{M}}_h. \tag{2.3.3}$$

Here, we can view $R_h p$ as the Riesz representation of $q_h \to (p, q_h)_{\tilde{Q}}$ as a functional on $(\tilde{\mathcal{M}}_h, (\cdot,\cdot)_h)$.

Remark 2.3.1. *In the case where $(\cdot,\cdot)_h$ coincides with the inner product on $\tilde{Q}$, R_h is the standard orthogonal projection onto $\tilde{\mathcal{M}}_h$.*

Since the space $\tilde{\mathcal{M}}_h$ is finite dimensional, there exist constants k_1, k_2 such that

$$k_1 \|q_h\|_{\tilde{Q}} \le \|q_h\|_h \le k_2 \|q_h\|_{\tilde{Q}} \quad \text{for all } q_h \in \tilde{\mathcal{M}}_h. \tag{2.3.4}$$

We further assume that the equivalence is uniform with respect to h, i.e., the constants k_1, k_2 are independent of h. Using the operators R_h and B, we define $\mathcal{M}_h$ as

$$\mathcal{M}_h := R_h B V_h \subset \tilde{\mathcal{M}}_h \subset \tilde{Q}.$$

The following proposition gives a sufficient condition on R_h to ensure that a discrete $\inf - \sup$ condition is satisfied and relates the stability of the families of spaces $\{(V_h, BV_h)\}$ and $\{(V_h, R_h BV_h)\}$.

Proposition 2.3.2. *Assume that*

$$\|R_h q_h\|_h \ge \tilde{c}\, \|q_h\|_{\tilde{Q}} \quad \textit{for all } q_h \in BV_h, \tag{2.3.5}$$

with a constant $\tilde{c}$ independent of h. Then $V_{h,0} \subset V_0$. Furthermore, the stability of the family $\{(V_h, BV_h)\}$, meaning $m_{h,0}$ defined in (2.3.1) satisfies $m_{h,0} > c_0 > 0$ for some constant c_0 independent of h, implies the stability of the family $\{(V_h, R_h BV_h)\}$.

Proof. Let $v_h \in V_{h,0}$. For any $p_h \in \mathcal{M}_h$,

$$0 = b(v_h, p_h) = (Bv_h, p_h)_{\tilde{Q}} = (R_h B v_h, p_h)_h.$$

Taking $p_h = R_h B v_h$ gives us $\|R_h B v_h\|_h = 0$, and the inclusion $V_{h,0} \subset V_0$ follows from (2.3.5). For the stability result, consider a generic function $p_h = R_h B w_h \in \mathcal{M}_h$, where $w_h \in V_{h,0}^{\perp}$. We obtain

$$\begin{aligned}
m_h = \inf_{p_h \in \mathcal{M}_h} \sup_{v_h \in V_h} \frac{b(v_h, p_h)}{|v_h|_V \|p_h\|_h} &= \inf_{w_h \in V_{h,0}^{\perp}} \sup_{v_h \in V_h} \frac{(Bv_h, R_h B w_h)_{\tilde{Q}}}{|v_h|_V \|R_h B w_h\|_h} \\
&= \inf_{w_h \in V_{h,0}^{\perp}} \sup_{v_h \in V_h} \frac{(R_h B v_h, R_h B w_h)_h}{|v_h|_V \|R_h B w_h\|_h} \\
&\geq \inf_{w_h \in V_{h,0}^{\perp}} \frac{\|R_h B w_h\|_h^2}{|w_h|_V \|R_h B w_h\|_h} \\
&\geq \tilde{c} \inf_{w_h \in V_{h,0}^{\perp}} \frac{\|B w_h\|_{\tilde{Q}}}{|w_h|_V} \\
&= \tilde{c}\, m_{h,0},
\end{aligned}$$

where $m_{h,0}$ is defined in (2.3.1). □

In practice, this result is beneficial as the no projection type trial space is usually easier to analyze. As a consequence of Proposition 2.3.2, under assumption (2.3.5) both variational formulations (2.2.3) and (2.2.4) have a unique solution $p_h \in \mathcal{M}_h$. Furthermore, using Proposition 2.1.2 for the discrete pair of spaces $(V_h, \mathcal{M}_h)$, the pair $(w_h = 0, p_h)$ is the solution of (2.2.4). The following result shows under condition (2.3.5) that p_h is a quasi-optimal solution to the original problem (2.0.1).

Proposition 2.3.3. *If p is the solution of (2.0.1), p_h is the solution of (2.2.3) (or $(0, p_h)$ is the solution of (2.2.4)), and R_h satisfies (2.3.5), then*

$$\|p - p_h\|_{\tilde{Q}} \leq C \inf_{q_h \in \mathcal{M}_h} \|p - q_h\|_{\tilde{Q}},$$

where C depends only on $\tilde{c}$ of (2.3.5) and the equivalence of norms constants of (2.3.4).

Proof. Under the assumptions on p and p_h, we obtain

$$0 = b(v_h, p - p_h) = (Bv_h, p - p_h)_{\tilde{Q}} \quad \text{for all} \quad v_h \in V_h.$$

In turn, this implies

$$(Bv_h, p - Q_h p)_{\tilde{Q}} = (Bv_h, p_h - Q_h p)_{\tilde{Q}} \quad \text{for all } v_h \in V_h, \tag{2.3.6}$$

where Q_h is the orthogonal projection onto $\mathcal{M}_h$. Note that

$$\|p_h - Q_h p\|_h = \sup_{q_h \in \mathcal{M}_h} \frac{|(p_h - Q_h p, q_h)_h|}{\|q_h\|_h}. \tag{2.3.7}$$

Using (2.3.5) and (2.3.6), we obtain

$$\begin{aligned}
\sup_{q_h \in \mathcal{M}_h} \frac{|(p_h - Q_h p, q_h)_h|}{\|q_h\|_h} &= \sup_{w_h \in V_{h,0}^{\perp}} \frac{|(p_h - Q_h p, R_h B w_h)_h|}{\|R_h B w_h\|_h} \\
&= \sup_{w_h \in V_{h,0}^{\perp}} \frac{|(p_h - Q_h p, B w_h)_{\tilde{Q}}|}{\|R_h B w_h\|_h} \\
&= \sup_{w_h \in V_{h,0}^{\perp}} \frac{|(p - Q_h p, B w_h)_{\tilde{Q}}|}{\|R_h B w_h\|_h} \\
&\leq \sup_{w_h \in V_{h,0}^{\perp}} \frac{\|p - Q_h p\|_{\tilde{Q}} \, \|B w_h\|_{\tilde{Q}}}{\|R_h B w_h\|_h} \\
&\leq \frac{1}{\tilde{c}} \|p - Q_h p\|_{\tilde{Q}}.
\end{aligned}$$

Hence,

$$\|Q_h p - p_h\|_{\tilde{Q}} \leq \frac{1}{k_1} \|Q_h p - p_h\|_h \leq \frac{1}{\tilde{c} k_1} \|p - Q_h p\|_{\tilde{Q}},$$

from (2.3.4), (2.3.7), and the above estimate. Thus,

$$\begin{aligned}
\|p - p_h\|_{\tilde{Q}} &\leq \|p - Q_h p\|_{\tilde{Q}} + \|Q_h p - p_h\|_{\tilde{Q}} \\
&\leq \left(1 + \frac{1}{\tilde{c} k_1}\right) \|p - Q_h p\|_{\tilde{Q}} \\
&= C \inf_{q_h \in \mathcal{M}_h} \|p - q_h\|_{\tilde{Q}}.
\end{aligned}$$

□

Remark 2.3.4. *The no projection trial space described in Section 2.3.1 can be viewed as a special case of the projection type trial space when $R_h = I$ and the $(\cdot, \cdot)_h$ inner product coincides with the inner product on $\tilde{Q}$. Hence, in what follows we will equip the trial space $\mathcal{M}_h$ with the inner product $(\cdot, \cdot)_h$ for both the no projection and projection type spaces for simplicity.*

2.4 An Iterative Solver

When solving (2.2.4) on $(V_h, \mathcal{M}_h = BV_h)$ or $(V_h, \mathcal{M}_h = R_h BV_h)$, a global linear system might be difficult to assemble as simple local bases may be hard to compute for the space $\mathcal{M}_h$, especially for the projection type. Nevertheless, it is possible to solve (2.2.4) without an explicit basis for $\mathcal{M}_h$ by using an Uzawa type iteration process, such as the Uzawa Conjugate Gradient (UCG) algorithm [31, 88].

Algorithm 2.4.1. *(UCG) Algorithm*

Step 1: Set $p_0 = 0 \in \mathcal{M}_h$. **Compute** $w_1 \in V_h$, $q_1, d_1 \in \mathcal{M}_h$ *by*

$$\begin{aligned} a(w_1, v) &= \langle f_h, v\rangle - b(v, p_0) && \text{for all } v \in V_h, \\ (q_1, q)_h &= b(w_1, q) && \text{for all } q \in \mathcal{M}_h, \quad d_1 := q_1. \end{aligned}$$

Step 2: For $j = 1, 2, \ldots,$ **compute** $h_j, \alpha_j, p_j, w_{j+1}, q_{j+1}, \beta_j, d_{j+1}$ *by*

$$\begin{aligned} &(\mathbf{UCG1}) && a(h_j, v) = -\, b(v, d_j) && \text{for all } v \in V_h \\ &(\mathbf{UCG}\alpha) && \alpha_j = -\,\frac{(q_j, q_j)_h}{b(h_j, q_j)} \\ &(\mathbf{UCG2}) && p_j = p_{j-1} + \alpha_j\, d_j \\ &(\mathbf{UCG3}) && w_{j+1} = w_j + \alpha_j\, h_j \\ &(\mathbf{UCG4}) && (q_{j+1}, q)_h = b(w_{j+1}, q) && \text{for all } q \in \mathcal{M}_h \\ &(\mathbf{UCG}\beta) && \beta_j = \frac{(q_{j+1}, q_{j+1})_h}{(q_j, q_j)_h} \\ &(\mathbf{UCG6}) && d_{j+1} = q_{j+1} + \beta_j d_j. \end{aligned}$$

Remark 2.4.2. *Instead of taking the initial iterate $p_0 = 0$ in Step 1 of the UCG algorithm for each level of refinement of a suitable mesh for the problem to be solved, if the refinements are nested we can take an approach in which $p_0 = 0$ on the coarsest mesh, but for all successive refinements p_0 is chosen as the extension of the final iterate from the previous level [8, 9, 10, 17, 22, 30, 32]. This approach will be referred to as the UCG Cascadic algorithm.*

Note that at each iteration step, one inversion involving the form $a(\cdot, \cdot)$ is required. Hence, a basis for V_h is needed. We will now show that an explicit basis for

$\mathcal{M}_h$ is not needed in the algorithm. More specifically, we will show q_1 and q_{j+1} in Steps 1 and UCG4 can be computed through the action of the operators B and R_h. For the no projection choice of trial space outlined in Section 2.3.1, Step UCG4 and the second equation of Step 1 can be written as

$$(q_{j+1}, q)_{\tilde{Q}} = b(w_{j+1}, q) = (Bw_{j+1}, q)_{\tilde{Q}} \quad \text{for all } q \in \mathcal{M}_h.$$

This implies

$$q_{j+1} = Bw_{j+1}.$$

Also, for the choice of a projection type trial space outlined in Section 2.3.2,

$$(q_{j+1}, q)_h = b(w_{j+1}, q) = (Bw_{j+1}, q)_{\tilde{Q}} = (R_h Bw_{j+1}, q)_h \quad \text{for all } q \in \mathcal{M}_h.$$

Hence, q_{j+1} is given by

$$q_{j+1} = R_h Bw_{j+1}.$$

Remark 2.4.3. *The steps involving the updates for the p_j's in Algorithm 2.4.1 recover the steps of the standard Conjugate Gradient Algorithm [59] for solving problem (2.2.5). Due to assumption (2.2.1), S_h is a symmetric, positive definite operator, see Lemma 2.2.1. Consequently, the iterates p_j converge to the solution p_h with a rate of convergence that depends on the condition number of S_h, which is*

$$\kappa(S_h) = \frac{M_h^2}{m_h^2} \leq \frac{M^2}{m_h^2}.$$

Theorem 2.4.4. *If (w_h, p_h) is the discrete solution of (2.2.4) and (w_{j+1}, p_j) is the j^{th} iteration for Algorithm 2.4.1, then $(w_{j+1}, p_j) \to (w_h, p_h)$ and*

$$\begin{aligned} \frac{1}{M^2}\|q_{j+1}\|_h \leq \|p_j - p_h\|_h \leq \frac{1}{m_h^2}\|q_{j+1}\|_h, \\ \frac{m_h}{M^2}\|q_{j+1}\|_h \leq \|w_{j+1} - w_h\|_V \leq \frac{M}{m_h^2}\|q_{j+1}\|_h. \end{aligned} \tag{2.4.1}$$

Proof. By induction over j, we obtain

$$a(w_{j+1}, v_h) + b(v_h, p_j) = \langle f_h, v_h \rangle \quad \text{for all } v_h \in V_h.$$

Combining this with the first equation of (2.2.4) gives us

$$a(w_{j+1} - w_h, v_h) = b(v_h, p_h - p_j) \quad \text{for all } v_h \in V_h. \tag{2.4.2}$$

Note that $\sigma(S_h) \subset [m_h^2, M_h^2]$ from Lemma 2.2.1. Hence,

$$m_h \|q_h\|_h = (S_h q_h, q_h)_h^{1/2} \leq M_h \|q_h\|_h \quad \text{for all } q_h \in \mathcal{M}_h. \tag{2.4.3}$$

By substituting $v_h = A_h^{-1} B_h^*(p_h - p_j)$ into (2.4.2),

$$|w_{j+1} - w_h|_V^2 = (S_h(p_h - p_j), p_h - p_j)_h = \|p_h - p_j\|_{S_h}^2.$$

The above equality and (2.4.3) gives us

$$m_h \|p_h - p_j\|_h \leq |w_{j+1} - w_h|_V \leq M_h \|p_h - p_j\|_h. \tag{2.4.4}$$

From Step UCG4, the second equation of (2.2.4), and (2.4.2) we obtain

$$q_{j+1} = B_h w_{j+1} = B_h(w_{j+1} - w_h) = S_h(p_h - p_j).$$

Thus,

$$m_h^2 \|p_h - p_j\|_h \leq \|S_h(p_h - p_j)\|_h = \|q_{j+1}\|_h \leq M_h^2 \|p_h - p_j\|_h. \tag{2.4.5}$$

The inequalities (2.4.1) follow from (2.4.4) and (2.4.5) and the fact that $M_h \leq M$. From Remark 2.4.3 and the standard estimate for the convergence rate of the conjugate gradient algorithm [31, 59], we obtain the estimate

$$\|p_h - p_j\|_{S_h} \leq 2 \left(\frac{M_h - m_h}{M_h + m_h} \right)^j \|p_h - p_0\|_{S_h}.$$

Hence, $p_j \to p_h$. From (2.4.1), we conclude that $w_{j+1} \to w_h$ as well. □

The first equation in (2.4.1) entitles $\|q_{j+1}\|_h$ as an efficient and uniform iteration error estimator for Algorithm 2.4.1. Furthermore, Theorem 2.4.4 says that the iteration error satisfies

$$\|p_j - p_h\|_{\tilde{Q}} \leq \frac{1}{k_1 \, m_h^2} \|q_{j+1}\|_h.$$

Thus, if the discretization error order of convergence is known, e.g., $\|p - p_h\| = \mathcal{O}(h^\alpha)$, and an estimate for m_h is also available, the iteration error can match the discretization error by imposing the stopping criteria

$$\|q_{j+1}\|_h \leq c\, m_h^2 h^\alpha.$$

Remark 2.4.5. *The SPLS discretization method for solving the general mixed problem* (2.0.1) *is related with the Bramble-Pasciak least squares approach presented in [34]. The Bramble-Pasciak least squares discretization can be formulated as: Find* $p_h \in \mathcal{M}_h$ *such that*

$$b(A_h^{-1}B_h^* q_h, p_h) = \langle f_h, A_h^{-1}B_h^* q_h\rangle = b(A_h^{-1}f_h, q_h) \quad \text{for all } q_h \in \mathcal{M}_h. \tag{2.4.6}$$

We note that the above problem is equivalent to the Schur complement problem (2.2.5)*. While we arrive at essentially the same normal equation for solving* (2.2.3)*, the saddle point approach is more direct. Also, in [34], to iteratively solve* (2.4.6) *bases for both the test and trial spaces are needed. In contrast, we solve the coupled saddle point problem* (2.2.4) *using Algorithm 2.4.1, which avoids the need of a basis for the trial space.*

2.5 A Special Case

In some applications, such as in the case of the time-harmonic Maxwell equations presented in Chapter 6, the space Q coincides with $\tilde{Q}$, and the theory presented in this chapter reduces to the framework introduced in [20]. In regards to this case, the following general estimate was proved in [20].

Theorem 2.5.1. *Let* $b : V \times Q \to \mathbb{R}$ *satisfy* (2.1.1) *and* (2.1.2) *and assume that* $F \in V^*$ *is given and satisfies* (2.1.3)*. Assume that* p *is the solution of* (2.0.1) *and* $V_h \subset V$, $\mathcal{M}_h \subset Q$ *are chosen such that the discrete* inf − sup *condition* (2.2.1) *holds. If* (w_h, p_h) *is the solution of* (2.2.4)*, then the following error estimate holds:*

$$\frac{1}{M}|u_h|_V \leq \|p - p_h\|_Q \leq \frac{M}{m_h} \inf_{q_h \in \mathcal{M}_h} \|p - q_h\|_Q. \tag{2.5.1}$$

Chapter 3

SADDLE POINT LEAST SQUARES PRECONDITIONING

In this chapter, we present a general way to precondition the discrete saddle point reformulation arising from the SPLS method of Chapter 2. When solving (2.2.4) by Algorithm 2.4.1, the process requires the exact inversion of the operator A_h associated with the inner product $a(\cdot,\cdot)$ on V_h. This can be seen by writing Step 1 and Step UCG1 in operator form as

$$w_1 = A_h^{-1}(f_h - B_h^* p_0), \quad \text{and} \quad h_j = A_h^{-1} B^* d_j,$$

respectively. A key observation of the SPLS framework is that the p_h component to the solution of problem (2.2.4) is independent of the choice of inner product on V_h. On this premise, the goal of this chapter is to construct an equivalent form on V_h, which will be denoted $\tilde{a}(\cdot,\cdot)$, where the action of the operator $\tilde{A}_h^{-1}$ associated with this new form is assumed to be fast and easy to implement, such as a suitable preconditioner for A_h. With this form, we can introduce a preconditioned discrete saddle point problem: Find $(w_h, p_h) \in V_h \times \mathcal{M}_h$ such that

$$\begin{aligned} \tilde{a}(w_h, v_h) \quad + \quad b(v_h, p_h) \quad &= \langle f_h, v_h \rangle \qquad \text{for all } v_h \in V_h, \\ b(w_h, q_h) \qquad\qquad\quad &= 0 \qquad\qquad \text{for all } q_h \in \mathcal{M}_h. \end{aligned}$$

The main benefit of this approach to preconditioning is that we can analyze the above saddle point problem in the same way as (2.2.4). This chapter is published in [13, 14].

This chapter is organized as follows. Section 3.1 describes the general theory for constructing the preconditioned form $\tilde{a}(\cdot,\cdot)$ and the resulting saddle point system. In Section 3.2, the no projection and projection type trial spaces are revisited and the stability is discussed when using the new form $\tilde{a}(\cdot,\cdot)$. An Uzawa type iterative solver is

outlined in Section 3.3 to solve the preconditioned saddle point system. In Section 3.4, an analogous result to Theorem 2.5.1 is proved. Section 3.5 provides an application of the approach that utilizes the theory of multilevel preconditioners.

3.1 The General Preconditioning Technique

In this section, we develop a general preconditioning strategy to approximate the solution of (2.0.1) based on the saddle point reformulation (2.2.4) and elliptic preconditioning of the operator associated with the inner product on V_h. First, we consider an operator $P_h : V_h^* \to V_h$ that is equivalent to A_h^{-1} in the following sense. We assume P_h satisfies

$$\langle g, P_h f\rangle = \langle f, P_h g\rangle \quad \text{for all } f, g \in V_h^*, \tag{3.1.1}$$

and

$$m_1^2 |v_h|_V^2 \le a(P_h A_h v_h, v_h) \le m_2^2 |v_h|_V^2 \quad \text{for all } v_h \in V_h, \tag{3.1.2}$$

for positive constants m_1, m_2.

Remark 3.1.1. *Assuming that m_1^2, m_2^2 are the smallest and the largest eigenvalues of $P_h A_h$, respectively, inequality (3.1.2) gives us the condition number of $P_h A_h$ satisfies*

$$\kappa(P_h A_h) = \frac{m_2^2}{m_1^2}. \tag{3.1.3}$$

With the operator $P_h : V_h^* \to V_h$, we define the form $\tilde{a} : V_h \times V_h \to \mathbb{R}$ by

$$\tilde{a}(u_h, v_h) := a((P_h A_h)^{-1} u_h, v_h) \quad \text{for all } u_h, v_h \in V_h. \tag{3.1.4}$$

Proposition 3.1.2. *Under assumptions (3.1.1) and (3.1.2), the form $\tilde{a}(\cdot,\cdot)$ is symmetric and equivalent with $a(\cdot,\cdot)$ on $V_h \times V_h$.*

Proof. For symmetry, it suffices to prove that the operator $P_h A_h$ is symmetric with respect to the $a(\cdot,\cdot)$ inner product. This follows from the definition of the operator A_h and (3.1.1) as

$$a(P_h A_h u_h, v_h) = \langle A_h v_h, P_h A_h u_h\rangle = \langle A_h u_h, P_h A_h v_h\rangle = a(u_h, P_h A_h v_h),$$

for any $u_h, v_h \in V_h$. The equivalence follows from (3.1.2) and (3.1.4) as

$$\frac{1}{m_2^2}|v_h|_V^2 \leq \tilde{a}(v_h, v_h) \leq \frac{1}{m_1^2}|v_h|_V^2 \quad \text{for all } v_h \in V_h, \tag{3.1.5}$$

and the fact that $|v_h|_V^2 = a(v_h, v_h)$. □

By Proposition 3.1.2, $\tilde{a}(\cdot,\cdot)$ defines an equivalent inner product on V_h. We define

$$|v_h|_P := \tilde{a}(v_h, v_h)^{1/2},$$

to be the norm induced by the inner product $\tilde{a}(\cdot,\cdot)$. Also, with the $\tilde{a}(\cdot,\cdot)$ inner product we define the operator $\tilde{A}_h : V_h \to V_h^*$ by

$$\langle \tilde{A}_h u_h, v_h \rangle := \tilde{a}(u_h, v_h) \quad \text{for all } u_h, v_h \in V_h.$$

Using the definitions of $A_h, \tilde{A}_h$, and $\tilde{a}(\cdot,\cdot)$, we obtain

$$\langle \tilde{A}_h u_h, v_h \rangle = \tilde{a}(u_h, v_h) = a((P_h A_h)^{-1} u_h, v_h) = \langle A_h (P_h A_h)^{-1} u_h, v_h \rangle,$$

for any $u_h, v_h \in V_h$. This implies

$$\tilde{A}_h = A_h (P_h A_h)^{-1} = P_h^{-1}.$$

Hence, we can view $\tilde{a}(\cdot,\cdot)$ as a "preconditioned" version of the form $a(\cdot,\cdot)$.

Using the $\tilde{a}(\cdot,\cdot)$ inner product on V_h, we consider the discrete saddle point problem: Find $(w_h, p_h) \in V_h \times \mathcal{M}_h$ such that

$$\begin{array}{lcll} \tilde{a}(w_h, v_h) & + \; b(v_h, p_h) & = \langle f_h, v_h \rangle & \text{for all } v_h \in V_h, \\ b(w_h, q_h) & & = 0 & \text{for all } q_h \in \mathcal{M}_h. \end{array} \tag{3.1.6}$$

We call problem (3.1.6) the preconditioned saddle point least squares formulation of (2.0.1). Using that $V_h \subset V$ and $\mathcal{M}_h \subset \tilde{Q}$ satisfy (2.2.1) and (2.2.2), we obtain

$$\tilde{m}_h := \inf_{p_h \in \mathcal{M}_h} \sup_{v_h \in V_h} \frac{b(v_h, p_h)}{|v_h|_P \, \|p_h\|_{\tilde{Q}}} \geq m_1 \, m_h > 0, \tag{3.1.7}$$

and

$$\tilde{M}_h := \sup_{p_h \in \mathcal{M}_h} \sup_{v_h \in V_h} \frac{b(v_h, p_h)}{|v_h|_P \, \|p_h\|_{\tilde{Q}}} \leq m_2 \, M_h \leq m_2 \, M. \tag{3.1.8}$$

Hence, (3.1.6) has a unique solution. The Schur complement associated with problem (3.1.6) is

$$\tilde{S}_h = B_h \tilde{A}_h^{-1} B_h^* = B_h P_h B_h^*.$$

Solving for p_h from (3.1.6), we obtain

$$\tilde{S}_h \, p_h = B_h (P_h B_h^*) \, p_h = B_h P_h f_h. \tag{3.1.9}$$

We call the component p_h of the solution (w_h, p_h) of (3.1.6) the preconditioned saddle point least squares approximation of the solution p of the original mixed prolem (2.0.1).

Remark 3.1.3. *In the approach taken in this chapter, we note that we are not preconditioning the full Schur Complement $S_h = B_h A_h^{-1} B_h^*$. We are essentially replacing the exact solve needed in Step 1 and Step UCG1 of Algorithm 2.4.1 with the action of a suitable preconditioner on the right side.*

3.2 The Discrete Spaces

In this section, we show for both the no projection trial space and the projection type trial space described in Sections 2.3.1 and 2.3.2 that an inf $-$ sup condition holds when V_h is equipped with the $\tilde{a}(\cdot,\cdot)$ inner product. Furthermore, we will show that similar approximability properties hold as in the previous chapter.

3.2.1 No Projection Trial Space

Recall that the no projection trial space is defined as

$$\mathcal{M}_h = BV_h.$$

From (2.3.1) and (3.1.5), we obtain

$$\inf_{p_h \in \mathcal{M}_h} \sup_{v_h \in V_h} \frac{b(v_h, p_h)}{|v_h|_P \, \|p_h\|_{\tilde{Q}}} \geq m_1 m_{h,0}.$$

Hence, an inf $-$ sup condition is satisfied. Also, since the space $V_{h,0}$ is independent of the norm on V_h, we have $V_{h,0} \subset V_h$. Furthermore, the approximability result from Section 2.3.1, namely

$$\|p - p_h\|_{\tilde{Q}} = \inf_{q_h \in \mathcal{M}_h} \|p - q_h\|_{\tilde{Q}},$$

is still valid.

3.2.2 Projection Type Trial Space

The projection type trial space is defined as

$$\mathcal{M}_h = R_h B V_h,$$

where $R_h : \tilde{Q} \to \tilde{\mathcal{M}}_h$ satisfies

$$(R_h p, q_h)_h = (p, q_h)_{\tilde{Q}} \quad \text{for all } q_h \in \tilde{M}_h,$$

and $\tilde{\mathcal{M}}_h$ is a finite dimensional subspace of $\tilde{Q}$. Using that $V_{h,0}$ is independent of the norm on V_h, if R_h satisfies (2.3.5) then Proposition 2.3.2 is still valid with an $\inf - \sup$ constant satisfying

$$\inf_{p_h \in \mathcal{M}_h} \sup_{v_h \in V_h} \frac{b(v_h, p_h)}{|v_h|_P \, \|p_h\|_h} \geq m_1 \tilde{c} \, m_{h,0}.$$

Furthermore, Proposition 2.3.3 still holds, and we have

$$\|p - p_h\|_{\tilde{Q}} \leq \left(1 + \frac{1}{\tilde{c} k_1}\right) \inf_{q_h \in \mathcal{M}_h} \|p - q_h\|_{\tilde{Q}}.$$

3.3 An Iterative Solver

We use a modified version of Algorithm 2.4.1 to solve (3.1.6) by replacing the form $a(\cdot, \cdot)$ by $\tilde{a}(\cdot, \cdot)$ in Step 1 and Step UCG1. With this modification, we obtain the following Uzawa Preconditioned Conjugate Gradient (UPCG) algorithm for mixed methods.

Algorithm 3.3.1. *(UPCG) Algorithm for Mixed Methods*

Step 1: Choose any $p_0 \in \mathcal{M}_h$. **Compute** $w_1 \in V_h$, $q_1, d_1 \in \mathcal{M}_h$ *by*

$$\begin{aligned} \tilde{a}(w_1, v_h) &= \langle f_h, v_h \rangle - b(v_h, p_0) \qquad \text{for all } v_h \in V_h \\ q_1 &= B_h w_1, \qquad\qquad\qquad\qquad d_1 := q_1. \end{aligned}$$

Step 2: For $j = 1, 2, \ldots,$ **compute** $h_j, \alpha_j, p_j, w_{j+1}, q_{j+1}, \beta_j, d_{j+1}$ *by*

$$\textbf{(PCG1)} \qquad \tilde{a}(v_h, h_j) = -\, b(v_h, d_j) \qquad \text{for all } v_h \in V_h$$

$$\textbf{(PCG}\alpha\textbf{)} \qquad \alpha_j = -\,\frac{(q_j, q_j)_h}{b(h_j, q_j)}$$

$$\textbf{(PCG2)} \qquad p_j = p_{j-1} + \alpha_j\, d_j$$

$$\textbf{(PCG3)} \qquad w_{j+1} = w_j + \alpha_j\, h_j$$

$$\textbf{(PCG4)} \qquad q_{j+1} = B_h w_{j+1},$$

$$\textbf{(PCG}\beta\textbf{)} \qquad \beta_j = \frac{(q_{j+1}, q_{j+1})_h}{(q_j, q_j)_h}$$

$$\textbf{(PCG6)} \qquad d_{j+1} = q_{j+1} + \beta_j d_j.$$

Remark 3.3.2. *Instead of taking the initial iterate* $p_0 = 0$ *in Step 1 of the UPCG algorithm for each level of refinement of a suitable mesh for the problem to be solved, if the refinements are nested we can take an approach in which* $p_0 = 0$ *on the coarsest mesh, but for all successive refinements* p_0 *is chosen as the extension of the final iterate from the previous level. This approach will be referred to as the UPCG Cascadic algorithm.*

Note that in operator form the first equation of Step 1 and Step PCG1 are

$$w_1 = P_h(f_h - B_h^* p_0), \text{ and } h_j = -P_h(B_h^* d_j).$$

For any preconditioner P_h and trial space $\mathcal{M}_h$ that is not defined via a global projection, the actions of P_h, B_h, and B_h^* do not involve inversion processes. Section 3.5 gives an example for the case when P_h is given as the BPX preconditioner [35]. Similar to Remark 2.4.3, we have the following.

Remark 3.3.3. *Algorithm 3.3.1 recovers in particular the steps of the Conjugate Gradient algorithm for solving problem* (3.1.9)*. Due to the discrete* inf − sup *condition* (2.2.1) *and the assumptions* (3.1.1) *and* (3.1.2) *on the preconditioner* P_h*, the Schur complement* $\tilde{S}_h$ *is a symmetric, positive definite operator. Consequently, the conjugate*

iterations p_j converge to the solution p_h of (3.1.9), and the rate of convergence depends on the condition number of $\tilde{S}_h$, which is

$$\kappa(\tilde{S}_h) = \frac{\tilde{M}_h^2}{\tilde{m}_h^2}.$$

The following result is analogous to Theorem 2.4.4.

