# Phase retrieval: a data-driven wavelet frame based approach

Tongyao Pang[a], Qingna Li[b,1], Zaiwen Wen[c,2,*], Zuowei Shen[a]

[a]*Department of Mathematics, National University of Singapore, Singapore*
[b]*School of Mathematics and Statistics, Beijing Institute of Technology, China*
[c]*Beijing International Center for Mathematical Research, Peking University, China*

**Abstract**

In this paper, we consider the phase retrieval problem for recovering a complex signal, given a number of observations on the magnitude of linear measurements. This problem has direct applications in X-ray crystallography, diffraction imaging and microscopy. Motivated by the extensively studied theory of (tight) wavelet frame and its great success in various applications, we propose a wavelet frame based model for phase retrieval using the balanced approach. A hybrid fidelity term is designed to deal with complicated noises and a hybrid penalty term is constructed for different pursuits of sparsity and smoothness. Consequently, a proximal alternating linearization algorithm is developed and its convergence is analyzed. In particular, our proposed algorithm updates both the internal weights in the hybrid penalty term and the penalty parameter balancing the fidelity and penalty terms in a data-driven way. Extensive numerical experiments show that our method is quite competitive with other existing algorithms. On one hand, our method can reconstruct the truth successfully from a small number of measurements even if the phase retrieval problem is ill-posed. On the other hand, our algorithm is very robust to different types of noise, including Gaussian noise, Poisson noise and their mixtures.

*Keywords:* phase retrieval, wavelet frame, hybrid fidelity term, hybrid penalty term, proximal alternating linearization, data-driven, complicated noise.

## 1. Introduction

Given a few observations on the magnitude of linear measurements of a complex signal $x \in \mathbb{C}^n$, one of the typical formulations of the phase retrieval

*Corresponding author
*Email addresses:* pangtongyao@u.nus.edu (Tongyao Pang), qnl@bit.edu.cn (Qingna Li), wenzw@math.pku.edu.cn ( Zaiwen Wen), matzuows@nus.edu.sg ( Zuowei Shen)
[1]This research was supported by NSFC grants 11671036.
[2]This research was supported in part by NSFC grants 11322109 and 91330202.

problem tries to identify the true signal $x$ by solving

$$\begin{aligned} \text{find} \quad & x \in \mathbb{C}^n \\ \text{s.t.} \quad & |Ax| = b, \end{aligned} \tag{1}$$

where $A \in \mathbb{C}^{m\times n}$, $b \in \mathbb{R}^m$. This problem has direct applications in X-ray crystallography [1], diffraction imaging [2] and microscopy [3]. In this paper, we mainly consider the physically inspired diffraction model: *coded diffraction pattern* (CDP) in [4]. In coded diffraction model, the measurements are of the form

$$b = |Ax|, \quad Ax = \begin{pmatrix} \mathcal{F}(I_1 \circ x) \\ \vdots \\ \mathcal{F}(I_m \circ x) \end{pmatrix}, \tag{2}$$

where $\mathcal{F}$ denotes the DFT matrix, $\circ$ is the Hadamard product, $I_1, \ldots, I_m$ are the illumination masks (also called *coded diffreaction patterns*). Since the phase information is missing, this challenging inverse problem is ill-posed in general. It is also nonconvex due to the amplitude constraints $|Ax| = b$.

Problem (1) is equivalent to identifying a phase variable $y$ and the signal variable $x$ simultaneously as

$$\begin{aligned} \min_{x\in\mathbb{C}^n, y\in\mathbb{C}^m} \quad & \|Ax - y\|_2^2 \\ \text{s.t.} \quad & |y| = b. \end{aligned} \tag{3}$$

Existing methods for (1) as well as (3) can roughly be summarized into four types: the greedy methods, the nonlinear least squares based methods, the convex relaxation based methods and the sparsity regularized methods. In the following, we give a brief review on them.

### *1.1. Existing Methods for Phase Retrieval*

Typical greedy methods for (1) and (3) include the Error-Reduction (ER) or alternating projection algorithm by Gerchberg and Saxton [5], the hybrid input-output (HIO) algorithm by Fienup [6], the hybrid projection-rejection method (HPR) by Bauschke et al. [7, 8] and so on. The ER method minimizes the quadratic error $\|Ax-y\|_2^2$ with respect to $x$ using the orthogonal projector $AA^\dagger$ ($A^\dagger$ is the pseudo-inverse of $A$), then normalizes the module of $y$ by adjusting the amplitude of each coordinate to $b$ and keeps the phase unchanged at the same time. If there is an additional object-domain constraint that the signal should be supported on a given set $D$, the HIO method improves the convergence by slightly changing the output of the ER method:

$$\begin{aligned} y_i^{k+1} &= b_i \frac{(Ax^k)_i}{|(Ax^k)_i|}, \qquad i = 1, \ldots, m, \\ x_i^{k+1} &= \begin{cases} \big(A^\dagger y^{k+1}\big)_i, & \text{if } i \in D \\ x_i^k - \beta\big(A^\dagger y^{k+1}\big)_i, & \text{otherwise} \end{cases}, \end{aligned} \tag{HIO}$$

where $\beta$ is positive. These methods are variants of the alternating projection on the range of $A$ and on the nonconvex set of $y$ such that $|y| = |Ax|$. They may have a good empirical performance but may get stuck in local minimum. The alternating direction methods of multipliers (ADMM) is used in [9] to solve both the classical and ptychographic phase retrieval problems. It is shown that the projection algorithms such as HIO are equivalent to ADMM with suitable parameters.

Another type of methods is based on minimizing the nonlinear least squares model:

$$\min_{x\in\mathbb{C}^n} \quad f(x) := \frac{1}{2m}\sum_{i=1}^{m}(b_i^2 - |a_i^* x|^2)^2, \tag{4}$$

where $a_i^*$ is the $i$-th row of matrix $A$. In [10], a Wirtinger flow (WirtingerFlow) method starts from an initial point $x^0$ obtained by a spectral method, then updates the next iteration using a gradient descent type scheme:

$$x^{k+1} = x^k - \frac{\mu^k}{\|x^0\|}\nabla f(x^k), \tag{WirtignerFlow}$$

where $\frac{\mu^k}{\|x^0\|}$ can be interpreted as a step size. Exact retrieval of phase information from a nearly minimal number of random measurements is proved. It is also shown that the Wirtinger flow algorithm converges to the global minimizer at a global linear rate. Ma, Liu and Wen [11] proposed a Levenberg-Marquadt method to solve (4) and the global linear and local quadratic convergence are achieved under certain conditions. For other methods such as ADMM by Wen et. al. [9] and the damped Gauss-Newton method by Yoav Shechtman et. al. in [12], the global convergence is not clear yet.

The PhaseLift and the PhaseCut method are two typical convex relaxation based approaches. Observing that $|Ax|^2$ is a linear function of $X = xx^*$, PhaseLift is proposed in [13] and [14] by dropping the rank one constraint and using a "lifting" technique which relaxes the original problem into an SDP (Semidefinite Programming) problem:

$$\begin{aligned} \min_{X\in S^n} \quad & Tr(X) \\ \text{s.t.} \quad & Tr(a_i a_i^* X) = b_i^2, i = 1,\ldots,m, \\ & X \succeq 0, \end{aligned} \tag{PhaseLift}$$

where $\mathcal{S}^n$ denotes the set of symmetric matrices of dimension $n$. If the SDP problem admits a rank-one solution, the relaxation is tight. Otherwise, the normalized leading eigenvector of the solution is chosen as an approximate solution. It has been shown in [15, 4, 14] that under certain conditions, the SDP problem (PhaseLift) can recover $x$ with high probabilities. Different from PhaseLift, the PhaseCut method [16, 17] proposes a semidefinite relaxation by eliminating the signal variable other than the phase variable. First, problem (3) is reformulated

as:

$$\begin{aligned} \min_{x\in\mathbb{C}^n, u\in\mathbb{C}^m} & \frac{1}{2}\|Ax - \mathrm{diag}(b)u\|_2^2 \\ \text{s.t.} & \quad |u_i| = 1, i = 1, \ldots, m. \end{aligned} \tag{5}$$

Given the phase variable $u$, the signal variable is $x = A^\dagger \,\mathrm{diag}\{b\}u$. Then (5) becomes

$$\begin{aligned} \min_{u\in\mathbb{C}^m} & \quad u^* M u \\ \text{s.t.} & \quad |u_i| = 1, i = 1, \ldots, m, \end{aligned} \tag{6}$$

where $M = \mathrm{diag}\{b\}(I - AA^\dagger)\,\mathrm{diag}\{b\}$ is positive semidefinite. Problem (6) is an extension of the MAXCUT problem over the unit complex torus and its SDP relaxation is

$$\begin{aligned} \min_{U\in S^m} & \quad Tr(UM) \\ \text{s.t.} & \quad U_{ii} = 1,\ i = 1, \cdots, m,\ U \succeq 0. \end{aligned} \tag{PhaseCut}$$

It is then solved by a provably convergent block coordinate descent algorithm. Other convex approaches include [18, 19, 20].

Recently, the advances in compressed sensing [21] has inspired a few sparsity regularized phase retrieval methods [22, 23, 24]. Chang et al. [24] first formulated the phase retrieval problem as a least square problem with a box constraint and a total variation (TV) regularization due to its efficiency to recover signals or images from incomplete data or limited measurements:

$$\min_{0<u<1} \mathscr{E}_{TVB}(u) = TV(u) + \frac{\lambda}{2}\|\,|Au| - b\|_2^2, \tag{TVB}$$

where $TV(u)$ denotes the discrete total variation semi-norm, i.e., the $\ell_1$ norm of the discrete gradient of $u$. Then, an ADMM method was employed to solve (TVB) in [24].

### *1.2. Wavelet Based Approaches*

During the last ten years, we have witnessed the successful applications of wavelet-based approaches in many areas, such as in image processing, electron microscopy [25, 26, 27, 28]. The wavelet frame approach assumes that the signal we are interested in, especially images, has a sparse approximation under the wavelet tight frames. For a given signal $u \in \mathbb{R}^n$ and its measurements $g = Au$ with $A \in \mathbb{R}^{m\times n}$, $g \in \mathbb{R}^n$, let $W \in \mathbb{R}^{L\times n}$ (usually $L \gg n$) denote the analysis operator associated with some wavelet tight frame system such that $W^\top W = I$. The corresponding synthesis operator is then its transpose $W^T$. There are three types of wavelet tight frame based approaches: synthesis approach, analysis approach and balanced approach. All these three types of approximation become equivalent when using an orthonormal wavelet basis, but different when using a redundant wavelet frame. Both the analysis and synthesis based approaches have their favorable data sets and applications. In this paper, we only consider the balanced approach because it balances the advantages

and disadvantages of the other two methods. Interested readers are referred to [29] and [30] for more details. The balanced approach recovers $u$ from linear measurements $g = Au$ by solving

$$\bar{u} = W^T\bar{\alpha}, \text{ where } \bar{\alpha} = \arg\min_{\alpha\in\mathbb{R}^L} \frac{1}{2}\|AW^T\alpha - g\|_2^2 + \frac{\kappa}{2}\|(I - WW^T)\alpha\|_2^2 + \|\lambda\circ\alpha\|_1 \tag{7}$$

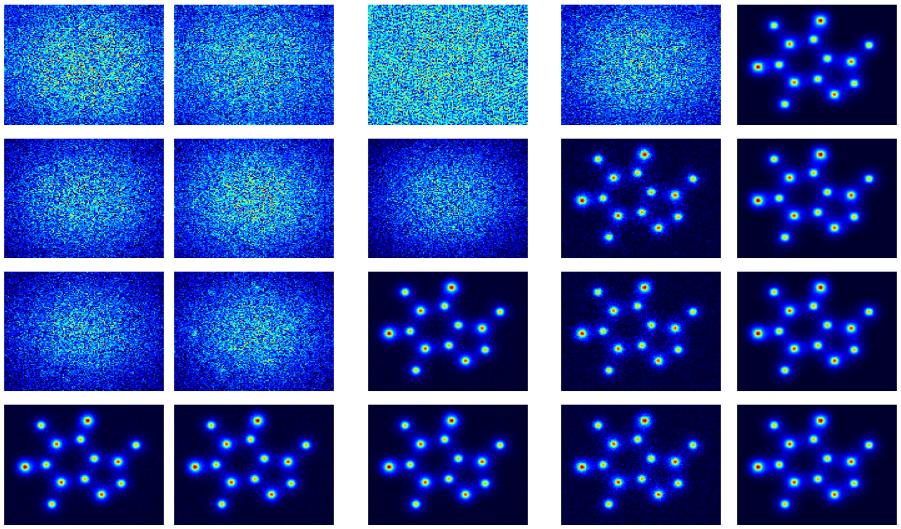
for some $0 \leq \kappa \leq \infty$, and $\lambda \in \mathbb{R}_+^L$. In fact, the so called synthesis and analysis approach is the model (7) with $\kappa = 0$ and $\kappa = \infty$, respectively. In the literature, the variables included in the wavelet model are usually real-valued while complex intermediate variables may appear in phase retrieval problems. However, wavelet methods still apply as long as the ground truth enjoys some smoothness properties and has sparse wavelet coefficients, which is true for most applications.

