

Undersampled Sparse Phase Retrieval via Majorization–Minimization

Tianyu Qiu and Daniel P. Palomar, *Fellow, IEEE*

Abstract—In the undersampled phase retrieval problem, the goal is to recover an N -dimensional complex-valued signal from only $M < N$ intensity measurements without phase information. This inverse system is not only nonconvex, but also underdetermined. In this paper, we propose to exploit the sparsity in the original signal and develop two low-complexity algorithms with superior performance based on the majorization–minimization framework. The proposed algorithms are preferred to existing benchmark methods, since at each iteration a simple convex surrogate problem is solved with a closed-form solution that monotonically decreases the objective function value. When the unknown signal is sparse in the standard basis, the first algorithm C-PRIME can produce a stationary point of the corresponding nonconvex phase retrieval problem. When the unknown signal is not sparse in the standard basis, the second algorithm SC-PRIME can find a coordinatewise stationary point of the more challenging phase retrieval problem through sparse coding. Experimental results validate that the proposed algorithms have higher successful recovery rate and less normalized mean square error than existing up-to-date methods under the same setting.

Index Terms—Phase retrieval, sparse coding, dictionary learning, majorization-minimization.

I. INTRODUCTION

PHASE retrieval aims at recovering a complex-valued signal $\mathbf{x} \in \mathbb{C}^N$ from the magnitude squared of M linear measurements (usually corrupted with additive noise $\{n_i\}_{i=1}^M$):

$$y_i = |\mathbf{a}_i^H \mathbf{x}|^2 + n_i \in \mathbb{R}, \quad i = 1, \dots, M. \quad (1)$$

This problem is motivated by the fact that most of the optical devices can easily measure the intensity, rather than the phase, of the incoming light. Hence, it is a challenging task to recover the original signal from only intensity measurements. The measurement vectors $\{\mathbf{a}_i \in \mathbb{C}^N\}_{i=1}^M$ are known. For example, they correspond to rows of the Discrete Fourier Transform (DFT) matrix in various imaging applications; to name a few, optical imaging [1], astronomy [2], crystallography [3], and microscopy [4]. Other non-imaging applications include audio signal

processing [5], etc. A comprehensive overview on the theory and applications of phase retrieval is discussed in [6].

The phase retrieval problem is indeed a non-linear non-convex inverse problem. The original signal $\mathbf{x} \in \mathbb{C}^N$ can only be recovered up to a global phase ambiguity as $\mathbf{x} \cdot e^{j\phi}$ yields the same intensity measurements. In general, the number of measurements M should exceed the dimension of the signal N . On the theoretical side, the authors in [7] have proved that the number of measurements M should at least be on the order of $N \log N$ for a successful recovery with high probability when the measurement vectors are chosen independently and uniformly at random on the unit sphere. Furthermore, result has been established that $M \geq 4N - 4$ measurements are sufficient to reconstruct the original signal $\mathbf{x} \in \mathbb{C}^N$ up to a global phase uncertainty by designing specific measurement vectors [8], [9]. When $N = 2^n + 1, n = 1, 2, \dots, \infty$, $M \geq 4N - 4$ measurements are also necessary [9]. A conjecture has been posed in [10] that $M = 4N - 4$ measurements are both necessary and sufficient for phase retrieval. In special cases where $N = 2$ and $N = 3$, the conjecture has been proved valid. However, when $N = 4$, a counterexample has been presented in [11] to successfully recover a signal in \mathbb{C}^4 from only $4N - 5 = 11$ injective measurements. On the algorithmic side, it has been verified empirically that $M \approx 4N$ measurements are required to recover the original signal $\mathbf{x} \in \mathbb{C}^N$ with high successful rate (close to 1) when the measurement vectors are drawn from independent and identical complex Gaussian distributions [12]. As for a real-valued signal, $M \geq 2N - 1$ measurements have been proved to be both necessary and sufficient to recover $\mathbf{x} \in \mathbb{R}^N$ up to a sign change [13].

The undersampled phase retrieval problem considers recovering an N -dimensional complex-valued signal (up to a global phase ambiguity) from only $M < N$ noisy magnitudes of the linear measurements. One potential approach is to exploit sparsity in the original signal [14]–[16]. It has been proved that $M \geq 8K - 2$ measurements are sufficient to recover a K -sparse (at most K non-zero elements) complex-valued signal $\mathbf{x} \in \mathbb{C}^N$ using random Gaussian measurement vectors ($M \geq 4K - 1$ for real-valued case) [16]. Encouragingly, it is possible to recover a signal with fewer intensity measurements than its actual dimension ($M < N$).

Most existing algorithms on undersampled phase retrieval either directly add a constraint to guarantee sparsity [17], [18] or incorporate an additional term in the objective function to promote sparsity [19], [20]. The convex ℓ_1 norm penalty term $\|\mathbf{x}\|_1$ is well-known for producing sparse solutions. [18] and [20]

Manuscript received December 30, 2016; revised April 27, 2017 and July 6, 2017; accepted August 3, 2017. Date of publication August 29, 2017; date of current version September 19, 2017. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Marco Moretti. This work was supported by the Hong Kong RGC 16206315 Research Grant. (Corresponding author: Tianyu Qiu.)

The authors are with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Hong Kong (e-mail: tqiu@ust.hk; palomar@ust.hk).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TSP.2017.2745459

share the same idea of incorporating into the classical Fienup algorithm [21] a projection step to a set of sparse signals. But [18] requires the exact number of the sparsity level K , and [20] requires the exact value of $\|\mathbf{x}\|_1$, which are not available in practice. The authors in [17] combine the damped Gauss-Newton method and the 2-opt local search method to iteratively update the solution and its support with rough information on the sparsity level K . But the combinatorial nature in the support update procedure makes this algorithm impractical when K is relatively large. A convex semidefinite programming problem has also been proposed to solve the undersampled phase retrieval problem through the matrix-lifting technique (introducing a new variable $\mathbf{X} := \mathbf{x}\mathbf{x}^H$) [22]. But the dimension increase in the lifting procedure limits the application to small scale problems. Another different approach is based on the generalized approximate message passing [23]. But assumptions on the signal and noise probability distributions may not be accurate. Recently, one robust algorithm (UPRwO) [19] has been shown to have small reconstruction error when the intensity measurements are corrupted with outliers and noise.

In this paper, we propose two efficient algorithms based on the majorization-minimization framework to solve the undersampled phase retrieval problem under two different problem settings. The first algorithm C-PRIME is a simple extension of [12] when the original signal is sparse in the standard basis. The second algorithm SC-PRIME is the main contribution of this paper, considering more general cases where the unknown signals are not sparse in the standard basis. We share the same idea of applying the sparse coding techniques to the phase retrieval problem [24], inspired by the fact that a lot of image and video signals can be sparsely approximated by a linear combination of a few columns in a dictionary [25]–[28]. Recently, the authors in [24] have shown encouraging results of exploiting sparse coding for the oversampled phase retrieval problem (DOLPHIn algorithm). In this paper, we propose an efficient algorithm to solve the undersampled phase retrieval problem by jointly designing the dictionary and the sparse codes.

The main contributions of this paper are two algorithms for the undersampled phase retrieval problem with the following properties:

- 1) Improved performance compared to the benchmark methods UPRwO and DOLPHIn;
- 2) Monotonicity and convergence guarantee of the sequence of points generated by the proposed algorithms;
- 3) Lower (or similar) computational complexity and faster (or similar) convergence speed compared to state-of-the-art methods UPRwO and DOLPHIn.

The remaining sections are organized as follows. We first provide a brief introduction of the majorization-minimization framework in Section II. Later, we propose an algorithm to solve the undersampled phase retrieval problem of sparse signals using the majorization-minimization techniques. When the unknown signals are not sparse in the standard basis, we propose another algorithm in Section III to solve the undersampled phase retrieval problem through sparse coding. Convergence analysis and computational complexity for both algorithms are provided in Section IV. Numerical results and comparisons with

up-to-date benchmark methods are presented and discussed in Section V. Finally, we conclude our work in Section VI.

Notation: Boldface upper case letters (e.g., \mathbf{X} , \mathbf{A}) denote matrices, boldface lower case letters (e.g., \mathbf{x} , \mathbf{a}) denote column vectors, and italics (e.g., x , a , D) denote scalars. \mathbb{R} and \mathbb{C} denote the field of real-valued numbers and the field of complex-valued numbers, respectively. For any complex-valued number x , $|x|$ denotes its magnitude, $\arg(x)$ denotes its phase, $\text{Re}[x]$ denotes its real part, and $\text{Im}[x]$ denotes its imaginary part. The superscripts $(\cdot)^T$, $(\cdot)^*$, and $(\cdot)^H$ denote transpose, conjugate, and conjugate transpose, respectively. The curled inequality symbol \succeq is used to denote generalized inequality; $\mathbf{a} \succeq \mathbf{b}$ means element-wise $a_i \geq b_i, \forall i$; and $\mathbf{A} \succeq \mathbf{B}$ means that $\mathbf{A} - \mathbf{B}$ is a Hermitian positive semidefinite matrix. \mathbf{I}_n is the $n \times n$ identity matrix (or simply \mathbf{I} when no confusion is caused). $\lambda_{\max}(\mathbf{A})$ denotes the largest eigenvalue of a Hermitian matrix \mathbf{A} . \odot denotes the Hadamard (element-wise) product of two matrices or vectors of the same size. $\mathbf{1}$ is a matrix or vector with all elements 1, and $\mathbf{0}$ all elements 0. For a vector $\mathbf{x} \in \mathbb{C}^N$, $[\mathbf{x}]_n$ (or x_n) denotes its n -th element, $\|\mathbf{x}\|_2 := \sqrt{\sum_{n=1}^N |x_n|^2}$ denotes its Euclidean norm, and $\|\mathbf{x}\|_1 := \sum_{n=1}^N |x_n|$ denotes its ℓ_1 norm, with $|\cdot|$ denoting modulus for complex-valued numbers and absolute value for real-valued numbers. For a matrix $\mathbf{X} \in \mathbb{C}^{M \times N}$, $[\mathbf{X}]_{mn}$ (or x_{mn}) denotes its element at the m -th row and the n -th column, $\|\mathbf{X}\|_F := \sqrt{\sum_{m=1}^M \sum_{n=1}^N |x_{mn}|^2}$ denotes its Frobenius norm, and \mathbf{X}^\dagger denotes its Moore-Penrose pseudoinverse.

II. COMPRESSIVE PHASE RETRIEVAL VIA MAJORIZATION-MINIMIZATION

In this section, we first provide a brief overview of the general majorization-minimization (MM) framework. Later, we propose a simple iterative algorithm C-PRIME to solve the undersampled phase retrieval problem of sparse signals via the MM techniques. This algorithm is a direct extension of our previous paper [12] but serves as a warm up for the next section where we propose the SC-PRIME algorithm to deal with more general cases when the unknown signals are not sparse in the standard basis.

A. The MM Algorithm

The majorization-minimization (MM) algorithm [29], [30] is an iterative optimization method. Instead of solving the original difficult problem, an MM algorithm deals with simple surrogate problems to produce a sequence of points that can drive the original objective function downhill.

