Proximal Heterogeneous Block Input-Output Method and application to Blind Ptychographic Diffraction Imaging

Robert Hesse[*] D. Russell Luke[†] Shoham Sabach[‡] Matthew K. Tam[§]

February 27, 2015

Abstract

We propose a general alternating minimization algorithm for nonconvex optimization problems with separable structure and nonconvex coupling between blocks of variables. To fix our ideas, we apply the methodology to the problem of blind ptychographic imaging. Compared to other schemes in the literature, our approach differs in two ways: (i) it is posed within a clear mathematical framework with practically verifiable assumptions, and (ii) under the given assumptions, it is provably convergent to critical points. A numerical comparison of our proposed algorithm with the current state-of-the-art on simulated and experimental data validates our approach and points toward directions for further improvement.

Keywords: Alternating minimization, deconvolution, Kurdyka-Łojasiewicz, nonconvex-nonsmooth minimization, ptychography.

1 Introduction

We consider algorithms for nonconvex constrained optimization problems of the following form

$$\text{Find } (\overline{x},\overline{y},\overline{z}) \in \arg\min\left\{F(x,y,z) \mid (x,y,z) \in C \equiv X \times Y \times Z\right\}. \tag{1.1}$$

Here $X \times Y \times Z \subset \mathbb{R}^p \times \mathbb{R}^q \times \mathbb{R}^r$ (that is, the constraints apply to disjoint blocks of variables) and F is a nonlinear penalty function characterizing the coupling between the blocks of variables. It will be convenient to reformulate problem (1.1) using indicator functions. The indicator function of a set C is defined as $\iota_C(x) = 0$ for $x \in C$ and $\iota_C(x) = +\infty$ for $x \notin C$. Define

$$\Psi(x,y,z) \equiv F(x,y,z) + \iota_X(x) + \iota_Y(y) + \iota_Z(z). \tag{1.2}$$

[*] Institut für Numerische und Angewandte Mathematik, Universität Göttingen, Lotzestr. 16–18, 37083 Göttingen, Germany. E-mail: hesse@math.uni-goettingen.de.

[†] Institut für Numerische und Angewandte Mathematik, Universität Göttingen, Lotzestr. 16–18, 37083 Göttingen, Germany. E-mail: r.luke@math.uni-goettingen.de.

[‡] Institut für Numerische und Angewandte Mathematik, Universität Göttingen, Lotzestr. 16–18, 37083 Göttingen, Germany. E-mail: s.sabach@math.uni-goettingen.de.

[§] CARMA Centre, University of Newcastle, Callaghan, NSW 2308, Australia. E-mail: matthew.tam@uon.edu.au.

An equivalent formulation of (1.1) is the formally unconstrained nonsmooth optimization problem

$$\text{Find } (\overline{x}, \overline{y}, \overline{z}) \in \underset{(x,y,z)\in\mathbb{R}^p\times\mathbb{R}^q\times\mathbb{R}^r}{\arg\min} \left\{\Psi\left(x, y, z\right)\right\}. \tag{1.3}$$

Algorithms for solving (1.1) or (1.3) typically seek only to satisfy first-order necessary conditions for optimality, and the algorithm we propose below is no different. These conditions are given compactly by

$$0 \in \nabla F\left(x^*, y^*, z^*\right) + \partial\iota_X\left(x^*\right) + \partial\iota_Y\left(y^*\right) + \partial\iota_Z\left(z^*\right), \tag{1.4}$$

where $\partial f\left(z\right)$ is a set, the *subdifferential*, that generalizes the notion of a gradient for nonsmooth, subdifferentially regular functions f defined precisely in Definition 3.1 below.

For the sake of fixing the ideas, we focus on the particular application of *blind ptychography*, however our goal and approach are much more general. The partially smooth character of the objective Ψ in (1.2) is a common feature in many optimization models which involve sums of functions, some of which are smooth and some of which are not. Forward-backward-type algorithms are frequently applied to such models, and our approach is no different. The particular three-block structure of the problem is easily generalized to M blocks. The crucial feature of the model for algorithms, and what we hope to highlight in the present study, is the quantification of continuity of the partial gradients of F with respect only to blocks of variables. This is in contrast to more classical approaches which rely on the continuity of ∇F with respect to *all the variables* simultaneously (see [4]). For the ptychography application, such a requirement prohibits a convergence analysis along the lines of [4] since the gradient ∇F is *not* Lipschitz continuous. However, the partial gradients with respect to the blocks of variables *are* Lipschitz continuous. Following [11], this allows us to prove, in Section 3, convergence of the blocked algorithm given below (Algorithm 2.1) to feasible critical points. Issues concerning noisy data (i.e. what is the quality of the stationary points?) are independent of the performance of the algorithm presented here. We provide conditions under which the algorithm is guaranteed to find a critical point, no matter the starting guess or the quality of the data. How good that critical point is depends, of course, on the quality and quantity of the data.

Our abstract formulation of the problem can be applied to many different applications beyond ptychography, including control, machine learning, and deconvolution. We do not attempt to provide a review of the many different approaches to these types of problems, or even a more focused review of numerical methods for ptychography, but rather to provide a common theoretical framework by which a variety of methods can be understood. Our focus on ptychography allows us to also treat phase retrieval and wavefront sensing as special cases, so the scope of the concrete application is broader than immediately obvious. Also, we were inspired by the success of two algorithms, one by Maiden and Rodenburg [27] and the other due to Thibault and collaborators [36]. These two touchstone methods represent, for us, fundamental computational methods whose structure serves as a central bifurcation in numerical strategies. Moreover, the prevalence of these two methods, and their easy specialization to other well-known methods ensures that our theoretical framework will have the greatest practical impact. (Which is not to say that the methods are the most efficient [30, 35].) We present an algorithmic framework in Section 2 by which these algorithms can be understood and analyzed. We present in Section 3 a theory of convergence of the most general Algorithm 2.1 which is refined with increasingly stringent assumptions until it achieves the form of Algorithm 3.4 that can be immediately applied to ptychography. The specialization of our algorithm to ptychography is presented in Section 4 and summarized in Algorithm 4.1. We compare, in Section 5, Algorithm 4.1 with the state-of-the-art on simulated and experimental data.

2 Algorithms and Modeling

The solution we seek is a triple, $(\overline{x}, \overline{y}, \overline{z})$ that satisfies *a priori* constraints, denoted by C, as well as a model characterizing the coupling between the variables. We begin naively with a very intuitive idea for solving (1.1): alternating minimization (AM) with respect to the three separate blocks of variables x, y and z. More precisely, starting with any $(x^0, y^0, z^0) \in X \times Y \times Z$, we consider the following algorithm:

$$x^{k+1} \in \underset{x \in X}{\arg\min} \left\{ F\left(x, y^k, z^k\right) \right\}, \tag{2.1a}$$

$$y^{k+1} \in \underset{y \in Y}{\arg\min} \left\{ F\left(x^{k+1}, y, z^k\right) \right\}, \tag{2.1b}$$

$$z^{k+1} \in \underset{z \in Z}{\arg\min} \left\{ F\left(x^{k+1}, y^{k+1}, z\right) \right\}. \tag{2.1c}$$

While the simplicity of the above algorithm is attractive, there are several considerations one must address:

(i) The convergence results for the AM method are limited and applicable only in the convex setting. It is unknown if the AM method converges in the nonconvex setting. Of course, in the general nonconvex setting we can not expect for convergence to global optimum but even convergence to critical points is not known. In [3] the authors prove convergence to critical points for a regularized variant of alternating minimization. We follow this approach in Algorithm 2.1 below, applying proximal regularization in each of the steps to obtain provable convergence results.

(ii) Each one of the steps of the algorithm involves solving an optimization problem over just one of the blocks of variables. Forward-backward-type methods are common approaches to solving such minimization problems [13], and can be mixed between blocks of variables [11]. Forward operators are typically applied to the ill-posed or otherwise computationally difficult parts of the objective (usually appearing within the smooth part of the objective) while backward operators are applied to the well-posed parts of the objective (appearing often in the nonsmooth part).

In the particular case of ptychography, the subproblems with respect to the x and y variables (steps (2.1a) and (2.1b) of the algorithm) are ill-posed. We handle this by applying a regularized backward operator (the prox operator) to blockwise-linearizations of the ill-posed forward steps. The objective is well-posed, and particularly simple, with respect to the third block of variables, z, so we need only to employ a backward operator for this step. Generalizing, our approach addresses issue (ii) above by handling each of the blocks of variables differently.

Our presentation of Algorithm 2.1 makes use of the following notation. For any fixed $y \in \mathbb{R}^q$ and $z \in \mathbb{R}^r$, the function $x \mapsto F(x, y, z)$ is continuously differentiable and its partial gradient, $\nabla_x F(x, y, z)$, is Lipschitz continuous with moduli $L_x(y, z)$. The same assumption holds for the function $y \mapsto F(x, y, z)$ when $x \in \mathbb{R}^p$ and $z \in \mathbb{R}^r$ are fixed. In this case, the Lipschitz moduli is denoted by $L_y(x, z)$. Define $L'_x(y, z) \equiv \max\{L_x(y, z), \eta_x\}$ where η_x is an arbitrary positive number. Similarly define $L'_y(x, z) \equiv \max\{L_y(x, z), \eta_y\}$ where η_y is an arbitrary positive number. The role of η_x and η_y is to address the following issue: if the Lipschitz constants $L_x(y, z)$ and/or $L_y(x, z)$ are zero then we should replace them with positive numbers (for the sake of well-definedness of the algorithm). In practice, it is better to chose them to be small numbers but for the analysis it can be chosen arbitrarily.

Algorithm 2.1 (Proximal Block Implicit-Explicit Algorithm).
Initialization. Choose $\alpha, \beta > 1$, $\gamma > 0$ and $(x^0, y^0, z^0) \in X \times Y \times Z$.
General Step $(k = 0, 1, \ldots)$

1. Set $\alpha^k = \alpha L'_x\left(y^k, z^k\right)$ and select

$$x^{k+1} \in \operatorname*{arg\,min}_{x \in X} \left\{ \left\langle x - x^k, \nabla_x F\left(x^k, y^k, z^k\right) \right\rangle + \frac{\alpha^k}{2} \left\| x - x^k \right\|^2 \right\}, \tag{2.2}$$

2. Set $\beta^k = \beta L'_y\left(x^{k+1}, z^k\right)$ and select

$$y^{k+1} \in \operatorname*{arg\,min}_{y \in Y} \left\{ \left\langle y - y^k, \nabla_y F\left(x^{k+1}, y^k, z^k\right) \right\rangle + \frac{\beta^k}{2} \left\| y - y^k \right\|^2 \right\}, \tag{2.3}$$

3. Select

$$z^{k+1} \in \operatorname*{arg\,min}_{z \in Z} \left\{ F\left(x^{k+1}, y^{k+1}, z\right) + \frac{\gamma}{2} \left\| z - z^k \right\|^2 \right\}. \tag{2.4}$$

Algorithm 2.1 can be interpreted as a combination of the algorithms proposed in [3] and [6]. The regularization parameters α^k and β^k, $k \in \mathbb{N}$, are discussed in Section 5. For the moment, suffice it to say that these parameters are inversely proportional to the step size in Steps (2.2) and (2.3) of the algorithm (see Section 3). Noting that α_k and β_k, $k \in \mathbb{N}$, are directly proportional to the respective partial Lipschitz moduli, the larger the partial Lipschitz moduli the *smaller* the step size, and hence the slower the algorithm progresses.

This brings to light another advantage of blocking strategies that goes beyond convergence proofs: algorithms that exploit block structures inherent in the objective function achieve better numerical performance by taking heterogeneous step sizes optimized for the separate blocks. There is, however, a price to be paid in the blocking strategies that we explore here: namely, they result in procedures that pass *sequentially* between operations on the blocks, and as such are not immediately parallelizable. Here too, the ptychography application generously rewards us with added structure, as we show in Section 3.3, permitting parallel computations on highly segmented blocks.

The convergence theory developed in Section 3 is independent of the precise form of the coupling function F and independent of the precise form of the constraints. For our analysis we require that F is differentiable with ∇F Lipschitz continuous on bounded domains, and the partial gradient $\nabla_x F$ (globally) Lipschitz continuous as a mapping on X for each $(y, z) \in Y \times Z$ fixed, and partial gradient $\nabla_y F$ (globally) Lipschitz continuous as a mapping on Y for $(x, z) \in X \times Z$ fixed. The constraint sets X, Y and Z could be very general, we only assume that they are closed and disjoint. This is discussed more precisely below. The analysis presented in Section 3 guarantees only that Algorithm 2.1 converges to a point satisfying (1.4), which, it is worth reiterating, are not necessarily solutions to (1.1).

2.1 Blind Ptychography

In scanning ptychography, an unknown specimen is illuminated by a localized electromagnetic beam and the resulting wave is recorded on a CCD array somewhere downstream along the axis of propagation of the wave (*i.e.*, in the *far field* or the *near field* of the object). A ptychographic dataset consists of a series of such observations, each differing from the others by a spatial shift of either the object or the illumination. In the original ptychographic reconstruction procedure [22] it was assumed that the illuminating beam was known. What we call *blind* ptychography, in analogy with blind deconvolution, reflects the fact that the beam is not completely known, this corresponds to what is commonly understood by ptychography in modern applications [33, 34, 27, 36]. Here the problem is to simultaneously reconstruct the specimen and illuminating beam from a given ptychgraphic dataset. We will treat the case of scanning x-ray ptychography with far field measurements. This is not exhaustive of all the different settings one might encounter, but the mathematical structure of the problem, our principal interest, is qualitatively the same for all cases. For a review of ptychothographic methods, see [32] and the reference herein.

We formulate the ptychography problem on the product space $\mathbb{C}^n \times \mathbb{C}^n \times \mathbb{C}^{n\times m}$ where the first block $\mathbb{C}^n$ corresponds to the model space for the probe, the second block corresponds to the model space for the specimen, and the third block corresponds to the model space for the data/observations. The physical model space equipped with the real inner product is isomorphic to the Euclidean space $\left(\mathbb{R}^2\right)^n \times \left(\mathbb{R}^2\right)^n \times \left(\mathbb{R}^2\right)^{n\times m}$ with the inner product $\langle (x,y,z), (x',y',z')\rangle \equiv \sum_{j=1}^n \left\langle x_j, x'_j\right\rangle + \sum_{j=1}^n \left\langle y_j, y'_j\right\rangle + \sum_{j=1}^n \sum_{i=1}^m \left\langle z_{ij}, z'_{ij}\right\rangle$ for $x_i, y_i, z_{ij} \in \mathbb{R}^2$. This is in fact how complex numbers are represented on a computer, and hence the model space $\mathbb{C}^n$ with real inner product is just an efficient shorthand for $\left(\mathbb{R}^2\right)^n$ with the standard inner product for such product spaces. We will therefore retain the complex model space with real inner product when describing this problem, noting that all linear operators on this space have analogues on the space $\left(\mathbb{R}^2\right)^n$. The theory, however, will be set on real finite dimensional vector spaces.

