

Duquesne University

Duquesne Scholarship Collection

Electronic Theses and Dissertations

Fall 2010

Compressive Sensing

Joshua Booth

Follow this and additional works at: <https://dsc.duq.edu/etd>

Recommended Citation

Booth, J. (2010). Compressive Sensing (Master's thesis, Duquesne University). Retrieved from <https://dsc.duq.edu/etd/340>

This Immediate Access is brought to you for free and open access by Duquesne Scholarship Collection. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of Duquesne Scholarship Collection.

COMPRESSIVE SENSING

A Thesis

Submitted to McAnulty College
and Graduate School of Liberal Arts

Duquesne University

In partial fulfillment of the requirements for
the degree of Masters of Science

By

Joshua Dennis Booth

August 2010

COMPRESSIVE SENSING

By

Joshua Dennis Booth

Approved August 17, 2010

Carl Toews, Ph.D.
Associate Professor of Mathematics
(Dissertation Director)

Donald Simon, Ph.D.
Associate Professor of Computer Science
(Director of Graduate Study)

Karl Wimmer, Ph.D.
Associate Professor of Mathematics
(Committee Member)

Jeffrey Jackson, Ph.D.
Professor of Computer Science
Department Chair

Christopher M. Duncan, Ph.D.
Dean
McAnulty College and Graduate School
of Liberal Arts

ABSTRACT

COMPRESSIVE SENSING

By

Joshua Dennis Booth

August 2010

Thesis Supervised by Dr. Carl Toews

This work is an expository overview of certain key elements in the area of compressive sensing. As a sub-discipline of signal processing, compressive sensing is concerned with both sampling and reconstruction techniques. In this expository, sampling will center on random matrices and expander graphs, while reconstruction will use multiple numerical optimization techniques. Although theoretical performance bounds for these techniques can be found scattered throughout the published literature, there are few practical rules for concrete problems. This thesis helps fill this gap by experimenting on the asymptotic bounds of the number of measurements needed to guarantee perfect reconstruction. These numerical experiments help to identify specific sensing regimes in which performance begin to break down.

Acknowledgment:

I would like to gratefully acknowledge my family and advisor, Carl Toews,
for their continued support in my exploration of knowledge.

I would also like to acknowledge James Cochick whose lessons sparked the
start of this adventure.

TABLE OF CONTENTS

Abstract	iv
Acknowledgment	v
List of Tables	viii
List of Figures	ix
Preface	x
1 Introduction	1
1.1 Notation and Background	2
1.2 Past Work	3
2 Sensing Matrices	6
2.1 Geometric and Probabilistic Design	6
2.1.1 Introduction to The Restricted Isometry Property (RIP)	12
2.1.2 A Deeper Look at RIP	15
2.1.3 RIP Matrices	17
2.2 Graph Construction	19
2.2.1 Expander Graph Construction	19
2.2.2 RIP-1	21
2.2.3 Remarks	22
3 Reconstruction	23
3.1 Pursuit Methods	24
3.1.1 Orthogonal Matching Pursuit	26
3.1.2 Basis Pursuit	29
3.2 Optimization	30

3.3	Bucket Recovery	34
4	Numerical Results	36
4.1	Publication Comparison Results	37
4.1.1	Vector Results	40
4.1.2	Digital Image Results	43
4.2	Bounds	45
4.3	Discussion	50
5	Conclusion and Future work	51
	Bibliography	53

LIST OF TABLES

4.1	Results of tests using signals with original size 17^3 and K randomly placed 1s that were compressed to the size 17^2	41
4.2	Results of tests using signals with original size 17^3 and K randomly placed spikes of values in the range of ± 300 that were compressed to the size 17^2	41
4.3	Minimum sketch lengths needed for recovery.	45
4.4	Parameters of fitted curve for asymptotic bound on the minimum number of measurements needed when using a binary graph sensing matrices.	47
4.5	Parameters of fitted curve for asymptotic bound on the minimum number of measurements needed when using a Gaussian random sensing matrices.	49

LIST OF FIGURES

3.1	Visual representation of bucket recovery.	35
4.1	Sample of visual assessment for reconstruction using a signal of original length 17^3 with 35 randomly placed spikes that were compressed to a length of 17^2 using three different sensing matrices.	43
4.2	Image reconstruction that were first compressed using compressive sensing.	44
4.3	Fitted curve of the asymptotic bound for expander graph based sensing matrix recovered using linear programming.	46
4.4	Fitted curve of the asymptotic bound for expander graph based sensing matrix recovered using a graph method.	48
4.5	Fitted curve of the asymptotic bound for Gaussian random sensing matrix recovered with linear programming.	49

Preface

This thesis explores an area of mathematics called compressive sensing. The work has two aims: one is to provide an expository overview of some of the key results in the field, uniting and connecting ideas that at present are scattered throughout the research literature. The other is to investigate the practical limits of certain theorems and algorithms by comparing their performance on a test suite of model problems. Together, these goals are designed to produce a work which will serve as a useful reference for future researchers in the field, especially those who are approaching it for the first time, and will provide a body of performance benchmarks against which their results can be judged.

Chapter 1

Introduction

Compressive sensing is a recently developed subset of digital signal processing. The processes of sampling, compression, and reconstruction of digital signal processing are the principle processes composing compressive sensing. The term sampling is generally related to the reduction of a continuous signal, such as a sinusoid wave, to a discrete or digital signal, such as a binary vector [Str07]. Sampling can be done in multiple ways, one of which is simple discretization. The reduction of data that represents a discrete signal is compression. The opposite for both sampling and compression is reconstruction. In reconstruction, the goal is to successfully reproduce the original signal from a smaller set of data [Don06].

In the past, the golden rule for sampling was the Nyquist-Shannon Sampling Theorem. This theorem has been the principal sufficient condition for reconstruction for over forty years. In general terms, the theorem states that a continuous bandlimited ¹ signal must be sampled at least twice that of the frequency in order for the sample points to provide enough information for perfect reconstruction [GW08].

Compressive sensing takes a different approach than standard sampling and, by doing so, can take fewer samples than can the Nyquist-Shannon's Sampling Theorem,

¹A bandlimited signal is a signal whose Fourier Transformation is zero above a certain finite frequency.

and it still can produce a result that can be reconstructed. Compressive sensing starts with a discrete signal that it assumes is sparse, i.e., a discrete signal that has predominantly zero valued entries. It then takes a number of inner products of these discrete signals, $\mathbf{x} \in \mathbb{R}^N$, where the number of inner products, n , is less than N . The reconstruction from these n inner products to the original N dimensional signal is probabilistic. The probabilistic reconstruction means that with high probability the original signal can be reconstructed if the original signal was sparse enough and the set of vectors used to take the inner products conform to certain properties [Don06].

The purpose of this document is to amalgamate current publications and produce a more coherent introductory manual on compressive sensing. This objective will be fulfilled in the following chapters that will outline current construction methods of a matrix whose row vectors are used in the inner products, reconstruction methods, and numerical results of current application.

1.1 Notation and Background

The underlining model of compressive sensing and its inner products can be simplified to a simple linear equation in the sensing direction, i.e.,

$$\mathbf{y} = \Phi\mathbf{x} + \tau. \tag{1.1}$$

In this model, \mathbf{x} is the original signal represented by a vector with dimension N , and \mathbf{y} is the signal after a linear transformation with the sensing matrix $\Phi \in \mathbb{R}^{n \times N}$. The goal of Compressed Sensing is to have $N \gg n$ while still being able to recover \mathbf{x} . This model allows for some additive noise in transfer, τ , with dimension n . An additional constraint on the model is that \mathbf{x} exhibits some level of sparsity. Formally,

Definition 1.1.0.1. k-Sparsity

A vector $\mathbf{x} \in \mathbb{R}^N$ is k -Sparse if the cardinality of its elements $\{x_i \neq 0\}$ is less than or equal to k .

Some models exist where the signal, \mathbf{x} , is not assumed to be sparse [BDB07]. These models are best represented with two matrices Φ and Ψ , where $\Psi \in \mathbb{R}^{N \times N}$ and

$$\mathbf{y} = \Phi\Psi\mathbf{x}. \tag{1.2}$$

The role of this representation is used to study the relationship between the sparsity-inducing basis $\{\psi_i\}$ and $\{\phi_i\}$. In particular, the condition known as incoherence of the two bases is required for recovery [BDB07]. However, only the first model will be studied in length, since the second model adds unnecessary complication.

The inverse, or reconstruction, of this seemingly easy linear transformation can take many shapes. Since, $n < N$, an inverse of Φ cannot be found, though reconstruction can be achieved in compressive sensing if the original signal is sparse enough and the sensing matrix has certain properties. A common setup for reconstruction in compressive sensing is to solve the optimization problem,

$$\min \|\Phi\tilde{\mathbf{x}} - \mathbf{y}\|_1 + \lambda\|\tilde{\mathbf{x}}\|_1. \tag{1.3}$$

The recovered signal is represented by $\tilde{\mathbf{x}}$, and the use of ℓ_1 norm will be discussed in detail in the chapter concerning recovery.

1.2 Past Work

To provide a background for the remaining chapters, a brief history of past work in compressive sensing is given.

Sampling at a rate lower than required by the Nyquist-Shannon Theorem has been

in existence for over four decades [XH07]. However, the real eruption of this field did not occur until around 2004, with the first discoveries by Emmanuel Candes. After Cande's work in magnetic resonance imaging, the first publications dealing with compressive sensing were done in relationship to statistical analysis of the use of random matrices to obtain linear measurements. However, the results of the measurements could not be verified without a recovery method [CRT06].

Since compressive sensing is an application that exploits the redundancy of structure in a sparse signal, a metric must be chosen to represent the concept of sparsity in recovery. The typical metric, l_0 ², would be a logical choice. Though l_0 may be a logical choice, computations with this metric can be troublesome. The trouble of using this metric is it lacks the properties of being strictly convex, smooth, and differentiable [Tro04, TG07, CT05]. However, other metrics have been found to work well with certain subsets of sensing matrices that have defining attributes. By using these other metrics, theorems can be stated for recovery that are necessary but may not be sufficient [Don06].

The first large set of publications on compressed sensing was done by David L. Donoho at Stanford University in 2004. In his work [Don06], Donoho outlines in detail the concept of transformation, measurements, and their interactions to form sensing matrices. The analysis is done from a geometric point of view, and the work addresses the relationship between l_0 and l_1 reconstruction. Though articles using l_1 based recovery on l_0 type recovery problems had already been published, this was one of the first articles that gave proofs relating to its use in compressive sensing. However, Donoho did not give a tangible description on the type of sensing matrices [Don06].

Many of the first works published on compressive sensing centered on the geometric analysis of recovery using a very broad set of matrices. Professionals in the

²The l_0 norm is a finite norm which counts the number of nonzero elements in a given vector.

fields of applied statistics and electrical engineering soon started to publish numerical results and construction algorithms. For nearly four years, these two fields used the holographic texts of Donoho and Candes to review linear measurements generated by random matrices and redundant dictionaries [BGI⁺08].

Recently, the field of electrical engineering has moved away from dense matrices based on geometric analysis to more sparse matrices to obtain linear measurements. One such form has a connection between compressive sensing and error correction codes [JXHC08]. Chapter two will look at sensing matrices generated based on geometric and graph analysis are outlined to show the formation of two general sensing matrices from current publication. The third chapter will look at reconstruction methods, and the fourth chapter on numerical results of sensing matrices and reconstruction techniques will be given.

Chapter 2

Sensing Matrices

In this chapter, two forms of sensing matrices that can be used for compressive sensing will be examined. The first of these matrices uses a construction based on the geometric analysis of embedding in a linear space, while the second uses graphs. The two forms of construction, though different, can be examined in a similar way by the use of spectrum analysis of the adjacency and Laplacian matrices of a graph. However, the connection would not provide a reader with a better understanding of current research in compressive sensing, since no publications exist in this area at this time.

2.1 Geometric and Probabilistic Design

The goal of compression is to exploit properties of the signal so that it can be described with less information. However, some attributes of a signal, such as the dimension of the space in which a signal is contained, gives little indication of how it should be compressed. A related real world example of this would be finding a building. The detailed instructions to 1600 Pennsylvania Avenue North West, Washington, D.C. from a general starting point, the origin, can be very long. However, knowing the area around the White House removes a large amount of

information about location that is needed. In place of holding a world map, one can replace it with a much smaller map. This example demonstrates the need to classify and generate more information about the original signal to help narrow down the information needed.

In an adaptive compression algorithm, the signal could be examined before deciding the best method for compression. In the example, the address is known before a trip to the map store. The information gives a reasonable idea of the maps that will be needed. When a sparse signal is given, an adaptive method would examine and locate the largest coefficients. These coefficients would best represent the signal since they dominate the total information of the signal [Don06].

