DISSERTATION

A POSTERIORI ANALYSIS OF OPERATOR DECOMPOSITION ON INTERFACE PROBLEMS

Submitted by

Timothy Wildey

Department of Mathematics

In partial fulfillment of the requirements

for the degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Summer 2007

COLORADO STATE UNIVERSITY

July 6, 2007

WE HEREBY RECOMMEND THAT THE DISSERTATION PREPARED UNDER OUR SUPERVISION BY TIMOTHY WILDEY ENTITLED "*A POSTERIORI* ANALYSIS OF OPERATOR DECOMPOSITION ON INTERFACE PROBLEMS" BE ACCEPTED AS FULFILLING IN PART REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY.

Committee on Graduate Work

Dr. Albert Ted Watson

Dr. Jiangguo Liu

Adviser: Dr. Donald Estep

Co-Adviser: Dr. Simon Tavener

Department Head: Dr. Simon Tavener

ABSTRACT OF DISSERTATION

A POSTERIORI ANALYSIS OF OPERATOR DECOMPOSITION ON INTERFACE PROBLEMS

This thesis is devoted to the *a posteriori* analysis of the effects of operator decomposition on interface problems. Operator decomposition offers an attractive solution strategy for multi-physics and multi-scale problems. This technique allows previously defined component codes optimized for single physics problems to be reused in an iterative manner to solve a multi-physics, multi-scale problem. This is also called loose-coupling or a partitioned approach. Unfortunately, this technique introduces additional errors due to the transfer of information between components. For interface problems, this results in a loss of accuracy for the converged approximation in the L^2 norm. In this thesis, we use *a posteriori* analysis to detect the source of this loss of accuracy and show that a common flux recovery technique can be used to recovery the expected accuracy.

Timothy Wildey
Department of Mathematics
Colorado State University
Fort Collins, Colorado 80523
Summer 2007

ACKNOWLEDGEMENTS

I would like to thank my advisors, Dr. Don Estep and Dr. Simon Tavener, for all of their guidance and support over the past few years. I cannot imagine a better environment for a graduate student to grow and to develop the skills to become successful researcher and mathematician. Most of all, I would like to thank my wife Rebecca for all of her patience and support through all of my late nights and early mornings.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

Interface problems arise naturally in the modeling of heat transfer between materials where a discontinuous coefficient represents the difference in the thermal properties in each region. In 1970, Babuska [4] studied elliptic problems with discontinuous coefficients on a smooth domain with a smooth interface. Since then, interface problems have been analyzed for elliptic and parabolic problems with smooth interfaces [24, 14], using an immersed finite volume element method [32], using least-squares finite element methods [19], and using mortar elements for nonmatching triangulations across the interface [2, 59, 50, 46]. All of these methods are designed to solve the fully coupled system since the solution in each subdomain behaves on the same scale and may be approximated with the same numerical method.

Operator decomposition is a widely used technique for solving multi-physics, multi-scale problems. The general approach is to decompose the problem into components involving simpler physics over a relatively limited range of scales, and then to seek the solution of the entire system through an iterative procedure involving solutions of the individual components. This approach is appealing because there is generally a good understanding of how to solve a broad spectrum of single physics problems accurately and efficiently, and because it provides an alternative to accommodating multiple scales in one discretization.

In the case of conjugate heat transfer, operator decomposition can be viewed as domain decomposition that allows for very different discretizations in each component. However, operator decomposition presents an entirely new set of accuracy and stability issues, some of which are obvious and some subtle, and all of which are difficult to correct. In the case of the conjugate heat transfer, the operator decomposition causes a loss in the approximation order of convergence.

In this thesis, we perform an *a posteriori* error analysis of a finite-element implementation of the operator decomposition technique and obtain accurate error estimates that are used to guide an adaptive discretization strategy. Our approach is based on the standard techniques using variational analysis, residuals and the generalized Green's function solution to an adjoint problem [10, 18, 30, 31, 28, 42], which we modify to account for several new features arising from the operator decomposition. These include the following.

- Numerical errors in the solution of each component are propagated through the boundary condition applied to the other component.
- Numerical errors in the solution of each of the components at one step of the iterative procedure are propagated to the next step.
- The solution operator of the full problem and the approximate solution operator of the problem are distinct operators with distinct adjoint operators and these differences must be recognized when seeking accurate error estimates.

These issues typically arise in operator decomposition solution processes, e.g. [21, 38], and generally require extensions of the usual *a posteriori* analysis techniques.

In addition to obtaining accurate estimates, we seek to improve the accuracy of the operator decomposition method in an efficient way. In particular, we adapt the "boundary element flux" technique developed by Wheeler [55] and Carey [34, 20] to compute normal derivatives on a boundary, and show that this can be used to improve accuracy, and in particular,

restore the order of convergence that is lost due to the operator decomposition.

In Chapter 2, we review some basic definitions and classical results in functional analysis and function spaces. This naturally leads to the definition of the adjoint operator. We conclude this chapter by computing the adjoint for a number of differential operators and show how the adjoint boundary conditions are determined.

In Chapter 3, we develop the Galerkin finite element method for linear elliptic differential equations and prove optimal order *a priori* and *a posteriori* error bounds. Next, we show how the adjoint solution may be used to estimate a linear functional of the forward solution or to adaptively refine the mesh. We conclude the chapter by deriving *a priori* and *a posteriori* error bounds for stationary nonlinear problems.

Chapter 4 describes the *a posteriori* estimation of the error in the flux of a finite element approximation on a piece of the boundary of the domain. The estimate is obtained via a generalized Green's function corresponding to the quantity of interest on the boundary. We investigate the effects of smoothing the data corresponding to the quantity of interest and explore the effective domain of dependence of the quantity. We relate this approach to previous work by M. F. Wheeler [55], G. F. Carey [34, 20], I. Babuska [7, 8, 6], et al, and M. Larson [35], et al.

In Chapter 5, we discuss the convergence of operator decomposition methods. We begin with covergence criteria for an analogous linear alegbra problem with a block structure. Next, we prove that operator decomposition methods for 1D interface problems only converge in certain situations. As an alternative in the divergent case, we present a relaxation method and

develop a Newton method for the boundary information. We complete the chapter by extending these results to operator decomposition methods for 2D and 3D interface problems and for globally coupled equations.

In Chapter 6, we consider the accuracy of an operator decomposition finite element method for a conjugate heat transfer problem consisting of two conducting materials coupled through a common boundary. We derive accurate *a posteriori* error estimates that account for the transfer of error between components of the operator decomposition method as well as the differences between the adjoints of the full problem and the discrete iterative system. We use these estimates to guide adaptive mesh refinement. In addition, we address a loss of order of convergence that results from the decomposition, and show that the approximation order of convergence is limited by the accuracy of the transferred gradient information. We employ a boundary flux recovery method to regain the expected order of accuracy in an efficient manner.

In Chapter 7, we explore a few advanced topics in finite element theory. First, we show how a commonly used finite volume method may be rewritten as a Petrov-Galerkin finite element method and optimal order *a priori* and *a posteriori* error bounds are derived. In addition, we apply adjoint based *a posteriori* error estimation techniques to estimate a linear functional of the error in the finite volume approximation. Next, we analyze mixed finite element methods and prove optimal order error bounds for the Stokes equations. We build on this result in the following section where we develop a finite element method for the Navier Stokes equations, derive the adjoint operator, and prove *a priori* and *a posteriori* error bounds. We conclude this chapter with an analysis of a finite element method for the steady Boussinesq equations.

In Chapter 8, we consider the accuracy of an operator decomposition finite element method for a conjugate heat transfer problem consisting of a fluid and a solid coupled through a common boundary. We derive accurate *a posteriori* error estimates that account for both local discretization errors and the effect of the transfer of error between components of the operator decomposition method. We use these estimates to guide adaptive mesh refinement. In addition, we again show that the order of convergence of the operator decomposition method is limited by the accuracy of the transferred (gradient) information, and how a simple boundary flux recovery method can be used to regain the optimal order of accuracy in an efficient manner.

In Chapter 9, we extend the results in the previous chapters to a time-dependent interface transfer problem. We begin by discussing the complication which arises due to the interaction between integration and iteration. Next, we use the adjoint problem to derive an error representation formula which accounts for both local discretization errors and the effect of the transfer of error between components of the operator decomposition method. In the final section, we present numerical results. The first demonstrates that certain operator decomposition methods are stable only if the time step is below a critical value. The second result shows that the accuracy of the operator decomposition method is affected by the transfer of derivative information, and that the flux recovery technique described in previous chapters may be extended to time-dependent problems and used to regain the loss of accuracy.

Chapter 2

ADJOINT OPERATORS

In this chapter, we review some relevant material on duality and adjoint operators. See [44, 10, 12, 42, 18, 30, 31, 28] for more details.

2.1 Banach spaces

Recall that a linear functional on a vector space V is a map $L : V \to \mathbb{R}$ such that $L(\alpha v_1 + \beta v_2) = \alpha L(v_1) + \beta L(v_2)$.

We define a special type of map on V, $\|\cdot\| : V \to \mathbb{R}$, called a norm, satisfying the following properties

1. $\|u + v\| \leq \|u\| + \|v\|, \quad \forall u, v \in V,$
2. $\|\alpha v\| = |\alpha|\|v\|, \quad \forall v \in V, \ \forall \alpha \in \mathbb{R},$
3. $\|v\| \geq 0, \quad \forall v \in V,$
4. $\|v\| = 0 \Rightarrow v = 0.$

A map $|\cdot| : V \to \mathbb{R}$ is called a semi-norm if it satisfies the first three properties, but not necessarily the fourth. We define a normed vector space to be a pair $(V, \|\cdot\|_V)$ with V a vector space and $\|\cdot\|_V$ a norm on V.

Definition 2.1.1. *A Banach space is a normed vector space that is complete with respect to $\|\cdot\|_V$, i.e. any Cauchy sequence in V converges under $\|\cdot\|_V$ to an element in V.*

Definition 2.1.2. *The dual space of a normed vector space V is the set of all bounded linear functionals on V and is denoted V^*. In addition, the dual space is a normed vector space under the dual norm*

$$\|L\|_{V^*} = \sup_{x \in V,\, x \neq 0} \frac{|L(x)|}{\|x\|_V}.$$

We use the bracket notation,

$$< \cdot, \cdot >: V^* \times V \to \mathbb{R}$$

to represent the dual pairing.

Definition 2.1.3. *Let X and Y be Banach spaces and $L : X \to Y$ a bounded linear operator. The adjoint operator, $L^* : Y^* \to X^*$, is defined so that*

$$< y^*, Lx >_Y = < L^* y^*, x >_X . \tag{2.1.1}$$

Example 2.1.1. *Let $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$ and $A : \mathbb{R}^n \to \mathbb{R}^m$. Then the adjoint $A^* : \mathbb{R}^m \to \mathbb{R}^n$ is the transpose matrix, often denoted A^T.*

2.2 Hilbert spaces

Definition 2.2.1. *An inner product space is a pair $(V, (\cdot, \cdot)_V)$ with V a vector space and $(\cdot, \cdot)_V$ an inner product on V.*

We can always take an inner product space and create a normed vector space using the norm induced by the inner product $\| \cdot \|_V = (\cdot, \cdot)_V^{1/2}$. This leads to an important connection between inner product spaces and Banach spaces.

Definition 2.2.2. *An inner product space, $(V, (\cdot, \cdot)_V)$, is called a Hilbert space if the associated normed vector space $(V, (v, v)_V^{1/2})$ is a Banach space.*

Let V be a Hilbert space with inner product $(\cdot, \cdot)_V$. Given $v \in V$, it is clear that $L(x) = (x, v)_V$ defines a linear functional for any $x \in V$. This naturally leads to the question if all linear functionals can be defined this way. The Riesz Representation Theorem states that this is the case on Hilbert spaces.

Theorem 2.2.1. *Riesz Representation Theorem*

Any bounded linear functional on a Hilbert space V may be uniquely represented as $L(v) = (u, v)_V$ for some $u \in V$ and $\|L\|_{V^*} = \|u\|_V$

We say that a Hilbert space is isometrically isomorphic to its dual space.

Definition 2.2.3. *A bilinear form, $a(\cdot, \cdot) : V \times V \to \mathbb{R}$, on a Hilbert space V is said to coercive and continuous if it satisfies*

$$a(v, v) \geq \alpha \|v\|_V^2, \text{ and } \quad |a(u, v)| \leq C\|u\|_V \|v\|_V, \tag{2.2.1}$$

respectively with $0 < \alpha \leq C$.

These conditions imply that the bilinear form induces a norm equivalent to $\|\cdot\|_V$. If the bilinear form is symmetric, continuous and coercive, then the Riesz Representation theorem guarantees there exists a unique u such that

$$a(u, v) =< f, v >, \quad \forall v \in V$$

We may reach the same conclusion if the bilinear form in non-symmetric, but the Lax-Milgram theorem must be used.

Theorem 2.2.2. *Lax-Milgram*

Given a Hilbert space $(V, (\cdot, \cdot))$, a continuous and coercive bilinear form, $a(\cdot, \cdot) : V \times V \to \mathbb{R}$, and a continuous linear functional $f \in V^*$, there exists a unique $u \in V$ such that

$$a(u, v) =< f, v >, \quad \forall v \in V.$$

2.3 Sobolev spaces

In this section, we introduce a special class of Hilbert spaces: the Sobolev spaces. First, we define the $L^p(\Omega)$ spaces,

$$L^p(\Omega) = \left\{ v \mid \left(\int_\Omega v^p \, dx \right)^{1/p} < \infty \right\},$$

where the integral is understood in the Lebesgue sense. The space $L^2(\Omega)$ is particularly important since it becomes a Hilbert space under the inner product

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

with the corresponding norm

$$\|v\|_{0,\Omega} = (v, v)^{1/2}.$$

We will frequently write $(\cdot, \cdot) = (\cdot, \cdot)_\Omega$ and $\|\cdot\|_0 = \|\cdot\|_{0,\Omega}$ when the domain is obvious.

Remark 2.3.1. *The dual space of $L^p(\Omega)$ is $L^q(\Omega)$ with $\frac{1}{p} + \frac{1}{q} = 1$ for any integers $1 < p, q < \infty$. $L^2(\Omega)$ is a special case since $(L^2(\Omega))^* = L^2(\Omega)$*

For convenience, we define a multi-index notation for partial derivatives. Let $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_m) \in \mathbb{N}^m$ with $|\alpha| = \alpha_1 + \ldots + \alpha_m$, and for $x \in \mathbb{R}^m$ let $x^\alpha = x_1^{\alpha_1} \cdots x_m^{\alpha_m}$. Similarly, we define the differential operator

$$D^\alpha = \left(\frac{\partial}{\partial x_1} \right)^{\alpha_1} \cdots \left(\frac{\partial}{\partial x_m} \right)^{\alpha_m}.$$

Definition 2.3.1. *We say $u \in L^2(\Omega)$ has a weak derivative $\phi = D^\alpha u$ in $L^2(\Omega)$ if*

$$(v, \phi)_0 = (-1)^{|\alpha|} (D^\alpha v, u)_0,$$

for all smooth v which are nonzero on a compact subset of Ω.

It is easy to see that this definition corresponds to the classical definition of integration by parts whenever u is sufficiently smooth, and the test function v (and its derivatives) are zero along the boundary.

Definition 2.3.2. *For an integer $m \geq 0$, define the space of functions*

$$H^m(\Omega) = \left\{ v \in L^2(\Omega) \mid D^\alpha u \in L^2(\Omega) \ \forall |\alpha| \leq m \right\}.$$

This space is a Hilbert space under the inner product

$$(u, v)_m = \sum_{|\alpha| \leq m} (D^\alpha u, D^\alpha v)_0,$$

with the norm

$$\|v\|_m = (v, v)_m^{1/2},$$

and the corresponding semi-norm

$$|v|_m = \left(\sum_{|\alpha|=m} \|D^\alpha u\|_0^2 \right)^{1/2}.$$

We are also interested in the subspace

$$H_0^m(\Omega) = \{ v \in H^m(\Omega) \mid v|_{\partial\Omega} = 0. \}.$$

Next, we define Sobolev spaces of positive real order.

Definition 2.3.3. *Let $k \geq 0$ be an integer, $\sigma \in (0, 1)$, and $s = k + \sigma$. We define the Sobolev space*

$$H^s(\Omega) = \left\{ v \in H^k(\Omega) \mid \frac{|D^\alpha v(x) - D^\alpha v(y)|}{\|x - y\|^{\sigma + d/2}} \in L^2(\Omega \times \Omega), \ \forall |\alpha| = k \right\},$$

with the norm

$$\|v\|_s = \left(\|v\|_k^2 + \sum_{|\alpha|=k} \int_{\Omega \times \Omega} \frac{|D^\alpha v(x) - D^\alpha v(y)|^2}{\|x - y\|^{2\sigma + d}} \, dx \, dy \right)^{1/2}.$$

Definition 2.3.4. *The negative norm Sobolev space, $H^{-s}(\Omega)$, is defined to be the dual space of $H^s(\Omega)$. It is useful to remember the following embeddings*

$$\ldots H^{-2}(\Omega) \supset H^{-1}(\Omega) \supset L^2(\Omega) \supset H_0^1(\Omega) \ldots .$$

The most common use of non-integer order Sobolev spaces is in dealing with functions defined over boundaries. The definition of $H^s(\partial\Omega)$ is slightly different [1, 3], but $H^s(\partial\Omega)$ is still a Hilbert space and gives rise to the following trace theorems.

Theorem 2.3.1. *If $u \in H^m(\Omega)$, then the trace $\gamma u = u|_{\partial\Omega}$ belongs to $H^{m-1/2}(\partial\Omega)$ and*

$$\|\gamma u\|_{m-1/2,\partial\Omega} \leq K_1 \|u\|_{m,\Omega}.$$

Conversely, if $v \in H^{m-1/2}(\partial\Omega)$ then there exists $u \in H^m(\Omega)$ with $v = u|_{\partial\Omega}$ and

$$\|u\|_{m,\Omega} \leq K_2 \|v\|_{m-1/2,\partial\Omega}.$$

Theorem 2.3.2. *Suppose that Ω has a Lipschitz boundary, and $u \in H^m(\Omega)$ with $m \geq 0$. Then there is a constant C, such that*

$$\|v\|_{m-1,\partial\Omega} \leq C \|v\|_{m-1,\Omega}^{1/2} \|v\|_{m,\Omega}^{1/2},$$

and if we set $|\partial\Omega| = r$ then there exists $C' > 0$ such that

$$\|v\|_{m-1,\partial\Omega} \leq C'(r^{-1}\|v\|_{m-1,\Omega} + r|v|_{m,\Omega}).$$

2.4 Computing adjoint differential operators

In general, computing adjoint operators can be difficult. There is a systematic formal process in the setting of a Hilbert space with the L^2

inner product. In this case, we compute adjoint differential operators by applying a version of the divergence theorem

$$-\int_{\Omega} \nabla \cdot (A\nabla u)v \; dx = -\int_{\partial\Omega} A\nabla u \cdot \mathbf{n} \; v \; dS + \int_{\Omega} A\nabla u \cdot \nabla v \; dx, \quad (2.4.1)$$

where $\mathbf{n}$ denote the unit outward normal, to move derivatives from the test function onto the trial function. We first compute the adjoint operator for functions having compact support in the domain, assuring that any boundary terms will drop out. To extend the adjoint to more general functions, we must define the adjoint boundary conditions.

Definition 2.4.1. *The formal adjoint boundary conditions are the minimal conditions to reduce to zero all of the boundary terms that arise when evaluating the bilinear indentity.*

Next, we compute the adjoint to a number of differential operators and indicate how the adjoint boundary conditions can change depending on the forward differential operator and boundary conditions.

2.4.1 Linear elliptic operators

Consider the elliptic operator

$$Lu = -\nabla \cdot (A\nabla u) + \mathbf{b} \cdot \nabla u + cu,$$

The adjoint differential operator, L^*, is defined such that

$$(Lu, \phi) = (u, L^*\phi).$$

We apply the divergence theorem and determine

$$L^*\phi = -\nabla \cdot (A\nabla \phi) - \mathbf{b} \cdot \nabla \phi + (c - \nabla \cdot \mathbf{b})\phi,$$

with extra boundary terms

$$-(A\partial_n u, \phi) + (A\partial_n \phi, u) + ((\mathbf{b}\cdot\mathbf{n})u, \phi)\,.$$

In Table 2.4.1, we assume Γ is a connected subset of the boundary and consider several common boundary conditions for the forward problem, and the resulting adjoint boundary conditions.

Forward BC on Γ	Adjoint BC on Γ
$u = 0$	$\phi = 0$
$A\partial_n u = 0$	$A\partial_n \phi + (\mathbf{b}\cdot\mathbf{n})\phi = 0$
$A\partial_n u - (\mathbf{b}\cdot\mathbf{n})u = 0$	$A\partial_n \phi = 0$
$A\partial_n u + \alpha u = 0$	$A\partial_n \phi + (\alpha + \mathbf{b}\cdot\mathbf{n})\phi = 0$

Table 2.1: Common forward boundary conditions and the corresponding adjoint conditions.

2.4.2 Linear parabolic operators

Consider the parabolic operator

$$u_t - \nabla\cdot(A\nabla u) + \mathbf{b}\cdot\nabla u + cu. \tag{2.4.2}$$

To determine the adjoint operator, we multiply by a smooth function ϕ, integrate over Ω and integrate in time over $[0, T]$. The divergence theorem is applied to move the spatial derivatives as before, and integration by parts gives

$$\int_0^T (u_t, \phi)\ dt = -\int_0^T (u, \phi_t)\ dt,$$

neglecting the boundary terms. Thus, the adjoint operator is

$$-\phi_t - \nabla\cdot(A\nabla\phi) - \mathbf{b}\cdot\nabla\phi + (c - \nabla\cdot\mathbf{b})\phi.$$

A parabolic operator such as (2.4.2) is well-posed given appropriate boundary conditions in space, as well as an initial condition. The adjoint boundary

conditions in space can be read from Table 2.4.1. To derive an "initial condition" for the adjoint problem, we recall that integration by parts in time results in

$$(u, \phi)|_{t=T} - (u, \phi)|_{t=0}.$$

Assuming the initial condition for the forward problem is $u(x, 0) = 0$, we set $\phi(x, T) = 0$ to remove these terms. Hence, the adjoint problem is given a final condition rather than an initial condition, but notice that the time derivative runs backwards, giving a well-posed problem.

2.4.3 Nonlinear operators

The adjoint of a nonlinear operator is more complicated to define. Consider the nonlinear problem,

$$F(u) = g,$$

and let U be an approximation to u. We want to estimate a linear functional of the error, $(e, \psi) =)(u - U, \psi)$. The residual,

$$R = g - F(U),$$

is computable, but in general, $F(e) \neq F(u) - F(U)$ since F is nonlinear. To obtain a linear operator we use the integral mean value theorem

$$F(u) - F(U) = \int_0^1 F'(su + (1 - s)U) \cdot (u - U)\, ds,$$

where F' is the Jacobian. This requires that F is Frechet differentiable and that the domain of F is convex.

We now have the linear equation for the error,

$$\overline{F}e = R,$$

where

$$\overline{F} = \int_0^1 F(su + (1-s)U)\, ds.$$

We define the linearized adjoint operator $\overline{F}^*$ such that

$$(\overline{F}e, \phi) = (e, \overline{F}^*\phi).$$

Unfortunately, this adjoint operator is not useful in practice as it requires the exact solution u. We can define a computable adjoint by linearizing around the approximate solution, $F'(U)$, giving the approximate adjoint problem,

$$(F'(U))^*\phi = \psi.$$

In practice, this appears to give meaningful error and stability estimates. In Eastman [25], the effects of linearization on *a posteriori* error estimation for elliptic problems is analyzed.

Chapter 3

FINITE ELEMENT METHODS

In this chapter, we recall some basic ideas about finite element methods. See [15, 13, 45, 27] for more details.

3.1 Weak formulations

Let Ω be a bounded domain with boundary $\partial\Omega$ and consider the model boundary value problem

$$\begin{cases} Lu = -\nabla \cdot (A\nabla u) + \mathbf{b} \cdot \nabla u + cu = f, & \mathbf{x} \in \Omega, \\ u = 0, & \mathbf{x} \in \partial\Omega, \end{cases} \tag{3.1.1}$$

with the data f and coefficients $A(x) \geq A_0 > 0, \mathbf{b}$ and c sufficiently smooth functions. To derive the weak form, we multiply the differential equation (3.1.1) by a smooth function, v, integrate over Ω, and apply the Divergence theorem to move derivatives from u to v. This gives

$$a(u,v) = \int_\Omega (A\nabla u \cdot \nabla v + \mathbf{b} \cdot u\, v + c\, u\, v)\, dx = \int_\Omega fv\, dx. \tag{3.1.2}$$

In order for this relation to make sense, we require $u, v \in H_0^1(\Omega)$ and $f \in H^{-1}(\Omega)$. We call this the weak formulation because we have weakened the regularity assumptions from the stronger relation (3.1.1).

Suppose that instead of homogeneous Dirichlet boundary conditions, we have $u = g$ on $\partial\Omega$. We assume $g \in H^{1/2}(\partial\Omega)$ and by the trace theorem there exists $G \in H^1(\Omega)$ such that $\gamma G = g$ along $\partial\Omega$. We let $u = u_0 + G$ and seek $u_0 \in H_0^1(\Omega)$ such that

$$a(u_0, v) = < f, v > - a(G, v), \quad \forall v \in H_0^1(\Omega).$$

Next we consider the boundary value problem (3.1.1) with mixed boundary conditions

$$\begin{cases} u = 0, & x \in \Gamma_D, \\ A\partial_n u = g_N, & x \in \Gamma_N, \end{cases}$$

where $\partial\Omega = \Gamma_D \cup \Gamma_N$. The application of the Divergence theorem now results a term along the boundary involving the Neumann data g_N. As a result, we define $V = \{v \in H^1(\Omega) \mid v|_{\Gamma_D} = 0\}$ and seek $u \in V$ such that

$$a(u, v) =< f, v > + < g_N, v >_{\Gamma_N}, \quad \forall v \in V.$$

Suppose we further generalize the boundary conditions to be

$$\begin{cases} u = 0, & x \in \Gamma_D, \\ A\partial_n u = g_N, & x \in \Gamma_N, \\ A\partial_n u + \beta u = g_R, & x \in \Gamma_R, \end{cases}$$

where we have split the boundary into Dirichlet, Neumann, and Robin components with $\partial\Omega = \Gamma_D \cup \Gamma_N \cup \Gamma_R$. The weak formulation seeks $u \in V$ such that

$$a(u, v) + < \beta u, v >_{\Gamma_R} =< f, v > + < g_N, v >_{\Gamma_N} + < g_R, v >_{\Gamma_R}, \quad \forall v \in V.$$

A similar conclusion can be reached using inhomogeneous Dirichlet data.

3.1.1 Existence and uniqueness

For simplicity, we focus on the Dirichlet problem (3.1.1) and the weak formulation (3.1.2). The bilinear form $a(\cdot, \cdot)$ is clearly continuous, and if we assume $\mathbf{b} = \mathbf{0}$ and $c(x) \geq 0$, it will also be coercive. Then, given $f \in H^{-1}(\Omega)$, existence and uniqueness of $u \in H_0^1(\Omega)$ satisfying

$$a(u, v) =< f, v >, \quad \forall v \in H_0^1(\Omega),$$

is guaranteed by the Riesz representation theorem (2.2.1).

If, on the other hand, $\mathbf{b} \neq \mathbf{0}$, then the bilinear form is not symmetric and (2.2.1) does not apply. Instead, we can show existence and uniqueness using the Lax-Milgram lemma (2.2.2) provided $c(x) - \frac{1}{2}\nabla \cdot \mathbf{b} \geq 0$, which guarantees coercivity.

3.1.2 Elliptic regularity

Let $u \in H^2(\Omega) \cap H_0^1(\Omega)$ be the solution to (3.1.1) with $f \in L^2(\Omega)$. We say u satisfies the elliptic regularity condition if

$$\|u\|_2 \leq C\|f\|_0, \tag{3.1.3}$$

for some positive constant C.

It is natural to wonder if assuming smoother data, i.e. $f \in H^s(\Omega)$ with $s > 0$, leads to smoother solutions. In general, this is not the case unless the boundary is more smooth than a polygonal domain. This presents a number of problems associated with curved boundaries [13, 15]. We do not address these issues here.

3.2 Piecewise polynomial spaces

3.2.1 Triangulations

One of the keys to the finite element method is the concept of a partition, or triangulation, of the domain into a finite number of subsets. Let $T_h = \{K\}$ denote a triangulation of Ω satisfying

1. $\overline{\Omega} = \cup_{K \in T_h} K$,
2. each K is closed with a nonempty interior,
3. for distinct $K_1, K_2 \in T_h$, $K_1 \cap K_2$ is either empty, or a common vertex, or a common edge.

To elaborate, we assume that the triangulation covers all of Ω, and that there are no hanging nodes. For an arbitrary element $K \in T_h$, let h_K denote the longest edge of K with

$$h = \max_{K \in T_h} h_K.$$

Furthermore, we say a triangulation is quasi-uniform if there exists a constant C_0 such that

$$\min_{K \in T_h} h_K \geq C_0 \max_{K \in T_h} h_K,$$

i.e. all the elements are of similar size.

3.2.2 The Lagrange basis

Next we use the triangulation, T_h, to define a finite dimensional space. Let $P^q(k)$ denote the space of piecewise polynomials of degree q on an element K, and define $\{N_i\}_K$ to be the set of linearly independent basis functions such that for any $p \in P^q(K)$,

$$p(x) = \sum_{i=1}^{M} \alpha_i N_i(x),$$

where $\alpha_i \in \mathbb{R}$.

Let $\{\eta_i\}_K$ denote the set of nodes on an element K. The Lagrange basis is defined so that $N_i(\eta_j) = \delta_{ij}$. For $P^1(K)$, the set of nodes is simply the corners of the element K. With higher degree polynomials, we need to add nodes to the element determine a basis. For $P^2(K)$, we require 6 basis functions, so we take the three corner nodes, as well as the midpoint of each edge. For $P^3(K)$ we need 10 basis functions, so we use the corner nodes, two nodes along each edge, and the centroid to form a basis.

Let $K_1, K_2 \in T_h$ represent two element sharing a common edge. It is an easy exercise to show that if $\{\eta_i\}_E$ denotes the set of nodes along the common edge, E, and if $v(\eta_i)|_{K_1} = v(\eta_i)|_{K_2}$ for all nodes in $\{\eta_i\}_E$, then v is continuous over K_1 and K_2.

3.2.3 Interpolation theory

At this point, it is useful to review some basis facts from interpolation theory. Let K be an arbitrary element of a triangulation T_h, and let S_h denote the space of continuous piecewise polynomial functions of degree q on T_h. Define the interpolation operator $\pi : H^{q+1}(\Omega) \rightarrow S_h$ such that

$$\pi v = \sum_{i=1}^{N_h} v_i \phi_i(x),$$

where $v_i = v(x_i)$ and $\phi_i(x)$ the Lagrange basis function associated with node x_i.

The following local estimates can be derived using the Bramble-Hilbert lemma

$$\|v - \pi v\|_{s,K} \leq C_K h_K^{q+1-s} |v|_{q+1,K}. \tag{3.2.1}$$

If we assume the triangulation is quasi-uniform, then we can prove the global bounds

$$\|v - \pi v\|_s \leq C h^{q+1-s} |v|_{q+1} \tag{3.2.2}$$

3.3 Galerkin finite element methods

3.3.1 Formulation

Consider the second order elliptic equation

$$\begin{cases} Lu = f, & \mathbf{x} \in \Omega, \\ u = 0, & \mathbf{x} \in \partial\Omega. \end{cases} \tag{3.3.1}$$

Let $a(\cdot,\cdot) : H_0^1(\Omega) \times H_0^1(\Omega) \rightarrow \mathbb{R}$ denote the bilinear form associated with (3.3.1) and assume $f \in H^{-1}(\Omega)$. The weak form of (3.3.1) seeks $u \in H_0^1(\Omega)$ such that

$$a(u,v) = < f, v >, \quad \forall\, v \in H_0^1(\Omega). \tag{3.3.2}$$

Since H_0^1 is an infinite dimensional space, we need to choose finite dimensional closed subspaces of $H_0^1(\Omega)$, The subspace containing the test functions, v, is frequently called the test space, while the subspace containing the approximation, U, is called the trial space. For now, we consider problems where the test and trial space are both S_h as previously defined. Then the Galerkin finite element method finds $U \in S_h$ such that

$$a(U, v) =< f, v >, \quad \forall\, v \in S_h. \tag{3.3.3}$$

For simplicity, we set

$$S_h = \{v \in C(\Omega) \mid v \in P^1(K)\ \forall K \in T_h\},$$

e.g. the space of continuous piecewise linear polynomials, unless stated otherwise.

3.3.2 H^1 error bounds

We begin by proving that the H^1 error in the finite element approximation is proportional to the best approximation in S_h.

Theorem 3.3.1. *If the bilinear form in (3.3.2) is continuous and coercive, then the finite element solution to (3.3.3) satisfies*

$$\|u - U\|_1 \leq C \inf_{v \in S_h} \|u - v\|_1,$$

and if $u \in H^2(\Omega) \cap H_0^1(\Omega)$ satisfies the elliptic regularity condition (3.1.3), then

$$\|u - U\|_1 \leq Ch\|f\|_0$$

Proof. Let v be an arbitrary element of S_h and consider

$$\begin{aligned}
\alpha\|u-U\|_1^2 &\leq a(u-U,u-U) && \text{Coercivity}\\
&= a(u-U,u) && \text{Galerkin Orthogonality}\\
&= a(u-U,u-v) && \text{Galerkin Orthogonality}\\
&\leq C_1\|u-U\|_1\|u-v\|_1 && \text{Continuity}\\
\|u-U\|_1 &\leq \tfrac{C_1}{\alpha}\|u-v\|_1 && \text{Division by } \alpha\|u-U\|_1.
\end{aligned}$$

Since $v \in S_h$ was arbitrary, we have

$$\|u-U\|_1 \leq C \inf_{v\in S_h} \|u-v\|_1.$$

The second assertion follows from taking $v = \pi u$ and applying the interpolation results and elliptic regularity.

3.3.3 L^2 error bounds

Error bounds in the L^2 norm are more complicated than H^1 or energy norm bounds, and come in two varieties: *a-priori* and *a-posteriori*. Both of these require the adjoint problem and some regularity of adjoint solutions. Let ϕ solve the adjoint problem

$$\begin{cases} L^*\phi = u-U, & \mathbf{x}\in\Omega,\\ \phi = 0, & \mathbf{x}\in\partial\Omega, \end{cases} \tag{3.3.4}$$

with weak formulation

$$a^*(\phi,w) = (u-U,w), \quad w\in H_0^1(\Omega).$$

First we prove the *a-priori* result.

Theorem 3.3.2. *Assume the bilinear form in (3.3.2) is continuous and coercive, $u \in H^2(\Omega)\cap H_0^1(\Omega)$ and satisfies the elliptic regularity condition (3.1.3), and let ϕ solve (3.3.4) with $\|\phi\|_2 \leq C\|u-U\|_0$. Then the finite element solution to (3.3.3) satisfies*

$$\|u-U\|_0 \leq Ch^2\|f\|_0,$$

Proof. Letting $w = u - U$ we have

$$\begin{array}{rcll} \|u-U\|_0^2 & = & a^*(\phi, u-U) & \text{Weak form of Adjoint} \\ & = & a(u-U,\phi) & \text{Def. of Adjoint} \\ & = & a(u-U,\phi-\pi\phi) & \text{Galerkin Orthogonality} \\ & \leq & C_1\|u-U\|_1\|\phi-\pi\phi\|_1 & \text{Continuity} \\ & \leq & C_2 h\|u\|_2\|\phi-\pi\phi\|_1 & \text{See (3.3.1)} \\ & \leq & C_3 h^2\|u\|_2\|\phi\|_2 & \text{Interp. Theory} \\ & \leq & C_4 h^2\|f\|_0\|u-U\|_0 & \text{Elliptic Regularity} \\ \|u-U\|_0 & \leq & C_4 h^2\|f\|_0 & \text{Division by } \|u-U\|_0 \end{array} \tag{3.3.5}$$

Notice that we got one power of h from the H^1 bound on $u - U$, and one from a similar bound on $\phi - \pi\phi$. In the next theorem, we prove the *a-posteriori* error bound, which gets both powers of h from $\phi - \pi\phi$.

First we make an important point. The residual we want to use is $f - LU$ which gives an indication how well the approximation satisfies the strong form of the differential equation. Unfortunately, $U \notin H^2(\Omega)$ so (LU, ϕ) does not make sense globally. However, we do have $U \in H^2(K)$ for any $K \in T_h$. Therefore, our estimates are computed locally which produces jump terms along the element boundaries. We will have to bound these terms separately, but this is easily accomplished using the trace theorems.

Theorem 3.3.3. *Assume the bilinear form in (3.3.2) is continuous and coercive, $u \in H^2(\Omega) \cap H_0^1(\Omega)$ and satisfies the elliptic regularity condition (3.1.3), and let ϕ solve (3.3.4) with $\|\phi\|_2 \leq C\|u-U\|_0$. Then the finite element solution to (3.3.3) satisfies*

$$\|u-U\|_0 \leq \sum_{K\in T_h} CS_K h_K^2 \left(\|f-LU\|_{0,K} + \|f\|_{0,K}\right) \tag{3.3.6}$$

where $S_K = \|\phi\|_{2,K}$ is a local stability factor.

Proof. Let ϕ solve

$$\begin{cases} L^*\phi = \frac{u-U}{\|u-U\|_0}, & \mathbf{x} \in \Omega, \\ \phi = 0, & \mathbf{x} \in \partial\Omega, \end{cases} \tag{3.3.7}$$

Notice that we have normalized the data for the adjoint problem. In general, we will not be able to divide by $\|u-U\|_0$ as in the *a-priori* error bound.

$$\begin{aligned}
\|u-U\|_0 &= a^*(\phi, u-U) && \text{Weak Adjoint}\\
&= a(u-U,\phi) && \text{Def. of Adjoint}\\
&= a(u-U,\phi-\pi\phi) && \text{Galerkin Orth.}\\
&= (f,\phi-\pi\phi)-a(U,\phi-\pi\phi) && \text{Use } a(u,v)=(f,v)\\
&= \textstyle\sum_{K\in T_h}(f,\phi-\pi\phi)_K - a(U,\phi-\pi\phi)_K\\
&= \textstyle\sum_{K\in T_h}(f-LU,\phi-\pi\phi)_K - \\
&\quad \tfrac{1}{2}([A\partial_n U],\phi-\pi\phi)_{\partial K} && \text{Green's Identity}\\
&= J_1+J_2
\end{aligned}$$

where $[A\partial_n U]$ denotes the jump in the normal derivative across an edge or a face. We bound each term separately.

$$\begin{aligned}
J_1 &= \textstyle\sum_{K\in T_h}(f-LU,\phi-\pi\phi)_K\\
&\le \textstyle\sum_{K\in T_h} C_1\|f-LU\|_{0,K}h_K^2\|\phi-\pi\phi\|_K && \text{Cauchy-Schwartz}\\
&\le \textstyle\sum_{K\in T_h} C_2\|f-LU\|_{0,K}h_K^2\|\phi\|_{2,K} && \text{Interp. Theory}
\end{aligned}$$

$$\begin{aligned}
J_2 &= \textstyle\sum_{K\in T_h}\tfrac{1}{2}([A\partial_n U],\phi-\pi\phi)_{\partial K}\\
&= \textstyle\sum_{K\in T_h}(A\partial_n(u-U),\phi-\pi\phi)_{\partial K} && \pm(A\partial_n u,\phi-\pi\phi)_{\partial K}\\
&\le \textstyle\sum_{K\in T_h} C_1\|u-U\|_{1,\partial K}\|\phi-\pi\phi\|_{0,\partial K} && \text{Cauchy-Schwartz}\\
&\le \textstyle\sum_{K\in T_h}\left(C_2\|u-U\|_{1,K}^{1/2}\|u-U\|_{2,K}^{1/2}\right)\cdot\\
&\quad \left(C_3\|\phi-\pi\phi\|_{0,K}^{1/2}\|\phi-\pi\phi\|_{1,K}^{1/2}\right) && \text{Trace Thm.}\\
&\le \textstyle\sum_{K\in T_h} C_4 h_K^2\|f\|_{0,K}\|\phi\|_{2,K} && \text{Interp. and Ellip. Reg.}
\end{aligned}$$

3.3.4 Estimating a linear functional

The use of the adjoint problem in the *a-posteriori* L^2 error bound leads to another important application. Namely, the estimation of linear functionals of the error, $u-U$. The goal is to compute some specific information from a computed approximation accurately. Quantities such as the average value, the drag on a wing, the normal flux through a boundary, or a point value are important, and all can be expressed as linear functionals

of the solution. There is a significant amount of literature on this subject, but we refer the reader to [26, 45, 10, 18, 30, 11] for more information and applications.

Consider the differential operator given by (3.3.1) and let ϕ solve

$$\begin{cases} L^*\phi = \psi, & \mathbf{x} \in \Omega, \\ \phi = 0, & \mathbf{x} \in \partial\Omega, \end{cases} \tag{3.3.8}$$

with $\psi \in H^{-1}(\Omega)$. Formally speaking, we do not use the strong form of the adjoint. Rather, we use the weak formulation,

$$a^*(\phi, w) = < \psi, w >, \quad w \in H_0^1(\Omega).$$

Now we let $w = e$ and observe

$$\begin{aligned} < \psi, e > &= a^*(\phi, e) \\ &= a(e, \phi) \\ &= a(e, \phi - \pi\phi) \\ &= (f, \phi - \pi\phi) - a(U, \phi - \pi\phi). \end{aligned} \tag{3.3.9}$$

This equation is frequently called the error representation formula. Formally, given ϕ we can compute $\pi\phi$ and determine the exact linear functional of the error.

3.3.5 Adaptive mesh refinement

The adjoint-based *a-posteriori* error estimate provides the capability of developing a global basis for adaptivity that takes into account both the local production of error from discreitization and the global effects of stability in terms of propagation, accumulation, and cancelation of error across the domain.

The standard approach is to write (3.3.9) as

$$|(\psi, e)| \leq \sum_{K \in T_h} |(f, \phi - \pi\phi)_K - a(U, \phi - \pi\phi)_K| \,,$$

with the obvious notation for localizing form to elements K. The right-hand side provides a computable bound after approximating ϕ. When the bound is larger than the stated tolerance, an element K is marked for refinement when the local element indicator

$$\eta_K = |(f, \phi - \pi\phi)_K - a(U, \phi - \pi\phi)|,$$

is larger than a local tolerance, typically the tolerance divided by the current number of elements. This is the standard, optimal control "Principle of Equidistribution", basis for error control.

The dual weights provided by the factors involving ϕ mean that the estimate and the local indicators reflect the stability properties of the quantity of interest. We can define a useful notion of *effective* domain of dependence [30] for a quantity of interest. In an elliptic problem, the domain of influence of the value of a solution in a localized region is formally the entire domain. However, in many cases, generalized Green's functions corresponding to local quantities of interest exhibit a decay away from the local region. We define the effective domain of influence as the region where the local indicators η_K are significantly larger than the rest, i.e. where the mesh must be refined in order to yield the quantity of interest to a desired accuracy. A large effective domain of influence corresponds to needing to refine a large portion of the domain in order to resolve a localized quantity.

3.3.6 Approximating the adjoint

In practice, ϕ is never known and must be approximated. The only constraint is that we cannot use the same finite element method that we used on the forward problem. Galerkin orthogonality would cause the error estimate to be zero.

In some approaches, the adjoint solution is computed on a finer mesh and in other approaches it is computed using higher order elements. Alternatively, there have been attempts to use patch recovery techniques [10, 47, 48].

3.4 Nonlinear problems

Consider the nonlinear reaction diffusion equation

$$\begin{cases} -\nabla \cdot (A\nabla u) + f(u) = g(x), & \mathbf{x} \in \Omega, \\ u = 0, & \mathbf{x} \in \partial\Omega. \end{cases} \tag{3.4.1}$$

with $A \geq A_0 > 0$, $g \in L^2(\Omega)$, and Ω a convex polygon in $\mathbb{R}^2$. We assume f is Lipschitz continuous, i.e.

$$\|f(u) - f(v)\|_0 \leq C\|u - v\|_0,$$

for all $u, v \in H_0^1(\Omega)$, and that the Gateaux derivative of f, which we denote f', exists and is bounded. Since the problem is nonlinear, we must define an iterative method to find the approximation. Given an initial guess, u_0, we consider two classical iterative methods: Newton's method (NM) and successive substitution (SS) defined in weak form by

$$\text{(NM)} \quad a(u^k, v) + \left(f'(u^{k-1})u^k, v\right) = \left(g(x) + f'(u^{k-1})u^{k-1} - f(u^{k-1}), v\right),$$

and

$$\text{(SS)} \quad a(u^k, v) + \left(f(u^{k-1}), v\right) = \left(g(x), v\right),$$

for $v \in H_0^1(\Omega)$ with $a(u, v) = \int_\Omega A\nabla u \cdot \nabla v \, dx$.

3.4.1 Coercivity

For the linear Poisson equation, the coercivity of the bilinear form was easy show. If on the other hand, we change Poisson's equation to

$$-\nabla \cdot (A\nabla u) + cu = f,$$

and if $c(x) \geq 0$ we see

$$\alpha \|e\|_1^2 \leq a(e,e) \leq a(e,e) + (ce,e),$$

and we may continue as in section 2.1. We can weaken the assumption on $c(x)$ if we use the positivity of $a(\cdot,\cdot)$ to balance a slightly negative $c(x)$. Suppose $c(x) \geq \beta$ where β is negative. Then

$$\begin{aligned} a(e,e) + (ce,e) &\geq \alpha \|e\|_1^2 + (ce,e) \\ &\geq \alpha \|e\|_1^2 + \beta \|e\|_0^2 \\ &\geq \alpha \|e\|_1^2 + C_P^2 \beta \|e\|_1^2 \end{aligned}$$

where C_P is the constant from the Poincare inequality

$$\|v\|_0 \leq C_P \|v\|_1, \quad v \in H_0^1(\Omega),$$

which depends only on the domain Ω. From this, we see that the problem is coercive if

$$\alpha + C_P^2 \beta > 0,$$

or, if

$$\beta > -\frac{\alpha}{C_P^2}.$$

For our nonlinear problem, we need to guarantee

$$a(e,e) + (f(u) - f(U), e) > 0.$$

First, we define the linearized form, $\overline{f}$, such that

$$(f(u) - f(U), e) = \left(\overline{f}e, e\right),$$

specifically,

$$\overline{f} := \int_0^1 f'(su + (1-s)U^k)\, ds. \tag{3.4.2}$$

Clearly, assuming the linearized form is positive definite will be a sufficient condition for coercivity. Alternatively, we may define

$$\beta = \inf_{v \in H_0^1(\Omega)} \frac{(\overline{f}v, v)}{\|v\|_0^2},$$

which leads to the same condition as the linear problem

$$\beta > -\frac{\alpha}{C_P},$$

where C_P is the constant from the Poincare inequality. Under this assumption we have

$$\alpha\|e\|_1^2 \leq a(e,e) + \left(f(u) - f(U^k), e\right).$$

3.4.2 Nonlinear orthogonality

We must be careful to apply Galerkin orthogonality correctly. Notice that u satisfies

$$a(u,v) + (f(u), v) = (g, v),$$

for any $v \in V_h$. The approximation, U^k, *does not* satisfy this equation. Instead, if we have used (NM) we have

$$a(U^k, v) + \left(f'(U^{k-1})U^k, v\right) = \left(g + f'(U^{k-1})U^{k-1} - f(U^{k-1}), v\right),$$

for any $v \in V_h$. Subtracting the two equations, we have

$$a(e,v) + (f(u), v) - \left(f'(U^{k-1})U^k, v\right) = \left(f(U^{k-1}) - f'(U^{k-1})U^{k-1}, v\right),$$

and after rearranging

$$a(e,v) + (f(u),v) = \left(f(U^{k-1}),v\right) + \left(f'(U^{k-1})(U^k - U^{k-1}),v\right),$$

To simplify, assume f is sufficiently smooth and expand

$$f(U^k) = f(U^{k-1}) + f'(U^{k-1})(U^k - U^{k-1}) + f''(U^{k-1})(U^k - U^{k-1})^2 + \text{h.o.t.},$$

where h.o.t. denotes higher order terms. Substituting this gives

$$a(e,v) + (f(u),v) - \left(f(U^k),v\right) = -\left(f''(U^{k-1})(U^k - U^{k-1})^2,v\right),$$

for any $v \in V_h$.

3.4.3 H^1 error bounds

Now we have

$$\begin{aligned}
\alpha\|e\|_1^2 \leq & a(e,e) + \left(f(u) - f(U^k),e\right) \\
= & a(e,u) + \left(f(u) - f(U^k),u\right) + \left(f''(U^{k-1})(U^k - U^{k-1})^2,U^k\right) \\
= & a(e,u-\pi u) + \left(f(u) - f(U^k),u-\pi u\right) \\
& + \left(f''(U^{k-1})(U^k - U^{k-1})^2,U^k - \pi u\right) \\
& = I_1 + I_2 + I_3
\end{aligned}$$

We bound each of these separately. For the first term we only need to use continuity:

$$\begin{aligned}
I_1 &= a(e,u-\pi u) \\
&\leq C\|e\|_1 \cdot \|u-\pi u\|_1.
\end{aligned}$$

For the second term we use continuity, the Lipschitz assumption on f, and the Poincare inequality:

$$\begin{aligned}
I_2 &= \left(f(u) - f(U^k),u-\pi u\right) \\
&\leq C\|e\|_0 \cdot \|u-\pi u\|_0 \\
&\leq C\|e\|_1 \cdot \|u-\pi u\|_1
\end{aligned}$$

For the third term we use only continuity

$$\begin{aligned} I_3 &= \left(f''(U^{k-1})(U^k - U^{k-1})^2, U^k - \pi u\right) \\ &\leq C\|f''(U^{k-1})\|_0 \cdot \|U^k - U^{k-1}\|_0^2 \cdot \|U^k - \pi u\|_0 \end{aligned}$$

We use these bounds along with interpolation and elliptic regularity results to conclude

$$\|e\|_1 \leq Ch\|g\|_0 + \frac{C\|f''(U^{k-1})\|_0 \cdot \|U^k - U^{k-1}\|_0^2 \cdot \|U^k - \pi u\|_0}{\alpha\|e\|_1}.$$

Of course, we have assumed f'' to be bounded, and that $\|U^k - U^{k-1}\|_0^2$ can be made as small as we like. This implies that the second term will not dominate.

If we use (SS) to solve the nonlinear problem, we can derive a similar bound

$$\|e\|_1 \leq Ch\|g\|_0 + \frac{C\|f'(U^{k-1})\|_0 \cdot \|U^k - U^{k-1}\|_0 \cdot \|U^k - \pi u\|_0}{\alpha\|e\|_1}.$$

Notice that we only require that f' is bounded, but the second term converges linearly with respect to $\|U^k - U^{k-1}\|$ rather than quadratically.

3.4.4 L^2 error bounds

Next, we derive L^2 error bounds in the case that (NM) is used to solve the forward problem. Let ϕ solve the linearized adjoint problem

$$\begin{cases} -\nabla \cdot (A\nabla\phi) + \overline{f}\phi = \psi, & \mathbf{x} \in \Omega, \\ \phi = 0, & \mathbf{x} \in \partial\Omega. \end{cases} \tag{3.4.3}$$

with $\overline{f}$ defined by (3.4.2). We let $\psi = e$, multiply by e, and integrate by parts to give

$$\begin{aligned}
\|e\|_0^2 =& a(\phi, e) + \left(\overline{f}\phi, e\right) \\
=& a(e, \phi) + \left(f(u) - f(U^k), \phi\right) \\
=& a(e, \phi - \pi\phi) + \left(f(u) - f(U^k), \phi - \pi\phi\right) \\
& - \left(f''(U^{k-1})(U^k - U^{k-1})^2, \pi\phi\right) \\
=& I_1 + I_2 + I_3.
\end{aligned}$$

We bound each of these terms individually. For the first term, we use continuity, the H^1 error bound, an interpolation result, and elliptic regularity.

$$\begin{aligned}
I_1 =& a(e, \phi - \pi\phi) \\
\leq& C\|e\|_1 \cdot \|\phi - \pi\phi\|_1 \\
\leq& Ch^2\|g\|_0 \cdot \|\phi\|_2 \\
\leq& Ch^2\|g\|_0 \cdot \|e\|_0
\end{aligned}$$

For the second term, we use continuity, the Lipschitz assumption on f, an interpolation result, and elliptic regularity.

$$\begin{aligned}
I_2 =& \left(f(u) - f(U^k), \phi - \pi\phi\right) \\
\leq& C\|f(u) - f(U^k)\|_0 \cdot \|\phi - \pi\phi\|_0 \\
\leq& Ch^2\|e\|_0 \cdot \|\phi\|_2 \\
\leq& Ch^2\|g\|_0 \cdot \|e\|_0
\end{aligned}$$

For the third term, we use continuity, the boundedness of f'', and the stability of the interpolant.

$$\begin{aligned}
I_3 =& -\left(f''(U^{k-1})(U^k - U^{k-1})^2, \pi\phi\right) \\
\leq& \|f''(U^{k-1})\|_0 \cdot \|U^k - U^{k-1}\|_0^2 \cdot \|\pi\phi\|_0 \\
\leq& C\|U^k - U^{k-1}\|_0^2
\end{aligned}$$

We combine these terms to give the *a-priori* error bound

$$\|e\|_0 \leq Ch^2\|g\|_0 + \frac{C\|U^k - U^{k-1}\|_0^2}{\|e\|_0}.$$

An *a-posteriori* error bound can be derived as follows,

$$\begin{aligned}
\|e\|_0^2 =& a(e, \phi - \pi\phi) + \left(f(u) - f(U^k), \phi - \pi\phi\right) \\
& - \left(f''(U^{k-1})(U^k - U^{k-1})^2, \pi\phi\right) \\
=& (g, \phi - \pi\phi) - a(U^k, \phi - \pi\phi) - \left(f(U^k), \phi - \pi\phi\right) \\
& - \left(f''(U^{k-1})(U^k - U^{k-1})^2, \pi\phi\right) \\
=& \sum_{K \in T_h} \left(g + \nabla \cdot (A\nabla U^k) - f(U^k), \phi - \pi\phi\right)_K \\
& + \frac{1}{2}\left([A\partial_n U^k], \phi - \pi\phi\right)_{\partial K} \\
& - \left(f''(U^{k-1})(U^k - U^{k-1})^2, \pi\phi\right) \\
\leq & \sum_{K \in T_h} Ch_K^2 \|g - f(U^k)\|_{0,K} \cdot \|e\|_{0,K} \\
& + Ch_K^2 \|g\|_{0,K} \cdot \|e\|_{0,K} \\
& + C\|U^k - U^{k-1}\|_0^2,
\end{aligned}$$

where we have omitted the standard arguments for the jump terms. The bound is complete after dividing by $\|e\|_0$.

Similar *a-priori* and *a-posteriori* error bounds can be derived if (SS) is used to solve the nonlinear problem. The only difference is that the third term again converges linearly with respect to $\|U^k - U^{k-1}\|_0$ rather than quadratically.

Chapter 4

BOUNDARY FLUX CALCULATIONS

4.1 Introduction

Goal-oriented error estimation is critically important in large scale computational science and engineering. Indeed, the situation in which the goal of a computation is to obtain an accurate approximation of a specific quantity of interest, e.g. the normal flux of the solution on a portion of the boundary of the domain, is very common, if not the norm, in practice. Moreover, it is very often possible to compute specific quantities of interest accurately using discretizations that yield poor global accuracy in the sense of some norm. This is critically important in applications that are too complex and large to allow asymptotically resolved discretizations.

However, computing a quantity of interest with true efficiency requires an understanding of exactly how the discretization errors affect that specific quantity of interest. Several different methods have been developed to estimate the error in a linear functional of the solution along a boundary. A popular technique for calculating boundary flux values was developed by Wheeler [55] and expanded later by Carey [34, 20]. Babuska and Miller introduced another technique for estimating linear functionals in [7, 8, 6]. In [35], Larson et al applied *a-posteriori* error analysis techniques based on the generalized Green's function [10, 12, 36, 26, 27, 30, 31, 28, 29] to estimate the error in a boundary flux computed by a post-processing technique.

In this chapter we build on the results in [35] to describe how to estimate the error in the average flux of the solution over a piece of the boundary using adjoint-based *a-posteriori* techniques. Our purpose is to clarify how the adjoint problem is defined in order to obtain this quantity of interest. We also relate this approach to the prior work and draw some interesting

connections between all of these approaches. Finally, we investigate the affect that smoothing the data for the adjoint problem corresponding to the quantity of interest has on the effective domain of dependence of the data as determined by the generalized Green's function.

This chapter is organized as follows. In the first section, we introduce the model problem. In the second section, we define the appropriate adjoint problem and derive the error representation formula. In the third section, we relate these results to previous work on boundary flux computations. In the last section, we apply these results to various test problems and one application.

4.2 Model problem

We consider a second order linear elliptic problem of the form

$$\begin{cases} Lu = -\nabla \cdot (A\nabla u) + \boldsymbol{b} \cdot \nabla u + cu = f(x), & x \in \Omega, \\ u = g_D, & x \in \Gamma_D, \\ A\partial_n u = g_N, & x \in \Gamma_N, \\ A\partial_n u + \alpha u = g_R, & x \in \Gamma_R, \end{cases} \tag{4.2.1}$$

posed on a convex bounded polygonal domain Ω with boundary $\partial\Omega$, where $\Gamma_D \cup \Gamma_N \cup \Gamma_R = \partial\Omega$, $\Gamma_D \neq \emptyset$, ∂_n denotes the unit outward normal derivative, the boundary data g_D, g_N, g_R and coefficients $A(x) \geq A_0 > 0$, $\boldsymbol{b}, c, f$ are sufficiently smooth functions.

Let $V = \{v \in H^1(\Omega) \mid v = 0 \text{ on } \Gamma_D\}$. The weak formulation of (4.2.1) seeks $u \in H^1(\Omega)$ such that $u = g_D$ on Γ_D and

$$a(u, v) = (f, v) + (g_N, v)_{\Gamma_N} + (g_R, v)_{\Gamma_R}, \text{ for all } v \in H_0^1(\Omega), \tag{4.2.2}$$

with

$$a(u, v) = \int_\Omega (A\nabla u \cdot \nabla v + \boldsymbol{b} \cdot \nabla u\, v + cuv)dx + \int_{\Gamma_R} \alpha uv\, ds,$$

and $(\cdot,\cdot)_{\Gamma_N}, (\cdot,\cdot)_{\Gamma_R}$ denote the integrals along Γ_N, Γ_R respectively. We assume that a is coercive. The Galerkin finite element method selects a finite dimensional subspace $S_h \subset H_0^1(\Omega)$ and seeks $U \in S_h$ such that (4.2.2) admits a unique weak solution,

$$a(U, v) = (f, v), \text{ for all } v \in S_h.$$

Throughout this paper, we use

$$S_h = \{v \in C(\Omega) \cap P^1(K), \forall K \in T_h\},$$

where T_h is a quasi-uniform triangulation of Ω.

4.3 Estimating the error using the generalized Green's function

4.3.1 Defining the adjoint

The standard adjoint to (4.2.1) for a quantity of interest in the interior is

$$\begin{cases} L^*\phi = -\nabla \cdot (A\nabla\phi) - \boldsymbol{b} \cdot \nabla\phi + (c - \nabla \cdot \boldsymbol{b})\phi = \psi(x), & x \in \Omega, \\ \phi = 0, & x \in \Gamma_D, \\ A\partial_n\phi + (\boldsymbol{b} \cdot \boldsymbol{n})\phi = 0, & x \in \Gamma_N, \\ A\partial_n\phi + (\alpha + \boldsymbol{b} \cdot \boldsymbol{n})\phi = 0, & x \in \Gamma_R. \end{cases} \tag{4.3.1}$$

Note that the presence of the convection term has altered the boundary conditions on the Neumann and Robin portions of the boundary.

Now we consider the case when the quantity of interest is the boundary flux on a portion of the boundary, $\Gamma \in \Gamma_D$, for (4.2.1). Computing formally, we modify the adjoint problem (4.3.1) to get

$$\begin{cases} L^*\phi = -\nabla \cdot (A\nabla\phi) - \boldsymbol{b} \cdot \nabla\phi + (c - \nabla \cdot \boldsymbol{b})\phi = 0, & x \in \Omega, \\ \phi = \psi, & x \in \Gamma, \\ \phi = 0, & x \in \Gamma_D \backslash \Gamma, \\ A\partial_n\phi + (\boldsymbol{b} \cdot \boldsymbol{n})\phi = 0, & x \in \Gamma_N, \\ A\partial_n\phi + (\alpha + \boldsymbol{b} \cdot \boldsymbol{n})\phi = 0, & x \in \Gamma_R, \end{cases} \tag{4.3.2}$$

where $\Gamma \subset \Gamma_D$ and $\psi \in H^{1/2}(\Gamma)$. The associated bilinear form for the adjoint problem is still $a^*(\phi, w)$ but the analysis for the error representation changes since the adjoint data appears on the boundary rather than the right-hand side of the adjoint equation. Thus,

$$0 = a^*(\phi, e) = a(e, \phi),$$

but ϕ is not zero everywhere on Γ_D. Consequently, the true solution of (4.2.1) satisfies

$$a(u, \phi) = (f, \phi) - (A\partial_n u, \phi)_\Gamma.$$

This also complicates the use of Galerkin orthogonality. We use a special interpolant. For $K \in T_h$, let $\{\eta_i\}_K$ denote the set of nodes of K. We define $\pi^0 : H^2 \to S_h$ via

$$\begin{cases} \pi^0 v(\eta_i) = \pi v(\eta_i), & \eta_i \notin \Gamma, \\ \pi^0 v(\eta_i) = 0, & \eta_i \in \Gamma, \end{cases}$$

where π is the Lagrange interpolant. Proceeding, we have

$$0 = a(e\phi) = a(e, \phi - \pi^0\phi) = (f, \phi - \pi^0\phi) - a(u - U, \phi - \pi^0\phi) + (A\partial_n u, \psi)_\Gamma.$$

The last line uses the fact that $\phi = \psi$ and $\pi^0\phi = 0$ on Γ. We obtain

$$-(A\partial_n u, \psi) = (f, \phi - \pi^0\phi) - a(U, \phi - \pi^0\phi).$$

If we define $\pi_\delta\phi = \pi^0\phi - \pi\phi$, we get an estimate of a linear functional of the normal derivative of the *true solution*

$$-(A\partial_n u, \psi) = (f, \phi - \pi\phi) - a(U, \phi - \pi\phi) + (f, \pi_\delta\phi) - a(U, \pi_\delta\phi). \quad (4.3.3)$$

Notice that $\pi_\delta\phi$ is nonzero only on elements adjacent to the boundary.