For a real-valued function $f(\boldsymbol{\theta})$, any function $g(\boldsymbol{\theta} | \boldsymbol{\theta}^{(m)})$ satisfying the following two conditions is a majorization function of $f(\boldsymbol{\theta})$ at the point $\boldsymbol{\theta}^{(m)}$:

$$\begin{aligned} g(\boldsymbol{\theta} | \boldsymbol{\theta}^{(m)}) &\geq f(\boldsymbol{\theta}), \quad \forall \boldsymbol{\theta}, \\ g(\boldsymbol{\theta}^{(m)} | \boldsymbol{\theta}^{(m)}) &= f(\boldsymbol{\theta}^{(m)}). \end{aligned} \quad (2)$$

The function $g(\boldsymbol{\theta} | \boldsymbol{\theta}^{(m)})$ is a global upper bound of $f(\boldsymbol{\theta})$ and touches it at the point $\boldsymbol{\theta}^{(m)}$. In addition, $f(\boldsymbol{\theta})$ is said to be majorized by $g(\boldsymbol{\theta} | \boldsymbol{\theta}^{(m)})$ at point $\boldsymbol{\theta}^{(m)}$ if $g(\boldsymbol{\theta} | \boldsymbol{\theta}^{(m)})$ is a majorization function of $f(\boldsymbol{\theta})$ at point $\boldsymbol{\theta}^{(m)}$.

Initialized by any feasible point $\boldsymbol{\theta}^{(0)}$, an MM algorithm generates a sequence of points $\{\boldsymbol{\theta}^{(m)}\}_m$ according to the updating rule

$$\boldsymbol{\theta}^{(m+1)} \in \arg \min_{\boldsymbol{\theta}} g(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(m)}). \quad (3)$$

This sequence of points $\{\boldsymbol{\theta}^{(m)}\}_m$ can drive the original objective function $f(\boldsymbol{\theta})$ downhill:

$$f(\boldsymbol{\theta}^{(m+1)}) \leq g(\boldsymbol{\theta}^{(m+1)} \mid \boldsymbol{\theta}^{(m)}) \leq g(\boldsymbol{\theta}^{(m)} \mid \boldsymbol{\theta}^{(m)}) = f(\boldsymbol{\theta}^{(m)}). \quad (4)$$

The first inequality and the first equality come from the definition of the majorization function (2). The second inequality is valid because $\boldsymbol{\theta}^{(m+1)}$ is a minimizer of $g(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(m)})$ according to (3).

B. C-PRIME

Instead of using the intensity measurements $\{y_i\}_{i=1}^M$, we can use the modulus information $\{\sqrt{y_i}\}_{i=1}^M$ as in [31], [32] (we assume $y_i \geq 0$ otherwise we just discard this measurement). Justification on the advantage of using modulus information $\{\sqrt{y_i}\}_{i=1}^M$ over intensity information $\{y_i\}_{i=1}^M$ is provided in Appendix A. We consider the following problem to balance the importance of minimizing the sum of squared error and utilizing the prior sparsity information of the original signal:

$$\underset{\mathbf{x} \in \mathbb{C}^N}{\text{minimize}} \quad \sum_{i=1}^M (\sqrt{y_i} - |\mathbf{a}_i^H \mathbf{x}|)^2 + \rho \|\mathbf{x}\|_1. \quad (5)$$

The first data fitting term measures how well the signal \mathbf{x} fits the modulus information. The second term promotes sparsity in \mathbf{x} , where $\rho > 0$ is a regularization parameter. If we define the measurement matrix $\mathbf{A} := [\mathbf{a}_1, \dots, \mathbf{a}_M]^H \in \mathbb{C}^{M \times N}$ and stack the modulus information together as a vector $\sqrt{\mathbf{y}} := [\sqrt{y_1}, \dots, \sqrt{y_M}]^T \in \mathbb{R}^M$, problem (5) can be formulated as

$$\underset{\mathbf{x} \in \mathbb{C}^N}{\text{minimize}} \quad \|\sqrt{\mathbf{y}} - |\mathbf{A}\mathbf{x}|\|_2^2 + \rho \|\mathbf{x}\|_1. \quad (6)$$

Here the square root operator $\sqrt{\cdot}$ and the modulus operator $|\cdot|$ are applied element-wise. This problem is not convex due to the modulus operator. Using the majorization-minimization technique, we propose an efficient method to solve the convex surrogate problem instead. Note that

$$\|\sqrt{\mathbf{y}} - |\mathbf{A}\mathbf{x}|\|_2^2 = \mathbf{x}^H \mathbf{A}^H \mathbf{A} \mathbf{x} - 2\sqrt{\mathbf{y}}^T |\mathbf{A}\mathbf{x}| + \text{const.}, \quad (7)$$

where *const.* is a constant independent from the variable \mathbf{x} .

Proposition 1: Let \mathbf{L} be an $N \times N$ Hermitian matrix and \mathbf{M} be another $N \times N$ Hermitian matrix such that $\mathbf{M} \succeq \mathbf{L}$. Then for any point $\mathbf{x}_0 \in \mathbb{C}^N$, the quadratic function $\mathbf{x}^H \mathbf{L} \mathbf{x}$ is majorized by $\mathbf{x}^H \mathbf{M} \mathbf{x} + 2\text{Re}[\mathbf{x}^H (\mathbf{L} - \mathbf{M}) \mathbf{x}_0] + \mathbf{x}_0^H (\mathbf{M} - \mathbf{L}) \mathbf{x}_0$ at \mathbf{x}_0 .

Proof: It is easy to verify the equality condition at point \mathbf{x}_0 . The inequality condition is valid simply by rearranging the terms in $(\mathbf{x} - \mathbf{x}_0)^H (\mathbf{M} - \mathbf{L})(\mathbf{x} - \mathbf{x}_0) \geq 0$. ■

According to Proposition 1, by treating $\mathbf{A}^H \mathbf{A}$ as \mathbf{L} and $C \cdot \mathbf{I}$ as \mathbf{M} for any constant $C \geq \lambda_{\max}(\mathbf{A}^H \mathbf{A})$, $\mathbf{x}^H \mathbf{A}^H \mathbf{A} \mathbf{x}$ can be

majorized¹ as

$$\mathbf{x}^H \mathbf{A}^H \mathbf{A} \mathbf{x} \leq C \mathbf{x}^H \mathbf{x} + 2\text{Re} \left[\mathbf{x}^H (\mathbf{A}^H \mathbf{A} - C\mathbf{I}) \mathbf{x}^{(k)} \right] + \text{const.} \quad (8)$$

at any point $\mathbf{x}^{(k)}$. Further, since

$$\begin{aligned} |\mathbf{A}\mathbf{x}| &= \left| (\mathbf{A}\mathbf{x}) \odot e^{-j \arg(\mathbf{A}\mathbf{x}^{(k)})} \right| \\ &\succeq \text{Re} \left[(\mathbf{A}\mathbf{x}) \odot e^{-j \arg(\mathbf{A}\mathbf{x}^{(k)})} \right], \end{aligned} \quad (9)$$

where $e^{(\cdot)}$ and $\arg(\cdot)$ are applied element-wise when the argument is a vector, $-2\sqrt{\mathbf{y}}^T |\mathbf{A}\mathbf{x}|$ can be majorized as

$$\begin{aligned} -2\sqrt{\mathbf{y}}^T |\mathbf{A}\mathbf{x}| &\leq -2\sqrt{\mathbf{y}}^T \text{Re} \left[(\mathbf{A}\mathbf{x}) \odot e^{-j \arg(\mathbf{A}\mathbf{x}^{(k)})} \right] \\ &= -2\text{Re} \left[\left(\sqrt{\mathbf{y}} \odot e^{-j \arg(\mathbf{A}\mathbf{x}^{(k)})} \right)^T \mathbf{A}\mathbf{x} \right]. \end{aligned} \quad (10)$$

Combining these two majorization functions together, the corresponding surrogate problem for (5) is

$$\begin{aligned} \underset{\mathbf{x} \in \mathbb{C}^N}{\text{minimize}} \quad & C \mathbf{x}^H \mathbf{x} + 2\text{Re} \left[\mathbf{x}^H (\mathbf{A}^H \mathbf{A} - C\mathbf{I}) \mathbf{x}^{(k)} \right] \\ & - 2\text{Re} \left[\left(\sqrt{\mathbf{y}} \odot e^{-j \arg(\mathbf{A}\mathbf{x}^{(k)})} \right)^T \mathbf{A}\mathbf{x} \right] + \rho \|\mathbf{x}\|_1, \end{aligned} \quad (11)$$

which is equivalent to

$$\underset{\mathbf{x} \in \mathbb{C}^N}{\text{minimize}} \quad C \|\mathbf{x} - \mathbf{c}\|_2^2 + \rho \|\mathbf{x}\|_1 \quad (12)$$

if defining the constant vector \mathbf{c} as

$$\mathbf{c} := \mathbf{x}^{(k)} - \frac{1}{C} \mathbf{A}^H \left(\mathbf{A}\mathbf{x}^{(k)} - \sqrt{\mathbf{y}} \odot e^{j \arg(\mathbf{A}\mathbf{x}^{(k)})} \right). \quad (13)$$

A simple closed-form solution for (12) is

$$\mathbf{x}^* = e^{j \arg(\mathbf{c})} \odot \max \left\{ |\mathbf{c}| - \frac{\rho}{2C} \mathbf{1}, \mathbf{0} \right\}. \quad (14)$$

Instead of dealing with the original non-convex non-differentiable problem (5), we only need to solve a surrogate problem (12) that has a simple closed-form solution at every iteration. We name our algorithm compressive phase retrieval via the majorization-minimization technique (C-PRIME for short) and summarize the procedures in Algorithm 1. In the algorithm, we also adopt the SQUAREM algorithm [33] to accelerate the convergence speed. Instead of updating $\mathbf{x}^{(k+1)}$ directly from $\mathbf{x}^{(k)}$ at the k -th iteration, SQUAREM seeks an intermediate point \mathbf{x}_3 based on $\mathbf{x}^{(k)}$ and updates the next point $\mathbf{x}^{(k+1)}$ from \mathbf{x}_3 . This updating rule may violate the descent property of the MM framework so we add a backtracking step (the while loop) to guarantee the descent property. We repeatedly halve the distance between α and -1 until the descent property is valid. In the worst case where $\alpha = -1$, the intermediate point satisfies $\mathbf{x}_3 = \mathbf{x}^{(k)} + 2\mathbf{r} + \mathbf{v} = \mathbf{x}_2$, which ensures that the algorithm will jump out of the while loop. In the simulation, it only takes several updates $\alpha \leftarrow (\alpha - 1)/2$ to maintain the descent property. Fig. 1 plots the objective value of the C-PRIME algorithm at different iterations with/without

¹It is easy to verify the equality condition at point $\mathbf{x}^{(k)}$. We omit it for conciseness from now on.

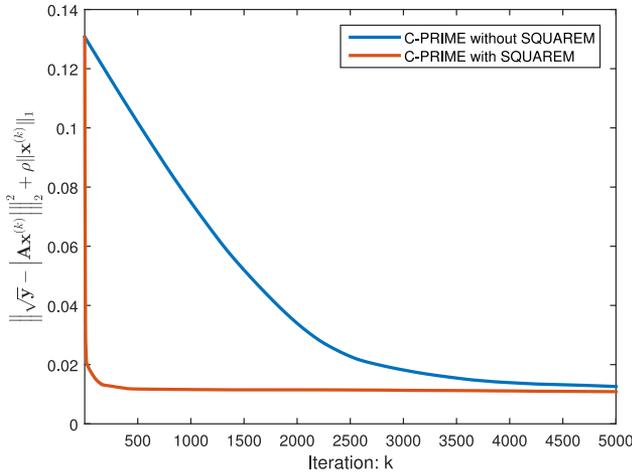


Fig. 1. Objective value of the C-PRIME algorithm at different iterations with/without acceleration using the SQUAREM algorithm; $N = 128$, $M = 64$, $K = 4$, and $\rho = 0.001$.

Algorithm 1: C-PRIME.