However, compressive sensing is a non-adaptive compression, or sampling, algorithm. One way to counteract not being able to adapt to the signal is to build an algorithm that can efficiently compress a limited set of signals. The problem occurs that a signal would still need to be examined to know if it belongs to this limited set. However, signals coming from the same device or application tend to have similar properties, and by knowing these properties, the algorithm can be selected without examining every signal. As in the above example, limiting plays a key role. If it was known before going to the map store that the destination is in the United States, there would be no need to carry an international map. However, if one knew that the only destinations possible lies in the United States, many characteristics would already be known as to how to find the location. The knowledge of points that may be near or similar to the original signal would generate characteristic data that can be used to locate the signal.

One way to measure the distance between signals is with a metric. Metrics have many forms. However, a set of natural metrics arises from norms. A set of norms exists known as p - norms [Sut75]. They are defined as,

Definition 2.1.0.2. *p – norm*

$$\|\mathbf{x}\|_p \equiv (\sum_i |x_i|^p)^{\frac{1}{p}} \text{ with } p \geq 1.$$

These *p – norms* all generate a metric, $distance(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_p$. Once a way of measuring and a general point that represents the limited set is picked, the area around the signal can be examined. The subspace around this point at a certain distance defined by the metric is termed the ℓ_p – *ball* centered at that point [Sut75].

As addressed in chapter one, compressive sensing wants to take advantage of sparsity, a large percent of zero entries. Signals can be grouped into classes by sparsity using the concept of distance. Sparsity is naturally represented by the ℓ_0 – *norm*. However, the ℓ_0 – *norm* is difficult to work with since it is not a *p – norm*. A reasonable solution to this problem is to replace the ℓ_0 – *norm* by the ℓ_1 – *norm* [CT05]. This substitution can be done on certain subsets of sparse signals, and it allows the modeling to be done on ℓ_p – *balls* that are convex. All ℓ_p – *balls* with $p \geq 1$ form convex sets, but these sets are not convex when $0 < p < 1$. The property of being a convex set is important in reconstruction with optimization because it allows for the existence of a local minimum [CT05].

A restriction on which signals can be placed into a class, \mathbf{X} , would be based on the size or potential information contained in the signal [Don06]. Therefore, \mathbf{X} would contain \mathbf{x} that satisfies

$$\|\mathbf{x}\|_p \equiv (\sum_i |x_i|^p)^{\frac{1}{p}} \leq R \tag{2.1}$$

and where $p \in (0, 2)$ and $R > 0$.

Once a class of signals is defined, the goal of embedding $\mathbf{x} \in \mathbf{X}$ in a smaller subspace, or describing it with a smaller amount of information, can be related to Gel’fand n-width.

Definition 2.1.0.3. [Gelfand n-width] [Don06]

The Gelfand n -width of \mathbf{X} with respect to ℓ_2 - norm in \mathbb{R}^N is defined as

$$d^n(\mathbf{X}; \ell_2) = \inf_{V_n} \sup\{\|\mathbf{x}\|_2 : \mathbf{x} \in V_n^\perp \cap \mathbf{X}\} \quad (2.2)$$

where the infimum is over n -dimensional linear subspaces of \mathbb{R}^N , and V_n^\perp denotes the ortho-complement of V_n with respect to the standard Euclidean inner product.

Gelfand n -width is the subspace V_n^\perp with the property that the norm of the projected signal is as small as possible [Don06]. However, it is important to measure not only the size of the projection, but the error in recovery to judge the efficiency of the compression.

Let Φ be a non-adapted function that maps the original signal, $\mathbf{x} \in \mathbb{R}^N$ to \mathbb{R}^n . Also, let Ψ be an unspecified, possibly nonlinear reconstruction operator, or algorithm. An approximation of error for a signal can be given by

$$E_n(\mathbf{X}) = \inf_{\Phi_n, \Psi_n} \sup_{\mathbf{x} \in \mathbf{X}} \|\mathbf{x} - \Psi_n(\Phi_n(\mathbf{x}))\|_2 \quad (2.3)$$

where n represents some pair of algorithms based in \mathbb{R}^n [Don06].

It was shown by Donoho that there exists a strong relationship between Gelfand n -widths and the error of recovery for non-adaptive methods given by

Theorem 2.1.0.1. [Don06]

$$d^n(\mathbf{X}; \ell_p) \leq E_n(\mathbf{X}) \leq 2^{\frac{1}{p}-1} \cdot d^n(\mathbf{X}; \ell_p) \quad (2.4)$$

where \mathbf{X} is a class containing element in \mathbb{R}^N defined by $0 < p \leq 1$ and $R > 0$ and d^n is the Gelfand n -width.

From the given theorem, the Gelfand n -width gives the exact value of optimal information when $p = 1$, and gives a good approximation for $p < 1$. In addition,

subspace dimensionality can also be studied using Kolmogorov n-widths [Don06]. In Kolmogorov n-widths, d_n measures the quality of approximation of \mathbf{X} possible by V_n , and these n-widths have a duality relationship with Gel'fand n-widths when $p = 1$.

Definition 2.1.0.4. [Kolmogorov n-width] [Don06]

Let $\mathbf{X} \subset \mathbb{R}^N$ be a bounded set. The Kolmogorov n-width of \mathbf{X} with respect to the ℓ_2 - norm in \mathbb{R}^N is defined as

$$d_n(\mathbf{X}; \ell_2) = \inf_{V_n} \sup_{\mathbf{x} \in \mathbf{X}} \inf_{\mathbf{y} \in V_n} \|\mathbf{x} - \mathbf{y}\|_2 \quad (2.5)$$

where the infimum is over n-dimensional linear subspaces of \mathbb{R}^N .

In addition, Gel'fand and Kolmogorov n-widths can be used to prove:

Theorem 2.1.0.2. Let (n, N_n) be a sequence of problem size with $n < N_n$, $n \rightarrow \infty$, and $N_n \sim \Psi n^\gamma$, $\gamma > 1$, $\Psi > 0$. Then for $0 < p \leq 1$, there is $C_p = C_p(\Psi, \gamma) > 0$ so that

$$E_n((X)) \leq C_p \cdot R \cdot (n/\log(N_n))^{\frac{1}{2} - \frac{1}{p}}, n \rightarrow \infty. \quad (2.6)$$

This theorem states that the amount of information lost is less than or equal to the product of the radius restriction, R , defining the class \mathbf{X} and the proportion of n to the $\log(N_n)$ for a non-adaptive method [Don06].

A similar expression can be generated for an adaptive method using the same method as above. In an adaptive method that takes L of the largest coefficients, the error of the original and approximation, \mathbf{x}_L would yield the expression:

$$\|\mathbf{x} - \mathbf{x}_L\|_2 \leq \xi_{2,p} \cdot \|\mathbf{x}\|_p \cdot (L + 1)^{\frac{1}{2} - \frac{1}{p}}, \text{ for } L = 0, 1, 2, \dots, \quad (2.7)$$

with the constant $\xi_{2,p}$ depending only on $p \in (0, 2)$ [Don06].

In comparing these two equations, a relationship can be made between the number of non-adaptive measurements needed to achieve results similar to the adaptive

method. Similar results can be achieved by letting $n \approx L \cdot \log(N)$ [Don06]. The result allows a method of compression to work for a class of signal with only needing to adjust the number of measurements by a factor of $\log(N)$. In addition, the slow growing nature of the \log function means the number of measurements will grow at a rate that is sublinear to the signal when the signal is large.

The errors of $E_n(\mathbf{X})$ and $E_n^{Adapt}(\mathbf{X})$, where

$$E_n^{Adapt}(\mathbf{X}) = \inf_{\Psi_n, \Phi_n^A} \sup_{\mathbf{x} \in \mathbf{X}} \|\mathbf{x} - \Psi_n(\Phi_n^A(\mathbf{x}))\|_2, \quad (2.8)$$

can be combined to give the following theorem.

Theorem 2.1.0.3. [Don06]

For $0 < p \leq 1$ and $C_p > 0$,

$$E_n(\mathbf{X}) \leq 2^{\frac{1}{p}} \cdot E_n^{Adapt}(\mathbf{X}). \quad (2.9)$$

The case with the smallest non-adaptive error is generated when $p = 1$. In this case, the error of the non-adaptive method is at most twice the error of the adaptive method. These results merit the need for tangible non-adaptive compression algorithm, Φ . However, the existence of Φ is not enough to design a sufficient algorithm. In addition, the existence of a Φ to be used for compression is not unique. Testing that a given Φ is a compressive sensing matrix can be time consuming [BDDW07]. In addition, a way of classifying the compression algorithms that would minimize $E_n(\mathbf{X})$ would be beneficial in testing suitable algorithms.

2.1.1 Introduction to The Restricted Isometry Property

(RIP)

The reconstruction problem of finding a sparse solution to an underdetermined systems of linear equations is NP-hard [CT05]. Verifying a viable Φ that can be used for compressive sensing with a sufficient condition of reconstruction can be a daunting task. However, the work of Emmanuel Cades and Terence Tao introduced the Restricted Isometry Property which provides a condition for matrices that will be sufficient for recovery with little to no error [CT05].

Theorem 2.1.1.1. [Restricted Isometry Property (RIP)] [CT05]

A given matrix $\Phi \in \mathbb{R}^{n \times N}$ satisfies the Restricted Isometry Property, RIP, of order k if there exists a $\delta_k \in (0, 1)$ such that

$$(1 - \delta_k) \cdot \|\mathbf{x}_T\|_{\ell_2}^2 \leq \|\Phi_T \mathbf{x}_T\|_{\ell_2}^2 \leq (1 + \delta_k) \cdot \|\mathbf{x}_T\|_{\ell_2}^2 \quad (2.10)$$

holds for all sets T of column vectors of Φ with $|T| \leq k$ such that $\Phi_T \in \mathbb{R}^{n \times |T|}$ and \mathbf{x}_T is the vector obtained by retaining only the entries of \mathbf{x} corresponding to the column indices T .

The RIP is a way of quantifying the degree Φ holds to the property deemed *restrictedly almost orthonormal systems*, which is a collection of vectors which behaves like an orthonormal system but only for sparse linear combinations [CT05]. A square matrix $\hat{\Phi} \in \mathbb{R}^{N \times N}$ that is orthonormal and produces the samples, $\hat{\mathbf{y}} = \hat{\Phi} \mathbf{x}$, can easily recover \mathbf{x} using basic linear algebra. Since the signal is sparse, only a subset of k column vectors of Φ will be used to calculate each linear measurement. Therefore, only subsets of column vectors with cardinality less than or equal to k need to be examined. Matrices, Φ_T , can be constructed from these subsets. The amount that each matrix, Φ_T , varies from being orthonormal can then be estimated [CT05].

The approximate amount that each Φ_T varies from being orthonormal can be expressed by substituting several properties of norms: For any given matrix A and vector \mathbf{x} , $\|A\mathbf{x}\|_B \leq \|A\|_{AB} \cdot \|\mathbf{x}\|_A$, and $\|A\|_2 = \sqrt{\lambda_{\max} \text{of}(A^*A)}$ [Str07]. In the case of the RIP, each Φ_T is desired to be close to orthonormal. In addition, orthonormal matrices have the properties of having all eigenvalues having magnitude equal to one and having its adjoint equivalent to its inverse, i.e., $\Phi_T^* \Phi_T = \Phi_T \Phi_T^* = I$ where I is the identity matrix. Therefore, $\|\Phi_T\|_2$ can be estimated to be close to the value 1.

In verifying that the RIP achieves the desired result, the Johnson-Lindenstrauss Lemma can be used. The original formulation of Johnson and Lindenstrauss is as follows:

Lemma 2.1.1.1. [Johnson-Lindenstrauss] [BDDW07]

Let $\epsilon \in (0, 1)$ be given. For every set Q of $\#(Q)$ points in \mathbb{R}^N , if n is a positive integer such that $n > n_0 = O(\ln(\#(Q))/\epsilon^2)$, there exists a Lipschitz mapping $f : \mathbb{R}^N \rightarrow \mathbb{R}^n$ such that

$$(1 - \epsilon) \cdot \|\mathbf{u} - \mathbf{v}\|_{\ell_2} \leq \|f(\mathbf{u}) - f(\mathbf{v})\|_{\ell_2} \leq (1 + \epsilon) \cdot \|\mathbf{u} - \mathbf{v}\|_{\ell_2} \quad (2.11)$$

for all $\mathbf{u}, \mathbf{v} \in Q$.

The Lipschitz mapping¹, f , can be taken as a linear mapping represented by an $n \times N$ matrix Φ under certain conditions. In particular, Φ is normally taken as a matrix whose entries are randomly drawn from certain probability distributions in many proofs of the Johnson-Lindenstrauss Lemma [BDDW07].