To estimate a linear functional of the error in the flux on Γ, we add $(A\partial_n U, \psi)_\Gamma$ to both sides and obtain the formal error representation,

Theorem 4.3.1. *The error $e = u - U$ satisfies*

$$-(A\partial_n e, \psi) = (f, \phi - \pi\phi) - a(U, \phi - \pi\phi) + (f, \pi_\delta\phi) - a(U, \pi_\delta\phi) + (A\partial_n U, \phi)_\Gamma. \tag{4.3.4}$$

4.3.2 Adaptive error control

Following the approach described in section 3.3.5, we write (4.3.4) as

$$|(A\partial_n e, \psi)| \leq \sum_{K in T_h} \Big| (f, \phi - \pi\phi)_K - a(U, \phi - \pi\phi)_K + (f, \pi_\delta\phi)_K - a(U, \pi_\delta\phi)_K + (A\partial_n U, \phi)_{K\cap\Gamma} \Big|,$$

with the obvious notation for localizing the forms to elements K. The right hand side provides a computable estimate after approximating ϕ. When the estimate is larger than the stated tolerance, an element K is marked for refinement when the local element indicator

$$\eta_K = \Big| (f, \phi - \pi\phi)_K - a(U, \phi - \pi\phi)_K + (f, \pi_\delta\phi)_K - a(U, \pi_\delta\phi)_K + (A\partial_n U, \phi)_{K\cap\Gamma} \Big|, \tag{4.3.5}$$

is larger than a local tolerance, typically the tolerance divided by the current number of elements.

4.3.3 Smoothing the boundary data

The representation (4.3.4) is optimal in the case that $\Gamma = \Gamma_D$ because ϕ has the prerequisite regularity for $\phi - \pi\phi$ to yield $O(h^2)$ estimates. However, in general (4.3.4) does not give $O(h^2)$ convergence when $\Gamma \neq \Gamma_D$. For example, the Dirichlet portion of the boundary conditions for the quantity of interest equal to the average error $\frac{1}{|\Gamma|}\int_\Gamma A\partial_n e \, ds$, namely

$$\begin{cases} \phi = 1/|\Gamma|, & x \in \Gamma, \\ \phi = 0, & x \in \Gamma_D \backslash \Gamma, \end{cases}$$

places discontinuities in the boundary data for the adjoint. Since the data is only in $H^{1/2-\epsilon}(\Gamma_D)$, for $\epsilon > 0$, ϕ is no longer in $H^2(\Omega)$. Consequently, $\phi - \pi\phi$ no longer yields $O(h^2)$ accuracy and it becomes harder to compute ϕ accurately.

We replace the ideal Dirichlet portion of the boundary conditions,

$$\begin{cases} \phi = \psi, & x \in \Gamma, \\ \phi = 0, & x \in \Gamma_D \backslash \Gamma, \end{cases} \tag{4.3.6}$$

by

$$\phi = \hat{\psi}, \quad x \in \Gamma_D, \tag{4.3.7}$$

where $\hat{\psi} \in H^{3/2}(\Gamma_D)$, $\hat{\psi} \approx \psi$ on Γ, and $\hat{\psi} \approx 0$ on $\Gamma_D \backslash \Gamma$. Note that this means we do not compute the quantity of interest exactly. The gain is a smoother Green's function that is easier to compute.

There is no unique way to construct $\hat{\psi}$. In §4.5, we consider a couple of different choices for a model problem and demonstrate that the choice of $\hat{\psi}$ can have a strong impact on the generalized Green's function and the effective domain of dependence. In addition, the closer we choose $\hat{\psi}$ to the nonsmooth data (4.3.6), the more difficult it is to approximate the Green's function.

4.4 Comparison with previous techniques

4.4.1 The Boundary-flux approach

In this section, we describe the post-processing techniques developed by Wheeler and Carey [34, 20, 55] to compute boundary fluxes, and show how the generalized Green's function may be used to prove a high order of accuracy for the recovered flux on unstructured meshes.

We begin with the analysis in $\mathbb{R}$ as this case has fewer technical details than $\mathbb{R}^n$. Let $\Omega = (0,1)$, and let T_h be a partition of Ω into $N-1$ subintervals. Define $S_h \subset H^1(\Omega)$ to be the space of continuous, piecewise linear polynomials on T_h. We use $S_{h,0}$ to denote the subset of S_h consisting of functions which are zero on the boundary.

Consider the elliptic problem,

$$\begin{cases} -\frac{d}{dx}\left(A\frac{du}{dx}\right) + b\frac{du}{dx} + cu = f, & x \in (0,1), \\ u = \alpha, & x = 0, \\ u = \beta, & x = 1. \end{cases} \tag{4.4.1}$$

The finite element method seeks $U \in S_h$ such that $U(0) = \alpha$, $U(1) = \beta$, and

$$a(U,v) = (f,v), \quad \forall v \in S_{h,0},$$

where

$$a(u,v) = \int_0^1 \left(A\frac{du}{dx}\cdot\frac{dv}{dx} + b\frac{du}{dx}\cdot v + cu\cdot v \right) dx.$$

Suppose we are interested in the boundary flux at $x = 1$. Let v be an arbitrary element of S_h. If we multiply (4.4.1) by v and integrate by parts, we have

$$-A\frac{du}{dx}v\Big|_{x=0}^{x=1} + a(u,v) = (f,v). \tag{4.4.2}$$

Now, suppose v_N is the basis function corresponding to $x = 1$. Then $v_N(1) = 1$, and $v_N = 0$ at every other node in the mesh. This gives

$$-A(1)\frac{du}{dx}(1) = (f,v_N) - a(u,v_N). \tag{4.4.3}$$

Of course, u is still unkown, so we define σ such that

$$-\sigma = (f,v_N) - a(U,v_N),$$

where U is the finite element solution.

Comparing (4.4.2) and (4.4.3), it appears that σ should give an approximation of the normal derivative. To prove that this is the case, let ϕ solve the adjoint problem

$$\begin{cases} -\frac{d}{dx}\left(A\frac{d\phi}{dx}\right) - b\frac{d\phi}{dx} + \left(c - \frac{db}{dx}\right)\phi = 0, & x \in (0,1), \\ \phi = 0, & x = 0, \\ \phi = 1, & x = 1. \end{cases} \tag{4.4.4}$$

To derive an error representation, we proceed as in §4.3,

$$-A(1)\frac{du}{dx}(1) = (f, \phi - \pi\phi) - a(U, \phi - \pi\phi) + (f, \pi_\delta\phi) - a(U, \pi_\delta\phi),$$

where $\pi_\delta\phi$ is nonzero only on the $N-1$ subinterval. Thus, $\pi_\delta\phi$ is a multiple of v_N and we can use (4.4.3) to substitute σ,

$$-A(1)\frac{du}{dx}(1) = (f, \phi - \pi\phi) - a(U, \phi - \pi\phi) - \sigma.$$

Moving σ to the other side, we have

$$-\left(A(1)\frac{du}{dx}(1) - \sigma\right) = (f, \phi - \pi\phi) - a(U, \phi - \pi\phi). \tag{4.4.5}$$

We take the absolute value of each side,

$$\left|\left(A(1)\frac{du}{dx}(1) - \sigma\right)\right| = \left|(f, \phi - \pi\phi) - a(U, \phi - \pi\phi)\right|,$$

and apply an interpolation result to prove the following theorem.

Theorem 4.4.1. *The error in the recovered boundary flux, σ, defined by (4.4.3), is bounded*

$$\left|\left(A(1)\frac{du}{dx}(1) - \sigma\right)\right| \leq Ch^2\|f\|_0 \cdot \|\phi\|_2.$$

We can also define σ to be the recovered flux at $x = 0$ and derive a similar result.

Remark 4.4.1. *If A is constant, then the solution to the adjoint problem (4.4.4) is a linear function. Therefore, $\phi - \pi\phi$ is zero everywhere, and the error in the recovered flux should be zero. In practice, quadrature errors in evaluating either $a(u,v)$ or (f,v) affect the accuracy, so σ may not be exact.*

Example 4.4.1. *We solve (4.4.1) with*

$$A = 1 + x, \quad b = 1, \quad c = 2,$$

$$f = \big((1+x)(4\pi^2) + 2\big)\cos(4\pi x),$$

and boundary conditions $u(0) = u(1) = 1$. The exact solution to this problem is $u = \cos(4\pi x)$. In Table 4.1 we compare the error in the finite element flux and the recovered boundary flux at $x = 0$ and at $x = 1$. We see that

h	$\lvert u'(0) - U'(0)\rvert$	$\lvert u'(0) - \sigma(0)\rvert$	$\lvert u'(1) - U'(1)\rvert$	$\lvert u'(1) - \sigma(1)\rvert$
0.1	7.1138	1.1903e-2	6.8154	2.7988e-3
0.01	7.9103e-1	1.3241e-4	7.8725e-1	3.1196e-5
0.001	7.8981e-2	1.3253e-6	7.8943e-2	3.1225e-7
0.0001	7.8959e-3	1.2888e-8	7.8956e-3	3.1729e-9

Table 4.1: Comparison the error in the finite element flux and the recovered boundary flux at $x = 0$ and at $x = 1$ in Example 4.4.1.

the finite element derivative converges $O(h)$ at both endpoints, while the recovered flux converges $O(h^2)$.

Complications arise in higher dimensions because of the geometry, e.g. the domain may have corners. In general, $A\nabla u \cdot \boldsymbol{n}$ is discontinuous at a corner, even if u is smooth, because the normal vector is discontinuous. In this case, we need to allow σ to be discontinuous as well. We address this situation in $\mathbb{R}^2$ and the extension to $\mathbb{R}^3$ follows easily.

Let Ω be a bounded set in $\mathbb{R}^2$, and consider the elliptic problem

$$\begin{cases} -\nabla \cdot (A\nabla u) + \mathbf{b} \cdot \nabla u + cu = f, & x \in \Omega, \\ u = 0, & x \in \partial\Omega. \end{cases} \tag{4.4.6}$$

If the boundary does not have a corner and if u is smooth, then $A\nabla u \cdot \boldsymbol{n}$ is smooth as well. If this is the case, we define $N_{\partial\Omega}$ to be the set of nodes on the boundary, and B_h the space of continuous piecewise polynomials on $\partial\Omega$. To be clear, a function $v \in B_h$ is a linear combination of the basis functions associated with the nodes in $N_{\partial\Omega}$.

Let $U \in S_h$ be the finite element function solving

$$a(U, v) = (f, v), \ \forall v \in S_h,$$

where S_h is defined as in §4.2, and

$$a(u, v) = \int_\Omega (A\nabla u \cdot \nabla v + \mathbf{b} \cdot \nabla u \ v + c \ u \ v) \ dx.$$

The boundary flux method seeks $\sigma \in B_h$ such that

$$-(\sigma, v)_{\partial\Omega} = (f, v) - a(U, v), \tag{4.4.7}$$

for all $v \in B_h$.

Next, we consider the case where a corner represents the intersection of a Dirichlet boundary, Γ_1 and a Neumann boundary, Γ_2. Let v_c denote the basis function associated with this corner node. As we can see in Fig. 4.1, v_c has support on the Neumann boundary as well. Let g_N denote the Neumann condition posed on Γ_2. We define N_{Γ_1} to be the set of nodes on Γ_1, and $B_{h,1}$ the space of continuous piecewise polynomials on Γ_1. We define $\sigma_1 \in B_{h,1}$ such that

$$-(\sigma_1, v)_{\Gamma_1} = (f, v) - a(U, v) + (g_N, v)_{\Gamma_2},$$

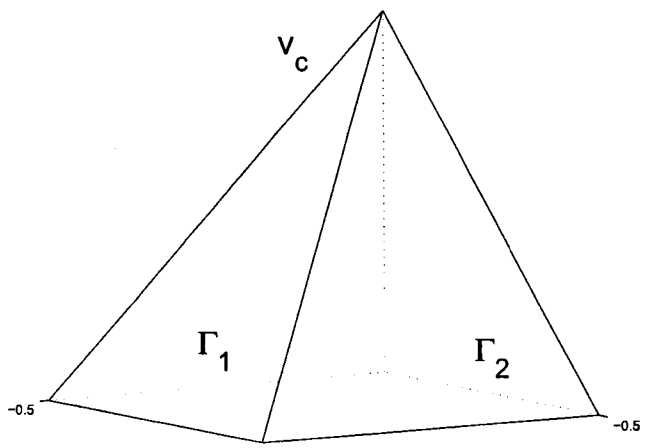

Figure 4.1: A piecewise linear basis function associated with a corner node.

for all $v \in B_{h,1}$. We note that the only basis function with support on Γ_2 is v_c, so the additional term on the right hand side is zero for most $v \in B_{h,1}$.

Finally, we consider the case where two segments, Γ_1 and Γ_2, with Dirichlet conditions meet at a corner, x_c. Suppose we are interested in recovering the boundary flux along Γ_1. As before, we denote this boundary flux σ_1. If we already know the recovered flux on Γ_2, σ_2, then we can define N_{Γ_1} and $B_{h,1}$ as in the previous case, and seek $\sigma_1 \in B_{h,1}$ such that

$$-(\sigma_1, v)_{\Gamma_1} = (f, v) - a(U, v) + (\sigma_2, v)_{\Gamma_2},$$

for all $v \in B_{h,1}$. Unfortunately, we often do not know σ_2 in advance, and computing σ_2 would similarly require σ_1.

As an alternative, let g_{D,Γ_1} and g_{D,Γ_2} represent the Dirichlet data on Γ_1 and Γ_2 respectively. Now, let $\mathbf{t}_1$ and $\mathbf{t}_2$ denote the tangent vectors to Γ_1 and Γ_2 respectively at x_c. We can easily compute $\nabla g_{D,\Gamma_1}(x_c) \cdot \mathbf{t}_1$ and $\nabla g_{D,\Gamma_2}(x_c) \cdot \mathbf{t}_2$ either analytically or numerically, and use these values to

reconstruct $\nabla u(x_c) \cdot \boldsymbol{n}_1$ and $\nabla u(x_c) \cdot \boldsymbol{n}_2$ at x_c where $\boldsymbol{n}_1$ and $\boldsymbol{n}_2$ represent the normal vectors to Γ_1 and Γ_2 at x_c.

Now we use N_{Γ_1} and $B_{h,1}$ as before, but we also define $B^0_{h,1}$ to be the space of functions in $B_{h,1}$ which are also zero at x_c. The boundary flux method seeks $\sigma_1 \in B_{h,1}$ such that $\sigma_1(x_c)$ is the reconstructed value using the derivatives of the Dirichlet data, and

$$-(\sigma_1, v)_{\Gamma_1} = (f, v) - a(U, v),$$

for all $v \in B^0_{h,1}$.

Example 4.4.2. *Let $\Omega = (0,1) \times (0,1)$, and consider the elliptic problem*

$$\begin{cases} -\nabla \cdot (\nabla u) = 8\pi^2 \sin(2\pi x)\sin(2\pi y), & x \in \Omega, \\ u = 0, & x \in \partial\Omega, \end{cases}$$

with exact solution $u = \sin(2\pi x)\sin(2\pi y)$. We use the technique described above when boundary segments with Dirichlet condition meet at a corner to compute the boundary flux along $x = 0$, $0 \leq y \leq 1$. In Table 4.2, we see

h	$\|\nabla u \cdot \boldsymbol{n} - \nabla U \cdot \boldsymbol{n}\|$	$\|\nabla u \cdot \boldsymbol{n} - \sigma\|$
0.1	1.5691	3.4639e-1
0.05	8.0070e-1	8.0524e-2
0.025	4.0273e-1	1.9358e-2
0.0125	2.0168e-1	4.6571e-3

Table 4.2: A comparison of the error in the finite element flux and the recovered boundary flux along $x = 0$ in Example 4.4.2.

that the error in the finite element flux converges only $O(h)$, while the error in the recovered flux converges $O(h^2)$.

This post-processing technique can also be used to estimate a linear functional along the boundary and to guide adaptivity. The boundary-flux

method finds a piecewise polynomial approximation to $A\partial_n u$ rather than an estimate of a functional. However, if $\pi_\delta u \in W_h$ and $\phi = \pi_\delta u = \psi$ on $\partial\Omega$,

$$-(\sigma - A\partial_n u, \psi)_{\partial\Omega} = (f, \phi - \pi\phi) - a(U, \phi - \pi\phi).$$

In other words, σ provides an estimate of a functional of the boundary flux,

$$-(\sigma, \psi)_{\partial\Omega} \approx -(A\partial_n u, \psi)_{\partial\Omega},$$

if $(f, \phi - \pi\phi) - a(U, \phi - \pi\phi)$ decays away from $\partial\Omega$ rapidly. If we use $(\sigma - A\partial_n u)_{\partial\Omega}$ as an error indicator for refinement, only those element next to the boundary are refined. If $(f, \phi - \pi\phi) - a(U, \phi - \pi\phi)$ does not decay away from $\partial\Omega$ rapidly, then those terms eventually become dominant.

The recovery technique developed by Larson et al [35] is essentially the same as the boundary-flux method. The difference is that the boundary-flux method results in a piecewise polynomial approximation of the normal flux with support on the elements touching the boundary, whereas the Larson et al technique provides an estimate of a linear functional of the normal derivative.

4.4.2 Extraction Function Approach

The adjoint method is closely related to another post-processing technique developed by I. Babuska and A. Miller [7, 8, 6]. This technique uses a so-called extraction function to estimate a linear functional of the true solution. It turns out that the extraction function is actually the generalized Green's function, although it is not described in this way. A close inspection reveals their approach provides an alternative method for solving the adjoint problem. Consider the adjoint problem

$$\begin{cases} L^*\phi = 0, & x \in \Omega, \\ \phi = \psi, & x \in \partial\Omega. \end{cases} \tag{4.4.8}$$

We choose a function ϕ_B such that $\phi_B = \psi$ on $\partial\Omega$ and ϕ_B decays rapidly in Ω as we move away from the boundary. We then write the adjoint solution as $\phi = \phi_I + \phi_B$. Since $\phi_B = \psi$ on the boundary, ϕ_I is zero on the boundary and therefore $\phi_I \in H_0^1(\Omega)$. In general, ϕ_I is still unknown, but we insert $\phi = \phi_I + \phi_B$ into (4.4.8) to obtain

$$\begin{cases} L^*\phi_I = -L^*\phi_B, & x \in \Omega, \\ \phi_I = 0, & x \in \partial\Omega. \end{cases} \tag{4.4.9}$$

Finally, we use ϕ to obtain an *a-posteriori* estimate which corresponds to choosing $\phi_B \pi_\delta \phi$.

Babuska and Miller consider Laplace's equation on a simple domain, in which case it is possible to get precise estimates on ϕ_I using classical Green's function theory. In general, we could solve (4.4.9) instead of (4.4.8) in the hope that it might be easier to compute an accurate solution.

4.5 Numerical results

In this section, we present some computations to illustrate the preceding discussion.

4.5.1 Square domain

For the first problem, we consider the solution of

$$\begin{cases} -\Delta u = f(x), & x \in \Omega, \\ u = 0, & x \in \Gamma_D = \partial\Omega. \end{cases} \tag{4.5.1}$$

The exact quantity of interest is

$$\int_\Gamma \nabla u \cdot n \, ds,$$

where Γ is the segment $x_1 = 1$, $1/3 \leq x_2 \leq 2/3$. Note that the exact value of the quantity of interest is 0. We obtain approximations corresponding to

three smoothed adjoint data

$$\psi_1(x) = \frac{1}{3 \times 1.04027} \left(\arctan(200(x_2 - 1/3)) - \arctan(200(x_2 - 2/3))\right),$$

$$\psi_2(x) = \frac{1}{3 \times 0.97816} \left(\arctan(20(x_2 - 1/3)) - \arctan(20(x_2 - 2/3))\right),$$

$$\psi_3(x) = 6x_2(1 - x_2),$$

each normalized to have integral norm 1. We plot the smoothed adjoint data in Fig. 4.2. The data ψ_1 provides a very close approximation to the

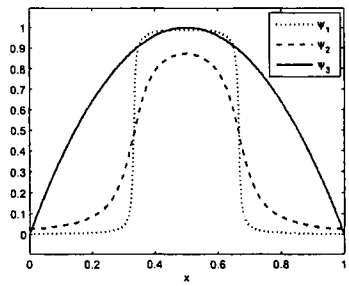

Figure 4.2: Plot of the adjoint data ψ_1, ψ_2, ψ_3 along the line $x = 0$, $0 \leq y \leq 1$.

ideal discontinuous ψ defining the exact quantity of interest. The data ψ_2 yields an average of the normal derivative over a larger region however the corresponding adjoint solution is significantly smoother.

In the first set of computations, we use

$$f(x) = 16\pi^2 \sin(2\pi x_1) \sin(2\pi x_2),$$

so $u(x) = \sin(2\pi x_1) \sin(2\pi x_2)$. In Fig. 4.3, we plot the adaptive meshes obtained by adaptive refinement after 5 refinement levels. The localized nature of the refinement needed to resolve the desired information is clearly visible. On the left in Fig. 4.4, we plot the effectivity index (error estimate/true

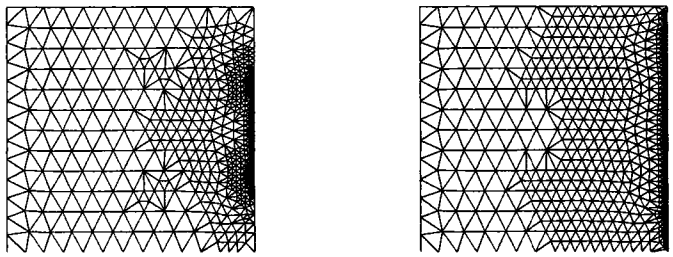

Figure 4.3: Adaptive meshes corresponding to ψ_1 (left) and ψ_2 (right).

error) for the three computations. The smoother ψ approximations yield much more accurate error estimates on coarser meshes, whereas using ψ_1 requires a mesh that is fine near the layers in that function. In the context of uniform meshes, using ψ_1 to approximate the quantity of interest means having to use a much finer mesh.

On the right in Fig. 4.4, we plot the final adaptive mesh when the function f is perturbed by an approximate delta function located away from the boundary and ψ_1 is the data for the adjoint problem. We see that the adaptive mesh refinement yields a mesh that is refined near the region where the forcing term is concentrated and the region where the desired information is computed. The former results from large residuals in the forward approximation and the latter results from large adjoint weights.

4.5.2 L-shaped domain

We next consider Laplace's equation defined on an L-shaped domain shown in Fig. 4.5. We impose homogeneous Dirichlet boundary conditions along the coordinate axis and choose Neumann boundary conditions on the remaining boundaries so that the true solution is $u(r,\theta) = r^{2/3}\sin(2\theta/3)$ in polar coordinates. The quantity of interest is the average value of the normal derivative $\frac{1}{|\Gamma_D|}\int_{\Gamma_D} \nabla u \cdot n \, ds$.

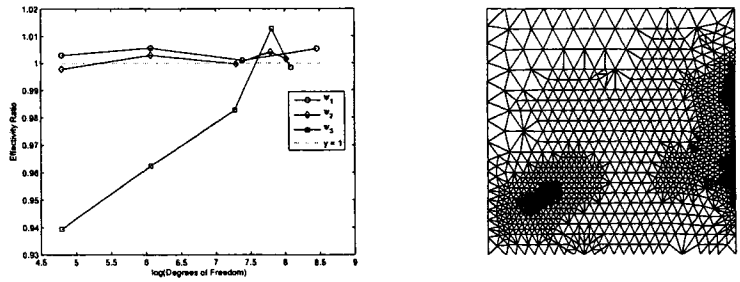

Figure 4.4: On the left, a plot of the effectivity indices for ψ_1, ψ_2, ψ_3. On the right, we plot an adapted mesh for a problem with a highly localized forcing and data ψ_1 for the adjoint.

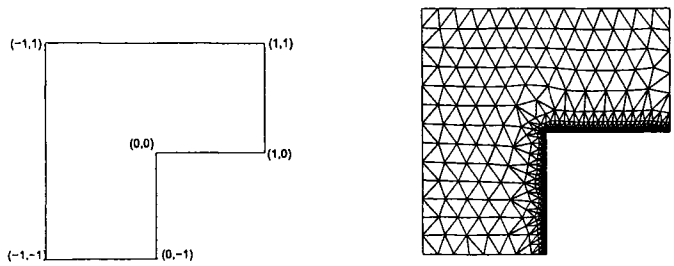

Figure 4.5: One the right, we illustrate the L-shaped domain. On the left, we plot the final adaptive mesh obtained using the adjoint problem (4.5.2).

In the first computation, we choose Γ_D to be the line segment along $x_1 = 0$, $-1 \leq x_2 \leq 0$ and $0 \leq x_1 \leq 1$ and $x_2 = 0$. The adjoint problem is

$$\begin{cases} -\Delta\phi = 0, & x \in \Omega, \\ \phi = 1/2, & x \in \Gamma_D, \\ \partial_n\phi = 0, & x \in \partial\Omega \backslash \Gamma_D, \end{cases} \tag{4.5.2}$$

which has exact solution $\phi = 1/2$. This implies that the adjoint weight $\phi - \pi\phi$ is negligible away from Γ_D. Table 4.3 displays the details of the error estimation. We obtain accurate estimates at all refinement levels. Figure 4.5 shows the final adaptive mesh from these calculations.

Elements	DOF	Adj. Est.	True Error	Effect. Ratio
242	156	-0.127067	-0.126349	1.0057
341	213	-0.080861	-0.080681	0.9998
531	322	-0.050952	-0.050907	1.0009
903	535	-0.032048	-0.032037	1.0003
1639	956	-0.020159	-0.020156	1.0001

Table 4.3: Error estimates and effectivity ratios for the L-shaped domain problem.

Now, we test the boundary-flux technique by using

$$-(\partial_n u, 1/2)_{\Gamma_D} \approx \sum_{K \cap \Gamma_D \neq \emptyset} (f, v)_K - (\nabla U, \nabla v) + (\partial_n U, 1/2)_{\Gamma_D}.$$

The error estimation results are given in Table 4.4. We can adapt the mesh along the boundary using the resulting estimate and then refine the mesh in the interior to match. In this special case, the boundary-flux technique compares favorably with the estimates using the adjoint problem.

Elements	DOF	B-Flux Est.	True Error	Effect. Ratio
242	156	-0.126342	-0.126349	0.9999
341	213	-0.080679	-0.080681	0.9999
531	322	-0.050907	-0.050907	1.0000
903	535	-0.032036	-0.032037	1.0000
1639	956	-0.020156	-0.020156	1.0000

Table 4.4: Error estimates and effictivity ration using the boundary-flux technique.

Next, we consider two different smooth approximations for the data ψ in the L-shaped domain problem. The quantity of interest is the normal flux across the line segment

$$\{x \mid x_2 = 0,\ 0 \leq x_1 \leq 1\}.$$

The first approximation to the discontinuous data

$$\psi = \begin{cases} 1, & x_2 = 0,\ 0 \leq x_1 \leq 1, \\ 0, & x_1 = 0,\ 0 \leq x_2 \leq 1, \end{cases}$$

we use

$$\psi_1 = \frac{1}{2} + \frac{1}{\pi}\arctan(100(x_2 + 0.1)),$$

which is nearly one when $x_2 = 0$ and drops off rapidly as x_2 decreases. For this problem, the exact normal flux is known, so we may use the functional

$$(\partial_n u, \psi_1)_{\Gamma_D} = -1.185761,$$

to compare with the adjoint estimates in Table 4.5. The final adaptive mesh is given is Figure 4.6.

Elements	DOF	Adj. Est.	True Error	Effect. Ratio
242	156	-0.239095	-0.210139	1.1378
881	494	-0.145261	-0.141630	1.0256
1523	829	-0.092468	-0.092524	0.9994
2171	1179	-0.057318	-0.056936	1.0067

Table 4.5: Error estimates and effectivity ratios using ψ_1

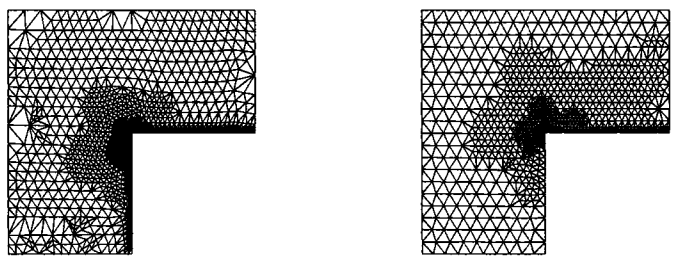

Figure 4.6: Plots of the final adaptive meshes corresponding to ψ_1 (left) and ψ_2 (right).

Next, we consider

$$\psi_2 = \frac{2}{\pi}\arctan(20x_1).$$

Using ψ_1 means regularizing the Dirichlet data on $\Gamma_D \backslash \Gamma$ so that $\phi \approx 1$ on Γ. Using ψ_2 means regularizing the data on the segment of interest. Using

the analytic solution, we evaluate

$$(\partial_n u, \psi_2) = -0.792209,$$

and summarize the results in Table 4.6. The final adaptive mesh is given in Figure 4.6.

Elements	DOF	Adj. Est.	True Error	Effect. Ratio
583	351	-0.024659	-0.033015	0.7469
1221	678	-0.012221	-0.015601	0.7834
1664	911	-0.005765	-0.007134	0.8081
2130	1166	-0.005765	-0.007134	0.8355
2764	1524	-0.001715	-0.001925	0.8906

Table 4.6: Error estimates and effectivity ratios using ψ_2

Chapter 5

OPERATOR DECOMPOSITION METHODS

5.1 Introduction

Operator decomposition methods are an attractive solution strategy for computing complex phenomena involving multiple physical processes, multiple scales or multiple domains. The general strategy is to decompose the problem into components involving simpler physics over a relatively limited range of scales, and then to seek the solution of the entire system through an iterative procedure involving solutions of the individual components. This approach is appealing because there is generally a good understanding of how to solve a broad spectrum of single physics problems accurately and efficiently, and because it provides an alternative to accommodating multiple scales in one discretization.

The trouble with this approach is that it defines a fixed point problem which may not converge. In this chapter, we investigate a few simple iterative procedures for operator decomposition, and, in special circumstances, we provide conditions for convergence. We use these results to derive a relaxation scheme and a Newton method to force the fixed point problem to converge in all situations. Numerical examples are provided throughout the chapter.

5.2 Linear Algebraic Systems

Consider the system

$$\begin{cases} A_{11}x_1 + A_{12}x_2 = b_1, \\ A_{21}x_1 + A_{22}x_2 = b_2, \end{cases} \tag{5.2.1}$$

with $A_{11} \in \mathbb{R}^{N\times N}$, $A_{12} \in \mathbb{R}^{N\times M}$, $A_{21} \in \mathbb{R}^{M\times N}$, $A_{22} \in \mathbb{R}^{M\times M}$, $x_1, b_1 \in \mathbb{R}^N$, and $x_2, b_2 \in \mathbb{R}^M$. This is equivalent to solving the full system $Ax = b$ where

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \quad x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}.$$

It is important to note that even if A is invertible, the individual blocks may not be. Suppose we can solve

$$A_{11}x_1 = \text{data}, \quad \text{and} \quad A_{22}x_2 = \text{data}, \tag{5.2.2}$$

but solving the full system is either impractical or impossible.

On the other hand, suppose A_{11} and/or A_{22} are invertible, but A is not. Determining conditions for the invertibility of A in terms of the off-diagonal blocks is far more difficult than it appears.

Example 5.2.1. *Suppose $A_{11}, A_{22} \in \mathbb{R}^{N\times N}$ are symmetric positive definite matrices. If we assume A_{12} and A_{21} are symmetric positive definite, can we guarantee a unique solution? The answer is no. Take $A_{12} = A_{22}$ and $A_{21} = A_{11}$. The full system has an eigenvalue of 0 with algebraic multiplicity of N!*

Example 5.2.2. *Suppose the linear system may be written*

$$\begin{pmatrix} A_{11} & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

where $A_{11} \in \mathbb{R}^{N\times N}$ is symmetric positive definite, and $B \in \mathbb{R}^{M\times N}$. Existence and uniqueness of solutions can be guaranteed if and only if B satisfies the inf-sup *condition*

$$\inf_{x\in\mathbb{R}^M} \sup_{y\in\mathbb{R}^N} \frac{x^T B y}{\|x\|\|y\|} \geq \gamma > 0$$

.

Theorem 5.2.1. *Suppose $A_{11} \in \mathbb{R}^{N\times N}$ and $A_{22} \in \mathbb{R}^{N\times N}$ are invertible matrices. Then the full matrix,*

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix},$$

is invertible if and only if neither

$$G_1 = A_{22}^{-1}A_{21}A_{11}^{-1}A_{12}, \tag{5.2.3}$$

$$G_2 = A_{11}^{-1}A_{12}A_{22}^{-1}A_{21}, \tag{5.2.4}$$

have an eigenvalue of 1.

Proof. Suppose A is not invertible. Then A has an eigenvalue of 0. Hence, there exist a nonzero $x = (x_1, x_2)^T$ such that $Ax = 0$. Equivalently,

$$A_{11}x_1 + A_{12}x_2 = 0,$$

$$A_{21}x_1 + A_{22}x_2 = 0.$$

We solve for x_1 in the first equation,

$$x_1 = -A_{11}^{-1}A_{12}x_2,$$

and substitute into the second and solve for x_2

$$x_2 = A_{22}^{-1}A_{21}A_{11}^{-1}A_{12}x_2.$$

Therefore, $x_2 = G_1x_2$ and $\lambda = 1$ is an eigenvalue of G_1 with eigenvector x_2.

Similarly, we can solve for x_2 in the second equation, substitute into the first equation and solve for x_1. This gives $x_1 = G_2x_1$. Therefore, $\lambda = 1$ is an eigenvalue of G_2 with eigenvector x_1. □

Remark 5.2.1. *Numerical experiments have demonstrated that if $N \geq M$, and λ is an eigenvalue of G_2, then either λ is an eigenvalue of G_1, or $\lambda = 0$. An analogous result seems to hold when $M \geq N$. This result remains unproven.*

In the sections to follow, we define iterative algorithms for computing the solution to (5.2.1) using only (5.2.2). In each case, the convergence of the algorithm depends on the spectral radius of G_1 and/or G_2. As we shall see, if the global system does not have a unique solution, then the algorithms do not converge since the spectral radius of both matrices is at least 1.

5.2.1 Block Jacobi Method

Assume that we have an initial guesses $x_1^{\{0\}}$ and $x_2^{\{0\}}$. To compute the solution of (5.2.1) we construct the following iterative method.

Block Jacobi Method

k = 0

while $(\|x_1^{\{k\}} - x_1^{\{k-1\}}\| + \|x_2^{\{k\}} - x_2^{\{k-1\}}\| > TOL)$ **do**

(a) k = k+1

(b) Given $x_1^{\{k-1\}}$ and $x_2^{\{k-1\}}$, solve

$$A_{11}x_1^{\{k\}} = b_1 - A_{12}x_2^{\{k-1\}}$$

$$A_{22}x_2^{\{k\}} = b_2 - A_{21}x_1^{\{k-1\}}$$

for $x_1^{\{k\}}$ and $x_2^{\{k\}}$.

end while

We use substitution to determine

$$x_1^{\{2k\}} = G_1^{\{k\}}x_1^{\{0\}} + \left(\sum_{i=0}^{k-2} G_1^{2i}\right) c_1,$$

$$x_2^{\{k\}} = G_2^{\{k\}}x_2^{\{0\}} + \left(\sum_{i=0}^{k-2} G_2^{2i}\right) c_2,$$

where $c_1 = A_{22}^{-1}(b_2 - A_{21}A_{11}^{-1}b_1)$, $c_2 = A_{11}^{-1}(b_1 - A_{12}A_{22}^{-1}b_2)$ and G_1 and G_2 are defined by (5.2.3) and (5.2.4). The convergence of this sequence depends on the spectral radius of G_1 and G_2. If $\rho(G_1) < 1$ and $\rho(G_2) < 1$, then the iterative method converges. Otherwise, the method diverges.

Example 5.2.3. *Take*

$$A_{11} = \begin{pmatrix} 1 & 2 \\ -1 & 3 \end{pmatrix}, \quad A_{12} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, \quad A_{21} = \begin{pmatrix} -1 & 2 \\ 0 & 3 \end{pmatrix}, \quad A_{22} = \begin{pmatrix} 0 & 3 \\ 2 & 1 \end{pmatrix},$$

$$b_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad b_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}.$$

The full system has solution $x_1 = [0.5, 1.5]^T$, $x_2 = [-2.5, -0.5]^T$. *We choose the initial guess* $x_2^{\{0\}} = [0, 0]^T$, *apply the block Jacobi algorithm, and the sequence converges within a tolerance of* 1×10^{-6} *after 55 iterations. In this case, the eigenvalues of both* G_1 *and* G_2 *are* $1/6$ *and* $3/5$, *so the spectral radius is less than 1.*

Example 5.2.4. *Change* A_{11} *in the previous example to*

$$A_{11} = \begin{pmatrix} 1 & 2 \\ -1 & 0 \end{pmatrix},$$

The solution to the full system is now $x_1 = [5, -3]^T$, $x_2 = [2, 4]^T$, *but the eigenvalues of both* G_1 *and* G_2 *are* $1/6$ *and* $3/2$. *We choose the initial guess,* $x_2^{\{0\}} = [0, 0]^T$, *and after 100 iterations the error in* x_2 *is* 1.84×10^8.

5.2.2 Block Gauss-Seidel Method

Assume that we have an initial guess $x_2^{\{0\}}$. To compute the solution of (5.2.1) we construct the following iterative method.

Block Gauss-Seidel Method

k = 0

while $(\|x_2^{\{k\}} - x_2^{\{k-1\}}\| > TOL)$ **do**

(a) k = k+1

(b) Given $x_2^{\{k-1\}}$, solve

$$A_{11}x_1^{\{k\}} = b_1 - A_{12}x_2^{\{k-1\}}$$

for $x_1^{\{k\}}$.

(c) Given $x_1^{\{k\}}$, solve

$$A_{22}x_2^{\{k\}} = b_2 - A_{21}x_1^{\{k\}}$$

for $x_2^{\{k\}}$.

end while

Recursively, we determine

$$x_2^{\{k\}} = G_1^k x_2^{\{0\}} + \left(\sum_{i=0}^{k-1} G_1^i\right) c_1,$$

where $c_1 = A_{22}^{-1}(b_2 - A_{21}A_{11}^{-1}b_1)$ and G_1 is defined by (5.2.3). The convergence of this sequence depends only on the spectral radius of G_1. If $\rho(G_1) < 1$, then the iterative method will converge. Otherwise, the method will diverge.

Example 5.2.5. *Take*

$$A_{11} = \begin{pmatrix} 1 & 2 \\ -1 & 3 \end{pmatrix}, \quad A_{12} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, \quad A_{21} = \begin{pmatrix} -1 & 2 \\ 0 & 3 \end{pmatrix}, \quad A_{22} = \begin{pmatrix} 0 & 3 \\ 2 & 1 \end{pmatrix},$$

$$b_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad b_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}.$$

The full system has solution $x_1 = [0.5, 1.5]^T$, $x_2 = [-2.5, -0.5]^T$. *We choose the initial guess* $x_2^{\{0\}} = [0, 0]^T$, *apply the block Gauss-Seidel algorithm, and the sequence converges within a tolerance of* 1×10^{-6} *after 29 iterations. In this case, the eigenvalues of* G_1 *are* $1/6$ *and* $3/5$, *so the spectral radius is less than 1.*

Example 5.2.6. *Change* A_{11} *in the previous example to*

$$A_{11} = \begin{pmatrix} 1 & 2 \\ -1 & 0 \end{pmatrix},$$

The solution to the full system is now $x_1 = [5, -3]^T$, $x_2 = [2, 4]^T$, *but the eigenvalues of* G_1 *are* $1/6$ *and* $3/2$. *We make the initial guess,* $x_2^{\{0\}} = [0, 0]^T$, *and after 100 iterations the error in* x_2 *is* 1.58×10^{18}. *Even if the initial guess is changed to* $x_2^{\{0\}} = [1.99, 4.01]^T$ *the error in* x_2 *after 100 iterations is* 1.98×10^{15}.

We make the observation that the block Gauss-Seidel method transforms the full linear system into a triangular system, allowing each equation to be solved in a particular order. Whether one chooses an upper triangular system or a lower triangular system affects the order in which the equations can be solved. In the above examples we solved a lower triangular system by making an initial guess for x_2 in the first equation. Alternatively, we may make an initial guess for x_1 in the second equation and solve the corresponding upper triangular system. We proceed as before to derive the sequence

$$x_1^{\{k\}} = G_2^k x_1^{\{0\}} + \left(\sum_{i=0}^{k-1} G_2^i \right) c_2,$$

where $c_2 = A_{11}^{-1}(b_1 - A_{12}A_{22}^{-1}b_2)$ and G_2 is defined by (5.2.4). Similar to before, the convergence of this sequence depends on the spectral radius of G_2. However, in Remark 5.2.1, we made the observation that in practice the spectral radius of G_1 and G_2 appear to be the same. Thus, there is no gain is switching the order of the equations.

5.3 Interface Transfer

A typical algorithm for solving a system of two coupled differential equations begins by making an initial guess of one function, say $u_1^{\{0\}}$ and using this to solve for the other unknown function, $u_2^{\{0\}}$. This approximation $u_2^{\{0\}}$ is used to construct a new approximation for the first function, $u_1^{\{1\}}$. This effectively defines an iterative procedure which may or may not converge. The naive assumption is that the approximations $u_1^{\{k\}}$ and $u_2^{\{k\}}$ converge to the true solutions as k increases.

Below, we present examples that demonstrate this is not always the case, and propose a Newton method where a Gateaux derivative of the approximate solutions with respect to the previous approximation is used to define a similar system, called the variational system, the solution of which gives the elements of the Jacobian required for Newton's method.

We begin with the simplest case, namely two Poisson equations in one dimension coupled at a point on the common boundary between two adjacent regions.

5.3.1 Convergence Criteria in 1D

In this section, we consider Poisson's equation defined on two adjacent domains with common boundary Γ. At this interface we impose continuity in the solution and continuity in the normal flux. The global problem is given by

$$\begin{cases} -\frac{\partial}{\partial x}\left(A_1\frac{\partial u_1}{\partial x}\right) = f_1(x), & x \in \Omega_1 = (0,1), \\ u_1 = \alpha, & x = 0, \\ \begin{cases} u_1 = u_2, \\ A_1\frac{\partial u_1}{\partial x} = A_2\frac{\partial u_2}{\partial x}, \end{cases} & x \in \Gamma = 1, \\ -\frac{\partial}{\partial x}\left(A_2\frac{\partial u_2}{\partial x}\right) = f_2(x), & x \in \Omega_2 = (1,2), \\ u_2 = \beta, & x = 2. \end{cases} \tag{5.3.1}$$

In addition, we assume that neither the function value, nor the normal flux at $x = 1$ is known. It is possible to write down a finite element method for the union of the two domains, [0,2], which can be solved all at once. However, we are interested in splitting the domain at the interface as a prelude to more complicated problems where different physics, and therefore different solvers, may be used on each side of the interface.

Assume we have an initial guess for the Dirichlet data, $z^{\{0\}}$. We use $z^{\{k\}}$ to avoid confusion with either $u_1^{\{k\}}$ or $u_2^{\{k\}}$, which may be different. To compute a numerical solution of (5.3.1) we construct the following iterative operator decomposition method.

Interface Operator Decomposition Method

k = 0

while $(\|z^{\{k\}} - z^{\{k-1\}}\| > TOL)$ **do**

(a) k = k+1

(b) Solve

$$\begin{cases} -\frac{\partial}{\partial x}\left(A_1 \frac{\partial u_1^{\{k\}}}{\partial x}\right) = f_1(x), & x \in \Omega_1 = (0,1), \\ u_1^{\{k\}} = \alpha, & x = 0, \\ u_1^{\{k\}} = z^{\{k-1\}}, & x = 1, \end{cases} \tag{5.3.2}$$

for $u_1^{\{k\}}$.

(c) Solve

$$\begin{cases} -\frac{\partial}{\partial x}\left(A_2 \frac{\partial u_2^{\{k\}}}{\partial x}\right) = f_2(x), & x \in \Omega_2 = (1,2), \\ A_2 \frac{\partial u_2^{\{k\}}}{\partial x} = A_1 \frac{\partial u_1^{\{k\}}}{\partial x}, & x = 1, \\ u_2^{\{k\}} = \beta, & x = 2, \end{cases} \tag{5.3.3}$$

for $u_2^{\{k\}}$.

(d) Set $z^{\{k\}} = u_2^{\{k\}}(1)$.

end while

This algorithm is similar to the block Gauss-Seidel algorithm described in the previous section.

In practice, we discretize (5.3.2) and (5.3.3) using the continuous piecewise linear finite element method with constant mesh parameters h and k respectively.

Proposition 5.3.1. *The above iterative procedure converges for finite element solutions if $A_1 < A_2$, is a 2-cycle if $A_1 = A_2$, and diverges if $A_1 > A_2$. This is independent of mesh size.*

Proof. We discretize (5.3.2) and (5.3.3) giving the linear system

$$AU^{\{1\}} = b^{\{0\}}$$

where

$$A = \begin{bmatrix} 1 & 0 & 0 & \cdots & & & & & & & \\ -A_1/h & 2A_1/h & -A_1/h & 0 & \cdots & & & & & & \\ 0 & \ddots & \ddots & \ddots & & & & & & & \\ \vdots & \cdots & -A_1/h & 2A_1/h & -A_1/h & 0 & \cdots & & & & \\ & & & 0 & 1 & 0 & 0 & & & & \\ & & & -A_1/h & A_1/h & A_2/k & -A_2/k & & & & \\ & & & 0 & 0 & -A_2/k & 2A_2/k & -A_2/k & & & \\ & & & & & \ddots & \ddots & \ddots & \ddots & & \\ & & & & & & & -A_2/k & 2A_2/k & -A_2/k \\ & & & & & & & 0 & 0 & 1 \end{bmatrix}$$

$$b^{\{0\}} = [\alpha, (f, v_2), \ldots, (f, v_n), z^{\{0\}}, (f, v_{n+2}), \ldots, \beta]^T,$$

$$U^{\{1\}} = [U_1^{\{1\}}, U_2^{\{1\}}, \ldots, U_{n+m+2}^{\{1\}}]^T.$$

Then we note that $z^{(1)} = U_{n+2}^{(1)}$, ie. the new Dirichlet value is taken from

Then we note that $z^{(1)} = U_{n+2}^{(1)}$, ie. the new Dirichlet value is taken from the weight on the first basis function on [1,2].

We know that (5.3.2) and (5.3.3) are well-posed problems, so A is invertible and we can write

$$z^{\{1\}} = U_{n+2}^{\{1\}} = A^{-1}(n+2,:)b^{\{0\}}$$

where $A^{-1}(n+2,:)$ denotes the $n+2$ row of A^{-1}. Similarly, we have

$$z^{(k+1)} = A^{-1}(n+2,:)b^{\{k\}}.$$

Since none of the components of the load vector change except the Dirichlet value, we can write

$$b^{\{k+1\}} = Cb^{\{k\}}$$

where

$$C = \begin{bmatrix} 1 & 0 & \dots & & & \\ 0 & \ddots & \ddots & & & \\ \vdots & & 1 & & & \\ c_1 & c_2 & \dots & c_{n+1} & & c_{2n+2} \\ & & 0 & 1 & 0 & \\ & & & & \ddots & \\ & & & & 0 & 1 \end{bmatrix}$$

and $A^{-1}(n+2,:) = [c_1, c_2, \dots, c_{2n+2}]$ as the $n+1$ row of C.

Recursively, we easily see that

$$b^{\{k+1\}} = C^k b^{\{0\}}.$$

This gives a simply way to determine the convergence of $b^{\{k\}}$ in terms of the eigenvalues of C. We compute these eigenvalues to be

$$\lambda_1 = c_{n+1}, \quad \lambda_{2\dots 2n+2} = 1.$$

The convergence of the fixed point problem depends only on the eigenvalue corresponding to the value on the interface. Therefore we only need to look at c_{n+1}, corresponding to the entry in the n+2 row and n+1 column of A^{-1}, to determine the convergence.

It is not practical to compute the entire inverse matrix for one entry, so we use the formula for the classical adjoint along with the relationship

between this matrix and the inverse to find c_{n+1}. We can find the (i,j) component of A^{-1} by

$$A^{-1}(i,j) = (-1)^{i+j}\frac{1}{\det(A)}\det(A(j',i'))$$

where $A(j',i')$ denotes the matrix obtained by deleting the j row and i column of A. A tedious calculation shows that

$$\det(A) = \frac{nA_1^nA_2^{n-1}}{k^{2n-1}},$$

and

$$\det(A(j',i')) = \frac{nA_1^{n-1}A_2^n}{k^{2n-1}}.$$

Thus, the critical eigenvalue of C is

$$\lambda_1 = -\frac{A_1}{A_2}.$$

This value corresponds to the (n+2,n+1) entry of A^{-1}, which provides the minus sign. From this, we conclude that the iteration converges if $A_1 < A_2$, diverges if $A_1 > A_2$, and enters a 2-cycle if $A_1 = A_2$. □

Example 5.3.1. *To demonstrate the results in Proposition 5.3.1, we choose appropriate boundary conditions away from the interface and data such that the exact solution is known. Next, we set* $h = k = 1/16$ *and compute finite element solutions with an initial guess* $z = 1.25$.

In Fig. 5.1, we plot the first four iterations when $A_1 = 1$ *and* $A_2 = 5$, *as well as the true solution for comparison. We see that the iterative method is converging.*

In Fig. 5.2, we plot the true solution and the first four iterations when $A_1 = A_2 = 5$. *We see that the first and third iterations are exactly the same, and the same can be said about the second and fourth iterations.*

Finally, in Fig. 5.3 we plot the true solution and the first four iterations when $A_1 = 5$ *and* $A_2 = 1$. *We see that this iterative method is diverging.*

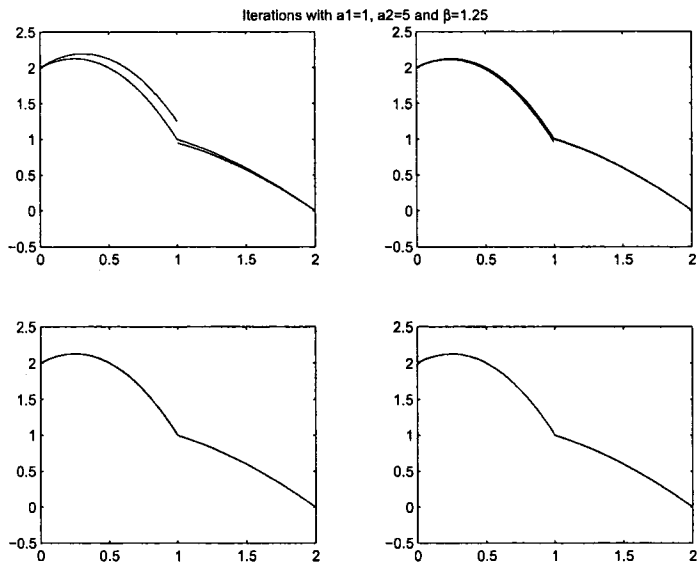

Figure 5.1: Starting in the upper left corner and moving clockwise, we show four iterations of the fixed point problem with $A_1 = 1$ and $A_2 = 5$

5.3.2 A Relaxation method

Showing that the iterative method converges in certain situation is interesting mathematically, but it does not indicate how to proceed if the problem starts to diverge. As an alternative, we define the following relaxed interface operator decomposition method.

Relaxed Interface Operator Decomposition Method

k = 0

Guess $z^{\{0\}}$

while $(\|z^{\{k\}} - z^{\{k-1\}}\| > TOL)$ **do**

(a) k = k+1

(b) Solve (5.3.2) for $u_1^{\{k\}}$.

(c) Solve (5.3.3) for $u_2^{\{k\}}$.

(d) Set $z^{\{k\}} = \alpha z^{\{k\}} + (1-\alpha)u_2^{\{k\}}(1)$,, where $\alpha \in [0,1]$.

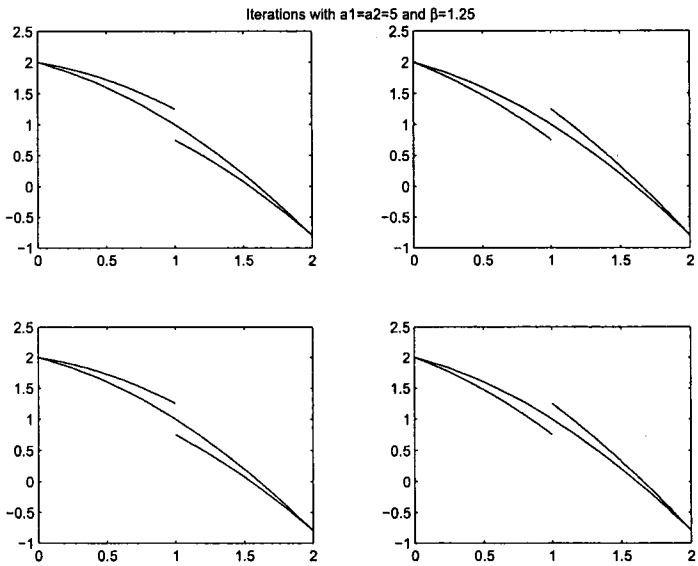

Figure 5.2: Starting in the upper left corner and moving clockwise, we show four iterations of the fixed point problem with $A_1 = 5$ and $A_2 = 5$

end while

Example 5.3.2. *We are interested in the effects of varying the relaxation parameter, α, on the number of iterations for iterative operator decomposition scheme to converge. To investigate this, we set $f_1 = f_2 = 1$ and choose homogeneous Dirichlet boundary conditions away from the interface.*

In Fig. 5.4 we plot the number of iterations required to drive $\|z^{\{k\}} - z^{\{k+1\}}\| < 1 \times 10^{-6}$ for a range of parameter values when $A_1 = 1$ and $A_2 = 5$. This is the convergent case from Proposition 5.3.1. We see that the iterative method converged quickly for small α, with the fastest convergence occuring when $\alpha = 0.15$ and $\alpha = 0.2$.

In Fig. 5.5 we plot the number of iterations required to drive $\|z^{\{k\}} - z^{\{k+1\}}\| < 1 \times 10^{-6}$ for a range of parameter values when $A_1 = 5$ and $A_2 = 5$. This is the 2-cycle case from Proposition 5.3.1. We see that the iterative

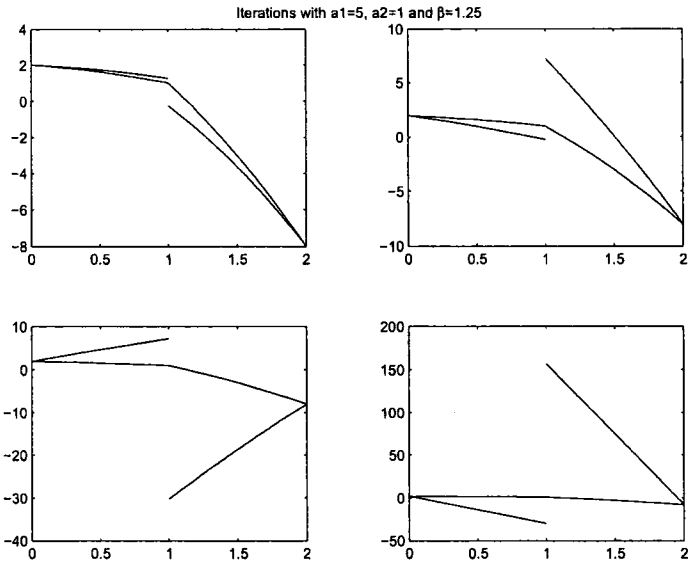

Figure 5.3: Starting in the upper left corner and moving clockwise, we show four iterations of the fixed point problem with $A_1 = 5$ and $A_2 = 1$

method converged fastest for $\alpha = 0.5$.

In Fig. 5.6 we plot the number of iterations required to drive $\|z^{\{k\}} - z^{\{k+1\}}\| < 1 \times 10^{-6}$ *for a range of parameter values when* $A_1 = 5$ *and* $A_2 = 1$. *This is the divergent case from Proposition 5.3.1. We see that the iterative method only converges when the parameter is between* 0.7 *and* 0.95, *with the fastest convergence occuring when* $\alpha = 0.85$.

5.3.3 A Newton Method

In this section we solve the model problem (5.3.1) using the following iterative algorithm.

Newton Interface Operator Decomposition Method

k = 0

Guess $z^{\{0\}}$

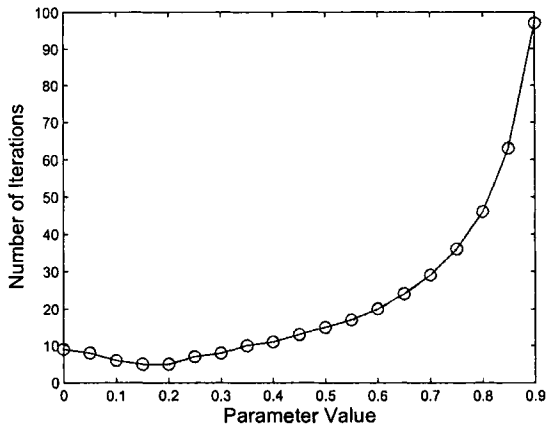

Figure 5.4: Plot showing the number of iterations for a variety of parameter values in Example 5.3.2 when $A_1 = 1$ and $A_2 = 5$. The method did not converge within 100 iterations for parameter values outside the window.

while $(\|z^{\{k\}} - z^{\{k-1\}}\| > TOL)$ **do**

(a) k = k+1

(b) Solve (5.3.2) for $u_1^{\{k\}}$.

(c) Solve (5.3.3) for $u_2^{\{k\}}$.

(d) Use a Newton method to determine $z^{\{k\}}$.

end while

The goal is to write a Newton method to find a root of the function

$$F(z) = z - u_2(1; z),$$

where we note that the solution, u_2, depends on the initial guess z. Newton's method amounts to solving the equation

$$F'(z^{\{k\}})\delta^{\{k\}} = -F(z^{\{k\}}),$$

for $\delta^{\{k\}} = z^{\{k+1\}} - z^{\{k\}}$. Here we treat z like a variable, but later we treat it as a function. For this reason we compute the Gateaux derivative of $F(z)$

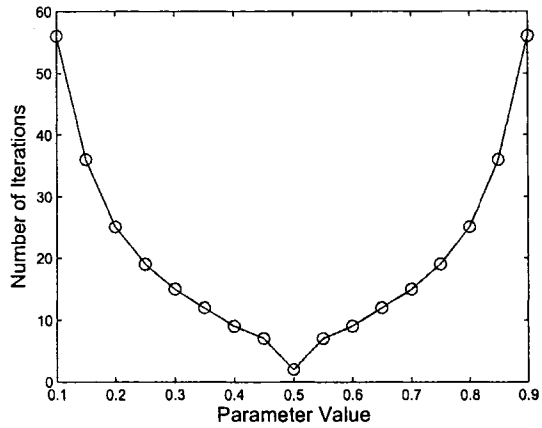

Figure 5.5: Plot showing the number of iterations for a variety of parameter values in Example 5.3.2 when $A_1 = 5$ and $A_2 = 5$. The method did not converge within 100 iterations for parameter values outside the window.

in the "direction" δ.

$$\begin{aligned} F'(z)\delta &= \lim_{\epsilon \to 0} \frac{F(z+\epsilon\delta) - F(z)}{\epsilon} \\ &= \lim_{\epsilon \to 0} \frac{(z+\epsilon\delta - u_2(1, z+\epsilon\delta)) - (z - u_2(1,z))}{\epsilon} \\ &= \delta - \frac{\partial u_2(1,z)}{\partial z}\delta. \end{aligned}$$

For the current problem, $F(z)$ is a function of one variable, so we can write

$$\left(1 - \frac{\partial u_2}{\partial z^{\{k\}}}|_{x=1}\right)\delta^{\{k\}} = -z^{\{k\}} + u_2(1; z^{\{k\}}). \tag{5.3.4}$$

Note that $z^{\{k\}}$ is given and $u_2(1; z^{\{k\}})$ is found by solving (5.3.2) and (5.3.3). However, $\frac{\partial u_2}{\partial z^{\{k\}}}|_{x=1}$ is still an unknown quantity.

To determine the dependence of $u_2(1; z)$ on z, we take the Gateaux derivative of (5.3.2) and (5.3.3) with respect to the input variable z in the "direction" of δ. This gives

$$\begin{cases} -\frac{\partial}{\partial x}\left(A_1 \frac{\partial v_1}{\partial x}\right)\delta = 0, & x \in \Omega_1 \\ v_1\delta = 0, & x = 0, \\ v_1\delta = \delta, & x = 1, \end{cases} \tag{5.3.5}$$

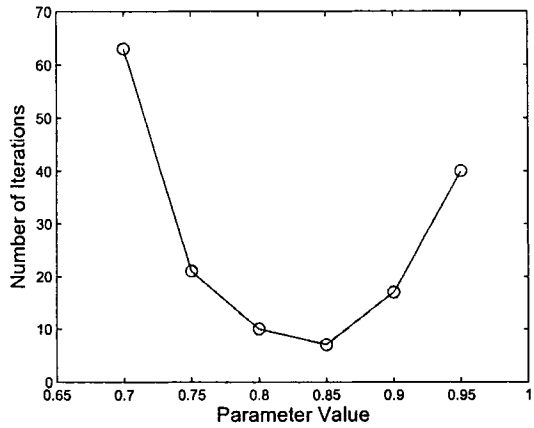

Figure 5.6: Plot showing the number of iterations for a variety of parameter values in Example 5.3.2 when $A_1 = 5$ and $A_2 = 1$. The method did not converge within 100 iterations for parameter values outside the window.

$$\begin{cases} -\frac{\partial}{\partial x}\left(A_2 \frac{\partial v_2}{\partial x}\right)\delta = 0, & x \in \Omega_2 \\ A_2 \frac{\partial v_2}{\partial x}\delta = A_1 \frac{\partial v_1}{\partial x}\delta, & x = 1, \\ v_2 \delta = 0, & x = 2 \end{cases} \tag{5.3.6}$$

where

$$v_1(x) = \frac{\partial u_1}{\partial z}, \quad v_2(x) = \frac{\partial u_2}{\partial z}.$$

We call this the *first variational system.* (In this case we note that $\delta = 1$ since z is a variable in $\mathbb{R}$, not a function or a vector. This is analogous to the observation that the Gateaux derivative is the same as the standard derivative in this case.) The advantage in solving this additional system is the observation that

$$v_2(1) = \frac{\partial u_2}{\partial z}|_{x=1},$$

which provides the Jacobian required for Newton's method. Note that there is no reason to solve the first variational system on the same grid, or even with the same method, as the original system. Newton's method converges

even if the Jacobian is not exact, although the convergence will likely be linear rather than quadratic.

Example 5.3.3. *If we assume A_1 and A_2 are constant functions, then we can solve (5.3.5) and (5.3.6) explicitly giving*

$$v_1(x) = x, \quad v_2(x) = \frac{A_1}{A_2}x - 2\frac{A_1}{A_2}.$$

This gives, $\frac{\partial u_2}{\partial z} = v_2(1) = -\frac{A_1}{A_2}$. We substitute into the Newton equation (5.3.4) and obtain

$$z^{\{k+1\}} = \frac{A_1}{A_1 + A_2} z^{\{k\}} + \left(1 - \frac{A_1}{A_1 + A_2}\right) u_2^{\{k\}}.$$

Therefore, the optimal relaxation parameter in this case is $\alpha = \frac{A_1}{A_1+A_2}$ which agrees with the results in the previous section.

Remark 5.3.1. *If A_1 and A_2 are constants, but we change the domains so that $\Omega_1 = (a, b)$ and $\Omega_2 = (b, c)$, where $a < b < c$, then the optimal relaxation parameter is*

$$\alpha = \frac{(c-b)A_1}{(c-b)A_1 + (b-a)A_2}.$$

Thus, the optimal relaxation parameter depends on the geometry of each domain as well as the thermal conductivities.