Input: \mathbf{A} , \mathbf{y} , ρ , t_0 (maximum iteration number)
1: Initial $\mathbf{x}^{(0)} \leftarrow$ Random vector
2: Choose a constant $C \geq \lambda_{\max}(\mathbf{A}^H \mathbf{A})$
3: **for** $k = 0, \dots, t_0 - 1$ **do**
4: $\mathbf{c}_1 = \mathbf{x}^{(k)} - \frac{1}{C} \mathbf{A}^H (\mathbf{A} \mathbf{x}^{(k)} - \sqrt{\mathbf{y}} \odot e^{j \arg(\mathbf{A} \mathbf{x}^{(k)})})$
5: $\mathbf{x}_1 = e^{j \arg(\mathbf{c}_1)} \odot \max\{|\mathbf{c}_1| - \frac{\rho}{2C} \mathbf{1}, \mathbf{0}\}$
6: $\mathbf{c}_2 = \mathbf{x}_1 - \frac{1}{C} \mathbf{A}^H (\mathbf{A} \mathbf{x}_1 - \sqrt{\mathbf{y}} \odot e^{j \arg(\mathbf{A} \mathbf{x}_1)})$
7: $\mathbf{x}_2 = e^{j \arg(\mathbf{c}_2)} \odot \max\{|\mathbf{c}_2| - \frac{\rho}{2C} \mathbf{1}, \mathbf{0}\}$
8: $\mathbf{r} = \mathbf{x}_1 - \mathbf{x}^{(k)}$
9: $\mathbf{v} = \mathbf{x}_2 - \mathbf{x}_1 - \mathbf{r}$
10: $\alpha \leftarrow -\frac{\|\mathbf{r}\|_2}{\|\mathbf{v}\|_2}$
11: $\mathbf{x}_3 \leftarrow \mathbf{x}^{(k)} - 2\alpha \mathbf{r} + \alpha^2 \mathbf{v}$
12: **while** $\|\sqrt{\mathbf{y}} - |\mathbf{A} \mathbf{x}_3|\|_2^2 + \rho \|\mathbf{x}_3\|_1 > \|\sqrt{\mathbf{y}} - |\mathbf{A} \mathbf{x}_2|\|_2^2 + \rho \|\mathbf{x}_2\|_1$ **do**
13: $\alpha \leftarrow (\alpha - 1)/2$
14: $\mathbf{x}_3 \leftarrow \mathbf{x}^{(k)} - 2\alpha \mathbf{r} + \alpha^2 \mathbf{v}$
15: **end while**
16: $\mathbf{c}_3 = \mathbf{x}_3 - \frac{1}{C} \mathbf{A}^H (\mathbf{A} \mathbf{x}_3 - \sqrt{\mathbf{y}} \odot e^{j \arg(\mathbf{A} \mathbf{x}_3)})$
17: $\mathbf{x}^{(k+1)} = e^{j \arg(\mathbf{c}_3)} \odot \max\{|\mathbf{c}_3| - \frac{\rho}{2C} \mathbf{1}, \mathbf{0}\}$
18: **end for**
Output: $\mathbf{x}^{(t_0)}$.

acceleration using the SQUAREM algorithm when $N = 128$, $M = 64$, $K = 4$, and $\rho = 0.001$. Details of the simulation can be found in Section V. The SQUAREM algorithm accelerates the convergence speed of the proposed algorithm significantly and meanwhile maintains the descent property.

III. SPARSE CODING FOR PHASE RETRIEVAL

In the last section, we proposed a MM-based algorithm to solve the undersampled phase retrieval problem for signals that are sparse in the standard basis. But what if the unknown signals are only sparse with regard to another (known or unknown)

basis,² or more generally a dictionary?³ Typical examples include many image processing applications where the target images are not sparse in the image domain, but instead are sparse in a transform (e.g., discrete cosine transform or wavelet transform) domain. The authors in [25], [26] have shown the advantage of learning an overcomplete dictionary to sparsely represent a signal in an image denoising application. Recently, the dictionary learning techniques [27], [28] have also been exploited to solve the oversampled phase retrieval problem [24]. Inspired by these sparse coding ideas, we propose an efficient algorithm to solve the undersampled phase retrieval problem for signals that are not sparse in the standard basis.

The original signal $\mathbf{x} \in \mathbb{C}^N$ is assumed to admit a sparse approximation over an unknown overcomplete dictionary $\mathbf{D} \in \mathbb{C}^{N \times L}$ with $L > N$; i.e., $\mathbf{x} \approx \mathbf{D} \mathbf{z}$, and $\mathbf{z} \in \mathbb{C}^L$ is the sparse code. Each column \mathbf{d}_l in the dictionary \mathbf{D} is called an atom and is restricted to be in the unit ball $\|\mathbf{d}_l\|_2 \leq 1$. The following problem is considered to jointly recover the signal and design the dictionary:

$$\begin{aligned} & \underset{\mathbf{x}, \mathbf{D}, \mathbf{z}}{\text{minimize}} && \|\sqrt{\mathbf{y}} - |\mathbf{A} \mathbf{x}|\|_2^2 + \mu \|\mathbf{x} - \mathbf{D} \mathbf{z}\|_2^2 + \rho \|\mathbf{z}\|_1 \\ & \text{subject to} && \mathbf{D} \in \mathcal{D}, \end{aligned} \quad (15)$$

where \mathcal{D} is a closed convex set defined as

$$\mathcal{D} := \{\mathbf{D} \in \mathbb{C}^{N \times L} \mid \|\mathbf{d}_l\|_2 \leq 1, \forall l = 1, \dots, L\}. \quad (16)$$

The data fitting term in the objective function is the same as the one in (5); the second term measures how well the unknown signal can be approximated by the dictionary; and the last term promotes sparse code so that only a few atoms are chosen to approximate the unknown signal. The two regularization parameters $\mu > 0$ and $\rho > 0$ are used to balance the weights on the data fitting, the dictionary representation, and the sparse code. Unfortunately, there is more than one solution for (15) because the unknown dictionary is considered as a variable.

Assume there are multiple (not necessarily independent) unknown signals $\{\mathbf{x}_p \in \mathbb{C}^N\}_{p=1}^P$, (we can divide a high dimensional signal into several equal-length low dimensional signals; e.g., divide a large image into small patches), and for every signal \mathbf{x}_p , we only have a few undersampled intensity measurements $\mathbf{y}_p \in \mathbb{R}^M$, $M < N$. Each signal \mathbf{x}_p is assumed to be sparsely approximated by a linear combination of a few atoms in a shared unknown overcomplete dictionary $\mathbf{D} \in \mathbb{C}^{N \times L}$; $\mathbf{x}_p \approx \mathbf{D} \mathbf{z}_p$. The following problem is considered to recover the multiple original signals:

$$\begin{aligned} & \underset{\{\mathbf{x}_p\}, \mathbf{D}, \{\mathbf{z}_p\}}{\text{minimize}} && \sum_{p=1}^P \left(\|\sqrt{\mathbf{y}_p} - |\mathbf{A} \mathbf{x}_p|\|_2^2 + \mu \|\mathbf{x}_p - \mathbf{D} \mathbf{z}_p\|_2^2 \right. \\ & && \left. + \rho \|\mathbf{z}_p\|_1 \right) \\ & \text{subject to} && \mathbf{D} \in \mathcal{D}. \end{aligned} \quad (17)$$

²Vector $\mathbf{x} = \Phi \mathbf{z}$ where $\Phi \in \mathbb{C}^{N \times N}$ is a basis and $\mathbf{z} \in \mathbb{C}^N$ is a sparse vector.

³Vector $\mathbf{x} = \mathbf{D} \mathbf{z}$ where $\mathbf{D} \in \mathbb{C}^{N \times L}$ is a dictionary and $\mathbf{z} \in \mathbb{C}^L$ is a sparse vector. Matrix \mathbf{D} is named a dictionary in the sense that \mathbf{x} can be represented as a linear combination of the columns in \mathbf{D} .

The number of atoms should be less than the number of unknown signals, $L < P$. Otherwise, each signal is trivially represented by a 1-sparse vector \mathbf{z}_p after including $\mathbf{x}_p / \|\mathbf{x}_p\|_2$ as an atom in the dictionary.

Problem (17) is not convex, not only because of the modulus operator, but also because of the quadratic term $\mathbf{D}\mathbf{z}_p$. However, the problem is convex with regard to \mathbf{D} if $\{\mathbf{x}_p\}$ and $\{\mathbf{z}_p\}$ are fixed. Also, it is convex with regard to $\{\mathbf{z}_p\}$ when $\{\mathbf{x}_p\}$ and \mathbf{D} are fixed. Another problem is that all variables are tangled together because of the shared dictionary \mathbf{D} . But once \mathbf{D} is fixed, (17) can be separated into P independent smaller problems. Therefore, we propose to solve this problem using the block successive upper-bound minimization (BSUM) method [34].

A. Updating the Sparse Codes $\{\mathbf{z}_p\}$

We first consider updating $\{\mathbf{z}_p^{(k+1)}\}$ at the k th iteration. When $\{\mathbf{x}_p\}$ and \mathbf{D} are fixed to be $\{\mathbf{x}_p^{(k)}\}$ and $\mathbf{D}^{(k)}$, (17) is equivalent to

$$\underset{\{\mathbf{z}_p\}}{\text{minimize}} \quad \sum_{p=1}^P \left(\mu \left\| \mathbf{D}^{(k)} \mathbf{z}_p - \mathbf{x}_p^{(k)} \right\|_2^2 + \rho \|\mathbf{z}_p\|_1 \right), \quad (18)$$

which can be separated into P independent problems:

$$\underset{\mathbf{z}_p \in \mathbb{C}^L}{\text{minimize}} \quad \mu \left\| \mathbf{D}^{(k)} \mathbf{z}_p - \mathbf{x}_p^{(k)} \right\|_2^2 + \rho \|\mathbf{z}_p\|_1. \quad (19)$$

This is a typical sparse coding problem [28]. If the dictionary $\mathbf{D}^{(k)}$ satisfies $(\mathbf{D}^{(k)})^H \mathbf{D}^{(k)} = \mathbf{I}$ (semi-unitary), (19) is equivalent to

$$\underset{\mathbf{z}_p \in \mathbb{C}^L}{\text{minimize}} \quad \mu \left\| \mathbf{z}_p - (\mathbf{D}^{(k)})^H \mathbf{x}_p^{(k)} \right\|_2^2 + \rho \|\mathbf{z}_p\|_1, \quad (20)$$

which has a simple closed-form solution

$$\mathbf{z}_p^* = e^{j \arg((\mathbf{D}^{(k)})^H \mathbf{x}_p^{(k)})} \odot \max \left\{ \left| (\mathbf{D}^{(k)})^H \mathbf{x}_p^{(k)} \right| - \frac{\rho}{2\mu} \mathbf{1}, \mathbf{0} \right\}. \quad (21)$$

When $(\mathbf{D}^{(k)})^H \mathbf{D}^{(k)} \neq \mathbf{I}$, it is difficult to find a closed-form solution for (19) directly. We propose to solve a surrogate problem instead. According to Proposition 1, $\|\mathbf{D}^{(k)} \mathbf{z}_p - \mathbf{x}_p^{(k)}\|_2^2$ can be majorized as

$$\begin{aligned} & \left\| \mathbf{D}^{(k)} \mathbf{z}_p - \mathbf{x}_p^{(k)} \right\|_2^2 \\ &= \mathbf{z}_p^H (\mathbf{D}^{(k)})^H \mathbf{D}^{(k)} \mathbf{z}_p - 2\text{Re} \left[(\mathbf{x}_p^{(k)})^H \mathbf{D}^{(k)} \mathbf{z}_p \right] + \text{const.} \\ &\leq E^{(k)} \mathbf{z}_p^H \mathbf{z}_p + 2\text{Re} \left[\mathbf{z}_p^H \left((\mathbf{D}^{(k)})^H \mathbf{D}^{(k)} - E^{(k)} \mathbf{I} \right) \mathbf{x}_p^{(k)} \right] \\ &\quad - 2\text{Re} \left[(\mathbf{x}_p^{(k)})^H \mathbf{D}^{(k)} \mathbf{z}_p \right] + \text{const.} \\ &= E^{(k)} \|\mathbf{z}_p - \mathbf{e}_p\|_2^2 + \text{const.}, \end{aligned} \quad (22)$$

where \mathbf{e}_p is a constant vector with regard to the variable \mathbf{z}_p :

$$\mathbf{e}_p := \mathbf{z}_p^{(k)} - \frac{1}{E^{(k)}} (\mathbf{D}^{(k)})^H \left(\mathbf{D}^{(k)} \mathbf{z}_p^{(k)} - \mathbf{x}_p^{(k)} \right). \quad (23)$$

