

ARGONNE NATIONAL LABORATORY
9700 South Cass Avenue
Argonne, IL 60439

**Visualizing and Improving the Robustness of Phase
Retrieval Algorithms**

Ash Tripathi, Sven Leyffer, Todd S. Munson, and Stefan M. Wild

Mathematics and Computer Science Division

Preprint ANL/MCS-P5272-0115

January 2015 (*Revised March 2015*)

This space is reserved for the Procedia header, do not use it

Visualizing and Improving the Robustness of Phase Retrieval Algorithms

Ashish Tripathi*, Sven Leyffer, Todd Munson, and Stefan M. Wild

Mathematics & Computer Science Division, Argonne National Laboratory, Argonne, IL, U.S.A.
{atripath, leyffer, tmunson, wild}@anl.gov

Abstract

Coherent x-ray diffractive imaging is a novel imaging technique that utilizes phase retrieval and nonlinear optimization methods to image matter at nanometer scales. We explore how the convergence properties of a popular phase retrieval algorithm, Fienup's HIO, behave by introducing a reduced dimensionality problem allowing us to visualize and quantify convergence to local minima and the globally optimal solution. We then introduce generalizations of HIO that improve upon the original algorithm's ability to converge to the globally optimal solution.

Keywords: Phase retrieval algorithms; inverse problems; nonlinear complex-valued optimization

1 Introduction

Coherent x-ray diffractive imaging (CXDI) is a microscopy technique that images a sample without optics [9]. In the experimental geometry shown in Fig. 1, monochromatic coherent plane wave x-rays interact with a sample to form an exit wave $\rho(r) \in \mathbb{C}^{m \times n}$, where $r \in R = \{(r_u, r_v) : u \in 0, \dots, n-1, v \in 0, \dots, m-1\}$ denotes a length scale that is the spatial resolution of the microscope and $mn \in \mathbb{Z}$ is the number of complex variables constituting the measured exit wave. A detector is placed in the far field so that a quantity proportional to the squared modulus of the Fourier transform of the exit wave, $\mathcal{F}[\rho]$, is measured. Thus, one obtains the measured diffraction pattern $D \propto |\mathcal{F}[\rho]|^2 \in \mathbb{R}_+^{m \times n}$, where $|\cdot|^2$ denotes the squared modulus $|a|^2 = \bar{a} \odot a$, with $\bar{\cdot}$ denoting the complex conjugate and \odot denoting the Hadamard (component wise) product for some $a \in \mathbb{C}^{m \times n}$.

CXDI attempts to recover the discrete representation of the exit wave, $\rho \in \mathbb{C}^{m \times n}$, from the measurement of its adequately sampled coherent diffraction pattern $D \in \mathbb{R}_+^{m \times n}$ using, for example, nonlinear optimization techniques. This approach solves the "phase problem," which comes from the inability of x-ray area detectors to measure a full complex-valued wave field. Recovery of the missing phase, $\tan^{-1}(\text{Im}(\rho) \div \text{Re}(\rho))$ (with the division taken component

*This material was based upon work supported by the U.S. Department of Energy, Office of Science, Advanced Scientific Computing Research program under contract number DE-AC02-06CH11357.

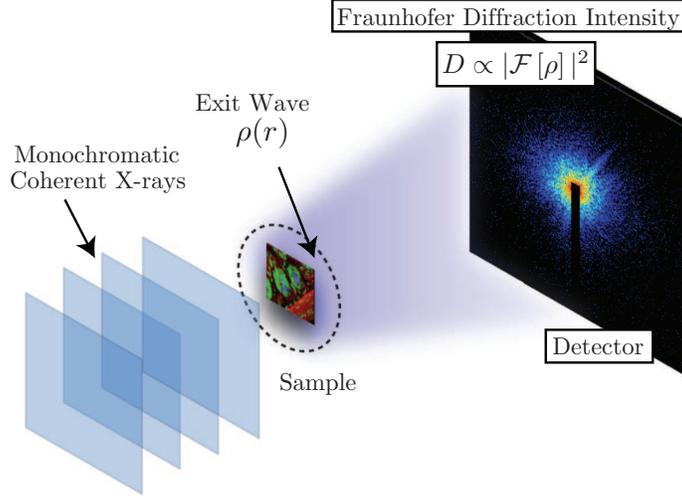


Figure 1: A CXDI experiment: Monochromatic coherent plane-wave x-rays interact with a sample and a detector is placed in the far field so that what is measured is proportional to the squared modulus of the Fourier transform of the exit wave ρ . The measurement is of size $m \times n$, and is determined by the number of pixels in the area detector used.

wise), starts with an initial exit wave guess and iteratively corrects the current exit wave iterate by using information known about the experiment. This information includes the measured diffraction intensities, D , as well as knowledge about the sample, most notably the support of the sample, which describes a subregion in the space R in whose complement the sample is known to not exist. CXDI has proved popular in practice, having been extended to many diverse samples and experimental regimes, and has been shown to yield a unique exit wave in special cases [1, 3, 11, 9].

In its simplest form, CXDI can be viewed as a feasibility problem [2],

$$\text{find some } \rho \in \mathcal{S} \cap \mathcal{M}, \quad (1)$$

which says that a solution is found when the recovered exit wave ρ satisfies constraints defined by the available information (in this case, the measurement and the support). The support constraint set \mathcal{S} is defined by

$$\mathcal{S} = \{\rho \in \mathbb{C}^{m \times n} : \rho(r) = 0 \ \forall r = (r_u, r_v) \notin S\},$$

where $S \subset R$ is the set of spatial indices corresponding to the support of the sample. The measurement constraint set \mathcal{M} is based on the measured coherent diffraction pattern intensity D and is defined by

$$\mathcal{M} = \left\{ \rho \in \mathbb{C}^{m \times n} : \rho = \mathcal{F}^{-1} \left[\sqrt{D} \odot \mathcal{F}[\rho] \div |\mathcal{F}[\rho]| \right] \right\}, \quad (2)$$

where the multiplication \odot and division \div are component wise: $[A \odot B \div C]_{u,v} = a_{u,v} \frac{b_{u,v}}{c_{u,v}}$.

