A Short Course on Duality, Adjoint Operators, Green's Functions, and A Posteriori Error Analysis

Donald J. Estep

August 6, 2004

Department of Mathematics
Colorado State University
Fort Collins, CO 80523

estep@math.colostate.edu
http://www.math.colostate.edu/∼estep

Contents

Abstract

Continuous optimization, data assimilation, determining model sensitivity, uncertainty quantification, and *a posteriori* estimation of computational error are fundamentally important problems in mathematical modeling of the physical world. There has been some substantial progress on solving these problems in recent years, and some of these solution techniques are entering mainstream computational science. A powerful framework for tackling all of these problems rests on the notion of duality and an adjoint operator. In the first part of this short course, we will discuss duality, adjoint operators, and Green′s functions; covering both the theoretical underpinnings and practical examples. We will motivate these ideas by explaining the fundamental role of the adjoint operator in the solution of linear problems, working both on the level of linear algebra and differential equations. This will lead in a natural way to the definition of the Green′s function.

In the second part of the course, we will describe how a generalization of the idea of a Green′s function is connected to a powerful technique for a posteriori error analysis of finite element methods. This technique is widely employed to obtain accurate and reliable error estimates in "quantities of interest". We will also discuss the use of these estimates for adaptive error control.

Finally, in the third part of the course, we will describe some applications of these analytic techniques. In the first, we will use the properties of Green′s functions to improve the efficiency of the solution process for an elliptic problem when the goal is to compute multiple quantities of interest and/or to compute quantities of interest that involve globally-supported information such as average values and norms. In the latter case, we introduce a solution decomposition in which we solve a set of problems involving localized information, and then recover the desired information by combining the local solutions. By treating each computation of a quantity of interest independently, the maximum number of elements required to achieve the desired accuracy can be decreased significantly. Time permitting, we will also discuss applications to a posteriori estimation of the effects of operator splitting in a multi-physics problem, estimation of the effect of random variation in parameters in a deterministic model (without using Monte-Carlo), and extensions to nonlinear problems.

The research activities of D. Estep are partially supported by the Department of Energy through grant 90143, the National Aeronautics and Space Administration through grant NNG04GH63G, the National Science Foundation through grants DMS-0107832, DGE-0221595003, and MSPA-CSE-0434354, the Sandia Corporation through contract number PO299784, and the United States Department of Agriculture through contract 58-5402-3-306.

Acknowledgments

The material in this course is collaborative work with a number of people. These include
Sean Eastman, Colorado State University
Michael Holst, University of California at San Diego
Claes Johnson, Chalmers University of Technology
Mats Larson, Umea University
Duane Mikulencak, Georgia Institute of Technology
David Neckels, Colorado State University
Tim Wildey, Colorado State University
Roy Williams, California Institute of Technology

CHAPTER 1

Duality, Adjoint Operators, and Green's Functions

Green's functions are a classic technique for the analysis of differential equations. The definition of the Green's function appears simple at first glance. For example, if u solves

$$\begin{cases} -\Delta u = f, & x \in \Omega, \\ u = 0, & x \in \partial\Omega, \end{cases}$$

where Ω is a domain in $\mathbb{R}^d$ with boundary $\partial\Omega$, the Green's function ϕ satisfies

$$\begin{cases} -\Delta \phi(y;x) = \delta_y(x), & x \in \Omega, \\ \phi(y;x) = 0, & x \in \partial\Omega, \end{cases}$$

where δ_y is the delta function at a point $y \in \Omega$. This gives the

$$\begin{aligned} u(y) = \int_\Omega \delta_y(x) u(x)\, dx &= \int_\Omega -\Delta\phi(y;x) u(x)\, dx \\ &= \int_\Omega \phi(y;x) \cdot -\Delta u(x)\, dx = \int_\Omega \phi(y;x) f(x)\, dx. \end{aligned}$$

or the *function representation formula*

$$u(y) = \int_\Omega \phi(y;x) f(x)\, dx.$$

The simplicity of this argument belies the fact that it depends on some deep mathematics involving the concepts of duality and the adjoint of a linear operator. Since these ideas are crucial to a number of important mathematical constructions, we will begin by discussing them.

1.1. Background in some basic linear algebra

We present a parallel development of ideas for finite dimensional vector spaces and infinite dimensional vector spaces of functions. We will not dwell on technical issues, but we will discuss the important ingredients. So, unfortunately, we have to begin by listing some definitions and concepts.

We will be working on a vector space X with norm $\| \|$. We assume the scalars are real numbers for simplicity. In all cases, the underlying space on which we work has an important property, which depends on the notion of a Cauchy sequence.

DEFINITION 1.1. A sequence $\{x_n\}$ in X is a **Cauchy sequence** if we can make the distance between elements in the sequence arbitrarily small by restricting the indices to be large. More precisely, for every $\epsilon > 0$ there is an N such that $\|x_n - x_m\| < \epsilon$ for all $n, m > N$.

EXAMPLE 1.2. Consider the sequence $\{1/n\}_{n=1}^{\infty}$ in $[0,1]$. This is a Cauchy sequence since

$$\left|\frac{1}{n}-\frac{1}{m}\right| = \left|\frac{m-n}{mn}\right| \leq 2\frac{\max\{m,n\}}{mn} = \frac{2}{\min\{m,n\}}$$

can be made arbitrarily small by taking m and n large. It converges to 0, which is in $[0,1]$.

The notion of a Cauchy sequence is fundamentally important for computational science because it gives a computable way to check a kind of convergent behavior when the limit of a sequence is unknown, which is most of the time. Comparing the distance between two elements in a sequence does not require the limit. This is essentially the motivation for checking how a numerical solution of a differential equation is doing by comparing results on two different discretizations for example. It is not hard to show that a sequence that converges to a limit is a Cauchy sequence. But, the reverse direction, i.e., Cauchy implies convergent, does not automatically hold.

EXAMPLE 1.3. Consider the sequence $\{1/n\}_{n=1}^{\infty}$ in $(0,1)$. While the sequence is a Cauchy sequence, it does not converge to a limit in $(0,1)$, because the limit 0 is not in $(0,1)$.

Spaces in which Cauchy sequences converge are greatly preferred.

DEFINITION 1.4. A **Banach space** is a vector space with a norm such that every Cauchy sequence converges to a limit in the space. We also say the space is **complete**.

EXAMPLE 1.5. The familiar vector space $\mathbb{R}^n$ with the norms defined for $x = (x_1, \cdots, x_n)^\top$,

$$\|x\|_1 = |x_1| + \cdots + |x_n|$$
$$\|x\|_2 = \left(|x_1|^2 + \cdots + |x_n|^2\right)^{1/2}$$
$$\|x\|_\infty = \max |x_i|.$$

are all Banach spaces. We use $\| \ \| = \| \ \|_2$ unless noted otherwise.

There are also Banach spaces of functions.

DEFINITION 1.6. For an interval $[a,b]$, the space of continuous functions is denoted $C([a,b])$, where we take the maximum norm $\|f\| = \max\limits_{a\leq x\leq b} |f(x)|$. We can extend this in a natural way to smoother functions. For example, $C^1([a,b])$ denotes the space of functions that have continuous first derivatives on $[a,b]$, where we use the norm $\|f\| = \max\limits_{a\leq x\leq b} |f(x)| + \max\limits_{a\leq x\leq b} |f'(x)|$.

DEFINITION 1.7. For a domain Ω in $\mathbb{R}^n$ and $1 \leq p \leq \infty$, L^p is the vector space of functions $L^p(\Omega) = \{f : f \text{ is measurable on } \Omega \text{ and } \|f\|_p < \infty\}$, where for $1 \leq p < \infty$,

$$\|f\|_p = \left(\int_\Omega \|f\|^p \, dx\right)^{1/p} \quad \text{and} \quad \|f\|_\infty = \operatorname{ess\,sup}_\Omega \|f\|.$$

L^2 is particularly important.

A key result is

THEOREM 1.8. *The L^p spaces and $C([a,b])$ are Banach spaces.*

EXAMPLE 1.9. The sequence of functions $\{x^n\}_{n=0}^{\infty}$ is a Cauchy sequence in $C([0,1/2])$. Assuming without loss of generality that $n \geq m$, we have for $0 \leq x \leq 1/2$ and any $\epsilon > 0$,

$$|x^n - x^m| = |x^{n-m} - 1| \times |x|^m \leq 1 \times \frac{1}{2^m} < \epsilon$$

for all $m, n \geq N$ provided $N > -\log(\epsilon)/\log(2)$. The sequence converges to the zero function.

EXAMPLE 1.10. The sequence of functions $\{x^n\}_{n=0}^{\infty}$ is not a Cauchy sequence in $C([0,1])$. Assuming that $n \geq m$, we can write $f(x) = |x^n - x^m| = x^m - x^n$ and use Calculus to determine that the maximum value of $f(x)$ occurs at $\bar{x} = (m/n)^{1/(n-m)}$. Using L'Hopital's rule, is easy to show that $\bar{x} \to 1$ as $n \to \infty$ for fixed m (this is also apparent from a graph). The maximum value of f is therefore

$$f(\bar{x}) = \left(1 - \frac{m}{n}\right) \left(\frac{m}{n}\right)^{\frac{m}{n-m}},$$

and both factors tend to 1 as $n \to \infty$ for fixed m.

EXAMPLE 1.11. It is a good exercise to show that $\{x^n\}_{n=0}^{\infty}$ is a Cauchy sequence in $L^1([0,1])$ (this follows because integration of x^n produces a $1/(n+1)$).

The concepts of duality and adjoint operators are intimately tied to linear operators, or maps, on normed vector spaces. We consider maps between two vector spaces X and Y with norms $\| \|_X$ and $\| \|_Y$ respectively.

DEFINITION 1.12. A **map** or **operator** L from X to Y is a rule or association that assigns to each x in X a unique element y in Y. A map $L : X \to Y$ is **linear** if $L(\alpha x_1 + \beta x_2) = \alpha L(x_1) + \beta L(x_2)$ for all numbers α, β and x_1, x_2 in X.

EXAMPLE 1.13. Every linear map from $\mathbb{R}^m$ to $\mathbb{R}^n$ is obtained by multiplying vectors in $\mathbb{R}^m$ by a $n \times m$ matrix, i.e., they have the form Ax, where A is a $n \times m$ matrix. Differentiation is a linear map from $C^1([a,b])$ to $C([a,b])$. Integration is a linear map from $C([a,b])$ into $\mathbb{R}$.

The maps we consider also have to behave continuously,

DEFINITION 1.14. A map $L : X \to Y$ is **continuous** if for every sequence $\{x_n\}$ in X that converges to a limit x in X, i.e., $x_n \to x$, we have $L(x_n) \to L(x)$.

EXAMPLE 1.15. Linear maps from $\mathbb{R}^m$ to $\mathbb{R}^n$ are continuous.

The property given in Def. 1.14 explains why we want continuity, but it is not very easy to check in practice. Luckily, there is an equivalent property for linear maps.

DEFINITION 1.16. A linear map $L : X \to Y$ is **bounded** if there is a constant $C > 0$ such that $\|Lx\|_Y \leq C\|x\|_X$ for all x in X.

EXAMPLE 1.17. Consider integration as a map from $C([a,b])$ into $\mathbb{R}$. Let $f \in C([a,b])$. Then,

$$\left| \int_a^b f(x)\, dx \right| \leq \int_a^b |f(x)|\, dx \leq (b-a) \max_{a \leq x \leq b} |f(x)| = (b-a)\|f\|.$$

We conclude that integration is a bounded map with constant $C = b - a$.

EXAMPLE 1.18. We can define another integration operator from $C([a,b])$ to $C^1([a,b])$ as $I(f)(x) = \int_a^x f(s)\,dx$ for $f \in C([a,b])$ and $a \leq x \leq b$. It is a good exercise to show that I is bounded with constant $C = b - a + 1$.

The equivalence is

THEOREM 1.19. *A linear map between normed vector spaces is continuous if and only if it is bounded.*

By the way, this theorem just says that a linear map is continuous if and only if it is Lipschitz continuous.

It is easy (if tedious) to verify that the set of all linear transformations between two vector spaces X and Y is itself a vector space. We care about the continuous maps.

DEFINITION 1.20. If X and Y are normed vector spaces, we use $\mathcal{L}(X,Y)$ to denote the vector space of all *bounded* linear maps from X to Y. $\mathcal{L}(X,Y)$ is a normed vector space under the **operator norm**

$$\|L\| = \sup_{\|x\|_X=1} \|Lx\|_Y = \sup_{x\neq 0} \frac{\|Lx\|_Y}{\|x\|_X}. \tag{1.1}$$

The operator norm measures the "size" of a linear transformation by the maximum degree by which it can "stretch or shrink" any unit vector. These norms for linear maps from $\mathbb{R}^n$ to $\mathbb{R}^n$ are likely the most familiar examples.

EXAMPLE 1.21. If the linear transformation L is given by the $n \times n$ matrix A, then

$$\|L\|_1 = \|A\|_1 = \max_{1\leq j\leq n} \sum_{i=1}^{n} |a_{ij}|$$

$$\|L\|_2 = \|A\|_2 = \sqrt{\sigma(A^T A)}$$

$$\|L\|_\infty = \|A\|_\infty = \max_{1\leq i\leq n} \sum_{j=1}^{n} |a_{ij}|$$

where $\sigma(A^\top A)$ is the spectral radius of $A^\top A$.

Importantly,

THEOREM 1.22. *If X and Y are normed vector spaces and Y is complete, then $\mathcal{L}(X,Y)$ is complete.*

One way we might generate a sequence of linear operators that converge to a limit is to construct a numerical discretization of some continuous operator, which will then be the limit of the numerical approximations for a sequence of discretization parameters.

1.2. Linear functionals and dual spaces

The concept of duality starts with linear functionals. A linear functional is just a special kind of linear map.

DEFINITION 1.23. A **linear functional** on a vector space X is a linear map from X to $\mathbb{R}$.

EXAMPLE 1.24. Let v in $\mathbb{R}^n$ be fixed. The map $F(x) = v \cdot x = (x, v)$ is a linear functional on $\mathbb{R}^n$.

EXAMPLE 1.25. Consider $C([a, b])$. Both $I(f) = \int_a^b f(x)\,dx$ and $F(f) = f(y)$ for $a \le y \le b$ are linear functionals.

It is useful to think of a linear functional as providing a "low dimensional snapshot" of a vector.

EXAMPLE 1.26. In Example 1.24, consider $v = e_i$, the i^{th} standard basis function. Then $F(x) = x_i$ where $x = (x_1, \cdots, x_n)$. As another example, we can take $v = (1, 1, \cdots, 1)/n$ and compute the average of the components of a given input vector.

EXAMPLE 1.27. Recall that we let δ_y denote the delta function at a point y in a region Ω. This gives a linear functional on sufficiently smooth, real valued functions via

$$F(u) = u(y) = \int_\Omega \delta_y(x) u(x)\,dx.$$

Another linear functional is the average value of an integrable function,

$$F(u) = \frac{1}{\text{vol. of } \Omega} \int_\Omega u(x)\,dx.$$

We can view the formulas defining the Fourier coefficients of a function as a set of linear functionals.

> *Note, there are some important* nonlinear *functionals, such as norms.*

We are interested in the continuous linear functionals. In this case, we define,

DEFINITION 1.28. If X is a normed vector space, the space $\mathcal{L}(X, \mathbb{R})$ of bounded linear functionals on X is called the **dual space of** or **on** or **to** X, and is denoted by X^*. The dual space is a normed vector space under the **dual norm** defined for $y \in X^*$ as

$$\|y\|_{X^*} = \sup_{\substack{x \in X \\ \|x\|_X = 1}} |y(x)| = \sup_{\substack{x \in X \\ x \neq 0}} \frac{|y(x)|}{\|x\|}.$$

EXAMPLE 1.29. Consider $X = \mathbb{R}^n$ with dot product $(\ ,\ )$ and norm $\|\ \| = \|\ \|_2$. In Ex. 1.24, we saw that every vector v in $\mathbb{R}^n$ is associated with a linear functional $F_v(\cdot) = (\cdot, v)$. This functional is clear bounded since $|(x, v)| \le \|v\|\,\|x\|$ (The "C" in the definition is $\|v\|$). A classic result in linear algebra is that *all* linear functionals on $\mathbb{R}^n$ have this form, i.e., we can make the identification $(\mathbb{R}^n)^* \simeq \mathbb{R}^n$.

EXAMPLE 1.30. For $C([a, b])$, consider $I(f) = \int_a^b f(x)\,dx$. It is easy to compute

$$\|I\|_{C([a,b])^*} = \sup_{\substack{f \in C([a,b]) \\ \max |f| = 1}} \left| \int_a^b f(x)\,dx \right|$$

by looking at a picture, see Fig. 1.1. The maximum value for $I(f)$ is clearly given by $f = 1$ or $f = -1$, since if $|f| \le 1$ then $\int |f|\,dx \le \int 1\,dx$, and we get $\|I\|_{C([a,b])^*} = b - a$.

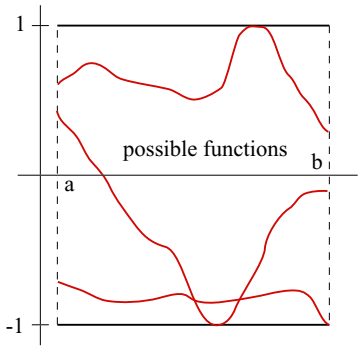

FIGURE 1.1. Computing the dual norm of the integration functional.

EXAMPLE 1.31. Recall Hölder's inequality for $f \in L^p(\Omega)$ and $g \in L^q(\Omega)$ with $\frac{1}{p} + \frac{1}{q} = 1$ for $1 \leq p, q \leq \infty$ is

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

This implies that each g in $L^q(\Omega)$ is associated with a bounded linear functional on $L^p(\Omega)$ when $\frac{1}{p} + \frac{1}{q} = 1$ and $1 \leq p, q \leq \infty$ by

$$F(f) = \int_\Omega g(x) f(x)\, dx.$$

An important, and difficult, result is that we can "identify" $(L^p)^*$ with L^q when $1 < p, q < \infty$, The cases $p = 1, q = \infty$ and $p = \infty, q = 1$ are trickier. The case L^2 is special in that we can identify $(L^2)^*$ with L^2.

Keeping in mind the interpretation of a linear functional as a sample of a vector, the dual space is the collection of "reasonable" possible samples. An important characteristic of a dual space is how much we can reveal about a vector by considering samples in the dual space.

EXAMPLE 1.32. By considering the set of n functionals corresponding to taking the inner product with $\{e_1, \cdots, e_n\}$, we can "reconstruct" any given vector in $\mathbb{R}^n$ by looking at the functional values.

The question of whether or not we can "recover" a vector u completely by computing sufficiently many linear functionals depends heavily on properties of the underlying spaces. In practice, we will often be content with one or just a few "snapshots".

In the same vein, we might wonder about the number of bounded linear functionals that exist on an arbitrary normed vector space. The celebrated Hahn-Banach theorem essentially says there is a great abundance of them. To understand this result, we consider the construction of bounded linear functionals on a Banach space X. Let $x_0 \neq 0$ be a fixed element of X. The set $X_0 = \{\alpha x_0 : \alpha \in \mathbb{R}\}$ forms a vector subspace of X. The linear functional $F(\alpha x_0) = \alpha$ is defined on X_0 and is bounded since $|F(\alpha x_0)| = |\alpha| = \|\alpha x_0\|/\|x\|$. So, we have found a bounded linear functional on a subspace of X. A natural question is whether or not we can extend this to be defined on all of X. This entails defining what "extending" a functional means, and determining if the norm of the functional increases when it is extended.

We also have to consider the possibility that we might have to consider an infinite number of subspaces in order to cover all of X. The Hahn-Banach theorem addresses these points.

THEOREM 1.33. **Hahn-Banach** *Let X be a Banach space and X_0 a subspace of X. Suppose that $F_0(x)$ is a bounded linear functional defined on X_0. There is a linear functional F defined on X such that $F(x) = F_0(x)$ for x in X_0 and $\|F\| = \|F_0\|$.*

We will not discuss the Hahn-Banach theorem in detail, but it is one of the planks in the foundation of this subject.

The dual space is of great value in analysis (with connections to the notions of distributions and weak convergence), but we will not dwell on its uses. One reason that the concept is useful is that the dual space can be better behaved than the original normed vector space. For example,

THEOREM 1.34. *If X is a normed vector space over $\mathbb{R}$, then X^* is a Banach space (whether or not X is a Banach space).*

There is a useful notation for the value of a functional.

DEFINITION 1.35. If x is in X and y is in X^*, we denote the value

$$y(x) =< x, y > .$$

This is called the **bracket notation**.

We finish by noting that norms on X and its dual X^* are closely related. Recall that if $y \in X^*$, then

$$\|y\|_{X^*} = \sup_{\substack{x \in X \\ \|x\|_X = 1}} |y(x)| = \sup_{\substack{x \in X \\ x \neq 0}} \frac{|y(x)|}{\|x\|}.$$

This leads to

DEFINITION 1.36. The **generalized Cauchy inequality** is

$$|< x, y >| \leq \|x\|_X \, \|y\|_{X^*}, \quad x \in X, \, y \in X^*.$$

Combining this with the idea of sampling and the Hahn-Banach theorem yields a "weak" representation of the norm on X.

THEOREM 1.37. *If X is a Banach space, then*

$$\|x\|_X = \sup_{\substack{y \in X^* \\ y \neq 0}} \frac{|y(x)|}{\|y\|_{X^*}} = \sup_{\substack{y \in X^* \\ \|y\|_{X^*} = 1}} |y(x)|$$

for all x in X.

1.3. Hilbert spaces and duality

In Ex. 1.29, we saw that $\mathbb{R}^n$ with the standard Euclidean norm $\| \, \| = \| \, \|_2$ can be identified with its dual space. Likewise, Ex. 1.31 says that L^2 can be identified with its dual space. Both of these spaces are Hilbert spaces.

Recall that one way to get a normed vector space is to place an inner product (i.e., dot product) on the space.

THEOREM 1.38. *If X has an inner product (x, y), then it is a normed vector space with norm $\|x\| = (x, x)^{1/2}$ for x in X.*

DEFINITION 1.39. A vector space with an inner product that is a Banach space with respect to the associated norm is called a **Hilbert** space.

EXAMPLE 1.40. $\mathbb{R}^n$ with $\| \| = \| \|_2$ and L^2 are both Hilbert spaces.

Remarkably, Ex. 1.29 generalizes to infinite dimensions. If X is a Hilbert space with inner product (x, y), then each $y \in X$ determines a linear functional $F_y(x) =< x, y >= (x, y)$ for x in X. This functional is bounded by Cauchy's inequality, which says that $|(x, y)| \leq \|x\| \, \|y\|$. The Riesz Representation theorem says this is the only kind of linear functional on a Hilbert space.

THEOREM 1.41. **Riesz Representation** *For every bounded linear functional F on a Hilbert space X, there is a* unique *element y in X such that*

$$F(x) = (x, y) \text{ for all } x \in X, \text{ and } \|y\|_{X^*} = \sup_{\substack{x \in X \\ x \neq 0}} \frac{|F(x)|}{\|x\|}.$$

This means that a Hilbert space is isometric to its dual space.

DEFINITION 1.42. Two normed vector spaces X and Y are **isometric** if there is a linear 1-1 and onto map $L : X \to Y$ such that $\|L(x)\|_Y = \|x\|_X$ for all x in X.

Abusing notation, it is common to replace the bracket notation and the generalized Cauchy inequality by the inner product and the "real" Cauchy inequality without comment.

The Sobolev spaces are Hilbert spaces based on L^2 that are particularly important for the study of differential equations. The rigorous definition requires the theory of distributions, which we avoid here.

EXAMPLE 1.43. For $k = 1, 2, 3, \cdots$, we define $H^k(\Omega)$ to be the *distribution* functions in $L^2(\Omega)$ whose partial derivatives of order k and less are also distributions in $L^2(\Omega)$. We use the **index notation**. For $\alpha = (\alpha_1, \cdots, \alpha_n)$ with integer coefficients, we define

$$|\alpha| = \alpha_1 + \cdots + \alpha_n$$
$$D = \left(\frac{\partial}{\partial x_1}, \cdots, \frac{\partial}{\partial x_n} \right),$$
$$D^\alpha = \frac{\partial^{|\alpha|}}{\partial x_1^{\alpha_1} \cdots \partial x_n^{\alpha_n}}$$

Then, $H^k(\Omega) = \{u, D^\alpha u \in L^2(\Omega),\ |\alpha| \leq k\}$. The inner products and norms are

$$(u, v)_k = \sum_{|\alpha| \leq k} (D^\alpha u, D^\alpha v), \quad \|u\|_k = (u, u)_k^{1/2}.$$

It turns out that we can extend this definition to fractional indices by a process called interpolation. Perhaps the easiest way to think about this is via Fourier analysis. The Fourier transform of a function in H^k has a specific decay rate depending on k as the Fourier variable tends to infinity. The formulation of this decay rate extends from integer values of k.

The Riesz Representation theorem 1.41 says that every linear functional on H^k has the form $(u, v)_k$ for some fixed v in H^k. But, defining the dual space to

H^k runs into subtle difficulties due to a collision with the requirements for using distribution theory. In particular, for technical reasons having to do with the fact that there are nonzero linear functionals that vanish on all test functions, we define the dual spaces to a subspace

$$H_0^k(\Omega) = \left\{ u \in H^k(\Omega) : u = \frac{\partial u}{\partial n} = \cdots = \frac{\partial^{k-1} u}{\partial n^{k-1}} = 0 \text{ on } \partial\Omega \right\},$$

where $\partial/\partial n$ denotes the normal derivative on the boundary $\partial\Omega$. We let $H^{-k}(\Omega)$ is the set of all linear functionals on $H_0^k(\Omega)$. We will not say more about the reasons for this definition, but it is good to be aware of it.

1.4. Adjoint operators - definition

We now explain how a linear transformation between two normed vector spaces X and Y is naturally associated with another linear transformation between Y^* and X^*. This is the infamous adjoint operator.

It is not difficult to define the adjoint in the context of linear transformations on finite dimensional vectors spaces, where we have access to matrices (the adjoint of a matrix is the transpose). But, we want to give a definition that is independent of dimension.

Suppose $L \in \mathcal{L}(X, Y)$ is a bounded linear transformation. For each $y^* \in Y^*$,

$$y^* \circ L(x) = y^*(L(x)) =< Lx, y^* >$$

assigns a number to each $x \in X$, hence defines a functional $F(x)$. $F(x)$ is clearly linear. It is also bounded since

$$|F(x)| = |y^*(L(x))| \leq \|y^*\|_{Y^*} \|L(x)\|_Y \leq \|y^*\|_{Y^*} \|L\| \, \|x\|_X,$$

where $C = \|y^*\|_{Y^*}\|L\|$ in the definition of boundedness. By the definition of the dual space, there is an $x^* \in X^*$ such that $y^*(L(x)) = x^*(x)$ for all $x \in X$. x^* is unique. Thus, to each $y^* \in Y^*$, we have assigned a unique $x^* \in X^*$ and thus have defined a linear transformation $L^* : Y^* \to X^*$.

We can write these relations as

$$y^*(L(x)) = L^* y^*(x)$$

or using the bracket notation,

$$< L(x), y^* >=< x, L^*(y^*) > \quad x \in X,\, y^* \in Y^*. \tag{1.2}$$

Definition 1.44. Equation (1.2) is called the **bilinear identity** and it serves to define the **adjoint operator** $L^* : Y^* \to X^*$ associated to a bounded linear transformation $L : X \to Y$. It is also called the **dual operator**.

Note that we have defined the adjoint transformation via sampling by elements in the dual space.

In the next example, we explain the relation between the adjoint operator and the transpose of a matrix.

