DISSERTATION

VARIATIONAL METHODS FOR UNCERTAINTY QUANTIFICATION

Submitted by

David Neckels

Department of Mathematics

In partial fulfillment of the requirements

for the degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Fall 2005

COLORADO STATE UNIVERSITY

August 31, 2005

WE HEREBY RECOMMEND THAT THE DISSERTATION PREPARED UNDER OUR SUPERVISION BY DAVID NECKELS ENTITLED "VARIATIONAL METHODS FOR UNCERTAINTY QUANTIFICATION " BE ACCEPTED AS FULFILLING IN PART REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY.

Committee on Graduate Work

Dr. Dennis Ojima

Dr. Thompson Hobbs

Dr. Richard Davis

Adviser:
Dr. Donald Estep

Department Head/Committee Member:
Dr. Simon Tavener

ABSTRACT OF DISSERTATION

VARIATIONAL METHODS FOR UNCERTAINTY QUANTIFICATION

A very common problem in science and engineering is the determination of the effects of uncertainty or variation in parameters and data on the output of a deterministic nonlinear operator. For example, such variations may describe the effect of experimental error or may arise as part of a sensitivity analysis of the model. In this dissertation, we present an approach for ascertaining the effects of variations and uncertainty in parameters in a differential equation that is based on techniques borrowed from *a posteriori error* analysis for finite element methods. The generalized Green's function is used to describe how variation propagates into the solution around localized points in the parameter space. This information can be used either to create a higher order method or produce an error estimate for information computed from a given representation. In the latter case, this provides the basis for adaptive sampling. Both the higher order method and the adaptive sampling methods provide a powerful alternative to traditional Monte-Carlo methods, and provide a more detailed picture of how various regions in parameter space affect the output of the nonlinear operator.

David Neckels
Department of Mathematics
Colorado State University
Fort Collins, Colorado 80523
Fall 2005

ACKNOWLEDGEMENTS

I extend thanks to my advisor, Don Estep, for his excellent advice and for his insightful guidance throughout this research project. From him I learned not only the techniques of mathematics, but also a tremendous amount about the art and profession of mathematics.

To the educators throughout my school career for their efforts, especially my excellent teachers at Laytonville High, who first showed me the magic and power of knowledge.

Also to my family for providing a lifetime of friendship and encouragement, and for keeping me off of the street during these "less than replete" years as a student. Only my family knows that, at this point, I am "educated (far) beyond my intelligence", but I am sure they will not let on to anyone.

And to my wife, Carina; it is through her encouragement that I starting down this road which has ended in the fulfillment of a lifetime goal. I don't know if she would encourage me to do this if she had a second chance, but once was enough! I thank her for her wonderful friendship throughout this project.

This work was supported by numerous grants and fellowships. Thanks especially to the CSU Mathematics Department, and the PRIMES program for their fellowships and grants. Other grants include: Department of Energy: DE-FG02-04ER25620, Sandia Corporation: PO299784, National Aeronautics and Space Administration: NNG04GH63G, National Science Foundation: DMS-0107832, DGE-0221595003, and MSPA-CSE-0434354.

DEDICATION

To Rosalie,
for her unfathomable courage

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

UNCERTAINTY QUANTIFICATION

1.1 Introduction

In this paper, we study an important and ubiquitous problem in science and engineering that can be described abstractly like this: Suppose that a deterministic nonlinear operator $\mathcal{F}$ maps a space of inputs, consisting of data and parameters, into a space of outputs so that to each fixed input state, $\mathcal{F}$ associates a unique output state. In our particular case, $\mathcal{F}$ is the solution operator for a differential equation. The problem is to describe the resulting output function.

This problem is usually framed in terms of uncertainty analysis. In this setting, some of the data and/or the parameters for $\mathcal{F}$ are unknown within a given range and/or subject to random variation. For example, uncertainty and variation in data and parameters might arise from experimental or modeling error. The problem is to determine the effect of the uncertainty or variation on the output of the operator. It is often the case that not all possible configurations of the input are equally plausible. It is natural, therefore, to weight different scenarios with different probabilities. This leads to the idea of considering the input to be a random vector associated with some probability distribution, and the output then becomes a random vector associated with a *new* distribution.

To illustrate, we plot the output distribution of the nonlinear function of one parameter $q(\lambda) = \tanh(2\lambda)/\tanh(2)$ for several distributions of the parameter λ on $[-1, 1]$ along with q in Fig. 1.1. The function q acts in a nearly linear fashion on the "narrow" normal distributions with small variance, so the resulting distributions are again approximately normal. For the "wide" distributions with large variance, the nonlinear behavior of q causes more of the mass in the output distribution to be concentrated around the values of 0 or 2. A key feature of this problem is demonstrated in this

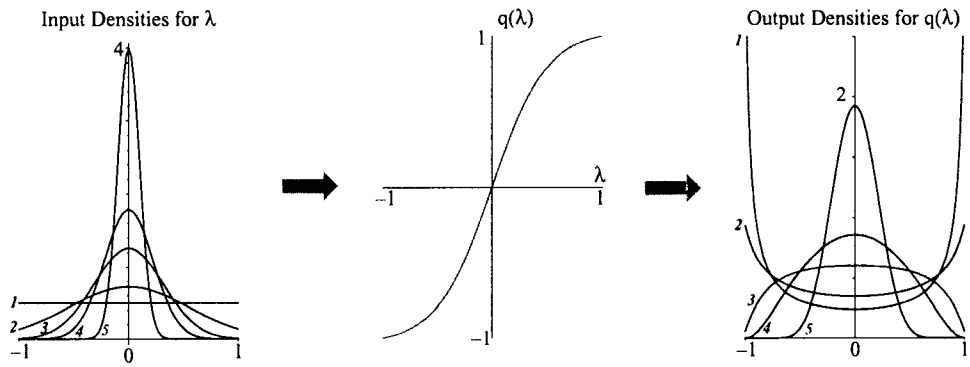

Figure 1.1: Left we plot several Input distributions for λ, normal with different variances, as well as the uniform distribution on $[-1, 1]$. We plot the function q in the middle. On the right, we plot the output distributions for $q(\lambda)$. As the variance of the input distribution grows, more of the mass in the output distribution is concentrated at -1 and 1.

example. When the input distribution has sufficiently small variance, the linearization of the function applied to the input distribution provides a reasonable approximation of the output distribution. On the other hand, nonlinearity can significantly affect the output distribution if the input distribution has a sufficiently large variance with respect to the nonlinear behavior of the function. In this case, the effect of the nonlinearity is difficult, if not impossible, to predict.

Another frame for this problem is sensitivity analysis, in which the goal is to quantify the sensitivity of the output information computed from a model with respect to variations in the input parameters. As well as revealing fundamental stability properties of a model, sensitivity analysis also has important applications. For example, it can be used to estimate the accuracy required in experimental measurements needed to compute parameter values. Unfortunately, it is usually very difficult to obtain accurate *a priori* bounds on the effects of variations in parameters. For example, a Gronwall argument is a standard tool for evolutionary problems and it yields bounds that grow exponentially in time regardless of the true behavior.

The Monte-Carlo method is the standard tool for determining the effects of variation in parameters and data on the output of a nonlinear function. In this approach, a sample of the input space is selected at random according to its distribution, and the model is solved for these parameter values. The resulting collection of output values can be used to approximate desired information, such as the distribution. The Monte-Carlo method is robust and easy to implement, especially on parallel computers. It is used throughout science and engineering. The drawback of Monte-Carlo is that, depending on the quantity of interest, Monte-Carlo estimates may converge slowly, requiring a large number of simulations in order to achieve good accuracy.

When the operator $\mathcal{F}$ is expensive or difficult to evaluate, the Monte-Carlo method is expensive to the point of being impractical. For example, weather prediction is plagued by this issue. In order to make Monte-Carlo more practical, when $\mathcal{F}$ is a complicated, nonlinear operator, it is often necessary to sample proportional to the variation of the output. Simply

sampling according to the distribution of the input may not suffice. In the example above, for instance, accurately representing the output corresponding to a *uniform* input distribution in an efficient way requires increased sampling near the boundaries.

In this paper, we present a new approach for ascertaining the effects of variations in parameters and data on a model in the case that the model depends smoothly and deterministically on the inputs. The focus on *deterministic* models, which may be affected by random processes through parameters and data, is worth emphasizing. Such problems have distinct features as compared to models described by stochastic differential equations. We use the distinct features of deterministic models to good effect when analyzing the effects of variation.

Our approach is based on techniques borrowed from *a posteriori* error analysis for finite element methods. The goal of *a posteriori* error analysis is to provide an accurate error estimate of an approximate solution using information obtained from the numerical solution as much as possible (see [9, 8]). One avenue [10, 15, 11] to *a posteriori* error analysis uses variational arguments involving the generalized Green's function. The generalized Green's function solves the (linearized) adjoint problem with data specific to the information to be computed from the solution of the original problem and describes how local variation in the model, parameters, and data propagates into the solution. In this paper, we apply this *a posteriori* analysis to the problem of determining the effects of variations in parameters and data on a model and show how information obtained from the generalized Green's function can be used either to create a higher order method or produce an error estimate for a given representation.

The organziation of the dissertation is as follows: In the following chapter, the Monte-Carlo approach is described in detail, including the convergence of this method and the variance reduction techniques which are commonly used. Next, the adjoint operator is described, to demonstrate key features which play a role in the methods we propose. Additionaly, we discuss some ways in which the adjoint is currently used. Next, an introduction of the basic idea behind the methods in this paper by demonstrating them in a simple case, a finite dimensional system of nonlinear equations. We present some of the author's perturbation theorems, which underly the methods that follow, the "Higher Order Parameter Sampling" (HOPS), "Reliably Accurate Parameter Sampling" (RAPS), and "Fast Adaptive Parameter Sampling" (FAPS) methods, which are then presented. We examine these techniques in a number of examples involving ODE's and PDE's. We discuss some possible future applications of the methods in inverse probability problems, and then close with a discussion of the software and numerical algorithms that are required to realize the techniques in the dissertation.

Chapter 2

THE MONTE-CARLO METHOD

2.1 Introduction and History

The Monte-Carlo method is a technique that provides approximate solutions to certain mathematical problems by using the theory of random numbers.

A number of mathematicians and physicists, including Lord Rayleigh, Courant, Kolmogorov, and Petrowsky, early on realized that many problems involving diffusion processes had connections to random walks.

In the World-War II era, the Monte-Carlo method was adopted as a research tool during work on the atomic bomb. A Polish born mathematician Stanislaw Ulam, who worked with John von Neumann on the Manhattan Project writes [7],

> The first thoughts and attempts I made to practice [the Monte Carlo Method] were suggested by a question which occurred to me in 1946 as I was convalescing from an illness and playing solitaires. The question was what are the chances that a Canfield solitaire laid out with 52 cards will come out successfully? After spending a lot of time trying to estimate them by pure combinatorial calculations, I wondered whether a more practical method than abstract thinking might not be to lay it out

> say one hundred times and simply observe and count the number of successful plays. This was already possible to envisage with the beginning of the new era of fast computers, and I immediately thought of problems of neutron diffusion and other questions of mathematical physics, and more generally how to change processes described by certain differential equations into an equivalent form interpretable as a succession of random operations. Later [in 1946, I] described the idea to John von Neumann, and we began to plan actual calculations.

Problems involving the simulation of neutron diffusion were cast as stochastic processes and solved by random simulations. In the years following there was an attempt to solve "every problem in sight" by Monte-Carlo methods, but this was not incredibly successful and it went into latency for some time.

The method, however, still is the only way to solve certain problems. For problems involving very high dimensions, for example, the Monte-Carlo method often performs asymptotically better than traditional numerical methods.

In this chapter, we first briefly describe some concepts from probability, and then formally introduce the Monte-Carlo problem as the evaluation of an integral. We discuss convergence of the method and the error analysis. We present some of the important techniques used to make the method more efficient (Variance Reduction), and provide some examples.

2.2 Probability

The Monte-Carlo methods are based in the framework of probability, so we briefly discuss some key ideas from probability that are needed.

We wish to formalize intuitive notions about probabilities gained through experimentations with coin tossing, dice, and other such diversions. Such experiments have a set of *outcomes* $\mathcal{O}$ (e.g. $\{1,2,3,4,5,6\}$ in the case of the die roll). We are interested in assigning probabilities to *events* which formally are sets $O \in 2^{\mathcal{O}}$ (the power set of $\mathcal{O}$). For example, the event that the die toss is even is the set $\{2,4,6\}$. Probabilities, then are real numbers p_i (one for each element of $\mathcal{O}$ that satisfy

$$p_i \geq 0, \sum_{i \in \mathcal{O}} p_i = 1.$$

The probability of an event $O \in \mathcal{O}$ is defined to be $P(O) = \sum_{i \in O} p_i$. This construct satisfies the important properties

i) $P(\emptyset) = 0$

ii) $P(\mathcal{O}) = 1$

iii) $\sum_i P(A_i) = P(\cup A_i)$ when $(A_i \cap A_j = \emptyset, i \neq j)$

iv) $\sum_i P(A_i) \leq P(\cup A_i)$

We wish to extend these ideas to more complicated outcome spaces such as infinite coin tosses $HTHTHHHTHHTTTHTT\ldots$, spaces of functions, sets in $\mathbb{R}^n$, and other settings. This leads to complications which are solved by the introduction of measure theory. The notion of probability in such spaces, then, involves a triple $(\Omega, P, \mathcal{F})$ where Ω is the space of *outcomes*, $\mathcal{F}$ is a σ-field of sets in Ω, and P is a measure on Ω satisfying the properties above, extended to countable summations. It turns out that often the collection of sets $\mathcal{F}$ is smaller than 2^{Ω}, since the inclusion of all sets leads to a breakdown in the intuitive properties $i - iv$ above. Hence we obtain the measurable sets.

A *random variable* is defined to be a function from Ω into $\mathbb{R}$ which is measurable $\mathcal{F}$, i.e. we can calculate probabilities of the type $P(f \in B)$ for borel sets B.

Concepts such as the average require integration theory, which is possible for such measurable f. We define the *expected value* (i.e. the *average*) of a random variable to be

$$E[f] = \int_{\Omega} f(\omega) dP(\omega).$$

The variance of a random variable is the quantity $E[f^2] - E[f]^2$.

We call the function $F(x) = P(f \leq x)$ the *cumulative distribution function* of f. For a (finite) collection of random variables $f_1, \cdots, f_n$, we can consider their joint cumulative distribution function $F(\boldsymbol{x}) = P(f_1 \leq x_1, \cdots, f_n \leq x_n)$. If it is the case that for all $\boldsymbol{x}$ we have $F(\boldsymbol{x}) = \prod_{i=1}^{n} P(f_i \leq x_i)$, then the random variables $f_i, i = 1 \cdots n$ are called *independent*. Effectively, this says that the knowledge of the value of any one of the variables does not give information concerning the values of the others (the probabilities are multiplicative). It is often useful to consider a set of random variables as a single vector $\boldsymbol{f} = (f_1, \ldots, f_n)$, with corresponding distribution function on $\mathbb{R}^n$ given by $F(\boldsymbol{x}) = P(f_1 \leq x_1, \ldots, f_n \leq x_n)$, as above. We generalize independence to random vectors by saying that two random vectors $\boldsymbol{f}, \boldsymbol{g}$ are independent if

$$P(f_1 \leq x_1, \ldots, f_n \leq x_n, g_1 \leq y_1, \ldots, g_m \leq y_m) = P(f_1 \leq x_1, \ldots, f_n \leq x_n) P(g_1 \leq y_1, \ldots, g_m \leq y_m)$$

for all choices of $\boldsymbol{x} = (x_1, \ldots, x_n)$ and $\boldsymbol{y} = (y_1, \ldots, y_m)$.

For random variable formula

$$\int_\Omega g(\boldsymbol{f}(\omega))\mathrm{d}P(\omega) = \int_{\mathbb{R}^n} g(\boldsymbol{x})\mathrm{d}F(\boldsymbol{x}),$$

$$\int_\Omega g(\boldsymbol{f}(\omega))\mathrm{d}P(\omega) = \int_{\mathbb{R}^n} g(\boldsymbol{x})\mathrm{d}F(\boldsymbol{x}),$$

where the measure $\mathrm{d}F$ is called the *distribution* of the random vector, and extends the idea of the Lebesgue-Stieltjes integral to higher dimensions.

If the function $F(x)$ is differentiable, then setting

$$f(\boldsymbol{x}) = \frac{\partial^d F(\boldsymbol{x})}{\partial_{x_1} \dots \partial_{x_d}},$$

we find that

$$\int_A \mathrm{d}F(\boldsymbol{x}) = \int_A f(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$$

for any (measurable) set A. The function f is called the probability density corresponding to the measure $\mathrm{d}F$.

Most of the results in Monte-Carlo are a result of a powerful theorem involving independent random variables, the **Strong Law of Large Numbers**.

Theorem 2.2.1. *Let the sequence of random variables $\{X_i\}_{i=1}^{\infty}$ be pairwise independent, identically distributed ($P(X_i \leq x) = P(X_j \leq x), \forall i, j, x$), and have finite mean. Then $\sum_{i=1}^{n} X_i/n \to \mu$ with probability 1, where μ is the (common) expected value of the X_i.*

The error analysis of the method relies on the **Central Limit Theorem**, an astonishing result relating the normal distribution to averages of independent random variables.

Theorem 2.2.2. *Suppose $\{X_i\}_{i=1}^{n}$ is an independent sequence of random variables having the same distribution with mean μ and finite positive variance σ^2. If $S_n = X_1 + \cdots + X_n$, then*

$$\frac{S_n - n\mu}{\sigma\sqrt{n}} \Rightarrow N(0,1)$$

Here $\Rightarrow$ indicates convergence in distribution, which means the cumulative distribution functions converge for each x (technically for each x where the distribution functions are continuous, but this is not an issue above). Also, independence for an infinite set of functions is defined to be independence for any finite subcollection.

For a proof of both of these theorems see [2].

2.3 Approximation of Integrals

The Standard Monte-Carlo problem can be phrased as the calculation of an integral,

$$\zeta = \int_{\mathbb{R}^d} \kappa(\boldsymbol{z})\, \mathrm{d}F(\boldsymbol{z}), \tag{2.1}$$

for a distribution function F on $\mathbb{R}^d$.

For example, the estimation of the volume of a bounded region $\mathcal{G}$ in $\mathbb{R}^n$ may be computed in this way. We simply contain the region in a rectangle $[a,b]^d$, and let $U_i, i = 1 \ldots d$ be uniform (independent) random variables on $[a,b]$, and set $\boldsymbol{U} = (U_1, \ldots, U_d)$. We let $\mathbf{1}_{\mathcal{G}}(\boldsymbol{x})$ be the indicator function of $\mathcal{G}$ (1 on $\mathcal{G}$, 0 everywhere else). By the change of variable formula we have

$$\begin{aligned} E[\mathbf{1}_{\mathcal{G}}(\boldsymbol{U})] &= \int_{\mathbb{R}^d} \mathbf{1}_{\mathcal{G}}(\boldsymbol{x}) \mathrm{d}U(\boldsymbol{x}) \\ &= \frac{1}{(b-a)^d} \int_{[a,b]^d} \mathbf{1}_{\mathcal{G}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \\ &= \frac{1}{(b-a)^d} |\mathcal{G}| \end{aligned}$$

Now, using the Strong Law of Large numbers, we take a sequence of independent vectors $\boldsymbol{U}_i$ with the distribution above and we have

$$\frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\mathcal{G}}(\boldsymbol{U}_i) \rightarrow E[\mathbf{1}_{\mathcal{G}}(\boldsymbol{U})] = \frac{1}{(b-a)^d} |\mathcal{G}|$$

(if $\boldsymbol{U}_i$ are independent as vectors, $g(\boldsymbol{U}_i)$ are independent as random variables). The quantity on the left is the number of times $\boldsymbol{U}_i$ lands in the region, divided by the number of trials, which approaches the ratio of the volume of $|\mathcal{G}|$ to the volume of the box that contains it. Hence by multiplying through by $(b-a)^d$, we have an estimator for the area $|\mathcal{G}|$.

More generally, if we have an algorithm for constructing independent random vectors $\boldsymbol{f}_i$ with distribution function $\mathrm{d}F$, then by the same reasoning as above,

$$\frac{1}{n}\sum_{i=1}^{n}\kappa(\boldsymbol{f}_i) \to \int_{\mathbb{R}^d}\kappa(\boldsymbol{z})\mathrm{d}F(\boldsymbol{z}) = \zeta.$$

At least one case in which this type of approach is useful is when it is impossible to write an algebraic epression for $\mathbf{1}_{\mathcal{G}}$. For example, in Fig. 2.1, we approximate the volume of the Mandelbrot Set for c uniformly distributed over $[-2,2]^2$ and $\mathbf{1}_{\mathcal{G}}$ the indicator of the Mandelbrot Set (which is approximated by iterating the point and seeing if its orbit stays bounded). It is known that the true value is approximately 1.507.

2.4 Error Estimates

The errors in approximating an integral by the Monte-Carlo method may be analyzed either by describing their distributional properties (the Central Limit Theorem approach), or by using large-deviation theory, which leads to the *Law of the Iterated Logarithm.*

2.4.1 Error via the Central Limit Theorem

In order to quantify the rate of convergence for these approximations to the integral, we use the Central Limit Theorem, Thm. 2.2.2, which states

$$\frac{\sum_{i=1}^{n}\kappa(\boldsymbol{f}_i) - n\zeta}{\sigma\sqrt{n}} \Rightarrow N(0,1), \tag{2.2}$$

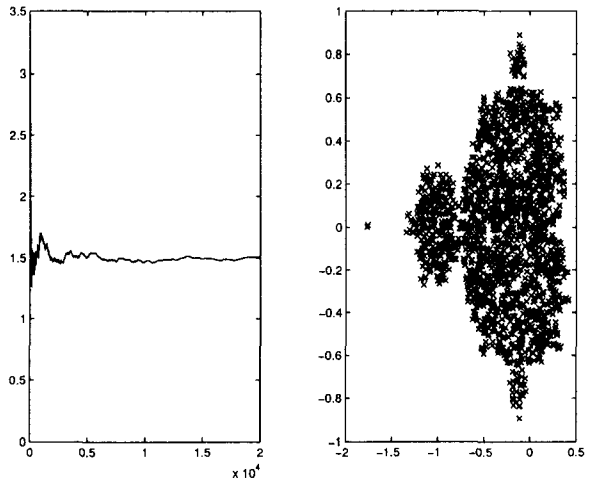

Figure 2.1: Left: *Area of Mandelbrot Set by Monte-Carlo. The $x-axis$ is the number of iterations. For each c, z^2+c is iterated 1000 times to test inclusion.* Right: *Samples points c which are in the set.*

where

$$\sigma^2 = \int_{\mathbb{R}^d} \kappa(\boldsymbol{z})^2 \mathrm{d}F(\boldsymbol{z}) - \zeta^2, \tag{2.3}$$

and $\boldsymbol{f}_i \sim \mathrm{d}F$. [1] Since $\mathrm{var}(aX) = a^2\mathrm{var}(X)$ for any random variable X, if we assume that n is large enough so the quantity on the left in (2.2) is close to a $N(0,1)$ variable, multiplication by $\sigma/\sqrt{n}$ gives

$$E_n = \left(\frac{\sum_{i=1}^n \kappa(\boldsymbol{f}_i)}{n} - \zeta \right) \approx N(0, \sigma^2/n), \tag{2.4}$$

where the "$\approx$" means "has the approximate distribution".

In deterministic error analysis, we attempt to insure that $|E_n| \leq \epsilon$ for a given n or all $n \geq N$ for some N. This is generally impossible in the random case, however. Even though $\lim_{n\to\infty} P(|E_n| \leq \epsilon) = 1$, unfortunately for each *finite* n we have $P(|E_n| > \epsilon) > 0$ whenever $\sigma \neq 0$. These problems leads to the use of the (ϵ, δ) criterion for convergence:

---

[1] In this context the symbol $\sim$ indicates that the variable to the left is a sample drawn from the distribution which appears to the right.

Theorem 2.4.1. *Given $\epsilon > 0$ and $0 \leq \delta < 1$, there is an integer $N(\epsilon, \delta)$ so that*

$$P(|E_n| \leq \epsilon) > 1 - \delta$$

for all $n \geq N$.

The number δ measures the *confidence* that the error is small. By making $\delta > 0$ (reducing the confidence slightly), we are able to achieve the error estimate for finite n.

Proof. We set $A_n = [|E_n| > \epsilon]$, and $B_n = \cup_{i=n}^{\infty} A_i$. By the Strong Law of Large Numbers, $P(\cap_{n=1}^{\infty} B_n) = 0$. Since $B_1 \supseteq B_2 \supseteq \cdots$,

$$P(\cap_{n=1}^{\infty} B_n) = \lim_n P(B_n) = 0.$$

Choosing N so $P(B_m) < \delta$ for $m \geq N$,

$$P([|E_m| \leq \epsilon]) = 1 - P(A_m) \geq 1 - P(B_m) > 1 - \delta. \tag{2.5}$$

□

To estimate the required $N(\epsilon, \delta)$, we invert (2.5) for the required N. However, this requires that we know the distribution of E_n, which we do not. We use the convergence to normality in (2.4), and make the approximation

$$P(|E_n| \leq \epsilon) \approx 2\Phi\left(\frac{\epsilon}{\sigma\sqrt{n}}\right) - 1,$$

where Φ is the standard normal density function, defined by

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-x^2/2} \mathrm{d}x,$$

and the transformation $2\Phi - 1$ accounts for the absolute values on E_n.

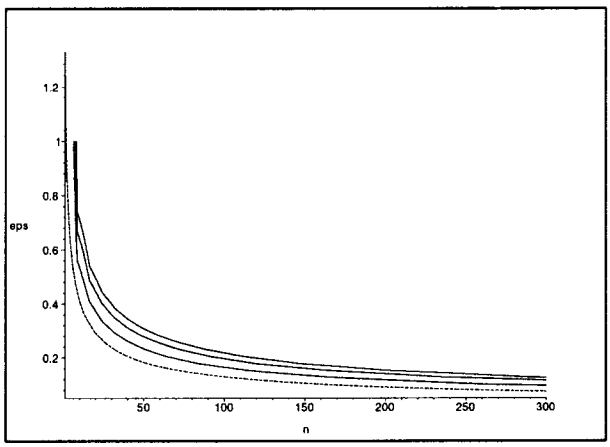

Figure 2.2: Implicit plots of $P(|E_n| \leq \epsilon) = 1 - \delta$ for $\delta = \{0.1, 0.5, 0.3\}$, $\sigma = 1$, along with $\frac{1}{\sqrt{n}}$ (bottom) (assuming normality).

It may be shown (as in Fig. 2.2) that this results in the relationship

$$\epsilon \approx \frac{\sigma C}{\sqrt{n}}$$

Note that in general we must multiply n by 4 in order to halve the error.

Despite this relatively slow convergence, one of the most attractive things about the Monte-Carlo technique is that this error depends only on the variance of the function, and not directly on the dimension of the integral. For deterministic quadrature rules, the error is more like $Cn^{-\frac{1}{d}}$ [17].

2.4.2 Errors via The Iterated Logarithm

Whereas the Central limit theorem describes the asymptotics of the distribution of the error in approximating an integral, it says nothing about the convergence of a specific estimator, or a *realization* of the process. Information about this convergence is provided by a companion theorem to the Strong Law of Large Numbers, namely the *Law of the Iterated Logarithm* [2],

Theorem 2.4.2. *Let $S_n = X_1 + \cdots + X_n$, where the X_n are independent, identically distributed simple random variables with mean 0 and variance 1. Then*

$$P\left[\limsup_n \frac{S_n}{\sqrt{2n \log\log n}} = 1\right] = 1.$$

For example, to compute the average value of a function g over a domain Λ, the Monte-Carlo approximation is given by

$$\frac{1}{\mathrm{Vol}(\Lambda)} \int_\Lambda g(\lambda)\, \mathrm{d}\lambda \approx \frac{1}{n} \sum_{i=1}^{n} g(\lambda_i(\omega)),$$

where $\{\lambda_i\}_{i=1}^n$ are independent with uniform distribution in Λ. With $X_i = g(\lambda_i)$ in the theorem above, we have

$$\sigma^2 = \frac{1}{\mathrm{Vol}(\Lambda)^2} \int_\Lambda g(\lambda)^2\, \mathrm{d}\lambda - \left(\frac{1}{\mathrm{Vol}(\Lambda)} \int_\Lambda g(\lambda)\, \mathrm{d}\lambda\right)^2.$$

We wish to examine the behavior of the error for a given realization ω, which is

$$\epsilon(n, \omega) = \frac{1}{n} \sum_{i=1}^{n} g(\lambda_i(\omega)) - \frac{1}{\mathrm{Vol}(\Lambda)} \int_\Lambda g(\lambda)\, \mathrm{d}\lambda.$$

Using the *Law of the Iterated Logarithm*,

$$P\left(\limsup_n \frac{\epsilon(n, \omega)}{\frac{1}{n}\sigma\sqrt{2n \log\log n}} = 1\right) = 1, \tag{2.6}$$

$$P\left(\liminf_n \frac{\epsilon(n, \omega)}{\frac{1}{n}\sigma\sqrt{2n \log\log n}} = -1\right) = 1. \tag{2.7}$$

This means that with probability 1, for any $C > 1$, all but a finite number of the errors $\epsilon(n, \omega)$ are bounded above and below by

$$\pm C \frac{\sigma\sqrt{2 \log\log n}}{\sqrt{n}}.$$

To illustrate, we compute the average value of $\lambda_1^2 + \lambda_2^2$ over $[-1, 1] \times [-1, 1]$ and plot the results in Fig. 2.3.

This result shows that the convergence is very near the $\sigma/\sqrt{n}$ commonly given in the literature, but is slightly more than this (by a factor of $\sqrt{\log\log n}$).

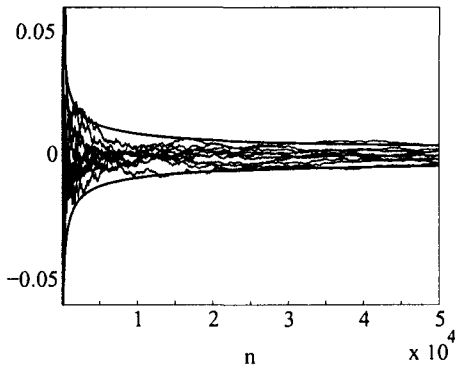

Figure 2.3: Using the Monte-Carlo method to approximate the average value of $\lambda_1^2 + \lambda_2^2$ over $[-1, 1] \times [-1, 1]$. We plot values of the error $\epsilon(n, \omega)$ as a function of increasing n for 10 different ω, along with the bounds given by the *Law of the Iterated Logarithm.*

2.5 Variance Reduction Techniques

The effectiveness of the Monte-Carlo method depends greatly on the variance of the integrand σ as defined in (2.3). In fact, since the error is proportional to σ, while it is only proportional to $\frac{1}{\sqrt{n}}$, reducing σ has a stronger effect than increasing n.

For example, consider estimating the area of the Mandelbrot Set. Intuition suggests we should select the smallest containing rectangle. It seems like a waste to sample from a larger box, say $[-10, 10]^2$, since many of the sample points land in regions that say nothing about the area we are after, even though it is still possible to to create a consistent sampler with such a box. The difference between the smaller and larger boxes is in the variance of the estimator. The larger box has a larger variance, and so, on average, we need more samples in order to get an estimate with the same error.

In general, setting up the integral carefully makes it possible to reduce σ, and therefore reduce the error in the estimator. Several approaches have

been developed to systematically reduce the variance. We consider two in this thesis.

2.5.1 Importance Sampling

In the case that $\mathrm{d}F$ has a density, we may write

$$\zeta = \int \kappa(\boldsymbol{x})\mathrm{d}F(\boldsymbol{X}) = \int \kappa(\boldsymbol{x})f(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x} = \int \kappa^*(\boldsymbol{x})f^*(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$$

where $\kappa^*(\boldsymbol{x}) = \kappa(\boldsymbol{x})f(\boldsymbol{x})/f^*(\boldsymbol{x})$ and f^* is another density. This yields another estimator for ζ, namely

$$\frac{1}{n}\sum_{i=1}^{n}\kappa^*(\boldsymbol{f}_i^*),$$

where $\boldsymbol{f}_i^*$ are vectors with distribution f^*. With σ as before and σ^* the variance of the estimator with κ^*, f^*, we have

$$\sigma^2 - (\sigma^*)^2 = \int \kappa^2(\boldsymbol{x})\left(1 - \frac{f(\boldsymbol{x})}{f^*(\boldsymbol{x})}\right)f(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$$

In order that the variance associated with f^* yields a benefit, we generally require that $f^* > f$ whenever $\kappa^2 f$ is large, i.e., *important*. The new sampling scheme then samples more frequently from the *important* region. Since f^* must integrate to 1, this also means making $f^* < f$ when $\kappa^2 f$ is small.

Returning to the Mandelbrot example, suppose we use uniform densities f_1, f_2 on $[-10, 10]^2$ and $[-2, 2]^2$ respectively. Then letting ζ_1, ζ_2 be the integrals associated with these densities and $\kappa(\boldsymbol{x}) = \mathbf{1}_M(\boldsymbol{x})$, we get variances

$$\sigma_1^2 = |M|(\frac{1}{100} - \frac{1}{100^2}) \approx 0.0099, \sigma_2^2 = |M|(\frac{1}{16} - \frac{1}{16^2}) \approx 0.058$$

This makes it seem like the larger box is better, but since we are trying to estimate $|M|$ (the area of the Mandelbrot Set), we are really interested in

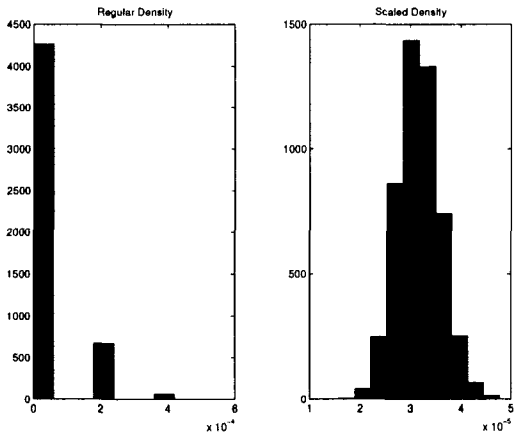

Figure 2.4: Histograms of 5000 estimates of ζ (5000 samples each) using standard and scaled densities. The standard density yields $\zeta \approx 0.0000323$, the scaled $\zeta \approx 0.00003172$. The variance are $6.3e{-}09$ and $1.8e{-}11$, respectively.

the values $100\zeta_1$ and $16\zeta_2$, which have variances 99 and 16, respectively. We are therefore better off using the smaller box. We can even try to "wrap" the Mandelbrot Set with a sequence of smaller rectangles, with a corresponding variance reduction. However, we then have to weigh the reduction in variance with the increased difficulty of generating the appropriate deviates.

As another example, when estimating *rare* events e.g., a tail event with $P(X > t)$ for large t, the Monte-Carlo method generally takes a large number of samples to converge, since it is very likely that for any small number of samples the estimate yields a 0. In order to achieve a reasonably accurate estimate of the probability of a rare event, the tails of the distribution must be emphasized. One method for doing this is choosing a density which lengthens the tails,

$$f^*(\boldsymbol{x}) = \frac{1}{a^d} f\left(\frac{\boldsymbol{x}}{a}\right),$$

for $a > 1$, which still yields a density. For example, if we wished to estimate

$$\zeta = \int \mathbf{1}_{[x \geq k]}(x) f(x)\, \mathrm{d}x,$$

which is the probability of a large value occuring, when $f(x)$ is the standard $N(0,1)$ density, for k very large, any medium or small sized sample is very likely to yield an estimate of 0. Although for this example standard quadrature is the fastest way to obtain a good estimate, consider using the density scaling. For $k = 5$, Maple gives the answer $\zeta \approx 0.00003167$.

Comparing estimators using the standard density, or the scaled density shows the more desirable nature of the scaled density (Fig. 2.4). In fact, the standard estimator shows the dangers in assuming the error of the estimate is $N(0, \sigma^2/\sqrt{n})$, since the distribution of the estimator is more like a Poisson distribution.

A number of other techniques are available. For a good overview, see [33].

2.5.2 Control Variates

We add and subtract a function ϕ,

$$\int \kappa(\boldsymbol{x}) f(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x} = \int \phi(\boldsymbol{x}) f(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x} + \int \left(\kappa(\boldsymbol{x}) - \phi(\boldsymbol{x})\right)\, f(\boldsymbol{x}) \mathrm{d}\boldsymbol{x},$$

where we can evaluate the first integral on the right relatively cheaply, either by analytic or numeric means. If ϕ is chosen to have a strong correlation with κ, then the variance of $\kappa - \phi$ is small and the estimator of the second integral on the right has small variance.

As an example, consider calculating the integral $\int_0^1 e^{-x^2}\, \mathrm{d}x$ by using the control variate e^{-x} and the uniform distribution of $[0,1]$. The integral $\int_0^1 e^{-x}\, \mathrm{d}x$ is easily evaluated to be $1 - 1/e$. We show the reduced variance of a sequence of estimates in Fig. 2.5.

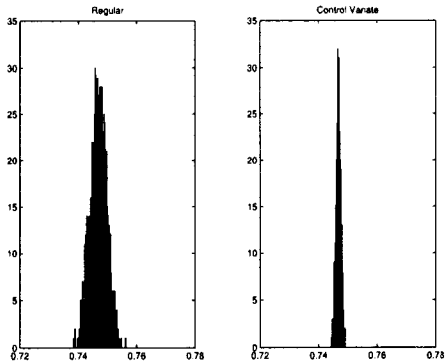

Figure 2.5: Monte Carlo estimates of $\int_0^1 e^{-x^2}\,\mathrm{d}x$ with and without control variates for a sequence of 1000 runs, 5000 samples per run. The true value is approximately 0.7468. The reduction of variance is evident in the control variate histogram.

2.5.3 Stratified Sampling

Another technique is to partition the parameter space $\mathcal{R}$ into a finite number of disjoint sets so that $\mathcal{R} = \bigcup_{i=1}^{r} \mathcal{R}_i$. We set $p_i = \int_{\mathcal{R}_i} f$ and write the equivalent expression for ζ,

$$\zeta = \sum_{i=1}^{r} p_i \zeta_i, \quad \text{where} \quad \zeta_i = \int_{\mathcal{R}_i} \kappa(\boldsymbol{x}) \frac{f(\boldsymbol{x})}{p_i}\,\mathrm{d}\boldsymbol{x}.$$

Suppose that we generate n_i samples $\boldsymbol{f}_j^i, j = 1 \ldots n_i$ chosen from the distribution of $f(\boldsymbol{x})/p_i$ restricted to $\mathcal{R}_i$, which is a probability density due to the normalization. Then the estimator

$$\bar{\zeta}_s = \sum_{i=1}^{r} \frac{p_i}{n_i} \sum_{j=1}^{n_i} \kappa(\boldsymbol{f}_j^i)$$

is unbiased with expected value ζ.

A short calculation, using the theorem that $\mathrm{var}(\sum X_i) = \sum \mathrm{var}(X_i)$ when X_i are independent random variables, yields the variance of $\bar{\zeta}_s$,

$$\mathrm{var}(\bar{\zeta}_s) = \sum_{i=1}^{r} \sigma_i^2 p_i^2 / n_i, \quad \text{where} \quad \sigma_i^2 = \int_{\mathcal{R}_i} (\kappa(\boldsymbol{x}) - \zeta_i)^2 \frac{f(\boldsymbol{x})}{p_i}\,\mathrm{d}\boldsymbol{x},$$

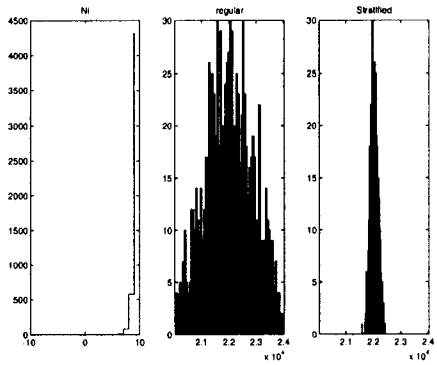

Figure 2.6: Left: *plot of* n_i. Center: *Estimates using standard Monte-Carlo.* Right: *Estimates using the stratified sampling* In both runs, $N = 5000$ and 1000 estimates are used for the histogram.

i.e., the variance is the sum of local variations of κ. It may be shown that this variance is always less than or equal to the variance of the standard estimator [17] for certain choices of n_i.

The idea behind stratified sampling is to choose the number of samples in each $\mathcal{R}_i$ so that they are proportional to the variation in the function σ_i^2 and the probability of the region p_i. This yields the greatest efficiency in making the estimate. For example, if $\epsilon > 0$ is given and we choose $n_i = \sigma_i^2 p_i/\epsilon$, the variance becomes

$$\mathrm{var}(\bar{\zeta}_s) = \sum_{i=1}^{r} \epsilon p_i = \epsilon$$

We use both standard Monte-Carlo and the stratified method to demonstrate variance reduction using the uniform distribution on $[-10, 10]$ and the function e^x in Fig. 2.6. As an estimate for σ_i^2, we assume that the variation of the integrand on an interval is proportional to the square of the derivative of the function on that interval times a function involving the length of the interval. We use $\mathcal{R}_i = [i, i+1], i = -10, \ldots, 9$ for the

stratification, and choose n_i to be $Ne^{2m_i}/\sum_j e^{2m_j}$, where m_i is the midpoint of the i^{th} interval and N is the total number of sample points. Since the length of the intervals is constant, we can ignore the interval length in this formulation. We round the n_i so that they are integers, which gives $\bar{N}$ samples. However, this number is very close to N, and we consider the sample sizes the same. We notice in this example the regions where the function has a larger derivative need to be sampled more intensely.

2.5.4 Stratified Sampling with Derivatives

Often, we are interested in the value of an integral involving a function $\kappa(\boldsymbol{\lambda}(\omega))$, where the parameter $\boldsymbol{\lambda}$ is a random variable with density f and the function depends smoothly on the parameters, i.e. $\nabla\kappa(\boldsymbol{x})$ exists. This is typical, for instance, when κ represents a deterministic model and we are interested in how the model output varies when the parameters are uncertain. If we stratify as above, dividing the region into disjoint subsets $\mathcal{R}_i, i = 1 \ldots r$, and consider the variance σ_i^2 as defined above, we can estimate these values using the derivative.

Claim 1. *If $\mathcal{R}_i$ is compact, convex, and if κ is differentiable on $\mathcal{R}_i$, then there is a point $\boldsymbol{x_i} \in \mathcal{R}_i$ such that*

$$\kappa(\boldsymbol{x}_i) = \zeta_i = \int_{\mathcal{R}_i} \kappa(\boldsymbol{x}) \frac{f(\boldsymbol{x})}{p_i} \, \mathrm{d}\boldsymbol{x}.$$

If $\mathcal{R}_i$ are rectangles or any other convex set, the theorem holds, but it also holds for more general strata, provided any two points in the set may be connected by a continuous curve. The theorem does not hold for the strata which contain the tails of the distribution, if there are any.

