Combinatorial Regression and Improved Basis Pursuit for Sparse Estimation

Thesis by

M. Amin Khajehnejad

In Partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy

California Institute of Technology
Pasadena, California

2012
(Defended February 2, 2012)

Acknowledgements

I would like to sincerely thank my adviser Dr. Babak Hassibi for his great support of my research endeavors, and all my collaborators and coauthors including Dr. Salman Avestimehr (Cornell University), Dr. Alex Dimakis (University of Southern California), Dr. Amir Khojastepour (NEC Laboratories America Inc), Dr. William Bradley (Analog Devices/Lyric Labs), Dr. Benjamin Vigoda (Analog Devices/Lyric Labs), Dr. Weiyu Xu, Samet Oymak, Juhwan Yoo and Matthew Thill. I enjoyed collaboration with these smart people. I am also thankful to the members of my candidacy and defense talk committee, Professors Joel Tropp, Yaser Abu Mostafa, P. P. Vaidyanathan, Alex Dimakis and Tracy Ho for the insightful comments on this dissertation.

Abstract

Sparse representations accurately model many real-world data sets. Some form of sparsity is conceivable in almost every practical application, from image and video processing, to spectral sensing in radar detection, to bio-computation and genomic signal processing. Modern statistics and estimation theory have come up with ways for efficiently accounting for sparsity in enhanced information retrieval systems. In particular, *compressed sensing* and *matrix rank minimization* are two newly born branches of dimensionality reduction techniques, with very promising horizons. Compressed sensing addresses the reconstruction of sparse signals from ill-conditioned linear measurements, a mathematical problem that arises in practical applications in one of the following forms: model fitting (regression), analog data compression, sub-Nyquist sampling, and data privacy. Low-rank matrix estimation addresses the reconstruction of multi-dimensional data (matrices) with strong coherence properties (low rank) under restricted sensing. This model is motivated by modern problems in machine learning, dynamic systems, and quantum computing.

This thesis provides an in-depth study of recent developments in the fields of compressed sensing and matrix rank minimization, and sets forth new directions for improved sparse recovery techniques. The contributions are threefold: the design of combinatorial structures for sparse encoding, the development of improved recovery algorithms, and extension of sparse vector recovery techniques to other problems.

We propose combinatorial structures for the measurement matrix that facilitate compressing sparse analog signal representations with better guarantees than any of the currently existing architectures. Our constructions are mostly deterministic and are based on ideas from expander graphs, LDPC error-correcting codes and combinatorial separators.

We propose novel reconstruction algorithms that are amenable to the combinatorial structures we study, and have various advantages over the conventional convex optimization techniques for sparse recovery. In addition, we separately study the convex optimization Basis Pursuit method for compressed sensing, and propose regularization schemes that expand the success domain for such algorithms. Our studies contain rigorous analysis, numerical simulations, and examples from practical applications.

Lastly, we extend some of our proposed techniques to low-rank matrix estimation and channel coding. These generalizations lead to the development of a novel and fast reconstruction algorithm for matrix rank minimization, and a modified regularized linear-programming-based decoding algorithm for detecting codewords of a linear LDPC code during an erroneous communication.

Table of Contents

List of Figures

List of Algorithms

Chapter 1

Introduction

Information theory teaches us that if data contains redundancy, it can be reliably compressed without losing the essential information content. Claude Shannon (1916–2001) quantified this statement by introducing the mathematical notion of information, and proposing a model for measuring it [Sha48]. In this model, information rate is somewhat equivalent to the measure of uncertainty in random data. Given this mathematical model, it is possible to accurately measure the uncertainty of a data source, or the joint information rate of data generated by various sources. Furthermore, using the same mathematical foundation, it is possible to quantify the reliability between the input and output of a communication channel. The former led to the development of *source coding*, which studies the limits and methods of reliably compressing data, and leads to ways for efficiently "storing" information. The latter laid the foundation for the field of *channel coding* that addresses reliable communication, or more generally "processing" data. In both cases, redundancy in the information content plays a major role, and information theory provides a way of quantifying and exploiting it. In source coding, one attempts to identify or predict the redundancy in the data and eliminate it. In channel coding, the goal is to add redundancy to the data so that the information content can be retrieved despite errors encountered while processing it.

1.1 Sparsity Is Common

A very common form of redundancy that exists in data extracted from natural events is "sparsity". The information content of many real-world signals is sparsely distributed, either in the original domain, or when projected via popular transforms. A signal can represent a database, a portfolio, a time-series function, a power spectral density, a probability distribution function, an array of numbers or any other form of *indexed* information derived from raw or processed data. The coarse definition of a sparse signal is a vector in which most of the nonzero entries are (almost) zero, and only a few coefficients are nonzero or significant. If the entries of a signal represent the energy content of an object, then a sparse signal has a highly unbalanced energy distribution. Sparse vector representations are conceivable in almost every practical application, from cosine or wavelet transforms of images and video frames, to spectral content of radar signals, to neural records, DNA micro-array read-outs, and fMRI images in genomic and biomedical applications. Further applications arise in sparse covariance matrices in dynamic systems, sparse principle component analysis (PCA) in machine learning, and portfolio representations in financial engineering and economics.

1.2 Sparse Recovery

Knowing that in many cases the signals we are interested in estimating or processing are sparse, a critical question arises. How can we benefit from this knowledge? The answer is we can potentially design better schemes for sampling or operating on sparse data with less complexity, larger noise margins, and looser storage requirements than normal methods which do not assume sparsity for the underlying signal. This approach has led to an explosion in the development and use of dimensionality reduction and sparse processing techniques in the past decade. In particular, the field

of *compressed sensing* has drawn a lot of recent attention and has evolved into an independent area of signal processing [ric]. Compressed sensing addresses the following *sparse linear inverse* problem:

Identify a sparse vector, given a set of *linear* combinations of the entries.

In other words, the objective is to identify a sparse vector from a set of linear measurements (observations). What makes this theory interesting is that for sparse signals, it is possible to successfully solve the inverse problem even when the number of linear observations is less than the number of unknowns, namely the dimension of the signal. In other words, it is possible to solve an under-determined linear system of equations provided that the unknown signal is sparse.

1.2.1 Mathematical Formulation

Suppose that $\mathbf{x} \in \mathbb{R}^n$ is a vector with at most $k << n$ nonzero coefficients. We call such a vector k-sparse. Assume that $\mathbf{x}$ is not directly observable, but is rather accessed through a succinct set of linear measurements in the form of:

$$\mathbf{y}^{m\times 1} = \mathbf{A}^{m\times n}\mathbf{x}^{n\times 1}, \tag{1.2.1}$$

where the number of measurements m is smaller than the ambient dimension of $\mathbf{x}$, i.e. $m < n$. The measurements can be noisy, in which case, we have:

$$\mathbf{y}^{m\times 1} = \mathbf{A}^{m\times n}\mathbf{x}^{n\times 1} + \mathbf{v}^{m\times 1}, \tag{1.2.2}$$

The set of linear equations in (1.2.1) is under-determined (see Figure 1.1). Therefore, there are many vectors $\mathbf{x}'$ that satisfy the above equation. If fact if, $\mathbf{x}'$ is the linear addition of $\mathbf{x}$ with an arbitrary vector $\mathbf{z}$ in the null space of $\mathbf{A}$ (i.e. $\mathbf{Az} = 0$), then $\mathbf{Ax}' = \mathbf{y}$, and $\mathbf{x}'$ is a solution of (1.2.1). However, it is hoped that the "sparse" solution of (1.2.1) is unique and can be determined. To see this, suppose that $\mathbf{x}$ and $\mathbf{x}'$ are two distinct vectors, both having at most k nonzero entries and satisfying:

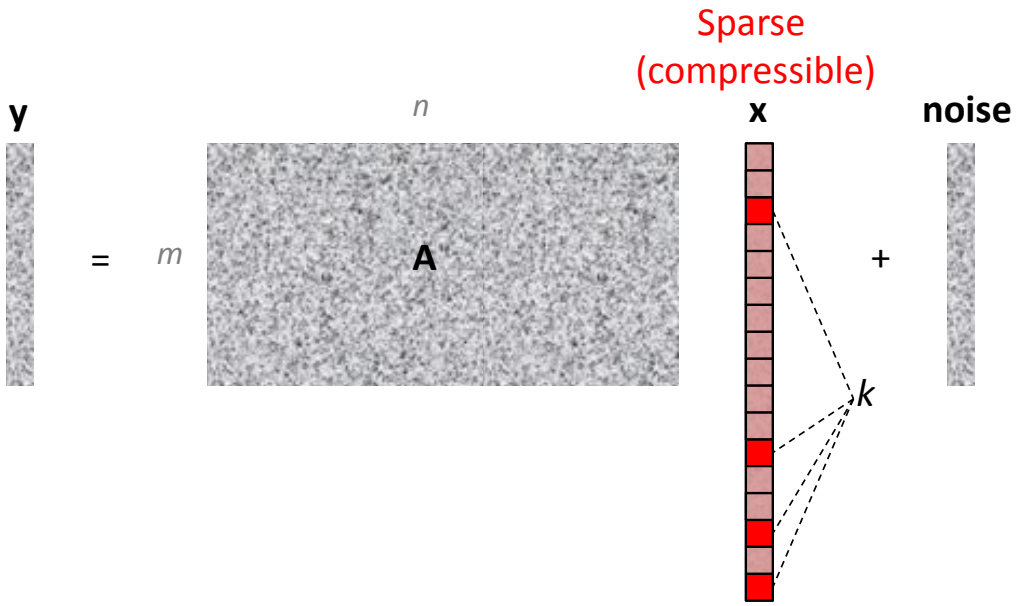


Figure 1.1: A pictorial demonstration of an under-determined system of linear equations acting on a sparse (compressible) vector

$$\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{x}' = \mathbf{y}. \tag{1.2.3}$$

Then, $\mathbf{A}(\mathbf{x} - \mathbf{x}') = 0$. However, the vector $\mathbf{x} - \mathbf{x}'$ has at most $2k$ nonzero entries. Therefore, to prevent the coexistence of two sparse solutions, it suffices to ensure that no $2k$ columns of $\mathbf{A}$ are linearly dependent. This is not very hard to guarantee if k is small enough. In fact, a simple dimension counting argument reveals that if the columns of $\mathbf{A}$ are chosen uniformly at random on the unit sphere in $\mathbb{R}^m$, then as long as $k \leq m/2$, this happens with probability 1; Every $2k$ columns of $\mathbf{A}$ are linearly independent and thus every k-sparse solution of (1.2.1) is the only k-sparse solution.

The same argument holds when the original signal is not sparse per se, but is known to be sparse over some linear "dictionary". In other words, assume that $\mathbf{x}$ is a vector to be estimated, which is sparse with respect to a linear dictionary (or transformation) $\mathbf{D}^{n\times n}$. In other words, $\varphi = \mathbf{D}\mathbf{x}$ is sparse. Assuming that $\mathbf{D}$ is an invertible matrix, we can write:

$$\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{D}^{-1}\varphi = \mathbf{\Phi}\varphi, \tag{1.2.4}$$

where $\mathbf{\Phi} \triangleq \mathbf{A}\mathbf{D}^{-1}$. The latter is an under-determined system of equations with the unknown φ being sparse, and therefore has the same structure as (1.2.1). Examples of well-known linear dictionaries include: Fourier, cosine, wavelet, chirplet and Gabor transforms.

A more important question than the uniqueness of a sparse solution is how to recover it? The problem of finding the "the sparsest solution" to (1.2.1) can be written as an optimization program:

$$\min_{\mathbf{Ax}=\mathbf{y}} \|\mathbf{x}\|_0, \tag{1.2.5}$$

where the ℓ_0 norm of a vector $\|\mathbf{x}\|_0$ is defined as the number of nonzero entries in $\mathbf{x}$, which is a non-convex function. Therefore, (1.2.5) is a non-convex optimization problem for which no known efficient solvers exist. In fact, an exhaustive search approach can solve this problem as follows: consider every collection of k columns of $\mathbf{A}$, and try to find an inverse solution $\mathbf{x}$ assuming that the nonzero entries of $\mathbf{x}$ are restricted to the indices corresponding to the considered set of columns. In other words, try every possible "support set" for the vector $\mathbf{x}$. Since $m > k$, these systems will be over-determined and only the correct support will yield a consistent $\mathbf{x}_0$. This approach takes $\binom{n}{k}$ operations and, unless $k = O(1)$, is exponentially complex.

One possible way to overcome the computational intensity is try to approximate (1.2.5) with a convex optimization. The closest convex relaxation to the ℓ_0 norm is the ℓ_1 norm defined as the sum of absolute values of the entries of $\mathbf{x}$. The resulting optimization becomes:

$$\min_{\mathbf{Ax}=\mathbf{y}} \|\mathbf{x}\|_1. \tag{1.2.6}$$

In fact, (1.2.6) is a *linear program*, and many efficient methods exist that can solve it in time polynomial in n ($\mathcal{O}(n^3)$ to be precise).

The theory of compressed sensing has evolved around finding efficient solutions to

the sparse recovery problems (1.2.1) and (1.2.5), including the convex optimization of (1.2.6). In this context, (1.2.6) is commonly known as ℓ_1 minimization, ℓ_1 regression, or Basis Pursuit, and has been extensively studied. We will soon return to discuss this in more details.

1.2.2 Interpretations

The linear inverse problem described in (1.2.1) can be interpreted in various ways depending on the context and physical application. We consider the following different outlooks.

Sub-Nyquist Sampling

Compressed sensing can be regarded as a method of sampling signals at sub-Nyquist rates, or equivalently a technique for jointly sampling and compressing real time analog data. The fundamental Shannon-Nyquist sampling theorem states that a continuous-time band-limited signal can be sampled at discrete points at a frequency rate equal to twice the bandwidth of the signal, without compromising the information content [Mar06]. This statement is very helpful in designing analog-to-digital systems and determines the bottleneck in storing and processing continuous real-time signals. Compressed sensing theory provides means of achieving lower sampling rates than the Nyquist rate. Assume the vector $\varphi = (\varphi_1, \cdots, \varphi_n)^T$ represents samples of a a band-limited function $f(t)$ taken at the fundamental Nyquist rate:

$$\varphi_i = f(i/T),\ 1 \leq i \leq n, \tag{1.2.7}$$

where $T = 1/(2B)$ is the sampling period, and B is the bandwidth of $f(t)$. The band-limited assumption on $f(t)$ assures that the frequency content of $f(\cdot)$ has a lot of vacancy, or equivalently that the vector $\mathbf{x} = \mathbf{F}\varphi$ is approximately sparse, where $\mathbf{F}$ is the discrete Fourier matrix of size $n \times n$. Therefore instead of storing samples φ, one might attempt to store the linear combinations obtained by

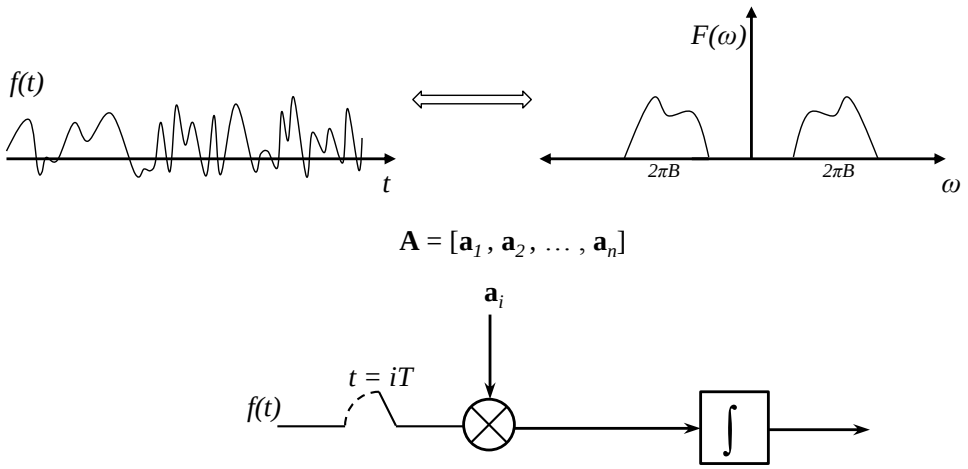


Figure 1.2: Schematics of a sub-Nyquist sampling system

$$\mathbf{y} = \mathbf{AF}\phi = \mathbf{Ax}, \tag{1.2.8}$$

where $\mathbf{AF}$ is a matrix appropriated for compressed sensing with $m < n$ rows. The resulting sampling rate is thus $\frac{m}{n}\frac{1}{T}$. When applied to real-time signals or streams of data, the matrix $\mathbf{A}$ acts as a correlation operator, and is therefore referred to as the *correlation matrix* or *sketches* (see Figure 1.2). For more information on this subject please refer to [ME11].

Estimation Under Ill-Conditioned Observations.

Another interpretation of the linear inverse problem (1.2.2) is linear regression under an under-determined set of observations. In the linear regression model, $\mathbf{x}$ (statisticians prefer β) refers to a set of parameters of a regression model to be estimated. The entries of $\mathbf{A}$ are called regressors. Different rows of $\mathbf{A}$ are independent realizations of independent variables, and $\mathbf{y}$ is the regressand vector.

The objective of regression models is usually to minimize the estimation risk. The (co)statistics of the parameters $\mathbf{x}$ and the error terms $\mathbf{v}$ should be known, and depending on the risk criteria chosen, different estimators can be selected. Least square and maximum likelihood solutions are archetypes of estimation criteria. When $\mathbf{A}$ does not have full column rank, the number of observations is not sufficient and

other conditions such as sparsity must be assumed for the parameters. Sparse linear regression models can be studied in the context of Bayesian compressed sensing [duk, JXC08].

Source Coding Scheme

Compressed sensing can also be regarded as an analog compression scheme for peeling off the redundancy of a sparse generating source. In this case, the matrix $\mathbf{A}$ is in fact an *encoding* matrix, and reconstruction is a *decoding* algorithm. Information theoretic bounds can be found that relate the compression rate to the information rate of sparse sources, as well as trade-offs between the compression rate and quantization error [FRG07] (rate distortion theory for compressed sensing).

Data Privacy

Suppose that $\mathbf{x}$ represents a private database, not directly accessible to outside users. The database can however be observed through random queries in the form of linear projections corrupted by noise, as $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{v}$. Understanding the limits of sparse recovery in such statistical setting helps protect private data, say by adding the right amount of noise. This model of privacy and its connection to compressed sensing has been studied in [DMT07].

1.2.3 History

Deconvolution of sparse signals using ℓ_1 regularized estimators dates back to the 1960s [Log65]. Starting in the 1970s, the use of such techniques became popular in determining marine surface structures from the reflection of acoustic pulses [TBM79]. Seismological traces are the convolution of a source wavelet (acoustic wave) with the impulse response of the marine surface which is often a sparse train of spikes [CB83]. Therefore sparse-promoting linear solvers such as ℓ_1 norm regularized least square techniques were used to extract surface patterns. In the 1990s and with the advent of powerful computerized solvers for linear programs, ℓ_1 regularized regressions were revisited and studied more rigorously by statisticians. D. Donoho [CD94] studied ℓ_1

minimization in the context of signal representation under alternative dictionaries, and suggested the Basis Pursuit (BP) method for atomic decomposition of signals in overcomplete dictionaries [CDS01]. Donoho's proposed scheme was in contrast to the rising Matching Pursuit (MP) algorithm proposed slightly earlier by Mallat and Zhang [MZ93]. The MP method was introduced as a way of denoising signals that are sparsely representable over redundant dictionaries. MP is a greedy algorithm that iteratively selects a basis function that matches best (i.e. has the highest correlation) with the signal projection. Tibshirani proposed the least absolute shrinkage and selection operator (LASSO) in 1996, an ℓ_1-norm-bounded constraint least-square minimization, and argued that LASSO is advantageous to most regression techniques in sparse model fitting [Tib96]. LASSO was studied in the context of Bayesian estimation, specifically for model fitting in medical applications [Tib97]. However, to this point, most of the results remained mainly empirical.

In the early 2000s an explosion of analytical results on sparse recovery techniques occurred. Donoho et al. first proved that for measurement matrices (or overcomplete dictionaries as preferred by some) with a sufficiently large mutual coherence, Basis Pursuit has a stable recovery in the presence of noise, provided that the unknown signal is sufficiently sparse [DET06]. Following that, in a series of breakthrough papers, Candès, Romberg, and Tao proved that ℓ_1 minimization allows exact reconstruction of sparse vectors from random Fourier dictionaries and random Gaussian projections with an almost optimal number of measurements [CRT06b, CT06a, CT05]. In [CRT06b, CT06a], it was shown using robust uncertainty principles that exact reconstruction of a k-sparse vector is possible with high probability using a random set of frequency projections of size $m = \mathcal{O}(k \log n)$. In [CT05] the notions of restricted orthogonality and restricted isometry property (RIP) were introduced, and it was shown that if the measurement matrix $\mathbf{A}$ satisfies these conditions, then exact sparse decomposition is possible for *all* sparse signals. This result is in particular very strong for its universality. The coarse definition of RIP for a matrix $\mathbf{A}$ is

that pairwise distances between sparse vectors is approximately preserved (thus the word isometry) when projected by $\mathbf{A}$. Candès and Tao further proved that random matrices with Gaussian entries satisfy RIP, and thus recovery of sparse signals and stable recovery of approximately sparse signals with an order-wise optimal number of measurements is possible. Soon after that, Donoho and Tanner proved similar universality results for BP under random projections and provided stronger guarantees than the previous results. Donoho et al. defined the notion of *neighborliness* for high-dimensional polytopes. k-neighborliness for a polytope implies that every subset of k vertices form a $k-1$ dimensional face, and are thus neighbors in that sense. The object with the highest order of neighborliness in $\mathbb{R}^n$ is the n-dimensional simplex. Donoho et al. proved that successful sparse reconstruction of k-sparse vectors using BP is equivalent to the k-neighborliness of the n-dimensional simplex projected by $\mathbf{A}$ [Don06b, DT05a, DT05b]. Under the assumption of asymptotically large vector size n, and proportional system dimensions, i.e. $m = \Theta(n), k = \Theta(n)$, it is possible to analyze the neighborliness property for random projections such as Gaussian measurements. The analysis requires advanced high-dimensional convex geometry techniques. Under such circumstances, Donoho et al. derived tight bounds known as *recovery thresholds* that accurately predict the success of ℓ_1 minimization for large-dimensional inverse systems. The characterization of the recovery thresholds presented by Donoho et al. is very involved, and does not allow for further exploring these bounds. Later works done by Xu et al. and Stojnic et al. came with easier characterizations and extensions of the recovery thresholds of ℓ_1 minimization [SXH08, XT10b, Sto10, Sto]. Specifically, [SXH08] provided an equivalent condition for the neighborliness property, which is expressed in terms of the null space of the measurement matrix. These null space properties are easier to analyze and result in a so-called "Grassmann manifold" framework for the study of the properties of ℓ_1 minimization. Using this framework, [XT10b] provided a tight analysis of the robustness properties of Basis Pursuit, which addresses its asymptotic performance

under the presence of noise and for approximately sparse signals. The more recent work of Stojnic et al. is based on a technique called "escape through the mesh" and results in much simpler and more explicit calculations for the thresholds of Basis Pursuit under than the Grassmann manifold approach.

Interest in the theory of sparse recovery sharply increased after the fundamental results of Donoho and Candès. Alongside the convex optimization methods, greedy algorithms were also revisited in the hope of more rigorous analysis. The Matching Pursuit technique was first refined to a more stable Orthogonal Mathing Pursuit (OMP) algorithm [PRK93, DMA97] without presenting significance analytical guarantees. OMP was studied later by various researchers including Chen et al. and Tropp et al. in the 2000s [CH06, Tro04, TGS06, TG07]. Specifically, it was shown in [TG07] that under the assumption that $\mathbf{A}$ is random Gaussian, $m = \mathcal{O}(k \log n)$ measurements are sufficient to guarantee successful recovery of sparse k-sparse signals with high probability, a bound which is order-wise equivalent to that of BP. Donoho et al. introduced a more advanced version of OMP called Stagewise Orthogonal Matching Pursuit (StOMP), and showed empirically that their proposed method has superior performance to Basis Pursuit in the asymptotical proportional regime for extremely under-determined systems ($\frac{m}{n} << 1$) [DTDS06]. Needell and Tropp proposed CoSaMP, a method that has a faster running time than OMP and guaranteed stability to noise [NT08].

Many more varieties of the described algorithms have been proposed over the past few years, and many alternative techniques have been developed. Today, sparse reconstruction techniques are not limited to convex relaxation or greedy approaches, and contain a vast number of alternative iterative, combinatorial, and algebraic techniques. We will mention many such techniques in the later chapters of this thesis.

1.2.4 A Few Fundamental Questions

Given the sparse linear inverse problem, a number of key questions can be asked.

What is a good reconstruction algorithm?

A good algorithm should be efficient, resilient to noise, and easy to implement. Furthermore, the exact definitions of these criteria depends on the particular application to which sparse estimation is mapped.

What are good measurement matrices A?

In addition to identifying matrices that make the reconstruction a feasible task, it is important to design structures that are amenable to certain recovery algorithms and vice versa.

What happens in the presence of noise?

In the presence of noise $\mathbf{v}$, we are interested in the "robustness" of recovery algorithms. Assume that $\hat{\mathbf{x}}$ is an approximation to $\mathbf{x}$ provided by a reconstruction algorithm. We say that the recovery algorithm is robust if for some norm functions f, g, and a constant c we can guarantee the following:

$$f(\mathbf{x} - \hat{\mathbf{x}}) \leq c \cdot g(\mathbf{v}). \tag{1.2.9}$$

.

What is the minimum number of measurements required for successful sparse estimation?

This question is very important, as its answer determines the performance limits of sparse recovery algorithms.

What is the tradeoff between performance and noise/quantization error level in sparse recovery?

This often arises in practical situations. Can we have any hope of reconstructing denser signals if the signal-to-noise ratio (SNR) improves? Conversely, how much information do we lose under noisier measurements or higher levels of quantization error?

How can an algorithm/matrix be tailored to a particular application?

As we will see, some measurement matrices and recovery algorithms are suitable for particular applications due to the inherent nature of the application, or for simplicity of implementation. This question has practical advantages too.

Is there an efficient method for assessing the goodness of an arbitrary measurement matrix A?

It is very important to have deterministic guarantees for practical applications. If $\mathbf{A}$ is a large measurement matrix, is it possible to determine (in a reasonable time) how good of an encoding matrix it is? In other words, is there a performance metric for a matrix with regards to sparse compression/decompression that can be verified in efficient time?

These questions (or criteria based on these questions) have served as the road map for research on sparse linear inverse problems. Below we briefly discuss the existing methodologies and guarantees in response to the above questions.

1.2.5 Existing Methodology

Random matrix ensembles are commonly used as the measurement matrix in the study of sparse recovery problems. The original results of compressed sensing were based on matrices with i.i.d Gaussian matrices [CT06a, CT05, Don06b] but were soon extended to other distributions such as Bernoulli or random ± 1 entries [BDDW08, DT10], and generally to all sub-Gaussian distributions [MPJ09, CR11]. Furthermore, random partial sub-dictionaries of most well-known dictionaries form relatively good measurement matrices. For instance, random selection of m rows from the $n \times n$ Fourier matrix is a good encoding matrix with high probability [CRT06b]. It is generally easier to study random matrices in the context of compressed sensing, due to the existence of advanced asymptotic spectral analysis tools for random matrices,

and high dimensional convex geometry techniques. However, a number of combinatorial and deterministic structures have also been introduced, due to practical benefits such as lower cost of encoding, ease of implementation, and better reconstruction algorithms. The most common classes of such matrices are sparse binary matrices, such as expander graphs [Ind08, XH07b], measurement matrices based on LDPC error-correcting codes, Van der Monde matrices [DeV06], circulant and Toeplitz structures [Rau], measurement matrices based on algebraic error-correcting codes such as Reed-Solomon, Reed-Muller, and p-ary codes [HCS08, PH08, AM11, AMM12], matrices based on hash functions and sketches used for streaming applications [CM04, CM05, CM06], and others (see e.g. [GSTV06, DMP11]). The diagram in Figure 1.3 summarizes these different categories of existing measurement matrices.

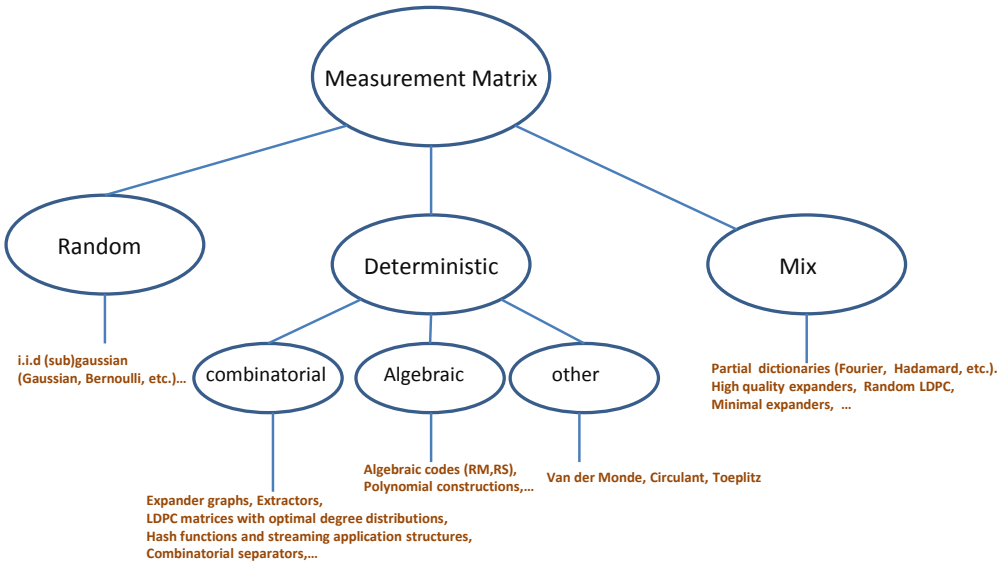


Figure 1.3: Summary of existing categories of measurement matrices for compressed sensing with examples

Existing reconstruction algorithms for compressed sensing can be divided into three main categories: geometric approaches, greedy methods, and combinatorial algorithms. Geometric methods are mostly based on convex optimization techniques such as linear or quadratic programming. Basis Pursuit (BP) and Basis Pursuit denoising (BPDN) are the most highlighted examples of reconstruction methods based

on convex optimization. BP is the ℓ_1 minimization program described in (1.2.6). BPDN [CDS01] is a quadratic programming which is essentially a regularized Lagrangian of the constrained ℓ_1 minimization:

$$\textbf{BPDN} \quad \min \tfrac{1}{2}\|\mathbf{Ax}-\mathbf{y}\|_2^2 + \lambda\|\mathbf{x}\|_1. \tag{1.2.10}$$

Other similar estimators have been used based on alternative forms of regularization which are common in statistical estimation, such as the LASSO [Tib96] and Dantzig selector [CT07]:

$$\begin{aligned} \textbf{LASSO} \qquad & \min \tfrac{1}{2}\|\mathbf{Ax}-\mathbf{y}\|_2^2 \\ & \text{subject to } \|\mathbf{x}\|_1 \le t. \end{aligned} \tag{1.2.11}$$

$$\begin{aligned} \textbf{DS} \qquad & \min \|\mathbf{x}\|_1 \\ & \text{subject to} \|\mathbf{A}^t(\mathbf{Ax}-\mathbf{y})\|_\infty. \end{aligned} \tag{1.2.12}$$

Greedy algorithms are mostly based on iterative approximations to a sparse solution, such as the Matching Pursuit method and its variations (OMP [TG07, CH06, Tro04, TGS06], StOPM [DTDS06], CoSaMP [NT08, NTV08], etc.), iterative least square techniques [DDFG10, CY08], and iterative thresholding methods [FR08, BD08]. These methods are generally inferior to their corresponding convex program as they require more careful regularization, but are easier to implement.

Combinatorial methods of sparse recovery are based on measurement matrices with particular structures, often deterministic. A plethora of alternative combinatorial methods have been proposed in the past few years based on specially designed forms of matrices, often resulting in simplicity of implementation, less complexity, and better performance bounds compared to the greedy or convex optimization-based algorithms. Selected examples are combinatorial algorithms based on expander

graphs, such as the bit flipping decoding algorithm [XH07b], expander Matching Pursuit (EMP) [IR08], sparse Matching Pursuit (SMP) [BIR08], and sequential Sparse Matching Pursuit (SSMP) [BI09], list-decoding algorithms and other reconstruction methods for constructions based on algebraic error-correcting codes [HCS08, HSC09, PH08], algorithms based on data streaming applications [CM05], message-passing decoding [CSW10, LMP08], Chaining Pursuit [GSTV06], and many more. The diagram in Figure 1.4 summarizes these different categories of existing reconstruction algorithm.

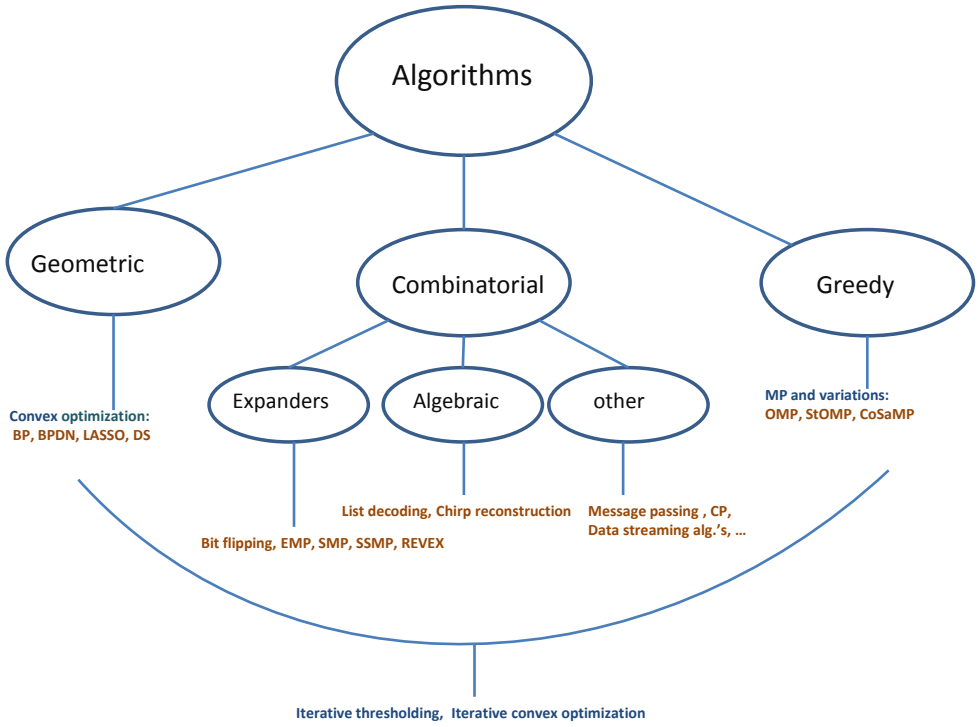


Figure 1.4: Summary of existing categories of algorithms for compressed sensing with examples

1.2.6 Applications

Research on compressed sensing in the past few years has identified many applications for the sparse linear inverse problem. We mention a few examples, while many more applications can be found in [ric] and other more recent references.

Image and Video Processing

Many forms of images are approximately sparse over some well-known dictionary. For example, natural images over the discrete cosine transform (DCT) domain are known to be concentrated mostly on lower frequencies. Astronomical images are often sparse in their regular spatial domain. Consecutive frames of a high-speed video have sparse residual. In light of this, image acquisition systems based on compressed sensing have two possible mechanisms. An original high-resolution image can be compressed using a linear encoding matrix. Alternatively, an input image can be directly measured and stored in the form of linear projections. The former is a compression scheme which is helpful in maintaining large databases. The latter is in particular useful in biomedical imaging, such as magnetic resonance imaging (MRI) [LDP07, LDSP08, KNN09], where obtaining higher resolution images requires patients to be exposed to stimulating signals (e.g. electromagnetic waves in an MRI scanner) for long periods of time. Linear combinations of image components (over some dictionary, such as wavelet) can instead be obtained by simultaneously combining different components of the stimulating signal. A reconstruction algorithm can then be used to recover the desired information. A similar application of this approach is in image or video cameras that record linear combinations of pixel intensities or consecutive frames, instead of a super-high-resolution pixel array [VRR11]. These ideas have led to the development of prototypes for imaging devices that function based on compressive sampling technologies, such as the single-pixel camera [DDT$^+$08].

Radar

Many forms of modern radars function based on real-time scanning of the frequency-time spectrum, and identifying target attributes based on the spectral content. The resolution and speed of detection for radars thus depend on the rate and the resolution at which they sense the spectrum. The bottleneck in high-speed spectrum sensing is the analog-to-digital (A/D) conversion. Radars work based on transmitting and

receiving continuous time pulses, but need to digitize data for processing. Sparse time-frequency spectrums can be sensed at higher rates using compressive sampling. Instead of sampling continuous time signals and digitizing the samples, we can obtain linear projections of the input stream using high-speed correlation circuits, and then digitize the projections [HS09]. Also, please refer to [CV08] for an alternative use of sparse recovery in MIMO radar detection.

Beyond radar applications, achieving higher A/D rates is extremely motivated, and numerous attempts at implementing analog-to-information compressed-sensing-based systems have been reported over the past few years [LKD$^+$07].

DNA Micro-Array

Micro-arrays are large sets of parallel microscopic DNA probes that can detect the expression levels and consequently identify the absence or presence of different genes inside a DNA genome solution. DNAs and RNAs can bind to the probes by a process called hybridization. Conventional DNA micro-arrays work based on the principle that every probe detects at most one target DNA sequence, and thus require a large number of genomic probes. In addition, cross-hybridization is a common phenomenon which degrades the performance of the sensor array. In contrast, a new micro-array technology which is based on compressed sensing has been developed where multiple DNA sequences can bind to each probe. The probes read-outs are thus roughly equal to linear combinations of gene expression levels. Sparse reconstruction algorithms are invoked to estimate the existing DNA genes and their expression levels [PVMH08, DMSB08, MBSR07, ES05, VPMH07].

Networks

Sparsity is an inherent characteristic of many forms of network data, including computer network traffic, connectivity patterns in wireless ad-hoc, and attributes of social networks. Recent research trends have tried to apply sparse reconstruction techniques to problems such as monitoring computer networks [CPR07], neighbor discovery in

ad-hoc networks [ZLG], and clique identification in social networks [JYG09].

Communications

Compressive sampling has led to novel mechanisms for sparse channel estimation, spectrum sensing, interference alignment, user detection, and design of communication protocols in wireless MIMO and ad-hoc networks over the past decade (see, e.g. [BHRN08, BHSN10, TH08]). For a comprehensive study, we refer the interested reader to the references provided in [ric].

More

Applications of sparse reconstruction techniques are not limited to the above examples. There are many more applications where a sparse linear inverse model accurately fits. This includes almost all areas of signal processing from financial engineering to astronomy. However, it is important to note that the requirements, methodology, and existing guarantees can be very different from one application to another. We will discuss this issue in more detail in future chapters.

1.3 Beyond Compressive Sampling

In many cases, there are other forms of common non-linear transforms that result in sparse (or compressible) representations of certain classes of signals. For example, if the unknown signal is a matrix $X^{n\times n}$ of rank $k << n$, then the singular value decomposition (SVD) of X results in a sparse set of singular values consisting of only k nonzero values. This approach can be generalized: Suppose that we have a signal that belongs to a family of low-dimensional manifolds, but is accessible through a small number of linear measurements. How can such a signal be estimated? This general form of the sparse linear inverse problem was discussed in [CRPW, BW09]. In particular, the problem of low-rank matrix estimation from linear projections is

highly motivated under the terms "rank minimization" and "matrix completion". Reconstruction of rank-deficient matrices from ill-posed lower projections or completion of such matrices from partially known entries is motivated by problems in covariance estimation, dynamic systems, quantum computation, and collaborative filtering. A great example is the well-known Netflix problem, the objective of which is to predict ratings of different movies by a large number of users. The results of such prediction would help Netflix provide targeted recommendations to its users. The table of users/movie ratings is a two-dimensional matrix that needs to be estimated in this case. A valid assumption often made is that by nature, such matrices are approximately low rank as many users tend to have similar or highly correlated interests. In addition, only a fraction of the target matrix is often available in the form of random entries, and the rest is to be estimated. The problem is therefore that of matrix rank minimization. Similar models show up in other forms of recommendation systems and search engines, such as the Google PageRank problem [AT05].

Matrix rank minimization and low-rank matrix completion have been the center of attention by many in the past few years. Fazel et al. proposed using a convex heuristic known as nuclear norm minimization (NNM) to estimate low-rank matrices. Nuclear norm is a convex relaxation to the rank function of a matrix and thus a low-rank promoting regularizer. This approach is very similar to the use of ℓ_1 minimization in compressed sensing. Rigorous analysis of the NNM method has been done in [RFP10, RXH08, OH, CR09, CT10]. However, despite significant efforts, little progress has been made when compared to compressed sensing. For example scant work exists on faster reconstruction algorithms or alternative forms of projections. Therefore, matrix rank minimization is currently a very open and motivated field of research.

1.4 Contributions of This Thesis

Despite a tremendous amount of recent effort, in most cases definite answers to the questions put forward in Section 1.2.4 remain open, and the existing answers are

restricted to specific contexts and applications. In this dissertation, we have tried to understand the boundaries of existing knowledge and methodology on sparse recovery techniques, and extend them in various ways. Below is a summary of the contributions of the current thesis and the descriptions of how they fit into the state-of-the-art literature.

1.4.1 Compressed Sensing

The seminal results of Candès et al. and Donoho et al. [CT05, Don06a] established that a class of convex optimization methods can solve the sparse linear inverse problem much more efficiently than exhaustive search. As discussed, these methods are known as Basis Pursuit or ℓ_1-regression techniques, and require an order-wise optimal number of measurements $m = \mathcal{O}(k \log(n/k))$ to succeed. Basis Pursuit algorithms are popular because they have universal guarantees and computable phase transition thresholds, and have a polynomial complexity $\mathcal{O}(n^3)$ that allow them to be implemented in regular modern computers for moderate problem dimensions ($n \leq 10^5$). However, they become inefficient for larger dimensions. Furthermore, because they are solved using geometrical approaches such as interior point methods and demand excessive memory requirements, it is difficult to implement them in parallelized platforms or as embedded software for practical applications. Another challenge is that thorough analysis of Basis Pursuit is difficult, and mostly exists for random and dense measurement matrices $\mathbf{A}$, whereas in many cases, the measurement matrix $\mathbf{A}$ is desired to be sparse or to meet certain other restrictions. For these reasons, several other algorithms have been developed, based on combinatorial or greedy approaches to the sparse recovery problem, some which were mentioned in Section 1.2.5. However, most of these methods are inferior to Basis Pursuit in different ways, such as performance or tolerance to noise. A good algorithm needs to compete with existing recovery methods in multiple ways, and thus designing new algorithms is an important and challenging research problem. Furthermore, many of the proposed recovery

methods are tailored to specific measurement matrices. Designing matrix structures that are amenable to existing or new methods of sparse reconstruction is therefore as important as algorithm development. In light of these issues, our contributions fall in the the following categories:

Deterministic Measurement Matrices

Although the performance of random and dense measurement matrices are easier to analyze, they pose several practical limitations, including extensive memory requirements, high recovery complexity, and lack of deterministic guarantees. One of our objectives is developing deterministic and/or sparse matrix constructions for sparse recovery, either for the existing reconstruction algorithms (such as Basis Pursuit), or in conjunction with novel estimation methods. In either case, providing theoretical guarantees for the proposed construction/reconstruction schemes is imperative. Three main categories of novel deterministic structures discussed in this thesis are the following:

• **Minimal expander structures.** Expander graphs are combinatorial objects with unique features that make them useful in various applications. Despite having small vertex degree, i.e. being sparse, the overall connectivity of expander graphs is high, and thus leads to fast mixing times, making them good candidates for Monte Carlo methods and as error-correcting codes. Expander graphs are characterized by an expansion coefficient $0 < \epsilon < 1$[1]. Larger ϵ corresponds to better connectivity and smaller mixing times.

A recent research trend has proposed using bipartite expander graphs as compressed sensing measurement matrices, leading to a number of combinatorial algorithms and methods to analyze Basis Pursuit for sparse constructions [XH07b, BGI+08]. Results in the literature prior to the contributions of this thesis only considered high-quality expander graphs with expansion coefficients $\epsilon \geq 3/4$. It is generally

[1]We shall formally define "expander graph" later in this thesis.

harder to construct expanders with higher expansion coefficients, and consequently the required oversampling factors m/k for successful recovery of sparse signals become extremely large. For example, when a compression ratio $m/n = 0.5$ is considered, the best existing results for high-quality expanders only guarantee[2] that sparsity fractions $k/n \approx 10^{-6}$ can be reconstructed efficiently. This means there should only be one nonzero coefficient in a million! In contrast, we introduce a class of minimal expanders in Chapter 2 with much smaller ϵ values, that can be deterministically constructed. We have analyzed the proposed constructions in the realm of the Basis Pursuit algorithm [KXDH10]. The resulting guaranteed sparsity fractions can be as high as $k/n \approx 10^{-2}$, filling the gap between the performance of dense and sparse measurement matrices for compressive sensing. Chapter 2 of this thesis is devoted to this subject.

• **Binary structures based on graphs with logarithmic girth.** Using results from channel coding, we have developed novel constructions for the measurement matrix $\mathbf{A}$. "Good" channel coding parity check matrices form "good" measurement matrices for compressed sensing. Using this idea, we have studied several constructions of LDPC codes in the context of compressed sensing that result in even tighter provable thresholds than minimal expanders. We show that the structure of the Tanner graph of a code and some of its fundamental properties, such as the minimum cycle length (girth) can be used to assess the goodness of a given binary matrix with respect to Basis Pursuit algorithms. The importance of this criterion is that it can be checked in polynomial time. Specifically, we show that LDPC codes with $\Omega(\log(n))$ girth offer very tight thresholds for the number of measurements for a robust ℓ_1/ℓ_1 approximation noise [KTDH11]. Our bounds are the tightest existing guarantees using sparse measurement matrices. These ideas are discussed in detail in Chapter 3 of this thesis.

[2]Not that these are theoretical results only. In practice, the observed performance is much better.

• **Summary-based structures for very fast and sub-linear compressed sensing.** There are many instance of sparse recovery problems where the signal dimension is extremely large (say $n = 10^6$–10^{12}), and the number of significant nonzero entries in the signal is in the range of 100–1000. Important examples where this setup arises are neighbor discovery in ad-hoc sensor networks and efficient multi-user RF-ID. In these cases, conventional recovery methods with recovery times polynomial in n fail in practice, as n is very large. Therefore, algorithms with "sub-linear" complexity have to be developed, where the complexity is logarithmic in n, i.e. $\mathcal{O}(\text{poly}(\log(n)))$, and polynomial in k. We have developed a new class of measurement structures with the motivation of designing algorithms that have a sub-linear complexity in the system dimension. These constructions will be presented in Chapter 4. The matrices that we propose are highly structured and facilitate summarized searches over the span of the unknown vector. Using ideas from combinatorial separators and hash functions, we developed matrices based on binary labeling and sub-labeling of the state space. In addition to the obvious time/memory advantages that result from the fast reconstruction algorithms we propose for these structures, the underlying constructions are greatly motivated by a variety of statistical inference problems such as popular political ranking, market basket analysis, revenue maximization, graphical models, and so on. In many cases, the regressors of the inference problem are in the form of a matrix $\mathbf{A}$ similar to summary based structures (see e.g. [KKH11, JS08, JYG09]). Chapter 4 is dedicated to these concepts.

Recovery Algorithms

In addition to designing recovery algorithms for the deterministic constructions discussed previously, in this thesis we look at the possibility of improving the Basis Pursuit algorithm. In Basis Pursuit, the relationship between the highest recoverable sparsity k/n and the compression ratio m/n demonstrates an asymptotic phase

transition threshold, which is explicitly computable. However, an important open problem is whether there exist other polynomial time algorithms that have superior thresholds to those of Basis Pursuit. This problem is addressed in this dissertation. A solution is provided that holds for most cases, with the exception of a restricted class of random signals where the distribution of the nonzero entries and all its finite derivatives vanish at the origin. For example, binary $(0, 1)$ and ternary $(0, \pm 1)$ sparse vectors are of this type.

Below is a summary of sparse recovery algorithms that will be introduced and analyzed in this thesis:

• **Two-step reweighted Basis Pursuit.** In Chapter 6, we introduce a two-step linear programming algorithm and prove that for many classes of random sparse signals, the proposed method has better phase transition thresholds than Basis Pursuit. To our knowledge this is the only result of this kind. The proposed algorithm is based on coarsely approximating the support set of a sparse signal with the help of Basis Pursuit, and separating the entries based on whether we "believe" they are zero or nonzero. Then, a second linear program is performed, in which the entries that are believed to be zero are penalized with a larger weight. The practical improvement of the proposed scheme is significant. For instance, for a compression ratio $m/n = 0.5$, the recoverable sparsity ratio k/n can be improved by 20%.

• **Reverse expansion linear inversion (REVEX) algorithm for minimal expanders.** We propose this algorithm in Chapter 2. In particular, this is an algorithm designed for the minimal expander constructions which will also be explained in Chapter 2. The method has $\mathcal{O}(k^2 n)$ complexity, which makes it significantly faster than Basis Pursuit ($\mathcal{O}(n^3)$) for the corresponding expander codes, and has a theoretically equivalent performance.

• **Support index inference (SIR) algorithm for the summary-based structures.** This is an algorithm specifically designed for the summary-based structures

which will be explained in Chapter 4. The routine is extraordinarily fast and can handle recovery of sparse sample points in extremely large dimensions. Theoretically, the decoding complexity is $\mathcal{O}(km\log(m))$ and the required number of measurements is almost optimal $m = \mathcal{O}(k\log(n)\log\log(n))$. The advantages are not limited to a theoretical level; the method can be of great practical value. For instance, preliminary simulations reveal that a $k = 100$ sparse vector of length $n = 10^{12}$ can be reconstructed using $m \approx 5000$ measurements within less than a minute on normal desktop computers. These figures are linearly scalable to higher dimensions, and stand far above the state-of-the-art efficient sparse recovery performances. In addition, the noise tolerance of the SIR method is competitive with the best existing methods. Such a combination of strong theoretical guarantees and practical evidence make this framework an ideal candidate for many real-world high-dimensional applications, potentially extending to hardware-level implementation.

1.4.2 Rank Minimization

Although the basic model and many heuristics in matrix rank minimization are similar to compressed sensing, the technical analysis requires far more detailed knowledge of random matrices and linear algebra. As a result, less progress has been made than in the vector recovery case. In particular, the majority of the results on matrix rank minimization are limited to random Gaussian measurements or partially observed elements [Faz02, CT10]. Similarly, reconstruction methods are mainly restricted to semi-definite programming techniques such as trace or nuclear norm minimization and singular-value thresholding methods. This is in contrast to compressed sensing, where many other instances of random and deterministic structures are proven to be practical, and various combinatorial and greedy algorithms have been developed. Designing deterministic structures for low-rank estimation and fast algorithms is therefore highly motivated. In Chapter 8, we introduce matrix operator objects that mimic expander graphs in the vector recovery case. Specifically, the proposed

operators have two key properties: 1) They have low density, which means that the projection of rank 1 matrices have at most rank $d = \mathcal{O}(1)$, and 2) despite having low density, the operators expand the rank of low-rank inputs, and as such are called "rank expanders". We propose a combinatorial algorithm that allows for the reconstruction of sufficiently low rank matrices using the proposed rank-expanders. The algorithm resembles the REVEX method developed earlier for minimal expanders in compressed sensing, and in a similar way is significantly faster than geometric methods based on semi-definite programming. Additionally, rank-expanders appear in a number of applications, such as system identification and quantum computing, unlike generic random Gaussian operators, further motivating their use.

1.4.3 LDPC Codes and Improved Channel Coding

Intuitions and ideas developed in compressed sensing can be applied to other signal-retrieval applications, one instance of which was described earlier for low-rank estimation problems. Channel coding is another possible area. Due to the maturity of channel coding techniques, it is often easier to adopt code designs and estimation techniques and apply them over the real numbers to the compressed sensing problem, as has been done in a collection of recent publications. We look at the possibility of a reverse approach, by adopting some sparse recovery techniques and using them to improve the performance of channel codes. We introduce a reweighted LP decoding algorithm for retrieving noisy codewords of a linear codebook from the output of bit-flipping communications channels. This method was inspired by the reweighted Basis Pursuit algorithm described in for sparse recovery (described in Chapter 6), but is analyzed in quite a different framework. When compared with the existing polynomial-time decoders for random LDPC codes, the performance curves of the proposed algorithm (namely bit-error-rate curves) are superior to the state-of-the-art techniques. As linear programming decoding is very close in performance to belief-propagation algorithms such as the min-sum method, this approach can lead

to systematic improvements in a variety of important inference problems.

1.5 Organization

The content of this paper is presented in three main parts. In the first part which contains Chapters 2–4, we discuss new combinatorial structures for the design of measurement matrix $\mathbf{A}$ in compressed sensing. We study these constructions in detail and evaluate the performance of various reconstruction algorithms based on these designs. An overview of the use of sparse matrices for compressed sensing is given in Chapter 2, and minimal expanders are introduced. In Chapter 3, we study connections between channel coding and compressed sensing, and prove certain performance results for bipartite graphs with logarithmic girth. In Chapter 4, sub-linear time algorithms are motivated and discussed, and summary-based structures are introduced which lend themselves to a very fast and practical reconstruction algorithm.

Part II has a relatively different theme, and is mostly focused on the Basis Pursuit algorithm and random Gaussian measurements for sparse recovery. We study ways to improve the theoretical and practical thresholds of the Basis Pursuit algorithms. These thresholds identify the asymptotic performance of an ℓ_1 minimization algorithm for the case of "proportional" system dimensions, i.e. k,m, and n are proportional to each other. In Chapter 5, we show that if prior information is available about the unknown signal in the form of non-uniform sparsity, then weighted ℓ_1 algorithms have higher thresholds than regular Basis Pursuit. In Chapter 6, we introduce and study a two-step reweighted ℓ_1 minimization algorithm with many of the technical tools borrowed from Chapter 5.

Finally, in Part III we generalize some of the results of the previous chapters to low-rank matrix estimation (Chapter 7) and channel coding (Chapter 8).

Despite the heterogeneity in the technical contents of the chapters, we have tried to adhere to a relatively similar format for all chapters: background, motivation, and prior work are presented at the beginning, and are followed by the main contributions

and discussions. Simulation results (if any) are presented at the end. For simplicity of reading, we have pushed long proofs to an appendix section at the end of each chapter. A table of important notations is also included before the introduction in each chapter.

1.6 Short Note on Notations

We have tried to use a consistent and simple set of notations throughout this dissertation. At the beginning of each chapter, notations used frequently are defined and ambiguities are resolved. As a general rule, we denote most scalars with small letters (e.g. $a, b, c, \alpha, \beta, \dots$), all vectors with small bold letters (e.g. $\mathbf{x}, \mathbf{y}, \mathbf{z}, \mathbf{v}, \dots$), all matrices with capital bold letters (e.g. $\mathbf{A}, \mathbf{H}, \mathbf{\Phi}, \dots,$), and sets with capital and non-bold letters (e.g. $S, T, X, Y, \dots$). Geometrical and high-dimensional objects and operators, such as subspaces, polytopes, and operators acting on matrices, are denoted by large and script letters (e.g. $\mathcal{S}, \mathcal{P}, \mathcal{A}(\cdot), \dots$). There are a few exceptions where some of these rules are violated, such as the use of capitalized letters for variables in some cases, but these should be very clear from the context.

Part I

Combinatorial Structures for Sparse Recovery

Chapter 2

Sparse Minimal Expanders

$\mathbf{A}$	measurement matrix
n	signal size
m	number of measurements
k	sparsity of the signal
$\mathcal{G}$	bipartite graph
$\Gamma(S)$	set of neighbors of S

2.1 Introduction

In this chapter, we focus on the design of sparse (i.e. low density) measurement matrices for compressed sensing. The-low density assumption for the matrices used is crucial for numerous reasons. In several applications, the cost of each measurement increases with the number of coordinates of the unknown vector $\mathbf{x}$ involved. For instance, in the design of DNA micro-arrays using the CS technology in which the pattern of micro-arrays translates directly to a suitable measurement matrix, every nonzero entry of the matrix represents a probe [PVMH08, DSMB09]. As a result, the overall cost of the micro-array panel is directly ruled by the density of the corresponding matrix. There are several other applications where only a sparse matrix assumption is close to a reasonable approximation of the underlying sketching process (see for example the motivation given in [GI10]). Sparse measurement

matrices have also made possible the design of faster decoding algorithms (e.g., [IR08, BIR08, XH07b, Tro04, XH07a, JXHC09]), apart from the general linear programming types of decoder [CRT06b, GLR08] originally proposed for dense matrices.

Unlike dense and random constructions (such as i.i.d. Gaussian matrices), where reasonably sharp bounds on the thresholds which guarantee linear programming to recover sparse signals have been obtained [DT05a], such sharp bounds do not exist for sparse measurements. Finding such sharp bounds for the special case where the k-sparse vector is *nonnegative* is the main focus of the current chapter. Although the nonnegativity constraint is primarily considered for ease of analysis, it represents a large class of practical interests. Signals arising in many problems are naturally nonnegative. Examples of positive real-world signals are natural or biomedical images, DNA micro-array data, network monitoring data, information collected based on hidden Markov models, and many more examples in which the actual data is of nonnegative nature. Compressed sensing for nonnegative signals has also been studied separately in various previous work [DT05b, BEZ08], but with different approaches.

In the remainder of this chapter, we carefully examine the connection between linear programming recovery and the fundamental properties of the measurement matrix, in light of the fact that the considered matrices are sparse.

2.2 Related Work

For a given measurement matrix, the success of linear programming recovery is often certified by the restricted isometry property (RIP) of the matrix [CT05]. For random dense matrices, these conditions have been studied to a great extent in the past few years. For sparse matrices, however, there were only a handful of promising results at the time when our very first results on this subject were published. Specifically, Berinde et al. [BGI$^+$08] showed that the adjacency matrices of suitable unbalanced expander graphs satisfy an RIP property for ℓ_1 norm. Please see Chapter 10 for the

definition of RIP. However, it turns out that RIP conditions are only sufficient conditions for the success of linear programming decoding, and often fail to characterize all the good measurement matrices. A complete characterization of good measurement matrices is given in terms of their null spaces [SXH08, FN03, LN06, Zha05]. A necessary and sufficient condition for the success of ℓ_1 minimization is therefore called the "null space property"[1]. Donoho et al. [Don06b] were the first to prove the validity of this condition with high probability for random i.i.d. Gaussian matrices, and were able to compute fairly tight thresholds regarding when linear-programming-based compressed sensing works [DT05a]. The first analysis of the null space for sparse matrices has been done by Berinde et al. [BI08], where in particular they consider measurement matrices that are adjacency matrices of *expander graphs*. It was shown that every $(2k,\epsilon)$ expander graph[2] with $\epsilon \leq \frac{1}{6}$ satisfies the null space property, and therefore every k-sparse vector can be recovered from the corresponding linear measurements. The recovery thresholds given by this result, namely the relationship between the sparsity-to-dimension ratio $\frac{k}{n}$, and the aspect ratio of the measurement matrix $\frac{m}{n}$ for which reconstruction is successful, are governed by the extents at which such suitable expander graphs exist. Expander graphs have either random or explicit construction (see, for example [GLW08] for explicit constructions of expander graphs). In either case, the resulting thresholds of [BI08] are very small (e.g., on the order of 10^{-5} for $\frac{m}{n} = 0.5$), due to the high expansion requirement, i.e., $\epsilon \leq 1/6$. In contrast, our analysis led to the design of sparse measurement matrices that obtained recovery thresholds around two to three orders of magnitude higher than the bounds of Berinde et al. (e.g., around 0.01 for $\frac{m}{n} = 0.5$), which, however, holds only for non-negative signals. The null space characterization and its use in compressed sensing has also been studied in [BEZ08], from a quite different perspective. In that paper, the authors show that a so called "coherence" measure on the measurement matrix

[1] Also referred to as the k-neighborly property [Don06b]

[2] We shall formally define (k, ϵ) expander graphs shortly.

is related to the null space property. Unfortunately, when applied to sparse matrices, this result does not yield very sharp bounds for the recovery threshold either.

Finally, there exist other related works in the literature that address the problem of sparse signal recovery in the case of sparse matrices, but for different recovery methods. The work of Wang et al. [WWR10] attempts to provide general information theoretic bounds for the feasibility of successful recovery, even if it requires brute force endeavor. The reference [ZP] provides a theoretical analysis (based on the density evolution technique) for a message-passing algorithm for recovering sparse signals measured by sparse measurement matrices. Another example is [LMP08], which considers the same problem, but for nonnegative signals. A shortcoming of the density evolution technique is that it can only determine asymptotic (infinite blocklength) results, and relies on an asymptotic limit exchange. Furthermore, it should be clear that, unlike these papers, we focus only on ℓ_1 minimization recovery. A further distinction of our analysis is the provision of a strong bound for sparse recovery, namely the bound for *all* nonnegative signals, which was not provided in [LMP08]. For further readings on compressive sampling techniques and analysis using sparse matrices, we refer the interested reader to the survey article of [GI10], which overviews recent results, including our own work presented here.

2.3 Contributions

We introduce sparse measurement matrices that result from adding perturbations to the adjacency matrices of expander graphs with a small critical expansion coefficient, hereby referred to as *minimal expanders*. We show that when ℓ_1 minimization is used to reconstruct nonnegative vectors, these constructions allow the recovery of sparse signals with many more nonzero entries—almost three orders of magnitude greater—than the existing theoretical results for sparse measurement matrices, namely the results of [BI08] and [BEZ08]. Please refer to Figure 2.5 for details. We provide

theoretical upper bounds for the so-called "weak" and "strong" recovery thresholds when ℓ_1 minimization is used. These bounds are very close (order-wise) to the bounds of Gaussian matrices for the nonnegative case (Section 2.7 and Figures 2.3 and 2.4). Furthermore, by carefully examining ℓ_1 minimization for sparse matrices, we deduce certain uniqueness results for the nonnegative solution of the linear equation when constant column sum matrices are used (see Section 2.5.1). We exploit this fact later to find faster alternatives to ℓ_1 minimization. In particular we present a novel recovery algorithm that directly leverages the minimal expansion property, and we prove that it is both optimal and robust to noise (Section 2.8).

One critical innovation of our work is that for expander graphs in the context of compressed sensing, we require a much smaller expansion coefficient in order to be effective. Throughout the literature, several sparse matrix constructions rely on adjacency matrices of expander graphs [BGI$^+$08, BI08, XH07b, JXHC09, Ind08]. In these works, the technical arguments require very large expansion coefficients, in particular, $1 - \epsilon \geq 3/4$, in order to guarantee a large number of unique neighbors [SS96] to the expanding sets. Our analysis is the first to obtain error-correction results through small expansion ($\epsilon > 1/2$). In fact we show that the *minimal expansion* we use in our constructions is actually necessary for any matrix that works for compressive sensing (even without the nonnegativity assumption). See Section 2.6.2 for more details on this.

2.4 Preliminaries

Recall that the goal of compressive sampling is to recover a sparse vector $\mathbf{x}$ from a set of under-determined linear equations. In many real-world applications the original data vector is nonnegative, which is the case that we will focus on in this chapter. The original problem of compressed sensing for the nonnegative input vectors is the

following:

$$\min_{\mathbf{Ax}=\mathbf{y},\mathbf{x}\geq 0} \|\mathbf{x}\|_0, \tag{2.4.1}$$

where $\mathbf{A}^{m\times n}$ is the measurement matrix, $\mathbf{y}^{m\times 1}$ is called the observation vector, $\mathbf{x}^{n\times 1}$ is the unknown vector which is known to be k-sparse (i.e., has only k nonzero entries), and where $\|\cdot\|_0$ is ℓ_0 norm (i.e., the number of nonzero entries of a given vector). (2.4.1) solves for the sparsest nonnegative solution in the constraint set $\{\mathbf{x} \mid \mathbf{Ax} = \mathbf{y}\}$. The typical situation is that $n > m > k$. Donoho and Tanner have shown in [DT05b] that, for a class of matrices $\mathbf{A}$ maintaining a so-called outwardly k-neighborly property and $\mathbf{x}$ being at most k-sparse, the solution to (2.4.1) is unique and an be recovered via the following linear programming problem:

$$\min_{\mathbf{Ax}=\mathbf{y},\mathbf{x}\geq 0} \|\mathbf{x}\|_1. \tag{2.4.2}$$

They also show that i.i.d. Gaussian random $m \times n$ matrices with $m = n/2$ are outwardly $m/8$-neighborly with high probability, and thus allow the recovery of $n/16$-sparse vectors $\mathbf{x}$ via linear programming. They further define a "weak" neighborly notion, based upon which they show that the same Gaussian random matrices allow perfect recovery of *almost* all $0.279n$-sparse nonnegative vectors $\mathbf{x}$ via ℓ_1 optimization for sufficiently large n.

Here, we primarily seek the answer to a similar question when the measurement matrix $\mathbf{A}$ is sparse. In particular our analysis lead to the adjacency matrices of a particular class of expander graphs as candidate solutions for appropriate sparse $\mathbf{A}$. An unbalanced expander graph is defined in the following paragraph. For conciseness, we denote a bipartite graph as $\mathcal{G} = (X, Y, E)$, where X, Y, and E are the set of left nodes, right nodes, and the edges of $\mathcal{G}$, respectively. Also, from now on throughout this dissertation, the set of the neighbors of any subset S of nodes is denoted by $\Gamma(S)$.

Definition 1. *A left regular bipartite graph $\mathcal{G}(X, Y, E)$ with X and Y as the set of left and right vertices, and regular left degree d is called a (k,ϵ)-unbalanced expander, if for every $S \subset X$ with $|S| \leq k$, the following holds: $|\Gamma(S)| \geq |S|d(1-\epsilon)$.*

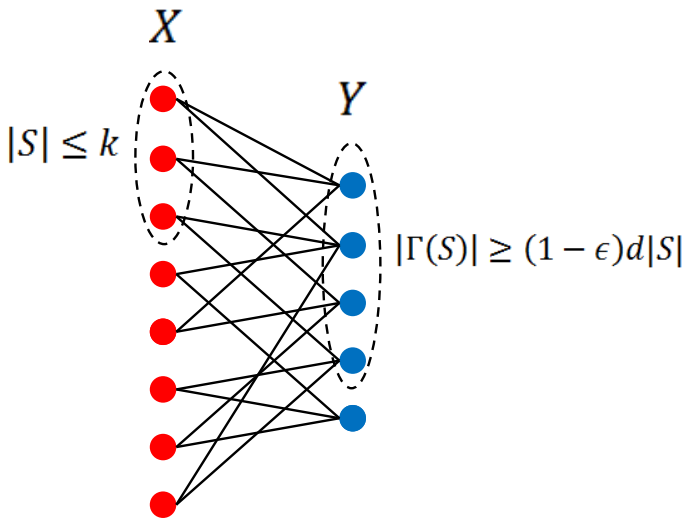


Figure 2.1: A pictorial example of a (k, ϵ)-expander graph

A pictorial demonstration of a (k, ϵ) expander graph is shown in Figure 2.1. Our aim is to analyze the outwardly neighborly conditions for this class of matrices and come up with sparse structures that allow the recovery of vectors with sparsity proportional to the number of equations.

2.5 Analysis of ℓ_1 Minimization

We begin by stating an equivalent version of the outwardly neighborly condition which is in fact similar to the null space property that was mentioned in the introduction, but for the nonnegative case. Later we show that this has a much more mundane interpretation for the special case of regular bipartite graphs, namely a combinatorial null space condition. We leverage this condition to derive bounds for the successful recovery of sparse signals when particular sparse matrices are used.

2.5.1 Null Space and Uniqueness Conditions

As mentioned in the introduction, the success of ℓ_1 minimization in recovering sparse signals can be characterized by the null space condition. This condition has been

previously stated for the general nondefinite sign signals in a couple of papers [SXH08, FN03, LN06, Zha05]. We present a similar condition for the success of ℓ_1 minimization in recovering nonnegative signals. We also show that under the assumption that the measurement matrix has constant column sum, this condition is equivalent to the uniqueness of *any* nonnegative solution to the under-determined system of linear equations.

We present the first theorem in the same style as in [DT05b].

Theorem 2.5.1. *Let* $\mathbf{A}$ *be an* $m \times n$ *matrix and* k *be a positive integer. The following two statements are equivalent:*

1. *For every nonnegative vector* $\mathbf{x}_0$ *with at most* k *nonzero entries,* $\mathbf{x}_0$ *is the unique solution to (2.4.2) with* $\mathbf{y} = \mathbf{A}\mathbf{x}_0$.

2. *For every vector* $\mathbf{w} \neq 0$ *in the null space of* $\mathbf{A}$*, and every index set* $S \subset \{1, 2, ..., n\}$ *with* $|S| = k$ *such that* $\mathbf{w}_{S^c} \geq 0$*, it holds that* $\sum_{i=1}^{n} w_i > 0$.

Here S^c *is the complement set of* S *in* $\{1, 2, ..., n\}$ *and* $\mathbf{w}_S$ *denotes the sub-vector of* $\mathbf{w}$ *constructed by those elements indexed in* S*.* $|S|$ *means the cardinality of the set* S*.*

Theorem 2.5.1 is in fact the counterpart of Theorem 1 of [SXH08] for nonnegative vectors. It gives a necessary and sufficient condition on the matrix $\mathbf{A}$, such that all the k-sparse vectors $\mathbf{x}_0$ can be recovered using (2.4.2). The condition is essentially that if a nonzero vector in the null space of $\mathbf{A}$ happens to have $n - k$ nonnegative entries, then the sum of all its entries must be positive. We call this property the nonnegative null space property.

Proof. Suppose $\mathbf{A}$ has the nonnegative null space property. We assume $\mathbf{x}_0$ is k-sparse and show that under the mentioned null space condition, the solution to (2.4.2) produces $\mathbf{x}_0$. We denote by $\mathbf{x}_1$ the solution to (2.4.2). Let S be the support set of $\mathbf{x}_0$.

We can write:

$$\begin{aligned} \|\mathbf{x}_1\|_1 &= \|\mathbf{x}_0 + (\mathbf{x}_1 - \mathbf{x}_0)\|_1 \\ &= \sum_{i=1}^{n} (\mathbf{x}_0)_i + (\mathbf{x}_1 - \mathbf{x}_0)_i \qquad (2.5.1) \\ &= \|\mathbf{x}_0\|_1 + \sum_{i=1}^{n} (\mathbf{x}_1 - \mathbf{x}_0)_i, \qquad (2.5.2) \end{aligned}$$

where $(\mathbf{x}_0)_i$ and $(\mathbf{x}_1 - \mathbf{x}_0)_i$ are the i^{th} entry of $\mathbf{x}_0$ and $\mathbf{x}_1 - \mathbf{x}_0$, respectively. The reason (2.5.1) and (2.5.2) are true is that $\mathbf{x}_1$ and $\mathbf{x}_0$ are both nonnegative vectors and their ℓ_1-norm is simply the sum of their entries. Now, if $\mathbf{x}_1$ and $\mathbf{x}_0$ are not equal, since $\mathbf{x}_1 - \mathbf{x}_0$ is in the null space of $\mathbf{A}$ and is nonnegative on S^c (because S is the support set of $\mathbf{x}_0$) we can write:

$$\sum_{i=1}^{n} (\mathbf{x}_1 - \mathbf{x}_0)_i > 0, \qquad (2.5.3)$$

which implies

$$\|\mathbf{x}_1\|_1 > \|\mathbf{x}_0\|_1.$$

But we know that $\|\mathbf{x}_1\|_1 \leq \|\mathbf{x}_0\|_1$ from the construction. This means we should have $\mathbf{x}_1 = \mathbf{x}_0$.

Conversely, suppose there is a nonzero vector $\mathbf{w}$ in the null space of $\mathbf{A}$ and a subset $S \subset \{1, 2, ..., n\}$ of size k with $\mathbf{w}_{S^c} \geq 0$ and $\sum_{i=1}^{n} w_i \leq 0$. We construct a nonnegative vector $\mathbf{x}_0$ supported on S, and show that there exists another nonnegative vector $\mathbf{x}_1 \neq \mathbf{x}_0$ such that $\mathbf{A}\mathbf{x}_0 = \mathbf{A}\mathbf{x}_1$ and $\|\mathbf{x}_1\|_1 \leq \|\mathbf{x}_0\|_1$. This means that $\mathbf{x}_0$ is not the unique solution to (2.4.2) with $\mathbf{y} = \mathbf{A}\mathbf{x}_0$ and will complete the proof. For simplicity we may assume $S = \{1, 2, ..., k\}$. We construct a nonnegative vector $\mathbf{x}_0$ supported on S that cannot be recovered via ℓ_1 minimization of (2.4.2). Without loss of generality we write

$$\mathbf{w} = (-\mathbf{w}_{S^-} \ \mathbf{w}_{S^+} \ \mathbf{w}_{S^c})^T, \qquad (2.5.4)$$

where $\mathbf{w}_{S^-}$ and $\mathbf{w}_{S^+}$ are both nonnegative vectors. Now set

$$\mathbf{x}_0 = (\mathbf{w}_{S^-} \ \mathbf{w}_{S^+} \ 0)^T, \ \mathbf{x}_1 = (0 \ 2\mathbf{w}_{S^+} \ \mathbf{w}_{S^c})^T. \qquad (2.5.5)$$

■

2.5.2 Null Space of Adjacency Matrices

Here, we will be considering measurement matrices $\mathbf{A}$ with two main properties: sparse and constant column sum. This class of matrices includes measurement matrices obtained from the adjacency matrices of regular left degree bipartite graphs (i.e., 0–1 matrices with a constant number of ones in each column), as well as the perturbed expanders introduced in Section 2.6.1. For this class of matrices we actually show that the condition for the success of ℓ_1 recovery is simply equivalent to the set $\{\mathbf{x}|\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{x}_0, \mathbf{x} \geq 0\}$ having a "single" element. To this end, we prove the following lemma and theorem.

Lemma 2.5.2. *Let $\mathbf{A}^{m\times n}$ be a matrix with constant column sum d. For any vector $\mathbf{w}$ in the null space of $\mathbf{A}$, the following is true*

$$\sum_{i=1}^{n} w_i = 0. \tag{2.5.6}$$

Proof. Let $\mathbf{1} = (1, 1, ..., 1)^T$ be the $1 \times m$ vector of all 1s. We have:

$$\mathbf{A}\mathbf{w} = 0 \Rightarrow \mathbf{1}^T\mathbf{A}\mathbf{w} = 0 \Rightarrow d\sum_{i=1}^{n} w_i = 0, \tag{2.5.7}$$

where d is the column sum of $\mathbf{A}$. ■

Theorem 2.5.3. *Let $\mathbf{A} \in \mathcal{R}^{m\times n}$ be a matrix with constant column sum. Then the following three statements are equivalent.*

1. *For all nonnegative k-sparse $\mathbf{x}_0$ with an arbitrary (a particular) support set S, it holds that $\{\mathbf{x}|\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{x}_0, \mathbf{x} \geq 0\} = \{\mathbf{x}_0\}$.*

2. *For every vector $\mathbf{w} \neq 0$ in the null space of $\mathbf{A}$, and every (a particular) index set $S \subset \{1, 2, ..., n\}$ with $|S| = k$ such that $\mathbf{w}_{S^c} \geq 0$, it holds that $\sum_{i=1}^{n} w_i > 0$.*

3. *For every (a particular) subset $S \subset \{1, 2, ..., n\}$ with $|S| = k$, there exists no vector $\mathbf{w} \neq 0$ in the null space of $\mathbf{A}$ such that $\mathbf{w}_{S^c} \geq 0$.*

Theorems 2.5.1 and 2.5.3 show that for the class of matrices with constant column sum, the condition for the success of ℓ_1 recovery is simply the condition for there being a "unique" vector in the constraint set $\{\mathbf{x}|\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{x}_0, \mathbf{x} \geq 0\}$. In this case, *any* optimization problem, e.g., $\min_{\mathbf{x}\geq 0, \mathbf{A}\mathbf{x}=\mathbf{y}} \|\mathbf{x}\|_2$, would also recover the desired $\mathbf{x}_0$.