The randomly drawn values can be represented by a probability measure space, (Ω, ρ) . In the probability measure space, (Ω, ρ) , the independent observed values of a random variable on Ω can be used to generate random matrices $\Phi(\iota), \iota \in \Omega^{nN}$ [BDDW07]. Given any set of points in Q , the matrix $f = \Phi(\iota)$ will satisfy the

¹A function is said to be a Lipschitz mapping, if there exists a constant $C \in \mathbb{R}^+$ such that for all x and y in the domain of f , $|f(x) - f(y)| \leq C \cdot |x - y|$ [Sut75].

Johnson-Lindenstrauss Lemma with high probability provided n satisfies the condition of Johnson-Lindenstrauss Lemma, i.e., n is a positive integer such that $n > n_0 = O(\ln(\#(Q))/\epsilon^2)$. The exact probability can be proven by first limiting the set of $\Phi(\iota)$ such that for any $\mathbf{x} \in \mathbb{R}^N$ the random variable $\|\Phi(\iota)\mathbf{x}\|_{\ell_2}^2$ has the expected value $\|\mathbf{x}\|_{\ell_2}^2$, i.e.,

$$\mathbb{E}(\|\Phi(\iota)\mathbf{x}\|_{\ell_2}^2) = \|\mathbf{x}\|_{\ell_2}^2. \quad (2.12)$$

The random matrix generated from this probability space can be shown to have the property that the

$$Pr(\|\Phi(\iota)\mathbf{x}\|_{\ell_2}^2 - \|\mathbf{x}\|_{\ell_2}^2 \geq \epsilon\|\mathbf{x}\|_{\ell_2}^2) \leq 2e^{-nc_0(\epsilon)}, \quad 0 < \epsilon < 1, \quad (2.13)$$

where the probability is taken over all $n \times N$ matrices $\Phi(\iota)$ and $c_0(\epsilon)$ is a constant depending only on ϵ such that for all $\epsilon \in (0, 1)$, $c_0(\epsilon) > 0$. This equation based on the moment condition allows one to show the probability that a random matrix will satisfy the Johnson-Lindenstrauss Lemma [BDDW07].

Lemma 2.1.1.2. [Probability of Random Matrix Satisfying JL-Lemma] [BDDW07]

Let $\Phi(\iota), \iota \in \Omega^{nN}$ be a random matrix of size $n \times N$ drawn according to any distribution that satisfies the above inequality. Then for any set T with $\#(T) = k < n$ and any $0 < \delta < 1$, we have

$$(1 - \delta) \cdot \|\mathbf{x}\|_{\ell_2} \leq \|\Phi(\iota)\mathbf{x}\|_{\ell_2} \leq (1 + \delta) \cdot \|\mathbf{x}\|_{\ell_2}, \quad \text{for all } \mathbf{x} \in \mathbf{X}_T \quad (2.14)$$

with the probability

$$\geq 1 - 2\left(\frac{12}{\delta}\right)^k e^{-c_0(\frac{\delta}{2})n} \quad (2.15)$$

Once the probability that $\Phi(\iota)$ will fail is known for each k dimensional space of \mathbf{X} , they can be combined to find the total probability of failure. There are $\binom{N}{k}$ subspaces for a given k and N [BDDW07]. Noting that $\binom{N}{k} \leq (eN/k)^k$, the RIP will

fail to hold with probability

$$\leq 2(eN/k)^k(12/\delta)^k e^{-c_0(\delta/2)n} = \exp(-c_0(\delta/2)n + k(\log(en/k) + \log(12/\delta)) + \log(2)). \tag{2.16}$$

Therefore the probability of a random matrix satisfying the RIP can be summarized as follows:

Theorem 2.1.1.2. [BDDW07]

Suppose that n , N , and $0 < \delta < 1$ are given. If the probability distribution generating the $n \times N$ matrices $\Phi(\iota)$, $\iota \in \Omega^{nN}$, satisfies the above inequality, then there exist constants $c_1, c_2 > 0$ depending only on δ such that the RIP holds for $\Phi(\iota)$ with the prescribed δ and any $k \leq c_1 n / \log(N/k)$ with the probability $\geq 1 - \exp(-c_2 n)$.

2.1.2 A Deeper Look at RIP

A sensing matrix that satisfies the RIP has the property of being close to an orthonormal matrix for sparse signals. The degree that the matrix, Φ , and its sub-matrices vary from orthonormal is quantified with the parameter δ_k . The smaller the value of δ_k , the closer the sub-matrices are to forming an ideal orthonormal system [CT05]. The question of how small δ_k has to be to achieve desired characteristics, such as the recovery of a unique signal, is the nature question. However, one cannot construct a random matrix based on δ_k that will guarantee the desirable characteristic. A solution to the problem is to build a stronger requirement based on increasing the size of k . Therefore, one would want to satisfy the RIP for a larger set, such as $2k$, so that the matrix would have the desired property. A random matrix can then be tested to have this property by examining the value of δ_k .

One of the first desirable properties in recovery would be the existence of a unique solution [CT05]. The ability to recover a unique solution after using one of these

sensing matrices was first outlined by Candes and Tao [CT05]. It is stated that:

Lemma 2.1.2.1. [CT05]

Suppose that $k \geq 1$ is such that $\delta_{2k} < 1$, and $|T| \leq k$. Let $\mathbf{y} := \Phi_T \mathbf{x}_T$ for some arbitrary $|T|$ -dimensional vector \mathbf{x} . Then the set T and the coefficients of \mathbf{x}_T can be reconstructed uniquely from the knowledge of \mathbf{y} and all the column vectors of Φ .

In addition, the norm error of the recovered signal, $\tilde{\mathbf{x}}$, can be reduced by limiting the size of δ for a larger set k [CT05]. In particular, if Φ satisfies the RIP with $\delta_{3k} < 1$, it can be shown that:

Lemma 2.1.2.2.

$$\|\mathbf{x} - \tilde{\mathbf{x}}\|_{\ell_2} \leq \frac{C \cdot \|\mathbf{x} - \mathbf{x}_k\|_{\ell_2}}{k^{1/2}} \quad (2.17)$$

for C depending only on δ_{3k} .

In combining the results of the two above lemmas, a new result of an exact solution can be formed [CRT06].

Lemma 2.1.2.3. For a given Φ that satisfies the RIP condition such that

$\delta_{2k} + \delta_{3k} < 1$, then the solution to

$$\min_{\tilde{\mathbf{x}} \in \mathbb{R}^N} \|\tilde{\mathbf{x}}\|_{\ell_1} \quad \text{subject to} \quad \Phi \tilde{\mathbf{x}} = \mathbf{y} \quad (2.18)$$

is exact.

These conditions are many times rolled up with a third condition known as stable recovery. In stable recovery, linear measurements \mathbf{y} are perturbed by some noise before reconstruction is considered. If $\mathbf{y} = \Phi \mathbf{x} + e$ with $\|e\|_{\ell_2} < \epsilon$, then the recovery of

$$\min_{\hat{\mathbf{x}} \in \mathbb{R}^N} \|\hat{\mathbf{x}}\|_{\ell_1} \quad \text{subject to} \quad \|\mathbf{y} - \Phi \hat{\mathbf{x}}\|_{\ell_2} \leq \epsilon \quad (2.19)$$

then for any $\mathbf{x} \in \mathbb{R}^N$

$$\|\mathbf{x} - \tilde{\mathbf{x}}\|_{\ell_2} \leq \frac{C_1 \cdot \|\mathbf{x} - \mathbf{x}_k\|_{\ell_2}}{k^{1/2}} + C_2 \cdot \epsilon. \quad (2.20)$$

The three properties related to the size of δ are normally termed: Tractable Recovery, Robustful Recovery, and Stable Recovery. Tractable recovery ensures that all k - *sparse* signals are perfectly recovered via ℓ_1 minimization [CT05]. Robustful Recovery is the property that is achieved in (2.17) and stable recovery as defined above. Empirical results have shown that all three of these properties can be achieved when $\delta_{2k} < .4$ [CT05].

Despite only having to verify up to δ_{2k} , testing a given matrix can be troublesome. The test requires checking eigenvalues of each submatrix, Φ_T . As a result, a set of well studied matrices with known properties are normally used in compressive sensing.

2.1.3 RIP Matrices

In many cases, random matrices are used in relationship to RIP. As stated earlier, the probability that a random matrix satisfies the RIP can be calculated using the Johnson-Lindenstrauss Lemma.

The Gaussian random matrix, Φ_G , is one of the best behaving and well studied random matrices for compressive sensing. The matrix draws entries as independent observed random variables of the distribution having mean value 0 and variance of $\frac{1}{n}$. The parameter, $c_0(\epsilon)$, that defines the probability that the matrix from the given distribution satisfies the expected value condition is $c_0(\epsilon) = \epsilon^2/4 - \epsilon^3/6$ [BDDW07]. This parameter makes the Gaussian random matrix ideal due to the high probability of being a suitable sensing matrix. In addition, the number of measurements, n , needs to be only $O(k \log(N/2k))$ [CT05]. This is one of the largest reductions using sensing matrices.

Random matrices are not the only type of matrix used with RIP. Attempts have been made to find deterministic matrices that fulfill the RIP. One such matrix was constructed by Ronald DeVore [DeV07]. DeVore utilizes the cyclic nature of polynomials on a prime finite field for his construction. In particular,

Theorem 2.1.3.1. [DeVore Construction] *Let Φ_0 be the $n \times N$ matrix with columns v_Q , $Q \in \mathbb{P}_r$ with these columns ordered lexicographically with respect to the coefficients of the polynomials. Then, the matrix $\Phi := \frac{1}{\sqrt{p}}\Phi_0$ satisfies the RIP with $\delta = (k - 1)r/p$ for any $k < p/r + 1$.*

This deterministic construction limits the choice of $n := p^2$, $N := p^{r+1}$, and $k < \frac{p}{r} + 1$ where p is a prime number and $0 < r < p$. These limitations are far more constricting than are most random matrices [DeV07].

A lower bound on the number of linear measurements needed for a deterministic build that satisfies the RIP using only the values $\{0, 1\}$ entries does exist and shows the restriction relating to the number of measurements needed for recovery. If the RIP condition were to be rephrased as the following:

Definition 2.1.3.1. [(k,D)-RIP]

Let Φ be an $n \times N$ matrix. Then, Φ satisfies (k,D)-RIP if there exists $c > 0$ such that all k - sparse $\mathbf{x} \in \mathbb{R}^N$

$$\|\Phi\mathbf{x}\|_{\ell_2} \leq \|\mathbf{x}\|_{\ell_2} \leq cD\|\Phi\mathbf{x}\|_{\ell_2}. \tag{2.21}$$

The lower bound n depends on D , or in general $O(\delta)$, such that $n \geq \min\{\frac{k^2}{D}, \frac{N}{D^2}\}$.

These bounds exist for matrices satisfying the RIP; however other sufficient conditions may exist for other types of deterministic sensing matrices.

2.2 Graph Construction

The geometric analysis used for the construction of sensing matrices in the previous sections led to the development of the RIP and reconstruction guarantees. Most sensing matrices that satisfy the RIP are non-sparse. In addition, there exists a correlation between the degree of sparsity of an RIP sensing matrix and the decrease of linear measurements [XH07]. The RIP-2² does provide a very large decrease in dimensionality of the original signal. However, this large decrease comes at the price of using dense matrices with polynomial time recovery methods [XH07].

Construction methods based on graphs and pseudo-random distributions offer a different perspective than geometric analysis. Many of the sensing matrices generated with this approach are sparse, some of which have deterministic construction. These very sparse binary matrices can be constructed fast and stored with little memory. In addition, the reconstruction process on several matrices can be performed in sub-linear time [XH07]. Many of these matrices do suffer the requirement of more linear measurements than do RIP-2 sensing matrices [BGI⁺08]. However, the increase in linear measurements is not as substantial as the stated in the last section.

2.2.1 Expander Graph Construction

The best deterministic RIP-2 sensing matrix requires the number of linear measurements be $\Omega(k^2)$ [BGI⁺08]. However, it is possible to construct a sensing matrix based on an unbalanced expander graph that can be used for compressive sensing that has measurements $O(k \log(n))$ [BGI⁺08]. Expander graphs can be thought of as graphs that are sparse in edges but that are still well connected. Expander Graphs have been used in the past by areas of information theory dealing with the similar dual problem of low-density parity-check codes and superimposed codes [BGI⁺08].

