

A PRIMAL-DUAL FLOW FOR AFFINE CONSTRAINED CONVEX OPTIMIZATION

HAO LUO^{*} 

Abstract. We introduce a novel primal-dual flow for affine constrained convex optimization problems. As a modification of the standard saddle-point system, our flow model is proved to possess the exponential decay property, in terms of a tailored Lyapunov function. Then two primal-dual methods are obtained from numerical discretizations of the continuous problem, and global nonergodic linear convergence rate is established *via* a discrete Lyapunov function. Instead of solving the subproblem of the primal variable, we apply the semi-smooth Newton iteration to the inner problem with respect to the multiplier, provided that there are some additional properties such as semi-smoothness and sparsity. Finally, numerical tests on the linearly constrained l_1-l_2 minimization and the total-variation based image denoising model have been provided.

Mathematics Subject Classification. 37M99, 37N40, 65K05, 90C25.

Received March 10, 2021. Accepted April 20, 2022.

1. INTRODUCTION

We are interested in the linearly constrained minimization problem

$$\min_{x \in \mathbb{R}^n} f(x) \quad \text{s.t. } Ax = b, \quad (1.1)$$

where $(A, b) \in \mathbb{R}^{m \times n} \times \mathbb{R}^m$ and $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ is proper, closed and convex. Let $\Omega := \mathbb{R}^n \times \mathbb{R}^m$ and introduce the Lagrangian

$$\mathcal{L}(x, \lambda) := f(x) + \langle \lambda, Ax - b \rangle \quad \forall (x, \lambda) \in \Omega, \quad (1.2)$$

where $\langle \cdot, \cdot \rangle$ denotes the standard l^2 -inner product, with $\|\cdot\| = \sqrt{\langle \cdot, \cdot \rangle}$ being the Euclidean norm. Assume (x^*, λ^*) is a saddle-point of $\mathcal{L}(x, \lambda)$, which means

$$\mathcal{L}(x^*, \lambda) \leq \mathcal{L}(x^*, \lambda^*) \leq \mathcal{L}(x, \lambda^*) \quad \forall (x, \lambda) \in \Omega,$$

Keywords and phrases: Convex optimization, linear constraint, dynamical system, Lyapunov function, exponential decay, discretization, nonergodic linear rate, primal-dual algorithm, semi-smooth Newton method, l_1-l_2 minimization, total-variation model.

School of Mathematical Sciences, Peking University, Beijing 100871, China.

* Corresponding author: luohao@math.pku.edu.cn

© The authors. Published by EDP Sciences, SMAI 2022

and denote by Ω^* the set of all saddle-points. Any $(x^*, \lambda^*) \in \Omega^*$ satisfies the Karush–Kuhn–Tucker (KKT) system

$$\begin{cases} 0 = \nabla_\lambda \mathcal{L}(x^*, \lambda^*) = Ax^* - b, \\ 0 \in \partial_x \mathcal{L}(x^*, \lambda^*) = \partial f(x^*) + A^\top \lambda^*, \end{cases} \quad (1.3)$$

where $\partial f(x^*)$ is the subdifferential of f at x^* .

For the standard model problem (1.1), there are a large body of primal-dual type algorithms that achieve the fast (ergodic) sublinear rate $\mathcal{O}(1/k^2)$ with strongly convex condition; see Section 1.2 for a brief review. Meanwhile, some existing works also focus on the asymptotic convergence from the continuous-time point of view, *i.e.*, the saddle-point dynamical system [23, 33]

$$\begin{cases} \lambda' = \nabla_\lambda \mathcal{L}(x, \lambda), \\ x' = -\nabla_x \mathcal{L}(x, \lambda). \end{cases} \quad (1.4)$$

In this work, we shall modify the conventional model (1.4) and introduce a novel primal-dual flow system which possesses exponential decay property. New primal-dual algorithms shall be obtained from proper time discretizations and nonergodic linear convergence rate will be proved *via* the tool of Lyapunov function.

To move on, let us make some conventions. We say a function $g : \mathbb{R}^n \rightarrow \mathbb{R}$ is L -smooth if it has L -Lipschitz continuous gradient:

$$\|\nabla g(x) - \nabla g(y)\| \leq L \|x - y\| \quad \forall x, y \in \mathbb{R}^n.$$

For a properly closed convex function $g : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$, it is called μ -convex if there exists $\mu \geq 0$ such that

$$g(x) + \langle p, y - x \rangle + \frac{\mu}{2} \|y - x\|^2 \leq g(y),$$

for all $p \in \partial g(x)$. The proximal mapping $\text{prox}_{\theta g} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ of g with $\theta > 0$ is defined by

$$\text{prox}_{\theta g}(x) := (\text{Id} + \theta \partial g)^{-1}(x) = \underset{y \in \mathbb{R}^n}{\operatorname{argmin}} \left\{ g(y) + \frac{1}{2\theta} \|y - x\|^2 \right\} \quad \forall x \in \mathbb{R}^n,$$

where Id denotes the identity operator. Clearly, if f is μ -convex, then according to (1.2), we claim that $\mathcal{L}(\cdot, \lambda)$ is also μ -convex and

$$\mathcal{L}(x, \lambda) + \langle p + A^\top \lambda, y - x \rangle + \frac{\mu}{2} \|y - x\|^2 \leq \mathcal{L}(y, \lambda) \quad \forall p \in \partial f(x). \quad (1.5)$$

1.1. Main results

Following the time rescaling technique from [19, 58], for smooth and μ -convex objective f , we propose a primal-dual flow

$$\begin{cases} \gamma x' = -\nabla_x \mathcal{L}(x, \lambda), \\ \beta \lambda' = \nabla_\lambda \mathcal{L}(x + x', \lambda), \end{cases} \quad (1.6)$$

where γ and β are two nonnegative scaling factors that are governed by $\gamma' = \mu - \gamma$ and $\beta' = -\beta$, respectively. Compared with the classical one (1.4), our new model (1.6) has two novelties: (i) it introduces two built-in time rescaling factors that unify the analysis for $\mu \geq 0$; (ii) the term $\nabla_\lambda \mathcal{L}(x + x', \lambda)$ (instead of the standard one

$\nabla_\lambda \mathcal{L}(x, \lambda)$) brings stability and reduces the oscillation; see Section 2.3 for an illustrative equilibrium analysis. Besides, the extra term x' in $\nabla_\lambda \mathcal{L}(x + x', \lambda)$ has subtle connection with the over-relaxation $x_{k+1} + (x_{k+1} - x_k)$ introduced in the primal-dual hybrid gradient (PDHG) method [16]; see Appendix A for a more reasonably intrinsic explanation.

We then equip the dynamical system (1.6) with a tailored Lyapunov function

$$\mathcal{E}(t) := \mathcal{L}(x(t), \lambda^*) - \mathcal{L}(x^*, \lambda(t)) + \frac{\gamma(t)}{2} \|x(t) - x^*\|^2 + \frac{\beta(t)}{2} \|\lambda(t) - \lambda^*\|^2, \quad t \geq 0,$$

which possesses the exponential decay (*cf.* Thm. 2.2)

$$\frac{d}{dt} \mathcal{E}(t) \leq -\mathcal{E}(t) \implies \mathcal{E}(t) \leq e^{-t} \mathcal{E}(0), \quad t \geq 0. \quad (1.7)$$

From (1.7) follows the estimate $\mathcal{L}(x(t), \lambda^*) - \mathcal{L}(x^*, \lambda(t)) \leq e^{-t} \mathcal{E}(0)$, and we can further prove that $|f(x(t)) - f(x^*)| + \|Ax(t) - b\| \leq Ce^{-t}$; see Corollary 2.3.

We also consider implicit and semi-implicit discretizations for the continuous flow (1.6) (in general nonsmooth setting) and obtain new primal-dual algorithms, which are close to the (linearized) proximal augmented Lagrangian method but adopt automatically changing parameters. In addition, instead of solving the subproblem of the primal variable, we apply the semi-smooth Newton (SsN) iteration to the subproblem with respect to the multiplier, provided that there are some hidden structures such as semi-smoothness and sparsity. By using a unified discrete Lyapunov function

$$\mathcal{E}_k = \mathcal{L}(x_k, \lambda^*) - \mathcal{L}(x^*, \lambda_k) + \frac{\gamma_k}{2} \|x_k - x^*\|^2 + \frac{\beta_k}{2} \|\lambda_k - \lambda^*\|^2,$$

we prove the contraction property:

$$\mathcal{E}_{k+1} - \mathcal{E}_k \leq -\alpha_k \mathcal{E}_{k+1} \quad \text{or} \quad \mathcal{E}_{k+1} - \mathcal{E}_k \leq -\alpha_k \mathcal{E}_k \quad \forall k \in \mathbb{N},$$

from which we obtain *nonergodic* convergence rates of the objective gap $|f(x_k) - f(x^*)|$ and the feasibility residual $\|Ax_k - b\|$. More precisely, the implicit discretization converges with (super) linear rate for convex objective f and the semi-implicit scheme possesses the rate $\mathcal{O}(\min\{L/k, (1 + \mu/L)^{-k}\})$ for the composite case $f = h + g$ where h is L -smooth and μ -convex and g is convex (possibly nonsmooth).

1.2. Related works

As one can add the indicator function of the constraint set to the objective and get rid of the linear constraint in (1.1), the proximal gradient method [7], as well as the accelerated proximal gradient method [6, 20, 58, 59], can be considered. However, they need projections onto the affine constraint set and are not suitable to handle the composite case $f = h + g$.

Therefore, prevailing algorithms are the augmented Lagrangian method (ALM) [8], the Bregman iteration [62] and their variants (linearization or acceleration) [45, 47, 48, 72, 75, 76, 82]. Another type of algorithm is the quadratic penalty method with continuation technique [49, 52]. Among those methods mentioned here, the fast rate $\mathcal{O}(1/k^2)$ is mainly in ergodic sense for primal variable and it is rare to see global nonergodic linear rate, even with strongly convex objectives. More recently, Li, Sun and Toh [53] proposed a (super) linearly convergent semi-smooth Newton based inexact proximal ALM for linear programming. Later, this method has been extended to quadratic programming [51, 60].

For the separable case: $f(x) = f_1(x_1) + f_2(x_2)$, $A = (A_1, A_2)$, we have alternating direction method of multipliers (ADMM) [35, 36] for primal problem and operator splitting methods [29, 30, 65] for dual problem. For ADMM type methods, the sublinear rate $\mathcal{O}(1/k^2)$ can be proved under partially strong convexity

assumption [70, 74, 77, 78] and global linear rate has been established as well for strongly convex (smooth) objectives [25, 26, 37]. In addition, (local) linear convergence can be derived from the error bound condition [1, 39, 55, 84, 87]. For a special case $A_1 = I$ or $A_2 = I$, there are primal-dual splitting methods [9, 16, 17, 31, 43, 46, 64, 89]. Generally speaking, we have sublinear rate $\mathcal{O}(1/k^2)$ for partially strongly convex case and linear rate for strongly convex case [18, 73, 79]. Moreover, equivalence between primal-dual splitting methods and ADMM type methods can be found in [13, 61, 85].

On the other hand, ordinary differential equation (ODE) solver approach has been revisited nowadays for investigating and developing optimization methods. For unconstrained problems, there are heavy ball model [3], asymptotically vanishing dynamical (AVD) model [71] and their extensions [2, 4, 54, 80, 81]. Besides, Luo and Chen [58] proposed the so-called Nesterov accelerated gradient flow and later generalized it to [19, 20, 57].

For linearly constrained problem (1.1), apart from the classical first-order saddle-point system (1.4), some second-order dynamics have been proposed as well. Zeng, Lei and Chen [88] generalized the AVD model and obtained the decay rate $\mathcal{O}(t^{-\min\{2,2\alpha/3\}})$ via a suitable Lyapunov function. He, Hu and Fang [41] extended the dynamical system in [88] to separable case. Revisiting the scaled alternating direction method of multipliers [10], Franca, Robinson and Vidal [34] derived a continuous model which is also related to the AVD model and proved the decay rate $\mathcal{O}(1/t^2)$. Yet, none of Zeng *et al.* [88], He *et al.* [41] and Franca *et al.* [34] neither considered numerical discretizations for their dynamical systems nor presented new optimization algorithms for the original optimization problem. For general minimax problems, there are some works on dynamical system approach [22, 56].

Comparing with existing works, we summarize our main contributions as below:

- The continuous primal-dual flow (1.6) adopts built-in time rescaling factors for both convex and strongly convex cases and has exponential decay rate with respect to a proper Lyapunov function.
- A simple but illustrative equilibrium analysis shows the gain of stability that is benefit from the modification introduced in (1.6).
- New primal-dual algorithms with automatically changing parameters are obtained from proper time discretizations of the continuous model and the semi-smooth Newton method is considered for the subproblem with respect to the multiplier.
- Nonergodic (super) linear convergence rate of the objective gap and feasibility residual is established via the tool of discrete Lyapunov function.

The rest of this paper is organized as follows. Section 2 starts from the classical saddle-point system and introduces a new primal-dual flow. Then Sections 3 and 4 consider implicit and semi-implicit discretizations respectively and establish the (super) linear convergence rates of the resulted primal-dual algorithms. Numerical performances on the l_1-l_2 minimization and the total-variation based denoising model are presented in Section 5 and finally, some concluding remarks are given in Section 6.

2. CONTINUOUS PROBLEMS

2.1. The saddle-point system

To present the main idea clearly, let us start from the rescaled saddle-point system

$$\begin{cases} \beta\lambda' = \nabla_\lambda \mathcal{L}(x, \lambda), \\ \gamma x' = -\nabla_x \mathcal{L}(x, \lambda), \end{cases} \quad (2.1)$$

with the initial condition $(x(0), \lambda(0)) = (x_0, \lambda_0) \in \Omega$, where γ and β are two artificial time rescaling factors and satisfy (cf. [19, 58])

$$\gamma' = \mu - \gamma, \quad \beta' = -\beta, \quad (2.2)$$

with positive initial condition $(\gamma(0), \beta(0)) = (\gamma_0, \beta_0)$. One can easily solve (2.2) to obtain

$$\gamma(t) = \gamma_0 e^{-t} + \mu(1 - e^{-t}) \quad \text{and} \quad \beta(t) = \beta_0 e^{-t} \quad t \geq 0, \quad (2.3)$$

which implies γ and β are positive and converge exponentially to μ and 0, respectively.

Assume $f \in C_L^1$ and define $F : \mathbb{R}_+ \times \Omega \rightarrow \Omega$ by that

$$F(t, Z) := \begin{pmatrix} -\frac{1}{\gamma(t)} \nabla_x \mathcal{L}(x, \lambda) \\ \frac{1}{\beta(t)} \nabla_\lambda \mathcal{L}(x, \lambda) \end{pmatrix} \quad \forall Z = \begin{pmatrix} x \\ \lambda \end{pmatrix} \in \Omega.$$

Then (2.1) can be rewritten as $Z'(t) = F(t, Z(t))$ and a direct calculation yields that for all $Z, Y \in \Omega$ and $0 \leq s \leq t$,

$$\|F(t, Z) - F(s, Y)\| \leq C_0(L + \|A\|) (|t - s| \|Z - Z^*\| + \|Z - Y\|) e^t,$$

where $Z^* = (x^*, \lambda^*) \in \Omega^*$ and the bounded positive constant C_0 depends only on γ_0, β_0 and μ . This means F is locally Lipschitz continuous and according to Proposition 6.2.1 of [40] and Corollary A.2 of [11], the first-order dynamical system (2.1) exists a unique solution $Z = (x, \lambda) \in C^1(\mathbb{R}_+; \Omega)$.