Proof. First, we show that for any matrix $\mathbf{A}$, the statements 1 and 3 of Theorem 2.5.3 are equivalent. Suppose that condition 3 holds for a specific subset $S \subset \{1, 2, \cdots, n\}$. Consider a nonnegative $n \times 1$ vector $\mathbf{x}_0$ supported on S. If there exists another nonnegative vector $\mathbf{x}_1$ with the property that $\mathbf{A}\mathbf{x}_1 = \mathbf{A}\mathbf{x}_0$, then $\mathbf{x}_1 - \mathbf{x}_0$ would be a vector in the null space of $\mathbf{A}$ which is also nonnegative on S^c, due to the nonnegativity of $\mathbf{x}_1$ and the fact that S is the support set of $\mathbf{x}_0$. This contradicts the earlier assumption of condition 3.

The proof of the converse is also straightforward. Suppose that condition 1 holds for a specific subset S and all nonnegative vectors $\mathbf{x}_0$ supported on S. Let's say one can find a nonzero vector $\mathbf{w}$ in the null space of $\mathbf{A}$ with $\mathbf{w}_{S^c} \geq 0$. As in the proof of Theorem 2.5.1, we may write $\mathbf{w}$ as

$$\mathbf{w} = (-\mathbf{w}_{S^-} \ \mathbf{w}_{S^+} \ \mathbf{w}_{S^c})^T, \tag{2.5.8}$$

where $\mathbf{w}_{S^-}$ and $\mathbf{w}_{S^+}$ are both nonnegative vectors. Now if

$$\mathbf{x}_0 = (\mathbf{w}_{S^-} \ \mathbf{w}_{S^+} \ 0)^T, \ \mathbf{x}_1 = (0 \ 2\mathbf{w}_{S^+} \ \mathbf{w}_{S^c})^T, \tag{2.5.9}$$

then $\mathbf{x}_0$ and $\mathbf{x}_1$ are distinct nonzero vectors and belong to the set $\{\mathbf{x}|\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{x}_0, \mathbf{x} \geq 0\}$. This is a contradiction to the assumption we earlier made.

So far we have shown that for any matrix $\mathbf{A}$ the two statements 1 and 3 are equivalent. Now we show that for matrices with constant column sum the two statements 2 and 3 are equivalent. We make use of Lemma 2.5.2, that for this special class of matrices with constant column sum, every vector in the null space has a zero sum of entries. Therefore, statement 2 can be true only if there is no $\mathbf{w}$ in the null space of $\mathbf{A}$ with $\mathbf{w}_{S^c} \geq 0$. Conversely if the condition in statement 3 holds, then there is no

$\mathbf{w} \in \mathcal{N}(\mathbf{A}) \setminus \{0\}$ such that $\mathbf{w}_{S^c}$ is nonnegative and therefore statement 2 is also true. ■

Corollary 2.5.4 (Corollary of Theorem 2.5.3)**.** *The three conditions of Theorem 2.5.3 are equivalent to the following statement.*

- *Every nonzero vector* $\mathbf{w}$ *in the null space of* $\mathbf{A}$ *has at least* $k+1$ *negative entries.*

Proof. Follows directly from the third statement of Theorem 2.5.3. ■

The results of this section show how the structure of the null space of the measurement matrix is related to the recoverability of sparse vectors. Thus, to achieve our primary goal of constructing optimal sparse measurement matrices, we need to find appropriate bipartite graphs, the adjacency matrices of which satisfy the nonnegative null space properties up to a maximal sparsity level (i.e., maximum recoverable number of nonzero entries). In what follows, we introduce class of sparse matrices based on a generalized notion of expander graphs, called perturbed expanders. Using a linear algebraic view of expanders, we are able to make a probabilistic analysis of the null space property for this class of sparse matrices.

2.6 Expander Graphs and Their Linear Algebraic View

Before proceeding, we define the notion of complete rank that appears as an important linear algebraic property of matrices in the analysis of sparse recovery.

Definition 2. *For a matrix* $\mathbf{A}^{m\times n}$ *we define the* complete rank *of* $\mathbf{A}$ *(denoted by* $\mathcal{H}(\mathbf{A})$*) to be the maximum integer* r_0 *with the property that every* r_0 *columns of* $\mathbf{A}$ *are linearly independent. In other words,* $\mathcal{H}(\mathbf{A}) = \min_{\mathbf{w}\in\mathcal{N}(\mathbf{A}),\mathbf{w}\neq 0}(|Supp(\mathbf{w})| - 1)$*, where* $Supp(\mathbf{w})$ *is the support set of* $\mathbf{w}$*.*

This notion is also known in linear algebra as "Kruskal rank" (see [Kru77]). It has also been given other names in the literature. The complete rank of a matrix $\mathbf{A}$ is equivalent to the minimum Hamming distance (weight) of the null space of $\mathbf{A}$ minus one [Wol83], which can also be recognized as the "spark" of the matrix minus one [JM].

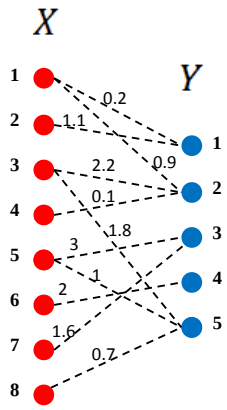
Expander graphs were defined in Section 2.4. Here, we also define generalized bipartite graphs that allow us to consider adjacency matrices with nonunitary entries.

Definition 3. *Let* $\mathbf{A}$ *be a nonnegative* $m \times n$ *matrix. Consider the weighted bipartite graph* $\mathcal{G} = (X, Y, E, W)$ *where* $X = \{v_1, v_2, \cdots, v_n\}$ *and* $Y = \{u_1, u_2, \cdots, u_m\}$ *are the sets of nodes,* E *is the set of edges, and* W *is the set of weights assigned to the edges, where* G *has the following property: for every nonzero entry* $\mathbf{A}_{ij}$ *of the matrix* $\mathbf{A}$*,* v_j *and* u_i *are connected together with an edge* e_{ij} *of weight* $w_{ij} = \mathbf{A}_{ij}$*. For the zero entries of* $\mathbf{A}$*, there is no edge connecting the corresponding nodes in* G*. We call* G *the generalized bipartite graph of* $\mathbf{A}$*, and refer to* $\mathbf{A}$ *as the generalized adjacency matrix of* G*.*

Example 1. A weighted partite graph and its corresponding generalized adjacency matrix in Figure 2.2. Note that the labels on the edges on the graph correspond to the entries of the adjacency matrix. If there is no edge between two nodes, the corresponding entry is zero in the adjacency matrix.

Note that the notions of "neighbor" of a vertex and "expansion" in a weighted bipartite graph are the same as in a unitary bipartite graph. So for instance, the neighbors of a node v in this graph are the set of nodes to which v is connected with an edge of nonzero weight. The following lemma connects the expansion property of a (generalized) bipartite graph to the complete rank of its (generalized) adjacency matrix:

Lemma 2.6.1. *Let* $\mathbf{A}$ *be a nonnegative matrix with exactly* d *nonzero entries in each column. The generalized bipartite graph of* $\mathbf{A}$ *is a* $(\mathcal{H}(\mathbf{A}), \frac{d-1}{d})$ *expander.*



$$\mathbf{A} = \begin{bmatrix} 0.2 & 1.1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.9 & 0 & 2.2 & 0.1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 & 1.6 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.7 \end{bmatrix}$$

Figure 2.2: A weighted bipartite graph and its generalized adjacency matrix

Proof. Let $\mathcal{G} = (X, Y, E, W)$ be the generalized bipartite graph of $\mathbf{A}$. If $S \subset X$ with $|S| \leq \mathcal{H}(\mathbf{A})$ then the columns of $\mathbf{A}$ corresponding to the elements of S are linearly independent. So the sub-matrix $\mathbf{A}_S$ of $\mathbf{A}$, produced by only those columns indexed in S must be of full rank. Therefore, $\mathbf{A}_S$ must have at least $|S|$ nonzeros rows, which is equivalent to $|\Gamma(S)| \geq |S| = |S|d(1 - \frac{d-1}{d})$. ∎

From the above proof, it follows immediately that:

$$\forall S \subseteq X,\ |\Gamma(S)| \geq \min(|S|, \mathcal{H}(\mathbf{A})). \tag{2.6.1}$$

The notion of complete rank is closely related to the expansion property, and is on the other hand, tied in with the null space characterization we are trying to establish, as elucidated in the following theorem.

Theorem 2.6.2. *Let* $\mathbf{A}$ *be a nonnegative matrix with exactly* d *nonzero entries in each column. For every nonzero vector* $\mathbf{w}$ *in the null space of* $\mathbf{A}$*, the number of negative elements of* $\mathbf{w}$ *is at least* $\frac{\mathcal{H}(\mathbf{A})}{d}$.

Proof. Let X and Y be the sets of left and right vertices of the generalized bipartite graph of $\mathbf{A}$. Let $S^+_{\mathbf{w}}$ be the set of vertices in X corresponding to the positive elements of $\mathbf{w}$, and $S^-_{\mathbf{w}}$ be the set of vertices corresponding to the negative elements.[3] Let

[3]We interchangeably use S and its variations to denote a set of vertices or a support set of a vector.

$S_{\mathbf{w}} = S^+_{\mathbf{w}} \cup S^-_{\mathbf{w}}$. Since $\mathbf{Aw} = 0$, we must have $\Gamma(S^+_{\mathbf{w}}) = \Gamma(S^-_{\mathbf{w}}) = \Gamma(S_{\mathbf{w}})$, since otherwise, there exists a vertex in Y connected to exactly one of the sets $S^+_{\mathbf{w}}$ or $S^-_{\mathbf{w}}$. Therefore, the coordinate of the vector $\mathbf{Aw}$ corresponding to this node will not sum up to zero. On the other hand, from the definition of $\mathcal{H}(\mathbf{A})$, we must have $|S_{\mathbf{w}}| > \mathcal{H}(\mathbf{A})$. The number of edges emanating from $S^-_{\mathbf{w}}$ is $d|S^-_{\mathbf{w}}|$, which is at least as large as the number of its neighbors $|\Gamma(S^-_{\mathbf{w}})|$. Hence:

$$d|S^-_{\mathbf{w}}| \geq |\Gamma(S^-_{\mathbf{w}})| = |\Gamma(S_{\mathbf{w}})| \geq \mathcal{H}(\mathbf{A}),$$

where the last inequality is a consequence of (2.6.1). ■

We now turn to the task of constructing adjacency matrices with complete rank proportional to the dimension n. Throughout this paper, all the thresholds that we achieve are asymptotic, i.e., they hold for the regime of very large n and m.

2.6.1 Perturbed Expanders

When n and $m = \beta n$ are large, we are interested in constructing 0-1 matrices $\mathbf{A}^{m\times n}$ with d (constant) 1's in each column such that $\mathcal{H}(\mathbf{A})$ is proportional to n. Furthermore, the maximum achievable value of $\frac{\mathcal{H}(\mathbf{A})}{nd}$ is of significant interest, as it determines the largest sparsity fraction that can be reconstructed using ℓ_1 minimization (see Theorem 2.6.2). This is a very difficult question to address in full generality for the class of binary matrices. However, it turns out to be much easier if we allow for a *small* perturbation of the nonzero entries of $\mathbf{A}$.

Lemma 2.6.3. *For a matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ which is the adjacency matrix of a bipartite left d-regular graph, if in the submatrix formed by any r_0 columns of $\mathbf{A}$, every $r \leq r_0$ columns have at least r nonzero rows, then it is possible to perturb the nonzero entries of $\mathbf{A}$ and obtain another nonnegative matrix $\tilde{\mathbf{A}}$ through this procedure, with $\mathcal{H}(\tilde{\mathbf{A}}) \geq r_0$. Furthermore, the perturbations can be done in a way that the sum of each column remains a constant d, and all perturbations are rational numbers representable by a finite number of bits.*

Proof. We add a random set of perturbations $\Delta = \{\delta_{i,j} \mid 1 \leq i \leq m, 1 \leq j \leq n, \mathbf{A}_{i,j} \neq 0\}$ to the nonzero elements of $\mathbf{A}$, while leaving the zero elements of $\mathbf{A}$ intact. We denote the perturbed matrix by $\tilde{\mathbf{A}}$. The way the random perturbations are generated is as follows. Suppose without loss of generality that we scale the matrix up by a prime integer q before adding perturbations, so that the nonzero entries of $\mathbf{A}$ are equal to q, and suppose that $\delta_{i,j}$s are integers in $\{0, 1, \cdots, q-1\}$. For each nonzero entry $\mathbf{A}_{i,j}$ of $\mathbf{A}$, we independently choose $\delta_{i,j}$ uniformly at random from the set $\{0, 1, \cdots, q\}$. We look at the submatrix $\mathbf{A}_{sub}$ formed by arbitrary r_0 distinct columns of $\mathbf{A}$. Without loss of generality, we index these columns by the set $N_{sub} = \{1, \cdots, r_0\}$. Then, according to a famous graph theoretic result known as Hall's matching theorem (see Appendix 10.2), there exists a perfect matching between these r_0 columns and some r_0 rows, which we denote by the set $M_{sub} = \{i_1, i_2, ..., i_{r_0}\}$ accordingly. First, we bound the probability that Δ makes the submatrix $\tilde{\mathbf{A}}_{sub}$ have rank smaller than r_0, i.e., $\mathbb{P}\{rank(\tilde{\mathbf{A}}_{sub}) < r_0\}$. In order for the submatrix $\tilde{\mathbf{A}}_{sub}$ to have rank smaller than r_0, the determinant of the square matrix $\tilde{\mathbf{A}}_{M_{sub},N_{sub}}$ must be zero, namely $\det(\tilde{\mathbf{A}}_{M_{sub},N_{sub}}) = 0$. By definition, $\det(\tilde{\mathbf{A}}_{M_{sub},N_{sub}})$ is a polynomial over the variables of Δ, say $p(\Delta)$, which contains a product term $\delta_{1,i_1}\delta_{2,i_2}\cdots\delta_{r_0,i_{r_0}}$ which certainly has a nonzero coefficient (actually its coefficient is either 1 or -1). Therefore, invoking the well-known Schwartz-Zippel lemma for polynomials, we can assert that $\mathbb{P}\{p(\Delta) = 0\} \leq \frac{r_0}{q}$. Furthermore, the number of ways the submatrix $\mathbf{A}_{sub}$ can be chosen is $\binom{n}{r_0}$. Applying a union bound over all possible choices of $\mathbf{A}_{sub}$, we can write:

$$\mathbb{P}\{\mathcal{H}(\tilde{\mathbf{A}}) \leq r_0\} \leq \binom{n}{r_0}\frac{r_0}{q}. \tag{2.6.2}$$

For given n and r_0, we can choose a finite q large enough such that the right-hand side of (2.6.2) is sufficiently small. Therefore, there exists a choice of perturbations $\delta_{i,j}$ so that the resulting $\tilde{\mathbf{A}}$ satisfies $\mathcal{H}(\tilde{\mathbf{A}}) \geq r_0$. Furthermore, after scaling down the perturbed matrix by q, each perturbation $\delta_{i,j}$ is a rational number of the form $\frac{t}{q}$,

$0 \leq t < q$, and is therefore representable by a finite number of bits.

■

It is worth noticing that, after modifying $\mathbf{A}$ based on perturbations of Lemma 2.6.3, Theorems 2.5.1, 2.5.3, and 2.6.2, and Lemmas 2.5.2 and 2.6.1 all continue to hold for this class of matrices $\tilde{\mathbf{A}}$. The reason is as follows. First of all, note that Lemma 2.5.2 and Theorem 2.5.3 require only that $\mathbf{A}$ be constant column sum, which is true for $\tilde{\mathbf{A}}$. Theorem 2.5.1 assumes no restriction on the matrix. Finally, Lemma 2.6.1 and Theorem 2.6.2 are valid, since they hold for nonnegative matrices with a constant number of nonzero entries in each column, and $\tilde{\mathbf{A}}$ is such a matrix.

The conclusion of this section so far is that if one starts off with the adjacency matrix $\mathbf{A}$ of a regular bipartite graph and perturb its nonzero entries to obtain a nonzero constant column sum matrix $\tilde{\mathbf{A}}$ with $\mathcal{H}(\tilde{\mathbf{A}}) \geq r_0$, then the following guarantee exists: ℓ_1 minimization perfectly recovers a $\lceil \frac{r_0}{d} - 1 \rceil$-sparse nonegative vector $\mathbf{x}_0$ from the measurements $\tilde{\mathbf{A}}\mathbf{x}_0$. Our goal now becomes constructing $(r_0, \frac{d-1}{d})$ expanders with the ratio $\frac{r_0}{nd}$ as large as possible. In Section 2.7, we use a probabilistic method to show that the desired $(r_0 = \mu n, \frac{d-1}{d})$ expanders exist and provide thresholds for $\frac{\mu}{d}$. Before continuing, note that we are using a $1 - \epsilon \geq \frac{1}{d}$ expansion coefficient for perfect recovery, which is very small compared to other schemes that use expanders (see, e.g., [XH07b, BGI+08, IR08, BI08, XH07a, JXHC09]) and require expansion coefficients at least larger than $1 - \epsilon \geq \frac{3}{4}$. The small expansion coefficient $1 - \epsilon \geq \frac{1}{d}$ required in our analysis is indeed the *critical* expansion coefficient needed for the suitability of a sparse measurement matrix. We shortly digress in a subsection to discuss this a little further.

2.6.2 Necessity of Expansion for Compressive Sensing

Consider the following definition:

Definition 4. *Let $\mathcal{G} = (X, Y, E, W)$ be a weighted bipartite graph and $S \subset X$ be a*

set of nodes in X. Let M be a subset of edges of G. M is called a partial perfect matching saturating S, if:

- $|M| = |S|$.
- *each node of S is incident to exactly one edge in M.*
- *no node in Y is incident to more that one edge in M.*

We prove that any *good* sparse measurement matrix corresponds to an expander graph with a critical (minimal) expansion coefficient.

Theorem 2.6.4. *Let $\mathbf{A}$ be a $m \times n$ nonnegative matrix, with d nonzero entries per column, and assume that $\mathbf{A}$ allows the recovery of all r_0-sparse vectors. Then the generalized bipartite graph of $\mathbf{A}$ is an $(r_0, 1 - \frac{1}{d})$ expander.*

Proof. The statement holds for any recovery algorithm. In fact we show that if the generalized bipartite graph of $\mathbf{A}$ is not an $(r_0, 1 - \frac{1}{d})$ expander, then even a stronger recovery algorithm that magically knows the support of the vector, fails to recover some r_0-sparse vectors. Assume that the bipartite graph is not a $(r_0, 1 - \frac{1}{d})$ expander, i.e., there exists a set of $r \leq r_0$ columns that is adjacent to $r - 1$ (or fewer) rows. Therefore the rank of the submatrix corresponding to these r columns must be strictly smaller than r regardless of what the nonzero entries are. By selecting an adversary sparse signal supported exactly on these r columns, we see it is impossible for any algorithm to recover it, even if the support is known, since there is a rank loss in the corresponding measurement submatrix.

■

2.7 Recovery Thresholds of Compressive Sensing for Minimal Expanders

In summary of Section 2.6, if one can construct a $(r_0, 1-\frac{1}{d})$ expander graph with left degree d, then from the adjacency matrix of this expander, it is possible to obtain a measurement matrix that allows the recovery of *every* $\frac{r_0}{d}$ sparse nonnegative signals using ℓ_1 minimization. The explicit relationship between the ratio $\frac{m}{n}$ and the relative size, $\frac{r_0}{dn}$, of a recoverable set is not yet clear. In this section, by addressing the question of existence of appropriate expander graphs, we find this explicit relationship known as the strong recovery threshold. We further derive a weak recovery threshold that is concerned with the recovery of *almost all* sparse vectors of a certain sparsity.

2.7.1 Strong Bound

For fixed values of $n > m > r_0$ and d we are interested in the existence of $(r_0,\epsilon = \frac{d-1}{d})$ expanders with constant left degree d. There are a few previous works that address the construction of expanders (random or deterministic), and try to find the relationships between their parameters. In [BM01] for instance, it has been shown that for any value of $\beta = \frac{m}{n}$ and $0 \leq \epsilon \leq 1$, there exists a left degree d bipartite $(\mu n,\ \epsilon)$ expander for some $0 \leq \mu \leq 1$ and some constant (not growing with n) d. Also, an explicit construction of constant regular left degree lossless (with $1-\epsilon$ arbitrarily close to 1) expanders is given in [CRVW02]. As a consequence of the results of the latter, for any fixed ϵ it is possible to explicitly construct a (k_{max},ϵ) expander with $k_{max} = O(\frac{m}{d\epsilon})$, and $d = O(\frac{\log(\beta)}{\epsilon})$. The main reason we cannot use these results directly here is that the relationship between the expansion factor $(1-\epsilon)$ and the left degree of the graph (d) is not clear in any of the previous works (even order wise). Besides, in most of the previous arguments, the relationship between the size of expansion set k, the left set size m and the factor ϵ is expressed in terms of order functions, and the explicit constants are omitted. Our attempt here is to derive those constants explicitly.

We use the standard first moment method argument to prove the existence of our expanders of interest, namely (r_0,$\epsilon = \frac{d-1}{d}$) expanders, for appropriate n, m, r_0, and d. The main result is given as follows:

Theorem 2.7.1. *For sufficiently large n, with $m = \beta n$ and $r_0 = \mu n$, there exists a bipartite graph with n left vertices and m right vertices, which is a $(r_0, \frac{d-1}{d})$ expander, if*

$$d > \frac{H(\mu) + \beta H(\frac{\mu}{\beta})}{\mu \log(\frac{\beta}{\mu})}, \tag{2.7.1}$$

where $H(.)$ is the Shannon entropy function defined as $H(x) = x \log_2 \frac{1}{x} + (1 - x) \log_2 \frac{1}{1-x}$.

More important is the question of how big the ratio $\frac{\mu}{d}$ can be, since we earlier proved that we can recover vectors with sparsity up to $\frac{r_0}{d} = \frac{\mu}{d} n$. A combination of the previous derivations and Theorem 2.7.1 directly implies the following theorem, stating the strong sparsity threshold for sparse measurement matrices.

Theorem 2.7.2 (Strong Threshold)**.** *For a fixed $\beta = \frac{m}{n}$, let d^* and μ^* be the solutions of the following optimization program*

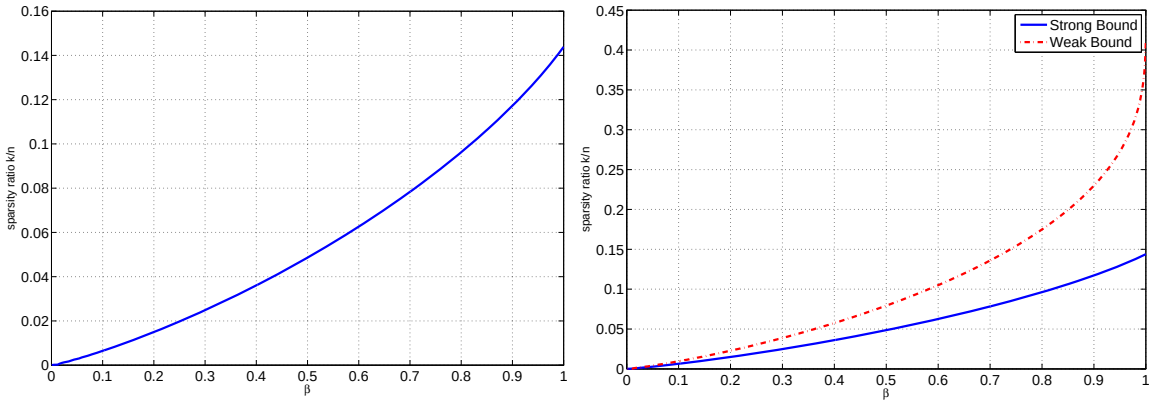
$$\begin{aligned} &\textit{maximize } \frac{\mu}{d} \\ &\textit{s.t } \left\{ 0 \leq \mu \leq 1,\ d \in \mathbb{Z}^+,\ d > \frac{H(\mu) + \beta H(\frac{\mu}{\beta})}{\mu \log(\frac{\beta}{\mu})} \right\}. \end{aligned}$$

For sufficiently large n, there exists a sparse measurement matrix $\mathbf{A}$ with d^* nonzero entries in each column, so that every $\frac{\mu^* n}{d^*}$-sparse nonnegative vector $\mathbf{x}$ can be recovered from $\mathbf{Ax}$ using ℓ_1 minimization.

Figure 2.3a illustrates the strong recovery threshold for different values of β, derived from Theorem 2.7.2.

2.7.2 Weak Bound

We are now interested in deriving conditions for recovering a specific support set S of size $k = \alpha n$, rather than obtaining a worst-case bound for matrices that work for



(a) Strong bound. (b) Weak vs strong bound.

Figure 2.3: Recoverable sparsity size, weak achievable bound of Section 2.7.2 and the strong achievable threshold of (2.7.1). β is the ratio $\frac{m}{n}$.

all support sets. Recall that $m = \beta n$, left degree is d, and define $\gamma_1 := (1 - e^{-d\frac{\alpha}{\beta}})\beta$.

Theorem 2.7.3 (Weak Threshold)**.** *Define the function*

$$F(\rho_1, \rho_2) := \alpha H(\frac{\rho_1}{\alpha}) + (1-\alpha)H(\frac{\rho_2}{1-\alpha}) + \beta H(\frac{\rho_1+\rho_2}{\beta}) + d(\rho_1+\rho_2)\log(\frac{\rho_1+\rho_2}{\beta}). \tag{2.7.2}$$

For every α such that $F(\rho_1, \rho_2) < 0$ for every ρ_1, ρ_2 that satisfy $\rho_1 < \alpha, \rho_2 < 1 - \alpha, \rho_1 + \rho_2 < \gamma_1$, a randomly selected subset of size $k = \alpha n$ is recoverable using ℓ_1 minimization of (2.4.2) from a random perturbed matrix $\tilde{\mathbf{A}}$ with probability $1 - o(1)$.

The bound that results from Theorem 2.7.3 is plotted in Figure 2.3b and has been compared to the strong threshold previously achieved. Also, the comparison of these bounds with those of dense Gaussian i.i.d. matrices for the nonnegative case that were obtained in the paper [DT05b] is illustrated in Figure 2.4. Note that compared with the performance bounds achievable for Gaussian dense matrices, the provable theoretical performance bounds from our derivations are generally smaller, but they are getting closer to the performance bounds achievable for Gaussian dense matrices (See Figure 2.4) and represent a step towards closing the gap between the provable bounds for dense and sparse measurement matrices. Finally, to highlight the importance of our achieved bounds, we have compared the strong threshold of this

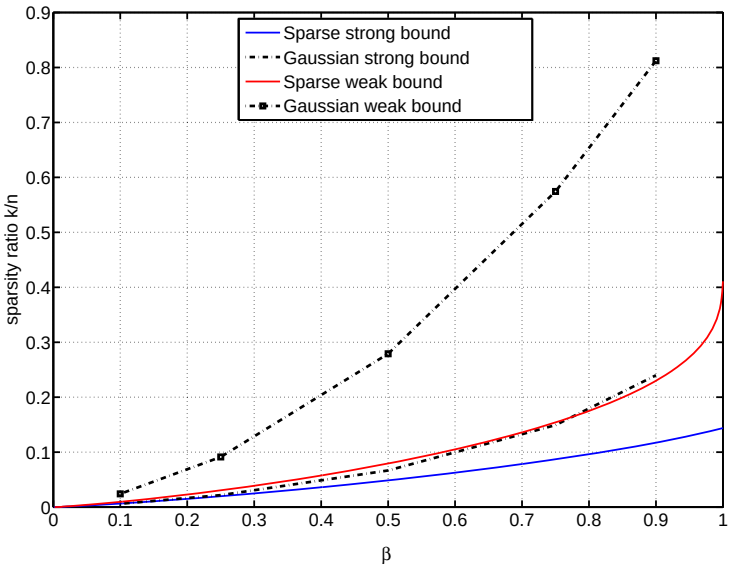


Figure 2.4: Comparison of weak and strong bounds for dense i.i.d. Gaussian matrices (and nonnegative signals) from [DT05b] with those of the current paper for sparse matrices. β here is equal to $\frac{m}{n}$.

paper with the thresholds of [BGI+08] that also uses expander graphs with $\epsilon \leq 1/6$ in Figure 2.5. Ostensibly, the use of minimal expanders has increased the theoretical recovery thresholds by almost three orders of magnitude.

The full proof of Theorem 2.7.3 is given in Section 2.10. The key argument is a matching condition for the recoverability of vectors supported on a specific subset S. The condition involves looking at the two-hop graph from S and checking if all sets of size up to $|\Gamma(S)| + 1$ are saturated by a partial perfect matching.

Lemma 2.7.4. *Given a set S, consider $\Gamma(S)$ and denote $S_2 = \Gamma(\Gamma(S)) \setminus S$. Let the bipartite two-hop graph of S be denoted by $B_S = (S \cup S_2, \Gamma(S \cup S_2))$. If every subset $S' \subset S \cup S_2$ of size $|S'| \leq |\Gamma(S)| + 1$ has minimal expansion: $|\Gamma(S')| \geq |S'|$ then there is a perturbation of nonzero entries of $\mathbf{A}$ resulting in the matrix $\tilde{\mathbf{A}}$ so that any nonnegative vector $\mathbf{x}_0$ supported on S can be recovered from $\mathbf{y} = \tilde{\mathbf{A}}\mathbf{x}_0$ using the optimization formulation (2.4.2).*

Proof. Consider the two-hop bipartite graph of S and let $C = (S \cup S_2)^c$ denote the remainder of the nodes in X. Further let $\mathbf{A}_S$ denote the submatrix of $\mathbf{A}$ corresponding

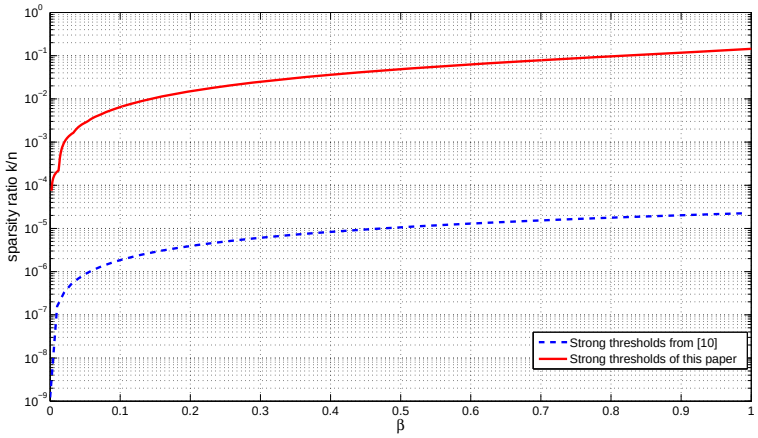


Figure 2.5: Comparison of size of recoverable sparsity (strong bound) of this paper with those from [BGI+08]. $\beta = \frac{m}{n}$.

to B_S. By Hall's theorem since every subset of $S \cup S_2$ of size up to $|\Gamma(S)| + 1$ has expansion equal to its size, it must also be saturated by a partial perfect matching. In other words, B_S corresponds to a $(|\Gamma(S) + 1|, 1 - \frac{1}{d})$ expander. Therefore by Lemma 2.6.3 it is possible to perturb its nonzero entries to obtain a matrix $\tilde{\mathbf{A}}_S$ with $\mathcal{H}(\tilde{\mathbf{A}}_S) \geq |\Gamma(S)|$.

To show that a set S can be recovered, it suffices to show that every nonzero vector $\mathbf{w}$ in the nullspace of $\tilde{\mathbf{A}}$ cannot have all its negative components in S. Assume otherwise: that some $\mathbf{w}$ has indeed all its negative support $S^-_{\mathbf{w}} \subseteq S$. Observe now that C cannot contain any of the positive support of $\mathbf{w}$, because every equation that is adjacent to a positive element must also be adjacent to a negative elements (since the matrix coefficients are nonnegative) and $\Gamma(S^-_{\mathbf{w}})$ does not intersect $\Gamma(C)$. Therefore the whole support of $\mathbf{w}$ must be contained in $S \cup S_2$.

Now we can show that $|S_{\mathbf{w}}| \leq |\Gamma(S)|$. Assume otherwise, that $|S_{\mathbf{w}}| > |\Gamma(S)|$. Then we could select a subset of $K \subseteq S_{\mathbf{w}}$ such that $|K| = |\Gamma(S)|+1$. This set K satisfies our assumption and is contained in B_S and therefore must have the minimal expansion $|\Gamma(K)| \geq |K| = |\Gamma(S)| + 1$. But since $\Gamma(S_{\mathbf{w}}) = \Gamma(S^-_{\mathbf{w}}) \subseteq \Gamma(S)$ and $K \subseteq S_{\mathbf{w}}$ (recall that $S^-_{\mathbf{w}} \subseteq S$ by assumption), it must hold that $|\Gamma(K)| \leq |\Gamma(S)|$, which contradicts the minimal expansion inequality.

Therefore, $|S_{\mathbf{w}}|$ must be saturated by a partial perfect matching which means that we can find a full rank submatrix $\mathbf{A}_{\mathbf{w}}$ (corresponding to that partial matching) such that $\mathbf{A}_{\mathbf{w}}\mathbf{w}_S = 0$ (where $\mathbf{w}_S$ means the vector $\mathbf{w}$ restricted to its support). Since $\mathbf{A}_{\mathbf{w}}$ is full rank, $\mathbf{w}$ must be the all-zeros vector which contradicts the assumption that $S^-_{\mathbf{w}}$ can be contained in S. ■

2.8 Fast Algorithm

We now describe a fast algorithm for the recovery of sparse nonnegative vectors from noiseless or noisy measurements. This algorithm relies on the minimal expansion we described in Section 2.6.2. We employ a $(kd+1,1-\frac{1}{d})$ expander and perturb it as Lemma (2.6.3) to obtain a sparse nonnegative matrix $\tilde{\mathbf{A}}$ with $\mathcal{H}(\tilde{\mathbf{A}}) \geq kd+1$. The algorithm has two variations, one that works specifically for k-sparse signals with no measurement noise, and one that also accounts for additive noise. When measurement noise is present, the observation vector is given by the following equation:

$$\mathbf{y} = \tilde{\mathbf{A}}\mathbf{x} + \mathbf{v}. \tag{2.8.1}$$

In either case, the sparsity k of the unknown signal is given to the algorithm.

Note that in the noiseless case, Algorithm 1 transforms the original under-determined system of equations to an over-determined subset of equations. In Section 2.10, we provide theoretical justification for the effectiveness of this algorithm. Also, in next theorem we claim that Algorithm 1 is robust to measurement noise. The proof is provided in Section 2.10.

Theorem 2.8.1. *If* $\mathbf{A}$ *is the adjacency matrix of a* $(2k,\epsilon)$ *expander with* $\epsilon < 0.5$, $\mathbf{x}$ *is a* k*-sparse nonnegative vector and* $\hat{\mathbf{x}}$ *is the output of Algorithm 1 with* $p=1$ *for* $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{v}$, *then* $\|\mathbf{x}-\hat{\mathbf{x}}\|_1 \leq \frac{7-4\epsilon}{1-2\epsilon}\|\mathbf{v}\|_1$.

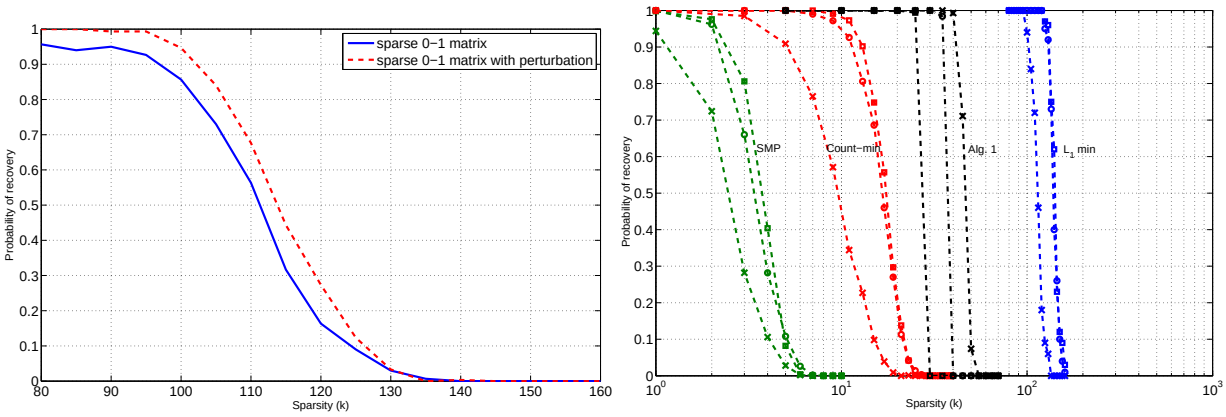
Algorithm 1 —Reverse Expansion Recovery (REVEX)

Input: Observation vector $\mathbf{y}$, measurement matrix $\mathbf{A} = [a_{i,j}]_{1\leq i\leq m, 1\leq j\leq n}$, its corresponding generalized bipartite graph $\mathcal{G} = (X, Y, E, W)$, parameter θ determining whether the observations are noiseless ($\theta = 0$) or noisy ($\theta = 1$), and parameter $p \geq 1$.
Output: Estimate $\hat{\mathbf{x}}$ of the sparse vector $\mathbf{x}$.
Set T_1 as the set of the smallest $m - kd$ of $\mathbf{y}$ in magnitude, and denote them by $\mathbf{y}_1$. Also set T_2 as the complement of T_1. Without loss of generality, assume that $\mathbf{y} = (\mathbf{y}_1\ \mathbf{y}_2)^t$.
Set $S_1 := \Gamma(T_1)$ and $S_2 := S_1^c$.
Set $\tilde{\mathbf{A}}_2 := [a_{i,j}]_{i\in S_2, j\in T_2}$.
Set $\hat{\mathbf{x}}_{S_1} := 0$.
Set $\tilde{\mathbf{A}}_2^\dagger := \left(\tilde{\mathbf{A}}^t\tilde{\mathbf{A}}\right)^{-1}\tilde{\mathbf{A}}^t$.
if $\theta = 0$ **then**
    Set $\hat{\mathbf{x}}_{S_2} := \tilde{\mathbf{A}}_2^\dagger \mathbf{y}_2$.
else
    Set $\hat{\mathbf{x}}_{S_2} := \operatorname{argmin}_{\mathbf{z}\in\mathbb{R}^{|S_2|\times 1}} \|\mathbf{A}_2\mathbf{z} - \mathbf{y}_2\|_p$.
end if

2.9 Experimental Evaluation

We generated a random $m \times n$ matrix $\mathbf{A}$ with $n = 2m = 500$, and $d = 3$ 1s in each column. We then multiplied random sparse vectors with different sparsity levels by $\mathbf{A}$, and tried recovering them via the linear program (2.4.2). Next, we added a random set of perturbations to the nonzero entries of $\mathbf{A}$ while keeping it nonnegative and constant column sum (one way to do that is by adding a uniformly random number in $[-1, 1]$ to each 1 of $\mathbf{A}$, and then normalizing each column) and applied the same sparse vectors to compare the recovery percentages in the two cases. This process was repeated for a few generations of $\mathbf{A}$ and the best of the improvements we obtained is illustrated in Figure 2.6a.

In Figure 2.6b we have plotted the recovery percentage of Algorithm 1 for a random $0 - 1$ sparse matrix $\mathbf{A}$ of size 250×500 with d ones in each column. We have compared the performance with the ℓ_1 minimization method, the count-min algorithm of [CM04] and the sparse Matching Pursuit (SMP) method of [BIR08], all specific to positive signals. Note that for the count-min algorithm, the measurement



(a) ℓ_1 minimization for sparse 0-1 matrices with and without perturbations.

(b) Comparison of algorithms.

Figure 2.6: (a) Probability of successful recovery of ℓ_1 minimization for a random $0-1$ sparse matrix of size 250×500 with $d = 3$ ones in each column, and the same probability when the matrix is randomly perturbed in the nonzero entries. (b) Comparison of ℓ_1 minimization nonnegative recovery, Algorithm 1, count-min algorithm of [CM04] and SMP algorithm of [BIR08] for sparse $0-1$ measurement matrices with d ones in each column. $\times$: $d = 3$, $\circ$: $d = 6$, $\square$: $d = 9$. Blue: ℓ_1 minimization, Black: Algorithm 1, Red: Count-min, Green: SMP

matrix must have other properties in addition to the constant number of 1s in each column. Please see [BIR08] for more details. Although the deterministic theoretical bounds for the proposed algorithm and the ℓ_1 minimization are the same, as observed in Figure 2.6b, in practice ℓ_1 minimization is more effective for less sparse signals. However Algorithm 1 is considerably faster than linear programming and easier to implement.

In general, the complexity of Algorithm 1 is $O(nk^2)$, which, when k is proportional to n, is similar to linear programming's $O(n^3)$. However the constants are much smaller, which is of practical advantage. Furthermore, taking advantage of fast matrix inversion algorithms for very sparse matrices, Algorithm 1 can be performed in dramatically fewer operations. Figure 2.7 shows the signal-to-error ratio as a function of signal-to-noise Ratio when Algorithm 1 with $p = 2$ has been used to recover noisy observations. Assuming that the output of the algorithm is $\hat{\mathbf{x}}$, signal-to-noise

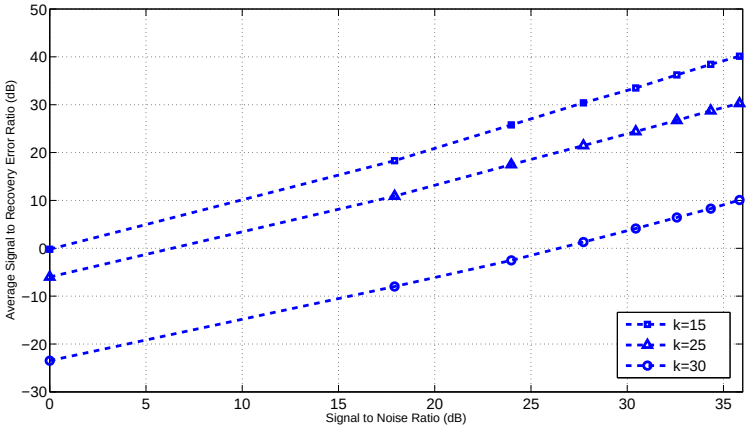


Figure 2.7: Simulation results for Algorithm 1, noisy case; signal-to-error ratio vs. signal-to-noise ratio.

ratio (SNR) and signal-to-error ratio (SER) functions are defined as

$$SNR = 10 \log \frac{\|\mathbf{Ax}\|_2^2}{\|\mathbf{v}\|_2^2}, \ SER = 10 \log \frac{\|\mathbf{x}\|_2^2}{\|\mathbf{x} - \hat{\mathbf{x}}\|_2^2}.$$

Measurement matrices are the same as before.

2.10 Proof of Theorems

Proof of Theorem 2.7.1.

Assuming that we generate a random matrix $\mathbf{A}$ by placing d ones in each column uniformly at random and the rest of the entries zero, it suffices to show that the probability that $\mathbf{A}$ has the desired expansion property is positive. For $1 \leq i_1 < i_2 < ... < i_r \leq m$, we denote by $E_{i_1,i_2,...,i_r}$ the event that the columns of $\mathbf{A}$ corresponding to the numbers i_1,i_2,...,i_r have at least $m-r+1$ entire 0 rows (rows that do not have a single nonzero element in the columns $\mathbf{A}_{i_1}$,$\mathbf{A}_{i_2}$,...,$\mathbf{A}_{i_k}$). In other words $E_{i_1,i_2,...,i_r}$ is the event that the set of nodes $\{i_1, i_2, ..., i_r\}$ in X contracts in Y.

$$\begin{aligned} \mathbb{P}\left(\mathbf{A} \text{ is a } (r_0, \frac{d-1}{d}) \text{ expander}\right) &= 1 - \mathbb{P}\left(\bigcup_{d \leq r \leq r_0, 1 \leq i_1 < i_2 < ... < i_r} E_{i_1,i_2,...,i_r}\right) \\ &\geq 1 - \sum_{r=d}^{r_0} \binom{n}{r} \mathbb{P}\left(E_{1,2,...,r}\right). \end{aligned}$$

We use the following combinatorial analysis to bound $\mathbb{P}[E_{1,2,\ldots,r}]$. The total number of 0 — 1 matrices of size $m \times r$ that have exactly d 1s in every column is $\binom{m}{d}^r$. In addition, if a matrix is to have at least $m - r + 1$ entirely zero columns, all of its nonzero entries must be included in a submatrix of size $r \times r$. There are $\binom{m}{r}$ choices for such a square submatrix, and for each selection, the number of matrices that can be constructed is by the same token as before equal to $\binom{r}{d}^r$ (It should be clear that we are multiple-counting many cases). This yields the following:

$$\mathbb{P}\left(E_{1,2,\ldots,r}\right) \leq \frac{\binom{m}{r}\binom{r}{d}^r}{\binom{m}{d}^r}.$$

Hence

$$\mathbb{P}\left(A \text{ is a } (r_0, \frac{d-1}{d}) \text{ expander}\right) \geq 1 - \sum_{r=d}^{r_0} \binom{n}{r} \frac{\binom{m}{r}\binom{r}{d}^r}{\binom{m}{d}^r}. \tag{2.10.1}$$

We show that for certain regimes of β, μ and d, the right-hand side of (2.10.1) becomes arbitrarily close to 1. To see this, we prove the following two lemmas.

Lemma 2.10.1. *If* $0 < \alpha < e^{\frac{d}{2-d}} \beta^{\frac{1-d}{2-d}}$ *and* $d \geq 3$ *then* $\sum_{r=d}^{\alpha n} \binom{n}{r} \frac{\binom{m}{r}\binom{r}{d}^r}{\binom{m}{d}^r} = \mathcal{O}(n^{1-d(d-2)})$, *as* $n \to \infty$.

Proof. We can write:

$$\sum_{r=d}^{\alpha n} \binom{n}{r} \frac{\binom{m}{r}\binom{r}{d}^r}{\binom{m}{d}^r} \leq \sum_{r=d}^{\alpha n} \binom{n}{r}\binom{m}{r} (\frac{r}{m})^{rd} \leq \sum_{r=d}^{\alpha n} (\frac{ne}{r})^r (\frac{me}{r})^r (\frac{r}{m})^{rd} = \sum_{r=d}^{\alpha n} (\frac{cr}{n})^{r(d-2)}, \tag{2.10.2}$$

where $c = e^{\frac{2}{d-2}} \beta^{\frac{1-d}{d-2}}$, and we have used the bounds $\frac{\binom{r}{d}}{\binom{m}{d}} \leq (\frac{r}{m})^d$ for $r < m$, and $\binom{n}{k} \leq (\frac{ne}{k})^k$. It is easy to show that when $\alpha < \frac{1}{ec}$, $(\frac{cr}{n})^r$ is decreasing in r, and thus replacing all the terms in (2.10.2) by the first term will only increase the sum. The whole term is thus smaller than $\alpha n (\frac{cd}{n})^{d(d-2)} = \lambda n^{1-d(d-2)}$ for some positive constant λ. ∎

Lemma 2.10.2. *For $m = \beta n$ and $r_0 = \mu n$, if $d > \frac{H(\mu)+\beta H(\frac{\mu}{\beta})}{\mu \log_2(\frac{\beta}{\mu})}$, then for any $0 < \alpha < \mu$ the sum $\sum_{r=\alpha n+1}^{\mu n} \binom{n}{r} \frac{\binom{m}{r}\binom{r}{d}^r}{\binom{m}{d}^r}$ decays exponentially as $n \longrightarrow \infty$.*

Proof. Using the standard bounds of (10.1.1) on binomial coefficients and the fact that $\frac{\binom{r}{d}}{\binom{m}{d}} \leq (\frac{r}{m})^d$ for $r < m$, we can write:

$$\sum_{r=\alpha n+1}^{\mu n} \binom{n}{r} \frac{\binom{m}{r}\binom{r}{d}^r}{\binom{m}{d}^r} \leq n^2 \sum_{r=\alpha n+1}^{\mu n} 2^{nH(\frac{r}{n})+mH(\frac{r}{m})+rd\log_2 \frac{r}{m}+\log_2\{(n+1)(m+1)\}}, \qquad (2.10.3)$$

where $H(x) = x\log(\frac{1}{x}) + (1-x)\log(\frac{1}{1-x})$ is the entropy function. Assuming that $\mu \leq \frac{\beta}{2}$, the largest term on the right-hand side of (2.10.3) is the one corresponding to $r = \mu n$ (since $H(x)$ and $x\log_2 x$ are both increasing for $0 \leq x \leq \frac{1}{2}$), and therefore we can write

$$\sum_{r=\alpha n+1}^{\mu n} \binom{n}{r} \frac{\binom{m}{r}\binom{r}{d}^r}{\binom{m}{d}^r} \leq 4n^5 2^{n(H(\mu)+\beta H(\frac{\mu}{\beta})+\mu d\log(\frac{\mu}{\beta}))}. \qquad (2.10.4)$$

Note that we have used the fact that $(n+1)(m+1) < 4n^2$. The right hand side of (2.10.4) vanishes as $n \to \infty$, if $d > \frac{H(\mu)+\beta H(\frac{\mu}{\beta})}{\mu \log_2(\frac{\beta}{\mu})}$. ∎

Proof of Theorem 2.7.3.

For a random set S that has a linear-size cardinality (i.e., $\frac{|S|}{n}$ is not asymptotically zero), and $\frac{|S|}{m}$ small enough, $\mathbb{E}|\Gamma(S)| = (1 - e^{-d\frac{|S|}{m}})\beta n =: \gamma_1 n$ when $n \to \infty$. We first need to ensure that $|\Gamma(S)|$ is concentrated. We can show that:

$$\mathbb{P}\left(|\Gamma(S)| \leq \mathbb{E}|\Gamma(S)| + \epsilon_1\right) > 1 - \frac{1}{n}.$$

This concentration bound can be obtained by the standard Martingale concentration arguments if $|S| \geq c_1 n$, for $\epsilon_1 = c_2\sqrt{n\log n}$, see [SS96, BM01]. Therefore we define the event $E_1 = \{|\Gamma(S)| \leq \gamma_1 n + \epsilon_1\}$. Consider the random graph created from placing d nonzero entries (with repetition) in every column of $\tilde{\mathbf{A}}$. From the set S, form $\Gamma(S)$, the corresponding S_2, and finally the bipartite graph $B_S = (S \cup S_2, \Gamma(S \cup S_2))$. Using the given combinatorial condition in Lemma 2.7.4, we can recover a signal supported

on S if every subset $S_i \subset S \cup S_2$ of size $|S_i| \leq |\Gamma(S)| + 1$ has sufficient expansion: $|\Gamma(S_i)| \geq |S_i|$ (note that subsequently we drop the $+1$ term since it is negligible for large n). First we condition on the concentration of $|\Gamma(S)|$:

$$\begin{aligned}\mathbb{P}\left(S \text{ not recoverable}\right) &= \mathbb{P}\left(S \text{ not recoverable}|E_1\right)\mathbb{P}[E_1] + \mathbb{P}\left(S \text{ not recoverable}|E_1^c\right)\mathbb{P}\left(E_1^c\right) \\ &\leq \mathbb{P}\left(S \text{ not recoverable}|E_1\right)\left(1 - \frac{1}{n}\right) + \frac{1}{n}, \end{aligned} \tag{2.10.5}$$

Therefore it suffices to bound the probability conditioned on $|\Gamma(S)|$ concentrated. We are going to do a union bound over all possible selections R_1 of r_1 nodes in S and R_2 of r_2 nodes in S_2 so that $r_1 + r_2 \leq |\Gamma(S)|$. Since we are conditioning on E_1, it suffices to have $r_1 + r_2 \leq \gamma_1 n + \epsilon_1$. The second problem is that the set S_2 is random and dependent on $\Gamma(S)$. We are going to avoid this conditioning by allowing the choice of R_2 to range over all the $n - k$ nodes in S^c.

$$\mathbb{P}\left(S \text{ not recoverable}|E_1\right) \leq \sum_{r_1+r_2 \leq \gamma_1 n + \epsilon_1} \binom{k}{r_1}\binom{n-k}{r_2}\mathbb{P}\left(R_1 \cup R_2 \text{ contracts}|E_1\right). \tag{2.10.6}$$

Now the problem is that conditioning on E_1 implies that the set R_1 does not expand too much, so it is actually increasing the probability of the bad contraction event. We can however easily show that this increase is at most a factor of 2:

$$\mathbb{P}\left(R_1 \cup R_2 \text{ contracts}|E_1\right) = \frac{\mathbb{P}\left(R_1 \cup R_2 \text{ contracts} \cap E_1\right)}{\mathbb{P}(E_1)} \leq \frac{\mathbb{P}\left(R_1 \cup R_2 \text{ contracts}\right)}{\mathbb{P}(E_1)}. \tag{2.10.7}$$

Now since $\mathbb{P}(E_1) \geq 1 - 1/n$, for sufficiently large n, $1/\mathbb{P}(E_1) \leq 2$, so

$$\mathbb{P}\left(R_1 \cup R_2 \text{ contracts}|E_1\right) \leq 2\mathbb{P}\left(R_1 \cup R_2 \text{ contracts}\right). \tag{2.10.8}$$

The probability that the set $R_1 \cup R_2$ contracts can be further bounded by assuming $|\Gamma(R_1 \cup R_2)| = r_1 + r_2$ (any smaller neighborhood will have smaller probability) so

$$\mathbb{P}\left(R_1 \cup R_2 \text{ contracts}\right) \leq \binom{m}{r_1 + r_2}\left(\frac{r_1 + r_2}{m}\right)^{d(r_1+r_2)}.$$

Putting everything together we obtain the bound

$$\mathbb{P}\left(S \text{ not recoverable}|E_1\right) \leq 2 \sum_{r_1+r_2 \leq \gamma_1 n+\epsilon_1} \binom{k}{r_1}\binom{n-k}{r_2}\binom{m}{r_1+r_2}\left(\frac{r_1+r_2}{m}\right)^{d(r_1+r_2)}. \tag{2.10.9}$$

We move everything to the exponent, assume $\epsilon_1/n \to 0$, and use standard binomial approximations to obtain

$$\mathbb{P}\left(S \text{ not recoverable}|E_1\right) \leq 2n^3 \sum_{r_1+r_2 \leq \gamma_1 n} 2^{kH(\frac{r_1}{k})+(n-k)H(\frac{r_2}{n-k})+mH(\frac{r_1+r_2}{m})+d(r_1+r_2)\log(\frac{r_1+r_2}{m})}. \tag{2.10.10}$$

Recall that the recoverable fraction is $k = \alpha n$, $m = \beta n$, and denote $\rho_1 = r_1/n$, $\rho_2 = r_2/n$. Define the function

$$F(\rho_1,\rho_2) := \alpha H(\frac{\rho_1}{\alpha}) + (1-\alpha)H(\frac{\rho_2}{1-\alpha}) + \beta H(\frac{\rho_1+\rho_2}{\beta}) + d(\rho_1+\rho_2)\log(\frac{\rho_1+\rho_2}{\beta}), \tag{2.10.11}$$

and observe that the bound on the probability of failure (2.10.10) becomes

$$\mathbb{P}\left(S \text{ not recoverable}|E_1\right) \leq 2n^3 \sum_{r_1+r_2 \leq \gamma_1 n} 2^{nF(\rho_1,\rho_2)}.$$

Therefore for fixed β and γ_1, we are trying to find the largest α^* that makes $F(\rho_1,\rho_2)$ negative for every ρ_1, ρ_2 for which $\rho_1+\rho_2 \leq \gamma_1$. For this α^*, we can recover sparse signals with sparsity no bigger than $\alpha^* n$ with polynomially high probability, conditioned on the fact that the sublinear sets do not contract (which has already been established).

Proof of Validity of Algorithm 1.

Algorithm 1 identifies a big zero portion of the output and eliminates two large sets of nodes from X and Y (Figure 2.8). Having done this, a smaller system of linear equation remains, which turns out to be an over-determined system and can be uniquely solved using matrix inversions. This procedure is therefore nothing but a block diagonalization (after rearranging the rows and columns) of $\tilde{\mathbf{A}}$ into a lower

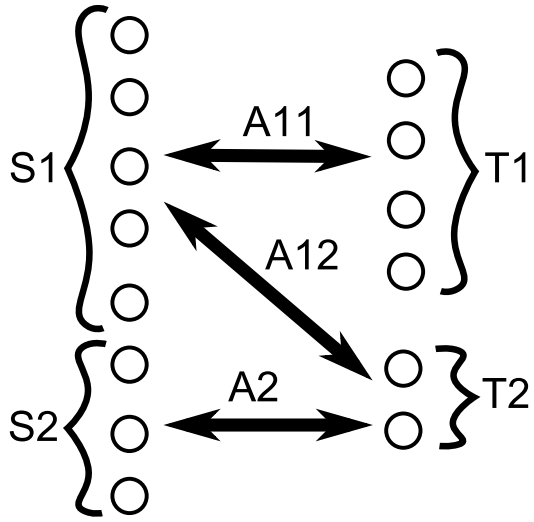


Figure 2.8: Decomposition of nodes and edges by Algorithm 1

triangular matrix:

$$\tilde{\mathbf{A}} = \begin{pmatrix} \tilde{\mathbf{A}}_{11} & 0 \\ \tilde{\mathbf{A}}_{12} & \tilde{\mathbf{A}}_2 \end{pmatrix} \tag{2.10.12}$$

where $\tilde{\mathbf{A}}_2$ is a square or tall full-rank matrix. The following theorem certifies that Algorithm 1 is indeed valid and it recovers any k-sparse vector without error.

Theorem 2.10.3. *(Validity of Algorithm 1) If* $\mathbf{x}$ *is a* k*-sparse nonnegative vector and* $\tilde{\mathbf{A}}$ *is a perturbed* $(kd+1, 1-\frac{1}{d})$ *expander with* $\mathcal{H}(\tilde{\mathbf{A}}) \geq kd+1$*, then* $\hat{\mathbf{x}} = \mathbf{x}$*.*

Proof. First of all, note that $\mathbf{y}$ is kd-sparse, since every column of $\tilde{\mathbf{A}}$ has exactly d nonzero entries. We also show that $|S_2| \leq |T_2|$ and therefore $\tilde{\mathbf{A}}_2$ is a full rank matrix. Suppose $|S_2| > |T_2|$. We also know that $|T_2| \leq kd$. Select an arbitrary subset $S_2{}' \subseteq S_2$ of size $|T_2|+1$. Because of the fact that $|S_2| \leq kd+1$ and the expansion property: $|\Gamma(S_2{}')| \geq |S_2{}'| > |T_2|$. But $\Gamma(S_2{}')$ is a subset of T_2 and this is a contradiction. Diagonalization of (2.10.12) and the fact that $\tilde{\mathbf{A}}_2$ is a tall matrix and $\mathcal{H}(\tilde{\mathbf{A}}) \geq kd+1$ together imply that $\tilde{\mathbf{A}}_2$ has full column rank.

We now show that $\hat{\mathbf{x}} = \mathbf{x}$. If any entry in $\mathbf{x}_{S_1}$ is greater than zero, then there is at least one entry in $\mathbf{y}$ which is indexed in T_1 and is nonzero, since $S_1 = \Gamma(T_1)$. This is in contradiction with the choice of T_1. Therefore $\mathbf{x}_{S_1} = 0 = \hat{\mathbf{x}}_{S_1}$. Also since $\tilde{\mathbf{A}}_2 \mathbf{x}_{S_2} = \mathbf{y}_2$ and $\tilde{\mathbf{A}}_2 \hat{\mathbf{x}}_{S_2} = \mathbf{y}_2$ and $\tilde{\mathbf{A}}_2$ is full-rank, we conclude that $\mathbf{x}_{S_2} = \hat{\mathbf{x}}_{S_2}$.

■

Proof of Robustness of Algorithm 1.

We first state the following lemma from [BGI+08]:

Lemma 2.10.4. *Consequence of Lemma 9 of [BGI+08]: If* $\mathbf{A}$ *is the adjacency matrix of a (k,*ϵ*) expander with* $\epsilon < 0.5$ *and* $\mathbf{u}$ *is a k-sparse vector, then* $d(1-2\epsilon)\|\mathbf{u}\|_1 \leq \|\mathbf{Au}\| \leq d\|\mathbf{u}\|_1$.

If the original vector $\mathbf{x}$ is k-sparse, and each column of $\mathbf{A}$ has d 1s, then $\mathbf{Ax}$ has at most kd 1s, i.e. $\mathbf{Ax}$ is kd-sparse. By rearranging the rows and columns of $\mathbf{A}$, we may assume $\mathbf{x} = (\mathbf{x}_1\ \mathbf{x}_2)^T$, $\mathbf{y} = (\mathbf{y}_1\ \mathbf{y}_2)^T$, $\mathbf{v} = (\mathbf{v}_1\ \mathbf{v}_2)^T$, and $\mathbf{A} = \begin{pmatrix} \mathbf{A}_{11} & 0 \\ \mathbf{A}_{12} & \mathbf{A}_2 \end{pmatrix}$, where $\mathbf{y}_1$ and $\mathbf{y}_2$ are those obtained by the algorithm, $\mathbf{x}_1 = \mathbf{x}_{S_1}$, and $\mathbf{x}_2 = \mathbf{x}_{S_2}$. Also let $\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}}$ be the reconstruction error vector. By (2.8.1) we then have

$$\mathbf{y}_1 = \mathbf{A}_{11}\mathbf{x}_1 + \mathbf{v}_1,\ \mathbf{y}_2 = \mathbf{A}_{12}\mathbf{x}_1 + \mathbf{A}_2\mathbf{x}_2 + \mathbf{v}_2,\ \mathbf{e} = (\mathbf{x}_1\ \mathbf{x}_2 - \hat{\mathbf{x}}_2)^T. \tag{2.10.13}$$

Hence we have:

$$\|\mathbf{x}_1\|_1 \leq \|\mathbf{A}_{11}\mathbf{x}_1\|_1 = \|\mathbf{y}_1 - \mathbf{v}_1\|_1 \leq \|\mathbf{y}_1\|_1 + \|\mathbf{v}_1\|_1 \leq 2\|\mathbf{v}\|_1. \tag{2.10.14}$$

The first inequality holds as a result of nonnegativity of $\mathbf{x}_1$ and $\mathbf{A}_{11}$, and the fact that every column of $\mathbf{A}_{11}$ has at least one 1. The last inequality holds for the following reason. We know that $\mathbf{Ax}$ is kd-sparse. Let T_0 be a set of $m - kd$ zeros of $\mathbf{Ax}$. We then have

$$\|\mathbf{y}_1\|_1 \leq \|\mathbf{y}_{T_0}\|_1 = \|\mathbf{v}_{T_0}\|_1 \leq \|\mathbf{v}\|_1. \tag{2.10.15}$$

Let us assume $\mathbf{y}_2 = \mathbf{A}_2\hat{\mathbf{x}}_2 + \delta_2$. From the way $\hat{\mathbf{x}}_2$ is driven in step 4 of the algorithm, it follows that:

$$\|\delta_2\|_1 \leq \|\mathbf{A}_{12}\mathbf{x}_1 + \mathbf{v}_2\|_1. \tag{2.10.16}$$

And thus

$$\|\mathbf{A}_2(\mathbf{x}_2 - \hat{\mathbf{x}}_2)\|_1 = \|\delta_2 - \mathbf{A}_{12}\mathbf{x}_1 - \mathbf{v}_2\|_1 \leq 2\|\mathbf{A}_{12}\mathbf{x}_1 + \mathbf{v}_2\|_1 \leq 2d\|\mathbf{x}_1\|_1 + 2\|\mathbf{v}_2\|_1. \tag{2.10.17}$$

We may note that $|S_2| < 2k$. Otherwise, since $\mathbf{A}$ corresponds to a $(2k, \epsilon)$ expander, we can choose an arbitrary subset $S_2{}' \subset S_2$ of size $2k$, and conclude that

$$|\Gamma(S_2)| \geq |\Gamma(S_2')| \geq 2k(1-\epsilon)d > kd. \tag{2.10.18}$$

However, from the construction of S_2, we know that $\Gamma(S_2) = T_2$ and $|T_2| = kd$. Therefore the vector $\mathbf{u} = (0\ \mathbf{x}_2 - \hat{\mathbf{x}}_2)^T$ is $2k$-sparse (because the size of $\mathbf{x}_2$ is at most $2k$), and we can apply the RIP-1 condition of Lemma 2.10.4 to it, which yields

$$c_1\|\mathbf{x}_2 - \hat{\mathbf{x}}_2\|_1 \leq 2d\|\mathbf{x}_1\|_1 + 2\|\mathbf{v}_2\|_1, \tag{2.10.19}$$

where $c_1 = (1-2\epsilon)d$. Equations (2.10.14) and (2.10.19) result in (assuming $d \geq 2$):

$$\|\mathbf{e}\|_1 \leq (2 + \frac{4d}{c_1})\|\mathbf{v}\|_1 + \frac{2}{c_1}\|\mathbf{v}_2\|_1 \leq \frac{7-4\epsilon}{1-2\epsilon}\|\mathbf{v}\|_1. \tag{2.10.20}$$

2.11 Conclusion

We considered the recovery of a nonnegative sparse vector using a sparse measurement matrix in the compressed sensing framework. We used the perturbed adjacency matrix of a bipartite expander graph to construct the sparse measurement matrix and proposed a novel fast algorithm. We computed recovery thresholds and showed that for measurement matrices with nonnegative entries and constant column sum the constraint set $\{\mathbf{x}|\mathbf{x} \geq 0, \mathbf{A}\mathbf{x} = \mathbf{y}\}$ is a singleton set, whenever ℓ_1 optimization is successful (which also means that any other nontrivial optimization scheme which can examine the feasible set would be successful). Finally, determining whether the matrices constructed satisfy an RIP-2 property, and constructing 0 — 1 matrices that have complete rank proportional to n are open problems that may be worthy of further scrutiny.

Chapter 3

Bipartite Graphs with Large Girth

In the previous chapter, we introduced a class of sparse matrices that are suitable for compressed sensing with basis pursuit recovery algorithms. The fundamental property of the proposed matrices was expansion, through which certain balancedness conditions are guaranteed for the corresponding null space, and thus certifying the success of nonnegative constrained ℓ_1 minimization. There are, however, several drawbacks with that proposal. First of all, the theoretical guarantees of the previous chapter were all derived under the assumption that the unknown signal is nonnegative. Secondly, although expander graphs can be generated deterministically, verifying whether a given sparse matrix corresponds to an expander graph with specific parameters can be very difficult. In particular, in the linear regime where m and k are both proportional to n, there exists (to our best knowledge) no polynomial time algorithm that can verify whether or not a given bipartite graph with n left nodes and m right nodes is a (k, ϵ)-expander, for a constant ϵ. Finally, as discussed, minimal expanders need perturbation for guaranteed success. This could potentially make their performance susceptible to noise and numerical disparities such as quantization and limited precision in practical situations.

In this chapter, we propose another criteria for sparse matrices, under which ℓ_1 minimization can work properly. We show that if a low-density binary matrix corresponds to a bipartite (Tanner) graph with large girth, then certain guarantees exist for the success of basis pursuit. The girth of a graph is defined as the length of

the shortest cycle. The techniques used to derive this conclusion are based on results for error-correcting codes. As we will explain, there is a connection between good linear block codes for error correcting and good measurement matrices for compressed sensing.

Unlike the previous chapter, the theoretical guarantees that result from using large girth codes hold for indefinite sign signals (not just nonnegative signals, unlike minimal expanders), and can also be generalized to the case of noisy observations or approximately sparse signals, thus providing robustness for basis pursuit under these conditions. Another critical importance of the girth criteria is that, unlike expansion, the exact numerical value for the girth of a graph can be computed within polynomial time.

$\mathbf{A}$	measurement matrix
n	signal size
m	number of measurements
k	sparsity of the signal
$\mathcal{N}(\mathbf{A})$	null space of $\mathbf{A}$
$\mathcal{C}$	binary linear error correcting code
$\mathbf{H_{CC}}$	parity check matrix of $\mathcal{C}$
C_R	robustness constant of ℓ_1 minimization

3.1 Introduction

As discussed previously, in compressed sensing, one tries to estimate an unknown vector $\mathbf{x} \in \mathbb{R}^n$ from the observation of m linear measurements in the form:

$$\mathbf{y} = \mathbf{A} \cdot \mathbf{x},$$

where $\mathbf{A}$ is a real-valued measurement matrix of size $m \times n$. When $m < n$ this is an underdetermined system of linear equations, and one fundamental compressed sensing problem involves recovering $\mathbf{x}$ assuming that it is also k-sparse, i.e., it has k or fewer nonzero entries. The sparse approximation problem goes beyond exactly sparse vectors and requires the recovery of a k-sparse vector $\hat{\mathbf{x}}$ that is close to $\mathbf{x}$, even if $\mathbf{x}$ is not

exactly k-sparse itself. Recent breakthrough results [CT05, Don06a, DT05a] showed that it is possible to construct measurement matrices with $m = O(k \log(n/k))$ rows that recover k-sparse signals exactly in polynomial time. This scaling is also optimal, as discussed in [GI10, BIPW10]. These results rely on randomized matrix (often dense) constructions and establish that the optimal number of measurements will be sufficient with high probability over the choice of the matrix and/or the signal, and rely on geometric or linear-algebraic conditions such as RIP or the null space property to be satisfied by the measurement matrix $\mathbf{A}$ [CT05, DH01, SXH08]. In addition, sparse measurement structures have also been constructed with guarantees for the Basis-Pursuit-types of algorithms, mostly based on expander graphs. As described in Chapter 2, prior to the our results for the use of minimal expanders, the existing sparse constructions for compressed sensing were based on high-quality expander. Unfortunately, there is no efficient way to *deterministically* construct matrices that satisfy the RIP, null space, or high-quality expansion properties to the same order of optimality as random constructions. In addition, given an arbitrary matrix, there are no known ways of verifying these conditions and obtaining the exact constants (e.g., RIP constant or specifications of the null space property). In fact, it is even conjectured that the problem of certifying RIP for an arbitrary matrix is non-polynomial [KZ].

There are several explicit constructions of measurement matrices (e.g., [BCJ10, DeV07]) which, however, require a slightly sub-optimal number of measurements (m growing super-linearly as a function of n for $k = p \cdot n$). Our focus here is in the *linear sparsity* regime where k is a fraction of n and optimal number of measurements will also be a fraction of n. This is a very important regime, as the performance of the Basis Pursuit algorithm can be completely characterized in this regime for asymptotically large dimensions. These are the so-called recovery thresholds for ℓ_1 minimization which were briefly discussed in Chapter 2, and are known to be tight for random structures. We will get back to this issue in details in Part II of this

dissertation. The problem we are addressing now is how much we can tighten the gap between the thresholds of Basis Pursuit for random structures and for deterministic structures. The explicit construction of measurement matrices with an optimal number of rows is a well-known open problem in compressed sensing theory (see, e.g., [GI10] and references therein).[1] A closely related issue is that of checking or certifying in polynomial time that a given candidate matrix has good recovery guarantees.

Minimal expanders introduced in the previous chapter are special-case near-optimal solutions to the above open problem, for nonnegative sparse vectors. In this chapter, we introduce a new class of sparse and binary measurements which result in very tight thresholds for Basis Pursuit, for indefinite sign vectors. These are the class of matrices that represent bipartite graphs with large girths (*i.e.* minimum cycle length). The girth condition can efficiently be analyzed for a sparse matrix. Our techniques are coding theoretic and rely on a recent connection of compressed sensing to LP relaxations for channel decoding.

3.2 Contributions

We prove the following important results. Consider a sparse matrix $\mathbf{A}$ in $\{0,1\}^{m\times n}$ that has d_c ones per row and d_v ones per column. If the bipartite graph corresponding to $\mathbf{A}$ has $\Omega(\log n)$ girth, then for $k = p \cdot n$ and an optimal number of measurements $m = c_2 \cdot n$, we show that $\mathbf{A}$ offers ℓ_1/ℓ_1 sparse approximation under the Basis Pursuit decoding algorithm.[2] Our technical requirement of girth, unlike expansion or RIP, is easy to check, and several deterministic constructions of matrices with $m = c \cdot n$ and $\Omega(\log n)$ exist, starting with the early construction in Gallager's thesis [Gal63], and

[1] We note that in the online draft of [DSV], p. 11, there is an incorrect comparison to prior work that suggests that the Capalbo et al. [CRVW02] explicit construction of high-quality expanders would give an optimal number of measurements. In fact, in the linear sparsity regime $k = c \cdot n$, the construction of [CRVW02] would require $m = c_2 n 2^{O(\log\log(n)^3)}$ measurements [BGI+08]. Therefore, to the best of our knowledge, [DSV] has the only known construction that recovers k sparse signals with a deterministic matrix and an optimal number of measurements.

[2] The ℓ_1/ℓ_1 approximation shall be defined soon.

the progressive edge-growth Tanner graphs of [HEA05].

Our result is a weak bound, also known as a "for-every signal" guarantee [GI10]. This means that we have a fixed deterministic matrix and show the ℓ_1/ℓ_1 sparse approximation guarantee with high probability over the support of the signal. To the best of our knowledge, this is the first deterministic construction of matrices with an optimal number of measurements and the strong bound ("for-all signals") equivalent remains open.

Our techniques are coding-theoretic and rely on recent developments that connect the channel decoding LP relaxation by Feldman et al. [FWK05] to compressed sensing [DSV]. We rely on a primal-based density evolution technique initiated by Koetter and Vontobel [KV06] and analytically strengthened in the breakthrough paper of Arora et al. [ADS09] that established the best-known finite-length threshold results for LDPC codes under LP decoding.

To show our ℓ_1/ℓ_1 sparse approximation result, we first need to extend [DSV] for non-sparse signals. Specifically, the first step in our analysis (Theorem 3.4.6) is to show that ℓ_1/ℓ_1 sparse approximation corresponds to a channel decoding problem for a *perturbed symmetric channel*. The second component is to extend the Arora et al. argument for this perturbed symmetric channel, an analysis achieved in Theorem 7.5.6. This is performed by showing how a similar tree recursion allows us to establish robustness of the fundamental cone (FCP), i.e. every pseudocodeword in the fundamental cone [KV06] has a nonnegative pseudoweight even if the flipped bit likelihoods are multiplied by a factor larger than one. The FCP condition shows that the matrix $\mathbf{A}$, taken as an LDPC code, can tolerate a constant fraction of errors for this perturbed symmetric channel under LP decoding.

We note that even though our analysis involves a rigorous density evolution argument, our decoder is always the Basis Pursuit linear relaxation, which is substantially different from the related work on message-passing algorithms for compressed sensing [DMM09, ZP].

3.3 Preliminaries

In this section we provide a brief background for the major topics we will discuss. We begin by introducing the noiseless compressed sensing problem and the basis pursuit. Then, we mention the channel coding problem and its relaxation. Since the focus of this chapter is compressed sensing, the description of the channel coding problem is not very involved. In Chapter 7, error-correcting codes and channel decoding via linear programming will be explained in more detail.

3.3.1 Compressed Sensing Preliminaries

The simplest noiseless compressed sensing (CS) problem for exactly sparse signals consists of recovering the sparsest real vector $\mathbf{x}$ of a given length n, from a set of m real-valued measurements $\mathbf{y}$, given by $\mathbf{A} \cdot \mathbf{x} = \mathbf{y}$; namely

$$\begin{aligned} \textbf{CS-OPT:} \quad & \text{minimize} \quad && \|\mathbf{x}\|_0 \\ & \text{subject to} \quad && \mathbf{A} \cdot \mathbf{x} = \mathbf{y}. \end{aligned}$$

Since ℓ_0 minimization is NP-hard, one can relax **CS-OPT** by replacing the ℓ_0 norm with ℓ_1, specifically

$$\begin{aligned} \textbf{CS-LPD:} \quad & \text{minimize} \quad && \|\mathbf{x}\|_1 \\ & \text{subject to} \quad && \mathbf{A} \cdot \mathbf{x} = \mathbf{y}. \end{aligned}$$

This LP relaxation is also known as Basis Pursuit. A central question in compressed sensing is under what conditions the solution given by **CS-LPD** equals (or is very close to, especially in the case of approximately sparse signals) the solution given by **CS-OPT**, i.e., the LP relaxation is tight. There has been a substantial amount of work in this area, see, e.g., [CT05, Don06a, DT05a, DH01, DT05a, GI10].

