PHASE TRANSITION USING DETERMINISTIC SENSING MATRIX IN COMPRESSED SENSING

By

Aisha Zulfiqar

A thesis submitted to the faculty of Electrical Engineering Department,
Military College of Signals, National University of Sciences and Technology,
Islamabad, Pakistan, in partial fulfillment of the requirements for the degree of MS in
Electrical(Telecommunication)Engineering Depatment

January 2016

ABSTRACT

Compressed sensing theory is based on the novel idea that, a specific class of signals i.e sparse signals can be recovered by sampling below the traditional sampling rate, as used in communication systems theory.

Thus the important challenge in compressed sensing is to reconstruct an undersampled sparse signal. The amount of undersampling possible for a signal having a particular sparsity needs to be known. Phase transition of a particular algorithm represents the sparsity vs undersampling ratios in the form of a phase diagram. Therefore, the limit for undersampling is revealed in the phase transition.

Phase transition improvement is required in order to be able to recover original sparse signal by even less number of samples. Currently, new recovery algorithms that provide higher phase transition curves are being developed.

In this thesis, we use the approach of utilizing deterministic sensing matrices in place of random sensing matrices for reconstruction. Random sensing matrices have already been employed to discover the limit on undersampling. The deterministic sensing matrices used in this thesis belong to a class of sensing matrices that have lower coherence. The deterministic sensing matrices utilized in this thesis were obtained using best antipodal spherical codes. Experiments have been performed on few recovery algorithms in order to check the impact of this particular type of deterministic matrices. The impact of noise is also visualized on the phase transition of these algorithms.

From the results of the experiments it is concluded that BP and AMP algorithms have almost same phase transitions for both random and deterministic sensing matrices. However, OMP shows a marked improvement by employing deterministic sensing matrix. Thus suitable sensing matrix may optimize phase transition of existing algorithms.

DEDICATION

This thesis is dedicated to

MY FAMILY, FRIENDS AND TEACHERS

for their love, endless support and encouragement.

ACKNOWLEDGEMENTS

I am grateful to Allah Almighty who has blessed me with the opportunity to seek knowledge and bestowed upon me the strength to accomplish this thesis and I am thankful to Him for without His mercy and benevolence I could not have completed this task.

I am also thankful to my thesis Supervisor Col Dr. Imran Rasheed whose cooperation and guidance was quite helpful in accomplishing this task. I express my sincere gratitude to my Co-Supervisor Lt Col Faisal Akram, without his encouragement, support and guidance the successful completion of this thesis would not have been possible.

The loving support of my family especially my parents and friends has been valuable, their prayers and wishes enabled me to complete my Master Degree in Electrical Engineering and especially the thesis.

TABLE OF CONTENTS

LIST OF FIGURES

NOTATION

$\mathbf{A}$ Sensing matrix.

$\mathbf{y}$ Measurement matrix.

$\mathbf{x}$ Sparse signal.

$\mathcal{D}$ Distance distribution.

$\mu(\mathbf{A})$ Coherence of $\mathbf{A}$.

$\mathcal{N}(\mathbf{A})$ Null space of matrix $\mathbf{A}$.

$g(\mathcal{D})$ Generalized potential energy function.

m Number of rows.

N Number of columns.

$\mathcal{E}$ Error threshold.

det Determinant of matrix.

k Sparsity.

δ Undersamplig fraction.

ρ Sparsity fraction.

$C_s(N, M)$ Spherical code matrix.

$C_{abs}(N, M)$ Best Antipodal Spherical code matrix.

$\mathbb{R}$ Set of real numbers.

δ_k Restricted isometry constant.

v Exponenial factor used in BASC algorithm convergence.

$||\mathbf{A}||_2$ Spectral norm of matrix $\mathbf{A}$.

ACRONYMS

Compressed sensing	CS
Null Space property	NSP
Restricted Isometry Property	RIP
Basis Pursuit	BP
Orthogonal Matching Pursuit OMP	OMP
Iterative Hard Thresholding	IHT
Iterative Soft Thresholding	IST
Approximate Message Passing	AMP
Phase Transition	PT
Mean Square Error	MSE
Best Spherica Codes	BSC
Best Antipodal Spherical Code	BASC
Distance	dist
Independent and identically distributed	i.i.d

Chapter 1

INTRODUCTION

In the modern digital world, the transmission and reception of data at a higher rate is the main task necessary for efficient communication. For this purpose data needs to be sampled according to some criterion. The Shannon/Nyquist sampling theorem has been used to sample analog signals. According to Shannon the signal must be sampled at a rate twice the highest frequency of the signal. However, after sampling signals at a rate that high, a large number of samples are obtained that need to be compressed for the purposes of transmission and storage. Also in image, audio and video processing the Shannon/Nyquist theorem is not optimal for sampling.

Compressed sensing is a novel theory in signal processing introduced about a decade ago [1] [2] . It has the ability to sample particular classes of signals at a rate below the Shannon/Nyquist rate. Compressed sensing exploits sparse signals (that have few non-zero entries in the signal vector) for efficient sampling as compared to Shannon/Nyquist sampling. Compressed sensing is important as it reduces the number of measurements required to accurately reconstruct a signal, these measurements are obtained by multiplying the sparse signal $\mathbf{x}$ with a sensing matrix $\mathbf{A}$ to obtain $\mathbf{y}$ which is the measurement vector. The equation obtained by the operation mentioned produces an underdetermined system of linear equations. Although, underdetermined system of linear equations have infinitely many solutions but sparsity(the number of non-zero values in a signal) provides a method to give a unique sparsest solution.

Compressed sensing is relatively a new field yet it has many applications. Like in MRI to take images of organs and tissues the patient has to stay still in the machine for several minutes to acquire samples, compressed sensing can help reduce the time by reducing the number of samples [3]. In imaging, it is important as it can reduce the cost of cameras, who have expensive and large number of sensors, by reducing the number of sensors required. Seismic data collection, is a time consuming as well as expensive process, compressed sensing has helped researchers to make it a faster and cheaper process. Other areas

of application are radars, analog to digital converter designs, astronomy, machine learning, ultrasound imaging system, computational biology, etc.

1.1 Problem Statement

Phase transition is an important parameter in compressed sensing theory. It is the sparsity vs undersampling tradeoff i.e it provides information on how much undersampling is possible for a particular sparsity level. Thus the phase transition needs to be improved. The possible ways to improve it is by designing new efficient algorithms or better sensing matrices. Then observe the effect of noise on phase transition.

1.2 Objectives

The improvement of phase transition is the task to be accomplished in this thesis. The possible ways to improve it is by designing new efficient algorithms or better sensing matrices. The approach we have used in this thesis is by using a better sensing matrix that is deterministic in nature to improve the phase transition.

1.3 Methodology

This thesis works to provide an improvement in the phase transition by using deterministic sensing matrices. We take few algorithms in existing literature whose phase transitions have previously been defined by using random Gaussian sensing matrices. We use a particular deterministic sensing matrix to see its effect on phase transition, for each algorithm separately. It is checked that does deterministic matrix optimizes the results or not. Then to view the effect of noise on phase transition curve, a noisy environment is introduced and for each algorithm the effect is analyzed.

1.4 Thesis Outline

This thesis is organized such that the after introduction, Chapter 2 includes the basic knowledge of CS to establish a foundation for the concepts discussed in later chapters. Chapter 3 undertakes the theory of signal recovery in CS and some algorithms for signal recovery being used in CS. In Chapter 4 the concept of phase transition which is an important parameter in CS that needs to be improved has been introduced and some prior work in this direction has been discussed. Chapter 5 highlights the importance of deterministic matrices and how they are better than random sensing matrices. The theory for low coherence sensing matrices is

established and the best antipodal spherical codes discussed in detail which produce sensing matrix of low coherence. The results of our work on phase transition using deterministic sensing matrix are displayed in Chapter 6 for both noiseless and noisy case.

For the ease of understanding the mathematical notations used throughout the thesis are mentioned here:

1.5 Mathematical Notation

- Standard letters are used for representing scalars: e.g, n $\in \mathbb{R}$
- Bold letters in the lower case are used to represent column vectors: e.g, $\mathbf{x} \in \mathbb{R}^n$
- Bold capital letters are used to represent matrices: e.g, $\mathbf{A} \in \mathbb{R}^{m \times n}$
- Calligraphic letters are used to show vector spaces formed by sets of vectors: e.g, $\mathcal{N}$

1.6 Software Tools

For simulations, MATLAB® (R2012a) is used.

Chapter 2

COMPRESSED SENSING AND ITS PRINCIPLE

The research area of compressed sensing was first introduced in 2006 by the two ground-breaking research papers of Donoho and of Candes, Romberg, and Tao [1] [2]. The main idea of compressed sensing is the exact recovery of a sparse signal using only few linear and non-adaptive measurements by convex optimization [4] . In other words, compressed sensing provides with a unique sparse solution to the under-determined system of linear equations. Compressed sensing approach differs from classical sampling in that it acquires measurements by taking inner product of signal and more general test functions. Typically measurements are taken by the linear system obtained as a result of the product of a sparse input signal vector and coefficient matrix called sensing matrix in CS. The basic equation of compressed sensing is as follows:

$$\mathbf{y}_{m\times 1} = \mathbf{A}_{m\times n}\mathbf{x}_{n\times 1}, \tag{2.1}$$

whereas in the equation above, the unknown signal to be retrieved is **x**, **A** is the sensing matrix, and both of these produce **y** the measurement vector.

2.1 Retrieving sparse vector

Firstly, we review the basic underlying concepts like norms, basis and frames. Then discuss the recovery of spare vector.

2.1.1 Norm

In present day signal processing, signals are modeled using vectors living in an appropriate vector space. In an n-dimensional Euclidean space denoted by $\mathbb{R}^n$ the l_p norm is defined as:

$$||\mathbf{x}||_p = \begin{cases} (\sum_{i=1}^{n} |x_i|^p)^{\frac{1}{p}} & if\ p \in [1,\infty], \\ \max\limits_{i=1,2....,n} |x_i| & if\ p = \infty, \end{cases} \tag{2.2}$$

where x_i are the components of the vector **x**. Norms are typically used as a measure of the strength of a signal, or the size of an error. The most commonly encountered vector norm is

the ℓ_2 norm which is the inner product, it may also be given by:

$$||\mathbf{x}||_2 = \sqrt{x_1^2 + x_2^2 + ... + x_n^2}. \tag{2.3}$$

The ℓ_1 norm also called the taxicab norm is basically the sum of the absolute values of columns of the vector **x** given by:

$$||\mathbf{x}||_1 = \sum_{i=1}^{n} |x_i|. \tag{2.4}$$

The ℓ_0 norm is the cardinality of the non-zero content of the vector, the support(supp(**x**)) of vector is the same thing so ℓ_0 norm is given by:

$$||\mathbf{x}||_0 = |supp(x)|. \tag{2.5}$$

When $0 < p < 1$ then the triangular inequality is not satisfied, then the ℓ_p norm is called quasi or pseudo norm. Figure 2.1 shows the unit spheres in $\mathbb{R}^2$ for ℓ_1, ℓ_2 and ℓ_∞ norms as well as $\ell_{\frac{1}{2}}$ quasi norm.

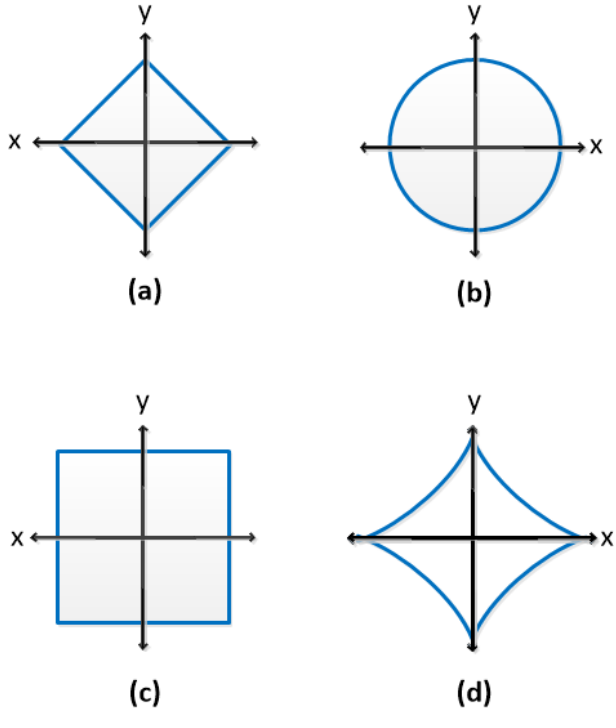


Figure 2.1: Unit spheres for ℓ_p norms.(a) ℓ_1 norm (b) ℓ_2 norm (c) ℓ_∞ norm (d) $\ell_{\frac{1}{2}}$

2.1.2 Sparsity

Sparsity of a vector is the number of non-zero components contained in the vector.

2.1.3 Basis

A set of vectors $\{\mathbf{a}_i\}_{i=1}^n$ in a vector space $\mathbb{R}^n$ is called basis if the vectors are linearly independent and all the vectors in that vector space form a linear combination of the the vector set. Then for each $\mathbf{x} \in \mathbb{R}^n$ there are unique coefficients $\{c_i\}_{i=1}^n$ such that:

$$\mathbf{X} = \sum_{i=0}^{n} c_i \mathbf{a}_i. \tag{2.6}$$

A vector space $\mathbb{R}^n$ can have many different bases, but there are always the same number of basis vectors in each of them. The dimension of $\mathbb{R}^n$ is the cardinality of its basis vectors. A special case of basis vectors is orthonormal basis. The set of vectors $\{\mathbf{a}_i\}_{i=1}^n$ are orthonormal if they satisfy

$$\langle \mathbf{a}_i, \mathbf{a}_j \rangle = \begin{cases} 1 & if\ i = j, \\ 0 & if\ i \neq j, \end{cases} \tag{2.7}$$

2.1.4 Frames

Frames are a generalization of the concept of vector basis, by taking into account sets of vectors that are linearly dependent. The frame is defined as a set of vectors $\{\mathbf{a}_i\}_{i=1}^n$ in $\mathbb{R}^k$ where $k < n$ corresponding to a matrix $\mathbf{P} \in \mathbb{R}^{d \times n}$, such that for all vectors $\mathbf{x} \in \mathbb{R}^d$

$$A||\mathbf{x}||_2^2 \leq ||\mathbf{P}||_2^2 \leq B||\mathbf{x}||_2^2, \tag{2.8}$$

with $0 < A \leq B < \infty$ and A implies that rows of $\mathbf{P}$ are linearly independent. A frame is called tight frame if A=B and in the case when A=B=1 the frame is called a Parseval frame. Every orthonormal basis is a Parseval frame but all Parseval frames do not form orthogonal basis. An equal norm frame is the one if there is a constant v such that $c > 0$ and $||\mathbf{a}_i||_2 = c$ for all i=1,...,n and is unit norm if c=1.

2.2 Sparse Signal model

In compressed sensing, sparsity is the prior information assumed of the vector, we intend to efficiently sense or whose dimension we intend to reduce. A signal $\mathbf{x}$ is said to be sparse if it has k non-zero entries at most, which is measured by the ℓ_0 norm of the signal i.e. if $||\mathbf{x}||_0 \leq k$ then mathematically let

$$\sum_k = \{\mathbf{x} : ||\mathbf{x}||_0 \leq k\}, \tag{2.9}$$

denote a set of all the k-sparse signals. In case we have signals that are not sparse themselves, then we check if they admit a sparse representation in some basis $\mathbf{\Phi}$ The signal **x** can still be referred as k-sparse while we can express **x** as $\mathbf{x} = \mathbf{\Phi}c$ where $||c||_0 \leq k$.

2.3 Compressed sensing technique

Compressed sensing theory rests of the recovery of signals by using under-determined sensing matrix. The basic equation of CS theory is

$$\mathbf{y}_{m\times 1} = \mathbf{A}_{m\times n}\mathbf{x}_{n\times 1}, \tag{2.10}$$

whereas in the equation above, the unknown signal to be retrieved is **x**, **A** is the sensing matrix, and both of these produce **y** the measurement vector. Generally, sampling may be termed as a linear operation performed on the signal which is basically the same as identity. Now consider a k sparse signal **x**, and multiply it with identity matrix it gives **y**(samples). Each entry of **y** can be viewed as the inner product of the identity matrix and the signal **x**. If the information in **x** is preserved by **y** since **y=x**(no information loss), therefore the signal **x** can be recovered.

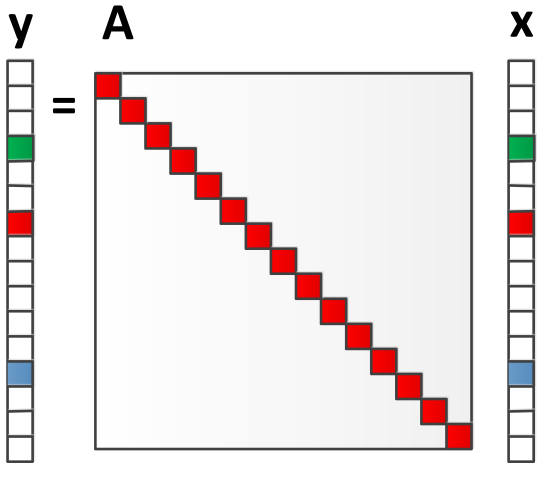


Figure 2.2: **N**o information loss when y=Ix.

However, in CS instead of multiplying with an identity matrix, an under-determined matrix $(m \ll n)$ is used for dimensionality reduction, taking m measurements. As the under-determined matrix is not a full rank matrix so information cannot be preserved and there are infinitely many solution for the same **x**. This can be visualized in fig 2.3

To obtain a unique solution the signal **x** must be k sparse. Therefore assuming **x** is sparse

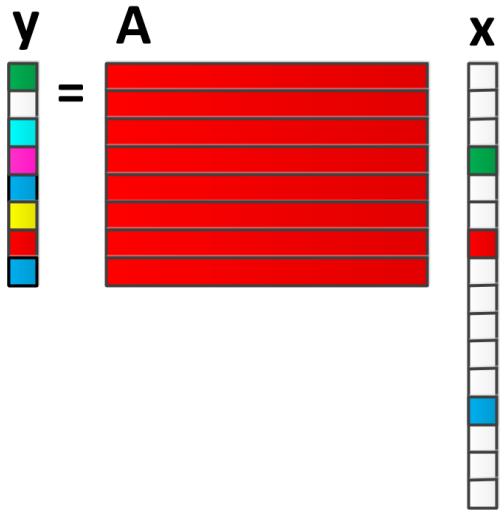


Figure 2.3: **W**hen **y=Ax** and **A**$(m \ll N)$ gives infinite solutions.

and since **y** is the linear combination of k columns of the matrix. So in this case the 3 columns corresponds to the 3 non-zero entries in the vector **x**.