Theorem 3.3.4. *If (w_h, p_h) is the discrete solution of (3.1.6) and (w_j, p_{j-1}) is the j^{th} iteration for Algorithm 3.3.1, then $(w_j, p_{j-1}) \to (w_h, p_h)$ and*

$$\begin{aligned} \frac{1}{M^2}\frac{1}{m_2^2}\|q_j\|_h \le \|p_{j-1} - p_h\|_h \le \frac{1}{m_h^2}\frac{1}{m_1^2}\|q_j\|_h, \\ \frac{m_h}{M^2}\frac{m_1^2}{m_2^2}\|q_j\|_h \le |w_j - w_h|_V \le \frac{M}{m_h^2}\frac{m_2^2}{m_1^2}\|q_j\|_h. \end{aligned} \tag{3.3.1}$$

Proof. By induction over j, we obtain

$$\tilde{a}(w_j, v_h) + b(v_h, p_{j-1}) = \langle f, v_h\rangle \quad \text{for all } v_h \in V_h.$$

Combining this with the first equation of (3.1.6) gives us

$$\tilde{a}(w_j - w_h, v_h) = b(v_h, p_h - p_{j-1}) \quad \text{for all } v_h \in V_h. \tag{3.3.2}$$

Note that $\sigma(\tilde{S}_h) \subset [\tilde{m}_h^2, \tilde{M}_h^2]$. Hence,

$$\tilde{m}_h\|q_h\|_h = (\tilde{S}_h q_h, q_h)_h^{1/2} \le \tilde{M}_h\|q_h\|_h \quad \text{for all } q_h \in \mathcal{M}_h. \tag{3.3.3}$$

By substituting $v_h = \tilde{A}_h^{-1}B_h^*(p_h - p_{j-1})$ into (3.3.2),

$$|w_j - w_h|_P^2 = (\tilde{S}_h(p_h - p_{j-1}), p_h - p_{j-1})_h = \|p_h - p_{j-1}\|_{\tilde{S}_h}^2.$$

The above equality, (3.1.5), and (3.3.3) give us

$$m_1\tilde{m}_h\|p_h - p_{j-1}\|_h \le |w_j - w_h|_V \le m_2\tilde{M}_h\|p_h - p_{j-1}\|_h. \tag{3.3.4}$$

From Step PCG4, the second equation of (3.1.6), and (3.3.2), we obtain

$$q_j = B_h w_j = B_h(w_j - w_h) = \tilde{S}_h(p_h - p_{j-1}).$$

Thus,

$$\tilde{m}_h^2 \|p_h - p_{j-1}\|_h \leq \|\tilde{S}_h(p_h - p_{j-1})\|_h = \|q_j\|_h \leq \tilde{M}_h^2 \|p_h - p_{j-1}\|_h. \tag{3.3.5}$$

The inequalities (3.3.1) follow from (3.3.4), (3.3.5), and the fact that $\tilde{m}_h \geq m_h m_1$ and $\tilde{M}_h \leq M m_2$. From Remark 3.3.3 and the standard estimate for the convergence rate of the conjugate gradient algorithm, we obtain

$$\|p_h - p_j\|_{\tilde{S}_h} \leq 2 \left(\frac{\tilde{M}_h - \tilde{m}_h}{\tilde{M}_h + \tilde{m}_h} \right)^j \|p_h - p_0\|_{\tilde{S}_h}. \tag{3.3.6}$$

Hence, $p_j \to p_h$. From (3.3.1), we conclude that $w_j \to w_h$ as well. □

The following estimates are a direct consequence of (3.1.3), (3.1.7), (3.1.8), (3.3.6), and the formula $\kappa(\tilde{S}_h) = \frac{\tilde{M}_h^2}{\tilde{m}_h^2}$.

Proposition 3.3.5. *The condition number of the Schur complement* $\tilde{S}_h = B_h P_h B_h^*$ *satisfies*

$$\kappa(\tilde{S}_h) \leq \frac{M_h^2}{m_h^2} \frac{m_2^2}{m_1^2} = \kappa(S_h) \cdot \kappa(P_h A_h). \tag{3.3.7}$$

Consequently, the convergence rate ρ_h *for* $\|p_j - p_h\|_{\tilde{S}_h}$ *in* (3.3.6) *satisfies*

$$\rho_h \leq \frac{\frac{M_h}{m_h}\frac{m_2}{m_1} - 1}{\frac{M_h}{m_h}\frac{m_2}{m_1} + 1}.$$

The first equation in (3.3.1) entitles $\|q_{j+1}\|_h$ as an efficient and uniform iteration error estimator for Algorithm 3.3.1. Furthermore, Theorem 3.3.4 says that the iteration error satisfies

$$\|p_j - p_h\|_{\tilde{Q}} \leq \frac{1}{k_1 m_1^2} \frac{1}{m_h^2} \|q_{j+1}\|_h.$$

Thus, if the discretization error order of convergence is known, e.g., $\|p - p_h\| = \mathcal{O}(h^\alpha)$, and an estimate for m_h is also available, the iteration error can match the discretization error by imposing the stopping criteria

$$\|q_{j+1}\|_h \leq c\, m_h^2 h^\alpha,$$

just as in the non-preconditioned approach of Chapter 2.

Remark 3.3.6. *The preconditioned SPLS discretization method for solving the general mixed problem (2.0.1) is related with the Bramble-Pasciak preconditioned least squares approach presented in [34]. In our notation, the Bramble-Pasciak least squares discretization can be formulated as: Find $p_h \in \mathcal{M}_h$ such that*

$$b(A_h^{-1}B_h^*q_h, p_h) = \langle f_h, A_h^{-1}B_h^*q_h\rangle = b(A_h^{-1}f_h, q_h) \quad \text{for all } q_h \in \mathcal{M}_h.$$

With a suitable preconditioner P_h replacing A_h^{-1}, the problem becomes: Find $p_h \in \mathcal{M}_h$ such that

$$b(P_hB_h^*q_h, p_h) = b(P_hf_h, q_h) \quad \text{for all } q_h \in \mathcal{M}_h. \tag{3.3.8}$$

We note that (3.3.8) is equivalent to the Schur complement problem (3.1.9). In [34], to iteratively solve (3.3.8) bases for both the test and trial spaces are needed. In contrast, we solve the coupled preconditioned saddle point problem (3.1.6) using Algorithm 3.3.1, which avoids the need of a basis for the trial space.

3.4 A Special Case

In this section, we prove an analogous result to Theorem 2.5.1 for the case when Q coincides with $\tilde{Q}$ and V_h is equipped with the $|\cdot|_P$ norm. The proof, as in the case of Theorem 2.5.1, is based on the Xu-Zikatanov argument, see [93].

Theorem 3.4.1. *Let $b : V \times Q \to \mathbb{R}$ satisfy (2.1.1) and (2.1.2) and assume $F \in V^*$ is given and satisfies (2.1.3). Assume that $V_h \subset V$, $\mathcal{M}_h \subset Q$ are chosen such that the discrete* inf − sup *condition (2.2.1) holds. If p is the solution of (2.0.1) and (w_h, p_h) is the solution of (3.1.6), then the following error estimate holds:*

$$\frac{1}{M}\frac{1}{m_2^2}|w_h|_V \le \|p - p_h\|_Q \le \frac{M}{m_h}\frac{m_2}{m_1}\inf_{q_h\in\mathcal{M}_h}\|p - q_h\|_Q.$$

Proof. Define the operator $T_h : Q \to Q$ by $T_hp = p_h$. Note that T_h is linear and idempotent. To show the latter, consider the problem: Find $(w_h^*, p_h^*) \in V_h \times \mathcal{M}_h$ such that

$$\begin{aligned} \tilde{a}(w_h^*, v_h) \quad + \quad b(v_h, p_h^*) \quad &= b(v_h, p_h) \qquad \text{for all } v_h \in V_h, \\ b(w_h^*, q_h) \qquad\qquad\quad &= 0 \qquad\qquad \text{for all } q_h \in \mathcal{M}_h. \end{aligned} \tag{3.4.1}$$

Since b satisfies (2.2.1), the inf $-$ sup condition (3.1.7) is satisfied. Thus, problem (3.4.1) has a unique solution. Since $(w_h^*, p_h^*) = (0, p_h)$ solves the problem, we conclude $T_h p_h = p_h$ which gives us $T_h^2 = T_h$. From Kato [61] and Xu and Zikatanov [93], this implies

$$\|I - T_h\|_{\mathcal{L}(Q,Q)} = \|T_h\|_{\mathcal{L}(Q,Q)}.$$

Using the above equality, we obtain

$$\|p - p_h\|_Q = \|(I - T_h)p\|_Q = \|(I - T_h)(p - q_h)\|_Q \leq \|T_h\| \, \|p - q_h\|_Q, \tag{3.4.2}$$

for an arbitrary $q_h \in \mathcal{M}_h$.

We now estimate $\|T_h\|$. First, define $\tilde{V}_{h,0}^{\perp}$ to be the orthogonal complement of $V_{h,0}$ with respect to the $\tilde{a}(\cdot,\cdot)$ inner product. From the first equation of (3.1.6) and the fact p solves (2.0.1), we obtain

$$b(v_h, p_h) = b(v_h, p) - \tilde{a}(w_h, v_h). \tag{3.4.3}$$

Also, (3.1.8) holds since $b(\cdot,\cdot)$ satisfies (2.1.1). Hence, from (3.1.7), (3.1.8), and (3.4.3) we obtain

$$\begin{aligned}
\|T_h p\|_Q &\leq \frac{1}{m_h \, m_1} \sup_{v_h \in V_h} \frac{b(v_h, T_h p)}{|v_h|_P} \\
&= \frac{1}{m_h \, m_1} \sup_{v_h \in \tilde{V}_{h,0}^{\perp}} \frac{b(v_h, p_h)}{|v_h|_P} \\
&= \frac{1}{m_h \, m_1} \sup_{v_h \in \tilde{V}_{h,0}^{\perp}} \frac{b(v_h, p) - \tilde{a}(w_h, v_h)}{|v_h|_P} \\
&\leq \frac{M m_2}{m_h m_1} \|p\|_Q.
\end{aligned} \tag{3.4.4}$$

The right inequality now follows from (3.4.2) and (3.4.4). For the left inequality, note that

$$|u_h|_P = \sup_{v_h \in V_h} \frac{\tilde{a}(w_h, v_h)}{|v_h|_P} = \sup_{v_h \in V_h} \frac{b(v_h, p - p_h)}{|v_h|_P} \leq M \, m_2 \|p - p_h\|_Q,$$

and

$$|w_h|_V \leq m_2 |w_h|_P.$$

□

3.5 Application of a Multilevel Preconditioner

In order to illustrate the applicability of the theory presented thus far, we outline a choice for P_h based on multilevel preconditioning techniques. More specifically, we consider the case when P_h is given by the BPX preconditioner with a diagonal scaling, see [35, 94]. Assume that we have a nested sequence of approximation spaces

$$V_1 \subset V_2 \subset \cdots \subset V_J = V_h,$$

and let $\{\phi_1^k, \phi_2^k, \ldots, \phi_{n_k}^k\}$ denote the nodal basis for V_k. For $f_h \in V_h^*$, the action of P_h is given by

$$P_h f_h = \sum_{k=1}^{J} \sum_{i=1}^{n_k} \frac{\langle f_h, \phi_i^k \rangle}{a(\phi_i^k, \phi_i^k)} \phi_i^k. \tag{3.5.1}$$

It is known that for $V = H_0^1(\Omega)$ and a nested sequence $\{V_k\}$ of piecewise linear functions that, under standard mesh uniformity conditions, P_h is a preconditioner for A_h satisfying (3.1.1) and (3.1.2), see [35, 60, 90, 91, 94]. Similarly, we can consider the standard BPX preconditioner in which

$$P_h f_h = \sum_{k=1}^{J} h_k^{2-d} \sum_{i=1}^{n_k} \langle f_h, \phi_i^k \rangle \phi_i^k, \tag{3.5.2}$$

for $f_h \in V_h^*$. In the above expression, h_k refers to the mesh size for each level of refinement k. For the remainder of this thesis, we will refer to the preconditioner in (3.5.2) as the standard BPX preconditioner and the preconditioner described in (3.5.1) as the scaled BPX preconditioner.

In the case of the scaled BPX preconditioner, the first equation in Step 1 of Algorithm 3.3.1 becomes

$$w_1 = P_h(f_h - B_h^* p_0) = \sum_{k=1}^{J} \sum_{i=1}^{n_k} \frac{\langle f_h, \phi_i^k \rangle - b(\phi_i^k, p_0)}{a(\phi_i^k, \phi_i^k)} \phi_i^k. \tag{3.5.3}$$

Furthermore, the iterates for h_j in Step PCG1 are given by

$$h_j = -\sum_{k=1}^{J} \sum_{i=1}^{n_k} \frac{b(\phi_i^k, d_j)}{a(\phi_i^k, \phi_i^k)} \phi_i^k. \tag{3.5.4}$$

This implies

$$b(h_j, q_j) = -\sum_{k=1}^{J}\sum_{i=1}^{n_k} \frac{b(\phi_i^k, d_j)b(\phi_i^k, q_j)}{a(\phi_i^k, \phi_i^k)}, \tag{3.5.5}$$

in Step PCGα. Thus, the implementation of Algorithm 3.3.1 does not involve matrix inversion. Similarly, no matrix inversion is required with the standard BPX preconditioner as well.

3.5.1 Implementation of the BPX Preconditioners

In this section, we will discuss the implementation of the BPX preconditioners described in Section 3.5. We note that while any elliptic preconditioner can be used for P_h, such as multigrid [36], we choose to show the details of the BPX preconditioners to emphasize the simplicity of implementation when dealing with mixed methods preconditioning.

First, we define $T_k \in \mathbb{R}^{n_J \times n_k}$ as the matrix representation of the nodal basis $\{\phi_1^k, \phi_2^k, \ldots, \phi_{n_k}^k\}$ for V_k in terms of the nodal basis $\{\phi_1^J, \phi_2^J, \ldots, \phi_{n_J}^J\}$ for V_J. In [91], it was shown that the implementation of the standard/scaled BPX preconditioners depends entirely on the transformation matrices T_k. More specifically, the algebraic form of the BPX preconditioners is given by

$$P_h = \sum_{k=1}^{J} T_k R_k T_k^t, \tag{3.5.6}$$

where $R_k = h_k^{2-d} I$ corresponds to the standard BPX preconditioner and $R_k = D_k^{-1}$, where

$$D_k = \text{diag}\left(a(\phi_1^k, \phi_1^k), a(\phi_2^k, \phi_2^k), \ldots, a(\phi_{n_k}^k, \phi_{n_k}^k)\right),$$

corresponds to the scaled BPX preconditioner. The following algorithm gives the action of P_h in (3.5.6) on a given vector in $\alpha \in \mathbb{R}^{n_J}$.

Algorithm 3.5.1. *(BPX)*

Set $\alpha_J = \alpha$;

for $k = J - 1 : 1$

$\alpha_k = (T_k^{k+1})^T \alpha_{k+1}$;

end

Set $\beta_1 = R_1\alpha_1;$

for $k = 2 : J$

$\beta_k = R_k\alpha_k + T_{k-1}^k\beta_{k-1};$

end

$P_h\alpha = \beta_J$

3.5.2 Computational Complexity of the Proposed UPCG Algorithm

In this section, we discuss the complexity of Algorithm 3.3.1 when P_h is given as the BPX preconditioner described in Section 3.5.1. From Step 1 or Step 2 of Algorithm 3.3.1, we observe at each step that the number of operations depends on the complexity of P_h and the dimension of the test space V_h, say $n = n_h$. A preconditioner P_h is of optimal complexity if $\mathcal{O}(n)$ operations are needed to compute its action, where n is the dimension of V_h. Preconditioners such as BPX, and even multigrid, are of optimal complexity. For the BPX preconditioner defined in (3.5.1) (or (3.5.2)), this is because for each k (using a standard refinement strategy, such as in 2D splitting each triangle into four smaller triangles) we have $n_k = \mathcal{O}(\alpha^k)$ for some $\alpha > 1$ depending on the dimension of the domain. Here, $n = n_J$ is the dimension of V_h. In this case, the action of P_h needs

$$\mathcal{O}\left(\sum_{k=1}^{J} n_k\right) = \mathcal{O}\left(\sum_{k=1}^{J} \alpha^k\right) = \mathcal{O}(\alpha^J) = \mathcal{O}(n),$$

operations. Using that the action of B_h is the action of a differential operator (most often of first order) on a finite element function in V_h, we can conclude from formulas (3.5.3), (3.5.4), and (3.5.5) that the rest of the operations in Step 1 or Step 2 of Algorithm 3.3.1 sum up to at most $\mathcal{O}(n)$ operations.

Regarding the global complexity and optimality of the algorithm, it is known that if the condition number of the symmetric, positive definite operator $\tilde{S}_h$, defined in (3.1.9), is independent of h then the number of iterations of the UPCG algorithm is bounded independent of h. Thus, if P_h is an optimal complexity preconditioner which is also a uniform preconditioner, i.e., the constants m_1, m_2 in (3.1.2) are independent of

h, and the discrete $\inf - \sup$ constant m_h of (2.2.1) is independent of h, then Algorithm 3.3.1 is optimal. That is, to achieve a certain accuracy, it needs a number of operations that is proportional to the dimension of the space V_h.

Chapter 4

SPLS FOR SECOND ORDER ELLIPTIC INTERFACE PROBLEMS

To illustrate the SPLS discretization and preconditioning techniques described thus far, we will apply the method to the second order elliptic problem

$$\begin{cases} -\text{div}(A\nabla u) &= f \quad \text{in} \quad \Omega, \\ u &= 0 \quad \text{on } \partial\Omega, \end{cases} \tag{4.0.1}$$

where A is a symmetric matrix whose entries are discontinuous, with possibly large jumps, across an interface. These interface problems have applications in a variety of different fields. In material science, they arise in the study and design of composite materials built from essentially different components, see [6, 23, 55, 64]. In fluid dynamics, they model several layers of fluids with different viscosities or diffusion through heterogeneous porous media [26, 53]. In addition, the elliptic interface problem is used to model stationary heat conduction problems with a conduction coefficient that is discontinuous across a smooth internal interface [58], as well as in biological systems [62].

In the SPLS approach for problem (4.0.1), we will directly target the flux $A\nabla u$, which, in practice, is a more important physical quantity than the solution itself. At the discrete level, the projection type trial space of Section 2.3.2 will be the primary focus, and it falls into the noncomforming setting. One benefit of this type of trial space is that we obtain a higher order of approximation for the flux compared with standard finite element techniques using piecewise linear functions. To this end, the SPLS approach using the projection type trial space is related to Gradient Recovery, a widely used and effective post-processing technique, see [1, 24, 45, 56, 57, 79, 87, 95, 96, 97]. The benefit of the SPLS approach is that we can approximate the flux well without the

need of post-processing and higher order convergence is obtained through the iterative process itself. This chapter is published in [13, 14, 15].

Throughout this chapter, $L^2(\Omega)$ will denote the space of square integrable functions with the inner product

$$(u, v) = \int_\Omega uv,$$

and corresponding norm

$$\|u\| = \left(\int_\Omega |u|^2\right)^{1/2} = (u, u)^{1/2}.$$

We will also denote by $(\cdot, \cdot)$ and $\|\cdot\|$ the inner product and norm of the vector-valued product space $L^2(\Omega)^d$. We further define the Sobolev space $H_0^1(\Omega)$ as the closure of $\mathcal{C}_0^\infty(\Omega)$, the space of smooth compactly supported functions in Ω, with respect to the norm

$$\|u\|_{H^1(\Omega)} := (\|u\|^2 + \|\nabla u\|^2)^{1/2}.$$

This chapter is organized as follows. Section 4.1 describes how problem (4.0.1) fits into the SPLS framework. The discretization and choices of discrete trial spaces are outlined in Section 4.2. Section 4.3 discusses the stability of the proposed discrete spaces using a piecewise linear test space. Lastly, Section 4.4 presents numerical results, with and without preconditioning, to show the performance of the SPLS method and the benefits of the projection type of trial space.

4.1 SPLS for a Second Order Elliptic Interface Problem

Let $\Omega \subset \mathbb{R}^d$ be a bounded polygonal domain with $\{\Omega_j\}_{j=1}^N$ a partition of Ω and $\mathbf{n}_j$ be the outward unit normal vector to $\partial\Omega_j$. Define $\Gamma_{km} := \partial\Omega_k \cap \partial\Omega_m$ to be the interface between Ω_k and Ω_m for $1 \le k < m \le N$. Given $f \in L^2(\Omega)$, we consider the problem of finding $u \in H_0^1(\Omega)$ such that

$$-\mathrm{div}(A\nabla u) = f \quad \text{in } \Omega, \tag{4.1.1}$$

with the continuity of the co-normal derivative condition

$$[\![A\nabla u \cdot \mathbf{n}]\!]_{\Gamma_{km}} = (A_k \nabla u_k \cdot \mathbf{n}_k + A_m \nabla u_m \cdot \mathbf{n}_m)\big|_{\Gamma_{km}} = 0 \quad \text{for all } k < m.$$

We assume the matrix A is symmetric and satisfies

$$a_{min}|\boldsymbol{\xi}|_e^2 \leq \langle A(x)\boldsymbol{\xi}, \boldsymbol{\xi}\rangle_e \leq a_{max}|\boldsymbol{\xi}|_e^2 \quad \text{for all } x \in \Omega, \boldsymbol{\xi} \in \mathbb{R}^d, \tag{4.1.2}$$

for positive constants $a_{min} \leq a_{max}$. In the above, $\langle\cdot,\cdot\rangle_e$ and $|\cdot|_e$ denote the standard Euclidean inner product and norm for vectors in $\mathbb{R}^d$, respectively.

Remark 4.1.1. *While the theory in this chapter will be focused on the case when the entries of the matrix A are discontinuous, we note that the theory can be adapted to the case when the entries of A are continuous functions. We discuss this further in Section 4.4.5.*

A standard variational formulation for (4.1.1) is: Find $u \in H_0^1(\Omega)$ such that

$$(A\nabla u, \nabla v) = (f, v) \quad \text{for all } v \in H_0^1(\Omega). \tag{4.1.3}$$

Changing the variable of interest to the flux $\mathbf{p} := A\nabla u$, we rewrite the above formulation as: Find $\mathbf{p} = A\nabla u$, with $u \in H_0^1(\Omega)$, such that

$$(\mathbf{p}, \nabla v) = (f, v) \quad \text{for all } v \in H_0^1(\Omega). \tag{4.1.4}$$

To fit (4.1.4) into the abstract formulation (2.0.1), we let $V := H_0^1(\Omega)$, $\tilde{Q} := L^2(\Omega)^d$, $Q := A\nabla V$, and define $b : V \times \tilde{Q} \to \mathbb{R}$ by

$$b(v, \mathbf{q}) := (\mathbf{q}, \nabla v) \quad \text{for all } v \in V, \mathbf{q} \in \tilde{Q}.$$

We also define $F \in V^*$ by

$$\langle F, v\rangle := (f, v) \quad \text{for all } v \in V.$$

We consider the weighted inner product

$$a(u, v) := (A\nabla u, \nabla v) \quad \text{for all } u, v \in V,$$

on V. On $\tilde{Q}$, we define the weighted inner product

$$(\mathbf{p}, \mathbf{q})_{\tilde{Q}} := (\mathbf{p}, A^{-1}\mathbf{q}) \quad \text{for all } \mathbf{p}, \mathbf{q} \in \tilde{Q}.$$

In this setting, the operator $B : V \to \tilde{Q}$ is given by

$$Bv = A\nabla v \quad \text{for all } v \in V.$$

Hence,

$$V_0 = Ker(B) = \{v \in V | Bv = 0\} = \{v \in H_0^1(\Omega) | A\nabla v = 0\} = \{0\},$$

and the compatibility condition (2.1.3) is automatically satisfied. Using the Cauchy-Schwarz inequality, the continuity constant satisfies

$$\begin{aligned} M &= \sup_{\mathbf{q}\in\tilde{Q}} \sup_{v\in V} \frac{b(v,\mathbf{q})}{|v|_V \, \|\mathbf{q}\|_{\tilde{Q}}} \\ &= \sup_{\mathbf{q}\in\tilde{Q}} \sup_{v\in V} \frac{(\mathbf{q}, \nabla v)}{|v|_V \, \|\mathbf{q}\|_{\tilde{Q}}} \\ &= \sup_{\mathbf{q}\in\tilde{Q}} \sup_{v\in V} \frac{(\mathbf{q}, A\nabla v)_{\tilde{Q}}}{|v|_V \, \|\mathbf{q}\|_{\tilde{Q}}} \\ &\leq \sup_{v\in V} \frac{\|A\nabla v\|_{\tilde{Q}}}{(A\nabla v, \nabla v)^{1/2}} \\ &= 1. \end{aligned} \tag{4.1.5}$$

Also, the $\inf - \sup$ constant satisfies

$$\begin{aligned} m &= \inf_{\mathbf{q}=A\nabla u\in Q} \sup_{v\in V} \frac{b(v,\mathbf{q})}{|v|_V \, \|\mathbf{q}\|_{\tilde{Q}}} \\ &= \inf_{u\in V} \sup_{v\in V} \frac{(A\nabla u, \nabla v)}{(A\nabla u, \nabla u)^{1/2} \, (A\nabla v, \nabla v)^{1/2}} \\ &\geq 1. \end{aligned} \tag{4.1.6}$$

Consequently, the variational problem (4.1.4) is well-posed and suitable for SPLS discretization and preconditioning.

Remark 4.1.2. *Defining $a(u, v) := (\nabla u, \nabla v)$ is also a suitable choice for the inner product on V as the p component of the solution to the saddle point reformulation is independent of the norm on V. The choice does, however, have an effect on the number of iterations of Algorithms 2.4.1 and 3.3.1. A more thorough discussion of this will be given in Section 4.4.2.*

4.2 SPLS Discretization for Second Order Elliptic Interface Problems

At the discrete level, we take $V_h \subset V = H_0^1(\Omega)$ to be the space of continuous piecewise polynomials of degree k with respect to the *interface-fitted* triangular mesh $\mathcal{T}_h$ for the test space. Several choices for the discrete trial space are now discussed.

4.2.1 No Projection Trial Space

Following Section 2.3.1, we consider the case when the trial space $\mathcal{M}_h$ is given by

$$\mathcal{M}_h := BV_h = A\nabla V_h,$$

equipped with the inner product from $\tilde{Q}$. By a similar argument used to show (4.1.6), we obtain

$$m_h := \inf_{\mathbf{q}_h = A\nabla u_h \in \mathcal{M}_h} \sup_{v_h \in V_h} \frac{b(v_h, \mathbf{q}_h)}{|v_h|_V \, \|\mathbf{q}_h\|_{\tilde{Q}}} \geq 1. \tag{4.2.1}$$

Thus, we do have stability in this case. The discrete mixed variational formulation is: Find $\mathbf{p}_h = A\nabla u_h$, with $u_h \in V_h$, such that

$$(\mathbf{p}_h, \nabla v_h) = (A\nabla u_h, \nabla v_h) = (f, v_h) \quad \text{for all } v_h \in V_h.$$

The discrete saddle point reformulation, using the $a(\cdot,\cdot)$ inner product, is: Find $(w_h, \mathbf{p}_h = A\nabla u_h)$ such that

$$\begin{aligned} (A\nabla w_h, \nabla v_h) \;+\; (\mathbf{p}_h, \nabla v_h) &= (f, v_h) \qquad \text{for all } v_h \in V_h, \\ A\nabla w_h &= \mathbf{0}. \end{aligned}$$

4.2.2 Projection Type Trial Space

First, we define $\tilde{\mathcal{M}}_h \subset \tilde{Q} = L^2(\Omega)^d$ to be

$$\tilde{\mathcal{M}}_h := \bigoplus_{i=1}^{N} AM_{h,0}|_{\Omega_i},$$

where N is the number of subdomains and where each component of $M_{h,0}|_{\Omega_i}$ consists of continuous piecewise polynomials of degree k with respect to the mesh $\mathcal{T}_{h,i} := \mathcal{T}_h|_{\Omega_i}$ with no restrictions on the boundary. Two different choices for the projection type

trial space, based on the inner product chosen for the space $\tilde{\mathcal{M}}_h$, are given. The first is outlined in this section. The second is outlined in Section 4.3.1.

For the first type of projection trial space, we equip $\tilde{\mathcal{M}}_h$ with the inner product

$$(A\tilde{\mathbf{q}}_h, A\tilde{\mathbf{p}}_h)_h = \sum_{i=1}^{N} (A\tilde{\mathbf{q}}_h, A\tilde{\mathbf{p}}_h)_{\tilde{Q},\Omega_i} \quad \text{for all } A\tilde{\mathbf{q}}_h, A\tilde{\mathbf{p}}_h \in \tilde{\mathcal{M}}_h.$$

Here, $(\cdot,\cdot)_{\tilde{Q},\Omega_i}$ is the inner product on $\tilde{Q}$ restricted to the subdomain Ω_i. Using the definition of R_h given in (2.3.3), we conclude that $R_h p$ is the orthogonal projection of $\mathbf{p}$ onto $\tilde{\mathcal{M}}_h$ with respect to the $(\cdot,\cdot)_{\tilde{Q}}$ inner product. In turn, this implies $R_h p|_{\Omega_j}$ is the orthogonal projection onto $\tilde{\mathcal{M}}_h|_{\Omega_j} = AM_{h,0}|_{\Omega_j}$ with respect to the $(\cdot,\cdot)_{\tilde{Q},\Omega_j}$ inner product. We define the trial space as

$$\mathcal{M}_h := R_h^{\text{orth}} A\nabla V_h.$$

Remark 4.2.1. *In general, $\mathcal{M}_h$ constructed in this way is not contained in Q. For simplicity, we will consider the case when $A = I$ in 3D. For any $v_h \in V_h$, the vector field $\mathbf{q}_h = R_h^{\text{orth}} \nabla v_h \in L^2(\Omega)^3$ can be decomposed as*

$$\mathbf{q}_h = \nabla u + \boldsymbol{\varphi},$$

for some $u \in H_0^1(\Omega)$ and $\boldsymbol{\varphi} \in L^2(\Omega)^3$ such that $\operatorname{div}(\boldsymbol{\varphi}) = 0$ [63, Theorem 4.23]. Taking v_h to be a nodal basis function, we can verify numerically that $\mathbf{q}_h = R_h^{\text{orth}} \nabla v_h$ is not curl free. Hence, $\mathbf{q}_h \notin Q = \nabla H_0^1(\Omega)$.

The discrete mixed variational formulation in this case is: Find $\mathbf{p}_h = R_h^{\text{orth}} A\nabla u_h$, with $u_h \in V_h$, such that

$$(\mathbf{p}_h, \nabla v_h) = (R_h^{\text{orth}} A\nabla u_h, \nabla v_h) = (f, v_h) \quad \text{for all } v_h \in V_h.$$

The discrete saddle point reformulation in this case is: Find $(w_h, \mathbf{p}_h = R_h^{\text{orth}} A\nabla u_h)$ such that

$$\begin{aligned} (A\nabla w_h, \nabla v_h) \;+\; (\mathbf{p}_h, \nabla v_h) &= (f, v_h) \qquad \text{for all } v_h \in V_h, \\ R_h^{\text{orth}} A\nabla w_h &= \mathbf{0}. \end{aligned} \tag{4.2.2}$$

4.3 Piecewise Linear Test Space

In this section, we discuss the stability for the family of spaces $\{(V_h, \mathcal{M}_h)\}$, where $\mathcal{M}_h$ is as outlined in Section 4.2.2, for the case when the matrix A is diagonal and has constant coefficients. For simplicity, we assume $\Omega \subset \mathbb{R}^2$ is a polygonal domain separated into two subdomains by a smooth interface $\Gamma \subset \Omega$. The results can easily be extended to N subdomains as well as polyhedral domains in $\mathbb{R}^3$. We also assume that the triangular mesh $\mathcal{T}_h$ is locally quasi-uniform. In what follows, the index $i = 1, 2$ will refer to the corresponding subdomain of Ω. Let $\{z_{1,i}, \ldots, z_{N_i,i}\}$ be the set of all nodes of $\mathcal{T}_{h,i}$ and assume all triangles adjacent to $z_{j,i}$ are of regular shape and their area is of order $h_{j,i}^2$. In this notation, the mesh size of $\mathcal{T}_h = \mathcal{T}_{h,1} \cup \mathcal{T}_{h,2}$ is

$$h := \max\{h_{1,1}, h_{2,1}, \ldots, h_{N_1,1}, h_{1,2}, h_{2,2}, \ldots, h_{N_2,2}\}.$$

We take V_h to be the space consisting of piecewise linear polynomials with respect to $\mathcal{T}_h$ vanishing on the boundary of Ω. Hence, each component of $M_{h,0}|_{\Omega_i}$ consists of continuous linear piecewise polynomials with respect to the mesh $\mathcal{T}_{h,i}$. Let $\{\mathbf{\Phi}_1^i, ..., \mathbf{\Phi}_{2N_i}^i\}$ denote a nodal basis for $M_{h,0}|_{\Omega_i}$ and assume that $\mathbf{\Phi}_j^i = (\phi_j^i, 0)^T$ and $\mathbf{\Phi}_{N_i+j}^i = (0, \phi_j^i)^T$ for $j = 1, \ldots, N_i$. Here, $\{\phi_1^i, \ldots, \phi_{N_i}^i\}$ denotes the nodal basis for the space of continuous piecewise linear polynomials with respect to $\mathcal{T}_{h,i}$. With this notation, we note that $\{A\mathbf{\Phi}_j^1\}_{j=1}^{2N_1} \cup \{A\mathbf{\Phi}_j^2\}_{j=1}^{2N_2}$ is a basis for $\tilde{\mathcal{M}}_h$. We define M_{A_i} to be the Gram matrix of the set $\{A\mathbf{\Phi}_j^i\}_{j=1}^{2N_i}$ with respect to the $(\cdot,\cdot)_{\tilde{Q}}$ inner product and $H_i := \operatorname{diag}\left(h_{1,i}^2, h_{2,i}^2, \ldots, h_{N_i,i}^2\right)$. Lastly, we let

$$D_i = \left[\begin{array}{c|c} a_{11}H_i & \\ \hline & a_{22}H_i \end{array}\right],$$

where a_{11}, a_{22} are the entries of the matrix A.

