ACCELERATION OF PRIMAL-DUAL METHODS BY PRECONDITIONING AND FIXED NUMBER OF INNER LOOPS*

YANLI LIU†, YUNBEI XU‡, AND WOTAO YIN†

Abstract. Primal-Dual Hybrid Gradient (PDHG) and Alternating Direction Method of Multipliers (ADMM) have been widely used due to their wide applicability and easy implementation. However, they may suffer from slow tail convergence. In view of this, many techniques have been proposed to accelerate them, including preconditioning and inexact solve of the subproblems. In this paper, we integrate these two techniques to achieve a further acceleration. Specifically, we give a criterion for choosing good preconditioners, and propose to solve one of the subproblems by only a fixed number (usually very few) of inner loops of several common routines. Global convergence is established for the proposed scheme. Since our method overcomes the previous restriction of choosing only diagonal preconditioners, we obtain significant accelerations on several popular applications.

Key words. Primal-Dual Hybrid Gradient, Alternating Direction Method of Multipliers, preconditioning, fixed number of inner loops, structured subproblem, suitable subproblem solver

AMS subject classifications. 49M29, 65K10, 65Y20, 90C25

1. Introduction.

In this paper, we consider the following optimization problem:

$$\underset{x\in\mathbb{R}^n}{\text{minimize}}\, f(x)+g(Ax), \tag{1.1}$$

together with its dual problem:

$$\underset{z\in\mathbb{R}^m}{\text{minimize}}\, f^*(-A^Tz)+g^*(z). \tag{1.2}$$

Here $f:\mathbb{R}^{\mathrm{n}}\to\mathbb{R}\cup\{+\infty\}$ and $g:\mathbb{R}^{\mathrm{m}}\to\mathbb{R}\cup\{+\infty\}$ are closed proper convex, and $A\in\mathbb{R}^{m\times n}$ is a matrix, f^* and g^* are the convex conjugates of f and g, respectively.

Many practical problems can be formulated in the form of (1.1) or (1.2), for example, image restoration [39], magnetic resonance imaging [35], network optimization [15], computer vision [30], and earth mover's distance [22].

Primal-Dual algorithms such as Primal-Dual Hybrid Gradient (PDHG) and Alternating Direction Method of Multipliers (ADMM) can be applied to solve (1.1). However, PDHG and ADMM suffer from slow (tail) convergence in practice. They may take more than a few thousand iterations and still cannot reach four digits of accuracy. In general, their performance is very sensitive to problem conditions. Therefore, efforts have been made to accelerate them. In the next subsection, we review two common acceleration techniques: preconditioning and inexact solve of subproblems.

1.1. Background.

The convergence rate of PDHG depends on its step sizes, which need an estimate of the operator norm of A. To accelerate PDHG and avoid estimating the norm of A, diagonal preconditioning [29] was proposed and analyzed. This technique improves the iteration complexity and adds only little computational cost per iteration. However, non-diagonal preconditioners can further reduce the iteration complexity significantly, but it remains open to apply such preconditioners

*Submitted to the editors XXX 2018.

Funding: This work is supported in part by NSF grant DMS-1720237 and ONR grant N000141712162.

†Department of Mathematics, University of California, Los Angeles, CA (yanli@math.ucla.edu, wotaoyin@math.ucla.edu).

‡Graduate School of Business, Columbia University, New York, NY (yunbei.xu@gsb.columbia.edu).

while still maintaining the computation cost per iteration. As another acceleration technique, *inexact PDHG* allows the PDHG subproblems to be solved approximately. To ensure convergence, [31] uses three types of errors to control the solution errors of the subproblems; all of them need to be summable over the iterations. Therefore, [31] requires the subproblems to be solved with increasing accuracies.

Unlike PDHG, a subproblem of ADMM minimizers the sum of $f(x)$ and a squared term involving Ax. In general, it may not have a closed form solution. Several versions of inexact ADMM have been studied. An *absolute error criterion* is introduced in [11], where the subproblem errors are controlled by sequences of error tolerances that are summable. To simplify the choice of the sequences, the *relative error criterion* was adopted in several works, where the subproblem errors are controlled by a single parameter and quantities generated naturally by the algorithm. In [26], the parameters need to be square summable. In [21], the parameters are constants but both objectives are required to be Lipschitz differentiable. In [12, 13], two possible outcomes of the algorithm are described: (i) infinite outer loops and finite inner loops, and (ii) finite out loops and the last inner loop is infinite, both of them guaranteeing convergence to a solution. On the other hand, it is unclear how to distinguish them, and since there is no bound on the number of inner loops in case (i), one may recognize it as case (ii) and stop the algorithm before it converges.

Certain types of preconditioning have been applied to accelerate ADMM. In [17], diagonal preconditioning is used with ADMM. After that, non-diagonal preconditioning is also analyzed [5, 6], where appropriate preconditioners are given for specific applications, and competitive numerical performances are observed. This work inverts one of the preconditioners (not needed in our method). Recently, preconditioning for strongly convex problems has also been discussed [18].

1.2. Contributions.

The contributions of this paper are three-fold.

First, we provide a criterion for choosing preconditioners based on an ergodic convergence result. We also show that ADMM corresponds to a special choice of preconditioners in Preconditioned PDHG (PPDHG).

Second, we show that PPDHG converges when its subproblem is solved inexactly to a specified relative-error condition. Remarkably, this condition does not need to be checked since it is naturally satisfied when one applies a *fixed number* of inner loops using any of several common subproblem solvers including proximal gradient descent, FISTA with restart, proximal block coordinate descent, as well as even faster block-coordinate-gradient-descent (BCGD) methods (e.g., [24, 1, 19]).

Third, the diverse choice of subproblem solution methods, especially the BCGD methods, lets us deal with the difficult subproblems that arise when we apply preconditioners. With appropriate preconditioners and subproblem solvers, both PDHG and ADMM can be accelerated and their total running time significantly reduced. The efficiency of our algorithm is demonstrated by numerical experiments.

It is worth mentioning that our fixed number of inner loops is different from the "finite inner loops" claimed in [5, 6]. In their settings, one subproblem is essentially applying a preconditioned proximal operator to a convex quadratic function at the current iterate, which has a closed form solution. The same operator is applied n times starting at the current iterate for any $n \geq 1$, and convergence can still be established, they call these n operations finite inner loops.

1.3. Organization.

The rest of this paper is organized as follows: Section 2 establishes notation and reviews some basic results. In Section 3, we first provide a criterion for choosing preconditioners of PDHG, then introduce the bounded relative error condition and show that it is satisfied by a fixed number of inner loops. We establish convergence of the inexact preconditioned PDHG. Section 4 provides numerical examples on several popular applications. Finally, Section 5 concludes the paper.

2. Preliminaries. In this section, we review some basic concepts, introduce our notation, and state some known results. For the sake of brevity, we omit proofs and direct references. We refer the reader to the textbook [3].

We use $\|\cdot\|$ for the ℓ_2-norm, and $\langle\cdot,\cdot\rangle$ for the usual dot product. $M \succ 0$ denotes a symmetric positive definite matrix M, and $M \succeq 0$ denotes a symmetric positive semidefinite matrix M.

$\lambda_{\min}(M)$ and $\lambda_{\max}(M)$ stand for the smallest eigenvalue and the largest eigenvalues of M, respectively. $\kappa(M) = \frac{\lambda_{\max}(M)}{\lambda_{\min}(M)}$ is the condition number of M. For $M \succeq 0$, let $\|\cdot\|_M$ and $\langle\cdot,\cdot\rangle_M$ denote the (semi-)norm and inner product induced by M, respectively.

For a proper closed convex function $\phi : \mathbb{R}^{\mathrm{n}} \to \mathbb{R} \cup \{+\infty\}$, its subdifferential at $x \in \mathbf{dom} f$ is defined by

$$\partial\phi(x) = \{v \in \mathbb{R}^{\mathrm{n}} \mid \phi(z) \geq \phi(x) + \langle v, z - x\rangle \ \forall z \in \mathbb{R}^{\mathrm{n}}\},$$

and its convex conjugate is

$$\phi^*(y) = \sup_{x\in\mathbb{R}^n} \{\langle y, x\rangle - \phi(x)\},$$

we have $y \in \partial\phi(x)$ if and only if $x \in \partial\phi^*(y)$.

For any symmetric $M \succ 0$, we define the extended proximal operator of ϕ to be

$$\operatorname{Prox}_\phi^M(x) := \arg\min_{y\in\mathbb{R}^{\mathrm{n}}} \{\phi(y) + \frac{1}{2}\|y - x\|_M^2\}, \tag{2.1}$$

When $M = \gamma^{-1}I$ where $\gamma > 0$, it reduces to the classic proximal operator.

For the extended proximal operator (2.1) we also have the following generalization of Moreau's Identity:

LEMMA 2.1 ([10], Theorem 3.1(ii)). *For any proper closed convex function ϕ and $M \succ 0$, we have*

$$x = \operatorname{Prox}_\phi^M(x) + M^{-1} \operatorname{Prox}_{\phi^*}^{M^{-1}}(Mx). \tag{2.2}$$

A proper closed function is said to be a Kurdyka-Łojasiewicz (KŁ) function, if for each $x_0 \in \mathbf{dom} f$, there exist $\eta \in (0, \infty]$, a neighborhood U of x_0 and a continuous concave function $\varphi : [0, \eta) \to \mathbb{R}_+$ such that:

1. $\varphi(0) = 0$,
2. φ is C^1 on $(0, \eta)$,
3. for all $s \in (0, \eta), \varphi'(s) > 0$,
4. for all $x \in U \cap \{x \mid f(x_0) < f(x) < f(x_0) + \eta\}$, the KŁ inequality holds:

$$\varphi'(f(x) - f(x_0))\operatorname{dist}(0, \partial f(x)) \geq 1.$$

3. Acceleration of PDHG. Throughout this section, the following regularity assumption is assumed:

ASSUMPTION 1.

1. *$f : \mathbb{R}^{\mathrm{n}} \to \mathbb{R} \cup \{+\infty\}$, $g : \mathbb{R}^{\mathrm{m}} \to \mathbb{R} \cup \{+\infty\}$ are proper closed convex.*

2. *A primal-dual solution pair* $(x^\star, z^\star)$ *of* (1.1) *and* (1.2) *exists, i.e.,*

$$\mathbf{0} \in \partial f(x^\star) + A^T z^\star, \quad \mathbf{0} \in \partial g(Ax^\star) - z^\star.$$

The problem (1.1) also has the following convex-concave saddle-point formulation:

$$\min_{x\in\mathbb{R}^n} \max_{z\in\mathbb{R}^m} \varphi(x,z) \coloneqq f(x) + \langle Ax, z\rangle - g^*(z) \tag{3.1}$$

A primal-dual solution pair $(x^\star, z^\star)$ is a solution of (3.1) and vice versa.