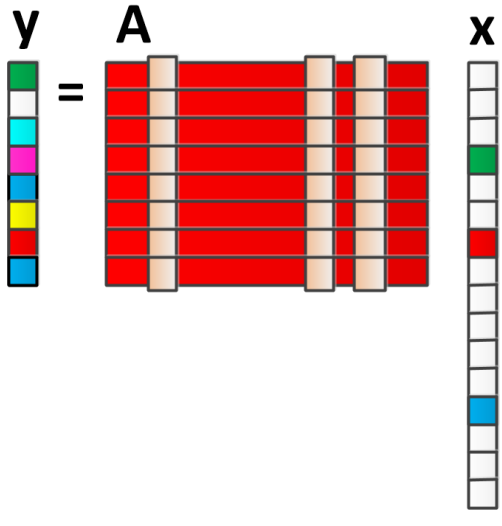


Figure 2.4: **T**aking k columns at a time from **A**.

Compressed sensing can be further extended to the case of non-sparse signals as well in addition to sparse signal vectors. This is achieved by converting the non-sparse signals to a representation that is sparse, by using sets of signals called dictionaries. The scope of compressed sensing has been extended by the possibility of representing signals in other basis. This way the CS theory is applicable to non-sparse signals in temporal and spectral domain. The dictionary matrix $\mathbf{\Psi}$ is constituted of a set of waveforms or components $(\mathbf{\Psi}_i)$ that may be interpreted as columns [7]. The matrix of Discrete Fourier Transform can be considered

as an example of dictionary containing sinusoidal waveforms. A signal that is consistent with dictionary $\mathbf{\Psi}$ can be represented by a vector $\boldsymbol{\alpha}$, which contains the coefficients for superposition of components $(\mathbf{\Psi}_i)$. As stated below

$$\mathbf{x}_{n\times 1} = \mathbf{\Psi}_{n\times l}.\boldsymbol{\alpha}_{n\times l}. \tag{2.11}$$

The representation is exact if the signal is completely consistent with the dictionary, otherwise it gives an approximation. To apply CS theory to a signal it is important to find an appropriate dictionary or domain in which the signal is sparse, for example a signal may be sparse in spectral domain but not in temporal. Measurement vector **y** is obtained from a signal **x** that is sparsely representable as mentioned below:

$$\mathbf{y}_{m\times 1} = \mathbf{\Phi}_{m\times n}.\mathbf{x}_{n\times 1}. \tag{2.12}$$

Combining the two operations performed in equ 2.11 and 2.12 into one equation we get:

$$\mathbf{y} = \mathbf{\Phi}.\mathbf{\Psi}.\boldsymbol{\alpha}. \tag{2.13}$$

Equation 2.13 represents the technique of CS theory such that it can be applied to the non-sparse signals as well.

2.4 Existence of Unique Minimal Sparse Solutions

Generally an under-determined system of linear equations has infinite solutions because the number of variables is more than the total number of equations. A unique solution may be obtained by considering geometric analysis. To get a unique sparsest solution with a high probability, it is required to know the sufficient condition on the sparsity of a signal vector. For a linear system to obtain a solution there are three possibilities it may have no solution at all, or a unique solution else many solutions.

2.4.1 Definitions and Representations

Some basic concepts and definitions are stated before going deep in to these possibilities of solutions.

Definition 2.1. Linear Independence of Vectors: For a set of vectors **A= a**$_1, ..., \mathbf{a}_m$ where $\mathbf{a} \in \mathbb{R}^n$ and a set of scalars $w_1, ., w_m$ where w $\in \mathbb{R}^n$, the vectors are linearly independent if the solution of the equation $w_1\mathbf{a}_1 + ... + w_m\mathbf{a}_m = 0$, provided the set of scalars w_j are all

zero [20].

If the above equation is satisfied for some other set of scalars that are not all zero then the vectors are linearly dependent because then some vector may be represented by a linear combination of the other.

Definition 2.2. Rank of a Matrix (A): The rank of a matrix is the maximum number of linearly independent column or row vectors [20].

Definition 2.3. Span/Vector Space: For a set **B** of vectors $\mathbf{a}_1, ..., \mathbf{a}_m$, such that $\mathbf{a} \in \mathbb{R}^n$, a set of all linear combinations of these vectors is called the span/vector space of m vectors [20].

2.4.2 Existence and Number of Solutions

While determining the solution of a set of linear equations, the concept of linear independence and rank of matrix plays an important role to discover the properties of the solution. The properties of the solution include the existence of solution and its uniqueness. Now we discuss the various possible types of solution for a linear system of equations with m equations and n variables. The matrix **A** and its augmented matrix $\mathbf{A}^c$ for the system of linear equations is given as

$$\mathbf{A} = \begin{bmatrix} a_{11} & . & . & a_{1n} \\ . & . & . & . \\ . & . & . & . \\ . & . & . & . \\ a_{m1} & . & . & a_{mn} \end{bmatrix} \qquad \mathbf{A}^c = \begin{bmatrix} a_{11} & . & . & a_{1n} & y_1 \\ . & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & . \\ a_{m1} & . & . & a_{mn} & y_m \end{bmatrix}.$$

Existence of Solutions: The solution for the above equations exists if and only if the ranks of the two matrices **A** and $\mathbf{A}^c$ is equal i.e. rank(**A**)= rank($\mathbf{A}^c$).

Unique solution: If the rank of the matrix **A** is equal to the number of columns in matrix **A** then a unique solution exists it can alternatively be stated as rank(**A**)= rank($\mathbf{A}^c$)=n.

Multiple solutions: The system has infinitely many solutions when the rank r is less than the number of columns of the matrix. These infinite solutions can then be obtained by the r coefficients.

The system of linear equations discussed above can be represented in another concise form i.e. in the form of matrix vector product

$$\mathbf{Ax} = \mathbf{y}. \tag{2.14}$$

The above relation may be manifested in another form where the components of **x** here may be interpreted as weights of column vectors of **A**. This other representation is given as follows

$$a_1\mathbf{x}_1 + a_2\mathbf{x}_2 + ... + a_n\mathbf{x}_n = \mathbf{y}, \tag{2.15}$$

where x_i while i=1,.,n are the components of **x**. Considering the possibilities of rank the rank of $\mathbf{A}^c$ can either be equal to the rank of **A** or can be equal to the rank(**A**)+1. Assume the solution of equation 2.15 to be χ then **y** is supposed to be linearly dependent on χ in this situation the rank(**A**) = rank($\mathbf{A}^c$)

2.4.3 Minimal sparse unique solution

In order to determine a unique sparse solution we need to define a limit for the number of non-zero elements present in the sparse vector so that the probability of the recovery remains high. A weak condition for the limit of sparsity is provided by spark, defined as follows:

Definition 2.4. Spark: The spark of a matrix **A** is the smallest number of columns that are linearly dependent [4].

The limit for sparsity defined in terms of spark is given by the following theorem:

Theorem 2.1: For any $\mathbf{y} \in \mathbb{R}^m$ there exists at most one k-sparse **x** such that **Ax = y** if and only if $spark(\mathbf{A}) < 2k$.

In light of the definition of spark and theorem 2.1 the limit on sparsity is $k > \frac{spark(\mathbf{A})}{2}$. The other possible limit that may be imposed on sparsity is interpreted by geometry and is $k < m - 1$. However, the limit imposed on sparsity by spark is not sufficient and therefore for a further insight into this limit brute force technique is considered.

Brute force technique is used to verify the uniqueness of the solution for a sparse signal. For this purpose consider an under-determined system of linear equations **Ax=y** where **A** $\in \mathbb{R}^{m\times n}$ and it is a full rank matrix i.e. rank(**A**) = m. For a k-sparse vector **x** when multiplied by this matrix **A** i.e. **Ax=y** the product is **y** which is the result of the weighting of the k non-zero elements of **x** against the k columns of A. This concludes that **y** is linearly dependent on **A** and thus lies in a sub-space made up of k linearly independent vectors from **A**. Brute force technique determines a sub space that is unique for **y** and works by taking **A** and **y** as inputs. For brute force technique to work determinants of all possible sub matrices, obtained by concatenating **y** with **A** need to be obtained. The algorithm then provides details about

the solution. The solution is considered unique if out of all the determinants calculated of the possible sub matrices, one determinant produces a zero.

In case there are more than one determinant that gives a zero then there exist multiple solutions. There may be two cases when multiple solutions emerge. One reason being the existence of linear dependence between the columns of the matrix **A**. Other case when multiple solutions are exhibited even though the matrix **A** has linearly independent columns. So to avoid multiple solutions the brute force approach needs to be improved in order to get unique solution. To overcome the problem initially it is required to distinguish between the two scenarios, which is done by taking different sets of matrices for each case. To form a new modified approach the determinant taking method is discarded (as determinants can only be taken for square matrices) due to the non-square sub-matrices due to sparsity level below $\mathrm{m}-1$. The new modified approach takes help of rank to get a unique solution. If different sub-matrices produce rank same as the sparsity, the same method is repeated by keeping sparsity at one level lower than the previous. Then the algorithm calculates rank of the concatenated sub-matrices of $\mathrm{m}-2$ columns of **A** with **y**.

2.5 Sensing matrix and necessary conditions on sensing matrices

An important problem in CS is that the necessary information needs to be preserved in the signal **x** when multiplied by the sensing matrix **A** to get the measurement vector **y**. To handle this problem an appropriate sensing matrix must be designed. The sensing matrix that has dimensions $m \times n$ where $m \ll n$ basically performs dimensionality reduction that is it maps $\mathbb{R}^n$ to $\mathbb{R}^m$. To use an appropriate sensing matrix there are some conditions in CS that must be fulfilled. These conditions are as follows:

2.5.1 Null space condition

The sensing matrix **A** required should be such that all the sparse vectors when multiplied to it give distinct measurement vectors, so that in the recovery process the right sparse vector is retrieved. Mathematically let **x** and **x'** be two distinct sparse vectors such that **x**,**x'** $\in \sum_k$ then it must be **Ax** $\neq$ **Ax'** so that **x** and **x'** can be reconstructed from their respective measurement vectors.

Null space of **A** is the set of signal vectors **x** that follow the condition **Ax** $= 0$ i.e when

product of these vectors is taken with the sensing matrix **A** they produce zero/null vector [4].

$$\mathcal{N}(A) = \{\mathbf{x} : \mathbf{Ax} = 0\}. \tag{2.16}$$

Consider if **Ax = Ax'** then **A**($\mathbf{x} - \mathbf{x}'$)=0 while **x - x'** $\in \sum_{2k}$ then **A** uniquely represents all **x** $\in \sum_k$ if and only if $\mathcal{N}(\mathbf{A})$ does not have any vectors in $\sum_{2k}$.

Now to define the Null Space Property (NSP) let a set of indices be $\mathbf{Y} \subset \{1, 2...n\}$ and $\mathbf{Y}_c$ = $\frac{\{1,2...n\}}{\mathbf{Y}}$. This means that the zeros in a vector $\mathbf{x_Y}$ will be in accordance with the set $\mathbf{Y}_c$.

Definition 2.5: Null space property (NSP) of order K exists for a matrix **A** if there is a constant $C > 0$ such that, [4]

$$||\mathbf{h_Y}||_2 \leq C\frac{||\mathbf{h}_{\mathbf{Y}_c}||_1}{\sqrt{K}}, \tag{2.17}$$

holds for all $\mathbf{h} \in \mathcal{N}(\mathbf{A})$ and for all **Y** such that $|\mathbf{Y}| \leq K$. The NSP then means that the null space vectors of a sensing matrix **A** should not be too concentrated on small subset of indices. NSP provides a good foundation for sparse recovery but does not provide sufficient guarantees in the presence of noise. NSP is a weak condition on the sensing matrix so more conditions are required which are discussed further.

2.5.2 Restricted Isometry Property

Candes and tao proposed the condition of restricted isometry on the sensing matrix **A** to provide immunity from noise and errors [19].

Definition 2.6: A matrix **A** satisfies the restricted isometry property (RIP) of order K if an element $\delta_k \in \{0, 1\}$ exists such that [4]

$$(1 - \delta_k)||\mathbf{x}||_2^2 \leq ||\mathbf{Ax}||_2^2 \leq (1 + \delta_k)||\mathbf{x}||_2^2, \tag{2.18}$$

where **x** is the k-sparse signal. If the sensing matrix **A** satisfies the RIP of order 2k, then it implies from equ 2.18 that **A** approximately preserves the distance between any pair of k-sparse vectors. The equation 2.18 shows RIP to be symmetric around 1, following is a more generalized version that uses arbitrary bounds

$$\alpha||\mathbf{x}||_2^2 \leq ||\mathbf{Ax}||_2^2 \leq \beta||\mathbf{x}||_2^2, \tag{2.19}$$

where $0 < \alpha < \beta < \infty$. Equation 2.19 is consistent with equ 2.18 after multiplying **A** with $\sqrt{\frac{2}{(\alpha+\beta)}}$ for scaling and δ= $\frac{\beta-\alpha}{\beta+\alpha}$.

2.5.3 Coherence

While NSP and RIP provide sufficient guarantee for CS recovery but to verify them for a matrix **A** the computational complexity is combinatorial so it is desirable to use some property that has lower complexity. Coherence is the desired property in that case.

Definition 2.7: Coherence of a sensing matrix **A** denoted by $\mu(\mathbf{A})$ may be defined as the largest absolute inner product of any two columns of **A** and mathematically given by:

$$\mu(\mathbf{A}) = max_{1 \leq i \leq j \leq n} \frac{|\langle \mathbf{a}_i, \mathbf{a}_j \rangle|}{||\mathbf{a}_i||_2 ||\mathbf{a}_j||_2}. \tag{2.20}$$

The range of coherence is $\mu(\mathbf{A}) \in \sqrt{\frac{n-1}{m(n-1)}}, 1$ and the lower bound is the Welch bound. The Welch bound i.e. the lower bound is $\mu(\mathbf{A}) \geq \frac{1}{\sqrt{m}}$ in the case when $n \gg m$ [4].

2.5.4 Relation between RIP and coherence

As discussed above the Restricted Isometry Property and Coherence conditions on the sensing matrix are taken as two separate conditions with different levels of simplicity and applicability. However a lemma exists that explains the relation between the two independent conditions.

Lemma 2.1: If the coherence of matrix **A** has the upper bound μ and the columns of the sensing matrix **A** are normalized, then for any $k < \mu^{-1} + 1$, **A** satisfies RIP$(k, (\text{k-1})\mu)$.
The above mentioned lemma is an application of Gresgorins theorem [37].

When the sensing matrices under observation are random in nature then the condition of coherence provides weaker results as compared to the RIP. The condition of coherence is considered advantageous because of the ease of calculation it provides for a particular matrix, it provides an edge when considering deterministic matrices, whereas for deterministic setting calculating RIP constant is a Non Polynomial time problem.

Chapter 3

RECOVERY ALGORITHMS FOR COMPRESSED SENSING

The compressed sensing theory is based on the exact recovery of a sparse signal by only few linear non-adaptive measurements by convex optimization [4]. In other words, compressed sensing provides with a unique sparse solution to the under-determined system of linear equations. Compressed sensing approach differs from classical sampling in that it acquires measurements by taking inner products of signal and more general test functions. Typically measurements are taken by the linear system obtained as a result of the product of a sparse input signal vector and coefficient matrix called sensing matrix in CS.

$$\mathbf{y}_{m\times 1} = \mathbf{A}_{m\times n}\mathbf{x}_{n\times 1}, \tag{3.1}$$

whereas in the equation above, the unknown signal to be retrieved is **x**, **A** is the sensing matrix, and both of these produce **y** the measurement vector.

For recovering a sparse vector there are few types of algorithms including convex optimization algorithms, greedy algorithms and combinatorial algorithms. Convex algorithms have greater computational complexity but work well with few measurements. On the other hand, the combinatorial algorithms require more measurements but are much faster. Greedy algorithms provide better compromise for complexity and number of measurements.

3.1 Combinatorial Algorithms

The combinatorial algorithms apply group testing to highly structured samples of the signal, but they are mostly not used in compressed sensing as compared to convex optimization algorithms and greedy algorithms.

3.2 Convex Optimization

To obtain the sparse vector that is the unique solution of the underdetermined system of linear equations using the following optimization problem

$$\hat{\mathbf{x}} = arg\ \ min_{\mathbf{x}}||\mathbf{x}||_{\ell_0} \quad subject\ to\ \ \mathbf{Ax} = \mathbf{y}. \tag{3.2}$$

We obtain the sparsest vector using this search technique. However, this approach is difficult because it involves the $||.||_{\ell_0}$ norm which is a non-convex problem. To reduce the computational complexity of the optimization problem we translate the problem from $||.||_{\ell_0}$ to $||.||_{\ell_1}$

$$\hat{\mathbf{x}} = arg\ min_{\mathbf{x}}||\mathbf{x}||_{\ell_1} \quad subject\ to \quad \mathbf{Ax} = \mathbf{y}. \tag{3.3}$$

The ℓ_1 minimization is a good approximation for ℓ_0 minimization as is evident in the figure 3.1. Obviously the two cannot be exactly the same yet ℓ_1 norm provides a feasible solution for this problem. The figure shows the projections at unit circles for ℓ_0,ℓ_1 and ℓ_2 norm.