Lemma 4.3.1. *Under the assumptions of Section 4.3, we have that for $i = 1, 2$*

$$\langle M_{A_i}\boldsymbol{\gamma}, \boldsymbol{\gamma}\rangle_e \leq c\,\langle D_i\boldsymbol{\gamma}, \boldsymbol{\gamma}\rangle_e \quad \text{for all } \boldsymbol{\gamma} \in \mathbb{R}^{2N_i}. \tag{4.3.1}$$

Consequently,

$$\langle M_{A_i}^{-1}\boldsymbol{\gamma}, \boldsymbol{\gamma}\rangle_e \geq c\langle D_i^{-1}\boldsymbol{\gamma}, \boldsymbol{\gamma}\rangle_e \quad \text{for all } \boldsymbol{\gamma} \in \mathbb{R}^{2N_i}, \tag{4.3.2}$$

where c is independent of h, a_{11}, and a_{22}.

Proof. We will prove the result when $i = 1$. The case when $i = 2$ is identical. Let $\boldsymbol{\gamma} \in \mathbb{R}^{2N_1}$ and define $\mathbf{q}_h := \sum_{j=1}^{2N_1} \boldsymbol{\gamma}_j \boldsymbol{\Phi}_j^1$. Note that

$$\langle M_{A_1}\gamma, \gamma\rangle_e = (A\mathbf{q}_h, \mathbf{q}_h) = \|A\mathbf{q}_h\|_{\tilde{Q}}^2 = \sum_{\tau\in\mathcal{T}_h} \|A\mathbf{q}_h\|_{\tau,\tilde{Q}}^2. \tag{4.3.3}$$

If $\tau = [z_{1_\tau}, z_{2_\tau}, z_{3_\tau}]$, then

$$\mathbf{q}_h\big|_\tau = \begin{pmatrix} \sum_{j=1}^{3} \gamma_{j_\tau} \phi_{j_\tau}^1 \\ \sum_{j=1}^{3} \gamma_{(j+N_1)_\tau} \phi_{j_\tau}^1 \end{pmatrix}.$$

Hence,

$$\|A\mathbf{q}_h\|_{\tau,\tilde{Q}}^2 \le c\,|\tau| \left(a_{11} \sum_{j=1}^{3} \gamma_{j_\tau}^2 + a_{22} \sum_{j=1}^{3} \gamma_{(j+N_1)_\tau}^2 \right). \tag{4.3.4}$$

Using (4.3.3), (4.3.4), and the fact that each coefficient γ_k can repeat at most three times, we obtain

$$\langle M_{A_1}\boldsymbol{\gamma}, \boldsymbol{\gamma}\rangle_e \le c \left(a_{11} \sum_{j=1}^{N_1} h_{j,1}^2 \gamma_j^2 + a_{22} \sum_{j=1}^{N_1} h_{j,1}^2 \gamma_{j+N_1}^2 \right) = c\,\langle D_1\boldsymbol{\gamma}, \boldsymbol{\gamma}\rangle_e .$$

The estimate (4.3.2) follows from (4.3.1). □

We now show that (2.3.5) is satisfied for the operator R_h^{orth} defined in Section 4.2.2.

Lemma 4.3.2. *Under the assumptions of Section 4.3, there exists a constant c, independent of h, a_{11}, and a_{22}, such that*

$$\|R_h^{\text{orth}} A\nabla v_h\|_h \ge c\,\|A\nabla v_h\|_{\tilde{Q}} \quad \text{for all } v_h \in V_h. \tag{4.3.5}$$

Proof. First, note that $\{A\boldsymbol{\Phi}_1^1, \ldots, A\boldsymbol{\Phi}_{2N_1}^1\}$ and $\{A\boldsymbol{\Phi}_1^2, \ldots, A\boldsymbol{\Phi}_{2N_2}^2\}$ are nodal bases for $\tilde{\mathcal{M}}_h|_{\Omega_1}$ and $\tilde{\mathcal{M}}_h|_{\Omega_2}$, respectively. Define $v_h^i := v_h|_{\Omega_i}$ for $v_h \in V_h$. For a fixed $A\nabla v_h$ with $v_h \in V_h$, we define the dual vectors $\mathbf{G}_h^1 \in \mathbb{R}^{2N_1}, \mathbf{G}_h^2 \in \mathbb{R}^{2N_2}$ by

$$(G_h^1)_i := (A\nabla v_h^1, A\boldsymbol{\Phi}_i^1)_{\tilde{Q}} = (A\nabla v_h^1, \boldsymbol{\Phi}_i^1)\ i = 1, ..., 2N_1,$$

$$(G_h^2)_i := (A\nabla v_h^2, A\boldsymbol{\Phi}_i^2)_{\tilde{Q}} = (A\nabla v_h^2, \boldsymbol{\Phi}_i^2)\ i = 1, ..., 2N_2,$$

and let

$$R_h^{\text{orth}} A\nabla v_h = \begin{cases} \sum_{i=1}^{2N_1} \alpha_i A\boldsymbol{\Phi}_i^1 & \text{in } \Omega_1, \\ \\ \sum_{i=1}^{2N_2} \beta_i A\boldsymbol{\Phi}_i^2 & \text{in } \Omega_2. \end{cases}$$

Thus, $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_{2N_1})^T$ and $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_{2N_2})^T$ are solutions to

$$M_{A_1}\, \boldsymbol{\alpha} = \mathbf{G}_h^1, \qquad \text{and} \qquad M_{A_2}\, \boldsymbol{\beta} = \mathbf{G}_h^2,$$

respectively. Using (4.3.2), we obtain

$$\begin{aligned} \|R_h^{\text{orth}} A\nabla v_h\|_h^2 &= \sum_{i,j=1}^{2N_1} \alpha_i\, \alpha_j \left(A\boldsymbol{\Phi}_i^1, \boldsymbol{\Phi}_j^1\right) + \sum_{i,j=1}^{2N_2} \beta_i\, \beta_j \left(A\boldsymbol{\Phi}_i^2, \boldsymbol{\Phi}_j^2\right) \\ &= \left\langle M_{A_1}^{-1}\mathbf{G}_h^1, \mathbf{G}_h^1\right\rangle_e + \left\langle M_{A_2}^{-1}\mathbf{G}_h^2, \mathbf{G}_h^2\right\rangle_e \\ &\geq c_1 \left\langle D_1^{-1}\mathbf{G}_h^1, \mathbf{G}_h^1\right\rangle_e + c_2 \left\langle D_2^{-1}\mathbf{G}_h^2, \mathbf{G}_h^2\right\rangle_e. \end{aligned}$$

We recall by definition of H_1, H_2 that we have $h_{i,1} = h_{i+N_1,1}$ for $i = 1, \ldots, N_1$ and $h_{i,2} = h_{i+N_2,2}$ for $i = 1, \ldots, N_2$. Hence,

$$\begin{aligned} \left\langle D_1^{-1}\mathbf{G}_h^1, \mathbf{G}_h^1\right\rangle_e &= \sum_{i=1}^{N_1} h_{i,1}^{-2} \left[a_{11} \left(\frac{\partial v_h^1}{\partial x}, \phi_i^1 \right)^2 + a_{22} \left(\frac{\partial v_h^1}{\partial y}, \phi_i^1 \right)^2 \right] \\ &= \sum_{i=1}^{N_1} \sum_{\tau \subset supp(\phi_i^1)} h_{i,1}^{-2} (1, \phi_i^1)_\tau^2 \left[a_{11} \left| \frac{\partial v_h^1}{\partial x} \right|_\tau^2 + a_{22} \left| \frac{\partial v_h^1}{\partial y} \right|_\tau^2 \right] \\ &\geq c_1 \|A\nabla v_h^1\|_{\Omega_1,\tilde{Q}}^2. \end{aligned}$$

Similarly, we can show

$$\left\langle D_2^{-1}\mathbf{G}_h^2, \mathbf{G}_h^2\right\rangle_e \geq c_2 \|A\nabla v_h^2\|_{\Omega_2,\tilde{Q}}^2.$$

Thus,

$$\|R_h^{\text{orth}} A\nabla v_h\|_h^2 \geq c \left(\|A\nabla v_h^1\|_{\Omega_1,\tilde{Q}}^2 + \|A\nabla v_h^2\|_{\Omega_2,\tilde{Q}}^2 \right) = c\|A\nabla v_h\|_{\tilde{Q}}^2,$$

as desired. □

As a consequence of Lemma 4.3.2, equation (4.2.1), and Proposition 2.3.2, we obtain the following result.

Theorem 4.3.3. *Let $\Omega \subset \mathbb{R}^2$ be a polygonal domain and $\{T_h\}$ be a family of locally quasi-uniform meshes for Ω. For each h, let V_h be the space of continuous linear functions with respect to the mesh $\{\mathcal{T}_h\}$ that vanish on $\partial\Omega$ and $\mathcal{M}_h = R_h^{\text{orth}} B V_h$. Then the family of spaces $\{(V_h, \mathcal{M}_h)\}$ is stable.*

4.3.1 Second Type of Projection Trial Space

For simplicity, we present the second type of projection trial space for the case when $N = 1$ (no interface). Using the same space $\tilde{\mathcal{M}}_h$ as defined in Section 4.2.2, we will consider an inner product on $\tilde{\mathcal{M}}_h$ related with lumping the mass matrix. More specifically, using the set $\{\mathbf{\Phi}_i\}$ as described in Section 4.3, we define the following inner product:

$$(A\mathbf{\Phi}_i, A\mathbf{\Phi}_j)_h := \delta_{ij}(1, A\mathbf{\Phi}_i).$$

Note that

$$\left(\sum_i \frac{(\mathbf{p}, A\mathbf{\Phi}_i)_{\tilde{Q}}}{(1, A\mathbf{\Phi}_i)} A\mathbf{\Phi}_i, A\mathbf{\Phi}_j \right)_h = (\mathbf{p}, A\mathbf{\Phi}_j)_{\tilde{Q}} \quad \text{for all } A\mathbf{\Phi}_j \in \tilde{\mathcal{M}}_h.$$

This implies $R_h : \tilde{Q} \to \tilde{\mathcal{M}}_h$ is given by

$$R_h \mathbf{p} = \sum_i \frac{(\mathbf{p}, A\mathbf{\Phi}_i)_{\tilde{Q}}}{(1, A\mathbf{\Phi}_i)} A\mathbf{\Phi}_i = \sum_i \frac{(\mathbf{p}, \mathbf{\Phi}_i)}{(1, A\mathbf{\Phi}_i)} A\mathbf{\Phi}_i,$$

from (2.3.3). For the application to the elliptic interface problem, we simply apply R_h locally on each subdomain with respect to the $(\cdot, \cdot)_h$ inner product as in Section 4.2.2. We define the trial space in this case as

$$\mathcal{M}_h := R_h^{\text{lump}} A \nabla V_h.$$

Remark 4.3.4. *Similar to the justification given in Remark 4.2.1, we can show in general that $\mathcal{M}_h \not\subset Q = \nabla H_0^1(\Omega)$.*

The problem to be solved using this projection type trial space is identical to (4.2.2). The following lemma is analogous to 4.3.2.

Lemma 4.3.5. *Under the assumptions of Section 4.3, there exists a constant c, independent of h, a_{11}, and a_{22}, such that*

$$\|R_h^{\text{lump}} A\nabla v_h\|_h \geq c\,\|A\nabla v_h\|_{\tilde{Q}} \quad \text{for all } v_h \in V_h. \tag{4.3.6}$$

Proof. Using the same notation from the proof of Lemma 4.3.2, we obtain

$$\begin{aligned}
\|R_h^{\text{lump}} A\nabla v_h\|_h^2 &= \sum_{j=1}^{2N_1} \frac{(A\nabla v_h^1, A\boldsymbol{\Phi}_j^1)_{\tilde{Q}}^2}{(1, A\boldsymbol{\Phi}_j^1)^2}(1, A\boldsymbol{\Phi}_j^1) + \sum_{j=1}^{2N_2} \frac{(A\nabla v_h^2, A\boldsymbol{\Phi}_j^2)_{\tilde{Q}}^2}{(1, A\boldsymbol{\Phi}_j^2)^2}(1, A\boldsymbol{\Phi}_j^2) \\
&= \sum_{j=1}^{2N_1} \frac{(A\nabla v_h^1, \boldsymbol{\Phi}_j^1)^2}{(1, A\boldsymbol{\Phi}_j^1)} + \sum_{j=1}^{2N_2} \frac{(A\nabla v_h^2, \boldsymbol{\Phi}_j^2)^2}{(1, A\boldsymbol{\Phi}_j^2)} \\
&\geq c_1 \left\langle D_1^{-1}\mathbf{G}_h^1, \mathbf{G}_h^1 \right\rangle_e + c_2 \left\langle D_2^{-1}\mathbf{G}_h^2, \mathbf{G}_h^2 \right\rangle_e,
\end{aligned}$$

where

$$\begin{aligned}
(G_h^1)_i &:= (A\nabla v_h^1, A\boldsymbol{\Phi}_i^1)_{\tilde{Q}} = (A\nabla v_h^1, \boldsymbol{\Phi}_i^1)\ i = 1, ..., 2N_1, \\
(G_h^2)_i &:= (A\nabla v_h^2, A\boldsymbol{\Phi}_i^2)_{\tilde{Q}} = (A\nabla v_h^2, \boldsymbol{\Phi}_i^2)\ i = 1, ..., 2N_2.
\end{aligned}$$

From the same techniques to estimate $\left\langle D_1^{-1}\mathbf{G}_h^1, \mathbf{G}_h^1 \right\rangle_e$ and $\left\langle D_2^{-1}\mathbf{G}_h^2, \mathbf{G}_h^2 \right\rangle_e$ as in the proof of Lemma 4.3.2, the result follows. □

As a consequence of Lemma 4.3.5, we have the following result.

Theorem 4.3.6. *Let $\Omega \subset \mathbb{R}^2$ be a polygonal domain and $\{T_h\}$ be a family of locally quasi-uniform meshes for Ω. For each h, let V_h be the space of continuous linear functions with respect to the mesh $\{\mathcal{T}_h\}$ that vanish on $\partial\Omega$ and $\mathcal{M}_h = R_h^{\text{lump}} B V_h$. Then the family of spaces $\{(V_h, \mathcal{M}_h)\}$ is stable.*

4.4 Numerical Results

In this section, we present results from applying the SPLS discretization on second order elliptic PDE of the form (4.1.1) with and without preconditioning. For

all examples, Ω will be a bounded polygonal or polyhedral domain and the test space $V_h \subset H_0^1(\Omega)$ will be the space of continuous piecewise linear polynomials with respect to the quasi-uniform, or locally quasi-uniform, meshes $\mathcal{T}_h$. We consider all types of trial spaces presented in this chapter: the no projection type presented in Section 4.2.1 and the projection types presented in Sections 4.2.2 and 4.3.1. In the case of no preconditioning, we use Algorithm 2.4.1. In the case of preconditioning, for Step 1 and Step PCG1 of Algorithm 3.3.1, we consider the cases when P_h is given by the scaled BPX preconditioner, described in Section 3.5, and a V-cycle multigrid preconditioner with a Gauss-Seidel smoother. For a thorough analysis of these preconditioners for elliptic interface problems, we refer to [36, 90, 92].

Based on the first inequalities of (2.4.1) and (3.3.1), we used a stopping criterion of

$$\|\mathbf{q}_j\|_h \leq c_0 h^2,$$

on each level for the case of convex domains and uniform refinement. This is because the maximum possible order for the discretization error $\|A\nabla u - R_h^{\text{orth}} A\nabla u_h\|_{\tilde{Q}}$, using the projection onto continuous piecewise linear polynomials, would be order two. In the two dimensional case with non-uniform refinement, we used a stopping criterion of

$$\|\mathbf{q}_j\|_h \leq c_0 N_{dof}^{-2},$$

on each level where N_{dof} is the number of degrees of freedom. In practice, we notice that we cannot achieve order two. This could be because on each subdomain we approximate, in a weighted L^2 norm, a possibly smooth component of the flux, but use subspaces of continuous piecewise linear functions as approximation spaces component wise.

Remark 4.4.1. *We note that while the flux $A\nabla u$ is targetted for the SPLS discretization of the interface problem, the primal variable u can be approximated along the process simultaneously by separately storing the u_j part of the iterates $\mathbf{p}_j = A\nabla u_j$, $\mathbf{p}_j = R_h^{\text{orth}}(A\nabla u_j)$, or $\mathbf{p}_j = R_h^{\text{lump}}(A\nabla u_j)$, which serve as a proxy $\mathbf{p}_j$, and follow the*

updates for $\mathbf{p}_j$ *as in the algorithm. However, for the piecewise linear approximation we consider here, we do not observe a higher order of approximation for the primal variable.*

In all examples presented, the constant c will denote the size of the jump in the coefficients of the matrix A. The level of mesh refinement will be denoted by k. Furthermore, error $= \|A\nabla u - \mathbf{p}_h\|_{\tilde{Q}}$ for all examples, where the SPLS solution $\mathbf{p}_h$ depends on the type of trial space used.

4.4.1 Example With Intersecting Interfaces

For the first example, we consider $\Omega = (0,1) \times (0,1)$ with the interface $\Gamma := \Omega \cap \{(x,y) \mid \ x = 1/2 \text{ or } y = 1/2\}$ as considered in [24]. The family of interface-fitted, locally quasi-uniform meshes $\{\mathcal{T}_h\}$ was obtained by a standard uniform refinement strategy starting with a uniform coarse mesh. We computed f such that for

$$A(x,y) = a(x,y)I_2, \text{ where } a(x,y) = \begin{cases} 1 & \text{if } (x,y) \in [0,1/2]^2 \cup [1/2,1]^2, \\ c & \text{if } (x,y) \in \Omega \setminus ([0,1/2]^2 \cup [1/2,1]^2), \end{cases}$$

the exact solution is

$$u(x,y) = a(x,y)^{-1} \sin(2\pi x) \sin(2\pi y).$$

Table 4.1 shows results for $c = 1/10, 1/100$, and $1/1000$ using SPLS discretization without preconditioning for both types of projection type trial spaces.

We observe higher order convergence for the flux for both types of projection trial spaces. Furthermore, the method is robust with respect to the jump in the coefficients of the matrix A. Table 4.2 shows results for the no projection trial space and both types of projection trial spaces using the scaled BPX preconditioner. As expected, the same order of convergence for the flux is observed in this case, along with a similar error, as the approximability properties of the trial spaces are independent of the norm on V_h. The same robustness properties are also observed. Table 4.3 shows results using the multigrid preconditioner with the no projection trial space as well as the

$\mathcal{M}_h = R_h^{\text{orth}} A\nabla V_h$

level k	$c = 1/10$			$c = 1/100$			$c = 1/1000$		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
1	5.177		1	15.686		1	49.383		1
2	1.258	2.041	4	3.812	2.041	4	12.001	2.041	4
3	0.339	1.893	7	1.026	1.893	8	3.231	1.893	10
4	0.097	1.868	11	0.281	1.868	13	0.885	1.868	16
5	0.025	1.877	17	0.076	1.880	22	0.240	1.880	28

$\mathcal{M}_h = R_h^{\text{lump}} A\nabla V_h$

level k	$c = 1/10$			$c = 1/100$			$c = 1/1000$		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
1	4.344		1	13.162		1	41.437		1
2	1.766	1.299	3	5.281	1.317	4	16.626	1.317	4
3	0.610	1.534	4	1.815	1.541	7	5.705	1.543	9
4	0.209	1.547	6	0.630	1.526	8	1.971	1.533	15
5	0.072	1.526	7	0.218	1.528	11	0.686	1.522	16

Table 4.1: Intersecting interface problem without preconditioning.

lump projection trial space. Compared with using the scaled BPX preconditioner, we observe similar error and order of convergence and see a decrease in iterations.

Remark 4.4.2. *We note that while combining the trial space $\mathcal{M}_h = R_h^{\text{orth}} A\nabla V$ with the scaled BPX preconditioner obtains the same order of convergence for the flux as in the case of no preconditioning, a drawback to this choice is having to invert local mass matrices in each iteration. When using the other types of trial spaces with preconditioning, the resulting versions of Algorithm 3.3.1 do not involve matrix inversion.*

4.4.2 Effects of Choosing a Different Inner Product on V

In reference to Remark 4.1.2, we demonstrate the benefit of choosing the weighted inner product on $V = H_0^1(\Omega)$ in comparison to the inner product $a(u, v) := (\nabla u, \nabla v)$. To illustrate this, we consider the interface problem of Section 4.4.1 using SPLS discretization without preconditioning for both types of projection type trial spaces. Table 4.4 collects the results using the different inner product for $c = 1/10, 1/100$, and $1/1000$.

$\mathcal{M}_h = A\nabla V_h$

level k	$c = 1/10$			$c = 1/100$			$c = 1/1000$		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
1	7.045		1	21.349		1	67.209		1
2	3.933	0.841	3	11.918	0.841	3	37.520	0.841	4
3	2.025	0.957	7	6.137	0.957	8	19.320	0.957	9
4	1.020	0.989	10	3.092	0.989	12	9.733	0.989	13
5	0.511	0.997	13	1.549	0.997	15	4.876	0.997	16

$\mathcal{M}_h = R_h^{\text{orth}} A\nabla V_h$

level k	$c = 1/10$			$c = 1/100$			$c = 1/1000$		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
1	5.177		1	15.686		1	49.383		1
2	1.258	2.041	4	3.812	2.041	4	12.001	2.041	4
3	0.339	1.893	10	1.026	1.893	12	3.231	1.893	13
4	0.093	1.868	24	0.281	1.868	26	0.885	1.868	31
5	0.025	1.877	48	0.076	1.880	59	0.240	1.880	66

$\mathcal{M}_h = R_h^{\text{lump}} A\nabla V_h$

level k	$c = 1/10$			$c = 1/100$			$c = 1/1000$		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
1	4.344		1	13.162		1	41.437		1
2	1.743	1.317	3	5.282	1.317	3	16.627	1.317	3
3	0.599	1.540	6	1.815	1.541	8	5.710	1.542	9
4	0.208	1.526	14	0.627	1.534	18	1.971	1.534	23
5	0.073	1.515	23	0.218	1.521	32	0.685	1.525	45

Table 4.2: Intersecting interface problem with scaled BPX preconditioner.

In comparison with Table 4.1, we see a significant increase in the number of iterations when this inner product is chosen. This is due to the fact that $\kappa(S_h)$, which is related to the convergence of Algorithm 2.4.1, is influenced by the size in the jump of coefficients. A similar behavior can be observed in the case of preconditioning as the factor $\kappa(S_h)$ appears in (3.3.7). Choosing the weighted inner product eliminates the influence of the jump in the coefficients from $\kappa(S_h)$ in all choices for the trial space as seen in estimates (4.1.5), (4.1.6), (4.3.5) and (4.3.6).

$\mathcal{M}_h = A\nabla V_h$

level k	$c = 1/10$			$c = 1/100$			$c = 1/1000$		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
1	7.045		1	21.349		1	67.209		1
2	3.933	0.841	2	11.918	0.841	2	37.520	0.841	2
3	2.025	0.957	2	6.137	0.957	3	19.320	0.957	3
4	1.020	0.989	3	3.092	0.989	3	9.733	0.989	4
5	0.511	0.997	4	1.549	0.997	4	4.876	0.997	4

$\mathcal{M}_h = R_h^{\text{lump}} A\nabla V_h$

level k	$c = 1/10$			$c = 1/100$			$c = 1/1000$		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
1	4.344		1	13.162		1	41.437		1
2	1.796	1.304	4	5.281	1.317	6	16.626	1.317	7
3	0.606	1.536	4	1.815	1.541	7	5.704	1.543	10
4	0.208	1.541	6	0.629	1.528	8	1.972	1.532	15
5	0.072	1.522	8	0.218	1.527	12	0.686	1.523	17

Table 4.3: Intersecting interface problem with multigrid preconditioner.

$\mathcal{M}_h = R_h^{\text{orth}} A\nabla V_h$

level k	$c = 1/10$			$c = 1/100$			$c = 1/1000$		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
1	5.177		4	15.686		4	49.383		4
2	1.261	2.037	10	3.947	1.990	12	15.827	1.641	11
3	0.339	1.895	16	1.070	1.883	27	3.607	2.133	29
4	0.097	1.802	17	0.307	1.803	33	0.985	1.873	63
5	0.027	1.849	22	0.086	1.832	44	0.295	1.738	76

$\mathcal{M}_h = R_h^{\text{lump}} A\nabla V_h$

level k	$c = 1/10$			$c = 1/100$			$c = 1/1000$		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
1	4.506		3	30.796		2	99.170		2
2	1.963	1.199	5	6.404	2.265	8	20.605	2.267	10
3	0.622	1.658	8	1.937	1.725	15	7.758	1.409	16
4	0.216	1.528	8	0.664	1.545	17	2.099	1.886	38
5	0.074	1.546	11	0.231	1.521	19	0.736	1.512	38

Table 4.4: Intersecting interface problem with inner product $(\nabla u_h, \nabla v_h)$.

4.4.3 Example With Gradient Singularity at the Origin

For the second example, we solved (4.1.1) for a problem where the gradient of the solution is singular at the origin, see [78]. The domain $\Omega = (-1,1)^2$ is decomposed as $\Omega_2 := \{(x,y) \in \Omega \,|\, 0 < \theta(x,y) < \pi/2\}$ and $\Omega_1 := \Omega \setminus \Omega_2$, where $\theta(x,y)$ is the angle in polar coordinates of the point (x,y). We computed f such that for

$$A(x,y) = a(x,y)I_2, \text{ where } a(x,y) = \begin{cases} 1 & \text{if } (x,y) \in \Omega_1, \\ c & \text{if } (x,y) \in \Omega_2, \end{cases}$$

the exact solution, given in polar coordinates, is $u(r,\theta) = r^\lambda(1-r)^2\mu(\theta)$ where

$$\mu(\theta) = \begin{cases} \cos(\lambda(\theta - \pi/4)) & \text{if } (x,y) \in \Omega_2, \\ b\cos(\lambda(\pi - |\theta - \pi/4|)) & \text{otherwise,} \end{cases}$$

and

$$\lambda = \frac{4}{\pi}\arctan\left(\sqrt{\frac{3+c}{1+3c}}\right), \qquad b = -c\frac{\sin\left(\lambda\frac{\pi}{4}\right)}{\sin\left(\lambda\frac{3\pi}{4}\right)}.$$

By using a similar standard uniform refinement strategy as in Section 4.4.1, Table 4.5 summarizes results using both types of projection trial spaces for $c = 10$ and $c = 100$. Using uniform meshes, we observe a convergence rate less than one.

To better handle the singularity of the gradient, a family of interface-fitted, locally quasi-uniform meshes $\{\mathcal{T}_h\}$ was obtained by a graded refinement strategy depending on a refinement parameter κ [18, 19]. The refinement is done by splitting each triangle into four smaller triangles. In particular, we divide every edge that contains the singular point (the origin in this case) under a fixed ratio κ such that the edge containing the singular point is κ times the other segment. In the case $\kappa = 1$, we recover the uniform refinement. Numerical results using graded meshes with $\kappa = 0.22$ are summarized in Table 4.6 for $c = 10$ and $c = 100$. Figure 4.1 depicts the mesh generated (at the final level of refinement) using the graded refinement strategy for $\kappa = 0.22$, as well as the x component of the computed flux, for the case of $c = 10$.

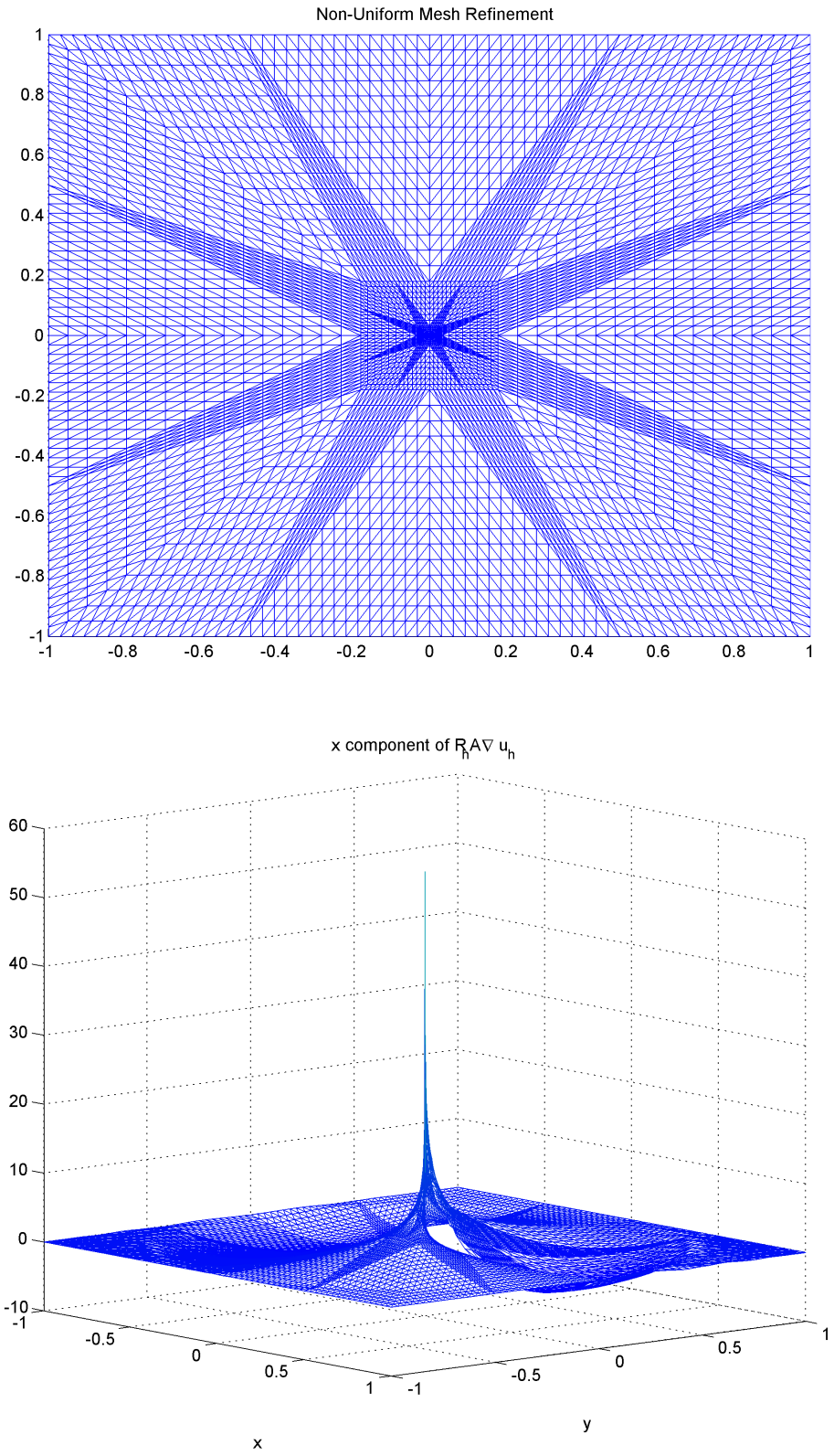

Figure 4.1: Mesh and x component of the computed flux for gradient singularity problem.

