A MULTI-RESOLUTION APPROACH TO AN INVERSE CONDUCTIVITY PROBLEM

by

Kenneth Chien-ko Chou

S.B., Massachusetts Institute of Technology (1985)

SUBMITTED TO THE DEPARTMENT OF
ELECTRICAL ENGINEERING AND COMPUTER SCIENCE
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREES OF

ELECTRICAL ENGINEER
and
MASTER OF SCIENCE
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

December 1987

Signature of Author _____
Department of Electrical Engineering and Computer Science
Dec. 2, 1987

Certified by _____
Alan S. Willsky
Thesis Supervisor

Accepted by _____
Arthur Smith
Chairman, Department Committee on Graduate Students

A Multi-Resolution Approach to An Inverse Conductivity Problem

by

Kenneth Chien-ko Chou

Submitted to the Department of Electrical Engineering
and Computer Science on Dec. 2, 1987 in partial
fulfillment of the requirements for the degrees of
Electrical Engineer and Master of Science

Abstract

In this work we study an inverse conductivity problem in which the fundamental problem is one of estimating the conductivity, a 2D function, in a bounded domain based on excitations and measurements along the boundary of the domain. The estimation problem is non-linear and the number of degrees of freedom in the parameter is extremely large. We investigate the idea of estimating the conductivity at multiple spatial resolutions in an attempt to control the large number of degrees of freedom. We derive performance bounds for specific cases that show how measurement noise limits estimation performance for a particular scale. These bounds provide a way of characterizing what scale a set of excitations and measurements supports. We also develop a multi-resolution algorithm that estimates the conductivity at successively finer scales. At each scale the algorithm consists of a sequence of linear relaxation schemes which are extremely parallelizable. We demonstrate the success of our algorithm on synthetic data while exploring algorithmic issues and studying the effects of noise on algorithm performance.

Thesis Supervisor: Alan S. Willsky
Title: Professor of Electrical Engineering

Acknowledgements

I would like to thank my thesis supervisor, Alan Willsky, for starting me off on this thesis in the first place and in general for making my first experiences with serious research challenging, interesting, and in the end extremely rewarding. He has been a boundless source of ideas, enthusiasm and insight. I would also like to acknowledge the contributions of Bernard Levy, who served as a second advisor for a large part of this research. My discussions with him on inverse problems and his extensive knowledge of the literature on the subject have been most helpful.

All of my close friends and family deserve mention here, for without them life would be significantly less meaningful. They are Pat and Shir, Katherine Barnum, Huang Mosian, Keith Nabors, Pat Li, Greg Williams, and my buddies from A^2.

My officemates, past and present, have been the true source of education for me at MIT. Their friendship and their constant willingness to discuss a wide of range of ideas have been most valuable. They are Carey Bunks, Mike Chin, Adam Ciarmicoli, Richard Lamb, Ramine Nikoukah, Cuneyt Ozveren, Jerry Prince and Ahmed Tewfik.

Undoubtedly, the most important people in terms of my general development and outlook on life have been my parents. They have always been supportive in my educational endeavors and have always stressed the importance of doing one's best.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

Inverse problems are particularly difficult for a variety of reasons. For example, if we view inverse problems from a system identification standpoint there exists the often non-trivial task of choosing a parametrization of the system, i.e. a formal model, which can be used in performing parameter estimation. Once a parametrization is chosen, in order to do parameter estimation one must then choose a criterion of optimality for the estimate. Given a useful parametrization of the system and criterion of optimality the parameter estimation problem itself may still be formidable with repect to the task of developing a good algorithm for computing the optimal estimate. This is especially the case when the parameter has a large number of degrees of freedom. One such case arises when the parameter is a spatial function of more than one dimension. Failing to address properly the fact that there are a large number of degrees of freedom in a problem can lead to unsuccessful algorithms. In particular, the attempt at processsing all the degrees of freedom at once often leads to numerically unstable and computationally inefficient algorithms.

In this thesis we look at an inverse conductivity problem with our focus being on the parameter estimation aspect of this problem. The fundamental problem is one of estimating a 2D conductivity function in a bounded domain based on a set of excitations and measurements along the boundary of the domain. Our work suggests a way to control the number of degrees of freedom in this problem by estimating the parameter at multiple spatial resolutions. We derive the maximum-likelihood solution for the problem and present a multi-resolution algorithm for computing this solution based on the idea of estimating the conductivity at successively finer spatial scales.

At each scale the algorithm we develop is composed of a sequence of linear, highly parallelizable relaxation schemes. In particular, we develop an iterative, relaxation approach to solving for the best conductivity estimate at several spatial scales, with estimates from coarser scales used to guide the finer scale iterations. The sequence of steps at any scale consists of the solution of *linear* estimation problems and also can be directly implemented in a highly parallel fashion. We demonstrate the results of our algorithm on synthetic data.

We also derive analytical bounds on estimation performance for a variety of cases. We have numerical results which demonstrate the usefulness of these bounds and in particular their ability to give further insight into the approach of estimating the conductivity at various spatial scales. These bounds also provide a useful tool for determining what scale a set of excitations and measurements support.

1.2 Problem Motivation

The idea of imaging the electrical conductivity in a cross-section of an object by numerical inversion of low-frequency, electromagnetic boundary data has applications in various fields of engineering; e.g. medical imaging and exploratory geophysics. For example, a particular geophysical scheme proposed for use in mapping conductivity within core samples can be found in [3]. In this scheme electrodes are placed around the boundary of the cross-section of a core sample. The voltages of each electrode are independently controlled and measurements of the current at each electrode are made. From these boundary data the conductivity image within the cross-section is to be inferred.

In [3] the conductivity image is modelled as a network of resistors. Then, Kirchoff's current law is used to relate the data(volatages and currents on the boundary) to the unknowns(resistances and interior voltages in the network). The solution is obtained by iterative optimization of this non-linear function. Because the problem involves a large number of unknowns for reasonably fine grid resolution, the resulting algorithm is computationally intensive involving the repeated inversion of large matrices. Also, the optimization scheme is prone to giving solutions corresponding to local minima of the cost function if not intitiated close to the solution. These pitfalls are indicative of some of the issues with which we are concerned in our research.

This thesis focuses on the inverse conductivity problem stated as follows: estimate the conductivity within the unit square, a 2D scalar function, based on a set of experiments, each of which consists of applying a known potential distribution along the boundary and measuring the current normal to the boundary. We view

the inverse problem as essentially one of multi-dimensional parameter estimation where the conductivity represents the parameter to be identified. Our solution is stated in terms of optimality with respect to a criterion that relates the parameter to be estimated with both the data and the physics of the problem. A similar approach to inverse problems in geophysics is discussed in [10], [11].

The particular criterion of optimality we choose is based on a likelihood function which we maximize in determining the optimal estimate. This maximum-likelihood approach can also be viewed as a weighted least-squares approach for the case of Gaussian measurement noise. Ultimately, we solve an optimization problem in which we minimize a cost function with certain constraints on the parameters. We solve this constrained problem by minimizing an augmented cost function in which a penalty function is used to represent these constraints[1]. The reason for choosing this penalty method is that it allows us to spatially separate the problem into a collection of simpler subproblems.

In seeking a computationally efficient algorithm we explore the idea of solving the inverse problem at various spatial resolutions, starting at a very coarse resolution then progressing to finer and finer resolutions. The main idea is that by starting at coarse resolutions involving fewer computations, then building up to finer and finer resolutions using information from previous resolutions, we arrive at a more computationally tractable algorithm, one that converges faster than an algorithm aimed at solving the problem exclusively at the finest resolution. Moreover, it is reasonable that a multi-resolution method would help in avoiding local minima. The idea is that by solving a sequence of problems at successively finer scales the

[1]For this approach to yield the precise solution we must in principle solve a set of problems in which the weight on the penalty term approaches inf. Obviously, we stop well short of this and consequently the constraints in the problem are only approximately satisfied.

estimate is likely to be guided towards the global minimum of the problem at the finest scale.

Aside from computational efficiency, the idea of being able to estimate the parameter at various scales is useful in its own right. The fact that estimating at a very fine scale may be difficult should not prohibit estimating features at very coarse scales. Our approach suggests a way of estimating coarse scale features efficiently, while also being able to estimate finer scale features by using information from coarser scales.

We are also interested in the issue of determining the scale at which inversion is supported by a particular set of excitations and measurements. Our approach is to derive Cramer-Rao bounds on estimation performance that show explicitly how the physics of the problem, the experiments performed, and measurement noise limit performance at a specific scale. The computation of these bounds at various scales essentially gives us the ability to determine the scale at which our measurements allow us to estimate the parameters.

The idea of performing computations on a parameter at various scales has been successfully implemented in a variety of problems under the general name of multigrid methods [2]. The use of multigrid methods has been explored to a large extent in the solution of partial differential equations with much work being done on elliptic PDE's [5], [8]. The methods have also been applied to image processing where typically problems are ill-posed and solutions are presented in terms of variational calculus [12]. The use of multigrid methods in image processing up to this point has been confined mainly to forward problems, i.e. the solution of PDE's with possible constraints.

Our problem can be viewed as one of estimating the coefficients(in fact spatially varying in this case) of a PDE based on a set of Dirichlet boundary conditions considered to be known excitations and a set of Neumann boundary conditions considered to be measurements. The choice of parametrization and criterion for solution in this case is much less obvious than in the case of solving the forward problem, i.e. solving the PDE based on boundary conditions and known coefficients.

One last note with regard to the relationship of our work to the work being done on multigrid methods. Whereas in multigrid methods the algorithms developed move from both coarse to fine scales and fine to coarse scales, the work in this thesis concentrates on the direction of coarse to fine. We acknowledge this difference as indication that further work on investigating a full multigrid implementation for this problem is highly warranted.

1.3 Thesis Overview

The following is an overview of the thesis describing the contents of each chapter.

In Chapter 2 we formulate the problem in an estimation-theoretic framework using the appropriate equations of mathematical physics and a 2D piecewise constant parametrization for the conductivity function. Based on this parametrization and a model for our observations we derive the optimization problem that yields the maximum-likelihood estimate for the 2D piecewise constant conductivity function.

In Chapter 3 we describe our multi-resolution algorithm. The algorithm consists of a sequence of iterative algorithms generating estimates at successively finer spatial scales. The estimate at each scale is used to initialize the iteration at the next scale. The iterative algorithm at each scale consists of an alternating sequence of linear

estimators, each of which is iterative and highly parallel. We give proofs of the convergence of these algorithms.

In Chapter 4 we study the case of estimating a constant conductivity background. We consider the effects of estimating a constant background when in fact the true background is spatially varying. Based on a linearization of the PDE we derive approximations to the bias and mean-square error for this case. We also plot performance characteristics based on Monte-Carlo simulations.

In Chapter 5 we study the case in which the conductivity is piecewise constant in four distinct regions. We derive the Cramer-Rao lower bound on the mean-square error of the estimate and numerically compute this bound for various excitation schemes and various conductivity backgrounds. In this chapter we also demonstrate the performance of our algorithm on synthetic data.

Chapter 6 focuses on the the case in which the conductivity is piecewise constant in sixteen distinct regions. We demonstrate the performance of our algorithm on synthetic data. We compare the performance, i.e. speed of convergence and accuracy, of our algorithm when we use information from the previous scale to the performance of our algorithm when we don't use this information. We also examine the ill-posedness of the problem by investigating the presence of local minima of the cost function.

Finally, Chapter 7 summarizes the conclusions made in the previous chapters and gives suggestions for future research.

Chapter 2

Problem Formulation

In this chapter we set up the basic problem with which we will be dealing for the rest of this thesis. We describe both the mathematical physics of the problem, the basis of our model for this problem, and the measurement scheme used for estimation. We also formulate the problem in terms of Maximum-Likelihood estimation and derive a solution in these terms. We follow this by a discussion of our 2D piecewise constant model for the conductivity field, $\sigma(x, y)$(we will also refer to this sometimes as $\sigma(r)$ for short). This model forms the basic framework in which we can discuss estimating $\sigma(x, y)$ at different scales. Finally, we derive a useful mapping between two different boundary conditions along a square and include a section on notation and indexing conventions.

2.1 PDE and Measurements

Consider the unit square in which we wish to estimate a 2D conductivity function. Within this domain the equations governing the physics of this problem are Gauss' Law,

$$\nabla \cdot J = 0 \tag{2.1}$$

where J is the vector current density function in 2D and Ohm's Law,

$$J = \sigma(x, y)E \tag{2.2}$$

where $\sigma(x, y)$ is the unknown conductivity function and E is the electric field, which can be related to the potential function as $E = \nabla\phi(x, y)$. We adopt the experiment scheme used in [3] in which voltages are applied and currents are measured along the boundary. The potential on the boundary of the square is applied and is therefore a known quantity. Corresponding to each set of independent boundary excitations, a set of measurements is made of the currents normal to this boundary. We consider these measurements to be noisy. The fundamental problem is to estimate $\sigma(x, y)$ within the unit square by applying potentials along the boundary and taking measurements of the normal current along the boundary. In estimating $\sigma(x, y)$ we wish to use our knowledge of how the excitations relate to the measurements as manifested by the mathematical physics.

The mathematical physics for the problem can be concisely described by a partial differential equation(PDE) which must be satisfied within the unit square. The excitations for each experiment provide boundary conditions on the PDE. If we substitute eq.(2.2) into eq.(2.1) we obtain the following PDE:

$$\nabla \cdot \sigma(x, y)\nabla\phi_i(x, y) = 0 \tag{2.3}$$

for $0 \leq x \leq 1, 0 \leq y \leq 1$ with boundary conditions,

$$\phi_i(s) = B_i(s) \tag{2.4}$$

where $s\epsilon\Gamma$, Γ being the boundary of the unit square, and B_i is the applied potential function along Γ. The subscript i indexes the boundary conditions and potential

for a particular experiment. As it will be useful to do so later on we rewrite eq.(2.3) as

$$\nabla^2\phi_i(x,y) + \nabla \ln \sigma(x,y) \cdot \nabla\phi_i(x,y) = 0 \tag{2.5}$$

Our measurements can be concisely described by the following observation equation,

$$R_i(s) = \sigma(s)\frac{\partial\phi_i(x,y)}{\partial n}\bigg|_s + n_i \tag{2.6}$$

where R_i is the observation function, i indexes the particular experiment, and n_i is the additive noise function associated with the observation. Figure 2.1 illustrates the setup for the problem.

A few points are in order with regard to the structure of these equations. The relationship between B_i and $\frac{\partial\phi_i(x,y)}{\partial n}\Big|_s$ is implicitly defined through the solution of the PDE, eq.(2.3). Note that the PDE for this problem is linear with respect to $\phi(x,y)$ conditioned on knowing $\sigma(x,y)$; i.e. the forward problem is linear. However, in the inverse problem, where we must solve for the unknown $\sigma(x,y)$, the PDE is non-linear with respect to $\sigma(x,y)$. Since the potential, $\phi_i(x,y)$, is a function of $\sigma(x,y)$ for a particular set of boundary conditions, the observation equation, eq.(2.6), is also a non-linear function of $\sigma(x,y)$. Hence, the problem of estimating $\sigma(x,y)$ is inherently nonlinear. Finally, we should note that for each set of boundary conditions the potential within the unit square is unknown; in particular it is defined implicitly as the solution to the PDE. Therefore in estimating $\sigma(x,y)$ within the unit square we must also be concerned with estimating the potential therein. In terms of parameter estimation the unknown interior potential can be considered as a nuisance parameter; i.e. although the fundamental problem is to estimate $\sigma(x,y)$, the structure of the problem is such that we must estimate $\phi_i(x,y)$ as well.

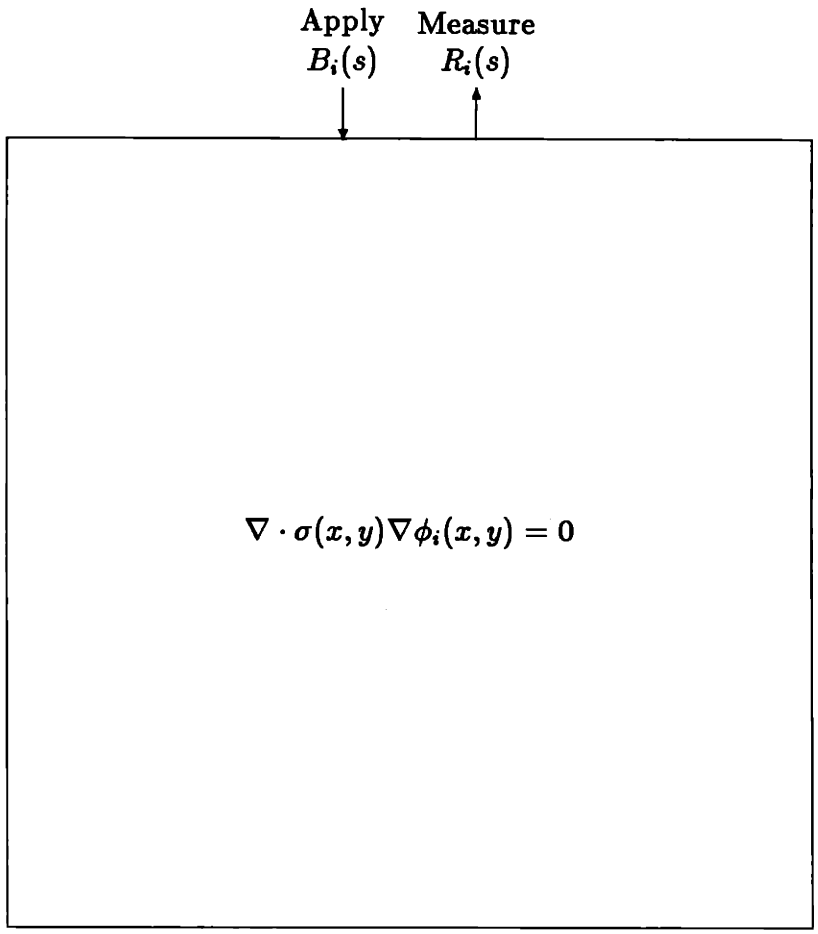

Figure 2.1: Illustration of a Particular Experiment

2.2 Maximum-Likelihood Estimation

In terms of parameter estimation we view the problem as being one of Maximum-Likelihood(ML) estimation. The optimal ML estimate is defined to be the value of the parameter which maximizes the likelihood function. The likelihood function is defined to be the conditional density, $p_{r|\sigma}(R|\sigma)$, where $r = \{r_1, ...r_M\}$, the set of M observations [13].

From our observation equation, eq.(2.6), where the set of observations consists of the observation functions R_i, we can derive the likelihood function and in particular the natural log of this function, the log-likelihood function, takes the following form.

$$\begin{aligned} \ln p_{r|\sigma}(R|\sigma) &= -\sum_i \int_\Gamma du \Psi_i(u) \int_\Gamma K_i^{-1}(u,v) dv \Psi_i(v) \\ &+ \alpha \end{aligned} \tag{2.7}$$

where

$$\Psi_i(s) = (R_i(s) - \sigma(s) \frac{\partial \phi_i(x,y)}{\partial n}|_s) \tag{2.8}$$

and $s \epsilon \Gamma$, Γ being the boundary of the unit square. The constant α is a normalization term which does not depend on σ. We have assumed the noise functions from experiment to experiment are independent, jointly Gaussian functions and the covariance function of each noise function is $K_i(u,v)$.

$$E\{n_i(u) n_j(v)\} = \delta_{ij} K_i(u,v) \tag{2.9}$$

For the case where the Gaussian measurement noise is correlated from experiment to experiment we would simply replace the summation in the log-likelihood function by a double summation, taking into account cross terms of ψ_i weighted by their correlations between experiments.

The maximum-likelihood solution results from maximizing the log-likelihood function, eq.(2.7), with respect to $\sigma(x,y)$. That is,

$$\hat{\sigma}_{ML} = \arg\max_{\sigma} \left\{ \ln p_{r|\sigma}(R|\sigma) \right\} \tag{2.10}$$

The ML estimation problem is fundamentally an optimization problem whereby one obtains an optimal solution by maximizing with respect to the parameter of interest. What makes the optimization non-trivial in our case is the fact that the normal derivative function, $\frac{\partial \phi_i(x,y)}{\partial n}|_s$, in the log-likelihood function is non-trivially related to the boundary conditions and the interior $\sigma(x,y)$. However, this relationship is implicitly defined through the solution of the PDE, eq.(2.3), for each set of Dirichlet boundary conditions, eq.(2.4). We wish to exploit this structure in developing an optimization scheme for finding $\hat{\sigma}_{ML}$.

2.3 Optimization Scheme

In this section we describe an optimization problem whose solution yields $\hat{\sigma}_{ML}$. The choice of cost function for the optimization is guided by the structure of the problem; it is chosen so as to combine the likelihood function with the PDE. In keeping with this structure we consider the two sets of parameters with respect to which we optimize to be: 1) $\sigma(x,y)$, the conductivity function which we wish to estimate 2) $\phi_i(x,y)$, the unknown interior potential function for each set of Dirichlet boundary conditions.

The likelihood function is the basis of the cost function. We choose to imbed the information contained in the PDE as an additional quadratic penalty term. This

term penalizes failure to satisfy the PDE. The resulting cost function is as follows:

$$\begin{aligned} f(\sigma,\phi_i) &= \sum_i \Big\{ \int_\Gamma du \Psi_i(u) \int_\Gamma K_i^{-1}(u,v) dv \Psi_i(v) \\ &+ \lambda \| \nabla \cdot \sigma(x,y) \nabla \phi_i(x,y) \|^2 \Big\} \end{aligned} \tag{2.11}$$

where

$$\Psi_i(s) = (R_i(s) - \sigma(s) \frac{\partial \phi_i(x,y)}{\partial n}|_s) \tag{2.12}$$

and $s \epsilon \Gamma$, Γ being the boundary of the unit square. The value of $\phi_i(x,y)$ on Γ is known for each experiment. The subscript i is used to index the ith experiment and the weight λ is a non-negative real number.

The first term is the negative log-likelihood function for the case where the noise functions from experiment to experiment are independent, jointly Gaussian functions and the covariance function of each noise function is $K_i(u,v)$.

$$E\{n_i(u) n_j(v)\} = \delta_{ij} K_i(u,v) \tag{2.13}$$

The second term of eq.(2.11) is a penalty term which is meant to represent the degree to which the parameters satisfy the PDE for each set of boundary conditions. The higher value of λ the higher the degree to which the PDE must be satisfied.

The values of σ, ϕ_i which minimize the function f represent the optimal solution. As we increase λ the optimal solution satisfies the PDE to a greater degree. As λ approaches infinity the optimal solution approaches the ML solution for σ.

2.4 Pixelating $\sigma(r)$

Our approach to estimating $\sigma(x,y)$ begins with a 2D piecewise constant model for $\sigma(x,y)$. The domain, the unit square, becomes a grid of pixels in which the value

of $\sigma(x,y)$ within each pixel is a constant. For example Figure 2.2 illustrates the 4 by 4 case. When we speak of multi-resolution estimation we are referring to various resolutions of $\sigma(x,y)$ pixelated in this 2D piecewise fashion.

The PDE for the case of this pixelated $\sigma(x,y)$ has an interesting interpretation. Referring to eq.(2.5) we see that for constant $\sigma(x,y)$ the PDE becomes Laplace's Equation independent of $\sigma(x,y)$.

$$\nabla^2\phi = 0 \tag{2.14}$$

This means that within each of the squares of our pixelated domain, where $\sigma(x,y)$ is a constant value, Laplace's Equation holds.

Along the edges of each square there exist discontinuities in $\sigma(x,y)$. By using the integral form of Gauss' Law we get constraints on boundary conditions along each interior edge. Namely, the normal current must be continuous across each interior edge. The normal current along an edge is simply equal to the derivative of $\phi_i(x,y)$ in the direction normal to that edge multiplied by the value of $\sigma(x,y)$ in the square to which the edge is associated. Note that for $\sigma(x,y)$ discontinuous across an edge $\nabla\phi_i(x,y)$ is also discontinuous across that edge, making what one calls the normal derivative of $\phi_i(x,y)$ along an edge dependent on the side of the edge to which one is referring. Figure 2.3 illustrates this for a particular vertical edge.

In summary, satisfying the PDE in the unit square for a pixelated $\sigma(x,y)$ amounts to, for a given set of Dirichlet boundary conditions along the unit square, satisfying Laplace's Equation in each square while also satisfying the condition that the normal current be continuous across each internal edge.

The pixelation of $\sigma(x,y)$ also lends itself to an interesting interpretation of the

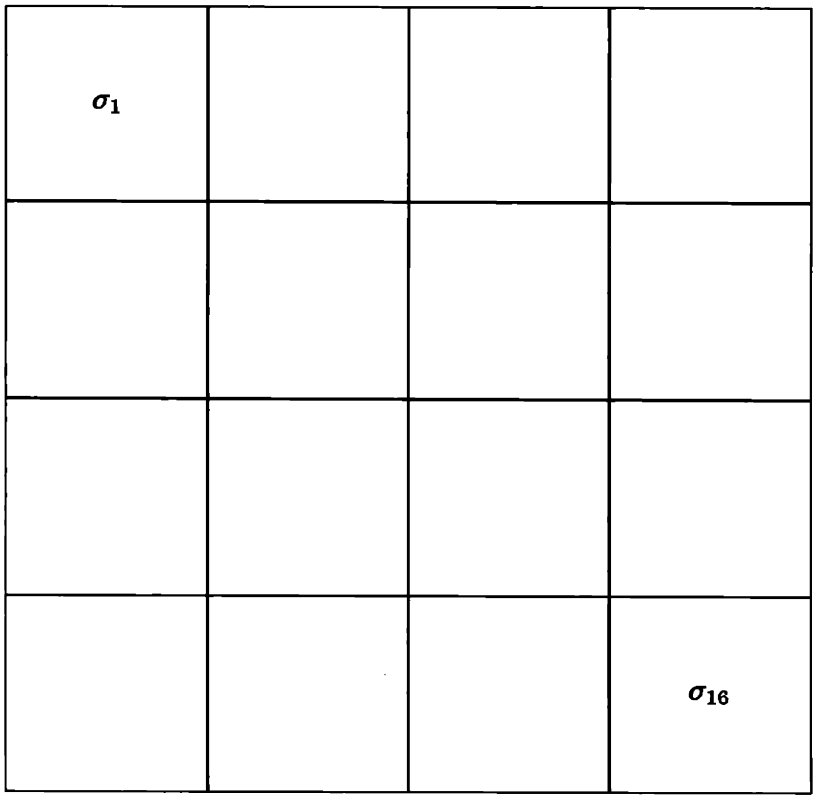

Figure 2.2: Pixelated $\sigma(r)$

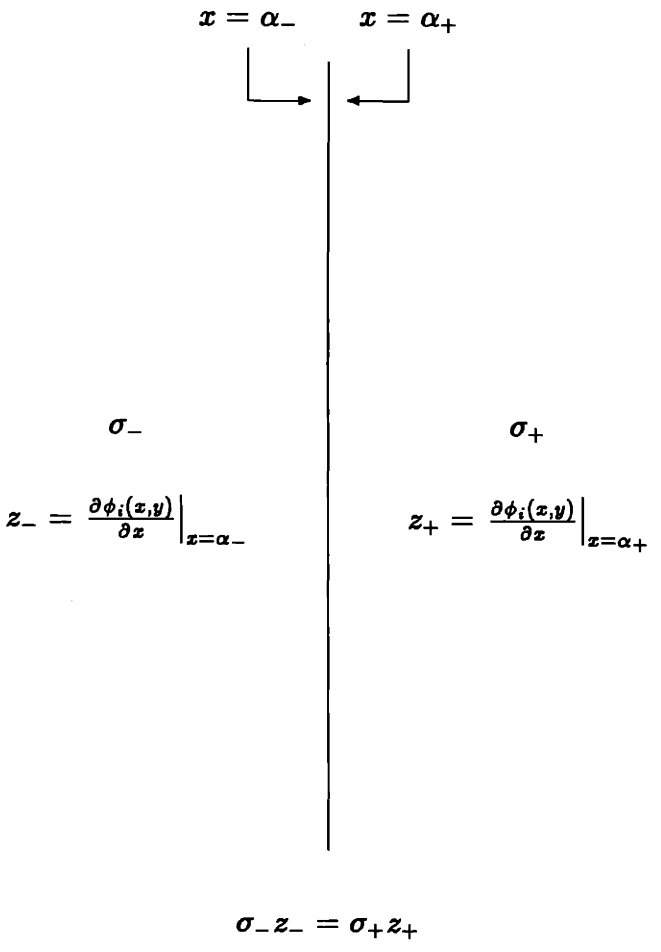

Figure 2.3: Continuity of Normal Current Across an Edge

overall estimation problem. We can think of the overall problem as being a collection of smaller problems defined on each square. Within each square we have the problem of estimating a constant value of $\sigma(x,y)$ given the potential and the normal current along the edges. For squares adjacent to the boundary of the unit square the potential along the edges adjacent to the boundary is known(it is in fact the Dirichlet boundary condition) while the normal current along these edges are actual measurements. Along the remaining edges in these squares as well as the edges in all squares non-adjacent to the boundary of the unit square, the potential is unknown while the normal current is dependent on the values of $\sigma(x,y)$ and $\frac{\partial \phi_i(x,y)}{\partial n}$ on the other sides of these edges.

The basic structure of the estimation problem consists of the following. Within each square we must estimate the value of $\sigma(x,y)$ in that square and as a nuisance parameter the potential along the edges non-adjacent to the overall boundary, Γ, of the large unit square. The potential along Γ is applied and therefore known. Our observations consist of the normal currents along the edges. In the cases where the edges are adjacent to the boundary the observations consist of actual measurements. In the remaining interior edges we actually have what we can think of as pseudo-measurements consisting of quantities associated with adjacent squares. This overall structure is suggestive of a highly distributed algorithm for performing the optimization necessary in computing the ML estimate of $\sigma(x,y)$.

Our next step is to write the cost function for the optimization problem in the case of our piecewise constant $\sigma(x,y)$. Before doing this we spend the next section developing notation and defining relevent quantities.

2.5 Notation and Definitions

In discussing various aspects of the problem of estimating a piece-wise constant $\sigma(r)$ it becomes necessary to define certain quantities along edges of the piece-wise constant squares as well as certain conventions pertaining to the numbering of these quantities. In this section we establish notation and numbering conventions that will be used in this thesis.

2.5.1 Numbering Squares

A square refers to a patch that exists as a result of subdividing the unit square so that $\sigma(r)$ is 2D piecewise constant. For example the four square case refers to the case where we discretize $\sigma(r)$ into four equal, piecewise constant regions in the unit square. Figure 2.4 illustrates the numbering convention to be used in referring to quantities associated with a particular square in the four square case. For example σ_1 refers to the value of $\sigma(r)$ defined in the region encompassed by square 1, σ_2 the value in square 2, etc. For cases where the number of squares is greater than four we will index the squares and quantities associated with them in lexicographic order(from top left to bottom right).

2.5.2 Definition of the Normal Derivative Function

We refer to the derivative of the potential along the edge of a square as the normal derivative function. Normal derivative functions are defined to be the dot product of the gradient of the potential, a 2D vector function, and the unit normal vector along the edge of a square. The unit normal vector points in the increasing x direction if the edge is vertical and in the increasing y direction if the edge is horizontal.

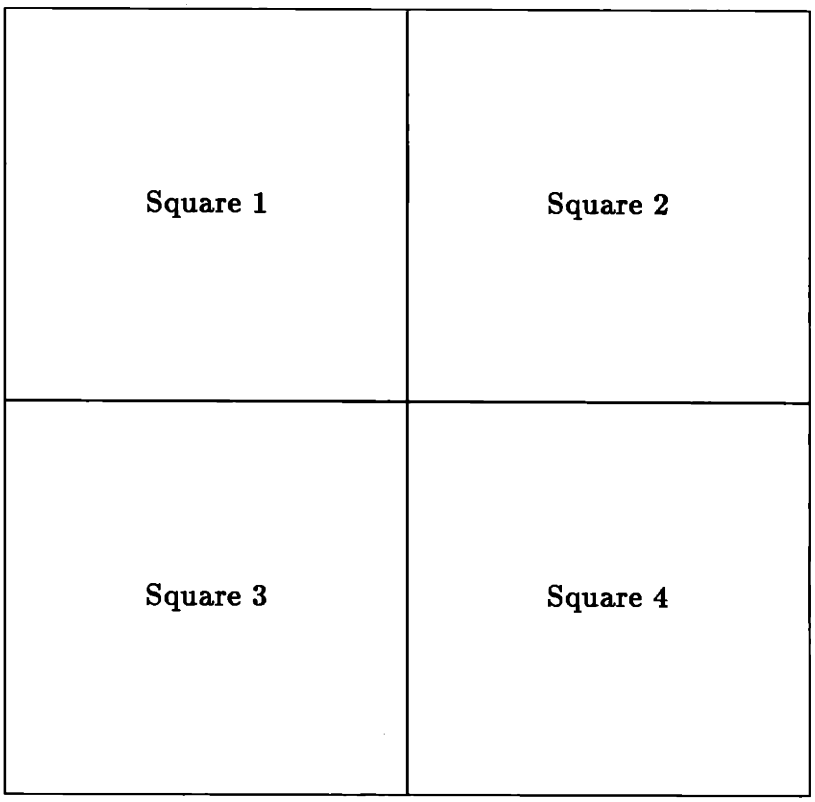

Figure 2.4: Four Square Numbering Convention

2.5.3 Defining Functions Along Edges of a Square

There are two functions of interest defined along the edges of each square. One is the normal derivative function which we call by the variable z. The other is the potential function for which we assign the variable b.

Normal derivative functions defined along a particular edge are indexed by two subscripts; the first subscript, n, indexes the square on which the function is defined, the second subscript, j, indexes the particular edge of the square on which the function is defined. The function $z_{n,j}$ for example refers to the normal derivative function on the jth edge of square n. Figure 2.5 illustrates the clockwise edge numbering convention used for a particular square.

Note with this double subscript convention it may seem odd that the interior edges are named twice depending on the square to which one associates the function. However, recall that for a piecewise constant $\sigma(r)$ the normal derivative function will in general have a different value on the two sides of each edge. This fact necessitates giving names to two functions, one defined to lie on one side of the edge and another defined to lie on the other side.

For convenience, we also use double subscripts for the potential functions defined along edges even though these are the same on either side of the boundary. For example, $b_{n,j}$ refers to the potential function on the jth edge of square n as illustrated in Figure 2.5.

