Scalable Convex Methods for Phase Retrieval

Alp Yurtsever, Ya-Ping Hsieh and Volkan Cevher
Laboratory for Information and Inference Systems (LIONS)
Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland
{alp.yurtsever, ya-ping.hsieh, volkan.cevher}@epfl.ch

***Abstract*—This paper describes scalable convex optimization methods for phase retrieval. The main characteristics of these methods are the cheap per-iteration complexity and the low-memory footprint. With a variant of the original PhaseLift formulation, we first illustrate how to leverage the scalable Frank-Wolfe (FW) method (also known as the conditional gradient algorithm), which requires a tuning parameter. We demonstrate that we can estimate the tuning parameter of the FW algorithm directly from the measurements, with rigorous theoretical guarantees. We then illustrate numerically that recent advances in universal primal-dual convex optimization methods offer significant scalability improvements over the FW method, by recovering full HD resolution color images from their quadratic measurements.**

I. Introduction

The problem of retrieving the phase of a signal from its intensity-only measurements has regained significant attention recently. Formally, the phase retrieval problem aims to recover a signal $\mathbf{x}^\natural \in \mathbb{C}^p$ from n noisy quadratic measurements

$$b_i = |\langle \mathbf{a}_i, \mathbf{x}^\natural \rangle|^2 + w_i, \qquad i = 1, 2, ..., n, \tag{1}$$

where $\mathbf{a}_i \in \mathbb{C}^p$ are given vectors, and w_i model the unknown noise. This problem arises in many applications, including X-ray crystallography, diffraction imaging, astronomical imaging and many others [1]–[5]. As an estimation problem, the nonlinear observation model (1) poses significant difficulties, since the standard maximum likelihood estimators yield in non-convex optimization problems.

Convex relaxations can be used in this setting as follows. We can equivalently express the quadratic measurements (1) as $b_i = \text{Tr}(\mathbf{a}_i \mathbf{a}_i^H \mathbf{x}^\natural \mathbf{x}^{\natural H}) + w_i$. This leads to the following linear observation model in the *lifted* dimensions of the variable $\mathbf{X}^\natural = \mathbf{x}^\natural \mathbf{x}^{\natural H} \in \mathbb{S}_+^{p\times p}$:

$$\mathbf{b} = \mathcal{A}(\mathbf{X}^\natural) + \mathbf{w}, \tag{2}$$

where $\mathcal{A} : \mathbb{S}_+^{p\times p} \to \mathbb{R}^n$ is a linear operator, denoting the positive semidefinite (PSD) cone by $\mathbb{S}_+^{p\times p}$. As a result, data fidelity costs can be measured using convex functions, albeit in terms of the matrix variables.

Since the lifted variable $\mathbf{X}^\natural$ is a rank-one positive semidefinite matrix, a naive approach for solving the phase retrieval problem is given by

$$\min_{\mathbf{X}\in\mathbb{S}_+^{p\times p}} \{\text{rank}(\mathbf{X}) : ||\mathcal{A}(\mathbf{X}) - \mathbf{b}|| \leq \epsilon\},$$

where $\|\cdot\|$ denotes the standard Euclidean norm, and ϵ is a parameter that depends on the noise model. The objective function of this problem is non-convex, and the problem is NP-hard. However, the standard convex relaxation of the rank function results in the following convex problem:

$$\min_{\mathbf{X}\in\mathbb{S}_+^{p\times p}} \{||\mathbf{X}||_* : ||\mathcal{A}(\mathbf{X}) - \mathbf{b}|| \leq \epsilon\}, \tag{3}$$

where $||\cdot||_*$ denotes the nuclear norm. This approach, which combines the semidefinite relaxation into the linear model and the convex relaxation of the rank function, is known as the PhaseLift [6], [7].

The statistical guarantees for the PhaseLift formulation are desirable. For example, for the noiseless case with $\mathbf{a}_i$ complex Gaussian, it is proved that $n \geq c_0 p$ for some constant c_0 suffices to recover $\mathbf{X}^\natural$ reliably [6], [7]. When $\mathbf{a}_i$ and w_i are Gaussian, the sample complexity $n \geq c_0 p$ is minimax optimal, and hence, essentially it can not be improved [8]. Moreover, the recovery is robust to noise [6].