$\mathcal{M}_h = R_h^{\text{orth}} A \nabla V_h$

level k	$c = 10$			$c = 100$		
$h = 2^{-k}$	error	rate	it	error	rate	it
1	2.318		3	21.653		4
2	0.785	1.562	7	8.244	1.393	10
3	0.419	0.906	10	4.820	0.774	18
4	0.249	0.751	19	3.032	0.669	35
5	0.150	0.730	29	1.915	0.663	64

$\mathcal{M}_h = R_h^{\text{lump}} A \nabla V_h$

level k	$c = 10$			$c = 100$		
$h = 2^{-k}$	error	rate	it	error	rate	it
1	2.212		1	20.549		3
2	0.805	1.457	3	8.287	1.310	9
3	0.460	0.807	6	5.271	0.653	13
4	0.276	0.738	8	3.365	0.647	24
5	0.167	0.722	12	2.131	0.659	36

Table 4.5: Gradient singularity problem on uniform meshes.

$\mathcal{M}_h = R_h^{\text{orth}} A \nabla V_h$

level k	$c = 10$			$c = 100$		
	error	rate	it	error	rate	it
1	1.769		3	16.633		5
2	0.985	0.845	6	8.776	0.922	14
3	0.272	1.859	11	2.540	1.789	22
4	0.094	1.535	12	0.899	1.498	32
5	0.031	1.599	16	0.301	1.579	43

$\mathcal{M}_h = R_h^{\text{lump}} A \nabla V_h$

level k	$c = 10$			$c = 100$		
	error	rate	it	error	rate	it
1	1.873		1	17.459		4
2	0.991	0.918	3	8.852	0.980	10
3	0.321	1.628	5	2.971	1.575	14
4	0.121	1.406	6	1.153	1.366	15
5	0.044	1.460	8	0.429	1.425	21

Table 4.6: Gradient singularity problem on graded meshes.

As in the previous example, we observe higher order convergence for the flux for each type of projection trial space. In addition, the method is robust with respect to the jump in the coefficients.

4.4.4 Example of an Interface Problem in $3D$

For the third example, we consider $\Omega \subset \mathbb{R}^3$ the unit cube with interface $\Gamma := \Omega \cap \{(x, y, z) \,|\, x = 1/2\}$. We computed f such that for

$$A(x, y, z) = a(x, y, z)I_3, \text{ where } a(x, y, z) = \begin{cases} 1 & \text{if } x < \frac{1}{2}, \\ c & \text{if } x \geq \frac{1}{2}, \end{cases}$$

the exact solution is

$$u(x, y, z) = \begin{cases} c\, x(x - \frac{1}{2})y(y - 1)z(z - 1) & \text{if } x < \frac{1}{2}, \\ (x - \frac{1}{2})(x - 1)y(y - 1)z(1 - z) & \text{if } x \geq \frac{1}{2}. \end{cases}$$

Table 4.7 shows the results for $c = 100, 1000$, and 10000 for both types of projection type trial spaces. As in the $2D$ examples, we observe higher order convergence for the flux, and the method is robust with respect to the jump in the coefficients. Table 4.8 shows results for the no projection trial space and the lump projection trial space with the scaled BPX preconditioner. We observe a similar convergence rate and error for the flux as well as robustness with respect to the jump in the coefficients. Table 4.9 shows results using a multigrid preconditioner and the same types of trial spaces as in the scaled BPX preconditioner case. Compared with using the scaled BPX preconditioner, we obtain similar error and order of convergence and see a decrease in the number of iterations.

4.4.5 Flux Recovery for Highly Oscillatory Coefficients

In reference to Remark 4.1.1, we will apply the SPLS discretization to an example where the entries of A are smooth functions. In particular, we will illustrate the advantage of SPLS discretization on an example where the matrix A has highly oscillatory coefficients. In this chapter, we proved (4.3.5) and (4.3.6) for the case when

$\mathcal{M}_h = R_h^{\text{orth}} A \nabla V_h$

level k	$c = 100$			$c = 1000$			$c = 10000$		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
1	5.177		1	15.686		1	49.383		1
2	1.258	2.041	4	3.812	2.041	4	12.001	2.041	4
3	0.339	1.893	7	1.026	1.893	8	3.231	1.893	10
4	0.097	1.868	11	0.281	1.868	13	0.885	1.868	16
5	0.025	1.877	17	0.076	1.880	22	0.240	1.880	28

$\mathcal{M}_h = R_h^{\text{lump}} A \nabla V_h$

level k	$c = 100$			$c = 1000$			$c = 10000$		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
1	4.344		1	13.162		1	41.437		1
2	1.766	1.299	3	5.281	1.317	4	16.626	1.317	4
3	0.610	1.534	4	1.815	1.541	7	5.705	1.543	9
4	0.209	1.547	6	0.630	1.526	8	1.971	1.533	15
5	0.072	1.526	7	0.218	1.528	11	0.686	1.522	16

Table 4.7: 3D interface problem without preconditioning.

$\mathcal{M}_h = A \nabla V_h$

level k	$c = 100$			$c = 1000$			$c = 10000$		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
1	0.837		1	8.337		1	83.334		1
2	0.572	0.549	2	5.700	0.549	3	56.972	0.549	4
3	0.320	0.838	6	3.188	0.838	8	31.864	0.838	11
4	0.165	0.953	11	1.647	0.953	15	16.462	0.953	18
5	0.083	0.987	19	0.831	0.988	24	8.302	0.988	29

$\mathcal{M}_h = R_h^{\text{lump}} A \nabla V_h$

level k	$c = 100$			$c = 1000$			$c = 10000$		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
1	0.837		1	8.337		1	83.334		1
2	0.312	1.426	1	2.995	1.477	2	29.774	1.485	5
3	0.120	1.374	4	1.139	1.395	8	11.390	1.386	14
4	0.046	1.397	8	0.414	1.458	20	4.141	1.460	29
5	0.017	1.436	13	0.148	1.485	32	1.463	1.500	57

Table 4.8: 3D interface problem with scaled BPX preconditioner.

$\mathcal{M}_h = A\nabla V_h$

level k	$c = 100$			$c = 1000$			$c = 10000$		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
1	0.837		1	8.337		1	83.334		1
2	0.572	0.549	1	5.700	0.549	1	56.972	0.549	2
3	0.320	0.838	2	3.188	0.838	2	31.864	0.838	3
4	0.165	0.953	3	1.647	0.953	4	16.462	0.953	4
5	0.083	0.987	3	0.831	0.988	4	8.302	0.988	5

$\mathcal{M}_h = R_h^{\text{lump}} A\nabla V_h$

level k	$c = 100$			$c = 1000$			$c = 10000$		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
1	0.837		1	8.337		1	83.334		1
2	0.314	1.404	1	2.995	1.481	3	29.775	1.484	6
3	0.115	1.452	3	1.139	1.391	5	11.389	1.386	10
4	0.044	1.384	3	0.414	1.459	9	4.141	1.460	16
5	0.015	1.517	5	0.148	1.490	12	1.464	1.500	26

Table 4.9: 3D interface problem with multigrid preconditioner.

the matrix A is diagonal and has constant coefficients. This is an improvement upon the same estimates compared with the case where A has smooth variable coefficients. For this case, it was proved in [15] that

$$\|R_h A\nabla v_h\|_h \geq c\frac{a_{min}}{a_{max}} \|A\nabla v_h\|_{\tilde{Q}} \quad \text{for all } v_h \in V_h,$$

where c is independent of h, R_h can be taken as either R_h^{orth} or R_h^{lump}, and a_{min} and a_{max} are as defined in (4.1.2).

We solved (4.1.1) on $\Omega = (0,1) \times (0,1)$ with $A = a(x,y)I_2$, where

$$a(x,y) = \frac{1}{4 + P(\sin(2\pi x/\varepsilon) + \sin(2\pi y/\varepsilon))}.$$

We computed f such that the exact solution is given by

$$u(x,y) = \frac{\sqrt{4-P^2}}{2}(x^2+y^2)\exp\left(\frac{1}{x^3-x} + \frac{1}{y^3-y}\right).$$

This is a small modification of a similar example presented in [76]. Table 4.10 shows results for various values of ε using both types of projection type trial spaces. In all computations, we chose $P = 1.8$.

$\mathcal{M}_h = R_h^{\text{orth}} A\nabla V_h$

level k	$\varepsilon = 0.2$			$\varepsilon = 0.1$			$\varepsilon = 0.05$		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
4	5.32e-05	1.692	1	6.74e-05	2.530	1	3.43e-04	1.462	1
5	1.47e-05	1.856	1	2.20e-05	1.617	1	3.61e-05	3.250	1
6	3.90e-06	1.915	1	6.61e-06	1.732	1	9.93e-06	1.863	1
7	1.04e-06	1.909	1	1.79e-06	1.884	1	3.03e-06	1.713	1

$\mathcal{M}_h = R_h^{\text{lump}} A\nabla V_h$

level k	$\varepsilon = 0.2$			$\varepsilon = 0.1$			$\varepsilon = 0.05$		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
4	1.10e-04	1.442	1	1.23e-04	1.687	1	2.93e-04	1.546	1
5	3.45e-05	1.677	1	4.79e-05	1.369	1	5.25e-05	3.483	1
6	9.63e-06	1.843	1	1.57e-05	1.609	1	2.17e-05	1.276	1
7	2.59e-06	1.896	1	4.44e-06	1.822	1	7.30e-06	1.571	1

Table 4.10: Results for highly oscillatory coefficients example.

The numerical results show almost $O(h^2)$ order of approximation for the flux on meshes that are small enough to capture the high frequency of the coefficients due to the size of ε. The method is also robust with respect to the size of ε. Figure 4.2 shows the x component of $A\nabla u$ with the x component of the approximated flux from the SPLS method.

4.4.6 A Comparison With the Standard PCG Method

In this section, we compare the performance of the preconditioned SPLS method, using the no projection trial space, with directly applying the standard Preconditioned Conjugate Gradient method for the matrix equation arising from the variational form (4.1.3). We consider (4.1.1) with $\Omega = (0,1) \times (0,1)$, $A = I$, and where f is computed such that the exact solution is $u(x,y) = x(1-x)y(1-y)$. We apply the standard PCG method, UPCG (Algorithm 3.3.1), and UPCG cascadic algorithms with the choice for P_h the standard BPX preconditioner given in (3.5.2). Table 4.11 compares the performance of the UPCG algorithm, as well as the cascadic version, with the standard PCG algorithm. In the table, error $= \|\nabla u - \nabla u_h\|$. We see the performance of the

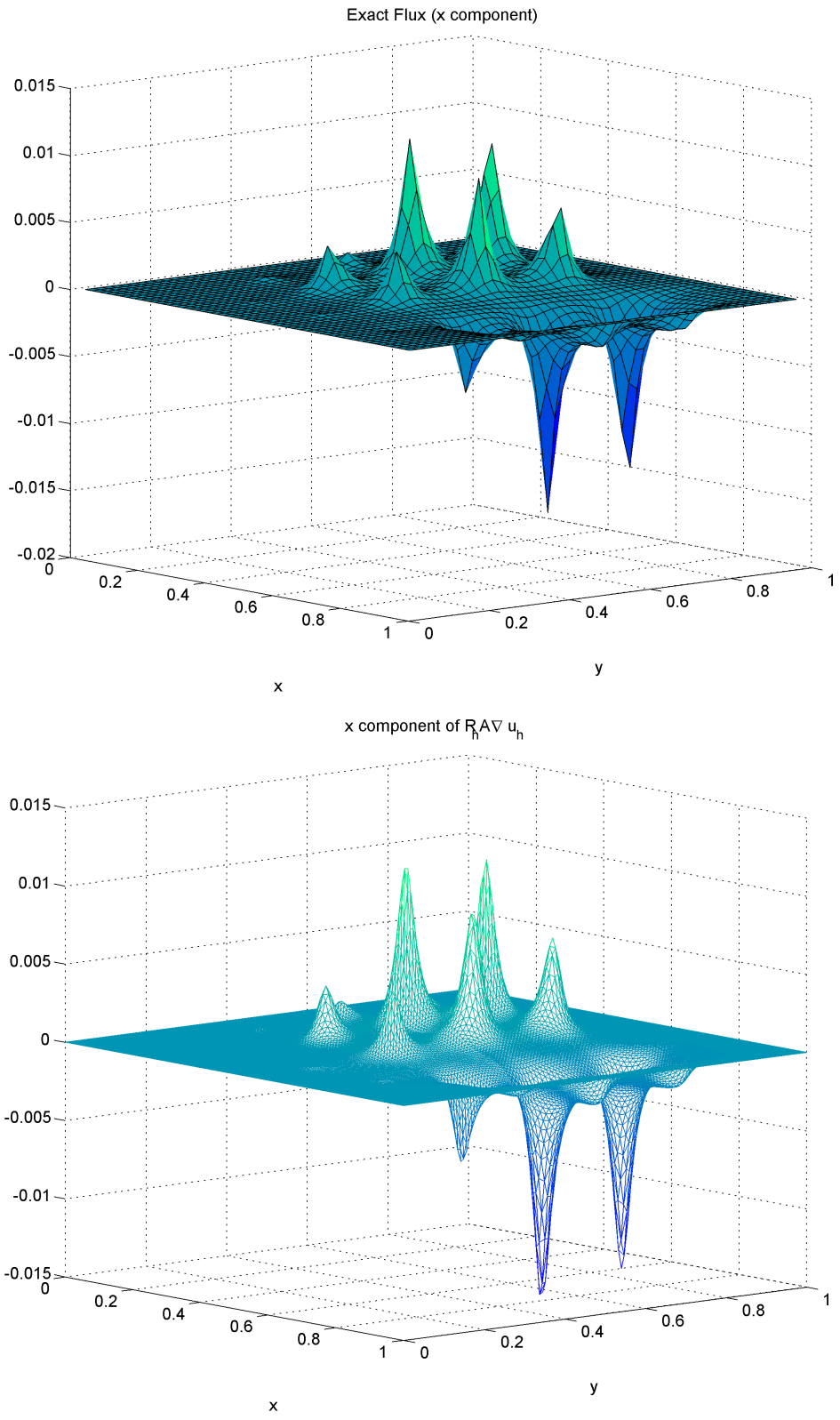

Figure 4.2: x component of the exact and computed flux for highly oscillatory coefficients example.

UPCG algorithm is comparable with standard PCG. In addition, there is a significant reduction in the number of iterations using the cascadic approach.

level k	PCG			UPCG			UPCG cascadic		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
1	0.045	0.000	1	0.045	0.000	2	0.045	0.000	2
2	0.024	0.903	5	0.024	0.896	4	0.025	0.837	2
3	0.012	0.974	7	0.012	0.944	5	0.013	0.952	3
4	0.006	0.991	8	0.006	1.016	7	0.007	0.974	3
5	0.003	0.996	9	0.003	0.988	8	0.003	0.994	3
6	0.001	1.002	11	0.001	1.012	10	0.002	1.002	3

Table 4.11: Comparison on unit square example.

For the next example, we consider a simple interface problem where $\Omega = (0,1) \times (0,1)$ with interface $\Gamma := \Omega \cap \{(x,y)|\ x = 1/2\}$. We computed f such that for

$$A(x,y) = a(x,y)I_2, \text{ where } a(x,y) = \begin{cases} \beta & \text{if } x \geq \frac{1}{2}, \\ 1 & \text{if } x < \frac{1}{2}, \end{cases}$$

the exact solution is

$$u(x,y) = \begin{cases} \beta\ x(x-\frac{1}{2})y(y-1) & \text{if } x < \frac{1}{2}, \\ (x-\frac{1}{2})(x-1)y(1-y) & \text{if } x \geq \frac{1}{2}. \end{cases}$$

In this case, the preconditioner is taken to be the scaled BPX preconditioner given in (3.5.1). Table 4.12 compares the performance of the UPCG algorithm, as well as the cascadic version, with the standard PCG algorithm for $\beta = 10$. Table 4.13 compares the same algorithms for $\beta = 100$. In both tables, error $= \|A\nabla u - A\nabla u_h\|_{\tilde{Q}}$. From Tables 4.12 and 4.13, we see a similar behavior in the performance of the UPCG and standard PCG algorithms as in the previous example.

4.4.7 Remarks on the SPLS Method

In the case of no preconditioning, we observe for both convex and non-convex domains that the approximation of the flux is super-linear, and the method works

level k	PCG			UPCG			UPCG cascadic		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
1	0.244	0.000	1	0.255	0.000	1	0.254	0.000	1
2	0.127	0.939	4	0.133	0.936	3	0.132	0.939	3
3	0.064	0.985	6	0.067	0.986	5	0.068	0.963	3
4	0.032	0.995	7	0.034	0.989	6	0.034	0.978	3
5	0.016	1.000	9	0.017	1.001	8	0.017	0.999	3
6	0.008	0.999	10	0.008	0.996	9	0.009	1.007	3

Table 4.12: Comparison on interface example with $\beta = 10$.

level k	PCG			UPCG			UPCG cascadic		
$h = 2^{-k}$	error	rate	it	error	rate	it	error	rate	it
1	2.427	0.000	1	2.439	0.000	2	2.439	0.000	2
2	1.265	0.940	4	1.271	0.940	5	1.271	0.940	4
3	0.639	0.985	6	0.642	0.985	7	0.642	0.985	5
4	0.320	0.994	7	0.322	0.996	9	0.322	0.996	6
5	0.160	1.000	9	0.161	0.999	11	0.161	0.999	5
6	0.080	0.999	10	0.081	0.999	12	0.081	0.999	6

Table 4.13: Comparison on interface example with $\beta = 100$.

well no matter the size of the jump discontinuity. Also, we notice that the number of iterations depends on the size of the jump as well as h in the case of the interface problems even with (4.1.5), (4.1.6), (4.3.5) and (4.3.6) independent of the coeffficients of the matrix A and h. This can partly be due to fact that the stopping criteria depends on the matrix A, as inherited by choice of the $\|\cdot\|_h$ norm on the trial space, as well as h.

In the case of preconditioning, we observe that the approximation of the flux is super-linear for the case of using a projection type trial space, as in the case of no preconditioning. The number of iterations depends on the size of the jump as well as the mesh size h. According to Remark 3.3.3 and estimate (3.3.7), the number of iterations of Algorithm 3.3.1 depends on the condition number of the Schur complement of the unpreconditioned problem $\kappa(S_h)$ and the condition number of the elliptic preconditioner $\kappa(P_h A_h)$. From Proposition 2.3.5, Lemma 4.3.2, and estimates (4.1.5),

(4.2.1), we obtain

$$\kappa(S_h) \leq \frac{M^2}{m_h^2} \leq c,$$

with c independent of the size of the jump and the mesh size. For both the BPX and multigrid preconditioners we used in our numerical experiments, according to [92],

$$\kappa(P_h A_h) \leq c \min\left\{c_d(h), \frac{a_{max}}{a_{min}}\right\},$$

where $c_d(h) = |\log h|^2$ when $d = 2$ and $c_d(h) = h^{-1}$ when $d = 3$ (d refers to the dimension). Combining (3.3.7) with the above two inequalities, we obtain

$$\kappa(\tilde{S}_h) \leq C \min\left\{c_d(h), \frac{a_{max}}{a_{min}}\right\}.$$

We also note that a slight dependence on h is also due to the imposed stopping criterion as described in the case of no preconditioning.

Chapter 5

SPLS FOR REACTION DIFFUSION EQUATIONS

In this chapter, we will apply the SPLS method to reaction diffusion problems of the form

$$\begin{cases} -\varepsilon\Delta u + cu = f & \text{in} \quad \Omega, \\ u = 0 & \text{on } \partial\Omega, \end{cases} \tag{5.0.1}$$

for non-negative constants ε and c. A particular problem of interest is the reaction dominated case in which $\varepsilon \ll 1$. These types of equations arise in heat transfer problems in thin domains [2] as well as when using small step sizes are used in implicit time discretizations of parabolic reaction diffusion type problems [71]. The solutions to these problems are characterized by exponential boundary layers of width $\mathcal{O}(\varepsilon^{1/2}\ln(1/\varepsilon))$ [81], which pose a challenge numerically.

Finite element methods for these types of problems have been intensively studied, see e.g., [47, 67, 68, 69, 70, 71, 72, 81, 82]. Some of these references include least-squares approaches. In [71], a mixed method approach is given by introducing a new variable for ∇u, rewriting (5.0.1) as a first order system, and utilizing $H(\text{div};\Omega)$ conforming spaces. We consider an approach in which we adopt the use of graph type trial spaces. The advantage of this is no new variables are introduced, and the formulation involves H^1 type spaces and piecewise linear approximation.

The chapter is organized as follows. In Section 5.1, we detail the steps to fit (5.0.1) into the SPLS framework. Section 5.2 involves the discretization and choices of discrete trial spaces using a piecewise linear test space. The stability of the proposed discrete spaces is discussed in Section 5.3. In Section 5.4, we describe the construction of a Shishkin mesh, which is a specific type of mesh used to resolve the boundary layers

exhibited by the solutions for small ε. Lastly, numerical results are given in Section 5.5 to support and show the performance of the SPLS approach.

5.1 SPLS for Reaction Diffusion Equations

In this section, we will describe how to apply the general SPLS theory to problem (5.0.1). A standard variational formulation for (5.0.1) is: Find $u \in H_0^1(\Omega)$ such that

$$(\varepsilon \nabla u, \nabla v) + (cu, v) = (f, v) \quad \text{for all } v \in H_0^1(\Omega). \tag{5.1.1}$$

To fit this equation into the SPLS framework, we let $V := H_0^1(\Omega)$, $\tilde{Q} := L^2(\Omega) \times L^2(\Omega)^d$, and Q be the graph of the operator $\varepsilon\nabla : H_0^1(\Omega) \to L^2(\Omega)^d$, i.e.,

$$Q := G(\varepsilon\nabla) = \left\{ \left(\begin{smallmatrix} v \\ \varepsilon\nabla v \end{smallmatrix} \right) \mid v \in H_0^1(\Omega) \right\}.$$

Since the operator $\varepsilon\nabla$ is bounded from $H_0^1(\Omega)$ to $L^2(\Omega)^d$, the space Q is closed by the Closed Graph Theorem. We define the bilinear form $b : V \times \tilde{Q} \to \mathbb{R}$ as

$$b(v, \left(\begin{smallmatrix} q \\ \mathbf{q} \end{smallmatrix} \right)) := (cq, v) + (\mathbf{q}, \nabla v) \quad \text{for all } v \in V, \left(\begin{smallmatrix} q \\ \mathbf{q} \end{smallmatrix} \right) \in \tilde{Q},$$

and the linear functional $F \in V^*$ as

$$\langle F, v \rangle := (f, v) \quad \text{for all } v \in H_0^1(\Omega).$$

With this setting, the SPLS formulation of (5.1.1) is: Find $\mathbf{p} = \left(\begin{smallmatrix} u \\ \varepsilon\nabla u \end{smallmatrix} \right) \in Q$ such that

$$b(v, \mathbf{p}) = (cu, v) + (\varepsilon \nabla u, \nabla v) = (f, v) \quad \text{for all } v \in V. \tag{5.1.2}$$

On V, the inner product that we consider is

$$a(u, v) = (\varepsilon \nabla u, \nabla v) + (cu, v) \quad \text{for all } u, v \in V,$$

which gives rise to the norm

$$|v|_V = \left(\|c^{1/2} v\|^2 + \|\varepsilon^{1/2} \nabla v\|^2 \right)^{1/2}.$$

On $\tilde{Q}$, we consider the inner product

$$\left(\left(\begin{smallmatrix} q \\ \mathbf{q} \end{smallmatrix} \right), \left(\begin{smallmatrix} p \\ \mathbf{p} \end{smallmatrix} \right) \right)_{\tilde{Q}} = (cq, p) + (\varepsilon^{-1} \mathbf{q}, \mathbf{p}) \quad \text{for all } \left(\begin{smallmatrix} q \\ \mathbf{q} \end{smallmatrix} \right), \left(\begin{smallmatrix} p \\ \mathbf{p} \end{smallmatrix} \right) \in \tilde{Q}.$$

The corresponding norm is

$$\| \begin{pmatrix} q \\ \mathbf{q} \end{pmatrix} \|_{\tilde{Q}} = \left(\|c^{1/2} q\|^2 + \|\varepsilon^{-1/2} \mathbf{q}\|^2 \right)^{1/2}.$$

The operator $B : V \to \tilde{Q}$ is given by

$$Bv = \begin{pmatrix} v \\ \varepsilon \nabla v \end{pmatrix} \qquad \text{for all } v \in V.$$

In the setting, the compatibility condition (2.1.3) is automatically satisfied as

$$V_0 = \mathrm{Ker}(B) = \{ v \in H_0^1(\Omega) \,|\, Bv = 0 \} = \{0\}.$$

In addition, we obtain

$$\begin{aligned} \sup_{v \in V} \frac{b(v, \begin{pmatrix} u \\ \varepsilon \nabla u \end{pmatrix})}{|v|_V} &= \sup_{v \in V} \frac{(\varepsilon \nabla u, \nabla v) + (cu, v)}{(\|c^{1/2} v\|^2 + \|\varepsilon^{1/2} \nabla v\|^2)^{1/2}} \\ &\geq \frac{\|c^{1/2} u\|^2 + \|\varepsilon^{1/2} \nabla u\|^2}{(\|c^{1/2} u\|^2 + \|\varepsilon^{1/2} \nabla u\|^2)^{1/2}} \\ &= \| \begin{pmatrix} u \\ \varepsilon \nabla u \end{pmatrix} \|_{\tilde{Q}}, \end{aligned} \tag{5.1.3}$$

for any $\begin{pmatrix} u \\ \varepsilon \nabla u \end{pmatrix} \in Q$. This implies the $\inf - \sup$ condition on $V \times Q$. For the continuity of the bilinear form $b(\cdot, \cdot)$, note that

$$\begin{aligned} b(v, \begin{pmatrix} q \\ \mathbf{q} \end{pmatrix}) &= (cq, v) + (\mathbf{q}, \nabla v) \\ &= (c^{1/2} q, c^{1/2} v) + (\varepsilon^{-1/2} \mathbf{q}, \varepsilon^{1/2} \nabla v) \\ &\leq \|c^{1/2} q\| \, \|c^{1/2} v\| + \|\varepsilon^{-1/2} \mathbf{q}\| \, \|\varepsilon^{1/2} \nabla v\| \\ &\leq (\|c^{1/2} q\|^2 + \|\varepsilon^{-1/2} \mathbf{q}\|^2)^{1/2} (\|c^{1/2} v\|^2 + \|\varepsilon^{1/2} \nabla v\|^2)^{1/2} \\ &= |v|_V \| \begin{pmatrix} q \\ \mathbf{q} \end{pmatrix} \|_{\tilde{Q}}, \end{aligned} \tag{5.1.4}$$

by the Cauchy-Schwarz inequality for any $v \in V$ and $\begin{pmatrix} q \\ \mathbf{q} \end{pmatrix} \in \tilde{Q}$. Thus, the variational problem (5.1.2) is suitable for SPLS discretization.

5.2 SPLS Discretization for Reaction Diffusion Problems

In this section, we will discuss possible choices for the discrete spaces as well as their stability. The choices for the trial space will be based on the no projection and projection type spaces outlined in Section 2.3.1 and 2.3.2. For the discrete test space, we take $V_h \subset V = H_0^1(\Omega)$ to be the space of continuous piecewise polynomials of degree k with respect to the mesh $\mathcal{T}_h$.

5.2.1 No Projection Trial Space

Following Section 2.3.1, we consider the case when the trial space $\mathcal{M}_h$ is given by

$$\mathcal{M}_h := BV_h = \begin{pmatrix} I \\ \varepsilon\nabla \end{pmatrix} V_h,$$

where $I : V_h \to V_h$ is the identity operator and the inner product is chosen to coincide with the inner product on $\tilde{Q}$. By a similar argument used to show (5.1.3), we obtain

$$\sup_{v_h \in V_h} \frac{b\left(v_h, \binom{u_h}{\varepsilon\nabla u_h}\right)}{|v_h|_V} \geq \left\| \binom{u_h}{\varepsilon\nabla u_h} \right\|_{\tilde{Q}}, \tag{5.2.1}$$

for any $\binom{u_h}{\varepsilon\nabla u_h} \in \mathcal{M}_h$. Thus, we do have stability in this case. Furthermore, the stability is independent of the parameters c and ε.

Remark 5.2.1. *Having the stability constant independent of the parameters associated with the problem is particularly beneficial when dealing with the case of small ε, the case presented in this chapter, or the case of reaction diffusion problems with discontinuous coefficients. Preliminary results for the latter case will be discussed in Chapter 7.*

The discrete mixed variational formulation in this case becomes: Find $\mathbf{p}_h = \binom{u_h}{\varepsilon\nabla u_h}$, with $u_h \in V_h$, such that

$$b(v_h, \mathbf{p}_h) = (\varepsilon\nabla u_h, \nabla v_h) + (cu_h, v_h) = (f, v_h) \quad \text{for all } v_h \in V_h.$$

The discrete saddle point reformulation to be solved is: Find $\left(w_h, \mathbf{p}_h = \binom{u_h}{\varepsilon\nabla u_h}\right)$ such that

$$\begin{aligned} \varepsilon(\nabla w_h + \nabla u_h, \nabla v_h) \quad + \quad c(w_h + u_h, v_h) \quad &= (f, v_h) \qquad \text{for all } v_h \in V_h, \\ \binom{w_h}{\varepsilon\nabla w_h} \qquad\qquad\qquad\qquad\qquad\qquad\quad &= \mathbf{0}. \end{aligned}$$

5.2.2 Projection Type Trial Space

For the projection type trial space, we first define $\tilde{\mathcal{M}}_h \subset \tilde{Q} = L^2(\Omega) \times L^2(\Omega)^d$ to be

$$\tilde{\mathcal{M}}_h := M_{h,0} \times \varepsilon\mathbf{M}_{h,0},$$

where $M_{h,0}$ consists of continuous piecewise polynomials of degree k with respect to the mesh $\mathcal{T}_h$ with no restrictions on the boundary. The space $\mathbf{M}_{h,0}$ is the vector-valued product space in which each component consists of continuous piecewise polynomials of degree k. Two different choices for the projection type trial space, based on the inner product chosen for $\tilde{\mathcal{M}}_h$, are given in a similar way as the previous chapter. The first is outlined in this section. The second is outlined in Section 5.3.1.