Proof. Since κ is differentiable, it is continuous, and so there are constants m, M and points $\boldsymbol{x}_m, \boldsymbol{x}_M \in \mathcal{R}_i$ so that

$$\kappa(\boldsymbol{x}_m) = m \leq \kappa(\boldsymbol{x}) \leq M = \kappa(\boldsymbol{x}_M), \quad \forall \boldsymbol{x} \in \mathcal{R}_i.$$

Let $\boldsymbol{c}(s)$ be a continuous curve in $\mathcal{R}_i$ with $\boldsymbol{c}(0) = \boldsymbol{x}_m$ and $\boldsymbol{c}(1) = \boldsymbol{x}_M$. Then, $\kappa(\boldsymbol{c}(s))$ is a continuous function on $[0, 1]$ that attains all points between m and M for some $s \in [0, 1]$.

Now multiplying the inequality above by f/p_i and integrating over $\mathcal{R}_i$, we get

$$m \leq \int_{\mathcal{R}_i} \kappa(\boldsymbol{x}) \frac{f(\boldsymbol{x})}{p_i} \, \mathrm{d}\boldsymbol{x} = \zeta_i \leq M.$$

Hence, there is a point $\bar{s} \in [0, 1]$ such that $\kappa(\boldsymbol{c}(\bar{s})) = \zeta_i$. Hence $\boldsymbol{x}_i = \boldsymbol{c}(\bar{s})$ is the desired point. □

As a result, to calculate an integral such as

$$\int_{\Omega} \kappa(\boldsymbol{\lambda}(\omega)) \, \mathrm{d}P(\omega) = \int_{\mathcal{R}} \kappa(\boldsymbol{x}) f_{\boldsymbol{\lambda}}(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x},$$

we may stratify the domain and estimate σ_i^2 in terms of derivatives of κ,

$$\begin{aligned}\sigma_i^2 = \int_{\mathcal{R}_i} (\kappa(\boldsymbol{x}) - \zeta_i)^2 \frac{f_{\boldsymbol{\lambda}}(\boldsymbol{x})}{p_i} \mathrm{d}\boldsymbol{x} &= \int_{\mathcal{R}_i} (\kappa(\boldsymbol{x}) - \kappa(\boldsymbol{x}_i))^2 \frac{f_{\boldsymbol{\lambda}}(\boldsymbol{x})}{p_i} \mathrm{d}\boldsymbol{x} \\ &= \int_{\mathcal{R}_i} (\nabla\kappa(\boldsymbol{\xi}_i)(\boldsymbol{x} - \boldsymbol{x}_i))^2 \frac{f_{\boldsymbol{\lambda}}(\boldsymbol{x})}{p_i} \, \mathrm{d}\boldsymbol{x},\end{aligned}$$

where $\boldsymbol{\xi}_i$ lies somewhere on the line between $\boldsymbol{x}$ and $\boldsymbol{x}_i$. This gives a bound

$$\sigma_i^2 \leq \mathrm{diam}(\mathcal{R}_i)^2 \sup_{\boldsymbol{x} \in \mathcal{R}_i} |\nabla\kappa(\boldsymbol{x})|^2,$$

and so a sampling strategy is to choose

$$n_i = C p_i |\nabla\kappa(\boldsymbol{u}_i)|^2 \mathrm{diam}(\mathcal{R}_i)^2, \tag{2.8}$$

for some point $\boldsymbol{u}_i$ in $\mathcal{R}_i$ for some C. This is, in fact, the strategy that was used in Fig. 2.6.

2.6 The Empirical Distribution function

Monte-Carlo techniques may be used to approximate the cumulative distribution function of a random variable. For a fixed x, the cumulative distribution function may be posed as the integral

$$P(X \leq x) = F(x) = \int_{\Omega} \mathbf{1}_{(-\infty,x]}\big(X(\omega)\big)\, \mathrm{d}P(\omega).$$

Provided we are able to obtain samples $X_i, i = 1, \ldots, N$ from the distribution of X, the Monte-Carlo approximation to this integral is

$$\hat{F}_N(x) = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}_{(-\infty,x]}(X_i).$$

The variance of this estimator is $\frac{1}{N}F(x)(1 - F(x))$, so it is clear that the empirical distribution function is most sensitive when $F(x) = 1/2$, i.e. at the median of the variable X.

While above we considered x as fixed, given the samples X_i, one may easily calculate $\hat{F}_N(x)$ for any x. A simple algorithm for this is to sort the samples X_i, and construct the function that has jumps of exactly $\frac{1}{N}$ at succesive X_i.

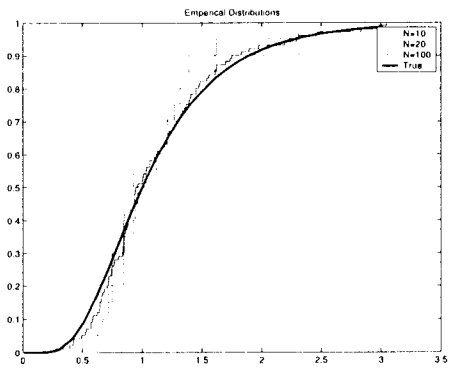

Figure 2.7: Emperical distribution functions for a lognormal random variable, $\sigma = 0.5$.

To assess the convergence $\hat{F}_N \to F$ for all x, we consider the metric on the space of distribution functions defined by

$$D_n(\omega) = \sup_x |F_n(x,\omega) - F(x)|,$$

which is known as the Kolmogorov-Smirnov, or K-S, statistic. The following, the **Glivenko-Cantelli Theorem**, holds.

Theorem 2.6.1. *If $X_1, X_2, \ldots$ are independent with common distribution function F, then $D_n \to 0$ with probability 1.*

The *Strong Law of Large Numbers* certainly guarantees that for any fixed x, $F_n(x) \to F(x)$, but this result is much stronger, stating that the convergence is, in fact, uniform in x. A proof is given in [2].

Some information on the asymptotic rate of convergence for a given realization is provided by the *Law of the Iterated Logarithm.* This law

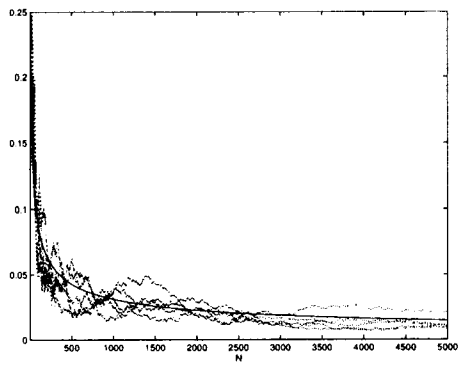

Figure 2.8: Plot of several realizations of $D_n(\omega)$, for uniformly distributed variables on $[0,1]$, along with the limit $\sqrt{\log\log n}/\sqrt{2n}$.

applies for each x, and since $D_n(\omega) \geq \hat{F}_n(x_{med},\omega) - F(x_{med})$ (where x_{med} is any point where $F(x_{med}) = \frac{1}{2}$), we have a.s. that

$$1 = \limsup_n \frac{\sqrt{2n}\left(\hat{F}_n(x_{med},\omega) - F(x_{med})\right)}{\sqrt{\log\log n}} \leq \limsup_n \frac{\sqrt{2n}D_n(\omega)}{\sqrt{\log\log n}}$$

(var $I_{[X_i \leq x_{med}]}(\omega) = \sigma^2 = (\frac{1}{2})^2$) so that for $\epsilon > 0$, a.s.

$$D_n(\omega) \geq (1-\epsilon)\frac{1}{\sqrt{2n}}\sqrt{\log\log n} \quad i.o.,$$

providing a *minimal* worst case scenario for the excursions of D_n above 0.

In fact the reverse inequality is true [6], so that a.s.

$$\limsup_n \frac{\sqrt{2n}D_n(\omega)}{\sqrt{\log\log n}} = 1,$$

which implies $P\left[D_n(\omega) \geq (1+\epsilon)\frac{1}{\sqrt{2n}}\sqrt{\log\log n} \quad i.o.\right] = 0$ for any fixed $\epsilon > 0$.

A surprising result for continuous distribution functions is that the distribution of $D_n(\omega)$ is invariant, i.e. the same regardless of the random variable!

Theorem 2.6.2. Invariance Principle: *The probability distribution of the statistic $D_n(\omega)$ is the same for all continuous distribution functions.*

Proof. See [28]. □

These results make $D_n(\omega)$ an attractive measure of accuracy. It is, however, a very strong notion of convergence that may or may not be suitable in particular applications.

2.7 Solution of Linear Operator Equations

The solutions of Fredholm integral equations

$$f(x) = g(x) + \int K(x,y)f(y)\,\mathrm{d}y$$

(solving for the function $f(x)$) and solving the related finite dimensional problem $\boldsymbol{x} = A\boldsymbol{x} + \boldsymbol{b}$ both may be accomplished by random walks. These methods are generally inefficient and only of academic interest. For a discussion, see [20] or [17].

2.8 Markov Methods

Given a countable state space S, sequence of random variables with values in S is called a Markov Chain if it has the property that

$$P(X_{i+1} = s_j | X_0 = s_0, \ldots, X_i = s_i) = P(X_{i+1} = s_j | X_i = s_i) = p_{ij}$$

for each realization and finite sequence of X_i. This condition describes a process which evolves with a dependence only on the last value of the process.

The Markov Chain Monte Carlo methods provide a method for constructing a special chain X_i such that

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \kappa(X_i) \to \int \kappa(\boldsymbol{x}) f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x}$$

The method requires only that we be able to evaluate the density $f(\boldsymbol{x})$, or even less stringent that we be able to evaluate $f(\boldsymbol{x})K$ for any K. This can be extremely useful, for instance if one wants to generate samples from a portion of a density but does not want to have to normalize by the probability of the portion.

The construction is surprisingly easy, yet proving the chain has these properties requires a very long discourse on Markov Chains which we will skip here. Details of the construction may be found in [20], and the necessary conditions for ergodicity (the convergence of the average along the trajectory to the mean value) are in [2]. Very generally, the procedure begins with a (candidate generating) random walk through $\mathbb{R}^d$. When a step is taken in this walk, the new point is only accepted conditionally, otherwise the chain remains where it is. The new step is taken conditionally

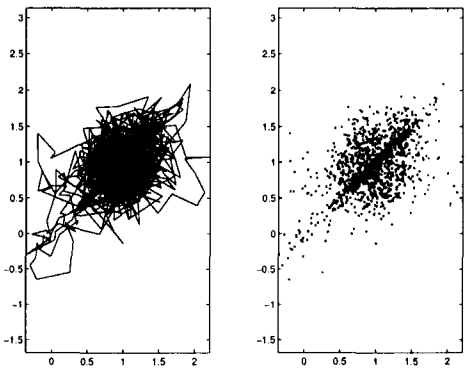

Figure 2.9: The corresponding Markov Chain walk and point values with $f(\boldsymbol{x})$ the density for a rotated and stretched Gaussian distribution.

by evaluating the relative probabilities (with respect to $f(\boldsymbol{x})$) of the current and new point. The resulting chain tends to a stationary probability distribution which is exactly that described by $f(\boldsymbol{x})$.

An example of such a walk is shown in figure (2.9).

2.9 Generating Random Numbers

Key to any of the Monte Carlo methods is the generation of random samples from the appropriate densities. It turns out that we do not have to generate samples from every possible distribution, but that we can build samples from more complicated densities from simpler variables.

For example, if we have a method to generate an independent sequence of random numbers Z_n where $P[Z_n = 0] = P[Z_n = 1] = 1/2$, e.g. coin tossing, we can generate a sequence of independent random variables $\{X_n\}$ of any distribution dF. The construction is outlined in [2], and illustrates that more complicated random variables may be constructed from simpler random variables.

2.9.1 Uniform Deviates

The basis of most Monte-Carlo methods are methods which generate a sequence of random integers between 0 and $RAND\_MAX$, where $RAND\_MAX$ is some integer near the maximum representable computer integer. Normalizing this yields a generator for random numbers in $[0, 1]$, from which we can construct most other deviates.

A very good generator [29] is simply to construct the sequence $I_{j+1} = aI_j (\text{mod } m)$ for carefully chosen a, m and initial seed I_o. This procedure is similar to the stretching and folding behavior which is found in chaotic dynamical systems, and produces a sequence of iterates which are uniformly distributed and independent. The iteration is generally implemented in assembly language, since the magnitudes of the integers are such that the operations may not be done directly in a high level language.

2.9.2 Transformations

Assuming we have a method to generate deviates for a random variable X with density $f_X(\boldsymbol{x})$, setting $Y = g(X)$ for a differentiable, one to one function g, we have

$$\begin{aligned} P[Y \in A] &= P[X \in g^{-1}(A)] \\ &= \int_{g^{-1}(A)} f_X(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x} \\ &= \int_A f_X \circ g^{-1}(\boldsymbol{x}) |\det D_{\boldsymbol{x}} g^{-1}(\boldsymbol{x})|\, \mathrm{d}\boldsymbol{x} \\ &= \int_A f_Y(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x}, \end{aligned}$$

where f_Y is the density for Y. Also, if $X_i, i = 1 \ldots$ are independent, then the $Y_i = g(X_i)$ are also independent.

Normal Deviates

The transformation method may be used to create normal deviates from uniform deviates, using the transformation

$$(n_1, n_2) = g(u_1, u_2) = \left(\sqrt{-2\ln u_1}\cos 2\pi u_2, \sqrt{-2\ln u_1}\sin 2\pi u_2\right)$$

This function transforms two deviates at a time, so a routine may save the extra one for a later call if necessary. It is a simple calculation to show that the change of variable formula yields a density for the output which is that of two independent $N(0,1)$ random variables. This calculation is best done using a package like Maple.

2.9.3 Accept-Reject

Given an easy to evaluate density $f(\boldsymbol{x})$ on $\mathbb{R}^d$ that does not correspond to any transformation rule, we take another density $g(\boldsymbol{x})$ on $\mathbb{R}^d$ from which we can draw samples. If for all $\boldsymbol{x}$ we can find a constant M, independent of $\boldsymbol{x}$, such that $f(\boldsymbol{x}) \leq Mg(\boldsymbol{x})$, then the accept-reject algorithm is

1. *Choose $Y \sim g$, and U uniform on $[0,1]$, independently.*

2. *If $U \leq f(Y)/Mg(Y)$, accept Y as a draw from f.*

If we call a sample accepted by this algorithm X (to distinguish it from the candidate Y), then for any Borel set $\mathcal{A}$ in $\mathbb{R}^d$, we have

$$\begin{aligned} P(X \in \mathcal{A}) &= P\big(Y \in \mathcal{A} | U \leq f(Y)/Mg(Y)\big) \\ &= P\big(Y \in \mathcal{A}, U \leq f(Y)/Mg(Y)\big)/P\big(U \leq f(Y)/Mg(Y)\big) \end{aligned}$$

To begin,

$$\begin{aligned} P\big(Y \in \mathcal{A}, U \le f(Y)/Mg(Y)\big) &= \int_\Omega \mathbf{1}_{\mathcal{A}}(Y(\omega))\mathbf{1}_{[0,f(Y(\omega))/Mg(Y(\omega))]}(U(\omega))\,\mathrm{d}P(\omega) \\ &= \int_{\mathbb{R}^{d+1}} \mathbf{1}_{\mathcal{A}}(\boldsymbol{y})\mathbf{1}_{[0,f(\boldsymbol{y})/Mg(\boldsymbol{y})]}(u)\,\mathrm{d}\mu_{Y,U}(\boldsymbol{y},u) \\ &= \int_{\mathcal{A}} \int_0^{f(\boldsymbol{y})/Mg(\boldsymbol{y})} \mathrm{d}u\, g(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y} \\ &= \frac{1}{M}\int_{\mathcal{A}} f(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y} \end{aligned}$$

since Y and U are independent with joint density $1 \cdot g(\boldsymbol{y})$ and the inner interval is trivially $f(\boldsymbol{y})/Mg(\boldsymbol{y})$.

Similarly,

$$\begin{aligned} P\big(U \le f(Y)/Mg(Y)\big) &= \int_{\mathbb{R}^d} \int_0^{f(\boldsymbol{y})/Mg(\boldsymbol{y})} \mathrm{d}u\, g(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y} \\ &= \frac{1}{M}\int_{\mathbb{R}^d} f(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y} = \frac{1}{M}, \end{aligned}$$

so that

$$P(X \in \mathcal{A}) = \int_{\mathcal{A}} f(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y},$$

and therefore $X \sim f$.

This method is efficient when Mg is a close bound for f, since the probability of acceptance (once $\boldsymbol{y}$ is chosen) is exactly $f(\boldsymbol{y})/Mg(\boldsymbol{y})$.

2.9.4 Other Methods

For a large class of deviates, the methods above work. Other important methods include numerically inverting the cumulative distribution function, or for very difficult multidimensional distributions the Markov-Chain method. Comprehensive discussions are given in [17, 29].

Chapter 3

THE ADJOINT

The techniques in this dissertation revolve around the use of the adjoint operator, which are described in detail in this chapter.

We first introduce the adjoint in its most abstract Banach space setting, which requires some background on Banach spaces and duality. We generalize to the Hilbert space setting, and then examine the adjoint in the setting of a Poisson problem, describing the various spaces, dual spaces, and duality pairings. This requires introducing basic Sobolev spaces and distribution theory. We then describe the same setup for a time dependent linear partial differential equation. We show how to use the adjoint in the context of numerical error estimation and we discuss techniques of adaptive error control. Lastly, we derive as an example the adjoint of a complex coupled system of equations.

3.1 Banach spaces

A Banach space is a set of elements with three important components:

- *Algebraic structure.* A Banach space is a vector space equipped with the notion of addition and scalar multiplication over a field $\mathbb{F}$, which is generally $\mathbb{R}$ in this dissertation (although it is common for $\mathbb{F}$ to be the complex numbers).

- *Norm.* A Banach space is equipped with a map $\|\cdot\| : X \to \mathbb{R}$ with the usual norm properties

  i) $\|x\| \geq 0, \forall x \in X.$

  ii) $\|x\| = 0 \Leftrightarrow x$ is the algebraic 0 in $X.$

  iii) $\|\alpha x\| = |\alpha|\|x\|, \forall x \in X, \alpha \in \mathbb{F}.$

  iv) $\|x + y\| \leq \|x\| + \|y\|, \forall x, y \in X.$

- *Completeness.* The space X is complete with respect to the norm $\|\cdot\|$. This means that any Cauchy sequence (a set $\{x_n\}_{n=1}^{\infty}$ with $\|x_n - x_m\| \to 0$ as $m, n \to \infty$) converges to a point $x \in X.$

A space with the first two properties is called a *normed linear space.*

3.1.1 Operators

A map from a normed linear space X to another normed linear space Y is called an *operator.* A *linear* operator L is such a map with the additional property that $L(\alpha x + y) = \alpha L(x) + L(y), \forall x, y \in X, \alpha \in \mathbb{F}.$ Of particular interest is the set of *continuous* (or *bounded*) operators, with the property $L(x_n) \to L(x)$ in Y whenever $x_n \to x$ in $X.$ This property is equivalent (for linear maps) to the existence of a C so that $\|L(x)\|_Y \leq C\|x\|_X, \forall x \in X.$ The mapping $L(x)$ for a linear operator is often denoted as simply $Lx.$

The space of all continuous linear operators from X to Y, denoted $\mathcal{L}(X, Y)$, is a normed linear space, with the algebraic structure

- $(L_1 + L_2)x \equiv L_1 x + L_2 x, x \in X$
- $(\alpha L_1)x \equiv \alpha(L_1 x), x \in X$

and norm

$$\|L\|_{\mathcal{L}(X,Y)} \equiv \inf\{C \big| \|Lx\|_Y \leq C\|x\|_X, \forall x \in X\}.$$

This space will not generally be a Banach space, however. An additional property is needed,

Theorem 3.1.1. *If X and Y are normed linear spaces a and Y is complete, then $\mathcal{L}(X,Y)$ is a Banach space.*

3.1.2 Functionals, duality

An important example is $\mathcal{L}(X,\mathbb{R})$, the *dual space* of a normed linear space X, which consists of the bounded, real valued functions on X. It is customary to denoted this space by X^*, and to denote elements of this space with a superscript $*$, e.g. $x^* \in X^*$. The action of an element of the dual $x^*(x), x^* \in X^*, x \in X$ is often denoted using the duality pairing bracket notation $x^*(x) = \langle x, x^* \rangle$. The reasoning behind this notation becomes clearer in the discussion of Hilbert spaces. Since X^* is again a normed linear space, we can consider $(X^*)^*$, denoted X^{**}. For many spaces X and X^{**} are isometrically isomorphic, in which case X is called *reflexive.* These second dual spaces play some role in analysis, but higher order dual spaces (beyond two) are of limited importance.

The space $X^* = \mathcal{L}(X,\mathbb{R})$ is an abstract space in the sense that we have no characterization of its elements, of how it acts on elements of X to produce real numbers. It is often useful to form a *realization* of this space, a space of elements and a concrete description of the duality pairing. Usually there are many possible realiations, up to isometry. A number of common spaces and realizations is given in Tab. 3.1.

Space (X)	Dual (X^*)	Pairing $\langle x, x^* \rangle$
$x \in \mathbb{R}^n$	$y \in \mathbb{R}^n$	$\sum_{i=1}^n x_i y_i$
$f \in L_p(\Omega), 1 \leq p < \infty$	$g \in L_q(\Omega)$, $\frac{1}{p} + \frac{1}{q} = 1$	$\int_\Omega f(x)g(x)\,\mathrm{d}x$
$x \in l_p, 1 \leq p < \infty$	$y \in l_q$, $\frac{1}{p} + \frac{1}{q} = 1$	$\sum_{i=1}^\infty x_i y_i$
$f \in C[a,b]$	$w \in BV[a,b]$ (normalized)	$\int_a^b f(x)\,\mathrm{d}w(x)$
$x \in H$, a Hilbert space	$y \in H$	(x,y)
$m \in M \subset H$ (closed subspace)	$h^* \in H^*/M^\perp$	$h^*_{\mid M}(m)$

Table 3.1: Example of spaces and some canonical realizations of their dual spaces.

3.2 The Abstract Adjoint

With this groundwork, it is possible to define the adjoint operator in its most general setting. We define the adjoint of a map $L \in \mathcal{L}(X, Y)$ to be a map $L^* \in \mathcal{L}(Y^*, X^*)$,

$$\begin{array}{ccc} X^* & \xleftarrow{L^*} & Y^* \\ X & \xrightarrow{L} & Y, \end{array}$$

that satisfies the *bilinear identity*

$$L^* y^*(x) = y^*(Lx), \quad \text{i.e.} \quad \langle x, L^* y^* \rangle = \langle Lx, y^* \rangle,$$

for all $x \in X, y^* \in Y^*$. To prove the existence of this operator, it is sufficient to check that the operator defined by this identity is linear and bounded.

If we consider the operator $* : \mathcal{L}(X, Y) \to \mathcal{L}(Y^*, X^*)$ defined by $*(L) = L^*$, we obtain several properties of the adjoint,

- $(L_1 + L_2)^* = L_1^* + L_2^*$,
- $(\alpha L)^* = \alpha L^*$,
- $\|L^*\|_{\mathcal{L}(Y^*, X^*)} = \|L\|_{\mathcal{L}(X,Y)}$,
- $(L_1 L_2)^* = L_2^* L_1^*$, $(L_1 \in \mathcal{L}(X, Y), L_2 \in \mathcal{L}(Y, Z))$.

The first of these properties show that $*$ is an isometric isomorphism of $\mathcal{L}(X,Y)$ onto the range of $*$ (which is $\mathcal{L}(Y^*,X^*)$ when Y is reflexive).

3.2.1 The shift operators, an example

To demonstrate some of these concepts, we consider the Banach space l_1, of absolutely summable sequences in $\mathbb{R}$. We define the addition of two sequences $\{x_n\}$ and $\{y_n\}$ to be the sequence $\{x_n+y_n\}$, we define $\alpha\{x_n\} = \{\alpha x_n\}$, and the norm $\|\{x_n\}\|_{l_1} = \sum_{n=1}^{\infty}|x_n|$. A realization of the dual of l_1 is l_∞, the space of bounded sequences in $\mathbb{R}$ with $\|\{y_n^*\}\|_{l_\infty} = \sup_n |y_n^*|$, and the same algebraic operations. The duality pairing takes the form

$$\langle\{x_n\},\{y_n^*\}\rangle \equiv \sum_{n=1}^{\infty} x_n y_n^*,$$

for $\{x_n\}\in l_1, \{y_n^*\}\in l_\infty$.

We consider the "right shift" operator $T\in\mathcal{L}(l_1,l_1)$ defined by

$$T\{x_1,x_2,x_3,\dots\} = \{0,x_1,x_2,x_3,\dots\}.$$

It is easy to show T is linear and bounded, with $\|T\| = 1$.

The adjoint of T is a map from $l_\infty \to l_\infty$, which satisfies the bilinear identity,

$$\begin{aligned}\langle\{x_n\},T^*\{y_n^*\}\rangle &= \langle T\{x_n\},\{y_n^*\}\rangle \\ &= \langle\{0,x_1,x_2,\dots\},\{y_n^*\}\rangle \\ &= x_1y_2^* + x_2y_3^* + \dots \\ &= \langle\{x_1,x_2,\dots\},\{y_2,y_3,\dots\}\rangle,\end{aligned}$$

so that $T^*\{y_n^*\} = \{y_2,y_3,\dots\}$ is the "left shift" operator.

3.3 Hilbert spaces

We consider momentarily the special Banach space $\mathbb{R}^n$. The norm in this space $\|x\|$ is given by $\|x\| = \sqrt{x^\top x}$, so that for vectors $x, y \in \mathbb{R}^n$,

$$\begin{aligned}\|x+y\|^2 + \|x-y\|^2 &= (x+y)^\top(x+y) + (x-y)^\top(x-y) \\ &= \|x\|^2 + \|y\|^2 + x^\top y + y^\top x \\ &\quad + \|x\|^2 + \|y\|^2 - x^\top y - y^\top x \\ &= 2(\|x\|^2 + \|y\|^2).\end{aligned}$$

Geometrically, this says the sum of the square of the length of the diagonals in a parallelogram equals the sum of the squares of the lengths of the sides. For this reason this equality is called the *parallelogram equality.*

The norm in this case is induced by an *inner product*, which is a function $(\cdot, \cdot)$ from $X \times X \to \mathbb{R}$ with the properties

i) $(x+y, z) = (x, y) + (y, z)$,

ii) $(\alpha x, y) = \alpha(x, y)$,

iii) $(x, y) = \overline{(y, x)}$,

iv) $(x, x) \geq 0$, and $(x, x) = 0 \Leftrightarrow x = 0$.

The vector product $y^\top x$ in the case of $\mathbb{R}^n$ satisfies these properties. In general, for any linear space X with an inner product, the function $\|x\| = \sqrt{(x, x)}$ will be a norm.

Now consider the Banach space $\mathrm{C}[a, b]$, the continuous functions on the interval $[a, b]$, with the norm $\|g(t)\|_{\mathrm{C}[a,b]} = \sup_{s \in [a,b]} |g(s)|$. Take the

functions $f(t) = 1$, $g(t) = (t-a)/(b-a)$. Easily, $\|f\| = \|g\| = 1$, $\|f+g\| = \sup_{s\in[a,b]}\{1 + \frac{s-a}{b-a}\} = 2$, and $\|f-g\| = \sup_{s\in[a,b]}\{1 - \frac{s-a}{b-a}\} = 1$. Now

$$\|f+g\|^2 + \|f-g\|^2 = 5 \neq 4 = 2(\|f\|^2 + \|g\|^2),$$

and so the parallelogram equality does not hold in this Banach space.

Since the computation leading to the parallelogram equality follows for an arbitrary norm induced by an inner product, we conclude that the norm in $C[a,b]$ cannot come from an inner product.

Theorem 3.3.1. *The norm in a Banach space X comes from an inner product iff the norm satisfies the parallelogram equality.*

A complete normed linear space whose norm is induced by an inner product is called a *Hilbert space*, and the existence of the inner product gives the space more structure than a general Banach space. The existence of a parallelogram equality is the key difference between these two types of spaces.

A fundamental difference with regards to duality is given by the following,

Theorem 3.3.2. Riesz Representation *For every element x^* in the dual of a Hilbert space X, there is a unique element $z_{x^*} \in X$ such that*

$$\langle x, x^* \rangle = (x, z_{x^*})_X, \quad \textit{and} \quad \|z_{x^*}\|_X = \|x^*\|_{X^*},$$

for all $x \in X$.

This mapping $x^* \mapsto z_{x^*}$ is an isometric isomorphism of X and X^*, in the case that $\mathbb{F} = \mathbb{R}$ (when $\mathbb{F} = \mathbb{C}$, it is only a conjugate isometric isomorphism).

To form the adjoint between two Hilbert spaces X and Y, first let $\sigma : X \to X^*$ be so that $\sigma(u)(x) = (x,u)_X$, and $\tau : Y \to Y^*$ so $\tau(v)(y) = (y,v)_Y$ (the inverse of the isomorphisms provided by the Riesz Representation theorem). For an operator $L \in \mathcal{L}(X,Y)$, we form an operator $L^+ : Y \to X$ by $L^+ = \sigma^{-1}L^*\tau$,

$$\begin{array}{ccc} X^* & \xleftarrow{L^*} & Y^* \\ {\scriptstyle \sigma^{-1}}\downarrow & & \uparrow{\scriptstyle \tau} \\ X & \xleftarrow{L^+} & Y. \end{array}$$

We have for $x \in X, y \in Y$,

$$\begin{aligned} \big(x, L^+y\big)_X &= \big(x, \sigma^{-1}L^*\tau(y)\big)_X \\ &= (x,u)_X = \sigma(u)(x) \\ &= L^*\tau(y)(x) \\ &= \tau(y)(Lx) \\ &= (Lx,y)_Y, \end{aligned}$$

and L^+ is called the Hilbert space adjoint of L. Alternately, it is customary to identify X with X^*, suppressing this explicit use of the isomorphisms σ and τ, and write

$$(x, L^*y)_X = \langle x, L^*y\rangle = \langle Lx, y\rangle = (Lx,y)_Y.$$

Identifying X^* with X itself, we see that the bracket notation for the duality pairing is used to emphasize that duality for an arbitrary Banach space may be viewed as a generalization of the inner product for a Hilbert space. The element of the dual space usually (not always) occupies the second slot in pairing because in the case of a complex Hilbert space the function $(\cdot, x)$ is linear, but $(x, \cdot)$ is only conjugate linear.

3.3.1 Matrices, a Hilbert space example

Let $X = \mathbb{R}^m$, and $Y = \mathbb{R}^n$. As mentioned above, these are Hilbert spaces, with the Euclidian inner product. Setting $\boldsymbol{e}_i, i = 1, \ldots, m$ be the standard basis vectors for $\mathbb{R}^m$ (all zeros save a 1 in the i^{th} position) and $\boldsymbol{v}_j, j = 1, \ldots, n$ the standard basis for $\mathbb{R}^n$, for any $x = \sum_{i=1}^m x_i \boldsymbol{e}_i \in X$, and $x^* \in X^*$, we have

$$x^*(x) = \sum_{i=1}^m x_i x^*(\boldsymbol{e}_i) = z_{x^*}^\top x,$$

where $z_{x^*} = \sum_{i=1}^m x^*(\boldsymbol{e}_i)\boldsymbol{e}_i$. In this sense, $(\mathbb{R}^m)^* \cong \mathbb{R}^m$, with the isomorphism $x \mapsto x^\top[\cdot]$. Similary, $(\mathbb{R}^n)^* \cong \mathbb{R}^n$. An element $L \in \mathcal{L}(X, Y)$ may be represented as a $n \times m$ matrix $L = [L_{i,j}]$. Identifying the dual of $\mathbb{R}^m$ and $\mathbb{R}^n$ with themselves, since $L^* \in \mathcal{L}(Y, X)$, it may be represented as an $m \times n$ matrix, $L^* = [L^*_{i,j}]$. In this case the bilinear identity gives

$$\begin{aligned}
(x, L^* y)_X &= \left(x, \left[\sum_{i=1}^n T^*_{j,i} y_i \right]_j \right)_X \\
&= \sum_{j=1}^m \sum_{i=1}^n T^*_{j,i} y_i x_j \\
&= (Lx, y)_Y \\
&= \left(\left[\sum_{j=1}^m T_{i,j} x_j \right]_i, y \right)_Y \\
&= \sum_{i=1}^n \sum_{j=1}^m T_{i,j} x_j y_i.
\end{aligned}$$

We see that this is satisfied iff $T^*_{j,i} = T_{i,j}$, i.e. $T^* = T^\top$ as matrices. We note here that the adjoint operator is independent of the basis chosen, whereas the matrix transpose depends on the basis.

3.4 Distribution theory

In order to discuss the adjoint in the context of differential operators it is necessary to introduce distribution theory, and the distributional derivative.

3.4.1 Multi-index notation for derivatives

For what follows, it is convenient to have a simple notation for the partial derivatives of a function on $\mathbb{R}^n$. For $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_m) \in \mathbb{N}^m$, we let $|\alpha| = \alpha_1 + \cdots + \alpha_m$, and for $x \in \mathbb{R}^m$ we let $x^\alpha = x_1^{\alpha_1} x_2^{\alpha_2} \ldots x_m^{\alpha_m}$. We define the differential operators,

$$D^\alpha = (\partial/\partial x_1)^{\alpha_1} (\partial/\partial x_2)^{\alpha_2} \ldots (\partial/\partial x_m)^{\alpha^m}.$$

3.4.2 Functions as linear functionals

We typically regard a function $f(x) : \Omega \to \mathbb{R}$ as a collection of values, namely its point values at x in Ω. In distribution theory, we instead tabulate a function instead by samples of the type $\langle f, \phi \rangle = \int_\Omega f(x)\phi(x)\,\mathrm{d}x$, for ϕ in an admissible class of functions.

For example, if $\phi_0(x)$ is constant near a point x_0 and nearly zero away from this point, with $\int_\Omega \phi_0(x)\,\mathrm{d}x = 1$, we have $\langle f, \phi_0 \rangle \approx f(x_0)$, and we can recover point values of f. However, more generally tabulations are possible; for instance if $\Omega = (-\infty, \infty)$, the values $b_n = \langle f, (2/\pi)\sin nx \rangle$ give the Fourier coefficients of f, from which f may be recovered as

$$f(x) = \sum_{n=1}^{\infty} b_n \sin nx.$$

In this way we can embed the space of functions in a larger space of linear functionals.

3.4.3 Test functions, distributions

The space of admissible functions that we use is $D(\Omega)$, the space of *test functions*, which are infinitely differentiable and vanish outside some compact subset of Ω (possibly a different subset for each function). This space is a linear space with the usual algebraic operations.

We define a very strict notion of convergence on this space. For a sequence of functions $\{\phi_n\}_{n=1}^{\infty}$ in $D(\Omega)$ we say that $\phi_n(x) \to 0$ if

1. The supports of all ϕ_n are contained in a fixed bounded domain in Ω and

2. for each $\alpha \in \mathbb{N}^m, m = 1, 2, \ldots, D^\alpha \phi_n \to 0$ uniformly over all of Ω.

This defines a linear space, but not a normed linear space. The space has a topology, however, induced by a particular set of seminorms [26], so there is a notion of continuity for functionals on this space.

We define the set of *distributions*, $D'(\Omega)$ to be the dual of $D(\Omega)$, or the set of continuous linear functionals on $D(\Omega)$ (this notion of duality is slightly more general than that presented above). It is customary to denote the action of $f \in D'(\Omega)$ on an element $\phi \in D(\Omega)$ as $\langle f, \phi \rangle$ (note the functional here is in the opposite position when compared to the customary bracket notation for the duality pairing). Continuity is equivalent to $\langle f, \phi_n \rangle \to 0$ in $\mathbb{R}$ whenever $\phi_n \to 0$ in $D(\Omega)$.

The simplest kind of linear functional is that given by a *locally integrable* function $f(x)$ on Ω. We define

$$\langle f, \phi \rangle = \int_\Omega f(x)\phi(x)\,\mathrm{d}x,$$

which defines a continuous linear functional on $D(\Omega)$. In this way, we can view any locally integrable function as an element of $D'(\Omega)$, and so the space of distributions is sometimes referred to as the space of generalized functions.

A distribution generated by a locally integrable function is called a regular distribution, but there are many distributions which are not of this form. The δ distribution, with action $\langle \delta, \phi \rangle = \phi(0)$ is one example.

3.4.4 Distributional derivatives

We define the distributional derivative D^α of a distribution f to be the distribution whose action is

$$\langle D^\alpha f, \phi \rangle = (-1)^{|\alpha|} \langle f, D^\alpha \phi \rangle.$$

This definition is motivated by the fact that for a function $f \in C^1(\Omega)$, integration by parts yields

$$\langle D^\alpha f, \phi \rangle = \int_\Omega D^\alpha f(x) \phi(x) \, \mathrm{d}x = (-1)^{|\alpha|} \int_\Omega f(x) D^\alpha \phi(x) \, \mathrm{d}x,$$

which also shows that the distributional derivative agrees with the ordinary derivative for C^1 functions.

The space of distributions is a linear space, with the obvious choice of addition and scalar multiplication. Further, we can define the multiplication of two distributions ψf by $\langle \psi f, \phi \rangle = \langle f, \psi \phi \rangle$, but only when $\psi \in D(\Omega)$. If we define the *formal adjoint* of a differential operator $p(D) = \sum_{i=1}^k c_i(x) D^{\alpha_i}$ (for $c_i(x) \in D(\Omega)$) to be $p(D)^* = \sum_{i=1}^k (-1)^{|\alpha_i|} D^\alpha (c_i(x)[\cdot])$ (motivated by the $L_2(\Omega)$ inner product), we see that the action of a (linear) differential operator on a distribution f is obtained through the action of its formal

adjoint on ϕ;

$$\begin{aligned}\langle p(D)f(x),\phi(x)\rangle &= \sum_{i=1}^{k}\langle c_i(x)D^{\alpha_i}f(x),\phi(x)\rangle \\ &= \sum_{i=1}^{k}\langle D^{\alpha_i}f(x),c_i(x)\phi(x)\rangle \\ &= \sum_{i=1}^{k}(-1)^{|\alpha_i|}\langle f(x),D^{\alpha_i}\left(c_i(x)\phi(x)\right)\rangle \\ &= \langle f(x),p(D)^{*}\phi(x)\rangle.\end{aligned}$$

3.5 Sobolev spaces

The adjoint of a partial differential equation also requires the spaces

$$H^k(\Omega) = \{u \in L_2(\Omega) \big| D^\alpha u \in L_2(\Omega), 1 \leq |\alpha| \leq k\},$$

where the derivative $D^\alpha u$ is viewed first as a distributional derivative , and if there is an $L_2(\Omega)$ function that generates this distribution, we say $D^\alpha u$ is in $L_2(\Omega)$.

These spaces are Hilbert spaces, with inner product

$$(u,v)_{H^k} = \sum_{0\leq|\alpha|\leq k}(D^\alpha u, D^\alpha v)_{L_2},$$

where $D^{(0,0,\cdots,0)}u \equiv u$.

By the Riesz Representation theorem, these Hilbert spaces are isomorphic to their dual spaces. The dual space, however, is the abstract space of continuous linear functionals, which has more than one realization (via isometry). It turns out that another realization is more useful in the variational setting of a PDE.

3.6 A discussion of the adjoint for Poisson's problem

We discuss at length the adjoint in the context of Poisson's problem

$$\begin{cases} -\nabla^2 u = f, & \Omega, \\ u = 0, & \partial\Omega. \end{cases} \tag{3.1}$$

We say that a distribution u satisfies 3.1 if for all $\phi \in D(\Omega)$

$$\langle -\nabla^2 u, \phi \rangle = \langle f, \phi \rangle,$$

or equivalently

$$\langle u, -\nabla^2 \phi \rangle = \langle f, \phi \rangle, \tag{3.2}$$

since $-\nabla^2$ is its own formal adjoint.

A solution to this equation, if it exists, falls into one of the following categories:

1. If u is sufficiently differentiable so that the derivatives exist in the classical sense, then (3.1) is an identity, and u is called a *classical solution* (u also satisfies 3.2).

2. If u satisfies 3.2, but does not have the necessary classical derivatives, we say u is a *weak solution.*

3. If u is a singular distribution that satisfies 3.2, we say u is a *distributional solution.*

To discuss the adjoint within the abstract framework presented in the previous sections, we need to define the domain and range of $-\nabla^2$. It turns out that we want the domain to be larger than the space $C^2(\Omega)$, so that we require distribution theory to define these spaces. The operator is well defined on the entire space of distributions, and the range is again some

subset of the distributions. This is too general of a setting to be useful, however. Hence we try to narrow the domain and range to smaller subsets of the distributions.

It is practical to consider the data f in 3.1 to be in $L_2(\Omega)$, in which case the solution u would necessarily be in $H^2(\Omega)$. Further considerations, however, included non-homogenous boundary conditions [16] lead to the more common formulation where f is taken to be in the space

$$H^{-1}(\Omega) = \{f \in D'(\Omega) | f = f_0 + \sum_{i=1}^{n} \partial_{x_i} f_i, f_j \in L_2(\Omega), j = 0, \ldots, n\},$$

where the derivatives $\partial_{x_i} f_i$ are the distributional derivatives of f_i, viewed as an element of $D'(\Omega)$.

The natural setting for the domain of A is thus $H^1(\Omega)$. We build the boundary condition $u = 0$ into the space by narrowing the domain to a subset of $H^1(\Omega)$, $H_0^1(\Omega)$, the closure of $D(\Omega) \subset H^1(\Omega)$ with respect to the $H^1(\Omega)$ norm. Under some circumstances involving the dimension of the physical space Ω and the regularity of f, u may also be a classical solution.

This framework leads to a natural scale of spaces and embeddings,

$$D(\Omega) \hookrightarrow H_0^1(\Omega) \hookrightarrow L_2(\Omega) = (L_2(\Omega))^* \hookrightarrow (H_0^1(\Omega))^* \hookrightarrow D'(\Omega). \qquad (3.3)$$

Each of the embeddings left of the '=' are continuous injections. We identify L_2 with its own dual (Riesz), with the duality pairing given by the L_2 inner product, which agrees with the $D \times D'$ duality pairing when one of the functions is in $D(\Omega)$, and extends this pairing to all of L_2. The embeddings on the right are also continuous, and since the embeddings on the left are dense, the embeddings to the right are also injective, so that we may view each of these spaces as various subspaces of $D'(\Omega)$.

Choosing the domain of $A = -\nabla^2$ to be $H_0^1(\Omega)$, and realizing that $Au \in H^{-1}(\Omega) \subset D'(\Omega)$, we seek to find the adjoint A^* to complete the picture

$$\begin{array}{ccc} (H_0^1(\Omega))^* & \xleftarrow{A^*} & (H^{-1}(\Omega))^* \\ H_0^1(\Omega) & \xrightarrow{A} & H^{-1}(\Omega). \end{array} \tag{3.4}$$

Here the duals of H_1^0 and H^{-1} are abstract spaces. Therefore we choose a *realization* of each dual space, and describe the duality pairing. We utilize the scale of embeddings (3.3) to consider all of these spaces as subsets of $D'(\Omega)$, and we identify each dual space with a particular set of distributions.