One sufficient way to certify that a given measurement matrix is "good" is through the well-known restricted isometry property (RIP), which guarantees that the LP

relaxation will be tight for all k-sparse vectors $\mathbf{e}$, and further that the recovery will be robust to approximate sparsity [CT05, Don06a]. However, RIP condition is not a complete characterization of the LP relaxation of "good" measurement matrices (see, e.g., [BCT10]). An alternative is to look at the null space characterization (see, e.g., [XH08, SXH08]) instead, which gives a necessary and sufficient condition for a matrix to be "good". In the previous chapter, the null space conditions were defined for the case where the nonzero entries of the vector $\mathbf{x}$ are known to be positive. The general form of the null space property that helps analyze Basis Pursuit is the following:

Definition 5. *Let $\mathcal{S} \subset \{1, \ldots, n\}$ and let $C_R \geq 0$. We say that $\mathbf{A}$ has the null space property $NSP_{\mathbb{R}}^{\leq}(\mathcal{S}, C_R)$, and write $\mathbf{A} \in NSP_{\mathbb{R}}^{\leq}(\mathcal{S}, C_R)$, if*

$$C_R.\|\nu_{\mathcal{S}}\|_1 \leq \|\nu_{\overline{\mathcal{S}}}\|_1, \text{ for all } \nu \in \mathcal{N}(\mathbf{A}).$$

We say that $\mathbf{A}$ has the strict null space property $NSP_{\mathbb{R}}^{<}(\mathcal{S}, C_R)$, and write $\mathbf{A} \in NSP_{\mathbb{R}}^{<}(\mathcal{S}, C_R)$, if

$$C_R.\|\nu_{\mathcal{S}}\|_1 < \|\nu_{\overline{\mathcal{S}}}\|_1, \text{ for all } \nu \in \mathcal{N}(\mathbf{A}) \setminus \{\mathbf{0}\}.$$

The next performance metric (see, e.g., [BGI+08, CDD08]) for CS involves recovering approximations to signals that are not exactly k-sparse. This is the main performance metric we use in this paper:

Definition 6. *An ℓ_1/ℓ_1 approximation guarantee means that the Basis Pursuit LP relaxation outputs an estimate $\hat{\mathbf{x}}$ that is within a constant factor from the best k-sparse approximation for $\mathbf{x}$, i.e.,*

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_1 \leq 2\frac{C_R + 1}{C_R - 1} \cdot \min_{\mathbf{x}' \in \Sigma_{\mathbb{R}^n}^{(k)}} \|\mathbf{x} - \mathbf{x}'\|_1, \tag{3.3.1}$$

where $\Sigma_{\mathbb{R}^n}^{(k)} = \{\mathbf{w} \in \mathbb{R}^n \mid |\text{supp}(\mathbf{w})| \leq k\}$. The null space condition is a necessary and sufficient for a measurement matrix to yield ℓ_1/ℓ_1 approximation guarantees [XH08, DSV].

3.3.2 Channel Coding Preliminaries

A linear binary code $\mathcal{C}$ of length n is defined by a $m \times n$ parity-check matrix $\mathbf{H_{CC}}$ with operations over the binary vector field $\mathbb{F}_2^n$, i.e., $\mathcal{C} \triangleq \{\mathbf{x} \in \mathbb{F}_2^n | \mathbf{H_{CC}} \cdot \mathbf{x} = \mathbf{0}\}$. These vectors $\mathbf{x}$ are called the codewords of $\mathcal{C}$. The bipartite graph $\mathcal{G}$ that represents $\mathbf{H_{CC}}$ is called the Tanner graph of $\mathbf{H_{CC}}$, which has n variable and m check nodes. We define the set of variable indices $\mathcal{I} \triangleq \mathcal{I}(\mathbf{H_{CC}}) \triangleq \{1, \ldots, n\}$, the set of check indices $\mathcal{J} \triangleq \mathcal{J}(\mathbf{H_{CC}}) \triangleq \{1, \ldots, m\}$, the set of check indices that involves the i-th bit position $\mathcal{J}_i \triangleq \mathcal{J}_i(\mathbf{H_{CC}}) \triangleq \{j \in \mathcal{J} | [\mathbf{H_{CC}}]_{j,i} = 1\}$, and the set of codeword positions that are involved in the j-th check $\mathcal{I}_j \triangleq \mathcal{I}_j(\mathbf{H_{CC}}) \triangleq \{i \in \mathcal{I} | [\mathbf{H}]_{j,i} = 1\}$. For a regular code, the degrees of the codeword adjacency $|\mathcal{J}_i|$ and the size of check equations $|\mathcal{I}_j|$ are denoted by d_v and d_c, respectively. If a codeword $\mathbf{x} \in \mathcal{C}$ is transmitted through a channel and an output sequence $\mathbf{y}$ is received, then one can potentially decode $\mathbf{x}$ by solving for the maximum likelihood codeword in $\mathcal{C}$, namely

$$\begin{aligned} \textbf{CC-MLD:} \quad \text{minimize} \quad & \lambda^T \mathbf{x}' \\ \text{subject to} \quad & \mathbf{x}' \in \text{conv}(\mathcal{C}), \end{aligned}$$

where λ is the likelihood vector defined by $\lambda_i = \log(\frac{\mathbb{P}(y_i|x_i=0)}{\mathbb{P}(y_i|x_i=1)})$, and $\text{conv}(\mathcal{C})$ is the convex hull of all codewords of $\mathcal{C}$ in $\mathbb{R}^n$. **CC-MLD** solves the ML decoding problem by the virtue of the fact that the objective $\lambda^T \mathbf{x}$ is minimized on a corner point of $\text{conv}(\mathcal{C})$, which is a codeword. **CC-MLD** is NP-hard and therefore an efficient description of the exact codeword polytope is very unlikely to exist.

The well-known channel decoding LP relaxation is:

$$\begin{aligned} \textbf{CC-LPD:} \quad \text{minimize} \quad & \lambda^T \mathbf{x}' \\ \text{subject to} \quad & \mathbf{x}' \in \mathcal{P}(\mathbf{H_{CC}}), \end{aligned}$$

where $\mathcal{P} = \mathcal{P}(\mathbf{H_{CC}})$ is known as the fundamental polytope [FWK05, KV06]. The fundamental polytope is compactly described as follows: If h_j^T is the j-th row of $\mathbf{H_{CC}}$,

then

$$\mathcal{P} = \cap_{1 \leq j \leq m} \text{conv}(\mathcal{C}_j), \tag{3.3.2}$$

where $\mathcal{C}_j = \{x \in \mathbb{F}^n \mid h_j^T x = 0 \bmod 2\}$. Due to the symmetries of the fundamental polytope [FWK05] we can focus on the cone around the all-zeros codeword without loss of generality. Given the parity check matrix $\mathbf{H_{CC}}$, its fundamental cone $\mathcal{K}(\mathbf{H_{CC}})$ is defined as the smallest cone in $\mathbb{R}^n$ that encompasses $\mathcal{P}(\mathbf{H_{CC}})$, and can be formally defined as follows.

Definition 7. *The fundamental cone* $\mathcal{K} \triangleq \mathcal{K}(\mathbf{H_{CC}})$ *of* $\mathbf{H_{CC}}$ *is the set of all vectors* $\mathbf{w} \in \mathbb{R}^n$ *that satisfy*

$$w_i \geq 0, \qquad \textit{for all } i \in \mathcal{I}, \tag{3.3.3}$$

$$w_i \leq \textstyle\sum_{i' \in \mathcal{I}_j} w_{i'}, \qquad \textit{for all } i \in \mathcal{I}_j, j \in \mathcal{J}. \tag{3.3.4}$$

Given the fundamental cone of a code $\mathcal{C}$, we define the following property.

Definition 8. *Let* $S \subset \{1, 2, \cdots, n\}$ *and* $C_R \geq 1$ *be fixed. A code* $\mathcal{C}$ *with parity check matrix* $\mathbf{H_{CC}}$ *is said to have the fundamental cone property FCP*$(\mathcal{S}, C_R)$ *if for every* $\mathbf{w} \in \mathcal{K}(\mathbf{H_{CC}})$ *the following holds:*

$$C_R \|\mathbf{w}_{\mathcal{S}}\|_1 < \|\mathbf{w}_{\overline{\mathcal{S}}}\|_1. \tag{3.3.5}$$

We introduce the perturbed symmetric channel (PSC) shown in Figure 3.1, where each bit $+1$ (-1)[3] is flipped into $-C_R$ $(+C_R)$ with probability p and remains unchanged with probability $(1-p)$. For $C_R = 1$, the perturbed symmetric channel is the same as the binary symmetric channel (BSC). Note that the data can be received error free through the perturbed channel. Through solving **CC-LPD**, however, it is apparent that the recovery probability of the perturbed channel with $C_R > 1$ is less than BSC.

[3]Note that the bits 0,1 are mapped to +1,-1, respectively.

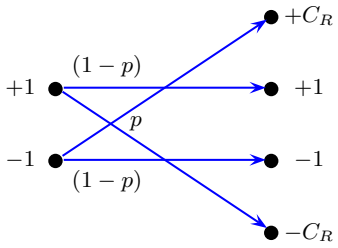


Figure 3.1: Perturbed symmetric channel model

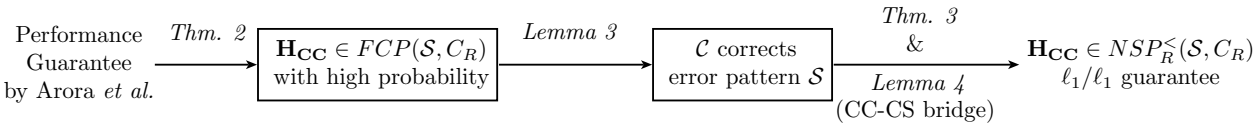


Figure 3.2: The procedure of importing the performance guarantee from LP decoding into compressed sensing

3.4 ℓ_1/ℓ_1 Guarantee for $\Omega(\log n)$-Girth Matrices

We start by stating the main theorem.

Theorem 3.4.1. *Let* $\mathbf{A}$ *be an* $m \times n$ *0-1 measurement matrix with girth* $g = \Omega(\log(n))$*, and column and row densities* d_v *and* d_c*, respectively. There exists a fraction* $p^*(d_c, d_v)$*, solely a function of* d_c *and* d_v*, so that for every* $0 \leq p \leq p^*(d_c, d_v)$*, there exists some* $C_R > 1$ *such that* $\mathbf{A}$ *provides* ℓ_1/ℓ_1 *guarantee (with parameter* C_R*) for the* ***CS-LPD*** *and for a randomly chosen support set* S *of size* $p \cdot n$*, with probability higher than* $1 - \mathcal{O}(e^{-\alpha n})$ *for some* $\alpha > 0$*.*

The function $p^*(d_c, d_v)$ and C_R are deterministic functions computed from a recursion on a tree (see Theorem 7.5.6). For example, for $d_c = 6, d_v = 3$, we can numerically conclude that $p* \geq 0.045$, and therefore we can recover almost all $k = 0.045n$-sparse signals. To the best of our knowledge, this is the best provable recovery threshold for sparse compressed sensing measurement matrices. Furthermore, for $p = 0.04$ we obtain $C_R \geq 1.08$, which means that for compressible signals, ℓ_1 minimization gives ℓ_1/ℓ_1 error bound, given by the constant $2\frac{C_R+1}{C_R-1} = 52$, i.e., the ℓ_1 norm of the recovery

error is bounded by a factor 52 of the ℓ_1 norm of the smallest $(1-0.04)n$ coefficient of the signal (see (3.3.1)).

Proof. To prove this theorem, we take the steps shown in Figure 3.2. We import the best performance guarantees for LDPC code $\mathcal{C}$ from LP decoding [ADS09] through the bridge established in [DSV]. First through Theorem 7.5.6, we show that the code $\mathcal{C}$ has the fundamental cone property $\mathrm{FCP}(\mathcal{S}, C_R)$ with probability at least $1-\mathcal{O}(e^{-\alpha n})$. Next, through Lemma 3.4.4, we demonstrate that $\mathcal{C}$ corrects the error configuration $\mathcal{S}$ at the output of the perturbed symmetric channel. Finally, by Theorem 3.4.6, we establish a connection between the properties of $\mathbf{H_{CC}}$ as parity check matrix (i.e., FCP condition) and its null space properties as a measurement matrix in compressed sensing. Consequently, we show that the solution of **CS-LPD** satisfies the ℓ_1/ℓ_1 condition. ■

Note that, if the girth is $g = \Omega(\log\log(n))$, then $\mathbf{H_{CC}}$ has ℓ_1/ℓ_1 guarantee with probability higher than $1-\mathcal{O}(1/n)$. Theorem 3.4.1 is our main result and we will build the blocks necessary to complete the proof in the remainder of this chapter.

3.4.1 Extension of CC-LPD

The work of Arora et al. [ADS09] provides the *best existing* guarantees for the performance of LP decoding. For a fixed code rate, the probabilistic analysis of [ADS09] yields a (weak) threshold p^* for the probability of bit flip, below which **CC-LPD** can recover the output of a binary bit flipping channel, with high probability, provided that the code has a girth at least doubly logarithmic in the size of the code. The resulting threshold is significantly tighter than the previously achieved bounds, namely those obtained in [DDKW08], which were established for expander-graph-based codes. To the best of our knowledge, the thresholds of [ADS09] are the highest proven thresholds for LP decoding, and, as mentioned, hold for codes that retain large girths, rather than expander codes.

Building upon the work of Arora et al., we further prove that codes with optimal girth maintain the FCP property in a weak notion. In other words, we prove that if the probability of bit flip p is sufficiently small, there is a finite $C_R > 1$ such that with high probability, the output of the perturbed symmetric channel in Figure 3.1 with C_R can be recovered by **CC-LPD**. The parameter C_R surely depends on p. We compute an upper bound on the achievable robustness factor C_R as a function of p, which approaches 1 as p approaches the recoverable threshold p^* of LP decoding based on the analysis of [ADS09]. Using the connection between the **CS-LPD** and **CC-LPD**, this result will allow us to explicitly find the relationship between the ℓ_1/ℓ_1 robustness factor C_R and the fraction p when the results are translated to the context of compressed sensing. In order to state the main theorem of this section, first we define η to be a random variable that takes the value $-C_R$ with probability p and value 1 with probability $1-p$, i.e., the output of the PSC when 0 is transmitted. Also, let the sequences of random variables $X_i, Y_i,\ i \geq 0$ be defined in the following way:

$$\begin{aligned} Y_0 &= \eta, \\ X_i &= \min\{Y_i^{(1)}, \ldots, Y_i^{(d_c-1)}\} \quad \forall i > 0, \\ Y_i &= 2^i\eta + X_{i-1}^{(1)} + \cdots + X_{i-1}^{(d_v-1)} \quad \forall i > 0, \end{aligned} \tag{3.4.1}$$

where $X^{(j)}$s are independent copies of a random variable X.

Theorem 3.4.2. *Let $\mathcal{C}$ be a regular (d_v, d_c)-LDPC code with girth equal to g, and $0 \leq p \leq 1/2$ be the probability of bit flip, and S be the random set of flipped bits. If for some $j \in \mathbb{N}$,*

$$c = \gamma^{1/(d_v-2)} \min_{t \geq 0} \mathbb{E}e^{-tX_j} < 1$$

where $\gamma = (d_c-1)\frac{C_R+1}{C_R}(\frac{C_R \cdot p}{1-p})^{1/(C_R+1)}(1-p) < 1$, then with probability at least $1 - O(n)c^{d_v(d_v-1)^{T-1}}$, the code $\mathcal{C}$ has the FCP$(\mathcal{S}, C_R)$, where T is any integer with $j \leq T < g/4$.

Note that the value c can be derived for specific values of p, d_v, d_c, and C_R. The proof of the above theorem falls along the same lines as the arguments of [ADS09]. The bottom line is the existence of a certificate in the primal LP problem for the success of LP decoding, which can be extended to the case of the perturbed channel. In order to define the certificate, we first bring the following definitions from [ADS09]. In the sequel, we denote the bipartite graph that corresponds to $\mathbf{A}$ by $\mathcal{G}$.

Definition 9. *A tree $\mathcal{T}$ of height $2T$ is called a skinny subtree of $\mathcal{G}$, if it is rooted at some variable node v_{i_0}, for every variable node v in $\mathcal{T}$ all the neighboring check nodes of v in $\mathcal{G}$ are also present in $\mathcal{T}$, and for every check node c in $\mathcal{T}$ exactly two neighboring variable nodes of c in $\mathcal{G}$ are present in $\mathcal{T}$.*

Definition 10. *Let $\mathbf{w} \in [0,1]^T$ be a fixed vector. A vector $\beta^{(\mathbf{w})}$ is called a minimal T-local deviation, if there is a skinny subtree of $\mathcal{G}$ of height $2T$, say $\mathcal{T}$, so that for every variable node v_i $1 \leq i \leq n$,*

$$\beta_i^{(w)} = \begin{cases} w_{h(i)} & \text{if } v_i \in \mathcal{T} \setminus \{v_{i_0}\} \\ 0 & \text{otherwise} \end{cases},$$

where $h_i = \frac{1}{2} d(v_{i_0}, v_i)$.

The key to the derivation of a certificate for LP decoding is the following lemma:

Lemma 3.4.3 (Lemma 1 of [ADS09])**.** *For any vector $\mathbf{z} \in \mathcal{P}$, and any positive vector $\mathbf{w} \in [0,1]^T$, there exists a distribution on the minimal T-local deviations $\beta^{(\mathbf{w})}$, such that*

$$\mathbb{E}\beta^{(\mathbf{w})} = \alpha \mathbf{z},$$

where $0 < \alpha \leq 1$.

Lemma 7.5.5 has the following interpretation: If a linear property holds for all minimal T-local deviations ($f(\beta^{(\mathbf{w})}) \geq 0$, where $f(.)$ is a linear function), then it also holds for all pseudo-codewords ($f(\mathbf{z}) \geq 0 \ \forall \mathbf{z} \in \mathcal{P}$). Interestingly enough, the success of LP

decoding over the perturbed symmetric channel of Figure 3.1 for a given set of bit flips $\mathcal{S}$ has a linear certificate, namely FCP$(\mathcal{S}, C_R)$. In other words, if we define:

$$f_C^{(\mathcal{S})}(x) = \sum_{i \in \overline{\mathcal{S}}} x_i - C_R \sum_{i \in \mathcal{S}} x_i, \tag{3.4.2}$$

then LP decoder is successful, if and only if $f_{C_R}^{(\mathcal{S})}(z) \geq 0$ for every pseudocodeword $z \in \mathcal{P}$. Therefore, according to Lemma 7.5.5, it suffices that the condition be true for all T-local deviations, which is equivalent to FCP$(\mathcal{S}, C_R)$.

Proof of Theorem 7.5.6. We denote the set of variable nodes and check nodes by X_v and X_c, respectively. For a fixed $\mathbf{w} \in [0,1]^T$, let $\mathcal{B}$ be the set of all minimal T-local deviations, and $\mathcal{B}_i$ be the set of minimal T-local deviations defined over a skinny tree rooted at the variable node v_i. Also, assume S is the random set of flipped bits, when the flip probability is p. Interchangeably, we also use S to refer to the set of variable nodes corresponding to the flipped bits indices. We are interested in the probability that for all $\beta^{(\mathbf{w})} \in \mathcal{B}$, $f_C^{(S)}(\beta^{(\mathbf{w})}) \geq 0$. For simplicity we denote this event by $f_C^{(\mathcal{S})}(\mathcal{B}) \geq 0$. Since the bits are flipped independently and with the same probability, we have the following union bound

$$\mathbb{P}\left(f_C^{(\mathcal{S})}(\mathcal{B}) \geq 0\right) \geq 1 - n\mathbb{P}\left(f_C^{(\mathcal{S})}(\mathcal{B}_1) \geq 0\right). \tag{3.4.3}$$

Now consider the full tree of height 2T, that is rooted at the node v_0, and contains every node u in $\mathcal{G}$ that is no more than $2T$ distant from v, i.e., $d(v_0, u) \leq 2T$. We denote this tree by $B(v_0, 2T)$. To every variable node u of $B(v_0, 2T)$, we assign a label, $I(u)$, which is equal to $-C_R\omega_{h(u)}$ if $u \in S$, and is $\omega_{h(u)}$ if $u \in S^c$, where $(\omega_0, \omega_2, \cdots, \omega_{2T-2}) = w$. We can now see that the event $f_C^{(\mathcal{S})}(\mathcal{B}_1) \geq 0$ is equivalent to the event that for all skinny subtrees $\mathcal{T}$ of $B(v_0, 2T)$ of height $2T$, the sum of the labels on the variable nodes of $\mathcal{T}$ is positive. In other words, if Γ_1 is the set of all skinny trees of height $2T$ that are rooted at v_0, then $f_C^{(\mathcal{S})}(\mathcal{B}_1) \geq 0$ is equivalent to:

$$\min_{\mathcal{T}\in\Gamma_1} \sum_{v\in\mathcal{T}\cap X_v} I(v) \geq 0. \tag{3.4.4}$$

We assign to each node u (either check or variable node) of $B(v_0, 2T)$ a random variable Z_u, which is equal to the contribution to the quantity $\min_{\mathcal{T}\in\Gamma_1} \sum_{v\in\mathcal{T}\cap X_v} I(v)$ by the offspring of the node u in the tree $B(v_0, 2T)$, and the node u itself. The value of Z_u can be determined recursively from all of its children. Furthermore, the distribution of Z_u only depends on the height of u in $B(v_0, 2T)$. Therefore, to find the distribution of Z_u, we use $X_0, X_1, \cdots, X_{T-1}$ as random variables with the same distribution as Z_u when u is a variable node (X_0 is assigned to the lowest-level variable node) and likewise $Y_1, Y_2, \cdots, Y_{T-1}$ for the check nodes. It then follows that:

$$\begin{aligned} Y_0 &= \omega_0 \eta_C, \\ X_i &= \min\{Y_i^{(1)}, \ldots, Y_i^{(d_c-1)}\} \quad \forall i > 0, \\ Y_i &= \omega_i \eta_C + X_{i-1}^{(1)} + \cdots + X_{i-1}^{(d_v-1)} \quad \forall i > 0, \end{aligned} \tag{3.4.5}$$

where $X^{(j)}$s are independent copies of a random variable X, and η_C is a random variable that takes the value $-C_R$ with probability p and value 1 with probability $1-p$. It follows that

$$\begin{aligned} \mathbb{P}\left(f_C^{(S)}(\mathcal{B}_1) \leq 0\right) &= \mathbb{P}\left(X_{T-1}^{(1)} + \cdots + X_{T-1}^{(d_v)} \leq 0\right) \\ &\leq (\mathbb{E}(e^{-tX_{T-1}}))^{d_v}. \end{aligned} \tag{3.4.6}$$

The last inequality is by Markov inequality and is true for all $t > 0$. The rest of the proof is modifications to the proof of Lemma 8 in [ADS09], for the Laplace transform evolution of the variables X_is and Y_is, to account for a non-unitary robustness factor C_R. By upper bounding the Laplace transform of the variables recursively, it is possible to show that (see Lemma 8 of [ADS09], the argument is exactly the same for our case)

$$\begin{aligned}\mathbb{E}e^{-tX_i} &\leq \left(\mathbb{E}e^{-tX_j}\right)^{(d_v-1)^{i-j}} \\ &\quad \prod_{0\leq k\leq i-j-1} \left((d_c-1)\mathbb{E}e^{-tw_{i-k}\eta}\right)^{(d_v-1)^k}, \end{aligned} \tag{3.4.7}$$

for all $1 \leq j \leq i < T$. If we take the weight vector as $\mathbf{w} = (1, 2, \cdots, 2^j, \rho, \rho, \cdots, \rho)$ for some integer $1 \leq j < T$, and use (7.9.13) for $i = T-1$, we obtain:

$$\begin{aligned}\mathbb{E}e^{-tX_{T-1}} \leq\ & (\mathbb{E}e^{-tX_j})^{(d_v-1)^{T-j-1}} \\ & \cdot \left((d_c-1)\mathbb{E}e^{-t\rho\eta}\right)^{\frac{(dv-1)^{T-j-1}-1}{d_v-2}}.\end{aligned}$$

ρ and t can be chosen to jointly minimize $\mathbb{E}e^{-tX_j}$ and $\mathbb{E}e^{-t\rho\eta}$ in the above, which along with (7.9.12) results in

$$\begin{aligned}\mathbb{P}(f_C^{\mathcal{S}}(\mathcal{B}_1 \leq 0)) &\leq (\mathbb{E}e^{-tX_{T-1}})^{d_v} \\ &\leq \gamma^{-d_v/(d_v-2)} \times c^{d_v(d_v-1)^{T-j-1}},\end{aligned}$$

where $\gamma = (d_c-1)\frac{C_R+1}{C_R}(1-p)(\frac{C_R \cdot p}{1-p})^{1/(C_R+1)}$ and $c = \gamma^{1/(d_v-2)} \min_{t\geq 0} \mathbb{E}e^{-tX_j}$. If $c < 1$, then probability of error tends to zero as stated in Theorem 7.5.6. ■

In the next lemma, we show that if a code $\mathcal{C}$ has the fundamental cone property FCP($\mathcal{S}, C_R$), then **CC-LPD** can correct the perturbed symmetric channel error configuration $\mathcal{S}$.

Lemma 3.4.4. *Let $\mathbf{H_{CC}}$ be a parity-check matrix of a code $\mathcal{C}$ and let $\mathcal{S} \subset \mathcal{I}(\mathbf{H_{CC}})$ be a particular set of coordinate indices that are flipped by a perturbed channel with cross-over probability $p > 0$. If and only if $\mathbf{H_{CC}}$ has the FCP($\mathcal{S}, C_R$), then the solution of **CC-LPD** equals the codeword that was sent.*

Proof. Without loss of generality, we can assume that the all-zero codeword is transmitted. We begin by proving the sufficiency. Let $+1$ be the log-likelihood ratio

associated with a received 0, and let $-C_R < -1$ be the log-likelihood ratio associated with a received 1. Therefore,

$$\lambda_i = \begin{cases} +1 & \text{if } i \in \mathcal{S} \\ -C_R & \text{if } i \in \overline{\mathcal{S}} \end{cases}.$$

Then it follows from the assumptions in the lemma statement that for any $\mathbf{w} \in \mathcal{K}(\mathbf{H_{CC}}) \setminus \{\mathbf{0}\}$

$$\begin{aligned} \lambda^T w &= \sum_{i \in \mathcal{S}} (+1).w_i + \sum_{i \in \overline{\mathcal{S}}} (-C_R).w_i \\ &\overset{(a)}{=} +1.\|\mathbf{w}_{\mathcal{S}}\|_1 - C_R.\|\mathbf{w}_{\overline{\mathcal{S}}}\|_1 > 0 \overset{(b)}{=} \lambda^T \cdot \mathbf{0}, \end{aligned}$$

where step (a) follows from the fact that $|w_i| = w_i$ for all $i \in \mathcal{I}(\mathbf{H_{CC}})$, and where step (b) follows from (3.3.5). Therefore, under **CC-LPD** the all-zero codeword has the lowest cost function value when compared to all nonzero pseudo-codewords in the fundamental cone, and therefore also compared to all nonzero pseudo-codewords in the fundamental polytope.

Note that the proof of the converse is direct and can easily be derived by taking the sufficiency proof steps, backward. ■

3.4.2 Establishing the Connection

In this section, through Lemma 3.4.5 (taken from [SV09]), we will establish a bridge between LP decoding and compressed sensing. Specifically, using this bridge, we will import performance results from LP-decoding context into compressed sensing.

Lemma 3.4.5. *(Lemma 6 in [SV09]): Let* $\mathbf{H_{CC}}$ *be a zero-one measurement matrix. Then*

$$\nu \in \mathcal{N}(\mathbf{H_{CC}}) \Rightarrow |\nu| \in \mathcal{K}(\mathbf{H_{CC}}).$$

Proof. Let $\mathbf{w} \triangleq |\nu|$. To show that such a vector $\mathbf{w}$ is indeed in the fundamental cone of $\mathbf{H_{CC}}$, we need to verify (3.3.3) and (3.3.4). It is apparent that $\mathbf{w}$ satisfies

(3.3.3). Therefore, let us focus on the proof that $\mathbf{w}$ satisfies (3.3.4). Namely, from $\nu \in \mathcal{N}(\mathbf{H_{CC}})$ it follows that for all $j \in \mathcal{J}$, $\sum_{i\in\mathcal{I}} h_{j,i}\nu_i = 0$, i.e., for all $j \in \mathcal{J}$, $\sum_{i\in\mathcal{I}_j} \nu_i = 0$. This implies

$$w_i = |\nu_i| = \left| - \sum_{i'\in\mathcal{I}_j\setminus\{i\}} \nu_{i'} \right| \leq \sum_{i'\in\mathcal{I}_j\setminus\{i\}} |\nu_{i'}| = \sum_{i'\in\mathcal{I}_j\setminus\{i\}} w_{i'},$$

for all $j \in \mathcal{J}$ and all $i \in \mathcal{I}_j$, showing that $\mathbf{w}$ indeed satisfies (3.3.4). ■

From here on, we deal with $\mathbf{H_{CC}}$ as a zero-one measurement matrix. Note that the bridge established in [DSV] and through Lemma 3.4.5 connects **CC-LPD** of the binary linear channel code and **CS- LPD** based on a zero-one measurement matrix over reals by viewing this binary parity-check matrix as a measurement matrix. This connection allows the translation of performance guarantees from one setup to the other. Using this bridge, we can show that parity-check matrices of "good" channel codes can be used as provably "good" measurement matrices under basis pursuit.

Using the bridge of Lemma 3.4.5, we show in the next theorem that the parity-check matrix $\mathbf{H_{CC}}$, as a measurement matrix, has the null space property, and as a result satisfies the ℓ_1/ℓ_1 guarantee.

Theorem 3.4.6. *Let $\mathbf{H_{CC}} \in \{0,1\}^{m\times n}$ be a parity-check matrix of the code $\mathcal{C}$ and let k be a nonnegative integer. Further assume that code $\mathcal{C}$ can correct the error configuration $\mathcal{S}$ over the perturbed symmetric channel where $|\mathcal{S}| \leq k$. Additionally, assume that $\mathbf{s} = \mathbf{H_{CC}} \cdot \mathbf{e}$. Then $\mathbf{H_{CC}}$ as a measurement matrix satisfies*

$$\mathbf{H_{CC}} \in NSP^{<}_{\mathbb{R}}(\mathcal{S}, C_R).$$

*Furthermore, the solution $\hat{\mathbf{e}}$ produced by **CS-LPD** will satisfy*

$$\|\mathbf{e} - \hat{\mathbf{e}}\|_1 \leq 2.\frac{C_R + 1}{C_R - 1}.\|\mathbf{e}_{\overline{\mathcal{S}}}\|_1.$$

Proof. We begin by proving the null space property. Since the code $\mathcal{C}$ corrects the configuration $\mathcal{S}$, from Lemma 3.4.4, for any point in the fundamental cone $\mathcal{K}(\mathbf{H_{CC}})$

including $\mathbf{w}$ and any set $\mathcal{S} \subset \{1, \ldots, n\}$, with $|\mathcal{S}| \leq k$, we have

$$C_R \|\mathbf{w}_{\mathcal{S}}\|_1 < \|\mathbf{w}_{\overline{\mathcal{S}}}\|_1. \tag{3.4.8}$$

We prove by contradiction. Assume $\mathbf{H_{CC}}$ does not have the strict null space property $NSP^{<}_{\mathbb{R}}(\mathcal{S}, C_R)$, i.e., there exists a point $\nu \in \mathcal{N}(\mathbf{H_{CC}})$ such that

$$C_R \|\nu_{\mathcal{S}}\|_1 \geq \|\nu_{\overline{\mathcal{S}}}\|_1.$$

Further, from Lemma 3.4.5, we know there exists $\mathbf{w} = |\nu|$ in $\mathcal{K}(\mathbf{H_{CC}})$. And

$$\begin{aligned} C_R \|\mathbf{w}_{\mathcal{S}}\|_1 &= C_R \||\nu_{\mathcal{S}}|\|_1 \\ &= C_R \|\nu_{\mathcal{S}}\|_1 \\ &\geq \|\nu_{\overline{\mathcal{S}}}\|_1 \\ &= \||\nu_{\overline{\mathcal{S}}}|\|_1 \\ &= \|\mathbf{w}_{\overline{\mathcal{S}}}\|_1, \end{aligned}$$

which contradicts the assumption and shows that no such a point ν exists.

We showed that $\mathbf{H_{CC}}$ has the claimed null space property. Since $\mathbf{H_{CC}} \cdot \mathbf{e} = \mathbf{s}$ and $\mathbf{H_{CC}} \cdot \hat{\mathbf{e}} = \mathbf{s}$, it easily follows that $\nu \triangleq \mathbf{e} - \hat{\mathbf{e}}$ is in the null space of $\mathbf{H_{CC}}$. So

$$\begin{aligned} \|\mathbf{e}_{\mathcal{S}}\|_1 + \|\mathbf{e}_{\overline{\mathcal{S}}}\|_1 &= \|\mathbf{e}\|_1 \\ &\overset{(a)}{\geq} \|\hat{\mathbf{e}}\|_1 \\ &= \|\mathbf{e} - \nu\|_1 \\ &= \|\mathbf{e}_{\mathcal{S}} - \nu_{\mathcal{S}}\|_1 + \|\mathbf{e}_{\overline{\mathcal{S}}} - \nu_{\overline{\mathcal{S}}}\|_1 \\ &\overset{(b)}{\geq} \|\mathbf{e}_{\mathcal{S}}\|_1 - \|\nu_{\mathcal{S}}\|_1 + \|\nu_{\overline{\mathcal{S}}}\|_1 - \|\mathbf{e}_{\overline{\mathcal{S}}}\|_1 \\ &\overset{(c)}{\geq} \|\mathbf{e}_{\mathcal{S}}\|_1 - \frac{C_R - 1}{C_R + 1}.\|\nu\|_1 - \|e_{\overline{\mathcal{S}}}\|_1, \end{aligned} \tag{3.4.9}$$

where step (a) follows from the fact that the solution of **CS-LPD** satisfies $\|\mathbf{e}\|_1 \leq \|\hat{\mathbf{e}}\|_1$, where step (b) follows from applying the triangle inequality property of the

ℓ_1-norm twice, where step (c) follows from

$$
\begin{aligned}
(C_R+1). \quad & (-\|\nu_{\mathcal{S}}\|_1 + \|\nu_{\overline{\mathcal{S}}}\|_1) \\
& = -C_R.\|\nu_{\mathcal{S}}\|_1 - \|\nu_{\mathcal{S}}\|_1 + C_R.\|\nu_{\overline{\mathcal{S}}}\|_1 + \|\nu_{\overline{\mathcal{S}}}\|_1 \\
& \overset{(d)}{\geq} -\|\nu_{\overline{\mathcal{S}}}\|_1 - \|\nu_{\mathcal{S}}\|_1 + C_R.\|\nu_{\overline{\mathcal{S}}}\|_1 + C_R.\|\nu_{\mathcal{S}}\|_1 \\
& = (C_R-1).\|\nu_{\mathcal{S}}\|_1 + (C_R-1).\|\nu_{\overline{\mathcal{S}}}\|_1 \\
& = (C_R-1)\|\nu\|_1,
\end{aligned}
$$

and where step (d) follows from applying twice the fact that $\nu \in \mathcal{N}(\mathbf{H_{CC}})$ and the assumption that $\mathbf{H_{CC}} \in NSP^{\leq}_{\mathbb{R}}(\mathcal{S}, C_R)$. Subtracting the term $\|\mathbf{e}_{\mathcal{S}}\|_1$ on both sides of (3.4.9), and solving for $\|\nu\|_1 = \|\mathbf{e} - \hat{\mathbf{e}}\|_1$ yields the promised result. ■

This theorem was the last piece in the proof of Theorem 3.4.1, and, with it, the proof is complete.

Note that it is easy to obtain deterministic constructions of measurement matrices that are regular, have the optimal order of measurements, and have $\Omega(\log n)$ girth. Therefore, a byproduct of our result is the explicit construction of measurement matrices with $m = cn$ rows which can recover almost all sparse signals with sparsity $k = p \cdot n$. These will be parity-check matrices of codes with girth $\Omega(\log n)$. To explicitly obtain such codes, we can use one of the known deterministic matrix constructions such as the progressive edge-growth (PEG) Tanner graphs [HEA05] or the deterministic construction suggested by Gallager [Gal63].

Chapter 4

Summary-Based Structures

The combinatorial structures based on minimal expanders and large girth graphs given in the last two chapters were designed in particular to be congruent with the basis pursuit recovery algorithms, and were analyzed in the "linear" regime, where the number of measurements M and the signal sparsity k are both linear in the ambient dimension N (i.e., $M = \Theta(N)$ and $k = \Theta(N)$).[1] In contrast, when the design criterion or the regime of interest in the problem dimensions varies, other factors come into play that determine the suitability of a particular matrix. Depending on how a CS-based system is implemented or what the underlying signal model is, the optimal choice of measurements change. For example, when compressed sensing is considered for spectrum-sensing applications or enhancement of analog-to-digital converters, measurements are applied as real-time sketches, often realized by a high-speed correlation circuit. In this case, the implementation of (pseudo)-random sketches is more difficult than certain structured measurements, whereas the density of the matrix is somewhat irrelevant. Similar constrains exist for real-time compressed sensing encoders that are based on other forms of physical integration or temporal modulation, such as the photonic strobing method used in [VRR11], or single-pixel imaging [DDT+08]. On the other hand, there are several other applications where the measurement matrix is a

[1] Only in this chapter we use N for the signal dimension and M for the number of measurements, as n, m are reserved for other parameters.

pre-built construction and is applied in other physical forms, such as in integrated pixel arrays, or CS-based DNA micro-arrays. As discussed in the previous chapter, in most of these cases the density of the matrix dictates the design cost, thus the desire to optimize it.

The decoder complexity is another governing factor that affects the choice of the measurement matrix. When decoding (i.e., reconstructing) is an off-line process happening independently of encoding, and the signal dimension is not extensive, reconstructing methods based on geometrical approaches such as Basis Pursuit or Matching Pursuit algorithms are ideal, and the measurement matrix can be assigned accordingly. The complexity of such decoding algorithms is polynomial in the signal dimension. In contrast, for real-time decoding in applications like streaming or hardware embedded systems, or for the cases where the signal size is extremely large (say exponentially), efficiency of the decoding is a bottleneck. In these cases, even the most efficient polynomial time algorithms are impractical, and combinatorial methods with complexity sublinear in the dimension of the signal are preferred, obviously limiting the choice of the measurement matrix.

In this chapter, we present the design of a new class of very structured combinatorial matrices called "summary-based structures" that lead to very efficient decoding algorithms, namely with complexities that grow logarithmically in the signal dimension. In addition, due to the particular structure of the proposed matrices and their minimal storage requirements, we believe that they are ideal candidates for real-time streaming encoding applications, in addition to some specific statistical inference problems.

Φ	measurement matrix
N	signal size
M	number of measurements
k	sparsity of the signal
n, d	parameters of a summary, $n = \log_2 N$
$\mathcal{C}$	summary codebook

4.1 Introduction

Compressed sensing facilitates the joint sampling and linear compression of high-dimensional sparse signals, based on the premises of which the information content of a large data set which is known to be sparsely representable in some linear domain (a.k.a dictionary) can be retrieved from linear projections of the data onto a lower-dimensional space. The simplicity of the encoding mechanism, universal performance guarantees, and the existence of reconstruction methods that do not rely on thorough understanding of the underlying signal model has made compressive sampling a major subject of interest in research on dimensionality reduction methods and high-speed analog signal acquisition. Upon proper implementation, sampling of signal streams at sub-Nyquist rates becomes feasible, and obvious advantages to real-time systems are introduced in terms of memory usage, time efficiency, and power consumption.

Stemming from the seminal work of Donoho et al. [Don06a] and Candès et al. [CT05], linear-programming-decoding based methods (a.k.a Basis Pursuit) and their variations were the first proposals for polynomial-time recovery techniques with provable universal performance bounds for perfect sparse reconstruction, and were shown to be robust against noise in performance [CRT06a, XH08, CWB08, CY08, KXAH10, DDFG10]. Linear programming decoding has $\mathcal{O}(N^3)$ complexity, when N is the signal dimension, and presumably attains its best performance for random matrix ensembles Φ. In contrast, a plethora of alternative recovery schemes were proposed having their roots in combinatorical/greedy perspectives of the sparse reconstruction problem, rather than the original high-dimensional geometrical approaches, as discussed to some extent in the previous chapters. These recovery methods are advantages to Basis Pursuit in different aspects such as simplicity of implementation, less complexity, and better performance bounds, and are often based on

deterministic choices of measurement matrices. Selected examples of such methods are Matching Pursuit and its variations [TG07, DTDS06, NV07], CoSamp algorithm [NT08], iterative thresholding methods [FR08, BD08], combinatorial algorithms based on expander graphs [BI09, BGI$^+$08, JXHC09, KXDH10], list-decoding algorithms for constructions based on algebraic error-correcting codes (Reed-Solomon or Reed-Muller codes) [PH08, HCS08], algorithms based on data-streaming applications [CM05, CM04], message-passing types of algorithms [CSW10, DMP11, LMP08], Chaining Pursuit [GSTV06], and many more.

Despite significant advances in compressed sensing and sparse recovery methods, certain aspects of this field remain relatively immature. Thus far, CS has been viewed primarily as a data-acquisition technique. As a result, the applicability of CS to other computational applications has not enjoyed commensurate investigation. In addition, to the best of the authors' knowledge, there is no implementation of a unified CS system for practical real-time applications. A few recent works have addressed the former by applying sparse reconstruction ideas to certain inference problems, including learning and adaptive computational schemes (e.g., [JS08, JYG09, Cev09, XMT11, KKH11]). Several other works have addressed the latter by designing hardware, which exploits the fact that CS enables the monitoring of a given bandwidth at a much lower sampling rate than traditional Nyquist-based methods (see, e.g., [Eld09]). The motivating factor behind these works is that for a given maximum sampling rate achievable by digitizing hardware (limited by the poor power consumption scaling with sampling rate), it is possible to either acquire signals over a much greater bandwidth, or with much less power for a given bandwidth. Recent work, inspired by this line of thought, has led to the development of hardware CS encoders (see, e.g., [LKD$^+$07, ME10, LLP08]). However, none of the previous works address the problem of real-time signal decoding, which is a critical requirement in many applications.

Although variant by the nature of the problem and physical constraints, perhaps

two bottlenecks in the practical implementation of CS are the following:

- Construction of measurement matrices that are provably good, certifiable, and inexpensive to implement. Many fundamental results of compressed sensing are limited to asymptotically large signal sizes and random ensembles, and are not optimally designed for practical purposes.

- Time-efficient and robust recovery algorithms. For real-time systems with very large signal spaces, most polynomial time algorithms are inefficient, given the excessive time and memory requirements. Recovery algorithms with sublinear complexity are more desirable for practical systems and hardware implementation.

The aim of this chapter is to introduce and provide an analysis for a sparse reconstruction system that addresses the aforementioned problems, and to allude to the extensions of CS in the less-explored directions. This is done by introducing a new class of combinatorial matrices that have two main advantages: 1)they are very structured and easily implementable with minimum memory requirements, 2)they lead to very efficient sublinear reconstruction algorithms. In addition, the proposed structures have practical motivations in certain statistical inference problems.

4.2 Related Work

The construction of deterministic matrices with provable guarantees for CS is an active ongoing field of research, discussed to some extent in previous chapters. Examples of recent work along this line can be found in [PH08, DeV07, CHJ10, KTDH11, Ind08, AM11] and the references therein. The results are, however, far from a consensus on optimal choices of measurement matrices for difference regimes of problem dimensions, signal-to-noise ratio, and other physical constraints.

There are also a handful of results on sublinear sparse recovery in the literature, mostly leveraging combinatorial structures. Most of the previous work can be found in [HCS08, BGI+08, DMP11, GSTV06, GSTV07, SBB06, CM06]. Cormode at al. [CM06] proposed a class of combinatorial structures and corresponding reconstruction methods based on group testing techniques which in the best case have a running time complexity of $\Omega(k^6 \log^6 N)$, and guarantee that $M = \Omega(k^4 \log^4 k)$ measurements are sufficient. The method developed by Gilbert et al. [GSTV06] is called Chaining Pursuit (CP), which is based on structured matrices with $M = \mathcal{O}(k \log^2 N)$ rows, and has a complexity of $\mathcal{O}(k \log^2 k \log^2 N)$. In addition, it provides a weak ℓ_1/ℓ_1 robustness to noise. In practice, however, the performance of the algorithm suffers from a large oversampling overhead which makes it impractical except for extremely large dimensions. In addition, the storage requirement for the method is proportional to signal dimension N. The chirp reconstruction method of Howard et al. [HCS08] is based on the second-order Reed-Muller codes originally designed for channel coding. The complexity of this algorithm is $\mathcal{O}(kM \log M)$ [HSC09]. Experimental evidence indicates that the method has a superior performance bound (in terms of recoverability region) to the method of Cormode et al. and the Chaining Pursuit, and is more resilient to relatively higher levels of observation noise for small sparsity levels k. However, for large signal dimensions and moderate values of sparsity k (say beyond 60), the running time is extremely high, and thus inefficient for practical purposes. To the best of our knowledge, there is no rigorous analysis of the performance of this method. A belief-propagation algorithm was introduced in [DMP11] based on dense structured matrices with high column coherence resulting in good RIP coefficients. The sublinear algorithm has $\mathcal{O}(k^2 \log N)$ complexity per iteration, and an indefinite convergence time and lacks theoretical guarantee (and existing empirical evidence) for robustness to noise. In contrast, the sublinear algorithm proposed in this paper is highly scalable to large system dimensions and moderate to large values of k.

The suggested algorithm retains a superior performance bound and empirical running time compared to most existing methods. In the presence of noise, our method is comparable to the chirp reconstruction method of [HCS08]. A limitation of the proposed algorithm is that the existing theoretical guarantees are only for the sparse signals where the disjoint subsets of entries have different sums of values, referred to as distinguishable vectors throughout the paper.

4.3 Contributions

We introduce a new class of measurement matrices for sparse recovery that are deterministic, structured, and highly scalable. The constructions are based on labeling the ambient state space with binary sequences of length $n = \log_2 N$, and summing up entries of $\mathbf{x}$ that share the same pattern (up to a fixed length) at various locations in their labeling sequences. Unlike most existing deterministic structures, the class of corresponding matrices of this paper do not maintain high column coherence, and, as such, are RIP-less. However, we propose recovery guarantees for a particular instance of ℓ_1 minimization when these constructions are used, and thus prove that, despite lacking RIP, our matrices are congruent with Basis Pursuit. In addition, we provide two efficient combinatorial algorithms along with theoretical guarantees for the defined structures. The proposed algorithms are sublinear in the ambient dimension of the signal. In particular, we propose a summarized index recovery (SIR) algorithm with a running time of $\mathcal{O}(kM \log M))$ that requires $M = \mathcal{O}(k \log N \log\log N)$ measurements to recover k-sparse vectors, and has an empirical oversampling factor significantly better than the existing sublinear methods. Due to the structure of the measurements and the advantages of decoding algorithms, we believe that the proposed compression/decompression framework is amenable to real-time CS implementation, and offers substantial simplification in the design of existing CS encoder/decoders. Furthermore, observations collected based on the proposed constructions appear as

low-order statistics or "summaries" in a number of practical situations in which a similar intrinsic labeling of the state space exists. This includes certain inference and discrete optimization problems such as market basket (commodity bundle) analysis, advertising, online recommendation systems, genomic feature selection, social networks, neighbor discovery in ad-hoc networks, etc. Some of these applications will be elaborated in more detail in the remainder of the paper.

4.4 Preliminaries

A vector $\mathbf{x}$ is called "k-sparse", if it has at most k nonzero entries. The support set of a sparse vector is the index set of its nonzero coefficients. A vector $\mathbf{x}$ is called "distinguishable" if no two disjoint subsets of the entries of $\mathbf{x}$ have the exact same sum of values. The following definitions will be useful in describing the proposed measurement structures:

Definition 11. *Let m, n, and d be integers. A (n, d) summary is a pair $X = (S, \mathbf{c})$, where S is a subset of $\{1, 2, \cdots, n\}$ of size d, and $\mathbf{c}$ is a binary label (sequence) of length d. A (m, n, d) summary codebook is a collection $\mathcal{C} = \{(S_i, \mathbf{c}_j) \mid 1 \leq i \leq m,\ 0 \leq j \leq 2^d - 1\}$ of (n, d) summaries, where S_is are distinct subsets, and $\mathbf{c}_j$ is the length d binary representation of the integer j. If $m = \binom{n}{d}$, $\mathcal{C}$ is called the complete (n, d) summary codebook.*

We also set the following conventions. We say that a binary label $\mathbf{b}$ of length n "conforms" to a (n, d) summary $(S, \mathbf{c})$, or interchangeably that the summary $(S, \mathbf{c})$ "appears" in $\mathbf{b}$ iff $\mathbf{b}(S) = \mathbf{c}$, where $\mathbf{b}(S)$ is the subsequence of $\mathbf{b}$ indexed by the elements of the set S in the same order that they appear in $\mathbf{b}$. In addition, two (n, d) summaries $(S, \mathbf{c})$ and $(S', \mathbf{c}')$ are said to conform, if there is a binary label $\mathbf{b}$ of length n that conforms to both of them. We use the operators $\|$ and $\nparallel$ to denote conformity and its complement, respectively. For a binary sequence $\mathbf{b}$, we denote its

decimal numerical value by $\Delta(\mathbf{b})$. The following definition will also be useful later in the technical discussions.

Definition 12. *For a set S and a number x, $\mathcal{I}(S, x)$ indicates the number of elements of S that are less or equal than x. Consequently, if the elements of S are listed in increasing order as $s_1 < s_2 < \cdots < s_d$, then $\mathcal{I}(S, s_i) = i \ \forall 1 \leq i \leq d$.*

4.5 Proposed Measurement Structures

To a given (m, n, d) summary codebook $\mathcal{C}$ (Definition 11), we associate a binary matrix Φ of size $M \times N$ where $M = 2^d \times m$ and $N = 2^n$, in the following way. For every $(S, \mathbf{c}) \in \mathcal{C}$, there is a row $\phi = (\varphi_1, \ldots, \varphi_N)$ in Φ that satisfies:

$$\varphi_j = \begin{cases} 1 & \mathbf{b}_j(S) = \mathbf{c} \\ 0 & \text{else} \end{cases} \quad 1 \leq j \leq N, \tag{4.5.1}$$

where $\mathbf{b}_j$ is the n-bit binary representation of j, and $\mathbf{b}(S)$ is the subsequence of the binary sequence $\mathbf{b}$, indexed by the elements of the set S in increasing order. In other words, ϕ has a 1 in the jth coordinate if the binary representation of $j-1$ conforms to $(S, \mathbf{c})$. Note that every column of Φ has exactly m ones, and each row has exactly 2^{n-d} ones.

Example 2. Consider a $(n = 4, d = 2)$ summary (S, d) defined by $S = \{1, 2\}$ and $\mathbf{c} = 10$. This summary conforms to every binary label of length 4 in which the first two leftmost bits are 1 and 0, respectively. A simpler characterization of this summary (or equivalently all binary labels that conform to it) is in terms of a ternary label $10 \times \times$, where the $\times$ symbols can be replaced by either 1 or 0. All such binary labels are listed in Figure 4.1. If these labels are converted to decimal values and increased by 1, they provide the indices of the columns where there is a 1 in the considered row, namely $9, 10, 11$, and 12. The full row ϕ of length 16 is displayed in Figure 4.1 as well.

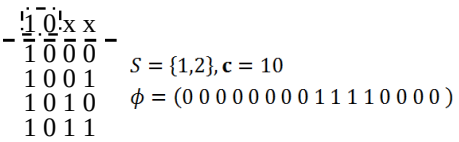


Figure 4.1: An example $(4, 2)$ summary and the corresponding row of the structured measurement matrix

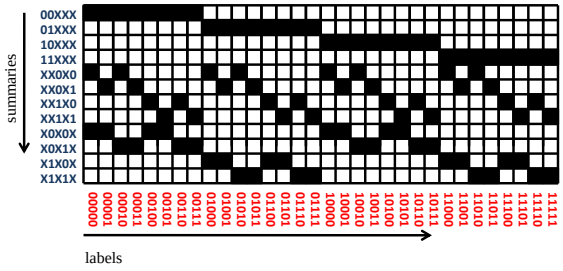


Figure 4.2: An example of a measurement matrix constructed based on a $(5, 2)$ summary codebook. Black is 1 and white is 0

Example 3. A 12×32 binary matrix is shown in Figure 4.2 which corresponds to a $(5, 2)$ summary codebook with 12 summaries. The rows of the displayed matrix are labeled with corresponding summaries and the columns are labeled by distinct binary sequences of length 5. Entries colored in black represent 1, and white represents 0.

Example 4. Another sample matrix of size 40×32768 which is based on a randomly generated $(15, 2)$ summary codebook is depicted in Figure 4.3.

4.6 Practical Motivations

The primary constraints and considerations that are used when designing CS matrices often require tailoring the structure to be more amenable to one or more sets of analysis techniques. For example, the original proposed matrices used for CS used entries that were completely randomly generated from either a Gaussian or Bernoulli distribution, or randomly sampled from some structured sets of functions such as Fourier basis. In contrast, the matrices defined in this paper were designed based on a very structured labeling of the ambient space (a binary vector field), which results in

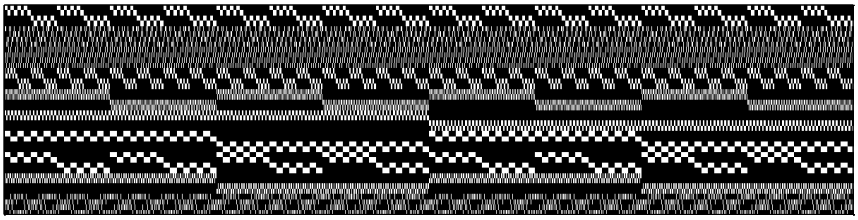


Figure 4.3: An example of a measurement matrix of size 40×32768 constructed based on a random $(15, 2)$ summary codebook of size 40

a number of advantages not present in the previous forms of measurement structures. For any application in which compressed sensing is used to analyze sets of data which are typically organized by several parameters, this organization is implicitly mapped to the indexing of the input vector. The indexing is almost always used as a method of classifying different types of data. Thus, this strategy of designing a CS matrix with a structure that exploits relationships of the matrix entries to the inference vector entries has an intuitive appeal in applications where analysis of organized data sets is being conducted. This is the case in problems such as market basket analysis or feature optimization, DNA micro-arrays, etc., as will be elaborated in the remainder of this section. In addition, the logarithmic indexing of the column space of the matrix facilitates an efficient storing mechanism and sublinear column space search techniques, eventually leading to sublinear reconstruction algorithms described later in this paper. These features, along with simplicity of description and real-time implementation of the matrices as the output of a high-speed correlation circuit, makes the proposed structures of this paper ideal candidates for large-scale implementation of several CS-related frameworks. In the current paper, we focus on two particular instances where the proposed constructions are useful.

4.6.1 Market Basket Analysis

This example stresses the usefulness of the proposed structures in problems where the given signal space retains an intrinsic structured labeling similar to those used

for defining the measurement matrix. The problem of market basket analysis is to learn and predict the popular shopping trends of customers given limited available training samples, which can be described in a more general setup as the following feature optimization problem: Assume that a set $\mathcal{F} = \{F_1, F_2, \cdots, F_n\}$ of *features* (or commodities) is available, and assume that certain accumulations (baskets) of features form *lucrative* profiles (or popular baskets). In particular, a lucrative profile can be a subset of features which is representable by a binary sequence $\mathbf{b} = b_1 b_2 \dots b_n$, where b_i determines the presence of the ith feature in the profile. A practical assumption is that lucrative profiles are limited and weighted, meaning that their profitabilities are different. The vector $\mathbf{x} = (p_1, p_2, \dots, p_{2^n})^T$ formed by the respective profits of all feature collections is thus an approximately sparse vector. Furthermore, in most real-world situations, the available information about the profitability of profiles is often derived from a pool of observations (e.g., surveys), and is mostly limited to some forms of summaries. More specifically, it might be assumed that the average profitability of a certain configuration of d features is learnable. For example, the average profit of a profile in which F_1 and F_2 are present and F_3 is absent (regardless of all other features) can be estimated as $\overline{p}$. The collection of such summaries form an observation vector $\mathbf{y}$, which is related to the desired $\mathbf{x}$ through a set of linear equations $\mathbf{y} = \Phi\mathbf{x}$. A careful examination reveals that Φ has a form similar to those obtained by using summary codebooks. Similar settings arise in many practical applications, such as market basket (commodity bundle) analysis, where the objective is to configure the structure of a market that complies the best with the needs and the behaviors of the customers. To that end, it is essential to understand which market configurations are lucrative and what packages of features (e.g., commodities, pricing options, interest rates, etc.) should be offered to customers, and with what percentages.

4.6.2 Wireless Ad-Hoc Networks

A particular instance of compressed sampling and reconstruction can be exploited in wireless ad-hoc networks for an important network discovery problem [ZLG]. Consider a large ad-hoc network where nodes are assigned unique IP addresses of length n. A critical feature in such networks is the ability of individual nodes to identify their neighbors, namely the nodes that are within an effective communication range. In lieu of centralized data aggregation and hand-shaking protocols, a random access protocol is often implemented by individual nodes, whereby every node accesses the channel and transmits its messages at random time slots. This way, with high probability, every node can transmit messages (IP address for instance) to the neighbors without interfering with others, provided that enough number of time intervals have elapsed. An alternative method based on CS is to map every IP address to a unique signature of length M, and then have every node simultaneously transmit its signature. The signal received at every node is therefore a complex linear combination of the signatures of neighboring nodes. Since every node has a few neighbors (far smaller than the total number of available IP addresses), this problem can be cast as a sparse reconstruction problem. The corresponding measurement matrix is of size $M \times 2^n$, where columns are indexed by IP addresses, and the numerical value of each column is equal to the signature of its corresponding IP. This scheme is illustrated in Figure 4.4, where it is assumed that only users A_1, A_3, A_5, A_6 are active and transmitting. The resulting time series vector at the receiver is thus a sparse linear combination of the columns of the matrix Φ with coefficients given by channel gains h_1, h_3, h_5, h_6, plus additive noise $\mathbf{v}$.

This model was proposed by Guo et al. in [ZLG], where it was shown that the described CS-based protocol can significantly reduce the number of transmission time slots required for neighbor discovery compared with the random access protocol ALOHA. Due to the enormous IP space size, a sublinear recovery algorithm is the only practical solution to the CS decoding for this problem at the present time.

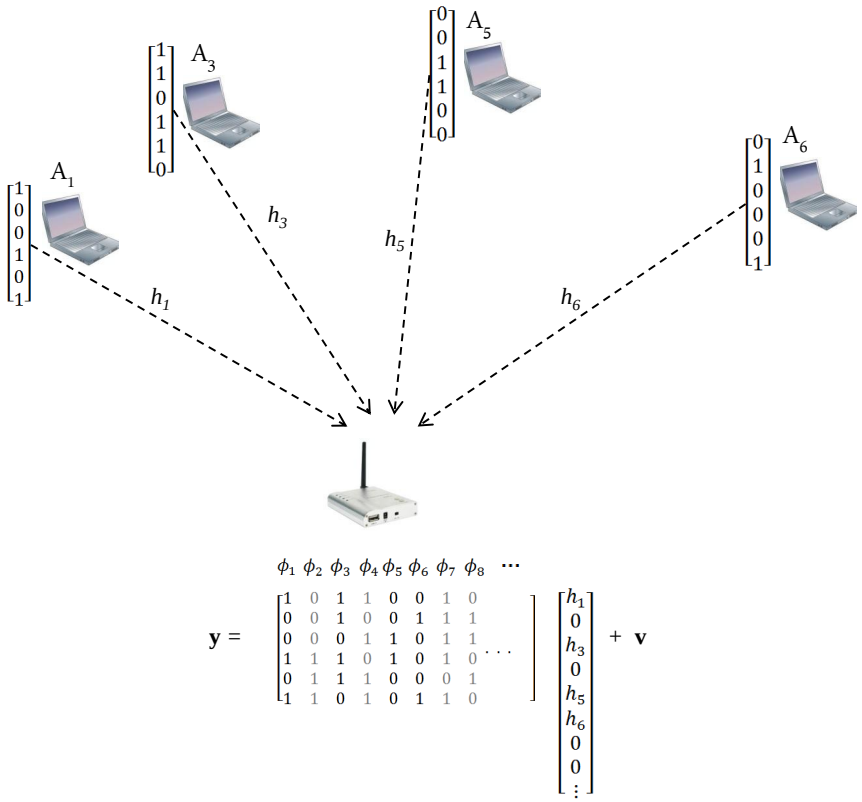


Figure 4.4: Demonstration of a CS-based communication protocol in an ad-hoc network

In [ZLG] the authors have used the Reed-Muller-based measurement matrices and the chirp reconstruction algorithm of [HCS08] as a candidate solution which allows neighbor discovery with up to 30 neighboring nodes. However, experiments reveal that the running time of the chirp reconstruction algorithm severely increases for $k \geq 60$ nonzero coefficients, and becomes practically intractable. Considering that a limited amount of processing resources and time is often dedicated to learning the configuration and initial setup of the network, this approach might not be of practical use for denser and larger networks. In contrast, the sublinear recovery algorithm that we propose for the summary-based constructions is highly scalable with respect to very large state space sizes and sparsity levels k.

4.7 Proposed Recovery Algorithms

For the measurement matrices described in the previous section we propose three reconstruction algorithms and provide success guarantees. These algorithms include the Basis Pursuit (a.k.a ℓ_1 minimization), as well as two fast algorithms that can recover sparse vectors from a sublinear number of measurements and using a sublinear number of operations.

4.7.1 Basis Pursuit

ℓ_1 minimization is the closest convex optimization relaxation to the problem of finding the sparsest signal $\mathbf{x}$ that satisfies the set of linear inequalities $\mathbf{y} = \Phi\mathbf{x}$. In the case of Basis Pursuit, we restrict our attention to nonnegative signals, and the following version of the ℓ_1 minimization algorithm:

$$\begin{aligned} &\text{minimize} \quad \|\mathbf{x}\|_1 \\ &\text{subject to } \Phi\mathbf{x} = \mathbf{y},\ \mathbf{x} \geq 0. \end{aligned} \tag{4.7.1}$$

The complexity of (4.7.1) is generally polynomial in the ambient dimension of the signal. Specifically, one can implement (4.7.1) in $\mathcal{O}(N^3)$ operations, without exploiting any of the available structural information of the measurement matrix. Although there are some advantages to Basis Pursuit compared with other sparse recovery algorithms, such as higher robustness to noise and universal performance bounds, its complexity is impractical for problems where N scales exponentially. In these situations, sublinear time algorithms are preferred.

The success of the basis pursuit algorithm for recovering sparse signals is certified by several conditions, two major classes of which are the restricted isometry property (RIP) [CT05] and the null space condition [CT05, SXH08]. It is provable that the measurement structures defined in this paper do not maintain the RIP properties,

due to the existence of columns with fairly large coherence. This however does not discard the suitability of these constructions for ℓ_1 minimization, since RIP is known to provide a sufficient condition (see, e.g., [BCT10]). Instead, we refer to the analysis presented in Chapter 2 for the nonnegative constrained ℓ_1 minimization, and prove that certain null space conditions hold for the considered class of matrices, and therefore provide a sparse signal recovery bound.

4.7.2 Summarized Index Recovery

The summarized index recovery (SIR) algorithm is based on iteratively inferring the nonzero entries of the signal based on one of the distinct values of $\mathbf{y}$ and its various occurrences. In every iteration of the algorithm, one support index label is identified. The main loop thus contains the following subroutines:

1) The *value classification* subroutine scans through the entries of $\mathbf{y}$ and groups together the nonzero coefficients with (almost) equal values. In the presence of observation noise, in all likelihood, no two values of the vector $\mathbf{y}$ are identical. To cancel out the effect of the noise on the value identification, some sort of quantization (binning) is used: Two coefficients of $\mathbf{y}$ are considered almost equal if they are within a pre-assigned bin size q distant.

2) The *coarse index identification* subroutine cycles through all the occurrences of a particular coefficient of the vector $\mathbf{y}$ grouped together by the value identification subroutine. For each group, the corresponding summaries are identified, and the subroutine attempts to identify a binary label that conforms to all of them. More specifically, suppose that a nonzero value of $\mathbf{y}$ is chosen which has t occurrences, say without loss of generality, $y_1 = y_2 = \cdots = y_t$. Also, let the (n, d) summary which corresponds to the ith row of Φ be denoted by $(S_i, \mathbf{c}_i)$. The subroutine explores the possibility that $y_1, y_2, \ldots, y_t$ are all equal to a single nonzero entry of $\mathbf{x}$, by trying to build a binary sequence $\mathbf{b}$ that conforms to the summaries $\{(S_i, \mathbf{c}_i)\}_{i=1}^{t}$, i.e., by

setting:

$$\mathbf{b}(S_i) := \mathbf{c}_i, \ \forall 1 \leq i \leq t. \tag{4.7.2}$$

In the presence of noise, this subroutine settles for finding a label that conforms to as many summaries as possible among $\{(S_i, \mathbf{c}_i)\}_{i=1}^{t}$. There are a number of ways to do this efficiently. A possible formulation is through an equivalent max-clique solution. Consider an undirected graph $\mathcal{G} = (V, E)$, in which there is a node v_i for every summary $(S_i, \mathbf{c}_i)$, for $1 \leq i \leq t$. Two nodes v_i and v_j are connected in $\mathcal{G}$, if and only if $(S_i, \mathbf{c}_i)||(S_j, \mathbf{c}_j)$. The problem of finding a binary label $\mathbf{b}$ that conforms to as many summaries as possible is equivalent to finding the maximum size clique in $\mathcal{G}$.

3) In the case that the enforced label is not unique and contains a few undetermined bits (identified by -1 in Algorithm 2), the *fine index identification* subroutine tries to estimate the remaining bits by a series of simple value counting rules. In the presence of noise, this turns into a set of hypothesis tests on particular subsets of coefficients of $\mathbf{y}$.

4) Finally, the rejection step verifies whether the estimated label is actually in the support set or not. If a certain statistical criteria for the measurements containing the estimated index is not satisfied, the label is rejected and the search continues. Specifically, if $\mathbf{b}$ is the binary label of a support index of a distinguishable $\mathbf{x}$, then we expect all of the coefficients of $\mathbf{y}$ that correspond to summaries conforming with $\mathbf{b}$ to be (almost) nonzero. If otherwise is observed, the inferred label is rejected.

A block diagram describing these fundamental steps and their interconnections is depicted in Figure 4.5. Also, the details are described in Algorithm 2. The proposed method can be implemented very efficiently, with $\mathcal{O}(\max(poly(M), k \log N))$ operations, which is sublinear in the dimension of the signal. To avoid exponentially long computations, the maximum clique solution of the coarse index identification subroutine can be approximated via polynomial time algorithms (see, e.g., [PX94]

Algorithm 2 —Summarized Index Recovery

Input: Vector $\mathbf{y} = (y_1, y_2, \ldots, y_M)^T$ and corresponding summaries $(S_j, \mathbf{c}_j),\ 1 \leq j \leq M$.
Output: Estimate $\hat{\mathbf{x}}$ of the sparse vector $\mathbf{x}$.
Initialize: Set $\hat{\mathbf{x}} := 0$.
Set $\mathcal{S} := \{S_j \ | 1 \leq j \leq M\}$
Identify $\Gamma := \{y_j \ | 1 \leq j \leq M \ y_j \not\approx 0\}$
for $\gamma \in \Gamma$ **do**
  Set $J := \{j | y_j = \gamma\}$
  if $\nexists j', j'' \in J$ s.t. $(S_{j'}, \mathbf{c}_{j'}) \nparallel (S_{j''}, \mathbf{c}_{j''})$ **then**
    Set $\mathbf{b} := -\mathbf{1}_{n\times 1}$.
    for $j \in J$ **do** $\mathbf{b}(S_j) := \mathbf{c}_j$ **end for**
  end if
  Set $S_1 := \{1 \leq i \leq n \mid \mathbf{b}(i) \neq -1\}$ and $S_2 := \{1, 2, \ldots, n\} \setminus S_1$.
  if $S_2 = \emptyset$ **then**
    Set $V := \{y_i | 1 \leq i \leq M, \mathbf{b} || (S_i, \mathbf{c}_i)\}$.
    if $0 \notin V$ **then** Set $t := \Delta(\mathbf{b}) + 1$, $\hat{\mathbf{x}}_t := \gamma$, $\mathbf{y} := \mathbf{y} - \Phi\hat{\mathbf{x}}$, **goto** 5 **end if**
  else
    for $S \in P(S)$ and $l \in S_2 \cap S$ **do**
      $R_0 := \{y_j | S_j = S, \mathbf{b} \| (S_j, \mathbf{c}_j), \mathbf{c}_j(\mathcal{I}(S, l)) = 0\}$
      $R_1 := \{y_j | S_j = S, \mathbf{b} \| (S_j, \mathbf{c}_j), \mathbf{c}_j(\mathcal{I}(S, l)) = 1\}$
      if $R_0 \approx \{0\}$ **then** $\mathbf{b}(l) := 1$, **goto** 12. **end if**
      if $R_1 \approx \{0\}$ **then** $\mathbf{b}(l) := 0$, **goto** 12. **end if**
    end for
  end if
end for

for a survey of different approaches). Here, we suggest using a greedy-based maximum degree selection strategy which has a linear complexity in the graph size. The remaining computational advantage of the SIR algorithm is owed for the most part to the structured form of the measurement matrix which facilitates sublinear search over its column space. In addition, the memory requirements for decoding are minimal (i.e., are not exponential in n), since the information about Φ and the current inferred indices of the unknown vector at each stage can be retained by only storing the corresponding binary indices.

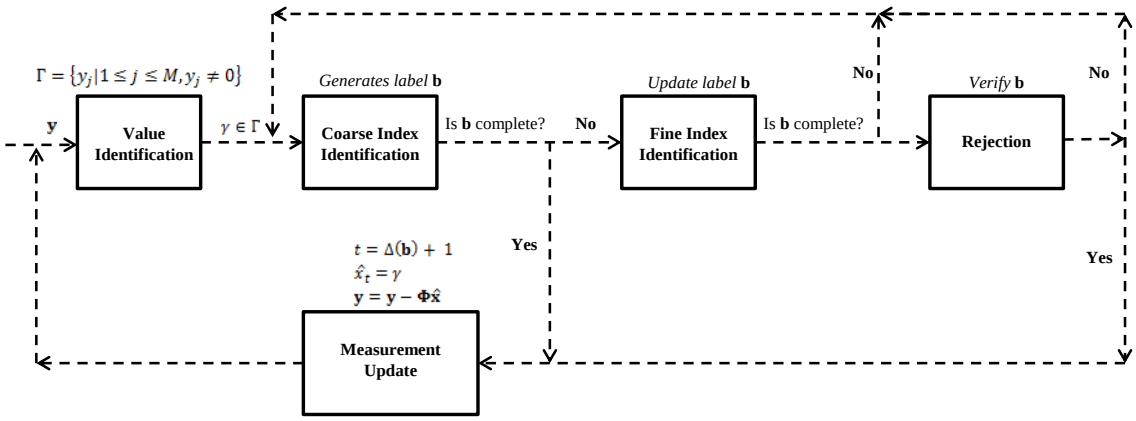


Figure 4.5: Block diagram describing the subroutines of Algorithm 2

4.7.3 Mix and Match Algorithm

We describe a third recovery method, which is along the lines of the algorithm proposed in [JS08], with slight modifications. The algorithm consists of two subroutines: a value identification phase in which the nonzero values of the unknown signal are determined, and a second phase for identifying the support set of $\mathbf{x}$. The method is based on measurements given by $\mathbf{y} = (\mathbf{y}^{(1)T}, \mathbf{y}^{(2)T})^T = (\Phi_1^T, \Phi_2^T)^T\mathbf{x}$, where only $\mathbf{y}^{(1)}$ is used for the first phase, and $\mathbf{y}^{(2)}$ and Φ_2 are used in the second phase. Details are described in Algorithm 3. We analyze this algorithm for the proposed measurement structures of this paper, which is different from the analysis of [JS08]. The complexity of Algorithm 3 is $\mathcal{O}(\max(poly(M), 2^k))$, and thus explodes when k grows. Therefore, this method is only practical for very small sparsity levels k.

4.8 Analysis

For the sake of the theoretical arguments that appear in the remainder of this section, we need to define the following notions:

Definition 13. *Let n and l be integers with $l < n$. Define $f_s(n, d)$, $f_w(n, d, p, \epsilon)$ and $f'_w(n, d, p, \epsilon)$ to be the largest integer k such that when k binary sequences of length n are selected at random, the following happens respectively:*

Algorithm 3 —Mix and Match

Input: Vector $\mathbf{y} = (y_1, y_2, \ldots, y_M)^T$ and corresponding summaries $(S_j, \mathbf{c}_j),\ 1 \leq j \leq M$.
Output: A set X of nonzero values of $\mathbf{x}$, and binary sequences $\mathbf{b}_x, x \in X$ of length n, representing the indices of the support set of $\mathbf{x}$.
Initialize: Set $V := \emptyset, S := \emptyset, X := \emptyset$ and a mapping $E : \mathbb{R} \to 2^{\mathbb{R}}$ with $E(r) = \emptyset, \forall r \in \mathbb{R}$.
Identify $\Gamma := \{\gamma \mid \gamma \neq 0,\ \exists 1 \leq i \leq M \text{ s.t. } \gamma = y_i\}$.
while $\gamma \nsubseteq S$ **do**
    Find $\gamma^* := \min\{\gamma \mid \gamma \in \Gamma,\ \gamma \notin S\}$.
    Set $X := X \cup \{\gamma\},\ E(\gamma + s) := E(s) \cup \{\gamma\}, \forall s \in S$.
    Set $S := X \cup \{\gamma + s \mid s \in S\}$.
end while
Set $\mathbf{b}_x = -\mathbf{1}_n, \forall x \in X$.
for $j = 1 : M$ **do**
    if $y_j \neq 0$ **then**
        for $x \in E(y_j)$ **do**
            $\mathbf{b}_x(S_j) := \mathbf{c}_j$.
        end for
    end if
end for

1. *With probability 1, there exists a (n, d) summary that appears in exactly one of the sequences.*

2. *With probability at least p, for each of the binary sequences, at least a fraction ϵ of its appearing (n, d) summaries are unique.*

3. *With probability at least p, for each of the binary sequences, at least a fraction ϵ of its appearing (n, d) summaries that include the first bit are unique.*

4.8.1 ℓ_1 Minimization

The general performance of the recovery algorithm (4.7.1) for binary matrices was studied in Chapter 1, which was based on the results published in [KXDH10]. We prove that a nonnegative vector $\mathbf{x}$ can be recovered from (4.7.1), if and only if it is the unique nonnegative solution of the given system of equations. For simplicity, we repeat the statement here:

Lemma 4.8.1 (Corollary of Theorem 2.5.3)**.** *Suppose* $\Phi \in \mathbb{R}^{m \times n}$ *is a matrix with constant column sum, and* $\mathbf{x}_0 \in \mathbb{R}^{n \times 1}$ *is a nonnegative vector.* $\mathbf{x}_0$ *is the unique solution to (4.7.1) if and only if* $\mathbf{x}_0$ *is the unique nonnegative solution to* $\Phi\mathbf{x} = \Phi\mathbf{x}_0$*.*

Using the above lemma and noting that the matrices constructed based on summary codebooks are constant column sum, we can evaluate the performance of the Basis Pursuit algorithm when used with the presented measurement matrices. The following theorem is fundamental to this analysis, and its proof appears in Section 4.10.