3.1. Preconditioned PDHG. The method of Primal-Dual Hybrid Gradient (PDHG) [39, 7] for solving (1.1) uses the iteration

$$\begin{aligned} x^{k+1} &= \mathrm{Prox}_{\tau f}(x^k - \tau A^T z^k),\\ z^{k+1} &= \mathrm{Prox}_{\sigma g^*}(z^k + \sigma A(2x^{k+1} - x^k)). \end{aligned} \tag{3.2}$$

Convergence of (3.2) to a primal-dual solution pair of (1.1) is established when $\frac{1}{\tau\sigma} \geq \|A\|^2$ [7]. In order to achieve faster convergence by exploiting the structure of subproblems, we can apply preconditioners $M_1, M_2 \succ 0$ (their choices are discussed below) to obtain Preconditioned PDHG (PPDHG):

$$\begin{aligned} x^{k+1} &= \mathrm{Prox}_f^{M_1}(x^k - M_1^{-1} A^T z^k),\\ z^{k+1} &= \mathrm{Prox}_{g^*}^{M_2}(z^k + M_2^{-1} A(2x^{k+1} - x^k)), \end{aligned} \tag{3.3}$$

where the extended proximal operators $\mathrm{Prox}_f^{M_1}$ and $\mathrm{Prox}_{g^*}^{M_2}$ are defined in (2.1).

Note that there is no need to form M_1^{-1} and M_2^{-1} since (3.3) is equivalent to

$$\begin{aligned} x^{k+1} &= \underset{x\in\mathbb{R}^n}{\arg\min}\{f(x) + \langle x - x^k, A^T z^k\rangle + \frac{1}{2}\|x - x^k\|_{M_1}^2\},\\ z^{k+1} &= \underset{z\in\mathbb{R}^m}{\arg\min}\{g^*(z) - \langle z - z^k, A(2x^{k+1} - x^k)\rangle + \frac{1}{2}\|z - z^k\|_{M_2}^2\}. \end{aligned} \tag{3.4}$$

3.2. Choice of preconditioners by an ergodic convergence result. The convergence of PPDHG is not new. In fact, PPDHG is a special case of a general primal-dual algorithm considered in [8]. In this section, we discuss how to select appropriate preconditioners M_1 and M_2 based on an ergodic convergence result from [8]. In particular, we show that ADMM corresponds to the choice $M_1 = \frac{1}{\tau} I_{n\times n}, M_2 = \tau AA^T$, which has faster convergence than PDHG in terms of outer iterations.

Let us start with the following lemma which characterizes primal-dual solution pairs of (1.1) and (1.2).

LEMMA 3.1. *Under Assumption 1,* (X, Z) *is a primal-dual solution pair of* (1.1) *if and only if* $\varphi(X, z) - \varphi(x, Z) \leq 0$ *for any* $(x, z) \in \mathbb{R}^{n+m}$.

Proof. If (X, Z) is a primal-dual solution pair of (1.1), then

$$-A^T Z \in \partial f(X), \quad AX \in \partial g^*(Z).$$

As a result, for any $(x, z) \in \mathbb{R}^{n+m}$ we have

$$f(x) \geq f(X) + \langle -A^T Z, x - X\rangle, \quad g^*(z) \geq g^*(Z) + \langle AX, z - Z\rangle,$$

adding them together gives $\varphi(X, z) - \varphi(x, Z) \leq 0$.

On the other hand, if $\varphi(X,z) - \varphi(x,Z) \le 0$ for any $(x,z) \in \mathbb{R}^{n+m}$, then

$$\langle AX, z\rangle + f(X) - g^*(z) - \langle Ax, Z\rangle - f(x) + g^*(Z) \le 0 \quad \text{for any } (x,z) \in \mathbb{R}^{n+m}.$$

Taking $x = X$ yields $\langle AX, z - Z\rangle - g^*(z) + g^*(Z) \le 0$, so $AX \in \partial g^*(Z)$; Similarly, taking $z = Z$ gives $\langle AX - Ax, Z\rangle + f(X) - f(x) \le 0$, so $-A^T Z \in \partial f(X)$. As a result, (X, Z) is a primal-dual solution pair of (1.1). ∎

On the other hand, we have the following ergodic convergence result, which is adapted from Theorem 1 of [8].

THEOREM 3.2. *Let $(x^k, z^k), n = 0, 1, ..., N$ be a sequence generated by PPDHG (3.3). Under Assumption 1, if in addition*

$$\tilde{M} := \begin{pmatrix} M_1 & -A^T \\ -A & M_2 \end{pmatrix} \succeq 0, \tag{3.5}$$

then, for any $x \in \mathbb{R}^{\mathrm{n}}$ and $z \in \mathbb{R}^{\mathrm{m}}$, it holds that

$$\varphi(X^N, z) - \varphi(x, Z^N) \le \frac{1}{2N}(x - x^0, z - z^0) \begin{pmatrix} M_1 & -A^T \\ -A & M_2 \end{pmatrix} \begin{pmatrix} x - x^0 \\ z - z^0 \end{pmatrix}, \tag{3.6}$$

where $X^N = \frac{1}{N}\sum_{i=1}^N x_i$ and $Z^N = \frac{1}{N}\sum_{i=1}^N z_i$.

Proof. This follows from Theorem 1 and Remark 3 of [8], by setting $L_f = 0$, $\frac{1}{\tau}D_x(x, x_0) = \frac{1}{2}\|x - x^0\|^2_{M_1}$, $\frac{1}{\sigma}D_z(z, z_0) = \frac{1}{2}\|z - z^0\|^2_{M_2}$, and $K = A$. ∎

In view of Lemma 3.1 and Theorem 3.2, in order to accelerate the ergodic convergence, the preconditioners M_1 and M_2 should be chosen such that (3.5) is satisfied, and the right hand side of (3.6) should be small. In view of this, we can obtain some useful criteria for choosing M_1 and M_2.

First, by Schur complement lemma, the condition (3.5) is equivalent to $M_2 \succeq AM_1^{-1}A^T$. Hence, for a given M_1, the optimal M_2 is $AM_1^{-1}A^T$.

Second, PDHG (3.2) corresponds to $M_1 = \frac{1}{\tau}I_{n\times n}$, $M_2 = \frac{1}{\sigma}I_{m\times m}$ with $\frac{1}{\tau\sigma} \ge \|A\|^2$. On the other hand, one can show that ADMM applied to (1.1) corresponds to $M_1 = \frac{1}{\tau}I_{n\times n}, M_2 = \tau AA^T$ (see Appendix A, this is also implicitly shown in [7, Sec. 4.3]), where M_2 is optimal for M_1 since $AM_1^{-1}A^T = \tau AA^T = M_2$. In this regard, ADMM corresponds to a better choice of preconditioners than PDHG, which explains why ADMM uses fewer outer iterations than PDHG in practice. This is also verified in our numerical experiments in Section 4.

Finally, there might be better choices of preconditioners than that of ADMM. This can bring us faster algorithms and is left as future work.

3.3. PPDHG with fixed finite inner loops. Solving the subproblems in (3.3) exactly or nearly so is wasteful. Choosing the number of inner loops based on a condition requires checking the condition. It is convenient if we can simply fix the number of inner loops.

In this subsection, we describe the "bounded relative error" of the $z-$subproblem in (3.3) and then show that this can be satisfied by running a fixed number of inner loops, uniformly for every outer loop.

DEFINITION 3.3. *Given x^k, x^{k+1} and z^k, the $z-$subproblem in PPDHG (3.3) is said to be solved with bounded relative error if there is a constant $c > 0$ such that*

$$\mathbf{0} \in \partial g^*(z^{k+1}) + M_2\big(z^{k+1} - z^k - M_2^{-1}A(2x^{k+1} - x^k)\big) + \varepsilon^{k+1}, \tag{3.7}$$

$$\|\varepsilon^{k+1}\| \le c\|z^{k+1} - z^k\|. \tag{3.8}$$

Remarkably, this condition does not need to be checked at any iteration. For a given $c > 0$, it can be satisfied by a fixed number of inner loops using proximal gradient descent (see Theorem 3.4). One can also use faster solvers for the $z-$subproblem, e.g., FISTA with restart [27], and solvers that suit the subproblem structure, e.g., cyclic proximal BCD (see Theorem 3.6). Although the error in solving $z-$subproblems appears to be neither summable nor square summable at first glance, convergence can still be established. We summarize our algorithm in Algorithm 3.1.

Algorithm 3.1 Inexact preconditioned PDHG

Input: $f : \mathbb{R}^n \to \mathbb{R}, g : \mathbb{R}^m \to \mathbb{R}$, $A \in \mathbb{R}^{m\times n}$, preconditioners M_1 and M_2, initial (x_0, z_0), subproblem solver S for the $z-$subproblem in (3.3), fixed inner iterations p, max outer iterations K.

Output: (x^K, z^K)

```
for k ← 0, 1, ..., K − 1 do
    x^{k+1} = Prox_f^{M_1}(x^k − M_1^{-1} A^T z^k);
    z_0^{k+1} = z^k;
    for i ← 0, 1, ..., p − 1 do
        z_{i+1}^{k+1} = S(z_i^{k+1}, x^{k+1}, x^k);
    end for
    z^{k+1} = z_p^{k+1};        ▷ approximate Prox_{g*}^{M_2}(z^k + M_2^{-1} A(2x^{k+1} − x^k))
end for
```

THEOREM 3.4. *Under Assumption 1, if $p \geq 1$ iterations of proximal gradient descent with stepsize $\gamma \in (0, \frac{2\lambda_{\min}(M_2)}{\lambda^2_{\max}(M_2)})$ are applied to solve the $z-$subproblem in (3.3), and is initialized with the last iterate z^k, then the subproblem is solved with bounded relative error with the following constant for (3.8)*

$$c = c(p) = \frac{\frac{1}{\gamma} + \lambda_{\max}(M_2)}{1 - \tau^p}(\tau^p + \tau^{p-1}), \tag{3.9}$$

where $\tau = \sqrt{1 - \gamma(2\lambda_{\min}(M_2) - \gamma\lambda^2_{\max}(M_2))} < 1$.

