Open Archive Toulouse Archive Ouverte

OATAO is an open access repository that collects the work of Toulouse researchers and makes it freely available over the web where possible

This is an author's version published in:
http://oatao.univ-toulouse.fr/24767

Official URL
DOI : https://doi.org/10.1109/SSP.2018.8450729

A SMOOTHING STOCHASTIC PHASE RETRIEVAL ALGORITHM FOR SOLVING RANDOM QUADRATIC SYSTEMS

Samuel Pinilla[1], Jorge Bacca[2], Jean-Yves Tourneret[3], Henry Arguello[2]

[1]Department of Electronic Engineering, Universidad Industrial de Santander.
[2]Department of Computer Science, Universidad Industrial de Santander.
[3]University of Toulouse, Toulouse, France.
{samuel.pinilla, jorge.bacca1}@correo.uis.edu.co, jean-yves.tourneret@enseeiht.fr, henarfu@uis.edu.co

ABSTRACT

A novel Stochastic Smoothing Phase Retrieval (SSPR) algorithm is studied to reconstruct an unknown signal $\mathbf{x} \in \mathbb{R}^n$ or $\mathbb{C}^n$ from a set of absolute square projections $y_k = |\langle \mathbf{a}_k, \mathbf{x} \rangle|^2$. This inverse problem is known in the literature as Phase Retrieval (PR). Recent works have shown that the PR problem can be solved by optimizing a non-convex and non-smooth cost function. Contrary to the recent truncated gradient descend methods developed to solve the PR problem (using truncation parameters to bypass the non-smoothness of the cost function), the proposed algorithm approximates the cost function of interest by a smooth function. Optimizing this smooth function involves a single equation per iteration, which leads to a simple scalable and fast method especially for large sample sizes. Extensive simulations suggest that SSPR requires a reduced number of measurements for recovering the signal $\mathbf{x}$, when compared to recently developed stochastic algorithms. Our experiments also demonstrate that SSPR is robust to the presence of additive noise and has a speed of convergence comparable with that of state-of-the-art algorithms.

Index Terms— Phase retrieval, Non-smooth problem, Smoothing function.

1. INTRODUCTION

Phase retrieval is an inverse problem that consists of recovering a signal from the squared modulus of some linear transforms, which has proved efficient in in various applications such as, optics [1], astronomy [2] and X-ray crystallography [3, 4, 5, 6]. Recent works [7, 8, 9] have been proposed to solve the phase retrieval problem by optimizing a non-convex and non-smooth objective function with a gradient descent algorithm based on the Wirtinger derivative with an appropriate initialization. Other stochastic or incremental methods described in [8, 9, 10] retrieve the phase by applying non-convex techniques, which have demonstrated to provide exact recovery from phaseless measurements [11]. More specifically, these methods can use the Incremental Reshaped Wirtinger Flow (IRWF)[8] or the Stochastic Truncated Amplitude Flow (STAF)[9]. Incremental algorithms also offer interesting solutions for signals with large sample size because of their fast convergence and low computational complexity. Moreover, gradient algorithms can be easily converge to a saddle point of the objective function for a non-convex optimization problem. In contrast, stochastic algorithms are often able to escape from these saddle points, and lead to better convergence properties [9, 12]. It is important to highlight that the functions optimized by the IRWF, and STAF methods are non-smooth. In particular, in order to address the non-smoothness of the cost function to be optimized, the STAF introduces truncation procedures to eliminate

the errors in estimated signs with high probability. However, the truncation procedure requires a specific parameter design to obtain a desired performance, which drastically modifies the search direction update, increasing the sampling complexity for phase recovery.

This paper proposes a new Stochastic Smoothing Phase Retrieval (SSPR) algorithm, which uses a specific smooth function to solve the phase retrieval problem. Specifically, SSPR bypass the difficulties resulting from the non-smooth optimization problem by approximating the non-smooth cost function used for phase retrieval by an appropriate smooth function. Theoretical results establish that SSPR is able to converge linearly to the true signal up to a global unimodular constant, because it requires a number of measurements that exceeds in a fixed numerical constant the size of the signal to solve the phase retrieval problem [13]. Additionally, it is interesting to mention that the proposed method does not require any truncation parameter. Simulation results are provided to validate the efficiency of SSPR compared to existing stochastic phase retrieval algorithms. In particular, the sample complexity of the proposed method is lower in terms of number of measurements required to recover the signal.