Theorem 4.8.2 (Strong Recovery for Basis Pursuit)**.** *Let* $k \leq f_s(n, d-1)$ *be an integer, and let* Φ *correspond to a complete* (n, d) *summary codebook. Then every* k*-sparse nonnegative vector* $\mathbf{x}$ *is perfectly recovered by (4.7.1).*

4.8.2 SIR Algorithm

The following two theorems are also proved in Section 4.10.

Theorem 4.8.3 (Strong Recovery for SIR)**.** *Let* $k \leq f_s(n, d-1)$ *be an integer, and let* Φ *correspond to a complete* (n, d) *summary codebook. Then every* k*-sparse distinguishable vector* $\mathbf{x}$ *is perfectly recovered by Algorithm 2.*

Theorem 4.8.4 (Weak Recovery for SIR)**.** *Let* $k \leq f'_w(n, d, p, \epsilon)$ *be an integer, and let* Φ *correspond to a random* (n, m, d) *summary codebook. Then, a random* k*-sparse distinguishable vector* $\mathbf{x}$ *is recovered by Algorithm 2 with probability at least* $1 - kn\left(1 - p + p(1 - \frac{\epsilon d}{n})^m\right)$*.*

The explicit recovery bounds given by the above theorems are calculated in Section 4.8.4.

4.8.3 M&M Algorithm

We prove the following weak recovery guarantee for the M&M algorithm.

Theorem 4.8.5 (Weak Recovery for M&M)**.** *Let $k \leq f_w(n, d, p, \epsilon)$ be an integer, and $\Phi = \left(\Phi_1^T \Phi_2^T\right)^T$, where Φ_1 and Φ_2 correspond to a random (m, n, d) summary codebook, and a complete $(n, 1)$ summary codebook, respectively. Then, a random nonnegative k-sparse distinguishable vector $\mathbf{x}$ is recovered by Algorithm 3 with probability at least $p\,(1 - k(1-\epsilon)^m)$.*

4.8.4 Recovery Bounds

We derive recovery bounds for (4.7.1) and Algorithms 2 and 3 by obtaining explicit bounds on the terms of Definition 13 and replacing them in the recovery guarantees of Section 4.8, namely Theorems 4.8.2–4.8.5. The proof of the following lemma is based on combinatorial techniques and Chernoff concentration bounds (see Section 4.10).

Lemma 4.8.6. *Let n,d, and k be integers, and $0 < \alpha < 1/2$. Also, let $\epsilon = 1 - k\binom{\frac{n}{2}(1+\sqrt{2\alpha})}{d} / \binom{n}{d}$, and $p = 1 - k^2 e^{-\alpha n}$. Then,*

1. *$f_s(n, d) \geq 2^d$.*
2. *$f_w(n, d, p, \epsilon) \geq k$.*
3. *$f'_w(n, d, p, \epsilon) \geq f_w(n-1, d-1, p, \epsilon)$.*

By exploiting the expressions of the above lemma in Theorems 4.8.2–4.8.5, we obtain the following bounds for different methods:

Basis Pursuit

If a complete (n, d) summary codebook is used to build Φ, then the number of measurements is $M = 2^d \binom{n}{d}$, and every sparsity $k \leq 2^{d-1}$ is guaranteed to be recovered. When put together (recall that $n = \log_2 N$), an upper bound on the the required number of measurements for recoverable sparsity k is given by:

$$M = 2k\binom{\log N}{\log k}. \tag{4.8.1}$$

In particular, for small values of k, the above bound is comparable with the $M = 2k \log N$ bound of ℓ_1 minimization for random Gaussian matrices [CRPW].

SIR Algorithm

We focus on the weak bound, namely the one obtained from Theorem 4.8.4. The general strategy is to take the values of p and ϵ according to Lemma 4.8.6, and choose k and m in such a way that firstly, ϵ is bounded away from zero, and secondly the probability of recovery failure approaches zero as $n \to \infty$. We take $k = \lambda 2^{-d \log_2\left(\sqrt{\alpha/2}+1/2\right)}$ for some constant $0 < \lambda < 1$, $\epsilon = 1 - k\binom{\frac{n}{2}(1+\sqrt{2\alpha})}{d}/\binom{n}{d}$, and $p = 1 - k^2 e^{-\alpha n}$. From Lemma 4.8.6, we conclude that:

$$k \leq f_w(n, d, p, \epsilon) \leq f'_w(n+1, d+1, p, \epsilon). \tag{4.8.2}$$

Therefore, from Theorem 4.8.4, the probability of failure in the reconstruction of a random vector $\mathbf{x}$ by the SIR algorithm is at most:

$$\begin{aligned} \mathbb{P}\left(\textit{failure}\right) &\leq kn\left(1 - p + p\left(1 - \epsilon\frac{d+1}{n+1}\right)^m\right) && (4.8.3)\\ &= kn\left(k^2 e^{-\alpha n} + p\left(1 - \left(1 - k\frac{\binom{\frac{n}{2}(1+\sqrt{2\alpha})}{d}}{\binom{n}{d}}\right)\frac{d+1}{n+1}\right)^m\right). && (4.8.4)\end{aligned}$$

Define $\beta = \sqrt{\alpha/2} + 1/2$. Then replacing k with $\lambda 2^{-d\log_2\beta}$, and p with the trivial upper bound 1 on the right-hand side of (4.8.4), it follows that:

$$\mathbb{P}\left(\textit{failure}\right) \leq kn\left(k^2 e^{-\alpha n} + \left(1 - \left(1 - \lambda 2^{-d\log_2\beta}\frac{\binom{\beta n}{d}}{\binom{n}{d}}\right)\frac{d+1}{n+1}\right)^m\right). \tag{4.8.5}$$

Note that for $1/2 < \beta < 1$ we have:

$$\begin{aligned} 2^{-d\log_2\beta}\frac{\binom{\beta n}{d}}{\binom{n}{d}} &= \beta^{-d} \times \frac{\beta n}{n} \times \frac{\beta n - 1}{n-1} \times \cdots \times \frac{\beta n - d + 1}{n - d + 1} && (4.8.6)\\ &\leq \beta^{-d} \times \beta^d = 1, && (4.8.7)\end{aligned}$$

and as a result, we may write:

$$\mathbb{P}\left(\textit{failure}\right) \leq k^3 n e^{-\alpha n} + kn\left(1-(1-\lambda)\frac{d}{n}\right)^m. \tag{4.8.8}$$

Note that in deriving the above inequality we have used the fact that $\frac{d}{n} < \frac{d+1}{n+1}$. The right-hand side of (4.8.10) goes to zero for large n, if $m = \Omega(n \log n)$. To see this, assume that $m > cn \log n$ for some $c > 0$. Then we have:

$$\mathbb{P}\left(\textit{failure}\right) \leq k^3 n e^{-\alpha n} + kn\left(1-(1-\lambda)\frac{d}{n}\right)^m \tag{4.8.9}$$

$$\leq k^3 n e^{-\alpha n} + kn\left(1-\frac{1-\lambda}{n}\right)^{cn\log n} \to kn^{1-c(1-\lambda)}, \text{ as } n \to \infty, \tag{4.8.10}$$

which is asymptotically zero for $c > \frac{2}{1-\lambda}$. Therefore, it follows that an upper bound on the required number of measurements for successful recovery with high probability is given by:

$$M = \mathcal{O}(k \log N \log\log N). \tag{4.8.11}$$

M&M Algorithm

We take $k = \lambda 2^{-d\log_2\left(\sqrt{\alpha/2}+1/2\right)}$ for some $0 < \lambda < 1$, $\epsilon = 1 - k\binom{\frac{n}{2}(1+\sqrt{2\alpha})}{d}/\binom{n}{d}$, and $p = 1 - k^2 e^{-\alpha n}$. From Lemma 4.8.6, we conclude that:

$$k \leq f_w(n, d, p, \epsilon). \tag{4.8.12}$$

Therefore from Theorem 4.8.5, we conclude that the probability of recovery failure for the M&M algorithm satisfies:

$$\mathbb{P}\left(\textit{failure}\right) \leq k^2 e^{-\alpha n} + k\lambda^m, \tag{4.8.13}$$

which asymptotically vanishes if $m = \Omega(\log k)$. Recall that the number of measurements in this case is determined by the matrix $\Phi = \left(\Phi_1^T, \Phi_2^T\right)^T$ described in Theorem

Table 4.1: Comparison of different sublinear compressed sensing algorithms. Red color indicates that the guarantee is only empirical.

Method	M	**Complexity**	**Storage**	**Robust**
CP [GSTV06]	$\mathcal{O}(k \log^2 N)$	$\mathcal{O}(k \log^2 k \log^2 N)$	$\Omega(N)$	Y
[CM06]	$\Omega(k^4 \log^4 N)$	$\Omega(k^4 \log^4 N)$	$\Omega(MN)$	Y
RM [HCS08]	——	$\Omega(kM \log M)$	$\mathcal{O}(M \log N)$	Y
[DMP11]	——	$\mathcal{O}(k^2 \log N)$ per iteration	$\mathcal{O}(M \log N)$	——
M&M	$\mathcal{O}(\log N + k \log k)$	$\mathcal{O}(M \log M + M2^k)$	$\mathcal{O}(M \log N)$	N
SIR	$\mathcal{O}(k \log N \log \log N)$	$\mathcal{O}(kM \log M)$	$\mathcal{O}(M \log N)$	Y

4.8.5, which is equal to $M = 2 \log N + m \times 2^d$. Therefore, it follows that an upper bound on the required number of measurements for successful recovery with high probability is given by:

$$M = 2 \log N + \mathcal{O}(k \log k). \tag{4.8.14}$$

In particular, when $k = o(\log \log N)$, it follows that only $\mathcal{O}(\log N)$ measurements are required, and the running time of the algorithm is $\mathcal{O}(\log N)$ (see Section 4.7), both of which are almost optimal.

Table 4.1 provides a more comprehensive comparison between the measurement matrix structure and the sublinear time algorithms proposed here with the ones that exist in the literature. We have enumerated the following factors in each case: minimum number of required measurements, running time complexity, storage requirement for the measurement matrix, and whether or not reconstruction is robust to noise. It is evident that the SIR algorithm is superior in most of these attributed to the other methods.

4.9 Simulations

Since Algorithm 3 is only efficient for very small values of k, we present the empirical performance of Algorithm 2 here and compare it with the existing sublinear algorithms in the literature. Due to the efficiency of the method, it is possible to

perform simulations for very large values of N. In Figure 4.6, the empirical required over-sampling rate for Algorithm 2 and the proposed constructions is plotted versus the signal dimension N, for various sparsity levels k. The required criteria here is that the probability of successful recovery be larger than 90%. Note that when N is increased by 3 orders of magnitude, the required number of measurements is increased by a factor of 3, which is an indication of the logarithmic dependence of M to N. Furthermore, as the signal becomes less sparse (i.e., k increases), the required oversampling factor decreases. For $k = 100$ this ratio is only about 5 for $N = 1024$ and about 25 for $N \approx 10^{12}$. This is significantly better than existing sublinear recovery algorithms. Note that the optimal value of d for constructing the measurement matrices for every k, N is found empirically.

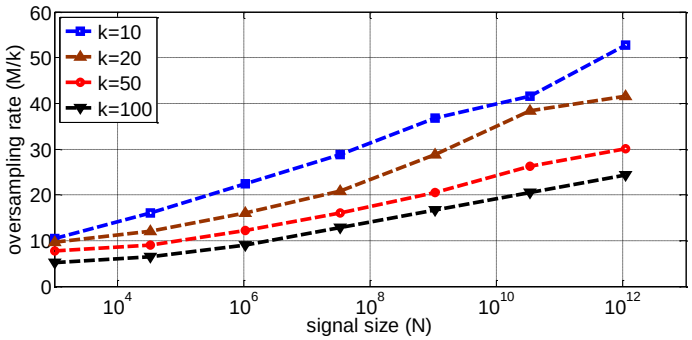


Figure 4.6: Required oversampling rate for successful recovery of Algorithm 2 on proposed constructions versus signal dimension for various sparsity levels.

In Table 4.2, we have compared the oversampling factor of the SIR algorithm with the Chaining Pursuit method and the chirp reconstruction method of [HCS08] for a few cases. Note that in most cases, our method is notably advantageous.

In Figure 4.7, the probability of successful recovery is plotted against the sparsity level k for $N = 32768$ and $M = 192, 240, 320, 448$. We can see that although the number of measurements has only increased by a factor of 2.3, the recoverable sparsity (given a fixed probability of success) has improved by almost a factor of 6 in some cases. Also, numerical evaluations reveal that these curves are comparable with the performance of ℓ_1 minimization over dense matrices (i.i.d. Gaussian), with the same

Table 4.2: Comparison of empirical minimum required over-sampling ratios for 90% recoverability guarantee for SIR, Reed-Muller (RM) decoding of [HCS08], and the Chaining Pursuit (CP) method of [GSTV06]

k	N	M/k (SIR)	M/k (RM)	M/k (CP)
10	2^{20}	22.4	25.6	420
10	2^{25}	28.8	51.2	520
50	2^{20}	12.16	10.24	> 850
50	2^{25}	16	40.96	> 1000

number of measurements but with $N = 3000$, which is an indication of the strong performance of the proposed scheme.

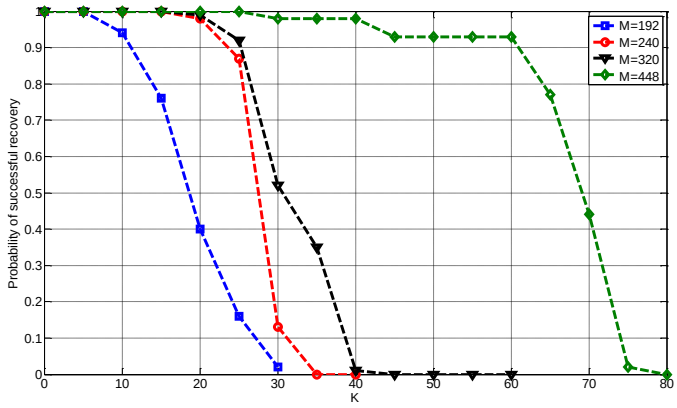


Figure 4.7: Probability of successful recovery of Algorithm 2 versus sparsity level k, for $N = 32768$ and $M = 192, 240, 320, 448$

Figure 4.8 shows a comparison the SIR algorithm with the chirp reconstruction method of [HCS08] for Reed-Muller-based matrices. For fixed values of M, N, we have compared successful recovery percentage and the running time of the algorithms for different sparsity levels. Note that in this case our proposed method has both higher recovery percentage and significantly faster execution time.

Finally, we also evaluate the performance of the SIR algorithm in the presence of noise. In this case, the observation vector $\mathbf{y}$ is assumed to be:

$$\mathbf{y} = \Phi\mathbf{x} + \mathbf{v}, \tag{4.9.1}$$

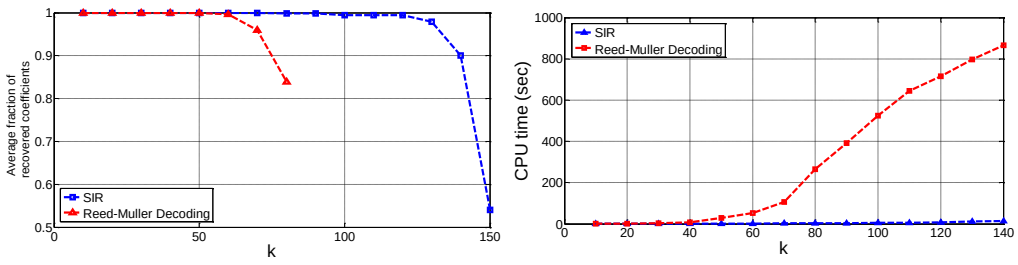


Figure 4.8: Comparison of the performance (a) and average running time (b) for the Reed-Muller decoding of [HCS08] and SIR (this thesis) for $N = 1048576$ and $M = 1024$. The SIR is based on a random $(20, 7)$ summary codebook.

where $\mathbf{v}$ is the noise vector with i.i.d. entries drawn from $\mathcal{N}(0, \sigma_v^2)$. Also, we assume that the nonzero entries of $\mathbf{x}$ are i.i.d. Gaussian with zero mean and a fixed variance σ_x^2. The average signal-to-noise ratio is therefore defined as:

$$\text{SNR} = \mathbb{E}10 \log_{10} \frac{x^T \mathbf{x}}{\mathbf{v}^T \mathbf{v}} = 20 \log_{10} \frac{k\sigma_x}{M\sigma_v}. \tag{4.9.2}$$

We also define the signal-to-error ratio for the approximate solution $\hat{\mathbf{x}}$ to the sparse vector $\mathbf{x}$ as follows:

$$\text{SER} = 20 \log_{10} \frac{\|\mathbf{x}\|_2}{\|\hat{\mathbf{x}} - \mathbf{x}\|_2}. \tag{4.9.3}$$

Given these definitions, we plot the average SER against the average SNR for three sublinear reconstruction algorithms: The SIR algorithm proposed in the current paper, the Reed-Muller decoding of [HCS08], and the Chaining Pursuit method. The results are shown in Figure 4.9. Note that despite being significantly faster, the performance curve of our proposed method is comparable to the Reed-Muller decoding algorithm (about 3dB less) and significantly better than Chaining Pursuit.

4.10 Proof of Theorems

Proof of Theorem 4.8.2

Let $k \leq f_s(n, d-1)$ and let $\mathbf{x}_0$ be a nonnegative k-sparse vector. Also, let the support set of $\mathbf{x}_0$ be the indices represented by the n-bit binary labels $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_k$. We show

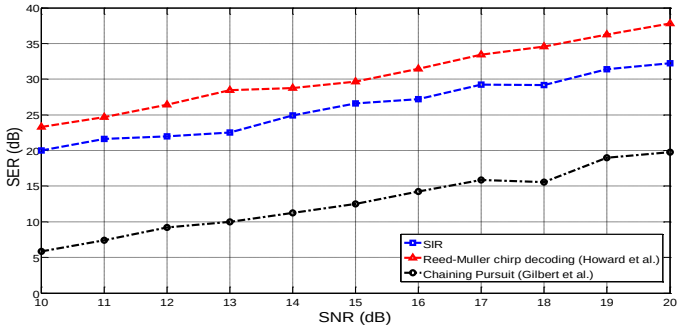


Figure 4.9: Average signal-to-recovery-error ratio (SER) versus average signal-to-noise-ratio (SNR) for different sublinear algorithms. The parameters are $N = 2^{20}$ and $M = 1024$ for the Reed-Muller and SIR algorithms, and $N = 2^{20}$ and $M = 3780$ for the Chaining Pursuit method. The sparsity is $k = 10$.

that if Φ corresponds to a complete (n, d) summary codebook, then $\mathbf{x}_0$ is the unique nonnegative solution to $\Phi\mathbf{x} = \Phi\mathbf{x}_0$. Therefore, from Lemma 4.8.1, it follows that $\mathbf{x}_0$ can be recovered via (4.7.1). We prove this by contradiction. Suppose that there is another nonnegative vector $\mathbf{x} \neq \mathbf{x}_0$ with $\Phi\mathbf{x} = \Phi\mathbf{x}_0$. Due to the nonnegativity assumption, we may assume that the support sets of $\mathbf{x}$ and $\mathbf{x}_0$ do not overlap. Let the n-bit labels of the support set of $\mathbf{x}$ be the binary sequences $\mathbf{b}'_1, \mathbf{b}'_2, \ldots, \mathbf{b}'_\ell$, with $1 \leq \ell \leq N$. From the definition of $f_s(\cdot)$, it follows that there is a $(n, d-1)$ summary that appears in exactly one of the sequences $\mathbf{b}_1, \ldots, \mathbf{b}_k$. Let us assume without the loss of generality that the first $d-1$ bits of $\mathbf{b}_1$ are unique, and that $\mathbf{b}_1$ is the all-zero binary sequence. Therefore, there are at least $n - d + 1$ coefficients in $\mathbf{y} = \Phi\mathbf{x}_0$ that are equal to the entry of $\mathbf{x}_0$ indexed by $\Delta(\mathbf{b}_1) + 1$. These measurements are those that correspond to the summaries

$$(\{1, 2, \ldots, d-1, i\}, \mathbf{0}_{1\times d}), \ d \leq i \leq n. \tag{4.10.1}$$

Since $\Phi\mathbf{x} = \Phi\mathbf{x}_0$, there must be nonzero entries in $\mathbf{x}$ with labeling indices that satisfy the above summaries. In particular, without loss of generality assume that the first d bits of some label $\mathbf{b}'_1$ are all zero. On the other hand, since the support sets of $\mathbf{x}$ and $\mathbf{x}_0$ do not overlap, $\mathbf{b}'_1$ is different from $\mathbf{b}_1$ in at least one bit, say $\mathbf{b}_1(j) \neq \mathbf{b}'_1(j)$ for some $j > d$. Now consider the summary $(S, \mathbf{c}) = (\{1, 2, \ldots, d-1, j\}, 00\ldots01)$,

which conforms to all binary sequences that are zero in the first $d-1$ bits, and one on the jth bit. Because Φ corresponds to a complete (n, d) codebook, there is a row of Φ that corresponds to $(S, \mathbf{c})$, and for which the corresponding coefficient in $\mathbf{y}$ is nonzero, because $\mathbf{b}'_1$ conforms to $(S, \mathbf{c})$. On the other hand this cannot be true when considering the equations $\mathbf{y} = \Phi\mathbf{x}$, since it entails one of the labels $\mathbf{b}_1, \ldots, \mathbf{b}_k$ to conform to $(S, \mathbf{c})$, which cannot be $\mathbf{b}_1$ (recall that $\mathbf{b}_1$ is the all-zero codeword, whereas $\mathbf{c}$ includes a 1). The existence of such a label contradicts the assumption that $\mathbf{b}_1$ is the only label whose $d-1$ first bits are all zero.

Proof of Theorem 4.8.3

Let $k \leq f_s(n, d-1)$ and let $\mathbf{x}$ be a k-sparse vector. Also, let the n-bit binary labels associated to the support set of $\mathbf{x}$ be $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_k$. We show that at least one of these labels can be inferred from one of the nonzero values of the vector $\mathbf{y} = \Phi\mathbf{x}$, by solving (4.7.2). From the definition, there is a $(n, d-1)$ summary that appears in exactly one of the labels $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_k$. Without loss of generality, let's assume that the first $d-1$ bits of $\mathbf{b}_1$ are unique, and that $\mathbf{b}_1$ is the all-zero binary sequence. Also, let the nonzero value of $\mathbf{x}$ in the position given by $\mathbf{b}_1$ be γ. Now consider all summaries $(S, \mathbf{c})$ for which the value of the corresponding entry in $\mathbf{y}$ is equal to γ. Let these summaries be denoted by $\{(S_i, \mathbf{c}_i)\}_{i=1}^{t}$, where t is the number of occurrences of γ in $\mathbf{y}$. We show that there is a unique binary sequence $\mathbf{b}'$ that conforms to all of these summaries. In other words, we prove that equation (4.7.2) has a unique solution which is equal to $\mathbf{b}' = \mathbf{b}_1$.

Due to the distinguishability assumption on the nonzero values of $\mathbf{x}$, the set $\{(S_i, \mathbf{c}_i)\}_{i=1}^{t}$ should include the following summaries:

$$(\{1, 2, \ldots, d-1, i\}, \mathbf{0}), \ d \leq i \leq n. \tag{4.10.2}$$

Where $\mathbf{0}$ indicates the all-zero bit sequence of length d. Clearly the only length-n binary sequence that conforms to all of the above summaries is the all-zero binary

sequence, namely $\mathbf{b}_1$. Thus, we only need to show that $\mathbf{b}_1(S_i) = \mathbf{c}_i$ for all other summaries $(S_i, \mathbf{c}_i),\ 1 \leq i \leq t$. This also follows immediately from the distinguishability assumption on $\mathbf{x}$, and the fact that every instance of γ in the vector $\mathbf{y}$ is only the result of the nonzero value in $\mathbf{x}$ labeled by $\mathbf{b}_1$ (i.e., it is not the direct sum of another subset of the entries of $\mathbf{x}$).

Proof of Theorem 4.8.4

We define an event $\mathcal{E}$ which is stronger than the success event of Algorithm 2, and thus provides a sufficient condition for the success of SIR. Let the n-bit binary labels associated to the support set of $\mathbf{x}$ be $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_k$, and let $\mathcal{C}$ be the (n, m, d) summary codebook corresponding to Φ. We define $\mathcal{E}$ to be the event that for every $1 \leq i \leq k$, and every bit $1 \leq j \leq n$, there exists a summary $(S, \mathbf{c})$ in $\mathcal{C}$ such that $j \in S$, $\mathbf{b}_i(S) = \mathbf{c}$, and $\mathbf{b}_\ell \neq \mathbf{c}\ \forall \ell \neq i$. In other words, for each of the k labels corresponding to the support of $\mathbf{x}$ and each of the n bits, there is a summary in the codebook that includes the considered bit and only conforms to that particular label.

We find a lower bound on the probability of the complementary event $\mathcal{E}^c$ by using union bounds. Note that there are m distinct subsets in the summaries of the codebook $\mathcal{C}$, which are chosen randomly. We assume that the subsets are chosen independently at random, and allow repetition. In case of repetition, the repeated subset is excluded, which is only a worst case and does not hurt the analysis. Consider a label $\mathbf{b}_1$. The probability that a randomly chosen length d subset of bits includes the first bit is $\frac{d}{n}$. Furthermore, assume that at least a fraction ϵ' of the summaries that conform to $\mathbf{b}_1$ and include the first bit, does not conform to the remaining $\mathbf{b}_i$s (i.e., only appear in $\mathbf{b}_1$). Then, when a random subset S is chosen, with probability at least $\frac{\epsilon' d}{n}$, the following happens:

$$1 \in S \text{ and } \mathbf{b}_i(S) \neq \mathbf{b}_1(S)\ \forall 1 < i \leq k. \tag{4.10.3}$$

Therefore, the probability that the above event does not happen for any of m randomly chosen subsets S is at most $(1 - \frac{\epsilon' d}{n})^m$. From the definition of $f'_w(\cdot)$ and the fact that $k \leq f'_w(n, d, p, \epsilon)$, we know that with probability at least p, $\epsilon' \geq \epsilon$, and therefore, the probability that (4.10.3) does not happen for any set S in the codebook $\mathcal{C}$ is at most $1 - p + p\left(1 - \frac{\epsilon d}{n}\right)^m$. If we union bound the probability of such event for all possible k labels and all possible n bits, we conclude that the probability of the undesirable event $\mathcal{E}^c$ is bounded by:

$$\mathbb{P}(\mathcal{E}^c) \leq nk \left(1 - p + p\left(1 - \frac{\epsilon d}{n}\right)^m\right). \tag{4.10.4}$$

Which concludes the proof of the theorem.

Proof of Theorem 4.8.5

Let the n-bit binary labels associated to the support set of $\mathbf{x}$ be $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_k$, and let $\mathcal{C}_1$ be the (n, m, d) summary codebook corresponding to Φ_1. It can be shown that the value identification subroutine of Algorithm 3 identifies all nonzero values of the nonnegative vector $\mathbf{x}$ correctly, if in the observation vector $\mathbf{y}$, every nonzero value of $\mathbf{x}$ appears at least once. We find the probability that this condition holds when the m subsets of the random coodbook $\mathcal{C}_1$ are chosen at random. For every $1 \leq i \leq k$, we define the following set of subsets of $\{1, 2, \ldots, n\}$:

$$\mathcal{U}_i = \{S \mid |S| = d,\ \mathbf{b}_j(S) \neq \mathbf{b}_i(S)\ \forall j \neq i\}. \tag{4.10.5}$$

If a subset S in the codebook $\mathcal{C}_1$ belongs to $\mathcal{U}_i$, then the nonzero entry γ_i that corresponds to the label $\mathbf{b}_i$ appears in the observation vector $\mathbf{y}$. Therefore, we are interested in finding the probability that the set of m subsets of $\mathcal{C}_1$ has a nonempty overlap with all $\mathcal{U}_i$s. Let us assume that for some $\epsilon' > 0$, the following holds:

$$|\mathcal{U}_i| \geq \epsilon' \binom{n}{d},\ \forall 1 \leq i \leq k. \tag{4.10.6}$$

When a subset S is chosen at random, the probability that it belongs to $\mathcal{U}_i$ is at least ϵ'. Therefore the probability that $\mathcal{U}_i$ does not overlap with the set of all subsets S

appearing in $\mathcal{C}_1$ is at most $(1-\epsilon')^m$. Using a union bound over all $1 \leq i \leq k$, we conclude that the probability that this undesirable event happens for at least one of the sets $\mathcal{U}_i$ is at most $k(1-\epsilon')^m$, which means that the probability of success is at least $1-k(1-\epsilon')^m$. However, we know from the definition of $f_w(\cdot)$, and the fact that $k \leq f_w(n,d,p,\epsilon)$, that with probability at least p, we have $\epsilon' > \epsilon$. Therefore the overall probability of success is at least:

$$1-p+p(1-k(1-\epsilon)^m) \geq p(1-k(1-\epsilon)^m). \tag{4.10.7}$$

Proof of Lemma 4.8.6

For the first statement, we use induction to prove that $f_s(n,d) \geq 2f_s(n-1,d-1)$. Let $k = f_s(n,d)$ and suppose that $\mathbf{b}_1, \ldots, \mathbf{b}_k$ are distinct binary sequences of length n, so that no (n,d) summary uniquely appears in exactly one of the sequences. Without loss of generality, assume that the sequences $\mathbf{b}_1, \ldots, \mathbf{b}_t$ start with 0 (i.e., the leftmost bit is zero), and $\mathbf{b}_{t+1}, \ldots, \mathbf{b}_k$ start with 1. we can assume that $1 < t < k$, since otherwise, we simply take a bit that has both 0 and 1 values in $\mathbf{b}_i$s and consider that bit instead. Also, let $\mathbf{b}'_i$ be the $n-1$ length sequence that is obtained from $\mathbf{b}_i$ by removing the first bit. Knowing that $\mathbf{b}_i$s are unique, we conclude that the set of sequences $\mathbf{b}'_i,\ 1 \leq i \leq t$ are distinct, and so are $\mathbf{b}'_j,\ t+1 \leq j \leq k$. Furthermore, no $(n-1,d-1)$ summary $(S', \mathbf{c}')$ appears in exactly one $\mathbf{b}'_i$, for $1 \leq i \leq t$, since otherwise, the (n,d) summary $(S' \cup \{1\}, \mathbf{0c})$ shows up only in $\mathbf{b}_i$, which contradicts the earlier assumption. The same is true for the sequences $\mathbf{b}_j,\ t+1 \leq j \leq k$. Therefore, from the definition, we can conclude that $t \geq f_s(n-1,d-1)$ and $k-t \geq f_s(n-1,d-1)$, which concludes that $k \geq 2f_s(n-1,d-1)$, and proves the main step of the induction. It is easy to see that for $l=1,\ f(n,1)=2$, which is the base of the induction. This completes the proof of the first statement.

Now we prove the second statement. Let $\mathbf{b}_1, \ldots, \mathbf{b}_k$ be k independent random

binary sequences of length n. From Chernoff bound, we can say that with probability at least $1 - e^{-\alpha n}$, $\mathbf{b}_2$ has at most $\frac{n}{2}(1 + \sqrt{2\alpha})$ bits in common with $\mathbf{b}_1$. We can therefore use a union bound to conclude that with probability at least $1 - ke^{-\alpha n}$, every $\mathbf{b}_i,\ i > 1$ has less than $\frac{n}{2}(1 + \sqrt{2\alpha})$ bits in common with $\mathbf{b}_1$. Therefore, every $\mathbf{b}_i$ can result in at most $\binom{\frac{n}{2}(1+\sqrt{2\alpha})}{d}$ of the (n, d) summaries of $\mathbf{b}_1$ not being unique. Therefore, with probability $1 - ke^{-\alpha n}$, a fraction $\epsilon = 1 - \binom{\frac{n}{2}(1+\sqrt{2\alpha})}{d} / \binom{n}{d}$ of the (n, d) summaries of $\mathbf{b}_1$ are unique (i.e., are not present in any $\mathbf{b}_i,\ i > 1$). Using another union bound, we conclude that the same condition holds simultaneously for all of $\mathbf{b}_i$s with probability at least $1 - k^2 e^{-\alpha n}$, therefore proving the second statement of Lemma 4.8.6.

The proof of the third statement is straightforward from the definition of f_w and f'_w. Every (n, d) summary that includes the first bit includes a $(n-1, d-1)$ summary over the remaining $n - 1$ bits, and if this reduced summary is unique in $\mathbf{b}_1$, so is the original (n, d) summary.

4.11 Conclusion

We proposed very structured binary matrices for sparse estimation, along with a fast and robust recovery algorithm that has a running time only logarithmic in the dimension of the unknown signal. The method allows reconstruction of sparse signal of very large ambient dimensions in efficient time, and is thus useful for statistical inference problems with very large state spaces or when real time decoding is needed.

Part II

Threshold Improvement in Basis Pursuit

Chapter 5

Compressed Sensing with Prior Information

The previous three chapters were focused on the design of combinatorial/deterministic matrices for sparse recovery, with the motivation that they often result in several advantages over random (e.g., Gaussian) ensembles. In particular, we discussed combinatorial algorithms that can be faster or easier to implement that geometrical approaches such as Basis Pursuit. In this part however, we solely focus on the Basis Pursuit algorithm, namely ℓ_1 minimization and its variations, and study them more closely for i.i.d. Gaussian matrices. The linear programming reconstruction methods have been extensively studied in the past couple of years following the breakthrough results of Donoho and Tanner, and Candes and Tao. In particular, a well-known property of this type of recovery method is the existence of a phase transition in the asymptotic proportional region of system dimensions in which the ratios $\frac{k}{n}$ and $\frac{m}{n}$ are constant. Recall that n,m, and k represent the signal dimension, number of measurements, and the sparsity of the unknown signal, respectively. Despite the reckless choice of the Gaussian measurement ensembles, and the seemingly limited choice of the reconstruction method, empirical evidence reveals that the phase transition curve of ℓ_1 minimization is a very fundamental limit. Firstly, no alternative choice

of measurement matrix (whether deterministic or random) exhibits a better transition curve for ℓ_1 minimization than the Gaussian sketches. In addition, with random structures, there is no existing polynomial time algorithm with universally better bounds than ℓ_1 minimization. Not to mention that the universality of the results of ℓ_1 minimization (namely that the performance is independent of the unknown signal distribution) is another fundamental property. For all the above reasons, the phase transition curve of linear programming decoding has somewhat been established as a milestone in analyzing the performance of sparse information retrieval. As a result, there is ongoing research trying to identify these bounds more explicitly or generalize the results to other forms of sparse reconstruction problems using atomic norms (see, e.g., [RXH08, Sto10]).

The focus of this chapter and the next one is to analyze the phase transition curve of ℓ_1 minimization, and investigate the possibility of improving it. In particular, we first consider the performance linear programming decoding under the presence of additional information about the signal sparsity pattern, or as will be referred to, under nonuniform sparsity assumptions. This will be studied in this chapter, where we show that an appropriate adjustment to ℓ_1 norm can tune the linear program to locate less-sparse vectors under nonuniform sparsity constraints. In Chapter 6, we explore ways of directly improving ℓ_1 minimization in lieu of any additional information.

$\mathbf{A}$	measurement matrix
n	signal size
m	number of measurements
k	sparsity of the signal
δ	aspect ratio of $\mathbf{A}$, m/n
μ	sparsity ratio k/n
δ_c	weak threshold of regularized ℓ_1 minimization

5.1 Introduction

The conventional approach to compressed sensing assumes no prior information on the unknown signal other than the fact that it is sufficiently sparse over a particular basis. In many applications, however, additional prior information is available in one or other forms, such in natural or biomedical images, genomic data processing, network monitoring, machine vision, etc. In fact, in many cases the signal recovery problem that compressed sensing addresses is a detection or estimation problem in some statistical setting. Some recent work along these lines can be found in [DWB06], which considers compressed detection and estimation, [JXC08], which studies Bayesian compressed sensing, and [BCDH10] which introduces model-based compressed sensing allowing for model-based recovery algorithms. In a more general setting, compressed sensing may be the inner loop of a larger estimation problem that feeds prior information on the sparse signal (e.g., its sparsity pattern) to the compressed sensing algorithm [CWB08, KXAH10].

We consider a particular model for the sparse signal where the entries of the unknown vector fall into a number u of classes, with each class having a specific fraction of nonzero entries. The standard compressed sensing model is therefore a special case where there is only one class. As mentioned above, there are many situations where such prior information may be available, such as in natural images, medical imaging, or in DNA micro-arrays. In the DNA micro-arrays applications, for instance, signals are often *block sparse*, i.e., the signal is more likely to be nonzero in certain blocks than in others [SPH09]. While it is possible (albeit cumbersome) to study this model in full generality, for now, we focus on the case where the entries of the unknown signal fall into a fixed number u of categories; in the ith set K_i with cardinality n_i, the fraction of nonzero entries is p_i. This model is rich enough to capture many of the salient features regarding prior information. We refer to the signals generated based on this model as *nonuniform sparse* signals.

A signal generated based on this model could resemble the vector representation of a natural image in the domain of some linear transform (e.g. discrete Fourier transform, discrete cosine transform, discrete wavelet transform, ...) or the spatial representation of some biomedical image (e.g., a brain fMRI image; see [LV09] for an application of model-based compressed sensing and modified sparsity models for MRI images). Although a brain fMRI image is not necessarily sparse, the subtraction of the brain image at any moment during an experiment from an initial background image of inactive brain mode is indeed a sparse signal which demonstrates the additional brain activity during the specific course of experiment. Moreover, depending on the assigned task, the experimenter might have some prior information. For example it might be known that some regions of the brain are more likely to be entangled with the decision making process than the others. This can be captured in the above *nonuniform sparse* model by considering a higher value p_i for the more active region. Similarly, this model is applicable to other problems like network monitoring (see [CPR07] for an application of compressed sensing and nonlinear estimation in compressed network monitoring), DNA microarrays [MBSR07, ES05, VPMH07], astronomy, satellite imaging, and many more practical examples. In particular, we will provide one example in satellite imaging with details where such a model is applicable for real data. In general, achieving such a probabilistic prior in practice requires comprehensive knowledge of the model from which the sparse signal is generated, or detailed analysis of some post-processing. We will elaborate on the latter later in this chapter.

We first study this model for the case where there are $u \geq 2$ categories of entries, and demonstrate through rigorous analysis and simulations that the recovery performance using basis pursuit can be significantly boosted by exploiting the additional information. We then generalize the results to an arbitrary class of entries. A further interesting question to be addressed in future work would be to characterize the gain in recovery percentage as a function of the number of distinguishable classes u.

5.2 Related Work

A somewhat similar model for prior information has been considered in [VL10] under the title of modified compressed sensing, almost concurrently with our initial results. There it has been assumed that part of the support is completely known a priori or due to previous processing, and consequently the signal recovery is performed under a partially known support condition. A modification of the regular ℓ_1 minimization based on the given information is proven to achieve significantly better recovery guarantees, mostly in terms of tightening the reconstruction error bound and RIP coefficients. Explicit derivations of recovery threshold improvements are thus missing in [VL10]. A variation of the modified CS algorithm for noisy recovery under partially known support is also considered in [LV10]. As will be discussed, this model can be cast as a special case of the nonuniform sparse model, where the sparsity fraction is equal to unity in one of the classes. Therefore, using the generalized tools of this work, we can explicitly find the recovery thresholds (improvement) for the method proposed in [VL10]. Our method of analyzing weigh ℓ_1 minimization is based on null space characterization, which is in contrast to the recovery guarantees of [VL10] given in terms of the restricted isometry property (RIP).

Another related method that uses weighted ℓ_1 optimization is by Candès et al. [CWB08]. The main difference is that in [CWB08], no prior information is assumed about the unknown vector, and as a result the weights are not fixed and depend on a preliminary estimate of the signal obtained by an initial ℓ_1 minimization. The method is thus an iterative approach, where at each step the ℓ_1 optimization is re-weighted using the estimate of the signal obtained in the last minimization step. This approach is studied separately in the next section when we discuss a re-weighted Basis Pursuit algorithm.

There are also a collection of papers on model-based compressive sampling which are mostly focused on a Bayesian approach to the sparse reconstruction problem,

which in some cases significantly eases the incorporation of additional or prior information into the reconstruction analysis. This approach falls outside the main scope of the current thesis. More information can be found in [DWB06, JXC08, BCDH10], the references therein, and the extensions build upon those works.

5.3 Contributions

We propose a weighted ℓ_1 minimization approach for sparse recovery where the ℓ_1 norms of different classes (K_is) are assigned different weights w_{K_i} ($1 \leq i \leq u$). Intuitively, one would want to give a larger weight to the entries with a higher chance of being zero and thus further force them to be zero. The second contribution is that we *explicitly* compute the relationship between p_i, w_{K_i},$\frac{n_i}{n}$, $1 \leq i \leq u$, and the number of measurements so that the unknown signal can be recovered with overwhelming probability as $n \to \infty$ (the so-called "weak" and "strong" thresholds) for measurement matrices drawn from an i.i.d. Gaussian ensemble. The analysis uses the high-dimensional geometry techniques first introduced by Donoho and Tanner [Don06b, DT05a] (e.g., Grassmann angles) to obtain sharp thresholds for compressed sensing. However, rather than the *neighborliness* condition used in [Don06b, DT05a], we find it more convenient to use the null space characterization of Xu and Hassibi [XH08, SXH08]. The resulting Grassmannian manifold approach is a general framework for incorporating additional factors into compressed sensing: in [XH08] it was used to incorporate approximately sparse signals; here it is used to incorporate prior information and weighted ℓ_1 optimization. The techniques for computing the probability decay exponents are adapted from the works of [Don06b, DT05a]. Our analytic results allow us to precisely compute the optimal weights for any p_i,n_i, $1 \leq i \leq u$. We also provide certain robustness conditions for the recovery scheme for compressible signals or under model mismatch. We present simulation results to show the advantages of the weighted method over standard ℓ_1 minimization. Furthermore, our results for

the case of two classes ($u = 2$) builds a rigid framework for analyzing certain classes of re-weighted ℓ_1 minimization algorithms. In a re-weighted ℓ_1 minimization algorithm, the post-processing information from the estimate of the signal at each step can be viewed as additional prior information about the signal, and can be incorporated into the next step as appropriate weights. Building upon the current subject, we have been able to analytically prove the threshold improvement in a reweighted ℓ_1 minimization using this framework, as is reported in [HKXA09, KXAH10]. These contributions will be discussed in the next chapter. It is worth mentioning that we have prepared a software package based on the results of this chapter for threshold computation using weighted ℓ_1 minimization, and it is available in [Sof].

5.4 Preliminaries

First consider the following definition:

Definition 14. *A random variable Y is said to have a half-normal distribution $HN(0, \sigma^2)$ if $Y = |X|$ where X is a zero mean normal variable $X \sim \mathcal{N}(0, \sigma^2)$.*

5.4.1 Nonuniform Sparsity Model

We first define the signal model. For completeness, we present a general definition.

Definition 15. *Let $\mathcal{K} = \{K_1, K_2, ..., K_u\}$ be a partition of $\{1, 2, \cdots, n\}$, i.e. ($K_i \cap K_j = \emptyset$ for $i \neq j$, and $\bigcup_{i=1}^{u} K_i = \{1, 2, ..., n\}$), and $\mathbf{p} = \{p_1, p_2, \cdots, p_u\}$ be a vector of positive numbers in $[0, 1]$. A $n \times 1$ vector $\mathbf{x} = (x_1, x_2, \cdots, x_n)^T$ is said to be a "random nonuniformly sparse" vector with sparsity fraction p_i over the set K_i for $1 \leq i \leq u$, if $\mathbf{x}$ is generated from the following random procedure:*

- *Over each set K_i, $1 \leq i \leq u$, the set of nonzero entries of $\mathbf{x}$ is a random subset of size $p_i|K_i|$. In other words, a fraction p_i of the entries are nonzero in K_i. p_i is called the sparsity fraction over K_i. The values of the nonzero entries of*

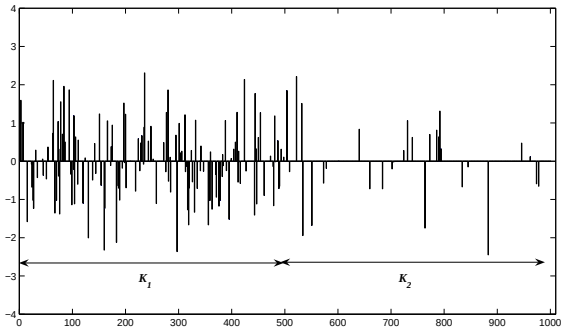


Figure 5.1: Illustration of a nonuniformly sparse signal

x *can arbitrarily be selected from any distribution. We can choose* $\mathcal{N}(0,1)$ *for simplicity.*

Example 5. In Figure 5.1, a sample nonuniformly sparse signal with Gaussian distribution for nonzero entries is plotted. The number of sets is considered to be $u = 2$ and both classes have the same size $\frac{n}{2}$, with $n = 1000$. The sparsity fraction for the first class K_1 is $p_1 = 0.3$, and for the second class K_2 is $p_2 = 0.05$. In fact, the signal is much sparser in the second half than it is in the first half.

This model is motivated by many practical applications in which the underlying unknown signal is both sparse and the energy content is non-uniformly distributed among different known classes of entries.

Example 6. The energy content of natural images follows some form of nonuniform sparsity. The DCT transform of natural images is of significance importance to signal processing, as it forms the domain for compression schemes such as JPEG. In fact, the DCT representation of a natural images is often sparse and concentrated on the low-frequency end of the spectrum. In Figure 5.2, a random natural image and its real DCT transform is shown, where this fact can be clearly verified.[1]

[1]Image downloaded from http://thetravelerpost.com/2011/08/persepolis-persian-capital-city-518b-c/

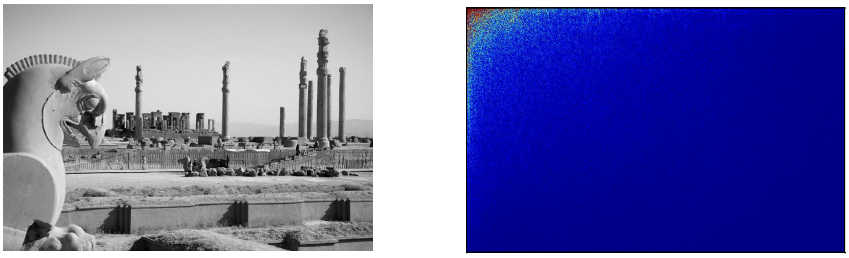


Figure 5.2: A natural image (left) and the amplitude of its two-dimensional DCT transform (right)

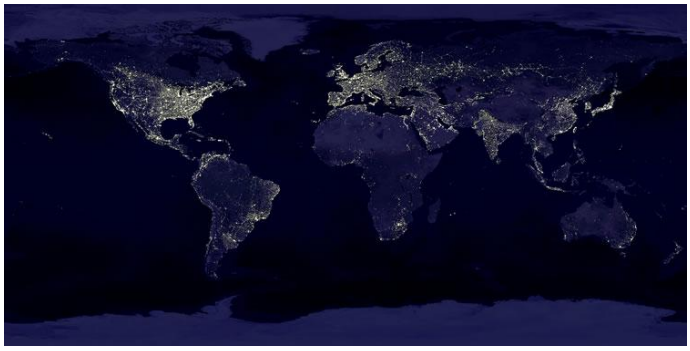


Figure 5.3: An overlay of satellite images showing the Earth at night

Example 7. Astronautical and satellite images are another class of signals with both sparsity and nonuniform distribution of energy content. In particular, we have depicted an overlay of satellite images of the Earth at night in Figure 5.3.[2] It would have been easy to guess that most of the significant brightness of the picture is concentrated on urban areas.

One of the advantageous features of the above model is that all the resulting computations are independent of the actual distribution on the amplitude of the nonzero entries. However, as expected, it is not independent of the properties of the measurement matrix. We assume that the measurement matrix $\mathbf{A}$ is a $m \times n$ matrix with i.i.d. standard Gaussian distributed $\mathcal{N}(0,1)$ entries, with $\frac{m}{n} = \delta < 1$. As in the previous chapters, the measurement vector $\mathbf{y}$ obeys the following:

$$\mathbf{y} = \mathbf{A}\mathbf{x}. \tag{5.4.1}$$

[2]Image downloaded from http://geology.com/articles/satellite-photo-earth-at-night.shtml

As we concentrate on variations of ℓ_1 minimization for recovery, it is useful to formally define some relevant concepts more formally in advance. As mentioned in previous chapters, ℓ_1 minimization can recover a randomly selected vector $\mathbf{x}$ with $k = \mu n$ nonzero entries with high probability, provided μ is less than a known function of δ. ℓ_1 minimization has the following form:

$$\min_{\mathbf{Ax}=\mathbf{y}} \|\mathbf{x}\|_1. \tag{5.4.2}$$

As previously discussed, the work of David Donoho [Don06b] provides an explicit relationship between μ and the minimum δ that guarantees success of ℓ_1 minimization recovery in the case of Gaussian measurements, and provides the corresponding numerical curve, generally known as the phase transition curves of ℓ_1 minimization. The optimization in (5.4.2) is a linear program and can be solved polynomially fast ($\mathcal{O}(n^3)$). However, it fails to encapsulate additional prior information of the signal nature, might there be any such information available. In the case of additional information about the signal model, one can simply think of modifying (5.4.2) to a weighted ℓ_1 minimization as follows:

$$\min_{\mathbf{Ax}=\mathbf{y}} \|\mathbf{x}\|_{\mathbf{w},1} = \min_{\mathbf{Ax}=\mathbf{y}} \sum_{i=1}^{n} w_i |x_i|. \tag{5.4.3}$$

The index, $\mathbf{w}$, on the norm is an indication of the $n \times 1$ positive weight vector. Now the questions are i) what is the optimal set of weights for a certain set of available prior information? and ii) can one improve the recovery threshold using the weighted ℓ_1 minimization of (6.5.2) by choosing a set of optimal weights? We have to be more clear with our objective at this point and clarify what we mean by improving the recovery threshold. Generally speaking, if a recovery method can reconstruct all signals of a certain model with certainty, then that method is said to be *strongly successful* on that signal model. If we have a class of models that can be identified with a parameter θ, and if for all models corresponding to $\theta < \theta_0$ a recovery scheme is strongly successful, then the threshold θ_0 is called a *strong recovery threshold* for the parameter θ. For

example, for fixed $\frac{m}{n}$, if $k < n$ is sufficiently small, then ℓ_1 minimization can provably recover all k-sparse signals, provided that appropriate linear measurements have been made from the signal. The maximum such k is called the strong recovery threshold of the sparsity for the success of ℓ_1 minimization. Likewise, for a fixed ratio $\mu = \frac{k}{n}$, the minimum ratio of measurements to ambient dimension $\frac{m}{n}$ for which ℓ_1 minimization always recovers k-sparse signals from the given m linear measurements is called the strong recovery threshold for the number of measurements for ℓ_1 minimization. In contrast, one can also look into the *weak recovery threshold*, defined as the threshold below which, with very high probability, a random vector generated from the model is recoverable. For the nonuniformly sparse model, the quantity of interest is the overall sparsity fraction of the model defined as $(\frac{\sum_{i=1}^{u} p_i n_i}{n})$. The question we ask is whether by adjusting w_is according to p_is one can extend the strong or weak recovery threshold for sparsity fraction to a value above the known threshold of ℓ_1 minimization. Equivalently, for given classes $K_1, \cdots, K_u$ and sparsity fractions p_is, how much can the strong or weak threshold be improved for the minimum number of required measurements, as opposed to the case of uniform sparsity with the same overall sparsity fraction.

5.5 Summary of Main Results

We state the two problems more formally using the notion of recovery thresholds that we defined in the previous section. We only consider the case of $u = 2$.

- **Problem 1.** Consider the random nonuniformly sparse model with two classes K_1, K_2 of cardinalities $n_1 = \gamma_1 n$ and $n_2 = \gamma_2 n$, respectively, and given sparsity fractions p_1 and p_2. Let $\mathbf{w}$ be a given weight vector. As $n \to \infty$, what is the weak (strong) recovery threshold for $\delta = \frac{m}{n}$ so that a randomly chosen vector (all vectors) $\mathbf{x}_0$ selected from the nonuniformly sparse model is successfully recovered by the weighted ℓ_1 minimization of (6.5.2) with high probability?

Upon solving Problem 1, one can exhaustively search for the weight vector $\mathbf{w}$ that results in the minimum recovery threshold for δ. This is what we recognize as the optimum set of weights. So the second problem can be stated as:

- **Problem 2.** Consider the random nonuniformly sparse model defined by classes K_1, K_2 of cardinalities n_1 and n_2, respectively, with $\gamma_1 = \frac{n_1}{n}$ and $\gamma_2 = \frac{n_2}{n}$, and given sparsity fractions p_1 and p_2. What is the optimum weight vector $\mathbf{w}$ in (6.5.2) that results in the minimum number of measurements for almost-sure recovery of signals generated from the given random nonuniformly sparse model?

- **Problem 3.** When optimal weights are used to reconstruct signals of a nonuniformly spare model, how much improvement (in the number of required measurements, say) is achieved compared with regular ℓ_1 minimization?

We have fully solved these problems. We first connect the misdetection event to the properties of the measurement matrix. For the non-weighted case, this has been considered in [SXH08] and is known as the null space property. We generalize this result to the case of weighted ℓ_1 minimization, and mention a necessary and sufficient condition for (6.5.2) to recover the original signal of interest. The theorem is as follows.

Theorem 5.5.1. *For all $n \times 1$ vectors $\mathbf{x}^*$ supported on the set $K \subseteq \{1, 2, ..., n\}$, $\mathbf{x}^*$ is the unique solution to the linear program $\min_{\mathbf{Ax}=\mathbf{y}} \sum_{i=1}^{n} w_i|x_i|$ with $\mathbf{y} = \mathbf{Ax}^*$, if and only if for every nonzero vector $\mathbf{z} = (z_1, z_2, \cdots, z_n)^T$ in the null space of $\mathbf{A}$, the following holds: $\sum_{i\in K} w_i|z_i| < \sum_{i\in \overline{K}} w_i|z_i|$.*

This theorem will be proved in Section 5.6. Note that if the null space condition given in the above theorem is weaker than the null space condition for regular ℓ_1 minimization, then it immediately implies that the weighted ℓ_1 minimization is better than ℓ_1 minimization. To clarify this, consider the following example: Suppose that a subset K' of the support set K is known a priori. If we assign weights $w_i = 0$ to $i \in K'$

and $w_i = 1$ to $i \in K$, then whenever the null space condition $\sum_{i\in K} |z_i| < \sum_{i\in \overline{K}} |z_i|$ holds for regular ℓ_1 minimization, the condition $\sum_{i\in K} w_i|z_i| < \sum_{i\in \overline{K}} w_i|z_i|$ also holds (since the left-hand size decreases and the right-hand size is untouched). Therefore the weighted ℓ_1 scheme is at least as good as ℓ_1 minimization (in the strong recovery sense).

As will be explained in Section 5.6.1, Theorem 5.5.1, along with known facts about the null space of random Gaussian matrices, helps us interpret the probability of recovery error in terms of a high-dimensional geometrical object called the complementary Grassmann angle; namely the probability that a uniformly chosen $(n-m)$-dimensional subspace $\mathcal{Z}$ shifted by a point $\mathbf{x}$ of unity weighted ℓ_1-norm, $\sum_{i=1}^n w_i x_i = 1$, intersects the weighted ℓ_1-ball $\mathcal{P}_{\mathbf{w}} = \{\mathbf{y} \in \mathbb{R}^n \mid \sum_{i=1}^n w_i|y_i| \leq 1\}$ nontrivially at some other point besides $\mathbf{x}$. The shifted subspace is denoted by $\mathcal{Z}+\mathbf{x}$. What we can take for granted without explicitly proving is that, due to the identical marginal distribution of the entries of $\mathbf{x}$ in each of the sets K_1 and K_2, the entries of the optimal weight vector take at most two (or in the general case u) distinct values, w_{K_1} and w_{K_2}, depending on their index. In other words

$$\forall i \in \{1, 2, \cdots, n\} \quad w_i = \begin{cases} w_{K_1} \ if \ i \in K_1 \\ w_{K_2} \ if \ i \in K_2 \end{cases} . \tag{5.5.1}$$

Leveraging on the existing techniques for computing the complementary Grassmann angle [San52, McM75], we will be able to state and prove the following theorem along the same lines, which upper bounds the probability that the weighted ℓ_1 minimization does not recover the signal. Please note that in the following theorem, the rigorous mathematical definitions to some of the terms (internal angle and external angle) is not presented, due to the extent of descriptions. They will, however, be defined rigorously later in the derivations of the main results in Section 5.6.

Theorem 5.5.2. *Let K_1 and K_2 be two disjoint subsets of $\{1, 2, \cdots, n\}$ such that $|K_1| = n_1, |K_2| = n_2$, and p_1 and p_2 be real numbers in $[0, 1]$. Also, let $k_1 = p_1 n_1$, $k_2 = p_2 n_2$, and E be the event that a random nonuniformly sparse vector $\mathbf{x}_0$ (Definition 15)*

with sparsity fractions p_1 and p_2 over the sets K_1 and K_2, respectively, is recovered via the weighted ℓ_1 minimization of (6.5.2) with $\mathbf{y} = \mathbf{A}\mathbf{x}_0$. Also, let E^c denote the complement event of E. Then

$$\mathbb{P}\{E^c\} \leq \sum_{Q_{\tilde{n}_1,\tilde{n}_2,\tilde{m}}} 2^{t_1+t_2+1} \binom{n_1-k_1}{t_1} \binom{n_2-k_2}{t_2} \beta(k_1,k_2|t_1,t_2)\zeta(t_1+k_1,t_2+k_2), \tag{5.5.2}$$

where

$$\tilde{n_1} = n_1 - k_1,\ \tilde{n_2} = n_2 - k_2,\ \tilde{m} = m - k_1 - k_2 + 1,$$
$$Q_{\tilde{n}_1,\tilde{n}_2,\tilde{m}} = \{t_1, t_2 \in \mathbb{Z} \mid 0 \leq t_1 \leq \tilde{n}_1,\ 0 \leq t_2 \leq \tilde{n_2},\ t_1 + t_2 > \tilde{m}\}.$$

$\beta(k_1,k_2|t_1,t_2)$ is the internal angle between a (k_1+k_2-1)-dimensional face $\mathcal{F}$ of the weighted ℓ_1-ball $\mathcal{P}_{\mathbf{w}} = \{y \in \mathbb{R}^n | \sum_{i=1}^n w_i|y_i| \leq 1\}$ with k_1 vertices supported on K_1 and k_2 vertices supported on K_2, and another $(k_1+k_2+t_1+t_2-1)$-dimensional face $\mathcal{G}$ that encompasses $\mathcal{F}$ and has t_1+k_1 vertices supported on K_1 and the remaining t_2+k_2 vertices supported on K_2. $\zeta(d_1,d_2)$ is the external angle between a face $\mathcal{G}$ supported on set L with $|L \cap K_1| = d_1$ and $|L \cap K_2| = d_2$ and the weighted ℓ_1-ball $\mathcal{P}_{\mathbf{w}}$. See Section 5.6.1 for the definitions of integral and external angles.

The proof of this theorem will be given in Section 5.6.2. We are interested in the regimes that make the above upper bound decay to zero as $n \to \infty$, which requires the cumulative exponent in (5.5.2) to be negative. We are able to calculate sharp upper bounds on the exponents of the terms in (5.5.2) by using large deviations of sums of normal and half-normal variables. More precisely, if we assume that the sum of the terms corresponding to particular indices t_1 and t_2 in (5.5.2) is denoted by $F(t_1,t_2)$, and define $\tau_1 = \frac{t_1}{n}$ and $\tau_2 = \frac{t_2}{n}$, then using the angle exponent method from [Don06b, DT05a], we are able to find and compute an exponent function $\psi_{tot}(\tau_1,\tau_2) = \psi_{com}(\tau_1,\tau_2) - \psi_{int}(\tau_1,\tau_2) - \psi_{ext}(\tau_1,\tau_2)$ so that $\frac{1}{n}\log F(t_1,t_2) \sim \psi_{tot}(\tau_1,\tau_2)$ as $n \to \infty$. The terms $\psi_{com}(\cdot,\cdot)$, $\psi_{int}(\cdot,\cdot)$, and $\psi_{ext}(\cdot,\cdot)$ are contributions to the cumulative

exponent ψ_{tot} by the so-called combinatorial, internal angle and external angle terms, respectively, existing in the upper bound (5.5.2). The derivations of these terms will be elaborated upon in Section 5.6.2. Consequently, we state a key theorem that is the implicit answer to Problem 1.

Theorem 5.5.3. *Let $\delta = \frac{m}{n}$ be the ratio of the number of measurements to the signal dimension, $\gamma_1 = \frac{n_1}{n}$ and $\gamma_2 = \frac{n_2}{n}$. For fixed values of γ_1, γ_2, p_1, p_2, $\omega = \frac{w_{K_2}}{w_{K_1}}$, define E to be the event that a random nonuniformly sparse vector $\mathbf{x}_0$ (Definition 15) with sparsity fractions p_1 and p_2 over the sets K_1 and K_2, respectively, with $|K_1| = \gamma_1 n$ and $|K_2| = \gamma_2 n$, is recovered via the weighted ℓ_1 minimization of (5.4.2) with $\mathbf{y} = \mathbf{A}\mathbf{x}_0$. There exists a critical threshold $\delta_c = \delta_c\left((\gamma_1, \gamma_2), (p_1, p_2), (1, \omega)\right)$[3] such that if $\delta = \frac{m}{n} \geq \delta_c$, then $\mathbb{P}\{E^c\}$ decays exponentially to zero as $n \to \infty$. Furthermore, δ_c is given by*

$$\delta_c = \min\{\delta \mid \psi_{com}(\tau_1, \tau_2) - \psi_{int}(\tau_1, \tau_2) - \psi_{ext}(\tau_1, \tau_2) < 0 \ \forall \ 0 \leq \tau_1 \leq \gamma_1(1-p_1), \\ 0 \leq \tau_2 \leq \gamma_2(1-p_2), \tau_1 + \tau_2 > \delta - \gamma_1 p_1 - \gamma_2 p_2\}$$

where ψ_{com}, ψ_{int}, and ψ_{ext} are obtained from the following expressions:
Define $g(x) = \frac{2}{\sqrt{\pi}} e^{-x^2}$, $G(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-y^2} dy$, and let $\varphi(.)$ and $\Phi(.)$ be the standard Gaussian pdf and cdf functions, respectively.

1. *(Combinatorial exponent)*

$$\psi_{com}(\tau_1, \tau_2) = \left(\gamma_1(1-p_1)H(\frac{\tau_1}{\gamma_1(1-p_1)}) + \gamma_2(1-p_2)H(\frac{\tau_2}{\gamma_2(1-p_2)}) + \tau_1 + \tau_2\right)\log 2 \tag{5.5.3}$$

   where $H(\cdot)$ is the entropy function defined by $H(x) = -x\log x - (1-x)\log(1-x)$.

2. *(External angle exponent) Define $c = (\tau_1 + \gamma_1 p_1) + \omega^2(\tau_2 + \gamma_2 p_2)$, $\alpha_1 = \gamma_1(1 - p_1) - \tau_1$ and $\alpha_2 = \gamma_2(1-p_2) - \tau_2$. Let x_0 be the unique solution to x of the*

[3] *$\delta_c(\cdot,\cdot,\cdot)$ has three arguments: a vector of the ratios of partition sizes, a vector of sparsity fractions, and a weight vector.*

following:

$$2c - \frac{g(x)\alpha_1}{xG(x)} - \frac{\omega g(\omega x)\alpha_2}{xG(\omega x)} = 0.$$

Then

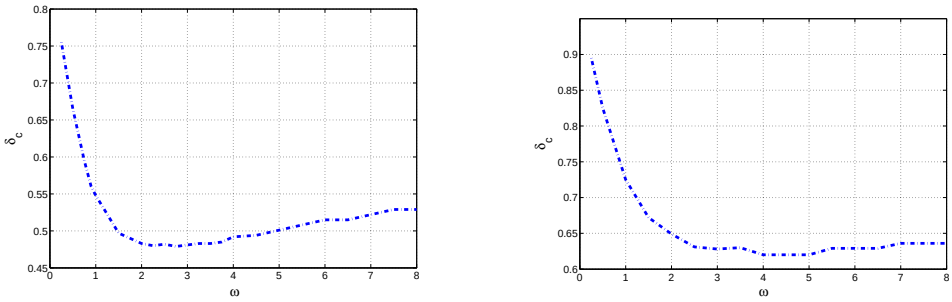
$$\psi_{ext}(\tau_1, \tau_2) = cx_0^2 - \alpha_1 \log G(x_0) - \alpha_2 \log G(\omega x_0). \tag{5.5.4}$$

3. *(Internal angle exponent) Let* $b = \frac{\tau_1+\omega^2\tau_2}{\tau_1+\tau_2}$, $\Omega' = \gamma_1 p_1 + \omega^2\gamma_2 p_2$, *and* $Q(s) = \frac{\tau_1\varphi(s)}{(\tau_1+\tau_2)\Phi(s)} + \frac{\omega\tau_2\varphi(\omega s)}{(\tau_1+\tau_2)\Phi(\omega s)}$. *Define the function* $\hat{M}(s) = -\frac{s}{Q(s)}$ *and solve for* s *in* $\hat{M}(s) = \frac{\tau_1+\tau_2}{(\tau_1+\tau_2)b+\Omega'}$. *Let the unique solution be* s^* *and set* $y = s^*(b - \frac{1}{\hat{M}(s^*)})$. *Compute the rate function* $\Lambda^*(y) = sy - \frac{\tau_1}{\tau_1+\tau_2}\Lambda_1(s) - \frac{\tau_2}{\tau_1+\tau_2}\Lambda_1(\omega s)$ *at the point* $s = s^*$, *where* $\Lambda_1(s) = \frac{s^2}{2} + \log(2\Phi(s))$. *The internal angle exponent is then given by:*

$$\psi_{int}(\tau_1, \tau_2) = (\Lambda^*(y) + \frac{\tau_1 + \tau_2}{2\Omega'}y^2 + \log 2)(\tau_1 + \tau_2). \tag{5.5.5}$$

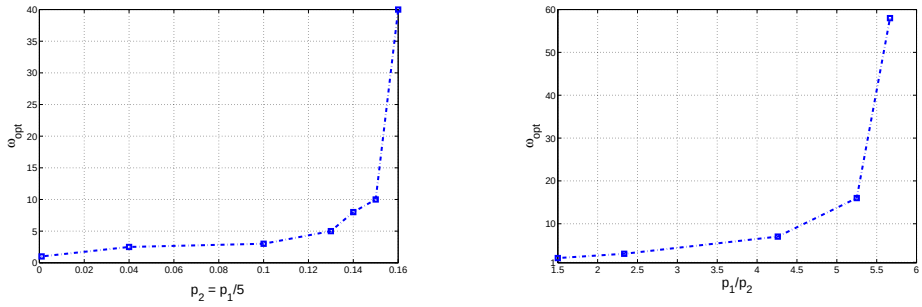
Theorem 5.5.3 is a powerful result, since it allows us to find (numerically) the optimal set of weights for which the fewest possible measurements are needed to recover the signals almost surely. To this end, for fixed values of γ_1, γ_2, p_1, and p_2, one should find the ratio $\frac{w_{K_2}}{w_{K_1}}$ for which the critical threshold $\delta_c\left((\gamma_1, \gamma_2), (p_1, p_2), (w_{K_1}, w_{K_2})\right)$ from Theorem 5.5.3 is minimum. We discuss this by some examples in Section 5.7. A generalization of Theorem 5.5.3 for a nonuniform model with an arbitrary number of classes ($u \geq 2$) will be given in Section 5.6.3.

As mentioned earlier, using Theorem 5.5.3, it is possible to find the optimal ratio $\frac{w_{K_2}}{w_{K_1}}$. It however requires an exhaustive search over the δ_c threshold for all possible values of ω. For $\gamma_1 = \gamma_2 = 0.5$, $p_1 = 0.3$, and $p_2 = 0.05$, we have numerically computed $\delta_c\left((\gamma_1, \gamma_2), (p_1, p_2), (w_{K_1}, w_{K_2})\right)$ as a function of $\frac{w_{K_2}}{w_{K_1}}$ and depicted the resulting curve in Figure 5.4a. This suggests that $\frac{w_{K_2}}{w_{K_1}} \approx 2.5$ is the optimal ratio that one can choose. Later we will confirm this using simulations. The value of δ_c for another choice of p_1, p_2 is shown in Figure 5.4b. Note that for given class sizes γ_1, γ_2 the optimal value of ω does not solely depend on $\frac{p_2}{P_1}$. To see this, we have obtained the numerical value of the optimal ω for a fixed ratio $\frac{p_1}{p_2} = 5$, for various values of p_2 using exhaustive



(a) $p_1 = 0.4$, $p_2 = 0.05$. (b) $p_1 = 0.65$, $p_2 = 0.1$.

Figure 5.4: δ_c as a function of $\omega = \frac{w_{K_2}}{w_{K_1}}$ for $\gamma_1 = \gamma_2 = 0.5$



(a) (b)

Figure 5.5: (a) Optimum value of weight $\omega = \frac{w_{K_2}}{w_{K_1}}$ vs. $p_2 = p_1/5$, $\gamma_1 = \gamma_2 = 0.5$. (b) Optimum value of weight $\omega = \frac{w_{K_2}}{w_{K_1}}$ vs. $\frac{p_1}{p_2}$, for $\gamma_1 = \gamma_2 = 0.5$, $p_1\gamma_1 + p_2\gamma_2 = 0.5$.

search and the result of Theorem 5.5.3. The result is displayed in Figure 5.5a. As can be seen, although $\frac{p_2}{P_1}$ is fixed, as p_1 approaches 1, the optimal value of weight becomes very large (in fact for the case of $p_1 = 1$, we later prove that the optimal ω is indeed $\omega_{\text{opt}} = \infty$). In Figure 5.5b, the optimal weight is plotted as a function of $\frac{p_1}{p_2}$ for a situation where the overall sparsity, i.e., $\gamma_1 p_1 + \gamma_2 p_2$, is constant.

Note that δ_c given in Theorem 5.5.3 is a weak bound on the ratio $\delta = \frac{m}{n}$. In other words, it determines the minimum number of measurements so that for a random sparse signal from the nonuniform sparse model and a random support set, the recovery is successful with high probability. It is possible to obtain a strong bound for δ, using a union bound on all possible support sets in the model, and all possible

sign patterns of the sparse vector. Similarly, a sectional bound can be defined which accounts for all possible support sets but almost all sign patterns. Therefore, the expressions for the strong and sectional thresholds (see [Don06b, DT05a] for definitions), which we denote by $\delta_c^{(S)}$ and $\delta_c^{(T)}$, are very similar to δ_c in Theorem 5.5.3, except for a slight modification in the combinatorial exponent term ψ_{com}. This will be elaborated upon in Section 5.6.2.

It is worthwhile to consider some asymptotic cases of the presented nonuniform model and some of their implications. First of all, when one of the subclasses is empty, e.g., $\gamma_1 = 0$, then the obtained weak and strong thresholds are equal to the corresponding thresholds of ℓ_1 minimization for a sparsity fraction $p = p_2$. Furthermore, if the sparsity fractions p_1 and p_2 over the two classes are equal, and a unitary weight $\omega = 1$ is used, then the weak threshold δ_c is equal to the threshold of ℓ_1 minimization for a sparsity fraction $p = p_1 = p_2$. In other words:

$$\delta_c\left((\gamma_1, \gamma_2), (p, p), (1, 1)\right) = \delta_c(1, p, 1), \tag{5.5.6}$$

where the right-hand side is simply the threshold of ℓ_1 minimization for sparsity fraction p. This follows immediately from the derivations of the exponents in Theorem 5.5.3. However, the latter is not necessarily true for the strong threshold. In fact the computation of the strong threshold for regular ℓ_1 minimization involves a union bound over a larger set of possible supports, and therefore the combinatorial exponent becomes larger. Therefore:

$$\delta_c^{(S)}\left((\gamma_1, \gamma_2), (p, p), (1, 1)\right) \leq \delta_c^{(S)}(1, p, 1). \tag{5.5.7}$$

A very important asymptotic case is when the unknown signal is fully dense over one of the subclasses, e.g., $p_1 = 1$, which accounts for a *partially known* support. This model is considered in the work of Vaswani et al. [VL10], with the motivation that in some applications (or due to previous processing steps), part of the support set can be fully identified ($p_1 = 1$), or can be approximated very well, which corresponds to

$p_1 < 1$. If the dense subclass is K_1 and $K_2 = K_1^c$, then [VL10] suggests solving the following minimization program:

$$\min_{\mathbf{Ax}=\mathbf{y}} \|\mathbf{x}_{K_2}\|_1. \tag{5.5.8}$$

It is possible to find exact thresholds for the above problem using the weighted ℓ_1 minimization machinery presented in this chapter. First, note that (5.5.8) is the asymptotic solution of the following weighted ℓ_1 minimization, when $\omega \to \infty$,

$$\min_{\mathbf{Ax}=\mathbf{y}} \|\mathbf{x}_{K_1}\|_1 + \omega\|\mathbf{x}_{K_2}\|_1. \tag{5.5.9}$$

Therefore the recovery threshold for (5.5.8) can be given by $\delta_c\left((\gamma_1, \gamma_2), (1, p_2), (1, \omega)\right)$ for $\omega \to \infty$. We prove the following theorem about the latter threshold:

Theorem 5.5.4. *If $\omega \to \infty$, then $\delta_c\left((\gamma_1, \gamma_2), (1, p_2), (1, \omega)\right) \to \gamma_1 + \gamma_2\delta_c(1, p_2, 1)$.*

The interpretation of the above theorem is that when a subset of entries of size $\gamma_1 n$ are known to be nonzero, the minimum number of measurements that is required for successful recovery with high probability using (5.5.8) is equal to the total number of measurements needed if we were allowed to independently measure everything in the first subclass (i.e., the known subset of the support), plus the number of measurements we needed for recovering the remaining entries using ℓ_1 minimization. The proof of this theorem is given in Section 5.8, which is based on exact computations of the threshold function δ_c. However, this result can be generalized to all nonuniform sparsity patterns and its proof does not require knowledge of the numerical values of δ_c and is based on a recent duality between phase transition thresholds in convex atomic norm minimization problems and the de-noising problem discussed in [DJM11, OKH12]. The generalization of the above theorem that we can prove is as follows:

Theorem 5.5.5. *Let ω^* be the optimal weight value of $\omega > 0$ that minimizes the recovery thresholds δ_c, namely:*

$$\omega^* = \mathit{argmin}_{\omega>0}\delta_c\left((\gamma_1, \gamma_2), (p_1, p_2), (1, \omega)\right).$$

Then:

$$\delta_c\left((\gamma_1,\gamma_2),(p_1,p_2),(1,\omega^*)\right) = \gamma_1\delta_c(1,p_1,1) + \gamma_2\delta_c(1,p_2,1). \tag{5.5.10}$$

More generally, for a nonuniform sparse model with class size ratios $\gamma = (\gamma_1, \gamma_2, \ldots, \gamma_u)$ *and sparsity fractions* $\mathbf{p} = (p_1, p_2, \ldots, p_u)$*, if* $\mathbf{w}^*$ *is the optimal weight vector for reconstruction using weighted* ℓ_1 *minimization, then:*

$$\delta_c(\gamma, \mathbf{p}, \mathbf{w}^*) = \sum_{i=1}^{u} \gamma_i \delta_c(1, p_i, 1). \tag{5.5.11}$$

This is a nice separation theorem with the following interpretations: If an optimal regularization (weighting) scheme is used, then for successful reconstruction, we would need as many measurements as if the different classes were compressed and reconstructed separately. In other words, in terms of the required number of measurements, this is equivalent to using a block diagonal compression matrix with i.i.d. Gaussian distribution for nonzero entries. In addition Theorem 5.5.5 allows us to answer problem 3 stated earlier, namely to quantify the improvement that we achieve by using an optimal regularization compared with standard ℓ_1 minimization. The answer is the "curvature" of the recovery threshold of ℓ_1 minimization. To see this, consider the recovery threshold of ℓ_1 minimization plotted in the form of $\delta = m/n$ versus the sparsity fraction $\mu = k/n$ in Figure 5.6. Suppose that a vector has a known nonuniform sparsity model $(p_1, p_2), (\gamma_1, \gamma_2)$. The improvement in the recovery threshold obtained by using the optimal weighted ℓ_1 scheme is equal to the difference between the thresholds of ℓ_1 minimization at the actual sparsity fraction $\gamma_1 p_1 + \gamma_2 p_2$, and the linear interpolation of the thresholds at the values p_1, p_2, as shown in Figure 5.6.