Example 1.45. Let $X = \mathbb{R}^m$ and $Y = \mathbb{R}^n$, where we take the standard inner product and norm. By the Riesz Representation theorem, the bilinear identity for $L \in \mathcal{L}(\mathbb{R}^m, \mathbb{R}^n)$ reads

$$(Lx, y) = (x, L^* y), \quad x \in \mathbb{R}^m,\, y \in \mathbb{R}^n.$$

We know that L is represented by a unique $n \times m$ matrix A so that if $y = L(x)$ then $y = Ax$ where

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nm} \end{pmatrix}, \quad y = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}, \quad x = \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix}$$

and

$$y_i = \sum_{j=1}^{m} a_{ij} x_j, \quad 1 \le i \le n.$$

For a linear functional $y^* = (y_1^*, \cdots, y_n^*)^\top \in Y^*$, we have

$$\begin{aligned} L^* y^*(x) = y^*(L(x)) &= \left((y_1^*, \cdots, y_n^*), \begin{pmatrix} \sum_{j=1}^m a_{1j} x_j \\ \vdots \\ \sum_{j=1}^m a_{nj} x_j \end{pmatrix} \right) \\ &= \sum_{j=1}^{m} y_1^* a_{1j} x_j + \cdots \sum_{j=1}^{m} y_n^* a_{nj} x_j \\ &= \sum_{j=1}^{m} (\sum_{i=1}^{n} y_i^* a_{ij}) x_j \end{aligned}$$

Therefore, $L^*(y^*)$ is given by the inner product with $\tilde{y} = (\tilde{y}_1, \cdots, \tilde{y}_m)^\top$ where

$$\tilde{y}_j = \sum_{i=1}^{n} y_i^* a_{ij}.$$

This implies the matrix A^* of L^* is

$$A^* = \begin{pmatrix} a_{11}^* & \cdots & a_{1n}^* \\ \vdots & & \vdots \\ a_{m1}^* & \cdots & a_{mn}^* \end{pmatrix} = \begin{pmatrix} a_{11} & a_{21} & \cdots & a_{n1} \\ \vdots & & & \vdots \\ a_{1m} & a_{2m} & \cdots & a_{nm} \end{pmatrix} = A^\top.$$

We can write the bilinear identity as

$$y^\top A x = x^\top A^\top y$$

using the fact that $(x, y) = (y, x)$.

The following theorem is crucial.

THEOREM 1.46. *$L^* \in \mathcal{L}(Y^*, X^*)$ and $\|L^*\| = \|L\|$.*

PROOF. The linearity is easy. We have already shown that

$$|L^* y^*(x)| \le \|y^*\|_{Y^*} \|L\| \|x\|_X.$$

Therefore,

$$\|L^* y^*\|_{X^*} = \sup_{x \neq 0} \frac{|L^* y^*(x)|}{\|x\|} \le \|y^*\|_{Y^*} \|L\|,$$

which implies that $L^* \in \mathcal{L}(Y^*, X^*)$ and $\|L^*\| \le \|L\|$. To show the reverse inequality, we prove that

$$\|Lx\|_Y \le \|L^*\| \, \|x\|_X, \quad x \in X.$$

The bilinear identity implies that

$$|y^*(Lx)| \le \|L^* y^*\|_{X^*} \|x\|_X \le \|L^*\| \|y^*\|_{Y^*} \|x\|_X$$

and

$$\sup_{y^* \neq 0} \frac{|y^*(Lx)|}{\|y^*\|_{Y^*}} \leq \|L^*\| \, \|x\|_X, \quad x \in X.$$

□

The adjoint operator has some properties that are easily verified.

THEOREM 1.47. *Let X, Y, and Z be normed linear spaces. Then,*

$$0^* = 0$$
$$(L_1 + L_2)^* = L_1^* + L_2^*, \quad \textit{all } L_1, L_2 \in \mathcal{L}(X, Y)$$
$$(\alpha L)^* = \alpha L^*, \quad \textit{all } \alpha \in \mathbb{R},\ L \in \mathcal{L}(X, Y)$$

If $L_2 \in \mathcal{L}(X, Y)$ and $L_1 \in \mathcal{L}(Y, Z)$, then $L_1 L_2 \in \mathcal{L}(X, Z)$, $(L_1 L_2)^* \in \mathcal{L}(Z^*, X^*)$, and

$$(L_1 L_2)^* = L_2^* L_1^*.$$

In a bit, we will discuss the computation of the adjoint when we are considering differential operators acting on function spaces. But, we first conclude this section by a discussing a technical issue that is relevant in that case. Namely, when talking about differential operators on function spaces, we are often dealing with linear operators that are *not* defined on the entire space.

EXAMPLE 1.48. Consider $D = d/dx$ on $X = C([0, 1])$. This linear map is only defined on the subspace $C^1([0, 1])$.

This often happens when dealing with differential equations - in fact, it is one of the "tricks" of modern theory. We extend the definitions to this situation.

DEFINITION 1.49. Let X and Y be normed vector spaces. A map L that assigns to each x in a subset $\mathcal{D}(L)$ of X a unique element y in Y is called a **map** or **operator** with **domain** $\mathcal{D}(L)$. L is **linear** if (1) $\mathcal{D}(L)$ is a vector subspace of X and (2) $L(\alpha x_1 + \beta x_2) = \alpha L(x_1) + \beta L(x_2)$ for all $\alpha, \beta \in \mathbb{R}$ and $x_1, x_2 \in \mathcal{D}(L)$.

We now define the dual of a linear operator by examining its behavior on its domain. We want

$$L^* y^*(x) = y^*(Lx) \quad \text{all } x \in \mathcal{D}(L).$$

We say that $y^* \in \mathcal{D}(L^*)$ if there is an $x^* \in X^*$ such that

$$x^*(x) = y^*(Lx), \quad \text{all } x \in \mathcal{D}(L).$$

The existence of x^* is no longer automatic. When such an x^* exists, we define $L^* y^* = x^*$. For this to work, x^* must be unique. In other words, $x^*(x) = 0$ for all $x \in \mathcal{D}(L)$ should imply $x^* = 0$. This depends on the "size" of $\mathcal{D}(L)$.

DEFINITION 1.50. A subspace A of a normed linear space X is **dense** if every point in X is either in A or the limit of a sequence of points in A.

EXAMPLE 1.51. The rational numbers are dense in the real numbers and the polynomials are dense in $C([a, b])$.

The property of being dense gives an important approximation property. For example, the rational numbers are dense in the real numbers, which means we can approximate any real number arbitrarily well by a rational number. This is crucial to computer mathematics of course. The fact that we can approximate

continuous functions arbitrarily well by polynomials is one reason for the heavy use of polynomials in numerical analysis. Interestingly, the density of the polynomials in the space of continuous functions is connected to the famous probability theorem called the Weak Law of Large Numbers, see [**Est02**] for further discussion.

The argument presented above works if and only if $\mathcal{D}(L)$ is dense in X. We can define L^* for any linear operator $L : X \to Y$ provided $\mathcal{D}(L)$ is dense in X. We define $\mathcal{D}(L^*)$ to be those $y^* \in Y^*$ for which there is an $x^* \in X^*$ with $x^*(x) = y^*(Lx)$ for all $x \in X$. This x^* is unique and $L^*y^* = x^*$.

The Hahn-Banach theorem implies that if there is a C such that $|y^*(Lx)| \leq C\|x\|$ for all $x \in \mathcal{D}(L)$, then $y^* \in \mathcal{D}(L^*)$.

1.5. Adjoint operators - motivation

Having defined the adjoint operator abstractly, it is important to compute some examples in the infinite dimensional case. First, however, we will give some motivation for defining the adjoint by discussing a very important result that is closely related to Green's functions.

Many problems in applications take the form: Given normed vector spaces X and Y, an operator $\mathcal{L}(X, Y)$, and $b \in Y$, find $x \in X$ such that

$$Lx = b. \tag{1.3}$$

We explain the role of the adjoint in solving this kind of problems.

DEFINITION 1.52. The set of b for which there is a solution of (1.3) is called the **range**, $\mathcal{R}(L)$, of L. The set of x for which $L(x) = 0$ is called the **null space**, $\mathcal{N}(L)$, of L.

Note that 0 is always in $\mathcal{N}(L)$. If it is the only element in $\mathcal{N}(L)$, then $Lx = b$ can have at most one solution. Since L is linear,

THEOREM 1.53. *$\mathcal{N}(L)$ is a subspace of X and $\mathcal{R}(L)$ is a subspace of Y.*

Now if $y \in \mathcal{R}(L)$, there is an x with $Lx = y$. For $y^* \in Y^*$,

$$y^*(Lx) = y^*(y).$$

By the definition of the adjoint,

$$L^*y^*(x) = y^*(y).$$

If $y^* \in \mathcal{N}(L^*)$, then $y^*(y) = 0$. Thus, a necessary condition that $y \in \mathcal{R}(L)$ is that $y^*(y) = 0$ for all $y^* \in \mathcal{N}(L^*)$. Is this sufficient? We require just one condition.

DEFINITION 1.54. A subset A of a normed vector space X is **closed** if every sequence $\{x_n\}$ in A that has a limit in X has its limit in A.

We have

THEOREM 1.55. *Let X and Y be normed linear spaces and $L \in \mathcal{L}(X, Y)$. A necessary condition that $y \in \mathcal{R}(L)$ is $y^*(y) = 0$ for all $y^* \in \mathcal{N}(L^*)$. This is a sufficient condition if $\mathcal{R}$ is closed in Y.*

EXAMPLE 1.56. Suppose that $L \in \mathcal{L}(X, Y)$ is associated with the $n \times m$ matrix A, i.e., $L(x) = Ax$. The necessary and sufficient condition for the solvability of $Ax = b$ is that b is orthogonal to all linearly independent solutions of $A^T y = 0$.

EXAMPLE 1.57. In the case X is a Hilbert space and $L \in \mathcal{L}(X, Y)$, then necessarily $\mathcal{R}(L^*) \subset \mathcal{N}(L)^\perp$, where $S^\perp$ is the subspace of vectors that are orthogonal to a subspace S. This follows because

$$\begin{aligned} x \in \mathcal{N}(L) &\Rightarrow Lx = 0 \\ &\Rightarrow (y, Lx) = 0 \text{ all } y \in X \\ &\Rightarrow (L^* y, x) = 0 \text{ all } y \in X, \end{aligned}$$

i.e., $L^* y \in \mathcal{N}(L)^\perp$ for all $y \in X$. The claim follows since $\mathcal{R}(L^*) = \{L^* y | \text{ all } y \in X\}$. If $\mathcal{R}(L^*)$ is "large", then $\mathcal{N}(L)^\perp$ must be "large" and $\mathcal{N}(L)$ must be "small". The existence of sufficiently many solutions of the homogeneous adjoint equation implies there is at most one solution of $Lx = b$ for a given b.

EXAMPLE 1.58. The version of this theorem in the setting of general partial differential equations is a well-known and important result.

THEOREM 1.59. **Holmgren Uniqueness** *The generalized initial value problem consisting of the equation*

$$L(u) = \sum_{|\alpha| \le m} A_\alpha(x) D^\alpha u = f(x), \quad x \in \mathbb{R}^n,$$

where $\{A_\alpha\}$ and f are analytic functions, together with the data

$$D^\beta u(x) = g_\beta(x), \quad |\beta| \le m - 1, \, x \in S$$

given on an analytic noncharacteristic surface S, has at most one solution in a neighborhood of S.

Without being specific, a "noncharacteristic" surface is one on which is it is valid to pose "initial values". For example, we could not give initial values for the heat equation on a surface that is parallel to the time axis.

We will discuss the role of the adjoint in the solution of a general $n \times m$ system $Ax = b$, where A is a $n \times m$ matrix, $x \in \mathbb{R}^m$, and $b \in \mathbb{R}^n$. The reason to dwell on such problems is that differential operators do not tend to be "square".

EXAMPLE 1.60. $y'' = d^2y/dx^2$ requires two boundary conditions to define a problem for the associated differential equation that has a unique solution. We may want to study the differential operator without any boundary conditions, or more or less than two conditions. Divergence,

$$\operatorname{div} u = \frac{\partial u_1}{\partial x_1} + \cdots + \frac{\partial u_n}{\partial x_n},$$

associates a scalar with a given vector function, and the associated differential equation is very "under-determined". The gradient,

$$\operatorname{grad} u = \left(\frac{\partial u_1}{\partial x_1}, \cdots, \frac{\partial u_n}{\partial x_n} \right),$$

associates a vector field with a given scalar function, and the associated equations are very "over-determined".

So consider the $n \times m$ system $Ax = b$, where A is a $n \times m$ matrix, $x \in \mathbb{R}^m$, and $b \in \mathbb{R}^n$. We enlarge the system by adding the adjoint $m \times n$ system $A^\top y = c$,

where y and c are independent of x and b. The new $(n+m)\times(n+m)$ system has some very favorable properties. In particular, it is *symmetric*. We write it as

$$Sz = d,$$

with

$$z = \begin{pmatrix} y \\ x \end{pmatrix}, \quad d = \begin{pmatrix} b \\ c \end{pmatrix}, \quad S = \begin{pmatrix} 0 & A \\ A^\top & 0 \end{pmatrix}.$$

Since S is symmetric, it is diagonalizable with eigenvalues satisfying

$$\begin{aligned} Av &= \lambda u \\ A^* u &= \lambda v \end{aligned}$$

or

$$\begin{aligned} AA^\top u &= \lambda^2 u \\ A^\top A v &= \lambda^2 v. \end{aligned}$$

The eigenvalues of S are the singular values of A and last two equations give the left and right singular vectors of A.

We can use this to determine all kinds of facts about the compatibility and deficiency of the linear system $Ax = b$. Theorem 1.55 for $\mathbb{R}^n$ falls out right away. In the over-determined and under-determined cases, it yields a "natural" definition of a solution or gives conditions for a solution to exist. It also gives a way of determining the condition of the solution process.

One interesting observation is that there is a reciprocal relationship between the degree of over/under-determination of the original system and the adjoint system, i.e., the more over-determined the original system, the more under-determined the adjoint system, and so forth.

EXAMPLE 1.61. Consider $2x_1 + x_2 = 4$, where $L : \mathbb{R}^2 \to \mathbb{R}$. $L^* : \mathbb{R} \to \mathbb{R}^2$ is given by $L^* = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$. The extended system is

$$\begin{pmatrix} 0 & 2 & 1 \\ 2 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 4 \\ c_1 \\ c_2 \end{pmatrix},$$

from which we see that $2c_1 = c_2$ is *required* in order to have a solution.

On the other hand, if the problem is

$$\begin{aligned} 2x_1 + x_2 &= 4 \\ x_2 &= 3, \end{aligned}$$

with $L : \mathbb{R}^2 \to \mathbb{R}^2$, then there is a unique solution. The extended system is

$$\begin{pmatrix} 0 & 0 & 2 & 1 \\ 0 & 0 & 0 & 1 \\ 2 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} c_1 \\ c_2 \\ 4 \\ 3 \end{pmatrix},$$

where we can specify any values for c_1, c_2.

For later reference, this example shows that in the under-determined case, we can eliminate the deficiency by posing the method of solution

$$\begin{array}{c} AA^\top y = b \\ x = A^\top y. \end{array} \quad \text{or} \quad \begin{array}{c} L(L^*(y)) = b \\ x = L^*(y). \end{array}$$

We have not discussed computing the adjoint to a differential operator yet, but this approach also works in this case.

EXAMPLE 1.62. Consider the under-determined problem

$$\operatorname{div} F = \rho.$$

It turns out that, roughly speaking, the adjoint to div is -grad, though boundary conditions are required to be precise. If we set $F = \operatorname{grad} u$, where u is subject to the boundary condition $u(\text{“}\infty\text{''}) = 0$, then we obtain the “square”, well-determined problem

$$\operatorname{div}\operatorname{grad} u = \Delta u = -\rho,$$

which has a unique solution because of the boundary condition.

We will discuss this a bit further, but first we consider a particular augmented system.

EXAMPLE 1.63. Consider

$$Ax = b$$
$$A^\top \phi = e_i$$

where A is a $n \times n$ invertible matrix. There are no constraints on the data for the adjoint problem, and we have specified the i^{th} standard basis vector of $\mathbb{R}^n$. We see that

$$x_i = (x, e_i) = (x, A^\top \phi) = (Ax, \phi) = (b, \phi).$$

y is the discrete Green's vector associated to $Ax = b$.

1.6. Adjoint operators - computation

Before turning to the topic of Green's functions, we will spend a little time talking about the computation of an adjoint of a given linear differential operator.

Actually, it is difficult to find general discussions of this topic. Most of the texts and the literature consider problems that are particularly easy in a specific respect. The computation of the adjoint to a general differential operator is not easy at all.

We can obtain the *form* of the adjoint operator - including any boundary terms - via a tedious formal computation. We take the problem for the linear operator, including any boundary and/or initial conditions imposed with the operator, and discretize it using a low order method on a uniform discretization. Extract the matrix from the resulting discrete system, and compute its transpose, which gives a new linear operator. Finally, let the discretization parameter tend to its limit and determine the differential operator that is approached by the transposed matrix. This is a formal computation because it reveals nothing about the underlying spaces on which the operators are defined. To make it rigorous, we would have to discuss what it means for the approximate operators to converge to the true operators, which involves the spaces on which the operators are defined.

While this is not elegant, it does determine one important fact.

Determination of the adjoint of a differential operator is heavily influenced by the boundary and/or initial conditions imposed on the operator, as well as the underlying spaces.

In particular, a differential operator posed with two different sets of data will generally yield two different adjoints.

EXAMPLE 1.64. A standard difference approximation of

$$\begin{cases} -u''(x) = f(x), & 0 < x < 1, \\ u(0) = 0, u(1) = 0, \end{cases}$$

yields the matrix

$$\frac{1}{h^2}\begin{pmatrix} 2 & -1 & & & 0 \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ 0 & & & -1 & 2 \end{pmatrix},$$

which is symmetric. Hence, the differential operator is self-adjoint. If we change the boundary conditions to

$$\begin{cases} -u''(x) = f(x), & 0 < x < 1, \\ u(0) = 0, u'(0) = 0, \end{cases}$$

we get a triangular matrix after discretization. The adjoint corresponds to a problem

$$\begin{cases} -v''(x) = g(x), & 0 < x < 1, \\ v(1) = 0, v'(1) = 0, \end{cases}$$

EXAMPLE 1.65. Some other examples include an initial value problem

$$\begin{cases} u' = f(x), & 0 < x < 1, \\ u(0) = 0 \end{cases} \quad \Rightarrow \quad \begin{cases} -v' = g(x), & 1 > x > 0, \\ v(1) = 0, \end{cases}$$

and an under-determined problem

$$\begin{cases} -u'' = f(x), & 0 < x < 1, \\ \text{no boundary conditions} \end{cases}$$

$$\Rightarrow \quad \begin{cases} -v'' = g(x), & 0 < x < 1, \\ v(0) = v'(0) = v(1) = v'(1) = 0. \end{cases}$$

In the situation in which we consider functions in a Hilbert space like L^2, which is quite often, there is a more elegant and mathematically fundamental approach based on the bilinear identity

$$< Lu, v^* > = < u, L^* v^* > \quad \text{all } u \in X,\ v^* \in Y^*, \tag{1.4}$$

which in this context can be written

$$(Lu, v) = (u, L^* v) \quad \text{all } \textit{suitable } u, v \in L^2(\Omega). \tag{1.5}$$

DEFINITION 1.66. We say that we are **evaluating the bilinear identity** when we compute

$$< Lu, v^* > - < u, L^*v^* >= (Lu, v) - (u, L^*v)$$

for some suitable functions u and v.

We start with a couple of observations.

- Since we are considering *differential* operators, these will not be defined on all of $L^2(\Omega)$, but only a subset of sufficiently smooth functions. Likewise, the adjoint operator will be defined on a set of sufficiently smooth functions. To be able to work in spaces that are useful for analysis, we use a limiting process and distribution theory to extend the definitions to a larger space of functions, e.g., we work in the Sobolev spaces H^k rather than the spaces C^p. However, this involves the same kinds of subtle and technical issues that affected the definition of H^{-k}.
- The L^2 inner product involves integration over the domain, and we can interpret the process producing the bilinear identity as a succession of integration by parts, and the bilinear identity as a kind of generalized Green's identity. However, it is clear that integration by parts will yield terms that involve the integrals over the boundary of Ω of the functions involved as well as certain derivatives. The computation of the adjoint will involve the boundary conditions posed with the differential operator.

Computing the adjoint using the bilinear identity proceeds in two stages. We first compute a *formal* adjoint neglecting all boundary terms by assuming that the functions involved have compact support inside Ω.

DEFINITION 1.67. A function on a domain Ω has **compact support** in Ω if it vanishes identically outside a bounded set contained inside Ω.

The procedure for computing the formal adjoint can be described simply: Take the differential operator applied to a smooth function with compact support, multiply by a smooth test function with compact support, integrate over the domain, integrate by parts to move all derivatives onto the test function while ignoring boundary terms. Functions that have compact support are identically zero anywhere near the boundary and any boundary terms arising from integration by parts will vanish.

DEFINITION 1.68. Let L be a differential operator. The **formal adjoint** L^* is the differential operator that satisfies

$$(Lu, v) = (u, L^*v) \qquad \left(\int_\Omega Lu \cdot v \, dx = \int_\Omega u \cdot L^*v \, dx\right)$$

for all sufficiently smooth u and v with compact support in Ω.

EXAMPLE 1.69. Consider

$$Lu(x) = -\frac{d}{dx}\left(a(x)\frac{d}{dx}u(x)\right) + \frac{d}{dx}(b(x)u(x))$$

on $[0,1]$. Integration by parts neglecting boundary terms gives

$$\begin{aligned}
&-\int_0^1 \frac{d}{dx}\left(a(x)\frac{d}{dx}u(x)\right)v(x)\,dx \\
&\qquad = \int_0^1 a(x)\frac{d}{dx}u(x)\frac{d}{dx}v(x)\,dx - a(x)\frac{d}{dx}u(x)v(x)\Big|_0^1 \\
&\qquad = -\int_0^1 u(x)\frac{d}{dx}\left(a(x)\frac{d}{dx}v(x)\right)dx + u(x)a(x)\frac{d}{dx}v(x)\Big|_0^1,
\end{aligned}$$

and

$$\int_0^1 \frac{d}{dx}(b(x)u(x))v(x)\,dx = -\int_0^1 u(x)b(x)\frac{d}{dx}v(x)\,dx + b(x)u(x)v(x)\Big|_0^1,$$

where all of the boundary terms vanish. Therefore,

$$L^*v = -\frac{d}{dx}\left(a(x)\frac{d}{dx}v(x)\right) - b(x)\frac{d}{dx}(v(x)).$$

The basic technique for obtaining the formal adjoint for differential operators posed in higher space dimensions is the divergence theorem.

EXAMPLE 1.70. A general linear second order differential operator L in $\mathbb{R}^n$ can be written

$$L(u) = \sum_{i=1}^n \sum_{j=1}^n a_{ij}\frac{\partial^2 u}{\partial x_i \partial x_j} + \sum_{i=1}^n b_i \frac{\partial u}{\partial x_i} + cu,$$

where $\{a_{ij}\}$, $\{b_i\}$, and c are functions of $x_1, x_2, \cdots, x_n$. Then,

$$L^*(u) = \sum_{i=1}^n \sum_{j=1}^n \frac{\partial^2 (a_{ij}v)}{\partial x_i \partial x_j} - \sum_{i=1}^n \frac{\partial (b_i v)}{\partial x_i} + cv.$$

It can be verified directly that

$$vL(u) - uL^*(v) = \sum_{i=1}^n \frac{\partial p_i}{\partial x_i},$$

where

$$p_i = \sum_{j=1}^n \left(a_{ij}v\frac{\partial u}{\partial x_j} - u\frac{\partial (a_{ij}v)}{\partial x_j}\right) + b_i uv.$$

The expression on the right is a divergence expression and the divergence theorem yields

$$\int_\Omega (vL(u) - uL^*(v))\,dx = \int_{\partial\Omega} p\cdot n\,ds = 0,$$

where $p = (p_1, \cdots, p_n)$ and n is the outward normal in $\partial\Omega$.

To see a typical term,

$$\begin{aligned}
va_{11}\frac{\partial^2 u}{\partial x_1^2} &= va_{11}\frac{\partial}{\partial x_1}\left(\frac{\partial u}{\partial x_1}\right) = \frac{\partial}{\partial x_1}\left(va_{11}\frac{\partial u}{\partial x_1}\right) - \frac{\partial (a_{11}v)}{\partial x_1}\frac{\partial u}{\partial x_1} \\
&= \frac{\partial}{\partial x_1}\left(va_{11}\frac{\partial u}{\partial x_1}\right) - \frac{\partial}{\partial x_1}\left(u\frac{\partial (a_{11}v)}{\partial x_1}\right) + u\frac{\partial^2 (a_{11}v)}{\partial x_1^2}
\end{aligned}$$

yielding

$$va_{11}\frac{\partial^2 u}{\partial x_1^2} - u\frac{\partial^2 (a_{11}v)}{\partial x_1^2} = \frac{\partial}{\partial x_1}\left(a_{11}v\frac{\partial u}{\partial x_1} - u\frac{\partial (a_{11}v)}{\partial x_1}\right).$$

EXAMPLE 1.71. Let L be a differential operator of order $2p$ of the form

$$Lu = \sum_{|\alpha|,|\beta|\leq p} (-1)^{|\alpha|} D^\alpha \big(a_{\alpha\beta}(x) D^\beta u\big),$$

then

$$L^* v = \sum_{|\alpha|,|\beta|\leq p} (-1)^{|\alpha|} D^\alpha \big(a_{\beta\alpha}(x) D^\beta v\big),$$

and L is elliptic if and only if L^* is elliptic. Some special cases.

$$\text{grad}^* = -\text{div}$$
$$\text{div}^* = -\text{grad}$$
$$\text{curl}^* = \text{curl}$$

and if

$$Lu = \sum_{|\alpha|\leq p} a_\alpha(x) D^\alpha u$$

then

$$L^* v = \sum_{|\alpha|\leq p} (-1)^{|\alpha|} D^\alpha (a_\alpha(x) v(x)).$$

EXAMPLE 1.72. If

$$Lu = \rho u_{tt} - \nabla \cdot (a \nabla u) + bu$$

then $L^* = L$. If

$$Lu = u_t - \nabla \cdot (a \nabla u) + bu$$

then

$$L^* v = -v_t - \nabla \cdot (a \nabla v) + bv$$

where time runs "backwards" as in Ex. 1.65.

This procedure also works for systems

EXAMPLE 1.73. Let

$$L(\vec{u}) = A\vec{u}_x + B\vec{u}_y + C\vec{u},$$

where A, B, and C are $n \times n$ matrices, then

$$L^*(\vec{v}) = (-A^\top \vec{v})_x - (B^\top \vec{v})_y + C^\top v,$$

so that

$$\vec{v}^\top L\vec{u} - \vec{u}^\top L^* \vec{v} = \nabla \cdot (\vec{v}^\top A\vec{u}, \vec{v}^\top B\vec{u}).$$

In the second stage of computing the adjoint, we deal with boundary conditions by removing the assumption that the functions involved have compact support. Now the integration by parts that produces the formal adjoint will yield additional terms involving integrals over the boundary of the functions involved and their derivatives.

Consider Examples 1.69 and 1.70. We want to determine boundary conditions such that the bilinear identity (1.5) still holds, e.g., such that any boundary terms that arise vanish. It turns out that the *form* of the boundary conditions imposed in the problem for L are important, but the values given for these conditions are not. If the boundary conditions are not homogeneous, we make them so for the purpose of determining the adjoint.

With this assumption, we define.

DEFINITION 1.74. The **adjoint boundary conditions** posed on the formal adjoint operator are the *minimal* conditions required to make the boundary terms that appear when evaluating the bilinear identity for general smooth functions vanish.