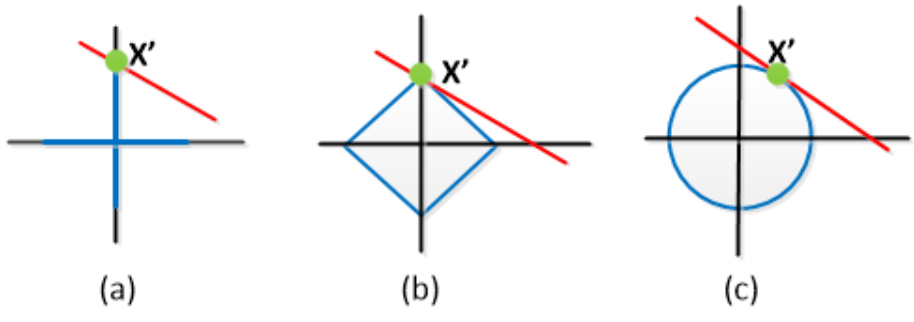


Figure 3.1: Projections at unit sphere for (a)ℓ_0, (b) ℓ_1 and (c)ℓ_2 norms.

If the measurements are contaminated by noise then the minimization problem is required to be changed to

$$min_{\mathbf{x}}||\mathbf{x}||_{\ell_1} \quad subject\ to \quad ||\mathbf{Ax} - \mathbf{y}||_2^2 \leq \epsilon, \tag{3.4}$$

while choosing such that $\epsilon > 0$. The unconstrained version equivalent of this ℓ_1 minimization for a specific regularization parameter $(\lambda) > 0$, is given by

$$min_{\mathbf{x}} \quad \frac{1}{2} \quad ||\mathbf{Ax} - \mathbf{y}||_2^2 + \lambda||\mathbf{x}||_1. \tag{3.5}$$

For compressed sensing problem the convex optimization algorithms that have been developed include interior-point methods [2] and projected gradient methods [6].

3.2.1 Basis Pursuit

Chen and Donoho suggested the Basis Pursuit algorithm (BP) which employs the convex optimization method i.e. it uses the ℓ_1 minimization instead of the ℓ_0 minimization as described in equations 3.3 and 3.2 respectively [8]. The ℓ_1 minimization method is convex i.e. polynomial time complex as opposed to the ℓ_0 minimization which is non-polynomial (NP)

hard. The basis pursuit algorithm may be mathematically described as [7]:

$$min_{\mathbf{x}}||\mathbf{x}||_{\ell_1} \quad subject\,to \;\; \mathbf{Ax} = \mathbf{y}. \tag{3.6}$$

Theorem 3.1: The basis pursuit recovers a signal if $k < \frac{1}{2}(\mu^{-1} + 1)$ and acquires a unique sparsest solution

The fore mentioned theorem states the equivalence of ℓ_0-minimization and basis pursuit, in other words if the signal to be recovered is sparse enough then basis pursuit can solve the NP-complete ℓ_0-minimization problem [38]. The theorem stated below proves that ℓ_0-minimization and ℓ_1-minimization can be treated as same if the condition of RIP is met [2].
Theorem 3.2 If a sensing matrix **A** fulfills RIP (2k,γ) for $\gamma \leq \frac{3}{(4+\sqrt{6})}$, then the sparsest solution can be recovered correctly by Basis Pursuit if the sparsity is below k.

The advantage of basis pursuit algorithm is that it can be transformed into a linear programming (LP) problem. LP works such that it optimizes an objective function by applying constraints in the form of linear equations and inequalities [9]. Optimal solution is obtained by optimizing an objective function, through identification of a similar problem which can be solved by a polynomial time algorithm. Thus, transforming a problem into its convex form helps in obtaining its solution.
To solve the linear programming problem there are following two types of algorithms:
Simplex algorithm: Simplex algorithm is used to solve LP, it is an iterative algorithm and works on the principle of searching optimal solution by moving from one appropriate solution to another lying on the edges of a convex shape [11]. This algorithm is efficient with a lower cost due to its working principle. But its performance is limited in case of degeneracy, and hence takes longer to converge.
Interior point algorithm: Interior point algorithms [10] have a better performance as compared to simplex algorithms for the case of large and degenerate problems. This type of algorithm searches for an optimal solution by iterating inside the convex set. Complexity of these algorithms is in polynomial time. Mosek solver [11] is employed in this thesis for l1 minimization of noise free signals, this solver is based on interior point algorithm.

3.3 Greedy algorithms

Greedy/iterative algorithms have the advantage of being simple to implement and less complex so are fast. Their performance is theoretically equivalent to those of ℓ_1 minimization algorithms. Greedy algorithms work iteratively by approximating two parameters one is the signal coefficients and other support of the original sparse signal until a condition for convergence is fulfilled or by finding an improved estimate of the sparse signal at each iteration to cater for the mismatch to the measured data.

Orthogonal Matching Pursuit (OMP) and iterative thresholding are two greedy algorithms that are oldest known as well as simple.

3.3.1 Orthogonal Matching Pursuit

The idea of matching pursuit was initiated in [12]. Orthogonal Matching Pursuit (OMP) [13] was introduced as an improvement of Matching Pursuit [12]. OMP works by initially selecting the columns of **A** that are most correlated with the measurements. A residue is calculated in each iteration and is obtained to estimate the error present in each subsequent step. Then the same step is performed again by correlating columns with the residue of the signal. Initially the residue is set equal to the measurement vector y, the estimate of the sparse signal is kept equal to 0 and the active set to ϕ. This is represented as

$$\mathbf{r}^0 = y \quad \mathbf{x}^0 = 0 \quad and \quad \mathbf{I}^0 = \phi. \tag{3.7}$$

At every step t, a new column of **A** is selected according to

$$\mathbf{i}_t^* \in arg\ max_i |\langle \mathbf{r}^{t-1}, \mathbf{A}_i \rangle|. \tag{3.8}$$

Then add this column to the active set

$$\mathbf{I}^t = \mathbf{I}^{t-1} \cup \{\mathbf{i}_t^*\}. \tag{3.9}$$

In the end it projects y on the range of $\mathbf{A}_{I_t}$ and the residual is renewed,

$$\mathbf{x}^t = (\mathbf{A}_{I_t}^* \mathbf{A}_{I_t})^{-1} \mathbf{A}_{I_t}^* \mathbf{y}, \tag{3.10}$$

$$\mathbf{r}^t = \mathbf{y} - \mathbf{A}\mathbf{x}^t. \tag{3.11}$$

The criterion to stop the algorithm is either reaching a predefined number of iterations(which also limits the number of non-zeros in $\mathbf{y}$) or the comparison of $\mathbf{A}.\hat{\mathbf{x}} \approx \mathbf{y}$.The details of OMP are given in Algorithm 1.

Algorithm 1 Orthogonal Matching Pursuit (OMP)

Input: $\mathbf{A} \epsilon \mathbb{R}^{m \times n}$, $\mathbf{y} \epsilon \mathbb{R}^{m}$
Result: $\hat{\mathbf{x}}$
Initialization: Residue: $\mathbf{r}^0 = y$,
Sparsity $= k$,
Working Matrix $\hat{\mathbf{A}}$=[],
Set of indices $\mathbf{I}_0$=[]
for $i = 1 : k$ **do**
  λ_i = $\max_{j=1,\ldots,n} |\langle \mathbf{r}^{i-1}, \mathbf{a}_j \rangle|$
  $\mathbf{I}_i = \mathbf{I}_{i-1} \cup \lambda_i$
  $\hat{\mathbf{A}}$ = $[\hat{\mathbf{A}}\mathbf{a}_i]$
  $\hat{\mathbf{x}}_i$ = $\arg\ min_{\hat{\mathbf{x}}}\ ||\mathbf{y} - \hat{\mathbf{A}}\mathbf{x}||_2$ {Least Sqaures for new estimate}
  $\boldsymbol{\eta}_i$= $\hat{\mathbf{A}}\mathbf{x}_i$ {New approximation}
  $\mathbf{r}^i$ = $\mathbf{y} - \boldsymbol{\eta}_i$ {updated residual}
end for

A sufficient condition for convergence is proved in [39] with reference to coherence, for the OMP algorithm the theorem states that:

Theorem 3.3: To obtain unique sparsest solution using OMP sparsity should be such that $k < \frac{1}{2}(\mu^{-1} + 1)$ to recover that solution.

The sufficient condition of RIP for OMP is as follows [40]:

Theorem 3.4: If the sensing matrix $\mathbf{A}$ follows RIP $(k + 1, \frac{1}{(3\sqrt{k})})$ then the OMP algorithm can recover a k-sparse signal accurately.

However, the RIP condition for OMP is weaker than the RIP condition for ℓ_1-minimization.

3.3.2 Iterative Hard thresholding

Generally greedy algorithms such as OMP are considered to be an approach better than the ℓ_1-minimization in terms of speed. Greedy algorithms are faster than interior point or homotopy methods to solve the CS problem. But because there is an inversion step required to be performed by OMP it is still not fast enough. Another family of greedy algorithms is considered better than OMP as it is faster than it. This family is that of iterative thresholding algorithms. Iterative hard thresholding (IHT) algorithm is a good example of iterative algorithms which are more straightforward.

Consider the hard thresholding function

$$T_{hard}(\mathbf{x}) = \begin{cases} \mathbf{x} & |\mathbf{x}| \geq \lambda, \\ \mathbf{0} & |\mathbf{x}| < \lambda. \end{cases} \tag{3.12}$$

And apply this element wise to the signal vector. The equation for iterative hard thresholding is

$$\mathbf{x}^{t+1} = \eta^H(\mathbf{x}^t + \mathbf{A}^*(\mathbf{y} - \mathbf{A}\mathbf{x}^t); \lambda^t), \tag{3.13}$$

where λ^t denotes the value of threshold and $\mathbf{x}^t$ represents the estimate of the sparse signal at time t. The threshold value may be dependent on the iteration t. The underlying thought is that the solution satisfies the CS problem equation $\mathbf{y} = \mathbf{A}\mathbf{x}$. The basic intuition is that since the solution satisfies the equation $\mathbf{y} = \mathbf{A}\mathbf{x}$, the algorithm proceeds by working in the direction of the gradient of $||\mathbf{y} = \mathbf{A}\mathbf{x}||^2$ and then promotes sparsity by applying a nonlinear thresholding function. The fig 3.2 depicts this concept and represents the geometric interpretation of iterative hard thresholding algorithm. The selection of the threshold parameter is an important challenge as it affects the performance of iterative thresholding algorithm. Assume that an oracle informs us of the true value of k. As known the nal solution is k sparse, the threshold may be set equal to the magnitude of the (k+1)th largest coefficient.

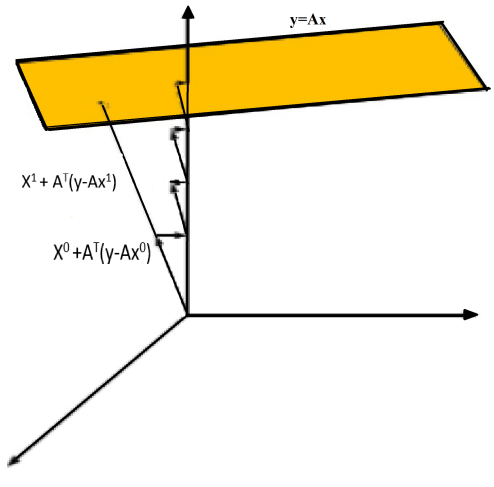


Figure 3.2: Geometric interpretation of the iterative process of IHT algorithm.

To prove the convergence of this algorithm the following theorems are stated now.

Theorem 3.5: Suppose that $k < \frac{1}{3.1}\mu^{-1}$ and $\frac{|\mathbf{x}_0(i)|}{|\mathbf{x}_0(i+1)|} < 3^{\ell_i - 4}$,$\forall_i$,$1 \leq i < k$. The IHT algorithm searches the correct active set in maximum $\sum_{i=1}^{k} \ell_i + k$ steps. After performing this

step all of these elements will remain in the active set and the error will exponentially decay to zero.

Theorem 3.6 Suppose that the sensing matrix **A** satisfies RIP$(3k, \gamma)$ for $\gamma < 1$ and **x** is k sparse. Then the estimate given by IHT at time t satisfies

$$||\mathbf{x}^t - \mathbf{x}_0||_2 \leq \gamma^{-t}||\mathbf{x}_0||_2. \tag{3.14}$$

The IHT algorithm has very simple iterations and the mathematical operations involved for each iteration of IHT are simple which include multiplication of a signal vector by a matrix **A** or $\mathbf{A}^*$. These operations are ordinary and can be efficiently performed for the different sorts of measurement matrices including sparse, partial Fourier,etc measurement matrices.

In iterative hard thresholding the signal estimate is initially kept zero i.e. $\hat{\mathbf{x}} = 0$. Then the algorithm performs a gradient descent step iteratively subsequently followed by hard thresholding until the condition for convergence is satisfied. The steps of iterative hard thresholding algorithm are given in Algorithm 2.

Although the results that are provided in this algorithm are slightly weaker than the results provided for ℓ_1-minimization, in practice and specially in compressed sensing, the algorithm performs much worse than ℓ_1 in the sparsity-measurement tradeoff.

Algorithm 2 Iterative Hard Thresholding (IHT)

Input: CS matrix/dictionary **A**, measurement vector **y**,sparsity level k.
Initialization: $\hat{\mathbf{x}}_0 = 0$.
for $i = 1; i := i + 1$ until stopping criterion is met **do**
$\hat{\mathbf{x}}_i = H_k(\hat{\mathbf{x}}_{i-1} + \mathbf{A}^T(\mathbf{y} - \mathbf{A}\hat{\mathbf{x}}_{i-1}))$
end for
Output: Sparse representation $\hat{\mathbf{x}}$

3.3.3 Iterative Soft thresholding

Among the iterative thresholding algorithms family there is another algorithm called the iterative soft thresholding algorithm. The soft thresholding function as given by

$$T_{soft}(\mathbf{x}) = \begin{cases} |\mathbf{x} - \lambda| & |\mathbf{x}| \geq \lambda, \\ \mathbf{0} & |\mathbf{x}| < \lambda. \end{cases} \tag{3.15}$$

Then the iterative soft thresholding algorithm is defined by the following equation:

$$\mathbf{x}^{t+1} = \eta^S(\mathbf{x}^t + \mathbf{A}^*(\mathbf{y} - \mathbf{A}\mathbf{x}^t); \lambda^t). \tag{3.16}$$

The iterative operation of iterative soft thresholding algorithm is similar to that of IHT. They differ in the methodology of thresholding as this algorithm uses the soft thresholding operator as opposed to the hard thresholding operator used in IHT. The soft thresholding operation used in the iterations provide an advantage that it has a connection with ℓ_1-minimization.

Let us look a bit deeper in to this concept by some more observations. To do this consider the following optimization problem:

$$min_{\mathbf{x}\in\mathbb{R}^n} \frac{1}{2}||\mathbf{z} - \mathbf{x}||_2^2 + \lambda||\mathbf{x}||_1. \tag{3.17}$$

The above mentioned function is minimized when $\mathbf{x} = \eta^S(\mathbf{z}; \lambda)$. The soft thresholding operation is designated as the proximity operator of the ℓ_1-norm and this calls for iterative soft thresholding algorithm with a constant threshold parameter for solving the following problem:

$$\mathbf{Q}_\lambda = \frac{1}{2}||\mathbf{y} - \mathbf{A}\mathbf{x}||_2^2 + \lambda||\mathbf{x}||_1. \tag{3.18}$$

A fascinating feature of the iterative soft thresholding algorithm is that, if the sensing matrix $\mathbf{A}$ is orthonormal and $\mathbf{A}^*\mathbf{A} = \mathbf{I}$, then the solution of $\mathbf{Q}_\lambda$ is going to be $\eta^S(\mathbf{A}^*\mathbf{x}; \lambda)$.

Theorem 3.7: Suppose that $k < \frac{1}{4.1}\mu^{-1}$ and $\frac{|\mathbf{x}_0(i)|}{|\mathbf{x}_0(i+1)|} < 2^{\ell_i - 5}$,$\forall_i$,$1 \leq i < k$. The IHT algorithm searches the correct active set in maximum $\sum_{i=1}^k \ell_i + k$ steps. After performing this step all of these elements will remain in the active set and the error will exponentially decay to zero.

The number of iterations needed to recover the active set, depends on the ratio of the coefficients in IST and IHT. But, this dependency is roughly logarithmic and therefore it works pretty well in practice even in case of high dynamic range signals. Also, the algorithms search the correct active set in a defined number of iterations and once they get the correct active set, they will converge to the exact solution exponentially fast.

Other well-known greedy algorithms are Stagewise OMP (StOMP) [14], Compressive Sampling MP (CoSaMP) [15] and Regularized OMP (ROMP) [16]. Orthogonal Matching Pursuit with Replacement (OMPR) [17] is yet another class of algorithms which include

most iterative (hard)-thresholding algorithms as special cases.

3.3.4 Approximate Message Passing algorithm

Approximate Message Passing (AMP) algorithm as according to compressed sensing theory considers a vector **y** of n measurements obtained from an unknown vector **x** of length N according to $\mathbf{y} = \mathbf{A}\mathbf{x}$, where **A** is the $n \times N$ measurement matrix $n < N$ [18]. Amp works by starting from an initial guess for $\mathbf{x}^0 = 0$, then the iterative first order approximate message passing (AMP) algorithm works according to

$$\mathbf{x}^{t+1} = \eta_t(\mathbf{A}^*\mathbf{z}^t + \mathbf{x}^t), \tag{3.19}$$

$$\mathbf{z}^t = \mathbf{y} - \mathbf{A}\mathbf{x}^t + \frac{1}{\delta}\mathbf{z}^{t-1}\langle\eta_t^{'}(\mathbf{A}^*\mathbf{z}^{t-1} + \mathbf{x}^{t-1})\rangle. \tag{3.20}$$

where $\eta_t(.)$ is the scalar threshold function and $\eta_t^{'}(.)$ is its derivative, $\mathbf{x}^t \in R^n$ is the current estimate of **x**, $\mathbf{z}^t \in R^n$ is the current residual. $\mathbf{A}^*$ denotes transpose of **A**.

AMP is similar to iterative thresholding algorithms as given by

$$\mathbf{x}^{t+1} = \eta_t(\mathbf{A}^*\mathbf{z}^t + \mathbf{x}^t), \tag{3.21}$$

$$\mathbf{z}^t = \mathbf{y} - \mathbf{A}\mathbf{x}^t. \tag{3.22}$$

AMP is different from iterative thresholding algorithms because it has an additional term $\frac{1}{\delta}\mathbf{z}^{t-1}\langle\eta_t^{'}(\mathbf{A}^*\mathbf{z}^{t-1} + \mathbf{x}^{t-1})\rangle$ absent in iterative thresholding, this term helps improve the sparsity-undersampling tradeoff.

