Distribution Agreement

In presenting this dissertation as a partial fulfillment of the requirements for an advanced degree from Emory University, I agree that the Library of the University shall make it available for inspection and circulation in accordance with its regulations governing materials of this type. I agree that permission to copy from, or to publish, this dissertation may be granted by the professor under whose direction it was written when such copying or publication is solely for scholarly purposes and does not involve potential financial gain. In the absence of the professor, the dean of the Graduate School may grant permission. It is understood that any copying from, or publication of, this dissertation which involves potential financial gain will not be allowed without written permission.

Signature:

Zhuojun Magnant	Date

Numerical Methods for Optimal Experimental Design of Ill-posed Problems

By

Zhuojun Magnant
Doctor of Philosophy

Department of Mathematics and Computer Science

Eldad Haber, Ph.D.
Advisor

James Nagy, Ph.D.
Advisor

Michele Benzi, Ph.D.
Committee Member

Alessandro Veneziani, Ph.D.
Committee Member

Lisa A. Tedesco, Ph.D.
Dean of the Graduate School

Accepted

Date

Numerical Methods for Optimal Experimental Design of Ill-posed Problems

By

Zhuojun Magnant

Advisor: Eldad Haber, Ph.D.
Advisor: James Nagy, Ph.D.

An abstract of
a dissertation submitted to the Faculty of the Graduate School
of Emory University in partial fulfillment
of the requirements of the degree of
Doctor of Philosophy

Department of Mathematics and Computer Science

2011

Abstract

Numerical Methods for Optimal Experimental Design of Ill-posed Problems

By

Zhuojun Magnant

The two goals of this thesis are to develop numerical methods for solving large-scale optimal experimental design problems efficiently and to apply optimal experimental design ideas to applications in regularization techniques and geophysics.

The thesis can be divided into three parts. In the fist part, we consider the problem of experimental design for linear ill-posed inverse problems. The minimization of the objective function in the classic A-optimal design is generalized to a Bayes risk minimization with a sparsity constraint. We present efficient algorithms for applications of such designs to large-scale problems. This is done by employing Krylov subspace methods for the solution of a subproblem required to obtain the experiment weights. The performance of the designs and algorithms is illustrated with a one-dimensional magnetotelluric example and an application to two-dimensional super-resolution reconstruction with MRI data.

In the second part, we find the optimal regularization for linear ill-posed problems. We propose an optimal ℓ^2 regularization approach enabling us to obtain inexpensive and good solutions to the inverse problem. In order to reduce the computational cost, several sparsity patterns are added to the regularization operator. Numerical experiments will show that our optimal ℓ^2 regularization approach provides much better results than the traditional Tikhonov regularization.

In the last part of the thesis, we design optimal placement of sources and receivers in a CO_2 injection monitoring. An optimal criteria is proposed based on a target zone and different treatments for placing sources and receivers are discussed.

Numerical Methods for Optimal Experimental Design of Ill-posed Problems

By

Zhuojun Magnant

Advisor: Eldad Haber, Ph.D.
Advisor: James Nagy, Ph.D.

A dissertation submitted to the Faculty of the Graduate School
of Emory University in partial fulfillment
of the requirements of the degree of
Doctor of Philosophy

Department of Mathematics and Computer Science

2011

Acknowledgement

First of all, I would like to give my great admiration and thanks to my advisor Dr. Eldad Haber. It is my great honor to be his student. Dr. Haber introduced a fantasy world of scientific computation to me. Particularly, it is under his direction that I had the opportunity to get so much valuable knowledge and research experience in this field. Without his instruction, I would not have had so much enthusiasm and progress in my graduate study and research. Dr. Haber is not only a prominent research advisor but a nice guide as well. His advice has brought me great benefit both in research and life.

Also, I am so grateful to Dr. James Nagy and Dr. Michele Benzi. They gave me very valuable advice both in class and research. The advice from them is very precious to my future research life. In addition, the delightful atmosphere in their Computational Math group made my five-year graduate study quite pleasant.

I would also like to thank Dr. Ajo-Franklin, who gave me much advice during my internship in the Berkeley National Lab, and Dr. Luis Tenorio, who gave me much help and instruction in writing the A-optimal design paper.

Finally, I will give my great thanks and love for my family. My parents have given me support not only in finance but in spirit as well. My dear husband, Dr. Colton Magnant, has given me constant support not only in our home but also in research by collaborating with me on the CO_2 project.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

This thesis develops numerical methods for optimal experimental designs for large-scale ill-posed inverse problems. Optimal experimental design has broad applications in various scientific areas such as geophysics, medical imaging and biology [14, 16, 35]. In recent years, the study of optimal experimental designs has become far more popular due to these and related applications across all the natural and social sciences.

The term 'experiment' is defined to be a random process that is conducted to support or disprove a hypothesis. It can be conducted to study almost any object such as people, animals, materials, etc. For example, rolling a die to test whether or not the die is fair is a very simple experiment. Other examples include taking an MRI for medical purposes [36], shooting a ray and studying the effects as it passes through certain media [1] and even clinical trials for testing the effects of medications [14].

An optimal experimental design is the design of an experiment, which is optimal with respect to some statistical criterion. As a specific example, researchers at

OPTEC have been studying the optimal characteristics and positions for dampers used to minimize vibrations in footbridges. Unfortunately, the dampers are expensive and the optimal placement is very difficult to determine due to the random nature of the load and patterns in foot traffic. Hence, they would like to design an experiment to determine what types of dampers to use and where to place them. Such optimally placed dampers would provide a balance between the chance of damage to the bridge and the financial cost.

We introduce the theory of optimal experimental design via studying the solution of a linear ill-posed inverse problem of the form

$$Am + \epsilon = b, \tag{1.1}$$

where A is a discretization of some linear operator, b is the data we have observed and ϵ is the noise contained in the data. Since the problem is ill-posed, there is no hope to recover the model m directly from the observed data. Hence, we include the following two elements and solve this optimization problem instead:

$$\hat{m} = \arg\min_{m} (Am - b)^\top W (Am - b) + R(L, m). \tag{1.2}$$

The first element we add is a diagonal matrix $W = \text{diag}\{\, w \,\}$, in which each entry on the diagonal stands for the relative frequency that each experiment is chosen. Suppose the rows of the matrix A represent n different experiments. In order to obtain the best possible solution, we need to use all of the n experiments. However, in most cases, the computation for this kind of problem is very expensive and time consuming, especially when n is large. Also, some experiments contribute

more to the solution than others. So, putting more weight on those entries in w that correspond to the important experiments would be a good idea. On the other hand, if the contributions of some experiments are so small that leaving them out would not harm the result much, then we might consider not conducting those at all. This can be done by setting the corresponding w entries to be 0.

The other element we consider is a regularization functional that adds additional information to the problem in order to introduce stability or to incorporate some a priori information about the desired solution.

It is obvious that, in order to obtain a good estimated solution, we would like to find the optimal W or L. Therefore, there are two questions we want to answer throughout this thesis: what is the best weight matrix W? What is the best regularization matrix L?

With this thought in mind, the two goals of this thesis are to develop numerical methods for solving large-scale optimal experimental design problems efficiently and to apply optimal experimental design ideas to applications in regularization techniques and geophysics.

1.1 Bayesian optimal experimental design

Design of experiments leads to specifying all aspects of an experiment, in other words, controlling the values of variables that are used to describe the experiment. This kind of control may include: choosing the set of experiments to study, deciding the sample size and determining a certain time length to perform the experiment.

Decisions need to be made before gathering information when designing experiments. In general, some information is usually available before the experiments

happen. For example, one usually has prior experience with the type of experiments to be conducted. One may even have collections of training models that provide information about the objects one can expect to recover. Thus, a Bayesian approach [14] seems ideal for experimental design, where such prior information is provided in terms of probability distribution functions.

Bayesian experimental design is based on using Bayesian inference to interpret the observed data from the experiment. In Bayesian inference, observations are used to calculate the probability that a hypothesis is true. It is called 'Bayesian' because of the use of the Bayes' theorem in the calculation process. As a consequence, the goal of the Bayesian design of an experiment is to obtain a high probability of reaching a correct conclusion before conducting any experiment.

We use Bayesian inference as opposed to frequentist inference, which uses only the probability of the observed data, thereby taking no prior probability of the hypothesis into account. The usage of prior information is the major difference between the two ways of inference.

Through the Bayesian approach, both prior knowledge of the parameters to be determined and uncertainties in the observations are taken into account. Hence, optimal decisions can be made based on the study of uncertainty, which can be controlled by properly choosing the values of the random variables of interest. In order to reduce this uncertainty, it is desirable to obtain estimates of the variables with small variance.

1.1.1 A brief review of Bayesian experimental designs

The goal of designing an experiment is to maximize the expected utility that is chosen to reflect purposes of the experiment. The utility is usually defined by measuring the accuracy of the information provided by the experiments. Various experimental design optimality criteria have been developed based on different choices of the utility function.

It is well known that the least squares estimator minimizes the variance of the unbiased estimators. For single variable models, the reciprocal of the variance of an estimator is called the 'Fisher information' of the estimator. Hence, minimizing the variance corresponds to maximizing this 'Fisher information'. However, when the statistical model consists of multiple variables, the mean of the estimator is a vector and its variance becomes a matrix. The inverse of this variance matrix is called the 'information matrix'. Thus, minimizing the variance is equivalent to 'maximizing' the 'information matrix'.

The difficulty of maximizing the information matrix is: how do we determine whether or not a matrix is maximized? Various optimality criteria have been developed to measure the largeness of this information matrix using statistical theory. In practice, the information matrix has been compressed into different real-valued functions so that it can be easily maximized. Some popular criteria are Bayesian A-, C-, D- and E-optimality. This class of criteria is called Bayesian alphabetical optimality.

Bayesian D-optimality

Stone, DeGroot and Bernardo [9, 18, 67, 68] chose a utility function based on the Shannon information. This leads to the Bayesian D-optimality, in which the expected gain in the Shannon information is maximized. In this case, the determinant of the inverse of the information matrix is minimized, which is equivalent to maximizing the determinant of the information matrix,

$$\min \ \ \phi_D = |(A^\top A)^{-1}|, \tag{1.3}$$

where $A^\top A$ is the information matrix.

The Bayesian D-optimality is the most common criterion for computer-generated optimal designs. It aims to maximize the geometric mean of the eigenvalues of $A^\top A$.

Bayesian A-optimality

The Bayesian A-optimality [12, 20, 61] is based on the sum of the variances of the estimated parameters for describing the model. Consequently, it minimizes the sum of the diagonal elements, the trace of the inverse of the information matrix,

$$\min \ \ \phi_A = \text{trace}(A^\top A)^{-1}. \tag{1.4}$$

This criterion results in minimizing the average variance of the estimates.

Bayesian C-optimality

The Bayesian C-optimality [13] is a special case of the A-optimality. It is used for estimating some linear combination of the parameters of interest,

$$\min \ \ \phi_C = c^\top (A^\top A)^{-1} c, \tag{1.5}$$

where $\text{rank}(A) = 1$ and $A = cc^\top$.

Bayesian E-optimality

Another design is the Bayesian E-optimality [14], in which we maximize the smallest eigenvalue of the information matrix. This is another natural approach because the eigenvalue spectrum is also one way to measure the largeness of a matrix,

$$\max \ \ \phi_E = \lambda_{\min}(A^\top A). \tag{1.6}$$

1.1.2 Numerical Challenges

The above designs are difficult to solve for large-scale problems. While developing numerical methods for experimental design of small-scale problems is relatively straightforward, large-scale problems, with a large number of experiments and parameters, present a difficult challenge. Non-trivial matrix-functions and their derivatives need to be evaluated and large-scale constrained optimization problems need to be solved. In addition, since many of the matrices in experimental design of inverse problems are large and dense, one is restricted to perform only matrix-vector products. Furthermore, some of the matrices may be ill-conditioned which

generates further difficulties.

As far as we know, the computation of Bayesian experimental design, especially for large-scale problems, has not been studied extensively. Therefore, it is imperative to develop sophisticated algorithms to solve large scale problems efficiently. Nonetheless, experimental design of large-scale, ill-posed problems is an emerging application whose treatment requires the development of new algorithms.

The goal of optimal experimental design (OED) is to control, in some sense, the quality of experimental results. For well-posed problems this usually means controlling the variance of unbiased least-squares estimates of the parameters of interest. Although the study of OED for well-posed problems is well established [7, 14, 24, 64], its application to ill-posed problems has not yet received much attention. A difficulty that arises in ill-posed problems is that the estimates are biased and this bias may dominate the overall error. Since the bias depends on the parameters to be recovered, its control hinges on prior information about the plausible parameters. For example, properties of the bias can be learned from the training data. It then makes sense to choose a design that minimizes the average mean squared error (under the prior). That is, we look for a design that minimizes the Bayes risk. This is a generalization of the A-optimal experimental design for well-posed problems.

As is well known, the posterior mean minimizes the Bayes risk but, in some cases, the evaluation of its risk may be computationally demanding. One may then choose to minimize the Bayes risk in a class of estimators that are computationally tractable. We develop efficient numerical methods to minimize the Bayes risk over the class of affine estimators. Note that this includes the case Gaussian-prior-Gaussian-likelihood that is often used.

Classical A-optimal designs have been treated in the literature (e.g., [24, 64]) but we do not know of algorithms capable of dealing with large-scale problems. Furthermore, when applying the traditional designs to problems that arise in geoscience and medical imaging, we have found them to be rather limited; new design criteria are clearly needed.

In this thesis, we start by presenting new design criteria that combine sparsity constraints with Bayes risk minimization. Based on different assumptions of the prior information, two different designs were developed to handle ill-posed problems. If the covariance matrix of the models is available, we define the A_B design. Otherwise, the A_π design is suggested. We then develop the optimization and linear algebra techniques required to apply the designs to realistic large-scale problems.

1.2 Optimal design for regularization

In various areas such as medical imaging, geophysics and tomography, inverse problems frequently arise in which parameters of a model can be obtained from some noisy observed data. Many of these problems are ill-posed, that is, either there is no unique solution to the problem or a small perturbation of the data can lead to a large change in the recovered solution. A variety of techniques for obtaining stable solutions have been developed; these techniques are so called regularization techniques, (see for example [10, 23, 54, 75] and references within). Initially, regularization techniques aim to stabilize the problem and incorporate a priori information or assumptions about the desired solution.

1.2.1 Review of regularization

From a Bayesian point of view, many regularization techniques correspond to imposing certain prior information on the model to be recovered. Based on different prior knowledge, people have developed various regularization functions, among which the most commonly used one is the Tikhonov regularization.

For linear ill-posed problems, the Tikhonov regularization takes the regularization functional as $R(x) = \frac{1}{2} \parallel Lx \parallel_2$. The regularization matrix L usually plays a role as a penalty function, such as restrictions for smoothness or bounds on the vector space norm. If the goal is to obtain a solution with smaller norms, then L can be chosen as an identity matrix. If prior information indicates that the solution needs to be smooth, then a discretization of a differential operator is commonly used [73] to force smoothness on the desired solution. Other regularization techniques such as TSVD [39, 40], for which the original ill-conditioned problem is replaced by a well-conditioned rank-deficient problem, have also been considered.

In some cases, especially when the problem size is large and only matrix-vector product is available, iterative regularization methods based on Golub Kahan bidiagonalization [28] and LSQR [32, 39] are very useful. In order to avoid drawbacks of the iterative methods, such as the convergence of small singular values in the Krylov subspace, hybrid methods [32, 59] have become popular. Hybrid methods first use a subspace to reduce the size of the problem while capturing all the large singular vectors. After that, a Tikhonov regularization is applied based on the derived subspace.

Other regularization techniques that use ℓ^1 norms are now commonly used (see for example [19, 21]) but they require considerably more sophisticated algorithms.

Although it is often claimed that the ℓ^1 approach is much better than the quadratic ℓ^2 approach (such as the Tikhonov regularization), in our experience, this may not be the case for many problems, especially if L is chosen appropriately. The question that we believe has not been fully answered is, how well can algorithms that use ℓ^2 regularization perform when the regularization operator is chosen judicially?

1.2.2 Motivation of finding the optimal regularization

Most of the literature focuses on improving the solution by choosing the optimal regularization parameter [38, 75, 76]. However, not much attention has been paid to the choice of the regularization functional. In many cases, the commonly used regularization functionals have limitations. For instance, the differential operator treats all pixels of the image equally. Hence, the spatial features at different locations of the image are ignored. In order to allow different treatments for different spatial features within the image, the goal of this thesis is to develop an optimal ℓ^2 regularization technique in order to obtain inexpensive and good solutions to the inverse problem.

The idea of finding the optimal regularization functional is not new. Lauter and Liero [50] have analyzed the optimal regularization for ill-posed problems by corrections of the data or the operator. Sugiyama and Ogawa [69, 70, 71] developed a method of choosing the optimal regularization functional and regularization parameter from given candidates based on the subspace information criterion. Their method gives an unbiased estimate of the generalization error with finite samples under certain conditions. Given a training set of feasible solutions, Haber and Tenorio [37] proposed a supervised learning approach to estimate the regulariza-

tion functional, which belongs to a family of functions parameterized by a vector parameter.

However, none of the above approaches consider the sparsity structure, which is more practical for real applications, when choosing the optimal regularization functional. In this work, we develop an optimal regularization functional methodology for ill-posed problems that solves an optimization problem based on the MSE of the estimate. In order to accomplish this goal, we assume the availability of a set of examples as our training models.

This methodology can be applied to many practical inverse problems by choosing different types of training models in various applications. Although we only consider the linear case in this thesis, it can be generalized to nonlinear problems. Moreover, the optimization problem we propose here is formulated such that only standard unconstrained optimization techniques are needed.

1.3 Optimal design in CO_2 injection monitoring

With the rapid development of modern technology, pollutants have been collecting in the atmosphere. As a result of this accumulation, the radiation from the sun enters the atmosphere but cannot leave. This is called the Greenhouse Effect. These pollutants are called Greenhouse Gases, which mainly contain carbon dioxide (CO_2), methane, ozone, etc.

The main complication of the Greenhouse Effect is the increase in average temperature of the Earth, which results in climate change and the melting of the polar ice caps, among other terrifying consequences. Therefore, it is imperative to obtain control over the amount of CO_2 near the Earth's surface.

1.3.1 Background of CO_2 injection

CO_2 is mainly produced by burning fossil fuels. Even if a hydrogen-based economy was technically feasible today, we would still have a long way to go before the world would be able to live without carbon-based fossil fuels. Until that time comes, carbon combustion will continue to produce Greenhouse Gases. Thus scientists need to find a way to reroute them.

Many techniques have been proposed for capturing or removing CO_2 from the atmosphere, among which the idea of piping liquefied CO_2 deep under the ocean or underground has received the most interest lately. Depleted oil and gas reservoirs and saline aquifers are generally considered prime candidates for large scale storage of CO_2. Preliminary studies suggest that the underground storage capacity is sufficient for the storage of hundreds of years worth of CO_2 injection, and that potential storage sites exist worldwide.

Basically, CO_2 injection uses compressors to force compressed CO_2 down a long pipe drilled into the underground reservoir. However, as time passes, the liquefied CO_2 will begin to evaporate and potentially escape to the surface, which could result in acidifying the water, affecting the soil chemistry and suffocating animals or people. Therefore, geologic studies need to be carried out to model and monitor the storage condition of the compressed underground CO_2.

1.3.2 Motivation of design for CO_2 injection monitoring

The injection of CO_2 causes a fluid substitution within the pore space. Studies have shown that time-lapse borehole and surface seismic surveys can be used to estimate the location of the injected CO_2 and monitor changes in reservoir properties [3,

31, 44].

The goal of seismic surveys is to record sound waves that travel through the media in the underground reservoir. It generates quantitative maps of variations in fluid saturation or pressure over spatial domains. In particular, crosswell seismic methods have been successfully applied to CO_2 injection monitoring [42, 51]. These methods detect changes in seismic velocity caused by CO_2 injection into reservoirs. Based on the crosswell survey, the CO_2 saturation between the wells is spatially mapped using tomographic imaging.

In order to obtain continuous crosswell seismic data, scientists need to install seismic sources and receivers using production tubing with a geochemical fluid sampling system [17]. The idea is, if the CO_2 saturation and/or plume thickness increases along a given raypath, the travel-time would decrease. This would allow detection with some spatial resolution, especially in the vertical direction. The difference in the travel-times recorded at different depths in an observation well in the close vicinity allows continuous monitoring of the growing CO_2 plume.

Using time-lapse borehole surveys, in stead of inverting for each velocity field, data were inverted based on the change in velocity. In other words, the data used in the tomographic inversion is the travel time difference between the post-injection time and the pre-injection time for each source and receiver pair. By inverting the difference data, some potential errors, such as miscalculation of the source and receiver locations, are minimized or possibly even eliminated [2, 66].

The major problem of the existing CO_2 injection monitoring is, geophysical experiments or surveys are expensive to perform. For example, in order to enable repeatability and control the sampling distance of the monitoring surveys, it is better to deploy permanent sensor arrays in boreholes and on the ocean floor.

However, the cost of permanent built-in sensors is usually extremely high. Furthermore, it is very difficult to reconfigure sensors after they are installed.

For this reason, it is imperative to design an optimal experiment based on the relation between the experimental cost and the geophysical purpose. A good design should be an optimal trade-off between the expected information about the parameters or models of interest and the cost of acquiring such information. In other words, an optimal experimental design should provide maximum information about the target geophysical structure at minimum financial cost.

Since 1995, tremendous work has been done to develop optimal design theories in various applications. Famous examples include determining optimal seismometer locations for locating earthquakes [65], searching optimal placement for sensors in time-lapse traveltime tomography [1, 16], designing optimal sensor geometries for acoustic tomography in detecting underwater conductivity structure [6], etc.

Besides the geophysical purpose and parameterized model, the third component of an experimental design framework is the experimental constraints, which are usually unique for each experiment. The biggest concern and difficulty of CO_2 injection monitoring is spatial limitations of placing sensors because, due to the geophysical complexity, it is usually not possible to place sources and receivers everywhere.

Therefore, the main purpose of this work is to design an optimal placement of sources and receivers in order to obtain the best possible estimated model by minimizing the averaged MSE in the target zone through optimization approaches. The optimal reconstructed model is obtained based on the resulting optimal design thereafter.

1.4 Overview of this thesis

The goal of this thesis is to develop numerical methods to efficiently solve large-scale optimal experimental design problems and to apply our design ideas to various applications such as regularization and CO_2 injection monitoring. The thesis is mainly divided into three parts.

The first part consists of Chapters 2-3, which cover the derivation and numerical methods of two new A-optimal designs.

- Chapter 2:

  In Chapter 2, we give a quick review of the classical A-optimal design for the well-posed case, which serves to review the basic definitions and introduce the sparsity-constrained design. We generalize the design to ill-posed cases where the estimates are biased. We discuss the algorithms to treat the sparsity constrained designs starting with the computation of the design functions and their derivatives. We then proceed to describe approximate solvers for the linear system subproblems that are the bottleneck of the computations. This chapter concludes with a discussion of the numerical optimization methods. We provide two examples: A one-dimensional example inspired by an actual magnetotelluric application and a two-dimensional, super-resolution example where the goal is to determine an optimal number of lower resolution MRI images required to recover a higher resolution one.

- Chapter 3:

  In this Chapter, we present numerical methods for solving the Bayesian E-optimal design. Derivative techniques based on inverse iteration are studied.

The performance of this design is examined on both a 1D problem and a borehole ray tomography experiment.

In the second part of this thesis (Chapters 4-5), an approach to find the optimal ℓ^2 regularization is proposed and some numerical techniques to solve the corresponding optimization problems are discussed.

- Chapter 4:

  In Chapter 4, an optimality criteria for finding the optimal ℓ^2 regularization matrix is developed. We show that this matrix can be obtained by solving an optimization problem that depends on our a priori information. Special derivative techniques are explored to solve large-scale matrix-based optimization problems.

- Chapter 5:

  In Chapter 5 we impose sparsity constraints on the structure of the optimal regularization matrix, such as the local diagonal pattern, the ℓ^1 norm pattern and the Kronecker product pattern. We discuss the numerical solution of the problem and propose an effective algorithm for the recovery of that matrix. We experiment with different regularization matrices based on several sparsity patterns. Their performances are discussed and compared with other commonly used regularizations.

The third part (Chapter 6) is dedicated to optimal design for CO_2 injection monitoring. A framework based on MSE is developed and the crosswell constraints for possible locations of sources and receivers are described in detail. The constrained optimization is solved by DIRECT which is commonly used for finding

the global minimum. Our algorithms were tested on a synthetic geographic tomography example provided by Dr. Jonathan Ajo-Franklin.

Chapter 7 is a summary of this thesis. We conclude the main work in this chapter and give opinions about the future work to extend the approaches proposed.

Chapter 2

Numerical methods for A-optimal design

In the first part of this thesis, we aim at finding the optimal weight matrix W. We start by reviewing the classical framework of experimental design for discrete linear problems.

The data vector $d \in \mathbb{R}^N$ and unknown model $m \in \mathbb{R}^k$ are related via indirect noisy measurements

$$d = Am + \epsilon. \tag{2.1}$$

The rows of the matrix A represent experiments; repeated rows correspond to repeated observations of the same experiment. Hence, if there are n different experiments and each is observed k_i times, then $\sum_{i=1}^{n} k_i = N$ is the total number of experiments. The random noise vector ϵ is assumed to have zero mean and covariance matrix $\sigma^2 I$.

Assuming that each experiment has its own variance, by defining w to be the inverse variance, $w = \frac{1}{\sigma^2}$, we are able to plant the information of variances into the inverse problem as follows:

$$\sqrt{W}Am + \sqrt{W}\epsilon = \sqrt{W}b.$$

This implies that $\sqrt{W}\epsilon \sim N(0, 1)$. Now the problem of finding the best W really becomes the problem of finding the optimal variances for conducting the experiments. We see that, if a certain entry on the diagonal of W is very small, which corresponds to large variance, we say that particular experiment might be very unreliable. On the other hand, if the entry is very large, which corresponds to small variance, then we can trust the corresponding experiment more.

Therefore, the goal of this work is to find the optimal W, thereby telling us which experiments are important for giving optimal estimates that provide reliable conclusions. The optimality is defined by choosing a function that, in some way, measures the performance of the estimate; an optimal experiment is one that minimizes such optimality function subject to appropriate constraints.

2.1 The well-posed case

In classical well-posed linear experimental design [7, 14, 24, 64] the matrix $A^\top A$ is nonsingular and m is typically estimated using least-squares:

$$\widehat{m} = \arg\min \frac{1}{2} \parallel Am - d \parallel^2 .$$

This is equivalent to minimizing

$$S_k(m) = \frac{1}{2}(\bar{d} - A_n m)^\top W(\bar{d} - A_n m),$$

where $\bar{d}_i$ is the average of the k_i observations of the ith experiment, the rows of A_n are the n rows of A corresponding to different experiments and $W = \text{diag}\{k_1/N, ..., k_n/N\}$. Note that $(\sigma^2/N)W^{-1}$ is the covariance matrix of $\bar{d}$. The selection of k_i controls the variances of the observations of different experiments.