5.3.4 Interface Transfer in $\mathbb{R}^n$

In this section, we extend the algorithms in the previous sections to higher dimensional problems. In each of the examples to follow, we use the same model problem. Define $\Omega_1 = (0,1) \times (0,1)$ and $\Omega_2 = (1,2) \times (0,1)$,

and $\Gamma = \partial\Omega_1 \cap \partial\Omega_2$. Consider the model problem,

$$\begin{cases} -\nabla \cdot (A_1 \nabla u_1) = f_1, & x \in \Omega_1, \\ u_1 = 0, & x \in \partial\Omega_1 \backslash \Gamma, \\ \begin{cases} u_1 = u_2, \\ A_1 \partial_n u_1 = A_2 \partial_n u_2, \end{cases} & x \in \Gamma, \\ -\nabla \cdot (A_2 \nabla u_2) = f_2, & x \in \Omega_2, \\ u_2 = 0, & x \in \partial\Omega_2 \backslash \Gamma \end{cases} \tag{5.3.7}$$

with $A_1 = 5$, $A_2 = 1$, and data chosen such that the true solutions are $u_1 = \sin(2\pi x)\sin(2\pi y)$, and $u_2 = 5\sin(2\pi x)\sin(2\pi y)$. We choose uniform triangulations $T_{1,h}$ of Ω_1 and $T_{2,h}$ of Ω_2 such that each contains 800 elements and the triangulations align along the interface.

Example 5.3.4. *We attempt to solve (5.3.7) using the* interface operator decomposition method *defined in §5.3.1. After 10 iterations,* $\|z^{\{k\}} - z^{\{k-1\}}\| = 6.1758 \times 10^5$. *The iterative method is clearly diverging.*

Example 5.3.5. *Next, we solve (5.3.7) using the* relaxed interface operator decomposition method *defined in §5.3.2. In Fig. 5.7, we plot the number of iterations to drive* $\|z^{\{k\}} - z^{\{k-1\}}\| < 1 \times 10^{-6}$ *for a range of parameter values. We see that the method converges only for a narrow range of parameter values with the fastest convergence occuring for* $\alpha = 0.25$.

Next, we want to solve (5.3.7) using the *Newton interface operator decomposition method* defined in §5.3.3. While the algorithm is basically the same, the Dirichlet condition, $z^{\{k\}}$, is now a function rather than a number. The Newton update is found by solving

$$\left(I - \frac{\partial u_2}{\partial z}\right)\delta = -z + u_2,$$

for δ.

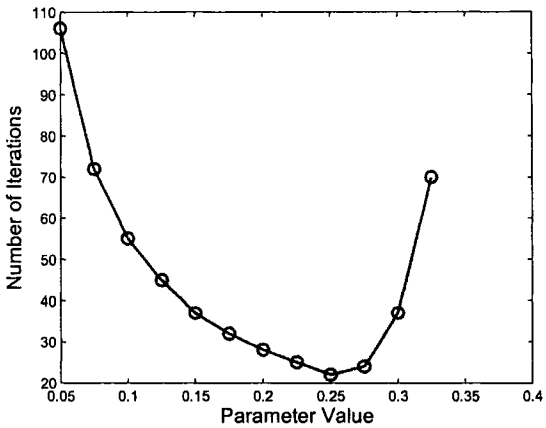

Figure 5.7: Plot showing the number of iterations for a variety of parameter values in Example 5.3.5 when $A_1 = 5$ and $A_2 = 1$. The method did not converge within 125 iterations for parameter values outside the window.

Formally, $v_2 = \frac{\partial u_2}{\partial z}$ is an operator defined using the Gateaux derivative. Since the equations are discretized, v_2 represents a matrix. If there are N degrees of freedom on the interface, then we need to solve N variational systems to construct the Jacobian. The j^{th} such system may be written as

$$\begin{cases} -\nabla \cdot \left(A_1 v_1^j\right) = 0, & x \in \Omega_1, \\ v_1^j = 0, & x \in \partial\Omega_1 \backslash \Gamma, \\ v_1^j = e^j, & x \in \Gamma, \\ -\nabla \cdot \left(A_2 v_2^j\right) = 0, & x \in \Omega_2, \\ v_2^j = 0, & x \in \partial\Omega_2 \backslash \Gamma, \\ A_2 \partial_n v_2^j = A_1 \partial_n v_1^j, & x \in \Gamma, \end{cases} \tag{5.3.8}$$

where e^j denotes the basis function associated with the j^{th} node on Γ. We construct the Jacobian by inserting v_2^j into the j^{th} row.

Example 5.3.6. *We solve (5.3.7) using the* Newton interface operator decomposition method *defined in §5.3.3.*

We plot a typical solution of the variational system (5.3.8) in Fig. 5.8. The iterative method converges to within 1.71×10^{-14} *after 3 iterations.*

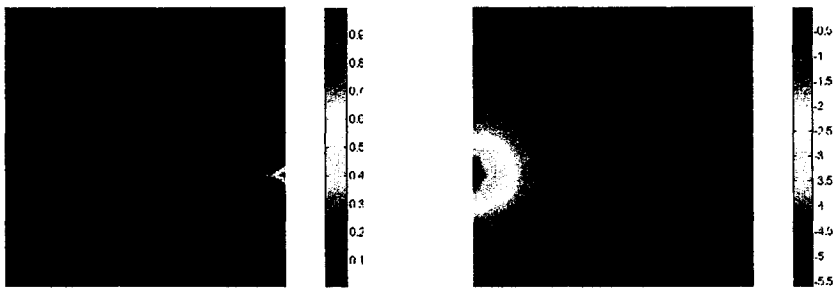

Figure 5.8: A typical solution to the variational problem (5.3.8). On the left, we plot v_1. On the right, we plot v_2.

This is the expected convergence rate for Newton's method since the first iteration uses the initial value.

Remark 5.3.2. *As we can see in Fig. 5.8, the solutions to the variational system (5.3.8) tend to decay rapidly away from the boundary. This indicates that a good approximation of the Jacobian may be obtained by solving a local problem on a patch of elements near the boundary. This would be much cheaper than solving each of the variational systems globally.*

5.4 Global Coupling

Consider the following system of equations:

$$\begin{cases} L_1(u_1, u_2) = f_1, & x \in \Omega, \\ u_1 = 0, & x \in \partial\Omega, \\ L_2(u_1, u_2) = f_2, & x \in \Omega, \\ u_2 = 0, & x \in \partial\Omega, \end{cases} \tag{5.4.1}$$

where L_1 and L_2 are linear differential operators on a convex domain $\Omega \subset \mathbb{R}^n$ with a Lipschitz continuous boundary $\partial\Omega$.

Rather than solving this system all at once, we construct an iterative operator decomposition method to solve each equation separately.

Global Operator Decomposition Method

k = 0

Guess $z^{\{0\}}$

while $(\|z^{\{k\}} - z^{\{k-1\}}\| > TOL)$ **do**

(a) k = k+1

(b) Solve

$$\begin{cases} L_1(u_1^{\{k\}}, z^{\{k\}}) = f_1, & x \in \Omega, \\ u_1^{\{k\}} = 0, & x \in \partial\Omega, \end{cases} \tag{5.4.2}$$

for $u_1^{\{k\}}$.

(c) Solve

$$\begin{cases} L_2(u_1^{\{k\}}, u_2^{\{k\}}) = f_2, & x \in \Omega, \\ u_2^{\{k\}} = 0, & x \in \partial\Omega, \end{cases} \tag{5.4.3}$$

for $u_2^{\{k\}}$.

(d) Set $z^{\{k\}} = u_2^{\{k\}}$,.

end while

Since L_1 and L_2 are linear operators, this is similar to the *block Gauss-Seidel method* described earlier. Therefore, we expect that this iterative method will diverge under certain conditions.

Example 5.4.1. *Define* $\Omega = (0,1) \times (0,1)$, *and consider the system*

$$\begin{cases} -\Delta u_1 + 25u_2 = f_1, & x \in \Omega, \\ u_1 = 0, & x \in \partial\Omega, \\ -\Delta u_2 - 25u_1 = f_2, & x \in \Omega, \\ u_2 = 0, & x \in \partial\Omega, \end{cases} \tag{5.4.4}$$

with data chosen such that the true solutions are $u_1 = \sin(2\pi x)\sin(2\pi y)$, *and* $u_2 = \sin(\pi x)\sin(\pi y)$. *To discretize, we use an unstructured triangulation of* Ω *with* 4951 *elements and* 2577 *degrees of freedom for each variable.*

We begin by making an initial guess $z(x) = 0$ *for* u_2 *and apply the* iterative global operator decomposition method. *A plot of the error for each iteration is shown in Fig. 5.9. It is clear that this iteration is diverging.*

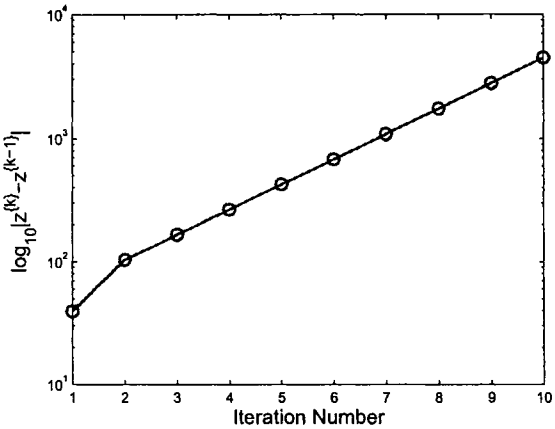

Figure 5.9: Plot of the absolute value of the error in Example 5.4.1 for ten iterations.

In an attempt to improve the convergence behavior, we define the following Newton method.

Newton Global Operator Decomposition Method

k = 0

Guess $z^{\{0\}}$

while $(\|z^{\{k\}} - z^{\{k-1\}}\| > TOL)$ **do**

(a) k = k+1

(b) Solve (5.4.2) for $u_1^{\{k\}}$.

(c) Solve (5.4.3) for $u_2^{\{k\}}$.

(d) Use a Newton method to determine $z^{\{k\}}$.

end while

Consider the functional

$$F(z) = z - u_2(x; z). \tag{5.4.5}$$

The goal of the iterative method is to find a root of $F(z)$. Newton's method for this problem is given by

$$F'(z)\delta = -F(z). \tag{5.4.6}$$

The operator giving the rate of change in u_2 with respect to z in a particular direction is the solution to the variational system differential equation. We determine this by taking the Gateaux derivative of the iterative system with respect to z as in previous sections.

In practice, we use a finite dimensional version of the variational system. Let $U_1, U_2 \in V_h$ be the finite element approximations of u_1 and u_2, where V_h is a finite dimensional subspace of $H_0^1(\Omega)$. For simplicity, we assume that u_1 and u_2 are approximated on the same mesh.

Define e^j to be the basis function associated with the j^{th} node. The j^{th} row of the Jacobian requires v_2^j which satisfy

$$\begin{cases} L_1(v_1^j, e^j) = 0, & x \in \Omega, \\ v_1^j = 0, & x \in \partial\Omega, \\ L_2(v_1^j, v_2^j) = 0, & x \in \Omega, \\ v_2 = 0, & x \in \partial\Omega. \end{cases} \tag{5.4.7}$$

Example 5.4.2. *The variational system for (5.4.4) corresponding to the j^{th} basis function is*

$$\begin{cases} -\Delta v_1^j + 25e^j = 0, & x \in \Omega, \\ v_1^j = 0, & x \in \partial\Omega, \\ -\Delta v_2^j - 25v_1^j = 0, & x \in \Omega, \\ v_2^j = 0, & x \in \partial\Omega, \end{cases} \tag{5.4.8}$$

To construct the exact Jacobian, we solve a variational system on the same mesh corresponding to each degree of freedom for u_2. In Example 5.4.1, this requires the solution of 2577 systems, each having 5154 degrees of freedom.

A well known feature of Newton's method is that a sufficiently accurate approximate Jacobian may lead to rapid convergence, although not at the optimal rate. To construct an approximate Jacobian, we design the following algorithm.

Jacobian Approximation Method

k = 0

Construct a coarse triangulation of Ω.

while k <= Degrees of freedom **do**

(a) k = k+1

(b) Project e^k onto the coarse mesh.

(c) Solve the variatioanl system on the coarse mesh for v_1^k and v_2^k.

(c) Project v_2^k onto the fine mesh.

(d) Update the k^{th} row of the Jacobian matrix.

end while

Example 5.4.3. *Consider the model problem (5.4.4). First, we solve the j^{th} variational system defined by (5.4.8) on the same mesh. This involves solving* 2577 *variational systems, each having* 5154 *degrees of freedom, to construct the Jacobian. In Fig. 5.10, we plot typical solutions to the variational system.*

Next, we approximate the Jacobian by solving the variational systems on a coarser mesh. We solve 2577 *systems, but we use uniform meshes*

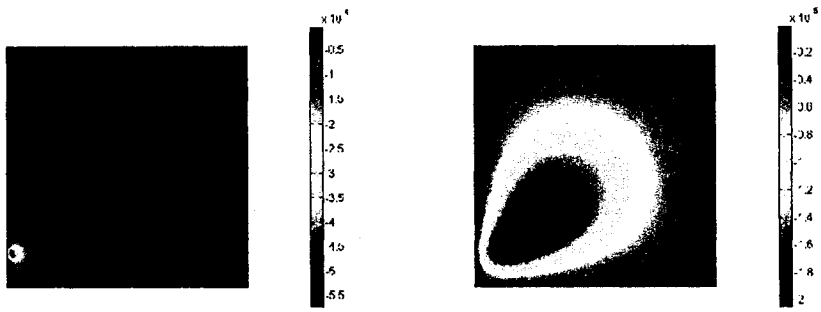

Figure 5.10: A typical solution to the variational system (5.4.8). On the left, we plot v_1. On the right, we plot v_2.

with $h = 0.05$, $h = 0.1$, *and* $h = 0.25$ *with* 882, 242, *and* 50 *degrees of freedom repectively. In Fig. 5.11, we show the original mesh and the mesh corresponding to* $h = 0.25$. *We use an* L^2 *projection to map the basis*

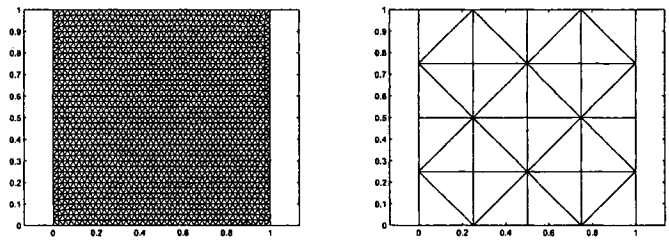

Figure 5.11: On the left, we plot the original mesh. On the right, we plot the coarse mesh used to approximate the Jacobian.

function to the coarse mesh, and to map the variational solutions to the fine mesh.

In Fig. 5.12, we plot the convergence of the iterative method using the full Jacobian, and using the Jacobian approximation method. *We see that using the true Jacobian yields high accuracy and converges in three iterations since the problem is linear. Using the approximate Jacobians yields convergence, although more iterations are required.*

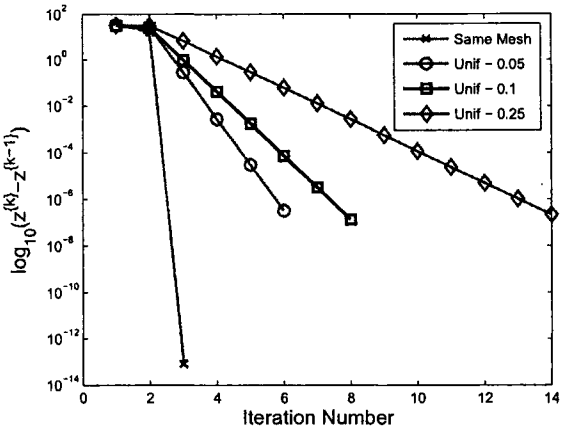

Figure 5.12: Plot of $\|z^{\{k\}} - z^{\{k-1\}}\|$ using the exact Jacobian and using projected Jacobians in Example 5.4.3.

On the other hand, we are usually more concerned with solution time than the number of iterations. In Fig. 5.13 and Table 5.1, we break down the solution times in each case. We see that using the exact Jacobian is

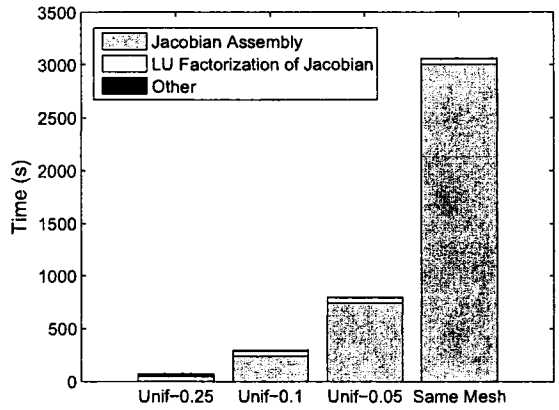

Figure 5.13: Plot of the solution times in Example 5.4.3 using the exact Jacobian and using projected Jacobians.

impractical since the construction of this Jacobian is so expensive. Using a coarse mesh to solve the variational problem and approximate the Jacobian is much faster despite requiring more iterations for convergence. We

Var. Mesh	Iterations	Jac. Time (s)	LU Time (s)	Total Time (s)
Original	3	3005.0	50.5	3060.0
Unif. - 0.05	6	738.0	50.5	769.9
Unif. - 0.1	8	236.3	50.4	298.3
Unif. - 0.25	14	48.0	6.8	71.2

Table 5.1: Number of iterations and break down of solution times in Example 5.4.3.

include the time required to compute the LU decomposition of the Jacobian to demonstrate that this is an expensive process due to the fact that the Jacobian is nearly full. For this problem, we only needed to compute the Jacobian and the LU decomposition once since the problem is linear.

Chapter 6

INTERFACE TRANSFER FOR ELLIPTIC EQUATIONS

6.1 Introduction

In this chapter, we consider the solution of a conjugate heat transfer problem by an operator decomposition approach. The goal is to compute a functional of the temperature of a body composed of two distinct components that share a common boundary or interface. The two components may have different conductivities and be subject to different heat sources and boundary conditions. The model consists of a system of boundary value problems for two elliptic equations coupled through boundary conditions posed on the common interface. Our interest is in situations where there is a small number of interfaces between the components.

The operator decomposition method we consider is closely related to the non-overlapping domain decomposition methods as described in, for example, [52, 58, 54, 59]. The major focus of these papers, however, is the convergence of different iterative methods and the design of effective pre-conditioners or relaxation parameters, while the accuracy of the final finite element approximation is usually not addressed, despite the fact that some of the numerical results indicate a loss of accuracy. To our knowledge, the analysis below is the first work to address the loss of accuracy. Our results would extend to a particular type of non-overlapping domain decomposition.

In §6.2, we introduce the conjugate heat transfer problem and provide some notation. We describe the iterative operator decomposition finite element method and some modifications in §6.3, as well as the boundary flux method used to compute gradients on the common interface. We perform two *a posteriori* error analyses in §6.4, using first the adjoint to the fully coupled problem and then the adjoint to the iterative scheme, and present

numerical results to highlight the differences between the two. We end the section with a brief discussion of adaptive mesh refinement. In §6.5, we carry out an analysis which identifies the transferred gradient information as being responsible for the loss of order and show that using the boundary flux method to compute the gradient information restores the order of convergence.

6.2 A model for conjugate heat transfer

Let Ω_1 and Ω_2 be convex polygonal domains in $\mathbb{R}^2$ or $\mathbb{R}^3$ with boundaries $\partial\Omega_1$ and $\partial\Omega_2$ intersecting along an interface $\Gamma = \partial\Omega_1 \cap \partial\Omega_2$. We consider a system of second order linear elliptic problems, where the components are coupled through boundary conditions imposed on Γ,

$$\begin{cases} L_1 u_1 = f_1, & \mathbf{x} \in \Omega_1, \\ u_1 = 0, & \mathbf{x} \in \partial\Omega_1 \backslash \Gamma, \\ \begin{cases} u_1 = u_2, \\ A_1 \partial_n u_1 = A_2 \partial_n u_2, \end{cases} & \mathbf{x} \in \Gamma, \\ L_2 u_2 = f_2, & \mathbf{x} \in \Omega_2, \\ u_2 = 0, & \mathbf{x} \in \partial\Omega_2 \backslash \Gamma, \end{cases} \tag{6.2.1}$$

where for $i = 1, 2$, $L_i u_i = -\nabla \cdot (A_i \nabla u_i) + c_i u_i$, $A_i \geq A_{i,0} > 0, c_i, f_i$ are sufficiently smooth functions and ∂_n denotes the unit outward normal derivative to $\partial\Omega_1$. The results of this chapter extend easily to general elliptic operators and general Dirichlet, Neumann, and Robin boundary conditions on the boundaries in the complement of the interface.

We let $L^2(\Omega_i)$ denote the space of square integrable functions on Ω_i with inner product $(\cdot,\cdot)_{\Omega_i}$ and norm $\|\cdot\|_{\Omega_i}$, but use $(\cdot,\cdot) = (\cdot,\cdot)_{\Omega_i}$ when the domain is clear. We use $H^s(\Omega_i)$ to denote the Sobolev space with real index s associated with the norm $\|\cdot\|_{\Omega_i,s}$ and seminorm $|\cdot|_{\Omega_i,s}$. We also use the subspaces $H_0^1(\Omega_i) = \{v \in H^1(\Omega_i), v = 0 \text{ on } \partial\Omega_i \backslash \Gamma\}$.

The weak formulation of (6.2.1) seeks $u_i \in H_0^1(\Omega_i)$ such that $u_1 = u_2$ on Γ and

$$\sum_{i=1}^{2} a_i(u_i, v_i) = \sum_{i=1}^{2} (f_i, v_i), \tag{6.2.2}$$

for all $v_i \in H_0^1(\Omega_i)$ with $a_i(u_i, v) = \int_{\Omega_i} (A_i \nabla u_i \cdot \nabla v + c_i u_i v)\, dx$ for $i = 1, 2$. Assuming that each $a_i(\cdot, \cdot)$ is coercive, (6.2.2) admits a unique weak solution in $H_0^1(\Omega_i)$ [4, 14, 24].

6.3 An iterative operator decomposition method

To compute a numerical solution of (6.2.1), we consider several iterative operator decomposition methods. Let $u_2^{(0)}$ be an initial guess for the Dirichlet data along the interface. We solve

$$\begin{cases} L_1 u_1^{(k)} = f_1, & \mathbf{x} \in \Omega_1, \\ u_1^{(k)} = 0, & \mathbf{x} \in \partial\Omega_1 \backslash \Gamma, \\ u_1^{(k)} = u_2^{(k-1)}, & \mathbf{x} \in \Gamma, \end{cases} \tag{6.3.1}$$

followed by

$$\begin{cases} L_2 u_2^{(k)} = f_2, & \mathbf{x} \in \Omega_2, \\ u_2^{(k)} = 0, & \mathbf{x} \in \partial\Omega_2 \backslash \Gamma, \\ A_1 \partial_n u_1^{(k)} = A_2 \partial_n u_2^{(k)}, & \mathbf{x} \in \Gamma. \end{cases} \tag{6.3.2}$$

We iterate until (hopefully) a convergence criteria is met, e.g., $\|u_2^{(k)} - u_2^{(k-1)}\|_\Gamma$ is smaller than a prescribed tolerance.

6.3.1 Finite element discretization

We let $T_{i,h}$ be a triangulation of Ω_i into elements K where the length of the longest edge is h_K and $h = \max_{K \in T_{i,h}} h_K$. We assume that each triangulation is locally quasi-uniform and $\overline{\Omega}_i = \cup_{K \in T_{i,h}} K$. However, the triangulations on either side of Γ are not assumed to be aligned.

We use the piecewise polynomial spaces

$$S_1 = \left\{ v \text{ continuous on } \Omega_1,\ v \in P^1(K) \text{ for all } K \in T_{1,h} \right\},$$

$$S_2 = \left\{ v \text{ continuous on } \Omega_2,\ v \in P^1(K) \text{ for all } K \in T_{2,h} \right\},$$

and the associated spaces

$$S_{1,0} = \{ v \in S_1 \mid v = 0, \mathbf{x} \in \partial\Omega_1 \},$$

$$S_{2,0} = \{ v \in S_2 \mid v = 0, \mathbf{x} \in \partial\Omega_2 \backslash \Gamma \},$$

where $P^1(K)$ denotes the space of linear polynomials on an element K. We let π_i be a projection into S_i as well as the projection into S_i along the interface Γ.

For the finite element approximation, we compute $U_1^{(k)} \in S_1$ satisfying

$$\begin{cases} a_1(U_1^{(k)}, v) = (f_1, v)_{\Omega_1}, & \text{for all } v \in S_{1,0}, \\ U_1^{(k)} = 0, & \mathbf{x} \in \partial\Omega_1 \backslash \Gamma, \\ U_1^{(k)} = \pi_1 U_2^{(k-1)}, & \mathbf{x} \in \Gamma, \end{cases} \tag{6.3.3}$$

followed by $U_2^{(k)} \in S_2$ such that

$$\begin{cases} a_2(U_2^{(k)}, v) = (f_2, v)_{\Omega_2} - (A_1 \partial_n U_1^{(k)}, v)_\Gamma, & \text{for all } v \in S_{2,0}, \\ U_2^{(k)} = 0, & \mathbf{x} \in \partial\Omega_2 \backslash \Gamma. \end{cases} \tag{6.3.4}$$

6.3.2 Relaxed iterations

Unfortunately, the simple iterative (6.3.3)-(6.3.4) may not converge. In particular, the convergence depends on the values of A_1 and A_2 along the interface and the geometry of each region [37, 52, 58]. As an alternative, we consider two "relaxed" iteration schemes.

(1) *Relaxed Dirichlet values*

We choose $\alpha \in [0, 1]$ and update the Dirichlet interface values with

$$U_1^{(k)} = \alpha U_1^{(k-1)} + (1 - \alpha)\pi_1 U_2^{(k-1)}. \tag{6.3.5}$$

Optimal values of α can be found in [52, 58], but $\alpha = 1/2$ works well in most situations.

(2) *Relaxed Neumann values*

We chose $\beta \in [0,1]$, set $N_\beta = \beta A_2 \partial_n U_2^{(k-1)} + (1-\beta) A_1 \partial_n U_1^{(k)}$, and seek $U_2^{(k)} \in S_2$ such that

$$a_2(U_2^{(k)}, v) = (f_2, v)_{\Omega_2} - (N_\beta, v)_\Gamma, \text{ for all } v \in S_{2,0}, \tag{6.3.6}$$

A proper choice of β reduces the number of iterations.

6.3.3 Flux correction

It turns out that the operator decomposition results in a loss of order arising from the relatively low order of accuracy in the derivative of the finite element solution, namely, $\mathbf{O}(h)$ when computed with a standard $\mathbf{O}(h^2)$ method. We show below that this reduces the order of the overall approximation to first order. To mitigate this effect, we use a post-processing technique developed by Wheeler [55] and Carey [34, 20] to compute more accurate boundary flux information. See Chapter 4.4 for more details.

We define the set of elements in $T_{1,h}$ that intersect the boundary by

$$T_{1,h}^\Gamma = \{K \in T_{1,h} \mid \overline{K} \cap \partial\Omega \neq \emptyset\},$$

and the corresponding space

$$W_h = \{v \in P^1(K) \text{ with } K \in T_{1,h}^\Gamma,\ v(\eta_i) = 0 \text{ if } \eta_i \notin \partial\Omega\},$$

where $\{\eta_i\}$ denotes the nodes of element K, so the degrees of freedom correspond to the nodes on the boundary. We seek $\sigma^{(k)} \in W_h$ satisfying

$$-(\sigma^{(k)}, v)_\Gamma = (f_1, v)_{\Omega_1} - a_1(U_1^{(k)}, v), \quad \text{for all } v \in W_h, \tag{6.3.7}$$

where $U_1^{(k)}$ solves (6.3.3). Green's identity implies that $\sigma^{(k)}$ gives an approximation to the normal flux on the boundary that is relatively inexpensive to compute.

In general, the accuracy of the flux approximation depends on the regularity of an associated Green's function [35, 56]. In many cases this post-processed flux is $\mathbf{O}(h^2)$ rather than the typical $\mathbf{O}(h)$ for the normal flux of a piecewise linear finite element approximation. However, it turns out that a fortunate cancellation of errors makes the accuracy of this recovered flux unimportant for our purposes.

We stress that $\sigma^{(k)}$ is not assumed to be continuous. In fact, if the domain has a corner on the boundary, the normal derivative is, in general, discontinuous due to the jump in the normal vector. When Dirichlet conditions are given on each boundary segment we implement the method described in [20] and allow two degrees of freedom to account for the discontinuity. For the application to more general boundary conditions, such as a Neumann or Robin condition on one segment, or an interface condition on both, we refer the reader to §4.4.

Now, as an alternative to (6.3.4), we may solve

$$\begin{cases} a_2(U_2^{(k)}, v) = (f_2, v)_{\Omega_2} - (\sigma^{(k)}, v)_\Gamma, & \text{for all } v \in S_{2,0}, \\ U_2^{(k)} = 0, & \mathbf{x} \in \partial\Omega_2 \backslash \Gamma. \end{cases} \tag{6.3.8}$$

Other potential approaches to mitigating the loss of order include approximating the boundary flux using a gradient recovery techniques such as the Zienkiewicz-Zhu (ZZ) patch recovery technique [59] or the polynomial preserving recovery (PPR) method [49], or using higher order polynomials near the interface to improve the accuracy of the finite element flux [9, 42]. We had mixed computational success using these other approaches.

6.4 *A posteriori* error analysis

To estimate the error of the operator decomposition finite element approximation, we apply *a posteriori* techniques based on variational analysis and the adjoint problem. In this case, however, the adjoint for the fully coupled original problem differs significantly from the adjoint problem associated with the discretization of the decomposed system. The motivation to use operator decomposition suggests that the adjoint of the full problem is unavailable in practice and therefore the decomposed adjoint is used to compute error estimates. The difference between the adjoint problems can be decreased by increasing the computational work used to solve the decomposed adjoint, however in practice, we must consider the issue that the two adjoints lead to significantly different error representations.

In order to understand the differences, we derive *a posteriori* error estimates using both adjoints. Both analyses begin in the same way. We wish to estimate the difference between the exact solution to the full problem and the numerical approximation to the solution of the iterative procedure, i.e. $E^{(k)} = u - U^{(k)}$. We decompose the error,

$$E^{(k)} = u - U^{(k)} = \left(u - u^{(k)}\right) + \left(u^{(k)} - U^{(k)}\right) = c^{(k)} + e^{(k)}.$$

The first component $c^{(k)}$ measures the difference between the exact solution to the full problem and the exact solution to the iterative problem. If the iterative method converges, then $c^{(k)} \rightarrow 0$. The second component $e^{(k)}$ measures the numerical error in solving the iterative problem.

It is helpful to remark that in the conventional approach, the adjoint is defined in such a way as to make the formal bilinear identity hold [44, 29], i.e., to make the boundary terms arising from integration by parts

equal to zero. In the analysis below, we find it convenient to consider the values passed between components as quantities of interest defined on the common boundary and we have to alter the definition of the adjoint problem accordingly.

6.4.1 The adjoint to the fully coupled problem

The adjoint boundary value problem for the quantity of interest $(\psi, \mathbf{u}) = (\psi_1, u_1) + (\psi_2, u_2)$ for the coupled problem (6.2.1) is

$$\begin{cases} L_1^*\phi_1 = \psi_1, & \mathbf{x} \in \Omega_1, \\ \phi_1 = 0, & \mathbf{x} \in \partial\Omega_1 \backslash \Gamma, \\ \begin{cases} \phi_1 = \phi_2, \\ A_1\partial_n\phi_1 = A_2\partial_n\phi_2, \end{cases} & \mathbf{x} \in \Gamma, \\ L_2^*\phi_2 = \psi_2, & \mathbf{x} \in \Omega_2, \\ \phi_2 = 0, & \mathbf{x} \in \partial\Omega_2 \backslash \Gamma, \end{cases} \tag{6.4.1}$$

where $L_i^*\phi_i = -\nabla \cdot (A_i \nabla \phi_i) + c_i\phi_i$. We use $a_i^*(\cdot, \cdot)$ to represent the weak form of the adjoint operator. We solve (6.4.1) numerically by using an iterative operator decomposition approach as for the forward problem. The iterations are completely independent of the forward iterations.

We can derive an error representation formula for the basic scheme (6.3.1)-(6.3.2), a weighted relaxation technique (6.3.5) or (6.3.6), or when using the post-processed flux as in (6.3.8). In the discussion below, we use $\theta_h^{(k)}$ to denote the numerical flux passed at the k^{th} iteration from Ω_1 to Ω_2.

To begin, we multiply the system (6.4.1) by $(\psi_1, \psi_2)^T$ and apply the divergence theorem, noting that $u_1 = u_2$ and $A_1\partial_n\phi_1 = A_2\partial_n\phi_2$ along Γ, to obtain

$$\begin{aligned} (\psi_1, e_1) + (\psi_2, e_2) = a_1(e_1, \phi_1) + a_2(e_2, \phi_2) \\ + (U_1^{(k)}, A_1\partial_n\phi_1)_\Gamma - (U_2^{(k)}, A_2\partial_n\phi_2)_\Gamma. \end{aligned}$$

Observe that the test space $S_{1,0}$ consists of functions that are zero along the interface, while in general, ϕ_1 is not zero along Γ. This means that the projection of ϕ_1 into $S_{1,0}$ cannot be the interpolant. We define a new projection $\pi_1^0 : H^2 \to S_{1,0}$ such that for any node x_i

$$\pi_1^0\phi(x_i) = \begin{cases} \pi_1\phi(x_i), & x_i \notin \Gamma, \\ 0, & x_i \in \Gamma. \end{cases} \tag{6.4.2}$$

We also observe that

$$a_2(e_2, \pi_2\phi_2) = -(A_1\partial_n u_1, \pi_2\phi_2)_\Gamma + (\theta_h^{(k)}, \pi_2\phi_2)_\Gamma,$$

and

$$a_1(u_1, \phi_1 - \pi_1^0\phi_1) = (f_1, \phi_1 - \pi_1^0\phi_1) + (A_1\partial_n u_1, \phi_1)_\Gamma,$$

$$a_2(u_2, \phi_2 - \pi_2\phi_2) = (f_2, \phi_2 - \pi_2\phi_2) - (A_1\partial_n u_1, \phi_2 - \pi_2\phi_2)_\Gamma,$$

since the adjoint solutions are not zero along Γ.

Using the projection (6.4.2) in the Galerkin orthogonality relation and the equalities above, we have

$$\begin{aligned}(\psi_1, e_1) + (\psi_2, e_2) = {} & (f_1, \phi_1 - \pi_1^0\phi_1) - a_1(U_1^{(k)}, \phi_1 - \pi_1^0\phi_1) \\ & + (f_2, \phi_2 - \pi_2\phi_2) - a_2(U_2^{(k)}, \phi_2 - \pi_2\phi_2) \\ & + (U_1^{(k)} - U_2^{(k)}, A_1\partial_n\phi_1)_\Gamma + (\theta_h^{(k)}, \pi_2\phi_2)_\Gamma.\end{aligned}$$

Next, we define $\pi_\partial\phi_1 = \pi_1\phi_1 - \pi_1^0\phi_1$ which is nonzero only near the interface due to the definition of $\pi_1^0\phi_1$. Substituting $\pi_1^0\phi_1 = \pi_1\phi_1 - \pi_\partial\phi_1$ gives

$$\begin{aligned}(\psi_1, e_1) + (\psi_2, e_2) = {} & (f_1, \phi_1 - \pi_1\phi_1) - a_1(U_1^{(k)}, \phi_1 - \pi_1\phi_1) \\ & + (f_2, \phi_2 - \pi_2\phi_2) - a_2(U_2^{(k)}, \phi_2 - \pi_2\phi_2) \\ & + (f_1, \pi_\partial\phi_1) - a_1(U_1^{(k)}, \pi_\partial\phi_1) \\ & + (U_1^{(k)} - U_2^{(k)}, A_1\partial_n\phi_1)_\Gamma + (\theta_h^{(k)}, \pi_2\phi_2)_\Gamma\end{aligned}$$

Finally (6.3.7) implies that the recovered boundary flux, $\sigma^{(k)}$ satisfies

$$-(\sigma^{(k)}, \pi_\partial \phi_1)_\Gamma = (f_1, \pi_\partial \phi_1)_{\Omega_1} - a_1(U_1^{(k)}, \pi_\partial \phi_1),$$

while $\pi_\partial \phi_1 = \pi_1 \phi_1$ along Γ. We conclude

Theorem 6.4.1. *The errors $e_1 = u_1 - U_1^{(k)}$ and $e_2 = u_2 - U_2^{(k)}$ satisfy*

$$\begin{aligned}(\psi_1, e_1) + (\psi_2, e_2) = {} & (f_1, \phi_1 - \pi_1\phi_1) - a_1(U_1^{(k)}, \phi_1 - \pi_1\phi_1) \\ & + (f_2, \phi_2 - \pi_2\phi_2) - a_2(U_2^{(k)}, \phi_2 - \pi_2\phi_2) \\ & + (U_1^{(k)} - U_2^{(k)}, A_1\partial_n\phi_1)_\Gamma \\ & + (\theta_h^{(k)}, \pi_2\phi_2)_\Gamma - (\sigma^{(k)}, \pi_1\phi_1)_\Gamma \end{aligned} \tag{6.4.3}$$

The error has been decomposed into two discretization components, an iterative component, and a component reflecting the error arising from the transfer of derivative information. The choice of iterative method does not affect the discretization component of the error, but it does influence both the iterative and the transfer components along Γ. We illustrate this for a few common iterative schemes:

- Suppose $U_1^{(k)} = \pi_1 U_2^{(k-1)}$, then

  $$\begin{aligned}&(U_1^{(k)} - U_2^{(k)}, A_1\partial_n\phi_1)_\Gamma \\ &\quad = (\pi_1 U_2^{(k-1)} - U_2^{(k-1)}, A_1\partial_n\phi_1)_\Gamma + (U_2^{(k)} - U_2^{(k-1)}, A_1\partial_n\phi_1)_\Gamma,\end{aligned}$$

  which represents a projection error and an iteration error.

- Suppose $U_1^{(k)} = \alpha U_1^{(k-1)} + (1-\alpha)\pi_1 U_2^{(k-1)}$, then

  $$\begin{aligned}(U_1^{(k)} - U_2^{(k)}, A_1\partial_n\phi_1)_\Gamma = {} & \sum_{i=1}^{k-2} \alpha^{i-1}(\pi_1(U_1^{(k-i-1)} - U_2^{(k-i)}), A_1\partial_n\phi_1)_\Gamma \\ & + (U_2^{(k-1)} - U_2^{(k)}, A_1\partial_n\phi_1)_\Gamma \\ & + (\pi_1 U_2^{(k-1)} - U_2^{(k-1)}, A_1\partial_n\phi_1)_\Gamma,\end{aligned}$$

which represents a series of iteration errors and a projection error. Notice that since $\alpha < 1$, the effect of the iteration error from previous iterations diminishes due to the increasing power on α.

- Suppose we set $\theta_h^{(k)} = A_1 \partial_n U_1^{(k)}$, then

$$(\theta_h^{(k)}, \pi_2\phi_2)_\Gamma - (\sigma^{(k)}, \pi_1\phi_1)_\Gamma = (A_1\partial_n U_1^{(k)} - \sigma^{(k)}, \pi_2\phi_2)_\Gamma + (\sigma^{(k)}, \pi_2\phi_2 - \pi_1\phi_1)_\Gamma,$$

which represents a transfer error and a projection error.

- Suppose we set $\theta_h^{(k)} = \sigma^{(k)}$, then

$$(\theta_h^{(k)}, \pi_2\phi_2)_\Gamma - (\sigma^{(k)}, \pi_1\phi_1)_\Gamma = (\sigma^{(k)}, \pi_2\phi_2 - \pi_1\phi_1)_\Gamma,$$

which represents only a projection error with *no transfer error.*

6.4.2 The adjoint to the iterative scheme

Next, we derive an error representation using the natural adjoint for the iterative system (6.3.1)-(6.3.2), where we set $U_1^{(k)} = \pi_1 U_2^{(k-1)}$ and $A_2\partial_n U_2^{(k)} = A_1\partial_n U_1^{(k)}$. This adjoint avoids the formulation of a globally coupled adjoint, and reads

$$\begin{cases} L_1^*\phi_1^{(k)} = \psi_1^{(k)}, & \mathbf{x} \in \Omega_1, \\ \phi_1^{(k)} = 0, & \mathbf{x} \in \partial\Omega_1 \backslash \Gamma, \\ \phi_1^{(k)} = \phi_2^{(k)}, & \mathbf{x} \in \Gamma, \end{cases} \tag{6.4.4}$$

$$\begin{cases} L_2^*\phi_2^{(k)} = \psi_2^{(k)}, & \mathbf{x} \in \Omega_2, \\ \phi_2^{(k)} = 0, & \mathbf{x} \in \partial\Omega_2 \backslash \Gamma, \\ A_2\partial_n\phi_2^{(k)} = A_1\partial_n\phi_1^{(k+1)}, & \mathbf{x} \in \Gamma, \end{cases} \tag{6.4.5}$$

for $k = N, \ldots, 1$ with $A_1\partial_n\phi_1^{(N+1)} = 0$. Note that the adjoint system is defined "backwards". In the reasonable case in which we seek information from the final iterate, the data for the adjoint problem would be $\psi_1^{(1)} = \psi_2^{(1)} = \ldots = \psi_1^{(N-1)} = \psi_2^{(N-1)} = 0$, with $\psi_1^{(N)}$ and $\psi_2^{(N)}$ chosen appropriately.

We derive the error representation formula for this system by observing

$$\sum_{k=1}^{N}\left((\psi_1^{(k)},e_1^{(k)})+(\psi_2^{(k)},e_2^{(k)})\right)=\sum_{k=1}^{N}\left((L_1^*\phi_1^{(k)},e_1^{(k)})+(L_2^*\phi_2^{(k)},e_2^{(k)})\right),$$

and apply the steps used to derive (6.4.3). This provides the following theorem.

Theorem 6.4.2. *The errors* $e_1^{(1)},e_2^{(1)},\ldots,e_1^{(N)},e_2^{(N)}$ *satisfy*

$$\begin{aligned}
&\sum_{k=1}^{N}\left((\psi_1^{(k)},e_1^{(k)})+(\psi_2^{(k)},e_2^{(k)})\right)\\
&\quad=\sum_{k=1}^{N}\Big((f_1,\phi_1^{(k)}-\pi_1\phi_1^{(k)})-a_1(U_1^{(k)},\phi_1^{(k)}-\pi_1\phi_1^{(k)})\\
&\qquad+(f_2,\phi_2^{(k)}-\pi_2\phi_2^{(k)})-a_2(U_2^{(k)},\phi_2^{(k)}-\pi_2\phi_2^{(k)})\\
&\qquad+(U_1^{(k)}-U_2^{(k-1)},A_1\partial_n\phi_1^{(k)})_\Gamma+(A_1\partial_nU_1^{(k)},\pi_1\phi_1^{(k)})_\Gamma-(\sigma^{(k)},\pi_2\phi_2^{(k)})_\Gamma\Big)
\end{aligned}\tag{6.4.6}$$

It is possible to define adjoints for more complicated iterative methods, but it may be difficult. For example, relaxation techniques couple $U_1^{(k)}$ to both $U_2^{(k-1)}$ and $U_1^{(k-1)}$ which affects how the adjoint variables are coupled. In addition, using the post-processed numerical flux given in section 3.2 requires defining the adjoint of the post-processing procedure.

6.4.3 Numerical results

We illustrate the *a posteriori* estimates in §4.1 and §4.2.

Example 6.4.1. We triangulate the domains $\Omega_1=[-0.25,0.25]\times[-0.25,0.25]$ and $\Omega_2=([-1,1]\times[-1,1])\setminus\Omega_1$ independently, see Fig. 6.1, and consider

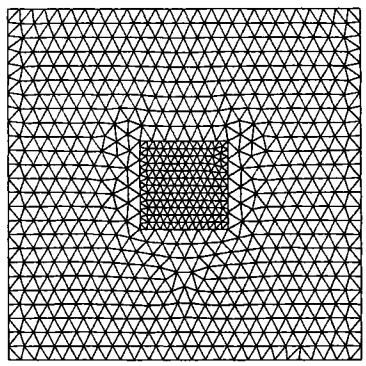

Figure 6.1: Triangulations of Ω_1 and Ω_2 that do not match along the interface.

the elliptic interface problem,

$$\begin{cases} -\nabla \cdot (\nabla u_1) = f_1, & \mathbf{x} \in \Omega_1, \\ \begin{cases} u_1 = u_2, \\ \partial_n u_1 = \partial_n u_2, \end{cases} & \mathbf{x} \in \Gamma, \\ -\nabla \cdot (\nabla u_2) = f_2, & \mathbf{x} \in \Omega_2, \\ u_2 = 0, & \mathbf{x} \in \partial\Omega_2 \backslash \Gamma, \end{cases} \tag{6.4.7}$$

where $f_1(x, y)$ and $f_2(x, y)$ are chosen so the true solutions are $u_2 = \sin(4\pi x)\sin(4\pi y)$ and $u_1 = 3u_2$. We solve this using (6.3.3)-(6.3.4) with $U_1^{(k)} = \pi_1 U_2^{(k-1)}$ and iterate until $\|U_1^{(k)} - U_1^{(k-1)}\|_\Gamma < 10^{-5}$. We consider several quantities of interest

1. the global average value,
2. the average value over Ω_1,
3. the average value over Ω_2,
4. the value of u_2 at the point $(0, 0)$ computed using the approximate delta function $\hat{\delta}_0(x, y) = \frac{400}{\pi}\exp(-400x^2 - 400y^2)$.

First we use an independent iterative scheme to solve the adjoint of the fully coupled problem using the relaxation parameter $\alpha = 1/2$ and iterating until $\|\Phi_1^{(k)} - \Phi_1^{(k-1)}\|_\Gamma < 10^{-6}$, where $\Phi_1^{(k)}$ represents the approximation of the adjoint solution at the k^{th} iteration. We display the error estimates for each of the quantities of interest in Table 6.1.

Next we solve the adjoint to the operator decomposition method and give these results in Table 6.2. Note that the number of iterations, as well as the values of the relaxation parameters, are determined by the number of iterations used in the solution of the forward operator decomposition method.

	ψ_1	ψ_2	True Error	F.C. Adjoint	Effect. Ratio
1.	4	4/15	-0.20	-0.20	0.99
2.	4	0	-0.15	-0.15	0.99
3.	0	4/15	-0.046	-0.046	1.00
4.	$\hat{\delta}_0$	0	-0.15	-0.15	0.99

Table 6.1: Error estimates and effectivity ratios using the adjoint for the fully coupled system.

	ψ_1	ψ_2	True Error	O.D. Adjoint	Effect. Ratio
1.	4	4/15	-0.20	-0.20	0.99
2.	4	0	-0.15	-0.15	0.99
3.	0	4/15	-0.046	-0.046	1.00
4.	$\hat{\delta}_0$	0	-0.15	-0.15	0.99

Table 6.2: Error estimates and effectivity ratios using the adjoint for the operator decomposition method.

Example 6.4.2. Next we consider neighboring domains $\Omega_1 = [0,1] \times [0,1]$ and $\Omega_2 = ([1,2] \times [0,1])$ with independent triangulations, and the elliptic

interface problem

$$\begin{cases} -\nabla \cdot (A_1 \nabla u_1) = f_1, & \mathbf{x} \in \Omega_1, \\ u_1 = 0, & \mathbf{x} \in \partial\Omega_1 \backslash \Gamma, \\ \begin{cases} u_1 = u_2, \\ A_1 \partial_n u_1 = A_2 \partial_n u_2, \end{cases} & \mathbf{x} \in \Gamma, \\ -\nabla \cdot (A_2 \nabla u_2) = f_2, & \mathbf{x} \in \Omega_2, \\ u_2 = 0, & \mathbf{x} \in \partial\Omega_2 \backslash \Gamma, \end{cases} \tag{6.4.8}$$

where $A_1 = 1$, $A_2 = 3$, and $f_1(x, y)$ and $f_2(x, y)$ are chosen so the true solutions are $u_2 = \sin(2\pi x)\sin(2\pi y)$ and $u_1 = 3u_2$. The quantity of interest is the average value over $\Omega_1 \cup \Omega_2$, so $\psi_1 = \psi_2 = 1$.

This experiment demonstrates that the error representation formulas (6.4.3) and (6.4.6) may give different estimates if relatively few iterations are used in the operator decomposition method for the forward problem. We set a fixed number of iterations in the operator decomposition method for (6.4.8) and use (6.3.3) and (6.3.4) to approximate the solution. We solve the fully coupled adjoint iteratively using sufficient iterations for convergence. However, the number of iterations for the adjoint of the iterative system is determined by the number of iterations for the forward problem.

In Fig. 6.2, we plot the number of iterations versus the effectivity ratios for the two estimates. We observe that the adjoint for the operator decomposition method does not produce accurate estimates until a sufficient number of iterations for the forward problem have been carried out. This implies that the iterative component of the error, $c^{(k)} = u - u^{(k)}$, dominates the estimate.

If the iterative method converges, we expect that the effect of the initial guess and the first few iterations would diminish. This is reflected in Fig. 6.2 where we plot the L^2 norm of the iterative adjoint $\phi_1^{(k)}$ and $\phi_2^{(k)}$ over Ω_1 and

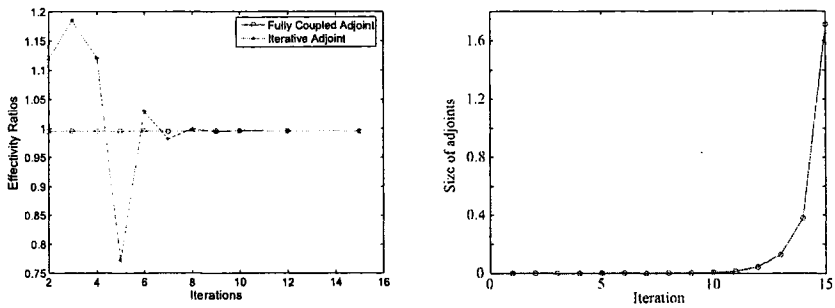

Figure 6.2: On the left, a comparison of the effectivity ratios using the two adjoints for a given number of iterations used for the operator decomposition method for the forward problem. On the right, we plot $\|\phi_1^{(k)}\| + \|\phi_2^{(k)}\|$ to show the decay of influence of errors that occur in previous iterations.

Ω_2 respectively at each iteration. We observe that the norm decays rapidly as k decreases, which indicates that the norm of the fixed point operator, and hence the norm of the adjoint fixed point operator, is small. Thus, the influence of errors in previous iterations are rapidly damped and have little influence on the current error. This implies that we can compute accurate estimates using the truncated series

$$\begin{aligned}
&(\psi_1^{(N)}, e_1^{(N)}) + (\psi_2^{(N)}, e_2^{(N)}) \\
&\qquad = \sum_{k=M}^{N} \Big((f_1, \phi_1^{(k)} - \pi_1\phi_1^{(k)}) - a_1(U_1^{(k)}, \phi_1^{(k)} - \pi_1\phi_1^{(k)}) \\
&\qquad\quad + (f_2, \phi_2^{(k)} - \pi_2\phi_2^{(k)}) - a_2(U_2^{(k)}, \phi_2^{(k)} - \pi_2\phi_2^{(k)}) \\
&\qquad\quad + (U_1^{(k)} - U_2^{(k-1)}, A_1\partial_n\phi_1^{(k)})_\Gamma \\
&\qquad\quad + (A_1\partial_n U_1^{(k)}, \pi_1\phi_1^{(k)})_\Gamma - (\sigma^{(k)}, \pi_2\phi_2^{(k)})_\Gamma \Big) \qquad (6.4.9)
\end{aligned}$$

where $1 \leq M \leq N$, in place of the full series (6.4.6). This significantly reduces the computational cost associated with computing a series of adjoint problems. In Fig. 6.3 we plot the effectivity ratio for the global average value where we truncate the error representation and compute only the last N terms.

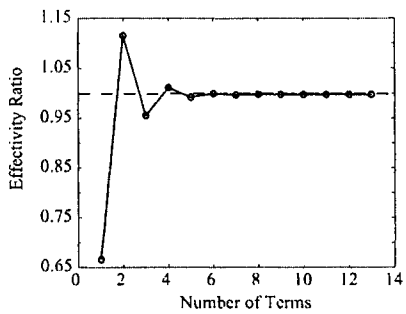

Figure 6.3: A comparison of the effectivity ratios computed using a truncated error representation.

6.4.4 Adaptive refinement

We use the *a posteriori* error estimate as the basis for adaptivity by employing the standard "optimization framework" after writing the estimate as a sum of element contributions and introducing norms [26, 27, 12, 10]. An element K is marked for refinement when the local error indicator is larger than a local tolerance, usually the global tolerance divided by the current number of elements.

Example 6.4.3. We demonstrate the adaptive procedure on the model problem (6.4.7) with the quantity of interest equal to the value of u_2 at the point $(1.75, 0.25)$, which we approximate by choosing $\psi_1 = 0$ and $\psi_2 = 400/\pi \exp(-400(x-1.75)^2 - 400(y-0.25)^2)$. Fig. 6.4 shows the results after 3 refinement steps. We solve (6.3.3)-(6.3.4) and observe that refinement occurs within Ω_1 along Γ which reflects the impact of numerical errors in the normal derivative on the quantity of interest.

6.5 An analysis of the loss of order

In practice, the iterative technique (6.3.3) and (6.3.4) is occasionally observed to result in $\mathbf{O}(h)$ convergence rather than the $\mathbf{O}(h^2)$ convergence

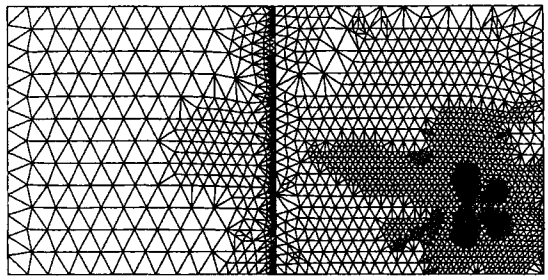

Figure 6.4: Adaptive mesh for the quantity of interest equal to the value of u_2 at the point $(1.75, 0.25)$.

that is obtained when solving the fully coupled problem. This loss of order results from passing the normal derivative of the finite element approximation, which is only $\mathbf{O}(h)$. Passing the recovered boundary flux restores the full order of convergence. We use the adjoint for the fully coupled problem to derive *a posteriori* error bounds for the iterative approximations in the case where $U_1^{(k)} = \pi_1 U_2^{(k-1)}$. The case where the Dirichlet values are updated using a relaxation technique can be analyzed using the same approach and gives identical results. Numerical examples are provided at the end of the section.

6.5.1 L^2 error bounds

Let $u_1 \in H^{1+\alpha_1}(\Omega_1)$ and $u_2 \in H^{1+\alpha_2}(\Omega_2)$ be the solutions to (6.2.1) with $0 \leq \alpha_1, \alpha_2 \leq 1$, and $U_1^{(k)}$ and $U_2^{(k)}$ be the solutions of (6.3.3) and (6.3.4) respectively at the k^{th} iteration. Let $\phi_1 \in H^{1+\alpha_1}(\Omega_1)$ and $\phi_2 \in H^{1+\alpha_2}(\Omega_2)$ and pose the adjoint problem (6.4.1) with $\psi_1 = e_1/\|e_1\|_{\Omega_1}$ and $\psi_2 = e_2/\|e_2\|_{\Omega_2}$. Starting with (6.4.3), integration by parts over each element K gives

$$\|e_1\|_{\Omega_1} + \|e_2\|_{\Omega_2} = I_1 + I_2 + I_3 + I_4,$$

where

$$I_1 = \sum_{K\in T_{1,h}} (f_1 - L_1U_1^{(k)}, \phi_1 - \pi_1\phi_1)_K + \frac{1}{2}([A_1\partial_n U_1^{(k)}], \phi_1 - \pi_1\phi_1)_{\partial K}$$
$$+ \sum_{K\in T_{2,h}} (f_2 - L_2U_2^{(k)}, \phi_2 - \pi_2\phi_2)_K + \frac{1}{2}([A_2\partial_n U_2^{(k)}], \phi_2 - \pi_2\phi_2)_{\partial K}$$
$$I_2 = (A_2\partial_n U_2^{(k)}, \phi_2 - \pi_2\phi_2)_\Gamma - (A_1\partial_n U_1^{(k)}, \phi_1 - \pi_1\phi_1)_\Gamma,$$
$$I_3 = (A_1\partial_n\phi_1, U_1^{(k)} - U_2^{(k)})_\Gamma,$$
$$I_4 = (A_1\partial_n U_1^{(k)}, \pi_2\phi_2)_\Gamma - (\sigma^{(k)}, \pi_1\phi_1)_\Gamma,$$

where $[A_i\partial_n U_i]$ denoting the jump in the normal derivative across an element edge.

The first term I_1 is a standard *a posteriori* error bound for elliptic problems and is not affected by non-matching triangulations along the interface or by transfer error. The second term I_2 is similar to the jump terms along element edges in I_1 and is the expected jump term when the triangulations align along the interface. The third term I_3 represents the jump in the Dirichlet values across the interface. Finally, the fourth term I_4 represents the difference between the flux passed from Ω_1 to Ω_2 and the flux obtained via the boundary-flux correction technique.

We first construct Lemmas 6.5.1 to 6.5.4 below to bound I_1 to I_4 individually. In each of these Lemmas we first provide the general bound when the triangulations do not match across the boundary and then show the simplification that arises for matching triangulations. We then combine these four lemmas into Theorems 6.5.1 and 6.5.2 which give error bounds for the basic iteration (6.3.3) and (6.3.4), and when using flux correction (6.3.8) respectively. These two theorems describe the general result when the triangulations do not match across the boundary while the simplification given matching triangulations is provided as a Corollary.

Lemma 6.5.1. *(Bound on I_1)*

$$I_1 \leq \sum_{K\in T_{1,h}} \begin{pmatrix} \|f_1 - L_1U_1^{(k)}\|_K \\ \frac{1}{2}\|h_K^{-1/2}[A_1\partial_n U_1^{(k)}]\|_{\partial K} \end{pmatrix} \cdot \begin{pmatrix} Ch_K^{1+\alpha_1}|\phi_1|_{K,1+\alpha_1} \\ Ch_K^{1+\alpha_1}|\phi_1|_{K,1+\alpha_1} \end{pmatrix}$$
$$+ \sum_{K\in T_{2,h}} \begin{pmatrix} \|f_2 - L_2U_2^{(k)}\|_K \\ \frac{1}{2}\|h_K^{-1/2}[A_2\partial_n U_2^{(k)}]\|_{\partial K} \end{pmatrix} \cdot \begin{pmatrix} Ch_K^{1+\alpha_2}|\phi_2|_{K,1+\alpha_2} \\ Ch_K^{1+\alpha_2}|\phi_2|_{K,1+\alpha_2} \end{pmatrix}.$$

Proof. Apply standard arguments using the Cauchy-Schwarz inequality and a trace inequality. □

If $u_1 \in H^2(\Omega_1)$, we can also provide a bound for the jump terms. Ordinarily, we would add and subtract $(A_1\partial_n u_1, \phi_1 - \pi_1\phi_1)_{\partial K}$ to estimate the jump term. However, this causes an iteration error $(A_1\partial_n u_1 - A_1\partial_n u_1^{(k)}, \phi_1 - \pi_1\phi_1)_\Gamma$, which may be quite complicated. Therefore, we let $\tilde{u}_1^{(k)}$ solve

$$\begin{cases} a_1(\tilde{u}_1^{(k)}, v) = (f_1, v)_{\Omega_1}, & \text{for all } v \in H_0^1(\Omega_1), \\ \tilde{u}_1^{(k)} = 0, & \mathbf{x} \in \partial\Omega\backslash\Gamma, \\ \tilde{u}_1^{(k)} = \pi_1 U_2^{(k-1)}, & \mathbf{x} \in \Gamma. \end{cases} \tag{6.5.1}$$

To be precise, $\tilde{u}_1^{(k)}$ is the exact weak solution over Ω_1 given the data from the previous iteration. Standard finite element theory can be applied to estimate

$$\|\tilde{u}_1^{(k)} - U_1^{(k)}\|_{\Omega_1,1} \leq Ch_1\|f_1\|_{\Omega_1}.$$

We add and subtract $(A_1\partial_n\tilde{u}_1^{(k)}, \phi_1 - \pi_1\phi_1)_{\partial K}$ and use the Cauchy-Schwarz and trace inequalities to obtain,

$$\|[A_1\partial_n U_1^{(k)}]\|_{\partial K} \leq Ch_K^{1/2}\|f_1\|_{\Omega_1}.$$

A similar argument holds if $u_2 \in H^2(\Omega_2)$.

Lemma 6.5.2. *(Bound on I_2) If the triangulations $T_{1,h}$ and $T_{2,h}$ do not match along the interface, then*

$$I_2 \leq \left(\|h_2^{-1/2}[A\partial_n U^{(k)}]\|_\Gamma\right)\cdot\left(Ch_2^{1+\alpha_2}|\phi_2|_{1+\alpha_2}\right)+\left(\|A_1\partial_n U_1^{(k)}\|_\Gamma\right)\cdot(\|\pi_1\phi_1-\pi_2\phi_1\|_\Gamma)$$

where

$$\begin{aligned}\|\pi_1\phi_1-\pi_2\phi_1\|_\Gamma &\leq \|\pi_1\phi_1-\phi_1\|_\Gamma+\|\phi_1-\pi_2\phi_1\|_\Gamma\\ &\leq C_1h_1^{1/2+\alpha_1}|\phi_1|_{\Omega_1,1+\alpha_1}+C_2h_2^{1/2+\alpha_2}|\phi_2|_{\Omega_2,1+\alpha_2}.\end{aligned}$$

If the triangulations $T_{1,h}$ and $T_{2,h}$ match along the interface, then

$$I_2 \leq \left(\|h_2^{-1/2}[A\partial_n U^{(k)}]\|_\Gamma\right)\cdot\left(Ch_2^{1+\alpha_2}|\phi_2|_{1+\alpha_2}\right).$$

where $[A\partial_n U^{(k)}] = A_2\partial_n U_2^{(k)} - A_1\partial_n U_1^{(k)}$ and $h_2 = \max_{K\in T_{2,h}} h_K$.