²The previous RIP will now be addressed as RIP-2 because it is based on ℓ_2

Definition 2.2.1.1. [Unbalanced Bipartite Expander Graph] [XH07]

An $(\alpha N, \beta d)$ unbalanced expander with regular left degree d is a bipartite simple graph $\mathbf{G} = [\mathcal{A}, \mathcal{B}]$ with N left variable nodes, \mathcal{A} , and n right parity check nodes, \mathcal{B} , such that for some $0 < \alpha, \beta < 1$ if for every subset, $\mathcal{V} \in \mathcal{A}$, with $|\mathcal{V}| \leq \alpha N$, $|\mathcal{N}(\mathcal{V})| > \beta d |\mathcal{V}|$, where $\mathcal{N}(\mathcal{V})$ is the set of neighbors of \mathcal{V} .

This type of graph construction can be written as an $n \times N$ adjacency matrix, Φ . Each column of Φ would have exactly d ones. Non-regular degree expander graphs can be used in substitution. However, non-regular expander graphs do not add any special results, and they only tend to make analysis more complex. In particular, the only parameter that has a detrimental effect of an expander graph being used as a sensing matrix is β [JXHC08]. The parameter α has the ability to help determine the type of recovery that can be used and normally follows from the selection of β . However, the parameter d has little to do with the ability of an expander being a good sensing matrix. In particular, it can be shown,

Theorem 2.2.1.1. [XH07]

Let $0 < \beta < 1$ and the ratio $r = \frac{n}{N}$ be given. Then for large enough N there exists a regular left degree d bipartite expander for some $0 < \alpha < 1$ and constant d that does not grow with N .

Inversely, the probability that a random binary adjacency matrix with a constant number of ones in a column is an expander graph does not depend directly on d .

Theorem 2.2.1.2. [BM01]

A left regular bipartite graph that is chosen by using vectors of random combinations will be an $(\alpha N, \beta d)$ expander with probability $1 - O(N)/N$ for any $\beta > 1 - \frac{1}{d}$.

In particular, the assumption that $d \geq 5$ and $\beta > 3/4$ is made for most expanders. This assumption does not always have to be the case, and these values are based on

sub-linear recovery algorithms. However, these values provide ideal reconstruction and deterministic construction of expanders produced using the zig-zag product [JXHC08]. These requirements can be lessened when examining sensing matrices based on graphs that wish to be recovered by optimization similar to RIP-2.

2.2.2 RIP-1

A similar property to RIP-2 can be constructed for a sensing matrix generated from graphs. This property, RIP-1, is similar but not equivalent and cannot be directly compared to RIP-2 using classical analysis. The definition of this more relaxed property for any p is:

Definition 2.2.2.1. [RIP-p] [BGI⁺08]

An $n \times N$ matrix Φ is said to satisfy RIP- p , if for any k -sparse vector \mathbf{x} , we have

$$\|\mathbf{x}\|_{\ell_p} \leq \|\Phi \mathbf{x}\|_{\ell_p} \leq (1 + \delta) \cdot \|\mathbf{x}\|_{\ell_p}. \quad (2.22)$$

This property can generate the same sufficient condition for recovery as the RIP-2 for $1 \leq p \leq 1 + O(1)/\log(N)$ if Φ is an adjacency matrix of a high-quality expander graph. In particular, any $n \times N$ matrix adjacency matrix of an $(k, 1 - \epsilon)$ left regular expander with degree d , such that $1/\epsilon$ and d are smaller than N would be a high-quality expander. Then the scaled matrix, $\frac{1}{d^{1/p}}\Phi$, satisfies the RIP- p for $1 \leq p \leq 1 + O(1)/\log(N)$ and $\delta = C \cdot \epsilon$ for some absolute constant $C > 1$. This theorem allows almost all unbalanced left regular expander graphs to be used with optimization based recovery similar to RIP-2 [BGI⁺08].

In parallel to the coefficients of δ_{2k} and δ_{3k} of RIP-2, the parameters of RIP- p can be adjusted to provide a better understanding of recovery with optimization.

However, δ is a very general parameter that can be applied to a large set of matrices.

Since the only type of matrix that needs to be examined is that generated by

expander graphs, the adjustment of a parameter to the individual expander graph makes more analytic sense. In particular, if an $(2k, 1 - \epsilon)$ expander graph satisfies the RIP- p , then the optimization problem will recover a $\tilde{\mathbf{x}}$ that satisfies $\Phi\tilde{\mathbf{x}} = \Phi\mathbf{x}$ where \mathbf{x} is the original signal. However, this property does not claim that the solution is unique, $\tilde{\mathbf{x}} = \mathbf{x}$ [JXHC08]. A unique solution can be generated with the tightened parameter on the expander graph. When the parameter $\alpha N = 3k$, the recovery will provide a unique solution under all the same constraints as the $2k$ case [JXHC08].

2.2.3 Remarks

Sensing matrices generated from graphs can provide properties that dense sensing matrices cannot. These light weight matrices require little time to generate and very little memory to store. In addition, sub-linear recovery methods for this type of sensing matrix provide faster recovery with less resources. These will be explained in the next chapter. The cost of these properties is that the number of linear measurements will be higher than those of a dense random sensing matrix. However, this cost may be worth the cost in portable devices that have limited memory and processing power.

In the next chapter, recovery algorithms for both forms of sensing matrices will be examined.

Chapter 3

Reconstruction

The reconstruction of a sparse signal after compressive sensing can be interpreted in many ways. The natural way of using sparsity would involve the use of the ℓ_0 - *norm*. The reconstruction problem with this norm can be written as the optimization problem

$$\tilde{\mathbf{x}} = \arg \min_{\mathbf{x} \in \mathbb{R}^N} \|\mathbf{x}\|_0 \quad \text{subject to} \quad \mathbf{y} = \Phi \mathbf{x}. \quad (3.1)$$

However, this problem written in this formation is NP-hard. Research on this form of problem, along with its dual used for sparse error correction, predates the first publications of compressive sensing [SWM08].

In the present research, reconstruction of the above equation can be classified into several broad categories. The first category is termed pursuit models. Several types of pursuit models exist, but they all share the theme of breaking the problem into smaller problems depending on a dictionary, i.e., a collection of vectors that span the entire space of \mathbb{R}^N . In particular, problems that deal with sparse reconstruction have over complete dictionaries [CDS98, Tro04]. There are many heuristic pursuit methods, such as Matching Pursuit (MP), Orthogonal Matching Pursuit (OMP), and m-Fold Matching Pursuit [CDS98], which can be used to recover a sparse signal. A

heuristic method called Orthogonal Matching Pursuit will be examined in more detail later in this chapter. In addition to the heuristic pursuit methods, several other methods, such as Basis Pursuit, exist [CDS98]. The second category is based on optimization. This second category is directly linked to and is a result of the development of Basis Pursuit, in which the optimization problem is cast to a ℓ_1 optimization problem. Many optimization techniques exist, and each technique claims to have smaller errors, faster computational times, or use less active memory [FNW07]. The third category that will be examined depends on graph based construction of the sensing matrix. These methods resemble greedy based pursuit methods. However, these methods model the problem using graph theory as opposed to its associated linear algebra representation.

3.1 Pursuit Methods

The most general way to define a pursuit method is an algorithm that holds onto both action-valued estimates and action preferences, i.e., an estimate of what is gained or lost taking a particular action and when this estimated value is high or low enough to justify the action. This general definition can be related to a wide range of algorithms that are used for signal recovery and many other fields, such as intelligent agents. The pursuit methods that will be examined here base their actions on rules that use information from a dictionary of vectors. Each vector in the dictionary is normally called an atom, which is referenced by α_i [Tro04].

The dictionary itself has several properties that determine if the rule the algorithm implements will lead to an action-valued estimate that is close to the desired result. One very rough property of a dictionary is termed *coherence*.

Definition 3.1.0.1. [Tro04]

The coherence of a dictionary is equal to:

$$\mu := \max_{j \neq k} |\langle \alpha_j, \alpha_k \rangle|. \quad (3.2)$$

The parameter μ of the dictionary gives a very blunt estimate of the amount that atoms in the dictionary overlap, i.e., linear dependent. For example, the coherence parameter of a set of atoms that are orthonormal would have $\mu = 0$. However, the addition of an identical atom would change $\mu = 1$ [Tro04].

A dictionary is said to be incoherent if the coherent parameter is negligibly close to zero. Therefore, an incoherent dictionary tends to behave like an orthonormal dictionary. However, the single parameter of coherence can be misleading when describing a dictionary [Tro04].

An additional property of the dictionary that would describe the relationship of the atoms is the babel function.

Definition 3.1.0.2. [Tro04]

The babel function for a dictionary, \mathcal{D} , with atoms, α_i , that are indexed by the set Ω is defined as

$$\mu_1(k) := \max_{|\Lambda|=k} \max_{\phi} \sum_{|\Lambda|} |\langle \phi, \alpha_\lambda \rangle|, \quad (3.3)$$

where the value of k is the sparsity of the vector the pursuit wishes to reconstruct and ϕ ranges over the atoms indexed by $\Omega \setminus \Lambda$.

This definition can be summed up as a function that quantifies the maximum total coherence between a fixed atom and a collection of k other atoms. For example, $\mu_1(1) = \mu$. It follows from algebra that a dictionary with coherence value μ would have a babel function such that,

$$\mu_1(k) \leq \mu \cdot k. \quad (3.4)$$

When the growth of the babel function is slow, the dictionary is said informally to be *quasi-incoherent* [Tro04]. This terminology can be interpreted by the same concepts that generate the Restricted Isometry Property. In the RIP, the goal was to have each sub-matrix close to orthonormal. If each sub-matrix of Φ was close to orthonormal, then the dictionary for the sub-matrix would be considered incoherent, where the cut off is based on $(1 \pm \delta_T)$. Thus, the babel function would look at nearly identical information. The dictionary for a sensing matrix would be redundant with high probability due to the dimensionality reduction. In addition, a k - *sparse* vector would need to use only k atoms for each measurement [Tro04].

It is under the above assumptions that pursuit methods for reconstruction are based. Tropp showed that a sufficient condition for exact recovery for OMP and Basis Pursuit could be generated based on the above definition and use of the babel function. However, the results require the fore-knowledge of the optimum set of atoms for any given vector which makes the result less utilitarian than the RIP. This sufficient condition based on Tropp is the same for both methods, and they have very similar proofs [Tro04].

3.1.1 Orthogonal Matching Pursuit

The precursor to Orthogonal Matching Pursuit, matching pursuit, was first introduced in detail by Mallet and Zhang in 1992. This greedy based algorithm attacked the problem of sparsity head on. It was posed that the algorithm would start with an approximation of the original signal having all zero elements with a residual of $\mathbf{y} = \Phi\tilde{\mathbf{x}}$, where $\tilde{\mathbf{x}}$ is the approximation. The algorithm would continue step-wise for k steps identifying atoms from the dictionary with the largest contribution and add them to the original approximation [CDS98]. At the i - *th* stage, $\tilde{\mathbf{x}}^i = \tilde{\mathbf{x}}^{i-1} + \langle R^{i-1}, \alpha_i \rangle \alpha_i$ where $R^i = \mathbf{y} - \tilde{\mathbf{x}}^i$. In general, this algorithm works perfectly if the dictionary is orthogonal. However, the dictionary of a sensing matrix

is only close to orthogonal [CDS98].

The Orthogonal Matching Pursuit, which was first outlined by the work of DeVore and Temlyakov added an additional step to matching pursuit. This extra step of orthogonalization at each of the k -steps requires that each new element added will be orthogonal to all terms already in the model. The step of orthogonalization added the need to solve a least square problem [CDS98], i.e.,

$$\min_{b_i} \left\| \mathbf{y} - \sum_{i=1}^k b_i \alpha_{\lambda_i} \right\|_2. \quad (3.5)$$

The resulting algorithm is outlined as:

Algorithm 1 Orthogonal Matching Pursuit [TAG07]

INPUT:

- $\Phi \in \mathbb{R}^{n \times N}$.
- $\mathbf{y} \in \mathbb{R}^n$.
- Sparsity level k .

OUTPUT:

- An estimation, $\tilde{\mathbf{x}} \in \mathbb{R}^N$, for the original signal.
- A set, Λ_k , containing k elements.
- An residual vector $R \in \mathbb{R}^n$.

