NOISY SPARSE PHASE RETRIEVAL USING THRESHOLDED WIRTINGER FLOW ALGORITHM

PROJECT REPORT

submitted by

ROSEMOL JOSE
(TRV18ECSP13)

to
the APJ Abdul Kalam Technological University
in partial fulfilment of the requirements for the award of the Degree
of
Master of Technology
in
Signal Processing

Department of Electronics and Communication
Government Engineering College
Barton Hill
Trivandrum-35
July 2020

DECLARATION

I hereby declare that the project report ("Noisy Sparse Phase Retrieval Using Thresholded Wirtinger Flow Algorithm") submitted with partial fulfilment of the requirements for the award of degree of Master of Technology of the APJ Abdul Kalam Technological University, Kerala, is a bona fide work done by me under the supervision of Dr.Birenjith P. S. This submission represents my ideas in my own words and wherever ideas or words of others have been included, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to the ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

Place	: Thiruvananthapuram	Signature of student	: .....
Date	: July, 2020	Name of student	: ROSEMOL JOSE

DEPARTMENT OF ELECTRONICS AND COMMUNICATION

Government Engineering College

Barton Hill

Trivandrum-35

CERTIFICATE

This is to certify that the project report entitled **Noisy Sparse Phase Retrieval Using Thresholded Wirtinger Flow Algorithm** submitted by **ROSEMOL JOSE (TRV18ECSP13)** to the APJ Abdul Kalam Technological University in partial fulfilment of the requirements for the award of the Degree of Master of Technology in Signal Processing is a bonafide record of the project work carried out by her under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Prof. Birenjith P. S.
Assistant Professor
Dept of ECE
(Project Guide)

Prof. Pradeep R.
Associate Professor
Dept of ECE
(Project Coordinator)

Prof. Asha Murali
Associate Professor
Dept of ECE
(Project Coordinator)

Prof. Rishidas S.
Associate Professor
Dept of ECE
(Head of the Department)

ACKNOWLEDGEMENT

I wish to record my indebtedness and thankfulness to all who helped me prepare the project report titled Noisy Sparse Phase Retrieval Using Thresholded Wirtinger Flow Algorithm and present it in a satisfactory way.

I am especially thankful to my guide and supervisor Prof. Birenjith P. S. in the Department of Electronics and Communication for giving me valuable suggestions and critical inputs in the preparation of this report. I am also thankful to Prof. Rishidas S., Head of Department of Electronics and Communication for his whole hearted encouragement. I also extend my sincere gratitude to the internal supervisors Prof. Pradeep R., Prof. Asha Murali and Prof. Celine Mary Stuart, who monitored the work continuously and suggested the required improvements in each stage.

My friends in my class have always been helpful and I am grateful to them for patiently listening to my presentations on my work related to the project.

ROSEMOL JOSE

(Reg. No. TRV18ECSP13)

M. Tech. (Signal Processing)

Department of Electronics and Communication

Government Engineering College, Barton Hill

ABSTRACT

Phase retrieval has been an important and difficult problem in the area of signal processing. The central problem is to estimate a signal when there is access to its magnitude but the phase is lost. Previously, Gradient Descent Algorithm proposed by Candes et al., was used to retrieve phase in noiseless conditions. But when the signal is sparse and independent noise samples are subexponential, the algorithm fails to recover the signal from noisy quadratic measurements. As an improvement, a thresholding step based on Gaussian model along with an initialisation step is added onto the existing algorithm by Cai et al. This modified algorithm is known as the Thresholded Gradient Descent Algorithm or the Thresholded Wirtinger Flow. It is computationally efficient, robust and gives optimal convergence rates and also finds applications in areas like X-ray crystallography, diffraction, array imaging, microscopy, astronomy. The goals are to implement and evaluate the algorithm against real datasets and study the convergence rates of the algorithm.

CONTENTS

LIST OF FIGURES

ABBREVIATIONS

(List in the alphabetical order)

Abbreviation	Expansion
ERM	Empirical Risk Minimisation
LAD	Least Absolute Deviations
NSR	Noise to Signal Ratio
NP	Nondeterministic Polynomial Time
PCA	Principal Component Analysis
TWF	Thresholded Wirtinger Flow

Chapter 1

INTRODUCTION

A signal is defined completely by both its magnitude and phase. These two components are necessary for the perfect reconstruction of a signal. The magnitude of a signal is the measure of energy contained in the sinusoids that form it whereas the phase gives the positioning or translation of the sinusoids. The phase of a signal is as important as the magnitude and its loss would lead to the loss of signal itself. In most of the signal processing applications, it is easy to retrieve the magnitude of a signal but unfortunately the phase is lost because it cannot be recorded directly as the magnitude. The importance of phase in reconstruction is illustrated in the figure 1.1.

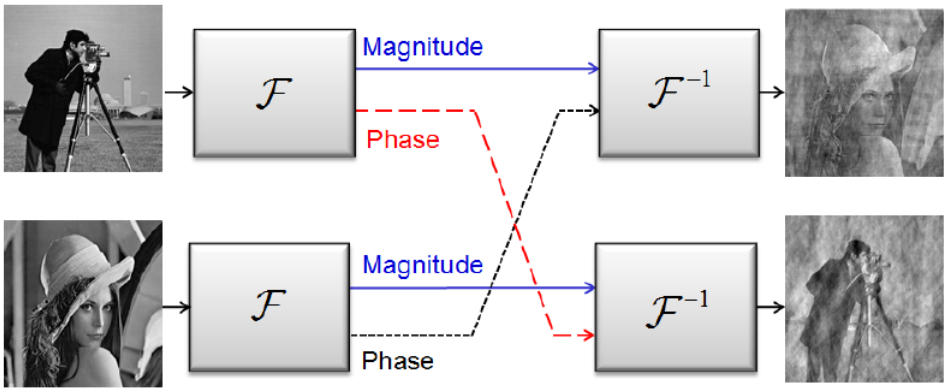

Figure 1.1: The above figure shows the importance of phase information for image reconstruction. Source: https://www.researchgate.net/figure/The-importance-of-Fourier-phase-Two-images-Cameraman-and-Lenna-are-Fourier transformed.