For each of the sets \mathcal{S} and \mathcal{M} , we can define the respective projection operators

$$[\pi_{\mathcal{S}}\rho](r) = \begin{cases} \rho(r) & \text{if } r \in S \\ 0 & \text{if } r \notin S, \end{cases} \quad \text{and} \quad \pi_{\mathcal{M}}\rho = \mathcal{F}^{-1} \left[\sqrt{D} \odot \mathcal{F}[\rho] \div |\mathcal{F}[\rho]| \right]. \quad (3)$$

One of the simplest algorithms for approximately solving (1) is the alternating projection algorithm known as “error reduction” (ER) in the phase retrieval community [5]:

$$\rho^{(k+1)} = \pi_{\mathcal{S}}\pi_{\mathcal{M}}\rho^{(k)}, \quad k = 0, 1, \dots \quad (4)$$

The ER algorithm repeatedly applies the measurement projection $\pi_{\mathcal{M}}$ and the support projection $\pi_{\mathcal{S}}$ to iterate between the sample space and the diffraction space; see Fig. 2a. Since ER can be viewed as projected steepest descent for the problem $\min_{\rho} \{\|D - |\mathcal{F}[\rho]|^2\|_F^2 : \rho \in \mathcal{S}\}$, it can stagnate at stationary points that do not solve (1); see trajectory #1 in Fig. 2b. As we reaffirm in our numerical results, convergence clearly depends on the initial exit wave guess $\rho^{(0)}$.

Although several methods have been developed to overcome such stagnation, the current workhorse of experimentalists remains Fienup’s “hybrid input-output” (HIO) method [5]. HIO can be viewed as a version of the Douglas-Rachford algorithm for nonconvex problems [2] through the relaxation parameter $\beta \in \mathbb{R}$:

$$\rho^{(k+1)} = \pi_{\mathcal{S}}\pi_{\mathcal{M}}\rho^{(k)} + \pi_{\mathcal{S}^c}(\mathbf{1} - \beta\pi_{\mathcal{M}})\rho^{(k)}, \quad k = 0, 1, \dots, \quad (5)$$

where $\pi_{\mathcal{S}^c}$ is a binary operator orthogonal to $\pi_{\mathcal{S}}$, with $\mathcal{S}^c = \{\rho \in \mathbb{C}^{m \times n} : \rho \notin \mathcal{S}\} \cup \{\mathbf{0}\}$ and where $\pi_{\mathcal{S}} + \pi_{\mathcal{S}^c} = \mathbf{1}$, with $\mathbf{1} \in \mathbb{R}^{m \times n}$ denoting the matrix containing all ones. As we illustrate in our numerical results, HIO is generally more robust than ER in avoiding stagnation at nonglobal solutions; but as will be shown more robust algorithms exist.

The contributions of this paper are as follows. We explore generalized formulations of HIO as a saddle-point optimization problem and present optimization-based strategies for making HIO more efficient and robust in its ability to escape from nonglobal solutions. We propose a visualization mechanism for a low-dimensional problem that allows one to gain intuition about the saddle-point objective and an algorithm’s traversal of this space. We then examine the HIO variants developed with this mechanism.

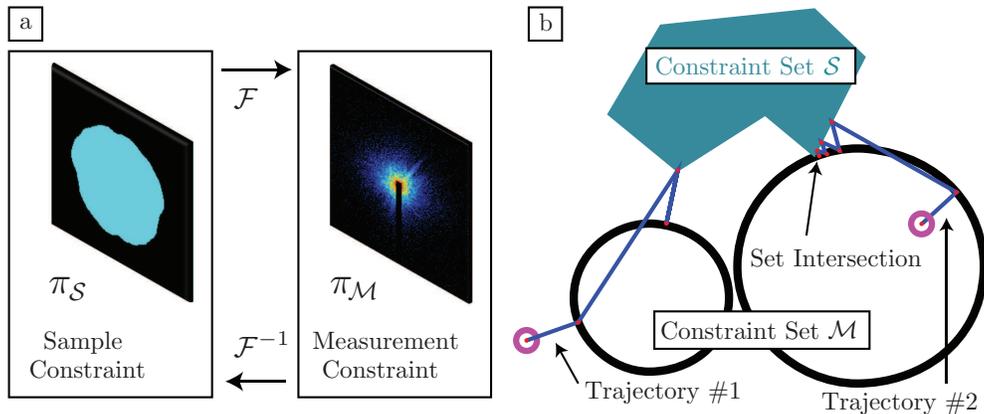


Figure 2: (a) Typical CXDI algorithms alternate between sample and diffraction-space representations using constraints on each representation and the Fourier and inverse Fourier transforms. (b) The performance of CXDI algorithms depends on the initialization $\rho^{(0)}$; different trajectories result using different initializations.