Despite of its statistical appeals, PhaseLift and its variants have not found much practical use, since two critical bottlenecks severely restrict their scalability. The first bottleneck is the obvious curse-of-dimensionality of the lifting procedure. Given that the original signal dimension p can be large, the lifted formulation operates in $d = p \times p$ dimensions and imposes a major burden on the working memory in addition to the increased computation.

The second bottleneck revolves around the PSD cone constraint, which often times requires full eigendecompositions of $p \times p$ matrices requiring $\mathcal{O}(p^3)$ computational effort. As a result, practitioners prefer the non-convex optimization algorithms that directly operate on (1), in order to avoid these two bottlenecks [9]–[12].

However, it appears that the convex optimization algorithms did not reach yet the limits of their performance in phase retrieval. In this paper, we show that even some classical convex optimization algorithms can provide scalable solutions for the phase retrieval problem, with careful engineering, after the following simple reformulation:

$$\min_{\mathbf{X}\in\mathbb{S}_+^{p\times p}} \left\{ \frac{1}{2} ||\mathcal{A}(\mathbf{X}) - \mathbf{b}||^2 : \|\mathbf{X}\|_* \leq \kappa \right\}, \tag{4}$$

where we should ideally set $\kappa = \|\mathbf{X}^\natural\|_*$, which may not be practically feasible. Yet, as we show in Section II, we can estimate $\|\mathbf{X}^\natural\|_*$ in a precise fashion given the assumptions that certify the validity of the convex relaxation.

The objective function of (4) has Lipschitz continuous gradients. Given the trace-norm constraint, under which the proximal operator is computationally demanding, one of the most convenient algorithmic choices is the FW method (*cf.*, [13] for a review). To the best of our knowledge, this paper presents the first approach that numerically solves a PhaseLift variant with a Frank-Wolfe-type method.

The paper is organized as follows. We first demonstrate that we can estimate $\|\mathbf{X}^\natural\|_*$ with theoretical guarantees in Section II. We then illustrate how to set up the FW method, in Section III, and show that the scalability of (4) can be improved beyond FW method using the recent advances in universal primal-dual convex optimization algorithms [14]. Section IV is devoted to showing how to tame the memory growth and preserve cheap per-iteration costs with careful engineering. Finally, Section V presents numerical evidence to assess the scalability of the formulation (4) along with the proposed algorithms.

II. Theory

In this section, we show that $\bar{b} = \frac{1}{n}\sum_{i=1}^{n} b_i$ is a precise estimate of $\|\mathbf{X}^\natural\|_* = \|\mathbf{x}^\natural \mathbf{x}^{\natural H}\|_* = \|\mathbf{x}^\natural\|^2$, provided that some minor requirements are satisfied. Before stating our main result, we need a classical notion of the restricted isometry property (RIP) [15]:

Definition 1. *A linear operator $\mathbf{A}$ is said to be s-RIP if there exists a $\delta > 0$ such that $(1-\delta)||\mathbf{x}|| \leq ||\mathbf{A}\mathbf{x}|| \leq (1+\delta)||\mathbf{x}||$ for all s-sparse $\mathbf{x}$ (a signal with at most s non-zero entries).*

Theorem 1. *Let $\mathbf{A}$ be the matrix with rows $\mathbf{a}_i$ in (1). Assume that $\mathbf{A}$ satisfies RIP with probability $1-p_1$. Assume also that the noise in (1) are iid sub-Gaussian with sub-Gaussian norm σ. Suppose that the minimizer of (3) achieves estimation error $\mathbb{E}||\hat{\mathbf{X}} - \mathbf{X}^\natural||_F \leq C\frac{||\mathbf{x}^\natural||}{\sqrt{n}}$ with probability $1-p_2$. Then the minimizer of the program (4) achieves estimation error $\mathbb{E}||\hat{\mathbf{X}} - \mathbf{X}^\natural||_F \leq C(1+\delta+t)\frac{||\mathbf{x}^\natural||}{\sqrt{n}}$ with probability $1-p_1-p_2-\exp(-\frac{nt^2}{2\sigma^2})$ for any $t > 0$.*

The proof involves the application of the concentration of sub-Gaussian random variables around its mean, followed by the standard triangle inequality and the union bound arguments. We omit the details due to the space restriction.