2. PROBLEM STATEMENT

Before defining the phase retrieval problem, we need to introduce some notations. Denote as $\mathbb{R}_+ = \{x \in \mathbb{R} : x \geq 0\}$ and $\mathbb{R}_{++} = \{x \in \mathbb{R} : x > 0\}$ the sets of positive and strictly positive real numbers. The conjugate and the conjugate transpose of the vector $\mathbf{w} \in \mathbb{C}^n$ will be denoted as $\mathbf{w}^* \in \mathbb{C}^n$ and $\mathbf{w}^H \in \mathbb{C}^n$, respectively. The distance between two complex vectors $\mathbf{w}_1, \mathbf{w}_2 \in \mathbb{C}^n$ used in this work is

$$d_r(\mathbf{w}_1, \mathbf{w}_2) = \min_{\theta \in [0, 2\pi)} \|\mathbf{w}_1 e^{-j\theta} - \mathbf{w}_2\|_2 \qquad (1)$$

where $\|\cdot\|_2$ denotes the Euclidean norm.

The phase retrieval problem can be formulated as determining the solution $\mathbf{x}$ of a system of m quadratic equations of the form

$$y_k = |\langle \mathbf{a}_k, \mathbf{x} \rangle|^2, k = 1, \cdots, m \qquad (2)$$

where $\mathbf{y} := [y_1, \cdots, y_m]^T \in \mathbb{R}^m$ is the measurement vector, $\mathbf{a}_k \in \mathbb{R}^n$(or $\mathbb{C}^n$) are the known sampling vectors and $\mathbf{x} \in \mathbb{R}^n$(or $\mathbb{C}^n$) is the desired unknown real or complex signal. More precisely, this work considers the real and complex Gaussian designs, where vectors $\mathbf{a}_k$ have components distributed according to $\mathcal{N}(0, \mathbf{I}_n)$ distributions (real design) or $\mathbf{a}_k \sim \mathcal{CN}(0, \mathbf{I}_n)$ are complex Gaussian vectors with independent real and imaginary parts distributed according to $\mathcal{N}(0, \frac{1}{2}\mathbf{I}_n)$ distributions (complex design). We also assume that the vectors $\mathbf{a}_1, ..., \mathbf{a}_m$ are independent and identically distributed (i.i.d.).

Adopting the least-squares criterion, the task of recovering a solution from the phaseless measurements in (2) can be formulated as minimizing the following cost function

$$\min_{\mathbf{z} \in \mathbb{C}^n} f(\mathbf{z}) = \frac{1}{m} \sum_{k=1}^{m} \left(|\mathbf{a}_k^H \mathbf{z}| - q_k \right)^2 \tag{3}$$

where $q_k = \sqrt{y_k}$. One motivation for using the square-roots of the measurements in (3) is that it leads to a better signal reconstruction in noisy scenarios as proved in [14]. The works conducted in [9, 10, 15] have proved that (3) can be solved using a sampling complexity of the order $m = \mathcal{O}(n)$. However, the memory requirements and computational complexity make it prohibitive for problems of large dimension, i.e., when sample size n of the signal $\mathbf{x}$ becomes large. In order to overcome these limitations, (3) can be formulated as the following stochastic optimization program

$$\min_{\mathbf{z} \in \mathbb{C}^n} g(\mathbf{z}) = \mathbb{E} \left[\left(|\mathbf{a}_{k_t}^H \mathbf{z}| - q_{k_t} \right)^2 \right] \tag{4}$$

where $\mathbb{E}[\cdot]$ is the expected value and $k_t \in \{1, 2, \cdots, m\}$ is an index per iteration $t \geq 0$. Note that the cost function $f(\mathbf{z})$ in (3) is non-convex and non-smooth [9].

This work proposes a new algorithm based on an auxiliary smoothing function $g_1(\cdot)$ approximating the cost function $g(\cdot)$, in order to solve the non-smooth optimization problem (4). The next section introduces the concept of smoothing function, defines the smoothing function $g_1(\cdot)$ approximating the function $g(\cdot)$ in (4), and introduces the assumptions required to guarantee the convergence of the proposed method.

