A WEAKLY-CONVEX FORMULATION FOR PHASELESS IMAGING

İ. Bayram⋆ E. Mason† B. Yazıcı†
⋆ Istanbul Technical University, Istanbul, Turkey
†Rensselaer Polytechnic Institute, Troy, NY , USA

ABSTRACT
We consider the problem of reconstructing an object given magnitudes of linear measurements. We follow the ‘lifting’ approach, but unlike previous work which use convex relaxations of the unit rank constraint, we use a weakly-convex matrix penalty. We derive a convergent algorithm and show that it is computationally more feasible than those obtained under convex relaxations. We demonstrate numerically that when the signal to noise ratio is high, the proposed algorithm can achieve almost error-free reconstruction with fewer measurements than when convex relaxation is employed.

Index Terms— Phaseless imaging, lifting, unit rank constraint, weakly-convex, majorization-minimization

1. INTRODUCTION
The phaseless imaging problem consists of estimating an object \( x \) given intensity measurements of the form \( y_i = |(a_i, x)|^2 \), for \( i = 1, 2, \ldots, m \) where \( a_i \) denotes the \( i \)-th row of a known imaging operator \( A \).

A direct approach to recover \( x \) is

\[
\hat{x} = \arg \min_y \| y - |A z|^2 \|_2, \tag{1}
\]

Unfortunately, this formulation is non-convex and algorithms attempting to solve (1) run the risk of getting trapped in local minima [1]. An equivalent convex problem can be obtained by ‘lifting’ the problem [2, 3, 4]. For this, observe that \( y = \text{diag}(A X A^T) \), where \( X = x x^T \) and \( \text{diag}(\cdot) \) extracts the diagonal of its operand. Notice that the compound operator \( L(X) = \text{diag}(A X A^T) \) is linear in \( X \). We can obtain \( x \) (upto a multiplicative factor) by estimating the unit rank positive semi-definite (psd) \( X \) via

\[
\hat{X} = \arg \min_{Z \in S_n} \| y - L(Z) \|_2^2 \quad \text{s.t.} \quad \begin{cases} \text{rank}(Z) = 1, \\ Z \succeq 0, \end{cases} \tag{2}
\]

where \( S_n \) denotes the set of \( n \times n \) symmetric matrices. An alternative formulation is

\[
\hat{X} = \arg \min_{Z \in S_n} \text{rank}(Z) \quad \text{s.t.} \quad \begin{cases} y = \text{diag}(A Z A^T), \\ Z \succeq 0. \end{cases} \tag{3}
\]

In both formulations (2), (3), the appearance of ‘\( \text{rank}(\cdot) \)’ makes the problem non-convex. Replacing it with a convex proxy like the nuclear norm, \( \| \cdot \|_* \), one obtains a convex problem that is guaranteed to reconstruct \( X \), under certain assumptions on \( A \) [3]. It was shown in [5] that it is also possible to drop the cost term entirely and still ensure exact recovery. Dropping the cost term leads to a feasibility problem as,

\[
\text{Find } Z \text{ such that } \begin{cases} y = \text{diag}(A Z A^T), \\ Z \succeq 0. \end{cases} \tag{4}
\]

In order to obtain a simple algorithm, this problem may also be written as,

\[
\min_{Z \succeq 0} \frac{1}{2} \| y - \text{diag}(A Z A^T) \|_2^2 \tag{5}
\]

For (5), the forward backward splitting algorithm [6] yields the following iterations.

1: \; \text{repeat} \\
2: \; Z \leftarrow Z - \alpha L^T (L(Z) - y) \quad \triangleright \text{take a gradient step} \\
3: \; Z \leftarrow P_\mathcal{S}(Z) \quad \triangleright \text{Project onto the cone of psd matrices} \\
4: \; \text{until some convergence criterion is met}

In the pseudocode, \( P_\mathcal{S}(\cdot) \) denotes the projection operator onto the set of positive semi definite (psd) matrices. Although this simple algorithm can correctly recover \( X \) given enough measurements, it occasionally fails to find a unit rank \( X \), when the number of measurements are fewer. Thus, even though the unit rank constraint complicates the problem significantly by introducing non-convexity, it can be useful for eliminating incorrect estimates. In fact, the unit rank constraint allows to eliminate the psd constraint. More precisely, consider the following problem

\[
\min_{Z \in S_n} \| y - L(Z) \|_2^2 \quad \text{s.t.} \quad \text{rank}(Z) = 1. \tag{6}
\]

For this problem, we have the following result.