Let $\mathbf{V} := \Omega \times \mathbb{R}_+ \times \mathbb{R}_+$ and for any $X = (x, \lambda, \gamma, \beta) \in \mathbf{V}$, introduce a Lyapunov function

$$\mathcal{E}(X) := \mathcal{L}(x, \lambda^*) - \mathcal{L}(x^*, \lambda) + \frac{\gamma}{2} \|x - x^*\|^2 + \frac{\beta}{2} \|\lambda - \lambda^*\|^2. \quad (2.4)$$

Our goal is to establish the exponential decay property of (2.4) along with the solution trajectory $X : \mathbb{R}_+ \rightarrow \mathbf{V}$. Below, we present a lemma which violates our goal but heuristically motivates us to the right way.

Lemma 2.1. *Assume f is L -smooth and μ -convex with $\mu \geq 0$ and let $X = (x, \lambda, \gamma, \beta) : \mathbb{R}_+ \rightarrow \mathbf{V}$ be the unique solution to (2.1) and (2.2), then*

$$\frac{d}{dt} \mathcal{E}(X) \leq -\mathcal{E}(X) - \gamma \|x'\|^2 - \langle Ax', \lambda - \lambda^* \rangle. \quad (2.5)$$

Proof. As discussed above, $(x, \lambda) \in C^1(\mathbb{R}_+; \Omega)$ exists uniquely and by (2.1), a direct computation gives

$$\begin{aligned} \frac{d}{dt} \mathcal{E}(X) &= \langle \nabla_x \mathcal{E}(X), x' \rangle + \langle \nabla_\lambda \mathcal{E}(X), \lambda' \rangle + \langle \nabla_\gamma \mathcal{E}(X), \gamma' \rangle + \langle \nabla_\beta \mathcal{E}(X), \beta' \rangle \\ &= \underbrace{-\frac{1}{\gamma} \langle \nabla_x \mathcal{L}(x, \lambda), \nabla_x \mathcal{L}(x, \lambda^*) \rangle}_{I_1} + \underbrace{\langle \nabla_\lambda \mathcal{L}(x, \lambda), \lambda - \lambda^* \rangle - \langle \nabla_x \mathcal{L}(x, \lambda), x - x^* \rangle}_{I_2} \\ &\quad \underbrace{-\frac{\beta}{2} \|\lambda - \lambda^*\|^2 + \frac{\mu - \gamma}{2} \|x - x^*\|^2}_{I_3} \\ &:= I_1 + I_2 + I_3. \end{aligned}$$

We split $\nabla_x \mathcal{L}(x, \lambda^*) = \nabla_x \mathcal{L}(x, \lambda) - A^\top(\lambda - \lambda^*)$ and use the relation $\gamma x' = -\nabla_x \mathcal{L}(x, \lambda)$ to get

$$I_1 = -\frac{1}{\gamma} \langle \nabla_x \mathcal{L}(x, \lambda), \nabla_x \mathcal{L}(x, \lambda) - A^\top(\lambda - \lambda^*) \rangle = -\gamma \|x'\|^2 - \langle Ax', \lambda - \lambda^* \rangle. \quad (2.6)$$

Also, we reformulate I_2 as follows

$$I_2 = \langle \nabla_\lambda \mathcal{L}(x, \lambda), \lambda - \lambda^* \rangle - \langle Ax - Ax^*, \lambda - \lambda^* \rangle - \langle \nabla_x \mathcal{L}(x, \lambda^*), x - x^* \rangle = -\langle \nabla_x \mathcal{L}(x, \lambda^*), x - x^* \rangle, \quad (2.7)$$

where we have used the optimality condition $Ax^* = b$. Since f is μ -convex, we know that $\mathcal{L}(\cdot, \lambda^*)$ is also μ -convex and it follows from (1.5) that

$$I_2 \leq \mathcal{L}(x^*, \lambda^*) - \mathcal{L}(x, \lambda^*) - \frac{\mu}{2} \|x - x^*\|^2 = \mathcal{L}(x^*, \lambda) - \mathcal{L}(x, \lambda^*) - \frac{\mu}{2} \|x - x^*\|^2.$$

Here, recall the fact that $\mathcal{L}(x^*, \cdot)$ is a constant. Consequently, collecting I_3 , (2.6) and (2.7) proves (2.5). \square

To obtain the exponential decay $\mathcal{E}'(X) \leq -\mathcal{E}(X)$ from (2.5), we shall prove $-\gamma \|x'\|^2 - \langle Ax', \lambda - \lambda^* \rangle \leq 0$. In stead of twisting on the existence of this, in the next section, we resort to introducing a subtle modification that cancels exactly the cross term $\langle Ax', \lambda - \lambda^* \rangle$ in (2.5) and finally leads to the desired estimate.

2.2. A new primal-dual flow

Although (2.5) fails to give the desired result, it suggests a simple remedy: replacing $\nabla_\lambda \mathcal{L}(x, \lambda)$ by $\nabla_\lambda \mathcal{L}(x + x', \lambda)$. Then the first part I_1 (cf. (2.6)) brings one more term $\langle Ax', \lambda - \lambda^* \rangle$ which offsets exactly the last term in (2.5) while both I_2 and I_3 keep unchanged.

Namely, we leave the parameter system (2.2) invariant but modify (2.1) properly to obtain a novel primal-dual flow

$$\begin{cases} \gamma x' = -\nabla_x \mathcal{L}(x, \lambda), \\ \beta \lambda' = \nabla_\lambda \mathcal{L}(x + x', \lambda). \end{cases} \quad (2.8a)$$

$$(2.8b)$$

Similar with (2.1), we claim that (2.8) admits a unique classical solution $(x, \lambda) \in C^1(\mathbb{R}_+; \Omega)$. We also mention that the extrapolation idea $x + x'$ in (2.8b) can be found previously in the second-order primal-dual ODE proposed by [88]. In the sequel, we shall complete the exponential decay of the Lyapunov function (2.4) and then provide an illustrative equilibrium analysis that gives a convincible explanation of the subtle modification $x + x'$. Additionally, in Appendix A, we present an over-relaxation perspective, which perhaps shows the intrinsic connection with the PDHG method [16].

Theorem 2.2. *Assume f is L -smooth and μ -convex with $\mu \geq 0$ and let $X = (x, \lambda, \gamma, \beta) : \mathbb{R}_+ \rightarrow \mathbf{V}$ be the unique solution to (2.2) and (2.8), then*

$$\frac{d}{dt} \mathcal{E}(X) \leq -\mathcal{E}(X) - \gamma \|x'\|^2. \quad (2.9)$$

Consequently, we have the exponential decay

$$\mathcal{E}(X(t)) + \int_0^t e^{s-t} \gamma(s) \|x'(s)\|^2 ds \leq e^{-t} \mathcal{E}(X(0)), \quad 0 \leq t < \infty. \quad (2.10)$$

Proof. According to the above discussions, the proof of (2.9) is in line with that of (2.5) and thus omitted here. The estimate (2.10) follows from (2.9) immediately. \square

Thanks to the two scaling factors introduced in (2.2), the exponential decay (2.10) holds uniformly for $\mu \geq 0$. Let $\gamma_{\min} := \min\{\gamma_0, \mu\}$, then by (2.3), we have

$$\gamma(t) \geq \max\{\gamma_{\min}, \gamma_0 e^{-t}\} \quad \forall t \geq 0. \quad (2.11)$$

Corollary 2.3. *Assume f is L -smooth and μ -convex with $\mu \geq 0$. Then for the unique solution $(x, \lambda) : \mathbb{R}_+ \rightarrow \Omega$ of (2.8), we have the following.*

- $\sqrt{\gamma_0 + \gamma_{\min} e^t} \|x'(t)\| \in L^2(0, \infty)$.
- $0 \leq \mathcal{L}(x(t), \lambda^*) - \mathcal{L}(x^*, \lambda(t)) \leq e^{-t} \mathcal{E}(X(0))$.
- $\lambda(t)$ is bounded: $\beta_0 \|\lambda(t) - \lambda^*\|^2 \leq 2\mathcal{E}(X(0))$.
- $x(t)$ is bounded: $\gamma_0 \|x(t) - x^*\|^2 \leq 2\mathcal{E}(X(0))$ and $\gamma_{\min} \|x(t) - x^*\|^2 \leq 2e^{-t} \mathcal{E}(X(0))$.
- $\|Ax(t) - b\| \leq e^{-t} \mathcal{R}_0$ and $|f(x(t)) - f(x^*)| \leq e^{-t} (\mathcal{E}(X(0)) + \mathcal{R}_0 \|\lambda^*\|)$, where

$$\mathcal{R}_0 := \sqrt{2\beta_0 \mathcal{E}(X(0))} + \beta_0 \|\lambda_0 - \lambda^*\| + \|Ax_0 - b\|.$$

Proof. The first to the fourth follow directly from (2.4), (2.10) and (2.11). Let us prove the last one. Define $\xi(t) := \lambda(t) - \beta^{-1}(t)(Ax(t) - b)$, then by (2.2) and (2.8b),

$$\frac{d\xi}{dt} = \lambda'(t) - \beta^{-1}(t)(Ax'(t) + Ax(t) - b) = 0, \quad (2.12)$$

which says $\xi(t) = \xi(0)$ and also implies that

$$\|Ax(t) - b\| = \beta(t) \|\lambda(t) - \xi(0)\| \leq \beta(t) (\|\lambda(t) - \lambda^*\| + \|\xi(0) - \lambda^*\|).$$

Hence, from the fact $\beta(t) = \beta_0 e^{-t}$ and the boundness of $\|\lambda(t) - \lambda^*\|$, we have

$$\|Ax(t) - b\| \leq e^{-t} \left(\sqrt{2\beta_0 \mathcal{E}(X(0))} + \beta_0 \|\xi(0) - \lambda^*\| \right) \leq e^{-t} \mathcal{R}_0. \quad (2.13)$$

Besides, it follows from (2.10) that

$$0 \leq \mathcal{L}(x(t), \lambda^*) - \mathcal{L}(x^*, \lambda(t)) = f(x(t)) - f(x^*) + \langle \lambda^*, Ax(t) - b \rangle \leq e^{-t} \mathcal{E}(X(0)),$$

which together with the previous estimate (2.13) gives

$$|f(x(t)) - f(x^*)| \leq \|\lambda^*\| \|Ax(t) - b\| + e^{-t} \mathcal{E}(X(0)) \leq e^{-t} (\mathcal{E}(X(0)) + \mathcal{R}_0 \|\lambda^*\|).$$

This establishes the exponential decay of the primal objective error and completes the proof. \square

Remark 2.4. From Corollary 2.3, we conclude that for $\mu \geq 0$, the primal-dual gap $\mathcal{L}(x(t), \lambda^*) - \mathcal{L}(x^*, \lambda(t))$, the primal objective residual $|f(x(t)) - f(x^*)|$ and the feasibility violation $\|Ax(t) - b\|$ decrease exponentially. We also have strong convergence: $\|x(t) - x^*\|^2 \leq C e^{-t}$ for the strongly convex case $\mu > 0$.

Remark 2.5. We mention that the well-posedness with general nonsmooth objective f is of interest to study further. As we can see, the modified system (2.8) promises the exponential decay (2.9) but it is totally different from the original one (2.1). In nonsmooth setting, (2.1) can be almost viewed as a dynamical system governed by a maximally monotone operator:

$$Z'(t) + \Lambda(t)M(Z(t)) \ni 0, \quad (2.14)$$

where $\Lambda(t) = \text{diag}(\gamma^{-1}(t)I_n, \beta^{-1}(t)I_m)$, $Z(t) = (x(t), \lambda(t))$ and the maximally monotone operator $M : \Omega \rightarrow 2^\Omega$ is defined by that

$$M(Z) := \begin{pmatrix} \partial f(x) + A^\top \lambda \\ b - Ax \end{pmatrix} \quad \forall Z = \begin{pmatrix} x \\ \lambda \end{pmatrix} \in \Omega. \quad (2.15)$$

According to Section 4.2 of [28], we claim that (2.14) admits a unique solution $Z = (x, \lambda) \in W_{\text{loc}}^{1,\infty}(\mathbb{R}_+; \Omega)$. However, our primal-dual flow (2.8) reads as (cf. (3.1))

$$Z'(t) + R(t)M(Z(t)) \ni 0, \quad (2.16)$$

where $R(t)$ is a lower triangular matrix:

$$R(t) = \begin{pmatrix} \gamma^{-1}(t) \text{Id} & O \\ \gamma(t)^{-1} \beta^{-1}(t) A & \beta^{-1}(t) \text{Id} \end{pmatrix}.$$

The existence and uniqueness of the solution to (2.16) is under studying. In addition, both the exponential decay (2.9) and (weak) convergence of the trajectory $Z(t)$ to a saddle-point $(x_\infty, \lambda_\infty) \in \Omega^*$ deserve future investigations.

2.3. A simple equilibrium analysis

Let $p > 2$ be a positive even integer and consider a simple smooth convex function

$$f(x) = \frac{1}{p} (x_1^p + x_2^p) \quad \forall x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \in \mathbb{R}^2,$$

with the linear constraint $ax = x_1 - x_2 = 0$, where $a = (1, -1)$. Clearly $(x^*, \lambda^*) = (0, 0, 0)$ is the unique saddle point. Take $\mu = 0$ and for simplicity we choose $\gamma_0 = \beta_0 = 1$, then $\gamma(t) = \beta(t) = e^{-t}$ and the original model (2.1) becomes

$$\begin{cases} \lambda' = e^t(x_1 - x_2), \\ x_1' = -e^t(x_1^{p-1} + \lambda), \\ x_2' = -e^t(x_2^{p-1} - \lambda). \end{cases} \quad (2.17)$$

The “linearization” around (x^*, λ^*) is

$$\begin{pmatrix} \hat{\lambda} \\ \hat{x} \end{pmatrix}' = e^t B \begin{pmatrix} \hat{\lambda} \\ \hat{x} \end{pmatrix} \quad \text{with} \quad B = \begin{pmatrix} 0 & a \\ -a^\top & O \end{pmatrix}.$$

Note that B has three distinct eigenvalues: $b_1 = 0$, $b_2 = -i\sqrt{2}$ and $b_3 = i\sqrt{2}$. This implies (x^*, λ^*) is stable but not asymptotically stable and the solution trajectory of (2.17) will spin around (x^*, λ^*) with high oscillation and thus converges dramatically slowly.