While reconstruction, the phases of the images on the left side are altered without changing their magnitude, and the resulting images are different from the original with a predominating imprint of the other image. This is conclusive of the

significance of the phase.

The scenario of phase loss is a very common and open problem. For example, consider an object illuminated by a coherent wave, in the far field, diffraction pattern gives the Fourier transform of the object. Optical detection devices such as cameras, human eye, photodiodes measure the photon flux (Fourier Transform) which is proportional to the magnitude square of the field. They do not allow the direct recording of the phase since the electromagnetic fields oscillate at a rate of $\sim$1 kHz. Thus, the phase is missing or corrupted in the measurements obtained. This difficult situation can also occur in areas like X-ray crystallography, astronomy, array imaging, microscopy, interferometry, optics, spectroscopy, tomography etc. Here we use various phase retrieval techniques.

1.1 MATHEMATICAL FORMULATION OF PHASE RETRIEVAL PROBLEM

Phase retrieval is simply the reconstruction of a signal from the (squared) modulus of its Fourier transform. Conventionally, it is addressed as the solving of linear equations with a missing sign or phase. Here we try to estimate a p-dimensional phaseless signal x by probing the signal using a set of sensing vectors $a_j^*, j = 1,2,\ldots,$ m from the available quadratic observations y_j.

The observations can be mathematically represented as:

$$y_j \quad = \quad |a_j^* x|^2, \qquad j = 1,2,\ldots,m \tag{1.1}$$

where $A = [a_1,\ldots,a_m]^*$ is the design matrix with a set of p-dimensional independent Gaussian sensing vectors a_j^*. In most of the cases, it is impossible to directly measure $a_j^* x$. The observations recorded may be the magnitude or energy of the signal $a_j^* x$. Also in real life applications, the observations are always incurred by noise.

So incorporating the noise factor into the formulation, the problem gets modified to:

$$y_j = |a_j^* x|^2 + \varepsilon_j, \quad j = 1,2,\ldots,m \tag{1.2}$$

where $\varepsilon_j = (\varepsilon_1,\ldots,\varepsilon_m)^*$ is a vector of stochastic noise with expectation $\mathbb{E}(\varepsilon) = 0$. This is a general non-convex quadratic problem. The sensing vectors are assumed real-valued, and then extended to the complex case. Our analysis requires x to be fixed and be independent of the sensing matrix A. Hence, the same A cannot be used in each iteration; we need to resample A in every iteration. The noise ε_j is assumed to follow independent centered sub-exponential random noise model. Sub exponential noise is a type of heavy tailed noise. The probability of occurrence of noise is higher at the tails (i.e., for larger values of x) as shown in figure 1.2 below.

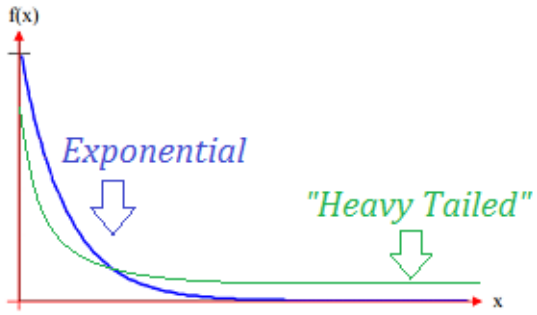

Figure 1.2: The above figure shows exponential and heavy tailed distributions.

Hence the problem becomes more complex unlike the usual Gaussian approximation of noise and thus the solution will lead to an optimal case. This approximation is used since heavy-tailed noise occur in many applications, especially in optics due to the photon counting.

In many applications, the signal $x \in R^n$ might have a sparse representation or becomes sparse after some deterministic linear transformation. For instance, X-ray

crystallography deals with sparsely distributed atoms or molecules while astronomical imaging assumes sparsely distributed stars. Hence, an additional assumption of signal being sparse is also considered. The sparse modeling applied to x (to both phase and amplitude) leads to an algorithm efficient for solving even heavily noisy data. Sparsity assumption also leads to reduced computation and easy reconstruction. Hence, the equation (1.2) is referred to as the noisy sparse phase retrieval model.

Due to the importance of phase retrieval in signal processing field, various algorithms are developed to recover the signal. The algorithms include Semidefinite Programming, Alternate Minimisation, Alternate Projection, Wirtinger Flow etc. In this work, we use thresholded Wirtinger flow as the base algorithm for the retrieval of phase.

Chapter 2

LITERATURE REVIEW

This chapter contains a brief account of the published literature related to the present investigation. The first reconstruction algorithms for phase retrieval was proposed by Gerchberg and Saxton in 1972 [1]. The algorithm involves iterative Fourier transformations between the object and its Fourier domains and application of the measured data or constraints in each domain [7]. It is also referred to as the error-reduction algorithm since the error decreases at each iteration. The algorithm starts from a random initial guess of data point, and is modified by applying a pair of projections. Each iteration requires solving a least-squares problem. When the squared error is zero, the solution is found.