3. STOCHASTIC SMOOTHING PHASE RETRIEVAL ALGORITHM

In this section, we develop a stochastic algorithm, named Stochastic Smoothing Phase Retrieval (SSPR). This algorithm results from a stochastic gradient descent method based on the Wirtinger derivative, which smoothes the stochastic (non-smooth) cost function $g(\cdot)$ in (4). The concept of Wirtinger derivative and smoothing function were introduced in [16, 17] and are recalled below

Definition 3.1. *Wirtinger derivative* [16]: The Wirtinger derivative of a real-valued function $h(\mathbf{w}) : \mathbb{C}^n \to \mathbb{R}$ with complex-valued argument $\mathbf{w} \in \mathbb{C}^n$ can be computed as

$$\partial h(\mathbf{w}) \triangleq 2 \frac{\partial h(\mathbf{w})}{\partial \mathbf{w}^*} = 2 \left[\frac{\partial h(\mathbf{w})}{\partial w_1^*}, \cdots, \frac{\partial h(\mathbf{w})}{\partial w_n^*} \right]^T \tag{5}$$

where w_i^* denotes the conjugate of w_i. More details related to Wirtinger derivation can be found in [16]. Note that this derivation has been recently used in state-of-the-art methods to solve the phase retrieval problem [11, 15, 13].

Definition 3.2. *Smoothing function:* Let $g : \mathbb{C}^n \to \mathbb{R}$ be a locally Lipschitz continuous function. Then $h : \mathbb{C}^n \times \mathbb{R}_+ \to \mathbb{R}$ is a smoothing function for $g(\cdot)$, if $h(\cdot, \mu)$ is smooth in $\mathbb{C}^n$ for any fixed $\mu \in \mathbb{R}_{++}$ and

$$\lim_{\mu \to 0^+} h(\mathbf{w}, \mu) = g(\mathbf{w}) \tag{6}$$

for any fixed $\mathbf{w} \in \mathbb{C}^n$.

According to Definition 3.2, we consider the following smooth function $\varphi_\mu : \mathbb{R} \to \mathbb{R}_{++}$ defined as

$$\varphi_\mu(w) = \sqrt{w^2 + \mu^2} \tag{7}$$

where $\mu > 0$. This paper proposes to replace (4) by the following smooth problem

$$\min_{\mathbf{z} \in \mathbb{C}^n} g_1(\mathbf{z}, \mu) = \mathbb{E} \left[\ell_{k_t}(\mathbf{z}, \mu) \right] \tag{8}$$

where $\ell_{k_t} = \left(\varphi_\mu(|\mathbf{a}_{k_t}^H \mathbf{z}|) - q_{k_t} \right)^2$. Note that setting $\mu = 0$ in (8) leads to the non-smooth problem in (3). Note also that recent works such as [9, 15] have addressed the non-smoothness of $g(\mathbf{z})$ in (4) by introducing truncation parameters into the gradient step in order to eliminate the errors in the estimated descent direction. However, this procedure can drastically modify the search direction and increases the sampling complexity of the phase retrieval problem. In contrast, introducing the auxiliary function $\varphi_\mu(\cdot)$ allows signal reconstruction without any error in the estimated descent direction for two reasons. First, the resultant cost function is smooth. Second, the search direction induced by the smooth function is unaltered with respect to the initial cost function. Moreover, by designing an appropriate update rule to iteratively decrease the parameter μ, we can ensure that the proposed method provides perfect signal recovery (up to a global unimodular constant).

In order to solve (8), SSPR has two fundamental steps as summarized in Algorithm 1. First, SSPR uses an appropriate initialization for the unknown vector $\mathbf{z}$, in this case the weighted maximal correlation initialization proposed in [9] (see Line 2 of Algorithm 1). Second, SSPR applies stochastic gradient iterations based on the Wirtinger derivative introduced in Definition 3.1 to refine the initial estimate. Specifically, the proposed gradient update iterations are defined by

$$\mathbf{z}_{t+1} = \mathbf{z}_t - \alpha \left(\mathbf{a}_{k_t}^H \mathbf{z}_t - q_{k_t} \frac{\mathbf{a}_{k_t}^H \mathbf{z}_t}{\sqrt{|\mathbf{a}_{k_t}^H \mathbf{z}_t|^2 + \mu_t^2}} \right) \mathbf{a}_{k_t} \tag{9}$$

where $\alpha \in (0, 1)$ is a constant (calculated in Line 4 of Algorithm 1). Following Algorithm 1, at iteration t, if the condition $\|\partial g_1(\mathbf{z}_{t+1}, \mu_t)\|_2 \geq \gamma \mu_t$ is not satisfied (Line 5 of Algorithm 1), the smoothing parameter μ is updated according to Line 8 of Algorithm 1. Using this update rule for the smoothing parameter μ, we can ensure that the generated sequence μ_t tends to zero, which is required to guarantee that SSPR leads to a perfect signal reconstruction. Moreover, the convergence guarantees for Algorithm 1 are established in Section 3.2.

