

Large-Scale Phase Retrieval via Stochastic Reweighted Amplitude Flow

Zhuolei Xiao^{1,2}, Yerong Zhang^{1*}, and Jie Yang³

¹ College of Electronic and Optical Engineering & College of Microelectronics, Nanjing University of Posts and Telecommunications, Nanjing 210023, China

[e-mail: sky_south@163.com (Z.X.); zhangyr@njupt.edu.cn (Y.Z.)]

² School of Computer and Information Engineering, Fuyang Normal University, Fuyang 236037, China

³ College of Telecommunication and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210023, China

[e-mail: jyang@njupt.edu.cn]

*Corresponding author: Yerong Zhang

*Received June 19, 2020; revised September 25, 2020; accepted November 15, 2020;
published November 30, 2020*

Abstract

Phase retrieval, recovering a signal from phaseless measurements, is generally considered to be an NP-hard problem. This paper adopts an amplitude-based nonconvex optimization cost function to develop a new stochastic gradient algorithm, named stochastic reweighted phase retrieval (SRPR). SRPR is a stochastic gradient iteration algorithm, which runs in two stages: First, we use a truncated sample stochastic variance reduction algorithm to initialize the objective function. The second stage is the gradient refinement stage, which uses continuous updating of the amplitude-based stochastic weighted gradient algorithm to improve the initial estimate. Because of the stochastic method, each iteration of the two stages of SRPR involves only one equation. Therefore, SRPR is simple, scalable, and fast. Compared with the state-of-the-art phase retrieval algorithm, simulation results show that SRPR has a faster convergence speed and fewer magnitude-only measurements required to reconstruct the signal, under the real- or complex- cases.

Keywords: Phase retrieval, Nonconvex optimization, sample stochastic variance reduction, stochastic reweighted gradient iteration

This research was supported by the Horizontal Foundation of Fuyang Normal University (XDXH201741), Anhui Provincial Major Projects of Scientific Research of Universities (KJ2018ZD036), Natural Science Research Key Project of Fuyang Normal University (2019FSKJ06ZD), Fuyang Normal University Young Talents Key Project (rcxm202006, rcxm202004).

1. Introduction

Phase retrieval, recovering a signal from given intensity or magnitude only measurements, is reflected in various fields of science and engineering, such as astronomy [1], electron microscopy [2], optics [3], and X-ray crystallography [4]. Generally, in optical imaging, due to optical detectors' physical limitations [5], optical detection equipment such as photosensitive film, charge-coupled device (CCD) camera, or human eye can only record the intensity but not the phase of a light wave. However, for an image, the structural information carried by its phase is far greater than that taken by its intensity. Hence it is challenging to reconstruct the signal from the intensity- or magnitude-only measurements are of great significance.

Mathematically speaking, Phase Retrieval is the problem of recovering N -dimensional real or complex signal \mathbf{x} from M linear transformation (or Fourier transform) measurements

$$y_i = |\langle \mathbf{a}_i, \mathbf{x} \rangle|^2, \quad 1 \leq i \leq m \quad (1)$$

where the given measurements vector $\mathbf{y} := [y_1 \cdots y_m]^T$ and the sensing/feature vector $\mathbf{a}_i \in \mathbb{R}^n$ or \mathbb{C}^n are known, and the signal vector $\mathbf{x} \in \mathbb{R}^n$ or \mathbb{C}^n is the wanted unknown vector. It is worth noting that y_i denotes the intensity or the squared modulus, whereas $\psi_i := \sqrt{y_i}$ denotes the magnitude vector or the amplitude vector. Phrased differently, we want to uniquely recover \mathbf{x} (up to the global sign) from the system of quadratic equations (1), or retrieve the sign or phase of $\{\langle \mathbf{a}_i, \mathbf{x} \rangle\}_{i=1}^m$ under the setting of real or complex values. Due to the loss of sign or phase information, to recover \mathbf{x} exactly, the number of measurements m and the dimension n of the wanted original signal should satisfy $m \geq 2n - 1$ or $m \geq 4n - 4$ when $\mathbf{a}_i, \mathbf{x} \in \mathbb{R}^n$ or \mathbb{C}^n , respectively, while $m = 2n - 1$ is also necessary [6], [7]. Nevertheless, the quadratic equations in (1) is still an *NP-hard* problem, and its solution is always challenging.

1.1 Related Works

The measurements are generally corrupted with noise. When these additional noises follow the Gaussian distribution, we consider using the least-squares method (which is consistent with the maximum likelihood estimate of the original signal), the problem of solving systems of quadratic equations in (1) can be rewritten as the following three different empirical loss functions: the *intensity-based* empirical loss [8], [9]

$$\underset{\mathbf{z} \in \mathbb{R}^n / \mathbb{C}^n}{\text{minimize}} \frac{1}{2m} \sum_{i=1}^m (y_i - |\langle \mathbf{a}_i, \mathbf{z} \rangle|^2)^2, \quad (2)$$

and its related Poisson likelihood [10]

$$\underset{\mathbf{z} \in \mathbb{R}^n / \mathbb{C}^n}{\text{minimize}} \frac{1}{2m} \sum_{i=1}^m (|\langle \mathbf{a}_i, \mathbf{z} \rangle|^2 - y_i \log(|\langle \mathbf{a}_i, \mathbf{z} \rangle|^2)), \quad (3)$$

and the *amplitude-based* empirical loss [11]

$$\underset{\mathbf{z} \in \mathbb{R}^n / \mathbb{C}^n}{\text{minimize}} \frac{1}{2m} \sum_{i=1}^m (\psi_i - |\langle \mathbf{a}_i, \mathbf{z} \rangle|)^2 \quad (4)$$

Obviously, due to the quadratic terms in (2) and (3), or the modulus in (4), these empirical loss functions are non-convex, so there will be many stationary points when minimizing the objective function, so this problem is *NP-hard*. In the 1-dimensional (1D) phase retrieval, there is no uniqueness based on spectral factorization, i.e., even with trivial ambiguities (global phase shift, conjugate inversion, and spatial shift) [12], there are multiple solutions with the same magnitude. There are several approaches to overcome this ill-posed problem. We can add constraints to the signal to be sought, such as sparsity or non-negativity [13]–[16]; Another solution is to introduce redundancy for the measurements, including short-time Fourier transform [17], [18], coded diffraction patterns using random masks and structured illuminations [19], or random measurement assumption [10], [11], [20]–[25]. In this paper, we will adopt this assumption $\{\psi_i\}$ with random Gaussian $\{\mathbf{a}_i\}$ designs, which is independently and identically distributed (i.i.d.). It's $\mathbf{a}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_n)$ for the real-valued Gaussian setting, and $\mathbf{a}_i \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_n) := \mathcal{N}(\mathbf{0}, \mathbf{I}_n/2) + j\mathcal{N}(\mathbf{0}, \mathbf{I}_n/2)$ for the complex-valued one.

There are two ways to solve such problems: convex and nonconvex ones. Based on the so-called matrix-lifting technique, the convex ones express solving semidefinite programming (SDP), such as PhaseLift [22], PhaseCut [26], and CoRk [27]. Other convex approaches rely on convex relaxation to reformulate the problem as sparse recovery, solving a basis pursuit problem in the dual domain, such as PhaseMax [28], SPARTA [29]. On the other hand, the nonconvex approaches include alternating projection (e.g. Gerchberg-Saxton, Fineup method) [30], [31], AltMinPhase [21], Wirtinger Flow and its variants (WF/TWF/RWF) [10], [20], [32], trust-region [33], proximal linear algorithms [33], [34], (stochastic) truncated amplitude flow variants (TAF/RAF/STAF) [11], [23]–[25]. These non-convex methods run directly on vector-optimized variables, so they have significant computational advantages over convex methods based on matrix lifting techniques.