Denote $\mathbf{z} \equiv (z_1, z_2, \ldots, z_m)$ with $z_j \in \mathbb{C}^n$ $(j = 1,2,\ldots,m)$. The objective function in our general optimization problem (1.1), $F : \mathbb{C}^n \times \mathbb{C}^n \times \mathbb{C}^{n\times m} \to \mathbb{R}_+$, is given by

$$F(x, y, \mathbf{z}) \equiv \sum_{j=1}^m \left\| S_j(x) \odot y - z_j \right\|^2 . \tag{2.5}$$

Here $S_j : \mathbb{C}^n \to \mathbb{C}^n$ denotes j-th shift operator which shifts the indexes in $x \in \mathbb{C}^n$ in some prescribed fashion and $\odot$ is the elementwise Haadamard product. This function measures in some sense the distance to the set

$$\mathcal{M} \equiv \left\{ (x, y, \mathbf{z}) \in \mathbb{C}^n \times \mathbb{C}^n \times \mathbb{C}^{n\times m} \mid S_j(x) \odot y = z_j,\ j = 1,2,\ldots,m \right\}. \tag{2.6}$$

Let $b_j \in \mathbb{R}^n$, $j = 1,2,\ldots,m$, denote the experimental observations, and $\mathcal{F}$ be a 2D discrete Fourier transform (of dimension $\sqrt{n} \times \sqrt{n}$) rearranged for vectors on $\mathbb{C}^n$. The constraints X, Y, and Z are separable and given by

$$\begin{aligned} X &\equiv \{\text{qualitative constraints on the probe}\}, && \text{(2.7a)}\\ Y &\equiv \{\text{qualitative constraints on the specimen}\}, && \text{(2.7b)}\\ Z &\equiv Z_1 \times Z_2 \times \cdots \times Z_m, & & \\ &\quad \text{where } Z_j \equiv \left\{ z \in \mathbb{C}^n \mid \left|(\mathcal{F}z)_l\right| = b_{j,l},\ (l = 1,2,\ldots,n) \right\} \quad (j = 1,2,\ldots,m). && \text{(2.7c)} \end{aligned}$$

The qualitative constraints characterized by X and Y are typically a mixture of support, support-nonnegativity or magnitude constraints corresponding respectively to whether the illumination and specimen are supported on a bounded set, whether these (most likely only the specimen) are "real objects" that somehow absorb or attenuate the probe energy, or whether these are "phase" objects with a prescribed intensity but varying phase. A support constraint for the set X, for instance, would be represented by

$$X \equiv \{x = (x_1, x_2, \dots, x_n) \in \mathbb{C}^n \mid |x_i| \le R \;\; (i = 1, 2, \dots, n) \text{ and, for } i \notin \mathbb{I}_X, \; x_i = 0\}, \tag{2.8}$$

where $\mathbb{I}_X$ is the index set corresponding to which pixels in the field of view the probe beam illuminates and R is some given amplitude. A mixture of support and amplitude constraints for the set Y would be represented by

$$Y \equiv \left\{y = (y_1, y_2, \dots, y_n) \in \mathbb{C}^n \mid 0 \le \underline{\eta} \le |y_i| \le \overline{\eta} \text{ and, for } i \notin \mathbb{I}_Y, y_i = 0\right\}, \tag{2.9}$$

where the index set $\mathbb{I}_Y$ is the analogous index set for the support of the specimen, and $\underline{\eta}/\overline{\eta}$ are lower/upper bounds on the intensity of the specimen. The set Z is nothing more than the *phase set* appearing in feasibility formulations of the phase retrieval problem [25].

Remark 2.1 (Feasibility versus minimization)**.** Since the algorithms discussed below involve, at some point, projections onto these sets, it is worthwhile noting here that, while, in most applications, the projections onto the sets X, Y and Z have a closed form and can be computed very accurately and efficiently, we are unaware of any method, analytic or otherwise, for computing the projection onto the set $\mathcal{M}$ defined by (2.6). For this reason, we have avoided formulation of the problem as a (nonconvex) *feasibility* problem

$$\text{Find } \overline{x} \in \mathcal{M} \cap (X \times Y \times Z)\,.$$

Nevertheless, this essentially two-set feasibility model suggests a wide range of techniques within the family of projection methods, alternating projections, averaged projections and Douglas–Rachford being representative members. In contrast to these, our approach is essentially a *forward-backward* method that avoids the difficulty of computing a projection onto the set $\mathcal{M}$ by instead minimizing a nonnegative coupling function F that takes the value 0 (only) on $\mathcal{M}$.

Remark 2.2 (Phase retrieval and wavefront sensing as special cases)**.** We point out various specializations of the ptychography problem that will be familiar to many practitioners.

1. Ptychography with a known illumination $\overline{x}$ and a single measurement set Z_1 is just the phase retrieval problem and Algorithm 2.1 reduces to a steepest descent algorithm for minimizing the distance between the object constraint set Y and the set of vectors satisfying the data, Z_1.

2. Ptychography with several unknown *on axis* illuminations and corresponding measurements (that is, the operator S_j in (2.5) is a shift in the direction of propagation of the field, not a lateral shift) and a known object $\overline{y}$ (usually a point source) is the wavefront reconstruction problem discussed in [25]. In this setting our algorithm reduces to steepest descent for minimizing the sum of squared distances between the illumination constraints and the sets of vectors satisfying the data.

3. Ptychography with several unknown *on axis* illuminations *and* an unknown object is what is known in astronomy as “phase diversity” [18, 25].

3 Algorithm Analysis

3.1 Mathematical Preliminaries

Algorithm 2.1 consists of three steps all of which reduce to the computation of a *projection* onto a given constraint set (convex and nonconvex). Recall that the *projection* onto a nonempty and closed subset Ω of a Euclidean space $\mathbb{R}^d$, is the (set-valued) mapping $P_\Omega : \mathbb{R}^d \rightrightarrows \Omega$ defined by

$$P_\Omega(v) \equiv \operatorname{arg\,min}\left\{\|u - v\| \;\mid u \in \Omega\right\}. \tag{3.1}$$

In Euclidean spaces the projection is single valued if and only if Ω is convex (in addition to being nonempty and closed). Specializing to the present application, for the constraints specified by (2.7), $P_C = (P_X, P_Y, P_Z)$ where P_X, P_Y and P_Z are, in general, multivalued (consider $P_C(0)$).

Since we are dealing with nonsmooth and nonconvex functions that can take the value $+\infty$, we require the following generalization of the derivative for nonconvex functions.

Definition 3.1 (Subdifferential [31])**.** Let $f : \mathbb{R}^d \to (-\infty, +\infty]$ be proper (not everywhere infinite) and lower semicontinuous (lsc).

- The *regular or Fréchet subdifferential* of f at $u \in \operatorname{dom} f$, denoted $\widehat{\partial} f(u)$, is the set of vectors $v \in \mathbb{R}^d$ which satisfy

$$\liminf_{\substack{w \neq u \\ w \to u}} \frac{f(w) - f(u) - \langle v, w - u\rangle}{\|w - u\|} \geq 0. \tag{3.2}$$

  If $u \notin \operatorname{dom} f$ then $\widehat{\partial} f(u) \equiv \emptyset$.

- The *limiting subdifferential* of f at $u \in \operatorname{dom} f$, denoted $\partial f(u)$, is the set of limits of limiting subdifferentials:

$$\partial f(u) \equiv \left\{ v \in \mathbb{R}^d \mid \exists u^k \to u \text{ with } f\left(u^k\right) \to f(u) \text{ and } v^k \to v \text{ with } v^k \in \widehat{\partial} f\left(u^k\right) \text{ as } k \to \infty \right\}.$$

We say that f is *subdifferentially regular* at u if $\partial f(u) = \widehat{\partial} f(u)$, and *subdifferentially regular* (without reference to the point u) if it is subdifferentially regular at every point in $\operatorname{dom} f$.

The notion of regularity of a set can be understood in terms of the subdifferential regularity of the indicator function of that set. We will call a set *Clarke regular* if the corresponding indicator function is subdifferentially regular (see [31, Definition 6.4]).

In [11], Bolte *et al.* present a general procedure for determining convergence to critical points of generic algorithms for nonsmooth and nonconvex problems. The procedure consists of verifying three criteria, two of which are quite standard and shared by most *descent* algorithms, see *e.g.*, [4]. The third criterion depends not on the algorithm but on the objective function: it must satisfy the *Kurdyka-Łojasiewicz (KL) property* (see [9, 10] and the references therein).

Definition 3.2 (Kurdyka-Łojasiewicz property)**.** Let $f : \mathbb{R}^d \to (-\infty, +\infty]$ be proper and lower semicontinuous. For $\eta \in (0, +\infty]$ define

$$\mathcal{C}_\eta \equiv \left\{ \varphi \in C\left[[0,\eta), \mathbb{R}_+\right] \text{ such that } \left\{ \begin{array}{ll} \varphi(0) = 0 & \\ \varphi \in C^1 & \text{on } (0,\eta) \\ \varphi'(s) > 0 & \text{for all } s \in (0,\eta) \end{array} \right\} \right\}. \tag{3.3}$$

The function f is said to have the Kurdyka-Łojasiewicz (KL) property at $\overline{u} \in \operatorname{dom} \partial f$ if there exist $\eta \in (0, +\infty]$, a neighborhood U of $\overline{u}$ and a function $\varphi \in \mathcal{C}_\eta$, such that, for all

$$u \in U \cap [f(\overline{u}) < f(u) < f(\overline{u}) + \eta],$$

the following inequality holds

$$\varphi\left(f\left(u\right) - f\left(\overline{u}\right)\right) \operatorname{dist}\left(0, \partial f\left(u\right)\right) \geq 1. \tag{3.4}$$

If f satisfies property (3.4) at each point of $\operatorname{dom} \partial f$, then f is called a *KL function*.

For a given function, the KL property can be verified indirectly by checking membership to certain classes of functions, in particular the class of *semi-algebraic* functions [9]. For the convenience of the reader, we recall here the definition of semi-algebraic functions.

Definition 3.3 (Semi-algebraic sets and functions)**.**

(i) A $S \subseteq \mathbb{R}^d$ is *(real) semi-algebraic* if there exists a finite number of real polynomial functions $p_{ij}, q_{ij} : \mathbb{R}^d \to \mathbb{R}$ such that

$$S = \bigcup_{j=1}^{N} \bigcap_{i=1}^{K} \left\{u \in \mathbb{R}^d : \; p_{ij}\left(u\right) = 0 \; \text{ and } \; q_{ij}\left(u\right) < 0\right\}.$$

(ii) A function $f : \mathbb{R}^d \to (-\infty, +\infty]$ is *semi- algebraic* if its graph

$$\left\{(u, t) \in \mathbb{R}^{d+1} \mid f\left(u\right) = t\right\},$$

is a semi-algebraic subset of $\mathbb{R}^{d+1}$.

The class of semi-algebraic sets is stable under the following operations: finite unions, finite intersections, complementation and Cartesian products. For a thorough catalog of semi-algebraic functions and sets see [2, 3, 4, 11] and the references therein.

While it may not be obvious how *directly* to verify the KL property it is easy to determine whether a function is semi-algebraic. The remarkable fact about semi-algebraic functions is that, as long as they are lower semi-continuous, they automatically satisfy the KL property on their domain as stated in the following result.

Theorem 3.1 ([9, Th. 3.3, pg. 1215])**.** *Let $f : \mathbb{R}^d \to (-\infty, +\infty]$ be a proper lower semicontinuous function. If f is semi-algebraic then it satisfies the KL property at any point in* $\operatorname{dom} f$.

We now show that the ptychography problem, and hence the phase retrieval problem, is semi-algebraic, *i.e.*, both the objective function F and the constraint set C are semi-algebraic.

Proposition 3.2 (Blind ptychography and phase retrieval are semi-algebraic)**.** *The objective function F, defined by* (2.5)*, is continuous and semi-algebraic. The constraint sets X, Y and Z, defined by* (2.7)*, are nonempty, closed and semi-algebraic. Consequently, the corresponding function Ψ, defined by* (1.2)*, is a KL function on $X \times Y \times Z$.*

Proof sketch. The physical model is formulated with complex-valued vectors, but $\mathbb{C}^n$ with the real inner product is isomorphic to the Euclidean space $(\mathbb{R}^2)^n$ with the inner product $\langle x, x' \rangle \equiv \sum_{i=1}^n (x_i, x_i')$ for $x_i, x_i' \in \mathbb{R}^2$. The function F defined by (2.5) is finite everywhere, continuous (indeed, differentiable), and the level sets of the objective F are quadratics with respect to y and z_j, and quadratic with respect to x under linear transformations. Thus F is semi-algebraic. The sets X and Y are either subspaces (support constraint only, (2.8)) or the intersection of a subspace with a box or ball (support and nonnegativity or support and amplitude constraints (2.9)), and so both of these are nonempty semi-algebraic. The set Z is equivalent to an amplitude constraint in the image space of the linear mapping $\mathcal{F}$ with respect to the 1-norm on each two-dimensional component of the product space $(\mathbb{R}^2)^n$. Thus Z is also nonempty semi-algebraic. That Ψ defined by (1.2) is then a KL-function for these F, X, Y, and Z then follows from Theorem 3.1. □

3.2 Convergence Analysis

Our convergence analysis is centered on Theorem 3.7, a general result concerning the application of Algorithm 2.1 to problem (1.4). The specialization to the ptychography problem, Proposition 4.1, is then easily achieved by verifying that the assumptions of a refinement, Theorem 3.10, are satisfied. Following [11], we carry out the three-step procedure, outlined in Section 3.1, for proving convergence of the sequence $\left\{\left(x^k, y^k, z^k\right)\right\}_{k \in \mathbb{N}}$, generated by Algorithm 2.1, to a point satisfying (1.4) provided that the initial point $(x^0, y^0, z^0) \in X \times Y \times Z$. The analysis rests on the following assumptions, collected here to avoid repetition.

Assumption 1. Let $\left\{\left(x^k, y^k, z^k\right)\right\}_{k \in \mathbb{N}}$ be iterates of Algorithm 2.1 for $(x^0, y^0, z^0) \in X \times Y \times Z$.

(i) $X \subset \mathbb{R}^p$, $Y \subset \mathbb{R}^q$, and $Z \subset \mathbb{R}^r$ are nonempty and closed.

(ii) $F : \mathbb{R}^p \times \mathbb{R}^q \times \mathbb{R}^r \to \mathbb{R}$ is differentiable and $\inf F > -\infty$. Moreover, $\nabla_x F$ and $\nabla_y F$ (as defined above) are Lipschitz continuous with moduli $L_x(y, z)$ and $L_y(x, z)$, respectively.