The modified system (2.8) reads as follows

$$\begin{cases} \lambda' = e^t(x_1 + x_1' - x_2 - x_2'), \\ x_1' = -e^t(x_1^{p-1} + \lambda), \\ x_2' = -e^t(x_2^{p-1} - \lambda), \end{cases} \quad (2.18)$$

and its “linearization” at (x^*, λ^*) is

$$\begin{pmatrix} \hat{\lambda} \\ \hat{x} \end{pmatrix}' = e^t \hat{B}(t) \begin{pmatrix} \hat{\lambda} \\ \hat{x} \end{pmatrix} \quad \text{with} \quad \hat{B}(t) = \begin{pmatrix} -2e^t & a \\ -a^\top & O \end{pmatrix}.$$

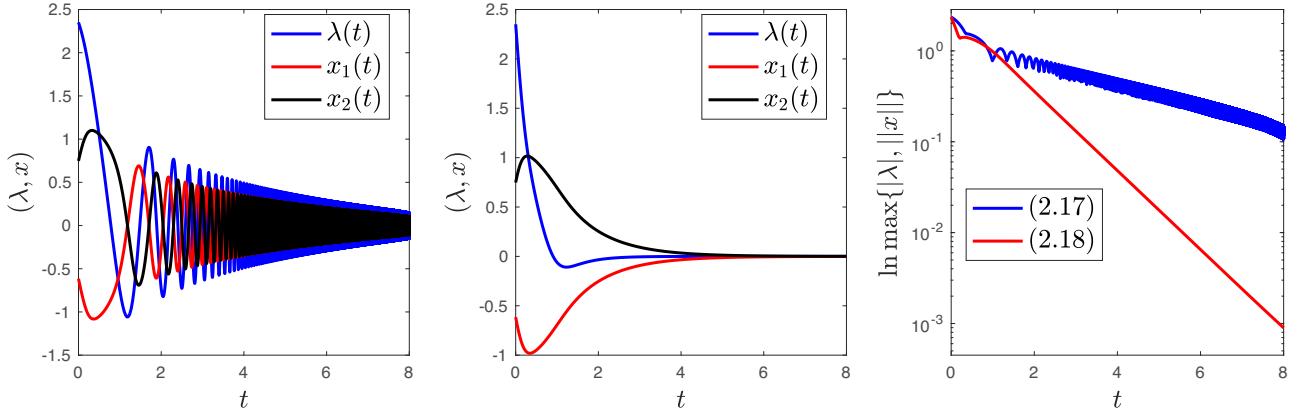


FIGURE 1. Solution trajectories (the left for (2.17) and the medium for (2.18)) and their errors (the right) with $p = 6$.

Given any fixed time $t \geq \ln \sqrt{2}$, all the eigenvalues of $\hat{B}(t)$ are

$$\hat{b}_1 = 0, \quad \hat{b}_2 = -\frac{2}{e^t + \sqrt{e^{2t} - 2}} \quad \text{and} \quad \hat{b}_3 = -e^t - \sqrt{e^{2t} - 2}.$$

From this, we observe more negativity of the real part of nonzero eigenvalues and hopefully the solution (x, λ) to the modified system (2.18) converges to (x^*, λ^*) more quickly.

In conclusion, our primal-dual flow (2.8) with subtle extrapolation $x + x'$ reduces the oscillation and accelerates the convergence; see Figure 1.

3. AN IMPLICIT SCHEME

From now on, we move to discrete level and consider general nonsmooth μ -convex objective f with $\mu \geq 0$. In this setting our primal-dual flow (2.8) becomes a differential inclusion

$$\begin{cases} \gamma x' \in -\partial_x \mathcal{L}(x, \lambda), \\ \beta \lambda' = \nabla_\lambda \mathcal{L}(x + x', \lambda), \end{cases} \quad (3.1a)$$

$$(3.1b)$$

where $\partial_x \mathcal{L}(x, \lambda) = \partial f(x) + A^\top \lambda$. As discussed in Remark 2.5, well-posedness of the solution to (3.1) in proper sense is left as a future topic. In what follows, we shall present new primal-dual algorithms based on implicit Euler discretization (this section) and semi-implicit discretization (the next section), respectively. Similarly with the continuous level, the tool of Lyapunov function plays important role in convergence rate analysis.

3.1. Implicit discretization

We first consider an implicit Euler scheme for (3.1):

$$\begin{cases} v_{k+1} = x_{k+1} + \frac{x_{k+1} - x_k}{\alpha_k}, \end{cases} \quad (3.2a)$$

$$\begin{cases} \beta_k \frac{\lambda_{k+1} - \lambda_k}{\alpha_k} = \nabla_\lambda \mathcal{L}(v_{k+1}, \lambda_{k+1}), \end{cases} \quad (3.2b)$$

$$\begin{cases} \gamma_k \frac{x_{k+1} - x_k}{\alpha_k} \in -\partial_x \mathcal{L}(x_{k+1}, \lambda_{k+1}), \end{cases} \quad (3.2c)$$

where $\alpha_k > 0$ denotes the step size and the parameter system (2.2) is also discretized implicitly

$$\frac{\gamma_{k+1} - \gamma_k}{\alpha_k} = \mu - \gamma_{k+1}, \quad \frac{\beta_{k+1} - \beta_k}{\alpha_k} = -\beta_{k+1}, \quad (3.3)$$

with $\gamma_0 > 0$ and $\beta_0 > 0$.

Let us transform the time discretization (3.2) to a primal-dual algorithm. From (3.2c) it follows that

$$\frac{x_{k+1} - x_k + \theta_k A^\top \lambda_{k+1}}{\theta_k} \in -\partial f(x_{k+1}), \quad (3.4)$$

where $\theta_k = \alpha_k/\gamma_k$. Plugging (3.2a) into (3.2b) and using (3.3), we find

$$\lambda_{k+1} = \lambda_k - \frac{1}{\beta_k} (Ax_k - b) + \frac{1}{\beta_{k+1}} (Ax_{k+1} - b). \quad (3.5)$$

Then, combining (3.4), (3.5) gives

$$\frac{x_{k+1} - \hat{x}_k}{\theta_k} + \frac{1}{\beta_{k+1}} A^\top (Ax_{k+1} - b) + A^\top \lambda_k \in -\partial f(x_{k+1}),$$

where $\hat{x}_k := x_k + \theta_k/\beta_k A^\top (Ax_k - b)$. Consequently, we obtain

$$\begin{cases} x_{k+1} = \underset{x \in \mathbb{R}^n}{\operatorname{argmin}} \left\{ \mathcal{L}(x, \lambda_k) + \frac{1}{2\beta_{k+1}} \|Ax - b\|^2 + \frac{1}{2\theta_k} \|x - \hat{x}_k\|^2 \right\}, \end{cases} \quad (3.6a)$$

$$\begin{cases} v_{k+1} = x_{k+1} + (x_{k+1} - x_k)/\alpha_k, \end{cases} \quad (3.6b)$$

$$\begin{cases} \lambda_{k+1} = \lambda_k + \alpha_k/\beta_k (Av_{k+1} - b). \end{cases} \quad (3.6c)$$

Note that in (3.2c) we used only the Lagrangian function $\mathcal{L}(x, \lambda)$ without the augmented term $\|Ax - b\|^2$. However, λ_{k+1} and x_{k+1} are coupled with each other, and this equips (3.6a) with the augmented term.

The method (3.6) is very close to the proximal ALM and the key is to solve the subproblem (3.6a) with respect to the primal variable x_{k+1} . On the other hand, from (3.4) we observe that $x_{k+1} = \mathbf{prox}_{\theta_k f}(x_k - \theta_k A^\top \lambda_{k+1})$. Putting this back to (3.6c) gives a nonlinear equation in terms of the multiplier λ_{k+1} :

$$\beta_{k+1} \lambda_{k+1} - A \mathbf{prox}_{\theta_k f}(x_k - \theta_k A^\top \lambda_{k+1}) - z_k = 0, \quad (3.7)$$

where $z_k = \beta_{k+1} (\lambda_k - \beta_k^{-1} (Ax_k - b)) - b$. As discussed later in Section 4.1, instead of computing x_{k+1} from (3.6a), we apply the semi-smooth Newton method [32] to obtain λ_{k+1} and then update x_{k+1} .

Remark 3.1. For a better understanding of (3.6a) and (3.7), we give an operator perspective. Notice that (3.2) is a nonlinear saddle-point type equation with respect to x_{k+1} and λ_{k+1} :

$$\mathcal{A} \begin{pmatrix} x_{k+1} \\ \lambda_{k+1} \end{pmatrix} = r_k \quad \text{where} \quad \mathcal{A} = \begin{pmatrix} \text{Id} + \theta_k \partial f & A^\top \\ -A & \beta_{k+1} \text{Id} \end{pmatrix}. \quad (3.8)$$

Formally, we have the following factorizations:

$$\begin{aligned}\mathcal{A} &= \begin{pmatrix} \text{Id} & A^\top/\beta_{k+1} \\ O & \text{Id} \end{pmatrix} \begin{pmatrix} \mathcal{P} & O \\ O & \beta_{k+1} \text{Id} \end{pmatrix} \begin{pmatrix} \text{Id} & O \\ -A/\beta_{k+1} & \text{Id} \end{pmatrix}, \\ \mathcal{A} &= \begin{pmatrix} \text{Id} & O \\ (\text{Id} + \theta_k \partial f)^{-1}(-A) & \text{Id} \end{pmatrix} \begin{pmatrix} \text{Id} + \theta_k \partial f & O \\ O & \mathcal{S} \end{pmatrix} \begin{pmatrix} \text{Id} & (\text{Id} + \theta_k \partial f)^{-1}(-A^\top) \\ O & \text{Id} \end{pmatrix},\end{aligned}$$

where $\mathcal{P} = \text{Id} + \theta_k \partial f + A^\top A/\beta_{k+1}$ and

$$\mathcal{S} = \beta_{k+1} \text{Id} - A(\text{Id} + \theta_k \partial f)^{-1}(-A^\top) = \beta_{k+1} \text{Id} - A\text{prox}_{\theta_k f}(-A^\top)$$

is nothing but the Schur complement. Hence, to solve (3.8), we can compute

$$\mathcal{P}^{-1} = (\text{Id} + \theta_k \partial f + A^\top A/\beta_{k+1})^{-1},$$

which corresponds to the augmented Lagrangian method (3.6a). On the other hand, one can calculate

$$\mathcal{S}^{-1} = (\beta_{k+1} \text{Id} - A\text{prox}_{\theta_k f}(-A^\top))^{-1},$$

which is equivalent to solve the nonlinear equation (3.7).

The implicit scheme (3.2) (*i.e.*, the method (3.6)) has been rewritten below as an algorithm framework, which is called the *implicit primal-dual* (Im-PD) method. According to Theorem 3.2, we have global linear rate $(1 + \hat{\alpha})^{-k}$ as long as the step size is bounded below $\alpha_k \geq \hat{\alpha} > 0$, and superlinear convergence follows if $\alpha_k \rightarrow \infty$. Note that this holds even for convex case $\mu = 0$. In fact, the fully implicit scheme (3.2) inherits the exponential decay (2.9) from the continuous level, and thus we have the contraction (3.10) which has no restriction on the step size α_k . Besides, the strong convexity constant μ of the objective f is not necessarily needed since one can set $\mu = 0$ in (3.2) and this leaves the final rate in Theorem 3.2 unchanged.

Algorithm 1 Im-PD method for problem (1.1) with f being μ -convex ($\mu \geq 0$)

Require: $\gamma_0 > 0$, $\beta_0 > 0$, $x_0 \in \mathbb{R}^n$, $\lambda_0 \in \mathbb{R}^m$.

- 1: **for** $k = 0, 1, \dots$ **do**
- 2: Choose the step size $\alpha_k > 0$.
- 3: Update $\beta_{k+1} = \beta_k/(1 + \alpha_k)$ and $\gamma_{k+1} = (\mu\alpha_k + \gamma_k)/(1 + \alpha_k)$.
- 4: Set $\theta_k = \alpha_k/\gamma_k$ and $z_k = \beta_{k+1}(\lambda_k - \beta_k^{-1}(Ax_k - b)) - b$.
- 5: Solve λ_{k+1} from (3.7) *via* the SsN iteration (4.10) with the line search procedure (4.11).
- 6: Update $x_{k+1} = \text{prox}_{\theta_k f}(x_k - \theta_k A^\top \lambda_{k+1})$.
- 7: **end for**

3.2. Convergence rate

We now prove the convergence rate of the implicit scheme (3.2) (*i.e.* Algorithm 1) *via* a discrete analogue to (2.4):

$$\mathcal{E}_k := \mathcal{L}(x_k, \lambda^*) - \mathcal{L}(x^*, \lambda_k) + \frac{\beta_k}{2} \|\lambda_k - \lambda^*\|^2 + \frac{\gamma_k}{2} \|x_k - x^*\|^2, \quad (3.9)$$

where $(x^*, \lambda^*) \in \Omega^*$ and $(x_k, \lambda_k, \gamma_k, \beta_k) \in \mathbf{V}$.

Theorem 3.2. Assume f is μ -convex with $\mu \geq 0$. Let $\{(x_k, \lambda_k, \gamma_k, \beta_k)\}$ be generated by Algorithm 1 with arbitrary step size $\alpha_k > 0$, then we have the contraction

$$\mathcal{E}_{k+1} - \mathcal{E}_k \leq -\alpha_k \mathcal{E}_{k+1}, \quad \text{for all } k \in \mathbb{N}. \quad (3.10)$$

Moreover, there holds that

$$\left\{ \begin{array}{l} \|Ax_k - b\| \leq \mathcal{R}_0 \times \prod_{i=0}^{k-1} \frac{1}{1 + \alpha_i}, \end{array} \right. \quad (3.11)$$

$$\left\{ \begin{array}{l} 0 \leq \mathcal{L}(x_k, \lambda^*) - \mathcal{L}(x^*, \lambda_k) \leq \mathcal{E}_0 \times \prod_{i=0}^{k-1} \frac{1}{1 + \alpha_i}, \end{array} \right. \quad (3.12)$$

$$\left\{ \begin{array}{l} |f(x_k) - f(x^*)| \leq (\mathcal{E}_0 + \mathcal{R}_0 \|\lambda^*\|) \times \prod_{i=0}^{k-1} \frac{1}{1 + \alpha_i}, \end{array} \right. \quad (3.13)$$

where $\mathcal{R}_0 := \sqrt{2\beta_0 \mathcal{E}_0} + \beta_0 \|\lambda_0 - \lambda^*\| + \|Ax_0 - b\|$.