**Proposition 1.** If \( \hat{X} \) is a local minimizer of (6), then \( \hat{X} \succeq 0 \).

**Proof.** Since \( X \) is unit rank, it can be expressed as \( \hat{X} = \beta \hat{x} \hat{x}^T \), for some vector \( \hat{x} \). Here \( \beta \in \mathbb{R} \) since \( \hat{X} \in S_n \). Consider the function \( g(\beta) = \| y - L(\beta \hat{x} \hat{x}^T) \|_2^2 \), which is a slice of the original cost, in the direction of \( \hat{X} \). Since \( y > 0 \), the function \( g(\beta) \) is differentiable.

It can be shown that the derivative is strictly negative when \( \beta < 0 \). Thus follows the claim.

\( \square \)

The following simple algorithm for (6) can be derived via the majorization-minimization framework [7].

1: \; \text{repeat} \\
2: \; Z \leftarrow Z - \alpha L^T (L(Z) - y) \quad \triangleright \text{take a gradient step} \\
3: \; Z \leftarrow P_\mathcal{S}(Z) \quad \triangleright \text{Find the closest unit rank matrix} \\
4: \; \text{until some convergence criterion is met}

Notice that the difference between this algorithm from the former is the replacement of \( P_\mathcal{S} \) with \( P_1 \), which maps \( Z \) to the unit rank matrix closest to \( Z \). This can be achieved by computing the largest magnitude eigenvalue and the corresponding eigenvector. Thus, in general, \( P_1 \) is much easier to realize than \( P_\mathcal{S} \) and therefore this latter algorithm is computationally more feasible.

We have observed that the algorithm introduced above for solving (6) performs quite well but it occasionally fails to find the correct solution, even if the number of measurements is high. We think this is due to non-convex nature of the formulation (6). In order to
circumvent this, we propose in this paper to relax the unit rank constraint using a penalty other than the nuclear norm and drop the psd constraint. This leads to a formulation as, 

$$\min_{Z \in S_n} \frac{1}{2} \| y - L(Z) \|_2^2 + q(Z),$$

(7)

where $q$ is a penalty function on $Z$, derived from our previous work [8]. The parameters of $q$ can be chosen such that $q(Z)$ is arbitrarily small for a unit rank $Z$ and increases rapidly if more than one singular value of $Z$ is non-zero. This property allows to significantly reduce a possible bias introduced by the addition of $q(\cdot)$, provided that a unit rank solution is found for the problem. The proposed $q$ is weakly-convex, that is, $q$ becomes convex after adding a quadratic function [9]. Thus, in contrast to earlier work, the non-convexity introducing rank constraint is not discarded [5] or relaxed to a nuclear norm [2, 3], but relaxed using a function that provides a better approximation of the constraint. The algorithm for solving (7) can be obtained by replacing $P$ by an approximation of the constraint. The algorithm for solving (7) can be derived from our previous work [9]. Thus, in contrast to earlier work, the non-convexity introduced in (7) is well-defined. This follows from the observation that $\alpha \lambda \gamma < 1$, then the cost function in (9) is strictly convex so that $J_{p\gamma}$ is well-defined. This follows from the observation that $\| x \|_0 \leq 1$, then the term enclosed in parentheses vanishes. The proximity operator [6] for this function is defined as, 

$$J_{p\gamma}(x) = \arg\min_{z} \frac{1}{2\alpha} \| x - z \|_2^2 + \lambda_{\gamma}(x).$$

(9)

If $\alpha \lambda \gamma < 1$, then the cost function in (9) is strictly convex so that $J_{p\gamma}$ is well-defined. This follows from the observation that $\| x \|_0 \leq 1$, then the term enclosed in parentheses vanishes. The proximity operator [6] for this function is defined as, 

$$J_{p\gamma}(x) = \arg\min_{z} \frac{1}{2\alpha} \| x - z \|_2^2 + \lambda_{\gamma}(x).$$

(9)

We remark that the description above is implicit since the number of non-zeros of $x$ (i.e., the integer $k$) is known in advance. However, a candidate $k$ can be validated by checking whether the threshold $\tau_k$ produces a $z$ with $k$ non-zero values [8]. Therefore, a linear search can be performed starting from $\hat{k} = 1$ to find the correct value of $k$. Here we note a special case of interest, which can also be found in [8].