The scalar $E^{(k)} \geq \lambda_{\max}((\mathbf{D}^{(k)})^H \mathbf{D}^{(k)})$ is a constant, and $E^{(k)} \geq L$ is sufficient for a valid majorization function (see

Appendix B). Therefore, the surrogate problem for (19) is

$$\underset{\mathbf{z}_p \in \mathbb{C}^L}{\text{minimize}} \quad \mu E^{(k)} \|\mathbf{z}_p - \mathbf{e}_p\|_2^2 + \rho \|\mathbf{z}_p\|_1, \quad (24)$$

and it has a simple closed-form solution

$$\mathbf{z}_p^* = e^{j \arg(\mathbf{e}_p)} \odot \max \left\{ |\mathbf{e}_p| - \frac{\rho}{2\mu E^{(k)}} \mathbf{1}, \mathbf{0} \right\}. \quad (25)$$

B. Updating the Estimated Signals $\{\mathbf{x}_p\}$

When \mathbf{D} and $\{\mathbf{z}_p\}$ are fixed to be $\mathbf{D}^{(k)}$ and $\{\mathbf{z}_p^{(k+1)}\}$, updating $\{\mathbf{x}_p\}$ leads to solving the following problem:

$$\underset{\{\mathbf{x}_p\}}{\text{minimize}} \quad \sum_{p=1}^P \left(\left\| \sqrt{\mathbf{y}_p} - |\mathbf{A}\mathbf{x}_p| \right\|_2^2 + \mu \left\| \mathbf{x}_p - \mathbf{D}^{(k)} \mathbf{z}_p^{(k+1)} \right\|_2^2 \right), \quad (26)$$

which can also be separated into P independent problems:

$$\underset{\mathbf{x}_p \in \mathbb{C}^N}{\text{minimize}} \quad \left\| \sqrt{\mathbf{y}_p} - |\mathbf{A}\mathbf{x}_p| \right\|_2^2 + \mu \left\| \mathbf{x}_p - \mathbf{D}^{(k)} \mathbf{z}_p^{(k+1)} \right\|_2^2. \quad (27)$$

This problem is not convex due to the modulus operator. We solve a surrogate problem instead. According to Proposition 1 and (10), choosing a constant $F \geq \lambda_{\max}(\mathbf{A}^H \mathbf{A})$, the objective function can be majorized as

$$\begin{aligned} & \left\| \sqrt{\mathbf{y}_p} - |\mathbf{A}\mathbf{x}_p| \right\|_2^2 + \mu \left\| \mathbf{x}_p - \mathbf{D}^{(k)} \mathbf{z}_p^{(k+1)} \right\|_2^2 \\ &= \mathbf{x}_p^H \mathbf{A}^H \mathbf{A} \mathbf{x}_p - 2\sqrt{\mathbf{y}_p}^T |\mathbf{A}\mathbf{x}_p| \\ &\quad + \mu \mathbf{x}_p^H \mathbf{x}_p - 2\mu \text{Re} \left[\mathbf{x}_p^H \mathbf{D}^{(k)} \mathbf{z}_p^{(k+1)} \right] + \text{const.} \\ &\leq F \mathbf{x}_p^H \mathbf{x}_p + 2\text{Re} \left[\mathbf{x}_p^H (\mathbf{A}^H \mathbf{A} - F\mathbf{I}) \mathbf{x}_p^{(k)} \right] \\ &\quad - 2\text{Re} \left[\left(\sqrt{\mathbf{y}_p} \odot e^{-j \arg(\mathbf{A}\mathbf{x}_p^{(k)})} \right)^T \mathbf{A}\mathbf{x}_p \right] \\ &\quad + \mu \mathbf{x}_p^H \mathbf{x}_p - 2\mu \text{Re} \left[\mathbf{x}_p^H \mathbf{D}^{(k)} \mathbf{z}_p^{(k+1)} \right] + \text{const.} \\ &= (F + \mu) \|\mathbf{x}_p - \mathbf{f}_p\|_2^2 + \text{const.}, \end{aligned} \quad (28)$$

where \mathbf{f}_p is a constant vector with regard to the variable \mathbf{x}_p :

$$\begin{aligned} \mathbf{f}_p := & \frac{1}{F + \mu} \left[F \mathbf{x}_p^{(k)} - \mathbf{A}^H \left(\mathbf{A}\mathbf{x}_p^{(k)} - \sqrt{\mathbf{y}_p} \odot e^{j \arg(\mathbf{A}\mathbf{x}_p^{(k)})} \right) \right. \\ & \left. + \mu \mathbf{D}^{(k)} \mathbf{z}_p^{(k+1)} \right]. \end{aligned} \quad (29)$$

Therefore, the surrogate problem for (27) is

$$\underset{\mathbf{x}_p \in \mathbb{C}^N}{\text{minimize}} \quad (F + \mu) \|\mathbf{x}_p - \mathbf{f}_p\|_2^2, \quad (30)$$

and it has a simple closed-form solution

$$\mathbf{x}_p^* = \mathbf{f}_p. \quad (31)$$

The constant vector \mathbf{f}_p is similar to the constant vector \mathbf{c} in (13), which was used to update the solution in the last section. The additional term $\mu \mathbf{D}^{(k)} \mathbf{z}_p^{(k+1)}$ in \mathbf{f}_p is due to the second approximation over the dictionary term in (17).

tional complexity $\mathcal{O}(MN)$ under a general measurement matrix setting, and $\mathcal{O}(M \log M)$ under a DFT measurement matrix setting by exploiting fast Fourier transform and inverse fast Fourier transform. When the unknown signal is not sparse in the standard basis, SC-PRIME utilizes the sparse coding techniques to approximate the unknown signal by a linear combination of a few columns in a dictionary. The computational complexity of SC-PRIME is $\mathcal{O}(LNP)$.

V. SIMULATION RESULTS

In this section, we investigate the numerical performance of the proposed algorithms, C-PRIME and SC-PRIME, and compare them with two up-to-date benchmark methods: UPRwO [19] and DOLPHIn [24], respectively. Simulation results validate that C-PRIME and SC-PRIME outperform their corresponding benchmark method in terms of successful recovery rate and normalized mean square error under the same setting. All experiments are conducted on a computer with a 3.20 GHz Intel Core i5-4570 CPU and 8.00 GB RAM running Matlab R2014b.

A. C-PRIME vs. UPRwO

We first investigate the performance of C-PRIME and compare it with the benchmark method UPRwO [19]. To implement the UPRwO algorithm, we use the code provided on the authors' homepage.⁴ We set the number of outliers to 0 and choose the additive white Gaussian noise setting. All other parameters are set as the default value. The default signal-to-noise ratio (SNR) is 40 dB.

The initialization steps of the UPRwO algorithm are summarized below:

- 1) Generating a random original signal $\mathbf{x}_o \in \mathbb{C}^N$ with cardinality K , where the support is also randomly selected;
- 2) Forming the measurement matrix $\mathbf{A} \in \mathbb{C}^{M \times N}$ by randomly selecting M rows in the $N \times N$ unitary DFT matrix (in this way $\mathbf{A}\mathbf{A}^H = \mathbf{I}_M$ and $\mathbf{I}_N \succeq \mathbf{A}^H\mathbf{A}$);
- 3) Generating the intensity measurements $\mathbf{y} = |\mathbf{A}\mathbf{x}_o|^2 + \mathbf{n}$, where $\mathbf{n} := [n_1, \dots, n_M]^T \in \mathbb{R}^M$ is a vector representing the additive white Gaussian noise.

For a fair comparison, we run the UPRwO code first with a fixed (N, M, K) value. Besides the final results, we also store the original signal \mathbf{x}_o , the measurement matrix \mathbf{A} , and the intensity measurements \mathbf{y} . Later, we run our C-PRIME code using the same measurement matrix \mathbf{A} and intensity measurements \mathbf{y} from the UPRwO simulation.

In detail, the length of the original signal N is set as the default value 128. Since we consider the undersampled phase retrieval problem, the number of intensity measurements is limited to be $M \in \{128, 64, 32, 16, 8\}$, and the sparsity level is set to be $K \in \{16, 8, 4, 2\}$ (a value larger than 16 ends up with unsuccessful recovery). For each of these possible (M, K) pairs, both algorithms are tested using the same measurement matrix and intensity measurements. Multiple random initializations are used by the benchmark method UPRwO in order to

TABLE I
SUCCESSFUL RECOVERY RATE OF UPRwO AND C-PRIME (PROPOSED) FOR AN $N = 128$ -LENGTH COMPLEX-VALUED SIGNAL UNDER DIFFERENT (M, K) SETTINGS. THE VALUE IS PRESENTED AS (UPRwO, C-PRIME)

M	128	0.93, 0.94	1.00, 1.00	0.91, 1.00	0.00, 0.03
	64	0.91, 0.89	0.98, 1.00	0.03, 0.40	0.00, 0.00
	32	0.86, 0.83	0.49, 0.72	0.00, 0.01	0.00, 0.00
	16	0.59, 0.68	0.01, 0.05	0.00, 0.00	0.00, 0.00
	8	0.10, 0.25	0.00, 0.00	0.00, 0.00	0.00, 0.00
Recovery Rate		2	4	8	16
		K			

TABLE II
NORMALIZED MEAN SQUARE ERROR (NMSE) OF UPRwO AND C-PRIME (PROPOSED) FOR AN $N = 128$ -LENGTH COMPLEX-VALUED SIGNAL UNDER DIFFERENT (M, K) SETTINGS. THE VALUE IS PRESENTED AS (UPRwO, C-PRIME)

M	128	0.003, 0.003	0.000, 0.000	0.065, 0.000	1.066, 0.952
	64	0.005, 0.005	0.008, 0.000	0.843, 0.383	1.054, 1.085
	32	0.000, 0.001	0.313, 0.089	0.866, 0.849	1.028, 1.103
	16	0.155, 0.006	0.778, 0.600	0.920, 0.932	1.006, 1.079
	8	0.542, 0.264	0.900, 0.807	0.960, 0.928	1.013, 0.999
NMSE		2	4	8	16
		K			

TABLE III
AVERAGE CPU TIME OF UPRwO AND C-PRIME (PROPOSED) FOR AN $N = 128$ -LENGTH COMPLEX-VALUED SIGNAL UNDER DIFFERENT (M, K) SETTINGS. THE VALUE IS PRESENTED AS (UPRwO, C-PRIME)

M	128	55.0, 56.5	59.3 , 82.8	59.8 , 125	58.3 , 133
	64	30.7, 13.7	33.3, 27.5	32.8 , 39.3	32.4 , 39.8
	32	28.3, 8.41	31.4, 21.9	31.5, 28.0	31.1, 29.0
	16	25.8, 7.19	27.4, 18.9	27.2, 22.0	27.4, 23.6
	8	40.8, 13.9	21.5, 19.1	21.6, 19.5	20.6, 20.2
Time(s)		2	4	8	16
		K			

increase the chance of finding the global optimal solution of the non-convex phase retrieval problem; specifically, 50 initializations are performed when $M < N$, and 100 when $M = N$. The proposed algorithm C-PRIME uses the identical number of random initializations for the same purpose of finding the global optimal solution of the non-convex problem (5) with a higher probability.