Chapter 4

PHASE TRANSITION OF RECOVERY ALGORITHMS

The Shannon Nyquist sampling theorem provides a criterion to sample a bandlimited signal. This theorem has been used in communication systems. But in some cases the desired signal does not require sampling at a rate as high as the Nyquist theorem. Instead sampling can be done at a much lower rate for a specific class of signals, this phenomenon is the subject matter of compressive sensing. It can sample particular signals that are sparse for at least a specific number of non-zero values k. Thus sparse signals form a class of signals that may be sampled below the normal rate hence causing undersampling that is the basis of compressive sampling.

The level of possible undersampling of signals in compressive sensing is necessary to be determined. This way it will be known beforehand that minimum how many samples/measurements may be taken of a signal to accurately reconstruct a signal. Determining this is the basic problem in compressive sensing. To determine the amount of undersampling possible various reconstruction algorithms are tested and designed to optimize it. The lesser the number of samples required to reconstruct the original signal the better the algorithm. Optimization may be achieved by designing algorithms having better capability to reconstruct signals that are highly undersampled or by designing sensing matrices that allow better reconstructions using the same algorithms.

To graphically observe this important phenomenon phase transition curves are taken into account. Before discussing phase transition let us define the notations. Consider an unknown vector $\mathbf{x} \in \mathbb{R}^n$ of interest; then there are the measurements $\mathbf{y} = \mathbf{A}\mathbf{x}$. Here $\mathbf{A}$ is an $m \times N$ matrix i.e. the sensing matrix and $N > m$. Although the system is underdetermined, it has been shown that, when it exists, sufficient sparsity of $\mathbf{x}$ may allow unique identification of $\mathbf{x}$.

4.1 Phase transition

The ℓ_1 minimization is the basic reconstruction procedure adapted in CS for sparse signal recovery. Therefore for the case of ℓ_1-minimization with a random sensing matrix $\mathbf{A}$, there

is a well-defined breakdown point, ℓ_1 norm can successfully recover the sparsest solution provided k is smaller than a certain definite fraction of n.

To define phase transition [32] consider a specific space having a standard set of sparsity/undersampling coordinates let them be $\delta = \frac{m}{N}$ which is the undersampling fraction and let $\rho = \frac{k}{m}$ be the sparsity fraction. As a result we obtain a two-dimensional phase space $(\delta, \rho) \in [0, 1]^2$ which is called the undersamplin/sparsity phase space. In this phase space $0 < \delta \leq 1$ and if δ is equal to 1 then $\mathbf{A}$ is a square matrix and hence the system $\mathbf{y} = \mathbf{Ax}$ will be well determined and so is marginally undersampled. On the other hand if $\delta \ll 1$ then the system $\mathbf{y} = \mathbf{Ax}$ is underdetermined and represents high undersampling. The ρ which is the sparsity/density of the signal that needs to be recovered, if it is close to zero then it means that the signal is very sparse and if it is close to 1 then the signal vector is dense having many non-zero entries.

The phase space $(\delta, \rho) \in [0, 1]^2$ describes the difficulty of a problem instance, as we move upward and towards the left side the problems start to become harder. The phase space is a graphical representation of the success and the failure of the ℓ_1 algorithm which is determined by the location in the phase space where the phases are separated by a curve as shown in fig 4.1.

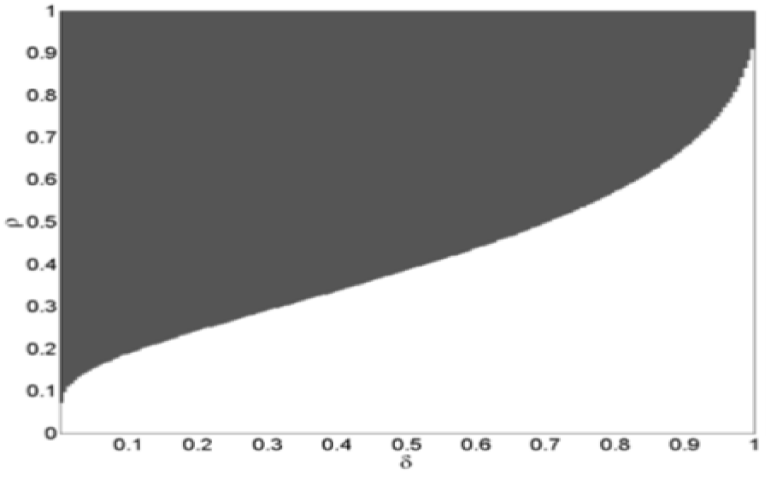


Figure 4.1: Phase Diagram of ℓ_1 minimization x-axis: undersampling fraction $\delta = \frac{m}{N}$. y-axis: sparsity fraction $\rho = \frac{k}{m}$. Dark region is region of failure. Light region: region of success

Clearly there are two phases: one where the fraction of success is essentially one and another where the fraction of success is essentially zero. There is a narrow transition zone

between these two phases where the fraction of success drops from 1 to 0. As seen from the figure 3.1 if (δ, ρ) lies above the curve then it is a success and if below then a failure. This figure is the phase transition diagram.

The phase transition curve may be obtained experimentally or theoretically. Theoretically it can be determined using mathematics specifically by obtaining explicit formulas for a function ρ using combinatorial/polytopes geometry. Let $\epsilon > 0$ then the probability that by P1(ℓ_1 minimization) the sparsest solution for CS recovery problem $\mathbf{y} = \mathbf{Ax}$ reaches 0 or 1 is given by:

$$k \sim n.(\rho_{\ell_1}(\frac{m}{N}) \pm \epsilon). \tag{4.1}$$

Empirical results for the fraction of success represent the transition from success to failure as we move down from one to zero. The theoretical curve lies in between the two extreme curves i.e. of absolute success and absolute failure. This zone between the two extreme curves which is the phase transition zone gets narrow and narrow as N increases and starts to match with a theorem, according to which as the limit of N becomes larger, the zone has vanishing width. In theory the same has been proven for ℓ_1 minimization i.e. Basis Pursuit as well as for other reconstruction algorithms used in compressed sensing.

Let us discuss this phenomenon formally by making an assumption that the signal **x** that is to be reconstructed is taken from a specific distribution. Let that distribution be

$$F_{\mathbf{x}}(\mathbf{x}_i) = (1 - \epsilon)\delta_0(\mathbf{x}_i) + \epsilon G(\mathbf{x}_i), \tag{4.2}$$

where δ_0 is 1 only when $\mathbf{x}_i$ is zero and is zero otherwise. G($\mathbf{x}_i$) is also a distribution function and it provides the distribution for the elements of the signal vector **x** that are not zero, but this is generally unknown at first. Thus the distribution equation shows that the probability that an entry of the signal **x** is zero is $1 - \epsilon$, while the probability that the element will be drawn from another distribution G(.) is ϵ. Now if the maximum number of elements of **x** are N then the expected number of elements that are not zero is $N\epsilon$. Also the Hoeding inequality says that the number of non-zeros elements will concentrate around this value quite rapidly as $N \Rightarrow \infty$.

As discussed in earlier chapter the compressed sensing theory the spare signal **x** is measured by a sensing matrix **A** that is underdetermined and its elements are drawn i.i.d from

a specific distribution. Now assume that the measurement vector **y** produced by $\mathbf{y} = \mathbf{Ax}$ is used by a reconstruction algorithm Alg to recover the signal **x**. However accurate is the algorithm Alg there still might be some samples of the random measurements where the algorithm Alg does not work good enough and cannot yield the correct result. This is possible if the sensing matrix **A** is a null matrix with all entries equal to zero or in the other case when the signal vector **x** is dense. Having considered this possibility, the performance is measured in terms of the probability with which the signal is recovered accurately by the algorithm as a function of (N, δ, ρ, F). For the case when a particular distribution F is taken, the probability of accurate recovery is taken as a function of(N, δ, ρ). In CS the signal vectors to be recovered are of very high dimensions so these probabilities are supposed to be considered as $N \Rightarrow \infty$. By calculating these probabilities a 2-dimensional mapping is obtained which is represented by (ρ, δ) manifesting the accuracy of recovery. This 2-D map is the phase diagram like the fig 3.1 represents the phase diagram of ℓ_1 minimization. Phase transition manifests the sparsity-measurement trade-o for an algorithm. Therefore it is useful for comparison and tuning purposes. Figure 3.2 shows the phase transition curve for IHT algorithm. The figure also shows that the transition becomes sharper and sharper as the dimension of the problem instance increases.

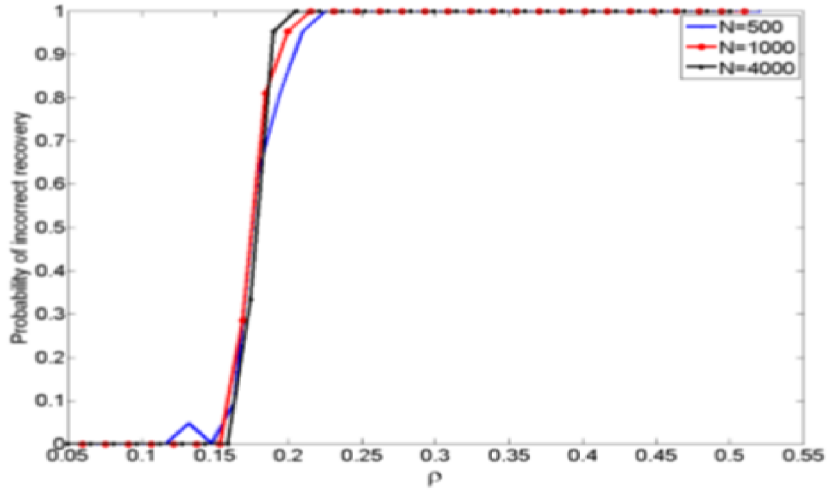


Figure 4.2: Phase Transitions for finite N. Probability of incorrect recovery attempts by IHT shown as a function of sparsity fraction ρ. 3 values of N are considered: 500, 1000, 4000; as N increases the transitions become steep.

4.2 Estimating empirical phase transition

To empirically find the phase transition an experimental setup is considered. So to conduct the experiment we fix a problem suite i.e. a coefficient distribution for generating the problem instance. Then take a problem instance having the measurements but not the original signal to find out if it can be reconstructed by an algorithm by giving a success or failure on the phase transition graph. For the complete experiment there are many problem instances. We fix the N to let us say N=1000 and vary $m = [\delta N]$ and $k = [\rho m]$. Now according to it change m over a range of values from 100 to 900 in nine equal steps. For each combination of m and N we change the sparsity k from 1 to m. As a result we have a grid of $[\delta, \rho]$ values in parameter space $[0, 1]^2$. At the end for every set of the problem instance a series of Monte Carlo trails are performed let them be M=100. After this there will be a number of successes S recorded out of the total M trails. For each problem instance a success is declared when

$$\frac{||\mathbf{x}_0 - \hat{\mathbf{x}}||_2}{||\mathbf{x}_0||_2} \leq tol, \tag{4.3}$$

where tol is the tolerance allowed which is set beforehand generally it may be set as 10^{-3}. The variable for the number of successes is S_i which denotes that there is a success at the ith iteration of the Monte Carlo trials. The distribution for the success is the binomial distribution $Bin(\pi, M)$ where π denotes the success probability $\pi \in [0, 1]$. The result of these Monte Carlo trials is observed by considering the total number of successes in those M trials. The probability of success depends on the parameters k, m and N and is represented as:

$$\pi = \pi(\rho|\delta; N). \tag{4.4}$$

The phase transition for finite-N is the value of ρ at which the reconstruction rate crosses 50% thus the probability of success is:

$$\pi(\rho|\delta; N) = \frac{1}{2} at \rho = \rho^*(\delta; \theta). \tag{4.5}$$

To empirically estimate the location of phase transition using logistic regression from the available data, we require the sparsity k along with the total number of trials M and the number of successes S out of the total M, i.e. this triple (k,M,S). This triple is taken for one value of (n,N), and model S(k,n,N)$\sim$ Bin(πk;M) using a generalized linear model with

logistic link

$$logit(\pi) = a + b\rho, \tag{4.6}$$

where $\rho = \frac{k}{m}$ and $logit(\pi) = log(\frac{\pi}{1-\pi})$ in biometric language, we suppose that the dose-response probability follows a logistic curve.

The parameters $\hat{a}$,$\hat{b}$, give the estimated phase transition from:

$$\hat{\rho}^*(\delta;\theta) = -\frac{\hat{a}}{\hat{b}}. \tag{4.7}$$

The empirical phase transition for two different sets of experiments taken in [32] are shown in fig 3.3. For each experimental data set (k,m,N) three success fraction S/M curves are taken one with 90% reconstruction, another with 50% reconstructions and the third for 10% reconstructions. These three are represented by blue, green and red respectively. An

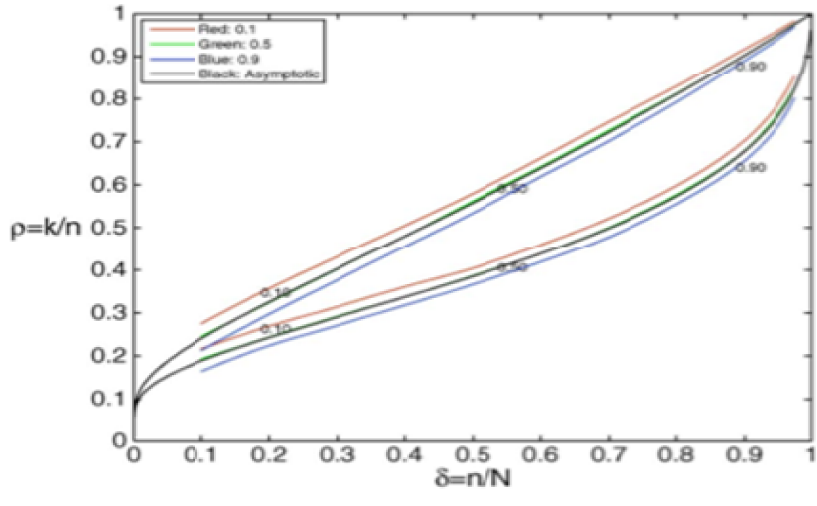


Figure 4.3: : The empirical phase transition for two sets of experiment. Both the lower and the upper set of success probabilities S/M at values 0.1(red), 0.5(green) and 0.9(blue). The black curve closer to the 0.5 curve is the theoretically obtained curve.

important thing to notice is that near the green curve which represents the 50% reconstructions is another curve in black color which is the representation of the formula for phase transition proved through combinatorial geometry. The closeness of the theoretical curve with the 50% reconstruction curves proves that in empirical analysis this is the curve that divides the graph into two regions or phases. Thus making this graph the phase diagram. And the transition zone between the 90% curve and 10% curve is the region where phenomenon of phase transition occurs. The theoretical curve or the 50% curve in experimental case gives

precise information on the number of samples needed to reconstruct **x** with high probability. This defines the limit of undersampling that is possible for a particular sparsity level.

Generally the sensing matrix ensembles considered follow the i.i.d Gaussian distribution. However, other matrix ensembles may also be considered to obtain the phase transition diagram. The empirical phase transition for matrix ensembles other than Gaussian are considered in [33]. These other ensembles are Partial Fourier (obtained by selecting m rows out of the $N \times N$ Fourier matrix), Partial Hadamard(m x N matrix where it has been obtained from Hadamard matrix of N by N by selecting m rows randomly),Rademacher (entries may be +1 or -1 with equal probability), Bernouli (with entries either 0 or 1 having equal probability), Random Sparse Expander Graphs(a random matrix having columns that are not duplicate of others with [p.n] entries as one and others zero), Random Ternary(entries of matrix are 0,+1,-1with probability of a zero entry p and the other two equally likely). The fig 3.4 shows the 50% recovery curves of the seven non-Gaussian matrices along with the associated theoretical curve. With upper set of curves for the non-negative signal solved by LP and seven lower curves using P1.

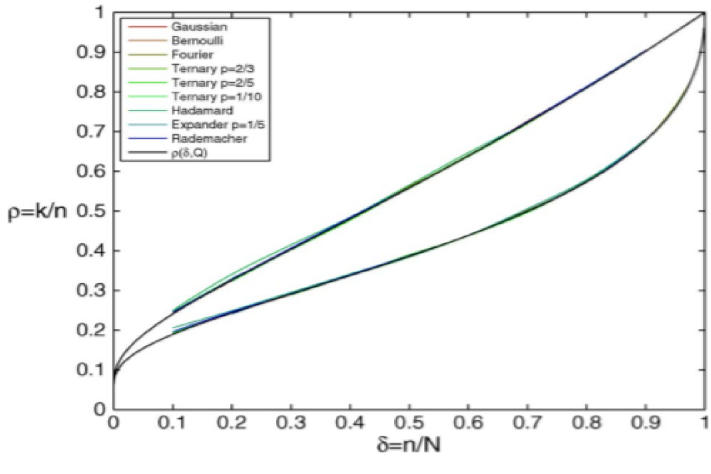


Figure 4.4: Empirical phase transition curves for 50% successful reconstruction of non-Gaussian random matrix ensembles. The upper set of curves obtained using LP and the lower set using P1. These curves closely match the theoretical curves proven for Gaussian ensemble overlaid on them.