### *1.3. Our Approach*

Based on (7), we derive our wavelet frame based model for phase retrieval problem. Here, we only provide a first glance of our model while a more detailed discussion is presented in section 3:

$$\begin{aligned}\min_{\alpha,y,\lambda}\ & \frac{1}{2}\|AW^T\alpha - y\|_2^2 + \frac{\kappa}{2}\|(I - WW^T)\alpha\|_2^2 + \sum_{i=1}^{L}\rho_1(\lambda_i)|\alpha_i| + \sum_{i=1}^{L}\rho_2(\lambda_i)|\alpha_i|^2\\ \text{s.t. } & \omega_1\|\,|y| - b\|_1 + \omega_2\|\,|y| - b\|_2^2 \leq \epsilon,\\ & \alpha\in\mathbb{C}^L, y\in\mathbb{C}^m, \lambda\in\mathbb{R}_+^L\end{aligned} \tag{8}$$

where $\rho_1(\cdot)$ and $\rho_2(\cdot)$ are two smooth functions. Similar to (3), our model (8) separates the phase variable $y$ and the signal variable $x$ and solves $x$ using wavelet frame based method. The different part is that the amplitude constraint in (3) is relaxed in our model because we take the noises on the measurements into consideration. Moreover, the hybrid fidelity term is introduced due to its effectiveness of removing mixed and unknown noises. We also adopt $\ell_1$-norm and $\ell_2$-norm hybrid penalty term in (8). It is known that $\ell_1$-norm penalization pursues sparsity and protects singularities while $\ell_2$-norm penalization averages the wavelet coefficients and keeps smoothness. Hence, the hybrid penalty term can keep the edges sharp and the other parts smooth simultaneously. In addition, once the patterns of $\rho_1(\cdot)$ and $\rho_2(\cdot)$ are given, we can develop a modified proximal alternating linearization method (PALM) to solve our model iteratively. In particular, the variables $\lambda$ which balances the $\ell_1$-norm and $\ell_2$-norm in the hybrid penalty term is updated in a data driven way, which helps us to learn the most suitable penalty terms automatically during the iterations. A data-driven update is also designed for the parameter that balances the fidelity and penalty term. These are the reasons why we refer to our method, including both the model and algorithm, as a data-driven wavelet frame based method. Our method is indeed competitive numerically to other methods mentioned in subsection 1.1. The recovered density maps of caffeine molecule using different



(a) HIO (b) PhaseCut (c) WirtingerFlow (d) TVB (e) our method

Figure 1: Recovered density maps of caffeine molecule with Gaussian noise ($\alpha$ =1E-3). The rows from top to bottom correspond to one to four bipolar masks, respectively.

numbers of bipolar masks under Gaussian noise are depicted in subsection 1.3. It shows that our method reconstructs the molecule's density map successfully even if there is only one mask and the measurements are contaminated by Gaussian noise. More experiments can be found in section 5.

### *1.4. Organization*

The organization of the rest of the paper is as follows. In section 2, some preliminaries on the tight wavelet frame and the Wirtinger derivatives are described. In section 3, we develop our wavelet frame based model in detail and propose a proximal alternating linearization algorithm. The convergence of the PALM algorithm is also provided in this section and its detailed proof is given in section 7. We next clarify the implementation details of our algorithms and the parameter settings in section 4. Numerical results are reported in section 5 to illustrate the effectiveness of the proposed algorithms. Some final conclusions are made in section 6.

### *1.5. Notations*

We use a standard linear algebra notation whenever possible to describe quantities required in the algorithms. Let $\mathbb{R}^n_+$ denote the nonnegative orthant. For $x \in \mathbb{C}^n$, we write $\mathrm{diag}\{x\}$ the square matrix with diagonal $x$. $a \circ b$ is the Hadamard (or componentwrise) product of vectors $a$ and $b$. $\bar{z}$ is the conjugate of $z$. $X^T$ is the transpose of $X$ while $X^*$ is the Hermitian transpose. The notation $\|x\|_1$ is the $\ell_1$ norm of vector $x \in \mathbb{C}^n$, while $|x| \in \mathbb{R}^n$ is the moduli vector whose $i$th component is $|x_i|$. The notation $\frac{x}{|x|}$ refers to the entrywise division with

$\frac{0}{0} := 1$. For $x, y \in \mathbb{C}^n$, $\langle x, y\rangle := \text{real}(x^* y)$. For vectors, we use superscript to denote the iteration number and subscript to denote the entries of each vector.

## 2. Preliminaries

In this section, we give some preliminaries on wavelet tight frames and Wirtinger derivatives.

### *2.1. Wavelet tight frames*

We first present here some basics of a tight frame in a Hilbert space $\mathcal{H}$. Interested readers are referred to [30, 31] for more details. Let $\|\cdot\|$ denote the norm of the Hilbert space $\mathcal{H}$. A sequence $\{\phi_n\}_{n\in\mathcal{L}} \subset \mathcal{H}$ is a tight frame for $\mathcal{H}$ if

$$\|f\|_2^2 = \sum_{n\in\mathcal{L}} |\langle f, \phi_n\rangle|^2, \quad \forall f \in \mathcal{H}.$$

For a given sequence $\{\phi_n\}_{n\in\mathcal{L}}$, the analysis operator and its adjoint operator $\mathcal{W}^*$, which is also called the synthesis operator, are defined by

$$\mathcal{W}: \ f \in \mathcal{H} \longrightarrow \{\langle f, \phi_n\rangle\} \in \ell^2(\mathbb{N}), \quad \mathcal{W}^*: \ \{a_n\} \in \ell^2(\mathbb{N}) \longrightarrow \sum_{n\in\mathcal{L}} a_n \phi_n \in \mathcal{H}.$$

Thus, a sequence $\{\phi_n\} \subset \mathcal{H}$ forms a tight frame if and only if $\mathcal{W}^*\mathcal{W} = I$, where $I : \mathcal{H} \to \mathcal{H}$ is the identity operator. A tight frame is a generalization of orthonormal basis in the sense that both have the same perfect reconstruction property:

$$f = \sum_{n\in\mathcal{L}} \langle f, \phi_n\rangle \phi_n, \quad \forall f \in \mathcal{H}.$$

In the discrete setting, a tight frame consists of a finite number of masks $\{\phi_n\}_{n=1}^L$ with their support on $\{1, 2, \ldots, N\}$. Thus the associated analysis operator and synthesis operator can be represented in a matrix form. Let matrix $W$ denote the analysis operator, then the corresponding synthesis operator is its transpose $W^T$. Clearly, $\{\phi_n\}_{n=1}^L$ forms a tight frame for $\mathbb{R}^N$ if and only if $W^T W = I_N$, where $I_N$ is the $N$-by-$N$ identity matrix.

One widely used tight frame in signal/image processing is the multi-resolution analysis (MRA) based wavelet tight frame. The wavelet tight frame for $\mathbb{C}^N$ can be constructed from the masks associated with some MRA-based framelets for the space of continuum (see [30] for more details). For simplicity, only single-level un-decimal wavelet tight frame system for $\mathbb{C}^N$ is introduced. For a given tight frame system, suppose that $\{a_0, a_1, \cdots, a_l\}$ are the associated masks with finite supports. For each mask $a_n$ ($n \in \{0, 1, \ldots, l\}$), we define an $N$-by-$N$ matrix $\mathcal{S}_{a_n}$ as the symmetric Toeplitz matrix with $a_n$ being the first row. Then the corresponding analysis operator $W$ takes the form of

$$[\mathcal{S}_{a_0}, \mathcal{S}_{a_1}, \cdots, \mathcal{S}_{a_l}], \tag{9}$$

and its transpose $W^T$ is the synthesis operator of the tight frame system. The perfect reconstruction property in matrix representation can be expressed as $W^T W = I_N$.

For 2-dimensional images, the tight frame analysis and synthesis operations can be obtained by tensor products of the corresponding operations on the x-axis and y-axis. Interested readers are referred to [27] for more details. For an easy notation, we vectorize 2-D images to 1-D signals formally in the following parts of this paper, although there is no vectorization in the realistic implementation of framelet operations.

In particular, one can construct tight framelet system from B-splines. In our simulation, we choose the piecewise linear framelet for simplicity of the masks and speed, or piecewise cubic when the smoothness of the framelet is desirable. The construction of tight framelets from B-splines of high orders can be found in [32].

*2.2. Wirtinger derivatives*

In section 1, the model (8) gives an objective function that is a real valued function over complex variables. We next briefly introduce the Wirtinger derivative as a tool to deal with such functions (See [33] for more details.). A real-valued function $g(z) = g(x + iy)$ is said to be *Wirtinger differentiable* if it is differentiable as a function of real variables $x$ and $y$, and its *Wirtinger derivatives* under the conjugate coordinates $\begin{bmatrix} z \\ \bar{z} \end{bmatrix} \in \mathbb{C}^d \times \mathbb{C}^d$ is given by $\nabla^w g = \begin{bmatrix} \partial_z g \\ \partial_{\bar{z}} g \end{bmatrix}$, where

$$\partial_z g := \frac{\partial g\big([z, \bar{z}]^T\big)}{\partial z}\Big|_{\bar{z}=\text{constant}}, \quad \partial_{\bar{z}} g := \frac{\partial g\big([z, \bar{z}]^T\big)}{\partial \bar{z}}\Big|_{z=\text{constant}}.$$

The Taylor's expansion of $g$ under the conjugate coordinates and Wirtinger calculus takes the form

$$g(z + \Delta z) = g(z) + \big(\nabla^w g(z)\big)^T \begin{bmatrix} \Delta z \\ \Delta \bar{z} \end{bmatrix} + \frac{1}{2} \begin{bmatrix} \Delta z \\ \Delta \bar{z} \end{bmatrix}^* H^w\big(z\big) \begin{bmatrix} \Delta z \\ \Delta \bar{z} \end{bmatrix} + h.o.t.,$$

where $H^w\big(z\big)$ is the Wirtinger Hessian matrix. Furthermore, as $g(z)$ is real-valued, we have $\overline{\partial_z g} = \partial_{\bar{z}} g$, which yields

$$\big(\nabla^w g(z)\big)^T \begin{bmatrix} \Delta z \\ \Delta \bar{z} \end{bmatrix} = \langle \partial_z g, \Delta z \rangle + \overline{\langle \partial_z g, \Delta z \rangle} = 2 \text{ real}(\langle \partial_z g, \Delta z \rangle) := \langle \partial_z g, \Delta z \rangle_\omega.$$

Similarly, we can simplify the second order term

$$\begin{aligned} \begin{bmatrix} \Delta z \\ \Delta \bar{z} \end{bmatrix}^* H^w\big(z\big) \begin{bmatrix} \Delta z \\ \Delta \bar{z} \end{bmatrix} &= 2 \text{ real}\big(\Delta z^* H_{zz} \Delta z + \Delta z^* H_{\bar{z}z} \Delta \bar{z}\big), \\ H_{zz} &= \frac{\partial}{\partial z}\Big(\frac{\partial g}{\partial z}\Big)^*, \; H_{z\bar{z}} = \frac{\partial}{\partial z}\Big(\frac{\partial g}{\partial \bar{z}}\Big)^*. \end{aligned}$$

We say that $h \in \mathbb{C}^d$ is a Wirtinger subgradient of $g : \mathbb{C}^d \to \mathbb{R}$ at $z \in \mathbf{dom}\ g$ if

$$g(z') \geq g(z) + \langle h, z' - z \rangle_w, \ \forall z' \in \mathbf{dom}\ g. \tag{10}$$

Let $\partial^w g(z) := \{h \in \mathbb{C}^d : \ h \text{ satisfies (10)}\}$ be the set of subgradients of $g$ at $z$. If $\partial^w g(z) \neq \emptyset$, $g$ is said to be Wirtinger sub-differentiable at $z$. If $g : \mathbb{C}^d \to \mathbb{R}$ is convex and $z \in \text{int } \mathbf{dom}\ g$, then $\partial^w g(z)$ is nonempty and bounded. A point $\hat{z}$ is a minimizer of a convex function $g : \mathbb{C}^d \to \mathbb{R}$ if and only if $g$ is Wirtinger sub-differentiable at $\hat{z}$ and $0 \in \partial^w g(\hat{z})$.

## 3. Wavelet framelet based recovery model and algorithms

In this section, a basic wavelet-based recovery model is first developed and some improvements on the fidelity and penalty term are imposed to achieve a better recovery performance. Then a proximal alternating linearization algorithm is designed for the proposed model.

### *3.1. Wavelet framelet based model*

For the complex signal $x \in \mathbb{C}^n$ and its phaseless measurements $b = |Ax|$ with $A \in \mathbb{C}^{m\times n}$, $b \in \mathbb{R}^n$, let $W \in \mathbb{R}^{L\times n}$ be the analysis operator associated with some wavelet tight frame system such that $W^\top W = I$. Inspired by (3), we firstly separate the signal variable $x$ and the phase variable $y$. The next step is to compute $\alpha$ using the balanced wavelet approach (7) instead of the least square method in (3):

$$\begin{aligned} \min_{\alpha\in\mathbb{C}^L, y\in\mathbb{C}^m} \quad & \frac{1}{2}\|AW^T\alpha - y\|_2^2 + \frac{\kappa}{2}\|(I - WW^T)\alpha\|_2^2 + \|\lambda \circ \alpha\|_1, \\ \text{s.t.} \quad & |y| = b. \end{aligned} \tag{11}$$

Then $x$ is recovered by $x = W^T\alpha$. The $l_1$-norm term is to impose certain sparsity both on the amplitude and phase.

