©Copyright 2014
Ryan Kappedal

Gravimetric Anomaly Detection using Compressed Sensing

Ryan Kappedal

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2014

Reading Committee:

Marina Meilă, Chair

Maryam Fazel

Vladimir Minin

Program Authorized to Offer Degree:
Department of Statistics, University of Washington

University of Washington

Abstract

Gravimetric Anomaly Detection using Compressed Sensing

Ryan Kappedal

Chair of the Supervisory Committee:
Associate Professor Marina Meilă
Department of Statistics

We address the problem of identifying underground anomalies (e.g. holes) based on gravity measurements. This is a theoretically well-studied yet difficult problem. In all except a few special cases, the inverse problem has multiple solutions, and additional constraints are needed to regularize it. Our approach makes general assumptions about the shape of the anomaly that can also be seen as sparsity assumptions. We can then adapt recently developed sparse reconstruction algorithms to address this problem. The results are extremely promising, even though the theoretical assumptions underlying sparse recovery do not hold for gravity problems of this kind. We examine several types of sparse bases in the context of this gravity inverse problem and compare and contrast their relative merits.

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

First, I would like to thank my wife, Elisa and sons, Rigel and Caelum for their unending support. Also, I would like to thank my advisor, Marina Marina Meilă, my research colleague, Hoyt Koepke, and my good friend, Colin Sowder. Lastly, special thanks to Jeff and Julie Vail for seeing my through the final push.

Chapter 1

INTRODUCTION

In this section I describe the gravitational inverse problem and give a general overview of the structure of the thesis.

1.1 What is Gravimetrics?

Graviometry is the measurement of the strength of a gravitational field. An instrument used to measure gravity is known as a gravitometer. Scales may be regarded as simple gravitometers. In one common form, a spring is used to counteract the force of gravity pulling on an object. The change in length of the spring may be calibrated to the force required to balance the gravitational pull. The resulting measurement may be made in units of force (such as the Newton).

More sophisticated gravitometers are used when precise measurements are needed. By measuring the Earth's gravitational field, we can find density variations in the ground underneath the sensors. This particular application of graviometry is aimed at discovering cavities, holes, and tunnels by use of highly precise measurements taken at the surface.

1.2 Gravitational Inverse Problems

In conventional modelling, data are collected, and a model is fit based on that data, which tells us the governing equations and their parameters. Inverse problems are for situations when we already know the governing equations of the system, and we can observe the solution to those equations [Isakov, 1990].

In our case, the physical system is Earth's gravitational field, the governing equation is Newton's Law of universal gravitation, the property we are attempting to solve for is mass density, and our observations are the gravitational field measurements.

Our mathematical model begins with Newton's Law of universal gravitation:

$$F = G\frac{m_1 m_2}{r^2}. \tag{1.1}$$

F is the force between the masses, G is the gravitational constant, m_1 is the first mass, m_2 is the second mass, and r is the distance between the centers of the masses. We know the value of m_1, as it is the test mass that we use to conduct our measurements, so we will specify a gravitational potential ϕ where

$$\phi = -\frac{F}{m_1} = -G\frac{m_2}{r^2}. \tag{1.2}$$

But the particular property that we are interested in is the density of the object that has mass m_2, called ρ. Now, consider two points in three dimensional space, the sensor location, $\mathbf{s} = [s_x, s_y, s_z]$ and the object location $\mathbf{o} = [o_x, o_y, o_z]$. Also, let us consider $\mathbf{o}$ to be within the domain $\mathbb{D}$ of the sensor's detection threshold in field strength. If the sensor is above the surface of the Earth, the air molecules (the Moon, the Sun, etc.) are outside of $\mathbb{D}$, but the ground below the sensor is within $\mathbb{D}$, and the sensor at location $\mathbf{s}$ is outside of $\mathbb{D}$. We are particularly interested in finding features of the mass density at different distances underground in a region directly below the sensor.

1.3 Description of Motivating Examples

The genesis of this problem is an attempt to find underground man-made structures, such as tunnels. Since from the perspective of gravimetrics, a tunnel represents a void, it should be represented by a large drop in the mass density. Some of the applications already attempted in gravimetric sensing were: detecting lava tunnels in Hawaii [Meglich et al., 2005], abandoned coal mines in Ohio [Munk and Sheets, 1997], mapping caves, and buried riverbeds and other geological features [Krasovec and Morgan, 1999]. There are also many potential military and security applications to being able to locate underground tunnels.

1.4 Structure of the Dissertation

In Chapter 2, we will review the general properties of Compressed Sensing, the main technique we use in this research on the gravimeteric inverse problem. In Chapter 3, we will

develop the gravity measurement model into a compressed sensing problem formulation and obtain results for the probabilities of the compressed sensing assumptions holding. In Chapter 4, we introduce a set of representational systems called dictionaries that we will examine in the gravity measurement compressed sensing problem. In Chapter 5, we will formulate our optimization problems and conduct extensive experiments on those dictionaries to examine the best detection characteristics and false positive rejection for when we get data that does not contain an anomaly. Finally, Chapter 6 will summarize our conclusions and suggest future research directions.

Chapter 2

COMPRESSED SENSING

In this section we talk about the background methods for compressed sensing and how it can apply to the gravimetric inverse problem.

2.1 Compressed Sensing Overview

Our research is based on a more general class of problems called Sparse Coding. Sparse coding seeks to represent a complex signal as a summation of fewer elements. The elements are columns in a matrix, and the summation of these columns is accomplished by a vector with a small amount of non-zero values.

The general problem of finding a representation of a signal with the smallest number of columns from an arbitrary basis has been shown to be NP-hard [Mallat and Zhang, 1993]. This has led to a the development of many sub-optimal schemes. These include algorithms which iteratively build up the signal approximation one coefficient at a time, such as Matching Pursuit [Mallat and Zhang, 1993], Orthogonal Matching Pursuit [Pati et al., 1993], and those that process all the coefficients simultaneously, such as Basis Pursuit [Chen and Donoho, 1994], Basis Pursuit De-Noising [Granai and Vandergheynst, 2004] and the Focal Underdetermined System Solver family of algorithms [Adler et al., 1996] [Rao and Kreutz-Delgado, 1999] [Murray and Kreutz-Delgado, 2001]. Our research is primarily concerned with using Compressed Sensing methods.

Compressed Sensing is a technique for finding sparse solutions to under-determined linear systems using L_1 minimization. We shall define "sparse" as using a small number of non-zero coefficients. Compressed Sensing gives us a process of acquiring and reconstructing a signal that is supposed to be sparse or compressible, provided some conditions are met [Tibshirani, 1996], [Candes and Tao, 2005], [Donoho, 2006].

Compressed sensing algorithms such as Basis Pursuit can not only recover sparse data

β exactly from $\mathbf{X}\beta = \mathbf{y}$, where $\mathbf{X} \in \mathbb{R}^{m \times n}$, and $m << n$ but can also recover compressible data (data which is approximately sparse [Tibshirani, 1996]), from $\mathbf{X}\beta + \epsilon = \mathbf{y}$ by a slight modication of the algorithm. These algorithms are robust with regards to noise. If we only have some noisy measurements, $\mathbf{y} = \mathbf{X}\beta + \epsilon$ of the signal β, where ϵ is a noise vector (e.g. I.I.D. Gaussian) with mean zero and standard deviation σ, then we can recover an approximation [Tibshirani, 1996], [Candes and Tao, 2005].

The purpose of Compressed Sensing in this particular application of graviometry is to discover cavities, holes, and tunnels, which we assume are rare, and therefore it is preferable to represent them in a sparse fashion to give us a chance to detect them, given many different types of shapes could have generated a gravimetric signal.

The original sparse recovery problem formulation is

$$\begin{gathered} \min_{\beta \in \mathbb{R}^n} \|\beta\|_0 \\ \text{st} \\ \mathbf{X}\beta = \mathbf{y} \end{gathered} \tag{2.1}$$

however; this problem is generally NP-hard [Natarajan, 1995], so we examine two convex Compressed Sensing formulations in order to make this problem tractable. Using these convex relaxations, under certain conditions (to be outlined later in this section)

The bounded variance form of the compressed sensing Linear Program (LP) is as follows [Candes and Tao, 2005]:

$$\begin{gathered} \min_{\beta \in \mathbb{R}^n} \|\beta\|_1 \\ \text{st} \\ \|\mathbf{X}\beta - \mathbf{y}\|_\infty \leq c\sigma, \end{gathered} \tag{2.2}$$

where σ is the variance of the error in the detected signal, and c is some constant. The constant in the optimization program is used to account for additional recovery requirements [Candes and Tao, 2005]. If the true parameter vector β is relatively sparse, the solution to this LP also solves perfectly for that parameter. We will discuss the exact sparsity requirements required for recovery later in this section as well as the types of $\mathbf{X}$ that ensure recovery. The bounded variance Linear Program may be easier to solve in terms of

computational complexity, as larger allocations of memory are required in many Quadratic Program solvers.

The regularized version of the bounded variance problem is

$$\min_{\beta} \|\mathbf{X}\beta - \mathbf{y}\|_{\infty} + \alpha \|\beta\|_1 ,$$

where α corresponds to the Lagrange multiplier in the dual problem, and also allows us to create an unbounded problem that may give us flexibility in solution algorithms.

The compressed sensing Quadratic Program (QP) version is as follows [Tibshirani, 1996]:

$$\begin{aligned} &\min_{\beta \in \mathbb{R}^n} \|\beta\|_1 , \\ &\text{st} \\ &\|\mathbf{X}\beta - \mathbf{y}\|_2 \leq c\sigma. \end{aligned} \tag{2.3}$$

Equally with this regularized version, also called Basis Pursuit De-Noising [Granai and Vandergheynst, 2004]:

$$\min_{\beta} \|\mathbf{X}\beta - \mathbf{y}\|_2 + \alpha \|\beta\|_1 .$$

The 2-norm has a better theoretical support in giving us recovery guarantees

Our research employs readily available software to solve the Compressed Sensing LP and QP [Becker et al., 2011].

2.1.1 Why does L_1 minimization work for sparse recovery?

Consider three balls below, the $L_{1/2}$ ball, the L_1 ball, and the L_2 ball.

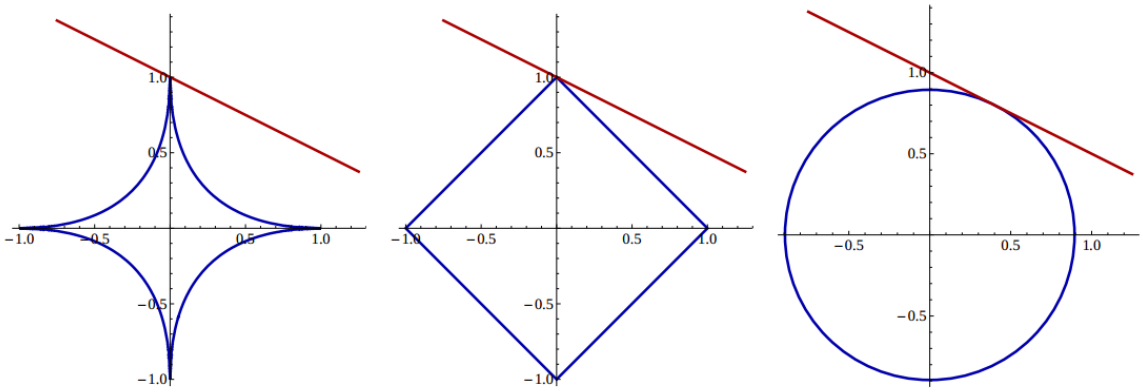

Figure 2.1: $L_{1/2}$, L_1, L_2 balls, the red line represents a linear cost function

As we can see, in the 2-dimension case, the L_1 solution recovers the same solution as the $L_{1/2}$, and would continue to do so for the L_0 solution, except in the pathological case where the null-space of $\mathbf{X}$ is perpendicular to one of the L_1 ball edges. This observation relies upon the true sparsity of β, and the how dissimilar the columns of $\mathbf{X}$ are (also called the incoherence of $\mathbf{X}$). Please note, that the multi-dimensional example does not necessarily follow directly, but some of the general geometry for L_0 and L_1 norms still holds [Candes et al., 2008].

The unit ball of the L_1 norm is a convex polytope (a geometric object with flat sides that exists in a general number of dimensions), and so there exists a class of such polytopes in which the level sets of our objective function will intersect the polytope if the objective function is convex. In the case of a orthogonally-constructed polytope, it is also easy to see that all of the points lie on the convex hull of that polytope. In this case, the L_1 minimization solves the L_0 minimization problem every time. However, when we don't have an orthogonal system, this may not always hold, but with a set of n points in m dimensions, as long as these n points all lie on their own convex hull , the level sets of a convex objective function makes L_1 minimization work for the L_0 problem [Donoho and Tanner, 2009].

Theorem 2.1.1 *For an* $\mathbf{X}$ *matrix that is* $m \times n$*, the problem* $[\ \min \|\beta\|_1 , \|\mathbf{X}\beta - \mathbf{y}\| \leq c\sigma]$ *is*

equivalent to $[\ \min \|\beta\|_0 , \|\mathbf{X}\beta - \mathbf{y}\| \leq c\sigma]$ *with probability* $f_s(\mathbf{X}T)/f_s(T)$ *where* $f_s(.)$ *is the number of s-faces of a polytope, and* T *is the* $(n-1)$ *simplex, and* $\mathbf{X}T$ *is the projection of that simplex by the matrix* $\mathbf{X}$ *[Donoho and Tanner, 2009].*

The problem with this condition is that face-counting of a high-dimensional polytope is typically a hard problem. Currently, the only methods of determining whether or not the high-dimensional geometry of a given problem meets the requirements of Theorem 2.1.2. for a given probability are via direct simulation [Donoho and Tanner, 2009].

Characterizing the class of matrices with a well-behaved phase transition in the high-dimensional geometry such that the probability of recovery is known is an open problem.

2.2 Sparsity and Incoherence

Whether or not the requirements of Theorem 2.1.2. hold depends greatly upon the sparsity of the true β and the incoherence of $\mathbf{X}$. So one needs to find the right representation, $\mathbf{X}$, in which the solution is sparse. We must also have the lowest possible correlation between the representation basis and the sensing basis; (this is called incoherence). Practically this is achieved through sampling in a random fashion using a Gaussian design matrix. Sometimes this is not possible, so the combination of two bases, like noiselets and Fourier, are required to give us both incoherence and sparsity [Candes and Wakin, 2008].

Generally, the coherence of a given matrix can be measured with two basic measures: maximum coherence $\mu(\mathbf{X})$, and average coherence $\nu(\mathbf{X})$ [baj]. They are defined as the following, assuming normalized columns of $\mathbf{X}$ (note that the operator $\langle \mathrm{x}_i, \mathrm{x}_j \rangle$ is the inner product of x_i and x_j):

Definition 2.2.1 *For a matrix* $\mathbf{X}$ *with unit norm columns that is* $m \times n$*, the maximum coherence is*

$$\mu(\mathbf{X}) \equiv \max_{i,j:i\neq j} |\langle \mathrm{x}_i, \mathrm{x}_j \rangle| .$$

Definition 2.2.2 *For a matrix* $\mathbf{X}$ *with unit norm columns that is* $m \times n$*, the average coherence is*

$$\nu(\mathbf{X}) \equiv \frac{1}{n-1} \max_i \left| \sum_{j:j\neq i} \langle \mathrm{x}_i, \mathrm{x}_j \rangle \right| .$$

As one can see, these values would both be zero when the $\mathbf{X}$ is an orthonormal basis. The problem is that most deterministic sensing matrices (like the gravimetric operator) are not incoherent except under very specific asymptotic conditions. From these measures we can derive a probability of adherence to the restricted isometry property below under certain conditions [Bajwa et al., 2012].

2.2.1 Restricted Isometry Property (RIP)

The restricted isometery property (RIP) is a sufficient property for robust sparse recovery using L_1 minimization, but it is not necessary. Indeed many matrices have been shown to not have the restricted isometery property but are still observed to meet the requirements of Theroem 2.1.2. with high probability [Chen et al., 2013].

Definition 2.2.3 (Restricted Isometry Property) *The (δ_s) restricted isometry property (RIP) states that for a given sparsity, s, and a matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ the following relationship must hold:*

$$(1 - \delta_s) \|\beta\|_2^2 \leq \|\mathbf{X}\beta\|_2^2 \leq (1 + \delta_s) \|\beta\|_2^2$$

where $\delta_s < 1$ is the smallest value for which the inequality is true (called the isometry constant) for every vector β with at most s nonzero entries.

The closer δ_s is to zero, the better. This property essentially requires that every set of columns with a cardinality less that s behave like an orthonormal system. This guarantees that any sparse system with cardinality less that or equal to s will be recovered by the linear program [Candes and Tao, 2005]. A detailed proof of a weaker condition is in Section 3.6.

The challenge presented is that we can never truly examine all of the columns of s or less than s for any reasonably sized sensing matrix. The combinatorial nature of examining all of n choose s columns is not practical in s. Hence, certifying RIP is an NP-hard problem [Bandeira et al., 2012].

When we have RIP, we can be assured that our solution is unique, as it implies that all pairwise distance between s-sparse signals are preserved in the measurement dictionary space. Accordingly, the following relationship holds for any two reconstructions:

$$(1-\delta_{2s})\left\|\beta_1-\beta_2\right\|_2^2 \leq \left\|\mathbf{X}\beta_1-\mathbf{X}\beta_2\right\|_2^2 \leq (1+\delta_{2s})\left\|\beta_1-\beta_2\right\|_2^2 \tag{2.4}$$

It is trivial to verify RIP when the columns of a square sensing matrix (in this case $\mathbf{X}$) are orthogonal. However, as the dimension of n gets larger and larger, the near orthogonality of submatrices of $\mathbf{X}$ small enough to be almost guaranteed in the sparse representation, provided the worst-case coherence, $\mu(\mathbf{X})$, average coherence $\nu(\mathbf{X})$ are reasonably low [baj].