Proof. The $z-$subproblem in (3.4) is of the form

$$\min_{z\in\mathbb{R}^m} h_1(z) + h_2(z), \tag{3.10}$$

where

$$\begin{aligned} h_1(z) &= g^*(z), \\ h_2(z) &= \frac{1}{2}\|z - z^k - M_2^{-1}A(2x^{k+1} - x^k)\|^2_{M_2}. \end{aligned}$$

In Algorithm 3.1, an inexact z^{k+1} is given by

$$\begin{aligned} z_0^{k+1} &= z^k, \\ z_{i+1}^{k+1} &= \mathrm{Prox}_{\gamma h_1}(z_i^{k+1} - \gamma\nabla h_2(z_i^{k+1})), \quad i = 0, 1, ..., p-1, \\ z^{k+1} &= z_p^{k+1}. \end{aligned}$$

The optimality condition of the last iteration above reads

$$\mathbf{0} \in \partial h_1(z_p^{k+1}) + \nabla h_2(z_{p-1}^{k+1}) + \frac{1}{\gamma}(z_p^{k+1} - z_{p-1}^{k+1}),$$

compare this with (3.7) and use $z_p^{k+1} = z^{k+1}$, we have

$$\varepsilon^{k+1} = \frac{1}{\gamma}(z_p^{k+1} - z_{p-1}^{k+1}) + \nabla h_2(z_{p-1}^{k+1}) - \nabla h_2(z_p^{k+1}),$$

we need to show that ε^{k+1} satisfies (3.8).

Let $z_\star^{k+1}$ be the solution of (3.10), $\alpha = \lambda_{\min}(M_2)$, and $\beta = \lambda_{\max}(M_2)$, then $h_1(z)$ is convex and $h_2(z)$ is α-strongly convex and β-Lipschitz differentiable. As a result, [3, Prop. 26.16(ii)] gives

$$\|z_i^{k+1} - z_\star^{k+1}\| \le \tau^i \|z_0^{k+1} - z_\star^{k+1}\|, \quad \forall i = 0, 1, ..., p,$$

where $\tau = \sqrt{1 - \gamma(2\alpha - \gamma\beta^2)}$.

Let $a_i = \|z_i^{k+1} - z_\star^{k+1}\|$, then $a_i \le \tau^i a_0$. Therefore,

$$\begin{aligned}
\|\varepsilon^{k+1}\| &\le (\frac{1}{\gamma} + \beta)\|z_p^{k+1} - z_{p-1}^{k+1}\| && (3.11)\\
&\le (\frac{1}{\gamma} + \beta)(a_p + a_{p-1}) && (3.12)\\
&\le (\frac{1}{\gamma} + \beta)(\tau^p + \tau^{p-1})a_0. && (3.13)
\end{aligned}$$

On the other hand, we have

$$\begin{aligned}
\|z^{k+1} - z^k\| &\ge a_0 - a_p \\
&\ge (1 - \tau^p)a_0. && (3.14)
\end{aligned}$$

Combining (3.11) and (3.14) gives

$$\|\varepsilon^{k+1}\| \le c\|z^{k+1} - z^k\|,$$

where c is given in (3.9). ∎

Remark 3.5. Similarly to proximal gradient descent, one can show that finite iterations of FISTA with restart also satisfies the bounded relative error condition in Def. 3.3. The proof is omitted due to space limitation.

THEOREM 3.6. *Let Assumption 1 holds and g be block separable, i.e., $z = (z_1, z_2, ..., z_l)$ and $g(z) = \sum_{i=1}^l g_i(z_i)$. Let the stepsize of cyclic proximal BCD be γ, which is small enough such that*

$$\begin{aligned}
0 < \gamma \le \min\Bigg\{ &\frac{2\lambda_{\min}(M_2))}{\lambda_{\max}^2(M_2))}, \frac{1 - \sqrt{1 - \gamma(2\lambda_{\min}(M_2) - \gamma\lambda_{\max}^2(M_2))}}{4\sqrt{2}\gamma l \lambda_{\max}(M_2)}, \\
&\frac{1}{4l\lambda_{\max}(M_2)}, \frac{2l\lambda_{\max}(M_2)}{17l\lambda_{\max}(M_2) + 2(\frac{1 - \sqrt{1 - \gamma(2\lambda_{\min}(M_2) - \gamma\lambda_{\max}^2(M_2))}}{\gamma})^2} \Bigg\}.
\end{aligned}$$

Then, if $p \ge 1$ epochs of cyclic proximal BCD are applied to solve the $z-$subproblem in (3.3), and is initialized with the last iterate z^k, then the subproblem is solved with bounded relative error with

$$c = c(p) = \frac{(l\lambda_{\max}(M_2) + \frac{1}{\gamma})(\rho^p + \rho^{p-1})}{1 - \rho^p}, \tag{3.15}$$

where $\rho = 1 - \frac{\left(1 - \sqrt{1 - \gamma(2\lambda_{\min}(M_2) - \gamma\lambda_{\max}^2(M_2))}\right)^2}{2\gamma} < 1$.

Proof. See Appendix B. ∎

3.4. Convergence of inexact PPDHG. In this subsection, we proceed to establish the convergence of Algorithm 3.1. First, we transform Algorithm 3.1 into an ADMM-like algorithm in Proposition 3.8 below. Then, in Theorems 3.11 and 3.12 below, we prove the convergence of Algorithm 3.1 by using a generalized augmented Lagrangian of this ADMM-like algorithm.

First, let us show that PPDHG (3.3) is equivalent to an ADMM-like algorithm applied on the dual problem (1.2), this is similar to the equivalence of PDHG (3.2) and Linearized ADMM applied to the dual problem (1.2) shown in [14]. Therefore, we call this ADMM-like algorithm Preconditioned Linearized ADMM (PLADMM):

$$\begin{aligned} z^{k+1} &= \mathrm{Prox}_{g^*}^{M_2}(z^k + M_2^{-1}AM_1^{-1}(-A^Tz^k - y^k + u^k)), \\ y^{k+1} &= \mathrm{Prox}_{f^*}^{M_1^{-1}}(u^k - A^Tz^{k+1}), \\ u^{k+1} &= u^k - A^Tz^{k+1} - y^{k+1}. \end{aligned} \tag{3.16}$$

Remark 3.7. When $M_1 = \frac{1}{\tau}I, M_2 = \lambda I$, PLADMM (3.16) is Linearized ADMM, or Split Inexact Uzawa [38].

Furthermore, the Inexact PPDHG in Algorithm 3.1 is equivalent to Inexact PLADMM, which is summarized in Algorithm 3.2.

Let us also define the following generalized augmented Lagrangian for PLADMM:

$$L(z, y, u) = g^*(z) + f^*(y) + \langle -A^Tz - y, M_1^{-1}u\rangle + \frac{1}{2}\|A^Tz + y\|^2_{M_1^{-1}}. \tag{3.17}$$

Inspired by the framework of [36], this generalized augmented Lagrangian will serve as a Lyapunov function to establish convergence of Algorithm 3.2 and 3.1.

Algorithm 3.2 Inexact preconditioned linearized ADMM

Input: $f^* : \mathbb{R}^{\mathrm{n}} \to \mathbb{R}, g^* : \mathbb{R}^{\mathrm{m}} \to \mathbb{R}$, $A \in \mathbb{R}^{m\times n}$, preconditioners M_1 and M_2, initial vector (z_0, y_0, u_0), subproblem solver S for the $z-$subproblem in (3.16), number of inner loops p, number of outer iterations K.

Output: (z^K, y^K, u^K)

for $k \leftarrow 0, 1, ..., K-1$ **do**
  $z_0^{k+1} = z^k$;
  for $i \leftarrow 0, 1, ..., p-1$ **do**
    $z_{i+1}^{k+1} = S(z_i^{k+1}, y^k, u^k)$;
  end for
  $z^{k+1} = z_p^{k+1}$; ▷ approximate $\mathrm{Prox}_{g^*}^{M_2}(z^k + M_2^{-1}AM_1^{-1}(-A^Tz^k - y^k + u^k))$.
  $y^{k+1} = \mathrm{Prox}_{f^*}^{M_1^{-1}}(u^k - A^Tz^{k+1})$;
  $u^{k+1} = u^k - A^Tz^{k+1} - y^{k+1}$;
end for

PROPOSITION 3.8. *Under Assumption 1 and the transforms $u^k = M_1x^k$, $y^{k+1} = u^k - A^Tz^k - u^{k+1}$, PPDHG (3.3) is equivalent to PLADMM (3.16), and the Inexact PPDHG in Algorithm 3.1 is equivalent to the Inexact PLADMM in Algorithm 3.2.*

Proof. First, let us transform PPDHG in (3.3) to PLADMM (3.16).

Set $u^k = M_1x^k$, $y^{k+1} = u^k - A^Tz^k - u^{k+1}$, then (2.2) and (3.3) gives

$$y^{k+1} = M_1x^k - A^Tz^k - M_1x^{k+1} = \mathrm{Prox}_{f^*}^{M_1^{-1}}(u^k - A^Tz^k),$$

and we also have

$$
\begin{aligned}
u^{k+1} &= u^k - A^T z^k - y^{k+1},\\
z^{k+1} &= \mathrm{Prox}_{g^*}^{M_2}(z^k + M_2^{-1} A M_1^{-1}(-A^T z^k - y^{k+1} + u^{k+1})).
\end{aligned}
$$

If z-update is performed first, then we arrive at PLADMM (3.16).

Notice that for the inexact PPDHG in Algorithm 3.1, we are solving the $z-$subproblem of PPDHG (3.3) with bounded relative error as in Definition 3.3, therefore we are essentially doing the same to the $z-$subproblem of PLADMM (3.16), which gives Algorithm 3.2. ∎

In order to establish convergence of Algorithm 3.1, we also need the following Assumption 2, in addition to Assumption 1.

ASSUMPTION 2.

1. *$f(x)$ is μ_f-strongly convex.*
2. *$g^*(z) + f^*(-A^T z)$ is coercive, i.e.,*

$$
\lim_{\|z\|\to\infty} g^*(z) + f^*(-A^T z) = \infty.
$$

3. *$g^*(z)$ is a KŁ function.*

THEOREM 3.9. *Under Assumptions 1 and 2. Choose the preconditioners M_1, M_2 and the number of inner loops p in Algorithm 3.1 such that*

$$
\begin{aligned}
C_1 &= \frac{1}{2} M_1^{-1} - \frac{\|M_1\|}{\mu_f^2} I \succ 0,\\
C_2 &= M_2 - \frac{1}{2} A M_1^{-1} A^T - c(p) I \succ 0,
\end{aligned}
$$

where $c(p)$ is related to the $z-$subproblem solver S and M_2 (see, e.g., (3.9) and (3.15)). Define $L^k := L(z^k, y^k, u^k)$, then the inexact PLADMM in Algorithm 3.2 satisfies the following sufficient descent and lower boundedness properties:

$$
L^k - L^{k+1} \geq \|y^k - y^{k+1}\|_{C_1}^2 + \|z^k - z^{k+1}\|_{C_2}^2, \tag{3.18}
$$

$$
L^k \geq g^*(z^\star) + f^*(-A^T z^\star) > -\infty. \tag{3.19}
$$

Proof. Since the $z-$subproblem of Algorithm 3.2 is solved with bounded relative error in Def. 3.3, we have

$$
\mathbf{0} \in \partial g^*(z^{k+1}) + M_2(z^{k+1} - z^k - M_2^{-1} A M_1^{-1}(-A^T z^k - y^k + u^k)) + \varepsilon^{k+1}, \tag{3.20}
$$