In reality, there are always noises on the observed magnitudes $b$, such as Gaussian noise due to the device measurement error and Poisson noise due to the randomness of photon capture. We should take them into consideration in our model if one wants to get a better recovery performance. Actually the basic model (11) can remove noises to some extent because of the $\ell_1$-norm penalization on the wavelet coefficients. But it is not enough to deal with complicated noises such as mixed and unknown noises in reality. The study in [34] shows that the fidelity term plays an important role in removing different types of noises and a simple weighted sum of $\ell_1$-norm and $\ell_2$-norm fidelity term works effectively and robustly for the removal of mixed and unknown noises. In the basic model (11), the magnitude constraint should be satisfied strictly. Taking the potential noises into consideration, we can relax the magnitude constraint to a certain extent. As mentioned above, we use the hybrid $l_1$-norm and $l_2$-norm fidelity term to deal with the mixed or unknown noises. Then the constraints in our model becomes

$$y \in \Delta = \{y \in \mathbb{C}^m \mid \omega_1\| \ |y| - b\|_1 + \omega_2\| \ |y| - b\|_2^2 \leq \epsilon\}, \tag{12}$$

where $\epsilon$ is the tolerance of the noise, $\omega_1,\ \omega_2 > 0$ are weighting parameters in the $l_1$-norm and $l_2$-norm fidelity terms.

One of the reasons behind the success of wavelet models in image restoration is that they use the $\ell_1$-penalty term on wavelet coefficients to pursue a sparse approximation and preserve the edges of the underlying solution. However, $\ell_1$-penalty term will also introduce artifact, or unwanted singularities in the smooth regions of images. Accordingly, a more suitable wavelet-based model is proposed in [35]. They model the images as piecewise smooth functions and partition the images into smooth and rough (singularities) regions so that the most suitable penalty norms can be used to different regions. They penalize the $\ell_1$-norm on the wavelet coefficients in the rough regions to pursue sparsity and keep the edges sharp while they penalize the $\ell_2$-norm in the smooth regions to remove noise and protect the smoothness. In addition, they introduce the singularity set as an unknown variable and optimize it using the alternative direction method so that the singularity set can be detected easily and automatically. Therefore, we merge their methods to get a new model:

$$\min_{\alpha\in\mathbb{C}^L, y\in\Delta, \mathbf{\Gamma}\subset\{1,2,\ldots,n\}} \quad f(\alpha, y, \mathbf{\Gamma}), \tag{13}$$

where

$$f(\alpha, y, \mathbf{\Gamma}) = \frac{1}{2}\|AW^T\alpha - y\|_2^2 + \frac{\kappa}{2}\|(I - WW^T)\alpha\|_2^2 + \|[\lambda\circ\alpha]_{\mathbf{\Gamma}}\|_{2,1} + \|[\gamma\circ\alpha]_{\mathbf{\Gamma_c}}\|_{2,2}^2. \tag{14}$$

Here, $\mathbf{\Gamma}$ is the singularity set, and $\mathbf{\Gamma_c}$ is the complement of $\mathbf{\Gamma}$. Note that the $\ell_1$-penalty term used by Cai et al. in [35] is defined in an isotropic way and it is a little bit different from the anisotropic $\ell_1$-penalty term used in model (11). The isotropic $\ell_1$-penalty term is used in model (14) so that the singularity set can be solved easily, while we use the anisotropic $\ell_1$-norm in for its convenience of generalization and determining the weights in a closed form.

Inspired by the concept behind model (13), we propose a more generalized hybrid penalization model as stated in (8). In particular, our model reduces to model (13) if we use the isotropic $\ell_1$-penalization and choose $\rho_1(\cdot)$ and $\rho_2(\cdot)$ to be the indicator functions of set $\mathbf{\Gamma}$ and $\mathbf{\Gamma_c}$, respectively. However, our penalty term in (8) is more flexible and general than that of model (13) because one can explore more patterns about $\rho_1(\cdot)$ and $\rho_2(\cdot)$. It can be interpreted that the parameterized hybrid penalty term tries to find the most suitable penalty strategy by self-adapting. The details will be discussed in subsection 3.2.

### *3.2. Algorithm based on proximal alternating linearization*

We first make the following assumption on two functions $\rho_1(\cdot)$ and $\rho_2(\cdot)$.

**Assumption 1.** *(i) $\rho_1 : \mathbb{R} \to \mathbb{R}_+$ and $\rho_2 : \mathbb{R} \to \mathbb{R}_+$ are coercive functions.*
*(ii) Both $\rho_1$ and $\rho_2$ are $C^2$ functions such that*

$$\rho_1''(\lambda) \geq 0,\ \rho_2''(\lambda) \geq 0,\ \rho_1''(\lambda) + \rho_2''(\lambda) > 0,\ \forall\lambda \in \mathbb{R}. \tag{15}$$

Then we rewrite our unified model (8) as follows:

$$\min_{\alpha\in\mathbb{C}^L, y\in\Delta, \lambda\in\mathbb{R}_+^L} \quad f(\alpha, y, \lambda) := F_1(\alpha, \lambda) + F_2(\alpha, y), \tag{16}$$

where

$$\begin{aligned} F_1(\alpha,\lambda) &= \sum_{i=1}^{L}\rho_1(\lambda_i)|\alpha_i| + \sum_{i=1}^{L}\rho_2(\lambda_i)|\alpha_i|^2, F_2(\alpha,y) \\ &= \frac{1}{2}\|AW^T\alpha - y\|_2^2 + \frac{\kappa}{2}\|(I-WW^T)\alpha\|_2^2. \end{aligned}$$

A variant of the alternating minimization method is used to solve (16) by updating the variables $\alpha$, $y$ and $\lambda$ one after another iteratively. In particular, the proximal linearization method is applied to the variable $\alpha$ while the objective function is minimized directly with respect to the variables $y$ and $\lambda$, i.e.,

$$\alpha^{k+1} = \arg\min_{\alpha\in\mathbb{C}^L} \ \tilde{f}_t(\alpha,\lambda^k,y^k;\alpha^k) \tag{17a}$$

$$y^{k+1} = \arg\min_{y\in\Delta} \ F_2(\alpha^{k+1},y), \tag{17b}$$

$$\lambda^{k+1} = \arg\min_{\lambda\in\mathbb{R}_+^L} \ F_1(\alpha^{k+1},\lambda), \tag{17c}$$

where $\tilde{f}_t$ is linearized with a proximal term under Wirtinger calculus:

$$\begin{aligned} \tilde{f}_t(\alpha,y^k,\lambda^k,\alpha^k) :=& \ 2\,\mathrm{real}(\langle g^k,\alpha-\alpha^k\rangle) + F_1(\alpha,\lambda^k) + \tfrac{t}{2}\|\alpha-\alpha^k\|^2 \\ =& \ F_1(\alpha,\lambda^k) + \tfrac{t}{2}\|\alpha-\alpha^k+\tfrac{2}{t}\overline{g^k}\|^2 - \tfrac{2}{t}\|\overline{g^k}\|^2, \end{aligned}$$

and

$$\overline{g^k} = \overline{\nabla_\alpha^w F_2(\alpha^k,y^k)} = \kappa(I-WW^\top)\alpha^k + WA^*(AW^\top\alpha^k - y^k).$$

Next, we focus on the solutions of subproblems (17a)–(17c).

(a). The subproblem of $\alpha^{k+1}$ is separable. We solve the $i$-th component of $\alpha^{k+1}$ as

$$\begin{aligned} \alpha_i^{k+1} &= \arg\min_{\alpha_i\in\mathbb{C}} \rho_1(\lambda_i^k)|\alpha_i| + \rho_2(\lambda_i^k)|\alpha_i|^2 + \tfrac{t}{2}|\alpha_i-\alpha_i^k+\tfrac{2}{t}\overline{g_i^k}|^2 \\ &= \tfrac{1}{1+2\rho_2(\lambda_i^k)/t}\Gamma_{\rho_1(\lambda_i^k)/t}(\alpha_i^k-\tfrac{2}{t}\overline{g_i^k}), \end{aligned} \tag{18}$$

where $\Gamma_\lambda(x)$ for $x\in\mathbb{C}$ is the optimal solution of the following problem

$$\min_{y\in\mathbb{C}} \ \frac{1}{2}\|y-x\|_2^2 + \lambda\|y\|_1,$$

and is also referred to as the soft-thresholding operator given by:

$$\Gamma_\lambda(x) = \begin{cases} 0, & \text{if } |x|\le\lambda, \\ \frac{x}{|x|}(|x|-\lambda), & \text{otherwise.} \end{cases}$$

For an easier notation, we denote the mixed thresholding operator in (18) as

$$\mathcal{M}_{(t,\lambda^k;\rho_1,\rho_2)}(\beta) := \frac{1}{1+2\rho_2(\lambda^k)/t}\Gamma_{\rho_1(\lambda^k)/t}(\beta). \tag{19}$$

**Remark.** The reason of using the proximal linearization method for $\alpha$ is that the subproblem with respect to $\alpha$:

$$\min_{\alpha\in\mathbb{C}^L} F_1(\alpha,\lambda^k)+F_2(\alpha,y^k)$$

has no closed form solution. For a better approximation, one can iterate multiple times for fixed $y^k$ and $\lambda^k$, i.e.,

$$\begin{cases} \alpha^{k_{j+1}}=\arg\min_{\alpha\in\mathbb{C}^L}\ \tilde{f}_t(\alpha,\lambda^k,y^k;\alpha^{k_j}),\ j=1,2,\ldots,J-1,\\ \alpha^{k+1}=\alpha^{k_J}. \end{cases}$$

Moreover, the accelerated proximal gradient (APG) method in [28] can be employed to speed up the convergence. However, our numerical experiments show that executing inner iterations multiple times may not improve the recovery performance too much, but the computational time is increased notably. Hence, in our paper, we only perform the step once at each outer loop. Note that the step size $t$ is fixed for convenience and ease of presentation but can be chosen adaptively.

(b). From (17b), we know $y^{k+1}$ is actually the projection of $AW^T\alpha^{k+1}$ on $\Delta$. Let $h^{k+1}=AW^T\alpha^{k+1}$, $p(\alpha^{k+1})=|h^{k+1}|-b$ and $p(\alpha^{k+1})^\uparrow$ be the vector with components arranged in an ascending order. For simplicity, we use $p$ and $p^\uparrow$ instead without clarification. The solution of (17b) is given by Lemma 1 whose proof is provided in section 7.

**Lemma 1.** *The solution of* (17b) *takes the following form:*

$$y^{k+1}=r^{k+1}\circ\frac{h^{k+1}}{\left|h^{k+1}\right|}.$$

*(i). If $h^{k+1}\in\Delta$, then $r^{k+1}=|h^{k+1}|$.*

*(ii). Otherwise,*

$$r^{k+1}=\frac{1}{2\omega_2\beta^{k+1}+1}\Gamma_{\omega_1\beta^{k+1}}\Big(|h^{k+1}|-b\Big)+b,$$

*where $\beta_{k+1}\in\left(\frac{|p_j^\uparrow|}{\omega_1},\frac{|p_{j+1}^\uparrow|}{\omega_1}\right]$ is a root of the equation*

$$\frac{\omega_1}{1+2\omega_2\beta^{k+1}}\sum_{i>j}(|p_i^\uparrow|-\omega_1\beta^{k+1})+\frac{\omega_2}{(1+2\omega_2\beta^{k+1})^2}\sum_{i>j}(|p_i^\uparrow|-\omega_1\beta^{k+1})^2=\epsilon.$$

(c). The subproblem for $\lambda$ is also separable. For the $i$-th component, we need to solve

$$\lambda_i^k=\arg\min_{\lambda_i\in\mathbb{R}_+}\rho_1(\lambda_i)|\alpha_i^k|+\rho_2(\lambda_i)|\alpha_i^k|^2,\ i=1,2,\ldots,L. \tag{20}$$

When $|\alpha_i^k| = 0$, the subproblem (20) is degenerated. In this case, we do not update $\lambda_i$ but keep $\lambda_i^k = \lambda_i^{k-1}$. Otherwise, for $i = \{i|\ |\alpha_i^k| \neq 0, i = 1, 2, \ldots, L\}$, we consider an equivalent form:

$$\lambda_i^k \quad = \arg\min_{\lambda_i \in \mathbb{R}} \rho_1(\lambda_i) + \rho_2(\lambda_i)|\alpha_i^k| + \delta_{\mathbb{R}_+}(\lambda_i), \tag{21}$$

where $\delta_{\mathbb{R}_+}(\lambda_i)$ is the indicator function of the set $\mathbb{R}_+$. When Assumption 1 is satisfied, we have

$$\rho_1''(\lambda_i) + \rho_2''(\lambda_i)|\alpha_i^k| > 0,$$

which means that $\rho_1(\cdot) + \rho_2(\cdot)|\alpha_i^k|$ is strictly convex. Hence, the function $\rho_1(\cdot) + \rho_2(\cdot)|\alpha_i^k| + \delta_{\mathbb{R}_+}(\cdot)$ is proper, lower semi-continuous, coercive and strictly convex. Therefore, the solution of (21) exits and is unique, which is given by

$$\lambda_i^k = \max\{0, \tilde{\lambda}_i^k\}, \tag{22}$$

where $\tilde{\lambda}_i^k$ satisfies

$$\rho_1'(\tilde{\lambda}_i^k) + \rho_2'(\tilde{\lambda}_i^k)|\alpha_i^k| = 0. \tag{23}$$

Note that the solution of (23) exits and is unique because $\rho_1(\cdot) + \rho_2(\cdot)|\alpha_i^k|$ is smooth, coercive and strictly convex.

As mentioned before, the purpose of adding the $\ell_1$-norm is for sparsity, while adding the $\ell_2$-norm is for smoothness. For a trade-off between sparsity and smoothness, it is natural to design $\rho_1$ and $\rho_2$ inversely related. In addition, we prefer the solution of (21) to have a simple and closed form. Here, we give two examples of the choices.