2.2.2 Weak Restricted Isometry Property

Theorem 2.2.4 (Weak Restricted Isometry Property) *An* $\mathbf{X}$ *matrix with unit-norm columns that is* $m \times n$ *has the* (δ_s, p) *weak restricted isometry property (RIP) if for a given sparsity,* s*, the following relationship holds:*

$$(1-\delta_s)\left\|\beta\right\|_2^2 \leq \left\|\mathbf{X}\beta\right\|_2^2 \leq (1+\delta_s)\left\|\beta\right\|_2^2$$

with probability $1-p$ *over all possible solutions where* β *is any* s*-sparse vector.*

2.2.3 Strong Coherence Property

It is also possible to estimate the probability of weak RIP, using the Strong Coherence Property [Bajwa et al., 2012].

Definition 2.2.5 (Strong Coherence Property) *If* X *is a matrix with unit norm columns that is* $m \times n$*, and meets the following conditions, it has the Strong Coherence Property:*

$$\mu(\mathbf{X}) \leq \tfrac{1}{164\log n} \quad \text{and} \quad \nu(\mathbf{X}) \leq \tfrac{\mu(\mathbf{X})}{\sqrt{m}}.$$

Theorem 2.2.6 *An* $\mathbf{X}$ *unit-norm frame that is* $m \times n$ *that has the Strong Coherence Property and* $m \geq 128$*, and* $2s\log n \leq \min\left\{\frac{\delta_s}{100\mu(\mathbf{X})^2}, m\right\}$ *has weak RIP with probability* $p = \frac{4s}{n^2}$*.*

We will show that these conditions rarely exist within our matrices, and accordingly we will introduce new methods to compute probabilities of our solution recovery guarantee.

2.2.4 D-Restricted Isometry Property

Should we need to change to a different basis that is more sparse, the D-RIP ensures that we can preserve RIP [Candes et al., 2011].

Definition 2.2.7 (D-Restricted Isometry Property) *The (δ_s) D-restricted isometry property (RIP) states that for a given sparsity, s, and a matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ that has RIP, and a matrix D that is an orthonormal frame, and Ω_s is the union of all s-sized (or smaller) selections of columns of D, the following relationship must hold:*

$$(1 - \delta_s) \|\beta\|_2^2 \leq \|\mathbf{X}\beta\|_2^2 \leq (1 + \delta_s) \|\beta\|_2^2$$

where $\delta_s < 1$ is the smallest value for which the inequality is still true (called the isometry constant), for all $\beta \in \Omega_s$.

2.2.5 One Step Thresholding and Statistical Orthogonality

Another approach to a probabilistic recovery guarantee is Statistical Orthogonality and the One Step Thresholding algorithm [Bajwa et al., 2010].

Algorithm 1 The One Step Thresholding (OST) Algorithm for Model Selection

1: Input: An $m \times n$ matrix $\mathbf{X}$, a vector $\mathbf{y} \in \mathbb{R}^m$, and a thresholding parameter $\lambda > 0$

2: Output: Compute $\beta \equiv \mathbf{X}^T\mathbf{y}$ and return an estimate of the model $\hat{S} \equiv j \in 1, ...n : |\beta_i| > \lambda$.

Now statistical orthogonality can give us some measure of a probability of success for this method.

Definition 2.2.8 (Statistical Orthogonality (StOrth)) *Let Π be a uniformly sampled s subset of $\{1...n\}$, and let $\Pi^c = \{1...n\} - \Pi$. Then, given $\epsilon, p \in [0, 1)$, $\mathbf{X}$ is said to satisfy the (s, ϵ, p)-statistical orthogonality condition if the following inequalities hold, if the columns of $\mathbf{X}$ are normalized:*

$$\left\| \left(\mathbf{X}_\Pi^T \mathbf{X}_\Pi - I \right) \mathbf{z} \right\|_\infty \leq \epsilon \|\mathbf{z}\|_2 ,$$

$$\left\|\mathbf{X}_{\Pi^c}^T \mathbf{X}_{\Pi} \mathbf{z}\right\|_{\infty} \leq \epsilon \left\|\mathbf{z}\right\|_2$$

for every fixed $\mathbf{z} \in \mathbb{R}^s$ *with a probability exceeding* $1 - p$.

This allows us to have a probability of recovery using One Step Thresholding that is very close to $1 - p$.

Theorem 2.2.9 *Let* $\Pi \equiv supp(\beta)$ *be a uniformly random s-subset of 1,...,n. Further, suppose that the matrix* $\mathbf{X}$ *satisfies the* (s, ϵ, p)*-statistical orthogonality condition, and choose the threshold as* $\lambda > 2 \max \epsilon, 2\sqrt{\sigma^2 \log n}$. *Then, under the assumption that* $\beta_{\min} > 2\lambda$, *the OST algorithm solution satisfies*

$$Pr\left(\hat{S} \neq \Pi\right) \leq p + 2\left(\sqrt{2\pi \log n} \cdot n\right)^{-1}.$$

Armed with all these properties, we shall move forward and examine the Gravimetric inverse problem in this context.

Chapter 3

PROBLEM STATEMENT

In this section I explore and examine the general problem structure, assumptions, construction of a measurement matrix, and a dictionary to turn the gravitational inverse problem into a compressed sensing program

3.1 Gradiometry Measurement Model

Since the gravitational field is measured at sensor location, $\mathbf{s}$ and we are interested in the density at object location, $\mathbf{o}$, we have a set of field measurements which are a function of $\mathbf{s}$ [Meila et al., 2008]. First, we will develop the gravimetric inverse problem into a linear system. From Chapter 1, Equation 1.2, by substitution with volume, V, and integrating over all possible point masses in the domain we obtain. The development is as follows:

$$dm_2 = \rho(\mathbf{o})dV(\mathbf{o}), \tag{3.1}$$

$$r = \|\mathbf{s}-\mathbf{o}\|_2, \tag{3.2}$$

$$\nabla\phi(\mathbf{s}) = -G\int \frac{\rho(\mathbf{o})}{\|\mathbf{s}-\mathbf{o}\|_2^2}\frac{\mathbf{s}-\mathbf{o}}{\|\mathbf{s}-\mathbf{o}\|_2}dV(\mathbf{o}). \tag{3.3}$$

3.1.1 Problem in Integral Form

The way a gravimetric reading is actually taken is based on the second derivative of $\phi(\mathbf{s})$. Let us declare the following relationships:

$$T_{xx}(\mathbf{s}) \equiv -\frac{\partial^2}{\partial o_x^2}\phi(\mathbf{s}) = G\int \frac{\rho(\mathbf{o})[3(s_x-o_x)^2 - \|\mathbf{s}-\mathbf{o}\|_2^2]}{\|\mathbf{s}-\mathbf{o}\|_2^5}dV(\mathbf{o}), \tag{3.4}$$

$$T_{yy}(\mathbf{s}) \equiv -\frac{\partial^2}{\partial o_y^2}\phi(\mathbf{s}) = G\int \frac{\rho(\mathbf{o})[3(s_y-o_y)^2 - \|\mathbf{s}-\mathbf{o}\|_2^2]}{\|\mathbf{s}-\mathbf{o}\|_2^5}dV(\mathbf{o}), \tag{3.5}$$

which are combined into:

$$T_{xx}(\mathbf{s}) - T_{yy}(\mathbf{s}) = G\int \frac{\rho(\mathbf{o})3[(s_x-o_x)^2 - (s_y-o_y)^2]}{\|\mathbf{s}-\mathbf{o}\|_2^5}dV(\mathbf{o}). \tag{3.6}$$

and then:

$$T_{xy}(\mathbf{s}) \equiv -\frac{\partial^2}{\partial o_x \partial o_y}\phi(\mathbf{s}) = G\int \frac{\rho(\mathbf{o})[3(s_x - o_x)(s_y - o_y)]}{\|\mathbf{s}-\mathbf{o}\|_2^5} dV(\mathbf{o}). \tag{3.7}$$

We receive measurements in this form:

$$[T_{xx} - T_{yy}, 2T_{xy}], \tag{3.8}$$

here $T_{xx} - T_{yy}$ is referred to as the *inline* measurement, and $2T_{xy}$ is referred to as the *crossline* measurement. The units of these measurements are called *Eotvos*.

3.1.2 Assumptions on types of noise as a function of T

Assumption 3.1.1 *The features we are looking for are thought to occur very rarely underground, so whatever basis we transform our problem into, a cavity must have a sparse representation.*

Using $A \in \mathbb{R}^{m\times p}$ to represent a gravimetric linear operator, and $\mathbf{b} \in \mathbb{R}^p$ to represent some discretized set of point masses in the sensing domain, we make the following assumption about the underground density.

Assumption 3.1.2 *If* $\mathbf{b}$ *is the set of masses at locations* $\mathbf{o}$ *in a given domain* $\mathbb{D}$*, and* A *is the linear operation that maps these point masses to set of measurement sites, then* $\mathbf{m}_{RAW}$ *is what we would expect to see in our measurements, assuming no error, and* $\mathbf{m}$ *is the measurement with an I.I.D. Gaussian noise with variance* σ^2*.*

Based on the real-world data, gradient measurements are taken on tracks, meaning the sensor is on a moving vehicle that takes several parallel tracks, giving us a total of 192 measurements. This is shown in Figure 3.1.

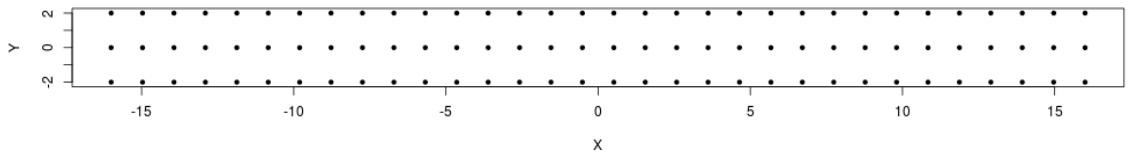

Figure 3.1: Surface Measurement Locations

The measurement model is

$$A\mathbf{b} = \mathbf{m}_{RAW}, \tag{3.9}$$

$$\mathbf{m} = \mathbf{m}_{RAW} + \epsilon. \tag{3.10}$$

Then, if we assume the variance of B is Σ_B, then it follows that

$$\Sigma_{\mathbf{m}_{RAW}} = A^T \Sigma_{\mathbf{b}} A, \tag{3.11}$$

$$\Sigma_{\mathbf{m}} = \Sigma_{\mathbf{m}_{RAW}} + diag(\sigma^2), \tag{3.12}$$

$$\Sigma_{\mathbf{m}} = A^T \Sigma_{\mathbf{b}} A + diag(\sigma^2). \tag{3.13}$$

We shall make several assumptions about the underlying processes of the variance in our measurements, and the way in which measurements are taken. A Gaussian process is a stochastic process whose realizations consist of random values associated with every point in a range of times (or of space) such that each random variable has a normal distribution. Moreover, every finite collection of those random variables has a multivariate normal distribution. With the exception of the man-made anomalies that we are attempting to detect, we will assume that underground density field comes from a Gaussian process. We believe a 3-D Gaussian random field is a good general model for this process (Ornstein-Uhlenbeck process) [Uhlenbeck and Ornstein, 1930] from personal conversations with faculty in the Earth and Space Sciences department [Bro], and it is a good general model to use. Unfortunately, actual data on the process (with variance $\Sigma_{\mathbf{b}}$) is hard to observe and rarely available

in large quantities. We only really have a small amount of real data which could be used to provide a rough estimate on $\Sigma_{\mathbf{m}}$.

Assumption 3.1.3 *The underground background density* $\mathbf{b}$ *is an Ornstein-Uhlenbeck Gaussian Process with variance* $\Sigma_{\mathbf{b}}$ *and is measured in the standard deviation in* kg/m^3 *over any 5 meter run.*

Throughout this work we will simulate test problems, assuming that both the measurements and underground masses come from some type of multivariate normal distribution when no anomaly is present. When an anomaly is present, we assume the anomaly's effect is added to the existing effects from that multivariate normal system. We will refer to the variance of the underground Gaussian process by examining the expected standard deviation in units of kg/m^3 over a 5 meter run in any direction, so they would follow a spherical correlation model.

3.1.3 Example: Examining a simple tunnel

Let us assume that a single tunnel lies at a x-position of o_x, a y-position from a to b, and a z-position of o_z. This tunnel is embedded in a constant density space. We also recall that then the $T_{xx} - T_{yy}$ measurement is.

$$T_{xx} - T_{yy} = G \int_a^b \frac{\rho(\mathbf{o}) 3 \left((s_x - o_x)^2 - (s_y - o_y)^2\right)}{\left((s_x - o_x)^2 + (s_y - o_y)^2 + (s_z - o_z)^2\right)^{\frac{5}{2}}} do_y \tag{3.14}$$

Let us further assume that we have a set of such measurements, j in $1..m/2$ that are taken at the same y-coordinate (there are $m/2$ measurements because there are two types of measurements taken at each location), directly at the mid-point of a and b. G and $\rho(\mathbf{o})$ can be grouped into a common constant K because the tunnel has constant x and z locations, and the material it exists in is uniform density. This simplifies to the following constant:

$$T_{xxj} - T_{yyj} = K \int_a^b \frac{3\left(s_{xj}^2 - o_y^2\right)}{\left(s_{xj}^2 + o_y^2 + (s_{zj} - o_z)^2\right)^{\frac{5}{2}}} do_y, \tag{3.15}$$

$$T_{xxj} - T_{yyj} = K \frac{o_y \left(o_z^2(3s_{xj} - o_y^2) + 2o_z s_{zj}(o_y^2 - 3s_{xj}^2) + 3s_{xj}^4 + s_{xj}^2(3s_{zj}^2 + o_y^2) - s_{zj}^2 o_y^2 \right)}{\left(o_z - 2o_z s_{zj} + s_{xj}^2 + s_{zj}^2 \right)^2 \left(o_z^2 - 2o_z s_{zj} + s_{xj} + s_{zj} + o_y^2 \right)^{\frac{3}{2}}}. \tag{3.16}$$

Consider a tunnel that is directly perpendicular to the center track in Figure 3.1. For a tunnel 32 meters long, 1.5 meters tall, 2 meters high, centered at $o_x = 0$, with depths of 5, 10 and 15 meters, and sensors at 0.5 meters above the surface ($o_z = [-5.5, -10.5, -15.5]$), embedded in a constant density space of 2500 kg/m^3 the inline gradient measurements are plotted below:

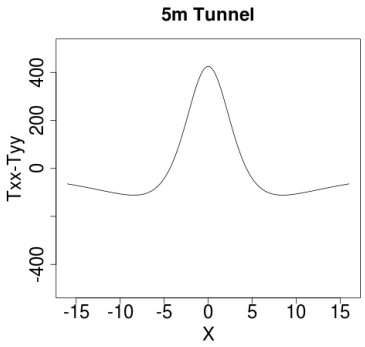

Figure 3.2: 5m Simulated tunnel in a constant density medium.

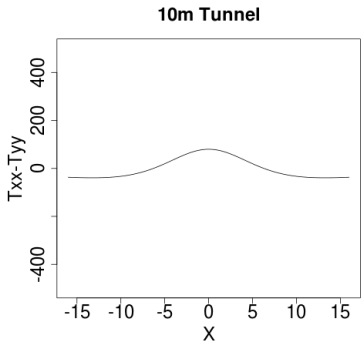

Figure 3.3: 10m Simulated tunnel in a constant density medium.

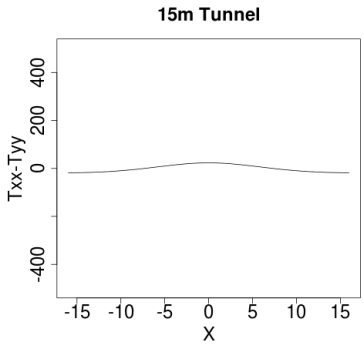

Figure 3.4: 15m Simulated tunnel in a constant density medium.

From these results, it is clear that the peak of the inline gradient measurement corresponds to the x-location of the tunnel, so locating the tunnel in x is trivial (the crossline measurements are omitted, as they are uninformative when tunnels are perpendicular to the measurement field). However, if we are to assume a simple ground noise model, the case is not so clear cut. Going back to the 3-D Gaussian Process we discussed earlier, consider such a process that uses a standard deviation of $50kg/m^3$ over a 5 meter run, $75kg/m^3$ over a 5 meter run, and $100kg/m^3$ over a 5 meter run.

Below are 500 realizations from our ground density model using that process, demon-

strating that the peak of the observations' relationship with the true location of the tunnel becomes far more difficult to detect, especially as depth increases.