In choosing the dual space and the duality pairing, we also characterize the adjoint A^*. We desire that A^* resembles its formal adjoint, which we obtain through integration by parts. For an arbitrary realization of the dual spaces, this is not possible and we may have no hope of characterizing A^*.

The appropriate realization of $(H_0^1(\Omega))^*$ is surprisingly simple, and turns out to be $H^{-1}(\Omega)$. For any $f \in H^{-1}(\Omega)$, choosing smooth functions $\phi_n \in D(\Omega)$ so that $\phi_n \to u$ (in $H_0^1(\Omega)$), define the linear functional T_f on $H_0^1(\Omega)$ by

$$T_f(u) = \lim_n \langle f_0 + \sum_{i=1}^n \partial_{x_i} f_i, \phi_n \rangle = \int_\Omega u f_0 - \sum_{i=1}^n f_i \partial_{x_i} u \, dx.$$

Note that the $H_0^1(\Omega)$ convergence of ϕ_n to u is exactly what is needed to make this functional well defined. It is easy to show that this functional is continuous with respect to the $H_0^1(\Omega)$ norm, so that under the identification $f \mapsto T_f$, $H^{-1}(\Omega) \hookrightarrow (H_0^1(\Omega))^*$.

On the other hand, for $T \in (H_0^1(\Omega))^*$, the Riesz Representation theorem guarantees a unique $u_T \in H_0^1(\Omega)$ so that for all $v \in H_0^1(\Omega)$,

$$\begin{aligned}(u_T, v)_{H^1} &= T(v) \\ &= \int_\Omega u_T v + \nabla u_T \cdot \nabla v \\ &= \lim_n \langle u_T - \nabla^2 u_T, \phi_n \rangle \\ &= T_f(v)\end{aligned}$$

where $D(\Omega) \ni \phi_n \underset{H_0^1}{\longrightarrow} v$ and $f = u_T - \nabla^2 u_T \in H^{-1}(\Omega)$, and so the map $f \mapsto T_f$ is a surjective mapping to the dual of $H_0^1(\Omega)$. We have, therefore,

Theorem 3.6.1. *A realization for the dual of $H_0^1(\Omega)$ is $H^{-1}(\Omega)$, with the duality pairing $\langle u, f \rangle_{H_0^1 \times H^{-1}} = T_f(u)$.*

This pairing has the properties that we want, namely when $f \in D(\Omega)$, this agrees with the pairing $D \times D'$. Further, if both u and f are in $L_2(\Omega)$, the pairing is exactly the $L_2(\Omega)$ inner product. For this reason, it is common to express this pairing as

$$\langle u, f \rangle_{H_0^1(\Omega) \times H^{-1}(\Omega)} = (u, f)_{L_2(\Omega)} = \int_\Omega u(x) f(x)\, \mathrm{d}x,$$

although this integral is not actually defined for all u, f.

Since $H^{-1}(\Omega)$ is a Hilbert space, it is reflexive, and so we may characterize its dual through $H_0^1(\Omega)$ simply by reversing the duality pairing, i.e.

$$\langle f, u \rangle_{H^{-1}(\Omega) \times (H^{-1}(\Omega))^*} \equiv \langle u, f \rangle_{H_0^1(\Omega) \times H^{-1}(\Omega)}.$$

We are now able to modify the diagram (3.4) by providing realizations of the dual spaces,

$$\begin{array}{ccc} H^{-1}(\Omega) & \xleftarrow{A^*} & H_0^1(\Omega) \\ H_0^1(\Omega) & \xrightarrow{A} & H^{-1}(\Omega), \end{array} \tag{3.5}$$

where the bilinear identity is now

$$\langle u, A^* v \rangle_{H_0^1 \times H^{-1}} = \langle Au, v \rangle_{H^{-1} \times H_0^1},$$

for all $u, v \in H_0^1(\Omega)$, or formally

$$\int_\Omega u\, A^* v \, \mathrm{d}x = \int_\Omega Au\, v \, \mathrm{d}x.$$

Now recalling that $A = -\nabla^2$, and taking ϕ_n so $\phi_n \underset{H_0^1}{\longrightarrow} v$,

$$\begin{aligned} \langle -\nabla^2 u, v \rangle_{H^{-1} \times H_0^1} &= \lim_n \langle -\nabla^2 u, \phi_n \rangle_{D' \times D} \\ &= \lim_n \int_\Omega \nabla u \cdot \nabla \phi_n \, \mathrm{d}x \\ &= \int_\Omega \nabla u \cdot \nabla v \, \mathrm{d}x \end{aligned}$$

On the other hand, if we assume that A^* is equal to its formal adjoint (which is also $-\nabla^2$), and take ψ_n so that $\psi_n \underset{H_0^1}{\longrightarrow} u$, we compute

$$\begin{aligned} \langle u, -\nabla^2 v \rangle_{H_0^1 \times H^{-1}} &= \lim_n \langle -\nabla^2 v, \psi_n \rangle_{D' \times D} \\ &= \lim_n \int_\Omega \nabla \psi_n \cdot \nabla v \, \mathrm{d}x \\ &= \int_\Omega \nabla u \cdot \nabla v \, \mathrm{d}x \end{aligned}$$

Therefore, we see that given these characterizations of the dual spaces, and duality pairings, A^* is indeed the formal adjoint of A, namely $A = A^* = -\nabla^2$.

3.7 Pivot spaces

The embedding

$$H_0^1 \hookrightarrow L_2 = (L_2)^* \hookrightarrow (H_0^1)^*$$

is a special cases of a standard trick for Hilbert spaces. In general, if a Hilbert space V is densely and continuously embedded in another Hilbert space H by injection j, then j^* embeds H^* densely and continuously in V^*. If we identify H with H^* using the Riesz representation, then we call H a *pivot space*, and we have

$$V \underset{j}{\rightarrow} H \cong H^* \underset{j^*}{\rightarrow} V^*,$$

where j^*j embeds V into V^* continuously and densely, and there is a unique bilinear form $(\cdot, \cdot)$ which extends the inner product on H to $H \times V$, $H^* \times V$, and $V^* \times V$. Indeed, unless $V = H$, this pairing between V^* and V is not the pairing given by the Riesz theorem.

3.8 Non-Dirichlet boundary conditions

It is difficult to attempt an in depth study of the Poisson problem for more general boundary conditions, where we seek the solution $u \in V$, for $H_0^1(\Omega) \subset V \subseteq H^1(\Omega)$. In most cases of interest, the test functions are not dense in V, so that the embedding of $V^* \hookrightarrow D'(\Omega)$ is not injective. Elements in the dual of V that act on the boundary are not distinguished by their action on test functions, and so we cannot describe the dual simply as a subset of $D'(\Omega)$. It is still possible to characterize this dual as a sum of elements from $H^{-1}(\Omega)$ and linear functionals that are "concentrated on the boundary" (see [1]).

A more standard approach is to use instead a "formal adjoint" A_0^* which satisfies the integration by parts formula

$$\int_\Omega Auv \, \mathrm{d}x = \int_\Omega u A_0^* v \, \mathrm{d}x$$

for sufficiently smooth u, v, vanishing on $\partial\Omega$, and then modify the bilinear identity to include the boundary term that results when using this simplified adjoint for $u, v \in V$. The resulting identity is called the *abstract Green's formula*, and plays a crucial role in solving partial differential equations. It is beyond the scope of this chapter to include the details, which are described fully in [1].

3.9 Time dependent problems

Many differential equations have the form

$$\begin{cases} \partial_t u + Lu = f, \\ \text{b.c. and i.c,} \end{cases}$$

where the operator $L \in \mathcal{L}(X, Y)$ for suitably chosen spaces X and Y, and has, therefore, adjoint $L^* \in \mathcal{L}(Y^*, X^*)$.

The natural setting for the solution u is a space of *Functions valued in a Banach Space*, i.e. mappings of the form $t \in [0, T] \mapsto X$. In many cases, it is sufficient to consider the space

$$L_2([0, T]; X)$$

of *measurable* [1] functions $u : [0, T] \to X$ such that

$$\|u\|^2_{L_2([0,T];X)} = \int_0^T \|u(t)\|^2_X \, \mathrm{d}t < \infty.$$

The dual of these spaces may be identified with $L_2([0, T]; X^*)$ (see [34]), where the duality pairing expressed by

$$\langle u, u^* \rangle = \int_0^T \langle u(t), u^*(t) \rangle_{X \times X^*} \, \mathrm{d}t.$$

---

[1] Measurablity here is complicated by the fact that u is valued in a Banach space, but essentially means we can define integrals in X such as $\int_0^T u(t)\,\mathrm{d}t$. See [16] or [34] for a more detailed discussion.

When X is a Hilbert space, $L_2([0,T];X)$ is again a Hilbert space, with inner product

$$(u,v)_{L_2([0,T];X)} = \int_0^T (u(t),v(t))_X \, \mathrm{d}t.$$

Writing

$$\partial_t u = -Lu + f,$$

and considering by the previous theory that (at least for a second order operator) $Lu \in X^*$, we naturally let $\partial_t u \in L_2([0,T];X^*)$. In fact, this is the framework in which general second order parabolic differential equations are solved ([16]).

In this setting, we may view the operator $P = (\partial_t + L)$, as a mapping from $L_2([0,T] : V) \to L_2([0,T];V')$, where we use V to denote a Hilbert space, and V' to denote its dual (standard PDE notation).

The bilinear identity is

$$\begin{aligned}\langle Pu, v\rangle &= \int_0^T \langle Pu(t), v(t)\rangle_{V'\times V} \, \mathrm{d}t \\ &= \int_0^T \langle \partial_t u(t), v(t)\rangle_{V'\times V} \, \mathrm{d}t + \int_0^T \langle Lu(t), v(t)\rangle_{V'\times V} \, \mathrm{d}t \\ &= \langle u, P^* v\rangle,\end{aligned}$$

where the (unlabelled) brackets represent the $L_2([0,T];V) \times L_2([0,T];V')$ duality pairing. We treat the first term formally, since it requires the theory of Banach valued distributions, and write

$$\int_0^T \langle \partial_t u(t), v(t)\rangle_{V'\times V} \, \mathrm{d}t = u(t)v(t)\big|_{t=0}^{t=T} + \int_0^T \langle u(t), -\partial_t v(t)\rangle_{V'\times V} \, \mathrm{d}t,$$

which may be justified in this extended distributional framework. The second term is handled by the previous theory. Assuming for the moment that either u or v (or both) disappear at $t = 0, T$, we have

$$\langle u, P^* v\rangle = \int_0^T \langle u(t), (-\partial_t + L^*)v(t)\rangle_{V'\times V} \, \mathrm{d}t,$$

so that the adjoint operator is $P^* = (-\partial_t + L^*)$. Hence this realization of the dual spaces and adjoint operator are

$$L_2([0,t];V') \xleftarrow{P^*} L_2([0,T];V)$$
$$L_2([0,T];V) \xrightarrow{P} L_2([0,T];V').$$

3.10 Nonlinear equations

Since the adjoint is only defined for a linear operator, when applying adjoint techniques to nonlinear equations, the simplest way to proceed is to linearize the equations around a solution of interest. For example, when solving

$$F(u) = b,$$

we might use a function U, close to u in some way (typically either a numerical approximation or a solution with a nearby parameter), and linearize the system around U. The linearized operator $F'(U)$ has an adjoint, $F'(U)^*$.

Often the goal is to compute a quantity such as (u, ψ), in which case we solve the linearized adjoint problem

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

obtaining the representation formula

$$(u, \psi) = (u, F'(U)^*\phi) = (F'(U)u, \phi) \approx (b - F(U), \phi)$$

where the approximation results due to dropping a higher order term (R, ϕ), where R is the remainder from the Taylor representation of F (which we assume exists). The quantity $b - F(U)$ is usually computable, and is often a type of residual.

An analogous development applies for the time dependent problem

$$\begin{cases} \partial_t u + F(u) = f, \\ \text{b.c. and i.c.} \end{cases}$$

The linearized adjoint problem is then

$$\begin{cases} -\partial_t \phi + F'(U)^* \phi = \psi, \\ \text{adj b.c. and i.c.} \end{cases}$$

3.11 The adjoint for numerical error estimation and adaptivity

We apply the adjoint to the case of solving a partial differential equation of the form

$$\begin{cases} Lu = f, & \Omega, \\ \text{suitable b.c.}, & \partial\Omega, \end{cases}$$

for u. Applying a numerical procedure, we obtain an approximate solution U. Often the goal of computing the solution u is to obtain a particular piece of information from u, called a *quantity of interest.* If the quantity is a linear functional, by the Riesz-Representation theorem it takes the integral form (u, ψ), for some data ψ. For example, in aerodynamics the goal may be to compute the lift or drag of an airfoil, which may be unaffected by many of the complex flow characteristics. Other examples are determining the heat applied to a particular boundary of an object, or the average value of some component of the solution.

To compute the numerical error in the quantity of interest, we solve an adjoint problem

$$\begin{cases} L^* \phi = \psi, & \Omega, \\ \text{adjoint b.c.}, & \partial\Omega, \end{cases}$$

for the *generalized Green's function* ϕ, to obtain the error representation

$$(u - U, \psi) = (u - U, L^* \phi) = (L(u - U), \phi) = (f - LU, \phi).$$

The quantity $f - LU$ is known as the *residual*, and measures how well the differential equation is satisfied by the approximate solution U. This quantity is computable, and we use its projection against ϕ to determine how well we must satisfy the differential equation to attain a desired level of accuracy in the quantity of interest.

3.11.1 Application to the Poisson problem

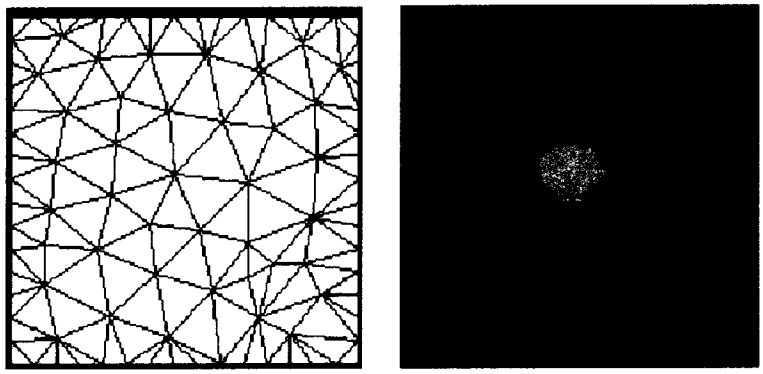

Figure 3.1: Original mesh and original solution, $a = 50$.

We return to the Poisson problem (3.1) in this context. Taking $\Omega = [-0.5, 0.5] \times [0.5, 0.5]$, we use the technique of *manufacturing solutions* (i.e., plug u into the equation and move everything that remains to the right) to obtain a solution $u(x, y) = \frac{a}{\pi} \exp(-a(x^2 + y^2))$, with

$$f(x) = \frac{4}{\pi} a^2 (1 - ax^2 - ay^2) \exp(-a(x^2 + y^2)).$$

For a large, u is nearly 0 on the boundary.

The *Galerkin method* for forming the approximate solution U is to form a subspace $V_h \subset H_0^1(\Omega)$ by triangulating the domain and letting V_h be the space of continuous linear functions on the nodes of the resulting triangulation $(\mathcal{T}_h)$. With $\phi_i, i = 1, \ldots, N$ the basis for this space, the Galerkin

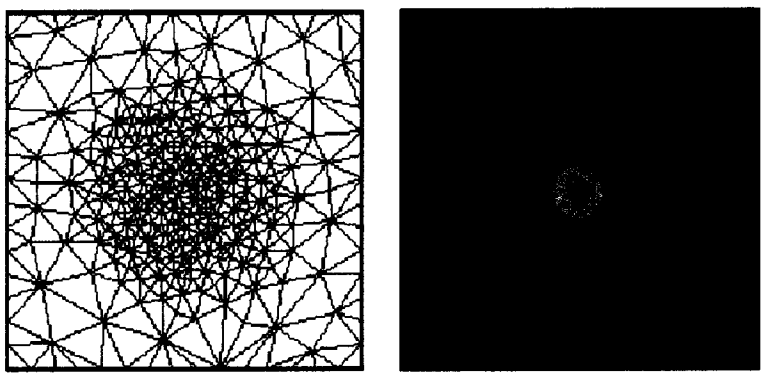

Figure 3.2: The final adaptive mesh, $a = 50$, with error $(u - U, \psi) = -0.0035$, $\psi = 1$. This choice of ψ gives the average error in the domain.

approximation satisfies the weak form of (3.1), $(\nabla U, \nabla \phi_i) = (f, \phi_i), i = 1, \ldots N$. Taking $U \in V_h$, we obtain the matrix equation $A\bar{U} = \bar{f}$, where $U = \sum_{i=1}^{N} \bar{U}_i \phi_i$, $\bar{f}_i = (f, \phi_i)$, and the matrix A has entries

$$A_{i,j} = (\nabla \phi_i, \nabla \phi_j) = \int_{\Omega} \nabla \phi_i \cdot \nabla \phi_j \, \mathrm{d}x.$$

To obtain the error estimate, given data ψ, we solve the adjoint problem

$$\begin{cases} -\nabla^2 \phi = \psi, & \Omega, \\ \phi = 0, & \partial\Omega, \end{cases}$$

for ϕ. We obtain the error representation

$$(u - U, \psi) = (u - U, -\nabla^2 \phi) = (-\nabla^2 u, \phi) - (-\nabla^2 U, \phi)$$

which we write as (after integrating by parts)

$$(f, \phi) - (\nabla U, \nabla \phi) = (f, \phi - \pi\phi) - (\nabla U, \nabla(\phi - \pi\phi)),$$

where $\pi\phi$ is a projection of ϕ into the space V_h, and the last equality is a property of the Galerkin method, called *Galerkin orthogonality.*

Using this error estimate, it is possible adaptively refine the mesh to achieve a desired error tolerance with an approximately minimal number of

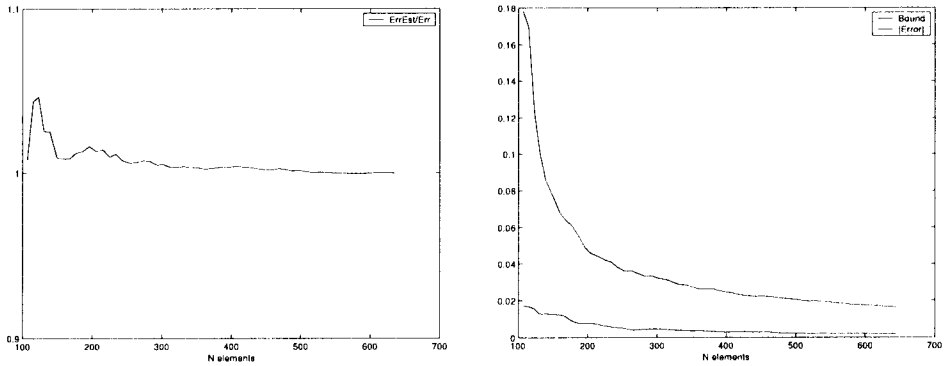

Figure 3.3: Left: *Accuracy of the error estimate, given by the ratio* $ErrEst/TrueErr$*, for a number of different mesh sizes.* Right: *Comparison of the error bound (sum of absolute errors) and the true error, showing how the bound greatly overestimates the error.*

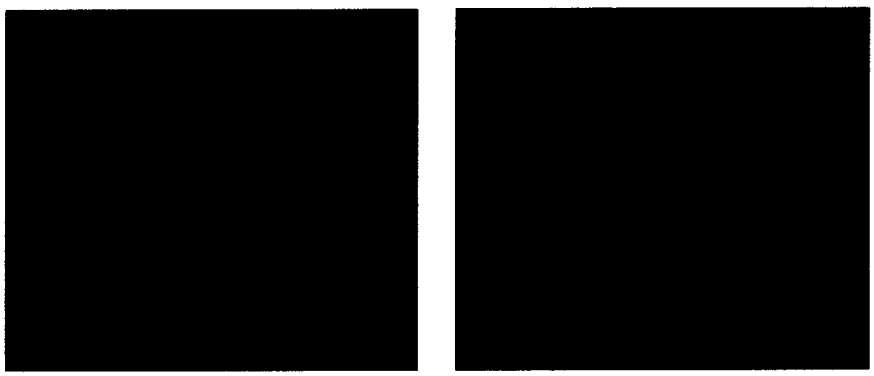

Figure 3.4: *Demonstration of the cancellation of the element errors throughout the domain (warmer colors mean greater error magnitude);* Left: *Positive error contributions.* Right: *Negative error contributions.*

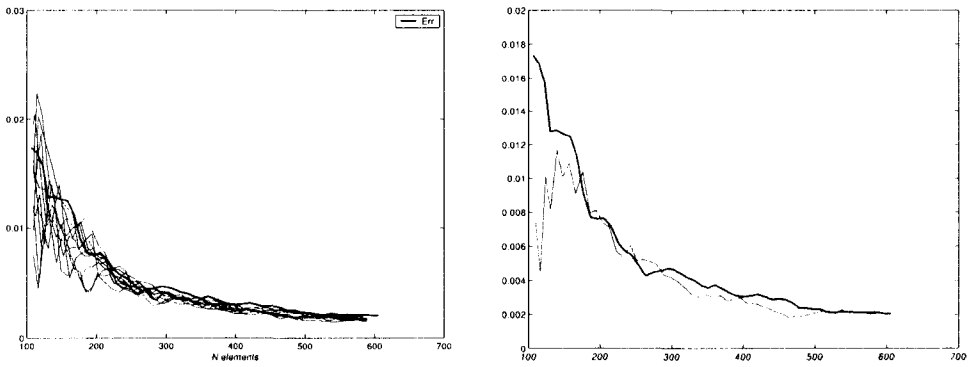

Figure 3.5: Left: *Error estimates based on refining the worst* 10 *elements (red), or selecting* 10 *elements randomly, distributed according to the element errors (green). Multiple realizations of the random algorithm are shown.* Right: *The random selection finds some truly "optimal meshes" where the error is extremely small with very little refinement. Depending on the desired TOL, it may be possible to stop the computation at one of these meshes, which is desirable since further refinement increases the error.*

triangles, or *elements.* We write the error representation as

$$(u - U, \psi) = \sum_{K \in \mathcal{T}_h} \int_\Omega (f, \phi - \pi\phi) - (\nabla U, \nabla(\phi - \pi\phi))\, \mathrm{d}x \tag{3.6}$$

$$\leq \sum_{K \in \mathcal{T}_h} \left| \int_\Omega (f, \phi - \pi\phi) - (\nabla U, \nabla(\phi - \pi\phi))\, \mathrm{d}x \right|, \tag{3.7}$$

and we refine some subset of the elements $K \in \mathcal{T}_h$ where the error is largest. The results of this procedure for this test problem are given in Fig. 3.1 and Fig. 3.2. The error estimates (3.6) are generally quite accurate, as demonstrated in Fig. 3.3, however, this figure also shows that the bounds (3.7) are generally much too large. There is a fundamental problem in adapting using (3.7), namely that we are *optimizing the mesh to reduce the error bound*, rather than optimizing the mesh to reduce the error. The bound removes all cancelation from one element to the next. This cancelation is demonstrated in Fig. 3.4, where the positive element contributions and the

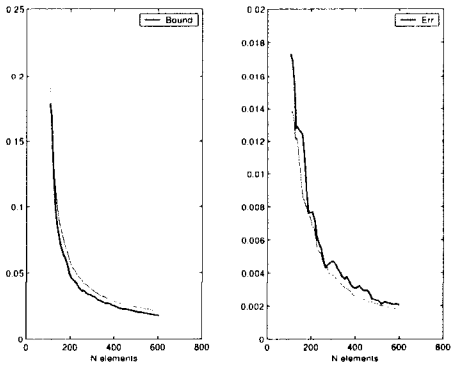

Figure 3.6: Left: *The error bound by the deterministic algorithm (red), and the mean behavior of the random algorithm (green).* Right: *The mean of the randomized error (green), which is better than the deterministic error (red).*

negative are plotting separately. The data here is $\psi = 1$, which gives the average error.

The mesh refinement algorithm can be viewed as a high dimensional search, where the search space consists of all possible paths of refinement. Experimentation shows that there are often "optimal meshes" on which the error is extremely small and yet the number of elements is also very small. Adaptivity based on the bound usually fails to find these meshes, but by reducing the bound it guarantees that a mesh is eventually found on which the error is acceptably small. The resulting mesh usually has far fewer elements than a mesh refined uniformly, but it is not "optimal" in the sense that there exist meshes with far fewer elements that have smaller errors.

To demonstrate this, we plot in Fig. 3.5 the true errors for the problem examined above against the number of elements for two methods of refinement. For the first method we refine the worst 10 elements at each level, and for the second method we refine randomly, selecting elements distributed according to the absolute value of the error estimates. For this

particular problem, the mean behavior of the randomly adaptive algorithm outperforms the deterministic algorithm (selecting the 10 worst points), as shown in Fig. 3.6.

3.12 The Boussinesq adjoint

To end this chapter we present the adjoint for a more complicated system, namely the Boussinesq system. This system of equations describes the evolution of a heated fluid, incorporating the forcing effects of thermal buoyancy.

We consider first the steady Boussinesq operator ($B : L_2(\Omega)^4 \rightarrow L_2(\Omega)^4$) defined as

$$B(\boldsymbol{u}, p, \theta) = \begin{cases} \boldsymbol{u} \cdot \nabla \boldsymbol{u} - \sqrt{\frac{Pr}{Ra}} \nabla^2 \boldsymbol{u} + \nabla p - \hat{j}\theta, \\ \nabla \cdot \boldsymbol{u}, \\ \boldsymbol{u} \cdot \nabla \theta - \frac{1}{\sqrt{Ra\, Pr}} \nabla^2 \theta, \end{cases}$$

where $\boldsymbol{u} = (u_1, u_2)^\top$ is the velocity field, and θ is the normalized temperature deviation $\theta = (T - T_r)/\Delta T$.

We are interested in two systems, first the steady Boussinesq equations

$$\begin{cases} B(\boldsymbol{u}, p, \theta) = \mathbf{0}, \\ \text{Boundary conditions}, \end{cases}$$

and the nonsteady Boussinesq flow,

$$\begin{cases} \begin{bmatrix} \partial_t \boldsymbol{u} \\ 0 \\ \partial_t \theta \end{bmatrix} + B(\boldsymbol{u}, p, \theta) = \mathbf{0}, \\ \text{Boundary conditions}, \\ \text{Initial conditions}. \end{cases}$$

To construct the adjoint, we first linearize this operator at some triple $(\boldsymbol{U}, P, \Theta)$ by considering

$$B(\boldsymbol{U}+\boldsymbol{u}, P+p, \Theta+\theta) - B(\boldsymbol{U}, P, \Theta) = \begin{cases} \boldsymbol{U}\cdot\nabla\boldsymbol{u} + \boldsymbol{u}\cdot\nabla\boldsymbol{U} + \boldsymbol{u}\cdot\nabla\boldsymbol{u} - \sqrt{\frac{Pr}{Ra}}\nabla^2\boldsymbol{u} + \nabla p - \hat{j}\theta \\ \nabla\cdot\boldsymbol{u} \\ \boldsymbol{U}\cdot\nabla\theta + \boldsymbol{u}\cdot\nabla\Theta + \boldsymbol{u}\cdot\nabla\theta - \frac{1}{\sqrt{Ra\,Pr}}\nabla^2\theta. \end{cases}$$

Dropping all terms that include products of the perturbations, we conclude (formally) that

$$B'(\boldsymbol{U}, P, \Theta)\begin{bmatrix}\boldsymbol{u}\\ p\\ \theta\end{bmatrix} = \begin{cases} \boldsymbol{U}\cdot\nabla\boldsymbol{u} + \boldsymbol{u}\cdot\nabla\boldsymbol{U} - \sqrt{\frac{Pr}{Ra}}\nabla^2\boldsymbol{u} + \nabla p - \hat{j}\theta \\ \nabla\cdot\boldsymbol{u} \\ \boldsymbol{U}\cdot\nabla\theta + \boldsymbol{u}\cdot\nabla\Theta - \frac{1}{\sqrt{Ra\,Pr}}\nabla^2\theta. \end{cases}$$

To study the effect of perturbations of the linear operator B (numerical discretization, perturbed parameters, etc...), we find the adjoint of $B'(\boldsymbol{U}, P, \Theta)$, which is a linear operator $B'(\boldsymbol{U}, P, \Theta)^*$ that satisfies

$$\left(B'(\boldsymbol{U}, P, \Theta)\begin{bmatrix}\boldsymbol{u}\\ p\\ \theta\end{bmatrix}, \begin{bmatrix}\boldsymbol{v}\\ q\\ \tau\end{bmatrix}\right)_{L_2(\Omega)^4} = \left(\begin{bmatrix}\boldsymbol{u}\\ p\\ \theta\end{bmatrix}, B'(\boldsymbol{U}, P, \Theta)^*\begin{bmatrix}\boldsymbol{v}\\ q\\ \tau\end{bmatrix}\right)_{L_2(\Omega)^4}$$

for all suitable $\boldsymbol{u}, \boldsymbol{v}, p, q, \theta, \tau$.

The inner product in $L_2(\Omega)^4$ is the sum of four scalar L_2 products (one for each component). Hence to proceed, rather than trying to compute the entire adjoint at once, we attack the problem term by term, unraveling all operations from the $\boldsymbol{u}, p, \theta$, and moving these onto $\boldsymbol{v}, q, \tau$. We then collect all terms involving a product with $\boldsymbol{u}$ to find the first two components of the adjoint, all terms involving p for the second, and all paired with θ to find the last component of the adjoint.

First,

$$(\boldsymbol{u}\cdot\nabla\boldsymbol{U}, \boldsymbol{v}) = (\mathrm{D}\boldsymbol{U}\boldsymbol{u}, \boldsymbol{v}) = (\boldsymbol{u}, \mathrm{D}\boldsymbol{U}^{\top}\boldsymbol{v}) = \boxed{(\boldsymbol{u}, \nabla\boldsymbol{U}\cdot\boldsymbol{v})}$$

where $D\boldsymbol{U}$ is the jacobian matrix of the vector $\boldsymbol{U}$ with respect to the spatial variables x and y, and $\nabla \boldsymbol{U} = [\nabla U_1 \nabla U_2]$ (so that the $\cdot$ represents matrix multiplication, i.e. $\boldsymbol{u} \cdot \nabla \boldsymbol{U} \equiv \boldsymbol{u}^\top [\nabla U_1 \nabla U_2]$ and $\nabla \boldsymbol{U} \cdot \boldsymbol{v} \equiv [\nabla U_1 \nabla U_2]\boldsymbol{v}$). Similarly, $(\boldsymbol{u} \cdot \nabla \Theta, \tau) = (u_1 \partial_x \Theta, \tau) + (u_2 \partial_y \Theta, \tau) = \boxed{(\boldsymbol{u}, \tau \nabla \Theta)}$.

The bouyancy term $(\hat{j}\theta, \boldsymbol{v}) = (0, v_1) + (\theta, v_2) = (\theta, v_2) = \boxed{(\theta, \hat{j} \cdot \boldsymbol{v})}$, the viscosity term

$$(-\sqrt{\frac{Pr}{Ra}} \nabla^2 \boldsymbol{u}, \boldsymbol{v}) = \boxed{(\boldsymbol{u}, -\sqrt{\frac{Pr}{Ra}} \nabla^2 \boldsymbol{v})}$$

and the temperature diffusion term

$$(-\frac{1}{\sqrt{Pr\, Ra}} \nabla^2 \theta, \tau) = \boxed{(\theta, -\frac{1}{\sqrt{Ra\, Pr}} \nabla^2 \tau)}$$

are easily dealt with. Here we assume the boundary terms vanish when integrating by parts. This may impose boundary conditions on the adjoint problem, but we ignore this issue until later.

The pressure and divergence terms swap roles;

$$(\nabla p, \boldsymbol{v}) = (\partial_x p, v_1) + (\partial_y p, v_2) = \boxed{(p, -\nabla \cdot \boldsymbol{v})},$$

whereas

$$(\nabla \cdot \boldsymbol{u}, q) = (\partial_x u_1, q) + (\partial_y u_2, q) = \boxed{(\boldsymbol{u}, -\nabla q)}.$$

The most difficult term is

$$\begin{aligned}
(\boldsymbol{U} \cdot \nabla \boldsymbol{u}, \boldsymbol{v}) &= (U_1 \partial_x u_1 + U_2 \partial_y u_1, v_1) + (U_1 \partial_x u_2 + U_2 \partial_y u_2, v_2) \\
&= -\big(u_1, \partial_x (U_1 v_1) + \partial_y (U_2 v_1)\big) - \big(u_2, \partial_x (U_1 v_2) + \partial_y (U_2 v_2)\big) \\
&= -\big(u_1, v_1 \partial_x U_1 + U_1 \partial_x v_1 + v_1 \partial_y U_2 + U_2 \partial_y v_1\big) \\
&\quad - \big(u_2, v_2 \partial_x U_1 + U_1 \partial_x v_2 + v_2 \partial_y U_2 + U_2 \partial_y v_2\big) \\
&= -\big(u_1, v_1 (\nabla \cdot \boldsymbol{U})\big) - \big(u_1, U_1 \partial_x v_1 + U_2 \partial_y v_1\big) \\
&\quad - \big(u_2, v_2 (\nabla \cdot \boldsymbol{U})\big) - \big(u_2, U_1 \partial_x v_2 + U_2 \partial_y v_2\big) \\
&= \boxed{\big(\boldsymbol{u}, -(\nabla \cdot \boldsymbol{U}) \boldsymbol{v} - \boldsymbol{U} \cdot \nabla \boldsymbol{v}\big)},
\end{aligned}$$

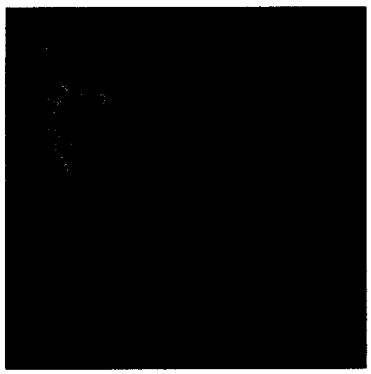

Figure 3.7: A typical Boussinesq flow, with heat applied to the left, cooling applied to the right. All other boundaries are adiabatic.

and the second most difficult is

$$\begin{aligned}(\boldsymbol{U}\cdot\nabla\theta,\tau) = (U_1\partial_x\theta,\tau) + (U_2\partial_y\theta,\tau) &= -\big(\theta,\partial_x(U_1\tau)+\partial_y(U_2\tau)\big)\\ &= -\big(\theta,\tau\partial_x U_1 + U_1\partial_x\tau + \tau\partial_y U_2 + U_2\partial_y\tau\big)\\ &= \boxed{\big(\theta,-(\nabla\cdot\boldsymbol{U})\tau - \boldsymbol{U}\cdot\nabla\tau\big)}\end{aligned}$$

Collecting terms in $\boldsymbol{u}, p, \theta$, we find that $B'(\boldsymbol{U},P,\Theta)^*\begin{bmatrix}\boldsymbol{v}\\ q\\ \tau\end{bmatrix} =$

$$\begin{cases}-(\nabla\cdot\boldsymbol{U})\boldsymbol{v} - \boldsymbol{U}\cdot\nabla\boldsymbol{v} + \nabla\boldsymbol{U}\cdot\boldsymbol{v} + \tau\nabla\Theta - \sqrt{\frac{Pr}{Ra}}\nabla^2\boldsymbol{v} - \nabla q\\ -\nabla\cdot\boldsymbol{v}\\ -(\nabla\cdot\boldsymbol{U})\tau - \boldsymbol{U}\cdot\nabla\tau - \frac{1}{\sqrt{Ra\,Pr}}\nabla^2\tau + \hat{j}\cdot\boldsymbol{v}\end{cases}$$

3.12.1 Application to Numerical Error

To use this adjoint for numerical error estimation, we let $\boldsymbol{U}, P, \Theta$ represent a numerical approximation. We assume a Taylor expansion for B,

$$\text{Residual} = B(\boldsymbol{u},p,\theta) - B(\boldsymbol{U},P,\Theta) = B'(\boldsymbol{U},P,\Theta)\begin{bmatrix}\boldsymbol{u}-\boldsymbol{U}\\ p-P\\ \theta-\Theta\end{bmatrix} + \text{Small},$$

which we expect to be true when $\boldsymbol{u}, p, \theta$ are close to $\boldsymbol{U}, P, \Theta$. We pose the adjoint problem

$$B'(\boldsymbol{U},P,\Theta)^*\begin{bmatrix}\boldsymbol{\phi_u}\\ \phi_p\\ \phi_\theta\end{bmatrix} = \begin{bmatrix}\boldsymbol{\psi_u}\\ \psi_p\\ \psi_\theta\end{bmatrix},$$

obtaining the error representation

$$\left(\begin{bmatrix} u-U \\ p-P \\ \theta-\Theta \end{bmatrix}, \begin{bmatrix} \psi_u \\ \psi_p \\ \psi_\theta \end{bmatrix} \right) = \left(\begin{bmatrix} u-U \\ p-P \\ \theta-\Theta \end{bmatrix}, B'(U,P,\Theta)^* \begin{bmatrix} \phi_u \\ \phi_p \\ \phi_\theta \end{bmatrix} \right)$$
$$= \left(B'(U,P,\Theta) \begin{bmatrix} u-U \\ p-P \\ \theta-\Theta \end{bmatrix}, \begin{bmatrix} \phi_u \\ \phi_p \\ \phi_\theta \end{bmatrix} \right) \approx \left(\text{Residual}, \begin{bmatrix} \phi_u \\ \phi_p \\ \phi_\theta \end{bmatrix} \right).$$

Chapter 4

A FINITE DIMENSIONAL EXAMPLE

To introduce the methods which follow, and to motivate the chapter on perturbation results below, we demonstrate our technique for solving the problem posed in the introduction in the simple case of a finite dimensional system of nonlinear equations.

4.1 A New Approach: A Finite Dimensional Example

We first consider the problem of solving the finite dimensional nonlinear system of equations,

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

for $\boldsymbol{x} \in \mathbb{R}^n$, where the parameter $\boldsymbol{\lambda}$ is a random vector in $\mathbb{R}^d$ and $\boldsymbol{f} : \mathbb{R}^{n+d} \to \mathbb{R}^n$ is smooth in both variables. In practice, the motivation for solving a nonlinear equation is often to compute a specific piece of information, or a *quantity of interest*, involving the solution. In many cases, this information can be represented as a linear functional of the solution, e.g., a particular component of $\boldsymbol{x}$ or the average of all the components. A functional value is a relatively low-dimensional piece of information and is easier to compute accurately than the entire solution, in general. By the Riesz representation theorem, there is a vector $\boldsymbol{\psi} \in \mathbb{R}^n$ such that $\langle \boldsymbol{x}, \boldsymbol{\psi} \rangle$ yields the quantity of

interest, where $\langle\ ,\ \rangle$ is the Euclidean inner product. For example, $\boldsymbol{\psi} = (1, 0, \cdots, 0)^\top$ yields the first component while $\boldsymbol{\psi} = (1, 1, \cdots, 1)^\top/n$ yields the average of the components.

We begin by assuming that $\boldsymbol{\lambda}$ is distributed closely about some reference value $\bar{\boldsymbol{\mu}}$ (an assumption that is relaxed later), and we solve the deterministic problem

$$\boldsymbol{f}(\boldsymbol{y}; \bar{\boldsymbol{\mu}}) = \boldsymbol{b}$$

for $\boldsymbol{y}$. To estimate the quantity of interest corresponding to $\boldsymbol{\psi}$, we use the *generalized Green's vector* $\boldsymbol{\phi}$ that solves the adjoint to the linearized problem,

$$A^T \boldsymbol{\phi} = \boldsymbol{\psi},$$

where $A = D_x \boldsymbol{f}(\boldsymbol{y}, \bar{\boldsymbol{\mu}})$. For example, if $\boldsymbol{\psi} = (1, 0, \cdots, 0)^\top$, then $\boldsymbol{\phi}$ is a Green's vector directly analogous to the Green's function in differential equations. The standard variational argument yields

$$\langle \boldsymbol{x}, \boldsymbol{\psi} \rangle = \langle \boldsymbol{x}, A^T \boldsymbol{\phi} \rangle = \langle A\boldsymbol{x}, \boldsymbol{\phi} \rangle,$$

which is the analog of the standard representation formula for the Green's function. Computing a Taylor expansion of $\boldsymbol{f}$ around $(\boldsymbol{y}, \bar{\boldsymbol{\mu}})$, we obtain

$$\boldsymbol{f}(\boldsymbol{x}; \boldsymbol{\lambda}) = \boldsymbol{f}(\boldsymbol{y}; \bar{\boldsymbol{\mu}}) + D_x \boldsymbol{f}(\boldsymbol{y}; \bar{\boldsymbol{\mu}})(\boldsymbol{x} - \boldsymbol{y}) + D_\lambda \boldsymbol{f}(\boldsymbol{y}; \bar{\boldsymbol{\mu}})(\boldsymbol{\lambda} - \bar{\boldsymbol{\mu}}) + \boldsymbol{R},$$

where $\boldsymbol{R}$ is a high order remainder. From this, we see that

$$\begin{aligned} \langle A\boldsymbol{x}, \boldsymbol{\phi} \rangle &= \big\langle \boldsymbol{f}(\boldsymbol{x}; \boldsymbol{\lambda}) - \boldsymbol{f}(\boldsymbol{y}; \bar{\boldsymbol{\mu}}) + A\boldsymbol{y} - D_\lambda \boldsymbol{f}(\boldsymbol{y}; \bar{\boldsymbol{\mu}})(\boldsymbol{\lambda} - \bar{\boldsymbol{\mu}}) - \boldsymbol{R}, \boldsymbol{\phi} \big\rangle \\ &= \langle A\boldsymbol{y}, \boldsymbol{\phi} \rangle - \big\langle D_\lambda \boldsymbol{f}(\boldsymbol{y}; \bar{\boldsymbol{\mu}})(\boldsymbol{\lambda} - \bar{\boldsymbol{\mu}}), \boldsymbol{\phi} \big\rangle - \langle \boldsymbol{R}, \boldsymbol{\phi} \rangle, \end{aligned}$$

and so

$$\langle \boldsymbol{x}, \boldsymbol{\psi} \rangle = \langle \boldsymbol{y}, \boldsymbol{\psi} \rangle - \big\langle D_\lambda \boldsymbol{f}(\boldsymbol{y}; \bar{\boldsymbol{\mu}})(\boldsymbol{\lambda} - \bar{\boldsymbol{\mu}}), \boldsymbol{\phi} \big\rangle - \langle \boldsymbol{R}, \boldsymbol{\phi} \rangle. \tag{4.2}$$

Neglecting the remainder term, we obtain an approximation for the quantity of interest corresponding to the parameter value $\boldsymbol{\lambda}$ in terms of the quantity of interest at the reference parameter value $\boldsymbol{\mu}$ plus an expression involving the change in $\boldsymbol{f}$ due to the parameter variation and the generalized Green's vector $\boldsymbol{\phi}$ corresponding to the quantity of interest. Taking the inner product with the generalized Green's vector $\boldsymbol{\phi}$, which we can view as a discrete convolution, translates the variations in the parameters into variations in the solution.