**Proposition 2.** For $x, z \in \mathbb{R}^n$, suppose $z = J_{p\gamma}(x)$. Suppose also that $x_i$ denotes the entry of $x$ with the largest magnitude. If 

$$\alpha \lambda \gamma > \max_{i \neq l} \frac{|x_i| - \alpha \lambda}{|x_l| - \alpha \lambda}$$

(12)

then $z_i = \text{soft}(x_i, \alpha \lambda)$ and $z_i = 0$, for $i \neq l$.

In words, all but the largest (in magnitude) component of $x$ is kept if the ratio of the largest component to the rest is greater than some threshold. This property is the key to imposing the unit rank constraint, as will be clarified below.

**2.2. The Induced Penalty on $S_n$**

We extend the domain of the penalty function $p$ from $\mathbb{R}^n$ to $S_n$ as follows. Suppose the eigendecomposition of $X \in S_n$ is given as $X = V \Lambda V^T$. We define

$$q_{\lambda,\gamma}(X) = p_{\lambda,\gamma}(\text{diag}(\Lambda)).$$

(13)
The proximity operator for $q$ is defined as,
\[
J_{\alpha q}(X) = \arg \min_{Z \in S_n} \frac{1}{2\alpha} \|X - Z\|_F^2 + q_{\alpha, \gamma}(Z). \tag{14}
\]
It follows by the discussion in [14] (see specifically [15]) along with the properties of $p$ that $J_{\alpha q}$ is well-defined when $\alpha \lambda \gamma < 1$. This proximity operator is related to $J_{\alpha p}$ as follows [15].

**Proposition 3.** Suppose the eigendecomposition of $X \in S_n$ is $X = V \Lambda V^T$. Also, let $\Lambda'$ denote the diagonal matrix that satisfies diag($\Lambda'$) = $J_{\alpha p}$(diag(\Lambda)). Then, $J_{\alpha q}(X) = V \Lambda' V^T$.

Suppose $Z = J_{\alpha q}(X)$. If $Z$ has $k$ non-zero eigenvalues, then it follows by the discussion preceding Prop. 2 that we need to compute only the largest magnitude $k + 1$ eigenvalue/eigenvector pairs of $X$. For $k = 1$ (the unit rank $Z$ case), we have the following result, which is a corollary of Prop. 2 and Prop. 3.

**Corollary 1.** Suppose the eigendecomposition of $X \in S_n$ is $X = V \Lambda V^T$ and $\Lambda_i$ is the diagonal entry of $\Lambda$ with the largest magnitude. If
\[
\alpha \lambda \gamma > \max_{i \neq i'} \frac{(|\Lambda_i| - \alpha \lambda)}{|\Lambda_i - \alpha \lambda|},
\]
then, $J_{\alpha q}(X) = \text{soft}((\Lambda, \alpha \lambda) v_i v_i^T)$, where $v_i$ is the $i^{th}$ column of $V$.

This corollary suggests that if the gap between the largest eigenvalue and the rest of the eigenvalues of $X$, is large enough, $J_{\alpha q}(\cdot)$ reduces $X$ to a unit rank matrix. We next derive and study the convergence of a monotone descent algorithm for solving (7).

### 3. A MINIMIZATION ALGORITHM

Recall the formulation in (7).
\[
\min_{Z \in S_n} \{C(Z) = \frac{1}{2} \|y - L(Z)\|_2^2 + q_{\alpha, \gamma}(Z)\}. \tag{16}
\]
It can be shown that provided $L^T L > (\lambda \gamma) I$, this problem is convex. However, even if this is not satisfied, it is possible to derive a monotone descent algorithm by employing majorization-minimization techniques [16, 7].

#### 3.1. Derivation of the Algorithm

Suppose $Z^k$ denotes our $k^{th}$ iterate. Consider the following update.
\[
Z^{k+1} = \arg \min_{Z \in S_n} \left\{C_k(Z) = C(Z) + \frac{1}{2\alpha} \|Z - Z^k\|_F^2 - \frac{1}{2} \|L(Z - Z^k)\|_2^2 \right\}. \tag{17}
\]
Observe that $C_k(Z^k) = C(Z^k)$. Provided $\sigma(L^T L) < 1$, it follows that $C_k(Z)$ upper bounds $C$, that is, $C_k(Z) \leq C(Z)$ for all $Z \in S_n$. In this case, we will have $C(Z^{k+1}) \leq C(Z^k)$, where the inequality is strict if $Z^k \neq Z^{k+1}$.