Note that under the DFT measurement matrix setting, any individual or combination of the following three trivial ambiguities conserve the Fourier magnitude:

- 1) Global constant phase shift: $\mathbf{x} \rightarrow \mathbf{x} \cdot e^{j\phi}$,
- 2) Circular shift: $[\mathbf{x}]_i \rightarrow [\mathbf{x}]_{(i+i_0) \bmod N}$,
- 3) Conjugate inversion: $[\mathbf{x}]_i \rightarrow [\mathbf{x}]_{N-i}^*$.

A disambiguation step is necessary to find the unique solution. For each solution \mathbf{x}^* returned by UPRwO and C-PRIME, we check all the possible candidates within the trivial ambiguities and choose the one with least normalized squared error (NSE) with regard to the original signal \mathbf{x}_o as the final solution. The

⁴<http://people.virginia.edu/~dsw8c/sw.html>

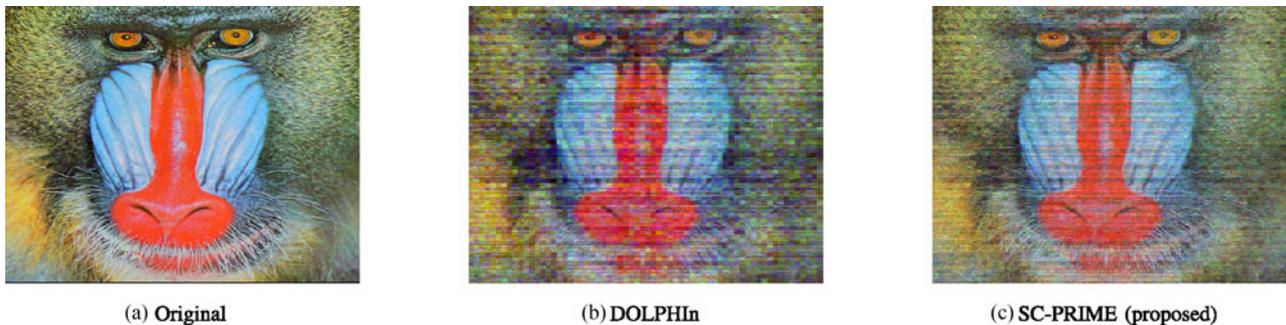


Fig. 2. Reconstruction results of DOLPHIn and SC-PRIME (proposed) on the 512×512 color mandrill image. (a) the original image; (b) image reconstructed by DOLPHIn, PSNR = 14.84 dB, SSIM = 0.4148, $t = 127.0$ s; (c) image reconstructed by SC-PRIME, PSNR = 17.17 dB, SSIM = 0.5773, $t = 76.79$ s.

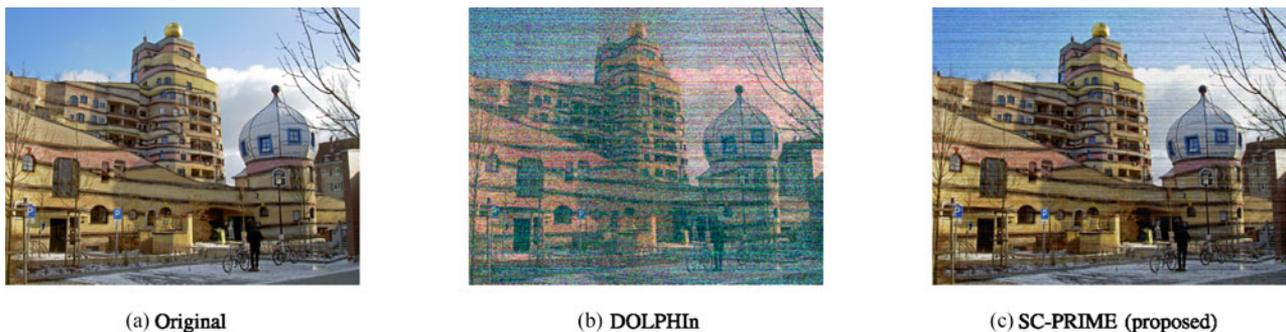


Fig. 3. Reconstruction results of DOLPHIn and SC-PRIME (proposed) on the 2816×2112 color waldspirale image. (a) the original image; (b) image reconstructed by DOLPHIn, PSNR = 10.81 dB, SSIM = 0.1143, $t = 3061$ s; (c) image reconstructed by SC-PRIME, PSNR = 20.60 dB, SSIM = 0.6583, $t = 1429$ s.

NSE between \mathbf{x}^* and \mathbf{x}_o is calculated as

$$\text{NSE}(\mathbf{x}^*, \mathbf{x}_o) = \min_{\mathbf{x} \in \mathcal{S}(\mathbf{x}^*)} \frac{\|\mathbf{x} - \mathbf{x}_o\|_2^2}{\|\mathbf{x}_o\|_2^2}, \quad (39)$$

where the set $\mathcal{S}(\mathbf{x}^*)$ contains all the possible signals within the trivial ambiguities of \mathbf{x}^* . Furthermore, since the original signal is generated as a random vector, the normalized mean square error (NMSE) is averaged over 100 Monte Carlo simulations for every (M, K) pair. Among these 100 Monte Carlo simulations, an algorithm is considered to successfully recover the original signal if the corresponding NSE is less than 10^{-3} .

Final experimental results of UPRwO and C-PRIME are presented in Table I on the successful recovery rate and Table II on the NMSE. Under most of the (M, K) settings, the proposed algorithm C-PRIME has a larger successful recovery rate and less NMSE than the benchmark algorithm UPRwO. Both algorithms can retrieve the original signal with high probability when $M \geq 16K$, but neither performance very well when $M \leq 8K$. Note that UPRwO has a double loop and needs to tune a lot of parameters. In comparison, the proposed algorithm C-PRIME only has one parameter ρ/C . The constant C is set as $C = 1$ since $\mathbf{I}_N \succeq \mathbf{A}^H \mathbf{A}$, and $\rho = 0.001$ is used in the simulations. In addition, the maximum iteration number t_0 in Algorithm 1 is set as $t_0 = 5000$. The average CPU time of UPRwO and C-PRIME over the 100 Monte Carlo simulations is presented in Table III. Both algorithms have a similar computational time.

B. SC-PRIME vs. DOLPHIn

We now investigate the performance of SC-PRIME and compare it with the benchmark method DOLPHIn [24] on practical test images. To implement DOLPHIn, we use the code provided on the authors' homepage.⁵ The test images are also downloaded from the same website. We choose the Gaussian measurement matrix setting and change the sampling rate from 4 to 0.5 ($M = 0.5N$) to set up a valid undersampled phase retrieval problem.⁶ All other parameters are kept as the default value.

At the initialization step, the DOLPHIn algorithm takes the 2D image as the original signal, thereby generating the intensity measurements using a random complex-valued Gaussian measurement matrix. The intensity measurements are corrupted with additive white Gaussian noise. The default SNR is 15 dB. First, we run the DOLPHIn code and store the measurement matrix as well as the noisy intensity measurements. Later, we run the SC-PRIME code using the same measurement matrix and noisy intensity measurements. The test images are divided into 8×8 non-overlapping patches (i.e., $N = 64$), and the number of columns in the dictionary is set as $L = 2N$. There is a

⁵<http://www.mathematik.tu-darmstadt.de/~tillmann/#software>

⁶The DOLPHIn algorithm [24] considers the oversampled phase retrieval problem where the default sampling rate is 4 in the code, i.e., $M = 4N$. In this paper, we consider the undersampled phase retrieval problem where the number of intensity measurements is less than the dimension of the unknown signal, i.e., $M \leq N$. When $M = 4N$, both algorithms can reconstruct the images with high visual quality, and SC-PRIME (proposed) still outperforms DOLPHIn in terms of PSNR and SSIM.

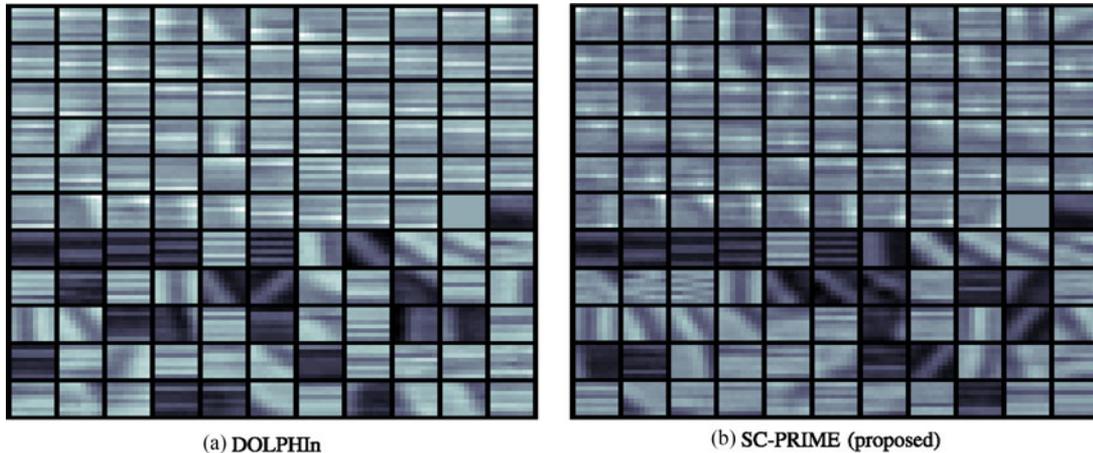

 Fig. 4. Dictionary learned by DOLPHIn and SC-PRIME (proposed) for the red channel on the 2816×2112 color waldspirale image (Fig. 3(a)).

 TABLE IV
 RECONSTRUCTION RESULTS OF DOLPHIN AND SC-PRIME (PROPOSED) ON DIFFERENT TEST IMAGES

PSNR(dB), SSIM		256 × 256 images			512 × 512 images				
		Cameraman	House	Peppers	Barbara	Boat	Fingerprint	Lena	Mandrill
DOLPHIn	X	16.2, 0.255	16.3, 0.235	16.1, 0.286	17.0, 0.291	17.1, 0.263	15.4, 0.394	17.5, 0.270	16.8, 0.282
	DZ	15.8, 0.266	15.9, 0.281	15.7, 0.308	16.4, 0.287	16.6, 0.281	15.1, 0.366	16.9, 0.319	16.2, 0.256
SC-PRIME	X	16.6, 0.296	18.0, 0.312	17.3, 0.344	18.3, 0.319	18.8, 0.285	16.4, 0.433	19.3, 0.284	19.2, 0.371
	DZ	16.0, 0.306	17.2, 0.387	16.5, 0.377	18.6, 0.359	19.0, 0.351	16.6, 0.446	19.7, 0.391	19.2, 0.390

 TABLE V
 AVERAGE CPU TIME OF DOLPHIN AND SC-PRIME (PROPOSED) ON DIFFERENT TEST IMAGES

Time(s)	256 × 256 images			512 × 512 images				
	Cameraman	House	Peppers	Barbara	Boat	Fingerprint	Lena	Mandrill
DOLPHIn	7.97	8.07	8.05	22.3	21.0	22.6	20.1	23.4
SC-PRIME	8.02	8.17	8.28	24.3	23.6	24.6	23.6	23.5

sorting step between the image signal and the patch-based signal in [24]. The corresponding change in the implementation of SC-PRIME is easy as changing the order of the elements in a matrix conserves its Frobenius norm. The regularization parameter μ is set as $\mu = N_P/80$ where N_P is the number of pixels in the test image. The other regularization parameter ρ is set as $\rho = 0.42\mu$.