2.5.4 Discretizing Functions

In our development we consider a discretized version of the problem in which the potential functions and normal derivative functions along edges become vectors

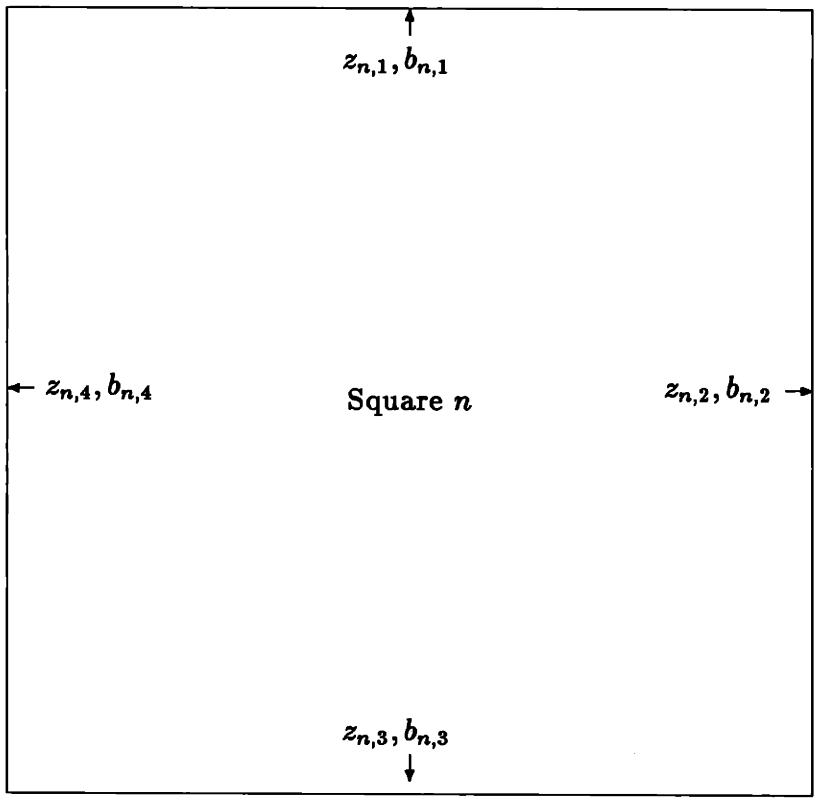

Figure 2.5: Normal Derivative and Potential Functions Defined on Edges of a Square

whose elements are equally spaced samples of the corresponding functions. The elements of these vectors are indexed such that the first element corresponds to the sample defined at the lowest value of x(or y) while the last element corresponds to the sample defined at the highest value of x(or y). For example for a vector of length N, the elements of the vector indexed 1 to N correspond to equally spaced samples of a function from left to right along a horizontal edge, from bottom to top along a vertical edge. Figure 2.6 illustrates this for potential vectors defined along the edges of a square.

We discretize our measurements along the boundary by splitting the observations into four vectors; one corresponding to the top of the unit square, one for the bottom, and one for the right and left sides as follows.

$$\begin{aligned} R1^l_m &= top \\ R2^l_m &= right \\ R3^l_m &= bottom \\ R4^l_m &= left \end{aligned} \tag{2.15}$$

where the superscript l indexes the experiment and the subscript m indexes the subportion of the vector that ranges from the $(m-1)r+1$th entry to the mrth entry; r is equal to the length of the vector divided by the number of σ pixels along an edge of the unit square(we assume this is an integer). These vectors represent samples of the corresponding measurement functions, sampled according to the previously described conventions.

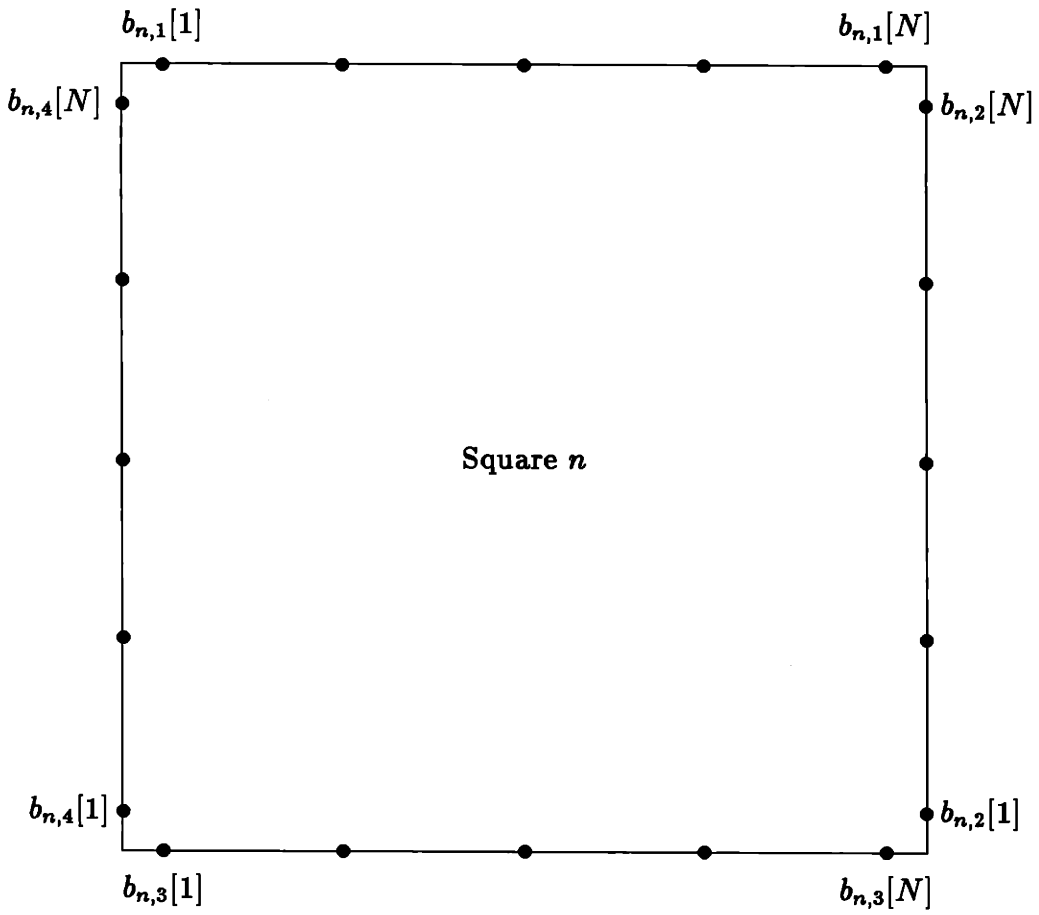

Figure 2.6: Discretized Functions on Edges of a Square

2.6 Cost Function for Piecewise Constant $\sigma(r)$

Armed with the notation and definitions of the previous section we are ready to write down the cost function, eq.(2.11), for the case of a piecewise constant $\sigma(r)$. In rewriting eq.(2.11) we simply acknowledge the interpretation the PDE has for our pixelated domain. Namely, satisfying the PDE is equivalent to satisfying the continuity of the current density across each internal edge. We arrive at the following[1].

$$\begin{aligned} f(\sigma,\phi_i) &= \sum_i \left\{ \int_\Gamma du \Psi_i(u) \int_\Gamma K_i^{-1}(u,v) dv \Psi_i(v) \right. \\ &+ \lambda \left\{ \sum_{t=0}^{N-1} \sum_{m=tN+1}^{N(t+1)-1} \|\sigma_m z^i_{m,2} - \sigma_{m+1} z^i_{m+1,4}\|^2 \right. \\ &+ \left. \left. \sum_{t=0}^{N-2} \sum_{m=tN+1}^{N(t+1)} \|\sigma_m z^i_{m,3} - \sigma_{m+N} z^i_{m+N,1}\|^2 \right\} \right\} \end{aligned} \tag{2.16}$$

where

$$\Psi_i(s) = (R_i(s) - \sigma(s) Z_i(s)) \tag{2.17}$$

and $s \epsilon \Gamma$, Γ being the boundary of the unit square. The superscript i on the terms $z^i_{m,j}$ for $j = 1,...4$ is used to indicate that these normal derivative functions are associated with the ith experiment.

The first term is the negative log-likelihood function as before. The other two terms, representing the penalty term, consist of the square of the L_2 norm of the difference between the normal currents along each interior edge of the unit square.

[1]Note that although σ is considered to be pixelated, all other quantities are still considered to be continuous and the norm $\|\cdot\|$ is the continuous L_2 norm.

2.7 Dirichlet to Neumann Map

In dealing with a piecewise constant $\sigma(r)$ we will often need to know the relationship between the potential vectors and the normal derivative vectors along the edges of a piece-wise constant patch or square. In this section we derive the explicit mapping between the Dirichlet boundary conditions and Neumann boundary conditions on a square whose sides are of some length α. We do this using the fact that Laplace's Equation holds within a square of constant $\sigma(r)$ and furthermore that it is an equation for which a closed form solution exists. By sampling the continuous solution and using the discrete sine transform we obtain a linear matrix relationship between samples of the potential and normal derivative functions along the edges of the square.

We can solve Laplace's equation in a square with edges of length α analytically using separation of variables giving a simple expression for the potential throughout the square as a function of the potential on the boundary of the square. The solution to Laplace's eq., ϕ_s, in such a square is given by [1],

$$\begin{aligned}\phi_s(x,y) = &\sum_{m=1}^{\infty} \hat{b}_1(m)\sin(\frac{m\pi x}{\alpha})\sinh(\frac{m\pi y}{\alpha}) \\ &+ \sum_{m=1}^{\infty} \hat{b}_2(m)\sinh(\frac{m\pi x}{\alpha})\sin(\frac{m\pi y}{\alpha}) \\ &+ \sum_{m=1}^{\infty} \hat{b}_3(m)\sin(\frac{m\pi x}{\alpha})\sinh(\frac{m\pi}{\alpha}(\alpha - y)) \\ &+ \sum_{m=1}^{\infty} \hat{b}_4(m)\sinh(\frac{m\pi}{\alpha}(\alpha - x))\sin(\frac{m\pi y}{\alpha})\end{aligned} \tag{2.18}$$

$$\hat{b}_t(m) = \frac{2}{\alpha\sinh(m\pi)}\int_0^{\alpha} b_t(s)\sin(\frac{m\pi s}{\alpha})ds \tag{2.19}$$

for $t = 1, 2, 3, 4$ where

$$\phi_s = \begin{cases} b_1(x) & 0 \leq x \leq \alpha \quad y = \alpha \\ b_2(x) & 0 \leq y \leq \alpha \quad x = \alpha \\ b_3(y) & 0 \leq x \leq \alpha \quad y = 0 \\ b_4(y) & 0 \leq y \leq \alpha \quad x = 0 \end{cases}$$

The functions b_t and $\hat{b}_t$ are defined to be the potential function and the weighted sine transform of the potential function respectively along the edges of the square.

We now take the normal derivative of (2.18) along each edge then sample the resulting functions uniformly along each edge to form the normal derivative vectors. That is,

$$\begin{aligned} z_1[i] &= \frac{\partial \phi_s(x,y)}{\partial y} \quad y = \alpha, \quad x = i\frac{\alpha}{N+1} \\ z_2[i] &= \frac{\partial \phi_s(x,y)}{\partial x} \quad y = i\frac{\alpha}{N+1}, \quad x = \alpha \\ z_3[i] &= \frac{\partial \phi_s(x,y)}{\partial y} \quad y = 0, \quad x = i\frac{\alpha}{N+1} \\ z_4[i] &= \frac{\partial \phi_s(x,y)}{\partial x} \quad y = i\frac{\alpha}{N+1}, \quad x = 0 \end{aligned} \tag{2.20}$$

for $i = 1, ...N$. Note z_t for $t = 1, ...4$ are $N \times 1$ vectors whose entries consist of the normal derivative functions sampled symmetrically along the edges of a square; i.e. the number of entries on one side of the midpoint of an edge equals the number of entries on the other side.

Note that the normal derivative vectors defined in eq.(2.20) depend on the functions $\hat{b}_t$. The functions $\hat{b}_t$ in turn depend on the potential functions along the boundary, b_t, through the continuous sine transform, eq.(2.19). We approximate eq.(2.19) using the discrete sine transform. We form $N \times 1$ potential vectors by sampling their corresponding potential functions along the edges in the same manner as described for the normal derivative vectors. We define the following N-point discrete sine transform.

$$\tilde{b}_t(l) = \frac{\alpha}{N} \sum_{k=1}^{N} b_t(k) \sin(\frac{kl\pi}{N}) \tag{2.21}$$

for $k,l = 1...N$.

If we substitute eq.(2.21) into eq.(2.20) and explicitly write each relationship, we arrive at the following.

$$\begin{aligned} z_1[i] &= \sum_{m=1}^{\infty} (\frac{m\pi}{\alpha}) \tilde{b}_1(m) \sin(\frac{m\pi x}{\alpha}) \cosh(m\pi) \\ z_2[i] &= \sum_{m=1}^{\infty} (\frac{m\pi}{\alpha}) \tilde{b}_2(m) \cosh(m\pi) \sin(\frac{m\pi y}{\alpha}) \\ z_3[i] &= \sum_{m=1}^{\infty} (-\frac{m\pi}{\alpha}) \tilde{b}_3(m) \sin(\frac{m\pi x}{\alpha}) \cosh(m\pi) \\ z_4[i] &= \sum_{m=1}^{\infty} (-\frac{m\pi}{\alpha}) \tilde{b}_4(m) \cosh(m\pi) \sin(\frac{m\pi y}{\alpha}) \end{aligned} \tag{2.22}$$

We can rewrite eq.(2.22) as a linear matrix relationship between the potential vectors and the normal derivative vectors of the square as follows.

$$\begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{pmatrix} = \begin{pmatrix} D & \hat{H}_o & -S_o & \hat{H}_{to} \\ \hat{H}_o & D & \hat{H}_{to} & -S_o \\ S_o & H_o & -D & H_{to} \\ H_o & S_o & H_{to} & -D \end{pmatrix}_{\alpha} \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{pmatrix}$$

where the $N \times 1$ vectors b_t and z_t are the boundary potential and boundary normal derivative vectors of the square. Note the subscript α is used to denote the length of the edges of the square being considered.

The following matrices are $N \times N$ where N represents the number of samples along an edge of the square.

$$S_{i,j} = \sin(\frac{ij\pi}{N+1}) \tag{2.23}$$

$$\begin{aligned} D &= \mathcal{D}S \\ \mathcal{D}(i,j) &= \frac{2j\pi \cosh(j\pi) \sin(\frac{ij\pi}{N+1})}{\alpha(N+1)\sinh(j\pi)} \end{aligned} \tag{2.24}$$

$$S_o = S_o S \tag{2.25}$$
$$S_o(i,j) = \frac{2j\pi \sin(\frac{ij\pi}{N+1})}{\alpha(N+1)\sinh(j\pi)}$$

$$H_o = \mathcal{H}_o S \tag{2.26}$$
$$\mathcal{H}_o(i,j) = \frac{2j\pi \sinh(\frac{ij\pi}{N+1})}{\alpha(N+1)\sinh(j\pi)}$$

$$H_{to} = \mathcal{H}_{to} S \tag{2.27}$$
$$\mathcal{H}_{to}(i,j) = \frac{2j\pi \sinh(j\pi(1-\frac{i}{N+1}))}{\alpha(N+1)\sinh(j\pi)}$$

$$\hat{H}_o = \hat{\mathcal{H}}_o S \tag{2.28}$$
$$\hat{\mathcal{H}}_o(i,j) = \frac{2j\pi \cos(j\pi)\sinh(\frac{ij\pi}{N+1})}{\alpha(N+1)\sinh(j\pi)}$$

$$\hat{H}_{to} = \hat{\mathcal{H}}_{to} S \tag{2.29}$$
$$\hat{\mathcal{H}}_{to}(i,j) = \frac{2j\pi \cos(j\pi)\sinh(j\pi(1-\frac{i}{N+1}))}{\alpha(N+1)\sinh(j\pi)}$$

for $i,j = 1,...N$.

Chapter 3

A Multi-Resolution Estimation Algorithm

3.1 Introduction

In this chapter we describe an algorithm with which we propose to compute the ML estimate of σ. The algorithm is hierarchical in nature, consisting of two main levels: 1) At the top level, where we incorporate the multi-resolution estimation idea, the algorithm moves from coarse scales of σ to successively finer ones. The algorithm begins by estimating a constant σ then moves to estimating σ at finer and finer scales. The change of scales is performed by using solutions at coarse scales to initiate iterations at finer scales. 2) At a particular scale of σ the algorithm determines the ML estimate of σ for that scale. The algorithm for computing the ML estimate at a particular scale is iterative and highly distributed. The algorithm alternates between finding the linear least squares estimate of σ conditioned on knowing $\phi(x,y)$ and finding the linear least squares estimate of $\phi(x,y)$ conditioned on knowing σ.

In addition to these two main levels there is also the issue of controlling the

value of the coefficient of the penalty term, λ. We allow for the value of λ to vary at two levels. We can increase λ with each change in scale of σ, forcing the constraints on σ to be satisfied more and more as we get to finer and finer resolutions of σ. We can also increase λ during the course of iterations at one scale. In this case we increase λ gradually in order to force the constraint of the PDE to be satsified more and more with each iteration. Figure 3.1 is a flow chart illustrating the main structure of the algorithm. Note that *G-S* stands for *Gauss-Seidel*, referring to the type of algorithm used.

3.2 Multi-resolution Scheme

The main idea of our a multi-resolution scheme is to compute the ML estimate of σ at a reasonably fine scale by computing the estimate for a sequence of scales starting with a very coarse scale then moving to successively finer scales. The hope is that by doing enough iterations at coarse scales we can minimize the number of iterations performed at fine scales where the computations are intensive. Also, the use of such a multi-resolution approach should prevent the desire to estimate fine detail from corrupting estimates of coarser scale features. We propose a scheme that solves a sequence of ML problems at successively finer scales. At each scale we initiate the iteration using the computed ML estimate for σ from the previous coarser scale.

The sequence of estimates at successively finer scales is computed using the following algorithm(refer to Figure 3.1).

1. Compute $\hat{\sigma}$ for constant σ throughout.

2. Subdivide each existing square into four equal squares. Initialize the value of

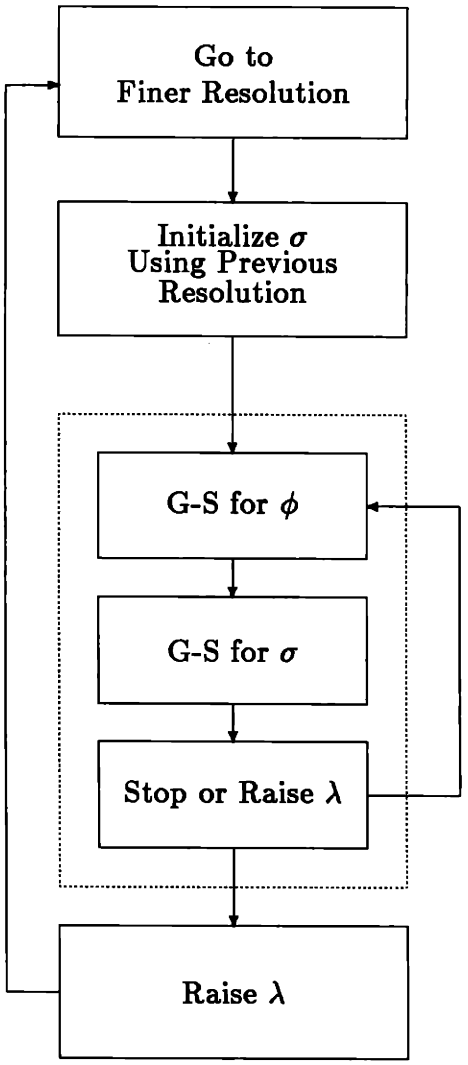

Figure 3.1: Structure of the Multi-resolution Algorithm

σ in each of these four squares with the value of $\hat{\sigma}$ computed for the larger square at the previous scale. Given these initial σ, initialize the edge potential vectors by optimizing with respect to ϕ.

3. Compute $\hat{\sigma}$, $\hat{\phi}$ using iterative and distributed(Gauss-Seidel-like) algorithms described in the next section.

4. If finer resolution is desired, repeat steps 2 and 3. Otherwise, stop.

The structure of the above algorithm suggests the following. At the finest resolution if the initial point chosen for σ at the start of the iteration is close to $\hat{\sigma}$, the total number of iterations at this resolution, where computations are relatively expensive, will be small. In addition by gradually moving towards the global minimum at the finest resolution by first estimating at coarse scales then moving to successively finer scales, we expect to stand a better chance of avoiding local minima than we would if we'd started our algorithm immediately at the finest resolution.

3.3 Estimation at One Scale Using Gauss-Seidel Relaxation Schemes

At each scale of σ the algorithm computes the ML solution for σ using the cost function, eq.(2.16). In developing an algorithm for computing the optimal solution for σ at a single scale we take advantage of the following rather key structure in the cost function. As we will show in this section, the problems of minimizing the cost function with respect to σ holding ϕ constant and minimizing with respect to ϕ holding σ constant are each *linear* problems. By taking advantage of this structure,

we develop an algorithm for minimizing the overall nonlinear function by solving a sequence of linear problems.

This method of optimization can be viewed as a type of coordinate descent as described in [7] where one attempts to minimize a cost function by making descents sequentially along different directions in the state space. From the standpoint of estimation our algorithm at a particular scale of σ consists of a sequence of estimators alternating between finding $\hat{\sigma}$ conditioned on knowing ϕ and finding $\hat{\phi}$ conditioned on knowing σ in which each of these estimators is a simple linear estimator.

The algorithms we develop for computing these linear estimators are both iterative and distributed. Jacobi and Gauss-Seidel relaxation schemes [4] are useful in that they are both iterative and highly conducive to distributed computation. The main idea of these algorithms from the standpoint of optimization is as follows. Suppose we have a parameter vector for which we would like the optimal solution. Relaxation schemes work on the principle that if we can optimize with respect to each entry of the vector regarding the other entries as known quantities, then by solving for each entry of the vector and repeating this for many iterations we hope to converge to the optimal solution. If optimization with respect to each entry lowers the overall cost function, then repeated iterations gradually lower the cost function until a minimum is attained. The global minimum is attained if the cost function is strictly convex.

The difference between Jacobi and Gauss-Seidel schemes is simply in the updating of the entries of the parameter vector. In Jacobi schemes the parameter vector isn't updated until each entry of the parameter vector is optimized once. This scheme is conducive to parallel computation since the entire parameter vector can

be updated simultaneously. In Gauss-Seidel schemes on the other hand a parameter entry is updated immediately after it has been optimized or computed.

It seems the Gauss-Seidel scheme is not conducive to a parallel implementation because it is inherently serial. However, many physical problems exhibit local dependencies; e.g. in our problem these localities are exhibited in our cost function. This suggests the organization of the parameter vector into subsets, each of which contains entries that do not depend on each other. This allows the entries of each subset to be updated simultaneously. When these subsets are large, of course, parallel computation is beneficial. The organization of the parameter vector into subsets of non-dependent entries is called a *coloring* scheme. In order to maximize the benefit of a parallel implementation for a specific problem one would naturally desire the fewest number of colors. A good example of this, in which the number of colors is just two, is in the solution of Poisson's equation using Gauss-Seidel with red-black coloring.

Because of the immediate updating of each entry of the parameter vector, the convergence of the Gauss-Seidel scheme is better than that of the Jacobi scheme. Because of this and the fact that our problem is conducive to a good coloring scheme, we choose to investigate Gauss-Seidel schemes in solving the ML problem at a single scale. Note that the convergence of both schemes can be accelerated using successive-overrelaxation(SOR).

We describe in the next two subsections the Gauss-Seidel relaxation schemes used to solve the problem of optimizing with respect to ϕ knowing σ and the problem of optimizing with respect to σ knowing ϕ. We will consider the scale to be such that σ is $N \times N$.

3.3.1 Optimizing with Respect to ϕ

Let us consider the problem of lowering the overall cost function, eq.(2.16), with respect to ϕ conditioned on knowing σ. Furthermore, let us consider lowering eq.(2.16) with respect to ϕ along a single edge assuming we know ϕ on every other edge. That is, we wish to lower the cost function with respect to $b^k_{n,j}$ for some measurement k, $1 \leq k \leq L$, some square n, $1 \leq n \leq N^2$ and some edge j, $1 \leq j \leq 4$ assuming σ_n to be known for $n = 1, ..., N^2$ and assuming $b^l_{m,i}$ to be known for every (m,i,l) not equal to (n,j,k). Note that for each edge of every square we have L different potential vectors, each of which corresponds to one of L different experiments. Let us rewrite the cost function, eq.(2.16), explicitly as a function of all the $b^l_{m,i}$ using our Dirichlet to Neumann map that relates $z^l_{m,i}$ to $b^l_{m,i}$. For convenience we rewrite the map using a different notation to give us the following relationship on square m for experiment l between $b^l_{m,i}$ and $z^l_{m,i}$ for some i.

$$\begin{pmatrix} z^l_{m,1} \\ \vdots \\ z^l_{m,4} \end{pmatrix} = \begin{pmatrix} T_{1,1} & \cdots & T_{1,4} \\ \vdots & & \vdots \\ T_{4,1} & \cdots & T_{4,4} \end{pmatrix} \begin{pmatrix} b^l_{m,1} \\ \vdots \\ b^l_{m,4} \end{pmatrix} \tag{3.1}$$

where $T_{o,p}$ for $1 \leq o \leq 4$, $1 \leq p \leq 4$ is an $M \times M$ matrix and M is the number of samples per edge. By substituting eq.(3.1) into eq.(2.16) we obtain,

$$\begin{aligned} f(\sigma, \phi_l) \;=\; & \sum_{l=1}^{L} \Big\{ \gamma \Psi_l \\ & + \lambda \Bigg\{ \sum_{t=0}^{N-1} \sum_{m=tN+1}^{N(t+1)-1} \| \sigma_m \sum_{i=1}^{4} T_{2,i} b^l_{m,i} - \sigma_{m+1} \sum_{i=1}^{4} T_{4,i} b^l_{m+1,i} \|^2 \\ & + \sum_{t=0}^{N-2} \sum_{m=tN+1}^{N(t+1)} \| \sigma_m \sum_{i=1}^{4} T_{3,i} b^l_{m,i} - \sigma_{m+N} \sum_{i=1}^{4} T_{1,i} b^l_{m+N,i} \|^2 \Bigg\} \Big\} \end{aligned} \tag{3.2}$$

where

$$\begin{aligned}\Psi_l &= \sum_{m=1}^{N} \|R1^l_m - \sigma_m \sum_{i=1}^{4} T_{1,i} b^l_{m,i}\|^2 \\ &+ \sum_{t=1}^{N} \|R2^l_{N+1-t} - \sigma_{N+(t-1)N} \sum_{i=1}^{4} T_{2,i} b^l_{N+(t-1)N,i}\|^2 \\ &+ \sum_{m=N(N-1)+1}^{N^2} \|R3^l_{m-N(N-1)} - \sigma_m \sum_{i=1}^{4} T_{3,i} b^l_{m,i}\|^2 \\ &+ \sum_{t=1}^{N} \|R4^l_{N+1-t} - \sigma_{1+(t-1)N} \sum_{i=1}^{4} T_{4,i} b^l_{1+(t-1)N,i}\|^2\end{aligned} \tag{3.3}$$

For simplicity we have assumed the noise to be independent and identically distributed from experiment to experiment. We have also assumed that the distribution of the measurement noise along the boundary for each experiment is white with variance γ^{-1}.

Note that eq.(3.2) is quadratic in $b^k_{n,j}$. To illustrate this consider the part of eq.(3.2) that depends on the particular vertical edge potential vector $b^k_{n,2}$.

$$\begin{aligned}f(b^k_{n,2}) &= \|(\sigma_n T_{2,2} - \sigma_{n+1} T_{4,4}) b^k_{n,2} - d_1\|^2 \\ &+ \sum_{\substack{i=1 \\ i\neq 2}}^{4} \|c_1 T_{i,2} b^k_{n,2} - d_2\|^2 \\ &+ \sum_{\substack{i=1 \\ i\neq 4}}^{4} \|c_2 T_{i,4} b^k_{n,2} - d_3\|^2\end{aligned} \tag{3.4}$$

where

$$\begin{aligned}c_1 &= \sigma_n \quad or \quad \tfrac{\gamma}{\lambda}\sigma_n \\ c_2 &= \sigma_{n+1} \quad or \quad \tfrac{\gamma}{\lambda}\sigma_{n+1}\end{aligned}$$

for $1 \leq n \leq N^2$, $n \bmod N = 0$. The vectors $d_1, ..., d_3$ do not depend on $b^k_{n,j}$.

In order to lower the cost function, eq.(3.2), with respect to the edge vector $b^k_{n,2}$ we can find the value of $b^k_{n,2}$ that minimizes eq.(3.4); i.e. by minimizing eq.(3.4)

with respect to $b^k_{n,2}$ while considering everything else to be known, we make a descent along the surface of the overall cost function. To do this we take the partial derivative of eq.(3.2) with respect to $b^k_{n,2}$ and set the result equal to zero. This results in a set of necessary conditions for the minimum of eq.(3.2) with respect to $b^k_{n,2}$. If we take the derivative of eq.(3.4) with respect to the vertical edge vector $b^k_{n,2}$ and set the result equal to zero, we obtain the following set of equations which we solve for $b^k_{n,2}$.

$$\left(\sigma_n^2 A1 + \sigma_{n+1}^2 A2 - \sigma_n \sigma_{n+1} A3\right) b^k_{n,2} = \sum_{l=1}^{9} v_l \tag{3.5}$$

where

$$A1 = \sum_{l=1}^{4} \alpha(l) T'_{l,2} T_{l,2} \tag{3.6}$$

$$A2 = \sum_{l=1}^{4} \beta(l) T'_{l,4} T_{l,4} \tag{3.7}$$

$$A3 = T'_{4,4} T_{2,2} + T'_{2,2} T_{4,4} \tag{3.8}$$

$$v1 = -\sigma_n^2 \sum_{l=1}^{4} \alpha(l) \left(T'_{l,2} \sum_{\substack{j=1 \\ j \neq 2}}^{4} T_{l,j} b^k_{n,j} \right) \tag{3.9}$$

$$v2 = -\sigma_{n+1}^2 \sum_{l=1}^{4} \beta(l) \left(T'_{l,4} \sum_{\substack{j=1 \\ j \neq 4}}^{4} T_{l,j} b^k_{n+1,j} \right) \tag{3.10}$$

$$v3 = \sigma_n \sigma_{n+1} \left(T'_{4,4} \sum_{\substack{j=1 \\ j \neq 2}}^{4} T_{2,j} b^k_{n,j} + T'_{2,2} \sum_{\substack{j=1 \\ j \neq 4}}^{4} T_{4,j} b^k_{n+1,j} \right) \tag{3.11}$$

$$\alpha(l) = \begin{cases} \frac{\gamma}{\lambda} & l = 1 \qquad n = 1, \ldots N \\ \frac{\gamma}{\lambda} & l = 3 \qquad N(N-1)+1 \leq n \leq N^2 \\ \frac{\gamma}{\lambda} & l = 4 \qquad n \bmod N = 1 \\ 1 & \textit{otherwise} \end{cases}$$

$$\beta(l) = \begin{cases} \frac{\gamma}{\lambda} & l = 1 \qquad n = 1, \ldots N \\ \frac{\gamma}{\lambda} & l = 2 \qquad (n+1) \bmod N = 0 \\ \frac{\gamma}{\lambda} & l = 3 \qquad N(N-1)+1 \leq n \leq N^2 \\ 1 & otherwise \end{cases}$$

and

$$v4 = \begin{cases} (\frac{\gamma}{\lambda})\sigma_n T'_{4,2} R4^k_{N-((n-1)/N)} & n \bmod N = 1 \\ \sigma_n \sigma_{n-1} \left(T'_{4,2} \sum_{j=1}^{4} T_{2,j} b^k_{n-1,j} \right) & otherwise \end{cases}$$

$$v5 = \begin{cases} (\frac{\gamma}{\lambda})\sigma_{n+1} T'_{2,4} R2^k_{N-((n+1)/N)+1} & (n+1) \bmod N = 0 \\ \sigma_{n+1} \sigma_{n+2} \left(T'_{2,4} \sum_{j=1}^{4} T_{4,j} b^k_{n+2,j} \right) & otherwise \end{cases}$$

$$v6 = \begin{cases} (\frac{\gamma}{\lambda})\sigma_n T'_{1,2} R1^k_n & n = 1, \ldots N \\ \sigma_n \sigma_{n-N} \left(T'_{1,2} \sum_{j=1}^{4} T_{3,j} b^k_{n-N,j} \right) & otherwise \end{cases}$$

$$v7 = \begin{cases} (\frac{\gamma}{\lambda})\sigma_n T'_{3,2} R3^k_{n-N(N-1)} & N(N-1)+1 \leq n \leq N^2 \\ \sigma_n \sigma_{n+N} \left(T'_{3,2} \sum_{j=1}^{4} T_{1,j} b^k_{n+N,j} \right) & otherwise \end{cases}$$

$$v8 = \begin{cases} (\frac{\gamma}{\lambda})\sigma_{n+1} T'_{1,4} R1^k_{n+1} & n+1 = 1, \ldots N \\ \sigma_{n+1} \sigma_{n-N+1} \left(T'_{1,4} \sum_{j=1}^{4} T_{3,j} b^k_{n-N+1,j} \right) & otherwise \end{cases}$$

$$v9 = \begin{cases} (\frac{\gamma}{\lambda})\sigma_{n+1} T'_{3,4} R3^k_{n+1-N(N-1)} & N(N-1)+1 \leq n+1 \leq N^2 \\ \sigma_{n+1} \sigma_{n+N+1} \left(T'_{3,4} \sum_{j=1}^{4} T_{1,j} b^k_{n+N+1,j} \right) & otherwise \end{cases}$$

We can use an analogous argument to derive equations for lowering the overall cost function with respect to an horizontal edge potential vector. Specifically, if we take the derivative of the cost function with respect to the horizontal edge vector $b^k_{n,3}$ and set the result equal to zero, we get the following set of equations which we

solve for $b^k_{n,3}$.

$$\left(\sigma_n^2 B1 + \sigma_{n+N}^2 B2 - \sigma_n \sigma_{n+N} B3\right) b^k_{n,3} = \sum_{l=1}^{9} w_l \tag{3.12}$$

where

$$B1 = \sum_{l=1}^{4} \alpha(l) T'_{l,3} T_{l,3} \tag{3.13}$$

$$B2 = \sum_{l=1}^{4} \beta(l) T'_{l,1} T_{l,1} \tag{3.14}$$

$$B3 = T'_{1,1} T_{3,3} + T'_{3,3} T_{1,1} \tag{3.15}$$

$$w1 = -\sigma_n^2 \sum_{l=1}^{4} \alpha(l) \left(T'_{l,3} \sum_{\substack{j=1 \\ j \neq 3}}^{4} T_{l,j} b^k_{n,j} \right) \tag{3.16}$$

$$w2 = -\sigma_{n+N}^2 \sum_{l=1}^{4} \beta(l) \left(T'_{l,1} \sum_{\substack{j=1 \\ j \neq 1}}^{4} T_{l,j} b^k_{n+N,j} \right) \tag{3.17}$$

$$w3 = \sigma_n \sigma_{n+N} \left(T'_{1,1} \sum_{\substack{j=1 \\ j \neq 3}}^{4} T_{3,j} b^k_{n,j} + T'_{3,3} \sum_{\substack{j=1 \\ j \neq 1}}^{4} T_{1,j} b^k_{n+N,j} \right) \tag{3.18}$$

$$\alpha(l) = \begin{cases} \frac{\gamma}{\lambda} & l = 1 \quad n = 1, \ldots N \\ \frac{\gamma}{\lambda} & l = 2 \quad n \bmod N = 0 \\ \frac{\gamma}{\lambda} & l = 4 \quad n \bmod N = 1 \\ 1 & otherwise \end{cases}$$

$$\beta(l) = \begin{cases} \frac{\gamma}{\lambda} & l = 2 \quad n \bmod N = 0 \\ \frac{\gamma}{\lambda} & l = 3 \quad N(N-1) + 1 - N \leq n \leq N^2 - N \\ \frac{\gamma}{\lambda} & l = 4 \quad n \bmod N = 1 \\ 1 & otherwise \end{cases}$$

and

$$w4 = \begin{cases} (\frac{\gamma}{\lambda}) \sigma_n T'_{1,3} R1^k_n & n = 1, \ldots N \\ \sigma_n \sigma_{n-N} \left(T'_{1,3} \sum_{j=1}^{4} T_{3,j} b^k_{n-N,j} \right) & otherwise \end{cases}$$

$$w5 = \begin{cases} (\frac{\gamma}{\lambda})\sigma_{n+N}T'_{3,1}R3^k_{n+N-N(N-1)} & N(N-1)+1 \leq n+N \leq N^2 \\ \sigma_{n+N}\sigma_{n+2N}\left(T'_{3,1}\sum_{j=1}^{4}T_{1,j}b^k_{n+2N,j}\right) & otherwise \end{cases}$$

$$w6 = \begin{cases} (\frac{\gamma}{\lambda})\sigma_n T'_{4,3}R4^k_{N-((n-1)/N)} & n \bmod N = 1 \\ \sigma_n\sigma_{n-1}\left(T'_{4,3}\sum_{j=1}^{4}T_{2,j}b^k_{n-1,j}\right) & otherwise \end{cases}$$

$$w7 = \begin{cases} (\frac{\gamma}{\lambda})\sigma_n T'_{2,3}R2^k_{N-(n/N)+1} & n \bmod N = 0 \\ \sigma_n\sigma_{n+1}\left(T'_{2,3}\sum_{j=1}^{4}T_{4,j}b^k_{n+1,j}\right) & otherwise \end{cases}$$

$$w8 = \begin{cases} (\frac{\gamma}{\lambda})\sigma_{n+N}T'_{4,1}R4^k_{N-((n-1)/N)-1} & n \bmod N = 1 \\ \sigma_{n+N}\sigma_{n-1+N}\left(T'_{4,1}\sum_{j=1}^{4}T_{2,j}b^k_{n-1+N,j}\right) & otherwise \end{cases}$$

$$w9 = \begin{cases} (\frac{\gamma}{\lambda})\sigma_{n+N}T'_{2,1}R2^k_{N-((n+N)/N)+1} & n \bmod N = 0 \\ \sigma_{n+N}\sigma_{n+1+N}\left(T'_{2,1}\sum_{j=1}^{4}T_{4,j}b^k_{n+1+N,j}\right) & otherwise \end{cases}$$

The solutions of eq.(3.5) and eq.(3.12), if they exist, determine vertical and horizontal edge potential vectors, respectively, that minimize the overall cost function conditioned on knowing ϕ on every other edge and σ in each square. Note that solving eq.(3.5) and eq.(3.5) require solving a set of systems of M equations in M unknowns where M is the number of samples per edge.