(iii) The gradient of F, ∇F, is Lipschitz continuous on bounded domains in $X \times Y \times Z$. Moreover, there exists $\lambda_x^+, \lambda_y^+ > 0$ such that

$$\sup\left\{L_x\left(y^k, z^k\right) \mid k \in \mathbb{N}\right\} \leq \lambda_x^+ \quad \text{and} \quad \sup\left\{L_y\left(x^{k+1}, z^k\right) \mid k \in \mathbb{N}\right\} \leq \lambda_y^+. \tag{3.5}$$

(iv) The iterates $\left\{\left(x^k, y^k, z^k\right)\right\}_{k \in \mathbb{N}}$ are bounded.

(v) The function Ψ defined by (1.2) is a KL function (see Definition 3.2).

Remark 3.1. In Section 4, we will show that our ptychography model (described in Section 2.1) satisfies these assumptions. For the general setting we point out the following.

(i) Boundedness of the iterates Assumption 1(iv) is a strong assumption that can be handled by a more technical treatment than we would like to present here. For our purposes this can be guaranteed by the physically natural assumption that the constraint set is bounded. Since Algorithm 2.1 is a *feasible point* algorithm, all iterates belong to the bounded feasible set, hence, in this case, the iterates are bounded.

(ii) Combining Assumptions 1(ii) and (iii) do not guarantee that the gradient ∇F is globally Lipschitz, as is the case in the application described below (see Section 4). The inequalities in (3.5) could be obtained in several scenarios, for example, when F is C^2 and using the boundedness assumption Assumption 1(iv).

We begin with a technical lemma.

Lemma 3.3 (Sufficient decrease property)**.** *Let $h : \mathbb{R}^d \to \mathbb{R}$ be a continuously differentiable function with gradient ∇h assumed to be L_h-Lipschitz continuous and let Ω be a nonempty and closed subset of $\mathbb{R}^d$. Fix any $t > L_h$. Then, for any $u \in \Omega$ and for $u^+ \in \mathbb{R}^d$ defined by*

$$u^+ \in P_\Omega\left(u - \frac{1}{t}\nabla h\left(u\right)\right),$$

we have

$$h\left(u^+\right) \leq h\left(u\right) - \frac{1}{2}\left(t - L_h\right)\left\|u^+ - u\right\|^2.$$

Proof. The result follows from [11, Lemma 2] where the nonsmooth function σ is the indicator function ι_Ω of the nonempty closed set Ω. □

Remark 3.2. When Ω is also convex, the conclusion of Lemma 3.3 can be improved (see [6, Lemma 2.3]) to the following

$$h\left(u^+\right) \leq h\left(u\right) - \left(t - \frac{L_h}{2}\right)\left\|u^+ - u\right\|^2.$$

This means that $t > L_h/2$ (rather than only $t > L_h$, as in the nonconvex case) is enough to guarantee decrease of function value after projected-gradient step.

Using Lemma 3.3 we can prove the following basic property of Algorithm 2.1.

Proposition 3.4 (Sufficient decrease)**.** *Let $\left\{\left(x^k, y^k, z^k\right)\right\}_{k\in\mathbb{N}}$ be a sequence generated by Algorithm 2.1 for some initial point $\left(x^0, y^0, z^0\right) \in X \times Y \times Z$. Suppose that conditions (i)-(ii) of Assumption 1 hold. Then the sequence $\left\{F\left(x^k, y^k, z^k\right)\right\}_{k\in\mathbb{N}}$ is decreasing and*

$$\sum_{k=1}^{\infty}\left\|\left(x^{k+1}, y^{k+1}, z^{k+1}\right) - \left(x^k, y^k, z^k\right)\right\|^2 < \infty.$$

Hence the sequence $\left\{F\left(x^k, y^k, z^k\right)\right\}_{k\in\mathbb{N}}$ converges to some $F^* > -\infty$ as $k \to \infty$.

Proof. We apply Lemma 3.3 to the first subproblem (see (2.2)) as follows. Take $h\left(\cdot\right) = F\left(\cdot, y^k, z^k\right)$, $\Omega = X$ and $t = \alpha^k > L_x'\left(y^k, z^k\right)$ to obtain that

$$\begin{aligned}
F\left(x^{k+1}, y^k, z^k\right) &\leq F\left(x^k, y^k, z^k\right) - \frac{1}{2}\left(\alpha^k - L_x\left(y^k, z^k\right)\right)\left\|x^{k+1} - x^k\right\|^2 \\
&\leq F\left(x^k, y^k, z^k\right) - \frac{1}{2}\left(\alpha^k - L_x'\left(y^k, z^k\right)\right)\left\|x^{k+1} - x^k\right\|^2 \\
&= F\left(x^k, y^k, z^k\right) - \frac{1}{2}\left(\alpha - 1\right)L_x'\left(y^k, z^k\right)\left\|x^{k+1} - x^k\right\|^2 \\
&\leq F\left(x^k, y^k, z^k\right) - \frac{1}{2}\left(\alpha - 1\right)\eta_x\left\|x^{k+1} - x^k\right\|^2,
\end{aligned}$$

where the second inequality follows from the facts that $\alpha^k = \alpha L'_x\left(y^k, z^k\right)$ (see Algorithm 2.1) and $L_x\left(y^k, z^k\right) \leq L'_x\left(y^k, z^k\right)$, the last inequality follows from the fact that $\eta_x \leq L'_x\left(y^k, z^k\right)$ and $\alpha > 1$. Similarly, applying Lemma 3.3 to the second subproblem (see (2.3)) with $h\left(\cdot\right) = F\left(x^{k+1}, \cdot, z^k\right)$, $\Omega = Y$ and $t = \beta^k > L'_y\left(x^{k+1}, z^k\right)$ yields

$$F\left(x^{k+1}, y^{k+1}, z^k\right) \leq F\left(x^{k+1}, y^k, z^k\right) - \frac{1}{2}\left(\beta - 1\right)\eta_y \left\|y^{k+1} - y^k\right\|^2.$$

On the other hand, immediately from the third updating rule (see (2.4)) we get that

$$F\left(x^{k+1}, y^{k+1}, z^{k+1}\right) \leq F\left(x^{k+1}, y^{k+1}, z^k\right) - \frac{\gamma}{2}\left\|z^{k+1} - z^k\right\|^2.$$

Summing up all these inequalities yields

$$\begin{aligned} F\left(x^{k+1}, y^{k+1}, z^{k+1}\right) \leq F\left(x^k, y^k, z^k\right) &- \frac{1}{2}\left(\alpha - 1\right)\eta_x \left\|x^{k+1} - x^k\right\|^2 - \frac{1}{2}\left(\beta - 1\right)\eta_y \left\|y^{k+1} - y^k\right\|^2 \\ &- \frac{\gamma}{2}\left\|z^{k+1} - z^k\right\|^2. \end{aligned}$$

Denote $\lambda^- \equiv (1/2)\min\left\{\left(\alpha - 1\right)\eta_x, \left(\beta - 1\right)\eta_y, \gamma\right\}$. Thus

$$F\left(x^{k+1}, y^{k+1}, z^{k+1}\right) \leq F\left(x^k, y^k, z^k\right) - \lambda^- \left\|\left(x^{k+1}, y^{k+1}, z^{k+1}\right) - \left(x^k, y^k, z^k\right)\right\|^2. \tag{3.6}$$

This proves that the sequence $\left\{F\left(x^k, y^k, z^k\right)\right\}_{k\in\mathbb{N}}$ is decreasing. Since, in addition, we know that F is bounded from below (see Assumption 1(ii)), we thus have a decreasing sequence on a compact interval and it follows that $\left\{F\left(x^k, y^k, z^k\right)\right\}_{k\in\mathbb{N}}$ converges to some $F^* > -\infty$. Summing up this inequality, for $k = 1, 2, \ldots, N$, yields

$$\begin{aligned} \sum_{k=1}^{N}\left\|\left(x^{k+1}, y^{k+1}, z^{k+1}\right) - \left(x^k, y^k, z^k\right)\right\|^2 &\leq \frac{1}{\lambda^-}\left(F\left(x^1, y^1, z^1\right) - F\left(x^{N+1}, y^{N+1}, z^{N+1}\right)\right) \\ &\leq \frac{F\left(x^1, y^1, z^1\right) - F^*}{\lambda^-}, \end{aligned} \tag{3.7}$$

where the last inequality holds true since $F\left(x^{N+1}, y^{N+1}, z^{N+1}\right) \geq F^*$. Taking the limit as $N \to \infty$ yields boundedness of the sum of step-lengths and completes the proof. □

Before proving the second step, we obtain the following immediate consequence.

Corollary 3.5 (Rate of asymptotic regularity)**.** *Let $\left\{\left(x^k, y^k, z^k\right)\right\}_{k\in\mathbb{N}}$ be a sequence generated by Algorithm 2.1 for some initial point $\left(x^0, y^0, z^0\right) \in X \times Y \times Z$ and define the corresponding sequence of steps $\left\{s^k\right\}_{k\in\mathbb{N}\setminus\{0\}}$ by $s^{k+1} \equiv \left(x^{k+1}, y^{k+1}, z^{k+1}\right) - \left(x^k, y^k, z^k\right)$. Suppose that conditions (i)-(ii) of Assumption 1 hold. Then $s^k \to 0$ as $k \to \infty$ with the following rate*

$$\min_{k=1,2,\ldots,N}\left\|s^{k+1}\right\| \leq \sqrt{\frac{F\left(x^1, y^1, z^1\right) - F^*}{N\lambda^-}},$$

where $\lambda^- \equiv (1/2)\min\left\{\left(\alpha - 1\right)\eta_x, \left(\beta - 1\right)\eta_y, \gamma\right\}$ and $F^* \equiv \lim_{k\to\infty} F\left(x^k, y^k, z^k\right)$.

Proof. From (3.7) we obtain that

$$N \min_{k=1,2,\ldots,N} \left\| s^{k+1} \right\|^2 \leq \sum_{k=1}^{N} \left\| s^{k+1} \right\|^2 \leq \frac{F\left(x^1, y^1, z^1\right) - F^*}{\lambda^-},$$

and thus

$$\min_{k=1,2,\ldots,N} \left\| s^{k+1} \right\|^2 \leq \frac{F\left(x^1, y^1, z^1\right) - F^*}{N\lambda^-}.$$

The result now easily follows. □

Proposition 3.6 (Lipschitz paths). *Let $\left\{ (x^k, y^k, z^k) \right\}_{k \in \mathbb{N}}$ be a sequence generated by Algorithm 2.1 for some initial point $(x^0, y^0, z^0) \in X \times Y \times Z$. Suppose that conditions (i)-(iv) of Assumption 1 hold. For each positive integer k, define the following three quantities: $A_z^k \equiv \gamma \left(z^{k-1} - z^k \right)$,*

$$A_x^k \equiv \alpha^{k-1} \left(x^{k-1} - x^k \right) + \nabla_x F\left(x^k, y^k, z^k\right) - \nabla_x F\left(x^{k-1}, y^{k-1}, z^{k-1}\right),$$

and

$$A_y^k \equiv \beta^{k-1} \left(y^{k-1} - y^k \right) + \nabla_y F\left(x^k, y^k, z^k\right) - \nabla_y F\left(x^k, y^{k-1}, z^{k-1}\right).$$

Then $A^k \equiv \left(A_x^k, A_y^k, A_z^k \right) \in \partial \Psi \left(x^k, y^k, z^k \right)$ and there exists $\delta > 0$ such that

$$\left\| A^k \right\| \leq \left(3\lambda^+ + 2\delta \right) \left\| \left(x^k, y^k, z^k \right) - \left(x^{k-1}, y^{k-1}, z^{k-1} \right) \right\|,$$

where

$$\lambda^+ \equiv \max \left\{ \alpha \lambda_x^+, \alpha \eta_x, \beta \lambda_y^+, \beta \eta_y, \gamma \right\} \tag{3.8}$$

for λ_x^+ and λ_y^+ defined by (3.5).

Proof. Let k be a positive integer. Writing the optimality condition of the first updating rule yields

$$\nabla_x F\left(x^{k-1}, y^{k-1}, z^{k-1}\right) + \alpha^{k-1} \left(x^k - x^{k-1} \right) + w_x^k = 0,$$

where $w_x^k \in \partial \iota_X \left(x^k \right)$. Hence

$$\nabla_x F\left(x^{k-1}, y^{k-1}, z^{k-1}\right) + w_x^k = \alpha^{k-1} \left(x^{k-1} - x^k \right).$$

It is clear from the definition of Ψ (see (1.2)), that

$$\partial_x \Psi \left(x^k, y^k, z^k \right) = \nabla_x F\left(x^k, y^k, z^k\right) + \partial \iota_X \left(x^k \right).$$

Combining these two facts proves that $A_x^k \in \partial_x \Psi \left(x^k, y^k, z^k \right)$. Following the same arguments applied on the second updating rule yields the desired result that $A_y^k \in \partial_y \Psi \left(x^k, y^k, z^k \right)$. Now, writing the optimality condition of the third updating rule yields

$$\nabla_z F\left(x^k, y^k, z^k\right) + \gamma \left(z^k - z^{k-1} \right) + w_z^k = 0,$$

for $w_z^k \in \partial \iota_Z \left(z^k \right)$, hence $A_z^k \in \partial_z \Psi \left(x^k, y^k, z^k \right)$.