Proof. We add and subtract $(A_1\partial_n U_1^{(k)}, \phi_2-\pi_2\phi_2)_\Gamma$ and use $\phi_1=\phi_2$ on Γ to write

$$I_2 = (A_2\partial_n U_2^{(k)} - A_1\partial_n U_1^{(k)}, \phi_2-\pi_2\phi_2)_\Gamma + (A_1\partial_n U_1^{(k)}, \pi_1\phi_1-\pi_2\phi_1)_\Gamma.$$

Observe $\pi_1\phi_1=\pi_2\phi_2$ if the triangulations match along the interface. We apply the Cauchy-Schwarz and trace inequalities to complete the proof. □

If $u_2 \in H^2(\Omega_2)$, we may define $\tilde{u}_2^{(k)}$ similarly to $\tilde{u}_1^{(k)}$ in (6.5.1) and obtain

$$\|[A\partial_n U]\|_\Gamma \leq Ch_2^{1/2}\|f_2\|_{\Omega_1}.$$

Lemma 6.5.3. *(Bound on I_3) If the triangulations $T_{1,h}$ and $T_{2,h}$ do not match along the interface, then*

$$I_3 \leq (\|A_1\partial_n\phi_1\|_\Gamma) \cdot \left(\|U_2^{(k-1)} - U_2^{(k)}\|_\Gamma\right) + (\|A_1\partial_n\phi_1\|_\Gamma) \cdot \left(\|\pi_1 U_2^{(k-1)} - U_2^{(k-1)}\|_\Gamma\right)$$

where

$$\|\pi_1 U_2^{(k-1)} - U_2^{(k-1)}\|_\Gamma \leq C_1 h_1^{1/2+\alpha_1} |\hat{u}|_{\Omega_1,1+\alpha_1} + C_2 h_2^{1/2+\alpha_2} |\hat{u}|_{\Omega_2,1+\alpha_2}.$$

If the triangulations $T_{1,h}$ and $T_{2,h}$ match along the interface, then

$$I_3 \leq (\|A_1\partial_n\phi_1\|_\Gamma) \cdot \left(\|U_2^{(k-1)} - U_2^{(k)}\|_\Gamma\right).$$

Proof. First, observe that $U_1^{(k)} = \pi_1 U_2^{(k-1)}$ and rewrite I_3 by adding and subtracting $(A_1\partial_n\phi_1, U_2^{(k-1)})_\Gamma$ to get

$$I_3 = (A_1\partial_n\phi_1, \pi_1 U_2^{(k-1)} - U_2^{(k-1)})_\Gamma + (A_1\partial_n\phi_1, U_2^{(k-1)} - U_2^{(k)})_\Gamma.$$

To bound $\|\pi_1 U_2^{(k-1)} - U_2^{(k-1)}\|_\Gamma$ we introduce a new function $\hat{u}$ such that $\hat{u} \in H^{1+\alpha_1}(\Omega_1)$ and $\hat{u} \in H^{1+\alpha_2}(\Omega_2)$, as well as $\pi_1\hat{u} = \pi_1 U_2^{(k-1)}$ and $\pi_2\hat{u} = U_2^{(k-1)}$. This yields

$$\|\pi_1 U_2^{(k-1)} - U_2^{(k-1)}\|_\Gamma \leq C_1 h_1^{1/2+\alpha_1} |\hat{u}|_{\Omega_1,1+\alpha_1} + C_2 h_2^{1/2+\alpha_2} |\hat{u}|_{\Omega_2,1+\alpha_2}.$$

Observe that $\pi_1 U_2^{(k-1)} = U_2^{(k-1)}$ if the triangulations match along the interface. We apply the Cauchy-Schwarz inequality to complete the proof. □

Lemma 6.5.4. *(Bound on I_4) If the triangulations $T_{1,h}$ and $T_{2,h}$ do not match along the interface, then*

$$I_4 \leq \left(\|A_1 \partial_n U_1^{(k)} - A_1 \partial_n \tilde{u}_1^{(k)}\|_\Gamma + \|A_1 \partial_n \tilde{u}_1^{(k)} - \sigma^{(k)}\|_\Gamma \right) \cdot (\|\pi_2 \phi_2\|_\Gamma) + \left(\|\sigma^{(k)}\|_\Gamma\right) \cdot (\|\pi_2 \phi_2 - \pi_1 \phi_1\|_\Gamma)$$

where

$$\|\pi_2 \phi_2 - \pi_1 \phi_1\|_\Gamma \leq C_1 h_1^{1/2+\alpha_1} |\phi_1|_{\Omega_1, 1+\alpha_1} + C_2 h_2^{1/2+\alpha_2} |\phi_2|_{\Omega_2, 1+\alpha_2}.$$

If the triangulations $T_{1,h}$ and $T_{2,h}$ match along the interface, then

$$I_4 \leq \left(\|A_1 \partial_n U_1^{(k)} - A_1 \partial_n \tilde{u}_1^{(k)}\|_\Gamma + \|A_1 \partial_n \tilde{u}_1^{(k)} - \sigma^{(k)}\|_\Gamma \right) \cdot (\|\pi_2 \phi_2\|_\Gamma).$$

where $\tilde{u}_1^{(k)}$ is defined by (6.5.1).

Proof. We add and subtract $(\sigma^{(k)}, \pi_2 \phi_2)_\Gamma$ and use $\phi_1 = \phi_2$ along Γ to get

$$I_4 = (A_1 \partial_n U_1^{(k)} - \sigma^{(k)}, \pi_2 \phi_2)_\Gamma + (\sigma^{(k)}, \pi_2 \phi_2 - \pi_1 \phi_2)_\Gamma.$$

Observe $\pi_1 \phi_1 = \pi_2 \phi_2$ if the triangulations match along the interface. We add and subtract $(A_1 \partial_n \tilde{u}_1^{(k)}, \pi_2 \phi_2)_\Gamma$ to the first term and apply the Cauchy-Schwarz inequality to complete the proof. □

In practice, the error in the normal derivative is typically the same accuracy as the H^1 error, namely $\mathbf{O}(h_1^{\alpha_1})$. However, an application of the trace theorem only proves $\mathbf{O}(h_1^{\alpha_1/2})$ accuracy. This is not an important issue, however, since we intend to use the fact that this term is less accurate than the others, and therefore pollutes the L^2 error. We assume the error in the normal derivative can be bounded,

$$\|A_1 \partial_n U_1^{(k)} - \tilde{u}_1^{(k)}\|_\Gamma \leq C h_1^\beta \|f_1\|_{\Omega_1},$$

for $\alpha_1/2 \leq \beta \leq \alpha_1$. The error in the recovered boundary flux can be bounded

$$\|A_1\partial_n\tilde{u}_1^{(k)} - \sigma^{(k)}\|_\Gamma \leq CS_1h_1^{1+\alpha_1}\|f_1\|_{\Omega_1},$$

where S_1 is a stability factor defined by an associated Green's function [35, 56].

Theorem 6.5.1. *Assume the triangulations $T_{1,h}$ and $T_{2,h}$* do not *match along the interface Γ. If $U_1^{(k)}$ and $U_2^{(k)}$ solve (6.3.3) and (6.3.4) respectively, then the errors $e_1 = u_1 - U_1^{(k)}$ and $e_2 = u_2 - U_2^{(k)}$ satisfy*

$$\begin{aligned}
\|e_1\|_{\Omega_1} + \|e_2\|_{\Omega_2} \leq & \sum_{K\in T_{1,h}} \begin{pmatrix} \|f_1 - L_1U_1^{(k)}\|_K \\ \frac{1}{2}\|h_K^{-1/2}[A_1\partial_nU_1^{(k)}]\|_{\partial K} \end{pmatrix} \cdot \begin{pmatrix} Ch_K^{1+\alpha_1}|\phi_1|_{K,1+\alpha_1} \\ Ch_K^{1+\alpha_1}|\phi_1|_{K,1+\alpha_1} \end{pmatrix} \\
& + \sum_{K\in T_{2,h}} \begin{pmatrix} \|f_2 - L_2U_2^{(k)}\|_K \\ \frac{1}{2}\|h_K^{-1/2}[A_2\partial_nU_2^{(k)}]\|_{\partial K} \end{pmatrix} \cdot \begin{pmatrix} Ch_K^{1+\alpha_2}|\phi_2|_{K,1+\alpha_2} \\ Ch_K^{1+\alpha_2}|\phi_2|_{K,1+\alpha_2} \end{pmatrix} \\
& + \left(\|h_2^{-1/2}[A\partial_nU^{(k)}]\|_\Gamma\right) \cdot \left(Ch_2^{1+\alpha_2}|\phi_2|_{1+\alpha_2}\right) \\
& + \left(\|A_1\partial_n\phi_1\|_\Gamma\right) \cdot \left(\|U_2^{(k-1)} - U_2^{(k)}\|_\Gamma\right) \\
& + \left(Ch_1^\beta\|f_1\|_{\Omega_1} + CS_1h_1^{1+\alpha_1}\|f_1\|_{\Omega_1}\right) \cdot \left(\|\pi_2\phi_2\|_\Gamma\right) \\
& + C_1h_1^{1/2+\alpha_1}\left((S_2+S_4)|\phi_1|_{\Omega_1,1+\alpha_1} + S_3|\hat{u}|_{\Omega_1,1+\alpha_1}\right) \\
& + C_2h_2^{1/2+\alpha_2}\left((S_2+S_4)|\phi_2|_{\Omega_2,1+\alpha_2} + S_3|\hat{u}|_{\Omega_2,1+\alpha_2}\right)
\end{aligned}$$

with $\alpha_1/2 \leq \beta \leq \alpha_1$, S_1 is a stability factor independent of h, $S_2 = \|A_1\partial_nU_1^{(k)}\|_\Gamma$, $S_3 = \|A_1\partial_n\phi_1\|_\Gamma$, and $S_4 = \|\sigma^{(k)}\|_\Gamma$.

Corollary 6.5.1. *Assume the triangulations $T_{1,h}$ and $T_{2,h}$ match along the interface Γ. If $U_1^{(k)}$ and $U_2^{(k)}$ solve (6.3.3) and (6.3.4) respectively, then the*

errors $e_1 = u_1 - U_1^{(k)}$ *and* $e_2 = u_2 - U_2^{(k)}$ *satisfy*

$$\begin{aligned}
\|e_1\|_{\Omega_1} + \|e_2\|_{\Omega_2} \leq & \sum_{K \in T_{1,h}} \begin{pmatrix} \|f_1 - L_1 U_1^{(k)}\|_K \\ \frac{1}{2}\|h_K^{-1/2}[A_1 \partial_n U_1^{(k)}]\|_{\partial K} \end{pmatrix} \cdot \begin{pmatrix} Ch_K^{1+\alpha_1}|\phi_1|_{K,1+\alpha_1} \\ Ch_K^{1+\alpha_1}|\phi_1|_{K,1+\alpha_1} \end{pmatrix} \\
& + \sum_{K \in T_{2,h}} \begin{pmatrix} \|f_2 - L_2 U_2^{(k)}\|_K \\ \frac{1}{2}\|h_K^{-1/2}[A_2 \partial_n U_2^{(k)}]\|_{\partial K} \end{pmatrix} \cdot \begin{pmatrix} Ch_K^{1+\alpha_2}|\phi_2|_{K,1+\alpha_2} \\ Ch_K^{1+\alpha_2}|\phi_2|_{K,1+\alpha_2} \end{pmatrix} \\
& + \left(\|h_2^{-1/2}[A \partial_n U^{(k)}]\|_\Gamma\right) \cdot \left(Ch_2^{1+\alpha_2}|\phi_2|_{1+\alpha_2}\right) \\
& + \left(\|A_1 \partial_n \phi_1\|_\Gamma\right) \cdot \left(\|U_2^{(k-1)} - U_2^{(k)}\|_\Gamma\right) \\
& + \left(Ch_1^\beta \|f_1\|_{\Omega_1} + CS_1 h_1^{1+\alpha_1}\|f_1\|_{\Omega_1}\right) \cdot \left(\|\pi_2 \phi_2\|_\Gamma\right),
\end{aligned}$$

with $\alpha_1/2 \leq \beta \leq \alpha_1$ *and* S_1 *is a stability factor independent of* h_1.

It is clear that the term containing h_1^β decreases at a slower rate than the other terms. In fact, if we assume $u_1 \in H^2(\Omega_1)$ and $u_2 \in H^2(\Omega_2)$ then the other terms are $\mathbf{O}(h_1^2)$ or $\mathbf{O}(h_2^2)$, while the finite element flux is $\mathbf{O}(h_1)$ at best. Suppose that instead we solve (6.3.8) rather than (6.3.4), i.e., we pass $\sigma^{(k)}$ instead of the finite element flux. This changes the fourth term in the error representation formula to

$$I_4 = (\sigma^{(k)}, \pi_2 \phi_2)_\Gamma - (\sigma^{(k)}, \pi_1 \phi_1)_\Gamma.$$

Theorem 6.5.2. *Assume the triangulations* $T_{1,h}$ *and* $T_{2,h}$ do not *match along the interface* Γ. *If* $U_1^{(k)}$ *and* $U_2^{(k)}$ *solve (6.3.3) and (6.3.8) respectively,*

then the errors $e_1 = u_1 - U_1^{(k)}$ *and* $e_2 = u_2 - U_2^{(k)}$ *satisfy*

$$\begin{aligned}\|e_1\|_{\Omega_1} + \|e_2\|_{\Omega_2} \leq & \sum_{K\in T_{1,h}} \begin{pmatrix} \|f_1 - L_1U_1^{(k)}\|_K \\ \frac{1}{2}\|h_K^{-1/2}[A_1\partial_n U_1^{(k)}]\|_{\partial K}\end{pmatrix} \cdot \begin{pmatrix} Ch_K^{1+\alpha_1}|\phi_1|_{K,1+\alpha_1} \\ Ch_K^{1+\alpha_1}|\phi_1|_{K,1+\alpha_1}\end{pmatrix} \\ & + \sum_{K\in T_{2,h}} \begin{pmatrix} \|f_2 - L_2U_2^{(k)}\|_K \\ \frac{1}{2}\|h_K^{-1/2}[A_2\partial_n U_2^{(k)}]\|_{\partial K}\end{pmatrix} \cdot \begin{pmatrix} Ch_K^{1+\alpha_2}|\phi_2|_{K,1+\alpha_2} \\ Ch_K^{1+\alpha_2}|\phi_2|_{K,1+\alpha_2}\end{pmatrix} \\ & + \left(\|h_2^{-1/2}[A\partial_n U^{(k)}]\|_\Gamma\right) \cdot \left(Ch_2^{1+\alpha_2}|\phi_2|_{1+\alpha_2}\right) \\ & + \left(\|A_1\partial_n\phi_1\|_\Gamma\right) \cdot \left(\|U_2^{(k-1)} - U_2^{(k)}\|_\Gamma\right) \\ & + C_1h_1^{1/2+\alpha_1}\left((S_2+S_4)|\phi_1|_{\Omega_1,1+\alpha_1} + S_3|\hat{u}|_{\Omega_1,1+\alpha_1}\right) \\ & + C_1h_2^{1/2+\alpha_2}\left((S_2+S_4)|\phi_2|_{\Omega_2,1+\alpha_2} + S_3|\hat{u}|_{\Omega_2,1+\alpha_2}\right),\end{aligned}$$

where $S_2 = \|A_1\partial_n U_1^{(k)}\|_\Gamma$, $S_3 = \|A_1\partial_n\phi_1\|_\Gamma$, *and* $S_4 = \|\sigma^{(k)}\|_\Gamma$.

Corollary 6.5.2. *Assume the triangulations* $T_{1,h}$ *and* $T_{2,h}$ *match along the interface* Γ. *If* $U_1^{(k)}$ *and* $U_2^{(k)}$ *solve (6.3.3) and (6.3.8) respectively, then the errors* $e_1 = u_1 - U_1^{(k)}$ *and* $e_2 = u_2 - U_2^{(k)}$ *satisfy*

$$\begin{aligned}\|e_1\|_{\Omega_1} + \|e_2\|_{\Omega_2} \leq & \sum_{K\in T_{1,h}} \begin{pmatrix} \|f_1 - L_1U_1^{(k)}\|_K \\ \frac{1}{2}\|h_K^{-1/2}[A_1\partial_n U_1^{(k)}]\|_{\partial K}\end{pmatrix} \cdot \begin{pmatrix} Ch_K^{1+\alpha_1}|\phi_1|_{K,1+\alpha_1} \\ Ch_K^{1+\alpha_1}|\phi_1|_{K,1+\alpha_1}\end{pmatrix} \\ & + \sum_{K\in T_{2,h}} \begin{pmatrix} \|f_2 - L_2U_2^{(k)}\|_K \\ \frac{1}{2}\|h_K^{-1/2}[A_2\partial_n U_2^{(k)}]\|_{\partial K}\end{pmatrix} \cdot \begin{pmatrix} Ch_K^{1+\alpha_2}|\phi_2|_{K,1+\alpha_2} \\ Ch_K^{1+\alpha_2}|\phi_2|_{K,1+\alpha_2}\end{pmatrix} \\ & + \left(\|h_2^{-1/2}[A\partial_n U^{(k)}]\|_\Gamma\right) \cdot \left(Ch_2^{1+\alpha_2}|\phi_2|_{1+\alpha_2}\right) \\ & + \left(\|A_1\partial_n\phi_1\|_\Gamma\right) \cdot \left(\|U_2^{(k-1)} - U_2^{(k)}\|_\Gamma\right).\end{aligned}$$

Comparing Theorem 6.5.2 with Theorem 6.5.1 and Corollary 6.5.2 with Corollary 6.5.1, we see that the terms involving h^β have dropped out and the optimal order of convergence has been restored.

6.5.2 Numerical results

Example 6.5.1. Let $\Omega_1 = [0,1] \times [0,1]$ and $\Omega_2 = [1,2] \times [0,1]$ and assume that the triangulations match along $\Gamma = \{(x,y) | \, x = 1, \, 0 \leq y \leq 1\}$. Consider the elliptic interface problem

$$\begin{cases} -\nabla \cdot (\nabla u_1) = f_1, & x \in \Omega_1, \\ \begin{cases} u_1 = u_2, \\ \partial_n u_1 = \partial_n u_2, \end{cases} & x \in \Gamma, \\ -\nabla \cdot (\nabla u_2) = f_2, & x \in \Omega_2, \end{cases} \tag{6.5.2}$$

where the data $f_1(x,y)$ and $f_2(x,y)$ are chosen so the true solutions are $u_1 = u_2 = \sin(\pi x/2)\sin(2\pi y)$. The purpose of this experiment is to show how the asymptotic $\mathbf{O}(h_1^\beta)$ convergence rate may be masked by apparently minor changes to the problem, and therefore may be difficult to observe in all situations. We change the boundary conditions along the top boundary, solving (6.5.2) with the boundary conditions

$$\begin{cases} u_1 = 0, & \text{along } x = 0 \text{ and } y = 0, \\ \partial_n u_1 = 2\pi \sin(\pi x/2), & \text{along } y = 1, \\ u_2 = 0, & \text{along } x = 2 \text{ and } y = 0, \\ \partial_n u_2 = 2\pi \sin(\pi x/2), & \text{along } y = 1. \end{cases}$$

We set $\alpha = 1/2$ and iterate until the convergence criteria is met. For comparison, we also solve (6.5.2) using a fully coupled method on the same mesh in *COMSOL*. We see, in Figure 6.5, that the fully coupled solution converges quadratically, while the iterative solution converges linearly. Next, we solve (6.5.2) with Dirichlet boundary conditions

$$\begin{cases} u_1 = 0, & x \in \partial\Omega_1 \backslash \Gamma, \\ u_2 = 0, & x \in \partial\Omega_2 \backslash \Gamma, \end{cases}$$

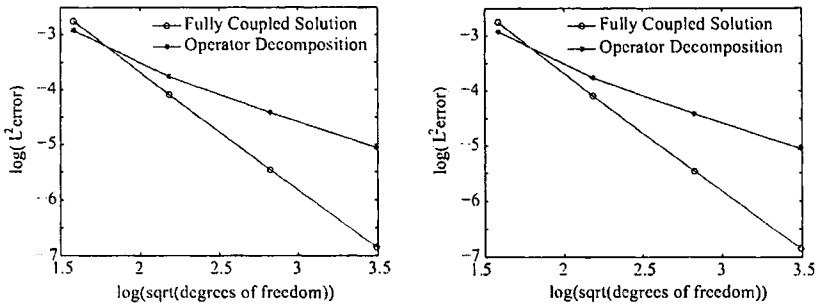

Figure 6.5: Comparison of L^2 error in the fully coupled approximation and the operator decomposition approximation over Ω_1 (left) and Ω_2 (right) with mixed boundary conditions on $\partial\Omega_i \backslash \Gamma$ for $i = 1, 2$.

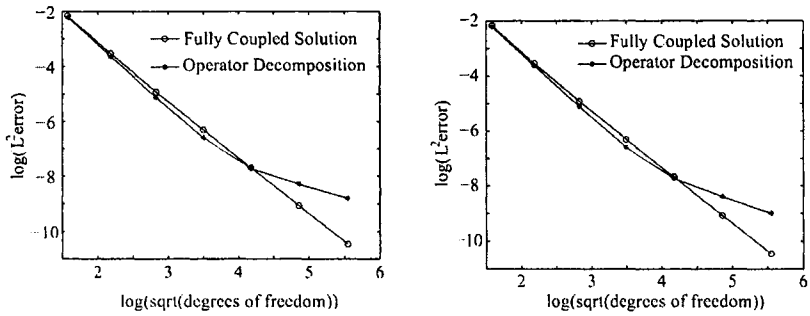

Figure 6.6: Comparison of L^2 error in the fully coupled approximation and the iterative approximation over Ω_1 (left) and Ω_2 (right) with Dirichlet boundary conditions on $\partial\Omega_i \backslash \Gamma$ for $i = 1, 2$.

and again we also solve the fully coupled problem for comparison. In Figure 6.6, we observe that the operator decomposition solution initially converges quadratically, but eventually the $\mathbf{O}(h_1^\beta)$ term dominates and the convergence is ultimately linear.

Example 6.5.2. In the final example we assume the triangulations do not match along the interface and consider (6.5.2) with the mixed boundary

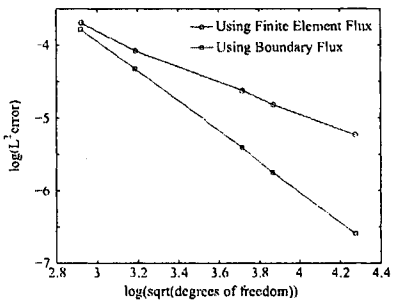

Figure 6.7: Comparison of L^2 error in the iterative approximation using the finite element flux and the boundary flux recovery method over $\Omega_1 \cup \Omega_2$ with mixed boundary conditions on $\partial\Omega_i \backslash \Gamma$ for $i = 1, 2$.

conditions

$$\begin{cases} u_1 = 0, & \text{along } x = 0, \\ A_1 \partial_n u_1 = 2\pi \sin(\pi x/2), & \text{along } y = 1, \\ A_1 \partial_n u_1 = -2\pi \sin(\pi x/2), & \text{along } y = 0, \\ u_2 = 0, & \text{along } x = 2, \\ A_2 \partial_n u_2 = 2\pi \sin(\pi x/2), & \text{along } y = 1, \\ A_2 \partial_n u_2 = -2\pi \sin(\pi x/2), & \text{along } y = 0. \end{cases}$$

First, we solve the problem iteratively by passing the finite element flux, $A_1 \partial_n U_1^{(k)}$. Next, we use the boundary flux method to compute and pass $\sigma^{(k)}$. In Figure 6.7, we compare the L^2 errors over $\Omega_1 \cup \Omega_2$ on a series of meshes. We see that the approximations using the finite element flux converge linearly, while the approximations using the boundary flux converge quadratically.

Chapter 7

FURTHER TOPICS IN FINITE ELEMENT METHODS

In this chapter, we review the standard error analysis for mixed finite element methods with particular emphasis on the Stokes and Navier-Stokes equations. See [15, 41, 40] for more details.

7.1 Mixed finite element methods

7.1.1 Abstract framework

We begin by proving some abstract convergence results for mixed finite element methods. Let V and S be Hilbert spaces with norms $\|\cdot\|_V$ and $\|\cdot\|_S$. Consider the variation problem seeking $u \in V$ and $p \in S$ such that

$$\begin{aligned} a(u,v) + b(v,p) &= (f,v), \\ b(u,q) &= (g,q), \end{aligned}$$

for all $v \in V$ and $q \in S$ with $f \in V^*$ and $g \in S^*$. The Stokes equations are a special case with $g = 0$. Let $V_h \subset V$ and $S_h \subset S$ be finite dimensional subspaces with $U \in V_h$ and $P \in S_h$ satisfying

$$\begin{aligned} a(U,v) + b(v,P) &= (f,v), \\ b(U,q) &= (g,q), \end{aligned}$$

for all $v \in V_h$ and $q \in S_h$. We make frequent use of the orthogonality relation

$$\begin{aligned} a(u-U,v) + b(v,p-P) &= 0, \\ b(u-U,q) &= 0, \end{aligned} \tag{7.1.1}$$

for all $v \in V_h$ and $q \in S_h$, and the linear space

$$Z_h(g) = \{v \in V_h \mid b(v,q) = (g,q),\ \forall q \in S_h\},$$

with the corresponding subspace of V_h

$$Z_h = \{v \in V_h \mid b(v,q) = 0,\ \forall q \in S_h\}.$$

We make the following assumptions on the bilinear forms:

(1) Continuity of $a(\cdot,\cdot)$

$$|a(u,v)| \leq C_1 \|u\|_V \cdot \|v\|_V,$$

for all $u, v \in V$.

(2) Z_h-coercivity of $a(\cdot,\cdot)$,

$$a(z_h, z_h) \geq \alpha \|z_h\|_V^2,$$

for all $z_h \in Z_h$.

(3) Continuity of $b(\cdot,\cdot)$,

$$|b(v,q)| \leq C_2 \|v\|_V \cdot \|q\|_S,$$

for all $v \in V$ and $q \in S$.

(4) The *inf-sup* condition,

$$\inf_{q \in S_h} \sup_{v \in V_h} \frac{b(v,q)}{\|v\|_V \cdot \|q\|_S} \geq \beta,$$

or, equivalently,

$$\sup_{v \in V_h} \frac{b(v,q)}{\|v\|_V} \geq \beta \|q\|_S,$$

for all $v \in V_h$ and $q \in S_h$.

Notice that the *inf-sup* condition is imposed over all of V_h and S_h, while the coercivity of $a(\cdot,\cdot)$ is only assumed over the subspace Z_h. It is well known that these four conditions and necessary and sufficient to prove existence and uniqueness of U and P [5, 16].

The first bound for $u-U$ uses only the first three conditions. Let z be an arbitrary element of $Z_h(g)$. Then,

$$\|u-U\|_V \leq \|u-z\|_V + \|U-z\|_V, \tag{7.1.2}$$

from the triangle inequality. Since $U, z \in Z_h(g)$, the quantity $(U-z) \in Z_h$. Now, we use coercivity to bound

$$\begin{aligned}\alpha\|U-z\|_V^2 &\leq a(U-z,U-z)\\ &= a(U,U-z) - a(z,U-z).\end{aligned}$$

From the orthogonality relation (7.1.1), we have

$$a(U,U-z) = a(u,U-z) + b(U-z,p-P),$$

since $(U-z) \in V_h$. This gives

$$\begin{aligned}\alpha\|U-z\|_V^2 &\leq a(U,U-z) - a(z,U-z)\\ &= a(u,U-z) + b(U-z,p-P) - a(z,U-z)\\ &= a(u-z,U-z) + b(U-z,p-P).\end{aligned}$$

Now, since $(U-z) \in Z_h$, $b(U-z,q) = 0$ for any $q \in S_h$. We apply this to replace $b(U-z,p-P)$ with $b(U-z,p-q)$ where q is an arbitrary element of S_h. Now, we have

$$\begin{aligned}\alpha\|U-z\|_V^2 &\leq a(u-z,U-z) + b(U-z,p-q)\\ &\leq C_1\|u-z\|_V \cdot \|U-z\|_V + C_2\|U-z\|_V \cdot \|p-q\|_S\\ \|U-z\|_V &\leq \frac{C_1}{\alpha}\inf_{z\in Z_h(g)}\|u-z\|_V + \frac{C_2}{\alpha}\inf_{q\in S_h}\|p-q\|_S.\end{aligned}$$

Substituting this into (7.1.2), we have

$$\|u - U\|_V \leq \left(1 + \frac{C_1}{\alpha}\right) \inf_{z \in Z_h(g)} \|u - z\|_V + \frac{C_2}{\alpha} \inf_{q \in S_h} \|p - q\|_S. \qquad (7.1.3)$$

This states that the error is proportional to the best approximation in $Z_h(g)$, rather than over all of V_h. This is problematic if we want to replace z with the interpolant πu, since this is unlikely to lie in $Z_h(g)$. We need to show

$$\inf_{z \in Z_h(g)} \|u - z\|_V \leq C \inf_{v \in V_h} \|u - v\|_V.$$

To do this, we use the *inf-sup* condition. Let v be an arbitrary element of V_h, and let $w \in V_h$ satisfy

$$b(w, q) = b(u - v, q),$$

for all $q \in S_h$ Continuity of $b(\cdot, \cdot)$ and the *inf-sup* condition imply that

$$\|w\|_V \leq \frac{C_2}{\beta} \|u - v\|_V.$$

Recall that $b(u, q) = (g, q)$ for all $q \in S_h$, so $(v + w) \in Z_h(g)$ as well. In addition,

$$\begin{aligned} \|u - (v + w)\|_V &\leq \|u - v\|_V + \|w\|_V \\ &\leq \left(1 + \frac{C_2}{\beta}\right) \|u - v\|_V. \end{aligned}$$

Since $v \in V_h$ was arbitrary, we have proven

$$\inf_{z \in Z_h(g)} \|u - z\|_V \leq \left(1 + \frac{C_2}{\beta}\right) \inf_{v \in V_h} \|u - v\|_V,$$

as well as

$$\|u - U\|_V \leq \left(1 + \frac{C_1}{\alpha}\right) \cdot \left(1 + \frac{C_2}{\beta}\right) \inf_{v \in V_h} \|u - v\|_V + \frac{C_2}{\alpha} \inf_{q \in S_h} \|p - q\|_S. \quad (7.1.4)$$

Next we derive a bound on $p - P$. Let q be an arbitrary element of S_h. We use the *inf-sup* condition, as well as the orthogonality relation (7.1.1) and continuity of the bilinear forms to estimate

$$\begin{aligned}\beta\|q - P\|_S &\leq \sup_{v\in V_h} \frac{|b(v, q - P)|}{\|v\|_V} \\ &= \sup_{v\in V_h} \frac{|b(v, p - P) + b(v, q - p)|}{\|v\|_V} \\ &= \sup_{v\in V_h} \frac{|-a(u - U, v) + b(v, q - p)|}{\|v\|_V} \\ &\leq C_1\|u - U\|_V + C_2\|p - q\|_S\end{aligned}$$

We combine this with the triangle inequality to give

$$\|p - P\|_S \leq \frac{C_1}{\beta}\|u - U\|_V + \left(1 + \frac{C_2}{\beta}\right) \inf_{q\in S_h} \|p - q\|_S.$$

and finally with (7.1.4) to obtain

$$\begin{aligned}\|p - P\|_S \leq \left(\frac{C_1}{\beta}\right) \cdot \left(1 + \frac{C_1}{\alpha}\right) \cdot \left(1 + \frac{C_2}{\beta}\right) \inf_{v\in V_h} \|u - v\|_V \\ + \left(1 + \frac{C_2}{\beta} + \frac{C_2}{\alpha}\right) \inf_{q\in S_h} \|p - q\|_S. \quad (7.1.5)\end{aligned}$$

7.1.2 The Stokes equations

Consider the Stokes equations with no-slip boundary conditions,

$$\begin{cases} -\nu\Delta u + \nabla p = f(x), & x \in \Omega, \\ -\nabla \cdot u = 0, & x \in \Omega, \\ u = 0, & x \in \partial\Omega. \end{cases} \quad (7.1.6)$$

The weak formulation seeks $u \in H_0^1(\Omega)$ and $p \in L_0^2(\Omega) = \{q \in L^2(\Omega) \mid \int_\Omega q\, dx = 0\}$ such that

$$a(u, v) + b(v, p) = (f, v)$$

$$b(u, q) = 0,$$

where $a(u,v) = \int_\Omega \nu \nabla u \cdot \nabla v \, dx$ and $b(u,p) = -\int_\Omega \nabla \cdot u \, p \, dx$. We point out that

$$\int_\Omega (\nabla p)\, v \, dx = \int_{\partial\Omega} (v \cdot n)\, p \, dS - \int_\Omega (\nabla \cdot v) \; p \, dx.$$

Therefore, $\int_\Omega (\nabla p)\, v \, dx = -b(v,p)$ as long as the boundary term vanishes. The quantity, $v \cdot n$, is zero whenever u or $u \cdot n$ is prescribed along the boundary. If this is not the case, i.e, an outflow boundary, we set $p = 0$ along this boundary.

To approximate the solutions to these equations, we choose finite dimensional subspaces $V_h \in H_0^1(\Omega)$ and $S_h \in L_0^2(\Omega)$ and seek $U \in V_h$ and $P \in S_h$ such that

$$a(U,v) + b(v,P) = (f,v),$$
$$b(U,q) = 0,$$

for all $v \in V_h$ and $q \in S_h$. A pair of subspaces satisfying the *inf-sup* condition is the Taylor Hood finite element pair, for which V_h is the space of continuous piecewise quadratic polynomials, and S_h is the space of continuous piecewise linear polynomials. The bilinear forms are easily seen to be continuous and coercive over $H_0^1(\Omega)$ and $L_0^2(\Omega)$. Assuming the true solutions are sufficiently smooth, i.e. $u \in H^3(\Omega) \cap H_0^1(\Omega)$ and $p \in H^2(\Omega) \cap L_0^2(\Omega)$, we define the projections $\pi_V : H^3 \to V_h$ and $\pi_S : H^2 \to S_h$ to be the Lagrange interpolants on the triangulation T_h. Then, the interpolation estimates

$$\|u - \pi_V u\|_s \leq C h^{3-s} \|u\|_{3-s},$$

$$\|p - \pi_S p\|_s \leq C h^{2-s} \|p\|_{2-s},$$

hold for $s = 0, 1$ [15].

Furthermore, we assume the elliptic regularity conditions,

$$\|u\|_2 + \|p\|_1 \leq \|f\|_0,$$

$$\|u\|_3 + \|p\|_2 \leq \|f\|_1,$$

are valid, although, as previously mentioned, the second inequality may be difficult to verify.

A direct application of (7.1.4) gives

$$\begin{aligned}\|u - U\|_1 &\leq \left(1 + \frac{C_1}{\alpha}\right) \cdot \left(1 + \frac{C_2}{\beta}\right) \|u - \pi_V u\|_1 + \frac{C_2}{\alpha}\|p - \pi_S p\|_0 \\ &\leq C_3 h^2 \|u\|_3 + C_4 h^2 \|p\|_2 \\ &\leq C_5 h^2 \|f\|_1. \end{aligned} \tag{7.1.7}$$

Similarly, the abstract error bound for the pressure (7.1.5) gives

$$\begin{aligned}\|p - P\|_0 &\leq C_6 \|u - \pi_V u\|_1 + C_7 \|p - \pi_S p\|_0 \\ &\leq C_8 h^2 \|u\|_3 + C_9 h^2 \|p\|_2 \\ &\leq C_{10} h^2 \|f\|_1. \end{aligned} \tag{7.1.8}$$

7.1.3 L^2 error bounds for the Stokes equations

Let ϕ and z solve the adjoint for the Stokes equations,

$$\begin{cases} -\nu\Delta\phi + \nabla z = \phi_u, & \mathbf{x} \in \Omega, \\ -\nabla \cdot \phi = \psi_p, & \mathbf{x} \in \Omega, \\ \phi = 0, & \mathbf{x} \in \partial\Omega. \end{cases} \tag{7.1.9}$$

We use ψ_u and ψ_p to denote the data for the adjoint since these frequently represent linear functionals for a quantity of interest in the errors $e_u = u - U$ and $e_p = p - P$ respectively. We assume that the regularity condition

$$\|\phi\|_2 + \|z\|_1 \leq \|\psi_u\|_0,$$

holds for the adjoint problem. We set $\psi_u = e_u$ and $\psi_p = 0$, and multiply the system by $(e_u, 0)^T$ and integrate by parts to give

$$\begin{aligned}
\|e_u\|_0^2 =& a(\phi, e_u) + (\nabla z, e_u) - (\nabla \cdot \phi, e_p) \\
=& a(e_u, \phi) + b(e_u, z) + b(\phi, e_p) \\
=& a(e_u, \phi - \pi_U \phi) + b(\phi - \pi_U \phi, e_p) + b(e_u, z - \pi_S z) \\
\leq& C\|e_u\|_1 \cdot \|\phi - \pi_U \phi\|_1 + C\|\phi - \pi_U \phi\|_1 \cdot \|e_p\|_0 + C\|e_u\|_1 \cdot \|z - \pi_S z\|_0 \\
\leq& Ch\|e_u\|_1 \left(\|\phi\|_2 + \|z\|_1\right) + Ch\|e_p\|_0 \cdot \|\phi\|_2 \\
\leq& Ch\|e_u\|_1 \cdot \|e_u\|_0 + Ch\|e_p\|_0 \cdot \|e_u\|_0.
\end{aligned}$$

After dividing by $\|e_u\|_0$, we have the *a-priori* estimate

$$\|e_u\|_0 \leq Ch\left(\|e_u\|_1 + \|e_p\|_0\right).$$

Combining this result with (7.1.7) and (7.1.8) easily leads to the optimal order *a-priori* L^2 bound

$$\|u - U\|_0 \leq Ch^3\|f\|_1. \tag{7.1.10}$$

An *a-posteriori* error bound can be derived as follows

$$\begin{aligned}
\|e_u\|_0^2 =& a(e_u, \phi - \pi_U\phi) + b(\phi - \pi_U\phi, e_p) + b(e_u, z - \pi_S z) \\
=& (f, \phi - \pi_U\phi) - a(U, \phi - \pi_U\phi) - b(\phi - \pi_u\phi, P) - b(U, z - \pi_S z) \\
=& \sum_{K\in T_h} (f + \nu\Delta U - \nabla P, \phi - \pi_U\phi)_K + ([\nu\partial_n U], \phi - \pi_U\phi)_{\partial K} \\
&+ (\nabla\cdot U, z - \pi_S z)_K \\
\leq& \sum_{K\in T_h} C_1\|f + \nu\Delta U - \nabla P\|_{0,K}\cdot\|\phi - \pi_U\phi\|_{0,K} \\
&+ C_2\|e_u\|_{1,K}^{1/2}\cdot\|e_u\|_{2,K}^{1/2}\cdot\|\phi - \pi_U\phi\|_{0,K}^{1/2}\cdot\|\phi - \pi_U\phi\|_{1,K}^{1/2} \\
&+ C_3\|\nabla\cdot U\|_{0,K}\cdot\|z - \pi_S z\|_{0,K} \\
\leq& \sum_{K\in T_h} C_4 h_K^2\|f + \nu\Delta U - \nabla P\|_{0,K}\|e_u\|_{0,K} \\
&+ C_5 h_K^{5/2}\|f\|_{1,K}^{1/2}\cdot\|e_u\|_{2,K}^{1/2}\cdot\|e_u\|_{0,K} \\
&+ C_6 h_K\|\nabla\cdot U\|_{0,K}\cdot\|e_u\|_{0,K}.
\end{aligned}$$

As before, we require the convergence of residuals to complete the estimate. Fortunately, one of the residuals is easy to bound,

$$\begin{aligned}
\|\nabla\cdot U\|_{0,K} =& \|\nabla\cdot e_u\|_{0,K} \\
\leq& C\|e_u\|_{1,K} \\
\leq& Ch_K^2\|f\|_{1,K}.
\end{aligned}$$

To bound the second residual we assume that the weak solutions, u and p, are smooth enough to be solutions to (7.1.6) as well. Then, using an inverse estimate and the energy error bounds, we have

$$\begin{aligned}
\|f + \nu\Delta U - \nabla P\|_{0,K} =& \| - \nu\Delta u + \nabla p + \nu\Delta U - \nabla P\|_{0,K} \\
\leq& C\|e\|_{2,K} + C\|p\|_{1,K} \\
\leq& Ch_K^{-1}\|e\|_{1,K} + Ch_K^{-1}\|p\|_{0,K} \\
\leq& Ch_K\|f\|_{1,K}.
\end{aligned}$$

7.2 Navier-Stokes equations

We are in position to talk about the error in numerical solutions to the Navier Stokes equations with no-slip boundary conditions,

$$\begin{cases} -\nu\Delta u + (u\cdot\nabla)u + \nabla p = f, & \mathbf{x}\in\Omega, \\ -\nabla\cdot u = 0, & \mathbf{x}\in\Omega, \\ u = 0, & \mathbf{x}\in\partial\Omega. \end{cases} \tag{7.2.1}$$

The weak formulation seeks $u \in H_0^1(\Omega)$ and $p \in L_0^2(\Omega)$ such that

$$\begin{aligned} a(u,v) + c(u,u,v) + b(v,p) &= (f,v), \\ b(u,q) &= 0, \end{aligned} \tag{7.2.2}$$

for all $v \in H_0^1(\Omega)$ and $q \in L_0^2(\Omega)$, with the trilinear form defined as

$$c(u,w,v) = \int_\Omega (w\cdot\nabla)\, u\; v\; dx.$$

We assume continuity,

$$|a(u,v)| \le C_1\|u\|_1\cdot\|v\|_1$$

$$|b(v,p)| \le C_2\|v\|_1\cdot\|p\|_0$$

$$|c(u,w,v)| \le C_3\|u\|_1\cdot\|w\|_1\cdot\|v\|_1$$

and the *inf-sup* condition,

$$\inf_{q\in L_0^2(\Omega)}\sup_{v\in H_0^1(\Omega)}\frac{b(v,q)}{\|v\|_1\|q\|_0} \ge \beta.$$

Existence of a unique solution to (7.2.2) can be shown if

$$\frac{N}{\nu^2}\|f\| < 1,$$

where

$$N = \sup_{u,v,w\in H_0^1(\Omega)}\frac{|c(u,v,w)|}{\|u\|_1\|v\|_1\|w\|_1},$$

eg., for sufficiently small data, or small Reynolds numbers [40, 41].

7.2.1 A finite element method

Let $V_h \subset H_0^1(\Omega)$ and $S_h \subset L_0^2(\Omega)$ be finite dimensional subspaces associated with a quasi-uniform triangulation of Ω, T_h. Since the problem is nonlinear, we make an initial guess, u^0, and use Newton's method to find $U^k \in V_h$ and $P^k \in S_h$ such that

$$a(U^k, v) + c(U^k, U^{k-1}, v) + c(U^{k-1}, U^k, v) + b(v, P) = (f, v) + c(U^{k-1}, U^{k-1}, v),$$
$$b(U^k, q) = 0, \tag{7.2.3}$$

for all $v \in V_h$ and $q \in S_h$. Subtracting (7.2.3) from (7.2.2) leads to the orthogonality relation

$$a(e_u, v) + c(u, u, v) - c(U^k, U^k, v) + b(v, e_p) = R_k, \quad \forall v \in V_h$$
$$b(e_u, q) = 0, \quad \forall q \in S_h \tag{7.2.4}$$

where R_k is a residual given by

$$R_k = c(U^k - U^{k-1}, U^k - U^{k-1}, v).$$

It is common to neglect this term since the continuity of the trilinear form gives

$$|R_h| \leq C_3 \|U^k - U^{k-1}\|_1 \cdot \|U^k - U^{k-1}\|_1 \cdot \|v\|_1,$$

which is small whenever Newton's method has converged.

Next, we define the space of weakly divergence free functions

$$Z_h = \{v \in V_h \mid b(v, q) = 0, \ \forall q \in S_h\}.$$

We assume $a(\cdot, \cdot)$ is coercive over $Z_h \times Z_h$, i.e.

$$a(z, z) \geq \alpha \|z\|_1^2, \ \forall z \in Z_h.$$

Next, we must choose V_h and S_h so that the *discrete inf-sup* condition

$$\inf_{q \in S_h} \sup_{v \in V_h} \frac{b(v,q)}{\|v\|_1 \|q\|_0} \geq \beta,$$

holds for $\beta > 0$ independent of h. To accomplish this, we choose V_h to be the space of continuous piecewise quadratic polynomials, and S_h to be the the space of continuous piecewise lineare polynomials over T_h.

7.2.2 Discrete maximum principles

For the Stokes equations, the above assumptions were enough to prove optimal order error bounds. Unfortunately, this is not the case for the Navier-Stokes equations due to the presence of the nonlinear term. Similar to the reaction diffusion equation in section 3, we need coercivity over $Z_h \times Z_h$

$$a(w,w) + c(U,U,w) - c(z,z,w) \geq \gamma \|w\|_1^2,$$

for some positive γ, where U is the finite element approximation, z an arbitrary element of Z_h, and $w = U - z$ their difference. First, we follow [40] and consider a slight perturbation of the Navier Stokes equations.

Consider a simple application of the divergence theorem:

$$c(u,v,w) = -c(u,w,v) - \int_\Omega (\nabla \cdot u) w \cdot v \, dx,$$

where $u, v, w \in H_0^1(\Omega)$. This leads the following observations on this trilinear form:

(1) If $\nabla \cdot u = 0$, then $c(u,v,w) = -c(u,w,v)$ for any $v, w \in H_0^1(\Omega)$.

(2) If $\nabla \cdot u = 0$, then $c(u,w,w) = 0$.

(3) Define $\tilde{c}(u,v,w) = \frac{1}{2}(c(u,v,w) - c(u,w,v))$. If $\nabla \cdot u = 0$, then $\tilde{c}(u,v,w) = c(u,v,w)$.

Of particular interest is the new trilinear form, $\tilde{c}(\cdot,\cdot,\cdot)$. Notice that this is antisymmetric, i.e. $\tilde{c}(u,v,w) = -\tilde{c}(u,w,v)$, even if $\nabla \cdot u \neq 0$. In [40], they consider a perturbation of the Navier-Stokes equations

$$\begin{aligned} a(u,v) + \tilde{c}(u,u,v) + b(v,p) &= (f,v), \\ b(u,q) &= 0, \end{aligned} \tag{7.2.5}$$

At the continuous level, these equations have the same solution as the Navier Stokes equations. The advantage of using these equations, rather than (7.2.2), is that the antisymmetry of the trilinear form is preserved, which has an important consequence in proving coercivity. Define the mesh dependent quantities

$$N_h = \sup_{u,v,w \in V_h} \frac{|\tilde{c}(u,v,w)|}{\|u\|_1 \|v\|_1 \|w\|_1},$$

and

$$\|f\|_h = \sup_{v \in V_h} \frac{|(f,v)|}{\|v\|_1},$$

where $N_h \to N$ and $\|f\|_h \to \|f\|_0$ as $h \downarrow 0$, see [40]. Let $U \in Z_h$ be the finite element approximation to (7.2.5), z be an arbitrary element of Z_h, and $w = U - z$. Then

$$\begin{aligned} a(w,w) + \tilde{c}(U,U,w) - \tilde{c}(z,z,w) &= a(w,w) + \tilde{c}(w,U,w) + \tilde{c}(z,w,w) \\ &= a(w,w) + \tilde{c}(w,U,w) \\ &\geq \nu \|w\|_1^2 - N_h \|U\|_1 \|w\|_1^2 \\ &\geq \left(\nu - N_h \frac{\|f\|_h}{\nu}\right) \|w\|_1^2 \\ &= \nu \left(1 - N_h \frac{\|f\|_h}{\nu^2}\right) \|w\|_1^2. \end{aligned}$$

Hence, the perturbed problem is coercive whenever

$$N_h \frac{\|f\|_h}{\nu^2} < 1.$$

The key to the above analysis, is that the antisymmetry of $\tilde{c}$ implies $\tilde{c}(z, w, w) = 0$. In the next section, we examine the original trilinear form, $c(\cdot, \cdot, \cdot)$, for which $c(z, w, w) \neq 0$.

The next goal is to provide criteria for coercivity of the weak form of the Navier Stokes equations over $Z_h \times Z_h$. In the last section, we considered a perturbation of the Navier Stokes equations, and although the two problems are the same at the continuous level, they are not the same at the finite dimensional level. Hence, a finite element approximation to one may not be the same as a finite element approximation to the other. To be precise,

$$c(u, v, w) = \tilde{c}(u, v, w),$$

but

$$c(U, v, w) \neq \tilde{c}(U, v, w),$$

since $\nabla \cdot U \neq 0$. Furthermore, we would like to consider general problems of the form

$$\begin{aligned} a(u, v) + c(u, u, v) + b(v, p) &= (f, v), \\ b(u, q) &= (g, q), \end{aligned} \tag{7.2.6}$$

where g is a smooth function. Notice that this corresponds to $\nabla \cdot u = g$, so c is not antisymmetric even at the continuous level. We need to provide conditions under which

$$a(w, w) + c(U, w, w) - c(z, w, w) \geq \gamma \|w\|_1^2,$$

for some positive γ, where U is a finite element approximation, z an arbitrary element of $Z_h(g)$, and $w = U - z$. Proceeding as before, we have

$$\begin{aligned} a(w,w) + c(U,U,w) - c(z,z,w) &= a(w,w) + c(w,U,w) + c(z,w,w) \\ &\geq \nu\|w\|_1^2 - N_h\|U\|_1\|w\|_1^2 + c(z,w,w) \\ &\geq \nu\|w\|_1^2 - N_h\frac{\|f\|_h}{\nu}\|w\|_1^2 + c(z,w,w), \end{aligned}$$

but now, $c(z,w,w)$ does not drop out. Instead, we use the divergence theorem to write

$$c(z,w,w) = -\frac{1}{2}\int_\Omega (\nabla \cdot z) w \cdot w \; dx.$$

Now, $z \in Z_h(g)$, which means

$$-(\nabla \cdot z, q) = (g,q), \quad \forall q \in S_h,$$

but $w \cdot w \notin S_h$. So we cannot simply replace $-\nabla \cdot z$ with g. Using the above relation, we have

$$\begin{aligned} \int_\Omega (\nabla \cdot z) w \cdot w \; dx &= \int_\Omega (\nabla \cdot z)(w \cdot w - \pi_S(w \cdot w)) \; dx + \int_\Omega g \pi_S(w \cdot w) \; dx \\ &\leq C_I h \|\nabla \cdot z\|_0 \cdot \|w\|_1^2 + \|g\|_0 \cdot \|\pi_S(w \cdot w)\| \\ &\leq C(z) h \|w\|_1^2 + C\|g\|_0 \|w\|_1^2. \end{aligned}$$

where $C(z)$ is a constant depending on z and the interpolation constant C_I, but not on h or U. The other constant, C, depends on the constant from the Poincare inequality and the stability constant from

$$\|\pi_S \phi\|_0 \leq C_S \|\phi\|_0.$$

Combining this with the previous result gives

$$a(w,w)+c(U,U,w)-c(z,z,w) \geq \left(\nu - N_h\frac{\|f\|_h}{\nu} - \frac{1}{2}C(z)h - \frac{1}{2}C\|g\|_0\right)\|w\|_1^2,$$

which leads to the condition

$$N_h \frac{\|f\|_h}{\nu^2} + \frac{C(z)}{2\nu} h + \frac{C}{2\nu} \|g\|_0 < 1.$$

Remark 7.2.1. *This is still incomplete. The "constant" $C(z)$ depends on the arbitrary function z, and the bound must hold for all $z \in Z_h(g)$. It would be better if we could bound either $\|\nabla \cdot z\|_0$, or $\|z\|_1$, for z satisfying $b(z,q) = (g,q)$ for all $q \in S_h$.*

7.2.3 The adjoint

Before presenting optimal order L^2 bounds on $u - U$, we discuss the adjoint for the Navier Stokes equations. First, we rewrite the Navier Stokes equations in the form

$$\begin{cases} -\nu \Delta u + c(u) + \nabla p = f, \\ -\nabla \cdot u = 0. \end{cases} \tag{7.2.7}$$

We define the linearized form

$$\overline{c} = \int_0^1 c'(su + (1-s)U)\, ds,$$

which satisfies

$$(c(u) - c(U), \phi) = (\overline{c}e, \phi) = (e, \overline{c}^* \phi).$$

The linearized adjoint of (7.2.7) is

$$\begin{cases} -\nu \Delta \phi + \overline{c}^* \phi + \nabla z = \psi, \\ -\nabla \cdot \phi = 0. \end{cases} \tag{7.2.8}$$

For the Navier Stokes equations,

$$\overline{c}^* \phi = -\alpha \cdot \nabla \phi + (\nabla \alpha - \nabla \cdot \alpha)\phi,$$

where $\alpha = \frac{1}{2}(u + U)$. The weak form of the linearized adjoint is given by

$$\begin{aligned} a^*(\phi, w) + b(w, z) &= (\psi, w), \\ b(\phi, \eta) &= 0, \end{aligned} \tag{7.2.9}$$

where

$$a^*(\phi, w) = (\nu\nabla\phi, \nabla w) + (\overline{c}^*\phi, w).$$

To prove existence and uniqueness of weak adjoint solutions, we only need to show coercivity of $a^*(\cdot,\cdot)$ since the *inf-sup* condition is the same as the forward problem. Consider the following

$$\begin{aligned} a^*(\phi,\phi) =& (\nu\nabla\phi, \nabla\phi) + (\overline{c}^*\phi, \phi) \\ =& \nu\|\phi\|_1^2 - (\alpha\cdot\nabla\phi, \phi) + ((\nabla\alpha - \nabla\cdot\alpha)\phi, \phi) \\ =& \nu\|\phi\|_1^2 + \left((\nabla\alpha - \frac{1}{2}(\nabla\cdot\alpha))\phi, \phi\right). \end{aligned}$$

Now, we use the definition of α, the fact that $\nabla\cdot u = 0$, the *a-priori* bound $\|u\|_1 \leq C\|f\|_0$, and the Poincare inequality $\|v\|_0 \leq C_P\|v\|_1$, to conclude

$$a^*(\phi,\phi) \geq \nu\|\phi\|_1^2 - \left(\frac{C_P}{\nu} + \frac{C_P h}{4\nu}\right)\|f\|_0 \cdot \|\phi\|_1^2.$$

Thus, the linearized adjoint problem is coercive if

$$\frac{C_P}{\nu^2}\left(1 + \frac{h}{4}\right)\|f\|_0 < 1.$$

7.2.4 Error bounds

Now, we assume that T_h is a quasi-uniform triangulation of Ω and that the weak formulation is coercive over $Z_h \times Z_h$, i.e.

$$a(w,w) + c(U,U,w) - c(z,z,w) \geq \gamma\|w\|_1^2,$$

for some $\gamma > 0$, with U, z and w defined as before. In addition, we assume the iterative method to solve (7.2.2) has converged, making R_h negligible.

Theorem 7.2.1. *The finite element approximation, U, is quasi-optimal over Z_h,*

$$\|u - U\|_1 \leq C\left(\inf_{z\in Z_h}\|u - z\|_1 + \inf_{q\in S_h}\|p - q\|_0\right), \tag{7.2.10}$$

where the constant C may depend on u and z, but not on h.

Proof. Let z be an arbitrary element of Z_h, $w = U - z$, and q an arbitrary element of S_h. We use the coercivity condition and the orthogonality relations (7.2.4) to bound

$$\begin{aligned}
\gamma\|w\|_1^2 \leq & a(w,w) + c(U,U,w) - c(z,z,w) \\
= & a(u-z,w) + c(u,u,w) - c(z,z,w) + b(w,p-P) \\
= & a(u-z,w) + c(u,u,w) - c(z,z,w) + b(w,p-q) \\
\leq & C_1\|u-z\|_1 \cdot \|w\|_1 + C_L\|u-z\|_1\|w\|_1 + C_2\|w\|_1 \cdot \|p-q\|_0 \\
\leq & C\left(\inf_{z\in Z_h}\|u-z\|_1 + \inf_{q\in S_h}\|p-q\|_1\right)
\end{aligned}$$

where C_L is a constant which may depend on u and z. If c is uniformly Lipshitz, then we may remove this dependence. The triangle inequality

$$\|u-U\|_1 \leq \|u-z\|_1 + \|U-z\|_1,$$

completes the proof.

Theorem 7.2.2. *If $b(\cdot,\cdot)$ satisfies the* inf-sup *condition, then*

$$\|u-U\|_1 \leq C\left(\inf_{v\in V_h}\|u-v\|_1 + \inf_{q\in S_h}\|p-q\|_0\right). \tag{7.2.11}$$

Furthermore, if $u \in H^3(\Omega)\cap H_0^1(\Omega)$ and $p \in H^2(\Omega)\cap L_0^2(\Omega)$, and $\|u\|_3 + \|p\|_2 \leq C\|f\|_1$, then

$$\|u-U\|_1 \leq Ch^2\|f\|_1. \tag{7.2.12}$$

Proof. The first part was proven section 4 using the *inf-sup* condition and continuity of $b(\cdot,\cdot)$. The second part can be shown by choosing $v = \pi_V u$ and $q = \pi_S p$, and applying standard interpolation results.

Theorem 7.2.3. *If $b(\cdot,\cdot)$ satisfies the* inf-sup *condition, then*

$$\|p-P\|_0 \leq C\left(\inf_{v\in V_h}\|u-v\|_1 + \inf_{q\in S_h}\|p-q\|_0\right). \tag{7.2.13}$$

Furthermore, if $u \in H^3(\Omega) \cap H_0^1(\Omega)$ *and* $p \in H^2(\Omega) \cap L_0^2(\Omega)$, *and* $\|u\|_3 + \|p\|_2 \leq C\|f\|_1$, *then*

$$\|p - P\|_0 \leq Ch^2\|f\|_1. \tag{7.2.14}$$

Proof. Let $q \in S_h$ be arbitrary. We use the *inf-sup* condition, the orthogonality relation (7.2.4), and continuity of $b(\cdot,\cdot)$ to obtain

$$\begin{aligned}
\beta\|q - P\|_0 \leq & \sup_{v \in V_h} \frac{|b(v, q - P)|}{\|v\|_1} \\
= & \sup_{v \in V_h} \frac{|b(v, p - P) + b(q - p)|}{\|v\|_1} \\
= & \sup_{v \in V_h} \frac{|-a(u - U, v) - c(u, u, v) + c(U, U, v) + b(v, q - p)|}{\|v\|_1} \\
\leq & C\left(\|u - U\|_1 + \|p - q\|_0\right) \\
\|q - P\|_0 \leq & C\left(\inf_{v \in V_h} \|u - v\|_1 + \inf_{q \in S_h} \|p - q\|_0\right).
\end{aligned}$$

This proves (7.2.13), and (7.2.14) follows by choosing v and q to be the interpolants.

Theorem 7.2.4. *Assume that the iterative method to solve (7.2.2) has converged so that the remainder term,* R_h, *is negligible, and that the weak form of the linearized adjoint problem (7.2.8) is coercive with solutions* $\phi \in H^2(\Omega) \cap H_0^1(\Omega)$ *and* $z \in H^2(\Omega) \cap L_0^2(\Omega)$ *satisfying the elliptic regularity condition*

$$\|\phi\|_2 + \|z\|_1 \leq C\|\psi\|_0.$$

Then,

$$\|u - U\|_0 \leq Ch\left(\|u - U\|_1 + \|p - P\|_0\right),$$

and if the assumptions of Theorems 5.2 and 5.3 hold

$$\|u - U\|_0 \leq Ch^3\|f\|_1.$$

Proof. Let $\psi = u - U$ in (7.2.8), multiply the system by $(e_u, e_p)^T = (u - U, p - P)^T$, and integrate by parts, giving

$$\begin{aligned}\|u-U\|_0^2 =& a^*(\phi, e_u) + b(e_u, z) + b(\phi, e_p) \\ =& (\nu\nabla\phi, \nabla e_u) + (\overline{c}^*\phi, e_u) + b(e_u, z) + b(\phi, e_p) \\ =& a(e_u, \phi) + c(u, u, \phi) - c(U, U, \phi) + b(e_u, z) + b(\phi, e_p).\end{aligned}$$

Next we apply orthogonality, Lipschitz continuity, and the continuity of the bilinear forms to give

$$\begin{aligned}\|u-U\|_0^2 =& a(e_u, \phi - \pi_V\phi) + c(u, u, \phi - \pi_V\phi) - c(U, U, \phi - \pi_V\phi) \\ &+ b(e_u, z - \pi_S z) + b(\phi - \pi_V\phi, e_p) \\ \leq& C\left(\|e_u\|_1 \cdot \|\phi - \pi_V\phi\|_1 + \|e_u\|_1 \cdot \|z - \pi_S z\|_1 + \|\phi - \pi_V\phi\|_1 \cdot \|e_p\|_0\right) \\ \leq& C\left(h\|e_u\|_1 \cdot \|\phi\|_2 + h\|e_u\|_1 \cdot \|z\|_1 + h\|e_p\|_0 \cdot \|\phi\|_2\right) \\ \leq& Ch\left(\|e_u\|_1 + \|e_p\|_0\right) \\ \|u-U\|_0 \leq& Ch\left(\|u-U\|_1 + \|p-P\|_0\right),\end{aligned}$$

which proves the first assertion. A direct application of Theorems 5.2 and 5.3 proves the second assertion.

Theorem 7.2.5. *Assume that the iterative method to solve (7.2.2) has converged so that the remainder term, R_h, is negligible, and that the weak form of the linearized adjoint problem (7.2.8) is coercive with solutions $\phi \in H^2(\Omega) \cap H_0^1(\Omega)$ and $z \in H^2(\Omega) \cap L_0^2(\Omega)$ with*

$$\|\phi\|_2 + \|z\|_1 \leq C\|\psi\|_0,$$

then

$$\|u-U\|_0 \leq \sum_{K \in T_h} C_1 S_K h_K^3 \|f\|_{1,K} + C_2 J_K h_K^3 \|f\|_{1,K},$$

where $S_K = \|\phi\|_{2,K}$ and $J_K = \|z\|_{1,K}$.