We now examine the conditions under which eq.(3.5) and eq.(3.12) have solutions and the conditions under which these solutions are unique. Let us first consider eq.(3.5). Let

$$\mathcal{A} = \sigma_n^2 A1 + \sigma_{n+1}^2 A2 - \sigma_n\sigma_{n+1}A3 \tag{3.19}$$

Note that from the structure of our Dirichlet to Neumann map as derived in Chap-

ter 2 we know that $T_{2,2} = -T_{4,4}$. This allows us to write eq.(3.19) as

$$\begin{aligned} \mathcal{A} &= \sigma_n^2 \sum_{\substack{l=1 \\ l \neq 2}}^{4} \alpha(l) T'_{l,2} T_{l,2} \\ &+ \sigma_{n+1}^2 \sum_{l=1}^{3} \beta(l) T'_{l,4} T_{l,4} \\ &+ Q \end{aligned} \tag{3.20}$$

where

$$Q = ((\sigma_n + \sigma_{n+1}) T_{2,2})' ((\sigma_n + \sigma_{n+1}) T_{2,2}) \tag{3.21}$$

We now examine the question of whether there exists a solution to eq.(3.5) and if so, whether the solution is unique. Note eq.(3.20) shows $\mathcal{A}$ to be the sum of matrices multiplied by their transposes. Equivalently, $\mathcal{A}$ is a sum of positive semi-definite matrices. If any one of these positive semi-definite matrices is strictly positive definite, then $\mathcal{A}$ is strictly positive definite. This would imply that the matrix $\mathcal{A}$ is non-singular and eq.(3.5) has a unique solution. But, as we show in Appendix B, the matrix $T_{2,2}$ is non-singular. This implies the matrix Q is non-singular and hence eq.(3.5) can always be solved uniquely.

A similar argument can be made with regard to eq.(3.12). The proof in this case requires that the matrix $T_{1,1}$ be non-singular. But this is true since $T_{1,1} = T_{2,2}$. Therefore, eq.(3.12) can always be solved uniquely.

Suppose we wished to minimize the overall cost function with respect to ϕ along every edge conditioned on knowing σ. We use a Gauss-Seidel type of algorithm to solve this problem which we call GS-ϕ. The idea of the GS-ϕ algorithm is to minimize eq.(3.2) with respect to the potential along a particular edge holding the potential along the remaining edges constant using eq.(3.5) for vertical edges and eq.(3.12) for horizontal edges. By doing this for each successive edge and each

measurement set, and cycling through the entire set of edges and measurement sets for many iterations we hope to converge to a solution set of edge potentials that minimizes the overall cost function given a particular σ. Note the fact that solving either eq.(3.5) or eq.(3.12) results in a unique solution guarantees that these solutions result in a descent of the overall cost function. But the algorithm GS-ϕ simply consists of a sequence of solutions of either eq.(3.5) or eq.(3.12). Therefore, GS-ϕ converges to a minimum of the overall cost function conditioned on knowing σ.

One possible version of GS-ϕ can be summarized as follows:

1. Solve the normal equations, eq.(3.5), for each of the vertical edges, $b^k_{n,2}$.
2. Solve the normal equations, eq.(3.5), for each of the horizontal edges, $b^k_{n,3}$.
3. If adequate convergence has occurred then stop. Otherwise, repeat steps 1 and 2.

Note the ordering of the edges in the above algorithm is arbitrary. The convergence rate, however, inevitably depends on the ordering. This issue deserves further investigation, as it would be useful to be able to order the relaxation so as to give the best convergence rate.

3.3.2 Optimizing with Respect to σ

We now discuss the problem of lowering the cost function with respect to σ conditioned on knowing ϕ on every edge. Analogously to the case of optimizing with respect to ϕ we begin by discussing the problem of lowering the overall cost function with respect to σ in a particular square conditioned on knowing ϕ on every edge and σ in every other square.

We begin by considering the cost function, eq.(3.2). We rewrite eq.(3.2) so that it is a function of σ conditioned on knowing ϕ on each edge. Since the $b^l_{m,i}$'s are known and there is a mapping from $b^l_{m,i}$ to $z^l_{m,i}$ we can rewrite eq.(3.2) as follows:

$$\begin{aligned} f(\sigma,\phi_l) &= \sum_{l=1}^{L}\Big\{\ \gamma\Psi_l \\ &+ \lambda\Bigg\{\sum_{t=0}^{N-1}\sum_{m=tN+1}^{N(t+1)-1}\|\sigma_m z^l_{m,2}-\sigma_{m+1}z^l_{m+1,4}\|^2 \\ &+ \sum_{t=0}^{N-2}\sum_{m=tN+1}^{N(t+1)}\|\sigma_m z^l_{m,3}-\sigma_{m+N}z^l_{m+N,1}\|^2\Bigg\}\Big\} \end{aligned} \tag{3.22}$$

where

$$\begin{aligned} \Psi_l &= \sum_{m=1}^{N}\|R1^l_m-\sigma_m z^k_{m,1}\|^2 \\ &+ \sum_{t=1}^{N}\|R2^l_{N+1-t}-\sigma_{N+(t-1)N}z^l_{N+(t-1)N,2}\|^2 \\ &+ \sum_{m=N(N-1)+1}^{N^2}\|R3^l_{m-N(N-1)}-\sigma_m z^l_{m,3}\|^2 \\ &+ \sum_{t=1}^{N}\|R4^l_{N+1-t}-\sigma_{1+(t-1)N}z^l_{1+(t-1)N,4}\|^2 \end{aligned} \tag{3.23}$$

We now consider lowering the overall cost function with respect to σ in a particular square. By taking the dervivative of eq.(3.2) with respect to σ_n, the value of σ in the *nth* square, and setting the result equal to zero we obtain an equation for finding the value of σ_n that minimizes eq.(3.2). Since σ_n is a scalar, computations are much simpler than in the case of optimizing with respect to ϕ; i.e. we needn't solve a system of linear equations.

$$\sigma_n = \left(\sum_{k=1}^{K}\sum_{i=1}^{4}c_n(z^k_{n,i})'z^k_{n,i}\right)^{-1}\sum_{k=1}^{K}\sum_{i=1}^{4}(\theta^k_n)'z^k_{n,i} \tag{3.24}$$

where

$$\theta^k_1 = \begin{cases} (\frac{\gamma}{\lambda})R1^k_n & 1\le n\le N \\ \sigma_{n-N}z^k_{n-N,3} & otherwise \end{cases}$$

$$\theta_2^k = \begin{cases} (\frac{\gamma}{\lambda})R2^k_{N-(n/N)+1} & (n \bmod N) = 0 \\ \sigma_{n+1} z^k_{n+1,4} & otherwise \end{cases}$$

$$\theta_3^k = \begin{cases} (\frac{\gamma}{\lambda})R3^k_{n-N(N-1)} & N(N-1)+1 \leq n \leq N^2 \\ \sigma_{n+N} z^k_{n+N,1} & otherwise \end{cases}$$

$$\theta_4^k = \begin{cases} (\frac{\gamma}{\lambda})R4^k_{N-((n-1)/N)} & (n \bmod N) = 1 \\ \sigma_{n-1} z^k_{n-1,2} & otherwise \end{cases}$$

and

$$c_1 = \begin{cases} \frac{\gamma}{\lambda} & 1 \leq n \leq N \\ 1 & otherwise \end{cases}$$

$$c_2 = \begin{cases} \frac{\gamma}{\lambda} & (n \bmod N) = 0 \\ 1 & otherwise \end{cases}$$

$$c_3 = \begin{cases} \frac{\gamma}{\lambda} & N(N-1)+1 \leq n \leq N^2 \\ 1 & otherwise \end{cases}$$

$$c_4 = \begin{cases} \frac{\gamma}{\lambda} & (n \bmod N) = 1 \\ 1 & otherwise \end{cases}$$

We now examine the questions of existence and uniquess of solution with regard to eq.(3.24). A necessary and sufficient condition for the existence of a unique solution to eq.(3.24) is that the inverse in eq.(3.24) exists. Note that the inverse exists if and only if $z^k_{n,i}$ is not equal to the zero vector for all $k = 1, ..., K$, and all $i = 1, ...4$. But the condition that the vectors $z^k_{n,i}$ are all zero corresponds to the case of zero excitations, which is a degenerate case. Therefore, for any non-zero set of excitations eq.(3.24) has a unique solution.

We now present an algorithm for minimizing the overall cost function with respect to σ in every square conditioned on knowing ϕ on every edge. We call this the GS-σ algorithm. One possible version of GS-σ can be summarized as follows:

1. Solve eq.(3.24) for σ_n $n = 1, ...N^2$.

2. If adequate convergence has occurred then stop. Otherwise, repeat step 1.

As in the case for GS-ϕ the order of performing a sweep through the entire grid of σ_n affects the rate of convergence.

We now show that this algorithm converges to the global minimum of the cost function, eq.(3.22), with respect to σ conditioned on knowing ϕ on every edge for each experiment. The fact that the solution to eq.(3.24) is unique for non-zero excitations guarantees that it lowers the cost function. This implies that by successively solving eq.(3.22) for each σ_n we must converge to a minimum. But the cost function, eq.(3.22), is strictly convex(see Appendix A) implying it has a unique global minimum. Therefore, the algorithm GS-σ converges to the global minimum of the cost function, eq.(3.22), with respect to σ.

3.3.3 Overall Algorithm at One Scale

We have shown that solving the optimization problem at one scale can be broken into two subproblems. One subproblem is to optimize with respect to σ holding ϕ constant and the other is to optimize with respect to ϕ holding σ constant. Each of these subproblems is an optimization problem with respect to a quadratic cost function which an be interpreted as a linear least squares estimation problem. Our overall algorithm at one scale consists of alternating between optimizing with respect to ϕ and optimizing with respect to σ.

ALGORITHM 1

1. Initialize σ_n, $b^k_{n,j}$, for $k = 1, ..., K$, $n = 1, ..., N^2$, $i = 1, ...4$.

2. Solve eq.(3.24) for σ_n for $n = 1, ..., N^2$ until convergence using GS-σ.

3. Solve eq.(3.5) and eq.(3.5) for $b^k_{n,j}$ for $k = 1,...,K$, $n = 1,...,N^2$, $i = 1,...4$ until convergence using GS-ϕ.

4. If all σ_n and $b^k_{n,j}$ have reached adequate convergence then stop. Otherwise, repeat steps 2 and 3.

ALGORITHM 2

1. Initialize σ_n, $b^k_{n,j}$, for $k = 1,...,K$, $n = 1,...,N^2$, $i = 1,...4$.

2. Solve eq.(3.24) for σ_n for $n = 1,...,N^2$, i.e. one sweep through the set of N^2 squares.

3. Solve eq.(3.5) and eq.(3.5) for $b^k_{n,j}$ for $k = 1,...,K$, $n = 1,...,N^2$, $i = 1,...4$, i.e. one sweep through the set of $2N(N-1)$ edges.

4. If all σ_n and $b^k_{n,j}$ have reached adequate convergence then stop. Otherwise, repeat steps 2 and 3.

3.4 Appendix A: Proof of Strict Convexity

In order to prove that the overall cost function is strictly convex with respect to σ conditioned on knowing ϕ we prove that the cost function, eq.(3.22), for a single experiment is strictly convex. The strict convexity of eq.(3.22) over all the experiments then follows from the fact that it is a sum of strictly convex functions.

Let us begin by reinterpreting the cost function, eq.(3.22), for a single experiment. We can think of optimizing over eq.(3.22) as solving a weighted least-squares problem. The weighted least-squares problem amounts to minimizing the weighted

quadratic form of the error vector ϵ with respect to our parameter of interest. Specifically,

$$\hat{\sigma} = \arg\min_{\sigma} \epsilon' Q \epsilon \tag{3.25}$$

where the matrix Q is symmetric positive definite.

For our problem we can divide ϵ into two main parts. One part is what is known in the system identification literature as the *prediction error*, which we will denote as ϵ_ν. For our problem this consists of the difference between what we predict to be the normal current on the external boundary, Γ, and what we actually measure. The other component of ϵ consists of the error due to failing to satisfy the constraints of our PDE. We denote this component as ϵ_{PDE}.

$$\epsilon = \begin{pmatrix} \epsilon_\nu \\ \cdots \\ \epsilon_{PDE} \end{pmatrix} \tag{3.26}$$

Since ϵ_ν and ϵ_{PDE} represent two different types of error, we weight them differently. We weight ϵ_ν by the inverse covariance matrix of the measurement noise while we weight ϵ_{PDE} by the weighting term λ. This is manifested by the following partition of the overall weighting matrix Q.

$$Q = \begin{pmatrix} K^{-1} & 0 \\ 0 & \lambda I \end{pmatrix} \tag{3.27}$$

where the dimension of the square matrix K^{-1} equals the length of ϵ_ν and the dimension of I equals the length of ϵ_{PDE}.

Let us define ϵ_ν. This vector consists of the difference between the normal derivative vector along each exterior edge scaled by the value of σ at that edge and the vector of measurements along that edge. We can express this as follows.

$$\epsilon_\nu = \theta_{ext}\sigma - b_{ext} \tag{3.28}$$

where

$$\theta_{ext} = \begin{pmatrix} z_{1,1} & \cdots & 0 & & & & & & \\ \vdots & \ddots & \vdots & & & & & & \\ 0 & \cdots & z_{N,1} & & & & & & \\ 0 & & & & & & & \cdots & z_{N^2,2} \\ 0 & & & & & \cdots & z_{N^2-N,2} & \cdots & 0 \\ 0 & \cdots & z_{N,2} & \cdots & & & & & 0 \\ & & & & & & z_{N(N-1)+1,3} & \cdots & 0 \\ & & & & & & \vdots & \ddots & \vdots \\ 0 & & & & & & 0 & \cdots & z_{N^2,3} \\ 0 & & & & & \cdots & z_{N(N-1)+1,4} & \cdots & 0 \\ 0 & & \cdots & z_{N(N-1)+1-N,4} & \cdots & & & & 0 \\ z_{1,4} & \cdots & & & & & & & 0 \end{pmatrix} \tag{3.29}$$

and

$$b_{ext} = \begin{pmatrix} R_1 \\ \vdots \\ R_4 \end{pmatrix} \tag{3.30}$$

We now derive an expression for ϵ_{PDE}. Let us begin by considering the constraints due to the PDE. Essentially, the constraints are that the normal current across each internal edge be continuous. This translates to the condition that the normal derivative vector on one side of an edge scaled by the value of σ on that side must equal the normal derivative vector on the other side of that edge scaled by the value of σ on that other side. This is illustrated in Figure 2.3. If we take the difference between the predicted currents on the two sides of each interior edge as the error in satisfying the PDE, we get the following.

$$\epsilon_{PDE} = \theta_{int}\sigma \tag{3.31}$$

The matrix θ_{int} consists of two submatrices one of which represents the constraints

on the vertical edges the other on the horizontal edges. Specifically,

$$\theta_{int} = \begin{pmatrix} \theta^v_{int} \\ \cdots \\ \theta^h_{int} \end{pmatrix} \tag{3.32}$$

The matrix representing constraints on vertical interior edges is defined as follows.

$$\theta^v_{int} = \begin{pmatrix} B_1 & 0 & \cdots & 0 \\ 0 & B_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & B_N \end{pmatrix} \tag{3.33}$$

where

$$B_i = \begin{pmatrix} z_{j,2} & -z_{j+1,4} & 0 & \cdots & 0 \\ 0 & z_{j+1,2} & -z_{j+2,4} & \cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & z_{j+N-2,2} & -z_{j+N-1,4} \end{pmatrix} \tag{3.34}$$

and $j = (i-1)N + 1$ for $i = 1, ...N$. The matrix representing constraints on horizontal interior edges is defined as follows.

$$\theta^h_{int} = \begin{pmatrix} z_{1,3} & 0 & \cdots & 0 & -z_{1+N,1} & 0 & \cdots & 0 \\ 0 & z_{2,3} & 0 & \cdots & 0 & -z_{2+N,1} & \cdots & 0 \\ \vdots & & \ddots & & & & \ddots & \vdots \\ 0 & & & z_{N(N-1),3} & 0 & \cdots & 0 & -z_{N^2,1} \end{pmatrix} \tag{3.35}$$

Let us arrange our expressions for the error vector as follows.

$$\epsilon = \theta\sigma - b \tag{3.36}$$

where

$$\theta = \begin{pmatrix} \theta_{ext} \\ \cdots \\ \theta_{int} \end{pmatrix} \tag{3.37}$$

and

$$b = \begin{pmatrix} b_{ext} \\ \cdots \\ 0 \end{pmatrix} \tag{3.38}$$

We now substitute eq.(3.36) into our weighted least squares equation, eq.(3.25), to get the following.

$$\hat{\sigma} = \arg\min_{\sigma} \{\sigma'\theta'Q\theta\sigma - \sigma'\theta'Qb + b'Qb\} \tag{3.39}$$

Since $b'Qb$ does not depend on σ we can rewrite eq.(3.39) as

$$\hat{\sigma} = \arg\min_{\sigma} f(\sigma) \tag{3.40}$$

where

$$f(\sigma) = \{\sigma'\theta'Q\theta\sigma - \sigma'\theta'Qb\} \tag{3.41}$$

Note that eq. (3.41) is precisely our discrete cost function, eq.(3.22) for one experiment.

Note that θ has full column rank for non-zero z_{ij}'s and the matrix Q is symmetric positive definite. This implies $\theta'Q\theta$ is positive definite. It follows that eq.(3.41) is a strictly convex function with respect to σ.

3.5 Appendix B: Proof of Non-singularity of Matrix $T_{i,i}$ for $i = 1, ..., 4$

Referring to the notation in Section 2.7 we see that the matrix $T_{i,i}$ for $i = 1, ..., 4$ is equal to either D or $-D$. Therefore, proving $T_{i,i}$ is non-singular for $i = 1, ..., 4$ is equivalent to proving that the matrix D is non-singular. We can decompose the $N \times N$ matrix D as follows:

$$D = SVS \tag{3.42}$$

where

$$S_{i,j} = \sin(\tfrac{ij\pi}{N+1}) \quad for \quad i,j = 1,...,N \tag{3.43}$$

and V is a diagonal matrix such that

$$V_{i,i} = \tfrac{2i\pi\cosh(i\pi)}{\alpha(N+1)\sinh(i\pi)} \quad for \quad i = 1,...,N \tag{3.44}$$

It can be shown that the matrix S is orthogonal [6]. The matrix V is clearly non-singular. Therefore, the matrix D, the product of three non-singular matrices, is non-singular.

Chapter 4

Estimating a Constant Background σ

4.1 Introduction

In this chapter we study the problem of estimating the conductivity image, $\sigma(r)$, where we assume a constant value of σ in the entire domain. This case represents the coarsest scale at which we would estimate $\sigma(r)$ where in fact the problem now becomes one of scalar estimation. More importantly, it represents a problem from which we can gain insight into the performance of estimating $\sigma(r)$ at a scale coarser than that of the true $\sigma(r)$. In studying this case we hope to understand precisely the relationship between the coarse scale estimate of σ and the true σ at a finer scale. The main result of this chapter is a careful study of the problem of trying to estimate a constant $\sigma(r)$ when in fact the true conductivity piecewise constant in four distinct regions. We pose a simple stochastic model for $\sigma(r)$ and characterize estimation performance as a function of the statistics of $\sigma(r)$.

The following issues will be explored in this chapter.

- 1) If the true conductivity is constant, i.e. $\sigma(r) = \sigma$ in the unit square, and there is additive measurement noise, we write down explicitly the linear

least-squares estimate $\hat{\sigma}$ and the meansquare error $E\{(\sigma - \hat{\sigma})^2\}$.

- 2) We perform a linearization of the PDE in the case where $\sigma(r)$ is piecewise constant in four squares in order to get an approximate relationship between $\sigma(r)$ and the normal derivative of the potential on the boundary which we then use to obtain analytical bounds on performance.

- 3) We investigate how the estimate behaves in the presence of modeling error; i.e. for the problem of estimating a constant $\sigma(r)$ we see how the mean-square error and the bias are affected when in fact the actual conductivity is not constant. We focus on the case where the actual conductivity is piecewise constant in four squares.

- 4) We extend our analysis of the four square case and consider σ to consist of random independent perturbations about a constant background in each of the four squares. We describe analytically and plot via simulations how the bias and the mean-square error of the estimator behave as functions of the variance of σ in the four squares.

4.2 Constant σ Problem Formulation

Let us first consider the case where $\sigma(r)$ is truly constant in the unit square and we wish to estimate this constant based on a number of independent excitations and measurements. In the case of a constant background where $\sigma(r) = \sigma$, the PDE simply becomes Laplace's equation, independent of σ.

$$\nabla^2 \phi(r) = 0 \tag{4.1}$$

with Dirichlet boundary conditions $\phi(s)$ where s is restricted to the boundary of the unit square.

The following is the measurement equation for experiments involving multiple excitations and measurements along the boundary of a unit square given constant σ in the entire square.

$$\begin{aligned} y_i(s) &= \sigma Z_i(s) + n_i(s) \\ E\{n_i(u)n_j(v)\} &= \delta_{i,j}R(u,v) \end{aligned} \tag{4.2}$$

where $1 \leq i \leq M$, M being the number of independent experiments made. y, Z, n represent observation, normal derivative, and noise functions, respectively, defined along the boundary of the unit square, Γ, where $Z_i(s) = \frac{\partial \phi_i(x,y)}{\partial n}\Big|_s$ on Γ and $\phi_i(x,y)$ satisfies eq. (4.1) for a particular set of Dirichlet boundary conditions. The function $R(u,v)$ is defined to be the covariance function of the zero-mean noise function $n(s)$.

The linear least squares estimate(LLSE) of σ is the following.

$$\hat{\sigma} = (\sum_{i=1}^{M} \int_\Gamma du Z_i(u) \int_\Gamma dv R^{-1}(u,v) Z_i(v) dv du)^{-1} \sum_{i=1}^{M} \int_\Gamma du Z_i(u) \int_\Gamma dv R^{-1}(u,v) y_i(v) \tag{4.3}$$

The estimate is unbiased and has the following meansquare error.

$$E\left\{(\sigma - \hat{\sigma})^2\right\} = (\sum_{i=1}^{M} \int_\Gamma Z_i(u) du \int_\Gamma dv R^{-1}(u,v) Z_i(v))^{-1} \tag{4.4}$$

We see that for this simple case of a constant background $\sigma(r)$ we have a linear estimation problem. As one would expect the meansquare error goes down as the number of independent measurements, M, goes up. Note that in the case of Gaussian n_i eq.(4.3) is also the maximum likelihood estimator and is an efficient estimator.

4.3 Linearization of the Four Square Case

Now we consider the case where $\sigma(r)$ is not truly constant in the unit square but rather a 2D function piecewise constant function. In particular we consider the case where $\sigma(r)$ is piecewise constant in four equal subdivisions of the unit square. In this section we solve a linearized version of the problem where the linearization is about some nominal constant background, σ_0. We produce a closed form approximation to the solution of the PDE which is numerically computable. This will be useful later in computing approximate performance characteristics. Referring to Figures 4.1, 4.2 and 4.3 we define the following quantities. For convenience we have adopted a single subscript notation for indexing b.

$$\sigma_i = \sigma_0 + \delta\sigma_i \tag{4.5}$$

$$b_i = b_i^0 + \delta b_i \tag{4.6}$$

$$z_{i,j} = z_{i,j}^0 + \delta z_{i,j} \tag{4.7}$$

Note that we are linearizing the problem about the same value of σ in each square, i.e. σ_0. Our objective is to solve for the potential along the interior boundaries(b_i's) as a function of σ_i and the Dirichlet boundary conditions. Once we know the b_i's we can use these as boundary conditions for solving Laplace's equation in each square, thus giving us the normal derivative of the potential along any boundary. The quantities b_i^0 and $z_{i,j}^0$ are the potential and normal derivative of the potential along the interior boundaries for constant σ_0 background. They are obtained from the solution of Laplace's equation and can be considered to be known. These quantities can be thought of as background field components.

For piecewise constant $\sigma(r)$ solving the PDE for this problem is equivalent to

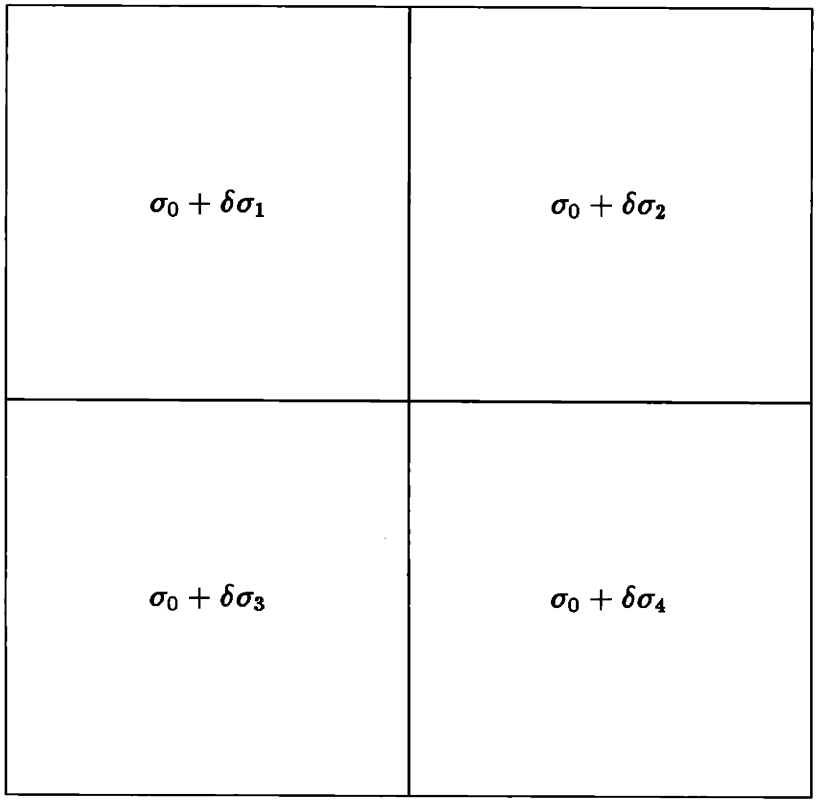

Figure 4.1: σ_i Defined on the Unit Square

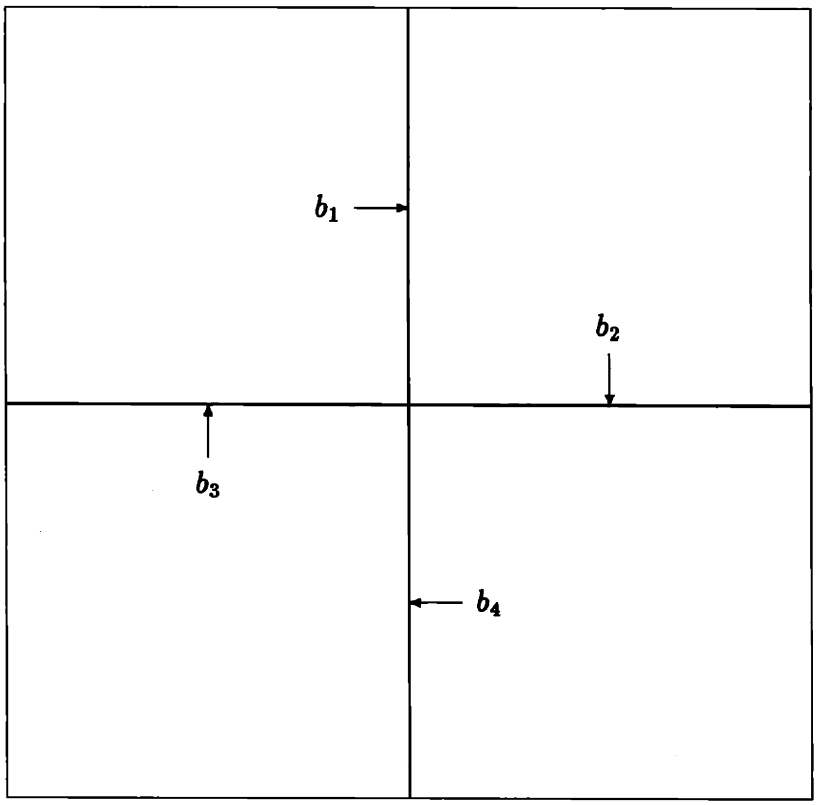

Figure 4.2: $b_i = b_i^0 + \delta b_i$ Defined on the Unit Square

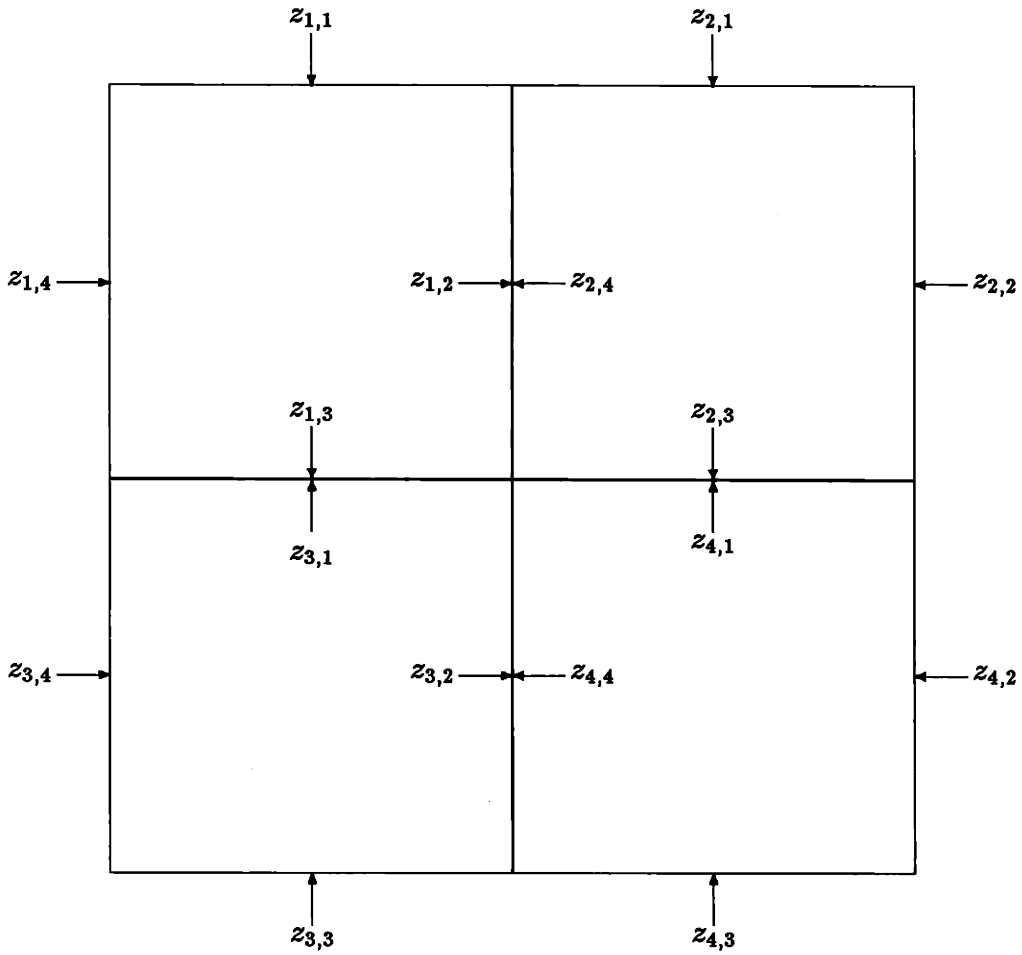

Figure 4.3: $z_{i,j} = z^0_{i,j} + \delta z_{i,j}$ Defined on the Unit Square

matching boundary conditions along each interior boundary. If we require the normal current to be continuous across each interior boundary we arrive at the following four equations.

$$\sigma_1 z_{1,2} = \sigma_2 z_{2,4} \tag{4.8}$$

$$\sigma_2 z_{2,3} = \sigma_4 z_{4,1} \tag{4.9}$$

$$\sigma_3 z_{3,1} = \sigma_1 z_{1,3} \tag{4.10}$$

$$\sigma_4 z_{4,4} = \sigma_3 z_{3,2} \tag{4.11}$$

The above equations are nonlinear with respect to σ_i since the normal derivative functions $z_{i,j}$ are functions of σ_i. If we linearize the equations with respect to σ_i about the nominal point σ_0 we get the following equations.

$$\delta z_{1,2} - \delta z_{2,4} = \frac{z_1^0}{\sigma_0}(\delta\sigma_2 - \delta\sigma_1) \tag{4.12}$$

$$\delta z_{2,3} - \delta z_{4,1} = \frac{z_2^0}{\sigma_0}(\delta\sigma_4 - \delta\sigma_2) \tag{4.13}$$

$$\delta z_{3,1} - \delta z_{1,3} = \frac{z_3^0}{\sigma_0}(\delta\sigma_1 - \delta\sigma_3) \tag{4.14}$$

$$\delta z_{4,4} - \delta z_{3,2} = \frac{z_4^0}{\sigma_0}(\delta\sigma_3 - \delta\sigma_4) \tag{4.15}$$

Since the z_i^0's are field solutions for a constant background and can be considered known, independent of the σ_i's, we confirm that the $\delta z_{i,j}$'s as approximated in eq.'s (4.12)-(4.15) are indeed linear functions of the $\delta\sigma_i$'s.