(1). $\rho_1(\lambda) = \mu\lambda$ and $\rho_2(\lambda) = (C - \lambda)^2$, where $\mu$ and $C$ are hyperparameters. The solution of (21) is

$$\lambda_i^k = \begin{cases} 0, & |\alpha_i^k| \leq \frac{\mu}{2C}, \\ C - \frac{\mu}{2|\alpha_i^k|}, & |\alpha_i^k| > \frac{\mu}{2C}, \end{cases} \qquad i = 1, 2, \ldots, L. \tag{24}$$

Assume that $\{\alpha^k\}$ converge to $\hat{\alpha}$. By substituting (24) into $\rho_1(\lambda_i)|\alpha_i^k| + \rho_2(\lambda_2)|\alpha_i^k|^2$ for $i = 1, 2, \ldots, L$, and taking the limit, we get that

$$\lim_{k\to\infty} \left(\rho_1(\lambda^i)|\alpha_i^k| + \rho_2(\lambda_2)|\alpha_i^k|^2\right) = \begin{cases} C^2|\hat{\alpha}_i|^2, & |\hat{\alpha}_i| \leq \frac{\mu}{2C}, \\ C\mu|\hat{\alpha}_i| - \frac{\mu^2}{4}, & |\hat{\alpha}_i| > \frac{\mu}{2C}, \end{cases} \tag{25}$$

which is actually the Huber norm adopted in [36].

(2). $\rho_1(\lambda) = \mu\lambda^2$ and $\rho_2(\lambda) = (C - \lambda)^2$, where $\mu$ and $C$ are hyperparameters. Then for $i = 1, 2, \ldots, L$, we obtain the solution of (21) as

$$\lambda_i^k = \frac{C|\alpha_i^k|}{\mu + |\alpha_i^k|}. \tag{26}$$

Assume that $\{\alpha^k\}$ converges to $\hat{\alpha}$ and substitute (26) into $\rho_1(\lambda_i)|\alpha_i^k| + \rho_2(\lambda_2)|\alpha_i^k|^2$. By taking the limit, we get that

$$\lim_{k\to\infty} \big(\rho_1(\lambda_i)|\alpha_i^k| + \rho_2(\lambda_2)|\alpha_i^k|^2\big) = \frac{\mu C^2|\hat{\alpha}_i|^2}{\mu + |\hat{\alpha}_i|}, \quad i = 1, 2, \ldots, L. \tag{27}$$

The equation (27) is similar to the reweighed $\ell_2$ scheme in [37].

Both of the above designed patterns satisfy Assumption 1 with positive $\mu$ and $C$. Then they converges to well-known penalty terms, which indicates that the data-driven penalty terms are learned well during iterations and they reach a suitable term finally. Our method also provides a possible way to explore better penalty terms.

### *3.3. Convergence results*

The Wirtinger Hessian matrix of $F_2(\alpha, y)$ with respect to $\alpha$ is

$$H_\alpha^w\big(F_2(\alpha, y)\big) = \begin{pmatrix} WA^*AW^T + \kappa(I - WW^T) & 0 \\ 0 & WA^*AW^T + \kappa(I - WW^T) \end{pmatrix}.$$

Let $H$ denote $WA^*AW^T + \kappa(I - WW^T)$, which is real in coded diffraction model. Let $\lambda_{\max}$ be the maximum eigenvalue of $H$. Note that in [38], Attouch et al. propose a proximal alternating linearized scheme for nonconvex and nonsmooth problems and establish global convergence result. Their method uses a proximal linearization scheme for each block in real domain while our algorithm updates $\alpha$ in complex domain and solves the subproblem of $y$ and $\lambda$ exactly. However, we still establish convergence in a similar fashion.

**Theorem 1.** *Suppose that $t > 2\lambda_{\max}$, and the terms $\rho_1(\cdot)$ and $\rho_2(\cdot)$ satisfy Assumption* 1*. Then the following facts holds.*

*(i) The sequence $\{f(\alpha^k, y^k, \lambda^k)\}_{k=0}^\infty$ generated by* (17) *is convergent. $\{\alpha^k\}_{k=0}^\infty$, $\{y^k\}_{k=0}^\infty$, $\{\lambda^k\}_{k=0}^\infty$ are all convergent.*

*(ii) Let $(\hat{\alpha}, \hat{y}, \hat{\lambda})$ be a limit of $\{(\alpha^k, y^k, \lambda^k)\}$. Then $(\hat{\alpha}, \hat{y}, \hat{\lambda})$ satisfies the following conditions*

$$\hat{\alpha} = \arg\min_\alpha F_1(\alpha, \hat{\lambda}) + F_2(\alpha, \hat{y}), \tag{28}$$

$$\hat{y} = \arg\min_{y\in\Delta} F_2(\hat{\alpha}, y), \tag{29}$$

$$\hat{\lambda} = \arg\min_{\lambda\in\mathbb{R}_+^L} F_1(\hat{\alpha}, \lambda). \tag{30}$$

*Thus $(\hat{\alpha}, \hat{y}, \hat{\lambda})$ is a Nash equilibrium of* (11).

Proof. The proof is given in subsection 7.2.

## 4. Set-up and implementation details

In this section, we clarify the set-up of numerical experiments and implementation details of our algorithm.

### *4.1. Measurement set-up*

In physical experiments, compactly supported signals are padded with $(s-1)n$ zeros to conduct an over sampled discrete Fourier transform by an integer factor $s \geq 2$ instead of the traditional DFT. In this paper, the oversampling factor $s$ is set to be 2.

In the following, we work with two kinds of patterns: octanary pattern and bipolar pattern. The former is suggested in [4] and [10] due to some statistical consideration and the latter is easy to be implemented in realistic experiments. For both patterns, $I_l$ $(l = 1, \ldots, k)$ are i.i.d. distributed and their entries are i.i.d. sampled from a distribution $d$.

- *Octanary pattern.* In this pattern, $d = c_1 c_2$, where

$$c_1 = \begin{cases} +1 & \text{with prob. } 1/4 \\ -1 & \text{with prob. } 1/4 \\ -i & \text{with prob. } 1/4 \\ +i & \text{with prob. } 1/4 \end{cases} \quad \text{and} \quad c_2 = \begin{cases} \sqrt{2}/2 & \text{with prob. } 4/5 \\ \sqrt{3} & \text{with prob. } 1/5 \end{cases}. \tag{31}$$

- *Bipolar pattern.* Here, $d$ is distributed as

$$d = \begin{cases} +1 & \text{with prob. } 1/2 \\ -1 & \text{with prob. } 1/2 \end{cases}. \tag{32}$$

The pseudo-inverse of $A$ is given by

$$A^{\dagger} \begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix} = \sum_{l=1}^{m} \mathcal{F}^{-1}(y_l) \circ I_l^{'},$$

where $I_l^{'}$ is the dual mask of $I_l$ defined as

$$I_l^{'} = \overline{I}_l / (\sum_s |I_s|^2).$$

Due to the special structures and the FFT tranform, the multiplications with respect to A and $A^{\dagger}$ can be computed efficiently.

Since the measurements may be contaminated by noises in reality, we simulate Poisson noise and Gaussian noise, as well as their mixture. The measurements contaminated by Poisson noise are obtained by

$$\mathcal{P}_{\alpha}(b) = \sqrt{\alpha \cdot \text{Poisson}\Big(\frac{|b|^2}{\alpha}\Big)}, \tag{P}$$

where $\alpha$ is the Poisson noise level and Poisson($\lambda$) is a random Poisson sample of mean $\lambda$. Similarly, the measurements contaminated by Gaussian noise are obtained by

$$\mathcal{G}_\beta(b) = \sqrt{\max\left\{0, |b|^2 + \beta \cdot \text{Gaussian}\Big(0, I\Big)\right\}}, \tag{G}$$

where $\beta$ is the Gaussian noise level and $\text{Gaussian}\Big(0, I\Big)$ is a random sample from standard normal distribution. Mixed noises are performed by superimposing (P) and (G). We measure the phase recovery performance by relative mean squared error (MSE). However, since the solution is not unique up to a global phase, it does not make sense to compute the distance between $x$ and its approximation $\hat{x}$. Instead we compute the distance modulo a global phase term and define the relative MSE between x and $\hat{x}$ as

$$\epsilon(x, \hat{x}) \triangleq \min_{c\in\mathbb{C},|c|=1} \frac{||x - c\hat{x}||_F}{||x||_F}. \tag{33}$$

It is easy to get that the optimal $c$ should be $\frac{\bar{\hat{x}}^\top x}{|\hat{x}^\top x|}$.

*4.2. Implementation details of our algorithm*

We next present the implementation details and parameter settings in our algorithm.

*4.2.1. Iterating in image domain*

The update of $\alpha^k$ solved by (18) performs the operations in the wavelet tight frame space, which is quite expensive due to $L \gg n$. In order to reduce the computational time, we propose some strategies such that the operations are done in image domain, rather than in wavelet coefficient domain. The update rule of (18) gives

$$\begin{aligned}
&\alpha^{k+1} = \tfrac{\Gamma_{\rho_1(\lambda^k)/t}}{1+2\rho_2(\lambda^k)/t}(\alpha^k - \tfrac{2}{t}\overline{g^k}) \\
&= \tfrac{\Gamma_{\rho_1(\lambda^k)/t}}{1+2\rho_2(\lambda^k)/t}\Gamma_{\rho_1(\lambda^k)/t}\Big(\alpha^k - \tfrac{2}{t}\big(\kappa(I - WW^T)\alpha^k + WA^*(AW^T\alpha^k - y^k)\big)\Big).
\end{aligned} \tag{34}$$

If we set $t = 2\kappa$, then

$$\alpha^{k+1} = \frac{1}{1 + \rho_2(\lambda^k)/\kappa}\Gamma_{\rho_1(\lambda^k)/(2\kappa)}(W(I - \frac{1}{\kappa}A^*A)W^T\alpha^k + \frac{1}{\kappa}WA^*y^k). \tag{35}$$

Apparently, the scheme in (35) can be rewritten equivalently as

$$\left\{\begin{array}{rcl} u^k &=& (I - \frac{1}{\kappa}A^*A)W^\top\alpha^k + \frac{1}{\kappa}A^*y^k, \\ \alpha^{k+1} &=& \frac{1}{1+\rho_2(\lambda_i^k)/\kappa}\Gamma_{\rho_1(\lambda_i^k)/(2\kappa)}(Wu^k). \end{array}\right. \tag{36}$$

In contrast to the update in (34), the iteration scheme (36) only performs the wavelet analysis and synthesis operations once, respectively. All addition and subtraction operations are completed in image domain. Therefore, it requires much less computational time than (35). Similar techniques are also used in [39].

The convergence results stated in Theorem 1 still apply here. Recall that $H = WA^*AW^T + \kappa(I - WW^T)$ and $\lambda_{\max}$ is the largest eigenvalue of $H$. We should set $\kappa$ such that $t(= 2\kappa)$ is greater than $2\lambda_{\max}$. This can be guaranteed since $\kappa I - A^*A = \kappa I - \mathrm{diag}(\sum_{l=1}^m |I_l|^2)$ in the coded diffraction model. In practice, we usually set $\kappa$ to be a diagonal matrix $\mathrm{diag}(1.1\sum_{l=1}^m |I_l|^2)$ and modify the corresponding terms accordingly. It is obvious that $\kappa I - A^*A$ is positive definite in this setting.

*4.2.2. Data-driven penalty parameter*

In subsection 3.2, we have developed a data-driven method for determining the weights of the $\ell_1$ and $\ell_2$ norm in the penalty term. However, we still need a parameter to balance the fidelity and penalty terms. Thus a penalty parameter is introduced as a positive scalar before the penalty term. The model (16) becomes

$$\min_{\alpha\in\mathbb{C}^L, y\in\Delta, \lambda\in\mathbb{R}_+^L} \quad \delta F_1(\alpha,\lambda) + F_2(\alpha, y). \tag{37}$$

More importantly, we update $\delta$ in a data-driven way.