The $\sigma = 50, 75, and 100 kg/m^3$ over a 5 meter run experiments (500 realization) are summarized on the following page:

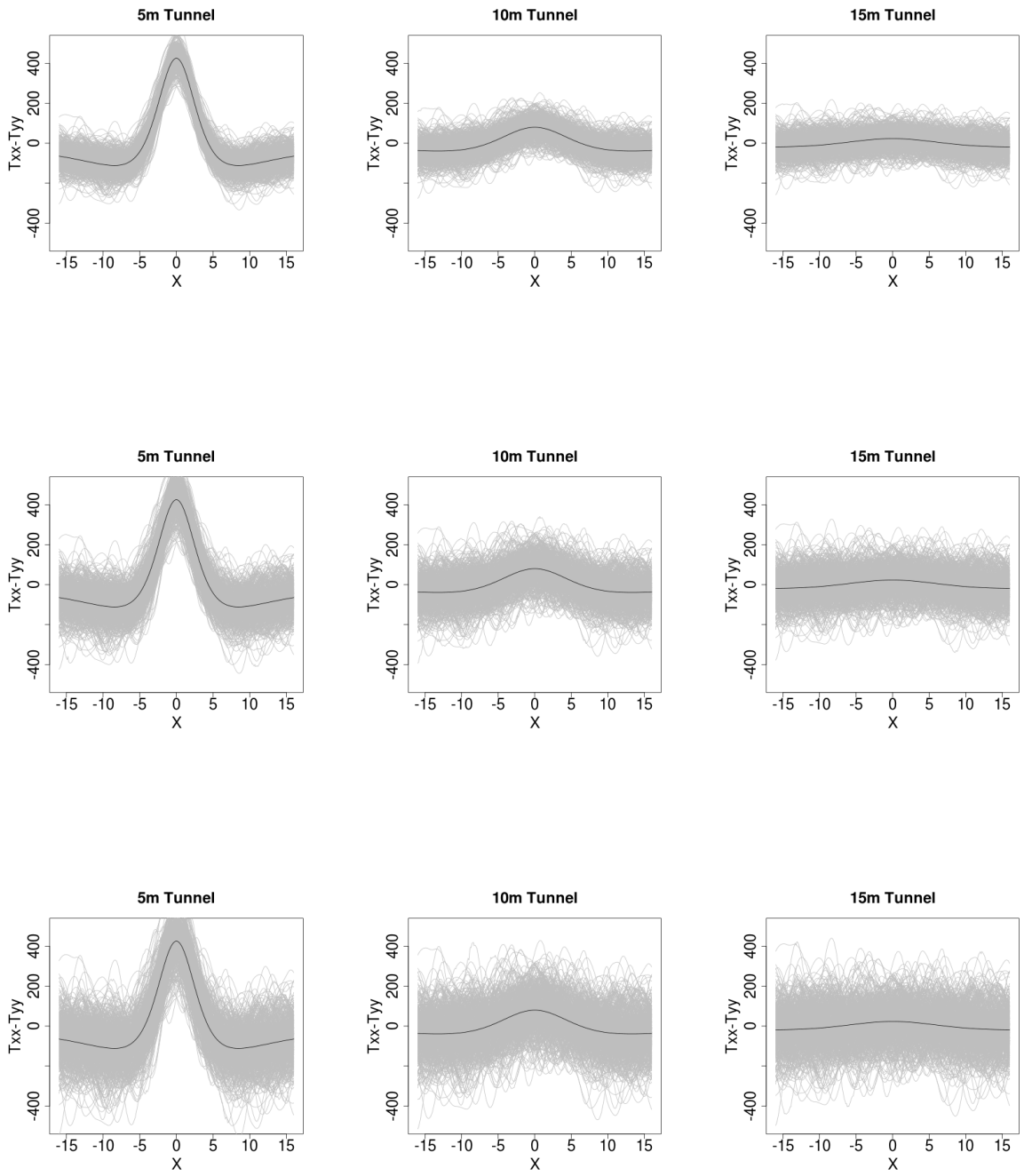

Figure 3.5: 5, 10, and 15 meter tunnels, and background models of $\sigma = 50, 75$, and $100 kg/m^3$

Estimating a confidence interval by finding the empirical standard deviation of the peaks

of these 500 realizations and then using $z_\alpha = 1.96$ yields the following results:

ground stdev	5 m	10 m	15 m
50	+/- 0.70	+/- 8.26	+/- 15.37
75	+/- 1.04	+/- 12.29	+/- 16.81
100	+/- 1.77	+/- 13.90	+/- 17.56

Table 3.1: 95 % CI of peak inline locations for all tunnel depths and variances.

It is clear here from Table 3.1 that anything other than a 5 meter deep tunnel cannot be located solely by examining the peak of the data. Furthermore, such a method would also require prior knowledge of the variance of the underground Gaussian Process, which as we have already stated is very difficult to know.

The following table shows an estimate of the standard deviation for the tunnel's observed peak based on several depths (columns of the table) and several variances over a 5 meter run (rows of the table, in term of standard deviation of kg/m^3).

ground model standard deviation in kg/m^3	Tunnel Depth in meters										
	5	6	7	8	9	10	11	12	13	14	15
5	0.03	0.07	0.12	0.22	0.34	0.48	0.66	0.85	1.05	1.30	1.53
10	0.07	0.14	0.26	0.41	0.62	0.82	1.06	1.37	1.62	2.06	2.54
15	0.11	0.22	0.37	0.59	0.82	1.09	1.41	1.83	2.32	2.96	3.67
20	0.14	0.29	0.48	0.72	1.02	1.36	1.86	2.42	3.09	4.04	4.95
25	0.18	0.35	0.59	0.88	1.20	1.70	2.30	3.06	3.99	5.19	5.98
30	0.22	0.41	0.68	1.00	1.37	2.07	2.81	3.71	4.89	5.91	6.56
35	0.26	0.48	0.76	1.12	1.70	2.50	3.41	4.67	5.70	6.47	7.00
40	0.29	0.54	0.87	1.39	1.96	2.92	4.32	5.41	6.28	6.88	7.41
45	0.33	0.61	0.96	1.59	2.39	3.54	4.76	5.91	6.69	7.14	7.72
50	0.36	0.65	1.19	1.79	2.85	4.17	5.45	6.44	6.96	7.53	7.89
55	0.39	0.70	1.30	2.15	3.34	4.77	5.87	6.71	7.14	7.74	8.09
60	0.43	0.76	1.48	2.50	3.77	5.19	6.31	6.90	7.47	7.84	8.27
65	0.45	1.01	1.67	2.86	4.31	5.57	6.61	7.09	7.71	8.00	8.35
70	0.49	1.06	1.85	3.33	4.74	5.90	6.80	7.25	7.76	8.21	8.46
75	0.52	1.14	2.13	3.72	5.22	6.34	6.90	7.51	7.88	8.28	8.56
80	0.56	1.22	2.49	3.95	5.49	6.62	7.03	7.68	7.99	8.37	8.68
85	0.60	1.33	2.93	4.39	5.78	6.76	7.24	7.76	8.25	8.44	8.74
90	0.62	1.78	3.17	4.71	5.98	6.84	7.41	7.85	8.25	8.58	8.87
95	0.87	1.99	3.46	5.01	6.31	6.97	7.54	7.98	8.33	8.71	8.92
100	0.89	2.02	3.65	5.35	6.53	7.07	7.71	8.10	8.44	8.72	8.95

Table 3.2: Empirical Standard Deviation of the Observed Peak in the Inline Signal (single track)

However, if we take the mean of the peaks of three parallel tracks from Figure 3.1, we are able to reduce the variance by an average of 32.73%.

ground model standard deviation in kg/m^3	Tunnel Depth in meters										
	5	6	7	8	9	10	11	12	13	14	15
5	0.02	0.05	0.10	0.18	0.27	0.38	0.52	0.65	0.78	0.96	1.12
10	0.06	0.11	0.21	0.33	0.47	0.63	0.79	1.01	1.18	1.40	1.67
15	0.08	0.18	0.30	0.46	0.63	0.81	1.05	1.26	1.56	1.91	2.34
20	0.12	0.23	0.38	0.56	0.75	1.01	1.26	1.59	2.00	2.48	3.04
25	0.15	0.28	0.46	0.66	0.88	1.18	1.51	1.98	2.49	3.10	3.56
30	0.18	0.33	0.53	0.77	1.04	1.38	1.88	2.34	3.04	3.52	3.92
35	0.21	0.38	0.59	0.85	1.21	1.64	2.19	2.84	3.42	3.85	4.16
40	0.24	0.42	0.68	0.98	1.39	1.96	2.56	3.20	3.74	4.06	4.39
45	0.26	0.47	0.74	1.12	1.61	2.21	2.90	3.49	3.93	4.25	4.59
50	0.29	0.52	0.81	1.29	1.86	2.53	3.21	3.77	4.09	4.46	4.76
55	0.31	0.58	0.91	1.44	2.10	2.83	3.45	3.93	4.29	4.62	4.91
60	0.34	0.62	1.04	1.63	2.39	3.07	3.72	4.08	4.44	4.77	5.04
65	0.36	0.69	1.16	1.80	2.65	3.29	3.88	4.22	4.59	4.87	5.11
70	0.40	0.75	1.32	2.04	2.82	3.50	3.98	4.35	4.70	4.98	5.19
75	0.46	0.85	1.44	2.26	3.02	3.71	4.10	4.48	4.81	5.04	5.29
80	0.49	0.94	1.61	2.51	3.20	3.85	4.22	4.59	4.89	5.12	5.35
85	0.51	1.06	1.77	2.69	3.37	3.95	4.30	4.69	4.99	5.19	5.41
90	0.56	1.18	1.95	2.84	3.55	4.05	4.42	4.82	5.04	5.27	5.49
95	0.60	1.26	2.13	2.98	3.68	4.12	4.52	4.88	5.10	5.35	5.52
100	0.67	1.37	2.29	3.12	3.84	4.21	4.64	4.92	5.14	5.39	5.55

Table 3.3: Empirical Standard Deviation of the Observed Peak in the Inline Signal, using 3 tracks

Still, the process of estimating this tunnel's x-coordinate by simply observing the peak of the measured signal does not seem practical. So we must turn to other techniques to locate the tunnel.

3.2 The Tunnel and the Background

To review, we shall assume that a given gravimetric signal, measured at the surface is a composition of two main components: a tunnel and a sub-surface background density. $Tunnel + Background = UndergroundDensity$. Figures 3.6, 3.7, and 3.8 illustrate what we mean by the two parts of the underground density.

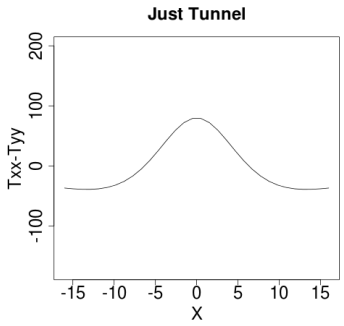

Figure 3.6: 10m tunne signal with no additional variance

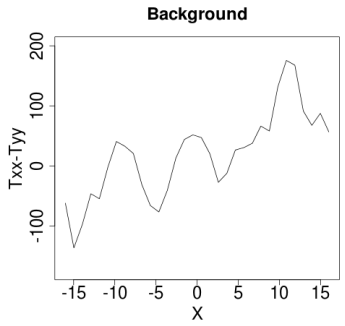

Figure 3.7: Background density from a simulation of $50kg/m^3$

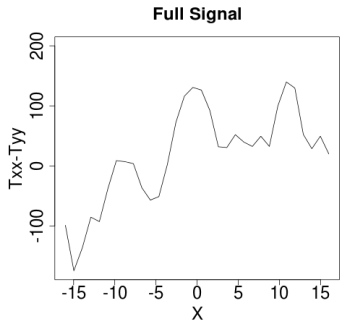

Figure 3.8: A typical inline signal measured at the surface

Our next assumption is that both of these parts of the signal have some sparse representation that we can approximate. We shall assume that the background model comes from the Gaussian Process that we outlined in Chapter 2, and that the standard deviation is a constant $50kg/m^3$ over a 5 meter run. This number comes from some conversations with Geo-sciences researchers, including Professor Michael Brown, however; there is a challenge in finding any set value for this density without knowing much more about the geography of the region in question. We chose a number that would make our problems challenging enough and have some basis in the true underground process.

Assumption 3.2.1 *We assume that the background Gaussian Process has a sparse representation.*

3.3 Sensing Matrixes

3.3.1 Problem in Discrete Form

Now, we shall look at the density integral of the field of $\rho(\mathbf{o})$ as a linear system. We will discretize $\mathbb{D}$ into volume elements. Let i in $[1, ..., n]$ be the index for these discrete volume elements with volume v_i, and let j in $[1, ..., m/2]$ the set of measurements. Then the integral becomes a summation to help find our approximation of $\rho(\mathbf{o})$, which we will now designate $\hat{\rho}_i$:

$$T_{xxj} - T_{yyj} \equiv G \sum_{i=1}^{n} \frac{v_i \hat{\rho}_i 3[(s_{xj} - o_{xi})^2 - (s_{yj} - o_{yi})^2]}{\|\mathbf{s}_j - \mathbf{o}_i\|_2^5}, \tag{3.17}$$

$$2T_{xyj} \equiv G \sum_{i=1}^{n} \frac{v_i \hat{\rho}_i [6(s_{xj} - o_{xi})(s_{yj} - o_{yi})]}{\|\mathbf{s}_j - \mathbf{o}_i\|_2^5}. \tag{3.18}$$

Define a vector of such measurements (in the form $T_{xx} - T_{yy}, 2T_{xy}$), $b \in \mathbb{R}^m$ with $m/2$ measurement sites. We also have $\hat{\rho}(\mathbf{o}) \in \mathbb{R}^n$ for the density of each of the volume elements of size v_i. Also define a matrix $A \in \mathbb{R}^{m \times n}$ which represents the linear system of equations that the above relationships imply.

$$A = \begin{bmatrix}
\frac{3Gv_1[(s_{x1}-o_{x1})^2-(s_{y1}-o_{y1})^2]}{\|\mathbf{s}_1-\mathbf{o}_1\|_2^5} & \cdots & \frac{3Gv_i[(s_{x1}-o_{xi})^2-(s_{y1}-o_{yi})^2]}{\|\mathbf{s}_1-\mathbf{o}_i\|_2^5} & \cdots & \frac{3Gv_n[(s_{x1}-o_{xn})^2-(s_{y1}-o_{yn})^2]}{\|\mathbf{s}_1-\mathbf{o}_n\|_2^5} \\
\frac{6Gv_1[(s_{x1}-o_{x1})(s_{y1}-o_{y1})]}{\|\mathbf{s}_1-\mathbf{o}_1\|_2^5} & \cdots & \frac{6Gv_i[(s_{x1}-o_{xi})(s_{y1}-o_{yi})]}{\|\mathbf{s}_1-\mathbf{o}_i\|_2^5} & \cdots & \frac{6Gv_n[(s_{x1}-o_{xn})(s_{y1}-o_{yn})]}{\|\mathbf{s}_1-\mathbf{o}_n\|_2^5} \\
\vdots & & \vdots & & \vdots \\
\frac{3Gv_1[(s_{xj}-o_{x1})^2-(s_{yj}-o_{y1})^2]}{\|\mathbf{s}_j-\mathbf{o}_1\|_2^5} & \cdots & \frac{3Gv_i[(s_{xj}-o_{xi})^2-(s_{yj}-o_{yi})^2]}{\|\mathbf{s}_j-\mathbf{o}_i\|_2^5} & \cdots & \frac{3Gv_n[(s_{xj}-o_{xn})^2-(s_{yj}-o_{yn})^2]}{\|\mathbf{s}_j-\mathbf{o}_n\|_2^5} \\
\frac{6Gv_1[(s_{xj}-o_{x1})(s_{yj}-o_{y1})]}{\|\mathbf{s}_j-\mathbf{o}_1\|_2^5} & \cdots & \frac{6Gv_i[(s_{xj}-o_{xi})(s_{yj}-o_{yi})]}{\|\mathbf{s}_j-\mathbf{o}_i\|_2^5} & \cdots & \frac{6Gv_n[(s_{xj}-o_{xn})(s_{yj}-o_{yn})]}{\|\mathbf{s}_j-\mathbf{o}_n\|_2^5} \\
\vdots & & \vdots & & \vdots \\
\frac{3Gv_1[(s_{xm/2}-o_{x1})^2-(s_{ym/2}-o_{y1})^2]}{\|\mathbf{s}_{m/2}-\mathbf{o}_1\|_2^5} & \cdots & \frac{3Gv_i[(s_{xm}-o_{xi})^2-(s_{ym/2}-o_{yi})^2]}{\|\mathbf{s}_{m/2}-\mathbf{o}_i\|_2^5} & \cdots & \frac{3Gv_n[(s_{xm/2}-o_{xn})^2-(s_{ym/2}-o_{yn})^2]}{\|\mathbf{s}_{m/2}-\mathbf{o}_n\|_2^5} \\
\frac{6Gv_1[(s_{xm/2}-o_{x1})(s_{ym/2}-o_{y1})]}{\|\mathbf{s}_{m/2}-\mathbf{o}_1\|_2^5} & \cdots & \frac{6Gv_i[(s_{xm/2}-o_{xi})(s_{ym/2}-o_{yi})]}{\|\mathbf{s}_{m/2}-\mathbf{o}_i\|_2^5} & \cdots & \frac{6Gv_n[(s_{xm/2}-o_{xn})(s_{ym/2}-o_{yn})]}{\|\mathbf{s}_{m/2}-\mathbf{o}_n\|_2^5}
\end{bmatrix} \tag{3.19}$$

$$b = \begin{bmatrix} T_{xx1} - T_{yy1} \\ 2T_{xy1} \\ \vdots \\ T_{xxj} - T_{yyj} \\ 2T_{xyj} \\ \vdots \\ T_{xxm/2} - T_{yym/2} \\ 2T_{xym/2} \end{bmatrix}. \tag{3.20}$$

So the system $A\hat{\rho} = b$ is over-determined when $n >> m$, which will almost always be the case, as for any solution to be of practical use, the voxels must be descretized to a small size. We also introduce a noise vector (assumed to have a variance σ from section 3.1.2). The new over-determined system becomes the equation $A\hat{\rho} + \varepsilon = b$. For all of our examples we shall use $m = 192$ from the earlier sensor field assumption, and we shall experiment with $n = [1024, 2048, 4086, 8192]$ voxels. The powers of 2 make some of the transformations that we apply later much simpler to implement.

3.4 Sources of Difficulty

With the system above, note that A is a linear operator in $\hat{\rho}$. However, the inverse problem, that of determining the source $\hat{\rho}$ given b is a much more difficult problem. In fact, passive (we cannot actively change A to help us find $\hat{\rho}$) and time independent inverse source problems of which gravity is one, are among the hardest of inverse problems, even if linear [Isakov, 1990].

3.4.1 Field indeterminacy

The null space of A contains many types objects. Consider a uniform sphere placed at the origin depends only on its mass; any sphere of fixed mass and location is equivalent to a point. Hence, the field measurement of the sphere, measured outside the sphere, does not determine the spheres radius. The only way we can address this problem is to limit our search a certain class of objects, z-convex objects of constant, known density [Isakov, 1990].

3.4.2 Algebraic indeterminacy

The second challenge comes from the discrete and noisy nature of real measurements. We have discretized the continuous density into a finite set of voxels, but very often, we have fewer measurements than voxels. Therefore, we have to solve an indeterminate linear system. To solve this issue, we will turn to Compressed Sensing. We are also physically constrained to measure the gravity field only above the ground. Algebraically this means that the discretized A may be ill-conditioned as well.

.