Some of the boundary terms that appear when evaluating the bilinear identity will vanish because of the boundary conditions imposed in the original problem. The point of this assumption is to make the formal adjoint serve as the true adjoint by pairing it with the correct boundary conditions.

This definition is rather vague, but it can be made completely precise. Issues that have to be dealt with include

- Placing conditions on the differential operator L so that evaluating the bilinear identity for general smooth functions results in expressions involving only values on the boundary.
- Making precise the meaning of "minimal conditions" needed for the adjoint problem, and proving these always exist.

This can be done, but it is complicated to write down. Instead, we settle for some examples.

EXAMPLE 1.75. Consider Newton's equation of motion $s''(x) = f(x)$ with $x =$ "time", normalized with mass 1. First, suppose we assume $s(0) = s'(0) = 0$, and $0 < x < 1$. We have

$$s''v - sv'' = \frac{d}{dx}(vs' - sv')$$

and

$$\int_0^1 (s''v - sv'')\,dx = (vs' - sv')\big|_0^1. \tag{1.6}$$

Now the boundary conditions imply the contributions at $x = 0$ vanish, while at $x = 1$ we have

$$v(1)s'(1) - v'(1)s(1).$$

To insure this vanishes, we must have $v(1) = v'(1) = 0$. (We cannot specify $s(1)$ or $s'(1)$ of course.) These are the adjoint boundary conditions.

Suppose instead we wish to impose conditions such that at the mass returns to the origin with zero speed at $x = 1$. This gives *four* boundary conditions $s(0) = s'(0) = s(1) = s'(1) = 0$ on the original problem. In this situation, all of the boundary terms in (1.6) are zero and *no* boundary conditions will be imposed on the adjoint.

It is interesting to find a solution of the over-determined problem. Based on the discussion above, we require the data f to be orthogonal to the solution of the adjoint problem $v'' = 0$, which is $v = a + bx$. Hence, f must be orthogonal in $L^2(0,1)$ to 1 and x. Assume for example that $f(x) = a + bx + cx^2$. It is easy to see that $(f,1) = 0$ and $(f,x) = 0$ forces $a = c/6$ and $b = -c$. Choosing $c = 1$ for example, means that $f(x) = 1/6 - x + x^2$. We solve the forward problem by integrating twice and using the boundary conditions at $x = 0$ to get a formula for the solution, which is easily seen to satisfy the conditions at $x = 1$ as well.

EXAMPLE 1.76. Since

$$\int_\Omega (u\Delta v - v\Delta u)\,dx = \int_{\partial\Omega} \left(u\frac{\partial v}{\partial n} - v\frac{\partial u}{\partial n}\right)\,ds,$$

the Dirichlet and Neumann boundary value problems for the Laplacian are their own adjoints.

EXAMPLE 1.77. Let $\Omega \subset \mathbb{R}^2$ be bounded with a smooth boundary and let $s =$ arclength along the boundary. Consider

$$\begin{cases} -\Delta u = f, & x \in \Omega, \\ \frac{\partial u}{\partial n} + \frac{\partial u}{\partial s} = 0, & x \in \partial\Omega. \end{cases}$$

Since

$$\int_\Omega (u\Delta v - v\Delta u)\, dx = \int_{\partial\Omega} \left(u\left(\frac{\partial v}{\partial n} - \frac{\partial v}{\partial s}\right) - v\left(\frac{\partial u}{\partial n} + \frac{\partial u}{\partial s}\right)\right) ds,$$

the adjoint problem is

$$\begin{cases} -\Delta v = g, & x \in \Omega, \\ \frac{\partial v}{\partial n} - \frac{\partial v}{\partial s} = 0, & x \in \partial\Omega. \end{cases}$$

1.7. Green's functions

We conclude the first chapter by defining Green's functions. This will be almost anticlimactic after the long introduction.

For simplicity, we consider a problem of the form

$$\begin{cases} Lu = f, & x \in \Omega, \\ \text{suitable b.c. and i.v.}, & x \in \partial\Omega, \end{cases} \tag{1.7}$$

where L is a linear differential operator, Ω is a space, time, or space-time domain, and we specify the correct boundary and/or initial conditions so that (1.7) has a unique solution.

DEFINITION 1.78. The **Green's function** for (1.7) satisfies

$$\begin{cases} L^*\phi(y,x) = \delta_y(x), & x \in \Omega, \\ \text{adjoint b.c. and i.v.}, & x \in \partial\Omega, \end{cases} \tag{1.8}$$

where L^* is the formal adjoint of L.

It is useful to think of the discussion in Sec. 1.5 and to realize we are solving the extended *system*,

$$\begin{cases} Lu = f, & x \in \Omega, \\ \text{suitable b.c. and i.v.}, & x \in \partial\Omega, \\ L^*\phi(y,x) = \delta_y(x), & x \in \Omega, \\ \text{adjoint b.c. and i.v.}, & x \in \partial\Omega. \end{cases}$$

The reason for this definition is simple. Based on the bilinear identity, we obtain a *representation formula* for the value of the solution of the original problem at a point $y \in \Omega$,

$$u(y) = (u, \delta_y) = (y, L^*\phi) = (Lu, \phi) = (f, \phi). \tag{1.9}$$

The imposition of the adjoint boundary conditions is key here.

EXAMPLE 1.79. For

$$\begin{cases} -\Delta u = f, & x \in \Omega, \\ u = 0, & x \in \partial\Omega, \end{cases}$$

ϕ solves

$$\begin{cases} -\Delta \phi(y;x) = \delta_y(x), & x \in \Omega, \\ \phi(y;x) = 0, & x \in \partial\Omega, \end{cases} \tag{1.10}$$

and the bilinear identity reads

$$u(y) = \int_\Omega f(x)\phi(y;x)\,dx.$$

We plot the Green's function in Fig. 1.2.

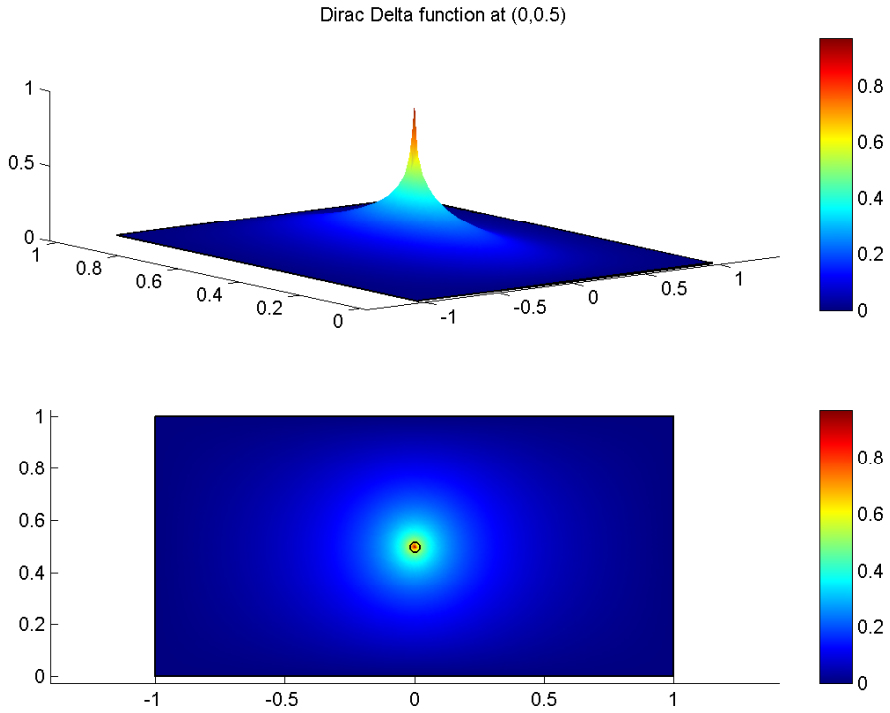

FIGURE 1.2. Plot of the Green's function solving (1.10).

EXAMPLE 1.80. For

$$\begin{cases} u_t - \Delta u = f, & x \in \Omega, 0 < t \le T, \\ u(t,x) = 0, & x\partial\Omega, 0 < t \le T, \\ u(0,x) = 0, & x \in \Omega, \end{cases}$$

ϕ solves

$$\begin{cases} -\phi_t - \Delta \phi = \delta_{(s,y)}(t,x), & x \in \Omega, T > t \ge 0, \\ \phi(t,x) = 0, & x\partial\Omega, T > t \ge 0, \\ \phi(T,x) = 0, & x \in \Omega, \end{cases}$$

and the bilinear identity reads

$$u(s,y) = \int_0^T \int_\Omega f(t,x)\phi(s,y;t,x)\,dxdt.$$

There are several motivations behind the definition of the Green's function. Some obvious ones include

- The Green's function is defined without reference to the particular data f that determines a particular solution. The Green's function depends on the operator and its properties. Having determined the Green's function, we can use the representation to consider the effect of choosing particular data.
- The Green's function generally has special properties arising from the properties of the delta function, such as localized support and symmetry, and we can often determine a great deal of information about a Green's function. In some cases, we can even get a formula.
- The Green's function gives a "low-dimensional" snapshot of the solution. We can recover the entire solution using infinitely many Green's functions.

EXAMPLE 1.81. The Green's function for the Dirichlet problem for the Laplacian $L = -\Delta$ on the ball Ω of radius r centered at the origin in $\mathbb{R}^3$ is

$$\phi(y;x) = \frac{1}{4\pi} \times \begin{cases} |y-x|^{-1} - r|y|^{-1}\left|\frac{r^2 y}{|y|^2} - x\right|^{-1}, & y \neq 0, \\ |x|^{-1} - r^{-1}, & y = 0, \end{cases} \tag{1.11}$$

where $|x|$ denotes the Euclidean norm of x, while the formula for the disk of radius r is

$$\phi(y;x) = \frac{1}{2\pi} \times \begin{cases} \ln\left(\frac{|y|\left|\frac{r^2 y}{|y|^2} - x\right|}{r|y-x|}\right), & y \neq 0, \\ \ln\left(\frac{r}{|x|}\right), & y = 0. \end{cases} \tag{1.12}$$

It is hard to decipher the meaning of these formulas, but we discuss them further below.

There are some important mathematical issues that have to be settled, such as the existence, uniqueness, and smoothness of the Green's function. These are problem dependent of course, and it requires distribution theory to complete the theory. As a general rule, everything goes as for the original problem except that the Green's function may not be very smooth or may even be undefined at a point. We mention one important point:

> *The point of evaluation y must lie inside the domain Ω. The Green's function often behaves badly in the limit of y approaching the boundary.*

By the way, the theory also extends to over-determined problems. However, if the original problem is under-determined, this approach fails.

If Green's functions are familiar, the focus on the adjoint might be confusing because many expositions avoid the adjoint. It turns out that when the original problem has a unique solution and when the original operator is the adjoint to the adjoint operator, the Green's function ϕ for the original problem and the Green's function for the adjoint problem ϕ^* are related via

$$\phi(y;x) = \phi^*(x;y).$$

This is known as the *reciprocity theorem*. This makes it possible to introduce Green's functions for some kinds of problems without talking about the adjoint, at a cost of a great deal of structure and motivation. Note that the question of the adjoint to the adjoint being the same as the original operator depends on the dual space of the dual space of the original space being identifiable with the original space. This question is also very dependent on the special properties of the delta function. We will not be able to use this approach in the sequel.

So far, we have avoided discussion of nonhomogeneous boundary conditions. Including nonhomogeneous conditions is really a minor issue that usually "solves itself" without trouble. We just carry out the analysis using the Green's function for the homogeneous problem and some additional integrals involving data and the Green's function will appear.

EXAMPLE 1.82. Suppose the problem is

$$\begin{cases} -\Delta u = f, & x \in \Omega, \\ u = g, & x \in \partial\Omega. \end{cases}$$

We define the Green's function as for the homogeneous problem, i.e.,

$$\begin{cases} -\Delta \phi(y;x) = \delta_y(x), & x \in \Omega, \\ \phi(y;x) = 0, & x \in \partial\Omega. \end{cases}$$

Evaluating the bilinear identity yields

$$\int_\Omega (u\Delta\phi - \phi\Delta u)\,dx = \int_{\partial\Omega} \left(u\frac{\partial\phi}{\partial n} - \phi\frac{\partial u}{\partial n} \right) ds = \int_{\partial\Omega} u\frac{\partial\phi}{\partial n}\,ds.$$

This yields

$$u(y) = \int_\Omega f(x)\phi(y;x)\,dx - \int_{\partial\Omega} g(s)\frac{\partial\phi(y;s)}{\partial n}\,ds.$$

The basic idea can also be varied to obtain different representations.

EXAMPLE 1.83. For

$$\begin{cases} -\Delta u = f, & x \in \Omega, \\ u = 0, & x \in \partial\Omega, \end{cases}$$

We can define the Green's function as the solution of

$$\begin{cases} -\Delta \phi(y;x) = 0, & x \in \Omega, \\ \phi(y;x) = \delta_y(x), & x \in \partial\Omega. \end{cases} \tag{1.13}$$

Evaluating the bilinear identity yields

$$\begin{aligned} \int_\Omega (u\Delta\phi - \phi\Delta u)\,dx &= \int_{\partial\Omega} \left(u\frac{\partial\phi}{\partial n} - \phi\frac{\partial u}{\partial n} \right) ds \\ &= -\int_{\partial\Omega} \phi\frac{\partial u}{\partial n}\,ds = -\int_{\partial\Omega} \delta_y \frac{\partial u}{\partial n}\,ds = -\frac{\partial u}{\partial n}(y). \end{aligned}$$

This gives the value of the normal derivative of u at a point y on the boundary,

$$\frac{\partial u}{\partial n}(y) = -\int_\Omega \phi(y;x) f(x)\,dx.$$

We plot the Green's function in Fig. 1.3.

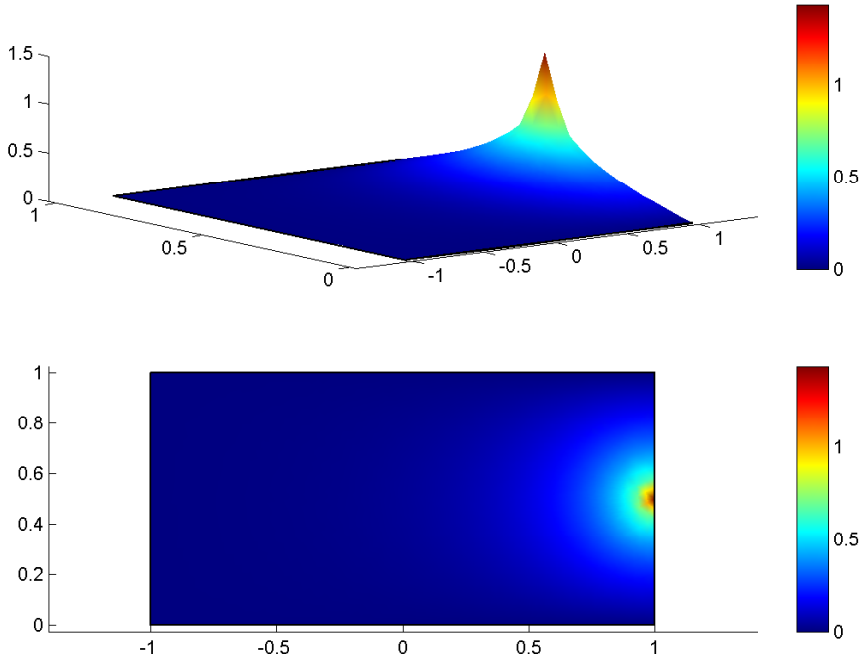

FIGURE 1.3. Plot of the Green's function solving (1.13).

The last example demonstrates the fact that the standard Green's function defined with a delta function at a point in the interior of Ω often behaves badly as this point approaches the boundary. That Green's function yields the value of the solution at a point, while actually putting the delta function at a point on the boundary yields the normal derivative of the function. To be fair, we note that the delta function in the interior and the delta function on the boundary are not the same "kind" of delta functions because they are defined on sets of different dimension. We discuss this briefly below.

CHAPTER 2

A Posteriori Error Analysis and Adaptive Error Control

In this chapter, we explain how the Green's function can be used as a basis of a very powerful approach to *a posteriori* error analysis for finite element methods. The most well-known version of this approach used for adaptive error control is called the *dual weighted residual* method, which has found widespread use in engineering. The basic error estimate is given in terms of residuals of the computed finite element solution weighted by stability factors determined from the solution of an adjoint problem.

The use of the adjoint problem to obtain information about stability in *a posteriori* error analysis was introduced in [**EJ91**]. The goal in that paper was to obtain reasonably accurate *a priori* bounds on the adjoint weights, and the paper dealt with the Poisson equation and the linear heat equation, for which this is possible. The idea of numerically solving the adjoint problem to compute very accurate *a posteriori* estimates was introduced experimentally in [**DE91**] and developed in the context of nonlinear ordinary differential equations in [**Est95**] and nonlinear partial differential equations in [**EW96**]. Since these early contributions, there has been substantial contributions and applications, see [**EEHJ95, ELW00, BR01, GS02, BR03**] for example.

This method is most often described in a framework of optimal control theory in which the adjoint has a fundamental role. But, the suitability of this interpretation is very problem dependent, and thinking of this approach from the point of view of Green's functions is more generally meaningful.

2.1. A generalization of the Green's function

Before presenting the *a posteriori* error analysis, we discuss a generalization of the notion of the Green's function. Recall that the idea behind the definition of the Green's function is that it yields a representation of the value of the solution of the differential equation,

$$u(y) = (u, \delta_y) = (\phi(y; x), f(x)).$$

Recall that the value of a function at a point is a linear functional. When solving differential equations, there are frequently other quantities of interest besides the value of a solution at a particular point. It turns out that many such quantities that can also be expressed as functionals. Furthermore, the Riesz Representation theorem 1.41 suggests that we can represent many linear functionals as inner products with particular distribution functions, i.e., as (u, ψ), where ψ is some distribution in a suitable Sobolev space. Some useful choices of ψ include:

- We use the delta function $\psi = \delta_y$ to get the error at a point y. Similarly, we can construct $\psi = \delta_c$ to get the average value $\oint_c e(s)\,ds$ of the error over a curve c in $\mathbb{R}^n$, $n = 2, 3$, and $\psi = \delta_s$ to get the average value of the error over a plane surface s in $\mathbb{R}^3$.

  This choice is actually trickier than it might appear. The issue is that a function that is merely in L^2 is only defined "almost everywhere" in the sense of measure theory, which means it does not make sense to ask for a value at a particular point. (Recall that a function is in L^2 if the integral of its square is bounded. Changing the values of such a function at isolated points does not affect the integral, and hence does not affect whether it is in L^2.) So, the function in question needs to have a certain smoothness, i.e., be in an H^s space for suitable s. The exact requirement is given by a famous theorem.

  THEOREM 2.1. **Sobolev** *If $s > k + n/2$, then there is a constant C such that for $f \in H^s(\mathbb{R}^n)$,*

$$\max_{|\alpha| \le k} \sup_{x \in \mathbb{R}^n} |D^\alpha(x)| \le C \|f\|_s$$

  This implies that the derivatives of f of order k and less are continuous.

  The Sobolev theorem shows that if $s > n/2$, the evaluation map $f \to f(x)$ is well-defined for $f \in H^s$. More generally, if $k \le n$ and $s > k/2$, restricting a function in H^s to a submanifold of (co)dimension k is well-defined. The submanifolds considered here are the curve or the surface mentioned above, which have lower (co)dimension than the space in which we pose the differential equation.
- We can obtain errors in derivatives using dipoles in a similar way.
- $\psi = \chi_\omega / |\omega|$ gives the error in the average value over a subset $\omega \subset \Omega$, where χ_ω is the characteristic function of ω. The average error has some interesting properties, such as it is possible for the average error to be small even when the error is large in norm. In elliptic problems in small regions, the average error tends to act like the error in the L^1 norm.
- In some problems, choosing ψ to be the residual of the finite element approximation (which we define below) yields the energy norm of the error of the approximation.
- $\psi = \chi_\omega e / \|e\|_\omega$, where e is the error of the finite element discretization, gives the $L^2(\omega)$ norm of the error. In practice, we do not have the error to use this choice exactly, but good approximations can be obtained with Richardson extrapolation using finite element solutions with different accuracy.

Note that only some of these data ψ have spatially local support.

Again, we consider a problem of the form

$$\begin{cases} Lu = f, & x \in \Omega, \\ \text{suitable b.c. and i.v.}, & x \in \partial\Omega, \end{cases}$$

where L is a linear differential operator and we specify the correct boundary and/or initial conditions so that (1.7) has a unique solution.

DEFINITION 2.2. The **generalized Green's function** for (1.7) corresponding to the quantity of interest represented by (u, ψ) satisfies

$$\begin{cases} L^*\phi(y,x) = \psi(x), & x \in \Omega, \\ \text{adjoint b.c. and i.v.}, & x \in \partial\Omega, \end{cases} \tag{2.1}$$

where L^* is the formal adjoint of L.

As in Ex. 1.83, there are minor variations of this definition in which we pose the data ψ on the boundary of Ω rather than the interior (i.e., as boundary or initial data). We also call these functions generalized Green's functions.

2.2. Discretization by the finite element method

For simplicity, we will concentrate on the general second order linear elliptic boundary value problem for a scalar unknown,

$$\begin{cases} Lu = f, & x \in \Omega, \\ u = 0, & x \in \partial\Omega, \end{cases} \tag{2.2}$$

where

$$L(D,x)u = -\nabla \cdot a(x)\nabla u + b(x) \cdot \nabla u + c(x)u(x), \tag{2.3}$$

with $u : \mathbb{R}^n \to \mathbb{R}$, a is a $n \times n$ matrix function of x, b is a n-vector function of x, and c is a function of x. We assume that $\Omega \subset \mathbb{R}^n$, $n = 2, 3$, is a smooth or polygonal domain; $a = (a_{ij})$, where $a_{i,j}$ are continuous in $\overline{\Omega}$ for $1 \le i, j \le n$ and there is a $a_0 > 0$ such that $v^\top a v \ge a_0$ for all $v \in \mathbb{R}^n \setminus \{0\}$ and $x \in \Omega$; $b = (b_i)$ where b_i is continuous in $\overline{\Omega}$; and finally c and f are continuous in $\overline{\Omega}$.

We discretize (2.2) by applying a finite element method to the associated variational formulation:

(2.4) Find $u \in H_0^1(\Omega)$ such that

$$A(u,v) = (a\nabla u, \nabla v) + (b \cdot \nabla u, v) + (cu, v) = (f, v) \text{ for all } v \in H_0^1(\Omega).$$

To construct a finite element discretization, we form a piecewise polygonal approximation of $\partial\Omega$ whose nodes lie on $\partial\Omega$ and which is contained inside Ω. This forms the boundary of a convex polygonal domain Ω_h. We let $\mathcal{T}_h$ denote a simplex triangulation of Ω_h that is locally quasi-uniform. We let h_K denote the length of the longest edge of $K \in \mathcal{T}_h$ and define the piecewise constant mesh function h by $h(x) = h_K$ for $x \in K$. We also use h to denote $\max_K h_K$. See Fig. 2.1. We choose a finite element solution from the space V_h of functions that are continuous on Ω, piecewise linear on Ω_h with respect to $\mathcal{T}_h$, zero on the boundary $\partial\Omega_h$, and finally extended to be zero in the region $\Omega \setminus \Omega_h$, see Fig. 2.1. With this construction, we have $V_h \subset H_0^1(\Omega)$, and for smooth functions, the error of interpolation into V_h is $\mathbf{O}(h^2)$ in $\| \, \|$, but not better (see [**JLTW87**]). The finite element method is:

(2.5) Compute $U \in V_h$ such that $A(U,v) = (f,v)$ for all $v \in V_h$.

In these notes, we take for granted the usual *a priori* convergence results for finite element methods and concentrate on the *a posteriori* analysis used to produce computational error estimates. In particular, by standard results, we know that U exists and converges to u as $h \to 0$.

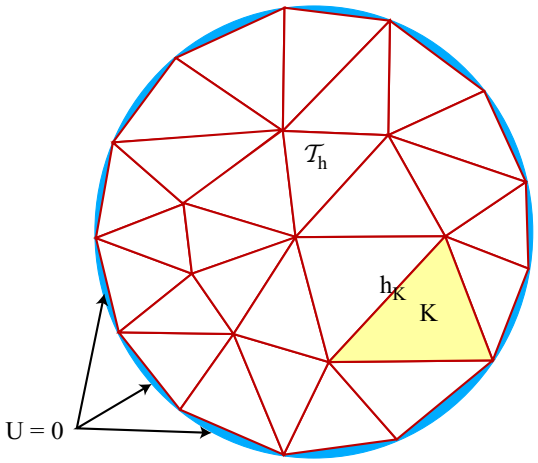

FIGURE 2.1. Discretization of a domain Ω with curved boundaries. We extend the piecewise linear functions defined on the triangulation to be zero in the parts of Ω not covered by the mesh.

2.3. An *a posteriori* analysis for an algebraic equation

The approach to a posteriori error analysis is relatively simply to explain in the context of the numerical solution of a system of algebraic equations. The problem here is to estimate the error of a numerical solution X of a system of algebraic equations

$$Ax = b \tag{2.6}$$

where $b, x \in \mathbb{R}^n$ and A is a $n \times n$ matrix. We assume that a numerical solution X of (2.6) has been computed in some fashion and we seek to estimate the unknown error $e = x - X$.

The residual error of X is defined simply as

$$R = AX - b$$

and is generally not zero. The residual error measures how well the approximate solution satisfies the true equation. The issue is to find a relation between the *computable* residual and the *unknown* error.

> *Recall that a standard numerical linear algebra result says that the residual can be small even if the error is large.*

This has profound consequences for adaptive error control, see [**ELW00**] for further discussion.

There are at least two ways to do this. First, we can use the fact that the residual error of the true solution is zero to write

$$Ae = -R.$$

We can then try to obtain an estimate of the error by solving this equation approximately in some fashion. This is a classic technique in numerical linear algebra sometimes called iterative improvement (often after obtaining e this way, it is added back to the approximation X to improve the accuracy). There are some subtle issues having to do with round-off error that must be resolved to get this to work

reliably. These are addressed by computing the numerical solution in single precision and the error in double precision, which is probably why this approach has lost favor. Moreover, in the context of solving differential equations, obtaining the entire error is generally a very bad idea if we simply want the error in certain quantities of interest.

Instead, we obtain an estimate on a quantity of interest represented by a linear functional of the error. Following the idea of extending the Green's function in Sec. 2.1, we introduce the generalized Green's vector solving the adjoint problem

$$A^\top \phi = \psi,$$

where ψ is any unit vector. The Riesz Representation theorem 1.41 says that we can obtain any linear functional by taking the inner product with a certain vector ψ in this way. To obtain an estimate on the size of the first component of e, we would choose $\psi = (1\,0\,0\,\cdots\,0)^\top$, whereas to obtain an estimate on the average of the components of the error, we would choose $\psi = (1\,1\,\cdots\,1)^\top/n$. If we could be so fortunate to choose $\psi = e/\|e\|$ for example, then we would get an estimate on $\|e\|$ (though this is a nonlinear functional).

Extending the argument in Ex. 1.63, we compute

$$|(e,\psi)| = |(e, A^\top\phi)| = |(Ae,\phi)| = |(R,\phi)|. \tag{2.7}$$

This error representation formula leads leads directly to the error bound

$$|(e,\psi)| \le \|\phi\|\,\|R\|. \tag{2.8}$$

Since the residual R is computable, if we compute a numerical approximation of the generalized Green's function ψ or obtain an estimate on the size of ϕ in some other way, then we obtain an estimate and a bound on the error in the quantity of interest.

DEFINITION 2.3. $\|\phi\|$ is called the stability factor for this problem.