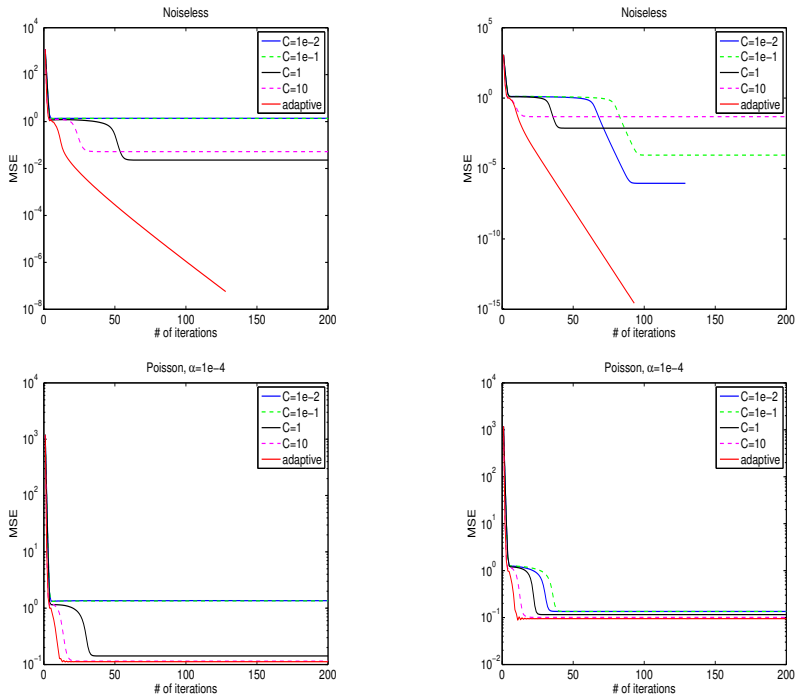
Intuitively, we know that the higher the noise level is, the larger the penalty parameter should be. But it is hard to estimate the noise level statistically, as the noise is usually of mixed and unknown types in reality. However, we find numerically that with the same penalty parameters, the solution obtained in the noiseless case will have a smaller relative residual of magnitude defined as

$$\text{rel.err} = \frac{\|\,|Ax| - b\|_2}{\|b\|^2}$$

than that in the noise case. Such observation is not hard to understand as the recovered signal from observations with high noise level tends to fit the fidelity worse. That is, there is a positive correlation between the noise level and the magnitude residual. Hence, it makes sense to self-adapt the penalty parameters during the iteration according to the residual of last step. Our extensive numerical experiments show that choosing the penalty parameter as

$$\delta = \min\{10, 2\cdot\left(e^{3\cdot\text{rel.err}} - 1\right)\} \tag{38}$$

often provides reasonable performance. In the following, we choose $\rho_1(\lambda) = \mu\lambda$ and $\rho_2(\lambda) = (C - \lambda)^2$ for model (8). At iteration $k$, the paramters $C^k$ and $\mu^k$ in $\rho_1$ and $\rho_2$ are updated as follows. Since $C^k$ takes the same role as the penalty parameter $\delta$ in (37), we update $C^k$ as in (38) and set $\mu^k$ to be $\mu_{\mathbf{i}}^k = 4C^k \cdot \text{percentile}\{|\alpha_{\mathbf{i}}^{k-1}|, 70\}$, where $\mathbf{i} \in \mathbb{B}$ and $\mathbb{B}$ is the wavelet band set. The notation $\text{percentile}\{|\alpha_{\mathbf{i}}^{k-1}|, 70\}$ stands for the 70th percentile of the $\mathbf{i}$-th band of $|\alpha^{k-1}|$.




(a) the number of bipolar masks is 2 (b) the number of bipolar masks is 8

Figure 2: MSE versus the iterations. The first and second row correspond to noiseless case and the one with Possion noise, respectively.

We next test the effectiveness of the data-driven penalty parameter in (38) on the caffeine molecule by comparing with the performance of different constant penalty parameters. In addition to the setting of $\nu, t, C^k, \mu^k$ discussed above, we always set $\omega_1 = \omega_2 = 1$ and $\epsilon = 10^{-2}$ in our algorithm without clarification. subsection 4.2.2 depicts the history of MSE for different numbers of masks and noise levels.

When the number of masks is small or the noise level is high, a larger constant penalty parameter performs better because a larger penalty parameter can remove the ambiguity and noise better for ill-posed noisy problem. When there is no noise and the number of masks is relatively large (see the top right figure), a smaller penalty parameter is preferred probably because the problem is more well-posed and large penalty parameters over smooth the result. Obviously, the proposed data-driven penalty parameter shows advantages over constant penalty parameters. In the noiseless case, constant penalty parameters always get stuck in some local optimum while data-driven penalty parameter leads to a quick convergence to the truth. At the beginning stage, a large penalty parameter is necessary to remove ambiguity while after a few iterations, the ambiguity has been removed enough and the recovery result is close to the truth. At this

time, the penalty parameter should be reduced to avoid over smoothing. In the noisy case, a large penalty parameter is always preferred to remove noise. Our data-driven parameter still performs well as it starts from a large parameter and does not reduce too much until the recovery result fits the observation well enough.

*4.2.3. Summary of our algorithms*

We summarize the improvements to derive our unified model and algorithms. The model is basically (8), with $\rho_1(\lambda) = \mu\lambda$ and $\rho_2(\lambda) = (C-\lambda)^2$. In general, we solve (8) by a three-block direction alternating projection method. At the $k$-th iteration, $\alpha^k$ is computed as (18); $y^k$ is solved by Lemma 1 where $\beta$ is determined by a binary search algorithm; the calculation of $\lambda^k$ is the same as (25). Simultaneously, we update the penalty parameter $C$ as (38) and perform iterations in the image domain stated in subsection 4.2.1 to reduce the computational time. Our data driven wavelet frame based method is outlined in Algorithm 1.

**Algorithm 1:** A Data-Driven Wavelet Frame based Algorithm

Choose $u^0 \in \mathbb{C}^n$, $\alpha^0 = Wu^0$, $y^0 = b \circ \frac{Au^0}{|Au^0|}$; set the parameters $\omega_1 = \omega_2 = 1$, $\epsilon = 1e-2$, and $\kappa = \text{diag}\left(1.1\sum_{l=1}^m |I_l|^2\right)$;

**while** *k=0,...,N-1* **do**

  set $C^{k+1} = \min\{10, 2\cdot\left(e^{3\cdot\text{rel.err}} - 1\right)\}$ and $\mu_{\mathbf{i}}^{k+1} = 4C^{k+1}\cdot\text{percentile}\{|\alpha_{\mathbf{i}}^k|, 70\}$, $\mathbf{i} \in \mathbb{B}$.

  compute $u^{k+1} = (I - \frac{1}{\kappa}A^*A)W^\top\alpha^k + \frac{1}{\kappa}A^*y^k$.

  compute $\alpha^{k+1} = \frac{1}{1+\rho_2(\lambda^k)/\kappa}\Gamma_{\rho_1(\lambda^k)/(2\kappa)}(Wu^{k+1})$.

  compute $y^{k+1}$ by Lemma Lemma 1 where $\beta$ is determined by Algorithm 2.

  update $\lambda^{k+1}$ as (25).

$x = W^T\alpha^N$.

*4.3. Settings of other selected algorithms*

We compare our method with three other algorithms: HIO, PhaseCut and WirtingerFlow. HIO is selected due to its easy implementation and its efficiency as a greedy algorithm. PhaseCut is selected as it is insensitive to the initial points. WirtingerFlow is selected due to its global convergence with high probability, and TVB is selected due to its involvement of sparsity prior.

Except that WitingerFlow uses a spectral initial guess stated in [10], HIO, PhaseCut, TVB and our method are initialized from a random Gaussian point $x_0$ and the initial phase of $y_0$ is set to be the phase of $Ax_0$. For HIO, the parameter $\beta$ in (HIO) is set to be 1.7. As suggested in [16], PhaseCut keeps only the largest 1000 observations and uses a rank-2 approximation block coordinate

descent solver to solve the lifted SDP problem, followed by a HIO refinement. In [10], the stepsize $\mu^k$ in (WirtignerFlow) was set to be

$$\mu^k = \min\{1 - e^{-k/330}, 0.4\},$$

which is unstable in our numerical experiments. In this paper, we improve the numerical performance of WitingerFlow by using the Barzilar-Borwein step size as well as a few other known strategies for $\mu^k$. For TVB, we use the parameters suggested in Table 3 of the original paper [24].

All these four algorithms are terminated if the relative residuals $\||Ax| - b\|_2/\|b\|_2 < \eta$ or a maximum number of iterations are reached. We also stop the iterations when the difference of the recovered images between two successive iterations is smaller enough, i.e., $\|x_{k+1} - x_k\|_F \leq \eta$. Here, we choose two sets of values for the terminal conditions including one for a high accuracy recovery and another for time saving.

(TC1) Let $\eta$=1e-10; set the maximum number of iterations for HIO (as well as the refinement of PhaseCut), WirtingerFlow, TVB, our method to be 5000, 2000, 500, 200, respectively;

(TC2) Let $\eta$=1e-8; set the maximum number of iterations for HIO (as well as the refinement of PhaseCut), WirtingerFlow, TVB, our method to be 1000, 1000, 200, 100, respectively.

Usually, (TC1) are employed for small size problems and noiseless cases. Otherwise, we use (TC2).

## 5. Numerical experiments

In this section, we perform some numerical experiments to illustrate the effectiveness of our algorithm. All the tests are conducted on a MacBook Pro with 2-core 2.4 GHz Inter core i5 processors and 8GB memory.

### *5.1. Performance on molecules with multiple masks*

Firstly, we test the performance of the selected algorithms on molecules in noiseless case. In X-ray crystallography, molecules are illuminated from different angles and their intensity of diffracted rays are collected, which can be modeled as the phase retrieval problem (2). The 3D molecular structure can be recovered if sufficient 2D projection slices of this molecule are recovered. In this paper, we only pay attention to the 2D projection recovery and refer the readers to [4] and [10] for the 3D recovery. Here fifty 3D molecules are selected from the Protein Data Bank for study. Their 2D projection slices from one illustration angle are shown in subsection 5.1.

In this subsection, the terminal condition (TC1) is employed. The reconstruction is said to be successful if the relative MSE between $x$ and $\hat{x}$ defined in (33) is smaller than $10^{-3}$. The number of masks ranges form 2 to 12 and the empirical successful probability is computed as an average over the selected fifty

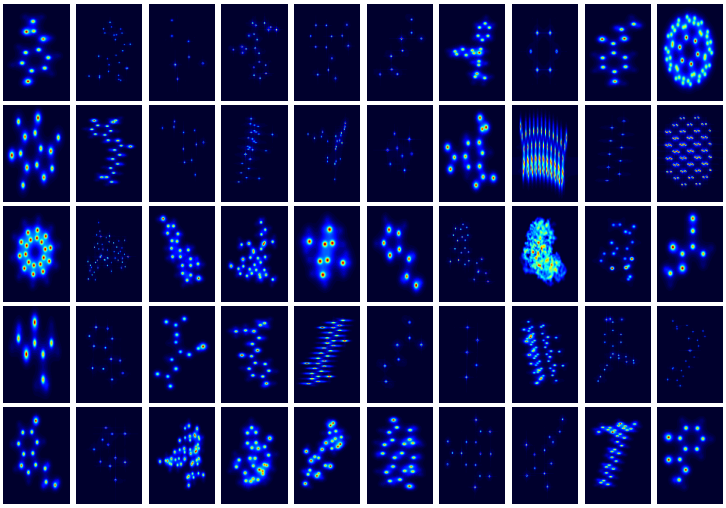


Figure 3: Database: 2D projection slices of the selected 50 molecules.

protein molecules shown in subsection 5.1. From subsection 5.1, we observe that the our method and TVB outperform than HIO, PhaseCut and WirtingerFlow in terms of the successful recovery rate when the number of masks is small. It is easy to understand because the sparsity prior removes ambiguity caused by losing of phase information well when the measurements are insufficient. PhaseCut improves the recovery rate of HIO slightly. However, since it lifts the variable to matrix, its computational time is most expensive. Our algorithm not only improves the recovery rate significantly, but also outperforms PhaseCut in both memory storage and CPU time. WirtingerFlow also shows a higher successful recovery rate than HIO. However, the spectral initialization in [10] assumes some statistical properties of the masks while our method uses a simple Gaussian random initialization and thus is easier to be extended to other kind of masks.

### *5.2. Robustness to complicated noise*

In this subsection, we simulate four types of noise on the measurements: single Gaussian noise ($\beta = 1e-3$), single Poisson noise ($\alpha = 1e-2$), mixed Gaussian noise ($\beta_1 = 1e-3$ and $\beta_2 = 1e-4$) and mixed Gaussian-Poisson noise ($\beta = 1e-4$ and $\alpha = 1e-2$). The stopping condition (TC2) is employed here and the number of masks is set to be 8 to guarantee that the selected algorithms work in noiseless case at least. The recovery results on the above molecule data set are displayed in subsection 5.2.

These figures show clearly that our method is better than the other three algorithms on all tested molecules and noise types. In general, the performance of HIO and PhaseCut is quite close to each other, which means that the initialization strategy of PhaseCut may not work very well in the presence of noise,

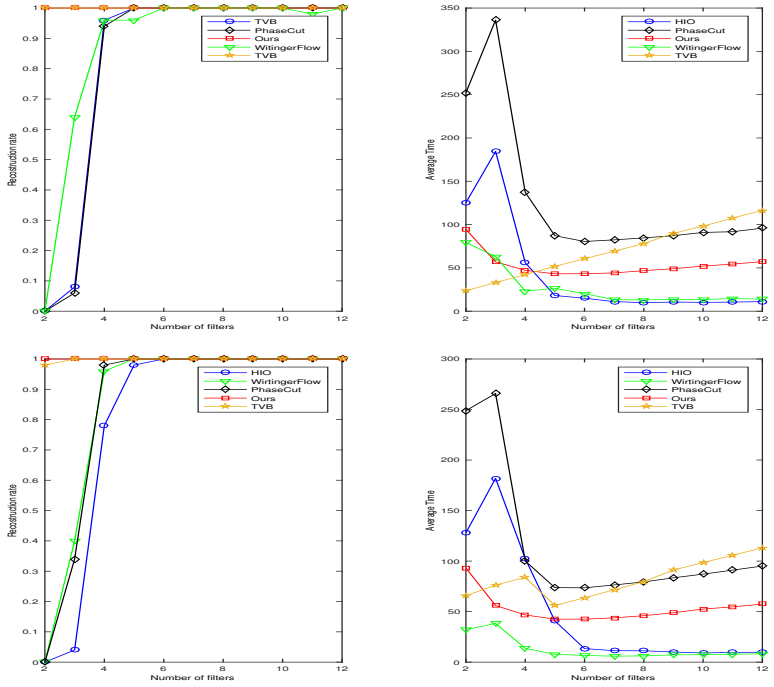



Figure 4: The left and right column are the reconstruction rate and average CPU time, respectively, in the noiseless case. The first and second row correspond to octanary masks and bipolar masks, respectively.