3.5 Voxel Representation

Continuing with the sensing matrix A, we then perform some simple transformations to get it into the form that will let us use a Compressed Sensing linear program. Right now, our vector $\hat{\rho}$ that we are trying to solve for has lots of non-zero entries, and a few zero entries (if a void exists). But we actually need the opposite condition of lots of zero entries and a few non-zero terms for the sparse representation to hold. To create that, we subtract off the average density $\bar{\rho}$ from the $\hat{\rho}$ vector. Since this is also a linear operator, we can specify $\theta = \hat{\rho} - \bar{\rho}$, and we have a new linear system: $C\theta + \varepsilon' = b$, where C is the new linear operator that applies the gravimetric operator and the subtraction of the mean. Also, since our new formulation changes our sparsity properties, the residual error will also change, denoted by ε'. So we can specify the following linear program (the VX subscript signifies a Voxel-space representation, in both the Lasso version

$$\begin{gathered} \min \|\theta_{VX}\|_1 \\ \text{st} \\ \|C_{VX}\theta_{VX} - b\|_2 \leq \varepsilon_{VX} \end{gathered} \tag{3.21}$$

and the bounded noise version

$$\begin{gathered} \min \|\theta_{VX}\|_1 \\ \text{st} \\ \|C_{VX}\theta_{VX} - b\|_\infty \leq \varepsilon_{VX} \end{gathered} \tag{3.22}$$

	Number of Voxels			Voxel size (meters)		
Dictionary Size	x	y	z	x	y	z
1024	16	8	8	2	0.5	2
2048	16	16	8	2	0.25	2
4096	16	16	16	2	0.25	1
8192	32	16	16	1	0.25	1

Table 3.4: Voxel Dictionary (VX) composition

This is the essence of the Compressed Sensing formulation. It will seek to use as few values of θ_{VX} as possible to make the map of $C_{VX}\theta_{VX} - b$ stay constrained within some maximum deviation of ε_{VX}. We shall consider four different Voxel Dictionary types of sizes 1024, 2048, 4096, and 8912 voxels respectively inside of the sensing domain. The sizes are chosen because powers of 2 make some of the transformations we will apply later easier, and because we want to be able to compare dictionaries to each other using the same size.

The composition of those dictionaries in terms of voxels is shown in Table 3.4.

3.5.1 Dictionary Types

We can also choose different representational systems, hereafter called dictionaries, to ensure the proper representation of the features we seek to detect. Dictionaries can come from many possible sources, but we will distinguish two primary types. The first is a dictionary designed to match some natural feature that is expected in the underground density; we will call this a "feature-space dictionary". A voxel dictionary would be one example of a feature-space dictionary, as it corresponds to the locations in the domain $\mathbb{D}$ and their densities. The second aims to find sparsity and incoherence by looking directly at the measurements; we will call this a "measurement-space dictionary". A Fourier basis could be a type of "measurement-space dictionary", as it attempts to break the signal down into a sum of waveforms. Our density signal ρ is comprised of two main parts: an anomaly (which may or may not be present), and the background (i.e.: the non-anomaly ground

density variations). We shall discuss the selection of the right type of dictionary to assign to each signal type with consideration to making that dictionary sparse enough for the compressed sensing program to work. In the context of machine learning, the dictionary represents elements of the signal that we think are fundamental to its representation, so more generally, dictionary selection could be thought of as a feature selection problem [Yang and Pedersen, 1997].

3.5.2 Changing Dictionaries

In addition to subtracting off the mean density, we can also change the dictionary to ensure our needed sparsity using a transformation matrix. We will discuss two types: a general change of dictionary, and an orthonormal change of dictionary. A general change of dictionary can be done with a full rank square matrix F. Assume such an operator F exists, such that $C_{old}F^{-1}F\theta_{old} + \varepsilon_{new} = b$. (We again use ε_{new}, as the error may have changed if the new representational system is better or worse are representing the data). Then, define $C_{new} = C_{old}F^{-1}$ and $\theta_{new} = F\theta_{old}$.

However, we may wish to use an orthonormal basis instead. Several methods can be used to create an orthonormal basis, such as the a Daubechies wavelet [Daubechies et al., 1992]. As long at the operator is a unitary matrix, it works for our purposes. Assume such an operator F exists, such that $C_{old}F^{T}F\theta_{old} + \varepsilon_{new} = b$. Then, define $C_{new} = C_{old}F^{T}$ and $\theta_{new} = F\theta_{old}$. This is where Definition 2.2.7 (D-RIP) is important, as the D-RIP properties are preserved under an orthonormal transform.

3.6 Incoherence Properties of Dictionaries

Examining the incoherence measures we defined in Chapter 2, we can quickly see that we don't have much of a hope of getting RIP with the Voxel Matrix.

	Dictionary Size			
	1024	2048	4096	8192
Voxels	0.997	0.999	1.000	1.000

Table 3.5: Max Coherence μ: Voxel Dictionary

	Dictionary Size			
	1024	2048	4096	8192
Voxels	0.963	0.988	0.989	0.989

Table 3.6: Average Coherence ν: Voxel Dictionary

The closer these values are to 1, the worse off we are, so it is clear that the Strong Coherence Property will not be met.

3.6.1 Bounding the probability of weak RIP using probability of Max Coherence

We hope to create some type of probabilistic bound on having weak RIP, despite not having a strong coherence property, so we set out to examine the subsets of s-sparse columns of a dictionary, C. This assumes we can quickly compute all the possible inner products of the columns of a normalized dictionary, which would be $n(n-1)/2$ inner products. We further assume that the inner product between two columns representing voxels decreases as a constant inverse factor of the distance between those two voxels.

First, we shall make use of Gerschgorin's theorem.

Theorem 3.6.1 *(Gerschgorin's Theorem) Suppose that $A \in \mathbb{R}^{n\times n}$ be a matrix with entries $a_{i,j}$, and let $\mathfrak{B}_i \subseteq \mathbb{R}$ be the ball centered at $a_{i,i}$ with radius:*

$$r_i = \sum_{j=1, j\neq i}^{n} |a_{i,j}| .$$

Then the eigenvalues of A lie in the union $\cup_{i=1}^{n}\mathfrak{B}_i$ [Gerschgorin, 1931].

Let C_s be an s-sized subset of the columns of a dictionary, C. If $\lambda_{\min}$ an $\lambda_{\max}$ are the max eigenvalues of $C_s^T C_s$, to show weak RIP for s, we need:

$$1 - \delta_s \leq \lambda_{\min} \leq \lambda_{\max} \leq 1 + \delta_s$$

for some $\delta_s < 1$. Since C_s has unit norm columns, the matrix $A = C_s^T C_s$ has entries $a_{i,j} = \langle \mathbf{c}_i, \mathbf{c}_j \rangle$, and $a_{i,i} = 1$, so:

$$r_i = \sum_{j \in s, j \neq i} |a_{i,j}| = \sum_{j \in s, j \neq i} |\langle \mathbf{c}_i, \mathbf{c}_j \rangle| \leq \mu(C) \sum_{j \in s, j \neq i} 1$$

So, $r_i \leq \mu(C_s) \times (s-1)$, and Gerschgorin's theorem gives us:

$$1 - \mu(C_s) \times (s-1) \leq \lambda_{\min} \leq \lambda_{\max} \leq 1 + \mu(C) \times (s-1).$$

So C satisfies RIP of order S with the following condition on δ_s:

$$\delta_s \leq (s-1)\mu(C_s),$$

$$\forall s < 1/\mu(C_s).$$

So, if we have probability of $\mu(C_s) \leq \mu_B$, a threshold for a given s, we can find the probability of a δ_s for RIP.

Generally, enumerating all $\binom{n}{s}$ sets of columns is not computationally feasible, but if we are using voxels (or anything with a spatial relationship to incoherence), we have another way. First, we need to find the value for μ_B. Let us assume that the inner product of the columns of for any two voxels is inversely proportional to the distance of those two voxels, and that our voxels fill a space that is $n_1 \times n_2 \times n_3$. This means our dictionary C would have $n = n_1 \times n_2 \times n_3$.

Fix $\delta_s = 1$ to get a bound for the worst possible adherence to RIP. This translates to a requirement of $\mu(C) \leq 1/(s-1)$, so we will compute $Pr(\mu(C_s) \leq 1/(s-1))$ to determine the value of p for weak RIP. Furthermore, we shall say that r_{min} is the minimum radius that guarantees that $\mu_B \leq 1/(s-1)$.

If our field has n voxels in the space, denote the number of voxels inside radius r_{min} of a given voxel by constant number of points b. We get the following probability bound for weak-RIP at $\delta_s < 1$:

Theorem 3.6.2 *A given set of n voxels has weak-RIP with p bounded by the following probability:*

$$Pr(\mu(C_s) < \mu_B) > 1 \left(\frac{n-b}{n-1}\right) \left(\frac{n-2b}{n-2}\right) ... \left(\frac{n-(s-1)b}{n-s-1}\right)$$

Where C_s is a random subset of of s columns from the voxel dictionary, $\mu_B < 1/(s-1)$, and b is the number of voxel centers inside the r_{min} radius where the dot product of that voxel's corresponding column's dot product exceeds $1/(s-1)$.

The proof of this theorem is simply based on the fact that the sample space is reduced by a smaller available number of voxels to chose from at each selection, staring with the first voxel, which is any voxel (represented by the 1), then the probability of choosing a voxel outside of r_{min} of the first voxel is $\left(\frac{n-b}{n-1}\right)$, and the probability of choosing the next voxel outside of r_{min} of the first two voxels is $\left(\frac{n-2b}{n-2}\right)$ and so on until the sparsity is achieved.

This is an over-estimate of the probability for weak RIP, since less than b voxels are removed from consideration if an already chosen voxel is near the border of the space. How much of an overestimate this is depends upon how large r_{min} is.

We can also look at this in the continuous case. Assuming now that the voxels are chosen from a 3-D uniform process, instead of a discrete grid.

Theorem 3.6.3 *A given set of n voxels (chosen from a random uniform 3D process) has weak-RIP with p bounded by the following probability:*

$$Pr(\mu(C_s) < \mu_B) > 1 \left(1-\omega\right) \left(1-2\omega\right) ... \left(1-(s-1)\omega\right)$$

Where C_s is a random subset of of s columns from the voxel dictionary, $\mu_B < 1/(s-1)$, and ω is the volume of the r_{min} radius sphere normalized to the volume of the space the voxels are in where the dot product of that voxel's corresponding column's dot product exceeds $1/(s-1)$.

Similarly to the proof of the discrete case, the sample space of choosing a point outside of r_{min} diminishes at each step, $(1-\omega)$, $(1-2\omega)$, and so on until sparsity is reached.

Like the discrete case, this is also an over-estimate, as it ignores the fact that points chosen near the border don't reduce the consideration space for future points as much as points closer to the interior, but it can serve as a bound. This can also be adapted into

a form for a general bound for any dictionary, where we assume the worst-case situation of taking the dictionary element with the maximum number of inner products that exceed the threshold μ_b, and set that number equal to b, and then $\omega = b/n$. This gives us a fast estimate on the probability of any dictionary having $\delta_s < 1$.

The probability that a given dictionary C has weak-RIP can be bounded by finding the column with the greatest number of coherence values that violate a minimum coherence condition. If the coherence is a function of a spatial relationship, then the bound estimate can be improved upon by enumerating border cases (which may be computationally prohibitive).

3.6.2 Empirically Estimating the Incoherence of a Single Dictionary by SVD

Another way of checking the conditioning of s-sized sub-matrices of C is by sampling the normalized columns and computing the max and min singular values ($\lambda_{\min}$ and $\lambda_{\max}$) and recording them. From this we can compute a notional δ_s. Recall from earlier that if $\lambda_{\min}$ and $\lambda_{\max}$ are the max eigenvalues of $C_s^T C_s$, to show weak RIP for s, we need:

$$1 - \delta_s \leq \lambda_{\min} \leq \lambda_{\max} \leq 1 + \delta_s$$

for some $\delta_s < 1$. So, we can compute the notional $\hat{\delta}_{s[l]}$ by recording the following quantity every time we take an s-sized sub-matrix in our $l = 1...L$ trials:

$$\hat{\delta}_{s[l]} = \max\left[1 - \lambda_{\min}, \lambda_{\max} - 1\right]$$

This is demonstrated in the Voxel dictionary below (where $L = 100,000$):

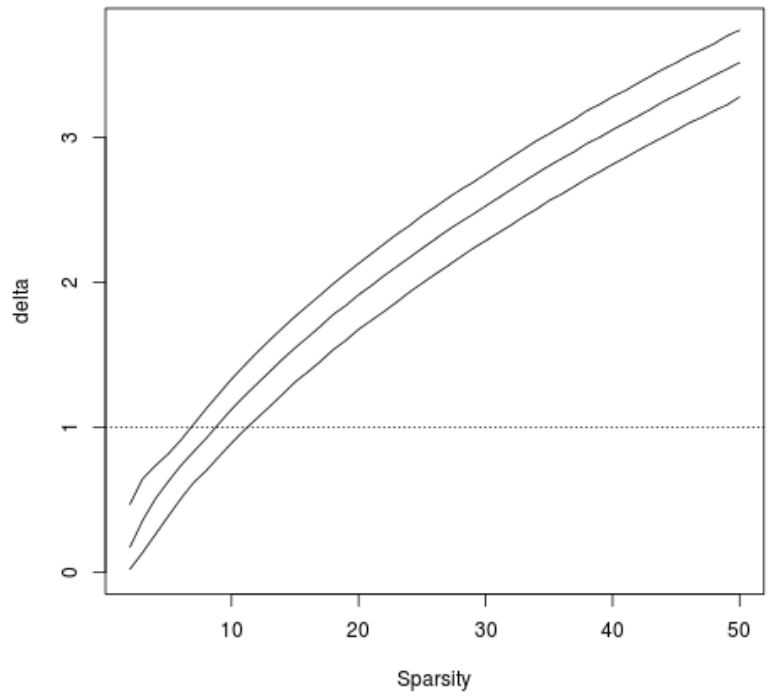

Figure 3.9: .1,.5,.9 quantiles of δ_s (isometry constant) vs sparsity, VX dictionary

The lines represent the 0.1, 0.5, and 0.9 quantiles. From this it is clear that for $s > 10$, our probability of having weak-RIP is exceedingly small (recall $\delta_s < 1$ must hold for RIP).

Chapter 4

DICTIONARIES

In this section I talk about the various methods for constructing, evaluating, and selecting dictionaries to solve the Compressed Sensing problem to locate underground tunnels. A dictionary differs from a measurement matrix, as a measurement matrix is specific to the gravimetric linear operator's relationship with underground density, whereas a dictionary may or may not be related to the underground density and may just be attempting to create a sparse representation of the background

4.1 Feature Space Dictionaries

We have already discussed the voxel dictionary. In this section, we shall propose two other types of feature space dictionaries.

4.1.1 Random Lines Dictionary

The Random Lines dictionary is a feature-space dictionary aiming to represent the tunnel-shaped anomalies. Each element of the Random Lines dictionary is a line segment location of random length and orientation. These lines will be used to create a dictionary for tunnels or other man-made structures that are thought to have straight lines.

To generate a Random Line dictionary element, we will choose the center of the line by picking a point from a random uniform 3-d process for the half-sphere that encompasses the sensing domain $\mathbb{D}$. Then, we choose a uniformly random angle in X-Y plane, and a uniformly random angle orthogonal to the X-Y plane. Lastly, we choose a random length for the line. To compute the elements of the measurement matrix for the random line, we place a set of r points on this line, giving them an equal volume, v. Each point is located at $\mathbf{o}_{ik}$

Now we modify Equation 3.17 and Equation 3.18 by summing over those locations and

volumes:

$$C_{RL(i,2j-1)} = \sum_{k=1}^{r} \frac{3Gv[(s_{xj} - o_{xik})^2 - (s_{yj} - o_{yik})^2]}{\|\mathbf{s}_j - \mathbf{o}_{ik}\|_2^5} \tag{4.1}$$

$$C_{RL(i,2j)} = \sum_{k=1}^{r} \frac{6Gv[(s_{xj} - o_{xik})(s_{yj} - o_{yik})]}{\|\mathbf{s}_j - \mathbf{o}_{ik}\|_2^5} \tag{4.2}$$

This parametrization allow us to propose many different locations, lengths, and orientations for the tunnels. The Random Lines dictionary was developed by Hoyt Koepke; however has not appeared in a published work to date. In experiments, I used dictionaries of 1024, 2048, 4096, and 8192 random lines. The length of the lines was allowed to vary from 2 to 10 meters.

The drawback is, we need to make the basis very large if we have any hope of finding the true tunnel anomaly in a large domain $\mathbb{D}$. There is also the danger of lots of lines fitting the noise.

To counteract this, we will attempt to create a dictionary to model the smoothly varying spatially correlated density incoherent with the random lines. Section 4.2.2 will examine a feature space dictionary to attempt this, and Section 4.3 will examine several measurement space dictionaries to accomplish this.

4.1.2 Wavelet-sparsified Voxel Dictionary

Let us briefly describe how a 3-D Wavelet transformation would be applied to a notional voxel sensing matrix as a candidate for C_B. A multi-dimensional wavelet transformation can be constructed by sequential application of 1-D transformations, 1 axis at a time. This is reflected in the equation below where F^{-1} is the linear operator that will apply the wavelet transformation.

$$C_{WV} = C_{VX}F^{-1} \tag{4.3}$$

The question now becomes one of filter selection that can model the underlying Gaussian Process we believe the background density to come from while remaining incoherence wit the anomaly dictionary. Wavelet Sparsified voxel dictionary to depict the noise, this will

not give us the incoherence we need. if we know the covariance matrix of that Gaussian Process, we could chose a filter that matches it by examining the eigenvectors of that matrix. However, in practice this matrix is not known, therefore; we choose the Daubches length 8 filter (DB8), as it gives us a balance of smoothness and representation. The DB8 filter is shown below.