The algorithm often works well in practice, but it depends on apriori information about the signals. The error decreases rapidly for the first few iterations than the later ones. The rapidness of the convergence also depends on the constraints. Also this algorithm gets stuck at stagnation points and fail to converge due to the non-convexity of the problem. There are some computational methods that perform without using a priori information of the signal. This includes semidefinite programming, polarisation, alternating minimization, gradient methods, alternating projection etc.

Convexification methods are studied and introduced to overcome the convergence problem seen in the above algorithms. These methods use the matrix format but replace the non-convex rank constraint by a convex constraint. They even perform well on non-random phase retrieval problems [14]. Unfortunately, this good precision comes with high computational cost. Convexification techniques

are therefore impractical for high dimensional signals.

Recently, fast non-convex algorithms are designed that could guarantee similar reconstruction as the convexified algorithms. Such algorithms necessarily consist of an additional initialisation algorithm that could provide a better starting point close to the solution. This scheme was first used in [13], with an alternating minimisation algorithm. The correctness of this algorithm was proved by Candes et al. [2] with high probability. Waldspurger then proposed phase retrieval by alternating projections which recovers phase from scalar products using random sensing vectors with complex normal distributions in a non convex setting [14]. The problem with the alternating projections was that it was not continuous.

Sparsity of signals is ignored in all these methods. But since sparsity issues are encountered in many science and engineering applications. Besides this, sparsity assumption also has its advantages like reconstruction of the signal with lesser constraints than the standard reconstruction techniques (Nyquist criteria) and easy computation requiring less memory etc.

Various algorithms are modified for including sparsity constraints. Semidefinite programming algorithm can be modified for noiseless sparse phase retrieval and it was found to be effective [6]. An algorithm named SPARse Truncated Amplitude flow (SPARTA) is also used to reconstruct a sparse signal from lesser number of measurements with lost phase which is NP-hard in general and emerges in many science and engineering applications [12]. Nearly minimax convergence rates with Gaussian noise have been established in [4] under sub-gaussian design matrices. But the procedure is not computationally feasible. Other empirical methods for the same include belief propagation and greedy methods [3]. Later on, it was found that any algorithm that could be used to retrieve a signal can be modified to the sparse case by adding a truncating procedure.

For any optimisation problem, gradient descent is the general option pre-

ferred. Based on a simpler gradient descent procedure, an algorithm [6] was proposed for noiseless phase retrieval. It was known as the Wirtinger Flow algorithm due to the use of Wirtinger derivatives (partial derivatives) in the algorithm. An initialisation algorithm based on a spectral method was also added to avoid the problem due to non convexity. It had low computational complexity, as in any gradient descent scheme. The main contribution was that this algorithm allowed the exact retrieval of phase information from a minimal number of random measurements. The sequence of successive iterations converges to the solution at a geometric rate so that the proposed scheme is efficient both in terms of computational and data resources. This algorithm became rather popular and successful. Subsequently, several algorithms were developed based on this. One among them is the median Truncated Wirtinger Flow [11]. It focuses on the situation when the measurements are corrupted by arbitrary outliers.

Another one which is the centre of focus of this project is the thresholded Wirtinger flow algorithm. The gradient descent based method in Wirtinger Flow [6] was modified for noisy conditions using a thresholding technique based on sample mean unlike the median method used above. Minimax optimal rates of convergence for noisy sparse phase retrieval under sub-exponential noise is also established for this algorithm. The sparsity of the signal x and the presence of sub-exponential noise are also dealt in this method. The convergence rates under the ℓ_2 loss, as long as the sensing vectors a_j^*'s are independent standard Gaussian vectors are also studied. This method is adopted for our phase retrieval problem with some modifications.

Chapter 3

METHODOLOGY

This chapter discusses the details of the methodology adopted for the phase retrieval problem in (1.2). There are two main steps for this algorithm. First, the thresholded gradient descent algorithm to sparsify the estimated signal x. Second, the initialisation algorithm to avoid the difficulty due to non convex nature of the problem. The combination of these two steps, introduced in reverse order, is known as the Thresholded Gradient Descent or the Thresholded Wirtinger Flow (TWF) algorithm. The methodology is proposed assuming that 'A' has standard Gaussian entries, though it could also be used even when this assumption doesn't hold.

The general procedure of solving any optimisation problem is to first formulate a loss function or the empirical risk function. In empirical risk function, the true probability distribution of the data that the algorithm will work on is unknown; instead, its performance is measured on the basis of a training data. The loss function is the measure of variance of predicted data from desired or true data. After finding the loss function, we need to find a solution that minimise the function using a suitable algorithm. Here the thresholded gradient descent algorithm is used for this purpose.

Let ℓ(x ; y) be a loss function. Any phase retrieval problem (1.1) can be solved by finding a solution to the following general optimisation function.

$$Minimise \quad f(z) \;=\; \frac{1}{2m}\sum_{j=1}^{m}\ell(y_j,|a_j^*z|^2), \quad z\in C^n \tag{3.1}$$

Minimising non-convex objectives, which may have many stationary points, is known to be NP-hard problem. Using this general concept, we try to formulate the empirical risk function to the phase retrieval problem here.

We have the sensing vectors a_j and the noisy magnitude measurements y_j as in (1.2) for j = 1, ... , m. The aim is to estimate z by minimising the following empirical risk function:

$$f(z) = \frac{1}{4m}\sum_{j=1}^{m}(|a_j^* z|^2 - y_j)^2 \tag{3.2}$$

where $|a_j^* z|^2$ is the predicted output whereas y_j are the observed signal measurements. The least-squares empirical risk is used in (3.2) although we could use any loss function. Least-squares are preferred to the Least Absolute Deviations (LAD) when p and m are proportional, even when the noises are sub exponential. Also close-form gradient methods can be induced from this empirical risk function. The next step is to minimise this function using the thresholded gradient descent algorithm.