We begin with an estimation of the norm of A_x^k. From Assumption 1(iii) and (iv), there exists $\delta > 0$ such that

$$\begin{aligned} \left\| \nabla_x F\left(x^k, y^k, z^k\right) - \nabla_x F\left(x^{k-1}, y^{k-1}, z^{k-1}\right) \right\| &\leq \left\| \nabla F\left(x^k, y^k, z^k\right) - \nabla F\left(x^{k-1}, y^{k-1}, z^{k-1}\right) \right\| \\ &\leq \delta \left\| \left(x^k, y^k, z^k \right) - \left(x^{k-1}, y^{k-1}, z^{k-1} \right) \right\|. \end{aligned}$$

Now, from the definition of λ^+, (3.8), we have, for all $k \in \mathbb{N}$,

$$\alpha^k \leq \alpha L'_x(y^k, z^k) \leq \alpha \max\{\lambda_x^+, \eta_x\} \leq \lambda^+.$$

This together with Assumption 1(iii) yields

$$\begin{aligned} \left\|A_x^k\right\| &\leq \alpha^{k-1} \left\|x^{k-1} - x^k\right\| + \left\|\nabla_x F\left(x^k, y^k, z^k\right) - \nabla_x F\left(x^{k-1}, y^{k-1}, z^{k-1}\right)\right\| \\ &\leq \lambda^+ \left\|x^{k-1} - x^k\right\| + \delta \left\|\left(x^k, y^k, z^k\right) - \left(x^{k-1}, y^{k-1}, z^{k-1}\right)\right\| \\ &\leq \left(\lambda^+ + \delta\right) \left\|\left(x^k, y^k, z^k\right) - \left(x^{k-1}, y^{k-1}, z^{k-1}\right)\right\|. \end{aligned}$$

By a similar argument, we have also

$$\left\|A_y^k\right\| \leq \left(\lambda^+ + \delta\right) \left\|\left(x^k, y^k, z^k\right) - \left(x^{k-1}, y^{k-1}, z^{k-1}\right)\right\|.$$

Thus

$$\begin{aligned} \left\|A^k\right\| \leq \left\|A_x^k\right\| + \left\|A_y^k\right\| + \left\|A_z^k\right\| &\leq \left(\lambda^+ + \lambda^+ + 2\delta\right) \left\|\left(x^k, y^k, z^k\right) - \left(x^{k-1}, y^{k-1}, z^{k-1}\right)\right\| + \gamma \left\|z^{k-1} - z^k\right\| \\ &\leq \left(3\lambda^+ + 2\delta\right) \left\|\left(x^k, y^k, z^k\right) - \left(x^{k-1}, y^{k-1}, z^{k-1}\right)\right\|. \end{aligned}$$

This proves the desired result. □

We are now ready to prove the main result of this section, namely convergence of Algorithm 2.1 to points satisfying (1.4) for any initial point $(x^0, y^0, z^0) \in X \times Y \times Z$. It is in deducing the last step of the general case that we use the assumption that Ψ satisfies the KL inequality (3.4).

Theorem 3.7 (Convergence to critical points)**.** *Let $\left\{\left(x^k, y^k, z^k\right)\right\}_{k \in \mathbb{N}}$ be a sequence generated by Algorithm 2.1 for some initial point $(x^0, y^0, z^0) \in X \times Y \times Z$. Suppose that Assumption 1 holds. Then following assertions hold.*

(a) The sequence $\left\{\left(x^k, y^k, z^k\right)\right\}_{k \in \mathbb{N}}$ has finite length, that is,

$$\sum_{k=1}^{\infty} \left\|\left(x^{k+1}, y^{k+1}, z^{k+1}\right) - \left(x^k, y^k, z^k\right)\right\| < \infty.$$

(b) The sequence $\left\{\left(x^k, y^k, z^k\right)\right\}_{k \in \mathbb{N}}$ converges to a point (x^*, y^*, z^*) satisfying (1.4).

Proof. The result follows from Propositions 3.4 and 3.6 together with [11, Theorem 1]. □

3.3 Acceleration of the PFB method

In this section we develop an accelerated version of Algorithm 2.1. To motivate our approach, we return to the naive alternating minimization method (2.1) with which we began. If each of the blocks were themselves separable, then we could recursively apply the blocking strategy discussed in Section 2 within the blocks. We first detail recursive blocking, which improves the step sizes, and then we discuss additional structures that enable efficient implementations via parallelization.

For simplicity, we focus our discussion on the first block X, the same strategy also can (and will) be applied to the block Y. Suppose the block X can be further subdivided into a product of smaller blocks: $X = X_1 \times X_2 \times \cdots \times X_P$ with $P \leq p$. For fixed $y \in \mathbb{R}^q$ and $z \in \mathbb{R}^r$ we consider the problem (2.1a),

$$\min_{x \in X_1 \times X_2 \times \cdots \times X_P} \left\{ F\left(x, y, z\right) \right\}. \tag{3.9}$$

This problem has the same difficulties with respect to the sub-blocks X_i, $i = 1, 2, \ldots, P$, and the other variables Y and Z as the original problem (1.3) has between the blocks X, Y and Z. We therefore use the same forward-backward strategy to solve the problem on the block, that is, we partially linearize F with respect to the *sub-blocks* of X (as opposed to a partial linearization with respect to the whole block) and compute the corresponding proximal operator.

More precisely, for $\xi \in X_i$ define $\zeta_i^k\left(\xi\right) \equiv \left(x_1^{k+1}, x_2^{k+1}, \ldots, x_{i-1}^{k+1}, \xi, x_{i+1}^k, \ldots, x_P^k\right) \in X_1 \times X_2 \times \cdots \times X_P$ and $u_i^k \equiv \left(x_1^{k+1}, x_2^{k+1}, \ldots, x_{i-1}^{k+1}, x_{i+1}^k, \ldots, x_P^k\right)$. Let $L_{x_i}\left(u_i^k, y^k, z^k\right)$ denote the modulus of Lipschitz continuity of the gradient of the mapping $x_i \mapsto F\left(\zeta_i^k\left(x_i\right), y^k, z^k\right)$. For some $\eta_{x_i} > 0$ ($i = 1, 2, \ldots, P$) fixed, define $L'_{x_i}\left(u_i^k, y^k, z^k\right) \equiv \max\left\{L_{x_i}\left(u_i^k, y^k, z^k\right), \eta_{x_i}\right\}$. From the iterate $x^k = \left(x_1^k, x_2^k, \ldots, x_P^k\right)$ we compute $x^{k+1} = \left(x_1^{k+1}, x_2^{k+1}, \ldots, x_P^{k+1}\right)$ by the following procedure.

Subroutine 3.1 (Successive sub-block x updating rule)**.** Define $x_0^{k+1} \equiv x_1^k$. Given $x_1^{k+1}, x_2^{k+1}, \ldots, x_{i-1}^{k+1}$ ($i = 1, 2, \ldots, P$) compute x_i^{k+1} by

$$x_i^{k+1} \in \operatorname*{arg\,min}_{x_i \in X_i} \left\{ \left\langle \left(x_i - x_i^k\right), \nabla_{x_i} F\left(\zeta_i^k\left(x_i^k\right), y^k, z^k\right)\right\rangle + \frac{\alpha_i^k}{2}\left\|x_i - x_i^k\right\|^2 \right\},$$

where $\alpha_i^k \equiv \alpha_i L'_{x_i}\left(u_i^k, y^k, z^k\right)$ for some fixed $\alpha_i > 1$.

Comparing this to (2.2), we note that the update for x^{k+1} computed by Subroutine 3.1 is computed with different stepsize in *each sub-block*, where the stepsize α_i^k depends, again, on the modulus of Lipschitz continuity of the gradient of the function defined on that sub-block. In contrast, the stepsize without recursive blocking, that is the stepsize α^k computed according to (2.2), depends on the modulus of Lipschitz continuity of the gradient of the function defined on the entire block, which is, by definition, larger than the constant associated with each sub-block. Consequently, the steps in Algorithm (2.1) without subblocking will be *smaller* than the steps computed via Subroutine 3.1.

Now, repeating this argument for the Y-block of variables yields an analogous sequential updating rule for this block. For $\mu \in Y_j$ and $Q \leq q$, define $\phi_j^k\left(\mu\right) \equiv \left(y_1^{k+1}, y_2^{k+1}, \ldots, y_{j-1}^{k+1}, \mu, y_{j+1}^k, \ldots, y_Q^k\right) \in Y_1 \times Y_2 \times \cdots \times Y_M = Y$ and $v_j^k \equiv \left(y_1^{k+1}, y_2^{k+1}, \ldots, y_{j-1}^{k+1}, y_{j+1}^k, \ldots, y_Q^k\right)$. Let $L_{y_j}\left(x^{k+1}, v_j^k, z^k\right)$ be the modulus of Lipschitz continuity of the gradient of the function $y_j \mapsto F\left(x^{k+1}, \phi_j^k\left(y_j\right), z^k\right)$. For some $\eta_{y_j} > 0$ ($j = 1, 2, \ldots, Q$) fixed, define $L'_{y_j}\left(x^{k+1}, v_j^k, z^k\right) \equiv \max\left\{L_{y_j}\left(x^{k+1}, v_j^k, z^k\right), \eta_{y_j}\right\}$. From the iterate $y^k = \left(y_1^k, y_2^k, \ldots, y_Q^k\right)$ we compute $y^{k+1} = \left(y_1^{k+1}, y_2^{k+1}, \ldots, y_Q^{k+1}\right)$ by the following procedure.

Subroutine 3.2 (Successive sub-block y updating rule)**.** Define $y_0^{k+1} \equiv y_1^k$. Given $y_1^{k+1}, y_2^{k+1}, \ldots, y_{j-1}^{k+1}$ $(j = 1, 2, \ldots, Q)$ compute y_j^{k+1} by

$$y_j^{k+1} \in \operatorname*{arg\,min}_{y_j \in Y_j} \left\{ \left\langle \left(y_j - y_j^k\right),\ \nabla_{y_j} F\left(x^{k+1}, \phi_j^k\left(y_i^k\right), z^k\right) \right\rangle + \frac{\beta_j^k}{2} \left\|y_j - y_j^k\right\|^2 \right\},$$

where $\beta_j^k \equiv \beta_j L'_{y_j}\left(x^{k+1}, v_j^k, z^k\right)$ for some fixed $\beta_j > 1$.

To generalize Algorithm 2.1 to the above recursive splitting, one simply replaces (2.2) and (2.3) with Subroutines 3.1 and 3.2, respectively.

Algorithm 3.3 (Proximal Heterogeneous Block Implicit-Explicit Algorithm).
Initialization. Choose $\alpha_i > 1$ $(i = 1, 2, \ldots, P)$, $\beta_j > 1$ $(j = 1, 2, \ldots, Q)$, $\gamma > 0$ and $(x^0, y^0, z^0) \in X \times Y \times Z$.
General Step $(k = 0, 1, \ldots)$

1. Update x^{k+1} according to Subroutine 3.1.

2. Update y^{k+1} according to Subroutine 3.2.

3. Select
$$z^{k+1} \in \operatorname*{arg\,min}_{z \in Z} \left\{ F\left(x^{k+1}, y^{k+1}, z\right) + \frac{\gamma}{2} \left\|z - z^k\right\|^2 \right\}.$$

The assumptions for proof of convergence of this algorithm in the generalized setting take the following form.

Assumption 2. Let $\left\{\left(x^k, y^k, z^k\right)\right\}_{k\in\mathbb{N}}$ be iterates generated by Algorithm 3.3 with

$$\left(x^0, y^0, z^0\right) \in X \times Y \times Z = (X_1 \times X_2 \times \cdots \times X_P) \times (Y_1 \times Y_2 \times \cdots \times Y_Q) \times Z.$$

(i) $X_i \subset \mathbb{R}^{p_i}$, $Y_j \subset \mathbb{R}^{q_j}$, and $Z \subset \mathbb{R}^r$ are nonempty and closed ($0 < p_i, q_j, r \in \mathbb{N}$ with $\sum_{i=1}^{P} p_i = p$ and $\sum_{j=1}^{Q} q_i = q$).

(ii) $F : \mathbb{R}^p \times \mathbb{R}^q \times \mathbb{R}^r \to \mathbb{R}$ is differentiable on $X \times Y \times Z$ and $\inf F > -\infty$. Moreover, $\nabla_{x_i} F$ ($i = 1, 2, \ldots, P$) and $\nabla_{y_j} F$ ($j = 1, 2, \ldots, Q$) are Lipschitz continuous with moduli $L_{x_i}(u_i, y, z)$ and $L_{y_j}(x, v_j, z)$, respectively. Here $u_i \in \mathbb{R}^{p-p_i}$ and $v_j \in \mathbb{R}^{q-q_j}$.

(iii) The gradient of F, ∇F, is Lipschitz continuous on bounded domains in $X \times Y \times Z$. Moreover, there exists $\lambda^+_{x_i}, \lambda^+_{y_j} > 0$ ($i = 1, 2, \ldots, P$) ($j = 1, 2, \ldots, Q$) such that

$$\sup\left\{L_{x_i}\left(u_i^k, y^k, z^k\right) \mid k \in \mathbb{N}\right\} \le \lambda^+_{x_i} \quad \text{and} \quad \sup\left\{L_{y_j}\left(x^{k+1}, v_j^k, z^k\right) \mid k \in \mathbb{N}\right\} \le \lambda^+_{y_j}.$$

(iv) The iterates $\left\{\left(x^k, y^k, z^k\right)\right\}_{k\in\mathbb{N}}$ are bounded.

(v) The function Ψ defined by (1.2) is a KL function (see Definition 3.2).

We now state the generalized convergence result analogous to Theorem 3.7.

Theorem 3.8 (Convergence to critical points - recursive)**.** *Let $\left\{\left(x^k, y^k, z^k\right)\right\}_{k\in\mathbb{N}}$ be a sequence generated by Algorithm 3.3 with*

$$\left(x^0, y^0, z^0\right) \in X \times Y \times Z = (X_1 \times X_2 \times \cdots \times X_P) \times (Y_1 \times Y_2 \times \cdots \times Y_Q) \times Z.$$

Suppose that Assumption 2 holds. Then following assertions hold.

(a) The sequence $\left\{\left(x^k, y^k, z^k\right)\right\}_{k\in\mathbb{N}}$ has finite length, that is,

$$\sum_{k=1}^{\infty} \left\|\left(x^{k+1}, y^{k+1}, z^{k+1}\right) - \left(x^k, y^k, z^k\right)\right\| < \infty.$$

(b) The sequence $\left\{\left(x^k, y^k, z^k\right)\right\}_{k\in\mathbb{N}}$ converges to a point (x^*, y^*, z^*) satisfying (1.4).

Proof sketch. The proof of convergence of the multi-block method follows by induction from the proof of the three-block case detailed in Section 3. □

As mentioned in Section 2, the trade-off for the larger step sizes used in recursive blocking is an $(P + Q + 1)$-step sequential algorithm instead of the original 3-step algorithm. In the next section we explore additional structures that permit parallelization.

3.4 Parallelization

We show here that the sequential Algorithm 3.3 can be parallelized within the blocks x and y under the following assumption:

Assumption 3. (i) For $y \in Y$ and $z \in Z$ fixed, the function $x \mapsto \nabla_x F(x, y, z)$ is separable in x in the following sense:

$$\nabla_x F(x, y, z) = \left(g_1(x_1, y, z), g_2(x_2, y, z), \ldots, g_P(x_P, y, z)\right), \tag{3.10}$$

where $g_i(\cdot, y, z) : X_i \to X_i$ for $i = 1, 2, \ldots, P$.

(ii) For $x \in X$ and $z \in Z$ fixed, the function $y \mapsto \nabla_y F(x, y, z)$ is separable in y in the following sense:

$$\nabla_y F(x, y, z) = \left(h_1(x, y_1, z), h_2(x, y_2, z), \ldots, h_Q(x, y_Q, z)\right), \tag{3.11}$$

where $h_j(x, \cdot, z) : Y_j \to Y_j$ for $j = 1, 2, \ldots, Q$.