2 HIO and Saddle-Point Optimization

The HIO algorithm in (5) can be viewed as a heuristic for finding a Nash equilibrium (see, e.g., [4]) of the two-person game

$$\begin{aligned} \min_{\rho_s \in \mathcal{S}} \quad & \mathcal{L}(\rho_s + \rho_{\underline{s}}) \\ \max_{\rho_{\underline{s}} \in \mathcal{S}^c} \quad & \mathcal{L}(\rho_s + \rho_{\underline{s}}), \end{aligned} \quad (6)$$

where $\rho_s = \pi_{\mathcal{S}}\rho$ and $\rho_{\underline{s}} = \pi_{\mathcal{S}^c}\rho = (\mathbf{1} - \pi_{\mathcal{S}})\rho$ represent an orthogonal decomposition of $\mathbb{C}^{m \times n}$ and the objective function $\mathcal{L} : \mathbb{C}^{m \times n} \rightarrow \mathbb{R}$ is given by

$$\mathcal{L}(\rho) = \varepsilon_{\mathcal{M}}^2(\rho) - \varepsilon_{\mathcal{S}}^2(\rho) = \|\pi_{\mathcal{M}}\rho - \rho\|_F^2 - \|\pi_{\mathcal{S}}\rho - \rho\|_F^2. \quad (7)$$

In this game, one player seeks to minimize the objective by controlling ρ inside the support, while the second player seeks to maximize the objective by controlling ρ outside the support. Nash equilibria for (6) correspond to particular saddle points of the function $f(\rho_s, \rho_{\underline{s}}) = \mathcal{L}(\rho_s + \rho_{\underline{s}})$. This fact motivates algorithmic approaches that solve related saddle-point problems [7].

2.1 Two-Dimensional Search and HIO Generalizations

Using Wirtinger calculus (where $\nabla_{\bar{\rho}} = \frac{\partial}{\partial \bar{\rho}} = \frac{1}{2}(\frac{\partial}{\partial \text{Re}(\rho)} + i\frac{\partial}{\partial \text{Im}(\rho)})$; see [10]), we compute the gradient of (7) with respect to $\bar{\rho}$ as $\nabla_{\bar{\rho}}\mathcal{L}(\rho) = (\pi_{\mathcal{S}} - \pi_{\mathcal{M}})\rho$. This (complex-valued) gradient can be decomposed into parts inside and outside the support, respectively:

$$\delta_s = \pi_{\mathcal{S}}\nabla_{\bar{\rho}}\mathcal{L}(\rho) = (\pi_{\mathcal{S}} - \pi_{\mathcal{S}}\pi_{\mathcal{M}})\rho \quad \text{and} \quad \delta_{\underline{s}} = \pi_{\mathcal{S}^c}\nabla_{\bar{\rho}}\mathcal{L}(\rho) = -\pi_{\mathcal{S}^c}\pi_{\mathcal{M}}\rho, \quad (8)$$

where we have used the fact that $\pi_{\mathcal{S}^c}\pi_{\mathcal{S}} = \mathbf{0}$, $\pi_{\mathcal{S}}\pi_{\mathcal{S}} = \pi_{\mathcal{S}}$, and where $\mathbf{0} \in \mathbb{R}^{m \times n}$ is the matrix containing all zeros.

Taking a step along the steepest descent direction inside the support and a step along the steepest ascent direction outside the support would thus correspond to the combined direction $(-\delta_s, \delta_{\underline{s}})$. If we allow for unequal steplengths (α, β) along these respective orthogonal directions, we obtain the first-order update

$$\rho^{(k+1)} = \rho^{(k)} - \alpha\delta_s^{(k)} + \beta\delta_{\underline{s}}^{(k)} = (1 - \alpha)\rho^{(k)} + \alpha(\pi_{\mathcal{S}}\pi_{\mathcal{M}})\rho^{(k)} + \pi_{\mathcal{S}^c}(\alpha\mathbf{1} - \beta\pi_{\mathcal{M}})\rho^{(k)}, \quad (9)$$

where we have used the fact that $\pi_{\mathcal{S}} + \pi_{\mathcal{S}^c} = \mathbf{1}$. The special case where $\alpha = 1$ then corresponds to the HIO algorithm of (5).

A generalization of the HIO algorithm can be obtained by looking beyond the $\alpha = 1$ case and considering more general values for β , rather than using a fixed value taken from the typical range of $\beta \in [0.5, 1]$ as is enforced by practical HIO implementations [5, 7]. One way of obtaining (α, β) values in each iteration of the form (9) is to solve the two-dimensional version of (6) with the common objective $\psi_k(\alpha, \beta) = \mathcal{L}(\rho^{(k)} - \alpha\delta_s^{(k)} + \beta\delta_{\underline{s}}^{(k)})$.

Using the notation $\frac{\partial}{\partial a} = \partial_a$ and $\frac{\partial^2}{\partial a \partial b} = \partial_{ab}$, we desire (α, β) such that $\partial_{\alpha}\psi_k(\alpha, \beta) = \partial_{\beta}\psi_k(\alpha, \beta) = 0$ and $\partial_{\alpha\alpha}\psi_k(\alpha, \beta) \geq 0 \geq \partial_{\beta\beta}\psi_k(\alpha, \beta)$. One approach is to use a modified Newton method for the problem $\min_{\alpha, \beta} \Phi_k(\alpha, \beta)$:

$$\begin{bmatrix} \alpha_{j+1} \\ \beta_{j+1} \end{bmatrix} = \begin{bmatrix} \alpha_j \\ \beta_j \end{bmatrix} - \mu \begin{bmatrix} |\partial_{\alpha\alpha}\psi_k(\alpha_j, \beta_j)| & \partial_{\alpha\beta}\psi_k(\alpha_j, \beta_j) \\ \partial_{\beta\alpha}\psi_k(\alpha_j, \beta_j) & -|\partial_{\beta\beta}\psi_k(\alpha_j, \beta_j)| \end{bmatrix}^{-1} \begin{bmatrix} \partial_{\alpha}\psi_k(\alpha_j, \beta_j) \\ \partial_{\beta}\psi_k(\alpha_j, \beta_j) \end{bmatrix}, \quad (10)$$

where $\Phi_k(\alpha, \beta) = \|\nabla\psi_k(\alpha, \beta)\|^2 = |\partial_{\alpha}\psi_k(\alpha, \beta)|^2 + |\partial_{\beta}\psi_k(\alpha, \beta)|^2$. The form of the second-order matrix in (10) is chosen to obtain the proper inertia for a minimization with respect to α and a maximization with respect to β . The step length μ along the Newton-like direction in (10) can be determined by a line search (e.g., using the strong Wolfe conditions) for the objective $\Phi_k(\alpha, \beta)$; a similar approach is taken in [7]. An example of this process is shown in Fig. 3.