As a consequence of *Theorem 1*, we can use the formulation (4), which allows us to use scalable convex optimization procedures, while we can achieve essentially the same statistical performance as (3). For illustration, let us first consider standard Gaussian measurements and sub-Gaussian noise model with zero mean and sub-Gaussian norm σ. In this setting, typical results for the PhaseLift (or its variants) are of the following form: for $n > c_0 p$, the minimizer of (3) satisfies $\mathbb{E}||\hat{\mathbf{X}} - \mathbf{X}^\natural||_F \leq C\frac{||\mathbf{x}^\natural||}{\sqrt{n}}$ with probability $1-\exp(-c'n)$ for some constants C, c_0, and c'. In other settings, the requirement for sample complexity might be more stringent, such as $n > c_0 p \log p$.

When $n > C_\delta s \log p$, Gaussian measurements satisfy s-RIP with probability $1-\exp(-c_1 n)$ [15]. Here C_δ and c_1 are constants depending only on δ. In the sparse regime where $s \log p < p$, this requirement is automatically satisfied by $n > c_0 p$, the sample complexity of phase retrieval problem. In the dense regime, we require $n > C_\delta p \log p$, so that an additional $\log p$ amount of samples must be included, as a minor requirement. With the sample requirement above, we can apply *Theorem 1* to conclude that our program (4) achieves estimation error $\mathbb{E}||\hat{\mathbf{X}} - \mathbf{X}^\natural||_F \leq C(1+\delta+t)\frac{||\mathbf{x}^\natural||}{\sqrt{n}}$ with probability $1-\exp(-c'n)-\exp(-c_1 n)-\exp(\frac{-nt^2}{2\sigma})$.

III. Algorithms

In this section, we illustrate how to set up the FW [13] and the accelerated universal primal-dual gradient methods [14] for solving the problem (4). For both algorithms, the key scalability workhorses are the following Fenchel-type operators (as opposed to the usual proximal-type operators), which we dub as *sharp*-operators:

$$[\mathbf{x}]^\sharp_\psi := \arg\min_{\mathbf{s}\in \mathrm{dom}\psi} \{\psi(\mathbf{s}) - \langle \mathbf{x}, \mathbf{s}\rangle\}, \tag{5}$$

where ψ is a convex function. As a special case, $\psi = I_{\mathcal{X}}$ corresponds to the so-called linear minimization oracle, where $I_{\mathcal{X}}$ represents the indicator function of the set $\mathcal{X}$.

Algorithm 1 presents the FW method. The main computational bottleneck is in computing $\mathbf{X}_k^*$. For the specific constraint set $\mathcal{X} = \mathbb{S}_+^{p\times p} \cap \{\mathbf{X} : ||\mathbf{X}||_* \leq \kappa\}$, it can be easily verified that $\mathbf{X}_k^*$ is a scaled rank-one approximation of $\nabla f(\mathbf{X}_k)$. More precisely, $\mathbf{X}_k^* = \kappa \mathbf{x}_k^* \mathbf{x}_k^{*H}$, where $\mathbf{x}_k^*$ is the unit norm eigenvector of $\nabla f(\mathbf{X}_k)$ that corresponds to the least eigenvalue, which can be computed efficiently by the (inverse) power method or the Lanczos algorithm, resulting in a cheap per-iteration complexity [13]. Note that, we do not need to form the gradient explicitly to find $\mathbf{x}_k^*$, and this saves a significant amount of memory space.