3.1 GRADIENT DESCENT METHOD

Gradient Descent is a common method used for minimising optimisation problems. The convergence to minimum can be guaranteed using Taylor's theorem. To find the local minimum of a function using gradient descent, one takes steps proportional to the negative of the gradient (or approximate gradient) of the function at the current point.

Let x^t be the random initial value as shown. The updation rule in general is given by:

$$x^{t+1} = x^t - \eta \nabla f(x^t) \tag{3.3}$$

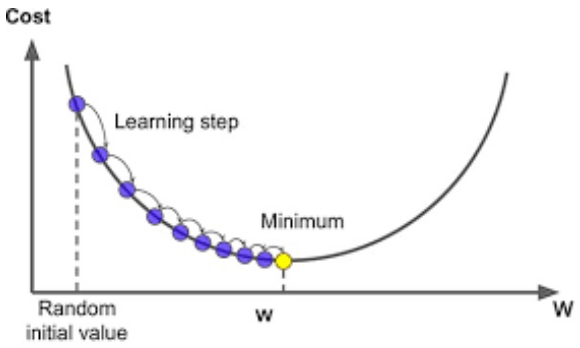

Figure 3.1: The above figure shows the gradient descent in a convex function.

where η is the learning rate and ∇ is the gradient operator or partial derivative of the loss function.

The function is convex and it has only one minimum point. The gradient descent essentially converges to the minimum point after specific number of iterations as shown in the figure 3.1.

For the empirical risk function in (3.2), for any current value of z, one updates the estimator by taking a step along the gradient direction until a stationary point is reached. The gradient of the function is given by:

$$\nabla f(z) = \frac{1}{4m} \sum_{j=1}^{m} (|a_j^* z|^2 - y_j)(a_j^* z) a_j^* \tag{3.4}$$

But the direct application of gradient descent is not ideal for noisy sparse phase retrieval because our function is non convex and may have multiple minima. Also it does not utilise the knowledge that the true signal x is sparse. Also the presence of noise may affect the convergence.

For eliminating this problems, we use certain modifications in the project. They include the utilisation of sparsity in the solutiom methodology and adding an initialisation technique to avoid the problems due to non convexity.

Chapter 4

IMPLEMENTATION OF PROPOSED ALGORITHM

This chapter discusses the stepwise description of the algorithm and its implementation in the Python platform.

The algorithm is implemented in two steps. The first one, is the general gradient descent algorithm with an additional thresholding step (normally called as "Thresholded Wirtinger Flow" itself). The second one is the initialisation algorithm that provide a proper sparse initial value to the first algorithm.

4.1 THRESHOLDED WIRTINGER FLOW

The Wirtinger flow algorithm is named after Wilhelm Wirtinger who introduced partial derivatives (Wirtinger operators) in 1927 in course of his study on complex variables. Wirtinger Flow method uses the gradient descent algorithm for phase retrieval [6]. The thresholding step is introduced in the algorithm to provide a sparse outcome. After applying the gradient descent algorithm, thresholding based on Gaussian model is done so that only the valid values which are above the threshold are taken to the next iteration level.

Let T_τ be any thresholding function

$$T_\tau(x) = 0, \ \ \forall x \in [-\tau, \tau] \ \ and \ \ |T_\tau(x) - x| \leq \tau, \ \ \forall x \in R \tag{4.1}$$

τ is calculated from the given data measurements using Gaussian approximation. The updation procedure is continued for specified number of iterations. In the end,

we obtain a sparsified result which is our true data.

4.1.1 Thresholded Wirtinger Flow Algorithm

Thresholded Wirtinger flow algorithm is implemented by the following steps. While implementing the algorithm on user generated observations, special care should be taken to include the sparsity factor k in it.

- **Input:** Data $\{a_j, y_j\}_{j=1}^{m}$; initial estimator $\hat{x}_{(0)}$; thresholding function T; gradient tuning parameter μ; threshold tuning parameter β; number of iterations N.
- **Output:** Final estimator $\hat{x}$.

1. Initialise n $\leftarrow 0$ $\hat{x}^{(0)} = \hat{x}_{(0)}$

   Repeat:

2. Compute threshold level

$$\tau(x^n) = \sqrt{\frac{\beta log(mp)}{m^2} \sum_{j=1}^{m} (|a_j^* \hat{x}^{(n)}|^2 - y_j)^2 |a_j^* \hat{x}^{(n)}|^2} \tag{4.2}$$

3. Update

$$\hat{x}^{(n+1)} = T_\tau \left(\hat{x}^{(n)} - \frac{\mu}{\phi^2} \nabla f(\hat{x}^{(n)} \right) \tag{4.3}$$

   until n=N

   where $\nabla f(z)$ defined in (3.4);

4. Return $\hat{x} = \hat{x}^{(N)}$

The choice of the threshold level in (4.2) is dependent on the data. The sensing vectors $\{a_j : j = 1, \ldots, m\}$ are independent standard Gaussian vectors. For a

fixed z, if each $(|a_j^* z|^2 - y_j)(a_j^* z)$ is a fixed constant, then the gradient in (3.4) is just a linear combination of Gaussian vectors and hence has i.i.d. Gaussian entries with mean zero and variance $\frac{1}{m^2}\sum_{j=1}^{m}(|a_j^* z|^2 - y_j)^2(a_j^* z)^2$. Therefore, the threshold $\tau(z)$ is simply $\sqrt{\beta log(mp)}$ times the standard deviation of these Gaussian random variables, which is the universal thresholding in the Gaussian sequence model literature [8]. The two tuning parameters μ and β are also treated as absolute constants. They are designed to proper values after trialling various values in the numerical simulations.