Note that for the case of a constant σ_0 background $z_{1,2}^0 = z_{2,4}^0$ and for convenience will be denoted as z_1^0. Similarly, $z_{2,3}^0 = z_{3,1}^0 = z_2^0$, etc. The z_i^0's are defined shown in Figure 4.4. The above equations give us the necessary equations to solve for δb_i provided we know the relationship between δb_i and $\delta z_{i,j}$. This relationship can be derived from the fact that Laplace's equation holds within each square; we are

essentially looking for the map from one set of boundary conditions(Dirichlet) to another(Neumann). If we consider the discrete problem so that the $b_{i,j}$'s and $z_{i,j}$'s are now vectors of equally spaced samples of their corresponding functions along the edges of each square, we can use the relationship between Dirichlet($b_{i,j}$) and Neumann($z_{i,j}$) boundary conditions for the problem of Laplace's equation in the square of length α as derived in Chapter 2 (note we are now using the double subscript notation for both the potential and normal derivative vectors along each edge).

$$\begin{pmatrix} z_{n,1} \\ z_{n,2} \\ z_{n,3} \\ z_{n,4} \end{pmatrix} = \begin{pmatrix} D & \hat{H}_o & -S_o & \hat{H}_{to} \\ \hat{H}_o & D & \hat{H}_{to} & -S_o \\ S_o & H_o & -D & H_{to} \\ H_o & S_o & H_{to} & -D \end{pmatrix}_{\alpha} \begin{pmatrix} b_{n,1} \\ b_{n,2} \\ b_{n,3} \\ b_{n,4} \end{pmatrix} \tag{4.16}$$

for $n = 1, ..., 4$ where the particular value of α, i.e. the length of each interior edge, for the case of the unit square being divided into four squares is equal to $1/2$.

If we make the appropriate substitutions into eq.'s (4.12), (4.13),(4.14), and (4.15), we arrive at the following linear system of equations for δb_1,...,δb_4.

$$\begin{pmatrix} 2D & -H_{to} & \hat{H}_{to} & 0 \\ H_{to} & -2D & 0 & -\hat{H}_{to} \\ -H_o & 0 & 2D & \hat{H}_o \\ 0 & H_o & -\hat{H}_o & -2D \end{pmatrix} \begin{pmatrix} \delta b_1 \\ \delta b_2 \\ \delta b_3 \\ \delta b_4 \end{pmatrix} = \frac{1}{\sigma_0} \begin{pmatrix} z_1^0(\delta\sigma_2 - \delta\sigma_1) \\ z_2^0(\delta\sigma_4 - \delta\sigma_2) \\ z_3^0(\delta\sigma_3 - \delta\sigma_4) \\ z_4^0(\delta\sigma_1 - \delta\sigma_3) \end{pmatrix} \tag{4.17}$$

which we can abbreviate as

$$F\delta b = \frac{1}{\sigma_0} G\delta\sigma \tag{4.18}$$

where

$$\delta b = \begin{pmatrix} \delta b_1 \\ \delta b_2 \\ \delta b_3 \\ \delta b_4 \end{pmatrix} \tag{4.19}$$

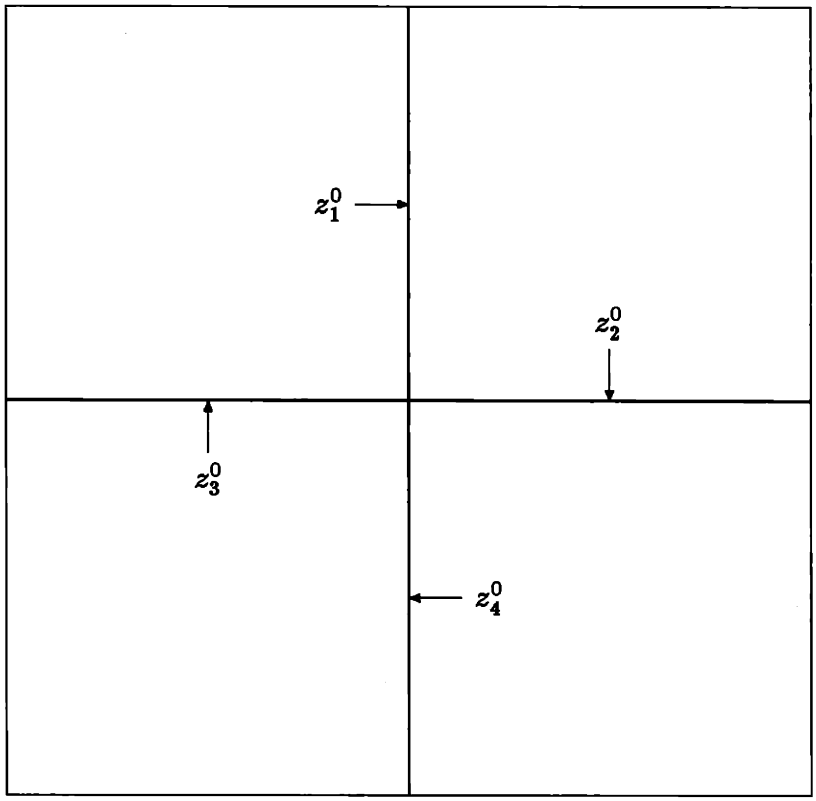

Figure 4.4: Normal Derivative Functions z_i^0 Defined Along Interior Edges

$$\delta\sigma = \begin{pmatrix} \delta\sigma_1 \\ \delta\sigma_2 \\ \delta\sigma_3 \\ \delta\sigma_4 \end{pmatrix} \tag{4.20}$$

$$F = \begin{pmatrix} 2D & -H_{to} & \hat{H}_{to} & 0 \\ H_{to} & -2D & 0 & -\hat{H}_{to} \\ -H_o & 0 & 2D & \hat{H}_o \\ 0 & H_o & -\hat{H}_o & -2D \end{pmatrix} \tag{4.21}$$

$$G = \begin{pmatrix} -z_1^0 & z_1^0 & 0 & 0 \\ 0 & -z_2^0 & 0 & z_2^0 \\ z_3^0 & 0 & -z_3^0 & 0 \\ 0 & 0 & z_4^0 & -z_4^0 \end{pmatrix} \tag{4.22}$$

It can be shown that F is nonsingular(see appendix) and hence we have the following linear relationship between the perturbed interior potential, δb, and the perturbed conductivity, $\delta\sigma$.

$$\delta b = \frac{1}{\sigma_0} F^{-1} G \delta\sigma \tag{4.23}$$

4.4 Estimator Sensitivity to Modeling Error

We now consider the problem of estimation in the presence of modeling errors in σ. For the general development in this section we consider $\sigma(r) = \sigma_0 + \delta\sigma(r)$ where $\delta\sigma(r)$ is a general spatially varying function defined on the unit square. In the next section we specialize the results of this section to the case of $\sigma(r)$ being piecewise constant in four squares. We begin by considering the following measurement equation for the problem of multiple excitations and measurements along the boundary and $\delta\sigma(r)$ some arbitrary function.

$$y_i(s) = (\sigma_0 + \delta\sigma(s))(Z_i^0(s) + \delta Z_i(s)) + n_i(s) \tag{4.24}$$

where s is the restriction to the boundary, Γ, and the noise statistics are the same as before. We define the function Z_i^0 to be the constant σ_0 background field along the boundary due to the *ith* excitation; the function δZ_i is the corresponding perturbed field due to a spatially varying $\delta\sigma$.

The main question is how does our estimator, derived under the assumption of a constant background conductivity, σ_0, perform on data generated from a domain consisting of a constant σ_0 plus deviations $\delta\sigma(r)$ throughout the domain? The problem is fundamentally one of modeling error; i.e. what we're interested in is the sensitivity of the estimator to a particular error in the modeling of the conductivity. Let us characterize this sensitivity by carefully determining the statistics $E\{(\sigma_0 - \hat{\sigma})|\sigma\}$ and $E\{(\sigma_0 - \hat{\sigma})^2|\sigma\}$, where the expectations are taken over the conditional density $p_{y|\sigma}(Y|\sigma)$(the only random quantity here is noise and we are conditioning on a particular $\sigma = \sigma_0 + \delta\sigma$).

The estimate is clearly biased with respect to σ_0; this is easily shown by evaluating $E\{(\sigma_0 - \hat{\sigma})|\sigma\}$. Substituting eq.(4.24) into eq.(4.3) then taking the expectation we arrive at the following.

$$
\begin{aligned}
E\{\hat{\sigma}|\sigma\} &= \sigma_0 \qquad (4.25)\\
&+ \Big(\sum_{i=1}^{M}\int_\Gamma Z_i^0(u)du\int_\Gamma dvR^{-1}(u,v)Z_i^0(v)\Big)^{-1}\sum_{i=1}^{M}\int_\Gamma Z_i^0(u)du\int_\Gamma dvR^{-1}(u,v)\delta\sigma(v)Z_i^0(v)\\
&+ \Big(\sum_{i=1}^{M}\int_\Gamma Z_i^0(u)du\int_\Gamma dvR^{-1}(u,v)Z_i^0(v)\Big)^{-1}\sum_{i=1}^{M}\sigma_0\int_\Gamma Z_i^0(u)du\int_\Gamma dvR^{-1}(u,v)\delta Z_i(v)\\
&+ \Big(\sum_{i=1}^{M}\int_\Gamma Z_i^0(u)du\int_\Gamma dvR^{-1}(u,v)Z_i^0(v)\Big)^{-1}\sum_{i=1}^{M}\int_\Gamma Z_i^0(u)du\int_\Gamma dvR^{-1}(u,v)\delta\sigma(v)\delta Z_i(v)
\end{aligned}
$$

If the bias is defined to be $E\{(\sigma_0 - \hat{\sigma})|\sigma\}$, the last three terms of the above expression is the negative bias. From these three terms we see that the bias is small if $\delta\sigma$ and δZ are small. This analytical expression for the bias is conditioned on

knowing $\delta\sigma$; recall δZ_i is a function of $\delta\sigma$.

Determining $E\{(\sigma_0 - \hat{\sigma})^2|\sigma\}$ is straightforward. Note the random variable $(\sigma_0 - \hat{\sigma})$ in this perturbed case differs from that in the case where $\sigma(r)$ is truly constant only in mean. In other words the perturbation only shifts the probability density of the error. Hence, the variance of $(\sigma_0 - \hat{\sigma})$ is the same in both perturbed and unperturbed cases.

$$\begin{aligned} Var\{(\sigma_0 - \hat{\sigma})|\sigma\} &= E\left\{(\sigma_0 - \hat{\sigma})^2\right\} - \{E\{(\sigma_0 - \hat{\sigma})\}\}^2 \\ &= (\sum_{i=1}^{M} \int_\Gamma Z_i^0(u)du \int_\Gamma dv R^{-1}(u,v) Z_i^0(v))^{-1} \end{aligned} \tag{4.26}$$

The mean-square error can then be written as

$$E\left\{(\sigma_0 - \hat{\sigma})^2|\sigma\right\} = (\sum_{i=1}^{M} \int_\Gamma du Z_i^0(u) \int_\Gamma dv R^{-1}(u,v) Z_i^0(v))^{-1} + \{E\{(\sigma_0 - \hat{\sigma})\}\}^2 \tag{4.27}$$

The modeling error essentially contributes the square of the bias to the meansquare error.

4.5 σ as a Random Variable

In the previous section we treated $\sigma(r)$ as being non-random and we derived measures of performance conditioned on knowing $\sigma(r)$. We now extend our analysis by considering the case where $\sigma(r)$ is a piecewise constant random field, and we proceed to take expectations of the conditional measures of performance over an assumed density for $\sigma(r)$. We concentrate on the case where $\sigma(r)$ is piecewise constant in four equal squares. Associated with each square, indexed by the subscript j, is a σ_j where σ_j is a random variable. Let us consider the case where the σ_j's are each independent Gaussian random variables with mean σ_0 and variance q_j.

Alternatively, $\sigma_j = \sigma_0 + \delta\sigma_j$ where the $\delta\sigma_j$'s are independent, zero-mean Gaussian random variables.

We define the following boundary functions relevant to the four square case. They represent functions defined on subsets of the boundary, Γ.

$$\Gamma = \textit{boundary of the unit square} \tag{4.28}$$

$$\Gamma_j = (\textit{ boundary of jth square }) \cap (\Gamma) \tag{4.29}$$

$$\delta\sigma(v) = \sum_{j=1}^{4} \delta\sigma_j u_j(v) \tag{4.30}$$

$$u_j(v) = \begin{cases} 1 & \textit{on } \Gamma_j \\ 0 & \textit{elsewhere} \end{cases} \tag{4.31}$$

$$z_j^0(v) = z^0(v)u_j(v) \tag{4.32}$$

$$\delta z_j^0(v) = \delta z^0(v)u_j(v) \tag{4.33}$$

$$n_j(v) = n(v)u_j(v) \tag{4.34}$$

$$R_j(u,v) = E\{n_j(u)n_j(v)\} \tag{4.35}$$

We rewrite the expectations, eq.'s (4.25) and (4.26), for the case of four squares treating them as conditional expectations conditioned on the four σ_j's. For simplicity we take the number of measurements, M, to be 1. Note in the following the subscript, j, indexes the square on which the associated functions is defined; it is not to be confused with the subscript, i, used previously to index measurements.

$$
\begin{aligned}
E\{\hat{\sigma}|\sigma\} &= \sigma_0 \\
&+ \gamma \sum_{j=1}^{4} \delta\sigma_j \int_{\Gamma_j} du z_j^0(u) \int_{\Gamma_j} dv R_j^{-1}(u,v) z_j^0(v) \\
&+ \gamma \sum_{j=1}^{4} \sigma_0 \int_{\Gamma_j} du z_j^0(u) \int_{\Gamma_j} dv R_j^{-1}(u,v) \delta z_j(v) \\
&+ \gamma \sum_{j=1}^{4} \delta\sigma_j \int_{\Gamma_j} du z_j^0(u) \int_{\Gamma_j} dv R_j^{-1}(u,v) \delta z_j(v)
\end{aligned} \tag{4.36}
$$

where

$$
\gamma = (\sum_{j=1}^{4} \int_{\Gamma_j} du z_j^0(u) \int_{\Gamma_j} dv R_j^{-1}(u,v) z_j^0(v))^{-1} \tag{4.37}
$$

In Section 4.3 where we linearized the PDE about a nominal constant background σ_0, we came up with analytical expressions for an approximation to the solution of the PDE as a function of the perturbations in σ_0. The case of $\sigma(r)$ being piecewise constant in four squares was considered where $\sigma_j = \sigma_0 + \delta\sigma_j$ for $j = 1, ..., 4$. We can now substitute these expressions into eq.(4.36), giving us an expression for $E\{\hat{\sigma}|\sigma\}$ explicitly as a function of the perturbations $\delta\sigma_j$. This will allow us to take the expectation of $E\{\hat{\sigma}|\sigma\}$ with respect to the densities of the $\delta\sigma_j$'s.

Again, we consider the discrete case where the vectors $z_j^0, \delta z_j$ consist of equally spaced samples of their corresponding functions. The covariance matrix, R_j, is the discrete version of the covariance function, $R_j(u,v)$. Recall eq.(4.23), which is a linearized expression for the interior potential vector, δb, as a function of the perturbations, $\delta\sigma$. From our earlier expression relating Dirichlet to Neumann boundary conditions, eq.(4.16), δz_j is just a linear function of δb. For example, referring to Figure 4.3 for $\delta z_{i,j}$, if we let

$$
\delta z_1 = \begin{pmatrix} \delta z_{1,1} \\ \delta z_{1,4} \end{pmatrix} \tag{4.38}
$$

we obtain the following expression,

$$\delta z_1 = L_1 \frac{1}{\sigma_0} F^{-1} G \delta\sigma \tag{4.39}$$

where

$$L_1 = \begin{pmatrix} \hat{H}_0 & 0 & -S_o & 0 \\ S_o & 0 & H_{to} & 0 \end{pmatrix} \tag{4.40}$$

Similarly, we can come up with expressions for $L_2 ... L_4$.

Substituting the above expressions for δz_j into eq.(4.36) then taking the expectation of the result over $\sigma(r)$ we get the following.

$$\begin{aligned} E_\sigma \{E\{\hat{\sigma}|\sigma\}\} &= \sigma_0 \\ &+ \frac{\gamma}{\sigma_0} \sum_{j=1}^{4} q_j \delta l_j' R_j^{-1} z_j^0 \end{aligned} \tag{4.41}$$

$$\begin{aligned} \gamma &= (\sum_{j=1}^{4} (z_j^0)' R_j^{-1} z_j^0)^{-1} \\ \delta l_j &= \left\{ L_j (F^{-1} G) \right\}_{jth} \end{aligned} \tag{4.42}$$

where "jth" denotes the jth column of the associated matrix. Note eq.(4.41) is not exact in the sense that we have used linearized expressions for the δz_j's. What we have then is the following analytical approximation to the bias.

$$\begin{aligned} bias &= E_\sigma \{E\{(\sigma_0 - \hat{\sigma})|\sigma\}\} \\ &= -\frac{\gamma}{\sigma_0} \sum_{j=1}^{4} q_j \delta l_j' R_j^{-1} z_j^0 \end{aligned} \tag{4.43}$$

As for the mean-square error rewriting eq.(4.27) for the four square case we have,

$$E\left\{(\sigma_0 - \hat{\sigma})^2 | \sigma\right\} = \gamma + \{E\{(\sigma_0 - \hat{\sigma})|\sigma\}\}^2 \tag{4.44}$$

$$
\begin{aligned}
&= \gamma \\
&+ \gamma^2 \left\{ \sum_{j=1}^{4} \delta\sigma_j \int_{\Gamma_j} du z_j^0(u) \int_{\Gamma_j} dv R_j^{-1}(u,v) z_j^0(v) \right. \\
&+ \sum_{j=1}^{4} \sigma_0 \int_{\Gamma_j} du z_j^0(u) \int_{\Gamma_j} dv R_j^{-1}(u,v) \delta z_j(v) \\
&+ \left. \sum_{j=1}^{4} \delta\sigma_j \int_{\Gamma_j} du z_j^0(u) \int_{\Gamma_j} dv R_j^{-1}(u,v) \delta z_j(v) \right\}^2
\end{aligned}
$$

Let us rewrite eq.(4.44) as follows.

$$
\begin{aligned}
E\left\{(\sigma_0 - \hat{\sigma})^2 | \sigma\right\} &= \gamma + \nu \\
\nu &= \gamma^2 \left\{ \sum_{j=1}^{4} \delta\sigma_j \int_{\Gamma_j} du z_j^0(u) \int_{\Gamma_j} dv R_j^{-1}(u,v) z_j^0(v) \right. \\
&+ \sum_{j=1}^{4} \sigma_0 \int_{\Gamma_j} du z_j^0(u) \int_{\Gamma_j} dv R_j^{-1}(u,v) \delta z_j(v) \\
&+ \left. \sum_{j=1}^{4} \delta\sigma_j \int_{\Gamma_j} du z_j^0(u) \int_{\Gamma_j} dv R_j^{-1}(u,v) \delta z_j(v) \right\}^2
\end{aligned}
\tag{4.45}
$$

Let us also define the expectation of eq.(4.45) over $\sigma(r)$.

$$
\begin{aligned}
E_\sigma\left\{E\left\{(\sigma_0 - \hat{\sigma})^2 | \sigma\right\}\right\} &= \gamma + \epsilon \\
\epsilon &= E_\sigma\{\nu\}
\end{aligned}
\tag{4.46}
$$

We see that $E_\sigma\{E\{(\sigma_0 - \hat{\sigma})^2|\sigma\}\}$ is the sum of two terms, γ and ϵ. Note γ is the mean-square error for the case where there exists no modeling error. This term owes its contribution entirely to the measurement noise. The term ϵ on the other hand is due entirely to errors in modeling $\sigma(r)$. Note for the case of $R_j(u,v) = \mu\delta(u,v)$ where μ is the noise variance the effect of the measurement noise covariance on ϵ cancels out in the term ϵ; i.e. the term ϵ does not even depend on μ. The particular $R_j(u,v)$ considered amounts to the case where the noise is white along the boundary.

We could proceed to substitute our linearized expressions for δz_j into eq.(4.44) then take the expectation of the result over $\sigma(r)$ as we did for the bias. The resulting expectation would consist of 2nd, 3rd, and 4th order moments of $\delta\sigma_j$ along with various cross products of these moments(see appendix). Note the final expression would again be an approximation since we have based our analysis on the linearized PDE.

In summary what we have derived are approximations to the bias and the mean-square error as a function of the statistics of a four square model for $\sigma(r)$ based on a linearization of the PDE. These expressions can be numerically computed to give estimates of performance in the presence of errors in modeling $\sigma(r)$. One could also approximate these performance measures using the following Monte-Carlo method.

1. Generate synthetic data based on the four square model of $\sigma(r)$ where the $\delta\sigma_j$'s are generated with a random number generator with appropriate statistics.

2. Compute $\hat{\sigma}$ using the estimator equation, eq.(4.3).

3. Compute $(\sigma_0 - \hat{\sigma})$ for the bias, $(\sigma_0 - \hat{\sigma})^2$ for the mean-square error.

4. Repeat first three steps until the number of trials is sufficient.

5. Form computed expectations

$$\frac{1}{L}\sum_{n=1}^{L}(\sigma_0 - \hat{\sigma})$$

and

$$\frac{1}{L}\sum_{n=1}^{L}(\sigma_0 - \hat{\sigma})^2$$

where L is the total number of trials.

This method essentially computes the expectation of the conditional bias and mean-square error over σ by computing the bias and mean-square error conditioned on specific values of $\delta\sigma_j$ over many trials then averaging the results of those trials. The computations become more accurate as the number of trials increases. Note for white measurement noise along the boundary the estimator equation, eq.(4.3), the conditional bias equation, eq.(4.36), and ν in eq.(4.45), do not depend on the value of the noise variance.

4.6 Numerical Computations-Simulations

In this section we explore the effects of modeling errors in $\sigma(r)$ on bias and mean-square error by numerically computing some of the expressions derived in the previous section and also by running large numbers of Monte-Carlo trials on simulated data. These computations will help to quantify the effects of modeling error on bias and mean-square error and in doing so allow us to compare these two performance measures to those in the absence of modeling error. In the case of bias we wish to see how significant the effect is compared to the unbiased, modeling error-free case. With regard to the mean-square error we wish to compare the magnitudes of the contributions due to measurement noise and to modeling errror. It would be useful to see, for example, how severe the modeling error must be before the mean-square error exceeds that in the absence of modeling error in which case the mean-square error is due solely to measurement noise. The modeling error being considered in all computations to follow is precisely the one developed in Section 4.5; i.e. in estimating $\sigma(r)$ one assumes a constant background, σ_0, when in fact $\sigma(r)$ is piecewise constant in four squares and the value of $\sigma(r)$ in each square is an independent

Gaussian random variable with mean σ_0 and variance q_j where j indexes the square.

The synthetic data used in performing the Monte-Carlo trials is generated on a 64 by 64 point grid using a five-point finite difference scheme representating an approximation of the PDE with a truncation error of order $O(h^2)$ where $h = 1/N$ and the grid is N by N. We run Monte-Carlo simulations in order to come up with plots of the bias and the mean-square error as functions of the variance of $\sigma(r)$ in each of the four squares. In all of our computations to follow we consider a single measurement scheme with the following boundary excitations defined along the boundary of the unit square.

$$\phi(x,y) = \begin{cases} \sin(2\pi x) & y = 1, \quad 0 \le x \le 1 \\ -\sin(2\pi y) & 0 \le y \le 1, \quad x = 1 \\ -\sin(2\pi x) & y = 0, \quad 0 \le x \le 1 \\ -\sin(2\pi y) & 0 \le y \le 1, \quad x = 0 \end{cases} \tag{4.47}$$

4.6.1 Bias

We compute our linearized approximation to the bias using the expression in eq.(4.43). Note from eq.(4.43) the bias is linearly related to the variance of $\delta\sigma_j$. That is

$$\begin{aligned} bias &= E_\sigma \{E\{(\sigma_0 - \hat{\sigma})|\sigma\}\} \\ &= -\frac{\gamma}{\sigma_0} \sum_{j=1}^{4} q_j \delta l_j' R_j^{-1} z_j^0 \end{aligned} \tag{4.48}$$

Using our boundary conditions, eq.(4.47), and letting q_j be the same for $j = 1, ..., 4$, we compute $\frac{\gamma}{\sigma_0}\delta l_j' z_j^0$ where we have taken R_j to be the identity matrix. The field components making up elements of the matrix G as defined in eq.(4.22)) are computed by solving Laplace's equation using a five-point finite difference scheme. For a discretization of 64 by 64 we obtain the computed quantities in Table 4.1. We see

$\delta l_1' z_1^0$	-1.22626e-09
$\delta l_2' z_2^0$	3.959146e-10
$\delta l_3' z_3^0$	4.181981e-08
$\delta l_4' z_4^0$	5.910316e-08
γ	4.56513e-05
$\gamma \sum_{j=1}^{4} \delta l_j' z_j^0$	4.56936e-12

Table 4.1: Relevant Quantities in Computing the Bias

from above that the slope,

$$\frac{\gamma}{\sigma_0} \sum_{j=1}^{4} \delta l_j' z_j^0$$

, is on the order of $10^{-12}/\sigma_0$. This shows that at least to first order the bias is insignificant for this particular set of boundary conditions, eq.(4.47). This would seem reasonable and to be somewhat expected considering our boundary conditions, which probe the region symmetrically about the boundary.

In order to verify this effect for even larger perturbations we can simulate the bias using Monte-Carlo trials and averaging results. The result is the plot in Figure 4.5. The plot shows negligible bias for even large variances of $\delta\sigma$ giving further indication of the fact that the estimator is strongly unbiased with respect to modeling errors. Again, this is not surprising given that the excitations are symmetrical about the boundary and the deviations in σ are independent and identically distributed in each square.

4.6.2 Mean-square Error

We now examine the mean-square error as a function of the variance of σ in each square, q. We consider the case in which the measurement noise is white along the boundary. Looking at the expression for the mean-square error, eq.(4.46), we see that it is a sum of two terms; a term due to noise, γ, which can also be thought

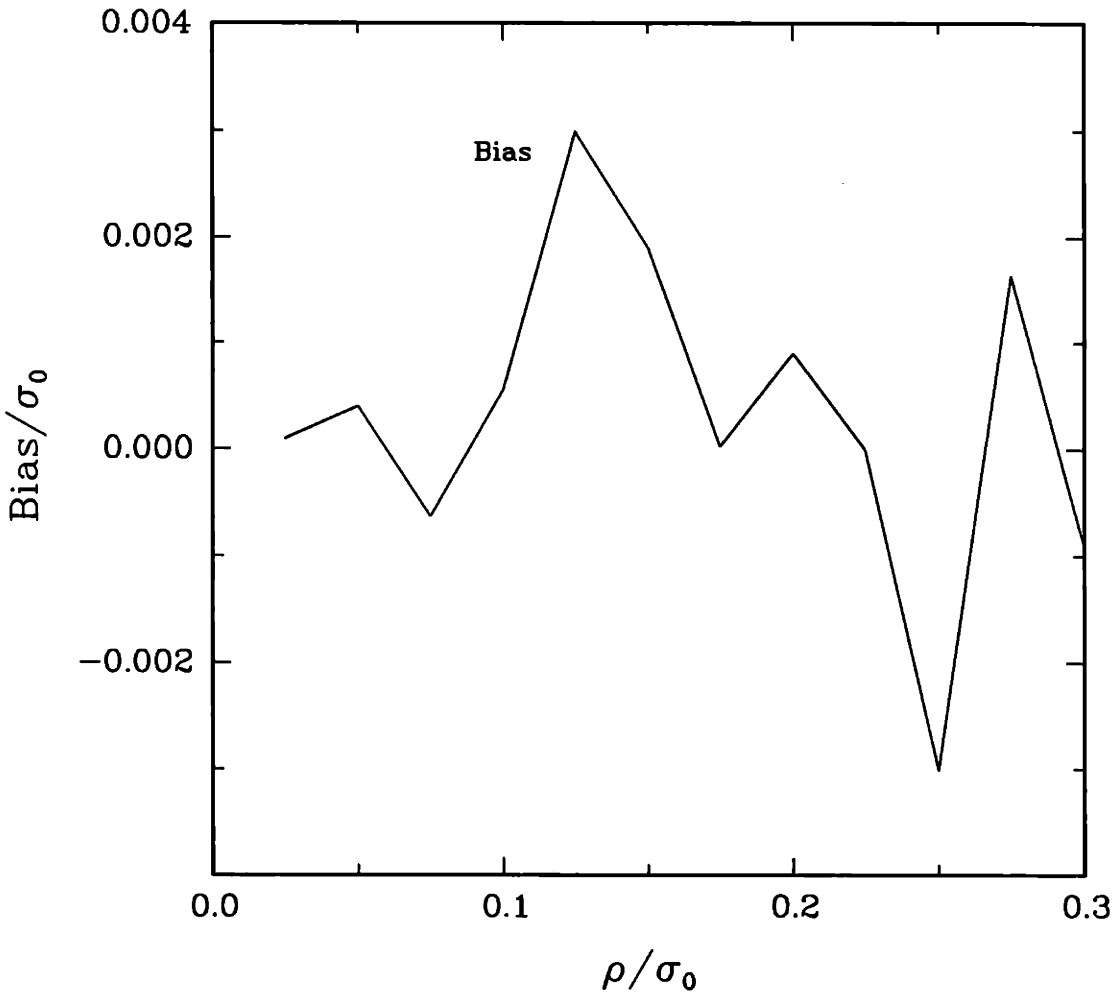

Figure 4.5: Bias vs. ρ, 1000 trials

of as the mean-square error in the absence of modeling error, and the term, ϵ, which accounts for the modeling error. The mean-square error term, ϵ, is solely a function of the distortion in modeling $\sigma(r)$. In our case the distortion consists of modeling $\sigma(r) = \sigma_0$ when in fact $\sigma(r)$ is piecewise constant in four squares. The mean-square error term, γ, on the other hand, is solely a function of the signal to noise ratio(SNR). We wish to compare these two terms, ϵ and γ, in order to see how estimation errors due to modeling distortions compare to errors due solely to measurement noise.

We compare these two terms by plotting the percentage error, $\sqrt{\epsilon}/\sigma_0$, as a function of the percentage distortion in $\sigma(r)$, ρ/σ_0, where ρ is the square-root of q, and comparing this to a plot of $\sqrt{\gamma}/\sigma_0$ versus the inverse SNR. The SNR is defined to be the root mean-square of the signal, σz_0 along the boundary, divided by the standard deviation of the additive white measurement noise. Specifically,

$$SNR = \frac{signal \quad rms}{noise \quad dev} \tag{4.49}$$

$$signal \quad rms = \sqrt{\frac{1}{4}\int (\sigma(s) z_0(s))^2 ds}$$

We use eq.(4.37) to compute γ. We see from eq.(4.37) that for the case of white measurement noise with covariance function $R(u,v) = \mu\delta(u,v)$ γ is proportional to μ so that

$$\gamma = \mu(\sum_{j=1}^{4} \int_{\Gamma_j} du (z_j^0(u))^2)^{-1} \tag{4.50}$$

Computing γ requires only the normal derivative vectors for the constant background case. We use Monte-Carlo trials in order to compute ϵ.

Figure 4.6 is a plot of ϵ/σ_0 as a function of ρ/σ_0. On this plot we have also indicated γ/σ_0 for various levels of inverse SNR. Figure 4.7 is a plot of γ/σ_0 as a

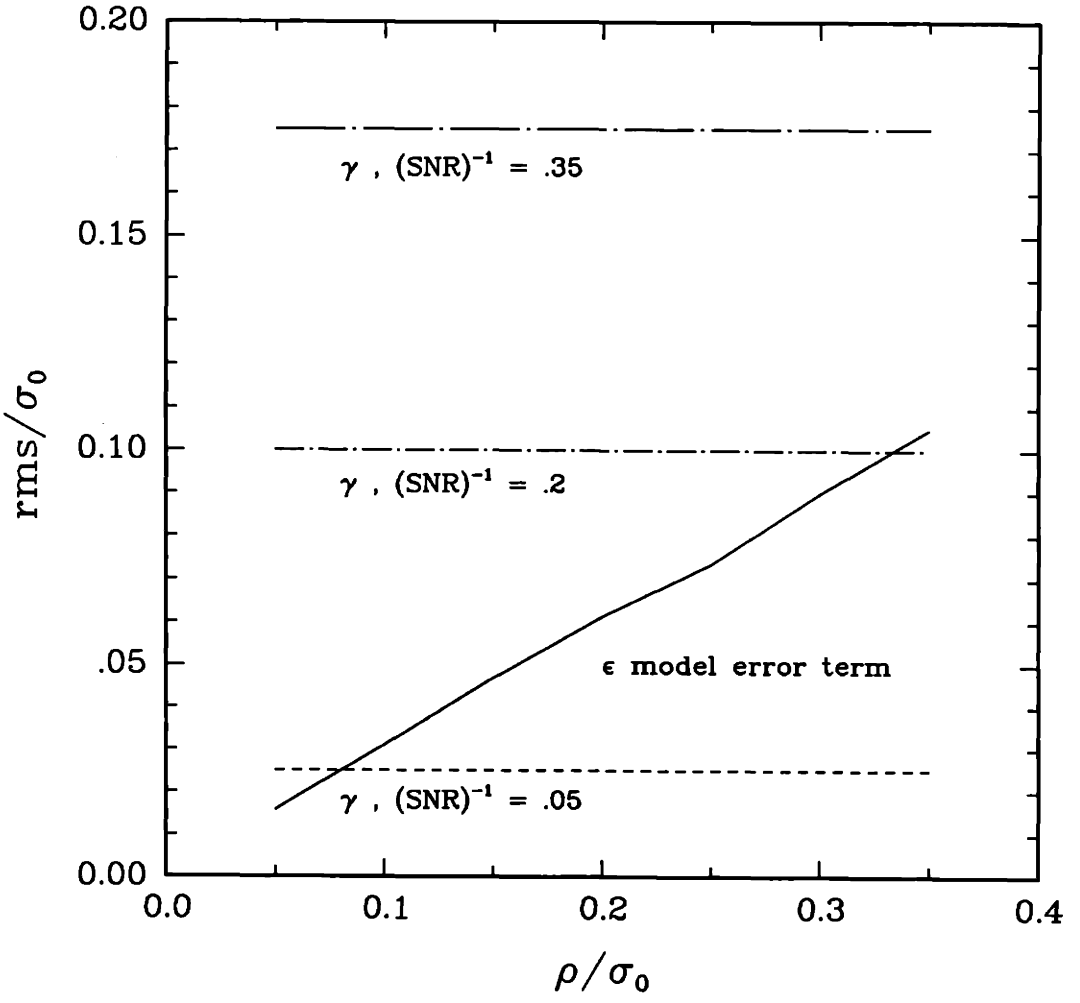

Figure 4.6: ϵ/σ_0 vs. ρ/σ_0, 1000 trials

function of $(SNR)^{-1}$. On this plot we have also indicated ϵ/σ_0 for various levels of distortion, ρ/σ_0. Finally, Figure 4.8 is a plot that combines the two previous plots into one by plotting both ϵ/σ_0 vs. ρ/σ_0 and γ/σ_0 vs. $(SNR)^{-1}$. We see from these three plots that the mean-square error term due to modeling error is actually less than the term due to noise when the distortion due to errors in modeling σ is equivalent to the inverse SNR. Specifically, if the inverse SNR is equal to the percentage distortion in σ, ρ/σ_0, the ratio between the two errors, ϵ/γ, is approximately 3/5. This indicates that the estimator is quite robust with respect to modeling errors in σ.

4.7 Conclusions

In this chapter we have studied estimation performance for the problem of estimating $\sigma(r)$ assuming a constant σ_0 background when in fact it is not constant. We studied the effects of this modeling error on the bias and the mean-square error in the specific case in which the true conductivity is piecewise constant in four squares. A linearization of the PDE provided expressions for Cramer-Rao-like bounds on estimation performance.