For the first type of projection trial space, we equip $\tilde{\mathcal{M}}_h$ with the inner product induced from $\tilde{Q}$. Using the definition of R_h given in (2.3.3), we obtain $R_h \left(\begin{smallmatrix} q \\ \mathbf{q} \end{smallmatrix}\right)$ is the orthogonal projection of $\left(\begin{smallmatrix} q \\ \mathbf{q} \end{smallmatrix}\right)$ onto $\tilde{\mathcal{M}}_h$ with respect to the $(\cdot,\cdot)_{\tilde{Q}}$ inner product. More specifically, we have that

$$R_h \begin{pmatrix} q \\ \mathbf{q} \end{pmatrix} = \begin{pmatrix} Q_h^1 q \\ Q_h^2 \mathbf{q} \end{pmatrix},$$

where $Q_h^1 : L^2(\Omega) \to M_{h,0}$ is the orthogonal projection with respect to the weighted inner product $(\cdot,\cdot)_c$ and $Q_h^2 : L^2(\Omega)^d \to \mathbf{M}_{h,0}$ is the orthogonal projection with respect to the weighted inner product $(\cdot,\cdot)_{\varepsilon^{-1}}$. We now define the projection type trial space as

$$\mathcal{M}_h := R_h^{\text{orth}} B V_h,$$

where the elements are given by

$$R_h^{\text{orth}} B v_h = \begin{pmatrix} Q_h^1 v_h \\ Q_h^2(\varepsilon \nabla v_h) \end{pmatrix}.$$

Remark 5.2.2. *In general, $\mathcal{M}_h$ constructed in this way is not contained in Q. The reasoning is similar with the discussion of the orthogonal projection type trial space of the previous chapter in Remark 4.2.1. A similar justification holds for the lump projection trial space outlined in Section 5.3.1.*

The discrete mixed variational formulation in this case is: Find $\mathbf{p}_h = R_h^{\text{orth}} B u_h$, with $u_h \in V_h$, such that

$$b(v_h, \mathbf{p}_h) = (f, v_h) \quad \text{for all } v_h \in V_h,$$

where $b(\cdot,\cdot)$ is defined in Section 5.1. The SPLS discretization (2.2.4) to be solved is: Find $(w_h, \mathbf{p}_h = R_h^{\text{orth}} A\nabla u_h)$ such that

$$\begin{aligned} a(w_h, v_h) \quad &+ \quad b(v_h, \mathbf{p}_h) \quad &&= (f, v_h) \qquad \text{for all } v_h \in V_h, \\ R_h^{\text{orth}} A\nabla w_h \quad & \qquad\qquad\quad &&= \mathbf{0}. \end{aligned} \tag{5.2.2}$$

5.3 Piecewise Linear Test Space

In this section, we discuss the stability for the family of spaces $\{(V_h, \mathcal{M}_h)\}$, where $\mathcal{M}_h$ is as outlined in Section 5.2.2. For simplicity, we assume $\Omega \subset \mathbb{R}^2$ is a polygonal domain. The results can be extended to polyhedral domains in $\mathbb{R}^3$. We also assume that the triangular mesh $\mathcal{T}_h$ is locally quasi-uniform. Let $\{z_1, \ldots, z_N\}$ be the set of all nodes of $\mathcal{T}_h$ and assume all triangles adjacent to z_j are of regular shape and their area is of order h_j^2. In this notation, the mesh size of $\mathcal{T}_h$ is $h := \max\{h_1, h_2, \ldots, h_N\}$.

Remark 5.3.1. *We note that while the analysis done in this section assumes that the mesh $\mathcal{T}_h$ is locally quasi-uniform, the Shishkin type mesh, that will be outlined in Section 5.4, does not satisfy this property. Nevertheless, the analysis presented in this section can be applied to reaction diffusion problems in which the solutions do not exhibit boundary layers, such as the problem presented in Section 5.5.1 or the interface problem presented in Chapter 7. A rigorous analysis of the stability of the family of spaces $\{(V_h, \mathcal{M}_h)\}$ on Shishkin type meshes, where $\mathcal{M}_h$ is of projection type, will be conducted in the near future.*

We take V_h to be the space consisting of piecewise linear polynomials with respect to $\mathcal{T}_h$ vanishing on the boundary of Ω. Also, we take $M_{h,0}$ to consist of continuous linear piecewise polynomials with respect to the mesh $\mathcal{T}_h$. Let $\{\phi_1, \ldots, \phi_N\}$ denote a nodal basis for $M_{h,0}$ with respect to the mesh $\mathcal{T}_h$ and $\{\mathbf{\Phi}_1, ..., \mathbf{\Phi}_{2N}\}$ denote a nodal basis for $\mathbf{M}_{h,0}$, where $\mathbf{\Phi}_j = (\phi_j, 0)^T$ and $\mathbf{\Phi}_{N+j} = (0, \phi_j)^T$ for $j = 1, \ldots, N$. With this notation, $\{\phi_j\}_{j=1}^N \cup \{\varepsilon\mathbf{\Phi}_j\}_{j=1}^{2N}$ is a basis for $\tilde{\mathcal{M}}_h$. We further define M_ε to be the matrix

whose entries are $(\varepsilon\mathbf{\Phi}_i, \varepsilon\mathbf{\Phi}_j)_{\varepsilon^{-1}} = (\varepsilon\mathbf{\Phi}_i, \mathbf{\Phi}_j)$ and $H := \operatorname{diag}(h_1^2, h_2^2, \ldots, h_N^2)$. Lastly, we let

$$D_\varepsilon = \left[\begin{array}{c|c} \varepsilon H & \\ \hline & \varepsilon H \end{array}\right].$$

Lemma 5.3.2. *Under the assumptions of Section 5.3,*

$$\langle M_\varepsilon \boldsymbol{\gamma}, \boldsymbol{\gamma}\rangle_e \leq C \langle D_\varepsilon \boldsymbol{\gamma}, \boldsymbol{\gamma}\rangle_e \quad \text{for all } \boldsymbol{\gamma} \in \mathbb{R}^{2N}. \tag{5.3.1}$$

Consequently,

$$\langle M_\varepsilon^{-1} \boldsymbol{\gamma}, \boldsymbol{\gamma}\rangle_e \geq C \langle D_\varepsilon^{-1} \boldsymbol{\gamma}, \boldsymbol{\gamma}\rangle_e \quad \text{for all } \boldsymbol{\gamma} \in \mathbb{R}^{2N}, \tag{5.3.2}$$

where c is independent of h and ε.

Proof. Let $\boldsymbol{\gamma} \in \mathbb{R}^{2N}$ and define $\mathbf{q}_h := \sum_{j=1}^{2N} \boldsymbol{\gamma}_j \mathbf{\Phi}_j$. Note that

$$\langle M_\varepsilon \boldsymbol{\gamma}, \boldsymbol{\gamma}\rangle_e = (\varepsilon \mathbf{q}_h, \mathbf{q}_h) = \|\varepsilon \mathbf{q}_h\|_{\varepsilon^{-1}}^2 = \sum_{\tau \in \mathcal{T}_h} \|\varepsilon \mathbf{q}_h\|_{\tau,\varepsilon^{-1}}^2. \tag{5.3.3}$$

If $\tau = [z_{1_\tau}, z_{2_\tau}, z_{3_\tau}]$, then

$$\mathbf{q}_h\big|_\tau = \begin{pmatrix} \sum_{j=1}^{3} \gamma_{j_\tau} \phi_{j_\tau} \\ \sum_{j=1}^{3} \gamma_{(j+N)_\tau} \phi_{j_\tau} \end{pmatrix}.$$

Hence,

$$\|\varepsilon \mathbf{q}_h\|_{\tau,\varepsilon^{-1}}^2 \leq C\,|\tau| \left(\varepsilon \sum_{j=1}^{3} \gamma_{j_\tau}^2 + \varepsilon \sum_{j=1}^{3} \gamma_{(j+N)_\tau}^2 \right). \tag{5.3.4}$$

Using (5.3.3), (5.3.4), and the fact that each coefficient γ_k can repeat at most three times, we obtain

$$\langle M_\varepsilon \boldsymbol{\gamma}, \boldsymbol{\gamma}\rangle_e \leq C \left(\varepsilon \sum_{j=1}^{N} h_j^2 \gamma_j^2 + \varepsilon \sum_{j=1}^{N} h_j^2 \gamma_{j+N}^2 \right) = C \, \langle D_\varepsilon \boldsymbol{\gamma}, \boldsymbol{\gamma}\rangle_e .$$

The estimate (5.3.2) follows from (5.3.1). □

We now show that (2.3.5) is satisfied for the operator R_h^{orth} defined Section 5.2.2.

Lemma 5.3.3. *Under the assumptions of Section 5.3, there exists a constant C, independent of h and ε, such that*

$$\|R_h^{\text{orth}} \begin{pmatrix} v_h \\ \varepsilon\nabla v_h \end{pmatrix}\|_h \geq C \|\begin{pmatrix} v_h \\ \varepsilon\nabla v_h \end{pmatrix}\|_{\tilde{Q}} \quad \text{for all } v_h \in V_h. \tag{5.3.5}$$

Proof. For a fixed $\begin{pmatrix} v_h \\ \varepsilon\nabla v_h \end{pmatrix}$, with $v_h \in V_h$, we define the vector $\mathbf{G}_h \in \mathbb{R}^{2N}$ by

$$(G_h)_i := (\varepsilon\nabla v_h, \varepsilon\mathbf{\Phi}_i)_{\varepsilon^{-1}} = (\varepsilon\nabla v_h, \mathbf{\Phi}_i) \quad i = 1, ..., 2N.$$

Recall that

$$R_h^{\text{orth}} \begin{pmatrix} v_h \\ \varepsilon\nabla v_h \end{pmatrix} = \begin{pmatrix} Q_h^1 v_h \\ Q_h^2(\varepsilon\nabla v_h) \end{pmatrix},$$

where Q_h^1 and Q_h^2 are defined in Section 5.2.2. Note that $Q_h^1 v_h = v_h$ and let

$$Q_h^2(\varepsilon\nabla v_h) = \sum_{i=1}^{2N} \alpha_i \varepsilon\mathbf{\Phi}_i$$

Thus, $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_{2N})^T$ is a solution to

$$M_\varepsilon \, \boldsymbol{\alpha} = \mathbf{G}_h.$$

Using (5.3.2), we obtain

$$\begin{aligned} \|R_h^{\text{orth}} \begin{pmatrix} v_h \\ \varepsilon\nabla v_h \end{pmatrix}\|_h^2 &= \|c^{1/2} v_h\|^2 + \sum_{i,j=1}^{2N} \alpha_i \, \alpha_j \, (\varepsilon\mathbf{\Phi}_i, \mathbf{\Phi}_j) \\ &= \|c^{1/2} v_h\| + \left\langle M_\varepsilon^{-1}\mathbf{G}_h, \mathbf{G}_h \right\rangle_e \\ &\geq C \left(\|c^{1/2} v_h\| + \left\langle D_\varepsilon^{-1}\mathbf{G}_h, \mathbf{G}_h \right\rangle_e \right). \end{aligned}$$

From the definition of the matrix H, we recall $h_i = h_{i+N}$ for $i = 1, \ldots, N$. Thus,

$$\begin{aligned} \left\langle D_\varepsilon^{-1}\mathbf{G}_h, \mathbf{G}_h \right\rangle_e &= \sum_{i=1}^{N} h_i^{-2} \left[\varepsilon \left(\frac{\partial v_h}{\partial x}, \phi_i \right)^2 + \varepsilon \left(\frac{\partial v_h}{\partial y}, \phi_i \right)^2 \right] \\ &= \sum_{i=1}^{N} \sum_{\tau \subset supp(\phi_i)} h_i^{-2} (1, \phi_i)_\tau^2 \left[\varepsilon \left| \frac{\partial v_h}{\partial x} \right|_\tau^2 + \varepsilon \left| \frac{\partial v_h}{\partial y} \right|_\tau^2 \right] \\ &\geq C \|\varepsilon\nabla v_h\|_{\varepsilon^{-1}}^2. \end{aligned}$$

Hence,

$$\|R_h^{\text{orth}} \begin{pmatrix} v_h \\ \varepsilon\nabla v_h \end{pmatrix}\|_h^2 \geq C \left(\|c^{1/2} v_h\|^2 + \|\varepsilon \nabla v_h\|_{\varepsilon^{-1}}^2 \right) = C \| \begin{pmatrix} v_h \\ \varepsilon\nabla v_h \end{pmatrix} \|_{\tilde{Q}}.$$

□

As a consequence of Lemma 5.3.3, equation (5.2.1), and Proposition 2.3.2, we obtain the following result.

Theorem 5.3.4. *Let $\Omega \subset \mathbb{R}^2$ be a polygonal domain and $\{T_h\}$ be a family of locally quasi-uniform meshes for Ω. For each h, let V_h be the space of continuous linear functions with respect to the mesh $\{\mathcal{T}_h\}$ that vanish on $\partial\Omega$ and $\mathcal{M}_h = R_h^{\text{orth}} B V_h$. Then the family of spaces $\{(V_h, \mathcal{M}_h)\}$ is stable.*

5.3.1 Second Type of Projection Trial Space

In this section, we consider an inner product on $\tilde{M}_h$ that is related with lumping the mass matrix and the theory presented in Section 4.3.1. Let $\begin{pmatrix} q_h \\ \mathbf{q}_h \end{pmatrix}, \begin{pmatrix} p_h \\ \mathbf{p}_h \end{pmatrix} \in \tilde{\mathcal{M}}_h$ be two arbitrary elements. We can write

$$\mathbf{q}_h = \sum_{i=1}^{2N} \alpha_i \varepsilon \mathbf{\Phi}_i, \text{ and } \mathbf{p}_h = \sum_{i=1}^{2N} \beta_i \varepsilon \mathbf{\Phi}_i,$$

for some $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_{2N})$ and $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_{2N})$. We consider the following inner product on $\tilde{\mathcal{M}}_h$:

$$\left(\begin{pmatrix} q_h \\ \mathbf{q}_h \end{pmatrix}, \begin{pmatrix} p_h \\ \mathbf{p}_h \end{pmatrix} \right)_h := (c q_h, p_h) + \sum_{i=1}^{2N} \alpha_i \beta_i (1, \varepsilon \mathbf{\Phi}_i).$$

For simplicity, we will denote

$$(\mathbf{q}_h, \mathbf{p}_h)_{\text{lump}} := \sum_{i=1}^{2N} \alpha_i \beta_i (1, \varepsilon \mathbf{\Phi}_i),$$

for the second part of the $(\cdot, \cdot)_h$ inner product. Note that for any $\mathbf{q} \in L^2(\Omega)^d$

$$\left(\sum_{i=1}^{2N} \frac{(\mathbf{q}, \varepsilon \mathbf{\Phi}_i)_{\varepsilon^{-1}}}{(1, \varepsilon \mathbf{\Phi}_i)} \varepsilon \mathbf{\Phi}_i, \varepsilon \mathbf{\Phi}_j \right)_{\text{lump}} = (\mathbf{q}, \varepsilon \mathbf{\Phi}_j)_{\varepsilon^{-1}} \quad \text{for all } \varepsilon \mathbf{\Phi}_j \in \mathbf{M}_{h,0}.$$

Using (2.3.3), we conclude $R_h : \tilde{Q} \to \tilde{\mathcal{M}}_h$ is given by

$$R_h \begin{pmatrix} q \\ \mathbf{q} \end{pmatrix} = \begin{pmatrix} Q_h^1 q \\ Q_h^{\text{lump}} \mathbf{q} \end{pmatrix},$$

where

$$Q_h^{\text{lump}} \mathbf{q} = \sum_{i=1}^{2N} \frac{(\mathbf{q}, \varepsilon \mathbf{\Phi}_i)_{\varepsilon^{-1}}}{(1, \varepsilon \mathbf{\Phi}_i)} \varepsilon \mathbf{\Phi}_i = \sum_{i=1}^{2N} \frac{(\mathbf{q}, \mathbf{\Phi}_i)}{(1, \mathbf{\Phi}_i)} \mathbf{\Phi}_i.$$

We define the projection type trial space in this case as

$$\mathcal{M}_h := R_h^{\text{lump}} B V_h.$$

The problem to be solved using this projection type trial space is identical to (5.2.2). The following lemma is analogous to 5.3.3.

Lemma 5.3.5. *Under the assumptions of Section 5.3, there exists a constant C, independent of h and ε, such that*

$$\| R_h^{\text{lump}} \begin{pmatrix} v_h \\ \varepsilon \nabla v_h \end{pmatrix} \|_h \geq C \, \| \begin{pmatrix} v_h \\ \varepsilon \nabla v_h \end{pmatrix} \|_{\tilde{Q}} \quad \text{for all } v_h \in V_h. \tag{5.3.6}$$

Proof. Using the same notation from the proof of Lemma 5.3.3, we obtain

$$\begin{aligned} \| R_h^{\text{lump}} \begin{pmatrix} v_h \\ \varepsilon \nabla v_h \end{pmatrix} \|_h^2 &= \| c^{1/2} v_h \|^2 + \sum_{i=1}^{2N} \frac{(\varepsilon \nabla v_h, \varepsilon \mathbf{\Phi}_i)_{\varepsilon^{-1}}^2}{(1, \varepsilon \mathbf{\Phi}_i)^2} (1, \varepsilon \mathbf{\Phi}_i) \\ &= \| c^{1/2} v_h \|^2 + \sum_{i=1}^{2N} \frac{(\varepsilon \nabla v_h, \mathbf{\Phi}_i)^2}{(1, \varepsilon \mathbf{\Phi}_i)} \\ &\geq C \left(\| c^{1/2} v_h \|^2 + \left\langle D_\varepsilon^{-1} \mathbf{G}_h, \mathbf{G}_h \right\rangle_e \right), \end{aligned}$$

where

$$(G_h)_i := (\varepsilon \nabla v_h, \varepsilon \mathbf{\Phi}_i)_{\varepsilon^{-1}} = (\varepsilon \nabla v_h, \mathbf{\Phi}_i) \quad i = 1, ..., 2N.$$

From the same techniques used to estimate $\langle D_\varepsilon^{-1} \mathbf{G}_h, \mathbf{G}_h \rangle_e$ as in the proof of Lemma 5.3.3, the result follows. □

As a consequence of Lemma 5.3.5, we obtain the following result.

Theorem 5.3.6. *Let $\Omega \subset \mathbb{R}^2$ be a polygonal domain and $\{T_h\}$ be a family of locally quasi-uniform meshes for Ω. For each h, let V_h be the space of continuous linear functions with respect to the mesh $\{\mathcal{T}_h\}$ that vanish on $\partial\Omega$ and $\mathcal{M}_h = R_h^{\text{lump}} B V_h$. Then the family of spaces $\{(V_h, \mathcal{M}_h)\}$ is stable.*

5.4 The Construction of a Shishkin Mesh

In this section, we describe the construction of a Shishkin mesh [86] for the unit square. These types of meshes are widely used when dealing with reaction dominated diffusion problems in order to resolve the boundary layers exhibited by the solution of the problem. This type of mesh will be used in Sections 5.5.2, 5.5.3, and 5.5.4. We will follow the outline given in [81] to construct a Shishkin mesh for a solution that exhibits boundary layers on all sides of the unit square.

We first assume N is an integer multiple of 8. This parameter will refer to the number of mesh intervals in the x and y directions. The mesh itself is the tensor product of two one-dimensional Shishkin meshes $\mathcal{T}_x \times \mathcal{T}_y$. The process for obtaining $\mathcal{T}_x$ (and $\mathcal{T}_y$) is as follows. The interval $[0,1]$ is first decomposed into three subintervals $[0,\lambda]$, $[\lambda, 1-\lambda]$, and $[1-\lambda, 1]$, where

$$\lambda = \min\left\{\frac{1}{4}, 2\sqrt{\frac{\varepsilon}{c^*}}\ln N\right\} \quad \text{with} \quad 0 < c^* < c. \tag{5.4.1}$$

The intervals $[0,\lambda]$ and $[1-\lambda,1]$ are then partitioned into $N/4$ subintervals of length $\dfrac{4\lambda}{N}$, while the interval $[\lambda, 1-\lambda]$ is partitioned into $N/2$ subintervals of length $\dfrac{2(1-2\lambda)}{N}$. The triangular mesh is obtained by drawing diagonals from the top left to bottom right of each quadrilateral. Figure 5.1 shows an example of the Shishkin mesh generated using $\varepsilon = 10^{-4}$ and $c^* = \sqrt{1/2}$ for $N = 16, 32$, respectively.

5.5 Numerical Results

In this section, we present results from applying the SPLS discretization techniques on second order elliptic PDE of the form (5.0.1). For all of the examples presented, Ω is a bounded polygonal domain, and the test space $V_h \subset H_0^1(\Omega)$ is taken to be the space of continuous piecewise linear polynomials with respect to the Shishkin mesh $\mathcal{T}_h$, unless otherwise noted. We consider all types of trial spaces presented in this chapter: the no projection type presented in Section 5.2.1 and the projection types

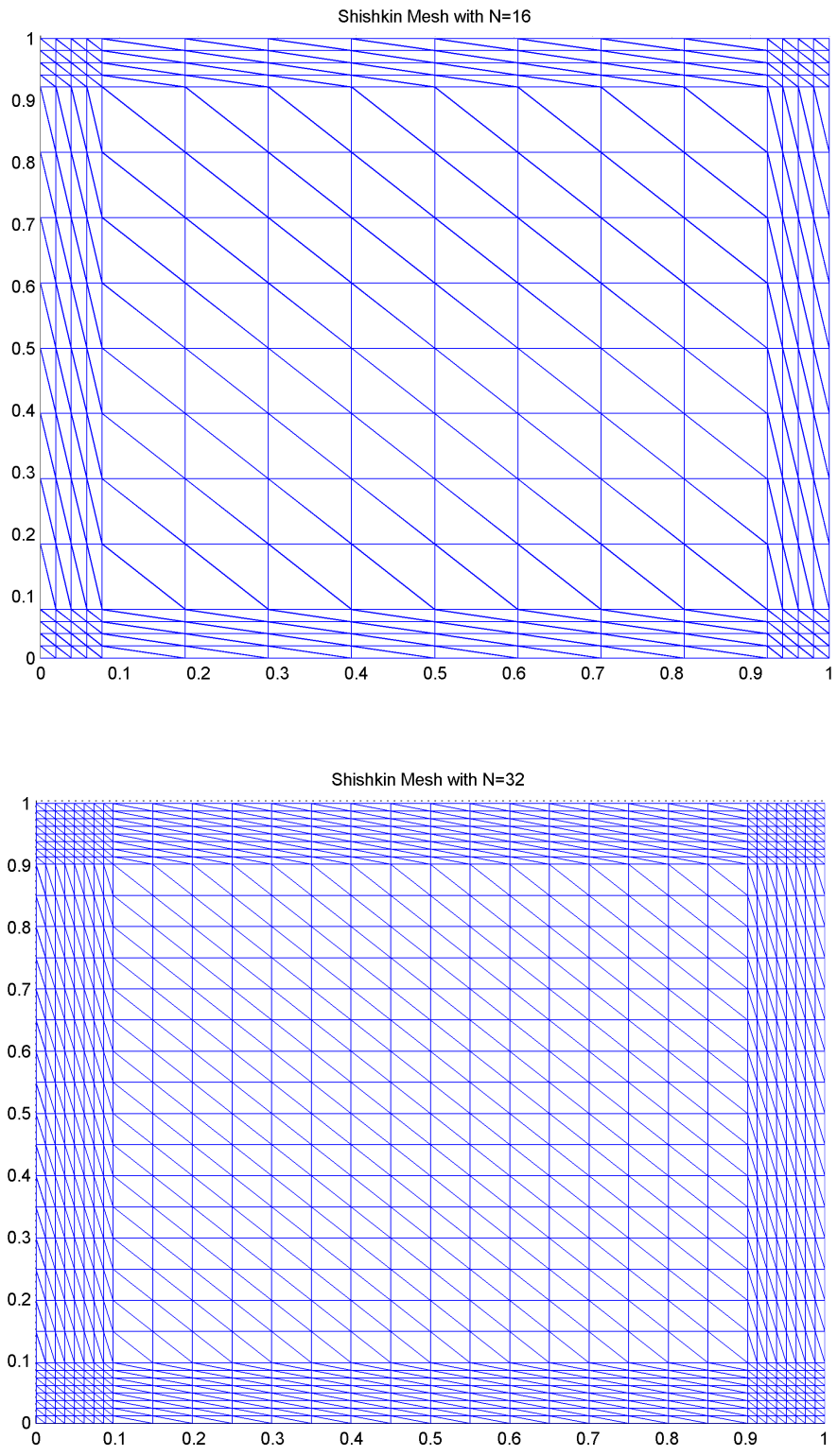

Figure 5.1: Example of a Shishkin mesh using $N = 16, 32$ subintervals.

presented in Sections 5.2.2 and 5.3.1. Also, we note that while the theory in this chapter considers c a non-negative constant, the theory extends to the case where c is a smooth positive function satisfying

$$0 < c_0 \leq c(\mathbf{x}) \leq c_1 \quad \text{for all } \mathbf{x} \in \Omega,$$

for constants c_0 and c_1.

For the singularly perturbed problems, we measure the SPLS solution in a balanced norm instead of the norm on $\tilde{Q}$. This is due to the fact that for small ε the L^2 part of the norm (on $\tilde{Q}$) dominates, leading to an unbalanced norm not adequate to accurately measure the error, see [71, 81]. More specifically, we measure

$$\text{error} = \left(\|u - u_h\|^2 + \varepsilon^{1/2}\|\nabla u - \nabla u_h\|^2\right)^{1/2},$$

for the no projection type trial space and measure

$$\text{error} = \left(\|u - u_h\|^2 + \varepsilon^{1/2}\|\nabla u - R_h \nabla u_h\|^2\right)^{1/2},$$

for the projecton type trial spaces. In the above equation, R_h can be taken as either the orthogonal projection described in Secton 5.2.2 or the lump projection described in Section 5.3.1.

When using a Shishkin mesh, we used a stopping criterion of

$$\|\mathbf{q}_j\|_h \leq c_0(N^{-1} \ln N),$$

for the no projection type of trial space. This is because standard Galerkin methods for (5.1.1) obtain a covergence rate of $\mathcal{O}(N^{-1} \ln N)$ using piecewise linear approximation [71, 81]. When using a Shishkin mesh and a projection type trial space, we used a stopping criterion of

$$\|\mathbf{q}_j\|_h \leq c_0(N^{-1} \ln N)^2.$$

The convergence rates when using a Shishkin mesh are computed under the assumption that we have a convergence rate of $\mathcal{O}((N^{-1} \ln N)^r)$. More specifically, if (N_1, e_{N_1})

and (N_2, e_{N_2}) correspond to the number of partitions in the Shishkin mesh and the discretization error for two consecutive levels of refinement, then

$$r = \frac{\ln e_{N_1} - \ln e_{N_2}}{\ln(N_1^{-1} \ln N_1) - \ln(N_2^{-1} \ln N_2)}.$$

5.5.1 Basic Unit Square Problem

For the first example, we solved (5.0.1) on the unit square with $c = 1$, $\varepsilon = 1$, and f computed such that the exact solution is given by

$$u(x, y) = x(1 - x)y(1 - y).$$

The family of locally quasi-uniform meshes $\{\mathcal{T}_h\}$ was obtained through a standard uniform refinement strategy starting with a uniform coarse mesh. Here, the mesh size is $h = 2^{-k}$ where k is the level of refinement. Based on the first inequality of (2.4.1), we used a stopping criterion of

$$\|\mathbf{q}_j\|_h \leq c_0 h^2,$$

on each level, and the error is computed in the $\tilde{Q}$ norm. Results for all three types of trial spaces are shown in Table 5.1. We see $\mathcal{O}(h)$ convergence for the no projection trial space and super-linear convergence for both types of projection type trial spaces.

level k	$\mathcal{M}_h = BV_h$			$\mathcal{M}_h = R_h^{\text{orth}} BV_h$			$\mathcal{M}_h = R_h^{\text{lump}} BV_h$		
	error	rate	it	error	rate	it	error	rate	it
1	0.045		1	0.0100		3	0.0202		3
2	0.024	0.903	1	0.0034	1.569	7	0.0090	1.168	6
3	0.012	0.974	1	0.0010	1.735	8	0.0035	1.364	8
4	0.006	0.993	1	3.1e-04	1.724	10	0.0013	1.440	13
5	0.003	0.998	1	8.9e-05	1.785	12	0.0004	1.471	16

Table 5.1: Results for basic unit square example.

5.5.2 Example With Boundary Layers on All Sides

For this example, we solved (5.0.1) on the unit square with variable coefficient $c = 2(1 + x^2 + y^2)$ and f computed such that the exact solution is given by

$$u(x,y) = x(1-x)\left(1 - e^{-y/\sqrt{\varepsilon}}\right)\left(1 - e^{(y-1)/\sqrt{\varepsilon}}\right) + y(1-y)\left(1 - e^{-x/\sqrt{\varepsilon}}\right)\left(1 - e^{(x-1)/\sqrt{\varepsilon}}\right),$$

as considered in [67]. For this example, the family of Shishkin meshes $\{\mathcal{T}_h\}$ was obtained as in Section 5.4 with λ in (5.4.1) computed with $c^* = \sqrt{1/2}$ and the number of subintervals in the x and y directions taken to be $N = 16, 32, 64, 128$, and 256. Table 5.2 shows results for no projection trial space for a variety of values for ε. We observe $\mathcal{O}(N^{-1} \ln N)$ convergence. Tables 5.3 and 5.4 display results for the orthogonal and lump projection type trial spaces. In this case, we observe $\mathcal{O}((N^{-1} \ln N)^2)$ convergence. Furthermore, for all three types of trial spaces we observe the order of convergence is robust with repect to ε.

$\mathcal{M}_h = BV_h$

N	$\varepsilon = 1$			$\varepsilon = 10^{-2}$			$\varepsilon = 10^{-4}$		
	error	rate	it	error	rate	it	error	rate	it
16	0.019		1	0.068		1	0.132		1
32	0.009	1.472	1	0.034	1.471	1	0.088	0.854	1
64	0.005	1.356	1	0.017	1.356	1	0.054	0.946	1
128	0.002	1.286	1	0.009	1.285	1	0.032	0.984	1
256	0.001	1.239	1	0.004	1.239	1	0.018	0.996	1
N	$\varepsilon = 10^{-8}$			$\varepsilon = 10^{-12}$			$\varepsilon = 10^{-16}$		
	error	rate	it	error	rate	it	error	rate	it
16	0.133		1	0.134		1	0.134		1
32	0.089	0.859	1	0.089	0.859	1	0.089	0.860	1
64	0.055	0.951	1	0.055	0.951	1	0.055	0.951	1
128	0.032	0.988	1	0.032	0.988	1	0.032	0.988	1
256	0.018	0.999	1	0.018	0.999	1	0.018	0.999	1

Table 5.2: Results for example with boundary layers on all sides and no projection trial space.

$\mathcal{M}_h = R_h^{\text{orth}} BV_h$

N	$\varepsilon = 1$			$\varepsilon = 10^{-2}$			$\varepsilon = 10^{-4}$		
	error	rate	it	error	rate	it	error	rate	it
16	0.0027		3	0.0177		4	0.073		5
32	0.0008	2.490	3	0.0054	2.509	4	0.038	1.417	8
64	0.0003	2.203	3	0.0018	2.190	4	0.016	1.708	12
128	9.0e-05	2.022	3	0.0005	2.191	5	0.006	1.903	19
256	3.1e-05	1.907	3	0.0002	1.910	5	0.002	1.978	28
N	$\varepsilon = 10^{-8}$			$\varepsilon = 10^{-12}$			$\varepsilon = 10^{-16}$		
	error	rate	it	error	rate	it	error	rate	it
16	0.073		4	0.073		5	0.073		6
32	0.038	1.419	6	0.038	1.419	8	0.038	1.419	10
64	0.016	1.710	9	0.016	1.711	12	0.016	1.711	16
128	0.006	1.903	12	0.006	1.906	19	0.006	1.906	25
256	0.002	1.972	17	0.002	1.981	28	0.002	1.981	40

Table 5.3: Results for example with boundary layers on all sides and orthogonal projection.

$\mathcal{M}_h = R_h^{\text{lump}} BV_h$

N	$\varepsilon = 1$			$\varepsilon = 10^{-2}$			$\varepsilon = 10^{-4}$		
	error	rate	it	error	rate	it	error	rate	it
16	0.0048		4	0.0281		5	0.099		3
32	0.0017	2.222	4	0.0088	2.455	6	0.058	1.148	4
64	0.0006	2.042	4	0.0028	2.197	6	0.027	1.515	6
128	0.0002	1.933	4	0.0010	2.052	7	0.010	1.839	8
256	7.3e-05	1.860	4	0.0003	1.898	7	0.003	1.972	11
N	$\varepsilon = 10^{-8}$			$\varepsilon = 10^{-12}$			$\varepsilon = 10^{-16}$		
	error	rate	it	error	rate	it	error	rate	it
16	0.100		4	0.100		5	0.100		7
32	0.058	1.153	7	0.058	1.153	9	0.058	1.154	11
64	0.027	1.524	10	0.027	1.524	13	0.027	1.524	17
128	0.010	1.855	14	0.010	1.856	21	0.010	1.856	27
256	0.003	2.015	21	0.003	2.016	32	0.003	2.016	44

Table 5.4: Results for example with boundary layers on all sides and lump projection.