An immediate consequence of the above assumption is the following.

Proposition 3.9 (Parallelizable separability)**.** *Suppose $F : X \times Y \times Z \to \mathbb{R}$ satisfies Assumption 3. Let $\left\{\left(x^k, y^k, z^k\right)\right\}_{k \in \mathbb{N}}$ be a sequence generated by Algorithm 3.3. Then*

$$\nabla_{x_i} F\left(\zeta_i^k\left(x_i^k\right), y^k, z^k\right) = \nabla_{x_i} F\left(x^k, y^k, z^k\right),$$

and

$$\nabla_{y_j} F\left(x^{k+1}, \phi_j^k\left(y_j^k\right), z^k\right) = \nabla_{y_j} F\left(x^{k+1}, y^k, z^k\right).$$

Consequently, the modulus of Lipschitz continuity of the gradient of the mapping $x_i \mapsto F\left(\zeta_i^k(x_i), y^k, z^k\right)$, $L_{x_i}\left(u_i^k, y^k, z^k\right)$ is dependent only on y^k and z^k, thus one can write $L_{x_i}\left(y^k, z^k\right)$ and $L'_{x_i}\left(y^k, z^k\right)$ for the corresponding Lipschitz constants. The same holds for the partial gradients with respect to y_j, where one can write $L_{y_j}\left(x^{k+1}, z^k\right)$ and $L'_{y_j}\left(x^{k+1}, z^k\right)$ for the corresponding Lipschitz constants.

An important consequence of Proposition 3.9 is that the successive steps of the respective Subroutines 3.1 and 3.2 can be computed in *parallel*. We summarize the results of this section with the following fully decomposable and parallelizable algorithm.

Algorithm 3.4 (**Proximal Parallel Heterogeneous Block Implicit-Explicit Algorithm).**
Initialization. Choose $\alpha_i > 1$ $(i = 1, 2, \dots, P)$, $\beta_j > 1$ $(j = 1, 2, \dots, Q)$, $\gamma > 0$ and $(x^0, y^0, z^0) \in X \times Y \times Z$.
General Step $(k = 0, 1, \dots)$

1. For each $i = 1, 2, \dots, P$, set $\alpha_i^k = \alpha_i L'_{x_i}\left(y^k, z^k\right)$ and select

$$x_i^{k+1} \in \underset{x_i \in X_i}{\arg\min} \left\{ \left\langle \left(x_i - x_i^k\right),\ \nabla_{x_i} F\left(x^k, y^k, z^k\right) \right\rangle + \frac{\alpha_i^k}{2} \left\| x_i - x_i^k \right\|^2 \right\}.$$

2. For each $j = 1, 2, \dots, Q$, set $\beta_j^k = \beta_j L'_{y_j}\left(x^k, z^k\right)$ and select

$$y_j^{k+1} \in \underset{y_j \in Y_j}{\arg\min} \left\{ \left\langle \left(y_j - y_j^k\right),\ \nabla_{y_j} F\left(x^{k+1}, y^k, z^k\right) \right\rangle + \frac{\beta_j^k}{2} \left\| y_j - y_j^k \right\|^2 \right\},$$

3. Select

$$z^{k+1} \in \underset{z \in Z}{\arg\min} \left\{ F\left(x^{k+1}, y^{k+1}, z\right) + \frac{\gamma}{2} \left\| z - z^k \right\|^2 \right\}.$$

We now state the generalized convergence result for the parallel algorithm, analogous to Theorem 3.8.

Theorem 3.10 (Convergence to critical points - parallel recursive)**.** *Let* $\left\{\left(x^k, y^k, z^k\right)\right\}_{k \in \mathbb{N}}$ *be a sequence generated by Algorithm 3.4 with*

$$\left(x^0, y^0, z^0\right) \in X \times Y \times Z = \left(X_1 \times X_2 \times \cdots \times X_P\right) \times \left(Y_1 \times Y_2 \times \cdots \times Y_Q\right) \times Z.$$

Suppose that Assumptions 2 and 3 hold. Then following assertions hold.

(a) The sequence $\left\{\left(x^k, y^k, z^k\right)\right\}_{k \in \mathbb{N}}$ *has finite length, that is,*

$$\sum_{k=1}^{\infty} \left\| \left(x^{k+1}, y^{k+1}, z^{k+1}\right) - \left(x^k, y^k, z^k\right) \right\| < \infty.$$

(b) The sequence $\left\{\left(x^k, y^k, z^k\right)\right\}_{k \in \mathbb{N}}$ *converges to a point* (x^*, y^*, z^*) *satisfying* (1.4).

Proof sketch. The proof of convergence of the parallel multi-block method follows by induction from the proof of the three-block case detailed in Section 3 and Proposition 3.9. □

4 Implementation for Blind Ptychography

We apply the above results to the ptychography problem described in Section 2.1 where the objective function F is given by (2.5) and the constraint set C by (2.7), (2.8) and (2.9). The sets $X, Y \subset \mathbb{C}^n$ decompose into the product of n complex planes. More precisely, $X \to X_1 \times X_2 \times \cdots \times X_n \subset (\mathbb{C})^n$ with

$$X_i \equiv \begin{cases} \{x \in \mathbb{C} \mid\ |x| \le R\}, & \text{for } i \in \mathbb{I}_X, \\ \{0\}, & \text{otherwise,} \end{cases} \qquad i = 1, 2, \dots, n, \tag{4.1}$$

where, again, $\mathbb{I}_X$ is the index set corresponding to the support of the probe beam and R is some given amplitude. Similarly, $Y \to Y_1 \times Y_2 \times \cdots \times Y_n \subset (\mathbb{C})^n$ with

$$Y_i \equiv \begin{cases} \left\{ y \in \mathbb{C} \mid 0 \leq \underline{\eta} \leq |y| \leq \overline{\eta} \right\}, & \text{for } i \in \mathbb{I}_Y, \\ \{0\}, & \text{otherwise,} \end{cases} \quad i = 1, 2, \ldots, n, \tag{4.2}$$

where the index set $\mathbb{I}_Y$ is the index set for the support of the specimen, and $\underline{\eta}/\overline{\eta}$ are given lower/upper bounds on the intensity of the specimen. We begin by showing that in this setting Assumptions 2 and 3 hold. In the context of the more general theory, in Assumption 2 for this application we have $P = Q = n$ and p_i , $q_i = 2$ for $i = 1, 2, \ldots, n$, where n is the number of pixels, and $r = 2mn$ where m is the number of images.

Proposition 4.1. *Let F be defined by (2.5) and let the constraint sets X, Y and Z be defined by (2.7). Then F, together with the constraints X, Y and Z satisfies Assumption 3 and the iterates of Algorithm 3.4 satisfy Assumption 2. Hence Algorithm 3.4 applied to the ptychography problem converges to a critical point from any feasible starting point.*

Proof. There several items from Assumption 2 that are trivial:

(i) the constraints X, Y, and Z are clearly nonempty and closed;

(ii) the objective function F is differentiable;

(iv) the generated sequence is bounded since this is a feasible point algorithm and the constraint set C is bounded (see Remark 3.1(i));

(iii) the Lipschitz continuity of ∇F on bounded subsets of $X \times Y \times Z$ follows immediately from the fact that F is C^2 and the fact that the generated sequence is bounded;

(v) by Proposition 3.2 the function Ψ (see (1.2)) is a KL function.

The only remaining parts needing verification are Lipschitz continuity of the partial gradients in Assumption 2(ii) and separability of the gradients in Assumption 3. The technical details of this calculation are left for an appendix where we show that (in a slight abuse of notation)

$$\nabla_x F(x, y, z) = \left(\nabla_{x_1} F(x_1, y, z), \nabla_{x_2} F(x_2, y, z), \ldots, \nabla_{x_n} F(x_n, y, z) \right), \tag{4.3}$$

$$\nabla_y F(x, y, z) = \left(\nabla_{y_1} F(x, y_1, z), \nabla_{y_2} F(x, y_2, z), \ldots, \nabla_{y_n} F(x, y_n, z) \right), \tag{4.4}$$

with respective moduli of continuity

$$L_{x_i}(y, z) = 2 \left(\sum_{j=1}^{m} S_j^* \left(\overline{y} \odot y \right) \right)_i, \qquad i = 1, 2, \ldots, n, \tag{4.5}$$

$$L_{y_i}(x, z) = 2 \left(\sum_{j=1}^{m} S_j \left(\overline{x} \odot x \right) \right)_i, \qquad i = 1, 2, \ldots, n. \tag{4.6}$$

Convergence of Algorithm 3.4 applied to critical points of the ptychography problem for any feasible initial guess then follows immediately from Theorem 3.10. □

We note that the partial gradients $\nabla_{x_i} F(x, y, z)$ (respectively $\nabla_{y_i} F(x, y, z)$) are with respect to the real and imaginary parts of $x_i \in \mathbb{C}$ (respectively $y_i \in \mathbb{C}$), or equivalently with respect to the two-dimensional real vectors $x_i \in \mathbb{R}^2$ (respectively $y_i \in \mathbb{R}^2$). So $\nabla_{x_i} F(x, y, z)$ (respectively $\nabla_{y_i} F(x, y, z)$) are actually mappings to vectors in $\mathbb{R}^2$ with moduli of Lipschitz continuity $L_{x_i}(y, z)$ (respectively $L_{y_i}(x, z)$).

The regularization parameters can be determined explicitly by the modulus of Lipschitz continuity of the gradient of F with respect to the isolated blocks of variables x and y, respectively. More precisely, for $i = 1, 2, \ldots, n$, we have

$$\begin{aligned} \alpha_i^k &= \alpha L_{x_i}\left(y^k, z^k\right) = \alpha \left(\sum_{j=1}^m S_j^* \left(\overline{y^k} \odot y^k \right) \right)_i , \qquad (4.7) \\ \beta_i^k &= \beta L_{y_i}\left(x^{k+1}, z^k\right) = \beta \left(\sum_{j=1}^m S_j \left(\overline{x^{k+1}} \odot x^{k+1} \right) \right)_i , \qquad (4.8) \end{aligned}$$

where $\alpha, \beta > 1$ are arbitrary.

In drawing the connections to other algorithms in the literature it is helpful to recognize that Steps (1) and (2) of Algorithm (3.4) are easily computed projections. Indeed,

$$\begin{aligned} x_i^{k+1} &\in \operatorname*{arg\,min}_{x_i \in X_i} \left\{ \left\langle x_i - x_i^k, \nabla_{x_i} F\left(x_i^k, y^k, \mathbf{z}^k\right) \right\rangle + \frac{\alpha_i^k}{2} \left\| x_i - x_i^k \right\|^2 \right\} \\ &= \operatorname*{arg\,min}_{x_i \in X_i} \left\{ \left\| x_i - \left(x_i^k - \frac{2}{\alpha_i^k} \sum_{j=1}^m \left[\left(S_j^* \left(\overline{y^k} \odot y^k \right) \right)_i \odot x_i^k - \left(S_j^* \left(\overline{y^k} \odot z_j^k \right) \right)_i \right] \right) \right\|^2 \right\} \\ &= P_{X_i} \left(x_i^k - \frac{2}{\alpha_i^k} \sum_{j=1}^m \left[\left(S_j^* \left(\overline{y^k} \odot y^k \right) \right)_i \odot x_i^k - \left(S_j^* \left(\overline{y^k} \odot z_j^k \right) \right)_i \right] \right), \qquad (4.9) \end{aligned}$$

where P_{X_i} is the projection onto the constraint set X_i. Similarly

$$\begin{aligned} y_i^{k+1} &\in \operatorname*{arg\,min}_{y_i \in Y_i} \left\{ \left\langle y_i - y_i^k, \nabla_{y_i} F\left(x^{k+1}, y_i^k, \mathbf{z}^k\right) \right\rangle + \frac{\beta_i^k}{2} \left\| y_i - y_i^k \right\|^2 \right\} \\ &= \operatorname*{arg\,min}_{y_i \in Y_i} \left\{ \left\| y_i - \left(y_i^k - \frac{2}{\beta_i^k} \sum_{j=1}^m \left[\left(S_j \left(\overline{x^{k+1}} \odot x^{k+1} \right) \right)_i \odot y_i^k - \left(S_j \left(\overline{x^{k+1}} \right) \odot z_j^k \right)_i \right] \right) \right\|^2 \right\} \\ &= P_{Y_i} \left(y_i^k - \frac{2}{\beta_i^k} \sum_{j=1}^m \left[\left(S_j \left(\overline{x^{k+1}} \odot x^{k+1} \right) \right)_i \odot y_i^k - \left(S_j \left(\overline{x^{k+1}} \right) \odot z_j^k \right)_i \right] \right), \qquad (4.10) \end{aligned}$$

where P_{Y_i} is the projection onto the constraint set Y_i. The last step is also a projection step given by

$$\begin{aligned} \mathbf{z}^{k+1} &\in \operatorname*{arg\,min}_{\mathbf{z} \in Z} \left\{ F\left(x^{k+1}, y^{k+1}, \mathbf{z}\right) + \frac{\gamma}{2} \left\| \mathbf{z} - \mathbf{z}^k \right\|^2 \right\} \\ &= \operatorname*{arg\,min}_{\mathbf{z} \in Z} \left\{ \sum_{j=1}^m \left\| \left(\frac{2}{2+\gamma} S_j\left(x^{k+1}\right) \odot y^{k+1} + \frac{\gamma}{2+\gamma} z_j^k \right) - z_j \right\|^2 \right\} \\ &= P_Z\left(\widetilde{\mathbf{z}}^{k+1}\right), \qquad (4.11) \end{aligned}$$

where $\widetilde{\mathbf{z}}^{k+1} \equiv \left(\widetilde{z_1}^{k+1}, \widetilde{z_2}^{k+1}, \ldots, \widetilde{z_m}^{k+1}\right)$ for

$$\widetilde{z_j}^{k+1} \equiv \frac{2}{2+\gamma} S_j\left(x^{k+1}\right) \odot y^{k+1} + \frac{\gamma}{2+\gamma} z_j^k, \quad j = 1, 2, \ldots, m. \tag{4.12}$$

Since Z is separable, the projection can be written written as

$$P_Z(\widetilde{\mathbf{z}}^{k+1}) = P_{Z_1}(\widetilde{z_1}^{k+1}) \times P_{Z_2}(\widetilde{z_2}^{k+1}) \times \cdots \times P_{Z_m}(\widetilde{z_m}^{k+1}), \tag{4.13}$$

so that $\mathbf{z}^{k+1} = \left(z_1^{k+1}, z_2^{k+1}, \ldots, z_m^{k+1}\right)$ where

$$z_j^{k+1} \in P_{Z_j}\left(\frac{2}{2+\gamma} S_j\left(x^{k+1}\right) \odot y^{k+1} + \frac{\gamma}{2+\gamma} z_j^k\right), \quad j = 1, 2, \ldots, m. \tag{4.14}$$

For a given point $z \in \mathbb{C}^n$ the projector onto the set Z_j (see (2.7c)) is given by [25]

$$P_{Z_j}(z) = \mathcal{F}^{-1}(\hat{z}) \qquad \text{where, for some } \theta \in (0, 2\pi], \quad \hat{z}_k = \begin{cases} b_{jk}\frac{[\mathcal{F}(z)]_k}{|[\mathcal{F}(z)]_k|}, & |[\mathcal{F}(z)]_k| \neq 0, \\ b_{jk}e^{i\theta}, & |[\mathcal{F}(z)]_k| = 0. \end{cases} \tag{4.15}$$

We summarize this discussion with the following specialization of Algorithm 3.4 to the blind ptychography problem.