To evaluate the quality of the reconstructed images, two standard image quality metrics are considered, namely, the peak signal-to-noise ratio (PSNR) and the structural similarity index (SSIM). PSNR is the ratio between the maximum possible power of the original image and the mean squared error between the reconstructed image and the original image. It is usually expressed in terms of the logarithmic decibel scale, and the larger the value the better the image quality. SSIM reflects the structural similarities between the reconstructed image and the original image. It is on a scale from 0 to 1, and a larger value represents more similarities in the structure.

Final reconstruction results of DOLPHIn and SC-PRIME are presented in Fig. 2 on the 512×512 color mandrill image and Fig. 3 on the 2816×2112 color waldspirale image. The three different channels, red, green, and blue, are processed

independently and sequentially. In both figures, the proposed algorithm SC-PRIME can reconstruct the image with a larger PSNR and SSIM value (averaged over three channels) as well as an impressively better visual quality than the benchmark algorithm DOLPHIn. Fig. 4 plots the learned dictionary (each column is arranged as an 8×8 patch) for the red channel by both algorithms on the 2816×2112 color waldspirale image (Fig. 3(a)). The horizontal streaks in the learned dictionary most likely result in the visible horizontal artifacts of the reconstructed images in Figs. 2 and 3 as each patch in the reconstructed image is a linear combination of the columns in the learned dictionary. Moreover, the PSNR and SSIM value of the reconstructed images for both algorithms on the rest of the test images (grayscale) is summarized in Table IV. Besides the results of the reconstructed images (\mathbf{X}^*), we also include the results of images approximated by the dictionary ($\mathbf{D}^*\mathbf{Z}^*$). All numbers in the table are averaged over 100 Monte Carlo simulations as the intensity measurements are corrupted with additive white Gaussian noise. For each of these test images, SC-PRIME can reconstruct the image with a larger PSNR and SSIM value than DOLPHIn, either directly from \mathbf{X}^* or through the dictionary approximation $\mathbf{D}^*\mathbf{Z}^*$. Interestingly, the images

approximated by the dictionary $\mathbf{D}^* \mathbf{Z}^*$ can have a slightly larger PSNR and SSIM value than those reconstructed directly from \mathbf{X}^* . In addition, the average CPU time of both algorithms over the 100 Monte Carlo simulations is presented in Table V. The proposed algorithm SC-PRIME needs slightly more CPU time than the benchmark method DOLPHIn.

VI. CONCLUSION

Undersampled phase retrieval aims at recovering an N -dimensional complex-valued signal from only $M < N$ intensity measurements. It is a non-convex and under-determined inverse problem. In this paper, we have proposed two efficient algorithms, exploiting the sparsity in the original signal, to solve the undersampled phase retrieval problem. Based on the majorization-minimization framework, the proposed algorithms solve a simple convex surrogate problem at every iteration with a closed-form solution that monotonically decreases the objective function value. When the unknown signal is sparse in the standard basis, the first algorithm C-PRIME can produce a stationary point of the corresponding non-convex phase retrieval problem. When the unknown signal is not sparse in the standard basis, the second algorithm SC-PRIME can find a coordinate-wise stationary point of the more challenging phase retrieval problem through sparse coding. According to experimental results on randomly generated data and practical test images, the proposed algorithms have higher successful recovery rate and less normalized mean square error than existing up-to-date methods under the same setting.

APPENDIX A

JUSTIFICATION FOR USING MODULUS INFORMATION

Recall the M noisy intensity measurements

$$y_i = |\mathbf{a}_i^H \mathbf{x}|^2 + n_i, \quad i = 1, \dots, M, \quad (40)$$

and the additive noise $\{n_i\}$ is assumed to be independent from the intensity measurements. We assume $y_i \geq 0$ (otherwise we just discard this measurement). Therefore, the modulus information is

$$\sqrt{y_i} = \sqrt{|\mathbf{a}_i^H \mathbf{x}|^2 + n_i} = |\mathbf{a}_i^H \mathbf{x}| \sqrt{1 + \frac{n_i}{|\mathbf{a}_i^H \mathbf{x}|^2}}, \quad \forall i. \quad (41)$$

Usually the noise level is much smaller than the value of the clean intensity measurements, $|n_i| \ll |\mathbf{a}_i^H \mathbf{x}|^2$. It is sufficient to make the following approximation, taking the first two terms in the Taylor series:

$$\sqrt{1 + \frac{n_i}{|\mathbf{a}_i^H \mathbf{x}|^2}} \approx 1 + \frac{n_i}{2|\mathbf{a}_i^H \mathbf{x}|^2}, \quad \forall i. \quad (42)$$

So the modulus information satisfies

$$\sqrt{y_i} \approx |\mathbf{a}_i^H \mathbf{x}| + \frac{n_i}{2|\mathbf{a}_i^H \mathbf{x}|}, \quad \forall i. \quad (43)$$

The first term $|\mathbf{a}_i^H \mathbf{x}|$ is the clean modulus information, and the second term can be regarded as the additive noise, with

expectation

$$\mathbb{E} \left[\frac{n_i}{2|\mathbf{a}_i^H \mathbf{x}|} \right] = \frac{\mathbb{E}[n_i]}{2|\mathbf{a}_i^H \mathbf{x}|}, \quad \forall i \quad (44)$$

and variance

$$\text{Var} \left[\frac{n_i}{2|\mathbf{a}_i^H \mathbf{x}|} \right] = \frac{\text{Var}[n_i]}{4|\mathbf{a}_i^H \mathbf{x}|^2}, \quad \forall i. \quad (45)$$

Therefore, the additive noise to the clean modulus information $|\mathbf{a}_i^H \mathbf{x}|$ has smaller expectation value and variance value than the additive noise to the clean intensity information $|\mathbf{a}_i^H \mathbf{x}|^2$ when $|\mathbf{a}_i^H \mathbf{x}| > \frac{1}{2}$. In addition, the signal-to-noise ratio (SNR) of the modulus information

$$20 \log_{10} \frac{|\mathbf{a}_i^H \mathbf{x}|}{\frac{|n_i|}{2|\mathbf{a}_i^H \mathbf{x}|}} = 20 \log_{10} \frac{2|\mathbf{a}_i^H \mathbf{x}|^2}{|n_i|} \quad (46)$$

is larger than the SNR of the intensity measurement

$$20 \log_{10} \frac{|\mathbf{a}_i^H \mathbf{x}|^2}{|n_i|}. \quad (47)$$

APPENDIX B

PROOF OF $\lambda_{\max}((\mathbf{D}^{(k)})^H \mathbf{D}^{(k)}) \leq L$

Note that the dictionary $\mathbf{D}^{(k)} = [\mathbf{d}_1^{(k)}, \dots, \mathbf{d}_L^{(k)}]$ satisfies $\|\mathbf{d}_l^{(k)}\|_2 \leq 1, \forall l = 1, \dots, L$, so

$$\begin{aligned} \lambda_{\max}((\mathbf{D}^{(k)})^H \mathbf{D}^{(k)}) &= \lambda_{\max}(\mathbf{D}^{(k)} (\mathbf{D}^{(k)})^H) \\ &= \max_{\mathbf{t} \neq \mathbf{0}} \frac{\mathbf{t}^H \mathbf{D}^{(k)} (\mathbf{D}^{(k)})^H \mathbf{t}}{\|\mathbf{t}\|_2^2} = \max_{\mathbf{t} \neq \mathbf{0}} \sum_{l=1}^L \frac{|\mathbf{t}^H \mathbf{d}_l^{(k)}|^2}{\|\mathbf{t}\|_2^2} \\ &\leq \max_{\mathbf{t} \neq \mathbf{0}} \sum_{l=1}^L \frac{\|\mathbf{t}\|_2^2 \cdot \|\mathbf{d}_l^{(k)}\|_2^2}{\|\mathbf{t}\|_2^2} = \sum_{l=1}^L \|\mathbf{d}_l^{(k)}\|_2^2 \leq L. \end{aligned} \quad (48)$$

The equality is achieved when all of the vectors $\{\mathbf{d}_l^{(k)}\}$ lie on the same line and $\|\mathbf{d}_l^{(k)}\|_2 = 1, \forall l$.

APPENDIX C

PROOF OF PROPOSITION 2

The objective function in (5) is a mapping from \mathbb{C}^N to \mathbb{R} , which is not holomorphic and therefore not complex-differentiable. We find an equivalent problem in the field of real-valued numbers. Defining a new real-valued variable $\tilde{\mathbf{x}}$ as

$$\tilde{\mathbf{x}} := \begin{bmatrix} \text{Re}[\mathbf{x}] \\ \text{Im}[\mathbf{x}] \end{bmatrix} \in \mathbb{R}^{2N} \quad (49)$$

and constant symmetric matrices $\{\tilde{\mathbf{A}}_i\}_{i=1}^M$ as in (50) shown at the bottom of the next page, it is easy to verify that

$$\tilde{\mathbf{x}}^T \tilde{\mathbf{A}}_i \tilde{\mathbf{x}} = |\mathbf{a}_i^H \mathbf{x}|^2 \geq 0, \quad \forall i = 1, \dots, M. \quad (51)$$

Therefore, (5) is equivalent to the following problem in the field of real-valued numbers:

$$\underset{\tilde{\mathbf{x}} \in \mathbb{R}^{2N}}{\text{minimize}} \quad \sum_{i=1}^M \left(\sqrt{y_i} - \sqrt{\tilde{\mathbf{x}}^T \tilde{\mathbf{A}}_i \tilde{\mathbf{x}}} \right)^2 + \rho \sum_{i=1}^N \sqrt{[\tilde{\mathbf{x}}]_i^2 + [\tilde{\mathbf{x}}]_{i+N}^2}. \quad (52)$$

The corresponding surrogate problem (12) is equivalent to

$$\underset{\tilde{\mathbf{x}} \in \mathbb{R}^{2N}}{\text{minimize}} \quad C \|\tilde{\mathbf{x}} - \tilde{\mathbf{c}}\|_2^2 + \rho \sum_{i=1}^N \sqrt{[\tilde{\mathbf{x}}]_i^2 + [\tilde{\mathbf{x}}]_{i+N}^2}, \quad (53)$$

where $\tilde{\mathbf{c}} \in \mathbb{R}^{2N}$ is a constant vector defined as

$$\tilde{\mathbf{c}} := \tilde{\mathbf{x}}^{(k)} - \frac{1}{C} \sum_{i=1}^M \tilde{\mathbf{A}}_i \tilde{\mathbf{x}}^{(k)} \left(1 - \frac{\sqrt{y_i}}{\sqrt{(\tilde{\mathbf{x}}^{(k)})^T \tilde{\mathbf{A}}_i \tilde{\mathbf{x}}^{(k)}}} \right). \quad (54)$$

The objective function $\tilde{f}(\tilde{\mathbf{x}})$ in (52) can be majorized by the following function at any point $\tilde{\mathbf{x}}^{(k)}$:

$$\begin{aligned} \tilde{g}(\tilde{\mathbf{x}} | \tilde{\mathbf{x}}^{(k)}) &:= C \|\tilde{\mathbf{x}} - \tilde{\mathbf{c}}\|_2^2 + \rho \sum_{i=1}^N \sqrt{[\tilde{\mathbf{x}}]_i^2 + [\tilde{\mathbf{x}}]_{i+N}^2} \\ &- C \|\tilde{\mathbf{c}}\|_2^2 + \|\sqrt{\mathbf{y}}\|_2^2 + C \|\tilde{\mathbf{x}}^{(k)}\|_2^2 + \sum_{i=1}^M (\tilde{\mathbf{x}}^{(k)})^T \tilde{\mathbf{A}}_i \tilde{\mathbf{x}}^{(k)}. \end{aligned} \quad (55)$$