Algorithm 1 SSPR: Stochastic Smoothing Phase Retrieval algorithm

---

1: **Input:** Data $\{(\mathbf{a}_k; q_k)\}_{k=1}^{m}$ and constants $\alpha = 1.6/n$, $\gamma_1 = 0.9$, $\mu_0 = 6 \times 10^4/m$, $\gamma = 0.01$ and maximum number of iterations: $T = 500m$.
2: Initial point $\mathbf{z}_0 = \sqrt{\frac{\sum_{k=1}^{m} q_k^2}{m}} \tilde{\mathbf{z}}_0$, where $\tilde{\mathbf{z}}_0$ is the leading eigenvector of $\mathbf{Y}_0 := \frac{1}{|I_0|} \sum_{k \in I_0} \sqrt{q_k} \frac{\mathbf{a}_k \mathbf{a}_k^H}{\|\mathbf{a}_k\|_2^2}$.
3: **for** $t = 0 : T - 1$ **do**
    Choose k_t uniformly at random from $\{1, 2, \cdots, m\}$
4:    $\mathbf{z}_{t+1} = \mathbf{z}_t - \alpha \left(\mathbf{a}_{k_t}^H \mathbf{z}_t - q_{k_t} \frac{\mathbf{a}_{k_t}^H \mathbf{z}_t}{\sqrt{|\mathbf{a}_{k_t}^H \mathbf{z}_t|^2 + \mu_t^2}} \right) \mathbf{a}_{k_t}$
5:    **if** $\|\partial g_1(\mathbf{z}_{t+1}, \mu_t)\|_2 \geq \gamma \mu_t$ **then**
6:        $\mu_{t+1} = \mu_t$
7:    **else**
8:        $\mu_{t+1} = \gamma_1 \mu_t$
9:    **end if**
10: **end for**
11: **return:** $\mathbf{z}_T$

3.1. Initialization

This work uses the weighted maximal orrelation initialization proposed in [9]. This initialization consists in calculating the vector $\mathbf{z}_0$, which is the leading eigenvector $\tilde{\mathbf{z}}_0$ of the matrix $\mathbf{Y}_0 := \frac{1}{|I_0|} \sum_{k \in I_0} \sqrt{q_k} \frac{\mathbf{a}_k \mathbf{a}_k^H}{\|\mathbf{a}_k\|_2^2}$ scaled by the quantity $\lambda_0 := \sqrt{\frac{\sum_{k=1}^m q_k^2}{m}}$, i.e, $\mathbf{z}_0 = \lambda_0 \tilde{\mathbf{z}}_0$. In [9] it was established that the distance between the initial guess $\mathbf{z}_0$ and the true signal $\mathbf{x}$ is given by

$$d_r(\mathbf{z}_0, \mathbf{x}) \leq \frac{1}{10} \|\mathbf{x}\|_2 \tag{10}$$

with probability exceeding $1 - c_3 e^{-c_4 m}$, providing that $m \geq c_1 |I_0| \geq c_2 n$ for some constants $c_1, c_2, c_3, c_4 > 0$ and sufficiently large n.

3.2. Convergence Conditions

This section provides theoretical guarantees to prove that the proposed method SSPR can reconstruct the true signal (up to global unimodular constant). To do that, we need first to establish how to calculate the Wirtinger derivative of $g_1(\mathbf{x}, \mu)$, which is a useful result to prove the global convergence of the SSPR algorithm in Theorem 3.3.

Lemma 3.1. *The Wirtinger derivative of $g_1(\mathbf{x}, \mu)$ is given by*

$$\partial g_1(\mathbf{x}, \mu) = \mathbb{E}\left[\partial \ell_{k_t}(\mathbf{x}, \mu)\right]. \tag{11}$$

Proof. The proof of this lemma is deferred to Appendix A. □

In order to guarantee the convergence of Algorithm 1, we need to ensure two conditions need to be satisfied: the local error contraction detailed in Theorem 3.2 and the generated sequence μ_t has to converge to zero when t increases. These two conditions are discussed below.