However, although non-convex optimization can obtain the best statistical guarantee in no-noise or noisy environment, even if random Gaussian design is introduced, non-convex optimization will still have several shortcomings: First, there will be a lot of stationary points in the non-convex optimization process, which will bring a lot of trouble to our calculation process, so that the optimization often falls into the local optimal, Our task is to develop an algorithm that approximates the global optimal based on existing algorithms; More troublesome is the existence of saddle points, which often makes optimization dead, how to get rid of saddle points is also one of the primary purposes of this paper; It is well known that there is an information theory limit $m = 2n - 1$ or $4n - 4$, hence it will be very challenging to develop a practical algorithm that can achieve perfect recovery and optimal statistical accuracy recovery when the number of measurements is close to the limit of information theory; Finally, the current application and data scale are getting larger and larger, the existing incremental gradient method [35] and stochastic gradient method [23], [36] can no longer meet the requirements of accurate recovery and convergence speed, it is also challenging to increase the convergence speed and reduce the computational complexity of the algorithm in large-scale applications.

1.2 Our Contribution

Inspired by TAF, this paper uses a nonconvex optimization formulation based on amplitude to develop a new stochastic gradient algorithm, referred to *stochastic reweighted phase retrieval* (SRPR). SRPR is an iterative algorithm based on and extends TAF [11] to be applied to large-scale phase retrieval applications such as imaging. There are two stages of the proposed algorithm: The first stage is the initialization stage, which adopts a sample stochastic variance reduction algorithm to initialize the objective function. Like TAF, we use the orthogonal promotion to determine the truncated threshold. In the gradient refinement stage, we use the weighted gradient method to update the estimated initial value, and the stochastic gradient method is used to accelerate the process.

Even in a high-dimensional environment, SRPR can perfectly reconstruct any signal given a minimum number of equations ($m = 2n - 1$) without any additional assumptions. This paper aims to use a stochastic variance reduction gradient (SVRG) variant algorithm to estimate the initialization stage's initial value. Compared with the classic SVRG algorithm [23] and power method using truncated procedure [11], the new algorithm retains a gradient snapshot at each iteration to replace the full gradient. It can significantly reduce the calculation time. In the gradient refinement phase, we use the stochastic gradient method to update the estimated initial value, and the truncated rule is used to eliminate the wrong estimate. By calculating these two stages, even under the condition of approaching the limit of information theory, SRPR can still accurately recover the signal. At the same time, in such relatively large-scale dimensions as image applications, SRPR is superior to existing algorithms in terms of accurate recovery performance and convergence speed, such as (T) AF [11], ITWF [37], (T)WF [8], [10], RAF [24], [25] and STAF [23].

The rest of this article is organized as follows. Section 2 describes and analyzes the amplitude-based loss function, describes the two stages of the SRPR algorithm in detail. Furthermore, simulated tests comparing the SRPR algorithm with (T)WF [8], [10], TAF [11], RAF [24], [25], STAF [23] are presented in Section 3.

Notation: Lower-(upper-) case boldface letters denote vectors (matrices). The symbol \mathcal{T} (\mathcal{H}) denotes transposition (conjugate transposition). The symbol $\lceil \cdot \rceil$ stands for the largest integer no more significant than the given number. If $\mathbf{x} \in \mathbb{R}^n$ is the true solution of (1), then $-\mathbf{x}$ is also. We define Euclidean distance of any estimate $\mathbf{z} \in \mathbb{R}^n$ to the true solution set $\{\pm \mathbf{x}\}$ of (1) as $\text{dist}(\mathbf{z}, \mathbf{x}) := \min \|\mathbf{z} \pm \mathbf{x}\|$.

2. Stochastic Reweighted Phase Retrieval

SRPR is based on TAF [11], and the stochastic gradient descent method is used in the initialization and gradient refinement stages. Therefore, we will detail SRPR from two stages.

For concreteness, we mainly discuss real-valued Gaussian models with a real signal $\mathbf{x} \in \mathbb{R}^n$ and i.i.d. Gaussian random designs $\mathbf{a}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $1 \leq i \leq m$. Nevertheless, SRPR still applies to complex-valued Gaussian models with a complex vector $\mathbf{x} \in \mathbb{C}^n$ and i.i.d. design vectors $\mathbf{a}_i \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_n) := \mathcal{N}(\mathbf{0}, \mathbf{I}_n/2) + j\mathcal{N}(\mathbf{0}, \mathbf{I}_n/2)$. In this paper, using the least-squares criterion, we can convert solving the phaseless quadratic equation in (1) to minimize the amplitude-based empirical loss function

$$\underset{\mathbf{z} \in \mathbb{R}^n}{\text{minimize}} \ell(\mathbf{z}) := \frac{1}{2m} \sum_{i=1}^m (\psi_i - |\langle \mathbf{a}_i, \mathbf{z} \rangle|)^2 \quad (5)$$

2.1 Sample Stochastic Variance Reducing Initialization

In this section, we will elaborate on the initialization stage. Since (5) is non-convex and non-smooth, it makes sense to obtain a good initialization. We developed a new initialization method based on the orthogonality-promoting initialization (OPI) method in TAF. Specifically, since any vectors are approximately orthogonal in pairs in high-dimensional space [11], we adopt

$$\cos^2 \theta_i := \frac{|\langle \mathbf{a}_i, \mathbf{x} \rangle|^2}{\|\mathbf{a}_i\|^2 \|\mathbf{x}\|^2} = \frac{\psi_i^2}{\|\mathbf{a}_i\|^2 \|\mathbf{x}\|^2}, \quad 1 \leq i \leq m \quad (6)$$

to represent the geometric relationship between \mathbf{a}_i and \mathbf{x} , where θ_i represents the angle between \mathbf{a}_i and \mathbf{x} . The larger θ_i , the smaller $\cos^2 \theta_i$, indicating the more orthogonal between \mathbf{a}_i and \mathbf{x} . Based on this feature, we can approximate \mathbf{x} by finding the most orthogonal vector \mathbf{z} to $\{\mathbf{a}_i\}_{i \in \mathcal{J}_0}$, where the index set \mathcal{J}_0 includes indexes of \mathbf{a}_i that make $\cos^2 \theta_i$ the smallest.

Without loss of generality, assuming $\|\mathbf{x}\| = 1$, we find the estimated vector $\tilde{\mathbf{z}}_0$ through

$$\tilde{\mathbf{z}}_0 := \arg \min_{\|\mathbf{z}\|=1} \mathbf{z}^T \mathbf{Y} \mathbf{z} := \mathbf{z}^T \left(\frac{1}{|\mathcal{J}_0|} \sum_{i \in \mathcal{J}_0} \frac{\mathbf{a}_i \mathbf{a}_i^T}{\|\mathbf{a}_i\|^2} \right) \mathbf{z} \quad (7)$$

where $\mathbf{Y} := \frac{1}{|\mathcal{J}_0|} \sum_{i \in \mathcal{J}_0} \frac{\mathbf{a}_i \mathbf{a}_i^T}{\|\mathbf{a}_i\|^2}$, $|\mathcal{J}_0|$ is on the order of n . On the other hand, when $\|\mathbf{x}\| \neq 1$, we can scale $\tilde{\mathbf{z}}_0$ by the norm estimation of \mathbf{x} to obtain the current estimate $\mathbf{z}_0 := \sqrt{\frac{1}{m} \sum_{i=1}^m y_i} \tilde{\mathbf{z}}_0$ [11].