PROCEDURE:

- Initialize $R^0 = \mathbf{y}$, the index set $\Lambda_0 = \emptyset$, and the iteration counter $t = 1$.
- Find the index λ_t that solves the optimization problem
 $\lambda_t = \arg \max_{j=1, \dots, N} |\langle R_{t-1}, \Phi_j \rangle|$.
If the maximum occurs for multiple indices, break the tie deterministically.
- Augment the index set $\Lambda_t = \Lambda_{t-1} \cup \{\lambda_t\}$ and the matrix of chosen atoms $\tilde{\Phi}_t = [\tilde{\Phi}_{t-1} \Phi_{\lambda_t}]$.
- Solve a least-squares problem to obtain a new signal estimate by computing
 $\mathbf{s}_t = \arg \min_{\mathbf{s}} \|\tilde{\Phi}_t \mathbf{s} - \mathbf{y}\|_2$.
- Calculate the new approximation of the data and residual:
 $\mathbf{a}_t = \tilde{\Phi}_t \mathbf{s}_t$ and $R_t = \mathbf{y} - \mathbf{a}_t$.
- Increment t .
- Continue to loop while $t < k$.
- $\tilde{\mathbf{x}}$ has nonzero indices at the components listed in Λ_k . The value of the estimate $\tilde{\mathbf{x}}$ in component λ_j equals the component $j - th$ component of \mathbf{s}_t .

OMP is known for its fast reconstruction with known sparsity level. The algorithm itself is very fast running since most computation goes into solving the small optimization problems that are bounded in spaces smaller than k . However, the cost of the speed for the OMP is being clairvoyant about the degree of sparsity. In

addition, empirical evidence shows a lower average error in reconstruction than Basis Pursuit produces [TAG07].

3.1.2 Basis Pursuit

Basis Pursuit is designed to reconstruct a sparse signal from over complete dictionaries using convex optimization. The development of Basis Pursuit is similar in nature to all pursuit methods. However, the Basis Pursuit looks at trying to find a global optimal, unlike greedy algorithms that look to find only the optimal move at each step [CDS98]. Both greedy and non-greedy algorithms have benefits and downfalls. In many cases, a greedy algorithms tend to be fast. An example of a greedy algorithm is making change. In this algorithm, the largest moves are made first, such as finding the largest number of quarters that can be given. However, greedy algorithm may not always give a global optimal. If the set of currency used changed to $\{25, 10, 6, 5, 1\}$, the greedy algorithm for giving change for 12 cents would yield $\{10, 1, 1\}$, which is not the global optimal of $\{6, 6\}$.

The global optimization problem for Basis Pursuit can be modeled as:

$$\min_{\tilde{\mathbf{x}} \in \mathbb{R}^N} \|\tilde{\mathbf{x}}\|_1 \quad \text{subject to} \quad \Phi \tilde{\mathbf{x}} = \mathbf{y}. \quad (3.6)$$

Though Basis Pursuit can be summed up to one equation, the total effort and sophistication needed to solve a convex, nonquadratic optimization problem of the above form is more than solving the smaller optimization problems of OMP [CDS98].

However, many forms of optimization used to solve the above equation use greedy based algorithms somewhere. In comparison to OMP, the simplest implementation of Basis Pursuit, such as simplex linear programming, works in an opposite way to achieve sparsity. In OMP, the recovery estimate is assumed to be a zero vector and then coefficients are added. However, Basis Pursuit solved using a simplex method

assumes the whole model is already complete and swaps members to improve the magnitude of the objective function [CDS98].

3.2 Optimization

The formation of the Basis Pursuit method is centered on using optimization to solve the global problem of recovery. Though many optimization based reconstructions exist, most hinge on the original work done with Basis Pursuit. The problem,

$$\min_{\tilde{\mathbf{x}} \in \mathbb{R}^N} \|\tilde{\mathbf{x}}\|_1 \quad \text{subject to} \quad \Phi \tilde{\mathbf{x}} = \mathbf{y}.$$

can be interpreted as a linear programming (LP) problem. The general form of a linear program is a constrained optimization problem with $\mathbf{x} \in \mathbb{R}^N$ such that

$$\min \mathbf{c}^T \mathbf{x} \quad \text{subject to} \quad A\mathbf{x} = \mathbf{b}, \quad \mathbf{x} \geq 0. \quad (3.7)$$

The equivalence of the two problems has been known since the 1950's, and the reformation is done by the transformation: $\mathbf{x} \Leftrightarrow (\mathbf{u}; \mathbf{v})$; $\mathbf{c} \Leftrightarrow (1, 1)$; $A \Leftrightarrow (\Phi, -\Phi)$; $\mathbf{b} \Leftrightarrow \mathbf{y}$ [CDS98].

Once the basis pursuit is transformed, there are multiple ways to solve the associated LP. Two ways are the simplex and interior-point (IP) methods. Both methods, look at feasible points of a convex polyhedron called a simplex. The simplex method walks around the edge of the simplex moving to the next edge node if objective function value, $\mathbf{c}^T \mathbf{x}$, is improved. The interior-point method starts in the center of the simplex and works its way out to the outermost nodes [CDS98]. The interior-point method has become the predominant method used to solve LP. However, there are multiple interior-point methods that can be used to solve the same problem. For example, one

method of IP requires the explicit construction of $\Phi^T\Phi$, but another does not [FNW07]. The research on optimization can be categorized by the reformation of the original optimization problem and solution methods that try to minimize memory usage while producing faster results.

Several of the reformations that have been developed for the above problem are as follows:

quadratically constrained linear program (QCLP),

$$\min_{\tilde{\mathbf{x}} \in \mathbb{R}^N} \|\tilde{\mathbf{x}}\|_1 \quad \text{subject to} \quad \|\mathbf{y} - \Phi\tilde{\mathbf{x}}\|_{\ell_2}^2 \leq \epsilon, \quad (3.8)$$

quadratic program (QP),

$$\min \|\mathbf{y} - \Phi\tilde{\mathbf{x}}\|_{\ell_2} \quad \text{subject to} \quad \|\tilde{\mathbf{x}}\|_1 \leq t, \quad (3.9)$$

and *bound-constrained quadratic program (BCQP),*

$$\min \frac{1}{2} \|\mathbf{y} - \Phi\tilde{\mathbf{x}}\|_{\ell_2}^2 + \tau \|\tilde{\mathbf{x}}\|_1, \quad (3.10)$$

for some nonnegative parameters τ, ϵ , and t [FNW07].

Each of these reformations has multiple algorithms that generate a solution. The QCLP reformation can be reworked as a second order cone program, and QP can be solved using a statistical method known as least absolute shrinkage and selection operator (LASSO). However, each of these reformations can be shown to achieve the same resulting $\tilde{\mathbf{x}}$, though each have different limitations [FNW07].

The LASSO method is a homotopy algorithm ¹ that finds the full path of solutions. In homotopy methods, pivoting operations are performed involving

¹Homotopy optimization algorithms are ones that impose a continuous homotopy function on the original variables plus a homotopy variable. When the homotopy variable is active, the function is equals the original function trying to be optimized or the function is homotopic to the original function.

sub-matrices of Φ and $\Phi^T\Phi$. In sparse reconstruction, the method requires at least as many pivoting operations as nonzero elements in the solution. The formation of $\Phi^T\Phi$ and sub-matrices takes an enormous amount of memory for a large signal that may have multiple nonzero elements. Therefore, homotopy methods tend to take as much memory and computational time as LP methods [FNW07].

Reformulations like QCLP and BCQP allow for a small amount of error in reconstruction. This error can be interpreted as some noise. In particular, the original work on Basis Pursuit proposed a Basis Pursuit De-Noising with a similar optimization equation [FNW07]. In BCQP, τ works as a regularization parameter that aids in producing a sparse signal in addition to denoising. One way of choosing τ is to consider soft-thresholding in an orthonormal basis. If Φ was an orthogonal matrix, then a solution $\tilde{\mathbf{y}} = \Phi^T \mathbf{y}$ would exist. In addition, the model can be examined as an ill-posed inverse problem with Tikhonov regularization [KKLB07]. The problem modeled with this form of regularization can be shown to have finite convergence to zero as $\tau \rightarrow \infty$. In addition, $\tau_{max} = \|\Phi^T \mathbf{y}\|_\infty$. Therefore, the value of τ should be taken as some fractional value of $\|\Phi^T \mathbf{y}\|_\infty$ based on the order of the noise to achieve convergence to a sparse nonzero solution [KKLB07].

One such method for solving BCQP is known as Gradient Projection for Sparse Signal Algorithm. This algorithm uses a gradient projection applied to a quadratic program in which the search path for each iteration is obtained by projection onto the feasible set in the negative gradient direction.

Algorithm 2 Gradient Projection [FNW07]

Let $\tilde{\mathbf{x}} = \mathbf{u} - \mathbf{v}$, $\mathbf{u} \geq 0$, $\mathbf{v} \geq 0$. The BCQP can be rewritten as,

$$\min_{\mathbf{z}} \mathbf{c}^T \mathbf{z} + \frac{1}{2} \mathbf{z}^T B \mathbf{z} := F(\mathbf{z}), \quad (3.11)$$

$$\text{such that } \mathbf{z} \geq 0, \mathbf{z} = \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix}, \mathbf{b} = \Phi^T \mathbf{y},$$

$$\mathbf{c} = \tau \mathbf{1}_{2n} + \begin{bmatrix} -\mathbf{b} \\ \mathbf{b} \end{bmatrix}, \text{ and } B = \begin{bmatrix} \Phi^T \Phi & -\Phi^T \Phi \\ -\Phi^T \Phi & \Phi^T \Phi \end{bmatrix}.$$

PROCEDURE:

- **Initialize:** Given \mathbf{z}^0 , choose parameters $\beta \in (0, 1)$ and $\mu \in (0, 1/2)$; set $k = 0$.

- **Compute:** $\alpha_0 = \arg \min_{\alpha} F(\mathbf{z}^k - \alpha \mathbf{g}^k)$
 where $\mathbf{g}^k = \begin{cases} (\nabla F(\mathbf{z}^k)) & \text{if } \mathbf{z}^k > 0 \text{ or } (\nabla F(\mathbf{z}^k)) < 0 \\ 0 & \text{otherwise} \end{cases}$

and replace α_0 by $\text{mid}(\alpha_{min}, \alpha_0, \alpha_{max})$.

- **Backtracking Line Search:**

Choose α^k to be the first number in the sequence $\alpha_0, \beta \alpha_0, \beta^2 \alpha_0, \dots$ such that

$$F((\mathbf{z}^k - \alpha^k \nabla F(\mathbf{z}^k))) \leq F(\mathbf{z}^k) - \mu \nabla F(\mathbf{z}^k)^T (\mathbf{z}^k - (\mathbf{z}^k - \alpha^k \nabla F(\mathbf{z}^k)))$$

and set $\mathbf{z}^{k+1} = (\mathbf{z}^k - \alpha^k \nabla F(\mathbf{z}^k))$.

- **Test:** Perform convergence test and terminate with approximate solution \mathbf{z}^{k+1} if it is satisfied; otherwise, set $k \leftarrow k + 1$ and repeat.
-

The above algorithm, along with other forms of optimization recovery, are strong tools for reconstruction. They have the advantage of smaller errors than greedy based pursuit methods without the need to know the desired degree of sparsity. However, the speed of computation of matching pursuit cannot be matched by optimization

[Tro04]. However, the next section addresses a very fast sublinear method for reconstructing when sampling using a graph based sensing matrix.

3.3 Bucket Recovery

Recovery based on the graph structure of the sensing matrix, Φ , is a greedy algorithm that is based on the probability structure of the sensing matrix. This algorithm can be generalized to the larger set of ϵ -expander graphs. However, by restricting the set of graphs to be bipartite $(\alpha N, \beta d)$ expander graphs with $\beta = \frac{3}{4}$ and $\alpha N \geq 2k$, the probability element of reconstruction is dropped, since the probability was based on the graph [JXHC08]. If the matrix, Φ , is an expander graph with the above parameters, then the below algorithm will yield a $\tilde{\mathbf{x}}$, such that $\Phi\tilde{\mathbf{x}} = \Phi\mathbf{x}$ where \mathbf{x} is the original vector.

Algorithm 3 Bucket Recovery [JXHC08]

- Initialize $\tilde{\mathbf{x}} = 0_{N \times 1}$.
 - Loop Until $\mathbf{y} = \Phi\tilde{\mathbf{x}}$.
 - Find a variable, $\tilde{\mathbf{x}}_j$ such that $g_i = y_i - \sum_{j=1}^N \Phi_{ij}\tilde{\mathbf{x}}_j$ have at least $\frac{d}{2}$ have identical values.
 - $\tilde{\mathbf{x}}_j = \tilde{\mathbf{x}}_j + g$. of the d measurement
-

The figure, 3.1, shows a solution of a variable. The ability to find at least $\frac{d}{2}$ identical gaps comes from the constants that are opposed on the expander graph.