Since $\widehat{m}$ is an unbiased estimator, the optimality criteria used to choose k are usually based only on characteristics of the covariance matrix of $\widehat{m}$ given by

$$\Sigma_{\widehat{m}}(W) = \frac{\sigma^2}{N}(A_n^\top W A_n)^{-1}.$$

The proof of the above covariance matrix is not difficult.

$$\begin{aligned}
\Sigma_{\widehat{m}}(W) &= \mathrm{E}(\widehat{m}\widehat{m}^\top) \\
&= \mathrm{E}\left[(A_n^\top W A_n)^{-1} A_n^\top W \bar{d}\bar{d}^\top W A_n (A_n^\top W A_n)^{-1}\right] \\
&= (A_n^\top W A_n)^{-1} A_n^\top W \mathrm{E}(\bar{d}\bar{d}^\top) W A_n (A_n^\top W A_n)^{-1} \\
&= (A_n^\top W A_n)^{-1} A_n^\top W \mathrm{Cov}(\bar{d}) W A_n (A_n^\top W A_n)^{-1} \\
&= \frac{\sigma^2}{N}(A_n^\top W A_n)^{-1}.
\end{aligned}$$

To this end, one defines an optimality function ϕ that scalarizes covariance matrices; different choices of ϕ may define different designs. For example, an A-

optimal design is defined by the function

$$\phi_A(\Sigma_{\widehat{m}}) = \mathrm{E} \parallel m - \widehat{m} \parallel^2 = \mathrm{MSE}(\widehat{m}) = \sigma^2 \mathrm{trace}(\Sigma_{\widehat{m}}),$$

which is proved as follows

$$\begin{aligned}
\phi_A(\Sigma_{\widehat{m}}) &= \mathrm{E}(\parallel (A_n^\top W A_n)^{-1} A_n^\top W \bar{d} - m \parallel_2^2) \\
&= \mathrm{E}(\parallel (A_n^\top W A_n)^{-1} A_n^\top W (A_n m + \bar{\epsilon}) - m \parallel_2^2) \\
&= \mathrm{E}(\parallel (A_n^\top W A_n)^{-1} A_n^\top W \bar{\epsilon} \parallel_2^2) \\
&= \mathrm{E}(\bar{\epsilon}^\top W A_n (A_n^\top W A_n)^{-2} A_n^\top W \bar{\epsilon}).
\end{aligned}$$

By denoting $H = W A_n (A_n^\top W A_n)^{-2} A_n^\top W$, we have

$$\begin{aligned}
\phi_A(\Sigma_{\widehat{m}}) &= \mathrm{E}(\bar{\epsilon}^\top H \bar{\epsilon}) \\
&= \mathrm{E}(\sum_{j=1}^{n} \sum_{i=1}^{n} \bar{\epsilon}_i \bar{\epsilon}_j H_{ij}) \\
&= \mathrm{E}(\sum_{i=1}^{n} \bar{\epsilon}_i^2 H_{ii} + \sum_{i,j=1; i \neq j}^{n} \bar{\epsilon}_i \bar{\epsilon}_j H_{ij}).
\end{aligned}$$

In general, the expected value operator is not multiplicative, i.e. $\mathrm{E}(xy) \neq \mathrm{E}(x)\mathrm{E}(y)$. The lack of multiplicativity gives rise to the study of covariance.

For $i = j$,

$$\mathrm{E}(\sum_{i=1}^{n} \bar{\epsilon}_i^2 H_{ii}) = \sum_{i=1}^{n} \mathrm{E}(\bar{\epsilon}_i^2) H_{ii} = \sum_{i=1}^{n} \mathrm{Cov}(\bar{\epsilon}_i, \bar{\epsilon}_i) H_{ii} = \frac{\sigma^2}{N} \mathrm{trace}(W^{-1} H).$$

For $i \neq j$,

$$\mathrm{E}(\sum_{i,j=1;i\neq j}^{n} \bar{\epsilon}_i \bar{\epsilon}_j H_{ij}) = \sum_{i,j=1;i\neq j}^{n} \mathrm{E}(\bar{\epsilon}_i \bar{\epsilon}_j) H_{ij} = \sum_{i,j=1;i\neq j}^{n} \mathrm{Cov}(\bar{\epsilon}_i, \bar{\epsilon}_j) H_{ij} = 0,$$

since $\mathrm{Cov}(x, y) = 0$ if x and y are independent. Therefore,

$$\begin{aligned} \phi_A\left(\Sigma_{\widehat{m}}\right) &= \mathrm{trace}\left[\frac{\epsilon^2}{N} W^{-1} W A_n (A_n^\top W A_n)^{-2} A_n^\top W\right] \\ &= \frac{\epsilon^2}{N} \mathrm{trace}\left[W^{-1} W A_n (A_n^\top W A_n)^{-2} A_n^\top W\right] \\ &= \sigma^2 \mathrm{trace}(\Sigma_{\widehat{m}}). \end{aligned}$$

There are other popular optimalities as well. For example, the D-optimal design is defined by $\phi_D\left(\Sigma_{\widehat{m}}\right) = \det(\Sigma_{\widehat{m}})$, and if the goal is to estimate the linear functional $c^\top m$ using the estimator $c^\top \widehat{m}$, then a C-design is defined by

$$\phi_C\left(\Sigma_{\widehat{m}}\right) = \mathrm{MSE}(c^\top \widehat{m}) = c^\top \Sigma_{\widehat{m}} c.$$

However, we will only consider the A-design because it has a natural generalization to ill-posed problems.

Given a chosen optimality function ϕ, k is selected by solving the integer optimization problem:

$$\widehat{k} = \arg\min \phi\left[\,\Sigma_{\widehat{m}}(W)\,\right] \quad \text{s.t.} \quad \sum_i k_i = N,\ W = \mathrm{diag}\left\{\, k_i/N \,\right\}.$$

The issue here is that k is a vector of integers. So solving this problem requires integer optimization techniques. That is difficult so an approximation that is

much easier to solve is defined by a 'relaxation' of the problem; one solves for real fractions that sum to one instead of integers summing to N (in the literature this design goes by different names such as 'relaxed design' [11] and 'continuous design' [24, 64]). For example, the A-design is changed to:

$$\widehat{w} = \arg\min \operatorname{trace}\left[\frac{\sigma^2}{N} \left(A_n^\top W A_n \right)^{-1} \right] \quad \text{s.t.} \quad \sum_i w_i = 1,\ w \geq 0, \tag{2.2}$$

with $W = \operatorname{diag}\{ w \}$. That is, the non-negative weights w_i replace k_i/N. The estimated weights are used as follows: the ith experiment is to be conducted $\widehat{k}_i = [N\widehat{w}_i]$ times and m is estimated by minimizing

$$\widehat{S}_k(m) = \frac{1}{2} \left(\widehat{d} - A_n m \right)^\top \widehat{W} \left(\widehat{d} - A_n m \right), \tag{2.3}$$

where $\widehat{d}_i$ is the average of the $\widehat{k}_i$ observations of the ith experiment and $\widehat{W} = \operatorname{diag}\{ \widehat{k}_1/N, ..., \widehat{k}_n/N \}$. Note that the solution of the optimization problem (2.2) provides a proportional allocation of each experiment; the fractions $\widehat{w}_i$ do not depend on the noise level σ or the total number of experiments N. For a given σ, the total N can be chosen to match a target MSE.

In many experimental settings one may be able to control the variance of the experiments by means other than replication; for example, by changing the exposure time of a particular experiment. In this case N is a continuous variable that can be used as a tuning parameter to obtain optimal target variances $\sigma^2/\widehat{k}_i$.

While it is clear that the MSE decreases to zero as $N \to \infty$, in practice there is usually a smallest achievable variance $\sigma^2_{\min}$ for the experiments. In this case the MSE cannot be arbitrarily small without violating this lower bound. We now

define an experimental design that focuses on finding optimal variances subject to a lower bound.

Let σ_i be the target variance of the ith experiment. Set $w_i = 1/\sigma_i^2$ and define d_w to be the $n \times 1$ data vector where the ith experiment is implemented to have variance $1/w_i$ for each i. An estimate of m is obtained by minimizing

$$V_w(m) = \frac{1}{2}\,(\,d_w - A_n m\,)^\top W(\,d_w - A_n m\,), \tag{2.4}$$

where again $W = \mathrm{diag}\,\{w_i\}$. This time the covariance matrix of $\widehat{m}$ is

$$\Sigma_{\widehat{m}}(W) = (\,A_n^\top W A_n\,)^{-1}.$$

The corresponding optimization problem for the A-design is:

$$\widehat{w} = \arg\min \mathrm{trace}\,\big[\,(\,A_n^\top W A_n\,)^{-1}\,\big] \quad \text{s.t.} \quad 0 \le w \le w_{\max}. \tag{2.5}$$

2.2 The sparsity control design

Clearly the solution of (2.5) is $w = w_{\max}$. However, in order to save the experimental cost, one needs to control the total number of different experiments. Such control is particularly important when the cost of conducting the experiments a second time is negligible compared to the cost of doing them the first time. For example, drilling a bore-hole to measure data once has a substantial cost but measuring again in the same bore-hole has a negligible cost. There are also situations where only a few experiments are actually needed or reasonable to realize, so performing all experiments is obviously wasteful. In this case it is better to select a

few different experimental configurations to be conducted a number of times.

This implies that we would like the vector w to be sparse. This is the idea studied in Haber's previous work [35] where they considered controlling the sparsity of w by minimizing

$$\min_{w} \quad \phi\,[\,\Sigma_{\widehat{m}}(W)\,] + \beta\|w\|_0 \quad \text{s.t.} \;\; 0 \le w \le w_{\max},$$

with $\|w\|_0 = \#\{\, w_i \neq 0 \,\}$. They have used this design in the context of ill-posed problems and called it sparsity control design. However, since the problem with $\|w\|_0$ is of combinatorial complexity, it is often approximated using the ℓ^1-norm $\|w\|_1$ instead of $\|w\|_0$ [35]. A sparsity controlled modification of (2.5) is

$$\widehat{w} = \arg\min \quad \text{trace}\,\big[\,(\,A_n^\top W A_n\,)^{-1}\,\big] + \beta \sum w_i \quad \text{s.t.} \quad 0 \le w \le w_{\max}. \tag{2.6}$$

The sparsity of the design is controlled with β. To determine a reasonable value of this parameter one can study the trade-off in MSE reduction as a function of $\|w\|_0$, which is, in turn, controlled by β. The plot of trace$[\,(A_n^\top \widehat{W} A_n)^{-1}\,]$ vs $\|w\|_0$, is often referred to as a Pareto curve. If this curve has an L-shape one can argue that the β corresponding to the corner is a reasonable choice. Note, however, that an optimal w_i may give a variance that is larger than that of the actual instrument. If this happens one needs to make a decision of whether to set w_i to match the instrument variance or to leave out the experiment.

An even more realistic approach to determine w would be to also consider the actual financial cost of the experiments. For example, bore-hole experiments are typically much more expensive than surface experiments, and deeper bore-holes

are more expensive than shallow ones. However, such cost-efficient methods are application dependent and will not be discussed here.

2.3 Formulation for ill-posed problems

We now consider the case where the recovery of m given the indirect noisy data is an ill-posed inverse problem.

2.3.1 The A_B design

To design an experiment one needs to have some information about the class of models to be recovered and about the type of noise to be expected. It is thus reasonable to consider a Bayesian framework where such information is given in terms of distribution functions. We assume that the distribution of d_w given m is Gaussian $N(Am, W^{-1})$ and m is a random vector with prior distribution π. Since we use MSE as the risk function, the idea is then to use the posterior mean $\widehat{m}$, which minimizes the Bayes risk among all estimators, as the Bayes estimate and choose w that minimizes its Bayes risk $\mathrm{E}_\pi\mathrm{MSE}(\widehat{m})$. However, given the large-scale problems we want to address, we reduce the computational cost by choosing $\widehat{m}$ to be the linear (or affine) function of the data d_w that minimizes the Bayes risk. To compute this estimator and its Bayes risk, only the first two moments of the prior π are required:

$$\widehat{m}(w) = (A^\top W A + \Sigma_m^{-1})^{-1}(A^\top W d_w + \Sigma^{-1}\mu), \tag{2.7}$$

where μ and Σ_m are, respectively, the prior mean and covariance matrix. The Bayes risk of $\widehat{m}$ is

$$\phi_{A_B}(W) \;=\; \text{trace}\left[\,(\,A^\top W A + \Sigma_m^{-1}\,)^{-1}\,\right], \tag{2.8}$$

in which we assume that the prior information is given by the covariance matrix.

A modification of the optimal design (2.6) for the ill-posed case is obtained by minimizing the Bayes risk with an ℓ^1 penalty:

$$\widehat{w} \;=\; \arg\min \phi_{A_B}(W) + \beta\|w\|_1 \tag{2.9}$$

$$\text{s.t.}\quad 0 \le w_i \le w_{\max}, \quad W = \text{diag}\,\{\,w_i\,\}. \tag{2.10}$$

To select appropriate values of β, we plot the Pareto curve and make a plot of $\phi_{A_B}(\widehat{W})$ as a function of $\|w\|_0$.

2.3.2 The A_π design

In some cases, it is difficult to compute the A_B design because it requires the inverse of the covariance matrix Σ_m, which will be nontrivial when the problem size is large. Thus some approximations are necessary. We may use the same ideas to choose the optimal linear estimator of a particular type, for example, the Tikhonov estimate $\widehat{m}$ in (2.11):

$$\begin{aligned}\widehat{m}(w) &= \arg\min\,(d_w - A_n m)^\top W (d_w - A_n m) + \alpha\|Lm\|^2 \\ &= (A^\top W A + \alpha L^\top L)^{-1} A^\top W d_w,\end{aligned} \tag{2.11}$$

where α is a regularization parameter, L is a chosen matrix (e.g., a discrete derivative operator) and $W = \text{diag}\{w_1, ..., w_n\}$ with $\text{Var}(d_{wi}) = 1/w_i$.

However, one of the problems we have this time is that the estimator $\widehat{m}$ is biased and thus its covariance matrix does not provide all the information required to assess its performance. Furthermore, its mean squared error (MSE), depends on the unknown m: The MSE of $\widehat{m}$ for a fixed α is

$$\text{MSE}(\widehat{m}) = \alpha^2 \| C(W)^{-1} L^\top L m \|^2 + \text{trace}\left[C(W)^{-2} A^\top W A \right], \tag{2.12}$$

where $C(W) = A^\top W A + \alpha L^\top L$. Hence we have two problems: (i) without knowing m one cannot control the MSE; (ii) the MSE requires an appropriate selection of α which should be adapted to m and the noise level.

The objective is then to choose w to minimize its Bayes risk given by the expectation of the MSE in (2.12) with respect to the prior:

$$\phi_{A_\pi}(w) = \alpha^2 \| C(W)^{-1} L^\top L \, \mu \|^2 + \text{trace}\left[\left(C(W)^{-2} \left[\alpha^2 L^\top L \, \Sigma_m L^\top L + A^\top W A \right] \right) \right]. \tag{2.13}$$

Here the α and L are chosen by the experimenter based on previous experience. Note that the function ϕ_{A_B} is used in the usual Bayesian A-design and ϕ_{A_π} is just the Bayes risk of a Tikhonov estimator with prior moment conditions [14, 24, 64].

We provide the proofs of 2.12 and 2.13 in the following.

$$
\begin{aligned}
\mathrm{MSE}(\widehat{m}) &= \mathrm{E}(\parallel \widehat{m} - m \parallel_2^2) \\
&= \mathrm{E}(\parallel (A^\top W A + \alpha L^\top L)^{-1} A^\top W d_w - m \parallel_2^2) \\
&= \mathrm{E}(\parallel (A^\top W A + \alpha L^\top L)^{-1} A^\top W (Am + \epsilon_w) - m \parallel_2^2) \\
&= \mathrm{E}(\parallel (A^\top W A + \alpha L^\top L)^{-1} A^\top W A m + \\
&\quad (A^\top W A + \alpha L^\top L)^{-1} A^\top W \epsilon_w - m \parallel_2^2) \\
&= \mathrm{E}(\parallel (A^\top W A + \alpha L^\top L)^{-1} (A^\top W A + \alpha L^\top L - \alpha L^\top L) m + \\
&\quad (A^\top W A + \alpha L^\top L)^{-1} A^\top W \epsilon_w - m \parallel_2^2) \\
&= \mathrm{E}(\parallel -\alpha (A^\top W A + \alpha L^\top L)^{-1} L^\top L m + \\
&\quad (A^\top W A + \alpha L^\top L)^{-1} A^\top W \epsilon_w \parallel_2^2) \\
&= \alpha^2 \parallel (A^\top W A + \alpha L^\top L)^{-1} L^\top L m \parallel_2^2 + \\
&\quad \mathrm{E}(\parallel (A^\top W A + \alpha L^\top L)^{-1} A^\top W \epsilon_w \parallel_2^2) \\
&= \alpha^2 \| C(W)^{-1} L^\top L m \|^2 + \mathrm{trace}\left[C(W)^{-2} A^\top W A \right].
\end{aligned}
$$

$$
\begin{aligned}
\phi_{A_\pi}(w) &= \mathrm{E}_\pi \mathrm{MSE}(\widehat{m}) \\
&= \mathrm{E}_\pi \left[\alpha^2 \| \, C(W)^{-1} L^\top L m \, \|^2\right] + \mathrm{trace}\left[\, C(W)^{-2} A^\top W A \,\right] \\
&= \alpha^2 \mathrm{E}_\pi (m^\top L^\top L C(W)^{-2} L^\top L m) + \mathrm{trace}\left[\, C(W)^{-2} A^\top W A \,\right] \\
&= \alpha^2 \mathrm{E}_\pi \left[\mathrm{trace}(m^\top L^\top L C(W)^{-2} L^\top L m)\right] + \mathrm{trace}\left[\, C(W)^{-2} A^\top W A \,\right] \\
&= \alpha^2 \mathrm{E}_\pi \left[\mathrm{trace}(m m^\top L^\top L C(W)^{-2} L^\top L)\right] + \mathrm{trace}\left[\, C(W)^{-2} A^\top W A \,\right] \\
&= \alpha^2 \mathrm{trace}\left[\mathrm{E}_\pi (m m^\top) L^\top L C(W)^{-2} L^\top L\right] + \mathrm{trace}\left[\, C(W)^{-2} A^\top W A \,\right] \\
&= \alpha^2 \mathrm{trace}\left[(\Sigma_m + \mu\mu^\top) L^\top L C(W)^{-2} L^\top L\right] + \mathrm{trace}\left[\, C(W)^{-2} A^\top W A \,\right] \\
&= \alpha^2 \mathrm{trace}(C(W)^{-2} L^\top L \Sigma_m L^\top L) + \\
&\quad \alpha^2 \, \| \, C(W)^{-1} L^\top L \mu \, \|_2^2 + \mathrm{trace}\left[\, C(W)^{-2} A^\top W A \,\right] \\
&= \alpha^2 \| \, C(W)^{-1} L^\top L \, \mu \, \|^2 + \\
&\quad \mathrm{trace}\left[\, \left(C(W)^{-2} \left[\, \alpha^2 \, L^\top L \, \Sigma_m L^\top L + A^\top W A \,\right] \right) \right] .
\end{aligned}
$$

2.3.3 More about the above designs

Notice that if π is Gaussian with $\mu = 0$ and $\Sigma_m = (L^\top L)^{-1}/\alpha$, then (2.7) is equal to (2.11) and (2.7) is the posterior mean, which minimizes the Bayes risk among all functions of the data.

To understand the different nature of the designs, it is important to say a few words about the different interpretation of the results depending on the type of chosen prior distribution. In the well-posed case the weights w control the trace of the covariance matrix of the estimator that the experimenter will use. Similarly, in the ill-posed case with a Gaussian prior, the weights control the trace of the covariance matrix of the posterior distribution that the experimenter will obtain. When the prior is not Gaussian and the weights are based on the affine estimator

with smallest risk, then this risk only provides a bound for the expected value of the trace of the posterior covariance matrix. More precisely, let $\widehat{m}$ be any function of the data (estimator) and $\Sigma_{m|y}$ the posterior covariance matrix of m given y. It is easy to see that

$$\mathrm{E}\|\widehat{m} - m\|^2 = \mathrm{E}\|\widehat{m} - \mathrm{E}(m|y)\|^2 + \mathrm{E}\,\mathrm{trace}\,\Sigma_{m|y}.$$

Consider now the case of a Gaussian prior. Let $\widehat{m}_a$ and $\widehat{m}_L$ be, respectively, the affine estimator that minimizes the risk and the Tikhonov estimator (2.11). Then $\widehat{m}_a$ is the posterior mean $\mathrm{E}(m|y)$, the trace of $\Sigma_{m|y}$ does not depend on y (so $\mathrm{E}\,\mathrm{trace}\,\Sigma_{m|y} = \mathrm{trace}\,\Sigma_{m|y}$) and

$$\mathrm{trace}\,\Sigma_{m|y} = \mathrm{E}\|\widehat{m}_L - m\|^2 - \mathrm{E}\|\widehat{m}_L - \widehat{m}_a\|^2 \leq \mathrm{E}\|\widehat{m}_L - m\|^2.$$

We see that the risk of $\widehat{m}_L$ provides a bound for the trace of the posterior matrix but it could be a poor bound if $\widehat{m}_L$ and $\widehat{m}_a$ are not close enough. In the general case of non-Gaussian priors, the trace of $\Sigma_{m|y}$ may depend on y and all we may be able to say is that

$$\mathrm{E}\,\mathrm{trace}\,\Sigma_{m|y} \leq \mathrm{E}\|\widehat{m}_a - m\|^2 \leq \mathrm{E}\|\widehat{m}_L - m\|^2.$$

Thus, the risks of $\widehat{m}_a$ and $\widehat{m}_L$ only constrain an average value of $\mathrm{trace}\,\Sigma_{m|y}$ but, if the sampling variability of the posterior covariance matrix is large, then neither $\widehat{m}_a$ nor $\widehat{m}_L$ may provide a good control of the posterior distribution of the actual experiment.

2.4 Numerical optimization of the A_B and A_π designs

The A_B and A_π designs are difficult to compute for large-scale problems as they involve the inverse of large matrices. In particular, it is difficult to evaluate the objective function ϕ and its derivatives. We now discuss methods to do this efficiently.

2.4.1 Evaluating the traces in the objective function

The definition of the objective function ϕ includes the traces of large dense matrices. Efficient approximations of such traces can be obtained using stochastic trace estimators [4, 29, 45]. If the vectors U_1,...,U_r are independent and each with independent entries taking the values 1 and -1 with equal probability, then the trace estimator

$$\widehat{T}_r(H) = \frac{1}{r}\sum_{i=1}^{r} U_i^\top H U_i$$

is an unbiased estimator of trace(H) with variance

$$\mathrm{Var}(\,\widehat{T}_r(H)\,) = \frac{2[\,\mathrm{trace}(H^2) - \|\,h\,\|^2\,]}{r}, \tag{2.14}$$

where $h = \mathrm{diag}\,(H)$. To estimate the quality of this estimator we would like to evaluate the relative error, that is we would like to estimate the variance (2.14). Let $\lambda_1, \ldots, \lambda_n$ be the eigenvalues of H. Also let

$$\bar{\lambda} = \frac{1}{n}\sum_{i} \lambda_i$$

.

We have that

$$\text{trace}(H^2) = \sum \lambda_i^2 \qquad \text{trace}(H)^2 = \left(\sum \lambda_i\right)^2 = n^2\bar{\lambda}^2$$

. We can therefore write

$$\frac{\text{trace}(H^2) - \| h \|^2}{\text{trace}(H)^2} = \frac{\sum \lambda_i^2 - \| h \|^2}{n^2\bar{\lambda}^2}.$$

To have an upper bound, we need to bound $\|h\|^2$ from below. We have that $h = \text{diag}\,(H)$ and $\text{trace}(H) = \sum \lambda_i = n\bar{\lambda} = \sum h_i$. We now seek the smallest $\|h\|^2$ subject to the above equality constraint. This can be easily achieved by minimizing

$$\min \ \|h\|^2 \quad \text{s.t} \ \ n\bar{\lambda} = \sum h_i.$$

The solution to this problem is obviously $h_i = \bar{\lambda}$. We can therefore write

$$\frac{\text{trace}(H^2) - \| h \|^2}{\text{trace}(H)^2} \leq \frac{\sum \lambda_i^2 - n^2\bar{\lambda}^2}{n^2\bar{\lambda}^2}.$$

This shows that the relative variance of the trace estimate will be small if the scatter of the eigenvalues is small compared to their mean. In our applications, H stems from an ill-posed problem. If no regularization is used, then the eigenvalues are bounded from above and tend to cluster at 0 as $n \to \infty$. For a regularized version in standard form, the eigenvalues cluster around the regularization parameter α. This implies that the total spread of the eigenvalues is typically small, which explains why stochastic trace estimators are so successful in ill-posed problems. It is also

clear that, for well posed problems where $\lambda_{\max}/\lambda_{\min}$ is close to 1, such estimators can be very effective.

Using randomized trace estimators with $r = 1$, the ϕ_{A_B} and ϕ_{A_π} objective functions are replaced by the following approximations

$$\phi_{A_B}(w) = v^\top (A^\top W A + \Sigma_m^{-1})^{-1} v \tag{2.15}$$

and

$$\phi_{A_\pi}(w) = \alpha^2 \| C(W)^{-1} L^\top L \mu \|^2 + v^\top \left(C(W)^{-2} \left[\alpha^2 L^\top L \Sigma_m L^\top L + A^\top W A \right] \right) v, \tag{2.16}$$

where v is a random vector whose entries take the values ± 1 with equal probability. Note that the computation of $C(W)^{-1}$ is not needed explicitly. Instead, we need to evaluate the action of $C(W)^{-1}$ on a vector, say q, and this can be done by solving the linear system $Cz = q$, which can be done by conjugate gradient. We discuss this step at length in Section 2.4.3.

2.4.2 Evaluating the derivatives

We start with the derivatives for the A_B design:

$$\nabla_w \phi_{A_B} = \nabla_w \left(v^\top (A^\top W A + \Sigma_m^{-1})^{-1} v \right)$$

and setting

$$z = (A^\top W A + \Sigma_m^{-1})^{-1} v \quad \leftrightarrow \quad (A^\top W A + \Sigma_m^{-1}) z - v = 0$$

yields

$$A^\top \operatorname{diag}(Az) + (A^\top W A + \Sigma_m^{-1})\nabla_w z = 0.$$

Therefore

$$\nabla_w z = -(\, A^\top W A + \Sigma_m^{-1}\,)^{-1} A^\top \operatorname{diag}(Az),$$

which implies

$$\nabla_w \phi_{A_B} = -\operatorname{diag}(Az)\, A(\, A^\top W A + \Sigma_m^{-1}\,)^{-1} v = -\operatorname{diag}(Az) Az = -Az \odot Az. \quad (2.17)$$

The computation of the objective function and its derivatives for the A_B design is summarized in Algorithm 1

Algorithm 1 Objective function and gradient for A_B design

(1) Solve the system $(A^\top W A + \Sigma_m^{-1})z = v$
(2) Set $\phi_{A_B} = v^\top z$ and $\nabla_w \phi_{A_B} = -Az \odot Az$

An interesting property of the A_B design is that the computation of its gradient is almost 'free' compared to the computation of the objective function. After the vector z is obtained, which requires a substantial amount of work, the objective function is computed by a simple inner product and its derivative requires only one more matrix vector product with the matrix A.