Solving (7) is equivalent to solving the eigenvector corresponding to the smallest eigenvalue of \mathbf{Y} , which requires eigendecomposition or matrix inversion, and the computational complexity is $\mathcal{O}(n^3)$. Define $\bar{\mathcal{J}}_0$ as the complement of \mathcal{J}_0 in $[m]$. In order to reduce the computational complexity, according to SLLN, for the i.i.d. standard Gaussian random matrix $\{\mathbf{a}_i \sim \mathcal{N}(0, I_n)\}_{i=1}^m$, there are

$$\frac{1}{m} \sum_{i=1}^m \frac{\mathbf{a}_i \mathbf{a}_i^T}{\|\mathbf{a}_i\|^2} \approx \mathbb{E} \left[\frac{\mathbf{a}_i \mathbf{a}_i^T}{\|\mathbf{a}_i\|^2} \right] = \frac{1}{n} I_n, \quad (8)$$

where $\mathbb{E}[\cdot]$ expresses expectations. Thus

$$\sum_{i \in \mathcal{J}_0} \frac{\mathbf{a}_i \mathbf{a}_i^T}{\|\mathbf{a}_i\|^2} = \sum_{i \in [m]} \frac{\mathbf{a}_i \mathbf{a}_i^T}{\|\mathbf{a}_i\|^2} - \sum_{i \in \bar{\mathcal{J}}_0} \frac{\mathbf{a}_i \mathbf{a}_i^T}{\|\mathbf{a}_i\|^2} \cong \frac{m}{n} I_n - \sum_{i \in \bar{\mathcal{J}}_0} \frac{\mathbf{a}_i \mathbf{a}_i^T}{\|\mathbf{a}_i\|^2}. \quad (9)$$

Defining $\mathbf{B} = [\mathbf{a}_i / \|\mathbf{a}_i\|]_{i \in \bar{\mathcal{J}}_0}$, the eigenvector corresponding to the minimum eigenvalue of solving \mathbf{Y} in (7) can be reduced to the eigenvector corresponding to the largest eigenvalue of solving $\bar{\mathbf{Y}} = \mathbf{B} \mathbf{B}^T$, i.e.

$$\tilde{\mathbf{z}}_0 := \arg \max_{\|\mathbf{z}\|=1} \mathbf{z}^T \bar{\mathbf{Y}} \mathbf{z} := \mathbf{z}^T \left(\frac{1}{|\bar{\mathcal{J}}_0|} \sum_{i \in \bar{\mathcal{J}}_0} \frac{\mathbf{a}_i \mathbf{a}_i^T}{\|\mathbf{a}_i\|^2} \right) \mathbf{z}, \quad (10)$$

which is a typical principal component analysis (PCA) problem. On the one hand, WF/TWF/TAF [8], [10], [11] uses the power method to solve this problem. The computational complexity required by the power method to obtain ϵ -accurate solutions is $\mathcal{O}\left(\frac{1}{\delta} n |\bar{\mathcal{J}}_0| \log(1/\epsilon)\right)$, where the eigengap $\delta > 0$ is defined as the difference between the largest and second-largest eigenvalues [40]. However, the power method is not suitable for relatively large dimensional data such as image applications, especially applications with small eigengaps. On the other hand, some stochastic algorithms are used to solve (10). STAF uses variance-reducing orthogonality-promoting initialization (VR-OPI) algorithm [11], which is a stochastic variance reduction algorithm (SVRG) [38]. It consists of a series of epochs, which contains a large number of iterative updates. VR-OPI scans the entire dataset every time the epoch starts and calculates a snapshot of the full gradient. And for each iteration in one epoch, the algorithm randomly selects an instance from $|\bar{\mathcal{J}}_0|$ to calculate the gradient, and combines the gradient of the outer loop epochs and the inner loop iterations to reduce the variance of the random gradient and achieve effective convergence [39]. However, if there are too many instances of the data set and the data dimension is very large, since the calculation of the full gradient in each epoch requires the scan the entire dataset, the snapshot of the full gradient results in a large number of gradient calculations, which is very time-consuming. If full-gradient snapshots can be avoided in each epoch, VR-OPI [11] can be significantly accelerated, which will bring better convergence performance.

This paper uses a new acceleration algorithm sampleVR [40] to solve the orthogonal acceleration initialization problem in (10), we named it *sample stochastic variance reducing orthogonality-promoting initialization* (SSVR-OPI), summarized in Algorithm 1. Specifically, SSVR-OPI iteratively updates two sets of loops. Each execution of the inner loop is called an iteration, and each execution of the outer loop is called an epoch. SSVR-OPI has a total of S epochs, and each epoch contains T iterations. The core of the algorithm is to estimate the snapshot of the full gradient $\frac{1}{|\bar{\mathcal{J}}_0|} \sum_{j=1}^{|\bar{\mathcal{J}}_0|} \mathbf{a}_{i_j} \mathbf{a}_{i_j}^T \tilde{\boldsymbol{\omega}}_{s+1}$ through the stochastic gradient $\mathbf{g} = \frac{1}{k} \sum_{j=1}^k \mathbf{a}_{i_j} \mathbf{a}_{i_j}^T \tilde{\boldsymbol{\omega}}_{s+1}$ in step 7, which replaces the snapshot of the full gradient in each epoch in the VR-OPI [11] algorithm, thereby reducing the computational complexity. The stochastic gradient \mathbf{g} is obtained by calculating k ($k \ll |\bar{\mathcal{J}}_0|$) sample instances from the inner loop, representing the average value of the stochastic gradient of k instances, and uses it as the progressive direction of the next outer loop used in Step 4.

We can rewrite (10) as

$$\mathbf{F}(\boldsymbol{\omega}) = \boldsymbol{\omega}^T \bar{\mathbf{Y}} \boldsymbol{\omega} = \frac{1}{|\bar{\mathcal{J}}_0|} \sum_{i \in \bar{\mathcal{J}}_0} f_i(\boldsymbol{\omega}), \quad (11)$$

where $f_i(\boldsymbol{\omega}) = \boldsymbol{\omega}^T \mathbf{a}_{i_t} \mathbf{a}_{i_t}^T \boldsymbol{\omega}$ is L -Lipshitz continuous, i.e., for all $\boldsymbol{\omega}_i$ and $\boldsymbol{\omega}_j$, there is $\|f_i(\boldsymbol{\omega}_i) - f_i(\boldsymbol{\omega}_j)\| \leq L \|\boldsymbol{\omega}_i - \boldsymbol{\omega}_j\|$. The following Proposition 1 [39, Theorem2] shows that SSVR-OPI causes the optimization goal to converge at a linear rate, up to a constant neighborhood of the optimal, proven in [40]. Since SSVR-OPI does not calculate the full

gradient in each epoch, but only calculates the average value of the stochastic gradient of k instances, it is significantly better than other stochastic algorithms in terms of computational complexity. These advantages can be found in the simulation test in Section III.

Proposition 1 [40]: Let k represents the number of sampled instances for the s th epoch. With any $\mu > 0$, when $\delta = \frac{1+4L\mu|\bar{\mathcal{J}}_0|\eta^2}{\mu|\bar{\mathcal{J}}_0|\eta(1-2\eta L-\frac{\eta}{\mu})} < 1$ holds, $F(\boldsymbol{\omega})$ converges as

$$\mathbb{E}[F(\tilde{\boldsymbol{\omega}}_{s+1}) - F(\tilde{\boldsymbol{\omega}}_*)] \leq \delta \mathbb{E}[F(\tilde{\boldsymbol{\omega}}_s) - F(\tilde{\boldsymbol{\omega}}_*)] + \mathcal{O}\left(\left(\frac{|\bar{\mathcal{J}}_0| - k}{|\bar{\mathcal{J}}_0|}\right)^2\right). \quad (12)$$

Although the full gradient estimate \mathbf{g} is an unbiased estimate, there will still be a variance from the proper full gradient, especially when the estimated value is close to the optimum. We can improve this by increasing the number k of sampling instances. An increase in k means that there will be an additional computational cost when estimating the full gradient \mathbf{g} . In contrast, a decrease in k means that the variance between the full gradient and the estimate increases. Here we need to effectively balance the value of k between computational consumption and variance.