4.2 INITIALISATION

If the initialisation $\hat{x}_{(0)}$ is sufficiently accurate, then the sequence will essentially converge toward a solution to the generalised phase problem (1.1). The initial guess $\hat{x}_{(0)}$ is computed by a spectral method. The initial estimator is essential since the empirical function is non convex in nature and it could have multiple local minimisers. So the success of this algorithm depends on the accuracy of the starting point. Also an accurate initialiser can reduce the number of iterations for convergence in the algorithm. An initialisation algorithm is proposed for the same. The basic aim of the algorithm is to compute the estimator of x focusing on a smaller subset of coordinates which are significant in energy. The motivation of the algorithm is similar to that of diagonal thresholding for sparse PCA.

4.2.1 Initialisation Algorithm

- **Input**: Data $\{a_j, y_j\}_{j=1}^{m}$; tuning parameter α.
- **Output**: Initial estimator $\hat{x}_0$.

1. Compute average of measurements:

$$\phi^2 = \frac{1}{m}\sum_{j=1}^{m} y_j \tag{4.4}$$

2. Find

$$I_l = \frac{1}{m}\sum_{j=1}^{m} y_j a_{jl}^2, \quad l = 1, ...p \tag{4.5}$$

3. Select a set of coordinates ℓ that are greater than particular limit

$$\hat{S}_0 = l \in [p] : I_l > \left(1 + \alpha\sqrt{\frac{log(mp)}{m}}\right)\phi^2 \tag{4.6}$$

4. Compute a pxp matrix using these coordinate values

$$W_{\hat{S}_0\hat{S}_0} = \frac{1}{m}\sum_{j=1}^{m} y_j a_{j\hat{S}_0} a^*_{j\hat{S}_0}, \tag{4.7}$$

5. Return

$$\hat{x}_0 = \phi\hat{v}_1 \tag{4.8}$$

where $\hat{v}_1$ is the leading eigen vector of $W_{\hat{S}_0\hat{S}_0}$

The quantity I_l in (4.5) captures the marginal signal strength of the l-th coordinate and $\hat{S}_0$ in (4.6) selects only those coordinates with big marginal signals. This step reduces most of the entries of the matrix to zero and thus sparsify the initial point. A square sparse matrix with dimension p is computed from these signal values. The eigen values of the matrix are computed and the leading eigen vector is then used as the initial value for x. Thus, the initial estimator computed in (4.7) and (4.8) focuses only on a subset of coordinates in $\hat{S}_0$ which makes the computation easier and efficient.

4.3 PROJECT IMPLEMENTATION

The project is implemented in Spyder 3.3 integrated development environment (IDE) for Python language (Python 3). Spyder was chosen because it is one of the best candidates for interactive testing and complex mathematical computations involving large matrices. We used several packages like math, scipy, numpy, matplotlib, hdf5storage for the implementation of the code.

The algorithms described above were stitched together to form a single algorithm. After setting the parameters given in the algorithms, we define a sparse signal x, design matrix A and noise ε to form the input observations y_j as per the equation 1.2. The aim is to retrieve this sparse signal x. After performing the initialisation algorithm, we get a proper starting point for performing the thresholded Wirtinger flow algorithm. Then this initial point is updated using gradient descent method and thresholded as per the algorithm. This is continued for the specified number of iterations, and we retrieve the sparse signal similar to the input sparse signal taken. The relative estimation error is also calculated to check the optimal convergence of the algorithm. Numerical simulations were also done on the real data to identify the dependencies of different parameters on signal recovery. The same procedure was followed for complex data except for some changes in the code, for example replacing transpose with a complex conjugate. We also tried running the algorithm on a public dataset for phase retrieval algorithms "Coherent Inverse Scattering via Transmission Matrices" [16] obtained from http://dsp.rice.edu/research/transmissionmatrices/.

Chapter 5

NUMERICAL SIMULATIONS AND RESULTS

In this chapter, the numerical simulation results on the thresholded Wirtinger flow algorithm are given. The relative estimation error depends on various parameters like thresholding parameter β, the noise-to-signal ratio (NSR) $\sigma/||x||_2$, the sample size m and the sparsity k. We fix the length of the signal $p = 1000$ and the initialisation parameter $\alpha = 0.1$, gradient parameter $\mu = 0.01$ and the number of iterations $T = 1000$ for thresholded Wirtinger flow. The estimator obtained is denoted as $\hat{x} = \hat{x}^{(1000)}$. The support of x is uniformly distributed at random for each fixed k. The non-zero values of x are independent and identically distributed $\sim N(0,1)$. The noise follows the distribution $\varepsilon \sim N(0, \sigma^2 I_m)$, where σ is determined by $||x||_2$ and the NSR $\sigma/||x||_2$ chosen. The design matrix A consists of independent standard Gaussian random variables.