We now derive similar expressions for the A_π design. We rewrite (2.16) in a symmetric form

$$\begin{aligned} \phi_{A_\pi}(w) &= \alpha^2 \mu^\top L^\top L C(W)^{-2} L^\top L \mu \\ &+ v^\top C(W)^{-1} \left(A^\top W A + \alpha^2\, L^\top L\, \Sigma_m L^\top L \right) C(W)^{-1} v. \end{aligned}$$

Defining as above $z = C(W)^{-1}v$, we can write $\nabla_w z = -C(w)^{-1}A^\top \mathrm{diag}(Az)$. Similarly, we define $y = C(W)^{-1}L^\top L\mu$ and write $\nabla_w y = -C(w)^{-1}A^\top \mathrm{diag}(Ay)$. We thus obtain

$$\begin{aligned}\nabla_w \phi_{A_\pi}(w) = &-2\alpha^2 \mathrm{diag}(Ay)AC(W)^{-1}y + \mathrm{diag}\,(Az)Az \\ &- 2\,\mathrm{diag}(Az)\,AC(W)^{-1}\left(A^\top WA + \alpha^2\, L^\top L\,\Sigma_m L^\top L\right)z. \end{aligned} \tag{2.18}$$

2.4.3 Solving the linear systems

Regardless of the chosen optimization criteria, a key component in the solution of the problem is the solution of the system

$$(\,A^\top WA + \Sigma_m^{-1}\,)m = A^\top Wb, \tag{2.19}$$

where, in the A_π design, $\Sigma_m^{-1} = \alpha L^\top L$. This system is large and dense and therefore, iterative methods are typically used for its solution. This system needs to be solved every time w is updated, that is, whenever we evaluate the objective function or its derivatives. Therefore, we need a method that enables us to quickly evaluate m given a new w. We now use the SVD of A to better understand the properties of the solution, and then, we propose an approximation of the SVD using Lanczos decomposition.

SVD analysis

To understand how to solve the large linear system quickly, we first turn to a simple SVD analysis. Assume first that $\Sigma_m^{-1} = I$ (this case is referred to as 'standard form'). The more general case can be transformed to this form [39].

Write the SVD of A as $A = U\Lambda V^\top$, where U is an orthogonal $n \times n$ matrix, V is a $k \times n$ matrix with orthonormal columns, and Λ is diagonal $n \times n$ with singular values $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$. Using this decomposition the linear system is rewritten as

$$(\Lambda U^\top W U \Lambda + I)\xi = \Lambda U^\top W b, \quad \xi = V^\top m. \tag{2.20}$$

We now start by reviewing the 'usual' unweighted case, which is commonly used to analyze the solution of the inverse problem (see, for example, [39]). When $W = \mu I$ we obtain $\Lambda U^\top W U \Lambda = \mu \Lambda^2$ and the system decouples:

$$(\mu\Lambda^2 + I)\xi = \mu \Lambda U^\top b, \qquad \xi_j = \frac{\mu\lambda_j}{\mu\lambda_j^2 + 1}\, u_j^\top b.$$

The ratio $\mu\lambda_j/(\mu\lambda_j^2 + 1)$ decreases to $1/\lambda$ for large λ and is close to zero for small λ. It stands to reason that an approximate solution can be obtained by truncating the singular values:

$$\xi_j = \begin{cases} \frac{1}{\lambda_j}\, u_j^\top b & \lambda_j \gg 0 \\ 0 & \text{otherwise.} \end{cases}$$

This is the truncated SVD (TSVD) solution. The problem is that it may be difficult to determine where to truncate the singular values. One option is to use hybrid methods that were proposed first in [59] and have been studied substantially in recent years. In this case the solution will be 0 only when its corresponding singular value is much smaller than the largest singular value λ_1. That is, we keep the first s singular values such that λ_s is less than some tolerance times the largest singular

value and truncate everything after that. Hence,

$$\xi_j = \begin{cases} \frac{\mu\lambda_j}{\mu\lambda_j^2+1}\, u_j^\top b & j = 1, \cdots, s \quad \text{where} \quad \lambda_s \leq \text{tol}\lambda_1 \\ 0 & \text{otherwise.} \end{cases}$$

This simple idea reduces the dimensionality of the problem and makes the solution less sensitive to the selection of a truncation level.

We now show that a similar analysis can be used in our case. If W is not a multiple of the identity, then

$$\Lambda U^\top W U \Lambda = \begin{pmatrix} \lambda_1^2(w \odot u_1)^\top u_1 & \lambda_1\lambda_2(w \odot u_1)^\top u_2 & \cdot & \cdot & \lambda_1\lambda_n(w \odot u_1)^\top u_n \\ \lambda_1\lambda_2(w \odot u_1)^\top u_2 & \lambda_2^2(w \odot u_2)^\top u_2 & \cdot & \cdot & \lambda_2\lambda_n(w \odot u_2)^\top u_n \\ & \cdot & & & \\ & \cdot & & & \\ \lambda_1\lambda_n(w \odot u_1)^\top u_n & \lambda_2\lambda_n(w \odot u_2)^\top u_n & \cdot & \cdot & \lambda_n^2(w \odot u_n)^\top u_n \end{pmatrix},$$

where $\odot$ represents the Hadamard product, for which the product is taken entry-wise. Thus, the system does not decouple as before. Nevertheless, typically $\lambda_\ell \ll \lambda_1$ for ℓ greater than some index s, and since in addition w is bounded, it makes sense to use a TSVD as before; we obtain a reduced $s \times s$ dense system $\Lambda_s U_s^\top W U_s \Lambda_s$ where $U_s = \begin{pmatrix} u_1, \ldots, u_s \end{pmatrix}$ and $\Lambda_s = \text{diag}\,(\lambda_1, \ldots, \lambda_s)$. If s is not too large (say, up to a few thousand) then it is possible to use direct methods to quickly solve the system for different w's. The algorithm is described in Algorithm 2.

The important observation is that the SVD is computed only once at the beginning of the optimization. We then use the decomposition to solve the optimization problem at a negligible cost. Thus, if the computation of the SVD is not pro-

Algorithm 2 SVD approximate solution of the system $(A^\top W A + I)m = A^\top W b$

(1) Compute the SVD: $A = U\Lambda V^\top$
(2) Choose an index s to truncate the SVD, $\lambda_s \leq \mathrm{tol}\,\lambda_1$
(3) Solve the systems $(\Lambda_s U_s^\top W U_s \Lambda_s + I)\xi = \Lambda_s U_s^\top W b$
(4) Set $m = V_s \xi$

hibitively expensive, then it is possible to quickly solve the problem.

Approximation using Lanczos decomposition

The computation of the SVD is typically not practical for large-scale problems; it has a computational complexity of $\mathcal{O}(n^2 k)$ and large storage requirements. We therefore turn to approximating the SVD using Lanczos methods.

Lanczos bidiagonalization has been extensively studied in the context of inverse problems [32, 38, 39, 55]. In its simplest formulation, s iterations of the Lanczos process are computed yielding an approximate decomposition

$$A \approx U_s B_s V_s^\top, \tag{2.21}$$

where $U_s = [u_1, \ldots, u_s]$ is an $n \times s$ matrix and $V_s = [v_1, \ldots, v_{s+1}]$ is a $k \times (s+1)$ matrix, and both have orthonormal columns. The matrix B_s is an $s \times (s+1)$ bidiagonal matrix. The computation of the Lanczos decomposition can be done with or without re-orthogonalization of the vectors. Without re-orthogonalization the vectors tend to loose the orthogonality, especially for large s. This motivated the study of a number of re-orthogonalization techniques [39]. An important property of the decomposition (2.21) is that the singular values of B_s approximate the singular values of A. In numerical experiments, it has been widely observed that the large singular values of A are approximated first, yielding an approximation to

the truncated SVD solution. This observation has motivated research on iterative regularization as well as hybrid regularization techniques [40].

As we have done with the SVD, using the truncated bidiagonalization we obtain

$$(V_s B_s^\top U_s^\top W U_s B_s V_s^\top + I)m = V_s B_s^\top U_s^\top W b, \tag{2.22}$$

and multiplying both sides of (2.22) by $V_s^\top$ from the left and setting $\xi = V_s^\top m$ yields

$$(B_s^\top U_s^\top W U_s B_s + I)\xi = B_s^\top U_s^\top W b. \tag{2.23}$$

Again, the system (2.23) is a small $s \times s$ dense system and its solution can be obtained quickly using direct methods. The algorithm for solving the system using the Lanczos process is described in Algorithm 3. We have experimented using

Algorithm 3 Lanczos approximate solution of the system $(A^\top W A + I)m = A^\top W b$

(1) Compute the Lanczos decomposition of A with starting vector b
(2) For each step compute the SVD of B_j, stopping at step s when $\lambda_s(B_s) \leq \text{tol}\, \lambda_1(B_s)$
(3) Solve the system $(B_s^\top U_s^\top W U_s B_s + I)\xi = B_s^\top U_s^\top W b$
(4) Set $m = V_s \xi$

the Lanczos process with and without re-orthogonalization and found that re-orthogonalization did not yield any benefit. We believe this is because, unlike the case $W = I$, we do not require $U_s^\top U_s = I$ and, anyway, the product $U_s^\top W U_s$ is computed when obtaining the solution.

2.4.4 Numerical optimization

Given the tools described in the last section, we can now use standard box-constrained optimization to solve the problem. We start by solving the optimization problem

$$\begin{aligned} \min \quad & \phi_\beta = \phi(w) + \beta\|w\|_1 \\ \text{s.t} \quad & 0 \leq w \leq w_{\max}. \end{aligned} \tag{2.24}$$

To solve the optimization problem (2.24) we use projected steepest descent as this method can easily deal with the non-differentiability of the ℓ^1 norm. For further discussion see [8, 25].

We also consider the ℓ^0 penalty $\beta\|w\|_0$ as it is expected to provide the sparsest solution. To approximate $\|w\|_0$ and the sparsest solution, we use an approximation described in [11].

We divide all experiments into two sets: $\mathcal{I}_0$ and $\mathcal{I}_A$. The set $\mathcal{I}_0$ contains all the indices for the zero entries of the solution to (2.24), namely $w_{\mathcal{I}} = 0$ and $\mathcal{I}_A$ contains the rest. Assuming that $\mathcal{I}_0$ is known a priori, which can be done by solving the ℓ^1 approach, the ℓ^0 solution could be obtained by solving this un-regularized optimization problem only on the set $\mathcal{I}_A$. We do not need any regularization term here because the zero set is already known. It has been shown that the approximated ℓ^0 solution may improve upon the ℓ^1 solution.

The ℓ^0 penalty is approximated by the solution of the problem

$$\begin{aligned} \min_{\mathcal{I}_A} \quad & \phi_\beta = \phi(w) && (2.25) \\ \text{s.t} \quad & w_{\mathcal{I}_0} = 0, \quad w_{\mathcal{I}_A} \geq 0 \\ & 0 \leq w_{\mathcal{I}_A} \leq w_{\max}. && (2.26) \end{aligned}$$

As we have shown in [35], an important aspect of the optimization is that to study the design space we need to solve (2.24) for different values of β that lead to different sparsity structure and thus provide information about the design cost as a function of the number of experiments. Here we have used a simple continuation strategy to achieve this goal [34].

2.5 Numerical experiments

We illustrate the performance of our algorithms using a small-scale 1D problem, where exact quantities can be easily computed and a realistic, large-scale super-resolution inverse problem.

2.5.1 An ill-posed 1D magnetotelluric example

The data are modeled as

$$d_j = \int_0^L \exp(-\alpha_j x) \cos(\gamma_j x)\, m(x)\, dx + \epsilon_j \quad j = 1, \cdots, n. \tag{2.27}$$

This kernel mimics the linearized 1D magnetotelluric experiment [58, 77]. The parameters γ_j correspond to the recording frequencies and the α_j depend on the

frequencies and the background conductivity. The objective is to select optimal values of these parameters to best evaluate a 1D conductivity structure by solving the following inverse problem

$$Am + \epsilon = d,$$

where A is a matrix generated by the kernel, m is the conductivity and d is the data we have observed.

The function m is discretized using 256 points. Figure 2.1 shows an example of the magnetotelluric kernel that is used in our simulations. The values of α_j are equally spaced between 0 and 3, namely $\alpha_j = 3j/100$ for $0 \leq j \leq 100$ and the values of γ_j are equally spaced between 1 and 10: $\gamma_j = 1 + 9j/100$. Each row in the kernel matrix represents one pair (α_j, γ_j), which gives a total of 100 different choices as our bank of experiments. The objective is to choose a subset of these 100 experiments to best estimate m.

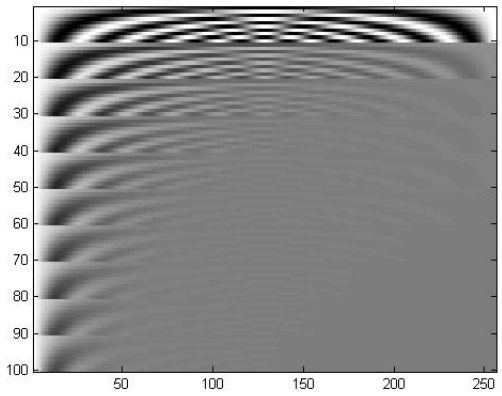

Figure 2.1: Geometry of the magnetotelluric kernel. The model is discretized using 256 points. Each row of the kernel represents one pair (α_j, γ_j).

The noise variance σ^2 is chosen for a SNR ≈ 10. The true test model is a

realization from a Gaussian $N(0, \Sigma)$ where the covariance matrix Σ is defined as $\Sigma_{i,j} = \exp\left(-(x_i - x_j)^2/2\tau^2\right)$ with $x_i, x_j \in (0, 2\pi)$. For simplicity, in this example, μ is set to be 0. The parameter τ^2 is chosen so that the correlation between $m(x_i)$ and $m(x_j)$ is less than 50% for $|x_i - x_j| > 1$. We set $w_{\max} = 10^2/\sigma^2$. The optimization is done using steepest descent.

Notice that it is possible for the optimization to yield $w_i < 1/\sigma^2$, indicating that the ith experiment is to be conducted with a variance larger than that of the instrument's noise. To avoid this, the w_i are thresholded as follows: tolerance values $w_{\mathrm{t}} < w_{\min}$ are chosen and w_i is set to zero if $w_i \leq w_{\mathrm{t}}$, basically, we will not conduct that experiment and to $w_{\min}$ if $w_{\mathrm{t}} < w_i \leq w_{\min}$. For the example we have used $w_{\mathrm{t}} = 10^{-5}/\sigma^2$ and $w_{\min} = 1/\sigma^2$. From now on it will be understood that w has been thresholded in this fashion.

The A_B design

We start by illustrating the controlled sparsity of the A_B design. Figure 2.2 shows the risk as a function of $\|w\|_0$. As is shown here, the risk decreases rapidly before the corner of the L-curve at $\|w\|_0 = 25$ and more slowly thereafter. Hence, this corner of the L-curve suggests an optimal compromise between the image reconstruction quality and the experimental cost. This implies that, even if you do more experiments, you are not going to improve the result much.

The right panel in Figure 2.2 depicts all the w vectors obtained from the experiments. Each column represents one of the 100 different experiments and the rows correspond to the different values of the risk for the chosen values of β used to draw the L-curve. The color shows the values of w_i for the corresponding experiment and risk. The figure shows which experiments should be conducted to

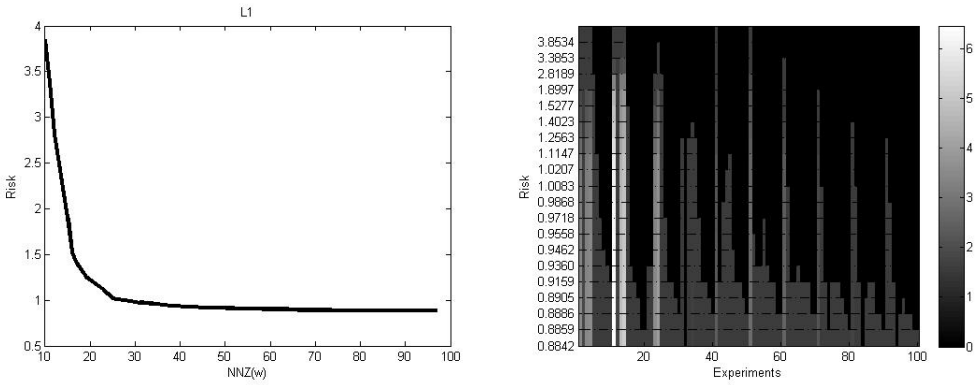

Figure 2.2: The left panel shows the risk as a function of sparsity $\text{nnz}(w) = \| w \|_0$ of the optimal w obtained with the A_B design. The columns and rows in the image on the right correspond, respectively, to different experiments and different values of the risk. The color refers to the values of the w_i.

achieve the corresponding risk; one can use the plot to decide if the reduction in risk is worth the increase in the number of experiments. This figure tells us not only the number of experiments but which ones they are as well. This information can be quite helpful to practitioners.

Figure 2.3 shows examples of model estimates using the optimal w_i determined by the corner in the L-curve. The figure shows the true model as well as three reconstructions from three different noise realizations. We see good reconstruction results using only the 25 selected experiments but we also see large sampling variability for $x > 3$.

To show the advantage of our optimal design, Figure 2.4 displays the model estimates obtained using all 100 experiments and also the estimates based on 25 equally-spaced naively chosen experiments. Note also that the sampling variability for $x > 3$ seems even larger. The figures show that the A_B design provides the best results. Compare, for example, the amplitudes of the last two peaks of the curve.

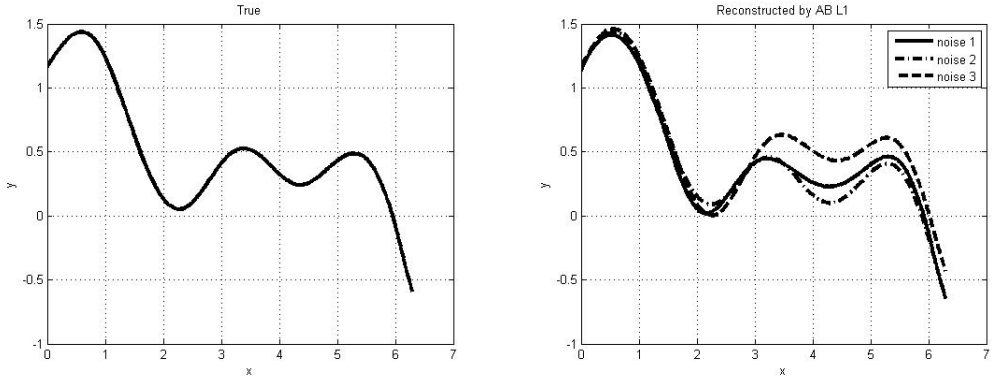

Figure 2.3: The left panel shows the true test model. The right panel shows examples of model estimates obtained with the A_B design using 25 optimal experiments for three different noise realizations.

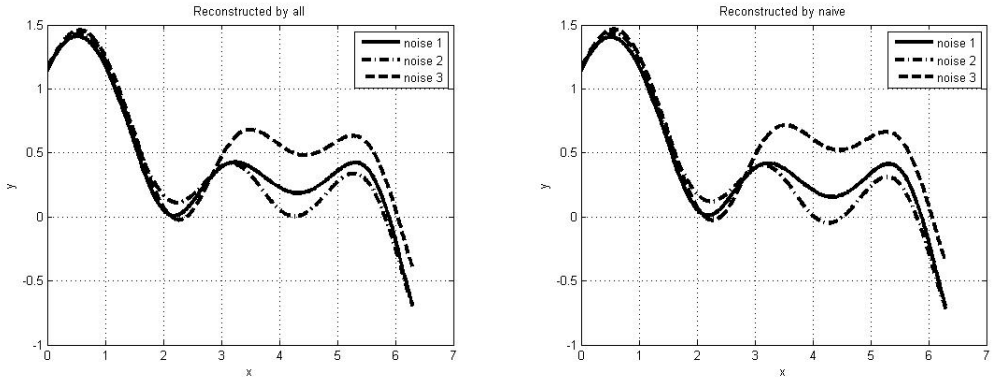

Figure 2.4: Model estimates obtained using all 100 experiments (left) and the equally-spaced naive design (right).

Table 2.1 shows the relative errors ($\|m_{\text{true}} - \widehat{m}\|/\|m_{\text{true}}\|$) of the reconstructions. Note that the estimates obtained using all 100 experiments are worse than those using only the 25 optimal experiments. This is because the full design does not use weights to compute the estimates. The optimal design selects a subset of the experiments and assigns to them optimal weights (variances).

Table 2.1: Relative errors of the model reconstructions obtained with the A_B design

design	error (noise 1)	error (noise 2)	error (noise 3)
A_B	7.75%	11.91%	14.81%
all	10.29%	20.12%	19.26%
naive	11.84%	23.94%	22.88%

The A_π design

For the A_π design we choose the matrix L to be the discrete 1D Laplacian operator. The parameter α is chosen by trial and error so that given all the data we obtain the best possible recovery error.

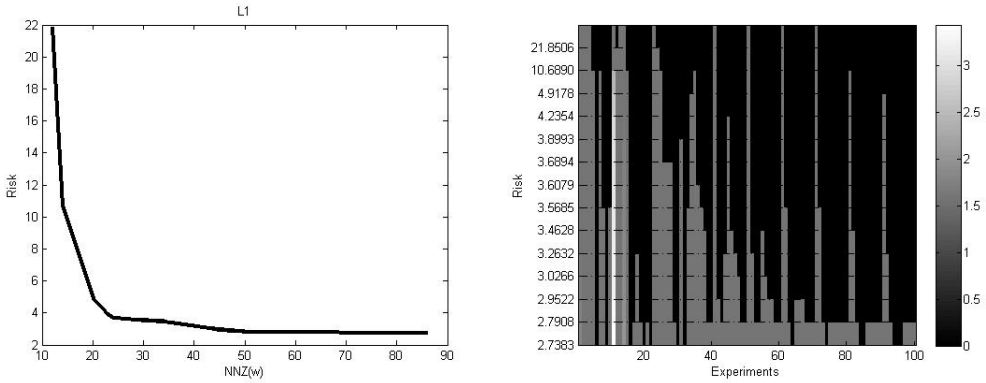

Figure 2.5: The left panel shows the risk as a function of sparsity $\text{nnz}(w) = \| w \|_0$ of the optimal w obtained with the A_π design. The right panel is similar to that in Figure 2.2 but for the A_π design using the ℓ^1-optimization.

In Figure 2.5, we choose the corner at $\|w\|_0 = 24$. This defines the optimal solution used in the example. Figure 2.5 shows the equivalent of Figure 2.2 for the A_π design.

Figure 2.6 shows the model estimates for three different noise realizations obtained using the ℓ^1 approach. Figure 2.7 shows the estimates obtained using all 100 experiments and using 24 equally-spaced naively chosen experiments. The relative

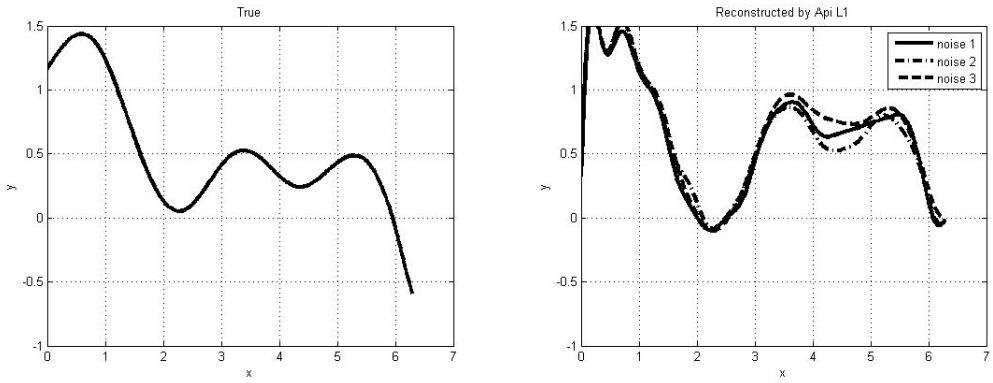

Figure 2.6: The true test model (left) and the estimated models obtained with ℓ^1-A_π design using the 24 optimal experiments with three different noise realizations (right).

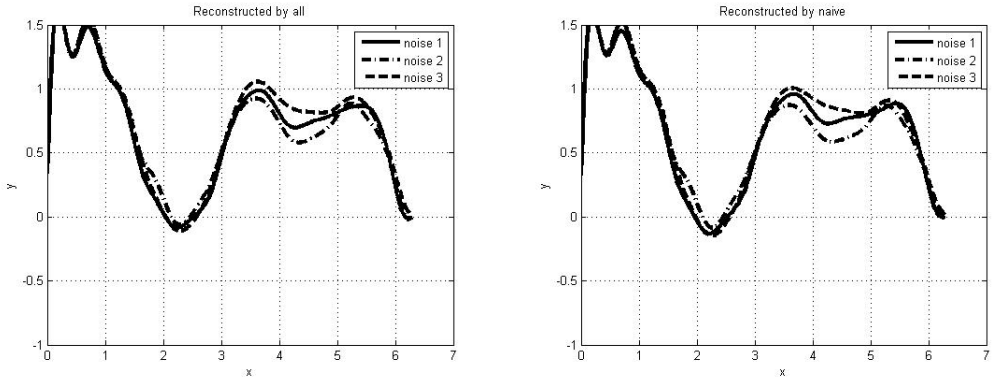

Figure 2.7: Model estimates obtained with the A_π design using all 100 experiments (left) and the equally-spaced naive design (right).

errors of the estimates are shown in Table 2.2. The optimal design still leads to smaller errors than the full and naive designs but the results are not as good as those obtained with the A_B design. This is to be expected as the the A_B design uses the Bayes estimate while the regularization for A_π design is restrained to a particular type. We omit the rest of the results for the ℓ^0 approach as they are very similar to those obtained with the ℓ^1 penalty.

Table 2.2: Relative errors of the model estimates obtained with the A_π (ℓ^1) design

design	error (noise 1)	error (noise 2)	error (noise 3)
A_π ℓ^1	39.57%	33.64%	45.98%
all	47.00%	41.07%	53.60%
naive	47.00%	39.60%	53.87%

2.5.2 Super-resolution

In this section we consider a large-scale super-resolution problem that has a wide range of applications [15, 22]. Super-resolution methods are techniques to enhance the resolution of an imaging system, in particular, to construct a high-resolution image by combining a set of lower resolution images.

The data are modeled as

$$d_j = K\,S(u_j)m + \epsilon_j, \quad j = 1, \cdots, k; \tag{2.28}$$

where d_j represent the low resolution images that have been collected, K is the sparse matrix approximating the averaging process, u_j are the relative displacements among the low resolution images and $S(u_j)$ is a sparse matrix representing the bilinear interpolation operation that connects the point in the displaced image to the four pixel values in the reference image that surround it (see [15] for details). In order to obtain high-quality, high-resolution images, it is desirable to have many low resolution ones. Nonetheless, generating and/or using many low resolution images can be very costly. Thus, we would like to find an optimal subset of low resolution images to generate one of high resolution and satisfactory quality.