Let

$$\mathcal{G} = \frac{1}{k} \sum_{j=1}^k \|\nabla f_i(\tilde{\boldsymbol{\omega}}_{s+1})\| = \frac{1}{k} \sum_{j=1}^k \|\mathbf{a}_{i_j} \mathbf{a}_{i_j}^T \tilde{\boldsymbol{\omega}}_{s+1}\|, \quad (13)$$

and its expectation is $\mathbb{E}(\mathcal{G}) = \frac{1}{|\bar{\mathcal{J}}_0|} \sum_{j=1}^{|\bar{\mathcal{J}}_0|} \|\nabla f_i(\tilde{\boldsymbol{\omega}}_{s+1})\|$. $f_i(\boldsymbol{\omega})$ is L -Lipchitz continuous, that is, $\|\nabla f_i(\boldsymbol{\omega})\| \leq L$. According to Hoeffding's Inequality, we have

$$\alpha = P(|\mathcal{G} - \mathbb{E}(\mathcal{G})| \geq t) \leq 2e^{\frac{-2k^2t^2}{\sum_{i=1}^k (L-0)^2}} = 2e^{\frac{-2k^2t^2}{kL^2}} = 2e^{\frac{-2kt^2}{L^2}}, \quad (14)$$

α denotes the significance level, $1 - \alpha$ is the confidence interval of $[-t, t]$, such k should satisfy

$$k \leq -L^2 \frac{\log \frac{\alpha}{2}}{2t^2} = -\frac{\log \frac{\alpha}{2}}{2t^2/L^2} = -\frac{\log \frac{\alpha}{2}}{\epsilon}, \quad (15)$$

where $\epsilon = 2t^2/L^2$ is a positive real number. In this way, the value of k will be determined by ϵ . With the increase of iterations, the decrease of ϵ causes a continuous increase of k . The variance of the estimated value continues to decrease, to achieve the purpose of significant convergence.

Algorithm 1: Sample Stochastic Variance Reducing Orthogonality-promoting Initialization.

1: **Input:** Data matrix $\mathbf{B} = \{\mathbf{a}_i / \|\mathbf{a}_i\|\}_{i \in \bar{\mathcal{J}}_0}$, truncated thresholds $|\bar{\mathcal{J}}_0| = \lceil m/6 \rceil$, step size $\eta = 0.5$, $\alpha = 0.01$, a positive number ϵ , as well as the number of epochs $S = 100$, and the number of iterations $T = |\bar{\mathcal{J}}_0|$ (by default).

2: **Initialize** the snap of the gradient $\mathbf{g} = 0$, and an initial vector $\tilde{\boldsymbol{\omega}}_0 = 0$.

3: **For** $s = 0$ **to** $S - 1$ **do**


```

 $\omega_0 = \tilde{\omega}_s$ 
4: For  $t = 0$  to  $T - 1$  do
    Pick an instance  $i_t \in \bar{\mathcal{I}}_0$  uniformly at random

     $\mathbf{r}_t = \mathbf{a}_{i_t} \mathbf{a}_{i_t}^T \omega_t - \mathbf{a}_{i_t} \mathbf{a}_{i_t}^T \tilde{\omega}_s + \mathbf{g}$ 

     $\omega_{t+1} = \omega_t - \eta \mathbf{r}_t$ 
     $\omega_{t+1} = \omega_{t+1} / \|\omega_{t+1}\|$ 
End For
5:  $\tilde{\omega}_{s+1} = \omega_i$ ,  $i$  is a random integer in  $[0, T)$ 
6:  $k = -\frac{s \log \frac{\alpha}{2}}{\epsilon}$ 
7:  $\mathbf{g} = \frac{1}{k} \sum_{j=1}^k \mathbf{a}_{i_j} \mathbf{a}_{i_j}^T \tilde{\omega}_{s+1}$ 
8: End For
9: Output:  $\tilde{\mathbf{z}}_0 = \tilde{\omega}_s$ .

```

2.2 Stochastic Reweighted Gradient Stage

The amplitude-based empirical loss function

$$\ell(\mathbf{z}) = \frac{1}{2m} \sum_{i=1}^m \ell_i(\mathbf{z}) = \frac{1}{2m} \sum_{i=1}^m (\psi_i - |\mathbf{a}_i^T \mathbf{z}|)^2 \quad (16)$$

in (5) is a non-convex non-smooth function, and its gradient is

$$\nabla \ell(\mathbf{z}) = \frac{1}{m} \sum_{i \in m} \nabla \ell_i(\mathbf{z}) = \frac{1}{m} \sum_{i \in m} \left(\mathbf{a}_i^T \mathbf{z} - \psi_i \frac{\mathbf{a}_i^T \mathbf{z}}{|\mathbf{a}_i^T \mathbf{z}|} \right) \mathbf{a}_i. \quad (17)$$

Since $\frac{\mathbf{a}_i^T \mathbf{z}}{|\mathbf{a}_i^T \mathbf{z}|}$ introduce a bias in the update direction, such as $\frac{\mathbf{a}_i^T \mathbf{z}_t}{|\mathbf{a}_i^T \mathbf{z}_t|} \neq \frac{\mathbf{a}_i^T \mathbf{x}}{|\mathbf{a}_i^T \mathbf{x}|}$, which may cause the Wirtinger derivative to be too large, resulting in a "bad/misleading" search direction. These gradients may cause \mathbf{z} to move in the opposite direction to \mathbf{x} , especially when approaching the information theory limit $m = 2n - 1$ or $m = 4n - 4$.

There are many ways to solve this problem. TAF [11] introduces a truncated rule to detect and remove these "bad" gradient components to ensure the right search direction. However, this method may cause too many gradient components to be lost or retain defective gradient components, thereby affecting the search direction. STAF [41] goes a step further. It refines the initial estimation by the stochastic truncated gradient iteration. Each iteration evaluates the current generalized gradient and randomly selects from the gradient components or with a given probability. This truncated rule is very likely to reject those "bad" gradient component. RAF [24], [25] takes a different approach and adopts the (timely) adaptive weighted gradient

method by adding different weights ω_i to each gradient vector $\nabla l_i(\mathbf{z})$ of the current point \mathbf{z}_t , to find a suitable search direction for the current iteration process, i.e.

$$\mathbf{z}_{t+1} = \mathbf{z}_t - \mu \nabla \ell(\mathbf{z}) = \mathbf{z}_t - \frac{\mu}{m} \sum_{i \in m} \omega_i \nabla l_i(\mathbf{z}). \quad (18)$$

The weights ω_i corresponding to different gradient components $\nabla l_i(\mathbf{z})$ are closely related to the truncated rule $\mathcal{L} := \left\{1 \leq i \leq m: \frac{|\mathbf{a}_i^T \mathbf{z}|}{|\mathbf{a}_i^T \mathbf{x}|} \geq \frac{1}{1+\gamma}\right\}$ in TAF. Studies in [11] have shown that the gradient component corresponding to a large $\frac{|\mathbf{a}_i^T \mathbf{z}|}{|\mathbf{a}_i^T \mathbf{x}|}$ will have a great probability of pointing in the direction of the true value \mathbf{x} . If $\frac{|\mathbf{a}_i^T \mathbf{z}|}{|\mathbf{a}_i^T \mathbf{x}|}$ is regarded as the confidence parameter α of the corresponding gradient component, then according to the difference of the confidence parameter α , the corresponding gradient components can be distinguished, so as to determine the contribution of these gradient components to the overall search direction.

The method of TAF/STAF is to discard the gradient components corresponding to the relatively small α through the truncated rule. RAF adds weights to all gradient components. The gradient components with large confidence parameters are added with large weights, and small weights are added to those "bad" gradient components. Based on the truncated rule, the weight corresponding to the gradient component ω_i is

$$\omega_i := \frac{1}{1 + \beta_i / (|\mathbf{a}_i^* \mathbf{z}| / |\mathbf{a}_i^* \mathbf{x}|)}, \quad 1 \leq i \leq m \quad (19)$$

where $\{\beta_i\}_{1 \leq i \leq m}$ are pre-selected parameters. In this way, compared with TAF/STAF, adding the weight ω_i not only strengthens the importance of good gradient components, but also retains the information of weak gradient components.

In large-scale applications, due to the high data dimension (more samples), the time cost of passes over the entire data by the classic gradient descent (GD) method is very high. Hence we adopt the stochastic gradient descent (SGD) method with fast convergence speed and low calculation complexity to refine the gradient. On the one hand, the number of measurements required for phase retrieval to achieve accurate recovery has not reached the information theory limit $m = 2n - 1$. TWF needs a ϵ -accurate solution to obtain $m \geq 4.5n$ noise-free measurement value, while TAF is about $3n$. In contrast, STAF, which also adopts a stochastic algorithm in the gradient stage, only needs about $2.3n$ measurement value.