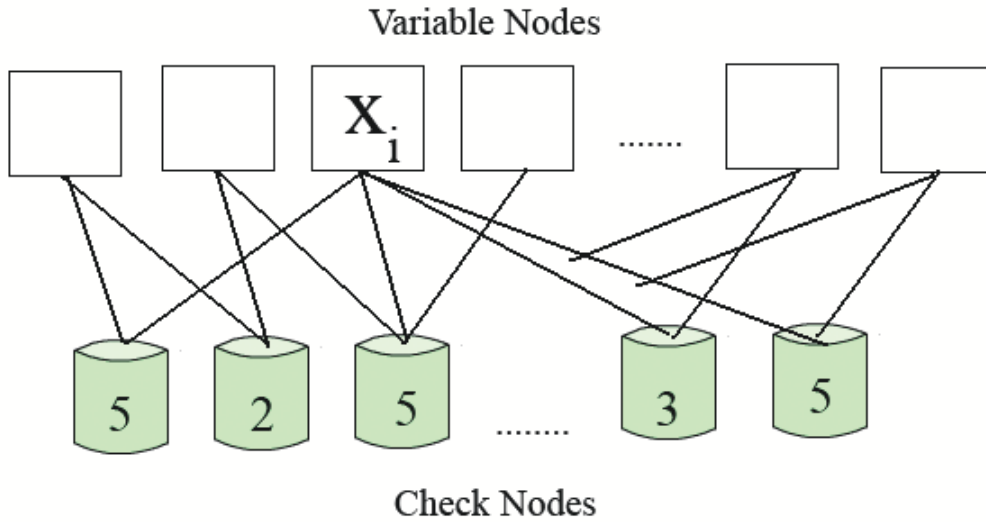


Figure 3.1: Figure demonstrating graph based recovery using buckets.

Theorem 3.3.0.1. [XH07]

When $\tilde{\mathbf{x}} = \mathbf{0}$, $\mathbf{y} \neq \Phi \tilde{\mathbf{x}}$ and $k \leq \frac{\alpha N}{2}$, there always exists a variable node such that $d' > \frac{d}{2}$ of the measurement equations it participates in has identical nonzero gaps.

While the above condition does provide recovery, a stronger parameter must be imposed for a unique recovery. This can be achieved by using an expander in which $\alpha N \geq 3k$. In particular, the uniqueness is equivalent to the questioning if the adjacency matrix has a null vector that is $3k$ sparse [JXHC08].

In addition, the run time of the algorithm is $O(dk)$, which is much smaller than most other recovery methods, since the recovery only grows with d and k , which are smaller than N [JXHC08]. A modified version of this algorithm along with several others from this chapter will be explored numerically in the next chapter and then compared.

Chapter 4

Numerical Results

In order to better understand the current research in compressive sensing, numerical experiments were conducted. These numerical experiments are designed to explore multiple sampling techniques based on several matrices forms, recovery techniques, and signal types. In doing such experimentation, two goals exist. The first goal was to provide empirical evidence of the findings reported in several articles. The second is to have a better understanding of the constants related to the asymptotic bounds of the number of required linear measurements. These two goals are designed to investigate the very practical limits of the theorems and algorithms covered in the expository overview.

The experimentation that demonstrates the results of current publications can be broken into two separate sets based on the form of the original signal, \mathbf{x} . The two forms are a one dimensional discrete signal, or vector, i.e., $\mathbf{x} \in \mathbb{R}^N$, and signals generated from a digital image.

The experimentation that tests the bounds on the minimum number of measurements needed for reconstruction was performed on binary and Gaussian random sensing matrices for multiple degrees of sparsity and original lengths. All signal used in these test are discrete vectors with k spikes, such that $-300 \leq \mathbf{x}_i \leq 300$

for $i = \pi(1), \pi(2), \pi(3), \dots, \pi(k)$. Recovery using linear programming and graph based reconstruction techniques were done for experiments using binary graph based sensing matrices, while the signals sampled using Gaussian random matrix were recovered using only linear programming.

Technique Specification

All numerical experimentation was conducted using MATLAB ®(2008b, MathWorks, Natick, Massachusetts). The machine the timing data was collected on had a AMD ML-30 CPU running Windows XP SP2 with 2GB of DDR RAM. The ML-30 had a maximum frequency of 1.6 GHZ, a 128KB L1 cache, and a 1MB L2 cache. The experimentation was run on a lower end machine for the time period. However, the use of a single core machine is important when timing algorithms that may want to be moved to hardware later, since MATLAB has started to use multi-core threading into some function calls.

Bounding result data was collected on multiple systems due to the over 10,000 reconstructions needed to be computed. These system included CyberStar at The Pennsylvania State University Institute for CyberScience, NSF Award Number 0821527, and Server3 at Duquesne University's Department of Mathematics and Computer Science.

4.1 Publication Comparison Results

Vector Experiments

In the vector based experiments, two forms of $k - sparse$ vectors of length N were examined. The first vector was a random binary vector with a maximum of k spikes of height one. The second vector was a random vector with a maximum of k spikes

with heights that ranged randomly in the interval ± 300 , such that $-300 \leq \mathbf{x}_i \leq 300$ for $i = \pi(1), \pi(2), \pi(3), \dots, \pi(k)$.

Three sensing matrices were used for each signal. The first matrix used was a Gaussian random matrix. The same Gaussian random matrix was used in all vector based tests to ensure that the matrix had the Restricted Isometry Property.

A second matrix of pseudo-random nature was also used. This matrix used a similar construction method based on bipartite expander graphs as defined in *Efficient Compressive Sensing with Deterministic Guarantees Using Expander Graphs* [XH07]. This matrix was freshly constructed for each of the trails. It was the goal in reconstructing this matrix to try to demonstrate two important characteristics of expander graph based matrices. Firstly, the probability of constructing a matrix in this fashion for sampling in which the original signal can be reconstructed from its measurements using linear programming is high. Secondly, the probability of constructing a matrix that can be recovered by methods that uses the graph structure is lower than that of reconstruction with linear programming [JXHC08].

The third matrix that was used for sampling was based on a deterministic construction using finite fields. This construction method was outlined by DeVore [DeV07]. This matrix construction has been shown to have the best compression ratio of a sparse deterministic construction containing only values of 0 and 1 can achieve while satisfying RIP-2. However, the use of this matrix restricted the vector size, N , and the number of measurements, n , to powers of a prime number [DeV07]. In these tests, the prime number 17 was used as the base, which resulted in $N = 17^3$ (4913) and $n = 17^2$ (289).

Two optimization based recovery method were used for all three matrices. The first method was a simple linear programming optimization that implemented the transformation that was introduced in the chapter on reconstruction and the built in MATLAB function `linprog`. The medium-scaling algorithm of `linprog` was used due to

the requirement that the matrix be sparse for large-scaling algorithm. Therefore, the Gaussian random matrix limited the choice. The linprog function was tried using both the normal simplex and projection method, which is based on simplex algorithm. In general pretesting, the projection method performed better than the standard simplex algorithm. Therefore, the measurements of the projection method are given.

The second optimization recovery method that was used in reconstruction was a gradient projection method which was also outlined in the chapter on reconstruction.

In addition to optimization based recovery methods, an algorithm based on the bipartite expander graph structure was used for tests that used a pseudo-random sensing matrix. This method, as outlined in the chapter on recovery, is a bucket based algorithm that tests combinations of elements in the measurement vector to determine the best bucket, i.e., \mathbf{x}_i , to place the value.

Digital Image

The experimentation on digital images was conducted on a 64×64 pixel square gray scale image. The 2-Dimensional Daubechies Wavelet-4 of level 3 was first applied to the image. After, the data was compressed using global thresholding. In global thresholding, one value is selected, and all measurements that are smaller than the threshold are replaced with the value zero. The global thresholding method that was implemented picked a threshold so that the percentage of zero elements in the coefficient vector was large enough for using compressive sensing techniques. A very similar method is used to compress an image by JPEG200 5/3, which rounds values of the Cohen-Daubechies-Freaveau Wavelets [GW08].

In particular, the compressive sensing techniques were implemented on the coefficient of the wavelet transformation used a matrix built using expander graphs. Linear programming and bucket based algorithms were used for reconstruction. The reason for the use of this sensing matrix was to limit the memory size so that it could

be stored in cache level memory. This would provide better timing because of latency and transportation costs per calculation. In addition, DeVore’s construction was not used because the sensing matrix limited the vectors used to those with a length that are powers of prime numbers.

4.1.1 Vector Results

Three degrees of sparsity were used for both vector forms. The degrees of sparsity were based on the asymptotic bounds of the minimum number of measurements given for each matrix. The minimum of the three bounds was produced by DeVore’s construction, which was $k \leq C\sqrt{n} \log(n) / \log(\frac{N}{n})$. If $C = 1$, then $k \lesssim 33$ [DeV07]. Therefore, the values of k that were used were $k = 15, 25, 35$. Each of these experiments was run for thirty trials.

The first row of the numerical results below states the recovery method used. The second row defines the matrix that was used. These matrices were DeVore’s construction, Random using a Gaussian random matrix, and Expander which used a pseudo-random matrix built on expander graphs.

Binary	LP	LP	LP	Gradient	Gradient	Gradient	Bucket
K=15	DeVore	Random	Expander	DeVore	Random	Expander	Expander
Average MSE	3.18E-18	3.8E-18	.868E-18	2.81E-6	5.30E-6	2.4E-6	9.07E-4
Time (Seconds)	12.6	98.6	1.423	1.649	.79	1.544	14.37

Binary	LP	LP	LP	Gradient	Gradient	Gradient	Bucket
K=25	DeVore	Random	Expander	DeVore	Random	Expander	Expander*
Average MSE	7.23E-19	7.23E-19	7.23E-19	6.73E-6	15.4E-6	19.49E-6	5.6E-3
Time (Seconds)	2.4	230	1.4	2.8	1.17	3.03	21.6839

Binary	LP	LP	LP	Gradient	Gradient	Gradient	Bucket
K=35	DeVore*	Random*	Expander*	DeVore*	Random*	Expander*	Expander*
Average MSE	5.78E-19	8.67E-19	14.4E-19	1.66E-5	3.67E-5	3.46E-5	1.02E-2
Time (Seconds)	2.4	103	1.42	.77	1.966	5.35	35.9

Table 4.1:

The table contains the results of tests using signals with original size 17^3 and K randomly placed 1s that were compressed to the size 17^2 . The first row references the reconstruction method used as follows: **LP**: *linear programming*, **Gradient**: *gradient projection method*, and **Bucket**: *graph based reconstruction*. The second row references the sensing matrix that was used in each test as: **DeVore**: *deterministic matrix based on finite fields*, **Random**: *Gaussian random matrix*, and **Expander**: *adjacency matrix of an expander graph*. An asterisk, *, identifies methods that failed to identify the location of all spikes in the signal.

Between ± 300	LP	LP	LP	Gradient	Gradient	Gradient	Bucket
K=15	DeVore	Random	Expander	DeVore	Random	Expander	Expander
Average MSE	7.9E-14	0	0	1.266	.7713	.5832	7.9E-15
Time (Seconds)	8.18	98.84	1.49	1.6107	.56	1.27	10.31

Between ± 300	LP	LP	LP	Gradient	Gradient	Gradient	Bucket
K=25	DeVore	Random	Expander	DeVore	Random	Expander	Expander
Average MSE	2.37E-14	2.37E-14	0	3.27	1.7714	2.5525	1.85E-14
Time (Seconds)	3.18	112.22	1.69	2.26	.8727	1.6351	18.31

Between ± 300	LP	LP	LP	Gradient	Gradient	Gradient	Bucket
K=35	DeVore*	Random*	Expander*	DeVore*	Random*	Expander*	Expander*
Average MSE	3.55E-14	3.55E-14	2.37	6.5418	3.62	7.248	16.187
Time (Seconds)	3.59	127.52	1.93	2.623	1.21196	3.53	22.844

Table 4.2:

The table contains the results of tests using signals with original size 17^3 and K randomly placed spikes of values in the range of ± 300 that were compressed to the size 17^2 . The first row references the reconstruction method used as follows: **LP**: *linear programming*, **Gradient**: *gradient projection method*, and **Bucket**: *graph based reconstruction*. The second row references the sensing matrix that was used in each test as: **DeVore**: *deterministic matrix based on finite fields*, **Random**: *Gaussian random matrix*, and **Expander**: *adjacency matrix of an expander graph*. An asterisk, *, identifies methods that failed to identify the location of all spikes in the signal.

As noted from the data, only several combinations failed. It is worthy to note that when the number of measures, n , is increased from 289 to 512, all methods have

perfect recovery. This would be the difference of only moving the compression ratio from 5.9 to 10.4 percent. In addition, one can make the unique observation from the data that when the range of values for the spikes are changed, the bucket recovery methods works even with a smaller number of measurements. One reason for this unexpected result is the distinction between values in the buckets and the number that corresponds to the bucket placement. This error is related to using a linear algorithm that is based on integers on a compact set. One solution to this problem is to scale the error term in the algorithm. However, the scaling of the error term ϵ would depend on N, n, k , and the interval, which would make it difficult to predict. In addition, the original signal would have to be examined to find the heights of the spikes. This would be undesirable, since compressive sensing is powerful because it needs to know very little about the original signal.