We apply our algorithms to one of the MRI examples provided by J. Orchard [60]. In order to reconstruct this 64×64 image, 100 slightly different 32×32 low

resolution images are generated. For displacements, both shifting and rotation are considered. Figure 2.8 shows one of the 100 low resolution images that we generated together with the original high resolution image.

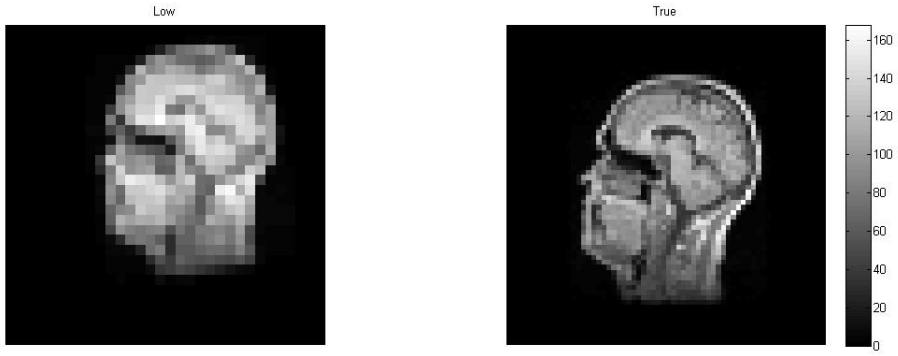

Figure 2.8: One of the 100 low resolution images (left) and the true high resolution image (right).

We apply the discretization of the gradient operator as a smoothing matrix to obtain the solution of the inverse problem. Since this is a realistic problem, we do not have the true covariance matrix of the MRI images but do have other MRI images that can be used to estimate it assuming that they are iid random realizations of the same stochastic process. We use the method described in [27]. For the prior mean μ we have used the sample mean of all MRI images.

Figure 2.9 shows the Pareto curves for the A_B and A_π optimal designs. This time the curves are not as much 'L-like' as in the 1D example. We have chosen the points corresponding, respectively, to $\|w\|_0 = 31$ and $\|w\|_0 = 30$ for the the A_B and A_π designs. Hence, we use 31 low resolution images in the A_B design and 30 with the A_π design to reconstruct a high resolution image. Figures 2.10 and 2.11 show the optimal image reconstructions for the A_B and A_π designs as well as the one obtained using 100 low resolution images.

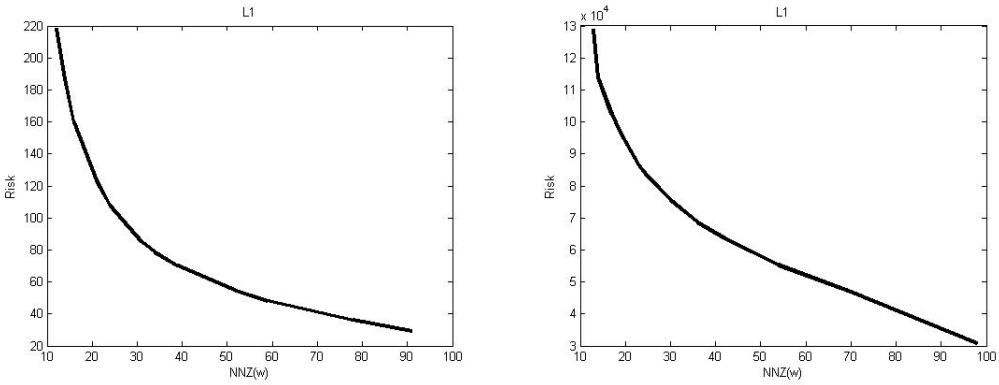

Figure 2.9: Risk as a function of sparsity $\|w\|_0$ of the optimal w obtained with the A_B design (left) and with the A_π design (right).

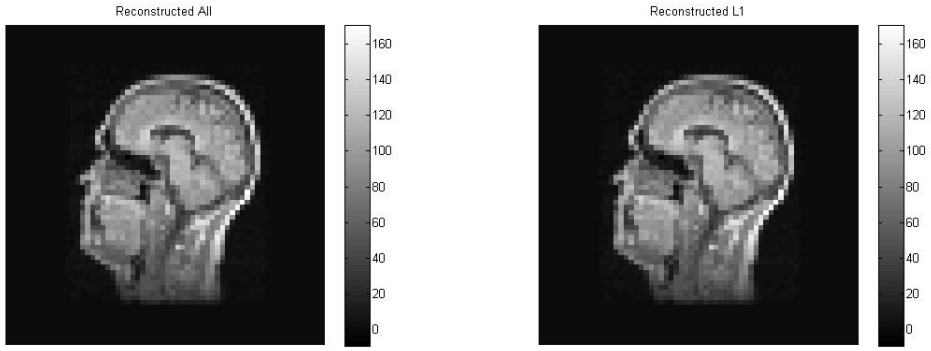

Figure 2.10: Reconstruction using 100 low resolution images (left) and those selected by the A_B design (right).

As expected, the A_B design provides a better reconstruction; it has fewer artifacts in the background. It is also evident that both optimal designs yield images that are very close to the ones obtained using all 100 low resolution images. This illustrates the advantage of our method, it enables us to obtain comparable results with far less data.

To show the difference between all designs, we list the relative 2-norm reconstruction errors in Tables 2.3 and 2.4.

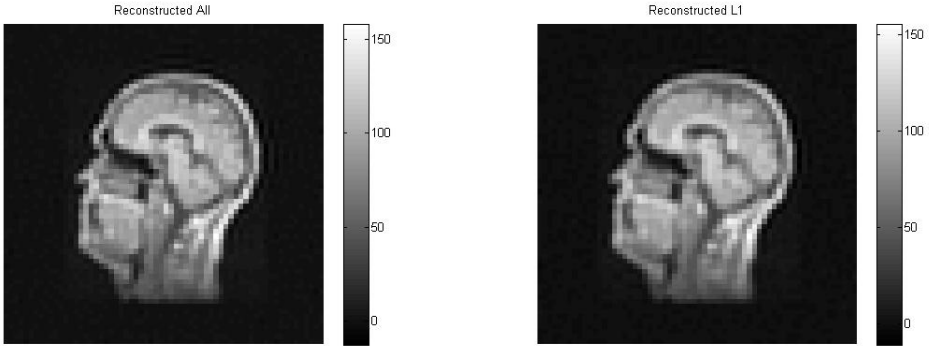

Figure 2.11: Reconstruction using 100 low resolution images (left) and those selected by the A_π design (right).

Table 2.3: Relative errors of the reconstructed images: A_B design

design	relative error
A_B	4.94%
full	4.95%

Table 2.4: Relative errors of the reconstructed images: A_π design

design	relative error
A_π	12.15%
full	9.42%

Chapter 3

Numerical methods for E-optimal design

In this chapter, we explore formulations of the Bayesian E-optimal design [5, 14], in which the the largest eigenvalue of the covariance matrix is minimized. For well-posed problems, this results in solving the optimization problem with sparsity control

$$\min_{w} \quad \phi_E \left[\Sigma_{\widehat{m}}(W) \right] + \beta \|w\|_0 \quad \text{s.t.} \;\; 0 \leq w \leq w_{\max},$$

where

$$\phi_E \left[\Sigma_{\widehat{m}}(W) \right] = \lambda_{\max} \left[\Sigma_{\widehat{m}}(W) \right],$$

given the covariance matrix

$$\Sigma_{\widehat{m}}(W) = \frac{\sigma^2}{N} (A_n^\top W A_n)^{-1}.$$

3.1 The E_B design

For ill-posed problems, following the ideas of the A_B and A_π designs, again we propose two different designs based on different prior information.

If the covariance matrix of the model is available, then the Bayes estimate is chosen as the model estimate,

$$\widehat{m}(w) = (A^\top W A + \Sigma_m^{-1})^{-1}(A^\top W d_w + \Sigma^{-1}\mu), \tag{3.1}$$

where μ and Σ_m are, respectively, the prior mean and covariance matrix. The E_B design minimizes its Bayes risk, which is given by

$$\phi_{E_B}(W) = \lambda_{\max}\left[\,(\,A^\top W A + \Sigma_m^{-1}\,)^{-1}\,\right]. \tag{3.2}$$

The optimal sparsity is decided based on the Pareto curve in the same way as is described in Chapter 2.

3.2 The E_{Tik} design

When the covariance matrix is not available or its inverse is difficult to obtain, we use the Tikhonov solution as the estimated model

$$\widehat{m}(w) = (A^\top W A + \alpha L^\top L)^{-1} A^\top W d_w, \tag{3.3}$$

where α is a regularization parameter and L is a chosen matrix (e.g., a discrete derivative operator).

In this case, the E_{Tik} design minimizes

$$\phi_{E_{\text{Tik}}}(W) = \lambda_{\max}\left[\,(\,A^\top W A + \alpha L^\top L\,)^{-1}\,\right]. \tag{3.4}$$

3.3 Numerical optimization of the E_B and E_{Tik} designs

The above approaches involve eigenvalue optimization techniques [52], which becomes very difficult when the problem size is large. In general there are two ways to handle this situation: Optimize evaluate or Evaluate optimize.

In the Optimize evaluate approach, we must somehow compute the largest eigenvalue of the covariance matrix. After that, the derivatives that are used to solve the optimization need to be found analytically. Both steps of this approach may be difficult.

Thus, we prefer the second option, the Evaluate optimize, in which we consider first approximating the largest eigenvalue of the covariance matrix and then evaluating the derivatives by differentiation.

3.3.1 Eigenvalue approximation

The first step in solving the proposed designs is to approximate the largest eigenvalue of the covariance matrix. By some simple linear algebra, it is obvious that minimizing the largest eigenvalue of the covariance matrix is equivalent to maximizing the smallest eigenvalue of the information matrix, which is the inverse

covariance matrix. Hence, the objective functions to maximize are:

$$\phi_{E_B}(w) = \lambda_{\min}(A^\top W A + \Sigma_m^{-1}) \tag{3.5}$$

and

$$\phi_{E_{\text{Tik}}}(w) = \lambda_{\min}(A^\top W A + \alpha L^\top L). \tag{3.6}$$

There are many eigenvalue approximation techniques in the literature, for example, methods based on power iteration [74] and SVD decomposition [56], etc. In this thesis we apply the inverse iteration to approximate the smallest eigenvalue of the information matrix [33].

Algorithm 4 Inverse iteration

(1) Choose u_0 such that $\| u_0 \| = 1$ and an integer k.
(2) For $j = 1, \cdots, k$, solve $Hu_j = \frac{1}{\sqrt{u_{j-1}^\top u_{j-1}}} u_{j-1}$.
(3) Set $\lambda_{\min} \approx \frac{1}{\sqrt{u_k^\top u_k}}$.

We start with a normalized vector u_0 and choose an appropriate integer k to be the number of iterations we would like to perform in each optimization loop. In each inverse iteration, we solve a linear system, where the matrix H is $A^\top W A + \Sigma_m^{-1}$ for the E_B design and $A^\top W A + \alpha L^\top L$ for the E_{Tik} design. Finally, we set the smallest eigenvalue $\lambda_{\min}$ that we are looking for to be the inverse of the norm of the converged eigenvector.

Below is the matrix form of the inverse iteration process. The original difficult eigenvalue optimization problem has been recast as a constrained optimization problem, which can be easily solved by standard optimization methods such as

Steepest Descent.

$$\min_{u,w} \frac{1}{2} u_k^\top u_k$$
$$\text{s.t.} \quad I(u)Bu - Eu - g = 0$$

where

$$B = \text{diag}(H, \cdots, H)$$
$$u = [u_1^\top, \cdots, u_k^\top]^\top, \quad g = [u_0^\top, 0, \cdots, 0]^\top$$
$$E = \begin{pmatrix} 0 & & & \\ I & 0 & & \\ & \ddots & \ddots & \\ & & I & 0 \end{pmatrix}, \quad I(u) = \begin{pmatrix} I & & & \\ & \sqrt{u_1^\top u_1} I & & \\ & & \ddots & \\ & & & \sqrt{u_{k-1}^\top u_{k-1}} I \end{pmatrix}.$$

One important thing to notice here is the choice of the integer k, in other words, how many inverse iterations we want to perform in each optimization loop. According to our observation, usually setting k to be between 5 and 10 should be enough. This will be demonstrated later in the experiments.

3.3.2 Evaluating the derivatives

We use the E_{Tik} design as an example to show the derivation of the derivatives of the objective functions. We rewrite the constraint in the optimization problem as

$$\begin{bmatrix} H & & & \\ & H & & \\ & & \ddots & \\ & & & H \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_k \end{bmatrix} = \begin{bmatrix} 0 & & & \\ \frac{1}{\sqrt{u_1^\top u_1}} & 0 & & \\ & \ddots & \ddots & \\ & & \frac{1}{\sqrt{u_{k-1}^\top u_{k-1}}} & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_k \end{bmatrix} + \begin{bmatrix} u_0 \\ 0 \\ \vdots \\ 0 \end{bmatrix},$$

which gives us the linear system

$$\left\{\begin{array}{rcl} A^\top \mathrm{diag}\,(Au_1) + H\frac{\partial u_1}{\partial w} &=& 0 \\ A^\top \mathrm{diag}\,(Au_2) + H\frac{\partial u_2}{\partial w} &=& [(u_1^\top u_1)^{-1/2} - (u_1^\top u_1)^{-3/2} u_1 u_1^\top]\frac{\partial u_1}{\partial w} \\ \vdots & & \\ A^\top \mathrm{diag}\,(Au_k) + H\frac{\partial u_k}{\partial w} &=& [(u_{k-1}^\top u_{k-1})^{-1/2} - (u_{k-1}^\top u_{k-1})^{-3/2} u_{k-1} u_{k-1}^\top]\frac{\partial u_1}{\partial w} \end{array}\right. .$$

Defining

$$B_i = (u_i^\top u_i)^{-1/2} - (u_i^\top u_i)^{-3/2} u_i u_i^\top,$$

we obtain

$$\begin{bmatrix} H & & & \\ -B_1 & H & & \\ & \ddots & \ddots & \\ & & -B_{k-1} & H \end{bmatrix} \begin{bmatrix} \frac{\partial u_1}{\partial w} \\ \frac{\partial u_2}{\partial w} \\ \vdots \\ \frac{\partial u_k}{\partial w} \end{bmatrix} = - \begin{bmatrix} A^\top \mathrm{diag}\,(Au_1) \\ A^\top \mathrm{diag}\,(Au_2) \\ \vdots \\ A^\top \mathrm{diag}\,(Au_k) \end{bmatrix}.$$

Thus, the derivative of the objective function is

$$\begin{aligned} &\frac{\partial}{\partial w}\left(\frac{1}{2} u_k^\top u_k\right) \\ &= - \begin{bmatrix} A^\top \mathrm{diag}\,(Au_1) \\ A^\top \mathrm{diag}\,(Au_2) \\ \vdots \\ A^\top \mathrm{diag}\,(Au_k) \end{bmatrix}^\top \begin{bmatrix} H & -B_1^\top & & & \\ & H & -B_2^\top & & \\ & & \ddots & \ddots & \\ & & & H & -B_{k-1} \\ & & & & H \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ u_k \end{bmatrix}. \end{aligned}$$

3.4 Numerical experiments

In this section, we will show the performance of the two E-optimal designs through first, the 1D magnetotelluric example that we have used in the A-optimal designs and then a borehole ray tomography example.

3.4.1 An ill-posed 1D magnetotelluric example

We apply both E-optimal designs to the same 1D magnetotelluric example that we have used for the A-optimal designs. For the purpose of comparison, all experiment and parameter setting as exactly the same as is in Chapter 2.

Quality of eigenvalue approximation

Before we start the experiments, we first test how well the eigenvalue approximation works for both designs. Tables 3.1 and 3.2 show the difference between the true eigenvalues and the estimated ones for different values of k with their corresponding CPU times that were needed to compute the eigenvalue approximation.

Table 3.1: Eigenvalue approximation for the E_B design

k	$\lambda_{\text{app}} - \lambda_{\text{true}}$	CPU time
3	$4.0500e-2$	$3.120020e-2$
5	$1.3190e-4$	$6.240040e-2$
8	0	$6.240040e-2$

Table 3.2: Eigenvalue approximation for the E_{Tik} design

k	$\lambda_{\text{app}} - \lambda_{\text{true}}$	CPU time
3	$1.0800e-2$	0
6	$1.0000e-5$	$3.120020e-2$
9	0	$6.240040e-2$

Based on the tables, we choose $k = 8$ for the E_B design and $k = 9$ for the E_{Tik} design. The approximation of eigenvalue is very accurate while the time cost is still reasonable.

The E_B design

First we show results from the E_B design. Figure 3.1 shows the L-curve, in which the corner is located at $\|w\|_0 = 20$.

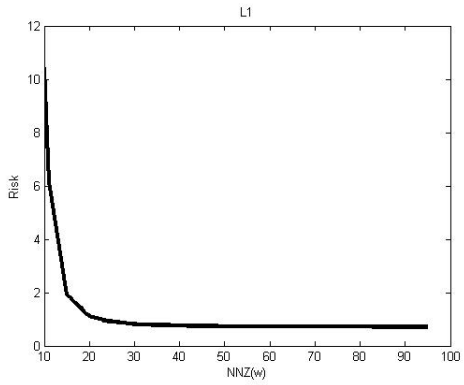

Figure 3.1: The panel shows the risk as a function of sparsity $\text{nnz}(w) = \| w \|_0$ of the optimal w obtained with the E_B design.

Figure 3.2 shows examples of model estimates using the optimal w_i determined by the corner in the L-curve. The figure shows the true model as well as three reconstructions from three different noise realizations. We see good reconstruction results using only the 20 selected experiments.

For comparison, we show the model estimates obtained using all 100 experiments and also the estimates based on 20 equally-spaced naively chosen experiments in Figure 3.3. It is obvious that the E_B design provides the best results.

Table 3.3 shows the relative errors ($\|m_{\text{true}} - \widehat{m}\| / \|m_{\text{true}}\|$) of the reconstructions. Again the estimates obtained using all 100 experiments are worse than those using

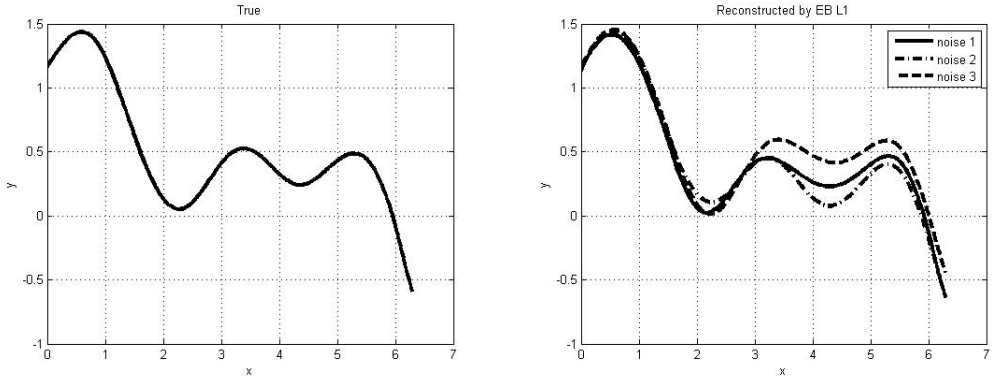

Figure 3.2: The left panel shows the true test model. The right panel shows examples of model estimates obtained with the E_B design using 20 optimal experiments for three different noise realizations.

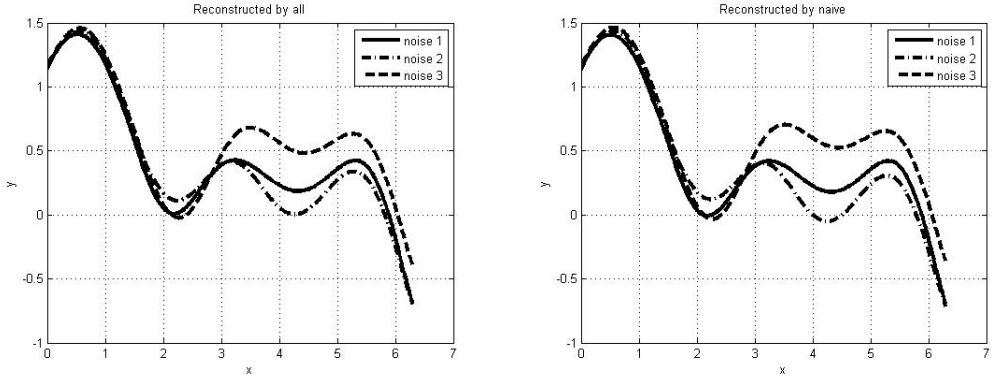

Figure 3.3: Model estimates obtained using all 100 experiments (left) and the equally-spaced naive design (right).

only the 20 optimal experiments.

Table 3.3: Relative errors of the model reconstructions obtained with the E_B design

design	error (noise 1)	error (noise 2)	error (noise 3)
E_B	7.32%	13.35%	12.44%
all	10.29%	20.12%	19.26%
naive	11.38%	24.40%	22.38%

The E_{Tik} design

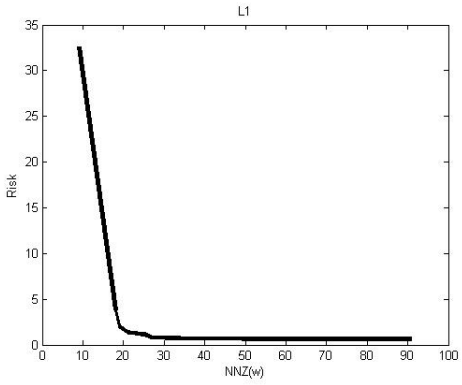

Figure 3.4: The panel shows the risk as a function of sparsity $\text{nnz}(w) = \| w \|_0$ of the optimal w obtained with the E_{Tik} design.

In Figure 3.4, we choose the corner at $\|w\|_0 = 21$. This defines the optimal solution used in the example.

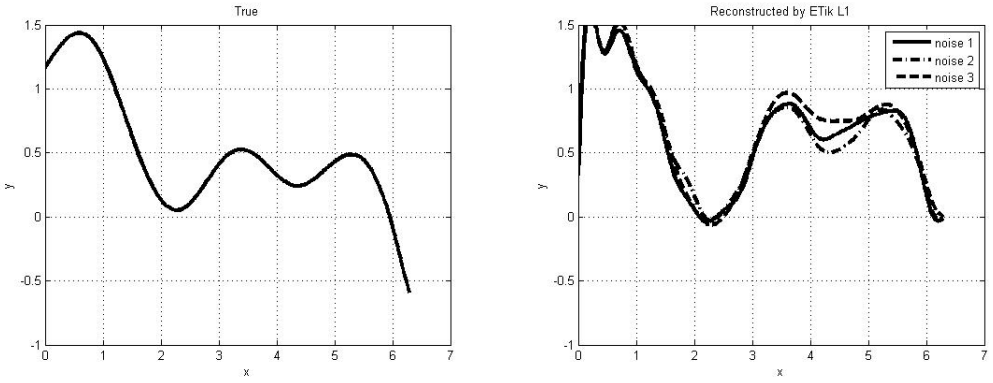

Figure 3.5: The true test model (left) and the estimated models obtained with ℓ^1-E_{Tik} design using the 21 optimal experiments with three different noise realizations (right).

Figure 3.5 shows the model estimates for three different noise realizations obtained using the ℓ^1 approach. Figure 3.6 shows the estimates obtained using all 100 experiments and using 21 equally-spaced naively chosen experiments. The

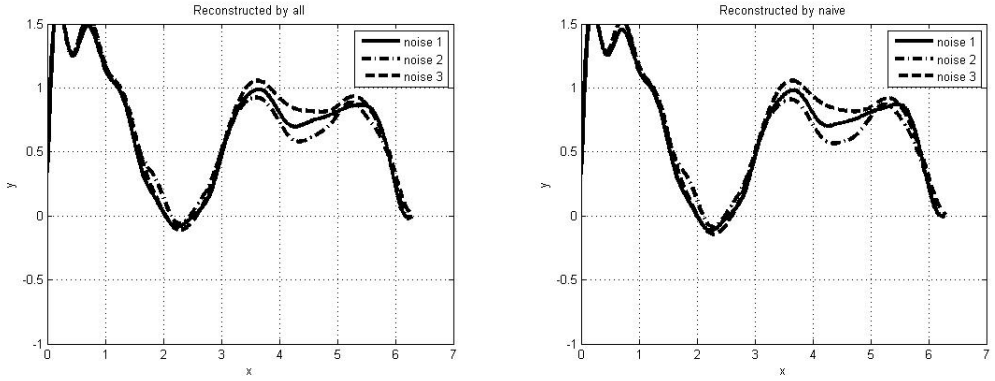

Figure 3.6: Model estimates obtained with the E_{Tik} design using all 100 experiments (left) and the equally-spaced naive design (right).

relative errors of the estimates are shown in Table 3.4. The optimal design still leads to smaller errors than the full and naive designs but the results are not as good as those obtained with the E_B design. We omit the rest of the results for the ℓ^0 approach as they are very similar to those obtained with the ℓ^1 penalty.

Table 3.4: Relative errors of the model estimates obtained with the E_{Tik} (ℓ^1) design

design	error (noise 1)	error (noise 2)	error (noise 3)
E_{Tik} ℓ^1	40.05%	35.41%	46.70%
all	47.00%	41.07%	53.60%
naive	46.81%	39.47%	53.71%

So far, we have worked on both the A-optimal designs and E-optimal designs. Although their results are comparable based on our results, in general the choice is application dependent. We know that the A-optimal design minimizes the variance in an average sense while the E-optimal design minimizes the situation that has the largest variance, in other words, it minimizes the worst case. The performance depends on whether your example is closer to the average case or to the worse case. For example, if a doctor wants to minimize the largest risk that a tumor

could cause, then E-optimal design would be a good choice. On the other hand, if a car company wants to decide how powerful the air bag should be. Then, in order to fit people's needs on average, A-optimal design is needed.

3.4.2 A borehole ray tomography example

Next we apply our approaches to a borehole ray tomography example [35] that is often used to illustrate purposes in geophysical inverse problems. The purpose of borehole ray tomography is to determine the slowness, which is the inverse of the velocity of a medium. Sources and receivers are placed along the boreholes and travel times from sources to receivers are recorded.

In our experiment, the medium is discretized by the square region $[0, 1] \times [0, 1]$ and boreholes are covering both sides of the region. Totally we have 1600 rays and our goal of an experimental design is to choose the optimal placement of sources and receivers.

Since this is a real problem, we do not have the true covariance matrix, we will apply the example only to the E_{Tik} design, for which the discretization of the gradient operator is used as a smoothing matrix.

Quality of eigenvalue approximation

Table 3.5 show the difference between the true eigenvalues and the estimated ones for different values of k with their corresponding CPU times that were needed to compute the eigenvalue approximation.

For solving the problem accurately, we choose $k = 10$.