Algorithm 4.1 (Ptychographic PHeBIE).
Initialization. Choose $\alpha_i > 1$ and $\beta_i > 1$ $(i = 1, 2, \ldots, n)$, $\gamma > 0$ and $(x^0, y^0, z^0) \in X \times Y \times Z$.
General Step $(k = 0, 1, \ldots)$

1. For each $i = 1, 2, \ldots, n$, set $\alpha_i^k = \alpha_i\left(\sum_{j=1}^m S_j^*\left(\overline{y^k} \odot y^k\right)\right)_i$ and select

$$x_i^{k+1} \in P_{X_i}\left(x_i^k - \frac{2}{\alpha_i^k}\sum_{j=1}^m\left[S_j^*\left(\overline{y^k} \odot y^k\right)_i \odot x_i^k - S_j^*\left(\overline{y^k} \odot z_j^k\right)_i\right]\right).$$

2. For each $i = 1, 2, \ldots, n$, set $\beta_i^k = \beta_i\left(\sum_{j=1}^m S_j\left(\overline{x^{k+1}} \odot x^{k+1}\right)\right)_i$ and select

$$y_i^{k+1} \in P_{Y_i}\left(y_i^k - \frac{2}{\beta_i^k}\sum_{j=1}^m\left[\left(S_j\left(\overline{x^{k+1}} \odot x^{k+1}\right)\right)_i \odot y_i^k - \left(S_j\left(\overline{x^{k+1}}\right) \odot z_j^k\right)_i\right]\right).$$

3. For each $j = 1, 2, \ldots, m$ select

$$z_j^{k+1} \in P_{Z_j}\left(\frac{2}{2+\gamma} S_j\left(x^{k+1}\right) \odot y^{k+1} + \frac{\gamma}{2+\gamma} z_j^k\right).$$

Convergence of Algorithm 4.1 to critical points has already been established in Proposition 4.1.

4.1 Relation to Current State-of-the-Art Algorithms

It is helpful to see Algorithm 4.1 in the context of two other blind ptychographic reconstruction algorithms, popular in the literature, namely, the methods of Thibault *et al.* [36], and Maiden and Rodenburg [27]. We show that these algorithms should not be expected to converge in general to a fixed point. However, the connection to Algorithm 4.1 and the attendant analysis immediately suggests how the methods of Thibault and Maiden and Rodenburg can be adjusted for greater stability or speed (or both). On the other hand, understanding these two methods in the context of the more general Algorithm 3.3 points the way to different constructions and compositions of the three basic steps of either Algorithm 3.3 or 3.4 for more efficient procedures. The analysis of such variants would then follow along the lines of the analytical methodology presented here.

4.1.1 Thibault *et al.* [36]

In order to explain the scheme suggested in [36] we first recall the definition of

$$Z \equiv Z_1 \times Z_2 \times \cdots \times Z_m \subset \mathbb{C}^{n\times m}. \tag{4.16}$$

Define the set D on the product space $\mathbb{C}^{n\times m}$:

$$D \equiv D_1 \times D_2 \times \cdots \times D_m \subset \mathbb{C}^{n\times m}, \tag{4.17}$$

where

$$D_j \equiv \left\{ z_j \mid z_j = S_j\left(x\right) \odot y \text{ for some } x, y \in \mathbb{C}^n \right\}. \tag{4.18}$$

If it were possible to compute the projection onto the set D (no closed form exists), then the Douglas–Rachford algorithm [14, 23, 5] could be applied to solve the feasibility problem:

$$\text{Find } \overline{x} \in D \cap Z.$$

More precisely, we have the following algorithm.

Algorithm 4.2 (Douglas–Rachford for ptychography).
Initialization. $(x^0, y^0, \mathbf{z}^0) \in X \times Y \times Z$.
General Step ($k = 0, 1, \ldots$)

1. Select an approximation $\mathbf{v^k}$ to some element from $P_D\mathbf{z}^k$.
2. Select
$$\hat{\mathbf{z}}^{k+1} \in P_Z\left(2\mathbf{v^k} - \mathbf{z^k}\right).$$
3. Set
$$\mathbf{z}^{k+1} = \mathbf{z}^k + \hat{\mathbf{z}}^{k+1} - \mathbf{v^k}. \tag{4.19}$$

As noted above, no closed form exists for the projection onto the set D. The method of [36] is an approximate Douglas–Rachford algorithm for set-feasibility with the following subroutine serving as an approximation to some element from the projector P_D. We describe the subroutine below as an approximation to the projector, however there has been no analysis to estimate exactly how good, or in what sense, it is an approximation, hence the qualifier *heuristic*.

Subroutine 4.3 (Heuristic approximation to P_D).
Input. $x^k \in \mathbb{C}^n, y^k \in \mathbb{C}^n, \mathbf{z}^k \in \mathbb{C}^{n\times m}, \Lambda \in \{1,2,3,\dots\}$
Initialization. Define $\hat{x}^0 \equiv x^k$, $\hat{y}^0 \equiv y^k$.
General Step ($l = 0, 1, \dots, \Lambda$).

1. Define $\alpha^l \in \mathbb{R}^m$ by

$$\alpha_i^l \equiv L_{x_i}(\hat{y}^l, z^k) = 2\left(\sum_{j=1}^m S_j^*\left(\overline{\hat{y}^l} \odot \hat{y}^l\right)\right)_i, \quad \text{for } i = 1,2,\dots,m,$$

and update $\hat{x}^{l+1}$ by

$$\hat{x}_i^{l+1} = \frac{2}{\alpha_i^l}\left(\sum_{j=1}^m S_j^*\left(\overline{\hat{y}^l} \odot z_j^k\right)\right)_i.$$

2. Define $\beta^l \in \mathbb{R}^m$ by

$$\beta_i^l \equiv L_{y_i}(\hat{x}^l, z^k) = 2\left(\sum_{j=1}^m S_j\left(\overline{\hat{x}^l} \odot \hat{x}^l\right)\right)_i, \quad \text{for } i = 1,2,\dots,m,$$

and update $\hat{y}^{k+1}$ by

$$\hat{y}_i^{l+1} = \frac{2}{\beta_i^l}\left(\sum_{j=1}^m S_j\left(\overline{\hat{x}^l}\right) \odot z_j^k\right)_i.$$

Final Step. Define $x^{k+1} \equiv \hat{x}^{\Lambda+1}$, $y^{k+1} \equiv \hat{y}^{\Lambda+1}$ and set

$$\widetilde{\mathbf{v}}^{k+1} \equiv \left(S_1\left(x^{k+1}\right) \odot y^{k+1}, \quad \cdots \quad, S_m\left(x^{k+1}\right) \odot y^{k+1}\right). \tag{4.20}$$

The method of [36] is Algorithm 4.2 with Step 1 replaced with the computation of $\widetilde{\mathbf{v}}^k$ via Subroutine 4.3. Subroutine 4.3, in turn, can be cast within our framework. Step 1 (respectively Step 2) of Algorithm (4.1) with $X_i = \mathbb{C}$ (respectively $Y_i = \mathbb{C}$) for each $i = 1,2,\dots,n$ is equivalent to Step 1 (respectively Step 2) of Subroutine 4.3.

Remark 4.1. Some further remarks on the method of Thibault *et al.* are in order.

(i) In an implementation of Thibault *et al.* , one monitors x^k and y^k (*i.e.*, the object and illumination function) rather than the iterate $\mathbf{z}^k$ itself. Since x^k and y^k are obtained during the computation of the so called *shadow iterates*, $P_D\mathbf{z}^k$, this can be interpreted as implicit monitoring of the *shadow sequence*, $\left(P_D\mathbf{z}^k\right)_{k=1}^{\infty}$.

(ii) The Douglas–Rachford methods is known to be sensitive to small perturbations in the constraint sets. In particular, if the intersection $D \cap Z$ is empty (not at all an improbable event with noisy, miss-specified data), then the Douglas–Rachford *cannot* converge [5]. The relaxation of the Douglas–Rachford algorithm studied in [26, 24] is one possibility for addressing this.

4.1.2 Maiden and Rodenburg [27]

In comparison to the other algorithms presented, the distinctive feature of the method of Maiden and Rodenburg [27] is that only a single magnitude measurement in used in each step. Their method can be described as follows.

Algorithm 4.4 (Maiden and Rodenburg).
Initialization. Choose $\alpha_i = \alpha \geq 2$ and $\beta_i = \beta \geq 2$ $(i = 1, 2, \ldots, n)$. Fix the mapping $\mathbb{I} : \mathbb{N} \mapsto \{1, 2, \ldots, m\}$ where the cardinality of the preimage of any $j \in \{1, 2, \ldots, m\}$ is infinite. Choose $(x^0, y^0, z^0) \in X \times Y \times Z_{\mathbb{I}(0)}$.
General Step $(k = 0, 1, \ldots)$

1. Set $\alpha^k = \alpha \left\| \sum_{j=1}^m S_j^* \left(\overline{y^k} \odot y^k \right) \right\|_\infty$ and, for each $i = 1, 2, \ldots, n$, select

$$x_i^{k+1} \in P_{X_i}\left(x_i^k - \frac{2}{\alpha^k} \sum_{j=1}^m \left[S_j^* \left(\overline{y^k} \odot y^k \right)_i \odot x_i^k - S_j^* \left(\overline{y^k} \odot z^k \right)_i \right] \right).$$

2. Set $\beta^k = \beta \left\| \sum_{j=1}^m S_j \left(\overline{x^k} \odot x^k \right) \right\|_\infty$ and, for each $i = 1, 2, \ldots, n$, select

$$y_i^{k+1} \in P_{Y_i}\left(y_i^k - \frac{2}{\beta^k} \sum_{j=1}^m \left[\left(S_j \left(\overline{x^k} \odot x^k \right) \right)_i \odot y_i^k - \left(S_j \left(\overline{x^k} \right) \odot z^k \right)_i \right] \right).$$

3. Select

$$z^{k+1} \in P_{Z_{\mathbb{I}(k+1)}} \left(S_j \left(x^{k+1} \right) \odot y^{k+1} \right).$$

Remark 4.2. In the context of Algorithm 4.1 several features of Algorithm 4.4 are worth noting.

(i) As established in Sections 3.2 and 4, the scalings α^k and β^k in Steps 1 and 2 of Algorithm 4.4 are Lipschitz constants of the partial gradient of F defined by (2.5) on the *entire* x and y blocks. This could be refined by using the scaling α_i^k given in Steps 1 and 2 of Algorithm 4.1.

(ii) Steps 1 and 2 of Algorithm 4.4 can be performed in *parallel* since the y update does not use information from the x update as in Algorithm 4.1.

(iii) As Algorithm 4.4 is essentially a cyclic projection algorithm, in practice one should expect the iterates to cycle.

5 Numerical Examples

To illustrate the differences between the various algorithms developed above, in Section 5.1 we compare algorithm performance on synthetic data where the problem "difficulty" is relatively well controlled (and the answer, shown in Figure 1, is known) and in Section 5.2 we compare algorithm performance on experimental data reported in [37]. In both the synthetic and experimental demonstrations we compare four different algorithms:

1. PHeBIE-I: Algorithm 3.3 specialized to ptychography with γ = 1e-30.
2. PHeBIE-II: Algorithm 4.1 (the ptychograohic specialization of Algorithm 3.4) with γ = 1e-30.
3. Thibault [36]: Algorithm 4.2 with Step 1 computed via Suboroutine 4.3 with $\Lambda = 3$.
4. Maiden and Rodenburg [27]: Algorithm 4.4.

5.1 Synthetic data

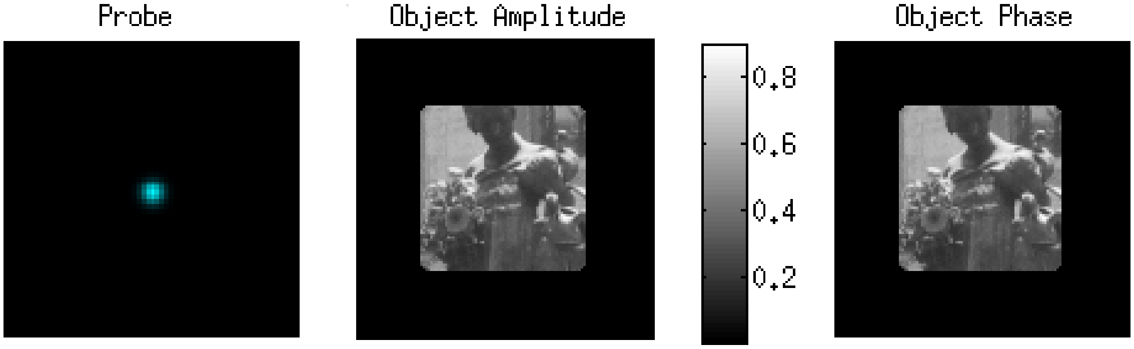


Figure 1: Simulated data (no noise), probe and object recovered by the PHeBIE II algorithm. All algorithms in the comparison with no noise yielded comparable recoveries.

Let x (respectively y) denote the true probe (true object). For the noiseless simulated data, we compute the measured data vectors using

$$b_j = |\mathcal{F}(S_j(x) \odot y)| \text{ for } j = 1, 2, \ldots, m.$$

For the simulated data with noise, we use Poisson noise with mean/variance $\lambda = 2$.

In typical ptychography experiments one more or less knows a priori what the probe looks like, though its precise structure, due to instrumentation aberrations, is unknown. The object, on the other hand, is assumed to be completely unknown except for certain qualitative properties, for example, that it is not absorbing. For the simulated data, the initial probe estimate consists of a circle of radius slightly larger than the true probe having constant amplitude and phase. Objects are initialized with a random initial guess. We demonstrate the stability of the algorithms in the results shown in Table 3 by purposely constraining the pupil to be smaller than the true pupil. This is not an unreasonable scenario since in practice the true pupil is not known.