We see from simulations and resulting plots that the estimator is quite robust with respect to modeling errors; i.e. for the case of $\sigma(r)$ being piecewise constant in four squares we can allow the variance of $\sigma(r)$ in each square to be quite large(i.e. on par with the noise) without significantly affecting the mean-square error. Also, the estimator is strongly unbiased in the presence of modeling errors. This is not surprising since we are applying excitations symmetrically about the boundary. Consequently, the net effect of the modeling error, i.e. using symmetric boundary

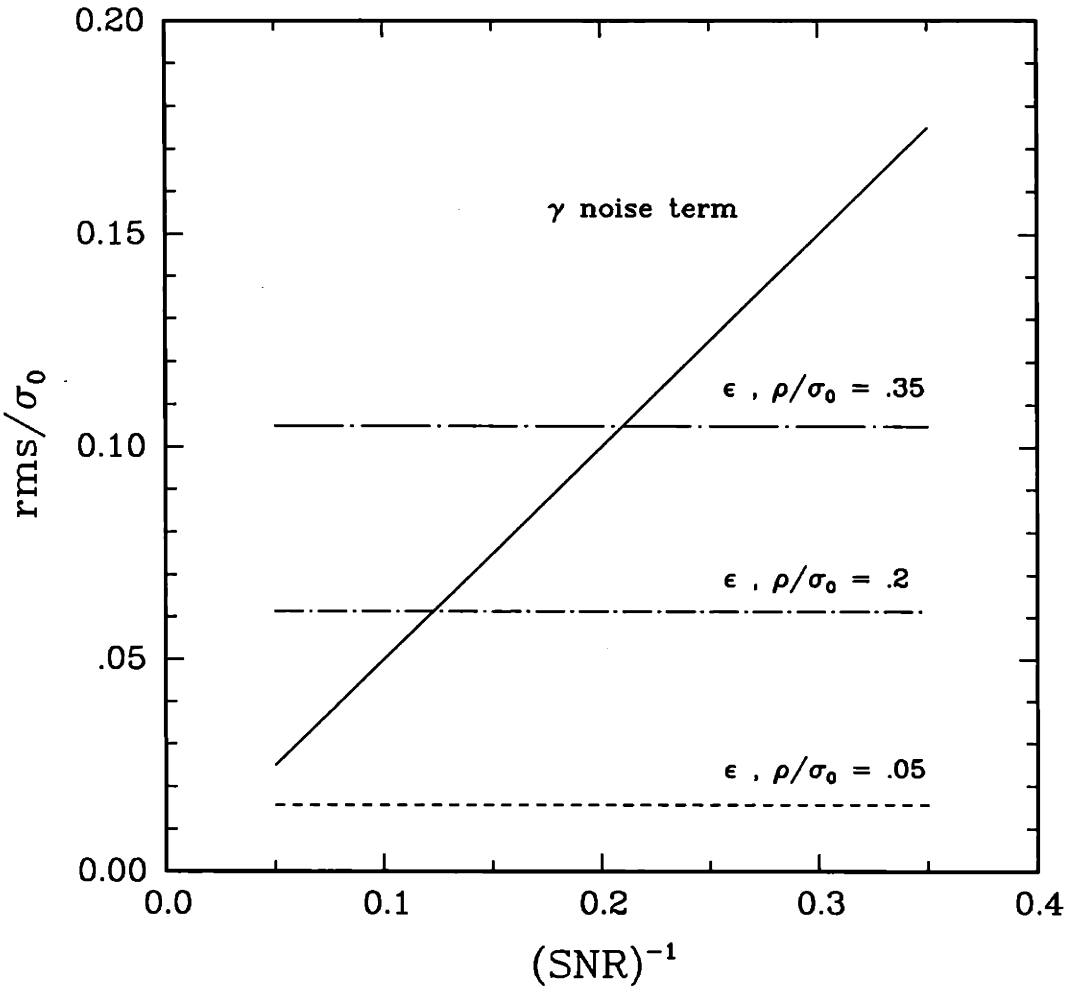

Figure 4.7: γ/σ_0 vs. $(SNR)^{-1}$, 1000 trials

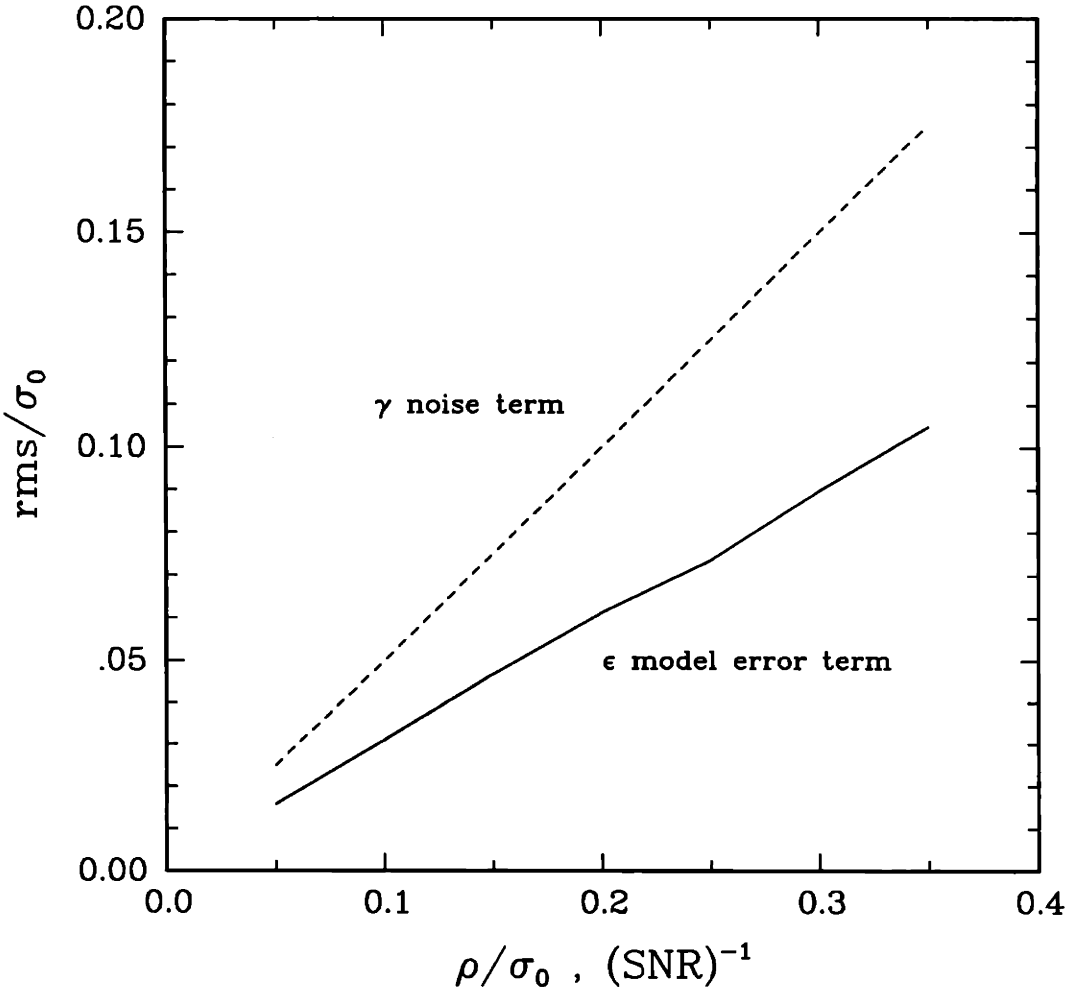

Figure 4.8: ϵ/σ_0 vs. ρ/σ_0, γ/σ_0 vs. $(SNR)^{-1}$, 1000 trials

excitations to estimate a constant $\sigma(r)$ when in fact it is piecewise constant in four squares, is essentially the addition of a "noise" term to the mean-square error.

As a final note we mention that we have analytical expressions for the bias and mean-square errors as functions of the deviations of $\sigma(r)$ from a constant background. As we discussed in Chapter 1 this may suggest a way of developing true multi-grid estimation schemes [2] in which iterations proceed both from coarse scales to fine scales and from fine to coarse.

4.8 Appendix: Expectation of Mean-square Error Over σ

We rewrite our expression for the conditional mean-square error, eq.(4.44), as follows.

$$\begin{aligned} E\left\{(\sigma_0-\hat{\sigma})^2|\sigma\right\} &= \gamma \\ &+ \gamma^2\left\{\sum_{j=1}^{4}\delta\sigma_j\alpha_j\right. \\ &+ X \\ &+ \left.\sigma_0\sum_{j=1}^{4}\sum_{i=1}^{4}\beta_{j,i}\delta\sigma_i\right\}^2 \end{aligned} \tag{4.51}$$

where

$$\alpha_j = \int_{\Gamma_j} du z_j^0(u)\int_{\Gamma_j} dv R_j^{-1}(u,v)z_j^0(v) \tag{4.52}$$

$$\beta_{j,i} = \int_{\Gamma_j} du z_j^0(u)\int_{\Gamma_j} dv R_j^{-1}(u,v)\sum_{i=1}^{4}\delta\sigma_i w_j \tag{4.53}$$

$$X = \sum_{j=1}^{4}\delta\sigma_j\sum_{i=1}^{4}\beta_{j,i}\delta\sigma_i \tag{4.54}$$

and w_j can be derived from our linearization techniques. By performing an expectation over σ we get the following:

$$\begin{aligned} E_\sigma\left\{E\left\{(\sigma_0-\hat{\sigma})^2|\sigma\right\}\right\} &= \gamma \\ &+ \sum_{j=1}^{4}\alpha_j^2 q_j + \sigma_0^2\sum_{k=1}^{4}\sum_{j=1}^{4}\sum_{i=1}^{4}\beta_{k,i}\beta_{j,i}q_i \\ &+ 2\sigma_0\sum_{j=1}^{4}\sum_{i=1}^{4}\beta_{j,i}\alpha_i q_i + 2E_\sigma\left\{\sigma_0\sum_{j=1}^{4}\sum_{i=1}^{4}\beta_{j,i}\delta\sigma_i X\right\} \\ &+ 2E_\sigma\left\{\sum_{j=1}^{4}\delta\sigma_j\alpha_j X\right\} + E_\sigma\left\{X^2\right\} \end{aligned} \tag{4.55}$$

where

$$E\left\{\delta\sigma_i\delta\sigma_j\right\} = \begin{cases} 0 & for \quad i\neq j \\ q_j & for \quad i=j \end{cases} \tag{4.56}$$

This expression can be simplified, making it a function solely of second order moments, by using Gaussian moment factoring. The result is the following:

$$\begin{aligned} E_\sigma\left\{E\left\{(\sigma_0-\hat{\sigma})^2|\sigma\right\}\right\} &= \gamma \\ &+ \sum_{j=1}^{4}\alpha_j^2 q_j + \sigma_0^2\sum_{k=1}^{4}\sum_{j=1}^{4}\sum_{i=1}^{4}\beta_{k,i}\beta_{j,i}q_i \\ &+ 2\sigma_0\sum_{j=1}^{4}\sum_{i=1}^{4}\beta_{j,i}\alpha_i q_i \\ &+ \sum_{i=1}^{4}3\beta_{i,i}^2 q_i^2 \end{aligned} \tag{4.57}$$

Chapter 5

Estimating Four Squares

5.1 Introduction

In this chapter we assume a model for σ that is piecewise constant in four squares and study the problem of trying to estimate σ in those four squares. We analyze estimation performance bounds for this problem in a manner similar to that of the last chapter. However, in this chapter we focus on the problem of estimating σ when in fact it is piecewise constant in four squares. We derive the actual Cramer-Rao bound for estimating four squares in the absence of any modeling errors. Although there is no modeling error in this case, the problem is nevertheless one of non-linear estimation. We are once again led to performing a linearization of the PDE in order to derive bounds on estimation performance. We also explore the performance of our algorithm for the first time; i.e. accuracy and speed of convergence of the algorithm.

The following issues will be explored in this chapter.

- We derive the Cramer-Rao bound for estimating σ in the case of four squares via linearization of the PDE about four nominal values $\sigma_1^0,\ldots,\sigma_4^0$.

- We compute the bound for various σ_1^0,...,σ_4^0. In particular we investigate the effect of these various backgrounds on the mean-square error in estimating σ_1,...,σ_4. We also investigate the correlation between the errors in estimating σ_1,...,σ_4.

- We compute the bound for various excitations, demonstrating the effects of the excitation scheme on performance.

- We perform numerical experiments based on two excitation schemes, one single, the other multiple, in order to study the performance of the algorithm based on synthetic data.

5.2 Problem Formulation

The fundamental problem is one of estimating $\sigma(x, y)$ in the unit square when our model for $\sigma(x, y)$ consists of a 2D piecewise constant function as illustrated in Figure 5.1. We define σ, the vector which we are interested in estimating, as follows.

$$\sigma = \begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_2 \\ \sigma_4 \end{pmatrix} \tag{5.1}$$

Since the primary purpose for deriving the Cramer-Rao bound for this problem is to compute the bound, for convenience we focus from the start on the discrete problem. We consider the unit square to be an $N \times N$ grid. Potential and normal derivative vectors represent equally spaced samples of their corresponding functions along edges of each square indexed according to the conventions described in

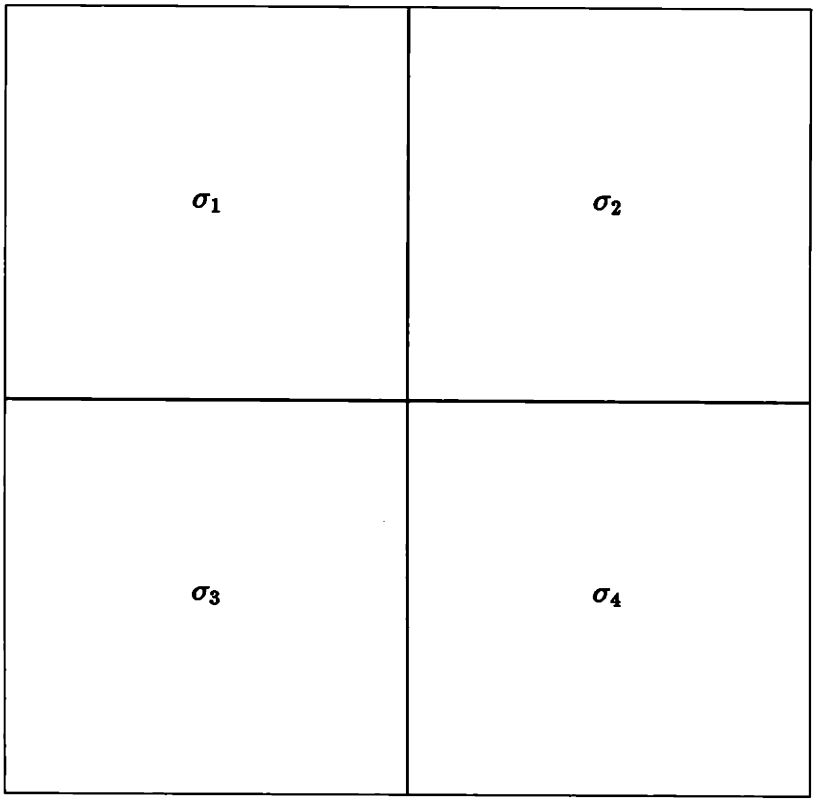

Figure 5.1: Four Square $\sigma(x, y)$ Defined on the Unit Square

Chapter 2. Since an edge of the unit square is of length N samples and the edges corresponding to each square is half this length, these vectors are $N/2 \times 1$.

We have the following observation equation:

$$r_k = \{h_k(\sigma)\}' + n_k \tag{5.2}$$

for $k = 1, ..., L$, where L is the number of experiments. The vectors r_k, and n_k are each $4N \times 1$. The row vector $h_k(\sigma)$ is $1 \times 4N$. We assume the vectors n_k, for $k = 1, ..., L$ are zero-mean, jointly Gaussian such that $E\{n_m n_n\} = R_m \delta_{mn}$ for $m, n = 1, ..., L$; i.e. the vectors are uncorrelated from experiment to experiment. The vector $h_k(\sigma)$ consists of normal derivative vectors multiplied by the value of $\sigma(x, y)$ in the square to which the vector is associated. We define $h_k(\sigma)$ in the following way.

$$\{h_k(\sigma)\}' = S\mathbf{z}_k \tag{5.3}$$

where

$$S = \begin{pmatrix} \sigma_1 I & 0 & 0 & 0 \\ 0 & \sigma_2 I & 0 & 0 \\ 0 & 0 & \sigma_3 I & 0 \\ 0 & 0 & 0 & \sigma_4 I \end{pmatrix}$$

$$\mathbf{z}_k = \begin{pmatrix} z_{1,4} \\ z_{1,1} \\ z_{2,1} \\ z_{2,2} \\ z_{3,3} \\ z_{3,4} \\ z_{4,2} \\ z_{4,3} \end{pmatrix}_k \tag{5.4}$$

(5.5)

where I is the identity matrix of dimension N. As consistent with indexing conventions established in Chapter 2 the vectors $z_{i,j}$ are as shown in Figure 5.2. For ease of notation rather than add another subscript to each of the vectors $z_{i,j}$ we have simply subscripted the entire vector $\mathbf{z}_k$ to denote that each of the vectors $z_{i,j}$ correspond to the normal derivative vectors for experiment k.

We also have the PDE which must be satisfied within the domain for each experiment. The solution of the PDE can be obtained by matching boundary conditions along each internal edge. As in Chapter 3 these boundary constraints are represented by the following four equations where the $z_{i,j}$'s represent normal derivative vectors for a particular experiment.

$$\sigma_1 z_{1,2} = \sigma_2 z_{2,4} \tag{5.6}$$

$$\sigma_2 z_{2,3} = \sigma_4 z_{4,1} \tag{5.7}$$

$$\sigma_3 z_{3,1} = \sigma_1 z_{1,3} \tag{5.8}$$

$$\sigma_4 z_{4,4} = \sigma_3 z_{3,2} \tag{5.9}$$

Given the observation equation, eq.(5.2), and the constraints of the PDE, eq.'s (5.6),...,(5.9), for an arbitrary number of experiments, the objective is to come up with an estimate of the vector σ. In the next section we come up with a lower bound on the performance of such an estimate based on the linearization of the PDE about some nominal background $\sigma(x, y)$.

5.3 Cramer-Rao Bound

The Cramer-Rao bound gives a lower bound on the error covariance of an estimate based on the conditional density, $p_{r|\sigma}(R|\sigma)$, where σ is our 4×1 parameter vector

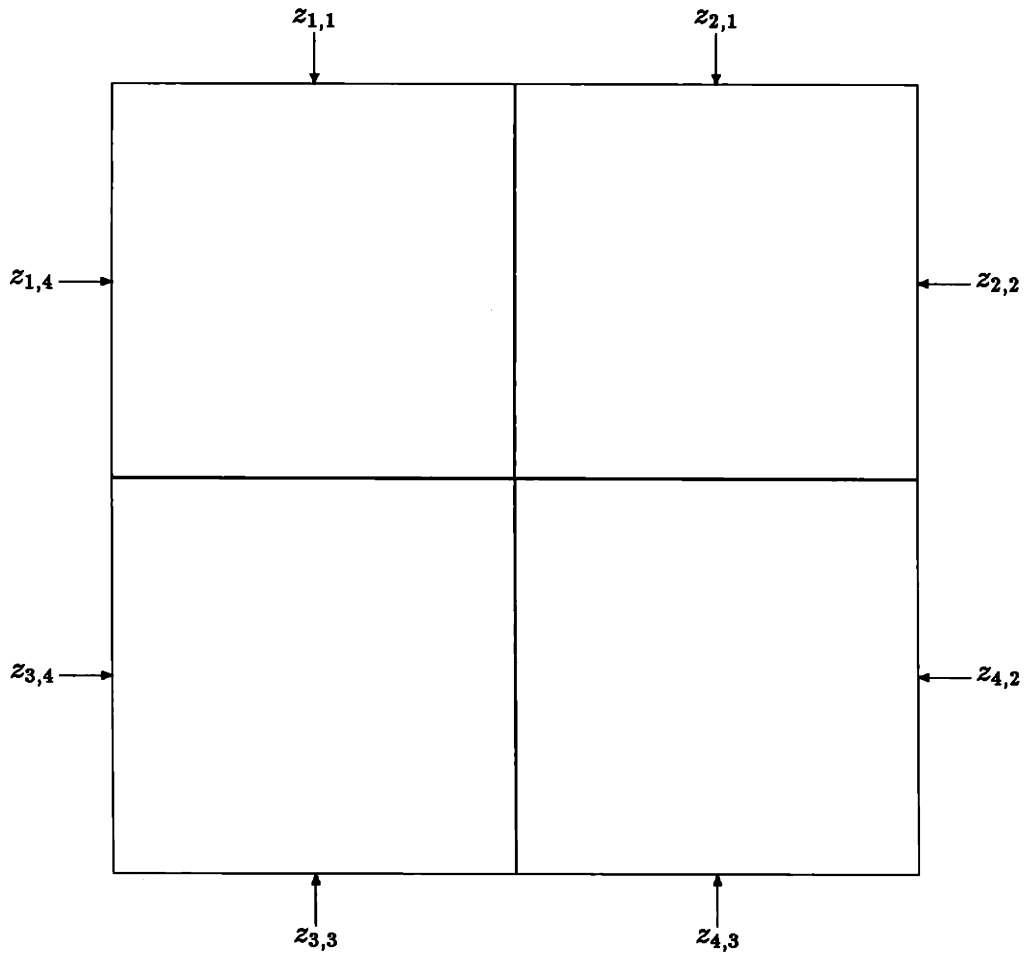

Figure 5.2: Normal Derivative Vectors z_{ij} Defined on the Boundary of the Unit Square

and $r = \{r_1, ..., r_L\}$, the set of L observation vectors [13]. For the multiple parameter case the Cramer-Rao bound provides a bound on the covariance matrix of the error in estimating the parameter vector. The diagonal elements of the bound are bounds on individual parameter error variances while the off-diagonal elements give us an indication of the correlation among these errors.

Let

$$\hat{\sigma} = \begin{pmatrix} \hat{\sigma_1} \\ \hat{\sigma_2} \\ \hat{\sigma_3} \\ \hat{\sigma_4} \end{pmatrix} \tag{5.10}$$

denote an unbiased estimate of σ. Let us also define the following error vector and its associated covariance matrix.

$$\epsilon = \begin{pmatrix} \hat{\sigma_1} - \sigma_1 \\ \hat{\sigma_2} - \sigma_2 \\ \hat{\sigma_3} - \sigma_3 \\ \hat{\sigma_4} - \sigma_4 \end{pmatrix} \tag{5.11}$$

and

$$\Lambda = E\{\epsilon\epsilon'\} \tag{5.12}$$

The following bound on the error covariance matrix exists [13].

$$\Lambda \geq J^{-1} \tag{5.13}$$

where the 4×4 matrix J, the Fisher information matrix, is given by

$$J \underline{\triangle} E(\left\{\nabla_\sigma \left[\ln p_{r|\sigma}(R|\sigma)\right]\right\} \left\{\nabla_\sigma \left[\ln p_{r|\sigma}(R|\sigma)\right]\right\}') \tag{5.14}$$

and the partial derivative operator, ∇_σ, is defined as

$$\nabla_\sigma \triangleq \begin{pmatrix} \frac{\partial}{\partial\sigma_1} \\ \frac{\partial}{\partial\sigma_2} \\ \frac{\partial}{\partial\sigma_3} \\ \frac{\partial}{\partial\sigma_4} \end{pmatrix} \tag{5.15}$$

This operator when applied to a $1 \times m$ row vector

$$a = \begin{pmatrix} a_1 & \dots & a_m \end{pmatrix} \tag{5.16}$$

yields

$$\nabla_\sigma a = \begin{pmatrix} \frac{\partial a_1}{\partial\sigma_1} & \dots & \frac{\partial a_m}{\partial\sigma_1} \\ \vdots & & \\ \frac{\partial a_1}{\partial\sigma_4} & \dots & \frac{\partial a_m}{\partial\sigma_4} \end{pmatrix} \tag{5.17}$$

From the observation equation, eq.(5.2), we can derive the log of the conditional density. Specifically, we have the following.

$$\begin{aligned} \ln p_{r|\sigma}(R|\sigma) &= -\frac{1}{2}\sum_{k=1}^{L}(r_k - \nabla_\sigma h_k(\sigma))' R_k^{-1}(r_k - \nabla_\sigma h_k(\sigma)) \\ &+ \alpha(R_k, L) \end{aligned} \tag{5.18}$$

where α is a normalization term which does not depend on σ. If we substitute eq.(5.18) into eq.(5.14) and use the fact that the noise vectors are uncorrelated from experiment to experiment in performing the expectation, we obtain

$$J = \left\{ \sum_{k=1}^{L} \nabla_\sigma h_k(\sigma) R_k^{-1} (\nabla_\sigma h_k(\sigma))' \right\} \tag{5.19}$$

We now focus on deriving expressions for $\nabla_\sigma h_k(\sigma)$. Taking the dervivative of $h_k(\sigma)$ with respect to σ we obtain the following.

$$\nabla_\sigma h_k(\sigma) = M_k' + S'(\nabla_\sigma \mathbf{z}_k')' \tag{5.20}$$

where

$$M_k = \begin{pmatrix} \begin{pmatrix} z_{1,4} \\ z_{1,1} \end{pmatrix}_k & 0 & 0 & 0 \\ 0 & \begin{pmatrix} z_{2,1} \\ z_{2,2} \end{pmatrix}_k & 0 & 0 \\ 0 & 0 & \begin{pmatrix} z_{3,3} \\ z_{3,4} \end{pmatrix}_k & 0 \\ 0 & 0 & 0 & \begin{pmatrix} z_{4,2} \\ z_{4,3} \end{pmatrix}_k \end{pmatrix} \tag{5.21}$$

(5.22)

Note with regard to the matrix M_k the subscript k signifies that the normal derivative vectors that make up the entries of M_k are defined for the kth experiment.

In eq.(5.20) the quantities M and Σ are well defined. The matrix M contains normal derivative vectors along the edge of the unit square for some background σ. The matrix Σ is made up of entries of this background σ. We do not, however, have an expression for the matrix $\nabla_\sigma \mathbf{z}'_k$, which must also be evaluated for some background σ. As in Chapter 3 we linearize the PDE with respect to σ in order to come up with an approximate solution based on a linearization about some particular background σ. This first order solution will in fact enable us to derive an expression for $\nabla_\sigma \mathbf{z}'_k$.

In the following discussion we focus on an individual experiment in order to derive an expression for $\nabla_\sigma \mathbf{z}'_k$. One must keep in mind that when computing the Fisher information matrix the quantity $\nabla_\sigma \mathbf{z}'_k$ must be computed for each experiment. Since our discussion focuses on quantities associated with an individual experiment, for ease of notation we drop the subscript k from these quantities.

The set-up is identical to that for the four square problem described in Chapter 3 except for the fact that we no longer assume a constant background throughout; i.e. in linearizing the problem we now allow perturbations in σ about arbitrary nominal values in each of the four squares. We define the following quantities for a particular set of boundary conditions.

$$\sigma_i = \sigma_i^0 + \delta\sigma_i \tag{5.23}$$

$$b_i = b_i^0 + \delta b_i \tag{5.24}$$

$$z_{i,j} = z_{i,j}^0 + \delta z_{i,j} \tag{5.25}$$

where $i,j = 1,...,4$. The scalars, σ_i, and the $N/2 \times 1$ vectors, b_i are defined as illustrated in Figure 5.3. The $N/2 \times 1$ vectors $z_{i,j}$, which are normal dervative vectors for the interior edges, are defined as illustrated in Figure 5.4.

Let us define the following 4×1 vectors.

$$\sigma^0 = \begin{pmatrix} \sigma_1^0 \\ \sigma_2^0 \\ \sigma_3^0 \\ \sigma_4^0 \end{pmatrix} \tag{5.26}$$

$$\delta\sigma = \begin{pmatrix} \delta\sigma_1 \\ \delta\sigma_2 \\ \delta\sigma_3 \\ \delta\sigma_4 \end{pmatrix} \tag{5.27}$$

(5.28)

Let us also define the following $2N \times 1$ vectors.

$$\mathbf{b} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{pmatrix} \tag{5.29}$$

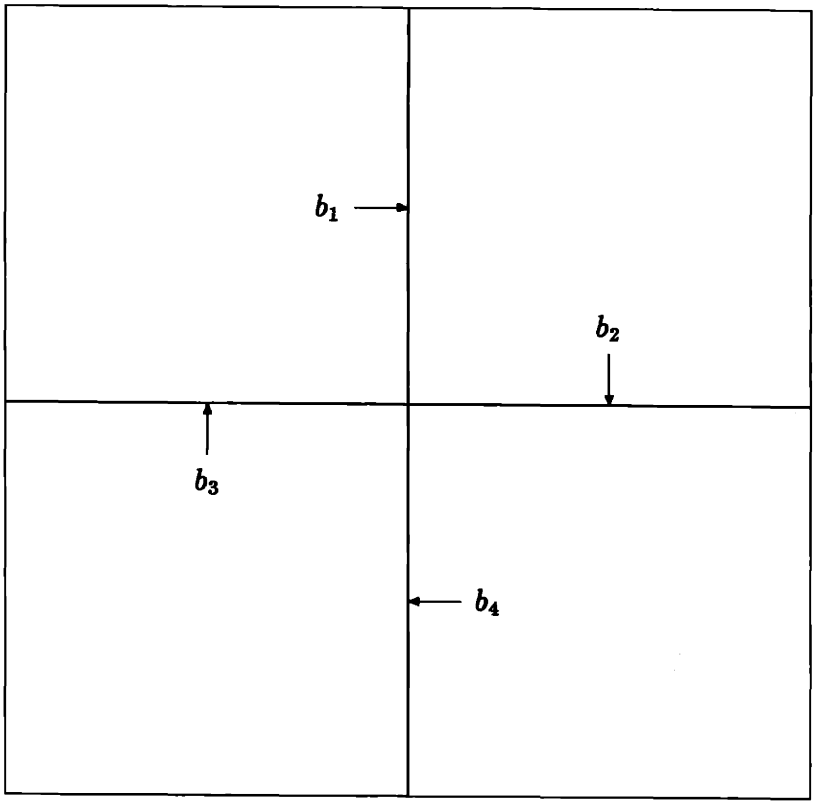

Figure 5.3: Interior Potential Vectors b_i Defined on the Unit Square

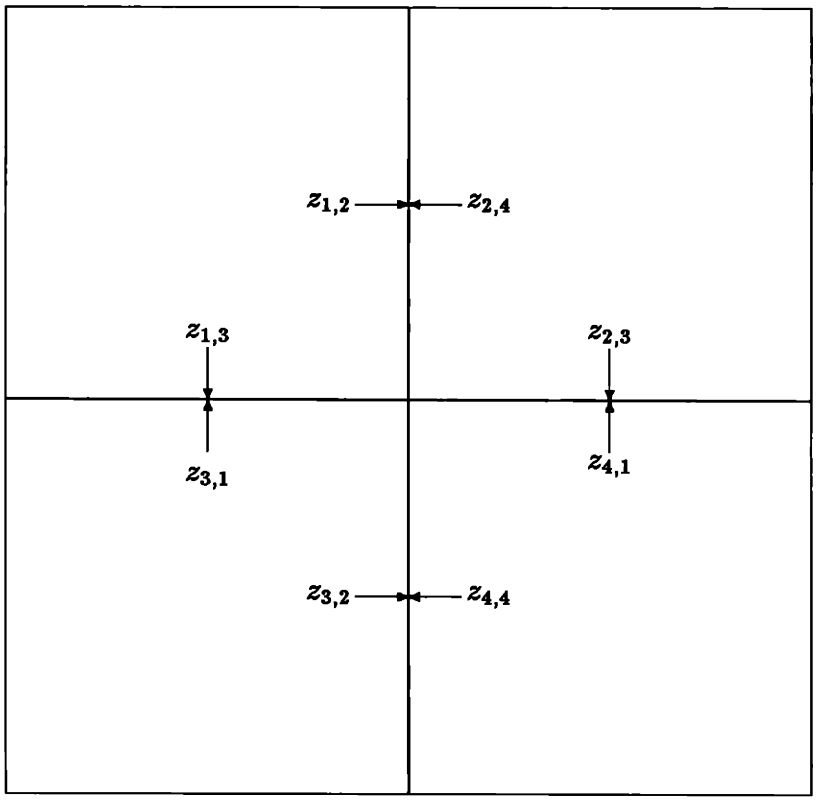

Figure 5.4: Interior Normal Derivative Vectors z_{ij} Defined on the Unit Square

$$\mathbf{b}^0 = \begin{pmatrix} b_1^0 \\ b_2^0 \\ b_3^0 \\ b_4^0 \end{pmatrix} \qquad (5.30)$$

$$\delta\mathbf{b} = \begin{pmatrix} \delta b_1 \\ \delta b_2 \\ \delta b_3 \\ \delta b_4 \end{pmatrix}$$

Recall that the Dirichlet to Neumann map derived in Chapter 2, eq.(4.16), gives a linear relationship between the potential and the normal derivative of the potential along the edges of each square. Therefore, there exists a linear relationship between $\mathbf{z}$, the normal derivative of the potential along the boundary of the entire unit square, and the potential vectors defined along the edges of each square.

The vector $\mathbf{b}$ contains the potential vectors along each of the interior edges. Let us also define the vector $\mathbf{d}$, which is to contain the potential vectors along each of the exterior boundary edges; i.e. the Dirichlet boundary conditions for an individual experiment.

$$\mathbf{d} = \begin{pmatrix} b_{11} \\ b_{21} \\ b_{22} \\ b_{33} \\ b_{34} \\ b_{42} \\ b_{43} \\ b_{14} \end{pmatrix} \qquad (5.31)$$

We can write the linear relationship between $\mathbf{z}$ and the two vectors $\mathbf{b}$ and $\mathbf{d}$ as

follows.

$$\mathbf{z} = T\mathbf{b} + T_d\mathbf{d} \tag{5.32}$$

where

$$T = \begin{pmatrix} S_o & 0 & 0 & H_{to} \\ \hat{H}_o & 0 & 0 & -S_o \\ \hat{H}_{to} & -S_o & 0 & 0 \\ -S_o & \hat{H}_{to} & 0 & 0 \\ 0 & 0 & H_o & S_o \\ 0 & 0 & S_o & H_o \\ 0 & \hat{H}_o & -S_o & 0 \\ 0 & S_o & H_{to} & 0 \end{pmatrix} \tag{5.33}$$

and

$$T_d = \begin{pmatrix} H_o & 0 & 0 & 0 & 0 & 0 & 0 & -D \\ D & 0 & 0 & 0 & 0 & 0 & 0 & \hat{H}_{to} \\ 0 & D & \hat{H}_o & 0 & 0 & 0 & 0 & 0 \\ 0 & \hat{H}_o & D & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \hat{H}_{to} & -S_o & 0 & 0 & 0 \\ 0 & 0 & 0 & -D & H_{to} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & H_o & -D & 0 \\ 0 & 0 & 0 & 0 & 0 & S_o & H_{to} & 0 \end{pmatrix} \tag{5.34}$$

Let us premultiply the transpose of each side of eq.(5.32) by the operator ∇_σ . Noting that $T_d\mathbf{d}$ does not depend on σ, we get the following relationship.

$$\nabla_\sigma \mathbf{z}' = \nabla_\sigma \mathbf{b}' T' \tag{5.35}$$

Now our problem of finding an expression for $\nabla_\sigma \mathbf{z}'$ amounts to finding an expression for $\nabla_\sigma \mathbf{b}'$.

Recall that the PDE is such that the interior potential, the vector $\mathbf{b}$, is a non-linear function of the vector σ. We have defined $\mathbf{b} = \mathbf{b}^0 + \delta\mathbf{b}$ such that $\delta\mathbf{b}$ is a perturbation of $\mathbf{b}$ about the nominal point $\mathbf{b}^0$. For small perturbations is a good

approximation to the linear or first order term of the Taylor Series expansion of the vector $\mathbf{b}$ with respect to σ. Specifically,

$$\delta \mathbf{b} \cong \{\nabla_\sigma \mathbf{b}'\}' \mid_{\sigma_0} \delta\sigma \tag{5.36}$$

In order to find an expression for $\nabla_\sigma \mathbf{b}'$ we linearize the PDE so that there is a linear relationship between $\delta\mathbf{b}$ and $\delta\sigma$ then identify this approximate relationship as $\nabla_\sigma \mathbf{b}'$.