If we think of $\boldsymbol{\lambda} - \boldsymbol{\mu}$ as a random vector associated with some distribution, the expression

$$-\left\langle D_{\lambda}\boldsymbol{f}(\boldsymbol{y};\bar{\boldsymbol{\mu}})(\boldsymbol{\lambda}-\bar{\boldsymbol{\mu}}),\boldsymbol{\phi}\right\rangle$$

yields a new random variable associated with a new distribution that is added to the quantity of interest $\langle \boldsymbol{y}, \boldsymbol{\psi}\rangle$ at the reference value. This new random variable is a linear approximation to the true random variable near that point. We call using (4.2) in this way the *Higher Order Parameter Sampling* Method or *HOPS*.

As a first application, we consider

$$\begin{cases} \lambda_1 x_1^2 + x_2^2 &= 1 \\ x_1^2 - \lambda_2 x_2^2 &= 1, \end{cases}$$

where $\lambda_1, \lambda_2 > 0$ are the parameters. Solutions $\boldsymbol{x} = (x_1, x_2)$ are intersections of the hyperbola and the ellipse. We concentrate on the solution in the first quadrant.

We take λ_1 and λ_2 to be independent with λ_1 uniformly distributed on $\mu_1 \pm 0.1$ and λ_2 normally distributed $N(\mu_2, 0.1)$ for some fixed (μ_1, μ_2). We choose the dual data $\boldsymbol{\psi} = (0,1)^{\top}$ so that $\langle \boldsymbol{x}, \boldsymbol{\psi}\rangle = x_2$.

We first use a Monte-Carlo computation by solving the system for $n = 10000$ points drawn from the distribution of (λ_1, λ_2). Next, we use HOPS by calculating $\boldsymbol{y} = (\tilde{y}_1, \tilde{y}_2)$ at the mean value $(\mu_1, \mu_2) = (0.5, 1)$, numerically solving for the generalized Green's vector, and approximating

$$x_2 = \langle \boldsymbol{x}, \boldsymbol{\psi} \rangle \approx y_2 - \langle D_{\lambda} \boldsymbol{f}(\boldsymbol{y}; \bar{\boldsymbol{\mu}})(\boldsymbol{\lambda} - \bar{\boldsymbol{\mu}}), \boldsymbol{\phi} \rangle. \tag{4.3}$$

To compare the results, we compute the value of the HOPS approximation at the 10000 points used in the Monte-Carlo computation and plot the resulting histograms in Fig. 4.1. (See Sec. A.5 for details on producing the plots of densities shown in this paper. Comparing plots of densities provides a visually pleasing measure of accuracy, but this may not be the best measure of error in the computed information. We discuss ways to quantify the error in Sec. 7.1.) The savings in computational effort are extreme since HOPS requires only one solution of the nonlinear system, one solution to the linear adjoint problem, and then a vector dot product for each evaluation of the HOPS model. The Monte-Carlo approach, on the other hand, requires the solution to the nonlinear system for 10000 points. The results of a Monte-Carlo run for 1000 points are also included to demonstrate the increased order of accuracy of HOPS.

In general, linearization around a single point is insufficient to describe the response of the system to variations throughout the parameter space. In this example, near $(\mu_1, \mu_2) = (0.89, 1)$ the solution is more sensitive to variations in the parameter. We can see the effect in the degraded accuracy of a HOPS approximation at the reference value, see Fig. 4.1. A small Monte-Carlo computation with 1000 simulations appears more accurate than the one point linear approximation. In Fig. 4.2, we plot the norm of the generalized Green's vector against values of λ_1. The increase in the norm as λ_1

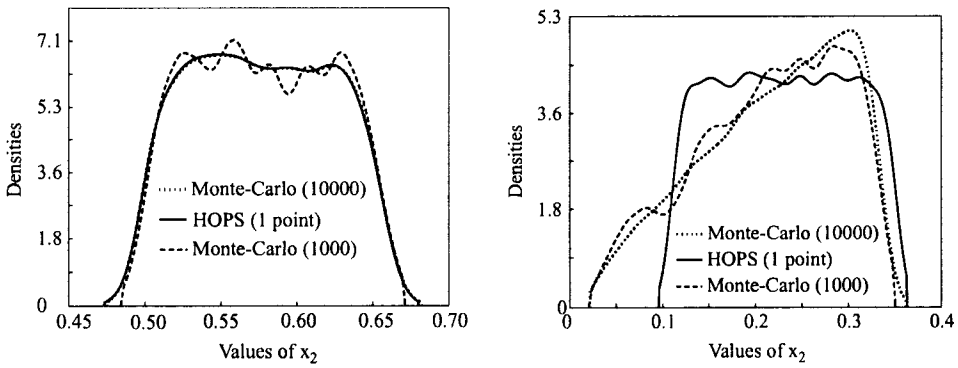

Figure 4.1: Comparisons of estimated densities for Monte-Carlo computations using 10000 and 1000 points along with HOPS one point linear approximations. On the left, the parameter values are $\mu_1 = 0.5, \mu_2 = 1$. On the right, the parameter values are $\mu_1 = 0.89, \mu_2 = 1$

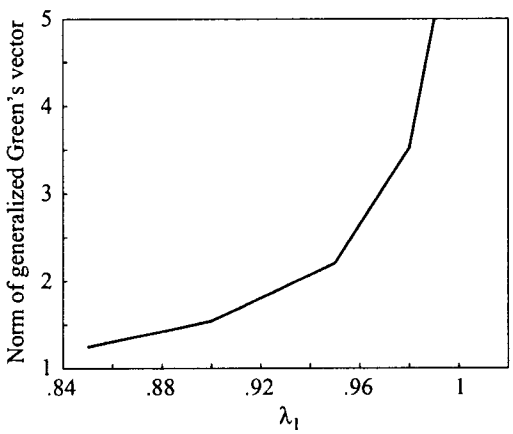

Figure 4.2: The norm of the generalized Green's vector, $|\phi|$, for λ_1 near 1.

approaches 1 indicates the increase in sensitivity of the solution. To address this, we compute HOPS approximations at five points chosen uniformly in the parameter set and combine the results to form a piecewise linear approximation (which we also call a HOPS approximation). The partition for the approximation is taken to be uniformly sized sub-intervals centered at the sample points. This yields an approximation with accuracy comparable to the small Monte-Carlo approximation, see Fig. 4.3. Counting both the linear adjoint and nonlinear solutions, we have solved only 10 problems, whereas the small Monte-Carlo computation uses 1000 solutions.

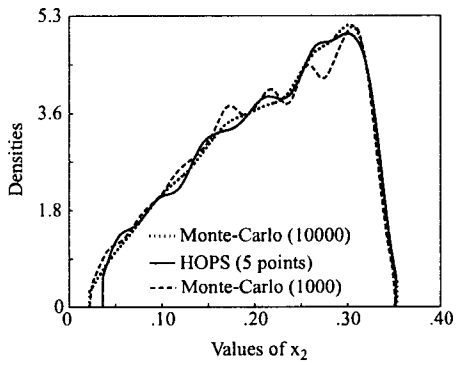

Figure 4.3: Comparison of estimated densities for Monte-Carlo computations using 10000 and 1000 points along with a HOPS approximation using five points in a sensitive region centered at $\mu_1 = 0.89, \mu_2 = 1$.

Chapter 5

PERTURBATION RESULTS

As a powerful application of the adjoint defined above, we develop a perturbation result for ordinary differential equations and for a system of Reaction Diffusion equations. These results form the basis of the HOPS, RAPS, and FAPS methods that follow.

5.1 Representation for ODE's

In the case of time dependent ODE's, we consider the problem of determining the effects of variations in parameters and data on a quantity of interest computed from the solution of the initial value problem

$$\begin{cases} \dot{\boldsymbol{x}}(t;\boldsymbol{\lambda}) = \boldsymbol{f}(\boldsymbol{x}(t;\boldsymbol{\lambda});\boldsymbol{\lambda}_1), \quad t > 0, \\ \boldsymbol{x}(0;\boldsymbol{\lambda}) = \boldsymbol{\lambda}_0 \end{cases} \tag{5.1}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ and $\boldsymbol{f} : \mathbb{R}^{n+p} \to \mathbb{R}^n$. The parameter $\boldsymbol{\lambda} = (\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_0)^\top$ is in $\mathbb{R}^d$ with $d = p + n$, where $\boldsymbol{\lambda}_1 \in \mathbb{R}^p$ represents parameters in the model $\boldsymbol{f}$ and $\boldsymbol{\lambda}_0 \in \mathbb{R}^n$ represents the initial conditions (which we also consider as parameters).

We wish to quantify how variation in $\boldsymbol{\lambda}$ affects the solution to (5.1). We consider $\boldsymbol{\lambda} = \boldsymbol{\lambda}(\omega)$ as a random vector on a probability space $(\Omega, \mathcal{B}, P)$, so that $\omega \in \Omega$. Under the standard Lipschitz assumption on $\boldsymbol{f}$ (see Theorem 5.1.2), $\boldsymbol{x}(t;\boldsymbol{\lambda})$ is continuous in the parameter $\boldsymbol{\lambda}$ and therefore also a

random vector of ω. The set of trajectories indexed by the possible values of ω may be viewed as a collection of random vectors $\boldsymbol{x}(t;\omega) = \boldsymbol{x}(t;\boldsymbol{\lambda}(\omega))$, indexed by t. This is a stochastic process, but this process has more structure than a general stochastic process since increments of the process are characterized deterministically by (5.1).

The methods discussed in this paper extend to non-autonomous systems directly. Given the system

$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, t), & T \geq t \geq 0, \\ \boldsymbol{x}(0) = \boldsymbol{x}_0, \end{cases}$$

we form the augmented vector $\boldsymbol{z} = (\boldsymbol{x}^\top, t)^\top$, and the augmented vector field $\boldsymbol{g} = (\boldsymbol{f}^\top, 1)^\top$. The initial condition becomes $\boldsymbol{z}(0) = (\boldsymbol{x}_0^\top, 0)^\top$. Using this trick, we simply increase the dimension of the system by one and consider time to be a dependent variable (see [27]). It should be noted that this can turn a linear system into a nonlinear one, and we also need to insure that the new vector field $\boldsymbol{g}$ has the necessary Lipshitz properties (below) with respect to the new variable t.

We consider the practical goal of computing a quantity of interest, which we assume can be represented as a linear functional of the form

$$q(\omega) = q(\boldsymbol{\lambda}(\omega)) = \int_0^T \langle \boldsymbol{x}(s;\boldsymbol{\lambda}(\omega)), \boldsymbol{\psi}(s) \rangle \, \mathrm{d}s, \tag{5.2}$$

where $\boldsymbol{\psi}$ is a function of time corresponding to the quantity of interest. Some common choices are

- $\boldsymbol{\psi}(s) = \delta(s-t)(0,\ldots,1,0,\ldots)^\top$, which yields the i^{th} component of $\boldsymbol{x}(t;\omega)$ at time t.
- $\boldsymbol{\psi}(s) = (1,\ldots,1)^\top/T$, which yields the time average over $[0,T]$ of all the components.

- $\psi(s) = (0, \ldots, 1, 0, \ldots)^\top / T$, which yields the time average over $[0, T]$ of a particular component of the solution.

We derive a useful representation formula for the quantity of interest (5.2) by solving the linearized adjoint problem,

$$\begin{cases} -\dot{\phi}(t) - A^\top(t)\phi(t) = \psi(t), & T \geq t \geq 0, \\ \phi(T) = \mathbf{0}, \end{cases} \tag{5.3}$$

where $A(t) \equiv D_{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{y}(t); \bar{\boldsymbol{\mu}}_1)$, $\boldsymbol{y}(t)$ is a deterministic solution solving the system (5.1) for the fixed parameter $\bar{\boldsymbol{\mu}} = (\bar{\boldsymbol{\mu}}_1, \bar{\boldsymbol{\mu}}_0)^\top$.

We obtain the following representation;

Theorem 5.1.1. *For $\boldsymbol{x}$, $\boldsymbol{y}$, ϕ and ψ as above,*

$$q(\boldsymbol{\lambda}) = \int_0^T \langle \boldsymbol{x}, \psi \rangle \, \mathrm{d}s \approx \int_0^T \langle \boldsymbol{y}, \psi \rangle \, \mathrm{d}s + \langle \boldsymbol{\lambda}_0 - \bar{\boldsymbol{\mu}}_0, \phi(0) \rangle + \int_0^T \langle D_{\boldsymbol{\lambda}_1} \boldsymbol{f}(\boldsymbol{y}; \bar{\boldsymbol{\mu}}_1)(\boldsymbol{\lambda}_1 - \bar{\boldsymbol{\mu}}_1), \phi \rangle \, \mathrm{d}s. \tag{5.4}$$

The last term on the right-hand side describes the effect of variations in the model parameters. Note, that this term suggests that knowledge of $D_{\boldsymbol{\lambda}_1} \boldsymbol{f}(\boldsymbol{y}; \bar{\boldsymbol{\mu}}_1)$ and ϕ are necessary to accurately estimate the density of the output corresponding to variation $\boldsymbol{\lambda}_1 - \bar{\boldsymbol{\mu}}_1$ in the input. The second to last term on the right-hand side of (5.4) describes the effect of variations in the initial conditions.

Proof. We consider a perturbation $\boldsymbol{\lambda} = \bar{\boldsymbol{\mu}} + \boldsymbol{h}$ along with the corresponding solution $\boldsymbol{x}(t) = \boldsymbol{x}(t; \boldsymbol{\lambda})$ of (5.1). We let $\boldsymbol{y}(t)$ and $\phi(t)$ be as defined in the

text and we $\boldsymbol{e}(t) \equiv \boldsymbol{x}(t) - \boldsymbol{y}(t)$. We have the relation

$$\begin{aligned}
\int_0^T \langle \boldsymbol{e}, \boldsymbol{\psi} \rangle \, \mathrm{d}s &= \int_0^T \langle \boldsymbol{e}, -\dot{\boldsymbol{\phi}} - A^T \boldsymbol{\phi} \rangle \mathrm{d}s \\
&= -\left[\langle \boldsymbol{e}(T), \boldsymbol{\phi}(T) \rangle - \langle \boldsymbol{e}(0), \boldsymbol{\phi}(0) \rangle - \int_0^T \langle \dot{\boldsymbol{e}}, \boldsymbol{\phi} \rangle \, \mathrm{d}s \right] - \int_0^T \langle \boldsymbol{e}, A^T \boldsymbol{\phi} \rangle \, \mathrm{d}s \\
&= \langle \boldsymbol{\lambda}_0 - \bar{\boldsymbol{\mu}}_0, \boldsymbol{\phi}(0) \rangle + \int_0^T \langle \dot{\boldsymbol{e}} - A\boldsymbol{e}, \boldsymbol{\phi} \rangle \, \mathrm{d}s \\
&= \langle \boldsymbol{\lambda}_0 - \bar{\boldsymbol{\mu}}_0, \boldsymbol{\phi}(0) \rangle + \int_0^T \langle \dot{\boldsymbol{x}} - A\boldsymbol{x}, \boldsymbol{\phi} \rangle \, \mathrm{d}s - \int_0^T \langle \dot{\boldsymbol{y}} - A\boldsymbol{y}, \boldsymbol{\phi} \rangle \, \mathrm{d}s,
\end{aligned}$$

since $\boldsymbol{\phi}(T) = \boldsymbol{e}(0) = \mathbf{0}$. Expanding $\boldsymbol{f}$ in a Taylor approximation,

$$\begin{aligned}
\boldsymbol{f}(\boldsymbol{x}; \boldsymbol{\lambda}_1) = \boldsymbol{f}(\boldsymbol{y}; \bar{\boldsymbol{\mu}}_1) &+ D_{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{y}; \bar{\boldsymbol{\mu}}_1)(\boldsymbol{x} - \boldsymbol{y}) \\
&+ D_{\boldsymbol{\lambda}_1} \boldsymbol{f}(\boldsymbol{y}; \bar{\boldsymbol{\mu}}_1)(\boldsymbol{\lambda}_1 - \bar{\boldsymbol{\mu}}_1) + \boldsymbol{R}(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{\lambda}_1, \bar{\boldsymbol{\mu}}_1),
\end{aligned}$$

where $|\boldsymbol{R}(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{\lambda}_1, \bar{\boldsymbol{\mu}}_1)|/(|\boldsymbol{x}-\boldsymbol{y}|+|\boldsymbol{\lambda}_1-\bar{\boldsymbol{\mu}}_1|) \to \mathbf{0}$ as $(|\boldsymbol{x}-\boldsymbol{y}|+|\boldsymbol{\lambda}_1-\bar{\boldsymbol{\mu}}_1|) \to \mathbf{0}$. Using the above, we get

$$\begin{aligned}
\int_0^T \langle \boldsymbol{e}, \boldsymbol{\psi} \rangle \, \mathrm{d}s = \langle \boldsymbol{\lambda}_0 - \bar{\boldsymbol{\mu}}_0, \boldsymbol{\phi}(0) \rangle &+ \int_0^T \langle D_{\boldsymbol{\lambda}_1} \boldsymbol{f}(\boldsymbol{y}; \bar{\boldsymbol{\mu}}_1)(\boldsymbol{\lambda}_1 - \bar{\boldsymbol{\mu}}_1), \boldsymbol{\phi} \rangle \, \mathrm{d}s \\
&+ \int_0^T \langle \boldsymbol{R}(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{\lambda}_1, \bar{\boldsymbol{\mu}}_1), \boldsymbol{\phi} \rangle \, \mathrm{d}s \quad (5.5)
\end{aligned}$$

or equivalently,

$$\begin{aligned}
\int_0^T \langle \boldsymbol{x}, \boldsymbol{\psi} \rangle \, \mathrm{d}s = \int_0^T \langle \boldsymbol{y}, \boldsymbol{\psi} \rangle \, \mathrm{d}s &+ \langle \boldsymbol{\lambda}_0 - \bar{\boldsymbol{\mu}}_0, \boldsymbol{\phi}(0) \rangle \\
&+ \int_0^T \langle D_{\boldsymbol{\lambda}_1} \boldsymbol{f}(\boldsymbol{y}; \bar{\boldsymbol{\mu}}_1)(\boldsymbol{\lambda}_1 - \bar{\boldsymbol{\mu}}_1), \boldsymbol{\phi} \rangle \, \mathrm{d}s \\
&+ \int_0^T \langle \boldsymbol{R}(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{\lambda}_1, \bar{\boldsymbol{\mu}}_1), \boldsymbol{\phi} \rangle \, \mathrm{d}s.
\end{aligned}$$

We interpret the second term as the effect of varying the initial condition and the third as the effect of varying the parameters in the model itself. Note that even when $\boldsymbol{f}$ is linear in $\boldsymbol{x}$ and in $\boldsymbol{\lambda}_1$, $\boldsymbol{R}$ is typically not zero since $\boldsymbol{f}$ is not linear in their combination. □

The nature of this approximation, i.e. what does "$\approx$" mean above, is described precisely in the following theorem.

Theorem 5.1.2. *If there is a constant L such that $\boldsymbol{f}$ satisfies the Lipschitz condition,*

$$|\boldsymbol{f}(\boldsymbol{x};\boldsymbol{\lambda}_1) - \boldsymbol{f}(\boldsymbol{y};\boldsymbol{\mu}_1)| \leq L(|\boldsymbol{x}-\boldsymbol{y}| + |\boldsymbol{\lambda}_1 - \boldsymbol{\mu}_1|), \tag{5.6}$$

for all $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{K}$ and $\boldsymbol{\mu}, \boldsymbol{\lambda} \in \Lambda$, where the solutions to (5.1) remain in the set $\mathcal{K}$ for all parameters in Λ and $t \in [0,T]$, then the functional (5.2) of the solution to (5.1) satisfies

$$\begin{aligned}\nabla q(\bar{\boldsymbol{\mu}})[\cdot] &= \nabla \int_0^T \langle \boldsymbol{x}(s;\boldsymbol{\lambda}), \boldsymbol{\psi}(s)\rangle \,\mathrm{d}s \Big|_{\boldsymbol{\lambda}=\bar{\boldsymbol{\mu}}} [\cdot] \\ &= \langle [\cdot]_0, \boldsymbol{\phi}(0)\rangle + \int_0^T \langle D_{\boldsymbol{\lambda}_1}\boldsymbol{f}(\boldsymbol{y};\bar{\boldsymbol{\mu}}_1)\,[\cdot]_1, \boldsymbol{\phi}\rangle \,\mathrm{d}s, \end{aligned} \tag{5.7}$$

where $[\cdot]$ appears since this derivative is a linear operator from $\mathbb{R}^d \to \mathbb{R}$, and $\boldsymbol{y}$, $\boldsymbol{\phi}$ are defined as above.

This states that the representation in (5.4) is a linear approximation to the function q at the parameter value $\bar{\boldsymbol{\mu}}$.

Proof. We apply the definition of the (Fréchet) derivative with $\boldsymbol{x} = \boldsymbol{x}(s;\bar{\boldsymbol{\mu}}+\boldsymbol{h})$ to get,

$$\begin{aligned} v(\boldsymbol{h}) &= \Bigg| \int_0^T \langle \boldsymbol{x}, \boldsymbol{\psi}\rangle \,\mathrm{d}s - \int_0^T \langle \boldsymbol{y}, \boldsymbol{\psi}\rangle \,\mathrm{d}s \\ &\qquad - \left(\langle \boldsymbol{h}_0, \boldsymbol{\phi}(0)\rangle + \int_0^T \langle D_{\boldsymbol{\lambda}_1}\boldsymbol{f}(\boldsymbol{y};\bar{\boldsymbol{\mu}}_1)\boldsymbol{h}_1, \boldsymbol{\phi}\rangle \,\mathrm{d}s \right) \Bigg| \\ &= \left| \int_0^T \langle \boldsymbol{R}(\boldsymbol{x},\boldsymbol{y};\bar{\boldsymbol{\mu}}_1 + \boldsymbol{h}_1, \bar{\boldsymbol{\mu}}_1), \boldsymbol{\phi}\rangle \,\mathrm{d}s \right| \\ &\leq \int_0^T |\boldsymbol{R}(\boldsymbol{x},\boldsymbol{y};\bar{\boldsymbol{\mu}}_1 + \boldsymbol{h}_1, \bar{\boldsymbol{\mu}}_1)|\,|\boldsymbol{\phi}| \,\mathrm{d}s. \end{aligned}$$

Writing the solution to (5.1) as

$$\boldsymbol{x}(t;\boldsymbol{\lambda}) = \boldsymbol{\lambda}_0 + \int_0^t \boldsymbol{f}(\boldsymbol{x}(s;\boldsymbol{\lambda});\boldsymbol{\lambda}_1)\,\mathrm{d}s,$$

and using the Lipschitz property of $\boldsymbol{f}$, we get

$$\begin{aligned}|\boldsymbol{x}(t) - \boldsymbol{y}(t)| &\le |\boldsymbol{\lambda}_0 - \bar{\boldsymbol{\mu}}_0| + \int_0^t |\boldsymbol{f}(\boldsymbol{x};\bar{\boldsymbol{\mu}}_1 + \boldsymbol{h}_1) - \boldsymbol{f}(\boldsymbol{y};\bar{\boldsymbol{\mu}}_1)|\,\mathrm{d}s \\ &\le L\int_0^t |\boldsymbol{x} - \boldsymbol{y}|\,\mathrm{d}s + (LT+1)|\boldsymbol{h}|,\end{aligned}$$

where ($|\boldsymbol{\lambda}_0 - \bar{\boldsymbol{\mu}}_0| \le |\boldsymbol{h}|$). By Gronwall's lemma [27]

$$|\boldsymbol{x}(t) - \boldsymbol{y}(t)| \le (LT+1)|\boldsymbol{h}|e^{Lt}.$$

We choose δ so ($|\boldsymbol{x} - \boldsymbol{y}| + |\boldsymbol{\lambda}_1 - \bar{\boldsymbol{\mu}}_1|$) $\le \delta$ implies $\boldsymbol{R}(\boldsymbol{x},\boldsymbol{y};\bar{\boldsymbol{\mu}}_1 + \boldsymbol{h}_1, \bar{\boldsymbol{\mu}}_1) \le \epsilon(|\boldsymbol{x} - \boldsymbol{y}| + |\boldsymbol{h}_1|) \le \epsilon(|\boldsymbol{x} - \boldsymbol{y}| + |\boldsymbol{h}|)$. For $\boldsymbol{h}$ sufficiently small ($|\boldsymbol{h}| \le \delta_1$),

$$|\boldsymbol{x}(s) - \boldsymbol{y}(s)| + |\boldsymbol{h}| \le (LT+1)|\boldsymbol{h}|e^{LT} + |\boldsymbol{h}| \le \delta.$$

For $|\boldsymbol{h}| \le \delta_1$,

$$\begin{aligned}|v(\boldsymbol{h})| &\le \int_0^T |\boldsymbol{R}(\boldsymbol{x},\boldsymbol{y};\bar{\boldsymbol{\mu}}_1 + \boldsymbol{h}_1, \bar{\boldsymbol{\mu}}_1)|\,|\boldsymbol{\phi}|\,\mathrm{d}s \\ &\le \epsilon\left[((LT+1)e^{LT}+1)\int_0^T |\boldsymbol{\phi}|\,\mathrm{d}s\right]|\boldsymbol{h}|,\end{aligned}$$

and so $|v(\boldsymbol{h})|/|\boldsymbol{h}| \le C\epsilon$ for any ϵ, provided we choose $\boldsymbol{h}$ small enough. Therefore $|v(\boldsymbol{h})|/|\boldsymbol{h}| \to 0$ as $|\boldsymbol{h}| \to 0$, which proves the result. □

To demonstrate this theorem, in Tab. 5.1 we compare the derivatives computed by the adjoint technique with finite difference approximations at one nominal value of the parameters. Errors in the approximation are introduced by the numerical error of solving the ODE, and by the numerical integration scheme used to compute the terms appearing in the representation formula (5.1.1). Details on the computation of these gradients is given in Sec. A.7.

Parameter	Value ($\boldsymbol{\lambda}$)	$\frac{q_1-q_2}{dx}$	Adjoint $\partial_{\lambda_i} q(\boldsymbol{\lambda})$
ar	0.2	-0.38087	-0.20200
rn	1.	21.90023	21.87893
K	100.	0.07715	0.07713
pr	0.1	-1.33461	-1.29951
dn	0.2	-27.00555	-27.10778
ri	0.2	-7.81719	-7.82529
di	1.	-11.72673	-11.82434

Table 5.1: Comparison of sensitivity derivatives for the SIR model (9.4) using finite difference approximations and the adjoint computation. The data ψ is as in the example leading up to Fig. 9.4. Here $dx = 0.01$, $q_1 = q(\boldsymbol{\lambda} + dx\boldsymbol{e}_i), i = 1, \ldots, 7$, and $q_2 = q(\boldsymbol{\lambda})$.

5.2 Reaction diffusion equations

We prove a similar result for a specific type of PDE. It is difficult to derive general results for PDE's, so we consider a system of reaction diffusion equations of the form,

$$\begin{cases} \partial_t \boldsymbol{u} - \nabla \cdot (a \nabla \boldsymbol{u}) = \boldsymbol{f}(\boldsymbol{u}, \boldsymbol{\lambda}) & \text{in} \quad \Omega \times (0, T), \\ \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}} = 0 & \partial\Omega \times (0, T), \\ \boldsymbol{u} = \boldsymbol{u}_0 & \Omega \times \{0\}, \end{cases} \tag{5.8}$$

where $\boldsymbol{u} = \boldsymbol{u}(x,t) = (u_1(x,t), \ldots, u_m(x,t))^\top$ represent concentrations of some entities such as chemicals, populations, or disease, which diffuse in space and react with one another as described by $\boldsymbol{f} = (f_1, \ldots, f_m)$, depending on a finite number of parameters $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_p)$. The choice of Neumann boundary condition is arbitrary, and introduced only to simplify the presentation while preserving interesting dynamics in the examples.

5.2.1 Weak Formulation

Taking the inner product of (5.8) with a test function $\boldsymbol{v}$, and integrating by parts in the second term, we arrive at the weak formulation of this

problem, which, with $V = H^1(\Omega)^m$, is to find a $\boldsymbol{u} \in L^2(0,T;V)$, with $\partial_t \boldsymbol{u} \in L^2(0,T;V')$ such that

$$\langle \partial_t \boldsymbol{u}, \boldsymbol{v} \rangle_{V' \times V} + B[\boldsymbol{u}, \boldsymbol{v}] = (\boldsymbol{f}(\boldsymbol{u}, \boldsymbol{\lambda}), \boldsymbol{v})_{L^2} \quad \text{a.e.} \quad 0 \leq t \leq T, \tag{5.9}$$

for all $\boldsymbol{v} \in V$, where $B[\cdot,\cdot]$ is the bilinear form associated with the dissipative term, namely $B[\boldsymbol{u}, \boldsymbol{v}] = \sum_{i=1}^m \int_\Omega a \nabla u_i \cdot \nabla v_i \, \mathrm{d}x$, and $\boldsymbol{u}(0) = \boldsymbol{u}_0$.

We use a simplified notation here, denoting by the L^2 inner product the $(L^2)^m$ inner product, which is the integral over Ω of the Euclidean inner product in $\mathbb{R}^m$. Also, the duality pairing of $V \times V'$ is the Euclidean product of the duality pairings for each component.

The existence of such a solution typically requires that $\boldsymbol{f}$ be lipschitz continuous in some invariant region for the solutions $\boldsymbol{u}$, whereupon a fixed point argument such as that in [16] can be used to prove the existence of a solution to the weak problem.

5.2.2 The Adjoint

We first solve the weak problem for fixed parameter $\boldsymbol{\mu}$, obtaining $\boldsymbol{w} \in V$. We linearize around this solution and solve an adjoint problem for $\boldsymbol{\phi} \in L^2(0,T;V)$, $\partial_t \boldsymbol{\phi} \in L^2(0,T;V')$, which satisfies

$$\begin{aligned} -\langle \boldsymbol{v}, \partial_t \boldsymbol{\phi} \rangle_{V \times V'} + B[\boldsymbol{v}, \boldsymbol{\phi}] = \\ (\boldsymbol{v}, D_{\boldsymbol{u}} \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{\mu})^\top \boldsymbol{\phi})_{L^2} + \langle \boldsymbol{v}, \boldsymbol{\psi} \rangle_{V \times V'} \quad \text{a.e.} \quad 0 \leq t \leq T, \end{aligned} \tag{5.10}$$

for all $\boldsymbol{v} \in H^1(\Omega)$, with dual data $\boldsymbol{\psi} \in L^2(0,T;V')$, and $\boldsymbol{\phi}(T) = \boldsymbol{0}$. This corresponds formally to the PDE

$$\begin{cases} -\partial_t \boldsymbol{\phi} - \nabla \cdot (a \nabla \boldsymbol{\phi}) = D_{\boldsymbol{u}} \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{\mu})^\top \boldsymbol{\phi} + \boldsymbol{\psi} & \text{in} \quad \Omega \times (0,T), \\ \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{n}} = 0 & \partial\Omega \times (0,T), \\ \boldsymbol{\phi} = \boldsymbol{0} & \Omega \times \{T\}. \end{cases} \tag{5.11}$$

Now if $\boldsymbol{u}$ is a solution of (5.9) corresponding to parameter $\boldsymbol{\lambda}$, setting $\boldsymbol{e} = \boldsymbol{u} - \boldsymbol{w}$, which is in V, using equations (5.9) and (5.10) we find

$$\begin{aligned}
&\int_0^T \langle \boldsymbol{e}, \boldsymbol{\psi} \rangle_{V \times V'} \, \mathrm{d}t = \\
&\quad \int_0^T -\langle \boldsymbol{e}, \partial_t \boldsymbol{\phi} \rangle_{V \times V'} \, \mathrm{d}t + \int_0^T \left(B[\boldsymbol{e}, \boldsymbol{\phi}] - (\boldsymbol{e}, D_{\boldsymbol{u}} \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{\mu})^\top \boldsymbol{\phi})_{L^2} \right) \, \mathrm{d}t \\
&\quad = \int_0^T \langle \boldsymbol{\phi}, \partial_t \boldsymbol{e} \rangle_{V \times V'} \mathrm{d}t + \int_0^T \left(B[\boldsymbol{\phi}, \boldsymbol{e}] - (\boldsymbol{\phi}, D_{\boldsymbol{u}} \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{\mu}) \boldsymbol{e})_{L^2} \right) \mathrm{d}t \\
&\quad = \int_0^T (\boldsymbol{f}(\boldsymbol{u}, \boldsymbol{\lambda}) - \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{\mu}) - D_{\boldsymbol{u}} \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{\mu})(\boldsymbol{u} - \boldsymbol{w}), \boldsymbol{\phi})_{L^2} \, \mathrm{d}t \\
&\quad = \int_0^T (D_{\boldsymbol{\lambda}} \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{\mu})(\boldsymbol{\lambda} - \boldsymbol{\mu}) + \boldsymbol{R}, \boldsymbol{\phi})_{L^2} \, \mathrm{d}t, \quad (5.12)
\end{aligned}$$

where $\boldsymbol{R}$ is a higher order term in the Taylor expansion of $\boldsymbol{f}$. We have, then, the representation formula

$$\begin{aligned}
&\int_0^T \langle \boldsymbol{u}, \boldsymbol{\psi} \rangle_{V \times V'} \, \mathrm{d}t \approx \\
&\qquad \int_0^T \langle \boldsymbol{w}, \boldsymbol{\psi} \rangle_{V \times V'} \, \mathrm{d}t + \int_0^T (D_{\boldsymbol{\lambda}} \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{\mu})(\boldsymbol{\lambda} - \boldsymbol{\mu}), \boldsymbol{\phi})_{L^2} \, \mathrm{d}t.
\end{aligned}$$

Using the continuity of the inner product on $\mathbb{R}^m$, the last term on the right has the more computationally friendly form

$$\int_0^T \int_\Omega \left(\boldsymbol{\lambda} - \boldsymbol{\mu}, D_{\boldsymbol{u}} \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{\mu})^\top \boldsymbol{\phi} \right)_{\mathbb{R}^m} \mathrm{d}x \, \mathrm{d}t = \left(\boldsymbol{\lambda} - \boldsymbol{\mu}, \int_0^T \int_\Omega D_{\boldsymbol{\lambda}} \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{\mu})^\top \boldsymbol{\phi} \, \mathrm{d}x \, \mathrm{d}t \right)_{\mathbb{R}^m}.$$

5.2.3 Convergence of the Representation

We begin by estimating $\|\boldsymbol{e}\|_{L^2}$. For $\boldsymbol{v} \in V$, it satisfies

$$\langle \partial_t \boldsymbol{e}, \boldsymbol{v} \rangle_{V' \times V} + B[\boldsymbol{e}, \boldsymbol{v}] = (\boldsymbol{f}(\boldsymbol{u}, \boldsymbol{\lambda}) - \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{\mu}), \boldsymbol{v})_{L^2} \quad \text{a.e.} \quad 0 \le t \le T,$$

in particular, with $\boldsymbol{v} = \boldsymbol{e}$, we have

$$\tfrac{1}{2} \tfrac{\mathrm{d}}{\mathrm{d}t} \|\boldsymbol{e}\|_{L^2}^2 + B[\boldsymbol{e}, \boldsymbol{e}] = (\boldsymbol{f}(\boldsymbol{u}, \boldsymbol{\lambda}) - \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{\mu}), \boldsymbol{e})_{L^2} \quad \text{a.e.} \quad 0 \le t \le T. \quad (5.13)$$

Under the Lipschitz assumption that

$$|\boldsymbol{f}(\boldsymbol{u}_1, \boldsymbol{\lambda}_1) - \boldsymbol{f}(\boldsymbol{u}_2, \boldsymbol{\lambda}_2)| \leq K\sqrt{|\boldsymbol{u}_1 - \boldsymbol{u}_2|^2 + |\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2|^2} \tag{5.14}$$

for $x \in \Omega, t \in [0, T]$ when $\boldsymbol{\lambda}_i \in H_p$ and $\boldsymbol{u}_i$ $(i = 1, 2)$ are solutions to (5.9), integrating (5.13) from 0 to t we have

$$\begin{aligned}
\|\boldsymbol{e}(t)\|_{L^2}^2 &\leq 2\int_0^t (\boldsymbol{f}(\boldsymbol{u}, \boldsymbol{\lambda}) - \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{\mu}), \boldsymbol{e})_{L^2}\, \mathrm{d}s \\
&\leq 2\int_0^t \|\boldsymbol{f}(\boldsymbol{u}, \boldsymbol{\lambda}) - \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{\mu})\|_{L^2} \|\boldsymbol{e}\|_{L^2}\, \mathrm{d}s \\
&\leq \int_0^t \|\boldsymbol{f}(\boldsymbol{u}, \boldsymbol{\lambda}) - \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{\mu})\|_{L^2}^2\, \mathrm{d}s + \int_0^t \|\boldsymbol{e}\|_{L^2}^2\, \mathrm{d}s \\
&\leq (K^2 + 1)\int_0^t \|\boldsymbol{e}\|_{L^2}^2\, \mathrm{d}s + TK^2|\Omega||\boldsymbol{\lambda} - \boldsymbol{\mu}|^2.
\end{aligned} \tag{5.15}$$

By Gronwall's inequality,

$$\|\boldsymbol{e}(t)\|_{L^2} \leq K\sqrt{(T|\Omega|e^{(K^2+1)T})}|\boldsymbol{\lambda} - \boldsymbol{\mu}|. \tag{5.16}$$

Theorem 5.2.1. *With* $q(\boldsymbol{\lambda}) = \int_0^T \langle \boldsymbol{u}, \boldsymbol{\psi}\rangle_{V \times V'}\, \mathrm{d}t$, *we have*

$$\nabla q(\boldsymbol{\mu})[\cdot] = \left(\cdot, \int_0^T \int_\Omega D_{\boldsymbol{\lambda}} \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{\mu})^\top \boldsymbol{\phi}\, \mathrm{d}x\, \mathrm{d}t\right)_{\mathbb{R}^m},$$

provided $\boldsymbol{\mu}$ *lies in a set* H_p *such that the lipschitz condition (5.14) holds for all solutions* $\boldsymbol{u}$ *to (5.9) for parameters in* H_p.

Proof. As above, we let $\boldsymbol{u}$ be the solution corresponding to $\boldsymbol{\lambda} = \boldsymbol{\mu} + \boldsymbol{h}$, and $\boldsymbol{w}$ the solution for parameter $\boldsymbol{\mu}$, and we have

$$\begin{aligned}
v(\boldsymbol{h}) &= \left|q(\boldsymbol{\mu} + \boldsymbol{h}) - q(\boldsymbol{\mu}) - \left(\boldsymbol{h}, \int_0^T \int_\Omega D_{\boldsymbol{\lambda}} \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{\mu})^\top \boldsymbol{\phi}\, \mathrm{d}x\, \mathrm{d}t\right)_{\mathbb{R}^m}\right| \\
&= \left|\int_0^T (\boldsymbol{R}, \boldsymbol{\phi})_{L^2}\, \mathrm{d}t\right|
\end{aligned} \tag{5.17}$$

□

By Taylor's Theorem applied repeatedly to $g_i(t) = f_i(t\boldsymbol{u}+(1-t)\boldsymbol{w}, \boldsymbol{u}+t\boldsymbol{h}), i = 1 \ldots m$, the remainder has the form

$$\begin{aligned}\boldsymbol{R}(\boldsymbol{u}, \boldsymbol{w}, \boldsymbol{\lambda}, \boldsymbol{\mu}) &= D^2_{\boldsymbol{\lambda u}}\boldsymbol{f}\,(\boldsymbol{\lambda}-\boldsymbol{\mu})(\boldsymbol{u}-\boldsymbol{w}) \\ &\quad + \tfrac{1}{2}D^2_{\boldsymbol{uu}}\boldsymbol{f}\,(\boldsymbol{u}-\boldsymbol{w})(\boldsymbol{u}-\boldsymbol{w}) + \tfrac{1}{2}D^2_{\boldsymbol{\lambda\lambda}}\boldsymbol{f}\,(\boldsymbol{\lambda}-\boldsymbol{\mu})(\boldsymbol{\lambda}-\boldsymbol{\mu}) \\ &= \boldsymbol{R}_I + \boldsymbol{R}_{II} + \boldsymbol{R}_{III}\end{aligned}$$

where each of $D^2_{\boldsymbol{\lambda u}}\boldsymbol{f}, D^2_{\boldsymbol{uu}}\boldsymbol{f}, D^2_{\boldsymbol{\lambda\lambda}}\boldsymbol{f}$ are bilinear forms composed of stacked Hessians, evaluated at points $\boldsymbol{\xi}_i, \boldsymbol{\eta}_i, (i = 1 \ldots m)$ on the lines between $\boldsymbol{u}, \boldsymbol{w}$ and $\boldsymbol{\lambda}, \boldsymbol{\mu}$. For instance,

$$\begin{aligned}&D^2_{\boldsymbol{\lambda\lambda}}\boldsymbol{f}(\boldsymbol{\lambda}-\boldsymbol{\mu})(\boldsymbol{\lambda}-\boldsymbol{\mu}) = \\ &\qquad \left((\boldsymbol{\lambda}-\boldsymbol{\mu})^\top H_1(\boldsymbol{\lambda}-\boldsymbol{\mu}), \ldots, (\boldsymbol{\lambda}-\boldsymbol{\mu})^\top H_m(\boldsymbol{\lambda}-\boldsymbol{\mu})\right)^\top,\end{aligned}$$

for matrices $H_i = [\partial^2_{\lambda_k\lambda_l} f_i(\boldsymbol{\xi}_i, \boldsymbol{\eta}_i)]_{k,l}$.

We estimate

$$\begin{aligned}&\|D^2_{\boldsymbol{\lambda\lambda}}\boldsymbol{f}(\boldsymbol{\lambda}-\boldsymbol{\mu})(\boldsymbol{\lambda}-\boldsymbol{\mu})\|^2_{L^2} \\ &\quad = \int_\Omega \left|\left((\boldsymbol{\lambda}-\boldsymbol{\mu})^\top H_1(\boldsymbol{\lambda}-\boldsymbol{\mu}), \ldots, (\boldsymbol{\lambda}-\boldsymbol{\mu})^\top H_m(\boldsymbol{\lambda}-\boldsymbol{\mu})\right)^\top\right|^2 \mathrm{d}x \\ &\quad = \int_\Omega |(\boldsymbol{\lambda}-\boldsymbol{\mu})^\top H_1(\boldsymbol{\lambda}-\boldsymbol{\mu})|^2 + \cdots + |(\boldsymbol{\lambda}-\boldsymbol{\mu})^\top H_m(\boldsymbol{\lambda}-\boldsymbol{\mu})|^2 \,\mathrm{d}x \\ &\quad \leq C\int_\Omega |\boldsymbol{\lambda}-\boldsymbol{\mu}|^4 \,\mathrm{d}x,\end{aligned}$$

where $C = m \max_i \|H_i\|$, which is independent of time when $\boldsymbol{f} \in C^2$ and the solution stays in a compact set. Hence, $\|D^2_{\boldsymbol{\lambda\lambda}}\boldsymbol{f}(\boldsymbol{\lambda}-\boldsymbol{\mu})(\boldsymbol{\lambda}-\boldsymbol{\mu})\|_{L^2} \leq C_{III}|\boldsymbol{\lambda}-\boldsymbol{\mu}|^2$, where C_{III} depends only on T and $|\Omega|$.