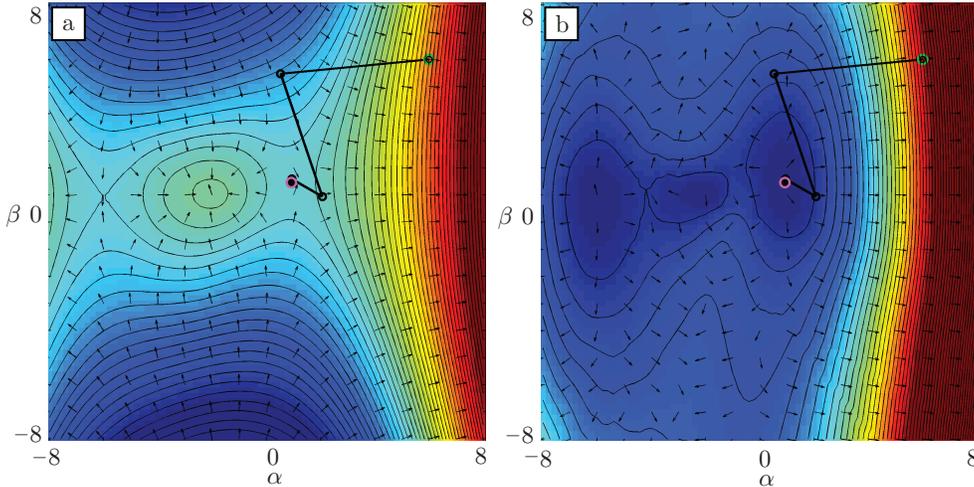


Figure 3: Simultaneous optimization of α and β by finding a particular saddle point of $\psi(\alpha, \beta)$. Contours of (a) the function $\psi(\alpha, \beta)$ and (b) the function $\Phi_k(\alpha, \beta)$. The trajectory using the modified Newton step in (10) is overlaid on both plots, with the green circle the initial (α_0, β_0) and the magenta circle the final (α_5, β_5) (after 5 iterations).

2.2 Quasi-Newton and Conjugate Gradient Update Directions

The bidirectional approach described above includes HIO as a special case, but one can also consider more general approaches to solving (6). We now propose two such approaches – based on L-BFGS and conjugate gradient (CG) direction steps, respectively – that use general directions d_k in the update

$$\rho^{(k+1)} = \rho^{(k)} + \alpha_k d_s^{(k)} + \beta_k d_s^{(k)}, \quad k = 0, 1, \dots, \quad (11)$$

instead of the gradient directions prescribed by (8) and (9). In all the results that follow, we initialize $d_s^{(0)} = -\delta_s^{(0)}$ and $d_s^{(0)} = \delta_s^{(0)}$.

The dimensionality of phase retrieval problems in typical experimental settings is on the order of $mn = 10^6$ complex-valued variables. Therefore, computing an approximation of the dense Hessian (with 10^{12} complex-valued variables) for use in quasi-Newton methods is prohibitively expensive in terms of storage. Instead, we look to limited-memory methods such as L-BFGS [8]. Our L-BFGS method follows the developments of [10] and is given in Algorithm 1. We note, for ease of exposition, that this algorithm works on the vectorized version, $\rho \in \mathbb{C}^{mn}$, and thus each of the quantities \mathbf{s}_{k-1} , \mathbf{y}_{k-1} , and \mathbf{g}_k are column vectors. Algorithm 1 can be used both inside ($\mathcal{A} = \mathcal{S}$) and outside ($\mathcal{A} = \mathcal{S}^c$) the support, with appropriate projection ($\pi_{\mathcal{S}}$ or $\pi_{\mathcal{S}^c}$) providing the required input. In our implementation, we keep the past $p = 5$ updates.

Algorithm 1 returns an approximate Newton step with the inertia of the quasi-Newton Hessian B_k determining whether one seeks a minimum or a maximum. We achieve the correct direction by an appropriate scaling of the initial quasi-Newton matrix (the identity matrix is used in our experiments). If the term

$$\frac{\langle \mathbf{y}_{k-1}, \mathbf{s}_{k-1} \rangle}{\|\mathbf{y}_{k-1}\|^2} = \frac{\text{Re}[\mathbf{y}_{k-1}^H \mathbf{s}_{k-1}]}{\text{Re}[\mathbf{y}_{k-1}^H \mathbf{y}_{k-1}]} \quad (12)$$

Algorithm 1 L-BFGS method (see, e.g., [10]) for complex-valued, vectorized variables.

Input: $\mathbf{g}_k = \pi_{\mathcal{A}} \nabla_{\bar{\rho}} \mathcal{L}(\rho^{(k)})$, $\{(\mathbf{y}_j = \mathbf{g}_{j+1} - \mathbf{g}_j, \mathbf{s}_j = \pi_{\mathcal{A}}(\rho^{(j+1)} - \rho^{(j)}))\}_{j=\max\{0, k-p\}}^{j=k-1}$, $p \geq 1$.