where ε^{k+1} satisfies (3.8):

$$
\|\varepsilon^{k+1}\| \leq c(p)\|z^{k+1} - z^k\|. \tag{3.21}
$$

The y and u updates gives

$$
\mathbf{0} = \nabla f^*(y^{k+1}) + M_1^{-1}(y^{k+1} - u^k + A^T z^{k+1}) = \nabla f^*(y^{k+1}) - M_1^{-1} u^{k+1}, \tag{3.22}
$$

$$
u^{k+1} = u^k - A^T z^{k+1} - y^{k+1}. \tag{3.23}
$$

In order to show (3.18), let us write

$$g^*(z^k) \geq g^*(z^{k+1}) + \langle M_2(z^k - z^{k+1}) + AM_1^{-1}(-A^T z^k - y^k + u^k) - \varepsilon^{k+1}, z^k - z^{k+1}\rangle,$$

$$f^*(y^k) \geq f^*(y^{k+1}) + \langle M_1^{-1} u^{k+1}, y^k - y^{k+1}\rangle,$$

Assembling these inequalities with (3.21) gives us

$$\begin{aligned}
L^k - L^{k+1} \geq\; & \|z^k - z^{k+1}\|^2_{M_2 - c(p)I} \\
& + \langle AM_1^{-1}(-A^T z^k - y^k + u^k), z^k - z^{k+1}\rangle + \langle M_1^{-1} u^{k+1}, y^k - y^{k+1}\rangle \\
& + \langle -A^T z^k - y^k, M_1^{-1} u^k\rangle - \langle A^T z^{k+1} - y^{k+1}, M_1^{-1}(u^k - A^T z^{k+1} - y^{k+1})\rangle \\
& + \frac{1}{2}\|A^T z^k + y^k\|^2_{M_1^{-1}} - \frac{1}{2}\|A^T z^{k+1} + y^{k+1}\|^2_{M_1^{-1}} \\
=\; & \|z^k - z^{k+1}\|^2_{M_2 - c(p)I} \\
\text{(A)} \quad & + \langle AM_1^{-1}(-A^T z^k - y^k), z^k - z^{k+1}\rangle + \langle M_1^{-1} u^{k+1}, y^k - y^{k+1}\rangle \\
\text{(B)} \quad & + \langle -y^k, M_1^{-1} u^k\rangle - \langle -y^{k+1}, M_1^{-1} u^k\rangle \\
& + \frac{1}{2}\|A^T z^k + y^k\|^2_{M_1^{-1}} - \frac{3}{2}\|A^T z^{k+1} + y^{k+1}\|^2_{M_1^{-1}},
\end{aligned}$$

where the terms in (A) and (B) simplify to

$$\langle AM_1^{-1}(-A^T z^k - y^k), z^k - z^{k+1}\rangle + \langle M_1^{-1}(-A^T z^{k+1} - y^{k+1}), y^k - y^{k+1}\rangle. \tag{3.24}$$

Now we will use the following cosine rule on the two inner products above:

$$\langle a - b, a - c\rangle_{M_1^{-1}} = \frac{1}{2}\|a - b\|^2_{M_1^{-1}} + \frac{1}{2}\|a - c\|^2_{M_1^{-1}} - \frac{1}{2}\|b - c\|_{M_1^{-1}}.$$

Set $a = A^T z^k, c = A^T z^{k+1}$, and $b = -y^k$ to obtain

$$\begin{aligned}
\langle AM_1^{-1}(-A^T z^k - y^k), z^k - z^{k+1}\rangle = & -\frac{1}{2}\|A^T z^k + y^k\|^2_{M_1^{-1}} - \frac{1}{2}\|A^T z^k - A^T z^{k+1}\|^2_{M_1^{-1}} \\
& + \frac{1}{2}\|y^k + A^T z^{k+1}\|^2_{M_1^{-1}}.
\end{aligned} \tag{3.25}$$

Set $a = y^{k+1}, c = y^k$, and $b = -A^T z^{k+1}$ to obtain

$$\begin{aligned}
\langle M_1^{-1}(-A^T z^{k+1} - y^{k+1}), y^k - y^{k+1}\rangle = & \frac{1}{2}\|A^T z^{k+1} + y^{k+1}\|^2_{M_1^{-1}} + \frac{1}{2}\|y^k - y^{k+1}\|_{M_1^{-1}} \\
& - \frac{1}{2}\|A^T z^{k+1} + y^k\|^2_{M_1^{-1}}.
\end{aligned} \tag{3.26}$$

Combining (3.24), (3.25), and (3.27) yields

$$\begin{aligned}
L^k - L^{k+1} \geq & \|z^k - z^{k+1}\|^2_{M_2 - \frac{1}{2}AM_1^{-1}A^T - c(p)I} + \|y^k - y^{k+1}\|^2_{\frac{1}{2}M_1^{-1}} \\
& - \|A^T z^{k+1} + y^{k+1}\|^2_{M_1^{-1}}.
\end{aligned} \tag{3.27}$$

Since f is μ_f-strongly convex, we know that ∇f^* is $\frac{1}{\mu_f}$−Lipschitz continuous. Consequently,

$$\begin{aligned}
\|A^T z^{k+1} + y^{k+1}\|^2_{M_1^{-1}} = \|u^k - u^{k+1}\|^2_{M_1^{-1}} & \leq \frac{1}{\lambda_{\min}(M_1^{-1})}\|M_1^{-1}(u^k - u^{k+1})\|^2 \\
& \overset{(3.22)}{\leq} \frac{\|M_1\|}{\mu_f^2}\|y^k - y^{k+1}\|^2.
\end{aligned} \tag{3.28}$$

Combining (3.27) and (3.28) gives (3.18).

Now, to show (3.19), we use (3.22) to get

$$f^*(y^k) \geq f^*(-A^T z^k) + \langle M_1^{-1} u^k, y^k + A^T z^k \rangle,$$

And, thus,

$$\begin{aligned} L^k &= g^*(z^k) + f^*(y^k) + \langle -A^T z^k - y^k, M_1^{-1} u^k \rangle + \frac{1}{2}\|A^T z^k + y^k\|^2_{M_1^{-1}} \\ &\geq g^*(z^k) + f^*(-A^T z^k) + \frac{1}{2}\|A^T z^k + y^k\|^2_{M_1^{-1}}, \end{aligned} \tag{3.29}$$

and finally (3.19). ∎

Remark 3.10. In order for $C_2 \succ 0$, we can set $M_2 = AM_1^{-1}A^T$ as suggested by subsection 3.2, since $c(p) \propto \alpha^p$ for some $0 < \alpha < 1$ in (3.9) and (3.15), we know that there exists $p_0 \geq 1$ such that $C_2 \succ 0$ for any $p \geq p_0$. In our numerical experiments, Algorithm 3.1 always converges for $p \geq 1$.

We conclude this section by showing the convergence of (x^k, z^k) in Algorithm 3.1 to a primal-dual solution pair of (1.1) and (1.2).

THEOREM 3.11. *Let the assumptions in Theorem 3.9 hold. Then, (x^k, z^k) in Algorithm 3.1 is bounded, and any cluster point of $\{x^k, z^k\}$ is a primal-dual solution pair of* (1.1) *and* (1.2)*.*

Proof. According to Theorem 3.8, We just need to show that $\{M_1^{-1}u^k, z^k\}$ is bounded and its cluster points are primal-dual solution pairs of (1.1).

Since L^k is noincreasing, (3.29) tells us that

$$g^*(z^k) + f^*(-A^T z^k) + \frac{1}{2}\|A^T z^k + y^k\|^2_{M_1^{-1}} \leq L^0 < +\infty.$$

Since $g^*(z) + f^*(-A^T z)$ is coercive, we get that $\{z^k\}$ is bounded, and from the boundedness of $\{A^T z^k + y^k\}$, the boundedness of $\{y^k\}$. Furthermore, (3.22) gives us

$$\|M_1^{-1}(u^k - u^0)\| \leq \frac{1}{\mu_f}\|y^k - y^0\|.$$

Therefore, $\{M_1^{-1}u^k\}$ is also bounded.

Suppose (z^c, y^c, u^c) is a cluster point of $\{z^k, y^k, u^k\}$. Let us show that (z^c, y^c, u^c) is saddle point of $L(z, y, u)$, i.e.,

$$\mathbf{0} \in \partial L(z^c, y^c, u^c), \tag{3.30}$$

or equivalently,

$$\begin{aligned} \mathbf{0} &\in \partial g^*(z^c) - AM_1^{-1}u^c, \\ \mathbf{0} &= \nabla f^*(y^c) - M_1^{-1}u^c, \\ \mathbf{0} &= A^T z^c + y^c, \end{aligned}$$

which ensures $(M_1^{-1}u^c, z^c)$ as a primal-dual solution pair of (1.1).

In order to show (3.30), we first notice that (3.17) gives

$$\begin{aligned} \partial_x L(z^{k+1}, y^{k+1}, u^{k+1}) &= \partial g^*(z^{k+1}) - AM_1^{-1}u^{k+1} + AM_1^{-1}(A^T z^{k+1} + y^{k+1}), \\ \nabla_y L(z^{k+1}, y^{k+1}, u^{k+1}) &= \nabla f^*(y^{k+1}) - M_1^{-1}u^{k+1} + M_1^{-1}(A^T z^{k+1} + y^{k+1}), \\ \nabla_u L(z^{k+1}, y^{k+1}, u^{k+1}) &= M_1^{-1}(-A^T z^{k+1} - y^{k+1}). \end{aligned}$$

Compare these with the optimality conditions (3.20), (3.22), and (3.23), we have

$$d^{k+1} = (d_z^{k+1}, d_y^{k+1}, d_u^{k+1}) \in \partial L(z^{k+1}, y^{k+1}, u^{k+1}),$$

where

$$\begin{aligned}
d_z^{k+1} &= M_2(z^k - z^{k+1}) + 2AM_1^{-1}(u^k - u^{k+1}) - AM_1^{-1}(u^{k-1} - u^k) - \varepsilon^{k+1},\\
d_y^{k+1} &= M_1^{-1}(u^k - u^{k+1}),\\
d_u^{k+1} &= M_1^{-1}(u^{k+1} - u^k).
\end{aligned}$$

Since (3.18), and (3.19) implies $z^k - z^{k+1}, y^k - y^{k+1} \to \mathbf{0}$, (3.22) gives $u^k - u^{k+1} \to \mathbf{0}$. Combine these with (3.8), we have $d^k \to \mathbf{0}$.