Similarly,

$$\|D^2_{\boldsymbol{\lambda u}}\boldsymbol{f}(\boldsymbol{\lambda}-\boldsymbol{\mu})(\boldsymbol{u}-\boldsymbol{w})\|_{L^2} \leq C|\boldsymbol{\lambda}-\boldsymbol{\mu}|\|\boldsymbol{u}-\boldsymbol{w}\|_{L^2} \leq C_I|\boldsymbol{\lambda}-\boldsymbol{\mu}|^2$$

by (5.16), where C_I depends only on T, $|\Omega|$ and the Lipschitz constant of $\boldsymbol{f}$.

The same type of estimation for the $D^2_{\boldsymbol{uu}}\boldsymbol{f}$ term fails, so we appeal to an assumption on the dual solution $\boldsymbol{\phi}$. In many problems it is reasonable to assume that $\|\boldsymbol{\phi}\|_{L^\infty}$ is uniformly bounded in space and $[0,T]$, in which case

$$
\begin{aligned}
&\left(D^2_{\boldsymbol{uu}}\boldsymbol{f}(\boldsymbol{u}-\boldsymbol{w})(\boldsymbol{u}-\boldsymbol{w}),\boldsymbol{\phi}\right)_{L^2} \\
&\qquad \le \int_\Omega \left|\left(D^2_{\boldsymbol{uu}}\boldsymbol{f}(\boldsymbol{u}-\boldsymbol{w})(\boldsymbol{u}-\boldsymbol{w}),\boldsymbol{\phi}\right)_{\mathbb{R}^m}\right| \,\mathrm{d}x \\
&\qquad \le c\|\boldsymbol{\phi}\|_{L^\infty}\int_\Omega \left|D^2_{\boldsymbol{uu}}\boldsymbol{f}(\boldsymbol{u}-\boldsymbol{w})(\boldsymbol{u}-\boldsymbol{w})\right| \,\mathrm{d}x \\
&= c\|\boldsymbol{\phi}\|_{L^\infty}\int_\Omega \sqrt{|(\boldsymbol{u}-\boldsymbol{w})^\top H_1(\boldsymbol{u}-\boldsymbol{w})|^2+\cdots+|(\boldsymbol{u}-\boldsymbol{w})^\top H_m(\boldsymbol{u}-\boldsymbol{w})|^2}\,\mathrm{d}x \\
&\le C\|\boldsymbol{\phi}\|_{L^\infty}\int_\Omega |\boldsymbol{u}-\boldsymbol{w}|^2\,\mathrm{d}x = c\|\boldsymbol{\phi}\|_{L^\infty}\|\boldsymbol{u}-\boldsymbol{w}\|^2_{L^2} \le C_{II}\|\boldsymbol{\phi}\|_{L^\infty}|\boldsymbol{\lambda}-\boldsymbol{\mu}|^2,
\end{aligned}
$$

where C_{II} is independent of $t\in[0,T]$.

Combining these results we have

$$
\begin{aligned}
v(\boldsymbol{h}) &\le \left|\int_0^T (\boldsymbol{R},\boldsymbol{\phi})_{L^2}\,\mathrm{d}t\right| \\
&\le \int_0^T |(\boldsymbol{R}_I,\boldsymbol{\phi})_{L^2}|\,\mathrm{d}x + \int_0^T |(\boldsymbol{R}_{II},\boldsymbol{\phi})_{L^2}|\,\mathrm{d}x + \int_0^T |(\boldsymbol{R}_{III},\boldsymbol{\phi})_{L^2}|\,\mathrm{d}x \\
&\le \int_0^T \|\boldsymbol{R}_I\|_{L^2}\|\boldsymbol{\phi}\|_{L^2}\,\mathrm{d}x + \tfrac{1}{2}TC_{II}|\boldsymbol{h}|^2 + \int_0^T \|\boldsymbol{R}_{III}\|_{L^2}\|\boldsymbol{\phi}\|_{L^2}\,\mathrm{d}x \\
&\le \left(C_I+\tfrac{1}{2}C_{III}\right)\|\boldsymbol{\phi}\|^2_{L^2([0,T];V)}|\boldsymbol{h}|^2 + \tfrac{1}{2}TC_{II}\max_{0\le t\le T}\|\boldsymbol{\phi}(t)\|_{L^\infty}|\boldsymbol{h}|^2 = C|\boldsymbol{h}|^2,
\end{aligned}
$$

and so $v(\boldsymbol{h})/|\boldsymbol{h}| \to 0$ as $|\boldsymbol{h}| \to 0$. By the linearity of the proposed gradient, $q(\boldsymbol{\lambda})$ is Fréchet differentiable at $\boldsymbol{\mu}$, and the theorem holds.

This theorem is demonstrated for the reaction diffusion example (10.1) in Tab. 5.2.

Parameter	Value ($\boldsymbol{\lambda}$)	$\frac{q_1-q_2}{dx}$	Adjoint $\partial_{\lambda_i} q(\boldsymbol{\lambda})$
r	5	0.48028	0.45700
s	1	14.71836	14.68010
p	1	-1.83012	-1.82710
q	1	-12.94280	-12.77330
w	1	13.29717	6.89120
k	10.1	-0.27989	-0.28430

Table 5.2: Comparison of sensitivity derivatives for the Predator-Prey model (10.1) using finite difference approximations and the adjoint computation. . Here $dx = 0.01$, $q_1 = q(\boldsymbol{\lambda} + dx\boldsymbol{e}_i), i = 1, \ldots, 6$, and $q_2 = q(\boldsymbol{\lambda})$. All other parameters are as in sec. 10.1, except that the ending time is T=5. The time step for the PDE integration here was 0.05.

Chapter 6

HIGHER ORDER PARAMETER SAMPLING (HOPS)

We introduced the HOPS method in the Finite Dimensional case above. In this chapter we discuss this method in full detail.

Our first goal is to develop a fast method to approximate the cumulative distribution function $F_q(x)$ of q, as defined in (5.2), given an arbitrary input distribution for $\boldsymbol{\lambda}$. Since we recover the entire distribution, we can compute the density as well as any desired statistic, e.g., means and moments, or other quantities of the form $E[\mathcal{S}(q(\omega))]$ for measurable functions $\mathcal{S}$. In particular, the plots of densities in this paper are computed using a kernel density estimator on a data set constructed to have the distribution of $F_{\tilde{q}}$, where $\tilde{q}$ is the computed approximation to q. See Sec. A.5 for details on how the densities are computed.

The accurate and efficient computation of a quantity of interest from the solution of a differential equation has received a great deal of attention in the context of *a posteriori* error estimation and adaptive error control for finite element methods, see [15]. In this setting, the discretization on which the equation is solved is refined only in the areas that contribute to the accuracy of the quantity of interest, which results in a highly efficient method for computing such information. We borrow tools developed for

a posteriori error analysis of finite element methods to tackle the problem of approximating F_q accurately and efficiently. Later, we explain how to use the results to adaptively sample parameter values in order to efficiently and accurately describe the effects of variation in the parameters on the solution.

The one point HOPS approximation can be written

$$q(\boldsymbol{\lambda}) \approx q(\bar{\boldsymbol{\mu}}) + \langle \nabla q(\bar{\boldsymbol{\mu}}), (\boldsymbol{\lambda} - \bar{\boldsymbol{\mu}}) \rangle.$$

We can use this to obtain a good approximation of the distribution of $q(\boldsymbol{\lambda})$, provided $\boldsymbol{\lambda}$ has small variance about $\bar{\boldsymbol{\mu}}$ and the function q is approximated well by its linearization at $\bar{\boldsymbol{\mu}}$. See Sec. A.7 for details on how the gradient of q is computed in practice.

The Lipschitz condition (5.6) is the natural extension of the standard assumption for a local existence and uniqueness result. For general $\boldsymbol{f}$, this requires the solution set $\mathcal{K}$ to be compact . The restrictions on the parameter set Λ are more complicated to determine and are highly problem dependent. However, in many cases, it appears that at least we have to assume that Λ is compact as well.

Consider the model

$$\begin{cases} x' = x^2, & 0 \leq t \leq T, \\ x(0) = x_0, \end{cases}$$

where x_0 is the uncertain parameter. The solution is $x(t) = 1/(x_0^{-1} - t)$, which has a finite time blow up at $t = 1/x_0$ whenever $x_0 > 0$. Any investigation of the solution over a time interval $[0, T]$ requires x_0 to be bounded above, otherwise there is no solution on any time interval. Less dramatically, consider computing the mean value of solutions of

$$\begin{cases} x' = \lambda x, & 0 \leq t \leq T, \\ x(0) = x_0, \end{cases}$$

with $\lambda < 0$. To determine the effect of variations in the parameter λ, it might seem natural to use the model $\lambda = N(\mu, \sigma^2)$ where $\mu < 0$ and $N(0, \sigma^2)$ is a mean zero normal random variable with small variance σ^2. Unfortunately, the expected values of the possible solutions is $E[x(t)] = x_0 e^{\mu t} e^{\sigma^2 t^2/2}$, which decays for a short time and then grows exponentially very rapidly! This suggests that a choice of normal perturbation for the parameter is inappropriate if we are interested in the mean value of the solutions, since the physically meaningless situation $\lambda > 0$ dominates the statistics of the output, no matter how negative the reference value μ might be and how improbable it is that $\lambda > 0$. Turning this around, if the normal distribution for the variation in the parameter needs to be used, then the mean value is not the right statistical information to try to compute.

In general, the differential equation is usually a valid model only for a certain range of parameters and ceases to be a relevant model when the parameters exceed this range. For general problems, the assumption that $\mathcal{K}$ is compact appears to be a natural *minimum* requirement. For specific problems with good dynamical behavior, this assumption can be relaxed.

We note that in practical Monte-Carlo computations, even when the normal distribution is involved, there is an implicit assumption that the operator being investigated dies off rapidly for large values of the random variable. When behavior of the function for large parameters is of interest, special techniques must be used to insure that the sampling is robust.

6.1 A Scalar Example

We consider a simple scalar example to demonstrate the ideas. Consider

$$\begin{cases} \dot{x}(t;\lambda) = \lambda x(t;\lambda), & t > 0, \\ x(0) = x_0, \end{cases} \tag{6.1}$$

where $\lambda(\omega)$ is a random variable with mean μ. Solving for the generalized Green's function corresponding to the parameter μ and the data $\psi(s) = \delta(s-t)$ yields the one point approximation to the functional q,

$$q(\lambda) = x(t;\lambda) \approx y(t;\mu) + (\lambda(\omega) - \mu) \int_0^t e^{\mu(t-s)} y(s;\mu)\, ds, \tag{6.2}$$

which is valid when the variance of $\lambda(\omega)$ is small.

In this example, it is possible to calculate the probability density of $x(t;\lambda(\omega))$ exactly. If ρ_λ is the density for $\lambda(\omega)$, then using a standard change of variable formula for densities gives the density ρ_x for $x(t;\lambda(\omega))$,

$$\rho_x(s;t) = \begin{cases} \rho_\lambda(-\frac{1}{t}\ln(\frac{s}{x_0}))\left|\frac{1}{ts}\right|, & sx_0 > 0,\ t > 0, \\ 0, & sx_0 < 0,\ t > 0, \\ \delta(s-x_0), & t = 0, \end{cases}$$

at each t.

The HOPS approximation has the form $y(t;\lambda) = a(t) + \lambda b(t)$ with

$$a(t) = x_0(1-\mu t)e^{\mu t}, \qquad b(t) = x_0 t e^{\mu t}.$$

Using the same change of variable formula, the density of this approximation is

$$\rho_y(s;t) = \rho_\lambda\left(\frac{s-a(t)}{b(t)}\right)\left|\frac{1}{b(t)}\right|$$

These densities are compared in Fig. 6.1. When $\lambda < 0$, the dissipative nature of the solutions makes it easy to obtain good approximations. However, when the distribution has support for positive λ or λ near zero, trajectories do not converge to a steady state, and the accuracy of a one-point HOPS approximation suffers.

We also note that differentiating the solution $y_0 e^{\mu t}$ at a parameter μ yields $y_0 t e^{\mu t}$, which is exactly the integral term in (6.2), thus illustrating Theorem 2.

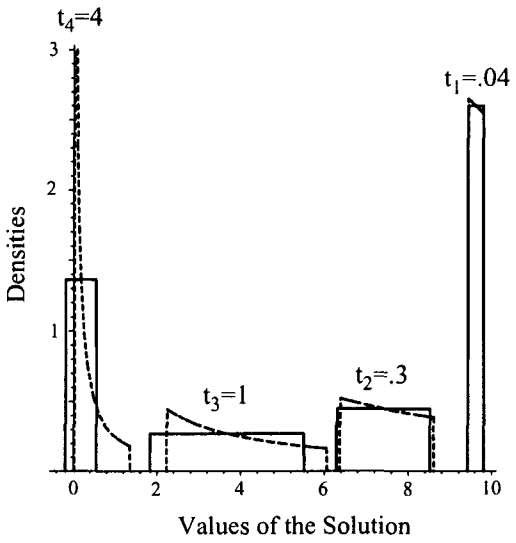

Figure 6.1: True (dashed) and approximate (solid) probability densities for (6.1) with $y_0 = x_0 = 10$, $\lambda(\omega)$ uniformly distributed on $[-3/2, -1/2]$, and $\psi(s) = \delta(s - t_i)$ at times $\{t_i\} = \{0.04, 0.3, 1, 4\}$. Integrating these functions from $-\infty$ to s gives the probability that the value of the solution x lies in $(-\infty, s]$. The consistent leftward "drift" of the support of the approximate density is a result of linearization error.

6.2 Multi-point HOPS Approximations

Generally, linearization around a single reference point cannot be used to accurately represent the response of a system to variations throughout the parameter space. Consequently, we combine HOPS approximations computed at multiple reference values to obtain an accurate global approximation. We construct such global approximations in two ways.

In the first approach, we choose a sample $\{\bar{\boldsymbol{\mu}}_i\}_{i=1}^N$ of the parameter space and then partition the parameter space into a collection of generalized rectangles $\{R_i\}_{i=1}^N$ with $\bar{\boldsymbol{\mu}}_i \in R_i$ for all i. The corresponding piecewise linear HOPS approximation is defined

$$q(\boldsymbol{\lambda}) \approx \tilde{q}(\boldsymbol{\lambda}) = \sum_{i=1}^{N} \left(q(\bar{\boldsymbol{\mu}}_i) + \langle \nabla q(\bar{\boldsymbol{\mu}}_i), (\boldsymbol{\lambda} - \bar{\boldsymbol{\mu}}_i)\rangle\right) \mathbf{1}_{R_i}(\boldsymbol{\lambda}), \tag{6.3}$$

where $\mathbf{1}_{R_i}$ is 1 for $\boldsymbol{\lambda} \in R_i$ and 0 otherwise. By computing the solution at each $\bar{\boldsymbol{\mu}}_i$ and applying the approximation to each piece, we expect this piecewise linear approximation to converge in distribution to the variable $q(\boldsymbol{\lambda})$ as the number of sample points increases.

In the second approach, we use a partition of unity. We let $\{\Lambda_i\}_{i=1}^N$ be a finite open cover of the compact parameter space Λ. A *Lipschitz partition of unity* subordinate to $\{\Lambda_i\}$ is a collection of functions $\{\theta_i\}_{i=1}^N$ with the properties

$$\text{supp}\,(\theta_i) \subset \overline{\Lambda}_i \text{ and } \theta_i \text{ is differentiable on } \Lambda_i, \quad 1 \le i \le N, \tag{6.4}$$

$$\|\theta_i\|_{L^\infty(\Lambda_i)} \le C \text{ and } \|\nabla\theta_i\|_{L^\infty(\Lambda_i)} \le C/\text{diam}\,(\Lambda_i), \quad 1 \le i \le N, \tag{6.5}$$

$$\theta_i \text{ is continuous on } \Lambda \text{ and } \sum_{i=1}^{N} \theta_i(\boldsymbol{x}) = 1, \quad \boldsymbol{x} \in \Omega, \tag{6.6}$$

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

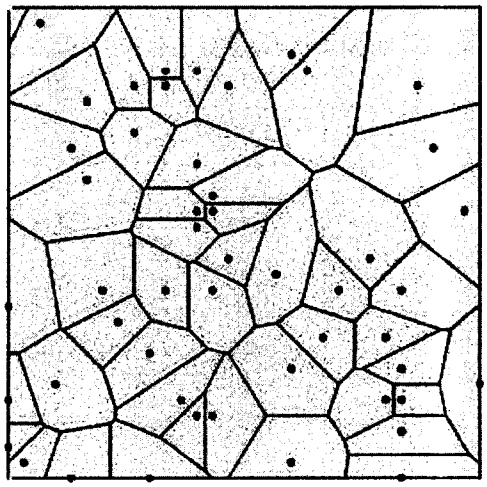

Figure 6.2: The Voronoi tesselation of scattered data in 2-d.

Several partitions of unity satisfying (6.4)-(6.5) exist. We use the partition of unity suggested for the Modified Shepard's Method by Renka [30]. The support of the partition of unity functions are spheres of radius $\mathcal{R}$ centered at the sample points $\{\bar{\boldsymbol{\mu}}_i\}_{i=1}^N$, where $\mathcal{R}$ is chosen so that each sphere contains exactly $\mathcal{N}$ of the sample points. The partition of unity functions are defined as

$$\theta_i(\boldsymbol{x}) = \frac{W_i(\boldsymbol{x})}{\sum_{j=1}^N W_j(\boldsymbol{x})},$$

where

$$W_i(\boldsymbol{x}) = \left(\frac{\max\{\mathcal{R} - \|\boldsymbol{x} - \bar{\boldsymbol{\mu}}_i\|, 0\}}{\mathcal{R}\|\boldsymbol{x} - \bar{\boldsymbol{\mu}}_i\|}\right)^2.$$

Assuming that $\bar{\boldsymbol{\mu}}_i$ is a point inside Λ_i for $i = 1, \ldots, N$, the approximation is defined as

$$q(\boldsymbol{\lambda}) \approx \tilde{q}(\boldsymbol{\lambda}) = \sum_{i=1}^N \left(q(\bar{\boldsymbol{\mu}}_i) + \langle \nabla q(\bar{\boldsymbol{\mu}}_i), (\boldsymbol{\lambda} - \bar{\boldsymbol{\mu}}_i)\rangle\right) \theta_i(\boldsymbol{\lambda}). \tag{6.7}$$

A third approach is to partition the parameter space by a Voronoi tesselation. In this case we let

$$R_i = \{\forall \boldsymbol{x} |\ \|\boldsymbol{x} - \boldsymbol{\mu}_j\| >= \|\boldsymbol{x} - \boldsymbol{\mu}_i\|, i \neq j\},$$

and proceed as above. When computing the HOPS density, it is relatively easy to determine which R_i the point belongs to, and so this algorithm is easily implemented, even in high dimensions. Such a tesselation in two-dimensions is shown in Fig. 6.2.

Chapter 7

RELIABLY ACCURATE PARAMETER SAMPLING (RAPS)

In this chapter we explore the possibility of using the sensitivity derivatives to form an error estimate related to our sampling. This estimate provides the basis for the adaptive sampling which follows in the next chapter.

In HOPS, we use the derivative information computed at each sample point to create a piecewise linear approximation to the distribution of the output value. The goal is to produce a higher order approximation that requires relatively few sample points. Gaining computational efficiency is an important scientific goal when the model is expensive to solve.

Another important scientific goal is to compute information from an approximation of the output distribution that is reliably accurate, i.e., whose accuracy can be quantified. In general, achieving reliable accuracy requires producing an error estimate for the computed information. If we write the one point HOPS approximation as

$$q(\boldsymbol{\lambda}) - q(\bar{\boldsymbol{\mu}}) \approx \langle \nabla q(\bar{\boldsymbol{\mu}}), (\boldsymbol{\lambda} - \bar{\boldsymbol{\mu}}) \rangle, \tag{7.1}$$

we see that the derivative information $\langle \nabla q(\bar{\boldsymbol{\mu}}), (\boldsymbol{\lambda} - \bar{\boldsymbol{\mu}}) \rangle$ provides a local estimate of the error that results from using the sample value $q(\bar{\boldsymbol{\mu}})$ in place of the actual value $q(\boldsymbol{\lambda})$ for $\boldsymbol{\lambda}$ near $\bar{\boldsymbol{\mu}}$. In this section, we describe several

methods that collectively we call *Reliably Accurate Parameter Sampling* methods, or *RAPS*, which uses this derivative information to provide various *a posteriori* estimates and bounds on the error of the information computed from the output distribution generated from a collection of sample values.

7.1 Measuring the error in an approximation of a quantity of interest

We start by discussing ways to measure the error in an approximation $\tilde{q}$ of a quantity of interest q. The approximation $\tilde{q}$ is constructed using a sample $\{\boldsymbol{\mu_i}\}_{i=1}^N$ and the corresponding values $\{q(\boldsymbol{\mu_i})\}_{i=1}^N$, where the $\boldsymbol{\mu}_i$ are chosen in any fashion, either deterministically (as in a quadrature approximation), or randomly (as in a Monte-Carlo computation). We form the approximate surface using either a piecewise constant function of the form

$$\tilde{q}(\boldsymbol{\lambda}) = \sum_{i=1}^{N} q(\boldsymbol{\mu_i}) \mathbf{1}_{R_i}(\boldsymbol{\lambda}),$$

where $\{R_i\}$ is some partition consisting of generalized rectangles such that $\boldsymbol{\mu_i} \in R_i$, or using a partition of unity $\{\theta_i\}$ associated to $\{\bar{\boldsymbol{\mu}}_i\}$ as above to form

$$\tilde{q}(\boldsymbol{\lambda}) = \sum_{i=1}^{N} q(\boldsymbol{\mu_i}) \theta_i(\boldsymbol{\lambda}).$$

Now we consider $\boldsymbol{\lambda}$ to be a random vector of ω on a fixed probability space. By composition, we obtain the approximation

$$q(\omega) = q(\boldsymbol{\lambda}(\omega)) \approx \tilde{q}(\boldsymbol{\lambda}(\omega)).$$

To measure the error in this approximation, it is natural to use the expected value of the error,

$$|E(q(\omega) - \tilde{q}(\omega))|.$$

More generally, we might consider the expected value of the difference of some statistical functions of q and $\tilde{q}$, i.e.,

$$|E(\mathcal{S}(q(\omega)) - \mathcal{S}(\tilde{q}(\omega)))|,$$

where, for example, $\mathcal{S}(s) = s$ yields the mean and $\mathcal{S}(s) = (s - E(q))^2$ yields the variance. Choosing such a statistical function amounts to choosing a statistical "quantity of interest" and, in general, we might choose a collection of such statistical functions to evaluate the accuracy.

A more certain way to guarantee that the approximate distribution converges to the true distribution is to control the L^1 norm. In other words, if

$$E(|q(\omega) - \tilde{q}(\omega)|) \tag{7.2}$$

tends to zero, then this implies convergence in probability and in distribution [2]). This in turn implies that cumulative distribution function F_q of q can be approximated arbitrarily well by $F_{\tilde{q}}$. By recovering the cumulative distribution function, we can compute any desired statistic, e.g., means and moments, that might be used to characterize the accuracy of the approximate distribution.

7.2 Viewing a sample as a piecewise constant approximation

Perhaps the most straightforward approach is based on interpreting a given sample, which may or may not be generated by a Monte-Carlo computation, as a piecewise constant approximation. We assume that there is a partition of the compact parameter space Λ into N generalized rectangles $\{R_i\}_{i=1}^N$ where $\bar{\boldsymbol{\mu}}_i$ is a point inside R_i and $q(\bar{\boldsymbol{\mu}}_i)$ is calculated for

$i = 1, \ldots, N$. This results in a piecewise constant approximation,

$$\tilde{q}(\boldsymbol{\lambda}) = \sum_{i=1}^{N} q(\bar{\boldsymbol{\mu}}_i) \mathbf{1}_{R_i}(\boldsymbol{\lambda}), \tag{7.3}$$

In the case that the sample is generated by a Monte-Carlo computation, this interpretation is certainly different than the standard view of a Monte-Carlo approximation. For example, consider that the Monte-Carlo approximation $\sum_{i=1}^{N} g(\omega_i)/N$ for an average value $\int_\Omega g(\omega)\,\mathrm{d}\omega \,/\, \mathrm{Vol}(\Omega)$ weights the sample values $\{g(\omega_i)\}$ equally. If we consider the sample values as defining a piecewise constant approximation, the natural approximation to the average value is

$$\sum_{i=1}^{N} g(\omega_i) \frac{\mathrm{Vol}\,(R_i)}{\mathrm{Vol}\,(\Omega)}.$$

Since the points $\{\omega_i\}$ are not uniformly spaced with probability 1, then the sample values $\{g(\omega_i)\}$ are not weighted equally when computing the average value. Nominally, the new interpretation yields a better asymptotic convergence rate, i.e., $O(1/N)$ versus $O(1/\sqrt{N})$, however the constants in the deterministic error bound grow rapidly as the dimension of the parameter space increases. This is why the Monte-Carlo interpretation is preferred in high dimensions.

Nonetheless, we show below that interpreting a sample as a piecewise constant approximation can be useful in some circumstances. Assuming that the error is measured by the expected value of the difference, we can represent it as a sum of "element contributions",

$$\begin{aligned} E(q(\omega) - \tilde{q}(\omega)) &= \int_\Omega \big(q(\boldsymbol{\lambda}(\omega)) - \tilde{q}(\boldsymbol{\lambda}(\omega))\big)\,\mathrm{d}P(\omega) = \int_{\mathbb{R}^d} \big(q(\boldsymbol{z}) - \tilde{q}(\boldsymbol{z})\big)\,\mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}) \\ &= \sum_i \int_{R_i} \big(q(\boldsymbol{z}) - \tilde{q}(\boldsymbol{z})\big)\,\mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}), \end{aligned}$$

where $\mu_{\boldsymbol{\lambda}}$ is the measure induced by $\boldsymbol{\lambda}$, i.e., $\mu_{\boldsymbol{\lambda}}(A) = P(\boldsymbol{\lambda} \in A)$ for Borel sets A. Note that the distribution of the input values weights the evaluation of the resulting changes in q.

Using a Taylor expansion in each of the R_i as in HOPS, we obtain

$$q(\boldsymbol{\lambda}) - \tilde{q}(\boldsymbol{\lambda}) = q(\boldsymbol{\lambda}) - q(\bar{\boldsymbol{\mu}}_i) = \langle \nabla q(\boldsymbol{\xi}_i), (\boldsymbol{\lambda} - \bar{\boldsymbol{\mu}}_i) \rangle,$$

for some unknown $\boldsymbol{\xi}_i$ on the line between $\boldsymbol{\lambda}$ and $\bar{\boldsymbol{\mu}}_i$. The error reduces to

$$\sum_{i=1}^{N} \int_{R_i} \langle \nabla q(\boldsymbol{\xi}_i), (\boldsymbol{z} - \bar{\boldsymbol{\mu}}_i) \rangle \, \mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}).$$

In practice, we compute an estimate using the quadrature formula

$$\mathcal{E}^{pc} = \sum_{i=1}^{N} \int_{R_i} \langle \nabla q(\bar{\boldsymbol{\mu}}_i), (\boldsymbol{z} - \bar{\boldsymbol{\mu}}_i) \rangle \, \mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}). \tag{7.4}$$

The analogous argument for the error measured using the expected value of some statistical function or the L^1 norm yields

$$\mathcal{E}^{pc} = \sum_{i=1}^{N} \int_{R_i} \mathcal{S}'(q(\bar{\boldsymbol{\mu}}_i)) \langle \nabla q(\bar{\boldsymbol{\mu}}_i), (\boldsymbol{z} - \bar{\boldsymbol{\mu}}_i) \rangle \, \mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}) \tag{7.5}$$

and

$$\mathcal{E}^{pc} = \sum_{i=1}^{N} \int_{R_i} |\langle \nabla q(\bar{\boldsymbol{\mu}}_i), (\boldsymbol{z} - \bar{\boldsymbol{\mu}}_i) \rangle| \, \mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}), \tag{7.6}$$

respectively.

$\mathcal{E}^{pc}$ is an approximation in the sense that if ∇q is continuous, then a sequence of such approximations computed on partitions with rectangles of decreasing size will converge to the true estimate. Unfortunately, this approach is adversely affected by the dimension of the parameter space. For example, the distances between the midpoint of a generalized rectangle to the corners increases as the dimension increases. Thus for geometric reasons, using a single sampled value to represent the behavior of an output distribution over a generalized rectangle becomes problematic for large dimensions.

7.2.1 A Scalar Example

To illustrate this approach, we consider the scalar example (6.1) with $x_0 = 1$ and λ taken uniformly in $[-0.5, 0.5]$. The results for the estimate of the error in the L^1 norm of $q(\boldsymbol{\lambda}) = x(t, \lambda)$ at $T = 10$ are given in Fig. 7.1. The systematic underestimation is a consequence of linearization and the concavity of the exponential. We also plot results for the L^1 norm of the

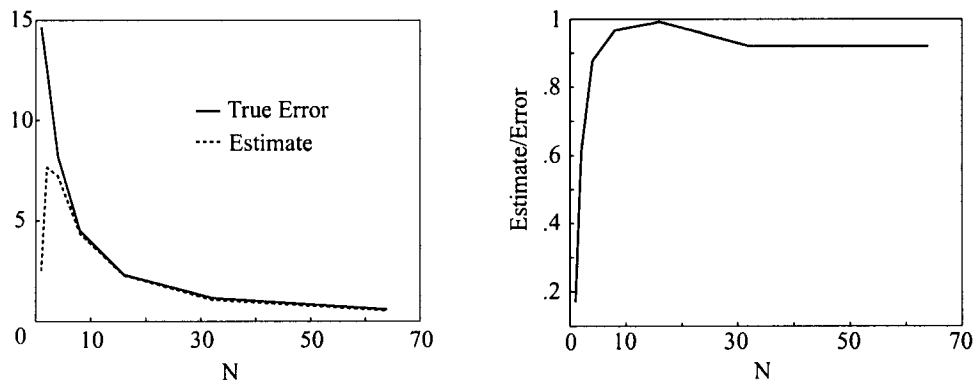

Figure 7.1: On the left, we plot the actual errors and the error estimates $\mathcal{E}^{pc}$ in the L^1 norm for the scalar example (6.1) with range of sample points N at $T = 10$. On the right, we plot the ratio of the estimate to the error.

error in the second moment, $E[|q(\boldsymbol{\lambda})^2 - \tilde{q}(\boldsymbol{\lambda})^2|]$, and corresponding estimate

$$\mathcal{E}^{pc} = \sum_{i=1}^{N} \int_{R_i} |2q(\bar{\boldsymbol{\mu}}_i)q'(\bar{\boldsymbol{\mu}}_i)(\boldsymbol{z} - \bar{\boldsymbol{\mu}}_i)| \, \mathrm{d}\mu_\lambda(\boldsymbol{z}) \tag{7.7}$$

at $T = 5$ in Fig. 7.2.

7.3 Computing an approximate error function using a partition of unity

In a second approach, we attempt to avoid the adverse effects of dimension by using a partition of unity to create both an approximation from a given sample and an approximate error function corresponding to the

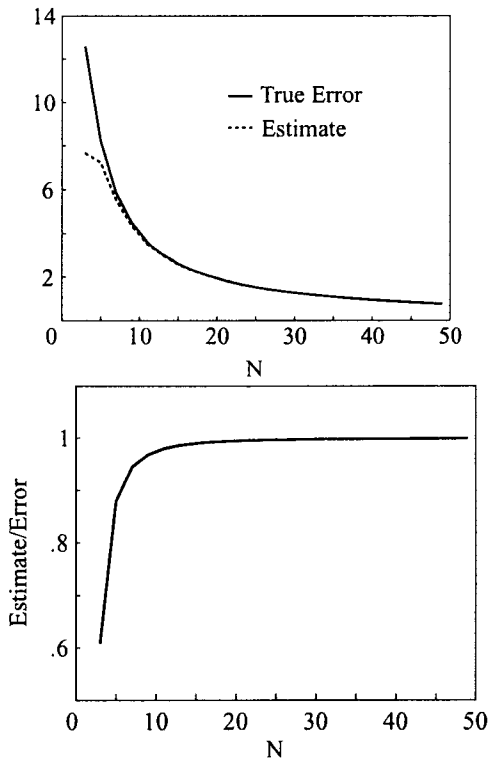

Figure 7.2: On the left, we plot the actual values and the error estimates $\mathcal{E}^{pc}$ for the L^1 norm of the second moment for the scalar example (6.1) with range of sample points N at $T = 5$. On the right, we plot the ratio of the estimate to the error.

approximation. As in Sec. 6.2, we use Renka's partition of unity. The approximation is defined as

$$\tilde{q}(\lambda) = \sum_{i=1}^{N} q(\bar{\boldsymbol{\mu}}_i)\theta_i(\lambda), \quad \lambda \in \Lambda. \tag{7.8}$$

Assuming first that the expected value of the error is to be estimated, we use (6.6) to write

$$E(q(\omega) - \tilde{q}(\omega)) = \sum_{i=1}^{N} E\big((q(\omega) - q(\bar{\boldsymbol{\mu}}_i))\theta_i(\omega)\big).$$

Now using (7.1), we estimate

$$\begin{aligned} E\big((q(\omega) - q(\bar{\boldsymbol{\mu}}_i))\theta_i(\omega)\big) &= \int_{\Lambda_i} (q(\boldsymbol{z}) - q(\bar{\boldsymbol{\mu}}_i))\theta_i(\boldsymbol{z})\,\mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}) \\ &\approx \int_{\Lambda_i} \langle \nabla q(\bar{\boldsymbol{\mu}}_i), (\boldsymbol{z} - \bar{\boldsymbol{\mu}}_i)\rangle \theta_i(\boldsymbol{z})\,\mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}). \end{aligned}$$

We obtain the estimate

$$\mathcal{E}^{pou} = \sum_{i=1}^{N} \int_{\Lambda_i} \langle \nabla q(\bar{\boldsymbol{\mu}}_i), (\boldsymbol{z} - \bar{\boldsymbol{\mu}}_i)\rangle \theta_i(\boldsymbol{z})\,\mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}). \tag{7.9}$$

The analogous argument for the error measured using the expected value of some statistical function or the L^1 norm yields

$$\mathcal{E}^{pou} = \sum_{i=1}^{N} \int_{\Lambda_i} \langle \mathcal{S}'(q(\bar{\boldsymbol{\mu}}_i))\nabla q(\bar{\boldsymbol{\mu}}_i), (\boldsymbol{z} - \bar{\boldsymbol{\mu}}_i)\rangle \theta_i(\boldsymbol{z})\,\mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}) \tag{7.10}$$

and

$$\mathcal{E}^{pou} = \sum_{i=1}^{N} \int_{\Lambda_i} |\langle \nabla q(\bar{\boldsymbol{\mu}}_i), (\boldsymbol{z} - \bar{\boldsymbol{\mu}}_i)\rangle \theta_i(\boldsymbol{z})|\,\mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}), \tag{7.11}$$

respectively.

Since

$$\begin{aligned} \int_{\Lambda_i} (\boldsymbol{z} - \bar{\boldsymbol{\mu}}_i)\theta_i(\boldsymbol{z})\,\mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}) &= \int_{\Lambda} (\boldsymbol{z} - \bar{\boldsymbol{\mu}}_i)\theta_i(\boldsymbol{z})\,\mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}) \\ &= \int_{\Omega} (q(\boldsymbol{\lambda}(\omega)) - \bar{\boldsymbol{\mu}}_i)\theta_i(\boldsymbol{\lambda}(\omega))\,\mathrm{d}P(\omega) \\ &= E((\boldsymbol{\lambda} - \bar{\boldsymbol{\mu}}_i)\theta_i), \end{aligned}$$

in the case of the expected value of a statistical function, we can rewrite the second estimate as

$$\mathcal{E}^{pou} = \sum_{i=1}^{N} \langle \mathcal{S}'(q(\bar{\boldsymbol{\mu}}_i)) \nabla q(\bar{\boldsymbol{\mu}}_i), E((\boldsymbol{\lambda} - \bar{\boldsymbol{\mu}}_i)\theta_i) \rangle. \tag{7.12}$$

7.3.1 A Scalar Example

To illustrate this approach, we again consider the scalar example (6.1) with $x_0 = 1$ and λ taken uniformly in $[-0.5, 0.5]$. The results for the estimate of the error in the L^1 norm of $q(\boldsymbol{\lambda}) = x(t, \lambda)$ at $T = 10$ are given in Fig. 7.3. We use the functions $\exp(-10N(x-\mu_i)^2)$ to generate the partition functions. We plot results for the L^1 norm of the error in the second moment

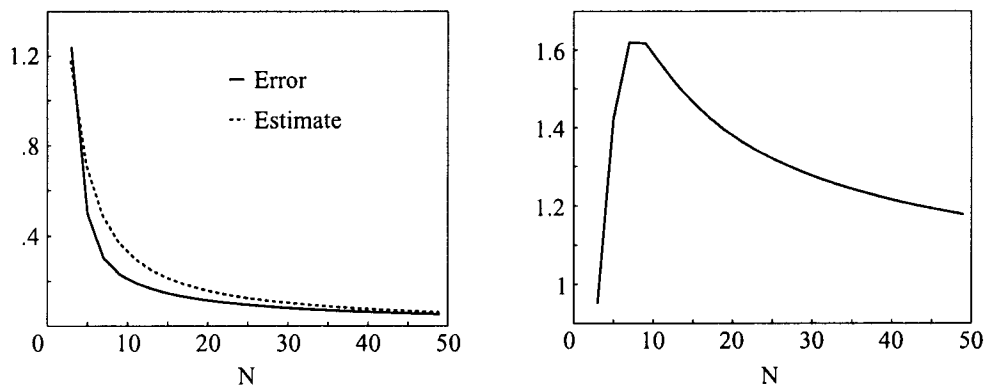

Figure 7.3: On the left, we plot the actual errors and the error estimates $\mathcal{E}^{pc}$ in the L^1 norm for the scalar example (6.1) with range of sample points N at $T = 10$. On the right, we plot the ratio of the estimate to the error.

and corresponding estimate (7.7) at $T = 5$ using $\exp(-15N(x - \mu_i)^2)$ to generate the partition functions in Fig. 7.4.

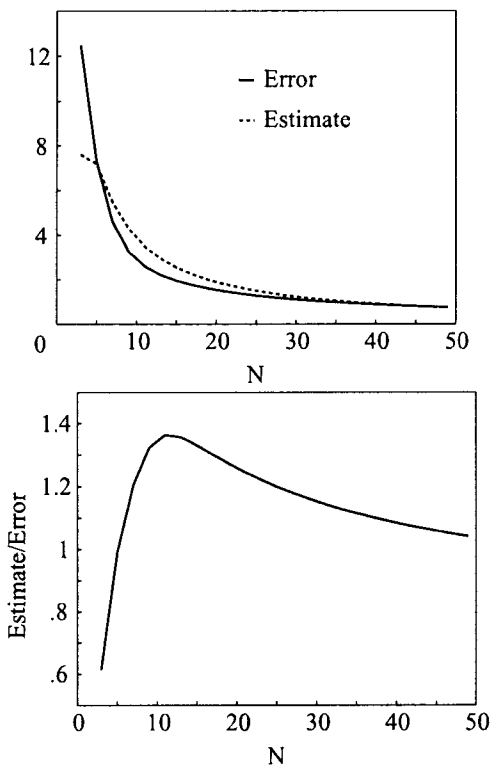

Figure 7.4: On the left, we plot the actual values and the error estimates $\mathcal{E}^{pc}$ for the L^1 norm of the second moment for the scalar example (6.1) with range of sample points N at $T = 5$. On the right, we plot the ratio of the estimate to the error.

Chapter 8

FAST ADAPTIVE PARAMETER SAMPLING (FAPS)

The error estimates provided in the previous chapter may be used to sample adaptively. This chapter explores this technique.

RAPS achieves increased reliability at the cost of computing derivative information at each sample point. The additional cost is justified if reliable accuracy is an important scientific concern. Moreover, we observe that obtaining an accurate estimate of the error for a given sample opens up the possibility of optimizing the sample process in order to reach a desired accuracy with minimal computational cost. This can offset the overhead associated with computing the estimate to the extent that the optimized computation, including the cost of error estimation, is still significantly faster than a non-optimized computation.

In general, constructing an optimal sample set is a complicated nonlinear problem that is difficult to solve. We borrow ideas from adaptive finite elements to construct two *Fast Adaptive Sampling Procedures*, or FAPS, for finding an approximate solution. The goal of the adaptive strategy is to find a set of sample points that has approximately minimal size yet yields the desired accuracy in the information computed from the corresponding approximation to the output distribution. The adaptive strategy works

in an iterative fashion. Given a current set of sample points, we examine the local contributions to the error estimate for the sample and choose additional sample points in regions in which the contributions to the error are estimated to be largest in order to obtain a new sample set for which, presumably, the error is smaller.

Constructing a FAPS method requires several choices. First, we must choose an error estimate or bound to guide the enrichment decision. Using an optimization approach requires an estimate or bound that can be written as a sum over local contributions with nonnegative summands. The estimates (7.4)-(7.5) and (7.9)-(7.10) might be used to determine that the computed information is not sufficiently accurate, but they cannot be used for standard adaptive error control because they allow cancellation of errors through the domain. Once the decision to enrich the current sample is made, we use a bound derived from the estimates (7.6) or (7.11) for the L^1 norm of the error to choose where to place additional sample points.

Another important choice is a procedure for adding additional sample points to a given sample set. We describe two approaches below.

The choice of the initial set of sample points is third important decision. One aspect is the density of the initial sample. On one hand, the standard approach is to use a relatively coarse set in order to increase the possibility of achieving a nearly optimal final sample set. On the other hand, using too coarse an initial sample set can mean that the error estimate fails to recognize significant behavior. For example, in the control problem example below, we find that choosing an initial sample set that is too coarse leads to underestimation of the tails of the distribution. In most of the examples in this paper, however, a single sample point centered in the parameter space was sufficient to seed the computation successfully.