A very important factor regarding the performance of any recovery method is its robustness. In other words, it is important to understand how resilient the recovery is in the case of compressible signals or in the presence of noise or model mismatch (i.e., incorrect knowledge of the the sets or sparsity factors). We address this in the following theorem.

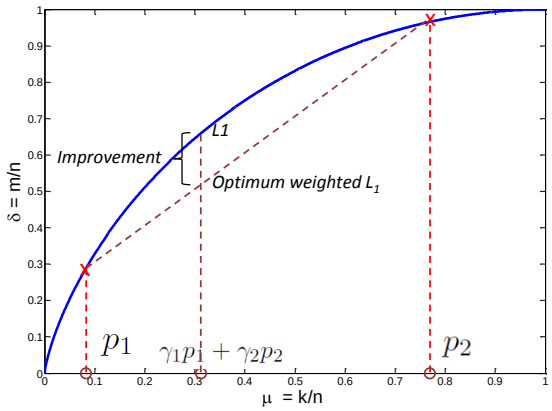


Figure 5.6: Illustration of the improvement in the recovery threshold of Basis Pursuit for nonuniformly sparse models using the optimal regularization

Theorem 5.5.6. *Let K_1 and K_2 be two disjoint subsets of $\{1, 2, \cdots, n\}$, with $|K_1| = \gamma_1 n, |K_2| = \gamma_2 n$ and $\gamma_1 + \gamma_2 = 1$. Also suppose that the dimensions of the measurement matrix $\mathbf{A}$ satisfy $\delta = \frac{m}{n} \geq \delta_c^{(S)}(\gamma_1, \gamma_2, p_1, p_2, \omega)$ for positive real numbers p_1 and p_2 in $[0, 1]$ and $\omega > 0$. For positive ϵ_1, ϵ_2, assume that L_1 and L_2 are arbitrary subsets of K_1 and K_2 with cardinalities $(1-\epsilon_1)\gamma_1 p_1 n$ and $(1-\epsilon_2)\gamma_2 p_2 n$, respectively. With high probability for every vector $\mathbf{x}_0$, if $\hat{\mathbf{x}}$ is the solution to the following linear program:*

$$\min_{\mathbf{Ax}=\mathbf{Ax}_0} \|\mathbf{x}_{K_1}\|_1 + \omega \|\mathbf{x}_{K_2}\|_1, \tag{5.5.12}$$

then the following holds:

$$\|(\mathbf{x}_0 - \hat{\mathbf{x}})_{K_1}\|_1 + \omega \|(\mathbf{x}_0 - \hat{\mathbf{x}})_{K_2}\|_1 \leq C_{\epsilon_1,\epsilon_2} \left(\|(\mathbf{x}_0)_{\overline{L_1} \cap K_1}\|_1 + \omega \|(\mathbf{x}_0)_{\overline{L_2} \cap K_2}\|_1 \right), \tag{5.5.13}$$

where

$$C_{\epsilon_1,\epsilon_2} = \frac{1 + \min(\frac{\epsilon_1 p_1}{1-p_1}, \frac{\epsilon_2 p_2}{1-p_2})}{1 - \min(\frac{\epsilon_1 p_1}{1-p_1}, \frac{\epsilon_2 p_2}{1-p_2})}.$$

The above theorem has the following implications. First, if $\mathbf{x}_0$ is a (compressible) vector, such that its "significant" entries follow a nonuniform sparse model, then the recovery error of the corresponding weighted ℓ_1 minimization can be bounded in terms of the ℓ_1 norm of the "insignificant" part of $\mathbf{x}_0$ (i.e., the part where a negligible fraction of the energy of the signal is located or most entries have significantly small values, compared to the other part that has an overall large norm). Theorem 5.5.6 can

also be interpreted as the robustness of weighted ℓ_1 scheme to the model mismatch. If K_1, K_2, p_1, p_2 are the estimates of an actual nonuniform decomposition for $\mathbf{x}_0$ (based on which the minimum number of required measurements have been estimated), then the recovery error can be relatively small if the model estimation error is slight. Theorem 5.5.6 will be proved in Section 6.6.

5.6 Derivation of the Results

In this section we provide detailed proofs to the claims of Section 5.5. We mention that, for better readability of the current thesis, and also due to the similarity of some proof techniques to those of [Don06b, DT05a, XH08], we have skipped proofs of some of the claims or the details of derivation. Those details can be found in [KXAH11]. Let $\mathbf{x}_0$ be a random nonuniformly sparse signal with sparsity fractions p_1 and p_2 over the index subsets K_1 and K_2, respectively (Definition 15), and let $|K_1| = n_1$ and $|K_2| = n_2$. Also let K be the support of $\mathbf{x}$. Let E be the event that $\mathbf{x}$ is recovered exactly by (6.5.2), and E^c be its complimentary event. In order to bound the conditional error probability $\mathbb{P}\{E^c\}$ we adopt the idea of [SXH08] to interpret the failure recovery event (E^c) in terms of the null space of the measurement matrix $\mathbf{A}$. This is stated in Theorem 5.5.1, which we prove here.

Proof of Theorem 5.5.1.

Suppose the mentioned null space condition holds and define $\hat{\mathbf{x}} = argmin_{\mathbf{Ax=y}} \sum_{i=1}^{n} w_i|x_i|$. Let $\mathbf{W} = diag(w_1, w_2, \cdots, w_n)$. If $\hat{\mathbf{x}} \neq \mathbf{x}$, then by triangular inequality, we have:

$$\begin{aligned}
\|\mathbf{W}\hat{\mathbf{x}}\|_1 &= \|(\mathbf{W}\hat{\mathbf{x}})_K\|_1 + \|(\mathbf{W}\hat{\mathbf{x}})_{\overline{K}}\|_1 \\
&= \|(\mathbf{W}\mathbf{x}^* + \mathbf{W}\hat{\mathbf{x}} - \mathbf{W}\mathbf{x}^*)_K\|_1 + \|(\mathbf{W}\hat{\mathbf{x}})_{\overline{K}}\|_1, \\
&\geq \|(\mathbf{W}\mathbf{x}^*)_K\|_1 - \|(\mathbf{W}\hat{\mathbf{x}} - \mathbf{W}\mathbf{x}^*)_K\|_1 + \|(\mathbf{W}\hat{\mathbf{x}} - \mathbf{W}\mathbf{x}^*)_{\overline{K}}\|_1, \\
&> \|\mathbf{W}\mathbf{x}^*\|_1,
\end{aligned}$$

where the last inequality is a result of the fact that $\hat{\mathbf{x}} - \mathbf{x}^*$ is a nonzero vector in the null space of $\mathbf{A}$ and satisfies the mentioned null space condition. However, by assumption, if $\hat{\mathbf{x}} \neq \mathbf{x}^*$ then $\|\mathbf{W}\hat{\mathbf{x}}\|_1 \leq \|\mathbf{W}\mathbf{x}^*\|_1$. This is a contradiction, and hence we should have $\hat{\mathbf{x}} = \mathbf{x}^*$. Conversely, suppose there is some vector $\mathbf{z}$ in $\mathcal{N}(\mathbf{A})$ such that $\|(\mathbf{W}\mathbf{z})_K\|_1 \geq \|(\mathbf{W}\mathbf{z})_{\overline{K}}\|_1$. Taking define $\mathbf{x}^* = (\mathbf{z}_K \ 0)^T$ and $\hat{\mathbf{x}} = (0 \ \mathbf{z}_{\overline{K}})^T$ implies that $\mathbf{A}\mathbf{x}^* = \mathbf{A}\hat{\mathbf{x}}$ and $\|\mathbf{W}\mathbf{x}^*\|_1 \geq \|\mathbf{W}\hat{\mathbf{x}}\|_1$. Therefore, $\mathbf{x}^*$ cannot be recovered from the weighted ℓ_1 minimization. ■

From this point on, we follow closely the steps towards calculating the upper bound on the failure probability from [XH08], but with appropriate modifications. The key to our derivations is the following lemma, the proof of which is skipped and can be found in [KXAH11].

Lemma 5.6.1. *For a certain subset $K \subseteq \{1, 2, ..., n\}$ with $|K| = k$, the event that the null-space $\mathcal{N}(A)$ satisfies*

$$\sum_{i \in K} w_i |z_i| \leq \sum_{i \in \overline{K}} w_i |z_i|, \forall \mathbf{z} \in \mathcal{N}(A), \tag{5.6.1}$$

is equivalent to the event that for each $\mathbf{x}$ supported on the set K (or a subset of K)

$$\sum_{i \in K} w_i |x_i + z_i| + \sum_{i \in \overline{K}} w_i |z_i| \geq \sum_{i \in K} w_i |x_i|, \forall \mathbf{z} \in \mathcal{N}(A). \tag{5.6.2}$$

5.6.1 Upper Bound on the Failure Probability

Knowing Lemma 5.6.1, we are now in a position to derive the probability that condition (5.6.1) holds for a support set K with $|K| = k$, if we randomly choose an i.i.d. Gaussian matrix $\mathbf{A}$. In the case of a random i.i.d. Gaussian matrix, the distribution of null space of $\mathbf{A}$ is right-rotationally invariant, and sampling from this distribution is equivalent to uniformly sampling a random $(n - m)$-dimensional subspace $\mathcal{Z}$ from the Grassmann manifold $\mathrm{Gr}_{(n-m)}(n)$. The Grassmann manifold $\mathrm{Gr}_{(n-m)}(n)$ is defined

as the set of all $(n-m)$-dimensional subspaces of $\mathbb{R}^n$. We need to upper bound the complementary probability $P = \mathbb{P}\{E^c\}$, namely the probability that the (random) support set K of $\mathbf{x}$ (of random sign pattern) fails the null space condition (5.6.2). We denote the null space of $\mathbf{A}$ by $\mathcal{Z}$. Because $\mathcal{Z}$ is a linear space, for every vector $\mathbf{z} \in \mathcal{Z}$, $\alpha\mathbf{z}$ is also in $\mathcal{Z}$ for all $\alpha \in \mathbb{R}$. Therefore, if for a $\mathbf{z} \in \mathcal{Z}$ and $\mathbf{x}$ condition (5.6.2) fails, by a simple re-scaling of the vectors, we may assume without loss of generality that $\mathbf{x}$ lies on the surface of any convex ball that surrounds the origin. Therefore we restrict our attention to those vectors $\mathbf{x}$ from the weighted ℓ_1-sphere:

$$\{\mathbf{x} \in \mathbb{R}^n \mid \sum_{i=1}^{n} w_i|x_i| = 1\}$$

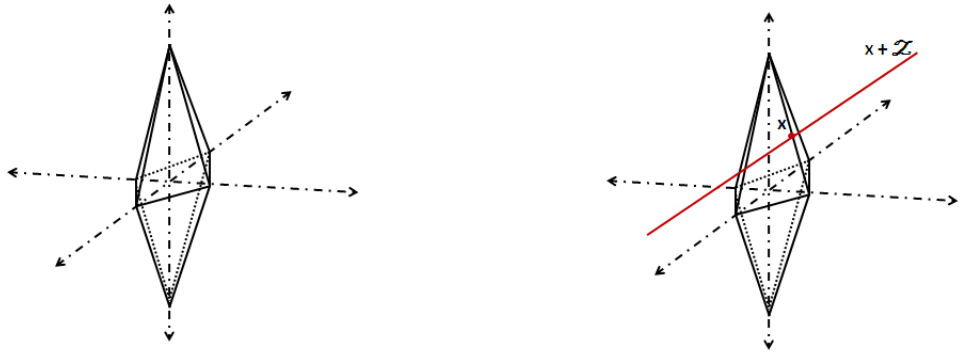
that are only supported on the set K, or a subset of it. Due to the similarity of the above weighted ℓ_1 balls in all orthants, we can write:

$$P = P_{K,-} \tag{5.6.3}$$

where $P_{K,-}$ is the probability that for a specific *support set* K, there exists a k-sparse vector $\mathbf{x}$ of a specific *sign pattern* which fails the condition (5.6.2). By symmetry, without loss of generality, we assume the signs of the elements of $\mathbf{x}$ to be non-positive. Now we can focus on deriving the probability $P_{K,-}$. Since $\mathbf{x}$ is a non-positive k-sparse vector supported on the set K and can be restricted to the weighted ℓ_1-sphere $\{\mathbf{x} \in \mathbb{R}^n \mid \sum_{i=1}^{n} w_i|x_i| = 1\}$, $\mathbf{x}$ is also on a $(k-1)$-dimensional face, denoted by $\mathcal{F}$, of the weighted ℓ_1-ball $\mathcal{P}_{\mathbf{w}}$:

$$\mathcal{P}_{\mathbf{w}} = \{\mathbf{y} \in \mathbb{R}^n \mid \sum_{i=1}^{n} w_i|y_i| \leq 1\}. \tag{5.6.4}$$

The subscript $\mathbf{w}$ in $\mathcal{P}_{\mathbf{w}}$ is an indication of the weight vector $\mathbf{w} = (w_1, w_2, \cdots, w_n)^T$. Figure 5.7a shows $\mathcal{P}_{\mathbf{w}}$ in $\mathbb{R}^3$ for some nontrivial weight vector $\mathbf{w}$. Now the probability $P_{K,-}$ is equal to the probability that there exists an $\mathbf{x} \in \mathcal{F}$, and there exists a $\mathbf{z} \in \mathcal{Z}$



(a) (b)

Figure 5.7: A weighted ℓ_1-ball, $\mathcal{P}_{\mathbf{w}}$, in $\mathbb{R}^3$ (a), and a linear hyperplane $\mathcal{Z}$ passing through a point $\mathbf{x}$ in the interior of a one-dimensional face of $\mathcal{P}_{\mathbf{w}}$ (b)

($\mathbf{z} \neq 0$) such that

$$\sum_{i \in K} w_i |x_i + z_i| + \sum_{i \in \bar{K}} w_i |z_i| \leq \sum_{i \in K} w_i |x_i| = 1. \tag{5.6.5}$$

We start by studying the case for a specific point $\mathbf{x} \in \mathcal{F}$ and, without loss of generality, we assume $\mathbf{x}$ is in the relative interior of this $(k-1)$-dimensional face $\mathcal{F}$. For this particular $\mathbf{x}$ on $\mathcal{F}$, the probability, denoted by $P'_{\mathbf{x}}$, that there exists a $\mathbf{z} \in \mathcal{Z}$ ($\mathbf{z} \neq 0$) such that

$$\sum_{i \in K} w_i |x_i + z_i| + \sum_{i \in \bar{K}} w_i |z_i| \leq \sum_{i \in K} w_i |x_i| = 1. \tag{5.6.6}$$

is essentially the probability that a uniformly chosen $(n-m)$-dimensional subspace $\mathcal{Z}$ shifted by the point $\mathbf{x}$, namely $(\mathcal{Z} + \mathbf{x})$, intersects the weighted ℓ_1-ball $\mathcal{P}_{\mathbf{w}}$ *non-trivially*, namely, at some other point besides $\mathbf{x}$ (Figure 5.7b). From the fact that $\mathcal{Z}$ is a *linear* subspace, the event that $(\mathcal{Z} + \mathbf{x})$ intersects $\mathcal{P}_{\mathbf{w}}$ is equivalent to the event that $\mathcal{Z}$ intersects nontrivially with the cone $\mathcal{C}_{\mathbf{w}}(\mathbf{x})$ obtained by observing the weighted ℓ_1-ball $\mathcal{P}_{\mathbf{w}}$ from the point $\mathbf{x}$. (Namely, $\mathcal{C}_{\mathbf{w}}(\mathbf{x})$ is conic hull of the point set $(\mathcal{P}_{\mathbf{w}} - \mathbf{x})$ and of course $\mathcal{C}_{\mathbf{w}}(\mathbf{x})$ has the origin of the coordinate system as its apex.) However, as noticed in the geometry for convex polytopes [Gru68, Gru03], the cones $\mathcal{C}_{\mathbf{w}}(\mathbf{x})$ are

identical for any $\mathbf{x}$ lying in the relative interior of the face $\mathcal{F}$. This means that the probability $P_{K,-}$ is equal to $P'_{\mathbf{x}}$, regardless of the fact that $\mathbf{x}$ is only a single point in the relative interior of the face $\mathcal{F}$. There are some singularities here because $\mathbf{x} \in \mathcal{F}$ may not be in the relative interior of $\mathcal{F}$, but it turns out that the $\mathcal{C}_{\mathbf{w}}(\mathbf{x})$ in this case is only a subset of the cone we get when $\mathbf{x}$ is in the relative interior of $\mathcal{F}$. So we do not lose anything if we restrict $\mathbf{x}$ to be in the relative interior of the face $\mathcal{F}$, namely we have

$$P_{K,-} = P'_{\mathbf{x}}.$$

Now we only need to determine $P'_{\mathbf{x}}$. From its definition, $P'_{\mathbf{x}}$ is exactly the *complementary Grassmann angle* [Gru68] for the face $\mathcal{F}$ with respect to the polytope $\mathcal{P}_{\mathbf{w}}$ under the Grassmann manifold $\mathrm{Gr}_{(n-m)}(n)$: a uniformly distributed $(n-m)$-dimensional subspace $\mathcal{Z}$ from the Grassmannian manifold $\mathrm{Gr}_{(n-m)}(n)$ intersecting non-trivially with the cone $\mathcal{C}_{\mathbf{w}}(\mathbf{x})$ formed by observing the weighted ℓ_1-ball $\mathcal{P}_{\mathbf{w}}$ from the relative interior point $\mathbf{x} \in \mathcal{F}$.

Building on the works by L.A. Santalö [San52] and P. McMullen [McM75] in high dimensional geometry and convex polytopes, the complementary Grassmann angle for the $(k-1)$-dimensional face $\mathcal{F}$ can be explicitly expressed as the sum of products of internal angles and external angles [AS92, McM75]:

$$2 \times \sum_{s \geq 0} \sum_{\mathcal{G} \in \Im_{m+1+2s}(\mathcal{P}_{\mathbf{w}})} \beta(\mathcal{F}, \mathcal{G}) \zeta(\mathcal{G}, \mathcal{P}_{\mathbf{w}}), \tag{5.6.7}$$

where s is any nonnegative integer, $\mathcal{G}$ is any $(m+1+2s)$-dimensional face of the $\mathcal{P}_{\mathbf{w}}$ ($\Im_{m+1+2s}(\mathcal{P}_{\mathbf{w}})$ is the set of all such faces), $\beta(\cdot,\cdot)$ stands for the internal angle and $\zeta(\cdot,\cdot)$ stands for the external angle, and are defined as follows [Gru03, McM75]:

- An internal angle $\beta(\mathcal{F}_1, \mathcal{F}_2)$ is the fraction of the hypersphere S covered by the cone obtained by observing the face $\mathcal{F}_2$ from the face $\mathcal{F}_1$.[4] The internal

[4]Note the dimension of the hypersphere S here matches the dimension of the corresponding cone discussed. Also, the center of the hypersphere is the apex of the corresponding cone. All these defaults also apply to the definition of the external angles.

angle $\beta(\mathcal{F}_1, \mathcal{F}_2)$ is defined to be zero when $\mathcal{F}_1 \nsubseteq \mathcal{F}_2$ and is defined to be one if $\mathcal{F}_1 = \mathcal{F}_2$.

- An external angle $\zeta(\mathcal{F}_3, \mathcal{F}_4)$ is the fraction of the hypersphere S covered by the cone of outward normals to the hyperplanes supporting the face $\mathcal{F}_4$ at the face $\mathcal{F}_3$. The external angle $\zeta(\mathcal{F}_3, \mathcal{F}_4)$ is defined to be zero when $\mathcal{F}_3 \nsubseteq \mathcal{F}_4$ and is defined to be one if $\mathcal{F}_3 = \mathcal{F}_4$.

The formula (5.6.7) comes from [McM75], based on the nonlinear angle sum of relations for polyhedral cones. In order to calculate the internal and external angles, it is important to use the symmetrical properties of the weighted cross-polytope $\mathcal{P}_{\mathbf{w}}$. First of all, $\mathcal{P}_{\mathbf{w}}$ is nothing but the convex hull of the following set of $2n$ vertices in $\mathbb{R}^n$

$$\mathcal{P}_{\mathbf{w}} = conv\{\pm\frac{\mathbf{e}_i}{w_i} \mid 1 \leq i \leq n\} \tag{5.6.8}$$

where $\mathbf{e}_i$ $1 \leq i \leq n$ is the standard unit vector in $\mathbb{R}^n$ with the ith entry equal to 1. Every $(k-1)$-dimensional face $\mathcal{F}$ of $\mathcal{P}_{\mathbf{w}}$ is simply the convex hull of k of the linearly independent vertices of $\mathcal{P}_{\mathbf{w}}$. In that case we say that $\mathcal{F}$ *is supported* on the index set K of the k indices corresponding to the nonzero coordinates of the vertices of $\mathcal{F}$ in $\mathbb{R}^n$. More precisely, if $\mathcal{F} = conv\{j_1\frac{\mathbf{e}_{i_1}}{w_{i_1}}, j_2\frac{\mathbf{e}_{i_2}}{w_{i_2}}, \cdots, j_k\frac{\mathbf{e}_{i_k}}{w_{i_k}}\}$ with $j_i \in \{-1, +1\}$ $\forall 1 \leq i \leq k$, then $\mathcal{F}$ is said to be supported on the set $K = \{i_1, i_2, \cdots, i_k\}$.

5.6.2 Special Case of Two Classes

The derivations of the previous section were for a general weight vector $\mathbf{w}$. We now restrict ourselves to the case of two classes, i.e., $u = 2$, namely K_1 and K_2 with $|K_1| = n_1$ and $|K_2| = n_2$. For this case, we may assume that w_is have the following particular form

$$\forall i \in \{1, 2, \cdots, n\} \quad w_i = \begin{cases} w_{K_1} \; if \; i \in K_1 \\ w_{K_2} \; if \; i \in K_2 \end{cases} . \tag{5.6.9}$$

Proof of Theorem 5.5.2. The choice of $\mathbf{w}$ as in (5.6.9) results in $\mathcal{P}_{\mathbf{w}}$ having two classes of geometrically identical vertices, and many of faces of $\mathcal{P}_{\mathbf{w}}$ being isomorphic. In fact,

two faces $\mathcal{F}$ and $\mathcal{F}'$ of $\mathcal{P}_{\mathbf{w}}$ that are, respectively, supported on the sets K and K', are geometrically isomorphic[5] if $|K \cap K_1| = |K' \cap K_1|$ and $|K \cap K_2| = |K' \cap K_2|$.[6] In other words, the only thing that distinguishes the morphology of the faces of $\mathcal{P}_{\mathbf{w}}$ is the proportion of their support sets that is located in K_1 or K_2. Therefore for two faces $\mathcal{F}$ and $\mathcal{G}$, with $\mathcal{F}$ supported on K and $\mathcal{G}$ supported on L ($K \subseteq L$), $\beta(\mathcal{F}, \mathcal{G})$ is only a function of the parameters $k_1 = |K \cap K_1|$, $k_2 = |K \cap K_2|$, $k_1 + t_1 = |L \cap K_1|$, and $k_2 + t_1 = |K \cap K_2|$. So, instead of $\beta(\mathcal{F}, \mathcal{G})$ we may write $\beta(k_1, k_2 | t_1, t_2)$ to indicate the internal angle between a $(k_1 + k_2 - 1)$-dimensional face $\mathcal{F}$ of $\mathcal{P}_{\mathbf{w}}$ with k_1 vertices supported on K_1 and k_2 vertices supported on K_2, and a $(k_1 + k_2 + t_1 + t_2 - 1)$-dimensional face $\mathcal{G}$ that encompasses $\mathcal{F}$ and has $t_1 + k_1$ vertices supported on K_1 and the remaining $t_2 + k_2$ vertices supported on K_2. Similarly, instead of $\zeta(\mathcal{G}, \mathcal{P}_{\mathbf{w}})$ we write $\zeta(t_1 + k_1, t_2 + k_2)$ to denote the external angle between a face $\mathcal{G}$ supported on set L with $|L \cap K_1| = d_1$ and $|L \cap K_2| = d_2$, and the weighted ℓ_1-ball $\mathcal{P}_{\mathbf{w}}$. Using this notation and recalling the formula (5.6.7) we can write:

$$\begin{aligned} P_{K,-} &= 2 \sum_{s \geq 0} \sum_{G \in \Im_{m+1+2s}(\mathcal{P}_{\mathbf{w}})} \beta(\mathcal{F}, \mathcal{G}) \zeta(\mathcal{G}, \mathcal{P}_{\mathbf{w}}) \\ &= \sum_{Q_e} 2^{t_1+t_2+1} \binom{n_1 - k_1}{t_1} \binom{n_2 - k_2}{t_2} \beta(k_1, k_2 | t_1, t_2) \zeta(t_1 + k_1, t_2 + k_2), \end{aligned} \tag{5.6.10}$$

where

$$Q_e = \{t_1, t_2 \in \mathbb{Z}^{\geq 0} \mid t_1 \leq n_1 - k_1,\ t_2 \leq n_2 - k_2,\ t_1 + t_2 + k_1 + k_2 - m \in \mathbb{E}\}.$$

where $\mathbb{E}$ is the set of positive even integers. Note that in (5.6.10) we have used the fact that the number of faces $\mathcal{G}$ of $\mathcal{P}_{\mathbf{w}}$ of dimension $k_1 + k_2 + t_1 + t_2 - 1$ that encompass $\mathcal{F}$ and have $k_1 + t_1$ vertices supported on K_1 and its remaining $k_2 + t_2$ are vertices

[5] This means that there exists a rotation matrix $\Theta \in \mathbb{R}^{n \times n}$ which is unitary (i.e., $\Theta^T \Theta = I$), and maps $\mathcal{F}$ isometrically to $\mathcal{F}'$ (i.e., $\mathcal{F}' = \Theta \mathcal{F}$).

[6] Remember that K_1 and K_2 are the same sets as defined in the model description of Section 5.4.1.

supported on K_2 is $2^{t_1+t_2}\binom{n_1-k_1}{t_1}\binom{n_2-k_2}{t_2}$. In fact $\mathcal{G}$ has $k_1+k_2+t_1+t_2$ vertices including the k_1+k_2 vertices of $\mathcal{F}$. The remaining t_1+t_2 vertices can each be independently in the positive or negative orthant, therefore resulting in the term $2^{t_1+t_2}$. The two other combinatorial terms are the number of ways one can choose t_1 vertices supported on the set K_1-K and t_2 vertices supported on K_2-K. From (5.6.10) and (5.6.3) we can conclude theorem 5.5.2. ∎

In the following sub-sections we will derive the internal and external angles for a face $\mathcal{F}$, and a face $\mathcal{G}$ containing $\mathcal{F}$, and will provide closed-form upper bounds for them. We combine the terms together and compute the exponents using the Laplace method in Section 5.6.2, and derive thresholds for the negativity of the cumulative exponent.

Computation of Internal and External Angles

Theorem 5.6.2. *Let Z be a random variable defined as*

$$Z = (k_1 w_{K_1}^2 + k_2 w_{K_2}^2)X_1 - w_{K_1}^2 \sum_{i=1}^{t_1} X_1' - w_{K_2}^2 \sum_{i=1}^{t_2} X_1'',$$

where $X_1 \sim N(0, \frac{1}{2(k_1 w_{K_1}^2 + k_2 w_{K_2}^2)})$ is a normal distributed random variable, and $X_i' \sim HN(0, \frac{1}{2w_{K_1}^2})$ $1 \le i \le t_1$, and $X_i'' \sim HN(0, \frac{1}{2w_{K_2}^2})$ $1 \le i \le t_2$ are independent (from each other and from X_1) half-normal distributed random variables. Let $p_Z(\cdot)$ denote the probability distribution function of Z and $c_0 = \frac{\sqrt{\pi}}{2^{l-k}}\left((k_1+t_1)w_{K_1}^2 + (k_2+t_2)w_{K_2}^2\right)^{1/2}$. Then

$$\beta(k_1, k_2 | t_1, t_2) = c_0 p_Z(0). \tag{5.6.11}$$

Theorem 5.6.3. *The external angle $\zeta(\mathcal{G}, \mathcal{P}_{\mathbf{w}}) = \zeta(d_1, d_2)$ between the face $\mathcal{G}$ and $\mathcal{P}_{\mathbf{w}}$, where G is supported on the set L with $|L \cap K_1| = d_1$ and $|L \cap K_2| = d_2$ is given by:*

$$\zeta(d_1, d_2) = \pi^{-\frac{n-l+1}{2}} 2^{n-l} \int_0^\infty e^{-x^2} \left(\int_0^{\frac{w_{K_1} x}{\xi(d_1,d_2)}} e^{-y^2}\, dy \right)^{r_1} \left(\int_0^{\frac{w_{K_2} x}{\xi(d_1,d_2)}} e^{-y^2}\, dy \right)^{r_2} dx, \tag{5.6.12}$$

where $\xi^2(d_1, d_2) = \sum_{i \in L} w_i^2 = d_1 w_{K_1}^2 + d_2 w_{K_2}^2$, $r_1 = n_1 - d_1$, and $r_2 = n_2 - d_2$.

The detailed proofs of these theorems, which are generalizations of similar Theorems in [XH08], are given in [KXAH11].

Derivation of the Critical Weak and Strong δ_c Threshold

So far we have proved that the probability of the failure event is bounded by the formula

$$\mathbb{P}\{E^c\} \leq \sum_{\substack{0 \leq t_1 \leq n_1 - k_1 \\ 0 \leq t_2 \leq n_2 - k_2 \\ t_1 + t_2 > m - k_1 - k_2 + 1}} 2^{t_1+t_2+1} \binom{n_1 - k_1}{t_1} \binom{n_2 - k_2}{t_2} \beta(k_1, k_2 | t_1, t_2) \zeta(t_1 + k_1, t_2 + k_2), \tag{5.6.13}$$

where we gave expressions for $\beta(t_1, t_2|k_1, k_2)$ and $\zeta(t_1 + k_1, t_2, k_2)$ in Section 5.6.2. Now our objective is to show that the R.H.S of (5.6.13) will exponentially decay to 0 as $n \to \infty$, provided that $\delta = \frac{m}{n}$ is greater than a critical threshold δ_c, which we are trying to evaluate. To this end we bound the exponents of the combinatorial, internal angle and external angle terms in (5.6.13), and find the values of δ for which the net exponent is strictly negative. The maximum such δ will give us δ_c. Starting with the combinatorial term, we use Stirling approximating on the binomial coefficients to achieve the following as $n \to \infty$ and $\epsilon \to 0$

$$\begin{aligned} &\frac{1}{n} \log \left(2^{t_1+t_2+1} \binom{n_1 - k_1}{t_1} \binom{n_2 - k_2}{t_2} \right) \to \\ &\left(\gamma_1(1-p_1) H\left(\frac{\tau_1}{\gamma_1(1-p_1)}\right) + \gamma_2(1-p_2) H\left(\frac{\tau_2}{\gamma_2(1-p_2)}\right) + \tau_1 + \tau_2 \right) \log 2, \end{aligned} \tag{5.6.14}$$

where $\tau_1 = \frac{t_1}{n}$ and $\tau_2 = \frac{t_2}{n}$.

For the external angle and internal angle terms we prove the following two exponents:

1. Let $g(x) = \frac{2}{\sqrt{\pi}} e^{-x^2}$, $G(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-y^2} dy$. Also define $c = (\tau_1 + \gamma_1 p_1) + \omega^2(\tau_2 + \gamma_2 p_2)$, $\alpha_1 = \gamma_1(1 - p_1) - \tau_1$, and $\alpha_2 = \gamma_2(1 - p_2) - \tau_2$. Let x_0 be the unique

solution to x of the following:

$$2c - \frac{g(x)\alpha_1}{xG(x)} - \frac{\omega g(\omega x)\alpha_2}{xG(\omega x)} = 0.$$

Define

$$\psi_{ext}(\tau_1, \tau_2) = cx_0^2 - \alpha_1 \log G(x_0) - \alpha_2 \log G(\omega x_0). \tag{5.6.15}$$

2. Let $b = \frac{\tau_1+\omega^2\tau_2}{\tau_1+\tau_2}$, and $\varphi(.)$ and $\Phi(.)$ be the standard Gaussian pdf and cdf functions, respectively. Also let $\Omega' = \gamma_1 p_1 + \omega^2\gamma_2 p_2$ and $Q(s) = \frac{\tau_1\varphi(s)}{(\tau_1+\tau_2)\Phi(s)} + \frac{\omega\tau_2\varphi(\omega s)}{(\tau_1+\tau_2)\Phi(\omega s)}$. Define the function $\hat{M}(s) = -\frac{s}{Q(s)}$ and solve for s in $\hat{M}(s) = \frac{\tau_1+\tau_2}{(\tau_1+\tau_2)b+\Omega'}$. Let the unique solution be s^* and set $y = s^*(b - \frac{1}{\hat{M}(s^*)})$. Compute the rate function $\Lambda^*(y) = sy - \frac{\tau_1}{\tau_1+\tau_2}\Lambda_1(s) - \frac{\tau_2}{\tau_1+\tau_2}\Lambda_1(\omega s)$ at the point $s = s^*$, where $\Lambda_1(s) = \frac{s^2}{2} + \log(2\Phi(s))$. The internal angle exponent is then given by:

$$\psi_{int}(\tau_1, \tau_2) = (\Lambda^*(y) + \frac{\tau_1+\tau_2}{2\Omega'}y^2 + \log 2)(\tau_1+\tau_2). \tag{5.6.16}$$

We now state the following lemmas, the proof techniques of which are very similar to similar arguments in [Don06b, DT05a]. The details of the proof can be found in [KXAH11].

Lemma 5.6.4. *Fix δ, $\epsilon > 0$. There exists a finite number $n_0(\delta, \epsilon)$ such that*

$$\frac{1}{n}\log(\zeta(t_1+k_1, t_2+k_2)) < -\psi_{ext}(\tau_1, \tau_2) + \epsilon, \tag{5.6.17}$$

uniformly in $0 \le t_1 \le n_1 - k_1$, $0 \le t_2 \le n_2 - k_2$, and $t_1 + t_2 \ge m - k_1 - k_2 + 1$, $n \ge n_0(\delta, \epsilon)$.

Lemma 5.6.5. *Fix δ, $\epsilon > 0$. There exists a finite number $n_1(\delta, \epsilon)$ such that*

$$\frac{1}{n}\log(\beta(t_1, t_2|k_1, k_2)) < -\psi_{int}(\tau_1, \tau_2) + \epsilon, \tag{5.6.18}$$

uniformly in $0 \le t_1 \le n_1 - k_1$, $0 \le t_2 \le n_2 - k_2$, and $t_1 + t_2 \ge m - k_1 - k_2 + 1$, $n \ge n_1(\delta, \epsilon)$.

Combining Lemmas 5.6.4 and 5.6.5, (5.6.14), and the bound in (5.6.13), we readily get the critical bound for δ_c, as in the Theorem 5.5.3.

Derivation of the strong and sectional threshold can be easily done using union bounds to account for all possible support sets and/or all sign patterns. The corresponding upper bound on the failure probability for the strong threshold is given by:

$$\binom{n_1}{k_1}\binom{n_2}{k_2}2^k P_{K,-}. \tag{5.6.19}$$

It then follows that the strong threshold of δ is given by δ_c in Theorem 5.5.3, except that the combinatorial exponent $\psi_{com}(\cdot,\cdot)$ must be corrected by adding a term

$$(\gamma_1 p_1 + \gamma_2 p_2 + \gamma_1 H(p_1) + \gamma_2 H(p_2)) \log 2, \tag{5.6.20}$$

to the RHS of (5.5.3). Similarly, for the sectional threshold, which deals with all possible support sets but almost all sign patterns, the modification in the combinatorial exponent term is as follows:

$$(\gamma_1 H(p_1) + \gamma_2 H(p_2)) \log 2. \tag{5.6.21}$$

5.6.3 Generalizations

Except for some subtlety in the large deviation calculations, the generalization of the results of the previous section to an arbitrary $u \geq 2$ classes of entries is straightforward. Consider a nonuniform sparse model with u classes $K_1, \cdots, K_u$ where $|K_i| = n_i = \gamma_i n$ and the sparsity fraction over the set K_i is p_i, and a recovery scheme based on weighted ℓ_1 minimization with weight ω_i for the set K_i. The bound in (5.6.7) is general and can always be used. Due to isomorphism, the internal and external angles $\beta(\mathcal{F},\mathcal{G})$ and $\zeta(\mathcal{G},\mathcal{P}_{\mathbf{w}})$ only depend on the number of vertices that the supports of $\mathcal{F}$ and $\mathcal{G}$ have in common with each K_i. Therefore, a generalization to

(5.5.2) would be:

$$\mathbb{P}\{E^c\} \leq 2 \sum_{\substack{0 \leq \mathbf{t} \leq \mathbf{n}-\mathbf{k} \\ \mathbf{1}^T\mathbf{t} > m - \mathbf{1}^T\mathbf{k} + 1}} \Pi_{1 \leq i \leq u} 2^{t_i} \binom{n_i - k_i}{t_i} \beta(\mathbf{k}|\mathbf{t}) \zeta(\mathbf{t}+\mathbf{k}), \tag{5.6.22}$$

where $\mathbf{t} = (t_1, \ldots, t_u)^T$, $\mathbf{k} = (k_1, \ldots, k_u)^T$, and $\mathbf{1}$ is a vector of all ones. Invoking generalized forms of Theorems 5.6.3 and 5.6.2 to approximate the terms $\beta(\mathbf{k}|\mathbf{t})$ and $\zeta(\mathbf{k}+\mathbf{t})$, we conclude the following theorem.

Theorem 5.6.6. *Consider a nonuniform sparse model with u classes $K_1, \ldots, K_u$, with $|K_i| = n_i = \gamma_i n$ and sparsity fractions $p_1, p_2, \ldots, p_u$, where n is the signal dimension. Also, let the functions $g(.), G(.), \psi(.), \Psi(.)$ be defined as in Theorem 5.5.3. For positive values $\{\omega_i\}_{i=1}^u$, the recovery thresholds (weak, sectional, and strong) of the weighted ℓ_1 minimization program:*

$$\min_{\mathbf{Ax}=\mathbf{y}} \sum_{i=1}^{u} \omega_i \|\mathbf{x}_{K_i}\|_1,$$

is given by the following expression:

$$\delta_c = \min\{\delta \mid \psi_{com}(\tau) - \psi_{int}(\tau) - \psi_{ext}(\tau) < 0 \ \forall \tau = (\tau_1, \cdots, \tau_u)^T : 0 \leq \tau_i \leq \gamma_i(1-p_i) \ \forall 1 \leq i \leq u, \ \sum_{i=1}^{u} \tau_i > \delta - \sum_{i=1}^{u} \gamma_i p_i\}$$

where ψ_{com}, ψ_{int}, and ψ_{ext} are obtained from the following expressions:

1. *$\psi_{com}(\tau) = \log 2 \sum_{i=1}^{u} \gamma_i (1-p_i) H(\frac{\tau_i}{\gamma_i(1-p_i)}) + \tau_i$, for the weak threshold. For sectional threshold this must be modified by adding a term $\log 2 \sum_{i=1}^{u} \gamma_i H(p_i)$. For strong threshold, $\sum_{i=1}^{u} \gamma_i p_i$ must also be added with.*

2. *$\psi_{ext}(\tau) = c x_0^2 - \sum_{i=1}^{u} \alpha_i \log G(\omega_i x_0)$, where $c = \sum_{i=1}^{u} \omega_i^2 (\tau_i + \gamma_i p_i)$, $\alpha_i = \gamma_i(1-p_i) - \tau_i$, and x_0 is the unique solution of $2c = \sum_{i=1}^{u} \omega_i \frac{g(\omega_i x_0)\alpha_i}{x_0 G(\omega x_0)}$.*

3. $\psi_{int}(\tau) = \lambda(\Lambda^*(y) + \frac{\lambda y^2}{2\sum_{i=1}^u \omega_i^2\gamma_i p_i} + \log 2)$, *where* $\lambda = \sum_{i=1}^u \tau_i$, *and* y *and* $\Lambda^*(y)$ *are obtained as follows. Let* $b = \frac{\sum_{i=1}^u \omega_i^2\tau_i}{\lambda}$, $Q(s) = \sum_{i=1}^u \frac{\tau_i\varphi(s)}{\lambda\Phi(s)}$. *Let* s^* *be the solution to* s *in* $-\frac{Q(s)}{s} = b + \frac{\sum_{i=1}^u \omega_i^2\gamma_i p_i}{\lambda}$, *and* $y = s^*(b - \frac{1}{\hat{M}(s^*)})$. *Then* $\Lambda^*(y) = s^*y - 1/\lambda\sum_{i=1}^u \tau_i\left(\frac{\omega_i^2 s^{*2}}{2} + \log(2\Phi(\omega_i s^*))\right)$.

5.6.4 Robustness To Noise

Proof of Theorem 5.5.6. We first state the following lemma, which is very similar to Theorem 2 of [XH08]. We skip its proof for brevity.

Lemma 5.6.7. *Let* $K \subset \{1, 2\cdots, n\}$ *and the weight vector* $\mathbf{w} = (w_1, w_2, \cdots, w_n)^T$ *be fixed. Define* $\mathbf{W} = diag(w_1, w_2, \cdots, w_n)$ *and suppose* $C > 1$ *is given. For every vector* $\mathbf{x}_0 \in \mathbb{R}^{n\times 1}$, *the solution* $\hat{\mathbf{x}}$ *of (6.5.2) satisfies*

$$\|\mathbf{W}(\mathbf{x}_0 - \hat{\mathbf{x}})\|_1 \leq 2\frac{C+1}{C-1}\sum_{i\in\overline{K}} w_i|(x_0)_i|, \tag{5.6.23}$$

if and only if for every $\mathbf{z} \in \mathcal{N}(\mathbf{A})$ *the following holds:*

$$C\sum_{i\in K} w_i|z_i| \leq \sum_{i\in\overline{K}} w_i|z_i|. \tag{5.6.24}$$

Let $\mathbf{z} = (z_1, \cdots, z_n)^T$ be a vector in the null space of $\mathbf{A}$, and assume that

$$C'\sum_{i\in L_1\cup L_2} w_i|z_i| = \sum_{i\in\overline{L_1}\cap\overline{L_2}} w_i|z_i|. \tag{5.6.25}$$

Let K_{ϵ_1} and K_{ϵ_2} be the solutions of the following problems

$$K_{\epsilon_1} \quad : \quad \max_{K_{\epsilon_1}\subset K_1\cap\overline{L_1}, |K_{\epsilon_1}| = \epsilon_1\gamma_1 p_1 n} \sum_{i\in K_{\epsilon_1}} w_i|z_i|, \tag{5.6.26}$$

$$K_{\epsilon_2} \quad : \quad \max_{K_{\epsilon_2}\subset K_2\cap\overline{L_2}, |K_{\epsilon_2}| = \epsilon_2\gamma_2 p_2 n} \sum_{i\in K_{\epsilon_2}} w_i|z_i|. \tag{5.6.27}$$

Let $L_1' = L_1 \cup K_{\epsilon_1}$ and $L_2' = L_2 \cup K_{\epsilon_2}$. From the definition of K_{ϵ_1} and K_{ϵ_2}, it follows

that

$$\sum_{i\in K_{\epsilon_1}} w_i|z_i| \;\geq\; \frac{\epsilon_1 p_1}{1-p_1}\sum_{i\in\overline{L_1'}\cap K_1} w_i|z_i|, \tag{5.6.28}$$

$$\sum_{j\in K_{\epsilon_2}} w_j|z_j| \;\geq\; \frac{\epsilon_2 p_2}{1-p_2}\sum_{j\in\overline{L_2'}\cap K_2} w_j|z_j|. \tag{5.6.29}$$

Adding $C'\left(\sum_{i\in K_{\epsilon_1}} w_i|z_i| + \sum_{j\in K_{\epsilon_2}} w_j|z_j|\right)$ to both sides of (5.6.25) and using (5.6.28) and (5.6.29), we can write:

$$\begin{aligned} C'\sum_{i\in L_1'\cup L_2'} w_i|z_i| &\geq \sum_{i\in\overline{L_1}\cap\overline{L_2}} w_i|z_i| + C'\left(\frac{\epsilon_1 p_1}{1-p_1}\sum_{i\in\overline{L_1'}\cap K_1} w_i|z_i| + \frac{\epsilon_2 p_2}{1-p_2}\sum_{i\in\overline{L_2'}\cap K_2} w_i|z_i|\right) \\ &\geq \left(1+(C'+1)\min(\frac{\epsilon_1 p_1}{1-p_1},\frac{\epsilon_2 p_2}{1-p_2})\right)\sum_{i\in\overline{L_1'}\cap\overline{L_2'}} w_i|z_i|. \end{aligned} \tag{5.6.30}$$

Note that $|L_1'| = \gamma_1 p_1 n$ and $|L_2'| = \gamma_2 p_2 n$. Therefore, since $\delta = \frac{m}{n} \geq \delta_c^{(S)}(\gamma_1,\gamma_2,p_1,p_2,\omega)$, we know that $\sum_{i\in L_1'\cup L_2'} w_i|z_i| \leq \sum_{i\in\overline{L_1'}\cap\overline{L_2'}} w_i|z_i|$. From this and (5.6.30) we conclude that

$$C' \geq \left(1+(C'+1)\min(\frac{\epsilon_1 p_1}{1-p_1},\frac{\epsilon_2 p_2}{1-p_2})\right), \tag{5.6.31}$$

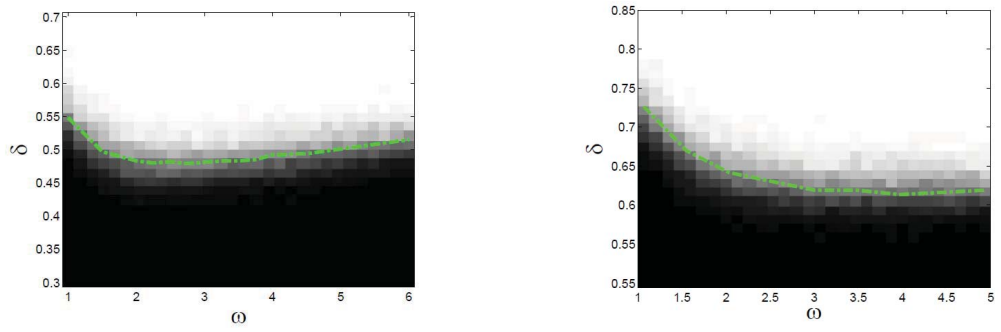
or equivalently

$$C' \geq \frac{1+\min(\frac{\epsilon_1 p_1}{1-p_1},\frac{\epsilon_2 p_2}{1-p_2})}{1-\min(\frac{\epsilon_1 p_1}{1-p_1},\frac{\epsilon_2 p_2}{1-p_2})}. \tag{5.6.32}$$

Using Lemma 5.6.7 and the above inequality, we conclude (7.4.4). ■

5.7 Simulation Results

We demonstrate by some examples that appropriate weights can boost the recovery percentage. In Figure 5.8 we have shown the empirical recovery threshold of weighted ℓ_1 minimization for different values of the weight $\omega = \frac{w_{K_1}}{w_{K_2}}$ for two particular nonuniform sparse models. Note that the empirical threshold is somewhat identifiable with the naked eye and is very similar to the theoretical curve of Figure 5.4 for similar

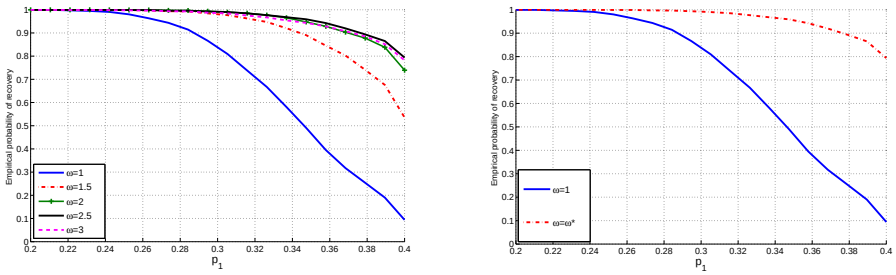


(a) $\gamma_1 = \gamma_2 = 0.5$, $p_1 = 0.4$ and $p_2 = 0.05$. (b) $\gamma_1 = \gamma_2 = 0.5$, $p_1 = 0.65$ and $p_2 = 0.1$.

Figure 5.8: Empirical recovery percentage of weighed ℓ_1 minimization for different weight values ω, and different number of measurements $\delta = \frac{m}{n}$ and $n = 200$. Signals have been selected from a nonuniform sparse models. White indicates perfect recovery..

settings. In another experiment, we fix p_2 and $n = 2m = 200$, and try ℓ_1 and weighted ℓ_1 minimization for various values of p_1. We choose $n_1 = n_2 = \frac{n}{2}$. Figure 5.9a shows one such comparison for $p_2 = 0.05$ and different values of w_{K_2}. Note that the optimal value of w_{K_2} varies as p_1 changes. Figure 5.9b illustrates how the optimal weighted ℓ_1 minimization surpasses the ordinary ℓ_1 minimization. The optimal curve is basically achieved by selecting the best weight of Figure 5.9a for each single value of p_1. Figure 5.10 shows the result of simulations in another setting where $p_2 = 0.1$ and $m = 0.75n$ (similar to the setting of Section 5.5). Note that these results match very well the theoretical results of Figures 5.4a and 5.4b.

Another nonuniform model with $\gamma_1 = 0.25$, $\gamma_2 = 0.75$ was considered for simulations. We fixed $\delta = 0.45$, and the overall sparsity fraction $\gamma_1 p_1 + \gamma_2 p_2 = 0.3$. For random vectors of size $n = 200$, the probability of successful recovery as a function of the sparsity of the second subclass, i.e., p_2, for various weights ω was obtained, and is depicted in Figure 5.11. As displayed, although ℓ_1 minimization fails in all of the cases to recover the sparse vectors, various weighted ℓ_1 approaches have higher chances of success, especially when the first subclass is very dense and a high weight



(a) (b)

Figure 5.9: Empirical probability of successful recovery for weighted ℓ_1 minimization with different weights (unitary weight for the first subclass and ω for the second one) and suboptimal weights in a nonuniform sparse setting. $p_2 = 0.05$, $\gamma_1 = \gamma_2 = 0.5$, and $m = 0.5n = 100$. ω^* in (b) is the optimum value of ω for each p_1 among the values shown in (a).

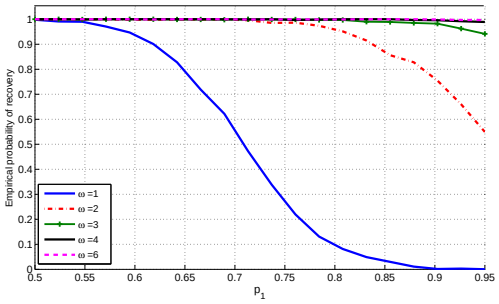


Figure 5.10: Empirical probability of successful recovery for different weights. $p_2 = 0.1$,$\gamma_1 = \gamma_2 = 0.5$ and $m = 0.75n = 150$

is used ($p_1 \approx 1$, $\omega >> 1$).

In Figure 5.12, we have displayed the performance of weighted ℓ_1 minimization in the presence of noise. The original signal is a nonuniformly sparse vector with sparsity fractions $p_1 = 0.4, p_2 = 0.05$ over two subclasses $\gamma_1 = \gamma_2 = 0.5$, with $n = 200$. However, a white Gaussian noise vector is added before compression, i.e., $\mathbf{y} = \mathbf{A}\mathbf{x}+\mathbf{v}$, where $\mathbf{v}$ (0, is an i.i.d. Gaussian vector. Figure 5.12 shows a scatter plot of all output signal-to-recovery-error ratios as a function of the input SNR, for all simulations. The input SNR in dB is defined as $10\log_{10}\frac{\|\mathbf{x}\|_2^2}{\|\mathbf{v}\|_2^2}$, and output signal-to-recovery-error in dB is defined as $10\log_{10}\frac{\|\mathbf{x}\|_2^2}{\|\hat{\mathbf{x}}-\mathbf{x}\|_2^2}$. The fact that the signal-to-recovery-error does not drop drastically in small SNR regimes, and is mostly concentrated around the

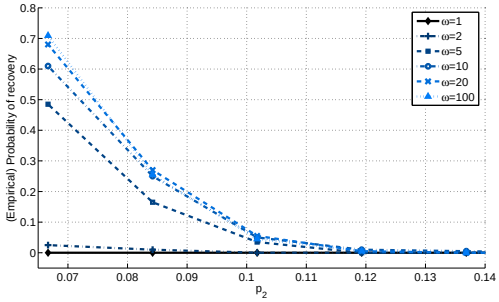
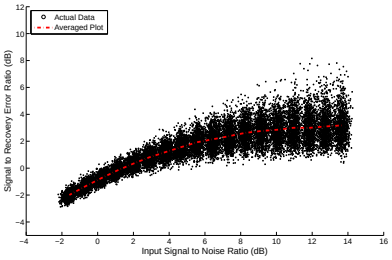


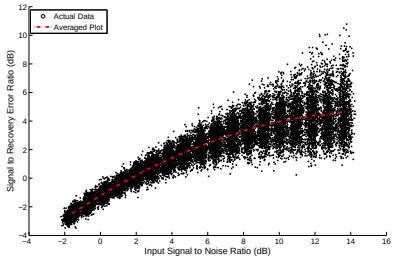
Figure 5.11: Probability of successful recovery (empirical) of nonuniform sparse signals with $\gamma_1 = 0.25, \gamma_2 = 0.75, p_1\gamma_1 + p_2\gamma_2 = 0.3$ vs. the sparsity of the second subclass p_2. $\delta = 0.45$.

average values, indicates the robustness of the weighted ℓ_1 algorithm in the presence of moderate-level noise. In Figure 5.13 the average curves are compared for different values of weight ω. We can see that in the high-input SNR regime, a non-unit weight $\omega = 3$ is advantageous over the regular ℓ_1 minimization. However, for stronger noise variances, ℓ_1 minimization seems to be more robust and yields better performance.

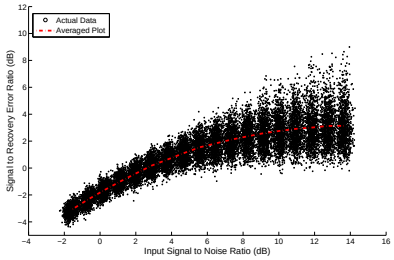
We have done some experiments with regular ℓ_1 and weighted ℓ_1 minimization recovery on some real-world data. We have chosen a pair of satellite images (Figure 5.14) taken in two different years, 1989 (left) and 2000 (right), from the New Britain rainforest in Papua New Guinea. The Images belong to the Royal Society for the Protection of Birds and were taken from an article on deforestation in the Guardian archive. These images are generally recorded to evaluate environmental effects such as deforestation. The difference between images taken at different times is generally not very significant, and thus can be thought of as compressible. In addition, the difference is usually more substantial over certain areas, e.g., forests, therefore, it can be cast in a nonuniform sparse model. We have applied ℓ_1 minimization to recover the difference image over the subframe (subset of the original images), identified by the red rectangles in Figure 5.14. In addition, recovery by weighted ℓ_1 minimization was also implemented. To assign weights, we divided the pixels of each frame into two classes of equal sizes, where the concentration of the forested area is larger over one



(a) $\omega = 1$.



(b) $\omega = 3$.



(c) $\omega = 5$.

Figure 5.12: Signal-to-recovery-error ratio for weighted ℓ_1 minimization with weight ω vs. input SNR for nonuniform sparse signals with $\gamma_1 = \gamma_2 = 0.5$, $p_1 = 0.4, p_2 = 0.05$ superimposed with Gaussian noise

of the classes, and hence the difference image is less sparse. This class is identified by the union of two rectangles with green frames on the left and bottom right of the image. The justification for such a decomposition is the concentration of green area. In other words, with the knowledge that the environmental changes are more apparent over forests in large scale, the difference image is expected to be denser over those regions. The original size of the image is 275×227. We reduced the resolution by roughly a factor of 0.05 for more tractability of ℓ_1 solver in MATLAB. In addition, only the gray-scale version of the difference image was taken into account, and was normalized so that the maximum intensity is 1. Furthermore, prior to compression, the difference image was further sparsified by rounding the intensities less than 0.1 to zero. We picked the weight value $\omega = 2$ for the weighted ℓ_1 recovery. We defined the normalized recovery error to be the sum square of the intensity differences in the recovered and the original image, divided by the sum square of the original image

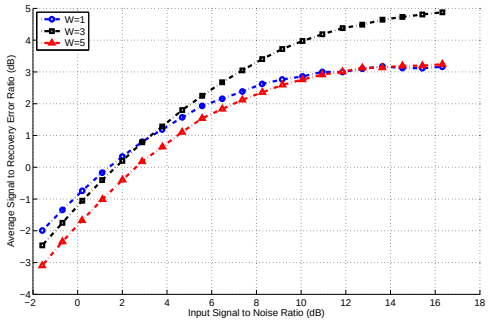


Figure 5.13: Average signal-to-recovery error ratio for weighted ℓ_1 minimization with weight ω vs. input SNR for nonuniform sparse signals with $\gamma_1 = \gamma_2 = 0.5$, $p_1 = 0.4, p_2 = 0.05$ superimposed with Gaussian noise

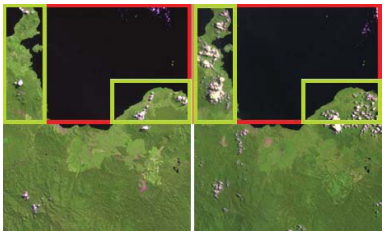


Figure 5.14: Satellite images taken from the New Britain rainforest in Papua Guina at 1989 (left) and 2000 (right). Red boxes identify the subframe used for the experiment, and green boxes identify the regions with higher associated weight in the weighted ℓ_1 recovery. Image belongs to the Royal Society for the Protection of Birds and was taken from an article on deforestation in the Guardian archive [sta].

intensity, i.e., $\sum_{i\in\text{frame}}(I_i - \hat{I}_i)^2 / \sum_{i\in\text{frame}} I_i^2$. The average normalized error for the two different recovery methods, namely ℓ_1 and weighted ℓ_1, is displayed in Figure 5.14 as a function of δ. The average is taken over 50 realizations of i.i.d. Gaussian measurement matrices for each δ. As can be seen, the recovery improvement is significant in the weighted ℓ_1 minimization.

Another experiment was done on a pair of brain fMRI images taken at two different instances of time, shown in Figure 5.15. Similar to the satellite images, the objective is recover the difference image from a set of compressed measurements. The significant portion of the difference image in fMRI lies on the regions where the brain is identified as most active. Depending on the particular task that the patient undertakes, these regions can be (roughly) known a priori. The original image size is

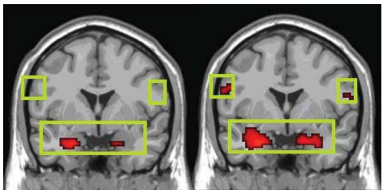
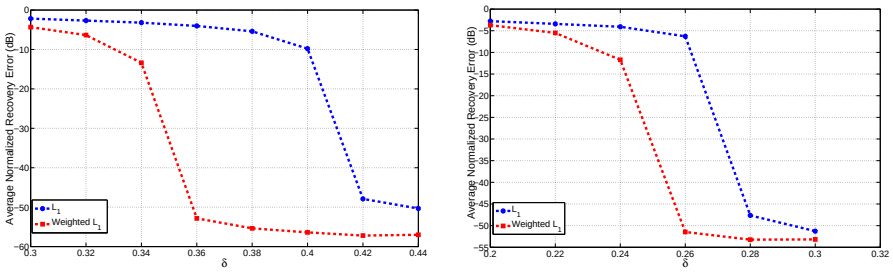


Figure 5.15: Functional MRI images of the brain at two different instances illustrating the brain activity. Green boxes identify the region with higher associated weight in the weighted ℓ_1 recovery. Image is adopted from https://sites.google.com/site/psychopharmacology2010/student-wiki-for-quiz-9.



(a) (b)

Figure 5.16: Average normalized recovery error for ℓ_1 and weighted ℓ_1 minimization recovery of the difference between the subframes of (a) a pair of satellite images shown in Figure 5.14, and (b) the pair of brain fMRI images shown in Figure 5.15. Data is averaged over different realizations of measurement matrices for each δ.

271×271, and similar preprocessing steps as for the satellite images were followed before compression. We used ℓ_1 minimization and weighted ℓ_1 minimization with a higher weight $\omega = 1.3$ on the regions identified by the green boxes. This choice of ω resulted in a slightly better performance in weighted ℓ_1 algorithm than $\omega = 2$. The average normalized recovery errors are displayed in Figure 5.15, from which we can infer similar conclusions, as in the case of the satellite images.

5.8 Proof of Theorems

Proof of Theorem 5.5.4.

Let $\delta' = \delta_c(\gamma_1, \gamma_2, 1, p_2, \omega)$ and $\delta'' = \delta_c(0, 1, 0, p_2, 1)$. From Theorem 5.5.3 we know

that:

$$\begin{aligned}\delta' =& \min\{\delta \mid \psi'_{com}(0,\tau_2) - \psi'_{int}(0,\tau_2) - \psi'_{ext}(0,\tau_2) < 0 \ \forall\ 0 \le \tau_2 \le \gamma_2(1-p_2),\\ & \tau_2 > \delta - \gamma_1 - \gamma_2 p_2\}\\ =& \gamma_2 \min\{\delta \mid \psi'_{com}(0,\gamma_2\tau_2) - \psi'_{int}(0,\gamma_2\tau_2) - \psi'_{ext}(0,\gamma_2\tau_2) < 0 \ \forall\ 0 \le \tau_2 \le 1-p_2,\\ & \tau_2 > \delta - p_2\} + \gamma_1, \qquad (5.8.1)\end{aligned}$$

$$\delta'' = \min\{\delta \mid \psi''_{com}(0,\tau_2) - \psi''_{int}(0,\tau_2) - \psi''_{ext}(0,\tau_2) < 0 \ \forall\ 0 \le \tau_2 \le 1-p_2, \tau_2 > \delta - p_2\}, \qquad (5.8.2)$$

where the exponents ψ'_{com},ψ'_{int},ψ'_{ext},ψ''_{com},ψ''_{int}, and ψ''_{ext} can be found using Theorem 5.5.3. From the definition of ψ_{com} in (5.5.3) it immediately follows that $\psi'_{com}(0,\gamma_2\tau_2) = \gamma_2\psi''_{com}(0,\tau_2)$. From (5.5.4) for $\omega \to \infty$ we know that $\psi'_{ext}(0,\gamma_2\tau_2) = c'{x'_0}^2 - \alpha'_2 \log G(\omega x'_0)$ and $\psi''_{ext}(0,\tau_2) = c''{x''_0}^2 - \alpha''_2 \log G(x''_0)$. Following the details of derivations as in Theorem 5.5.3, we realize that: $c' = \gamma_2\omega^2 c''$, $\omega x'_0 = x''_0$, $\alpha'_2 = \gamma_2\alpha''_2$,, which implies $\psi'_{ext}(0,\gamma_2\tau_2) = \gamma_2\psi''_{ext}(0,\tau_2)$. Finally, from (5.5.5), we know that $\psi'_{int}(0,\gamma_2\tau_2) = (\Lambda^*(y') + \frac{\gamma_2\tau_2}{2\Omega'}y'^2 + \log 2)\gamma_2\tau_2$, $\psi''_{int}(0,\tau_2) = (\Lambda^*(y'') + \frac{\tau_2}{2\Omega''}y''^2 + \log 2)\tau_2$. Following the details of derivations as in Theorem 5.5.3, we realize that if $\omega \to \infty$, then $y' = y''$, $\Omega' = \gamma_2\Omega''$, which implies that $\psi'_{int}(0,\gamma_2\tau_2) = \gamma_2\psi''_{int}(0,\tau_2)$. From (6.6.3), (6.6.8), and the above conclusions, it follows that $\delta' = \gamma_2\delta'' + \gamma_1$.

Proof of Theorem 5.5.5.

The proof is based on a recently found connection between the phase transition thresholds of atomic norm minimizations and robustness of de-noising problems stated formally in the following lemma:

Lemma 5.8.1 (Deduced from [DJM11, OKH12])**.** *Let* $\mathbf{x}$ *be a random signal of size* n *randomly selected from a sparse family* F*,* $\mathbf{A}$ *be a random i.i.d. Gaussian matrix of size* $m \times n$*,* $\mathbf{y} = \mathbf{A}\mathbf{x}_0$*, and* $\mathbf{w}$ *be a weight vector. Let* $\delta(F,\mathbf{w}) = m/n$ *be the weak threshold that guarantees successful recovery of* $\mathbf{x}_0$ *with high probability using*

the following optimization:

$$\min_{\mathbf{Ax}=\mathbf{y}} \|\mathbf{w}^T\mathbf{x}\|_1. \tag{5.8.3}$$

Also, let $\mathbf{v}$ *be a vector of length* n *(noise vector) with i.i.d entries with distribution* $\mathcal{N}(0,\sigma_v^2)$*, and* $\mathbf{z}=\mathbf{x}_0+\mathbf{v}$*, and* $\hat{x}^{(\lambda)}=arg\min_{\mathbf{x}}\|\mathbf{wx}\|_1+\lambda\|\mathbf{x}-\mathbf{z}\|_2^2$*. Then:*

$$\mathbb{E}\min_{\lambda>0}\|\mathbf{x}_0-\hat{\mathbf{x}}^{(\lambda)}\|_2^2=n\delta(F,\mathbf{w})\sigma_v^2. \tag{5.8.4}$$

Now let $\mathbf{x}=(\mathbf{x}_1,\mathbf{x}_2)^T$ be a nonuniformly sparse signal with sparsity fractions (p_1,p_2) and class size ratios (γ_1,γ_2), and let $\omega>0$ be a weight constant. According to the above theorem, if

$$\mathbf{z}=\mathbf{x}+\mathbf{v}=\begin{pmatrix}\mathbf{x}_1\\ \\ x_2\end{pmatrix}+\begin{pmatrix}\mathbf{v}_1\\ \\ \mathbf{v}_2\end{pmatrix},$$

where the entries of $\mathbf{v}$ are i.i.d. $\mathcal{N}(0,\sigma_v^2)$, and if $\hat{\mathbf{x}}^{(\lambda,\omega)}=(\hat{\mathbf{x}}_1^{(\lambda,\omega)},\hat{\mathbf{x}}_2^{(\lambda,\omega)})^T$ is the de-noising solution:

$$(\hat{\mathbf{x}}_1^{(\lambda,\omega)},\hat{\mathbf{x}}_2^{(\lambda,\omega)})=\arg\min_{\mathbf{x}_1,\mathbf{x}_2}\|\mathbf{x}_1\|_1+\omega\|\mathbf{x}_2\|_1+\lambda\|\mathbf{x}_1-\mathbf{z}_1\|_2^2+\lambda\|\mathbf{x}_2-\mathbf{z}_2\|_2^2, \tag{5.8.5}$$

then we have:

$$\mathbb{E}\min_{\lambda>0}\left(\|\hat{\mathbf{x}}_1^{(\lambda,\omega)}-\mathbf{z}_1\|_2^2+\|\hat{\mathbf{x}}_2^{(\lambda,\omega)}-\mathbf{z}_2\|_2^2\right)=n\delta_c((p_1,p_2),(\gamma_1,\gamma_2),(1,\omega))\sigma_v^2. \tag{5.8.6}$$

Therefore:

$$\mathbb{E}\min_{\lambda>0,\omega>0}\left(\|\hat{\mathbf{x}}_1^{(\lambda,\omega)}-\mathbf{z}_1\|_2^2+\|\hat{\mathbf{x}}_2^{(\lambda,\omega)}-\mathbf{z}_2\|_2^2\right)=n\delta_c((p_1,p_2),(\gamma_1,\gamma_2),(1,\omega^*))\sigma_v^2, \tag{5.8.7}$$

where ω^* is the best possible weight. However, it is not hard to see that for the optimal de-noising and ℓ_1 regularization factors λ^*,ω^*, the error term is decomposed into the sum of errors of two independent de-noising problems. Specifically, if $\tilde{\mathbf{x}}_1^{(\lambda_1)}=\arg\min_{\mathbf{x}_1}\|\mathbf{x}_1\|_1+\lambda\|\mathbf{x}_1-\mathbf{z}_1\|_2^2$, $\tilde{\mathbf{x}}_2^{(\lambda_2)}=\arg\min_{\mathbf{x}_2}\|\mathbf{x}_2\|_1+\lambda\|\mathbf{x}_2-\mathbf{z}_2\|_2^2$, and λ^*,λ_1^*, and λ_2^* are the optimal values of regularization parameters in (5.8.7) and the two latter equations, respectively, then:

$$\hat{\mathbf{x}}_1^{(\lambda^*,\omega^*)}=\tilde{\mathbf{x}}_1^{(\lambda_1^*)},\ \hat{\mathbf{x}}_2^{(\lambda^*,\omega^*)}=\tilde{\mathbf{x}}_2^{(\lambda_2^*)}. \tag{5.8.8}$$

The reason for this is that minimizing the right-hand side of (5.8.5) is equivalent to minimizing the terms for $\mathbf{x}_1$ and $\mathbf{x}_2$ separately when λ and ω can independently take arbitrary values. Therefore, we have:

$$\mathbb{E}\left(\|\hat{\mathbf{x}}_1^{(\lambda^*,\omega^*)}-\mathbf{z}_1\|_2^2+\|\hat{\mathbf{x}}_2^{(\lambda^*,\omega^*)}-\mathbf{z}_2\|_2^2\right)=\mathbb{E}\|\tilde{\mathbf{x}}_1^{(\lambda_1^*)}-\mathbf{z}_1\|_2^2+\mathbb{E}\|\tilde{\mathbf{x}}_2^{(\lambda_2^*)}-\mathbf{z}_2\|_2^2, \quad (5.8.9)$$

which by the discussions above means:

$$n\delta((p_1,p_2),(\gamma_1,\gamma_2),(1,\omega^*))\sigma_v^2=n_1\delta(1,p_1,1)\sigma_v^2+n_2\delta(1,p_2,1)\sigma_v^2, \quad (5.8.10)$$

where $n_1=\gamma_1 n$ and $n_2=\gamma_2 n$ are the lengths of $\mathbf{x}_1$ and $\mathbf{x}_2$, respectively. Therefore:

$$\delta((p_1,p_2),(\gamma_1,\gamma_2),(1,\omega^*))=\gamma_1\delta(1,p_1,1)+\gamma_2\delta(1,p_2,1). \quad (5.8.11)$$

5.9 Conclusion

We analyzed the performance of the weighted ℓ_1 minimization for nonuniform sparse models. We computed explicitly the phase transition curves for the weighted ℓ_1 minimization, and showed that with proper weighting, the recovery threshold for weighted ℓ_1 minimization can be higher than that of regular ℓ_1 minimization. We provided simulation results to verify this, both in the noiseless and the noisy situation. Some of our simulations were performed on real-world data of satellite images, where the nonuniform sparse model is a valid assumption. Future work shall address generalizing the results of this paper to other measurement matrices with different distributions than i.i.d Gaussian. In particular, by using a similar idea as [Don04], one might be able to assert that for a class of distributions of measurement matrices including random Fourier ensembles, random Bernoulli, etc., similar sharp thresholds can be obtained for the weighted ℓ_1 minimization. A further interesting research topic to be addressed in future work would be to characterize the gain in recovery percentage as a function of the number of distinguishable classes u in the nonuniform model. In addition, we

have used the results presented in this chapter to build iterative reweighted ℓ_1 minimization algorithms that are provably strictly better than ℓ_1 minimization, when the nonzero entries of the sparse signals are known to come from certain distributions (in particular Gaussian distributions) [KXAH10, HKXA09]. The basic idea there is that a simple post-processing procedure on the output of ℓ_1 minimization results, with high probability, in a hypothetical nonuniform sparsity model for the unknown signal, which can be exploited for improved recovery. This approach will be elaborated upon in full detail in the next chapter.

Chapter 6

Reweighted Basis Pursuit

In the previous chapter, we studied the performance of weighted linear programming under the assumption of nonuniform sparsity, and showed that the phase transition thresholds of regular ℓ_1 minimization can be improved upon an appropriate choice of the weight vector. In this chapter, we concentrate on improving the phase transition curve (a.k.a recovery threshold) of linear programming, when no prior information about the signal sparsity model (such as the nonuniform sparse model previously discussed) exists. The idea is to use an iterative scheme that uses the results of a preliminary linear program to obtain post-processing reliability information about the sparsity structure of the unknown signal. This information is in the form of a hypothetic nonuniform sparsity assumption, and can then be utilized by an additional weighted ℓ_1 minimization step to improve the recovery performance. The analysis of the previous chapter is a solid prerequisite for the derivations of this chapter.

$\mathbf{A}$	measurement matrix
n	signal size
m	number of measurements
k	sparsity of the signal
δ	aspect ratio of $\mathbf{A}$, m/n
$\mu_W(\delta)$	weak recovery threshold of ℓ_1 minimization for k/n
C_S	robustness parameter for ℓ_1 minimization
$supp(\mathbf{x})$	support set of vector $\mathbf{x}$

6.1 Introduction

The initial results that established the fundamental phase transition thresholds for signal recovery using ℓ_1 minimization are due to Donoho and Tanner [DT05a, Don06b], where it is shown that if the measurement matrix is i.i.d. Gaussian, for a given ratio of $\delta = \frac{m}{n}$, ℓ_1 minimization can successfully recover *every* k-sparse signal, provided that $\mu = \frac{k}{n}$ is smaller than a certain threshold. This statement is true asymptotically as $n \to \infty$ and with high probability. This threshold guarantees the recovery of *all* sufficiently sparse signals and, as discussed before, is therefore referred to as a *strong* threshold. It thus does not depend on the actual distribution of the nonzero entries of the sparse signal and as such is a universal result. However, at this point, it is not known whether there exist other polynomial-time algorithms with strong thresholds superior to those of ℓ_1 minimization.

Another notion introduced and computed in [DT05a, Don06b] is that of a *weak* threshold where signal recovery is guaranteed for *almost all* support sets and *almost all* sign patterns of the sparse signal, with high probability as $n \to \infty$. The weak threshold is the one that can be observed in simulations of ℓ_1 minimization and allows for signal recovery beyond the strong threshold. The weak threshold of ℓ_1 minimization is also universal from the vantage point of signal distribution; the amplitudes of the nonzero entries of a sparse signal do not affect its recoverability via ℓ_1 minimization. In other words, if a sparse signal with a support set S and a particular sign pattern is

recoverable using ℓ_1 minimization, so is every other signal with the same support and sign pattern. It is worth noting that the weak thresholds of ℓ_1 minimization can be generalized to a broader class of random measurement matrices, including those with null spaces that are random orthant symmetric and generic subspaces (e.g., matrices with i.i.d. Bernoulli or uniform (-1,1) entries, etc.) [DT10]. Finally, similar to the strong thresholds, it is not known whether there exist other polynomial-time algorithms with weak thresholds superior to ℓ_1 minimization. Therefore, the problem of extending the phase transition thresholds of ℓ_1 minimization using polynomial time algorithms is a very interesting and well-motivated open problem.