Table 3.5: Eigenvalue approximation for the E_{Tik} design

k	$\lambda_{\text{app}} - \lambda_{\text{true}}$	CPU time
4	$1.5406e-5$	$9.874863e$
7	$2.5856e-6$	$1.647371e+1$
10	$8.3150e-7$	$2.266695e+1$

The E_{Tik} design

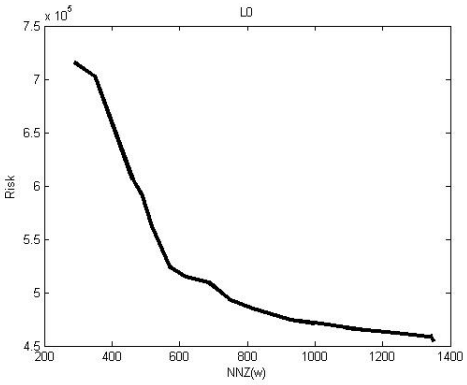

Figure 3.7: The panel shows the risk as a function of sparsity $\text{nnz}(w) = \| w \|_0$ of the optimal w obtained with the E_{Tik} design.

Figure 3.7 shows the Pareto curves for the E_{Tik} optimal design. The corner happens where $\|w\|_0 = 746$. Figures 3.8 show the optimal raypaths reconstructions for the E_{Tik} design as well as the one obtained using all 1600 raypaths.

It is evident that our optimal design yields images that are very close in quality to the ones obtained using all 1600 raypaths.

To show the difference between all designs, we list the relative 2-norm reconstruction errors in Tables 3.6.

Based on 3.8 and 3.6, we observe that the reconstruction from our optimal design provides results almost as good as using all 1600 raypaths. However, we have cut out more than half of the cost.

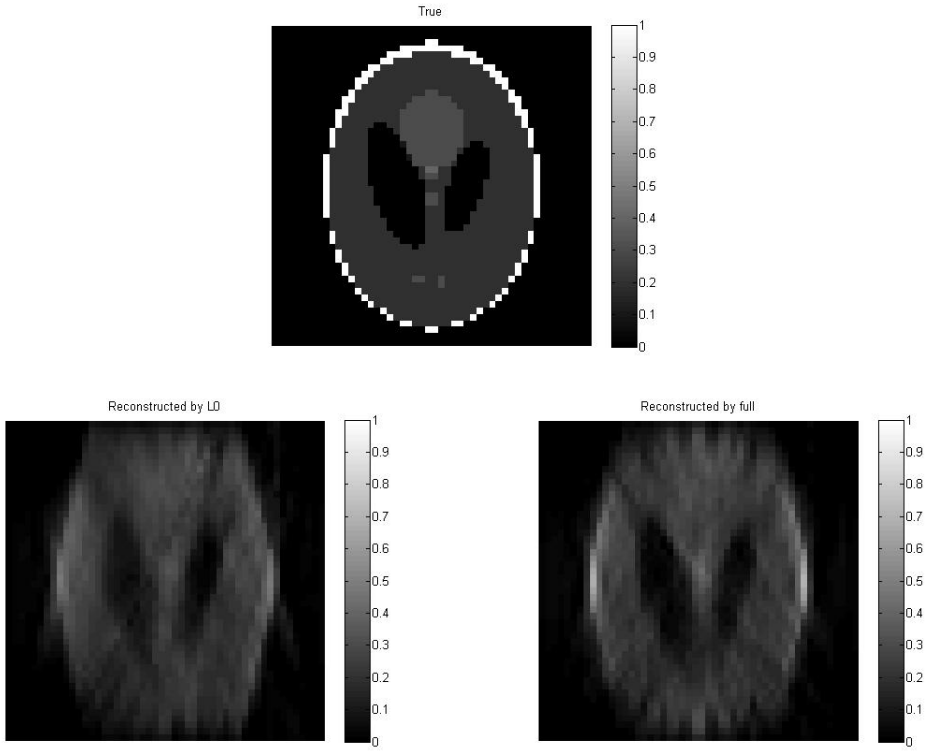

Figure 3.8: The true image (top), reconstructions using 1600 raypaths (bottom left) and those selected by the E_{Tik} design (bottom right).

Table 3.6: Relative errors of the reconstructed images: E_{Tik} design

design	relative error
E_{Tik}	69.63%
full	65.06%

Chapter 4

Optimal design for regularization

In this chapter, we talk about finding an optimal ℓ^2 regularization that plays important roles in imaging science. We consider a discrete linear ill-posed problem of the form

$$Ax + \epsilon = b, \tag{4.1}$$

where $A : R^m \to R^n$ is a discretization of some linear (typically integral) operator and b is the observed data. The random vector ϵ is the noise contained in the data, which is assumed to be iid Gaussian with standard deviation σ and 0 mean.

The general goal is to recover the model x from the observed noisy data b. However, since the problem is ill-posed it is not likely to recover x just from the data itself directly. Thus, we recover x by using a Tikhonov-like regularization and solve the optimization problem

$$\min \quad \frac{1}{2}\|Ax - b\|^2 + \alpha R(x), \tag{4.2}$$

where $R(x)$ is a regularization functional that penalizes unwanted solutions and α is a regularization parameter that is chosen to control the balance between the mis-fitting term and the regularization term so that the solution does not over-fit the data.

For linear ill-posed problems, a regularization $R(x) = \frac{1}{2}\|Lx\|^2$ is commonly used [63, 73], where L is a discretization of a differential operator based on smoothness of the solution. Obviously, the Tikhonov regularization functional is quadratic and therefore, the solution to the problem can be easily obtained by solving the linear system

$$(A^\top A + \alpha L^\top L)x = A^\top b, \tag{4.3}$$

using either direct, but more typically, iterative methods such as the Preconditioned Conjugate Gradient method and Lanczos tridiagonalization method [30, 43].

4.1 An optimal regularization operator

In order to develop an optimal regularization operator we need to define an optimality criteria. In the following we develop such criteria. As we see next, the criteria heavily depends on the assumptions or a-priori information known. Different assumptions lead to a different regularization operators.

Consider a solution to the quadratic regularization problem

$$\hat{x} = (A^\top A + \alpha L^\top L)^{-1} A^\top b.$$

where b is given by equation (4.1). If we are given the "true" solution x^t, we may ask how well $\hat{x}$ reconstructs x^t. The answer to this question is the well known Mean Square Error:

$$\begin{aligned} \text{MSE} = \mathbf{E} \parallel \hat{x} - x^t \parallel^2 &= \alpha^2 \parallel (A^\top A + \alpha L^\top L)^{-1} L^\top L x^t \parallel^2 \\ &\quad + \sigma^2 \text{trace}\left[A (A^\top A + \alpha L^\top L)^{-2} A^\top \right]. \end{aligned}$$

where the expected value is on the noise. The first term which depends on x^t is referred to as the square of the bias and the second term is referred to as the variance. The decomposition of the MSE into the bias and variance is a major point in our discussion. The bias is required if stable solutions to the problem are desired. This implies that even for problems with no noise, the recovered solution $\widehat{x}$ will differ from the true solution x^t. An important question is therefore, how should we decrease the error in recovery?

4.1.1 The first complication: MSE is dependent on the true solution

One could reduce the bias by decreasing α but, as is well documented, this will increase the variance. Thus, the goal is to introduce the "right" L that leads to the "right" bias, that is, a bias that will lead us closer to the true solution x^t and decrease the overall MSE.

This implies that we need some information on x^t, which leads to the first complication in finding the optimal L. Assume that one chose an L that minimizes the MSE. Since the MSE depends on the true solution x^t, minimizing the MSE that

depends on x^t may lead to a large MSE for different "true" models. To overcome this disadvantage, a number of approaches can be used to eliminate the dependency of the MSE with respect to x^t. In the following we explore two different scenarios where different information on x^t is given.

A1: If the covariance of the true x^t exists and is known, then it is possible to use the covariance to define an average MSE. The average MSE in this case, is identical to the so called Bayesian risk.

A2: If some training models $X = [x_1^t, \ldots, x_s^t]$ are given, then it is possible to obtain an empirical estimation of the average MSE. This case is identical to the empirical Bayesian risk.

We now treat each of these cases and discuss the optimization problems that stem from each assumption.

The covariance design

If the mean μ and the covariance matrix Σ of the true model are known, then we can replace the norm of the bias by its average. An optimal solution, in this case, is the one that minimizes the average MSE. This leads to the following optimization problem

$$\begin{aligned} &\min_{L} \ \alpha^2 \parallel (A^\top A + \alpha L^\top L)^{-1} L^\top L \mu \parallel^2 + \\ &\text{trace}\left[(A^\top A + \alpha L^\top L)^{-1}(\alpha^2 L^\top L \Sigma L^\top L + \sigma^2 A^\top A)(A^\top A + \alpha L^\top L)^{-1}\right]. \end{aligned} \tag{4.4}$$

At this point, it may seem surprising but L can be found analytically. For simplicity, we set the mean μ to be zero. Setting $B = L^\top L$, $C = A^\top A + \alpha L^\top L$ and

combining the traces we have

$$\text{AMSE} = \text{trace}\left(C^{-1}(\alpha^2 B\Sigma B + \sigma^2 A^\top A)C^{-1}\right) = \sum_{i=1}^{m} {e_i}^\top \left[C^{-1}(\alpha^2 B\Sigma B + \sigma^2 A^\top A)C^{-1}\right] e_i,$$

where $e_i, i = 1, \cdots, m$ are the standard unit basis vectors. In order to minimize the AMSE, we set its derivative with respect to B to be 0 as follows:

$$\frac{\partial \text{AMSE}}{\partial B} = \sum_{i=1}^{m} \left[-\alpha C^{-1} e_i {e_i}^\top C^{-1}(\alpha^2 B\Sigma B + \sigma^2 A^\top A)C^{-1} + \alpha^2 C^{-1} e_i {e_i}^\top C^{-1} B\Sigma\right] = 0.$$

It is straight forward to verify that, if we chose B to be

$$B = \frac{\sigma^2}{\alpha}\Sigma^{-1} = L^\top L,$$

then the gradient of the AMSE vanishes. That is, the "optimal" $L^\top L$ is a proportion of the inverse covariance of the models. This should not come as a surprise. The average MSE can be interpreted as the Bayesian risk and thus, if the model is Gaussian with mean zero and covariance matrix Σ, then the Maximum A Posterior Estimate (MAP) also yields the minimal Bayes risk. The surprising result (at least for us) is the fact that, for *any* distribution of models with zero mean and covariance matrix Σ, even for distributions that are far from Gaussian, the inverse covariance is the best regularization matrix.

Assume that we are given the covariance matrix and we would like to use its inverse for the solution of the problem. For small scale problems, this is straightforward. However, for large scale problems, working with a general covariance

matrix and its inverse is highly nontrivial. For example, in medical and geophysical imaging, x^t can be a vector with millions of entries. For these applications, working with a dense covariance matrix with 10^{12} entries makes computations infeasible unless some special structure exists. For example, one can assume that the covariance matrix is space invariant which leads to a simple Σ. However, this assumption is rather unrealistic and even if it is true, estimating Σ^{-1} can be highly nontrivial.

The training design

When we do not have the covariance matrix, we use a set of related models which are used as training references. These models are chosen for applications in different situations. For example, if the goal is to deblur some MRI images then X can be chosen as a set of clean MRI examples. In this case, the MSE to minimize is

$$\text{AMSE} = \text{trace}\left[(A^\top A + \alpha L^\top L)^{-1}(\alpha^2 L^\top L M L^\top L + \sigma^2 A^\top A)(A^\top A + \alpha L^\top L)^{-1}\right], \tag{4.5}$$

where $M = \frac{1}{s}\sum_{j=1}^{s} x_j^t {x_j^t}^\top$ and s is the number of training models that we use.

4.1.2 The second complication: The computational complexity

As we have stated above, using a dense regularization matrix for large scale problems will be very expensive in computation. The weakness of the analytic optimal L strengthens the motivation for our work.

To overcome the second disadvantage, the problem of the computational com-

plexity, we can choose L to have some structure that is easy to compute with. We return to the average MSE and minimize it over all $B = L^\top L$ assuming that L has some given sparsity structure. Let $\mathcal{S}$ be the set of matrices with a specific sparsity structure and solve the following *constrained* optimization problem

$$\begin{aligned} \min_{L} \quad & \text{AMSE} \\ \text{s.t.} \quad & L \in \mathcal{S}. \end{aligned} \tag{4.6}$$

Here are a few comments:

- It is rather clear that, in general, the solution in this case is not that $\frac{\sigma^2}{\alpha}\Sigma^{-1} = L^\top L$ since Σ^{-1} may not posses the appropriate sparsity pattern. Also, simple "sparsification" of Σ^{-1} (that is, projecting Σ^{-1} into the constraint set) is in general, not the optimal solution.

- As before, one could set $B = L^\top L$ and solve for B directly, under the constraint that B is symmetric positive and semidefinite (PSD). However, note that, for this case, the AMSE function is non-convex. Furthermore, even if the function was convex, working with the constraint that B is PSD is difficult for large scale problems. In fact, while there is a body of work that deals with estimating the inverse of the covariance matrix, we are not aware of any papers that work well in estimating the inverse covariance matrix when the size of the problem is very large.

- While being optimal is a novel goal, for most practical applications a significant improvement over existing regularization methods will suffice. Thus, although we do not have a convex problem, if it is possible to obtain useful

solutions, this will be welcome for many of the applications we aim for.

In the rest of this chapter, we focus on solving the first complication, while the complication of the computational complexity will be discussed in details in the next chapter.

4.2 Numerical optimization of the optimal regularization

In this section, we discuss numerical optimization methods of the optimal regularization. Different approaches are explored in order to solve individual problems that arise from different designs.

4.2.1 Matrix-based derivative techniques

We start with the derivatives for the proposed designs. Most literature approaches optimization problems by minimizing vectors, while it is not common to minimize with respect to matrices. Hence, we will take some effort to review some matrix-based derivative techniques [62] first.

Here we list a couple basic derivative rules that we have used in deriving our derivatives. Let N and P be matrix variables and c be a constant. i, j, k, l refer to

different matrix entry index.

$$\begin{aligned}
\partial(cN) &= c\partial(N) \\
\partial(N+P) &= \partial(N)+\partial(P) \\
\partial(NP) &= N\partial(P)+P\partial(N) \\
\partial(N^{-1}) &= -N^{-1}\partial(N)N^{-1} \\
\partial(N^{\top}) &= (\partial N)^{\top} \\
\frac{\partial N_{k,l}}{\partial N_{i,j}} &= \delta_{i,k}\delta_{l,j}
\end{aligned} \tag{4.7}$$

In particular, the following matrix differentiation rules have been very handy. Let q be a vector variable and f, g be two constant vectors. J^{ij} is a matrix with 1 on the ij^{th} entry and 0 elsewhere.

$$\begin{aligned}
\frac{\partial(N^{-1})}{\partial q} &= -N^{-1}\frac{\partial N}{\partial q}N^{-1} \\
\frac{\partial(N^{-1})_{k,l}}{\partial N_{i,j}} &= -(N^{-1})_{k,i}(N^{-1})_{j,l} \\
\frac{\partial f^{\top}N^{-1}g}{\partial N} &= -N^{-\top}fg^{\top}N^{-\top} \\
\frac{\partial f^{\top}Ng}{\partial N} &= fg^{\top} \\
\frac{\partial f^{\top}N^{\top}g}{\partial N} &= gf^{\top} \\
\frac{\partial f^{\top}Nf}{\partial N} &= ff^{\top} \\
\frac{\partial N}{\partial N_{i,j}} &= J^{ij} \\
\frac{\partial f^{\top}N^{\top}Ng}{\partial N} &= N(fg^{\top}+gf^{\top})
\end{aligned} \tag{4.8}$$

4.2.2 The covariance design approach

By applying the Stochastic trace approximation, the original covariance design can be rewritten as

$$\begin{aligned} \mathrm{AMSE}_{\mathrm{Cov}} &= \alpha^2 \parallel (A^\top A + \alpha L^\top L)^{-1} L^\top L\mu \parallel^2 \\ &+ v^\top (A^\top A + \alpha L^\top L)^{-1} (\alpha^2 L^\top L \Sigma L^\top L + \sigma^2 A^\top A)(A^\top A + \alpha L^\top L)^{-1} v \end{aligned} \tag{4.9}$$

where v is a random vector of 1 and -1 with equal distribution.

We discuss the derivatives of the two terms in the AMSE with respect to the regularization matrix L separately. We denote the first quadratic term to be U and the second term V. Hence, the objective functional can be expressed as

$$\mathrm{AMSE} = \alpha^2 U + V.$$

We rewrite the U as

$$U = \mu^\top L^\top L (A^\top A + \alpha L^\top L)^{-2} L^\top L \mu.$$

The derivation of its derivative is as follows:

$$\begin{aligned}
\frac{\partial U}{\partial L} &= \frac{\partial[\mu^\top L_{nf}^\top L(A^\top A+\alpha L^\top L)^{-1}(A^\top A+\alpha L^\top L)^{-1}L^\top L\mu]}{\partial L} && (4.10)\\
&+ \frac{\partial[\mu^\top L^\top L_{nf}(A^\top A+\alpha L^\top L)^{-1}(A^\top A+\alpha L^\top L)^{-1}L^\top L\mu]}{\partial L} && (4.11)\\
&+ \frac{\partial[\mu^\top L^\top L(A^\top A+\alpha L_{nf}^\top L)^{-1}(A^\top A+\alpha L^\top L)^{-1}L^\top L\mu]}{\partial L} && (4.12)\\
&+ \frac{\partial[\mu^\top L^\top L(A^\top A+\alpha L^\top L_{nf})^{-1}(A^\top A+\alpha L^\top L)^{-1}L^\top L\mu]}{\partial L} && (4.13)\\
&+ \frac{\partial[\mu^\top L^\top L(A^\top A+\alpha L^\top L)^{-1}(A^\top A+\alpha L_{nf}^\top L)^{-1}L^\top L\mu]}{\partial L} && (4.14)\\
&+ \frac{\partial[\mu^\top L^\top L(A^\top A+\alpha L^\top L)^{-1}(A^\top A+\alpha L^\top L_{nf})^{-1}L^\top L\mu]}{\partial L} && (4.15)\\
&+ \frac{\partial[\mu^\top L^\top L(A^\top A+\alpha L^\top L)^{-1}(A^\top A+\alpha L^\top L)^{-1}L_{nf}^\top L\mu]}{\partial L} && (4.16)\\
&+ \frac{\partial[\mu^\top L^\top L(A^\top A+\alpha L^\top L)^{-1}(A^\top A+\alpha L^\top L)^{-1}L^\top L_{nf}\mu]}{\partial L}. && (4.17)
\end{aligned}$$

where we treat L as fixed constant and L_{nf} as non-fixed variable. Let's discuss these equations one by one.

For (4.10) and (4.17), we have

$$(4.10) = (4.17) = L(A^\top A+\alpha L^\top L)^{-2}L^\top L\mu\mu^\top.$$

For (4.11) and (4.16),

$$(4.11) = (4.16) = L\mu(A^\top A+\alpha L^\top L)^{-2}L^\top L\mu^\top.$$

For (4.12) and (4.15), we have

$$\begin{aligned}
(4.12) &= \mu^\top L^\top L \frac{\partial (A^\top A + \alpha L_{nf}^\top L)^{-1}}{\partial L} (A^\top A + \alpha L^\top L)^{-1} L^\top L\mu \\
&= -\mu^\top L^\top L (A^\top A + \alpha L^\top L)^{-1} \frac{\partial (A^\top A + \alpha L_{nf}^\top L)}{\partial L} \\
&\quad (A^\top A + \alpha L^\top L)^{-1} (A^\top A + \alpha L^\top L)^{-1} L^\top L\mu \\
&= -\alpha \frac{\partial [\mu^\top L^\top L (A^\top A + \alpha L^\top L)^{-1} L_{nf}^\top L (A^\top A + \alpha L^\top L)^{-2} L^\top L\mu]}{\partial L} \\
&= -\alpha L (A^\top A + \alpha L^\top L)^{-2} L^\top L\mu (A^\top A + \alpha L^\top L)^{-1} L^\top L\mu^\top = (4.15).
\end{aligned}$$

For (4.13) and (4.14), we have

$$\begin{aligned}
(4.13) &= \mu^\top L^\top L \frac{\partial (A^\top A + \alpha L^\top L_{nf})^{-1}}{\partial L} (A^\top A + \alpha L^\top L)^{-1} L^\top L\mu \\
&= -\mu^\top L^\top L (A^\top A + \alpha L^\top L)^{-1} \frac{\partial (A^\top A + \alpha L^\top L_{nf})}{\partial L} \\
&\quad (A^\top A + \alpha L^\top L)^{-1} (A^\top A + \alpha L^\top L)^{-1} L^\top L\mu \\
&= -\alpha \frac{\partial [\mu^\top L^\top L (A^\top A + \alpha L^\top L)^{-1} L^\top L_{nf} (A^\top A + \alpha L^\top L)^{-2} L^\top L\mu]}{\partial L} \\
&= -\alpha L (A^\top A + \alpha L^\top L)^{-1} L^\top L\mu (A^\top A + \alpha L^\top L)^{-2} L^\top L\mu^\top = (4.14).
\end{aligned}$$

Hence, we obtain

$$\begin{aligned}
\frac{\partial U}{\partial L} &= 2L(A^\top A + \alpha L^\top L)^{-2} L^\top L\mu\mu^\top \\
&+ 2L\mu (A^\top A + \alpha L^\top L)^{-2} L^\top L\mu^\top \\
&- 2\alpha L (A^\top A + \alpha L^\top L)^{-2} L^\top L\mu (A^\top A + \alpha L^\top L)^{-1} L^\top L\mu^\top \\
&- 2\alpha L (A^\top A + \alpha L^\top L)^{-1} L^\top L\mu (A^\top A + \alpha L^\top L)^{-2} L^\top L\mu^\top.
\end{aligned}$$

With respect to the second term V, we define $D = \alpha^2 L^\top L \Sigma L^\top L + \sigma^2 A^\top A$ and

yield

$$\nabla_L V = \nabla_L (v^\top C^{-1} D C^{-1} v).$$

Setting

$$z = C^{-1} v \quad \text{and} \quad y = C^{-1} D z$$

yields

$$V = z^\top D z.$$

Following the same derivative techniques as described above, we obtain

$$\nabla_L V = -2\alpha L(yz^\top + zy^\top) + 2\alpha^2 L(\Sigma L^\top L z z^\top + z z^\top L^\top L \Sigma).$$

Finally, the derivative of the objective functional is expressed as

$$\frac{\partial \text{AMSE}_{\text{Cov}}}{\partial L} = \alpha^2 \frac{\partial U}{\partial L} + \frac{\partial V}{\partial L}.$$

4.2.3 The training design approach

The training design can be rewritten as follows again using the Stochastic trace approximation

$$\text{AMSE} = v^\top (A^\top A + \alpha L^\top L)^{-1} (\alpha^2 L^\top L M L^\top L + \sigma^2 A^\top A)(A^\top A + \alpha L^\top L)^{-1} v. \quad (4.18)$$

This formulation looks a lot like the second term V in the covariance design. Therefore, we will give the derivative directly.

$$\nabla_L \text{AMSE} = -2\alpha L(y' z^\top + z y'^\top) + 2\alpha^2 L(M L^\top L z z^\top + z z^\top L^\top L M),$$

where $y' = C^{-1}D'z$ and $D' = \alpha^2 L^\top LML^\top L + \sigma^2 A^\top A$.

4.2.4 Numerical Optimization

The numerical optimization is carried out by the steepest descent method. One important thing to notice here is that, when checking if the derivatives are properly derived, we need to check if the relation

$$\text{AMSE}(L + hL_s) = \text{AMSE}(x) + (\frac{\partial \text{AMSE}}{\partial L}, hL_s) + o(h^2)$$

holds. In this case since both $\frac{\partial \text{AMSE}}{\partial L}$ and hL_s are matrices with the same size as L, the normal inner product will give us another matrix which does not add up to the AMSE as it is a scalar. Thus, we introduce the so called standard inner product for the two matrices [53]

$$(\frac{\partial \text{AMSE}}{\partial L}, hL_s) = \text{trace}[(\frac{\partial \text{AMSE}}{\partial L})^\top (hL_s)].$$

Chapter 5

Optimal sparse regularization

In the previous chapter, we have discussed the formulation and derivative techniques for finding the optimal regularization matrix. As you may see, computing the derivatives is very time consuming since PCG is used everywhere in the computation. Also, considering the structure of the image, in some cases, it is not really necessary to find L with the dense pattern. Thus, we need to add certain sparsity constraints to L in order to save the computational cost.

In this chapter, we propose three different sparsity patterns for L: The local diagonal pattern, the ℓ^1 norm pattern and the Kronecker product pattern.

5.1 The local diagonal pattern

The first one we call the local diagonal pattern, which means we only use information from the k-diagonals that are close to the main diagonal. In 1D problems, this indicates that there are only nonzero entries on the main, sub and super diagonals. In 2D problems, we choose the pattern of the Laplacian matrix as our structure

for L, in other words, the nonzero pattern of L is just 5 diagonals. The reason for doing this will be easy to explain on an image. Images consist of pixels and each pixel is often more related to the pixels that surround it. Thus we choose the 5 diagonals which exactly represent the 4 pixels in the neighborhood, up, down, left and right as is shown as the black grids in 5.1.

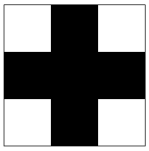

Figure 5.1: The picture shows that each pixel is often more related to the four pixels that surround it.

The formulation of the local diagonal pattern is the same as the dense pattern that we have discussed in Chapter 4, except that this time the pattern of L is restrained to only have nonzero entries in several diagonals. The problem to solve for the local diagonal pattern is

$$\begin{aligned} \min_{L} \quad & \text{AMSE} \\ \text{s.t.} \quad & L \in \text{local diagonal}. \end{aligned}$$

5.2 The ℓ^1 norm pattern

A second approach is to have a sparse L with only few nonzero entries so that the computational cost is much less. We consider adding an ℓ^1 norm to the average MSE to control the sparsity of L. The number of nonzeros in L is reduced by increasing the value of β.

Thus, the optimization problem to solve for the ℓ^1 norm pattern:

$$\min_L \text{AMSE} + \beta \parallel L \parallel_1 .$$

In order to find the best sparsity, we plot the AMSE as a function of the number of nonzeros in L, as is described in the A-optimal design work. The optimal sparsity is decided by the corner of the L-curve, which suggests an optimal compromise between the reconstruction quality and the sparsity of L.

5.3 The Kronecker product pattern

The last sparse pattern we consider is the Kronecker product pattern, for which we define

$$L = L_1 \otimes L_2,$$

where L_1 and L_2 are two smaller matrices. Hence, rather than minimizing with respect to a large dense matrix L we now minimize the objective functional with respect to smaller matrices L_1 and L_2. Since all the derivatives and optimization process are now done on smaller matrices, the computational cost is greatly reduced.