Finally, let us take a subsequence $\{z^{k_s}, y^{k_s}, u^{k_s}\} \to (z^c, y^c, u^c)$, since $d^{k_s} \to \mathbf{0}$ as $s \to +\infty$, [33, Def. 8.3] and [33, Prop. 8.12] yield (3.30), which tells us that $(M_1^{-1}u^c, z^c)$ is a primal-dual solution pair of (1.1). ∎

Also, we can show that (x^k, z^k) in Algorithm 3.1 actually converges. Since the proof consists of a standard technique of using the KŁ property in Assumption 2, which is not very relevant to the main idea of this subsection, we leave it to Appendix C.

THEOREM 3.12. *Let the assumptions in Theorem 3.9 hold, then the $\{x^k, z^k\}$ in Algorithm 3.1 converges to a primal-dual solution pair of* (1.1).

Proof. See Appendix C. ∎

4. Numerical experiments.

In this section, we compare our inexact preconditioned PDHG in algorithm 3.1 with PDHG (3.2) and PDHG with diagonal preconditioning [29]. We consider three popular applications of PDHG: TV-L^1 denoising, graph cuts, and estimation of earth mover's distance. Although they do not satisfy all the assumptions in our theory, we still observe significant speedup compared to other algorithms.

When we write these examples in the form of (1.1), A is one of the following:

Case 1: The 2D discrete gradient operator $D : \mathbb{R}^{M\times N} \to \mathbb{R}^{2M\times N}$:

Let the images be of size $M \times N$, and h be the length of discretization interval, then

$$(Du)_{i,j} = \begin{pmatrix} (Du)^1_{i,j} \\ (Du)^2_{i,j} \end{pmatrix},$$

where

$$(Du)^1_{i,j} = \begin{cases} \frac{1}{h}(u_{i+1,j} - u_{i,j}) & \text{if } i < M,\\ 0 & \text{if } i = M, \end{cases}$$

$$(Du)^2_{i,j} = \begin{cases} \frac{1}{h}(u_{i,j+1} - u_{i,j}) & \text{if } j < N,\\ 0 & \text{if } j = N. \end{cases}$$

Case 2: The weighted gradient operator $D_w : \mathbb{R}^{M\times N} \to \mathbb{R}^{2M\times N}$:

$$D_w = \operatorname{diag}(w)D,$$

where $w \in (\mathbb{R}^+)^{2MN}$ is a weight vector.

Case 3: The 2D discrete divergence operator div: $\mathbb{R}^{2M\times N} \to \mathbb{R}^{M\times N}$:

$$\operatorname{div}(p)_{i,j} = h(p^1_{i,j} - p^1_{i-1,j} + p^2_{i,j} - p^2_{i,j-1}), \tag{4.1}$$

where $p = (p^1, p^2)^T \in \mathbb{R}^{2M\times N}$, $p^1_{0,j} = p^1_{M,j} = 0$ and $p^2_{i,0} = p^2_{i,N} = 0$ for $i = 1, ..., M$, $j = 1, ..., N$.

In view of the special structures of these operators, we choose cyclic proximal Block Coordinate Descent (BCD) as the z-subproblem solver in Algorithm 3.1. In particular, we split $\{1, 2, ...m\}$ into 2 blocks (in case 3) or 4 blocks (in cases 1 and 2) according to Claims 4.1 and 4.2, which are inspired by the popular red-black ordering [34] for solving sparse linear system.

According to Theorem 3.6, finite inner loops of cyclic proximal BCD satisfy the bounded relative error condition in Def.3.3, and we can expect that this solver brings faster overall convergence. The intuition is that when g^* is linear (or equivalently, g is a δ function), the $z-$subproblem in Alg.3.1 reduces to a linear system with a structured sparse matrix AA^T. As a result, Gradient Descent amounts to Richardson method [32, 34] and cyclic BCD becomes Gauss-Seidel method [16, 34]. In view of the special structures of A, Gauss-Seidel is faster (see Chapter 4 of [34]). Therefore, we can anticipate a faster convergence when cyclic proximal BCD is used.

Furthermore, the following two claims tell us that under special block designs inspired by red-black ordering, the two subproblems have closed-form solutions, which are easy to implement and compute. Furthermore, updates within each block can be implemented in parallel.

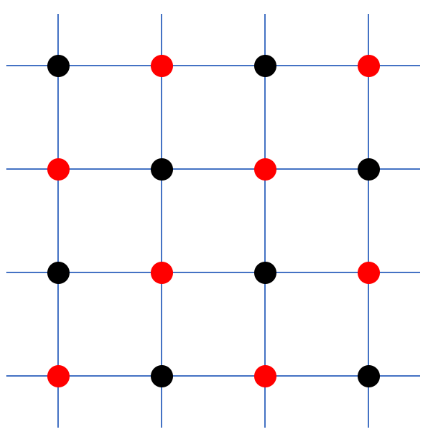

FIG. 1. *two-block ordering in Claim 4.1*

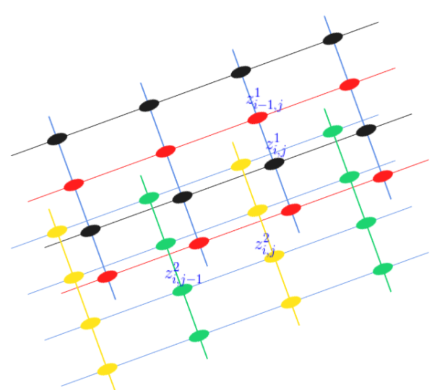

FIG. 2. *four-block ordering in Claim 4.2*

CLAIM 4.1. *When $A = \operatorname{div}$ (i.e. $A^T = -D$) and $M_2 = \tau AA^T$, for $z \in \mathbb{R}^{M\times N}$, we separate z into two block z_b, z_r where*

$$z_b := \{z_{i,j} \,|\, i + j \text{ is even}\}, \;\; z_r := \{z_{i,j} \,|\, i + j \text{ is odd}\},$$

for $1 \le i \le M$, $1 \le j \le N$. If $g(z) = \Sigma_{i,j} g_{i,j}(z_{i,j})$ and $\mathrm{prox}_{\lambda g^*_{i,j}}$ have closed-form solutions for all $1 \le i \le M$, $1 \le j \le N$ and $\lambda > 0$, then BCD subproblems on the $z-$subproblem of Algorithm 3.1 have closed-form solutions, and updates within each block can be implemented in parallel.

Proof. As illustrated in Fig. 1, on the $z-$subproblem, the update of every black node depends only on its neighbor red nodes, thus all the black nodes can be updated in parallel and with closed-form solutions. The same argument applies to the red nodes. See Appendix D for a complete explanation. ∎

CLAIM 4.2. *When $A = D$ or $A = D_w$ (i.e. $A^T = -\text{div}$ or $A^T = -\text{div}\,\text{diag}(\text{w})$) and $M_2 = \tau AA^T$, for $z = (z^1, z^2)^T \in \mathbb{R}^{2M\times N}$, we separate z into four blocks z_b, z_r, z_y and z_g, where*

$$z_b = \{z^1_{i,j} \,|\, i \text{ is odd}\}, \quad z_r = \{z^1_{i,j} \,|\, i \text{ is even}\},$$
$$z_y = \{z^2_{i,j} \,|\, j \text{ is odd}\}, \quad z_g = \{z^2_{i,j} \,|\, j \text{ is even}\},$$

for $1 \le i \le M$, $1 \le j \le N$. If $g(z) = \Sigma_{i,j} g_{i,j}(z_{i,j})$ and $prox_{\lambda g^*_{i,j}}$ have closed-form solutions for all $1 \le i \le M$, $1 \le j \le N$ and $\lambda > 0$, then BCD subproblems on the $z-$subproblem of Algorithm 3.1 have closed-form solutions, and updates within each block can be implemented in parallel.

Proof. In Figure 2, the 4 blocks are in 4 different colors. Nodes with the same color can be updated in parallel with closed-form solutions, as within one color nodes are independent with each other during the updates. See Appendix D for details. ∎

In Table 1, Table 2 and Figure 7, PDHG denotes the PDHG in (3.2); DP-PDHG denotes the diagonal preconditioned PDHG in [29], PPDHG denotes PPDHG in (3.3) where the $(k+1)$th z-subproblem is solved until $\frac{\|z^k - z^{k+1}\|_2}{\max\{1, \|z^{k+1}\|_2\}} < 10^{-5}$ using the TFOCS [4] implementation of FISTA with restart; Alg. 3.1, BCD denotes our inexact preconditioned PDHG in Algorithm 3.1, where the inner loop solver S is cyclic proximal BCD. Except for DP-PDHG, only the best runtime over certain choices of parameters is presented.

Comparision of PDHG and DP-PDHG have already been presented for TV-L^1 denoising and graph cuts in [29], and PDHG is proposed to estimate the earth mover's distance in [22]. In order to provide a direct comparision, we use their problem formulations.

4.1. Total variation based image denoising. The (discrete) TV-L^1 model for image denoising can be expressed as

$$\text{minimize} \quad \Phi(u) = \|Du\|_1 + \lambda\|u - f\|_1,$$

where D is the 2D diecrete gradient operator with $h = 1$, $u \in \mathbb{R}^{M\times N}$ is the sought solution, $f \in \mathbb{R}^{M\times N}$ is a noisy input image, and λ is a regularization parameter. In our experiment we input a 1024×1024 image with noise level 0.15 and set $\lambda = 1$, see Fig. 3. We run the algorithms until $\delta^k := \frac{|\Phi^k - \Phi^\star|}{|\Phi^\star|} < 10^{-6}$, where Φ^k is the objective value at kth iteration and Φ^* is the optimal objective value obtained by calling CVX.

Our Numerical results on TV-L^1 model are summarized in Table 1, where the best results for $\tau \in \{10, 1, 0.1, 0.01, 0.001\}$ and $p \in \{1, 2, 3\}$ are presented. Our Algorithm 3.1 is significantly faster than the other three algorithms.

Remarkably, our algorithm's number of outer iterations is less than that of PPDHG with the stopping criterion $\frac{\|z^k - z^{k+1}\|_2}{\max\{1, \|z^{k+1}\|_2\}} < 10^{-5}$, as this kind of stopping criteria may become looser as z^k is closer to $z^\star$. In this example, $\frac{\|z^k - z^{k+1}\|_2}{\max\{1, \|z^{k+1}\|_2\}} < 10^{-5}$ only requires 1 inner iteration of FISTA when Outer Iter ≥ 368, while as high as 228 inner loops on average during the first 100 outer iterations. In comparison, our algorithm achieves both less outer iterations and cheaper cost per outer iteration.

In addition, the diagonal preconditioner designed in [29] has little effects when $A = D$. In fact, $M_1 = \mathrm{diag}(\Sigma_i |A_{i,j}|)$ will be $4I_n$ and $M_2 = \mathrm{diag}(\Sigma_j |A_{i,j}|)$ will be $2I_m$ if we ignore the Neumann boundary condition. With these fixed and almost the same parameters, DP-PDHG performs even worse than PDHG.