On the other hand, as can be seen from [33], the phase retrieval problem is essentially a non-convex optimization problem. Therefore, there are many saddle points in its search area, and the gradient iteration will most likely fall into the saddle point. SGD method can automatically escape from the saddle point and the poor local best and converge globally to a local minimum with strong generalization.

Therefore, to better apply the RAF-based gradient optimization method to large-scale data, we introduce the gradient method into the weighted gradient iteration and refine the initial estimate \mathbf{z}_0 through this *stochastic reweighted gradient* iteration method to accelerate the gradient iteration stage. Specifically, SRPR uses the following *stochastic reweighted gradient* iteration to update continuously \mathbf{z}

$$\mathbf{z}_{t+1} = \mathbf{z}_t - \mu \omega_i \nabla \ell_{k_t}(\mathbf{z}_t) = \mathbf{z}_t - \mu \omega_i \left(\mathbf{a}_{k_t}^T \mathbf{z}_t - \psi_i \frac{\mathbf{a}_{k_t}^T \mathbf{z}_t}{|\mathbf{a}_{k_t}^T \mathbf{z}_t|} \right) \mathbf{a}_{k_t}, \quad (20)$$

where the step size μ is set to a constant. Each iteration of SRPR selects the current gradient component function $\nabla \ell_{k_t}(\mathbf{z}_t)$ by randomly sampling the index k_t , where t is the current iteration number, and the sampling method of the index k_t can be random sampling from $\{1, 2, \dots, m\}$, or sampling with a given probability. After obtaining the gradient component $\nabla \ell_{k_t}(\mathbf{z}_t)$, we use (19) to evaluate the additional weight ω_i . The proposed SRPR algorithm is expressed in Algorithm 2.

It is worth emphasizing that this stochastic reweighted gradient method is very effective. If the randomly selected gradient component is reliable, its additional weight will be higher, which will make a more outstanding contribution to the search direction and iterative process. Conversely, if it is an unreliable gradient component, its additional weight will be minimal, resulting in almost no refinement in the current iteration. Moreover, if the current iteration point \mathbf{z}_t satisfies $\mathbf{a}_i^T \mathbf{z}_t = 0$, that is, there is a nonsmooth item, its weight is set to $\omega_i = 0$ to eliminate the nonsmooth item and prevent it from affecting the search direction and the iteration process. The specific experimental results will be presented in the experimental part of Section 3.

Algorithm 2: Stochastic Reweighted Phase Retrieval (SRPR)

- 1: **Input:** Data $\{(\mathbf{a}_i; \psi_i)\}_{1 \leq i \leq m}$; step size $\mu = 2$ or 6 for real or complex Gaussian models; weighting parameters $\beta_i = 10$ or 5 for real or complex Gaussian models; the maximum number of iterations T .
 - 2: **Initialize** $\mathbf{z}_0 := (\sum_{i=1}^m \psi_i^2 / m) \tilde{\mathbf{z}}_0$, where $\tilde{\mathbf{z}}_0$ is the leading eigenvector of $\bar{\mathbf{Y}} = \frac{1}{|\bar{\mathcal{J}}_0|} \sum_{i \in \bar{\mathcal{J}}_0} \frac{\mathbf{a}_i \mathbf{a}_i^T}{\|\mathbf{a}_i\|^2}$ obtained via Algorithm 1.
 - 3: **For** $t = 0$ **to** $T - 1$ **do**
 Choose k_t uniformly at random from $\{1, 2, \dots, m\}$
 - 4: $\mathbf{z}_{t+1} = \mathbf{z}_t - \mu \omega_i \nabla \ell_{k_t}(\mathbf{z}_t) = \mathbf{z}_t - \mu \omega_i \left(\mathbf{a}_{k_t}^T \mathbf{z}_t - \psi_i \frac{\mathbf{a}_{k_t}^T \mathbf{z}_t}{|\mathbf{a}_{k_t}^T \mathbf{z}_t|} \right) \mathbf{a}_{k_t}$
 where $w_i := \frac{1}{1 + \beta_i / (|\mathbf{a}_i^* \mathbf{z}| / |\mathbf{a}_i^* \mathbf{x}|)}$, $1 \leq i \leq m$.
 - 5: **End For**
 - 6: **Output:** \mathbf{z}_T
-

3. Simulation Results

This section presents the evaluation of the performance of the proposed algorithm SRPR relative to TWF [8], [10], TAF [11], RAF [24], [25], and STAF [23]. To be fair, the parameters in each algorithm have adopted their suggested values. We use the relative error $:= \text{dist}(\mathbf{z}, \mathbf{x}) / \|\mathbf{x}\|_2$ as the performance metric of the algorithm, where $\text{dist}(\mathbf{z}, \mathbf{x}) := \min \|\mathbf{z} \pm \mathbf{x}\|$ is the Euclidean distance modular constant of the estimation $\mathbf{z} \in \mathbb{R}^n$ to the true solution $\{\pm \mathbf{x}\}$. All simulation tests are performed 100 independent Monte Carlo trials. In each trial, the algorithm runs 100 initialization iterations, while the number of gradient refinement iterations are 1000. It is worth emphasizing that each iteration of stochastic algorithms such as STAF and SRPR

is equivalent to m stochastic iterations over the entire data. A success is declared in a trial when the returned estimate attains a relative error less than 10^{-5} .

The first experimental comparison is in the initialization stage, including the Power Method in TAF, the VR-OPI algorithm in STAF, and the SSVR-OPI algorithm proposed in this paper. We compare two aspects of exact recovery and convergence speed to observe these initialization algorithms' performance in large-scale applications. We use real- or complex-valued Gaussian models with a data dimension of $n = 10000$, and all tests are based on known theoretical limits of information, i.e., $m = 2n - 1$ in the real case and $m = 4n - 4$ in the complex case. The first comparison is the error evolution of the iterates used in the initialization algorithm; that is, the number of iterations used by each algorithm to obtain its exact solution. The error is expressed in logarithmic form, defined as $\log_{10}(1 - \|\mathbf{B}^T \boldsymbol{\omega}_i\|^2 / \|\mathbf{B}^T \mathbf{u}_0\|^2)$, where \mathbf{u}_0 is the exact leading eigenvector of $\bar{\mathbf{Y}} = \mathbf{B}\mathbf{B}^T$ in (10) accurately calculated by SVD.