5.1 THRESHOLDING EFFECT

We fix $\alpha = 0.1$, $m = 7000$, $k = 100$, and $\sigma/||x||_2 = 1$. For each $\beta = 0$, 0.25, 0.5, ..., 3, the algorithm is implemented with independently generated A, x, and ε and then the average of the independent relative errors $\min(||\hat{x}-x||_2, ||\hat{x}+x||_2)/||x||_2$ is taken. The relation between the average relative error and the choice of β is plotted in the figure 5.1 below. As the thresholding parameter β increases to 0.249, error first decreases and beyond a point, the error increases again. The above experiments are implemented again with the only difference $\alpha = 0.5$. The performance of the algorithm is very close to the case $\alpha = 0.1$. When the experiment

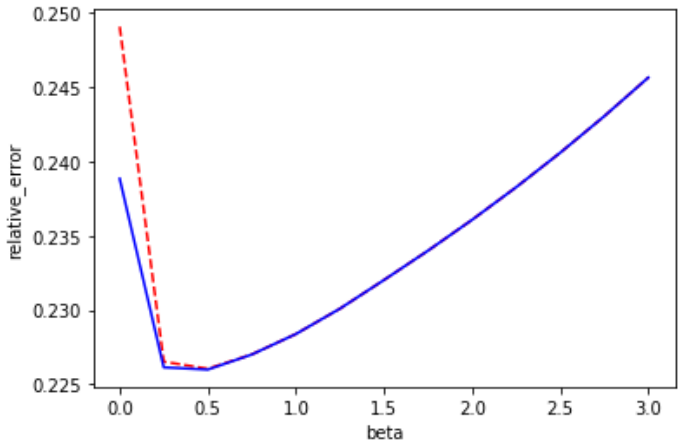

Figure 5.1: The average relative error versus the thresholding parameter β for $T = 1000$. Red curve for $\alpha = 0.1$ and blue for $\alpha = 0.5$.

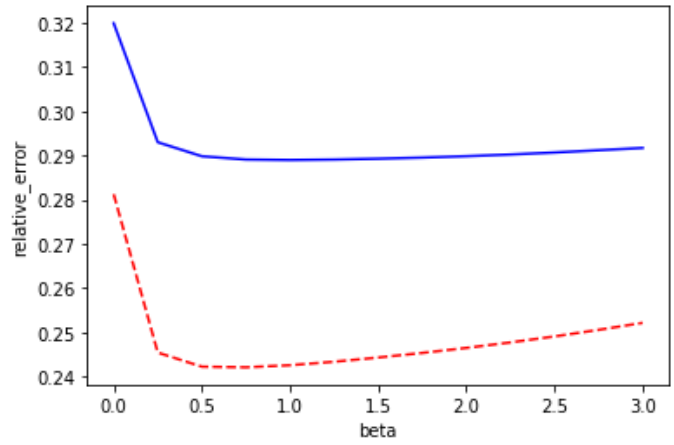

Figure 5.2: The average relative error versus the thresholding parameter β for $T = 100$. Red curve for $\alpha = 0.1$ and blue for $\alpha = 0.5$.

was repeated for $T = 100$, it was observed that the maximum average relative error increased to 0.3. Also a large gap in the performance was observed for the different α. This is illustrated in figure 5.2.

5.2 NOISE EFFECT

We fix $m = 7000$, $k = 100$ and $\beta = 1$. For each NSR $\sigma/||x||_2^2 = 0, 0.1, \ldots, 1$, with A, x, ε generated independently, the average relative error $\min(||\hat{x} - x||_2, ||\hat{x} + x||_2)/||x||_2$ is computed. Figure 5.3 shows the dependency of average relative error

on NSR. As NSR increases from 0 to 1, relative error increase from 0 to 0.27 and then the increase is relatively slow.

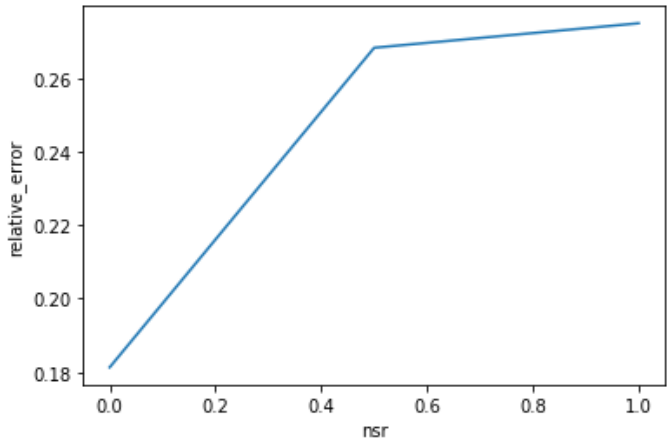

Figure 5.3: The average relative error versus the noise-to-signal ratio $\sigma/||x||_2^2$. Parameters: $p = 1000$, $m = 1000$, $k = 100$, $\beta = 1$, $\alpha = 0.1$, $\mu = 0.01$ and $T = 1000$.

5.3 SAMPLE SIZE EFFECT

We fix $k = 100$, $\sigma/||x||_2^2 = 1$ and $\beta = 1$. For each case of m = 2000, 3000, ...,11000, with A, x, ε generated independently, the average relative error $\min(||\hat{x} - x||_2, ||\hat{x} + x||_2)/||x||_2$ is computed. Figure 5.4 shows how the average relative error depends on the sample size. When sample size increases from 2000 to 10000, the average relative error decreases from 1.00 to 0.3. When the sample size is just twice or thrice the length of p, that is, for m = 2000 and 3000, the relative error is the highest and it leads to poor recovery of the original signal. When m is increased, error reduces to around 0.3. Thresholded Wirtinger flow algorithm gives optimal convergence for noisy sparse phase retrieval only if the sample size is sufficiently large. This is an important conclusion obtained from the experiments performed.