Method	Parameters	Outer Iter	Runtime(s)
PDHG	$\tau = 0.01, M_1 = \tau^{-1} I_n, M_2 = \tau \|D\|^2 I_m$	2990	114.2576
DP-PDHG	$M_1 = \mathrm{diag}(\Sigma_i \|D_{i,j}\|), M_2 = \mathrm{diag}(\Sigma_j \|D_{i,j}\|)$	8856	329.7890
PPDHG (3.3)	$\tau = 0.1, M_1 = \tau^{-1} I_n, M_2 = \tau DD^T$	963	5.9777×10^3
Alg. 3.1, BCD	$\tau = 0.01, M_1 = \tau^{-1} I_n, M_2 = \tau DD^T$, $p = 1$	**541**	**26.2704**

TABLE 1
TV-L^1 denoising.

FIG. 3. *Noisy image*

FIG. 4. *Denoised image (Alg. 3.1, BCD)*

4.2. Graph cuts. The total-variation-based graph cut model is to minimize the follow weighted TV energy:

$$\begin{array}{ll} \text{minimize} & \|D_w u\|_1 + \langle u, \omega^u \rangle \\ \text{subject to} & 0 \le u \le 1, \end{array}$$

where $w^u \in \mathbb{R}^{M \times N}$ is a vector of unary weights, $w^b \in \mathbb{R}^{2MN}$ is a vector of binary weights, and $D_w = \mathrm{diag}(w^b)D$, where D is the 2D diecrete gradient operator with $h = 1$. Specifically, $w^u_{i,j} = \alpha(\|I_{i,j} - \mu_f\|^2 - \|I_{i,j} - \mu_b\|^2)$, $w^{b,1}_{i,j} = \exp(-\beta|I_{i+1,j} - I_{i,j}|)$ and $w^{b,2}_{i,j} = \exp(-\beta|I_{i,j+1} - I_{i,j}|)$. In our experiment the image is of the size 660×720, and we set $\alpha = 1/2$, $\beta = 10$, $\mu_f = [0; 0; 1]$ (for the blue foreground) and $\mu_b = [0; 1; 0]$ (for the green background). We run all algorithms until $\delta^k := \frac{|\Phi^k - \Phi^\star|}{|\Phi^\star|} < 10^{-8}$, where Φ^k is the objective value at kth iteration and Φ^* is the optimal objective value obtained by calling CVX.

The best results of $\tau \in \{10, 1, 0.1, 0.01, 0.001\}$ and $p \in \{1, 2, 3\}$ are summarized in Table 2, where we can see that our algorithm yields the best performance on runtime. Also, our algorithm's number of outer iterations is close to that of PPDHG.

Method	Parameters	Outer Iter	Runtime(s)
PDHG	$\tau = 1, M_1 = \tau^{-1} I_n, M_2 = \tau \|D_w\|^2 I_m$	5529	140.5777
DP-PDHG	$M_1 = \text{diag}(\Sigma_i \lvert D_{wi,j}\rvert), M_2 = \text{diag}(\Sigma_j \lvert D_{wi,j}\rvert)$	3572	108.3573
PPDHG (3.3)	$\tau = 10, M_1 = \tau^{-1} I_n, M_2 = \tau D_w D_w^T$	**282**	938.3787
Alg. 3.1, BCD	$\tau = 10, M_1 = \tau^{-1} I_n, M_2 = \tau D_w D_w^T$, $p = 2$	**411**	**14.9663**

Table 2
Graph cuts

Fig. 5. *Input image*

Fig. 6. *Graph cut (Alg. 3.1, BCD)*

4.3. Estimation of earth mover's distance. We consider the estimation of earth mover's distance, which is a popular model in image processing, computer vision and statistics [20, 25, 28]. From [22] we know that the problem can be formulated as

$$\begin{array}{ll} \text{minimize} & \|m\|_{1,2} \\ \text{subject to} & \text{div}(m) + \rho^1 - \rho^0 = 0, \end{array} \tag{4.2}$$

where $m \in \mathbb{R}^{2M \times N}$ is the sought flux vector on the $M \times N$ grid, and ρ^0, ρ^1 represents two mass distributions on the $M \times N$ grid. The setting in our experiment here is the same with that in [22], i.e. $M = N = 256$, $h = \frac{N-1}{4}$, and for ρ^0 and ρ^1 see Fig. 8.

Since the iterates m^k may not satisfy the linear constraint, the objective $\Phi(m) = I_{\{m|div(m)=\rho^0-\rho^1\}} + \|m\|_{1,2}$ is not comparable. Instead, we compare $\|m^k\|_{1,2}$ and the constraint violation until $k = 100000$ outer iterations in Fig. 7, where we set $\tau = 3 \times 10^{-6}$ as in [22], and $\sigma = \frac{1}{\tau \|\text{div}\|^2}$. In Fig. 7, we can see that our algorithm provides much lower constraint violation as well as much better estimation for the earth mover's distance $\|m\|_{1,2}$. Fig. 8 shows the solution obtained by Alg. 3.1, where m is the flux that moves the standing cat ρ^1 into the crouching cat ρ^0. DP-PDHG and PPDHG are extremely slow in this example. Similar to 4.1, when $A = \text{div}$, the diagonal preconditioners proposed in [29] are approximately equivalent to fixed constant parameters $\tau = \frac{1}{2h}$, $\sigma = \frac{1}{4h}$ and they lead to extremely slow convergence. As for PPDHG, it suffers from the expensive cost per outer iteration as in the previous two experiments.

It is worth mentioning that unlike [22], the algorithms in our experiments are not implemented in a parallel fashion. On the other hand, in our Algorithm 3.1 with cyclic proximal BCD as the inner loop solver, coordinates in each block in the block designs of Fig. 1 and 2 can be updated in parallel. Therefore, one can expect a further speed up by a parallel implementation.

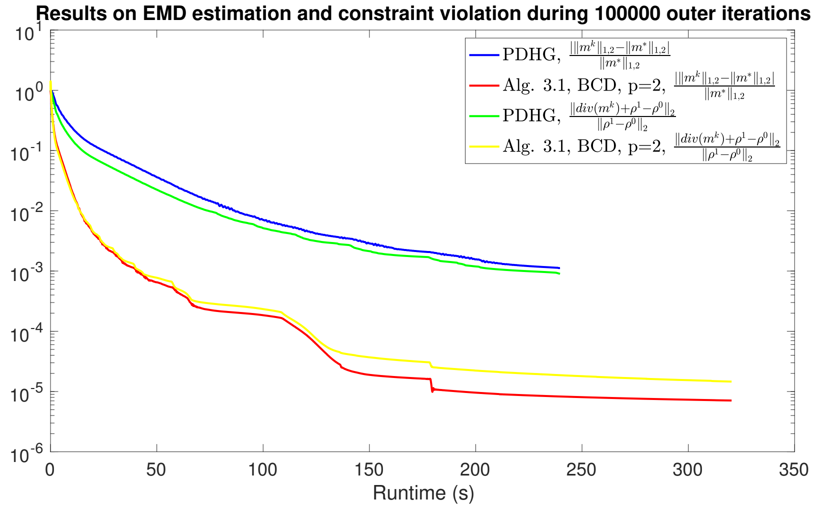

Fig. 7. *For PDHG,* $\tau = 3 \times 10^{-6}$*,* $\sigma = \frac{1}{\tau\|\mathrm{div}\|^2}$*; For Alg 3.1, BCD,* $\tau = 3 \times 10^{-6}$*,* $M_1 = \tau^{-1} I_n$*,* $M_2 = \tau \mathrm{div}\mathrm{div}^{\mathrm{T}}$*,* $p = 2$*.* $\|m^*\|\|_{1,2} = 0.6718$ *is given by gurobi of CVX.*

Fig. 8. ρ^0*,* ρ^1 *are the white standing cat, and the black crouching cat, respectively. The images are of the size* 256×256*, and the earth mover's distance between* ρ^0 *and* ρ^1 *is 0.6718.*

5. Concluding Remarks. In this paper, We provide an algorithmic framework for apply preconditioning and fast subproblem solvers on PDHG and ADMM with convergence guarantees. Remarkably, we allow a fixed number of inner iterations for one of the subproblems. Although the examples in our numerical experiments do not satisfy all the assumptions, significant accelerations in both outer iteration and runtime are observed when proper preconditioners and subproblem solvers are applied.

There are still some interesting questions, which need to addressed in the future: (a) According to Theorem 3.2, there may be better preconditioners than $M_1 = \frac{1}{\tau}I_{n\times n}, M_2 = \tau AA^T$, which lead to ADMM iterations. (b) It is possible that convergence of Algorithm 3.1 can also be established for even faster acclerated subproblem solvers like APCG [23], NU_ACDM [1], and A2BCD [19]. (c) It is possible that a broad class of algorithms can be accelerated by integrating preconditioning, fixed number of inner loops, and suitable subproblem solvers. We hope our framework can be applied on more algorithms with faster convergence guarantees.

Appendix A. ADMM as a special case of PPDHG.

In this section we show that if we choose $M_1 = \frac{1}{\tau}$ and $M_2 = \tau AA^T$ in PPDHG (3.3), then it is equivalent to ADMM on the primal problem (1.1).

By Theorem 1 of [37], we know that ADMM is primal-dual equivalent, in the sense that one can recover primal iterates from dual iterates and vice versa. Therefore, it suffices to show that $M_1 = \frac{1}{\tau}$ and $M_2 = \tau AA^T$ in PPDHG (3.3) on the primal problem is equivalent to ADMM on the dual problem (1.2).

In Theorem 3.8 we have shown that, under an appropriate change of variables, PPDHG on the primal is equivalent to PLADMM in (3.16) on the dual. As a result, we just need to demonstrate that PLADMM on the dual is exactly ADMM on the dual when $M_1 = \frac{1}{\tau}I_{n\times n}$ and $M_2 = \tau AA^T$.