5.5.3 Example With Nonhomogeneous Boundary Condition

For this example, we solved (5.0.1) on the unit square with variable coefficient $c = 1 + x^2y^2e^{xy/2}$ and f computed such that the exact solution is

$$u(x,y) = x^3(1+y^2) + \sin(\pi x^2) + \cos(\pi y/2) \\ + (x+y)\left(e^{-2x/\sqrt{\varepsilon}} + e^{2(x-1)/\sqrt{\varepsilon}} + e^{-3y/\sqrt{\varepsilon}} + e^{3(y-1)/\sqrt{\varepsilon}}\right),$$

as considered in [71]. The family of Shishkin meshes $\{\mathcal{T}_h\}$ is obtained as in Section 5.5.2. Table 5.5 shows results for the no projection trial space and various values of ε. We observe $\mathcal{O}(N^{-1}\ln N)$ convergence and that the order is robust with respect to ε. Figure 5.2 shows the exact solution and numerical approximation for $\varepsilon = 10^{-4}$, respectively.

$\mathcal{M}_h = BV_h$

N	$\varepsilon = 1$			$\varepsilon = 10^{-2}$			$\varepsilon = 10^{-4}$		
	error	rate	it	error	rate	it	error	rate	it
16	0.205		1	1.082		1	2.009		1
32	0.103	1.468	1	0.595	1.273	1	1.666	0.398	1
64	0.051	1.355	1	0.306	1.303	1	1.220	0.610	1
128	0.026	1.286	1	0.154	1.273	1	0.791	0.804	1
256	0.013	1.239	1	0.077	1.235	1	0.472	0.921	1
N	$\varepsilon = 10^{-8}$			$\varepsilon = 10^{-12}$			$\varepsilon = 10^{-16}$		
	error	rate	it	error	rate	it	error	rate	it
16	1.989		1	1.988		1	1.988		1
32	1.652	0.394	1	1.652	0.394	1	1.652	0.394	1
64	1.212	0.607	1	1.212	0.607	1	1.212	0.607	1
128	0.786	0.802	1	0.786	0.802	1	0.786	0.802	1
256	0.470	0.920	1	0.470	0.920	1	0.470	0.920	1

Table 5.5: Results for non-homogeneous example with no projection trial space.

5.5.4 Example With Boundary Layers on Two Sides

For the last example, we solved 5.0.1 on the unit square with $c = 2$ and f computed such that the exact solution is given by

$$u(x,y) = y(1-y)\left(1 - e^{-x/\sqrt{\varepsilon}}\right)\left(1 - e^{(x-1)/\sqrt{\varepsilon}}\right),$$

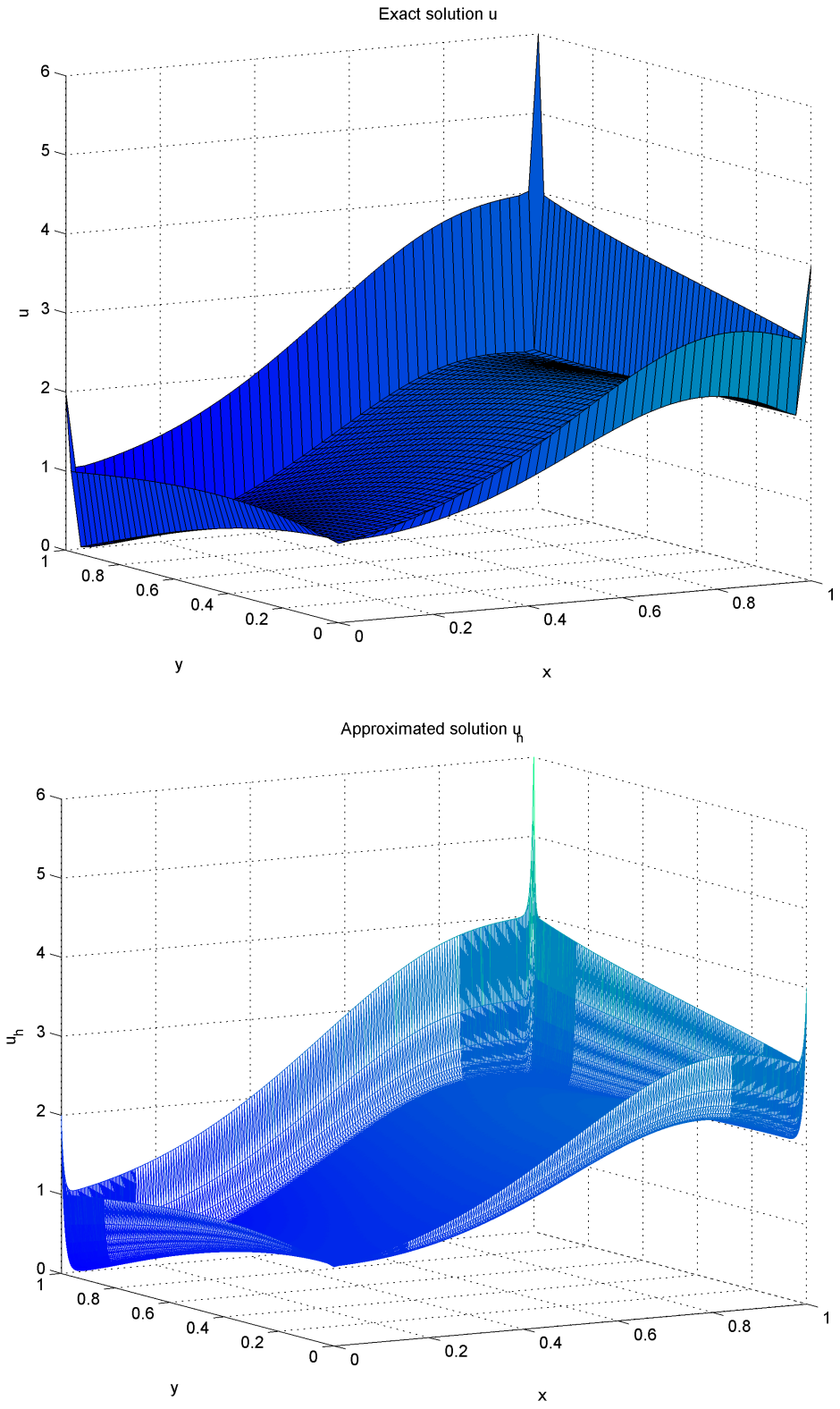

Figure 5.2: Exact and SPLS solution for $\varepsilon = 10^{-4}$.

as considered in [67]. Due to the nature of the solution, we expect boundary layers at $x = 0$ and $x = 1$. To this end, we construct the family of Shishkin meshes $\{\mathcal{T}_h\}$ such that the subintervals in the x direction are partitioned as described in Section 5.4 using $c^* = \sqrt{1/2}$ and $N = 16, 32, 64, 128, 256$, while the partition in the y direction is uniform with N subintervals. Figure 5.3 shows the mesh generated with $\varepsilon = 10^{-4}$ and $N = 16, 32$, respectively.

Table 5.6 shows results for the no projection trial space for various values of ε. As in the previous two examples, we observe $\mathcal{O}(N^{-1} \ln N)$ convergence in the balanced norm. Tables 5.7 and 5.8 display results for the orthogonal and lump projection type spaces, respectively. We observe $\mathcal{O}((N^{-1} \ln N)^2)$ convergence in the balanced norm as in Section 5.5.2. Furthermore, the order of convergence is robust with respect to ε.

$\mathcal{M}_h = BV_h$

N	$\varepsilon = 1$			$\varepsilon = 10^{-2}$			$\varepsilon = 10^{-4}$		
	error	rate	it	error	rate	it	error	rate	it
16	0.0094		1	0.040		1	0.091		1
32	0.0047	1.472	1	0.020	1.460	1	0.062	0.839	1
64	0.0024	1.356	1	0.010	1.353	1	0.038	0.935	1
128	0.0012	1.286	1	0.005	1.285	1	0.022	0.978	1
256	0.0006	1.239	1	0.002	1.238	1	0.013	0.993	1
N	$\varepsilon = 10^{-8}$			$\varepsilon = 10^{-12}$			$\varepsilon = 10^{-16}$		
	error	rate	it	error	rate	it	error	rate	it
16	0.091		1	0.091		1	0.091		1
32	0.061	0.835	1	0.061	0.835	1	0.061	0.835	1
64	0.038	0.934	1	0.038	0.934	1	0.038	0.934	1
128	0.022	0.977	1	0.022	0.977	1	0.022	0.977	1
256	0.013	0.993	1	0.013	0.993	1	0.013	0.993	1

Table 5.6: Results for example with boundary layers on two sides and no projection trial space.

5.5.5 Remarks on the SPLS Approach

In this chapter, we presented an approach to solving reaction diffusion equations that utilizes graph type trial spaces. We observe that the method performs well no

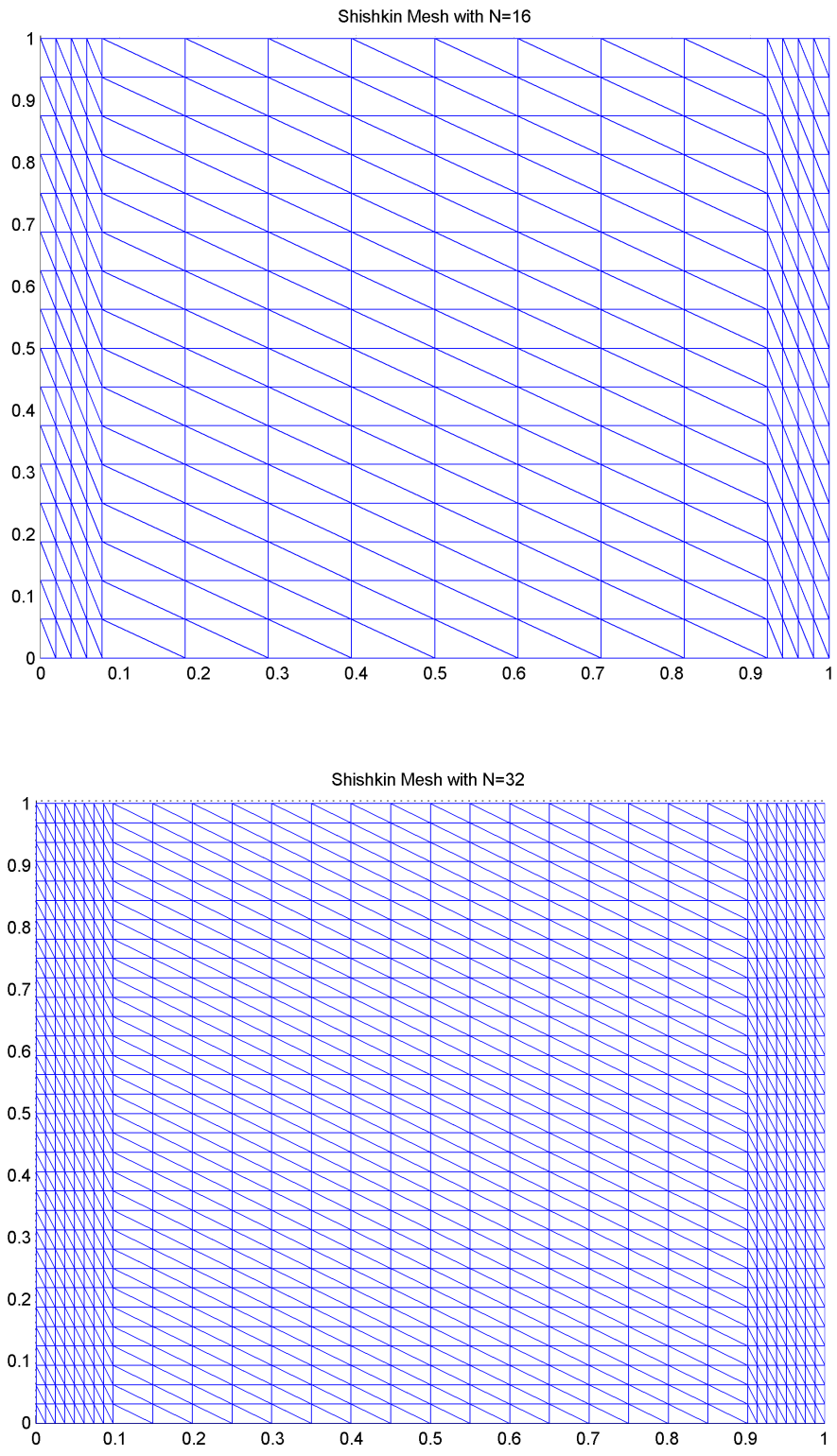

Figure 5.3: Shishkin mesh used for example with boundary layers at $x = 0$ and $x = 1$ for $N = 16, 32$.

$\mathcal{M}_h = R_h^{\text{orth}} BV_h$

N	$\varepsilon = 1$			$\varepsilon = 10^{-2}$			$\varepsilon = 10^{-4}$		
	error	rate	it	error	rate	it	error	rate	it
16	0.0015		2	0.0110		3	0.050		4
32	0.0005	2.378	2	0.0032	2.573	4	0.025	1.469	7
64	0.0002	2.126	2	0.0011	2.149	4	0.010	1.780	12
128	5.6e-05	1.976	2	0.0004	1.982	4	0.004	1.941	19
256	1.9e-05	1.882	2	0.0001	1.884	4	0.001	1.988	29
N	$\varepsilon = 10^{-8}$			$\varepsilon = 10^{-12}$			$\varepsilon = 10^{-16}$		
	error	rate	it	error	rate	it	error	rate	it
16	0.050		3	0.050		4	0.050		5
32	0.025	1.464	6	0.025	1.464	7	0.025	1.464	9
64	0.010	1.777	9	0.010	1.779	12	0.010	1.779	14
128	0.004	1.932	12	0.004	1.942	19	0.004	1.942	23
256	0.001	1.962	17	0.001	1.989	29	0.001	1.990	41

Table 5.7: Results for example with boundary layers on two sides and orthogonal projection.

$\mathcal{M}_h = R_h^{\text{lump}} BV_h$

N	$\varepsilon = 1$			$\varepsilon = 10^{-2}$			$\varepsilon = 10^{-4}$		
	error	rate	it	error	rate	it	error	rate	it
16	0.0024		3	0.0161		4	0.068		3
32	0.0008	2.202	3	0.0051	2.442	5	0.038	1.226	4
64	0.0003	2.032	3	0.0017	2.131	5	0.016	1.637	6
128	0.0001	1.928	3	0.0006	1.972	5	0.006	1.900	8
256	3.8e-05	1.857	3	0.0002	1.974	6	0.002	1.911	10
N	$\varepsilon = 10^{-8}$			$\varepsilon = 10^{-12}$			$\varepsilon = 10^{-16}$		
	error	rate	it	error	rate	it	error	rate	it
16	0.067		4	0.067		4	0.067		5
32	0.038	1.231	7	0.038	1.231	8	0.038	1.231	9
64	0.016	1.650	10	0.016	1.650	14	0.016	1.650	15
128	0.006	1.947	15	0.006	1.948	21	0.006	1.948	28
256	0.002	2.028	21	0.002	2.035	33	0.002	2.035	44

Table 5.8: Results for example with boundary layers on two sides and lump projection.

matter the size of ε, and we obtain convergence rates of $\mathcal{O}((N^{-1} \ln N)^2)$ using just piecewise linear approximation and the projection type trial spaces. These rates of

convergence are similar to those obtained by Lin and Stynes in [71], where a mixed finite element approach was taken involving $H(\mathrm{div};\Omega)$ conforming spaces. One of the main advantages of the SPLS approach presented in this chapter is that the implementation, compared with their approach, is simpler due to the use of piecewise linear spaces. Also, when using the projection type spaces we obtain $\mathcal{O}((N^{-1}\ln N)^2)$ without the need to post-process the solution, which is the approach taken in [67] to obtain higher order convergence for $\varepsilon^{1/4}\nabla u$ in the L^2 norm.

Chapter 6

SPLS FOR TIME-HARMONIC MAXWELL'S EQUATIONS

Efficient approximation of the time-harmonic Maxwell equations is of significant importance to practical applications, such as analog signal packages. For Maxwell's equations, one needs a robust methodology independent of frequency. In this chapter, we will apply the SPLS framework to the time-harmonic Maxwell equations. In the process, an efficient iterative solver is constructed that is able to simultaneously approximate the electric and magnetic field solutions to the equations. Furthermore, standard finite element spaces are utilized to obtain a simple to implement version of Algorithm 2.4.1. This chapter is published in [16].

Let $\Omega \subset \mathbb{R}^3$ be a polyhedral domain with boundary Γ. Consider two positive functions

$$\varepsilon, \mu \in L^\infty(\Omega), \qquad \varepsilon_1 > \varepsilon \geq \varepsilon_0 > 0, \qquad \mu_1 > \mu \geq \mu_0 > 0 \qquad \text{in } \Omega.$$

We seek a solution to the time-harmonic Maxwell problem given by the equations

$$\nabla \times \mathbf{h} - \lambda\, \varepsilon\, \mathbf{e} = \mathbf{j} \qquad \text{in } \Omega, \tag{6.0.1a}$$

$$\nabla \times \mathbf{e} + \lambda\, \mu\, \mathbf{h} = \mathbf{m} \qquad \text{in } \Omega, \tag{6.0.1b}$$

$$(\mu\, \mathbf{h}) \cdot \mathbf{n} = 0 \qquad \text{on } \Gamma, \tag{6.0.1c}$$

$$\mathbf{e} \times \mathbf{n} = \mathbf{0} \qquad \text{on } \Gamma, \tag{6.0.1d}$$

where $\mathbf{h}$ and $\mathbf{e}$ are the magnetic and electric vector fields, $\mathbf{j}$ and $\mathbf{m}$ are the electric and magnetic current densities, ε is the electric permittivity, μ is the magnetic permeability, and $\lambda = -i\omega$. Here, $\omega \in \mathbb{R}$ is given and represents the frequency of propagation of the electromagnetic waves. The boundary conditions correspond to a region surrounded by a perfect conductor.

There are many methods designed to efficiently approximate (6.0.1), see [75] and the references therein. Most of these methods use curl-conforming edge elements. In [77], an interior penalty DG method based on a mixed variational formulation was introduced. Many other techniques, including adaptive methods for (6.0.1), starting with the work in [52], have been investigated in the last two decades. The SPLS approach for discretizing (6.0.1) is similar with the work of Bramble and Pasciak in [33]. We assume L^2 type spaces for the magnetic and electric vector fields and start from a natural weak formulation as presented in [33]. For discretization, we require that the test spaces be H^1-conforming with suitable boundary conditions. We depart from the Bramble-Pasciak least squares method in the way the discrete trial spaces are chosen. Namely, the discrete trial spaces are built using the action of the continuous first order differential operator B associated with problem (6.0.1). Both the no projection and projection type trial spaces will be analyzed.

In addition to the advantages that are characteristic to the SPLS method, the main contribution of the proposed discretization for the time-harmonic Maxwell equations resides in investigating the stability of the proposed families of discretization spaces. For the no projection discrete trial space, we investigate the numerical stability of the proposed family of discrete spaces, see Section 6.4. For the projection type trial space, we prove that the stability is at least as good as the stability in the no projection case, see Section 2.3.2 and Theorem 6.4.2.

The chapter is organized as follows. In Section 6.1, we will review some basic material of the Sobolev spaces needed for the analysis of the problem. In Section 6.2, we discuss the weak variational formulation of (6.0.1) and the connection between the weak formulation and the original formulation. In Section 6.3, we apply the abstract discretization theory of Chapter 2 to the operator B associated with Maxwell's equations. Starting with a common test space, we propose three ways to choose the trial space and investigate the stability and approximability of the corresponding pairs of discrete spaces. Numerical results to support the SPLS approach to the discretization of Maxwell's equations are presented in Section 6.5.

6.1 Notation and Background

Throughout this chapter, if V is a space of scalar valued functions, then $\mathbf{V} := V^3$ will be the vector-valued product space endowed with the product topology. Also, we will denote the inner product and norm on $L^2(\Omega)$ and $\mathbf{L}^2(\Omega)$ by $(\cdot,\cdot)$ and $\|\cdot\|$, respectively. We define the three Sobolev spaces

$$H^1(\Omega) := \{u \in L^2(\Omega) \,|\, \nabla u \in L^2(\Omega)\},$$
$$H(\mathrm{div};\Omega) := \{\mathbf{u} \in \mathbf{L}^2(\Omega) \,|\, \nabla \cdot \mathbf{u} \in L^2(\Omega)\},$$
$$H(\mathrm{curl};\Omega) := \{\mathbf{u} \in \mathbf{L}^2(\Omega) \,|\, \nabla \times \mathbf{u} \in \mathbf{L}^3(\Omega)\}.$$

We also define $H_0(\mathrm{div};\Omega)$ as the closure of $\boldsymbol{\mathcal{C}}_0^\infty(\Omega)$ in the norm

$$\|\mathbf{u}\|_{H(\mathrm{div};\Omega)} := \left(\|\mathbf{u}\|^2 + \|\nabla \cdot \mathbf{u}\|^2\right)^{1/2},$$

and $H_0(\mathrm{curl};\Omega)$ as the closure of $\boldsymbol{\mathcal{C}}_0^\infty(\Omega)$ in the norm

$$\|\mathbf{u}\|_{H(\mathrm{curl};\Omega)} := \left(\|\mathbf{u}\|^2 + \|\nabla \times \mathbf{u}\|^2\right)^{1/2}.$$

For more details on these Sobolev spaces, we refer to [54, 75].

The representation of the dual space of $H_0^1(\Omega)$ as a space of distributions will be denoted $H^{-1}(\Omega)$. Similarly, we will need the space $\widetilde{H}^{-1}(\Omega) := H^1(\Omega)^*$ which is not a space of distributions. We will also use the following Gelfand triples

$$H_0^1(\Omega) \subset L^2(\Omega) \subset H^{-1}(\Omega), \qquad \text{and} \qquad H^1(\Omega) \subset L^2(\Omega) \subset \widetilde{H}^{-1}(\Omega). \tag{6.1.1}$$

Lastly, duality products on $V^* \times V$ will be denoted using $\langle\cdot,\cdot\rangle$, unsubscripted unless otherwise needed.

6.2 Variational Formulation of the Problem

In this section, we follow the approach of [33] and derive the variational formulation of (6.0.1). By writing all the complex functions that appear in (6.0.1) using the real and imaginary parts, one may conclude that both the real and imaginary parts satisfy a related real-valued problem to the (6.0.1) system, see Remark 2.2 of [33].

More specifically, $(\mathbf{h}, \mathbf{e}, \mathbf{j}, \mathbf{m})$ satisfies the (6.0.1) system with $\lambda = -i\omega$ if and only if $(\Re(\mathbf{h}), \Im(\mathbf{e}), \Re(\mathbf{j}), \Im(\mathbf{m}))$ and $(-\Im(\mathbf{h}), \Re(\mathbf{e}), -\Im(\mathbf{j}), \Re(\mathbf{m}))$ satisfy

$$\nabla \times \mathbf{h} - \omega\, \varepsilon\, \mathbf{e} = \mathbf{j} \qquad \text{in } \Omega, \tag{6.2.1a}$$

$$\nabla \times \mathbf{e} - \omega\, \mu\, \mathbf{h} = \mathbf{m} \qquad \text{in } \Omega, \tag{6.2.1b}$$

$$(\mu\, \mathbf{h}) \cdot \mathbf{n} = 0 \qquad \text{on } \Gamma, \tag{6.2.1c}$$

$$\mathbf{e} \times \mathbf{n} = \mathbf{0} \qquad \text{on } \Gamma. \tag{6.2.1d}$$

Thus, we can restrict our considerations only to real functions and a real parameter ω. Without loss of generality, we can assume that all Hilbert spaces in this chapter are real Hilbert spaces.

Strong solutions of (6.2.1) are thought of in the following spaces:

$$\begin{aligned} \mathbf{h} \in \mathbf{X}_1(\mu) &:= H(\mathrm{curl}; \Omega) \cap \{\mathbf{h} \in \mathbf{L}^2(\Omega) \,|\, \nabla \cdot (\mu\, \mathbf{h}) \in L^2(\Omega),\ (\mu\, \mathbf{h}) \cdot \mathbf{n} = 0\}, \\ \mathbf{e} \in \mathbf{X}_2(\varepsilon) &:= H_0(\mathrm{curl}; \Omega) \cap \{\mathbf{e} \in \mathbf{L}^2(\Omega) \,|\, \nabla \cdot (\varepsilon\, \mathbf{e}) \in L^2(\Omega)\}. \end{aligned}$$

If $(\mathbf{h}, \mathbf{e})$ is a strong solution to (6.2.1), then we necessarily have

$$\nabla \cdot (\varepsilon\, \mathbf{e}) = -\omega^{-1} \nabla \cdot \mathbf{j}, \qquad \nabla \cdot (\mu\, \mathbf{h}) = -\omega^{-1} \nabla \cdot \mathbf{m}. \tag{6.2.2}$$

For this reason, it is natural to assume $\mathbf{j}$ and $\mathbf{m}$ satisfy

$$\mathbf{j} \in H(\mathrm{div}; \Omega), \qquad \mathbf{m} \in H_0(\mathrm{div}; \Omega).$$

Lastly, we assume that ω is not a Maxwell eigenvalue, i.e., if $\mathbf{j} = \mathbf{0}$ and $\mathbf{m} = \mathbf{0}$, then the only solution of (6.2.1) in $\mathbf{X}_1(\mu) \cap \mathbf{X}_2(\varepsilon)$ is $\mathbf{h} = \mathbf{0}$, $\mathbf{e} = \mathbf{0}$.

To define the global differential operators associated with (6.2.1), we first define the following spaces and inner products:

$$L^2_\beta(\Omega) := \{u : \Omega \to \mathbb{R} \,|\, \beta^{1/2}\, u \in L^2(\Omega)\}, \quad (u, v)_\beta := (\beta\, u, v), \quad \beta \in \{\mu, \varepsilon\}.$$

Next, following the notation of [33], we consider the following four operators:

$$\mathbf{curl}_1 : \mathbf{L}^2_\mu(\Omega) \to \mathbf{H}^{-1}(\Omega) \qquad \langle \mathbf{curl}_1 \mathbf{h}, \boldsymbol{\phi} \rangle := (\mathbf{h}, \mu^{-1} \nabla \times \boldsymbol{\phi})_\mu = (\mathbf{h}, \nabla \times \boldsymbol{\phi}),\ \boldsymbol{\phi} \in \mathbf{H}^1_0(\Omega),$$

$$\mathbf{curl}_2 : \mathbf{L}^2_\varepsilon(\Omega) \to \widetilde{\mathbf{H}}^{-1}(\Omega) \qquad \langle \mathbf{curl}_2 \mathbf{e}, \boldsymbol{\psi} \rangle := (\mathbf{e}, \varepsilon^{-1}\nabla \times \boldsymbol{\psi})_\varepsilon = (\mathbf{e}, \nabla \times \boldsymbol{\psi}),\ \boldsymbol{\psi} \in \mathbf{H}^1(\Omega),$$
$$\mathrm{div}_{1,\mu} : \mathbf{L}^2_\mu(\Omega) \to \widetilde{H}^{-1}(\Omega) \qquad \langle \mathrm{div}_{1,\mu}\mathbf{h}.\psi \rangle := -(\mathbf{h}, \nabla\psi)_\mu = -(\mu\,\mathbf{h}, \nabla\psi),\ \psi \in H^1(\Omega),$$
$$\mathrm{div}_{2,\varepsilon} : \mathbf{L}^2_\varepsilon(\Omega) \to H^{-1}(\Omega) \qquad \langle \mathrm{div}_{2,\varepsilon}\mathbf{e}, \phi \rangle := -(\mathbf{e}, \nabla\phi)_\varepsilon = -(\varepsilon\,\mathbf{e}, \nabla\phi),\ \phi \in H^1_0(\Omega).$$

Note that $\mathbf{curl}_1$ is the distributional curl acting on elements of $\mathbf{L}^2_\mu(\Omega) \equiv \mathbf{L}^2(\Omega)$ and that $\mathrm{div}_{2,\varepsilon} = \nabla \cdot (\varepsilon\,\cdot)$, with the divergence operator taken in the sense of distributions acting on $\mathbf{L}^2(\Omega)$ vector fields. The two remaining operators cannot be understood as distributional differentiation operators. We also need the two multiplication operators

$$\mu : \mathbf{L}^2_\mu(\Omega) \to \mathbf{H}^{-1}(\Omega) \qquad \mu\,\mathbf{e} := (\mathbf{e}, \cdot)_\mu = (\mu\,\mathbf{e}, \cdot) : \mathbf{H}^1_0(\Omega) \to \mathbb{R},$$
$$\varepsilon : \mathbf{L}^2_\varepsilon(\Omega) \to \widetilde{\mathbf{H}}^{-1}(\Omega) \qquad \varepsilon\,\mathbf{e} := (\mathbf{e}, \cdot)_\varepsilon = (\varepsilon\,\mathbf{e}, \cdot) : \mathbf{H}^1(\Omega) \to \mathbb{R}.$$

These are multiplication operators followed by compact inclusions in the corresponding right side of the Gelfand triples in (6.1.1).

We now define the global operators associated with (6.2.1). We let

$$Q := \mathbf{L}^2_\mu(\Omega) \times \mathbf{L}^2_\varepsilon(\Omega) \equiv Q^*,$$

endowed with its weighted product norm and inner product and

$$V := \mathbf{H}^1_0(\Omega) \times \mathbf{H}^1(\Omega) \times H^1(\Omega) \times H^1_0(\Omega),$$

endowed with the norm

$$|(\boldsymbol{\phi}, \boldsymbol{\psi}, \psi, \phi)|^2_V := \|\nabla\boldsymbol{\phi}\|^2 + \|\boldsymbol{\psi}\|^2 + \|\nabla\boldsymbol{\psi}\|^2 + \|\psi\|^2 + \|\nabla\psi\|^2 + \|\nabla\phi\|^2.$$

The inner product that induces the norm on V is denoted by $a(\cdot,\cdot)$. Let $\mathbf{v} = (\boldsymbol{\phi}, \boldsymbol{\psi}, \psi, \phi)$ and $\mathbf{p} = (\mathbf{h}, \mathbf{e})$. We define the bilinear form $b(\cdot,\cdot) : V \times Q \to \mathbb{R}$ by

$$\begin{aligned} b(\mathbf{v}, \mathbf{p}) &= b((\boldsymbol{\phi}, \boldsymbol{\psi}, \psi, \phi), (\mathbf{h}, \mathbf{e})) \\ &:= (\nabla\times\boldsymbol{\phi} - \omega\mu\boldsymbol{\psi} - \mu\nabla\psi,\ \mathbf{h}) + (-\omega\varepsilon\boldsymbol{\phi} + \nabla\times\boldsymbol{\psi} - \varepsilon\nabla\phi,\ \mathbf{e}). \end{aligned}$$

With the form $b(\cdot,\cdot)$, we associate the operators $B : V \to Q$ and $B^* : Q \to V^*$ given by the matrix operators

$$B := \begin{bmatrix} \mu^{-1}\nabla\times & -\omega & -\nabla & 0 \\ -\omega & \varepsilon^{-1}\nabla\times & 0 & -\nabla \end{bmatrix}, \qquad B^* := \begin{bmatrix} \mathbf{curl}_1 & -\omega\varepsilon \\ -\omega\mu & \mathbf{curl}_2 \\ \mathrm{div}_{1,\mu} & 0 \\ 0 & \mathrm{div}_{2,\varepsilon} \end{bmatrix}.$$

Following Section 2 of [33], the weak form of equations (6.2.1), incorporating the information of (6.2.2), is

$$B^* \begin{bmatrix} \mathbf{h} \\ \mathbf{e} \end{bmatrix} = \mathbf{F}, \tag{6.2.3}$$

where

$$\mathbf{F} := \begin{bmatrix} \mathbf{j} \\ \mathbf{m} \\ -\omega^{-1}\nabla\cdot\mathbf{m} \\ -\omega^{-1}\nabla\cdot\mathbf{j} \end{bmatrix}.$$

All boundary conditions in (6.2.1) are hidden in the dualization process for the definition of B^* (see [33] for details). The corresponding variational formulation of (6.2.3) is: Find $\mathbf{p} = (\mathbf{h}, \mathbf{e}) \in Q$ such that

$$b(\mathbf{v}, \mathbf{p}) = \langle \mathbf{F}, \mathbf{v} \rangle \quad \text{for all } \mathbf{v} = (\boldsymbol{\phi}, \boldsymbol{\psi}, \psi, \phi) \in V. \tag{6.2.4}$$

The relevant information that we need for solving (6.2.1) via the weak formulation (6.2.3) or (6.2.4) is concentrated in the following theorem proved in [33].