Proof. Let $\psi = \frac{u-U}{\|u-U\|_0}$ in (7.2.8), multiply the system by $(e_u, e_p)^T$, and integrate by parts, giving

$$\begin{aligned}
\|u-U\|_0 =& a^*(\phi, e_u) + b(e_u, z) + b(\phi, e_p) \\
=& (\nu\nabla\phi, \nabla e_u) + (\vec{c}^*\phi, e_u) + b(e_u, z) + b(\phi, e_p) \\
=& a(e_u, \phi) + c(u, u, \phi) - c(U, U, \phi) + b(e_u, z) + b(\phi, e_p) \\
=& a(e_u, \phi - \pi_V\phi) + c(u, u, \phi - \pi_V\phi) - c(U, U, \phi - \pi_V\phi) \\
&+ b(e_u, z - \pi_S z) + b(\phi - \pi_V\phi, e_p) \\
=& (f, \phi - \pi_V\phi) - a(U, \phi - \pi_V\phi) - c(U, U, \phi - \pi_V\phi) \\
&- b(U, z - \pi_S z) - b(\phi - \pi_V\phi, P) \\
=& \sum_{K\in T_h} (f + \nu\Delta U - (U\cdot\nabla)U - \nabla P, \phi - \pi_V\phi)_K \\
&+ ([\nu\partial_n U], \phi - \pi_V\phi)_{\partial K} + (\nabla\cdot U, z - \pi_S z)_K \\
=& \sum_{K\in T_h} I_1 + I_2 + I_3
\end{aligned}$$

The bounds for the second and third terms are exactly the same as in section 4.2 for the Stokes equations. For the first term we use an inverse estimate and the H^1 error bound:

$$\begin{aligned}
I_1 &= \sum_{K\in T_h} (f + \nu\Delta U - (U\cdot\nabla)U - \nabla P, \phi - \pi_V\phi)_K \\
&\le \sum_{K\in T_h} \|f + \nu\Delta U - (U\cdot\nabla)U - \nabla P\|_K \cdot \|\phi - \pi_V\phi\|_K \\
&\le \sum_{K\in T_h} Ch_K^2 \|f + \nu\Delta U - (U\cdot\nabla)U - \nabla P\|_K \cdot \|\phi\|_{2,K} \\
&\le \sum_{K\in T_h} Ch_K^2 \|e\|_{2,K} \cdot \|\phi\|_{2,K} \\
&\le \sum_{K\in T_h} Ch_K^2 \left(h_K^{-1}\|e\|_{1,K}\right) \cdot \|\phi\|_{2,K} \\
&\le \sum_{K\in T_h} Ch_K^3 \|f\|_{1,K} \cdot \|\phi\|_{2,K}.
\end{aligned}$$

7.3 The Boussinesq equations

The equations of motion for a heat conducting Newtonian fluid can be approximated by the Boussinesq equations when the temperature difference is not large enough to affect the density, and the flow is driven by buoyancy. We write the Boussinesq equation as

$$\begin{cases} -\mu\Delta\mathbf{u} + \rho_0 (\mathbf{u}\cdot\nabla)\,\mathbf{u} + \nabla p + \rho_0\beta\mathbf{g}T = \rho_0\mathbf{g}\,(1+\beta T_{\text{ref}}))\,, & x\in\Omega, \\ -\nabla\cdot\mathbf{u} = 0, & x\in\Omega, \\ -\kappa\Delta T + \rho_0 c_p\mathbf{u}\cdot\nabla T = Q, & x\in\Omega, \end{cases} \tag{7.3.1}$$

where μ denotes the viscocity, β the coefficient of thermal expansion, ρ_0 the density and c_p the specific heat. For simplicity, we assume no-slip boundary conditions on the fluid and specified heat for the temperature, i.e.

$$\mathbf{u} = 0 \text{ and } T = T_B, \quad x\in\partial\Omega.$$

The weak formulation seeks $\mathbf{u}\in(H_0^1(\Omega))^d$, $p\in L_0^2(\Omega)$, and $T\in H^1(\Omega)$ such that $T = T_B$ for $x\in\partial\Omega$ and

$$\begin{cases} a_1(\boldsymbol{u},\boldsymbol{v}) + c_1(\boldsymbol{u},\boldsymbol{u},\boldsymbol{v}) + b(\boldsymbol{v},p) + d(T,\boldsymbol{v}) = (\mathbf{f},\boldsymbol{v}), \\ b(\boldsymbol{u},q) = 0, \\ a_2(T,w) + c_2(\boldsymbol{u},T,w) = (Q,w), \end{cases} \tag{7.3.2}$$

for all $v \in (H_0^1(\Omega))^d$, $q \in L_0^2(\Omega)$ and $w \in H_0^1(\Omega)$, where $\mathbf{f} = \rho_0 \boldsymbol{g}(1 + \beta T_{ref})$ and

$$
\begin{aligned}
a_1(\boldsymbol{u}, \boldsymbol{v}) &= \int_\Omega \mu \nabla \boldsymbol{u} : \nabla \boldsymbol{v} \; dx \\
a_2(T, w) &= \int_\Omega \kappa \nabla T \cdot \nabla w \; dx \\
b(\boldsymbol{v}, q) &= -\int_\Omega (\nabla \cdot \boldsymbol{v}) q \; dx \\
c_1(\boldsymbol{u}, \boldsymbol{v}, \mathbf{z}) &= \int_\Omega \rho_0 \left(\boldsymbol{u} \cdot \nabla\right) \boldsymbol{v} \cdot \mathbf{z} \; dx \\
c_2(\boldsymbol{u}, T, w) &= \int_\Omega \rho_0 c_p \left(\boldsymbol{u} \cdot \nabla T\right) \; w \; dx \\
d(T, \boldsymbol{v}) &= \int_\Omega \rho_0 \beta T \mathbf{g} \cdot \boldsymbol{v} \; dx
\end{aligned}
$$

We assume the bilinear forms are continuous

$$
\begin{aligned}
|a_1(\boldsymbol{u}, \boldsymbol{v})| &\leq C_1 \|\boldsymbol{u}\|_1 \cdot \|\boldsymbol{v}\|_1 \\
|a_2(T, w)| &\leq C_2 \|T\|_1 \cdot \|w\|_1 \\
|b(\boldsymbol{v}, q)| &\leq C_3 \|\boldsymbol{v}\|_1 \cdot \|q\|_0 \\
|d(T, \boldsymbol{v})| &\leq C_4 \|T\|_0 \cdot \|\boldsymbol{v}\|_0
\end{aligned}
$$

and the nonlinear terms are Lipshitz continuous

$$
\begin{aligned}
|c_1(\boldsymbol{u}, \boldsymbol{v}, \mathbf{z}) - c_1(\boldsymbol{U}, \mathbf{V}, \mathbf{z})| &\leq C_5 \left(\|\boldsymbol{u} - \boldsymbol{U}\|_1 + \|\boldsymbol{v} - \mathbf{V}\|_1\right) \cdot \|\mathbf{z}\|_1 \\
|c_2(\boldsymbol{u}, T, w) - c_2(\boldsymbol{U}, S, w)| &\leq C_6 \left(\|\boldsymbol{u} - \boldsymbol{U}\|_1 + \|T - S\|_1\right) \cdot \|w\|_1.
\end{aligned}
$$

7.3.1 A finite element method

To discretize, let τ_h be a quasi-uniform triangulation of Ω and define the piecewise polynomial spaces

$$W_h = \{v \in H^1(\Omega) \cap C(\Omega) \mid v \in P^2(K)\},$$

$$W_{h,0} = \{v \in W_h \mid w = 0, \; x \in \partial\Omega\},$$

$$\mathbf{V}_h = \{\boldsymbol{v} \in \mathbf{H}_0^1(\Omega) \cap \mathbf{C}(\Omega) \mid v_i \in P^2(K)\},$$

$$S_h = \{q \in L_0^2(\Omega) \cap C(\Omega) \mid q \in P^1(K)\},$$

$$Z_h = \{v \in \mathbf{V}_h \mid b(\boldsymbol{v}, q) = 0,\ \forall\ q \in S_h.\}$$

We use π_V, π_S, and π_W to represent a projection into $\mathbf{V}_h$, S_h and W_h respectively. We refer to $\mathbf{V}_h$ and S_h as the Taylor-Hood pair, which are known to satisfy the discrete *inf-sup* condition

$$\inf_{q \in S_h} \sup_{v \in \mathbf{V}_h} \frac{b(v,q)}{\|v\|_1 \|q\|_0} \geq \beta.$$

The finite element method seeks $\boldsymbol{u}_h \in \mathbf{V}_h$, $p_h \in S_h$, and $T_h \in W_h$ such that $T_h = \pi T_B$ along $\partial\Omega$ and

$$\begin{cases} a_1(\boldsymbol{u}_h, \boldsymbol{v}) + c_1(\boldsymbol{u}_h, \boldsymbol{u}_h, \boldsymbol{v}) + b(\boldsymbol{v}, p_h) + d(T_h, \boldsymbol{v}) = (\mathbf{f}, \boldsymbol{v}), \\ b(\boldsymbol{u}_h, q) = 0, \\ a_2(T_h, w) + c_2(\boldsymbol{u}_h, T_h, w) = (Q, w), \end{cases} \tag{7.3.3}$$

for all $\boldsymbol{v} \in \mathbf{V}_h$, $q \in S_h$ and $w \in W_h$.

7.3.2 The adjoint

Let $B(\boldsymbol{u}, p, T)$ denote the Boussinesq operator. Defining the adjoint to the Boussinesq operator is complicated by the fact that it is a nonlinear operator. Let $\boldsymbol{u}_h$, p_h and T_h be approximation to $\boldsymbol{u}$, p, and T respectively and define $\mathbf{e} = (\boldsymbol{u} - \boldsymbol{u}_h, p - p_h, T - T_h)$. Formally, we define the linearized adjoint operator, $\overline{B}^*$, such that

$$(B(\boldsymbol{u}, p, T), \boldsymbol{\phi}) - (B(\boldsymbol{u}_h, p_h, T_h), \boldsymbol{\phi}) = (\overline{B}(\mathbf{e}), \boldsymbol{\phi}) = (\mathbf{e}, \overline{B}^*(\boldsymbol{\phi}))$$

where the linearized operator, $\overline{B}$, is defined by

$$\overline{B}(\boldsymbol{e}) = \int_0^1 \mathbf{B}'(s\boldsymbol{u} + (1-s)\boldsymbol{u}_h, sp + (1-s)p_h, sT + (1-s)T_h) \cdot \boldsymbol{e}\ ds.$$

where

$$\mathbf{B}'(\boldsymbol{u},p,T)=\left(\frac{\partial B}{\partial \boldsymbol{u}}(\boldsymbol{u},p,T),\frac{\partial B}{\partial p}(\boldsymbol{u},p,T),\frac{\partial B}{\partial T}(\boldsymbol{u},p,T)\right)^T.$$

First, we define the linearized forms. Let $c_1(\boldsymbol{v},\boldsymbol{v})=\rho_0(\boldsymbol{v}\cdot\nabla)\boldsymbol{v}$. The Gateaux derivative in the direction δ is given by

$$\begin{aligned} c_1'(\boldsymbol{v},\boldsymbol{v})\delta &= \lim_{\epsilon\to 0}\frac{c_1(\boldsymbol{v}+\epsilon\delta,\boldsymbol{v}+\epsilon\delta)-c_1(\boldsymbol{v},\boldsymbol{v})}{\epsilon} \\ &= \frac{\rho_0\left((\boldsymbol{v}+\epsilon\delta)\cdot\nabla\right)(\boldsymbol{v}+\epsilon\delta)-\rho_0(\boldsymbol{v}\cdot\nabla)\boldsymbol{v}}{\epsilon} \\ &= \rho_0(\delta\cdot\nabla)\boldsymbol{v}+\rho_0(\boldsymbol{v}\cdot\nabla)\delta \end{aligned}$$

We define the linearized form, $\bar{c}_1$ such that

$$\begin{aligned} \bar{c}_1(\boldsymbol{e_u}) &= \int_0^1 (\rho_0(\boldsymbol{e_u}\cdot\nabla)(\boldsymbol{u}s+\boldsymbol{u}_h(1-s))+\rho_0((\boldsymbol{u}s+\boldsymbol{u}_h(1-s))\cdot\nabla)\boldsymbol{e_u})\ ds \\ &= \rho_0(\boldsymbol{e_u}\cdot\nabla)\left(\frac{1}{2}\boldsymbol{u}+\frac{1}{2}\boldsymbol{u}_h\right)+\rho_0\left(\left(\frac{1}{2}\boldsymbol{u}+\frac{1}{2}\boldsymbol{u}_h\right)\cdot\nabla\right)\boldsymbol{e_u} \end{aligned}$$

The adjoint linearized operator, $\bar{c}_1^*(\phi)$, is defined such that $(\bar{c}_1(\boldsymbol{e_u}),\phi)=(\boldsymbol{e_u},\bar{c}_1^*(\phi))$. Using the divergence theorem, we obtain

$$\bar{\boldsymbol{c}}_1^*(\boldsymbol{\phi})=\rho_0\nabla\left(\frac{1}{2}\boldsymbol{u}+\frac{1}{2}\boldsymbol{u}_h\right)\cdot\boldsymbol{\phi}-\rho_0\left(\frac{1}{2}\boldsymbol{u}+\frac{1}{2}\boldsymbol{u}_h\right)\cdot\nabla\boldsymbol{\phi}-\rho_0\left(\nabla\cdot\left(\frac{1}{2}\boldsymbol{u}+\frac{1}{2}\boldsymbol{u}_h\right)\right)\boldsymbol{\phi}.$$

Similarly, we define $c_2(\boldsymbol{v},w)=\rho_0 c_p(\boldsymbol{v}\cdot\nabla)w$. The partial Gateaux derivative in the direction δ is given by

$$\begin{aligned} \frac{\partial c_2(\boldsymbol{v},w)}{\partial \boldsymbol{v}}\delta &= \lim_{\epsilon\to 0}\frac{c_2(\boldsymbol{v}+\epsilon\delta,w)-c_2(\boldsymbol{v},w)}{\epsilon} \\ &= \frac{\rho_0 c_p\left((\boldsymbol{v}+\epsilon\delta)\cdot\nabla\right)(w)-\rho_0 c_p(\boldsymbol{v}\cdot\nabla)w}{\epsilon} \\ &= \rho_0 c_p(\delta\cdot\nabla)w \end{aligned}$$

We define the linearized form, $\bar{c}_{2\boldsymbol{u}}$ such that

$$\bar{c}_{2\boldsymbol{u}}(\boldsymbol{e_u}) = \int_0^1 (\rho_0 c_p(\boldsymbol{e_u}\cdot\nabla)(T_F s+T_{F,h}(1-s))\ ds$$

$$= \rho_0 c_p (\boldsymbol{e_u} \cdot \nabla) \left(\frac{1}{2} T_F + \frac{1}{2} T_{F,h} \right)$$

The partial Gateaux derivative in the direction γ is given by

$$\begin{aligned} \frac{\partial c_2(\boldsymbol{v}, w)}{\partial w} \gamma &= \lim_{\epsilon \to 0} \frac{c_2(\boldsymbol{v}, w + \epsilon\gamma) - c_2(\boldsymbol{v}, w)}{\epsilon} \\ &= \frac{\rho_0 c_p \left((\boldsymbol{v}) \cdot \nabla \right) (w + \epsilon\gamma) - \rho_0 c_p (\boldsymbol{v} \cdot \nabla) w}{\epsilon} \\ &= \rho_0 c_p (\boldsymbol{v} \cdot \nabla) \gamma \end{aligned}$$

We define the linearized form, $\bar{c}_{2T}$ so that

$$\begin{aligned} \bar{c}_{2T}(e_{T_F}) &= \int_0^1 \left(\rho_0 c_p ((s\boldsymbol{u} + (1-s)\boldsymbol{u}_h) \cdot \nabla) \right) (e_{T_F}) \, ds \\ &= \rho_0 c_p \left(\left(\frac{1}{2} \boldsymbol{u} + \frac{1}{2} \boldsymbol{u}_h \right) \cdot \nabla \right) (e_{T_F}) \end{aligned}$$

The adjoint linearized operators, $\bar{c}^*_{2\boldsymbol{u}}(\theta)$ and $\bar{c}^*_{2T}(\theta)$, are defined to satisfy $(\bar{c}_{2\boldsymbol{u}}(\boldsymbol{e_u}), \theta) = (\boldsymbol{e_u}, \bar{c}^*_{2\boldsymbol{u}}(\theta))$ and $(\bar{c}_{2T}(e_{T_F}), \theta) = (e_{T_F}, \bar{c}^*_{2T}(\theta))$ respectively. Using the divergence theorem, we find

$$\bar{\boldsymbol{c}}^*_{2u}(\theta) = \rho_0 c_p \nabla \left(\frac{1}{2} T + \frac{1}{2} T_h \right) \theta,$$

and

$$\bar{c}^*_{2T}(\theta) = -\rho_0 c_p \left(\frac{1}{2} \boldsymbol{u} + \frac{1}{2} \boldsymbol{u}_h \right) \cdot \nabla\theta - \rho_0 c_p \left(\nabla \cdot \left(\frac{1}{2} \boldsymbol{u} + \frac{1}{2} \boldsymbol{u}_h \right) \right) \theta.$$

We write the formal adjoint to the Boussinesq equations in strong form

$$\begin{cases} -\mu \Delta \boldsymbol{\phi} + \bar{\boldsymbol{c}}^*_1(\boldsymbol{\phi}) + \nabla z + \bar{\boldsymbol{c}}^*_{2u}(\theta) = \boldsymbol{\psi_u}, & x \in \Omega, \\ -\nabla \cdot \boldsymbol{\phi} = \psi_p, & x \in \Omega, \\ -k \Delta\theta + \bar{c}^*_{2T}(\theta) + \rho_0 \beta \boldsymbol{g} \cdot \boldsymbol{\phi} = \psi_T, & x \in \Omega, \end{cases} \tag{7.3.4}$$

with the adjoint boundary conditions

$$\begin{cases} \boldsymbol{\phi} = \boldsymbol{0}, & x \in \Gamma_{\boldsymbol{u},D}, \\ \theta = 0, & x \in \Gamma_{T_F,D}. \end{cases} \tag{7.3.5}$$

To derive the error representation, we multiply the system (7.3.4) by $(\boldsymbol{e_u}, e_p, e_T)^T$, apply the Divergence theorem and the definition of the linearized adjoint terms:

$$
\begin{aligned}
(\boldsymbol{\psi_u}, \boldsymbol{e_u}) + (\psi_p, e_p) + (\psi_T, e_T) = \\
a_1(\boldsymbol{e_u}, \boldsymbol{\phi}) + c_1(\boldsymbol{u}, \boldsymbol{u}, \boldsymbol{\phi}) - c_1(\boldsymbol{u_h}, \boldsymbol{u_h}, \boldsymbol{\phi}) \\
+ b(\boldsymbol{e_u}, z) + d(e_T, \boldsymbol{\phi}) + b(\boldsymbol{\phi}, e_p) \\
+ a_2(e_T, \theta) + c_2(\boldsymbol{u}, T, \theta) - c_2(\boldsymbol{u_h}, T_h, \theta)
\end{aligned}
$$

Next, we apply Galerkin orthogonality for nonlinear problems,

$$
\begin{aligned}
(\boldsymbol{\psi_u}, \boldsymbol{e_u}) + (\psi_p, e_p) + (\psi_T, e_T) = \\
a_1(\boldsymbol{e_u}, \boldsymbol{\phi} - \pi_V\boldsymbol{\phi}) + c_1(\boldsymbol{u}, \boldsymbol{u}, \boldsymbol{\phi} - \pi_V\boldsymbol{\phi}) - c_1(\boldsymbol{u_h}, \boldsymbol{u_h}, \boldsymbol{\phi} - \pi_V\boldsymbol{\phi}) \\
+ b(\boldsymbol{e_u}, z - \pi_S z) + d(e_T, \boldsymbol{\phi} - \pi_V\boldsymbol{\phi}) + b(\boldsymbol{\phi} - \pi_V\boldsymbol{\phi}, e_p) \\
+ a_2(e_T, \theta - \pi_W\theta) + c_2(\boldsymbol{u}, T, \theta - \pi_W\theta) - c_2(\boldsymbol{u_h}, T_h, \theta - \pi_W\theta) \quad (7.3.6)
\end{aligned}
$$

and use the fact that the true solutions solve (7.3.2) to replace these with the data,

$$
\begin{aligned}
(\boldsymbol{\psi_u}, \boldsymbol{e_u}) + (\psi_p, e_p) + (\psi_T, e_T) = \\
(\mathbf{f}, \boldsymbol{\phi} - \pi_V\boldsymbol{\phi}) - a_1(\boldsymbol{u_h}, \boldsymbol{\phi} - \pi_V\boldsymbol{\phi}) - c_1(\boldsymbol{u_h}, \boldsymbol{u_h}, \boldsymbol{\phi} - \pi_V\boldsymbol{\phi}) \\
- b(\boldsymbol{u_h}, z - \pi_S z) - d(T_h, \boldsymbol{\phi} - \pi_V\boldsymbol{\phi}) - b(\boldsymbol{\phi} - \pi_V\boldsymbol{\phi}, p_h) \\
(Q, \theta - \pi_W\theta) - a_2(T_h, \theta - \pi_W\theta) - c_2(\boldsymbol{u_h}, T_h, \theta - \pi_W\theta).
\end{aligned}
$$

7.3.3 Error bounds

We assume that τ_h is a quasi-uniform triangulation of Ω and that the weak formulation is coercive,

$$\gamma\|\mathbf{y}\|_1^2 \leq a_1(\boldsymbol{w},\boldsymbol{w}) + c_1(\boldsymbol{u}_h,\boldsymbol{u}_h,\boldsymbol{w}) - c_1(\boldsymbol{z},\boldsymbol{z},\boldsymbol{w}) + d(\eta,\boldsymbol{w}) + a_2(\eta,\eta) + c_2(\boldsymbol{u}_h,T_h,\eta) - c_2(\boldsymbol{z},s,\eta), \quad (7.3.7)$$

for some $\gamma > 0$, with $\boldsymbol{u}_h$ and T_h the finite element solutions, $\boldsymbol{z}$ an arbitrary element of Z_h, s an arbitrary element of W_h, $\boldsymbol{w} = \boldsymbol{u}_h - \boldsymbol{z}$, $\eta = T_h - s$, and $\boldsymbol{y} = (\boldsymbol{w},\eta)^T$. In addition, we assume the iterative method to solve (7.2.2) has converged, making the nonlinear residual negligible.

Theorem 7.3.1. *The finite element approximations, $\boldsymbol{u}_h$ and T_h, are quasi-optimal,*

$$\|\boldsymbol{u}-\boldsymbol{u}_h\|_1+\|T-T_h\|_1 \leq C\left(\inf_{\boldsymbol{z}\in Z_h}\|\boldsymbol{u}-\boldsymbol{z}\|_1 + \inf_{q\in S_h}\|p-q\|_0 + \inf_{s\in W_h}\|T-s\|_1\right), \quad (7.3.8)$$

where the constant C may depend on $\boldsymbol{u}$ and $\boldsymbol{z}$, but not on h.

Proof. Let $\boldsymbol{z}$ be an arbitrary element of Z_h, $\boldsymbol{w} = \boldsymbol{u}_h - \boldsymbol{z}$, q an arbitrary element of S_h, s an arbitrary element of W_h, $\eta = T_h - s$, and $\boldsymbol{y} = (\boldsymbol{w},\eta)^T$ We use the coercivity condition and the Galerkin orthogonality relations to

bound

$$\begin{aligned}
\gamma\|\boldsymbol{y}\|_1^2 \leq & a_1(\boldsymbol{w},\boldsymbol{w}) + c_1(\boldsymbol{u}_h,\boldsymbol{u}_h,\boldsymbol{w}) - c_1(\boldsymbol{z},\boldsymbol{z},\boldsymbol{w}) + d(\eta,\boldsymbol{w}) \\
& + a_2(\eta,\eta) + c_2(\boldsymbol{u}_h,T_h,\eta) - c_2(\boldsymbol{z},s,\eta) \\
= & a(\boldsymbol{u}-\boldsymbol{z},\boldsymbol{w}) + c(\boldsymbol{u},\boldsymbol{u},\boldsymbol{w}) - c(\boldsymbol{z},\boldsymbol{z},\boldsymbol{w}) + d(T-s,\boldsymbol{w}) \\
& + b(\boldsymbol{w},p-p_h) + a_2(T-s,\eta) + c_2(\boldsymbol{u},T,\eta) - c_2(\boldsymbol{z},s,\eta) \\
= & a(\boldsymbol{u}-\boldsymbol{z},\boldsymbol{w}) + c(\boldsymbol{u},\boldsymbol{u},\boldsymbol{w}) - c(\boldsymbol{z},\boldsymbol{z},\boldsymbol{w}) + d(T-s,\boldsymbol{w}) \\
& + b(\boldsymbol{w},p-q) + a_2(T-s,\eta) + c_2(\boldsymbol{u},T,\eta) - c_2(\boldsymbol{z},s,\eta) \\
\leq & C\|\boldsymbol{u}-\boldsymbol{z}\|_1 \cdot \|\boldsymbol{w}\|_1 + C\|\boldsymbol{u}-\boldsymbol{z}\|_1\|\boldsymbol{w}\|_1 \\
& + C\|T-s\|_1 \cdot \|\boldsymbol{w}\|_1 + C\|\boldsymbol{w}\|_1 \cdot \|p-q\|_0 \\
& + C\|T-s\|_1 \cdot \|\eta\|_1 + C\left(\|T-s\|_1 + \|\boldsymbol{u}-\boldsymbol{z}\|_1\right)\|\eta\|_1 \\
\|\boldsymbol{y}\|_1 \leq & C\left(\inf_{\boldsymbol{z}\in Z_h}\|\boldsymbol{u}-\boldsymbol{z}\|_1 + \inf_{q\in S_h}\|p-q\|_1 + \inf_{s\in W_h}\|T-s\|_1\right)
\end{aligned}$$

The triangle inequality

$$\|\boldsymbol{u}-\boldsymbol{u}_h\|_1 + \|T-T_h\|_1 \leq \|\boldsymbol{u}-\boldsymbol{z}\|_1 + \|T-s\|_1 + \|\boldsymbol{y}\|_1,$$

completes the proof.

Theorem 7.3.2. *If* $\mathbf{V}_h$ *and* S_h *satisfy the* inf-sup *condition, then*

$$\|\boldsymbol{u}-\boldsymbol{u}_h\|_1 + \|T-T_h\|_1 \leq C\left(\inf_{\boldsymbol{v}\in\mathbf{V}_h}\|\boldsymbol{u}-\boldsymbol{v}\|_1 + \inf_{q\in S_h}\|p-q\|_0 + \inf_{s\in W_h}\|T-s\|_1\right). \tag{7.3.9}$$

Furthermore, if $\boldsymbol{u} \in \mathbf{H}^3(\Omega)\cap\mathbf{H}_0^1(\Omega)$, $p \in H^2(\Omega)\cap L_0^2(\Omega)$, $T \in H^3(\Omega)$, *and* $\|\boldsymbol{u}\|_3 + \|p\|_2 + \|T\|_3 \leq C\|\mathbf{f}\|_1 + \|Q\|_1$, *then*

$$\|\boldsymbol{u}-\boldsymbol{u}_h\|_1 + \|T-T_h\|_1 \leq Ch^2\left(\|\mathbf{f}\|_1 + \|Q\|_1\right). \tag{7.3.10}$$

Proof. The first part is proven using the same techniques as in § 7.1 and § 7.2 for the Stokes and Navier Stokes equations, e.g. the *inf-sup* condition and

continuity of $b(\cdot,\cdot)$. The second part can be shown by choosing $\boldsymbol{v} = \pi_V \boldsymbol{u}$, $q = \pi_S p$ and $s = \pi_W T$, and applying standard interpolation results.

Theorem 7.3.3. *If* $\mathbf{V}_h$ *and* S_h *satisfy the* inf-sup *condition, then*

$$\|p - p_h\|_0 \leq C \left(\inf_{\boldsymbol{v} \in \mathbf{V}_h} \|\boldsymbol{u} - \boldsymbol{v}\|_1 + \inf_{q \in S_h} \|p - q\|_0 + \inf_{s \in W_h} \|T - s\|_1 \right). \quad (7.3.11)$$

Furthermore, if $\boldsymbol{u} \in \mathbf{H}^3(\Omega) \cap \mathbf{H}_0^1(\Omega)$, $p \in H^2(\Omega) \cap L_0^2(\Omega)$, *and* $T \in H^3(\Omega)$, *and* $\|\boldsymbol{u}\|_3 + \|p\|_2 + \|T\|_3 \leq C\|\mathbf{f}\|_1 + \|Q\|_1$, *then*

$$\|p - p_h\|_0 \leq Ch^2 \left(\|\mathbf{f}\|_1 + \|Q\|_1 \right). \quad (7.3.12)$$

Proof. Let $q \in S_h$ be arbitrary. We use the *inf-sup* condition, Galerkin orthogonality, and continuity of $b(\cdot,\cdot)$ to obtain

$$\begin{aligned}
\beta \|q - p_h\|_0 &\leq \sup_{\boldsymbol{v} \in \mathbf{V}_h} \frac{|b(\boldsymbol{v}, q - p_h)|}{\|\boldsymbol{v}\|_1} \\
&= \sup_{\boldsymbol{v} \in \mathbf{V}_h} \frac{|b(\boldsymbol{v}, p - p_h) + b(q - p)|}{\|\boldsymbol{v}\|_1} \\
&= \sup_{\boldsymbol{v} \in \mathbf{V}_h} \Big| \frac{-a(\boldsymbol{u} - \boldsymbol{u}_h, \boldsymbol{v}) - c(\boldsymbol{u}, \boldsymbol{u}, \boldsymbol{v}) + c(\boldsymbol{u}_h, \boldsymbol{u}_h, \boldsymbol{v})}{\|\boldsymbol{v}\|_1} \\
&\quad + \frac{d(T - T_h, \boldsymbol{v}) + b(\boldsymbol{v}, q - p)}{\|\boldsymbol{v}\|_1} \Big| \\
&\leq C \left(\|\boldsymbol{u} - \boldsymbol{u}_h\|_1 + \|p - q\|_0 + \|T - T_h\|_1 \right) \\
\|q - p_h\|_0 &\leq C \left(\inf_{\boldsymbol{v} \in \mathbf{V}_h} \|\boldsymbol{u} - \boldsymbol{v}\|_1 + \inf_{q \in S_h} \|p - q\|_0 + \inf_{s \in W_h} \|T - s\|_1 \right).
\end{aligned}$$

An application of the triangle inequality proves (7.3.11), and (7.3.12) follows by choosing $\boldsymbol{v}$, q, and s to be the interpolants.

Theorem 7.3.4. *Assume that the iterative method to solve (7.3.2) has converged so that the nonlinear remainder is negligible, and that the solution of the adjoint problem satisfy the regularity condition (7.3.13) with* $\boldsymbol{\psi} = (\boldsymbol{\psi}_{\boldsymbol{u}}, \psi_p, \psi_T)^T$. *Then,*

$$\|\boldsymbol{u} - \boldsymbol{u}_h\|_0 + \|T - T_h\|_0 \leq Ch \left(\|\boldsymbol{u} - \boldsymbol{u}_h\|_1 + \|p - p_h\|_0 + \|T - T_h\|_1 \right),$$

and if the assumptions of Theorems 7.3.2 and 7.3.3 hold

$$\|\boldsymbol{u}-\boldsymbol{u}_h\|_0+\|T-T_h\|_0 \leq Ch^3\left(\|\mathbf{f}\|_1+\|Q\|_1\right).$$

Proof. Let $\boldsymbol{\psi}_{\boldsymbol{u}}=\boldsymbol{u}-\boldsymbol{u}_h$ and $\psi_T=T-T_h$ in (7.2.8), multiply the system by $(e_u,e_p)^T=(u-U,p-P)^T$, apply the divergence theorem, and use the definition of the linearized adjoint to obtain

$$\begin{aligned}\|\boldsymbol{u}-\boldsymbol{u}_h\|_0^2+\|T-T_h\|_0^2 =& a_1(\boldsymbol{e}_{\boldsymbol{u}},\boldsymbol{\phi})+c_1(\boldsymbol{u},\boldsymbol{u},\boldsymbol{\phi})-c_1(\boldsymbol{u}_h,\boldsymbol{u}_h,\boldsymbol{\phi})\\ &+b(\boldsymbol{e}_{\boldsymbol{u}},z)+d(e_T,\boldsymbol{\phi})+b(\boldsymbol{\phi},e_p)\\ &+a_2(e_T,\theta)+c_2(\boldsymbol{u},T,\theta)-c_2(\boldsymbol{u}_h,T_h,\theta)\end{aligned}$$

Next, we apply orthogonality,

$$\begin{aligned}\|\boldsymbol{u}-\boldsymbol{u}_h\|_0^2+\|T-T_h\|_0^2 =& a_1(\boldsymbol{e}_{\boldsymbol{u}},\boldsymbol{\phi}-\pi_V\boldsymbol{\phi})+c_1(\boldsymbol{u},\boldsymbol{u},\boldsymbol{\phi}-\pi_V\boldsymbol{\phi})\\ &-c_1(\boldsymbol{u}_h,\boldsymbol{u}_h,\boldsymbol{\phi}-\pi_V\boldsymbol{\phi})+b(\boldsymbol{e}_{\boldsymbol{u}},z-\pi_Z z)\\ &+d(e_T,\boldsymbol{\phi}-\pi_V\boldsymbol{\phi})+b(\boldsymbol{\phi}-\pi_V\boldsymbol{\phi},e_p)\\ &+a_2(e_T,\theta-\pi_W\theta)+c_2(\boldsymbol{u},T,\theta-\pi_W\theta)\\ &-c_2(\boldsymbol{u}_h,T_h,\theta-\pi_W\theta)\end{aligned}$$

the continuity of bilinear and trilinear forms,

$$\begin{aligned}\|\boldsymbol{u}-\boldsymbol{u}_h\|_0^2+\|T-T_h\|_0^2 \leq& C\|\boldsymbol{e}_{\boldsymbol{u}}\|_1\cdot\|\boldsymbol{\phi}-\pi_V\boldsymbol{\phi}\|_1+C\|\boldsymbol{e}_{\boldsymbol{u}}\|_1\cdot\|\boldsymbol{\phi}-\pi_V\boldsymbol{\phi}\|_1\\ &+C\|\boldsymbol{e}_{\boldsymbol{u}}\|_1\cdot\|z-\pi_Z z\|_0+C\|e_T\|_1\cdot\|\boldsymbol{\phi}-\pi_V\boldsymbol{\phi}\|_1\\ &+C\|\boldsymbol{\phi}-\pi_V\boldsymbol{\phi}\|_1\cdot\|e_p\|_0+C\|e_T\|_1\cdot\|\theta-\pi_W\theta\|_1\\ &+C\left(\|\boldsymbol{e}_{\boldsymbol{u}}\|_1+\|e_T\|_1\right)\|\theta-\pi_W\theta\|_1\end{aligned}$$

an interpolation result,

$$\begin{aligned}\|\boldsymbol{u}-\boldsymbol{u}_h\|_0^2+\|T-T_h\|_0^2 \leq & Ch\|\boldsymbol{e_u}\|_1\cdot\|\boldsymbol{\phi}\|_2+Ch\|\boldsymbol{e_u}\|_1\cdot\|\boldsymbol{\phi}\|_2\\ &+Ch\|\boldsymbol{e_u}\|_1\cdot\|z\|_1+Ch\|e_T\|_1\cdot\|\boldsymbol{\phi}\|_1\\ &+Ch\|\boldsymbol{\phi}\|_2\cdot\|e_p\|_0+Ch\|e_T\|_1\cdot\|\theta\|_2\\ &+Ch\left(\|\boldsymbol{e_u}\|_1+\|e_T\|_1\right)\|\theta\|_2\end{aligned}$$

and finally the regularity estimate (7.3.13) to conclude

$$\|\boldsymbol{u}-\boldsymbol{u}_h\|_0+\|T-T_h\|_0 \leq Ch\left(\|\boldsymbol{e_u}\|_1+\|e_p\|_0+\|e_T\|_1\right),$$

which proves the first assertion. A direct application of Theorems 7.3.2 and 7.3.3 proves the second assertion.

Theorem 7.3.5. *Assume that the iterative method to solve (7.3.2) has converged so that the nonlinear remainder is negligible, and that the solution of the adjoint problem satisfy the regularity condition*

$$\|\boldsymbol{\phi}\|_2+\|z\|_1+\|\theta\|_2 \leq C\|\boldsymbol{\psi}\|_0, \tag{7.3.13}$$

where $\boldsymbol{\psi}=(\boldsymbol{\psi_u},\psi_p,\psi_T)^T$. *Then,*

$$\|\boldsymbol{u}-\boldsymbol{u}_h\|_0+\|T-T_h\|_0 \leq \sum_{K\in\tau_h} Ch_K^3\left(\|\mathbf{f}\|_{1,K}+\|Q\|_{1,K}\right)\cdot\left(|\boldsymbol{\phi}|_{2,K}+|z|_{1,K}+|\theta|_{2,K}\right). \tag{7.3.14}$$

Proof. We set

$$\boldsymbol{\psi}=\left(\frac{\boldsymbol{e_u}}{\|\boldsymbol{e_u}\|_0},0,\frac{e_T}{\|e_T\|_0}\right),$$

multiply the system by $(\boldsymbol{e_u},e_p,e_T)^T$, and integrate by parts, yielding

$$\begin{aligned}\|\boldsymbol{e_u}\|_0+\|e_T\|_0=&\sum_{K\in\tau_{F,h}}(R_1,\boldsymbol{\phi}-\pi_V\boldsymbol{\phi})_K+\frac{1}{2}\left([\mu\partial_n\boldsymbol{u}_h],\boldsymbol{\phi}-\pi_V\boldsymbol{\phi}\right)_{\partial K}\\ &+\sum_{K\in\tau_{F,h}}(R_2,z-\pi_Z z)_K\\ &+\sum_{K\in\tau_{F,h}}(R_3,\theta-\pi_W\theta_F)_K+\frac{1}{2}\left([k\partial_n T_h],\theta-\pi_W\theta\right)_{\partial K}\end{aligned}$$

with

$$R_1 = \mathbf{f} + \mu\Delta\boldsymbol{u}_h - \rho_0\left(\boldsymbol{u}_h \cdot \nabla\right)\boldsymbol{u}_h - \nabla p_h - \rho_0\beta T_h\mathbf{g}$$

$$R_2 = \nabla \cdot \boldsymbol{u}_h$$

$$R_3 = Q_F + k\Delta T_h - \rho_0 c_p \boldsymbol{u}_h \cdot \nabla T_h$$

First, we use the definition of $\boldsymbol{u}$, p, and T to rewrite the residuals

$$\begin{aligned} R_1 = & -\mu\Delta\left(\boldsymbol{u} - \boldsymbol{u}_h\right) + \rho_0\left(\boldsymbol{u} \cdot \nabla\right)\boldsymbol{u} - \rho_0\left(\boldsymbol{u}_h \cdot \nabla\right)\boldsymbol{u}_h + \nabla\left(p - p_h\right) \\ & + \rho_0\beta\left(T - T_h\right)\mathbf{g} \\ R_2 = & -\nabla \cdot \left(\boldsymbol{u} - \boldsymbol{u}_h\right) \\ R_3 = & -k\Delta\left(T - T_h\right) + \rho_0 c_p \boldsymbol{u} \cdot \nabla T - \rho_0 c_p \boldsymbol{u}_h \cdot \nabla T_h \end{aligned}$$

Now, we bound each term individually.

The first term is $(R_1, \phi - \pi_V\phi)_K$. We apply the Cauchy-Schwarz inequality

$$|(R_1, \phi - \pi_V\phi)_K| \leq \|R_1\|_K \cdot \|\phi - \pi_V\phi\|_K.$$

To bound the residual, we require the inverse estimate

$$\|\boldsymbol{u}-\boldsymbol{u}_h\|_{2,K}+\|p-p_h\|_{1,K}+\|T-T_h\|_{2,K} \leq Ch_K\left(\|\mathbf{f}\|_{1,K} + \|Q\|_{1,K}\right), \quad (7.3.15)$$

which assumes that the mesh is quasi-uniform. Each term in R_1 may be bounded using either the inverse estimate or the *a priori* error bound (7.3.2), giving

$$\|R_1\|_K \leq Ch_K\left(\|\mathbf{f}\|_{1,K} + \|Q\|_{1,K}\right).$$

The other component, $\|\phi - \pi_V\phi\|_K$, is easily bounded

$$\|\phi - \pi_V\phi\|_K \leq Ch_K^2|\phi|_{2,K},$$

using an interpolation result.

The next residual term is $(R_2, z - \pi_Z z)_K$. We apply the Cauchy-Schwarz inequality,

$$|(R_2, z - \pi_Z z)_K| \leq C\|R_2\|_K \cdot \|z - \pi_Z z\|_K,$$

where the *a priori* error bound (7.3.2) gives

$$\|R_2\|_K \leq Ch_K^2 \left(\|\mathbf{f}\|_{1,K} + \|Q\|_{1,K}\right),$$

and an interpolation result gives

$$\|z - \pi_Z z\|_K \leq Ch_K \|z\|_{1,K}.$$

Combining these results, we have

$$|(R_2, z - \pi_Z z)_K| \leq Ch_K^3 \left(\|\mathbf{f}\|_{1,K} + \|Q\|_{1,K}\right) \cdot \|z\|_{1,K}.$$

The third residual term is $(R_3, \theta - \pi_W \theta)_K$. We apply the Cauchy-Schwarz inequality,

$$|(R_3, \theta - \pi_W \theta)_K| \leq C\|R_3\|_K \cdot \|\theta - \pi_W \theta\|_K.$$

Each of the terms in the residual may be bounded with either the inverse estimate (7.3.15) or the *a priori* error bound (7.3.2)giving

$$\|R_3\|_K \leq Ch_K \left(\|\mathbf{f}\|_{1,K} + \|Q\|_{1,K}\right),$$

and an interpolation result gives

$$\|\theta - \pi_W \theta\|_K \leq Ch_K^2 \|\theta\|_{2,K}.$$

Combining these results, we have

$$|(R_3, \theta - \pi_W \theta)_K| \leq Ch_K^3 \left(\|\mathbf{f}\|_{1,K} + \|Q\|_{1,K}\right) \cdot \|\theta\|_{2,K}.$$

The remaining terms represent jump terms over element edges. We provide the details for $\frac{1}{2}\left([\mu\partial_n \boldsymbol{u}_h], \boldsymbol{\phi} - \pi_V\boldsymbol{\phi}\right)_{\partial K}$ and note that the other term, $\frac{1}{2}\left([k\partial_n T_h], \theta - \pi_W\theta\right)_{\partial K}$ may be handled similarly. We add and subtract $(\mu\partial_n \boldsymbol{u}, \boldsymbol{\phi} - \pi_V\boldsymbol{\phi})_{\partial K}$ to give

$$\frac{1}{2}\left([\mu\partial_n \boldsymbol{u}_h], \boldsymbol{\phi} - \pi_V\boldsymbol{\phi}\right)_{\partial K} = \frac{1}{2}\left(\mu\partial_n \boldsymbol{u} - \mu\partial_n \boldsymbol{u}_{h,K}, \boldsymbol{\phi} - \pi_V\boldsymbol{\phi}\right)_{\partial K} + \frac{1}{2}\left(\mu\partial_n \boldsymbol{u}_{h,K'} - \mu\partial_n \boldsymbol{u}, \boldsymbol{\phi} - \pi_V\boldsymbol{\phi}\right)_{\partial K},$$

where $\boldsymbol{u}_{h,K}$ represents the approximation on element K and $\boldsymbol{u}_{h,K'}$ represents the approximation on a neighboring element K'. Next, we use the trace inequality

$$\|v\|_{\partial K} \leq C\|v\|_K^{1/2} \cdot \|v\|_{1,K}^{1/2} \tag{7.3.16}$$

along with the *a priori* bound (7.3.2) and an interpolation result to conclude

$$\left|\frac{1}{2}\left([\mu\partial_n \boldsymbol{u}_h], \boldsymbol{\phi} - \pi_V\boldsymbol{\phi}\right)_{\partial K}\right| \leq \frac{1}{2}Ch_K^3\left(\|\mathbf{f}\|_{1,K} + \|Q\|_{1,K}\right) \cdot |\boldsymbol{\phi}|_{2,K} + \frac{1}{2}Ch_{K'}^3\left(\|\mathbf{f}\|_{1,K'} + \|Q\|_{1,K'}\right) \cdot |\boldsymbol{\phi}|_{2,K'}$$

When we sum over all elements, each edge is counted twice, which removes the factor of $1/2$.

Chapter 8

INTERFACE TRANSFER IN FLUID/STRUCTURE INTERACTION PROBLEMS

8.1 Introduction

In this chapter, we consider an operator decomposition approach for the solution of a (conjugate) heat transfer problem between a heat conducting fluid and a solid. We assume that we have one method for solving the temperature field in the solid and another for determining the velocity and temperature field in the fluid, and we are required to ensure continuity of temperature and heat flux across their common boundary. Obtaining a solution of the fully coupled system by this operator decomposition approach proceeds via a (nominally) infinite fixed point iteration during with the current solution in one domain provides boundary conditions for the new solution in the other domain. We model the temperature field in the solid using the simple heat equation and apply the Boussinesq approximation within the fluid, and consider situations where there are only a small number of interfaces between fluid and solid domains.

Our goal is to ensure that the error in a given a functional of the temperature field satisfies a user-specified tolerance. We perform an *a posteriori* error analysis of a finite-element implementation of the operator decomposition technique and obtain error estimates that can be used to guide an adaptive discretization strategy and show how the operator decomposition causes a loss in the approximation order of convergence. Our approach is based on the standard techniques using variational analysis, residuals and the generalized Green's function solution to an adjoint problem [10, 18, 30, 31, 28, 42], which we modify to account for the fact that numerical errors in the solution of each component are propagated to the other component through the boundary conditions and from one step of the iterative procedure to the next. Both effects are characteristic of operator

decomposition discretizations, e.g. [21, 38], and require extensions of the usual *a posteriori* analysis techniques.

The "boundary element flux" technique developed by Wheeler [55] and Carey [34, 20] provides an efficient way to compute a more accurate estimate of the normal derivatives on a boundary. We show how that this technique can be used to improve the accuracy of the operator decomposition method, and more subtly, how the use of this modified flux restores the order of convergence that is lost due to the operator decomposition.

In §8.2, we consider the flow of a hot fluid past a cylinder and construct the general conjugate heat transfer problem. We describe the iterative operator decomposition finite element method in §8.3. In §8.4 we present an example which demonstrates the loss of order of convergence due to operator decomposition in order to motivate the consideration of the boundary flux method in §8.5. We perform an *a posteriori* error analysis in §8.6 and describe on adaptive mesh refinement strategy based on the error estimates we obtain. We provide numerical examples in §8.7 and demonstrate the optimal convergence rate obtained using the "boundary element flux". In §8.8, we carry out an analysis which identifies the transferred gradient information as being responsible for the loss of order and show that using the boundary flux method to compute the gradient information restores the order of convergence.

8.2 Flow of a hot fluid past a cylinder

We consider the steady flow of a heat conducting viscous Newtonian fluid past a solid cylinder as shown in Fig. 8.1.

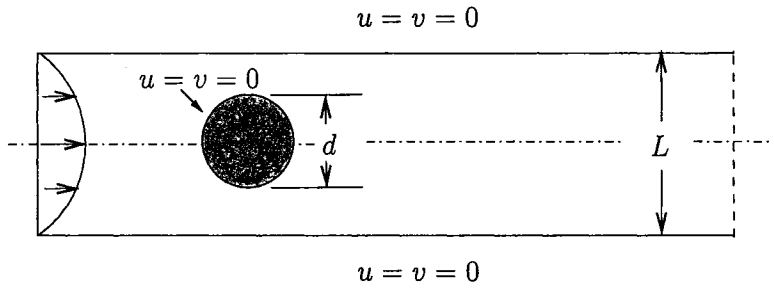

Figure 8.1: Computational domain for flow past a cylinder

Our primary interest is in the heat transfer between fluid and solid. We solve the heat equation in the solid. In the fluid, we use the Boussinesq approximation to solve the conservation of momentum, mass and heat equations. The temperature field is advected by the fluid and couples back to the momentum equations via the buoyancy term. Our approach is to estimate and drive adaptivity to minimize the error in a given functional of the solution such as the temperature in a given region, or the shear stress on part of a boundary. This approach is therefore similar to that of Giles, et. al [37, 35].

8.2.1 The general conjugate heat transfer problem

Let Ω_1 and Ω_2 be polygonal domains in $\mathbb{R}^2$ with boundaries $\partial\Omega_1$ and $\partial\Omega_2$ intersecting along an interface $\Gamma_I = \partial\Omega_1 \cap \partial\Omega_2$. We consider fluid/structure interface problems of the form

$$\begin{cases} -\mu\Delta\mathbf{u} + \rho_0 (\mathbf{u}\cdot\nabla)\,\mathbf{u} + \nabla p = \mathbf{g}\rho_0 \left(1 - \beta(T_F - T_{\text{ref}})\right), & x \in \Omega_F, \\ -\nabla\cdot\mathbf{u} = 0, & x \in \Omega_F, \\ -k_F\Delta T_F + \rho_0 c_p \mathbf{u}\cdot\nabla T_F = Q_F, & x \in \Omega_F, \\ \begin{cases} T_S = T_F, \\ k_F \dfrac{\partial T_F}{\partial \boldsymbol{n}} = k_S \dfrac{\partial T_S}{\partial \boldsymbol{n}}, \end{cases} & x \in \Gamma_I, \\ -k_S\Delta T_S = Q_S, & x \in \Omega_S. \end{cases} \tag{8.2.1}$$

where ∂_n represents the unit normal derivative into the fluid, μ denotes the viscocity, k_F and k_S the thermal conductivities of the fluid and solid, c_p the specific heat, β the coefficient of thermal expansion, and ρ_0 and T_{ref} are reference values for the density and temperature respectively.

To simplify the notation, we define $\Gamma_{\boldsymbol{u},D}$ and $\Gamma_{\boldsymbol{u},N}$ to be the sets of boundaries where we pose Dirichlet conditions and Neumann conditions respectively for the fluid velocities, and set

$$\begin{cases} \boldsymbol{u} = \boldsymbol{g}_{\boldsymbol{u},D}, & x \in \Gamma_{\boldsymbol{u},D}, \\ \mu\, \partial \boldsymbol{u}/\partial \boldsymbol{n} = \boldsymbol{g}_{\boldsymbol{u},N}, & x \in \Gamma_{\boldsymbol{u},N}. \end{cases}$$

Similarly, we define $\Gamma_{T_F,D}$, $\Gamma_{T_F,N}$, $\Gamma_{T_S,D}$, and $\Gamma_{T_S,N}$ to be the sets of boundaries where we pose Dirichlet and Neumann conditions for the temperature fields in the fluid and the solid respectively, and set

$$\begin{cases} T_F = g_{T_F,D}, & x \in \Gamma_{T_F,D}, \\ k_F\, \partial T_F/\partial \boldsymbol{n} = g_{T_F,N}, & x \in \Gamma_{T_F,N}, \\ T_S = g_{T_S,D}, & x \in \Gamma_{T_S,D}, \\ k_S\, \partial T_S/\partial \boldsymbol{n} = g_{T_S,N}, & x \in \Gamma_{T_S,N}. \end{cases}$$

To simplify the discussion, we assume that these boundary conditions can be interpolated exactly in the finite element space, meaning that errors due to interpolating the Dirichlet boundary conditions are zero.

8.2.2 Weak formulation

We let $L^2(\Omega)$ denote the space of square integrable functions on Ω with inner product $(\cdot,\cdot)_\Omega$ and norm $\|\cdot\|_\Omega$, or simply $(\cdot,\cdot)$ when the domain is clear. We use $H^s(\Omega)$ to denote the Sobolev space with real index s associated with the norm $\|\cdot\|_{\Omega,s}$ and seminorm $|\cdot|_{\Omega,s}$ [1, 15] with the obvious generalization to vector valued functions.

The weak formulation of (8.2.1) seeks $\boldsymbol{u} \in \mathbf{V}_F$, $p \in L_0^2(\Omega_F)$, $T_F \in W_F$ and $T_S \in W_S$ such that

$$\begin{cases} a_1(\boldsymbol{u}, \boldsymbol{v}) + c_1(\boldsymbol{u}, \boldsymbol{u}, \boldsymbol{v}) + b(\boldsymbol{v}, p) + d(T_F, \boldsymbol{v}) = (\mathbf{f}, \boldsymbol{v}), \\ b(\boldsymbol{u}, q) = 0, \\ a_2(T_F, w_F) + c_2(\boldsymbol{u}, T_F, w_F) + a_3(T_S, w_S) = (Q_F, w_F) + (Q_S, w_S), \end{cases} \tag{8.2.2}$$

for all $v \in \mathbf{V}_{F,0}$, $q \in L_0^2(\Omega_F)$, $w_F \in W_{F,0}$ and $w_S \in W_{S,0}$, where

$$\begin{aligned} \mathbf{f} &= \rho_0 \left(1 + \beta T_{\text{ref}}\right) \mathbf{g} \\ a_1(\boldsymbol{u}, \boldsymbol{v}) &= \int_{\Omega_F} \mu \nabla \boldsymbol{u} : \nabla \boldsymbol{v} \, dx \\ a_2(T_F, w_F) &= \int_{\Omega_F} k_F \nabla T_F \cdot \nabla w_F \, dx \\ a_3(T_S, w_S) &= \int_{\Omega_S} \boldsymbol{k}_S \nabla T_S \cdot \nabla w_S \, dx \\ b(\boldsymbol{v}, q) &= -\int_{\Omega_F} (\nabla \cdot \boldsymbol{v}) q \, dx \\ c_1(\boldsymbol{u}, \boldsymbol{v}, \mathbf{z}) &= \int_{\Omega_F} \rho_0 \left(\boldsymbol{u} \cdot \nabla\right) \boldsymbol{v} \cdot \mathbf{z} \, dx \\ c_2(\boldsymbol{u}, T, w) &= \int_{\Omega_F} \rho_0 c_p \boldsymbol{u} \cdot \nabla T \, w \, dx \\ d(T, \boldsymbol{v}) &= \int_{\Omega_F} \rho_0 \beta T \mathbf{g} \cdot \boldsymbol{v} \, dx. \end{aligned}$$

Here

$$\begin{aligned} \mathbf{V}_F &= \left\{ \boldsymbol{v} \in \mathbf{H}^1(\Omega_F) \mid \boldsymbol{v} = \boldsymbol{g}_{\boldsymbol{u},D} \text{ on } \Gamma_{\boldsymbol{u},D} \right\}, \\ W_F &= \left\{ w \in H^1(\Omega_F) \mid w = g_{T_F,D} \text{ on } \Gamma_{T_F,D} \right\}, \\ W_S &= \left\{ w \in H^1(\Omega_S) \mid w = g_{T_S,D} \text{ on } \Gamma_{T_S,D} \right\}, \end{aligned}$$

and

$$\begin{aligned}
\mathbf{V}_{F,0} &= \{\boldsymbol{v} \in V_F \mid \boldsymbol{v} = \mathbf{0} \text{ on } \Gamma_{\boldsymbol{u},D}\}, \\
W_{F,0} &= \{w \in W_F \mid w = 0 \text{ on } \Gamma_{T_F,D}\}, \\
W_{S,0} &= \{w \in W_S \mid w = 0 \text{ on } \Gamma_{T_S,D}\}, \\
L_0^2(\Omega) &= \left\{v \in L^2(\Omega) \mid \int_\Omega v \, dx = 0\right\}.
\end{aligned}$$

We assume that the source terms and the boundary data are sufficiently small and the viscocity μ and thermal conductivities k_F and k_S are sufficiently large so that (6.2.2) admits a regular weak solution.

8.3 An iterative operator decomposition method

Assume that we have an initial guess for the Dirichlet data along the interface between fluid and solid domains, $T_F^{\{0\}}$. To compute a numerical solution of (8.2.1) we construct the following iterative operator decomposition method.

Iterative Operator Decomposition Method

k = 0

while $(\|T_F^{\{k\}} - T_F^{\{k-1\}}\|_{\Gamma_I} > TOL)$ **do**

(a) k = k+1

(b) Given $T_F^{\{k-1\}}$ on Γ_I, compute $T_S^{\{k\}} \in \Omega_S$ by solving

$$\begin{cases} -k_S \Delta T_S^{\{k\}} = Q_S, & \mathbf{x} \in \Omega_S, \\ T_S^{\{k\}} = T_F^{\{k-1\}}, & \mathbf{x} \in \Gamma_I, \end{cases} \tag{8.3.1}$$

(c) Given $T_S^{\{k\}}$, compute $\boldsymbol{u}^{\{k\}}, p^{\{k\}}, T_F^{\{k\}} \in \Omega_F$ by solving

$$\begin{cases} -\mu \Delta \boldsymbol{u}^{\{k\}} + \rho_0 \left(\boldsymbol{u}^{\{k\}} \cdot \nabla\right) \boldsymbol{u}^{\{k\}} + \nabla p^{\{k\}} + \rho_0 \beta T_F^{\{k\}} \boldsymbol{g} = \mathbf{f}, & \boldsymbol{x} \in \Omega_F, \\ -\nabla \cdot \boldsymbol{u}^{\{k\}} = 0, & x \in \Omega_F, \\ -k_F \Delta T_F^{\{k\}} + \rho_0 c_p \boldsymbol{u}^{\{k\}} \cdot \nabla T_F^{\{k\}} = Q_F, & \boldsymbol{x} \in \Omega_F, \\ k_F \dfrac{\partial T_F^{\{k\}}}{\partial \boldsymbol{n}} = k_S \dfrac{\partial T_S^{\{k\}}}{\partial \boldsymbol{n}}, & \boldsymbol{x} \in \Gamma_I. \end{cases} \tag{8.3.2}$$

and subject to the appropriate boundary conditions on the velocity, pressure and temperature fields away from the interface.

end while

8.3.1 Finite element discretization

Let $\tau_{F,h}$ and $\tau_{S,h}$ be locally quasi-uniform triangulations of Ω_F and Ω_S respectively. We do not assume that the triangulations on either side of Γ_I are aligned.

We use the piecewise polynomial spaces

$$\begin{aligned}
\mathbf{V}_F^h &= \left\{v \in \mathbf{V}_F \mid \boldsymbol{v} \text{ continuous on } \Omega_F,\ v_i \in P^2(K) \text{ for all } K \in \tau_{F,h}\right\},\\
Z^h &= \left\{z \in Z \mid z \text{ continuous on } \Omega_F,\ z \in P^1(K) \text{ for all } K \in \tau_{F,h}\right\},\\
W_F^h &= \left\{w \in W_F \mid w \text{ continuous on } \Omega_F,\ w \in P^2(K) \text{ for all } K \in \tau_{F,h}\right\},\\
W_S^h &= \left\{w \in W_S \mid w \text{ continuous on } \Omega_S,\ w \in P^2(K) \text{ for all } K \in \tau_{S,h}\right\},
\end{aligned}$$

and the associated subspaces

$$\begin{aligned}
\mathbf{V}_{F,0}^h &= \left\{\boldsymbol{v} \in \mathbf{V}^h \mid \boldsymbol{v} = 0 \text{ on } \Gamma_{\boldsymbol{u},D}\right\},\\
W_{F,0}^h &= \left\{w \in W_F^h \mid w = 0 \text{ on } \Gamma_{T_F,D}\right\},\\
W_{S,0}^h &= \left\{w \in W_S^h \mid w = 0 \text{ on } \Gamma_{T_S,D}\right\}.
\end{aligned}$$

where $P^q(K)$ denote the space of polynomials of degree q on an element K by $P^q(K)$. We let $\pi_{\mathbf{V}}$, π_{W_F}, π_{W_S}, and π_Z be projections into $\mathbf{V}_F^h$, W_F^h, W_S^h and Z^h respectively. We also use π_{W_F} and π_{W_S} to denote projections into W_F^h and W_S^h respectively along the interface Γ_I.

Operator Decomposition Finite Element Method: OD-FEM

k = 0

while $(\|T_F^{\{k\}} - T_F^{\{k-1\}}\|_{\Gamma_I} > TOL)$ **do**

(a) k = k+1

(b) Find $T_{S,h}^{\{k\}} \in W_S^h$ such that $T_{S,h}^{\{k\}} = \pi_1 T_{F,h}^{\{k-1\}}$ along the interface Γ_I and

$$a_3(T_{S,h}^{\{k\}}, w) = (Q_S, w), \quad \forall w \in W_{S,0}^h, \tag{8.3.3}$$

(c) Find $\boldsymbol{u}_h^{\{k\}} \in \mathbf{V}_F^h$, $p_h^{\{k\}} \in Z^h$, and $T_{F,h}^{\{k\}} \in W_F^h$ such that

$$\begin{cases} a_1(\boldsymbol{u}_h^{\{k\}}, \boldsymbol{v}) + c_1(\boldsymbol{u}_h^{\{k\}}, \boldsymbol{u}_h^{\{k\}}, \boldsymbol{v}) + b(\boldsymbol{v}, p_h^{\{k\}}) + d(T_{F,h}^{\{k\}}, \boldsymbol{v}) = (\mathbf{f}, \boldsymbol{v}), \\ b(\boldsymbol{u}_h^{\{k\}}, q) = 0, \\ a_2(T_{F,h}^{\{k\}}, w) + c_2(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}}, w) = (Q_F, w) - (\chi^{\{k\}}, w)_{\Gamma_I}, \\ \qquad\qquad\qquad\qquad\qquad\qquad \text{where } \chi^{(k)} = k_S \partial_n T_{S,h}^{\{k\}}, \end{cases} \tag{8.3.4}$$

$\forall \boldsymbol{v} \in \mathbf{V}_{F,0}^h$, $q \in Z^h$, and $w \in W_{F,0}^h$.

end while

We have chosen V_F^h and Z^h to be the Taylor-Hood finite element pair which are known to satisfy the discrete *inf-sup* condition

$$\inf_{q \in Z^h} \sup_{\boldsymbol{v} \in \mathbf{V}_F^h} \frac{b(\boldsymbol{v}, q)}{\|\boldsymbol{v}\|_1 \cdot \|q\|_0} \geq \beta > 0. \tag{8.3.5}$$

8.3.2 Relaxation schemes

Unfortunately, the simple iterative scheme OD-FEM may not converge. In particular, the convergence depends on the values of A_1 and A_2 along the interface and the geometry of each region [37, 52, 58]. As an alternative, we consider two "relaxed" iteration schemes.

(1) *Relaxed Dirichlet values*

We choose $\alpha \in [0, 1]$ and update the Dirichlet interface values with

$$T_{S,h}^{\{k\}} = \alpha T_{S,h}^{\{k-1\}} + (1 - \alpha)\pi_{T_S} T_{F,h}^{\{k-1\}}. \tag{8.3.6}$$

before solving equation (8.3.3). Optimal values of α can be found in [52, 58], but $\alpha = 1/2$ works well in most situations.

(2) *Relaxed Neumann values*

We chose $\beta \in [0, 1]$, set

$$N_\beta^{\{k\}} = \beta k_F \partial_n T_{F,h}^{\{k-1\}} + (1-\beta) k_S \partial_n T_{S,h}^{\{k\}},$$

and replace $\chi^{\{k\}} = N_\beta^{\{k\}}$ in the rhs of the temperature equation in equation (8.3.4), i.e.,

$$a_2(T_{F,h}^{\{k\}}, w) + c_2(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}}, w) = (Q_F, w) - (N_\beta^{\{k\}}, w)_{\Gamma_I}. \qquad (8.3.7)$$

A proper choice of β reduces the number of iterations.

8.4 Motivational example illustrating loss of order

We apply the OD-FEM to the steady flow of a viscous Newtonian fluid in a channel connected along one boundary to a solid which is heated from below as shown in Fig. 8.2.

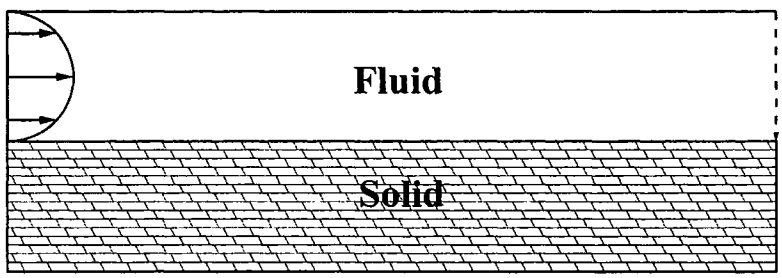

Figure 8.2: Computational domain for motivational example.

The Reynolds number (based on the channel width and the flux averaged inlet velocity) is $Re = 2.5$, and thermal conductivities,

$$k_F = 0.9 \text{ and } k_S = 1 + 0.5 \sin(2\pi x) \sin(2\pi y),$$

were chosen so that the solution is smooth, but nontrivial. The temperature fields are given in Fig. 8.3.

We solved the problem iteratively by passing the finite element flux, $k_S \nabla T_{S,h}^{\{k\}} \cdot \boldsymbol{n}$ in equation (8.3.4). We also compute a reference solution with a higher order method for comparison. In Fig. 8.4 we compare the L^2 errors in the temperature fields over $\Omega_S \cup \Omega_F$ on a series of meshes which align along the interface [51].

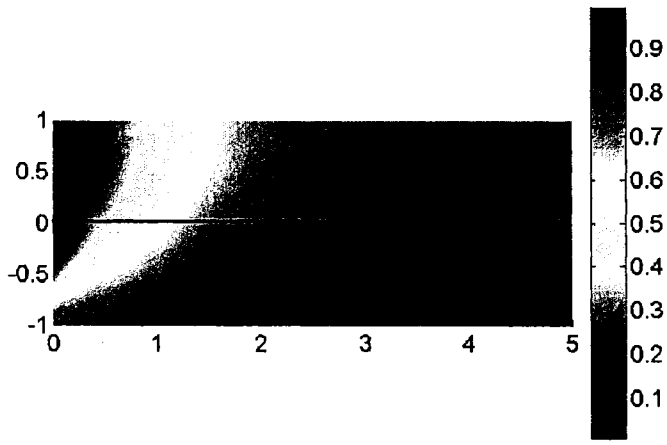

Figure 8.3: Temperature fields within the fluid and the solid.