We proceed to linearize the PDE. Recall that solving the PDE amounts to satisfiying the conditions in eq.'s(5.6)-(5.9). If we substitute eq.'s(5.23)-(5.25) into eq.'s(5.6)-(5.9), we obtain the following:

$$(\sigma_1^0 + \delta\sigma_1)(z_{1,2}^0 + \delta z_{1,2}) = (\sigma_2^0 + \delta\sigma_2)(z_{2,4}^0 + \delta z_{2,4}) \tag{5.37}$$

$$(\sigma_2^0 + \delta\sigma_2)(z_{2,3}^0 + \delta z_{2,3}) = (\sigma_4^0 + \delta\sigma_4)(z_{4,1}^0 + \delta z_{4,1}) \tag{5.38}$$

$$(\sigma_3^0 + \delta\sigma_3)(z_{3,1}^0 + \delta z_{3,1}) = (\sigma_1^0 + \delta\sigma_1)(z_{1,3}^0 + \delta z_{1,3}) \tag{5.39}$$

$$(\sigma_4^0 + \delta\sigma_4)(z_{4,4}^0 + \delta z_{4,4}) = (\sigma_3^0 + \delta\sigma_3)(z_{3,2}^0 + \delta z_{3,2}) \tag{5.40}$$

If we multiply out the terms in eq.'s(5.37)-(5.40) and keep only the linear terms, we arrive at the following set of equations relating the first order perturbations of the interior normal derivative vectors to the first order perturbations of the conductivity.

$$\sigma_1^0 \delta z_{1,2} - \sigma_2^0 \delta z_{2,4} = \delta\sigma_2 z_{2,4}^0 - \delta\sigma_1 z_{1,2}^0 \tag{5.41}$$

$$\sigma_2^0 \delta z_{2,3} - \sigma_4^0 \delta z_{4,1} = \delta\sigma_4 z_{4,1}^0 - \delta\sigma_2 z_{2,3}^0 \tag{5.42}$$

$$\sigma_3^0 \delta z_{3,1} - \sigma_1^0 \delta z_{1,3} = \delta\sigma_1 z_{1,3}^0 - \delta\sigma_3 z_{3,1}^0 \tag{5.43}$$

$$\sigma_4^0 \delta z_{4,4} - \sigma_3^0 \delta z_{3,2} = \delta\sigma_3 z_{3,2}^0 - \delta\sigma_4 z_{4,4}^0 \tag{5.44}$$

We can use our Dirichlet to Neumann map, eq.(4.16), to write the first order perturbations of the normal derivative vectors in eq.'s(5.41)-(5.44) as functions of first order perturbations of the interior potential, δb_k. Specifically, we arrive at the following linearized equations.

$$F_{\sigma^0}\delta\mathbf{b} = G^0\delta\sigma \tag{5.45}$$

where

$$F_{\sigma^0} = \begin{pmatrix} (\sigma_1^0+\sigma_2^0)D & -\sigma_2^0 H_{to} & \sigma_1^0\hat{H}_{to} & 0 \\ \sigma_2^0 H_{to} & -(\sigma_2^0+\sigma_4^0)D & 0 & -\sigma_4^0\hat{H}_{to} \\ -\sigma_1^0 H_o & 0 & (\sigma_1^0+\sigma_3^0)D & \sigma_3^0\hat{H}_o \\ 0 & \sigma_4^0 H_o & -\sigma_3^0\hat{H}_o & -(\sigma_3^0+\sigma_4^0)D \end{pmatrix} \tag{5.46}$$

$$G^0 = \begin{pmatrix} -z_{1,2}^0 & z_{2,4}^0 & 0 & 0 \\ 0 & -z_{2,3}^0 & 0 & z_{4,1}^0 \\ z_{1,3}^0 & 0 & -z_{3,1}^0 & 0 \\ 0 & 0 & z_{3,2}^0 & -z_{4,4}^0 \end{pmatrix} \tag{5.47}$$

For invertible F_{σ^0} we can solve for $\delta\mathbf{b}$.

$$\delta\mathbf{b} = (F_{\sigma^0})^{-1}G^0\delta\sigma \tag{5.48}$$

We have a linear relationship between $\delta\mathbf{b}$ and $\delta\sigma$ and from eq.(5.36) we can deduce the following.

$$\{\nabla_\sigma \mathbf{b}'\}' \mid_{\sigma_0} = F_{\sigma^0}^{-1}G^0 \tag{5.49}$$

Finally, we can derive our expression for $\nabla_\sigma z_k'$. Substituting eq.(5.49) into eq.(5.35) we get,

$$\nabla_\sigma \mathbf{z}_k' \mid_{\sigma_0} = (G^0)'(F_{\sigma^0}^{-1})' \mid_{\sigma_0} T' \tag{5.50}$$

Note that the matrices F_{σ^0} and T are constant matrices independent of the experiment. However, as one can see from eq.(5.47) the matrix G^0 contains normal

derivative vectors which must be computed for each experiment. Therefore, for notational purposes we let $G^0 = G_k$ so that

$$\nabla_\sigma \mathbf{z}_k' \mid_{\sigma_0} = G_k'(F_{\sigma^0}^{-1})' \mid_{\sigma_0} T' \tag{5.51}$$

In summary, we can compute the Cramer-Rao bound by taking the inverse of the Fisher information matrix. We can compute the Fisher information matrix, J, in the following way. We first substitute eq.(5.51) into eq.(5.20) in order to compute the matrix $\nabla_\sigma h_k(\sigma)$ for a particular experiment. We then substitute the matrix $\nabla_\sigma h_k(\sigma)$ into eq.(5.19) for each experiment and perform the specified summation over all the experiments. Note that the bound is computed for a specific set of nominal values $\sigma_1^0, ..., \sigma_4^0$.

5.4 Numerical Computation of Bound for Different Backgrounds and Excitations

A measure of estimation performance for an unbiased estimate of σ is the error covariance matrix Λ. On the basis of the Cramer-Rao bound, eq.(5.13), we can expect the inverse Fisher information matrix J^{-1} to give us a reasonable indication of Λ.

In this section we numerically compute the matrix J^{-1} using the expressions derived in the previous section. In our computations we take the unit square to be a 32×32 grid. We use a five-point finite difference scheme to solve the PDE for specific σ_0's in order to compute the necessary normal derivative vectors required in forming the matrices M_k and G_k. In computing J^{-1}, using the expression for J, eq.(5.19), we need a value for the error variance of the noise vector for each

experiment, R_k. We assume $R_k = \mu I$; i.e. the noise is white with variance μ for each experiment. Our definition for the SNR is as defined in Chapter 3; i.e. it is the ratio of the root mean-square of the signal to the standard deviation of the noise. Specifically,

$$\text{SNR} = \left\{ \left[\frac{1}{L} \sum_{k=1}^{L} (h_k(\sigma) h_k'(\sigma))/4N \right] / \mu \right\}^{\frac{1}{2}} \tag{5.52}$$

where L equals the number of experiments.

A useful statistic in indicating how well correlated two random variables are is the normalized correlation between the two or *correlation coefficient.* The correlation coefficient η_{ij} between two random variables, x_i and x_j, is defined as follows.

$$\eta_{ij} = E\{x_i x_j\}/(E\{x_i^2\}E\{x_j^2\})^{\frac{1}{2}} \tag{5.53}$$

where

$$|\eta_{ij}| \leq 1 \tag{5.54}$$

Based on the correlation coefficient we define a normalized J^{-1}, P. The entries of P are defined as follows. For $i, j = 1, ...4$,

$$P_{ij} = J_{ij}^{-1}/(J_{ii}^{-1} J_{jj}^{-1})^{\frac{1}{2}} \tag{5.55}$$

We compute J^{-1} and P for various σ^0 backgrounds, i.e. for various values of $\sigma_1^0, ...\sigma_4^0$, in order to investigate how these backgrounds affect estimation performance. We also compute the matrix for various boundary excitations to illustrate the effects of boundary excitations on performance. Finally, by observing the magnitudes of the correlations between the errors in estimating each square(the off diagonal terms of the inverse Fisher Information matrix) we get an idea of how correlated the error in one square is with that in another. This is of particular interest

in the context of our multi-resolution estimation approach in that low correlation among the performances in estimating each square would justify an approach that tries to break the problem down by subdividing the domain and trying to extract features within each domain.

5.4.1 J^{-1} for Various Excitations

We first compute J^{-1} for the sinusoidal boundary conditions,

$$\phi(x,y) = \begin{cases} \sin(2\pi x) & y = 1, \quad 0 \le x \le 1 \\ -\sin(2\pi y) & 0 \le y \le 1, \quad x = 1 \\ -\sin(2\pi x) & y = 0, \quad 0 \le x \le 1 \\ -\sin(2\pi y) & 0 \le y \le 1, \quad x = 0 \end{cases} \tag{5.56}$$

The following is J^{-1} for a constant background, $\sigma_1^0 = \sigma_2^0 = \sigma_3^0 = \sigma_4^0 = 1$, and SNR $= 1$.

$$\begin{pmatrix} 3.213024e-02 & -4.489673e-04 & -4.489673e-04 & 1.769225e-05 \\ -4.489673e-04 & 3.213024e-02 & 1.769223e-05 & -4.489673e-04 \\ -4.489673e-04 & 1.769223e-05 & 3.213024e-02 & -4.489673e-04 \\ 1.769225e-05 & -4.489673e-04 & -4.489673e-04 & 3.213024e-02 \end{pmatrix} \tag{5.57}$$

The following is the corresponding matrix of correlation coefficients, P.

$$\begin{pmatrix} 1 & -1.397336e-02 & -1.397336e-02 & 5.506416e-04 \\ -1.397336e-02 & 1 & 5.50641e-04 & -1.397336e-02 \\ -1.397336e-02 & 5.50641e-04 & 1 & -1.397336e-02 \\ 5.506416e-04 & -1.397336e-02 & -1.397336e-02 & 1 \end{pmatrix} \tag{5.58}$$

As we would expect the correlation between the errors in squares one and four is less than that of squares one and two due to the proximities of the squares; i.e. $J_{12}^{-1} \le J_{14}^{-1}$. The symmetry of this particular excitation scheme is exhibited by the

equivalence of the correlations between adjacent squares, i.e. $J_{12}^{-1} = J_{13}^{-1} = J_{42}^{-1} = J_{43}^{-1}$, and diagonally opposing squares, i.e. $J_{14}^{-1} = J_{23}^{-1}$, and the fact that J_{ii} is independent of i. Note also that the magnitudes of the offdiagonal terms of P are quite small indicating very low correlation of errors between squares.

Next we compute J^{-1} for a boundary condition consisting of an impulse in the upper left corner.

$$\phi(x,y) = \begin{cases} \delta(x) & y = 1 \\ 0 & elsewhere \end{cases} \tag{5.59}$$

The following is J^{-1} for a constant background, $\sigma_1^0 = \sigma_2^0 = \sigma_3^0 = \sigma_4^0 = 1$, and SNR = 1.

$$\begin{pmatrix} 7.813642e-03 & 0.15047 & 0.15047 & 0.26502 \\ 0.15047 & 38631.4 & 265.971 & 12691.5 \\ 0.15047 & 265.971 & 38631.4 & 12691.5 \\ 0.26502 & 12691.5 & 12691.5 & 2.897401e+05 \end{pmatrix} \tag{5.60}$$

The following is the corresponding matrix of correlation coefficients.

$$\begin{pmatrix} 1 & 8.660616e-03 & 8.660616e-03 & 5.569841e-03 \\ 8.660616e-03 & 1 & 6.884837e-03 & 0.11996 \\ 8.660616e-03 & 6.884837e-03 & 1 & 0.11996 \\ 5.569841e-03 & 0.11996 & 0.11996 & 1 \end{pmatrix} \tag{5.61}$$

Notice $J_{11}^{-1} \leq J_{22}^{-1} \leq J_{44}^{-1}$. This indicates that the performance degrades as one gets further away from the impulse. This is reasonable since the energy from the impulse dies away quite quickly; i.e. for a constant background where the PDE is simply Laplace's Equation, we have an analytical solution, eq.(4.1), from which it is easy see that the potential decreases exponentially away from the source.

Once again, the magnitudes of the offdiagonal terms of P are quite small indicating very low correlation of errors between squares.

Finally, we compute J^{-1} for a multiple experiment scheme consisting of 16 independent impulses symmetrically distributed about the boundary. We define $\phi_k(x,y)$, the boundary condition for the *kth* experiment, as

$$\phi_k(x,y) = \begin{cases} \delta(x - \frac{(k-1)}{4}) & y = 1 \quad 1 \le k \le 4 \\ \delta(y - \frac{(k-5)}{4}) & x = 1 \quad 5 \le k \le 8 \\ \delta(x - \frac{(k-9)}{4}) & y = 0 \quad 9 \le k \le 12 \\ \delta(y - \frac{(k-13)}{4}) & x = 0 \quad 13 \le k \le 16 \end{cases} \tag{5.62}$$

The following is J^{-1} for a constant background, $\sigma_1^0 = \sigma_2^0 = \sigma_3^0 = \sigma_4^0 = 1$, and SNR = 1.

$$\begin{pmatrix} 2.023046e-03 & -3.557312e-05 & -3.557306e-05 & 1.224941e-06 \\ -3.557312e-05 & 2.023046e-03 & 1.22494e-06 & -3.557306e-05 \\ -3.557306e-05 & 1.22494e-06 & 2.023046e-03 & -3.557312e-05 \\ 1.224941e-06 & -3.557306e-05 & -3.557312e-05 & 2.023046e-03 \end{pmatrix} \tag{5.63}$$

The following is the corresponding matrix of correlation coefficients.

$$\begin{pmatrix} 1 & -1.758394e-02 & -1.758391e-02 & 6.054933e-04 \\ -1.758394e-02 & 1 & 6.054928e-04 & -1.758391e-02 \\ -1.758391e-02 & 6.054928e-04 & 1 & -1.758394e-02 \\ 6.054933e-04 & -1.758391e-02 & -1.758394e-02 & 1 \end{pmatrix} \tag{5.64}$$

We see from the diagonal entries of J^{-1} that performance is better than that of either of the previous two single experiment schemes. This comes as no surprise since we are using 16 times as many measurements. It is interesting to compare the values of the diagonals of J^{-1}, which are all equivalent for excitations symmetric about the boundary, for the two symmetric cases; i.e. sixteen measurement case and the single measurement sinusoidal excitation case. The ratio of the error variance in the sixteen measurement case to the error variance in the single measurement case is roughly 1/16.

As in the case of the two single experiment schemes, the magnitudes of the offdiagonal terms of P are quite small indicating very low correlation of errors between squares.

5.4.2 Effect of Background σ on J^{-1}

We wish to characterize J^{-1} as a function of the conductivity values about which we are linearizing, i.e. $\sigma_1^0, ..., \sigma_4^0$, by considering two different situations. In the first situation we consider the case of a background in which $\sigma_i^0 = \sigma_0 + \delta\sigma_i$ for $i = 1, ..., 4$ where σ_0 is the same for all four squares and the $\delta\sigma_i$'s are independent identically distributed perturbations. We plot the square-root of the average mean-square error, i.e. $\frac{1}{4}\sum_{i=1}^{4} J_{ii}^{-1}$, as a function of the magnitude of the standard deviation of $\delta\sigma_i$. In the second situation we consider the case in which the value of σ in three of the four squares are equal while the fourth square is randomly perturbed. We plot the square-root of the average mean-square error as a function of the standard deviation of this perturbation. In both cases we use the multiple experiment scheme consisting of the 16 independent impulses symmetrically distributed about the boundary of the unit square.

We begin with the constant background case. Consider the case where $\sigma_i = \sigma_0 + \delta\sigma_i$ where the $\delta\sigma_i$'s are zero-mean, jointly Gaussian and

$$E\{\delta\sigma_i \delta\sigma_j\} = q\delta_{ij} \tag{5.65}$$

To get an idea of the average performance in each square we define the following quantity.

$$\tau = \left\{ E_\sigma \left[\frac{1}{4} \sum_{i=1}^{4} J_{ii}^{-1} \right] \right\}^{\frac{1}{2}} \tag{5.66}$$

We can use Monte-Carlo trials in order to plot τ/σ_0 as a function of the distortion in σ, ρ/σ_0, where

$$\rho = \sqrt{q} \tag{5.67}$$

The result is shown in Figure 5.5.

The plot in Figure 5.5 indicates that the average performance in a square is a monotonically increasing function of the distortion in the background. This indicates that the errors in estimating each square get larger as the variation in the background increases. However, the range of the errors, .0045 - .00495, compared with the range of the distortion in the conductivity background, 0.0 - 0.6, indicates that the bound is fairly insensitive to the values of σ in each square.

Let us also look at the correlation coefficients as a function of the distortion for this case. We can estimate the correlation coefficients from looking at the matrix P as defined in eq.(5.55). The symmetries in the boundary conditions give rise to a highly structured P in that the correlations in any two adjacent squares are equivalent and likewise for any two diagonally opposing squares. Therefore, we need only look at $P_{1,2}$ and $P_{1,4}$.

Figure 5.6 is a plot of $P_{1,2}$, the correlation between the errors in squares one and two, as a function of the distortion in σ, ρ/σ_0, generated using Monte-Carlo trials. The plot indicates that the errors in adjacent squares are essentially uncorrelated. However, the correlation, in this case it is negative, increases as the variation in the background increases.

Figure 5.7 is a plot of $P_{1,4}$ as a function of the distortion in σ, ρ/σ_0, generated using Monte-Carlo trials. We can see that correlation between diagonally opposing squares is two orders of magnitude less than that of adjacent squares. The

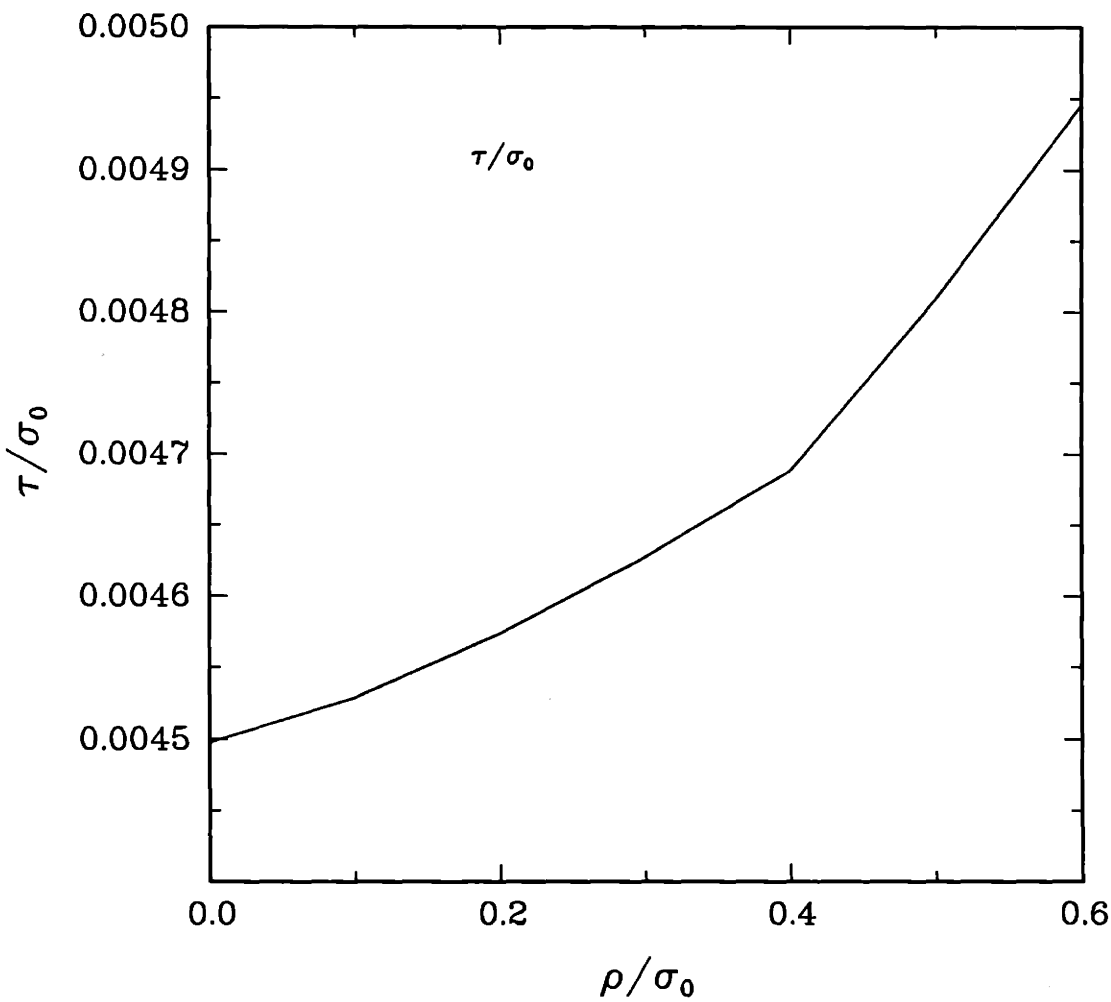

Figure 5.5: τ/σ_0 vs. ρ/σ_0 for 100 trials, SNR=10

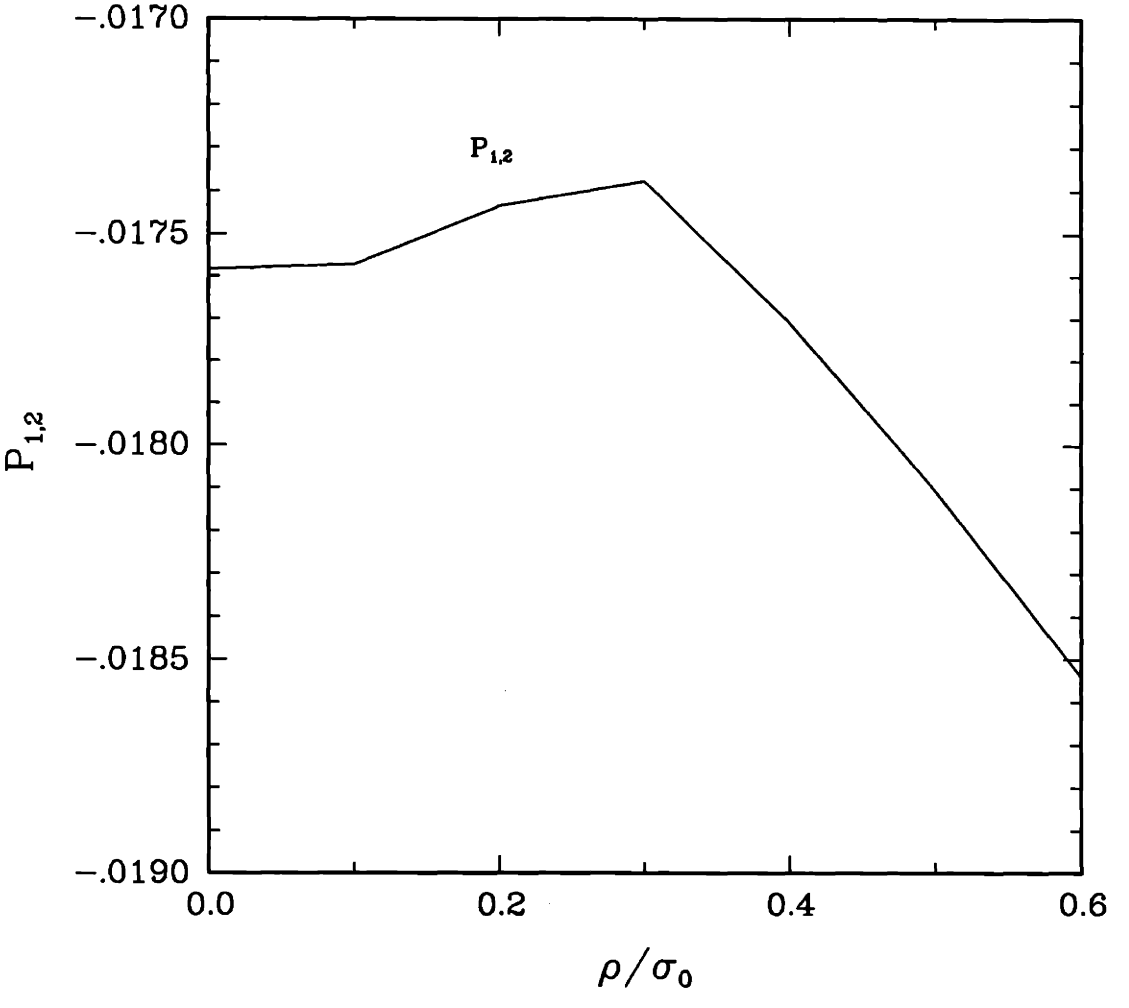

Figure 5.6: $P_{1,2}$ vs. ρ/σ_0 for 100 trials, SNR=10

correlation decreases as a function of the variation in the background.

We now consider the case where $\sigma(r)$ is constant in three of the four squares and free to vary in the fourth square. Consider the case of $\sigma_1 = \sigma_2 = \sigma_3 = \sigma_0$ and $\sigma_4 = \sigma_0 + \delta\sigma$ where $\delta\sigma$ is a zero-mean, Gaussian random variable with

$$E\left\{\delta\sigma^2\right\} = q \tag{5.68}$$

Let us once again investigate the average performance in each square, as defined in eq.(5.66), as a function of the distortion in σ_4. We use Monte-Carlo trials in order to plot τ/σ_0 as a function of the distortion in σ_4, ρ/σ_0, where once again $\rho = \sqrt{q}$. The result is shown in Figure 5.8.

The range of magnitudes of the mean-square errors in Figure 5.8 is roughly the same as the range in Figure 5.5. This further indicates that the Cramer-Rao bound is insensitive to the values of σ in each square.

Let us now look at the correlation coefficients as a function of the distortion in square four. Figure 5.9 is a plot of $P_{1,2}$ as a function of the distortion in σ, ρ/σ_0, generated using Monte-Carlo trials. Figure 5.10 is a plot of $P_{1,4}$ as a function of the distortion in σ, ρ/σ_0, generated using Monte-Carlo trials. As in the case where we added random perturbations to all four squares, the values of $P_{1,2}$ and $P_{1,4}$ range over very small values. This gives further indication that the errors in estimating each square are essentially uncorrelated with each other.

5.4.3 Conclusions

Our numerical computations have demonstrated both the usefulness and the versatility of the Cramer-Rao bound. We were able to explore quantitatively the effects of excitations on estimation performance. We were also able to explore the depen-