Proof. To prove (3.10), we mimic the continuous level (cf. Sect. 2) but replace the derivative with the difference $\mathcal{E}_{k+1} - \mathcal{E}_k = I_1 + I_2 + I_3$, where

$$\left\{ \begin{array}{l} I_1 := \mathcal{L}(x_{k+1}, \lambda^*) - \mathcal{L}(x_k, \lambda^*), \\ I_2 := \frac{\beta_{k+1}}{2} \|\lambda_{k+1} - \lambda^*\|^2 - \frac{\beta_k}{2} \|\lambda_k - \lambda^*\|^2, \\ I_3 := \frac{\gamma_{k+1}}{2} \|x_{k+1} - x^*\|^2 - \frac{\gamma_k}{2} \|x_k - x^*\|^2. \end{array} \right. \quad (3.14)$$

Let $p_{k+1} = (x_{k+1} - (x_k - \theta_k A^\top \lambda_{k+1})) / \theta_k$, then by (3.4), we have $A^\top \lambda^* - p_{k+1} \in \partial \mathcal{L}(x_{k+1}, \lambda^*)$. Since $\mathcal{L}(\cdot, \lambda^*)$ is μ -convex, from (1.5) we obtain

$$I_1 \leq \langle A^\top \lambda^* - p_{k+1}, x_{k+1} - x_k \rangle - \frac{\mu}{2} \|x_{k+1} - x_k\|^2.$$

Shift λ^* to λ_{k+1} and use the relation

$$p_{k+1} - A^\top \lambda_{k+1} = x_{k+1} - x_k \quad (3.15)$$

to lighten the previous estimate as follows

$$I_1 \leq -\langle Ax_{k+1} - Ax_k, \lambda_{k+1} - \lambda^* \rangle, \quad (3.16)$$

where the surplus negative term $-\|x_{k+1} - x_k\|^2$ has been omitted.

Then let us focus on I_2 and I_3 . By (3.3), a direct computation yields

$$\begin{aligned} I_2 &= -\frac{\alpha_k \beta_{k+1}}{2} \|\lambda_{k+1} - \lambda^*\|^2 + \frac{\beta_k}{2} \left(\|\lambda_{k+1} - \lambda^*\|^2 - \|\lambda_k - \lambda^*\|^2 \right) \\ &= -\frac{\alpha_k \beta_{k+1}}{2} \|\lambda_{k+1} - \lambda^*\|^2 + \beta_k \langle \lambda_{k+1} - \lambda_k, \lambda_{k+1} - \lambda^* \rangle - \frac{\beta_k}{2} \|\lambda_{k+1} - \lambda_k\|^2. \end{aligned}$$

Plugging (3.2b) into the second term and dropping the last negative term lead to

$$I_2 \leq -\frac{\alpha_k \beta_{k+1}}{2} \|\lambda_{k+1} - \lambda^*\|^2 + \alpha_k \langle Ax_{k+1} - b, \lambda_{k+1} - \lambda^* \rangle + \langle Ax_{k+1} - Ax_k, \lambda_{k+1} - \lambda^* \rangle. \quad (3.17)$$

Similarly, we have

$$\begin{aligned} I_3 &= \frac{\gamma_{k+1} - \gamma_k}{2} \|x_{k+1} - x^*\|^2 + \frac{\gamma_k}{2} \left(\|x_{k+1} - x^*\|^2 - \|x_k - x^*\|^2 \right) \\ &= \frac{\alpha_k}{2} (\mu - \gamma_{k+1}) \|x_{k+1} - x^*\|^2 + \gamma_k \langle x_{k+1} - x_k, (x_{k+1} + x_k)/2 - x^* \rangle. \end{aligned} \quad (3.18)$$

By (3.15), we divide the last term by that

$$\begin{aligned} \gamma_k \langle x_{k+1} - x_k, (x_{k+1} + x_k)/2 - x^* \rangle &= \gamma_k \langle x_{k+1} - x_k, x_{k+1} - x^* \rangle - \frac{\gamma_k}{2} \|x_{k+1} - x_k\|^2 \\ &= -\alpha_k \langle A^\top \lambda_{k+1} - p_{k+1}, x_{k+1} - x^* \rangle - \frac{\gamma_k}{2} \|x_{k+1} - x_k\|^2. \end{aligned}$$

Since (3.4) implies $A^\top \lambda_{k+1} - p_{k+1} \in \partial \mathcal{L}(x_{k+1}, \lambda_{k+1})$, we obtain

$$\begin{aligned} &-\alpha_k \langle A^\top \lambda_{k+1} - p_{k+1}, x_{k+1} - x^* \rangle \\ &\leq \alpha_k (\mathcal{L}(x^*, \lambda_{k+1}) - \mathcal{L}(x_{k+1}, \lambda_{k+1})) - \frac{\mu \alpha_k}{2} \|x_{k+1} - x^*\|^2 \\ &= \alpha_k (\mathcal{L}(x^*, \lambda_{k+1}) - \mathcal{L}(x_{k+1}, \lambda^*)) - \frac{\mu \alpha_k}{2} \|x_{k+1} - x^*\|^2 - \alpha_k \langle Ax_{k+1} - b, \lambda_{k+1} - \lambda^* \rangle, \end{aligned}$$

which promises the following bound

$$\begin{aligned} I_3 &\leq \alpha_k (\mathcal{L}(x^*, \lambda_{k+1}) - \mathcal{L}(x_{k+1}, \lambda^*)) - \frac{\alpha_k \gamma_{k+1}}{2} \|x_{k+1} - x^*\|^2 \\ &\quad - \alpha_k \langle Ax_{k+1} - b, \lambda_{k+1} - \lambda^* \rangle - \frac{\gamma_k}{2} \|x_{k+1} - x_k\|^2. \end{aligned} \quad (3.19)$$

Consequently, combining (3.16), (3.17), (3.19) proves (3.10).

From (3.10) we conclude that $\mathcal{E}_k \leq \mathcal{E}_0 \times \prod_{i=0}^{k-1} \frac{1}{1+\alpha_i}$, which together with (3.9) implies (3.12) and that $\beta_0 \|\lambda_k - \lambda^*\|^2 \leq 2\mathcal{E}_0$. Hence it is sufficient to prove (3.11) and (3.13). From (3.5) follows that

$$\lambda_k - \frac{1}{\beta_k} (Ax_k - b) = \lambda_0 - \frac{1}{\beta_0} (Ax_0 - b) \quad \text{for all } k \in \mathbb{N}, \quad (3.20)$$

and the estimate (3.11) is derived as below

$$\begin{aligned} \|Ax_k - b\| &= \beta_k \|\lambda_k - \lambda_0 + \beta_0^{-1} (Ax_0 - b)\| \leq \beta_k \|\lambda_k - \lambda_0\| + \frac{\beta_k}{\beta_0} \|Ax_0 - b\| \\ &\leq \beta_k \|\lambda_k - \lambda^*\| + \beta_k \|\lambda_0 - \lambda^*\| + \frac{\beta_k}{\beta_0} \|Ax_0 - b\| \leq \frac{\beta_k}{\beta_0} \mathcal{R}_0. \end{aligned}$$

In addition, it is clear that

$$0 \leq \mathcal{L}(x_k, \lambda^*) - \mathcal{L}(x^*, \lambda_k) = f(x_k) - f(x^*) + \langle \lambda^*, Ax_k - b \rangle \leq \mathcal{L}(x_k, \lambda^*) - \mathcal{L}(x^*, \lambda_k),$$

and thus

$$|f(x_k) - f(x^*)| \leq \|\lambda^*\| \|Ax_k - b\| + \mathcal{L}(x_k, \lambda^*) - \mathcal{L}(x^*, \lambda_k) \leq \frac{\beta_k}{\beta_0} (\mathcal{E}_0 + \|\lambda^*\| \mathcal{R}_0).$$

This establishes (3.13) and completes the proof of this theorem. \square

4. COMPOSITE OPTIMIZATION

In this section, we move to the composite case

$$\min_{x \in \mathbb{R}^n} f(x) = h(x) + g(x) \quad \text{s.t. } Ax = b, \quad (4.1)$$

where h is L -smooth and μ -convex with $\mu \geq 0$ and g is properly closed convex (possibly nonsmooth). Instead of the fully implicit scheme (3.2), to utilize the composite structure of $f = g + h$, we adopt a semi-implicit discretization that corresponds to the operator splitting (also known as the forward-backward technique). Note also that if h is only convex but the nonsmooth part g is μ -convex, then we can consider $f = \hat{h} + \hat{g}$ with $\hat{h}(x) = h(x) + \mu/2 \|x\|^2$ and $\hat{g}(x) = g(x) - \mu/2 \|x\|^2$, which agrees with the current assumption for (4.1) and $\text{prox}_{\hat{g}}$ can be computed by prox_g (cf. [63], Sect. 2.2).

4.1. A semi-implicit primal-dual proximal gradient method

Based on (3.2), we replace $\partial_x \mathcal{L}(x_{k+1}, \lambda_{k+1})$ with $\nabla h(x_k) + \partial g(x_{k+1}) + A^\top \lambda_{k+1}$ to obtain

$$\begin{cases} v_{k+1} = x_k + \frac{x_{k+1} - x_k}{\alpha_k}, \end{cases} \quad (4.2a)$$

$$\begin{cases} \beta_{k+1} \frac{\lambda_{k+1} - \lambda_k}{\alpha_k} = \nabla_\lambda \mathcal{L}(v_{k+1}, \lambda_{k+1}), \end{cases} \quad (4.2b)$$

$$\begin{cases} \gamma_{k+1} \frac{x_{k+1} - x_k}{\alpha_k} \in -\nabla h(x_k) - \partial g(x_{k+1}) - A^\top \lambda_{k+1}, \end{cases} \quad (4.2c)$$

where the parameter system (2.2) is discretized explicitly by

$$\frac{\gamma_{k+1} - \gamma_k}{\alpha_k} = \mu - \gamma_k, \quad \frac{\beta_{k+1} - \beta_k}{\alpha_k} = -\beta_k. \quad (4.3)$$

Similarly as before, the numerical scheme (4.2) admits a primal-dual formulation:

$$\begin{cases} x_{k+1} = \underset{x \in \mathbb{R}^n}{\text{argmin}} \left\{ f(x) + \langle \nabla h(x_k) + A^\top \lambda_k, x \rangle + \frac{1}{2\beta_{k+1}} \|Ax - b\|^2 + \frac{1}{2\eta_k} \|x - \hat{x}_k\|^2 \right\}, \end{cases} \quad (4.4a)$$

$$\begin{cases} v_{k+1} = x_k + (x_{k+1} - x_k)/\alpha_k, \end{cases} \quad (4.4b)$$

$$\begin{cases} \lambda_{k+1} = \lambda_k + \alpha_k/\beta_{k+1} (Av_{k+1} - b), \end{cases} \quad (4.4c)$$

where $\eta_k = \alpha_k/\gamma_{k+1}$ and $\hat{x}_k = x_k + \eta_k/\beta_k A^\top (Ax_k - b)$. In (4.4a), the smooth part h has been linearized while the nonsmooth part g uses implicit discretization. This is analogous to the proximal gradient method [7, 63], and we have to impose proper restriction on the step size α_k (see Algorithm 2).

Notice also that the subproblem (4.4a) with respect to the primal variable x_{k+1} is not easy to solve. From (4.2c) we have $x_{k+1} = \text{prox}_{\eta_k g}(x_k - \eta_k \nabla h(x_k) - \eta_k A^\top \lambda_{k+1})$, and putting this into (4.4c) gives

$$\beta_{k+1} \lambda_{k+1} - A \text{prox}_{\eta_k g}(y_k - \eta_k A^\top \lambda_{k+1}) = z_k, \quad (4.5)$$

where $y_k = x_k - \eta_k \nabla h(x_k)$ and $z_k = \beta_{k+1} (\lambda_k - \beta_k^{-1} (Ax_k - b)) - b$. Below, we present a semi-smooth Newton method to solve the nonlinear equation (4.5) in terms of the multiplier λ_{k+1} . This can be very efficient for some practical cases that (i) the multiplier has lower dimension than the primal variable; (ii) the problem (4.5) itself possesses some nice properties such as semi-smoothness and simple closed proximal formulation of g ; (iii) efficient iterative methods for updating the Newton direction can be considered.

4.1.1. A semi-smooth Newton method for the subproblem (4.5)

Define a mapping $F_k : \mathbb{R}^m \rightarrow \mathbb{R}^m$ by that

$$F_k(\lambda) := \beta_{k+1} \lambda - \text{Prox}_{\eta_k g} (y_k - \eta_k A^\top \lambda) - z_k \quad \forall \lambda \in \mathbb{R}^m. \quad (4.6)$$

Then (4.5) is equivalent to $F_k(\lambda_{k+1}) = 0$. By Moreau's identity (cf. [5], Thm. 6.45)

$$\text{prox}_{\eta g}(x) + \eta \text{prox}_{g^*/\eta}(x/\eta) = x, \quad (4.7)$$

where g^* denotes the conjugate function of g , we find that $F_k(\lambda) = \nabla \mathcal{F}_k(\lambda)$, with $\mathcal{F}_k(\cdot)$ being defined by

$$\mathcal{F}_k(\lambda) := \frac{\beta_{k+1}}{2} \|\lambda\|^2 - \langle z_k, \lambda \rangle + g^* \left(\text{prox}_{g^*/\eta_k} (y_k/\eta_k - A^\top \lambda) \right) + \frac{1}{2\eta_k} \|\text{prox}_{\eta_k g}(y_k - \eta_k A^\top \lambda)\|^2. \quad (4.8)$$

Let $\partial \text{prox}_{\eta_k g}(\lambda)$ be the generalized Clarke subdifferential [24] of the Lipschitz mapping $\text{prox}_{\eta_k g}(\lambda)$. If $P_k(\lambda) \in \partial \text{prox}_{\eta_k g}(y_k - \eta_k A^\top \lambda)$ is symmetric (this is indeed true when g is either the indicator function or the support function for some nonempty convex polyhedral [38]), then for any $\lambda \in \mathbb{R}^m$ we can define an SPD matrix

$$JF_k(\lambda) := \beta_{k+1} I + \eta_k A P_k(\lambda) A^\top \in \mathbb{R}^{m \times m}. \quad (4.9)$$

The semi-smooth Newton (SsN) method for solving (4.5) reads as follows: given an initial guess $\lambda^0 \in \mathbb{R}^m$, do the iteration

$$\lambda^{j+1} = \lambda^j - [JF_k(\lambda^j)]^{-1} F_k(\lambda^j), \quad j \geq 0. \quad (4.10)$$

Theoretically, it possesses local superlinear convergence provided that F_k is semismooth [66, 67]. Practically, it can be terminated under some suitable criterion and for global convergence, a line search procedure [27] shall be supplemented: given a Newton direction $d^j = -[JF_k(\lambda^j)]^{-1} F_k(\lambda^j)$ at step j , find the smallest nonnegative integer $r \in \mathbb{N}$ such that

$$\mathcal{F}_k(\lambda^j + \delta^r d^j) \leq \mathcal{F}_k(\lambda^j) + \nu \delta^r \langle F_k(\lambda^j), d^j \rangle, \quad (4.11)$$

where $\nu \in (0, 1/2)$, $\delta \in (0, 1]$ and $\mathcal{F}_k(\cdot)$ has been defined in (4.8). Generally the inverse operation $[JF_k(\lambda^j)]^{-1}$ in (4.10) shall be approximated by some iterative process such as the (preconditioned) conjugate gradient method [69]. For more discussions about the linear solver, we refer to Section 5.1.2.