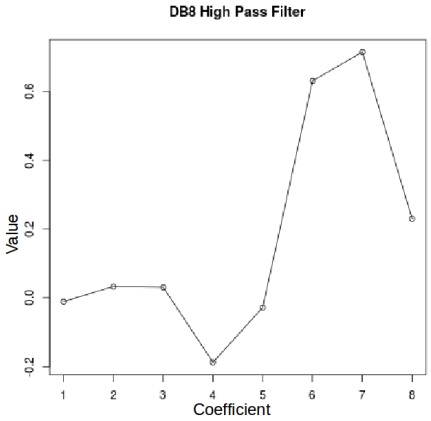

Figure 4.1: DB8 High Pass

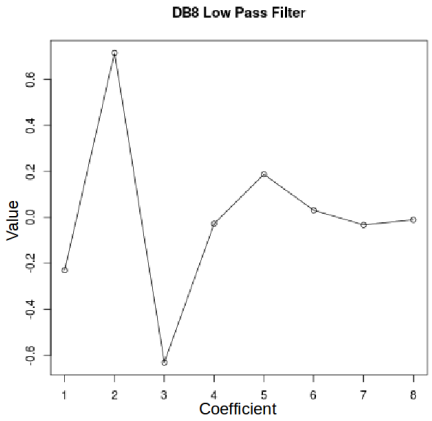

Figure 4.2: DB8 Low Pass

The wavelet filter implementation we will use has periodization to handle the problem when the filter length exceeds the length of the signal at the lower filter levels. What this means is that the coefficients at the far ends are wrapped and added to the transformation matrix at the opposite end, to the effect of aliasing. This aliasing is very small in practice.

In our implementation, we cannot use the same voxels from Chapter 3, as they are not the same size in all three dimensions. We created a $32m \times 32m \times 32m$ cube centred under the sensor field and only allowed a C_{VX} coefficient to be used if it was within the voxel range that we wanted to use. The sizes of those ranges for our dictionaries are shown in Table 4.1.

Number of Voxels	X dim		Y dim		Z dim	
1024	-8	8	-4	4	0	-8
2048	-8	8	-4	4	0	-16
4096	-8	8	-8	8	0	-16
8192	-8	16	-8	8	0	-16

Table 4.1: Voxel Dictionary sensing domain ranges (in meters) for Wavelet Transforms

While this transform may buy us some sparsity, it is important to note that its effects on the Restricted Isometry property are not known. We will examine weak-RIP and statistical orthogonality of the wavelet-sparsified voxel dictionary using empirical experiments.

4.2 Measurement Space Dictionaries

For modelling ground noise, because we don't want to reconstruct it perfectly, we can construct a dictionary directly in measurement space. Dictionaries of this sort have an advantage as they are much lower dimensional and can be constructed to have the desirable properties of RIP, sparsity, and mutual incoherence with other feature-space dictionaries, as they are not constrained by adhering to the gravimetric linear operator. All of these dictionaries will all be considered as candidates for C_B.

4.2.1 Gaussian Dictionary

Random ensembles are largely incoherent with any fixed basis [Candes and Tao, 2006, Rauhut, 2010]. This is a useful property which allows us to non-adaptively choose a random dictionary, C for any type of signal. We will use a simple Gaussian ensemble as our naive baseline.

The purpose of this proof is to show that even if the requirements for RIP are not met, some probability of weak-RIP is always calculable, as well as provide a baseline for our multivariate Gaussian proof later in this work.

Theorem 4.2.1 *For an I.I.D. Gaussian ensemble dictionary that is $m \times n$ and θ, a random*

s-sparse vector with equal entries, the weak-RIP holds with the following probability on the δ_s*:*

$$Pr\left[\left|\frac{\chi_m^2}{m} - 1\right| \leq \delta_s\right],$$

where χ_m^2 *is a chi-squared random variable with* m *degrees of freedom.*

Proof 4.2.2 *Consider a dictionary made up of I.I.D. Gaussian* $(0, 1/\sqrt{m})$ *entries. We shall designate this dictionary,* $C_{GS} \in \mathbb{R}^{m\times n}$*, and we are solving for a sparse vector* $\theta_{GS} \in \mathbb{R}^m$*. In this notation the RIP looks like this:*

$$(1 - \delta_S) \|\theta_{GS}\|_2^2 \leq \|C_{GS}\theta_{GS}\|_2^2 \leq (1 + \delta_S) \|\theta_{GS}\|_2^2 \tag{4.4}$$

We can simplify this as follows:

$$\left|\|C_{GS}\theta_{GS}\|_2^2 - \|\theta_{GS}\|_2^2\right| \leq \delta_S \|\theta_{GS}\|_2^2 \tag{4.5}$$

Further assume that any random s-sparse vector, θ_{GS} *is a set of* $\frac{1}{\sqrt{s}}$ *entries so that* $\|\theta_{GS}\|_2^2 = 1$*.*

so we are left with:

$$\left|\|C_{GS}\theta_{GS}\|_2^2 - 1\right| \leq \delta_S \tag{4.6}$$

Now we really have s *columns of* C_{GS} *that have been scaled by a factor of* $\frac{1}{\sqrt{s}}$*. So, then* $C_{GS}\theta_{GS}$ *becomes:*

$$C_{GS}\theta_{GS} = \sum_{l=1}^{s} \frac{c_{GS(l)}}{\sqrt{s}} \tag{4.7}$$

Where $c_{GS(l)}$ *are the selected columns of* C_{GS}*. Since we scaled them by a factor of* $\frac{1}{\sqrt{s}}$*, the variance of the* $C_{GS}\theta_{GS}$ *is the same as the variance of any column of* C_{GS}*, in this case it is simply* $\frac{1}{\sqrt{m}}I_m$*.*

So now, we are simply summing up m *squared I.I.D. Gaussian variables, so it follows that* $C_{GS}\theta_{GS} \sim \frac{\chi_m^2}{m}$*.*

Which proves that a bound of weak-RIP is

$$Pr\left[\left|\frac{\chi^2_m}{m} - 1\right| \leq \delta_s\right]. \tag{4.8}$$

This allows us to compute the probability of weak-RIP for any size Gaussian dictionary.

4.2.2 K-SVD Learned Dictionary

The K-SVD algorithm is a method of creating overcomplete dictionaries for sparse signal representation [Aharon et al., 2006]. We will utilize it as a method to find a background dictionary, C_B. The KSVD algorithm is simple, and designed to be a truly direct generalization of the K-Means clustering algorithm. As such, when forced to work with one column per signal, and have a unit coefficient for this column, it exactly reproduces the K-Means algorithm. Of course, it suffers from all the shortcomings of the K-Means, and in particular, it might be trapped into a local minimum solution. To use this method, we require a large library of training measurement data created from the same ground density distributions as the test problem. These training examples must not contain tunnels. We can either take these problems from real data or generate them through simulation.

K-SVD requires some sparse solver. For this implementation, we will be using Batch OMP [Rubinstein et al., 2008] to attempt to speed up the training process. Other pursuit algorithms could be used, but the complexity properties of Batch OMP stay under control by comparison.

Algorithm 2 K-SVD

1: Input: Measurement set $\mathbf{B} \in \mathbb{R}^{m \times n}$, initial Dictionary $\mathbf{C}_0 \in \mathbb{R}^{m \times n}$, target sparsity s, number of iterations $maxiter$
2: Output: Dictionary C and sparse matrix $\boldsymbol{\theta}$ such that $\mathbf{B} \approx C\boldsymbol{\theta}$
3: Initialization: Set $C := \mathbf{C}_0$
4: **for** $iter = 1...maxiter$ **do**
5: $\quad \forall i : \boldsymbol{\theta}_i := \underset{\theta}{\text{Argmin}} \|b_i - C\theta\|_2^2$ Subject to $\|\theta\|_0 \leq s$ (by Batch OMP)
6: $\quad$ **for** $j = 1...n$ **do**
7: $\quad\quad C_j := \mathbf{0}$
8: $\quad\quad I :=$ indices of the signals in $\mathbf{B}$ that use C_j from $i-1$
9: $\quad\quad \mathbf{E} := \mathbf{B}_I - C\boldsymbol{\theta}_I$
10: $\quad\quad c, g := \underset{c,g}{\text{Argmin}} \left\|\mathbf{E} - cg^T\right\|_F^2$ Subject to $\|c\|_2 = 1$
11: $\quad\quad C_j := c$
12: $\quad\quad \boldsymbol{\theta}_{j,I} := g^T$
13: $\quad$ **end for**
14: **end for**
15: Return: C_n, $\boldsymbol{\theta}_n$

There are two major computational bottlenecks in Algorithm 2 that drastically affect the speed of K-SVD. The first is in Line 9 of Algorithm 2, when we compute the matrix $\mathbf{E}$, and the second is in Line 10 of Algorithm 2when we have to compute an SVD. Therefore, the following implementation circumvents those matrix and SVD calculations [Rubinstein et al., 2008]. K-SVD only converges to a local optimum and can really only be viewed as an improvement algorithm as it just improves a dictionary, but does not find the best possible dictionary [Aharon et al., 2006], so some approximation of the SVD is probably reasonable.

Algorithm 3 Approximate K-SVD

Input: Measurement set $\mathbf{B} \in \mathbb{R}^{m \times n}$, initial Dictionary $\mathbf{C}_0 \in \mathbb{R}^{2m \times K}$, target sparsity s, number of iterations $maxiter$
Output: Dictionary C and sparse matrix $\boldsymbol{\theta}$ such that $\mathbf{B} \approx C\boldsymbol{\theta}$
Initialization: Set $C := \mathbf{C}_0$
for $iter = 1...maxiter$ **do**
    $\forall i : \boldsymbol{\theta}_i := \underset{\theta}{\operatorname{argmin}} \|b_i - C\theta\|_2^2$ Subject to $\|\theta\|_0 \leq s$
    for $j = 1...n$ **do**
        $C_j := \mathbf{0}$
        $I :=$ indices of the signals in $\mathbf{B}$ that use C_j from $i-1$
        $g := \boldsymbol{\theta}_{j,I}^T$
        $c := \mathbf{B}_I g - C\boldsymbol{\theta}_I g$
        $c := c / \|c\|_2$
        $g := \mathbf{B}^T c - (C\boldsymbol{\theta})^T c$
        $C_j := c$
        $\boldsymbol{\theta}_{j,I} := g^T$
    end for
end for

Complexity of various K-SVD applications is explored by [Sahoo and Makur, 2013]. One of the challenges is that we must know the measurement locations on all the training problems, and those locations cannot be irregular without using some type of interpolation to create test examples that are from the same distribution.

4.2.3 Sample Problem Dictionary

If we have training data available without tunnels, we can use the data itself as a dictionary. If we assume that the observed signal comes from some multivariate normal distribution:

$$C_i \overset{i.i.d.}{\sim} MVN_m(\mathbf{0}, \Sigma). \tag{4.9}$$

The question becomes, can we find a way to represent the specific instance of that multivariate normal random variable using a dictionary of multivariate normal realizations? A multivariate normal random variable can be represented as a linear combination of IID multivariate normal random variables. Using the same notation, where the θ_i values represent the unique solutions to the linear system, we have the following:

$$Var\left(\theta_i \mathbf{c}_i\right) = \sum_i \left(\theta_i^2 Var(\mathbf{c}_i)\right) = \left(\sum_i \theta_i^2\right) \Sigma. \tag{4.10}$$

So, the correlation will always be the same as the generating distribution, and the covariance will be the same as the generating distribution if $\sum_i \theta_i^2 = 1$. What is useful about this, is that we can use training data that has a different variance than the example problem that we are able to filter the anomaly from, and then estimate the variance in the example problem by analyzing the solution's value of $\sum_i \theta_i^2$. We can constrain our linear system by adding the quadratic constraint: $\sum_i \theta_i^2 = 1$ if we are fairly certain that the training data has the same variance as the test data. However, adding a quadratic constraint to our problem massively increases the computational complexity of our solution.

Multivariate Gaussian dictionaries can also be shown to meet the compressed sensing requirements using the Restricted Eigenvalue Condition [Raskutti et al., 2010]. The purpose of this proof is to show that even if the requirements for RIP are not met, some probability of weak-RIP is calculable.

Theorem 4.2.3 *The example problem dictionary that is $m \times n$ has (δ_s, p) weak-RIP with the following probability based on δ_s*

$$Pr\left[\left|\chi_m^2(\gamma) - 1\right| \leq \delta_s\right],$$

where γ is a non-centrality parameter that is determined by Σ and μ of the Gaussian Process that generates the example problem dictionary.

Proof 4.2.4 *Now we turn to the case where our selection of the columns of C approximates a generic multivariate norm with mean μ and correlation structure Σ. So in this case we have:*

$$c_{(l)} \sim_{i.i.d} MVN_m(\mu, \Sigma) \tag{4.11}$$

Beginning here with an $C \in \mathbb{R}^{m \times n}$ *dictionary and a* $\theta \in \mathbb{R}^n$ *s-sparse vector. Just as we did in (4.9) in the uncorrelated Gaussian proof, modifying the RIP from Definition 2.2.3 gives us this:*

$$\left| \|C\theta\|_2^2 - \|\theta\|_2^2 \right| \leq \delta_S \|\theta\|_2^2 \tag{4.12}$$

Again, assume that any random s-sparse vector, θ *is a set of entries so that* $\|\theta\|_2^2 = 1$. *so we are left with:*

$$\left| \|C\theta\|_2^2 - 1 \right| \leq \delta_S \tag{4.13}$$

Now we really have s columns of C *that have been scaled by a factor of* $\frac{1}{\sqrt{s}}$. *So, then* $C\theta$ *becomes:*

$$C\theta = \sum_{l=1}^{s} \frac{c_{(l)}}{\sqrt{s}} \tag{4.14}$$

Again the variance of the $C\theta$ *is the same as the variance of any column of* C *(or* c_l*, which is now* Σ.

Now we are summing up the entries of a multi-variate normal instead of i.i.d., but this sum is still a linear operator, so the variance is still traceable.

So, we can creating a sum that is properly stabilized with respect to variance (which is very close to the L2 norm which we seek):

Starting with some notations:

$$Y \equiv (C\theta)^T \Sigma^{-1} C\theta, \tag{4.15}$$

$$Z \equiv \Sigma^{-1/2} C\theta, \tag{4.16}$$

$$W \equiv \Gamma Z, \tag{4.17}$$

where Γ *is some orthogonal* $m \times m$ *matrix with first row* $\Sigma^{1/2}\mu / \sqrt{\mu^T \Sigma^{1/2} \mu}$.

Then

$$W \equiv \Gamma Z \sim MVN_m(\Gamma\Sigma^{1/2}, \Gamma\Gamma^T) = MVN_m((|\Sigma^{1/2}\mu|, 0, ..., 0)^T, I_m) \tag{4.18}$$

So then

$$Z \sim MVN_m(\Sigma^{1/2}\mu, I_m) \tag{4.19}$$

and

$$Y \equiv (C\theta)^T\Sigma^{-1}C\theta = Z^TZ = W^T\Gamma\Gamma W = W^TW \tag{4.20}$$

which yields a non-central χ^2_m distribution with location parameter $\gamma = \mu^T\Sigma^{-1}\mu$

In order to get a probabilistic bound on RIP for this dictionary, we must compute:

$$Pr\left[\left|\chi^2_m(\gamma) - 1\right| \leq \delta_s\right] \tag{4.21}$$

This now allows us to find a probability for weak-RIP for any multivariate Gaussian dictionary, demonstrating that if we can classify a dictionary as coming from a multivariate Gaussian, some probability of weak-RIP exists.

4.2.4 Incoherence Between Two Dictionaries by Simulation

In addition to all the other incoherence metrics that we have outlined, we can also examine the mutual incoherence of the dictionaries by generating problems from one dictionary, and attempt to solve them in the other. If the dictionaries have a high degree of mutual incoherence, the residual error will always be better in the trained dictionary. This will also give us some clue in finding the correct size and sparsity for both of our dictionaries.

To judge mutual incoherence, we will generate test problems from one dictionary and solve them in another dictionary. Since we will be doing this with a large number of problems, we will return to using Batch OMP [Rubinstein et al., 2008] as the sparse decoding algorithm, as the process will need to be done many times. Differing from 4.1 and 4.2, Batch OMP requires a target sparsity value s, as opposed to using a λ of σ parameter to control sparsity.

We begin by normalizing the columns of our dictionaries, in this case C_A and C_B. Then we shall set θ_B equal to a random set of s_B coefficients of the form, where θ_{RAW} is vector of 1's and 0's with only s_B terms equal to 1:

$$\theta_B = \frac{1}{\sqrt{s_B}}[\theta_{RAW}]$$

Then we will compute $b_B = C_B\theta_B$, and attempt solve batch OMP problem using C_A using Batch OMP with the target sparsity s_A and record the average L_2 norm error over all of our generated signals. We will repeat this process with by generating signals from the C_A and attempting to solve them with C_B. Since we have already chosen the sparsity for C_B, the only parameter that we need to vary is s_A in order to find the correct sparsity that maximizes incoherence between the dictionaries. This is demonstrated with the K-SVD and Random Lines dictionaries

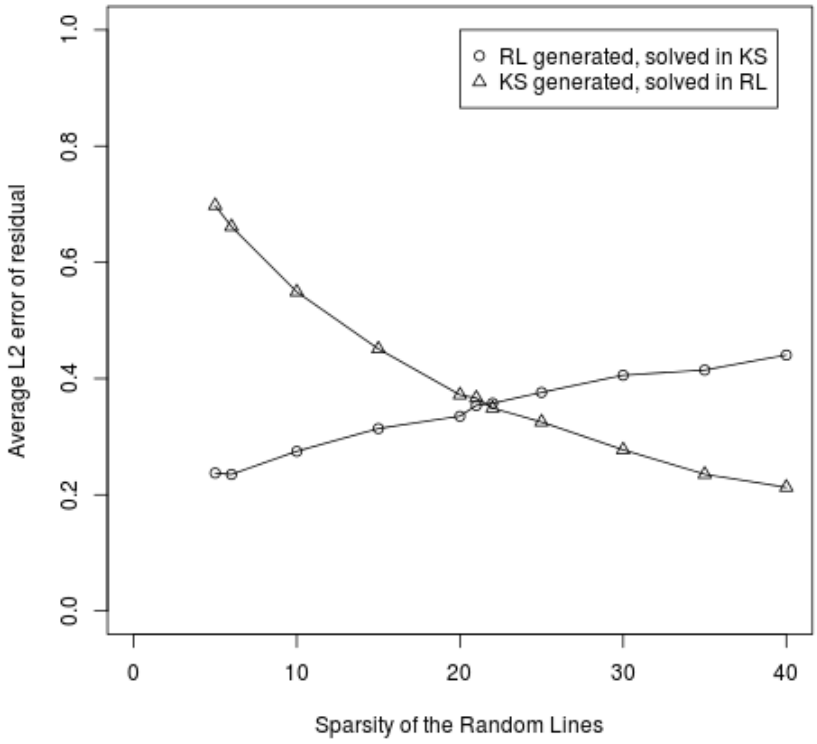

Figure 4.3: Mutual Incoherence Test RL vs KS: Triangles represent the average L2 residual error in 1000 problems generated in the K-SVD dictionary, but solved in the Random Lines Dictionary. Circles represent the average L2 residual error in 1000 problems generated in the Random Lines dictionary, but solved in the K-SVD dictionary

4.3 Feature Space Dictionaries L_2 error from Batch-OMP recovery

On all of the following summary charts, we examine the recovery error for various settings of n and s and compare it to the information theory requirement that $m \geq O(s \log_2 n)$. Recall that $m = 192$ for all experiments, and experiments use the background model of $\sigma = 50kg/m^3$. In Table 4.2, NT signifies test problems with no tunnel in them, and T signifies the standard test problem with a tunnel 32 meters long, 1.5 meters tall, 2 meters high, centered at $o_x = 0$, with a depth of 10 meters.