Algorithm 1 The Frank-Wolfe Algorithm (FW)

Template: $\min_{\mathbf{X}} \{f(\mathbf{X}) : \mathbf{X} \in \mathcal{X}\}$
Initialization: Choose an initial point $\mathbf{X}_0 \in \mathcal{X}$.
for $k = 0$ **to** $k_{\max}$ **do**
1. Compute $\mathbf{X}_k^* \in \arg\min_{\mathbf{X}\in\mathcal{X}} \langle \mathbf{X}, \nabla f(\mathbf{X}_k)\rangle \equiv [-\nabla f(\mathbf{X}_k)]^\sharp_{I_{\mathcal{X}}}$
2. Update $\mathbf{X}_{k+1} = (1-\gamma_k)\mathbf{X}_k + \gamma_k \mathbf{X}_k^*$, where $\gamma_k = \frac{2}{k+2}$.
end for

Another convenient convex optimization framework for solving (4) is our recently introduced universal primal-dual gradient method (UniPDGrad), and its accelerated variant (AccUniPDGrad) [14], which apply to the general constrained convex minimization template that includes (3). These algorithms leverage the Fenchel-type oracles (5), by exploiting the smoothness structure in the dual space with a special line-search strategy. Compared to the FW, our algorithms apply to a broader set of problems, and exhibit better numerical performance for some important real-world problems.

In this paper, we study the performance of the AccUniPDGrad and the FW methods in solving the problem (4). In order to apply AccUniPDGrad, we first reformulate (4) by introducing a slack variable $\mathbf{r} = \mathcal{A}(\mathbf{X}) - \mathbf{b}$:

$$\min_{\mathbf{X}\in\mathbb{S}_+^{p\times p}, \mathbf{r}\in\mathbb{R}^n} \left\{ \frac{1}{2}||\mathbf{r}||^2 : \mathcal{A}(\mathbf{X}) - \mathbf{b} - \mathbf{r} = \mathbf{0}, \|\mathbf{X}\|_* \leq \kappa \right\}.$$

Then, the Lagrangian corresponding to the equality constraint is

$$\mathcal{L}(\lambda, \mathbf{X}, \mathbf{r}) = \frac{1}{2}\|\mathbf{r}\|^2 + \langle \lambda, \mathcal{A}(\mathbf{X}) - \mathbf{b} - \mathbf{r}\rangle,$$

where λ is the dual variable. Denoting $\mathbf{X}^*(\lambda) \in [-\mathcal{A}^H(\lambda)]^\sharp_{I_{\mathcal{X}}} \equiv \arg\min_{\mathbf{X}\in\mathcal{X}} \langle \mathcal{A}^H(\lambda), \mathbf{X}\rangle$, we can write the negation of the Lagrange dual function as:

$$g(\lambda) = -\min_{\substack{\mathbf{X}\in\mathcal{X} \\ \mathbf{r}\in\mathbb{R}^n}} \mathcal{L}(\lambda, \mathbf{X}, \mathbf{r}) = \frac{1}{2}\|\lambda\|^2 + \langle \lambda, \mathbf{b} - \mathcal{A}(\mathbf{X}^\star(\lambda))\rangle.$$

We present the AccUniPDGrad method for the problem (4) in Algorithm 2.

IV. Implementation

This section shows that with careful engineering, we can often tame the memory growth and preserve cheap per-iteration costs with the convex methods above.

We start by writing the linear operator $\mathcal{A}(\cdot)$ and its Hermitian $\mathcal{A}^H(\cdot)$, in the following form:

$$\mathcal{A}(\mathbf{X}) = \mathrm{diag}(\mathbf{A}\mathbf{X}\mathbf{A}^H), \quad \text{and} \quad \mathcal{A}^H(\lambda) = \mathbf{A}^H \mathbf{D}_{(\lambda)} \mathbf{A}, \tag{6}$$

where $\mathbf{A}$ is the matrix with ith row $\mathbf{a}_i$, $\mathbf{D}_{(\lambda)}$ is the diagonal matrix with ith diagonal entry λ_i, and $\mathrm{diag}(\cdot)$ is the operator that takes a square matrix as an input, and returns its diagonal entries in a vector.

Note that, we use the Hermitian operator $\mathcal{A}^H(\cdot)$ only inside the sharp operator, i.e., to compute the eigenvector $\mathbf{x}_k^*$. Therefore, we do not need to form $\mathcal{A}^H(\hat{\lambda}_k)$ explicitly (or $\nabla f(\mathbf{X}_k) = \mathcal{A}^H(\mathcal{A}(\mathbf{X}_k) - \mathbf{b})$ for FW), which would require the multiplication of large matrices. Instead, we can compute $\mathbf{x}_k^*$ using the well known iterative approaches, such as the Lanczos method, which will require few matrix vector multiplications over $\mathbf{A}^H \mathbf{D}_{(\lambda)} \mathbf{A}$.