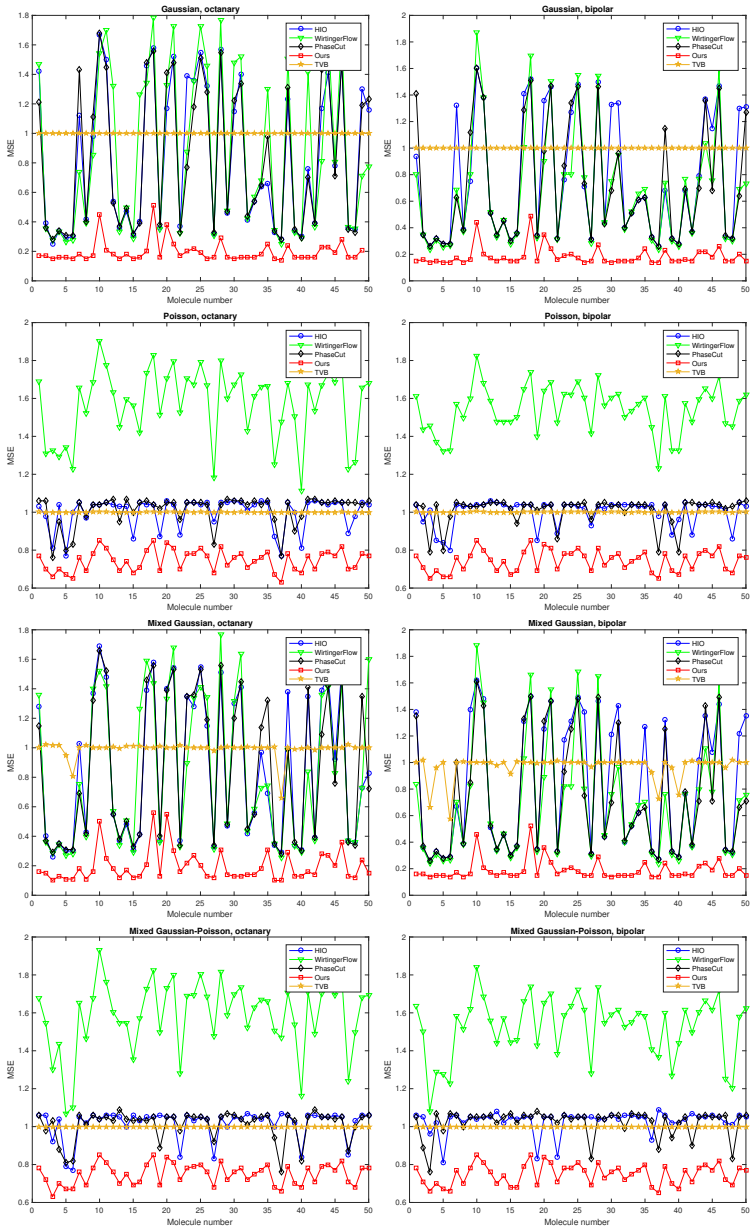
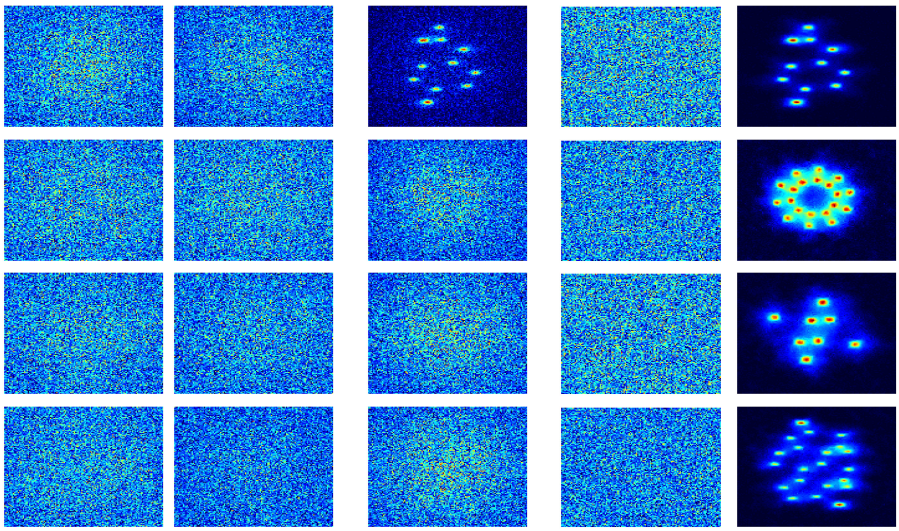



Figure 5: Comparison of MSEs of the five algorithms in noisy case. The first to fourth rows are the results of Gaussian, Possion, mixed Gaussian, mixed Gaussion-Possion noises. The left and right column correspond to octanary masks and bipolar masks, respectively.



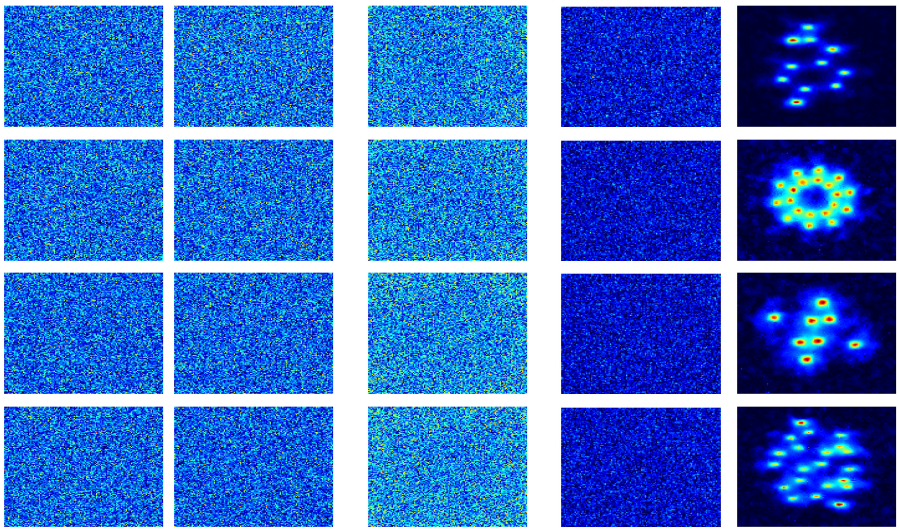
(a) HIO (b) PhaseCut (c) WirtingerFlow (d) TVB (e) our method

Figure 6: Recovered density maps under Mixed Gaussian noise.

although it can improve the recovery rate slightly in noiseless case. WirtingerFlow works a little bit better than HIO and PhaseCut for Gaussian noise removal with bipolar masks while the performance is the opposite with octanary masks. For TVB, the MSEs are around one in most cases because the coefficients of recovered images are very small. It may be attributed to over smoothing caused by the sparsity regularization in the TVB model. Note that this phenomenon can not be fixed even if we use smaller regularization thresholding parameters. Nevertheless, our method avoids over smoothing in noisy case due to the design of fidelity and penalty terms. For a closer visualization, we select four molecules with the bipolar masks. subsection 5.2 and subsection 5.2 display the recovered density maps under mixed Gaussian noise and mixed Gaussian-Poisson noise, respectively. In most cases, only our method can reconstruct the atom locations successfully.

### *5.3. Successful recovery with single mask*

This subsection further demonstrates the recovery using only a single mask. The numerical results in subsection 5.1 show that HIO, PhaseCut and WirtingerFlow fail even if there are two masks. Thus, only TVB and our method are compared in this subsection. Firstly, we test the recovery performance in noiseless case and run over all the molecules in subsection 5.1. The recovered results with one bipolar mask is shown in subsection 5.3, where (TC1) is employed. It can be observed that our method recovers all molecules successfully, while TVB failed for three molecules (boxed in red). Next, we test the recovery performance in noisy case and (TC2) is used. subsection 5.3 presents the recovered images by TVB and our method with one bipolar mask under different types of noise. It



(a) HIO (b) PhaseCut (c) WirtingerFlow (d) TVB (e) our method

Figure 7: Recovery results under mixed Gaussian-Poisson noise.

shows that TVB fails with chaotic outputs, while our method computes better results in the terms of recovering the locations of the atoms successfully.

In summary, the performance of TVB and our method is close in noiseless case, although our method is slightly better. However, our method offers better recovery than TVB in noisy case.

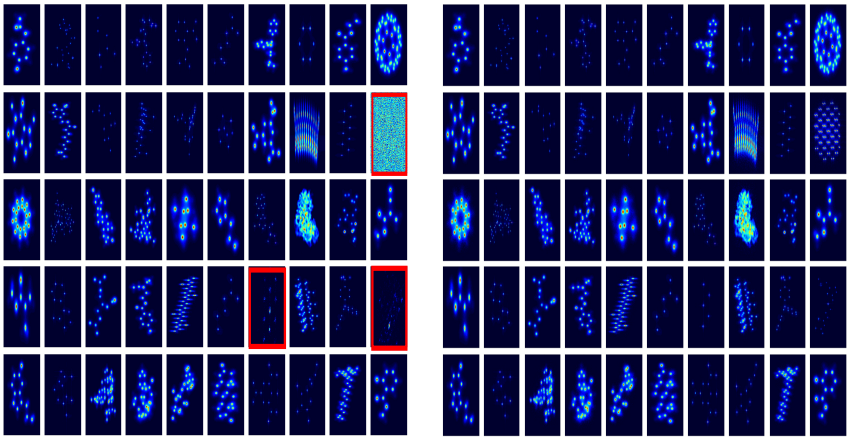
## 6. Conclusions

In this paper, we apply the wavelet tight frame approaches to phase retrieval problem. Hybrid fidelity and penalty terms are used and a data-driven algorithm is designed. Numerical results show that our method outperforms several state-of-the-art methods in terms of both the minimal number of measurements required for a high quality recovery and robustness to complicated noise. This ability makes our method quite practical in applications such as X-ray crystallography, where the radiation dose should be controlled strictly. Future directions include the investigation of our method on real data from X-ray experiments and so on.

## 7. Proofs

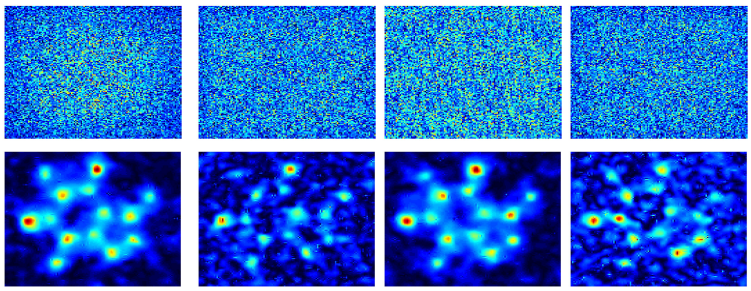
### *7.1. Proof of Lemma* 1

Proof. Let $y = r \circ \exp(i\theta)$, where $r \in \mathbb{R}^m_+$, $\theta \in \mathbb{R}^m$. Denote $h^{k+1} = |h^{k+1}| \circ \exp(i\phi^{k+1})$, $\phi^{k+1} \in [0, 2\pi)^m$. If $|h^{k+1}_j| = 0$, we define $\phi^{k+1}_j = 0$. Then (17b)



(a) TVB (b) our method

Figure 8: Recovered results with one bipolar mask in noiseless case.



(a) (b) (c) (d)

Figure 9: Recovered images of TVB (the first row) and our method (the second row) with one bipolar mask. (a) single Poisson noise, $\alpha = 1e-2$; (b) single Gaussian noise, $\beta = 1e-3$; (c) mixed Gaussian noise, $\beta_1 = 1e-3$, $\beta_2 = 1e-4$; (d) mixed Gaussian-Poisson noise, $\beta = 1e-4$, $\alpha = 1e-2$.

reduces to

$$\begin{array}{ll} \min\limits_{r\in\mathbb{R}^m_+,\theta\in\mathbb{R}^m} & \varphi(r,\theta)=\frac{1}{2}\|r\exp(i\theta)-h^{k+1}\|_2^2 \\ \text{s.t.} & \omega_1\|r-b\|_1+\omega_2\|r-b\|_2^2\le\epsilon. \end{array} \tag{39}$$

Obviously, the optimal solution of $\theta$ is achieved when $\theta_j^{k+1}=\phi_j^{k+1}$, $j=1,\cdots,m$. Substituting $\theta^{k+1}$ into (39) yields the following optimization problem which only involves $r$:

$$\begin{array}{ll} \min\limits_{r\in\mathbb{R}^m_+} & \frac{1}{2}\|r-|h^{k+1}|\|_2^2 \\ \text{s.t.} & \omega_1\|r-b\|_1+\omega_2\|r-b\|_2^2\le\epsilon. \end{array} \tag{40}$$

Actually, the non-negative constraint on $r$ can be thrown away. Because if $r\in\mathbb{R}^m$ is feasible, $\max\{0,r\}\in\mathbb{R}^m_+$ is also feasible and has a smaller objective value. Then (40) can be reduced to the following problem

$$r^{k+1}=\arg\min_{r\in\Delta'}\frac{1}{2}\|r-|h^{k+1}|\|_2^2, \tag{41}$$

where $\Delta'=\{r\in\mathbb{R}^m:\omega_1\|r-b\|_1+\omega_2\|r-b\|_2^2\le\epsilon\}$. Note that $\Delta'$ is a convex set. Hence, the projection onto $\Delta'$ is unique. In other words, $r^{k+1}$ is a unique solution of (41).