Below we summarize the semi-implicit scheme (4.2) in Algorithm 2, which is called the *semi-implicit primal-dual proximal gradient* (Semi-PDPG) method. As suggested later by Theorem 4.3, the step size α_k is determined simply by $\alpha_k(L + \gamma_{k+1}) = \gamma_{k+1}$, which promises the convergence rate $\mathcal{O}(\min\{L/k, (1 + \mu/L)^{-k}\})$ (cf. (4.16)).

4.2. Proof of the convergence rate

To move on, the following two lemmas are needed.

Algorithm 2 Semi-PDPG method for (4.1) with h being L -smooth and μ -convex ($\mu \geq 0$)

Require: $\gamma_0 > 0$, $\beta_0 > 0$, $x_0 \in \mathbb{R}^n$, $\lambda_0 \in \mathbb{R}^m$.

- 1: **for** $k = 0, 1, \dots$ **do**
- 2: Set $\sigma_k = L + 2\gamma_k - \mu$ and $\Delta_k = \sigma_k + \sqrt{\sigma_k^2 + 4\gamma_k(\mu - \gamma_k)}$.
- 3: Compute the step size $\alpha_k = 2\gamma_k/\Delta_k \in (0, 1)$.
- 4: Update $\beta_{k+1} = \beta_k(1 - \alpha_k)$ and $\gamma_{k+1} = \mu\alpha_k + (1 - \alpha_k)\gamma_k$.
- 5: Set $\eta_k = \alpha_k/\gamma_{k+1}$ and $y_k = x_k - \eta_k \nabla h(x_k)$.
- 6: Set $z_k = \beta_{k+1}(\lambda_k - \beta_k^{-1}(Ax_k - b)) - b$.
- 7: Solve λ_{k+1} from (4.5) via the SsN iteration (4.10) with the line search procedure (4.11).
- 8: Update $x_{k+1} = \text{prox}_{\eta_k g}(y_k - \eta_k A^\top \lambda_{k+1})$.
- 9: **end for**

Lemma 4.1. Assume h is L -smooth and μ -convex with $\mu \geq 0$ and g is properly closed convex. Let $\{(x_k, \lambda_k, \gamma_k, \beta_k)\}$ be generated by (4.2) and (4.3), then for all $y \in \mathbb{R}^n$,

$$\mathcal{L}(x_{k+1}, \lambda_{k+1}) - \mathcal{L}(y, \lambda_{k+1}) + \frac{\gamma_{k+1}}{\alpha_k} \langle x_{k+1} - x_k, x_k - y \rangle \leq -\frac{\mu}{2} \|y - x_k\|^2 + \frac{L\alpha_k - 2\gamma_{k+1}}{2\alpha_k} \|x_{k+1} - x_k\|^2. \quad (4.12)$$

Proof. Define $\phi(x) := h(x) + \langle \lambda_{k+1}, Ax - b \rangle$ for all $x \in \mathbb{R}^n$. As h is L -smooth and μ -convex, there holds that

$$\begin{aligned} \phi(x_k) - \phi(y) + \langle \nabla \phi(x_k), y - x_k \rangle &\leq -\frac{\mu}{2} \|y - x_k\|^2, \\ \phi(x_{k+1}) - \phi(x_k) - \langle \nabla \phi(x_k), x_{k+1} - x_k \rangle &\leq \frac{L}{2} \|x_{k+1} - x_k\|^2. \end{aligned}$$

In addition, by (4.2c), we have

$$\gamma_{k+1} \frac{x_k - x_{k+1}}{\alpha_k} - \nabla \phi(x_k) \in \partial g(x_{k+1}),$$

and it follows that

$$\begin{aligned} g(x_{k+1}) - g(y) &\leq \left\langle \gamma_{k+1} \frac{x_k - x_{k+1}}{\alpha_k} - \nabla \phi(x_k), x_{k+1} - y \right\rangle \\ &= \frac{\gamma_{k+1}}{\alpha_k} \langle x_k - x_{k+1}, x_k - y \rangle - \langle \nabla \phi(x_k), x_{k+1} - y \rangle - \frac{\gamma_{k+1}}{\alpha_k} \|x_{k+1} - x_k\|^2. \end{aligned}$$

Collecting the above estimates and using the fact $\mathcal{L}(\cdot, \lambda_{k+1}) = \phi(\cdot) + g(\cdot)$, we obtain (4.12) and conclude the proof. \square

Recall γ_{\min} defined in (2.11) and for later use we set $\gamma_{\max} := \max\{\gamma_0, \mu\}$.

Lemma 4.2. Let $\{(\gamma_k, \beta_k)\}$ be defined by (4.3) with $\alpha_k(L + \gamma_{k+1}) \leq 2\gamma_{k+1}$, then $\alpha_k \in (0, 1]$ for all $k \in \mathbb{N}$. Moreover, if $\alpha_k(L + \gamma_{k+1}) = \gamma_{k+1}$ then

$$\prod_{i=0}^{k-1} (1 - \alpha_i) \leq \min \left\{ \frac{L + \gamma_{\max}}{\gamma_0 k + L + \gamma_{\max}}, \left(\frac{L}{L + \gamma_{\min}} \right)^k \right\}. \quad (4.13)$$

Proof. Let us first verify the existence of the sequence $\{\alpha_k\} \subset (0, 1]$. As $\alpha_k(L + \gamma_{k+1}) \leq 2\gamma_{k+1}$ and $\gamma_{k+1} = \gamma_k + \alpha_k(\mu - \gamma_k)$ (cf. (4.3)), we obtain $\psi_k(\alpha_k) := (\mu - \gamma_k)\alpha_k^2 + (L + 3\gamma_k - 2\mu)\alpha_k - 2\gamma_k \leq 0$. As $\gamma_0 > 0$, we have

$\psi_0(0) = -2\gamma_0 < 0$ and $\psi_0(1) = L - \mu \geq 0$. Hence, there must be at least one (actually unique) root $\alpha^* \in (0, 1]$ of $\psi_0(\alpha) = 0$. Hence, any $\alpha_0 \in (0, \alpha^*]$ satisfies $\alpha_0(L + \gamma_1) \leq 2\gamma_1$. Repeating this process for $\psi_k(\alpha)$ and noticing that $\gamma_k > 0$ yield the existence of $\alpha_k \in (0, 1]$ for all $k \geq 1$.

From (4.3) we have $\beta_k = \beta_0 \prod_{i=0}^{k-1} (1 - \alpha_i)$. It remains to investigate the asymptotic decay behavior of β_k with $\alpha_k(L + \gamma_{k+1}) = \gamma_{k+1}$. To do this, let us start from the identity

$$\frac{1}{\beta_{k+1}} - \frac{1}{\beta_k} = \frac{\beta_k - \beta_{k+1}}{\beta_k \beta_{k+1}} = \frac{\alpha_k}{\beta_{k+1}},$$

which indicates

$$\frac{\gamma_{k+1}}{\gamma_k} \geq 1 - \alpha_k = \frac{\beta_{k+1}}{\beta_k} \implies \gamma_k \geq \frac{\gamma_0}{\beta_0} \beta_k.$$

Besides, it follows from this and the relation $\alpha_k(L + \gamma_{k+1}) = \gamma_{k+1}$ that

$$\frac{1}{\beta_{k+1}} - \frac{1}{\beta_k} \geq \frac{\gamma_0 \alpha_k}{\beta_0 \gamma_{k+1}} = \frac{\gamma_0}{\beta_0(L + \gamma_{k+1})} \geq \frac{\gamma_0}{\beta_0(L + \gamma_{\max})}.$$

Hence, we get

$$\frac{\beta_k}{\beta_0} \leq \frac{L + \gamma_{\max}}{\gamma_0 k + L + \gamma_{\max}}. \quad (4.14)$$

On the other hand, since $\gamma_{k+1} \geq \gamma_{\min}$, we have $\alpha_k = \gamma_{k+1}/(L + \gamma_{k+1}) \geq \gamma_{\min}/(L + \gamma_{\min})$. Therefore, another bound follows immediately

$$\frac{\beta_k}{\beta_0} = \prod_{i=0}^{k-1} (1 - \alpha_i) \leq \left(\frac{L}{L + \gamma_{\min}} \right)^k.$$

Combining this with the previous estimate (4.14) establishes (4.13) and completes the proof of this lemma. \square

We now prove the convergence rate of Algorithm 2 by using the discrete Lyapunov function (3.9).

Theorem 4.3. *Assume h is L -smooth and μ -convex with $\mu \geq 0$ and g is properly closed convex. Let $\{(x_k, \lambda_k, \gamma_k, \beta_k)\}$ be generated by (4.2) and (4.3) with $\alpha_k(L + \gamma_{k+1}) \leq 2\gamma_{k+1}$, then we have $\{\alpha_k\} \subset (0, 1]$ and*

$$\mathcal{E}_{k+1} - \mathcal{E}_k \leq -\alpha_k \mathcal{E}_k, \quad \text{for all } k \in \mathbb{N}. \quad (4.15)$$

Moreover, if $\alpha_k(L + \gamma_{k+1}) = \gamma_{k+1}$, then

$$\mathcal{L}(x_k, \lambda^*) - \mathcal{L}(x^*, \lambda_k) + |F(x_k) - F(x^*)| + \|Ax_k - b\| \leq C_0 \times \min \left\{ \frac{L + \gamma_{\max}}{\gamma_0 k + L + \gamma_{\max}}, \left(\frac{L}{L + \gamma_{\min}} \right)^k \right\}, \quad (4.16)$$

where $C_0 := \mathcal{E}_0 + \mathcal{R}_0(1 + \|\lambda^*\|)$ with $\mathcal{R}_0 := \sqrt{2\beta_0 \mathcal{E}_0} + \beta_0 \|\lambda_0 - \lambda^*\| + \|Ax_0 - b\|$.

Proof. The existence of the step size sequence $\{\alpha_k\} \subset (0, 1]$ has been proved in Lemma 4.2. Once the contraction (4.15) is established, we obtain $\mathcal{E}_k \leq \mathcal{E}_0 \times \prod_{i=0}^{k-1} (1 - \alpha_i)$, and the estimate (4.16) can be obtained by using Lemma 4.2 and the same procedure for proving (3.11), (3.12) and (3.13).

Following the proof of Theorem 3.2, we start from the difference $\mathcal{E}_{k+1} - \mathcal{E}_k = I_1 + I_2 + I_3$, where I_1 , I_2 and I_3 are defined in (3.14). By (4.3), we have

$$\begin{aligned} I_2 &= \frac{\beta_{k+1} - \beta_k}{2} \|\lambda_k - \lambda^*\|^2 + \frac{\beta_{k+1}}{2} \left(\|\lambda_{k+1} - \lambda^*\|^2 - \|\lambda_k - \lambda^*\|^2 \right) \\ &= -\frac{\alpha_k \beta_k}{2} \|\lambda_k - \lambda^*\|^2 + \beta_{k+1} \langle \lambda_{k+1} - \lambda_k, \lambda_{k+1} - \lambda^* \rangle - \frac{\beta_{k+1}}{2} \|\lambda_{k+1} - \lambda_k\|^2. \end{aligned}$$

Plugging (4.2b) into the second term and dropping the last negative term lead to

$$I_2 \leq -\frac{\alpha_k \beta_k}{2} \|\lambda_k - \lambda^*\|^2 + \alpha_k \langle Av_{k+1} - b, \lambda_{k+1} - \lambda^* \rangle. \quad (4.17)$$

Similarly, for I_3 , it holds that

$$\begin{aligned} I_3 &= \frac{\gamma_{k+1} - \gamma_k}{2} \|x_k - x^*\|^2 + \frac{\gamma_{k+1}}{2} \left(\|x_{k+1} - x^*\|^2 - \|x_k - x^*\|^2 \right) \\ &= \frac{\alpha_k(\mu - \gamma_k)}{2} \|x_k - x^*\|^2 + \gamma_{k+1} \langle x_{k+1} - x_k, x_k - x^* \rangle + \frac{\gamma_{k+1}}{2} \|x_{k+1} - x_k\|^2, \end{aligned}$$

and invoking Lemma 4.2 gives

$$I_3 \leq \alpha_k (\mathcal{L}(x^*, \lambda_{k+1}) - \mathcal{L}(x_{k+1}, \lambda_{k+1})) - \frac{\alpha_k \gamma_k}{2} \|x_k - x^*\|^2 + \frac{L\alpha_k - \gamma_{k+1}}{2} \|x_{k+1} - x_k\|^2.$$

To match the right hand side of (4.15), we shift (x_{k+1}, λ_{k+1}) to (x_k, λ_{k+1}) and then to (x_k, λ^*) and obtain that

$$\begin{aligned} I_3 &\leq \alpha_k (\mathcal{L}(x^*, \lambda_k) - \mathcal{L}(x_k, \lambda^*)) - \frac{\alpha_k \gamma_k}{2} \|x_k - x^*\|^2 \\ &\quad - \alpha_k \langle Ax_k - b, \lambda_{k+1} - \lambda^* \rangle + \frac{L\alpha_k - \gamma_{k+1}}{2} \|x_{k+1} - x_k\|^2 \\ &\quad + \alpha_k (\mathcal{L}(x_k, \lambda_{k+1}) - \mathcal{L}(x_{k+1}, \lambda_{k+1})). \end{aligned}$$

To offset the last term in the above estimate, we shall divide I_1 as follows

$$\begin{aligned} I_1 &= \mathcal{L}(x_{k+1}, \lambda^*) - \mathcal{L}(x_k, \lambda^*) \\ &= \alpha_k (\mathcal{L}(x_{k+1}, \lambda_{k+1}) - \mathcal{L}(x_k, \lambda_{k+1})) - \langle Ax_{k+1} - Ax_k, \lambda_{k+1} - \lambda^* \rangle \\ &\quad + (1 - \alpha_k) (\mathcal{L}(x_{k+1}, \lambda_{k+1}) - \mathcal{L}(x_k, \lambda_{k+1})). \end{aligned}$$

Applying Lemma 4.2 again implies

$$\begin{aligned} I_1 &\leq \alpha_k (\mathcal{L}(x_{k+1}, \lambda_{k+1}) - \mathcal{L}(x_k, \lambda_{k+1})) - \langle Ax_{k+1} - Ax_k, \lambda_{k+1} - \lambda^* \rangle \\ &\quad + \frac{1 - \alpha_k}{2\alpha_k} (L\alpha_k - 2\gamma_{k+1}) \|x_{k+1} - x_k\|^2, \end{aligned}$$

which together with the relation $v_{k+1} = x_k + (x_{k+1} - x_k)/\alpha_k$ yields that

$$\begin{aligned} I_1 + I_3 &\leq \alpha_k (\mathcal{L}(x^*, \lambda_k) - \mathcal{L}(x_k, \lambda^*)) - \frac{\alpha_k \gamma_k}{2} \|x_k - x^*\|^2 - \alpha_k \langle Av_{k+1} - b, \lambda_{k+1} - \lambda^* \rangle \\ &\quad + \frac{\alpha_k (L + \gamma_{k+1}) - 2\gamma_{k+1}}{2\alpha_k} \|x_{k+1} - x_k\|^2. \end{aligned} \quad (4.18)$$

Consequently, combining this with the estimate (4.17) for I_2 implies

$$\mathcal{E}_{k+1} - \mathcal{E}_k \leq -\alpha_k \mathcal{E}_k + \frac{\alpha_k(L + \gamma_{k+1}) - 2\gamma_{k+1}}{2\alpha_k} \|x_{k+1} - x_k\|^2,$$

where the last term is negative since $\alpha_k(L + \gamma_{k+1}) \leq 2\gamma_{k+1}$. This establishes (4.15) and completes the proof. \square

5. NUMERICAL EXPERIMENTS

In this part, we investigate practical performances of Algorithms 1 and 2 for the l_1 - l_2 minimization (5.1) and the total-variation based image denoising model (5.10).