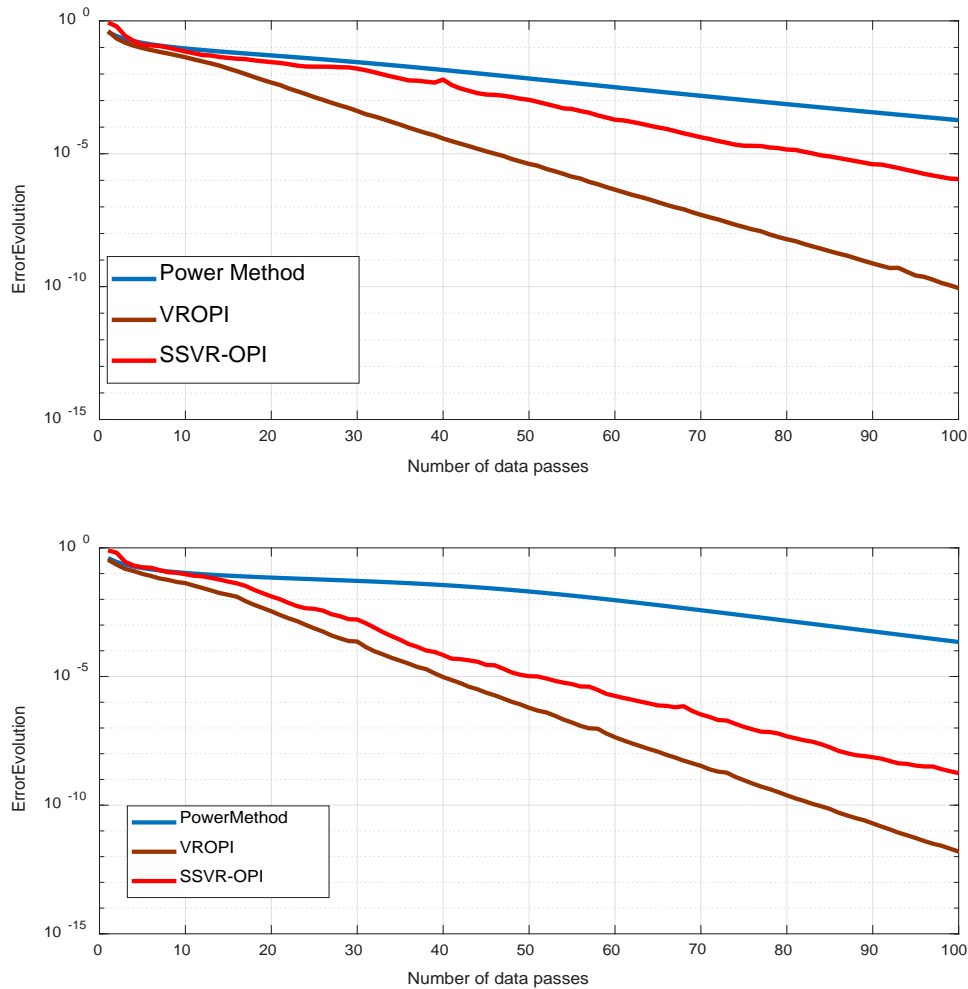


Fig. 1. Error evolution of the iterates using different initialization to solve the problem (11): (top) Noiseless real-valued Gaussian model with a data dimension of $n = 10000$ and $m = 2n - 1$; (bottom) Noiseless complex-valued Gaussian model with a data dimension of $n = 10000$ and $m = 4n - 4$.

Fig. 1 shows the error evolution capabilities of Power Method, VR-OPI, and SSVR-OPI. To be honest, due to the limit of the number of iterations, each iteration of SSVR-OPI only estimates the gradient's snapshot, which does not pass the entire data to estimate the entire gradient-like VR-OPI. Hence the exact recovery effect of SSVR-OPI is slightly weaker than VR-OPI but significantly higher than the power method. On the other hand, because SSVR-OPI does not require passing through all data for each iteration like VR-OPI, it requires less computational complexity or time consumption, about 0.813s per iteration whereas 1.129s in VR-OPI under the real case as shown in **Table 1**.

Table 1. Comparison of time cost between VR-OPI and SSVR-OPI

Algorithms	Real Case	Complex Case
VR-OPI	1.129	6.240
SSVR-OPI	0.813	3.043

In the second experiment, the algorithms equipped with their own initial method and suggested parameter settings were simulated to compare the empirical success rate and the convergence speed. We test the experiment separately under noiseless real- and complex-Gaussian models. In the real case, the signal is generated as $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{1000})$, i.i.d. sampling vector is $\mathbf{a}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{1000})$, and $\mathbf{x} \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_{1000})$, $\mathbf{a}_i \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_{1000})$ under the complex case. **Fig. 2** depicts the empirical success rate of all algorithms. For the real case, when $m/n = 1.7$, the success rate of the SRPR exceeds 90%, and guarantees the perfect recovery when $m = 1.9n$, as shown in **Fig. 2**.

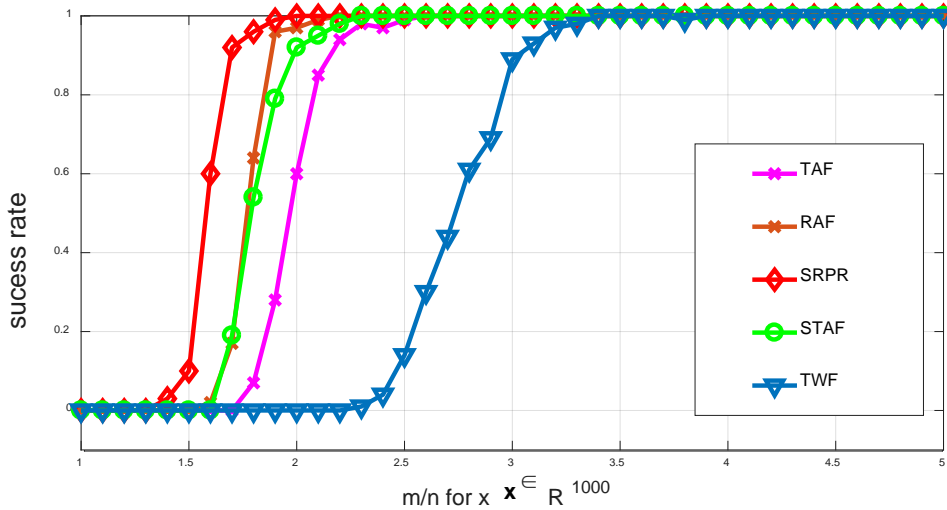


Fig. 2. Empirical success rate with $n = 1000$. Noiseless real-valued Gaussian model with m/n varying by 0.1 from 1 to 5.

Fig. 3 uses $m = 2n - 1$ and $m = 4n - 4$ to compare the convergence speed of each algorithm in the real and complex cases, respectively. It is worth mentioning that TWF was not added in **Fig. 3** (a) because it did not perform well. SRPR can solve the phase retrieval problems with fewer iterations compared with other competitive algorithms.

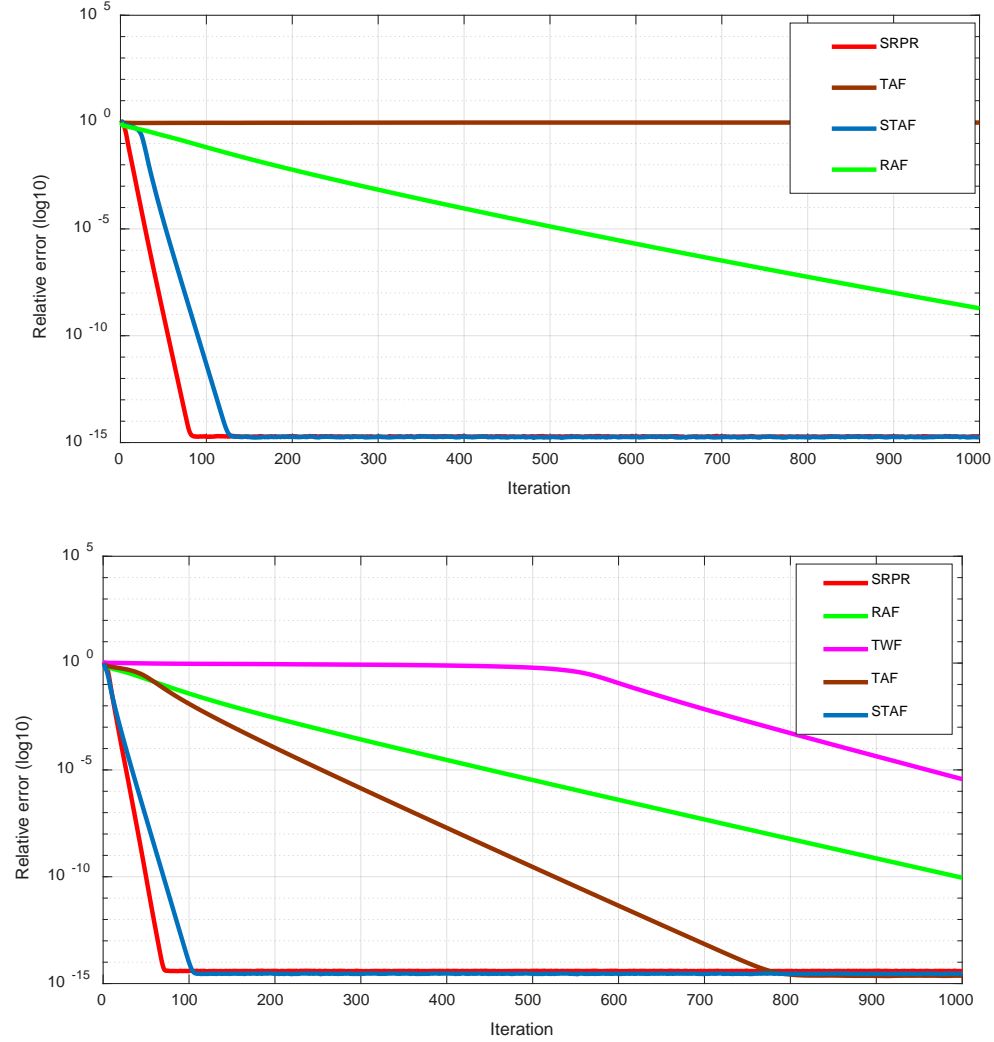


Fig. 3. Relative error versus iterations for $n = 1000$. (top) Noiseless real-valued Gaussian model with $m/n = 2$. (bottom) Noiseless complex-valued Gaussian model with $m/n = 4$.