To accurately predict the phase transition a formula that identifies the amount of undersampling allowed for generalized sparse objects [34]. The defined formula is for approximate message passing (AMP) algorithms for compressed sensing. For this particular formula the sparsity and undersampling parameters be $\epsilon = \frac{k}{N}$ and $\delta = \frac{m}{N}$ respectively. Hence, the

phase space is defined by $(\epsilon, \delta) \in [0,1]^2$ for all sorts of limiting situations that may be encountered once (k,m,N) grow large. The formula used to get the curve for phase transition using AMP where in compressed sensing setting the distribution for the signal class is denoted as $F_{N,\epsilon}$ and the asymptotic minimax Mean Square Error(MSE) per coordinate using denoiser is given by

$$\delta > \mathcal{MF}(\epsilon|\eta). \tag{4.8}$$

This is the case when for success and for failure the inequality is reversed. Where $M(\epsilon|\eta)$ is given by

$$\mathcal{MF}(\epsilon|\eta) = inf_{\tau>0}\, sup_{v\in F_{i,\epsilon}} E_v\{|X - \eta(X+Z;\tau)|^2\}. \tag{4.9}$$

The figure 3.5 represents the asymptotic phase transition of AMP i.e. obtained by minimax denoising

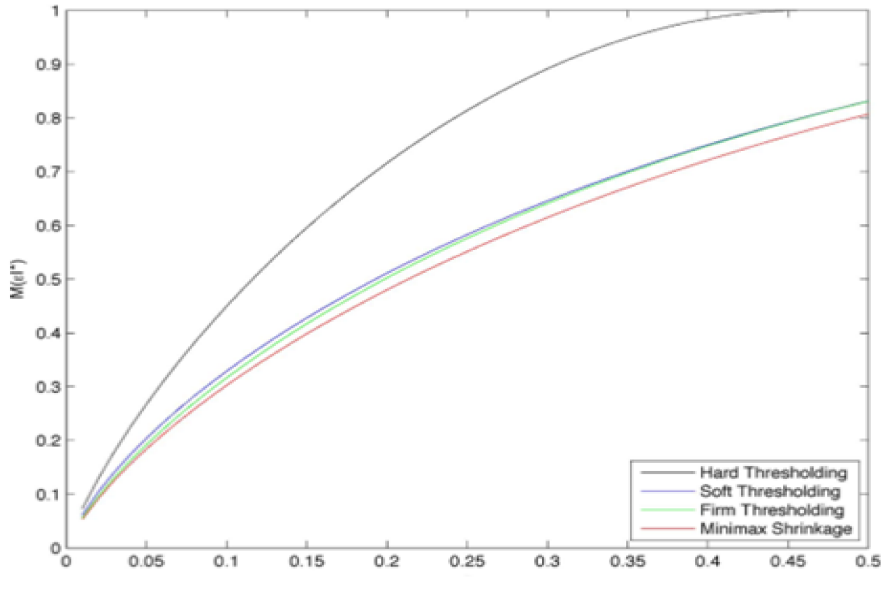


Figure 4.5: Minimax MSE of different separable denoisers in terms of the sparsity parameter ϵ.

If the measurements of a signal have been corrupted with Gaussian noise then the phase transition obtained by minimax MSE of matrix denoising is the same as the phase transition of a noiseless signal measurement with Gaussian distribution [35].

The phase transition for some deterministic matrices have been taken in [36] in which experimental results for various deterministic matrix ensembles such as Spikes and Sines, Spikes and Noiselets, Paley Frames, Delsarte-Goethals Frames, Chirp Sensing Matrices, and Grassmannian Frames have been observed. It was concluded that the Gaussian phase

transition is the same as that obtained for these deterministic matrices. Convex optimization gives successful reconstruction for a sparse signal using deterministic matrices at the same points as the Gaussian random matrices. Fig 3.6 and fig 3.7 show the results from [36].

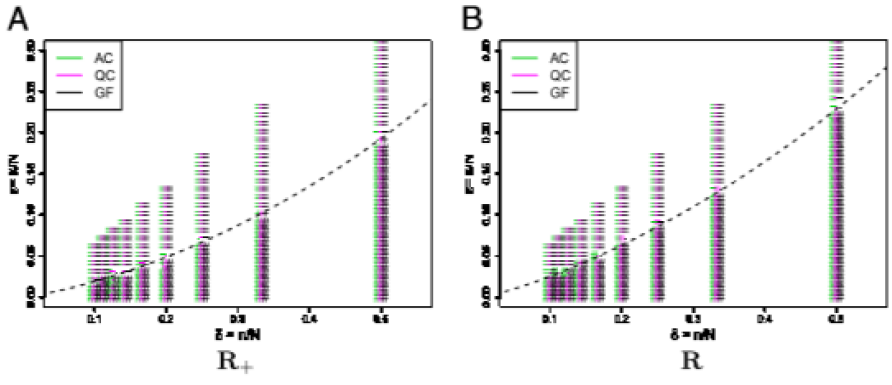


Figure 4.6: Phase transition of chirping frames with different coefficient fields (A) represents real while (B) represents complex.

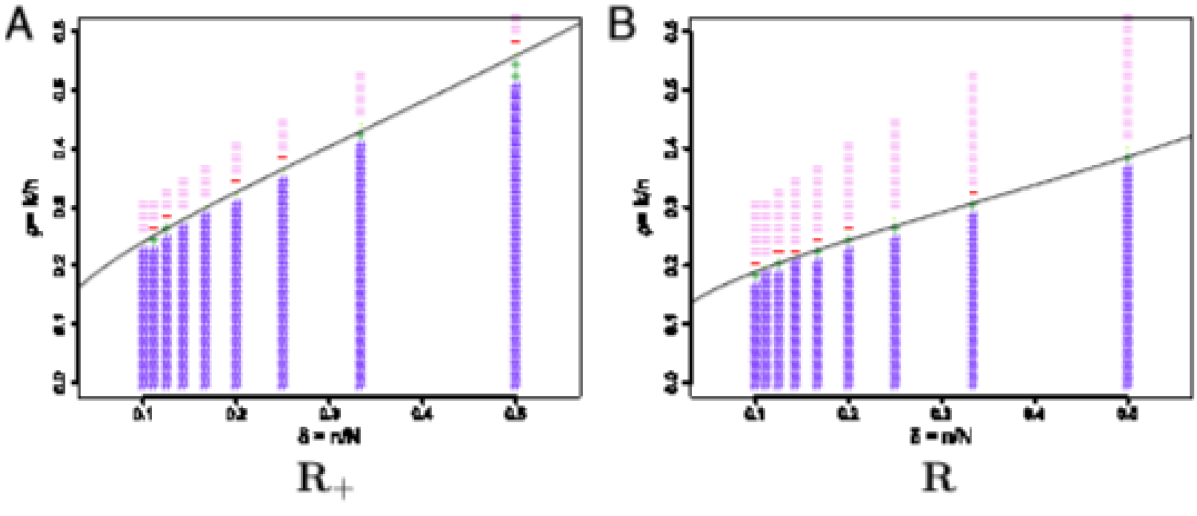


Figure 4.7: Phase transition of DG frames with different coefficient fields (A) represents positive while (B) represents real.

The next figure 6.3 depicts the comparison of the phase transition curves for random sensing matrix and that of BASC based deterministic sensing matrix for OMP. This graph is obtained by employing Orthogonal Matching Pursuit(OMP) algorithm.

The figure shows that the BASC based deterministic sensing matrix which have low coherence show a much better phase transition for the OMP algorithm. Thus depicts that lower coherence of sensing matrix does have an impact on the number of successful reconstructions. Therefore causing an improvement of the phase transition as compared to that of

Random Gaussian based sensing matrix.

The BP algorithm provides better phase transition as compared to OMP if the sensing matrix is random in nature. the phase transition of AMP is nearly the same as that of BP. Therefor for the case of random matrix BP has the highest phase boundary. A comparison is made of the highest possible phase boundary by BP using random matrix to the phase transition obtained by using deterministic matrix for OMP in figure 6.4. The analysis of this figure shows that low coherence based matrices improve phase transition.

Chapter 5

DETERMINISTIC SENSING MATRICES

Random measurements form the basis for testing and simulating in compressed sensing, and its role can be considered similar to the role of random coding in traditional communication system theory i.e. Shannon theory [21]. They provide the lower bound i.e. in terms of performance represent the worst case scenario. In traditional communication system theory, deterministic codes that have fast encoding and decoding algorithms are designed to improve typical rather than worst case performance for reliability. So in compressive sensing deterministic measurements play the same role.

Random sensing matrices are advantageous in the sense that they are easy to construct and provide high success probability in reconstruction. However, they have several drawbacks. The foremost being the excessive complexity in reconstruction, which is the cost for efficient sampling. Then they require significant space to store the entries of a random sensing matrix. Another important one being the absence of any reliable and efficient method of verifying if the random sensing matrix satisfies the RIP, which is the performance criterion for a sensing matrix. In case of random measurements every k-sparse signal is considered to have equal probability as opposed to the approaches used in sensor signal processing, which employ the prior probability distributions [21].

To overcome the drawbacks of random sensing matrix we may exploit specific structures of deterministic sensing matrices. The most important advantage of deterministic sensing is that it provides simplicity in sampling and recovery process. Not only deterministic measurements provide a degree of simplicity but also an improvement in the efficiency and accuracy of reconstruction. If there is a priori information regarding the location of non-zero elements then it is possible to further improve reconstruction efficiency. As opposed to random matrices the entries of a deterministic sensing matrix do not require much storage space [21].

The important conditions that a sensing matrix should satisfy have been discussed in chapter 2. These conditions include Null Space Property, Restricted Isometry Property and Co-

herence. The sensing matrix we employ for sparse recovery should satisfy these conditions. Of the three conditions mentioned coherence is the most feasible and easy to verify condition, because it has a lower complexity. Generally deterministic sensing matrices can be classified into two categories. One type of matrices is based on coherence and the other are the matrices that are based on RIP or some weaker RIPs.

The criterion for the construction of a better sensing matrix is to optimize the mutual coherence present between the columns of the sensing matrix i.e. lower the coherence the better for sparse recovery. Best spherical code algorithm can be used to construct sensing matrices that have low coherence and thus may be suitable for recovery in compressed sensing problems. Before discussing the best spherical codes in detail we briefly present the important prerequisite concepts.

5.1 Basic concepts and definition

Basic concepts that will help develop algorithm for constructing a sensing matrix using best spherical codes are discussed here. The concept of sphere packing is important in getting a set of optimum vectors having desired distance properties .It is defined as:

Definition 5.1. Sphere packing: The method of arranging spheres in a particular space, whereas the spheres are non-overlapping, is called sphere packing [22].

This concept is applicable where spheres whose centers are at a unit distance from the origin, need to be placed without overlapping each other in a particular space. To achieve this optimization with maximum possible radius of these spheres, the idea of sphere packing needs to be applied. To extend it to unit norm vectors, the centers of the spheres are considered unit norm positional vectors. To lower the mutual coherence the distance between vectors needs to be maximized. If the size of the spheres increases so does the distance between their centers meaning that the distance between vectors is increased. Hence the desired result of lower mutual coherence can be achieved.

Frame is an important prerequisite concept in CS

Definition 5.2. Frame: The frame [4] is defined as a set of vectors $\{\phi\}_{i=1}^{n}$ in $\mathbb{R}^k$ where $k < n$ corresponding to a matrix $\mathbf{\Phi} \in \mathbb{R}^{d \times n}$, such that for all vectors $\mathbf{x} \in \mathbb{R}^d$

$$A||\mathbf{x}||_2^2 \leq ||\mathbf{\Phi}||_2^2 \leq B||\mathbf{x}||_2^2, \tag{5.1}$$

with $0 < A \leq B < \infty$ and **A** implies that rows of $\mathbf{\Phi}$ are linearly independent. A frame is called tight frame if A=B and in the case when A=B=1 the frame is called a Parseval frame. Every orthonormal basis is a Parseval frame but the converse is not always true. An equal norm frame is the one if there is a constant v such that $c > 0$ $||\phi_i||_2 = c$ for all i=1, n and is unit norm if c=1. A frame is called an exact frame if no proper subset of the frame spans the inner product space. Each basis for an inner product space is an exact frame for the space (thus a basis is a special case of a frame).

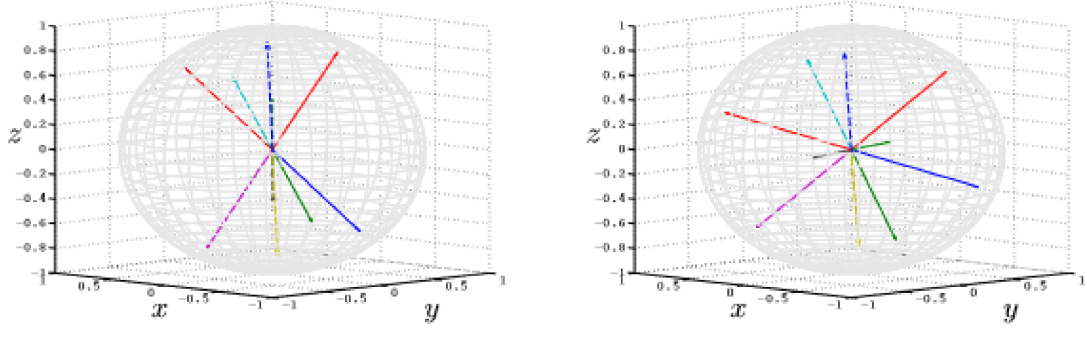
Definition 5.3. Maximum Frame Correlation: The maximum frame correlation for a unit norm frame $\{\mathbf{a}_i\}_{i=1}^n \in \mathbb{R}^m$ is mathematically defined by

$$\mathcal{MF}[\{\mathbf{a}_i\}_{i=1}^n] = max_{i,j,i\neq j}\{|\langle \mathbf{a}_i, \mathbf{a}_j \rangle|\}. \tag{5.2}$$

An important kind of frame in CS is the grassmannian frame defined as:

Definition 5.4. Grassmanian Frame: A frame that consists of vectors $\{\mathbf{a}_i\}_{i=1}^n \in \mathbb{R}^m$ gained as a result of minimizing the maximum frame correlations is a grassmannian frame, mathematically represented as:

$$min\{\mathcal{MF}[\{\mathbf{a}_i\}_{i=1}^n]\}.$$



(a) (b)

Figure 5.1: (a) A frame having 5 vectors along with their antipodals in $\mathbb{R}^3$ (b) Grassmannian frame having 5 equiangularly spaced vectors along with their antipodals in a unit norm sphere.

And the minimum is taken on all unit norm frames $\{\mathbf{a}_i\}_{i=1}^n \in \mathbb{R}^m$. The angle between two unit norm vectors is the parameter that defines the correlation between them. So for

a minimum of the maximum frame correlation the angle between the unit norm should be maximum. A grassmannian frame can also be termed as an equiangular frame, because the later has optimal angles between frame vectors.

The frame constituent vectors and their respective antipodals are plotted on a unit sphere in Figure 3.1(a). The maximized angles between vectors result in a Grassmannian frame.

5.2 Best Spherical Codes

A unit sphere that has its center on the origin of an N-dimensional Euclidean space $\mathbb{R}^N$ having a set of M points located at its surface is called a spherical code and is represented by $C_s(N, M)$ [22]. A point contained in $C_s(N, M) = \{s_i\}_{i=1}^{n}$ is represented in a Cartesian coordinate system by the respective coordinates as $s_i = \{s_{i1}, ., s_{in}\}$ where $s_{ij} \in \mathbb{R}$. Another way of representing the spherical codes is in the form of a matrix $C_s(N, M)$ in $\mathbb{R}^{M \times N}$, where the N columns represent the M-dimensional spherical code unit vectors or points in M-dimensional Cartesian coordinate system.

By maximizing the minimum Euclidean distance between the unit vectors an improved code is obtained known as the best spherical codes represented as $C_{bs}(N, M)$ [24]. As the distance is maximum for the spheres in a best spherical code, so its matrix $C_{bs}(N, M)$ represents spheres that are not intersecting and lie at the center of unit norm spheres and are largest possible in size. Thus they exhibit the densest packing for a specific configuration [26]. The distance distribution exhibited by the best spherical code matrix $C_{bs}(N, M)$ is the same in all coordinate axes orientations. The best spherical codes matrix $C_{bs}(N, M)$ for any particular configuration may be unique or may have more than one distance distribution for the same minimum distance.

For the construction of best spherical codes matrix $C_{bs}(N, M)$, consider a unit norm sphere and let us say n number of charged particles exist on the sphere with a repulsive force field among the particles [27]. The particles move under the influence of this repulsive force and have a local minimum potential energy at some point. To obtain best spherical codes, the local minima needs to be obtained. The generalized potential energy function having a relationship with the distance distribution $g(\mathcal{D})$ given in [28] helps find out the local minima. The distance distribution $g(\mathcal{D})$ in terms of potential energy is represented

as [29]:

$$g(\mathcal{D}) = \sum_{1 \leq i < l \leq n} |s_i - s_l|^{-(v-2)}, \tag{5.3}$$

where $v \in N(v > 2)$. The global minima of $g(\mathcal{D})$ may be obtained to get spherical codes if $v \to \infty$.