The second issue that we address in this section is the application of the linear operator $\mathcal{A}(\cdot)$, in order to form the gradient (∇f for FW

Algorithm 2 AccUniPDGrad for Phase Retrieval

Initialization: Choose an initial variable $\lambda_{(0)} = \hat{\lambda}_{(0)} \in \mathbb{R}^n$ and an accuracy term $\epsilon > 0$. Estimate M_0 the smoothness parameter. Set $S_{-1} = 0, t_0 = 1$ and $\mathbf{X}_0 = \mathbf{0}^p$.

for $k = 0$ **to** $k_{\max}$ **do**

1. Compute $\kappa \mathbf{x}_k^* \mathbf{x}_k^{*H} = \mathbf{X}_k^* \in [-\mathcal{A}^H(\hat{\lambda}_k)]_{I_{\mathcal{X}}}^{\sharp}$.
2. Form $\nabla g(\hat{\lambda}_k) = \hat{\lambda}_k + \mathbf{b} - \mathcal{A}(\mathbf{X}_k^*)$
3. **Line-search** Set $M_{k,0} = M_{k-1}$. For $i = 0$ to $i_{\max}$

3.a. $\lambda_{k,i} = \hat{\lambda}_k - M_{k,i}^{-1} \nabla g(\hat{\lambda}_k)$.

3.b. If the following condition holds:

$$g(\lambda_{k,i}) \leq g(\hat{\lambda}_k) - \frac{1}{2M_{k,i}} \|\nabla g(\hat{\lambda}_k)\|^2 + \frac{\epsilon}{t_k},$$

set $i_k = i$ and terminate the line-search. Otherwise, set $M_{k,i+1} = 2M_{k,i}$.

End of line-search

4. Set and $\lambda_{k+1} = \lambda_{k,i_k}$ and $M_k = M_{k,i_k}$.
5. Compute $\omega_k = t_k / M_k$, $S_k = S_{k-1} + \omega_k$ and $\gamma_k = \omega_k / S_k$.
6. Set $t_{k+1} = \frac{1}{2}[1 + \sqrt{1 + 4t_k^2}]$.
7. Update $\hat{\lambda}_{k+1} = \lambda_{k+1} + \frac{t_k - 1}{t_{k+1}} (\lambda_{k+1} - \lambda_k)$.
8. Update $\mathbf{X}_{k+1} = (1 - \gamma_k)\mathbf{X}_k + \gamma_k \mathbf{X}_k^*$.

end for

and ∇g for AccUniPDGrad). Fortunately, we need to apply $\mathcal{A}(\cdot)$ only to $\mathbf{X}_k^*$ in both algorithms, which is known to be the rank-one matrix $\kappa \mathbf{x}_k^* \mathbf{x}_k^{*H}$. As a result, we can equivalently apply the cheap nonlinear operator (1), since $(\mathcal{A}(\mathbf{X}_k^*))_i = \kappa |\langle \mathbf{a}_i, \mathbf{x}_k^* \rangle|^2$.

Finally, in the weighting step of the algorithms, we adapt the efficient thin singular value decomposition (SVD) approach of [16], which we describe briefly as follows: Denote the thin singular value decomposition of the estimate $\mathbf{X}_k$ at iteration k as $\mathbf{U}_k \mathbf{S}_k \mathbf{V}_k^H$, then the SVD of $\mathbf{X}_{k+1}$ is given by

$$\mathbf{U}_{k+1} \mathbf{S}_{k+1} \mathbf{V}_{k+1}^H = ([\mathbf{U}_k \;\; \hat{\mathbf{p}}]\, \underline{\mathbf{U}})\, \underline{\mathbf{S}}\, ([\mathbf{V}_k \;\; \hat{\mathbf{q}}]\, \underline{\mathbf{V}})^H,$$