			VX		VW		RL		KS		SP		GS	
n	s	$s \log_2 n$	NT	T	NT	T	NT	T	NT	T	NT	T	NT	T
1024	20	200	329	337	305	325	303	305	192	205	225	251	225	251
1024	15	150	365	377	344	372	340	344	238	259	269	304	269	304
1024	10	100	415	431	398	441	389	396	306	341	334	380	334	380
1024	5	50	490	519	482	549	467	481	411	467	439	504	439	504
2048	20	220	322	331	302	323	275	278	174	187	205	228	205	228
2048	15	165	359	371	340	369	313	318	220	239	249	279	249	279
2048	10	110	410	426	393	434	366	374	286	320	313	354	313	354
2048	5	55	486	516	477	542	451	466	398	452	421	481	421	481
4096	20	240	312	322	269	296	256	259	161	173	189	210	189	210
4096	15	180	350	363	312	345	296	301	205	225	231	259	231	259
4096	10	120	402	419	370	415	351	359	272	304	295	333	295	333
4096	5	60	481	509	462	530	440	455	387	439	404	462	404	462
8192	20	260	300	309	241	281	239	243	151	164	175	194	175	194
8192	15	195	338	349	290	341	280	285	194	215	216	241	216	241
8192	10	130	391	404	356	423	337	345	263	293	278	314	278	314
8192	5	65	472	495	455	546	429	444	380	433	389	444	389	444

Table 4.2: Dictionary Test Results. The numbers represent the average L2 residual error in 1000 test problems without a tunnel (NT) and with a tunnel (T). The two factors that are of interest is reducing the L2 error as much as possible, but retaining a difference in L2 error between the NT and T problems

4.3.1 Voxel Dictionary

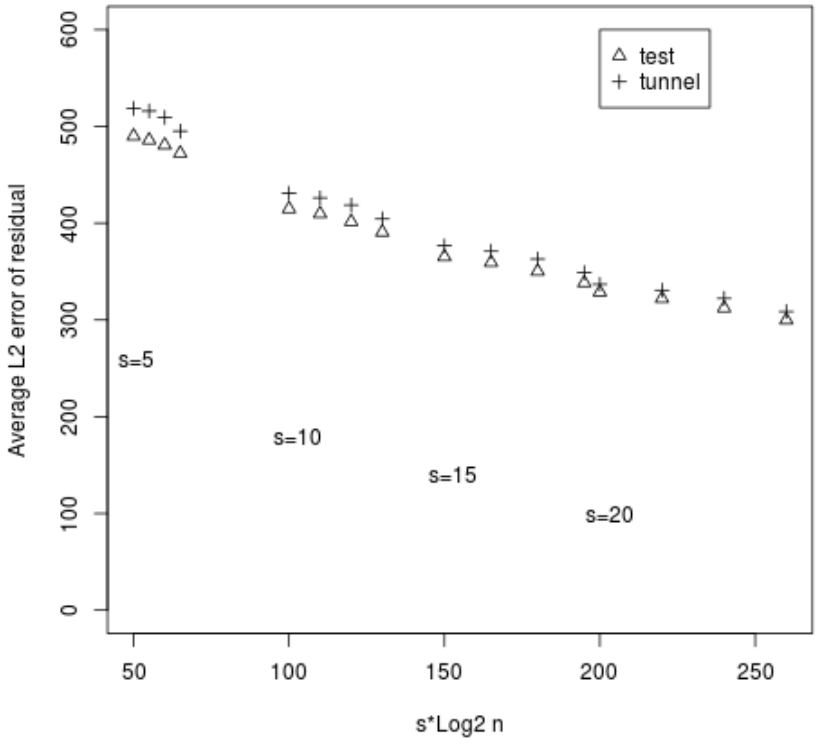

Figure 4.4: L2 results: Voxels Dictionary

This basis appears to have very little incoherence with respect to tunnel problems vs. non-tunnel problems. This makes this basis a better candidate for an anomaly basis, but it still may not be able to sparsely represent tunnels, as a large amount of voxels may still be required.

4.3.2 Wavelet-sparsified Voxel Dictionary

While this transform may buy us some sparsity, it is important to note that it will not gain us any further adherence to the Restricted Isometry Principle than we already had.

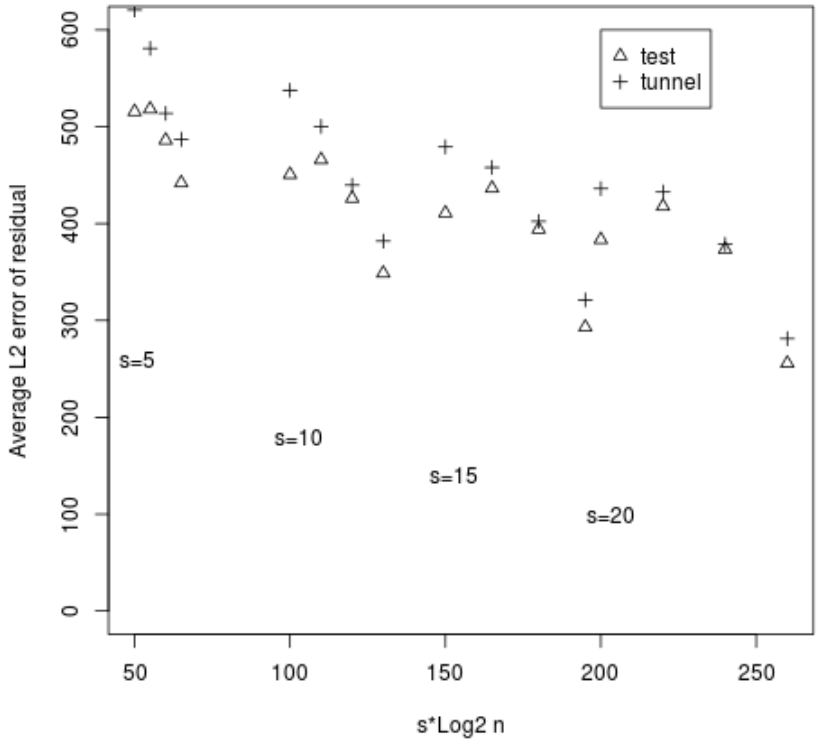

Figure 4.5: L2 results: Voxels to Wavelet Dictionary (DB4)

This transformation may have bought us some incoherence with respect to tunnel problems, so this basis is now a candidate for filtering the background when paired with a good anomaly dictionary.

4.3.3 Linefield Dictionary

The results of the Linefield density on our test problems are as follows:

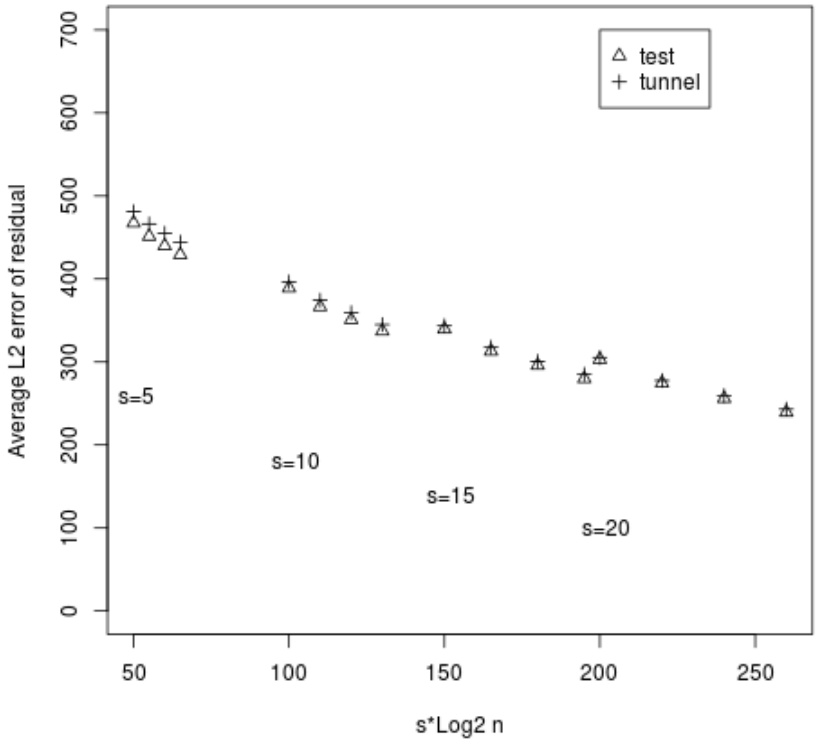

Figure 4.6: L2 results: Random Lines Dictionary

This dictionary, like the voxels has very little incoherence with respect to the tunnel problems. However, if we paired it with a better background dictionary, it may be useful. A line also more discretely represents a tunnel than voxels.

4.4 Measurement Space Dictionaries L_2 error from Batch-OMP recovery

4.4.1 K-SVD Learned Dictionary

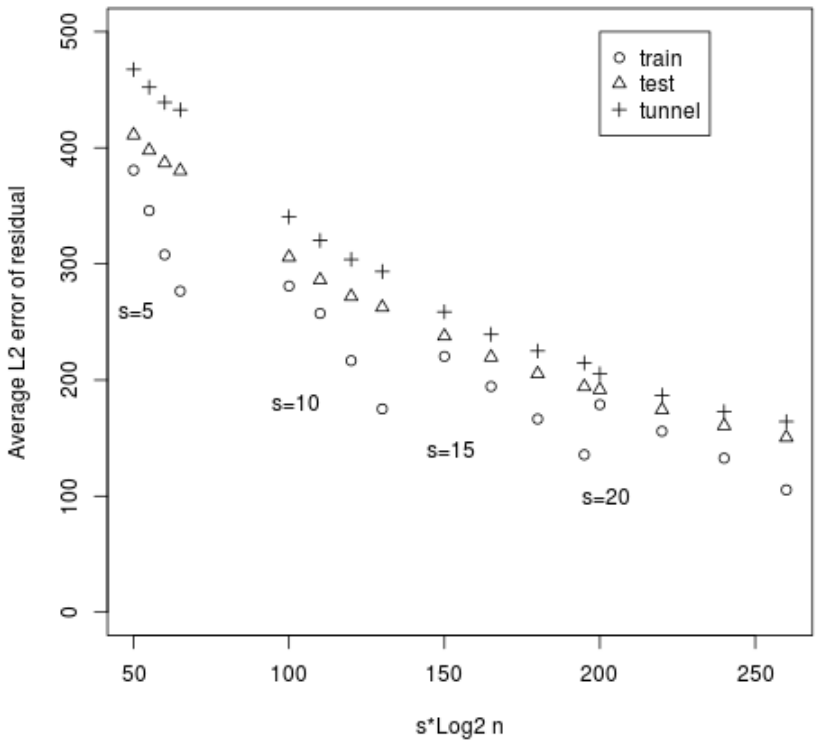

Figure 4.7: L2 results: K-SVD

We also want to examine the convergence behavior of the K-SVD algorithm.

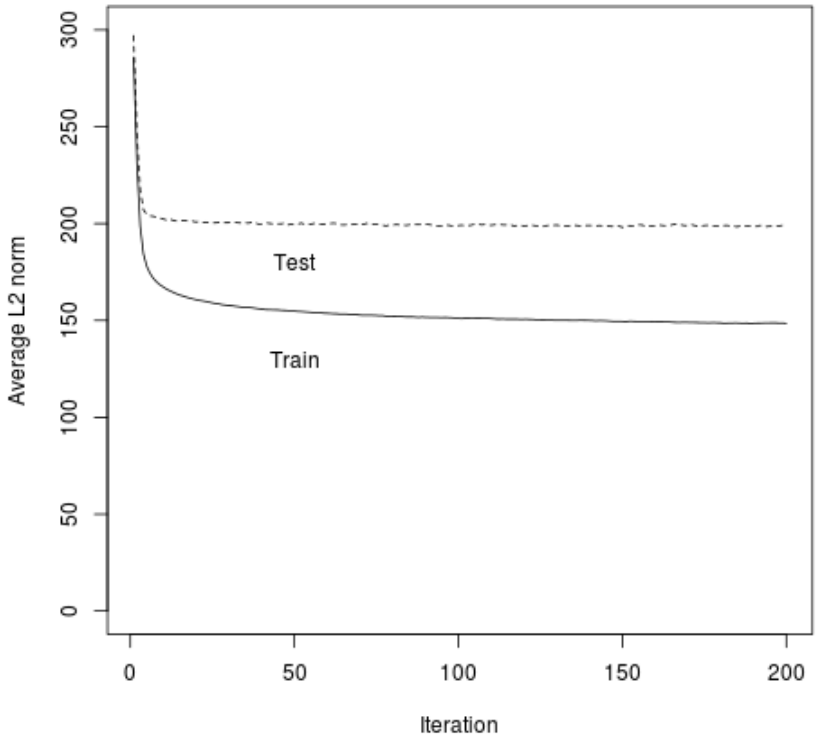

Figure 4.8: K-SVD algorithm convergence, iteration vs. L2 error: The training set of 10,000 problems is used inside of the algorithm and compared against a validation set of 1,000 problems at every iteration

We are probably converged as best as we can possibly be in terms of validation error by 100 iterations. Since K-SVD is susceptible to local optimal, the original paper suggestions two strategies for tweaking the final result [Aharon et al., 2006]. First, we can replace utilized dictionary elements with the least represented training problems, and second we can replace similar dictionary elements with the least represented training problems. These small fixes may gain us some incoherence in the dictionary.

4.4.2 Sample Problem Dictionary

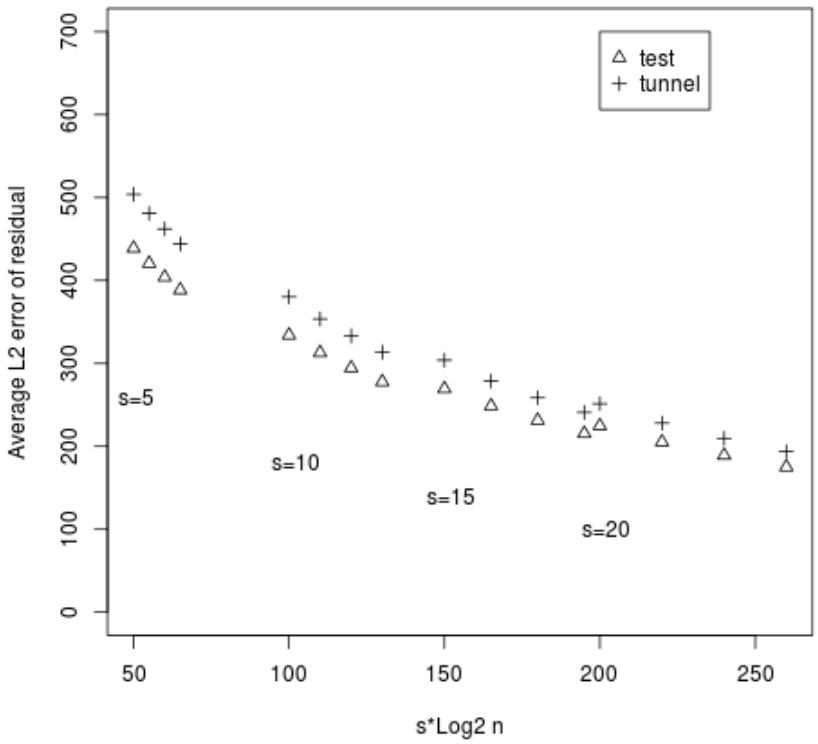

Figure 4.9: L2 results: Example Problems

The benefit of this dictionary is that it doesn't require a training algorithm, however; it still requires a large quantity of training data which may not be available. These results also demonstrate to us just how much training buys us in terms of diminishing L2 error.