Now to consider the problem of obtaining spherical codes for the most appropriate arrangement which can be resolved by representing it as an optimization problem

$$\begin{aligned} Minimize: \ g(\mathcal{D}) = \sum_{1 \leq i < l \leq n}, \ |s_i - s_l|^{-(v-2)} \\ Subject\ to: \ \left\{ \boldsymbol{\phi}_i = \sum_{j=1}^{m} s_{ij}^2 - 1 = 0 \right\}_{i=1}^{n}, \end{aligned} \tag{5.4}$$

This optimization problem is resolved by lagrangian multipliers with $\lambda = \{\lambda_i\}_{i=1}^{n}$.The lagrangian function for the potential energy may be defined as follows:

$$g(\mathcal{D}, \lambda) = g(\mathcal{D}) + \sum_{i=1}^{n} \lambda_i \boldsymbol{\phi}_i. \tag{5.5}$$

There are some necessary conditions required for finding the global minimum of the potential energy which are stated here:

$$\frac{\partial g(\mathcal{D}, \lambda)}{\partial s_{ij}} = 0, \quad and \ \frac{\partial g(\mathcal{D}, \lambda)}{\partial \lambda_i} = 0, \ with \ i = \{1, 2, ..., n\}, \ and \ j = \{1, 2, ..., m\}, \tag{5.6}$$

In order to utilize lagrangian multipliers for finding a solution to the optimization problem defined in 5.4 following procedure is carried out. The distance between two position vectors is given by:

$$|s_i - s_l| = \delta_{il} = \left\{ \sum_{j=1}^{m} (s_{ij} - s_{lj}) \right\}^{\frac{1}{2}}, i, l = \{1, 2, ..., n\}. \tag{5.7}$$

So equation 5.5 becomes

$$g(\mathcal{D}, \lambda) = \sum_{1 \leq i < l \leq n} f(\delta_{il}) + \sum_{i=1}^{n} \lambda_i \boldsymbol{\phi}_i, \tag{5.8}$$

taking partial derivative of lagrangian function w.r.t λ_i

$$\begin{aligned} \frac{\partial g(\mathcal{D},\lambda)}{\partial \lambda_i} &= \frac{\partial}{\partial \lambda_i}\sum_{i=1}^{n}\lambda_i \boldsymbol{\phi}_{_i}, \\ \frac{\partial g(\mathcal{D},\lambda)}{\partial \lambda_i} &= \left\{\sum_{j=1}^{m}(s_{ij}^2-1)\right\}_{i=1}^{n} = 0. \end{aligned} \tag{5.9}$$

Now partial differentiation of lagrangian function w.r.t s_{ij}

$$\begin{aligned} \frac{\partial g(\mathcal{D},\lambda)}{\partial s_{ij}} &= \sum_{1\le i<l\le n}\frac{\partial f_{il}(\delta_{il})}{\partial s_{ij}} + \frac{\partial}{\partial s_{ij}}\sum_{i=1}^{n}\lambda_i\boldsymbol{\phi}_i, \\ \frac{\partial g(\mathcal{D},\lambda)}{\partial s_{ij}} &= \sum_{1\le i<l\le n}\frac{\partial f_{il}(\delta_{il})}{\partial s_{ij}} + \sum_{i=1}^{n}2\lambda_i s_{ij}. \end{aligned} \tag{5.10}$$

As $\frac{\partial g(\mathcal{D},\lambda)}{\partial s_{ij}} = 0$ and $\sum_{i=1}^{n}$ does not contribute to result so:

$$\begin{aligned} \sum_{1\le i<l\le n}\frac{\partial f_{il}(\delta_{il})}{\partial s_{ij}} + 2\lambda_i s_{ij} &= 0, \\ s_{ij} &= -\frac{\sum_{1\le i<l\le n}\frac{\partial f_{il}(\delta_{il})}{\partial s_{ij}}}{2\lambda_i}. \end{aligned} \tag{5.11}$$

taking square on both sides and summation gives:

$$\sum_{j=1}^{m}s_{ij}^2 = \sum_{j=1}^{m}\frac{\left[\sum_{1\le i<l\le n}\frac{\partial f_{il}(\delta_{il})}{\partial s_{ij}}\right]^2}{4\lambda_i^2}, \tag{5.12}$$

as $\sum_{j=1}^{m}s_{ij}^2 = 1$ thus,

$$\sum_{j=1}^{m}\frac{\left[\sum_{1\le i<l\le n}\frac{\partial f_{il}(\delta_{il})}{\partial s_{ij}}\right]^2}{4\lambda_i^2} = 1. \tag{5.13}$$

Now taking square root

$$\sqrt{\sum_{j=1}^{m}\frac{\left[\sum_{1\le i<l\le n}\frac{\partial f_{il}(\delta_{il})}{\partial s_{ij}}\right]^2}{4\lambda_i^2}} = 1, \tag{5.14}$$

$$\sqrt{\sum_{j=1}^{m}\left[\sum_{1\leq i<l\leq n}\frac{\partial f_{il}(\delta_{il})}{\partial s_{ij}}\right]^{2}} = 2\lambda_i. \tag{5.15}$$

Replacing $2\lambda_i$ in equ 5.10

$$\sum_{1\leq i<l\leq n}\frac{\partial f_{il}(\delta_{il})}{\partial s_{ij}} + s_{ij}\sqrt{\sum_{j=1}^{m}\left[\sum_{1\leq i<l\leq n}\frac{\partial f_{il}(\delta_{il})}{\partial s_{ij}}\right]^{2}} = 0. \tag{5.16}$$

After $\delta_{il} = \sqrt{\sum_{j=1}^{m}(s_{ij}-s_{lj})^2}$ and after taking partial derivative the summation may be ignored so:

$$\frac{\partial \delta_{il}}{\partial s_{ij}} = \frac{1}{2}.\frac{2(s_{ij}-s_{lj})}{\delta_{il}} = \frac{s_{ij}-s_{lj}}{\delta_{il}}. \tag{5.17}$$

After processing equation 5.16 using the above mentioned relations we get an equation whose normalized version is given as follows:

$$\left\{ s_i = \sum_{j=1,j\neq i}^{m}\frac{s_i - s_j}{|s_i - s_j|^v} = \sum_{j=1,j\neq i}^{m}\delta_{ij} \right\}_{i=1}^{n} \tag{5.18}$$

An arrangement of spherical codes is shown in the following figure. Solutions of the system

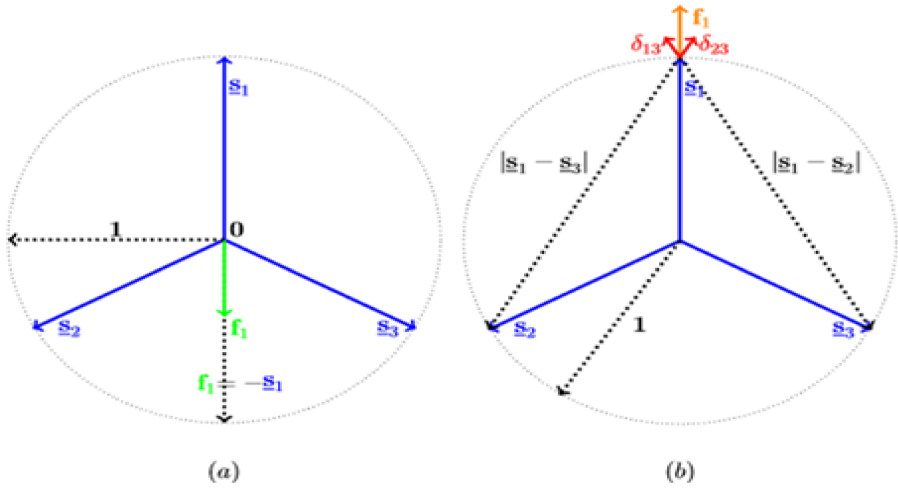


Figure 5.2: Figure 3.2(a) represents an arrangement of spherical codes with m=2 and n=3 while the (b) shows mutual interacting forces and the respective vectors.

of n nonlinear equations given in Equation 5.18 can be seen as a set of fixed points with the

mapping:

$$\mathbf{F}[C_{bs}(N,M)] = \{\mathbf{f}_i(C_{bs}(N,M))\}_{i=1}^{n} \tag{5.19}$$

Of all spherical codes in to themselves. Here $\mathbf{f}_i$ can be written as

$$\mathbf{f}_i = \frac{\sum_{j=1,j\neq i}^{m} \frac{(s_i - s_j)}{|s_i - s_j|^v}}{\left| \sum_{j=1,j\neq i}^{m} \frac{(s_i - s_j)}{|s_i - s_j|^v} \right|}. \tag{5.20}$$

Of all spherical codes in to themselves. This may be applied to equation 5.18.

To obtain spherical codes the local minima search needs to be carried out, so a mapping that follows the nonlinear motion of the charged particle is given as:

$$\Phi[C_{bs}(N,M)] = \{s_i + \alpha \mathbf{f}_i\}_{i=1}^{n}, \tag{5.21}$$

here $\alpha \in \mathbb{R}$ is the dampening factor. The mappings $\mathbf{F}$ and Φ have the same set of fixed points. The dampening factor α supports convergence during the iteration and is calculated again before each iteration [30].

$$C_{bs}(N,M)^{(t+1)} = \mathbf{\Phi}(C_{bs}(N,M)^{(t)}); t = 0, 1, ... \tag{5.22}$$

To fixed points with local minima of the potential energy of the system. As in [31] the mapping is fixed for some points in case of $m > 2$ and $n > m$. While for other settings of m and n there are many more fixed points. The fixed points that result by the method of iteration following the equation 5.22 are quite close to the best optimal arrangement. Hence the codes obtained are called best spherical codes(BSC).

5.3 Best Antipodal Spherical Codes

One of the parameters of judging a sensing matrix is the coherence between its columns the lower the coherence the better. To achieve lower coherence arrangements similar to those of a grassmanian frame are desired to be achieved as they provide the lower bound on coherence known as the Welch bound. To achieve this desired result the best spherical codes discussed earlier need to be modified.

The BSC help maximize the distance between the vectors, but linear dependence between the codes may arise as an issue. Because in such a case the inner product of the different column vectors of the matrix may not be minimal enough.

The distance and inner product of vectors are related as following:

$$dist\langle \mathbf{a}, \mathbf{b} \rangle = \sqrt{||\mathbf{a}||^2 + ||\mathbf{b}||^2 - 2\langle \mathbf{a}.\mathbf{b} \rangle}. \tag{5.23}$$

The simplified version of the above equation for the particular case of best spherical codes is:

$$\{dist\langle \mathbf{a}, \mathbf{b} \rangle\}^2 \propto -\langle \mathbf{a}.\mathbf{b} \rangle. \tag{5.24}$$

Because the vectors each have unit norms for BSC. This relation dictates that by enhancing one the other reduces i.e. for maximal distance the dot product of vectors will be minimum.

$$max\{dist\langle \mathbf{a}, \mathbf{b} \rangle\} \rightarrow min\{\langle \mathbf{a}.\mathbf{b} \rangle\}. \tag{5.25}$$

Since the coherence of vectors depends on the dot product. Coherence as defined earlier in chapter 2 is the maximum of normalized absolute inner product of the column vectors of a sensing matrix. The signs of the vectors effects their dot product as shown below:

$$-\langle \mathbf{a}.\mathbf{b} \rangle = \langle \mathbf{a}. - \mathbf{b} \rangle = \langle -\mathbf{a}.\mathbf{b} \rangle. \tag{5.26}$$

The response to direction changes to distance and dot product are favorable as the distance increases further and the dot product also gets high. A better version of equation 5.25 is given as:

$$max\{dist\langle \mathbf{a}, \mathbf{b} \rangle \; and \; dist\langle \mathbf{a}, -\mathbf{b} \rangle\} \rightarrow min\{\langle \mathbf{a}.\mathbf{b} \rangle\}. \tag{5.27}$$

Conclusively it is known now that the antipodals of the originally existing positional vectors play a vital role in the optimization process and thus must be considered during it. Hence the lower bound of coherence i.e. the Welch bound is approachable via utilizing the antipodals. This modified approach is designated as Best Antipodal Spherical Code (BASC) and the corresponding set is denoted as $C_{abs}(N, M)$. The equation 5.18 previously satisfying the best spherical code arrangement needs to be modified for the new formation to satisfy the modified approach, thus the equilibrium for mutually interacting forces [30] becomes

$$\left\{ s_i = \sum_{j=1, j\neq i}^{m} \left[\frac{s_i - s_j}{|s_i - s_j|^v} + \frac{s_i + s_j}{|s_i + s_j|^v} \right] \right\}_{i=1}^{n}. \tag{5.28}$$

Another term introduced in equation 5.18 to form the above equation caters for the modification in BSC representing the effect of the antipodals. The resulting sensing matrix from

the modified code $C_{abs}(N, M)$ has a worst case coherence closer to the Welch bound. By reversing the signs of the position vectors in the original matrix representing the position of charged particles on the unit norm sphere the modification of antipodals is achieved. The algorithm given describes in detail the process followed for best antipodal spherical codes. The parameter v in the algorithm which exponentially increased in each iteration may have a maximum value v=1024. During the convergence process if the value of v reaches v=2048 then the vectors whose distance from the principal vectors exceeds 1 produce a much larger value in the denominator and consequently leads to an infinite value. Even though the higher the value of v the finer the convergence but if the infinite values show up then the already converged spherical codes get corrupted. Hence to avoid this situation the limitation of v=1024 is necessary. In Figure 3.3, the position vectors, their respective antipodals and the mutually interacting forces are shown for $C_{abs}(2, 3)$.

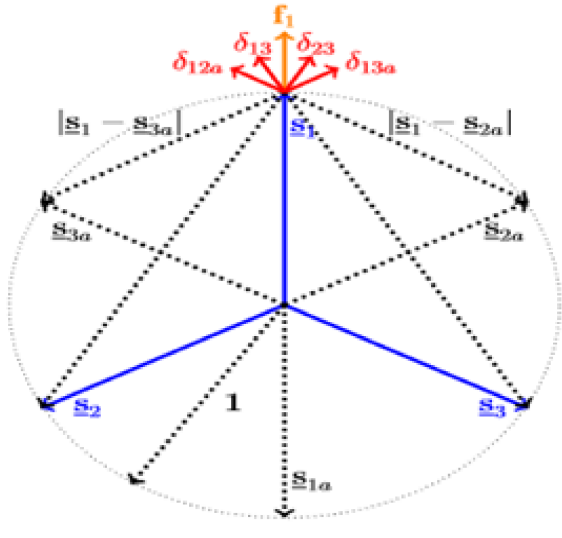


Figure 5.3: Best Antipodal Spherical codes $C_{abs}(2, 3)$ in equilibrium.

Chapter 6

SIMULATIONS AND RESULTS

Phase transition improvement is the goal we have tried to achieve in this thesis. For this purpose deterministic matrices are considered to experiment its impact on few reconstruction algorithms. The algorithms considered are Basis Pursuit(BP), Orthogonal Matching Pursuit(OMP) and Approximate Message Passing(AMP). The set of deterministic matrix belong to the Best Antipodal Spherical Code(BASC). This special kind of deterministic provides the advantage of lower coherence. Phase transition of the mentioned algorithms have been sought using random Gaussian matrices as well as deterministic matrices for the sake of comparison. The experiments are performed in a noiseless environment as well as in the presence of noise.

6.1 Phase Transition

Phase transition is an important parameter in CS theory. It is the sparsity vs undersampling trade-off. Therefore defines a limit to the amount of undersampling possible at a specific sparsity level. The attempt is to somehow improve this limit. There is already a limit defined using random Gaussian matrices.

The phase transition curve represents 50% successful reconstructions. The sparse signal after processing with the sensing matrix gives the measurement vector. Using the measurement vector an estimate of the signal is produced using one of the reconstruction algorithms. The success rate results in the estimation of phase transition. The two environments considered are noise-free and noisy environments. Lets consider both in detail in the next sections.

6.1.1 Noiseless Environment

In a noiseless environment there is no interference, the entire focus is on performance and thus may be considered as ideal case. The original sparse signal vector is reconstructed through three different algorithms Basis Pursuit(BP), Orthogonal Matching Pursuit(OMP) and Approximate Message Passing(AMP) and were described in detail in Chapter 3. The analysis is performed with the following parameters:

- Error Threshold: The error threshold is kept at 10^{-3}.
- $Iteration_{max}$: The maximum number of iterations are taken as 1000.
- Length of signal: The length of sparse signal is taken to be N=400.

The empirical phase transition for the Basis Pursuit algorithm is shown in Figure 6.1.The figure depicts a comparison of the phase transition obtained by using both random Gaussian sensing matrix and deterministic sensing matrix, the Best antipodal spherical code(BASC). As evident from the figure the result for both the sensing matrices are almost identical. Thus showing no improvement by the use of BASC based sensing matrix.

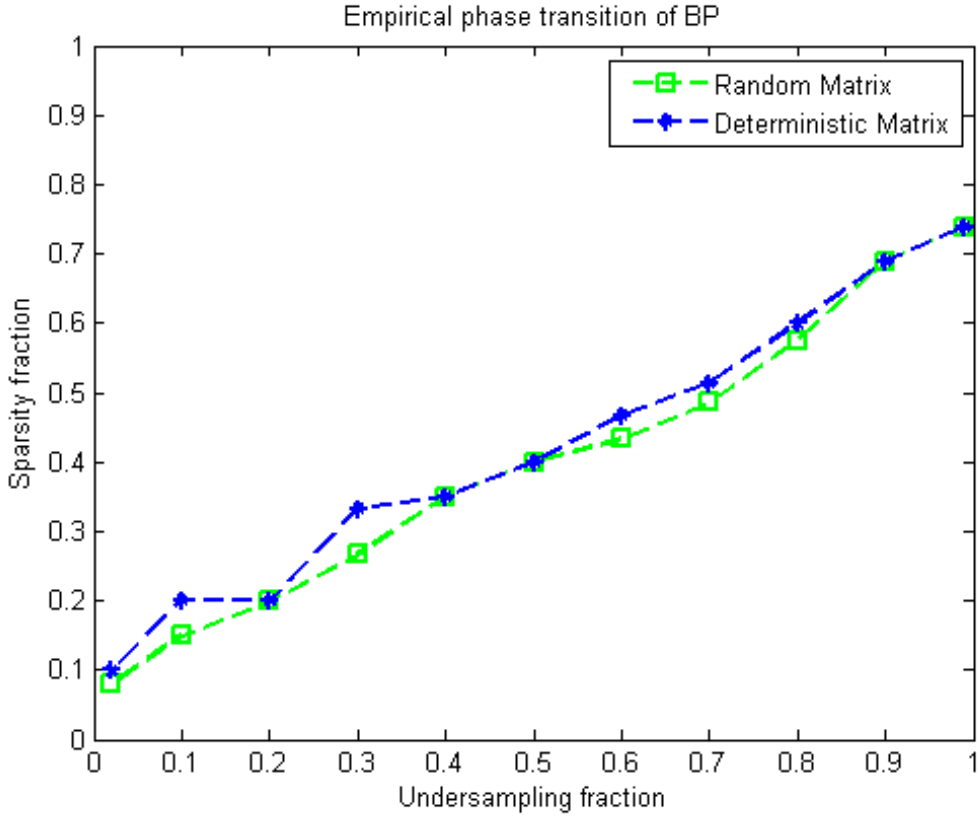


Figure 6.1: Empirical Phase transition of Basis Pursuit: random vs deterministic sensing matrix.

The Figure 6.2 is a representation of phase transition comparison of random and deterministic sensing matrices when the reconstruction algorithm utilized is Approximate message passing (AMP) algorithm. Like BP, this algorithm also shows no improvement with BASC based sensing matrix.

Previous work on AMP algorithm also provides evidence that the deterministic matrices considered in [36] did not provide an improvement but was the same as that of the random Gaussian matrix.