8.1 Adaptive sampling via a standard h-refinement approach

The first approach we describe is analogous to the standard approach to adaptive error control for finite element methods. We use one or more of the estimates (7.4)-(7.6) to decide if a given sample set needs to be enriched. If enrichment is needed, we use a bound derived from (7.6) to choose additional sample points. We add points using a strategy that is closely analogous to the standard h-refinement strategy in adaptive finite element methods. We define $\mathcal{E}_i^{pc}$ to be the approximate contribution to the error bound from rectangle R_i, i.e.,

$$\mathcal{E}_i^{pc} = \int_{R_i} |\langle \nabla q(\bar{\boldsymbol{\mu}}_i), (\boldsymbol{z} - \bar{\boldsymbol{\mu}}_i) \rangle| \, \mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}).$$

The adaptive strategy is to refine some fraction of the rectangles on which $\mathcal{E}_i^{pc}$ is largest. We refine along one dimension at a time since q may be very sensitive to changes in one parameter, but not in others. To measure the contribution to the error bound from each dimension, we define $\mathcal{E}_{i,k}^{pc}$, $k = 1, \ldots, d$, by

$$\mathcal{E}_{i,k}^{pc} = \int_{R_i} |\partial_{\lambda_k} q(\bar{\boldsymbol{\mu}}_i)(z^k - \mu_i^k)| \, \mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}), \tag{8.1}$$

where $\boldsymbol{z} = (z^1, \cdots, z^d)^\top$ and $\bar{\boldsymbol{\mu}}_i = (\mu_i^1, \cdots, \mu_i^d)^\top$ so

$$\mathcal{E}^{pc} = \sum_{i=1}^{N} \mathcal{E}_i^{pc} \leq \sum_{i=1}^{N} \sum_{k=1}^{d} \mathcal{E}_{i,k}^{pc}.$$

For a rectangle where $\mathcal{E}_i^{pc}$ is large enough for refinement, we find the maximum contribution $\mathcal{E}_{i,k}^{pc}$, $k = 1, \ldots, d$ and we divide the rectangle along this dimension. Since the value at the center of the rectangle is known, we split in thirds along this dimension, compute the values on the two new rectangles, and iterate. See Fig. 8.1 for an illustration.

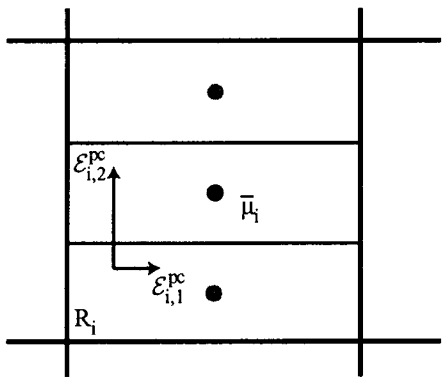

Figure 8.1: Illustration of partition refinement.

Note that since the computations can proceed entirely in parallel, it is possible to divide all rectangles whose error contribution is some number of standard deviations above the mean error contribution and refine these simultaneously.

Given an error tolerance TOL, we repeat the iteration until the selection of estimates (7.4)-(7.6) used to judge accuracy of the computed information are (all) bounded by TOL. At that point, we compute the distribution function of the simple function $q(\bar{\boldsymbol{\mu}}_i), i = 1, \ldots, N$. The cumulative distribution function of the approximation may be recovered from (7.3). We sort the $\tilde{q}(\bar{\boldsymbol{\mu}}_i)$ in ascending order, and the jump of the resulting cumulative distribution function between $\tilde{q}(\bar{\boldsymbol{\mu}}_i)$ and $\tilde{q}(\bar{\boldsymbol{\mu}}_{i+1})$ is simply $\mu_{\boldsymbol{\lambda}}(R_{i+1})$.

For this approach, we use a uniform initial partition of the parameter space. This is reasonable when the parameter space does not have high dimension. However, the number of sample points in a uniform partition increases exponentially as the dimension of the parameter space increases. A natural idea is to use a hybrid method in which a Monte-Carlo approximation is used to obtain an initial sample set, and then a FAPS computation

is used after that in order to obtain a reliably accurate approximation in an efficient way. We will explore this idea in later work.

We illustrate the rectangular FAPS in Sections 9.1–9.4 below.

8.2 Adaptive sampling according to a density representing the error

Using a partition of unity to generate an approximate error function suggests another approach to adaptive sampling. Suppose that we have an initial sample set, which might be the result of a Monte-Carlo computation, and we use a partition of unity to generate one or more of the estimates (7.9)-(7.11). We can construct a density function using the element contributions in the estimate that describes how well the nonlinear operator is sampled throughout various regions in parameter space. On the next iteration in the adaptive procedure, we could add additional sample points, perhaps using a Monte-Carlo method, according to this density function.

Using Renko's partition of unity, the Integral Mean Value Theorem implies that the approximation $\mathcal{E}^{pou}$ to the expected value is itself approximated by

$$\mathcal{E}^{pou}_{loc}(\boldsymbol{\lambda}) = \sum_{i=1}^{N} \langle \nabla q(\bar{\boldsymbol{\mu}}_i), \boldsymbol{\lambda} - \bar{\boldsymbol{\mu}}_i \rangle \theta_i(\boldsymbol{\lambda}) f_{\boldsymbol{\lambda}}(\boldsymbol{\lambda})$$

at each point λ, where $f_{\boldsymbol{\lambda}}$ is the density corresponding to $\bar{\boldsymbol{\mu}}_{\boldsymbol{\lambda}}$.

We use the error estimate $|\mathcal{E}^{pou}_{loc}|$ as a sampling density to select additional points in such a way that there is increased sampling in regions where this function is relatively large. To do this, we use a Markov Chain Monte-Carlo method [20] that creates a large sample of points with distribution $|\mathcal{E}^{pou}_{loc}|$, from which we draw a few points for evaluation. Once the function has been sufficiently sampled, we draw a large selection of points from the

distribution of $\boldsymbol{\lambda}$, apply the partition of unity approximation $\tilde{q}(\boldsymbol{\lambda})$ to these points, and finally approximate the density of the resulting sample using the techniques described above.

We illustrate this adaptive approach in Sec. 9.2 below.

8.2.1 Creating a density that indicates regions with insufficient sampling

There are other ways to create a density function. Consider the problem of representing the function $q(\boldsymbol{\lambda})$ accurately pointwise, for which there is a local error estimate,

$$|q(\boldsymbol{\lambda}) - q(\bar{\boldsymbol{\mu}}_i)| \lesssim |\nabla q(\bar{\boldsymbol{\mu}}_i)|\,|\boldsymbol{\lambda} - \bar{\boldsymbol{\mu}}_i|.$$

The values of $q(\boldsymbol{\lambda})$ are approximated by $q(\bar{\boldsymbol{\mu}}_i)$ to within a tolerance TOL for all $\boldsymbol{\lambda}$ satisfying

$$|\boldsymbol{\lambda} - \bar{\boldsymbol{\mu}}_i| \lesssim \frac{TOL}{|\nabla q(\bar{\boldsymbol{\mu}}_i)|},$$

i.e., for $\boldsymbol{\lambda}$ in the ball of radius $r_i = TOL/|\nabla q(\bar{\boldsymbol{\mu}}_i)|$ centered at $\bar{\boldsymbol{\mu}}_i$. We set V_i to be the volume of the ball of radius r_i centered at $\bar{\boldsymbol{\mu}}_i$,

$$V_i = \frac{\pi^{d/2} r_i^d}{\Gamma(d/2+1)},$$

and set $V = \left(\sum_{i=1}^N V_i^{-1}\right)^{-1}$, so $\sum_{i=1}^N \frac{V}{V_i} = 1$. We interpolate the set of values $\{V/V_i\}$ at the points $\{\bar{\boldsymbol{\mu}}_i\}$ to obtain a density function that describes the degree to which the function is represented pointwise by its values at $\{\bar{\boldsymbol{\mu}}_i\}$. Note that the dimension-dependent constants in V_i cancel in the ratio V/V_i. In fact,

$$\frac{V}{V_i} = \frac{|\nabla q(\bar{\boldsymbol{\mu}}_i)|^d}{\sum_{j=1}^N |\nabla q(\bar{\boldsymbol{\mu}}_j)|^d} = \left(\sum_{j=1}^N \left(\frac{|\nabla q(\bar{\boldsymbol{\mu}}_j)|}{|\nabla q(\bar{\boldsymbol{\mu}}_i)|}\right)^d\right)^{-1}. \tag{8.2}$$

As d increases, the difference in the values of the density function corresponding to large and small values of $|\nabla q(\bar{\mu}_i)|$ is exaggerated. In contrast, simply removing the balls from the parameter space before selecting the additional sample points has decreasing effect on the selection process as the dimension increases.

To illustrate, we consider the function

$$q(\boldsymbol{\lambda}) = e^{-.5(\lambda_1^2+\lambda_2^2)} + .01\lambda_1^3 + .01\lambda_2^2 \tag{8.3}$$

on $[0,4] \times [0,4]$. We plot q in Fig. 8.2 for a uniform 21×21 set of sample points. In Fig. 8.3, we plot the regions in which the function is approximated

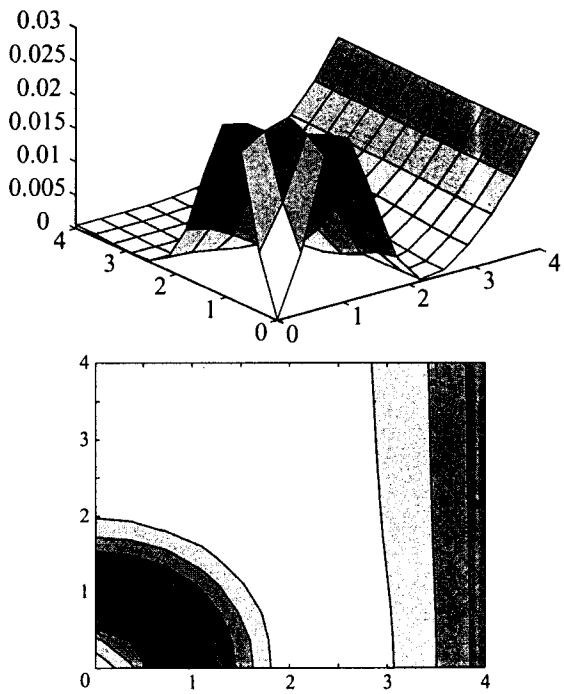

Figure 8.2: Surface and contour plots of function q (8.3) corresponding to $TOL = .01$ and a uniform set of 12×12 sample points. Dark areas indicate regions in which q is not approximated pointwise as well.

pointwise to within a tolerance of $TOL = .01$ for three sets of uniformly spaced sample points.

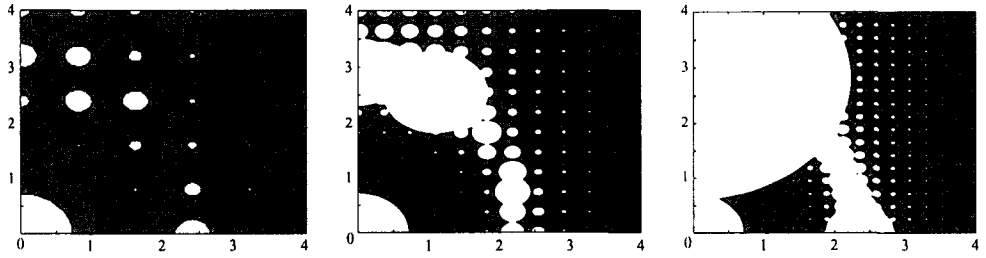

Figure 8.3: Plots of the regions in which the function is approximated pointwise to within a tolerance of $TOL = .01$ for three sets of uniformly spaced sample points. From left to right, the sets have 6×6, 12×12, and 18×18 points respectively. Shaded regions indicate areas where q is not sufficiently well approximated.

This approach extends directly to the estimate (7.11) after choosing a partition of unity. For example, suppose that Λ_i is a ball of radius r_i centered at $\bar{\boldsymbol{\mu}}_i$ with volume V_i. We can bound

$$\int_{\Lambda_i} |(\boldsymbol{z} - \bar{\boldsymbol{\mu}}_i)\theta_i(\boldsymbol{z})| \, \mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}) \lesssim C r_i \mu_{\boldsymbol{\lambda}}(\Lambda_i) \approx C r_i f_{\boldsymbol{\lambda}}(\boldsymbol{\lambda}_i) V_i,$$

where $\boldsymbol{\lambda}_i$ is some point in Λ_i, C is a constant that depends on the construction of θ_i, and

$$|\mathcal{E}^{pou}| \leq \sum_{i=1}^{N} C |\nabla q(\bar{\boldsymbol{\mu}}_i)| f_{\boldsymbol{\lambda}}(\boldsymbol{\lambda}_i) r_i V_i.$$

We seek to choose the minimum number of points such that

$$\sum_{i=1}^{N} C |\nabla q(\bar{\boldsymbol{\mu}}_i)| r_i f_{\boldsymbol{\lambda}}(\boldsymbol{\lambda}_i) V_i \leq TOL.$$

The standard "Principle of Equidistribution" argument implies that the optimal result is obtained when the summands are equal, which yields a formula for r_i. Defining $V = \left(\sum_{i=1}^{N} V_i^{-1}\right)^{-1}$ as above, we find that in this case,

$$\frac{V}{V_i} = \frac{(|\nabla q(\bar{\boldsymbol{\mu}}_i)| f_{\boldsymbol{\lambda}}(\boldsymbol{\lambda}_i))^{d/(d+1)}}{\sum_{j=1}^{N} (|\nabla q(\bar{\boldsymbol{\mu}}_j)| f_{\boldsymbol{\lambda}}(\boldsymbol{\lambda}_j))^{d/(d+1)}} = \left(\sum_{j=1}^{N} \left(\frac{|\nabla q(\bar{\boldsymbol{\mu}}_j)| f_{\boldsymbol{\lambda}}(\boldsymbol{\lambda}_j)}{|\nabla q(\bar{\boldsymbol{\mu}}_i)| f_{\boldsymbol{\lambda}}(\boldsymbol{\lambda}_i)} \right)^{d/(d+1)} \right)^{-1}. \tag{8.4}$$

Using this, we produce a density function as described in Sec. 8.2.1. Note that in sharp contrast to (8.2), there is decreasing effect on the values of the density as d increases. This suggests that the evaluation of q should not be treated as a purely pointwise interpolation problem when the parameter space has large dimension.

We illustrate this adaptive approach in Sec. 9.2 below.

Chapter 9

APPLICATION TO ODE'S

We use the methods outlined above in a variety of examples involving ordinary differential equations.

9.1 The Control Problem for a Pendulum

We consider the problem of controlling a pendulum, consisting of a rigid, massless arm attached to a frictionless pivot at one end and a mass at the other end, such that it remains in the (unstable) upright position by applying a torque $u(t)$ at the joint. With the pivot located at the origin, we use the "inverted" angle x measured from the positive vertical axis to locate the pivot. We assume $x(t)$ and $\dot{x}(t)$ can be measured and use *proportional-derivative* control $u(t) = -\alpha x(t) - \beta \dot{x}(t)$. The complete system is then

$$\begin{cases} m\ddot{x} - mg\sin(x) = -\alpha x(t) - \beta \dot{x}(t), & t \geq 0, \\ x(0) = x_0, \dot{x}(0) = \dot{x}_0, \end{cases} \tag{9.1}$$

where m is the pendulum mass and g the gravitational constant [32].

The goal is to choose parameters α and β so that the pendulum returns to the upright position with $x(t) \to 0$ and $\dot{x} \to 0$. We can obtain an approximate control strategy for $x \approx 0$ by the linearization $\sin(x) \approx x$. This gives a second order constant coefficient equation with characteristic roots

$$z = \frac{-\beta \pm \sqrt{\beta^2 - 4(m\alpha - m^2 g)}}{2m}$$

If $\beta > 0$ and $\alpha > mg$, the real part of these roots are negative, and the problem is stable. Further, if we choose

$$\beta > 2\sqrt{m(\alpha - mg)}$$

the roots are both real and the solution decays exponentially to the desired equilibrium $x = 0, \dot{x} = 0$. This avoids oscillations in the pendulum.

We wish to study the return of this system to equilibrium for a variety of initial conditions and masses. To this end, we consider the behavior of the system for a distribution for the parameters $m, \phi_0, \dot{\phi}_0$. In Fig. 9.1, we show

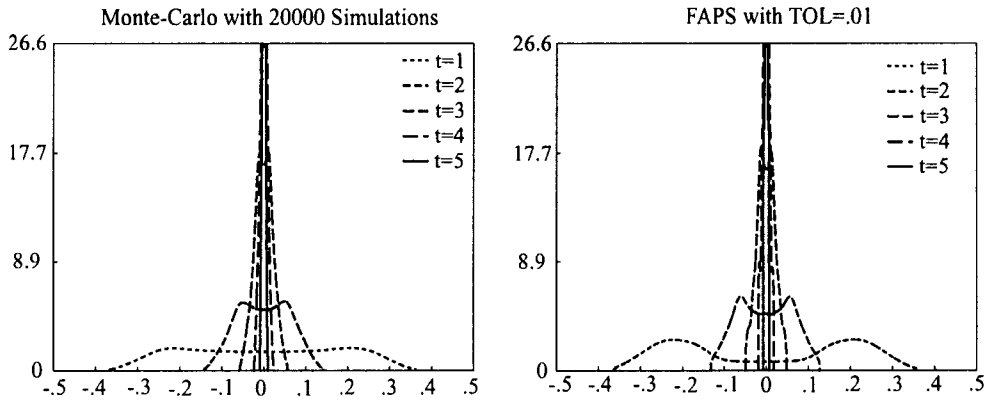

Figure 9.1: The density of $x(t)$ for several values of t computed using a 20000 point Monte Carlo simulation (left) and a rectangular FAPS approximation with TOL=.01 (right). $(x_o, \dot{x}_o)$ is uniform on $(-\pi/4, \pi/4) \times (-0.2, 0.2)$, and m is uniform on $(0.9, 1.1)$, $\alpha = 14.8, \beta = 5.5$. The number of simulations used in the FAPS computation are 586 at $t = 1$, 54 at $t = 2$, and 18 at $t = 3$, 4, and 5. We use a HOPS approximation on the last FAPS mesh.

the results for the Monte-Carlo and rectangular FAPS computations. The results suggest that the solutions converge to the desired equilibrium. Note that the number of simulations needed to meet a fixed TOL in the FAPS computation decreases as time increases, since the solutions are converging to a fixed point.

9.2 The Schaefer Model for a Harvested Fish Population

The Schaefer model describes the evolution of a fish population that is harvested at a constant level of effort, i.e., by the same number of boats each day. The fish population x is assumed to grow according to a logistic mechanism and the rate of harvesting is the effort times the number of fish. Thus,

$$\begin{cases} \dot{x}(t;\boldsymbol{\lambda_1}) = \frac{R}{K}\left(K - x(t;\boldsymbol{\lambda_1})\right)x(t;\boldsymbol{\lambda_1}) - Hx(t;\boldsymbol{\lambda_1}), & t > 0, \\ x(0;\boldsymbol{\lambda_1}) = x_0, \end{cases} \tag{9.2}$$

where $\boldsymbol{\lambda_1} = (R, K, H)^\top$. We consider the initial conditions to be fixed. A straightforward calculation shows

$$D_{\boldsymbol{\lambda_1}} f(x;\boldsymbol{\lambda_1}) = \left(\frac{-x(x-K)}{K}, \frac{x^2 R}{K^2}, -x\right),$$
$$D_x f(x;\boldsymbol{\lambda_1}) = -x\frac{R}{K} + R\left(1 - \frac{x}{K}\right) - H.$$

We first examine the case $H = 0$, which is the logistic equation. We take data $\psi(s) = \delta(s-3)$ to estimate the value of the solution at time $t = 3$. We set $x_0 = 1$. Plotting the norm of the generalized Green's function,

$$\int_0^3 |\phi(s)|\, \mathrm{d}s,$$

for various R, K, we see that it is large in the area near $R = 0$, See Fig. 9.2, so we expect that if the parameter distribution covers this area refinement will be needed there. In Fig. 9.4, we plot the final adaptive grid. In Fig. 9.3, we plot the HOPS and rectangular FAPS approximations.

For the case $H \neq 0$, the system has two fixed points $\tilde{x}_1 = 0$ and $\tilde{x}_2 = K(1 - \frac{H}{R})$. The stability is determined by the derivative $\partial_x f$, which is $R - H$ at $\tilde{x}_1$ and $H - R$ at $\tilde{x}_2$. As $t \to \infty$, the distribution of the solutions should split as they are drawn to different fixed points. A bifurcation

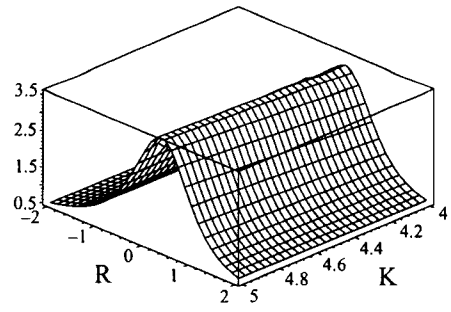

Figure 9.2: Norm of the Green's function $\phi(s)$ for the data $\psi(s) = \delta(s-3)$, $x_0 = 1$. It is largest near $R = 0$, where small changes in R cause large changes in the accumulated growth at $t = 3$.

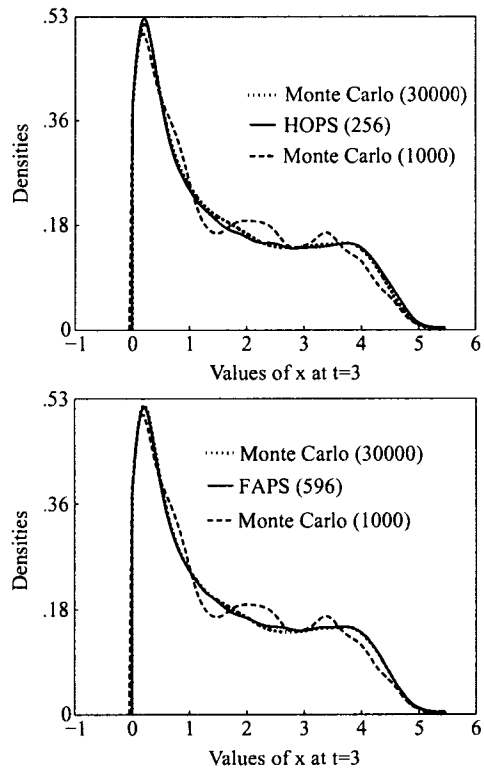

Figure 9.3: Approximate distributions of solutions to the (9.2) with $H = 0$ at $t = 3$. (R, K) are independent and $N(0.1, 0.5)$, $N(4.5, 0.1)$ (truncated) respectively. On the left, we plot the results of HOPS using 256 simulations. On the right, we plot the results of a rectangular FAPS using 596 simulations. For comparison, we show the results for a Monte-Carlo compuations using 30000 and 1000 simulations.

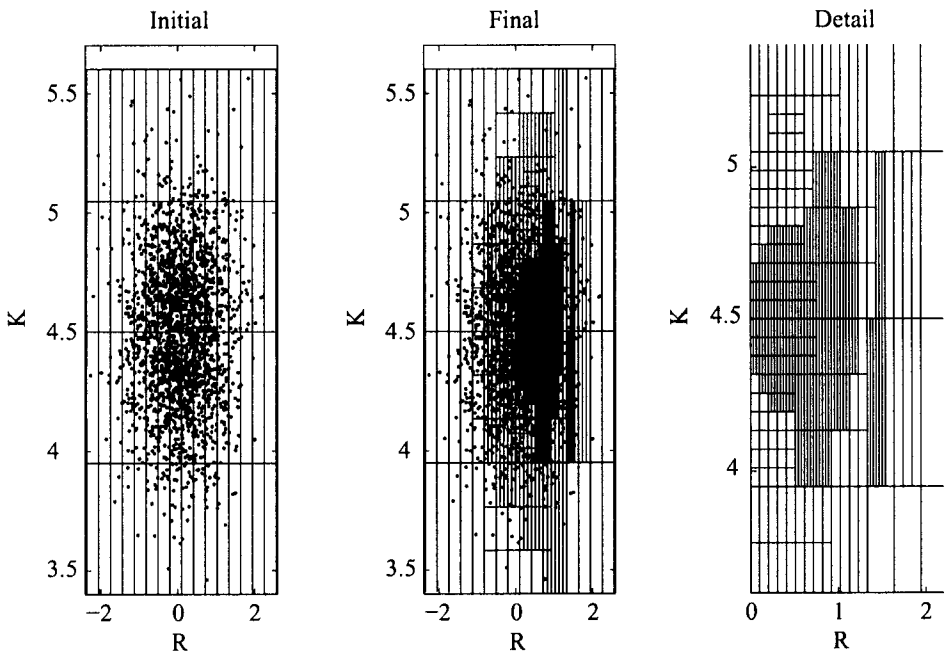

Figure 9.4: Initial and final adaptive meshes for the logistic example for the rectangular FAPS. The plotted points show the distribution of the input parameters. The FAPS refinement near $R = 0$ is apparent.

occurs at $R = H$; $\tilde{x}_1$ is stable and $\tilde{x}_2$ unstable for $R < H$ and the opposite when $R > H$. Hence as $t \to \infty$, the measure of the solutions that tend to zero should be the measure of $[R < H]$, and the rest of the solutions should be distributed as the random variable $K(1 - H/R)$ (their eventual limiting value). The case $R = H$ can be ignored when this line has measure zero in the parameter distribution. In Fig. 9.5, the asymptotic ($t = \infty$), Monte-Carlo and rectangular FAPS computations are compared. We notice discrepancies where R is small, which is expected since the system approaches equilibrium much slower in this case.

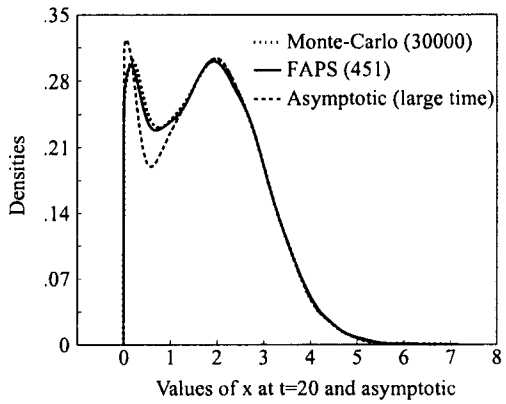

Figure 9.5: Comparison of 30000 point Monte-Carlo, 451 point rectangular FAPS, and asymptotic ($t = \infty$) distributions; $y_0 = 1$, $t = 20$, (R, K, H) independent, $N(0.8, 0.5), N(4.5, 0.01), N(0.5, 0.2)$ (truncated) respectively.

We examine the convergence properties of both rectangular FAPS and HOPS in Fig. 9.6. A useful measure of the distance between an approximate and exact cumulative distribution function is the Kolmogorov-Smirnov (K-S) test statistic $\max_{x \in \mathbb{R}} |F_q(x) - F_{\hat{q}}(x)|$). We test each of the methods against Monte-Carlo computations of varying numbers of simulations using this test statistic. To form the approximate cumulative distribution function for the Monte-Carlo approach, we compute the empirical cumulative

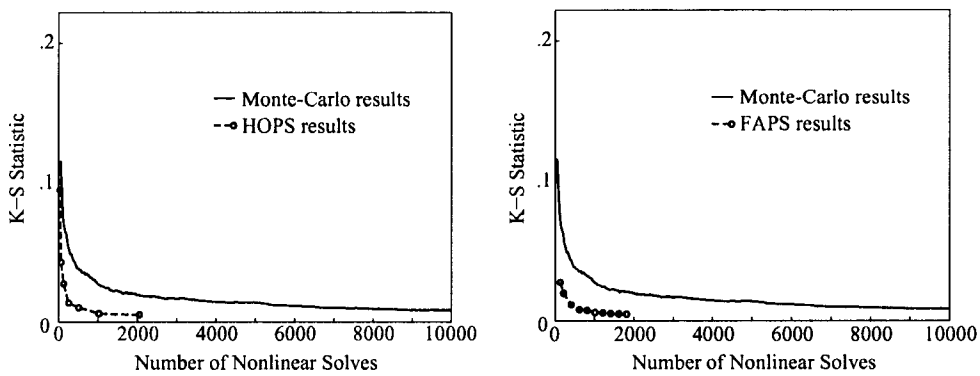

Figure 9.6: Comparison of the K-S statistic for the HOPS, rectangular FAPS, and Monte-Carlo methods applied to (9.2). The error for the Monte-Carlo method is the average over 20 cases. A 70000 point Monte-Carlo simulation is taken as the reference distribution. We use the number of nonlinear solves as a crude measure of computational work.

distribution function

$$F_n^{MC}(x) = \frac{\{\text{number of } i \text{ with } q(\boldsymbol{\lambda}_i) \leq x\}}{n},$$

for sample points $\boldsymbol{\lambda}_i$ and solutions for the quantity of interest $q(\boldsymbol{\lambda}_i)$.

Since F_n^{MC} depends on the random sequence, it is actually a random variable (in fact, the distribution of the (K-S) statistic for a random sample is independent of the distribution of the variable being sampled). Our cumulative distribution is deterministic, however. To be fair in the comparison of the (K-S) statistics, we compare to the mean F_n^{MC} for a number of Monte-Carlo draws.

Since we do not have the exact cumulative distribution function for the quantity of interest in the logistic example, we compare both results to a massive Monte-Carlo simulation of 70000 points. The results in Fig. 9.6 show that both the HOPS and rectangular FAPS algorithms provide far better approximations in the sense of the (K-S) statistic, especially for numbers of simulations in the ranges feasible for the solution of a quantity

of interest from a differential equation. The graph shows that simulations of less that 1000 points (in both cases) give results with the accuracy of a 10000 point Monte-Carlo simulation.

We conclude by presenting some results for the partition of unity FAPS applied to (9.2) with $H = 0$, $x_0 = 1$, and $T = 3$. We first consider sampling according to the density described in Sec. 8.2, which represents the error. In Fig. 9.7, we plot the true error $(q(\boldsymbol{\lambda}) - \tilde{q}(\boldsymbol{\lambda}))f_{\boldsymbol{\lambda}}(\boldsymbol{\lambda})$ and the approximation $\mathcal{E}_{loc}^{pou}(\boldsymbol{\lambda})$ computed using 20 sample points. In Fig. 9.8, we plot a part of

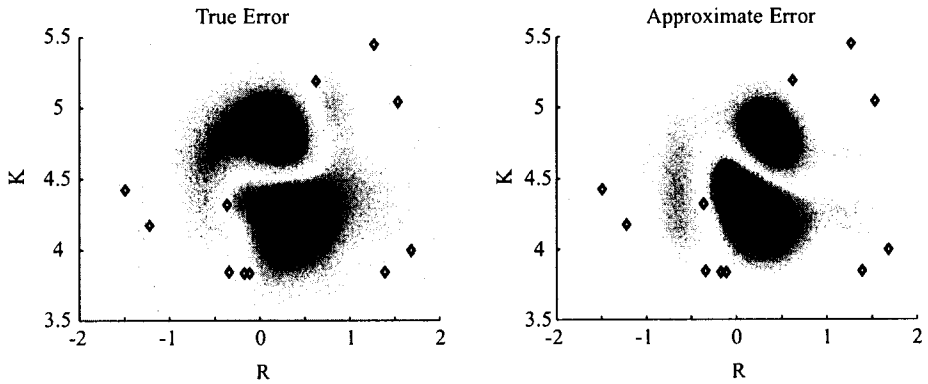

Figure 9.7: On the left, we plot the true error. On the right, we plot the partition of unity FAPS approximation $\mathcal{E}_{loc}^{pou}(\boldsymbol{\lambda})$ computed from 20 sample points using the density described in Sec. 8.2. The sample points are indicated with circles.

the random walk produced by the Markov Chain Monte Carlo that is used to create a set of possible samples and ten new sample points chosen from this set. Finally, in Fig. 9.9, we plot the approximate density created from a FAPS using the partition of unity strategy with 80 points.

As seen in Fig. 9.7, the approximate error function represents the the dominant features of the true error. The new samples (Fig. 9.8) are added largely in the region of the dominant peak. As a result, the error reduced

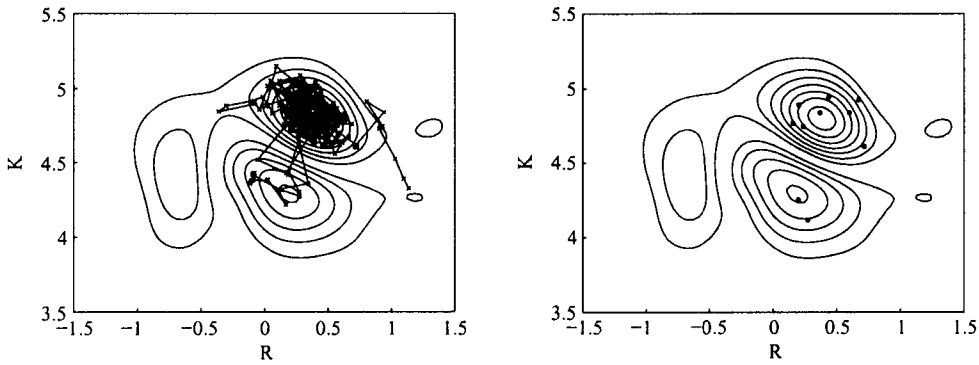

Figure 9.8: On the left, we plot part of the random walk produced by the Markov Chain Monte Carlo that is used to create a set of possible samples from the density described in Sec. 8.2. On the right, we plot ten new sample points chosen from this set.

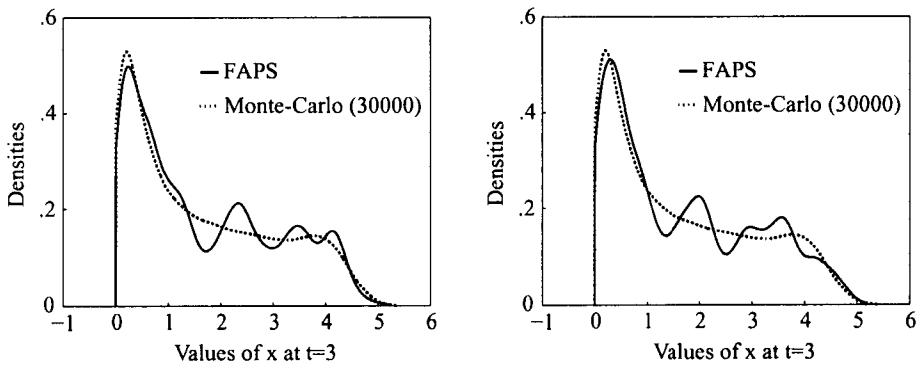

Figure 9.9: On the left, we plot the approximate density computed from a partition of unity FAPS using a sample of 80 points computed using the density described in Sec. 8.2. On the right, we plot the result obtained using the density described in Sec. 8.2.1. We compare to a Monte-Carlo computation with 30000 points.

very quickly, leading to a reasonable density estimate with a small number of evaluations, as shown in Fig. 9.9.

9.3 The Chaotic Lorenz Model

We investigate the Lorenz equations

$$\begin{cases} \dot{x}_1 = \sigma(x_2 - x_1), \\ \dot{x}_2 = r x_1 - x_2 - x_1 x_3, \\ \dot{x}_3 = x_1 x_2 - b x_3, \end{cases} \tag{9.3}$$

where we fix the parameters σ, r, b at standard values believed to yield chaotic behavior as well as the initial values $x_{2,0} = 0, x_{3,0} = 24$, but take the initial value for x_1 uniformly distributed in $[-2, 2]$. We follow the value of the x_1 coordinate of the solution for a sequence of times, so that the functional is $q(\boldsymbol{\lambda}) = q(x_{1,0}) = x_1(t; x_{1,0})$.

All trajectories approach an attractor and the initial distribution is eventually spread onto this attractor, see Fig. 9.10. However, since the

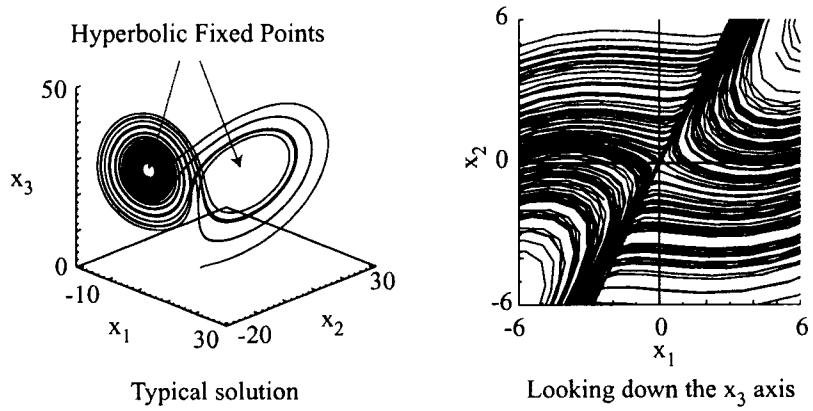

Figure 9.10: On the left, we plot a typical solution of the Lorenz equations. On the right, we plot many solutions from the perspective of looking down the x_3 axis. The separatix manifold lies between the two solutions plotted with thick grey lines. The hyperbolic fixed points lie off the lower left and upper right corners.

problem is chaotic, solutions that start close to each other eventually diverge. There is a separatix manifold (called the razor in [12]) originating out of the x_3-axis that separates trajectories in the sense that close nearby solutions passing by this manifold rapidly move to neighborhoods of different hyperbolic fixed points, see Fig. 9.10. In Fig. 9.11, we see that the initial strip of points is split by this razor into two separate masses. These masses spiral away from the fixed points, and then begin to fold across the razor, creating a series of layers that double in number at each pass. This is evident in the densities, see Fig. 9.12, as we see a sequence of peaks in the distribution.

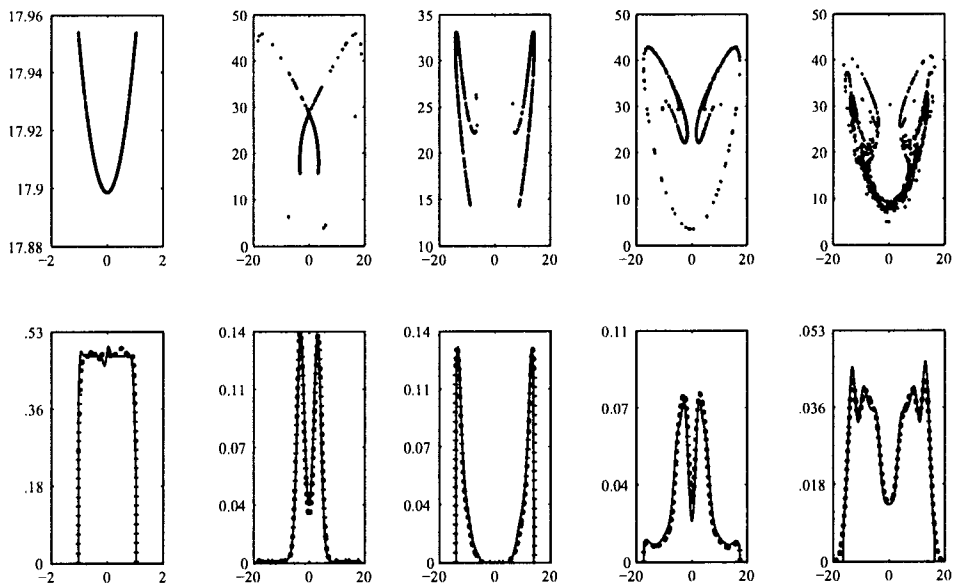

Figure 9.11: Top: The ensemble of points at various times. Plotted are the projections of the solutions onto the $x_1 - x_3$ plane. Bottom: Rectangular FAPS-computed density of $q(x_{1,0}) = x_1(t; x_{1,0})$ compared with a 40000 point Monte Carlo computation. Times are $0.1, 1, 2, 3, 5$ left to right. All Rectangular FAPS simulations use $TOL = 0.01$, with corresponding number of simulations, $64, 94, 134, 240, 336$.

As the solutions pass the razor, the functional $q(x_{1,0})$ develops a steep gradient because solutions with nearby initial conditions go in opposite directions. At each subsequent pass, the gradient becomes nearly vertical at a sequence of $2, 4, 8, \ldots$ points, see Fig. 9.12. The rectangular FAPS algorithm places a large number of samples at each of these gradients. The adjoint computation identifies these doubling regions, and thus identifies regions in the parameter space where solutions are rapidly diverging.

Since the number of vertical regions doubles each time the mass returns to the razor, eventually the algorithm begins placing points almost uniformly throughout the parameter space. The function q becomes so complex that traditional Monte-Carlo methods become an attractive approach.

This doubling is the same phenomenon that makes it impossible to rely on numerical solutions to this system beyond a certain time, since small errors in the integration results in the numerical solution eventually passing on the wrong side of the razor and diverging at an exponential rate, see [15]. The generalized Green's function is, however, a way to determine when such sensitive regions are encountered. It is used in conjunction with numerical adaptivity to extend the ability to compute such solutions accurately by resolving the solution carefully near the razor.

9.4 An SIR Model of Disease with a Number of Parameters

We investigate the behavior of an SIR model of disease in a population that includes birth/death processes and the possibility that the offspring of the resistant class may inherit the resistance. Assuming the birth/death

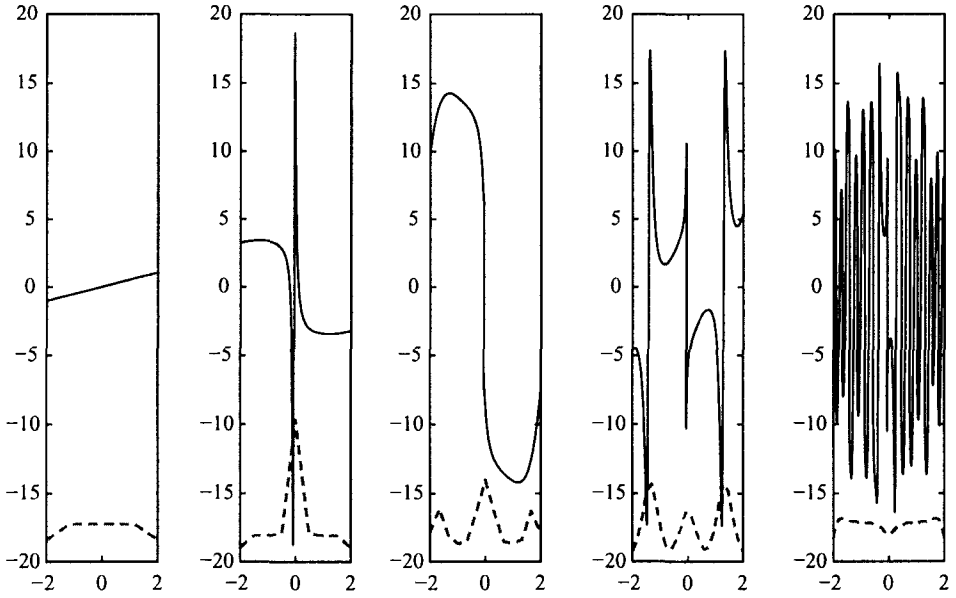

Figure 9.12: Plot of the functional $q(x_{1,0})$, at times $0.1, 1, 2, 3, 5$, against $x_{1,0}$ (solid). The density of mesh points ($\{\bar{\boldsymbol{\mu}}_i\}$) used in the rectangular FAPS is plotted with a dashed line. This indicates refinement where ∇q is large.

processes are described by a logistics model, the model is

$$\begin{cases} \dot{x}_1 = r_n(1 - \frac{N}{k})\,(x_1 + x_2 + (1 - p_R)x_3) - d_n x_1 - r_I x_1 x_2, \\ \dot{x}_2 = r_I x_1 x_2 - (d_n + d_I)x_2 - a_R x_2, \qquad 0 \le t \le T, \\ \dot{x}_3 = p_R r_n(1 - \frac{N}{k})x_3 - d_n x_3 + a_R x_2, \\ x_1(0) = x_{1,0}, x_2(0) = x_{2,0}, x_3(0) = x_{3,0}, \end{cases} \tag{9.4}$$

where x_1 represents the population susceptible to the disease, x_2 the infected population, x_3 the resistant class, and the total population is $N = x_1 + x_2 + x_3$. We consider this problem for the parameter values given in Table 9.1.