The stability factor is related to the condition number of A. In fact, it follows that

$$\left|\left(\frac{e}{\|x\|},\psi\right)\right| \le \operatorname{cond}_\psi(A)\,\frac{\|R\|}{\|b\|},$$

where

$$\operatorname{cond}_\psi(A) = \|\phi\|\,\|A\| = \|A^{-\top}\psi\|\,\|A\|$$

is a kind of "weak" condition number of A with respect to the targeted quantity of interest. If we take the maximum of $\operatorname{cond}_\psi(A)$ over all possible ψ, we obtain the standard condition number of A. Hence, the stability factor obtained from the generalized Green's function is a measure of the sensitivity of numerical solutions of the problem to computational errors.

> *It is important to realize that the error in a quantity of interest can be small even if some norm of the error is large.*

2.4. An *a posteriori* analysis for a finite element method

The goal of the *a posteriori* error analysis is to estimate the error in a quantity of interest computed from the finite element solution U. To do this, we use a generalized Green's function ϕ solving the adjoint problem corresponding to a special choice of data ψ.

Classical analysis of finite element methods tends to focus on estimating the error in global norms, such as $\| \quad \|_{L^2(\Omega)}$, $\| \quad \|_{L^\infty(\Omega)}$, and of course the energy norm. In practice, however, this may not be meaningful. Often, the practical goal for solving a differential equation is to compute specific information from the solution, and in those situations, we should naturally be concerned with the error in the desired information. This may not have much to do with the error in some global norm. The implications for adaptive error control are significant.

> *It may be computationally infeasible as well as very inefficient to attempt to control the error in a global norm when all that is desired is accuracy in some quantities of interest.*

Therefore, we assume that the information we wish to compute can be represented as (u, ψ). We compute the generalized Green's function ϕ as the solution of the weak adjoint problem,

(2.9) Find $\phi \in H_0^1(\Omega)$ such that

$$A^*(v, \phi) = (\nabla v, a\nabla\phi) - (v, \operatorname{div}(b\phi)) + (v, c\phi) = (v, \psi) \text{ for all } v \in H_0^1(\Omega),$$

corresponding to the adjoint problem $L^*(D, x)\phi = \psi$. Extending the analysis behind the Green's function described in Sec. 1.7,

$$\begin{aligned}(e, \psi) &= (\nabla e, a\nabla\phi) - (e, \operatorname{div}(b\phi)) + (e, c\phi)\\ &= (a\nabla e, \nabla\phi) + (b \cdot \nabla e, \phi) + (ce, \phi)\\ &= (a\nabla u, \nabla\phi) + (b \cdot \nabla u, \phi) + (cu, \phi) - (a\nabla U, \nabla\phi) - (b \cdot \nabla U, \phi) - (cU, \phi)\\ &= (f, \phi) - (a\nabla U, \nabla\phi) - (b \cdot \nabla U, \phi) - (cU, \phi).\end{aligned}$$

Letting $\pi_h\phi$ denote an approximation of ϕ in V_h, using Galerkin orthogonality (2.5), we conclude

THEOREM 2.4. *The error in the quantity of interest computed from the finite element solution (2.5) satisfies the* **error representation**,

$$(e, \psi) = (f, \phi - \pi_h\phi) - (a\nabla U, \nabla(\phi - \pi_h\phi)) - (b \cdot \nabla U, \phi - \pi_h\phi) - (cU, \phi - \pi_h\phi), \tag{2.10}$$

where the generalized Green's function ϕ satisfies the adjoint problem (2.9) corresponding to data ψ.

The most accurate *a posteriori* error estimates are obtained by using (2.10) directly as opposed to making further estimates. To use the estimate, we approximate ϕ using a finite element method. Since $\phi - \pi_h\phi \sim \sum_{|\alpha|=2} D^\alpha\phi$ where ϕ is smooth, we use a higher order finite element than that used to solve the original boundary value problem. For example, good results are obtained using the space V_h^2 of continuous, piecewise quadratic functions with respect to $\mathcal{T}_h$. The approximate generalized Green's function is

(2.11) Compute $\Phi \in V_h^2$ such that

$$A^*(v, \Phi) = (\nabla v, a\nabla\Phi) - (v, \operatorname{div}(b\Phi)) + (v, c\Phi) = (v, \psi) \text{ for all } v \in V_h^2.$$

DEFINITION 2.5. The **approximate error representation** is

$$(e, \psi) \approx (f, \Phi - \pi_h\Phi) - (a\nabla U, \nabla(\Phi - \pi_h\Phi)) - (b \cdot \nabla U, \Phi - \pi_h\Phi) - (cU, \Phi - \pi_h\Phi). \tag{2.12}$$

Recently, we have been using even higher order, smoother finite element methods on coarse, uniform meshes. Another important detail is the choice of quadrature used to evaluate the integrals in the terms in (2.10). We have found that accurate evaluation of the estimate requires relatively high order quadratures. The reason appears to be that there is a great deal of cancellation of contributions among these integrals in general.

We note that there is a wide variation in how (2.10) or results derived from (2.10) are used to compute error estimates in practice. There has been relatively little theoretical analysis directed towards understanding the effect of the approximations required for implementation on the accuracy and reliability of the result.

We present several computational examples below that are performed using *FETkLab* [**EH02**]. This adaptive finite element code, running under *MATLAB*, can solve general nonlinear elliptic systems on general domains in two space dimensions. It implements the *a posteriori* error estimate, allowing up to 16 simultaneous adjoint data ψ_i to be specified. In the computations below, we use bisection or red-green quadrisection to refine elements, where the elements marked for refinement are refined using quadrisection while the resulting nonconforming border elements are fixed using bisection. To reduce over-refinement in any one level, only those elements whose element indicators are larger than the mean plus one standard deviation of all of the element indicators in that level are refined.

EXAMPLE 2.6. To illustrate the accuracy that characterizes this approach to *a posteriori* error estimation, we consider the elliptic problem on the unit square $\Omega = (0,1) \times (0,1)$,

$$\begin{cases} -\Delta u = 200 \sin(10\pi x) \sin(10\pi y), & (x,y) \in \Omega, \\ u(x,y) = 0, & (x,y) \in \partial\Omega, \end{cases} \tag{2.13}$$

which has the highly oscillatory solution

$$u(x,y) = \sin(10\pi x) \sin(10\pi y).$$

We plot the solution in Fig. 2.2 We estimate the error in the average value by

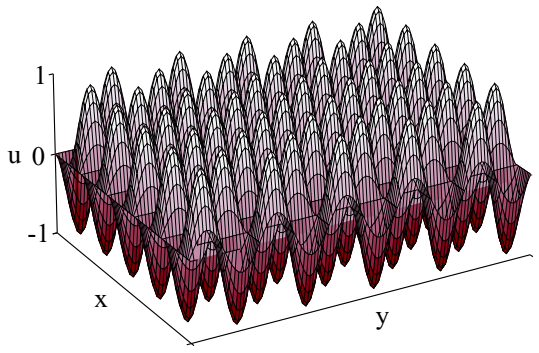

FIGURE 2.2. The highly oscillatory solution of (2.13).

choosing $\psi \equiv 1$. The generalized Green's function is approximated on the same mesh using a piecewise quadratic finite element function. To show how the accuracy in the estimate varies with respect to the resolution in the mesh, we plot the ratios

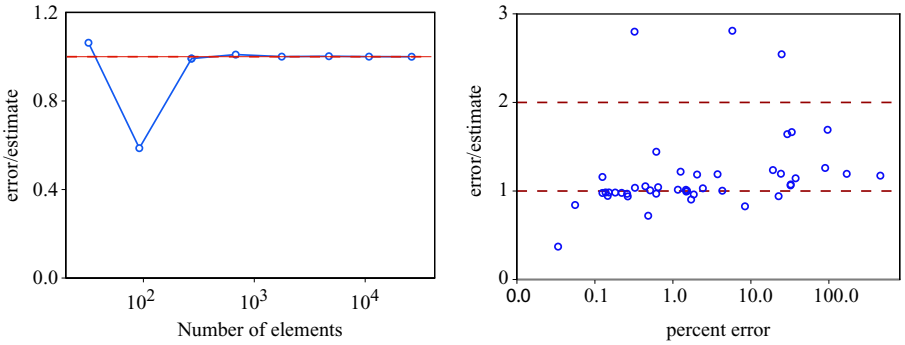

FIGURE 2.3. Plots of the error/estimate ratio for numerical solutions of (2.13). In the plot on the left, we show the sequence of ratios corresponding to regular sequence of uniformly refined meshes. On the right, we plot the ratio for a "random" collection of uniform meshes of various resolutions.

error/estimate in Fig. 2.3 for a wide variety of meshes. We see that the estimate is remarkable accurate even on meshes in which the solution is very poorly resolved.

It is natural to wonder why some of the computations give low or high ratios. Factors that can affect verification results include effects of superconvergence or some special cancellation of errors that arises from the choice of quadratures used to evaluate the error and/or the estimate. For example, the low ratio in the plot on the left occurs for a computation on a uniform mesh with 10 elements on a side, while the true solution oscillates with a frequency proportional to 10.

2.5. Adaptive error control

We now discuss briefly the use of *a posteriori* error estimates for the purpose of adaptive error control. We start by pointing out that despite the huge amount of literature on adaptive error control, there is actually very little theory underlying adaptive error control using *accurate* estimates. All current adaptive error control strategies share the same flaw. Nonetheless, adaptive error control proves useful in practice on many kinds of problems.

A typical goal of adaptive error control is to generate a mesh with a relatively small number of elements such that for a given tolerance TOL and data ψ,

$$|(e,\psi)| \leq \text{TOL}. \tag{2.14}$$

We note that (2.14) cannot be verified in practice because the error is unknown. We must use an error estimate. For the purpose of implementing (2.12) to obtain a computational error estimate and for adaptive error control, we rewrite it as a sum of element contributions,

$$(e,\psi) \approx \sum_{K\in\mathcal{T}_h} \int_K \big((f - b\cdot\nabla U - cU)(\Phi - \pi_h\Phi) - a\nabla U\cdot\nabla(\Phi-\pi_h\Phi)\big)\,dx. \tag{2.15}$$

We use (2.15) to replace (2.14) with the practical goal of satisfying the following condition.

DEFINITION 2.7. The **mesh acceptance criterion** is

$$\left| \sum_{K\in\mathcal{T}_h} \int_K \big((f - b\cdot\nabla U - cU)(\Phi - \pi_h\Phi) - a\nabla U\cdot\nabla(\Phi - \pi_h\Phi)\big)\,dx \right| \leq \text{TOL}. \tag{2.16}$$

If the current approximation satisfies (2.16), then the solution is deemed acceptable and the refinement process is stopped.

The difficulties start when (2.16) is not satisfied. We have to decide how to "enrich" the discretization, e.g., refine the mesh or increase the order of the element functions, in order to improve the accuracy. The problem is that generally there is a great deal of cancellation among the contributions from each element. For example, consider that large positive contributions from one subregion might cancel the large negative contributions from another region so that the sum of the contributions from the two regions together is small.

> *There is currently no theory or practical method for accommodating cancellation of errors in an adaptive error control in a way that achieves true optimality of efficiency.*

Currently, the standard approach is to formulate the discretization selection problem as an optimization problem. This requires an estimate consisting of a sum over elements of positive quantities. We obtain this from (2.15) by inserting norms in some way, e.g., we use

$$|(e,\psi)| \leq \sum_{K\in\mathcal{T}_h} \int_K \big|(f - b\cdot\nabla U - cU)(\Phi - \pi_h\Phi) - a\nabla U\cdot\nabla(\Phi - \pi_h\Phi)\big|\,dx. \tag{2.17}$$

Thus, if (2.16) is *not* satisfied, then the mesh is refined in order to achieve the more conservative condition,

$$\sum_{K\in\mathcal{T}_h} \int_K \big|(f - b\cdot\nabla U - cU)(\Phi - \pi_h\Phi) - a\nabla U\cdot\nabla(\Phi - \pi_h\Phi)\big|\,dx \leq \text{TOL}. \tag{2.18}$$

The problem with any claims of "optimal" mesh selection is that generically the estimate obtained from (2.17) is 1-3 orders of magnitude larger than the estimate obtained from (2.15).

In any case, we can now use calculus of variations to derive a condition that gives an optimal mesh. This is called the "Principle of Equidistribution" and it states that the element contributions on a nearly optimal mesh are roughly equal across the elements. Depending on the argument, we may use the following conditions to evaluate each element.

DEFINITION 2.8. Two **element acceptance criteria** for the **element indicators** are

$$\max_K \big|(f - b\cdot\nabla U - cU)(\Phi - \pi_h\Phi) - a\nabla U\cdot\nabla(\Phi - \pi_h\Phi)\big| \lesssim \frac{\text{TOL}}{|\Omega|}, \tag{2.19}$$

or

$$\int_K \big|(f - b\cdot\nabla U - cU)(\Phi - \pi_h\Phi) - a\nabla U\cdot\nabla(\Phi - \pi_h\Phi)\big|\,dx \lesssim \frac{\text{TOL}}{M}, \tag{2.20}$$

where M is the number of elements in $\mathcal{T}_h$.

Computing a mesh using these criteria is usually performed by a "compute-estimate-mark-refine" adaptive strategy that begins with a coarse mesh and then refines those elements on which (2.19) respectively (2.20) fail successively.

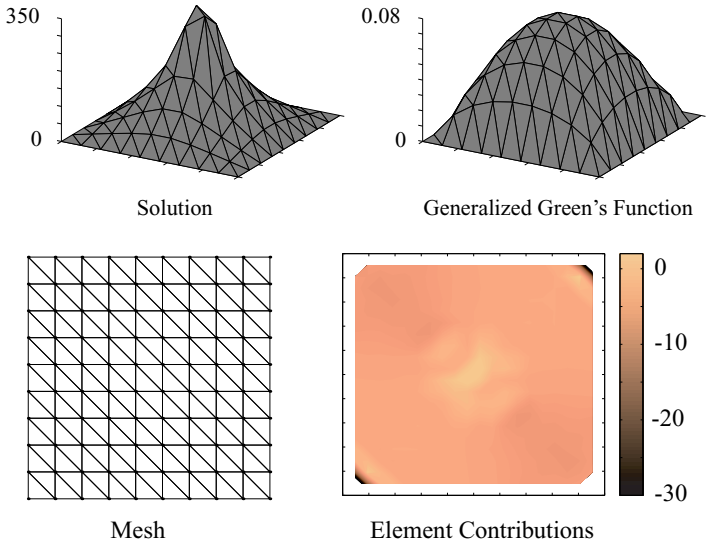

FIGURE 2.4. Plots for the initial refinement level for the computation on (2.21). In the upper set, we plot the solution and the generalized Green's function. In the lower set, we plot the mesh and the element contributions (log scale).

EXAMPLE 2.9. We consider the problem

(2.21)
$$\begin{cases} -\Delta u = \\ \qquad \frac{4800}{\pi}\big(1 - 400((x-.5)^2 + (y-.5)^2)\big)e^{-400((x-.5)^2+(y-.5)^2)}, & (x,y) \in \Omega, \\ u(x,y) = 0, & (x,y) \in \partial\Omega, \end{cases}$$

where the data f is a modified approximation of a delta function for the point $(.5, .5)$. We control the error in the average value using a tolerance of .05%.

We plot the initial 16×16 mesh, solution, generalized Green's function, and the element contributions in Fig. 2.4.

The refinement process took 10 iterations using bisection of the elements. We plot the final mesh, solution, generalized Green's function, and the element contributions in Fig. 2.5. We plot the error/estimate ratio for the sequence of meshes in Fig. 2.6.

2.6. Further analysis on the *a posteriori* error estimate

Most of the literature using this approach to *a posteriori* error estimation does not directly use (2.10) as we have described. Instead, the estimate is massaged analytically. In general, we do not use the estimate that is obtained below in practice. However, we require it in the sequel for a specific example.

For simplicity, we derive the alternative estimate for the simple problem with $L(u) = -\Delta u$. The general result will be clear. In this case, the error representation

formula becomes

$$(e,\psi) = \int_\Omega f(\phi - \pi_h\phi)\,dx - \int_\Omega \nabla U \cdot \nabla(\phi - \pi_h\phi)\,dx. \tag{2.22}$$

Next, we break up the second integral on the right as

$$\int_\Omega \nabla U \cdot \nabla(\phi - \pi_h\phi)\,dx = \sum_{K\in\mathcal{T}_h} \int_K \nabla U \cdot \nabla(\phi - \pi_h\phi)\,dx.$$

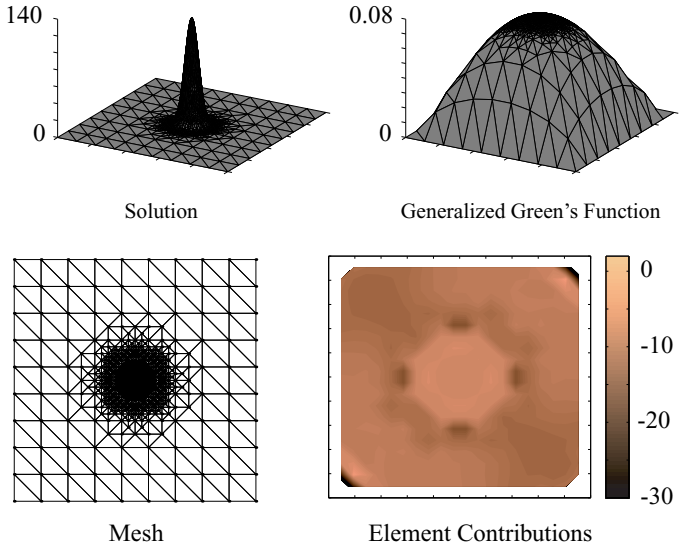

FIGURE 2.5. Plots for the initial refinement level for the computation on (2.21). In the upper set, we plot the solution and the generalized Green's function. In the lower set, we plot the mesh and the element contributions (log scale).

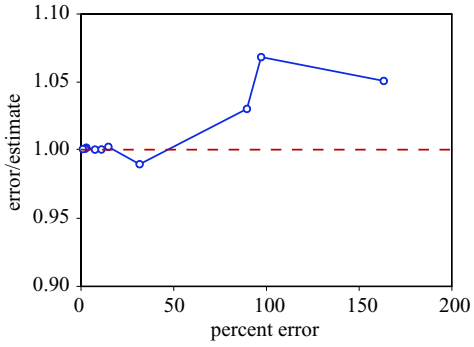

FIGURE 2.6. The error/estimate ratio for the computations for (2.21).

Using Green's formula, we have

$$\int_K \nabla U \cdot \nabla(\phi - \pi_h\phi)\,dx = -\int_K \Delta U(\phi - \pi_h\phi)\,dx + \int_{\partial K} \nabla U \cdot n_{\partial K}(\phi - \pi_h\phi)\,ds,$$

where the last term is a line integral and $n_{\partial K}$ denotes the outward normal to ∂K.

Upon summing over all elements $K \in \mathcal{T}_h$, the boundary integrals give two contributions from each element edge, computed in opposite directions. Suppose $K_1, K_2 \in \mathcal{T}_h$ share a common edge $\sigma_1 \subset \partial K_1 = \sigma_2 \subset \partial K_2$. The contribution from that edge is

$$\begin{aligned}\int_{\sigma_1} \nabla U|_{K_1} \cdot n_{\sigma_1}(\phi - \pi_h\phi)\,ds &+ \int_{\sigma_2} \nabla U|_{K_2} \cdot n_{\sigma_2}(\phi - \pi_h\phi)\,ds \\ &= \int_{\sigma_1} \nabla U|_{K_1} \cdot n_{\sigma_1}(\phi - \pi_h\phi)\,ds - \int_{\sigma_1} \nabla U|_{K_2} \cdot n_{\sigma_1}(\phi - \pi_h\phi)\,ds \\ &= -\int_{\sigma_1} [\nabla U] \cdot n_{\sigma_1}(\phi - \pi_h\phi)\,ds,\end{aligned}$$

where $[U] = \nabla U|_{K_2} - \nabla U|_{K_1}$ denotes the "jump" in ∇U across σ_1 in the direction of the normal $n_{\partial K_1}$.

When summing over the elements, we associate half of the common contribution across a shared edge between two elements with each element. We obtain an alternate error representation,

$$(e, \psi) = -\sum_{K \in \mathcal{T}_h} \left(\int_K (\Delta U + f)(\phi - \pi_h\phi)\,dx - \frac{1}{2}\int_{\partial K} [\nabla U] \cdot n_{\partial K}(\phi - \pi_h\phi)\,ds \right).$$

Finally, we define the **residual** and corresponding **adjoint** or **dual weights**,

$$\mathcal{R}_K = \begin{pmatrix} \|\Delta U + f\|_K \\ \|h^{-1/2}[\nabla U]\|_{\partial K}/2 \end{pmatrix}, \; \mathcal{W}_K = \begin{pmatrix} \|\phi - \pi_h\phi\|_K \\ \|h^{1/2}(\phi - \pi_h\phi)\|_{\partial K} \end{pmatrix}. \tag{2.23}$$

We obtain an *a posteriori* error bound similar to the results in [**EJ91**] and repeated in many later references,

THEOREM 2.10. *The error of the finite element approximation is bounded by*

$$|(e, \psi)| \le \sum_{K \in \mathcal{T}_h} \mathcal{R}_K \cdot \mathcal{W}_K.$$

It is possible to obtain *a priori* bounds on the residual and dual weights. First, note that there is a constant C independent of the mesh such that

$$\mathcal{R}_K \le C|K|^{1/2},$$

where $|K|$ denote the area of $K \in \mathcal{T}_h$. The bound on the first component of $\mathcal{R}_K$ is simple, $\|\Delta U + f\|_K = \|f\|_K \le \max_\Omega |f| \times |K|^{1/2}$. To bound the second component, consider an integral over the common edge σ between two elements K_1 and K_2,

$$\|[\nabla U]\|_\sigma = \|\nabla U|_{K_2} - \nabla U|_{K_1}\|_\sigma \le \|\nabla U|_{K_2} - \nabla u|_\sigma\|_\sigma + \|\nabla u|_\sigma - \nabla U|_{K_1}\|_\sigma.$$

By a trace inequality, the standard energy norm convergence result, and a standard elliptic regularity result, we have

$$\begin{aligned}\|\nabla U|_{K_i} - \nabla u|_\sigma\|_\sigma &\le \|\nabla U - \nabla u\|_{K_i}^{1/2} \|\nabla U - \nabla u\|_{1,K_i}^{1/2} \le C\|hu\|_{2,K_i}^{1/2} \|u\|_{2,K_i}^{1/2} \\ &\le C\|h^{1/2} f\|_{K_i},\end{aligned}$$

for $i = 1, 2$. The local quasi-uniformity of the mesh implies $\frac{1}{2}\|h^{-1/2}[\nabla U]\|_{\partial K} \leq C \max_\Omega |f| \times |K|^{1/2}$.

Clearly, the convergence of the Galerkin approximation is strongly influenced by the dual weights $\phi - \pi_h\phi$, i.e. by the approximation properties of V_h and the smoothness of ϕ. This reflects the importance of the cancellation of errors inherent to the Galerkin method.

CHAPTER 3

The Effective Domain of Influence and Solution Decomposition

A characteristic property of elliptic partial differential equations is a *global* domain of influence. That is, a local perturbation of data near one point affects the solution throughout the domain of the problem. Indeed, in the extreme case of an analytic harmonic function, prescribing the values of a solution on any small sub-domain or even on a piece of curve suffices to define its values throughout the domain. Of course, this property has profound consequences for the numerical solution of elliptic equations.

Yet when taken out of context, this property can give a misleading impression. In particular, elliptic problems often have the property that the strength of the effect of a localized perturbation on a solution decays significantly with the distance from the support of the perturbation, at least in some directions. It turns out that this property also has profound consequences for the numerical solution of elliptic problems, which we explore in this chapter.

One way to see the decay of influence in an elliptic problem is to use the properties of Green's functions. We want to analyze the effects of perturbations on the data. Consider the Green's function for the Dirichlet problem for the Laplacian $-\Delta u = f$ on the ball Ω of radius r centered at the origin in $\mathbb{R}^3$ discussed in Ex. 1.81. If the data f is perturbed by a smooth function δf, the perturbation in the value of the solution $\delta u(y)$ is given by

$$\delta u(y) = \int_\Omega \phi(y;x)\delta f(x)\,dx, \quad y \in \Omega. \tag{3.1}$$

We use the formula for the Green's function (1.11) to conclude that if δf has compact support $\operatorname{supp}(\delta f) \subset \Omega$, then

$$|y - x| \leq \left| \frac{r^2 y}{|y|^2} - x \right|, \quad x \in \operatorname{supp}(\delta f),\, y \in \Omega \setminus \operatorname{supp}(\delta f).$$

We conclude that

$$|\delta u(y)| \leq \frac{\max|\delta f| \,\times\, \text{volume of } \operatorname{supp}(\delta f) \times \left(1 + \frac{r}{|y|}\right)}{4\pi \times \text{the distance from } y \text{ to } \operatorname{supp}(\delta f)},$$

and the effects of a local perturbation in the data decays with the distance to the support of the perturbation.

In this chapter, we explore the consequences of the decay of influence for the numerical solution of elliptic problems. Using the generalized Green's function, we define the notion of an *effective* domain of influence. In order to achieve accuracy in the desired quantity, a mesh must be sufficiently refined inside the effective domain

of influence, while outside the effective domain, the mesh may be relatively coarse. This turns out to be useful in terms of computing efficiently.

3.1. A concrete example: Poisson's equation in a disk

To introduce the ideas in a concrete way and to demonstrate how the decay of influence can affect the accuracy of a finite element solution, we analyze an example for which there is a formula for the Green's function. We let Ω denote the disk of radius r centered at the origin in $\mathbb{R}^2$, and consider the Dirichlet problem for the Laplacian $L = -\Delta u = f$. Suppose that ω is a small region contained in Ω located well away from $\partial\Omega$ and that we wish to estimate the error $e = u - U$ in the energy norm $\|e\|_{1,\omega}$ in ω. See Fig. 3.1. Recall that we can evaluate the norm weakly via

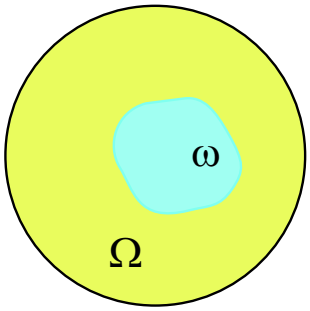

FIGURE 3.1. We wish to estimate the error in the energy norm in the region ω.

Theorem 1.37 as

$$\|e\|_{1,\omega} = \sup_{\substack{\psi \in H^{-1}(\omega) \\ \|\psi\|_{-1,\omega}=1}} (e, \psi). \tag{3.2}$$

By the Riesz Representation theorem, the supremum is achieved for some $\psi \in H^{-1}(\omega)$. We extend this ψ to $H^{-1}(\Omega)$ by setting it to zero in $\Omega \setminus \omega$. We use this function to define the generalized Green's function.