Theorem 6.2.1. *Assume that ω is not a Maxwell eigenvalue. The operator $B^* : Q \to V^*$ is injective and has closed range. If $\mathbf{j} \in H(\mathrm{div};\Omega)$ and $\mathbf{m} \in H_0(\mathrm{div};\Omega)$, then $(\mathbf{j}, \mathbf{m}, -\omega^{-1}\nabla\cdot\mathbf{m}, -\omega^{-1}\nabla\cdot\mathbf{j})^\top$ is in the range of B^*, and the unique solution of (6.2.3), or (6.2.4), is a strong solution of (6.2.1).*

The previous theorem allows us to approximate the solution of (6.2.1) by discretizing the mixed weak formulation (6.2.4) using the SPLS approach.

6.3 Discretization for Maxwell's Equations

In this section, we apply the SPLS discretization theory to the operator B associated with Maxwell's equations. We recall that $B : V \to Q$ is given by the matrix operator

$$B := \begin{bmatrix} \mu^{-1}\nabla\times & -\omega & -\nabla & 0 \\ -\omega & \varepsilon^{-1}\nabla\times & 0 & -\nabla \end{bmatrix}.$$

Let $\mathcal{T}_h$ be a shape regular tetrahedralization of Ω, and consider the lowest order finite element spaces

$$H_h := \{u_h \in \mathcal{C}(\overline{\Omega}) \mid u_h|_K \in \mathbb{P}_1 \quad \forall K \in \mathcal{T}_h\},$$

and

$$H_h^0 := H_h \cap H_0^1(\Omega).$$

We define the product space

$$V_h := \mathbf{H}_h^0 \times \mathbf{H}_h \times H_h \times H_h^0, \tag{6.3.1}$$

as the test space.

Note that

$$B(V_h) \subset \mathcal{P}_0(\mathcal{T}_h)^6 + H_h^6 \subset \mathcal{P}_1(\mathcal{T}_h)^6,$$

where $\mathcal{P}_k(\mathcal{T}_h)$ is the space of discontinuous piecewise $\mathcal{P}_k$ functions (polynomials of degree k) on the tetrahedral partition $\mathcal{T}_h$. For the remainder of this chapter, we will assume that the coefficients μ, ε are continuous functions. The case where μ, ε are discontinuous will be analyzed in the near future.

6.3.1 No Projection Trial Space

As outlined in Section 2.3.1, we define the no projection type trial space as

$$\mathcal{M}_h = B(V_h),$$

where the inner product on $\mathcal{M}_h$ is taken to be the inner product on Q. Recall that the inner product on Q contains the weight μ for the first three components and the

weight ε for the last three. A discrete inf $-$ sup condition for the pair $(V_h, \mathcal{M}_h)$ holds as described in the abstract case, see (2.3.1). The approximability in this case is given by (2.3.2). In the next section, we will address the stability of the pair $(V_h, \mathcal{M}_h)$.

6.3.2 Orthogonal Projection Space

We start with defining the space $\tilde{\mathcal{M}}_h := H_h^6$ equipped with the inner product on Q. With the orthogonal projection

$$R_h^{\text{orth}} : Q \longrightarrow H_h^6,$$

we define the orthogonal projection trial space by

$$\mathcal{M}_h := R_h^{\text{orth}}(B(V_h)) \subset H_h^6.$$

A discrete inf $-$ sup condition for the pair $(V_h, \mathcal{M}_h)$ holds as described in Section 2.3.2 and an estimate for approximability is given by (2.5.1). The (numerical) stability of the pair $(V_h, \mathcal{M}_h)$ will be analyzed via the estimate (2.3.5) in the next section.

6.3.3 Lump Projection Space

For this trial space, we start with defining $\tilde{\mathcal{M}}_h := H_h^6$ as in the previous section, but equip $\tilde{\mathcal{M}}_h$ with a different inner product related with lumping the mass matrix. To be more precise, given the nodal Lagrange basis $\{\varphi_1, \ldots, \varphi_N\}$ of H_h and the function $\beta \in \{\varepsilon, \mu\}$, the $\beta-$weighted lumped mass matrix is the diagonal matrix with elements

$$d_i := \int_\Omega \beta \varphi_i = \sum_{j=1}^{N} \int_\Omega \beta \, \varphi_i \, \varphi_j. \tag{6.3.2}$$

For functions $u_h, v_h \in H_h$, the associated inner product is defined by

$$(u_h, v_h)_{\beta,\text{lump}} = \left(\sum_{j=1}^{N} u_j \varphi_j, \sum_{i=1}^{N} v_i \varphi_i \right)_{\beta,\text{lump}} := \sum_{i=1}^{N} u_i \, d_i \, v_i,$$

where d_i is given in (6.3.2). The inner product $(\cdot, \cdot)_{Q,\text{lump}}$ in $\tilde{\mathcal{M}}_h$ is then defined by lumping the mass matrices of each of the six components. The lump orthogonal

projection with respect to the inner product $(\cdot,\cdot)_{Q,\text{lump}}$ is denoted $R_h^{\text{lump}} : Q \longrightarrow H_h^6$. The corresponding trial space is defined as

$$\mathcal{M}_h := R_h^{\text{lump}}(B(V_h)) \subset H_h^6.$$

As with the other trial spaces, we discuss the discrete stability for the pair $(V_h, \mathcal{M}_h)$ in the next section.

6.4 Stability and Numerical Stability of the Proposed Discretizations

From Proposition 2.3.2, the stability of the trial spaces defined in Sections 6.3.2 and 6.3.3 follows from the stability of the no projection trial space of Section 6.3.1 if condition (2.3.5) is satisfied. The investigation of stability in the no projection case is equivalent with building a stable right inverse for the operator B restricted to the space V_h. This problem is difficult even when B is a simpler first order differential operator, such as the divergence operator [4, 84, 85]. To this end, we will estimate the $\inf - \sup$ constant m_h numerically for the pair $(V_h, B(V_h))$. While we believe that stability can be shown under certain conditions for the mesh $\mathcal{T}_h$, an investigation of such conditions is not explored in this thesis.

In order to estimate m_h for the pair $(V_h, B(V_h))$, we use that m_h is the square root of the Schur complement S_h associated with the discrete saddle point system (2.2.4), see e.g., [7]. The action of S_h can be computed by slightly modifying Algorithm 2.4.1, and a standard power method for the Schur complement can be used to obtain estimates for the eigenvalues of S_h. We consider $\mathcal{T}_h$ for the unit cube obtained by uniform refinement, splitting each cube into eight cubes of half side and then splitting each small cube into six tetrahedra by a standard procedure. To see the behavior of $m_h = m_h(\omega)$, we applied this technique for four levels of uniform refinement, $\varepsilon = \mu = 1$, and various ω.

From Table 6.1, we see the stability of m_h depends on ω, but even in the worse case of $\omega = 1$ we still have $m_h \geq O(h)$, where $h = 2^{-k}$ and k is the level of refinement. For small values of $\omega > 0$, we notice stability with respect to both h and ω. In order

level	$\omega = 1$	$\omega = 2$	$\omega = 4$	$\omega = 16$	$\omega = 64$	$\omega = 256$
	m_h	m_h	m_h	m_h	m_h	m_h
1	0.1003	0.1976	0.1145	1.6253	6.4516	25.7893
2	0.0697	0.1172	1.1185	0.5075	2.858	10.9398
3	0.0465	0.0766	1.0072	1.1426	2.5893	14.7614
4	0.0271	0.9991	1.0150	1.2425	3.6897	17.2878

Table 6.1: Approximations of $m_h(\omega)$ for the no projection trial space.

to discuss stability for the other choices of trial spaces, we will prove estimate (2.3.5). Then, by Proposition 2.3.2, the stability is at least as good as the stability for the no projection case.

Let $\{\varphi_1, \ldots, \varphi_N\}$ be the nodal Lagrange basis of H_h and $M_{h,\beta}$ be the mass matrix with entries $(\varphi_i, \varphi_j)_\beta$, where $\beta \in \{\varepsilon, \mu\}$. We will also let $D_{h,\beta}$ be the diagonal matrix with entries

$$d_i = \int_\Omega \beta \varphi_i.$$

Since the mesh $\mathcal{T}_h$ is uniform, we can assume that $(1, \varphi_i) \approx h^3$. Also, without loss of generality we assume

$$0 < \beta_0 < \beta < \beta_1 \text{ in } \Omega.$$

In what follows, $\langle \cdot, \cdot \rangle_e$ will denote the standard euclidean inner product on $\mathbb{R}^N$.

Lemma 6.4.1. *Under the assumptions from this section,*

$$\langle M_{h,\beta}\, \boldsymbol{\gamma}, \boldsymbol{\gamma} \rangle_e \leq c\, \frac{\beta_1}{\beta_0} \langle D_{h,\beta}\, \boldsymbol{\gamma}, \boldsymbol{\gamma} \rangle_e \quad \text{for all } \boldsymbol{\gamma} \in \mathbb{R}^N, \tag{6.4.1}$$

with a constant c independent of h. Consequently,

$$\langle M_{h,\beta}^{-1}\, \boldsymbol{\gamma}, \boldsymbol{\gamma} \rangle_e \geq c\, \frac{\beta_0}{\beta_1} \langle D_{h,\beta}^{-1}\, \boldsymbol{\gamma}, \boldsymbol{\gamma} \rangle_e \quad \text{for all } \boldsymbol{\gamma} \in \mathbb{R}^N, \tag{6.4.2}$$

with a constant c independent of h.

Proof. For any $\boldsymbol{\gamma} = (\gamma_1, \gamma_2, \ldots, \gamma_N) \in \mathbb{R}^N$, let

$$p_h := \sum_{j=1}^{N} \gamma_j\, \varphi_j.$$

Using that $\|p_h\| \approx h^3 \sum_{i=1}^{N} \gamma_i^2$, we obtain

$$\langle M_{h,\beta}\boldsymbol{\gamma}, \boldsymbol{\gamma}\rangle_e = \sum_{i,j=1}^{N} \gamma_i\gamma_j(\varphi_i, \varphi_j)_\beta = \|p_h\|_\beta^2 \le \beta_1 \|p_h\|^2 \le c\beta_1 h^3 \sum_{i=1}^{N} \gamma_i^2.$$

Since $h^3 \approx (1, \varphi_i)$ and $\beta_0(1, \varphi) \le (1, \varphi)_\beta$, it follows

$$\langle M_{h,\beta}\boldsymbol{\gamma}, \boldsymbol{\gamma}\rangle_e \le c\frac{\beta_1}{\beta_0} \sum_{i=1}^{N} (1, \varphi_i)_\beta \gamma_i^2 = c\,\frac{\beta_1}{\beta_0} \langle D_{h,\beta}\,\boldsymbol{\gamma}, \boldsymbol{\gamma}\rangle_e,$$

which proves (6.4.1). Estimate (6.4.2) follows. □

Theorem 6.4.2. *If $R_h^{\mathrm{orth}} : Q \longrightarrow H_h^6$ is the orthogonal projection, then*

$$\|R_h^{\mathrm{orth}}\mathbf{q}_h\|_Q \ge \tilde{c}\,\|\mathbf{q}_h\|_Q \quad \text{for all } \mathbf{q}_h \in B(V_h), \tag{6.4.3}$$

with a constant $\tilde{c}$ independent of h.

Proof. Since $B(V_h) \subset \mathcal{P}_0(\mathcal{T}_h)^6 + H_h^6 \subset \mathcal{P}_1(\mathcal{T}_h)^6$, it suffices to prove that the estimate (6.4.3) holds component-wise on $\mathcal{P}_1(\mathcal{T}_h)$, i.e.,

$$\|Q_{h,\beta}\, q_h\|_\beta \ge \tilde{c}\,\|q_h\|_\beta \quad \text{for all } q_h \in \mathcal{P}_1(\mathcal{T}_h),$$

where $Q_{h,\beta} : L^2_\beta(\Omega) \to H_h$ is the orthogonal projection with respect to the L^2_β-inner product. In what follows, the constant c that appears is generic and may be different at different occurrences, but is always independent of h. Let $q_h \in \mathcal{P}_1(\mathcal{T}_h)$ and $\tilde{\mathbf{q}}_h$ denote the dual vector of q_h, i.e.,

$$\tilde{\mathbf{q}_h} = ((q_h, \varphi_1)_\beta, \cdots, (q_h, \varphi_N)_\beta)^T.$$

We define $\boldsymbol{\alpha} = (\alpha_1, \cdots, \alpha_N)^T \in \mathbb{R}^N$ such that

$$Q_{h,\beta}\, q_h = \sum_{i=1}^{N} \alpha_i\, \varphi_i.$$

Hence, $\boldsymbol{\alpha} = M_{h,\beta}^{-1}\,\tilde{\mathbf{q}}_h$. Using (6.4.2), we obtain

$$\|Q_{h,\beta}\, q_h\|_\beta^2 = \langle M_{h,\beta}\alpha, \alpha\rangle_e = \langle M_{h,\beta}^{-1}\,\tilde{q_h}, \tilde{q_h}\rangle_e \ge c\,\frac{\beta_0}{\beta_1} \langle D_{h,\beta}^{-1}\,\tilde{q_h}, \tilde{q_h}\rangle_e. \tag{6.4.4}$$

From the definition of $D_{h,\beta}$, it follows that

$$\langle D_{h,\beta}^{-1}\,\tilde{\mathbf{q}}_h, \tilde{\mathbf{q}}_h\rangle_e = \sum_{i=1}^{N} \frac{(q_h,\varphi_i)_\beta^2}{(1,\varphi_i)_\beta} \geq \frac{\beta_0^2}{\beta_1} \sum_{i=1}^{N} \frac{(q_h,\varphi_i)^2}{(1,\varphi_i)}. \tag{6.4.5}$$

From the uniformity of the mesh, the spectral properties of the local mass matrix, and the fact that q_h is linear on all tetrahedra of $\mathcal{T}_h$, we obtain

$$\sum_{i=1}^{N} \frac{(q_h,\varphi_i)^2}{(1,\varphi_i)} \geq c\|q_h\|^2 \geq c\frac{1}{\beta_1}\|q_h\|_\beta^2. \tag{6.4.6}$$

Combining (6.4.4), (6.4.5), and (6.4.6) gives us

$$\|Q_{h,\beta}\,q_h\|_\beta \geq c\left(\frac{\beta_0}{\beta_1}\right)^{3/2}\|q_h\|_\beta,$$

as desired. □

A similar result can be obtained for the lump projection.

Theorem 6.4.3. *If $R_h^{\text{lump}} : Q \longrightarrow H_h^6$ is the lump projection, then*

$$\|R_h^{\text{lump}}\mathbf{q}_h\|_{Q,\text{lump}} \geq \tilde{c}\,\|\mathbf{q}_h\|_Q \quad \textit{for all } \mathbf{q}_h \in B(V_h), \tag{6.4.7}$$

with a constant $\tilde{c}$ independent of h.

Proof. Similar to the proof of Theorem 6.4.2, it suffices to prove that the estimate (6.4.7) holds component-wise on $\mathcal{P}_1(\mathcal{T}_h)$, i.e.,

$$\|Q_{h,\beta}^{\text{lump}}\,q_h\|_{\beta,\text{lump}} \geq \tilde{c}\,\|q_h\|_\beta \quad \text{for all } q_h \in \mathcal{P}_1(\mathcal{T}_h),$$

where $Q_{h,\beta}^{\text{lump}} : L_\beta^2(\Omega) \to H_h$ is the orthogonal projection with respect to the lumped L_β^2-inner product. Let $q_h \in \mathcal{P}_1(\mathcal{T}_h)$. Using the definition of the lumped inner product, we obtain

$$\begin{aligned}
\|Q_{h,\beta}^{\text{lump}}\,q_h\|_{\beta,\text{lump}}^2 &= \left(\sum_{j=1}^{N}\frac{(q_h,\varphi_j)_\beta}{(1,\varphi_j)_\beta}\varphi_j, \sum_{i=1}^{N}\frac{(q_h,\varphi_i)_\beta}{(1,\varphi_i)_\beta}\varphi_i\right)_{\beta,\text{lump}} \\
&= \sum_{j=1}^{N}\frac{(q_h,\varphi_j)_\beta^2}{(1,\varphi_j)_\beta} \\
&\geq \frac{\beta_0^2}{\beta_1}\sum_{j=1}^{N}\frac{(q_h,\varphi_j)^2}{(1,\varphi_j)}.
\end{aligned}$$

Combining the above estimate with (6.4.6) gives us

$$\|Q_{h,\beta}^{\text{lump}} q_h\|_{\beta,\text{lump}} \geq c\frac{\beta_0}{\beta_1}\|q_h\|_\beta.$$

□

6.5 Numerical Results

In this section, we report some numerical results when approximating the solution of (6.2.1) using the SPLS method. In all cases, we considered $\mathcal{T}_h$ to be a tetrahedralization of Ω and the test space V_h was taken to be the finite element space of piecewise linear functions as defined in (6.3.1). Algorithm 2.4.1 was implemented for all three types of trial spaces for various frequencies ω using five levels of refinement from the original coarse mesh. Since the sequence of meshes are nested, we employed the cascadic-multilevel approach that was outlined in Remark 2.4.2. The stopping criterion was based on (2.4.1) with best upper estimates for $O(\|\mathbf{p} - \mathbf{p}_h\|)$. For examples on non-convex domains, we used a zero vector as the initial guess $\mathbf{p}_0$ on each level.

6.5.1 Numerical Results on the Unit Cube

First, we discretized (6.2.1) on the unit cube with coefficients $\mu = \varepsilon = 1$. The data was chosen such that the exact solution is

$$\mathbf{h} = (x(1-x), y(1-y), z(1-z))^T,$$
$$\mathbf{e} = (y(1-y)z(1-z), x(1-x)z(1-z), y(1-y)x(1-x))^T.$$

We performed numerical tests for various values of $\omega \in \mathbb{R}$. Tables 6.2 and 6.3 show results for the no projection trial space and both projection type trial spaces for $\omega = 1$ and $\omega = 16$, respectively. Table 6.4 shows results for the lump projection trial space for $\omega = 100$. Table 6.5 displays results for the lump projection trial space and $\omega = 1000$. In regards to small ω, Table 6.6 shows results for all three types of trial spaces for $\omega = 1/1000$.

We see that the approximation for the orthogonal projection trial space is better than the no projection trial space, and the approximation using the lump projection

trial space is similar with the approximation using the orthogonal projection space for $\omega = 1, 16$. We also notice that the solver based on the lump projection performs well for large values of ω. Also, the method is robust with respect to values of ω that are small.

$\mathcal{M}_h = BV_h$

level k	$\|\mathbf{h} - \mathbf{h}_h\|$	Conv. Rate	$\|\mathbf{e} - \mathbf{e}_h\|$	Conv. Rate	# of iter
k=1	0.1192		0.0269		1
k=2	0.0561	1.09	0.0135	1.00	1
k=3	0.0265	1.08	0.0062	1.12	1
k=4	0.0127	1.07	0.0028	1.14	2
k=5	0.0064	0.98	0.0014	1.00	2

$\mathcal{M}_h = R_h^{\text{orth}} BV_h$

level k	$\|\mathbf{h} - \mathbf{h}_h\|$	Conv. Rate	$\|\mathbf{e} - \mathbf{e}_h\|$	Conv. Rate	# of iter
k=1	0.1169		0.0245		1
k=2	0.0337	1.79	0.0091	1.43	2
k=3	0.0133	1.34	0.0042	1.13	2
k=4	0.0048	1.46	0.0013	1.67	3
k=5	0.0012	1.97	0.0003	2.04	5

$\mathcal{M}_h = R_h^{\text{lump}} BV_h$

level k	$\|\mathbf{h} - \mathbf{h}_h\|$	Conv. Rate	$\|\mathbf{e} - \mathbf{e}_h\|$	Conv. Rate	# of iter
k=1	0.1202		0.0257		1
k=2	0.0416	1.53	0.0109	1.23	2
k=3	0.0132	1.65	0.0042	1.39	3
k=4	0.0044	1.58	0.0011	1.97	4
k=5	0.0013	1.80	0.0003	1.86	6

Table 6.2: Numerical results for $\omega = 1$ on unit cube.

6.5.2 Numerical Results on a 3D L-Shaped Domain

We also tested the approach on a three dimensional L-shaped domain where the $\mathbf{e}$ component of the exact solution for (6.2.1) is not smooth. More precisely, we defined

$$\Psi = (1 - x^2)(1 - y^2)(z - z^2)\ r^{2/3}\ \sin(2\theta/3),$$

$\mathcal{M}_h = BV_h$

level k	$\|\mathbf{h} - \mathbf{h}_h\|$	Conv. Rate	$\|\mathbf{e} - \mathbf{e}_h\|$	Conv. Rate	# of iter
k=1	0.1163		0.0361		3
k=2	0.0651	0.84	0.0314	0.20	6
k=3	0.0396	0.71	0.0209	0.59	8
k=4	0.0236	0.75	0.0215	0.74	9
k=5	0.0136	0.80	0.0069	0.85	13

$\mathcal{M}_h = R_h^{\text{orth}} BV_h$

level k	$\|\mathbf{h} - \mathbf{h}_h\|$	Conv. Rate	$\|\mathbf{e} - \mathbf{e}_h\|$	Conv. Rate	# of iter
k=1	0.0084		0.0226		3
k=2	0.0067	0.34	0.0155	0.54	7
k=3	0.0054	0.30	0.0077	1.01	9
k=4	0.0027	0.99	0.0026	1.56	12
k=5	0.0009	1.54	0.0007	1.97	18

$\mathcal{M}_h = R_h^{\text{lump}} BV_h$

level k	$\|\mathbf{h} - \mathbf{h}_h\|$	Conv. Rate	$\|\mathbf{e} - \mathbf{e}_h\|$	Conv. Rate	# of iter
k=1	0.0819		0.0287		4
k=2	0.0449	0.87	0.0227	0.34	6
k=3	0.0168	1.42	0.0107	1.09	10
k=4	0.0053	1.66	0.0027	1.98	14
k=5	0.0016	1.75	0.0008	1.75	19

Table 6.3: Numerical results for $\omega = 16$ on unit cube.

$\mathcal{M}_h = R_h^{\text{lump}} BV_h$

level k	$\|\mathbf{h} - \mathbf{h}_h\|$	Conv. Rate	$\|\mathbf{e} - \mathbf{e}_h\|$	Conv. Rate	# of iter
k=1	0.0820		0.0295		8
k=2	0.0463	0.83	0.0245	0.27	11
k=3	0.0192	1.27	0.0072	1.76	31
k=4	0.0072	1.41	0.0024	1.61	44
k=5	0.0025	1.51	0.0006	2.10	81

Table 6.4: Numerical results with lump projection, $\omega = 100$.

where (r, θ) are the polar coordinates in the xy-plane. For $\mu = 1$ and $\varepsilon = 1$, we computed the data such that the exact solution is

$$\mathbf{e} = \nabla(\Psi), \quad \text{and} \quad \mathbf{h} = (x(1-x)(1+x), y(1-y)(1+y), z(1-z))^T .$$

$\mathcal{M}_h = R_h^{\text{lump}} BV_h$

level k	$\|\mathbf{h} - \mathbf{h}_h\|$	Conv. Rate	$\|\mathbf{e} - \mathbf{e}_h\|$	Conv. Rate	# of iter
k=1	0.0820		0.0295		12
k=2	0.0463	0.82	0.0264	0.26	18
k=3	0.0192	1.27	0.0074	0.74	62
k=4	0.0073	1.40	0.0017	2.11	147
k=5	0.0027	1.45	0.0005	1.69	234

Table 6.5: Numerical results with lump projection, $\omega = 1000$.

$\mathcal{M}_h = BV_h$

level k	$\|\mathbf{h} - \mathbf{h}_h\|$	Conv. Rate	$\|\mathbf{e} - \mathbf{e}_h\|$	Conv. Rate	# of iter
k=1	30.2359		0.0218		1
k=2	9.3185	1.70	0.0152	0.52	2
k=3	2.5041	1.90	0.0054	1.50	3
k=4	0.6396	1.97	0.0028	0.94	2
k=5	0.1609	1.99	0.0015	0.95	2

$\mathcal{M}_h = R_h^{\text{orth}} BV_h$

level k	$\|\mathbf{h} - \mathbf{h}_h\|$	Conv. Rate	$\|\mathbf{e} - \mathbf{e}_h\|$	Conv. Rate	# of iter
k=1	36.0886		0.0208		4
k=2	10.2212	1.82	0.0097	1.10	6
k=3	2.5759	1.99	0.0028	1.81	6
k=4	0.6445	1.99	0.0007	1.95	5
k=5	0.1611	2.00	0.0003	1.26	4

$\mathcal{M}_h = R_h^{\text{lump}} BV_h$

level k	$\|\mathbf{h} - \mathbf{h}_h\|$	Conv. Rate	$\|\mathbf{e} - \mathbf{e}_h\|$	Conv. Rate	# of iter
k=1	36.2702		0.0220		6
k=2	10.2365	1.82	0.0122	0.85	10
k=3	2.5761	1.99	0.0034	1.85	9
k=4	0.6443	2.00	0.0009	1.92	6
k=5	0.1611	2.00	0.0003	1.38	5

Table 6.6: Numerical results for $\omega = 1/1000$ on unit cube.

Note that $\mathbf{e} \notin \mathbf{H}^1(\Omega)$. We implemented Algorithm 2.4.1 using both uniform and non-uniform refinement strategies for all three types of discrete trial spaces. The family of locally quasi-uniform meshes $\{\mathcal{T}_h\}$ used for discretization in the case of non-uniform

refinement was obtained by a graded refinement strategy using the simple coordinate transformation

$$x_j := x_j \cdot |x_j|^{-1+1/q} \qquad j = 1, 2,$$

as shown in [2, 3]. Note that if $q = 1$, we recover the case of uniform refinement. The results for uniform refinement and the no projection and orthogonal projection type trial spaces are shown in Table 6.7 for $\omega = 1$. Table 6.8 displays results for all three types of trial spaces and non-uniform refinement with $\omega = 1$. Table 6.9 shows results for the no projection type trial space and non-uniform refinement with $\omega = 10$. In the case of non-uniform refinement, the parameter q in the coordinate transformation was chosen to be $q = 0.9$ for the no projection discrete trial space and $q = 0.55$ for the projection type trial spaces. Figure 6.1 shows the mesh generated on the fifth level of refinement using both $q = 0.9$ and $q = 0.55$. From the figure, we see that $q = 0.9$ results in only a slight shift of the coodinates.

$\mathcal{M}_h = BV_h$

level k	$\|\mathbf{h} - \mathbf{h}_h\|$	Conv. Rate	$\|\mathbf{e} - \mathbf{e}_h\|$	Conv. Rate	# of iter
k=1	0.2838		0.2672		2
k=2	0.1607	0.82	0.1595	0.74	1
k=3	0.0834	0.95	0.0852	0.90	2
k=4	0.0423	0.98	0.0441	0.95	1
k=5	0.0212	1.00	0.0229	0.95	3

$\mathcal{M}_h = R_h^{\text{orth}} BV_h$

level k	$\|\mathbf{h} - \mathbf{h}_h\|$	Conv. Rate	$\|\mathbf{e} - \mathbf{e}_h\|$	Conv. Rate	# of iter
k=1	0.2465		0.2108		2
k=2	0.0765	1.69	0.0718	1.55	2
k=3	0.0225	1.77	0.0342	1.07	2
k=4	0.0062	1.86	0.0155	1.14	3
k=5	0.0021	1.59	0.0086	0.86	5

Table 6.7: Non-convex domain example with uniform refinement and $\omega = 1$.

The regularity of the $\mathbf{h}$ component of the solution is higher than the regularity of the $\mathbf{e}$ component. This is reflected in the approximation of the solution using the projection type spaces as shown in Table 6.7 and Table 6.8. Also, we obtain an order

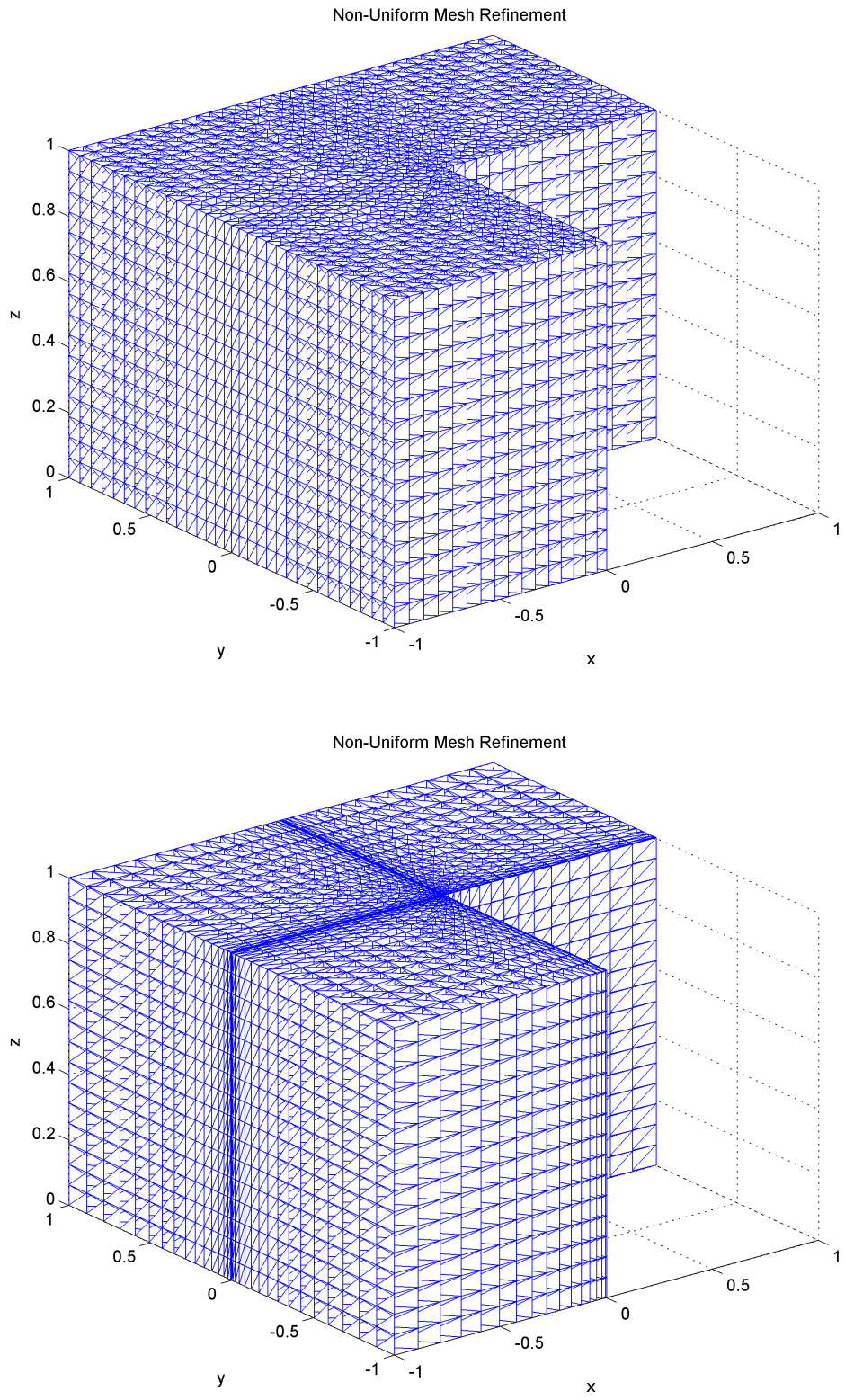

Figure 6.1: Non-uniform refinement with $q = 0.9$ (top) and $q = 0.55$ (bottom).