For the $z-$update in (3.16), we have

$$
\begin{aligned}
z^{k+1} &= \arg\min_{z\in\mathbb{R}^m}\{g^*(z) - \tau\langle z - z^k, A(-A^Tz^k - y^k + u^k)\rangle + \frac{\tau}{2}\|z - z^k\|^2_{AA^T}\} \\
&= \arg\min_{z\in\mathbb{R}^m}\{g^*(z) - \tau\langle z - z^k, A(-y^k + u^k)\rangle + \frac{\tau}{2}\|z\|^2_{AA^T}\} \\
&= \arg\min_{z\in\mathbb{R}^m}\{g^*(z) + \tau\langle z, A(y^k - u^k)\rangle + \frac{\tau}{2}\|A^Tz\|^2\} \\
&= \arg\min_{z\in\mathbb{R}^m}\{g^*(z) + \tau\langle A^Tz, -u^k\rangle + \frac{\tau}{2}\|A^Tz + y^k\|^2\} \\
&= \arg\min_{z\in\mathbb{R}^m}\{g^*(z) + \tau\langle -A^Tz - y^k, u^k\rangle + \frac{\tau}{2}\|A^Tz + y^k\|^2\}. \qquad \text{(A.1)}
\end{aligned}
$$

and for the y-update we have

$$
\begin{aligned}
y^{k+1} &= \mathrm{Prox}_{f^*}^{M_1^{-1}}(u^k - A^Tz^{k+1}) \\
&= \arg\min_{y\in\mathbb{R}^n}\{f^*(y) + \frac{\tau}{2}\|y - u^k + A^Tz^{k+1}\|^2\} \\
&= \arg\min_{y\in\mathbb{R}^n}\{f^*(y) + \tau\langle -A^Tz^{k+1} - y, u^k\rangle + \frac{\tau}{2}\|A^Tz^{k+1} + y\|^2\}. \qquad \text{(A.2)}
\end{aligned}
$$

Define $v^k = \tau u^k$, (A.1), (A.2), and the $u-$update in (3.16) become

$$
\begin{aligned}
z^{k+1} &= \arg\min_{z\in\mathbb{R}^m}\{g^*(z) + \langle -A^Tz - y^k, v^k\rangle + \frac{\tau}{2}\|A^Tz + y^k\|^2\}, \\
y^{k+1} &= \arg\min_{y\in\mathbb{R}^n}\{f^*(y) + \langle -A^Tz^{k+1} - y, v^k\rangle + \frac{\tau}{2}\|A^Tz^{k+1} + y\|^2\}, \\
v^{k+1} &= v^k - \tau(A^Tz^{k+1} + y^{k+1}),
\end{aligned}
$$

which are ADMM iterations on the dual problem (1.2).

Appendix B. Proof of Theorem 3.6: Cyclic proximal BCD satisfies bounded relative error condition.

The $z-$subproblem in (3.3) is of the form

$$\min_{z\in\mathbb{R}^{\mathrm{m}}} h_1(z)+h_2(z),$$

where

$$h_1(z)=g^*(z)=\sum_{i=1}^{l} g_i^*(z_i),$$
$$h_2(z)=\frac{1}{2}\|z-z^k-M_2^{-1}A(2x^{k+1}-x^k)\|_{M_2}^2.$$

And z^{k+1} is given by

$$z_0^{k+1}=z^k,$$
$$z_{i+1}^{k+1}=S(z_i^{k+1},x^{k+1},x^k),\quad i=0,1,...,p-1,$$
$$z^{k+1}=z_p^{k+1}.$$

Here the inner loop solver S is cyclic proximal BCD.

Let us define

$$T(z)=\mathrm{Prox}_{\gamma g^*(z)}(z-\gamma\nabla h_2(z))),$$
$$B(z)=\frac{1}{\gamma}(z-T(z)),$$

and the ith coordinate operator of B:

$$B_i(z)=(0,...,(B(z))_i,...,0).$$

Then
$$z_{i+1}^{k+1}=S(z_i^{k+1},x^{k+1},x^k)=(I-\gamma B_l)(I-\gamma B_2)...(I-\gamma B_1)z_i^{k+1}.$$

By [3, Prop. 26.16(ii)], we know that $T(z)$ is a contraction with coefficient $\theta=\sqrt{1-\gamma(2\lambda_{\min}(M_2)-\gamma\lambda_{\max}^2(M_2))}$. Together with [3,], we know that for $\forall z_1,z_2\in\mathbb{R}^{\mathrm{m}}$ we have,

$$\langle B(z_1)-B(z_2),z_1-z_2\rangle=\frac{1}{\gamma}\|z_1-z_2\|^2-\frac{1}{\gamma}\langle T(z_1)-T(z_2),z_1-z_2\rangle$$
$$\geq\mu\|z_1-z_2\|^2,$$

where $\mu=\frac{1-\theta}{\gamma}$.

Let $z_\star^{k+1}=\arg\min_{z\in\mathbb{R}^{\mathrm{m}}}\{h_1(z)+h_2(z)\}$. By [9, Thm 3.5], we know that

$$\|z_i^{k+1}-z_\star^{k+1}\|\leq\rho^i\|z_0^{k+1}-z_\star^{k+1}\|,\quad\forall i=1,2,...,p. \tag{B.1}$$

where $\rho=1-\frac{\gamma\mu^2}{2}$.

Let $y_j = (I - \gamma B_j)...(I - \gamma B_1)z_{p-1}^{k+1}$ for $j = 1, ..., l$ and $y_0 = z_{p-1}^{k+1}$. Note that $(z_p^{k+1})_j = (y_j)_j$ for $j = 1, 2, ..., l$, and the blocks of y_j satisfies

$$(y_j)_t = \begin{cases} \Big(\operatorname{Prox}_{\gamma g^*}\big(y_{j-1} - \gamma \nabla h_2(y_{j-1})\big)\Big)_t, & \text{if } t = j \\ (y_{j-1})_t, & \text{otherwise.} \end{cases}$$

On the other hand, we have

$$\operatorname{Prox}_{\gamma g^*}\big(y_{j-1} - \gamma \nabla h_2(y_{j-1})\big) = \operatorname*{arg\,min}_{y \in \mathbb{R}^m} \{g^*(y) + \frac{1}{2\gamma}\|y - y_{j-1} + \gamma \nabla h_2(y_{j-1})\|^2\}.$$

Since g^* and $\|\cdot\|^2$ are separable, we obtain

$$\mathbf{0} \in \partial g_j^*((y_j)_j) + \frac{1}{\gamma}\Big((y_j)_j - (y_{j-1})_j + \gamma\big(\nabla h_2(y_{j-1})\big)_j\Big), \quad \forall j = 1, 2, ..., l,$$

or equivalently,

$$\mathbf{0} \in \partial g_j^*((z_p^{k+1})_j) + \frac{1}{\gamma}\Big((z_p^{k+1})_j - (z_{p-1}^{k+1})_j + \gamma\big(\nabla h_2(y_{j-1})\big)_j\Big), \quad \forall j = 1, 2, ..., l.$$

As a result,

$$\mathbf{0} \in \partial g^*(z_p^{k+1}) + \frac{1}{\gamma}\Big(z_p^{k+1} - z_{p-1}^{k+1} + \gamma \xi_p\Big), \quad \forall j = 1, 2, ..., l,$$

where $(\xi_p)_j = \big(\nabla h_2(y_{j-1})\big)_j$ for $j = 1, 2, ..., l$. Compare this with (3.7), we know that

$$\varepsilon^{k+1} = \xi_p - \nabla h_2(z_p^{k+1}) + \frac{1}{\gamma}(z_p^{k+1} - z_{p-1}^{k+1}).$$

Notice that the first $j-1$ blocks of y_{j-1} are the same with those of $y_l = z_p^{k+1}$, and the rest of the blocks are the same with those of $y_0 = z_{p-1}^{k+1}$, so we have

$$\begin{aligned} \|\varepsilon^{k+1}\| &\le \sum_{j=1}^{l} \lambda_{\max}(M_2)\|y_{j-1} - z_p^{k+1}\| + \frac{1}{\gamma}\|z_p^{k+1} - z_{p-1}^{k+1}\| \\ &\le l\lambda_{\max}(M_2)\|z_{p-1}^{k+1} - z_p^{k+1}\| + \frac{1}{\gamma}\|z_p^{k+1} - z_{p-1}^{k+1}\| \\ &\le (l\lambda_{\max}(M_2) + \frac{1}{\gamma})(\|z_p^{k+1} - z_\star^{k+1}\| + \|z_{p-1}^{k+1} - z_\star^{k+1}\|) \end{aligned}$$

Combine this with (B.1), we arrive at

$$\|\varepsilon^{k+1}\| \le (l\lambda_{\max}(M_2) + \frac{1}{\gamma})(\rho^p + \rho^{p-1})\|z_0^{k+1} - z_\star^{k+1}\|. \tag{B.2}$$

We also have

$$\begin{aligned} \|z^{k+1} - z^k\| &= \|z_p^{k+1} - z_0^{k+1}\| \\ &\ge \|z_0^{k+1} - z_\star^{k+1}\| - \|z_p^{k+1} - z_\star^{k+1}\| \\ &\ge (1 - \rho^p)\|z_0^{k+1} - z_\star^{k+1}\| \end{aligned}$$

Combine this with (B.2), we obtain

$$\|\varepsilon^{k+1}\| \leq \frac{(l\lambda_{\max}(M_2) + \frac{1}{\gamma})(\rho^p + \rho^{p-1})}{1-\rho^p}\|z^{k+1} - z^k\|.$$

Appendix C. Proof of Theorem 3.12: KŁ property gives sequence convergence.

According to Theorem 3.8, We just need to show that $\{M_1^{-1}u^k, z^k\}$ converges to a primal-dual solution pair of (1.1).

By Theorem 3.11, we can take $\{z^{k_s}, y^{k_s}, u^{k_s}\} \to (z^c, y^c, u^c)$. Note that $L(z^{k_s}, y^{k_s}, u^{k_s})$ is monotonic nonincreasing and lower bounded due to Theorem 3.9, which implies the convergence of $L(z^{k_s}, y^{k_s}, u^{k_s})$. Since L is lower semicontinuous, we have

$$L(z^c, y^c, u^c) \leq \lim_{s\to\infty} L(z^{k_s}, y^{k_s}, u^{k_s}). \tag{C.1}$$

Since the only potentially discontinuous terms in L is g^*, we have

$$\lim_{s\to\infty} L(z^{k_s}, y^{k_s}, u^{k_s}) - L(z^c, y^c, u^c) \leq \limsup_{s\to\infty} g^*(z^{k_s}) - g^*(z^c). \tag{C.2}$$

By (3.20), we know that

$$\begin{aligned} g^*(z^c) \geq\ & g^*(z^{k_s}) \\ & + \langle M_2(z^{k_s-1} - z^{k_s}) + AM_1^{-1}(-A^T z^{k_s-1} - y^{k_s-1} + u^{k_s-1}) - \varepsilon^{k_s}, z^c - z^{k_s}\rangle, \end{aligned}$$

By Theorem 3.9, we know that $z^{k_s-1} - z^{k_s} \to \mathbf{0}$. Since $z^{k_s} \to z^c$ and $\{z^k, y^k, u^k\}$ is bounded, we obtain

$$\limsup_{s\to\infty} g^*(z^{k_s}) - g^*(z^c) \leq 0.$$

Combine this with (C.1) and (C.2), we conclude that $\lim_{s\to\infty} L(z^{k_s}, y^{k_s}, u^{k_s}) = L(z^c, y^c, u^c)$.

Since g^* is a KŁ function, L is also KŁ. As a result, similar to Theorem 2.9 of [2], we can claim the convergence of $\{z^k, y^k, u^k\}$ to $\{z^c, y^c, u^c\}$.