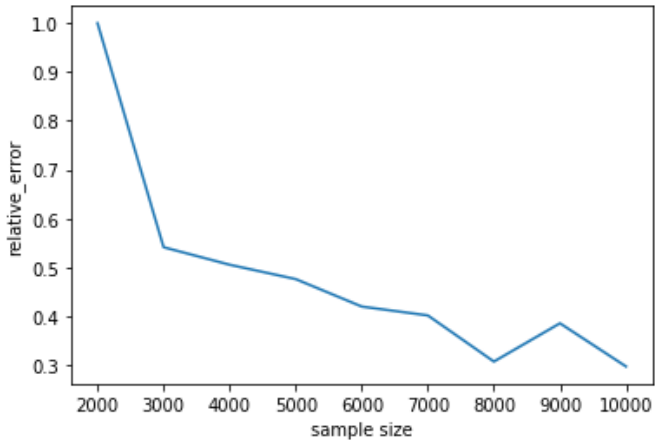

Figure 5.4: The average relative error versus the sample size m. Parameters $p = 1000$, $\sigma/||x||_2^2 = 1$, $k = 100$, $\beta = 1$, $\alpha = 0.1$, $\mu = 0.01$ and $T = 1000$.

5.4 SPARSITY EFFECT

We fix $m = 7000$, $\sigma/||x||_2^2 = 1$ and $\beta = 1$. For each case of sparsity $k = 25$, 50, ..., 200, with A, x, ε generated independently, the average of the relative error $\min(||\hat{x}-x||_2, ||\hat{x}+x||_2)/||x||_2$ is computed. In figure 5.5, the relation between the average relative error and the sparsity k is demonstrated. As sparsity k increases from 25 to 200, the average relative error essentially increases from 0.08 to 0.58. As sparsity or the number of non-zero entries in the signal increases, error also tend to increase steadily.

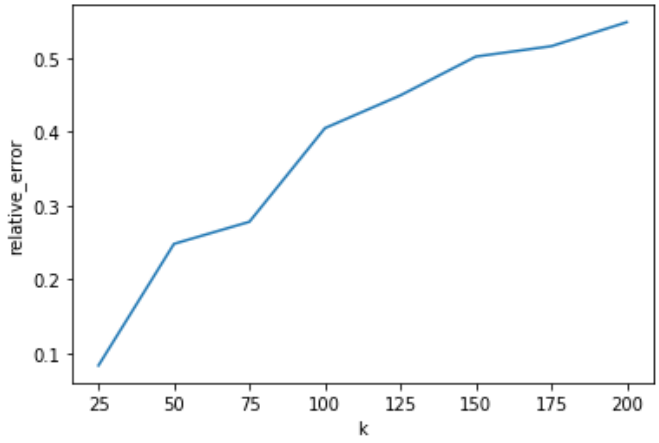

Figure 5.5: The average relative error versus sparsity k. Parameters $p = 1000$, $\sigma/||x||_2^2 = 1$, $\beta = 1$, $\alpha = 0.1$, $\mu = 0.01$ and $T = 1000$.

5.5 SIGNAL RECOVERY

The success rate of recovery is plotted against the number of iterations and the following graph is obtained. Success is computed as the inverse of the average relative error for T= 200, 400, ..., 2000 with the parameters $p = 1000$, $\sigma/||x||_2^2$=1, $k = 100$, β=1, $\alpha = 0.1$, $\mu = 0.01$. From the graph, the signal recovery rate is seen to increase for larger number of iterations. But as the iterations increase beyond a particular limit, the graph goes constant.

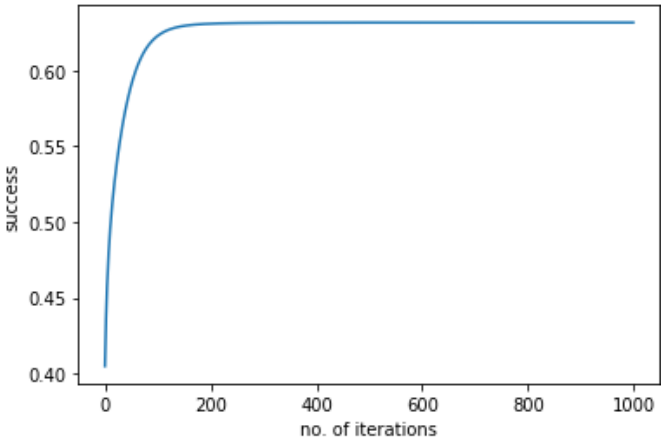

Figure 5.6: The success rate versus number of iterations. Parameters $p = 1000$, $\sigma/||x||_2^2 = 1$, $k = 100$, $\beta = 1$, $\alpha = 0.1$, $\mu = 0.01$ and $T = 1000$.

Other parameters like gradient tuning parameter μ and initialisation parameter α were also analysed and it was found of less significance in signal recovery in the given setup. The algorithm was also extended to complex case and found absolutely working. The convergence rate was found to be bounded by $\frac{3\sigma}{|||x|_2}\sqrt{\frac{klogp}{m}}$. The dataset obtained was not as per the dimensional restrictions of the algorithm and also required high computational capability. The dataset was modified slightly and the algorithm was run partially.

Chapter 6

APPLICATIONS

This chapter discusses various applications where thresholded Wirtinger flow algorithm could be used. Since phase retrieval in signal processing is a common problem, the algorithm is significant. The attractive feature of this algorithm is that it gives optimal convergence even in the noisy conditions.

6.1 COHERENT DIFFRACTIVE IMAGING

The optical detectors like charge coupled device (ccd) cameras, human eye, photosensitive films measure only the magnitude of the diffracted light from an object when illuminated by a monochromatic wave. They cannot record the phase of the image as the electromagnetic field oscillates at rates $\sim$ 1 kHz. The diffraction pattern obtained here is the Fourier transform of the object of interest. This is called Fraunhofer diffraction. The data acquisition here is similar to the phase retrieval problem in 1.2. We can identify 'x' as the object of interest, 'a_j' as the complex sinusoids, and 'y_j' the recorded data. Then we can apply the algorithm as discussed in the previous chapters.