5.1. The l_1 - l_2 minimization

We first consider the linearly constrained l_1 - l_2 minimization:

$$\min_{x \in \mathbb{R}^n} \frac{\rho}{2} \|x\|^2 + \|x\|_1 \quad \text{s.t. } Ax = b, \quad (5.1)$$

where $\rho > 0$, $b \in \mathbb{R}^m$ and $A \in \mathbb{R}^{m \times n}$ with $m \ll n$. This is a regularized model for the so-called basis pursuit [21], which corresponds to the limit case $\rho = 0$ and is related to compressed sensing [14].

For $g(x) = \|x\|_1$ and any $\eta > 0$, the proximal mapping $y = \mathbf{prox}_{\eta g}(x) = \text{sgn}(x) \odot \max\{|x| - \eta, 0\}$ is well known as the *soft thresholding operator*, with the i -th component of y being given by $y_i = \text{sgn}(x_i) \max\{|x_i| - \eta, 0\}$. Here and in what follows, \odot and \oslash stand respectively for element-wise multiplication and division operations. The conjugate function g^* of g is the indicator function of the cube $[-1, 1]^n$ and thus $\mathbf{prox}_{\eta g^*}(x) = \min\{\max\{x, -1\}, 1\}$.

5.1.1. Comparison with ALB

There are some well-known Bregman iterative methods for solving (5.1); see [12, 45, 48, 86]. Both of the two accelerated variants in [45, 48] possess the nonergodic sublinear rate $\mathcal{O}(1/k^2)$ for the dual objective but the method in [48] involves a subproblem for the primal variable. In contrast, the accelerated linearized Bregman (ALB) method in [45] linearizes the augmented term and admits closed update formulation in each step. More precisely, it reads as follows: given $\lambda_0, \tilde{\lambda}_0 \in \mathbb{R}^m$, do the iteration

$$\begin{cases} x_{k+1} = \mathbf{prox}_{g/\rho}(-A^\top \tilde{\lambda}_k / \rho), \\ \lambda_{k+1} = \tilde{\lambda}_k + \tau(Ax_{k+1} - b), \\ \tilde{\lambda}_{k+1} = t_k \lambda_{k+1} + (1 - t_k) \lambda_k, \end{cases} \quad (5.2)$$

where $t_k = (2k + 3)/(k + 3)$ and $\tau = \rho / \|A\|^2$.

We apply Algorithm 2 to the problem (5.1). In this case, as g is piecewise affine, $\mathbf{prox}_{\eta g}$ is strongly semismooth [32] and so is the nonlinear mapping $F_k(\cdot)$ defined by (4.6). For $\eta > 0$ and $x \in \mathbb{R}^n$, define a diagonal matrix

$$P_\eta(x) = \text{diag}(p) \in \mathbb{R}^{n \times n} \quad \text{with } p_i = \begin{cases} 1 & \text{if } |x_i| \geq \eta, \\ 0 & \text{if } |x_i| < \eta. \end{cases} \quad (5.3)$$

Then it is easy to see that $P_\eta(x) \in \partial \mathbf{prox}_{\eta g}(x)$, and we obtain a generalized Clarke subgradient for (4.5):

$$JF_k(\lambda) = \beta_{k+1} I + \eta_k A P_{\eta_k} [v_k(\lambda)] A^\top \in \mathbb{R}^{m \times m}, \quad (5.4)$$

where $v_k(\lambda) = y_k - \eta_k A^\top \lambda$. Note that $P_{\eta_k}[v_k(\lambda)] = \text{diag}(p)$ where p is defined by (5.3), and thus $JF_k(\lambda)$ is always SPD. Moreover, the function (4.8) becomes

$$\mathcal{F}_k(\lambda) = \frac{\beta_{k+1}}{2} \|\lambda\|^2 - \langle z_k, \lambda \rangle + \frac{1}{2\eta_k} \|\mathbf{prox}_{\eta_k g}[v_k(\lambda)]\|^2.$$

Algorithm 3 Inexact Semi-PDPG method for the l_1-l_2 minimization problem (5.1)

Require: $\gamma_0 > 0$, $\beta_0 > 0$, $x_0 \in \mathbb{R}^n$ and $\lambda_0 \in \mathbb{R}^m$.

- 1: Problem setting: $\rho > 0$, $b \in \mathbb{R}^m$ and $A \in \mathbb{R}^{m \times n}$.
- 2: SsN setting: $\nu = 0.2$, $\delta = 0.9$ and $j_{\max} = 10$.
- 3: Tolerances: $\text{KKT_Tol} = 10^{-6}$ and $\text{SsN_Tol} = 10^{-8}$.
- 4: **for** $k = 0, 1, \dots$ **do**
- 5: Set $\sigma_k = 2\gamma_k$ and $\Delta_k = \sigma_k + \sqrt{\sigma_k^2 + 4\gamma_k(\rho - \gamma_k)}$.
- 6: Compute the step size $\alpha_k = 2\gamma_k/\Delta_k \in (0, 1)$.
- 7: Update $\beta_{k+1} = \beta_k(1 - \alpha_k)$ and $\gamma_{k+1} = \rho\alpha_k + (1 - \alpha_k)\gamma_k$.
- 8: Set $\eta_k = \alpha_k/\gamma_{k+1}$ and $y_k = x_k - \eta_k \rho x_k$.
- 9: Set $z_k = \beta_{k+1}(\lambda_k - \beta_k^{-1}(Ax_k - b)) - b$.
- 10: Solve λ_{k+1} from the nonlinear equation

$$F_k(\lambda) := \beta_{k+1}\lambda - A\mathbf{prox}_{\eta_k g}(y_k - \eta_k A^\top \lambda) - z_k = 0 \quad (5.5)$$

via the following SsN iteration with $\lambda = \lambda_k$ and $j = 0$:

- 11: **while** $\|F_k(\lambda)\| > \text{SsN_Tol}$ and $j < j_{\max}$ **do** {SsN iteration}
- 12: Compute $v_k = y_k - \eta_k A^\top \lambda$.
- 13: Find $P_{\eta_k}(v_k) \in \partial \mathbf{prox}_{\eta_k g}(v_k)$ via (5.3).
- 14: Compute $JF_k(\lambda) = \beta_{k+1}I + \eta_k A P_{\eta_k}(v_k) A^\top$.
- 15: Solve $JF_k(\lambda)d = -F_k(\lambda)$.
- 16: Find the smallest integer $r \in \mathbb{N}_+$ such that $\mathcal{F}_k(\lambda + \delta^r d) \leq \mathcal{F}_k(\lambda) + \nu \delta^r \langle F_k(\lambda), d \rangle$.
- 17: Update $\lambda = \lambda + \delta^r d$ and $j = j + 1$.
- 18: **end while**
- 19: Update $\lambda_{k+1} = \lambda$ and $x_{k+1} = \mathbf{prox}_{\eta_k g}(y_k - \eta_k A^\top \lambda_{k+1})$.
- 20: **if** $\text{Res}(k) \leq \text{KKT_Tol}$ **then**
- 21: **break**
- 22: **end if**
- 23: **end for**

We rewrite Algorithm 2 in Algorithm 3, where a practical inexact setting is considered. The SsN iteration (cf. lines 11–18 in Algorithm 3) is stopped either $\|F_k(\lambda)\| \leq \text{SsN_Tol} = 10^{-8}$ or $j_{\max} = 10$. For the line search procedure, we adopt $\nu = 0.2$ and $\delta = 0.9$. All initial guesses β_0 , x_0 and λ_0 are generated randomly, and we chose $\gamma_0 = \mu + \sigma$ with σ obeying the uniform distribution on $[0, 1]$. By Theorem 4.3 we the linear rate 2^{-k} (with exact computation).

Recall the optimality condition of problem (5.1): $Ax^* = b$ and $x^* = \mathbf{prox}_g((1 - \rho)x^* - A^\top \lambda^*)$. Hence, we consider the stopping criterion:

$$\text{Res}(k) := \max \{\text{Res}(x_k), \text{Res}(\lambda_k)\} \leq \text{KKT_Tol} = 10^{-6}, \quad (5.6)$$

TABLE 1. Performances of inexact semi-PDPG (*i.e.*, Algorithm 3) and ALB method (5.2) for solving (5.1). Here, “direct” and “PCG” mean that the linear system in step 15 of Algorithm 3 is solved respectively by direct method and PCG.

	m	n	Inexact Semi-PDPG(direct)				Inexact Semi-PDPG(PCG)				ALB	
			its	SsN	time(sec)	its	SsN	time (s)	its	time(sec)	its	time(sec)
$\rho = 0.5$	5e+02	2e+03	21	42	5.20	21	40	3.59	537	4.20		
	8e+02	3e+03	21	46	10.76	21	43	11.40	593	10.86		
	1e+03	4e+03	21	39	12.33	21	42	12.37	546	16.15		
$\rho = 0.1$	2e+02	1e+03	20	34	0.70	20	43	1.08	2330	2.68		
	5e+02	3e+03	21	37	3.66	19	51	5.66	1967	23.81		
	1e+03	5e+03	20	43	15.61	20	47	14.00	2118	81.83		
$\rho = 0.01$	5e+02	2e+03	19	56	4.50	18	60	5.92	13174	103.54		
	9e+02	4e+03	18	56	12.49	22	87	41.76	12712	379.83		
	2e+03	8e+03	17	63	87.29	19	82	246.34	13819	1693.99		
$\rho = 0.005$	8e+02	3e+03	21	86	23.39	19	75	39.25	19793	375.07		
	2e+03	6e+03	20	86	153.48	23	126	579.69	20811	1778.27		
	3e+03	9e+03	19	83	509.93	24	139	1933.28	21568	6592.60		

where the relative KKT residuals are defined by

$$\text{Res}(\lambda_k) := \frac{\|Ax_k - b\|}{1 + \|b\|} \quad \text{and} \quad \text{Res}(x_k) := \frac{\|x_k - \text{prox}_g((1 - \rho)x_k - A^\top \lambda_k)\|}{1 + \|x_k\|}.$$

In step 15 of Algorithm 3, we have to solve a linear system and we consider two ways: one is direct method as $m \ll n$ and the other is preconditioned conjugate gradient (PCG) method (*cf.* [69], Algorithm 9.1) with diagonal preconditioner. The PCG iteration is stopped either the relative residual is smaller than 10^{-8} or the maximal iteration number 5000 is attained.

Computational results are reported in Table 1, which includes (i) **its**: the number of iterations, (ii) **SsN**: the total number of the SsN iterations for the inner problem (5.5), and (iii) **time**: the running time (in seconds). To achieve the tolerance (5.6), the number of iterations of Algorithm 3 is almost $k^* = 6 \ln 10 / \ln 2 \approx 20$. This can be observed from Table 1. However, as ρ becomes small, the problem (5.1) itself is more degenerate and the number of iterations of the ALB method grows dramatically.

5.1.2. Performance of the PCG iteration

From Table 1 we see that Algorithm 3 with PCG solver is slightly inferior than that with direct solver, both for total iteration number and running time. We now investigate the performance of the PCG iteration.

The linear system arising from step 15 of Algorithm 3 is $JF_k(\lambda)d = -F_k(\lambda)$, where $JF_k(\cdot) = \beta_{k+1}I + \eta_k \mathcal{A}_0(\cdot)$ is defined by (5.4) and $\mathcal{A}_0(\cdot)$ is symmetric semi-positive definite. Note that $JF_k(\cdot)$ is always SPD but also nearly singular as $\beta_{k+1} \rightarrow 0$. Hence, the number of iterations increases as k does. Fortunately, for large k , we may expect that $F_k(\cdot)$ is close to zero (as the algorithm converges) and the nearly singular property is not a serious problem.

Recall that we used the diagonal preconditioner, *i.e.*, Jacobi iteration, and the terminal criterion is relative residual $\leq 10^{-8}$, with the maximal iteration number 5000. In every k -th step of Algorithm 3, we record the

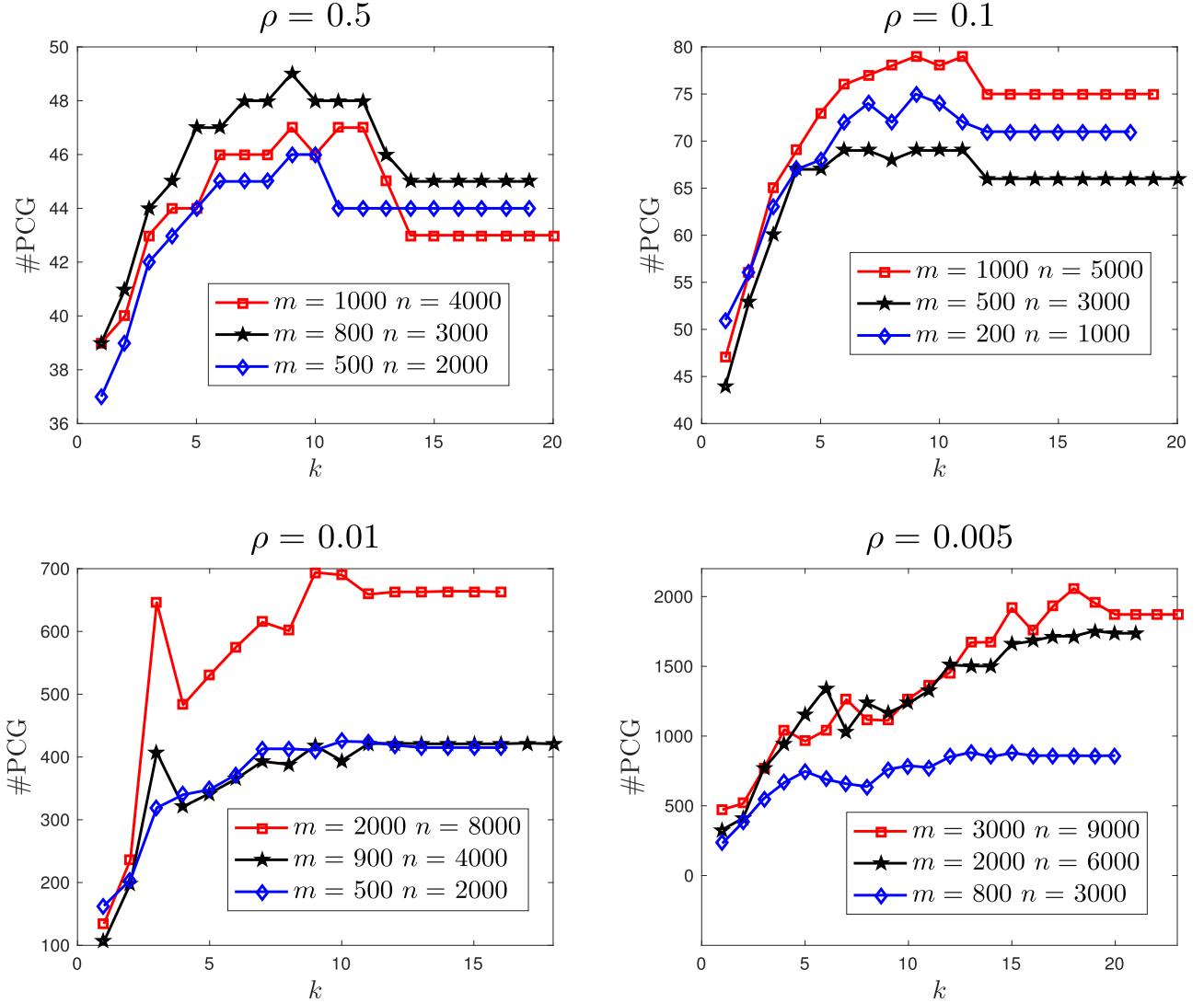


FIGURE 2. Averaged PCG iterations of Algorithm 3 for solving (5.1) with different problem size and ρ .