In this case, the original covariance design turns into an optimization problem

with respect to two small matrices L_1 and L_2 as

$$\begin{aligned}
&\min_{L1,L2} \ \alpha^2 \parallel (A^\top A + \alpha(L_1 \otimes L_2)^\top (L_1 \otimes L_2))^{-1} (L_1 \otimes L_2)^\top (L_1 \otimes L_2)\mu \parallel^2 + \\
&\text{trace}[(A^\top A + \alpha(L_1 \otimes L_2)^\top (L_1 \otimes L_2))^{-1} \\
&(\alpha^2 (L_1 \otimes L_2)^\top (L_1 \otimes L_2)\Sigma(L_1 \otimes L_2)^\top (L_1 \otimes L_2) + \sigma^2 A^\top A) \\
&(A^\top A + \alpha(L_1 \otimes L_2)^\top (L_1 \otimes L_2))^{-1}].
\end{aligned} \tag{5.1}$$

Similarly, the training design is

$$\begin{aligned}
&\min_{L1,L2} \ \text{trace}[(A^\top A + \alpha(L_1 \otimes L_2)^\top (L_1 \otimes L_2))^{-1} \\
&(\alpha^2 (L_1 \otimes L_2)^\top (L_1 \otimes L_2) M (L_1 \otimes L_2)^\top (L_1 \otimes L_2) + \sigma^2 A^\top A) \\
&(A^\top A + \alpha(L_1 \otimes L_2)^\top (L_1 \otimes L_2))^{-1}].
\end{aligned} \tag{5.2}$$

5.4 Numerical optimization of different sparse patterns

As we have mentioned, in this section, totally we need to deal with three different patterns for the structure of L and the matrix derivatives techniques we use are different in each pattern. We will discuss them separately in this section. Below is the formulation for the covariance design. The training design is very similar to the second term in the covariance design, therefore we skip the details.

5.4.1 The local diagonal pattern

The computation of derivatives for the local diagonal pattern is very different from what we have seen in the dense pattern in the previous chapter because the derivatives are taken entry-wise. Again we separate the AMSE into two terms U

and V.

We derive the derivative of the first term in details as an example. In order to distinguish the local diagonal pattern from the dense pattern, we denote the regularization matrix as L_l.

$$\begin{aligned}
\frac{\partial U}{\partial L_l(i,j)} &= \frac{\partial[\mu^\top L_{lnf}^\top L_l (A^\top A + \alpha L_l^\top L_l)^{-2} L_l^\top L_l \mu]}{\partial L_l(i,j)} && (5.3)\\
&+ \frac{\partial[\mu^\top L_l^\top L_{lnf} (A^\top A + \alpha L_l^\top L_l)^{-2} L_l^\top L_l \mu]}{\partial L_l(i,j)} && (5.4)\\
&+ \frac{\partial[\mu^\top L_l^\top L_l (A^\top A + \alpha L_{lnf}^\top L_l)^{-1} (A^\top A + \alpha L_l^\top L_l)^{-1} L_l^\top L_l \mu]}{\partial L_l(i,j)} && (5.5)\\
&+ \frac{\partial[\mu^\top L_l^\top L_l (A^\top A + \alpha L_l^\top L_{lnf})^{-1} (A^\top A + \alpha L_l^\top L_l)^{-1} L_l^\top L_l \mu]}{\partial L_l(i,j)} && (5.6)\\
&+ \frac{\partial[\mu^\top L_l^\top L_l (A^\top A + \alpha L_l^\top L_l)^{-1} (A^\top A + \alpha L_{lnf}^\top L_l)^{-1} L_l^\top L_l \mu]}{\partial L_l(i,j)} && (5.7)\\
&+ \frac{\partial[\mu^\top L_l^\top L_l (A^\top A + \alpha L_l^\top L_l)^{-1} (A^\top A + \alpha L_l^\top L_{lnf})^{-1} L_l^\top L_l \mu]}{\partial L_l(i,j)} && (5.8)\\
&+ \frac{\partial[\mu^\top L_l^\top L_l (A^\top A + \alpha L_l^\top L_l)^{-2} L_{lnf}^\top L_l \mu]}{\partial L_l(i,j)} && (5.9)\\
&+ \frac{\partial[\mu^\top L_l^\top L_l (A^\top A + \alpha L_l^\top L_l)^{-2} L_l^\top L_{lnf} \mu]}{\partial L_l(i,j)}, && (5.10)
\end{aligned}$$

where we treat L_l as fixed constant and L_{lnf} as non-fixed variable. Let's discuss these equations one by one. Denote $J_{i,j}$ to be the single-entry matrix with 1 at (i,j) and 0 elsewhere.

For (5.3) and (5.10),

$$\begin{aligned}
(5.3) &= \mu^\top \frac{\partial L_{lnf}^\top}{\partial L_l(i,j)} L_l (A^\top A + \alpha L_l^\top L_l)^{-2} L_l^\top L_l \mu\\
&= \mu^\top (\frac{\partial L_{lnf}}{\partial L_l(i,j)})^\top L_l (A^\top A + \alpha L_l^\top L_l)^{-2} L_l^\top L_l \mu\\
&= \mu^\top J_{i,j}^\top L_l (A^\top A + \alpha L_l^\top L_l)^{-2} L_l^\top L_l \mu = (5.10).
\end{aligned}$$

For (5.4) and (5.9),

$$\begin{aligned}(5.4) &= \mu^\top L_l^\top \frac{\partial L_{lnf}}{\partial L_l(i,j)}(A^\top A + \alpha L_l^\top L_l)^{-2} L_l^\top L_l \mu \\ &= \mu^\top L_l^\top J_{i,j}(A^\top A + \alpha L_l^\top L_l)^{-2} L_l^\top L_l \mu = (5.9).\end{aligned}$$

For (5.5) and (5.8),

$$\begin{aligned}(5.5) &= \mu^\top L_l^\top L_l \frac{\partial (A^\top A + \alpha L_{lnf}^\top L_l)^{-1}}{\partial L_l(i,j)}(A^\top A + \alpha L_l^\top L_l)^{-1} L_l^\top L_l \mu \\ &= -\mu^\top L_l^\top L_l (A^\top A + \alpha L_l^\top L_l)^{-1} \frac{\partial (A^\top A + \alpha L_{lnf}^\top L_l)}{\partial L_l(i,j)}(A^\top A + \alpha L_l^\top L_l)^{-2} L_l^\top L_l \mu \\ &= -\alpha \mu^\top L_l^\top L_l (A^\top A + \alpha L_l^\top L_l)^{-1} J_{i,j}^\top L_l (A^\top A + \alpha L_l^\top L_l)^{-2} L_l^\top L_l \mu = (5.8).\end{aligned}$$

For (5.6) and (5.7),

$$\begin{aligned}(5.6) &= \mu^\top L_l^\top L_l \frac{\partial (A^\top A + \alpha L_l^\top L_{lnf})^{-1}}{\partial L_l(i,j)}(A^\top A + \alpha L_l^\top L_l)^{-1} L_l^\top L_l \mu \\ &= -\mu^\top L_l^\top L_l (A^\top A + \alpha L_l^\top L_l)^{-1} \frac{\partial (A^\top A + \alpha L_l^\top L_{lnf})}{\partial L_l(i,j)}(A^\top A + \alpha L_l^\top L_l)^{-2} L_l^\top L_l \mu \\ &= -\alpha \mu^\top L_l^\top L_l (A^\top A + \alpha L_l^\top L_l)^{-1} L_l^\top J_{i,j} (A^\top A + \alpha L_l^\top L_l)^{-2} L_l^\top L_l \mu = (5.7).\end{aligned}$$

Hence,

$$\begin{aligned}\frac{\partial U}{\partial L_l(i,j)} &= 2\mu^\top J_{i,j}^\top L_l (A^\top A + \alpha L_l^\top L_l)^{-2} L_l^\top L_l \mu \\ &+ 2\mu^\top L_l^\top J_{i,j} (A^\top A + \alpha L_l^\top L_l)^{-2} L_l^\top L_l \mu \\ &- 2\alpha \mu^\top L_l^\top L_l (A^\top A + \alpha L_l^\top L_l)^{-2} L_l^\top J_{i,j} (A^\top A + \alpha L_l^\top L_l)^{-1} L_l^\top L_l \mu \\ &- 2\alpha \mu^\top L_l^\top L_l (A^\top A + \alpha L_l^\top L_l)^{-1} L_l^\top J_{i,j} (A^\top A + \alpha L_l^\top L_l)^{-2} L_l^\top L_l \mu.\end{aligned}$$

For V:

For each nonzero entry, the derivative is given by

$$\nabla_{L_l(i,j)} V = -2\alpha L_l^i (yz^j + zy^j) + 2\alpha^2 L_l^i (\Sigma L_l^\top L_l z z^j + z z^j L_l^\top L_l \Sigma).$$

where L_l^i denotes the i^{th} low of L and z^j and y^j denote the j^{th} entry of the vectors z and y.

Finally, we obtain

$$\frac{\partial \text{AMSE}}{\partial L_l(i,j)} = \alpha^2 \frac{\partial U}{\partial L_l(i,j)} + \frac{\partial V}{\partial L_l(i,j)}.$$

Notice, in the derivative of AMSE with respect to L_l, we need to compute $L_l^\top J_{i,j}$, which is basically just a single-column matrix generated by putting the transpose of the i^{th} row of the matrix L_l on the j^{th} column. Also, we only need to do the above derivatives on non-zero elements of L_l. Hence, the computational cost is greatly reduced.

In particular, for 1D problems, we only have nonzero entries on the tri-diagonals of the regularization matrix. Thus, the above computation is only done of those three diagonals. Since the cost of this derivative is less than $3n$ while it is n^2 in the case of the dense L, the local diagonal is considered to be one of the most efficient patterns.

5.4.2 The ℓ^1 norm pattern

There are two parts in the ℓ^1 pattern: the AMSE and the ℓ^1 norm. The only part that is new to us is the ℓ^1 norm of a matrix, which is simply the maximum absolute

column sum of the matrix based on the theory of the induced matrix norm:

$$\| L \|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^{m} |L_{i,j}|.$$

However, it is very difficult to define a proper derivative for the above norm. Therefore, we treat the ℓ^1 norm as the sum of the absolute value of all entries in L.

$$\| L \|_1 = \sum_{i,j=1}^{m,n} |L_{i,j}|.$$

The derivative of this norm is simply a matrix of the signs of the entries of L. In other words,

$$\frac{\partial \| L \|_1}{\partial L_{i,j}} = \begin{cases} 1 & \text{if} \quad L_{i,j} > 0 \\ -1 & \text{if} \quad L_{i,j} < 0 \\ 0 & \text{if} \quad L_{i,j} = 0. \end{cases}$$

5.4.3 The Kronecker product pattern

The difficulty in deriving derivatives for this pattern is that the regularization matrix is a function of two other matrices. Hence, the Chain rule for matrix needs to be applied [62].

The Chain rule

Let $S = f(N)$, that is, the matrix S is a function of another matrix N. The goal is to find the derivative of the function $g(S)$ with respect to N:

$$\frac{\partial g(S)}{\partial N} = \frac{\partial g(f(N))}{\partial N}.$$

By applying the Chain rule, we obtain the entry-wise derivative

$$\frac{\partial g(S)}{\partial N_{i,j}} = \sum_{k=1}^{n}\sum_{l=1}^{n} \frac{\partial g(S)}{\partial S_{k,l}} \frac{\partial S_{k,l}}{\partial N_{i,j}}.$$

Using matrix notation, this can be written as:

$$\frac{\partial g(S)}{\partial N_{i,j}} = \text{trace}\left[(\frac{\partial g(S)}{\partial S})^{\top} \frac{\partial S}{\partial N_{i,j}} \right].$$

Derivative of AMSE **with respect to** L_1

Using the Chain Rule for matrix derivatives, we need to take derivative of AMSE with respect to each element of L_1, say $L_1(i,j)$ and then construct $\frac{\partial \text{AMSE}}{\partial L_1} = \{\frac{\partial \text{AMSE}}{\partial L_1(i,j)}\}_{i,j}^{n}$, where

$$\frac{\partial \text{AMSE}}{\partial L_1(i,j)} = \text{trace}[(\frac{\partial \text{AMSE}}{\partial L})^{\top} \frac{\partial L}{\partial L_1(i,j)}].$$

From the formula above, we observe that both $\frac{\partial \text{AMSE}}{\partial L}$ and $\frac{\partial L}{\partial L_1(i,j)}$ have the size of $n^2 \times n^2$. Thus, the computational cost is pretty expensive when n is large. In order to reduce the computational cost, we need to do some further observation.

As an example, for the derivative with respect to the element $L_1(i,j)$,

$$\frac{\partial \text{AMSE}}{\partial L_1(i,j)} = \text{trace}(A_{j,i} L_2)$$

where $A_{j,i}$ is the $(j,i)^{th}$ $n \times n$ block of the matrix $A = \frac{\partial \text{AMSE}}{\partial L}$. Thus, the computational cost is reduced from $n^2 \times n^2$ to $n \times n$.

Derivative of AMSE **with respect to** L_2

Similarly, we take derivative of AMSE with respect to each element of L_2, say $L_2(i,j)$ and then construct $\frac{\partial \text{AMSE}}{\partial L_2} = \{\frac{\partial \text{AMSE}}{\partial L_2(i,j)}\}_{i,j}^{n}$, where

$$\frac{\partial \text{AMSE}}{\partial L_2(i,j)} = \text{trace}[(\frac{\partial \text{AMSE}}{\partial L})^{\top} \frac{\partial L}{\partial L_2(i,j)}].$$

Again, we can reduce the computational cost after some observation. As an example, for the derivative with respect to the element $L_2(i,j)$,

$$\frac{\partial \text{AMSE}}{\partial L_2(i,j)} = \text{trace}(B_{j,i} L_1)$$

where $B_{j,i}$ is the matrix constructed by all the $(j,i)^{th}$ elements in each $n \times n$ block in A. Again, the computational cost is reduced from $n^2 \times n^2$ to $n \times n$.

5.5 Numerical experiments

In this section, we experiment with different regularization matrices based on several sparsity patterns. Their performances are discussed and compared with other commonly used regularizations. The experiments will be carried out on first a 1D toy example and then a real-data MRI example.

5.5.1 The 1D magnetotelluric problem

For the 1D example, we use the magnetotelluric problem as is described in chapter 2 again. Since the problem is discretized into 256 points, the size of the regularization matrix is 256×256, which will give us totally $65,536$ entries. Because this number

is quite large, the goal of this problem is to find a sparse L that still leads to a satisfactory recovery of the model x.

For our numerical experiments, we provide both the covariance matrix and a set of training models so that the experiments can be done on both the covariance and training designs. In this experiment, we apply only the ℓ^1 norm pattern to show the power of sparsity control.

The covariance design

We start by showing results from the covariance design. The mean of the model is set to be 0. In order to check how well our approaches perform, we design the covariance matrix such that the model will have some peaks.

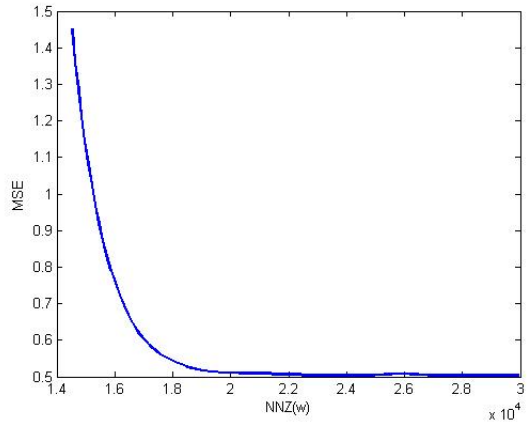

Figure 5.2: The risk as a function of sparsity $\text{nnz}(L) = \| L \|_0$ of the optimal L obtained with the covariance design.

Figure 5.2 shows the L-curve, in which we plot the AMSE as a function of the sparsity of L. As we can see, the AMSE decreases rapidly before the corner, which happens at $\| L \|_0 = 16,467$, as opposed to the total $65,536$ entries in the dense L pattern, and more slowly thereafter. Hence, this corner suggests an optimal

compromise between the image reconstruction quality and the sparsity of L.

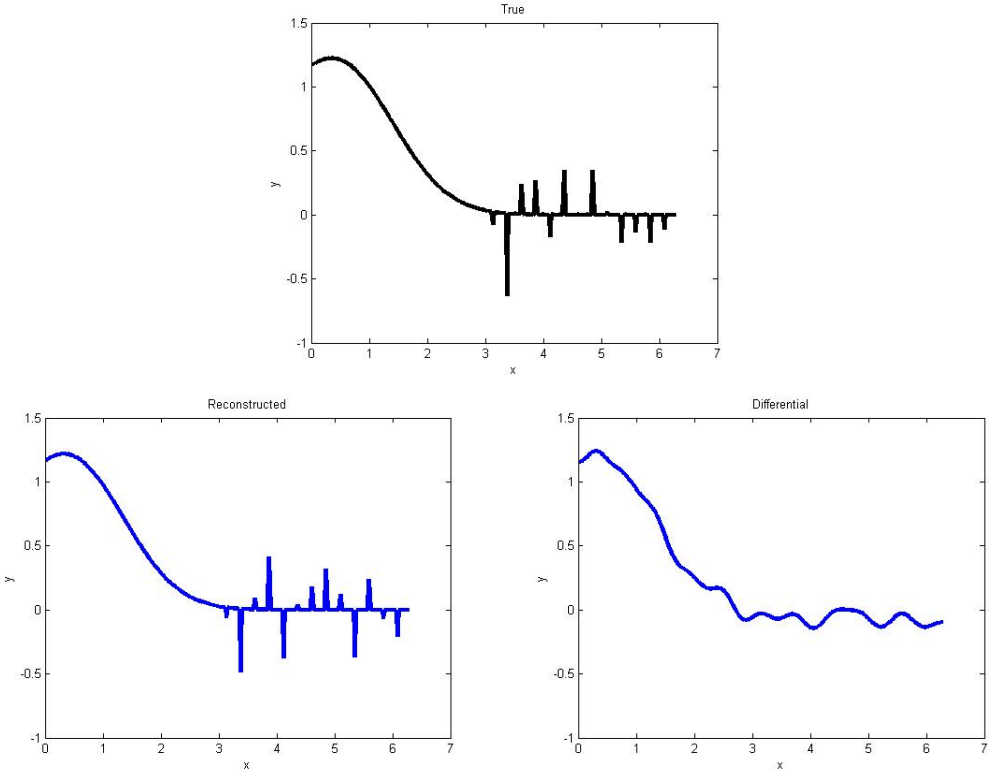

Figure 5.3: The true image (top), reconstructed image using the optimal $16,467$ entries (bottom left) and using the differential operator (bottom right).

Figure 5.3 shows the true model and we want to compare the reconstructed model from our approach to the result from the commonly used differential operator. Table 5.1 displays the relative errors between the true model and the reconstructed ones. The advantage of our result is obvious.

Table 5.1: Relative errors of the reconstructed images: covariance design

design	relative error
ℓ^1	8.19%
diff	17.02%

Also, we display the pattern of the sparse L together with the pattern of the

analytic L in Figure 5.4. We observe that the sparse L has kept the parts that mainly contribute to the result and saved half of the effort.

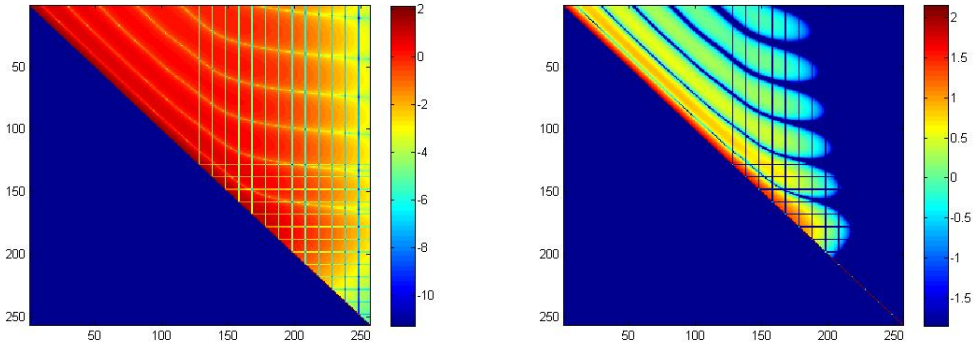

Figure 5.4: The left panel shows the pattern of the analytic L and the right panel shows the pattern of the sparse L .

The training design

In this section, we show results from the training design. In this case, the corner is located where the number of nonzeros of L is $23,802$ as is shown in Figure 5.5.

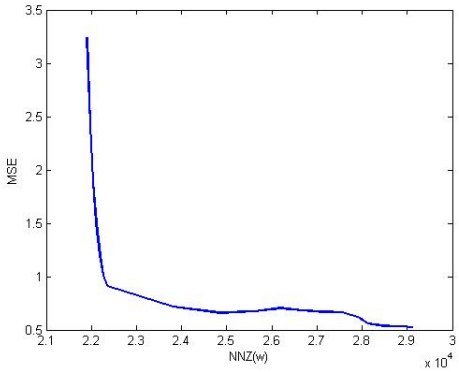

Figure 5.5: The risk as a function of sparsity $\text{nnz}(L) = \| L \|_0$ of the optimal L obtained with the training design.

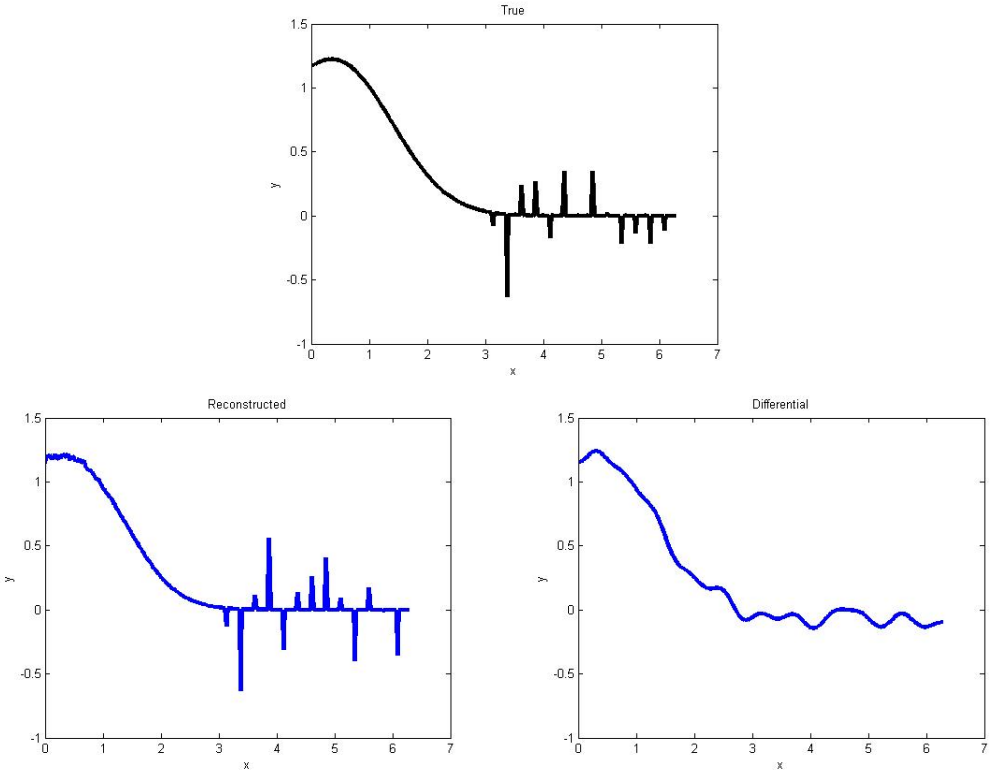

Figure 5.6: The true image (top), reconstructed image using the optimal $23,802$ entries (bottom left) and using the differential operator (bottom right).

Table 5.2: Relative errors of the reconstructed images: training design

design	relative error
ℓ^1	9.60%
diff	17.02%

From the resulting images in Figure 5.6 and Table 5.2, we can see that again our result is much better than the one from the differential operator.

5.5.2 An MRI example

In the second experiment, we will apply our algorithms on some MRI examples. Since this is a real problem and we do not have the true covariance matrix, we will

use the training design.

For the choice of the training models, we use the MRI data set in MATLAB which contains 27 different image slices of a human head. The slices go from the nose gradually to the top of the head.

In each experiment, we choose four different sparse patterns of L: the dense pattern, the commonly used differential operator, the Kronecker product pattern and the 5-diagonal pattern since this is a 2D problem. Figure 5.7 show the matrix $L^\top L$ of the 5-diagonal pattern and the differential operator.

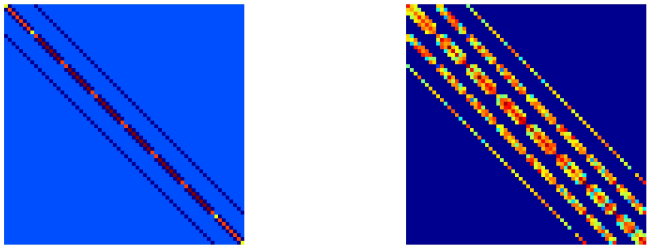

Figure 5.7: Images of $L^\top L$: the differential operator (left) and the 5-diagonal pattern (right).

In the differential operator case, both the structure and the entry values are fixed. In the 5-diagonal case, we only keep the structure of the matrix while all these entry values are decided through optimization. For the choice of α, we use the discrepancy principle to decide it before we start the iterations and set it as a constant during the optimization process.

Experiment 1

In the first experiment, we choose the 10^{th}, 11^{th}, 13^{th} and 14^{th} MRI examples to be our training models. Our goal is to use them to reconstruct the 12^{th} MRI, which

is similar to the training set. Figure 5.8 show the four training models we have used.

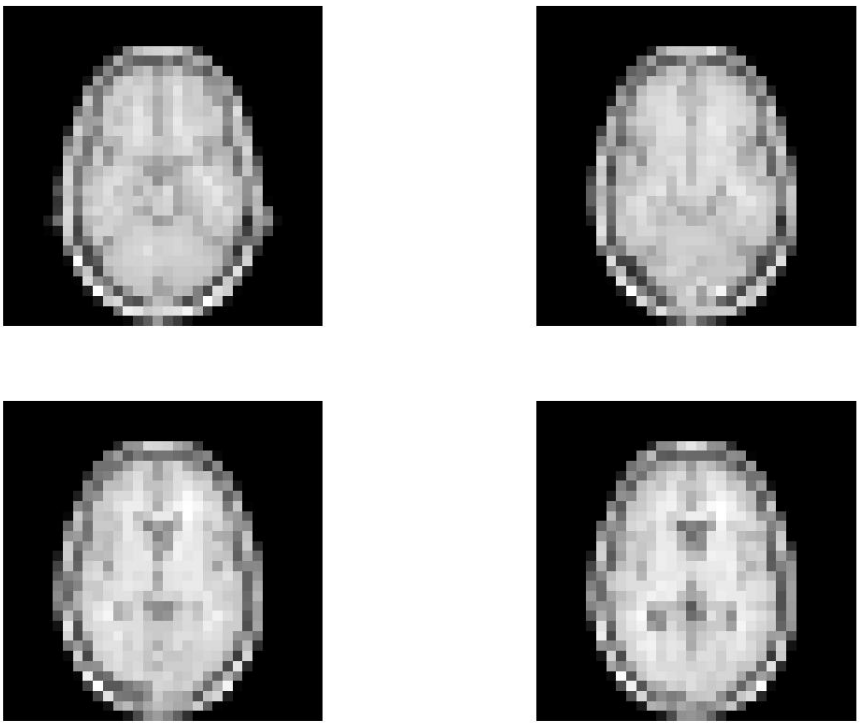

Figure 5.8: Four training MRI models: 10^{th} (top-left), 11^{th} (top-right), 13^{th} (bottom-left) and 14^{th} (bottom-right).