4.4.3 Gaussian Dictionary

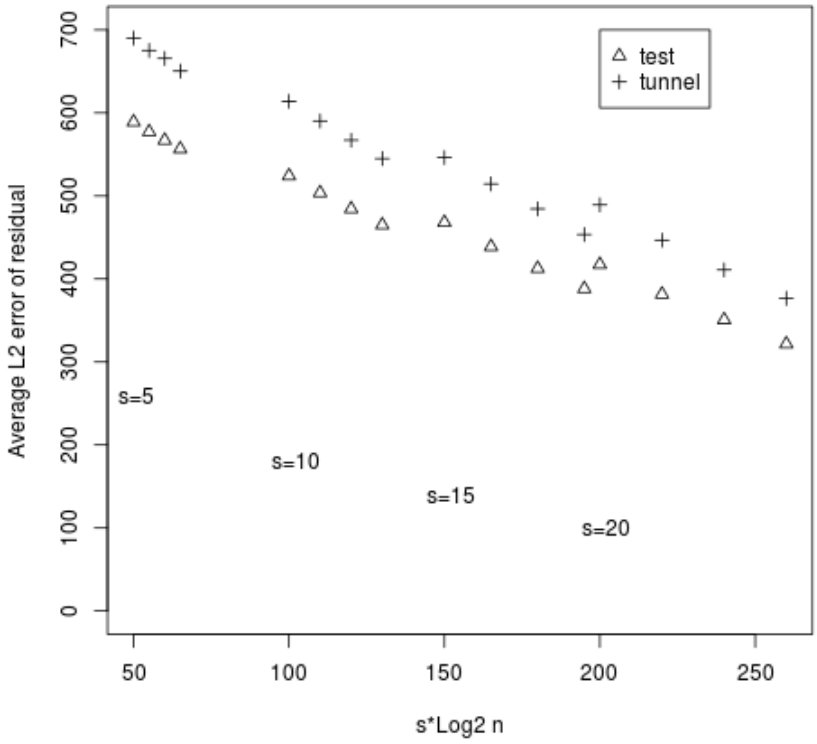

Figure 4.10: L2 results: Gaussian

4.5 The Incoherence Properties of Our Dictionaries

4.5.1 Maximum and Average Coherence

These are the results for the worst case (μ_F) and average (ν_F) measures of coherence using the established methods [Bajwa et al., 2012]. The computations below are according to the definitions in Section 2.2.

	Dictionary Size			
	1024	2048	4096	8192
Gaussian	0.393	0.388	0.361	0.384
K-SVD	0.602	0.663	0.661	0.666
Linefield	0.999	0.998	0.999	1.000
Voxels	0.997	0.999	1.000	1.000
Voxels to Wavelet	1.000	1.000	1.000	1.000
SCP threshold	0.100	0.091	0.083	0.077

Table 4.3: Max Coherence of Dictionaries

	Dictionary Size			
	1024	2048	4096	8192
Gaussian	0.246	0.258	0.271	0.281
K-SVD	0.398	0.431	0.441	0.443
Linefield	0.939	0.952	0.965	0.974
Voxels	0.963	0.988	0.989	0.989
Voxels to Wavelet	0.977	0.987	0.876	0.861
SCP threshold	0.007	0.007	0.006	0.006

Table 4.4: Average Coherence of Dictionaries

We can satisfy a Strong Coherence Property (SCP) [Bajwa et al., 2012] if the following conditions are met for an m by n dictionary

$$\mu_F \leq O\left(\frac{1}{\log n}\right),$$

$$\nu_F \leq \frac{\mu_F}{\sqrt{m}},$$

and it is apparent that none of these dictionaries will meet it.

We may be able to improve max coherence through selective atom deletion, and we may be able to improve average coherence through an atom sign flipping algorithm [Bajwa et al., 2012].

4.5.2 Estimated weak RIP

Because μ_F and ν_F do not imply coherence, we estimate it empirically by simulating from random sets of s sparse vectors, and computing the value at which RIP holds at equality. The following charts represent the .1, .5, and .9 quantiles of those simulations.

Anomaly Dictionaries

Both anomaly dictionaries don't appear to have much of a chance at weak-RIP adherence except when s is small, as a value of $\delta_s < 1$ is required at a minimum. This does justify our choice in the Random Lines dictionary (RL), as it can represent a tunnel more sparsely than voxels.

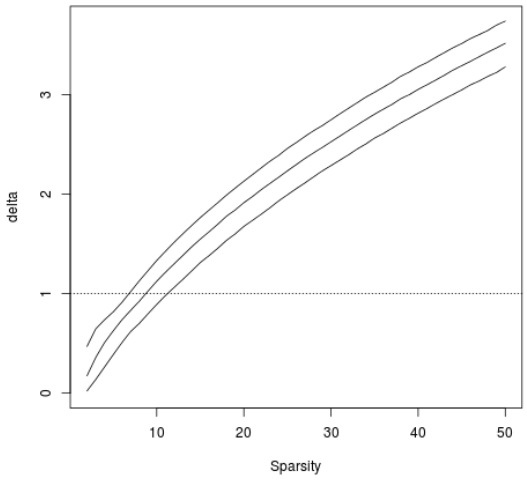

Figure 4.11: .1,.5,.9 quantiles of δ_s (isometry constant) vs sparsity, VX

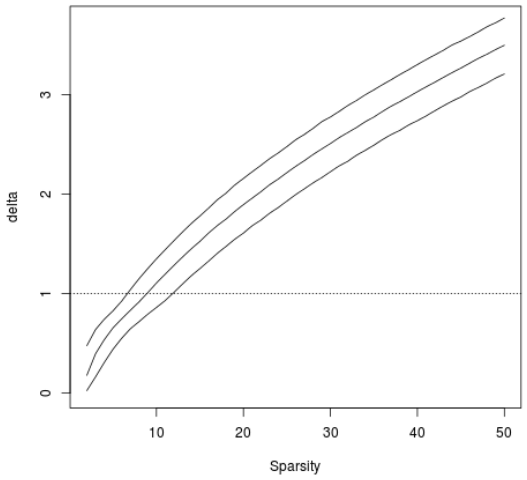

Figure 4.12: .1,.5,.9 quantiles of δ_s (isometry constant) vs sparsity, RL

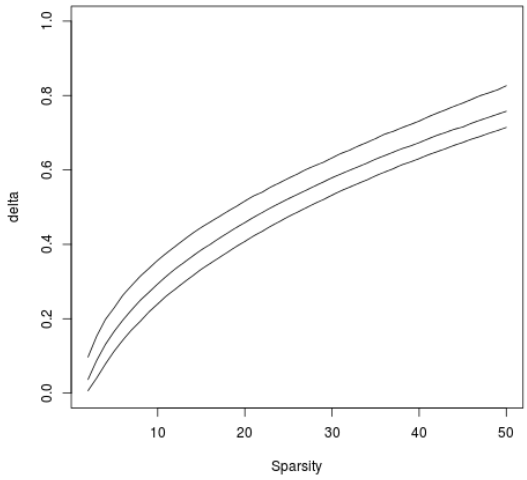

Figure 4.13: .1,.5,.9 quantiles of δ_s (isometry constant) vs sparsity, KS

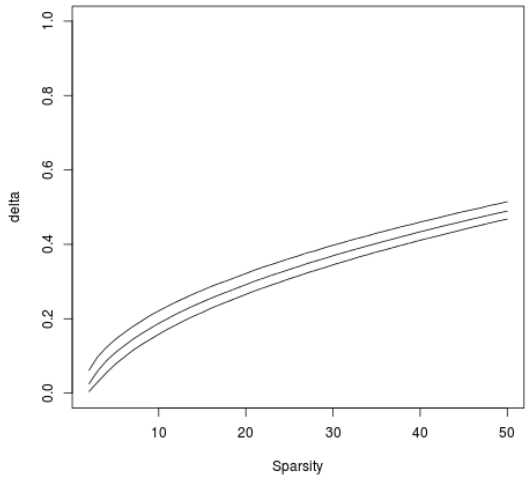

Figure 4.14: .1,.5,.9 quantiles of δ_s (isometry constant) vs sparsity, GS

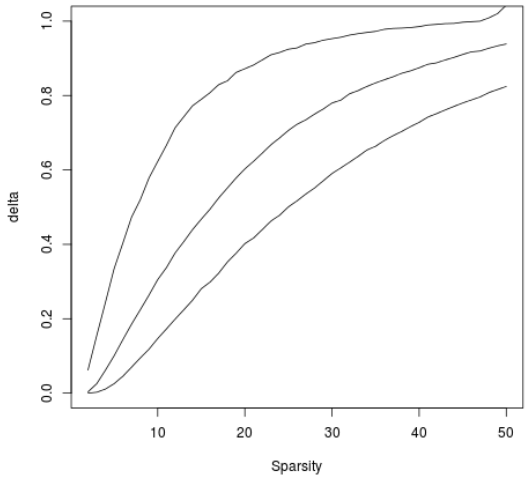

Figure 4.15: .1,.5,.9 quantiles of δ_s (isometry constant) vs sparsity, VW

As expected, the Gaussian dictionary (GS) appears to have the best adherence to weak-RIP for many test values of s, however, the performance of K-SVD (KS) and the voxels to wavelet (VW) is encouraging. The fact that a transformation may have bought us some weak-RIP adherence is good. The sample problem dictionary's weak-RIP properties were omitted.

4.5.3 Statistical Orthogonality (StOrth)

We shall also examine the statistical orthogonality properties of our dictionaries found via simulation as defined in Section 2.2. In these examples, we used dictionaries of size $n = 8192$, and used 1000 random samples of random vectors and random column subsets and examined the probability quantiles of ϵ . The sample problem dictionary's StOrth properties have been omitted. In these charts, we hope to see small values of ϵ for high probability quantiles. We prefer values less than 1/3 or 0.333.

		Probability quantile								
		0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
sparsity	5	0.43	0.46	0.49	0.53	0.57	0.61	0.68	0.74	0.82
	10	0.42	0.46	0.49	0.53	0.56	0.59	0.63	0.68	0.75
	15	0.43	0.46	0.49	0.52	0.56	0.59	0.62	0.67	0.73
	20	0.44	0.48	0.51	0.53	0.56	0.59	0.63	0.67	0.75
	25	0.44	0.48	0.51	0.54	0.57	0.60	0.64	0.67	0.74
	30	0.45	0.49	0.52	0.55	0.58	0.61	0.65	0.70	0.76

Table 4.5: VX Statistical Orthogonality, ϵ values

Our Voxel Dictionary (VX) dictionary's values are still pretty high in most cases, so this property is probably not present. Even for a modest probability of 0.5, we don't see a single value above 0.333 for any sparsity.

		Probability quantile								
		0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
sparsity	5	0.44	0.45	0.47	0.50	0.54	0.59	0.65	0.70	0.76
	10	0.39	0.43	0.46	0.49	0.52	0.54	0.58	0.62	0.70
	15	0.38	0.41	0.44	0.47	0.49	0.52	0.55	0.59	0.65
	20	0.37	0.41	0.43	0.46	0.48	0.51	0.54	0.58	0.63
	25	0.37	0.40	0.43	0.45	0.48	0.50	0.53	0.57	0.63
	30	0.36	0.39	0.42	0.44	0.47	0.49	0.53	0.57	0.62

Table 4.6: VW Statistical Orthogonality, ϵ values

The Wavelet transform for WV dictionary may have bought a small amount of adherence to StOrth, but not much.

		Probability quantile								
		0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
sparsity	5	1.00	1.12	1.21	1.30	1.37	1.43	1.50	1.59	1.70
	10	1.47	1.59	1.68	1.75	1.84	1.91	1.99	2.07	2.19
	15	1.84	1.98	2.07	2.14	2.20	2.28	2.36	2.44	2.57
	20	2.18	2.28	2.37	2.45	2.51	2.58	2.66	2.73	2.86
	25	2.42	2.55	2.63	2.71	2.78	2.85	2.91	3.01	3.11
	30	2.71	2.83	2.91	2.99	3.07	3.14	3.21	3.29	3.40

Table 4.7: RL Statistical Orthogonality, ϵ values

We have little hope of having StOrth with the Random Lines Dictionary (RL), as there are usually very similar lines in the complement of any s subset permutation.

		Probability quantile								
		0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
sparsity	5	0.00	0.02	0.03	0.05	0.06	0.07	0.08	0.10	0.12
	10	0.04	0.05	0.07	0.08	0.09	0.10	0.12	0.13	0.15
	15	0.06	0.08	0.09	0.10	0.11	0.12	0.13	0.14	0.16
	20	0.07	0.09	0.10	0.11	0.12	0.13	0.14	0.15	0.17
	25	0.08	0.09	0.10	0.11	0.12	0.14	0.15	0.16	0.18
	30	0.09	0.10	0.11	0.12	0.13	0.14	0.15	0.16	0.18

Table 4.8: GS Statistical Orthogonality, ϵ values

As expected, our Gaussian Dictionary (GS) does well.

		Probability quantile								
		0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
sparsity	5	0.01	0.03	0.04	0.06	0.08	0.09	0.11	0.13	0.16
	10	0.05	0.07	0.09	0.10	0.12	0.13	0.14	0.16	0.19
	15	0.07	0.10	0.11	0.12	0.13	0.15	0.16	0.18	0.21
	20	0.09	0.11	0.13	0.14	0.15	0.16	0.17	0.19	0.21
	25	0.10	0.12	0.14	0.15	0.16	0.18	0.19	0.21	0.24
	30	0.11	0.13	0.14	0.15	0.17	0.18	0.19	0.21	0.23

Table 4.9: KS Statistical Orthogonality, ϵ values

The K-SVD Dictionary (KS) also does quite well, as we see a high probability of $\epsilon < 0.333$, making it a reasonable choice. While the probability of $\epsilon > 0.333$ is larger with KS than GS, it is still less than 0.9.

4.6 Preliminary Conclusion

The combination of K-SVD Dictionary (KS) and Random Lines (RL) is probably our best chance at obeying the requirements of Compressed Sensing. Those are the two dictionaries that we will use going forward.

Chapter 5

OPTIMIZATION FORMULATIONS AND EXPERIMENTS

5.1 Combining the Dictionaries Using a Single Program

Our strategy is to come up with a methodology to combine two into dictionaries into one solver and decompose the signal into the two parts we believe it was generated from, as described in Section 3.2. We will use an anomaly dictionary C_A, and a background dictionary C_B, and the following program (where $n = n_A + n_B$):

$$\begin{gathered} \min \|\theta_A\|_1 + \lambda \|\theta_B\|_1 , \\ \text{st} \\ \left\| \begin{bmatrix} C_A & C_B \end{bmatrix} \begin{bmatrix} \theta_A \\ \theta_B \end{bmatrix} - b \right\|_2 \leq \varepsilon_2 = (\sqrt{2 \log_2 n})\sigma. \end{gathered} \tag{5.1}$$

Now here, the σ should only represent measurement error, if our background dictionary and anomaly dictionary are correct. However, we are fairly certain that we will not be able to find a perfectly sparse solution for the background. The large benefit of this formulation is that λ is the only parameter that is truly free.

From this point on, our experiments will be confined to the Random Lines dictionary for the anomaly, and the K-SVD dictionary for the background. We have a theoretical value for the right-hand side of the constraint ε_2, but we shall also test other several values [Candes and Tao, 2005]. We will attempt to find λ to be in the range we see in our test problems when the values of θ_A has a sparsity s_A low enough to obey weak-RIP or Statistical Orthogonality with a reasonably high (.9) probability. We will also be examining the ranges of ε_2 and λ where we have the best chance of truly detecting a tunnel when one exists, and not detecting one when it doesn't exist.

The main experiments we shall use will have $\sigma = 50kg/m^3$ in the data, and the Random Lines Dictionary with $n_A = 8192$ lines, and the K-SVD dictionary with $n_B = 8192$ columns.

5.2 Detection Experiments

For our optimization problem, we will consider a problem where there is at least one non-zero random lines coefficient as being a positive detection of a tunnel, and $\hat{\theta}_A = \mathbf{0}$ being a negative result. Table 5.1 shows our results on 20 no tunnel test problems and 20 tunnel test problems. When there is a no tunnel, but we get a positive detection, we shall call that a "false positive", and when there is a tunnel and we do not get a positive detection, we shall call that a "false negative".

	ε_2 No Tunnel					ε_2 Tunnel				
λ	200	250	300	400	500	200	250	300	400	500
0.037	1	1	1	1	1	14	15	15	14	12
0.041	2	1	2	2	3	17	17	17	17	17
0.045	4	4	4	4	4	20	20	20	19	17
0.050	4	4	4	4	4	20	20	20	19	17
0.135	20	20	20	20	18	20	20	20	20	20
0.368	20	20	20	20	18	20	20	20	20	20

Table 5.1: Number of solutions with at least one non-zero RL coefficient, $\varepsilon_2 = 250$ is near the correct theoretical value

Since we are aware that we have a background model of $\sigma = 50kg/m^3$, examining the theoretical value for $\varepsilon_2 = (\sqrt{2\log_2 n})\sigma$, using a dictionary of size 8192, we would expect $\varepsilon_2 = 254.9$. We will look at $\varepsilon_2 = 250$ as our theoretical value, and experimentally, we will choose λ anywhere between 0.045 and 0.05, where is gives us a range where $theta_{RL}$ is has at least one non-zero element 20/20 times when there is a tunnel in the data, and $theta_{RL}$ has the least number of times where there is at least one non-zero element and there is actually no tunnel in the data (4/20).

Examining that solution a little further, we will look at the sparsity in the Random Lines dictionary.

	non-zero θ_{RL}		Avg θ_{RL}		Max θ_{RL}	
Test Problem	NT	T	NT	T	NT	T
1	1	3	1.08	4.07	1.08	11.10
2	0	1	0	5.80	0	5.80
3	0	1	0	0.93	0	0.93
4	0	1	0	7.58	0	7.58
5	3	4	1.91	3.79	3.26	5.53
6	1	3	1.53	4.09	1.53	6.16
7	1	3	1.53	2.78	1.53	4.09
8	0	1	0	10.36	0	10.36
9	0	1	0	3.78	0	3.78
10	0	3	0	2.14	0	2.84
11	0	1	0	6.17	0	6.17
12	0	2	0	5.36	0	8.95
13	0	1	0	6.23	0	6.23
14	0	3	0	3.68	0	5.00
15	0	4	0	2.29	0	4.24
16	0	1	0	1.02	0	1.02
17	0	3	0	2.94	0	3.79
18	0	2	0	5.15	0	8.85
19	0	1	0	1.01	0	1.01
20	0	2	0	5.81	0	11.07
mean			1.51	4.25	1.85	5.72
stdev			0.34	2.38	0.96	3.18

Table 5.2: Summary of Random Lines coefficients on the test problems: For each of the 20 test problems, we recorded the number of non-zero values in θ_{RL}, the average value of a θ_{RL} value, if non-zero, and the max θ_{RL} value

Unfortunately, it doesn't look like we can reject any of the false-positives based on their coefficient values alone. However, our experiments do support our theoretical value for ε_2. Our best value of λ also gives very sparse vectors (4 or less), which supports the experimental results in Chapter 4 that showed for StOrth or weak-RIP to hold for the Random Lines dictionary, the sparsity should be 5 or lower.