This majorization function $\tilde{g}(\tilde{\mathbf{x}} | \tilde{\mathbf{x}}^{(k)})$ is actually the objective function in (53) plus four constant terms. Letting $\tilde{\mathbf{x}}^{(k+1)}$ be the solution to the convex surrogate problem (53), the decent property is still maintained:

$$\tilde{f}(\tilde{\mathbf{x}}^{(k+1)}) \leq \tilde{g}(\tilde{\mathbf{x}}^{(k+1)} | \tilde{\mathbf{x}}^{(k)}) \leq \tilde{g}(\tilde{\mathbf{x}}^{(k)} | \tilde{\mathbf{x}}^{(k)}) = \tilde{f}(\tilde{\mathbf{x}}^{(k)}). \quad (56)$$

Since $\{\tilde{f}(\tilde{\mathbf{x}}^{(k)})\}$ is a non-increasing sequence and is lower-bounded by 0, it will converge to a stationary point. Assume there exists a subsequence $\{\tilde{\mathbf{x}}^{(k_l)}\}$ that converges to a limit point $\tilde{\mathbf{z}}$. Then

$$\begin{aligned} \tilde{g}(\tilde{\mathbf{x}}^{(k_l+1)} | \tilde{\mathbf{x}}^{(k_l+1)}) &= \tilde{f}(\tilde{\mathbf{x}}^{(k_l+1)}) \leq \tilde{f}(\tilde{\mathbf{x}}^{(k_l+1)}) \\ &\leq \tilde{g}(\tilde{\mathbf{x}}^{(k_l+1)} | \tilde{\mathbf{x}}^{(k_l)}) \leq \tilde{g}(\tilde{\mathbf{x}} | \tilde{\mathbf{x}}^{(k_l)}), \quad \forall \tilde{\mathbf{x}}. \end{aligned} \quad (57)$$

Letting $l \rightarrow +\infty$, we obtain

$$\tilde{g}(\tilde{\mathbf{z}} | \tilde{\mathbf{z}}) \leq \tilde{g}(\tilde{\mathbf{x}} | \tilde{\mathbf{z}}), \quad \forall \tilde{\mathbf{x}}, \quad (58)$$

which implies

$$\nabla \tilde{g}(\tilde{\mathbf{x}} | \tilde{\mathbf{z}}) \Big|_{\tilde{\mathbf{x}}=\tilde{\mathbf{z}}} = \mathbf{0}. \quad (59)$$

Furthermore, it is easy to verify the following equality at any point $\tilde{\mathbf{x}}^{(k)}$:

$$\nabla \tilde{g}(\tilde{\mathbf{x}} | \tilde{\mathbf{x}}^{(k)}) \Big|_{\tilde{\mathbf{x}}=\tilde{\mathbf{x}}^{(k)}} = \nabla \tilde{f}(\tilde{\mathbf{x}}) \Big|_{\tilde{\mathbf{x}}=\tilde{\mathbf{x}}^{(k)}}, \quad (60)$$

hence at the limiting point $\tilde{\mathbf{z}}$

$$\nabla \tilde{f}(\tilde{\mathbf{x}}) \Big|_{\tilde{\mathbf{x}}=\tilde{\mathbf{z}}} = \nabla \tilde{g}(\tilde{\mathbf{x}} | \tilde{\mathbf{z}}) \Big|_{\tilde{\mathbf{x}}=\tilde{\mathbf{z}}} = \mathbf{0}, \quad (61)$$

which implies that $\tilde{\mathbf{z}}$ is a stationary point of $\tilde{f}(\tilde{\mathbf{x}})$. Since (49) is an injective mapping, we can always project the real-valued variable $\tilde{\mathbf{x}}$ back to the complex-valued variable \mathbf{x} . Therefore, every limit point of the sequence generated by the C-PRIME algorithm is a stationary point of problem (5).

APPENDIX D

PROOF OF PROPOSITION 3

SC-PRIME fits the framework of the BSUM algorithm where an unified convergence analysis is provided in [35]. After rewriting the problem (17) in terms of real-valued quantities, we verify all the conditions of [35, Theorem 2 (a)].

Defining real-valued variables $\{\tilde{\mathbf{x}}_p\}$, $\{\tilde{\mathbf{z}}_p\}$, and $\tilde{\mathbf{D}}$ as

$$\tilde{\mathbf{x}}_p := \begin{bmatrix} \text{Re}[\mathbf{x}_p] \\ \text{Im}[\mathbf{x}_p] \end{bmatrix} \in \mathbb{R}^{2N}, \quad \forall p = 1, \dots, P, \quad (62)$$

$$\tilde{\mathbf{z}}_p := \begin{bmatrix} \text{Re}[\mathbf{z}_p] \\ \text{Im}[\mathbf{z}_p] \end{bmatrix} \in \mathbb{R}^{2L}, \quad \forall p = 1, \dots, P, \quad (63)$$

$$\tilde{\mathbf{D}} := \begin{bmatrix} \text{Re}[\mathbf{D}], & -\text{Im}[\mathbf{D}] \\ \text{Im}[\mathbf{D}], & \text{Re}[\mathbf{D}] \end{bmatrix} \in \mathbb{R}^{2N \times 2L}, \quad (64)$$

problem (17) is equivalent to

$$\begin{aligned} \underset{\{\tilde{\mathbf{x}}_p\}, \tilde{\mathbf{D}}, \{\tilde{\mathbf{z}}_p\}}{\text{minimize}} \quad & \sum_{p=1}^P \left[\sum_{i=1}^M \left(\sqrt{y_i} - \sqrt{\tilde{\mathbf{x}}_p^T \tilde{\mathbf{A}}_i \tilde{\mathbf{x}}_p} \right)^2 \right. \\ & \left. + \mu \left\| \tilde{\mathbf{x}}_p - \tilde{\mathbf{D}} \tilde{\mathbf{z}}_p \right\|_2^2 + \rho \sum_{i=1}^L \sqrt{[\tilde{\mathbf{z}}]_i^2 + [\tilde{\mathbf{z}}]_{i+L}^2} \right] \\ \text{subject to} \quad & \tilde{\mathbf{D}} \in \tilde{\mathcal{D}}, \end{aligned} \quad (65)$$

where $\tilde{\mathcal{D}}$ is a closed convex set defined as

$$\begin{aligned} \tilde{\mathcal{D}} := \left\{ \tilde{\mathbf{D}} \in \mathbb{R}^{2N \times 2L} \mid [\tilde{\mathbf{D}}]_{1:N,1:L} &= [\tilde{\mathbf{D}}]_{N+1:2N,L+1:2L}, \right. \\ & \left. [\tilde{\mathbf{D}}]_{N+1:2N,1:L} = -[\tilde{\mathbf{D}}]_{1:N,L+1:2L}, \|\tilde{\mathbf{d}}_l\|_2 \leq 1, \forall l \in [1, L] \right\}. \end{aligned} \quad (66)$$

A. Updating the Sparse Codes $\{\tilde{\mathbf{z}}_p\}$

Problem (19) is equivalent to

$$\underset{\tilde{\mathbf{z}}_p \in \mathbb{R}^{2L}}{\text{minimize}} \quad \mu \left\| \tilde{\mathbf{D}}^{(k)} \tilde{\mathbf{z}}_p - \tilde{\mathbf{x}}_p^{(k)} \right\|_2^2 + \rho \sum_{i=1}^L \sqrt{[\tilde{\mathbf{z}}]_i^2 + [\tilde{\mathbf{z}}]_{i+L}^2}. \quad (67)$$

The corresponding surrogate problem (24) is equivalent to

$$\underset{\tilde{\mathbf{z}}_p \in \mathbb{R}^{2L}}{\text{minimize}} \quad \mu E^{(k)} \|\tilde{\mathbf{z}}_p - \tilde{\mathbf{c}}_p\|_2^2 + \rho \sum_{i=1}^L \sqrt{[\tilde{\mathbf{z}}]_i^2 + [\tilde{\mathbf{z}}]_{i+L}^2}, \quad (68)$$

$$\tilde{\mathbf{A}}_i := \begin{bmatrix} \text{Re}[\mathbf{a}_i] \text{Re}^T[\mathbf{a}_i] + \text{Im}[\mathbf{a}_i] \text{Im}^T[\mathbf{a}_i], & \text{Re}[\mathbf{a}_i] \text{Im}^T[\mathbf{a}_i] - \text{Im}[\mathbf{a}_i] \text{Re}^T[\mathbf{a}_i] \\ \text{Im}[\mathbf{a}_i] \text{Re}^T[\mathbf{a}_i] - \text{Re}[\mathbf{a}_i] \text{Im}^T[\mathbf{a}_i], & \text{Re}[\mathbf{a}_i] \text{Re}^T[\mathbf{a}_i] + \text{Im}[\mathbf{a}_i] \text{Im}^T[\mathbf{a}_i] \end{bmatrix} \in \mathbb{R}^{2N \times 2N}, \quad \forall i = 1, \dots, M, \quad (50)$$

where $\tilde{\mathbf{e}}_p \in \mathbb{R}^{2L}$ is a constant vector defined as

$$\tilde{\mathbf{e}}_p := \tilde{\mathbf{z}}_p^{(k)} - \frac{1}{E^{(k)}} \left(\tilde{\mathbf{D}}^{(k)} \right)^T \left(\tilde{\mathbf{D}}^{(k)} \tilde{\mathbf{z}}_p^{(k)} - \tilde{\mathbf{x}}_p^{(k)} \right). \quad (69)$$

B. Updating the Estimated Signals $\{\tilde{\mathbf{x}}_p\}$

Problem (27) is equivalent to

$$\underset{\tilde{\mathbf{x}}_p \in \mathbb{R}^{2N}}{\text{minimize}} \sum_{i=1}^M \left(\sqrt{y_i} - \sqrt{\tilde{\mathbf{x}}_p^T \tilde{\mathbf{A}}_i \tilde{\mathbf{x}}_p} \right)^2 + \mu \left\| \tilde{\mathbf{x}}_p - \tilde{\mathbf{D}}^{(k)} \tilde{\mathbf{z}}_p^{(k+1)} \right\|_2^2. \quad (70)$$

The corresponding surrogate problem (30) is equivalent to

$$\underset{\tilde{\mathbf{x}}_p \in \mathbb{R}^{2N}}{\text{minimize}} (F + \mu) \left\| \tilde{\mathbf{x}}_p - \tilde{\mathbf{f}}_p \right\|_2^2, \quad (71)$$

where $\tilde{\mathbf{f}}_p \in \mathbb{R}^{2N}$ is a constant vector defined as

$$\tilde{\mathbf{f}}_p := \frac{1}{F + \mu} \left[F \tilde{\mathbf{x}}_p^{(k)} - \sum_{i=1}^M \tilde{\mathbf{A}}_i \tilde{\mathbf{x}}_p^{(k)} \left(1 - \frac{\sqrt{y_i}}{\sqrt{(\tilde{\mathbf{x}}_p^{(k)})^T \tilde{\mathbf{A}}_i \tilde{\mathbf{x}}_p^{(k)}}} \right) + \mu \tilde{\mathbf{D}}^{(k)} \tilde{\mathbf{z}}_p^{(k+1)} \right]. \quad (72)$$