Let $q=r-b$, $p=|h^{k+1}|-b\in\mathbb{R}^m$. Then $r=q+b$, and (41) can be equivalently written as

$$\begin{array}{ll} \min\limits_{q\in\mathbb{R}^m} & \frac{1}{2}\|q-p\|_2^2 \\ \text{s.t.} & \omega_1\|q\|_1+\omega_2\|q\|_2^2\le\epsilon. \end{array} \tag{42}$$

For convenience, we define $\Lambda'=\{q\in\mathbb{R}^m:\omega_1\|q\|_1+\omega_2\|q\|_2^2\le\epsilon\}$.

**(i)** is obvious, we only need to prove **(ii)**.

**(ii).** For constrained problem (42), the KKT conditions are as follows

$$0\in q_i-p_i+\beta\omega_1\partial\|q_i\|_1+2\beta\omega_2 q_i,\ i=1,\cdots,m, \tag{43a}$$

$$\beta(\epsilon-\omega_1\|q\|_1-\omega_2\|q\|_2^2)=0,\ \beta\ge0, \tag{43b}$$

$$\omega_1\|q\|_1+\omega_2\|q\|_2^2\le\epsilon. \tag{43c}$$

If $\beta=0$, (43a) gives $q=p$, which violates condition (43c). Therefore, $\beta>0$, and (43c) holds with equality. Then, the optimal solution of (42) is given by

$$q(\beta)=\frac{1}{1+2\beta\omega_2}\Gamma_{\beta\omega_1}(p). \tag{44}$$

Furthermore, $q(\beta)$ has to satisfy

$$\omega_1\|q(\beta)\|_1+\omega_2\|q(\beta)\|_2^2=\epsilon. \tag{45}$$

Denote $l(\beta)=\omega_1\|q(\beta)\|_1+\omega_2\|q(\beta)\|_2^2-\epsilon$. We have the following observations.

(a) $l(\beta)$ is a monotonically decreasing continuous function at $\beta \in [0, +\infty)$; (b) $l(0) = w_1\|p\|_1 + w_2\|p\|_2^2 - \epsilon > 0$, $l(\frac{|p_m^\uparrow|}{\omega_1}) = -\epsilon < 0$; (c) $l$ is piecewise smooth with nonsmooth points $\frac{|p_i^\uparrow|}{\omega_1}$, $i = 1, \cdots, m$. Items (a) and (b) imply that (45) has unique root in $\left(0, \frac{|p_m^\uparrow|}{\omega_1}\right)$. We denote it as $\beta^{k+1}$. Once $\beta^{k+1}$ is obtained, $q(\beta^{k+1})$ can be calculated accordingly. Therefore, $r^{k+1} = q(\beta^{k+1}) + b$ and $y^{k+1} = r^{k+1} \circ e^{i\theta^{k+1}}$, which gives Lemma 1.

Next, we take a further look at $\beta^{k+1}$. Based on (a) - (c), we can find a unique $\beta$ satisfying (45) in one of the intervals $\left(0, \frac{|p_1^\uparrow|}{\omega_1}\right], \left(\frac{|p_1^\uparrow|}{\omega_1}, \frac{|p_2^\uparrow|}{\omega_1}\right], \cdots, \left(\frac{|p_{m-1}^\uparrow|}{\omega_1}, \frac{|p_m^\uparrow|}{\omega_1}\right]$. If the wanted $\beta$ falls into the interval $\left(\frac{|p_j^\uparrow|}{\omega_1}, \frac{|p_{j+1}^\uparrow|}{\omega_1}\right]$ ($j = 0, 1, \ldots, m-1$ with $\frac{|p_0^\uparrow|}{\omega_1}$ denotes 0), then (45) reduces to Lemma 1, the closed form of $\beta$ can be solved easily. The proof is completed.

The remaining problem is to determine the interval $\left(\frac{|p_j^\uparrow|}{\omega_1}, \frac{|p_{j+1}^\uparrow|}{\omega_1}\right]$ which $\beta$ falls into. By the intermediate value theorem, if we can find out two successive nodes $\frac{|p_j^\uparrow|}{\omega_1}$ and $\frac{|p_{j+1}^\uparrow|}{\omega_1}$ such that $l\left(\frac{|p_j^\uparrow|}{\omega_1}\right) > 0$ and $l\left(\frac{|p_{j+1}^\uparrow|}{\omega_1}\right) \leq 0$, then there exists a zero point of $p(\beta)$ on the interval $\left(\frac{|p_j^\uparrow|}{\omega_1}, \frac{|p_{j+1}^\uparrow|}{\omega_1}\right]$. The binary search algorithm stated in Algorithm 2 can be used to find out such an interval $\left(\frac{|p_j^\uparrow|}{\omega_1}, \frac{|p_{j+1}^\uparrow|}{\omega_1}\right]$.

**Algorithm 2:** Binary search algorithm to find the interval $\left(\frac{|p_j^\uparrow|}{\omega_1}, \frac{|p_{j+1}^\uparrow|}{\omega_1}\right)$

1 Set $L = 0, R = m$.
2 **while** $R > L + 1$ **do**
3 $\quad j = L + \lfloor \frac{R-L}{2} \rfloor$
4 $\quad$ If $l\left(\frac{|p_j^\uparrow|}{\omega_1}\right) < 0$ , $R = j$; else $L = j$
5 Output: $\beta$ locates in the interval $\left(\frac{|p_L^\uparrow|}{\omega_1}, \frac{|p_{L+1}^\uparrow|}{\omega_1}\right]$ and is the root of the quadratic equation with $j = L$ in Lemma 1.

*7.2. Proofs of Theorem* 1

We first prove a lemma that shows the decay of the objective function values of the iterative sequences. After some simplification, the Taylor's expansion of $F_2(\alpha, y)$ with respect to $\alpha$ takes the form

$$F_2(\alpha + \Delta\alpha, y) = F_2(\alpha, y) + \langle \nabla_\alpha^w F_2(\alpha, y), \Delta\alpha \rangle_w + \Delta\bar{\alpha} H \Delta\alpha. \tag{46}$$

**Lemma 2.** *Let* $\{(\alpha^k, y^k, \lambda^k)\}$ *be generated by* (17) *with* $t > 2\lambda_{\max}$. *Then it holds*

$$f(\alpha^k, y^{k-1}, \lambda^{k-1}) - f(\alpha^{k+1}, y^k, \lambda^k) \geq (\frac{t}{2} - \lambda_{\max})\|\alpha^{k+1} - \alpha^k\|^2. \tag{47}$$

PROOF. Since $\alpha^{k+1}$ is the optimal solution of the subproblem

$$\min_{\alpha\in\mathbb{C}^L} \langle \nabla_\alpha^w F_2(\alpha^k, y^k), \alpha - \alpha^k\rangle_w + F_1(\alpha, \lambda^k) + \frac{t}{2}\|\alpha - \alpha^k\|^2,$$

we have

$$\langle \nabla_\alpha^w F_2(\alpha^k, y^k), \alpha^{k+1} - \alpha^k\rangle_w + F_1(\alpha^{k+1}, \lambda^k) + \frac{t}{2}\|\alpha^{k+1} - \alpha^k\|^2 \le F_1(\alpha^k, \lambda^k).$$

Together with (46), it gives

$$\begin{aligned}
& f(\alpha^{k+1}, y^k, \lambda^k) = F_1(\alpha^{k+1}, \lambda^k) + F_2(\alpha^{k+1}, y^k) \\
= \;& F_1(\alpha^{k+1}, \lambda^k) + F_2(\alpha^k, y^k) + \langle \nabla_\alpha^w F_2(\alpha, y), \alpha^{k+1} - \alpha^k\rangle_w \\
& +(\bar\alpha^{k+1} - \bar\alpha^k)H(\alpha^{k+1} - \alpha^k) \\
\le \;& F_2(\alpha^k, y^k) + F_1(\alpha^k, \lambda^k) + (\bar\alpha^{k+1} - \bar\alpha^k)H(\alpha^{k+1} - \alpha^k) - \frac{t}{2}\|\alpha^{k+1} - \alpha^k\|^2 \\
\le \;& f(\alpha^k, y^k, \lambda^k) + (\lambda_{\max} - \frac{t}{2})\|\alpha^{k+1} - \alpha^k\|^2.
\end{aligned}$$

That is,

$$f(\alpha^k, y^k, \lambda^k) - f(\alpha^{k+1}, y^k, \lambda^k) \ge (\frac{t}{2} - \lambda_{\max})\|\alpha^{k+1} - \alpha^k\|^2. \tag{48}$$

Moreover, as $y^k$ minimizes $F_2(\alpha^k, y)$ and $\lambda^k$ minimizes $F_1(\alpha^k, \lambda)$, there holds

$$f(\alpha^k, y^{k-1}, \lambda^{k-1}) - f(\alpha^k, y^k, \lambda^k) \ge 0. \tag{49}$$

Combining (48) and (49), the inequality (47) is proved.

Now using Lemma 2, we give a proof of Theorem 1 as follows.

PROOF. Let $g(\alpha, y, \lambda) = f(\alpha, y, \lambda) + \delta_\Delta(y) + \delta_{\mathbb{R}_+^L}(\lambda)$, where $\delta_\Delta(y)$ and $\delta_{\mathbb{R}_+^L}(\lambda)$ are the indicator functions of the amplitude constraint set $\Delta$ and the set $\mathbb{R}_+^L$, respectively. Then $g(\alpha, y^k, \lambda^k) = f(\alpha, y^k, \lambda^k)$ for the sequences $\{y^k\}$ generated by (17), as $y^k \in \Delta$ and $\lambda^k \in \mathbb{R}_+^L$ for all $k \in \mathbb{N}$. Since $g$ is semi-algebraic, it satisfies the Kurdyka-Łojasievicz property [40]. The proximal alternating minimization (17) yields

$$0 \in \nabla_\alpha^w F_2(\alpha^{k+1}, y^k) + \frac{t}{2}(\bar\alpha^{k+1} - \bar\alpha^k) + \partial_\alpha^w F_1(\alpha^{k+1}, \lambda^k), \tag{50a}$$

$$0 \in \nabla_y^w F_2(\alpha^k, y^k) + \partial_y \delta_\Delta(y^k), \tag{50b}$$

$$0 \in \nabla_\lambda F_1(\alpha^k, \lambda^k) + \partial_\lambda \delta_{\mathbb{R}_+^L}(\lambda^k). \tag{50c}$$

Thus, we have

$$\Psi(\alpha^{k+1}, y^k, \lambda^k) = \begin{pmatrix} \frac{t}{2}(\bar\alpha^k - \bar\alpha^{k+1}) \\ \nabla_y^w F_2(\alpha^k, y^k) - \nabla_y^w F_2(\alpha^{k+1}, y^k) \\ \nabla_\lambda F_1(\alpha^k, \lambda^k) - \nabla_\lambda F_1(\alpha^{k+1}, \lambda^k) \end{pmatrix} \in \partial g(\alpha^{k+1}, y^k, \lambda^k). \tag{51}$$

A straightforward calculation gives

$$\begin{aligned}\|\nabla_y^w F_2(\alpha^k, y^k) - \nabla_y^w F_2(\alpha^{k+1}, y^k)\|_2 &= 4\|AW(\alpha^k - \alpha^{k+1})\|_2 \\ &\leq 4\|AW\|_2\|\alpha^k - \alpha^{k+1}\|_2\end{aligned} \tag{52}$$

and

$$\begin{aligned}&\|\nabla_\lambda F_1(\alpha^k, \lambda^k) - \nabla_\lambda F_1(\alpha^{k+1}, \lambda^k)\|_2^2 \\ =\ & \textstyle\sum_{i=1}^L \left[\left(\rho_1'(\lambda_i^k)(|\alpha_i^k| - |\alpha_i^{k+1}|)\right) + \left(\rho_2'(\lambda_i^k)(|\alpha_i^k|^2 - |\alpha_i^{k+1}|^2)\right)\right]^2 \\ =\ & \textstyle\sum_{i=1}^L \left[\rho_1'(\lambda_i^k) + \rho_2'(\lambda_i^k)(|\alpha_i^k| + |\alpha_i^{k+1}|)\right]^2 (|\alpha_i^k| - |\alpha_i^{k+1}|)^2 \\ \leq\ & \textstyle\sum_{i=1}^L \left[\rho_1'(\lambda_i^k) + \rho_2'(\lambda_i^k)(|\alpha_i^k| + |\alpha_i^{k+1}|)\right]^2 |\alpha_i^k - \alpha_i^{k+1}|^2 \\ \leq\ & M\|\alpha^k - \alpha^{k+1}\|_2^2,\end{aligned} \tag{53}$$

where $M = \max\{\left[\rho_1'(\lambda_i^k) + \rho_2'(\lambda_i^k)(|\alpha_i^k| + |\alpha_i^{k+1}|)\right]^2 : 1 \leq i \leq L, k \in \mathbb{N}\}$. It is clear that $g(\alpha, y, \lambda)$ is coercive. On the other hand, Lemma 2 states the decay of $g(\alpha^{k+1}, y^k, \lambda^k)$ equivalently, so $\{\lambda^k\}$ and $\{\alpha^k\}$ are bounded. In addition, $\rho_1'$ and $\rho_2'$ are smooth functions on $\mathbb{R}$. Thus, $M$ is finite. Combining (51), (52), and (53), we have

$$\begin{aligned}\|\Psi(\alpha^{k+1}, y^k, \lambda^k)\|_2 \leq & \frac{t}{2}\|\bar{\alpha}^k - \bar{\alpha}^{k+1}\|_2 + \|\nabla_y^w F_2(\alpha^k, y^k) - \nabla_y^w F_2(\alpha^{k+1}, y^k)\|_2 \\ & + \|\nabla_\lambda F_1(\alpha^k, \lambda^k) - \nabla_\lambda F_1(\alpha^{k+1}, \lambda^k)\|_2 \\ \leq & \left(\frac{t}{2} + 4\|AW\|_2 + \sqrt{M}\right)\|\alpha^k - \alpha^{k+1}\|_2,\end{aligned} \tag{54}$$

which means a subgradient lower bound for the iteration gaps similar as Lemma 4 in [40]. Together with Lemma 2, we can conclude that $\alpha^k$ is convergent using the same argument as Theorem 1 in [40].