5.3 Localization Experiments

In order to come up with a spatial representation of the anomaly based on the random lines solution, we created a set voxels in the measurement space, and then computed that each active lines' contribution to each voxel, and summed all lines' contributions to all voxels. This generates the following plots for 4 of our tunnel problems (y-axis labelling is omitted out of concern for space, but the y-range is -2 to 2, some distortion exists, so the angle deviation is not as wide as it appears in these figures):

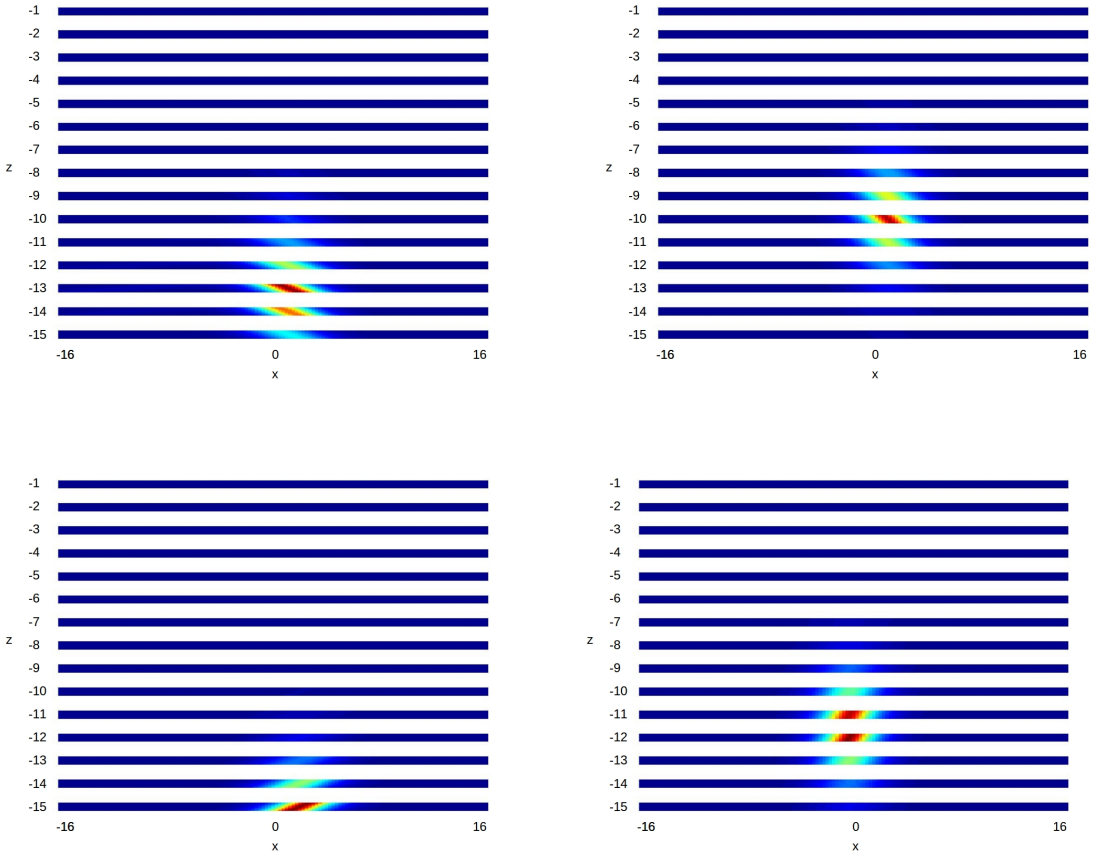

Figure 5.1: Location and depth of estimated tunnel (true tunnel at x=0, x = -10), red represents negative density with respect to average ground density.

We can also estimate the location of the minimum density in both x and z directions. The following table shows our results for 20 test problems and 4 different tunnel locations.

	x=0,z=-10		x=0,z=-5		x=10,z=-10		x=10,z=-5	
Test Problem	x	z	x	z	x	z	x	z
1	1.25	-12.5	0.5	-6.5	9.25	-14.5	9.5	-8.5
2	1	-9.5	0.5	-7.5	10.5	13.5	9.75	-7.5
3	2	-14.5	0.5	-7.5	10.5	-10.5	9.5	-8.5
4	-0.25	11.5	0.5	-7.5	10.5	-10.5	9.5	-8.5
5	-1.5	-12.5	0	-6.5	7.25	-9.5	9.75	-9.5
6	0.25	-12.5	0.5	-7.5	9.25	-14.5	10	-9.5
7	0.75	-13.5	0.5	-7.5	13	-14.5	9.75	-8.5
8	-1.25	-10.5	0.25	-7.5	12	-11.5	9.75	-7.5
9	0	-12.5	0.5	-7.5	9.25	-14.5	9.75	-8.5
10	0.25	-12.5	0.5	-7.5	9.75	-12.5	9.75	-8.5
11	0	-12.5	0.5	-7.5	9.5	-12.5	9.75	-8.5
12	-1.25	-10.5	0.25	-7.5	11.25	-9.5	10.25	-8.5
13	0	-12.5	0.5	-7.5	9.75	-12.5	9.75	-8.5
14	0.25	-12.5	0.25	-7.5	10	-14.5	9.5	-8.5
15	0.25	-12.5	0.5	-7.5	10.5	-11.5	9.75	-7.5
16	2.5	-9.5	0.5	-7.5	9.75	12.5	9.5	-8.5
17	0.25	-12.5	0.5	-7.5	9.5	-8.5	9.5	-8.5
18	0	-12.5	0.5	-7.5	9.25	-12.5	9.5	-7.5
19	2.5	-12.5	0.5	-7.5	10	-14.5	9.5	-8.5
20	-1.25	-10.5	-0.25	-7.5	14.5	-14.5	9.75	-8.5
Average	0.29	-12	0.4	-7.4	9.54	-12.45	9.69	-8.4
St Dev	1.15	1.28	0.21	0.31	4.18	1.96	0.20	0.55

Table 5.3: Estimated Minimum Density locations on 20 test problems: Based on the strength of the θ_{RL}, the minimum density locations were computed for 20 test problems and 4 scenarios. A depth bias is apparent, as almost all the solutions seem to estimate depth lower than it is.

5.4 Setting λ

We chose to examine two values of λ with more experiments to show the trade-off that occurs in the following hypothesis test:

$$H_0 : \text{No Tunnel}, H_a : \text{Tunnel}$$

λ	Type I error	Type II error
0.041	0.13	0.135
0.05	0.29	0.01

Table 5.4: Estimated Type I and Type II error probabilities based on 200 replicates

The localization of the tunnel in both settings of λ are shown in the following plots.

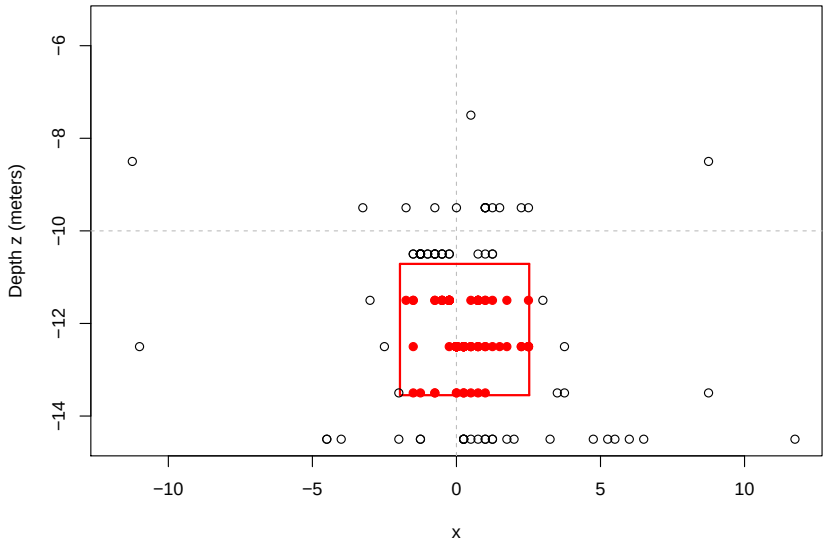

Figure 5.2: Estimated location of the minimum density in the x=0, z=-10 problem for $\lambda = 0.05$. The red 122 of 200 replicates are within the 1 stdev of the estimated mean box.

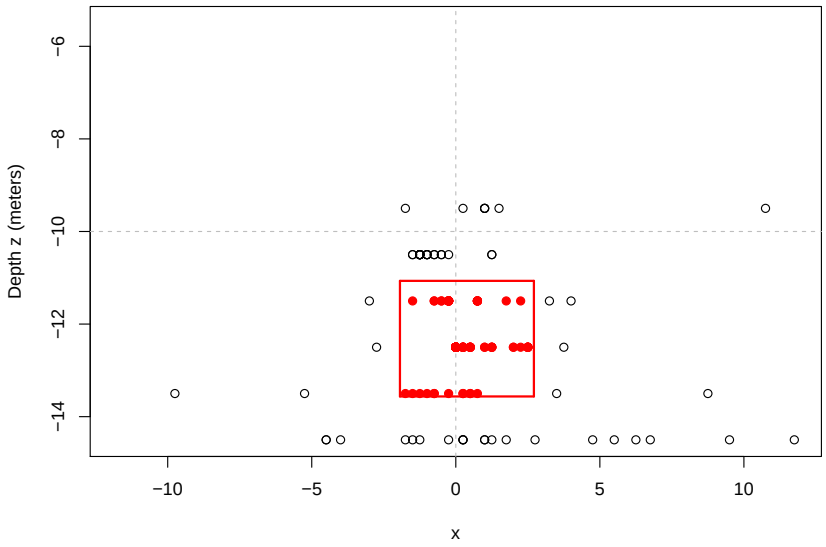

Figure 5.3: Estimated location of the minimum density in the x=0, z=-10 problem for $\lambda = 0.041$. The red 115 of 200 replicates are within the 1 stdev of the estimated mean box.

While we see a great deal of bias in the z-coordinate, our ability to estimate the x-coordinate appears to be relatively consistent.

Chapter 6

CONCLUSIONS

6.1 Summary of work

In this work, we examined several different types of feature-space and measurement-space dictionaries in order to find the best way to both filter background and detect a tunnel-shaped anomaly in a set of surface gravimetric gradient measurements. The compressed sensing program was our main solution method, and we were able to show that true tunnels can be detected in simulated data sets.

6.2 Recommended follow on work

The problem of 4 of 20 false positives still persists. Several methods were attempted to remove them from the data, including a sequential Monte Carlo approach, and other ways to re-sample the random lines. One area that could be profitable would be to solve the same problem with several different types of background dictionaries and see if there is some degree of agreement across all types of background filters.

Another challenge presents itself when we have more than one tunnel in the data, or a complex tunnel structure. Our choice of λ is usually informed by the sparsity required for weak-RIP of Stochastic Orthogonality to hold. However, with two tunnels that are sufficiently far apart, a greater sparsity may be required. This creates challenges in setting λ that need to be further analyzed.

It would also be interesting to utilize this method on an extra-terrestrial object, like an asteroid. If the sensors are available, we would have the ability to scan in all 3 dimensions. This may give us additional information and improve the accuracy of our predictions of the sub-surface density.

BIBLIOGRAPHY

J. Adler, B.D. Rao, and K. Kreutz-Delgado. Comparison of basis selection methods. In *Signals, Systems and Computers, 1996. Conference Record of the Thirtieth Asilomar Conference on*, volume 1, pages 252 –257 vol.1, 1996.

M. Aharon, M. Elad, and A. Bruckstein. On the uniqueness of overcomplete dictionaries, and a practical way to retrieve them. *Linear Algebra and its Applications*, 416:48–67, 2006.

W. Bajwa, R. Calderbank, and D. Mixon. Two are better than one: Fundamental parameters of frame coherence. *Applied and Computational Harmonic Analysis*, 33(1):58–78, 2012.

Waheed U Bajwa, Robert Calderbank, and Sina Jafarpour. Model selection: Two fundamental measures of coherence and their algorithmic significance. In *Information Theory Proceedings (ISIT), 2010 IEEE International Symposium on*, pages 1568–1572. IEEE, 2010.

A Bandeira, E. Dobriban, D. Mixon, and W. Sawin. Certifying the restricted isometry property is hard. *IEEE Transactions on Information Theory*, 59:3448–3450, 2012.

S. R. Becker, E. J. Candès, and M. C. Grant. Templates for convex cone problems with applications to sparse signal recovery. *Mathematical Programming Computation*, 3(3): 165–218, 2011.

E. Candes and T. Tao. The Dantzig selector: Statistical estimation when p is much larger than n. *Annals of Statistics*, 35(6):2313–2351, 2005.

E. Candes and T. Tao. Near-optimal signal recovery from random projections: Universal encoding strategies? *Information Theory, IEEE Transactions on*, 52(12):5406–5425, 2006.

E. Candes and M. Wakin. An introduction to compressive sampling. *Signal Processing Magazine, IEEE*, 25(2):21 –30, 2008.

Emmanuel J Candes and Terence Tao. Decoding by linear programming. *Information Theory, IEEE Transactions on*, 51(12):4203–4215, 2005.

Emmanuel J Candes, Michael B Wakin, and Stephen P Boyd. Enhancing sparsity by reweighted 1 minimization. *Journal of Fourier analysis and applications*, 14(5-6):877–905, 2008.

J. Candes, E, Y. C. Eldar, D. Needell, and P. Randall. Compressed sensing with coherent and redundant dictionaries. *Applied and Computational Harmonic Analysis*, 31(1):59–73, 2011.

Guangliang Chen, Atul Divekar, and Deanna Needell. Guaranteed sparse signal recovery with highly coherent sensing matrices. *arXiv preprint arXiv:1311.0314*, 2013.

S. Chen and D. Donoho. Basis pursuit. In *Signals, Systems and Computers, 1994. 1994 Conference Record of the Twenty-Eighth Asilomar Conference on*, volume 1, pages 41 –44 vol.1, 1994.

Ingrid Daubechies et al. *Ten Lectures on Wavelets*, volume 61. SIAM, 1992.

D. Donoho. Compressed sensing. *IEEE Trans. Inform. Theory*, 52(4):1289–1306, 2006.

D Donoho and J Tanner. Observed universality of phase transitions in high-dimensional geometry, with implications for modern data analysis and signal processing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 367(1906):4273–4293, 2009.

S Gerschgorin. On bounding the eigenvalues of a matrix. *Izv. Akad. Nauk. SSSR Otd Mat. Estest*, 1:749–754, 1931.

L. Granai and P. Vandergheynst. Sparse decomposition over multi-component redundant dictionaries. In *Multimedia Signal Processing, 2004 IEEE 6th Workshop on*, pages 494 – 497, 2004.

V. Isakov. *Inverse Source Problems*. American Mathematical Society, 1990.

M. Krasovec and F. D. Morgan. Gravity in cave exploration. In *Proceedings of the Symposium on the Application of Geophysics to Engineering and Environmental Problems*, pages 73–280. Environmental and Engineering Geophysical Society, 1999.

S. G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *Signal Processing, IEEE Transactions on*, 41(12):3397 –3415, 1993.

T. M. Meglich, M. C. Williams, and K. Hanna. *Subsurface Imaging of Lava Tubes: Roadway Applications*. US Department of Transportation, Federal Highway Administration, Central Federal Lands Highway Division, 2005.

M. Meila, C. Marzban, and U. Yurtsever. Gravimetric detection by compressed sensing. *Geoscience and Remote Sensing Symposium, 2008. IGARSS 2008. IEEE International*, 2:II–193, 2008.

J. Munk and R. A. Sheets. Detection of underground voids in Ohio by use of geophysical methods. Technical report, 1997.

J.F. Murray and K. Kreutz-Delgado. An improved focuss-based learning algorithm for solving sparse linear inverse problems. In *Signals, Systems and Computers, 2001. Conference Record of the Thirty-Fifth Asilomar Conference on*, volume 1, pages 347 –351 vol.1, 2001.

B. K. Natarajan. Sparse approximate solutions to linear systems. *SIAM J. Comput.*, 24: 227–234, 1995.

Y.C. Pati, R. Rezaiifar, and P.S. Krishnaprasad. Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition. In *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*, pages 40 –44 vol.1, 1993.

B.D. Rao and K. Kreutz-Delgado. An affine scaling methodology for best basis selection. *Signal Processing, IEEE Transactions on*, 47(1):187 –200, 1999.

G. Raskutti, M. J. Wainwright, and B. Yu. Restricted eigenvalue properties for correlated Gaussian designs. *The Journal of Machine Learning Research*, 99:2241–2259, 2010.

H. Rauhut. Compressive sensing and structured random matrices. *Theoretical Foundations and Numerical Methods for Sparse Recovery*, 9:1–92, 2010.

R. Rubinstein, M. Zibulevsky, and M. Elad. Efficient implementation of the K-SVD algorithm using batch orthogonal matching pursuit. Technical report, 2008.

S Sahoo and A Makur. Dictionary training for sparse representation as generalization of k-means clustering. *Signal Processing Letters, IEEE*, 20(6):587–590, 2013.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 99:267–288, 1996.

G. E. Uhlenbeck and L. S. Ornstein. On the theory of the brownian motion. *Phys. Rev.*, 36:823–841, 1930.

Yiming Yang and Jan O Pedersen. A comparative study on feature selection in text categorization. In *ICML*, volume 97, pages 412–420, 1997.

VITA

Ryan Kappedal is a Lieutenant Colonel in the United States Air Force and a PhD student at the University of Washington. He attended the United States Air Force Academy, and the Air Force Institute of Technology where his thesis was entitled Intelligence Surveillance and Reconnaissance Asset Assignment for Optimal Mission Effectiveness.