Here we display the true image and all the four reconstructed images from different patterns of L in Figure 5.9. The top panel is the true image, the middle-left panel is the reconstructed image from the dense pattern, the middle-right panel is from the differential operator, the bottom-left panel is from the Kronecker product pattern and the bottom-right panel is from the 5-diagonal pattern.

From the images, we observe that the dense pattern and the 5-diagonal pattern provide very nice reconstructions. The brain is very clear in these two images while it is just a big blur in the other two images. Also, we observe that the Kronecker

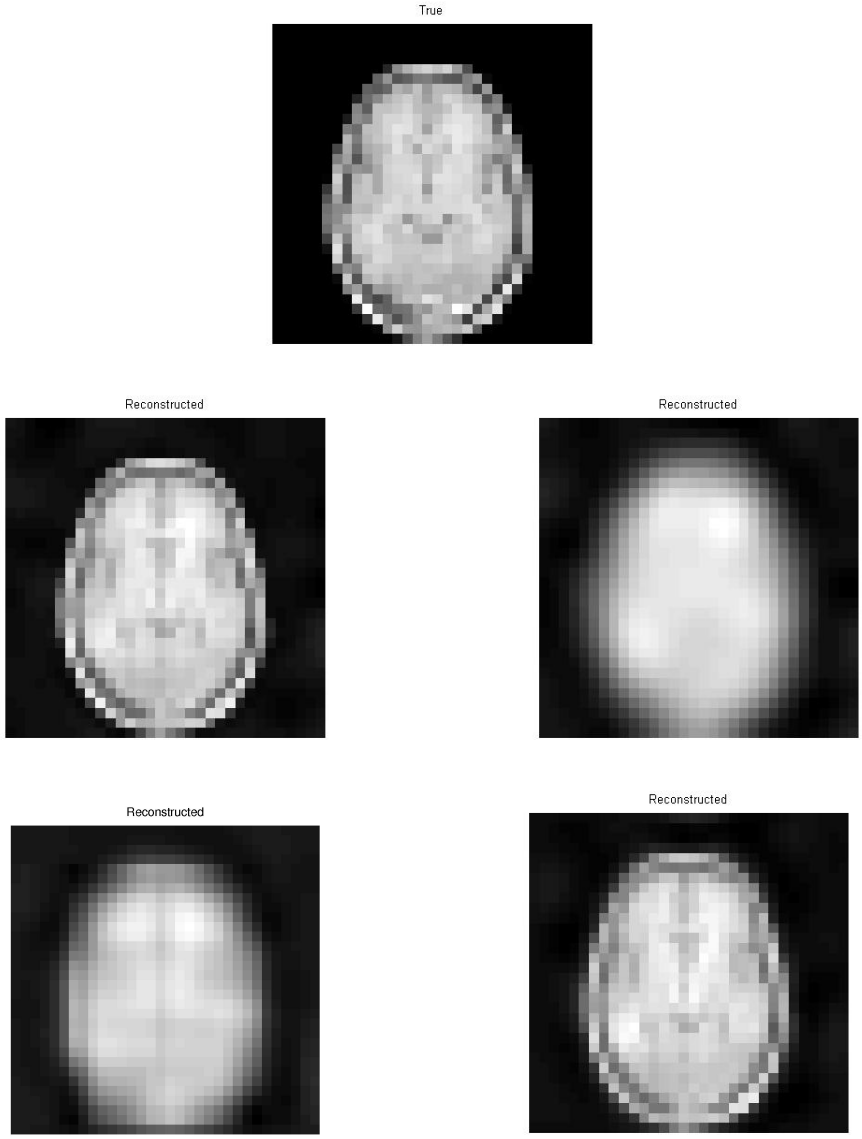

Figure 5.9: The top panel is the true image, the middle-left panel is the reconstructed image from the dense pattern, the middle-right panel is from the differential operator, the bottom-left panel is from the Kronecker product pattern and the bottom-right panel is from the 5-diagonal pattern.

product pattern works a little bit better than the simple differential operator.

The relative errors between all the reconstructed images and the true image

are listed in Table 5.3.

Table 5.3: Relative errors of the reconstructed images from four designs

design	relative error
diff	25.27%
Kron	21.65%
5-diag	12.29%
dense	9.96%

The explanation for the order of the reconstruction quality is straightforward: In the case of dense pattern, since no constraint is imposed on the regularization matrix L, the reconstruction should have the best possible quality for images that are similar to the training set. When it comes to the 5-diagonal pattern, based on the fact that each pixel has stronger connection with the four pixels that surround it than other pixels, we only consider information in the neighborhood, therefore accuracy may be decreased a little bit because we completely ignore the influence from the rest of pixels in the image. The reconstruction will get better and better as we increase the number of diagonals. For example, if we use nine diagonals, we should be able to get better results since now each pixel uses information from the eight pixels that surround it. However, although the reconstruction quality of the dense pattern is a bit better, the computational cost becomes tremendous.

In the case of the Kronecker product pattern, each column in the image is treated in the same way by the small matrix L_2 and each row is treated in the same way by L_1. For most image examples, this certainly will not work well. That is why the reconstruction from the Kronecker product pattern is much worse than the 5-diagonal pattern. The same thing happens in the case of differential operator. Moreover, it only considers the one-sided difference while the Kronecker product pattern considers the centered difference. Thus, it is even worse than the

Kronecker product pattern. Therefore, we have demonstrated that the order of reconstruction quality is as expected.

Therefore, we say our 5-diagonal pattern seems to be the optimal choice of the regularization matrix. It is very cheap in computation and the results are almost as good as the ones from the dense pattern.

One thing interesting to observe is the diagonal of the matrix $L^\top L$ from the optimal L we have obtained from the 5-diagonal pattern, as is plotted in Figure 5.10.

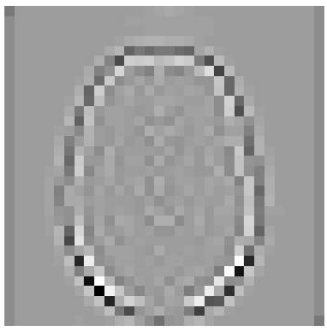

Figure 5.10: The image of the diagonal of the matrix $L^\top L$.

We analyze the meaning of the figure as follows. Write the matrix L as

$$L = [l_1, l_2, \cdots, l_n],$$

where l_i denotes a column of the matrix L. Hence,

$$\text{diag}\,(L^\top L) = [l_1^\top l_1, l_2^\top l_2, \cdots, l_n^\top l_n].$$

On the other hand, we observe that

$$\begin{aligned} \| Lx \|^2 &= L_{11}^2 x_1^2 + L_{12}^2 x_2^2 + \cdots + L_{1n}^2 x_n^2 \\ &\vdots \\ &\quad + L_{n1}^2 x_1^2 + L_{n2}^2 x_2^2 + \cdots + L_{nn}^2 x_n^2, \end{aligned}$$

which gives us

$$\begin{aligned} \| Lx \|^2 &= (L_{11}^2 + \cdots + L_{n1}^2) x_1^2 + \cdots + (L_{1n}^2 + \cdots + L_{nn}^2) x_n^2 \\ &= l_1^\top l_1 x_1^2 + \cdots + l_n^\top l_n x_n^2. \end{aligned}$$

Hence, we have shown that image of diag $(L^\top L)$ shows the amount of penalty that is added to each pixel of the image.

Experiment 2

We notice that, in the previous experiment, the training models we have chosen are very similar to the model we want to reconstruct. Unfortunately, for an unknown image, we do not know if it is similar to or different from the training models we have. What we could do is to gather lots of training models with relatively different information and see what will happen. 'Relatively different' means they cannot be too different. For example, if we are trying to reconstruct the image of a patient's brain, then images containing different slices of the brain or views from different angles will be a good set of training models, while images of a tree certainly will not be helpful.

In this experiment, we randomly chose twenty images out of the twenty-seven MRI examples in Matlab to be our training models and the goal is to recover

the remaining six images. In Table 5.4 we list the relative errors between each reconstructed image from the simple differential operator, the 5-diagonal pattern and the true one.

Table 5.4: Relative errors of the reconstructed images from two designs

	1	8	9	13	19	24
diff	34.57%	33.95%	29.43%	24.12%	25.28%	34.03%
diag	32.42%	24.65%	23.38%	19.19%	20.33%	36.25%

By comparing the relative errors, we see that our approach works quite reasonably. The improvement differs in each image due to the different levels of similarity of that image to the training set. In real applications, we expect even better results because now we are taking different slices of the whole head while, in real cases, for example, a brain doctor will only take images of the brain part. Hence, the choice of training models will be more appropriate, which will result in better reconstruction quality.

Chapter 6

Optimal design in CO_2 injection monitoring

As we have mentioned in the introduction, in order to reduce the Greenhouse effect, companies and organizations have piped the liquefied CO_2 into underground storage, which brings up the need for the research into monitoring techniques to verify that the CO_2 remains effectively trapped underground.

It has been shown that time-lapse seismic imaging is an effective technique for underground CO_2 monitoring [3, 31]. Borehole seismic techniques can image the change in seismic velocity caused by the movement of CO_2 [17].

However, due to the huge expense in deployment of the monitoring sensors, it is imperative for us to find a cost-effective arrangement for placing the sensors while the quality of the images is still guaranteed. Therefore, in this chapter, we will present an approach to find the optimal design for CO_2 injection monitoring.

6.1 Crosswell array constraints

We begin this chapter with a brief description of the basic structure of the experimental setting used in CO_2 injection monitoring. Figure 6.1 provides a schematic of the crosswell tomography experiment.

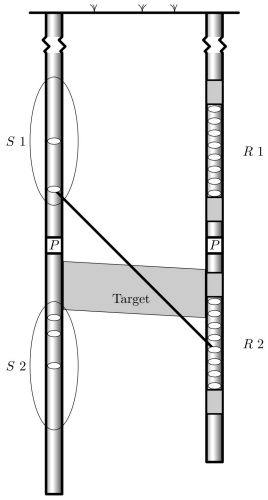

Figure 6.1: A schematic of the crosswell tomography experiment.

This schematic has brought us various crosswell array constraints that have to be taken into account when designing the CO_2 injection monitoring. We mention some major constraints in the following:

- **Fixed packers:** The seismic sensors are deployed on production tubing, both above and below a packer in the observation well. The locations of packers in both observation wells are fixed.

- **Jewelry on tubing:** The pressure compensator is at the bottom of each receiver array and the pre-amplifier is at the top. They both have their individual sizes.

- **Well boundary:** The possible positions of the sensors are bounded from both above and below.
- **Offset between the wells:** The distance between the two wells is also fixed.
- **Equally spaced receivers:** Although sources can be placed almost anywhere, the two groups of receivers must have equal distances between consecutive pairs (within the groups).
- **Length of sensors:** Each sensor has its individual length which cannot be ignored.
- **Maximum ray length:** The maximum length of any possible ray path is not allowed to exceed certain limit.

All these constraints bring significant difficulties in the optimal design.

6.2 Mathematical framework

In this section, we discuss the mathematical framework of the CO_2 injection monitoring problem. Before the formulation is proposed, we will describe a bit about the generation of the forward modeling operator.

6.2.1 The forward modeling operator

If the underground CO_2 saturation and/or plume thickness increase along a given raypath, the traveltime of a ray would decrease. From the time-distance data, a tomographic reconstruction of the wave speed can be carried out by using sensitivity kernels obtained in the ray approximation.

When we discretize the region of interest, the integrated sensitivity kernel in a given grid cell becomes the length of the ray path in that cell. Therefore, the quality of the imaging reconstruction depends on the accuracy of the sensitivity kernels employed.

We apply the Fresnel-Zone-Based Kernel [48] hence, the sensitivity function is formulated as

$$S(x) = KA(s,x)A(x,r)\cos\left(\frac{2\pi\Delta t(x)}{T}\right)\exp\left(-\left(\frac{\alpha\Delta t(x)}{T/4}\right)^2\right),$$

where $S(x)$ is the sensitivity at the position vector x, A represents the amplitude of the wave field propagation and K is a normalization constant. The parameter α controls the degree of cancellation in Fresnel zones beyond the first, $\Delta t(x)$ stands for the delay time between the first arrival and the arrival of waves that have been scattered at x and T is the dominant period of the wave.

6.2.2 Formulation of the inversion

The CO_2 injection monitoring problem can be expressed by this inverse problem,

$$Gm + \epsilon = d,$$

where G, is a matrix in which each row represents a tomographic ray path that connects a source to a receiver. We calculate G in a 2.5 dimensional geometry designed to accommodate heterogeneous 2D structure and well trajectories with out-of-plane deviations. The solution to the eikonal equation is calculated using a finite-difference eikonal solver [41, 47]. Our numerical experiments are based on

the FAST package developed by Zelt [78]. The vector m is the geophysical model we would like to reconstruct, ϵ is the noise that is assumed to be Gaussian iid with 0 mean and standard deviation σ and finally d is the data we have observed.

Due to the existence of the complicated spatial constraints that we have raised in the previous section, we cannot hope to control the quality of the reconstructed images by simply abandoning certain raypaths or adjusting the frequency with which certain raypaths is chosen. Therefore, the optimal design that uses a weight matrix W as was described in chapter 2 will not be helpful. Therefore, we intend to explore a method to gradually modify the kernel matrix G in each iteration as we update the placement of sensors, thereby converging to an optimal kernel G with a corresponding optimal placement of sensors.

Since geophysical inverse problems are usually ill-posed, regularization in general is needed. In this problem, we apply the Tikhonov regularization framework as was described in chapter 4. Hence, an estimated model is given by

$$\hat{m} = \text{argmin } (d - Gm)^\top (d - Gm) + \alpha \parallel Lm \parallel^2 .$$

In this case, we set L to be a discrete derivative operator to introduce smoothness to the recovered model. The solution of the above problem is given by

$$\hat{m} = (G^\top G + \alpha L^\top L)^{-1} G^\top d.$$

In order to determine the quality of the estimates, we calculate the ℓ^2 norm difference between the true model and the corresponding tomographic reconstructed one for a given geometry. In many cases, the desired information is usually only

within a particular zone of the whole model, therefore the quality measurement is only considered within the target zone:

$$D = \| Q \odot (\hat{m} - m_{true}) \|_2^2, \tag{6.1}$$

where Q is a window function that is used to specify the zone that we are interested in.

The optimality criterion is defined by the average MSE that is given in the following

$$\begin{aligned} \text{MSE} &= \alpha^2 \| Q \odot (G^\top G + \alpha L^\top L)^{-1} L^\top L m_{true} \|_2^2 \\ &+ \sigma^2 \text{trace}[G(G^\top G + \alpha L^\top L)^{-1} \odot Q^\top Q \odot (G^\top G + \alpha L^\top L)^{-1} G^\top]. \end{aligned} \tag{6.2}$$

In most real applications, the true model in the AMSE formulation above is not available. In its place, we use a suite of reference models that are related to the model to be reconstructed and solve the problem via a training approach.

$$\begin{aligned} \text{AMSE} &= \frac{1}{s} \sum_{j=1}^{s} \alpha^2 \| Q \odot (G^\top G + \alpha L^\top L)^{-1} L^\top L m_{true,j} \|_2^2 \\ &+ \sigma^2 \text{trace}[G(G^\top G + \alpha L^\top L)^{-1} \odot Q^\top Q \odot (G^\top G + \alpha L^\top L)^{-1} G^\top], \end{aligned} \tag{6.3}$$

where s is the number of reference models.

Therefore, the main purpose of this work is to determine an optimal placement of sources and receivers in order to provide maximum information about the target geophysical structure at minimum financial cost under the spatial constraints. In particular, the best possible estimated model is obtained by minimizing the AMSE through optimization approaches and the optimal reconstructed model is obtained

based on the resulting optimal design thereafter.

6.3 Numerical optimization through DIRECT algorithm

Obviously standard gradient-based optimization methods are not very effective for solving our problem due to the existence of those constraints on the solution domain. In stead, Ajo-Franklin [1] has chosen two non gradient-based optimization methods: the Nelder-Mead downhill simplex method [57] and the multilevel coordinate search algorithm [46]. However, the work did not provide a way to find the optimal locations of sensors. Also the discussion of location constraints was not presented.

6.3.1 The DIRECT algorithm

In order to accommodate those location constraints for placing sensors, we apply the DIRECT algorithm, which was proposed by Daniel E. Finkel [26]. The DIRECT optimization algorithm was first introduced in [49] based on the Lipschitzian Optimization for solving difficult global optimization problems with bound constraints and a real-valued objective function.

The algorithm converges to the global minimum of the objective functional without requiring its gradient. It samples points in rectangular domains by taking the midpoints of the searching spaces. This overcomes the shortcoming of the Lipschitzian Optimization when solving high dimensional problems. The DIRECT algorithm uses all values in the current rectangle to determine if a region of the

domain should be broken into sub-rectangles. Therefore, it does not require the estimate of the Lipschitz constant or the continuity of the objective functional [49].

In a very brief description, DIRECT algorithm starts with a unit hyper-cube and divides it into smaller hyper-rectangles with the restriction that the division is only done along the longest dimension(s) of the hyper-rectangles. This ensures that the rectangles will be divided along all directions and no dimension will be ignored.

For our application of the DIRECT algorithm, we only need to provide the evaluation of the objective functional and the upper and lower bounds of the variables we intend to estimate. Thus, the crosswell ray constraints can be easily satisfied if we carefully choose the variable bounds.

6.3.2 Discussion of the constraints

We now discuss details about finding the appropriate bounds for locations of sources and receivers. First, we need to determine the size of the kernel G in the inverse problem, which represents all possible locations of sources and receivers.

Based on the knowledge of the maximum ray length (we denote as l_{MaxRay}) and the offset (d_{off}) between the two wells, we obtain the maximum distance that a raypath can reach along the well direction $d_{\max} = \sqrt{(l_{\text{MaxRay}})^2 - (d_{\text{off}})^2}$.

In the source well, we assume that the lowest source above the packer can be placed right on top of the packer and the highest source below the packer can be placed right under it. Similar assumptions work for the receiver well as well. Also we denote l_{source} to be the length of the sources and l_{receiver} to be the length of the receivers.

Thus, in order to guarantee that all rays from the sources are able to reach all receivers, we must not place any sensors in areas that a ray cannot reach. Therefore, the upper bound on the lowest receiver below the packer (we denote as R_{bottom}) is at: top of the packer in the source well - $l_{\text{source}}/2 + d_{\max}$. Similarly, the lower bound on the highest receiver above the packer (R_{top}) is at: bottom of the packer in the source well + $l_{\text{source}}/2$ - $d_{\max}$. Following the same approach, the upper bound on the lowest source below the packer (S_{bottom}) is determined to be at: top of the packer in the receiver well - l_{attach} - $l_{\text{receiver}}/2 + d_{\max}$ and the lower bound on the highest source above the packer (S_{top}) is at: bottom of the packer in the receiver well + $l_{\text{attach}} + l_{\text{receiver}}/2$ - $d_{\max}$, where l_{attach} is the length of the attached materials on receivers such as the pressure compensator and the pre-amplifier.

The size of the kernel G, or in other words, the size of the space for all possible locations of sources and receivers is decided by the above four bounds.

Boundaries for sensor location constraints

Since the sources can be placed at any location within the space provided by the kernel G, we must determine individual bounds for each of them. We assume that the number of sources above the packer is n_{Stop} and the number below the packer is n_{Sbottom}. Similarly, we assign numbers to n_{Rtop} and n_{Rbottom} as well.

We start with the possible locations for the highest source. Its lower bound is simply S_{top} and its upper bound can be calculated as: top of the packer in the source well - $l_{\text{source}}/2$ - $(n_{\text{Stop}} - 1)\ l_{\text{source}}$. With respect to the second highest source, the lower bound is $S_{\text{top}} + l_{\text{source}}$ because the space to hold the highest source must be kept. Similarly, its upper bound is l_{source} more than the upper bound of the highest source. Following the same routine, it is straightforward to obtain bounds

for all sources.

For the receivers, since each group is equally spaced, we only need the four bounds on the top and bottom receiver of the group above the packer and below it. The lower bound for the top receiver in the top group is simply R_{top} and its upper bound is calculated as: top of the packer in the receiver well - l_{attach} - $l_{\text{receiver}}/2$ - $(n_{\text{Rtop}} - 1)l_{\text{receiver}}$. The other bounds can be determined in the same fashion. Finally, after translating the absolute bounds into the coordinate system of the kernel G, we put the bounds for all sources and receivers in a $n_{\text{Stop}}+n_{\text{Sbottom}}+ n_{\text{Rtop}} + n_{\text{Rbottom}}$ by 2 matrix.

Special treatment for receivers

Above, we have decided the general boundaries for placing sources and receivers. However, due to the complicated constraints we have listed in section 6.1, there are certain places inside the solution interval between upper and lower bounds where we cannot place sensors. Hence, special treatments need to be considered when finding the optimal placement. Since sources and receivers have different spatial constraints, different treatments need to be carried out in the two cases.

Let us start with the easy case: the receivers. We use the top group of receivers as an example. There are two things we need to check every time we update to a new arrangement of receivers. First, if the distance between the first and last receivers in this group is less than $l_{\text{receiver}}(n_{\text{Rtop}}-1)$, we stretch the whole group until this constraint is satisfied. Here, distance between two sensors means the distance between the centers of the two. Since, in our problem, we stretch the receivers downwards, we need to check the position of the last receiver. If it is below the top of the packer, we will move all receivers up until this conflict is resolved. Of

course, it is possible to stretch the receivers upwards as well. However, since the target zone is usually in the middle area between the two wells, it is natural for us to always try to gather sensors close to the target zone. Therefore, for the top group of receivers, we stretch downwards.

A symmetric process is performed on the bottom group of receivers, for which we stretch the conflicting receivers upwards. In this case, we need to check the position of the first receiver in this group. If it is above the bottom of the packer, we will move all receivers down until this conflict is resolved.

Special treatment for sources

Things are more complicated for sources. If the DIRECT algorithm generates the distance between two sources to be less than l_{source}, the resulting system is not feasible because these two sources will overlap each other. In order to fix this problem, we suggest a perturbation process. Again, we use the top set of sources as an example. A symmetric process can be applied to the group below the packer.

If two (or more) sources are conflicting, we arbitrarily choose one and move it to the closest position, within the space between the packer and the ceiling of the working space, which does not conflict with another source. This process is repeated until no conflicts remain among all sources.

Note that no source will be moved more than the distance of $(n_{\text{Stop}}-1)(2l_{\text{source}}-1)$. The proof of this claim is given as well.

Proposition 6.1 *Each source moves at most* $(n_{\text{Stop}}-1)(2l_{\text{source}}-1)$.

Proof: First note that the DIRECT algorithm will not place any sources in conflict with the packer. Thus, if $n_{\text{Stop}} = 2$ and the two sources are in conflict, we

may move one source to the opposite side of the other, thereby moving the source by at most $2l_{\text{source}} - 1$. This would be the case if source 1 is located right next to the packer while source 2 is at a distance of $l_{\text{source}} - 1$ away.

If $n_{\text{Stop}} > 2$, the largest possible distance that a source could move is clearly $(n_{\text{Stop}}-1)(2l_{\text{source}}-1)$, which is the situation when the first source to be considered is still located right next to the packer, the second is $l_{\text{source}} - 1$ away, the third is $2(l_{\text{source}} - 1) + l_{\text{source}}$ away and so on. In this situation, the i^{th} source would be located $(i-1)(l_{\text{source}} - 1) + (i-2)l_{\text{source}}$ from the packer for all $i \geq 2$. □

Unfortunately, this process does not always yield a solution that is as close as possible to the output of the DIRECT algorithm, but we can easily bound the difference. We claim that the total movement of the sources is less than $n_{\text{Stop}}^2 l_{\text{source}}$. The proof is given in the following.

Proposition 6.2 *The sum of the movements of all the sources is at most* $\frac{n_{\text{Stop}}(n_{\text{Stop}}-1)(2l_{\text{source}}-1)}{2}$.

Proof: We prove this result by induction on n_{Stop}. First suppose $n_{\text{Stop}} = 2$. Then, by Proposition 6.1, one source will move at most $2l_{\text{source}} - 1$. If one source moves, the other will clearly remain unmoved so the total movement is at most $2l_{\text{source}} - 1 = \frac{n_{\text{Stop}}(n_{\text{Stop}}-1)(2l_{\text{source}}-1)}{2}$.

Now assume $n_{\text{Stop}} > 2$ and suppose the result holds for fewer than n_{Stop} sources. Suppose we have $n_{\text{Stop}} - 1$ sources placed and we would like to add a new source. By Proposition 6.1, this source could potentially have to move at most $(n_{\text{Stop}} - 1)(2l_{\text{source}} - 1)$. The total of the other sources movements, by induction, is at most $\frac{(n_{\text{Stop}}-1)(n_{\text{Stop}}-2)(2l_{\text{source}}-1)}{2}$ for a total of

$$\begin{aligned} & \frac{(n_{\text{Stop}}-1)(n_{\text{Stop}}-2)(2l_{\text{source}}-1)}{2} + (n_{\text{Stop}}-1)(2l_{\text{source}}-1) \\ = \quad & \frac{n_{\text{Stop}}(n_{\text{Stop}}-1)(2l_{\text{source}}-1)}{2}, \end{aligned}$$

as desired. □

Of course, there are cases where other algorithms will produce results that are closer to the output than our method. However, the computational complexity of these algorithms is orders of magnitude larger than the complexity of our algorithm.

At each step in our correction process, we move a source one distance unit in both directions and then check its position relative to all the other sources along with the packer and the outer boundary of the well (either the surface of the ground if these sources are above the packer or the bottom of the well if they are below). Thus, for each unit moved, we check a total of $2(n_{\text{Stop}}+1)$ pairs for conflicts. This makes the computational complexity of our algorithm exactly

$$\begin{aligned} & 2(n_{\text{Stop}}+1)\frac{n_{\text{Stop}}(n_{\text{Stop}}-1)(2l_{\text{source}}-1)}{2} \\ = \quad & n_{\text{Stop}}(n_{\text{Stop}}+1)(n_{\text{Stop}}-1)(2l_{\text{source}}-1) \\ = \quad & O(n_{\text{Stop}}^3 l_{\text{source}}). \end{aligned}$$

There are two obvious limitations in the method that we have chosen. One drawback of our algorithm is that, in some cases, the order in which the sources are considered causes the movement to increase. Hence, one may consider all possible

orderings of the sources and choose the ordering which minimizes the movement. Unfortunately, the number of such orderings is $n_{\text{Stop}}!$ so this would greatly increase the computational complexity to $O(n_{\text{Stop}}! n_{\text{Stop}}^3 l_{\text{source}})$. Also, consider the following extreme example. Suppose all sources start in the same position, right next to the packer. Call this example A. In this case, our correction process finds the best possible configuration regardless of the order in which the sources are considered. This example shows that considering all orderings of the sources would likely be a waste of effort.