PCG iteration $\#_{k,j}$ of the j -th SsN iteration and obtain an averaged number $\#_k = \frac{1}{s_k} \sum_{j=1}^{s_k} \#_{k,j}$, where s_k denotes the number of SsN iterations for solving the subproblem (5.5).

In Figure 2, we plot the averaged PCG iterations of Algorithm 3 with the same problem size and ρ used in Table 1. As predicted above, due to the nearly singular property, the PCG iteration number grows up as k increases but it stays flat for large k . Moreover, it is not robust with respect to the problem size and ρ .

5.1.3. Restarting and warm-up

Note that in the few starting steps, *i.e.*, for small k , the SsN iteration may not achieve the desired tolerance $\|F_k(\lambda^j)\| \leq \text{SsN_Tol}$ within $j_{\max} = 10$ iterations and the KKT residual $\text{Res}(k)$ (*cf.* (5.6)) might not decay linearly while β_k has already attained a small number, which makes the subproblem (5.5) degenerate. Hence, to ensure the stability, we adopt the restart technique.

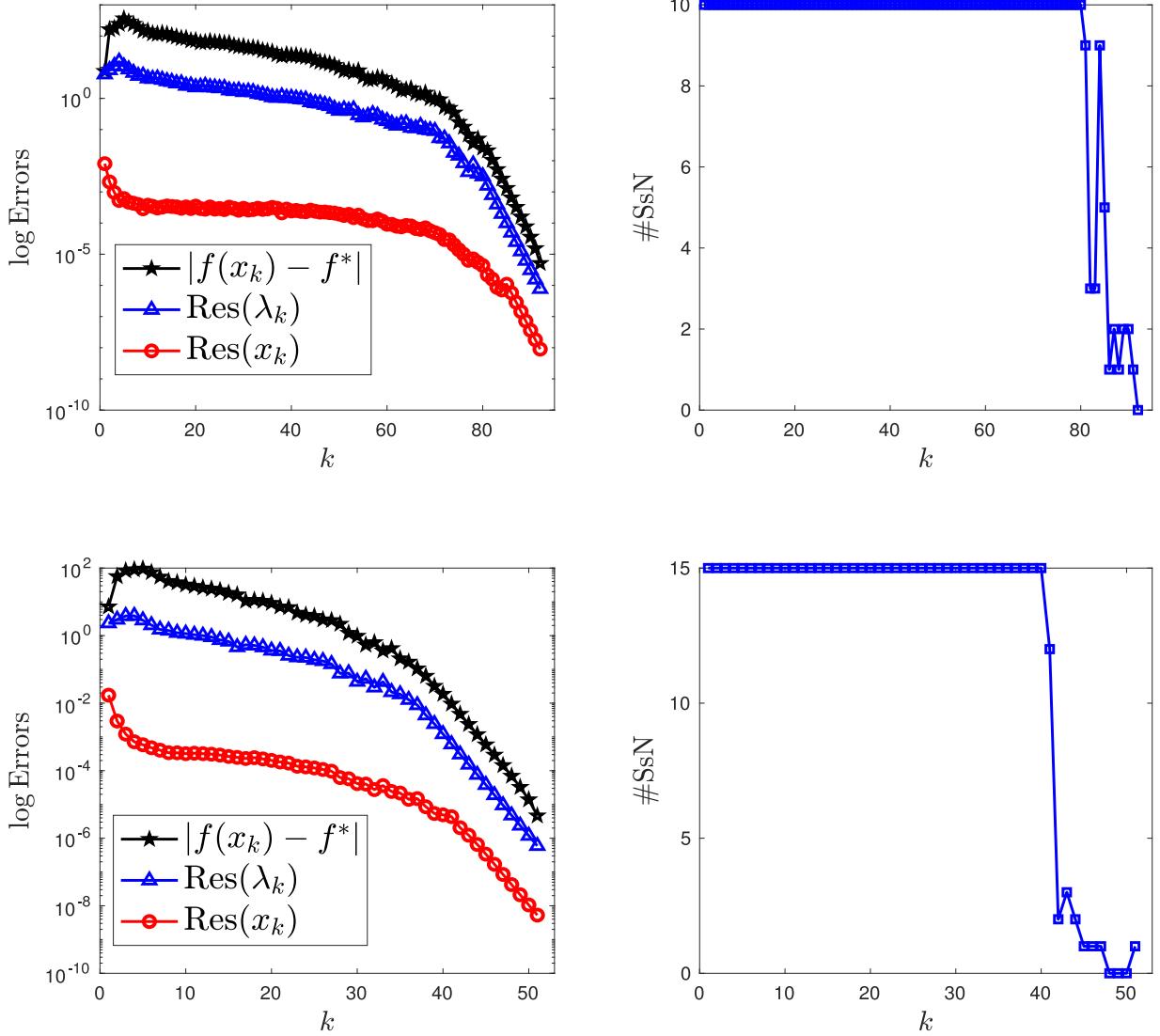


FIGURE 3. Performance of Algorithm 3 for solving (5.1) with $m = 2000$, $n = 5000$ and $\rho = 0.0005$. The maximal iteration numbers of the SsN iteration for the top row and the bottom row are $j_{\max} = 10$ and $j_{\max} = 15$, respectively. The left part plots the decay behavior of the errors and the right part shows the number of SsN iteration in each step.

For a more singular case $\rho = 0.0005$, we restart Algorithm 3 whenever $\beta_k \leq 10^{-7}$ and the KKT residual $\text{Res}(k)$ increases. From Figure 3, we observe that for this extreme case, (i) the total iteration number increases; (ii) in more than half of the total number of iterations, the errors decay slowly and the SsN iteration number attains its maximal value j_{\max} (we set $j_{\max} = 10$ for the top row and $j_{\max} = 15$ for the bottom row), but after that, fast local linear convergence arises and the number of SsN iterations decreases.

As suggested by the results in Figure 3, a warm-up procedure might improve the performance of the algorithm and we show this in Figure 4, where the initial guess is obtained by running the ALB method 500 times. This works well indeed and the convergence behavior is much better than that in Figure 3.

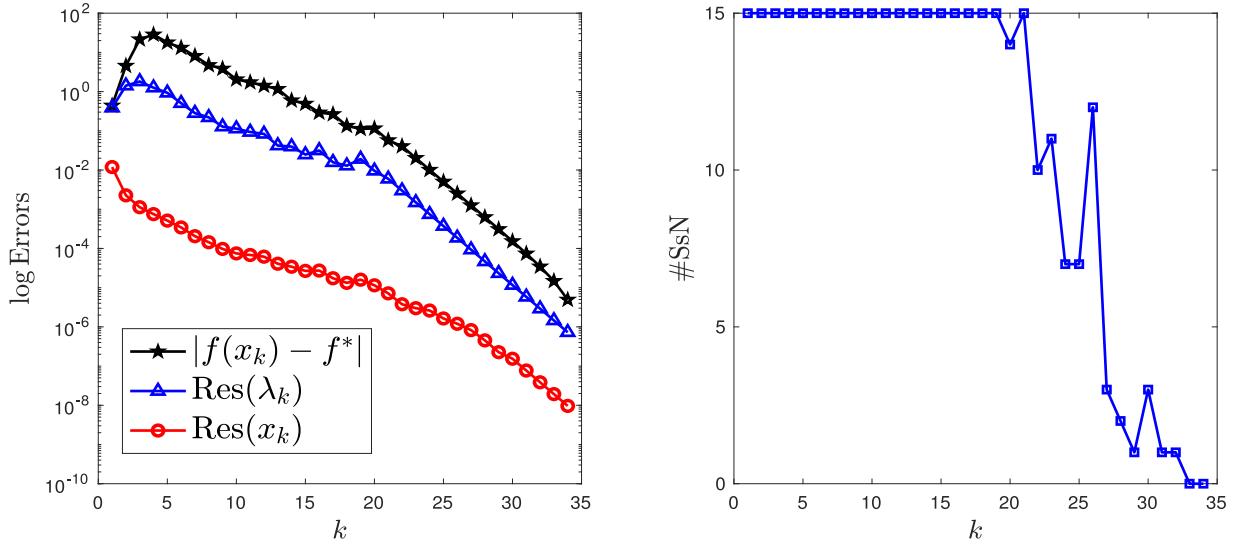


FIGURE 4. Performance of Algorithm 3 for solving (5.1) with warm-up procedure. Here, we take $m = 2000$, $n = 5000$ and $\rho = 0.0005$, and the maximal number of the SsN iteration is $j_{\max} = 15$. The initial guess is obtained *via* running the ALB method 500 times.

5.2. Total-variation based image denoising

Given a noised image $g \in L^2(\Omega)$ with the domain $\Omega \subset \mathbb{R}^2$, the total variation based denoising model proposed by Rudin, Osher and Fatemi (ROF for short) [68] reads as follows

$$\min_u \int_{\Omega} |\nabla u| \, dx + \frac{\rho}{2} \|u - g\|_{L^2(\Omega)}^2, \quad (5.7)$$

where $\rho > 0$ is the regularization parameter and $|\nabla u| := \sqrt{|\nabla_x u|^2 + |\nabla_y u|^2}$.

5.2.1. Discrete formulations

In discrete setting, problem (5.7) becomes

$$\min_{U \in \mathbb{R}^{m \times n}} \sum_{i=1}^m \sum_{j=1}^n \sqrt{|(\mathcal{D}(U))_{i,j,1}|^2 + |(\mathcal{D}(U))_{i,j,2}|^2} + \frac{\rho}{2} \|U - \Xi\|_F^2, \quad (5.8)$$

where $\Xi \in \mathbb{R}^{m \times n}$ and $\mathcal{D} : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{m \times n \times 2}$ denotes the discrete gradient operator, *i.e.*,

$$(\mathcal{D}(U))_{i,j,1} := \begin{cases} U_{i+1,j} - U_{i,j} & \text{if } i < m, \\ 0 & \text{if } i = m, \end{cases} \quad \text{and} \quad (\mathcal{D}(U))_{i,j,2} := \begin{cases} U_{i,j+1} - U_{i,j} & \text{if } j < n, \\ 0 & \text{if } j = n, \end{cases}$$

for all $1 \leq i \leq m$ and $1 \leq j \leq n$. Let $\text{vec}(\times)$ be the vector expanded by the matrix \times by its column. Then rewrite (5.8) as a composite problem

$$\min_{u \in \mathbb{R}^{mn}} \psi(Au) + \frac{\rho}{2} \|u - \xi\|^2, \quad (5.9)$$

where $\xi = \text{vec}(\Xi)$ and $A := \begin{pmatrix} I_n \otimes D_m \\ D_n \otimes I_m \end{pmatrix}$, with the difference matrices D_m and D_n being defined such that $(I_n \otimes D_m)\text{vec}(U) = \text{vec}(\mathcal{D}(U)_{i,j,1})$ and $(D_n \otimes I_m)\text{vec}(U) = \text{vec}(\mathcal{D}(U)_{i,j,2})$.

Let $\mathcal{A} = (-A, I)$ and introduce a function $\psi : \mathbb{R}^{2mn} \rightarrow \mathbb{R}$ by that

$$\psi(\mathbf{p}) := \sum_{i=1}^{mn} \sqrt{p_i^2 + q_i^2} \quad \forall \mathbf{p} = \begin{pmatrix} p \\ q \end{pmatrix} \in \mathbb{R}^{2mn}.$$

Then (5.9) can be written as the standard form (1.1):

$$\min_{X=(u, \mathbf{p})} f(X) := \frac{\rho}{2} \|u - \xi\|^2 + \psi(\mathbf{p}) \quad \text{s.t. } \mathcal{A}X = 0. \quad (5.10)$$

5.2.2. Accelerated primal-dual methods

There are some well-known accelerated primal-dual methods for solving the discrete ROF model (5.8). Here, we choose two baseline algorithms: the primal-dual hybrid gradient (PDHG) method ([16], Algorithm 2) and the accelerated alternating direction method of multipliers (A-ADMM) ([82], Algorithm 2). Ergodic convergence rate $\mathcal{O}(1/k^2)$ is achieved by those two methods. For completeness, we list them as below.

– **PDHG method ([16], Algorithm 2)** This method starts from the minimax formulation of (5.9):

$$\min_{u \in \mathbb{R}^{mn}} \max_{\boldsymbol{\lambda} \in \mathbb{R}^{2mn}} \langle Au, \boldsymbol{\lambda} \rangle + \frac{\rho}{2} \|u - \xi\|^2 - \psi^*(\boldsymbol{\lambda}), \quad (5.11)$$

where

$$\boldsymbol{\lambda} = \begin{pmatrix} v \\ w \end{pmatrix} \in \mathbb{R}^{2mn} \quad \text{and} \quad \psi^*(\boldsymbol{\lambda}) := \begin{cases} 0 & \text{if } \sqrt{v_i^2 + w_i^2} \leq 1 \text{ for all } 1 \leq i \leq mn, \\ +\infty & \text{else.} \end{cases}$$

More precisely, it reads as follows: given $\sigma_0 = 0$, $\boldsymbol{\lambda}_0 \in \mathbb{R}^{2mn}$ and $u_{-1} = u_0 \in \mathbb{R}^{mn}$, do the iteration

$$\begin{cases} \bar{u}_k = u_k + \sigma_k(u_k - u_{k-1}), \\ \boldsymbol{\lambda}_{k+1} = \text{prox}_{\theta_k \psi^*}(\boldsymbol{\lambda}_k + \theta_k A \bar{u}_k), \\ u_{k+1} = \frac{u_k - \tau_k A^\top \boldsymbol{\lambda}_{k+1}}{1 + \rho \tau_k} + \frac{\rho \tau_k \xi}{1 + \rho \tau_k}, \\ \sigma_{k+1} = 1/\sqrt{1 + 2\rho \tau_k}, \quad \tau_{k+1} = \sigma_{k+1} \tau_k, \quad \theta_{k+1} = \theta_k / \sigma_{k+1}, \end{cases} \quad (5.12)$$

where $\tau_0 \theta_0 \|A\|^2 \leq 1$ with $\|A\|^2 \leq 8$ (cf. [15]). Thanks to Moreau's identity (4.7), for all $\theta > 0$, we have $\text{prox}_{\theta \psi^*}(\boldsymbol{\lambda}) = (v \odot \sigma(\boldsymbol{\lambda}), w \odot \sigma(\boldsymbol{\lambda}))$, where $\sigma(\boldsymbol{\lambda}) := 1 - \tau(v, w)$ with $\tau(v, w)$ being defined by (5.14).