The last experiment was to test the robustness of SRPR against additive noise destruction. We set up a real noisy Gaussian model $\psi_i = |\langle \mathbf{a}_i, \mathbf{x} \rangle| + \eta_i$ with $\eta_i \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$, where σ^2 is set to satisfy the signal-to-noise ratio (SNR)

$$SNR := 10 \log_{10} \sum_{i=1}^m |\langle \mathbf{a}_i, \mathbf{x} \rangle|^2 / m \sigma^2, \quad (21)$$

which is varied from 10 dB to 50 dB. And the noisy data was generated as $\psi_i = \sqrt{y_i}$. Fig. 4 show the normalized mean-square error

$$NMSE := \text{dist}^2(\mathbf{z}_T, \mathbf{x}) / \|\mathbf{x}\|^2 \quad (22)$$

as a function of SNR for different m/n values. It can be seen from Fig. 4 that the values of NMSE with different m/n are inversely proportional to SNR. From Fig. 4, we can see that when the ratio of m/n is 3 to 6, the curves almost overlap, which proves the stability of SRPR under the influence of additive noise.

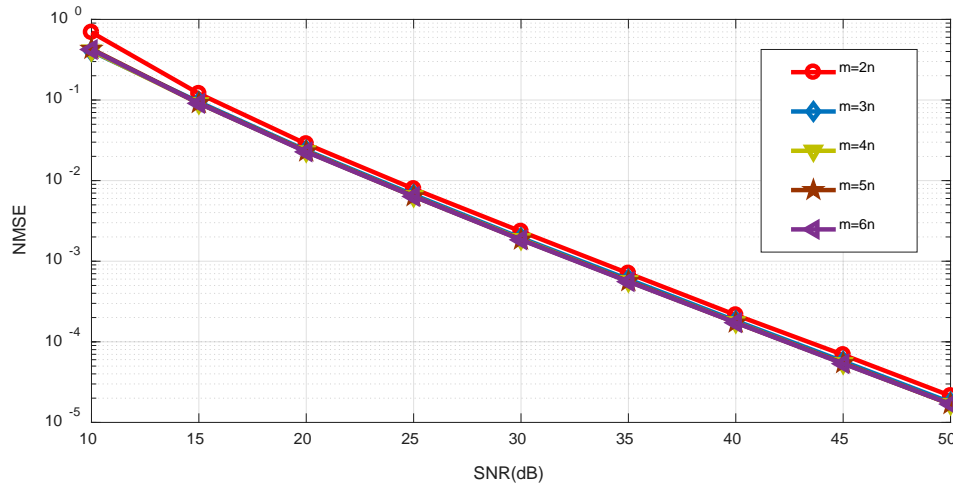


Fig. 4. The normalized mean-square error as a function of SNR for different m/n values.

4. Conclusion

This paper proposed a stochastic reweighted gradient algorithm to solve large-scale phase retrieval problems. The algorithm is divided into two stages. First, a new stochastic variance reduction method is used to solve the initialization stage's valuation problem. Then by a new stochastic reweighted amplitude-based iterations method, we refined the initial estimated value. The simulation test proves that, compared with TAF, TWF, STAF, RAF, the recovery success rate and convergence speed of SRPR are improved in both real and complex cases. Moreover, approaching the limit of information theory, SRPR can still perfectly recover the signal, about $1.9n$ under real Gaussian models.

5. References

- [1] J. C. Dainty and J. R. Fienup, "Phase Retrieval and Image Reconstruction for Astronomy," *Image Recover. theory Appl.*, pp. 231–275, 1987.
- [2] S. C. Mayo et al., "X-ray phase-contrast microscopy and microtomography," *Opt. Express*, vol. 11, no. 19, pp. 2289–2302, 2003. [Article \(CrossRef Link\)](#).
- [3] R. P. Millane, "Phase retrieval in crystallography and optics," *J. Opt. Soc. Am. A*, vol. 7, no. 3, p. 394–411, 1990. [Article \(CrossRef Link\)](#).
- [4] J. Miao, P. Charalambous, J. Kirz, and D. Sayre, "Extending the methodology of X-ray crystallography to allow imaging of micrometre-sized non-crystalline specimens," *Nature*, vol. 400, no. 6742, pp. 342–344, 1999. [Article \(CrossRef Link\)](#).
- [5] C. Jaramillo, R. G. Valenti, L. Guo, and J. Xiao, "Design and analysis of a single-Camera omnistereo sensor for quadrotor Micro Aerial Vehicles (MAVs)," *Sensors (Switzerland)*, vol. 16, no. 2, 2016. [Article \(CrossRef Link\)](#).