Appendix D. Two-block ordering in Claim 4.1 and Four-block ordering in Claim 4.2.

According to (3.4), when $M_2 = \tau AA^T$, the $z-$subproblem of Algorithm 3.1 is

$$z^{k+1} = \operatorname*{arg\,min}_{z\in\mathbb{R}^m}\{g^*(z) - \langle z - z^k, A(2x^{k+1} - x^k)\rangle + \frac{\tau}{2}\|A^T(z - z^k)\|_2^2\}. \tag{D.1}$$

Let us prove Claim 4.1 first.

In claim 4.1, $A = \text{div} \in \mathbb{R}^{MN\times 2MN}$ and $z \in \mathbb{R}^{MN}$. Following the definition of the sets z_b and z_r in Claim 4.1, we separate the MN columns of $A^T = -D$ into two blocks L_b, L_r associated with z_b and z_r, respectively. Therefore, we have $A^T z = L_b z_b + L_r z_r$ for any $z \in \mathbb{R}^{MN}$.

By the red-black ordering in Fig. 1, different columns of L_b are orthogonal to each other, therefore, ${L_b}^T L_b$ is diagonal. Similarly, ${L_r}^T L_r$ is also diagonal.

Let b be the set of black nodes and r the set of red nodes, then we can rewrite (D.1) as

$$\begin{aligned} z^{k+1} = \operatorname*{arg\,min}_{z_b, z_r\in\mathbb{R}^{MN/2}} & \{g_b^*(z_b) + g_r^*(z_r) + \langle z_b + z_r, c^k\rangle \\ & + \frac{\tau}{2}\|L_b(z_b - z_b^k) + L_r(z_r - z_r^k)\|_2^2\}, \end{aligned} \tag{D.2}$$

where $g_b^*(z_b) = \sum_{(i,j)\in b} g_{i,j}^*(z_{i,j})$, $g_r^*(z_r) = \sum_{(i,j)\in r} g_{i,j}^*(z_{i,j})$, and $c^k = -A(2x^{k+1} - x^k)$.

The cyclic proximal BCD applied on black and red blocks is then

$$z_b^{k+\frac{t+1}{p}} = \operatorname{prox}^{\tau {L_b}^T L_b}_{g_b^*(\cdot)+\langle \cdot, \tau {L_b}^T L_r (z_r^{k+\frac{t}{p}} - z_r^k) + c_b^k\rangle}(z_b^{k+\frac{t}{p}}), \tag{D.3}$$

$$z_r^{k+\frac{t+1}{p}} = \operatorname{prox}^{\tau {L_r}^T L_r}_{g_r^*(\cdot)+\langle \cdot, \tau {L_r}^T L_b (z_b^{k+\frac{t+1}{p}} - z_b^k) + c_r^k\rangle}(z_r^{k+\frac{t}{p}}). \tag{D.4}$$

for $t = 0, 1, ..., p-1$, where p is the number of inner loops as in Algorithm 3.1.

These updates have closed-form solutions since $L_b^T L_b$ and $L_r^T L_r$ are diagonal, and all $prox_{\lambda g_{i,j}^*}$ have closed form solutions. Furthermore, updates within each block can be implemented in parallel.

The proof of Claim 4.2 follows in a similar way. When $A = D$ or $A = D_w$, we separate the columns of A^T into four blocks L_b, L_r, L_y, L_g associated with z_b, z_r, z_y ,z_g, respectively. Therefore, we have $A^T z = L_b z_b + L_r z_r + L_y z_y + L_g z_g$ for all $z \in \mathbb{R}^{2MN}$. Similarly, by the block design in Fig. 2, we know that cyclic proximal BCD iterations on the $z-$subproblem have closed-form solutions, and updates within each block can be implemented in a parallel fashion.

REFERENCES

[1] Z. Allen-Zhu, Z. Qu, P. Richtárik, and Y. Yuan, *Even faster accelerated coordinate descent using non-uniform sampling*, in International Conference on Machine Learning, 2016, pp. 1110–1119.

[2] 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*, Mathematical Programming, 137 (2013), pp. 91–129.

[3] H. H. Bauschke, P. L. Combettes, et al., *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, vol. 2011, Springer, 2017.

[4] S. R. Becker, E. J. Candès, and M. C. Grant, *Templates for convex cone problems with applications to sparse signal recovery*, Mathematical programming computation, 3 (2011), p. 165.

[5] K. Bredies and H. Sun, *Preconditioned douglas–rachford splitting methods for convex-concave saddle-point problems*, SIAM Journal on Numerical Analysis, 53 (2015), pp. 421–444.

[6] K. Bredies and H. Sun, *A proximal point analysis of the preconditioned alternating direction method of multipliers*, Journal of Optimization Theory and Applications, 173 (2017), pp. 878–907.

[7] A. Chambolle and T. Pock, *A first-order primal-dual algorithm for convex problems with applications to imaging*, Journal of mathematical imaging and vision, 40 (2011), pp. 120–145.

[8] A. Chambolle and T. Pock, *On the ergodic convergence rates of a first-order primal–dual algorithm*, Mathematical Programming, 159 (2016), pp. 253–287.

[9] Y. T. Chow, T. Wu, and W. Yin, *Cyclic coordinate-update algorithms for fixed-point problems: Analysis and applications*, SIAM Journal on Scientific Computing, 39 (2017), pp. A1280–A1300.

[10] P. L. Combettes and N. N. Reyes, *Moreau's decomposition in banach spaces*, Mathematical Programming, 139 (2013), pp. 103–114.

[11] J. Eckstein and D. P. Bertsekas, *On the douglas—rachford splitting method and the proximal point algorithm for maximal monotone operators*, Mathematical Programming, 55 (1992), pp. 293–318.

[12] J. Eckstein and W. Yao, *Relative-error approximate versions of douglas–rachford splitting and special cases of the admm*, Mathematical Programming, pp. 1–28.

[13] J. Eckstein and W. Yao, *Approximate admm algorithms derived from lagrangian splitting*, Computational Optimization and Applications, 68 (2017), pp. 363–405.

[14] E. Esser, X. Zhang, and T. F. Chan, *A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science*, SIAM Journal on Imaging Sciences, 3 (2010), pp. 1015–1046.

[15] D. Feijer and F. Paganini, *Stability of primal–dual gradient dynamics and applications to network optimization*, Automatica, 46 (2010), pp. 1974–1981.

[16] C. F. Gauss, *Werke (in German), 9, Göttingen: Köninglichen Gesellschaft der Wissenschaften*, 1903.

[17] P. Giselsson and S. Boyd, *Diagonal scaling in douglas-rachford splitting and admm*, in Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on, IEEE, 2014, pp. 5033–5039.

[18] P. Giselsson and S. Boyd, *Linear convergence and metric selection for douglas-rachford splitting and admm*, IEEE Transactions on Automatic Control, 62 (2017), pp. 532–544.

[19] R. Hannah, F. Feng, and W. Yin, *A2bcd: An asynchronous accelerated block coordinate descent algorithm with optimal complexity*, arXiv preprint arXiv:1803.05578, (2018).

[20] E. Levina and P. Bickel, *The earth mover? s distance is the mallows distance: Some insights from statistics*, in null, IEEE, 2001, p. 251.

[21] M. Li, L.-Z. Liao, and X. Yuan, *Inexact alternating direction methods of multipliers with logarithmic–quadratic proximal regularization*, Journal of Optimization Theory and Applications, 159 (2013), pp. 412–436.

[22] W. Li, E. K. Ryu, S. Osher, W. Yin, and W. Gangbo, *A parallel method for earth mover's distance*, UCLA Comput. Appl. Math. Pub.(CAM) Rep, (2017), pp. 17–12.

[23] Q. Lin, Z. Lu, and L. Xiao, *An accelerated proximal coordinate gradient method*, in Advances in Neural Information Processing Systems, 2014, pp. 3059–3067.

[24] Q. Lin, Z. Lu, and L. Xiao, *An accelerated randomized proximal coordinate gradient method and its application to regularized empirical risk minimization*, SIAM Journal on Optimization, 25 (2015), pp. 2244–2273.

[25] L. Métivier, R. Brossier, Q. Mérigot, E. Oudet, and J. Virieux, *Measuring the misfit between seismograms using an optimal transport distance: application to full waveform inversion*, Geophysical Supplements to the Monthly Notices of the Royal Astronomical Society, 205 (2016), pp. 345–377.

[26] M. K. Ng, F. Wang, and X. Yuan, *Inexact alternating direction methods for image recovery*, SIAM Journal on Scientific Computing, 33 (2011), pp. 1643–1668.

[27] B. O'donoghue and E. Candes, *Adaptive restart for accelerated gradient schemes*, Foundations of computational mathematics, 15 (2015), pp. 715–732.

[28] O. Pele and M. Werman, *Fast and robust earth mover's distances.*, in ICCV, vol. 9, 2009, pp. 460–467.

[29] T. Pock and A. Chambolle, *Diagonal preconditioning for first order primal-dual algorithms in convex optimization*, in Computer Vision (ICCV), 2011 IEEE International Conference on, IEEE, 2011, pp. 1762–1769.

[30] T. Pock, D. Cremers, H. Bischof, and A. Chambolle, *An algorithm for minimizing the mumford-shah functional*, in Computer Vision, 2009 IEEE 12th International Conference on, IEEE, 2009, pp. 1133–1140.

[31] J. Rasch and A. Chambolle, *Inexact first-order primal-dual algorithms*, arXiv preprint arXiv:1803.10576, (2018).

[32] L. F. Richardson, *Ix. the approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam*, Phil. Trans. R. Soc. Lond. A, 210 (1911), pp. 307–357.

[33] R. T. Rockafellar and R. J.-B. Wets, *Variational Analysis*, vol. 317, Springer Science & Business Media, 2009.

[34] Y. Saad, *Iterative methods for sparse linear systems*, vol. 82, siam, 2003.

[35] T. Valkonen, *A primal–dual hybrid gradient method for nonlinear operators with applications to mri*, Inverse Problems, 30 (2014), p. 055012.

[36] Y. Wang, W. Yin, and J. Zeng, *Global convergence of ADMM in nonconvex nonsmooth optimization*, arXiv preprint arXiv:1511.06324, (2015).

[37] M. Yan and W. Yin, *Self equivalence of the alternating direction method of multipliers*, in Splitting Methods in Communication, Imaging, Science, and Engineering, Springer, 2016, pp. 165–194.

[38] X. Zhang, M. Burger, and S. Osher, *A unified primal-dual algorithm framework based on bregman iteration*, Journal of Scientific Computing, 46 (2011), pp. 20–46.

[39] M. Zhu and T. Chan, *An efficient primal-dual hybrid gradient algorithm for total variation image restoration*, UCLA CAM Report, 34 (2008).