The changes between the binary and the larger spikes also changed the general average mean squares error, MSE. Even if the reconstruction identifies the spike, the difference in the size of the two spikes can be much larger. This can be seen as the average MSE has increased in all combinations except the one that used bucket reconstruction for the larger spike tests. This scaling of error to spike size is one reason that MSE is not the sole deciding factor of the performance. However, visual assessment is not always a good measurement as well. The figure below, figure 4.1, shows a failed reconstruction with the regularization parameter $\tau = .1$. At first view, the data seems to be correctly recovered. However, many of the smaller spikes are missing or misplaced. The MSE of this figure was about 9 for both DeVore and Expander matrices and 4 for the Random matrix.

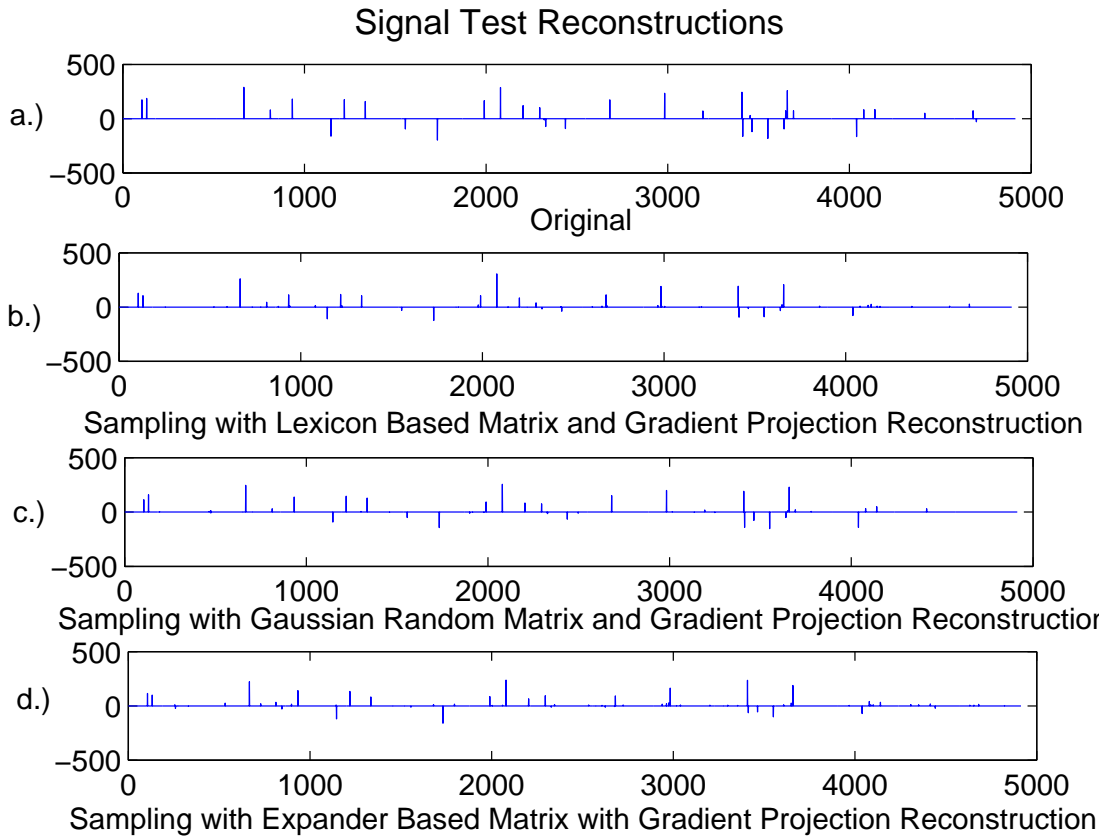


Figure 4.1: The above figure demonstrates the need to perform both a MSE measurement and a visual assessment for reconstruction using a signal of original length 17^3 with 35 randomly placed spikes that were compressed to a length of 17^2 using three different sensing matrices.

4.1.2 Digital Image Results

The image that was chosen for the experimentation was a gray scale shading image. Since compressive sensing provides perfect reconstruction if the degree of sparsity and number of measurements are met, the experimentation focused not on clarity of the reconstruction but on the number of measurements needed. The 64×64 pixel image resulted in a vector of 5782 coefficients after the 2D wavelet transformation. Thresholding retained 85.73 percent of the original image by a global

norm while producing a coefficient vector where 99.6 percent of the elements are zeros. This led to a fixed sparsity, k , of 196 for the image experiments. The experiment was done multiple times for a multitude of measurements, m . The results showed a boundary around 1320 measurements that prevented the image from being recovered.

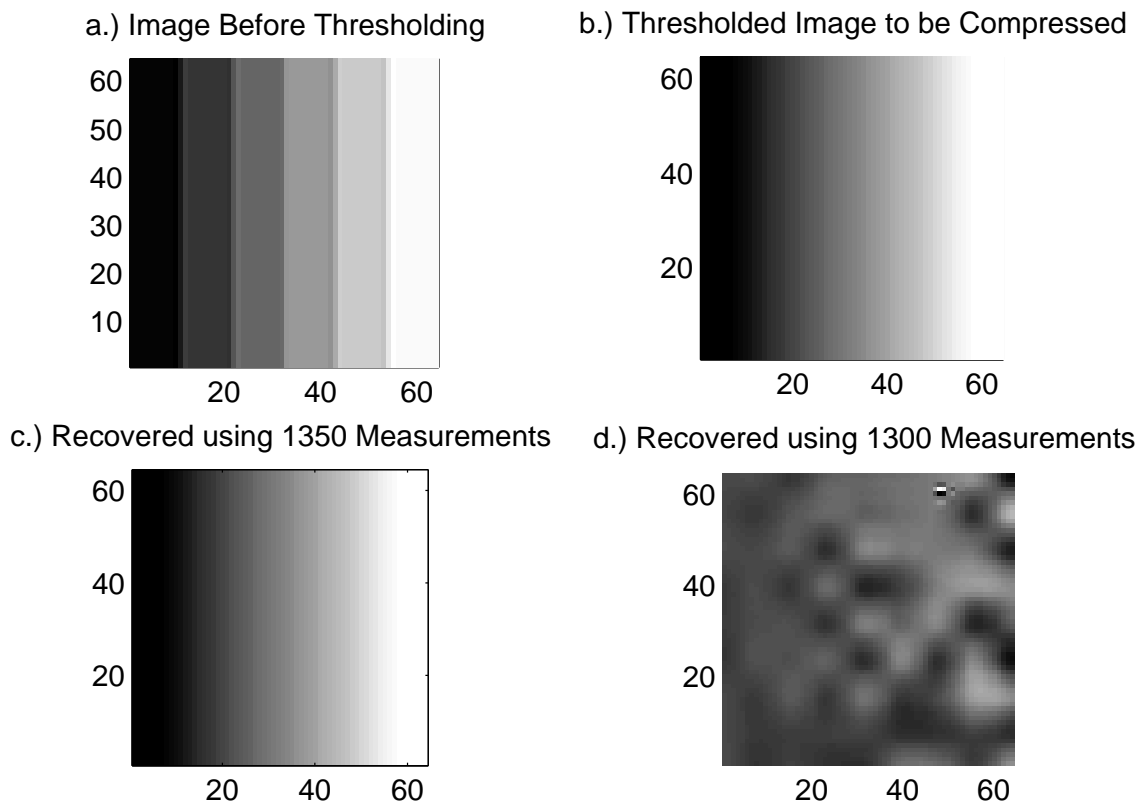


Figure 4.2: The figure depicts the use of compressive sensing on a 64×64 pixel image. a.) Depicts the image before using thresholding in the wavelet domain. b.) Depicts the image after thresholding was performed in the wavelet domain. This image is the one that is compressed. c.) Depicts a perfect reconstruction of the thresholded image, b. d.) Depicts a failed reconstruction attempt of the thresholded image, b.

The time needed to recover the imaged was dependent on the number of measurements. The cases that failed had very fast times, while the times for the recovered image was between 23 seconds for n close to 1320 and up to 3 minutes for n

around 2000. When the number of measurements was greater than the boundary of 1320 by around 150 but less than an increase of 500, the recovery time was less than the the time of trials closer to the boundary and farther away from the boundary. The surprise was how the slight increase from a vector size of 17^3 to 5782 was able to scale and provided better data on the scaling.

4.2 Bounds

One of the main objectives of performing the above experimentation was to find results on the implicit constants that are related to the asymptotic bounds for minimum sketch length need for recovery. The below table summarizes the publication data.

Sensing Matrix	Recovery Method	Sketch Length	Cite
Binary	LP	$O(k \log(\frac{N}{k}))$	[JXHC08]
Binary	Graph Based	N.A.	[JXHC08]
Gaussian Random	LP	$O(k \log(N))$	[BDDW07]

Table 4.3: The above table displays the minimum sketch lengths for a signal using a certain sensing matrix and recovery method.

The data was collected for each of the combinations in the table above. Each experiment was performed thirty times, and the average is reported in the graphs below. In figure 4.4, curves were fitted to each data set using the model $c \cdot k \cdot \log(\frac{N}{k}) + b$, where c and b are constant parameters of the curve.

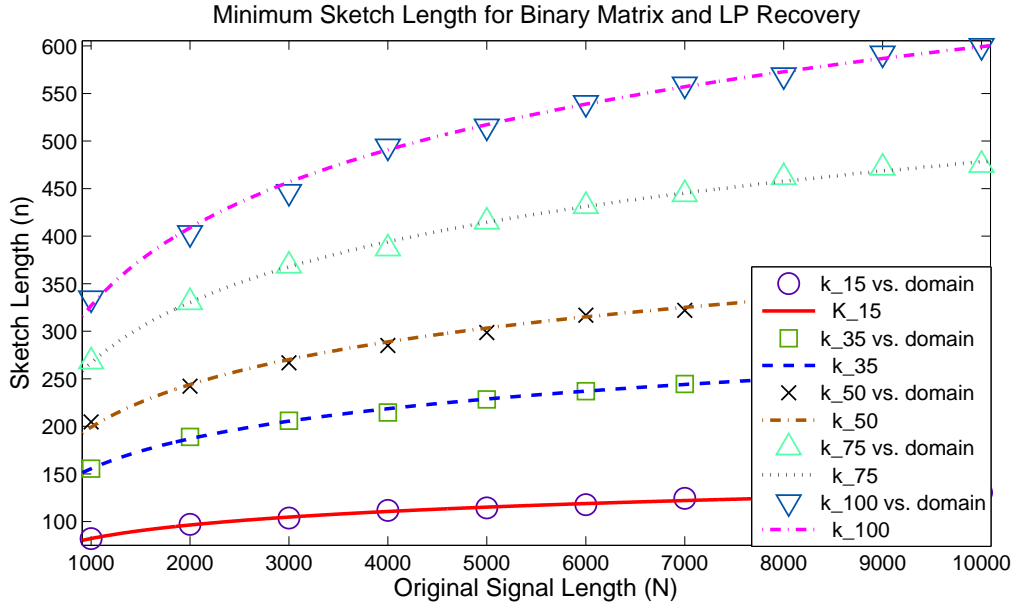


Figure 4.3: The figure plots the minimum number of measurements needed to ensure recovery when performing compressive sensing. Each line represents a different degree of sparsity that is plotted along the domain of the original signal length.

The parameters for each fitted curve is tabulated below. From the data, the constants in front of the dominate contributing factor, c , tend to be small. In addition, c decreases as k increases. This relationship demonstrates the signal is more influential in determining the minimum number of measures needed than any constant factor.

k	c	b	R-Squared
15	1.37	-4.04	.99
35	1.30	2.90	1.00
50	1.30	5.08	.99
75	1.22	29.26	1.00
100	1.18	54.41	1.00

Table 4.4: The above is a table of the parameters of the fitted curve, $c \cdot k \cdot \log\left(\frac{N}{k}\right) + b$, for a binary sensing matrix using LP recovery, where N is the original signal length and k is the degree of sparsity. In addition, the r-squared value is recorded to give some level of how well the curve fitted the data points.

The figure below, 4.4, plots the data for the experiments that were run with a binary sensing matrix with graph based recovery. As seen in the graph, the number of measurements does not increase as the size of the original signal increases. Though the initial sketch size is larger than sensing recovered with linear programming, sketch length does not have to be increased. However, there is a lack of data to conclude that the sketch length is a constant related to only the sparsity of the original vector. Indeed, other models, such as one that would use the \log^* function, may provide a fitted curve for sketch length.