Output: $d^{(k)} = -\mathbf{d} = -B_k^{-1} \nabla_{\bar{\rho}} \mathcal{L}(\rho^{(k)})$

```

 $\mathbf{d} \leftarrow \mathbf{g}_k$ 
for  $j = k - 1, \dots, \max\{0, k - p\}$  do
     $\varrho_j \leftarrow (\langle \mathbf{y}_j, \mathbf{s}_j \rangle)^{-1}$ ;     $\nu_j \leftarrow \varrho_j \langle \mathbf{s}_j, \mathbf{d} \rangle$ ;     $\mathbf{d} \leftarrow \mathbf{d} - \nu_j \mathbf{y}_j$ 
end for
 $\mathbf{d} \leftarrow \frac{\langle \mathbf{y}_{k-1}, \mathbf{s}_{k-1} \rangle}{\|\mathbf{y}_{k-1}\|^2} \mathbf{d}$ 
for  $j = \max\{0, k - p\}, \dots, k - 1$  do
     $\xi \leftarrow \varrho_j \langle \mathbf{y}_j, \mathbf{d} \rangle$ ;     $\mathbf{d} \leftarrow \mathbf{d} + (\nu_j - \xi) \mathbf{s}_j$ 
end for

```

in Algorithm 1 is positive, where \cdot^H is the Hermitian transpose, then we are returning the quasi-Newton step in a downhill direction, whereas if (12) is negative, then we are returning the quasi-Newton step in an uphill direction.

We also consider nonlinear CG directions, which have the form

$$d_s^{(k+1)} = -\delta_s^{(k+1)} + \gamma_s^{(k+1)} d_s^{(k)} \quad \text{and} \quad d_{\bar{s}}^{(k+1)} = \delta_{\bar{s}}^{(k+1)} + \gamma_{\bar{s}}^{(k+1)} d_{\bar{s}}^{(k)}, \quad k = 0, 1, \dots, \quad (13)$$

with δ_s and $\delta_{\bar{s}}$ defined from (8). Several alternatives for the CG parameter γ exist (see, e.g., [6]), and we consider the seven variants listed in Table 1. We employ separate updates for the variables ρ_s and $\rho_{\bar{s}}$, so that $\gamma_s^{(k)}$ ($\gamma_{\bar{s}}^{(k)}$) is determined by using $\mathbf{g}_k = \delta_s^{(k)}$ ($\mathbf{g}_k = -\delta_{\bar{s}}^{(k)}$) and $\mathbf{d}_k = d_s^{(k)}$ ($\mathbf{d}_k = d_{\bar{s}}^{(k)}$). These two sets of choices are made based on whether we are updating in \mathcal{S} (minimizing) or in \mathcal{S}^c (maximizing).

3 Numerical Experiments with HIO Variants

We now examine the effectiveness of the methods described in Sec. 2 in terms of their robustness for solving the low-dimensional problem whose exit wave $\rho \in \mathbb{R}^{16 \times 16}$ and diffraction pattern $D \in \mathbb{R}_+^{16 \times 16}$ are shown in Figs. 4a and 4b, respectively. The exit wave is real-valued and consists of three pixels, $\rho(r_a) = 0.05$, $\rho(r_b) = 0.8$, and $\rho(r_c) = 0.125$, arranged in an upside-down-L shape. The remaining pixels are zero, and the correct support $S = \{r_a, r_b, r_c\}$ is assumed given.

Fletcher-Reeves (FR):	$\gamma = \frac{\ \mathbf{g}_{k+1}\ ^2}{\ \mathbf{g}_k\ ^2}$	Polak-Ribière (PR):	$\gamma = \frac{\langle \mathbf{g}_{k+1}, \mathbf{y}_k \rangle}{\ \mathbf{g}_k\ ^2}$
Hestenes-Stiefel (HS):	$\gamma = \frac{\langle \mathbf{g}_{k+1}, \mathbf{y}_k \rangle}{\langle \mathbf{d}_k, \mathbf{y}_k \rangle}$	Liu-Storey (LS):	$\gamma = \frac{\langle \mathbf{g}_{k+1}, \mathbf{y}_k \rangle}{\langle -\mathbf{d}_k, \mathbf{g}_k \rangle}$
Dai-Yuan (DY):	$\gamma = \frac{\ \mathbf{g}_{k+1}\ ^2}{\langle \mathbf{d}_k, \mathbf{y}_k \rangle}$	Conjugate Descent (CD):	$\gamma = \frac{\ \mathbf{g}_{k+1}\ ^2}{\langle -\mathbf{d}_k, \mathbf{g}_k \rangle}$
Hager-Zhang (HZ):	$\gamma = \frac{\langle \mathbf{y}_k - 2\mathbf{d}_k \frac{\ \mathbf{y}_k\ ^2}{\langle \mathbf{d}_k, \mathbf{y}_k \rangle}, \mathbf{g}_{k+1} \rangle}{\langle \mathbf{d}_k, \mathbf{y}_k \rangle}$		

Table 1: CG parameter expressions for the algorithms considered. We define $\mathbf{y}_k = \mathbf{g}_{k+1} - \mathbf{g}_k$, $\|a\|^2 = \langle a, a \rangle$, and the inner product $\langle a, b \rangle = e^T \text{Re}[\bar{a} \odot b] e$, where e is a generic vector of ones.

For visualization purposes, we treat $\rho(r_c)$ as known; this leaves us with a 255-complex-variable-dimensional problem, with the only two nonzeros being $\rho(r_a)$ and $\rho(r_b)$, a setting inspired by a similar synthetic problem in [7].