Theorem 3.2. *(Local error contraction): Consider the noiseless measurements $q_k = |\langle \mathbf{a}_k, \mathbf{x} \rangle|$ for an arbitrary signal $\mathbf{x} \in \mathbb{C}^n$, and i.i.d vectors $\{\mathbf{a}_k \sim \mathcal{CN}(0, \mathbf{I}_n)\}_{k=1}^m$. If $\alpha \in (0, \alpha_0/n]$ and $m \geq c_0 n$ then, with probability at least $1 - 2e^{-\epsilon^2 m/2}$, the SSPR algorithm detailed in Algorithm 1 satisfies the following inequality*

$$\mathbb{E}_{k_t}\left[d_r^2(\mathbf{z}_{t+1}, \mathbf{x})\right] \leq \rho (1 - \upsilon)^{t+1} \|\mathbf{x}\|_2^2 \tag{12}$$

for a fixed $\mu_t > 0$, $\rho = 1/10$ and some numerical constant $\upsilon \in (0, 1)$, where the expectation is taken over the random variable k_t, and c_0 is a universal constant.

Proof. See `http://diffraction.uis.edu.co/pdfs/auxiliarCSSPR.pdf`. □

Theorem 3.2 shows that the sequence $\{\mathbf{z}^t\}_{t \geq 1}$ generated by Algorithm 1 is a monotonically decreasing sequence $\{g(\mathbf{z}^t, \mu)\}_{t \geq 1}$, for a fixed value of μ. To prove that the proposed method solves the original optimization problem (2), we have to show that $\{\mu_t\}_{t \geq 1}$ tends to zero, i.e., $\mu_t \to 0$, which is summarized in Theorem 3.3.

Theorem 3.3. *In the setup of Theorem 3.2 we can prove the following result*

- *The sequences $\{\mu_t\}$ and $\{\mathbf{z}_t\}$ generated by Algorithm 1 satisfy $\lim_{t \to \infty} \mu_t = 0$, and $\lim_{t \to \infty} \|\partial g_1(\mathbf{z}_t, \mu_{t-1})\|_2 = 0$.*

Proof. See `http://diffraction.uis.edu.co/pdfs/auxiliarCSSPR.pdf`. □

Note that, since Theorem 3.2 guarantees the local error contraction for any fixed μ_t (adjusted in Line 4 of Algorithm 1), and Theorem 3.3 establishes that $\mu_t \to 0$, the proposed SSPR algorithm ensures an asymptotic perfect reconstruction of the unknown signal, up to a global unimodular constant. Moreover, the SSPR algorithm achieves a linear convergence, since the number of equations m and the number of unknowns exceed a fixed numerical constant [13].

4. SIMULATIONS AND NUMERICAL RESULTS

This section compares the performance of the proposed algorithm with respect to IRWF [8] and STAF [8]. Also, we compare SSPR with some recent non-stochastic phase retrieval methods such as Truncated Amplitude Flow (TAF) [15], and Truncated Wirtinger Flow (TWF) [18]. Note that all the parameters used for the implementation of IRWF and STAF were adjusted as recommended in the related references. The signal considered in this paper was generated as $\mathbf{x} \sim \mathcal{N}(0, \mathbf{I}_n)$ with $n = 1,000$. In the real case, we generated independent vectors $\mathbf{a}_k \sim \mathcal{N}(0, \mathbf{I}_n)$ for $k = 1, ..., m$. In the complex Gaussian case, $\mathbf{x} \sim \mathcal{N}(0, \mathbf{I}_n) + j\mathcal{N}(0, \mathbf{I}_n)$ (with $j^2 = -1$) and the vectors $\mathbf{a}_k$ were generated independently as $\mathbf{a}_k \sim \mathcal{N}(0, \frac{1}{2}\mathbf{I}_n) + j\mathcal{N}(0, \frac{1}{2}\mathbf{I}_n)$ for $k = 1, ..., m$.

The performance metric is the relative error defined as $\frac{d_r(\mathbf{z}, \mathbf{x})}{\|\mathbf{x}\|_2}$, where $d_r(\cdot, \cdot)$ has been defined in (1). We also evaluated the performance using the empirical success rate for 100 Monte Carlo runs. For each run, $1,000$ iterations were used for each algorithm. Note that for stochastic methods, one iteration corresponds to m gradient evaluations of the component functions ℓ_{k_t}. All simulations were implemented in Matlab 2017a on an Intel Core i7 3.41Ghz CPU with 32 GB RAM. For reproducibility, the Matlab code of our SSPR algorithm is publicly available at `http://diffraction.uis.edu.co/codes.html`.