Consistent with existing literature, we run several iterations of each algorithm without updating the probe to obtain a better initial object guess. The results of this “warm-up” procedure are then used as initial point $(x^0, y^0, \mathbf{z}^0)$ for the main algorithm of which 300 iterations were performed. Experimentally, the “warm-up” procedure could also be accomplished with an “empty” beam data set consisting of beam images taken without specimen.

Where convenient, we use u^k to denote $(x^k, y^k, \mathbf{z}^k)$. Five trials of each problem instance were performed with random object initializations. Tables 1, 2 and 3 report the average, and in brackets, the worst result for the following statistics.

1. The final value of the least-squares objective given by (2.5).

2. The square of the norm of the change between the final two iterations, *i.e.*, $\|u^{300} - u^{299}\|^2$.

3. The Root-mean-squared error of the final object and probe as described in [19]. The error is computed up to translation, a global phase shift and a global scaling factor.[1]

4. The *R-factor* at iteration 300, where

$$R\text{-factor}^k = \frac{\sum_{j=1}^m \|b_j - S_j(x^k) \odot y^k\|}{\sum_{j=1}^m b_j}. \tag{5.1}$$

   As in (2.7c), b_j denotes the experimental observations.

5. The total time (seconds) for the "warm-up" and main algorithm.

Remark 5.1 (Error metrics)**.** Theorem 3.8(a) guarantees that the difference between the iterates of Algorithm 2.1 and Algorithm 3.4 converge in norm to zero. To compute the RMS-error a knowledge of the true object and probe are required, which in real applications are not known. The R-factor can still be evaluated in experimental settings (see Figure 3) and used as a measure of quality of the reconstruction, though the theoretical behavior of this metric is not covered by our analysis.

Table 1: Average (worst) results for noiseless simulated data.

Algorithm	$F(u^{300})$	$\|u^{300} - u^{299}\|^2$	RMS-Object	RMS-Probe	R-factor^{300}	Time (s)
PHeBIE-I	99.63 (126.64)	0.5931 (0.9383)	0.0410 (0.0461)	0.0155 (0.0222)	0.0131 (0.0154)	913.75 (925.85)
PHeBIE-II	70.76 (77.17)	0.2210 (0.3522)	0.0423 (0.0471)	0.0081 (0.0154)	0.0101 (0.0108)	636.74 (652.17)
Rodenburg & Madien	948.13 (1499.11)	5.5164 (8.4885)	0.0542 (0.0590)	0.0952 (0.1714)	0.0350 (0.0419)	1178.21 (1198.72)
Thibault	4347.08 (4554.28)	28.8622 (34.4422)	0.0515 (0.0642)	0.0240 (0.0378)	0.0244 (0.0264)	875.94 (887.76)

Table 2: Average (worst) results for simulated data with Poisson noise.

Algorithm	$F(u^{300})$	$\|u^{300} - u^{299}\|^2$	RMS-Object	RMS-Probe	R-factor^{300}	Time (s)
PHeBIE-I	1.4415e+07 (6.9222e+07)	4.1504 (14.0823)	0.1928 (0.6840)	0.1896 (0.7084)	0.3499 (1.2698)	899.30 (933.54)
PHeBIE-II	1.4364e+07 (6.8972e+07)	521.9450 (2600.9689)	0.2807 (0.9940)	0.2537 (0.9746)	0.4001 (1.5135)	685.67 (714.21)
Rodenburg & Madien	6.7894e+04 (3.1414e+05)	14633.8000 (61868.3743)	0.2654 (0.9996)	0.3205 (0.9507)	0.3827 (1.2814)	1168.36 (1177.71)
Thibault	1.4520e+07 (6.9688e+07)	247.3130 (976.2039)	0.2476 (1.0000)	0.0700 (0.2498)	0.1748 (0.5686)	868.07 (892.19)

For the noiseless simulated dataset, the quality of the reconstructed object and probe from each of the methods examined are comparable. This applies to the quantitative error metrics recorded in Table 1, as well as to a visual comparison of reconstructions (not shown). In the absence of noise, all

[1] Computed using code written Mauel Guizar available online at http://www.mathworks.com/matlabcentral/fileexchange/18401-efficient-subpixel-image-registration-by-cross-correlation

Table 3: Average (worst) results for noiseless simulated data with over-restrictive pupil constraint.

Algorithm	$F(u^{300})$	$\|u^{300} - u^{299}\|^2$	RMS-Object	RMS-Probe	R-factor^{300}	Time (s)
PHeBIE-I	25653.20 (25656.12)	0.1474 (0.1682)	0.0443 (0.0501)	0.0492 (0.0494)	0.2936 (0.2937)	959.26 (1108.53)
PHeBIE-II	25653.80 (25660.13)	0.0622 (0.0852)	0.0314 (0.0356)	0.0496 (0.0499)	0.2936 (0.2937)	632.61 (645.10)
Rodenburg & Madien	3987.67 (4413.77)	6.2921 (16.3055)	0.0689 (0.0760)	0.0550 (0.0570)	0.2834 (0.2839)	1190.19 (1309.23)
Thibault	306602.00 (378554.38)	219.2110 (267.9303)	0.9432 (0.9437)	0.1898 (0.2435)	0.3634 (0.39160)	897.67 (962.45)

the methods examined worked well. It is worth noting, that Algorithm 4.1 was significantly faster than all the other methods.

With the addition of Poisson noise, the quality of the reconstructed objects and probes deteriorates. The error metrics are mixed, and no clear "winner" emerges from the values reported in Table 2. The method of Thibault could be expected to be more unstable since it is based on the Douglas–Rachford algorithm, it has the advantage of pushing past local minima that might otherwise trap Algorithm 3.4.

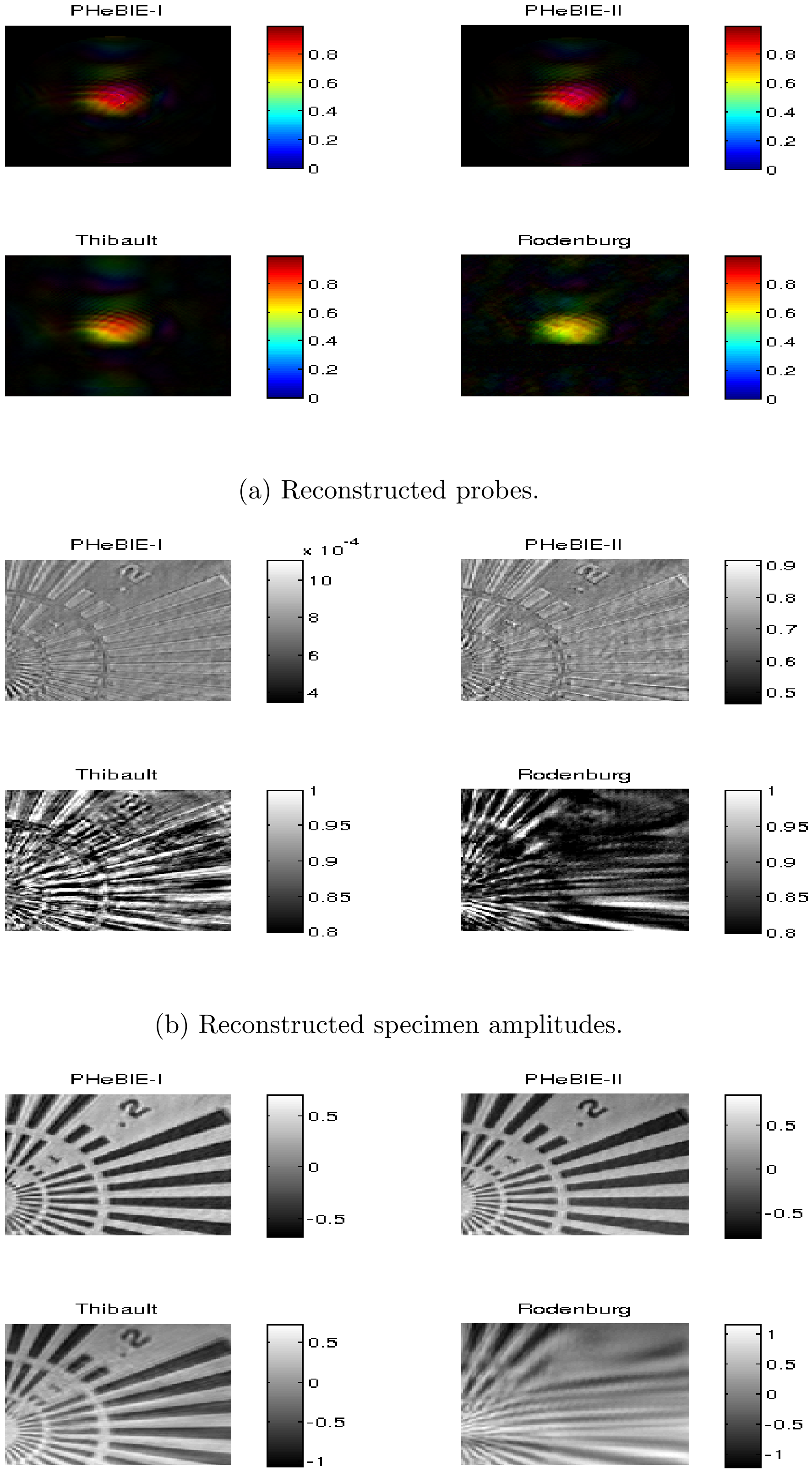
The results for a improperly specified pupil constraint (too small) demonstrate the relative stability of the respective methods. The method of Thibault *et al.* was the most sensitive to the over-restrictive pupil constraint and performed the worst. This is expected since the modeling errors lead to *inconsistency* of the underlying feasibility problem: it is well known that Douglas–Rachford does not have fixed points for inconsistent feasibility problems [5]. Visually, the method of Thibault was not able to recover any semblance of the true solution. Algorithms 3.3 and 3.4 are clearly more robust. Visual comparisons also bear this out.

Tables 1, 2 and 3 suggest that is it not appropriate to compare the final objective value and stepsize of the various algorithms directly. Significant variability was exhibited in these metrics between the methods, despite all recording having similar RMS and R-factor errors in the ideal case of noiseless data.

5.2 Experimental Data

In this section we examine the four algorithms applied to an experimental data set (from [37]) in which the actual illumination function and specimen are unknown. The reconstructed illumination functions and specimens obtained from the four algorithms are shown in Figure 2. By visual inspection, the reconstructions are of comparable quality, with the exception of the results from method of Madien and Rodenburg, which is of noticeably poorer quality.

In Figure 3 compare two error metrics as a function of number of iterations for the four algorithms. The first graph, Figure 3(a) shows the norm of the difference of successive iterates, which, for PHeBIE, is the only quantity guaranteed to converge to zero by the theory we have developed above. The second graph, Figure 3(b) shows the R-factor which, as discussed in Remark 5.1 is computable in experimental settings. The best performance, with respect to both of these metrics, were observed for the fully decomposed parallel PHeBIE-II (Algorithm 4.1). We do not make any direct comparison with the reconstructions in [37], however, because there the authors implement routines beyond the scope of our theory. A more complete benchmarking study on experimental data is forthcoming.

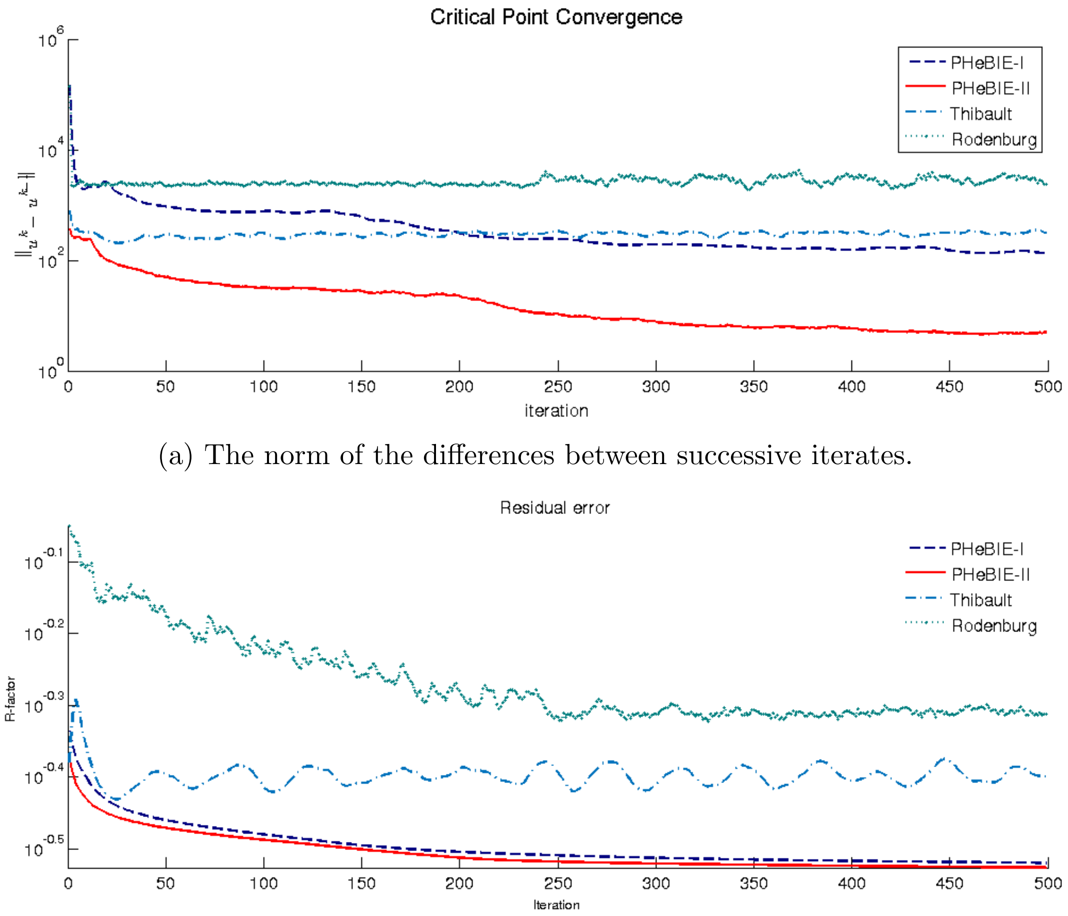


(a) Reconstructed probes.

(b) Reconstructed specimen amplitudes.

(c) Reconstructed specimen phases.

Figure 2: Results for the experimental dataset for the four different algorithms.



(a) The norm of the differences between successive iterates.

(b) The R-factor of the iterates defined by (5.1).

Figure 3: Performance profiles for the four algorithms applied to experimental data.