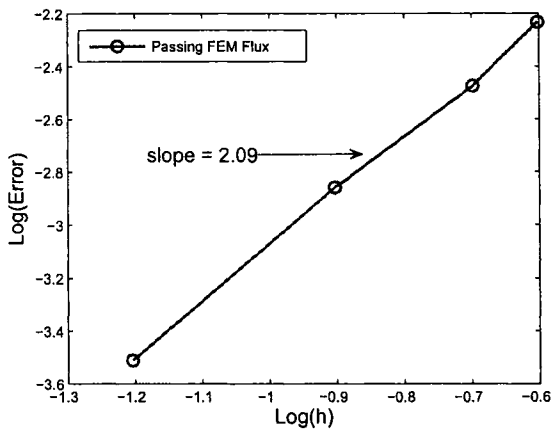

Figure 8.4: Comparison of the mesh size, h, versus the L^2 error in the temperature field when the finite element flux is passed.

We see that the approximation using the finite element flux converges quadratically with respect to the mesh size, rather than the expected cubic rate. This loss of order must be a result of the operator decomposition technique. The most likely reason is the use of the normal derivative of the temperature field that is based on the computed temperature field in the solid and applied as a boundary condition to the computation in the fluid domain. This normal derivative is known to be less accurate than the values of the temperature field itself. We examine a well known procedure for computing a more accurate estimate for the normal flux in the following §8.5.

8.5 Flux Correction

The numerical example in the previous suggestion suggests that the operator decomposition technique results in a loss of order of convergence. In an attempt to mitigate this effect, we use a post-processing technique

developed by Wheeler [55] and Carey [34, 20] to compute more accurate boundary flux information.

We define the set of elements in $\tau_{S,h}$ that intersect the boundary by

$$\tau_{S,h}^{\Gamma_I} = \{K \in \tau_{S,h} \mid \overline{K} \cap \partial\Omega \neq \emptyset\},$$

and the corresponding space

$$\Sigma_h = \{v \in P^2(K) \text{ with } K \in \tau_{S,h}^{\Gamma_I},\ v(\eta_i) = 0 \text{ if } \eta_i \notin \partial\Omega\},$$

where $\{\eta_i\}$ denotes the nodes of element K, so the degrees of freedom correspond to the nodes on the boundary. We seek $\sigma^{\{k\}} \in \Sigma_h$ satisfying

$$-(\sigma^{\{k\}}, v)_{\Gamma_I} = (Q_S, v)_{\Omega_1} - a_3(T_{S,h}^{\{k\}}, v), \quad \text{for all } v \in \Sigma_h, \tag{8.5.1}$$

where $T_{S,h}^{\{k\}}$ solves (8.3.3). Green's identity implies that $\sigma^{\{k\}}$ gives an approximation to the normal flux on the boundary which is relatively inexpensive to compute. This provides a third relaxation strategy.

(3) *Use of the recovered flux*

We set $\chi^{(k)} = \sigma^{\{k\}}$ in the rhs of the temperature equation in equation (8.3.4) , i.e.,

$$a_2(T_{F,h}^{\{k\}}, w) + c_2(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}}, w) = (Q_F, w) - (\sigma^{\{k\}}, w)_{\Gamma_I}. \tag{8.5.2}$$

In general, the accuracy of the flux approximation depends on the regularity of an associated Green's function [35, 56], but in many cases this post-processed flux is $\mathcal{O}(h^3)$ rather than the standard $\mathcal{O}(h^2)$ for the normal flux of a piecewise linear finite element approximation. In fact, as we will show in §8.8 the improvement that arises by using the recovered flux is due to a fortunate cancellation of errors rather than the precise details

of the accuracy of the recovered flux. Consistent with this observation, we had only mixed success with other methods of increasing the accuracy of the normal derivative such as the Zienkiewicz-Zhu (ZZ) patch recovery technique [60], the polynomial preserving recovery (PPR) method [49], and using higher order polynomials near the interface [9, 42].

8.6 *A posteriori* error analysis of OD-FEM

To estimate the error of the operator decomposition finite element approximation, we apply *a posteriori* techniques based on variational analysis and the adjoint problem.

8.6.1 The adjoint to the heat equation

Consider the linear elliptic boundary value problem,

$$\begin{cases} Lu = f, & x \in \Omega, \\ u = 0, & x \in \partial\Omega, \end{cases} \tag{8.6.1}$$

where $Lu = -k\Delta u + \mathbf{b} \cdot \nabla u + cu$. Let u_h be an approximation to u and define $e = u - u_h$. We define the formal adjoint operator, L^*, such that

$$(Le, \phi) = (e, L^*\phi).$$

When L is a differential operator, this involves the application of the divergence theorem to transfer derivatives to the test function ϕ. This usually involves additional terms along the boundary. We define the adjoint boundary conditions such that these terms vanish. Thus, the formal adjoint boundary value problem to (8.6.1) is

$$\begin{cases} L^*\phi = \psi, & x \in \Omega, \\ \phi = 0, & x \in \partial\Omega, \end{cases} \tag{8.6.2}$$

where $L^*\phi = -k\Delta\phi - \mathbf{b} \cdot \nabla\phi + (c - \nabla \cdot \mathbf{b})\phi$.

The adjoint to the heat equation within the solid is the special case where $\mathbf{b}$ and c are zero.

8.6.2 The adjoint to the Boussinesq equations

Let $B(\boldsymbol{u}, p, T)$ denote the Boussinesq operator. Defining the adjoint to the Boussinesq operator is complicated by the fact that it is a nonlinear operator. Let $\boldsymbol{u}_h$, p_h and T_h be approximation to $\boldsymbol{u}$, p, and T respectively and define $\mathbf{e} = (\boldsymbol{u} - \boldsymbol{u}_h, p - p_h, T - T_h)$. Formally, we define the linearized adjoint operator, $\overline{B}^*$, such that

$$(B(\boldsymbol{u}, p, T), \boldsymbol{\phi}) - (B(\boldsymbol{u}_h, p_h, T_h), \boldsymbol{\phi}) = (\overline{B}(\mathbf{e}), \boldsymbol{\phi}) = (\mathbf{e}, \overline{B}^*(\boldsymbol{\phi}))$$

where the linearized operator, $\overline{B}$, is defined by

$$\overline{B}(\boldsymbol{e}) = \int_0^1 \mathbf{B}'(s\boldsymbol{u} + (1-s)\boldsymbol{u}_h, sp + (1-s)p_h, sT + (1-s)T_h) \cdot \mathbf{e}\ ds,$$

with

$$\mathbf{B}'(\boldsymbol{u}, p, T) = \left(\frac{\partial B}{\partial \boldsymbol{u}}(\boldsymbol{u}, p, T), \frac{\partial B}{\partial p}(\boldsymbol{u}, p, T), \frac{\partial B}{\partial T}(\boldsymbol{u}, p, T)\right)^T.$$

First, we define the linearized forms. Let $c_1(\boldsymbol{v}, \boldsymbol{v}) = \rho_0(\boldsymbol{v} \cdot \nabla)\boldsymbol{v}$. The Gateaux derivative in the direction δ is given by

$$\begin{aligned} c_1'(\boldsymbol{v}, \boldsymbol{v})\delta &= \lim_{\epsilon \to 0} \frac{c_1(\boldsymbol{v} + \epsilon\delta, \boldsymbol{v} + \epsilon\delta) - c_1(\boldsymbol{v}, \boldsymbol{v})}{\epsilon} \\ &= \frac{\rho_0\left((\boldsymbol{v} + \epsilon\delta) \cdot \nabla\right)(\boldsymbol{v} + \epsilon\delta) - \rho_0(\boldsymbol{v} \cdot \nabla)\boldsymbol{v}}{\epsilon} \\ &= \rho_0(\delta \cdot \nabla)\boldsymbol{v} + \rho_0(\boldsymbol{v} \cdot \nabla)\delta \end{aligned}$$

We define the linearized form, $\bar{c}_1$ such that

$$\begin{aligned} \bar{c}_1(\boldsymbol{e}_{\boldsymbol{u}}) &= \int_0^1 (\rho_0(\boldsymbol{e}_{\boldsymbol{u}} \cdot \nabla)(\boldsymbol{u}s + \boldsymbol{U}(1-s)) + \rho_0((\boldsymbol{u}s + \boldsymbol{U}(1-s)) \cdot \nabla)\boldsymbol{e}_{\boldsymbol{u}})\ ds \\ &= \rho_0(\boldsymbol{e}_{\boldsymbol{u}} \cdot \nabla)\left(\frac{1}{2}\boldsymbol{u} + \frac{1}{2}\boldsymbol{U}\right) + \rho_0\left(\left(\frac{1}{2}\boldsymbol{u} + \frac{1}{2}\boldsymbol{U}\right) \cdot \nabla\right)\boldsymbol{e}_{\boldsymbol{u}} \end{aligned}$$

The adjoint linearized operator, $\bar{c}_1^*(\boldsymbol{\phi})$, is defined such that $(\bar{c}_1(\boldsymbol{e}_{\boldsymbol{u}}), \boldsymbol{\phi}) = (\boldsymbol{e}_{\boldsymbol{u}}, \bar{c}_1^*(\boldsymbol{\phi}))$. Using the divergence theorem, we obtain

$$\bar{c}_1^*(\boldsymbol{\phi}) = \rho_0 \nabla\left(\frac{1}{2}\boldsymbol{u} + \frac{1}{2}\boldsymbol{u}_h\right) \cdot \boldsymbol{\phi} - \rho_0\left(\frac{1}{2}\boldsymbol{u} + \frac{1}{2}\boldsymbol{u}_h\right) \cdot \nabla\boldsymbol{\phi} - \rho_0\left(\nabla \cdot \left(\frac{1}{2}\boldsymbol{u} + \frac{1}{2}\boldsymbol{u}_h\right)\right)\boldsymbol{\phi}.$$

Similarly, define $c_2(\boldsymbol{v}, w) = \rho_0 c_p(\boldsymbol{v} \cdot \nabla)w$. The partial Gateaux derivative in the direction δ is given by

$$\begin{aligned}\frac{\partial c_2(\boldsymbol{v}, w)}{\partial \boldsymbol{v}}\delta &= \lim_{\epsilon \to 0} \frac{c_2(\boldsymbol{v} + \epsilon\delta, w) - c_2(\boldsymbol{v}, w)}{\epsilon} \\ &= \frac{\rho_0 c_p \left((\boldsymbol{v} + \epsilon\delta) \cdot \nabla\right)(w) - \rho_0 c_p(\boldsymbol{v} \cdot \nabla)w}{\epsilon} \\ &= \rho_0 c_p(\delta \cdot \nabla)w\end{aligned}$$

We define the linearized form, $\bar{c}_{2\boldsymbol{u}}$ such that

$$\begin{aligned}\bar{c}_{2\boldsymbol{u}}(\boldsymbol{e_u}) &= \int_0^1 \left(\rho_0 c_p(\boldsymbol{e_u} \cdot \nabla)\left(T_F s + T_{F,h}(1-s)\right)\right) ds \\ &= \rho_0 c_p(\boldsymbol{e_u} \cdot \nabla)\left(\frac{1}{2}T_F + \frac{1}{2}T_{F,h}\right).\end{aligned}$$

The partial Gateaux derivative in the direction γ is given by

$$\begin{aligned}\frac{\partial c_2(\boldsymbol{v}, w)}{\partial w}\gamma &= \lim_{\epsilon \to 0} \frac{c_2(\boldsymbol{v}, w + \epsilon\gamma) - c_2(\boldsymbol{v}, w)}{\epsilon} \\ &= \frac{\rho_0 c_p \left((\boldsymbol{v}) \cdot \nabla\right)(w + \epsilon\gamma) - \rho_0 c_p(\boldsymbol{v} \cdot \nabla)w}{\epsilon} \\ &= \rho_0 c_p(\boldsymbol{v} \cdot \nabla)\gamma.\end{aligned}$$

We define the linearized form, $\bar{c}_{2T}$ such that

$$\begin{aligned}\bar{c}_{2T}(e_{T_F}) &= \int_0^1 \left(\rho_0 c_p((s\boldsymbol{u} + (1-s)\boldsymbol{U}) \cdot \nabla)\left(e_{T_F}\right)\right) ds \\ &= \rho_0 c_p\left(\left(\frac{1}{2}\boldsymbol{u} + \frac{1}{2}\boldsymbol{U}\right) \cdot \nabla\right)\left(e_{T_F}\right).\end{aligned}$$

The adjoint linearized operators, $\bar{c}^*_{2\boldsymbol{u}}(\theta)$ and $\bar{c}^*_{2T}(\theta)$, are defined to satisfy $(\bar{c}_{2\boldsymbol{u}}(\boldsymbol{e_u}), \theta) = (\boldsymbol{e_u}, \bar{c}^*_{2\boldsymbol{u}}(\theta))$ and $(\bar{c}_{2T}(e_{T_F}), \theta) = (e_{T_F}, \bar{c}^*_{2T}(\theta))$ respectively. Using the divergence theorem, we find

$$\bar{c}^*_{2u}(\theta) = \rho_0 c_p \nabla\left(\frac{1}{2}T + \frac{1}{2}T_h\right)\theta,$$

and

$$\bar{c}^*_{2T}(\theta) = -\rho_0 c_p\left(\frac{1}{2}\boldsymbol{u} + \frac{1}{2}\boldsymbol{u}_h\right) \cdot \nabla\theta - \rho_0 c_p\left(\nabla \cdot \left(\frac{1}{2}\boldsymbol{u} + \frac{1}{2}\boldsymbol{u}_h\right)\right)\theta.$$

We write the formal adjoint to the Boussinesq equations in strong form

$$\begin{cases} -\mu\Delta\boldsymbol{\phi} + \overline{\mathbf{c}}_1^*(\boldsymbol{\phi}) + \nabla z + \overline{\mathbf{c}}_{2u}^*(\theta) = \boldsymbol{\psi}_{\boldsymbol{u}}, & x \in \Omega, \\ -\nabla \cdot \boldsymbol{\phi} = \psi_p, & x \in \Omega, \\ -k_F\Delta\theta + \overline{c}_{2T}^*(\theta) + \rho_0\beta\boldsymbol{g}\cdot\boldsymbol{\phi} = \psi_T, & x \in \Omega, \end{cases} \tag{8.6.3}$$

with the adjoint boundary conditions

$$\begin{cases} \boldsymbol{\phi} = \mathbf{0}, & x \in \Gamma_{\boldsymbol{u},D}, \\ \mu\dfrac{\partial\boldsymbol{\phi}}{\partial\boldsymbol{n}} = \mathbf{0}, & x \in \Gamma_{\boldsymbol{u},N}, \\ \theta = 0, & x \in \Gamma_{T_F,D}, \\ k_F\dfrac{\partial\theta}{\partial\boldsymbol{n}} = 0, & x \in \Gamma_{T_F,N}. \end{cases} \tag{8.6.4}$$

In practice we cannot use the true adjoint (8.6.3) since the exact solution is unknown. We linearize the nonlinear operator around the approximate solutions and compute the adjoint of the linearized system. Computing as before, we have

$$\begin{cases} -\mu\Delta\boldsymbol{\phi} - \rho_0\boldsymbol{u}_h\cdot\nabla\boldsymbol{\phi} + \rho_0\left(\nabla\boldsymbol{u}_h\cdot\boldsymbol{\phi} - (\nabla\cdot\boldsymbol{u}_h)\boldsymbol{\phi}\right) + \nabla z + (\rho_0 c_p\nabla T_h)\theta = \boldsymbol{\psi}_{\boldsymbol{u}}, \\ -\nabla\cdot\boldsymbol{\phi} = \psi_p, \\ -k_S\Delta\theta - \rho_0 c_p\boldsymbol{u}_h\cdot\nabla\theta - \rho_0 c_p(\nabla\cdot\boldsymbol{u}_h)\theta + \rho_0\beta\boldsymbol{g}\cdot\boldsymbol{\phi} = \psi_T, \end{cases} \tag{8.6.5}$$

with the adjoint boundary conditions

$$\begin{cases} \boldsymbol{\phi} = \mathbf{0}, & x \in \Gamma_{\boldsymbol{u},D}, \\ \mu\dfrac{\partial\boldsymbol{\phi}}{\partial\boldsymbol{n}} = \mathbf{0}, & x \in \Gamma_{\boldsymbol{u},N}, \\ \theta = 0, & x \in \Gamma_{T_F,D}, \\ k_F\dfrac{\partial\theta}{\partial\boldsymbol{n}} = 0, & x \in \Gamma_{T_F,N}. \end{cases} \tag{8.6.6}$$

8.6.3 The adjoint to the conjugate heat transfer problem

We define $\boldsymbol{e}_u = \boldsymbol{u} - \boldsymbol{u}_h^{\{k\}}$, $e_p = p - p_h^{\{k\}}$, $e_{T_F} = T_F - T_{F,h}^{\{k\}}$ and $e_{T_S} = T_S - T_{S,h}^{\{k\}}$. The adjoint boundary value problem for the quantity of interest

$$(\psi, \mathbf{e}) = (\boldsymbol{\psi}_{\boldsymbol{u}}, \boldsymbol{e}_{\boldsymbol{u}}) + (\psi_p, e_p) + (\psi_{T_F}, e_{T_F}) + (\psi_{T_S}, e_{T_S})$$

for the coupled problem (8.2.1) is

$$\begin{cases} -\mu\Delta\boldsymbol{\phi} + \overline{\mathbf{c}}_1^*(\boldsymbol{\phi}) + \nabla z + \overline{\mathbf{c}}_{2u}^*(\theta_F) = \boldsymbol{\psi}_{\boldsymbol{u}}, & x \in \Omega_F, \\ -\nabla \cdot \boldsymbol{\phi} = \psi_p, & x \in \Omega_F, \\ -k_F\Delta\theta_F + \overline{c}_{2T}^*(\theta_F) + \rho_0\beta\boldsymbol{g}\cdot\boldsymbol{\phi} = \psi_{T_F}, & x \in \Omega_F, \\ \begin{cases} \theta_F = \theta_S, \\ k_F\dfrac{\partial\theta_F}{\partial\boldsymbol{n}} = k_S\dfrac{\partial\theta_S}{\partial\boldsymbol{n}}, \end{cases} & \mathbf{x} \in \Gamma_I, \\ -k_S\Delta\theta_S = \psi_{T_S}, & \mathbf{x} \in \Omega_S, \end{cases} \tag{8.6.7}$$

with adjoint boundary conditions

$$\begin{cases} \boldsymbol{\phi} = \mathbf{0}, & x \in \Gamma_{\boldsymbol{u},D}, \\ \mu\partial_n\boldsymbol{\phi} = \mathbf{0}, & x \in \Gamma_{\boldsymbol{u},N}, \\ \theta_F = 0, & x \in \Gamma_{T_F,D}, \\ k_F\dfrac{\partial\theta_F}{\partial\boldsymbol{n}} = 0, & x \in \Gamma_{T_F,N}, \\ \theta_S = 0, & x \in \Gamma_{T_S,D}, \\ k_S\dfrac{\partial\theta_S}{\partial\boldsymbol{n}} = 0, & x \in \Gamma_{T_S,N}. \end{cases} \tag{8.6.8}$$

We solve (8.6.7) numerically using the approximate Boussinesq adjoint (8.6.5) and an iterative operator decomposition approach as for the forward problem. These iterations are completely independent of the forward iterations.

8.6.4 An error representation

We can derive an error representation formula for the basic scheme (8.3.1)-(8.3.2), a weighted relaxation technique (8.3.6) or (8.3.7), or when using the post-processed flux as in (8.5.2). In the discussion below, we use $\chi_h^{\{k\}}$ to denote the numerical flux passed at the k^{th} iteration from Ω_S to Ω_F. To begin, we multiply the system (6.4.1) by $\boldsymbol{\psi}$ and apply the divergence

theorem, noting that $\theta_F = \theta_S$ and $k_F\partial_n\theta_F = k_S\partial_n\theta_S$ along Γ_I, to obtain

$$\begin{aligned}(\psi, \mathrm{e}) = \; & a_1(e_{\boldsymbol{u}}, \boldsymbol{\phi}) + c_1(\boldsymbol{u}, \boldsymbol{u}, \boldsymbol{\phi}) - c_1(\boldsymbol{u}_h^{\{k\}}, \boldsymbol{u}_h^{\{k\}}, \boldsymbol{\phi}) + b(\boldsymbol{\phi}, e_p) + d(e_{T_F}, \boldsymbol{\phi}) \\ & + b(e_{\boldsymbol{u}}, z) + a_2(e_{T_F}, \theta_F) + c_2(\boldsymbol{u}, T_F, \theta_F) - c_2(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}}, \theta_F) \\ & + a_3(e_{T_S}, \theta_S) \\ & + (T_{S,h}^{\{k\}}, k_S\partial_n\theta_S)_{\Gamma_I} - (T_{F,h}^{\{k\}}, k_F\partial_n\theta_F)_{\Gamma_I}.\end{aligned}$$

Observe that the test space $W_{S,0}^h$ consists of functions that are zero along the interface, while in general, θ_S is not zero along Γ_I. This means that the projection of θ_S into $W_{S,0}^h$ cannot be the interpolant. We define a new projection $\pi_{W_S}^0 : H^2 \to W_{S,0}^h$ such that for any node x_i

$$\pi_{W_S}^0\theta_S(x_i) = \begin{cases} \pi_{W_S}\theta_S(x_i), & x_i \notin \Gamma_I, \\ 0, & x_i \in \Gamma_I. \end{cases} \tag{8.6.9}$$

We also observe that

$$\begin{aligned}a_2(e_{T_F}, \pi_{W_F}\theta_F) + c_2(\boldsymbol{u}, T_F, \pi_{W_F}\theta_F) - c_2(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}}, \pi_{W_F}\theta_F) \\ = -(k_S\partial_n T_S, \pi_{W_F}\theta_F)_{\Gamma_I} + (\chi_h^{\{k\}}, \pi_{W_F}\theta_F)_{\Gamma_I},\end{aligned}$$

and

$$a_3(T_S, \theta_S - \pi_{W_S}^0\theta_S) = (Q_S, \theta_S - \pi_{W_S}^0\theta_S) + (k_S\partial_n T_S, \theta_S)_{\Gamma_I},$$

$$a_2(T_F, \theta_F - \pi_{W_F}\theta_F) = (Q_F, \theta_F - \pi_{W_F}\theta_F) - (k_S\partial_n T_S, \theta_F - \pi_{W_F}\theta_F)_{\Gamma_I},$$

since the adjoint solutions are not zero along Γ_I.

Using the projection (8.6.9) in the Galerkin orthogonality relation and the equalities above, we have

$$
\begin{aligned}
(\psi, e) = (\mathbf{f}, \phi - \pi_V\phi) - a_1(\boldsymbol{u}_h^{\{k\}}, \phi - \pi_V\phi_1) - c_1(\boldsymbol{u}_h^{\{k\}}, \boldsymbol{u}_h^{\{k\}}, \phi - \pi_V\phi) \\
- b(\phi - \pi_V\phi, p_h) - d(T_{F,h}^{\{k\}}, \phi - \pi_V\phi) - b(\boldsymbol{u}_h^{\{k\}}, z - \pi_Z z) \\
+ (Q_F, \theta_F - \pi_{W_F}\theta_F) - a_2(T_{F,h}^{\{k\}}, \theta_F - \pi_{W_F}\theta_F) - c_2(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}}, \theta_F - \pi_{W_F}\theta_F) \\
+ (Q_S, \theta_S - \pi_{W_S}^0\theta_S) - a_3(T_{S,h}^{\{k\}}, \theta_S - \pi_{W_S}^0\theta_S) \\
+ (T_{S,h}^{\{k\}} - T_{F,h}^{\{k\}}, k_S\partial_n\theta_S)_{\Gamma_I} + (\chi_h^{\{k\}}, \pi_{W_F}\theta_F)_{\Gamma_I}. \quad (8.6.10)
\end{aligned}
$$

Next, we define $\pi_\partial\theta_S = \pi_{W_S}\theta_S - \pi_{W_S}^0\theta_S$ which is nonzero only near the interface due to the definition of $\pi_{W_S}^0\theta_S$. Substituting $\pi_{W_S}^0\theta_S = \pi_{W_S}\theta_S - \pi_\partial\theta_S$ gives

$$
\begin{aligned}
(Q_S, \theta_S - \pi_{W_S}^0\theta_S) - a_3(T_{S,h}^{\{k\}}, \theta_S - \pi_{W_S}^0\theta_S) = (Q_S, \theta_S - \pi_{W_S}\theta_S) \\
- a_3(T_{S,h}^{\{k\}}, \theta_S - \pi_{W_S}\theta_S) - (Q_S, \pi_\partial\theta_S) + a_3(T_{S,h}^{\{k\}}, \pi_\partial\theta_S)
\end{aligned}
$$

Finally (6.3.7) implies that the recovered boundary flux, $\sigma^{\{k\}}$ satisfies

$$
-(\sigma^{\{k\}}, \pi_\partial\theta_S)_{\Gamma_I} = (Q_S, \pi_\partial\theta_S) - a_1(\boldsymbol{u}_h^{\{k\}}, \pi_\partial\theta_S),
$$

while $\pi_\partial\theta_S = \pi_{W_S}\theta_S$ along Γ_I.

We substitute into (8.6.10) and conclude

Theorem 8.6.1. *The errors* $\boldsymbol{e_u} = \boldsymbol{u} - \boldsymbol{u}_h^{\{k\}}$, $e_p = p - p_h^{\{k\}}$, $e_{T_F} = T_F - T_{F,h}^{\{k\}}$ *and* $e_{T_S} = T_S - T_{S,h}^{\{k\}}$ *satisfy*

$$\begin{aligned}(\psi, e) = (\mathbf{f}, \phi - \pi_V\phi) - a_1(\boldsymbol{u}_h^{\{k\}}, \phi - \pi_V\phi_1) - c_1(\boldsymbol{u}_h^{\{k\}}, \boldsymbol{u}_h^{\{k\}}, \phi - \pi_V\phi) \\ - b(\phi - \pi_V\phi, p_h) - d(T_{F,h}^{\{k\}}, \phi - \pi_V\phi) - b(\boldsymbol{u}_h^{\{k\}}, z - \pi_Z z) \\ + (Q_F, \theta_F - \pi_{W_F}\theta_F) - a_2(T_{F,h}^{\{k\}}, \theta_F - \pi_{W_F}\theta_F) - c_2(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}}, \theta_F - \pi_{W_F}\theta_F) \\ + (Q_S, \theta_S - \pi_{W_S}\theta_S) - a_3(T_{S,h}^{\{k\}}, \theta_S - \pi_{W_S}\theta_S) \\ + (T_{S,h}^{\{k\}} - T_{F,h}^{\{k\}}, k_S\partial_n\theta_S)_{\Gamma_I} \\ + (\chi_h^{\{k\}}, \pi_{W_F}\theta_F)_{\Gamma_I} - (\sigma^{\{k\}}, \pi_{W_S}\theta_S)_{\Gamma_I}. \quad (8.6.11)\end{aligned}$$

The error has been decomposed into four discretization components, an iterative component, and a component reflecting the error arising from the transfer of derivative information. The choice of iterative method does not affect the discretization component of the error, but it does influence both the iterative and the transfer components along Γ_I. We illustrate this for a few common iterative schemes:

- Suppose $T_{S,h}^{\{k\}} = \pi_{W_S} T_{F,h}^{\{k-1\}}$, then

$$\begin{aligned}&(T_{S,h}^{\{k\}} - T_{F,h}^{\{k\}}, k_S\partial_n\theta_S)_{\Gamma_I} \\ &\quad = (\pi_{W_S}T_{F,h}^{\{k-1\}} - T_{F,h}^{\{k-1\}}, k_S\partial_n\theta_S)_{\Gamma_I} + (T_{F,h}^{\{k\}} - T_{F,h}^{\{k-1\}}, k_S\partial_n\theta_S)_{\Gamma_I},\end{aligned}$$

  which represents a projection error and an iteration error.

- Suppose $T_{S,h}^{\{k\}} = \alpha T_{S,h}^{\{k-1\}} + (1-\alpha)\pi_1 T_{F,h}^{\{k-1\}}$, then

$$\begin{aligned}(T_{S,h}^{\{k\}} - T_{F,h}^{\{k\}}, k_S\partial_n\theta_S)_{\Gamma_I} = &\sum_{i=1}^{k-2} \alpha^{i-1}(\pi_{W_S}(T_{S,h}^{(k-i-1)} - T_{F,h}^{(k-i)}), k_S\partial_n\theta_S)_{\Gamma_I} \\ &+ (T_{F,h}^{\{k-1\}} - T_{F,h}^{\{k\}}, k_S\partial_n\theta_S)_{\Gamma_I} \\ &+ (\pi_{W_S}T_{F,h}^{\{k-1\}} - T_{F,h}^{\{k-1\}}, k_S\partial_n\theta_S)_{\Gamma_I},\end{aligned}$$

which represents a series of iteration errors and a projection error. Notice that since $\alpha < 1$, the effect of the iteration error from previous iterations diminishes due to the increasing power on α.

- Suppose we set $\chi_h^{\{k\}} = k_S \partial_n T_{S,h}^{\{k\}}$, then

$$(\chi_h^{\{k\}}, \pi_{W_F}\theta_F)_{\Gamma_I} - (\sigma^{\{k\}}, \pi_{W_S}\theta_S)_{\Gamma_I} = (k_S \partial_n T_{S,h}^{\{k\}} - \sigma^{\{k\}}, \pi_{W_F}\theta_F)_{\Gamma_I} + (\sigma^{\{k\}}, \pi_{W_F}\theta_F - \pi_{W_S}\theta_S)_{\Gamma_I},$$

which represents a transfer error and a projection error.

- Suppose we set $\chi_h^{\{k\}} = \sigma^{\{k\}}$, then

$$(\chi_h^{\{k\}}, \pi_{W_F}\theta_F)_{\Gamma_I} - (\sigma^{\{k\}}, \pi_{W_S}\theta_S)_{\Gamma_I} = (\sigma^{\{k\}}, \pi_{W_F}\theta_F - \pi_{W_S}\theta_S)_{\Gamma_I},$$

which represents only a projection error with *no transfer error.*

8.6.5 Adaptive refinement

We use the *a posteriori* error estimate as the basis for adaptivity by employing the standard "optimization framework" after writing the estimate as a sum of element contributions and introducing norms [26, 27, 12, 10],

$$\begin{aligned}\eta_K = \big|&(\mathbf{f}, \phi - \pi_V\phi)_K - a_1(\boldsymbol{u}_h^{\{k\}}, \phi - \pi_V\phi_1)_K - c_1(\boldsymbol{u}_h^{\{k\}}, \boldsymbol{u}_h^{\{k\}}, \phi - \pi_V\phi)_K \\ &- b(\phi - \pi_V\phi, p_h)_K - d(T_{F,h}^{\{k\}}, \phi - \pi_V\phi)_K - b(\boldsymbol{u}_h^{\{k\}}, z - \pi_Z z)_K \\ &+ (Q_F, \theta_F - \pi_{W_F}\theta_F)_K - a_2(T_{F,h}^{\{k\}}, \theta_F - \pi_{W_F}\theta_F)_K - c_2(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}}, \theta_F - \pi_{W_F}\theta_F)_K \\ &+ (Q_S, \theta_S - \pi_{W_S}\theta_S)_K - a_3(T_{S,h}^{\{k\}}, \theta_S - \pi_{W_S}\theta_S)_K \\ &+ (T_{S,h}^{\{k\}} - T_{F,h}^{\{k\}}, k_S \partial_n \theta_S)_{\partial K \cap \Gamma_I} \\ &+ (\chi_h^{\{k\}}, \pi_{W_F}\theta_F)_{\Gamma_I} - (\sigma^{\{k\}}, \pi_{W_S}\theta_S)_{\partial K \cap \Gamma_I}\big|, \quad (8.6.12)\end{aligned}$$

with the obvious notation for localizing the forms to an element K. An element is marked for refinement when the local error indicator is larger than a local tolerance, which may be

- a predetermined element tolerance,
- a global tolerance divided by the current number of elements,
- a tolerance based on the percentage of the elements we are willing to refine.

In section 8.7, we use (8.6.12) with a predetermined element tolerance to drive adaptivity for a variety of linear functionals.

8.7 Numerical Results

In this section, we present computational results illustrating the adaptive procedure given in §8.6.

8.7.1 Motivational Problem revisited

We reconsider the steady flow of a viscous Newtonian fluid in a channel connected along one boundary to a solid which is heated from below as shown in Fig. 8.2.

As before, we first solve the problem using the finite element flux, i.e., using $\chi^{(k)} = k_S \nabla T_{S,h}^{\{k\}} \cdot \boldsymbol{n}$ in (8.3.4). Next, we compute and pass the recovered boundary flux, i.e., using $\chi^{(k)} = \sigma^{\{k\}}$ in (8.3.4). We also compute a reference solution with a higher order method for comparison. In Fig. 8.5 we compare the L^2 errors in the temperature fields over $\Omega_S \cup \Omega_F$ on a series of meshes which align along the interface [51]. We observe that the approximation using the finite element flux converges quadratically with respect to the mesh size, rather than the expected cubic rate. We also see in Fig. 8.5, that solving the iterative system using the recovered boundary flux restores the optimal cubic order of convergence.

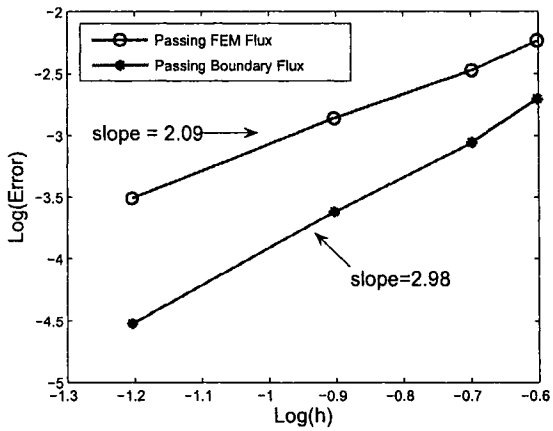

Figure 8.5: Comparison of the mesh size, h, versus the L^2 error in the temperature field when the finite element flux is passed, and when the recovered boundary flux is passed.

8.7.2 Flow past a cylinder

We now return to our original problem and consider the flow past a cylinder as shown in Figure 8.1. We solve the steady non-dimensionalized Boussinesq equations in the fluid domain

$$\begin{cases} Re\,(\boldsymbol{u}\cdot\nabla)\,\boldsymbol{u} = -\nabla p + \Delta\boldsymbol{u} - Pe\left(\dfrac{1}{PrFr} - Ra\,T_F\right)\mathrm{j}, \\ -\nabla\cdot\boldsymbol{u} = 0, \\ \boldsymbol{u}\cdot\nabla T_F = \dfrac{1}{Pe}\Delta T_F + Q_F \end{cases}$$

and the non-dimensional heat equation in the solid domain

$$\left\{ \frac{k_r}{Pe}\Delta T_S = Q_S, \right.$$

coupled by the interface condition

$$\begin{cases} T_S = T_F, \\ k_r \nabla T_S\cdot\boldsymbol{n} = \nabla T_F\cdot\boldsymbol{n}. \end{cases}$$

The non-dimensional groups based on length scale L, velocity scale $\bar{U}$ and a temperature non-dimensionalization $T = (T^* - T_0^*)/\Delta T$ are

$$Re = \frac{\bar{U}L}{\nu}, \quad Pe = \frac{\bar{U}L}{\kappa}, \quad Pr = \frac{\nu}{\kappa}, \quad Fr = \frac{\bar{U}^2}{gL}, \quad Ra = \frac{g\alpha L^3 \Delta T}{\nu\kappa}, \quad k_r = \frac{k_S}{k_F},$$

and $\boldsymbol{n}$ points from the fluid into the solid and $\kappa = k_F/(\rho c_p)$. Defining the velocity scale, $\bar{U}$, to be

$$\bar{U} = \frac{1}{L}\int_{-L/2}^{L/2} u(y)\, dy,$$

and for a blockage ratio, $B = d/L = 0.5$, the computational study by Chen, et. al [23] indicates that the critical Reynolds number at which the flow becomes unstable at a Hopf bifurcation point exceeds 100.

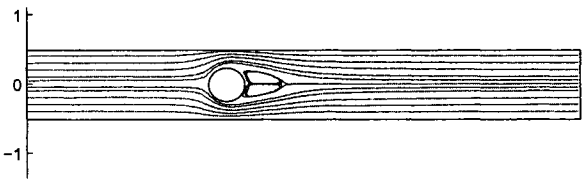

Figure 8.6: Streamlines for flow past a cylinder.

We use the error indicator given by (8.6.12) to adapt the mesh when the quantity of interest is the temperature at a small region in the wake above the cylinder. The final adaptive fluid mesh is given in Fig. 8.7. The adaptive scheme concentrates mesh refinement near the region of interest and upstream of the region of interest, locating more elements between the cylinder and the top wall than the cylinder and the bottom wall since the flow advecting heat to the region of interest passes above rather than below the cylinder. The solution downstream of the region of interest can be computed with much less accuracy as is recognized by the much coarser mesh. We note that the presence of gravity causes the mesh refinement to be slightly nonsymmetric.

In Fig. 8.8, we compare the mesh produced by the adaptive strategy within the solid when using $\chi^{(k)} = k_S \nabla T_{S,h}^{\{k\}} \cdot \boldsymbol{n}$ in (8.3.4) (left) and when using $\chi^{(k)} = \sigma^{\{k\}}$ (right).

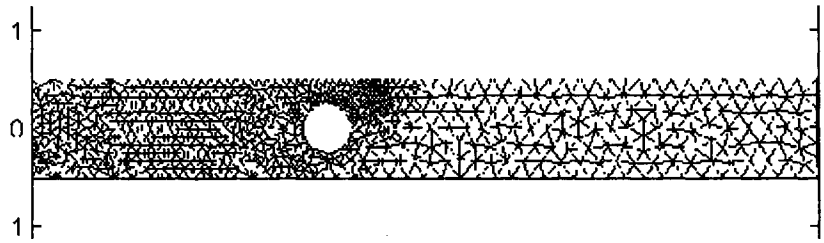

Figure 8.7: Final adaptive mesh in the fluid when the quantity of interest is the temperature in a small region in the wake above the cylinder.

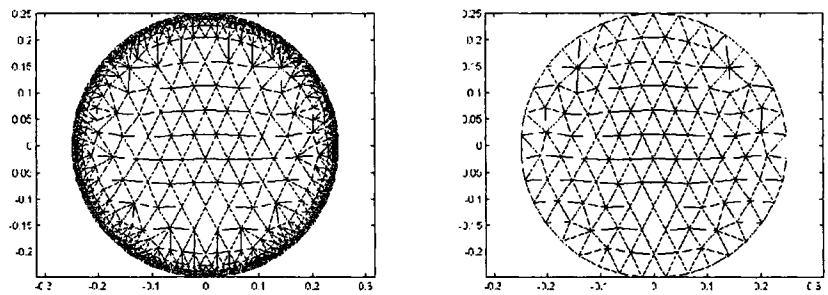

Figure 8.8: Final adaptive mesh in the solid when the quantity of interest is the temperature in a small region in the wake above the cylinder in the case when the finite element flux is passed (left), and in the case when the recovered boundary flux is passed (right).

When using the finite element flux (left), the grid is greatly refined near the solid boundary in order to increase the accuracy of the normal derivative that is computed in the solid and used as a boundary condition in the fluid computation. When using the recovered boundary flux, the solid is essentially refined uniformly since there is no need for local grid refinement in order to increase accuracy of the normal derivative used for the fluid computation. At the expense of solving a simple lower dimensional

problem for the recovered flux, the computation using the recovered boundary flux required 1581 elements in the combined domains while the original computation using the finite element flux required 3284 elements, or over twice the number of elements. These extra elements are all contained in the solid, since the fluid mesh in both computations was essentially the same.

Next, we use the error indicator given by (8.6.12) to adapt the mesh when the quantity of interest is the temperature at a small region in the center of the cylinder. The final adaptive fluid mesh is given in Fig. 8.9. The adaptive scheme concentrates mesh refinement upstream of the cylinder while the solution downstream of the region of interest can be computed with much less accuracy as is recognized by the much coarser mesh.

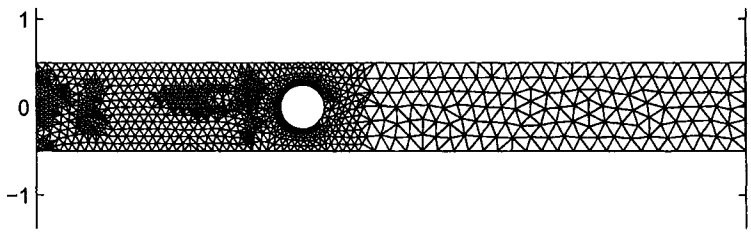

Figure 8.9: Final adaptive mesh in the fluid when the quantity of interest is the temperature in a small region in the center of the solid.

In Fig. 8.10, we compare the mesh produced by the adaptive strategy within the solid when using $\chi^{(k)} = k_S \nabla T_{S,h}^{\{k\}} \cdot \boldsymbol{n}$ in (8.3.4) (left) and when using $\chi^{(k)} = \sigma^{\{k\}}$ (right). As before, the final adaptive mesh in the solid is refined near the boundary when the finite element flux is passed, reflecting the fact that the error in the finite element flux makes a significant contribution to the error in the quantity of interest. When the recovered boundary flux is passed, there is no additional refinement along the edge of the cylinder, resulting in much fewer elements required to achieve the same

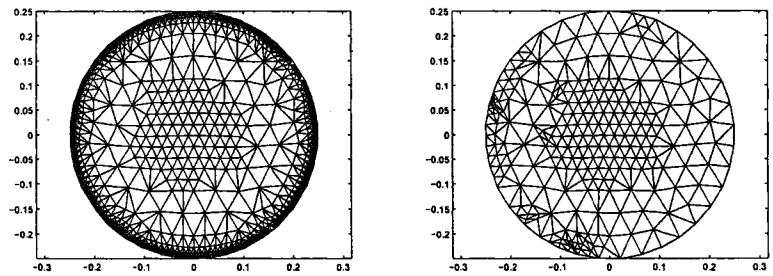

Figure 8.10: Final adaptive mesh in the solid when the quantity of interest is the temperature in a small region in the center of the solid in the case when the finite element flux is passed (left), and in the case when the recovered boundary flux is passed (right).

accuracy. We also note that when the recovered flux is passed, the adaptive procedure refines along the boundary in the upstream direction, indicating that the temperature at the center of the cylinder depends strongly on upstream information.

8.8 An analysis of the loss of order

We now analyse the loss of order and show the flux recovery technique recovers the optimal convergence rate. We use the adjoint for the fully coupled problem to derive *a posteriori* error bounds for the iterative approximations in the case where all of the variables are sufficiently smooth and $T_{S,h}^{(k)} = \pi_{W_S} T_{F,h}^{\{k-1\}}$. The case where the Dirichlet values are updated using a relaxation technique can be analyzed using the same approach and gives identical results.

8.8.1 L^2 error bounds using finite element flux

Let $\boldsymbol{u} \in H^3(\Omega_F)$, $p \in H^2(\Omega_F)$, $T_F \in H^3(\Omega_F)$, and $T_S \in H^3(\Omega_S)$ be the solutions to (8.2.1), and $\boldsymbol{u}_h^{\{k\}}$, $p_h^{\{k\}}$, $T_{F,h}^{\{k\}}$ and $T_{S,h}^{\{k\}}$ be the finite element

solutions from the operator decomposition method at the k^{th} iteration. Let $\phi \in H^3(\Omega_F)$, $z \in H^2(\Omega_F)$, $\theta_F \in H^3(\Omega_F)$ and $\theta_S \in H^3(\Omega_S)$ and pose the adjoint problem (8.6.7) with $\psi_{\boldsymbol{u}} = \mathbf{0}$, $\psi_p = 0$, $\psi_{T_F} = e_{T_F}/\|e_{T_F}\|_{\Omega_F}$ and $\psi_{T_S} = e_{T_S}/\|e_{T_S}\|_{\Omega_S}$. Starting with (8.6.11), integration by parts over each element K gives

$$\|e_{T_F}\|_{\Omega_F} + \|e_{T_S}\|_{\Omega_S} = I_1 + I_2 + I_3 + I_4,$$

where

$$\begin{aligned}
I_1 = &\sum_{K\in\tau_{F,h}} \left(R_1(\boldsymbol{u}_h^{\{k\}}, p_h^{\{k\}}, T_{F,h}^{\{k\}}), \phi - \pi_V\phi\right)_K + \frac{1}{2}\left([\mu\partial_n \boldsymbol{u}_h^{\{k\}}], \phi - \pi_V\phi\right)_{\partial K} \\
&+ \sum_{K\in\tau_{F,h}} \left(R_2(\boldsymbol{u}_h^{\{k\}}), z - \pi_Z z\right)_K \\
&+ \sum_{K\in\tau_{F,h}} \left(R_3(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}}), \theta_F - \pi_{W_F}\theta_F\right)_K + \frac{1}{2}\left([k_F\partial_n T_{F,h}^{\{k\}}], \theta_F - \pi_{W_F}\theta_F\right)_{\partial K} \\
&+ \sum_{K\in\tau_{S,h}} \left(R_4(T_{S,h}^{\{k\}}), \theta_S - \pi_{W_S}\theta_S\right)_K + \frac{1}{2}\left([k_S\partial_n T_{S,h}^{\{k\}}], \theta_S - \pi_{W_S}\theta_S\right)_{\partial K} \\
I_2 = &(k_F\partial_n T_{F,h}^{\{k\}}, \theta_F - \pi_{W_F}\theta_F)_{\Gamma_I} - (k_S\partial_n T_{S,h}^{\{k\}}, \theta_S - \pi_{W_S}\theta_S)_{\Gamma_I}, \\
I_3 = &(k_S\partial_n\theta_S, T_{S,h}^{\{k\}} - T_{F,h}^{\{k\}})_{\Gamma_I}, \\
I_4 = &(\chi^{\{k\}}, \pi_{W_F}\theta_F)_{\Gamma_I} - (\sigma^{\{k\}}, \pi_{W_S}\theta_S)_{\Gamma_I},
\end{aligned}$$

with $[\cdot]$ denoting the jump across an element edge and

$$\begin{aligned}
R_1(\boldsymbol{u}_h^{\{k\}}, p_h^{\{k\}}, T_{F,h}^{\{k\}}) &= \mathbf{f} + \mu\Delta\boldsymbol{u}_h^{\{k\}} - \rho_0\left(\boldsymbol{u}_h^{\{k\}}\cdot\nabla\right)\boldsymbol{u}_h^{\{k\}} - \nabla p_h^{\{k\}} - \rho_0\beta T_{F,h}^{\{k\}}\mathbf{g} \\
R_2(\boldsymbol{u}_h^{\{k\}}) &= \nabla\cdot\boldsymbol{u}_h^{\{k\}} \\
R_3(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}}) &= Q_F + k_F\Delta T_{F,h}^{\{k\}} - \rho_0 c_p \boldsymbol{u}_h^{\{k\}}\cdot\nabla T_{F,h}^{\{k\}} \\
R_4(T_{S,h}^{\{k\}}) &= Q_S + k_S\Delta T_{S,h}^{\{k\}}
\end{aligned}$$

The first term I_1 is a standard *a posteriori* residual term and is not affected by non-matching triangulations along the interface or by transfer error. The second term I_2 is similar to the jump terms along element

edges in I_1 and is the expected jump term when the triangulations align along the interface. The third term I_3 represents the jump in the Dirichlet values across the interface. Finally, the fourth term I_4 represents the difference between the flux passed from Ω_S to Ω_F and the flux obtained via the boundary-flux correction technique.

Below, we present Lemmas 8.8.1 to 8.8.4 below to bound I_1 to I_4 individually. In each of these Lemmas we first provide the general bound when the triangulations do not match across the boundary and then show the simplification that arises for matching triangulations. We then combine these four lemmas into Theorems 8.8.1 and 8.8.2 which give error bounds for the basic iteration (8.3.3) and (8.3.4), and when using flux correction (8.5.2) respectively. These two theorems describe the general result when the triangulations do not match across the boundary while the simplification given matching triangulations is provided as a Corollary.

In each of the Lemmas to follow, we will need to add and subtract a smooth function to complete the error bounds. However, $T_S - T_{S,h}^{\{k\}}$, for example, contains iteration and pollution errors as well as discretization errors. Similarly, $T_S^{\{k\}} - T_{S,h}^{\{k\}}$ contains pollution and discretization errors. Therefore, we need to define a new function, $\hat{T}_S^{\{k\}}$, such that $\hat{T}_S^{\{k\}} - T_{S,h}^{\{k\}}$ contains only discretization errors. We define $\hat{T}_S^{\{k\}}$ such that

$$\begin{cases} -k_S \Delta \hat{T}_S^{\{k\}} = Q_S, & \mathbf{x} \in \Omega_S \\ \hat{T}_S^{\{k\}} = g_{\Gamma_I,D}^{\{k\}}, & \mathbf{x} \in \Gamma_I \\ \hat{T}_S^{\{k\}} = g_{T_S,D}, & \mathbf{x} \in \Gamma_{T_S,D} \\ k_S \partial_n \hat{T}_S^{\{k\}} = g_{T_S,N}, & \mathbf{x} \in \Gamma_{T_S,N}, \end{cases} \tag{8.8.1}$$

where the Dirichlet data $g_{\Gamma_I,D}$ along the interface in the solid is defined to be smooth function with $\pi_{W_S} g_{\Gamma_I,D} = \pi_{W_S} T_{F,h}^{\{k-1\}}$ and $\pi_{W_F} g_{\Gamma_I,D} = T_{F,h}^{\{k-1\}}$.

Next, we define $\hat{\boldsymbol{u}}^{\{k\}}$, $\hat{p}^{\{k\}}$ and $\hat{T}_F^{\{k\}}$ such that

$$\begin{cases} -\mu\Delta\hat{\boldsymbol{u}}^{\{k\}} + \rho_0\left(\hat{\boldsymbol{u}}^{\{k\}}\cdot\nabla\right)\hat{\boldsymbol{u}}^{\{k\}} + \nabla\hat{p}^{\{k\}} + \rho_0\beta\hat{T}_F^{\{k\}}\boldsymbol{g} = \mathbf{f}, & x\in\Omega_F \\ -\nabla\cdot\hat{\boldsymbol{u}}^{\{k\}} = 0, & x\in\Omega_F \\ -k_F\Delta\hat{T}_F^{\{k\}} + \rho_0 c_p\hat{\boldsymbol{u}}^{\{k\}}\cdot\nabla\hat{T}_F^{\{k\}} = Q_F, & x\in\Omega_F \\ k_F\partial_n\hat{T}_F^{\{k\}} = g_{\Gamma_I,N}, & \mathbf{x}\in\Gamma_I, \end{cases} \tag{8.8.2}$$

with the boundary conditions

$$\begin{cases} \hat{\boldsymbol{u}}^{\{k\}} = \boldsymbol{g}_{\boldsymbol{u},D}, & \mathbf{x}\in\Gamma_{\boldsymbol{u},D} \\ \mu\partial_n\hat{\boldsymbol{u}}^{\{k\}} = \boldsymbol{g}_{\boldsymbol{u},N}, & \mathbf{x}\in\Gamma_{\boldsymbol{u},N} \\ \hat{T}_F^{\{k\}} = g_{T_F,D}, & \mathbf{x}\in\Gamma_{T_F,D} \\ \hat{T}_F^{\{k\}} = g_{T_F,N}, & \mathbf{x}\in\Gamma_{T_F,N} \end{cases} \tag{8.8.3}$$

where the Neumann data $g_{\Gamma_I,N}$ along the interface in the solid is defined to be a smooth function such that $(g_{\Gamma_I,N}, v) = (k_S\partial_n T_{S,h}^{\{k\}}, v)$ for all $v\in P^1(K)$ for all $K\in\tau_{S,h}$ and $(g_{\Gamma_I,N}, v) = (k_F\partial_n T_{F,h}^{\{k\}}, v)$ for all $v\in P^1(K)$ for all $K\in\tau_{F,h}$.

We assume that each of the domains are sufficiently regular and that the data for each problem is sufficiently smooth that the following regularity estimates hold [13, 40]:

$$\|\hat{T}_S^{\{k\}}\|_{3,\Omega_S} \le C\|\hat{d}_S^{\{k\}}\|_{1,\Omega_S}, \tag{8.8.4}$$

where

$$\|\hat{d}_S^{\{k\}}\|_{1,\Omega_S} = \|Q_S\|_{1,\Omega_S} + \|g_{\Gamma_I,D}^{\{k\}}\|_{1,\Gamma_I} + \|g_{T_S,D}\|_{1,\Gamma_{T_S,D}} + \|g_{T_S,N}\|_{1,\Gamma_{T_S,N}}, \tag{8.8.5}$$

and

$$\|\hat{T}_F^{\{k\}}\|_{3,\Omega_F} + \|\hat{\boldsymbol{u}}^{\{k\}}\|_{3,\Omega_F} + \|\hat{p}^{\{k\}}\|_{2,\Omega_F} \le \|\hat{d}_F^{\{k\}}\|_{1,\Omega_F}, \tag{8.8.6}$$

where

$$\begin{aligned} \|\hat{d}_F^{\{k\}}\|_{1,\Omega_F} = {} & \|Q_F\|_{1,\Omega_F} + \|\mathbf{f}\|_{1,\Omega_F} + \|\mathbf{g}_{\boldsymbol{u},D}\|_{1,\Gamma_{\boldsymbol{u},D}} + \|\mathbf{g}_{\boldsymbol{u},N}\|_{1,\Gamma_{\boldsymbol{u},D}} \\ & + \|g_{T_F,D}\|_{1,\Gamma_{T_F,D}} + \|g_{T_F,N}\|_{1,\Gamma_{T_F,N}} + \|g_{\Gamma_I,N}^{\{k\}}\|_{1,\Gamma_I}. \end{aligned} \tag{8.8.7}$$

Furthermore, if we assume the data is sufficiently small in relation to certain nondimensional groups to guarantee coercivity [41, 40], then the following *a priori* error bounds hold:

$$\|\hat{T}_S^{\{k\}} - T_{S,h}^{\{k\}}\|_{1,\Omega_S} \leq Ch^2 \|\hat{d}_S^{\{k\}}\|_{1,\Omega_S} \tag{8.8.8}$$

and

$$\|\hat{T}_F^{\{k\}} - T_{F,h}^{\{k\}}\|_{1,\Omega_F} + \|\hat{p}^{\{k\}} - p_h^{\{k\}}\|_{0,\Omega_F} + \|\hat{\boldsymbol{u}}^{\{k\}} - \boldsymbol{u}_h^{\{k\}}\|_{1,\Omega_F} \leq Ch^2 \|\hat{d}_F^{\{k\}}\|_{1,\Omega_F}. \tag{8.8.9}$$

Lemma 8.8.1. *(Bound on I_1)*

$$I_1 \leq \sum_{K \in \tau_{F,h}} \left(Ch_K^3 \|\hat{d}_F^{\{k\}}\|_K \cdot (|\boldsymbol{\phi}|_{K,2} + |z|_{K,2} + |\theta_F|_{K,2}) \right) + \sum_{K \in \tau_{S,h}} \left(Ch_K^3 \|\hat{d}_S^{\{k\}}\|_K \cdot |\theta_S|_{K,2} \right)$$

with the obvious meaning for localizing $\hat{d}_F^{\{k\}}$ and $\hat{d}_S^{\{k\}}$ to an element K.

Proof. Recall that

$$\begin{aligned} I_1 = & \sum_{K \in \tau_{F,h}} \left(R_1(\boldsymbol{u}_h^{\{k\}}, p_h^{\{k\}}, T_{F,h}^{\{k\}}), \boldsymbol{\phi} - \pi_V \boldsymbol{\phi} \right)_K + \frac{1}{2} \left([\mu \partial_n \boldsymbol{u}_h^{\{k\}}], \boldsymbol{\phi} - \pi_V \boldsymbol{\phi} \right)_{\partial K} \\ & + \sum_{K \in \tau_{F,h}} \left(R_2(\boldsymbol{u}_h^{\{k\}}), z - \pi_Z z \right)_K \\ & + \sum_{K \in \tau_{F,h}} \left(R_3(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}}), \theta_F - \pi_{W_F} \theta_F \right)_K + \frac{1}{2} \left([k_F \partial_n T_{F,h}^{\{k\}}], \theta_F - \pi_{W_F} \theta_F \right)_{\partial K} \\ & + \sum_{K \in \tau_{S,h}} \left(R_4(T_{S,h}^{\{k\}}), \theta_S - \pi_{W_S} \theta_S \right)_K + \frac{1}{2} \left([k_S \partial_n T_{S,h}^{\{k\}}], \theta_S - \pi_{W_S} \theta_S \right)_{\partial K} \end{aligned}$$

with

$$\begin{aligned} R_1(\boldsymbol{u}_h^{\{k\}}, p_h^{\{k\}}, T_{F,h}^{\{k\}}) &= \mathbf{f} + \mu \Delta \boldsymbol{u}_h^{\{k\}} - \rho_0 \left(\boldsymbol{u}_h^{\{k\}} \cdot \nabla \right) \boldsymbol{u}_h^{\{k\}} - \nabla p_h^{\{k\}} - \rho_0 \beta T_{F,h}^{\{k\}} \mathbf{g} \\ R_2(\boldsymbol{u}_h^{\{k\}}) &= \nabla \cdot \boldsymbol{u}_h^{\{k\}} \\ R_3(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}}) &= Q_F + k_F \Delta T_{F,h}^{\{k\}} - \rho_0 c_p \boldsymbol{u}_h^{\{k\}} \cdot \nabla T_{F,h}^{\{k\}} \\ R_4(T_{S,h}^{\{k\}}) &= Q_S + k_S \Delta T_{S,h}^{\{k\}} \end{aligned}$$

with

$$
\begin{aligned}
R_1(\boldsymbol{u}_h^{\{k\}}, p_h^{\{k\}}, T_{F,h}^{\{k\}}) &= \mathbf{f} + \mu\Delta\boldsymbol{u}_h^{\{k\}} - \rho_0\left(\boldsymbol{u}_h^{\{k\}}\cdot\nabla\right)\boldsymbol{u}_h^{\{k\}} - \nabla p_h^{\{k\}} - \rho_0\beta T_{F,h}^{\{k\}}\mathbf{g} \\
R_2(\boldsymbol{u}_h^{\{k\}}) &= \nabla\cdot\boldsymbol{u}_h^{\{k\}} \\
R_3(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}}) &= Q_F + k_F\Delta T_{F,h}^{\{k\}} - \rho_0 c_p\boldsymbol{u}_h^{\{k\}}\cdot\nabla T_{F,h}^{\{k\}} \\
R_4(T_{S,h}^{\{k\}}) &= Q_S + k_S\Delta T_{S,h}^{\{k\}}
\end{aligned}
$$

First, we use the definition of $\hat{\boldsymbol{u}}^{\{k\}}$, $\hat{p}^{\{k\}}$, $\hat{T}_F^{\{k\}}$, and $\hat{T}_S^{\{k\}}$ to rewrite the residuals

$$
\begin{aligned}
R_1(\boldsymbol{u}_h^{\{k\}}, p_h^{\{k\}}, T_{F,h}^{\{k\}}) = &-\mu\Delta\left(\hat{\boldsymbol{u}}^{\{k\}} - \boldsymbol{u}_h^{\{k\}}\right) + \rho_0\left(\hat{\boldsymbol{u}}^{\{k\}}\cdot\nabla\right)\hat{\boldsymbol{u}}^{\{k\}} \\
&- \rho_0\left(\boldsymbol{u}_h^{\{k\}}\cdot\nabla\right)\boldsymbol{u}_h^{\{k\}} + \nabla\left(\hat{p}^{\{k\}} - p_h^{\{k\}}\right) \\
&+ \rho_0\beta\left(\hat{T}_F^{\{k\}} - T_{F,h}^{\{k\}}\right)\mathbf{g} \\
R_2(\boldsymbol{u}_h^{\{k\}}) = &-\nabla\cdot\left(\hat{\boldsymbol{u}}^{\{k\}} - \boldsymbol{u}_h^{\{k\}}\right) \\
R_3(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}}) = &-k_F\Delta\left(\hat{T}_F^{\{k\}} - T_{F,h}^{\{k\}}\right) \\
&+ \rho_0 c_p\hat{\boldsymbol{u}}^{\{k\}}\cdot\nabla\hat{T}_F^{\{k\}} - \rho_0 c_p\boldsymbol{u}_h^{\{k\}}\cdot\nabla T_{F,h}^{\{k\}} \\
R_4(T_{S,h}^{\{k\}}) = &-k_S\Delta\left(\hat{T}_S^{\{k\}} - T_{S,h}^{\{k\}}\right)
\end{aligned}
$$

Now, we bound each term individually.

The first term is $(R_1(\boldsymbol{u}_h^{\{k\}}, p_h^{\{k\}}, T_{F,h}^{\{k\}}), \phi - \pi_V\phi)_K$. We apply the Cauchy-Schwarz inequality

$$
|(R_1(\boldsymbol{u}_h^{\{k\}}, p_h^{\{k\}}, T_{F,h}^{\{k\}}), \phi - \pi_V\phi)_K| \leq \|R_1(\boldsymbol{u}_h^{\{k\}}, p_h^{\{k\}}, T_{F,h}^{\{k\}})\|_K \cdot \|\phi - \pi_V\phi\|_K.
$$

To bound the residual, we require the inverse estimate

$$
\|\hat{\boldsymbol{u}}^{\{k\}} - \boldsymbol{u}_h^{\{k\}}\|_{2,K} + \|\hat{p}^{\{k\}} - p_h^{\{k\}}\|_{1,K} + \|\hat{T}_F^{\{}k\} - T_{F,h}^{\{k\}}\|_{2,K} \leq Ch_K\|\hat{d}_F^{\{k\}}\|_{1,K}, \tag{8.8.10}
$$

assuming that the mesh is quasi-uniform. Each term in $R_1(\boldsymbol{u}_h^{\{k\}}, p_h^{\{k\}}, T_{F,h}^{\{k\}})$ may be bounded using either the inverse estimate or the *a priori* error bound (8.8.9), giving

$$\|R_1(\boldsymbol{u}_h^{\{k\}}, p_h^{\{k\}}, T_{F,h}^{\{k\}})\|_K \leq Ch_K \|\hat{d}_F^{\{k\}}\|_{1,K}.$$

The other component, $\|\phi - \pi_V \phi\|_K$, is easily bounded

$$\|\phi - \pi_V \phi\|_K \leq Ch_K^2 |\phi|_{2,K},$$

using an interpolation result.