Another drawback of our approach is that there are cases where moving one source at a time cannot produce the best possible result. For example, suppose $l_{\text{source}} = 3$, source 1 starts at point 0, source 2 starts at point 4, source 3 starts at point 5 and source 4 starts at point 9 (see Figure 6.2). Clearly sources 2 and 3 are in conflict but moving only one source at a time, as in our algorithm, cannot place these two in their optimal positions (clearly 3 and 6 respectively). Thus, a slightly more sophisticated algorithm would allow pairs of sources to move at the same time, but this would have a computational complexity of at least $O(n_{\text{Stop}}^4 l_{\text{source}})$ since we would have to consider all possible pairs of sources which introduces an n_{Stop}^2 term as the number of pairs where we considered only the n_{Stop} sources before. Furthermore, as seen in example A, this more complicated algorithm could not always provide better results than our approach.

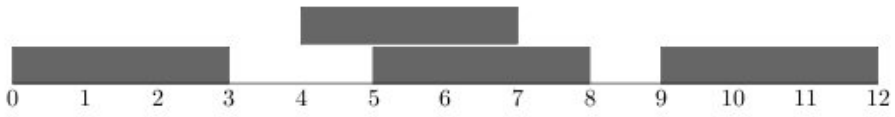

Figure 6.2: An example of source placement.

The process we propose is very fast even when the number of sources gets large. Therefore, we feel that our repair process provides a sufficient balance between computational cost and the result accuracy.

6.4 Numerical experiments

For our experiment we have generated four time-lapse images of a CO_2 flood progressing through a permeable layer as is shown here. We use the top three images as our reference models and we want to recover the one on the bottom row.

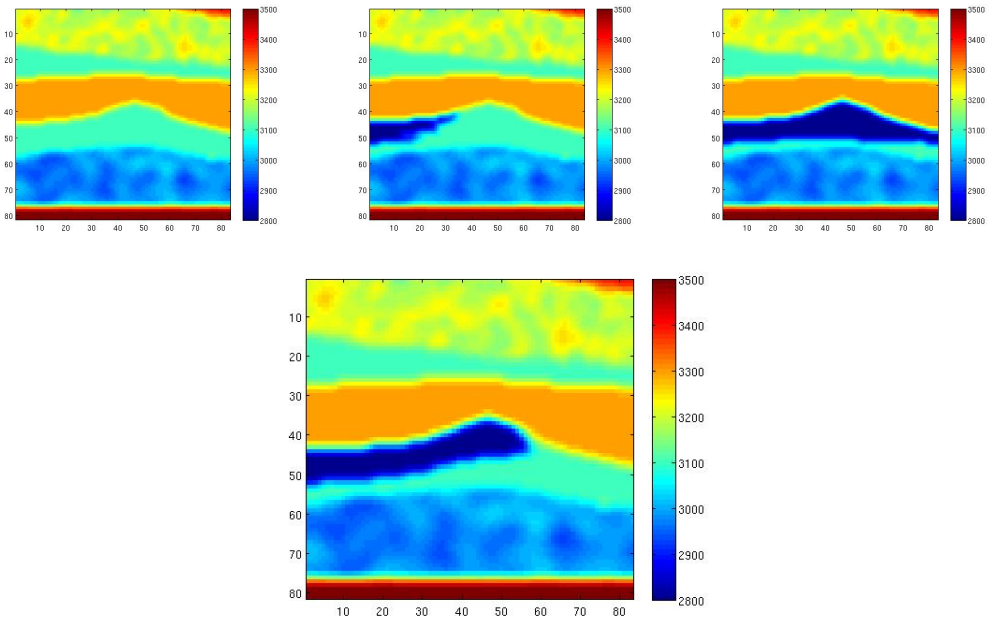

Figure 6.3: Four time-lapse images of a CO_2 flood progressing through a permeable layer. The top three images are our reference models and the goal is to recover the one on the bottom row.

We designed a crosswell continuous active-source seismic monitoring (CASSM) experiment in which data would be acquired continuously during injection along a set of fixed raypaths.

We give a list of parameters we have used in our experiments. We assume that the length of the attached materials on the receiver groups is 2 feet. The maximum length of raypaths is 150 feet and the offset between the two wells is 80 feet. Hence, we obtain that the maximum distance that a raypath can reach along the well direction is $d_{\max} = 126$ feet. The length of both sources and receivers is 2 feet. The model images have the size of 81 feet by 83 feet. The length of packers is 2 feet and they are located at pixel 38 to 40. We discretize the well space into grids of size 1 foot by 1 foot.

After the generation of the bounds for all sources and receivers, we apply the DIRECT algorithm and terminate the iteration when the difference between the two consecutive objective functional values is less than a certain tolerance. We set the maximum number of iterations to be 100, the maximum number of functional evaluations to be 2000 and the maximum number of rectangle divisions to be 1000. The Jones parameter is 10^{-5}.

The homogeneous background velocity model is provided by the Cranfield CASSM experiment. It is based on the extrapolation of the CFU 31F-1 sonic extrapolated along a -1.9 degree dipping to the F_2 and F_3 well section. All experiments are done using the infinite frequency kernel.

It is evident that the kernel G is different in each iteration of the DIRECT algorithm due to the change in the locations of sources and receivers. Thus, the goal of the optimal experimental design in this work is to generate the optimal kernel G that gives us the best reconstructed image with reasonable financial cost.

6.4.1 Numerical Experiment 1: $S/R = 0.5$

We first show the case in which the ratio of numbers of sources and receivers is fixed. In our case, we set the ratio to be 0.5.

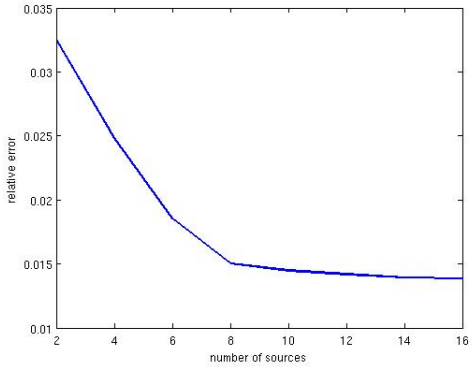

Figure 6.4: L-curve for CO_2 injection monitoring.

We have tested our algorithm on different numbers of sources, with the number ranging from 2 to 32. The corner of the L-curve happens when we use eight sources and sixteen receivers. It is obvious that this placement has cut out half of the financial cost while the result quality is almost as good.

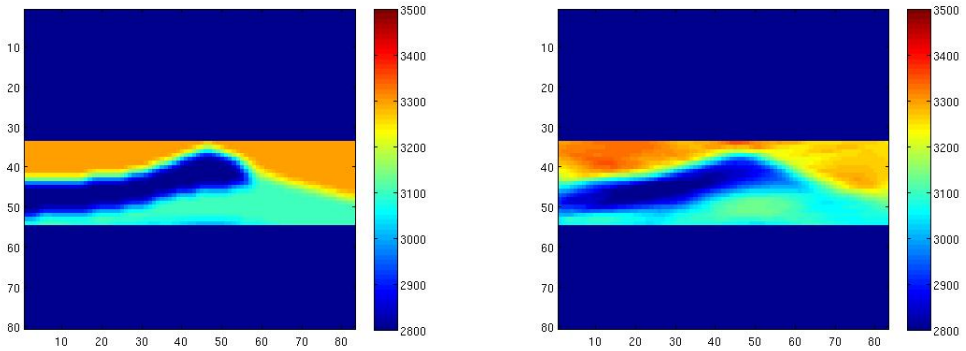

Figure 6.5: The true image (left) and the reconstructed one within the interest zone based on the optimal number of source and receivers (right).

We display the true image and the reconstructed one within the interest zone based on the optimal number of source and receivers. The reconstruction quality is quite satisfying. The relative error is only about 1.5%.

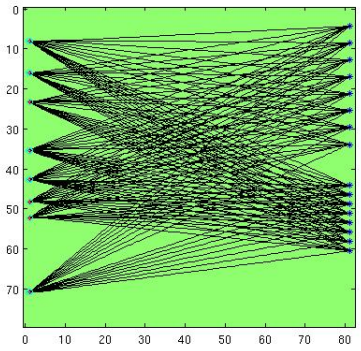

Figure 6.6: The resulting raypaths based on the optimal sources and receivers.

The bounds for placing the four sources above the packer are $[1, 31]$, $[3, 33]$, $[5, 35]$, $[7, 37]$ and $[41, 72]$, $[43, 74]$, $[45, 76]$, $[47, 78]$ for the four sources below the packer. For receivers, the bounds are $[1, 21]$, $[15, 35]$ for the top group and $[43, 64]$, $[57, 78]$ for the bottom group. In Figure 6.6 we show the resulting raypaths based on the optimal placement of sources and receivers. The position for the four top sources are at 8, 16, 23.3, 35.3 feet from the top of the image and 42.7, 48.2, 70.8, 52.2 feet away for the bottom four sources. For receivers, the highest one in the top group is located at 4.3 feet away from the top of the image and the lowest receiver is located at 33.9 feet away. For the bottom group, the highest one is at 44.2 feet away and the lowest one is at 60.5 feet away.

6.4.2 Numerical Experiment 2: S/R unfixed

In general, there is no fixed ratio between numbers of sources and receivers; this means that the relative error is a function of two independent variables: the number of sources and the number of receivers. In fact, each of these variables can be split into top and bottom groups. This will give us a 4D hyper-curve, which is very difficult to determine where the corner point of the curve is.

In order to still obtain an L-like curve so that the optimal solution is obvious, we consider plotting the relative error as a function of the financial cost of the sources and receivers. Since sources and receivers have different costs and these costs also differ based on the placement relative to the packer, we assign values and simply total the cost. For each total cost, we choose the smallest relative error among the trials we ran.

In the absence of real cost data, we have assigned a cost of 10 monetary units for a source above the packer and a cost of 20 units for a source below the packer. Also we assign costs of 5 and 10 units for receivers above and below the packer, respectively. The resulting L-curve is shown in Figure 6.7.

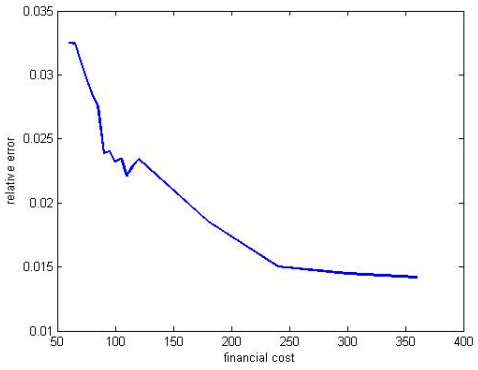

Figure 6.7: The L-curve shows the relative error as a function of the financial cost of the sources and receivers.

According to the L-curve, we choose the point corresponding a monetary value of 110 units, which yields a relative error of 2.21%. In our experiment, this corresponds to placing two sources above the packer and three below, also two recivers both above and below the packer.

This curve is very useful in practice because the amount of funding available is frequently bounded and thus, one can easily find a point on this curve which corresponds to the best reconstruction quality possible within the budget.

Figure 6.8 displays the true model together with the reconstructed one based on the optimal placement we have chosen within the interested zone. From the result, we observe that the optimal placement has cut out more than 2/3 of the financial cost while still maintaining a quite good reconstruction quality.

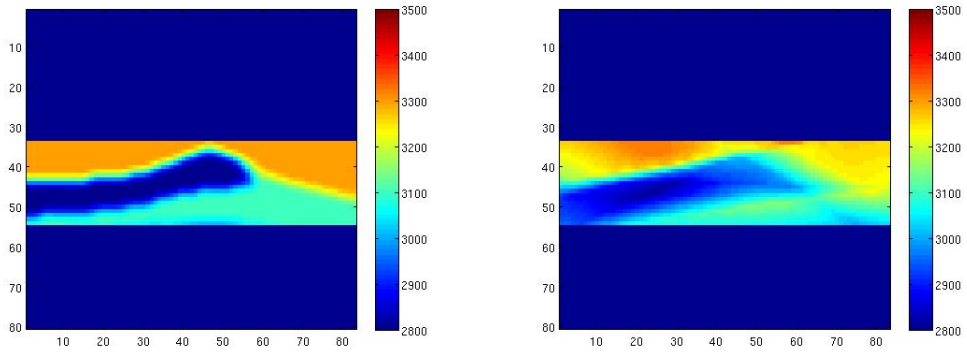

Figure 6.8: The true image (left) and the reconstructed one within the interest zone based on the optimal number of source and receivers (right).

Chapter 7

Summary and future work

The goal of this thesis was to develop numerical methods for optimal design problems. In particular, we have proposed new criteria of optimal design approaches for applications in areas such as medical imaging and geophysics.

First, in Chapter 2, we have introduced two new design criteria for the A-optimal design that minimize the Bayes risk based on sparsity. Also, we have developed numerical approaches based on Krylov subspace methods for large scale ill-posed problems. For numerical experiments, we have tested our algorithms on a 1D problem as well as a large scale super resolution problem and demonstrated the efficiency of our approach.

In Chapter 3, we have developed numerical methods for the E-optimal design. Inverse iteration has been applied to approximate the smallest eigenvalue of the information matrix and the corresponding derivative techniques have been discussed. Numerical experiments have been done on the same 1D problem as was used in Chapter 2 for the purpose of comparison and also a borehole ray tomography example.

In Chapters 4 and 5, an optimal ℓ^2 regularization approach has been proposed and two designs based on availability of different types of prior information have been developed. Also, several sparsity patterns have been imposed on the optimal regularization matrix in order to reduce the computational cost with very little loss of reconstruction quality. Matrix-based derivative techniques have been discussed in detail for all different sparsity patterns and several numerical experiments have shown that our optimal regularization approach gives very promising results. Especially when the model to reconstruct is similar to the training set, the training design provides much better results than the traditional differential operator.

In Chapter 6, we have applied our optimal design idea to a geophysical problem: CO_2 injection monitoring. Special spatial constraints on placement of sensors have been introduced and different treatments have been discussed for sources and receivers, respectively. We have done experiments on a synthetic problem based on a target zone and the results are quite promising.

Future work to be done is to continue developing numerical methods for solving optimal experimental design problems. There are many other optimality criteria that have proved to be very useful for various applications. One of the popular examples is the D-optimality. As far as we know, there have not been any numerical methods developed for solving ill-posed large-scale problems. My goal is to explore efficient methods for large-scale determinant optimization.

Also, we intend to improve the optimal regularization techniques for solving nonlinear inverse problems. The difficulty arises because, in this case, there is no simple decomposition of the bias and variance; hence, there is no closed formula for the MSE. Also, the formulation will involve a bi-level optimization problem, which is very difficult to solve.

Furthermore, since the CO_2 problem we mentioned is in a large class of real-world discrete optimization problems, we intend to further develop algorithms to solve discrete optimal experimental design problems.

Bibliography

[1] J. B. Ajo-Franklin. Optimal experimental design for time-lapse traveltime tomography. *Society of Exploration Geophysicists*, 2008.

[2] J. B. Ajo-Franklin, J. Urban and J. M. Harris. Temporal integration of seismic traveltime tomography. *Society of Exploration Geophysicists Annual Meeting, Expanded Abstracts*, 25:2468, 2006.

[3] R. Arts, R. Elsayed, L. Van Der Meer, O. Eiken, O. Ostmo, A. Chadwick, G. Kirby and B. Zinszner. Estimation of the mass of injected CO_2 at Sleipner using time-lapse seismic data. *Paper H-16, EAGE 64th Annual Conference*, 2002.

[4] Z. Bai, M. Fahey and G. H. Golub. Some large-scale matrix computation problems. *J. Computational & Applied Math*, 74:71-89, 1996.

[5] A. Bardow. Optimal experimental design of ill-posed problems: The METER approach. Computers and Chemical Engineering, 32:115-124, 2008.

[6] N. Barth and C. Wunsch. Oceanographic experiment design by simulated annealing. *J. Phys. Oceanography*, 20:12491263, 1990.

[7] I. Bauer, H. G. Bock, S. Körkel and J. P. Schlöder. Numerical methods for optimum experimental design in DAE systems. *J. Comput. Appl. Math.*, 120:1-15, 2000.

[8] E. Van Den Berg and M. P. Friedlander. Probing the Pareto frontier for basis pursuit solutions. *SIAM J. Sci. Comput.*, 31:890-912, 2008.

[9] J. M. Bernardo. Expected information as expected utility. *Ann. Statist.*, 7:686-690, 1979.

[10] M. Bertero and P. Boccacci. Introduction to inverse problems in imaging. *Institute of Physics Publishing*, Bristol, 1998.

[11] S. Boyd and L. Vandenberghe. Convex optimization. *Cambridge University press*, 2004.

[12] R. J. Brooks. On the choice of an experiment for prediction in linear regression. *Biometrika*, 61:303-311, 1974.

[13] K. Chaloner. Optimal Bayesian experimental designs for linear models. *Ann. Statist.*, 12:283-300, 1984.

[14] K. Chaloner and I. Verdinelli. Bayesian experimental design: A review. *Statist. Sci.*, 10:237-304, 1995.

[15] J. Chung, E. Haber, and J. Nagy. Numerical methods for coupled super-resolution. *Inverse Problems*, 22:1261-1272, 2006.

[16] A. Curtis. Optimal experimental design: Cross-borehole tomographic examples. Geophys. J. Int., 136:637-650, 1999.

[17] T. M. Daley, R. D. Solbau, J. B. Ajo-Franklin and S. M. Benson. Continuous active-source seismic monitoring of CO_2 injection in a brine aquifer. *Society of Exploration Geophysicists*, 2007.

[18] M. H. DeGroot. Concepts of information based on utility. *In recent developments in the foundations of utility and risk theory*, L. Daboni, A. Montesano, and M. Lines eds. 265-275. Reidel, Dordrecht, Holland, 1986.

[19] D. L. Donoho and M. Elad. Optimally sparse representation in general (nonorthogonal) dictionaries via l^1 minimization. *Proc. Natl. Acad. Sci. USA*, 2003.

[20] G. Duncan and M. H. DeGroot. A mean squared error approach to optimal design theory. *Proceedings of the 1976 conference on information: science and systems*, 217-221, The Johns Hopkins University.

[21] M. Elad and A. M. Bruckstein. A generalized uncertainty principle and sparse representation in pairs of bases. *IEEE Trans. Inform. Theory*, 2002.

[22] M. Elad and A. Feuer. Restoration of a single super-resolution image from several blurred, noisy, and undersampled measured images. *IEEE Trans. Image Proc.*, 6(12):1646-1658, 1997.

[23] H. W. Engl, M. Hanke and A. Neubauer. Regularization of inverse problems. *volume 375 of Mathematics and its Applications*, Kluwer Academic Publishers Group, Dordrecht, 1996.

[24] V.V. Fedorov and P. Hackl. Model-Oriented Design of Experiments. *Lecture Notes in Statistics*, Springer, 1997.

[25] M. Figueiredo, R. Nowak, and S. J. Wright. Gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems. *IEEE J. Selected Topics in Signal Process*, 586-597, 2007.

[26] D. E. Finkel. Global optimization with the direct algorithm. *North Carolina State University*, 2005.

[27] J. Friedman, T. Hastie and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9:432-441, 2008.

[28] G. H. Golub and C. F. Van Loan. Matrix Computations. *The John Hopkins University Press, Second Edition*, 1996.

[29] G. H. Golub and U. Van Matt. Tikhonov regularization for large scale problems. *Technical Report SCCM 4-79*, 1997.

[30] Anne Greenbaum. Iterative methods for solving linear systems. *Frontiers in Applied Mathematics*, 1997.

[31] R. Gritto, T. M. Daley and L. R. Myer. Joint cross-well and single-well seismic studies at Lost Hills, California. *Geophys Prospect*, 52:323339, 2004.

[32] E. Haber. Numerical strategies for the solution of inverse problems. *Ph.D. Thesis*, The University of British Columbia, 1997.

[33] E. Haber. A multilevel, level-set method for optimizing eigenvalues in shape design problems. *J. Comput. Phys.*, 198:518-534, 2004.

[34] E. Haber, U. Ascher, and D. Oldenburg. On optimization techniques for solving nonlinear inverse problems. *Inverse problems*, 16:1263-1280, 2000.

[35] E. Haber, L. Horesh, and L. Tenorio. Numerical methods for experimental design of large-scale linear ill-posed inverse problems. *Inverse Problems*, 24, 2008.

[36] E. Haber, Z. Magnant, C. Lucero and L. Tenorio. Numerical methods for A-optimal designs with a sparsity constraint for ill-posed inverse problems. *Computational Optimization and Applications*, 2011.

[37] E. Haber and L. Tenorio. Learning regularization functionals-a supervised training approach. *Inverse Problems*, 19:611-626, 2003.

[38] M. Hanke and P. C. Hansen. Regularization methods for large-scale problems. *Surveys Math. Indust.*, 3:253-315, 1993.

[39] P. C. Hansen. Rank-deficient and discrete ill-posed problems. *Society for Industrial and Applied Mathematics*, Philadelphia, PA, USA, 1998.

[40] P. C. Hansen, J. G. Nagy and D. P. O'Leary. Deblurring images matrices, spectra, and filtering. *SIAM*, 2006.

[41] T. M. Hansenand K. Mosegaard. VISIM: Sequential simulation for linear inverse problems. *Comput. Geosci.*, 34:53-76, 2008.

[42] J. M. Harris, R. C. Nolen-Hoeksema, R. T. Langan, M. Van Schaack, S. K. Lazaratos and J. W. Rector. High-resolution crosswell imaging of a west Texas carbonate reservoir: Part 1Project summary and interpretation. *Geophysics*, 60:667681, 1995.

[43] I. Hnětynková and Z. Strakoš. Lanczos tridiagonalization and core problems. *Linear Algebra and its Applications*, 421(2-3):243-251, 2007.

[44] G. M. Hoversten, R. Gritto, J. Washbourne and T. M. Daley. Pressure and fluid saturation prediction in a multicomponent reservoir, using combined seismic and electromagnetic imaging. *Geophysics*, 68:15801591, 2003.

[45] M.F. Hutchinson. A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines. *J. Commun. Statist. Simul.*, 19, 1990.

[46] W. Huyer and A. Neumaier. Global optimization by multilevel coordinate search. *Journal of Global Optimization*, 14:331355, 1999.

[47] J. M. Jensen and B. H. Jacobsen. Sensitivity Kernels for Time-Distance Inversion. 1999.

[48] J. M. Jensen, B. H. Jacobsen and J. Christensen-Dalsgaard. Sensitivity kernels for time-distance inversion. *Solar Physics*, 192:231239, 2000.

[49] D. R. Jones, C. D. Perttunen and B. E. Stuckman. Lipschitzian optimization without the Lipschitz constant. *J. Optim. Theory Appl.*, 79:157-181, 1993.

[50] H. Lauter and H. Liero. Ill-Posed Inverse Problems and Their Optimal Regularization. 1997.

[51] S. K. Lazaratos and B. P. Marion. Crosswell seismic imaging of reservoir changes caused by CO2 injection. *Lead Edge*, Lead Edge 16:13001306, 1997.

[52] A. S. Lewis and Von Neumann. The Mathematics Of Eigenvalue Optimization. 2003.

[53] C. Meyer. Matrix analysis and applied linear algebra. *Society for Industrial and Applied Mathematics (SIAM)*, 2000.

[54] V. A. Morozov. Regularization methods for ill-posed problems. *CRC Press*, Boca Raton, FL, 1993.

[55] J. Nagy and K. Palmer. Steepest descent, CG and iterative regularization of ill-posed problems. *BIT*, 43:1003-1017, 2003.

[56] J. C. Nash and S. Shlien. Simple algorithms for the partial singular value decomposition. *The Computer Journal*, 30:268-275, 1987.

[57] J. Nelder and R. Mead. Asimplex method for function minimization. *The-Computer Journal*, 7:308313, 1965.

[58] D. W. Oldenburg. One-dimensional inversion of natural source magnetotelluric measurements. *Geophysics*, 44:1218-1244, 1979.

[59] D. P. O'Leary and J. A. Simmons. A bidiagonalization-regularization procedure for large-scale regularization of ill-posed problems. *SIAM J. Sci. Stat. Comput.*, 2:474-489, 1981.

[60] J. Orchard (2005, April 27). His Brain. *Retrieved March 26, 2010, from The University of Waterloo website: http://www.cs.uwaterloo.ca/ jorchard/mri/*

[61] R. J. Owen. The Optimum design of a two-factor experiment using prior information. *Ann. Statist.*, 41:1917-34, 1970.

[62] K. B. Petersen and M. S. Pedersen. *http://matrixcookbook.com.*

[63] D. L. Phillips. A technique for the numerical solution of certain integral equations of the first kind. *J. Assoc. Comput. Mach.*, 9:84-97, 1962.

[64] F. Pukelsheim. Optimal Design of Experiments. *John Wiley & Sons*, 1993.

[65] N. Rabinowitz and D. M. Steinberg. Optimal Configuration of a seismographic network: a statistical approach. *Bull. seism. Soc. Am.*, 80(1):187-196, 1990.

[66] J. Spetzler. Time-lapse seismic crosswell monitoring of steam injection in tar sand. *Society of Exploration Geophysicists Annual Meeting, Expanded Abstracts*, 25:3120, 2006.

[67] M. Stone. Application of a measure of information to the design and comparison of regression experiment. *Ann. Stat.*, 30:55-70, 1959.

[68] M. Stone. Discussion of Kiefer. *J. Roy. Statist. Soc. Ser. B*, 21:313-315, 1959.

[69] M. Sugiyama and H. Ogawa. Subspace information criterion for model selection. *Neural Comput.*, 13:1863-1889, 2001.

[70] M. Sugiyama and H. Ogawa. Optimal design of regularization term and regularization parameter by subspace information criterion. *Neural Netw.*, 3:349-361, 2002.

[71] M. Sugiyama and H. Ogawa. Theoretical and experimental evaluation of the subspace information criterion. *Mach. Learn.*, 48:25-50, 2002.

[72] L. Tenorio, F. Andersson, M. de Hoop and P. Ma. Data analysis tools for uncertainty quantification of inverse problems. Submitted.

[73] A. N. Tikhonov and V. Y. Arsenin. Solutions of ill-posed problems. *V. H. Winston & Sons, Washington, D.C.: John Wiley & Sons*, New York, 1977.

[74] L. N. Trefethen and D. Bau. Numerical Linear Algebra. *SIAM: Society for Industrial and Applied Mathematics*, 1997.

[75] C. R. Vogel. Computational methods for inverse problems. *volume 23 of Frontiers in Applied Mathematics*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2002.

[76] G. Wahba. Practical approximate solutions to linear operator equations when the data are noisy. *SIAM J. Numer. Anal.*, 14:651-667, 1977.

[77] K. P. Whittall and D. W. Oldenburg. Inversion of Magnetotelluric Data for a One Dimensional Conductivity, volume 5. *SEG monograph*, 1992.

[78] C. A. Zelt and P. J. Barton. Three-dimensional seismic refraction tomography: A comparison of two methods applied to data from the Faeroe Basin. *J. Geophys. Res.*, 103:7187-7210, 1998.