6.2 Contributions

In this chapter, we prove that a certain *two-step reweighted* ℓ_1 algorithm indeed has higher weak recovery guarantees than ordinary ℓ_1 minimization for particular classes of sparse signals, including sparse Gaussian signals. We initially introduced the algorithm in [HKXA09], and proved that for a very restricted class of *polynomially decaying* sparse signals it outperforms standard ℓ_1 minimization. Later on, we extended this result to a much wider and more reasonable class of sparse signals [KXAH10]. The key to our result is the fact that for these classes of signals, ℓ_1 minimization has an *approximate support recovery* property which can be exploited in a reweighted ℓ_1 algorithm to obtain a provably superior weak threshold. In particular, we consider Gaussian sparse signals, namely sparse signals in which the nonzero entries are i.i.d. Gaussian. Our analysis of Gaussian sparse signals relies on concentration bounds on the partial sum of their order statistics. Furthermore, we show that for continuous distributions with sufficiently fast decaying tails and nonzero value at the origin, similar improvements for the weak threshold can be postulated. More generally, we show that as long as the nonzero entries of the sparse signal are independently drawn from a continuous distribution $f(\cdot)$ that has a nonzero finite-order derivative at the

origin, the weak recovery threshold of our proposed two-step reweighted ℓ_1 algorithm is strictly larger than that of ℓ_1 minimization. Although not specifically derived, our analysis suggests that the improvement rate is a function of the smallest integer r for which $f^{(r)}(0) \neq 0$; the smaller such r is, the larger the improvement is. The derivations are supported by empirical evidence, as shall be demonstrated in this chapter.

6.3 Related Work

Different variations of reweighted ℓ_1 algorithms have been recently introduced in the literature and have shown experimental improvement over ordinary ℓ_1 minimization [Nee09, CWB08]. In [Nee09], approximately sparse signals have been considered, where perfect recovery is often not achieved. The question is therefore not that of an explicit extension of phase transition curves. Instead, it has been shown that the reconstruction error can be reduced using an iterative scheme. In [CWB08], a similar algorithm is suggested and is empirically shown to outperform ℓ_1 minimization for exactly sparse signals with certain continuous distributions. In particular, it was empirically witnessed that the proposed algorithm does not improve the signal recovery for sparse vectors with constant amplitude nonzero entries (i.e., a nonzero entry is either 1 or -1). Unfortunately, [CWB08] provides no theoretical analysis or performance guarantees for the success or failure of the method. The particular reweighted ℓ_1 minimization algorithm that we propose and analyze is of significantly less computational complexity than the earlier ones (it only solves two linear programs). Furthermore, experimental results confirm that it exhibits much better performance than previous reweighted methods.

6.4 Preliminaries

We denote the support (set) of a vector $\mathbf{x}$ by $supp(\mathbf{x})$. For a vector $\mathbf{x}$ that is not necessarily k-sparse, we define the k-support of $\mathbf{x}$ to be the index set of the largest k entries of $\mathbf{x}$ in amplitude, and denote it by $supp_k(\mathbf{x})$. Finally, $\max|\mathbf{x}|$ and $\min|\mathbf{x}|$ mean the absolute value of the maximum and minimum entry of $\mathbf{x}$ in magnitude, respectively.

Although more general than this, the analysis of this chapter is for sparse random signals with i.i.d. nonzero coefficients drawn from a given continuous distribution (in particular Gaussian). In other words, we assume that the unknown sparse signal is an $n \times 1$ vector $\mathbf{x}$ with exactly k nonzero entries, where each nonzero entry is independently derived from a distribution $f(\cdot)$ (e.g., standard normal distribution $\mathcal{N}(0,1)$). When $f(\cdot)$ is a normal distribution, we refer to the resulting signals as "Gaussian sparse" vectors. As before, we denote the measurement matrix by $\mathbf{A}$ which is assumed to be an $m \times n$ matrix with i.i.d. Gaussian entries with an aspect ratio $\delta = \frac{m}{n}$. The strong and weak recovery thresholds of the required number of measurements for ℓ_1 minimization were previously defined as functions of the sparsity level. In this chapter, we find it easier to work with the thresholds of sparsity level as a function of the matrix aspect ratio. More formally, we define the strong threshold $\mu_S(\delta)$ to be the largest fraction μ such that with high probability, every μn sparse signal can be recovered from $m = \delta n$ i.i.d Gaussian measurements using ℓ_1 optimization as $n \to \infty$. Similarly, the weak threshold $\mu_W(\delta)$ is the largest fraction μ such that with high probability, almost all μn-sparse signals can be recovered from $m = \delta n$ i.i.d Gaussian measurements using ℓ_1-optimization, as $n \to \infty$. Note that the functions $\mu_S(\cdot)$ and $\mu_W(\cdot)$ are inverse functions of the thresholds $\delta_S(\cdot)$ and $\delta_W(\cdot)$ for the number of measurements defined in the previous chapter, which, in addition, were extended to the case of weighted ℓ_1 minimization. The sectional threshold $\mu_T(\cdot)$ which deals with the reconstruction of signals with all support sets and almost all sign patterns

can be characterized in a similar way. In fact, tight and easy computation for the function $\mu_W(\cdot)$ is available from the derivations of [Sto10] which conform to the less explicit formulations of [Don06b]. For strong and sectional thresholds, only lower bound approximations exist. The plot of $\mu_W(\cdot)$ is shown in Figure 6.1.

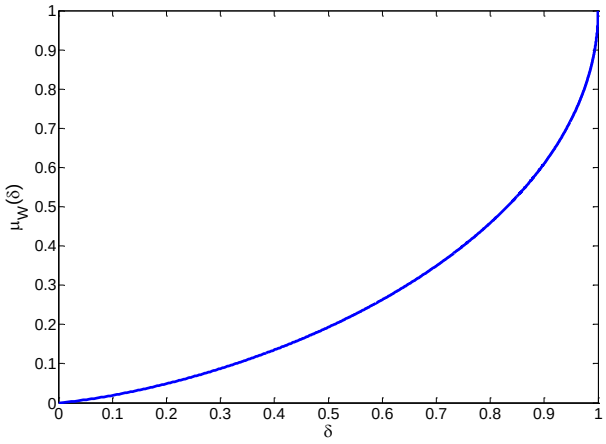


Figure 6.1: Plot of the weak recovery threshold $\mu_W(\delta)$ for ℓ_1 minimization, calculated based on the formulation of [Sto10]

Since we are interested in extending the phase transition thresholds of ℓ_1 minimization, we consider sparse signals that fall outside the recoverability regime of ℓ_1 minimization. In other words, we assume that the support size of $\mathbf{x}$, namely k, is slightly larger than the weak phase transition threshold of ℓ_1 minimization. In other words, $k = (1 + \epsilon_0)\mu_W(\delta)n$ for some $\epsilon_0 > 0$. This means that if we use ℓ_1 minimization, a randomly chosen $\mu_W(\delta)n$-sparse signal will be recovered perfectly with very high probability, whereas a randomly selected k-sparse signal will not. We would like to show that for a strictly positive ϵ_0, our proposed two-step reweighted ℓ_1 algorithm (presented in the sequel) can indeed recover a randomly selected k-sparse signal with high probability, implying that the proposed method has a superior weak threshold.

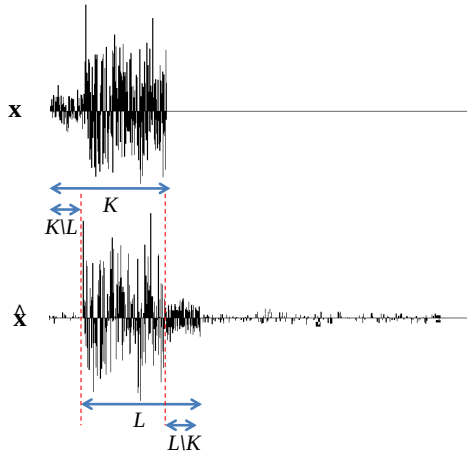


Figure 6.2: A pictorial example of a sparse signal and its ℓ_1 minimization approximation

6.5 Two-Step Weighted ℓ_1 Algorithm

We propose the following method outlined in Algorithm 4, consisting of two linear programming steps: a standard ℓ_1 minimization and a weighted one. The input to the algorithm is the vector $\mathbf{y} = \mathbf{Ax}$, where $\mathbf{x}$ is the unknown k-sparse signal with $k = (1 + \epsilon_0)\mu_W(\delta)n$, and the output is an approximation $\mathbf{x}^*$ to the unknown vector $\mathbf{x}$. We assume that the sparsity k (or an upper bound on it) is known. However, the algorithm assumes no knowledge of the distribution of the nonzero entries of the unknown signal. Also $\omega > 1$ is a predetermined weight.

Algorithm 4 —Two-Step Reweighted ℓ_1 minimization.

1: **Input:** Measurement matrix $\mathbf{A}^{m\times n}$, measurement vector $\mathbf{y}^{m\times 1}$, integer $k < n$, predetermined real valued weight $\omega > 1$.
2: **Output:** Sparse vector $\mathbf{x}$ with $\mathbf{Ax} = \mathbf{y}$.
3: Solve the ℓ_1 minimization problem:

$$\hat{\mathbf{x}} = \arg\min \|\mathbf{z}\|_1 \quad \text{subject to} \quad \mathbf{Az} = \mathbf{y}. \tag{6.5.1}$$

4: Obtain an approximation for the support set of $\mathbf{x}$: find the index set $L \subset \{1, 2, ..., n\}$ which corresponds to the largest k elements of $\hat{\mathbf{x}}$ in magnitude.
5: Solve the following weighted ℓ_1 minimization problem and declare the solution as output:

$$\mathbf{x}^* = \arg\min \|\mathbf{z}_L\|_1 + \omega\|\mathbf{z}_{\overline{L}}\|_1 \quad \text{subject to} \quad \mathbf{Az} = \mathbf{y}. \tag{6.5.2}$$

The intuition behind the algorithm is as follows. In the first step, a standard ℓ_1 minimization is performed. If the sparsity of the signal is beyond the weak threshold $\mu_W(\delta)n$, then ℓ_1 minimization is most probably not capable of recovering the signal. However, we use the output of the ℓ_1 minimization to identify an index set, L, which we "hope" contains most of the nonzero entries of $\mathbf{x}$ (see Figure 6.2). We finally perform a weighted ℓ_1 minimization by penalizing those entries of $\mathbf{x}$ that are not in L (ostensibly because they have a lower chance of being nonzero). Consequently, Algorithm 4 is capable of recovering less sparse signals, or, equivalently has a higher weak threshold than that of ℓ_1 minimization. This intuition is formalized in the following theorem.

Theorem 6.5.1 (Weak threshold of Algorithm 4)**.** *Let* $\mathbf{A}$ *be an* $m \times n$ *i.i.d. Gaussian matrix with* $\frac{m}{n} = \delta$*. There exist* $\epsilon_0 > 0$ *and* $\omega > 0$ *so that Algorithm 4 perfectly recovers a random* $(1+\epsilon_0)\mu_W(\delta)n$*-sparse vector with i.i.d. Gaussian entries with high probability as* n *grows to infinity.*

The interpretation of the above theorem is that for sparse signals whose nonzero entries follow a Gaussian distribution, Algorithm 4 has a recovery threshold beyond that of standard ℓ_1 minimization. We have empirically computed the recovery thresholds of Algorithm 4 for sparse random Gaussian signals, and plotted them along with the thresholds of regular ℓ_1 minimization in Figure 6.3. Notice the improvement in the recoverable fraction of nonzero coefficient μ.

The proof is provided in the following sections as follows. In Section 6.6, we prove that there is a large overlap between the index set L, found in step 2 of the algorithm, and the support set of the unknown signal $\mathbf{x}$ denoted by K (see Theorem 6.6.3 and Figure 6.2). Then in Section 6.7, we show that the large overlap between K and L can result in perfect recovery of $\mathbf{x}$, beyond the standard weak threshold, when a weighted ℓ_1 minimization is used in step 3. The formal proof of Theorem 6.5.1 appears in Section 6.7.

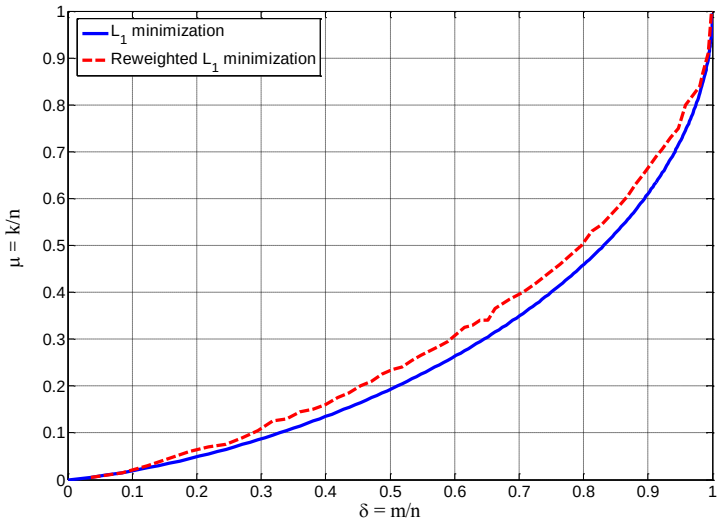


Figure 6.3: Comparison of the weak thresholds of ℓ_1 minimization with that of Algorithm 4 computed empirically for Gaussian sparse signals

6.6 Approximate Support Recovery, Steps 1 and 2 of the Algorithm

In this section, we carefully study the first two steps of Algorithm 4. The unknown signal $\mathbf{x}$ is assumed to be a Gaussian k-sparse vector with support set K, where $k = |K| = (1+\epsilon_0)\mu_W(\delta)n$, for some $\epsilon_0 > 0$. By a Gaussian k-sparse vector, we mean one where the nonzero entries are i.i.d. Gaussian (e.g., zero-mean and unit-variance). It should be noted that the Gaussian distribution is only considered as a canonical choice. We later extend our analysis to other signal distributions. The solution $\hat{\mathbf{x}}$ to the ℓ_1 minimization obtained in step 1 of Algorithm 4 is in all likelihood a dense vector. The set L, as defined in the algorithm, is the k-support set of $\hat{\mathbf{x}}$ (i.e., $L = supp_k(\hat{\mathbf{x}})$). We show that for small enough ϵ_0, the intersection of L and K is with high probability very large, so that L can be counted as a good approximation of K (Figure 6.2).

In order to find a decent lower bound on $|L \cap K|$, we point out three separate facts and establish a connection between them. First, we prove a general lemma that provides a lower bound on the quantity $|L \cap K|$ as a function of $\|\mathbf{x} - \hat{\mathbf{x}}\|_1$. Then, we

discuss a critical property of ℓ_1 minimization known as *weak robustness* which helps provide an upper bound on the quantity $\|\mathbf{x}-\hat{\mathbf{x}}\|_1$. The robustness result is due to Xu et al. and was first proved in [XH09]. However, we provide explicit scaling laws for the robustness of ℓ_1 minimization beyond the implicit results of [XH09]. Finally, we leverage some concentration results for order statistics to derive explicit formulae for the obtained bounds. These steps will be elaborated in the remainder of this section.

Definition 16. *For a k-sparse signal $\mathbf{x}$, we define $W(\mathbf{x},\lambda)$ to be the size of the largest subset of nonzero entries of $\mathbf{x}$ that has a ℓ_1 norm less than or equal to λ, i.e.,*

$$W(\mathbf{x},\lambda) \triangleq \max\{|S| \mid S \subseteq supp(\mathbf{x}),\ \|\mathbf{x}_S\|_1 \leq \lambda\}.$$

Note that $W(\mathbf{x},\lambda)$ is increasing in λ.

Lemma 6.6.1. *Let $\mathbf{x}$ be a k-sparse vector and $\hat{\mathbf{x}}$ be another vector. Also, let K be the support set of $\mathbf{x}$ and L be the k-support set of $\hat{\mathbf{x}}$. Then*

$$|K \cap L| \geq k - W(\mathbf{x}, \|\mathbf{x}-\hat{\mathbf{x}}\|_1). \tag{6.6.1}$$

Proof. Let x_i be the ith entry of $\mathbf{x}$ and $\mathbf{e}^* = (e_1, e_2, \ldots, e_n)^T$ be the solution to the following minimization problem:

$$\begin{aligned} &\text{minimize } \|\mathbf{e}\|_1 \\ &\text{s.t. } \max|(\mathbf{x}+\mathbf{e})_{K\setminus L}| \leq \min|(\mathbf{x}+\mathbf{e})_L|, \end{aligned} \tag{6.6.2}$$

where $K \setminus L$ denotes the subset of the entries of K that are not in L. Note that the vector $\hat{\mathbf{x}}-\mathbf{x}$ satisfies the constraint of the minimization problem (6.6.2). This is because $\mathbf{x}+(\hat{\mathbf{x}}-\mathbf{x}) = \hat{\mathbf{x}}$ and L is the k-support of $\hat{\mathbf{x}}$. Therefore every entry of $\hat{\mathbf{x}}$ outside the set L is smaller in amplitude than every entry inside L. Therefore, since $\mathbf{e}^*$ is the optimal solution of (6.6.2), we must have:

$$\|\mathbf{e}^*\|_1 \leq \|\hat{\mathbf{x}}-\mathbf{x}\|_1. \tag{6.6.3}$$

Let $a = \max|(\mathbf{x}+\mathbf{e}^*)_{K\setminus L}|$. Then for each $i \in K \setminus L$, using the triangular inequality we have:

$$|x_i| - |e_i| \leq |x_i + e_i| \leq a, \ \forall i \in K \setminus L, \tag{6.6.4}$$

and so:

$$|e_i| \geq \max(|x_i| - a, 0), \ \forall i \in K \setminus L. \tag{6.6.5}$$

Therefore, by summing up the inequalities in (6.6.5) for $i \in K \setminus L$, we have:

$$\|\mathbf{e}^*_{K\setminus L}\|_1 \geq \sum_{i\in K\setminus L, |x_i|>a} (|x_i| - a). \tag{6.6.6}$$

On the other hand, for all $i \in L \setminus K$ we have $|e_i| > a$, and therefore:

$$\|\mathbf{e}^*_{L\setminus K}\|_1 \geq a|L \setminus K|. \tag{6.6.7}$$

But $|L \setminus K| = |K \setminus L|$ and hence it follows that

$$\begin{aligned} \|\mathbf{e}^*\|_1 &\geq \|\mathbf{e}^*_{L\setminus K}\|_1 + \|\mathbf{e}^*_{K\setminus L}\|_1 \\ &\geq a|K \setminus L| + \sum_{i\in K\setminus L, |x_i|>a} (|x_i| - a) \\ &\geq \sum_{i\in K\setminus L} |x_i| = \|\mathbf{x}_{K\setminus L}\|_1. \end{aligned} \tag{6.6.8}$$

(6.6.3) and (6.6.8) together imply that $\|\mathbf{x}-\hat{\mathbf{x}}\|_1 \geq \|\mathbf{x}_{K\setminus L}\|_1$, which by definition means that $W(\mathbf{x}, \|\mathbf{x}-\hat{\mathbf{x}}\|_1) \geq |K \setminus L|$. ■

We now introduce the notion of weak robustness, which allows us to bound $\|\mathbf{x} - \hat{\mathbf{x}}\|_1$, and has the following formal definition [XH09].

Definition 17. *Let the set $S \subset \{1, 2, \cdots, n\}$ and the subvector $\mathbf{x}_S$ be fixed. An approximation $\hat{\mathbf{x}}$ to $\mathbf{x}$ is called weakly robust with respect to the set S if, for some $C_S > 1$, it holds that*

$$\|(\mathbf{x} - \hat{\mathbf{x}})_S\|_1 \leq \frac{2C_S}{C_S - 1}\|\mathbf{x}_{\overline{S}}\|_1, \tag{6.6.9}$$

and

$$\|\mathbf{x}_S\| - \|\hat{\mathbf{x}}_S\| \leq \frac{2}{C_S - 1}\|\mathbf{x}_{\overline{S}}\|_1. \tag{6.6.10}$$

C_S is called the robustness parameter of the considered approximation for the set S.

The weak robustness notion allows us to bound the error in $\|\mathbf{x}-\hat{\mathbf{x}}\|_1$ in the following way. If $\hat{\mathbf{x}}$ is a weakly robust approximation to $\mathbf{x}$ with respect to the set S and parameter $C_S > 1$, such that $\mathbf{A}\mathbf{x} = \mathbf{A}\hat{\mathbf{x}}$, and if the matrix $\mathbf{A}_S$ obtained by retaining only those columns of $\mathbf{A}$ that are indexed by S has full column rank, then the quantity

$$\kappa = \max_{\mathbf{A}\mathbf{w}=0,\mathbf{w}\neq 0} \frac{\|\mathbf{w}_S\|_1}{\|\mathbf{w}_{\overline{S}}\|_1},$$

must be finite, and one can conclude that

$$\|\mathbf{x}-\hat{\mathbf{x}}\|_1 \leq \frac{2C_S(1+\kappa)}{C_S-1}\|\mathbf{x}_{\overline{S}}\|_1. \tag{6.6.11}$$

This result is due to [XH09], where, in addition, it has been shown that for Gaussian i.i.d. measurement matrices $\mathbf{A}$, the solution of ℓ_1 minimization provides a weakly robust approximation with high probability. In other words, for a randomly chosen subset S with $\frac{|S|}{n} < \mu_W(\delta)$, there exists a robustness factor $C_S > 1$ as a function of $\frac{|S|}{n}$ for which (6.6.9) and (6.6.10) hold with high probability for an arbitrary vector $\mathbf{x}$, where $\hat{\mathbf{x}}$ is the solution obtained by ℓ_1 minimization. Now let $k_1 = (1-\epsilon_1)\mu_W(\delta)n$ for some small $\epsilon_1 > 0$, and K_1 be the k_1-support set of $\mathbf{x}$, namely, the set of the largest k_1 entries of $\mathbf{x}$ in magnitude. Based on equation (7.4.4) we may write

$$\|\mathbf{x}-\hat{\mathbf{x}}\|_1 \leq \frac{2C(\epsilon_1)(1+\kappa)}{C(\epsilon_1)-1}\|\mathbf{x}_{\overline{K_1}}\|_1, \tag{6.6.12}$$

where for a fixed value of δ, we have emphasized that the constant C for the set K_1 is a function of ϵ_1. Furthermore, $C(\epsilon_1)$ becomes arbitrarily close to 1 as $\epsilon_1 \to 0$. κ is also a bounded function of ϵ_1, and therefore we may replace it with an upper bound κ^*. This provides a bound on $\|\mathbf{x}-\hat{\mathbf{x}}\|_1$. To explore this inequality and understand its asymptotic behavior, we apply a third result, which is a certain concentration bound on the order statistics of Gaussian random variables.

Lemma 6.6.2. *Suppose $X_1, X_2, \cdots, X_N$ are N i.i.d. $\mathcal{N}(0,1)$ random variables. Let $S_N = \sum_{i=1}^{N} |X_i|$ and let S_M be the sum of the largest M numbers among the $|X_i|$s, for each $1 \leq M < N$. Then for every $\epsilon > 0$ sufficiently small, as $N \to \infty$, if the ratio*

M/N is kept constant, we have

$$\mathbb{P}\left(\left|\frac{S_N}{N}-\sqrt{\frac{2}{\pi}}\right|>\epsilon\right)\to 0, \tag{6.6.13}$$

$$\mathbb{P}\left(\left|\frac{S_M}{S_N}-\exp(-\frac{\Psi^2(\frac{M}{2N})}{2})\right|>\epsilon\right)\to 0, \tag{6.6.14}$$

where $\Psi(x)=Q^{-1}(x)$ with $Q(x)=\frac{1}{\sqrt{2\pi}}\int_x^\infty e^{-\frac{y^2}{2}}dy$.

To make the content of the chapter more fluent, we mention the general idea of the proof of the above lemma very coarsely here. The detailed proof is outlined in Section 6.10. For a particular instance of $X_1,\ldots,X_N$, if $0<a<1$ is such that exactly a fraction M/N of $|X_i|$s are larger than a, then every $|X_i|$ which is larger than a contributes to the sum S_M. Therefore S_M can be thought of as those $|X_i|$s that are larger than a. This can be expressed in another way: Let $\hat{X}_i$ be a random variable which is equal to $|X_i|$ if $|X_i|>a$, and is 0 otherwise. We therefore conclude that S_M is equal to the sum of $\sum_{i=1}^n \hat{X}_i$. Furthermore, when N is large, it can be shown using concentration lemmas that a will be arbitrarily close to the fixed number $\Psi(\frac{M}{2N})$, and thus the distributions of $\hat{X}_i$s converge to the same distribution, namely the truncated absolute value of a normal distribution. Besides, when a is constant, $\hat{X}_i$s are independent, and therefore one can apply the law of large numbers to conclude that $S_M/S_N\approx \mathbb{E}\hat{X}_1/\mathbb{E}|X_1|$, which is the desired conclusion. These arguments are rigorously outlined in Section 6.10.

Recall that we assumed $\mathbf{x}$ is a k-sparse random Gaussian signal with $k=(1+\epsilon_0)\mu_W(\delta)n$, and defined K_1 to be the k_1-support of $\mathbf{x}$, where $k_1=(1-\epsilon_1)\mu_W(\delta)n$. We denoted by K the support set of $\mathbf{x}$. Also, if $\hat{\mathbf{x}}$ is the approximation to $\mathbf{x}$ obtained by ℓ_1 minimization, we denoted by L the k-support set of $\hat{\mathbf{x}}$. As a direct consequence of Lemma 6.6.2 we can write:

$$\mathbb{P}\left(\left|\frac{\|\mathbf{x}_{\overline{K_1}}\|_1}{\|\mathbf{x}\|_1}-(1-e^{-0.5\Psi^2(0.5\frac{1-\epsilon_1}{1+\epsilon_0})})\right|>\epsilon\right)\to 0, \tag{6.6.15}$$

for $\epsilon > 0$ sufficiently small as $n \to \infty$. Define

$$\zeta(\epsilon_0) \triangleq \inf_{\epsilon_1 > 0} \frac{2C(\epsilon_1)(1+\kappa^*)}{C(\epsilon_1)-1}(1 - e^{-0.5\Psi^2(0.5\frac{1-\epsilon_1}{1+\epsilon_0})}). \tag{6.6.16}$$

Incorporating (6.6.12) into (6.6.15) we may write

$$\mathbb{P}\left(\frac{\|\mathbf{x}-\hat{\mathbf{x}}\|_1}{\|\mathbf{x}\|_1} - \zeta(\epsilon_0) < \epsilon\right) \to 1, \tag{6.6.17}$$

for $\epsilon > 0$ sufficiently small as $n \to \infty$.

Let us summarize our conclusions so far. First, we were able to show that $|K \cap L| \geq k - W(\mathbf{x}, \|\mathbf{x}-\hat{\mathbf{x}}\|_1)$. The weak robustness of ℓ_1 minimization and the Gaussian nature of the signal then led us to the fact that for large n with high probability, $\|\mathbf{x}-\hat{\mathbf{x}}\|_1 \leq \zeta(\epsilon_0)\|\mathbf{x}\|_1$. These results build up the next key theorem, which is the conclusion of this section.

Theorem 6.6.3 (Approximate Support Recovery)**.** *Let* $\mathbf{A}$ *be an i.i.d. Gaussian* $m \times n$ *measurement matrix with* $\frac{m}{n} = \delta$*. Let* $k = (1+\epsilon_0)\mu_W(\delta)$ *and* $\mathbf{x}$ *be an* $n \times 1$ *random Gaussian* k*-sparse signal. Suppose that* $\hat{\mathbf{x}}$ *is the approximation to* $\mathbf{x}$ *given by* ℓ_1 *minimization, i.e.,* $\hat{\mathbf{x}} = argmin_{\mathbf{Az}=\mathbf{Ax}}\|\mathbf{z}\|_1$*. Then, as* $n \to \infty$*, for all* $\epsilon > 0$*,*

$$\mathbb{P}\left(\frac{|supp(\mathbf{x}) \cap supp_k(\hat{\mathbf{x}})|}{k} - 2Q(\sqrt{-2\log(1-\zeta(\epsilon_0))}) > -\epsilon\right) \to 1, \tag{6.6.18}$$

where $\zeta(\cdot)$ *is defined in (6.6.16).*

Before proving the above theorem, we mention the following useful lemma, the proof of which will be given in Section 6.10.

Lemma 6.6.4. *Let* $\mathbf{x}$ *be a random* k*-sparse Gaussian vector of size* n*, and* $0 < \alpha < 1$*. For any positive* ϵ*, the following happens with high probability as* $n, k \to \infty$*:*

$$\frac{W(\mathbf{x}, \alpha\|\mathbf{x}\|_1)}{k} < (1 - 2Q(\sqrt{-2\log(1-\alpha)})) + \epsilon. \tag{6.6.19}$$

Proof of Theorem 6.6.3. From equation (6.8.3), for every $\epsilon' > 0$ and large enough n, with high probability we have $\|\mathbf{x} - \hat{\mathbf{x}}\|_1 < (\zeta(\epsilon_0) + \epsilon')\|\mathbf{x}\|_1$. Therefore, from Lemma 6.6.1 and the fact that $W(\mathbf{x}, \lambda)$ is increasing in λ, $|K \cap L| \geq k - W(\mathbf{x}, (\zeta(\epsilon_0) + \epsilon')\|\mathbf{x}\|_1)$ with high probability. Replacing $W(\mathbf{x}, (\zeta(\epsilon_0) + \epsilon'))$ with the upper bound given by Lemma 6.6.4, it follows that with very high probability $\frac{|K \cap L|}{k} \geq 2Q(\sqrt{-2\log(1 - \zeta(\epsilon_0) - \epsilon')}) - \epsilon''$. We can now let ϵ' go to zero and the proof is completed. ■

Note that if $\lim_{\epsilon_0 \to 0} \zeta(\epsilon_0) = 0$, then Theorem 6.6.3 implies that $\frac{|K \cap L|}{k}$ becomes arbitrarily close to 1, which means that, using ℓ_1 minimization, it is possible to closely estimate the support set of $\mathbf{x}$. We show in the sequel that this is in fact the case.

6.6.1 Scaling Law of ℓ_1 Minimization

In order to show that the robust approximation of the sparse signal at step 1 of Algorithm 4 leads to perfect recovery at step 3, we need to obtain an explicit bound for the term $\zeta(\epsilon_0)$. This in turn requires calculating a solid relationship between the robustness parameter $C(\epsilon_1)$, and the back-off fraction ϵ_1. For i.i.d. Gaussian matrices, we derive an explicit lower bound on $C(\epsilon_1)$ as a function of ϵ_1 through the following theorem.

Theorem 6.6.5 (Scaling law of ℓ_1 minimization for Gaussians)**.** *Let* $\mathbf{A}$ *be an* $m \times n$ *i.i.d. Gaussian matrix with* $m = \delta n$*, and* $\mu_W(\delta)$ *be the weak recovery threshold of* ℓ_1 *minimization for* A*. For sufficiently large* n*, the (weak) robustness parameter* $C(\epsilon_1)$ *for a randomly chosen* k_1*-support* K_1 *of size* $k_1 = (1 - \epsilon_1)\mu_W(\delta)n$ *(see equation 6.6.12) satisfies:*

$$C(\epsilon_1) \geq \frac{1}{\sqrt{1 - \epsilon_1}}. \tag{6.6.20}$$

The proof of of the above theorem which we have presented in [KWYH12] is common for the most part with the technical details of [XH09], which are based on

Grassman manifold techniques for the performance analysis of compressed sensing. The method is basically the extension of the high-dimensional techniques of Donoho et al. [DT05a, DT06] for incorporating noise into the performance bounds of ℓ_1 minimization.

We now derive an asymptotic upper bound on the term $\zeta(\epsilon_0)$ using the above relationship. Replacing the bound of (6.6.20) in the definition of $\zeta(\epsilon_0)$, we obtain:

$$\begin{aligned}
\zeta(\epsilon_0) &= \inf_{\epsilon_1>0} \frac{2C(\epsilon_1)(1+\kappa^*)}{C(\epsilon_1)-1}\left(1-e^{-0.5\Psi^2(0.5\frac{1-\epsilon_1}{1+\epsilon_0})}\right) \\
&\leq \inf_{\epsilon_1>0} \frac{2(1+\kappa^*)}{1-\sqrt{1-\epsilon_1}}\left(1-e^{-0.5\Psi^2(0.5\frac{1-\epsilon_1}{1+\epsilon_0})}\right) \qquad (6.6.21) \\
&\leq \frac{4(1+\kappa^*)}{\epsilon_0}\left(1-e^{-0.5\Psi^2(0.5\frac{1-\epsilon_0}{1+\epsilon_0})}\right), \qquad (6.6.22)
\end{aligned}$$

where (6.6.22) is obtained by simply taking $\epsilon_1 = \epsilon_0$, and using the fact that $\frac{1}{1-\sqrt{1-\epsilon_0}} \leq 2/\epsilon_0$. We use the Taylor approximation of the inverse error function to bound the right-hand side of (6.6.22). Note that:

$$\begin{aligned}
\Psi(0.5\frac{1-\epsilon_0}{1+\epsilon_0}) &= \sqrt{2}\cdot \mathrm{erf}^{-1}\left(\frac{2\epsilon_0}{1+\epsilon_0}\right) \qquad (6.6.23) \\
&= \sqrt{2\pi}\cdot\epsilon_0 + o(\epsilon_0^2). \qquad (6.6.24)
\end{aligned}$$

It follows that:

$$\zeta(\epsilon_0) \leq 4\pi(1+\kappa^*)\epsilon_0 + \mathcal{O}(\epsilon_0^2), \qquad (6.6.25)$$

as $\epsilon_0 \to 0$. Therefore, we can immediately see that $\lim_{\epsilon_0\to 0}\zeta(\epsilon_0) = 0$.

6.7 Perfect Recovery, Step 3 of the Algorithm

In Section 6.6 we showed that if ϵ_0 is small, the k-support of $\hat{\mathbf{x}}$, namely $L = supp_k(\hat{\mathbf{x}})$, has a significant overlap with the true support of $\mathbf{x}$. We even found a quantitative lower bound on the size of this overlap in Theorem 6.6.3. In step 3 of Algorithm 4, weighted ℓ_1 minimization is used, where the entries in $\overline{L}$ are assigned a higher weight than those in L. Interestingly, this fits into the model described in Chapter 5. Recall

that in Chapter 5, we were able to analyze the performance of such weighted ℓ_1 minimization algorithms in the presence of a nonuniform sparsity assumption. The idea was that if a sparse vector $\mathbf{x}$ can be partitioned into two sets L and $\overline{L}$, where the fraction p_1 of nonzero entries in L is much larger than the fraction p_2 of nonzero entries in $\overline{L}$, then (6.5.2) can potentially recover $\mathbf{x}$ with an appropriate choice of the weight $\omega > 1$, even though ℓ_1 minimization cannot. Recall the notion of sectional threshold $\delta_c^{(T)}(\gamma_1, \gamma_2, p_1, p_2, \omega)$ from Chapter 5. If $\mathbf{x}$ is a vector with nonuniform sparsity fractions p_1 and p_2 over the classes of entries L and $\overline{L}$, respectively, when $|L_1| = \gamma_1 n$ and $|\overline{L}| = \gamma_2 n$ then the weighted ℓ_1 minimization of (6.5.2) will recover $\mathbf{x}$ with high probability. The choice of "sectional" threshold here dictates that the successful recovery is guaranteed for all possible support sets for $\mathbf{x}$ (and not just a randomly chosen set), but for almost all sign patterns (i.e., for a randomly chosen pattern). This is consistent with the reliability information extracted from the preliminary ℓ_1 minimization in the first stage of the algorithm; Despite the fact that the support set of $\mathbf{x}$ was chosen randomly, given the classification of entries $L, \overline{L}$ provided by the ℓ_1 approximation, and assuming sparsity fractions p_1 and p_2, the support set of $\mathbf{x}$ is not uniformly distributed over all possible index sets K that share an overlap fraction p_1 with L and an overlap fraction p_2 with $\overline{L}$. Therefore, to analyze the success of the weighted ℓ_1 minimization, a worst-case study over all possible support sets has to be done, which is accounted for in the definition of sectional threshold. Consequently, when the sparsity fractions p_1 and p_2 are known to be bounded by:

$$p_1 \geq f_1, \ p_2 \leq f_2, \tag{6.7.1}$$

the required number of measurements that guarantees successful recovery of the signal in the last stage is given by:

$$\delta \geq \lambda_c(\frac{k}{n}, 1 - \frac{k}{n}, f_1, f_2, \omega), \tag{6.7.2}$$

where

$$\lambda_c(\gamma_1, \gamma_2, f_1, f_2, \omega) \triangleq \max_{f_1' \geq f_1, f_1'\gamma_1 + f_2'\gamma_2 = f_1\gamma_1 + f_2\gamma 2} \delta_c^{(T)}(\gamma_1, \gamma_2, f_1', f_2', \omega). \tag{6.7.3}$$

Consequently, the following theorem can be deduced:

Theorem 6.7.1. *Let $L \subset \{1, 2, \cdots, n\}$, $\omega > 1$, and the fractions $f_1, f_2 \in [0, 1]$ be given. Let $\gamma_1 = \frac{|L|}{n}$ and $\gamma_2 = 1 - \gamma_1$. There exists a threshold $\lambda_c(\gamma_1, \gamma_2, f_1, f_2, \omega)$ given by equation (6.7.3) such that with high probability, almost all random sparse vectors $\mathbf{x}$ with* at least *$f_1\gamma_1 n$ nonzero entries over the set L, and* at most *$f_2\gamma_2 n$ nonzero entries over the set $\overline{L}$ can be perfectly recovered using $\min_{\mathbf{Az}=\mathbf{Ax}} \|\mathbf{z}_L\|_1 + \omega\|\mathbf{z}_{\overline{L}}\|_1$, where $\mathbf{A}$ is a $\lambda_c n \times n$ matrix with i.i.d. Gaussian entries.*

We are now in a position to state the full proof of Theorem 6.5.1.

Proof of Theorem 6.5.1. Recall that the solution of ℓ_1 minimization in the first stage of Algorithm 4 is the vector $\hat{\mathbf{x}}$. We denoted by L the k-support set of $\hat{\mathbf{x}}$, and by L^c its complement set. The last stage of the algorithm is a weighted ℓ_1 minimization that puts more weight on the entries of $\mathbf{x}$ outside the set L. The justification for this is the fact that the fraction of the nonzero entries of the target signal $\mathbf{x}$ over the set L is supposedly larger than the fraction of the nonzero entries over L^c. Let us denote these fractions by p_1 and p_2, respectively, namely $p_1 = \frac{|L \cap K|}{|L|}$ and $p_2 = \frac{|\overline{L} \cap K|}{|\overline{L}|}$, where K is the support of the target signal, unknown to the algorithm before running the weighted ℓ_1 minimization of the last stage. Suppose we know the bounds:

$$p_1 \geq f_1, \ p_2 \leq f_2. \tag{6.7.4}$$

Since we are using a weighted ℓ_1 minimization, $\mathbf{x}$ will be recovered perfectly with high probability if the number of measurements is larger than the threshold of weighted ℓ_1 minimization for the nonuniform sparsity model of the target signal, namely if:

$$\lambda_c(\frac{k}{n}, 1 - \frac{k}{n}, f_1, f_2, \omega) \leq \delta, \tag{6.7.5}$$

where λ_c is defined as in (6.7.3). This is directly implied by Theorem 6.7.1. On the other hand, through Theorem 6.6.3, we provided a explicit value for the lower bound f_1 (and consequently the upper bound f_2), and we showed that as $\epsilon_0 \to 0$, f_1 converges to 1 (and consequently f_2 approaches zero). The asymptotic value of $\lambda_c(\frac{k}{n}, 1-\frac{k}{n}, f_1, f_2, \omega)$ will therefore be equal to $\lambda_c(\mu_W(\delta), 1-\mu_W(\delta), 1, 0, \omega)$, as $\epsilon \to 0$ (recall that $k = (1+\epsilon_0)\mu_W(\delta)n$). Furthermore, from the computations of Chapter 5, it can be shown that $\lambda_c(\mu_W(\delta), 1-\mu_W(\delta), 1, 0, \omega) < \delta$ for an appropriate choice of $\omega > 1$, and that for a fixed ω the function $\lambda_c(\gamma_1, \gamma_2, f_1, f_2)$ is a continuous function of γ_1, f_1, and f_2. Furthermore, k, f_1, and f_2 obtained from Theorem 6.6.3 are all continuous functions of ϵ_0 in this case. Therefore, we can conclude that for a strictly positive ϵ_0 and corresponding overlap fractions f_1 and f_2, $\lambda_c((1+\epsilon_0)\mu_W(\delta), 1-(1+\epsilon_0)\mu_W(\delta), f_1, f_2, \omega) < \delta$. This means that for some strictly positive ϵ_0 the number of measurements that is required to reconstruct the signal precisely in the last stage of the algorithm is less than the number of measurements in $\mathbf{A}$, i.e., $\mathbf{x}$ will be recovered with high probability, despite the fact that it has more nonzero entries than the weak threshold of ℓ_1 minimization. This completes the proof. ■

6.8 Generalization to Beyond Gaussians

The theoretical threshold improvement of the proposed iterative ℓ_1 minimization algorithm was demonstrated for the case of i.i.d. Gaussian matrices, and sparse vectors with independent Gaussian nonzero entries. It is reasonable to ask if we can extend these results to sparse signals with other distributions. We address this problem in this section. In summary, we prove that the theoretical threshold improvement can be generalized to sparse signals whose nonzero entries obey a more general class of distributions, namely continuous symmetric distributions with a non-vanishing finite order derivative at the origin. This is outlined in the following section.

6.8.1 Arbitrary Distributions

The attentive reader will note that the only step where we used the Gaussian nature of the signal in the proof of threshold improvement was in the the order statistics results of Lemma 6.6.2. This result has the following interpretation: For N i.i.d. random variables, the ratio $\frac{S_M}{S_N}$ can be approximated by a known function of $\frac{M}{N}$. In the Gaussian case, this function behaves as $1-(1-\frac{M}{N})^2$ as $M \to N$. For constant magnitude signals (say BPSK), the function behaves as $1-\frac{M}{N}$, for $M \to N$, which predicts that the reweighted method yields no improvement. A more careful analysis reveals that the improvement over ℓ_1 minimization depends on the behavior of $\frac{S_M}{S_N}$, as $M \to N$, which in turn depends on the smallest order n for which $f^{(n)}(0) \neq 0$, i.e., the smallest n such that the n-th derivative of the distribution at the origin is nonzero. We formalize these results by generalizing the arguments of the previous section. First, we present a generalization of Lemma 6.6.2 for arbitrary symmetric distributions.

Lemma 6.8.1. *Suppose $X, X_1, X_2, \cdots, X_n$ are N i.i.d. random variables, drawn from a symmetric distribution $f(\cdot)$. Let $S_N = \sum_{i=1}^{N} |X_i|$ and let S_M be the sum of the largest M numbers among $|X_i|$s, for each $1 \leq M < N$. If $f(\cdot)$ is integrable, and if for every finite $a > 0$, the integral $\int_a^\infty x^2 f(x)dx$ is finite, then for every $\epsilon > 0$ sufficiently small, as $N \to \infty$ and the ratio M/N is kept constant, the following holds*

$$\mathbb{P}\left(\left|\frac{S_M}{S_N} - (1 - 2\frac{\int_0^{\Psi_f(\frac{M}{2N})} x \cdot f(x)dx}{\mathbb{E}_{f(\cdot)}|X|})\right| > \epsilon\right) \to 0, \tag{6.8.1}$$

where $\Psi_f(x) = Q_f^{-1}(x)$ with $Q_f(x) = \int_x^\infty f(y)dy$.

Using the above lemma, we can modify the concentration term of equation (6.6.15) for the term $\frac{\|\mathbf{x}_{\overline{K_1}}\|_1}{\|\mathbf{x}\|_1}$, where the distribution of the nonzero entries of $\mathbf{x}$ is $f(\cdot)$. The resulting concentration thus becomes:

$$\mathbb{P}\left(\left|\frac{\|\mathbf{x}_{\overline{K_1}}\|_1}{\|\mathbf{x}\|_1} - 2\frac{\int_0^{\Psi_f(\frac{(1-\epsilon_1)}{2(1+\epsilon_0)})} x \cdot f(x)dx}{\mathbb{E}_{f(\cdot)}|X|}\right| > \epsilon\right) \to 0, \tag{6.8.2}$$

which, when put together with the bound of (6.6.12), results in:[1]

$$\mathbb{P}\left(\frac{\|\mathbf{x}-\hat{\mathbf{x}}\|_1}{\|\mathbf{x}\|_1}-\zeta_f(\epsilon_0)<\epsilon\right)\to 1, \tag{6.8.3}$$

for every $\epsilon>0$. Here $\zeta_f(\epsilon_0)$ is defined by:

$$\zeta_f(\epsilon_0)\triangleq\inf_{\epsilon_1>0}\frac{2C(\epsilon_1)(1+\kappa^*)}{C(\epsilon_1)-1}\times 2\frac{\int_0^{\Psi_f(\frac{(1-\epsilon_1)}{2(1+\epsilon_0)})}x\cdot f(x)dx}{\mathbb{E}_{f(\cdot)}|X|}. \tag{6.8.4}$$

Consequently, following similar arguments as in the proofs of Theorem 6.6.3, we can state the following theorem as a generalization of the approximate support recovery of ℓ_1 minimization for arbitrary distributions, the proof of which is immediate.

Theorem 6.8.2 (Approximate Support Recovery/Generalization)**.** *Let* $\mathbf{A}$ *be an i.i.d. Gaussian* $m\times n$ *measurement matrix with* $\frac{m}{n}=\delta$*. Let* $k=(1+\epsilon_0)\mu_W(\delta)n$ *and* $\mathbf{x}$ *be an* $n\times 1$ *k-sparse signal whose nonzero entries are independently drawn from a distribution* $f(\cdot)$ *which satisfies the conditions of Lemma 6.8.1. Suppose that* $\hat{\mathbf{x}}$ *is the approximation to* $\mathbf{x}$ *given by the* ℓ_1 *minimization, i.e.,* $\hat{\mathbf{x}}=argmin_{\mathbf{Az}=\mathbf{Ax}}\|\mathbf{z}\|_1$*. Then, as* $n\to\infty$*, for* $\epsilon>0$ *sufficiently small, we have*

$$\mathbb{P}\left(\frac{|supp(\mathbf{x})\cap supp_k(\hat{\mathbf{x}})|}{k}-2Q_f(\sqrt{-2\log(1-\zeta_f(\epsilon_0))})>-\epsilon\right)\to 1, \tag{6.8.5}$$

where $\zeta_f(\cdot)$ *is defined in (6.8.4).*

Note that $Q_f(\cdot)$ is always a decreasing function which is equal to zero at the origin for symmetric distributions. Therefore, the overlap fraction given by Theorem 6.8.2 can be arbitrarily close to 1, provided that $\zeta_f(\epsilon_0)$ is sufficiently small. Therefore, the key in further conclusions on the above bound is to derive a bound on the term $\zeta_f(\epsilon_0)$, and show that it becomes arbitrarily small. We prove that this is in fact the case for distributions $f(\cdot)$ for which one of the finite order derivatives at the origin is nonzero, stated formally in the following lemma:

[1]Note that the bound in (6.6.12) is independent from the distribution of $\mathbf{x}$

Lemma 6.8.3. *Let $f(\cdot)$ be a symmetric distribution which satisfies the conditions of Lemma 6.8.1. If for some integer $r \geq 0$, the r'th order derivative of $f(\cdot)$ at origin exists and does not vanish, i.e., $f^{(r)}(0) \neq 0$, then $\zeta_f(\epsilon_0) = \mathcal{O}(\epsilon_0^{1/(r+1)})$ as $\epsilon_0 \to 0$. Consequently, the support set approximation of ℓ_1 minimization is asymptotically perfect with high probability as $\epsilon_0 \to 0$.*

Proof. For simplicity, we take ϵ_1 in the definition of $\zeta_f(\epsilon_0)$ to be equal to ϵ_0, which only provides an upper bound. Since $f^{(r)}(0) > 0$ and $f(\cdot)$ is continuous, we conclude that for some constant $c > 0$, and sufficiently small x, $f(x) \geq c \times x^r$. Therefore,

$$1/2 - Q_f(x) = \int_0^x f(t)dt \geq \frac{c}{r+1} x^{r+1}, \tag{6.8.6}$$

and thus

$$x \geq \Psi_f(1/2 - \frac{c}{r+1} x^{r+1}), \tag{6.8.7}$$

for sufficiently small x. Note that we have used the fact that $\Psi_f(\cdot)$ is a decreasing function. Equivalently, (6.8.7) means that

$$\Psi_f(1/2 - x) = \mathcal{O}(x^{1/(r+1)}) \tag{6.8.8}$$

as $x \to 0$. On the other hand, note that $\frac{1-\epsilon_0}{2(1+\epsilon_0)} \geq 1/2 - \epsilon_0$, and thus

$$\Psi_f(\frac{1-\epsilon_0}{2(1+\epsilon_0)}) \leq \Psi_f(1/2 - \epsilon_0). \tag{6.8.9}$$

It follows from the above, (6.8.8), and the fact that $f(x) = \mathcal{O}(x^r)$ as $x \to 0$ that

$$\int_0^{\Psi_f(\frac{(1-\epsilon_0)}{2(1+\epsilon_0)})} x \cdot f(x)dx = \mathcal{O}(\epsilon_0^{1+1/(r+1)}), \tag{6.8.10}$$

as $\epsilon_0 \to 0$. Furthermore, from Theorem 6.6.5, we know that $C(\epsilon_1) \geq 1/\sqrt{1-\epsilon_0}$ (note that $\epsilon_1 = \epsilon_0$), and therefore $\frac{2C(\epsilon_1)(1+\kappa^*)}{C(\epsilon_1)-1} = \mathcal{O}(1/\epsilon_0)$ as $\epsilon_0 \to 0$. Also, $\mathbb{E}_{f(\cdot)}|X| > 0$ is constant. Therefore, from these conclusions and the definition of $\zeta_f(\cdot)$, it follows that $\zeta_f(\epsilon_0) = \mathcal{O}(\epsilon_0^{1/(r+1)})$ as $\epsilon_0 \to 0$.

■

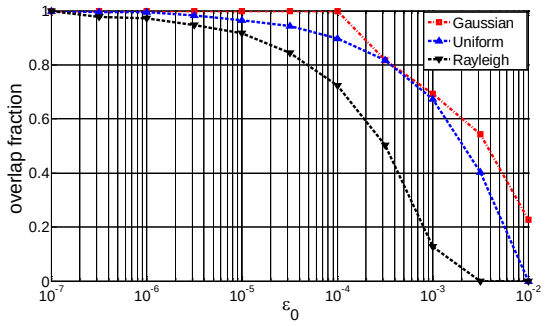


Figure 6.4: Theoretical lower bound on the correct support estimation of ℓ_1 minimization, as a function of the weak threshold exceeding fraction ϵ_0. The plots are based on the theoretical results of Theorem 6.8.2, and are derived for Gaussian, uniform, and two-sided Rayleigh distributions.

As a numerical example, we compute a theoretical bound for the approximate support recovery of ℓ_1 minimization and threshold improvement in the case of $\delta = 0.5555$. It is easy to verify numerically that the conditions of Theorem 6.5.1 hold. The value of κ^* is no more than $\sqrt{3}$ in this case. A theoretical bound on the overlap fraction between the k-support set of $\hat{\mathbf{x}}$ and the support set of the k-sparse $\mathbf{x}$ for an arbitrary distribution is provided by Theorem 6.8.2, where $k = (1 + \epsilon_0)\mu_W(\delta)n$. We have computed this bound for three different distributions: Gaussian, uniform (-1,1), and a two-sided Rayleigh distribution. The value of r, namely the smallest nonzero derivative order, is 0 for Gaussian and uniform distributions, and 1 for the Rayleigh distribution. The computed bounds are plotted in Figure 6.4. Furthermore, using a value of $\omega = 10$, and based on the premise of Theorem 6.5.1 and the computed bounds, we can certify an improvement of $\epsilon_0 = 5 \times 10^{-4}$ in the weak recovery threshold in the case of Gaussian distribution. For the uniform and Rayleigh distributions, the theoretical predictions in the improvement of recovery thresholds are smaller than the case of Gaussian, but are still strictly positive. These improvement guarantees are of course much smaller than the practical values we would observe in practice, as will be illustrated in the following section.

6.9 Simulations

We demonstrate the validity of the theoretical results of the previous sections, and the performance of Algorithm 4 by a few numerical simulations. The purpose of the simulations of this section is both to evaluate the performance of the proposed reweighted ℓ_1 algorithm in practice, and to verify its distribution-dependent behavior. Figure 6.5 shows the empirical performance of Algorithm 4 for sparse signals with various distributions. Here the signal dimension is $n = 200$, and the number of measurements is $m = 112$, which corresponds to a value of $\delta = 0.5555$. We generated random sparse signals with i.i.d. entries coming from certain distributions, namely Gaussian, uniform, Rayleigh, square root of χ-square with 4 degrees of freedom, and square root of χ-square with 6 degrees of freedom. All of these distributions are continuous and have some finite-order non-vanishing derivative at the origin. In fact, in an increasing order of the mentioned distributions, the smallest order of nonzero derivative at the origin varies from 0 to 3. In other words, the pdf of a Gaussian and a uniform $(-1, 1)$ distribution is nonzero at 0. The pdf of the Rayleigh distribution is zero at the origin, but has a nonzero derivative. Finally, the pdfs of square root of a χ-square with 4 and 6 degrees of freedom have second and third nonzero derivatives at the origin, respectively. In Figure 6.5, solid lines represent the simulation results for ordinary ℓ_1 minimization, and different colors indicate different distributions. Dashed lines are used to show the results for Algorithm 4. Notice that the more derivatives that vanish at the origin, the less significant improvement over ℓ_1 minimization is observed, which is consistent with the analysis of Section 6.8. The Gaussian and uniform distributions are flat and nonzero at the origin and show an impressive improvement (more than 20%) in the weak threshold (from 45 to 55 in this case).

In Figure 6.6, the overlap between the support set of a k-sparse signal $\mathbf{x}$ and the k-support set of the approximation $\hat{\mathbf{x}}$ given by ℓ_1 minimization averaged over 400

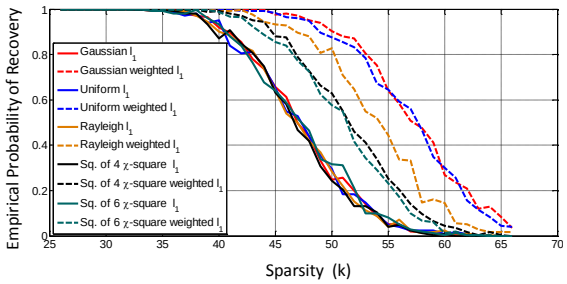


Figure 6.5: Empirical recovery percentage for $n = 200$ and $\delta = 0.5555$

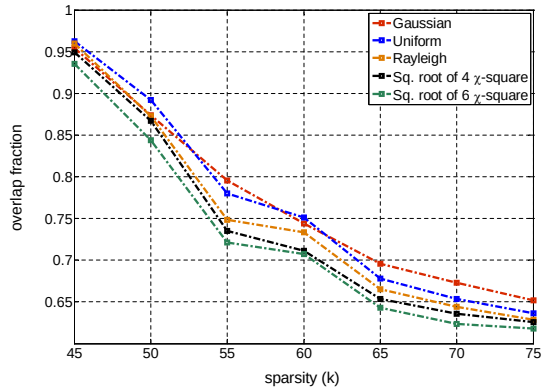


Figure 6.6: Empirical overlap between the support set of a k-sparse vector and the k-support set of the ℓ_1 optimum, for $n = 200$ and $\delta = 0.5555$. Nonzero coefficients of signal are drawn from five different distributions (displayed). The average is over 400 samples.

random samples is plotted. Again, five different distributions were considered. It is apparent that overlap fraction is a decreasing function of k, and depends on the smoothness of the probability distribution at origin.

We also report experimental results using regular ℓ_1 and reweighted ℓ_1 minimization recovery algorithms over real-world data. We have chosen a pair of satellite images (Figure 6.7) taken in two different years, 1989 (left) and 2000 (right), from the New Britain rainforest in Papua New Guinea. The images belong to the Royal Society for the Protection of Birds and were taken from an article on deforestation in the Guardian archive. These images are generally recorded to evaluate environmental effects such as deforestation. The difference between images taken at different times is generally not very significant, and thus can be thought of as compressible. We have

Figure 6.7: Satellite images taken from the New Britain rainforest in Papua Guina at 1989 (left) and 2000 (right). Red boxes identify the subframe used for the experiment, and green boxes identify the regions with higher associated weight in the weighted ℓ_1 recovery. Image belongs to the Royal Society for the Protection of Birds and was taken from an article on deforestation in the Guardian archive [sta].

applied ℓ_1 minimization to recover the difference image over the subframe (subset of the original images) identified by the red rectangles in Figure 6.7. In addition, we also implemented the reweighted ℓ_1 minimization of Algorithm 4, with $k = 0.1n$ (n being the total number of frame pixels), which assumed no prior knowledge about the structural sparsity of the signal or the nonzero coefficients. This value of k was chosen heuristically, and is close to the actual support size of the signal. The original size of the image is 275×227. We reduced the resolution by roughly a factor of 0.05 for more tractability of ℓ_1 solver in MATLAB. In addition, only the gray-scale version of the difference image was taken into account, and was normalized so that the maximum intensity is 1. Furthermore, prior to compression, the difference image was further sparsified by rounding the intensities less than 0.1 to zero. We picked the weight value $\omega = 2$ for the weighting stage of the reweighted ℓ_1 algorithms. The normalized recovery error is defined to be the sum square of the intensity differences in the recovered and the original image, divided by the sum square of the original image intensity, i.e., $\sum_{i\in\text{frame}}(I_i - \hat{I}_i)^2 / \sum_{i\in\text{frame}} I_i^2$. The average normalized error for ℓ_1 minimization and reweighted ℓ_1 minimization is displayed in Figure 6.9a as a function of δ. The average is taken over 50 realizations of i.i.d. Gaussian measurement matrices for each δ. As can be seen, the recovery improvement is significant in the reweighted ℓ_1 minimization.

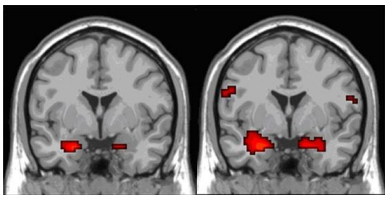


Figure 6.8: Functional MRI images of the brain at two different instances illustrating the brain activity. Green boxes identify the region with higher associated weight in the weighted ℓ_1 recovery. Image is adopted from https://sites.google.com/site/psychopharmacology2010/student-wiki-for-quiz-9.

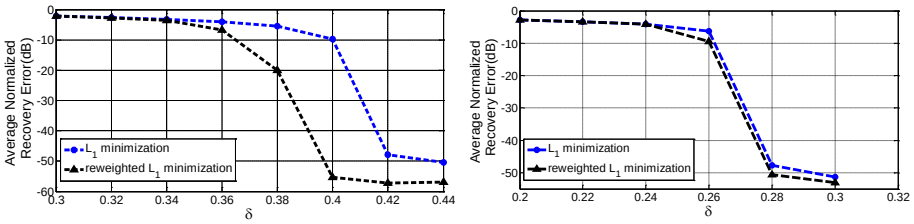


Figure 6.9: Average normalized recovery error for ℓ_1, and reweighted ℓ_1 minimization recovery of the difference between the subframes of (a) a pair of satellite images shown in Figure 6.7, and (b) the pair of brain fMRI images shown in Figure 6.8. Data is averaged over different realizations of measurement matrices for each δ.

Another experiment was done on a pair of brain fMRI images taken at two different instances of time, shown in Figure 6.8. Similar to the satellite images, the objective is to recover the difference image from a succinct set measurements. The original image size is 271×271, and similar preprocessing steps as for the satellite images were followed before compression. We used ℓ_1 minimization and Algorithm 4 with no presumed prior information, with $k = 0.1n$ and $\omega = 1.3$. The average normalized recovery errors are displayed in Figure 6.9b, from which we can infer similar conclusions as in the case of the satellite images.

6.10 Proof of Theorems

Proof of Lemma 6.6.2

Let $a = \Psi(\frac{M}{2N})$. We consider random variables $\hat{X}_i = |X_i| \cdot \mathbf{1}(|X_i| > a)$ for each

$1 \leq i \leq N$, where $\mathbf{1}(|X_i| > a)$ is equal to 1 if $|X_i| > a$, and is 0 otherwise. Also, let $\hat{S} = \hat{X}_1 + \hat{X}_2 + \cdots + \hat{X}_N$. We first note that the empirical average of the $\hat{X}_i$s converge to its expectation. More formally, an application of the Bernstein concentration inequality (see, e.g., [Mas03]) implies that for every $\epsilon' > 0$ and for some $c_1 > 0$, the following holds:

$$\mathbb{P}\left(|\hat{S}/N - \mathbb{E}(\hat{S}/N)| > \epsilon'\right) < \exp(-c_1 N\epsilon'). \tag{6.10.1}$$

On the other hand:

$$\mathbb{E}(\hat{S}/N) = \mathbb{E}\hat{X}_1 = \mathbb{P}(|X_1| > a) = \sqrt{2/\pi}e^{-\frac{a^2}{2}}. \tag{6.10.2}$$

Similarly, for the random variable $S_N = X_1 + X_2 + \cdots + X_N$, we can write the following concentration inequality using Chernoff bound for some $c_2 > 0$:

$$\mathbb{P}\left(|S_N/N - \mathbb{E}(S_N/N)| > \epsilon'\right) < \exp(-c_2 N\epsilon'). \tag{6.10.3}$$

Since $\mathbb{E}(S_N/N) = \sqrt{2/\pi}$, this establishes (6.6.13).

Let the random variable M' be the number of nonzero $\hat{X}_i$s. First of all, note that $\hat{S} = S_{M'}$. The rest of the proof includes the following steps. We prove that $S_{M'}/S_N$ is concentrated around $\mathbb{E}S_{M'}/\mathbb{E}S_N$ with high probability. Then we use the fact that M' also converges to its expected values, M, to show that S_M/S_N becomes arbitrarily close to $S_{M'}/S_N$. As a result, S_M/S_N will be concentrated around $\mathbb{E}S_{M'}/\mathbb{E}S_N$ with high probability, which is the desired result.

Concentration of $S_{M'}/S_N/$ is shown by using equations (6.10.2) and (6.10.3) simultaneously. Combining the two inequalities, we conclude that:

$$\mathbb{P}\left(\left|\frac{S_{M'}}{N} - \sqrt{2/\pi}e^{-\frac{a^2}{2}}\right| \leq \epsilon' \text{ and } \left|\frac{S_N}{N} - \sqrt{\frac{2}{\pi}}\right| \leq \epsilon'\right) \geq 1 - e^{-c_1 N\epsilon'} - e^{-c_2 N\epsilon'}, \tag{6.10.4}$$

and thus:

$$\mathbb{P}\left(\frac{\sqrt{2/\pi}e^{-\frac{a^2}{2}} - \epsilon'}{\sqrt{2/\pi} + \epsilon'} \leq \frac{S_{M'}}{S_N} \leq \frac{\sqrt{2/\pi}e^{-\frac{a^2}{2}} + \epsilon'}{\sqrt{2/\pi} - \epsilon'}\right) \geq 1 - e^{-c_1 N\epsilon'} - e^{-c_2 N\epsilon'}, \tag{6.10.5}$$

and consequently:

$$\mathbb{P}\left(\left|\frac{S_{M'}}{S_N} - e^{-\frac{a^2}{2}}\right| \leq \frac{2\sqrt{2/\pi}(e^{-\frac{a^2}{2}}+1)\epsilon'}{2/\pi - \epsilon'^2}\right) \geq 1 - e^{-c_1 N\epsilon'} - e^{-c_2 N\epsilon'}. \tag{6.10.6}$$

If ϵ' is sufficiently small, then $\frac{2\sqrt{2/\pi}(e^{-\frac{a^2}{2}}+1)\epsilon'}{2/\pi-\epsilon'^2} \leq \alpha\epsilon'$, for some constant $\alpha > 0$. Taking $\epsilon'' = \alpha\epsilon'$, $\alpha_1 = c_1/\alpha$, and $\alpha_2 = c_2/\alpha$, we can say that for sufficiently small ϵ'' the following holds:

$$\mathbb{P}\left(\left|\frac{S_{M'}}{S_N} - e^{-\frac{a^2}{2}}\right| \leq \epsilon''\right) \geq 1 - e^{-\alpha_1 N\epsilon''} - e^{-\alpha_2 N\epsilon''}. \tag{6.10.7}$$

Now we show that the quantity $\frac{|S_M - S_{M'}|}{S_N}$ will be arbitrarily small for large N. To do so, assume without loss of generality that $|X_1| \geq |X_2| \geq \cdots \geq |X_N|$, and that $M_1 = \min(M, M')$, and $M_2 = \max(M, M')$. We then have:

$$|S_M - S_{M'}| = |X_{M_1+1}| + |X_{M_1+2}| + \cdots + |X_{M_2}|, \tag{6.10.8}$$

and

$$\begin{aligned}
|S_N| &= |X_1| + |X_2| + \cdots + |X_N| \\
&\geq |X_1| + |X_2| + \cdots |X_{M_1}| \\
&\geq (N - M_1)|X_{M_1}| \\
&\geq \frac{N - M_1}{M_2 - M_1}|S_{M_2} - S_{M_1}| \qquad (6.10.9) \\
&\geq \frac{N - M}{|M - M'|}|S_M - S_{M'}|. \qquad (6.10.10)
\end{aligned}$$

Note that equation (6.10.9) holds because $|X_{M_1}|$ is larger than all the values $|X_{M_1+1}|$,$|X_{M_1+2}|$,. . . ,$|X_M$ and is therefore larger than their mean. It directly follows from (6.10.10) that:

$$\frac{|S_M - S_{M'}|}{S_N} \leq \frac{|M' - M|}{N - M}. \tag{6.10.11}$$

Therefore, to show the concentration of the left-hand side in the above inequality, it suffices to show that $\frac{|M'-M|}{N-M}$ concentrates. Since the variables $X_i' = \mathbf{1}(|X_i| > a)$

are independent Bernoulli random variables with probability $2Q(a) = \frac{M}{N}$ of being nonzero, a Chernoff concentration bound on their empirical average implies that

$$\mathbb{P}(\left|\frac{\sum_{i=1}^{n} X_i'}{N} - \mathbb{E}X'\right| \leq \epsilon''') \geq 1 - e^{-c_3\epsilon'''N}, \tag{6.10.12}$$

for some $c_3 > 0$, and for every $\epsilon''' > 0$, where X' has the same distribution as all X_i's. Noting that $\sum_{i=1}^{n} X_i' = M'$ and $\mathbb{E}X' = M/N$, the above implies that:

$$\mathbb{P}(\frac{|M - M'|}{N} \leq \epsilon''') = \mathbb{P}(\frac{|M - M'|}{N - M} \leq \frac{1}{1 - M/N}\epsilon''') \geq 1 - e^{-c_3\epsilon'''N}. \tag{6.10.13}$$

If the ratio M/N is kept constant, the quantity $\frac{\epsilon'''}{1-M/N}$ will be smaller than any $\tilde{\epsilon} > 0$ as ϵ''' becomes arbitrarily small, which shows the concentration of $\frac{|M-M'|}{N-M}$. Using this and the inequality of (6.10.11) we can conclude that $\frac{|S_M - S_{M'}|}{S_N} \leq \tilde{\epsilon}$ with probability $1 - e^{-\alpha_3\tilde{\epsilon}N}$ for some constant $\alpha_3 > 0$. Combining this latter conclusion with (6.10.7), it follows that:

$$\mathbb{P}\left(\left|\frac{S_M}{S_N} - e^{-\frac{a^2}{2}}\right| \leq \epsilon'' + \epsilon'''\right) \geq 1 - e^{-\alpha_1 N\epsilon''} - e^{-\alpha_2 N\epsilon''} - e^{-\alpha_3 N\tilde{\epsilon}}. \tag{6.10.14}$$

Consequently, we conclude that if ϵ is sufficiently small, the following holds:

$$\mathbb{P}\left(\left|\frac{S_M}{S_N} - e^{-\frac{a^2}{2}}\right| \leq \epsilon\right) \geq 1 - 3e^{-cN\epsilon}, \tag{6.10.15}$$

for some $c > 0$, which concludes the proof of (6.6.14).

Proof of Lemma 6.6.4

Let $\beta = 1 - 2Q(\sqrt{-2\log(1-\alpha)})$, and without loss of generality assume that the k nonzero values of $\mathbf{x}$ are $x_1, x_2, \ldots, x_k$, with $|x_1| \leq |x_2| \leq \cdots \leq x_k$. In order to show that $W(\mathbf{x}, \alpha\|\mathbf{x}\|_1) < k(\beta + \epsilon)$, it suffices to show that $\sum_{i=1}^{k(\beta+\epsilon)} |x_i| > \alpha\|\mathbf{x}\|_1$. Applying the order statistic result of Lemma 6.6.2, we have that with high probability:

$$\frac{\sum_{i=1}^{k(\beta+\epsilon)} |x_i|}{\sum_{i=1}^{k} |x_i|} \approx 1 - \exp(-\frac{\Psi(\frac{1-\beta-\epsilon}{2})}{2}) > 1 - \exp(-\frac{\Psi(\frac{1-\beta}{2})}{2}) = f, \tag{6.10.16}$$

which concludes the proof.

6.11 Conclusion

We introduced a new two-step reweighted ℓ_1 minimization for the recovery of linearly compressed sparse signals. We proved that for sparse signals, the nonzero entries of which are drawn from a broad class of continuous distributions, the proposed algorithm achieves a recovery threshold strictly better than that of ℓ_1 minimization. Our theoretical analysis predicts that the performance improvement strongly depends on the distribution of the nonzero entries, and should be better for distributions with a smaller non-vanishing order of derivative at the origin. This was very closely verified by our numerical simulations. For distributions with no finite order non-vanishing derivative at origin, our analysis does not predict any improvement in the performance. This is also the case in practice: For ternary signals with nonzero values equal to ± 1, no improvement is observed in the empirical recovery threshold over the regular ℓ_1 minimization. Our analysis was based on random Gaussian measurement matrices, and the robustness results of ℓ_1 minimization. Possible related future research could address other measurement matrix ensembles, and the development of reweighted algorithms that can universally improve the recovery performance of linear programming. On the other hand, the improvement predictions using our theoretical tools are not tight, due to upper bounding techniques and worst-case considerations in various parts of our proofs, especially in predicting the approximate support recovery potential of ℓ_1 minimization. Future work can concentrate on tightening these bounds through more clever techniques, and consequently achieving more promising performance guarantees for reweighted linear programming.

Part III

Connections and Extensions of Methods

Chapter 7

LP Decoding for Error-Correction Codes

The last part of this dissertation addresses extensions and connections of the methods of the previous chapters to problems beyond sparse vector recovery. Specifically, in this chapter, we study channel coding using LDPC codes, and discuss a novel iterative LP decoding algorithm, the basic idea for which is driven by similar ideas in compressed sensing discussed in Chapter 6. In doing so, we introduce the concept of robustness for LP decoding and, in a very similar fashion to the standard ℓ_1 regression for compressed sensing, we prove that LP decoding can provide approximate error set estimation for an unknown binary codewords. In the next chapter, we study the low-rank matrix recovery problem, and show that there is a natural extension of expander graphs to linear operators that act on matrices. We call the resulting measurements subspace expanders. We show that in the first place such objects exist, and that they lend themselves to combinatorial-type recovery algorithms for low-rank matrix recovery, in very much the same way that expander graphs do in compressed sensing.

Figure 7.1: Schematics of a channel coding scheme

$\mathcal{C}$	binary linear error correcting code
n	codeword length
R	code rate
$\mathbf{H}$	parity check matrix of $\mathcal{C}$
$\mathcal{G}$	Tanner graph of $\mathcal{C}$
$\mathcal{P}$	fundamental (marginal) polytope of $\mathcal{C}$
γ	log-likelihood ratio vector
$\mathbf{x}^{(c)}$	codeword of $\mathcal{C}$
$\mathbf{x}^{(r)}$	output vector of the channel
$\mathbf{x}^{(p)}$	pseudo-codeword solution of LP decoding
$\Gamma(S)$	set of the neighbors of S

7.1 Introduction

Error control coding was introduced in 1940s (presumably pioneered by Hamming) and, along with the invention of information theory, was recognized as a means of achieving the information theoretic capacity of communication channels (a.k.a. channel coding) and increasing the reliability of information processing systems. Today, error detection and correction codes are used in almost every digital system from cellular transceivers to Blu-ray discs and flash memories. The schematic of an error correcting system used at the front ends of a communication channel is shown in Figure 7.1. The idea is that the "message" m that contains information is "encoded" to form x, possibly by adding redundancy to protect it against error that is caused by the channel. If the encoding scheme is good enough, the decoder's approximation, $\hat{x}$, would be equal to x, and thus the message can be retrieved perfectly.

Error correcting codes are of two main types: Block codes and convolutional (tree) codes. Block codes consist of blocks of symbols called "codewords" the collection

of which form a "codebook". Examples of block codes are Hamming, Hadamard, BCH, Reed-Solomon, LDPC codes, etc. On the other hand, convolutional codes are generated based on sequential encoding of the message by convolving it with an input impulse response. The most highlighted example of convolutional codes is the family of turbo codes [BGT93], used widely in wireless and satellite communications.

Block codes can be linear or nonlinear. Codewords of linear block codes are defined over finite fields. Every codeword is a vector of length n that belongs to a finite (Galios) field $\mathbb{F}^n$ (such as the binary field), and is closed under the addition operator (i.e., the linear addition of every two codewords in the codebook form another codeword). n is called the code length. In contrast, nonlinear codes can have variable block lengths. BCH (Bose Chaudhuri Hocquenghem) codes, Reed-Solomon codes, and low density parity check (LDPC) codes are all binary linear codes, and examples of interesting nonlinear codes are the Levenshtein and Delsarte-Goethals codes.

In particular, LDPC codes are a family of linear block codes that have become very popular because of their asymptotic near-Shannon-capacity-achieving property[1] and modern non-algebraic decoding schemes. These codes were introduced by Galager in his thesis in the 1960s [Gal63], but were revisited with a great deal of interest in the late 1990s [MN99], when they were shown to have a performance very competitive with the convolutional Turbo codes.

Unlike analytic codes and similar to Turbo codes, most efficient decoders for LDPC codes are iterative, and are based on message-passing algorithms such as the bit flipping algorithm [Gal63], or belief-propagation-type algorithms such as the min-sum or sum-product methods [Wym07]. Exact analysis of belief propagation algorithms for LDPC codes is cumbersome. Many efforts have been taken with this aim, including the density evolution technique of [LMSS98, RU01]. However, in most cases, the analysis is limited to asymptotically large block lengths and is valid with the condition

[1] It is now known that Polar codes can achieve the capacity of binary symmetric memory-less channels [Ari09].

of some simplifying assumptions such as cycle-free Tanner graphs. It is worthwhile to mention that belief-propagation decoding techniques are now widely used and studied outside the context of error-correcting in other statistical inference problems, hidden Markov and graphical models, and by researchers of particle physics and statistical mechanics (see, e.g., [Fre98]).

A more systematic treatment of LDPC decoding became available with the introduction of linear programming (LP) decoding techniques by Feldman, Karger, and Wainwright [Fel03, FWK05]. The method is based on solving a linear-programming relaxation of an integer program that corresponds to finding the codeword of maximum likelihood, i.e., maximum-likelihood (ML) decoding. LP decoding is connected (but not equivalent) to message-passing decoding [FKW02, WJW02] and graph covers [KV03, VK06], and has received substantial recent attention (see, e.g., [VK06], and [TS06]). The practical performance of LP decoding is roughly comparable to min-sum decoding and slightly inferior to sum-product decoding. In contrast to message-passing decoding, however, the LP decoder either concedes failure on a problem, or returns a codeword along with a guarantee that it is the ML codeword, thereby eliminating any undetected decoding errors.