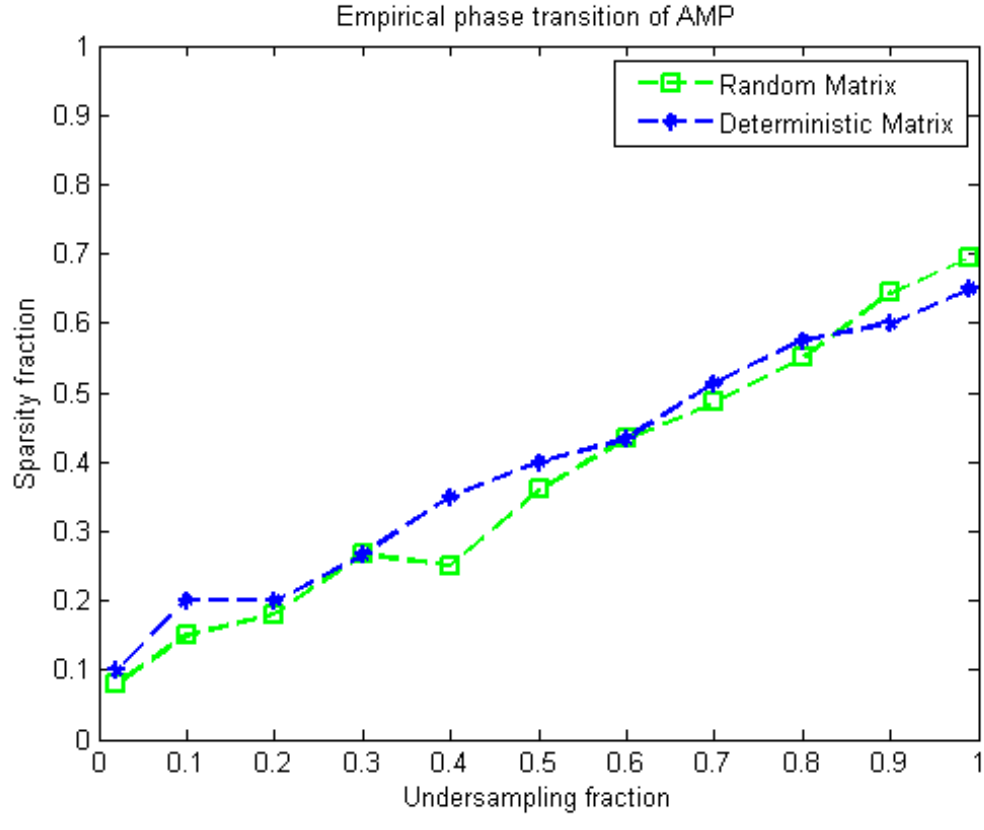


Figure 6.2: Empirical Phase transition of Approximate Message Passing Algorithm: random vs deterministic sensing matrix.

The next figure 6.3 depicts the comparison of the phase transition curves for random sensing matrix and that of BASC based deterministic sensing matrix for OMP. This graph is obtained by employing Orthogonal Matching Pursuit(OMP) algorithm.

The figure shows that the BASC based deterministic sensing matrix which have low coherence show a much better phase transition for the OMP algorithm. Thus depicts that lower coherence of sensing matrix does have an impact on the number of successful reconstructions. Therefore causing an improvement of the phase transition as compared to that of

Random Gaussian based sensing matrix.

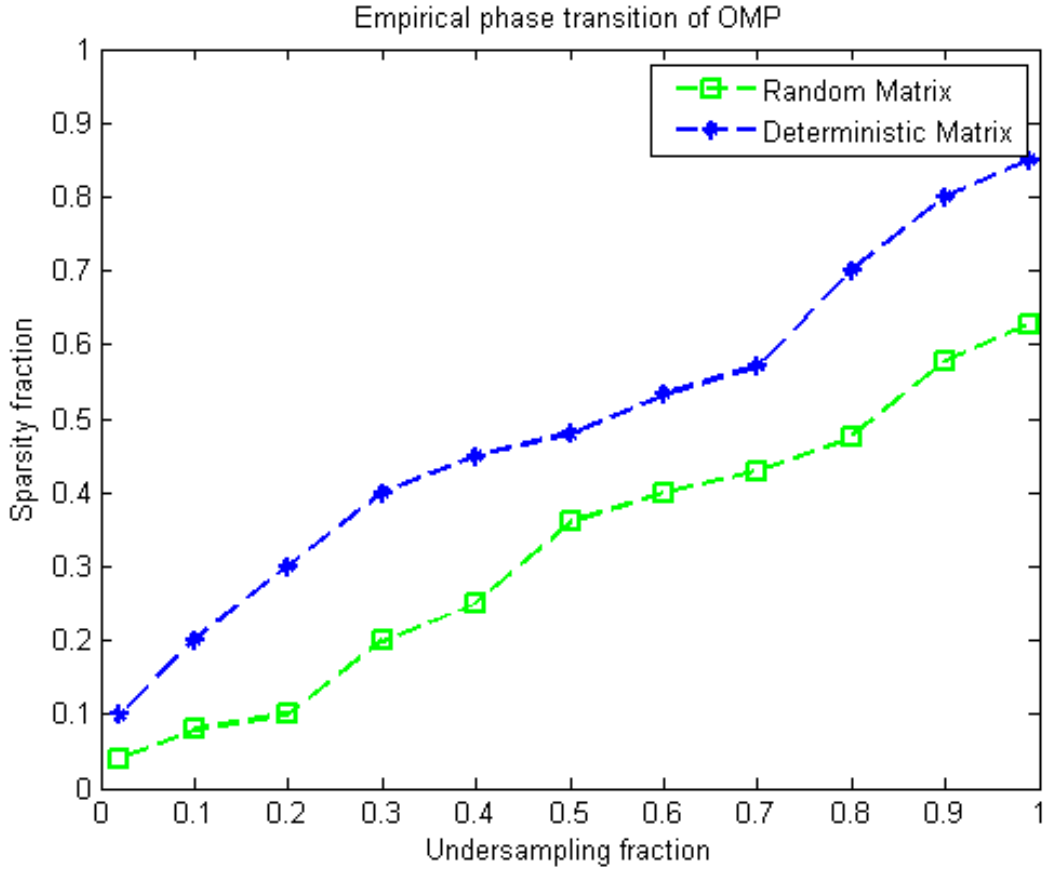


Figure 6.3: Empirical Phase transition of Orthogonal Matching Pursuit(OMP): random vs deterministic sensing matrix.

The BP algorithm provides better phase transition as compared to OMP if the sensing matrix is random in nature. the phase transition of AMP is nearly the same as that of BP. Therefor for the case of random matrix BP has the highest phase boundary. A comparison is made of the highest possible phase boundary by BP using random matrix to the phase transition obtained by using deterministic matrix for OMP in figure 6.4. The analysis of this figure shows that low coherence based matrices improve phase transition. Hence, in the noise-free environment BASC based sensing matrix provides an upper bound on phase transition using OMP algorithm.

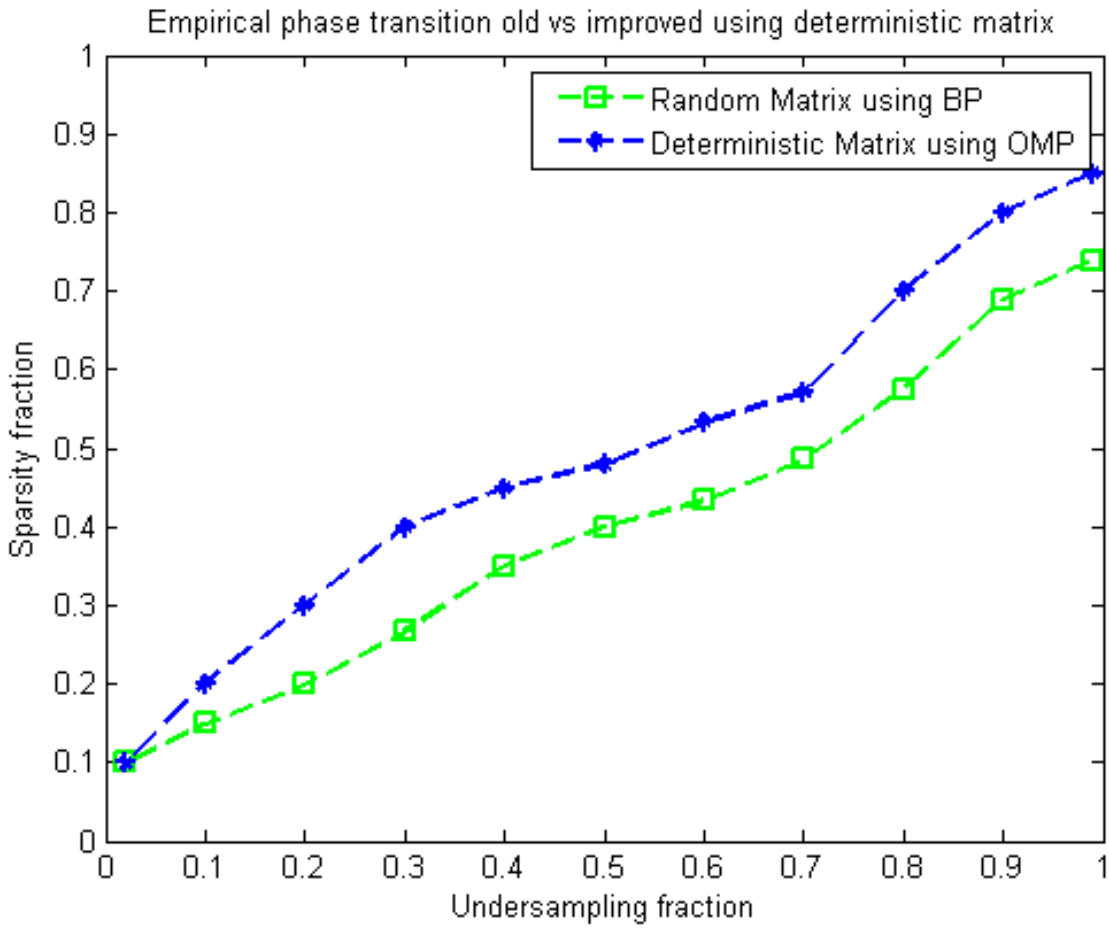


Figure 6.4: Empirical Phase transition old using random vs improved using deterministic sensing matrix.

6.1.2 Noisy Environment

After the analysis in an ideal noise-free environment, the same experiments with same sensing matrices is done in a noisy environment. Same reconstruction algorithms are used to analyze the impact of noise on these algorithms and determine how immune these algorithms are to noise. In this case, signal to noise ratio (SNR) is introduced in the analysis as a third dimension. This allows a study of reconstruction under varying levels of noise. The analysis is performed with the following parameters:

- Error Threshold: The error threshold is kept at 10^{-3}.
- $Iteration_{max}$: The maximum number of iterations are taken as 1000.
- Length of signal: The length of sparse signal is taken to be N=400.
- Signal to Noise Ratio: In order to allow a fair study, SNR is varied from 5 to 25dB.

Noise power is determined in relation to the norm of the measurement vector **y**. Noise power n_p is calculated as follows:

$$n_p = \frac{||\mathbf{y}||_2}{10^{\frac{SNR}{10}}}, \tag{6.1}$$

A normalized random vector **n** is scaled with noise power n_p and added to the measurement vector **y** in order to have a noise incorporated measurement vector $\mathbf{y}_n$:

$$\mathbf{y}_n = \mathbf{y} + \frac{(\mathbf{n}.n_p)}{||\mathbf{n}||_2}. \tag{6.2}$$

Now consider the impact of noise on the phase transition of each algorithm. Figure 6.5 shows the phase transition curves after introducing noise on BP algorithm. As the SNR is varied from 5db to 25 db the curves begin to rise. It is evident from these curves that at 5db and 10db SNR levels the reconstruction rate is below 50% that is why the points lie on the x-axis. However, at SNR=15db the curve starts to rise such that if the SNR is 25db then the impact of noise hardly affects the performance as this curve reaches the noise-free curve on top. Thus to ignore the effect of noise the SNR should be atleast 25db. Figure 6.6 shows the behaviors of AMP in noise which is almost same as that of BP.

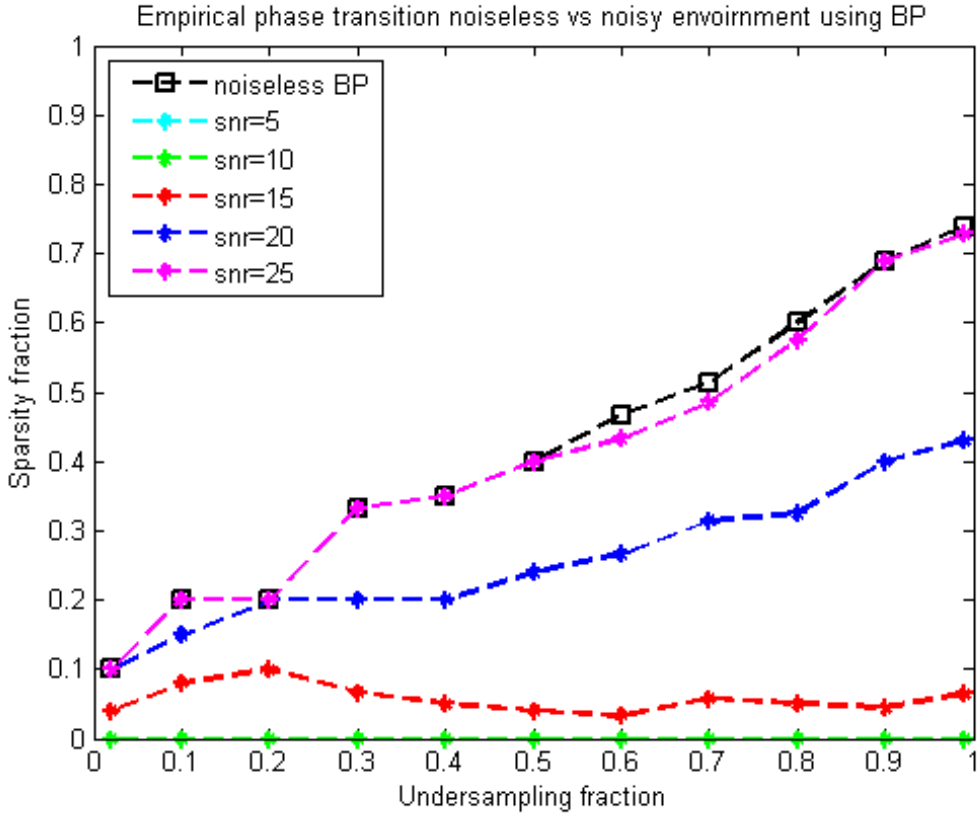


Figure 6.5: Empirical Phase transition of BP in the presence of noise.

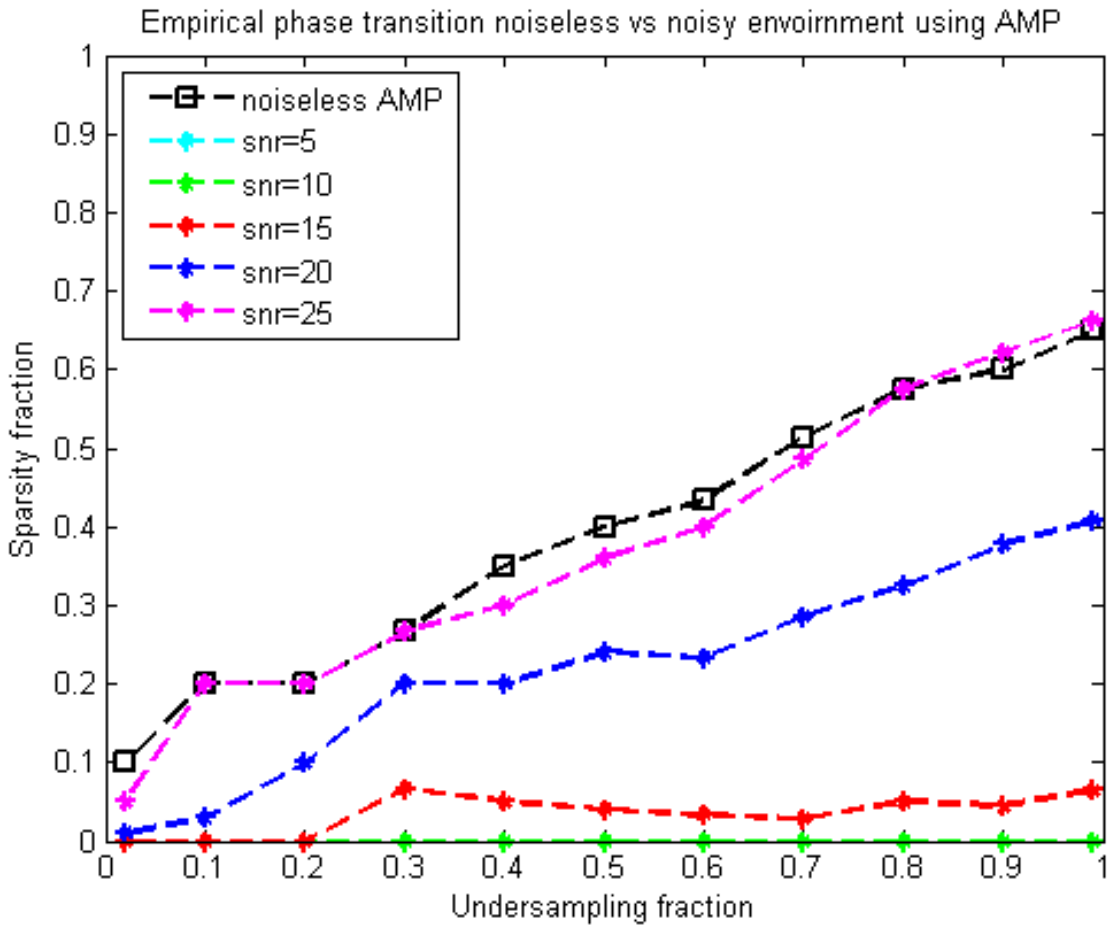


Figure 6.6: Empirical Phase transition of AMP in the presence of noise.

As discussed earlier the improvement in phase transition is visible in case of OMP algorithm. Now lets consider the impact of noise on OMP in the deterministic setting by comparing it with BP and OMP in random setting.