The next residual term is $\left(R_2(\boldsymbol{u}_h^{\{k\}}), z - \pi_Z z\right)_K$. We apply the Cauchy-Schwarz inequality,

$$\left|\left(R_2(\boldsymbol{u}_h^{\{k\}}), z - \pi_Z z\right)_K\right| \leq C\|R_2(\boldsymbol{u}_h^{\{k\}})\|_K \cdot \|z - \pi_Z z\|_K,$$

where the *a priori* error bound (8.8.9) gives

$$\|R_2(\boldsymbol{u}_h^{\{k\}})\|_K \leq Ch_K^2 \|\hat{d}_F^{\{k\}}\|_{1,K},$$

and an interpolation result gives

$$\|z - \pi_Z z\|_K \leq Ch_K \|z\|_{1,K}.$$

Combining these results, we have

$$\left|\left(R_2(\boldsymbol{u}_h^{\{k\}}), z - \pi_Z z\right)_K\right| \leq Ch_K^3 \|\hat{d}_F^{\{k\}}\|_{1,K} \cdot \|z\|_{1,K}.$$

The third residual term is $\left(R_3(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}}), \theta_F - \pi_{W_F}\theta_F\right)_K$. We apply the Cauchy-Schwarz inequality,

$$\left|\left(R_3(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}}), \theta_F - \pi_{W_F}\theta_F\right)_K\right| \leq C\|R_3(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}})\|_K \cdot \|\theta_F - \pi_{W_F}\theta_F\|_K.$$

Each of the terms in the residual may be bounded with either the inverse estimate (8.8.10) or the *a priori* error bound (8.8.9)giving

$$\|R_3(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}})\|_K \leq Ch_K \|\hat{d}_F^{\{k\}}\|_{1,K},$$

and an interpolation result gives

$$\|\theta_F - \pi_{W_F}\theta_F\|_K \leq Ch_K^2 \|\theta_F\|_{2,K}.$$

Combining these results, we have

$$\left|\left(R_3(\boldsymbol{u}_h^{\{k\}}, T_{F,h}^{\{k\}}), \theta_F - \pi_{W_F}\theta_F\right)_K\right| \leq Ch_K^3 \|\hat{d}_F^{\{k\}}\|_{1,K} \cdot \|\theta_F\|_{2,K}.$$

Finally, we apply the Cauchy-Schwarz inequality to the fourth residual term

$$\left|\left(R_4(T_{S,h}^{\{k\}}), \theta_S - \pi_{W_S}\theta_S\right)_K\right| \leq C\|R_4(T_{S,h}^{\{k\}})\|_K \cdot \|\theta_S - \pi_{W_S}\theta_S\|_K.$$

The only term in the residual is bounded using the inverse estimate

$$\|\hat{T}_S^{\{}k\} - T_{S,h}^{\{k\}}\|_{2,K} \leq Ch_K \|\hat{d}_S^{\{k\}}\|_{1,K}. \tag{8.8.11}$$

An interpolation result bounds

$$\|\theta_S - \pi_{W_S}\theta_S\|_K \leq Ch_K^2 |\theta_S|_{2,K}.$$

We combine these results to conclude

$$\left|\left(R_4(T_{S,h}^{\{k\}}), \theta_S - \pi_{W_S}\theta_S\right)_K\right| \leq Ch_K^3 \|\hat{d}_S^{\{k\}}\|_{1,K} \cdot \|\theta_S\|_{2,K}.$$

The remaining terms represent jump terms over element edges. We provide the details for $\frac{1}{2}\left([\mu\partial_n \boldsymbol{u}_h^{\{k\}}], \phi - \pi_V\phi\right)_{\partial K}$ and note that the other two

terms, $\frac{1}{2}\left([k_F\partial_n T_{F,h}^{\{k\}}], \theta_F - \pi_{W_F}\theta_F\right)_{\partial K}$ and $\frac{1}{2}\left([k_S\partial_n T_{S,h}^{\{k\}}], \theta_S - \pi_{W_S}\theta_S\right)_{\partial K}$, may be handled similarly. We add and subtract $(\mu\partial_n\hat{\boldsymbol{u}}^{\{k\}}, \boldsymbol{\phi} - \pi_V\boldsymbol{\phi})_{\partial K}$ to give

$$\frac{1}{2}\left([\mu\partial_n\boldsymbol{u}_h^{\{k\}}], \boldsymbol{\phi} - \pi_V\boldsymbol{\phi}\right)_{\partial K} = \frac{1}{2}\left(\mu\partial_n\hat{\boldsymbol{u}}^{\{k\}} - \mu\partial_n\boldsymbol{u}_{h,K}^{\{k\}}, \boldsymbol{\phi} - \pi_V\boldsymbol{\phi}\right)_{\partial K} + \frac{1}{2}\left(\mu\partial_n\boldsymbol{u}_{h,K'}^{\{k\}} - \mu\partial_n\hat{\boldsymbol{u}}^{\{k\}}, \boldsymbol{\phi} - \pi_V\boldsymbol{\phi}\right)_{\partial K},$$

where $\boldsymbol{u}_{h,K}^{\{k\}}$ represents the approximation on element K and $\boldsymbol{u}_{h,K'}^{\{k\}}$ represents the approximation on a neighboring element K'. Next, we use the trace inequality

$$\|v\|_{\partial K} \leq C\|v\|_K^{1/2} \cdot \|v\|_{1,K}^{1/2}, \tag{8.8.12}$$

along with the *a priori* bound (8.8.9) and an interpolation result to conclude

$$\left|\frac{1}{2}\left([\mu\partial_n\boldsymbol{u}_h^{\{k\}}], \boldsymbol{\phi} - \pi_V\boldsymbol{\phi}\right)_{\partial K}\right| \leq \frac{1}{2}Ch_K^3\|\hat{d}_F^{\{k\}}\|_{1,K} \cdot |\boldsymbol{\phi}|_{2,K} + \frac{1}{2}Ch_{K'}^3\|\hat{d}_F^{\{k\}}\|_{1,K'} \cdot |\boldsymbol{\phi}|_{2,K'}$$

When we sum over all elements, each edge is counted twice which removes the factor of $1/2$. □

Lemma 8.8.2. *(Bound on I_2) If the triangulations $\tau_{F,h}$ and $\tau_{S,h}$ do not match along the interface, then*

$$I_2 \leq \left(Ch_F^3\|\hat{d}_F^{\{k\}}\|_{1,\Omega_F} \cdot |\theta_F|_{2,\Omega_F}\right) + \left(\|g_{\Gamma_I,N}^{\{k\}}\|_{\Gamma_I}\right) \cdot \left(\|\pi_{W_S}\theta_S - \pi_{W_F}\theta_F\|_{\Gamma_I}\right) + \left(Ch_S^3\|\hat{d}_S^{\{k\}}\|_{1,\Omega_S} \cdot |\theta_S|_{2,\Omega_S}\right)$$

where

$$\|\pi_{W_S}\theta_S - \pi_{W_F}\theta_F\|_{\Gamma_I} \leq C_1h_S^{5/2}|\theta_S|_{3,\Omega_S} + C_2h_F^{5/2}|\theta_F|_{3,\Omega_F}.$$

If the triangulations $\tau_{F,h}$ and $\tau_{S,h}$ match along the interface, then

$$I_2 \leq \left(Ch_F^3\|\hat{d}_F^{\{k\}}\|_{1,\Omega_F} \cdot |\theta_F|_{2,\Omega_F}\right) + \left(Ch_S^3\|\hat{d}_S^{\{k\}}\|_{1,\Omega_S} \cdot |\theta_S|_{2,\Omega_S}\right).$$

Proof. Recall

$$I_2 = \left(k_F \partial_n T_{F,h}^{\{k\}}, \theta_F - \pi_{W_F}\theta_F\right)_{\Gamma_I} - \left(k_S \partial_n T_{S,h}^{\{k\}}, \theta_S - \pi_{W_S}\theta_S\right)_{\Gamma_I}.$$

We add and subtract $(g_{\Gamma_I,N}^{\{k\}}, \theta_S - \pi_{W_S}\theta_S)_{\Gamma_I}$ and $(g_{\Gamma_I,N}^{\{k\}}, \pi_{W_F}\theta_S)_{\Gamma_I}$ to write

$$\begin{aligned} I_2 = &(k_F \partial_n T_{F,h}^{\{k\}} - g_{\Gamma_I,N}^{\{k\}}, \theta_F - \pi_{W_F}\theta_F)_{\Gamma_I} \\ &+ (g_{\Gamma_I,N}^{\{k\}}, \pi_{W_S}\theta_S - \pi_{W_F}\theta_S)_{\Gamma_I} + (g_{\Gamma_I,N}^{\{k\}} - k_S \partial_n T_{S,h}^{\{k\}}, \theta_S - \pi_{W_S}\theta_S)_{\Gamma_I} \end{aligned}$$

To bound the first term, we note that $k_F \partial_n \hat{T}_F^{\{k\}} = g_{\Gamma_I,N}^{\{k\}}$ and apply the Cauchy-Schwarz inequality,

$$|(k_F \partial_n T_{F,h}^{\{k\}} - k_F \partial_n \hat{T}_F^{\{k\}}, \theta_F - \pi_{W_F}\theta_F)_{\Gamma_I}| \leq \|k_F \partial_n T_{F,h}^{\{k\}} - k_F \partial_n \hat{T}_F^{\{k\}}\|_{\Gamma_I} \cdot \|\theta_F - \pi_{W_F}\theta_F\|,$$

followed by a trace theorem, the *a priori* estimate (8.8.9), and an interpolation result to give

$$|(k_F \partial_n T_{F,h}^{\{k\}} - g_{\Gamma_I,N}^{\{k\}}, \theta_F - \pi_{W_F}\theta_F)_{\Gamma_I}| \leq C h_F^3 \|\hat{d}_F^{\{k\}}\|_{1,\Omega_F} \cdot |\theta_F\|_{2,\Omega_F}.$$

The second term is bounded using the Cauchy-Schwarz inequality and an interpolation result.

To bound the third term we apply the Cauchy-Schwarz inequality,

$$|(g_{\Gamma_I,N}^{\{k\}} - k_S \partial_n T_{S,h}^{\{k\}}, \theta_S - \pi_{W_S}\theta_S)_{\Gamma_I}| \leq \|g_{\Gamma_I,N}^{\{k\}} - k_S \partial_n T_{S,h}^{\{k\}}\|_{\Gamma_I} \cdot \|\theta_S - \pi_{W_S}\theta_S\|_{\Gamma_I}.$$

Next, we use that fact that the L^2 projection of $g_{\Gamma_I,N}^{\{k\}}$ onto τ_S along Γ_I is $k_S \partial_n T_{S,h}^{\{k\}}$, along with a trace theorem and an interpolation result to give

$$|(g_{\Gamma_I,N}^{\{k\}} - k_S \partial_n T_{S,h}^{\{k\}}, \theta_S - \pi_{W_S}\theta_S)_{\Gamma_I}| \leq C h_S^3 \|\hat{d}_S^{\{k\}}\|_{1,\Omega_S} \cdot |\theta_S\|_{2,\Omega_S}.$$

□

Lemma 8.8.3. *(Bound on I_3) If the triangulations $\tau_{F,h}$ and $\tau_{S,h}$ do not match along the interface, then*

$$I_3 \leq (\|k_S \partial_n \theta_S\|_{\Gamma_I}) \cdot \left(\|T_{F,h}^{\{k-1\}} - T_{F,h}^{\{k\}}\|_{\Gamma_I} \right) + (\|k_S \partial_n \theta_S\|_{\Gamma_I}) \cdot \left(\|\pi_{W_S} T_{F,h}^{\{k-1\}} - T_{F,h}^{\{k-1\}}\|_{\Gamma_I} \right)$$

where

$$\|\pi_{W_S} T_{F,h}^{\{k-1\}} - T_{F,h}^{\{k-1\}}\|_{\Gamma_I} \leq C_1 h_S^{5/2} |\hat{d}_S^{\{k\}}|_{1,\Omega_S} + C_2 h_F^{5/2} |\hat{d}_F^{\{k\}}|_{1,\Omega_F}.$$

If the triangulations $\tau_{F,h}$ and $\tau_{S,h}$ match along the interface, then

$$I_3 \leq (\|k_S \partial_n \theta_S\|_{\Gamma_I}) \cdot \left(\|T_{F,h}^{\{k-1\}} - T_{F,h}^{\{k\}}\|_{\Gamma_I} \right).$$

Proof. First, observe that $T_{S,h}^{\{k\}} = \pi_{W_S} T_{F,h}^{\{k-1\}}$ and rewrite I_3 by adding and subtracting $(k_S \partial_n \theta_S, T_{F,h}^{\{k-1\}})_{\Gamma_I}$ to get

$$I_3 = (k_S \partial_n \theta_S, \pi_{W_S} T_{F,h}^{\{k-1\}} - T_{F,h}^{\{k-1\}})_{\Gamma_I} + (k_S \partial_n \theta_S, T_{F,h}^{\{k-1\}} - T_{F,h}^{\{k\}})_{\Gamma_I}.$$

To bound the first term we add and subtract $\hat{T}_S^{\{k\}}$ and use the fact that $\pi_{W_S} \hat{T}_S^{\{k\}} = \pi_{W_S} T_{F,h}^{\{k-1\}}$ and $\pi_{W_F} \hat{T}_S^{\{k\}} = T_{F,h}^{\{k-1\}}$ along Γ_I and apply the Cauchy-Schwarz and trace inequalities followed by an interpolation result to give give

$$\|\pi_{W_S} T_{F,h}^{\{k-1\}} - T_{F,h}^{\{k-1\}}\|_{\Gamma_I} \leq C_1 h_S^{5/2} |\hat{d}_S^{\{k\}}|_{1,\Omega_S} + C_2 h_F^{5/2} |\hat{d}_F^{\{k\}}|_{1,\Omega_F}.$$

Observe that $\pi_{W_S} T_{F,h}^{\{k-1\}} = T_{F,h}^{\{k-1\}}$ if the triangulations match along the interface. We apply the Cauchy-Schwarz inequality on the second term to complete the proof. □

Lemma 8.8.4. *(Bound on I_4) If the triangulations $\tau_{F,h}$ and $\tau_{S,h}$ do not match along the interface, then*

$$I_4 \leq \left(\|\chi^{\{k\}} - k_S \partial_n \hat{T}_S^{\{k\}}\|_{\Gamma_I} + \|k_S \partial_n \hat{T}_S^{\{k\}} - \sigma^{\{k\}}\|_{\Gamma_I} \right) \cdot \left(\|\pi_{W_F} \theta_F\|_{\Gamma_I} \right) + \left(\|\sigma^{\{k\}}\|_{\Gamma_I} \right) \cdot \left(\|\pi_{W_F} \theta_F - \pi_{W_S} \theta_S\|_{\Gamma_I} \right)$$

where

$$\|\pi_{W_F} \theta_F - \pi_{W_S} \theta_S\|_{\Gamma_I} \leq C_1 h_S^{5/2} |\theta_S|_{\Omega_S,3} + C_2 h_F^{5/2} |\theta_F|_{\Omega_F,3}.$$

If the triangulations $\tau_{1,h}$ and $\tau_{2,h}$ match along the interface, then

$$I_4 \leq \left(\|\chi^{\{k\}} - k_S \partial_n \hat{T}_S^{\{k\}}\|_{\Gamma_I} + \|k_S \partial_n \hat{T}_S^{\{k\}} - \sigma^{\{k\}}\|_{\Gamma_I} \right) \cdot \left(\|\pi_{W_F} \theta_F\|_{\Gamma_I} \right).$$

Proof. We add and subtract $(\sigma^{\{k\}}, \pi_{W_F} \theta_F)_{\Gamma_I}$ and use $\theta_S = \theta_F$ along Γ_I to get

$$I_4 = (\chi^{\{k\}} - \sigma^{\{k\}}, \pi_{W_F} \theta_F)_{\Gamma_I} + (\sigma^{\{k\}}, \pi_{W_F} \theta_F - \pi_{W_S} \theta_F)_{\Gamma_I}.$$

Observe $\pi_{W_S} \theta_S = \pi_{W_F} \theta_F$ if the triangulations match along the interface. We add and subtract $(k_S \partial_n \hat{T}_S^{\{k\}}, \pi_{W_F} \theta_F)_{\Gamma_I}$ to the first term and apply the Cauchy-Schwarz inequality to complete the proof. □

In practice, the error in the normal derivative is typically the same accuracy as the H^1 error, in this case $\mathcal{O}(h_S^2)$. However, an application of the trace theorem only proves $\mathcal{O}(h_S^{3/2})$ accuracy. This is not an important issue, however, since we intend to use the fact that this term is less accurate than the others, and therefore pollutes the L^2 error. We assume the error in the normal derivative can be bounded,

$$\|k_S \partial_n T_{S,h}^{\{k\}} - k_S \partial_n \hat{T}_S^{\{k\}}\|_{\Gamma_I} \leq C h_S^{\beta} \|\hat{d}_S^{\{k\}}\|_{1,\Omega_S},$$

for $3/2 \leq \beta \leq 2$. The error in the recovered boundary flux can be bounded

$$\|k_S \partial_n \hat{T}_S^{\{k\}} - \sigma^{\{k\}}\|_{\Gamma_I} \leq C S_1 h_S^3 \|\hat{d}_S^{\{k\}}\|_{1,\Omega_S},$$

where S_1 is a stability factor defined by an associated Green's function [35, 56].

Theorem 8.8.1. *Assume the triangulations $\tau_{S,h}$ and $\tau_{F,h}$* do not *match along the interface Γ_I. If $\boldsymbol{u}_h^{\{k\}}$, $p_h^{\{k\}}$, $T_{F,h}^{\{k\}}$ and $T_{S,h}^{\{k\}}$ solve (8.3.3) and (8.3.4) respectively, then the errors $e_{T_S} = T_S - T_{S,h}^{\{k\}}$ and $e_{T_F} = T_F - T_{F,h}^{\{k\}}$ satisfy*

$$\begin{aligned}
\|e_{T_S}\|_{\Omega_S} + \|e_{T_F}\|_{\Omega_F} \leq & \sum_{K \in \tau_{F,h}} \left(C h_K^3 \|\hat{d}_F^{\{k\}}\|_K \cdot (|\phi|_{K,2} + |z|_{K,2} + |\theta_F|_{K,2}) \right) \\
& + \sum_{K \in \tau_{S,h}} \left(C h_K^3 \|\hat{d}_S^{\{k\}}\|_K \cdot |\theta_S|_{K,2} \right) \\
& + \left(C h_F^3 \|g_{\Gamma_I,N}^{\{k\}}\|_{\Omega_F} \cdot |\theta_F|_{2,\Omega_F} \right) + \left(C h_S^3 \|\hat{d}_S^{\{k\}}\|_{\Omega_S} \cdot |\theta_S|_{2,\Omega_S} \right) \\
& + (\|k_S \partial_n \theta_S\|_{\Gamma_I}) \cdot \left(\|T_{F,h}^{\{k-1\}} - T_{F,h}^{\{k\}}\|_{\Gamma_I} \right) \\
& + \left(C h_S^\beta \|\hat{d}_S^{\{k\}}\|_{\Omega_S} + C S_1 h_S^3 \|\hat{d}_S^{\{k\}}\|_{\Omega_S} \right) \cdot (\|\pi_{W_F} \theta_F\|_{\Gamma_I}) \\
& + C_1 h_S^{5/2} \left((S_2 + S_4) |\theta_S|_{\Omega_S,3} + S_3 |\hat{d}_S^{\{k\}}|_{1,\Omega_S} \right) \\
& + C_2 h_F^{5/2} \left((S_2 + S_4) |\theta_F|_{\Omega_F,3} + S_3 |\hat{d}_F^{\{k\}}|_{1,\Omega_F} \right)
\end{aligned}$$

where $\hat{d}_F^{\{k\}}$ and $\hat{d}_S^{\{k\}}$ are defined by (8.8.7) and (8.8.5) respectively, $3/2 \leq \beta \leq 2$, S_1 is a stability factor independent of h_S, $S_2 = \|g_{\Gamma_I,N}^{\{k\}}\|_{\Gamma_I}$, $S_3 = \|k_S \partial_n \theta_S\|_{\Gamma_I}$, and $S_4 = \|\sigma^{\{k\}}\|_{\Gamma_I}$.

Corollary 8.8.1. *Assume the triangulations $\tau_{S,h}$ and $\tau_{F,h}$ match along the interface Γ_I. If $\boldsymbol{u}_h^{\{k\}}$, $p_h^{\{k\}}$, $T_{F,h}^{\{k\}}$ and $T_{S,h}^{\{k\}}$ solve (8.3.3) and (8.3.4) respec-*

tively, then the errors $e_{T_S} = T_S - T_{S,h}^{\{k\}}$ *and* $e_{T_F} = T_F - T_{F,h}^{\{k\}}$ *satisfy*

$$\begin{aligned}\|e_{T_S}\|_{\Omega_S} + \|e_{T_F}\|_{\Omega_F} \leq & \sum_{K\in\tau_{F,h}} \left(Ch_K^3 \|\hat{d}_F^{\{k\}}\|_K \cdot (|\phi|_{K,2} + |z|_{K,2} + |\theta_F|_{K,2})\right) \\ & + \sum_{K\in\tau_{S,h}} \left(Ch_K^3 \|\hat{d}_S^{\{k\}}\|_K \cdot |\theta_S|_{K,2}\right) \\ & + \left(Ch_F^3 \|g_{\Gamma_I,N}^{\{k\}}\|_{\Omega_F} \cdot |\theta_F|_{2,\Omega_F}\right) + \left(Ch_S^3 \|\hat{d}_S^{\{k\}}\|_{\Omega_S} \cdot |\theta_S|_{2,\Omega_S}\right) \\ & + (\|k_S \partial_n \theta_S\|_{\Gamma_I}) \cdot \left(\|T_{F,h}^{\{k-1\}} - T_{F,h}^{\{k\}}\|_{\Gamma_I}\right) \\ & + \left(Ch_S^\beta \|\hat{d}_S^{\{k\}}\|_{\Omega_S} + CS_1 h_S^3 \|\hat{d}_S^{\{k\}}\|_{\Omega_S}\right) \cdot (\|\pi_{W_F}\theta_F\|_{\Gamma_I})\end{aligned}$$

where $\hat{d}_F^{\{k\}}$ *and* $\hat{d}_S^{\{k\}}$ *are defined by (8.8.7) and (8.8.5) respectively,* $3/2 \leq \beta \leq 2$, S_1 *is a stability factor independent of* h.

8.8.2 L^2 error bounds using "boundary element flux"

It is clear that the term containing h_S^β decreases at a slower rate than the other terms. Suppose we solve (8.5.2) rather than (8.3.4), i.e., we pass $\sigma^{\{k\}}$ instead of the finite element flux. This changes the fourth term in the error representation formula to

$$I_4 = (\sigma^{\{k\}}, \pi_{W_F}\theta_F)_{\Gamma_I} - (\sigma^{\{k\}}, \pi_{W_S}\theta_S)_{\Gamma_I}.$$

Theorem 8.8.2. *Assume the triangulations* $\tau_{S,h}$ *and* $\tau_{F,h}$ do not *match along the interface* Γ_I. *If* $\boldsymbol{u}_h^{\{k\}}$, $p_h^{\{k\}}$, $T_{F,h}^{\{k\}}$ *and* $T_{S,h}^{\{k\}}$ *solve (8.3.3) and (8.5.2)*

respectively, then the errors $e_{T_S} = T_S - T_{S,h}^{\{k\}}$ *and* $e_{T_F} = T_F - T_{F,h}^{\{k\}}$ *satisfy*

$$\begin{aligned}\|e_{T_S}\|_{\Omega_S} + \|e_{T_F}\|_{\Omega_F} \leq & \sum_{K\in\tau_{F,h}} \left(Ch_K^3 \|\hat{d}_F^{\{k\}}\|_K \cdot (|\phi|_{K,2} + |z|_{K,2} + |\theta_F|_{K,2})\right) \\ & + \sum_{K\in\tau_{S,h}} \left(Ch_K^3 \|\hat{d}_S^{\{k\}}\|_K \cdot |\theta_S|_{K,2}\right) \\ & + \left(Ch_F^3 \|g_{\Gamma_I,N}^{\{k\}}\|_{\Omega_F} \cdot |\theta_F|_{2,\Omega_F}\right) + \left(Ch_S^3 \|\hat{d}_S^{\{k\}}\|_{\Omega_S} \cdot |\theta_S|_{2,\Omega_S}\right) \\ & + (\|k_S \partial_n \theta_S\|_{\Gamma_I}) \cdot \left(\|T_{F,h}^{\{k-1\}} - T_{F,h}^{\{k\}}\|_{\Gamma_I}\right) \\ & + C_1 h_S^{5/2} \left((S_2 + S_4)|\theta_S|_{\Omega_S,3} + S_3 |\hat{d}_S^{\{k\}}|_{1,\Omega_S}\right) \\ & + C_2 h_F^{5/2} \left((S_2 + S_4)|\theta_F|_{\Omega_F,3} + S_3 |\hat{d}_F^{\{k\}}|_{1,\Omega_F}\right),\end{aligned}$$

where $\hat{d}_F^{\{k\}}$ *and* $\hat{d}_S^{\{k\}}$ *are defined by (8.8.7) and (8.8.5) respectively,* $S_2 = \|k_S \partial_n \hat{T}_F^{\{k\}}\|_{\Gamma_I}$, $S_3 = \|k_S \partial_n \theta_S\|_{\Gamma_I}$, *and* $S_4 = \|\sigma^{\{k\}}\|_{\Gamma_I}$.

Corollary 8.8.2. *Assume the triangulations* $\tau_{S,h}$ *and* $\tau_{F,h}$ *match along the interface* Γ_I. *If* $\boldsymbol{u}_h^{\{k\}}$, $p_h^{\{k\}}$, $T_{F,h}^{\{k\}}$ *and* $T_{S,h}^{\{k\}}$ *solve (8.3.3) and (8.5.2) respectively, then the errors* $e_{T_S} = T_S - T_{S,h}^{\{k\}}$ *and* $e_{T_F} = T_F - T_{F,h}^{\{k\}}$ *satisfy*

$$\begin{aligned}\|e_{T_S}\|_{\Omega_S} + \|e_{T_F}\|_{\Omega_F} \leq & \sum_{K\in\tau_{F,h}} \left(Ch_K^3 \|\hat{d}_F^{\{k\}}\|_K \cdot (|\phi|_{K,2} + |z|_{K,2} + |\theta_F|_{K,2})\right) \\ & + \sum_{K\in\tau_{S,h}} \left(Ch_K^3 \|\hat{d}_S^{\{k\}}\|_K \cdot |\theta_S|_{K,2}\right) \\ & + \left(Ch_F^3 \|g_{\Gamma_I,N}^{\{k\}}\|_{\Omega_F} \cdot |\theta_F|_{2,\Omega_F}\right) + \left(Ch_S^3 \|\hat{d}_S^{\{k\}}\|_{\Omega_S} \cdot |\theta_S|_{2,\Omega_S}\right) \\ & + (\|k_S \partial_n \theta_S\|_{\Gamma_I}) \cdot \left(\|T_{F,h}^{\{k-1\}} - T_{F,h}^{\{k\}}\|_{\Gamma_I}\right)\end{aligned}$$

where $\hat{d}_F^{\{k\}}$ *and* $\hat{d}_S^{\{k\}}$ *are defined by (8.8.7) and (8.8.5) respectively.*

Comparing Theorem 8.8.2 with Theorem 8.8.1 and Corollary 8.8.2 with Corollary 8.8.1, we see that the terms involving h^β have dropped out and the optimal order of convergence has been restored.

Chapter 9

TIME-DEPENDENT INTERFACE TRANSFER

9.1 Introduction

Operator decomposition offers an attractive solution strategy for time-dependent problems where different components of the solution evolve at different time scales. In this chapter, we extend the operator decomposition techniques and error analysis to time dependent problems coupled across an interface. However, the interaction between integration and iteration complicates the analysis. First, we introduce a model problem and define a space-time finite element method. Next, we define several operator decomposition schemes and present analysis for a subset of strategies. Finally, we present numerical results demonstrating the stability and accuracy of these schemes.

9.2 Model problem

Let Ω_1 and Ω_2 be polygonal domains in $\mathbb{R}^2$ with boundaries $\partial\Omega_1$ and $\partial\Omega_2$ intersecting along an interface $\Gamma = \partial\Omega_1 \cap \partial\Omega_2$.

We consider parabolic interface problems of the form

$$\begin{cases} \frac{\partial u_1}{\partial t} + L_1 u_1 = f_1, & (\mathbf{x}, t) \in \Omega_1 \times (0, t_N], \\ u_1(\mathbf{x}, 0) = u_{1,0}, & t = 0, \\ u_1 = 0, & \mathbf{x} \in \partial\Omega_1 \backslash \Gamma, \\ \begin{cases} u_1 = u_2, \\ A_1 \partial_n u_1 = A_2 \partial_n u_2, \end{cases} & \mathbf{x} \in \Gamma \\ \frac{\partial u_2}{\partial t} + L_2 u_2 = f_2, & (\mathbf{x}, t) \in \Omega_2 \times (0, t_N], \\ u_2(\mathbf{x}, 0) = u_{2,0}, & t = 0, \\ u_2 = 0, & \mathbf{x} \in \partial\Omega_2 \backslash \Gamma, \end{cases} \tag{9.2.1}$$

where $L_i u_i = -\nabla \cdot (A_i \nabla u_i) + c_i u_i$, and ∂_n denotes the unit normal derivative directed out of Ω_1, with data f_i, and coefficients $A_i \geq A_{i,0} > 0, \mathbf{b}_i, c_i, u_{i,0}$ sufficiently smooth functions for $i = 1, 2$. We assume the initial values satisfy the boundary conditions and the interface condition.

9.3 A Finite Element Method

Let $T_{1,h}$ and $T_{2,h}$ be quasi-uniform triangulations of Ω_1 and Ω_2 respectively, which do not necessarily match along the interface, and divide the time axis into subintervals $I_n = [t_{n-1}, t_n]$ where $0 < t_1 < t_2 < \ldots < t_N$. Define the piecewise polynomial spaces

$$S_1 = \left\{ v \in C(\Omega_1),\ v \in P^1(K) \text{ for all } K \in T_{1,h} \right\},$$

$$S_2 = \left\{ v \in C(\Omega_2),\ v \in P^1(K) \text{ for all } K \in T_{2,h} \right\},$$

and

$$V_1 = \left\{ v \in S_1,\ v \in L^2(0,T) \mid v \in P^0(I_n)\ \forall n \right\}.$$

$$V_2 = \left\{ v \in S_2,\ v \in L^2(0,T) \mid v \in P^0(I_n)\ \forall n \right\}.$$

We use $V_{1,0}$ to represent the functions in V_1 that are zero along the boundary and the interface. Similarly, $V_{2,0}$ consists of functions in V_2 that are zero on the boundaries excepting the interface. Let g be a smooth function, $\pi_i g$ the projection of g into S_i, and $(\pi_i \rho)g = (\rho \pi_i)g$ the projection into V_i.

The discretization in space results in a system of ordinary differential equations in time. We solve this ODE using the method of lines. Let U^n be the finite element solution at time t_n and approximate the time derivative with the backward difference quotient $(U^n - U^{n-1})/k_n$ where $k_n = t_n - t_{n-1}$. This is called the backward Euler method.

9.4 An Operator Decomposition Method

We solve the model problem iteratively. In Ω_1, we compute Dirichlet condition, $D(U_1, U_2)$, along the interface, which may involve the U_1 and/or

U_2 at a previous time or the current time, and solve for U_1^n. In Ω_2, we compute Neumann condition, $N(U_1, U_2)$, along the interface, which may involve the U_1 and/or U_2 at a previous time or the current time, and solve for U_2^n.

Some of the simplest iterative schemes are:

- The up-tooth method shown in Fig. 9.1. The name come from the fact that it resembles a saw blade facing upwards. This method corresponds to $D(U_1, U_2) = U_2^{n-1}$ and $N(U_1, U_2) = A_1 \partial_n U_1^n$.

- The down-tooth method shown in Fig. 9.2. The name come from the fact that it resembles a saw blade facing downwards. This method corresponds to $D(U_1, U_2) = U_2^n$ and $N(U_1, U_2) = A_1 \partial_n U_1^{n-1}$.

- The x-cross method shown in Fig. 9.3. The name come from the pattern in the figure. This method corresponds to $D(U_1, U_2) = U_2^{n-1}$ and $N(U_1, U_2) = A_1 \partial_n U_1^{n-1}$.

- The subcycled x-cross method shown in Fig. 9.4. This method is similar to the x-cross method, but the information is updated within a time step until convergence. This method corresponds to $D(U_1, U_2) = U_2^{n,\{k\}}$ and $N(U_1, U_2) = A_1 \partial_n U_1^{n,\{k\}}$, where $U_1^{n,\{k\}}$ denotes the solution at time node t_n at the k^{th} iteration. The initial values for the iteration are $U_1^{n,\{0\}} = U_2^{n-1}$ and $A_2 \partial_n U_2^{n,\{0\}} = A_1 \partial_n U_1^{n-1}$.

Formally, we seek $U_1^n \in V_1$ such that

$$\int_{I_n} \left(((U_1^n - U_1^{n-1})/k_n, v) + a_1(U_1^n, v) \right) \, dt = \int_{I_n} (f_1, v) \, dt, \qquad (9.4.1)$$

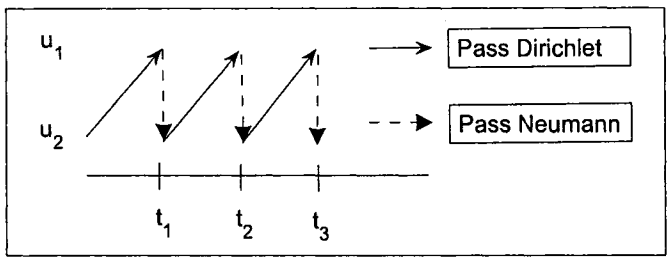

Figure 9.1: The up-tooth iterative method for the parabolic interface problem.

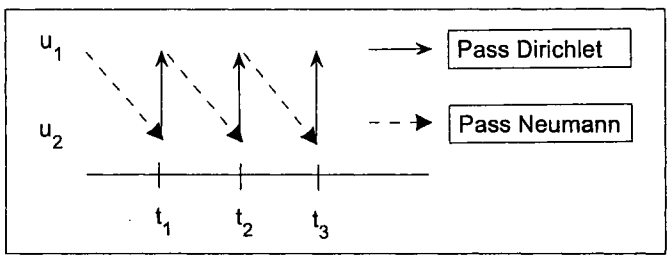

Figure 9.2: The down-tooth iterative method for the parabolic interface problem.

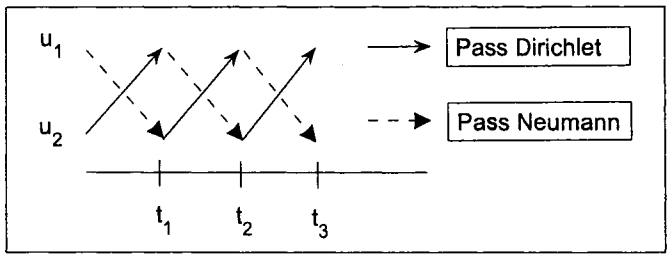

Figure 9.3: The x-cross iterative method for the parabolic interface problem.

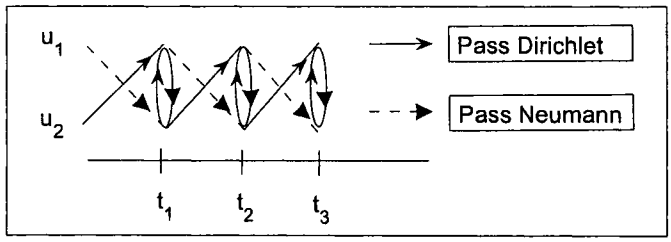

Figure 9.4: The subcycled x-cross iterative method for the parabolic interface problem.

for all $v \in V_{1,0}$ with $U_1^n = D(U_1, U_2)$ along Γ. Next we seek $U_2^n \in V_2$ such that

$$\int_{I_n} \left(((U_2^n - U_2^{n-1})/k_n, v) + a_2(U_2^n, v)\right)\ dt = \int_{I_n} (f_2, v) + (N(U_1, U_2), v)\ dt, \tag{9.4.2}$$

for all $v \in V_{2,0}$.

9.5 The adjoint problem and error representation formula

Consider the model problem (9.2.1). The standard adjoint boundary value problem for the quantity of interest $(\psi_1, u_1(\mathbf{x}, t_N)) + (\psi_2, u_2(\mathbf{x}, t_N))$ is

$$\begin{cases} -\frac{\partial \phi_1}{\partial t} + L_1^* \phi_1 = 0, & (\mathbf{x}, t) \in \Omega_1 \times (t_N, 0], \\ \phi_1(\mathbf{x}, t_N) = \psi_1, & t = t_N, \\ \phi_1 = 0, & \mathbf{x} \in \partial\Omega_1 \backslash \Gamma, \\ \begin{cases} \phi_1 = \phi_2, \\ A_1 \partial_n \phi_1 = A_2 \partial_n \phi_2, \end{cases} & \mathbf{x} \in \Gamma, \\ -\frac{\partial \phi_2}{\partial t} + L_2^* \phi_2 = 0, & (\mathbf{x}, t) \in \Omega_2 \times (t_N, 0], \\ \phi_2(\mathbf{x}, t_N) = \psi_2, & t = t_N, \\ \phi_2 = 0, & \mathbf{x} \in \partial\Omega_2 \backslash \Gamma, \end{cases} \tag{9.5.1}$$

where $L_i^* \phi_i = -\nabla \cdot (A_i \nabla \phi_i) + c_i \phi_i$, and $a_i^*(\cdot, \cdot)$ are the corresponding weak forms for $i = 1, 2$. We derive the error representation formula by combining the techniques used in the previous chapters.

$$
\begin{aligned}
0 =& \int_0^{t_N} \left(\left(-\frac{\partial \phi_1}{\partial t}, e_1 \right) + (L_1^* \phi_1, e_1) \right) \, dt \\
&+ \int_0^{t_N} \left(\left(-\frac{\partial \phi_2}{\partial t}, e_2 \right) + (L_2^* \phi_2, e_2) \right) \, dt \\
=& \sum_{I_n} \int_{I_n} \left(\left(-\frac{\partial \phi_1}{\partial t}, e_1 \right) + a_1^*(\phi_1, e_1) \right) \, dt \\
&+ \int_{I_n} \left(\left(-\frac{\partial \phi_2}{\partial t}, e_2 \right) + a_2^*(\phi_2, e_2) \right) \, dt \\
&+ \sum_{I_n} \int_{I_n} ((A_2 \partial_n \phi_2, e_2)_\Gamma - (A_1 \partial_n \phi_1, e_1)_\Gamma) \, dt \\
=& \sum_{I_n} -(\phi_1, e_1)|_{t_{n-1}}^{t_n} + \int_{I_n} \left(\left(\frac{\partial e_1}{\partial t}, \phi_1 \right) + a_1(e_1, \phi_1) \right) \, dt \\
&+ \sum_{I_n} -(\phi_2, e_2)|_{t_{n-1}}^{t_n} + \int_{I_n} \left(\left(\frac{\partial e_2}{\partial t}, \phi_2 \right) + a_2(e_2, \phi_2) \right) \, dt \\
&+ \sum_{I_n} \int_{I_n} (A_1 \partial_n \phi_1, U_1^n - U_2^n)_\Gamma \, dt
\end{aligned}
$$

We let $[U_i]_{n-1}$ denote the jump term at time t_{n-1} and rearrange some terms,

$$
\begin{aligned}
(\psi_1, e_1^N) + (\psi_2, e_2^N) =& (\phi_1, e_1^0) + \sum_{I_n} (\phi_1, [U_1]_{n-1}) \\
&+ \int_{I_n} \left(\left(\frac{\partial e_1}{\partial t}, \phi_1 \right) + a_1(e_1, \phi_1) \right) \\
&+ (\phi_2, e_2^0) + \sum_{I_n} (\phi_2, [U_2]_{n-1}) + \\
&\int_{I_n} \left(\left(\frac{\partial e_2}{\partial t}, \phi_2 \right) + a_2(e_2, \phi_2) \right) \\
&+ \sum_{I_n} \int_{I_n} (A_1 \partial_n \phi_1, U_1^n - U_2^n)_\Gamma \, dt
\end{aligned}
$$

We observe that the test space $V_{1,0}$ consists of functions that are zero along the interface, while in general, the adjoint solutions are nonzero along the interface. Therefore, we use $\pi_1^0 \phi_1$ as in previous chapters to denote the

projection of ϕ_1 into $V_{1,0}$. This gives

$$\begin{aligned}(\psi_1, e_1^N) + (\psi_2, e_2^N) =&(\phi_1, e_1^0) + \sum_{I_n}(\phi_1 - (\rho\pi_1^0)\phi_1, [U_1]_{n-1}) \\ &+ \int_{I_n}\left(\left(\frac{\partial e_1}{\partial t}, \phi_1 - (\rho\pi_1^0)\phi_1\right) + a_1(e_1, \phi_1 - (\rho\pi_1^0)\phi_1)\right) \\ &+ (\phi_2, e_2^0) + \sum_{I_n}(\phi_2 - (\rho\pi_2)\phi_2, [U_2]_{n-1}) \\ &+ \int_{I_n}\left(\left(\frac{\partial e_2}{\partial t}, \phi_2 - (\rho\pi_2)\phi_2\right) + a_2(e_2, \phi_2 - (\rho\pi_2)\phi_2)\right) \\ &+ \sum_{I_n}\int_{I_n}(A_1\partial_n\phi_1, U_1^n - U_2^n)_\Gamma\, dt \\ &+ \sum_{I_n}\int_{I_n}(A_1\partial_n e_1, (\rho\pi_2)\phi_2)_\Gamma\, dt.\end{aligned}$$

We substitute the data f_1 and f_2 for the true solutions and apply the divergence theorem, taking care to include the necessary boundary terms. We also note that $(U_i)_t = 0$ over each I_n.

$$\begin{aligned}(\psi_1, e_1^N) + (\psi_2, e_2^N) =&(\phi_1, e_1^0) + \sum_{I_n}(\phi_1 - (\rho\pi_1^0)\phi_1, [U_1]_{n-1}) \\ &+ \int_{I_n}\left((f_1, \phi_1 - (\rho\pi_1^0)\phi_1) + a_1(U_1^n, \phi_1 - (\rho\pi_1^0)\phi_1)\right) \\ &+ (\phi_2, e_2^0) + \sum_{I_n}(\phi_2 - (\rho\pi_2)\phi_2, [U_2]_{n-1}) \\ &+ \int_{I_n}\left((f_2, \phi_2 - (\rho\pi_2)\phi_2) + a_2(U_2^n, \phi_2 - (\rho\pi_2)\phi_2)\right) \\ &+ \sum_{I_n}\int_{I_n}(A_1\partial_n\phi_1, U_1^n - U_2^n)_\Gamma\, dt \\ &+ \sum_{I_n}\int_{I_n}(N(U_1, U_2), (\rho\pi_2)\phi_2)_\Gamma\, dt,\end{aligned}$$

where $N(U_1, U_2)$ is the normal derivative information obtained from previously computed values of U_1 and/or U_2.

We isolate the error in the normal derivative by rewriting $\pi_1^0\phi_1 = \pi_1\phi_1 - w\phi_1$ where the projection $w\phi$ has support only near the boundary. This

results in an additional term

$$([U_1]_{n-1}, w\phi_1) + \int_{I_n} ((f_1, w\phi_1) - a_1(U_1^n, w\phi_1))\ dt,$$

over each subinterval I_n.

We define the recovered boundary flux at time t_n to be σ^n satisfying

$$-\int_{I_n} (\sigma, v)_\Gamma\ dt = ([U_1]_{n-1}, v) + \int_{I_n} ((f_1, v) - a_1(U_1^n, v))\ dt,$$

for all suitable test functions, v. As in previous sections, we use the basis function in V_1 as the test and trial functions for σ^n, but we allow σ^n to be discontinuous wherever the boundary has a corner.

We substitute σ^n into the error representation formula and finish with

$$\begin{aligned}
(\psi_1, e_1^N) + (\psi_2, e_2^N) =& (\phi_1, e_1^0) + \sum_{I_n} (\phi_1 - (\rho\pi_1)\phi_1, [U_1]_{n-1}) \\
&+ \int_{I_n} ((f_1, \phi_1 - (\rho\pi_1)\phi_1) + a_1(U_1^n, \phi_1 - (\rho\pi_1)\phi_1)) \\
&+ (\phi_2, e_2^0) + \sum_{I_n} (\phi_2 - (\rho\pi_2)\phi_2, [U_2]_{n-1}) \\
&+ \int_{I_n} ((f_2, \phi_2 - (\rho\pi_2)\phi_2) + a_2(U_2^n, \phi_2 - (\rho\pi_2)\phi_2)) \\
&+ \sum_{I_n} \int_{I_n} (A_1\partial_n\phi_1, U_1^n - U_2^n)_\Gamma\ dt \\
&+ \sum_{I_n} \int_{I_n} ((N(U_1, U_2), (\rho\pi_2)\phi_2)_\Gamma - (\sigma^n, (\rho\pi_1)\phi_1)_\Gamma)\ dt.
\end{aligned}$$

The first four lines can be interpreted as typical contributions to initial, space, and time errors and are easily shown to be $\mathbf{O}(h^2 + k)$. We focus our attention on the last two lines.

The last line represents the difference between the flux we compute and pass, and the recovered boundary flux. Suppose we set $N(U_1, U_2) = A_1\partial_n U_1^n$ as in the up-tooth scheme shown in Fig. 9.1. Let $A_1\partial_n\tilde{u}_1^n$ be the exact normal

derivative given the data from the previous time step. We add and subtract this term to obtain two components

$$\sum_{I_n} \int_{I_n} (A_1 \partial_n U_1^n - A_1 \partial_n \tilde{u}_1^n, (\rho\pi_2)\phi_2)_\Gamma + (A_1 \partial_n \tilde{u}_1^n - \sigma^n, (\rho\pi_2)\phi_2)_\Gamma \, dt.$$

If we insert norms, we see that the first component is $O(h_1)$ at best, while the second component is typically $O(h_1^2)$. The $O(h_1)$ term has a pollution effect and causes the global L^2 error to be $O(h_1)$ as well.

As an alternative, suppose we set $N(U_1, U_2) = \sigma^n$. This leaves

$$\sum_{I_n} \int_{I_n} (\sigma^n, (\rho\pi_2)\phi_2)_\Gamma - (\sigma^n, (\rho\pi_1)\phi_1)_\Gamma) \; dt.$$

If the triangulations match along the interface, i.e. $(\rho\pi_1)\phi_1 = (\rho\pi_2)\phi_2$, then this term vanishes from the error representation formula. If the triangulations do not match along the interface, then we are left with a projection error. We can control this by assuming the meshes are not too badly misaligned.

The other term we are interested in,

$$\sum_{I_n} \int_{I_n} (A_1 \partial_n \phi_1, U_1^n - U_2^n)_\Gamma \, dt,$$

represents a jump term along the interface. Recall that we defined $U_1^n = D(U_1, U_2)$. If we set $D(U_1, U_2) = U_2^{n-1}$ as in Fig. 9.1 or Fig. 9.3, we can rewrite this as

$$\sum_{I_n} \int_{I_n} (A_1 \partial_n \phi_1, U_2^{n-1} - U_2^n)_\Gamma \, dt.$$

This jump term depends on the coefficients A_1 and A_2, the size of each region, and the length of the time step. We suspect that if $A_1 >> A_2$, then the accumulation this jump term will be significant. If this is the case, we may need to consider subcycling within a time step to drive this error below the tolerance.

9.6 Numerical Results

In this section, we present numerical results illustrating the effects of operator decomposition on time-dependent interface problems. The first example addresses the stability of a particular iterative scheme and shows how subcycling allows the use of larger time steps. In the second example, we show that passing the finite element flux introduces an additional error in the solution, and that passing the recovered boundary flux removes this error.

Example 9.6.1. *Let* $\Omega_1 = [0,1] \times [0,1]$ *and* $\Omega_2 = [1,2] \times [0,1]$. *Consider* 9.2.1 *with* $A_1 = 10$, $A_2 = 1$, $c_1 = c_2 = 0$, *and initial conditions* $u_1(\mathbf{x}, 0) = u_2(\mathbf{x}, 0) = 0$. *The boundary conditions and data,* f_1 *and* f_2, *are chosen so the exact solutions are*

$$u_1 = \sin(\pi t)x^3 \sin(2\pi x)\sin(2\pi y),$$

$$u_2 = 10\sin(\pi t)x^3 \sin(2\pi x)\sin(2\pi y).$$

To discretize, we triangulate each of Ω_1 *and* Ω_2 *into 800 elements. These triangulations match along the interface.*

We set $t_N = 0.02$ *and use the x-cross scheme shown in Fig. 9.3 with three different time steps:* $k = 0.001, k = 0.0005, k = 0.0001$. *In Fig. 9.5, we see that the approximation grows rapidly when* $k = 0.001$ *and if we decrease the time step to* $k = 0.0005$, *the approximation grows even more rapidly. This indicates that the x-cross scheme is unstable for large time steps due to an accumulation of errors from the transfer of boundary information. If we decrease the time step to* $k = 0.0001$ *the approximation is stable. This indicates that there is a critical value for the time step, below which the x-cross scheme is stable.*

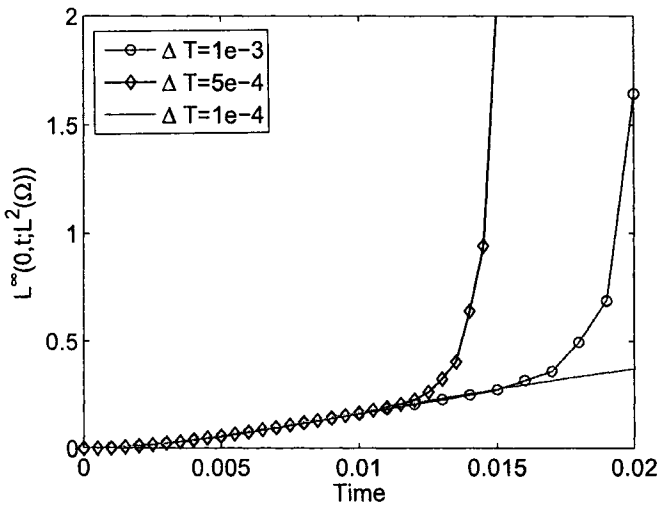

Figure 9.5: Plot of the $L^\infty(0,t;L^2(\Omega))$ errors for the x-cross method in Example 9.6.1 for $k = 0.001$, $k = 0.0005$, and $k = 0.0001$.

Next, we use the subcycled x-cross scheme shown in Fig. 9.4 with $k = 0.05$. *We use a relaxation parameter to force convergence of the Dirichlet values within each time step. In Fig. 9.6, we see that this method allows much larger time steps while avoiding unbounded growth.*

Example 9.6.2. *Let* $\Omega_1 = [0,1] \times [0,1]$ *and* $\Omega_2 = [1,2] \times [0,1]$. *Consider 9.2.1 with* $A_1 = 10$, $A_2 = 10$, $c_1 = c_2 = 0$, *and initial conditions* $u_1(\mathbf{x},0) = u_2(\mathbf{x},0) = 0$. *The boundary conditions and data,* f_1 *and* f_2, *are chosen so the exact solutions are*

$$u_1 = \sin(\pi t)x^3 \sin(2\pi x)\sin(2\pi y),$$

$$u_2 = \sin(\pi t)x^3 \sin(2\pi x)\sin(2\pi y).$$

To discretize, we triangulate each of Ω_1 *and* Ω_2 *into 800 elements. These triangulations match along the interface.*

We set $t_N = 0.5$ *and use the subcycled x-cross scheme shown in Fig. 9.4 with* $k = 0.01$. *We subcycle within a time step until the difference in the*

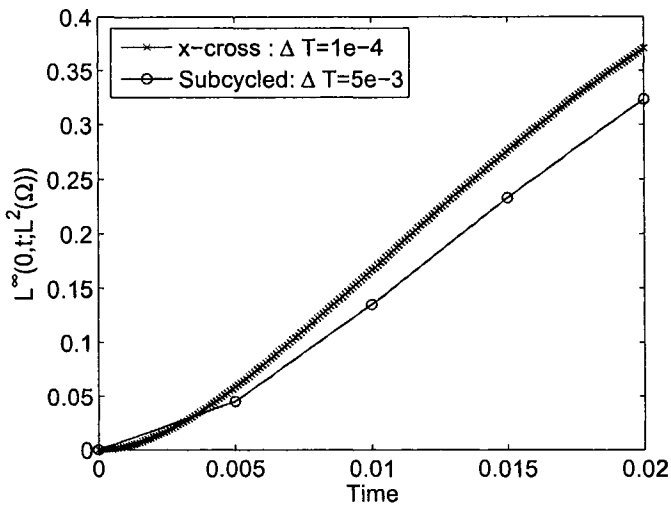

Figure 9.6: Comparison of the $L^\infty(0,t;L^2(\Omega))$ errors in Example 9.6.1 for the x-cross method with $k = 0.0001$, and the subcycled x-cross method with $k = 0.05$.

Dirichlet value is less than 1×10^{-6}. *For comparison, we also compute the fully coupled approximation on the same space-time mesh and plot the* $L^\infty(0,t;L^2(\Omega))$ *in Fig. 9.7. We see that there is an additional error in the operator decomposition solution. Since the subcycling removes iteration error, we suspect that this error is due to passing the finite element flux from* Ω_1 *to* Ω_2. *In Fig. 9.8, we compare the subcycled x-cross scheme and a similar scheme when the recovered boundary flux is passed rather than the finite element flux. Comparing Fig. 9.7 and Fig. 9.8, we see that passing the recovered boundary flux removes the additional error.*

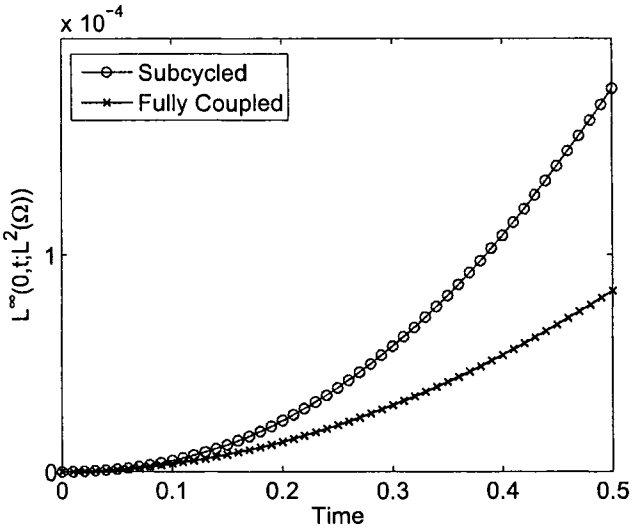

Figure 9.7: Comparison of the $L^{\infty}(0,t;L^2(\Omega))$ errors in Example 9.6.2 for the subcycled x-cross method and the fully coupled scheme with $k = 0.01$.

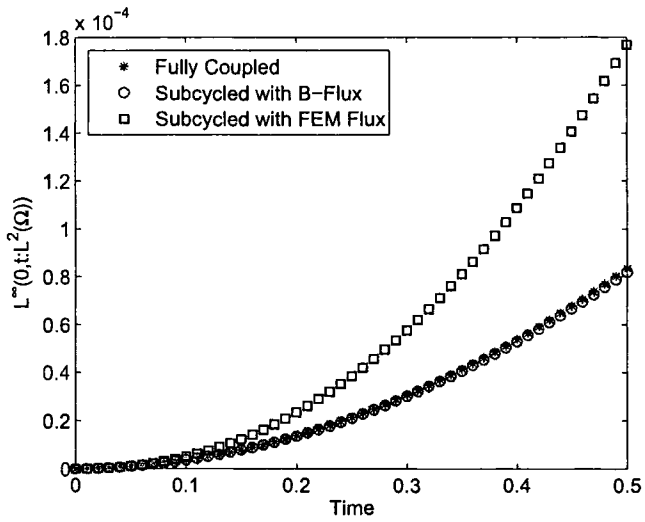

Figure 9.8: Comparison of the $L^{\infty}(0,t;L^2(\Omega))$ errors in Example 9.6.2 for the subcycled x-cross method when the finite element flux is passes and when the recovered boundary flux is passed.

Bibliography

[1] Robert A. Adams. *Sobolev Spaces.* Academic Press, 1975.

[2] Todd Arbogast, Lawrence C. Cowsar, Mary F. Wheeler, and Ivan Yotov. Mixed finite element methods on nonmatching multigrid blocks. *SIAM J. Numer. Anal.*, 37:1295–1315, 2000.

[3] Kendall Atkinson and Weimin Han. *Theoretical Numerical Analysis.* Springer, 2001.

[4] I. Babuska. The finite element method for elliptic equations with discontinuous coefficients. *Computing*, 5:207–213, 1970.

[5] I. Babuska. The finite element method with Lagrange multipliers. *Numer. Math.*, 20:179–192, 1973.

[6] I. Babuska and A. Miller. The post-processing approach in the finite element method-part 1: A posteriori error estimates and adaptive mesh selecton. *Int. J. Numer.Methods Engr.*, 20:2311–2324, 1984.

[7] I. Babuska and A. Miller. The post-processing approach in the finite element method-part 1: Calculation of displacements, stresses and other higher derivatives of the displacement. *Int. J. Numer.Methods Engr.*, 20:1085–1109, 1984.

[8] I. Babuska and A. Miller. The post-processing approach in the finite element method-part 1: The calculation of stress intensity factors. *Int. J. Numer.Methods Engr.*, 20:1111–1129, 1984.

[9] Ivo Babuska and Manil Suri. The p and hp versions of the finite element method, basic principles and properties. *SIAM Review*, 36:578–632, 1994.

[10] Wolfgang Bangerth and Rolf Rannacher. *Adaptive Finite Element Methods for Differential Equations.* Birkhauser Verlag, 2003.

[11] Timothy J. Barth and Mats G. Larson. *A Posteriori* error estimates for higher order Godunov finite volume methods on unstructured meshes. Technical report, NASA, 2002.

[12] R. Becker and R. Rannacher. An optimal control approach to a posteriori error estimation in finite element methods. *Acta Numerica*, pages 1–102, 2001.

[13] Dietrich Braess. *Finite Elements.* Cambridge University Press, 1997.

[14] James H. Bramble and J. Thomas King. A finite element method for interface problems in domains with smooth boundaries and interfaces. *Advances in Comp. Math.*, 67:1–19, 1998.

[15] Susanne C. Brenner and L. Ridgeway Scott. *The Mathematical Theory of Finite Element Methods.* Springer, 2002.

[16] F. Brezzi. On the existence, uniqueness and approximation of saddle point problems arising from Lagrange multipliers. *R.A.I.R.O. Numer. Anal.*, 8:129–151, 1974.

[17] Z. Cai. On the finite volume element method. *Numer. Math*, 58:713–735, 1991.

[18] T. Cao, D.W. Kelly, and M. Ainsworth. Some useful techniques for pointwise and local error estimates of the quantities of interest in the finite element approximation. *ANZIAM*, 42:317–339, 2000.

[19] Yanzhao Cao and Max D. Gunzburger. Least-squares finite element approximations to solutions of interface problems. *SIAM J. Numer. Anal.*, 33:393–405, 1998.

[20] G.F. Carey. Derivative calculation from finite element solutions. *Comp. Meth. in Applied Mech. and Engr.*, 35:1–14, 1982.

[21] V. Carey, D. Estep, and S. Tavener. *A-posteriori* error control of one-way coupled elliptic systems. In preparation, 2006.

[22] P. Chatzipantelidis and R. D. Lazarov. Error estimates for the finite volume element method for elliptic PDE's in nonconvex polygonal domains. *SIAM J. Numer. Anal.*, 42:1932–1958, 2005.

[23] J.-H. Chen, W.G. Pritchard, and S.J. Tavener. Bifurcation for flow past a cylinder between parallel planes. *J. Fluid Mech.*, 284:23–41, 1995.

[24] Zhiming Chen and Jun Zou. Finite element methods and their convergence for elliptic and parabolic interface problems. *Numer. Math*, 79:175–202, 1998.

[25] Sean Eastman. Analysis and application of the nonlinear power method. PhD Thesis, Colorado State University, 2005.

[26] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson. Introduction to adaptive methods for differential equations. *Acta Numerica*, pages 105–158, 1995.

[27] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson. *Computational Differential Equations*. Cambridge University Press, New York, 1996.

[28] D. Estep. A posteriori error bounds and global error control for approximations of ordinary differential equations. *SIAM Journal on Numerical Analysis*, 32:1–48, 1995.

[29] Donald Estep. A short course on duality, adjoint operators, Green's functions, and a-posteriori error analysis. Sandia National Laboratories, Albuquerque, New Mexico, 2004.

[30] Donald Estep, Michael Holst, and Mats Larson. Generalized Green's functions and the effective domain of influence. *SIAM Jour. Sci. Comp.*, 2004.

[31] Donald Estep, Roy Williams, and Mats Larson. Estimating the error of numerical solutions of systems of reaction-diffusion equations. *Memoirs of the American Mathematical Society*, 146, 2000.

[32] Richard E. Ewing, Zhilin Li, Tao Lin, and Yanping Lin. The immersed finite volume element methods for the elliptic interface problem. *Int. J. Math. and Comp. in Simulation*, 1662:1–14, 1999.

[33] Richard E. Ewing, Tao Lin, and Yanping Lin. On the accuracy of the finite volume element method based on piecewise linear polynomials. *SIAM J. Numer. Anal.*, 39:1865–1888, 2002.

[34] M.K. Seager G.F. Carey, S.S. Chow. Approximate boundary-flux calculations. *Comp. Meth. in Applied Mech. and Engr.*, 50:107–120, 1985.

[35] M. Giles, M. G. Larson, J. M. Levenstam, and E. Suli. Adaptive error control for finite element approximations of the lift and drag coefficients in viscous flow. Technical Report NA-97-06, Oxford University Computing Laboratory, 1997.

[36] M. Giles and E. Suli. Adjoint methods for PDE's: *A posteriori* error analysis and postprocessing by duality. *Acta Numerica*, pages 145–236, 2002.

[37] M.B. Giles. Stability analysis of numerical interface boundary conditions in fluid-structure thermal analysis. *International Journal for Numerical Methods in Fluids*, 25:421–436, 1997.

[38] V. Ginting, D. Estep, J. Shadid, and S. Tavener. *A-posteriori* analysis of operator splitting for ordinary differential equations. In preparation, 2006.

[39] Victor Ginting. Analysis of two-scale finite volume element method for elliptic problems. Texas A&M University, 2003.

[40] Vivette Girault and Pierre-Arnoud Raviart. *Finite Element Methods for Navier-Stokes Equations.* Springer-Verlag, 1986.

[41] Max D. Gunzburger. *Finite Element Methods for Viscous Incompressible Flows.* Academic Press, 1989.

[42] V. Heuveline and R. Rannacher. Duality-based adaptivity in the hp-finite element method. *J. Numer. Math*, 0:1–18, 2003.

[43] Huang Jianguo and Xi Shitong. On the finite volume element method for general self-adjoint elliptic problems. *SIAM J. Numer. Anal.*, 35:1762–1774, 1998.

[44] Cornelius Lanczos. *Linear Differential Operators.* Dover Publications, 1997.

[45] Stig Larsson and Vidar Thomee. *Partial Differential Equations with Numerical Methods.* Springer, 2003.

[46] William J. Layton, Friedhelm Schieweck, and Ivan Yotov. Coupling fluid flow with porous media flow. *SIAM J. Numer. Anal.*, 40:2195–2218, 2003.

[47] X.D. Li and N.E. Wiberg. A posteriori error estimate by element patch postprocessing, adaptive analysis in energy norm and L_2 norms. *Comput. and Structures*, 53:907–919, 1994.

[48] X.D. Li and N.E. Wiberg. Superconvergent patch recovery of finite element solution and a posteriori error L_2 norm estimates. *Comm. Numer. Methods Eng.*, 10:313–320, 1994.

[49] Ahmed Naga and Zhimin Zhang. A posteriori error estimates based on the polynomial preserving recovery. *SIAM J. Numer. Anal.*, 42:1780–1800, 2004.

[50] Gergina Penchev and Ivan Yotov. Balancing domain decomposition for mortar mixed finite element methods. *Numer. Linear Algebra Appl.*, 10:159–180, 2003.

[51] Per-Olof Persson and Gilbert Strang. A simple mesh generator in matlab. *SIAM Review*, 46, 2004.

[52] J.R. Rice, P. Tsompanopoulus, and E. Vavalis. Interface relaxation methods for elliptic differential equations. *Applied Numerical Mathematics*, 32:219–245, 2000.