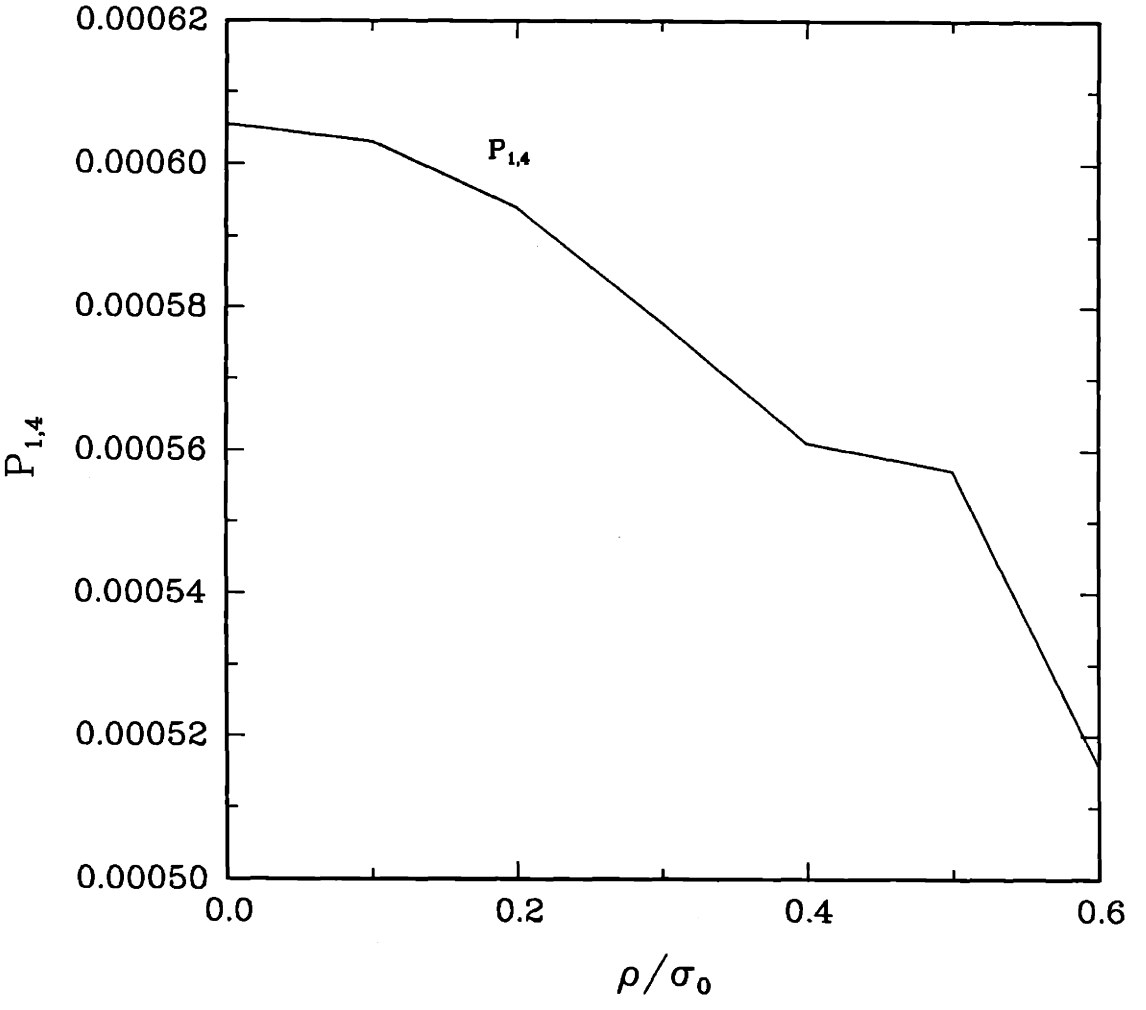

Figure 5.7: $P_{1,4}$ vs. ρ/σ_0 for 100 trials, SNR=10

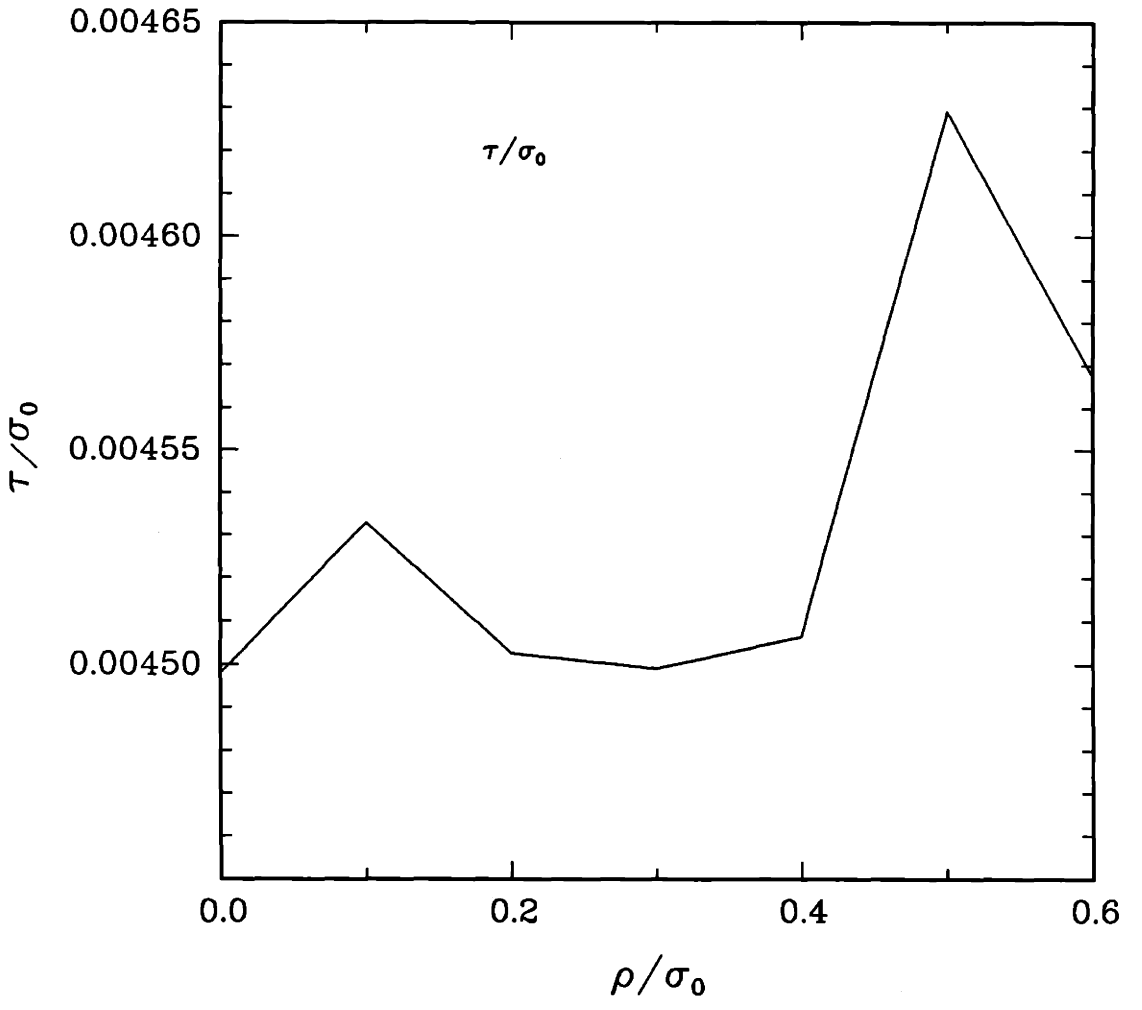

Figure 5.8: τ/σ_0 vs. ρ/σ_0 for 100 trials, SNR=10

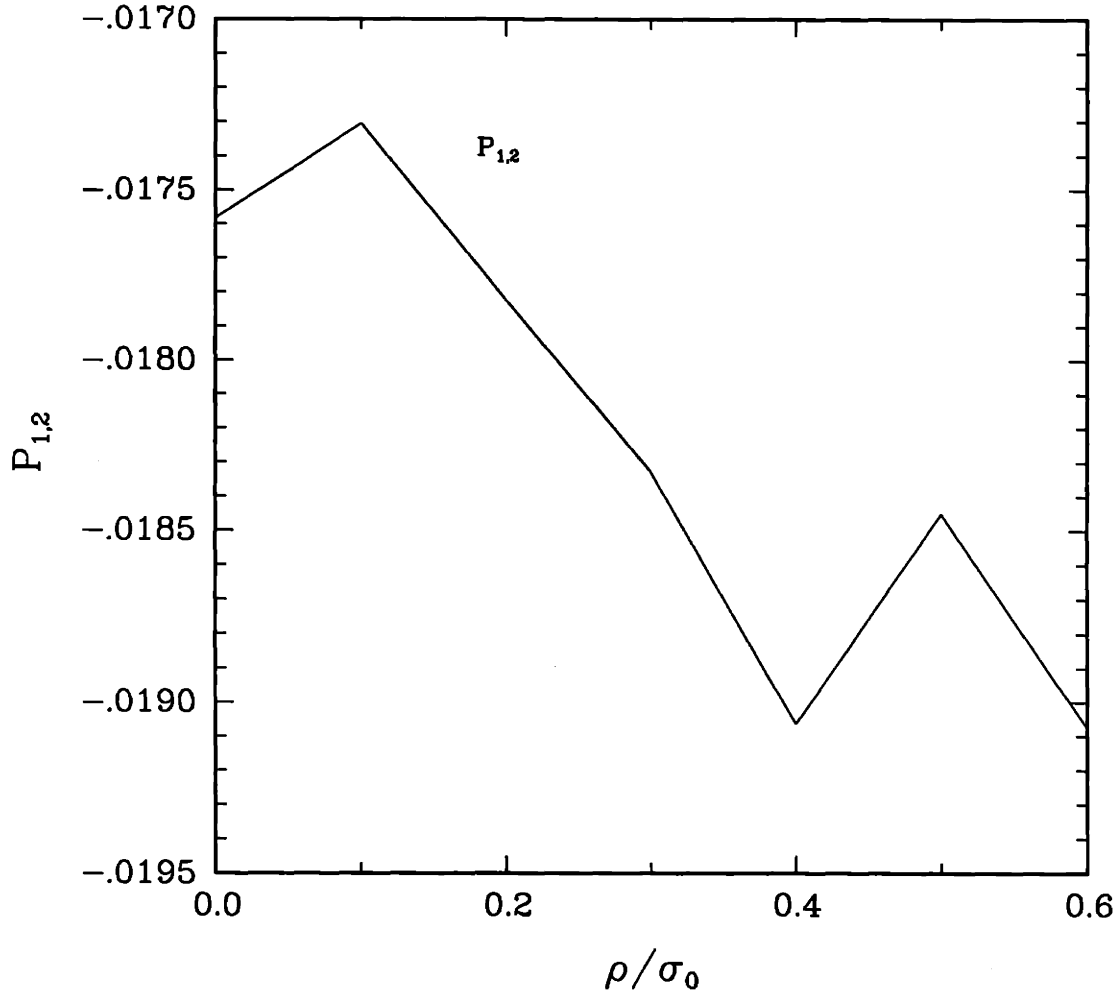

Figure 5.9: $P_{1,2}$ vs. ρ/σ_0 for 100 trials, SNR=10

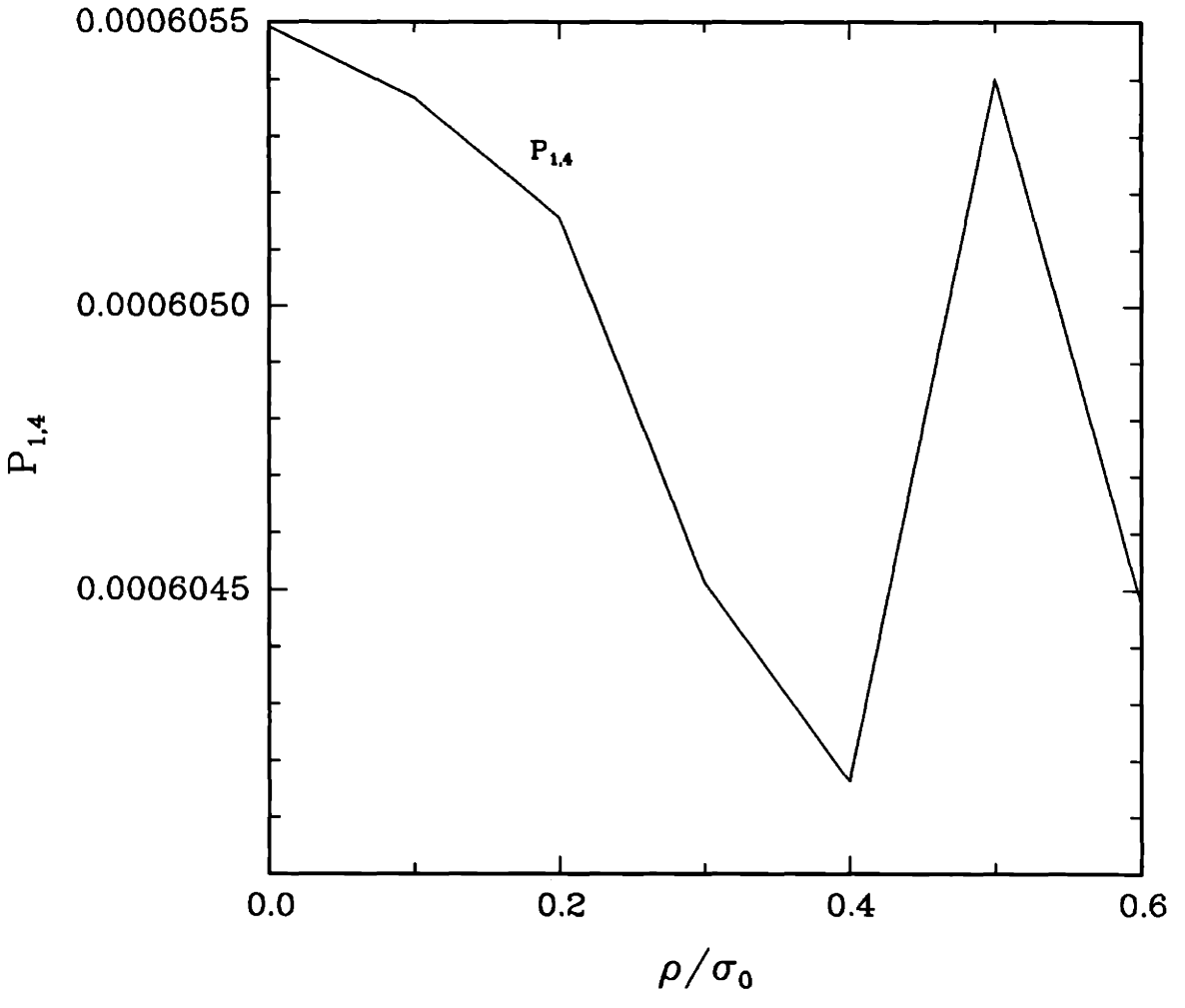

Figure 5.10: $P_{1,4}$ vs. ρ/σ_0 for 100 trials, SNR=10

dency of estimation performance on the particular values of the parameters we are estimating. In fact our plots indicate that the dependency of the error variance on the background conductivity is relatively small. We also produced plots that indicate that the errors in estimating each square are approximately uncorrelated.

As a final note on our work with the Cramer-Rao bound, we point out that this method can be easily extended to cases of higher pixel resolution. Though the numerical computations would of course become more intensive, the general method we have outlined is valid for determining the bound on the error variance in estimating σ at any scale. This is useful in addressing the question of what spatial resolution of σ the data supports.

5.5 Numerical Investigation of Algorithm Performance

In this section we demonstrate the performance of our algorithm on synthetic data for the four-square case. Again we assume white measurement noise along the boundary with variance μ uncorrelated between experiments. We use eq.(5.52) for our definition of SNR. We use a 16 by 16 point discretization. Synthetic data was generated using a five-point finite difference scheme to solve the forward problem. We use an excitation scheme consisting of 16 independent impulse excitations symmetric about the boundary as described in eq.(5.62). The following four square conductivity image is used.

45.8927	91.1246
103.463	100

Table 5.1: True $\sigma_1, ..., \sigma_4$

The following is a table of the percentage estimation errors for each of the four squares after a given number of iterations of the algorithm. There are four cases ordered in decreasing values of SNR. An initial value of $\sigma = 100$ is used in each square to start the iteration at this scale. The value of $\frac{\gamma}{\lambda}$ is fixed to be 1. The second version of the two single scale algorithms is used; i.e. Algorithm 2 in Chapter 2.

iteration	5
4.68089e-05	-6.43654e-05
-7.20061e-05	-3.941219e-05
iteration	10
2.642314e-05	-3.023373e-05
-4.06811e-05	-3.317543e-05
iteration	20
2.296145e-05	-1.180416e-05
-1.810873e-05	-1.369082e-05
iteration	30
1.541231e-05	-5.003927e-06
-8.675999e-06	-6.134511e-06
iteration	50
5.528373e-06	-1.129088e-06
-2.318928e-06	-1.503834e-06

Table 5.2: Noiseless Data

From Tables 5.2-5.5 we see that the algorithm converges very quickly for this four square case. In each table we can see that near convergence is achieved after only 5 iterations. The algorithm demonstrates good performance even for an SNR as low as 2. The success of the algorithm at this scale is not surprising considering each square touches the external boundary, on which we have measurements. The problem becomes much more difficult when we introduce internal squares that do not have direct contact with measurements as we shall see in the next chapter.

iteration	5
3.703576e-03	2.721963e-03
3.39146e-04	3.622481e-03
iteration	10
3.684357e-03	2.776592e-03
3.780045e-04	3.624302e-03
iteration	20
3.674152e-03	2.815012e-03
4.051386e-04	3.632788e-03
iteration	30
3.658201e-03	2.830112e-03
4.144929e-04	3.63143e-03
iteration	50
3.638269e-03	2.839538e-03
4.192376e-04	3.626659e-03

Table 5.3: SNR = 10

iteration	5
7.356638e-03	5.510119e-03
7.358055e-04	7.325821e-03
iteration	10
7.351087e-03	5.58565e-03
7.915342e-04	7.323798e-03
iteration	20
7.340818e-03	5.637447e-03
8.290836e-04	7.315819e-03
iteration	30
7.314838e-03	5.653806e-03
8.38593e-04	7.299293e-03
iteration	50
7.280291e-03	5.660676e-03
8.401449e-04	7.277789e-03

Table 5.4: SNR = 5

iteration	5
1.830088e-02	1.388103e-02
1.831383e-03	1.868888e-02
iteration	10
1.840937e-02	1.402192e-02
1.993039e-03	1.867902e-02
iteration	20
1.843799e-02	1.407471e-02
2.096677e-03	1.858983e-02
iteration	30
1.837196e-02	1.405326e-02
2.108236e-03	1.849114e-02
iteration	50
1.826622e-02	1.400461e-02
2.091093e-03	1.837635e-02

Table 5.5: SNR = 2

Chapter 6

Estimating Sixteen Squares

6.1 Introduction

In this chapter we demonstrate the performance of our algorithm on synthetic data generated using a sixteen-square parametrization for σ. The sixteen-square case provides a good test for the algorithm because it is a case in which there are four interior squares whose edges do not coincide with the external boundary. Since these interior squares are not in direct contact with the measurements, one would expect the estimation performance in these squares to be inferior to that of squares that are in direct contact with the measurements. We would also expect the ill-posedness of the problem to manifest itself in the existence of local minima of the cost function. We numerically explore the following issues.

- 1) A comparison of convergence properties of the two versions of the algorithm, Algorithm 1 and Algorithm 2.
- 2) We investigate the presence of local minima by plotting the cost function, evaluated using noiseless data, along certain trajectories of the parameter space.

- 3) The influence of coarse scale information on performance at a finer scale.
- 4) Performance of the overall algorithm at 3 separate scales using both noiseless and noisy data. For the case of noisy data we let the penalty weighting coefficient λ increase gradually during the iterations at the sixteen-square scale and investigate the performance characteristics. We also give an example in which the noise level is extremely high in order to illustrate the resulting degradation in estimation performance.
- 5) The effects of noise on the performance of the algorithm as a function of location in the σ image. In particular we compare the performance of estimating the inner squares to that of estimating the outer squares. We also plot the performance for the problem of estimating the inner four squares as one square, i.e. estimating the mean of the four squares, and compare this with both the performance of estimating the inner squares and that of estimating the outer squares.

In the experiments to follow we use the same set-up as in Chapter 5; i.e. white measurement noise along the boundary, uncorrelated between experiments. A five-point finite difference scheme is used to solve the forward problem in order to generate synthetic data. A 16 by 16 point discretization is used for both generating the synthetic data and implementing the estimation algorithm. We use the excitation scheme consisting of 16 independent impulse excitations symmetric about the boundary as described in eq.(5.62). In all experiments involving noiseless data the value of $\frac{\gamma}{\lambda}$ is fixed to be 1.

6.2 Comparison of Performances of Algorithms 1 and 2

In this section we compare the rates of convergence of the two versions of the algorithm, Algorithm 1 and Algorithm 2, on noiseless data. Because we are simply interested in the convergence properties at a specific scale we do not run the full multi-scale algorithm. Instead, we run the algorithms exclusively at the sixteen-square scale. In order to initialize the iterations at this scale for both Algorithms 1 and 2 we use as the starting point for σ the average of the true σ at a four-square scale.

Algorithm 1 essentially alternates between solving two subproblems. One problem is that of finding the optimal σ conditioned on knowing ϕ; the other is the problem of finding the optimal ϕ conditioned on knowing σ. If we think in terms of the cost function, the algorithm alternates between going to the minimum of the cost function conditioned on ϕ and going to the minimum of the cost function conditioned on σ.

Algorithm 2 on the other hand alternates between single relaxation sweeps with respect to σ and ϕ. It alternates between making moves downward along the cost function in the direction of σ and making moves downward in the direction of ϕ. Another way of distinguishing between Algorithm 2 from Algorithm 1 is that instead of solving each subproblem fully, as Algorithm 1 does, Algorithm 2 solves each subproblem *partially.*

We compare the two algorithms by applying both of them on data corresponding to the conductivity image in Table 6.1. For each algorithm we iterate until the average percentage error of the inner squares reaches a certain certain level[1]. Since the total number of iterations required to attain a certain percentage error gives

[1] We use the performance of the inner squares as our criterion since the errors there dominate the errors of the outer squares.

a good indication of the performance of an algorithm, we compare the number of iterations required for each algorithm to attain a certain percentage error. We do this for several values of percentage error in order to compare their performances over a range of errors.

56.7142	92.8996	102.77	100
117.513	68.5311	110.48	133
140.345	64.9578	122.151	86.9013
134.194	106.852	100	135.18

Table 6.1: True σ

For the case of Algorithm 2 a single iteration is well defined; i.e. an iteration consists of one sweep with respect to σ and one sweep with respect to ϕ. For the case of Algorithm 1 a single iteration consists of two sub-iterations, one for σ and one for ϕ. The number of iterations involved in each of these sub-iterations is the number required for convergence in each case. We first define convergence for the two sub-iterations.

With respect to σ we take adequate convergence to be the point in the iteration at which the percentage change of σ in the inner squares falls below a certain threshold. With respect to ϕ we take adequate convergence to be the point in the iteration at which the percentage change in ϕ along the edges of the inner squares falls below a certain threshold. In our examples we take the value of the threshold to be .0001 for both cases.

For a given percentage error criterion we compare overall performance by looking at two different measures: a) the total number of σ iterations performed in Algorithms 1 and 2 b) the total number of ϕ iterations performed in Algorithms 1 and 2. Figure 6.1 is a log-log plot of the total number of σ iterations performed for the

following 4 percentage error criteria: .5, .05, .005, .0005[2]. Figure 6.1 is a log-log plot of the total number of ϕ iterations performed for the same set of percentage error criteria.

Note that in both Figure 6.1 and Figure 6.2 Algorithm 2 shows better performance characteristics than Algorithm 1 does, i.e. the total number of iterations is fewer, down to a certain level of percentage error. These plots indicate that for percentage errors down to approximately .001 Algorithm 2 performs better than Algorithm 1. For percentage errors lower than this, however, Algorithm 1 performs better than Algorithm 2, but the difference in performance decreases considerably.

These results indicate that down to a reasonably small percentage error criterion it is not productive to spend too many iterations on minimizing the cost function with respect to either σ conditioned on ϕ or ϕ conditioned on σ. Rather, faster convergence is attained by alternating between single relaxation sweeps with respect to σ and ϕ. Below a certain percentage error the difference becomes neglibible. Note that for noisy data we would not expect to achieve very low percentage errors. Therefore, in cases where the noise in the measurements is significant one would use Algorithm 2.

For the case of low measurement noise, one can imagine using a two phase algorithm in which we start with Algorithm 2 until a certain percentage error is achieved then switch to Algorithm 1 for fine tuning. Alternatively, one could use an hybrid algorithm that slowly increases the number of iterations in σ and ϕ at each stage.

[2]These percentage errors are expressed in absolute units rather than in units of percent. We will henceforth adhere to this convention.

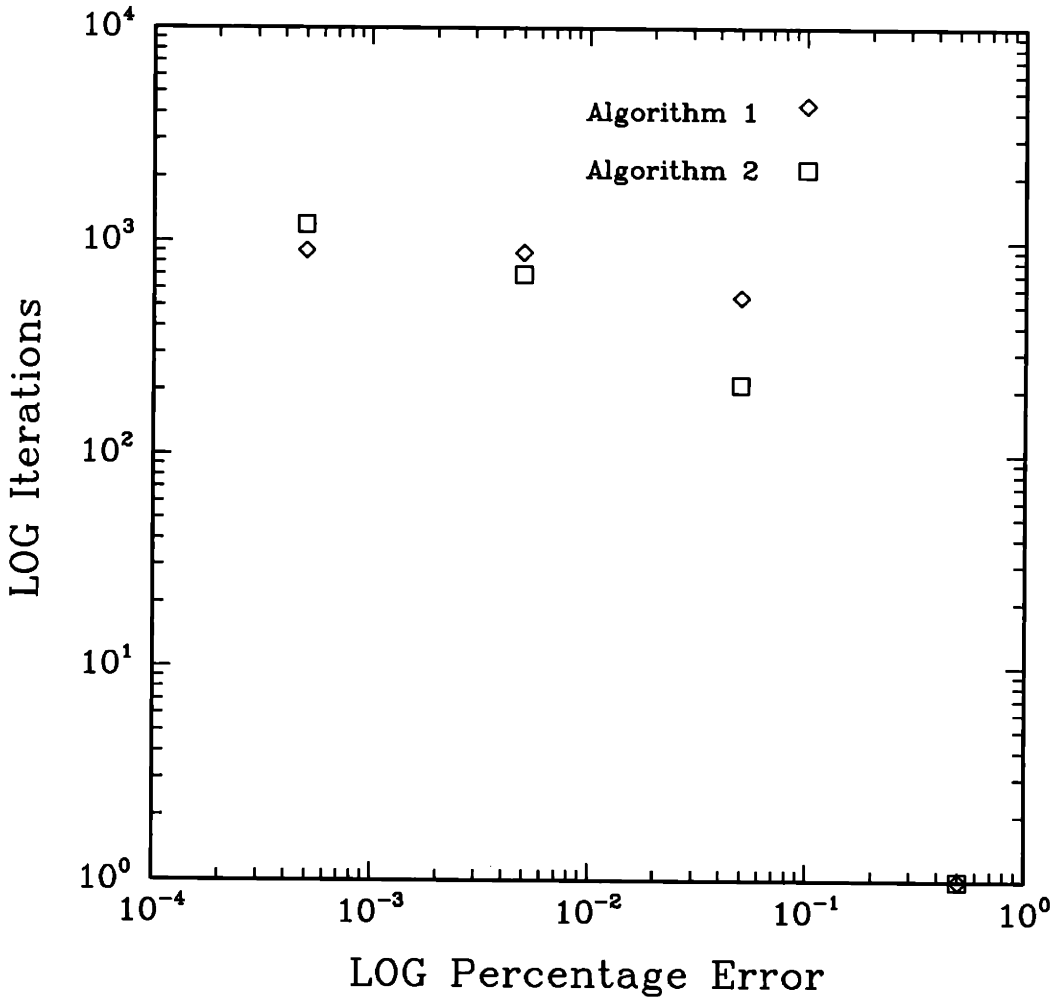

Figure 6.1: Log-log Plot of Total Number of σ Iterations Performed For 4 Different Percentage Error Criteria

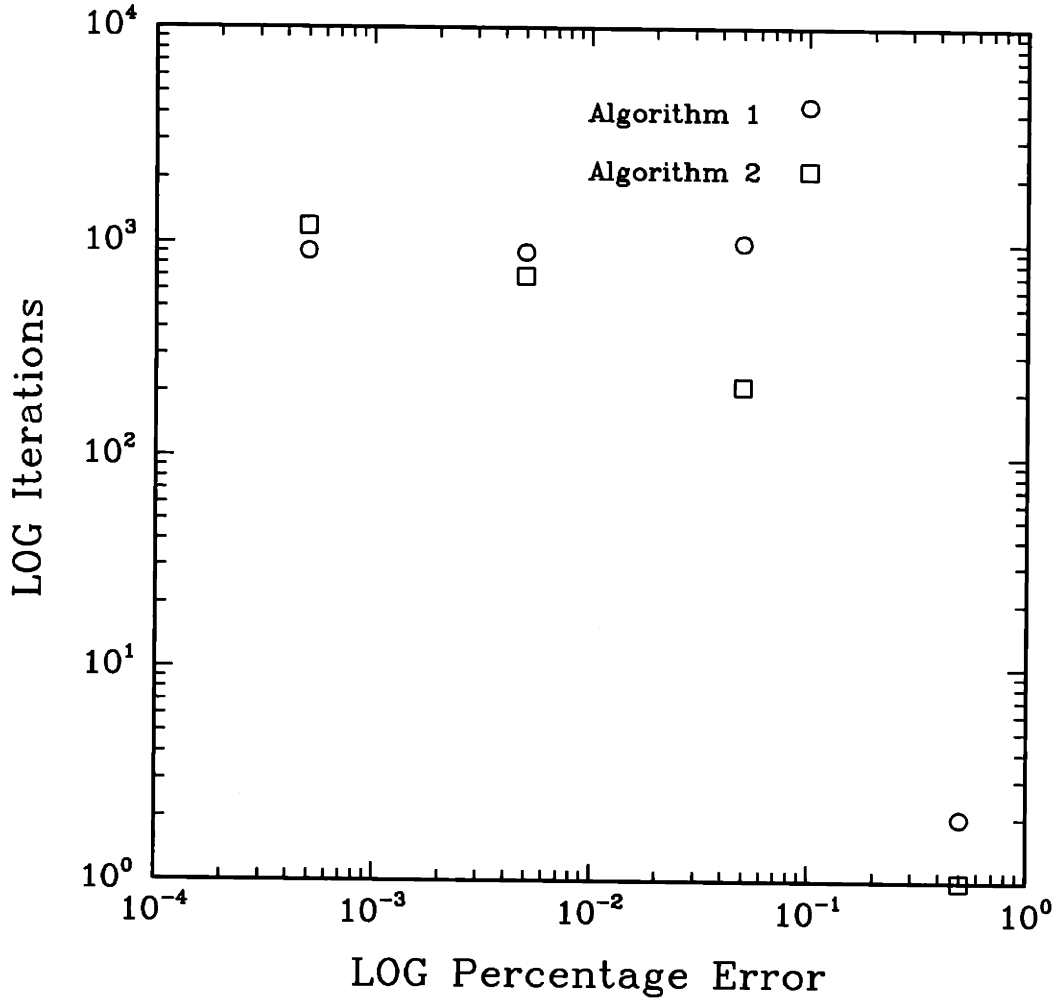

Figure 6.2: Log-log Plot of Total Number of ϕ Iterations Performed For 4 Different Percentage Error Criteria

6.3 Presence of Local Minima

In this section we investigate the presence of local minima by plotting the cost function, evaluated using noiseless data, along certain trajectories in the parameter space. It is clear that if we consider the parameter space to include both the unknown interior ϕ for each set of experiments as well as the unknown σ image, the number of degrees of freedom in the cost function is extremely large. Hence, we would expect the cost function to exhibit local minima along various trajectories in this space.

We define our parameter space to be the space of vectors, x, comprised of both σ and ϕ. Specifically,

$$x = \begin{pmatrix} \sigma \\ \phi \end{pmatrix} \tag{6.1}$$

where σ is a vector comprising the values of the conductivities in each of the sixteen squares and ϕ is a vector comprising samples of the potentials along each edge for each set of experiments.

Figure 6.3 is a plot of the cost function with respect to a line in the space of parameters going from the true parameter(the point at which $f = 0$), x_0, to an arbitrary parameter far away from x_0, which we denote $\tilde{x}$. In particular, the plot shows the cost function, $f(x;t)$, as a function of the scalar parameter t, which parametrizes the line as follows.

$$x = (1-t)x_0 + t\tilde{x} \tag{6.2}$$

where the parameter t varies from 0 to a positive value at which no entry of the σ part of the corresponding vector x goes negative[3]. The σ associated with $\tilde{x}$ is

[3]We do not plot in the domain of x where there exists negative σ since this corresponds to a non-physical situation.

depicted in Table 6.2. The interior ϕ associated with $\tilde{x}$ is set to 0.

10	40	90	10
25	30	49	640
81	100	121	10
16	196	20	560

Table 6.2: σ corresponding to $\tilde{x}$

Note that the plot in Figure 6.3 reveals the presence of a local minimum. This minimum occurs at $t = .93$ corresponding to $\tilde{x}$. Note that $\tilde{x}$ is not a feasible solution with respect to our cost function as λ approaches infinity, but indicates that local minimum structure is encountered when σ and ϕ are allowed to vary arbitrarily, as they are in each stage of our algorithm which involves *finite* values of λ.

We now consider the structure of the cost function from the true parameter, x_0, to a parameter having a constant σ image. Let the true parameter have a σ component associated with the σ image in Table 6.3. Again, we plot along a line, this time from the vector x_0 to the vector which we call $\hat{x}$. The vector $\hat{x}$ is composed of a constant σ image of value 2500 and its associated ϕ components consist of solutions to the PDE with this constant σ(in this case these are the samples of the solution to Laplace's equation along each edge for each experiment). The plot is shown in Figure 6.4.

Note that the plot in Figure 6.4 is convex, even though the σ images corresponding to x_0 and $\hat{x}$ differ greatly. The ϕ associated with $\hat{x}$, however, was not randomly

134.283	857.993	1055.41	1000
1350.26	370.622	1209.6	1660
1806.9	299.156	1443.02	738.026
1683.87	1137.04	1000	1703.6

Table 6.3: True σ

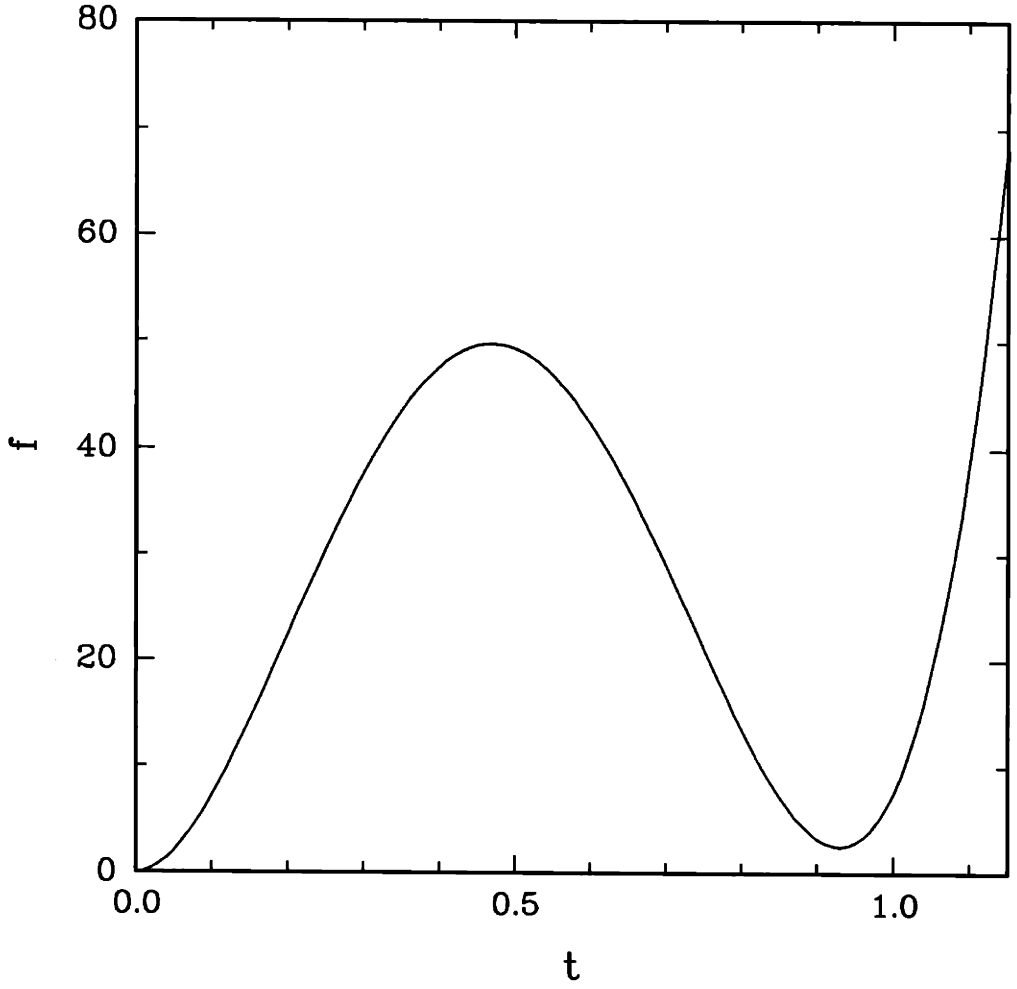

Figure 6.3: Cost Function vs. t; $\tilde{x}$ corresponds to $t = .93$

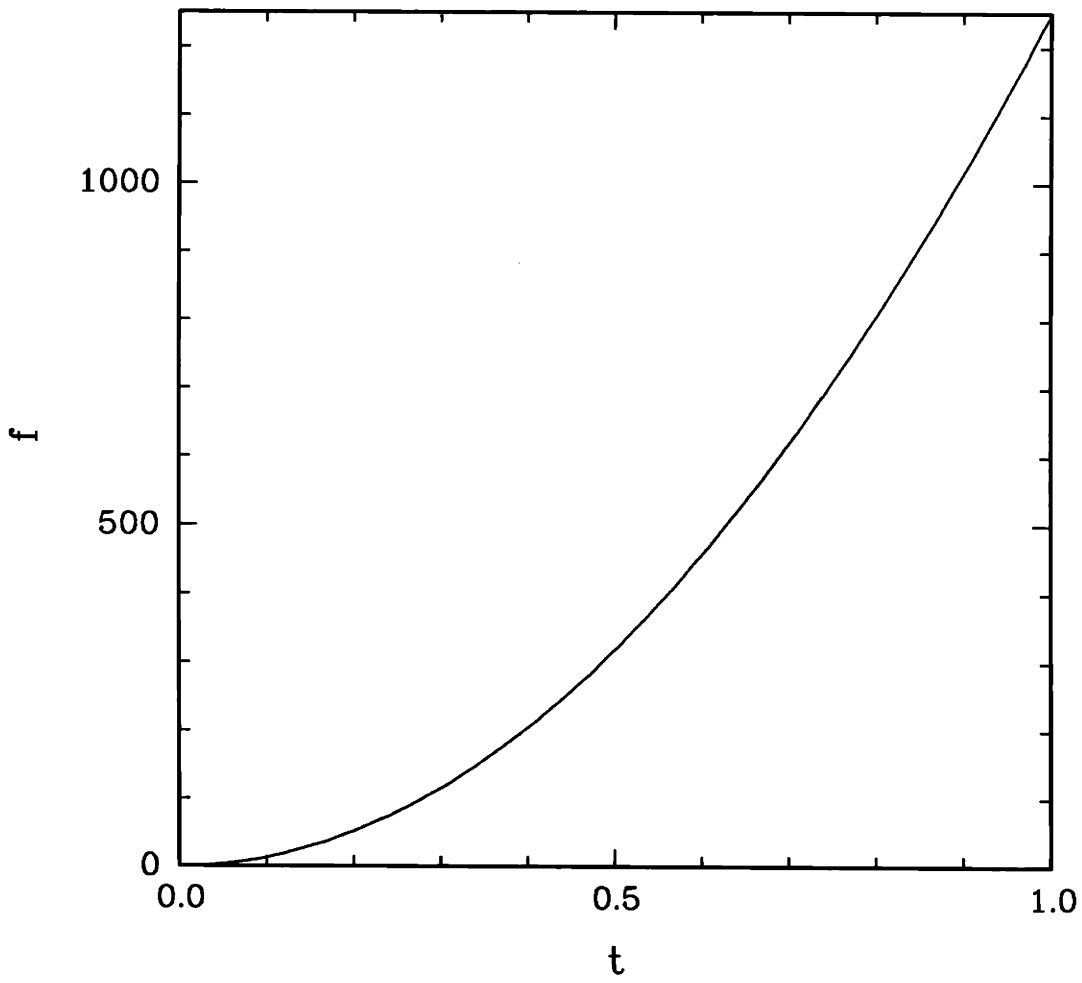

Figure 6.4: Cost Function vs. t; $\hat{x}$ corresponds to $t = 1$

chosen this time. Rather, it was chosen to satisfy the PDE. Results from additional experiments of this type indicate that for x_0 and $\hat{x}$ with greatly differing σ images, if the ϕ corresponding to $\hat{x}$ satisfies the PDE the cost function along the resulting trajectory is convex.

Additional experimentation with plots along numerous trajectories of the type shown in this section indicate that there are trajectories which exhibit local minima in the cost function. However, by confining oneself to vectors x in which ϕ must satisfy the PDE associated with the corresponding value for σ, the cost function appears to be convex. For the sixteen-square scale the presence of local minima seems to be an issue only when ϕ is allowed to vary greatly and does not even approximately correspond to the solution of the PDE. Since our algorithm starts each scale with σ and ϕ that match and each successive relaxation step keeps σ and ϕ close, we would not expect it to fall into local minima when applied to noiseless data. All of our experiments on noiseless data seem to support this claim.

Note that for higher resolutions one would expect the issue of local minima to become increasingly more important. Note also that the presence of measurement noise, even at the sixteen-square resolution, may very well introduce local minima.

6.4 Using Coarse Scale Information to Initiate Fine Scale Iteration

In this section we wish to demonstrate the effects of coarse scale information on the performance of the algorithm at a fine scale by considering how the number of iterations at the sixteen-square scale required to meet a certain performance criterion(e.g percentage error) might be reduced by initiating these iterations using coarse scale

10.5	12.5	105	75
9.5	7.5	95	125
95	80	15	10.6
105	120	5	9.4

Table 6.4: True σ for Section 6.4

information. We do this by considering the performance of the algorithm at the sixteen-square scale for the following initial conditions.

1. Random initial σ.

2. Constant initial σ where the constant equals the mean of the sixteen values of the true σ image.

3. Initial σ is constant over blocks of four squares such that the resulting four blocks can be thought of as corresponding to a four-square scale of the image. The constant for each of these four blocks is equal to the mean of the four squares in each block of the true σ image.

One can think of the second set of initial conditions as corresponding to information at a constant scale and the third set as corresponding to information at a four-square scale. We apply Algorithm 2 in each case to noiseless data generated using the true σ image in Table 6.4. Since we are using noiseless data $\frac{\gamma}{\lambda}$ is fixed to be 1.

In order to compare performance under the different initial conditions, we apply the algorithm to the data until a certain percentage error is achieved. Specifically, we stop the algorithm once the average percentage error in the four inner squares reaches 5 percent. We compare the performances under each set of initial conditions by plotting for each case the average percentage errors of the inner four squares as a function of the number of iterations. We essentially compare how the errors decrease

10.4998	12.4903	104.998	75.0003
9.49703	8.45016	94.8412	125
95.0018	79.7948	13.9734	10.6021
105	120.008	5.00385	9.40003

Table 6.5: Random Initial σ, Example 1.

10.4998	12.4901	104.999	75.0003
9.49697	8.46586	94.7443	125.001
95.003	79.624	14.049	10.6019
105	120.01	5.00391	9.40003

Table 6.6: Random Initial σ, Example 2.

as a function of the number of iterations for each set of initial conditions.