We use the *a posteriori* bound in Theorem 2.10 and the subsequent analysis on the factors in the bound. We use the formula for the Green's function on the disk given in (1.12) to analyze the behavior of the generalized Green's function ϕ. If we let $G(x; y)$ denote the Green's function for the Laplacian on Ω, then

$$\phi(x) = \int_\Omega G(x; y)\psi(y)\, dy = \int_\omega G(x; y)\psi(y)\, dy.$$

There are two cases to consider. For $y \in \omega$, $G(x; y)$ is a smooth function of x for $x \in \Omega \setminus \omega$, and therefore so is ϕ. We assume that $\delta > 0$ is small enough that $\omega_\delta = \{x \in \Omega : \text{dist}\,(x, \omega) \le \delta\}$ is contained in Ω, but large enough that for $K \subset \Omega \setminus \omega_\delta$, the union $\mathcal{N}(K)$ of K and the elements bordering K does not intersect ω, see Fig. 3.2. For $K \subset \Omega \setminus \omega_\delta$, we let π_h be the Lagrange nodal interpolant with respect to $\mathcal{T}_h$, so that

$$\|\phi - \pi_h\phi\|_K \le C \sum_{|\alpha|=2} \|h^2 D^\alpha \phi\|_K.$$

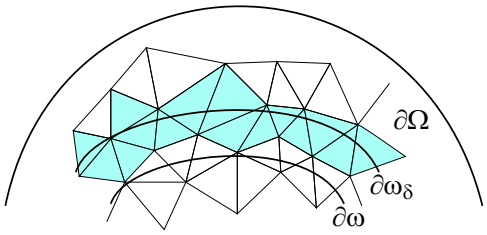

FIGURE 3.2. The choice of ω_δ.

On the other hand, ϕ is not so smooth in ω, and in particular, is only in $H^1(\omega)$ in general. We have to use an averaging approximation that allows for an estimate requiring less smoothness. For $K \cap \omega_\delta \neq \emptyset$, we let π_h be the Scott-Zhang interpolant ([**BS94**]), for which

$$\|\phi - \pi_h\phi\|_K \leq C|h\phi|_{1,\mathcal{N}(K)},$$

for a mesh-independent constant C.

The second component of $\mathcal{W}_K$ is bounded similarly after using a trace theorem,

$$\|h^{1/2}(\phi - \pi_h\phi)\|_{\partial K} \leq \|\phi - \pi_h\phi\|^{1/2}_{\mathcal{N}(K)} \|h(\phi - \pi_h\phi)\|^{1/2}_{1,\mathcal{N}(K)},$$

and the local quasi-uniformity of the mesh. We conclude,

THEOREM 3.1. *For any $\delta > 0$ small enough that $\omega_\delta \subset \Omega$ but large enough that $\mathcal{N}(K) \cap \omega = \emptyset$ for $K \subset \Omega \setminus \omega_\delta$, there is a constant C such that*

$$\|e\|_{1,\omega} \leq \sum_{K \subset \Omega \setminus \omega_\delta} \sum_{|\alpha|=2} C\|h^2 D^\alpha \phi\|_K |K|^{1/2} + \sum_{K \cap \omega_\delta \neq \emptyset} C|h\phi|_{1,\mathcal{N}(K)} |K|^{1/2}. \tag{3.3}$$

To understand the implications of (3.3) for mesh selection in an adaptive setting, we further estimate the quantities on the right in (3.3). To handle the first sum, we estimate the derivatives using the Green's function as

$$\begin{aligned}
\|D_x^\alpha \phi\|_K^2 &= \int_K \left(\int_\omega D_x^\alpha G(x,y)\psi(y)\, dy \right)^2 dx \leq \int_K \|D_x^\alpha G(x,\cdot)\|^2_{1,\omega} \|\psi\|^2_{-1,\omega}\, dx \\
&= \sum_{|\beta|=1} \int_K \int_\omega |D_x^\alpha D_y^\beta G(x,y)|^2\, dydx + \int_K \int_\omega |D_x^\alpha G(x,y)|^2\, dydx.
\end{aligned}$$

The formula (1.12), there is a constant C such that

$$|D_x^\alpha D_y^\beta G(x,y)| \leq \frac{C}{|x-y|^2}, \quad x \neq y \in \Omega,\ |\alpha| = 2,\ |\beta| \leq 1.$$

We conclude there is a constant C independent of the mesh such that for $K \subset \Omega \setminus \omega_\delta$,

$$\|\phi - \pi_h\phi\|_K \leq \frac{Ch_K^2}{\operatorname{dist}(K,\omega)^2} |K|^{1/2}.$$

To handle the second sum on the right of (3.3), we use the basic stability estimate,

$$\|\phi\|_{1,\Omega} \leq \|\psi\|_{-1,\Omega} = \|\psi\|_{-1,\omega} = 1.$$

If we assume a uniform (small) size $h_K = \underline{h}$ for elements such that $K \cap \omega_\delta \neq \emptyset$, we obtain

$$\sum_{K\cap\omega_\delta\neq\emptyset} Ch_K|\phi|_{1,\mathcal{N}(K)} \leq C\underline{h}\|\phi\|_{1,\Omega} = C\underline{h} = \frac{C}{|\omega_\delta|}\sum_{K\cap\omega_\delta\neq\emptyset}\underline{h}|K|.$$

We conclude

THEOREM 3.2. *For any $\delta > 0$ small enough that $\omega_\delta \subset \Omega$ but large enough that $\mathcal{N}(K) \cap \omega = \emptyset$ for $K \subset \Omega \setminus \omega_\delta$, there is a constant C such that*

$$\|e\|_{1,\omega} \leq \sum_{K\subset\Omega\setminus\omega_\delta} \frac{Ch_K^2}{\operatorname{dist}(K,\omega)^2}|K| + \sum_{K\cap\omega_\delta\neq\emptyset} C\underline{h}|K|. \tag{3.4}$$

Applying the "Principle of Equidistribution", we take the element contributions to the error to be approximately equal in order to obtain a nearly optimal mesh. In (3.4), the element indicators are $Ch_K^2/\operatorname{dist}(K,\omega)^2$ respectively $C\underline{h}$, and in an optimal mesh,

$$\frac{h_K^2}{\operatorname{dist}(K,\omega)^2} \approx \underline{h} \quad \text{or} \quad h_K \approx \underline{h}^{1/2} \times \operatorname{dist}(K,\omega), \quad K \subset \Omega \setminus \omega_\delta.$$

The decay of influence inherent to the Laplacian on the disk means that away from the region ω where we estimate the norm, we can choose elements asymptotically larger than the element size used in ω_δ. Moreover, the elements can increase in size as the distance to ω_δ increases. In this problem, ω_δ is an the effective domain of influence for the error in the energy norm in ω. We extend this definition in general.

DEFINITION 3.3. An **effective domain of influence** corresponding to the data ψ is the region ω_ψ in which the corresponding elements *must* be significantly smaller in size than the elements used in the complement $\Omega \setminus \omega_\psi$ in order to satisfy (2.16). Equivalently, if $\mathcal{T}_h$ comprises uniformly sized elements, then the effective domain of influence comprises those elements on which the element indicators (2.19), alternatively (2.20), are substantially larger than those in the complement.

3.2. A decomposition of the solution

It is often the case that the goal of solving a differential equation is to compute several pieces of information. For example, we might wish to compute values of the solution at a number of points and internal boundaries. In this section, we explain how the problem of computing multiple quantities of interest also arises naturally when the data ψ for the adjoint problem does not have spatially localized support, such as the case when the quantity of interest is an average or norm over the domain Ω for example.

The motivation is that we cannot expect a significant localization effect from the decay of influence when the support of the data for the adjoint problem is not spatially localized. Recall from Sec. 3.1 that the decay of influence was determined by the adjoint weighting factor $\phi - \pi_h\phi$. If the data ψ has the property that the corresponding adjoint weight $\phi - \pi_h\phi$ has a more-or-less uniform size throughout Ω, then the degree of non-uniformity in an adapted mesh depends largely on the spatial variation of the residual.

However, we can use a partition of unity to "localize" a problem in which $\operatorname{supp}(\psi)$ does not have local support.

DEFINITION 3.4. Suppose that $\{\Omega_i\}_{i=1}^N$ is a finite open cover of Ω. A **Lipschitz partition of unity** subordinate to $\{\Omega_i\}$ is a collection of functions $\{p_i\}_{i=1}^N$ with the properties

$$\text{supp}\,(p_i) \subset \overline{\Omega}_i, \quad 1 \le i \le N, \quad \sum_{i=1}^N p_i(x) = 1, \quad x \in \Omega, \tag{3.5}$$

$$p_i \text{ is continuous on } \Omega \text{ and differentiable on } \Omega_i, \quad 1 \le i \le N, \tag{3.6}$$

$$\|p_i\|_{L^\infty(\Omega)} \le C \text{ and } \|\nabla p_i\|_{L^\infty(\Omega_i)} \le C/\text{diam}\,(\Omega_i), \quad 1 \le i \le N, \tag{3.7}$$

where C is a constant and $\text{diam}\,(\Omega_i)$ is the diameter of Ω_i.

Several partitions of unity satisfying (3.5)-(3.7) exist, see e.g. [**GS00**].

We use a partition of unity $\{p_i\}$ to write $\psi \equiv \sum_{i=1}^N \psi p_i$. This suggests:

DEFINITION 3.5. The quantities $\{(U, \psi p_i)\}$ corresponding to the data $\{\psi_i = \psi p_i\}$ are called the **localized information** corresponding to the partition of unity.

We now consider the problem of estimating the error in the localized information for $1 \le i \le N$. Correspondingly, we obtain a finite element solution via:

$$\text{Compute } \hat{U}_i \in \hat{V}_i \text{ such that } A(\hat{U}_i, v) = (f, v) \text{ for all } v \in \hat{V}_i, \tag{3.8}$$

where $\hat{V}_i$ is a space of continuous, piecewise linear functions on a locally quasi-uniform simplex triangulation $\mathcal{T}_i$ of Ω obtained by (presumably local) refinement of an initial coarse triangulation $\mathcal{T}_0$ of Ω. We emphasize that the space $\{\hat{V}_i\}$ is globally defined and the "localized" problem (3.8) is solved over the entire domain, though we hope that (3.8) will require a locally refined mesh because the corresponding data has localized support.

We can obtain a partition of unity approximation in the sense of Babuška and Melenk [**BM97**] by defining the truly local approximations $U_i = \chi_i \hat{U}_i$, $1 \le i \le N$, where χ_i is the characteristic function of Ω_i. The local approximation U_i is in the local finite element space $V_i = \chi_i \hat{V}_i$.

DEFINITION 3.6. The **partition of unity approximation** is defined by $U_p = \sum_{i=1}^N U_i p_i$, which is in the **partition of unity finite element space**

$$V_p = \sum_{i=1}^N V_i p_i = \left\{ \sum_{i=1}^N v_i p_i : v_i \in V_i \right\}.$$

The basic convergence results for this method are proved in [**Hol01**] and [**Hol02**] using ideas of Babuška and Melenk [**BM97**] and Xu and Zhou [**XZ00**]. The upshot is that the partition of unity approximation recovers the full convergence properties of an approximation of the original solution. Note that

$$U_p = \sum_{i=1}^N U_i p_i = \sum_{i=1}^N \chi_i \hat{U}_i p_i \equiv \sum_{i=1}^N \hat{U}_i p_i.$$

In words, the values of U_i or $\hat{U}_i$ outside of Ω_i are immaterial in forming the global partition of unity approximation.

To estimate the error in the localized information corresponding to ψ_i, we use the generalized Green's function satisfying the adjoint problem:

$$\text{Find } \phi_i \in H_0^1(\Omega) \text{ such that } A^*(v, \phi_i) = (v, \psi_i) \text{ for all } v \in H_0^1(\Omega). \tag{3.9}$$

We expand the global error in the partition of unity approximation as

$$(u - U_p, \psi) = \sum_{i=1}^{N} \big((u - U_i)p_i, \psi\big).$$

We estimate each summand on the right as

$$\begin{aligned}\big((u - U_i)p_i, \psi\big) &= (u - \hat{U}_i, \psi_i) = A^*(u - \hat{U}_i, \phi_i) \\ &= (f, \phi_i) - (a\nabla\hat{U}_i, \nabla\phi_i) - (b \cdot \nabla\hat{U}_i, \phi_i) - (c\hat{U}_i, \phi_i).\end{aligned}$$

Letting $\pi_i\phi_i$ denote an approximation of ϕ_i in $\hat{V}_i$, using Galerkin orthogonality, we conclude

THEOREM 3.7. *The error of the partition of unity finite element solution U_p satisfies the error representation,*

(3.10)

$$\begin{aligned}(u - U_p, \psi) = \sum_{i=1}^{N} \big((f, \phi_i - \pi_i\phi_i) - (a\nabla\hat{U}_i, \nabla(\phi_i - \pi_i\phi_i)) - (b \cdot \nabla\hat{U}_i, \phi_i - \pi_i\phi_i) \\ - (c\hat{U}_i, \phi_i - \pi_i\phi_i)\big),\end{aligned}$$

where ϕ_i is the solution of the adjoint problem (3.9) and $\hat{U}_i$ solves the finite element problem (3.8) corresponding to the localized data ψ_i.

In practice, we compute approximate generalized Green's functions via;

(3.11)

$$\text{Compute } \Phi_i \in V_i^2 \text{ such that } A^*(v, \Phi_i) = (v, \psi_i) \text{ for all } v \in V_i^2, \quad 1 \le i \le N,$$

where V_i^2 is the space of continuous, piecewise quadratic functions with respect to $\mathcal{T}_i$. The corresponding approximate error representation for each computation is

(3.12)

$$\begin{aligned}(u - \hat{U}_i, \psi_i) \approx (f, \Phi_i - \pi_i\Phi_i) - (a\nabla\hat{U}_i, \nabla(\Phi_i - \pi_i\Phi_i)) - (b \cdot \nabla\hat{U}_i, \Phi_i - \pi_i\Phi_i) \\ - (c\hat{U}_i, \Phi_i - \pi_i\Phi_i).\end{aligned}$$

Note that the proof of Theorem 3.7 also implies that if the localized error satisfies

$$\big|\big(u - \hat{U}_i, \psi_i\big)\big| \le \frac{\text{TOL}}{N}, \quad 1 \le i \le N, \tag{3.13}$$

then $|(u - U_p, \psi)| \le \text{TOL}$. This justifies treating the N "localized" problems independently in terms of mesh refinement. Note however that (3.13) is based on the pessimistic assumption that there is no cancellation of errors when combining the "localized" solutions to get the full solution. Using TOL/N for the tolerance for the "localized" solutions turns out to be much too pessimistic in practice. Finding more reasonable tolerances is an interesting problem.

3.3. Efficient computation of multiple quantities of interest

In this section, we develop an algorithm for computing multiple quantities of interest efficiently using knowledge of the effective domains of influence of the corresponding Green's functions. We assume that the information is specified as $\{(U, \psi_i)\}_{i=1}^N$ for a set of N functions $\{\psi_i\}_{i=1}^N$. These data might arise as particular

goals or via localization through a partition of unity. We assume that the goal is to compute the information associated to ψ_i so that the error is smaller than a tolerance TOL_i for $1 \leq i \leq N$.

At least two approaches for this problem come to mind:

Approach 1: **A Global Computation**

Find one triangulation such that the corresponding finite element solution satisfies $|(e, \psi_i)| \leq \text{TOL}_i$, for $1 \leq i \leq N$.

Approach 2: **A Decomposed Computation**

Find N independent triangulations and finite element solutions U_i so that the errors satisfy $|(e_i, \psi_i)| \leq \text{TOL}_i$, for $1 \leq i \leq N$.

Note that the Global Computation can be implemented with a straightforward modification of the standard adaptive strategy in which the N corresponding mesh acceptance criteria are checked on each element and if any of the N criteria fail, the element is marked for refinement.

Generally, if the correlation, i.e., overlap, between the effective domains of influence associated to the N data $\{\psi_i\}$ is relatively small and the effective domains of influence are relatively small subsets of Ω, then each individual solution in the Decomposed Computation will require significantly fewer elements than the solution in the Global Computation to achieve the desired accuracy. This can yield significant computational advantage in terms of lowering the maximum memory requirement to solve the problem. We provide some examples showing the possible gain in Sec. 3.5.

Decreasing the maximum memory required to solve a problem can be significant in at least two situations. First, if the individual solutions in the Decomposed Computation are computed in parallel, then the time needed for the Decomposed Computation is determined roughly by the time it takes to solve for the solution requiring the largest number of elements. If the individual solutions in the Decomposed Computation require significantly fewer elements than the Global Computation, we can expect to see significant speedup. Second, if we are solving in an environment with limited memory capabilities, then decomposing a Global Computation requiring a large number of elements into a set of significantly smaller computations can greatly increase the accuracy of the solution that can be computed and/or decrease the time of solution. In this case, the individual solutions in the Decomposed Computation may be computed serially.

Vice versa, if the effective domains of influence associated to the N data $\{\psi\}$ have relatively large intersections, then the individual solutions in the Decomposed Computation will require roughly the same number of elements as the solution for the Global Computation. In this case, there is little to be gained in using the Decomposed Computation. In general, we can expect that some of the N effective domains of influence associated to data $\{\psi_i\}$ in the Decomposed Computation will correlate significantly and the rest will have low correlation. We can optimize the use of resources by combining computations for data whose associated domains of influence have significant correlation and treating the rest independently.

An algorithm for the decomposition of the solution process using effective domains of influence is:

ALGORITHM 3.8. **Determining the Solution Decomposition**

(1) Discretize Ω by an initial coarse triangulation $\mathcal{T}_0$ and compute an initial finite element solution U_0.

(2) Estimate the error in each quantity (U_0, ψ_i) by solving the N approximate adjoint problems (3.11) and then using (3.12).
(3) Using the element indicators associated to (3.12) to identify the effective domains of influence for the data $\{\psi_i\}$ in terms of the mesh $\mathcal{T}_0$ and significant correlations between the effective domains of influence.
(4) Decide on the number of approximate solutions to be computed and the subset of information to be computed from each solution.
(5) Compute the approximate solutions independently using adaptive error control aimed at computing the specified quantity or quantities of interest accurately.

We address the key step 3. in the practical implementation of this algorithm in Sec. 3.4.

3.4. Identifying significant correlations

The key issue in implementing Algorithm 3.8 is identifying the effective domains of influence for the various generalized Green's functions and recognizing significant correlation, or overlap, between different effective domains of influence in Step 3. In this section, we present a method to do this.

Recall that the mesh refinement decisions are based on the sizes of the element indicators on element K,

$$\mathcal{E}_i|_K = \max_K \left|(f - b\cdot\nabla\hat{U}_i - c\hat{U}_i)(\Phi_i - \pi_i\Phi_i) - a\nabla\hat{U}_i\cdot\nabla(\Phi_i - \pi_i\Phi_i)\right| \tag{3.14}$$

or

$$\mathcal{E}_i|_K = \int_K \left|(f - b\cdot\nabla\hat{U}_i - c\hat{U}_i)(\Phi_i - \pi_i\Phi_i) - a\nabla\hat{U}_i\cdot\nabla(\Phi_i - \pi_i\Phi_i)\right| dx, \tag{3.15}$$

associated to the estimate (3.12).

Definition 3.9. We let $\mathcal{E}_i(x)$ denote the piecewise constant **element error indicator function** associated to data ψ_i with $\mathcal{E}_i(x) \equiv \mathcal{E}_i|_K$ for $K \in \mathcal{T}_0$.

Identifying the effective domain of influence associated to a data means finding a set of elements on which the element error indicators are significantly larger than on the complement, if such a dichotomy exists. Identifying significant correlation between the effective domains of influence of two data entails showing that the effective domains of influence have a significant number of elements in common.

To do this, we borrow techniques from pattern matching in signal processing. Of particular importance is the **(cross-)correlation** of two functions $f \in L^p(\Omega)$ and $g \in L^q(\Omega)$, defined as:

$$(f \circ g)(y) = \int_\Omega f(x)g(y+x)\,dx,$$

which is an $L^1(\Omega)$ function. In template matching algorithms used in image and signal processing, the correlations between an input signal and a library of signals are computed and the closest match from the library is the signal containing the "largest" correlation function in some measure. Since each correlation function is itself a real-valued function of n variables, determining the goodness of a match requires computing some real-valued **correlation indicator** $c(f, g)$ of the correlation function $(f \circ g)$, which is typically an L^p-norm.

For the problem of recognizing correlation between effective domains of influence, we treat the element error indicator functions $\{\mathcal{E}_i\}$ as signal functions. In this case, if one signal matches the other signal only after a translation or rotation, we do not consider the functions to be well correlated since this coincides with two primarily disjoint effective domains of influence. Without translation or rotation, correlation of $\mathcal{E}_i$ and $\mathcal{E}_j$ reduces to the L^2-inner-product:

$$(\mathcal{E}_i \circ \mathcal{E}_j)(0) = \int_\Omega \mathcal{E}_i(x)\mathcal{E}_j(x)\,dx = (\mathcal{E}_i, \mathcal{E}_j)_\Omega.$$

The correlation function evaluated at $u = 0$ is just a real number, so that the correlation indicator $c(\mathcal{E}_i, \mathcal{E}_j)$ can be taken as $c(\mathcal{E}_i, \mathcal{E}_j) = |(\mathcal{E}_i \circ \mathcal{E}_j)(0)| = (\mathcal{E}_i, \mathcal{E}_j)_\Omega$.

We mark the effective domain of influence associated to ψ_i as significantly correlated to the domain of influence associated to ψ_j if two conditions hold:

(1) The correlation of $\mathcal{E}_i$ and $\mathcal{E}_j$ is larger than a fixed fraction of the norm of $\mathcal{E}_j$, or mathematically,

$$\text{Correlation Ratio 1} = \frac{c(\mathcal{E}_i, \mathcal{E}_j)}{\|\mathcal{E}_j\|^2} \geq \gamma_1, \tag{3.16}$$

for some fixed $0 \leq \gamma_1 \leq 1$. This means that the projection of $\mathcal{E}_i$ onto $\mathcal{E}_j$ is sufficiently large.

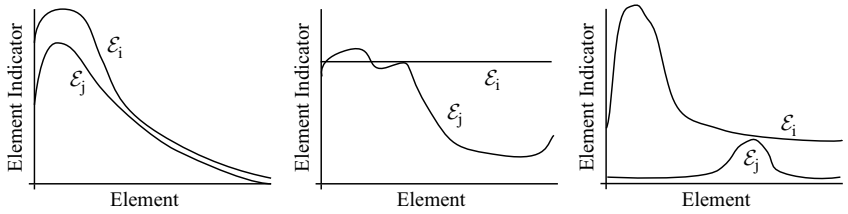

FIGURE 3.3. Three examples of significant correlation of $\mathcal{E}_i$ with $\mathcal{E}_j$. Plotted are the element indicator functions $\mathcal{E}_i(x), \mathcal{E}_j(x)$ versus the element number.

(2) The component of $\mathcal{E}_j$ orthogonal to $\mathcal{E}_i$ is smaller than a fixed fraction of the norm of $\mathcal{E}_j$, or mathematically,

$$\text{Correlation Ratio 2} = \frac{\left\| \mathcal{E}_j - \dfrac{c(\mathcal{E}_j, \mathcal{E}_i)}{\|\mathcal{E}_i\|^2}\mathcal{E}_i \right\|}{\|\mathcal{E}_j\|} \leq \gamma_2, \tag{3.17}$$

for some fixed $0 \leq \gamma_2 \leq 1$. This corrects for the potential difficulties in the mesh refinement decision that arise when $\mathcal{E}_i$ is much larger than $\mathcal{E}_j$ and the corresponding computations are combined.

We illustrate these definitions with a simple example.

EXAMPLE 3.10. In Fig. 3.5, we plot a number of artificial element indicator functions $\{\mathcal{E}_i\}$ versus the element number. Applying conditions 1 and 2 with $\gamma_1 = .9$ and $\gamma_2 = .7$ yields the significant correlations:

$\mathcal{E}_1$ with $\mathcal{E}_8$	$\mathcal{E}_4$ with none	$\mathcal{E}_7$ with none
$\mathcal{E}_2$ with $\mathcal{E}_6, \mathcal{E}_7$	$\mathcal{E}_5$ with $\mathcal{E}_2, \mathcal{E}_6$	$\mathcal{E}_8$ with none
$\mathcal{E}_3$ with $\mathcal{E}_1, \mathcal{E}_8$	$\mathcal{E}_6$ with none	$\mathcal{E}_9$ with none

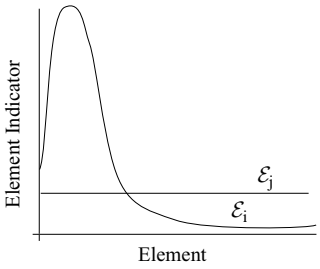

FIGURE 3.4. An example in which condition 2 fails. Plotted are the element indicator functions $\mathcal{E}_i(x), \mathcal{E}_j(x)$ versus the element number.

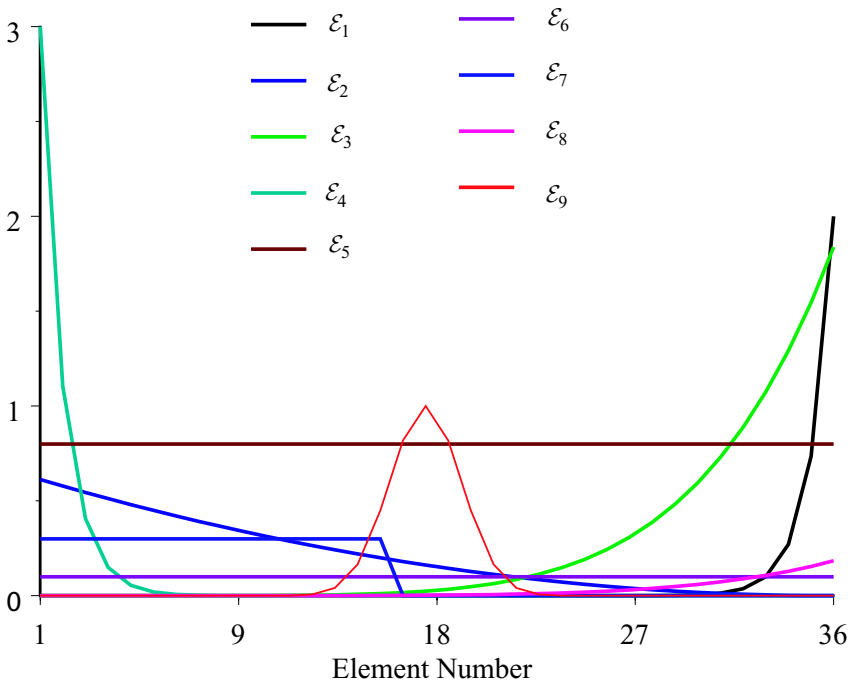

FIGURE 3.5. Plots of nine element indicator functions $\mathcal{E}_i$ versus the element number.

We investigate the properties of these definitions further in Ex. 3.5.3.

We emphasize that the initial identification of significant correlation between effective domains of influence of various Green's functions in a computation is carried out on a coarse initial partition of the domain and hence is relatively inexpensive.

3.5. Examples

We now present several computational examples illustrating and testing the ideas discussed above. In these experiments, we solve various elliptic problems using adaptive mesh refinement to achieve a specified accuracy in a specified set of quantities of interest first using a Global Computation and then using a Decomposed Computation implemented using Algorithm 3.8. The results suggest that the individual solutions in the Decomposed Computation require significantly fewer elements to achieve the desired accuracy than the Global Computation in a variety of situations.

Determining the overall gain in efficiency or capability due to reducing the number of elements to achieve a desired accuracy is difficult. In general, the principle factors determining the time it takes for a solution to be computed, including the solution of the nonlinear system determining the approximation, the marking and refinement of meshes in each refinement level, and, in a massively parallel setting, the IO of the data, all scale super-linearly with the number of elements. Moreover, these factors depend heavily on the algorithm, implementation, and machine. So, as a relatively universal measure of the gain from using the Decomposed Computation, we use

Definition 3.11. The *Final Element Ratio* is the number of elements in the final mesh refinement level required to achieve the specified accuracy in the specified quantities of interest in the Global Computation to the *maximum* number of elements in the final mesh refinement levels for the individual computations in the Decomposed Computation.

> *Generally, we expect the gain in efficiency to scale super-linearly with the Final Element Ratio.*

We compute the Final Element Ratio using solutions that are have roughly the same accuracy. In some cases, this may mean adjusting the tolerance and/or the number of elements in the initial mesh in order to achieve the desired accuracy. Generally, the actual error of solutions depends smoothly on the number of elements, but since we do not un-refine elements, the number of elements does not vary smoothly with the tolerance. So, it is better to compare solutions of approximately the same accuracy rather than solutions computed with the same tolerance.