C. Updating the Dictionary $\tilde{\mathbf{D}}$

Problem (33) is equivalent to

$$\underset{\tilde{\mathbf{D}} \in \mathbb{R}^{2N \times 2L}}{\text{minimize}} \left\| \tilde{\mathbf{X}}^{(k+1)} - \tilde{\mathbf{D}} \tilde{\mathbf{Z}}^{(k+1)} \right\|_F^2$$

subject to $\tilde{\mathbf{D}} \in \tilde{\mathcal{D}}, \quad (73)$

where $\tilde{\mathbf{X}}^{(k+1)} := [\tilde{\mathbf{x}}_1^{(k+1)}, \dots, \tilde{\mathbf{x}}_p^{(k+1)}] \in \mathbb{R}^{2N \times P}$ and $\tilde{\mathbf{Z}}^{(k+1)} := [\tilde{\mathbf{z}}_1^{(k+1)}, \dots, \tilde{\mathbf{z}}_p^{(k+1)}] \in \mathbb{R}^{2L \times P}$. Because of the special structure of $\tilde{\mathbf{D}}$, it is sufficient to find the first L columns $\{\tilde{\mathbf{d}}_l\}_{l=1}^L$. To update $\tilde{\mathbf{d}}_l$, problem (37) is equivalent to

$$\underset{\tilde{\mathbf{d}}_l \in \mathbb{R}^{2N}}{\text{minimize}} \left\| \tilde{\mathbf{z}}_{l,T}^{(k+1)} \right\|_2^2 \cdot \left\| \tilde{\mathbf{d}}_l - \tilde{\mathbf{g}}_l \right\|_2^2$$

subject to $\left\| \tilde{\mathbf{d}}_l \right\|_2 \leq 1, \quad (74)$

where $\tilde{\mathbf{g}}_l \in \mathbb{R}^{2N}$ is a constant vector defined as

$$\tilde{\mathbf{g}}_l := \tilde{\mathbf{d}}_l^{(k)} + \frac{1}{\left\| \tilde{\mathbf{z}}_{l,T}^{(k+1)} \right\|_2^2} \left(\tilde{\mathbf{X}}^{(k+1)} - \tilde{\mathbf{D}}^{(k)} \tilde{\mathbf{Z}}^{(k+1)} \right) \left(\tilde{\mathbf{z}}_{l,T}^{(k+1)} \right)^T, \quad (75)$$

and $\tilde{\mathbf{z}}_{l,T}^{(k+1)}$ is a row vector denoting the l -th row of $\tilde{\mathbf{Z}}^{(k+1)}$.

Now the objective function in (68) (plus some constants) is a valid majorization function of the objective function in (67) at any point $\tilde{\mathbf{z}}_p^{(k)}$; the objective function in (71) (plus some constants) is a valid majorization function of the objective function in (70) at any point $\tilde{\mathbf{x}}_p^{(k)}$; and the objective function in (74) (plus some constants) is exact the same as the objective function in (73) treating only one column $\tilde{\mathbf{d}}_l$ as the variable. It is easy to check that [35, Assumption 2] is satisfied. Moreover, all the three objective functions in (68), (71), and (74) are convex with regard to their corresponding variables. Problems (68),

(71), and (74) all have a unique solution. Hence, according to [35, Theorem 2], every limit point of the sequence generated by the SC-PRIME algorithm is a coordinate-wise stationary point of problem (65). Therefore, every limit point of the sequence generated by the SC-PRIME algorithm is a coordinate-wise stationary point of problem (17) if we project all the problems to their corresponding equivalent problems in the field of complex-valued numbers.

REFERENCES

- [1] A. Walther, "The question of phase retrieval in optics," *Opt. Acta, Int. J. Opt.*, vol. 10, no. 1, pp. 41–49, 1963.
- [2] C. Fienup and J. Dainty, "Phase retrieval and image reconstruction for astronomy," in *Image Recovery: Theory and Application*. New York, NY, USA: Academic, 1987, pp. 231–275.
- [3] R. W. Harrison, "Phase problem in crystallography," *J. Opt. Soc. Amer. A*, vol. 10, no. 5, pp. 1046–1055, May 1993.
- [4] J. Miao, T. Ishikawa, B. Johnson, E. H. Anderson, B. Lai, and K. O. Hodgson, "High resolution 3D X-ray diffraction microscopy," *Physical Rev. Lett.*, vol. 89, Aug. 2002, Art. no. 088303.
- [5] T. Gerkmann, M. Krawczyk-Becker, and J. Le Roux, "Phase processing for single-channel speech enhancement: History and recent advances," *IEEE Signal Process. Mag.*, vol. 32, no. 2, pp. 55–66, Mar. 2015.
- [6] Y. Shechtman, Y. Eldar, O. Cohen, H. 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, May 2015.
- [7] 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.
- [8] B. G. Bodmann and N. Hammen, "Stable phase retrieval with low-redundancy frames," *Adv. Comput. Math.*, vol. 41, no. 2, pp. 317–331, 2015.
- [9] A. Conca, D. Edidin, M. Hering, and C. Vinzant, "An algebraic characterization of injectivity in phase retrieval," *Appl. Comput. Harmon. Anal.*, vol. 38, no. 2, pp. 346–356, 2015.
- [10] 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.
- [11] C. Vinzant, "A small frame and a certificate of its injectivity," in *Proc. 2015 Int. Conf. Sampling Theory Appl.*, May 2015, pp. 197–200.
- [12] T. Qiu, P. Babu, and D. P. Palomar, "PRIME: Phase retrieval via majorization-minimization," *IEEE Trans. Signal Process.*, vol. 64, no. 19, pp. 5174–5186, Oct. 2016.
- [13] R. Balan, P. Casazza, and D. Edidin, "On signal reconstruction without phase," *Appl. Comput. Harmon. Anal.*, vol. 20, no. 3, pp. 345–356, 2006.
- [14] 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, May 2015.
- [15] V. Voroninski and Z. Xu, "A strong restricted isometry property, with an application to phaseless compressed sensing," *Appl. Comput. Harmon. Anal.*, vol. 40, no. 2, pp. 386–395, 2016.
- [16] X. Li and V. Voroninski, "Sparse signal recovery from quadratic measurements via convex programming," *SIAM J. Math. Anal.*, vol. 45, no. 5, pp. 3019–3033, 2013.
- [17] Y. Shechtman, A. Beck, and Y. Eldar, "GESPAR: Efficient phase retrieval of sparse signals," *IEEE Trans. Signal Process.*, vol. 62, no. 4, pp. 928–938, Feb. 2014.
- [18] S. Mukherjee and C. S. Seelamantula, "Fienup algorithm with sparsity constraints: Application to frequency-domain optical-coherence tomography," *IEEE Trans. Signal Process.*, vol. 62, no. 18, pp. 4659–4672, Sep. 2014.
- [19] D. Weller, A. Pnueli, G. Divon, O. Radzyner, Y. Eldar, and J. Fessler, "Undersampled phase retrieval with outliers," *IEEE Trans. Comput. Imag.*, vol. 1, no. 4, pp. 247–258, Dec. 2015.
- [20] M. L. Moravec, J. K. Romberg, and R. G. Baraniuk, "Compressive phase retrieval," *Proc. SPIE*, vol. 6701, 2007, Art. no. 670120.
- [21] J. R. Fienup, "Phase retrieval algorithms: A comparison," *Appl. Opt.*, vol. 21, no. 15, pp. 2758–2769, 1982.

- [22] H. Ohlsson, A. Yang, R. Dong, and S. Sastry, "CPRL—An extension of compressive sensing to the phase retrieval problem," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Eds. Dutchess County, NY, USA: Curran Associates, Inc., 2012, pp. 1367–1375.
- [23] P. Schniter and S. Rangan, "Compressive phase retrieval via generalized approximate message passing," *IEEE Trans. Signal Process.*, vol. 63, no. 4, pp. 1043–1055, Feb. 2015.
- [24] A. M. Tillmann, Y. C. Eldar, and J. Mairal, "DOLPHIn—Dictionary learning for phase retrieval," *IEEE Trans. Signal Process.*, vol. 64, no. 24, pp. 6485–6500, Dec. 2016.
- [25] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.
- [26] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.
- [27] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *J. Mach. Learn. Res.*, vol. 11, pp. 19–60, Mar. 2010.
- [28] J. Mairal, F. Bach, and J. Ponce, "Sparse modeling for image and vision processing," *Found. Trends Comput. Graph. Vis.*, vol. 8, nos. 2/3, pp. 85–283, 2014.
- [29] D. R. Hunter and K. Lange, "A tutorial on MM algorithms," *Amer. Statist.*, vol. 58, no. 1, pp. 30–37, 2004.
- [30] Y. Sun, P. Babu, and D. P. Palomar, "Majorization-minimization algorithms in signal processing, communications, and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 3, pp. 794–816, Feb. 2017.
- [31] G. Wang, G. B. Giannakis, and Y. C. Eldar, "Solving systems of random quadratic equations via truncated amplitude flow," 2016. arXiv: 1605.08285.
- [32] 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, Apr. 2017.
- [33] R. Varadhan and C. Roland, "Simple and globally convergent methods for accelerating the convergence of any EM algorithm," *Scand. J. Statist.*, vol. 35, no. 2, pp. 335–353, 2008.
- [34] M. Hong, M. Razaviyayn, Z. Q. Luo, and J. S. Pang, "A unified algorithmic framework for block-structured optimization involving big data: With applications in machine learning and signal processing," *IEEE Signal Process. Mag.*, vol. 33, no. 1, pp. 57–77, Jan. 2016.
- [35] M. Razaviyayn, M. Hong, and Z.-Q. Luo, "A unified convergence analysis of block successive minimization methods for nonsmooth optimization," *SIAM J. Optim.*, vol. 23, no. 2, pp. 1126–1153, 2013.

Tianyu Qiu received the B.Eng. degree in electronic and information from the Huazhong University of Science and Technology, Wuhan, China, in 2013. He is currently working toward the Ph.D. degree in electronic and computer engineering at The Hong Kong University of Science and Technology, Hong Kong. His research interests include statistical signal processing, mathematical optimization, and machine learning.



Daniel P. Palomar (S'99–M'03–SM'08–F'12) received the Electrical Engineering and Ph.D. degrees from the Technical University of Catalonia (UPC), Barcelona, Spain, in 1998 and 2003, respectively.

Since 2006, he has been at The Hong Kong University of Science and Technology (HKUST), Hong Kong, where he is currently a Professor in the Department of Electronic and Computer Engineering. Since 2013, he is a Fellow of the Institute for Advance Study, HKUST. He has previously held several research appointments, namely, at King's College

London, London, UK; Stanford University, Stanford, CA, USA; Telecommunications Technological Center of Catalonia, Barcelona, Spain; Royal Institute of Technology (KTH), Stockholm, Sweden; University of Rome "La Sapienza," Rome, Italy; and Princeton University, Princeton, NJ, USA. His research interests include applications of convex optimization theory, game theory, and variational inequality theory to financial systems, big data systems, and communication systems.

Dr. Palomar received the 2004/06 Fulbright Research Fellowship, the 2004 and 2015 (coauthor) Young Author Best Paper Awards by the IEEE Signal Processing Society, the 2015–2016 HKUST Excellence Research Award, the 2002/03 Best Ph.D. prize in information technologies and communications by UPC, the 2002/03 Rosina Ribalta first prize for the Best Doctoral Thesis in information technologies and communications by the Epsilon Foundation, and the 2004 prize for the Best Doctoral Thesis in advanced mobile communications by the Vodafone Foundation and COIT. He was a Guest Editor of the IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING 2016 special issue on "Financial Signal Processing and Machine Learning for Electronic Trading," an Associate Editor of the IEEE TRANSACTIONS ON INFORMATION THEORY, and of the IEEE TRANSACTIONS ON SIGNAL PROCESSING, a Guest Editor of the IEEE SIGNAL PROCESSING MAGAZINE 2010 special issue on "Convex Optimization for Signal Processing," the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS 2008 special issue on "Game Theory in Communication Systems," and the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS 2007 special issue on "Optimization of MIMO Transceivers for Realistic Communication Networks."