4.1. Test 1: Sampling Complexity

Numerical results were conducted to evaluate the algorithm complexities for the for real and complex cases by varying the number of measurements m/n (with a stepsize of 0.1). A trial was declared as successful when the returned estimate attains a relative error smaller than 10^{-5}. Our results are summarized in Fig. 1. In the real case displayed in Fig. 1(a), SSPR achieves a success rate larger than 93% for $m/n = 1.8$ and guarantees perfect recovery from about $1.9n$ measurements. In the complex case, Fig. 1(b) shows that the SSPR achieves perfect recovery from about $2.7n$ measurements. Note that SSPR requires a reduced number of measurements (for both real and complex cases) to achieve a given performance, when compared with STAF, TAF, TWF, and IRWF. These numerical results confirm the effectiveness of the proposed smoothing scheme.

4.2. Test 2: Noise Robustness

Additional experiments were conducted to demonstrate the robustness of SSPR to additive noise corruption. These experiments were conducted for the real and complex Gaussian models defined by $\hat{y}_k = |\mathbf{a}_k^H \mathbf{x}|^2 + \eta_k$ with $\eta_k \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_n)$. The noisy data was generated as $q_k = \sqrt{\hat{y}_k}$ and we used $\sigma^2 = 0.1^2 \|\mathbf{x}\|_2^2$ with $m/n = 8$. All results are summarized in Figure 2 for the real and complex cases. Fig. 2 shows that SSPR has a faster convergence speed when compared to the other methods (for both real and complex cases), since it requires a smaller number of iterations to solve the phase retrieval problem. Thus, the proposed SSPR algorithm seems to be
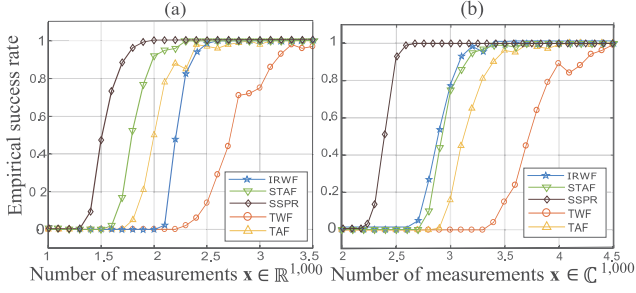
Fig. 1: Empirical success rate versus number of measurements with $n = 1,000$ for the noiseless Gaussian model (a) real case (b) complex case.

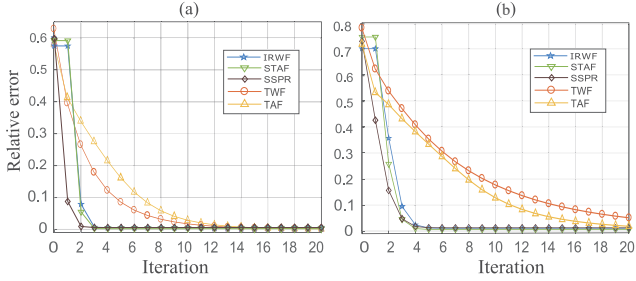more robust than state-of-the-art methods to the presence of additive noise.

Fig. 2: Relative error versus iteration with $n = 1,000$ and $m/n = 8$ for a noisy Gaussian model. (a) Real case (b) Complex case.

4.3. Test 3: Speed of Convergence

Finally, simulations were conducted to compare the convergence speed and sample complexity of all algorithms for complex data in absence of noise. Figs. 3 shows the convergence speed of the different methods in term of number of iterations. More precisely, Figs. 3 shows how the relative error decreases versus the number of iterations for all the algorithms until to achieve a relative error of 10^{-14}. From Fig. 3 it can be observed that SSPR can solve the phase retrieval with less iterations than TAF and TWF, and with a similar number of stochastic iterations with respect to IRWF, and STAF (for both real and complex cases).
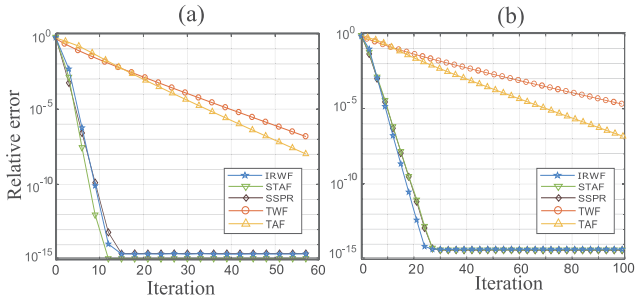

Fig. 3: Relative error versus iteration with $n = 1,000$, $m/n = 8$ for a noiseless Gaussian model. (a) Real case (b) Complex case.