$\mathcal{M}_h = BV_h$

level k	$\|\mathbf{h} - \mathbf{h}_h\|$	Conv. Rate	$\|\mathbf{e} - \mathbf{e}_h\|$	Conv. Rate	# of iter
k=1	0.2783		0.2699		2
k=2	0.1634	0.77	0.1628	0.73	3
k=3	0.0862	0.92	0.0872	0.90	4
k=4	0.0439	0.97	0.0449	0.96	6
k=5	0.0221	0.99	0.0230	0.97	8

$\mathcal{M}_h = R_h^{\text{orth}} BV_h$

level k	$\|\mathbf{h} - \mathbf{h}_h\|$	Conv. Rate	$\|\mathbf{e} - \mathbf{e}_h\|$	Conv. Rate	# of iter
k=1	0.2400		0.2856		2
k=2	0.1554	0.63	0.1783	0.68	4
k=3	0.0764	1.02	0.0587	1.60	6
k=4	0.0266	1.52	0.0169	1.79	7
k=5	0.0081	1.72	0.0049	1.79	9

$\mathcal{M}_h = R_h^{\text{lump}} BV_h$

level k	$\|\mathbf{h} - \mathbf{h}_h\|$	Conv. Rate	$\|\mathbf{e} - \mathbf{e}_h\|$	Conv. Rate	# of iter
k=1	0.2980		0.2864		2
k=2	0.1564	0.93	0.1536	0.90	5
k=3	0.0821	0.93	0.0479	1.68	7
k=4	0.0281	1.54	0.0132	1.85	11
k=5	0.0088	1.67	0.0040	1.73	13

Table 6.8: Non-convex domain example with non-uniform refinement and $\omega = 1$.

$\mathcal{M}_h = BV_h$

level k	$\|\mathbf{h} - \mathbf{h}_h\|$	Conv. Rate	$\|\mathbf{e} - \mathbf{e}_h\|$	Conv. Rate	# of iter
k=1	0.3203		0.3422		7
k=2	0.2945	0.12	0.2685	0.35	16
k=3	0.1689	0.80	0.1463	0.88	30
k=4	0.0819	1.04	0.0766	0.93	45
k=5	0.0388	1.08	0.0395	0.96	64

Table 6.9: Non-convex domain example with non-uniform refinement and $\omega = 10$.

of convergence for $\|\mathbf{e} - \mathbf{e}_h\|$ that is higher than 2/3 even though $\mathbf{e} \notin \mathbf{H}^{2/3}(\Omega)$ due to the use of graded meshes.

6.6 Remarks on the SPLS Approach

In this chapter, we proposed a new least squares discretization method for the time-harmonic Maxwell equations written as a first order system. The approximability of the proposed method depends on how well the solution $\mathbf{p}$ can be represented as $\mathbf{p} = B\mathbf{w}$. The higher the regularity of the representant function $\mathbf{w}$, the better the SPLS approximation $\mathbf{p}_h$ of $\mathbf{p}$ becomes. For the no projection choice of trial space, the discretization error $\|\mathbf{p}-\mathbf{p}_h\|$ is independent of the $\inf - \sup$ constant m_h associated with the SPLS discrete system. Using the projection type trial spaces, the approximability is better if compared with other finite element approximation techniques that rely on piecewise linear approximation functions. In addition, the method is robust with respect to the frequency parameter ω, and it is efficient for solving problems on both convex and non-convex domains.

The fact that the operators B and B^* depend on the parameters ε, μ, ω affects the stability of the problem at the continuous and discrete levels. Indeed, since the operator B depends on ω the condition number of the discrete Schur complement depends on ω. Consequently, the number of iterations for Algorithm 2.4.1 depends on ω. From Table 6.2 through Table 6.6 and Tables 6.8 and 6.9, we can see that the number of iterations increases as $\omega \to \infty$ and $h \to 0$. Still, for a large range of values of ω we obtain an order of convergence to be close to or higher than one with just piecewise linear approximation.

Chapter 7

CONCLUSION AND FUTURE DIRECTIONS

In this thesis, we considered a saddle point least squares method to solve second order elliptic PDEs, as well as first order systems of PDEs, written as mixed variational formulations. In Chapter 2, the theory described connected the area of approximating the solutions of elliptic problems with the area of approximating the solutions to symmetric saddle point problems. The main advantage of the framework, and distinguishing characteristic from the original SPLS method, is the allowance of the choice of working with nonconforming trial spaces with desirable approximability properties.

In Chapter 3, a general preconditioned approach to solving mixed variational formulations was presented. The method relies on the classical theory of symmetric saddle point problems and on the theory of preconditioning symmetric, positive definite operators. The main idea was to replace the inner product on the discrete test space, that arises naturally through the saddle point reformulation, with an equivalent inner product that gives rise to efficient elliptic inversion or preconditioning. A major benefit of this is that we were able to analyze the resulting preconditioned formulation in a similar manner as the formulation in Chapter 2, and the approximability properties of the discrete trial spaces do not depend on the chosen norm on the test space.

We applied the SPLS framework with and without preconditioning to the discretization of second order elliptic interface problems in Chapters 4. The proposed method is easy to implement using Uzawa type algorithms, and the adoption of nonconforming trial spaces leads to higher order approximation if compared with standard finite element (non-mixed) techniques based on linear element approximation. In addition, the method works well when solving second order problems with variable coefficients, including highly oscillatory coefficients, and problems where the solution

has less regularity. In the case of preconditioning, the problem reduces to elliptic preconditioning associated with inner products on the test spaces, usually of H^1 type. We plan to further combine the SPLS discretization method with known multilevel and adaptive techniques [1, 8, 17, 38, 46, 74, 89].

In Chapter 5, we applied the SPLS framework to reaction diffusion equations with an emphasis on the reaction dominated case. The method, using the projection type trial spaces, leads to higher order approximation as seen in Chapter 4. We plan to provide a more thorough analysis of the method when specifically using Shishkin type meshes in the near future. In addition, we plan to apply the preconditioning theory of Chapter 3 to reaction diffusion equations, as well as combine the theory and techniques of Chapter 4 to reaction diffusion equations with discontinuous coefficients. In regards to the latter, preliminary numerical results have been obtained for a simple interface problem. More specifically, we solved

$$\begin{cases} -\mathrm{div}(A\nabla u) + cu = f & \text{in } \Omega, \\ u = 0 & \text{on } \partial\Omega, \end{cases}$$

on the unit square with $c = 1$ and f computed such that for

$$A(x,y) = a(x,y)I_2, \text{ where } a(x,y) = \begin{cases} 1 & \text{if } x < 1/2, \\ \mu & \text{if } x \geq 1/2, \end{cases}$$

the exact solution is given by

$$u(x,y) = \begin{cases} \mu x(x-1/2)y(y-1) & \text{if } x < 1/2, \\ (x-1/2)(x-1)y(1-y) & \text{if } x \geq 1/2. \end{cases}$$

Table 7.1 shows results for $\mu = 10, 100, 1000$ using a family of interface-fitted, locally quasi-uniform meshes $\{\mathcal{T}_h\}$ obtained through a standard uniform refinement strategy with a mesh size of $h = 2^{-k}$. We observe similar results compared with the interface problems presented in Chapter 4. We also plan to apply the SPLS approach to reaction diffusion equations where the coefficient c is discontinuous.

$\mathcal{M}_h = BV_h$

level k	$\mu = 10$			$\mu = 100$			$\mu = 1000$		
	error	rate	it	error	rate	it	error	rate	it
1	0.255		1	2.448		1	24.368		1
2	0.133	0.943	1	1.272	0.944	1	12.665	0.944	1
3	0.067	0.986	1	0.642	0.986	1	6.340	0.986	1
4	0.034	0.986	1	0.322	0.996	1	3.206	0.996	1
5	0.017	0.999	1	0.161	0.999	1	1.604	0.999	1

$\mathcal{M}_h = R_h^{\text{orth}} BV_h$

level k	$\mu = 10$			$\mu = 100$			$\mu = 1000$		
	error	rate	it	error	rate	it	error	rate	it
1	0.140		1	1.331		2	13.246		2
2	0.040	1.794	2	0.379	1.810	4	3.771	1.812	6
3	0.011	1.804	3	0.106	1.843	6	1.043	1.854	13
4	0.004	1.682	3	0.029	1.849	8	0.286	1.864	20
5	0.001	1.768	4	0.008	1.863	11	0.078	1.884	25

$\mathcal{M}_h = R_h^{\text{lump}} BV_h$

level k	$\mu = 10$			$\mu = 100$			$\mu = 1000$		
	error	rate	it	error	rate	it	error	rate	it
1	0.134		1	1.287		1	12.808		3
2	0.055	1.289	2	0.508	1.341	5	5.055	1.341	8
3	0.020	1.478	3	0.181	1.490	9	1.798	1.491	20
4	0.007	1.517	4	0.063	1.520	12	0.625	1.524	28
5	0.002	1.522	5	0.022	1.521	15	0.217	1.525	39

Table 7.1: Results for reaction diffusion interface example.

In Chapter 6, we proposed a new least squares discretization method for the time-harmonic Maxwell equations written as a first order system. The approach follows the methodology in [33], but different spaces are chosen at the discrete level. The SPLS approach provides good approximations for both the electric and magnetic fields, and the method is robust with respect to the frequency parameter ω. Furthermore, the projection type trial spaces provide a better approximation compared with the no projection trial space. We plan to further investigate whether the SPLS method can be applied to solving Maxwell's equations with different types of boundary conditions

and combine the preconditioning techniques of Chapter 3 with the SPLS discretization of Maxwell's equations. We also plan to further investigate the stability of the families of spaces presented.

An application of particular interest for the SPLS method is the Helmholtz equation

$$\Delta u + k^2 u = -f,$$

where k represents the wave number. As seen in Chapters 4 and 5, the stability of the families of discrete spaces, through the approach taken in this thesis, is independent of the parameters associated with the given PDE. With similar techniques, we plan on investigating how the Helmholtz equation fits into the SPLS framework for low and high wave number. In addition, we plan to explore the application of the SPLS method to the Stokes system written as a first order system, as well as linear elasticity.

BIBLIOGRAPHY

[1] M. Ainsworth and J.T. Oden. *A posteriori error estimation in finite element analysis*. Wiley-Interscience, New York, 2000.

[2] T. Apel. *Anisotropic finite elements: local estimates and applications*. Advances in Numerical Mathematics. B. G. Teubner, Stuttgart, 1999.

[3] T. Apel and S. Nicaise. The finite element method with anisotropic mesh grading for elliptic problems in domains with corners and edges. *Mathematical Methods in the Applied Sciences*, 21(6):519–549, 1998.

[4] D. Arnold, L.R. Scott, and M. Vogelius. Regular inversion of the divergence operator with dirichlet boundary conditions on polygon. *Ann. Scuola Norm. Sup. Pisa Cl. Sci.*, 15:169–196, 1988.

[5] A. Aziz and I. Babuška. Survey lectures on mathematical foundations of the finite element method. *The Mathematical Foundations of the Finite Element Method with Applications to Partial Differential Equations, A. Aziz, editor*, 1972.

[6] I. Babuška, G. Caloz, and J.E. Osborn. Special finite element methods for a class of second order elliptic problems with rough coefficients. *SIAM Journal on Numerical Analysis*, 31(4):945–981, 1994.

[7] C. Bacuta. Schur complements on Hilbert spaces and saddle point systems. *J. Comput. Appl. Math.*, 225(2):581–593, 2009.

[8] C. Bacuta. Cascadic multilevel algorithms for symmetric saddle point systems. *Comput. Math. Appl.*, 67(10):1905–1913, 2014.

[9] C. Bacuta, J.H. Bramble, and J. Pasciak. New interpolation results and applications to finite element methods for elliptic boundary value problems. *East-West J. Numer. Math*, 9(3):179–198, 2001.

[10] C. Bacuta, J.H. Bramble, and J. Pasciak. Using finite element tools in proving shift theorems for elliptic boundary value problems. *Num. Lin. Alg. Appl.*, 10(1-2):33–64, 2003.

[11] C. Bacuta, J.H. Bramble, and J. Xu. Regularity estimates for elliptic boundary value problems in Besov spaces. *Math. Comp.*, 72:1577 –1595, 2003.

[12] C. Bacuta, J.H. Bramble, and J. Xu. Regularity estimates for elliptic boundary value problems with smooth data on polygonal domains. *J. Numer. Math.*, 11(2):75–94, 2003.

[13] C. Bacuta and J. Jacavage. Saddle point least squares preconditioning of mixed methods. *Computers & Mathematics with Applications*, 77(5):1396–1407, 2019.

[14] C. Bacuta and J. Jacavage. Least squares preconditioning for mixed methods with nonconforming trial spaces. *Applicable Analysis*, DOI: 10.1080/00036811.2019.1582032, 2019.

[15] C. Bacuta and J. Jacavage. A non-conforming saddle point least squares approach for an elliptic interface problem. *Computational Methods in Applied Mathematics*, doi:10.1515/cmam-2018-0202, 2019.

[16] C. Bacuta, J. Jacavage, K. Qirko, and F.J. Sayas. Saddle point least squares iterative solvers for the time harmonic maxwell equations. *Comput. Math. Appl.*, 70(11):2915–2928, 2017.

[17] C. Bacuta and P. Monk. Multilevel discretization of symmetric saddle point systems without the discrete LBB condition. *Appl. Numer. Math.*, 62(6):667–681, 2012.

[18] C. Bacuta, V. Nistor, and L. Zikatanov. Improving the rate of convergence of 'high order finite elements' on polygons and domains with cups. *Numerische Mathematik*, 100(2):165 –184, 2005.

[19] C. Bacuta, V. Nistor, and L. Zikatanov. Improving the rate of convergence of high-order finite elements on polyhedra. I. A priori estimates. *Numer. Funct. Anal. Optim.*, 26(6):613–639, 2005.

[20] C. Bacuta and K. Qirko. A saddle point least squares approach to mixed methods. *Comput. Math. Appl.*, 70(12):2920–2932, 2015.

[21] C. Bacuta and K. Qirko. A saddle point least squares approach for primal mixed formulations of second order PDEs. *Comput. Math. Appl.*, 73(2):173–186, 2017.

[22] C. Bacuta, P. Vassilevski, and S. Zhang. A new approach for solving Stokes systems arising from a distributive relaxation method. *Numer. Methods Partial Differential Equations*, 27(4):898–914, 2011.

[23] N. Bakhvalov and G. Panasenko. *Homogenisation: averaging processes in periodic media*, volume 36 of *Mathematics and its Applications (Soviet Series)*. Kluwer Academic Publishers Group, Dordrecht, 1989.

[24] R.E. Bank and J. Xu. Asymptotically exact a posteriori error estimators II: General unstructured grids. *SIAM J. Numer. Anal.*, 41(6):2313–2332, 2003.

[25] M. Benzi, G. Golub, and J. Liesen. Numerical solutions of saddle point problems. *Acta Numerica*, 14:1–137, 2005.

[26] C. Bernardi and R. Verfürth. Adaptive finite element methods for elliptic equations with non-smooth coefficients. *Numerische Mathematik*, 85(4):579–608, 2000.

[27] P. Bochev, Z. Cai, T.A. Manteuffel, and S. F. McCormick. Analysis of velocity-flux first-order system least-squares principles for the Navier-Stokes equations. I. *SIAM J. Numer. Anal.*, 35(3):990–1009, 1998.

[28] P. Bochev and M.D. Gunzburger. Least-squares finite element methods. In *International Congress of Mathematicians. Vol. III*, pages 1137–1162. Eur. Math. Soc., Zürich, 2006.

[29] P. Bochev and M.D. Gunzburger. *Least-squares finite element methods*, volume 166 of *Applied Mathematical Sciences*. Springer, New York, 2009.

[30] F.A. Bornemann and P. Deuflhard. The cascadic multigrid method for elliptic problems. *Numerische Mathematik*, 75:135–152, 1996.

[31] D. Braess. *Finite elements: theory, fast solvers, and applications in solid mechanics*. Cambridge University Press, Cambridge, 1997.

[32] D. Braess and W. Dahmen. A cascadic multigrid algorithm for the Stokes equations. *Numer. Math.*, 82(2):179–191, 1999.

[33] J.H. Bramble, J. Pasciak, and T. Kolev. A least-squares method for the time-harmonic Maxwell equations. *J. Numer. Math.*, 13:237–320, 2005.

[34] J.H. Bramble and J.E. Pasciak. A new approximation technique for div-curl systems. *Math. Comp.*, 73:1739–1762, 2004.

[35] J.H. Bramble, J.E. Pasciak, and J. Xu. Parallel multilevel preconditioners. *Math. Comp.*, 55(191):1–22, 1990.

[36] J.H. Bramble and X. Zhang. The analysis of multigrid methods. In *Handbook of numerical analysis, Vol. VII*, pages 173–415. North-Holland, Amsterdam, 2000.

[37] S. Brenner and L.R. Scott. *The mathematical theory of finite element methods*. Springer-Verlag, New York, 1994.

[38] S.C. Brenner, H. Li, and L.Y. Sung. Multigrid methods for saddle point problems: Stokes and Lamé systems. *Numer. Math.*, 128(2):193–216, 2014.

[39] F. Brezzi. On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers. *Rev. Française Automat. Informat. Recherche Opérationnelle Sér. Rouge*, 8(R-2):129–151, 1974.

[40] F. Brezzi and M. Fortin. *Mixed and hybrid finite element methods*. Springer-Verlag, New York, 1991.

[41] Z. Cai, C.O. Lee, T.A. Manteuffel, and S.F. McCormick. First-order system least squares for linear elasticity: numerical results. *SIAM J. Sci. Comput.*, 21(5):1706–1727, 2000.

[42] Z. Cai, T.A. Manteuffel, and S.F. McCormick. First-order system least squares for the Stokes equations, with application to linear elasticity. *SIAM J. Numer. Anal.*, 34(5):1727–1741, 1997.

[43] Z. Cai, T.A. Manteuffel, S.F. McCormick, and S.V. Parter. First-order system least squares (FOSLS) for planar linear elasticity: pure traction problem. *SIAM J. Numer. Anal.*, 35(1):320–335, 1998.

[44] Z. Cai, T.A. Manteuffel, S.F. McCormick, and J. Ruge. First-order system LL* (FOSLL*): scalar elliptic partial differential equations. *SIAM J. Numer. Anal.*, 39(4):1418–1445, 2001.

[45] Z. Cai and S. Zhang. Flux recovery and a posteriori error estimators: conforming elements for scalar elliptic equations. *SIAM J. Numer. Anal.*, 48(2):578–602, 2010.

[46] C. Carstensen, M. Eigel, R.H.W. Hoppe, and C. Löbhard. A review of unified a posteriori finite element error control. *Numer. Math. Theory Methods Appl.*, 5(4):509–558, 2012.

[47] C. Clavero, J.L. Gracia, and E. O'Riordan. A parameter robust numerical method for a two dimensional reaction-diffusion problem. *Math. Comput.*, 74:1743–1758, 2005.

[48] A. Cohen, W. Dahmen, and G. Welper. Adaptivity and variational stabilization for convection-diffusion equations. *ESAIM Math. Model. Numer. Anal.*, 46(5):1247–1273, 2012.

[49] W. Dahmen, C. Huang, C. Schwab, and G. Welper. Adaptive Petrov-Galerkin methods for first order transport equations. *SIAM J. Numer. Anal.*, 50(5):2420–2445, 2012.

[50] L. Demkowicz and J. Gopalakrishnan. A class of discontinuous Petrov-Galerkin methods. Part I: the transport equation. *Comput. Methods Appl. Mech. Engrg.*, 199(23-24):1558–1572, 2010.

[51] L. Demkowicz and J. Gopalakrishnan. A primal DPG method without a first-order reformulation. *Comput. Math. Appl.*, 66(6):1058–1064, 2013.

[52] L. Demkowicz and L. Vardapetyan. Modeling of electromagnetic absorption/scattering problems using *hp*-adaptive finite elements. *Comput. Methods Appl. Mech. Engrg.*, 152(1-2):103–124, 1998.

[53] Y. Efendiev and T. Hou. Multiscale finite element methods for porous media flows and their applications. *Applied Numerical Mathematics*, 57(5):577 – 596, 2007.

[54] V. Girault and P.A. Raviart. *Finite element methods for Navier-Stokes equations*, volume 15. Springer-Verlag, Berlin, 1986.

[55] D. Gueribiz, F. Jacquemin, and S. Fréour. A moisture diffusion coupled model for composite materials. *European Journal of Mechanics - A/Solids*, 42:81–89, 2013.

[56] H. Guo and Z. Zhang. Gradient recovery for the crouzeix-raviart element. *Journal of Scientific Computing*, 64(2):456–476, 2015.

[57] H. Guo, Z. Zhang, R. Zhao, and Q. Zou. Polynomial preserving recovery on boundary. *Journal of Computational and Applied Mathematics*, 307:119–133, 2016.

[58] A. Hansbo and P. Hansbo. An unfitted finite element method, based on Nitsche's method, for elliptic interface problems. *Computer Methods in Applied Mechanics and Engineering*, 191(47-48):5537–5552, 2002.

[59] M.R. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *J. Research Nat. Bur. Standards*, 49:409–436, 1952.

[60] M. Jung, S. Nicaise, and J. Tabka. Some multilevel methods on graded meshes. *Journal of Computational and Applied Mathematics*, 138(1):151–171, 2002.

[61] T. Kato. Estimation of iterated matrices, with application to the Von Neumann condition. *Numer. Math.*, 2:22–29, 1960.

[62] B. C. Khoo, Z. Li, and P. Lin, editors. *Interface problems and methods in biological and physical flows*, volume 17 of *Lecture Notes Series. Institute for Mathematical Sciences. National University of Singapore*. World Scientific Publishing Co. Pte. Ltd., Hackensack, NJ, 2009.

[63] A. Kirsch and F. Hettlich. *The Mathematical Theory of Time-Harmonic Maxwell's Equations: Expansion-, Integral-, and Variational Methods*, volume 190 of *Applied Mathematical Sciences*. Springer International Publishing, 2014.

[64] A. Knyazev and O. Widlund. Lavrentiev regularization + ritz approximation = uniform finite element error estimates for differential equations with rough coefficients. *Mathematics of Computation*, 72(241):17–40, 2003.

[65] B. Lee, T. A. Manteuffel, S. F. McCormick, and J. Ruge. First-order system least-squares for the Helmholtz equation. *SIAM J. Sci. Comput.*, 21(5):1927–1949, 2000.

[66] E. Lee and T.A. Manteuffel. FOSLL* method for the eddy current problem with three-dimensional edge singularities. *SIAM J. Numer. Anal.*, 45(2):787–809, 2007.

[67] J. Li. Convergence and superconvergence analysis of finite element methods on highly nonuniform anisotropic meshes for singularly perturbed reactiondiffusion problems. *Applied Numerical Mathematics*, 36(2):129–154, 2001.

[68] J. Li and I.M. Navon. Uniformly convergent finite element methods for singularly perturbed elliptic boundary value problems i: Reaction-diffusion type. *Computers & Mathematics with Applications*, 35(3):57–70, 1998.

[69] R. Lin. Discontinuous discretization for least-squares formulation of singularly perturbed reaction-diffusion problems in one and two dimensions. *SIAM J. Numer. Anal.*, 47(1):89–108, 2008.

[70] R. Lin. Discontinuous galerkin least-squares finite element methods for singularly perturbed reaction-diffusion problems with discontinuous coefficients and boundary singularities. *Numerische Mathematik*, 112(2):295–318, 2009.

[71] R. Lin and M. Stynes. A balanced finite element method for singularly perturbed reaction-diffusion problems. *SIAM Journal on Numerical Analysis*, 50(5):2729–2743, 2012.

[72] T. Lin. *Layer-Adapted Meshes for Reaction-Convection-Diffusion Problems*. Lecture Notes in Mathematics. Springer-Verlag, Berlin, 2010.

[73] T.A. Manteuffel, S.F. McCormick, J. Ruge, and J.G. Schmidt. First-order system LL* (FOSLL*) for general scalar elliptic problems in the plane. *SIAM J. Numer. Anal.*, 43(5):2098–2120, 2005.

[74] K. Mekchay and R.H. Nochetto. Convergence of adaptive finite element methods for general second order linear elliptic PDEs. *SIAM Journal on Numerical Analysis*, 43(5):1803–1827, 2005.

[75] P. Monk. *Finite element methods for Maxwell's equations*. Numerical Mathematics and Scientific Computation. Oxford University Press, New York, 2003.

[76] L. Mu, J. Wang, and X. Ye. A weak galerkin generalized multiscale finite element method. *Journal of Computational and Applied Mathematics*, 305(Supplement C):68–81, 2016.

[77] I. Perugia, D. Schötzau, and P. Monk. Stabilized interior penalty methods for the time-harmonic Maxwell equations. *Comput. Methods Appl. Mech. Engrg.*, 191(41-42):4675–4697, 2002.

[78] M. Petzoldt. *Regularity and error estimators for elliptic problems with discontinuous coefficients*. PhD thesis, 2001.

[79] B. Pouliot, M. Fortin, A. Fortin, and A. Chamberland. On a new edge-based gradient recovery technique. *International Journal for Numerical Methods in Engineering*, 93(1):52–65, 2013.

[80] A. Quarteroni and A. Valli. *Domain decomposition methods for partial differential equations*. Clarendon Press, Oxford, 1999.

[81] H.G. Roos and M. Schopf. Convergence and stability in balanced norms of finite element methods on shishkin meshes for reaction-diffusion problems: Convergence and stability in balanced norms. *ZAMM Journal of applied mathematics and mechanics: Zeitschrift fur angewandte Mathematik und Mechanik*, 95(6):551–565, 2014.

[82] H.G. Roos, M. Stynes, and L. Tobiska. *Robust Numerical Methods for Singularly Perturbed Differential Equations: Convection-Diffusion-Reaction and Flow Problems*, volume 24 of *Springer Series in Computational Mathematics*. Springer Berlin Heidelberg, 2nd edition, 2008.

[83] F.J. Sayas. Infimum-supremum. *Pre-publicaciones del Seminario Matemático "García de Galdeano"*, (10):19–40, 2007.

[84] L.R. Scott and M. Vogelius. Conforming finite element methods for incompressible and nearly incompressible continua. In *Large-scale computations in fluid mechanics, Part 2 (La Jolla, Calif., 1983)*, volume 22 of *Lectures in Appl. Math.*, pages 221–244. Amer. Math. Soc., Providence, RI, 1985.

[85] L.R. Scott and M. Vogelius. Norm estimates for a maximal right inverse of the divergence operator in spaces of piecewise polynomials. *RAIRO Modél. Math. Anal. Numér.*, 19(1):111–143, 1985.

[86] G.I. Shishkin. Grid approximation of singularly perturbed boundary value problems with a regular boundary layer. *Sov. J. Numer. Anal. Math. Model.*, 4(5):397–417, 1989.

[87] L. Song and Z. Zhang. Superconvergence property of an over-penalized discontinuous galerkin finite element gradient recovery method. *Journal of Computational Physics*, 299:1004 – 1020, 2015.

[88] R. Verfürth. A combined conjugate gradient-multigrid algorithm for the numerical solution of the Stokes problem. *IMA J. Numer. Anal.*, 4(4):441–455, 1984.

[89] R. Verfürth. Robust a posteriori error estimators for a singularly perturbed reaction-diffusion equation. *Numerische Mathematik*, 78(3):479–493, 1998.

[90] J. Xu. Iterative methods by space decomposition and subspace correction. *SIAM Review*, 34:581–613, 1992.

[91] J. Xu and J. Qin. Some remarks on a multigrid preconditioner. *SIAM Journal on Scientific Computing*, 15(1):172–184, 1994.

[92] J. Xu and Y. Zhu. Uniform convergent multigrid methods for elliptic problems with strongly discontinuous coefficients. *Mathematical Models and Methods in Applied Sciences*, 18(1):77–105, 2008.

[93] J. Xu and L. Zikatanov. Some observations on Babuška and Brezzi theories. *Numer. Math.*, 94(1):195–202, 2003.

[94] X. Zhang. Multilevel schwarz methods. *Numerische Mathematik*, 63(1):521–539, 1992.

[95] J.Z. Zhu and O.C. Zienkiewicz. Superconvergence recovery technique and a posteriori error estimators. *International Journal for Numerical Methods in Engineering*, 30(7):1321–1339, 1990.

[96] O.C. Zienkiewicz and J.Z. Zhu. The superconvergent patch recovery and a posteriori error estimates. part 1: The recovery technique. *International Journal for Numerical Methods in Engineering*, 33(7):1331–1364, 1992.

[97] O.C. Zienkiewicz and J.Z. Zhu. The superconvergent patch recovery and a posteriori error estimates. part 2: Error estimates and adaptivity. *International Journal for Numerical Methods in Engineering*, 33(7):1365–1382, 1992.

Appendix A

PERMISSIONS

Copyright Clearance Center RightsLink®

Home | Create Account | Help

Title: Saddle point least squares iterative solvers for the time harmonic Maxwell equations

Author: Constantin Bacuta,Jacob Jacavage,Klajdi Qirko,Francisco-Javier Sayas

Publication: Computers & Mathematics with Applications

Publisher: Elsevier

Date: 1 December 2017

LOGIN

If you're a copyright.com user, you can login to RightsLink using your copyright.com credentials.

Already **a RightsLink user** or want to learn more?

Please note that, as the author of this Elsevier article, you retain the right to include it in a thesis or dissertation, provided it is not published commercially. Permission is not required, but please ensure that you reference the journal as the original source. For more information on this and on your other retained rights, please visit: https://www.elsevier.com/about/our-business/policies/copyright#Author-rights

BACK CLOSE WINDOW

RightsLink®

Home | Create Account | Help

Taylor & Francis
Taylor & Francis Group

Title: Least squares preconditioning for mixed methods with nonconforming trial spaces

Author: Constantin Bacuta, , Jacob Jacavage

Publication: Applicable Analysis

Publisher: Taylor & Francis

Date: Feb 27, 2019

Rights managed by Taylor & Francis

LOGIN

If you're a copyright.com user, you can login to RightsLink using your copyright.com credentials.

Already **a RightsLink user** or want to learn more?

Thesis/Dissertation Reuse Request

Taylor & Francis is pleased to offer reuses of its content for a thesis or dissertation free of charge contingent or resubmission of permission request if work is published.

BACK | CLOSE WINDOW

Copyright Clearance Center

Confirmation Number: 11819135
Order Date: 05/29/2019

Customer Information

Customer: Jacob Jacavage
Account Number: 3001461391
Organization: University of Delaware
Email: jjacav@udel.edu
Phone: +1 (570) 985-9445
Payment Method: Invoice

This is not an invoice

Order Details

Journal of computational methods in applied mathematics

Billing Status: **N/A**

Order detail ID:	71911136
ISSN:	1609-9389
Publication Type:	Journal
Volume:	
Issue:	
Start page:	
Publisher:	Institute of Mathematics of the National Academy of Sciences of Belarus
Author/Editor:	Instytut matėmatyki (Natsyi͡anal'nai͡a akadėmii͡a navuk Belarusi)

Permission Status: Granted
Permission type: Republish or display content
Type of use: Thesis/Dissertation
Order License Id: 4598260804087

Requestor type	Author of requested content
Format	Print, Electronic
Portion	chapter/article
Number of pages in chapter/article	17
The requesting person/organization	Jacob Jacavage
Title or numeric reference of the portion(s)	Entire article
Title of the article or chapter the portion is from	A Non-conforming Saddle Point Least Squares Approach for an Elliptic Interface Problem
Editor of portion(s)	N/A
Author of portion(s)	Jacob Jacavage
Volume of serial or monograph	N/A
Page range of portion	1-17
Publication date of portion	published online on 2019-04-13
Rights for	Main product
Duration of use	Life of current edition
Creation of copies for the disabled	no
With minor editing privileges	no
For distribution to	Worldwide
In the following language(s)	Original language of publication
With incidental promotional use	no
Lifetime unit quantity of new product	More than 2,000,000
Title	A Non-conforming Saddle Point Least Squares Approach for an Elliptic Interface Problem

Note: This item was invoiced separately through our **RightsLink service.** More info **$ 0.00**