Next, we prove that $\{y^k\}$ is convergent. In Lemma 1, $y^k$ is given by

$$y^k = r^k \circ \frac{h^k}{|h^k|}.$$

First, we show that $\{r^k\}$ is convergent. We know that $r^k$ is the unique projection of $|h^k|$ on convex set $\Delta'$, where $h^k = AW^T\alpha^k$. As $\{\alpha^k\}$ is convergent, it is obvious that $\{h^k\}$ is also convergent. Let $\hat{h}$ denote the limit of $\{h^k\}$ and let $\hat{r}$ denote the projection of $\hat{h}$ on $\Delta'$. Because $\Delta'$ is closed and convex, the projection operator $\mathcal{P}_{\Delta'}$ is continuous and nonexpansive. Thus we have

$$\|r^k - \hat{r}\|_2 = \|\mathcal{P}_{\Delta'}(h^k) - \mathcal{P}_{\Delta'}(\hat{h})\|_2 \leq \|h^k - \hat{h}\|_2 \to 0.$$

That is, $\{r^k\}$ converges to $\hat{r}$. In particular, it holds $r^k \to 0$ if $h^k \to 0$. As a consequence, $\{y^k\}$ always converges. Moreover, the limit of $\{y^k\}$, denoted by $\hat{y}$, is still the optimal solution of

$$\hat{y} = \arg\min_{y \in \Delta} \|y - \hat{h}\|. \tag{55}$$

Let $\hat{\alpha}$ denote the limit of $\{\alpha^k\}$. If $\hat{\alpha}_i \neq 0$, there exits a positive number $K$ such that $\alpha_i^k \neq 0$ for all $k > K$. When Assumption 1 is satisfied, the condition (15) gives

$$\rho_1''(\tilde{\lambda}_i^k) + \rho_2''(\tilde{\lambda}_i^k)|\alpha_i^{k+1}| \neq 0, \ \forall k > K. \tag{56}$$

Together with (23), we can conclude that $\tilde{\lambda}_i^k$ is a continuous function of $\alpha_i^{k+1}$ using the implicit function theorem. As the maximum function is also continuous, $\lambda_i^k$ is continuous with respect to $\alpha_i^{k+1}$. Therefore, $\{\lambda_i^k\}$ is convergent given that $\{\alpha_i^k\}$ is convergent. Let $\hat{\lambda}_i$ denote the limit of $\{\lambda_i^k\}$. Taking limits on (22), we get that $\hat{\lambda}_i = \max\{0, \hat{\tilde{\lambda}}_i\}$, where $\hat{\tilde{\lambda}}_i$ satisfies

$$\rho_1'(\hat{\tilde{\lambda}}_i) + \rho_2'(\hat{\tilde{\lambda}}_i)|\hat{\alpha}_i| = 0.$$

It is clear that

$$\hat{\lambda}_i = \arg\min_{\lambda_i \in \mathbb{R}_+} \rho_1'(\lambda_i) + \rho_2'(\lambda_i)|\hat{\alpha}_i|. \tag{57}$$

If $\hat{\alpha}_i = 0$, let $\{\alpha_i^{k_j}\}$ denote all non-zero subsequences of $\{\alpha_i^k\}$. When $\{\alpha_i^{k_j}\}$ is empty or finite, the convergence of $\{\lambda_i^k\}$ is trivial. Otherwise, $\lambda_i^{k_j}$ satisfies (22) and

$$\lambda_i^k = \lambda_i^{k_j}, \ \text{when } k_j \leq k < k_{j+1}.$$

Hence, $\{\lambda_i^k\}$ is convergent if $\{\lambda_i^{k_j}\}$ is convergent. The proof of the convergence of $\{\lambda_i^{k_j}\}$ is similar as above and (57) still holds.

(ii) $F_1(\alpha, \lambda)$ and $F_2(\alpha, y)$ are both convex with respect to $\alpha$. We have

$$-\big(\nabla_\alpha^w F_2(\alpha^{k+1}, y^k) + \frac{t}{2}(\bar{\alpha}^{k+1} - \bar{\alpha}^k)\big) \in \partial_\alpha^w F_1(\alpha^{k+1}, \lambda^k). \tag{58}$$

For any $\alpha$, there holds

$$F_1(\alpha, \lambda^k) - F_1(\alpha^{k+1}, \lambda^k) + \langle \nabla_\alpha^w F_2(\alpha^{k+1}, y^k) + \frac{t}{2}(\bar{\alpha}^{k+1} - \bar{\alpha}^k), \alpha - \alpha^{k+1}\rangle_w \geq 0. \tag{59}$$

Taking limits on both sides of the above inequality, and using $\alpha^{k+1} - \alpha^k \to 0$, we obtain

$$F_1(\alpha, \hat{\lambda}) - F_1(\hat{\alpha}, \hat{\lambda}) + \langle \nabla_\alpha^w F_2(\hat{\alpha}, \hat{y}), \alpha - \hat{\alpha}\rangle_w \geq 0, \tag{60}$$

which gives

$$0 \in \nabla_\alpha^w F_2(\hat{\alpha}, \hat{y}) + \partial_\alpha^w F_1(\hat{\alpha}, \hat{\lambda}). \tag{61}$$

Thus, (28) is proved.

We can get (30) directly from (57) and the relationship (29) has already been proved by (55). This completes the proof.

## References


[1] R. W. Harrison, Phase problem in crystallography, JOSA A 10 (5) (1993) 1046–1055.

[2] O. Bunk, A. Diaz, F. Pfeiffer, C. David, B. Schmitt, D. K. Satapathy, J. F. van der Veen, Diffractive imaging for periodic samples: retrieving one-dimensional concentration profiles across microfluidic channels, Acta Crystallographica Section A: Foundations of Crystallography 63 (4) (2007) 306–314.

[3] J. Miao, T. Ishikawa, Q. Shen, T. Earnest, Extending x-ray crystallography to allow the imaging of noncrystalline materials, cells, and single protein complexes, Annu. Rev. Phys. Chem. 59 (2008) 387–410.

[4] E. J. Candes, X. Li, M. Soltanolkotabi, Phase retrieval from coded diffraction patterns, Applied and Computational Harmonic Analysis 39 (2) (2015) 277–299.

[5] R. W. Gerchberg, W. O. Saxton, A practical algorithm for the determination of the phase from image and diffraction plane pictures, Optik 35 (1972) 237–246.

[6] J. R. Fienup, Phase retrieval algorithms: a comparison, Applied optics 21 (15) (1982) 2758–2769.

[7] H. H. Bauschke, P. L. Combettes, D. R. Luke, Phase retrieval, error reduction algorithm, and fienup variants: a view from convex optimization, JOSA A 19 (7) (2002) 1334–1345.

[8] H. H. Bauschke, P. L. Combettes, D. R. Luke, Hybrid projection–reflection method for phase retrieval, JOSA A 20 (6) (2003) 1025–1034.

[9] Z. Wen, C. Yang, X. Liu, S. Marchesini, Alternating direction methods for classical and ptychographic phase retrieval, Inverse Problems 28 (11) (2012) 115010.

[10] E. J. Candes, X. Li, M. Soltanolkotabi, Phase retrieval via wirtinger flow: Theory and algorithms, Information Theory, IEEE Transactions on 61 (4) (2015) 1985–2007.

[11] C. Ma, X. Liu, Z. Wen, Convergent levenberg-marquardt method for phase retrieval, http://www.optimization-online.org/DB_HTML\2016\06\5522.html.

[12] Y. Shechtman, A. Beck, Y. C. Eldar, Gespar: Efficient phase retrieval of sparse signals, IEEE Transactions on Signal Processing 62 (4) (2013) 928 – 938.

[13] A. Chai, M. Moscoso, G. Papanicolaou, Array imaging using intensity-only measurements, Inverse Problems 27 (1) (2010) 015005.

[14] E. J. Candes, T. Strohmer, V. Voroninski, Phaselift: Exact and stable signal recovery from magnitude measurements via convex programming, Communications on Pure and Applied Mathematics 66 (8) (2013) 1241–1274.

[15] E. J. Candes, Y. C. Eldar, T. Strohmer, V. Voroninski, Phase retrieval via matrix completion, SIAM Review 57 (2) (2015) 225–251.

[16] F. Fogel, I. Waldspurger, A. d'Aspremont, Phase retrieval for imaging problems, arXiv preprint arXiv:1304.7735.

[17] I. Waldspurger, A. d'Aspremont, S. Mallat, Phase recovery, maxcut and complex semidefinite programming, Mathematical Programming 149 (1-2) (2015) 47–81.

[18] S. Bahmani, J. Romberg, Phase retrieval meets statistical learning theory: A flexible convex relaxationarXiv:1610.04210.
URL `https://arxiv.org/abs/1610.04210`

[19] P. Hand, V. Voroninski, Corruption robust phase retrieval via linear programmingarXiv:1612.03547.
URL `https://arxiv.org/abs/1612.03547`

[20] O. Dhifallah, C. Thrampoulidis, Y. M. Lu, Phase retrieval via linear programming: Fundamental limits and algorithmic improvementsarXiv:1710.05234.
URL `https://arxiv.org/abs/1710.05234`

[21] E. J. Candes, T. Tao, Near-optimal signal recovery from random projections: Universal encoding strategies?, IEEE transactions on information theory 52 (12) (2006) 5406–5425.

[22] E. Pauwels, A. Beck, Y. C. Eldar, S. Sabach, On fienup methods for regularized phase retrieval, arXiv preprint arXiv:1702.08339.

[23] A. Repetti, E. Chouzenoux, J.-C. Pesquet, A nonconvex regularized approach for phase retrieval, in: Image Processing (ICIP), 2014 IEEE International Conference on, IEEE, 2014, pp. 1753–1757.

[24] H. Chang, Y. Lou, M. K. Ng, T. Zeng, Phase retrieval from incomplete magnitude information via total variation regularization, SIAM Journal on Scientific Computing 38 (6) (2016) A3672–A3695.

[25] J.-F. Cai, R. Chan, L. Shen, Z. Shen, Restoration of chopped and nodded images by framelets, SIAM Journal on Scientific Computing 30 (3) (2008) 1205–1227.

[26] B. Dong, J. Li, Z. Shen, X-ray ct image reconstruction via wavelet frame based regularization and radon domain inpainting, Journal of Scientific Computing 54 (2-3) (2013) 333–349.

[27] B. Dong, Z. Shen, Mra-based wavelet frames and applications: Image segmentation and surface reconstruction, in: SPIE Defense, Security, and Sensing, International Society for Optics and Photonics, 2012, pp. 840102–840102.

[28] Z. Shen, K.-C. Toh, S. Yun, An accelerated proximal gradient algorithm for frame-based image restoration via the balanced approach, SIAM Journal on Imaging Sciences 4 (2) (2011) 573–596.

[29] M. Elad, P. Milanfar, R. Rubinstein, Analysis versus synthesis in signal priors, Inverse problems 23 (3) (2007) 947.

[30] Z. Shen, Wavelet frames and image restorations, in: Proceedings of the International Congress of Mathematicians, Vol. 4, 2010, pp. 2834–2863.

[31] A. Ron, Z. Shen, Affine systems inl 2 ( d) ii: Dual systems, Journal of Fourier Analysis and applications 3 (5) (1997) 617–637.

[32] B. Dong, Z. Shen, et al., Mra based wavelet frames and applications, IAS Lecture Notes Series, Summer Program on "The Mathematics of Image Processing", Park City Mathematics Institute 19.

[33] K. Kreutz-Delgado, The complex gradient operator and the cr-calculus, arXiv preprint arXiv:0906.4835.

[34] Z. Gong, Z. Shen, K.-C. Toh, Image restoration with mixed or unknown noises, Multiscale Modeling & Simulation 12 (2) (2014) 458–487.

[35] J.-F. Cai, B. Dong, Z. Shen, Image restoration: a wavelet frame based model for piecewise smooth functions and beyond, Applied and Computational Harmonic Analysis.

[36] A. Guitton, W. W. Symes, Robust inversion of seismic data using the huber norm, Geophysics 68 (4) (2003) 1310–1319.

[37] I. Daubechies, R. DeVore, M. Fornasier, C. S. Güntürk, Iteratively reweighted least squares minimization for sparse recovery, Communications on Pure and Applied Mathematics 63 (1) (2010) 1–38.

[38] H. Attouch, J. Bolte, P. Redont, A. Soubeyran, Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the kurdyka-?ojasiewicz inequality, Mathematics of Operations Research 35 (2) (2010) 438–457.

[39] M. Li, Z. Fan, H. Ji, Z. Shen, Wavelet frame based algorithm for 3d reconstruction in electron microscopy, SIAM Journal on Scientific Computing 36 (1) (2014) B45–B69.

[40] J. Bolte, S. Sabach, M. Teboulle, Proximal alternating linearized minimization for nonconvex and nonsmooth problems, Mathematical Programming 146 (1-2) (2014) 459–494.