This problem allows us to visualize the solution in the two-dimensional subspace $(\rho(r_a), \rho(r_b))$, $\rho(r_c) = 0.125, \rho_s = 0$ where $\mathcal{L}(\rho)$ reduces to the modulus objective function $\varepsilon_{\mathcal{M}}^2(\rho) = \|\pi_{\mathcal{M}}\rho - \rho\|_F^2$ as a function of $\rho(r_a)$ and $\rho(r_b)$; see Fig. 4c. The minimum labeled m_G is the global minimum of this metric and corresponds to the input exit wave (with $\rho(r_a) = 0.05$ and $\rho(r_b) = 0.8$). The nonglobal minimum labeled m_1 arises because phase retrieval is in general insensitive to global phase shifts in the exit wave (i.e., $|\mathcal{F}[\rho]| = |\mathcal{F}[\rho e^{i\phi_0}]|$ for a constant phase shift $\phi_0 \in \mathbb{R}$). For the m_1 minimum, we have a phase shift of $\phi_0 = \pi$, which corresponds to the negative of the input exit wave ($\rho(r_a) = -0.05$ and $\rho(r_b) = -0.8$). However, such global phase shifts are not equivalent in our problem, since the value of $\rho(r_c)$ is known. The nonglobal minima labeled m_2 and m_3 arise because of Fourier transform symmetries, whereby $|\mathcal{F}[\rho]| = |\mathcal{F}[\chi]|$ when χ is $\bar{\rho}$ rotated by 180° . The minimum labeled m_2 corresponds to the exit wave $\bar{\rho}$ rotated by 180° , while the minimum labeled m_3 corresponds to the exit wave $-\bar{\rho}$ rotated by 180° . Similar to m_1 , the m_2 and m_3 minima are not global minima because of the Fourier transform symmetry-breaking effects of our knowledge of the value of $\rho(r_c)$.

Using this problem setup, we now propose a means of visualizing the exit wave recovered as a function of a methods starting values $(\rho^{(0)}(r_a), \rho^{(0)}(r_b))$. For a given method, we consider 441 starting points $(\rho^{(0)}(r_a), \rho^{(0)}(r_b))$ taken in steps of 0.15 from the box $[-1.5, 1.5]^2$. We then examine which of the $\varepsilon_{\mathcal{M}}^2$ minima (m_G, m_1, m_2 , or m_3) in the two-dimensional space $(\rho(r_a), \rho(r_b))$ the method converges to from the selected starting point. An example of this visualization is provided in Fig. 4d for the ER method from (4). Since the ER method is projected steepest descent, we expect to arrive at the minima closest to the starting point. This result is indeed seen in Fig. 4d, with the obtained minimum indicated by a color coding of the starting point.

We repeat these 441 runs using an implementation of each of the presented methods, with care taken to ensure consistent experimental conditions. We start with a $\rho^{(0)}$ of zeros, except for the prescribed $(\rho^{(0)}(r_a), \rho^{(0)}(r_b))$ values and the fixed $\rho^{(0)}(r_c)$ value. Each of the new variants uses initial search directions $d_s^{(0)} = -\delta_s^{(0)}$ and $d_s^{(0)} = \delta_s^{(0)}$ and in the first iteration determines optimal (α_0, β_0) using the method discussed in Sec. 2.1 to update $\rho^{(1)}$ as in (11). For subsequent iterations, we compute $\nabla_{\bar{\rho}}\mathcal{L}(\rho^{(k)})$ and use the L-BFGS or CG update to compute the new search directions $d^{(k)}$. Once this update is done, we determine optimal (α_k, β_k) and update $\rho^{(k+1)}$. After computing $d^{(k)}$ each iteration, we check the sign on the directional derivatives in \mathcal{S} and \mathcal{S}^c by computing $\text{Re}[\sum_r [\bar{\delta}^{(k)} \odot d^{(k)}](r)]$; if when updating in \mathcal{S} we have a positive directional derivative (are going uphill when we should be going down) or if updating in \mathcal{S}^c we have a negative directional derivative (are going downhill when we should be going up), we reset the offending update to be the standard HIO update in (8), i.e. steepest descent in \mathcal{S} and steepest ascent in \mathcal{S}^c . When determining optimal (α_k, β_k) , we allow only five iterations for the saddle-point optimization process in (10).

Our visualization in Fig. 4e shows that the HIO method from (9) with optimal (α, β) and search directions inside and outside the support given in (8) can avoid the local minima m_2 and m_3 , but the results also show that this HIO variant is susceptible to stagnation in the nonglobal minimum m_1 . HIO with optimal (α, β) can find the global minimum m_G about 75% of the time out of all 441 starting points explored. In Fig. 4f-i, we show results for some representative combinations of using CG and L-BFGS inside and outside the support together with optimal (α, β) . For example, in Fig. 4f we use CG search directions with the DY update (see Table 1) inside the support while using the FR update outside the support; clearly some combinations