For the mean values chosen for the parameters in Table 9.1, the introduction of a small number of infected leads to an *epidemic* in the form of a transient spike of infections. Thereafter, the population tends towards a steady state, see Fig. 9.14. Taking $\boldsymbol{\psi}(s) = (0, 1, 0)^\top$ gives an appropriate linear functional to measure the impact of the infection over an interval

Description	Name	Mean	Perturbation
Recovery rate	a_R	0.2	± 0.1
Natural growth rate	r_n	1	± 0.2
Carrying capacity	k	100	± 5
Probability of inheriting resistance	p_R	0.1	± 0.01
Natural death rate	d_n	0.2	± 0.1
Contraction rate	r_I	0.2	± 0.1
Death rate from disease	r_I	1	± 0.2

Table 9.1: Parameter means and ranges for SIR model. We take the parameters to be uniformly distributed in the ranges given.

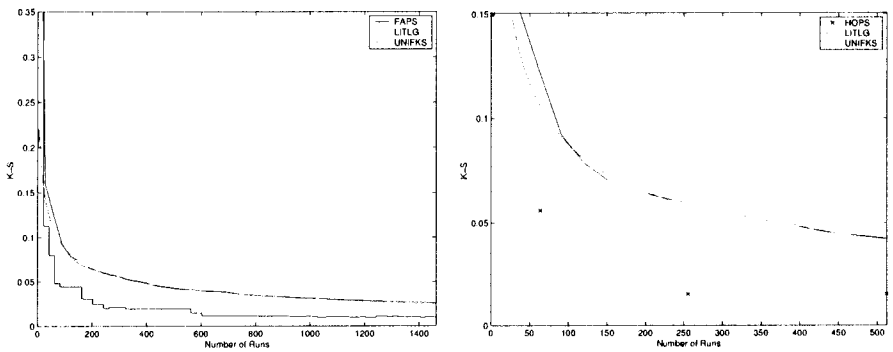

Figure 9.13: Comparison of the K-S statistic for FAPS and HOPS on the SIR model. We compare against the mean Monte-Carlo behavior (MCAVG), and the *law of the iterated logarithm* result (LITLG). Left: *FAPS*, Right: *HOPS*.

$t \in [0, T]$,

$$q(\boldsymbol{\lambda}) = \int_0^T x_2(s; \boldsymbol{\lambda})\, \mathrm{d}s,$$

where $\boldsymbol{\lambda}$ denotes the vector of parameters in Table 9.1. The HOPS procedure produces a reasonable approximation for a very small number of points. FAPS also does a very good job, as shown in Fig. 9.4.

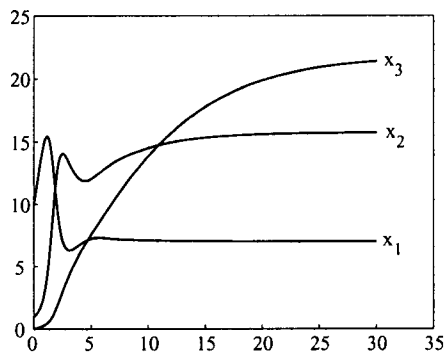

Figure 9.14: Solution of the SIR model for $x_{1,0} = 10$, $x_{2,0} = 1$, $x_{3,0} = 0$ and mean parameters given in Table 9.1.

Chapter 10

APPLICATION TO PARTIAL DIFFERENTIAL EQUATIONS

We apply the previous methods to a specific system of reaction diffusion equations modelling predator-prey interactions, as described in [18].

10.1 Predator prey with Holling II functional response

Description	Name	Mean	Perturbation
Prey saturation	w	1	±50%
Prey growth rate	r	5	±50%
Predator death rate	s	1	±50%
Encounter loss (prey)	p	1	±50%
Encounter gain (pred)	q	1	±50%
Response gain	k	10.1	±50%

Figure 10.1: Parameter means and ranges for the predator/prey model.

We examine the theory above applied to the system

$$\begin{cases} \partial_t v - \delta_1 \nabla^2 v = qvh(k,u) - sv, & \Omega \times (0,T], \\ \partial_t u - \delta_1 \nabla^2 v = ru(1-\frac{u}{w}) - pvh(k,u), & \\ \partial_n v = \partial_n u = 0, & \partial\Omega \times (0,T], \\ v = v_0, u = u_0, & \Omega \times \{0\}, \end{cases} \tag{10.1}$$

where $h(k,u)$ satisfies

i) $h(0) = 0$,

ii) $\lim_{x\to\infty} h(x) = 1$,

Figure 10.2: Left: *Predator* Right: *Prey* In this sequence, we can see two wavefronts of the predator population chasing the prey. Eventually the predator eats itself "out of house and home", and its population decays. The prey then replenish, and the cycle continues. Warmer colors indicate greater numbers.

iii) h strictly increasing.

The population v models a predator that is sustained solely by they prey u, which grows according to a logistic model. Both species diffuse spatially, typically with $\delta_1 < \delta_2$.

The Neumann boundary conditions model an enclosed region where neither prey nor predator can escape the region.

We use truncated normal distributions with support on $[p_i - 0.5p_i, p_i + 0.5p_i], i = 1, \ldots, 6$. For the quantity of interest, we set $\psi(t) = \delta(t - 10) \cdot (0, 1)^\top$, which yields the L_1 norm of the prey population at $t = 10$ (since the number of prey is always positive).

Using a Monte-Carlo of 12400 points as a reference, in Fig. 10.3 the FAPS approach is shown to work very well. The K-S measure shows that the FAPS approximates the reference distribution better than the average Monte-Carlo ensemble. Since FAPS is deterministic, we consider this to be very good performance, especially since a given realization of the Monte-Carlo is (on average) as often above the mean (and so much worse than

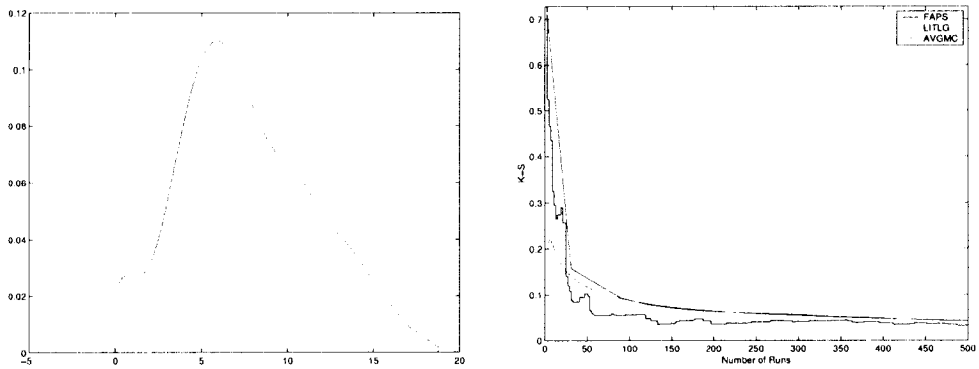

Figure 10.3: Left: *The reference density using a* 12400 *point Monte-Carlo* Right: *The K-S statistic for FAPS, compared with the average behavior for a Monte-Carlo (AVGMC), and the Law of the Iterated Logarithm asymptotic behavior for the Monte-Carlo (LITLG). The Monte-Carlo on the left is used as the reference distribution.*

FAPS) as below. In this sense, FAPS is a very reliable way to approach recovering the distribution in question.

A plot of the coordinates of each sample, Fig. 10.4, show that large contributions to the uncertainty are present for w, s and q.

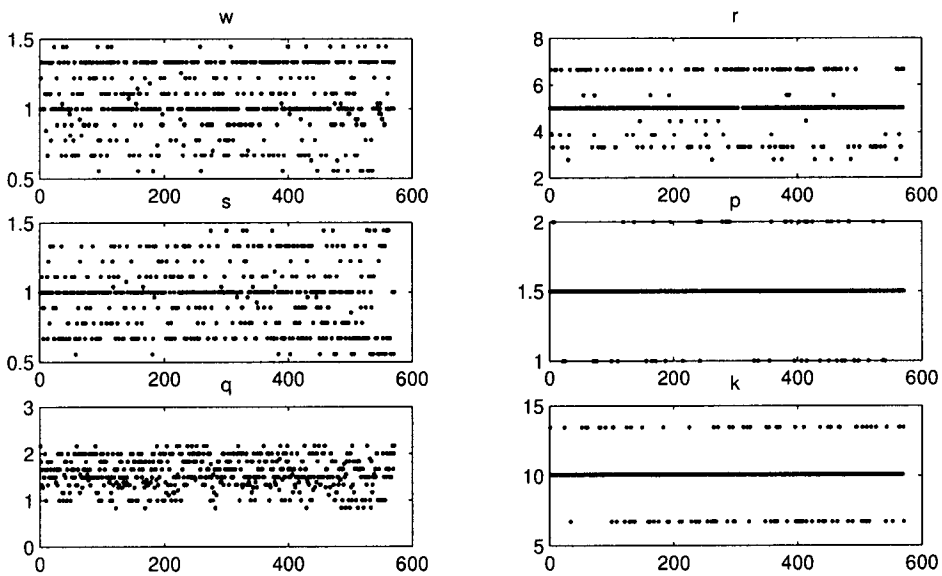

Figure 10.4: Centers of each sample, plotted against the sequence of samples. Parameters with large stratification indicate greater contribution to the overall uncertainty.

Chapter 11

ANALYSIS OF THE SENSITIVITY PROPERTIES OF A VECTOR-BORNE MODEL OF THE PLAGUE

In this chapter, we study the parameter sensitivity of a vector-borne disease model of the bubonic plague in a rodent population proposed by Keeling and Gilligan [23]. To carry out the analysis, we use the techniques developed above. This approach both provides a significantly more efficient means to explore the sensitivity properties of a model and provides additional information regarding sensitivity that is not easily obtained using Monte-Carlo methods. In particular, we use the new approach to determine the sensitivity of a quantity of interest to various parameters, determine a region over which linearization at given parameter value is valid, develop a global picture of the output surface, and search for extremal values in a given region in parameter space. This chapter is a joint work with Megan Buzby [3].

11.1 Introduction

In [23], Keeling and Gilligan develop and analyze a deterministic vector-borne model of bubonic plague enroute to developing a stochastic metapopulation model. The ultimate goal is to use the disease and population dynamics of rats and fleas to explain how an unexpected epidemic can occur in

human populations. The behavior of the deterministic model with respect to the eleven parameters that are used in the model comprise an important part of Keeling and Gilligan's considerations. In turn, their conclusions about the model behavior depends critically on an analysis of the sensitivity of the model with respect to the parameters. This sensitivity analysis is essentially based on the properties of one-dimensional slices, i.e., varying one parameter at a time, of the linearization of the model at a single reference value for the parameters. The behavior of the linearization at this single reference value is claimed to represent the behavior of the model over an entire large (generalized) rectangle in parameter space.

In hind sight, there is little mathematical basis to believe that the behavior of the linearization of a nonlinear operator at a single point can suffice to accurately describe the behavior of the operator over a large region in eleven dimensions. Indeed, we find that it does not in this particular case.

11.1.1 A Model of Plague

We consider the SIRNF model developed by Keeling and Gilligan, [23], that describes the dynamics of the bubonic plague. We follow their description rather closely. They begin with an SIR-type model for the rat population that gives the changes in the number of susceptible (S_R), infectious (I_R), and resistant (R_R) rats,

$$\begin{cases} \dot{S}_R = r_R S_R(1 - \frac{T_R}{K_R}) + r_R R_R(1-p) - d_R S_R - \beta_R \frac{S_R}{T_R} F(1 - \exp(-aT_R)), \\ \dot{I}_R = \beta_R \frac{S_R}{T_R} F(1 - \exp(-aT_R)) - (d_R + m_R) I_R, \\ \dot{R}_R = r_R R_R(p - \frac{T_R}{K_R}) + m_R g_R I_R - d_R R_R, \end{cases} \tag{11.1}$$

where $\cdot$ denotes the time derivative and $T_R = S_R + I_R + R_R$ is the total size of the rat population. The net reproduction rate of susceptible and infected

rats is given by r_R with a carrying capacity of K_R. The proportion of offspring that inherit resistance to the disease from their parents is given by the parameter p. The natural death rate of all rats is d_R. The number of free infected fleas looking for a host is F. The infection term, $(1-\exp(-aT_R))$, corresponds to infected fleas randomly searching for a new rat host for some given time period, [25]. If they find a host and it is susceptible, then the rat becomes infected with a given probability. Thus, β_R is the transmission rate to rats and a is a measure of the searching efficiency of the fleas. Rats leave the infected class at a rate of m_R and a fraction g_R of these survive to become resistant; the remainder die and release their infected fleas back into the environment.

The flea population dynamics are modeled by the flea index N, which is the average number of fleas living on a rat (each rat is assumed to have the same number of fleas) and the number of free infectious fleas F that are searching for a new host,

$$\begin{cases} \dot{N} = r_F N(1-\frac{N}{K_F}) + \frac{d_F}{T_R} F(1-\exp(-aT_R)), \\ \dot{F} = (d_R + m_R(1-g_R))I_R N - d_F F. \end{cases} \tag{11.2}$$

The net reproduction rate of the fleas is given by r_F with a carrying capacity of K_F, while the death rate is d_F.

In addition, Keeling and Gilligan describe the human aspect of the model. However, human population dynamics do not affect the number of new human cases since only pneumonic symptoms from the plague can be passed between humans. The potential for human cases is directly related to the number of infected fleas that fail to find a rat host and so choose a human as their next best option,

$$\mathcal{F}_I = F\exp(-aT_R). \tag{11.3}$$

The ***quantity of interest*** $\mathcal{F}_I$ (labeled λ_H in [23]) represents the upper bound or the *potential* force of infection to humans.

Keeling and Gilligan use parameter values derived from experiments or field observations when possible and the remaining parameters are set within biologically realistic bounds. We list the reference values in Table 11.1. Keeling and Gilligan make assertions about the behavior of the

Parameter	Value	Interpretation
$\mu_1 = r_R{}^*$	5/yr	rat reproductive rate
$\mu_2 = p^*$	0.975	probability of rat resistance
$\mu_3 = K_R$	2500/km^2	rat carrying capacity
$\mu_4 = d_R{}^*$	0.2/yr	rat death rate
$\mu_5 = \beta_R{}^*$	4.7/yr	transmission rate
$\mu_6 = m_R$	20yr	(infectious period)$^{-1}$
$\mu_7 = g_R{}^*$	0.02	probability of recovery
$\mu_8 = a$	4x10^{-3}	flea searching efficiency
$\mu_9 = r_F$	20/yr	flea reproductive rate
$\mu_{10} = d_F{}^*$	10/yr	flea death rate
$\mu_{11} = K_F{}^*$	6.57	flea carrying capacity

Table 11.1: Reference values for the parameters. Values marked with an asterisk have been estimated from either laboratory experiments or field observations, see references in [23].

model (11.1)-(11.2) over a large generalized rectangle in the parameter space centered at the reference values.

Keeling and Gilligan use a stochastic model derived from (11.1)-(11.2) to describe a mechanism by which the plague can lay dormant for many years but then break out in an epidemic in the human population. A key part of their explanation depends on consideration of plots of $\mathcal{F}_I$ versus F and T_R as well as plots of $\mathcal{F}_I$ and I_R/m_R versus time. These show that their model predicts that the highest potential for a human plague epidemic occurs after a major epizootic in the rat population, which leaves a large

number of infected fleas looking for a host. This is discussed in terms of the reproductive ratio of the disease, $R_0 = \frac{\beta_R K_F}{d_F}(1 - \exp(-aK_R))$.

11.1.2 Four Assertions

The analysis below addresses four assertions that Keeling and Gilligan make about the deterministic model (11.1)-(11.2). These assertions depend critically on an underlying claim about the sensitivity of the model with respect to the parameters. As a measure of sensitivity of an output variable V at a fixed time T with respect to a parameter λ, Keeling and Gilligan use

$$S = \lim_{\lambda \to \mu} \frac{\log(\frac{V(\lambda)}{V(\mu)})}{\log(\frac{\lambda}{\mu})} = \frac{\mu}{V(\mu)} \frac{\partial V}{\partial \lambda}, \tag{11.4}$$

where μ is the reference value. This implies that

$$V(\lambda) \approx V(\mu) \left(\frac{\lambda}{\mu}\right)^S, \tag{11.5}$$

so, for example, when V is proportional to λ, the sensitivity is equal to 1.

The four assertions are

1. "From multiple simulations, we note that even when parameters are changed by a factor of two, the essential pattern of sensitivity ... remains, showing that $S = \frac{\mu}{V(\mu)} \frac{\partial V}{\partial \lambda}$ is a robust measure of the effects of parameter change."

2. "Only K_F, the carrying capacity of fleas per rat, has any effect (on $\mathcal{F}_I$ and I_R/m_R) that is much stronger than linear, although, $r_R, 1-p, K_R$, and a all have effects on the number of rat or human cases that are close to linear."

3. "... it is sufficient to consider whether results are robust to changes in the parameter a."

4. "The basic reproductive ratio of the disease, R_0, is related to the parameters via aK_R in the following way: $R_0 = \frac{\beta_R K_F}{d_F}(1 - \exp(-aK_R))$. From standard theory, rat epizootics can occur whenever $R_0 > 1$, which corresponds to $aK_R > 0.39$. It can also be seen that a large human outbreak is possible whenever $0.5 < aK_R < 20$."

A critical part of Keeling and Gilligan's analysis of (11.1)-(11.2) leading to Assertions 1-4 depends on the underlying mathematical claim that the linearization of the model with respect to the parameters at the reference values describes the behavior of the model over a large region in parameter space. However, there is no *a priori* reason to believe that Taylor's theorem can be used in this way. In this paper, we present some new analysis that shows that in fact this claim does not hold for (11.1)-(11.2).

11.1.3 The Quantity of interest

Since the quantity of interest $\mathcal{F}_I$ is a nonlinear functional, we cannot use the representation formula directly. We note that

$$\partial_{\lambda_i}\mathcal{F}_I = \partial_{\lambda_i}F\exp(-aT) - aF\exp(-aT)\partial_{\lambda_i}T,$$

for $i \neq 8$, and so it suffices to find $\partial_{\lambda_i}F$ and $\partial_{\lambda_i}T$, which are linear functionals of the solution corresponding to data $\psi_1 = \delta(t-T)(0,0,0,0,1)^\top$ and $\psi_2 = \delta(t-T)(1,1,1,0,0)^\top$. Since $\lambda_8 = a$, a slight modification above gives $\partial_{\lambda_8}\mathcal{F}_I$.

To calculate these derivatives, we simultaneously integrate two adjoint solutions corresponding to $\psi_{1,2}$, linearizing around the same forward solution.

11.2 Analysis of the Model

Theorem 5.1.1 shows that the adjoint problem provides a relatively cheap way to compute the gradient of the quantity of interest with respect to parameter at each sample point. We use this gradient information in several ways.

- We find a relatively small rectangle in parameter space over which the linearization at the reference value provides a reasonably accurate description of the behavior of the model.
- We construct a piecewise constant approximation of the quantity of interest over a large region in parameter space that is close in size to the region over which Keeling and Gilligan want to draw conclusions about the model behavior.
- We implement a method of steepest ascent and descent to find local extrema in a large region in parameter space.

From these computations, we can draw several conclusions about the model sensitivity and re-examine the validity of the Assertions 1-4.

We make some initial observations. First, Keeling and Gilligan state that their conclusions about the model behavior hold for a region over which parameter values change by a factor of two. This is somewhat imprecise, especially given that several parameters should remain strictly positive. Instead, we consider a large rectangle $\mathcal{R}_L$ with sides $[\mu_i/2, 3\mu_i/2]$ for each i (if either end point goes past a physical bound, we use the physical bound instead). Even a region of that size allows a wide range of behaviors, see Fig. 11.1. Second, $\mathcal{F}_I$ depends on the final time T for which the model is

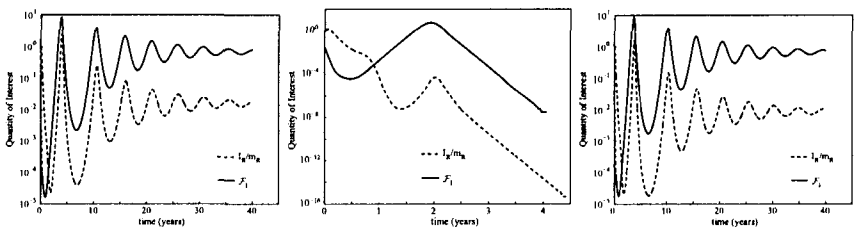

Figure 11.1: Plots of the time behavior of the quantities of interest $\mathcal{F}_I$ and I_R/m_R. The results on the left are for the reference parameter values, with $a = .004$. The results in the middle use parameter values from an "upper" boundary of the larger rectangle $\mathcal{R}_L$ with sides $[\mu_i/2, 3\mu_i/2]$, with $a = .006$ and $p = 1.0$. This demonstrates the wide range of behavior found in a region that size. The plot on the right are for parameter values in one corner of a smaller rectangle $\mathcal{R}_S$ centered at the reference value with sides $[31\mu_i/32, 33\mu_i/32]$, with $a = .004125$.

solved. Keeling and Gilligan use $T = 100$. However, the deterministic model is actually used to construct a stochastic model that, as stated explicitly by Keeling and Gilligan, does not have the same long time behavior as the deterministic model. Indeed, it is the transient behavior of the deterministic model that is relevant to the stochastic model derived by the standard process. Therefore, we use a smaller time $T = 10$ for the analysis of model sensitivity. This affects the conclusions quantitatively but not qualitatively.

11.2.1 Linearization at the Reference Value

Sufficient sampling of $\mathcal{F}_I$ shows that the linearization at the reference values μ does not describe the behavior of the model over the large rectangle $\mathcal{R}_L$. For example, if we use a uniform distribution on $\mathcal{R}_L$, the partial derivatives $\partial\mathcal{F}_I/\partial\lambda_i$ are centered at 0 for $i = 4, 5, 6, 7$ but have variances $210, 3300, .19, 49000$ respectively, see Fig. 11.2. This has several implications. For example, varying more than one parameter simultaneously results in complicated behavior and $\mathcal{F}_I$ depends nonlinearly on some parameters, see Fig. 11.3.

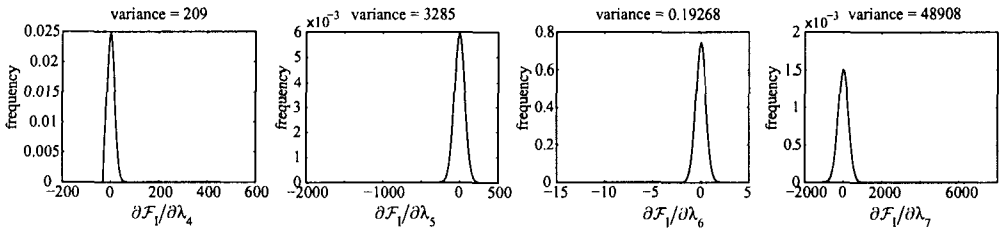

Figure 11.2: Output distributions of $\partial\mathcal{F}_I/\partial\lambda_i$, $i = 4, 5, 6, 7$, for a uniform input distribution on $\mathcal{R}_L$. Note the wide range of variance values, indicated above each plot

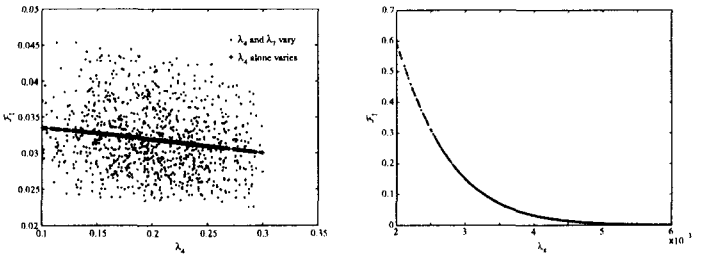

Figure 11.3: On the left, we plot $\mathcal{F}_I$ versus λ_4 when λ_4 alone is varied and when both λ_4 and λ_7 are varied simultaneously. On the right, we plot $\mathcal{F}_I$ versus λ_8 holding the other parameters fixed. Parameter values are taken from $\mathcal{R}_L$.

To find a region over which the linearization at the reference value provides a reasonable approximation, we start with the large rectangle $\mathcal{R}_L$ and then successively refine it to obtain a sequence of smaller rectangles centered at the reference value. The criteria for the search is based on the observation that a linear transformation maps a normal input distribution to another normal distribution. On a given rectangle, we choose a normal distribution truncated to the rectangle (see [3] for details on the truncation) and then compute the corresponding output distribution of $\mathcal{F}_I$ using a kernel density estimator based on a piecewise constant approximation computed using the new approach. If the output distribution is not approximately normal with

a reasonable variance, we refine the rectangle by halving the dimensions, and try again. Using a goal of achieving a variance with relative size 1, we stop the process after 16 iterations to obtain a smaller rectangle $\mathcal{R}_S$ with side lengths $[31\mu_i/32, 33\mu_i/32]$. We did not refine further because the variances for λ_2 and λ_8 remain significantly large despite further reductions. Sampling on $\mathcal{R}_S$ demonstrates that the time behavior of the quantities of interest $\mathcal{F}_I$ and I_R/m_R is the same over $\mathcal{R}_S$, see Fig. 11.1.

11.2.2 Adaptive Sampling

The computation of the FAPS adaptive piecewise constant approximation itself is quite revealing in terms of the model sensitivity. For example, recall that at each step we refine by adding two new sample points in the parameter direction in which the partial derivative of $\mathcal{F}_I$ is largest. The number of additional sample points added in each parameter is an indication of the sensitivity of $\mathcal{F}_I$ with respect to that parameter. In Fig. 11.4, we plot the value of each parameter at the center of each division in FAPS against the number of samples used to create the piecewise constant approximation at each iteration. This shows that $\mathcal{F}_I$ is highly sensitive to parameters $\lambda_1, \lambda_3, \lambda_5, \lambda_{10}$, and λ_{11} and not very sensitive to $\lambda_2, \lambda_4, \lambda_6, \lambda_7, \lambda_8$, and λ_9 on the larger rectangle $\mathcal{R}_L$. We emphasize that the same conclusion can be made from a FAPS computation with just 50 sample points.

To compare the convergence rate of the FAPS algorithm to the properties of Monte-Carlo simulations, we use the Kolmogorov-Smirnov (K-S) statistic, $D_n(\omega) = \|\hat{F}_n(x,\omega) - F(x)\|_\infty$, which measures the supremum of the error of an approximate distribution function $\hat{F}_n(x) \approx F(x)$. In this case, we approximate the errors by using a very accurate approximation to

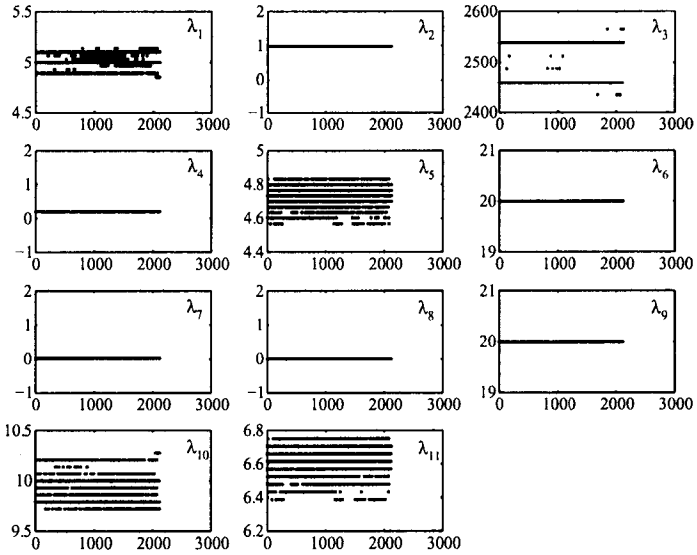

Figure 11.4: Plots of the FAPS sampling. The vertical axes are the parameter values and the horizontal axes are the number of points used in FAPS.

the unknown $F(x)$ computing using an extremely large Monte Carlo simulation as a "reference" value. In Fig. 11.5, we plot the K-S statistic for FAPS versus the number of sample points along with the error of the mean of several Monte Carlo samples using uniform random sampling along with the asymptotic upper bound on the error given by the *Law of the Iterated Logarithm*, [14]. We also plot the individual errors of the Monte-Carlo simulations to show the effects of the possible variation in that approach.

We are interested in the convergence rate of FAPS mainly for a small number of sample points and, in this regime, we see that FAPS converges at the same rate as the average Monte-Carlo computations and the theoretical bound. Restricting the addition of sample points to the directions with the largest derivatives leads to the staircase appearance of the plots and the leveling out of the statistic for moderate sample sizes. In [13], we combine the derivative information in Theorem 5.1.1 with the standard techniques

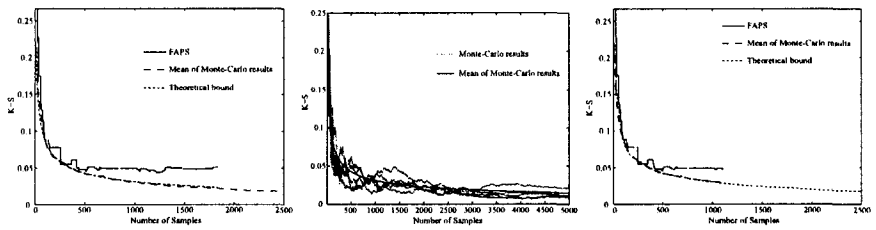

Figure 11.5: Plots of the K-S statistic. On the left, we plot the results for FAPS, along with the upper bound given by the Law of the Iterated Logarithm and the mean of several Monte Carlo samples. In the middle, we plot the individual K-S statistic for the Monte-Carlo computations. On the right, we plot the K-S statistic for FAPS where $\lambda_2, \lambda_4, \lambda_6, \lambda_7, \lambda_8$, and λ_9 are held fixed at the reference value.

for increasing the efficiency of the Monte-Carlo method called stratified sampling and control variates and obtain significant improvements that also avoid the inherent biases of a deterministic sampling scheme.

In Fig. 11.5, we also plot the K-S statistic for FAPS when $\lambda_2, \lambda_4, \lambda_6, \lambda_7, \lambda_8$, and λ_9 are held fixed at the reference value. This confirms the observation from Fig. 11.4 that the $\mathcal{F}_I$ is sensitive to the parameters $\lambda_1, \lambda_3, \lambda_5, \lambda_{10}$, and λ_{11}, and we can safely reduce the dimension of the model by considering variations in these parameters alone.

11.2.3 Finding Extrema in Parameter Space

As part of predicting human epidemics, Keeling and Gilligan examine the quantity of interest R_0 at extreme values of a and K_R while keeping the other parameters fixed at the reference value. There is only a very narrow window, $.39 < aK_R < .5$, for which the plague can affect the rat population significantly without causing a human epidemic. Their approach is reasonable if the model is insensitive to parameter changes, as claimed by Keeling and Gilligan, given that the partial derivative of the quantity of interest is very large with respect to $\lambda_8 = a$ at the reference value. However,

we have seen that the model does depend on five parameters and moreover is sensitive to simultaneous parameter changes.

A natural extension of Keeling and Gilligan's idea is to implement a method of steepest ascent and descent to search for local extrema in parameter space. Recall that the gradient at a point gives the direction of the steepest increase and decrease for the function. Starting at a point, we can take a small step in the direction of the gradient (or minus the gradient) and recompute new values before stepping again. In this way, we create a path of linked sample points along which we search for an extremal value (where the gradient is zero) or for a point where the path leaves the rectangle. See [3] for details on implementation.

There are two ways to use this approach. First, we can take the reference value as the starting point and following the corresponding trajectory on both the smaller rectangle $\mathcal{R}_S$ and on a larger rectangle $\tilde{\mathcal{R}}_L$ with side lengths $[\mu_i/2, 2\mu_i]$. Keeling and Gilligan's claim amounts to stating that the trajectory should leave the larger rectangle along the $\lambda_8 = a$ axis. Second, we can choose some points near to the reference value and follow those trajectories to see if they stay close to the reference trajectory for all sample points. We choose five additional points.

On the smaller rectangle $\mathcal{R}_S$, the trajectories corresponding to all six starting points and following the gradient (the direction of steepest ascent) exit the rectangle on the λ_8 face. In that sense, Keeling and Gilligan's conclusion is correct. However, the exit points vary greatly over this face as the trajectories do not remain close and variations in the other parameters do have an important effect. On the larger rectangle $\tilde{\mathcal{R}}_L$, the trajectory corresponding to the reference value and following the gradient leaves the

larger rectangle $\tilde{\mathcal{R}}_L$ on the lower face for λ_7, $\lambda_7 = .1$, and the upper face for λ_2, $\lambda_2 = 1$, though at a wide range of points. Interestingly, the trajectories corresponding to the five additional points all reach local minima inside the rectangle, again at different points. On the other hand, all of the trajectories corresponding to following the gradient leave on the λ_2 face of the larger rectangle, though at six different points.

11.3 Conclusions

We begin by addressing Keeling and Gilligan's Assertions 1-4.

1. Regarding Assertion 1, the linearization at the reference values simply does not accurately represent the behavior of the nonlinear model over the large rectangle $\mathcal{R}_L$. In particular, there is enormous variance in the sensitivity on $\mathcal{R}_L$, simultaneously varying multiple parameters on $\mathcal{R}_L$ leads to unpredictable results, and there are some definite nonlinear dependencies. The linearization at the reference value does accurately represent the behavior of the model over a much smaller rectangle $\mathcal{R}_S$.

2. Regarding Assertion 2, we find that the quantity of interest $\mathcal{F}_I$ is most sensitive to variations in parameters $\lambda_1 = r_R$, $\lambda_3 = K_R$, $\lambda_5 = \beta_R$, $\lambda_{10} = d_F$, and $\lambda_{11} = K_F$ in contrast to the findings of Keeling and Gilligan, which predict the most sensitive dependence on $\lambda_1 = r_R$, $\lambda_2 = p$, $\lambda_3 = K_R$, $\lambda_8 = a$, and $\lambda_{11} = K_F$.

3. Regarding Assertion 3, we cannot faithfully predict the behavior of the model by fixing all the parameters except for a at the reference values and considering the effect of varying a on even the smallest

rectangle $\mathcal{R}_S$. Moreover, there is no apparent correlation between the direction of the gradient at the reference point and nearby points and the location of extremal values along paths given by the method of steepest ascent and descent.

4. Regarding Assertion 4, when predicting the behavior of the model for values of aK_R, we have to consider the range for the other parameter values.

We want to emphasize that these observations do not necessarily contradict the scientific conclusions of Keeling and Gilligan in [23] because the main results are based on the properties of a stochastic model derived from (11.1)-(11.2). The stochastic and deterministic models certainly have different behavior, and, in fact, the connection between their behaviors is not clear at all. On the other hand, it is also clear that the dependence of nonlinear models like (11.1)-(11.2), even when composed of well understood mechanisms, can exhibit complicated nonlinear sensitivity to parameters that are not predictable from linearization at a single point in parameter space. Keeling and Gilligan's assertions about the model are reasonable on a relatively small region in parameter space surrounding the reference value. The scientific question is whether or not this region is sufficiently large, i.e., that the behavior of the model with respect to parameters is sufficiently robust, to have practical meaning.

When analyzing the sensitivity of a model with respect to parameters, the ability to compute gradient information at sample points is invaluable. We have demonstrated that the new approach based on the adjoint problem, variational analysis, and the generalized Green's function encapsulated

by Theorem 5.1.1 provides a very efficient way to compute gradient information. In turn, this provides several powerful tools for investigating model sensitivity.

Chapter 12

APPLICATION TO TIME DEPENDENT RANDOMNESS

We explore the use of the methods outlined above to the case of a differential equation involving a stochastic process as a coefficient. We begin with some background material on stochastic processes and their spectral expansion, the K-L expansion.

12.1 Stochastic processes

The problems described previously were concerned with some quantity involving a d dimensional random vector. However, many problems with uncertainty involve infinite dimensional random vectors, which are called stochastic processes or fields. In particular, consider the differential equation

$$\begin{cases} \dot{\boldsymbol{x}}(t) = f(\boldsymbol{x}(t); \zeta(t)), & 0 \leq t \leq T \\ \boldsymbol{x}(0) = \boldsymbol{x}_0, \end{cases}$$

where the coefficient $\zeta(t), t \in [0, T]$ is random at each instance in time. Such a coefficient is called a stochastic process, and is an infinite dimensional analogue to a random vector.

In its most general form, a stochastic process $[X_t, t \in T]$ is a collection of random variables indexed by (any) set T. Such processes are generally

characterized by their *finite dimensional distributions* (*fidis* for short), which are the measures formed by

$$\mu_{t_1 \ldots t_k}(A) = P(X_{t_1}, \ldots, X_{t_k} \in A),$$

for Borel sets A in $\mathbb{R}^k$.

Under certain consistency conditions Kolmogorov's extistence theorem guarantees there is a process with the given *fidis*. The case $T = [0, \infty)$ is used to talk about processes which are random in time, but in Kolmogorov's theorem the space T can just as well be a spatial field, such as some subset of $\mathbb{R}_n$, in which case we refer to the stochastic process as a *random field.*

12.2 Karhunen-Loeve Expansion

If a stochastic process, $\zeta(t) = \zeta(t, \omega)$, has measurable paths, finite second moments ($E[\zeta(t)^2] < \infty$), and is *mean square continuous*, i.e., $E[|\zeta(t) - \zeta(t_0)|^2] \to 0$ as $t \to t_0$, then its covariance function $B(t_1, t_2) = E[\zeta(t_1)\zeta(t_2)]$ is a continuous, symmetric, non-negative definite kernel in the square $[0, T] \times [0, T]$. The integral operator

$$\mathcal{T} f(t) = \int_{[0,T]^2} B(t, s) f(s)\, \mathrm{d}s$$

is thus compact and self adjoint, admitting a sequence of eigenvalues λ_n which decay to zero and a corresponding set of orthonormal eigenfunctions, $\phi_n(t)$. By Mercer's theorm, $B(t, s) = \sum_{i=1}^{\infty} \lambda_i \phi_i(t) \phi_i(s)$.

Setting $\xi_n(\omega) = \int_0^T \zeta(s,\omega)\phi_n(s)\,\mathrm{d}s$, and using Fubini's theorem, we have

$$E[\xi_n\xi_m] = E\left[\int_0^T \int_0^T \zeta(t)\zeta(s)\phi_n(t)\phi_m(s)\,\mathrm{d}t\,\mathrm{d}s\right]$$
$$= \int_0^T \int_0^T B(t,s)\phi_n(t)\phi_m(s)\,\mathrm{d}t\,\mathrm{d}s = \int_0^T \phi_m(s)\mathcal{T}\phi_n(s)\,\mathrm{d}s$$
$$= \lambda_n \int_0^T \phi_m(s)\phi_n(s)\,\mathrm{d}s = \lambda_n\delta_{nm},$$

i.e., ξ_n are orthonormal random variables with variance λ_n.

Further,

$$E[\zeta(t)\xi_n] = \int_0^T B(t,s)\phi_n(s)\,\mathrm{d}s = \lambda_n\phi_n(t),$$

and so

$$E\left[\left|\zeta(t) - \sum_{i=1}^{n}\xi_i\phi_i(t)\right|^2\right] =$$
$$B(t,t) - 2\sum_{i=1}^{n}\phi_i(t)E[\zeta(t)\xi_i] + \sum_{i=1}^{n}\lambda_n\phi_i(t)^2$$
$$= B(t,t) - \sum_{i=1}^{n}\lambda_n\phi_i(t)^2 \to 0,$$

as $n \to \infty$. In fact, the convergence is uniform in t [19].

Thus the stochastic process $\zeta(t,\omega)$ admits a "spectral" representation, the Karhunen-Loeve (K-L) expansion,

$$\zeta(t,\omega) = \sum_{i=1}^{\infty}\xi_i(\omega)\phi_i(t).$$

Therefore, a natural finite dimensional approximation to the process is

$$\bar{\zeta}(t,\omega) = \sum_{i=1}^{p}\xi_i(\omega)\phi_i(t).$$

The new approximate problem is then

$$\begin{cases} \dot{\boldsymbol{x}}(t) = f(\boldsymbol{x}(t); \bar{\zeta}(t)) = f(\boldsymbol{x}(t), t; \boldsymbol{\xi}), & 0 \leq t \leq T \\ \boldsymbol{x}(0) = \boldsymbol{x}_0, \end{cases}$$

where $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_p)$ is a p-dimensional random vector representing the coefficients in the K-L expansion. The problem is no longer autonomous, since the $\phi_i(t)$ are part of the f, but by adding a single dimension to the vector field, we can make the system autonomous, and the approximate problem then is of the form we have discussed previously.

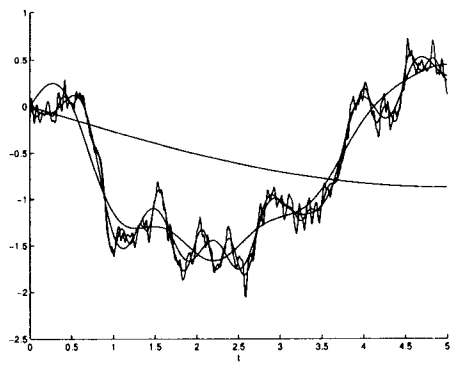

Figure 12.1: *Approximation of Brownian Motion by a truncated K-L expansion, for* $n = 2, 10, 20, 50,$ *and* 250 *terms.*

12.3 Example problem

We investigate the problem

$$\begin{cases} \dot{x}(t) = (\lambda_0 + \sigma B(t, \omega))\, x(t), & 0 \leq t \leq T \\ x(0) = x_0, \end{cases} \tag{12.1}$$

where the process $B(t, \omega)$ is Brownian Motion. This is a process that starts at 0, and is characterized by independent, normally distributed increments, with expected value 0, and variance equal to the length of the increment. Without loss of generality it is possible to assume that almost every path

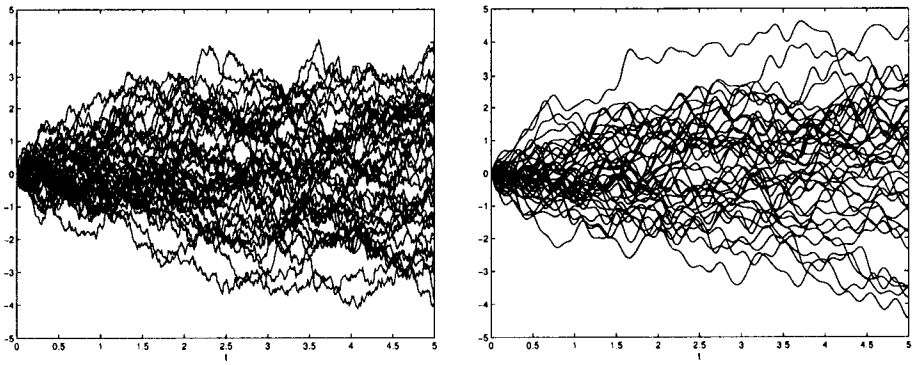

Figure 12.2: Left: 40 *sample paths of Brownian Motion.* Right: 40 *samples paths of the truncated K-L expansion for* $n = 50$.

of Brownian Motion is continuous, and so the system (12.1) has classical (continously differentiable) solutions.