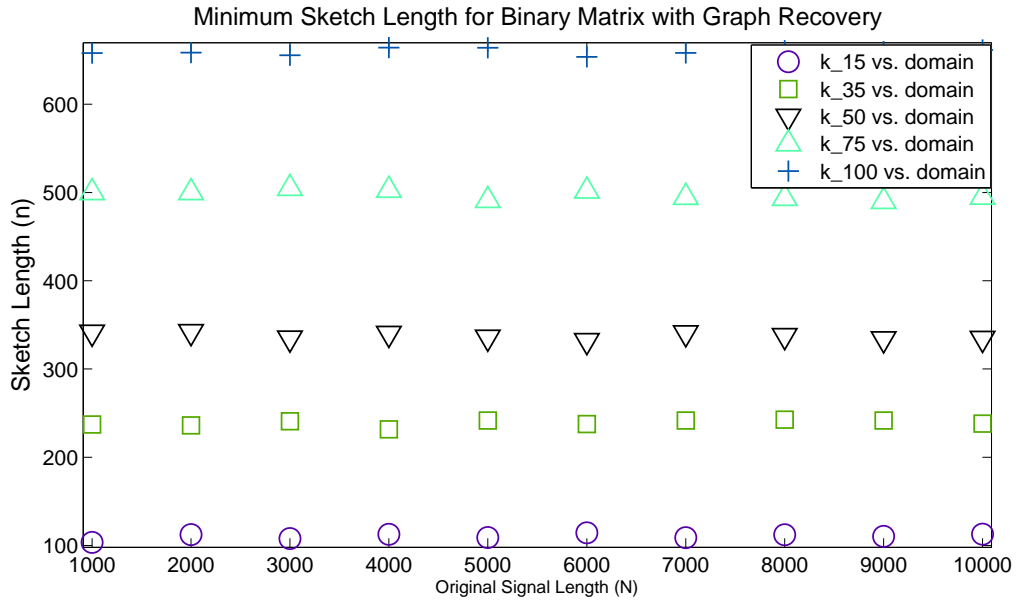


Figure 4.4: The figure plots the minimum number of measurements needed to ensure recovery when performing compressive sensing. Each symbol represents a different degree of sparsity that is plotted along the domain of the original signal length.

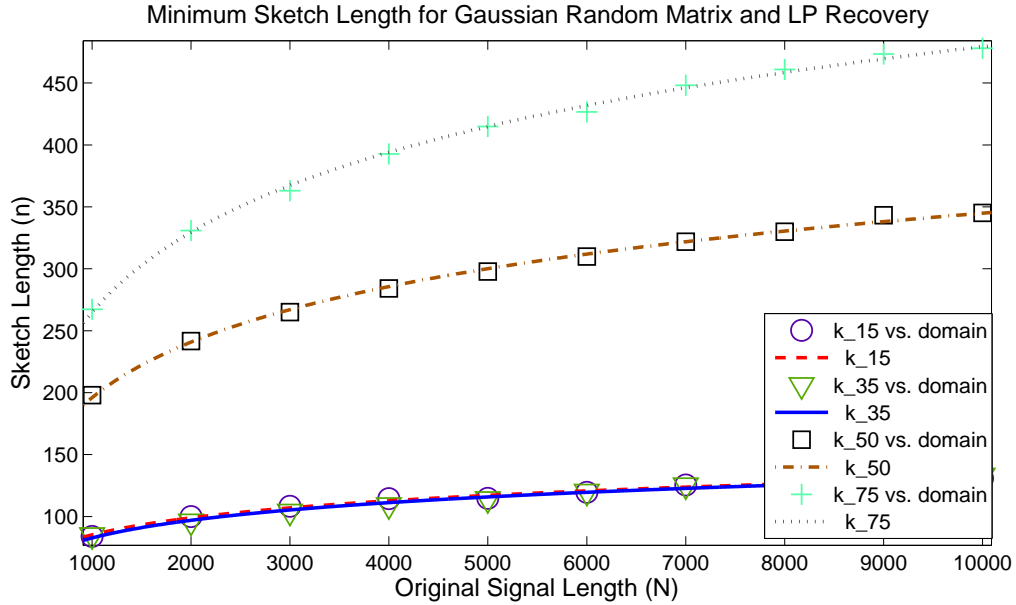


Figure 4.5: The figure plots the minimum number of measurements needed to ensure recovery when performing compressive sensing. Each line represents a different degree of sparsity that is plotted along the domain of the original signal length.

k	c	b	R-Squared
15	1.32	-51.55	.99
35	.59	-59.39	.98
50	1.29	-250.6	1.00
75	1.24	-377.8	1.00

Table 4.5: The above is a table of the parameters of the fitted curve, $c \cdot k \cdot \log(N) + b$, for a Gaussian random sensing matrix using LP recovery where N is the original signal length and k is the degree of sparsity. In addition, the r-squared value is recorded to give some level of how well the curve fitted the data points.

In the last figure of these results, the data points and the fitted curves for the Gaussian random sensing matrix with linear programming are displayed. These

results are similar to those of the binary sensing matrices when comparing the size of the model parameter, c . However, the sketch size itself is much smaller than the minimum sketch size needed for binary sensing matrices. This is a similar result found in publication. The decrease in sketch length, though, comes at the high price of storage and an increase in time needed to recover. For $k = 50$, the average recovery time in MATLAB was about 100 times longer for sensing using Gaussian matrices.

4.3 Discussion

In the results above, empirical evidence was presented to demonstrate the usefulness of compressive sensing and its performance. The vector data showed how sensing matrix choice along with recovery may change the results and performance. These demonstrate results in current publications. Also they demonstrate the increase sketch size needed when using binary sensing matrices compared to the sketch size needed for Gaussian random matrices. However, the storage problem of using these large dense matrices, such as the Gaussian random matrix, and their higher computational cost in terms of time is also noted. This was echoed in the digital image section along with the section on bounds.

The data from all three sections demonstrates the play of constants and their dependence over multiple variables, such as signal length, sparsity, and recovery method. In particular, the constants that affect the number of measurements were shown to be small. In the section on bounds, the constant, c , of the curve fitting model of both binary and Gaussian random metrics was smaller than 2.

Chapter 5

Conclusion and Future work

This work provides an overview of sensing matrices and recovery techniques used in compressive sensing. The first chapter while providing some general introduction and notation outlines core past publications in the area. These publications are used in conjunction with several other publications to provide an expository overview of sensing matrices in chapter two and recovery techniques in chapter three.

The overview of sensing matrices explores two different construction analysis methods. The first uses a combination of geometric and probabilistic distributions to show that a reduction in dimensions could be achieved on a discrete sparse signal by taking a number of random linear projections. In addition, this analysis shows this number of random linear projections could be used to reconstruct the original sparse signal if the sensing matrix used for the projections satisfied the Restricted Isometry Property. The second construction analysis uses expander graphs. This construction technique generates sparse sensing matrices that are the adjacency matrices of expander graphs. A theorem that is similar to but not equivalent to the Restricted Isometry Property provides a guarantee for recovery. Both construction analysis methods have advantages and disadvantages. Sensing matrices generated that satisfy the Restricted Isometry Property tend to require less measurements to guarantee

recovery. However, these matrices tend to be dense which makes storage and reconstruction more difficult.

Multiple methods of reconstruction for both dense and sparse sensing matrices are covered in chapter three. The chapter starts by laying out a family of pursuit methods. The orthogonal matching pursuit method leads naturally to ℓ_1 based optimization. The section on optimization discusses linear programming and other methods. These other methods all have a list of trade offs, including the ability to tune the method to a given application or level of noise. The third recovery method which can be used only on sensing matrices generated using graphs demonstrate a fast recovery method. Several of the recovery methods are included in the numerical results chapter.

The numerical results chapter provides constants relating to the asymptotic bounds on the number of measurements needed for recovery along with multiple examples of the use of compressive sensing on signals and images. Though the details of the findings are related to the sensing matrix and reconstruction technique, the findings report a constant of no more than 2 for standard combinations.

These chapters give a good starting point for further exploration into compressive sensing and its multiple applications. Future work can be done in exploring and following up on the odd results relating to the numerical bounds on expander graph based sensing matrices recovered using the bucket method. Publications in the areas of combinatorial analysis and group testing would provide some of the theory needed for such an analysis [GSI08]. In addition, these chapters could be used to provide the background necessary for such topics as multi-channel compressive sensing which is related to active research in MRI and networks [WCA08].

As compressive sensing becomes a highly researched topic, these chapters will give a good framework of the underlining ideas for any future work that may come from compressive sensing.

Bibliography

- [BDB07] Petros Boufounos, Marco F. Duarte, and Richard G. Baraniuk, *Sparse signal reconstruction from noisy compressive measurements using cross validation*, Statistical Signal Processing, SSP '07. IEEE/SP 14th Workshop on, Aug. 2007, pp. 299–303.
- [BDDW07] Richard Baraniuk, Mark Devenport, Ronald DeVore, and Michael Wakin, *A simple proof of the restricted isometry property for random matrices*, Constr. Approx **2008** (2007).
- [BGI⁺08] R. Berinde, A.C. Gilbert, P. Indyk, H. Karloff, and M.J. Strauss, *Combining geometry and combinatorics: A unified approach to sparse signal recovery*, Communication, Control, and Computing, 46th Annual Allerton Conference on, Sept. 2008, pp. 798–805.
- [BM01] D. Burshtein and G. Miller, *Expander graph arguments for message-passing algorithms*, Information Theory, IEEE Transactions on **47** (2001), no. 2, 782–790.
- [CDS98] Scott Shaobing Chen, David Donoho, and Michael Saunders, *Atomic decomposition by basis pursuit*, SIAM Journal on Scientific Computing **20** (1998).

- [CRT06] Emmanuel Candes, Justin Romberg, and Terence Tao, *Stable signal recovery from incomplete and inaccurate measurements*, Communications on Pure and Applied Mathematics **59** (2006), no. 8, 1207–1223.
- [CT05] Candes and Tao, *Decoding by linear programming*, IEEE Transactions on Information Theory **51** (2005).
- [DeV07] Ronald A. DeVore, *Deterministic constructions of compressed sensing matrices*, Journal of Complexity **23** (2007), no. 4-6, 918–925.
- [Don06] D.L. Donoho, *Compressed sensing*, IEEE Transactions on Information Theory **52** (2006), no. 4, 1289–1306.
- [FNW07] Mario Figueiredo, Robert Nowak, and Stephen Wright, *Gradient projection for sparse reconstruction*, Selected Topics in Signal Processing, IEEE Journal of **1** (2007), no. 4, 586–597.
- [GSI08] A.C. Gilbert, M.J. Strauss, and M.A. Iwen, *Group testing and sparse signal recovery*, Communication, Control, and Computing, 46th Annual Allerton Conference on, Sept. 2008.
- [GW08] Rafael C. Gonzalez and Richard E. Woods, *Digital image processing*, third ed., Prentice Hall, 2008.
- [JXHC08] Sina Jafarpour, Weiyu Xu, Babak Hassibi, and A. Robert Calderbank, *Efficient and robust compressed sensing using high-quality expander graphs*, ArXiv e-prints **0806.3802** (2008), Provided by the SAO/NASA Astrophysics Data System.
- [KKLB07] S. Kim, K. Koh, M. Lustig, and S. Boyd, *An interior-point method for large scaled l_1 -regularized least squares*, Selected Topics in Signal Processing, IEEE Journal of **1** (2007), no. 4, 606–617.

- [Str07] G. Strang, *Computational science and engineering*, Wellesley-Cambridge Press, 2007.
- [Sut75] W.A. Sutherland, *Introduction to metric and topological spaces*, Oxford Scinece Publications, 1975.
- [SWM08] Yoav Sharon, John Wright, and Yi Ma, *Computation and relaxation of conditions for equivalence between l_1 and l_0 minimization*, submitted to IEEE Transactions on Information Theory (2008).
- [TAG07] Joel A. Tropp, Anna, and C. Gilbert, *Signal recovery from random measurements via orthogonal matching pursuit*, IEEE Trans. Inform. Theory **53** (2007), 4655–4666.
- [TG07] J.A. Tropp and A.C. Gilbert, *Signal recovery from random measurements via orthogonal matching pursuit*, Information Theory, IEEE Transactions on **53** (2007), no. 12, 4655–4666.
- [Tro04] J.A. Tropp, *Greed is good: algorithmic results for sparse approximation*, Information Theory, IEEE Transactions on **50** (2004), no. 10, 2231–2242.
- [WCA08] T. Wan, N. Canagarajah, and A. Achim, *Compressive image fusion*, ICIP, 2008, pp. 1308–1311.
- [XH07] Weiyu Xu and B. Hassibi, *Efficient compressive sensing with deterministic guarantees using expander graphs*, Information Theory Workshop, IEEE, Sept. 2007, pp. 414–419.