where $\mathbf{p} = \mathbf{x}_k^* - \mathbf{U}_k \mathbf{U}_k^H \mathbf{x}_k^*$, $\mathbf{q} = \mathbf{x}_k^* - \mathbf{V}_k \mathbf{V}_k^H \mathbf{x}_k^*$, $\hat{\mathbf{p}}$ and $\hat{\mathbf{q}}$ are the unit vectors along the directions of $\mathbf{p}$ and $\mathbf{q}$, and $\underline{\mathbf{U}}\, \underline{\mathbf{S}}\, \underline{\mathbf{V}}^H$ is the SVD of $\mathbf{K}$:

$$\mathbf{K} = (1 - \gamma_k) \begin{bmatrix} \mathbf{S}_k & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix} + \gamma_k \kappa \begin{bmatrix} \mathbf{U}_k^H \mathbf{x}_k^* \\ \|\mathbf{p}\| \end{bmatrix} \begin{bmatrix} \mathbf{V}_k^H \mathbf{x}_k^* \\ \|\mathbf{q}\| \end{bmatrix}^H.$$

While this methods is not theoretically guaranteed to alleviate the expansion of the rank, as we will demonstrate numerically below, it often keeps a low memory footprint. Note that, we do not need to form any variable in the ambient lifted dimension p^2, throughout the algorithms. Instead, we directly update the SVD of $\mathbf{X}_k$, by using the p dimensional vector $\sqrt{\kappa} \mathbf{x}_k^*$.

V. Numerical Results

In this section, we present numerical evidence to assess the scalability of the formulation (4) along with the proposed algorithms. Our numerical experiments are based on the coded diffraction pattern measurements with octonary modulation, which is also considered in [12], [17]. We consider the random design of the modulating waveforms from [12]. A similar setup is also considered in [17].

In our first two experiments, we consider the random Gaussian signal model: $\mathbf{x}^\natural \in \mathbb{C}^p$ is a random Gaussian vectors with i.i.d. entries, where the real and the imaginary parts of the each entry entry of $\mathbf{x}^\natural$ and sampled from the standard Gaussian distribution.

In the first experiment, we show the estimation accuracy of the constraint parameter. For each data size p, we generate 100 random Gaussian signals, and we take the samples by modulating each signal with $L = 20$ random waveforms. For each signal, we compute the normalized error ($|\bar{b} - \|\mathbf{x}^\natural\|^2| / \|\mathbf{x}^\natural\|^2$) in the parameter estimation. The solid line in the first plot of Figure 1 presents the average error over 100 trials, and the shaded area shows the distribution of the errors.

In the second experiment, we compare our framework against the original PhaseLift formulation. For this purpose, we measure the computational time to reach 10^{-2} reconstruction error ($\|\mathbf{x}^\natural - \mathbf{x}\| / \|\mathbf{x}^\natural\|$) using the Frank-Wolfe, the AccUniPDGrad, and the solver provided in [17], which is based on the MATLAB package TFOCS [18], and solves the formulation (3) by using the Auslender and Teboulle's method [19]. The second plot in Figure 1 shows the average performance over 10 random trials for each data size p. We set the constraint parameter $\kappa = \|\mathbf{x}^\natural\|^2$, since the estimate $\bar{b}$ is not accurate when p is small.

We test our framework on some images of different sizes, in the third experiment. The first image is an EPFL campus image of size 1280×720, and the second one is the Milky Way galaxy of size 1920×1080 from [12]. Each image consists of 3 color channels, and we take the samples by modulating each channel with $L = 20$ random waveforms as in [12]. We start both algorithms from the zero vector, and we set the constraint parameter $\kappa = \bar{b}$. We keep track of the reconstruction error as the performance measure.

The last two plots of Figure 1 present the performance of the Frank-Wolfe and the AccUniPDGrad methods, by showing the reconstruction error ($\|\mathbf{x}^\natural - \mathbf{x}\|_F / \|\mathbf{x}^\natural\|_F$) with respect to the iteration counter and the computational time. The plots correspond to the averaged performance over the three color channels, and the solid and the dashed lines correspond to the Milky Way and the EPFL campus images respectively.