Table 1 reports the number of iterations and time cost for all the algorithms to achieve the relative error of 10^{-14} averaged over 10 trials. The proposed SSPR method has a larger computational complexity compared to the other methods (STAF, IRWF, TWF, and

TAF). However, it is important to note that SSPR requires a smaller number of measurements, and that it exhibits a higher performance against the noise to solve the phase retrieval problem.

Table 1: Comparison of iteration count and time cost among algorithms

Algorithms	Real Case		Complex Case	
	Iterations	Time (s)	Iterations	Time (s)
IRWF	13	1.01	25	10.32
STAF	12	2.37	28	25.24
SSPR	12	3.11	26	28.75
TWF	125	1.00	343	9.31
TAF	105	2.17	372	13.61

5. CONCLUSION

This paper presented a new algorithm to solve the phase retrieval problem based on a specific smoothing function. The proposed method involves a single equation per iteration, allowing us to solve this problem using a similar number of iterations with respect to state-of-the-art stochastic algorithms. The performance of the proposed strategy was shown to be competitive with respect to existing methods, in term of computational complexity, speed of convergence and number of measurements required to obtain a given reconstruction error. Future work includes the development of a heuristic rule to update the smoothing parameter μ in order to reduce the computational complexity of the proposed method. The variable k_t used in the gradient update step was chosen from a uniform distribution. It would also be interesting to study how the distribution of k_t impacts the convergence of the proposed method.

A. PROOF OF LEMMA 3.1

Proof. From Eq. (8) we have that

$$g_1(\mathbf{x}, \mu) = \mathbb{E}\left[\left(\varphi_\mu(|\mathbf{a}_{k_t}^H \mathbf{x}|) - q_{k_t}\right)^2\right]$$
$$= \mathbb{E}\left[\varphi_\mu^2(|\mathbf{a}_{k_t}^H \mathbf{x}|)\right] - 2\mathbb{E}\left[q_{k_t}\varphi_\mu(|\mathbf{a}_{k_t}^H \mathbf{x}|)\right] + \mathbb{E}[q_{k_t}^2].$$
(13)

Since k_t is sampled uniformly at random from $\{1, 2, \cdots, m\}$ then we have that

$$g_1(\mathbf{x}, \mu) = \frac{1}{m}\sum_{k=1}^m \left(\varphi_\mu(|\mathbf{a}_k^H \mathbf{x}|) - q_k\right)^2 = \frac{1}{m}\sum_{k=1}^m \ell_k(\mathbf{x}, \mu) \quad (14)$$

where $\ell_k(\mathbf{x}, \mu) = \left(\varphi_\mu(|\mathbf{a}_k^H \mathbf{x}|) - q_k\right)^2$. From Eq. (14) it can be obtained that

$$\partial g_1(\mathbf{x}, \mu) = \frac{2}{m}\sum_{k=1}^m (\varphi_\mu(|\mathbf{a}_k^H \mathbf{x}|) - q_k)\partial\varphi_\mu(|\mathbf{a}_k^H \mathbf{x}|). \quad (15)$$

On the other hand, note that

$$\mathbb{E}\left[\partial\ell_{k_t}(\mathbf{x}, \mu)\right] = \mathbb{E}\left[2(\varphi_\mu(|\mathbf{a}_{k_t}^H \mathbf{x}|) - q_{k_t})\partial\varphi_\mu(|\mathbf{a}_{k_t}^H \mathbf{x}|)\right]$$
$$= \frac{2}{m}\sum_{k=1}^m (\varphi_\mu(|\mathbf{a}_k^H \mathbf{x}|) - q_k)\partial\varphi_\mu(|\mathbf{a}_k^H \mathbf{x}|). \quad (16)$$

Combining Eqs. (15) and (16) yields

$$\partial g_1(\mathbf{x}, \mu) = \mathbb{E}\left[\partial\ell_{k_t}(\mathbf{x}, \mu)\right]. \quad (17)$$

which concludes the proof. □

B. REFERENCES

[1] Y. Xu, Z. Ren, K. K. Wong, and K. Tsia, "Overcoming the limitation of phase retrieval using Gerchberg–Saxton-like algorithm in optical fiber time-stretch systems," *Optics letters*, vol. 40, no. 15, pp. 3595–3598, 2015.