3.5.1. Example 1. In the first example, we test the partition of unity decomposition of a solution aimed at computing information corresponding to data with global support. We approximate u satisfying the Poisson problem with smooth data,

$$\begin{cases} -\frac{1}{10\pi^2}\Delta u(x) = \sin(\pi x)\sin(\pi y), & (x,y) \in \Omega, \\ u(x,y) = 0, & (x,y) \in \partial\Omega, \end{cases} \tag{3.18}$$

on the domain $\Omega = [0,8] \times [0,8]$. The solution is $u(x,y) = 5\sin(\pi x)\sin(\pi y)$. We solve this problem with the goal of controlling the error in the average value of u by choosing $\psi \equiv 1/|\Omega| = 1/64$.

For the Global Computation, we adapt the mesh so that the error in the average value of u is smaller than the error tolerance of 5%. We begin with an initial mesh of 10×10 elements. After five refinement levels, we end up with 3505 elements,

achieving an error of .022. We plot both the initial and final meshes in Fig. 3.6. We plot the numerical solution on the final mesh in Fig. 3.7.

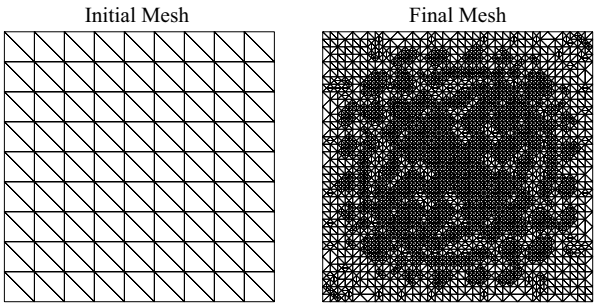

FIGURE 3.6. Initial and final meshes for Example 1 with data ψ giving the average error.

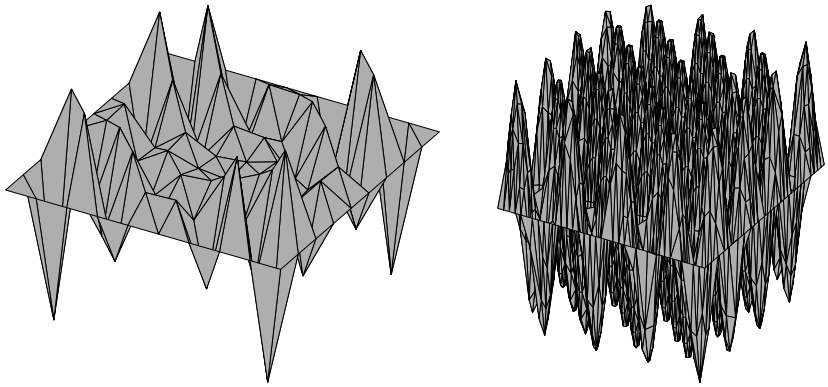

FIGURE 3.7. Numerical solutions on the initial (left) and final (right) meshes for Example 1 with data ψ giving the average error.

Since we know the true solution, we can compute the actual average error and so evaluate the accuracy of the estimate. Below, we list the estimates, errors, and error/estimate ratios:

Level	Elements	Estimate	Error	Ratio
1	100	.1567	.1534	.9786
2	211	.1157	.1224	1.058
3	585	.3063	.3078	1.005
4	1309	.1159	.1166	1.006
5	3505	.02163	.02148	.9975

We see the excellent accuracy of the computed error estimate at all levels of mesh refinement.

For the sake of comparison, we present results for the estimation of the $L^2(\Omega)$ norm of the error. This is possible in this example because the error is known. Hence, we can choose $\psi = e/\|e\|_\Omega$ to get $(e, \psi) = \|e\|_\Omega$. We start the computation with the same 10×10 mesh used above, however we use a tolerance of 1% in order

to get five refinement levels with the number of elements in each refinement level comparable to those used in the computation for the average error. The results are:

Level	Elements	Estimate	Error	Ratio
1	100	12.89	19.19	1.488
2	245	13.36	16.21	1.213
3	681	7.120	7.905	1.110
4	1281	4.729	4.830	1.021
5	3267	1.929	2.008	1.041

Again, the results are rather impressive.

In the rest of the examples, we use average error as a globally-defined goal for estimation. We do this to make it easier to compare results from different examples. We do not have the true error available in some of the examples, and estimating the L^2 norm of the error raises significant issues regarding approximation of the dual data. In the tests we conducted on examples in which the error is known, using the average error and the L^2 norm of the error as globally-defined goals produces the same qualitative results.

The data $\psi \equiv 1/64$ is a natural candidate for localization using a partition of unity. We begin with a partition with the four domains shown on the left in Fig. 3.8. Introducing the corresponding partition of unity yields four data $\{\psi_1, \psi_2, \psi_3, \psi_4\}$

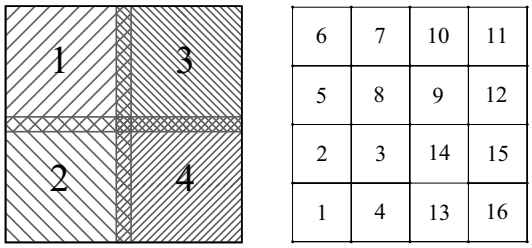

FIGURE 3.8. Domains for the first (left) and second (right) partitions of unity used in Example 1.

corresponding to the regions indicated in Fig. 3.8.

In the first Decomposed Computation, we compute the four localized approximations $\{\hat{U}_1, \cdots, \hat{U}_4\}$ using the same initial mesh as shown in Fig. 3.6. Using $\gamma_1 = .9$ and $\gamma_2 = .5$ in the conditions on the Correlation Ratios (3.16) and (3.17) indicates that all four localized solutions should be computed independently.

For the first Decomposed Computation, we obtain acceptable results using the tolerance of 5%. Details of the final computed solutions are listed below:

Data	Level	Elements	Estimate
ψ_1	3	618	.01242
ψ_2	3	575	−.0009109
ψ_3	3	618	.01242
ψ_4	3	575	−.0009109

Combining these solutions yields a partition of unity solution U_p with accuracy .023. Using the Decomposed Computation yields a Final Element Ratio of $3505/618 \approx 5.7$.

We plot the final meshes for two of the computations in Fig. 3.9. We plot the generalized Green's functions for the global average error and the localized solution

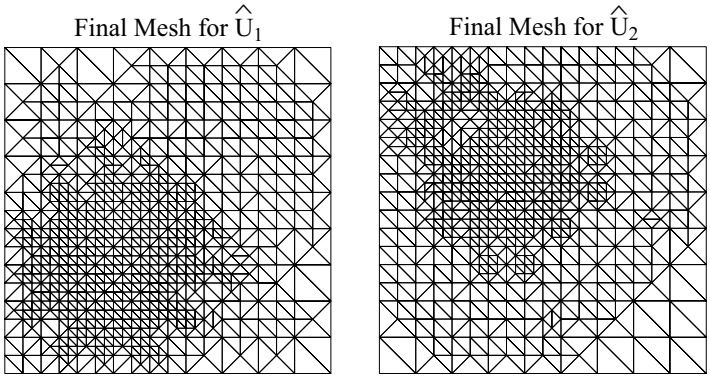

FIGURE 3.9. Final meshes for $\hat{U}_1$ and $\hat{U}_2$ for Example 1 with a partition of unity on four domains.

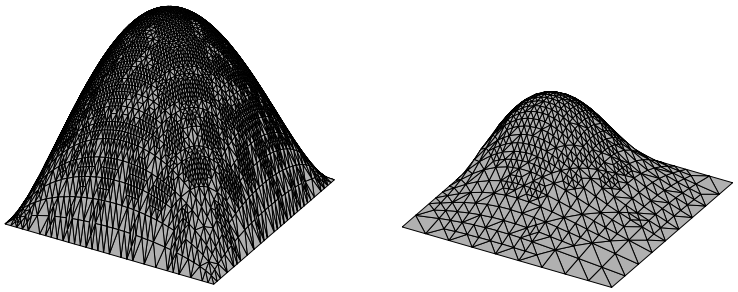

FIGURE 3.10. The generalized Green's functions for the global average error and the localized solution $\hat{U}_2$ corresponding to ψ_2 with a partition of unity on four domains.

corresponding to ψ_2 in Fig. 3.10. The decay of influence away from the support of ψ_2 is clearly visible in the solution on the right.

Next, we perform a Decomposed Computation using a partition of unity on the 16 equal-sized regions shown on the right in Fig. 3.8. We again use an error tolerance of 5% and start the localized computations with the same initial 10×10 mesh used above. Computing the correlation ratios, we find these significant correlations:

$$\begin{array}{cccc} \mathcal{E}_2 \text{ with } \mathcal{E}_3 & \mathcal{E}_5 \text{ with } \mathcal{E}_8 & \mathcal{E}_{10} \text{ with } \mathcal{E}_9 & \mathcal{E}_{13} \text{ with } \mathcal{E}_{14} \\ \mathcal{E}_4 \text{ with } \mathcal{E}_3 & \mathcal{E}_7 \text{ with } \mathcal{E}_8 & \mathcal{E}_{12} \text{ with } \mathcal{E}_9 & \mathcal{E}_{15} \text{ with } \mathcal{E}_{14} \end{array}$$

This suggests that we should see less gain on this partition. We report the results for the accepted approximations:

Data	Level	Elements	Estimate	Data	Level	Elements	Estimate
ψ_1	2	187	$-.0005256$	ψ_9	4	1371	$-.006256$
ψ_2	3	560	.002904	ψ_{10}	3	560	.002904
ψ_3	4	1371	$-.006256$	ψ_{11}	2	187	$-.0005256$
ψ_4	3	560	.002904	ψ_{12}	3	560	.002904
ψ_5	3	569	.001520	ψ_{13}	3	569	.001520
ψ_6	2	212	.002566	ψ_{14}	4	1285	$-.009831$
ψ_7	3	569	.001520	ψ_{15}	3	569	.001520
ψ_8	4	1285	$-.009831$	ψ_{16}	2	212	.002566

In order to obtain an acceptable accuracy in the four sub-domains closest to the center, we have to use an extra refinement level in the computation of the corresponding local solutions. The error in the average of the resulting partition of unity solution is .011. If we use the Decomposed Computation, the most intensive individual computations are those for ψ_3 and ψ_9, which yields a Final Element Ratio of $3505/1371 \approx 2.6$. There is still a significant gain over the Global Computation, but not as large as for the partition with four sub-domains.

3.5.2. Example 2. In the second experiment, we estimate the error in some point values and the average value of u solving

$$\begin{cases} -\nabla \cdot \big((1.1 + \sin(\pi x)\sin(\pi y))\nabla u(x,y)\big) \\ \qquad = -3\cos^2(\pi x) + 4\cos^2(\pi x)\cos^2(\pi x) \\ \qquad\quad +2.2\sin(\pi x)\sin(\pi y) + 2 - 3\cos^2(\pi y), & (x,y) \in \Omega, \\ u(x,y) = 0, & (x,y) \in \partial\Omega, \end{cases} \tag{3.19}$$

where $\Omega = [0,2] \times [0,2]$ and the exact solution is $u(x,y) = \sin(\pi x)\sin(\pi y)$. We compute the average error corresponding to $\psi_1 \equiv 1/4$ and then four point values corresponding to $\psi_2 \approx \delta_{(.5,.5)}$, $\psi_3 \approx \delta_{(.5,1.5)}$, $\psi_4 \approx \delta_{(1.5,1.5)}$, and $\psi_5 \approx \delta_{(1.5,.5)}$. We use

$$\hat{\delta}_{(c_x,c_y)} = \frac{400}{\pi} e^{-400((x-c_x)^2+(y-c_y)^2)}$$

to approximate the delta function $\delta_{(c_x,c_y)}$.

In the Global Computation, we compute a mesh that gives all of the desired information accurately using a tolerance of 2%. We begin with an 8×8 mesh. We list the results below:

		ψ_1			ψ_2			ψ_3		
Lev.	Elt's	Est.	Err.	Rat.	Est.	Err.	Rat.	Est.	Err.	Rat.
1	64	.035	.035	1.0	.090	.29	3.3	.24	.022	.091
2	201	.0088	.0089	1.0	.042	.082	1.9	.0024	.014	6.0
3	763	.0027	.0027	1.0	.020	.020	.99	.0020	.0020	1.0
4	2917	.00044	.00044	1.0	.0050	.00504	1.0	.0049	.00504	1.0

The error estimates for the point values are not very accurate on the coarser meshes, but become very accurate on mesh of moderate density and finer. It is simply an issue of locating a sufficient number of elements near the centers of the delta functions so that the approximation of the generalized Green's functions is accurate.

We obtain an acceptably accurate solution after four refinement levels using a mesh with 2917 elements. We plot both the initial and final meshes in Fig. 3.11.

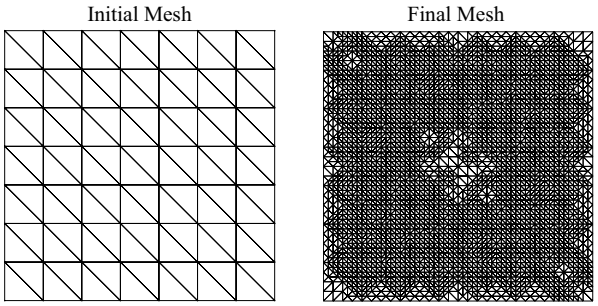

FIGURE 3.11. Initial and final meshes for Example 2 with for the solution computing all five data.

We next perform a Decomposed Computation by solving for approximate solutions $\{\hat{U}_1, \cdots, \hat{U}_5\}$ corresponding to each data $\{\psi_1, \cdots, \psi_5\}$ independently. Checking the Correlation Ratios reveals no significant correlations between the independent error indicators. There is no partition of unity involved in this decomposition and we simply use the same tolerance 2% for each independent computation. However, to obtain final independent solutions that yield roughly the same accuracy in the computed quantities as the solution of the Global Computation, we vary the initial meshes; using 7×7 for $\hat{U}_1$; 9×9 for $\hat{U}_2$ and $\hat{U}_4$; and 12×12 for $\hat{U}_3$ and $\hat{U}_5$. The final results for each computation are listed below:

Data	Level	Elements	Estimate
ψ_1	3	409	$-.0004699$
ψ_2	4	1037	$-.007870$
ψ_3	2	281	$-.005571$
ψ_4	4	1037	$-.007870$
ψ_5	2	281	$-.005571$

The Final Element Ratio is $2917/1037 \approx 2.8$. Since the solution corresponding to the average error is not the dominant cost in the independent computations, we do not bother to do a partition of unity decomposition on that problem. Finally, we plot some of the final meshes in Fig. 3.12.

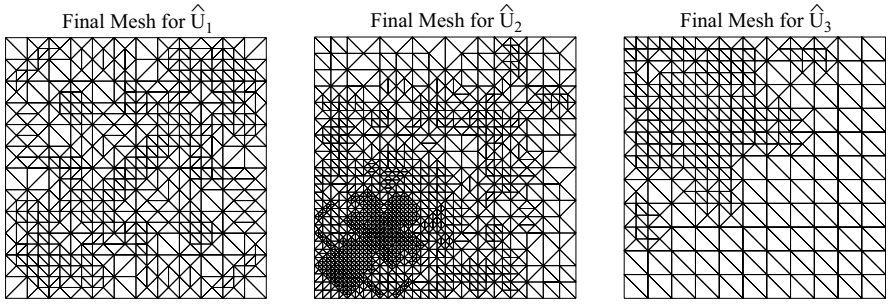

FIGURE 3.12. Final meshes for $\{\hat{U}_1, \hat{U}_2, \hat{U}_3\}$ in Example 2. The mesh for $\hat{U}_4$ is symmetric across $y = 2 - x$ with the mesh for $\hat{U}_2$ and the mesh for $\hat{U}_5$ is symmetric across $y = x$ with the mesh for $\hat{U}_3$.

3.5.3. Example 3. In this section, we investigate some properties of the correlation indicators using the problem,

$$\begin{cases} -\Delta u = 16(y - y^2 + x - x^2) & (x, y) \in \Omega, \\ u(x, y) = 0, & (x, y) \in \partial\Omega, \end{cases} \tag{3.20}$$

where $\Omega = [0, 1] \times [0, 1]$ and the exact solution is $u(x, y) = 8x(1 - x)y(1 - y)$.

In the two examples considered so far, there has been little or no significant correlation in the error indicators of different data, and computing the corresponding solutions independently leads to a substantial gain in terms of decreasing the maximum number of elements required to achieve a desired accuracy in specified quantities of interest. In the first computation in this example, we consider a problem in which two data are substantially correlated.

We estimate the error in the average value of u solving (3.20). Since the domain is relatively small and the solution and the generalized Green's function are both very smooth, the gain from decomposing the solution using a partition of unity is greatly reduced compared the previous examples. Beginning with a 4×4 mesh and using a tolerance of 1%, we obtain a sufficiently accurate solution using a Global Computation after five refinements. The final mesh uses 885 elements and produces an error of .0008699. If we partition the domain using four equal regions as pictured in Fig. 3.8, we find no substantial correlations between the error indicators $\{\mathcal{E}_1, \cdots, \mathcal{E}_4\}$. Computing the four solutions independently in the Decomposed Computation yields a Final Element Ratio of around 1.5.

If we partition the domain using sixteen equal regions as pictured on the right in Fig. 3.8, we find a number of substantial correlations. For example, we find that

Correlation Ratio 1 for $\mathcal{E}_1$ on $\mathcal{E}_2 = .98$, Correlation Ratio 2 for $\mathcal{E}_1$ on $\mathcal{E}_2 = .44$,
Correlation Ratio 1 for $\mathcal{E}_2$ on $\mathcal{E}_1 = .82$, Correlation Ratio 2 for $\mathcal{E}_2$ on $\mathcal{E}_1 = .44$.

Computing $\hat{U}_1$ corresponding to the localized data ψ_1 using a tolerance of 1%, we obtain a sufficiently accurate solution after 5 refinements, producing a mesh with 367 elements and yielding an error estimate of $-.000047$. Repeating the computation for $\hat{U}_2$ also requires five refinements, producing a mesh with 494 elements and yielding an accuracy of $-.000066$. On the other hand, combining these two computations by using data equal to the sum of the two partition functions for the regions Ω_1 and Ω_2, results in a problem that requires 5 refinements, producing a mesh with 496 elements and an accuracy of $-.000097$. Thus, we gain almost nothing by computing $\hat{U}_1$ and $\hat{U}_2$ independently from each other. We plot the final meshes in Fig. 3.13.

In the second computation in this example, we investigate the effect on the robustness of the Correlation Indicators from computing the Indicators on coarse discretizations. We consider the error in the average value and the point values at $(.25, .25)$ and $(.5, .5)$. We use a partition of unity decomposition for the error in the average to get data $\{\psi_1, \cdots, \psi_4\}$. We let $\psi_5 \approx \delta_{(.25,.25)}$ and $\psi_6 \approx \delta_{(.5,.5)}$. We compare the correlation indicators on initial meshes ranging from 16 to 144 or 400 uniformly sized elements by plotting the Correlation Ratios versus the number of elements. We show a sample of results in Fig. 3.14.

In general, we find that all Correlation Ratios converge to a limit as the number of elements increases (and we can actually prove this is so). What is more important however is the degree of variation on coarse meshes. Generally, the second Correlation Ratio varies relatively little as the mesh density increases for all data.

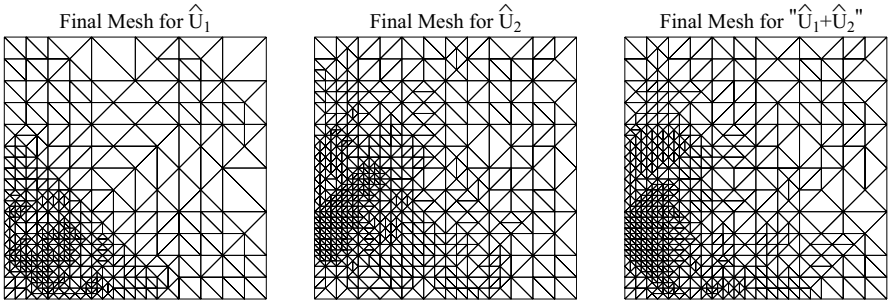

FIGURE 3.13. Final meshes for $\hat{U}_1$, $\hat{U}_2$, and the "combined" solution in Example 3.

The first Correlation Ratio between data representing a partition of unity decomposition also varies relatively little. However, it is not surprising to see that the first Correlation Ratio varies quite a bit on coarse meshes when one of the data is an approximate delta function. In terms of determining significant correlation, we find that the determination that two effective domains of influence are *not* closely correlated seems to be relatively robust with respect to the density of the mesh on which the indicators are computed. The determination that two effective domains of influence are correlated is less robust. Practically, this means that there is a mild tendency to combine computations that are more efficiently treated independently if the correlation indicators are computed on very coarse meshes.

3.5.4. Example 4. We turn to consider some problems for which we can not expect to obtain precise analytic information about the generalized Green's function. In this example, we consider a problem with diffusion that is nearly singular at one point and that has strong convection. We estimate the error in the average value of u solving

$$\begin{cases} -\nabla \cdot \Big(\big(.05 + \tanh\big(10(x-5)^2 + 10(y-1)^2\big)\big) \nabla u \Big) \\ \qquad\qquad + \begin{pmatrix} -100 \\ 0 \end{pmatrix} \cdot \nabla u = 1, & (x,y) \in \Omega, \\ u(x,y) = 0, & (x,y) \in \partial\Omega, \end{cases} \tag{3.21}$$

where $\Omega = [0,10] \times [0,2]$. We plot the diffusion in Fig. 3.15. Because of the sign of the convection, we expect that perturbations to the solution at a point with x-coordinate x_0 will affect the solution's values "downstream" for $x < x_0$ most strongly. The Peclet number for this problem is $Pe = 1000$.

We begin the computations with an initial mesh of 80 elements. For the Global Computation, we use an error tolerance of $TOL = .04\%$. We list some details of

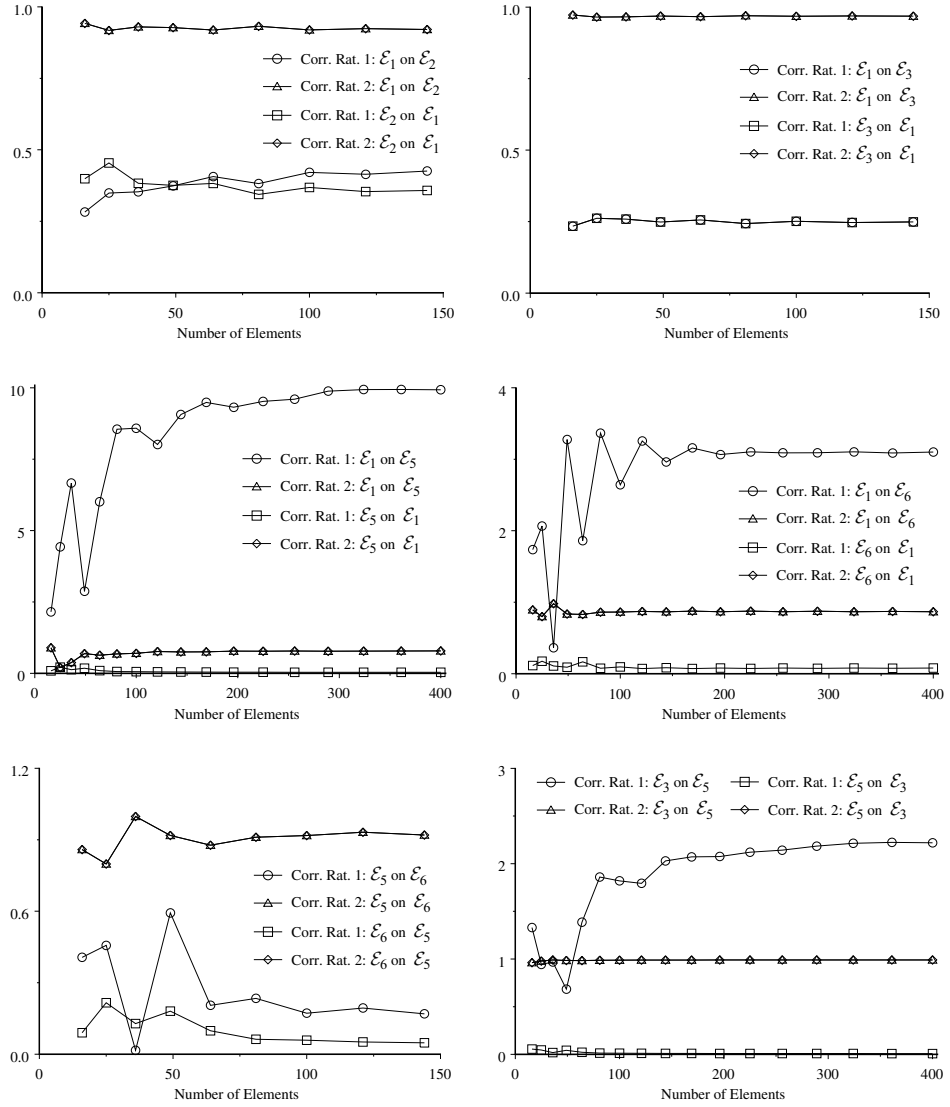

FIGURE 3.14. Plots of Correlation Ratios for a sample of computations in Example 3.

the computation below:

Level	Elements	Estimate
1	80	−.0005919
2	193	−.001595
3	394	−.0009039
4	828	−.0003820
5	1809	−.0001070
6	3849	−.00004073
7	9380	−.00001715
8	23989	−.000007553

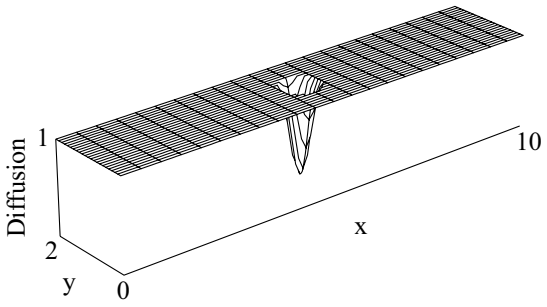

FIGURE 3.15. Plot of the diffusion coefficient for Example 4.

We plot the final mesh in Fig. 3.16. The effects of the convection are clear in the

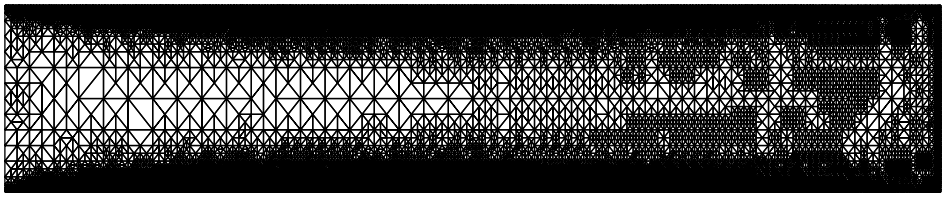

FIGURE 3.16. Plot of the final mesh for Example 4 with data ψ giving the average error.

pattern of mesh refinement. For the sake of comparison, we compute a numerical solution of the same problem except posing a velocity vector of $b = (-.01, 0)^\top$, corresponding to a Peclet number $Pe = .1$. We plot meshes from the original computation and the altered problem of approximately the same number of elements in Fig. 3.17. In the altered problem, the mesh refinement is much more heterogeneous.