To find an expression for $Z^{k+1}$, we expand $C_k$ and find that
\[
Z^{k+1} = \arg \min_{Z \in S_n} \frac{1}{2\alpha} \|Z - \tilde{Z}^k\|_F^2 + q_{\alpha, \gamma}(Z), \tag{18}
\]
where $\tilde{Z}^k = Z^k - \alpha L^T (L(Z^k) - y)$. Provided that $\alpha < 1/(\lambda \gamma)$, we have
\[
Z^{k+1} = J_{\alpha q}(\tilde{Z}^k). \tag{19}
\]

### Algorithm 1 A Descent Algorithm for (7)

1: $k \leftarrow 1$, initialize $Z^1$, set $\alpha$ (see Prop. 4)
2: repeat
3: $Z \leftarrow Z^k - \alpha L^T (L(Z^k) - y)$ \Comment{take a gradient step}
4: $Z^{k+1} \leftarrow J_{\alpha q}(Z)$ \Comment{Apply the prox. operator for $q$}
5: $k \leftarrow k + 1$
6: until some convergence criterion is met

A summary is provided in Algorithm 1.

The foregoing discussion, along with Thm. 3.1 of [17], leads to the following proposition.

**Proposition 4.** Suppose $\alpha < \min\{1/(\lambda \gamma), 1/\sigma(L^T L)\}$. Then, $Z^k$’s in Algorithm 1 monotonically decrease the cost, i.e., $C(Z^{k+1}) \leq C(Z^k)$ and the sequence of cost values $c_k = C(Z^k)$ converge to a value $C(Z^*)$, where $Z^*$ is a stationary point of $C(\cdot)$.

#### 3.2. Convergence of the Iterates

In practice, we are not only interested in the convergence of the sequence of cost values. It is also desired that the sequence of iterates themselves converge. This is addressed in the following.

**Proposition 5.** Suppose $\alpha < \min\{1/(\lambda \gamma), 1/\sigma(L^T L)\}$. Then, $Z^k$’s in Algorithm 1 converge to a local minimizer of (7).

Prop. 5 is a consequence of Thm. 5.1 of [18] which implies that the iterates of monotone descent algorithms converge to a local minimum when the cost function is semi-algebraic\(^1\). Sums of semi-algebraic functions are semi-algebraic [18] and a quadratic function (like the first term of $C(\cdot)$) is semi-algebraic since it is a polynomial of its arguments. Therefore, for Prop. 5, it is sufficient to show that $q$ is semi-algebraic, which is addressed in the following lemma.

**Lemma 1.** Viewing an $n \times n$ matrix $X$ as an element of $\mathbb{R}^{n^2}$, the nuclear norm $\|X\|_*$ and the regularizer $q$ introduced in (13) are semi-algebraic functions defined on $\mathbb{R}^{n^2}$.

**Proof.** We start by showing that the nuclear norm is semi-algebraic. Let $\mathcal{Q}$ denote the subset of $\mathbb{R}^{n^2}$ associated with orthogonal matrices (matrices formed by placing the components in a specific order). Notice that $\mathcal{Q}$ (or $\mathcal{Q} \times \mathcal{Q}$) is a compact semi-algebraic set, since it is the solution set of a finite number of polynomial equations. Observe also that for $Q_1 \in \mathcal{Q}$, we have $\text{tr}(Q_1^T X Q_2) \leq \|X\|_*$, where equality is satisfied if the SVD of $X$ is $X = Q_1 \Sigma Q_2^T$. Therefore,
\[
\|X\|_* = \max_{Q_1, Q_2 \in \mathcal{Q} \times \mathcal{Q}} \{h(X, Q_1, Q_2) = \text{tr}(Q_1 X Q_2)\}. \tag{20}
\]

But in this equation $h(X, Q_1, Q_2)$ is a polynomial of its arguments. Thus the function $\|X\|_*$ is semi-algebraic (see e.g. the discussion following Thm 2.2 in [18]). Notice now that $q(X) = \|X\|_* - \|X\|_2$. Since the sum and products of semi-algebraic functions are semi-algebraic, it follows that $q$ is semi-algebraic. $\square$

---

\(^1\)In a nutshell, a function is semi-algebraic if its graph can be expressed as a finite intersection of polynomial inequalities. Due to lack of space, we refer to [18] and the references therein for the precise definition and discussion of semi-algebraic sets and functions.
Fig. 1. The average recovery error for different signal lengths and number of measurements for (a) optimizationless reconstruction [5], (b) unit rank algorithm, (c) the proposed reconstruction algorithm. (d) shows slices of $E_x$ for all three algorithms for $n = 50$.