[2] C. Fienup and J. Dainty, "Phase retrieval and image reconstruction for astronomy," *Image Recovery: Theory and Application*, pp. 231–275, 1987.

[3] S. Pinilla, J. Poveda, and H. Arguello, "Coded diffraction system in x-ray crystallography using a Boolean phase coded aperture approximation," *Optics Communications*, vol. 410, pp. 707–716, 2018.

[4] Samuel Pinilla, Hans García, Luis Díaz, Juan Poveda, and Henry Arguello, "Coded aperture design for solving the phase retrieval problem in x-ray crystallography," *Journal of Computational and Applied Mathematics*, vol. 338, pp. 111–128, 2018.

[5] S. Pinilla, C. Noriega, and H. Arguello, "Stochastic truncated Wirtinger flow algorithm for phase retrieval using Boolean coded apertures," in *Proc. 2017 IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, New-Orleans, USA, Mar. 2017, pp. 6050–6054.

[6] S. Pinilla and H. Arguello, "Phase recovery from diffraction patterns using Boolean coded apertures and the truncated Wirtinger flow algorithm," in *3D Image Acquisition and Display: Technology, Perception and Applications*. Optical Society of America, 2016, pp. JT3A–28.

[7] E. Candes, X. Li, and M. Soltanolkotabi, "Phase retrieval from coded diffraction patterns," *Applied and Computational Harmonic Analysis*, vol. 39, no. 2, pp. 277–299, 2015.

[8] H. Zhang and Y. Liang, "Reshaped Wirtinger flow for solving quadratic system of equations," in *Proc. Advances in Neural Information Processing Systems*, Barcelona, Spain, Dec. 2016, pp. 2622–2630.

[9] G. Wang, G. B. Giannakis, and J. Chen, "Solving large-scale systems of random quadratic equations via stochastic truncated amplitude flow," in *Proc. 25th European Signal Processing Conference (EUSIPCO)*, Kos island, Greece, Aug. 28 - Sept. 2 2017, pp. 1420–1424.

[10] R. Kolte and A. Özgür, "Phase retrieval via incremental truncated Wirtinger flow," *arXiv preprint arXiv:1606.03196*, 2016.

[11] E. Candes, X. Li, and M. Soltanolkotabi, "Phase retrieval via Wirtinger flow: Theory and algorithms," *IEEE Trans. Inform. Theory*, vol. 61, no. 4, pp. 1985–2007, 2015.

[12] R. Ge, F. Huang, C. Jin, and Y. Yuan, "Escaping from saddle points – Online stochastic gradient for tensor decomposition," in *Proc. Conf. Learning Theory*, Paris, France, July 2015, pp. 797–842.

[13] Y. Chen and E. Candes, "Solving random quadratic systems of equations is nearly as easy as solving linear systems," in *Proc. Advances in Neural Information Processing Systems*, Montréal, Canada, Dec. 2015, pp. 739–747.

[14] T. Qiu and D. P. Palomar, "Undersampled phase retrieval via majorization-minimization," *arXiv preprint arXiv:1609.02842*, 2016.

[15] G. Wang, G. B. Giannakis, and Y. C. Eldar, "Solving systems of random quadratic equations via truncated amplitude flow," *arXiv preprint arXiv:1605.08285*, 2016.

[16] R. Hunger, *An introduction to complex differentials and complex differentiability*, Munich University of Technology, Inst. for Circuit Theory and Signal Processing, 2007.

[17] C. Zhang and X. Chen, "Smoothing projected gradient method and its application to stochastic linear complementarity problems," *SIAM Journal on Optimization*, vol. 20, no. 2, pp. 627–649, 2009.

[18] E. Candès and X. Li, "Solving quadratic equations via phaselift when there are about as many equations as unknowns," *Foundations of Computational Mathematics*, vol. 14, no. 5, pp. 1017–1026, 2014.