The correlation function for Brownian Motion is $B(t,s) = \min(t,s)$, and so the eigenfunctions in the K-L expansion satisfy

$$\mathcal{T}\phi_n(t) = \int_0^T \min(t,s)\phi_n(s)\,\mathrm{d}s = \int_0^t s\phi_n(s)\,\mathrm{d}s + \int_t^T t\phi_n(s)\,\mathrm{d}s = \lambda_n\phi_n(t).$$

$$\mathcal{T}\phi_n(t) = \int_0^T \min(t,s)\phi_n(s)\,\mathrm{d}s = \int_0^t s\phi_n(s)\,\mathrm{d}s + \int_t^T t\phi_n(s)\,\mathrm{d}s = \lambda_n\phi_n(t)^{\text{ove,}}$$

we have

$$\int_t^T \phi_n(s)\,\mathrm{d}s = \lambda_n\phi_n'(t),$$

so that $\phi_n'(T) = 0$. Differentiating again, we find $\lambda_n\phi_n''(t) = -\phi_n(t)$. This equation along with the boundary conditions has (normalized) solutions

$$\phi_n(t) = \tfrac{\sqrt{2}}{\sqrt{T}}\sin\left((n+\tfrac{1}{2})\frac{\pi}{T}t\right), \qquad \lambda_n = \frac{T^2}{(n+\frac{1}{2})^2\pi^2}, \qquad n = 0,1,\ldots$$

Thus Brownian Motion has the expansion

$$B(t,\omega) = \tfrac{\sqrt{2}}{\sqrt{T}}\sum_{n=0}^{\infty}\xi_n(\omega)\sin\left((n+\tfrac{1}{2})\frac{\pi}{T}t\right),$$

where $\xi_n(t)$ are independent Gaussian random variables with zero mean and variance λ_n (this follows from the integral representation of ξ_n). Truncating

this series to p terms, and letting $x_2 = t$, the approximate problem for (12.1) becomes,

$$\begin{cases} \dot{x_1} = \lambda_0 x_1 + \sigma \sum_{i=0}^{p-1} \xi_i(\omega)\phi_i(x_2)\ x_1, \\ \dot{x_2} = 1, & 0 \le t \le T \\ x_1(0) = x_0, \\ x_2(0) = 0. \end{cases} \tag{12.2}$$

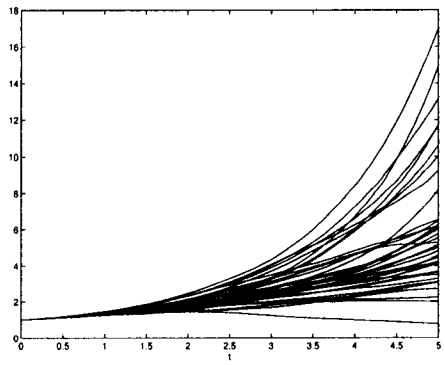

Figure 12.3: *Typical solution trajectories for equation (12.1), with* $\lambda_0 = 0.3, \sigma = 0.1, x_0 = 1$.

We compute solutions to the approximate problem, and solve the associated adjoint problem to obtain the sensitivities to the paramters. We try to approximate the solution, $x(5)$. As a reference solution, we approximate Brownian Motion on a very fine grid by accumulating normally distributed random variables with variance $\sqrt{\Delta t}$, and solve the differential equation numerically on the same grid. Solving for a sample of the distribution of $\boldsymbol{\xi}$ gives a reference distribution for $x(5, \omega)$. A FAPS with a small number of modes (Fig. 12.4) captures the density quite well.

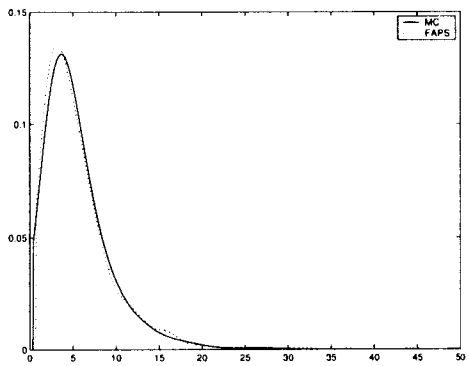

Figure 12.4: FAPS approximation for $x(5, \omega)$, with all the same parameters as in (Fig. 12.3) except $p = 5$, and 47 samples.

Chapter 13

INVERSE PARAMETER RANGE SELECTION

13.1 Introduction

In this chapter we address the problem of controlling the variance of the output in a model, given uncertatin input parameters. Since the input parameters can rarely be determined exactly, there is a range of possible output values. If this output range is too large, it is hard to obtain anything useful from a model, i.e. "anything is possible". Therefore, we pose the problem of prescribing a tolerance on the output variation for a given quantity of interest. We develop technique which may be used by a practioner to determine how the input parameter variances may be restricted to achieve a desired output variance. Such information could be used to guide the optimal placement of field measurements for certain input parameters.

13.2 One Point Solution

We first attempt to answer this question by using a one point approximation $q(\boldsymbol{\lambda}) \approx \tilde{q}(\boldsymbol{\lambda}) = q(\boldsymbol{\mu}) + \nabla q(\boldsymbol{\mu})(\boldsymbol{\lambda} - \boldsymbol{\mu})$, at the mean parameter $\boldsymbol{\mu}$. A simple calculation shows that $\text{var}[\tilde{q}(\boldsymbol{\lambda})] = \nabla q^{\top} C \nabla q$, where $C = E[(\boldsymbol{\lambda} - \boldsymbol{\mu})^{\top}(\boldsymbol{\lambda} - \boldsymbol{\mu})]$ is the covariance matrix for $\boldsymbol{\lambda}$. For the sake of calculation it is useful to weaken the goal of *guaranteeing* the quantity of

interest is within certain bounds to guaranteeing this condition with a high *confidence*, i.e.

$$P\left(\tilde{q}(\boldsymbol{\lambda}) \in [a,b]\right) > 1-\epsilon. \tag{13.1}$$

For instance, if $\boldsymbol{\lambda} \sim N(\boldsymbol{\mu}, C)$, $\tilde{q}$ is $N(q(\boldsymbol{\mu}), \sigma^2)$, with $\sigma^2 = \nabla q^\top C \nabla q$). Choosing the interval $[a,b]$ to be $q(\boldsymbol{\mu}) \pm \delta$, to satisfy (13.1), we solve for σ in

$$P\left(\tilde{q} \in [a,b]\right) = 1 - 2P\left(\tilde{q} \leq a\right) = -\mathrm{erf}\left(\frac{x - q(\boldsymbol{\mu})}{\sigma\sqrt{2}}\right) = 1-\epsilon,$$

to obtain the maximum allowed variance, σ^2. Since $\sigma^2 = \sum_{i,j} \partial_{\lambda_i} q \partial_{\lambda_j} q \, c_{i,j}$, and since the partial derivatives are fixed, we reduce each of the $c_{i,j}$. To obtain a desired σ^2, we assure that $|c_{i,j}| \leq \frac{\sigma^2}{N|\partial_{\lambda_i} q \partial_{\lambda_j}|}$, where N is the number of nonzero $c_{i,j}$. Thus, the allowed variance of the input parameter $\boldsymbol{\lambda}$ is prescribed.

13.3 Multi Point Case

The preceeding technique is limited by the accuracy of the linearization. To complement this technique for functions with worse nonlinearities, or for distributions with support large enough so that linearization is not accurate over the entire parameter range, we write

$$\begin{aligned}
\mathrm{var}(q) &= E\left[(q-\bar{q})^2\right] \\
&= \sum_i \int_{R_i} (q(\boldsymbol{z}) - \bar{q})^2 \, \mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}) = \sum_i \int_{R_i} (q(\boldsymbol{z}) - q_i + q_i - \bar{q})^2 \, \mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}) \\
&\leq 2\sum_i \int_{R_i} (q(\boldsymbol{z}) - q_i)^2 \, \mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}) + 2\sum_i \int_{R_i} (q_i - \bar{q}))^2 \, \mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z}) \\
&= \boldsymbol{I} + \boldsymbol{II},
\end{aligned}$$

where $\bar{q} = E[q]$, and $q_i = q(\boldsymbol{\mu}_i)$.

The first term is similar to the one point case but applied at each point. It may be approximated by

$$\boldsymbol{I} \approx \sum_i \nabla q(\boldsymbol{\mu}_i)^\top C_i \nabla q(\boldsymbol{\mu}_i),$$

where the matrix

$$C_i = \int_{R_i} (\boldsymbol{z} - \boldsymbol{\mu}_i)(\boldsymbol{z} - \boldsymbol{\mu}_i)^\top \, \mathrm{d}\mu_{\boldsymbol{\lambda}}(\boldsymbol{z})$$

measures the local variation in the probability measure.

The second term is something new, measure the extent of the range of q over the input distribution. For a nonlinear function each of these factors contributes to the variance, and they may operate independently, in contrast to the linear case where steep gradients and large ranges are directly tied together.

To control the output variance in this case both $\boldsymbol{I}$ and $\boldsymbol{II}$ must be corrected.

Chapter 14

SUMMARY AND FUTURE RESEARCH

This dissertation presents a suite of new methods for analyzing parameter uncertainty in deterministic nonlinear systems of differential equations. Some of the highlights of this research are presented here, with emphasis on the scientific impact of the techniques.

14.1 Summary

The first part of this dissertation provides an introduction to the standard technique for uncertainty analysis, the Monte-Carlo method. A comprehensive discussion of the common variance reduction techniques are presented, with a number of examples demonstrating their impact.

A detailed introduction to the adjoint provides a basis for the variational arguments that follow. This chapter brings together material from PDE theory, Functional Analysis, and Finite Element theory into a comprehensive framework. Bringing these various pieces into a single chapter is an education accomplishment, since the material often spans several textbooks, in different fields.

The representation theorems and perturbation results provide a means for researchers to calculate sensitivities to various quantities of interest.

These theorems could be used in data assimilation, and optimization, in addition to the uses in this dissertation.

The HOPS, RAPS, and FAPS methods are introduced as new methods for parameter sampling, which have shown (in the examples) to be very efficient ways to explore uncertainty. The response surfaces $\tilde{q}(\boldsymbol{\lambda})$ could be used for further explorations of the model behavior under parameter variation. Recovering the distribution of $q(\boldsymbol{\lambda})$ was the particular application explored in this dissertation. The HOPS method is demonstrated to be an extremely effecient method for a moderate number of parameters. The RAPS method is demonstrated to work mainly through the efficiency of the FAPS method, which is shown to compete with the "Curse of Dimensionality" by searching through parameter space only in directions of interest. This procedure also provides information on the relative sensitivities for each parameter, often leading to evidence for dimension reduction.

The brief introduction to the inverse problem could lead to very fruitful future work. This method could provide a way to "map" sensitivities throughout parameter space. Such information could help scientists make better parameter measurements in areas which contribute more to the uncertatinty, possibly leading to great savings.

This work has been interdisciplinary, combining pure mathematics, numerical methods, computer science, statistics and ecology. In particular, combining the use of the generalized Green's function with a statistical sampling method is a very new idea. The particular application of these techniques to the plague model represents an exploration involving all of the disciplines above. It is the authors hope that the techniques will be used in many fields using differential equations.

Bibliography

[1] Jean-Pierre Aubin. *Applied Functional Analysis.* Wiley-Interscience, New York, 2000.

[2] Patrick Billingsley. *Probability and Measure.* Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons Inc., New York, third edition, 1995. A Wiley-Interscience Publication.

[3] Megan Buzby. Quantitative sensitivity analysis of a vector-borne disease model using adjoint techniques, 2005. Masters Thesis.

[4] Yanzhao Cao, M. Yousuff Hussaini, and Thomas A. Zang. On the exploitation of sensitivity derivatives for improving sampling methods. *AIAA*, 2003.

[5] R. Courant and D. Hilbert. *Methods of Mathematical Physics. Vol. I.* Interscience Publishers, Inc., New York, N.Y., 1953.

[6] M. Csörgő and P. Révész. *Strong Approximations in Probability and Statistics.* Probability and Mathematical Statistics. Academic Press Inc. [Harcourt Brace Jovanovich Publishers], New York, 1981.

[7] Roger Eckhardt. Stan Ulam, John von Neumann, and the Monte Carlo method. *Los Alamos Science*, 1987.

[8] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson. Introduction to adaptive methods for differential equations. *Acta Numerica*, pages 105–158, 1995.

[9] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson. *Computational Differential Equations.* Cambridge University Press, New York, 1996.

[10] D. Estep. A posteriori error bounds and global error control for approximations of ordinary differential equations. *SIAM J. Numer. Anal.*, 32:1–48, 1995.

[11] D. Estep, M. Holst, and M. Larson. Generalized Green's functions and the effective domain of influence. *SIAM J. Sci. Comput.*, 2004. to appear.

[12] D. Estep and C. Johnson. The computability of the Lorenz system. *Math. Models Meth. Appl. Sci.*, 8:1277–1305, 1998.

[13] D. Estep and D. Neckels. Applying adjoint techniques to the methods of stratified sampling and control variates in Monte-Carlo computations, 2005. in preparation.

[14] D. Estep and D. Neckels. Fast and reliable methods for determining the evolution of uncertain parameters in differential equations. *J. Comput. Physics*, 2005. in revision.

[15] Donald J. Estep, Mats G. Larson, and Roy D. Williams. Estimating the error of numerical solutions of systems of reaction-diffusion equations. *Mem. Amer. Math. Soc.*, 146(696):viii+109, 2000.

[16] Lawrence C. Evans. *Partial Differential Equations*, volume 19 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, 1998.

[17] George S. Fishman. *Monte Carlo*. Springer Series in Operations Research. Springer-Verlag, New York, 1996. Concepts, algorithms, and applications.

[18] M.R. Garvie and Trenchea C. Global existence for reaction-diffusion systems modeling predator-prey interactions with the holling type ii functional response. *Submitted:Journal of Mathematiical Analysis and Applications*, 2004.

[19] Iosif I. Gikhman and Anatoli V. Skorokhod. *The Theory of Stochastic Processes. I.* Classics in Mathematics. Springer-Verlag, Berlin, 2004. Translated from the Russian by S. Kotz, Reprint of the 1974 edition.

[20] J. M. Hammersley and D. C. Handscomb. *Monte Carlo methods*. Methuen & Co. Ltd., London, 1965.

[21] Philip Hartman. *Ordinary Differential Equations*. Birkhäuser Boston, Mass., second edition, 1982.

[22] Harry Hochstadt. *Integral Equations*. John Wiley & Sons, New York-London-Sydney, 1973. Pure and Applied Mathematics.

[23] M. J. Keeling and C. A. Gilligan. Bubonic plague: a metapopulation model of a zoonosis. *Proc. R. Soc. Lond. B.*, 267(1458):2219–2230, 2000.

[24] Stig Larsson and Vidar Thomée. *Partial Differential Equations with Numerical Methods.* Springer-Verlag, Berlin, 2003.

[25] A. J. Nicholson and V. A. Bailey. The balance of animal populations. part I. *Proc. Zool. Soc. Lond.*, 3:551–598, 1935.

[26] R.S. Pathak. *A Course in Distribution Theory.* Narosa Publishing House, New Delhi, 2001.

[27] Lawrence Perko. *Differential Equations and Dynamical Systems*, volume 7 of *Texts in Applied Mathematics.* Springer-Verlag, New York, third edition, 2001.

[28] Wiebe R. Pestman. *Mathematical Statistics.* Walter de Gruyter, New York, 1998.

[29] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in C.* Cambridge University Press, Cambridge, second edition, 1992. The art of scientific computing.

[30] Robert Renka. Multivariate interpolation of large sets of scattered data. *ACM Trans. Math. Software*, 14:139–148, 1988.

[31] M. Rosenblatt. Remarks on a multivariate transformation. *Annals of Mathematical Statistics*, 1952.

[32] Eduardo D. Sontag. *Mathematical Control Theory*, volume 6 of *Texts in Applied Mathematics.* Springer-Verlag, New York, second edition, 1998. Deterministic finite-dimensional systems.

[33] Rajan Srinivasan. *Importance Sampling.* Springer-Verlag, Berlin, 2002. Applications in communications and detection.

[34] Francois Treves. *Basic Linear Partial Differential Equations.* Academic Press, Inc., New York, 1975.

Appendix A

SOFTWARE AND NUMERICS

In this appendix we collect implementation details for the various results used in this paper.

A.1 Parallel Monte-Carlo

The larger Monte-Carlo simulations in this project, especially those involving PDE's, were run on the Math Department's 64-processor linux cluster.

The parallelization of a large Monte-Carlo is relatively trivial, since each process is entirely independent of the other, and no effort must be made to communicate between the processes. We use MPI to run a process on each machine, gathering the results from time to time in a single process to save them on disk.

One must take care when generating the independent random numbers needed by the Monte-Carlo. Each process cannot call the C-library random number generator, because 1) if they use the same seed they get the same numbers, or 2) if they use different seeds they get different sequences which are not guaranteed to be component-wise independent. To deal with this, we generate a suitable sequence of random numbers in advance (using Matlab), and have each process load this file of numbers, using a specific "run"

of the numbers that is calculated based on the number of processors and number of computations each processor is to do.

A.2 FAPS code

The FAPS algorithm uses a set of C++ classes to grid the parameter space and adaptively sample.

A.2.1 Cell

The most fundamental class implements a generalized rectangle. Two vectors are stored, one for the "upper" corner, $\boldsymbol{u}$, and one for the "lower" corner, $\boldsymbol{l}$, so that the rectangle is exactly

$$R = \{\forall \boldsymbol{x} |\ l_1 \leq x_1 < u_1, l_2 \leq x_2 < u_2, \ldots, l_d \leq x_d < u_d\}.$$

At a particular stage in the adaptive algorithm, a list, Γ, of current rectangles (cells) is stored in an STL vector class, sorted in descending order by the appropriate *a-posteriori* error estimate. Some number of the rectangles with the largest error are selected for sampling, and each is divided into three cells (as described above), simply by forming the three rectangles,

$$\begin{aligned}
R_l &= \{\forall \boldsymbol{x} |\ l_1 \leq x_1 < u_1, l_2 \leq x_2 < u_2, \ldots, \\
&\quad l_i \leq x_i < l_i + \tfrac{1}{3}(u_i - l_i), l_d \leq x_d < u_d\}, \\
R_m &= \{\forall \boldsymbol{x} |\ l_1 \leq x_1 < u_1, l_2 \leq x_2 < u_2, \ldots, \\
&\quad l_i + \frac{1}{3}(u_i - l_i) \leq x_i < l_i + \tfrac{2}{3}(u_i - l_i), l_d \leq x_d < u_d\}, \\
R_r &= \{\forall \boldsymbol{x} |\ l_1 \leq x_1 < u_1, l_2 \leq x_2 < u_2, \ldots, \\
&\quad l_i + \frac{2}{3}(u_i - l_i) \leq x_i < u_i, \leq x_d < u_d\},
\end{aligned}$$

where the i^{th} dimension is chosen for refinement.

On each of the new cells R_l, R_r, a new computation is started by opening a pipe to the program responsible for calculation $q, \nabla q$. When the results are obtained, the cell object uses this data to compute the error contribution by calling up a distribution object (described below) to integrate the error estimates over the cell. The new cells are then added to the list Γ in sorted order, starting from the beginning of Γ.

A.2.2 Distributions

In order to support different input parameter distributions, we use *polymorphism*. In C++ this means we define an abstract *distribution* class, with the all the necessary interface functions declared *virtual*. For each different distribution object, we write a derived class, in which we implement the necessary capabilities to provide the interface for the given distribution. This approach allows the algorithm code (code using distribution objects) to operate on a number of different distributions without special hooks or cases for each different type. If also makes adding new distribution very easy, since the algorithm is isolated from the implmentation of the new object.

The main task required is the ability to integrate a (user provided) function over a given rectangle. Each distribution provides an integration routine that calls the user's function at some quadrature points, weights this integral according to the appropriate measure, and returns the value.

For some of the advanced algorithms, the distribution is required to return samples consistent with conditioning on a given rectangle. This is done by using an accept-reject method, bounding the density of the distri-

bution on the rectangle, and using this as the bound in the accept-reject algorithm.

Each of the objects below assumes independent components. Most of the integrals required by FAPS thus reduce to iterated integrals along each dimension.

Uniform distribution

This is the simplest to implement. The distribution must be supported on an initial rectangle, which the user provides. The integrations reduce to standard quadrature rules.

Data set distribution

The user may provide a data set, which induces the measure with a $\frac{1}{N}\delta$ concentrated at each data point. To integrate over a given rectangle, the object finds the points contained in the rectangle, evaluates the function at those points, sums, and multiplies by $\frac{1}{N}$.

Truncated normal distributions

Given the normal random variable $\boldsymbol{\lambda}$, we form the truncated random variable $\hat{\boldsymbol{\lambda}}$ with support B by conditioning $\boldsymbol{\lambda}$ on $[\boldsymbol{\lambda} \in B]$.

The resulting density for $\hat{\boldsymbol{\lambda}}$ may be expressed in terms of the density for $\boldsymbol{\lambda}$,

$$f_{\hat{\boldsymbol{\lambda}}}(\boldsymbol{x}) = \frac{1}{C} f_{\boldsymbol{\lambda}}(\boldsymbol{x}) \chi_B(\boldsymbol{x}),$$

where

$$C = \int_B f_{\boldsymbol{\lambda}}(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x}$$

and χ_B is the indicator function of B.

In the case that $\boldsymbol{\lambda}$ is comprised of independent components, and $B = \prod_i [\mu_i - 0.5w_i, \mu_i + 0.5w_i]$,

$$C = \int_B f_{\boldsymbol{\lambda}}\, \mathrm{d}x = \prod_{i=1}^{k} \int_{\mu_i - 0.5w_i}^{\mu_i + 0.5w_i} f_{\lambda_i}\, \mathrm{d}x = \prod_{i=1}^{k} \text{-erf}\left(\frac{-w_i}{\sqrt{2}\sigma_i}\right),$$

where μ_i, σ_i are the parameters for the i^{th} component, λ_i, of $\boldsymbol{\lambda}$.

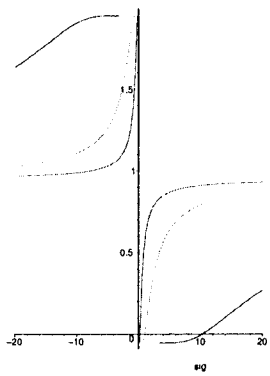

Figure A.1: Plots of $F(\sigma_i)$ for $w_i = 0.5, 2, 10, 20$. Newton's method performs very poorly for these functions when the initial guess is not close to the root, so a Quasi-Newton's method is used with stepsize control.

The code allows the user to specify the means, the widths, and a mass parameter $0 < \alpha \leq 1$. The software solves for the appropriate standard deviations σ_i so that

$$\int_{\mu_i - 0.5w_i}^{\mu_i + 0.5w_i} f_{\lambda_i} \, \mathrm{d}x = \alpha.$$

A value of α near 1 guarantees that most of the mass of the bell curve is in the over the finite interval which supports $f_{\boldsymbol{\lambda}}$. We solve for this parameter by posing the fixed point problems

$$\begin{aligned} F(\sigma_i) &= \alpha - \int_{\mu_i - 0.5w_i}^{\mu_i + 0.5w_i} f_{\lambda_i} \, \mathrm{d}x \\ &= \alpha + \mathrm{erf}\left(\frac{-w_i}{\sigma\sqrt{2}}\right) = 0, \end{aligned}$$

for $i = 1, \ldots, p$ and solving using a Quasi-Newton method iteration

$$\sigma_i^{n+1} = \sigma_i^n - \frac{F(\sigma_i^n)}{F'(\sigma_i^n)}.$$

This iteration behaves very poorly for bad a initial guess Fig. A.1, so we improve the performance by evaluating F at each new point and setting

$\sigma_i^{n+1} = \sigma_i^n + \frac{1}{2}(\sigma_i^n - \sigma_i^{n+1})$ when $|F(\sigma_i^{n+1})| > |F(\sigma_i^n)|$ (indicating no improvement has been made). We also use the fact that $\sigma_i > 0$, by setting $\sigma_i^{n+1} = 0.5\sigma_i^n$ whenver the standard Newton iteration yields a negative σ_i^{n+1}.

A.3 Solving the ODE's

The ODE's in this paper are solved with a C++ integrator object, which operates very much like Matlab's ODE45. The user provides a C function that returns the derivatives given the current state vector and parameters. The integrator can perform a variety of time stepping algorithms, and the one used for the examples is the Runge-Kutta fourth-order algorithm, which solves the ODE,

$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, t), \\ \boldsymbol{x}(t_0) = \boldsymbol{x}_0, \end{cases}$$

by the update algorithm,

$$\begin{aligned} \boldsymbol{k}_1 &= \boldsymbol{f}(\boldsymbol{X}_n, t_n) \\ \boldsymbol{k}_2 &= \boldsymbol{f}(\boldsymbol{X}_n + \frac{\boldsymbol{k}_1}{2}, t_n + \frac{h}{2}) \\ \boldsymbol{k}_3 &= \boldsymbol{f}(\boldsymbol{X}_n + \frac{\boldsymbol{k}_2}{2}, t_n + \frac{h}{2}) \\ \boldsymbol{k}_4 &= \boldsymbol{f}(\boldsymbol{X}_n + \boldsymbol{k}_3, t_n + h) \\ \boldsymbol{X}_{n+1} &= \boldsymbol{X}_n + h(\frac{\boldsymbol{k}_1}{6} + \frac{\boldsymbol{k}_2}{3} + \frac{\boldsymbol{k}_3}{3} + \frac{\boldsymbol{k}_4}{6}). \end{aligned}$$

The truncation error (per step) is $O(h^5)$, and so the error $|\boldsymbol{X}(t_n) - \boldsymbol{x}(t_n)|$, where $\boldsymbol{x}$ represents the true solution, is of order h^4.

Extensive use of Maple was made to code the vector fields, $D_{\boldsymbol{x}}\boldsymbol{f}$, and $D_{\boldsymbol{u}}\boldsymbol{f}$. The vector field $\boldsymbol{f}$ is input, and each of these matrices is formed by symbolically differentiating the vector field and automatically generating C code with the CodeGen package. It is the author's experience that this

approach saves a tremendous amount of time in debugging, since it avoids the possibility of typo's, except at the single point of entering the vector field.

A.4 Solving the PDE's with DFE

The partial differential equations were solved with the author's finite element code DFE (David's Finite Element), a C++ finite element code. This code implements a subspace of linear (or quadratic) piecewise continuous polynomials, on a triangulation, $\mathcal{T}_h$, of the region Ω. This subspace, $V_h \subset H^1(\Omega)^m$, and the numerical method employed by the code, to solve the system (5.8), is to find for each time t_n a function $\boldsymbol{U}_n \in V_h$ such that

$$(\bar{\partial}_t \boldsymbol{U}_n, \boldsymbol{v})_{L_2} + (a\nabla \boldsymbol{U}_n, \nabla \boldsymbol{v})_{L_2} = (\boldsymbol{f}(\boldsymbol{U}_{n-1}, \boldsymbol{\lambda}), \boldsymbol{v})_{L_2}, \forall \boldsymbol{v} \in V_h,$$

where $\bar{\partial}_t \boldsymbol{U}_n = k^{-1}(\boldsymbol{U}_n - \boldsymbol{U}_{n-1})$ is the *backward Euler* difference operator.

Expanding $\boldsymbol{U}_n$ in terms of a finite dimensional basis $\boldsymbol{\phi}_j, j = 1, \ldots, M_h$, with coefficients $\boldsymbol{\alpha}_n$, this leads to a system of algebraic equations,

$$B\boldsymbol{\alpha}_n + kA\boldsymbol{\alpha}_n = B\boldsymbol{\alpha}_{n-1} + k\boldsymbol{b}_n.$$

In DFE, this matrix equation is solved using an iterative scheme, namely the *conjugate gradient method.*

This method of discretization uses the finite element method to handle the spacial derivatives, and the finite difference method in time. It may be shown [24] that for a suitable family of triangulations $\mathcal{T}_h$, and given certain conditions on $\boldsymbol{F}$, that

$$\|\boldsymbol{U}_n - \boldsymbol{u}(t_n)\|_{L^2} \leq C_1 h^2 + C_2 k,$$

where $\boldsymbol{u}$ is the true solution of (5.8), and the constants, $C_{1,2}$, are independent of h and k. The parameter k is the time step, and $h = \max_{K \in \mathcal{T}_h} \operatorname{diam}(K)$ is a measure of the fineness of the spatial discretization. This result says that by refining the mesh in space and simultaneously reducing the time step, the numerical solution converges to the true solution.

A.5 Density Plots

At the end of the RAPS and FAPS procedures, the piecewise constant function $\tilde{q}(\boldsymbol{\lambda})$ leads immediately to an approximation of $F_{\tilde{q}}(x) = \sum_{q(\bar{\boldsymbol{\mu}}_i) \leq x} \mu_{\boldsymbol{\lambda}}(R_i)$. At the end of the HOPS procedure, the piecewise linear function $\tilde{q}(\boldsymbol{\lambda})$ may either be converted to a piecewise constant by applying the FAPS procedure to this linear approximation, or be converted into a sample of points by applying $\tilde{q}$ to a sample of points from $\boldsymbol{\lambda}$. In any case, the function

$$F_{\tilde{q}}(x) = \frac{1}{N} \{\text{Number of points} \leq x\}$$

naturally approximates F_q.

In general, depending on $\boldsymbol{\psi}$ and the differential equation, we expect $q(\omega)$ to have a density function $f_q(x)$, and since plots of a density are more illuminating than plots of a cumulative distribution (compare Fig. 9.3 to Fig. 9.4), we usually approximate the density. However, the cumulative distribution functions created by the FAPS and HOPS algorithms are not differentiable, and so these approximating distributions do not have associated densities.

To approximate f_q, we take any density $k(x)$, define $k_\zeta(x) = k(x/\zeta)/\zeta$, $K_\zeta = \int_{-\infty}^{x} k_\zeta(x)\,\mathrm{d}x$, and

$$k_\zeta * F_{\tilde{q}}(x) = \int_{-\infty}^{\infty} K_\zeta(x - y)\,\mathrm{d}F_{\tilde{q}}(y).$$

This gives an approximation $K_\zeta * F_{\tilde{q}}$ to $F_{\tilde{q}}$ that is differentiable and

$$(K_\zeta * F_{\tilde{q}})'(x) = k_\zeta * F_{\tilde{q}}(x) = \int_{-\infty}^{\infty} k_\zeta(x-y)\,\mathrm{d}F_{\tilde{q}}(y) = \frac{1}{\zeta}\sum_{i=1}^{N} k\left(\frac{x-q(\bar{\boldsymbol{\mu}}_i)}{\zeta}\right)\bar{\boldsymbol{\mu}}_{\boldsymbol{\lambda}}(R_i)$$

in the case $F_{\tilde{q}}$ is computed from the piecewise constant approximation, or

$$(K_\zeta * F_{\tilde{q}})'(x) = \frac{1}{\zeta N}\sum_{i=1}^{N} k\left(\frac{x-q(\bar{\boldsymbol{\mu}}_i)}{\zeta}\right)$$

when the cumulative distribution function is constructed from a data set of N points. A typical choice of k is the standard $N(0,1)$ density function.

As $\zeta \to 0$ and $F_{\tilde{q}} \to F_q$, formally

$$|K_\zeta * F_{\tilde{q}}(x) - F_q(x)| = \int_{-\infty}^{\infty} (F_{\tilde{q}}(x-y) - F_q(x-y))\,k_\zeta(y)\,\mathrm{d}y + \int_{-\infty}^{\infty} (F_q(x-y) - F_q(x))\,k_\zeta(y)\,\mathrm{d}y \to 0 + \delta(y-0)\,(F_q(x-y) - F(x)) = 0,$$

and so $K_\zeta * F_{\tilde{q}}$ is a differentiable cumulative distribution function that approximates F_q. Choosing the "bandwidth" parameter ζ is somewhat subjective, and is a balance between increasing the variance of the approximating density on the one hand, or the bias on the other [28].

A.6 Computing Error Integrals for FAPS

The FAPS method requires the evaluation of the errors $\mathcal{E}_{i,k}^{pc}$ (8.1) with respect to the measure $\mu_{\boldsymbol{\lambda}}$. There are at least two ways to do this, depending on how the distribution of $\boldsymbol{\lambda}$ is represented numerically.

In the first method a list of M data points representing the distribution of $\boldsymbol{\lambda}$ (a sample) is read in. The measure $\mu_{\boldsymbol{\lambda}}$ is approximated by the discrete measure that assigns

$$\mu_{\boldsymbol{\lambda}}(A) = \frac{1}{M}(\text{Number of points in } A).$$

The approximate error then becomes

$$\mathcal{E}_{i,k}^{pc} \approx \frac{1}{M} \sum_{z_j \in R_i} |\partial_{\lambda_k} q(\bar{\boldsymbol{\mu}}_i)(z_j^k - \mu_i^k)|,$$

where the $z_j = (z_j^1, \ldots, z_j^d)^\top$ are the points from the sample that lie in R_i.

If instead we choose to represent the distribution analytically, where the distribution has a density $\rho_{\boldsymbol{\lambda}}(\boldsymbol{z})$, the integral becomes

$$\mathcal{E}_{i,k}^{pc} = \int_{R_i} |\partial_{\lambda_k} q(\bar{\boldsymbol{\mu}}_i)(z^k - \mu_i^k)| \rho_{\boldsymbol{\lambda}}(\boldsymbol{z})\, \mathrm{d}\boldsymbol{z},$$

which may be computed using standard quadrature rules. In the special case that the components of $\boldsymbol{\lambda}$ are independent, then the density may be factored $\rho_{\boldsymbol{\lambda}}(\boldsymbol{z}) = \prod_{i=1}^d \rho_{\lambda_i}(z^i)$ and so the error becomes

$$\begin{aligned}\mathcal{E}_{i,k}^{pc} &= \int_{R_i} |\partial_{\lambda_k} q(\bar{\boldsymbol{\mu}}_i)(z^k - \mu_i^k) \rho_{\lambda_1}(z^1) \cdots \rho_{\lambda_p}(z^d)\, \mathrm{d}\boldsymbol{z} \\ &= \frac{\mu_{\boldsymbol{\lambda}}(R_i)}{b} \int_{R_i^k} |\partial_{\lambda_k} q(\bar{\boldsymbol{\mu}}_i)(z^k - \mu_i^k)| \rho_{\lambda_k}(z^k)\, \mathrm{d}z^k\end{aligned}$$

where R_i^k is the interval along the k^{th} side of R_i and $b = \int_{R_i^k} \rho_{\lambda_k}\, \mathrm{d}z^k$. This integral for large d is much simpler to compute, since it involves only two 1-dimensional integrals as opposed to an integral over $\mathbb{R}^d$. In particular, the curse of dimensionality is avoided in this case.

For a density with non-independent components, we propose the use of a transformation T, writing $q(\boldsymbol{\lambda}) = q \circ T^{-1} \circ T(\boldsymbol{\lambda})$, where the random variable $\boldsymbol{u} = T(\boldsymbol{\lambda})$ has independent components. Setting $q_1 = q \circ T^{-1}$, we proceed as above. Such transforms are often available; the Rosenblatt transform, for example, may be used to transform a multivariate-Gaussian into a collection of i.i.d uniform variables [31].

A.7 Computation of Gradients

In practice, we want to numerically approximate the derivative in (5.7) and apply the derivative to points $\boldsymbol{\lambda} - \bar{\boldsymbol{\mu}}$. It is necessary to rewrite the integral (second) term in the derivative, since we wish to avoid calculating an integral for many points in the distribution of $\boldsymbol{\lambda}$. We may rewrite the derivative with respect to the $\boldsymbol{\lambda}_1$ as

$$\partial_{\boldsymbol{\lambda}_1} q(\bar{\boldsymbol{\mu}})[\cdot] = \langle [\cdot], \int_0^T D_{\boldsymbol{\lambda}_1} \boldsymbol{f}(\boldsymbol{y}; \bar{\boldsymbol{\mu}}_1)^T \boldsymbol{\phi} \, \mathrm{d}s \rangle,$$

so that we do not have to integrate random variables to calculate the derivative. Rather, we calculate the derivative and perform a dot product.

We also note that for $i \leq p$,

$$\partial_{\lambda_i} q(\bar{\boldsymbol{\mu}}) = \partial_{\boldsymbol{\lambda}} q(\bar{\boldsymbol{\mu}}) \boldsymbol{e}_i = \sum_{j=1}^{n} \int_0^T \partial_{\lambda_i} f_j(\boldsymbol{y}; \bar{\boldsymbol{\mu}}_1) \phi_j \mathrm{d}s, \tag{A.1}$$

where $\boldsymbol{f} = (f_1, \ldots, f_n)^\top$ and $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_n)^\top$. Once we have numerically solved for $\boldsymbol{y}$ and $\boldsymbol{\phi}$, the integrals may be computed on the time grid by any quadrature rule. For instance, if we solve the forward and adjoint problems on the mesh $t_0 = 0 < t_1 < \cdots < t_m = T$,

$$\int_0^T \partial_{\lambda_i} f_j(\boldsymbol{y}; \bar{\boldsymbol{\mu}}_1) \phi_j \mathrm{d}s \approx$$
$$\sum_{k=0}^{m-1} \frac{1}{2} (\partial_{\lambda_i} f_j(\boldsymbol{y}(t_{k+1}); \bar{\boldsymbol{\mu}}_1) \phi_j(t_{k+1}) + \partial_{\lambda_i} f_j(\boldsymbol{y}(t_k); \bar{\boldsymbol{\mu}}_1) \phi_j(t_k))(t_{j+1} - t_j) \tag{A.2}$$

is the trapezoidal rule.

A.8 Details for computing the adjoint solution

There are a few details about solving the adjoint problem worth mention. The adjoint as posed in (5.3) is given as a final value problem, so to

solve this numerically we first make this a forward time problem by setting $\boldsymbol{\Phi}(t) = \boldsymbol{\phi}(T-t)$. Then $\boldsymbol{\Phi}$ solves the forward problem

$$\begin{cases} \dot{\boldsymbol{\Phi}}(t) - A(T-t)^T \boldsymbol{\Phi}(t) = \boldsymbol{\psi}(T-t), & 0 \le t \le T, \\ \boldsymbol{\Phi}(0) = \mathbf{0}, \end{cases} \tag{A.3}$$

In the particular case that $\boldsymbol{\psi}(t) = \delta(t-t_0)\boldsymbol{e}_i, t_0 < T$, where $\boldsymbol{e}_i$ is the the i^{th} standard basis vector, we have the following result,

Claim 2. *If $\boldsymbol{\Phi}$ solves*

$$\begin{cases} \dot{\boldsymbol{\Phi}}(t) - A^T(T-t)\boldsymbol{\Phi}(t) = \mathbf{0}, & T - t_0 \le t \le T, \\ \boldsymbol{\Phi}(t) = \mathbf{0}, & 0 \le t < T - t_0, \\ \boldsymbol{\Phi}(T-t_0) = \boldsymbol{e}_i, \end{cases}$$

then the representation (5.4) holds for $\boldsymbol{\phi}(t) = \boldsymbol{\Phi}(T-t)$.

Note that from the argument leading to the representation (5.4) shows that the result is true if we can prove

$$\langle \boldsymbol{v}(0), \boldsymbol{\phi}(0) \rangle + \int_0^T \langle \dot{\boldsymbol{v}} - A\boldsymbol{v}, \boldsymbol{\phi} \rangle \, \mathrm{d}s = \langle \boldsymbol{v}(t_0), \boldsymbol{e}_i \rangle = v_i(t_0)$$

for any $\boldsymbol{v} \in C^1[0,T]$. Now

$$\begin{aligned} \int_0^T \langle \dot{\boldsymbol{v}} - A\boldsymbol{v}, \boldsymbol{\phi} \rangle \, \mathrm{d}s &= \int_0^{t_0-\epsilon} \langle \dot{\boldsymbol{v}} - A\boldsymbol{v}, \boldsymbol{\phi} \rangle \, \mathrm{d}s + \int_{t_0+\epsilon}^T \langle \dot{\boldsymbol{v}} - A\boldsymbol{v}, \boldsymbol{\phi} \rangle \, \mathrm{d}s \\ &+ \int_{t_0-\epsilon}^{t_o+\epsilon} \langle \dot{\boldsymbol{v}} - A\boldsymbol{v}, \boldsymbol{\phi} \rangle \, \mathrm{d}s = [\langle \boldsymbol{v}(t_0-\epsilon), \boldsymbol{\phi}(t_0-\epsilon) \rangle - \langle \boldsymbol{v}(0), \boldsymbol{\phi}(0) \rangle] \\ &+ [\langle \boldsymbol{v}(T), \boldsymbol{\phi}(T) \rangle - \langle \boldsymbol{v}(t_0+\epsilon), \boldsymbol{\phi}(t_o+\epsilon) \rangle] + \int_{t_0-\epsilon}^{t_0+\epsilon} \langle \dot{\boldsymbol{v}} - A\boldsymbol{v}, \boldsymbol{\phi} \rangle \, \mathrm{d}s, \end{aligned}$$

where we have integrated by parts where $\boldsymbol{\phi}$ is smooth, and used the fact that $-\dot{\boldsymbol{\phi}} - A^T\boldsymbol{\phi} = \mathbf{0}$ on these regions. Regularity of $\boldsymbol{v}$ and $\boldsymbol{\phi}$ insure that

$$\int_{t_0-\epsilon}^{t_o+\epsilon} \langle \dot{\boldsymbol{v}} - A\boldsymbol{v}, \boldsymbol{\phi} \rangle \, \mathrm{d}s \to 0 \quad \text{as} \quad \epsilon \to 0$$

(these properties actually depend on A and the original differential equation, but are true for a large class of problems). Also, since $\boldsymbol{\phi}(T) = \boldsymbol{\Phi}(0) = \mathbf{0}$ and $\boldsymbol{\phi}(t_0 + \epsilon) = \boldsymbol{\Phi}(T - t_0 - \epsilon) = \mathbf{0}$

$$\begin{aligned}\int_0^T \langle \dot{\boldsymbol{v}} - A\boldsymbol{v}, \boldsymbol{\phi} \rangle \, \mathrm{d}s + \langle \boldsymbol{v}(0), \boldsymbol{\phi}(0) \rangle &= \lim_{\epsilon \to 0} \langle \boldsymbol{v}(t_0 - \epsilon), \boldsymbol{\phi}(t_0 - \epsilon) \rangle \\ &= \lim_{\epsilon \to 0} \langle \boldsymbol{v}(t_0 - \epsilon), \boldsymbol{\Phi}(T - t_0 + \epsilon) \rangle \\ &= \langle \boldsymbol{v}(t_0), \boldsymbol{e}_i \rangle = v_i(t_0),\end{aligned}$$

by the continuity of $\boldsymbol{v}$ and right continuity of $\boldsymbol{\Phi}$.

In order to solve the adjoint problem, we solve (A.3) by setting the initial condition at t_0 to $\boldsymbol{e}_i$ and integrating forward in time. Effectively, we may assume that $T = t_0$ since $\boldsymbol{\Phi} = \mathbf{0}$ between T and t_0.