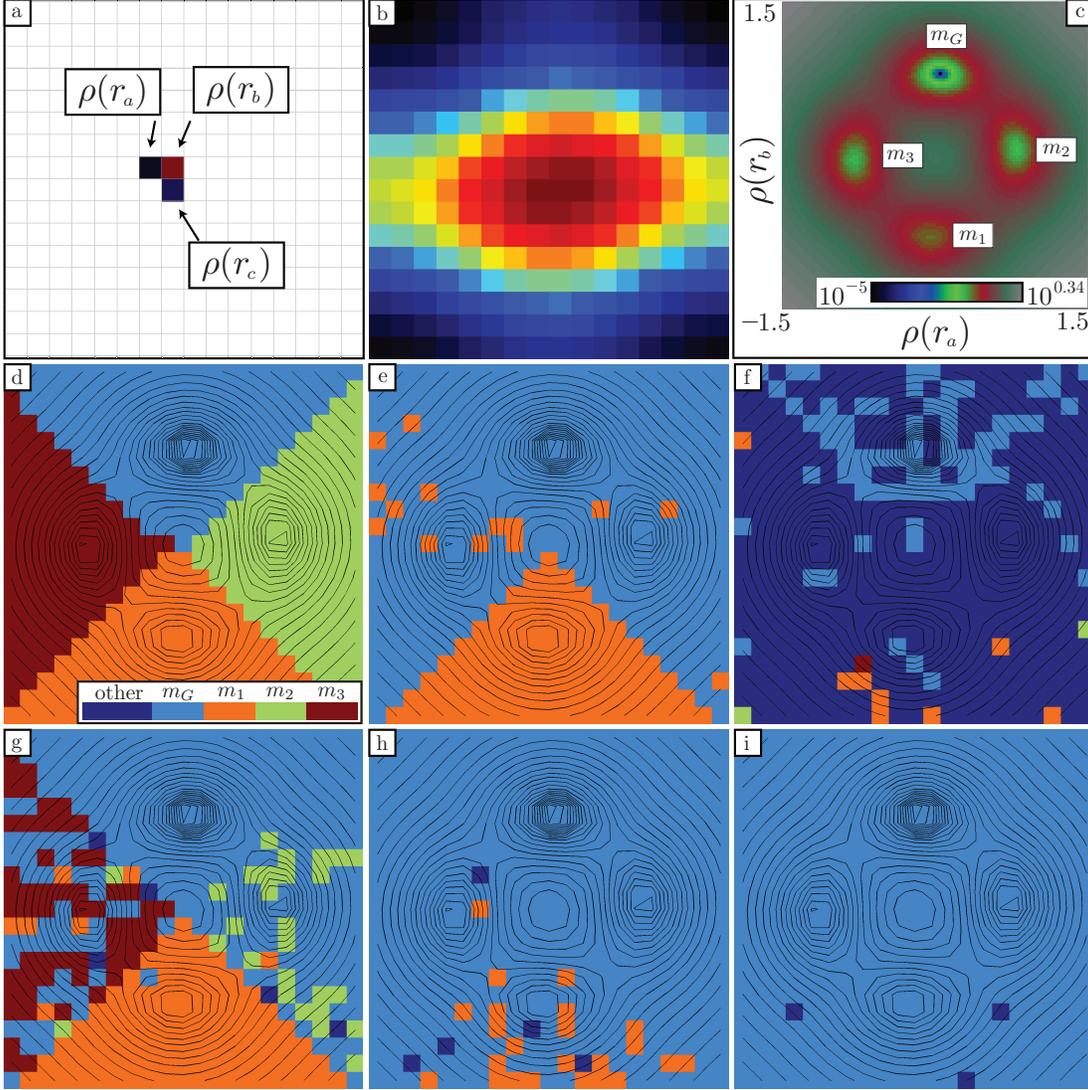


Figure 4: (a) The exit wave $\rho(r) \in \mathbb{R}^{16 \times 16}$ used. The top two pixels $\rho(r_a)$ and $\rho(r_b)$ are assumed unknown while the bottom pixel $\rho(r_c)$ is assumed known. (b) The diffraction pattern corresponding to the exit wave in (a). (c) As we have only two unknowns ($\rho(r_a), \rho(r_b)$), we can use brute force to compute what the modulus objective function $\varepsilon_{\mathcal{M}}^2(\rho) = \|\pi_{\mathcal{M}}\rho - \rho\|_F^2$ looks like, and this is shown here. (d) Use of the local minimizer ER (projected steepest descent) to attempt to solve the phase problem; depending on the starting point, we will end up in the closest of one of the four minima m_G, m_1, m_2 , or m_3 . (e-i) Convergence to these minima using saddle-point optimization to find optimal (α, β) and (e) standard HIO directions (8). Sometimes using CG update directions can make things worse: use of (f) Dai-Yuan in \mathcal{S} and Fletcher-Reeves in \mathcal{S}^c , and (g) Liu-Storey in \mathcal{S} and Dai-Yuan in \mathcal{S}^c . (h) Use of Hager-Zhang in \mathcal{S} and Polak Ribière in \mathcal{S}^c vastly improves the result in (e). (i) Use of L-BFGS in \mathcal{S} and Hestenes-Steifel in \mathcal{S}^c allow us to find m_G over virtually all starting points (99% success rate).