6.2 X-RAY CRYSTALLOGRAPHY

X-ray crystallography is a widely used technique to determine the position of each atom in the molecule. Due to the periodic structure of the crystal, the far-field information contains strong peaks which gives only the intensity. Also short wavelength of X-rays (1 A^o) make it difficult to directly image the molecules.

Here phase retrieval problem arises. The diffraction pattern formed by the scattered X-rays is measured, and the data are processed numerically and solved using the algorithm.

6.3 ASTRONOMY

In optical astronomy, a telescope is used to record the image to obtain the intensity of an object. The atmospheric turbulence limits the resolution for earth based observations compared to the limit set by diffraction $\alpha \sim \lambda/D$, where λ is the wavelength and D is the telescope diameter. So interferometry techniques are used in which the spatial coherence in the far field of a thermal light source is proportional to the Fourier transform of the object intensity. But accurate phase estimate is still hard to obtain. The reconstruction of object from limited phase information might lead to errors. Here we use the algorithm for retrieving phase.

Various algorithms for phase retrieval problem has led to many significant accomplishments, including the discovery of DNA double helical structure from diffraction patterns (1953), and the characterisation of aberrations in the Hubble Space Telescope from the measured point spread functions (PSF).

Chapter 7

CONCLUSION

Thresholded Wirtinger Flow algorithm gives exact recovery of sparse phase under noisy conditions. Using TWF, we also obtain the optimal convergence rates under the Gaussian design in the presence of sub-exponential noise, provided that the sample size is sufficiently large, i.e., $m > 4p$. The optimal rate of convergence for estimating the signal x under the ℓ_2 loss is found to be approximately $\frac{3\sigma}{|||x|]_2}\sqrt{\frac{klogp}{m}}$, where k is the sparsity of x, $||.||_2$ is the Euclidean norm and σ characterises the noise. We validated the recovery of phase over real as well as complex random observations using the algorithm. It was found that the sparsity and the sub-exponential nature of the noise affects the estimation of the signal x. Also the addition of sparse nature helps in the successful reconstruction of the signal. The algorithm also satisfies the sample size requirement. The thresholded gradient descent procedure is thus both rate-optimal and computationally efficient.

REFERENCES

[1] **R. W. Gerchberg and W. O. Saxton**. A practical algorithm for the determination of phase from image and diffraction plane pictures. *Optik*, 35:237 -246, 1972.

[2] **E. J. Candes, Y. C Eldar, T. Strohmer, and V. Voroninski**. Phase retrieval via matrix completion. *SIAM Journal on Imaging Sciences*, 6(1):199-225, 2013.

[3] **Y. Shechtman, A. Beck, and Y. C. Eldar**. GESPAR: Efficient phase retrieval of sparse signals. *IEEE Transactions on Signal Processing*, 62(4):928-938, 2014.

[4] **Y. Chen, Y. Chi, and A. J. Goldsmith**. Exact and stable covariance estimation from quadratic sampling via convex programming. *IEEE Transactions on Information Theory*, 2014.

[5] **X. Li and V. Voroninski**. Sparse signal recovery from quadratic measurements via convex programming. *SIAM Journal on Mathematical Analysis*, 45(5):3019-3033, 2013.

[6] **E. J. Candes , X. Li, and M. Soltanolkotabi**. Phase retrieval via Wirtinger flow: Theory and algorithms. *IEEE Transactions on Information Theory*, To appear, 2014.

[7] **J. R. Fienup**. Phase retrieval algorithms: a comparison. *Applied Optics*, 21(15):2758-2769, 1982.

[8] **I.M. Johnstone**. Gaussian Estimation: Sequence and Wavelet Models, 2013. Available at *http://www-stat.stanford.edu/ imj/.*

[9] **E. J. Candes** and X. Li. Solving quadratic equations via PhaseLift when there are about as many equations as unknowns. *Foundations of Computational Mathematics*, 14(5):1017-1026,2014

[10] **T. Tony Cai, Xiaodong Li, and Zongming Ma**. Optimal Rates of Convergence for Noisy Sparse Phase Retrieval via Thresholded Wirtinger Flow, *University of Pennsylvania* June 11, 2015

[11] **Huishuai Zhang, Yuejie Chi, Yingbin Liang**. Provable Non-convex Phase Retrieval with Outliers: Median Truncated Wirtinger Flow. *Proceedings of The 33rd International Conference on Machine Learning*, 45(5):3019-3033, 2016.

[12] **Gang Wang, Liang Zhang, Georgios B. Giannakis, Mehmet Akcakaya, Jie Chen**. Sparse Phase Retrieval via Truncated Amplitude Flow. *IEEE Transactions on Signal Processing*, vol. 66, no. 2, January 15, 2018.

[13] **Netrapalli, P. Jain, and S. Sanghavi**. Phase retrieval using alternating minimization. In *Advances in Neural Information Processing Systems 26*, pages 1796-2804, 2013.

[14] **Irne Waldspurger**, Phase retrieval with random Gaussian sensing vectors by alternating projections, *https://arxiv.org.math* 2016.

[15] **Vladimir Katkovnik**, Phase retrieval from noisy data based on sparse approximation of object phase and amplitude, *arXiv:1709.01071v1 [cs.NA]*, 4 Sep 2017.

[16] **Christopher A. Metzler, Manoj K. Sharma , Sudarshan Nagesh**, Coherent Inverse Scattering via Transmission Matrices: Efficient Phase Retrieval

Algorithms and a Public Dataset, *International Conference on Computational Photography* 2017.

[17] **Q. Gu, Wang and H Liu**, Low-Rank and Sparse Structure Pursuit via Alternating Minimization, *International Conference on Artificial Intelligence and Statistics, AISTATS*, May 2016.