[53] G.E. Schneider. *Handbook of Numerical Heat Transfer*, chapter Elliptic Systems: Finite Element Method I. John Wiley and Sons, 1988.

[54] Barry Smith, Petter Bjorstad, and William Gropp. *Domain Decomposition.* Cambridge University Press, 1994.

[55] M.F. Wheeler. A Galerkin procedure for estimating the flux for two-point boundary-value problems using continuous picewise-polynomial spaces. *Numer. Math*, 2:99–109, 1974.

[56] T. Wildey, D. Estep, and S. Tavener. *A-posteriori* error estimation of boundary flux. In preparation, 2006.

[57] Daoqi Yang. A parallel nonoverlapping schwarz domain decomposition method for elliptic interface problems. *IMA Journal on Numerical Analysis*, 16:75–91, 1996.

[58] Ivan Yotov. A multilevel Newton-Krylov interface solver for multiphysics coupling of flow in porous media. *Numer. Linear Algebra Appl.*, 8:551–570, 2001.

[59] O.C. Zienkiewicz and J.Z. Zhu. The superconvergent patch recovery and a posteriori error estimates. *Intert. J. Numer. Methods Engrg.*, 33:1331–1364 and 1365–1382, 1992.

Appendix A

FINITE VOLUME ELEMENT METHODS

A.1 Formulation

Petrov-Galerkin finite element methods are different from the Galerkin method in that the test and trial spaces are not the same. In fact, the test and trial spaces may not even be subspaces of the same Hilbert space. For this reason, we need to generalize the Lax-Milgram theorem to include bilinear forms mapping $a : U \times V \to \mathbb{R}$ where U and V may be different Hilbert spaces.

Theorem A.1.1. *Let W, V be Hilbert spaces and $a(\cdot,\cdot) : W \times V \to \mathbb{R}$. The variational problem seeking $u \in W$ such that*

$$a(u,v) =< f, v >, \text{ for all } v \in V$$

with $f \in V^*$ has a unique solution if and only if

1. *(Continuity) There exists $C > 0$ such that*

$$|a(u,v)| \leq C\|u\|_W\|v\|_V,$$

   for all $u \in W$ and $v \in V$.

2. *(Inf-sup) There exists $\alpha > 0$ such that*

$$\sup_{v \in V} \frac{a(u,v)}{\|v\|_V} \geq \alpha\|u\|_W,$$

   for all $u \in W$.

3. *For all $v \in V$ there exists $u \in W$ with $a(u,v) \neq 0$.*

Remark A.1.1. The Lax-Milgram theorem follows as a special case of theorem A.1.1. Notice that if $W = V$ then coercivity, $a(u,u) \geq \alpha\|u\|_1^2$ implies the *inf-sup* condition.

A.2 The finite volume element method

The key to a finite volume method is the discrete conservation of certain quantities over a given volume. There are two ways to derive this. The first technique integrates a governing partial differential equation over a given control volume. Consider the nonlinear conservation law

$$\partial_t u + \nabla \cdot \mathbf{f}(u) = 0, \quad x \in \Omega \tag{A.2.1}$$

with initial condition $u(x,0) = u_0(x)$ and appropriate boundary conditions on $\partial\Omega$. Integrating (A.2.1) over a control volume, V, and applying the divergence theorem gives

$$\frac{d}{dt}\int_V u + \int_{\partial V} \mathbf{f}(u)\cdot\mathbf{n}\, dS = 0. \tag{A.2.2}$$

where $\mathbf{n}$ is the outward pointing normal of unit length. The second derivation does not assume a global, pointwise-defined differential equation. Instead, it begins with a physical conservation balance to relate volume integrals with net surface integrals. Observe that (A.2.2) states that the rate of change of a quantity, u, over a control volume V is equal to the total flux of the quantity through the boundary, ∂V. It is straightforward to include a source term on the right hand side of (A.2.2).

Example A.2.1. *Scalar Transport*
Consider a scalar quantity, ϕ, which solves the PDE

$$\partial_t\phi - \nabla\cdot(\alpha\nabla\phi) + \nabla\cdot(\mathbf{b}\phi) = s.$$

The corresponding balance equation for a control volume V is given by

$$\int_V \partial_t\phi\, dV - \int_{\partial V} \alpha\nabla\phi\cdot\mathbf{n}\, dS + \int_{\partial V}\phi\mathbf{b}\cdot\mathbf{n}\, dS = \int_V s\, dV.$$

Example A.2.2. *Conservation of Momentum*
Consider the pointwise conservation equations for momentum in fluid flow

$$\frac{\partial}{\partial t}(\rho u_i) + \frac{\partial}{\partial x_j}(\rho u_j u_i) - \frac{\partial}{\partial x_j}\left[\mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_i}{\partial x_j}\right)\right] + \frac{\partial P}{\partial x_i} = s_i$$

using the Einstein summation notation. The corresponding control volume balance equation is

$$\int_V \frac{\partial}{\partial t}(\rho u_i)\, dV + \int_{\partial V}(\rho u_j u_i)n_j\, dS - \int_{\partial V}\mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)n_j\, dS + \int_{\partial V} P n_i\, dS = \int_V s_i\, dV \tag{A.2.3}$$

Given a triangulation, T_h, there are numerous ways to construct control volumes. Most of these are either cell-centered of vertex-centered finite volume schemes. A cell-centered finite volume scheme uses the elements themselves as control volumes. While this may be the simplest construction, the resulting approximation is usually discontinuous along element interfaces which may not be desirable. A vertex-centered finite volume

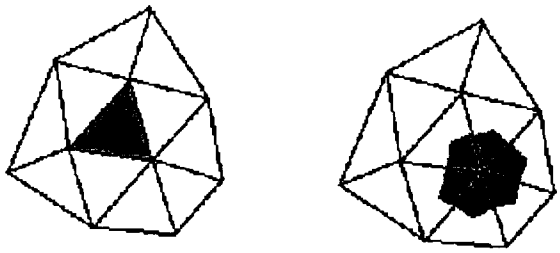

Figure A.1: Example of cell-centered and vertex-centered control volumes.

scheme decomposes each element into smaller sub-control-volumes and associates each of these with a node as shown in Fig. A.1. In this case the collection of these control volumes is often referred to as the dual mesh, T_h'. Nearly all techniques for constructing the dual mesh connect a point in the interior of an element, such as the barycenter, with points on the element edge, eg. the midpoints. Since each node in T_h is associated with a control volume in the dual mesh, this provides a natural projection into the space of continuous piecewise linear polynomials on T_h. This is one of the advantages it holds over traditional cell-centered schemes. The remainder of this survey uses vertex-centered control volumes and we will refer to the resulting method as the finite volume element (FVE) method.

For simplicity, we will focus on discretizing the stationary elliptic problem

$$\begin{cases} -\nabla \cdot (A\nabla u) = f, & \mathbf{x} \in \Omega \\ u = 0, & \mathbf{x} \in \partial\Omega \end{cases} \tag{A.2.4}$$

where $a(x)$ is a smooth function with $a(x) \geq a_0 > 0$ and source term $f(x) \in L^2(\Omega)$. Let T_h be a triangulation of Ω and T_h' the dual mesh with V a control volume. Define

$$S_h = \{v \in C(\Omega) \cap P^1(K),\ \forall K \in T_h\},$$

the space of continuous piecewise linear polynomials on T_h, and the corresponding space

$$S_h^0 = \{v \in S_h \mid v|_{\partial\Omega} = 0\}.$$

The goal of the finite volume method is to find $U \in S_h^0$ which satisfies

$$-\int_{\partial V} A\nabla U \cdot \mathbf{n}\, dS = \int_V f\, dV, \tag{A.2.5}$$

for all control volumes $V \in T_h'$. In general, these integrals must be evaluated using quadrature. The midpoint rule will be used for all surface and volume integrals as depicted in Fig. A.2. The discussion in [53] treats these

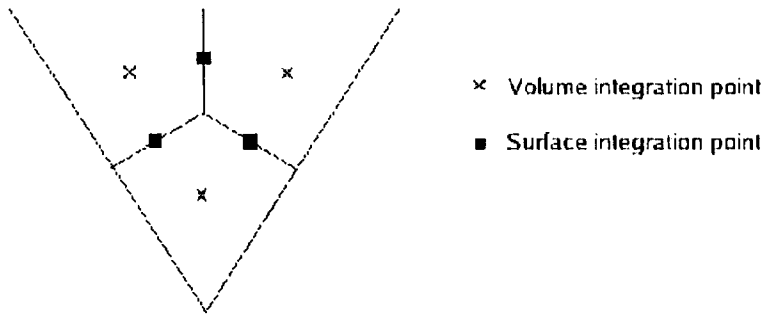

Figure A.2: Midpoint quadrature rules for sub-control volumes and surfaces.

integration points as new variables which must be determined in terms of the nodal variables via integration point operators. The natural way to do this is to use direct interpolation of the Lagrange basis functions. For example, an integration point at the centroid of an element would depend equally on all of the nodal variables. This works well for diffusion dominated problems, but often fails when the problem has strong convection. In this case, it is common to use an upwind method to relate the integration variables to the nodal variables [53].

The trial space for the finite volume problem (A.2.4) is

$$S_h = \{v \in C(\Omega) \mid v \in P^1(K),\ K \in T_h\},$$

and the test space is

$$S_h' = \{v \in L^2(\Omega) \mid v \in P^0(V),\ V \in T_h'\}.$$

We define the projection $\pi' : C(\Omega) \to S_h'$ by

$$\pi' v = \sum_{V \in T_h'} v(x_V)\chi_V,$$

where x_V is the node associated with the control volume V and χ_V is the characteristic function over V. In addition, let $\pi : C(\Omega) \to S_h$ denote the usual nodal interpolant. The FVE method for (A.2.4) finds $U \in S_h$ such that

$$a(U, v) = (f, v), \quad \forall v \in S_h', \tag{A.2.6}$$

with

$$a(u,v) = \sum_{V \in T_h'} - \int_{\partial V} A\nabla u \cdot \mathbf{n}\, v\, dS,$$

and (f,v) the standard L^2 inner product.

Remark A.2.1. The bilinear form in (A.2.6) is actually defined to be

$$a(u,v) = \sum_{V \in T_h'} \int_V A\nabla u \cdot \nabla v\, dV - \int_{\partial V} A\nabla u \cdot \mathbf{n}\, v\, dS.$$

If $v \in S_h'$, then $\nabla v = 0$ and the first term drops out. On the other hand, if v and ∇u are continuous across control volume interfaces, then the second term drops out.

Existence and uniqueness will be shown for (A.2.6) in the discrete spaces S_h and S_h'. First we prove a couple of lemmas for this particular bilinear form.

Lemma A.2.1. *Let v_h and w_h be arbitrary functions in S_h. Then*

$$a(v_h, w_h - \pi' w_h) = \sum_{T \in T_h} \int_{\partial T} A\partial_n v_h (w_h - \pi' w_h) dS.$$

Proof. We use the definition of the bilinear form and apply the divergence theorem over each sub-control volume

$$\begin{aligned}
a(v_h, w_h - \pi' w_h) &= \sum_{V \in T_h'} \int_V A\nabla v_h \cdot \nabla (w_h - \pi' w_h) dx \\
&\quad - \int_{\partial V} A\partial_n v_h (w_h - \pi' w_h) dS \\
&= \sum_{V \in T_h'} \sum_{SCV \in V} \int_{SCV} A\nabla v_h \cdot \nabla (w_h - \pi' w_h) dx \\
&\quad - \int_{\partial V} A\partial_n v_h (w_h - \pi' w_h) dS \\
&= \sum_{V \in T_h'} \sum_{SCV \in V} \int_{SCV} -\nabla \cdot (A\nabla v_h)(w_h - \pi' w_h) dx \\
&\quad + \int_{\partial SCV} A\partial_n v_h (w_h - \pi' w_h) dS \\
&\quad - \int_{\partial V} A\partial_n v_h (w_h - \pi' w_h) dS \\
&= \sum_{K \in T_h} \int_{\partial K} A\partial_n v_h (w_h - \pi' w_h) dS
\end{aligned}$$

The last line follows from the fact that part of the boundary of each sub-control volume is the boundary of an element, K, and part is the boundary of a control volume, V. Those over the control volumes cancel with the boundary terms already present, leaving only the integrals over the boundaries of the elements in T_h.

Lemma A.2.2. *Let v_h and w_h be arbitrary functions in S_h. Assume u is the true solution of (3.3.1) and satisfies (3.1.3), and $A(x) > A_0 > 0$ a smooth function. Furthermore, assume that the control volumes are formed using the barycenter of each element and the midpoint of each edge. Then we have*

$$|a(v_h, w_h - \pi' w_h)| \leq Ch^2 \|w_h\|_1 \|f\|_0.$$

Proof. We have restricted the control volumes to be formed using the barycenter and the midpoint of each edge and because w_h is piecewise linear and $\pi' w_h$ is defined to be the interpolant at the nodes, we have

$$\int_{\partial K} (w_h - \pi' w_h) dS = 0.$$

Since $\partial_n v_h$ is constant on each element we may write

$$\int_{\partial K} A \partial_n v_h (w_h - \pi' w_h) dS = \int_{\partial K} (A - \overline{A}) \partial_n v_h (w_h - \pi' w_h) dS,$$

where $\overline{a}$ is a piecewise constant function defined to be the average value of $a(x)$ over an edge. Now we use the continuity of $\partial_n u$ across element edges to bound

$$\begin{aligned} |a(v_h, w_h - \pi' w_h)| &= \sum_{K \in T_h} \int_{\partial K} (A - \overline{A}) \partial_n (u - v_h)(w_h - \pi' w_h) ds \\ &\leq \sum_{K \in T_h} \|A - \overline{A}\|_{\partial K} \|u - v_h\|_{1,\partial K} \|w_h - \pi' w_h\|_{\partial K} \\ &\leq \sum_{K \in T_h} Ch_K^2 \|A\|_{1,\partial K} \|u - v_h\|_{1,\partial K} \|w_h\|_{1,\partial K} \end{aligned}$$

where we have used Taylor's theorem to estimate

$$\|A - \overline{A}\|_{\partial K} \leq Ch_K \|A\|_{1,\partial K}$$

and the interpolation result

$$\|w_h - \pi' w_h\|_{\partial \Omega} \leq Ch_K \|w_h\|_{1,\partial K}.$$

Now, the fact that $w_h \in C(\Omega)$ is linear over K, implies $\|w_h\|_{1,\partial K} \leq C\|w_h\|_{1,K}$. We use this result and a trace theorem to conclude

$$\begin{aligned} |a(v_h, w_h - \pi' w_h)| &\leq \sum_{K \in T_h} C h_K^2 \|A\|_{2,K} \|u\|_{2,K} \|w_h\|_{1,K} \\ &\leq C h^2 \|w_h\|_1 \|f\|_0. \end{aligned}$$

A similar argument proves the following lemma, which may also be found in [22, 33, 32, 39, 43].

Lemma A.2.3. *Let v_h and w_h be arbitrary functions in S_h and $A(x) \geq A_0 > 0$ a smooth function. Then we may bound*

$$|a(v_h, \pi' w_h)| \leq C h \|v_h\|_1 \|w_h\|_1.$$

We are now in position to prove existence and uniqueness.

Theorem A.2.1. *Existence-Uniqueness*

Let T_h be a quasi-uniform triangulation of Ω, T_h' a dual triangulation, and $a(x) \geq a_0 > 0$. Assume that the bilinear form $a(\cdot, \cdot)$ is coercive over $S_h \times S_h$, i.e. there exists $\alpha > 0$ such that $a(v_h, v_h) \geq \alpha \|v_h\|_1$ for all $v_h \in S_h$. Then the weak problem

$$a(U, v) = < f, v >, \text{ for all } v \in S_h',$$

has a unique solution $U \in S_h$ for any $f \in L^2(\Omega)$.

Proof. We need to show the three conditions in Theorem A.1.1 are satisfied. Continuity of the bilinear form is guaranteed by Lemma A.2.3. The third condition is apparent from the bilinear form. The most difficult to prove is the *inf-sup* condition. From Lemma A.2.1 we have

$$a(U, \pi' U) = a(U, U) - \sum_{K \in T_h} \int_{\partial K} A \partial_n U (U - \pi' U) dS.$$

We have assumed that the bilinear form is coercive over $S_h \times S_h$, so we have

$$\begin{aligned} |a(U, \pi' U)| &= \left| a(U, U) - \sum_{K \in T_h} \int_{\partial K} A \partial_n U (U - \pi' U) dS \right| \\ &\geq |a(U, U)| - \left| \sum_{K \in T_h} \int_{\partial K} A \partial_n U (U - \pi' U) dS \right| \end{aligned}$$

$$\begin{aligned} &\geq \alpha\|U\|_1^2 - \left|\sum_{K\in T_h}\int_{\partial K} A\partial_n U(U-\pi' U)dS\right| \\ &\geq \alpha\|U\|_1^2 - Ch\|U\|_1^2 \end{aligned}$$

Now we choose h small enough so that $\alpha - Ch > 0$.

Intuitively, one would expect the error in the L^2 norm, $\|u-U\|_{L^2(\Omega)}$, to be $O(h^2)$ since U is a piecewise linear approximation. This would be consistent with the piecewise linear interpolant and the standard Galerkin approximation [15]. But we also expect that there should be a trade-off from using piecewise constant test functions rather than piecewise linear functions. The main result in [33] shows that we need to assume some additional smoothness in the source term $f(x)$ to recover second order accuracy.

Lemma A.2.4. *Assume that Ω is a convex polygonal domain and u and U are the solutions to (A.2.4) and (A.2.6) respectively. Further, suppose $u \in H^2(\Omega)$, $f \in H^\beta$ with $0 \leq \beta \leq 1$, $A(x)$ a smooth function with $A \geq A_0 > 0$, and w_h an arbitrary function in S_h. Then there exists a constant $C > 0$ such that*

$$|a(u-U, w_h - \pi' w_h)| \leq Ch\|w_h\|_1 \left(h^\beta \|f\|_\beta + h\|f\|_0\right). \tag{A.2.7}$$

Proof. We use the fact that $a(\cdot,\cdot)$ is linear in the first argument to split

$$a(u-U, w_h - \pi' w_h) = a(u, w_h - \pi' w_h) - a(U, w_h - \pi' w_h),$$

and estimate each term independently. Since u solves (A.2.4) we have

$$a(u, w_h - \pi' w_h) \;=\; (f, w_h - \pi' w_h) \tag{A.2.8}$$

If the control volumes are formed using the barycenter and the midpoints of each edge, then we have

$$\int_K (w_h - \pi' w_h)\, dx = 0,$$

for $K \in T_h$. Note that this only works because w_h is a linear function on K. This allows us to write

$$a(u, w_h - \pi' w_h) = (f - f_K, w_h - \pi' w_h),$$

where f_K is a piecewise constant function equal to the average value of f over each element. We would like to claim $\|w_h - \pi' w_h\|_K \leq Ch_K \|w_h\|_{1,K}$,

but this is not immediately clear since $\pi' w_h \notin H^1(K)$. The approach in [33, 32] uses a discrete norm

$$\|w_h\|_{(1,h),K} = (\pi' w_h, \pi' w_h) + \sum_{x_i \in N_h} \sum_{x_j \in \Pi(i)} \text{meas}(V_i) \left((U(x_i) - U(x_j))/d_{ij}\right)^2,$$

where N_h is the set of nodes in the triangulation, $\Pi(i)$ the set of nodes which share an element with x_i, and d_{ij} the distance between x_i and x_j. This discrete norm is first introduced in [17] where it is shown that $\|w_h - \pi' w_h\| \leq Ch\|w_h\|_{(1,h),K}$, and that the discrete norm is equivalent to the H^1 norm for $w_h \in S_h$. This allows us to bound

$$\begin{aligned} |a(u, w_h - \pi' w_h)| &= |(f - f_K, w_h - \pi' w_h)| \\ &\leq \sum_{K \in T_h} \|f - f_K\|_K \|w_h - \pi' w_h\|_K \\ &\leq \sum_{K \in T_h} Ch^{1+\beta} \|f\|_{\beta,K} \|w_h\|_{1,K} \\ &\leq Ch^{1+\beta} \|f\|_\beta \|w_h\|_1. \end{aligned}$$

This provides a bound for the first term. The second term, $a(U, w_h - \pi' w_h)$, is easy to bound using Lemma A.2.2

$$|a(U, w_h - \pi' w_h)| \leq Ch^2 \|w_h\|_1 \|f\|_0.$$

Theorem A.2.2. *H^1 Error Bound*

Assume that Ω is a convex polygonal domain and u and U are the solutions to (A.2.4) and (A.2.6) respectively. Further, suppose $u \in H^2(\Omega)$, $f \in L^2(\Omega)$ and $A(x)$ a smooth function with $A \geq A_0 > 0$. Then there exists a constant $C > 0$ such that

$$\|u - U\|_1 \leq Ch\|f\|_0. \tag{A.2.9}$$

Proof. This proof resembles the H^1 error bound for the Galerkin finite element method. The key difference is the Galerkin orthogonality, $a(e, v) = 0$, which only holds for $v \in S_h'$. Before beginning, we define $w_h = \pi w - U$ and note that

$$\|w_h\|_1 \leq \|u - U\|_1 + Ch\|u\|_2,$$

using the triangle inequality and an interpolation result. Let C denote a generic constant independent of h.

$$\begin{aligned}
\alpha\|u-U\|_1^2 &\le a(u-U,u-U) && \text{coercivity}\\
&= a(u-U,u-\pi u)\\
&\quad +a(u-U,\pi u-U) && \pm a(u-U,\pi u)\\
&= a(u-U,u-\pi u)\\
&\quad +a(u-U,w_h) && w_h=\pi u-U\\
&= a(u-U,u-\pi u)\\
&\quad +a(u-U,w_h-\pi' w_h) && \text{orthogonality}\\
&\le C\|u-U\|_1\|u-\pi u\|_1 && \text{continuity}\\
&\quad +Ch\|w_h\|_1\left(\|f\|_0+h\|f\|_0\right) && \text{Lemma A.2.4}\\
&\le C\|u-U\|_1\|u-\pi u\|_1+Ch\|w_h\|_1\|f\|_0\\
&\le C\|u-U\|_1\|u-\pi u\|_1\\
&\quad +Ch\left(\|u-U\|_1+Ch\|u\|_2\right)\|f\|_0 && \text{see above}\\
\|u-U\|_1 &\le C\|u-\pi u\|_1+Ch\|f\|_0 && \text{div. } \alpha\|u-U\|_1\\
&\le Ch\|f\|_0
\end{aligned}$$

Theorem A.2.3. *L^2 Error Bound*

Assume that Ω is a convex polygonal domain and u and U are the solutions to (A.2.4) and (A.2.6) respectively. Further, suppose $u\in H^2(\Omega)$, $f\in H^\beta$ with $0\le\beta\le 1$, and $A(x)$ a smooth function with $A\ge A_0>0$. Then there exists a constant $C>0$ such that

$$\|u-U\|_0\le C\left(h^2\|u\|_2+h^{1+\beta}\|f\|_\beta+h^2\|f\|_0\right). \tag{A.2.10}$$

Proof. The proof is similar to the L^2 error bound for the Galerkin finite element method. We let ϕ solve the adjoint problem

$$\begin{cases} -\nabla\cdot(A\nabla\phi)=\psi, & x\in\Omega\\ \phi=0, & x\in\partial\Omega, \end{cases} \tag{A.2.11}$$

and denote $\phi_h=\pi\phi$ to avoid writing $\pi'(\pi\phi)$.

$$\begin{aligned}
\|u-U\|_0^2 &= a^*(\phi,u-U) && \text{weak adjoint}\\
&= a(u-U,\phi) && \text{def. of adjoint}\\
&= a(u-U,\phi-\phi_h)+a(u-U,\phi_h) && \pm a(u-U,w_h)\\
&= a(u-U,\phi-\phi_h)+a(u-U,\phi_h-\pi'\phi_h) && \text{orthog.}\\
&\le C\|u-U\|_1\|\phi-\phi_h\|_1 && \text{continuity}\\
&\quad +Ch\|\phi_h\|_1\left(h^\beta\|f\|_\beta+h\|f\|_0\right) && \text{Lemma A.2.4}\\
&\le Ch^2\|u\|_2\|\phi\|_2 && H^1 \text{ bounds}\\
&\quad +Ch\|\phi_h\|_1\left(h^\beta\|f\|_\beta+Ch\|f\|_0\right)\\
&\le Ch^2\|u\|_2\|u-U\|_0\\
&\quad +Ch\|u-U\|_0\left(h^\beta\|f\|_\beta+h\|f\|_0\right) && \text{regularity}\\
\|u-U\|_0 &\le C\left(h^2\|u\|_2+h^{1+\beta}\|f\|_\beta+h^2\|f\|_0\right) && \text{div. } \|u-U\|_0
\end{aligned}$$

Theorem A.2.4. *a-posteriori L^2 Error Bound*

Assume the bilinear form in (3.3.2) is continuous and coercive, and u satisfies (3.1.3). Then the finite element solution to (3.3.3) satisfies

$$\begin{aligned}\|u-U\|_0 \;\leq\; & \sum_{K\in T_h} CS_K h_K^2 \left(\|f-LU\|_{0,K}+\|f\|_{0,K}\right) \\ & + \sum_{K\in T_h} CS'_K h_K \left(h_K^{\beta}\|f\|_{\beta,K}+h_K\|f\|_{0,K}\right)\end{aligned}$$

where $S_K = \|\phi\|_{2,K}$ and $S'_K = \|\phi_h\|_{1,K}$ are local stability factors.

Proof. Let ϕ solve the adjoint problem (A.2.11) and recall from the previous theorem

$$\|u-U\|_0 = a(u-U,\phi-\phi_h)+a(u-U,\phi_h-\pi'\phi_h),$$

where $\phi_h = \pi\phi$. The first term was estimated in Theorem 3.3.3 where we showed

$$|a(u-U,\phi-\phi_h)| \leq \sum_{K\in T_h} CS_K h_K^2 \left(\|f-LU\|_{0,K}+\|f\|_{0,K}\right).$$

Next, notice that within the proofs of Lemmas A.2.2 and A.2.4 we showed

$$|a(u,\phi_h-\pi'\phi_h)| \leq \sum_{K\in T_h} Ch_K^{1+\beta}\|\phi_h\|_{1,K}\|f\|_{\beta,K},$$

and

$$|a(U,\phi_h-\pi'\phi_h)| \leq \sum_{K\in T_h} CS_K h_K^2\|\phi_h\|_{1,K}\|f\|_{0,K},$$

respectively. Combining these three results gives the desired error bound.

A.3 Estimating a linear functional

Consider the model problem

$$\begin{cases} -\nabla\cdot(A\nabla u) = f, & \mathbf{x}\in(0,1)\times(0,1) \\ u = 0, & \mathbf{x}\in\partial\Omega \end{cases} \tag{A.3.1}$$

with $A = 1+x+y$ and $f(x,y)$ chosen so that the true solution is $u(x,y) = x^2(1-x)y^2(1-y)$.

We let $e = u-U$ and use the adjoint problem

$$\begin{cases} -\nabla\cdot(A\nabla\phi) = \psi, & \mathbf{x}\in(0,1)\times(0,1) \\ \phi = 0, & \mathbf{x}\in\partial\Omega \end{cases} \tag{A.3.2}$$

to estimate the linear functional $< \psi, e >$ with the error representation formula

$$\begin{aligned} < \psi, e > &= a^*(\phi, e) \\ &= a(e, \phi) \\ &= < f, \phi > -a(U, \phi). \end{aligned}$$

In the first example, we take $\psi = 1$ to estimate the average error. We use the piecewise quadratic finite element method to solve the adjoint and display the results in Table A.1.

We also want to compare different methods for solving the adjoint problem. It is clear that any approximation of the adjoint solution, which we denote Φ, cannot lie in the test space, for this would imply $a(e, \Phi) = 0$. On the other hand, Petrov-Galerkin methods produce approximations which are not in the test space. Therefore, it is natural to wonder if we can achieve accurate estimates if we use the same method to solve the primal and adjoint problems. Table A.2 shpws the estimates achieved using the piecewise quadratic finite element method (cG(2)), the piecewise linear finite element method (cG(1)), and the FVE method. We see that the two piecewise linear approximations do not capture enough information from the adjoint problem to provide reliable estimates.

In the second example we want to estimate the error at the point $(0.5, 0.5)$. We use

$$\psi = \frac{400}{\pi} \exp(-400(x - 0.5)^2 - 400(y - 0.5)^2)$$

to approximate a delta function centered at $(0.5, 0.5)$. The results are provided in Table A.3. Note that we have compared the estimates with the exact functional $< \psi, e >$ rather than the actual error at $(0.5, 0.5)$.

Elements	DOF	True Error	Adjoint Est.	Effect. Ratio
192	120	1.2925e-4	1.3796e-4	1.0674
517	293	4.3009e-5	4.2971e-5	0.9991
1229	667	2.0268e-5	2.0262e-5	0.9997
2148	1143	1.1502e-5	1.1500e-5	0.9998

Table A.1: Estimates of the average error using piecewise quadratic finite elements to solve the adjoint problem.

Elements	DOF	Adj. cG(2) Est.	Adj. cG(1) Est.	Adj. FVE Est.
192	120	1.3796e-4	-9.2348e-7	-9.2146e-7
517	293	4.2971e-5	-6.9282e-7	-6.9296e-7
1229	667	2.0262e-5	-1.3175e-7	-1.3178e-7
2148	1143	1.1500e-5	-1.1081e-7	-1.1082e-7

Table A.2: Comparison of using different numerical methods to solve the adjoint.

Elements	DOF	True Fctl. Error	Adjoint Est.	Effect. Ratio
192	120	1.7932e-4	1.7922e-4	0.9994
1229	667	2.6965e-5	2.6958e-5	0.9997
2148	1143	1.4896e-5	1.4894e-5	0.9999
4951	2580	6.3841e-6	6.3837e-6	0.9999

Table A.3: Estimates using the linear functional $\psi = \frac{400}{\pi}\exp(-400(x - 0.5)^2 - 400(y - 0.5)^2)$.

Appendix B

SOFTWARE DOCUMENTATION

B.1 Introduction

ACES (**A**daptive **C**oupled **E**quation **S**olver) is a MATLAB software package designed to solve multiphysics problems, linear or nonlinear, stationary or time-dependent, with a flexible framework allowing the user to apply tight and loose coupling to any component of the differential equation. This includes problems where a variable in one domain is coupled to another variable in a different domain via an interface condition, even if different numerical methods are used in each domain and the discretizations do not align along the interface. ACES also incorporates the adjoint-based error estimation and adaptivity as described in [10, 26, 27, 18, 30, 31, 28].

ACES is a powerful research tool for anyone interested in developing new techniques for solving small to medium scale (¡500,000 unknowns) multiphysics problems. The code is free to download from my webpage:

`http:\\www.math.colostate.edu\~wildey`

Please feel free to modify the code to best meet your own needs. Comments, questions, suggestions, and code contributions can be sent to

`t_wildey@yahoo.com`

B.2 User Manual

To create an application, the user designs an input file which calls the appropriate subroutines. The basic structure of the input file is as follows:

- Geometry: define the mesh and the boundary information
- Physics: define the coefficients
- Initialize: sets up the basis functions and interpolation variables on the mesh
- Solve: compute the approximation
- Post-processing: visualize the results

The main structure used by the code is called *ACES*. In the next section, we describe some common fields associated with this structure.

B.2.1 Common fields in *ACES*

In the following descriptions, N will denote the number of variables, P_n the number of degrees of freedom for the n^{th} variable, M the number of meshes, K_m the number of elements on the m^{th} mesh, T the number of time nodes, and $P_{n,k}$ the number of degrees of freedom for the n^{th} variable on the k^{th} element.

ACES.physics An $N \times N$ structure array containing the coefficients and boundary conditions for the differential equation.

ACES.settings A structure containing the application settings such as the dimension, element shape, degree of the basis functions, quadrature rule, etc.

ACES.mesh A $1 \times M$ structure array containing the nodes and element pointers for each mesh. Also contains a list of the nodes and elements on each boundary as well as the additional nodes that higher order polynomials use.

ACES.elist A $1 \times M$ structure array containing the field *element*, which is a $1 \times K_m$ structure array containing all of the integration information, basis functions, and neighbor information for the elements on that particular mesh.

ACES.basisMaps A structure containing the sparse matrices which map values at the nodes to values at the integration points. These maps are used to assemble the global matrix and to efficiently compute nonlinear coefficients. The sparse matrices stored in

```
ACES.basisMaps(m1,m2).vars(p).qrule(q)
```

map the nodal values for a polynomial of degree p on mesh *m2*, to the integration points for quadrature rule q on mesh *m1*.

ACES.boundaryMaps A structure containing the sparse matrices which map values at the nodes to values at the integration points on the boundary.

ACES.matrixContr An $N \times N$ structure array containing the contributions to each block in the global matrix and the global load vector.

ACES.matrix The sparse global matrix.

ACES.load The global load vector.

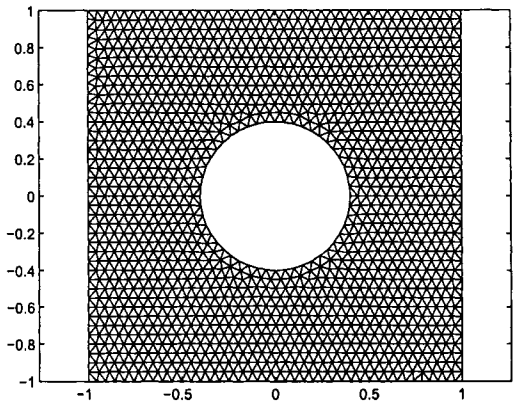

Figure B.1: A sample mesh generated by *distmesh*.

ACES.constraint A $1 \times N$ structure array containing the constraint information, such as boundary constraints, average value constraints, and point constraints, for each variables.

ACES.sol The global solution.

ACES.localsol A $1 \times N$ structure array containing the solution for each variables extracted from the global solution.

ACES.solution A $N \times T$ structure array, where T is the number of time nodes. This contains the fields *sol* and *polysol*. Both are $P_n \times P_{n,k}$ matrices. The field *sol* contains the solution on each element as node values, while *polysol* contains the solution on each element as a polynomial.

B.2.2 Creating a mesh

Any mesh generation software can be used with ACES, as long as it outputs a list of nodes, and a list of pointers (triangles). MATLAB's Delaunay triangulation is an example of code with such output.

The mesh generation program used for most of the benchmark problems is called *distmesh*, by Per-Olof Persson [51] and is freely available on the MATLAB file exchange. This package is capable of creating nearly uniform triangulations of any geometry defined by a set of parametric curves. A demonstration of the capabilities of *distmesh* is provided in distmeshdemo.m.

Example B.2.1. *To create the mesh shown in Fig. B.2.1, we type,*

```
fd=inline('ddiff(drectangle(p,-1,1,-1,1),...
dcircle(p,0,0,0.4))','p');
box=[-1,-1;1,1];
fix=[-1,-1;-1,1;1,-1;1,1];
hsize=0.05;
[p,t]=distmesh2d(fd,@huniform,hsize,box,fix);
```

B.2.3 Defining the boundary information

The boundary information is defined in

ACES.physics(n,n).boundary(j),

where n is the variable number, $1 \leq j \leq M$ and M is the number of boundary segments. If there is only one variable, then the *(n,n)* reference may be dropped.

The code expects

ACES.physics(n,n).boundary(j).location

to be an inline function which evaluates to zero when a point is on the boundary. The field

ACES.physics(n,n).boundary(j).type

is a string giving the type of the boundary condition. The code recognizes 'Dirichlet' or 'D', 'Neumann' or 'N', and 'Robin' or 'R'. The last field in this category is

ACES.physics(n,n).boundary(j).value,

which gives the value of the boundary condition. Typically, this is an inline function or a constant.

Example B.2.2. *Suppose we want to use homogeneous Dirichlet conditions on the unit square.*

```
ACES.physics.boundary(1).location=inline('x-1');
ACES.physics.boundary(2).location=inline('y');
ACES.physics.boundary(3).location=inline('y-1');
ACES.physics.boundary(4).location=inline('x');

ACES.physics.boundary(1).type='Dirichlet';
ACES.physics.boundary(2).type='Dirichlet';
ACES.physics.boundary(3).type='Dirichlet';
ACES.physics.boundary(4).type='Dirichlet';

ACES.physics.boundary(1).value=inline('0');
```

```
ACES.physics.boundary(2).value=inline('0');
ACES.physics.boundary(3).value=inline('0');
ACES.physics.boundary(4).value=inline('0');
```

Example B.2.3. *Suppose we want to define boundary information for the mesh in Fig. B.2.1. The boundary conditions for the box will be inhomogeneous Dirichlet, and for the circle will be homogeneous Neumann.*

```
ACES.physics.boundary(1).location=inline('x-1');
ACES.physics.boundary(2).location=inline('y+1');
ACES.physics.boundary(3).location=inline('y-1');
ACES.physics.boundary(4).location=inline('x+1');
ACES.physics.boundary(5).location=inline('x^2+y^2-0.5^2');

ACES.physics.boundary(1).type='Dirichlet';
ACES.physics.boundary(2).type='Dirichlet';
ACES.physics.boundary(3).type='Dirichlet';
ACES.physics.boundary(4).type='Dirichlet';
ACES.physics.boundary(5).type='Neumann';

ACES.physics.boundary(1).value=inline('sin(pi*y)');
ACES.physics.boundary(2).value=inline('sin(pi*x)');
ACES.physics.boundary(3).value=inline('sin(pi*x)');
ACES.physics.boundary(4).value=inline('sin(pi*y)');
ACES.physics.boundary(5).value=inline('0');
```

Over 40 predefined geometries and boundary conditions are included in the ACES package. Each of these are defined in separate function files within the *geometries* directory. These functions can be accessed directly from the input file with the command

```
ACES=useGeometry('geometry_name',hsize);
```

where *hsize* is the average element size. To add another mesh, type

```
ACES=useGeometry('geometry_name',hsize,ACES);
```

At this point, *ACES.physics(1,1)* contains all of the boundary information for the first mesh, and *ACES.physics(2,2)* has the information for the second.

If more than two variables are to be used, be sure to copy the correct boundary information.

B.2.4 Defining the physics

We think of the physics structure as an $N \times N$ block system of fields, where N is the number of variables. This perspective is particularly useful when dealing with large systems of coupled differential equations. The coefficients are usually defined as inline functions or constants.

Example B.2.4. *Consider the differential equation*

$$-\nabla \cdot ((2+\sin(x))\nabla u) + \left(x^2, 1\right)^T \cdot \nabla u + e^{\sin(x)}u = 1,$$

We would define the physics as

```
ACES.physics.a=inline('2+sin(x)');
ACES.physics.bx=inline('x.^2');
ACES.physics.by=inline('1');
ACES.physics.c=inline('exp(sin(x))');
ACES.physics.f=1;
```

Example B.2.5. *Consider the coupled system*

$$\begin{cases} -\Delta u_1 + 3u_1 + 4u_2 = 1, & x \in \Omega \\ -1.5\Delta u_2 - \nabla u_1 = 0, & x \in \Omega \end{cases}$$

We would define the physics to be

```
ACES.physics(1,1).a=1;
ACES.physics(1,1).c=3;
ACES.physics(1,1).f=1;

ACES.physics(1,2).c=4;

ACES.physics(2,1).bx=-1;
ACES.physics(2,1).by=-1;

ACES.physics(2,2).a=1.5;
ACES.physics(2,2).f=0;
```

B.2.5 Initialization

Initialization is one of the key steps in solving a system of differential equations in ACES. Given a mesh and a number of user defined settings, the function *initialize.m* assembles all of the pre-solve information. This includes

- Determining all of the integration points and weights.

- Creating all of the basis functions.
- Evaluating the basis function (and derivatives) at the appropriate integration points.
- Assembling the neighbor and boundary information.
- Creating sparse matrices mapping nodal values to integration point values.

A list of all of the functions called within *initialize.m* and their purpose is provided in Table B.1. The basic command to initialize the problem is

Function	Purpose
updateSettings	Changes the settings from the defaults
createJacobian	Creates the integration maps and Jacobians
gaussPoints	Maps the integration points to each element.
getIntegrationInfo	Assembles the global integration information.
getEdgeInfo	Gets the neighbor and boundary information.
projectOldSol	Projects an old solution to the current mesh.
basis	Determines the basis functions.
basisVals	Creates the sparse maps.
getBoundaryMaps	Creates the sparse boundary maps.

Table B.1: Functions called within *initialize.m*.

```
ACES=initialize(ACES);
```

although a number of settings can be changed from the input file. A list of the most common settings for *initialize.m* is given in Table B.2. As an

Setting	Purpose	Value	Default
usemesh	mesh for each variable	$1 \times N$ vector	ones(1,N)
degree	degree for each variable	$1 \times N$ vector	ones(1,N)
quadrule	quadrule for each variable	$1 \times N$ vector	ones(1,N)
method	method for each variable	'FE' or 'FV'	'FE'
status	turns statusbar on/off	0=off, 1=on	1
appinfo	output the application info	0=off, 1=on	1

Table B.2: Optional settings for *initialize.m*. N is the number of variables.

example, suppose the problem has five variables, and we want the first, second and fourth variable to be piecwise linear, the third to be piecewise quadratic, and the fifth to be piecewise cubic. If all of the variables use the

same mesh and their respective natural quadrature rules, then the command will be

```
ACES=initialize(ACES,'degree',[1 1 2 1 3]);
```

If all of the variables use the same mesh and the same quadrature rule, then the command will be

```
ACES=initialize(ACES,'degree',[1 1 2 1 3],'quadrule',3);
```

where '3' represents the standard quadrature rule for piecewise cubic functions.

B.2.6 Solving

After defining the geometry, settings the physics and initializing the problem, the next step is solving the problem. This is accomplished by typing

```
ACES=solve(ACES);
```

The optional argumuments are entered as setting-value pairs as previously described for *initialize.m*. A list of the settings available in *solve.m* are provided in Table B.3. Next, we briefly describe the matrix assembly process.

Setting	Purpose	Value	Default
solver	change the solver	GE, NL, TD	GE
whichvars	variables to solve for	$1 \times N_p$ vector	$[1, 2, \ldots, N]$
status	turns statusbar on/off	0=off, 1=on	1
reAssemble	reassemble the pw poly	0=off, 1=on	1
timemethod	time integrator	BWE, FWE, CN	BWE
timetol	inner tolerance	small number	1E-4
timespan	time nodes	$1 \times N_T$ vector	$0 : 0.1 : 1$
reCM	matrix - how often	number	1
reBC	BC's - how often	number	1
staggerSolve	stagger variables	cell	$\{1 : N_p\}$
lumpedmass	lump mass matrix	0=off, 1=on	1
maxNLiter	max. nonlinear iter.	number	25
NLtol	nonlinear tolerance	small number	1E-6

Table B.3: Optional settings for *solve.m*.

We provide the details for computing the stiffness matrix, A, in 2D where

$$A_{ij} = \int_{\Omega} A \nabla v_j \cdot \nabla v_i \, dx.$$

Define N_p and N_q to be the number of degrees of freedom and the number of integration points respectively in the mesh. Let M_x and M_y be the sparse $N_q \times N_p$ matrices mapping the nodal values to the x-derivative values and the y-derivative values at the integration points. Let A_q be a vector containing the value of the diffusion coefficient at the integration points.

The stiffness matrix can be written

$$A = M_x^T D M_x + M_y^T D M_y,$$

where D is an $N_q \times N_q$ sparse diagonal matrix with A_q on the main diagonal.

Since the basis maps are computed and saved in the initialization process, assembling the stiffness matrix requires an evaluation of the diffusion coefficient at the integration points, followed by a few sparse matrix multiplications (even if the problem is nonlinear). In MATLAB, this is much faster than an assembly algorithm using element matrices.

B.2.7 Postprocessing

No finite element software package would be complete without some basic tools for post-processing the results. The main component is usually a visualization tool. In ACES, this function is called *FEMplot.m*. A number of quantities can be plotted for 1D or 2D problems. For example,

```
FEMplot(ACES,'u')
```

plots the first variable,

```
FEMplot(ACES,'u3x')
```

plots the x-derivative of the third variable,

```
FEMplot(ACES,'mesh2')
```

plot the second mesh. For a list of all of the options in FEMplot, type

```
help FEMplot
```

Another common post-processing tool integrates certain quantities over the domain. This is particularly useful to compute linear functionals, or to provide a basis for adaptivity. In ACES, the integration tool is called *integrate.m*. This function uses a high order integration rule to integrate an inline expression. The basic command is

```
[I,ACES]=integrate(ACES,expression)
```

where expression is an inline function which can depend on any of the spatial variables, the finite element solutions, or any of the physics coefficients. See *getFunctionVals.m* for a complete list of options. The value of the global integral is the output I. The value of the integral over the k^{th} element on the M^{th} mesh is saved in *ACES.elist(M).element(k).expression* and can be plotted by typing

```
FEMplot(ACES,'expressionM')
```

where M is actually the mesh number.

B.2.8 Sample input files

Example B.2.6. *Define* $\Omega = (0,1) \times (0,1)$ *and consider the elliptic differential equation,*

$$\begin{cases} -\nabla \cdot (\nabla u) + 10u = (8\pi^2 + 10)\sin(2\pi x)\sin(2\pi y), & x \in \Omega \\ u = 0, & x \in \partial\Omega. \end{cases}$$

with exact solution $u(x,y) = \sin(2\pi x)\sin(2\pi y)$. *We solve this problem using piecewise linear finite elements on a mesh with* $hsize = 0.05$. *The input file is provided below.*

```
%_Define the Geometry_____________________________________

[ACES]=useGeometry('unit_square',0.05);

%_Define_physics__________________________________________

ACES.physics.a=inline('1');
ACES.physics.c=10;
ACES.physics.f=inline('(8*pi^2+10)*sin(2*pi*x)*sin(2*pi*y)');

%_Initialize______________________________________________

[ACES]=initialize(ACES);

%_Solve___________________________________________________

[ACES]=solve(ACES);

%_Post_process____________________________________________

FEMplot(ACES,'u')
```

Example B.2.7. *Define* $\Omega = (0,4) \times (0,1)$ *and consider the Navier-Stokes equations,*

$$\begin{cases} -\nu \Delta \boldsymbol{u} + \rho_0 (\boldsymbol{u} \cdot \nabla) \boldsymbol{u} + \nabla p = 0, \\ -\nabla \cdot \boldsymbol{u} = 0, \end{cases}$$

with $\boldsymbol{u} = (u, v)^T$. *We define the inflow boundary condition*

$$u = y * (1 - y), \quad v = 0,$$

along $x = 0$, *and the outflow boundary condition*

$$\nu \partial_n u = 0, \quad \nu \partial_n v = 0,$$

along $x = 4$, *and set* $u = v = 0$ *along the remaining boundaries. To constrain the pressure, we set* $p(4, 0.5) = 0$.

Below, we provide the input file to solve the problem using the Taylor-Hood finite element pair on a mesh with $hsize = 0.025$. *The viscocity and the density are provided as parameters. We use Newton's method to define the physics coefficients.*

```
%_Define the Geometry_____________________________________

[ACES]=useGeometry('long_rectangle',0.025);

%_Define_physics__________________________________________

ACES.physics(2,2)=ACES.physics(1,1); % Copy the boundary info
ACES.physics(3,3)=ACES.physics(1,1); % Copy the boundary info

ACES.settings.parameter=[0.01 1]; %[\nu \rho_0]

ACES.physics(1,1).a=inline('p1');
ACES.physics(1,1).bx=inline('p2*u');
ACES.physics(1,1).by=inline('p2*u2');
ACES.physics(1,1).c=inline('p2*ux');
ACES.physics(1,1).f=inline('p2*u.*ux+p2*u2.*uy');

ACES.physics(1,2).c=inline('p2*uy');
ACES.physics(1,3).bx=1;

ACES.physics(2,2).a=inline('p1');
ACES.physics(2,2).bx=inline('p2*u');
ACES.physics(2,2).by=inline('p2*u2');
ACES.physics(2,2).c=inline('p2*u2y');
```

```
ACES.physics(2,2).f=inline('p2*u.*u2x+p2*u2.*u2y');

ACES.physics(2,1).c=inline('p2*u2x');
ACES.physics(2,3).by=1;

ACES.physics(3,1).bx=-1;
ACES.physics(3,2).by=-1;
ACES.physics(3,3).f=0;

%_Set the boundary conditions_____________________________
%_Assuming the conditions provided in useGeometry are
%_homogeneous Dirichlet

ACES.physics(1,1).boundary(1).type='Neumann';
ACES.physics(1,1).boundary(4).value=inline('y*(1-y)');
ACES.physics(2,2).boundary(1).type='Neumann';

for j=1:length(ACES.physics(3,3).boundary)
    ACES.physics(3,3).boundary(j).type='none';
end

ACES.constraint(3).point=[4 0.5 0];

%_Initialize______________________________________________

[ACES]=initialize(ACES,'degree',[2 2 1],'quadrule',[2 2 2]);

%_Solve___________________________________________________

[ACES]=solve(ACES,'solver','NL');

%_Post_process____________________________________________

FEMplot(ACES,'flow')
```

B.2.9 The GUI

The ACES package comes with a basic graphical user interface (GUI) designed to run an application without the use of an input file.

The user can easily solve a linear or nonlinear system of equations on a variety of geometries and visualize the results. In addition, the adjoint

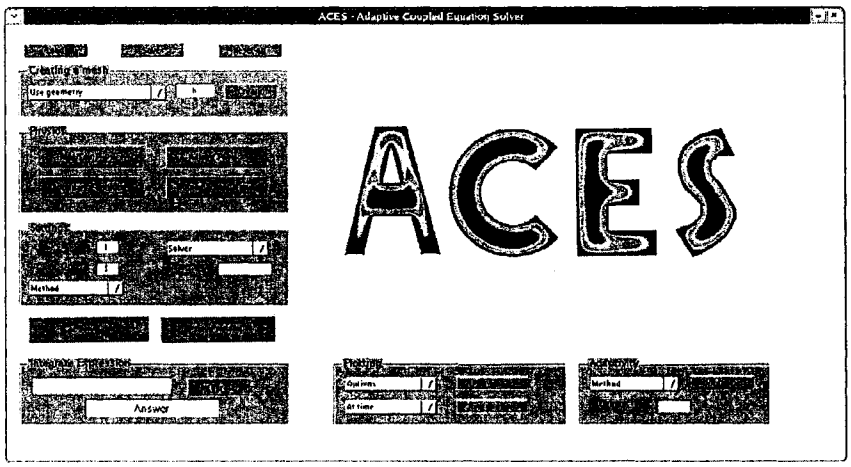

Figure B.2: Layout of the ACES graphical user interface (GUI).

problem can also be approximated and used for *a posteriori* error estimation and adaptivity.

B.3 Benchmark Problems

To demonstrate the accuracy and efficiency of the code, we have put together a series of standard benchmark problems. Some of these problems are well established test problems for finite element or finite volume codes. Others been chosen so that the approximations can be compared to a known analytic solution.

B.3.1 Linear stationary problem

Define $\Omega = (0,1) \times (0,1)$ and consider the elliptic differential equation

$$\begin{cases} -\nabla \cdot (A\nabla u) + \mathbf{b} \cdot \nabla u + cu = f, & x \in \Omega, \\ u = 0, & x \in \partial\Omega \end{cases} \tag{B.3.1}$$

with $A(x,y) = 2 + x^2 + y^2$, $\mathbf{b}(x,y) = (0.5(x^2+y^2), 1.5(x^2+y^2))^T$, $c(x,y) = x^2 + y^2$, and $f(x,y)$ chosen appropriately such that the true solution is

$$u(x) = \sin(2\pi x)\sin(2\pi y).$$

We, use a sequence of uniform meshes to demonstrate the optimal order accuracy in the L^2 norm for the piecewise linear finite element method (cG(1)), the piecewise quadratic finite element method (cG(2)), and the piecwise cubic finite element method (cG(3)). In Table B.3.1 and Fig. B.3, we see that the cG(1) method converges $O(h^2)$, the cG(2) method converges $O(h^3)$, and the cG(3) method converges $O(h^4)$.

Method	$h = 0.1$	$h = 0.05$	$h = 0.025$	Rate
cG(1)	4.838e-2	1.289e-2	3.261e-3	1.95
cG(2)	2.435e-3	3.047e-4	3.829e-5	2.99
cG(3)	1.378e-4	8.364e-6	5.384e-7	4.00

Table B.4: L^2 errors for each method on uniform meshes and the slope of the best fit line on a log-log plot.

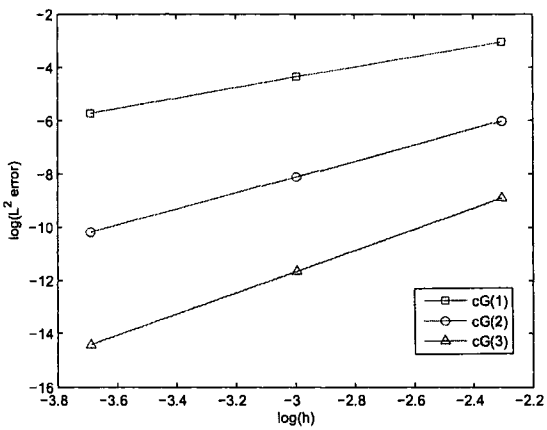

Figure B.3: Convergence rates using piecewise linear elements (cG(1)), piecewise quadratic elements (cG(2)), and piecewise cubic elements (cG(3)).

B.3.2 Nonlinear stationary problem

Define $\Omega = (0,1)\times(0,1)$ and consider the nonlinear stationary problem,

$$\begin{cases} -\nabla \cdot (A\nabla u) + u^4 = f, & x \in \Omega, \\ u = 0, & x \in \partial\Omega \end{cases} \tag{B.3.2}$$

with $A(x,y) = 2$, and

$$f(x,y) = 16\pi^2 \sin(2\pi x)\sin(2\pi y) + (\sin(2\pi x)\sin(2\pi y))^4.$$

The data has been chosen such that the exact solution is

$$u(x) = \sin(2\pi x)\sin(2\pi y).$$

The goal of this example is to show how to define the physics for a successive substitution method and for Newton's method, and to demonstrate the expected convergence of the nonlinear iterations from an initial guess of $U = 0$.

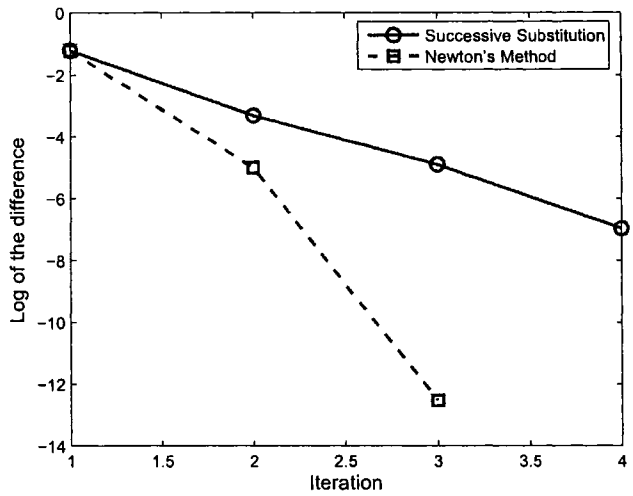

Figure B.4: Plot of iteration number vs. $\log(\|U^k - U^{k-1}\|)$ for (SS) and (NM).

The (SS) method seeks $U^k \in S_h$ such that

$$a(U^k, v) = (f, v) - \left((U^{k-1})^4, v\right), \quad \forall v \in S_h.$$

Recall that the physics coefficients always use the previous value for u. We define the physics to be

```
ACES.physics.a=2;
ACES.physics.f=inline('16*pi^2*sin(2*pi*x).*sin(2*pi*y)+...
(sin(2*pi*x).*sin(2*pi*y)).^4-u.^4');
```

For Newton's method, we seek $U^k \in S_h$ such that

$$a(U^k, v) + \left(4(U^{k-1})^3 U^k, v\right) = (f, v) + \left(3(U^{k-1})^4, v\right) \quad \forall v \in S_h.$$

We define the physics to be

```
ACES.physics.a=2;
ACES.physics.c=inline('4*u.^3');
ACES.physics.f=inline('16*pi^2*sin(2*pi*x).*sin(2*pi*y)+...
(sin(2*pi*x).*sin(2*pi*y)).^4+3*u.^4');
```

The norm of the difference between iterations is given in Fig. B.4. We clearly see that both (NM) and (SS) converge, although (NM) is much faster.

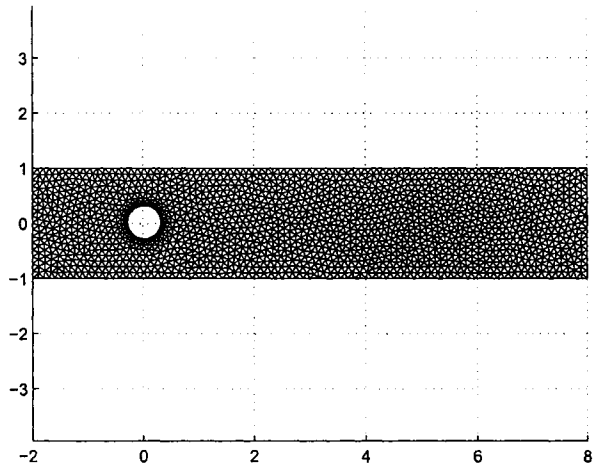

Figure B.5: Domain and mesh used to compute the flow past a cylinder.

B.3.3 Flow past a cylinder

Define the computational domain as shown in Fig. B.5. Consider the Navier-Stokes equations

$$\begin{cases} -\nu\Delta\boldsymbol{u} + \rho_0(\boldsymbol{u}\cdot\nabla)\boldsymbol{u} + \nabla p = 0, \\ -\nabla\cdot\boldsymbol{u} = 0, \end{cases}$$

with no-slip boundary conditions for the velocity except along the inflow we set

$$u_1 = 1 - y^2, \quad u_2 = 0,$$

giving a mean flow of $4/3$, and along the outflow we set

$$\nu\partial_n\boldsymbol{u} = 0, \quad p = 0.$$

The Navier-Stokes equations provide an excellent demonstration of ACES's ability to solve a fully coupled system of nonlinear equations using different degree polynomial spaces for different variable, e.g. quadratic for the velocity and linear for the pressure. In this experiment, we set $\rho_0 = 1$ and decrease the kinematic viscocity, ν, to observe the onset and growth of the recirculation zone behind the cylinder.

In Fig. B.6, we clearly see that for relatively large values of ν there is no recirculation, but as ν decreases the flow becomes convection dominated and the region behind the cylinder begins to show recirculation.

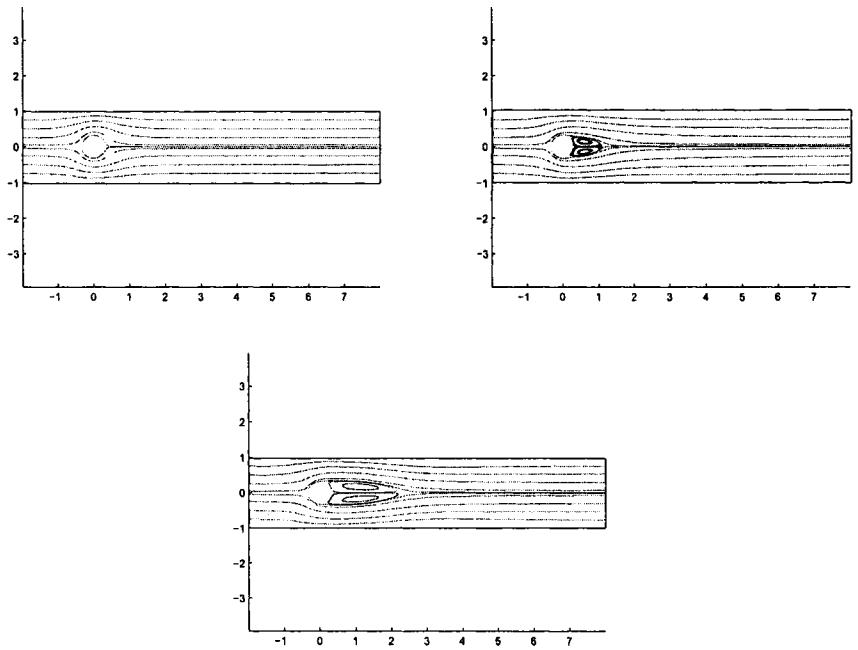

Figure B.6: Streamlines for the flow past a cylinder with $\nu = 1$ (top left), $\nu = 1/100$ (top right), and $\nu = 1/300$ (bottom).

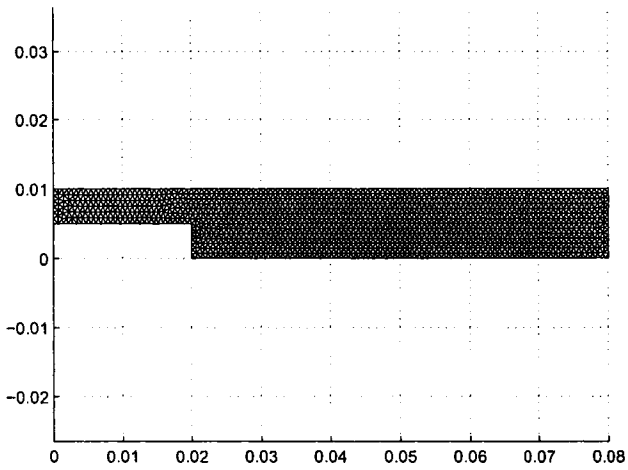

Figure B.7: Computational domain for the backstep benchmark problem.

B.3.4 Back-step

Define the computational domain as shown in Fig. B.7. Consider the Navier-Stokes equations

$$\begin{cases} -\nu\Delta \boldsymbol{u} + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u} + \nabla p = 0, \\ -\nabla\cdot\boldsymbol{u} = 0, \end{cases}$$

with no-slip boundary conditions for the velocity except along the inflow we set

$$u_1 = 43520(y - 0.005)(0.01 - y), \quad u_2 = 0,$$

giving a mean flow of 0.544, and along the outflow we set

$$\nu\partial_n \boldsymbol{u} = 0, \quad p = 0.$$

The purpose of this example is to compare the results from ACES with the results using the commericial software COMSOL Multiphysics.

We compute the average values of the velocity and pressure fields for both approximations and give the results in Table B.3.4. The slight difference in values is mostly due to the fact that the meshes used by the two software programs are not identical.

	ACES	COMSOL
Average x-velocity	2.1760e-4	2.1760e-4
Average y-velocity	-7.5392e-6	-7.5862e-6
Average pressure	-2.7633e-5	-2.7773e-5

Table B.5: Comparison of average values using ACES and the COMSOL Multiphysics package.

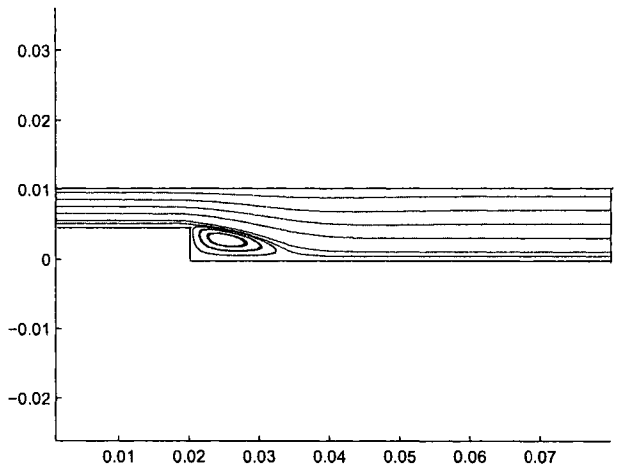

Figure B.8: Streamlines for the backstep problem.