Next, we consider the partition of unity with 20 subdomains shown in Fig. 3.18. Computing the Correlation Ratios, we find the significant correlations:

$\mathcal{E}_3$ with $\mathcal{E}_4$ $\quad$ $\mathcal{E}_6$ with $\mathcal{E}_7$ $\quad$ $\mathcal{E}_7$ with $\mathcal{E}_6$ $\quad$ $\mathcal{E}_9$ with $\mathcal{E}_8$ $\quad$ $\mathcal{E}_{10}$ with $\mathcal{E}_8$, $\mathcal{E}_9$
$\mathcal{E}_{13}$ with $\mathcal{E}_{14}$ $\quad$ $\mathcal{E}_{16}$ with $\mathcal{E}_{17}$ $\quad$ $\mathcal{E}_{17}$ with $\mathcal{E}_{16}$ $\quad$ $\mathcal{E}_{19}$ with $\mathcal{E}_{18}$ $\quad$ $\mathcal{E}_{20}$ with $\mathcal{E}_{18}$, $\mathcal{E}_{19}$

Note, there are no significant correlations in the cross-wind direction.

We compute the localized solutions $\{\hat{U}_i\}$ in the Decomposed Computation using two tolerances. The solutions are completely symmetric across $y = 1$. Details of the final computed solutions are listed below:

Data	TOL	Level	Elements	Estimate
ψ_1	.04%	7	7334	-6.927×10^{-7}
ψ_2	.04%	7	8409	-5.986×10^{-7}
ψ_3	.04%	7	7839	-5.189×10^{-7}
ψ_4	.04%	7	7177	-5.306×10^{-7}
ψ_5	.04%	7	7301	-4.008×10^{-7}
ψ_6	.02%	7	6613	-2.471×10^{-7}
ψ_7	.02%	7	4396	-2.938×10^{-7}
ψ_8	.02%	7	4248	-1.656×10^{-7}
ψ_9	.02%	7	3506	-1.221×10^{-7}
ψ_{10}	.02%	7	1963	-5.550×10^{-8}

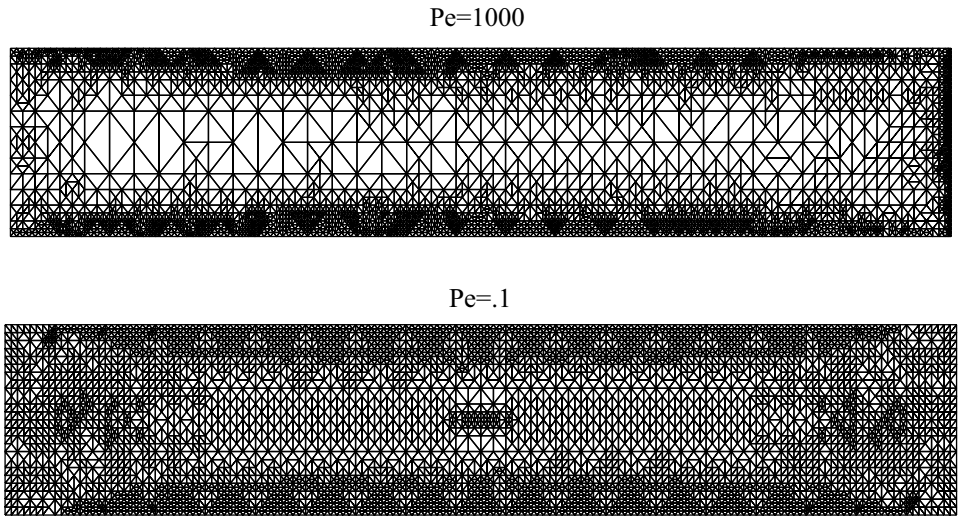

FIGURE 3.17. Plots of the mesh in the original problem with $Pe = 1000$ at refinement level 6 (number of elements = 3849 and the altered problem with $Pe = .1$ (number of elements = 4192) for Example 4 with data ψ giving the average error. We display the meshes from early refinement levels to make the qualitative features of the refinement clearer.

11	12	13	14	15	16	17	18	19	20
1	2	3	4	5	6	7	8	9	10

FIGURE 3.18. Domains for the partition of unity used in Example 4.

The estimate on the total average error of U_p is 7.24×10^{-6} and the Final Element Ratio is $23909/8409 \approx 2.9$.

We show a sample of the final meshes for the Decomposed Computation in Fig. 3.19. Note the effect of the convection is clearly visible in the pattern of mesh refinement. We can also see this in the graphs of the generalized Green's functions. We plot a sample in Fig. 3.20. Note the support of the two functions.

> *We emphasize that effective domains of influence may not be spatially compactly-shaped, as is generally the case for Poisson's equation.*

We can see this clearly in the upper plot in Fig. 3.19. The effective domain of influence for the average value of the solution in the lower left corner of the domain, close to the outflow boundary at $x = 0$, contains the immediate neighborhood of the boundary along $y = 0$, a swath that cuts up from the center of the outflow boundary through the center of the domain up to the upper boundary, and most of the inflow boundary.

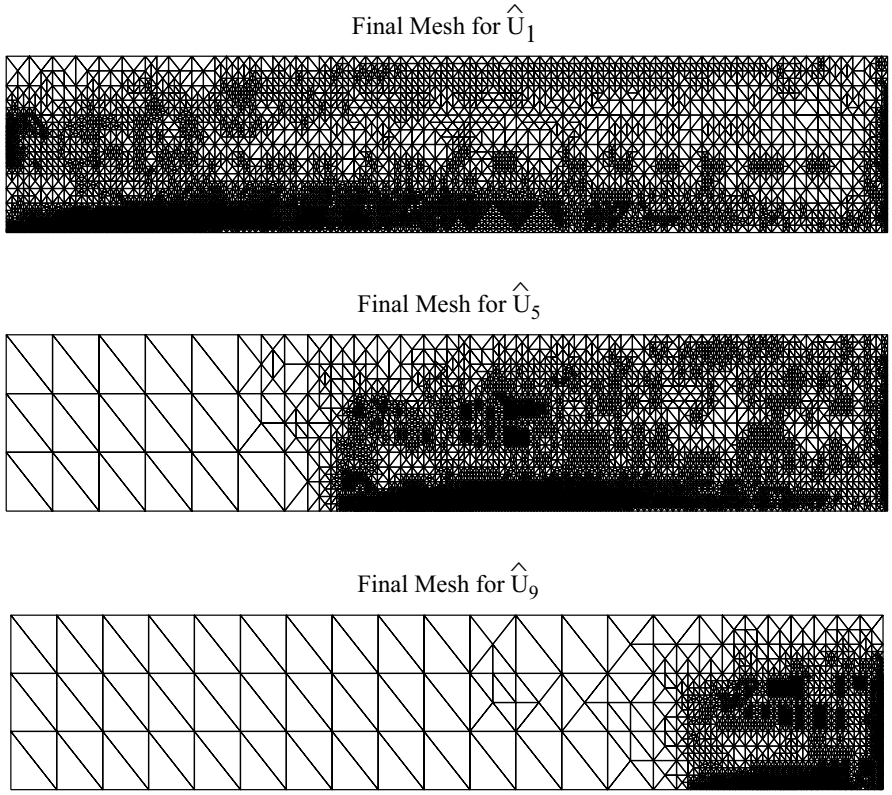

FIGURE 3.19. Plots of the final meshes for the localized solutions $\hat{U}_1$, $\hat{U}_5$, and $\hat{U}_9$ in Example 4.

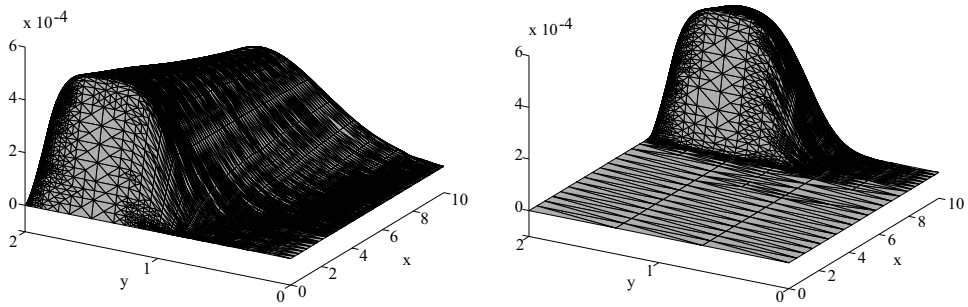

FIGURE 3.20. Plots of the generalized Green's functions corresponding to ψ_{11} (left) and ψ_{19} (right) in Example 4.

Keeping in mind the significant correlations listed above, we combine some of the localized computations by solving for localized solutions corresponding to summing the two of the partition of unity data. We list details of the final computed

solutions below:

Data	TOL	Level	Elements	Estimate
$\psi_3 + \psi_4$	.04%	7	8330	-9.8884×10^{-7}
$\psi_6 + \psi_7$	.02%	7	5951	-5.897×10^{-7}
$\psi_8 + \psi_9$	.02%	7	4406	-3.486×10^{-7}
$\psi_9 + \psi_{10}$	.02%	7	3202	-2.243×10^{-7}

The solutions for $\psi_3 + \psi_4$ and $\psi_8 + \psi_9$ use a few more elements than required for either of the original localized solutions. The solutions for $\psi_6 + \psi_7$ and $\psi_9 + \psi_{10}$ use less than the maximum required for the individual localized solutions.

3.5.5. Example 5. In the last example, we consider a problem posed on a more complicated domain. We estimate the error in the average value of u solving

$$\begin{cases} -\frac{1}{\pi^2}\Delta u = 2 + 4e^{-5((x-.5)^2+(y-2.5)^2)}, & (x,y) \in \Omega, \\ u(x,y) = 0, & (x,y) \in \partial\Omega, \end{cases} \tag{3.22}$$

where Ω is the "square annulus" $\Omega = [0,3] \times [0,3] \setminus [1,2] \times [1,2]$. The domain Ω is shown in Fig. 3.23. Note that we introduce some local variation in the forcing to make the solution more interesting.

We begin the computations with an initial mesh of 48 elements. For the Global Computation, we use an error tolerance of $TOL = 1\%$. We list some details of the computation below:

Level	Elements	Estimate
1	48	−5.168
2	125	−1.584
3	380	−.6879
4	894	−.3029
5	2075	−.1435

We plot the initial and final meshes in Fig. 3.21. Note the expected refinement

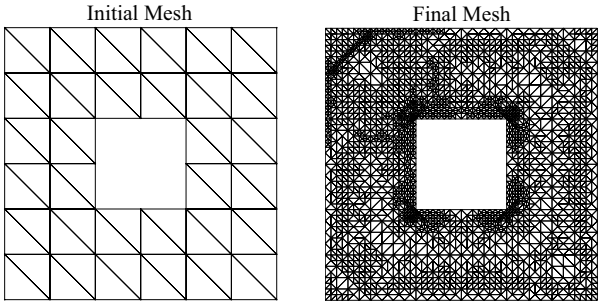

FIGURE 3.21. Plots of the initial (left) and final (right) meshes for Example 5 with data ψ giving the average error.

required near the interior corners. We plot the final solution and generalized Green's function in Fig. 3.22.

Next, we consider the partition of unity with 8 subdomains shown in Fig. 3.23. Checking the Correlation Ratios reveals no significant correlations. We obtain acceptable results in the Decomposed Computation using the same tolerance of 1% as used for the Global Computation. Details of the final computed solutions are

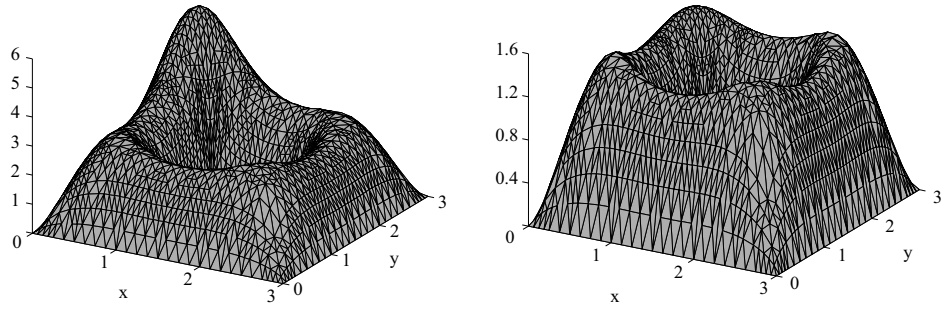

FIGURE 3.22. Plots of the final solution (left) and generalized Green's function (right) for Example 5 with data ψ giving the average error.

7	6	5
8		4
1	2	3

FIGURE 3.23. Domains for the partition of unity used in Example 5.

listed below:

Data	Level	Elements	Estimate	Data	Level	Elements	Estimate
ψ_1	5	1082	$-.01935$	ψ_5	5	1104	$-.01436$
ψ_2	5	1101	$-.01399$	ψ_6	5	1110	$-.01587$
ψ_3	5	1144	$-.01540$	ψ_7	5	1074	$-.02529$
ψ_4	5	1107	$-.01360$	ψ_8	5	1098	$-.01660$

Combining these solutions yields a partition of unity solution U_p with accuracy $-.1344$. Using the Decomposed Computation yields a Final Element Ratio of ≈ 1.8. We show a sample of the final meshes in Fig. 3.24. The most significant factor leading to a reduction in the number of elements required to achieve a desired accuracy is the fact that the localized computations do not refine near corners that are not in the immediate neighborhood of the support of the data.

We plot a couple of the final generalized Green's functions in Fig. 3.25.

We also tried a partition of unity on a finer decomposition of Ω obtained by dividing each sub-domain in the first partition into four equal squares. However, the Final Element Ratio is only 1.09.

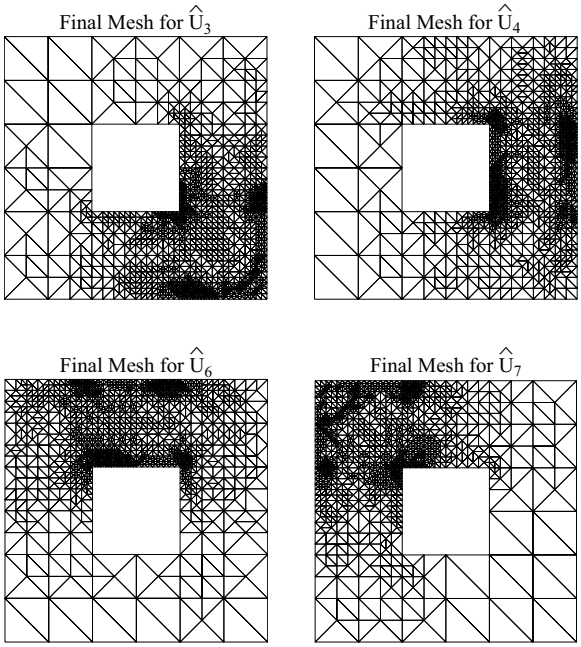

FIGURE 3.24. Plots of the final meshes for the localized solutions $\hat{U}_3$, $\hat{U}_4$, $\hat{U}_6$, and $\hat{U}_7$ in Example 5.

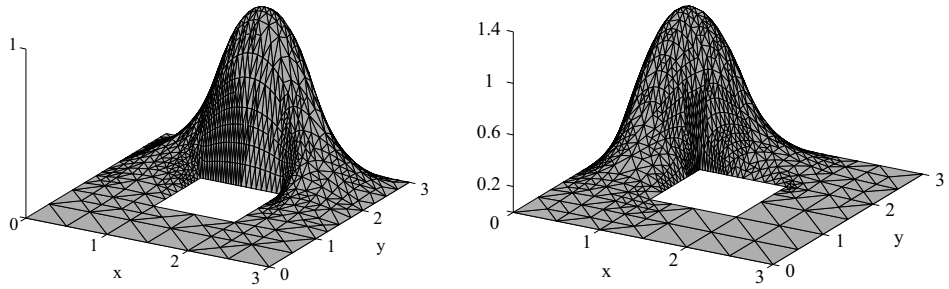

FIGURE 3.25. Plots of the generalized Green's functions corresponding to ψ_6 (left) and ψ_7 (right) for the partition of unity decomposition for Example 5.

CHAPTER 4

Nonlinear Problems

We conclude these notes by talking about the use of adjoint operators and the generalized Green's function for nonlinear problems. We have explained the connection between the solution of a linear problem and the adjoint problem. The connection for nonlinear problems is not as strong. For example, there are actually several valid adjoint problems for a given nonlinear problem in general.

4.1. An *a posteriori* analysis for a nonlinear algebraic equation

The problem is to estimate the error of a numerical solution X of a system of nonlinear algebraic equations,

$$f(x) = b \tag{4.1}$$

where the data b, nonlinearity f, and the solution x all have the same dimension. We assume that a numerical solution X of (4.1) has been computed in some fashion and we seek to estimate the unknown error $e = x - X$.

The residual error is now

$$R = f(X) - b$$

which immediately gives

$$f(x) - f(X) = -R. \tag{4.2}$$

The error, if indeed it can be obtained from the left-hand side, is related to the residual through a nonlinear equation.

To obtain a linear equation for the error, we use the mean value theorem for integrals in the form

$$f(x) - f(X) = \int_0^1 f'(sx + (1-s)X)\,(x - X)\,ds$$

where f' is the Jacobian matrix of f. Applying this to the last relation, we get

$$\tilde{A}e = -R$$

with

$$\tilde{A} = \int_0^1 f'(sx + (1-s)X)\,ds,$$

which is the linear problem obtained by linearizing f around an average of x and $-X$. We now follow the variational analysis in Ex. 1.63 to obtain

$$|(e, \psi)| = |(e, \tilde{A}^\top \phi)| = |(\tilde{A}e, \phi)| = |(R, \phi)|.$$

In this approach, we have used linearization via Taylor's theorem to produce a linear problem that in turn is used to define an adjoint problem. Unfortunately, the linearization requires knowledge of the computed numerical solution and the

true unknown solution in order to determine the correct generalized Green's vector. This is generally true.

In practice, for example, we can try to replace x by X in the definition of $\tilde{A}$,

$$\tilde{A} \to A = \int_0^1 f'(sX + (1-s)X)\, ds = f'(X),$$

to obtain a computable adjoint problem. Of course, this raises the issue of the effect of this substitution on the accuracy and reliability of the resulting *a posteriori* error estimate. This is an open research question.

4.2. Defining the adjoint to a nonlinear operator

We describe a general framework for defining the adjoint to an nonlinear operator. The main point is to explain that there are several valid ways to do this.

We assume that the Banach spaces X and Y are actually Sobolev spaces, as defined in Sec. 1.3, and use the notation $(\ ,\ )$ for the L^2 inner product, and so forth. Without explaining all of the details, we need this kind of structure in order to justify everything mathematically. In particular, we employ "smoothness" of the nonlinear operator and we have to make sense of that, which depends on the spaces.

We actually define the adjoint for a specific kind of nonlinear operator. The motivation is the nonlinear equation (4.2) relating the error of a numerical solution to its residual. In general, assume that f is a nonlinear map from X into Y, where we assume that the domain $\mathcal{D}(f)$ is a convex set.

Definition 4.1. A subset A of a vector space is **convex** if for any $a, b \in A$, the set of points on the "line segment" joining a and b, i.e., $\{sa + (1-s)b |\, 0 \le s \le 1\}$ is contained in A.

This is a standard requirement when dealing with nonlinear operators, as it allows the use of some form of the Mean Value Theorem among other things. Note that we do not assume that $\mathcal{D}(f)$ is a vector subspace. We choose u and U inside $\mathcal{D}(f)$ and define the new nonlinear operator

$$F(e) = f(u+e) - f(u), \tag{4.3}$$

where $e = U - u$. The domain of F is

$$\mathcal{D}(F) = \{v \in X |\, v + u \in \mathcal{D}(f)\}.$$

We assume that $\mathcal{D}(F)$ is independent of e and dense in X. Note that $0 \in \mathcal{D}(F)$ and $F(0) = 0$. We also assume that $\mathcal{D}(F)$ is a vector subspace of X. It is clear that there is a lot of mathematical work to do when verifying these assumptions!

We define an adjoint to an operator F of the form (4.3). There are two reasons to do this.

- As we saw in Sec. 4.1, this is the kind of nonlinearity that arises when estimating the error of a numerical solution of a nonlinear problem. In general, studying the effects of perturbations in a nonlinear problem, e.g., model uncertainty, yields the same kind of nonlinearity.
- Nonlinear problems typically do *not* enjoy the global solvability that characterizes linear problems. Instead, there is only a local solvability in the sense that we can expect there to be solutions only nearby a fixed given solution.

We base the *first* definition of the adjoint on the bilinear identity.

DEFINITION 4.2. An operator $A^*(e)$ with domain $\mathcal{D}(A^*) \subset Y^*$ and range in X^* is an **adjoint operator** corresponding to F if

$$(F(v), w) = (v, A^*(v)w) \qquad \text{for all } v \in \mathcal{D}(F),\ w \in \mathcal{D}(A^*).$$

Note that we say is an adjoint operator associated with F, not the adjoint operator to F. Several operators may satisfy this definition.

EXAMPLE 4.3. Suppose that F can be represented as $F(e) = A(e)e$, where $A(e)$ is a linear operator with $\mathcal{D}(F) \subset \mathcal{D}(A)$. For a fixed $e \in \mathcal{D}(F)$, we can define the adjoint of A satisfying $(A(e)w, v) = (w, A^*(e)v)$ for all $w \in \mathcal{D}(A)$, $v \in \mathcal{D}(A^*)$ as usual. Substituting $w = e$ shows this defines an adjoint of F as well. If there are several such linear operators A, then there will be several different possible adjoints.

EXAMPLE 4.4. Consider $(t, x) \in \Omega = (0, 1) \times (0, 1)$, with $X = X^* = Y = Y^* = L^2$ be the space of periodic functions in t and x, with period equal to 1. Consider a periodic problem of the form

$$F(e) = \frac{\partial e}{\partial t} + e\frac{\partial e}{\partial x} + ae = f$$

where $a > 0$ is a constant and the domain of F is the set of continuously differentiable functions. We can write $F(e) = A_i(e)e$ where

$$A_1(e)v = \frac{\partial v}{\partial t} + e\frac{\partial v}{\partial x} + av$$

$$A_2(e)v = \frac{\partial v}{\partial t} + \left(a + \frac{\partial e}{\partial x}\right) v$$

$$A_3(e)v = \frac{\partial v}{\partial t} + \frac{1}{2}\frac{\partial (ev)}{\partial x} + av.$$

Using the usual integration by parts argument, we can verify construct the adjoints

$$A_1^*(e)w = -\frac{\partial w}{\partial t} - \frac{\partial (ew)}{\partial x} + aw$$

$$A_2^*(e)w = -\frac{\partial w}{\partial t} + \left(a + \frac{\partial e}{\partial x}\right) w$$

$$A_3^*(e)w = -\frac{\partial w}{\partial t} - \frac{e}{2}\frac{\partial w}{\partial x} + aw.$$

We base the *second* definition of an adjoint on the integral mean value theorem, as in Sec. 4.1. We assume that the original nonlinearity is Frechet differentiable (in the finite dimensional case, this means that the Jacobian is defined and is continuous). The integral mean value theorem states

$$f(U) = f(u) + \int_0^1 f'(u + se)\, ds\ e$$

where $e = U - u$ and f' is the Frechet derivative of f.

We rewrite this as

$$F(e) = f(U) - f(u) = A(e)e$$

with

$$A(e) = \int_0^1 f'(u + se)\, ds.$$

Note that we can apply the integral mean value theorem to F and obtain the formula

$$A(e) = \int_0^1 F'(se)\,ds.$$

Since we have introduced differentiation, we necessarily have to derive some results about the smoothness of F. It is not difficult, but it does require the calculus for operators, so we skip that.

DEFINITION 4.5. For a fixed e, the adjoint operator $A^*(e)$, defined in the usual way for the linear operator $A(e)$, is said to be an adjoint for F.

This is the same adjoint used in Sec.4.1.

EXAMPLE 4.6. Consider Ex. 4.4. We find that

$$F'(e)v = \frac{\partial v}{\partial t} + e\frac{\partial v}{\partial x} + \left(a + \frac{\partial e}{\partial x}\right) v.$$

After some technical analysis of the domains of the operators involved, we find that

$$A^*(e)w = -\frac{\partial w}{\partial t} - \frac{e}{2}\frac{\partial w}{\partial x} + aw.$$

This coincides with the third adjoint computed above.

4.3. *A posteriori* error analysis for a space-time finite element method

To illustrate how these ideas are used in practice, we consider a concrete example. We study a system of D reaction-diffusion equations consisting of d, $1 \le d \le D$, parabolic equations and $D - d$ ordinary equations for the $\mathbb{R}^D$ valued function $u = (u_i)$:

$$\begin{cases} \dot{u}_i - \nabla \cdot (\epsilon_i(u, x, t)\nabla u_i) = f_i(u, x, t), & (x, t) \in \Omega \times \mathbb{R}^+,\ 1 \le i \le D, \\ u_i(x, t) = 0, & (x, t) \in \partial\Omega \times \mathbb{R}^+,\ 1 \le i \le d, \\ u(x, 0) = u_0(x), & x \in \Omega, \end{cases} \tag{4.4}$$

where Ω is an interval in $\mathbb{R}^1$ and a convex polygonal domain in $\mathbb{R}^2$ with boundary $\partial\Omega$, $\dot{u}_i$ denotes the partial derivative of u_i with respect to time, and there is a constant $\epsilon > 0$ such that

$$\epsilon_i(u, x, t) \ge \epsilon \text{ for } 1 \le i \le d \text{ and } \epsilon_i(u, x, t) \equiv 0 \text{ for the rest.}$$

We also assume that $\epsilon = (\epsilon_i)$ and $f = (f_i)$ have smooth second derivatives and for simplicity, we write $\epsilon_i(u, x, t) = \epsilon_i(u)$ and $f(u, x, t) = f(u)$. We use u^p and u^o to denote the parts of u associated to the parabolic and ordinary differential equations respectively. In other words, $u_i^p = u_i$ for $1 \le i \le d$ and $u_i^p = 0$ for $d < i \le D$ and $u^o = u - u^p$.

The presence of ordinary differential equations in the system (4.4) has strong consequences for the smoothness of solutions. In particular, we can expect parabolic smoothing to occur only for u^p, while the smoothness of u^o is generally determined by the smoothness of u^p and the initial data since f is smooth.

We describe two finite element space-time discretizations of (4.4) called the continuous and discontinuous Galerkin method. We partition $[0, \infty)$ as $0 = t_0 < t_1 < t_2 < \cdots < t_n < \ldots$, denoting each time interval by $I_n = (t_{n-1}, t_n]$ and time step by $k_n = t_n - t_{n-1}$. To each interval I_n, we associate a triangulation $\mathcal{T}_n$ of Ω arranged so the union of the elements in $\mathcal{T}_n$ is Ω while the intersection of any two

elements is either a common edge, node, or is empty. We assume that the smallest angle of any triangle in a triangulation is bounded below by a fixed constant, or equivalently that there is a constant λ_0 independent of the triangulation $\mathcal{T}_n$ such that $\text{area}(K) \geq \lambda_0 \,\text{diam}(K)^2$, where $\text{diam}(K)$ is the length of the largest side of K, for any triangle $K \in \mathcal{T}_n$.

Note that mesh changes can occur across time nodes. To measure the size of the elements of $\mathcal{T}_n$, we use a piecewise constant function h_n, the so-called **mesh function**, defined so $h_n|_K = \text{diam}(K)$ for $K \in \mathcal{T}_n$, We also use $h_{n,\min} = \min h_n(\cdot)$ and $h_{n,\max} = \max h_n(\cdot)$ and denote the global mesh function by h, where $h|_{I_n} = h_n$. Similarly, we use k to denote the piecewise constant function that is k_n on I_n. When the time level is clear in the context, we abuse notation by dropping the subscript n.