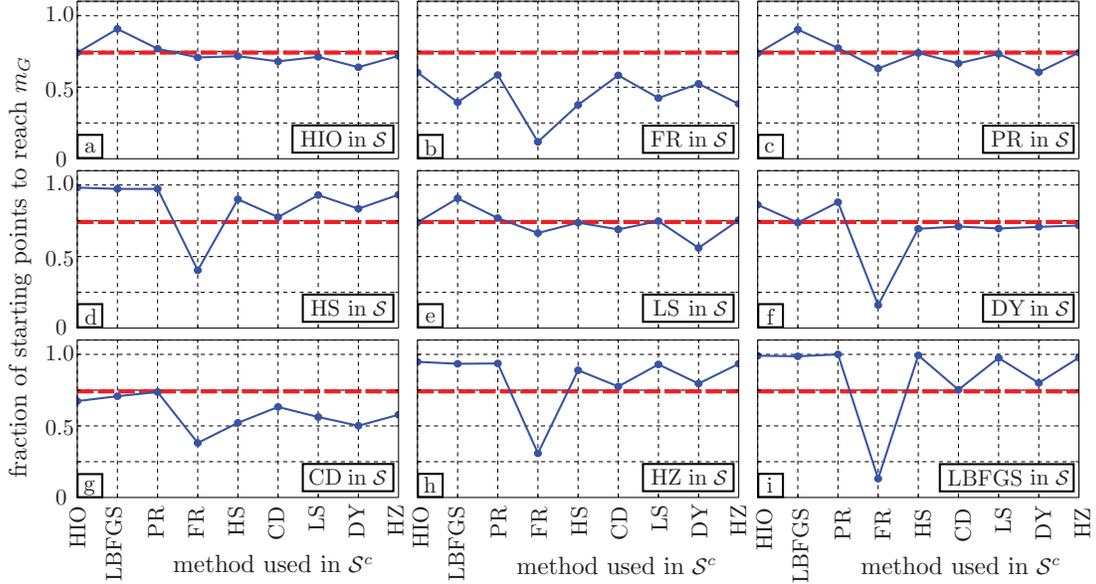


Figure 5: The fraction of starting points the global minimum m_G is recovered for combinations of normal HIO directions (9), CG direction updates (Table 1), and L-BFGS direction updates (Algorithm 1). (a) Use of normal HIO direction updates from (9) inside the support versus direction update method outside the support together with optimal (α, β) . (b-i) Inside the support, use of (b) Fletcher-Reeves (FR), (c) Polak-Ribière (PR), (d) Hestenes-Steifel (HS), (e) Liu-Storey (LS), (f) Dai-Yuan (DY), (g) Conjugate Descent (CD), (h) Hager-Zhang (HZ), and (i) L-BFGS direction updates versus direction update methods used outside the support. For comparison, the red dotted line is the fraction of times the global minimum was found by using normal HIO both inside and outside the support together with the optimal (α, β) .

of CG updates inside and outside of the support have significant adverse effects. In Fig. 4g we use the LS update inside the support and the DY update outside the support; while the use of these CG update parameters still has significant adverse effects, they are less severe than that in Fig. 4f. In Fig. 4h, we use the HZ update inside the support and the PR update outside the support, while in Fig. 4i we use the L-BFGS direction update from (1) inside the support and the HS update from Table 1 outside the support; these are representatives of CG and L-BFGS update combinations that significantly improve the algorithm’s beneficial ability to converge to the global solution.

In Fig. 5 we summarize the results for the 81 variants obtained by coupling different approaches for the updates inside and outside the support. The plots show the fraction of the 441 starting points in the interval where the global minimum m_G is obtained. From these results we can determine whether mixing L-BFGS and CG direction updates in \mathcal{S} and \mathcal{S}^c is an effective way of obtaining more robust performance. The CG methods of PR, LS, and DY in Figs. 5c, e, and f, respectively, appear to have similar to slightly worse behavior when compared with the normal HIO update in Fig. 5a. The CG method of FR in Fig. 5b appears to have only harmful effects on convergence to m_G when used in \mathcal{S} and generally harmful effects when used in \mathcal{S}^c . The CG method of CD in Fig. 5g generally has harmful effects when used in \mathcal{S} and indifferent effects when used in \mathcal{S}^c . The CG methods of HZ and HS in Fig. 5d and Fig. 5h,

respectively, and the L-BFGS method in Fig. 5i all have beneficial effects, with L-BFGS being the most effective. In some cases these variants converge to the global minimum from almost all the 441 starting points.

4 Outlook

We have explored how a popular phase retrieval algorithm, Fienup’s HIO, behaves when updates to an exit wave inside and outside the support are generalized by using nonlinear conjugate gradient and limited-memory quasi-Newton updates along with an optimal weighting of these updates. We have examined the robustness of these methods by studying a low-dimensional, synthetic problem in order to visualize how these generalized updates can improve or harm convergence to an optimal solution. Our study has shown that certain combinations of CG and L-BFGS updates dramatically improve the ability of an algorithm to recover the prescribed exit wave. We also have shown that some combinations should be avoided because of limitations of their robustness.

A standard procedure of experimentalists using HIO is to use many different starting guesses for the exit wave, to run many different independent trials, and then to compare the exit waves recovered from these runs. Solutions obtained in this way are invariably different, and what are considered “good” solutions is sometimes left to more subjective, qualitative criteria. We anticipate that application of the generalized updates presented here will increase the confidence of experimentalists to reduce the number of starting points considered.

References

- [1] B. Abbey, K.A. Nugent, G.J. Williams, J.N. Clark, A.G. Peele, M.A. Pfeifer, M. de Jonge, and I. McNulty. Keyhole coherent diffractive imaging. *Nature Phys.*, 4(5):394–398, 2008.
- [2] H. H. Bauschke, P. L. Combettes, and D. R. Luke. Phase retrieval, error reduction algorithm, and Fienup variants: A view from convex optimization. *J. Opt. Soc. Am. A*, 19(7):1334–1345, 2002.
- [3] M. Dierolf, A. Menzel, P. Thibault, P. Schneider, C.M. Kewish, R. Wepf, O. Bunk, and F. Pfeiffer. Ptychographic x-ray computed tomography at the nanoscale. *Nature*, 467(7314):436–439, 2010.
- [4] M. C. Ferris and J. S. Pang. Engineering and economic applications of complementarity problems. *SIAM Rev.*, 39(4):669–713, 1997.
- [5] J. R. Fienup. Phase retrieval algorithms: A comparison. *Appl. Optics*, 21(15):2758–2769, 1982.
- [6] W. W. Hager and H. Zhang. A survey of nonlinear conjugate gradient methods. *Pacific J. Optim.*, 2(1):35–58, 2006.
- [7] S. Marchesini. Phase retrieval and saddle-point optimization. *J. Opt. Soc. Am. A*, 24(10):3289–3296, Oct 2007.
- [8] J. Nocedal and S.J. Wright. *Numerical Optimization*. Springer-Verlag, New York, 1999.
- [9] D. Paganin. *Coherent X-ray Optics*. Oxford Press, 2006.
- [10] L. Sorber, M. Barel, and L. Lathauwer. Unconstrained optimization of real functions in complex variables. *SIAM J. Optimization*, 22(3):879–898, 2012.
- [11] A. Tripathi, J. Mohanty, S.H. Dietze, O.G. Shpyrko, E. Shipton, E.E. Fullerton, S.S. Kim, and I. McNulty. Dichroic coherent diffractive imaging. *Proc. Natl. Acad. Sci. USA*, 108(33):13393–13398, 2011.