A related line of work has studied various improvements to either standard iterative decoding [Fos01, PNF04] or to LP decoding via nonlinear extensions [YFW06] or loop corrections [CC06], or via heuristics such as facet guessing [DGW09] or mixed integer iteration [DYW07]. Understanding the dynamics of LP decoding for LDPC codes and improving upon them is, therefore, a highly motivated task.

In this chapter, we consider the possibility of improving upon the LP decoding for binary-input-binary-output channels, by using ideas from compressed sensing. This is a somewhat reverse path (and not intuitive!), as most results in the literature have focused on adopting channel coding methodology and using it over the real numbers to sparse recovery. We elaborated on this point in Chapter 3, where we showed that LDPC matrices of large girth can be used as measurement matrices for

sparse recovery, and proved near-optimal recovery thresholds using the Basis Pursuit algorithm. However, the focus of this chapter is on the channel coding problem itself.

7.2 Contributions

We introduce a novel algorithm for decoding binary linear codes by linear programming. The algorithm is build upon the LP decoding algorithm of Feldman et al., with additional post-processing steps. If the LP decoder fails to retrieve a valid integer codeword, a second linear program is used with a reweighted objective function based on the outcome of the original LP decoder. It might sound trivial to the astute reader at this point that the idea for this method was inspired by the reweighted Basis Pursuit method introduced in Chapter 6 in the context of compressed sensing. Our analysis shows that for some LDPC ensembles we can improve the provable threshold guarantees compared to standard LP decoding. We also show significant empirical performance gains for the reweighted LP decoding algorithm with very small additional computational complexity.

The main idea of the technique is to add a second LP as a post-processing step when original LP decoding fails and outputs a fractional pseudo-codeword. We use the difference between the input channel likelihood and the pseudo-codeword coordinate to find a measure of disagreement or unreliability for each bit. We subsequently use this unreliability to bias the objective function and re-run the LP with the reweighted objective function. The reweighting increases the cost of changing reliable bits and decreases the cost for unreliable bits. We present an analysis that the provable recovery thresholds for binary bit-flipping channels improve for certain families of LDPC codes. We stress that the actual thresholds, even for the original LP decoding algorithm, remain unknown. Our analysis only establishes that the obtainable lower bounds on the fraction of recoverable errors are improved compared to the corresponding bounds for LP decoding. It is possible, however, that this is just

an artifact of the lower bounding techniques and that the true threshold is identical for both algorithms. In any case, the empirical performance gains we observe in our preliminary experimental analysis seem quite substantial.

A central idea in our analysis is the notion of robustness to changes in the binary channel bit-flipping probability. As with the proposed algorithm itself, this concept was inspired by a similar notion in compressive sensing [CWB08, KXAH10], elaborated and used in Chapter 6. We note that the reweighting idea discussed in this chapter involves changing the objective function of the linear program from the reweighted max-product algorithm [WJW05].

7.3 Preliminaries

The rate of a linear binary code $\mathcal{C}$ is denoted by R, and is defined as the ratio of the codeword length to the message length. More specifically, if the parity check matrix of $\mathcal{C}$ is $\mathbf{H} \in \mathbb{F}_2^{m \times n}$, then n is the length of each codeword in $\mathcal{C}$, m is the length of the message mapped to a codeword, and $R = \frac{m}{n}$. The Tanner (factor) graph corresponding to $\mathcal{C}$ is denoted by $\mathcal{G} = (X_v, X_c, \mathcal{E})$, where X_v and X_c are the sets of variable nodes and check nodes, respectively, and $\mathcal{E}$ is the set of edges. The parity check matrix $\mathbf{H}$ of a code $\mathcal{C}$ is in fact the adjacency matrix of the bipartite Tanner graph $\mathcal{G}$. For a variable node $v \in X_v$, the set of the neighbors of v in X_c is denoted by $\Gamma(v)$. Similarly, for a check node $c \in X_c$, the set of the neighbors of c in X_v is denoted by $\Gamma(c)$. Furthermore, for a subset F of the nodes of a Tanner graph $\mathcal{G}$ (either variable or check nodes), $\Gamma(F)$ denotes the set of the neighbors of the nodes in F. For regular Tanner graphs, d_v and d_c denote the degree of variable and check nodes, respectively. The girth of a graph $\mathcal{G}$, denoted by girth($\mathcal{G}$), is defined to be the size of the smallest cycle in $\mathcal{G}$.

A memoryless channel C is one that acts on every bit of the input sequence individually. For a channel with binary input and an output alphabet $\mathcal{Y}$, a set of

transition probabilities $P_{Y|X}(y|x)$ are defined that determine the probability of the input symbol getting mapped to different output symbols. For a received symbol y, we can define a log likelihood ratio as $\log(\frac{P_{Y|X}(y|x=0)}{P_{Y|X}(y|x=1)})$, where x is the transmitted symbol. If a codeword $\mathbf{x}^{(c)}$ of length n from the linear code $\mathcal{C}$ is transmitted through the channel, and an output vector $\mathbf{x}^{(r)}$ is received, a maximum likelihood decoder can be used to estimate the transmitted codeword by finding the *most likely* transmitted input codeword. Let γ_i be the likelihood ratio assigned to the i^{th} received bit in $\mathbf{x}_i^{(r)}$, and γ be the likelihood vector $\gamma = (\gamma_1, \cdots, \gamma_n)^T$. The ML decoder can be formalized as follows [Fel03]:

$$\begin{aligned} \text{ML decoder:} \quad & \text{minimize } \gamma^T \mathbf{x} \\ & \text{subject to } \mathbf{x} \in \text{conv}(\mathcal{C}), \end{aligned} \tag{7.3.1}$$

where $\text{conv}(\mathcal{C})$ is the convex hull of all the codewords of $\mathcal{C}$ in $\mathbb{R}^n$. The linear program (7.3.1) solves the ML decoding problem by the virtue of the fact that the objective $\gamma^T \mathbf{x}$ is minimized at a corner point (a.k.a. vertex) of $\text{conv}(\mathcal{C})$, which is necessarily a codeword. In fact, the set of vertices of $\text{conv}(\mathcal{C})$ is equal to set of the codewords of $\mathcal{C}$. Since decoding for general linear codes is NP hard, it is unlikely that $\text{conv}(\mathcal{C})$ can be efficiently described through a polynomial number of linear (or convex) inequalities. Feldman et al. introduced a relaxation of (7.3.1) by replacing the polytope $\text{conv}(\mathcal{C})$ with a new polytope $\mathcal{P}$ that has much fewer facets, contains $\text{conv}(\mathcal{C})$, and retains the codewords of $\mathcal{C}$ as its vertices [Fel03]. One way to construct $\mathcal{P}$ is the following. If the parity check matrix of $\mathcal{C}$ is the $m \times n$ matrix H and if $\mathbf{h}_j^T$ is the j^{th} row of H, then

$$\mathcal{P} = \cap_{1 \le j \le m} \text{conv}(\mathcal{C}_j), \tag{7.3.2}$$

where $\mathcal{C}_j = \{\mathbf{x} \in \mathbb{F}^n \mid \mathbf{h}_j^T \mathbf{x} = 0 \bmod 2\}$. $\mathcal{P}$ is called the LP polytope or the marginal polytope for the code $\mathcal{C}$, and its vertices are called "pseudo-codewords", including all of the codewords in $\mathcal{C}$. In addition, $\mathcal{P}$ has other redundant vertices with fractional

coefficients in $[0,1]^n$. Moreover, if a pseudo-codeword is integral, i.e., if it has 0 or 1 coefficients, then it is definitely a codeword. The LP relaxation of (7.3.1) over the marginal polytope is as follows:

$$\begin{aligned} \text{LP decoder:} \quad & \text{minimize } \gamma^T \mathbf{x} \\ & \text{subject to } \mathbf{x} \in \mathcal{P}. \end{aligned} \tag{7.3.3}$$

The number of facets of $\mathcal{P}$ is exponential in the maximum Hamming weight of a row of H, and polynomial in m. Therefore, for LDPC codes with constant row density, $\mathcal{P}$ has a polynomial number of facets. Consequently, (7.3.3) is solvable in polynomial time.

For channels with binary outputs, (7.3.3) can be expressed in a more intuitive way. In this case, minimizing the log-likelihood objective $\gamma^T \mathbf{x}$ is equivalent to minimizing the Hamming distance between the codeword $\mathbf{x}$ and the output of the channel, $\mathbf{x}^{(r)}$. Furthermore, since the coefficients of pseudo-codewords in the marginal polytope are in $[0,1]$, we may replace the Hamming distance $d_{\mathbf{H}}(\mathbf{x}, \mathbf{x}^{(r)})$with the ℓ_1 distance $\|\mathbf{x} - \mathbf{x}^{(r)}\|_1$. This implies that the decoder (7.3.3) is equivalent to

$$\begin{aligned} \text{BSC-LP decoder:} \quad & \text{minimize } \|\mathbf{x} - \mathbf{x}^{(r)}\|_1 \\ & \text{subject to } \mathbf{x} \in \mathcal{P}. \end{aligned} \tag{7.3.4}$$

The above formulation can be interpreted as follows: For a received output binary vector $\mathbf{x}^{(r)}$, the solution to the LP decoder is essentially the closest (in ℓ_1 norm) pseudo-codeword to $\mathbf{x}^{(r)}$.

Linear programming decoding was first introduced by Feldman et al. [Fel03, FWK05]. Subsequently [FMS$^+$07] it was shown that if the parity check matrix is chosen to be the adjacency matrix of a high-quality expander, LP decoding can correct a constant fraction of errors. A fundamental lemma in [FWK05] and used in the

results therein, is that the marginal polytope $\mathcal{P}$ has the same topology seen from every codeword. Consequently, for the analysis of LP decoding, it can be assumed without loss of generality that the transmitted codeword is the all-zero codeword.

The theoretical results of [FMS$^+$07] were based on a dual witness argument, i.e., a feasible set of variables which set the dual of LP equal to zero, and thus guarantees that the original codeword is the minimizer of the objective function over the entire marginal polytope. However, the bounds on success threshold of LP decoding achieved by this technique are considerably smaller than the empirical recovery thresholds of LP decoder in practice. A later analysis of LP decoding by Daskalakis et al. [DDKW08] improved upon those bounds for random expander codes, through employing a different dual witness argument, and considering a "weak" notion of LP success rather than the "strong" notion of [FMS$^+$07]. A strong threshold means that *every* set of errors of up to a certain size can be corrected, whereas a weak threshold implies that *almost all* error sets of a certain size are recoverable. Note that there is a gap of about one order of magnitude between the error-correcting thresholds of [DDKW08] and those observed in practice.

The arguments of [FMS$^+$07] and [DDKW08] are based on the existence of dual certificates that guarantee the success of the LP decoder and require codes that are based on bipartite expander graphs. A more recent work of Arora et al. uses a quite different certificate based on the primal LP problem [ADS09]. This approach results in fairly easier computations and significantly better thresholds for LP decoding. However, the underlying codes discussed in [ADS09] are based on factor graphs with a large girth (at least doubly logarithmic in the length of the codewords, n), rather than unbalanced expanders considered in previous arguments. Note that similar to [DDKW08], the bounds of [ADS09] are weak bounds, certifying that for a random set of errors up to a fraction of code length, LP decoding succeeds with high probability, for sufficiently large n. The largest such fraction is called the weak recovery threshold.

The arguments of this chapter are based on the systematic connections between

the problems of channel-coding LP and compressed sensing ℓ_1 relaxation developed in [DV09, DSV]. We build on those bridges to improve LP decoding, and further extend the ideas of robustness and reweighted ℓ_1 minimization in compressed sensing to channel-coding LP.

7.4 Extended Certificate and Robustness of LP Decoder

The success of LP decoder is often certified by the existence of a "dual witness" [FMS$^+$07, DDKW08]. Similarly, for ℓ_1 minimization in the context of CS, a dual witness certificate can guarantee that the recovery of sparse signals is successful [CT05]. However, it has proven more promising to express the success condition of ℓ_1 minimization in terms of the properties of the null space of the measurement matrix [DH01, SXH08, CDD08]. The condition is called the "null space property", through which it is possible to characterize one class of "good" measurement matrices for CS, namely matrices that are congruent with ℓ_1 minimization decoding. The advantage of the null space interpretation, apart from the fact that it results in sharper analytical bounds, is that with proper parametrization it can also be used to evaluate the performance of ℓ_1 minimization in the presence of noise. This is known as the "robustness" of ℓ_1 minimization. A consequence of the robustness property is that when ℓ_1 minimization fails to recover a sparse signal, it often gives a decent approximation to it [KXAH10]. To the best of our knowledge, a similar certificate has not been introduced in the context of channel-coding linear programming. In other words, when LP decoding fails to return an integral solution, it is not known how close the resulting pseudo-codeword is to the original codeword. We answer this question in the following way: We introduce a property of arbitrary linear codes $\mathcal{C}$ called fundamental cone property (FCP), and show that for communication channels with binary outputs, FCP is related to the robustness of the solution of the LP decoder

over the marginal polytope of the considered code. The robustness of LP decoding has two consequences. First, it implies that the linear program is tolerant to limited mismatch in implementation. Second, it implies that the pseudo-codeword obtained by LP decoding provides additional probabilistic information about the structure of the bit errors in the form of additional bit-flip reliability. This information can be utilized in further processing steps, such as reweighted linear program to improve the overall performance of the decoder. We discuss these issues in the following sections.

Definition 18. *Let H be a parity check matrix. Define $\mathcal{J}$ and $\mathcal{I}$ to be the set of rows and columns of H. Also, for each $j \in \mathcal{J}$, define $\mathcal{I}_j = \{i \in \mathcal{I} \mid H(j,i) = 0\}$. The fundamental cone, $\mathcal{K}(H)$, of H is the set of all vectors $\omega = (\omega_1, \omega_2, \ldots, \omega_n)^T$ that satisfy*

$$\omega_i \geq 0, \qquad \forall 1 \leq i \leq n, \tag{7.4.1}$$

$$\omega_i \leq \sum_{i' \in \mathcal{I}_j \setminus i} \omega_{i'}, \quad \forall j \in \mathcal{J} \ \forall i \in \mathcal{I}_j. \tag{7.4.2}$$

$\mathcal{K}(H)$ is the smallest cone in $\mathbb{R}^n$ that encompasses the polytope $\mathcal{P}$. If a vector lies on an edge of $\mathcal{K}$, it is called a "minimal pseudo-codeword". For simplicity, in the sequel, we use $\mathcal{K}$ instead of $\mathcal{K}(H)$ whenever there is no ambiguity.

Definition 19. *Let $S \subset \{1, 2, \cdots, n\}$ and $C \geq 1$ be fixed. A code $\mathcal{C}$ with parity check matrix H is said to have the fundamental cone property FCP(S, C) if for every nonzero vector $\omega \in \mathcal{K}(H)$, the following holds:*

$$C\|\omega_S\|_1 < \|\omega_{S^c}\|_1. \tag{7.4.3}$$

If for every index set S of size k, $\mathcal{C}$ has the FCP(S, C), then we say that $\mathcal{C}$ has the fundamental cone property FCP(k, C).

In the next lemma we show how the FCP can be used to evaluate the performance of an LP decoder, even when it fails to recover the true codeword. The key assumption

here is that the channel is a bit-flipping channel, i.e., the channel output has binary format.

Lemma 7.4.1. *Let $\mathcal{C}$ be a code that has the FCP(S, C) for some subset S of bits and some $C \geq 1$. Suppose that a codeword $\mathbf{x}^{(c)}$ from $\mathcal{C}$ is transmitted through a bit-flipping channel, and the received codeword is $\mathbf{x}^{(r)}$. If the nonzero pseudo-codeword $\mathbf{x}^{(p)}$ is the output of the LP decoder for the received codeword $\mathbf{x}^{(r)}$, then the following holds:*

$$\|\mathbf{x}^{(p)} - \mathbf{x}^{(c)}\|_1 < 2\frac{C+1}{C-1}\|(\mathbf{x}^{(r)} - \mathbf{x}^{(c)})_{S^c}\|_1. \tag{7.4.4}$$

Proof. Without loss of generality, we may assume that the all-zero codeword was transmitted, i.e., $\mathbf{x}^{(c)} = 0$. We have

$$\begin{aligned}
\|\mathbf{x}_S^{(r)}\|_1 + \|\mathbf{x}_{S^c}^{(r)}\|_1 &= \|\mathbf{x}^{(r)}\|_1 \\
&\overset{(a)}{\geq} \|\mathbf{x}^{(p)} - \mathbf{x}^{(r)}\|_1 \\
&= \|(\mathbf{x}^{(p)} - \mathbf{x}^{(r)})_S\|_1 + \|(\mathbf{x}^{(p)} - \mathbf{x}^{(r)})_{S^c}\|_1 \\
&\overset{(b)}{\geq} \|\mathbf{x}_S^{(r)}\|_1 - \|\mathbf{x}_S^{(p)}\|_1 + \|\mathbf{x}_{S^c}^{(p)}\|_1 - \|\mathbf{x}_{S^c}^{(r)}\|_1.
\end{aligned} \tag{7.4.5}$$

(a) is true because from (7.3.4), $\|\mathbf{x}^{(p)} - \mathbf{x}^{(r)}\|_1 \leq \|\mathbf{x}^{(c)} - \mathbf{x}^{(r)}\|_1$. Also (b) holds by the triangular inequality. Note that $\mathbf{x}^{(p)} \in \mathcal{K}(H)$, so by definition, $C\|\mathbf{x}_S^{(p)}\|_1 < \|\mathbf{x}_{S^c}^{(p)}\|_1$. This implies that

$$\|\mathbf{x}_{S^c}^{(p)}\|_1 - \|\mathbf{x}_S^{(p)}\|_1 > \frac{C-1}{C+1}\|\mathbf{x}^{(p)}\|_1. \tag{7.4.6}$$

Applying this to the left-hand side of (7.4.5) we obtain

$$2\frac{C+1}{C-1}\|\mathbf{x}_{S^c}^{(r)}\|_1 > \|\mathbf{x}^{(p)}\|_1, \tag{7.4.7}$$

which is the desired result.

■

Note that Lemma 7.4.1 can serve as a certificate for the success of LP decoder. More specifically, let S be the index set of the flipped bits in the transmitted codeword, i.e., the set of bits that differ in $\mathbf{x}^{(r)}$ and $\mathbf{x}^{(c)}$. If FCP(S, C) holds for some $C > 1$, then Lemma 7.4.1 implies that LP decoding can successfully recover the original codeword. Now suppose that the error set (i.e., the set of flipped bits) is a superset of S, with cardinality slightly larger than $|S|$. Then the vector $(\mathbf{x}^{(r)} - \mathbf{x}^{(c)})_{S^c}$ has a few (but not too many) nonzero entries. Therefore, even if the LP decoder output $\mathbf{x}^{(p)}$ is not equal to the actual codeword, it is still possible to obtain an upper bound on its ℓ_1 distance to the unknown codeword. We recognize this as the robustness of LP decoder, and characterize it by FCP(S, C) for $C > 1$. We define two notions of robustness here. Strong robustness means that for *every* set S of up to some cardinality k, the FCP(S, C) holds. Equivalently, this means that FCP(k, S) holds. Weak robustness on the other hand, is an *almost all* notion, meaning that for a random set S of up to a certain size k, the FCP(S, C) holds with high probability. In the next section we present a thorough analysis of LP robustness for two categories of codes: expander codes and codes with $\Omega(\log \log n)$ girth. For these two classes of codes, rigorous analysis has been done on the performance of LP decoders in [FMS$^+$07, DDKW08, ADS09]. We build on the existing arguments to incorporate the robustness condition and analyze the fundamental cone property.

7.5 Sufficient Conditions for LP Robustness

In most cases, if there exists a certificate for the success of LP decoder, it can be often extended to guarantee that the LP decoder is robust, namely that the FCP condition is satisfied for some $C > 1$. By carefully re-examining the analysis of LP decoder, one might be able to make such a generalization. This is the main focus of this section. We consider three major methods that exist in the literature for analyzing the performance of LP decoders. The first one is due to Feldman et al. [FMS$^+$07],

and is based on using a dual-witness type of argument to certify the success of LP decoder for expander graphs. The second one is due to Daskalakis et al. [DDKW08], which also considers linear programming decoding in expander codes. Specifically, [DDKW08] analyzes the dual of LP and finds a simple combinatorial condition for the dual value to be zero (implying that the LP decoder is successful). The condition is essentially equivalent to the existence of a so-called "hyperflow" from the set of nodes corresponding to flipped bits to the set of nodes corresponding to unflipped bits in the Tanner graph of the code. The existence of a valid hyperflow can be secured by the presence of so-called (p, q)-matchings. It then follows from a detailed series of probabilistic calculations that (p, q)-matchings of interest exist for certain expander codes. The main difference between this analysis and that of [FMS$^+$07] is the probabilistic nature of the arguments in [DDKW08], which account for the weak recovery threshold.

A third analysis of the LP decoder was done by Arora et al., [ADS09], which is based on factor graphs with a doubly logarithmic girth. Unlike previous dual feasibility arguments, the authors in [ADS09] introduce a certificate in the primal domain which is of the following form: If, in the primal LP problem, the value of the objective function for the original codeword is smaller than its value for all vectors within a local deviation from the original codeword, then LP decoder succeeds. Local deviations are defined by weighted minimal local trees whose induced subgraphs are cycle-free. These concepts are explained in detailed in the sequel.

7.5.1 Strong LP Robustness for Expander Codes

Strong thresholds of LP decoding for expander codes are derived in [FMS$^+$07], through the use of the dual of the linear program (7.3.3). Assuming that the all-zero codeword was transmitted, showing that it is the optimal solution to the LP decoding is equivalent to the log-likelihood vector γ being an internal point of the dual of the cone $\mathcal{K}$. This can be easily verified by the definition of the dual cone and the formulation of

(7.3.3). In [FMS$^+$07] it is shown that the existence of a set of dual feasible variables is a sufficient condition for γ being an internal point of the dual of $\mathcal{K}$. Here, we take the notion of dual feasible variables and relate it to robustness of LP decoding through the FCP property. Recall that the factor graph of $\mathcal{C}$ is denoted by $\mathcal{G} = (X_v, X_c, \mathcal{E})$. A dual feasible set is defined as follows.

Definition 20. *For a given log likelihood vector $\gamma = (\gamma_1, \gamma_2, \ldots, \gamma_n)^T$, a set of dual feasible variables is a labeling of the edges of the factor graph $\mathcal{G}$, e.g., $\{\tau_{ij} \mid v_i \in X_v \; c_j \in X_c\}$, where the following two conditions are satisfied:*

i) For every check node $c_j \in X_c$ and every two disjoint neighbors of c_j, e.g., $v_i, v_{i'} \in \Gamma(c_j)$, we have $\tau_{ij} + \tau_{i'j} \geq 0$.

ii) For every variable node $v_i \in X_v$, we have $\sum_{c_j \in \Gamma(v_i)} \tau_{ij} < \gamma_i$.

We now show how a set of dual feasible variables can certify FCP properties for a code. The key is the following lemma, which will be proved in Section 7.9.

Lemma 7.5.1. *Suppose that a set of dual variables satisfy the feasibility conditions (Definition 20) for some log-likelihood vector γ. Then for every vector $\omega \in \mathcal{K}(\mathcal{C})$, the following holds*

$$\sum_{1 \leq i \leq n} \gamma_i \omega_i > 0. \tag{7.5.1}$$

Now consider a subset S of the bits and a constant $C > 1$, and let the vector $\gamma^{(S,C)} = (\gamma_1^{(S,C)}, \gamma_2^{(S,C)}, \ldots, \gamma_n^{(S,C)})^T$ be defined as follows:

$$\gamma_i^{(S,C)} = \begin{cases} -C & i \in S \\ 1 & i \in S^c \end{cases}, \; 1 \leq i \leq n. \tag{7.5.2}$$

Then it follows immediately from Lemma 7.5.1 and the definition of FCP that:

Corollary 7.5.2. *If a set of dual feasible variables exists for $\gamma^{(S,C)}$, then FCP(S, C) holds.*

Consequently, whenever an appropriate set of dual feasible variables exist, certain robustness conditions can be asserted for LP decoding. The existence of dual feasible sets were originally (and only, to the best of the authors' knowledge) proposed for the family of expander codes in [FMS$^+$07]. Pursuing similar techniques, we can generalize the arguments for construction of dual feasible variables and obtain FCP properties for expander codes as follows. For the consistency and completeness of the discussions, we first mention the definition of expander graphs (codes).

Definition 21. *Let $\mathcal{G} = (X_v, X_c, \mathcal{E})$ be a bipartite Tanner graph with regular variable degree d_v. $\mathcal{G}$ is called a $(k, \epsilon d_v)$-expander for $1 \leq k < |X_v|$ and $0 < \epsilon < 1$ if for every subset $F \subset X_v$ of size less than or equal to k, $|\Gamma(F)| \geq \epsilon d_v |F|$.*

The following theorem is proved in Section 7.9.

Theorem 7.5.3 (Strong Robustness of LP for Expanders)**.** *Let $\mathcal{G}$ be the factor graph of a code $\mathcal{C}$ of length n and rate $R = \frac{m}{n}$. If $\mathcal{G}$ is a bipartite $(\alpha n, \delta d_v)$-expander graph for some $\delta > 2/3 + 1/(3d_v)$, then for $t = \frac{3\delta - 2}{2\delta - 1}\alpha$ and $1 \leq C < \frac{2\delta - 1}{2\delta - 1 - 1/d_v}$, FCP$(S, C)$ holds.*

7.5.2 Weak LP Robustness for Expander Codes

We show that for random expander codes a probabilistic analysis similar to the dual witness analysis of [DDKW08] can be used to find the extents of the fundamental cone property for expander codes, in a weak sense. We rely on the matching arguments of [DDKW08], with appropriate adjustments. The following definition is given in [DDKW08].

Definition 22. *For nonnegative integers p and q, and a set F of variable nodes, a (p, q)-matching on F is defined by the following conditions:*

(a) Every variable node $v_i \in F$ must be matched with p distinct check nodes, and

(b) Every variable node $v_{i'} \in F^c$ must be matched with

$$X_{i'} := \max\{q - d_v + Z_{i'}, 0\} \tag{7.5.3}$$

check nodes in the set $\Gamma(F)$ which are different from the check nodes that the nodes in F are matched to, where $Z_{i'}$ is defined as $Z_{i'} := |\Gamma(v_{i'}) \cap \Gamma(F)|$.

We prove the following theorem that relates the existence of a (p, q)-matching to the fundamental cone property of a code $\mathcal{C}$. This theorem is proved in Section 7.9.

Theorem 7.5.4 (Weak Robustness of LP for Expanders)**.** *Let $\mathcal{C}$ be a code of rate R with a bipartite factor graph $\mathcal{G} = (X_v, X_c, \mathcal{E})$ where every variable node has degree d_v. Let S be a subset of the variable nodes. If a (p, q)-matching on S exists in $\mathcal{G}$, then $\mathcal{C}$ has the FCP($S, \frac{2p-d_v}{d_v-q}$).*

7.5.3 Weak LP Robustness for Codes with $\Omega(\log\log(n))$ Girth

Recall that $\mathcal{G} = (X_v, X_c, \mathcal{E})$ is used to denote the factor graph of the parity check matrix H (or of code $\mathcal{C}$), where X_v and X_c are the sets of variable and check nodes, respectively, and $\mathcal{E}$ is the set of edges. Also recall that the girth of $\mathcal{G}$ is defined as the size of the shortest cycle in $\mathcal{G}$. Without loss of generality, we assume that $X_v = \{v_1, v_2, \cdots, v_n\}$, where v_i is the variable node corresponding to the i^{th} bit of the codeword. Let $T \leq \frac{1}{4}\text{girth}(\mathcal{G})$ be fixed. The following notions are defined in [ADS09].

Definition 23. *A tree $\mathcal{T}$ of height $2T$ is called a skinny subtree of $\mathcal{G}$ if: it is rooted at some variable node v_{i_0}, for every variable node v in $\mathcal{T}$ all the neighboring check nodes of v in $\mathcal{G}$ are also present in $\mathcal{T}$, and for every check node c in $\mathcal{T}$ exactly two neighboring variable nodes of c in $\mathcal{G}$ are present in $\mathcal{T}$.*

Definition 24. *Let $\mathbf{w} \in [0, 1]^T$ be a fixed vector. A vector $\beta^{(\mathbf{w})}$ is called a minimal T-local deviation if there exists a skinny subtree of $\mathcal{G}$ of height $2T$ (e.g., $\mathcal{T}$), such that*

for every variable node $v_i, 1 \leq i \leq n$,

$$\beta_i^{(\mathbf{w})} = \begin{cases} \mathbf{w}_{h_i} & \text{if } v_i \in \mathcal{T} \setminus \{v_{i_0}\} \\ 0 & \text{otherwise} \end{cases},$$

where $h_i = \frac{1}{2} d(v_{i_0}, v_i)$.

The key to the derivations of [ADS09] is the following lemma.

Lemma 7.5.5 (Lemma 1 of [ADS09])**.** *For any vector* $\mathbf{z}$ *in the marginal polytope* $\mathcal{P}$*, and any positive vector* $\mathbf{w} \in [0,1]^T$*, there exists a distribution on the minimal* T*-local deviations* $\beta^{(\mathbf{w})}$ *such that*

$$\mathbb{E}\beta^{(\mathbf{w})} = \alpha \mathbf{z}$$

where $0 < \alpha \leq 1$.

Lemma 7.5.5 has the following interpretation. If a linear property holds for all minimal T-local deviations (e.g., $f(\beta^{(\mathbf{w})}) > 0$, where $f(\cdot)$ is a linear operator), then it also holds for all pseudo-codewords of the marginal polytope (i.e., $f(\mathbf{z}) > 0 \ \forall \mathbf{z} \in \mathcal{P}$). Interestingly enough, the robustness of LP decoding over a subset S of entries has a linear certificate, namely FCP(S, C).[2] In other words, if we define:

$$f_C^{(S)}(\mathbf{x}) = \sum_{i \in S^c} x_i - C \sum_{i \in S} x_i,$$

then FCP(S, C) holds if and only if $f_1^{(S)}(\mathbf{z}) > 0$ for every pseudo-codeword $\mathbf{z} \in \mathcal{P}$. On the other hand, based on Lemma 7.5.5, in order to show that this condition holds, it suffices to show that it is true for all minimal T-local deviations. In other words, one has to prove that $f_C^{(S)}(\beta^{(\mathbf{w})}) > 0$ for all minimal T-local deviations $\beta^{(\mathbf{w})}$. This simple observation helps us extend the probabilistic analysis of [ADS09] to robustness results for LP decoding. The key theorem is mentioned below while its proof can be found in Section 7.9. First we define η_C to be a random variable that takes the value

[2] Note that this is only true for bit-flipping channels, where the output alphabet is $\mathbb{F}_2$.

$-C$ with probability p and value 1 with probability $1-p$. Also, define the sequences of random variables $X_i, Y_i,\ i \geq 0$, in the following way:

$$\begin{aligned} Y_0 &= \eta_C, \\ X_i &= \min\{Y_i^{(1)}, \ldots, Y_i^{(d_c-1)}\} \quad \forall i > 0, \\ Y_i &= 2^i \eta_C + X_{i-1}^{(1)} + \cdots + X_{i-1}^{(d_v-1)} \quad \forall i > 0, \end{aligned} \tag{7.5.4}$$

where $X^{(j)}$s are independent random variables with the same distribution as X.

Theorem 7.5.6 (Weak Robustness of LP for High Girth Codes)**.** *Let $\mathcal{C}$ be an LDPC code with length n, Tanner graph $\mathcal{G}$, and regular check and variable degrees d_c and d_v, respectively. Also Let $0 \leq p \leq 1/2$, and S be a random subset of bits of size pn. If for some $j \in \mathbb{N}$,*

$$c = \zeta^{1/(d_v-2)} \min_{t \geq 0} \mathbb{E} e^{-tX_j} < 1,$$

where $\zeta = (d_c - 1)\frac{C+1}{C}(\frac{p \cdot C}{1-p})^{1/(C+1)}(1-p) < 1$, then with probability at least $1 - O(n)c^{d_v(d_v-1)^{T-1}}$ the code $\mathcal{C}$ has the FCP(S, C), where T is any integer with $j \leq T < 1/4 girth(\mathcal{G})$.

For $d_c = 6$ and $d_v = 3$, a lower bound on the robustness parameter C that results from Theorem 7.5.6 is plotted against the probability of bit flip p, in Figure 7.2.

7.6 Implications of LP Robustness

7.6.1 Mismatch Tolerance

One of the direct consequences of the robustness of LP decoding is that if there is a slight mismatch in the implementation of the LP decoder, its performance does not degrade significantly. More formally, suppose that, due to noise, quantization,

or some other factor, a mismatched log-likelihood vector $\gamma' = \gamma + \Delta\gamma$ is used to solve the linear program (7.3.3). In other words, despite the fact that the true log-likelihood vector γ is given to the decoder, an uncontrolled mismatch $\Delta\gamma$ is introduced when LP is implemented. This situation can happen, for instance, when a hardware implementation of the decoder is considered. In that case, $\Delta\gamma$ can be a stochastic vector resulting from physical imperfections, and is only introduced after the true log-likelihood vector is given as the input to the decoder. We refer to such decoder as a "mismatched LP decoder". Since the channel output is binary, the entries of γ all have the same amplitude g. We also define $\delta = \max_i |\Delta\gamma_i|$, and assume that $\delta < g$. We can prove the following theorem.

Theorem 7.6.1 (Mismatch Tolerance of LP Decoder)**.** *Suppose that S is the set of bit errors. Let $C = \frac{g+\delta}{g-\delta}$. If $\mathcal{C}$ has FCP(S, C), then the mismatched LP decoder corrects all errors and recovers the original codeword.*

Proof. We assume without loss of generality that the all-zero codeword is transmitted. We show that if FCP(S, C) holds, then the all-zero codeword is the minimum cost vector in the polytope $\mathcal{P}$. Suppose ω is a nonzero vector in the fundamental code $\mathcal{K}$. We begin with the definition of FCP(S, C) and write

$$-C\sum_{i\in S}\omega_i + \sum_{i\in S^c}\omega_i > 0. \tag{7.6.1}$$

Multiply both sides by $(g - \delta)$:

$$-\sum_{i\in S}(g+\delta)\omega_i + \sum_{i\in S^c}(g-\delta)\omega_i > 0. \tag{7.6.2}$$

We also know from the definition of δ that $\gamma'_i > (g - \delta)$ for $i \in S^c$, and $\gamma'_i > -g - \delta$ for $i \in S^c$, and that $\omega \geq 0$. Therefore

$$-\sum_{i\in S\cup S^c}\gamma'_i\omega_i > 0, \tag{7.6.3}$$

which proves that the all-zero codeword is the unique minimum cost solution of the mismatched LP.

■

7.6.2 Pseudo-codewords and High-Error-Rate Subsets

We showed in Section 7.4 that for an appropriate code $\mathcal{C}$ satisfying FCP, even when LP decoder fails to recover an actual codeword from the output of a bit-flipping channel, the reconstruction error can be bounded (see equation (7.4.4)). We now show that this property allows us to use the output of LP decoder to find a "high-error-rate" subset of the bits, namely a subset of bits over which the fraction of errors is significantly larger than the fraction of errors in the entire received codeword. We show that the size of such subset can be up to a constant fraction of the code length. Obtaining such *importance* subset is very crucial, since it provides additional reliability information about a significant proportion of the bits. Consequently, one can use this reliability information to bias the objective function and re-run a reweighted LP, or one with additional soft/hard constraints. This forms the idea for the proposed iterative LP decoding algorithm which will be outlined in Section 7.7. In this section, we focus on the derivation of the high-error-rate subset using the robustness results of LP decoding.

Theorem 7.6.2 (Error Set Approximation of LP)**.** *Suppose that a codeword* $\mathbf{x}^{(c)}$ *is transmitted through a bit-flipping channel, and the output* $\mathbf{x}^{(r)}$ *differs from the input in a set* K *of the bits with* $|K| = p^*(1+\epsilon)n$*, for some* $0 < p^* < 1$ *and* $\epsilon > 0$*. Also, suppose that for a subset* $K_1 \subset K$ *of size* p^*n*, FCP*(K_1, C) *holds for the considered code, for some* $C > 1$*, and that the optimal LP solution is the nonzero pseudo-codeword* $\mathbf{x}^{(p)}$*. If* L *is the set of the* $p^*(1+\epsilon)n$ *largest entries of the vector* $\mathbf{x}^{(r)} - \mathbf{x}^{(p)}$ *in magnitude, then:*

$$\frac{|L \cap K|}{|L|} \geq 1 - 2\frac{C+1}{C-1}\epsilon. \tag{7.6.4}$$

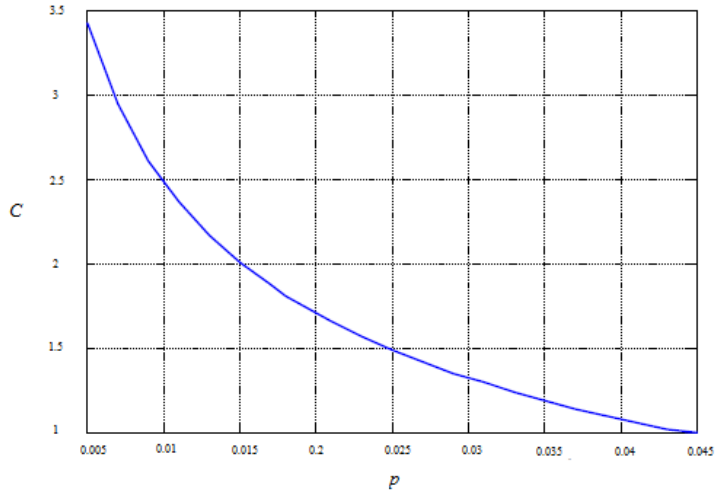


Figure 7.2: Approximate upper bound for the robustness factor C as a function of error probability p for $d_c = 6$ and $d_v = 3$, based on Theorem 7.5.6

Note that the above theorem provides a lower bound on the the fraction of errors in the received vector $\mathbf{x}^{(r)}$ over the set L. Consequently, if the resulting lower bound is large enough, the set L can be thought of as a high-error-rate subset of the bits. Also, note that the size of L is a constant fraction of the code length. Theorem 7.6.2 is proved using the following definition and lemma, as follows.

Definition 25. *Let $\mathbf{x} \in \mathbb{R}^n$ be a k-sparse vector. For $\lambda > 0$, We define $W(\mathbf{x}, \lambda)$ to be the size of the largest subset of nonzero entries of $\mathbf{x}$ that has ℓ_1 norm less than or equal to λ,* i.e.*,*

$$W(\mathbf{x}, \lambda) := \max\{|S| \mid S \subseteq supp(\mathbf{x}),\ \|\mathbf{x}_S\|_1 \leq \lambda\}. \tag{7.6.5}$$

Here we use Lemma 6.6.1 proved in Chapter 6 which for completeness, we repeat here.

Lemma 7.6.3 (proved in Chapter 6)**.** *Let $\mathbf{x}$ be a k-sparse vector and $\hat{\mathbf{x}}$ be another vector. Also, let K be the support set of $\mathbf{x}$ and L be the k-support set of $\hat{\mathbf{x}}$, namely*

the set of k largest entries of $\hat{\mathbf{x}}$. If $d = \|\mathbf{x} - \hat{\mathbf{x}}\|_1$, then

$$|K \cap L| \geq k - W(\mathbf{x}, d). \tag{7.6.6}$$

Proof of Theorem 7.6.2. Define $k = p^*(1+\epsilon)n$, and apply Lemma 6.6.1 to the k-sparse vector $\mathbf{x}^{(c)} - \mathbf{x}^{(r)}$, and the vector $\mathbf{x}^{(p)} - \mathbf{x}^{(r)}$. If L is the index set of the largest k entries of $\mathbf{x}^{(p)} - \mathbf{x}^{(r)}$ in magnitude, then from Lemma 6.6.1 we have

$$|K \cap L| \geq k - W(\mathbf{x}^{(c)} - \mathbf{x}^{(r)}, \Delta), \tag{7.6.7}$$

where $\Delta = \|\mathbf{x}^{(c)} - \mathbf{x}^{(p)}\|_1$. Since $\|\mathbf{x}^{(r)} - \mathbf{x}^{(c)}\|$ has only ± 1 nonzero entries, (7.6.7) can be written as

$$|K \cap L| \geq k - \|\mathbf{x}^{(c)} - \mathbf{x}^{(p)}\|_1. \tag{7.6.8}$$

We use the inequality in (7.4.4) to further lower bound the right-hand side of (7.6.8). Recall that $K_1 \subset K$ is such that $\mathcal{C}$ has FCP(K_1, C). Therefore, we can write:

$$\begin{aligned} |K \cap L| &\geq k - 2\frac{C+1}{C-1}\|(\mathbf{x}^{(r)} - \mathbf{x}^{(c)})_{K_1^c}\|_1 & (7.6.9)\\ &= k - 2\frac{C+1}{C-1}(k - p^*n). & (7.6.10) \end{aligned}$$

Dividing both sides by $k = |L| = |K|$, we conclude that at least a fraction $1 - 2\frac{C+1}{C-1}\epsilon$ of the set L are errors. ∎

7.7 Iterative Reweighted LP Algorithm and Improved Threshold

As stated in Theorem 7.6.2, by examining the deviation of the LP optimal (pseudo-codeword) and the received vector, it is possible to identify a high-error-rate (HER)

subset of bits in which the fraction of bit errors is higher than the overall probability of error. In other words, the code block can be divided into two regions L and L^c, one with a large fraction of errors, and the other one with a small fraction of errors. One way this imbalance can be exploited is by using a weighted LP scheme. This is outlined in the following iterative algorithm.

Algorithm 5 —Reweighted LP Decoding.

1: **Input:** Marginal polytope $\mathcal{P}$, received binary vector $\mathbf{x}^{(r)}$, fixed parameters $\lambda_1 > 0$ and $\lambda_2 < 0$.
2: **Output:** Approximation $\hat{\mathbf{x}}$ to $\mathbf{x}^{(r)}$.
3: Run LP decoding:

$$\min_{\mathbf{x} \in \mathcal{P}} \|\mathbf{x} - \mathbf{x}^{(r)}\|_1 \tag{7.7.1}$$

4: If the output $\mathbf{x}^{(p)}$ is integral terminate, otherwise proceed.
5: Set $\mathbf{x}^{(d)} := \mathbf{x}^{(r)} - \mathbf{x}^{(p)}$.
6: Sort the entries of $\mathbf{x}^{(d)}$ in terms of absolute value, and denote by L the index set of the **largest** pn entries.
7: Solve the following weighted LP and return the solution as the output:

$$\min_{\mathbf{x} \in \mathcal{P}} \lambda_1 \|(\mathbf{x} - \mathbf{x}^{(r)})_L\|_1 + \lambda_2 \|(\mathbf{x} - \mathbf{x}^{(r)})_{L^c}\|_1. \tag{7.7.2}$$

Algorithm 5 is only twice as complex as LP decoding. Recall that the subset L obtained in step 6 of the algorithm is an approximation for the original error set K. The intuition behind the final reweighted LP decoding step and the choice of λ_1, λ_2 is as follows. If the majority of the bits in L are believed to be errors, one possible approach to error correction is to flip all of those bits in the received vector $\mathbf{x}^{(r)}$, in which case the number of the errors will be reduced. If by doing so, the number of errors in the new vector becomes significantly smaller (namely, if the fraction of the errors in L is significantly larger than $1/2$), it is more likely that LP decoding can correct all errors. Note that flipping a subset of the bits and re-running LP is equivalent to negating the log-likelihood ratio assigned with those bits and running the reweighted LP decoding, which is essentially equivalent to a special case of the last step of Algorithm 5 with $\lambda_1 = -1$ and $\lambda_2 = 1$. However, one might be able to do

better by selecting the weights more optimally. To see how that might be, one should consider decoding codewords under non-uniform error rate models. In this case, the code block can be divided into two regions L and L^c, with respective error fractions p_1 and p_2, where, from the previous discussions we assume that $p_1 >> p_2$. A similar, more intuitive model to describe such non-uniformity in the error structure is follows: suppose that every bit in the set L is an error with probability p_1, and every bit in L^c is an error with probability p_2. If we write the maximum likelihood estimation of the codeword, it will be as equation (7.3.1), except that the vector γ does not just consist of ± 1 entries. The value of γ_i will be $\pm \log \frac{p_1}{1-p_1}$ if $i \in L$, and $\pm \log \frac{p_2}{1-p_2}$ if $i \in L^c$. Now if p_1 is very close to 1 and p_2 is very close to 0, the log likelihood ratio for a received bit will have opposite signs whether or not it is in the set L. That is the basic motivation for the choice of negative signs for λ_1 and λ_2. In practice, one might have to to try a range of different values for λ_1, λ_2 depending on the available estimations on the fractions p_1, p_2.

In the remainder of this section, we provide preliminary analysis on the performance of the proposed method. Ideally, one would be tempted to show that error correction thresholds of the reweighted scheme are higher than those of regular LP decoding. We could not show this at this point. Instead, we prove that the recovery threshold of the proposed reweighted scheme is strictly higher than any "provable robust" threshold that exists for LP decoding. To formalize the discussion, we first define the recovery thresholds of LP decoding as follows.

Definition 26 (Recovery thresholds)**.** *For a given code $\mathcal{C}$ of sufficiently large code length, "strong recovery threshold" of LP decoding is denoted by p_s^*, and is defined as the largest fraction such that every set of bit errors of size $p_s^* n$ is recoverable via LP decoding. "Weak recovery threshold" is denoted by p_w^*, and is defined by the largest fraction such that almost all sets of bit errors size $p_w^* n$ are recoverable via LP. In contrast, a fraction $0 \leq p_{sr}^* \leq 1$ is called a "robust strong threshold" of LP decoding, if for some constant $C > 1$, and for every subset S of bits of size $p_{sr}^* n$, FCP(S, C)*

holds. Similarly, $0 \leq p^*_{wr} \leq 1$ *is a "robust weak threshold" of LP decoding, if for some constant* $C > 1$*, and for almost all subsets* S *of bits of size* $p^*_{wr}n$*,* $FCP(S, C)$ *holds.*

Our next theorem states that the recovery thresholds of the proposed iterative method are higher than every robust threshold that one can prove for LP decoding.

Theorem 7.7.1 (Threshold Improvement of Algorithm 5)**.** *Let* p^*_{sr} *and* p^*_{wr} *be, respectively, a robust strong and a robust weak threshold for a code* $\mathcal{C}$*. There exist* $\epsilon_1 > 0$*,* $\epsilon_2 > 0$*,* $\lambda_1 < 0$*, and* $\lambda_2 > 0$ *so that every error set of size* $(1 + \epsilon_1)p^*_{sd}$*, and almost all error sets of size* $(1 + \epsilon_2)p^*_{wd}$ *can be corrected by Algorithm 5.*

To prove this theorem, we first consider decoding codewords under non-uniform error rate conditions. Suppose that we somehow know that the code block can be divided into two regions with different error rates. The following lemma shows that when there is such additional information about the distribution of bit errors in a codeword, a biased linear programming decoding can be used.

Lemma 7.7.2. *Suppose a codeword* $\mathbf{x}^{(c)}$ *is transmitted through a binary channel. Also suppose that the bits of* $\mathbf{x}^{(c)}$ *can be divided into two sets* L *and* L^c*, so that at least a fraction* p_1 *of the bits in* L *are flipped, and at most a fraction* p_2 *of the bits in* L^c *are flipped. Then the following optimization program:*

$$\min_{\mathbf{x} \in \mathcal{P}} -\|(\mathbf{x} - \mathbf{x}^{(r)})_L\|_1 + \|(\mathbf{x} - \mathbf{x}^{(r)})_{L^c}\|_1, \tag{7.7.3}$$

can recover $\mathbf{x}^{(c)}$*, provided that*

$$(1 - p_1)|L| + p_2|L^c| \leq p^*_{sr}, \tag{7.7.4}$$

when $0 < p^*_{sr} < 1$ *is a robust strong threshold of LP decoding.*

Proof. We assume without loss of generality that the all-zero codeword has been transmitted. We show that if the condition of (7.7.4) holds, then for every nonzero

pseudo-codeword $\mathbf{x}$, the objective function in (7.7.3) is larger than that of the all-zero vector, namely that:

$$-\|(\mathbf{x}-\mathbf{x}^{(r)})_L\|_1+\|(\mathbf{x}-\mathbf{x}^{(r)})_{L^c}\|_1>-\|(\mathbf{x}^{(r)})_L\|_1+\|(\mathbf{x}^{r})_{L^c}\|_1. \tag{7.7.5}$$

To see this, suppose that the set of flipped bits is denoted by K. Then (7.7.5) is equivalent to

$$-\sum_{i\in K\cap L}(1-x_i)-\sum_{i\in K^c\cap L}(x_i)+\sum_{i\in K\cap L^c}(1-x_i)+\sum_{i\in K^c\cap L^c}(x_i)>-|L\cap K|+|L^c\cap K|, \tag{7.7.6}$$

or equivalently:

$$\sum_{i\in S}x_i>\sum_{i\in S^c}x_i, \tag{7.7.7}$$

where $S=(K\cap L)\cup(K^c\cap L^c)$. However, note that if FCP(S,C) holds for some $C>1$, then (7.7.7) is certainly true. On the other hand, since the cardinality of S is at most $(1-p_1)|L|+p_2|L^c|$, which, by assumption, is less than or equal to p^*_{sr}, then by definition, for some $C>1$, FCP(S,C) holds. This completes the proof. ■

Proof of Theorem 7.7.1. We set $\lambda_1=-1$ and $\lambda_2=1$. Suppose without loss of generality that the all-zero codeword has been transmitted, and the received binary vector $\mathbf{x}^{(r)}$ has pn errors, where $p=(1+\epsilon_0)p^*_{sr}$. By definition, FCP(p^*_{sr},C) holds for some $C>1$. Therefore, if we apply Theorem 7.6.2 to the output of LP, namely $\mathbf{x}^{(p)}$, we conclude that the set L of most pn deviated bits in $\mathbf{x}^{(p)}$ with respect to $\mathbf{x}^{(r)}$, and the set S of the errors in $\mathbf{x}^{(r)}$, have at least a fraction $1-2\frac{C+1}{C-1}\epsilon_1$ overlap. Define $p_1=\frac{|L\cap S|}{|L|}$ and $p_2=\frac{|L^c\cap S|}{|L^c|}$. We must have

$$\begin{aligned} p_1 &\geq 1-2\frac{C+1}{C-1}\epsilon_0, &(7.7.8)\\ p_1|L|+p_2|L^c| &= p. &(7.7.9)\end{aligned}$$

Therefore, as $\epsilon_0\to 0$, $p_1\to 1$, and $p_2\to 0$. So, for some small enough ϵ_0, the following will eventually hold

$$(1-p_1)|L|+p_2|L^c|\leq p^*_{sr}. \tag{7.7.10}$$

Thus, according to Lemma 7.7.2, the weighted LP step of Algorithm 5 corrects all errors. Similarly, if a random set of pn bits are flipped, then when $p = (1 + \epsilon_2)p^*_{wr}$ by definition we can say that with high probability there exists a $C > 1$ so that FCP(S_1, C) holds for a random subset S_1 of the bit errors of size $p^*_{wr}n$. Therefore, using Theorem 7.6.2, it follows that the set L of most pn deviated bits in $\mathbf{x}^{(p)}$ with respect to $\mathbf{x}^{(r)}$, and the set of errors in $\mathbf{x}^{(r)}$, have at least an overlap fraction of $1 - 2\frac{C+1}{C-1}\epsilon_2$. The remainder of the proof is completed as the previous case, i.e., by applying Lemma 7.7.2. ■

Note that provable robust thresholds for a given family of codes (i.e., p^*_{sr} and p^*_{wr}) might be smaller than the actual thresholds of LP decoding (i.e., p^*_s and p^*_w). Therefore, the threshold improvement result of Theorem 7.7.1 does not directly translate to practical threshold improvement guarantees for any arbitrary code. Instead, it only proves that certain provable thresholds can be improved, as long as those thresholds are within robustness conditions for LP decoding. This is, for instance, the case for expander codes and their best existing thresholds of LP decoding, obtained in [FMS$^+$07, DDKW08]. Note that according to Theorems 7.5.3 and 7.5.4, the recovery thresholds of LP decoding for expander codes proved in [FMS$^+$07, DDKW08] are robust thresholds (see Definition 26). Therefore, according to Theorem 7.7.1, we can guarantee that our proposed algorithm has strong and weak recovery thresholds higher than those proved for LP decoding in the case of expander codes in previous works. In any case, in our preliminary simulation results, the proposed iterative algorithm shows empirical improvement over the regular LP decoding for certain families of random LDPC codes, as will be demonstrated in the next section.

7.8 Simulations

We have implemented Algorithm 5 on a random LPDC code of size $n = 1000$ and rate $R = 3/4$, and have compared the results with other existing methods. The variable

node degree is $d_v = 3$, and thus $d_c = 4$. The algorithm is compared with the mixed-integer method of Draper and Yedidia [DYW07], and the random facet-guessing algorithm of [DGW09]. The mixed-integer algorithm re-runs the LP decoding by setting integer constraints on a small subset of "least certain" bits, namely the positions where the LP minimal pseudo-codeword entries are closest to 0.5. We have taken the size of the constrained subset to be $M = 5$, which means the number of extra iterations is 32 for the mixed-integer method. We also choose to execute 20 more random iterations for facet guessing. In random facet guessing, a face (facet) of the polytope $\mathcal{P}$ is selected at random, among all the faces on which the LP minimal pseudo-codeword does not reside. Then, LP decoder is re-run with the additional constraint that the solution is on the selected face. In contrast, Algorithm 5 has only one extra iteration. All methods are simulated in MATLAB where LP decoder is implemented via the cvx toolbox [cvx]. We have plotted the BER curves versus the probability of error p in Figure 7.3. For Algorithm 5, for each p, we have experimentally found the optimal λ_1 and λ_2 by choosing the values that, on average, result in the best performance. For most of the cases the chosen values were in the ranges $-3 \leq \lambda_1 \leq -0.5$ and $1 \leq \lambda_2 \leq 3$. Observe the superior BER performance of Algorithm 5, which becomes more significant for smaller values of p. For $p = 0.11$, the BER improvement in the reweighted LP method is at least one order of magnitude. In our preliminary experimental evaluation we observe that the BER curves eventually collapse into the same curve as the LP curve, except for the reweighted LP algorithm, which is an indication of the fact that the empirical thresholds of Algorithm 5 are better than those of LP decoder and existing polynomial time post-processing methods.

7.9 Proof of Theorems

Proof of Lemma7.5.1

We first prove the following lemma.

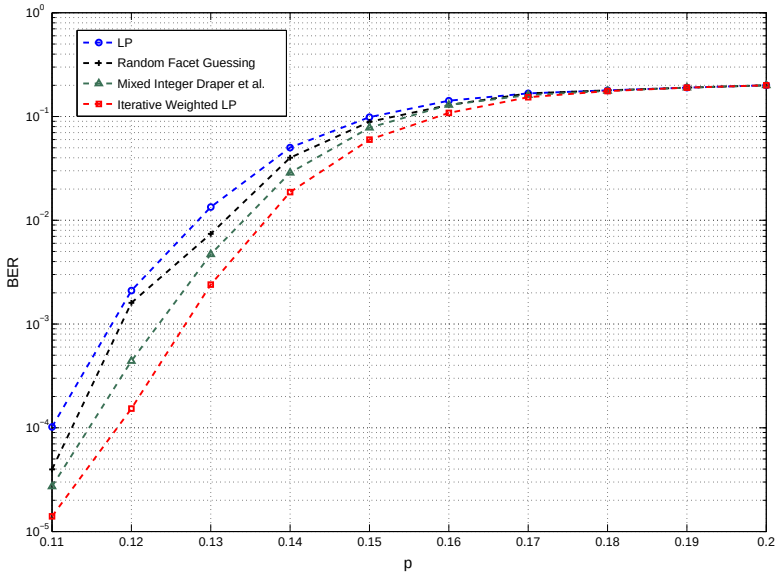


Figure 7.3: BER curves as a function of channel flip probability p, for LP decoding and different iterative schemes; random facet guessing of [DGW09], mixed integer method of [DYW07], and the suggested iterative reweighted LP of Algorithm 5. The code is a random LDPC(3,4) of length $n = 1000$.

Lemma 7.9.1. *Suppose* $\{\tau_{ij} \mid 1 \leq i \leq n, 1 \leq j \leq m\}$ *is a set of dual feasible variables on the edges of the factor graph* $\mathcal{G}$ *of the code* $\mathcal{C}$ *for some arbitrary log-likelihood vector* γ. *Then for every vector* $\mathbf{w} \in \mathcal{K}(\mathcal{C})$ *and every check node* c_j, *the following holds*

$$\sum_{v_i \in \Gamma(c_j)} w_i \tau_{ij} \geq 0. \tag{7.9.1}$$

Proof. We only use condition (i) of a feasible set of dual variables. Note that among the variable nodes in $\Gamma(c_j)$, there can be at most one node v_i with $\tau_{ij} < 0$. Let v_i be such a variable node. From the definition of $\mathcal{K}$ we can write

$$w_i \leq \sum_{i' \in \Gamma(j) \backslash i} w_{i'},$$

or equivalently:

$$\tau_{ij} w_i + \sum_{v_{i'} \in \Gamma(v_j) \backslash i} |\tau_{ij}| w_{i'} \geq 0. \tag{7.9.2}$$

Moreover, by the definition of v_i and condition (i) of the dual feasibility, we have $\tau_{i'j} \geq -\tau_{ij} = |\tau_{ij}|$ for $i' \neq i$. Therefore, replacing $\tau_{i'j}$ with $|\tau_{ij}|$ for each $i' \neq i$ does not decrease the left hand side of (7.9.2), and thus

$$\sum_{v_i \in \Gamma(c_j)} w_i \tau_{ij} \geq 0.$$

■

We now invoke Lemma 7.9.1, that for every check node c_j, $\sum_{v_i \in \Gamma(c_j)} w_i \tau_{ij} \geq 0$. If we sum these inequalities for all check nodes c_j we obtain:

$$\sum_{c_j \in X_c} \sum_{v_i \in \Gamma(c_j)} w_i \tau_{ij} = \sum_{v_i \in X_v} w_i \sum_{c_j \in \Gamma(v_i)} \tau_{ij} \geq 0,$$

when X_v and X_c are the sets of variable and check nodes, respectively. Since τ_{ij}s are feasible variables, from condition (ii) of feasibility (Definition 20), we must have $\sum_{c_j \in \Gamma(v_i)} \tau_{ij} < \gamma_i$. It then follows that

$$\sum_{v_i \in X_v} \gamma_i w_i > 0.$$

Proof of Theorem 7.5.3

We basically repeat the argument of [FMS$^+$07] with some slight adjustments. Let S be the set of flipped bits, or, interchangeably, the set of corresponding variable nodes in the factor graph $\mathcal{G}$ (we use v_i to refer to the variable node corresponding to the i^{th} bit).

Definition 27 ((δ, λ) matching from [FMS$^+$07])**.** *A (δ, λ) matching of the set S is a set M of edges of the factor graph $\mathcal{G}$, so that no two edges are connected to the same check node, every node in S is connected to at least δd_v edges of M, and every node in S' is connected to at least λd_v edges of M. Here S' is the set of variable nodes that are connected to at least $(1-\lambda)d_v + 1$ check nodes in $\Gamma(S)$.*

If there is a (δ, λ) matching on the set S, then we consider the following labeling of the edges of $\mathcal{G}$. For a check node v_j, if it is adjacent to an edge τ_{ij} in M, then set $\tau_{ij} = -x$ and $\tau_{i'j} = x$ for every other variable node $v_i' \in \Gamma(v_j)$ $i' \neq i$. Otherwise, label all of the edges adjacent to j by 0. It can be seen that for this labeling $\{\tau_{ij}\}$ satisfies condition (i) of dual feasibility (Definition 20), and furthermore:

$$\sum_{j \in \Gamma(i)} \tau_{ij} \leq \begin{cases} (1-2\delta)d_v x & i \in S \\ (1-\lambda)d_v x & i \in S^c \end{cases}. \tag{7.9.3}$$

We now take $\lambda = 2 - 2\delta + 1/d_v$. Let us define a new likelihood vector γ' by

$$\gamma' = \begin{cases} -C & i \in S \\ 1 & i \in S^c \end{cases}. \tag{7.9.4}$$

If a dual feasible set exists that satisfies the feasibility condition for the vector γ', then this implies that the FCP(S, C) holds. Now, since $C < \frac{2\delta - 1}{1-\lambda}$, if we choose x to be

$$x = \frac{1}{(1-\lambda)d_v}, \tag{7.9.5}$$

then it is clear that $(1-2\delta)d_v x < -C$. So the dual feasibility condition is satisfied, if we can construct the required (δ, λ) matching for S. From [FMS$^+$07], if $|S| \leq \frac{3\delta - 2}{2\delta - 1}\alpha$, and $\mathcal{G}$ is a bipartite $(\alpha n, \delta d_v)$ expander, the desired matching exists. This proves that FCP(S, C) holds. Since this argument holds for every set S of size $t = \frac{3\delta - 2}{2\delta - 1}\alpha$, we conclude that $\mathcal{C}$ has FCP(t, C).

Proof of Theorem 7.5.4

Consider a vector ω in the fundamental cone $\mathcal{K} = \mathcal{K}(H)$ of the parity check matrix H. Without loss of generality, we may assume that $S = \{1, 2, \cdots, t\}$. For each $1 \leq i \leq t$, let the neighbors of the variable node v_i in the (p, q)-matching on S be denoted by $c_1^i, c_2^i, \cdots, c_p^i$. The check nodes c_j^i are $p \times t$ distinct nodes. From the definition of $\mathcal{K}$, if $\omega \in \mathcal{K}$, then for each c_j^i we may write:

$$\omega_i \leq \sum_{l \in \Gamma(c_j^i) \setminus v_i} \omega_l, \quad \forall 1 \leq i \leq t \; 1 \leq j \leq p. \tag{7.9.6}$$

We add all inequalities of (7.9.6) for $1 \leq i \leq t$ and $1 \leq j \leq p$. For $i \leq t$, ω_i appears exactly p times on the left-hand side of the sum, and at most $d_v - p$ times on the right. For $i > t$, ω_i appears in at most $d_v - q$ inequalities and on the right-hand side. This comes directly from the definition of a (p, q)-matching on the set S. Therefore

$$p \sum_{i \in S} \omega_i \leq (d_v - p) \sum_{i \in S} \omega_i + (d_v - q) \sum_{i \in S^c} \omega_i, \tag{7.9.7}$$

and thus,

$$\frac{2p - d_v}{d_v - q} \sum_{i \in S} \omega_i \leq \sum_{i \in S^c} \omega_i, \tag{7.9.8}$$

which proves that $\mathcal{C}$ has the desired fundamental cone property.

Proof of Theorem 7.5.6

We denote the set of variable nodes and check nodes by X_v and X_c, respectively. For a fixed $\mathbf{w} \in [0, 1]^T$, let $\mathcal{B}$ be the set of all minimal T-local deviations, and $\mathcal{B}_i$ be the set of minimal T-local deviations that result from a skinny tree rooted at the variable node v_i. Also, assume S is the random set of flipped bits, when the flip probability is p. Interchangeably, we also use S to refer to the set of variable nodes corresponding to the indices of the flipped bits . We are interested in the probability that for all $\beta^{(\mathbf{w})} \in \mathcal{B}$, $f_C^{(S)}(\beta^{(\mathbf{w})}) > 0$. Recall that

$$f_C^{(S)}(\mathbf{x}) := \sum_{i \in S^c} x_i - C \sum_{i \in S} x_i.$$

For simplicity we denote this event by $\{f_C^{(S)}(\mathcal{B}) > 0\}$. Since the bits are flipped independently and with the same probability, we have the following union bound

$$\mathbb{P}\{f_C^{(S)}(\mathcal{B}) > 0\} = 1 - \mathbb{P}\left(\bigcup_{i=1}^{n} \{f_C^{(S)}(\mathcal{B}_i) \leq 0\}\right) \geq 1 - n\mathbb{P}\left(f_C^{(S)}(\mathcal{B}_1) \leq 0\right). \tag{7.9.9}$$

Now consider the full tree of height 2T, that is rooted at the node v_1, and contains every node u in $\mathcal{G}$ that is no more than $2T$ distant from v, i.e., $d(v_1, u) \leq 2T$. We denote this tree by $B(v_1, 2T)$. To every variable node u of $B(v_1, 2T)$, we assign

a label, $I(u)$, which is equal to $-C\omega_{h(u)}$ if $u \in S$, and is $\omega_{h(u)}$ if $u \in S^c$, where $(\omega_0, \omega_2, \cdots, \omega_{2T-2}) = \mathbf{w}$. We can now see that the event $f_C^{(S)}(\mathcal{B}_1) \geq 0$ is equivalent to the event that for all skinny subtrees $\mathcal{T}$ of $B(v_1, 2T)$ of height $2T$, the sum of the labels on the variable nodes of $\mathcal{T}$ is positive. In other words, if Γ_1 is the set of all skinny trees of height $2T$ that are rooted at v_1, then $f_C^{(S)}(\mathcal{B}_1) \leq 0$ is equivalent to:

$$\min_{\mathcal{T} \in \Gamma_1} \sum_{v \in \mathcal{T} \cap X_v} I(v) \leq 0, \tag{7.9.10}$$

where $\mathcal{T} \cap X_v$ denotes the set of nodes of $\mathcal{T}$ that are in X_v. We assign to each node u (either check or variable node) of $B(v_1, 2T)$ a random variable Z_u, which is equal to the contribution to the quantity $\min_{\mathcal{T} \in \Gamma_1} \sum_{v \in \mathcal{T} \cap X_v} I(v)$ by the offspring of the node u in the tree $B(v_1, 2T)$, and the node u itself. The value of Z_u can be determined recursively from all of its children. Furthermore, the distribution of Z_u only depends on the height of u in $B(v_1, 2T)$. Therefore, to find the distribution of Z_u, we use $X_0, X_1, \cdots, X_{T-1}$ as random variables with the same distribution as Z_u when u is a variable node (X_0 is assigned to the lowest-level variable node) and likewise $Y_1, Y_2, \cdots, Y_{T-1}$ for the check nodes. It then follows that:

$$\begin{aligned} Y_0 &= \omega_0 \eta_C, \\ X_i &= \min\{Y_i^{(1)}, \ldots, Y_i^{(d_c-1)}\} \quad \forall i > 0, \\ Y_i &= \omega_i \eta_C + X_{i-1}^{(1)} + \cdots + X_{i-1}^{(d_v-1)} \quad \forall i > 0, \end{aligned} \tag{7.9.11}$$

where $X^{(j)}$s are independent copies of a random variable X, and η_C is a random variable that takes the value $-C$ with probability p and value 1 with probability $1-p$. It follows that

$$\begin{aligned}\mathbb{P}\left(f_C^{(S)}(\mathcal{B}_1) \leq 0\right) &= \mathbb{P}\left(X_{T-1}^{(1)} + \cdots + X_{T-1}^{(d_v)} \leq 0\right) \\ &\leq (\mathbb{E}(e^{-tX_{T-1}}))^{d_v}. \end{aligned} \tag{7.9.12}$$

The last inequality is by Markov inequality and is true for all $t > 0$. The rest of the proof we bring here is basically appropriate modifications of the derivations of [ADS09] for the Laplace transform evolution of the variables X_is and Y_is, to account for a non-unitary robustness factor C. By upper bounding the Laplace transform of the variables recursively it is possible to show that:[3]

$$\begin{aligned}\mathbb{E}e^{-tX_i} \leq &\left(\mathbb{E}e^{-tX_j}\right)^{(d_v-1)^{i-j}} \\ &\prod_{0\leq k\leq i-j-1} \left((d_c-1)\mathbb{E}e^{-t\omega_{i-k}\eta_C}\right)^{(d_v-1)^k}, \end{aligned} \tag{7.9.13}$$

for all $1 \leq j \leq i < T$.

If we take the weight vector as $\omega = (1, 2, \cdots, 2^j, \rho, \rho, \cdots, \rho)$ for some integer $1 \leq j < T$, and use equation (7.9.13), we obtain:

$$\begin{aligned}\mathbb{E}e^{-tX_{T-1}} \leq &\left(\mathbb{E}e^{-tX_j}\right)^{(d_v-1)^{T-j-1}} \\ &\cdot \left((d_c-1)\mathbb{E}e^{-t\rho\eta_C}\right)^{\frac{(dv-1)^{T-j-1}-1}{d_v-2}}. \end{aligned}$$

ρ and t can be chosen to jointly minimize $\mathbb{E}e^{-tX_j}$ and $\mathbb{E}e^{-t\rho\eta_C}$ in the above, which along with (7.9.12) results in

$$\begin{aligned}\mathbb{P}(f_C^{\mathcal{S}}(\mathcal{B}_1 \leq 0)) &\leq (\mathbb{E}e^{-tX_{T-1}})^{d_v} \\ &\leq \zeta^{-d_v/(d_v-2)} \times c^{d_v(d_v-1)^{T-j-1}}, \end{aligned}$$

where $\zeta = (d_c-1)\frac{C+1}{C}(1-p)(\frac{C.p}{1-p})^{1/(C+1)}$ and $c = \zeta^{1/(d_v-2)} \min_{t\geq 0} \mathbb{E}e^{-tX_j}$. If $c < 1$, then probability of error tends to zero as stated in Theorem 7.5.6.

[3]See Lemma 8 of [ADS09], our argument is the same.

7.10 Conclusion

In this chapter, we studied post-processing techniques for detecting codewords of a linear code transmitted through a bit-flipping channel using LP decoding. We studied the output of LP decoder, and proved that, aside from the known fact that LP decoding solution is equal to the true codeword when the number of errors is less than a constant fraction of code length, it is not far from the actual codeword in other cases. This is known as the robustness of LP decoding. We characterized LP robustness for two families of LDPC codes, namely expander codes and codes with $\Omega(\log \log n)$ girth. The robustness has two implications. Firstly, if LP decoding is implemented with mismatched values of log-likelihood ratios, such as what might happen in a hardware implementation, then the solution is tolerant to certain levels of mismatch. Secondly, even when the LP decoder output is a non-integral pseudo-codeword, it can be used to derive critical information about the structure of the errors in the code block. We showed that it is possible to separate the code block into two regions with high and low error rates based on this premise. The unreliability information obtained in this way can then be used to bias the LP decoder, motivating a following weighted LP decoder. We described the whole procedure in a two-step reweighted LP decoding algorithm. We proved that for certain families of (expander) codes, the recovery thresholds of the proposed method are strictly higher than the existing thresholds for LP decoding. The exact bounds on the improvement levels are not explicit at this point and are probably loose. Future work shall address tightening those bounds, and deriving explicit numerical values for them, as well as the actual thresholds of LP decoding.

Chapter 8

Matrix Rank Minimization

X	low-rank matrix
$\mathcal{A}(\cdot)$	linear operator
n	number of rows and columns of X
m	number of rows and columns of $\mathcal{A}(X)$
δ	aspect ratio of $\mathbf{A}$, m/n
$\mu_W(\delta)$	weak recovery threshold of ℓ_1 minimization for k/n
C_S	robustness parameter for ℓ_1 minimization
$supp(\mathbf{x})$	support set of vector $\mathbf{x}$

8.1 Introduction

In this chapter, we consider the problem of low-rank matrix estimation, and apply some of the techniques developed in previous chapters to the recovery of matrices. Low-rank matrix recovery addresses the problem of estimating a high dimensional matrix of (approximately) low-rank from under-determined linear measurements. In other words, the objective is find a matrix X with the lowest possible rank that satisfies a set of linear constraints $Y = \mathcal{A}(X)$, often of smaller size than the number of matrix entries. This problem is of high practical importance, as it arises in many situations such as system identification and collaborating filtering (such as the Netflix problem).

The rank minimization (RM) problem in its full generality is NP-hard. However, a number of recent papers have demonstrated that under certain circumstances, and when the unknown matrix is sufficiently low rank, RM can be solved via convex optimization programs, mainly the nuclear norm minimization (NNM) method [Faz02, RFP10, RXH08, OH]. NNM is a natural generalization of the Basis Pursuit method from sparse vectors to low-rank matrices, and is formulated as follows:

$$\begin{aligned} &\min \|X\|_* \\ &\text{subject to } \mathcal{A}(X) = Y, \end{aligned} \tag{8.1.1}$$

where $\|\cdot\|_*$ denotes the nuclear norm operation which will be defined soon. As with compressed sensing, original results on RM were focused on random linear operators, and often in the literature Gaussian measurements are considered as a standard choice for $\mathcal{A}(\cdot)$, for which certain recovery guarantees can be asymptotic when the reconstruction method is the NNM. Interestingly, conditions similar to compressed sensing conditions exist that guarantee the tightness of NNM approach. Two main conditions that certify the success of NNM are the restricted isometry property (RIP) [RFP10] and null space conditions [RXH08], which, in the case of RM, are more advanced and obviously harder to analyze. Very recently, based on the analysis initially developed by Stojnic for sparse recovery, tight analytical thresholds for NNM have been found by Oymak et al. and Chandrasekaran et al. using an "escape through the mesh" analysis of the null space conditions when measurement operators are i.i.d. Gaussian [OH, CRPW, Sto10, Sto]. In parallel, there have also been promising results on the matrix completion problem as well, whereby one observes a subset of the entries of a low-rank matrix, rather than linear combinations [CR09, CT10]. These results are mostly based on RIP conditions and the typicality of the observation set.

As mentioned, the RM problem is often regarded as a generalization of compressed sensing, both of which are instances of the general form of linear inverse problems.

As emphasized in the first part of this dissertation, it is now well understood that, although certain random ensembles of measurement matrices (e.g., Gaussian, partial Fourier, etc.) are legitimate choices for compressed sensing, carefully designed matrices can lead to additional benefits for the sparse compression/recovery. To name a few, we discussed faster encoding time and recovery algorithms for sparse matrices using expander graphs in Chapter 2 (see also [XH07b, BI08, BI09, KXDH10]), and using other deterministic structures in Chapter 4. In addition, as pointed out in Chapters 4 and 3, higher recovery thresholds for sparse recovery can be achived using algebraic coding/decoding methods inspired by error-correcting codes (see also [PH08, AT08]). To the best of our knowledge, this point has not been fully appreciated in the RM problem, where only random measurement ensembles (mostly Gaussian) have been studied. The techniques and constructions used for low-rank estimation are far less progressed and computationally more intense. In particular, the following important question has been left unanswered:

Question. *Do alternative recovery algorithms for RM problem exist with success guarantees on certain classes of carefully designed linear measurement operators?*

8.2 Contributions

In this chapter, we introduce a new class of measurement operators for the RM problem, along with a novel recovery algorithm that is provably successful for the proposed operators, and is faster than NNM. The proposed recovery algorithm assumed that the measurement operator $\mathcal{A}(\cdot) : \mathbb{R}^{n\times n} \to \mathbb{R}^{m\times m}$ has three key properties: Hermitian, low-density, and rank expansion. The low-density property means that $\mathcal{A}(X)$ can be described as the linear combination of d linear operators $\mathcal{A}(X) = \sum_{i=1}^{d} \mathcal{A}_i(X)$, where d is a constant, and for each $1 \leq i \leq d$, the rank of $\mathcal{A}_i(X)$ is not larger than the rank of X. In fact, the low-density operators that we introduce are characterized by only

$O(mn)$ variables, as opposed to full i.i.d Gaussian linear measurements that require $(mn)^2$ variables. The interpretation of expansion is that $\mathcal{A}(\cdot)$ maps a sufficiently low-dimensional subspace to a higher-dimensional subspace, which is equivalent to mapping every positive semi-definite matrix X to a positive semi-definite matrix $\mathcal{A}(X)$ with rank greater than $c \cdot \text{rank}(X)$, where $c > 1$ is a constant.