The approximations are polynomials in time and piecewise polynomials in space on each space-time "slab" $S_n = \Omega \times I_n$. In space, we let $V_n \subset (H_0^1(\Omega))^d \times (H^1(\Omega))^{D-d}$ denote the space of piecewise linear continuous vector-valued functions $v(x) \in \mathbb{R}^D$ defined on $\mathcal{T}_n$, where the first d components of v are zero on $\partial\Omega$. Then on each slab, we define

$$W_n^q = \big\{ w(x,t) : \, w(x,t) = \sum_{j=0}^{q} t^j v_j(x), \, v_j \in V_n, \, (x,t) \in S_n \big\}.$$

Finally, we let W^q denote the space of functions defined on the space-time domain $\Omega \times \mathbb{R}^+$ such that $v|_{S_n} \in W_n^q$ for $n \geq 1$. Note that functions in W^q are generally discontinuous across the discrete time levels and we denote the jump across t_n by $[w]_n = w_n^+ - w_n^-$ where $w_n^\pm = \lim_{s \to t_n \pm} w(s)$. To define the methods, we use the L^2 projection operator P_n onto V_n, i.e. $P_n : L^2(\Omega) \to V_n$ is defined by $(P_n v, w) = (v, w)$ for all $w \in V_n$, where $(\cdot, \cdot)$ denotes the $L_2(\Omega)$ inner product. We use $\| \quad \|$ for the L_2 norm. The global projection operator P is defined by setting $P = P_n$ on S_n. We also use the L^2 projection operator into the piecewise polynomial functions in time, denoted by $\pi_n : L^2(I_n) \to \mathcal{P}^q(I_n)$, where $\mathcal{P}^q(I_n)$ is the space of polynomials of degree q or less defined on I_n. The global projection operator π is defined by setting $\pi = \pi_n$ on S_n.

DEFINITION 4.7. The **continuous Galerkin** cG(q) approximation $U \in W^q$ satisfies $U_0^- = P_0 u_0$ and for $n \geq 1$, the **Galerkin orthogonality relation**

(4.5)
$$\begin{cases} \displaystyle\int_{t_{n-1}}^{t_n} \big((\dot{U}_i, v_i) + (\epsilon_i(U) \nabla U_i, \nabla v_i) \big) \, dt = \int_{t_{n-1}}^{t_n} (f_i(U), v_i) \, dt \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{for all } v \in W_n^{q-1}, \, 1 \leq i \leq D, \\ U_{n-1}^+ = P_n U_{n-1}^-. \end{cases}$$

Note that U is continuous across time nodes over which there is no mesh change. In particular, it is usually the case that $U_0^- = U_0^+$.

DEFINITION 4.8. The **discontinuous Galerkin** dG(q) approximation $U \in W^q$ satisfies $U_0^- = P_0 u_0$ and for $n \geq 1$,

$$\int_{t_{n-1}}^{t_n} \big((\dot{U}_i, v_i) + (\epsilon_i(U) \nabla U_i, \nabla v_i) \big) \, dt + \big([U_i]_{n-1}, v_i^+ \big) = \int_{t_{n-1}}^{t_n} (f_i(U), v_i) \, dt \quad \text{for all } v \in W_n^q, \, 1 \leq i \leq D. \tag{4.6}$$

Note that the true solution satisfies both (4.5) and (4.6).

EXAMPLE 4.9. To illustrate, we discretize the scalar problem

$$\begin{cases} \dot{u} - \Delta u = f(u), & (x,t) \in \Omega \times \mathbb{R}^+, \\ u(x,t) = 0, & (x,t) \in \partial\Omega \times \mathbb{R}^+, \\ u(x,0) = u_0(x), & x \in \Omega, \end{cases} \tag{4.7}$$

using the dG(0) method. Since U is constant in time on each time interval, we let $\vec{U}_n$ denote the M_n vector of nodal values with respect to the nodal basis $\{\eta_{n,i}\}_{i=1}^{M_n}$ for V_n on I_n. We let $B_n : \big(B_n\big)_{ij} = \big(\eta_{n,i}, \eta_{n,j}\big)$ for $1 \le i, j \le M_n$ and $B_{n,n-1} : \big(B_{n,n-1}\big)_{ij} = \big(\eta_{n,i}, \eta_{n-1,j}\big)$ for $1 \le i \le M_n$, $1 \le j \le M_{n-1}$ denote the *mass matrices* and $A_n : \big(A_n\big)_{ij} = \big(\nabla\eta_{n,i}, \nabla\eta_{n,j}\big)$ denote the *stiffness matrix*. Then U_n satisfies

$$\big(B_n + k_n A_n\big)\vec{U}_n - \vec{F}(U_n^-)k_n = B_{n,n-1}\vec{U}_{n-1}, \quad n \ge 1,$$

where $(\vec{F}(U_n^-))_i = (f(U_n^-), \eta_{n,i})$.

With appropriate use of quadrature to evaluate the integrals in the variational formulation, these Galerkin methods yield standard difference schemes.

EXAMPLE 4.10. In the example above, if M_n is constant and the lumped mass quadrature is used to evaluate the coefficients of B_n and $B_{n,n-1} = B_n$, then the resulting set of equations for the dG(0) approximation is the same as the equations for the nodal values of the backward Euler difference scheme for (4.7).

The dG(0) method is related to the backward Euler method, the cG(1) method is related to the Crank-Nicolson scheme, and the dG(1) method is related to the third order sub-diagonal Padé difference scheme.

Under general assumptions, the cG(q) and dG(q) have order of accuracy $q+1$ in time and 2 in space at any point. In addition, they enjoy a superconvergence property in time at time nodes. The dG(q) method will have order of accuracy $2q+1$ in time and the cG(q) method will have order $2q$ in time at time nodes for sufficiently smooth solutions. In terms of stability characteristics, the discontinuous Galerkin method has stability properties that make it well suited for the solution of dissipative problems. In particular, it is often possible to show that the error of the dG approximation of a dissipative problem is either bounded or grows only very slowly with time. Similarly, the continuous Galerkin method is "energy" preserving which has the consequence that sometimes the error of the cG approximation accumulates at a slower rate than the error of nonconserving schemes in problems with a conserved quantity.

We define the coefficients for the adjoint problem by linearizing around an average of the true and approximate solutions as in the second definition of the adjoint.

$$\begin{aligned} \bar{\epsilon}_i = \bar{\epsilon}_i(u,U) &= \int_0^1 \epsilon_i\big(us + U(1-s)\big)\,ds, \\ \bar{\beta}_{ij} = \bar{\beta}_{ij}(u,U) &= \int_0^1 \frac{\partial \epsilon_j}{\partial u_i}\big(us + U(1-s)\big)\nabla\big(u_i s + U_i(1-s)\big)\,ds, \\ \bar{f}_{ij} = \bar{f}_{ij}(u,U) &= \int_0^1 \frac{\partial f_j}{\partial u_i}\big(us + U(1-s)\big)\,ds. \end{aligned} \tag{4.8}$$

The regularity of u and U typically imply that $\bar{\epsilon}$ and $\bar{f}$ are piecewise continuous with respect to t and continuous, H^1 functions in space while $\bar{\beta}$ is discontinuous in time and space.

Written out pointwise for convenience, the adjoint problem to (4.4) for the generalized Green's function associated to the data ψ, which determines the quantity of interest, is

(4.9)
$$\begin{cases} -\dot{\phi}_i - \nabla \cdot \left(\bar{\epsilon}_i \nabla \phi_i\right) + \sum_{j=1}^{D} \bar{\beta}_{ji} \cdot \nabla \phi_j - \sum_{j=1}^{D} \bar{f}_{ij} \phi_j = \psi_i, & (x,t) \in \Omega \times (t_n, 0], \\ & \quad 1 \le i \le D, \\ \phi_i\,(x,t) = 0, & (x,t) \in \partial\Omega \times (t_n, 0], \\ & \quad 1 \le i \le d, \\ \phi(x,t_n) = 0, & x \in \Omega, \end{cases}$$

Example 4.11. In the case of the scalar problem with constant diffusion, the adjoint problem is

$$\begin{cases} -\dot{\phi} - \epsilon \Delta \phi - \bar{f}\phi = \psi, & (x,t) \in \Omega \times (t_n, 0], \\ \phi(x,t) = 0, & (x,t) \in \partial\Omega \times (t_n, 0], \\ \phi(x,t_n) = 0, & x \in \Omega. \end{cases}$$

Example 4.12. In the case of one parabolic equation with nonlinear diffusion coupled to one ordinary differential equation, the dual problem is

$$\begin{cases} -\dot{\phi}_1 - \nabla \cdot \bar{\epsilon}_1 \nabla \phi_1 + \bar{\beta}_{11} \nabla \phi_1 - \bar{f}_{11}\phi_1 - \bar{f}_{12}\phi_2 = \psi_1, & (x,t) \in \Omega \times (t_n, 0], \\ -\dot{\phi}_2 + \bar{\beta}_{12} \nabla \phi_1 - \bar{f}_{21}\phi_1 - \bar{f}_{22}\phi_2 = \psi_2, & (x,t) \in \Omega \times (t_n, 0], \\ \phi_1(x,t) = 0, & (x,t) \in \partial\Omega \times (t_n, 0], \\ \phi(x,t_n) = 0, & x \in \Omega. \end{cases}$$

This choice for the adjoint yields the following error representation formulas. For the cG method, we have

$$\int_0^{t_n} (e,\psi)\,dt = (e^+(0),\phi(0)) + \int_0^{t_n} \left((\dot{U}, \pi P\phi - \phi) + (\epsilon(U)\nabla U, \nabla(\pi P\phi - \phi)) - (f(U), \pi P\phi - \phi) \right) dt.$$

For the dG method, we get

$$\int_0^{t_n} (e,\psi)\,dt = (e^-(0),\phi(0)) + \sum_{j=1}^{n} \left([U]_{j-1}, (\pi P\phi - \phi)^+_{j-1} \right) + \int_0^{t_n} \left((\dot{U}, \pi P\phi - \phi) + (\epsilon(U)\nabla U, \nabla(\pi P\phi - \phi)) - (f(U), \pi P\phi - \phi) \right) dt.$$

Important mathematical questions regarding these representation formulas include

- Does the finite element approximation have sufficient smoothness in order for these formulas to make sense?
- Is the adjoint problem well-posed, and what are the smoothness properties of the generalized Green's function?
- What is the effect of linearization error on the accuracy of the formulas?

These issues are complicated in nonlinear problems because the adjoint problem depends on the approximation, e.g., if the approximation is badly behaved, then the generalized Green's function might be badly behaved, and these formulas may not mean very much. The way this additional complication is addressed is by restricting the problems so that the true solution and the approximation both enjoy special stability properties. See [**ELW00**] for a complete theory for nonlinear reaction-diffusion problems.

4.4. The bistable problem

We conclude the notes by investigating the behavior of the generalized Green's function corresponding to a well-known reaction diffusion problem called variously the bistable, Chafee-Infante, and Allen-Cahn problem. This has the form (4.4) with $D = 1$, $\epsilon > 0$ constant, $\beta \equiv 0$, and $f(u) = u - u^3$.

$$\begin{cases} \dfrac{\partial u}{\partial t} - \epsilon \dfrac{\partial^2 u}{\partial x^2} = u - u^3, & 0 < x < 1,\ 0 < t, \\ \dfrac{\partial u}{\partial x}(0,t) = \dfrac{\partial u}{\partial x}(1,t) = 0, & 0 < t, \\ u(x,0) = u_0(x), & 0 < x < 1. \end{cases} \tag{4.10}$$

In one dimension, the bistable equation has been used to model the motion of domain walls in ferromagnetic materials. It is also used as a prototypical example of "metastability" in one dimension and "motion by mean curvature" in two dimensions. It is one of the simplest problems that produce evolution to equilibrium in the presence of competing stable steady states. When ϵ is sufficiently small, the only stable equilibrium solutions are $u \equiv 1$ and $u \equiv -1$ and all solutions, except unstable equilibrium solutions, converge to one of these two steady-states. However, this convergence may be extremely slow because solutions can exhibit dynamic metastability. Generic initial data forms a pattern of transition layers between the values -1 and 1 during an initial transient, after which the layers coalesce by moving more or less in a horizontal direction. The time scale for substantial motion of the layers is $\exp(Cd/\sqrt{\epsilon})$ where C is a constant and d is the distance between neighboring layers. When two layers become sufficiently close, a rapid transient occurs during which the layers collapse together. The solution then forms a new, simpler metastable pattern and the process begins anew.

We illustrate with a computation made with $\epsilon = .0009$ and

$$u_0(x) = \begin{cases} \tanh((.2 - x)/(2\sqrt{\epsilon})), & 0 \le x < .28, \\ \tanh((x - .36)/(2\sqrt{\epsilon})), & .28 \le x < .4865, \\ \tanh((.613 - x)/(2\sqrt{\epsilon})), & .4865 \le x < .7065, \\ \tanh((x - .8)/(2\sqrt{\epsilon})), & .7065 \le x \le 1, \end{cases}$$

which produces a function that is very close to a metastable state. We display the evolution of the corresponding numerical solution in Fig. 4.1. The "well" on the left is slightly thinner and collapses first. Care is needed when computing. For example, computing without a sufficiently fine time step or space mesh causes "locking" in which a metastable pattern actually becomes artificially stable.

In this example, we investigate the behavior of the generalized Green's function corresponding to determining point values of the solution of the bistable problem at many points by reporting on the values of the associated *stability factors*. Recall

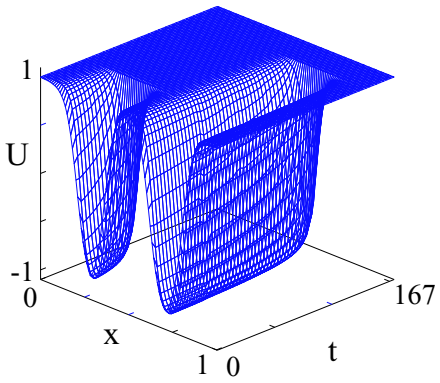

FIGURE 4.1. Evolution of a metastable solution starting with two "wells" and $\epsilon = .0009$. The left well is thinner and collapses at $t \approx 41$ and the second well collapses at $t \approx 141$.

from Sec. 2.3, that we define the stability factors by deriving bounds on the *a posteriori* error representation formulas. The stability factors are a form of condition number for the particular solution of the differential equation being studied. They depend on the particular solution in a nonlinear problem because of the linearization used to form the adjoint problem.

To be precise, we would have to define appropriate residuals for a finite element solution of (4.4), and carry out the analysis to obtain a bound. We forgo this and just state the form of the stability factors appropriate to (4.4).

If the generalized Green's function is sufficiently smooth, more precisely,

$$\phi \in L_\infty((0,t_n);L_2(\Omega)),\ D_t^\alpha \phi \in L_1((0,t_n);L_2(\Omega)),$$
$$\text{and } D^2\phi^p \in L_1((0,t_n);L_2(\Omega)),$$

where $0 \le \alpha \le 1$ for the cG(1) and dG(0) methods and $0 \le \alpha \le 2$ for the dG(1) method, then we can take optimal interpolation estimates on the adjoint weights $\pi P\phi - \phi$ that appear in the error representation formulas.

The stability factor associated to the propagation of the initial error is defined:

$$S^i(0,t_n) = \|\phi(0)\|.$$

The stability factor associated with time discretization by means of the cG(q) or dG(q-1) method is defined by

$$S_\alpha^t(0,t_n) = C_\alpha^t \int_0^{t_n} \|D_t^\alpha \phi\|\,dt, \quad 0 \le \alpha \le q,$$

where C_α^t is the interpolation constant in the L_1 error bound for the L_2 projection into the space of scalar polynomials of degree α. In order to define the stability factors associated to space discretization, we denote the part of ϕ associated to the parabolic and ordinary differential equations by ϕ^p and ϕ^o respectively. Then,

$$S^p(0,t_n) = C^p \int_0^{t_n} \|D^2\phi^p\|\,dt \text{ and } S^o(0,t_n) = \int_0^{t_n} \|\phi^o\|\,dt,$$

where C^p is the standard interpolation constant for the L_2 error bound for the L_2 projection into the space of continuous piecewise linear functions V_n.

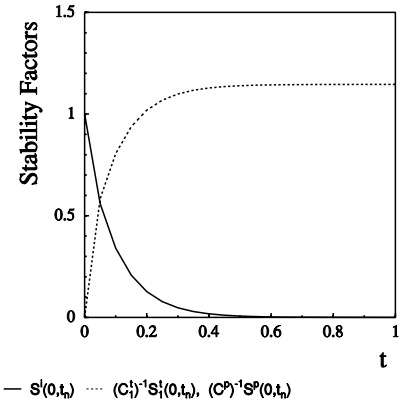

FIGURE 4.2. Plot of $S^i(0,t_n)$ and $(C_1^t)^{-1}S_1^t(0,t_n) = (C^p)^{-1}S^p(0,t_n)$ versus t_n for the heat equation.

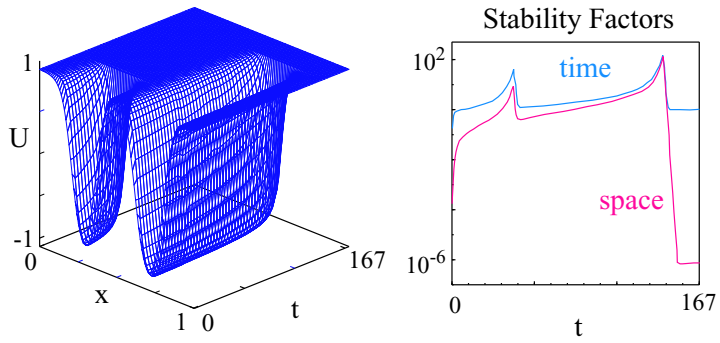

FIGURE 4.3. We plot the approximate stability factors versus time for the trajectory of the bistable problem computed above.

To illustrate, we plot the stability factors for the heat equation on the interval $[0,1]$ with Dirichlet boundary conditions for a generic choice of adjoint data in Fig. 4.2. We see that $S^i(0,t_n)$ decays exponentially to zero as $t_n \to \infty$, as expected for the heat equation. The other stability factors tend exponentially to a constant value ≈ 1.146, indicating that there is essentially no accumulation of discretization errors after sufficient time has passed.

In Fig. 4.3, we plot the approximate stability factors for the numerical solution of the bistable problem plotted in Fig. 4.1. After an initial transient, the stability factors approach a value close to 1. Thereafter, they grow super-exponentially during the metastable periods, yet overall remain moderately sized because they decrease extremely rapidly to a value close to one during the transient between metastable periods. This indicates that the trajectory becomes quite stable during these transients. This behavior appears to be characteristics of metastable solutions. We conclude that it is possible to compute accurate numerical solutions over long time intervals.

We also illustrate the "linearization effect" that arises because we solve an approximate adjoint problem obtained by linearization around the computed solution.

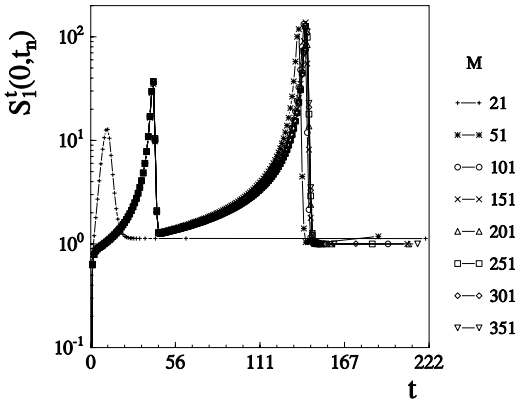

FIGURE 4.4. A plot of approximate stability factors $S_1^t(t)$ for the bistable problem computed using trajectories computed with varying accuracy in space.

We compute approximations using uniform space meshes with elements M ranging from $M = 21$ to $M = 351$. We maintain the contribution to error due to time integration to be less than .0001. For $M \leq 50$, the numerical solutions are subject to "locking", which means that one or more metastable layers become artificially stable, while the correct behavior is observed for $M \geq 51$. When $M = 21$, the thinner of the two wells collapses (though at a different time than for larger M) while the wider well becomes fixed. We plot the approximate stability factors $S_1^t(t)$ versus time for a sample of computations in Fig. 4.4. The locking phenomena is clearly reflected in the values of the stability factor for $M = 21$, which remains 1 after the first well collapses indicating that the resulting pattern is stable.

Even though the numerical solutions corresponding to $M = 32$ and $M = 64$ are nearly identical to the eye, the behavior of the two is radically different. In Fig. 4.5, we plot numerical solutions for equally spaced meshes with $M = 32$ and $M = 64$ at $t \approx 5.6$ and again at $t \approx 389$. The two solutions are very close at early times but because the solution on the coarser mesh becomes locked, the numerical solutions end up quite different at later times. The bistable problem is sensitive to linearization in the neighborhood of these two approximate trajectories. Fortunately, the *a posteriori* error bound estimates the error to be 2.23, i.e. more than %200, in the numerical solution with $M = 32$ elements at the time when the first well collapses. Note that Fig. 4.4 shows that the problem is not sensitive to linearization around numerical trajectories that are sufficiently accurate. $M = 101$, $M = 201$ and $M = 351$ all produce nearly the same behavior and stability factors.

This example is an illustration of the general observation about the nature of nonlinearity in the context of computing error estimates and determining model sensitivity. In this setting, a highly nonlinear problem is one in which nearby solutions have wildly different stability properties with respect to data and/or parameters. It is in this case that linearization can lead to inaccurate results. Unfortunately, it seems to be difficult to deal with this kind of nonlinear behavior analytically.

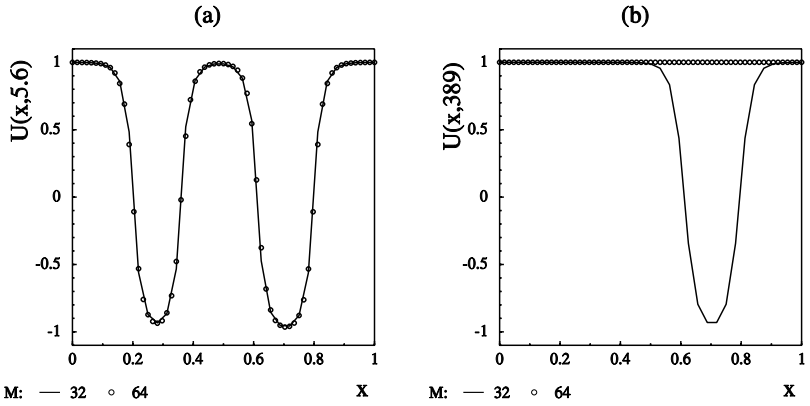

FIGURE 4.5. A plot of numerical solutions computed using equally spaced meshes with $M = 32$ and $M = 64$ at (a) $t \approx 5.6$ and (b) $t \approx 389$. .

Bibliography

[BH00] R. Bank and M. Holst, *A new paradigm for parallel adaptive mesh refinement*, SIAM J. Sci. Comput. **22** (2000), 1411–1443.

[BM97] I. Babuška and J. Melenk, *The partition of unity finite element method*, Internat. J. Numer. Methods Engrg. **40** (1997), 727–758.

[BR01] R. Becker and R. Rannacher, *An optimal control approach to a posteriori error estimation in finite element methods*, Acta Numerica (2001), 1–102.

[BR03] W. Bangerth and R. Rannacher, *Adaptive finite element methods for differential equations*, Birkhauser, Boston, 2003.

[BS94] S. Brenner and L. R. Scott, *The mathematical theory of finite element methods*, Springer-Verlag, New York, 1994.

[DE91] L. Dieci and D. Estep, *Some stability aspects of schemes for the adaptive integration of stiff initial value problems*, SIAM J. Sci. Stat. Comput. **12** (1991), 1284–1303.

[Duf01] D. Duffy, *Green's functions with applications*, Chapman and Hall/CRC, New York, 2001.

[EEHJ95] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson, *Introduction to adaptive methods for differential equations*, Acta Numerica (1995), 105–158.

[EEHJ96] ———, *Computational differential equations*, Cambridge University Press, New York, 1996.

[EH02] D. Estep and M. Holst, *FETkLab: Finite Element Toolkit for MATLAB*, 2002, can be obtained from http://www.fetk.org.

[EHM02] D. Estep, M. Holst, and D. Mikulencak, *Accounting for stability:* a posteriori *error estimates based on residuals and variational analysis*, Comm. Num. Meth. Engin. **18** (2002), 15–30.

[EJ91] K. Eriksson and C. Johnson, *Adaptive finite element methods for parabolic problems I: A linear model problem*, SIAM J. Numer. Anal. **28** (1991), 43–77.

[ELW00] Donald J. Estep, Mats G. Larson, and Roy D. Williams, *Estimating the error of numerical solutions of systems of reaction-diffusion equations*, Memoirs A.M.S. **146** (2000), 1–109.

[Est94] D. Estep, *An analysis of numerical approximations of metastable solutions of the bistable equation*, Nonlinearity **7** (1994), 1445–1462.

[Est95] ———, *A posteriori error bounds and global error control for approximations of ordinary differential equations*, SIAM J. Numer. Anal. **32** (1995), 1–48.

[Est02] ———, *Practical analysis in one variable*, Springer-Verlag, New York, 2002.

[EW96] D. Estep and R. Williams, *Accurate parallel integration of large sparse systems of differential equations*, Math. Models Meth. Appl. Sci. **6** (1996), 535–568.

[FH89] G. Fusco and J. Hale, *Slow-motion manifolds, dormant instability, and singular perturbations*, J. Dynam. Differ. Equa. **1** (1989), 75–94.

[Fol84] G. Folland, *Real analysis*, John Wiley and Sons, New York, 1984.

[Fol95] ———, *Introduction to partial differential equations*, Princeton University Press, Princeton, New Jersey, 1995.

[GS00] M. Griebel and M. Schweitzer, *A particle-partition of unity method for the solution of elliptic, parabolic, hyperbolic pdes*, SIAM J. Sci. Comput. **22** (2000), 853–890.

[GS02] M. Giles and E. Süli, *Adjoint methods for pdes: a posteriori error analysis and postprocessing by duality*, Acta Numerica (2002), 145–236.

[Hal87] P. Halmos, *Finite-dimensional vector spaces*, Springer-Verlag, New York, 1987.

[Hol01] M. Holst, *Adaptive numerical treatment of elliptic systems on manifolds*, Adv. Comput. Math. **15** (2001), no. 1–4, 139–191.

[Hol02] ———, *Applications of domain decomposition and partition of unity methods in physics and geometry (plenary paper)*, Fourteenth International Conference on Domain Decomposition Methods, January 2002, Mexico City, Mexico, 2002.

[JLTW87] C. Johnson, S. Larsson, V. Thomée, and L. Wahlbin, *Error estimates for spatially discrete approximations of semilinear parabolic equations with nonsmooth initial data*, Math. Comput. **49** (1987), 331–357.

[KA64] L. Kantorovich and G. Akilov, *Functional analysis in normed spaces*, Macmillan Company, New York, 1964.

[Lan96] C. Lanczos, *Linear differential operators*, SIAM, Philadelphia, 1996.

[LM72] J. Lions and E. Magenes, *Non-homogeneous boundary value problems and applications*, vol. 1, Springer-Verlag, New York, 1972.

[MAS96] G. Marchuk, V. Agoshkov, and V. Shutyaev, *Adjoint equations and perturbation algorithms in nonlinear problems*, CRC Press, New York, 1996.

[RR93] M. Renardy and R. Rogers, *An introduction to partial differential equations*, Springer-Verlag, New York, 1993.

[Sch01] M. Schechter, *Principles of functional analysis*, American Mathematical Society, Providence, Rhode Island, 2001.

[Smo01] J. Smoller, *Shock waves and reaction-diffusion equations*, Springer-Verlag, New York, 2001.

[XZ00] J. Xu and A. Zhou, *Local and parallel finite element algorithms based on two-grid discretizations*, Math. Comput. **69** (2000), 881–909.

[Zau98] E. Zauderer, *Partial differential equations of applied mathematics*, John Wiley and Sons, New York, 1998.