6 Conclusion

This study is an attempt to set the stage for more sophisticated variants of the basic template explored here. As noted in Remark 2.2, Algorithm 2.1 specializes to other well known algorithms in other contexts involving diffraction imaging. Our results, then yield, to the best of our knowledge, the most general proofs of global convergence of these algorithms *to critical points*. In [25, Theorem 6.1] global convergence of generic descent algorithms was proved under the assumption of global Lipschitz continuity of the coupling function F. Theorem 3.7 generalizes this result by requiring Lipschitz continuity only with respect to the blocks of variables and applying descent steps only with respect to the blocks. Also in [25, Section 4.2] it was pointed out that steepest descents without step-length optimization is equivalent to averaged projections, which is in turn equivalent to alternating projections on the product space [24], and thus extensions of the techniques here can be used to prove global convergence to critical points of projection algorithms like the Gerchberg-Saxton and error reduction algorithms [17, 15]. Again, it should be stressed that it is well known that these algorithms get stuck in bad local minima: the results say nothing about the quality of the critical points, only that critical points will be found. Moreover, the convergence of the algorithm is entirely independent of noise in the data, so long as the assumptions on the analytical properties of the coupling function are satisfied. There is some analysis showing that, if enough images are collected, local minima can be ruled out, and convergence to a global solution is guaranteed for certain algorithms [7, 8, 21]. A next step is to develop a similar analysis for Algorithm 2.1.

Regarding more sophisticated variations of our basic algorithmic template, an anonymous referee raised the possibility of considering a variable metric variant of Algorithm 2.1, pointing to [1, 12, 16], for example. It is also possible to rearrange the ordering of the blocks, and update some more frequently than others, change the size of the blocks, etc. Many of these ideas have in fact been employed within the framework of the basic algorithms proposed by [27, 36] (see for instance [20] and [37]). These issues have been studied in the context of ptychography in [28] and for multiresolution statistical image processing in [29]. A systematic study of all of these ideas would certainly be of interest.

Acknowledgments

RH and DRL were supported by DFG grant SFB755TPC2. SDS was supported by an Alexander von Humboldt Postdoctoral Fellowship. MKT was supported by an Australian Postgraduate Award. We would like to thank Robin Wilke and Tim Salditt of the Institute for X-ray Physics at the University of Göttingen for generously making their data available to us.

References

[1] F. Abboud, E. Chouzenoux, J.-C. Pesquet, J.-H. Chenot and L. Laborelli. A hybrid alternating proximal method for blind video restoration. *In Proceedings of the 22nd European Signal Processing Conference (EUSIPCO 2014)*, Lisbon, Portugal, 1-5 September 2014.

[2] H. Attouch and J. Bolte. On the convergence of the proximal algorithm for nonsmooth functions involving analytic features. *Math. Program.*, 116:5–16, 2009.

[3] H. Attouch, J. Bolte, P. Redont, and A. Soubeyran. Proximal alternating minimization and projection methods for nonconvex problems: an approach based on the Kurdyka-Łojasiewicz inequality. *Math. Oper. Res.*, 35:438–457, 2010.

[4] H. Attouch, J. Bolte, and B. F. Svaiter. Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward-backward splitting, and regularized Gauss-Seidel methods. *Math. Program.*, 137:91–129, 2013.

[5] H. H. Bauschke, P. L. Combettes, and D. R. Luke. Finding best approximation pairs relative to two closed convex sets in Hilbert spaces. *J. Approx. Theory*, 127:178–192, 2004.

[6] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.*, 2:183–202, 2009.

[7] T. Blumensath and M. Davies. Iterative hard thresholding for compressed sensing. *Applied and Computational Harmonic Analysis*, 27(3):265–274, 2009.

[8] T. Blumensath and M. Davies. Normalised iterative hard thresholding; guaranteed stability and performance. *IEEE Journal of Selected Topics in Signal Processing*, 4(2):298–309, 2010.

[9] J. Bolte, A. Daniilidis, and A. Lewis. The Łojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems. *SIAM J. Optim.*, 17:1205–1223, 2006.

[10] J. Bolte, A. Daniilidis, A. Lewis, and M. Shiota. Clarke subgradients of stratifiable functions. *SIAM J. on Optimization*, 18:556–572, 2007.

[11] J. Bolte, S. Sabach, and M. Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Math. Program., Ser. A,*146:459–494 (2014).

[12] E. Chouzenoux, J.-C. Pesquet, and A. Repetti. A block coordinate variable metric forward-backward algorithm. *Tech. Rep., 2014*.

[13] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Model. Simul.*, 4:1168–1200, 2005.

[14] J. Douglas and H. H. Rachford. On the numerical solution of heat conduction problems in two or three space variables. *Trans. Amer. Math. Soc.*, 82:421–439, 1956.

[15] J. R. Fienup, Phase retrieval algorithms: A comparison, *Appl. Opt.*, **21**:2758–2769 (1982).

[16] P. Frankel, G. Garrigos, and J. Peypouquet. Splitting Methods with Variable Metric for KurdykaŁojasiewicz Functions and General Convergence Rates. *J. Optim. Theory Appl.*, 2014. DOI 10.1007/s10957-014-0642-3.

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

[18] R. A. Gonsalves. Phase retrieval and diversity in adaptive optics. *Opt. Eng.*, 21 (1982), pp. 829–832.

[19] M. Guizar-Sicairos, S.T. Thurman, and J.R. Fienup. Efficient subpixel image registration algorithms. *Optics letters*, 33(2):156–158, 2008.

[20] J. Hagemann, A.-L. Robisch, D. R. Luke, C. Homann, T. Hohage, P. Cloetens, H. Suhonen and T. Salditt. Reconstruction of wave front and object for inline holography from a set of detection planes. *Optics Express*, 22:11552–11569, 2014.

[21] R. Hesse, D. R. Luke and P. Neumann Alternating Projections and Douglas-Rachford for Sparse Affine Feasibility. *IEEE Transactions on Signal Processing*, 62(18):4868–4881, 2014.

[22] R. Hegerl and W. Hoppe. Dynamische theorie der kristallstrukturanalyse durch elektronenbeugung im inhomogenen primrstrahlwellenfeld. *Ber. Bunsenges. Phys. Chem*, 74(11):1148–1154, 1970.

[23] P. L. Lions and B. Mercier. Splitting Algorithms for the Sum of Two Nonlinear Operators. *SIAM Journal on Numerical Analysis*, 16(6):964–979, 1979.

[24] D. R. Luke. Finding best approximation pairs relative to a convex and a prox-regular set in a Hilbert space. *SIAM J. Optim.*, 19:714–739, 2008.

[25] D. R. Luke, J. V. Burke, and R. G. Lyon. Optical wavefront reconstruction: theory and numerical methods. *SIAM Rev.*, 44:169–224, 2002.

[26] D. R. Luke. Relaxed averaged alternating reflections for diffraction imaging. *Inverse Problems*, 21(1):37, 2005.

[27] A. M. Maiden and J. M. Rodenburg. An improved ptychographical phase retrieval algorithm for diffractive imaging. *Ultramicroscopy*, 109:1256–1262, 2009.

[28] P. Mattsson. *Projection Methods in Ptychographic Imaging*, Masters Thesis, Georg-August-Universität Göttingen, 2013.

[29] L. Meyer *Blocking Strategies for Large-scale Split Feasibility*, Masters Thesis, Georg-August-Universität Göttingen, 2014.

[30] J. Qian, C. Yang, A. Schirotzek, F. Maia, and S. Marchesini. Efficient algorithms for ptychographic phase retrieval. *Contemporary Mathematics*, 2014.

[31] R. T. Rockafellar and R. J. Wets. *Variational Analysis.* Grundlehren der mathematischen Wissenschaften. Springer-Verlag, Berlin, 1998.

[32] J. M. Rodenburg. Ptychography and related diffractive imaging methods. *Adv. Imaging Electron Phys.*, 150:87–184, 2008.

[33] J. M. Rodenburg and R. H. T. Bates. The theory of super-resolution electron microscopy via wigner-distribution deconvolution. *Philos. Trans. R. Soc. London Ser. A*, 339:521–553, 1992.

[34] J. M. Rodenburg, A. C. Hurst, A. G. Cullis, B. R. Dobson, F. Pfeiffer, O. Bunk, C. David, K. Jefimovs, and I. Johnson. Hard-x-ray lensless imaging of extended objects. *Phys. Rev. Lett.*, 98:034801, 2007.

[35] Y. Shechtman, Y. C. Eldar, O. Cohen, H. N. Chapman, J. Miao and M. Segev. Phase retrieval with application to optical imaging. *IEEE Magazine*, 2014.

[36] P. Thibault, M. Dierolf, O. Bunk, A. Menzel, and F. Pfeiffer. Probe retrieval in ptychographic coherent diffractive imaging. *Ultramicroscopy*, 109:338–343, 2009.

[37] R. N. Wilke, M. Priebe, M. Bartels, K. Giewekemeyer, A. Diaz, P. Karvinen, and T. Salditt. Hard X-ray imaging of bacterial cells: nano-diffraction and ptychographic reconstruction. *Optics Express*, 20(17):19232–19254, 2012.

7 Appendix: Proof of Equations (4.3)-(4.6)

First, we compute the partial gradient of both functions.

$$\begin{aligned}\nabla_x F\left(x,y,\mathbf{z}\right) &= 2\sum_{j=1}^{m}\left[S_j\left(\cdot\right)\odot y\right]^*\left(S_j\left(x\right)\odot y - z_j\right) = 2\sum_{j=1}^{m}S_j^*\left(\overline{y}\odot\left(S_j\left(x\right)\odot y - z_j\right)\right)\\ &= 2\sum_{j=1}^{m}\left[S_j^*\left(\overline{y}\odot y\right)\odot x - S_j^*\left(\overline{y}\odot z_j\right)\right] \qquad (7.1)\end{aligned}$$

and

$$\begin{aligned}\nabla_y F\left(x,y,\mathbf{z}\right) &= 2\sum_{j=1}^{m}\left[S_j\left(x\right)\odot\left(\cdot\right)\right]^*\left(S_j\left(x\right)\odot y - z_j\right) = 2\sum_{j=1}^{m}\overline{S_j\left(x\right)}\odot\left(S_j\left(x\right)\odot y - z_j\right)\\ &= 2\sum_{j=1}^{m}\left[S_j\left(\overline{x}\odot x\right)\odot y - S_j\left(\overline{x}\right)\odot z_j\right], \qquad (7.2)\end{aligned}$$

where S_j^*, $j = 1, 2, \ldots, m$, denotes the adjoint transformation of S_j and $\overline{z}$ denote the element-wise complex conjugate of z. We remind the reader that $S_j^* = S_j^{-1}$. We also used the following two facts:

$$[S_j(\cdot) \odot y]^* = S_j^*(\overline{y} \odot (\cdot)) \quad \text{and} \quad [S_j(x) \odot (\cdot)]^* = \overline{S_j(x)} \odot (\cdot).$$

Using (7.1) we obtain, for any $x, x' \in \mathbb{C}^n$ that

$$\begin{aligned} \nabla_x F(x, y, \mathbf{z}) - \nabla_x F(x', y, \mathbf{z}) &= 2\sum_{j=1}^{m} \left[S_j^*(\overline{y} \odot y) \odot x - S_j^*(\overline{y} \odot y) \odot x'\right] \\ &= 2\sum_{j=1}^{m} S_j^*(\overline{y} \odot y) \odot (x - x') \\ &= 2\left(\sum_{j=1}^{m} S_j^*(\overline{y} \odot y)\right) \odot (x - x'), \end{aligned} \tag{7.3}$$

which means that

$$\|\nabla_x F(x, y, \mathbf{z}) - \nabla_x F(x', y, \mathbf{z})\| \leq 2\left\|\sum_{j=1}^{m} S_j^*(\overline{y} \odot y)\right\|_\infty \cdot \|x - x'\|,$$

the last inequality follows from the following fact

$$\|u \odot v\|^2 = \sum_{j=1}^{m}(u_j v_j)^2 \leq \sum_{j=1}^{m}(|u_{j*}| \cdot |v_j|)^2 = u_{j*}^2 \sum_{j=1}^{m} v_j^2 = \|u\|_\infty^2 \|v\|^2,$$

where $j*$ is the index of the largest entry in absolute value of u. This proves that

$$L_x(y, z) \leq 2\left\|\sum_{j=1}^{m} S_j^*(\overline{y} \odot y)\right\|_\infty.$$

On the other hand, choosing $x' = 0$ and $x = e_i$ (which is the i-th standard unit vector) and using (7.3) shows that

$$\nabla_x F(x, y, \mathbf{z}) - \nabla_x F(x', y, \mathbf{z}) = 2\left(\sum_{j=1}^{m} S_j^*(\overline{y} \odot y)\right) \odot e_i = 2\left(\sum_{j=1}^{m} S_j^*(\overline{y} \odot y)\right)_i,$$

where $(v)_i$ denotes the i-th component of the vector v. This means that we take $i = j*$, the largest entry in absolute value of $\sum_{j=1}^{m} S_j^*(\overline{y} \odot y)$, then we obtain that

$$\nabla_x F(x, y, \mathbf{z}) - \nabla_x F(x', y, \mathbf{z}) = 2\left\|\sum_{j=1}^{m} S_j^*(\overline{y} \odot y)\right\|_\infty.$$

This shows that

$$L_x(y, z) = 2\left\|\sum_{j=1}^{m} S_j^*(\overline{y} \odot y)\right\|_\infty.$$

Similar arguments shows that

$$L_y\left(x,z\right) = 2\left\|\sum_{j=1}^{m} S_j\left(\overline{x}\odot x\right)\right\|_{\infty}.$$

As a direct consequence of (7.3) we achieve

$$L_{x_i}\left(y,z\right) = 2\left[\sum_{j=1}^{m} S_j^*\left(\overline{y}\odot y\right)\right]_i$$

and by similar argument

$$L_{y_i}\left(x,z\right) = 2\left[\sum_{j=1}^{m} S_j\left(\overline{x}\odot x\right)\right]_i$$

which are (4.5) and (4.6) respectively. □

Remark 7.1 (Block partial Lipschitz constants)**.** For more general variable blocks of the form considered in Section 3.3, the corresponding formula for $L_{x_i}(y,z)$ (resp. $L_{y_i}(x,z)$) are given by taking twice largest entry in the block X_i (resp. Y_i) from the summation. That is,

$$L_{x_i}\left(y,z\right) = 2\left\|\left(\sum_{j=1}^{m} S_j^*\left(\overline{y}\odot y\right)\right)\Bigg|_{X_i}\right\|_{\infty}, \quad L_{y_i}\left(x,z\right) = 2\left\|\left(\sum_{j=1}^{m} S_j\left(\overline{x}\odot x\right)\right)\Bigg|_{Y_i}\right\|_{\infty},$$

where $|_{X_i}$ (res. $|_{Y_i}$) denotes the restriction to the block X_i (resp. Y_i).

From these formulae one immediately recovers (4.5) and (4.6) as special cases.