Finally, Figure 2 and 3 show the reconstructed images of EPFL campus and the Milky Way respectively, with the AccUniPDGrad algorithm. Figure 2 corresponds to the estimate $\mathbf{x_k}$ after 41 iterations of the AccUniPDGrad. The PSNR of the reconstructed image is 45.54 dB. Similarly, Figure 3 corresponds to the estimate $\mathbf{x_k}$ after 40 iterations, and the PSNR is 54.44 dB.

We implement all three experiments in MATLAB and use the built-in `eigs` function (which is based on the Lanczos algorithm) to evaluate the sharp operator, with 10^{-3} relative error tolerance. We time our experiments on a computer cluster, and restricting the computational resource to 8 CPU of 2.40 GHz and 32 GB of memory space per simulation.

VI. Conclusions

Despite of its statistical appeal, PhaseLift has been thought to result in computationally difficult convex optimization problems with limited practical use. In this paper, we show that the phase retrieval problem can be solved efficiently, even in extremely high dimensions, using a variant of PhaseLift formulation and appropriate convex optimization algorithms. While non-convex approaches are useful in solving phase retrieval, they typically apply to specific problem formulations. Convex approaches, on the other side, can handle more general objectives scalably. Although our main focus in this paper was the formulation (4), AccUniPDGrad method also applies to the original PhaseLift formulation (3), and even the Poisson phase retrieval can have scalable convex solutions [20].

Acknowledgments

This work was supported in part by ERC Future Proof, SNF 200021-146750 and SNF CRSII2-147633.

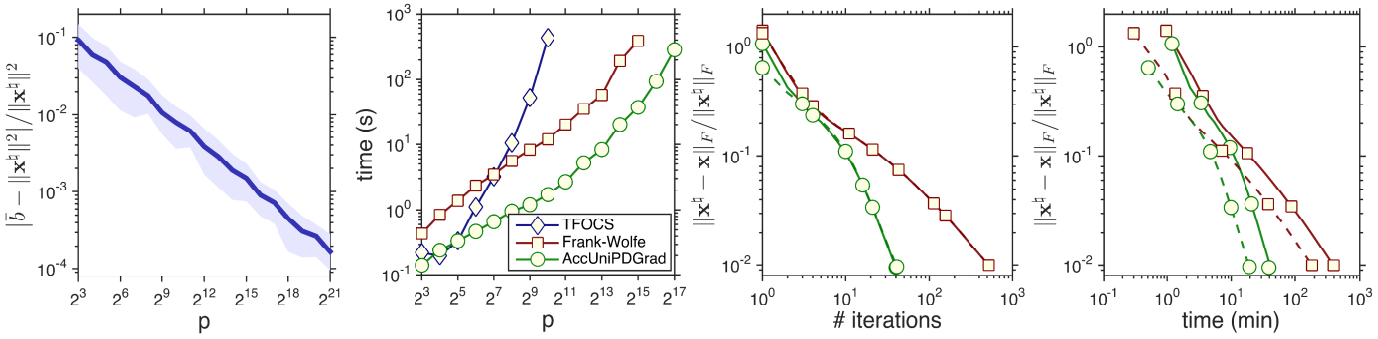

Fig. 1. The first plot shows the normalized error in the constraint parameter estimation, and the second one presents the time required by each algorithm to reach 10^{-2} reconstruction error, with respect to the data size p. The last two plots illustrate the convergence behavior of the algorithms in the test with real images. The solid lines correspond to the Milky Way, and the dashed lines correspond to the EPFL image.

Fig. 2. EPFL image of size 1280×720, reconstructed in 20 minutes by 41 iterations of the AccUniPDGrad: PSNR = 45.54 dB

Fig. 3. Milky Way image of size 1920×1080, reconstructed in 42 minutes by 40 iterations of the AccUniPDGrad: PSNR = 54.44 dB

REFERENCES

[1] R. P. Millane, "Phase retrieval in crystallography and optics," *J. Opt. Soc. Am. A*, vol. 7, no. 3, pp. 394–411, Mar 1990.