Our contributions are thus threefold. We first prove that low-density rank expander operators exist. The operators are very similar in functionality to the minimal expanders introduced in Chapter 2. We then provide a uniqueness result, in the sense that a sufficiently low-rank positive definite (PSD) matrix X is the unique PSD solution of the equations given by a high-quality rank expander. We further propose a new recovery algorithm and provide theoretical recovery guarantees when the suggested rank expander measurement operators are exploited. This algorithm is an extension of the REVEX algorithm proposed in Chapter 2 for sparse nonnegative recovery, and is therefore named M-REVEX ("M" for Matrix).

We stress that although most of the results presented in this chapter hold for the case of PSD matrices, we also present a generalization of our results to the case of Hermitian matrices. This includes both the existence of high-quality expanders operating on Hermitian matrices, and the validity of the M-REVEX algorithm. Numerical simulations presented at the end further validate the theoretical guarantees.

8.3 Preliminaries

Let $\mathbb{S}^n$ denote the space of Hermitian matrices of size $n \times n$, and $\mathbb{S}^n_+$ denote the set of positive semi-definite (PSD) matrices. An orthogonal projection is a matrix $P \in \mathbb{S}^n_+$ with $P^2 = P$. We say that $U \in \mathbb{R}^{n_1 \times n_2}$ is a partial unitary matrix if $U^T U = I$, i.e., the columns of U form an orthonormal set. Notice that UU^T is an orthogonal projection. Let $\eta_+(X), \eta_-(X), \eta_0(X)$ denote the number of positive, negative, and zero eigenvalues of X, respectively. Also for a Hermitian matrix X, let X_- and

X_+ denote the PSD matrices induced by the negative and positive eigenvalues of X respectively (i.e., $X = X_+ - X_-$).

For a given matrix $X \in \mathbb{R}^{n_1 \times n_2}$, $\lambda_i(X)$ and $\sigma_i(X)$ denotes the ith largest eigenvalue and the ith largest singular value, respectively. The nuclear norm, spectral norm and Frobenius norm operators are denoted by $\|\cdot\|_\star$, $\|\cdot\|$, and $\|\cdot\|_F$, respectively, and are defined by $\|X\|_\star = \sum_{i=1}^{\min\{n_1,n_2\}} \sigma_i(X)$, $\|X\| = \sigma_1(X)$, and $\|X\|_\star = \left(\sum_{i=1}^{\min\{n_1,n_2\}} \sigma_i^2(X)\right)^{1/2}$. In addition, we define $\mathrm{Span}_C(X), \mathrm{Span}_R(X)$ to be the linear spaces spanned by the columns and rows of X, respectively.

A function $f : \mathbb{R}^n \to \mathbb{R}$ is called L-Lipschitz if $|f(x)-f(y)| \leq L\|x-y\|_{\ell_2}$, for every x, y. For Hermitian matrices A, B, $A \succeq B$ means that $A-B$ is positive semi-definite. For a linear operator $\mathcal{A}(\cdot)$ acting on a linear space, we denote the null space of $\mathcal{A}$ by $\mathcal{N}(\mathcal{A})$, i.e., $W \in \mathcal{N}(\mathcal{A})$ iff $\mathcal{A}(W) = 0$. We denote by $\mathcal{G}(d_1, d_2)$ the ensemble of real $d_1 \times d_2$ matrices in which the entries are i.i.d. $\mathcal{N}(0, 1)$ (zero-mean, unit variance Gaussian).

The following lemmas are crucial to the technical discussions given in the remainder of this chaper.

Lemma 8.3.1. *Let $f(X)$ be a function on matrices in the following form: $f(X) = \sum_{i=1}^m a_i\sigma_i(X)$ for some real constants $\{a_i\}_{i=1}^m$. Then $f(X)$ is a $\sqrt{\sum_{i=1}^m a_i^2}$ Lipschitz function of X.*

Lemma 8.3.2 (A Gaussian concentration inequality [LT91])**.** *Let $\mathbf{x}$ be drawn from $\mathcal{G}(n, 1)$ and $f : \mathbb{R}^n \to \mathbb{R}$ be a function with Lipschitz constant L. Then, we have the following concentration inequality*

$$\mathbb{P}(|f(x) - \mathbf{E}f(x)| \geq t) \leq 2\exp(-\frac{t^2}{2L^2}). \tag{8.3.1}$$

Lemma 8.3.3 (Weyl's Inequalities, [Bha96])**.** *Let $A, B \in \mathbb{S}^n$. Then:*

$$\lambda_j(A + B) \leq \lambda_i(A) + \lambda_{j-i+1}(B) \quad \forall\ i \leq j \tag{8.3.2}$$

$$\lambda_j(A + B) \geq \lambda_i(A) + \lambda_{j-i+n}(B) \quad \forall\ j \leq i. \tag{8.3.3}$$

8.4 Rank Expanders and Proposed Operators

Definition 28. *Let $\mathcal{A} : \mathbb{R}^{n\times n} \to \mathbb{R}^{m\times m}$ be a linear operator with $m < n$. For $d > 0, 0 \leq \varepsilon < 1, 1 \leq r_0 \leq n$, we say that $\mathcal{A}$ is an unbalanced (ε, d, r_0, n)-rank expander, if it satisfies the following conditions:*

1. *For every $X \in \mathbb{S}^n_+$, $\mathcal{A}(X) \in \mathbb{S}^m_+$.*
2. *For every $X \in \mathbb{S}^n$, rank$(\mathcal{A}(X)) \leq d \cdot$ rank(X).*
3. *For all orthogonal projections P with rank$(P) = r \leq r_0$, $rd \geq$ rank$(\mathcal{A}(P)) > (1-\varepsilon)rd$.*

This definition is inspired by the definition of unbalanced expander graphs that maintain similar properties with respect to positive vectors (instead of PSD matrices) and with ℓ_0-norm (instead of rank). An unbalanced d-regular (k, ϵ)-expander graph is a bipartite graph with n nodes on the left and m nodes on the right, and regular degree d for left-hand side nodes, such that every subset S of left nodes with $|S| \leq r_0$ has a neighborhood $\Gamma(S)$ of size at least $|\Gamma(S)| \geq (1-\epsilon)|S|d$. Unbalanced expander graphs have been proven to have elegant properties that make them suitable for sparse vector recovery (i.e., compressed sensing), in addition to being useful as parity check matrices for error-correcting codes. With that in mind, one might be inspired to generalize the notion of expander graphs to subspace (rank) expanders, in order to obtain operators that can be used in low-rank-matrix recovery. The following lemma is immediate.

Lemma 8.4.1. *If $\mathcal{A}(.)$ is an (ε, d, r_0, n)-rank expander then, for every $X \in \mathbb{S}^n_+$ with rank $r \leq r_0$, we have $rd \geq$ rank$(\mathcal{A}(X)) \geq (1-\varepsilon)rd$.*

We now move on to describe the proposed measurement structures. Afterwards, we prove that these constructions indeed result in rank expanders and that the expansion property allows us to find alternative fast reconstruction algorithms for the RM problem.

8.4.1 Proposed Measurement Operator

Let $G_1, \dots G_d \in \mathbb{R}^{m \times n}$ be matrices to be specified later. The proposed measurement operator $\mathcal{A}(\cdot)$ has the following low-density form:

$$\mathcal{A}(X) = \sum_{i=1}^{d} G_i X G_i^T, \tag{8.4.1}$$

where $X \in R^{n \times n}$. We will prove that upon appropriate choices of G_is, $\mathcal{A}(\cdot)$ is an unbalanced rank expander. It is easy to check that with this choice of $\mathcal{A}(.)$, conditions 1 and 2 of Definition 28 are immediately satisfied. Furthermore, as long as the $G_i X G_i^T$s are almost incoherent, one would expect their ranks to add up. In particular, it is easy to show that when X is fixed and $\{G_i\}_{i=1}^d$ are drawn i.i.d. from $\mathcal{G}(m, n)$, we have

$$\mathbb{P}(\text{rank}(\mathcal{A}(X)) = \min\{d \times \text{rank}(X), m\}) = 1. \tag{8.4.2}$$

However, the challenge of condition 3 is in the fact that the rank expansion property must hold for every sufficiently low-rank X. Let $X \in \mathbb{S}_+^n$ and $X^{1/2}$ denote an arbitrary square root of X (i.e., $X = X^{1/2} X^{T/2}$). Note that $\mathcal{A}(X)$ can be written in the following form:

$$\mathcal{A}(X) = \left(G_1 X^{1/2} \ \dots \ G_d X^{1/2}\right) \left(G_1 X^{1/2} \ \dots \ G_d X^{1/2}\right)^T. \tag{8.4.3}$$

It then follows that

$$\text{rank}(\mathcal{A}(X)) = \text{rank}\left(G_1 X^{1/2} \ G_2 X^{1/2} \ \dots \ G_d X^{1/2}\right). \tag{8.4.4}$$

For analyzing the rank expansion property, we can thus limit ourselves to the form (8.4.4).

8.4.2 Existence of Rank Expanders

Our goal is to prove the existence of high-quality (small ε) rank-expanders for certain regimes of d, ε, r_0, and n. Based on Lemma 8.4.1, we can restrict our attention to X

being an orthogonal projection of rank at most r_0. Our analysis is for the case when G_is are chosen i.i.d. from $\mathcal{G}(m,n)$.[1] The main existence theorem is the following:

Theorem 8.4.2 (Existence of Rank Expander)**.** *For any $0 < \varepsilon < 1$ there are constants C_1 and C_2 so that for any n and $r_0 \leq n$, whenever $m = \sqrt{C_1 C_2 n r_0}$ and $d = \sqrt{\frac{C_2 n}{C_1 r_0}}$ and $\{G_i\}_{i=1}^d s$ are independent instances of $\mathcal{G}(m,n)$, the operator $\mathcal{A}(X) = \sum_{i=1}^d G_i X G_i^T$ is an (ε, d, r_0, n) expander with probability at least $1 - \exp(-\Omega(n))$.*

Before explaining the technicalities involved in the proof of the above theorem, consider the following argument. Given $\mathcal{A} : \mathbb{R}^{n\times n} \to \mathbb{R}^{m\times m}$, suppose for all $X \in \mathbb{S}_+^n$ with $\operatorname{rank}(X) \leq r^*$, X can be uniquely decoded from $\mathcal{A}(X)$, for example, by exhaustive search. It then follows that $\mathcal{A}$ has to be injective on the restricted domain $\{X \in \mathbb{S}_+^n : \operatorname{rank}(X) \leq r^*\}$. It will soon become apparent in the sequel that given an (ε, d, r_0, n) expander, for $r^* = r_0/2$ this condition holds. A simple argument counting the degrees of freedom of the low-rank domain and the corresponding range of A reveals that the problem parameters should satisfy the following relationship:

$$m = \Omega(\sqrt{nr_0}),\ md = \Omega(n),\ dr_0 = O(m). \tag{8.4.5}$$

In fact, it turns out that Theorem 8.4.2 is true as long as (8.4.5) holds asymptotically, which implies the optimality of the number of measurements in the suggested expander operators. For the proof, we set $m = C_1 d r_0$ and $dm = C_2 n$ where $C_1 > 1$, $C_2 > 1$ will be the constants in Theorem 8.4.2.

Proof sketch of Theorem 8.4.2. The proof is based on three major technical steps:
Step 1: We consider an ε_0 cover with operator norm $\|\cdot\|$ over the set of orthogonal projections of rank $r \leq r_0$. From [Sza97], we know that there is such a cover of size at most $M = (C_0/\varepsilon_0)^{nr}$, which we denote by $\{U_i U_i^T\}_{i=1}^M \in \mathbb{S}_+^n$, with $U_i \in \mathbb{R}^{n\times r}$. Also, we first focus on a particular rank $r \leq r_0$, and later union bound the undesirable probability over all values of $r \leq r_0$.

[1] Existence of expanders using other ensembles of matrices G, and in particular sparse matrices, shall remain as an interesting open problem.

Step 2: Now consider a U_i from the ϵ_0 cover. Denote $\mathcal{B}(U_i) = [G_1 U_i \ \ldots \ G_d U_i] \in \mathbb{R}^{m \times dr}$. Since $\text{rank}(\mathcal{A}(U_i U_i^T)) = \text{rank}(\mathcal{B}(U_i))$, we can focus on $\mathcal{B}(U_i)$. Note that since U_i is a fixed partial unitary, due to the unitary invariance of i.i.d. Gaussian matrices, $\mathcal{B}(U_i)$ has i.i.d. Gaussian distribution. Now define the function $f(X) = \sum_{i=(1-\varepsilon)dr+1}^{dr} \sigma_i(X)$ for $X \in \mathbb{R}^{m \times rd}$. Notice that $f(X) > 0$ implies $\text{rank}(f(X)) > (1-\varepsilon)dr$, because it means some of the smallest εdr singular values of X are nonzero. On the other hand, $f(X)$ is a linear function of the singular values of X, and thus it satisfies the Lipschitz condition of Lemma 8.3.1. Since $\mathcal{B}(U_i)$ is Gaussian, we can apply Lemma 8.3.2 to get the following concentration bound

$$\mathbb{P}(f(\mathcal{B}(U_i)) < \delta \frac{e}{\sqrt{m}}) < \exp(-\frac{(C_3 - \delta)^2 e}{2}) \tag{8.4.6}$$

where $C_3 = 1 - \sqrt{1/C_1}$ and $e = \varepsilon drm$. Here δ serves as a safety margin, in order to account for the perturbation $P - U_i U_i^T$, when we consider a certain orthogonal projection P which we know is the proximity of some $U_i U_i^T$ of the cover, i.e., $\|P - U_i U_i^T\| < \varepsilon_0$. In other words, as will be shown in the next step, lower bounding $f(\mathcal{B}(U_i))$ certifies that $\text{rank}(\mathcal{A}(P))$ will also be large, and thus (8.4.6) basically determines a lower bound on the probability of failure. The exponent of the right-hand side of (8.4.6) is $e = \varepsilon drm = \varepsilon C_2 rn$, which is proportional to the exponent of the size of the cover $\log(M) = O(rn)$. Consequently, with careful choices of parameters, using a union bound on failures, we can make sure $f(\mathcal{B}(U)) \geq \delta \frac{e}{\sqrt{m}}$ for all $U \in \{U_i\}$ w.h.p. In particular we need:

$$(C_3 - \delta)^2 \varepsilon C_2 > (\log C_0 - \log \varepsilon_0). \tag{8.4.7}$$

Step 3: Now it remains to show that if $E = P - P_i$ is a perturbation on P_i, which makes $\text{rank}(P) \leq (1-\varepsilon)dr$, then $\|E\|$ has to be large because $f(\mathcal{B}(U_i))$ is large. In particular, showing $\|E\| > \varepsilon_0$ will finish the proof, since we know that $\|P - P_i\| \leq \varepsilon_0$. In order to show this step, we make use of Lemma 8.3.3 to find $\lambda_{i+(1-\varepsilon)dr}(\mathcal{A}(P_i)) \leq \lambda_i(\mathcal{A}(E))$, and hence to deduce that $\|\mathcal{B}(E_+^{1/2})\|_\star \geq f(\mathcal{B}(U_i))$. We carry out some more

arguments to upper bound $\|\mathcal{B}(E_+^{1/2})\|_\star$ in terms of $\|E\|$ to get a contradiction as long as $\delta\sqrt{\varepsilon} > 2\sqrt{\varepsilon_0}$ $(*)$ holds. Finally, we conclude that whenever $(*)$ and the condition (8.4.7) are satisfied, with high probability $f(\mathcal{B}(\sqrt{P})) > 0$ for all projections P with rank r, which implies $\text{rank}(\mathcal{A}(P)) > (1-\varepsilon)dr$, as desired. In particular, sufficiently large values of C_1, C_2 will do the job. Also, inequalities (8.4.5) will similarly work since increasing C_1, C_2 only improves the conditions. ■

8.5 Fast Recovery Algorithm

Before presenting the main algorithm, we will provide some results about low-rank positive-semi-definite matrix recovery. Specifically, we emphasize a uniqueness result concerning PDS matrices and the rank expander operators. Suppose a matrix $X_0 \in \mathbb{S}_+^n$ and a linear operator $\mathcal{A}(\cdot)$ are given, and we ask under what conditions X_0 is the unique (PSD) inverse image of $\mathcal{A}(X_0)$—i.e., is it possible to recover X_0 by simply characterizing the set $\{X \mid X \succeq 0, \mathcal{A}(X) = \mathcal{A}(X_0)\}$? The following lemma, which is adopted from [XT10a, OH], provides an answer to this question.

Lemma 8.5.1. *Any PSD matrix X of rank at most r is the unique PSD inverse image of $\mathcal{A}(X)$, if and only if every nonzero Hermitian $W \in \mathcal{N}(\mathcal{A})$ has at least $r+1$ negative eigenvalues.*

Now, we explain how rank expanders can facilitate the existence of the condition in Lemma 8.5.1.

Lemma 8.5.2. *Let $\mathcal{A}(.)$ be an (ε, d, r_0, n)-rank expander with $\varepsilon < 1/2$. Then for every nonzero Hermitian $W \in \mathcal{N}(\mathcal{A})$ we have that $\eta_-(W) > r_0/2$.*

Proof. For any $W \in \mathcal{N}(\mathcal{A})$, we write $W = W_+ - W_-$. Since $\mathcal{A}(W) = 0$, we have $B = \mathcal{A}(W_+) = \mathcal{A}(W_-)$. Assume $\eta_-(W) \leq r_0/2$. Let $r = \min\{\eta_-(W), \eta_+(W)\}$. Then $dr \geq \text{rank}(B)$. On the other hand $\text{rank}(W_+ + W_-) = \text{rank}(W_+) + \text{rank}(W_-) \geq 2r$, hence we have $\text{rank}(\mathcal{A}(W_+ + W_-)) \geq 2(1-\varepsilon)dr$, since $2r \leq r_0$. Note that

Algorithm 6 —M-REVEX: Reconstruct a low-rank PSD matrix X from under-determined linear measurements $Y = \sum_{i=1}^{d} A_i X A_i^*$.

Input:
Constant integer $d \geq 1$.
Matrices $A_i \in \mathbb{R}^{m \times n}, 1 \leq i \leq d$, and $Y \in \mathbb{R}^{m \times m}$.
Output:
Low-rank PSD matrix X.
Initialize
Compute $Y = S\Sigma S^*$, with S full column rank (SVD).
Set $P = I - SS^*$.
Set $Q := \text{Null}\big((PA_1)^T, \ldots, (PA_d)^T\big)^T$.
Compute $B_i = A_i Q$, and set $M = \sum_{i=1}^{d} B_i \otimes B_i$.
Find $X \in \mathbb{R}^{n \times n}$ with $\text{vec}(X) = (Q \otimes Q) M^{\dagger} vec(Y)$.

$\mathcal{A}(W_+ + W_-) = \mathcal{A}(W_+) + \mathcal{A}(W_-) = 2B$. It follows that $dr \geq \text{rank}(B) = \text{rank}(2B) \geq 2(1-\varepsilon)dr \implies 1 \geq 2(1-\varepsilon) \iff \varepsilon \geq 1/2$, which is a contradiction. ■

The combination of Lemmas 8.5.1 and 8.5.2 suggests that by using an (ε, d, r_0, n)-rank expander with $\varepsilon < 1/2$ as a measurement operator, one can guarantee that every PSD matrix X_0 of rank at most $r_0/2$ is the unique PSD solution to the measurements $\mathcal{A}(X_0)$. Therefore, every program (e.g., SDP) that can identify a point in the feasible set $\{X \mid X \succeq 0, \mathcal{A}(X) = \mathcal{A}(X_0)\}$ successfully returns X_0. Quite interestingly, for the case of high-quality expanders (small ε), we propose an alternative method for identifying one feasible point, which by the token of the aforementioned uniqueness argument can successfully recover low-rank PSD matrices. The algorithm is based on using the expansion property of the suggested linear operator, and the fact that the original matrix is low-rank. The key point is that for rank expanders, the original under-determined system can be equivalently transformed to an over-determined linear system after a few simple processing steps, which mostly involve taking SVD and finding null spaces. The new system of linear equation can then be solved by matrix inversion. This routine is described in Algorithm 6, and is in fact the generalization of a positive sparse vector recovery algorithm elaborated upon in [KXDH10]. The next theorem provides a guarantee for the success of the proposed algorithm.

Theorem 8.5.3 (PSD Recovery)**.** *If the operator $\mathcal{A}(X) = \sum_{i=1}^{d} A_i X A_i^T$ is a (ε, d, r_0, n)-rank expander with $\epsilon < 1/2$, then for every $k \leq r_0(1-\varepsilon)$, every PSD matrix X of rank k can be perfectly recovered from $\mathcal{A}(X)$ using Algorithm 6.*

In order to prove Theorem 8.5.3, we first prove the following lemma.

Lemma 8.5.4. *Suppose that the operator $\mathcal{A}(X) = \sum_{i=1}^{d} A_i X A_i^T$ is an unbalanced (ε, d, r_0, n)-rank expander with $k/(1-\varepsilon) < m < n$, where A_is are $m \times n$. Also suppose that $X \in \mathbb{S}^n$ has rank $k \leq r_0(1-\varepsilon)/2$. Further, let $\mathcal{S}$ be the linear space of all $n \times 1$ vectors u such that $Span_C\{\mathcal{A}(uu^T)\} \in Span_C\{\mathcal{A}(X)\}$. Then $dim(\mathcal{S}) < k/(1-\varepsilon)$.*

Proof. If $\dim(\mathcal{S}) \geq k/(1-\varepsilon)$, then we can find an orthonormal matrix U of size $n \times k/(1-\varepsilon)$ such that all of its columns are in $\mathcal{S}$. Therefore, by definition, $\text{Span}_C\{\mathcal{A}(UU^T)\} \in \text{Span}_C\{\mathcal{A}(X)\}$, and thus $\text{rank}(\mathcal{A}(UU^T)) \leq \text{rank}(\mathcal{A}(X)) \leq k \cdot d$. However, since $k/(1-\varepsilon) < r_0$, from the definition of rank expander, we must have $\text{rank}(\mathcal{A}(UU^T)) > d(1-\varepsilon)\text{rank}(UU^T) = k \cdot d$, which is a contradiction. ∎

Proof of Theorem 8.5.3. Let $\mathcal{S}$ be as defined in Lemma 8.5.4 with $\dim(\mathcal{S}) = r$, and $Q^{n \times r}$ be a basis for $\mathcal{S}$, and let $X = X^{1/2}X^{T/2}$. It is easy to check that the columns of $X^{1/2}$ are all in $\mathcal{S}$, and thus $X^{1/2} = QV^{1/2}$, for some unknown $V^{1/2}$ of size $r \times k$. Therefore, we can write:

$$Y = \mathcal{A}(X) = \sum_{i=1}^{d} A_i Q V^{1/2} V^{T/2} Q^T A_i^T \tag{8.5.1}$$

or equivalently

$$\text{vec}(Y) = \left(\sum_{i=1}^{d} B_i \otimes B_i^T \right) \text{vec}(V) \tag{8.5.2}$$

where $B_i = A_i Q$. (8.5.2) is a linear system of equations with m^2 equations and r^2 unknowns. Moreover, from Lemma 8.5.4, we know that $r < k/(1-\varepsilon) < m$. In addition, from Lemma 8.5.2, the solution to the system of linear equations in (8.5.2) is unique and must be $V = X$, since otherwise $X - V$ is in the null space of $\mathcal{A}(\cdot)$ and

has at most $r < r_0/2$ negative eigenvalues. Therefore, (8.5.2) is an over-determined linear system and can be solved by matrix inversion, as given in line 11 of Algorithm 6. ■

8.6 Extension to Hermitians

In this section, we briefly extend the results of the previous sections to the case of Hermitian matrices. Specifically, we prove the existence of expanders for Hermitian matrices (rather than only PSD matrices, as was the case discussed previously), and then state a theorem certifying the success of Algorithm 6 for these classes of low-rank matrices.

8.6.1 Expansion

Lemma 8.6.1 (Expansion for Hermitians)**.** *Assume $\mathcal{A}$ is a (ε, d, r_0, n) expander and let X be Hermitian with rank$(X) \leq r_0$. Then, rank$(\mathcal{A}(X)) \geq (1 - 4\varepsilon)d \cdot$ rank(X).*

The formal proof of the above lemma is skipped, but it is mostly based on dimension-counting arguments and the fact that $\text{rank}(A+B) = \text{rank}(A) + \text{rank}(B)$ is equivalent to $\text{Span}_C(A) \cap \text{Span}_C(B) = \emptyset$ and $\text{Span}_R(A) \cap \text{Span}_R(B) = \emptyset$ (see, e.g., [Cal98]).

8.6.2 Recovery

Assume $X \in \mathbb{S}$ with $\text{rank}(X) = r \leq r_0$. With Gaussian measurements, $\text{rank}(\mathcal{A}(X)) = rd$ almost surely, due to (8.4.2) and the result of [Cal98]. Notice that if the eigenvalue decomposition of X is $X = \sum_{i=1}^{r} \lambda_i u_i u_i^T$, then $\mathcal{A}(X) = \lambda_i \sum_{i=1}^{r} \mathcal{A}(u_i u_i^T)$, and using $\text{rank}(\mathcal{A}(X)) = r \times d$ and $\text{rank}(\mathcal{A}(u_i u_i^T)) \leq d$, it follows that $\text{Span}_C(\mathcal{A}(u_i u_i^T)) \subset \text{Span}_C(\mathcal{A}(X))$ for all $i \leq r$. Consequently, similar to the case of PSD matrices, we need to find the space of u such that $\text{Span}_C(\mathcal{A}(uu^T)) \subset \text{Span}_C(\mathcal{A}(X))$. All technical steps follow identically and similar to the PDS case, and we can assert that sufficiently

low-rank Hermitian solutions to (8.5.2) are unique, as follows.

Theorem 8.6.2 (Hermitian Recovery)**.** *Let $X_0 \in \mathbb{S}$ with rank$(X_0) \leq r_0(1-\varepsilon)$. Suppose $\mathcal{A}(\cdot)$ is as described in Theorem 8.4.2. With probability at least $1-\exp(-\Omega(n))$, X_0 can be perfectly recovered from $\mathcal{A}(X_0)$ by Algorithm 6.*

Note that although the same algorithm works for both cases of PSD and Hermitian matrices, there is a significant difference. In Theorem 8.5.3, all $X \in \mathbb{S}_+^n$ with sufficiently low ranks are recoverable, whereas a similar fact is true for almost all Hermitian matrices of low rank. These notions are often distinguished in the literature by the terms "strong" recovery and "weak" recovery, respectively. Furthermore, note that for the case of PSD recovery, one can alternatively use convex optimization to find the unique PSD solution. However for the recovery of Hermitians, we do not know of any other method but our proposed Algorithm 6.

8.7 Simulation Results

Numerical simulations were performed to verify the validity of Algorithm 6. We used $n = 50$, linear measurement operators in the form of (8.4.1) with $d = 2, 3, 4$, and two types of distributions for G_is: 1) i.i.d Gaussian matrices, and 2) sparse matrices where every row of each G_i has exactly one 1 in a random location. We did not explicitly prove that the sparse constructions are also rank expanders. However, such low-density structures are of high practical interests. The resulting curves of successful recovery thresholds are given in Figure 8.1. In all of our simulations for Gaussian matrices, the transition between successful recovery and failure was very sharp, i.e., either failed all times or succeeded, depending on the number of measurements and the rank of X. Figure 8.1 illustrates the empirical transition phase or the recovery threshold of Algorithm 6. The corresponding curves for sparse matrices are also shown in Figure 8.1. The transition for sparse matrices was not as sharp as for Gaussians. On the same curves, the performance of the standard trace minimization

with nonnegativity (PSD) constraint is also displayed. Observe that the performance of the proposed algorithm is very comparable to the convex relaxation method. In addition, the curves for sparse measurements collapse into the same curves as the dense measurements for sufficiently large m. In practice, Algorithm 6 is much faster than NNM. To give an example, for simulations in MATLAB on a 2.4 GHz Intel Core i5 with 4 GB RAM, and for the case of $d = 2, m = 39, k = 10$ with Gaussian G_is, each reconstruction took on average 34s for the NNN, but only about 0.05s for Algorithm 6. For sparse structures, each reconstruction took on average about 1.8s for NNM and 0.05s for Algorithm 6. NNM was solved via the SEDUMI toolbox.

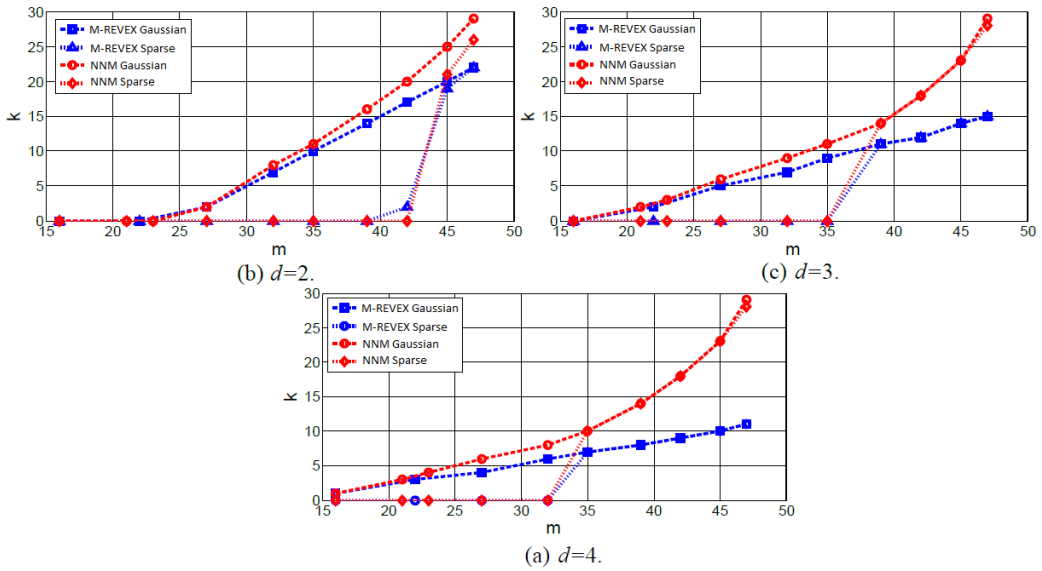


Figure 8.1: Empirical recovery thresholds of Algorithm 6 and NNM for 50×50 matrices with linear operators. Measurements are $m \times m$, and k is the rank.

Chapter 9

Future Work

A few questions related to the results of this thesis that are worth further explorations are as follows:

Minimal expander-graphs discussed in Chapter 2 are based on random perturbations. Constructing binary matrices that have complete rank proportional to the signal dimension is an open problem. Binary matrices are easier to store and implement than structures with perturbation. Furthermore, determining whether expander-graph based structures or their perturbed variations satisfy RIP-2 is an interesting theoretical problem.

Summary-based structures and the SIR algorithm discussed in Chapter 4 are very suitable for sparse regression problems with extremely large dimensions. Therefore, it is of high practical value to come up with ways for making the proposed sublinear time algorithms more robust to noise, and providing theoretical justifications. Furthermore, numerical simulations that verify the usefulness of the algorithm for model selection and statistical inference problems are in order. In [KKH11], we have reported a preliminary application of this approach to identifying a hypothetical tree mixture model from restricted queries. More practical applications, such as partial political ranking and estimation of graphical models (such as Markov random fields), can be immediately investigated.

Model-based sparse reconstruction algorithms are very powerful, and are becoming subjects of more inspection, thanks to the availability of more advanced statistical signal models. Extensions and applications of our proposed regularized Basis Pursuit algorithm have already been reported (see, e.g., [KOH12, LV, OKH11] for theoretical results, and [LLAV11, Das10] for applications in fMRI and particle filtering for machine vision). The results are further improvable.

Designing iterative algorithms that can universally improve upon Basis Pursuit remains an open problem. In our analysis of iterative BP given in Chapter 6, exact threshold improvements are missing. In a more recent work, we have addressed this issue and have found explicit upper bounds that closely estimate the exact threshold improvements [KTH12]. Stronger results and tighter approximations are in order. In general, precise characterization of the dynamics of ℓ_1 regression beyond the recovery threshold results of [Don04] and robustness analysis of Weiyu [XH09] is a cumbersome and highly motivated problem. Recent work that has addressed this includes [CRPW, XT10b, BM].

Exact threshold computations for the proposed reweighted LP algorithm for error-correcting code is even more challenging than the reweighted Basis Pursuit analysis. For the subspace expander operators proposed in Chapter 8, we did not prove the existence of such structures with sparse matrix multipliers (G_i). However simulations verified that the REVEX algorithm is successful for sparse operators as well. Further, it would be very interesting to design greedy low-rank-matrix estimation algorithms that resemble other types of combinatorial methods, such as the bit-flipping algorithm or message-passing methods for expander graphs. Such reconstruction schemes would be highly efficient and preferable to SDP type methods for rank minimization.

Developing ways to efficiently verify the success conditions, such as null space property or RIP for given matrices, is a high-profile research direction. We suggested that girth can be used as a deterministic and efficiently verifiable guarantee (see Chapter 3). However, this criterion is only applicable to sparse LDPC matrices.

Furthermore, it does not provide a sufficient condition for successful regression using Basis Pursuit. As a first step to find more universal measures, perhaps it is worthwhile to look into the model-based sparse recovery problems and identify easily verifiable goodness measures for the measurement matrix in those contexts. Furthermore, as the goodness of a matrix for Basis Pursuit is completely characterized by its null space, there might be ways for designing elegant measurement matrices by specifying the null space directly. A paper of Stojnic et al. has studied this to a limited extent by considering random matrix ensembles with Bernoulli null spaces [Sto09]. Again, pursuing such a direction is perhaps easier for model-based sparsity problems.

Finally, new applications for sparse estimation always lead to new research directions for compressed sensing, both from experimental and theoretical points of view. New applications introduce new constraints, requiring novel encoding/decoding technologies.

Chapter 10

Appendix

10.1 Elementary Bounds on Binomial Coefficients

For each $\beta \in (0,1)$, define the binomial entropy $H(\beta) = -\beta \log_2 \beta - (1-\beta)\log_2(1-\beta)$ (and $H(0) = H(1) = 0$ by continuity). We make use of the following standard bounds on the binomial coefficients from [CT06b]:

$$n\left[H\left(\frac{k}{n}\right) - \frac{\log_2(n+1)}{n}\right] \;\leq\; \log_2 \binom{n}{k} \;\leq\; n\left[H\left(\frac{k}{n}\right) + \frac{\log_2(n+1)}{n}\right]. \quad (10.1.1)$$

10.2 Hall's Matching Theorem

Hall's matching (marriage) theorem gives a necessary and sufficient condition for the existence of a perfect matching in a bipartite graph. Suppose $(X, Y, \mathcal{E})$ is a bipartite graph with sets of right and left nodes X, Y with $|X| \leq |Y|$. A perfect matching from X to Y is a one-to-one mapping from X to Y using the edges in $\mathcal{E}$. According to Hall's theorem (see, e.g. [HV50]):

$$\text{A perfect matching from } X \text{ to } Y \text{ exists} \iff \forall S \subseteq X, \Gamma(S) \geq |S|$$

where $\Gamma(S)$ is the neighborhood set of S in Y. This theorem can be generalized. A perfect c-matching is an invertible one-to-c mapping from X to Y using the edges in

$\mathcal{E}$, meaning that every node in X is mapped to exactly c neighboring nodes in Y, and no two nodes of X have a common node in their mappings. A generalization of Hall's theorem states that:

$$\text{A perfect } c\text{-matching from } X \text{ to } Y \text{ exists } \iff \forall S \subseteq X, \Gamma(S) \geq c|S|.$$

10.3 Restricted Isometry Property(RIP)

A matrix $\mathbf{A}^{m\times n}$ satisfies the RIP(p, k, δ) for integer k, $p > 0$ and $0 < \delta < 1$, if for all k-sparse vectors $\mathbf{x}$, the following holds:

$$(1-\delta)\|\mathbf{x}\|_p^p \leq \|\mathbf{A}\mathbf{x}\|_p^p \leq (1+\delta)\|\mathbf{x}\|_p^p. \tag{10.3.1}$$

In other words, RIP condition for $\mathbf{A}$ means that the (ℓ_p) topology of the set of sparse vectors in $\mathbb{R}^n$ is approximately preserved under the projection to $\mathbb{R}^m$ by $\mathbf{A}$. Therefore, it should be easier to invert mappings that satisfy stronger isometry properties, and equivalently find the sparse solution of a linear equation. The RIP condition is often only referred to by the norm parameter p, when clear from the context. Sparse recovery using Basis Pursuit is mostly related to RIP-2, as Candés et al. proved that it ensures exact recovery of sparse signals and stable recovery in the present of noise [CT05, CRT06c]. RIP-$(2, k, \delta)$ is related to the coherence measure of $\mathbf{A}$ and is equivalent to all singular values of all of the $m \times k$ sub-matrices of $\mathbf{A}$ being bounded away from zero, i.e.,

$$\min_{S\subset\{1,2,\ldots,n\},|S|=k} \sigma_{\min}(\mathbf{A}_S) \geq \delta. \tag{10.3.2}$$

10.4 RIP-1 for Expanders

We present a simple argument to show that the adjacency matrix $\mathbf{A}$ of a (k, ϵ) expander graph with $\epsilon < 1/2$ has RIP-1 property, which means that for every k-sparse

vector $\mathbf{x}$ and suitable constants c_1, c_2, the ℓ_1 norm of $\|\mathbf{Ax}\|_1$ is close to the norm of $\mathbf{x}$:

$$(1 - c_1)\|\mathbf{x}\|_1 \leq \|\mathbf{Ax}\|_1 \leq (1 + c_2)\|\mathbf{x}\|_1, \tag{10.4.1}$$

where c_1 and c_2 are two constants depending on the expansion coefficient.

It should be acknowledged that Berinde et al. [BGI+08] already prove this property, generally for p norms where $p \leq 1 - 1/\log n$. The argument we present here is arguably simpler and easily extends to the case where the matrix is perturbed in the nonzero entries.

Consider $\mathbf{A}$ to be the adjacency matrix of a (k, ϵ) unbalanced expander for $\epsilon < 1/2$. Consider S the support set of a k-sparse $\mathbf{x}$. By Hall's theorem, since every set S of size up to k has $d(1 - \epsilon)|S|$ neighbors, there must exist a partial $d(1 - \epsilon)$-matching, i.e., one in which every node in S can be matched to $d(1 - \epsilon)$ unique neighbors. Decompose the measurement matrix

$$\mathbf{A} = \mathbf{A}_M + \mathbf{A}_C, \tag{10.4.2}$$

where $\mathbf{A}_M$ is supported on the partial $d(1 - \epsilon)$-matching (i.e., every row has one nonzero entry and every column has $d(1-\epsilon)$ nonzero entries). The remainder matrix $\mathbf{A}_C$ has ϵd nonzero entries in each column; notice that the decomposition is adapted to the support of the vector $\mathbf{x}$. By the triangle inequality:

$$\|\mathbf{Ax}\|_1 \geq \|\mathbf{A}_M\mathbf{x}\|_1 - \|\mathbf{A}_C\mathbf{x}\|_1. \tag{10.4.3}$$

It is easy to see that

$$\|\mathbf{A}_M\mathbf{x}\|_1 \geq d(1 - \epsilon)\|\mathbf{x}\|_1, \tag{10.4.4}$$

since $\mathbf{A}_M\mathbf{x}$ is a vector that contains $d(1-\epsilon)$ copies of each entry of $\mathbf{x}$. Also since each column of $\mathbf{A}_C$ contains ϵd nonzero entries,

$$\|\mathbf{A}_C\mathbf{x}\|_1 \leq \epsilon d\|\mathbf{x}\|_1, \tag{10.4.5}$$

since each entry of $\mathbf{A}_C\mathbf{x}$ is a summation of the coefficients of $\mathbf{x}$, and $\|\mathbf{A}_M\mathbf{x}\|_1$ is also a summation in which each entry of $\mathbf{x}$ appears $d\epsilon$ times. A similar argument implies

the upper bound:

$$\|\mathbf{A}\mathbf{x}\|_1 \leq d\|\mathbf{x}\|_1. \tag{10.4.6}$$

Therefore, putting these together, we obtain:

$$(1-2\epsilon)\|\mathbf{x}\|_1 \leq \|\mathbf{A}\mathbf{x}\|_1 \leq d\|\mathbf{x}\|_1. \tag{10.4.7}$$

Bibliography

[ADS09] S. Arora, C. Daskalakis, and D. Steurer. Message-passing algorithms and improved LP decoding. In *Proceedings of 41st Annual ACM Symposium on Theory of Computing, Bethesda, MD, USA, May 31–June 2*, 2009.

[AM11] A. Amini and F. Marvasti. Deterministic construction of binary, bipolar and ternary compressed sensing matrices. *IEEE Transactions on Information Theory*, 57(4):2360–2370, 2011.

[AMM12] A. Amini, V. Montazerhodjat, and F. Marvast. Matrices with small coherence using p-ary block codes. *IEEE Transactions on Signal Processing*, 60(1):172–181, 2012.

[Ari09] E. Arikan. Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels. *IEEE Transactions on Information Theory*, 55:3051–3073, 2009.

[AS92] F. Affentranger and R. Schneider. Random projections of regular simplices. *Discrete and Computational Geometry*, 7:219–226, 1992.

[AT05] A. Altman and M. Tennenholtz. Ranking systems: The pagerank axioms. In *Proceedings of the 6th ACM Conference on Electronic Commerce, Vancouver, BC,Canade*, 2005.

[AT08] M. Akcakaya and V. Tarokh. A frame construction and a universal distortion bound for sparse representations. *IEEE Transactions on Signal Processing*, 56(6):2443–2450, 2008.

[BCDH10] R. G. Baraniuk, V. Cevher, M. F. Duarte, and C. Hegde. Model-based compressive sensing. *IEEE Transactions on Information Theory*, 56(4):1982–2001, 2010.

[BCJ10] W. U. Bajwa, R. Calderbank, and S. Jafarpour. Why Gabor frames? Two fundamental measures of coherence and their role in model selection. (Preprint), 2010.

[BCT10] J. Blanchard, C. Cartis, and J. Tanner. Compressed sensing: How sharp is the restricted isometry property? *SIAM Review* 53:105–125, 2010.

[BD08] T. Blumensath and M. E. Davies. Iterative thresholding for sparse approximations. *The Journal of Fourier Analysis and Applications*, 14(5):629–654, December 2008.

[BDDW08] R. G. Baraniuk, M. Davenport, R. A. DeVore, and M. Wakin. A simple proof of the restricted isometry property for random matrices. *Constructive Approximation*, 28:253–263, 2008.

[BEZ08] A. M. Bruckstein, M. Elad, and M. Zibulevsky. A nonnegative and sparse enough solution of an underdetermined linear system of equations is unique. *IEEE Transactions on Information Theory*, 54(11):4813–4820, November 2008.

[BGI+08] R. Berinde, A. Gilbert, P. Indyk, H. Karloff, and M. Strauss. Combining geometry and combinatorics: A unified approach to sparse signal recovery. In *46th Annual Allerton Conference, September 23–26*, 2008.

[BGT93] C. Berrou, A. Glavieux, and P. Thitimajshima. Near shannon limit error-correcting coding and decoding: Turbo-codes. 1. In *IEEE International Conference on Communications, ICC 93, Geneva, Switzerland*, 1993.

[Bha96] R. Bhatia. *Matrix Analysis (Graduate Texts in Mathematics)*. Springer, 1996.

[BHRN08] W. U. Bajwa, J. Haupt, G. Raz, and R. Nowak. Compressed channel sensing. In *42nd Annual Conference on Information Sciences and Systems, CISS*, 2008.

[BHSN10] W. U. Bajwa, J. Haupt, A. M. Sayeed, and R. Nowak. Compressed channel sensing: A new approach to estimating sparse multipath channels. *Proceedings of the IEEE*, 98(6):1058 – 1076, 2010.

[BI08] R. Berinde and P. Indyk. Sparse recovery using sparse matrices. Technical report, Computer Science and Artificial Intelligence Laboratory (MIT-CSAIL-TR), January 10, 2008.

[BI09] R. Berinde and P. Indyk. Sequential sparse matching pursuit. In *Allerton Conference*, 2009.

[BIPW10] K. Do Ba, P. Indyk, E. Price, and D. Woodruff. Lower bounds for sparse recovery. In *Proceedings of ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2010.

[BIR08] R. Berinde, P. Indyk, and M. Ruzic. Practical near-optimal sparse recovery in the l_1 norm. In *Proceedings of 46th Annual Alletron Conference*, 2008.

[BM] M. Bayati and A. Montanari. The LASSO risk for Gaussian matrices. arXiv:1008.2581.

[BM01] D. Burshtein and G. Miller. Expander graph arguments for message passing algorithms. *IEEE Transactions on Information Theory*, 47:782–790, February 2001.

[BW09] R. G. Baraniuk and M. B. Wakin. Random projections of smooth manifolds. *Journal of Foundations of Computational Mathematics*, 9(1), 2009.

[Cal98] D. Callan. When is 'rank' additive? *The College Mathematics Journal*, 29(2):145–147, 1998.

[CB83] N. R. Chapman and I. Barrodale. Deconvolution of marine seismic data using the ℓ_1 norm. *Geophysics Journal of the Royal Astronomical Society*, 72:93–100, 1983.

[CC06] M. Chertkov and V. Y. Chernyak. Loop calculus helps to improve belief propagation and linear programming decoding of LDPC codes. In *Allerton Conference on Communications, Control and Computing*, 2006.

[CD94] S. Chen and D. Donoho. Basis pursuit. In *Twenty-Eighth Asilomar Conference on Signals, Systems and Computers*, 1994.

[CDD08] A. Cohen, W. Dahmen, and R. DeVore. Compressed sensing and best k-term approximation. *Journal of American Mathematical Society*, 22:211–231, 2008.

[CDS01] S. Chen, D. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Review*, 43(1):129–159, 2001.

[Cev09] V. Cevher. Learning with compressible priors. In *Neural Information Processing Systems (NIPS)*, 2009.

[CH06] J. Chen and X. Huo. Theoretical results on sparse representations of multiple-measurement vectors. *IEEE Transactions on Signal Processing*, 54(12):4634–4643, 2006.

[CHJ10] R. Calderbank, S. Howard, and S. Jafarpour. Construction of a large class of deterministic sensing matrices that satisfy a statistical isometry property. *IEEE Journal of Selected Topics in Signal Processing, Special Issue on Compressed Sensing*, 4(2):358–374, 2010.

[CM04] G. Cormode and S. Muthukrishnan. Improved data stream summaries: The count-min sketch and its applications. In *Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS)*, 2004.

[CM05] G. Cormode and S. Muthukrishnan. Towards an algorithmic theory of compressed sensing. Technical report, 2005.

[CM06] G. Cormode and S. Muthukrishnan. Combinatorial algorithms for compressed sensing. In *Conference on Information Sciences and Systems (CISS)*, 2006.

[CPR07] J. Coates, Y. Pointurier, and M. Rabbat. Compressed network monitoring. In *Proceedings of IEEE Workshop on Statistical Signal Processing, Madison, WI, USA, August*, 2007.

[CR09] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9(6):717–772, 2009.

[CR11] E. J. Candès and B. Recht. Simple bounds for low-complexity model reconstruction. (Preprint), 2011.

[CRPW] V. Chandrasekaran, B. Recht, P. Parrilo, and A. Willsky. The convex geometry of linear inverse problems. arXiv:1012.0621v1.

[CRT06a] E. Candès, J. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 59(8):1207–1223, August 2006.

[CRT06b] E. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509, February 2006.

[CRT06c] E. J. Candѐs, J. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 59 (8):1207–1223, 2006.

[CRVW02] M. Capalbo, O. Reingold, S. Vadhan, and A. Wigderson. Randomness conductors and constant-degree lossless expanders. In *Proceedings of 34th Annual ACM Symposium on Theory of Computing (STOC)*, 2002.

[CSW10] V. Chandar, D. Shah, and G. W. Wornell. A simple message-passing algorithm for compressed sensing. In *IEEE International Symposium on Information Theory (ISIT)*, 2010.

[CT05] E. J. Candès and T. Tao. Decoding by linear programming. *IEEE Transactions on Information Theory*, 51(12):4203–4215, 2005.

[CT06a] E. J. Candès and T. Tao. Near-optimal signal recovery from random projections: universal encoding strategies. *IEEE Transactions on Information Theory*, 52(12):5406–5425, 2006.

[CT06b] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, 2006.

[CT07] E. J. Candès and T. Tao. The Dantzig selector: Statistical estimation when p is much larger than n. *Annals of Statistics*, 35:2313–2351, 2007.

[CT10] E. J. Candès and T. Tao. The power of convex relaxation: Near-optimal matrix completion. *IEEE Transactions on Information Theory*, 56(5):2053–2080, 2010.

[CV08] C. Chen and P. P. Vaidyanathan. Compressed sensing in MIMO radar. In *Asilomar Conference on Signal, Systems, and Computers*, 2008.

[cvx] cvx toolbox webpage: http://cvxr.com/cvx/.

[CWB08] E. Candès, M. Wakin, and S. Boyd. Enhancing sparsity by reweighted l_1 minimization. *Journal of Fourier Analysis and Applications*, 14:877–905, 2008.

[CY08] R. Chartrand and W. Yin. Iteratively reweighted algorithms for compressive sensing. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Las Vegas, NV, USA*, April 2008.

[Das10] S. Das. *Particle Filtering on Large Dimensional State Spaces and Applications in Computer Vision.* PhD thesis, Iowa State University, 2010.

[DDFG10] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Gntrk. Iteratively reweighted least squares minimization for sparse recovery. *Communications on Pure and Applied Mathematics*, 63(1):1–38, 2010.

[DDKW08] C. Daskalakis, A. G. Dimakis, R. M. Karp, and M. J. Wainwright. Probabilistic analysis of linear programming decoding. *IEEE Transactions on Information Theory*, 54(8), 2008.

[DDT+08] M. F. Duarte, M. A Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk. Single-pixel imaging via compressive sampling. *IEEE Signal Processing Magazine*, 25(2):83–91, 2008.

[DET06] D. Donoho, M. Elad, and V.N. Temlyakov. Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Transactions on Information Theory*, 52(1):6–18, 2006.

[DeV06] R. A. DeVore. Optimal computation. *In Proceedings of the International Congress of Mathematicians, Madrid, Spain*, 2006.

[DeV07] R. A. DeVore. Deterministic constructions of compressed sensing matrices. *Journal of Complexity*, 23(4–6), 2007.

[DGW09] A. G. Dimakis, A. A. Gohari, and M. Wainwright. Guessing facets: Polytope structure and improved LP decoder. *IEEE Transactions on Information Theory*, 55(8):1369–1373, 2009.

[DH01] D. L. Donoho and X. Huo. Uncertainty principles and ideal atomic decomposition. *IEEE Transactions on Information Theory*, 47(7):2845–2862, 2001.

[DJM11] D. Donoho, I. Johnstone, and A. Montanari. Accurate prediction of phase transitions in compressed sensing via a connection to minimax denoising. (Preprint), 2011.

[DMA97] G. Davis, S. Mallat, and M. Avellaneda. Adaptive greedy approximations. *Journal of Constructive Approximation*, 13:57–98, 1997.

[DMM09] D. L. Donoho, A. Maleki, and A. Montanari. Message passing algorithms for compressed sensing. In *Proceedings of the National Academy of Science*, 2009.

[DMP11] W. Dai, O. Milenkovic, and H. Pham. Structured sublinear compressive sensing via dense belief propagation. (Preprint), 2011.

[DMSB08] W. Dai, O. Milenkovi, M. A. Sheikh, and R. G. Baraniuk. Probe design for compressive sensing DNA microarrays. In *IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 2008.

[DMT07] C. Dwork, F. McSherry, and K. Talwar. The price of privacy and the limits of LP decoding. In *Symposium on Theory of Computing (STOC), San Diego, CA, USA*, June, 2007.

[Don04] D. Donoho. For most large underdetermined systems of linear equations the minimal l1-norm solution is also the sparsest solution. In *Communications on Pure and Applied Mathematics*, 2004.

[Don06a] D. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006.

[Don06b] D. Donoho. High-dimensional centrally symmetric polytopes with neighborliness proportional to dimension. *Discrete and Computational Geometry*, 102(27):617–652, 2006.

[DSMB09] W. Dai, M. Sheikh, O. Milenkovic, and R. Baraniuk. Compressive sensing DNA microarrays. *EURASIP Journal on Bioinformatics and Systems Biology (available online at http://www-ece.rice.edu)*, 2009.

[DSV] A. G. Dimakis, R. Smarandache, and P. O. Vontobel. LDPC codes for compressed sensing. arXiv:1012.0602.

[DT05a] D. Donoho and J. Tanner. Neighborlyness of randomly-projected simplices in high dimensions. *Proceedings of the National Academy of Sciences*, 102(27):9452–9457, 2005.

[DT05b] D. Donoho and J. Tanner. Sparse nonnegative solution of underdetermined linear equations by linear programming. *Proceedings of the National Academy of Sciences*, 102(27):9446–9451, 2005.

[DT06] D. Donoho and J. Tanner. Thresholds for the recovery of sparse solutions via l1 minimization. In *Proceedings of the Conference on Information Sciences and Systems*, 2006.

[DT10] D. Donoho and J. Tanner. Counting the faces of randomly-projected hypercubes and orthants, with applications. *Discrete and Computational Geometry*, 43(3):522–541, 2010.

[DTDS06] D.. Donoho, Y. Tsaig, I. Drori, and J. Starck. Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit. Technical report, Stanford University, 2006.

[duk] Bayesian compressed sensing online resources:. http://people.ee.duke.edu/l̃carin/BCS.html.

[DV09] A. G. Dimakis and P. O. Vontobel. LP decoding meets LP decoding: A connection between channel coding and compressed sensing. In *Allerton Conference on Communications, Control and Computing*, 2009.

[DWB06] M. Davenport, M. Wakin, and R. Baraniuk. Detection and estimation with compressive measurements. Technical report, Rice ECE Department, November 2006.

[DYW07] S. C. Draper, J. S. Yedidia, and Y. Wang. ML decoding via mixed-integer adaptive linear programming. In *IEEE International Symposium on Information Theory (ISIT)*, 2007.

[Eld09] Y. C. Eldar. Compressed sensing of analog signals in shift-invariant spaces. *IEEE Transactions on Signal Processing*, 57(8):2986–2997, 2009.

[ES05] S. Erickson and C. Sabatti. Empirical Bayes estimation of a sparse vector of gene expression. In *Statistical Applications in Genetics and Molecular Biology*, 2005.

[Faz02] M. Fazel. *Matrix Rank Minimization with Applications*. PhD thesis, Stanford University, 2002.

[Fel03] J. Feldman. *Decoding Error-Correcting Codes via Linear Programming*. PhD thesis, Massachusetts Institute of Technology, 2003.

[FKW02] J. Feldman, D. R. Karger, and M. J. Wainwright. Linear programming-based decoding of Turbo-like codes and its relation to iterative approaches. In *Proceedings of the 40th Annual Allerton Conference on Communication, Control, and Computing*, 2002.

[FMS+07] J. Feldman, T. Malkin, R. A. Servedio, C. Stein, and M. J. Wainwright. LP decoding corrects a constant fraction of errors. *IEEE Transactions on Information Theory*, 53(1):82–89, 2007.

[FN03] A. Feuer and A. Nemirovski. On sparse representation in pairs of bases. *IEEE Transactions on Information Theory*, 49(6):1579–1581, 2003.

[Fos01] M. P. C. Fossorier. Iterative reliability-based decoding of low-density parity check codes. *IEEE Transactions on Information Theory*, 19(5):908–917, 2001.

[FR08] M. Fornasier and H. Rauhut. Iterative thresholding algorithms. *Applied and Computational Harmonic Analysis*, 25(2):187–208, 2008.

[Fre98] B. J. Frey. *Graphical Models for Machine Learning and Digital Communication*. M.I.T Press, Cambridge, MA, 1998.

[FRG07] A. K. Fletcher, S. Rangan, and V. K. Goyal. On the rate-distortion performance of compressed sensing. In *IEEE International Conference on*

Acoustics, Speech, and Signal Processing (ICASSP), Honolulu, Hawaii, USA, April 2007.

[FWK05] J. Feldman, M. J. Wainwright, and D. R. Karger. Using linear programming to decode binary linear codes. *IEEE Transactions on Information Theory*, 51(3):954–972, 2005.

[Gal63] R. G. Gallager. *Low-Density Parity-Check Codes*. M.I.T. Press, Cambridge, MA, 1963.

[GI10] A. Gilbert and P. Indyk. Sparse recovery using sparse matrices. *Proceedings of the IEEE*, 98(6):937–947, 2010.

[GLR08] V. Guruswami, J. R. Lee, and A. Razborov. Almost Euclidean subspaces of l_1^n via expander codes. In *Proceedings of Symposium on Discrete Algorithms (SODA)*, 2008.

[GLW08] V. Guruswami, J. R. Lee, and A. Wigderson. Euclidean sections of l_1^n with sublinear randomness and error-correction over the reals. In *Proceedings of Randomization and Approximation Techniques in Computer Science Conference (RANDOM)*, 2008.

[Gru68] B. Grunbaum. Grassmann angles of convex polytopes. *Acta Mathematica*, 121:293–302, 1968.

[Gru03] B. Grunbaum. *Convex polytopes (volume 221 of Graduate Texts in Mathematics)*. Springer-Verlag, New York, second edition, 2003.

[GSTV06] A. C. Gilbert, M. J. Strauss, J. A. Tropp, and R. Vershynin. Algorithmic linear dimension reduction in the ℓ_1 norm for sparse vectors. Proceedings of the Allerton Conference, 2006.

[GSTV07] A. Gilbert, M. Strauss, J. Tropp, and R. Vershynin. One sketch for all: Fast algorithms for compressed sensing. In *ACM Symposium on Theory of Computing (STOC)*, 2007.

[HCS08] S. D. Howard, A. R. Calderbank, and S. J. Searle. A fast reconstruction algorithm for deterministic compressive sensing using second order Reed-Muller codes. In *Proceedings of Conference on Information Sciences & Systems*, 2008.

[HEA05] X. Y. Hu, E. Eleftheriou, and D. M. Arnold. Regular and irregular progressive edge-growth Tanner graphs. *IEEE Transactions on Information Theory*, 51(1):386–398, 2005.

[HKXA09] B. Hassibi, A. Khajehnejad, W. Xu, and S. Avestimehr. Breaking the ℓ_1 recovery thresholds with reweighted ℓ_1 optimization. In *Proceedings of Allerton Conference*, 2009.

[HS09] M. A. Herman and T. Strohmer. High-resolution radar via compressed sensing. *IEEE Transactions on Signal Processing*, 57(6):2275–2284, 2009.

[HSC09] S. Howard, S. Searle, and R. Calderbank. Chirp sensing codes: Deterministic compressed sensing measurements for fast recovery. *Applied and Computational Harmonic Analysis*, 2009.

[HV50] P. R. Halmos and H. E. Vaughan. The marriage problem. *American Journal of Mathematics*, 72(1):214–215, 1950.

[Ind08] P. Indyk. Explicit constructions for compressed sensing of sparse signals. In *Proceedings of ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2008.

[IR08] P. Indyk and M. Ruzic. Near-optimal sparse recovery in the l_1 norm'. In *Proceedings of 49th Symposium on Foundations of Computer Science (FOCS)*, 2008.

[JM] S. Jokar and V. Mehrmann. Sparse representation of solutions of Kronecker product of systems. arXiv:0902.4587.

[JS08] S. Jagabathula and D. Shah. Inferring popular rankings under constrained sensing. In *Neural Information Processing Systems Conference (NIPS)*, 2008.

[JXC08] S. Ji, Y. Xue, and L. Carin. Bayesian compressive sensing. *IEEE Transactions on Signal Processing*, 56(6):2346–2356, 2008.

[JXHC09] S. Jafarpour, W. Xu, B. Hassibi, and R. Calderbank. Efficient and robust compressed sensing using high-quality expander graphs. *IEEE Transactions on Information Theory*, 55(9):4299–4308, 2009.

[JYG09] X. Jiang, Y. Yao, and L. Guibas. Stable identification of cliques with restricted sensing. In *Neural Information Processing Systems Conference (NIPS)*, 2009.

[KKH11] A. Khajehnejad, A. Khojastepour, and B. Hassibi. Compressed network tomography for probabilistic tree mixture models. In *Globecomm Conference*, 2011.

[KNN09] Y. Kim, S. S. Narayanan, and K. S. Nayak. Accelerated three-dimensional upper airway MRI using compressed sensing. *Magnetic Resonance in Medicine*, 61:1434–1440, 2009.

[KOH12] A. K. Krishnaswamy, S. Oymak, and B. Hassibi. A simpler approach to weighted ℓ_1 minimization. In *International Conference on Acoustics, Speach and Signal Processing (ICASSP)*, 2012.

[Kru77] J.B. Kruskal. Three-way arrays: Rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. *Linear Algebra Applications*, 18, 1977.

[KTDH11] A. Khajehnejad, A. Saber Tehrani, A. G. Dimakis, and B. Hassibi. Explicit matrices for sparse approximation. In *IEEE International Symposium on Information Theory (ISIT)*, 2011.

[KTH12] A. Khajehnejad, M. Thill, and B. Hassibi. Projected ℓ_1 minimization for compressed sensing. In *International Conference on Acoustic, Speech and Signal Processing*, 2012.

[KV03] R. Koetter and P. O. Vontobel. Graph-covers and iterative decoding of finite length codes. In *Proceedings of 3rd International Symposium on Turbo Codes*, 2003.

[KV06] R. Koetter and P. O. Vontobel. On the block error probability of LP decoding of LDPC codes. In *Proceedings of the Inaugural Workshop of the Center for Information Theory and Applications, UC San Diego, La Jolla, CA, USA*, February 6-10, 2006.

[KWYH12] A. Khajehnejad, A. S. Avestimehr W. Yu, and B. Hassibi. Improving the thresholds of sparse recovery: An analysis of a rwo-step reweighted basis pursuit algorithm. *(Submitted to IEEE Transactions on Information Theory)*, 2012. arXiv:1111.1396.

[KXAH10] A. Khajehnejad, W. Xu, S. Avestimehr, and B. Hassibi. Improved sparse recovery thresholds with two step reweighted ℓ_1 minimization. In *IEEE International Symposium on Information Theory (ISIT)*, 2010.

[KXAH11] A. Khajehnejad, W. Xu, S. Avestimehr, and B. Hassibi. Analyzing weighted ℓ_1 minimization for sparse recovery with nonuniform sparse models. *IEEE Transactions on Signal Processing*, 59(5):1985–2001, 2011.

[KXDH10] A. Khajehnejad, W. Xu, A. G. Dimakis, and B. Hassibi. Sparse recovery of nonnegative signals with minimal expansion. *IEEE Transactions on Signal Processing*, 59(1):196–208, 2010.

[KZ] P. Koiran and A. Zouzias. On the certification of the restricted isometry property. arXiv:1103.4984.

[LDP07] M. Lustig, D. Donoho, and J. M. Paul. Sparse MRI: The application of compressed sensing for rapid MR imaging. *Magnetic Resonance in Medicine*, 58(6):1182–95, 2007.

[LDSP08] M. Lustig, D. Donoho, J.M. Santos, and J.M. Pauly. Compressed sensing MRI. *IEEE Signal Processing Magazine*, 25(2):72 – 82, 2008.

[LKD+07] J. Laska, S. Kirolos, M. Duarte, T. Ragheb, R. Baraniuk, and Y. Massoud. Theory and implementation of an analog-to-information converter using random demodulation. In *IEEE International Symposium on Circuits and Systems (ISCAS)*, 2007.

[LLAV11] W. Lu, T. Li, I. Atkinson, and N. Vaswani. Modified-CS-residual for recursive reconstruction of highly undersampled functional MRI sequences. In *IEEE International Conference on Image Processing (ICIP)*, 2011.

[LLP08] J. Luo, Y. Lu, and B. Prabhaka. Prototyping counter braids on NetFPGA. Technical report, 2008.

[LMP08] Y. Lu, A. Montanari, and B. Parabhakar. Counter braids: Asymptotic optimality of the message passing decoding algorithm. In *Proceedings of the Allerton Conference*, September 2008.

[LMSS98] M. Luby, M. Mitzenmacher, A. Shokrollahi, and D. Spielman. Analysis of low density codes and improved designs using irregular graphs. In *Proceedings of 30th Annual ACM Symposium on Theory of Computing*, 1998.

[LN06] N. Linial and I. Novik. How neighborly can a centrally symmetric polytope be? *Discrete and Computational Geometry*, pages 273–281, 2006.

[Log65] B. F. Logan. *Properties of high-pass signals*. PhD thesis, Columbia University, 1965.

[LT91] M. Ledoux and M. Talagrand. *Probability in Banach spaces*. Springer-Verlag, 1991.

[LV] W. Lu and N. Vaswani. Exact reconstruction conditions for regularized modified basis pursuit. arXiv:1108.3350.

[LV09] W. Lu and N. Vaswani. Modified compressive sensing for real-time dynamic MR imaging. In *IEEE International Conference image Processing (ICIP)*, 2009.

[LV10] W. Lu and N. Vaswani. Regularized modified-BPDN for compressive sensing with partially known support. (Preprint), 2010.

[Mar06] R. Marks. *Introduction to Shannon Sampling and Interpolation Theory.* Springer, 2006.

[Mas03] P. Massart. Concentration inequalities and model selection. *Lecture Notes in Mathematics École d'Été de Probabilités de Saint-Flour*, Springer, 2003.

[MBSR07] O. Milenkovic, R. Baraniuk, and T. Simunic-Rosing. Compressed sensing meets bionformatics: A new DNA microarray architecture. In *Information Theory and Applications Workshop, San Diego, CA, USA*, 2007.

[McM75] P. McMullen. Non-linear angle-sum relations for polyhedral cones and polytopes. *Mathematical Proceedings of the Cambridge Philosophical Society*, 78(2):247–261, 1975.

[ME10] M. Mishali and Y. C. Eldar. From theory to practice: Sub-Nyquist sampling of sparse wideband analog signals. *IEEE Journal of Selected Topics In Signal Processing*, 4(2):375–391, 2010.

[ME11] M. Mishali and Y. C. Eldar. Sub-Nyquist sampling: Bridging theory and practice. *IEEE Signal Processing Magazine*, 28(6):98–124, November 2011.

[MN99] D. J. MacKay and R. M. Neal. Good error-correcting codes based on very sparse matrices. *IEEE Transactions on Information Theory*, 45(2):399–431, 1999.

[MPJ09] S. Mendelson, A. Pajor, and N. T. Jaegermann. Uniform uncertainty principle for Bernoulli and subGaussian ensembles. *Constructive Approximation*, 28:277–289, 2009.

[MZ93] S. G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transacctions on Signal Processing*, 41(12):3397–3415, 1993.

[Nee09] D. Needell. Noisy signal recovery via iterative reweighted L1-minimization. In *Proceedings of the Asilomar Conference on Signals, Systems, and Computers*, 2009.

[NT08] D. Needell and J. A. Tropp. Cosamp: Iterative signal recovery from incomplete and inaccurate samples. *Applied and Computational Harmonic Analysis*, 26:301–321, 2008.

[NTV08] D. Needell, J. Tropp, and R. Vershynin. Greedy signal recovery review. In *4nd Asilomar Conference on Signals, Systems and Computers*, 2008.

[NV07] D. Needell and R. Vershynin. Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit. (Preprint), 2007.

[OH] S. Oymak and B. Hassibi. New null space results and recovery thresholds for matrix rank minimization. arXiv:1011.6326v1.

[OKH11] S. Oymak, A. Khajehnejad, and B. Hassibi. Weighted compressed sensing and rank minimization. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011.

[OKH12] S. Oymak, A. Khajehnejad, and B. Hassibi. On the phase transition of convex problems. Submitted to International Symposium on Informaiton Theory (ISIT), 2012.

[PH08] F. Parvaresh and B. Hassibi. Explicit measurements with almost optimal thresholds for compressed sensing. In *IEEE International Conference on*

Acoustics, Speech and Signal Processing (ICASSP), Las Vegas, NV, USA, April 2008.

[PNF04] H. Pishro-Nik and F. Fekri. On decoding of low-density parit-check codes over the erasure channel. *IEEE Transactions on Information Theory*, 50(3):439–454, 2004.

[PRK93] Y .C. Pati, R. Rezaiifar, and P. S. Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Twenty-Seventh Asilomar Conference on Signals, Systems and Computers*, 1993.

[PVMH08] F. Parvaresh, H. Vikalo, S. Misra, and B. Hassibi. Recovering sparse signals using sparse measurement matrices in compressed DNA microarrays. *IEEE Journal of Selected Topics In Signal Processing*, 2(3):275–285, June 2008.

[PX94] P. M. Padralos and J. Xue. The maximum clique problem. *Journal of Global Optimization*, 4:301–328, 1994.

[Rau] H. Rauhut. Circulant and toeplitz matrices in compressed sensing. arXiv:0902.4394.

[RFP10] B. Recht, M. Fazel, and P.A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, 52(3):471–501, 2010.

[ric] Compressed sensing online resources. http://dsp.rice.edu/cs.

[RU01] T. J. Richardson and R. Urbanke. The capacity of low-density paritycheck codes under message-passing decoding. *IEEE Transactions on Information Theory*, 47(2):599618, 2001.

[RXH08] B. Recht, W. Xu, and B. Hassibi. Necessary and sufficient conditions for success of the nuclear norm heuristic for rank minimization. *In IEEE Conference on Decision and Control (CDC)*, 2008.

[San52] L. A. Santaló. Geometría integral en espacios de curvatura constante. *Rep. Argetina Publ. Com. Nac. Energí Atómica*, 1, 1952.

[SBB06] D. Sarvotham, D. Baron, and R. Baraniuk. Sudocodes-Fast measurement and reconstruction of sparse signals. In *IEEE International Symposium on Information Theory (ISIT)*, 2006.

[Sha48] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 623–656, 1948.

[Sof] http://www.its.caltech.edu/∼amin/weighted_l1_codes/.

[SPH09] M. Stojnic, F. Parvaresh, and B. Hassibi. On the reconstruction of block-sparse signals with an optimal number of measurements. *In IEEE Transactions on Signal Processing*, 2009.

[SS96] M. Sipser and D. A. Spielman. Expander codes. *IEEE Transactions on Information Theory*, 42(6):1710–1722, 1996.

[sta] Satellite images reveal deforestation threatening endemic bird species. http://www.guardian.co.uk/environment/2008/jan/09 /endangeredspecies.endangeredhabitats.

[Sto] M. Stojnic. Various thresholds for ℓ_1-optimization in compressed sensing. arXiv:0907.3666v1.

[Sto09] M. Stojnic. A simple performance analysis of ℓ_1 optimization in compressed sensing. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2009.

[Sto10] M. Stojnic. Towards improving ℓ_1 optimization in compressed sensing. *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, 2010.

[SV09] R. Smarandache and P. O. Vontobel. Absdet-pseudo-codewords and perm-pseudo-codewords: definitions and properties. In *IEEE International Symposium on Information Theory (ISIT)*, 2009.

[SXH08] M. Stojnic, W. Xu, and B. Hassibi. Compressed sensing—probabilistic analysis of a null-space characterization. In *IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP)*, 2008.

[Sza97] S. J. Szarek. Metric entropy of homogeneous spaces, 1997. arXiv:math/9701213v1.

[TBM79] H. L. Taylor, S. C. Banks, and J. F. McCoy. Deconvolution with the $\ell 1$ norm. *Geophysics*, 44(1):3952, 1979.

[TG07] J. Tropp and A. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Transactions on Information Theory*, 53(12):4655–4666, 2007.

[TGS06] J. A. Tropp, A. C. Gilbert, and M. J. Strauss. Algorithms for simultaneous sparse approximation. Part I: Greedy pursuit. *Signal Processing, Special Issue: Sparse Approximations in Signal and Image Processing*, 86:572–588, 2006.

[TH08] G. Taubock and F. Hlawatsch. A compressed sensing technique for ofdm channel estimation in mobile environments: Exploiting channel sparsity for reducing pilots. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2008.

[Tib96] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58(1):267–288, 1996.

[Tib97] R. Tibshirani. The LASSO method for variable selection in the Cox model. *Statistics in Medicine*, 16:385–395, 1997.

[Tro04] J. A. Tropp. Greed is good: Algorithmic results for sparse approximation. *IEEE Transactions on Information Theory*, 50(10):2231–2242, 2004.

[TS06] M. H. Taghavi and P. H. Siegel. Adaptive linear programming decoding. In *IEEE International Symposium on Information Theory (ISIT)*, 2006.

[VK06] P. O. Vontobel and R. Koetter. Towards low-complexity linear-programming decoding. In *Proceedings of International Conference on Turbo Codes and Related Topics*, 2006.

[VL10] N. Vaswani and W. Lu. Modified-CS: Modifying compressive sensing for problems with partially known support. *In IEEE Transactions on Signal Processing*, 58(9):4595–4607, 2010.

[VPMH07] H. Vikalo, F. Parvaresh, S. Misra, and B. Hassibi. Recovering sparse signals using sparse measurement matrices in compressed DNA microarrays. In *Asilomor Conference*, November 2007.

[VRR11] A. Veeraraghavan, D. Reddy, and R. Raskar. Coded strobing photography: Compressive sensing of high speed periodic videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(4):671–686, 2011.

[WJW02] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. MAP estimation via agreement on (hyper)trees: Message-passing and linear programming approaches. In *Proceedings of Allerton Conference on Communication, Control and Computing*, 2002.

[WJW05] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. Exact MAP estimates via agreement on (hyper)trees: Linear programming and message-passing. *IEEE Transactions on Information Theory*, 51(11):3697–3717, 2005.

[Wol83] J. Wolf. Redundancy, the discrete Fourier transform and impulse noise cancellation. *IEEE Transactions on Communications*, 31(3), 1983.

[WWR10] W. Wang, M. J. Wainwright, and K. Ramchandran. Information-theoretic limits on sparse signal recovery: Dense versus sparse measurement matrices. *IEEE Transactions on Information Theory*, 56(6):2967–2979, 2010.

[Wym07] H. Wymeersch. *Iterative Receiver Design*. Cambridge University Press, 2007.

[XH07a] W. Xu and B. Hassibi. Further results on performance analysis for compressive sensing using expander graphs. In *Proceedings of the Forty-First Asilomar Conference on Signals, Systems and Computers*, 2007.

[XH07b] W. Xu and B. Hassibi. Efficient compressive sensing with determinstic guarantees using expander graphs. *In Proceedings of the IEEE Information Theory Workshop, Lake Tahoe*, September 2007.

[XH08] W. Xu and B. Hassibi. Compressed sensing over the Grassmann manifold: A unified analytical framework. In *46th Annual Allerton Conference on Communication, Control, and Computing*, 2008.

[XH09] W. Xu and B. Hassibi. On sharp performance bounds for robust sparse signal recoveries. In *International Symposium on Information Theory (ISIT)*, 2009.

[XMT11] W. Xu, E. Mallada, and A. K. Tang. Compressive sensing over graphs. In *IEEE International Conference on Computer Communications (INFOCOM)*, 2011.

[XT10a] W. Xu and A. Tang. On the uniqueness of positive semidefinite matrix solution under compressed observations. In *International Symposium on Information Theory (ISIT)*, 2010.

[XT10b] W. Xu and A. K. Tang. On the dynamics of ℓ_1 decoding: A microscopic approach. In *International Symposium on Information Theory (ISIT)*, 2010.

[YFW06] K. Yang, J. Feldman, and X. Wang. Nonlinear programming approaches to decoding low-density parity-check codes. *IEEE Journal of Selected Areas in Communication*, 24(8):1603–1613, 2006.

[Zha05] Y. Zhang. When is missing data recoverable? Technical report, Rice CAAM Department (available online at http://www.caam.rice.edu/ỹzhang/reports/tr0615.pdf), 2005.

[ZLG] L. Zhang, J. Luo, and D. Guo. Compressed neighbor discovery for wireless networks. arXiv:1012.1007.

[ZP] F. Zhang and H. D. Pfister. On the iterative decoding of high rate LDPC codes with applications in compressed sensing. arXiv:0903.2232.