4. NUMERICAL EXPERIMENTS

In this section, we compare the proposed algorithm with the two algorithms provided in the introduction that solve (5) and (6), which we will refer to as ‘optimizationless’ and ‘unit rank’ in what follows. For the purpose of comparison, we performed experiments similar to those of [5]. We remark that the optimizationless algorithm has been used earlier in [10] as a benchmark algorithm (see, in particular, comparisons with PhaseLift [3]). We also note that, at least in the high SNR case, the trace term in the PhaseLift formulation introduces a certain bias and the reconstruction is different than those obtained by the algorithms considered in the experiments below, which are almost error-free, provided that the number of observations are sufficient (see Fig. 3 in [5]).

Experiment 1. We randomly produce the object $x$ by uniformly sampling the $n$-dimensional sphere, and the $m \times n$ observation matrix, $A$, by sampling a Gaussian distribution. We produce the observations as,

$$y = |Ax|^2 + u,$$

where $u$ denotes a Gaussian noise term. We can define two different recovery errors, related to $x$ and $X = xx^T$. If $X$ denotes the reconstruction obtained by an algorithm, the normalized error for recovering $X$ can be taken as $E_X = \|X - X\|_F/\|X\|_F$. Given the matrix estimate $X$, we take $\hat{x} = \sqrt{\lambda}e$, where $\lambda$ is the greatest eigenvalue of $X$ and $e$ is the corresponding unit norm eigenvector. The recovery error for $x$ (recall that $x$ is unit norm) is then defined as $E_x = \min_{h \rightarrow 0} \|(-1)^k \hat{x} - x\|_2$. We remark that [5] uses $E_X$ but we think that $E_x$ is of interest since $x$ is the object we aim to reconstruct. We have not seen a marked difference between the behaviors of the two distinct error terms. In the setup described above, for each $m$ and $n$ value, we average the recovery errors over twenty trials in order to obtain an average figure. We set the constants used in the algorithms as $\alpha = 10^{-4}$, $\lambda = 10^{-5}$, $\gamma = 0.05/\lambda$. Notice that, with these settings, we have $\alpha \lambda \gamma < 1$, as required in the proposed algorithm.

Figure 1 shows the phase transition diagrams as $n$ and $m$ vary, where the observation SNR is 50 dB. To produce these figures, we ran the proposed algorithms for 1000 iterations. We observe that the proposed algorithm and the unit rank algorithm both have sharper transitions between success and failure, compared to optimizationless reconstruction. Compared to unit rank, the proposed algorithm performs more stably and requires fewer measurements for successful recovery. In particular, we see in Fig.1d that unit rank has not improved much when the number of measurements, $m$, is in the range [100, 150] and has an unexpected local maximum at $m = 175$, whereas the proposed algorithm steadily improves as $m$ increases.

Experiment 2. In order to demonstrate the sharp transition noted above and test robustness against noise, we performed similar experiments for $n = 100$ with different SNR values. In this case, we set $\alpha = 10^{-4}/5$ (to ensure convergence) where the other parameters are kept as in the previous experiment. The reconstruction errors $E_x$ are shown in Fig. 2 for SNR values 50 dB and 5 dB. In this case, we ran the algorithms for 5000 iterations, to avoid premature stopping.

In both cases, the average error for optimizationless recovery appears to be smooth with respect to $m$ and especially towards the right end of the graphs, it exceeds the error obtained by the proposed algorithm. As for the unit rank algorithm and the proposed algorithm, we see a similar trend as in the previous experiment. The proposed algorithm achieves a lower error than unit rank for all $m$. We also observe that the proposed algorithm monotonically decreases the average error as $m$ increases (except when the number of measurements is too few), whereas the unit rank algorithm can produce incorrect estimates even when $m \geq 300$—e.g., when $m = 400$ in Fig. 2b.

5. CONCLUSION

The proposed algorithm falls somewhere in between algorithms obtained via convex relaxation and an algorithm that imposes the non-convex unit rank constraint at each iteration, both in terms of performance as well as computational load. Therefore, weak-convex relaxation allows to achieve a ‘tunable’ performance. Currently, due to lifting, the number of variables are significantly increased and hence memory requirements are high for real-life problems. In order to make the approach feasible for larger problems, we plan to complement the algorithm with a smart initialization strategy to achieve a reasonable approximation with a fewer number of iterations.
6. REFERENCES