- [6] R. Balan, P. Casazza, and D. Edidin, "On signal reconstruction without phase," *Appl. Comput. Harmon. Anal.*, vol. 20, no. 3, pp. 345–356, 2006. [Article \(CrossRef Link\)](#).
- [7] A. S. Bandeira, J. Cahill, D. G. Mixon, and A. A. Nelson, "Saving phase: Injectivity and stability for phase retrieval," *Appl. Comput. Harmon. Anal.*, vol. 37, no. 1, pp. 106–125, 2014. [Article \(CrossRef Link\)](#).
- [8] E. J. Candès, X. Li, and M. Soltanolkotabi, "Phase retrieval via Wirtinger flow: Theory and algorithms," *IEEE Trans. Inf. Theory*, vol. 61, no. 4, pp. 1985–2007, 2015. [Article \(CrossRef Link\)](#).
- [9] Y. C. Eldar and S. Mendelson, "Phase retrieval: Stability and recovery guarantees," *Appl. Comput. Harmon. Anal.*, vol. 36, no. 3, pp. 473–494, 2014. [Article \(CrossRef Link\)](#).
- [10] Y. Chen and E. J. Candès, "Solving Random Quadratic Systems of Equations Is Nearly as Easy as Solving Linear Systems," *Commun. Pure Appl. Math.*, vol. 70, no. 5, pp. 822–883, 2017. [Article \(CrossRef Link\)](#).
- [11] G. Wang, G. B. Giannakis, and Y. C. Eldar, "Solving systems of random quadratic equations via truncated amplitude flow," *IEEE Trans. Inf. Theory*, vol. 64, no. 2, pp. 773–794, 2018. [Article \(CrossRef Link\)](#).
- [12] Y. Shechtman, Y. C. Eldar, O. Cohen, H. N. Chapman, J. Miao, and M. Segev, "Phase Retrieval with Application to Optical Imaging: A contemporary overview," *IEEE Signal Process. Mag.*, vol. 32, no. 3, pp. 87–109, 2015. [Article \(CrossRef Link\)](#).
- [13] Y. Shechtman, A. Beck, and Y. C. Eldar, "GESPAR: Efficient phase retrieval of sparse signals," *IEEE Trans. Signal Process.*, vol. 62, no. 4, pp. 928–938, 2014. [Article \(CrossRef Link\)](#).
- [14] Z. Xiao, Y. Zhang, K. Zhang, D. Zhao, and G. Gui, "GARLM: Greedy autocorrelation retrieval Levenberg-Marquardt algorithm for improving sparse phase retrieval," *Appl. Sci.*, vol. 8, no. 10, 2018. [Article \(CrossRef Link\)](#).
- [15] Y. C. Eldar, P. Sidorenko, D. G. Mixon, S. Barel, and O. Cohen, "Sparse phase retrieval from short-time fourier measurements," *IEEE Signal Process. Lett.*, vol. 22, no. 5, pp. 638–642, 2015. [Article \(CrossRef Link\)](#).
- [16] K. Jaganathan, S. Oymak, and B. Hassibi, "Sparse phase retrieval: Uniqueness guarantees and recovery algorithms," *IEEE Trans. Signal Process.*, vol. 65, no. 9, pp. 2402–2410, 2017. [Article \(CrossRef Link\)](#).
- [17] K. Jaganathan, Y. C. Eldar, and B. Hassibi, "STFT Phase Retrieval: Uniqueness Guarantees and Recovery Algorithms," *IEEE J. Sel. Top. Signal Process.*, vol. 10, no. 4, pp. 770–781, Jun. 2016.
- [18] T. Bendory, Y. C. Eldar and N. Boumal, "Non-Convex Phase Retrieval From STFT Measurements," *IEEE Transactions on Information Theory*, vol. 64, no. 1, pp. 467–484, Jan. 2018. [Article \(CrossRef Link\)](#).
- [19] E. J. Candès, X. Li, and M. Soltanolkotabi, "Phase retrieval from coded diffraction patterns," *Appl. Comput. Harmon. Anal.*, vol. 39, no. 2, pp. 277–299, 2015. [Article \(CrossRef Link\)](#).
- [20] E. J. Candès, X. Li, and M. Soltanolkotabi, "Phase retrieval via wirtinger flow: Theory and algorithms," *IEEE Trans. Inf. Theory*, vol. 61, no. 4, pp. 1985–2007, 2015. [Article \(CrossRef Link\)](#).
- [21] P. Netrapalli, P. Jain, and S. Sanghavi, "Phase retrieval using alternating minimization," *IEEE Trans. Signal Process.*, vol. 63, no. 18, pp. 4814–4826, 2015. [Article \(CrossRef Link\)](#).
- [22] E. J. Candès, T. Strohmer, and V. Voroninski, "PhaseLift: Exact and stable signal recovery from magnitude measurements via convex programming," *Commun. Pure Appl. Math.*, vol. 66, no. 8, pp. 1241–1274, 2013. [Article \(CrossRef Link\)](#).
- [23] G. Wang, G. B. Giannakis, and J. Chen, "Scalable Solvers of Random Quadratic Equations via Stochastic Truncated Amplitude Flow," *IEEE Trans. Signal Process.*, vol. 65, no. 8, pp. 1961–1974, 2017.
- [24] G. Wang, G. B. Giannakis, Y. Saad, and J. Chen, "Phase Retrieval via Reweighted Amplitude Flow," *IEEE Trans. Signal Process.*, vol. 66, no. 11, pp. 2818–2833, 2018.
- [25] L. Zhang, G. Wang, G. B. Giannakis, and J. Chen, "Compressive Phase Retrieval via Reweighted Amplitude Flow," *IEEE Trans. Signal Process.*, vol. 66, no. 19, pp. 5029–5040, 2018. [Article \(CrossRef Link\)](#).

- [26] I. Waldspurger, A. D'Aspremont, and S. Mallat, "Phase recovery, MaxCut and complex semidefinite programming," *Math. Program.*, vol. 149, no. 1–2, pp. 47–81, 2015. [Article \(CrossRef Link\)](#).
- [27] K. Huang, Y. C. Eldar, and N. D. Sidiropoulos, "Phase Retrieval from 1D Fourier Measurements: Convexity, Uniqueness, and Algorithms," *IEEE Trans. Signal Process.*, vol. 64, no. 23, pp. 6105–6117, 2016. [Article \(CrossRef Link\)](#).
- [28] T. Goldstein and C. Studer, "PhaseMax: Convex Phase Retrieval via Basis Pursuit," *IEEE Trans. Inf. Theory*, vol. 64, no. 4, pp. 2675–2689, 2018. [Article \(CrossRef Link\)](#).
- [29] G. Wang, L. Zhang, G. B. Giannakis, M. Akçakaya, and J. Chen, "Sparse phase retrieval via truncated amplitude flow," *IEEE Trans. Signal Process.*, vol. 66, no. 2, pp. 479–491, 2018. [Article \(CrossRef Link\)](#).
- [30] R. W. Gerchberg and W. O. Saxton, "A practical algorithm for the determination of phase from image and diffraction plane pictures," *Optik (Stuttg)*, vol. 35, no. 2, pp. 237–246, 1972.
- [31] J. R. Fienup, "Reconstruction of an object from the modulus of its Fourier transform," *Opt. Lett.*, vol. 3, no. 1, pp. 27–29, 1978. [Article \(CrossRef Link\)](#).
- [32] H. Zhang, Y. Zhou, Y. Liang, and Y. Chi, "Reshaped Wirtinger Flow and Incremental Algorithm for Solving Quadratic System of Equations," to be published.
- [33] J. Sun, Q. Qu, and J. Wright, "A Geometric Analysis of Phase Retrieval," *Found. Comput. Math.*, vol. 18, no. 5, pp. 1131–1198, 2018. [Article \(CrossRef Link\)](#).
- [34] T. Qiu and D. P. Palomar, "Undersampled Sparse Phase Retrieval via Majorization-Minimization," *IEEE Trans. Signal Process.*, vol. 65, no. 22, pp. 5957–5969, 2017. [Article \(CrossRef Link\)](#).
- [35] R. Kolte and A. Özgür, "Phase Retrieval via Incremental Truncated Wirtinger Flow," to be published.
- [36] K. Wei, "Solving systems of phaseless equations via Kaczmarz methods: A proof of concept study," *Inverse Probl.*, vol. 31, no. 12, 2015. [Article \(CrossRef Link\)](#).
- [37] R. Kolte and A. Özgür, "Phase Retrieval via Incremental Truncated Wirtinger Flow," to be published.
- [38] O. Shamir, "Fast stochastic algorithms for SVD and PCA: Convergence properties and convexity," in *Proc. of 33rd International Conference on Machine Learning, ICML 2016*, pp. 248–256, 2016. [Article \(CrossRef Link\)](#).
- [39] R. Johnson and T. Zhang, "Accelerating stochastic gradient descent using predictive variance reduction," *Advances in Neural Information Processing Systems*, vol. 26, pp. 1–9, 2013. [Article \(CrossRef Link\)](#).
- [40] E. Min, J. Long and J. Cui, "Analysis of the Variance Reduction in SVRG and a New Acceleration Method," *IEEE Access*, vol. 6, pp. 16165–16175, 2018. [Article \(CrossRef Link\)](#).
- [41] G. Wang, G. B. Giannakis and J. Chen, "Solving large-scale systems of random quadratic equations via stochastic truncated amplitude flow," in *Proc. of 2017 25th European Signal Processing Conference (EUSIPCO), Kos*, pp. 1420–1424, 2017. [Article \(CrossRef Link\)](#).



Zhuolei Xiao received the B.E. degree in electronics science and technology from the Wuhan University of Technology, Wuhan, China, in 2006. He is currently pursuing the Ph.D. degree in circuits and systems with the Nanjing University of Posts and Telecommunications, Nanjing, China. His current research interests include intelligent information processing, compressed sensing theory in imaging, and signal processing and machine learning.



Yerong Zhang was born in Anhui, China, in 1963. He received the Ph.D. degree in electromagnetic field and microwave technology from the University of Electronic Science and Technology in 1996. He is currently a Professor and a Ph.D. Supervisor of the College of Electronic Science and Engineering, Nanjing University of Posts and Telecommunications, Nanjing, China. His research interests include the planning and optimization of mobile communication network, wireless communication and electromagnetic compatibility, the numerical simulation of electromagnetic wave propagation, and electromagnetic scattering and imaging in the mobile wireless environment.



Jie Yang (M'17) received the B.Sc., M.Sc., and Ph.D. degrees in communication engineering from the Nanjing University of Posts and Telecommunications, Nanjing, China, in 2003, 2006, and 2018, respectively. She is currently an Assistant Professor with the Nanjing University of Posts and Telecommunications.