We begin with case of random initial σ. We give two examples, each of which uses random initial σ with the following statistics.

$$\begin{aligned} \sigma_i &= 100 + \delta\sigma_i \\ i &= 1, ..., 16 \end{aligned} \tag{6.3}$$

where

$$\begin{aligned} \left(E\left\{\delta\sigma_i^2\right\}\right)^{\frac{1}{2}} &= 40 \\ E\left\{\delta\sigma_i \delta\sigma_j\right\} &= \delta_{ij} \end{aligned} \tag{6.4}$$

Tables 6.5-6.6 show the initial σ images used for the two examples.

Next, we consider the case of initial conditions at a constant scale. In this case the initial σ is constant over the sixteen squares where the constant is equal to the mean of the sixteen squares of the true σ. Table 6.7 shows the initial conditions used for this example.

Finally, we consider the case of initial conditions at a four-square scale. In this case the initial σ is constant over blocks of four squares where each of the four blocks

55	55	55	55
55	55	55	55
55	55	55	55
55	55	55	55

Table 6.7: Constant Scale Initial σ

10	10	100	100
10	10	100	100
100	100	10	10
100	100	10	10

Table 6.8: Four-square Scale Initial σ

corresponds to a square at the four-square scale. The value of the constant in a block equals the mean of the corresponding four squares of the true σ. Table 6.8 shows the initial conditions used for this example.

Figure 6.5 is a plot of the log of the average percentage error in the inner four squares versus the number of iterations for the four different initial conditions. This plot illustrates the performance of the algorithm under the various initial conditions. The performance using constant scale initial conditions is far better than either of the two performances using random initial conditions. The performance using four-square initial conditions is much better than that using constant scale intitial conditions. This plot inidicates that coarse scale information in initiating iterations at a finer scale does help overall performance at that fine scale. Furthermore, while constant scale initial conditions provide significant improvement in performance over the case of using random initial conditions, four-square initial conditions provide an even greater improvement.

The numerical examples in this section support the idea that coarse scale information is useful in helping performance at a fine scale. Coarse scale information can

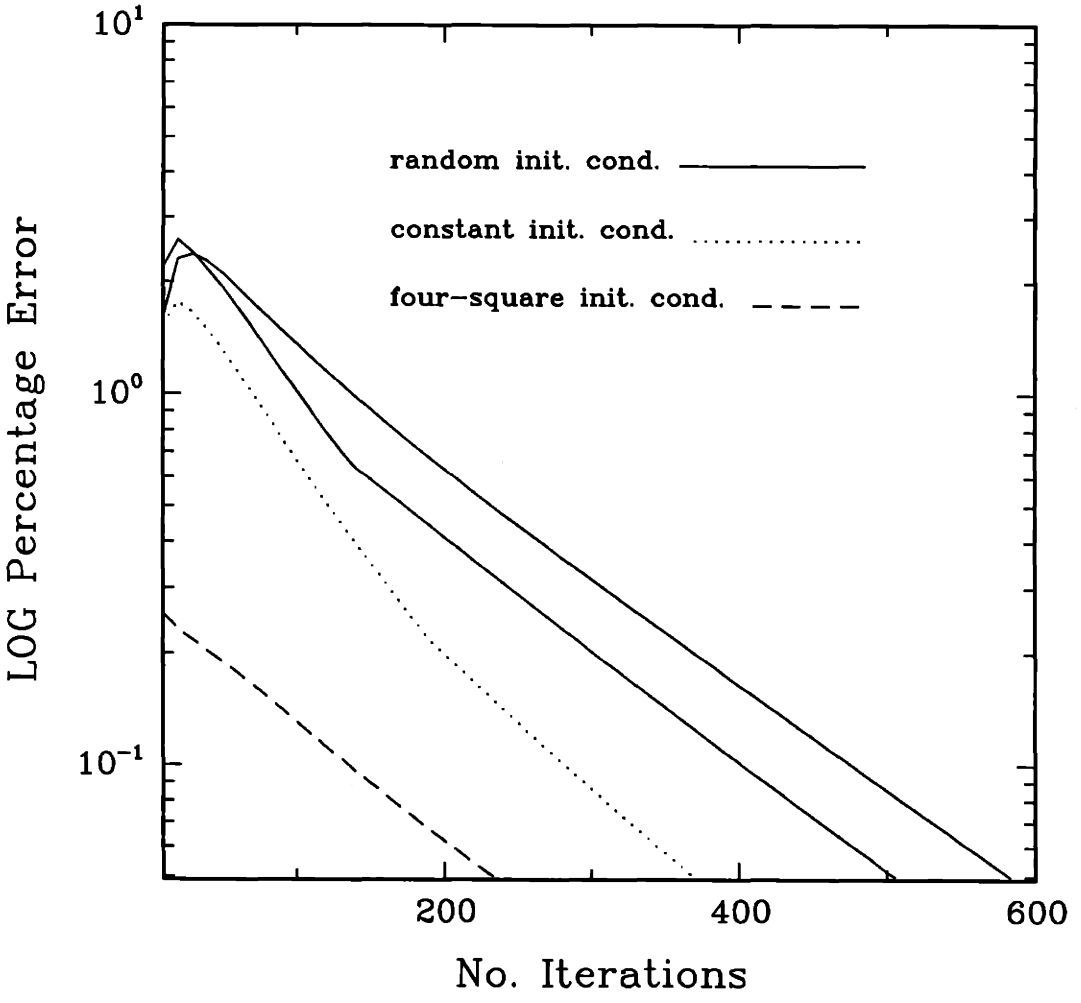

Figure 6.5: Semi-log Plot of Percentage Error vs. Number of Iterations for Four Different Initial Conditions

be used to provide a starting point for iterations at a fine scale that is closer to the global minimum than a point provided randomly. This idea is all the more valuable in view of the fact that computation at coarse scales is considerably cheaper than computation at fine scales.

6.5 Performance of the Overall Algorithm on Noiseless and Noisy Data

In this section we demonstrate the performance of the overall algorithm on both noiseless and noisy data. The overall algorithm consists of estimating σ at three different scales starting from a constant σ, then proceeding to a four-square σ, and finally culminating in a sixteen-square σ. For the case of noiseless data we simply demonstrate the success of the algorithm and give an indication of its performance characteristics at the various scales. For the case of noisy data we wish to explore the effects of measurement noise in corrupting the performance of the overall algorithm. The presence of noise in the measurements also makes the issue of varying $\frac{\gamma}{\lambda}$ important. We investigate the effects of varying $\frac{\gamma}{\lambda}$ over the course of iterating at the sixteen-square scale by applying Algorithm 2 with a particular $\frac{\gamma}{\lambda}$ schedule for a moderate noise level(SNR = 50). We also give an example in which the noise level is particularly high(SNR = 10); this illustrates the effects of high noise levels on algorithm performance.

Let us start by considering the overall algorithm applied to noiseless data generated using the σ image in Table 6.9. This table shows σ represented at three separate scales: constant, four-square, and sixteen-square. The constant and four-square scales represent the true sixteen-square image averaged at the appropriate

104.531

83.9145	111.563
111.587	111.058

56.7142	92.8996	102.77	100
117.513	68.5311	110.48	133
140.345	64.9578	122.151	86.9013
134.194	106.852	100	135.18

Table 6.9: True σ for Section 6.5; 3 Separate Scales

scale.

We run Algorithm 2 using the full multi-resolution scheme over the 3 separate scales. Since we are using noiseless data, $\frac{\gamma}{\lambda}$ is fixed to be 1. The implementation of the overall algorithm can be summarized as follows where we denote the estimated σ as $\hat{\sigma}_i$. The subscript i denotes the scale where $i = 0$ is the constant scale, $i = 1$ is the four-square scale, and $i = 2$ is the sixteen-square scale.

1. Solve PDE for constant σ case(Laplace's equation) for ϕ corresponding to each set of excitations.

2. Solve for $\hat{\sigma}_0$ - linear least squares problem.

3. Initiate all four squares of $\hat{\sigma}_1$ with $\hat{\sigma}_0$.

4. Initialize ϕ at this scale; i.e. solve PDE for ϕ for each set of excitations.

5. Alternate between optimizing for σ and ϕ using Algorithm 2. Continue until adequate convergence has occurred.

6. Initiate blocks of four squares of $\hat{\sigma}_2$ with corresponding values of $\hat{\sigma}_1$.

7. Initialize ϕ at this scale; i.e. solve PDE for ϕ for each set of excitations.

5 iterations

2.5529e-01

19 iterations

-0.10365	-4.757537e-02
0.16729	7.239e-02

1072 iterations

3.22905e-08	2.5241e-06	-5.208323e-06	7.516032e-09
1.242908e-06	-3.154515e-04	8.111469e-04	-4.790822e-06
-2.214538e-06	4.80085e-04	-1.502684e-03	6.174205e-06
-5.050663e-08	-1.63544e-06	6.160476e-06	7.208871e-08

Table 6.10: Full-scale Algorithm on Noiseless Data; $\beta = .0001$

8. Alternate between optimizing for σ and ϕ using Algorithm 2. Continue until adequate convergence has occurred.

We take adequate convergence to be the point in the iteration at which the percentage change of $\hat{\sigma}_i$ in the inner squares falls below a certain threshold which we call β. Table 6.10 gives results for an example where $\beta = .0001$. This table shows the number of iterations required for convergence corresponding to β and the final percentage error at each scale.

Table 6.10 illustrates the performance of the multi-resolution algorithm on noiseless data. Note that the percentage errors increase as the scale gets coarser. This essentially shows the effects of modelling error(i.e. σ varies at a finer scale) and in particular the fact that this error gets worse as the scale gets coarser. Note also the small number of iterations required at the constant and four-square scale. These rather inexpensive computations provide a way of initiating the iterations at the sixteen-square scale. As we saw in Section 6.4 coarse scale information helps to start the optimization at a fine scale closer to the global minimum than could be done by just randomly choosing the initial point.

Let us now introduce measurement noise into our data. With the introduction of measurement noise, we must concern ourselves with varying $\frac{\gamma}{\lambda}$. We consider varying it over the course of iterations at the sixteen-square scale. In particular we supply a schedule for varying $\frac{\gamma}{\lambda}$ such that the algorithm at the sixteen-square scale is implemented in the following way.

1. Initialize $\frac{\gamma}{\lambda} = 1$
2. Perform Algorithm 2 until adequate performance is achieved.
3. Decrease $\frac{\gamma}{\lambda}$.
4. Repeat step 2.

   ⋮

5. Stop

Table 6.11 gives an example of a particular schedule for $\frac{\gamma}{\lambda}$ at the sixteen-square scale. The full multi-resolution algorithm is applied to noisy measurements where the SNR = 50. The value of $\frac{\gamma}{\lambda}$ is changed after convergence is reached to within .001 percent. The table shows the percentage errors in the image for each value of $\frac{\gamma}{\lambda}$, the number of iterations required at each stage, the value of the penalty term, and the average percentage errors in the inner four squares.

We can see for this example that as one would expect, as λ increases($\frac{\gamma}{\lambda}$ decreases) the value of the penalty term decreases. We can also see that increasing the weighting of the penalty term helps to decrease the average percentage error in the inner squares; i.e. as λ increases the average percentage error in the inner squares decreases steadily, slowly converging to approximately 7 percent. Note,

$\frac{\gamma}{\lambda} = 1$

no. iterations = 146, penalty term = 14953

average percentage error in inner squares = .0841124

1.116865e-03	3.573474e-03	5.468815e-03	-1.209706e-03
2.928781e-03	-5.457093e-02	-0.11152	2.732688e-03
1.092839e-03	7.3854e-02	-9.650476e-02	1.143622e-03
-4.23279e-03	-1.082793e-03	5.147992e-04	-2.631879e-04

$\frac{\gamma}{\lambda} = 0.1$

no. iterations = 19, penalty term = 849.412

average percentage error in inner squares = .078456

1.384348e-03	4.028474e-03	5.519846e-03	-1.594384e-03
2.758936e-03	-4.819568e-02	-0.11222	3.861575e-03
9.992681e-04	5.797102e-02	-9.543775e-02	2.589183e-04
-4.341527e-03	-1.501125e-03	4.857844e-04	-2.630233e-04

$\frac{\gamma}{\lambda} = 1.e\text{-}02$

no. iterations = 3, penalty term = 511.132

average percentage error in inner squares = .0778334

1.509411e-03	4.771796e-03	6.151331e-03	-1.955762e-03
2.337945e-03	-4.717167e-02	-0.11184	4.813939e-03
9.467833e-04	5.658424e-02	-9.573809e-02	3.098729e-04
-4.428486e-03	-2.019515e-03	2.331867e-04	-5.84557e-04

$\frac{\gamma}{\lambda} = 1.e\text{-}03$

no. iterations = 29, penalty term = 57.783

average percentage error in inner squares = .0747858

3.10745e-03	7.596387e-03	8.599716e-03	-2.781739e-03
2.280133e-03	-4.063808e-02	-0.107	9.097123e-03
3.643813e-04	5.183566e-02	-9.96698e-02	3.801678e-04
-4.753442e-03	-4.770085e-03	-1.250691e-03	-2.010307e-03

$\frac{\gamma}{\lambda} = 1.e\text{-}04$

no. iterations = 10, penalty term = 29.1853

average percentage error in inner squares = .0743213

3.493694e-03	8.072462e-03	9.009605e-03	-2.861106e-03
2.368201e-03	-3.952806e-02	-0.1062	9.74429e-03
2.505082e-04	5.119723e-02	-0.10036	4.006887e-04
-4.768379e-03	-5.251727e-03	-1.496641e-03	-2.286959e-03

Table 6.11: Percentage Errors at Sixteen-square Scale After Each Change in $\frac{\gamma}{\lambda}$; SNR = 50

however, that the bulk of the decrease in the error has occurred during the iteration at which $\frac{\gamma}{\lambda} = 1$; in particular, at the end of the iterations at this stage the average percentage error in the inner squares is roughly 8 percent. This indicates that although increasing λ helps in enforcing the PDE constraint more and more, as evidenced by the decrease in the penalty term, the effects in the percentage errors of the inner σ is relatively small.

The errors of σ in the outer squares actually increases slightly with increasing λ. Note, however, that these errors are an order of magnitude smaller than the errors of the inner squares. The effect of increasing λ is to improve the performance of the inner squares at the expense of a slight decrease in performance of the outer squares. But the errors of the inner squares dominate the errors of the outer squares, and the average error among all the squares actually decreases with increasing λ.

The results in Table 6.11 show that although overall performance is improved by increasing λ, this improvement is relatively small. From an algorithmic standpoint, one does fairly well iterating at the sixteen-square scale while keeping $\frac{\gamma}{\lambda}$ fixed to be 1; i.e. one makes relatively small gains in performance by increasing λ.

We now consider the same schedule applied to data of a lower SNR. Table 6.12 shows results for the algorithm applied under the same conditions as for Table 6.11 to data in which the SNR is equal to 10. Note that the penalty term is decreasing as we increase λ indicating that our PDE constraint is being enforced gradually. Note also, however, that the average percentage error in the inner squares is actually increasing with each increase in λ. The algorithm is converging to a point in which the errors on the inner squares is relatively large. This is an indication of the ill-conditioning of the problem for data in which there is a low SNR. Note,

$\frac{\gamma}{\lambda} = 1$

no. iterations = 3305, penalty term = 3.750448e+05

average percentage error in inner squares = .8188875

5.150245e-03	1.83985e-02	3.144681e-02	-6.073196e-03
1.60879e-02	-0.24051	-0.68039	1.695653e-02
1.025042e-02	-0.54712	1.80753	-5.425443e-04
-2.088104e-02	6.341654e-04	-2.637193e-03	-1.258177e-03

$\frac{\gamma}{\lambda} = 0.1$

no. iterations = 8302, penalty term = 8661.59

average percentage error in inner squares = .93955

6.577001e-03	1.416598e-02	2.855178e-02	-6.305037e-03
1.528711e-02	0.13215	-0.62046	1.551851e-02
8.512232e-03	-0.51042	2.49517	-2.325263e-03
-2.12778e-02	4.696011e-04	-3.509145e-03	-1.192344e-03

$\frac{\gamma}{\lambda} = 1.e\text{-}02$

no. iterations = 2, penalty term = 3681.05

average percentage error in inner squares = .940305

6.636289e-03	1.567911e-02	2.98482e-02	-6.995385e-03
1.386343e-02	0.13374	-0.62066	1.664483e-02
8.066284e-03	-0.51147	2.49535	-2.214647e-03
-2.159426e-02	-9.814331e-05	-3.775518e-03	-1.68347e-03

$\frac{\gamma}{\lambda} = 1.e\text{-}03$

no. iterations = 191, penalty term = 50.6249

average percentage error in inner squares = .9464425

9.157433e-03	2.277429e-02	3.557535e-02	-1.061865e-02
1.5214e-02	0.18388	-0.59652	2.810223e-02
4.915636e-03	-0.52192	2.48345	-3.245904e-03
-2.161407e-02	-2.831905e-03	-5.024187e-03	-5.430857e-03

$\frac{\gamma}{\lambda} = 1.e\text{-}04$

no. iterations = 25, penalty term = 32.7646

average percentage error in inner squares = .9468125

9.471466e-03	2.320997e-02	3.583852e-02	-1.083947e-02
1.542539e-02	0.18621	-0.59581	2.86109e-02
4.739987e-03	-0.52216	2.48307	-2.952086e-03
-2.166181e-02	-2.873671e-03	-5.055524e-03	-6.06941e-03

Table 6.12: Percentage Errors at Sixteen-square Scale After Each Change in $\frac{\gamma}{\lambda}$; SNR = 10

however, that the errors in the outer squares remain small. This indicates that the performance of the outer squares is robust with respect to measurement noise, whereas the performance of the inner squares is not.

6.6 The Effects of Noise on Algorithm Performance and the Use of Inhomogeneous Spatial Scales

In this section we apply our algorithm to noisy data and characterize performance in terms of the average mean-square error of the estimate. Since our interest in this section is primarily on the effects of noise on estimation performance, and in particular the comparison of this performance for the case of outer squares versus the case for inner squares, we choose the following algorithmic implementation. We use Algorithm 2 exclusively at the sixteen-square scale, initiating the iteration using good initial conditions; i.e. using the average of the true σ in blocks of four. In each case we fix $\frac{\gamma}{\lambda}$ to be 1 and run the algorithm using an arbitrary number of iterations.

In particular we plot the average mean-square error of the conductivity estimate for a variety of values of SNR. By comparing plots of this type for estimates of the outer twelve squares with those for estimates of the inner four squares, we characterize the effects of noise on estimation performance as a function of distance from the overall boundary. We would expect performance to be better for the squares that have direct contact with the measurements than it would be for the inner squares.

Since we would expect the performance of the inner squares to be inferior to that of the outer squares, we also investigate the notion of treating the inner four

squares as one square, i.e. thinking of these squares as existing at a coarser scale than that of the outer squares. We do this by plotting the performance of estimating the *mean* of the inner four squares. Specifically, we plot the mean-square error as a function of SNR where the error is now the difference between the mean of the true values of the inner four squares and the mean of the estimated values of the inner four squares.

We consider a sixteen-square σ image in which the value of σ in each square is the sum of the average value σ_0 and an independent, Gaussian disturbance of standard deviation ρ. We use an image in which ρ/σ_0 is thirty percent. The value of the estimate $\hat{\sigma}$ corresponds to the value of σ computed using Algorithm 2 after 20 iterations. The reason for keeping the number of iterations low is to keep the number of computations required for the Monte-Carlo runs from being prohibitively large. Experiments generating a few points using a large number of iterations indicate that this is justified.

We define the following statistics.

$$\epsilon_1 = \left\{ \frac{1}{12} \sum_o E\left[(\sigma_o - \hat{\sigma}_o)^2 \right] \right\}^{\frac{1}{2}} \tag{6.5}$$

$$\epsilon_2 = \left\{ \frac{1}{4} \sum_i E\left[(\sigma_i - \hat{\sigma}_i)^2 \right] \right\}^{\frac{1}{2}} \tag{6.6}$$

$$\epsilon_3 = \left\{ E\left[(\frac{1}{4} \sum_i \sigma_i - \frac{1}{4} \sum_i \hat{\sigma}_i)^2 \right] \right\}^{\frac{1}{2}} \tag{6.7}$$

where

$$o \epsilon \{1, 2, 3, 4, 5, 8, 9, 12, 13, 14, 15, 16\} \tag{6.8}$$

and

$$i \epsilon \{6, 7, 10, 11\} \tag{6.9}$$

The statistic ϵ_1 represents the average of the root mean-square error of the estimate over the twelve outer squares, while ϵ_2 is the average of the root mean-square error of the estimate over the four inner squares. The statistic ϵ_3 the root mean-square of the error between the average of the four inner squares and the average of their estimates. We compute the expectations using Monte Carlo trials.

Figure 6.6 is a plot of the statistics ϵ_1, ϵ_2, and ϵ_3 normalized by σ_0, i.e. the percentage errors, as a function of SNR where the SNR varies from 1 to 100. Figure 6.7 is a plot of the same thing with focus on the range of SNR from 1 to 50.

Figures 6.6 and Figure 6.7 show the outer squares to have much better performance than the inner squares. The statistic ϵ_1/σ_0 converges towards an error of approximately .25 percent for large SNR whereas ϵ_2/σ_0 converges towards approximately a 22 percent error. This indicates a wide disparity between performance of the outer squares and performance of the inner squares.

Note the similarity in performance between the statistic ϵ_3 and the statistic ϵ_1. For $SNR = 100$ ϵ_3/σ_0 is approximately .6 percent, which is on the order of the value of ϵ_1/σ_0 for the same SNR. Clearly, the performance of statistics ϵ_3 and ϵ_1 is far better than that of statistic ϵ_2. This indicates that by treating the four inner squares as one large square we can estimate the mean of these squares with the same order of error variance as that for the outer squares.

In summary, the plots in this section indicate that the performance of the algorithm on outer squares is much better than the performance on inner squares, in fact by as much as two orders of magnitude. This gives evidence of the fact that estimation performance deteriorates at points removed from the boundary, on which there are excitations and measurements. If we treat the four inner squares as

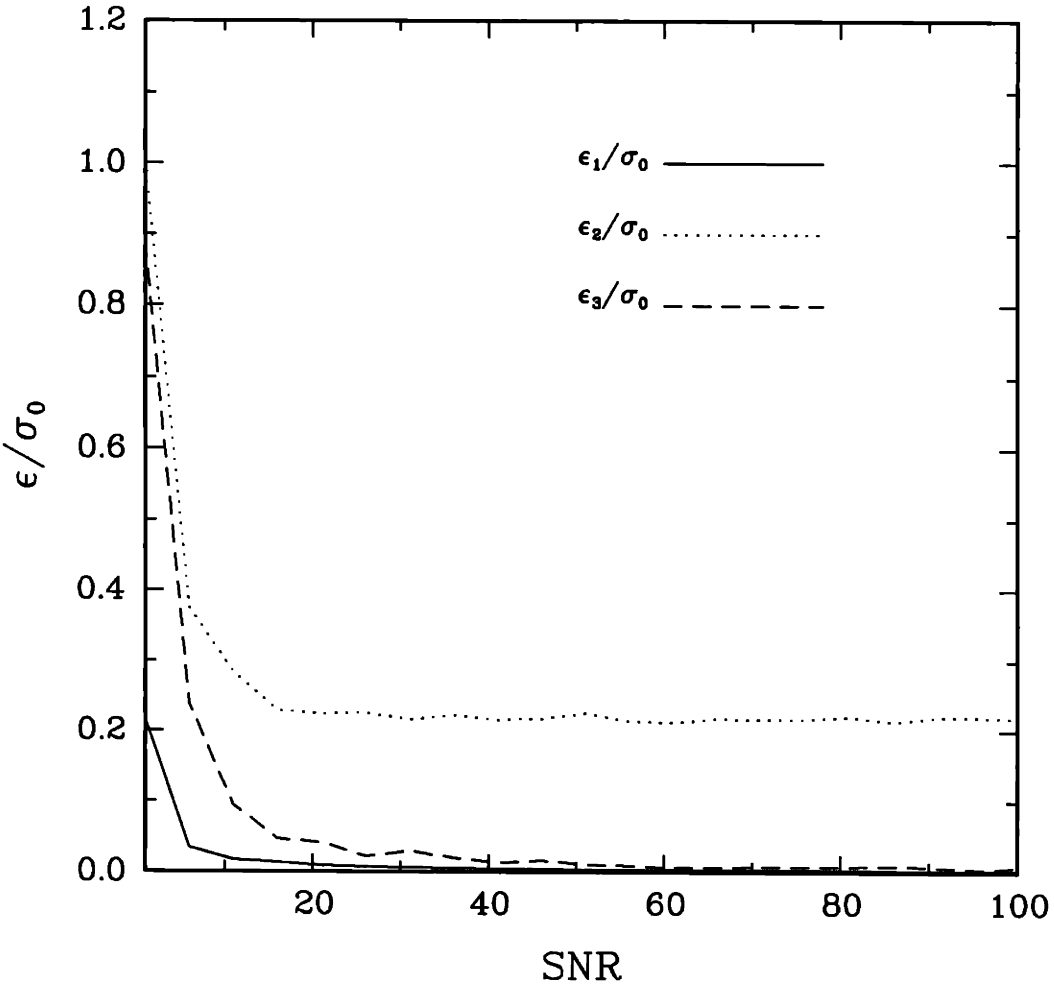

Figure 6.6: Plot of the Statistics ϵ_1/σ_0, ϵ_2/σ_0, and ϵ_3/σ_0 vs. SNR

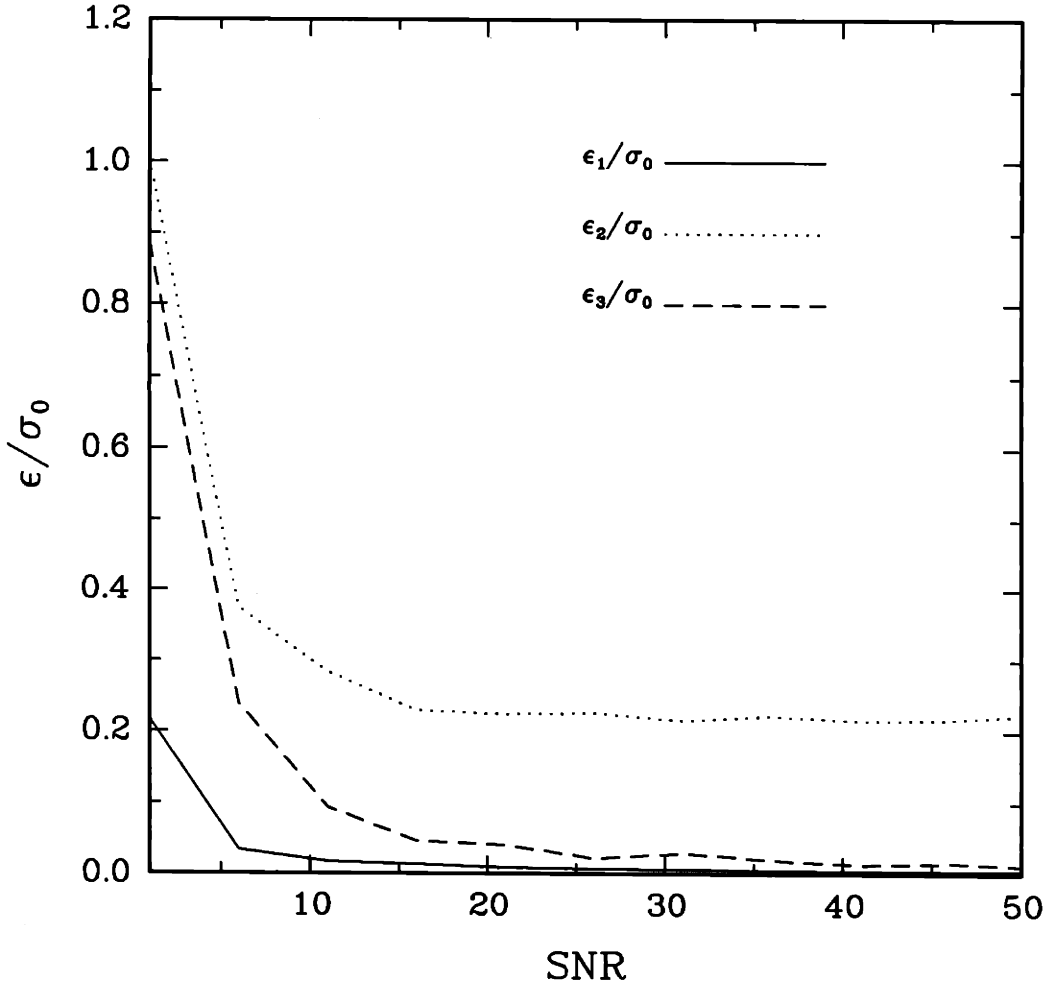

Figure 6.7: Close-up of Figure 6.6

one large square, however, the performance in estimating an average conductivity for this large square is on the same order as the performance for the outer squares. This result suggests the possibility of treating the inner squares at a coarser scale than that of the outer squares.

6.7 Conclusions

Numerical results show that good convergence is obtained by doing single relaxation sweeps in σ and ϕ. The results in Section 6.2 indicate that spending too much time on either optimizing with respect of σ or optimizing with respect to ϕ is not beneficial for the case in which we are trying to achieve a percentage error above a small threshold; this case occurs for instance when there is noise in the data and hence, we would not expect to achieve percentage errors below some threshold. For the case of data of extremely high SNR in which we might expect to achieve very low percentage errors we can imagine an algorithm that starts with single sweeps in σ and ϕ then gradually increases the number of sweeps per iteration as the percentage errors decrease.

Our plots of the cost function for noiseless data along various trajectories in the state space indicate the presence of local minima when σ and ϕ are allowed to vary arbitrarily. The presence of local minima for the noiseless case does not seem to be an issue, however, when ϕ is chosen to solve the PDE associated with σ. Since our algorithm starts each scale with σ and ϕ that match and each successive relaxation step keeps them close, we would not expect to encounter local minima for relatively noise-free data.

Numerical results support the idea that coarse scale information helps in speed-

ing up the convergence of the algorithm at a fine scale. The closer the scale of the information to the fine scale, the better the improvement obtained by using this information in initiating fine scale iterations. This supports the idea of the multi-resolution algorithm being computationally efficient.

With regard to the issue of varying λ, it seems for relatively low noise levels the effects of increasing λ from a performance standpoint are not great. One achieves relatively good performance by iterating sufficiently while fixing $\frac{\gamma}{\lambda}$ to be 1. By increasing λ one can decrease the average error, but this decrease is not great. Our experiments show that at the sixteen-square scale by fixing $\frac{\gamma}{\lambda}$ we can achieve relatively low errors, despite the fact that the PDE constraint is not satisfied as strictly as in the cases where λ is increased toward infinity.

And finally, our experiments with noisy data confirm the idea that estimation performance degrades at points removed from the boundary, on which there are direct measurements. Our experiments with high noise levels indicate that while the performance on the outer squares is rather robust, the performance on the inner squares is quite sensitive to these noise levels. This suggests the idea of treating inner squares at a coarser scale from that of outer squares. Our experiments on the performance of the mean of the inner four squares show this idea to have merit.

Chapter 7

Conclusions

This thesis has presented a way of controlling the large number of degrees of freedom for an inverse conductivity problem by estimating the conductivity at various spatial scales. We have developed a multi-resolution algorithm that is based on a nested sequence of highly parallelizable relaxation schemes and have demonstrated the success of this algorithm on synthetic data as well as investigated various algorithmic issues. We have also produced analytical results based on Cramer-Rao bounds that characterize estimation performance at various scales. These results provide a way of determining the scale which a particular set of excitations and measurements supports. And finally, we have explored the effects of noise on algorithm performance. In particular we have shown how performance suffers for points removed from the boundary.

7.1 Thesis Contributions

We now summarize the main results of this thesis. The following is a list of the contributions made in the order of their presentation in the thesis.

1. We presented a maximum-likelihood formulation of the inverse conductivity problem for the case of a 2D piecewise constant conductivity image. In our formulation we solve an optimization problem in which we minimize a cost function, the negative likelihood function, with constraints on the parameters in the form of a PDE. We parametrized the problem in such a way so as to solve the constrained problem by minimizing an augmented cost function in which a penalty function is used to represent these constraints. This resulted in an optimization problem that is spatially separable and highly conducive to a parallel computational scheme.

2. We presented a multi-resolution algorithm that solves the ML problem at the finest scale by starting at a coarse scale then moving to successively finer scales. The algorithm at each spatial scale is highly iterative, consisting of a nested sequence of simple, linear relaxation algorithms which are extremely parallelizable.

3. We presented numerical results that characterize the problem of estimating at a coarser scale than that of the true conductivity by studying in detail the problem of estimating a constant σ when in fact σ is piecewise constant in four squares.

4. We presented a general method for computing the Cramer-Rao bound for estimating the conductivity at a particular scale. We presented numerical results for the four-square case. Our method is equally applicable to finer resolutions and provides a way of characterizing estimation performance for an arbitrary scale of σ.

5. We demonstrated the success of our algorithm on synthetic data for low noise levels and explored various algorithmic issues. Our results in general show the multi-resolution approach to have merit. We have also shown how performance deteriorates at points removed from the boundaries for sufficiently high noise levels.

7.2 Suggestions for Future Research

- There is room for improvement of the convergence properties of the σ and ϕ Gauss-Seidel iteration schemes. A *successive-overelaxation*(SOR) method would be helpful in speeding up convergence [4].

- An interesting area for future research would be to pursue the idea of developing an algorithm that moved from fine scales back to coarse scales as well as from coarse to fine. This would coincide more with the multi-grid philosophy of doing coarse grid corrections based on iterations at fine grids [2], [5].

- Our experimental results for the sixteen-square case indicate that it may be fruitful to aggregate the pixels more coarsely as one moves inward from the boundary. This is reasonable given that the performance deteriorates moving inward from the boundary. A possible area of research is the investigation of an algorithm that pixelates the σ more coarsely at points removed from the boundaries.

- Our multi-resolution estimation approach suggests possible application to various other inverse problems in which the number of degrees of freedom in the parameter is large.

Bibliography

[1] W. F. Ames, *Numerical Methods for Partial Differential Equations*, Academic Press, New York, N. Y., 1977.

[2] A. Brandt, *Multi-level Adaptive Solutions to Boundary Value Problems*, Math. Comp. Vol. 13, pp.333-390, 1977.

[3] K. Dines and R. Lytle, *Analysis of Electrical Conductivity Imaging*, Geophysics, Vol. 46, No.7, pp. 1025-1036, 1981.

[4] G. Golub and C. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, MD, 1983.

[5] W. Hackbusch and U. Trottenberg, Eds., *Multigrid Methods*, Springer-Verlag, N.Y., N.Y., 1982.

[6] A. K. Jain and E. Angel, "Image Restoration Modelling and Reduction of Dimensionality," *IEEE Trans. Computers*, Vol. C-23, No. 5, pp. 470-476, May 1974.

[7] D. Luenberger, *Linear and Nonlinear Programming*, Addison Wesley, Reading, MA, 1984.

[8] D. Paddon and H. Holstein, Eds., *Multigrid Methods for Integral and Differential Equations*, Clarendon Press, Oxford, England, 1985.

[9] G. Strang, *Linear Algebra and its Applications*, Academic Press, New York, N.Y., 1980.

[10] A. Tarantola and B. Valette, *Generalized Nonlinear Inverse Problems Solved Using the Least Squares Criterion*, Reviews of Geophysics and Space Physics, Vol. 20, No.2, pp. 219-232, 1982.

[11] A. Tarantola and B. Valette, *Inverse Problems = Quest for Information*, Journal of Geophysics, Vol. 50, pp. 159-170, 1982.

[12] D. Terzopoulos, *Image Analysis Using Multigrid Relaxation Methods*, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, No. 2, pp. 129-139, March 1986.

[13] H. L. Van Trees, *Detection, Estimation and Modulation Theory*, vol. 1, John Wiley, New York, NY, 1968.