The figure 6.7(a) represents the comparison when the SNR is 5db. At this BP shows no successful reconstructions at all. While OMP in both random and deterministic setting show a slight success at higher undersampling ratios.

Then in figure 6.7(b) when the SNR=10 db the OMP algorithm starts to show some success at the same rate in both random and deterministic case. However, BP still fails at this SNR.

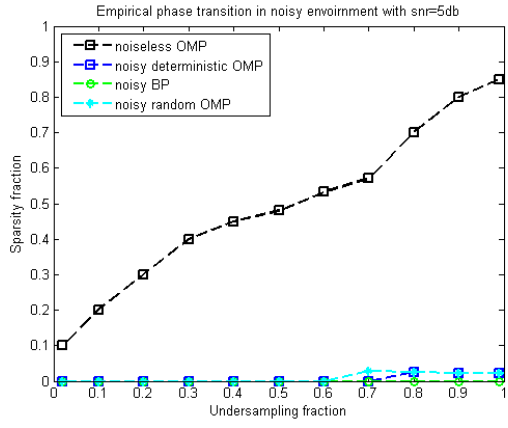
Figure 6.7(c) which is for SNR=15 shows that now BP has also started to show a degree of success however still lower than both cases of OMP.

Then finally in Figure 6.7(d) which represents the impact of noise when when SNR=20. The deterministic matrix based OMP curves reaches closer to the deterministic noiseless curve and leaving the random matrix based curve for OMP.

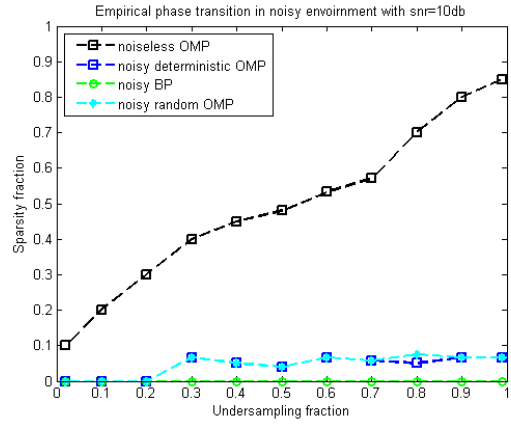
The SNR where all curves almost escape the impact of noise and reach the noise-free curves of their respective algorithms and sensing matrix settings is when SNR=25. This is visible in Figure 6.7(e).

Figure 6.8 combines these curves for different SNR's ranging from 5db up to 25db for the deterministic setting of OMP algorithm. Showing that an SNR of 25 can overcome the effect of noise.

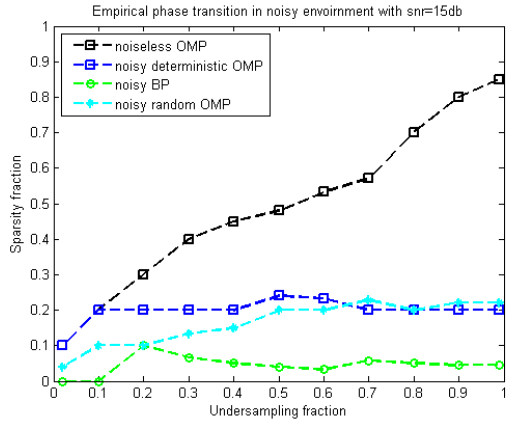
All these figures provide a comparison between the two algorithms BP/AMP and OMP. It can easily be noted that BP and AMP both have lower noise immunity as opposed to OMP. As both do not show any success at SNR 5db and 10db. OMP however, manifested some amount of success which proves that it has a better immunity for noise.However, SNR of 25db can overcome noise in case of all algorithms BP, OMP and AMP.



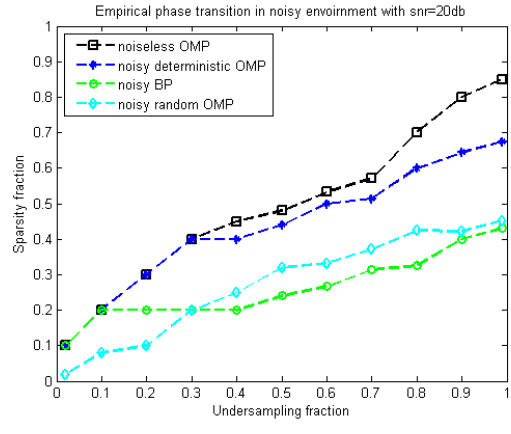
(a) SNR=5



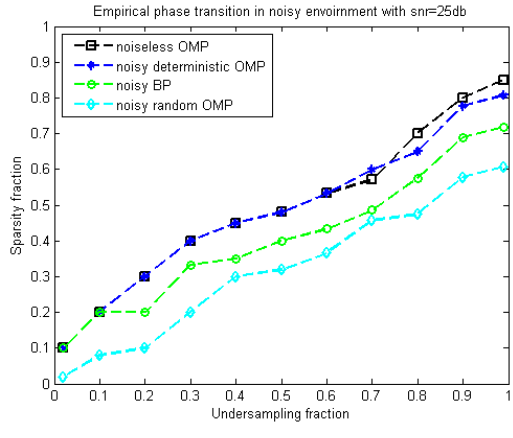
(b) SNR=10



(c) SNR=15



(d) SNR=20



(e) SNR=25

Figure 6.7: Empirical phase transition analysis in noisy environment at different SNRs as compared to the improved phase transition using deterministic matrix (a)SNR=5 (b)SNR=10 (c)SNR=15 (d)SNR=20 (e)SNR=25.

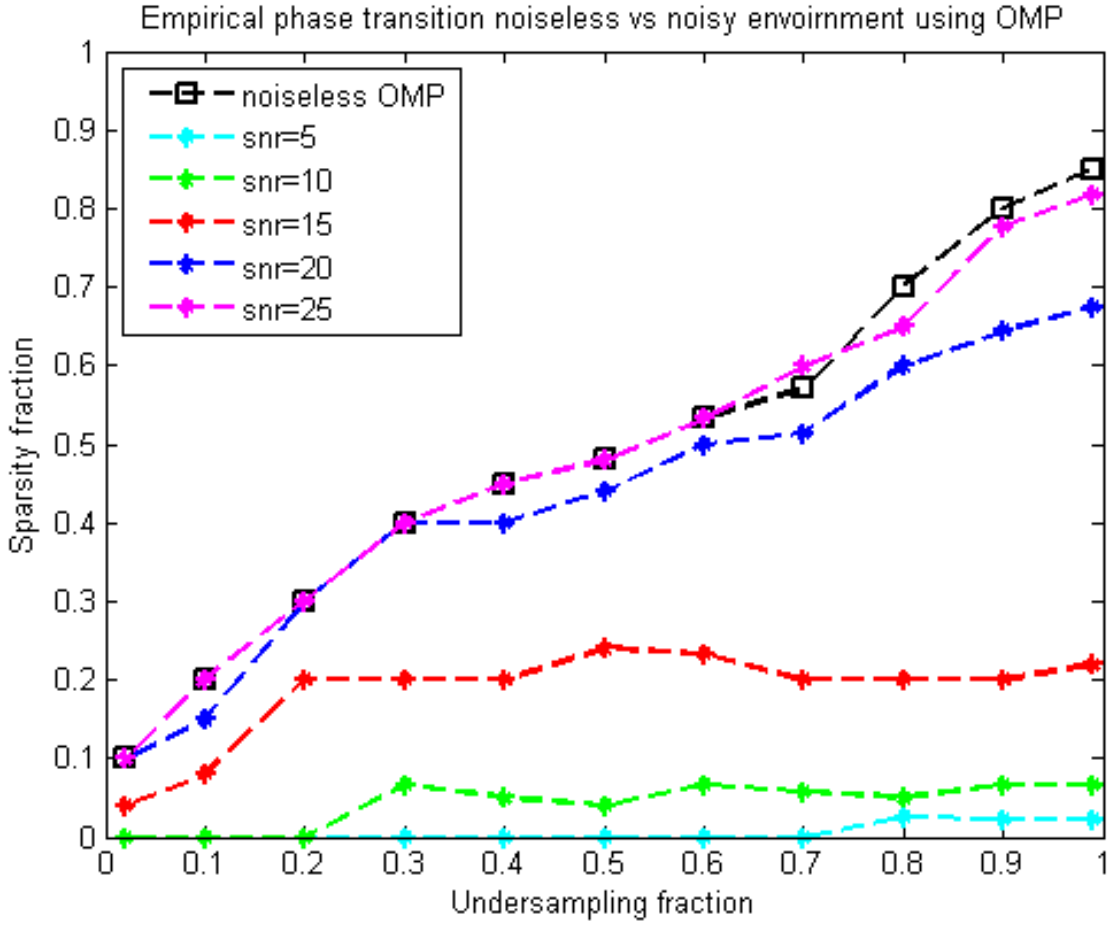


Figure 6.8: Empirical Phase transition in the presence of noise at different SNRs for OMP algorithm.

Chapter 7

CONCLUSION AND FUTURE WORK DIRECTIONS

The basic idea of CS is that a particular class of signals called the sparse signals can be reconstructed using the theory of compressed sensing. The major difference between ordinary communication system theory and CS theory is the number of samples required to retrieve a signal in the later are far less. Thus an important challenge in CS theory is to find out ways to reduce the number of samples necessary to reconstruct a signal of a particular sparsity.

One way is to use sensing matrices that produce better results. Specially designed sensing matrices termed as deterministic matrices reduce the number of samples required to reconstruct a sparse signal. This work has been undertaken in this thesis to use deterministic matrices to reduce the number of samples required to efficiently retrieve a signal of particular sparsity thus improving the phase transition. The deterministic matrix used has a much lower coherence and has improved phase transition in comparison to the random sensing matrix.

In this thesis, we have studied the notion of phase transition i.e. the sparsity undersampling tradeoff, which measures the probability of exact reconstruction. It was analyzed for the BP, OMP and AMP algorithms, by considering deterministic sensing matrices. We show here that reconstruction of sparse objects by these algorithms works well for the deterministic matrices. Particularly for BP and AMP just as well as for Gaussian random matrices. While for OMP deterministically obtained phase transition rises much higher than the random one. The new phase transition for OMP is even higher than that of BP and AMP. We can infer from this that low coherence sensing matrices work best for OMP making it better for reconstruction as compared to BP and AMP which show almost same results in both cases i.e. random and deterministic. Then in the presence of noise it was observed that BP and AMP both have lower noise immunity as opposed to OMP. As both do not show any success at SNR 5db and 10db. OMP however, manifested some amount of success which proves that it has a better immunity for noise.However, SNR of 25db can overcome noise in case of all algorithms BP, OMP and AMP.

A number of areas were identied for future work in similar line of research.

- This work may be extended for phase transition of other types of sparse signals known as group sparse signals such as block sparse signals.
- The phase transition obtained by using sensing matrix of BASC algorithm can be compared to the results obtained by using similar techniques e.g, to simulated annealing.
- The same theory may be utilized for new and better reconstruction algorithms that will further help the cause of CS theory.
- The phase transition of sparse signals having complex values may be checked for improvement using this deterministic matrix.

BIBLIOGRAPHY

[1] D. Donoho, " Compressed sensing," *IEEE Trans. Inform. Theory*, 52:12891306, 2006.

[2] E. Cand'es, J. Romberg, and T. Tao, " Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, 52(2):489509, February 2006.

[3] M. Lustig, D. L. Donoho, and J. Pauly, " Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magnetic Resonance in Medicine*,58(6):11821195, December 2007.

[4] Y.C. Eldar and G. Kutyniok, " Compressed Sensing: Theory and Applications," *Cambridge University Press*, 2012.

[5] I. Daubechies, M. Defrise, and C. De Mol, " An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Comm. Pure Appl. Math.*, 57:1413-1457, 2004.

[6] M.A.T. Figueiredo, R.D. Nowak, and S.J. Wright, " Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE J. Sel. Top. Signa,* 1:586 597, 2007.

[7] S. Chen, D. Donoho, and M. Saunders, " Atomic decomposition by basis pursuit," *SIAM Journal on Scientic Computing,* 20:3361, 1996

[8] S. Chen and D. Donoho, " Basis Pursuit," *Technical Report, Department of Statistics, Stanford university.*

[9] T. Cormen, C. Stein, R. Rivest, and C. Leiserson, " Introduction to Algorithms" *McGraw-Hill Higher Education, 2nd edition,* 2001.

[10] D. Bertsimas and J. Tsitsiklis, " Introduction to Linear Optimization," *Athena Scientic, 1st edition,* 1997.

[11] MOSEK. version 6.0.0.148. MOSEK ApS, Fruebjergvej 3,*Copenhagen,* 2012.

[12] S. Mallat and Z. Zhang, " Matching pursuit with time-frequency dictionaries," *IEEE Transactions on Signal Processing,* 41:33973415, 1993.

[13] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, " Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," *In Proc. of the 27th Asilomar Conference on Signals, Systems and Computers,* 1:40-44, 1993.

[14] D. Donoho, Y. Tsaig, I. Drori, and J. Starck, " Sparse Solution of Underdetermined Linear Equations by Stagewise Orthogonal Matching Pursuit " *Preprint,* 2007.

[15] D. Needell and J. Tropp, " CoSaMP: Iterative signal recovery from incomplete and inaccurate samples " *Appl. Comput. Harmon. Anal.,* 26:301321, 2008.

[16] D. Needell and R. Vershynin, " Uniform Uncertainty Principle and signal recovery via Regularized Orthogonal Matching Pursuit, " *Found. of Comput. Math.,* 9:317334, 2009.

[17] P. Jain, A. Tewari, and I. Dhillon, "Orthogonal Matching Pursuit with Replacement," *In Proc. Neural Inform. Process. Systems Conf. (NIPS),* 2011.

[18] D. Donoho, A. Maleki, and A. Montanari, " Message passing algorithms for compressed sensing," *Proc. Natl. Acad. Sci. USA, 106:1891418919,* 2009.

[19] E. Cand'es and T. Tao, " Decoding by linear programming," *IEEE Trans. Inform. Theory,* 51(12):42034215, 2005.

[20] E. Kreyszig, " Advanced Engineering Mathematics," *Wiley, 7th edition,* 1993.

[21] C. Howard, Sina and Jafarpour," Construction of a Large Class of Deterministic Sensing Matrices that Satisfy a Statistical Isometry Property," *The Journal of Selected Topics in Signal Processing (J-STSP)* (Vol:4 , Issue: 2) , 2010.

[22] J. H. Conway, N. J. A. Sloane, and E. Bannai, " Sphere-packings, lattices, and groups," *Springer-Verlag New York, Inc.,* New York, NY, USA, 1987.

[23] T. Strohmer and R. Heath, " Grassmannian frames with applications to coding and communication," *Applied and Computational Harmonic Analysis,* 14(3):257275, 2003.

[24] N. Sloane, " Tables of sphere packings and spherical codes," *IEEE Transactions on Information Theory,* 27(3):327338, September 2006.

[25] MATLAB. version 8.0.0.783 (R2012b). *The MathWorks Inc., Natick, Massachusetts,* 2012

[26] R. Rankin, " The closest packing of spherical caps in n dimensions," *Proceedings of the Glasgow Mathematical Association,* 2:139144, 6 1955.

[27] J. Leech, " Equilibrium of Sets of Particles on a Sphere," *The Mathematical Gazette,* 41(336):8190, 1957.

[28] D. Lazich, " Class of block codes for the gaussian channel," *Electronics Letters, 16(5):185186,* February 1980.

[29] D. Lazich, T. Bece, and P. Krstajic. On the construction of the best spherical code by computing the xed point *IEEE International Symposium on Information Theory,* page 74, 1986.

[30] D. Lazich, H. Zorlein, and M. Bossert, " Low coherence sensing matrices based on best spherical codes," *In Systems, Communication and Coding (SCC), Proceedings of 2013 9th International ITG Conference on,* pages 16, 2013.

[31] D. Lazich, Senk V., and Zamurovic R. , " An efficient numerical procedure for generating best spherical arrangements of points," *In Systems, Communication and Coding (SCC), Proceedings of the International AMSE88, Istanbul Conference Modeling and Simulation, volume 1C,* pages 267 278, 1988.

[32] D. Donoho and J. Tanner, Precise undersampling theorems, *Proc. IEEE, vol. 98, no. 6,* pp. 913924, 2010.

[33] D. Donoho and J. Tanner, Observed universality of phase transitions in high-dimensional geometry, with implications for modern data analysis and signal processing, *Phil. Trans. R. Soc. A, vol. 367, no. 1906,* pp. 42734293, 2009

[34] D. Donoho, I. Johnstone, and A. Montanari, Accurate Prediction of Phase Transitions in Compressed Sensing via a Connection to Minimax Denoising, *IEEE Transaction on Information Theory,* Vol.59 ,NO.6 ,2013.

[35] D. Donohoa, M. Gavisha, and A.Montanari, The phase transition of matrix recovery from Gaussian measurements matches the minimax MSE of matrix denoising, *Proceedings of the National Academy of Sciences,* 2013.

[36] H. Monajemia, S. Jafarpourb, M. Gavishc, and D. Donoho, Deterministic matrices matching the compressed sensing phase transitions of Gaussian random matrices, *Proceedings of the National Academy of Sciences,* 2012.

[37] R. A. Horn and C. R. Johnson, Matrix analysis *Cambridge University Press, rst edition,* 1985.

[38] D. L. Donoho and M. Elad, Maximal sparsity representation via minimization, *Proceedings of National Academy of Sciences,* 100:21972202, March 2003.

[39] J. A. Tropp, Just relax: Convex programming methods for identifying sparse signals, *IEEE Transactions on Information Theory,* 51(3):10301051, 2006.

[40] M. A. Davenport and M. B. Wakin, Analysis of orthogonal matching pursuit using the restricted isometry property, *IEEE Transactions on Information Theory,* 2010.

[41] T. Blumensath and M. E. Davies, Iterative hard thresholding for compressed sensing, *Applied and Computational Harmonic Analysis,* 27(3):265274, 2009.

[42] D. L. Donoho, "De-noising by soft-thresholding," *IEEE Transactions on Information theory*, vol. 41, no. 3, pp. 613-627, 1995.