[2] A. Walther, "The question of phase retrieval in optics," *Optica Acta: International Journal of Optics*, vol. 10, no. 1, pp. 41–49, 1963.

[3] R. W. Harrison, "Phase problem in crystallography," *J. Opt. Soc. Am. A*, vol. 10, no. 5, pp. 1046–1055, 1993.

[4] O. Bunk, A. Diaz, F. Pfeiffer, C. David, B. Schmitt, D. K. Satapathy, and J. F. van der Veen, "Diffractive imaging for periodic samples: retrieving one-dimensional concentration profiles across microfluidic channels," *Acta Crystallographica Section A*, vol. 63, no. 4, pp. 306–314, 2007.

[5] Y. Shechtman, Y. C. Eldar, O. Cohen, H. N. Chapman, J. Miao, and M. Segev, "Phase retrieval with application to optical imaging," *IEEE Sig. Process. Mag.*, vol. 32, no. 3, pp. 87–109, 2015.

[6] E. J. Candès, T. Strohmer, and V. Voroninski, "PhaseLift: Exact and stable signal recovery from magnitude measurements via convex programming," *Commun. Pure Appl. Math.*, vol. LXVI, pp. 1241–1274, 2013.

[7] E. J. Candès, Y. C. Eldar, T. Strohmer, and V. Voroninski, "Phase retrieval via matrix completion," *SIAM J. Imaging Sci.*, vol. 6, no. 1, pp. 199–225, 2013.

[8] G. Lecué and S. Mendelson, "Minimax rate of convergence and the performance of empirical risk minimization in phase recovery," *Electron. J. Probab.*, vol. 20, no. 57, pp. 1–29, 2015.

[9] S. Mukherjee and C. S. Seelamantula, "An iterative algorithm for phase retrieval with sparsity constraints: application to frequency domain optical coherence tomography," in *IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 2012, pp. 553–556.

[10] P. Netrapalli, P. Jain, and S. Sanghavi, "Phase retrieval using alternating minimization," in *Adv. Neural Information Processing Systems 26*, 2013.

[11] Y. Shechtman, A. Beck, and Y. Eldar, "Gespar: Efficient phase retrieval of sparse signals," *IEEE Trans. Signal Process.*, vol. 62, no. 4, pp. 928–938, 2014.

[12] E. J. Candès, X. Li, and M. Soltanolkotabi, "Phase retrieval via Wirtinger flow: Theory and algorithms," *IEEE Trans. Inf. Theory*, vol. 61, no. 4, pp. 1985–2007, 2015.

[13] M. Jaggi, "Revisiting Frank-Wolfe: Projection-free sparse convex optimization," in *Proc. 30th Int. Conf. Machine Learning*, 2013.

[14] A. Yurtsever, Q. Tran-Dinh, and V. Cevher, "A universal primal-dual convex optimization framework," in *29th Ann. Conf. Neural Information Processing Systems*, 2015.

[15] E. J. Candès, "The restricted isometry property and its implications for compressed sensing," *C. R. Acad. Sci. Paris, Ser. I*, vol. 346, pp. 589–592, 2008.

[16] M. Brand, "Fast low-rank modifications of the thin singular value decomposition," *Linear Algebra Appl.*, vol. 415, no. 1, pp. 20–30, 2006.

[17] E. J. Candès, X. Li, and M. Soltanolkotabi, "Phase retrieval from coded diffraction patterns," *Appl. Comput. Harmon. Anal.*, vol. 39, pp. 277–299, 2015.

[18] S. R. Becker, E. J. Candès, and M. C. Grant, "Templates for convex cone problems with applications to sparse signal recovery," *Math. Prog. Comp.*, vol. 3, pp. 165–218, 2011.

[19] A. Auslender and M. Teboulle, "Interior gradient and proximal methods for convex and conic optimization," *SIAM J. Optim.*, vol. 16, no. 3, pp. 697–725, 2006.

[20] G. Odor, Y.-H. Li, A. Yurtsever, Y.-P. Hsieh, Q. Tran-Dinh, and V. Cevher, "Frank-Wolfe works for non-Lipschitz continuous gradient objectives: Scalable poisson phase retrieval," 2015.