- **A-ADMM ([82], Algorithm 2)** Applying this method to problem (5.10) leads to the iteration: given $\theta \geq \|A\|^2, \lambda_0 = 0, \mathbf{p}_0 \in \mathbb{R}^{2mn}$ and $u_0 \in \mathbb{R}^{mn}$, compute

$$\begin{cases} \theta_k = \frac{2\theta}{\rho(k+1)}, \\ \mathbf{p}_{k+1} = \mathbf{prox}_{\theta_k \psi}(Au_k - \theta_k \lambda_k), \\ u_{k+1} = (\rho\theta_k I + A^\top A)^{-1} (A^\top(\mathbf{p}_{k+1} + \theta_k \lambda_k) + \rho\theta_k \xi), \\ \lambda_{k+1} = \lambda_k + \frac{1}{\theta_k} (\mathbf{p}_{k+1} - Au_{k+1}), \end{cases} \quad (5.13)$$

where $\mathbf{prox}_{\theta_k \psi}(\cdot)$ is defined by (5.15) and the inverse operation $(\rho\theta_k I + A^\top A)^{-1}$ can be realized *via* fast Fourier transform.

5.2.3. Inexact implicit primal-dual method

We apply our Algorithm 1 to problem (5.10) and obtain an inexact Im-PD method; see Algorithm 4. For clarity, we provide some details about the proximal calculations. Given $a, b \in \mathbb{R}$ and $\theta > 0$, define $\tau_\theta(a, b) \in \mathbb{R}$ and $\mathcal{T}_\theta(a, b) \in \mathbb{R}^{2 \times 2}$ respectively by that

$$\begin{aligned} \tau_\theta(a, b) &:= 1 - \frac{\theta}{\max\{\theta, \sqrt{a^2 + b^2}\}}, \\ \mathcal{T}_\theta(a, b) &:= \begin{cases} \tau_\theta(a, b)I + \frac{1 - \tau_\theta(a, b)}{a^2 + b^2} \begin{pmatrix} a^2 & ab \\ ab & b^2 \end{pmatrix} & \text{if } \sqrt{a^2 + b^2} \geq \theta, \\ O_{2 \times 2} & \text{else.} \end{cases} \end{aligned}$$

If $a, b \in \mathbb{R}^n$, then $\tau_\theta(a, b) \in \mathbb{R}^n$ can be understood as point wise operation:

$$\tau_\theta(a, b) := \mathbf{1}_n - \theta \mathbf{1}_n \oslash \max\{\theta \mathbf{1}_n, \sqrt{a \odot a + b \odot b}\}. \quad (5.14)$$

For $\theta = 1$, we simply write $\tau_\theta(a, b) = \tau(a, b)$.

For $X = (u, \mathbf{p}) \in \mathbb{R}^{3mn}$ and $\theta > 0$, the proximal mapping of f is given by $\mathbf{prox}_{\theta f}(X) = (\frac{u + \rho\theta\xi}{1 + \rho\theta}, \mathbf{prox}_{\theta \psi}(\mathbf{p}))$, where

$$\mathbf{prox}_{\theta \psi}(\mathbf{p}) = (p \odot \tau_\theta(p, q), q \odot \tau_\theta(p, q)). \quad (5.15)$$

According to Chapter 7 of [32], f is strongly semismooth and so is the nonlinear mapping $F_k(\cdot)$ defined by (5.17). Moreover, a direct computation shows that $P_\theta(X) \in \partial \mathbf{prox}_{\theta f}(X)$ where

$$P_\theta(X) := \begin{pmatrix} \frac{1}{1+\rho\theta} I & O \\ O & T \end{pmatrix} \quad \text{with} \quad T = \begin{pmatrix} \text{diag}(\tau_{11}) & \text{diag}(\tau_{12}) \\ \text{diag}(\tau_{21}) & \text{diag}(\tau_{22}) \end{pmatrix}. \quad (5.16)$$

In (5.16), T is block diagonal and $\mathcal{T}_\theta(p_i, q_i) = \begin{pmatrix} (\tau_{11})_i & (\tau_{12})_i \\ (\tau_{21})_i & (\tau_{22})_i \end{pmatrix}$ for all $1 \leq i \leq mn$. For $Y = (s, \boldsymbol{\lambda}) \in \mathbb{R}^{3mn}$, it is not hard to find that $f^*(Y) = \frac{1}{2\rho} \|s\|^2 + \langle s, \xi \rangle + \psi^*(\boldsymbol{\lambda})$, and thus the function (4.8) becomes

$$\mathcal{F}_k(\boldsymbol{\lambda}) = \frac{\beta_{k+1}}{2} \|\boldsymbol{\lambda}\|^2 - \langle Z_k, \boldsymbol{\lambda} \rangle + f^*(Y_k(\boldsymbol{\lambda})) + \frac{1}{2\theta_k} \|\mathbf{prox}_{\theta_k f}[Y_k(\boldsymbol{\lambda})]\|^2,$$

where $Z_k = \beta_{k+1}(\boldsymbol{\lambda}_k - \beta_k^{-1} \mathcal{A}X_k)$ and $Y_k(\boldsymbol{\lambda}) = X_k - \theta_k \mathcal{A}^\top \boldsymbol{\lambda}$.

As motivated by the first example (*cf.* Fig. 4), in line 3 of Algorithm 4, we consider a warm-up step to provide a reasonable initial guess $(X_0, \boldsymbol{\lambda}_0)$ and therefore enhance the performance. Besides, in step (13), the linear SPD system has special sparse structure that T_k is a 2×2 block matrix with each block being diagonal (see (5.16)) and

$$AA^\top = \begin{pmatrix} H_{11} & H_{12} \\ H_{12}^\top & H_{22} \end{pmatrix} = \begin{pmatrix} I_n \otimes D_m D_m^\top & D_n^\top \otimes D_m \\ D_m^\top \otimes D_n & D_n D_n^\top \otimes I_m \end{pmatrix},$$

where H_{11} is block diagonal and both H_{12} and H_{22} are block tridiagonal. Hence, we consider the incomplete Cholesky factorization (*cf.* [69], Chap. 10) as a preconditioner and apply preconditioned CG to step (13) to obtain an approximation with relative residual $\leq 10^{-8}$.

Algorithm 4 Inexact Im-PD method for the discrete ROF model (5.10)

Require: $\beta_0 > 0$, $\nu = 0.2$ and $\delta = 0.9$.

- 1: Problem setting: $\rho > 0$ and $\xi \in \mathbb{R}^{mn}$.
- 2: Tolerances: $\text{KKT_Tol} = 10^{-6}$ and $\text{SsN_Tol} = 10^{-8}$.
- 3: Perform a warm-up step to obtain: $X_0 = (u_0, \mathbf{p}_0) \in \mathbb{R}^{3mn}$ and $\boldsymbol{\lambda}_0 \in \mathbb{R}^{2mn}$.
- 4: **for** $k = 0, 1, \dots$ **do**
- 5: Choose the step size $\alpha_k > 0$ and update $\beta_{k+1} = \beta_k / (1 + \alpha_k)$.
- 6: Set $\theta_k = \alpha_k / \beta_k$ and $\rho_k = 1 / (1 + \rho \theta_k)$.
- 7: Set $Z_k = \beta_{k+1}(\boldsymbol{\lambda}_k - \beta_k^{-1} \mathcal{A}X_k)$.
- 8: Solve $\boldsymbol{\lambda}_{k+1}$ from the nonlinear equation

$$F_k(\boldsymbol{\lambda}) := \beta_{k+1}\boldsymbol{\lambda} - \mathcal{A}\mathbf{prox}_{\theta_k f}(X_k - \theta_k \mathcal{A}^\top \boldsymbol{\lambda}) - Z_k = 0 \quad (5.17)$$

via the following SsN iteration with the initial guess $\boldsymbol{\lambda} = \boldsymbol{\lambda}_k$:

- 9: **while** $\|F_k(\boldsymbol{\lambda})\| > \text{SsN_Tol}$ **do** {SsN iteration}
- 10: Compute $Y_k = X_k - \theta_k \mathcal{A}^\top \boldsymbol{\lambda}$.
- 11: Find $P_k(Y_k) = \begin{pmatrix} \rho_k I & O \\ O & T_k \end{pmatrix} \in \partial \mathbf{prox}_{\theta_k f}(Y_k)$ via (5.16).
- 12: Compute $JF_k(\boldsymbol{\lambda}) = \beta_{k+1}I + \theta_k T_k + \rho_k \theta_k A A^\top$.
- 13: Solve $JF_k(\boldsymbol{\lambda})\mathbf{d} = -F_k(\boldsymbol{\lambda})$ approximately via preconditioned CG.
- 14: Find the smallest integer $r \in \mathbb{N}_+$ such that $\mathcal{F}_k(\boldsymbol{\lambda} + \delta^r \mathbf{d}) \leq \mathcal{F}_k(\boldsymbol{\lambda}) + \nu \delta^r \langle F_k(\boldsymbol{\lambda}), \mathbf{d} \rangle$.
- 15: Update $\boldsymbol{\lambda} = \boldsymbol{\lambda} + \delta^r \mathbf{d}$.
- 16: **end while**
- 17: Update $\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}$ and $X_{k+1} = \mathbf{prox}_{\theta_k f}(X_k - \theta_k \mathcal{A}^\top \boldsymbol{\lambda}_{k+1})$.
- 18: **if** $\text{Res}(k) \leq \text{KKT_Tol}$ **then**
- 19: **break**
- 20: **end if**
- 21: **end for**

TABLE 2. Performances of Algorithm 4, PDHG (5.12) and A-ADMM (5.13) for solving (5.8).

	$m = n$	ρ	Inexact Im-PD (Algorithm 4)				A-ADMM (5.13)		PDHG (5.12)		
			its	SsN	warm-up(sec)	time(sec)	its	time(sec)	its	Res(k_{\max})	time(sec)
barb	512	50	10	182	47.11	830.53	1572	1497.40	10^5	5.15e-06	5169.55
		150	10	141	41.76	568.34	3445	3192.33		3.59e-06	5213.23
boat	512	40	9	84	54.53	457.05	1300	1145.84	10^5	5.49e-06	5228.70
		180	10	141	44.34	611.90	3866	3316.70		3.42e-06	5223.88
cameraman	256	20	7	52	8.48	78.91	724	124.58	10^5	7.77e-06	1299.35
		100	10	81	8.36	70.26	2575	448.94		4.10e-06	1262.76
lena	256	50	9	111	8.55	117.86	1554	288.23	10^5	5.29e-06	1229.09
		200	11	157	8.47	158.41	4099	758.60		3.55e-06	1210.58

5.2.4. Numerical results

We adopt four benchmark images from the literature: **barb**, **boat**, **cameraman** and **lena**. These images are noised with standard normal distribution. Note that both (5.11), (5.10) admit the same optimality condition

$$\begin{cases} 0 = \rho(u^* - \xi) - A^\top \boldsymbol{\lambda}^* \\ 0 \in \boldsymbol{\lambda}^* + \partial\psi(\mathbf{p}^*) \\ 0 = \mathbf{p}^* - Au^* \end{cases} \iff \begin{cases} 0 = \rho(u^* - \xi) - A^\top \boldsymbol{\lambda}^* \\ 0 = \mathbf{p}^* - \mathbf{prox}_\psi(\mathbf{p}^* - \boldsymbol{\lambda}^*) \\ 0 = \mathbf{p}^* - Au^* = 0 \end{cases}.$$

Hence, we consider the stopping criterion:

$$\text{Res}(k) := \max \{ \text{Res}(u_k), \text{Res}(\mathbf{p}_k), \text{Res}(\boldsymbol{\lambda}_k) \} \leq \text{KKT_Tol} = 10^{-6}, \quad (5.18)$$

where the relative KKT residuals are defined by

$$\text{Res}(u_k) := \frac{\|\rho(u_k - \xi) - A^\top \boldsymbol{\lambda}_k\|}{1 + \|\xi\|}, \quad \text{Res}(\mathbf{p}_k) := \frac{\|\mathbf{p}_k - \mathbf{prox}_\psi(\mathbf{p}_k - \boldsymbol{\lambda}_k)\|}{1 + \|\mathbf{p}_k\|} \quad \text{and} \quad \text{Res}(\boldsymbol{\lambda}_k) := \frac{\|\mathbf{p}_k - Au_k\|}{1 + \|\mathbf{p}_k\|}.$$

For all methods, the maximal iteration number is $k_{\max} = 1e5$. For inexact Im-PD (*i.e.* Algorithm 4), we run the A-ADMM with 50 steps to obtain an initial guess $(u_0, \mathbf{p}_0, \boldsymbol{\lambda}_0)$ with $\max \{ \text{Res}(u_0), \text{Res}(\mathbf{p}_0), \text{Res}(\boldsymbol{\lambda}_0) \} \approx 10^{-2}$ and choose the step size $\alpha_k = 1 + \sigma$ where σ obeys the uniform distribution on $[0, 1]$. Then by Theorem 3.2, we have the linear rate ϱ^k with $\varrho = \mathbb{E}[\frac{1}{1+\alpha_k}] = \int_0^1 \frac{1}{2+\sigma} d\sigma = \ln 3/2$ and the required iteration number for (5.18) is about $k^* = -4 \ln 10 / \ln \varrho \approx 10$.

Computational results are summarized in Table 2, including the number of iterations (**its**) and running time (**time**). For Inexact Im-PD, we also report the total number of SsN iterations (**SsN**) and the time used for initialization (**warm-up**). For all cases, PDHG has not achieved the tolerance (5.18) within the maximal iteration number $k_{\max} = 10^5$, and we also record the KKT residual $\text{Res}(k_{\max})$ at the last iterate. As we can see, Algorithm 4 outperforms much better than other two methods and the total iteration number is almost 10, as expected above. Particularly, we observe that A-ADMM is more efficient than PDHG.

Moreover, in Figure 5, we plot the averaged PCG iteration number of Algorithm 4 for all cases. Similar as before (*cf.* Fig. 2), it increases along with the iteration. Therefore, this deserves further study for more robust and efficient linear solvers such as algebraic multilevel methods [50, 83].

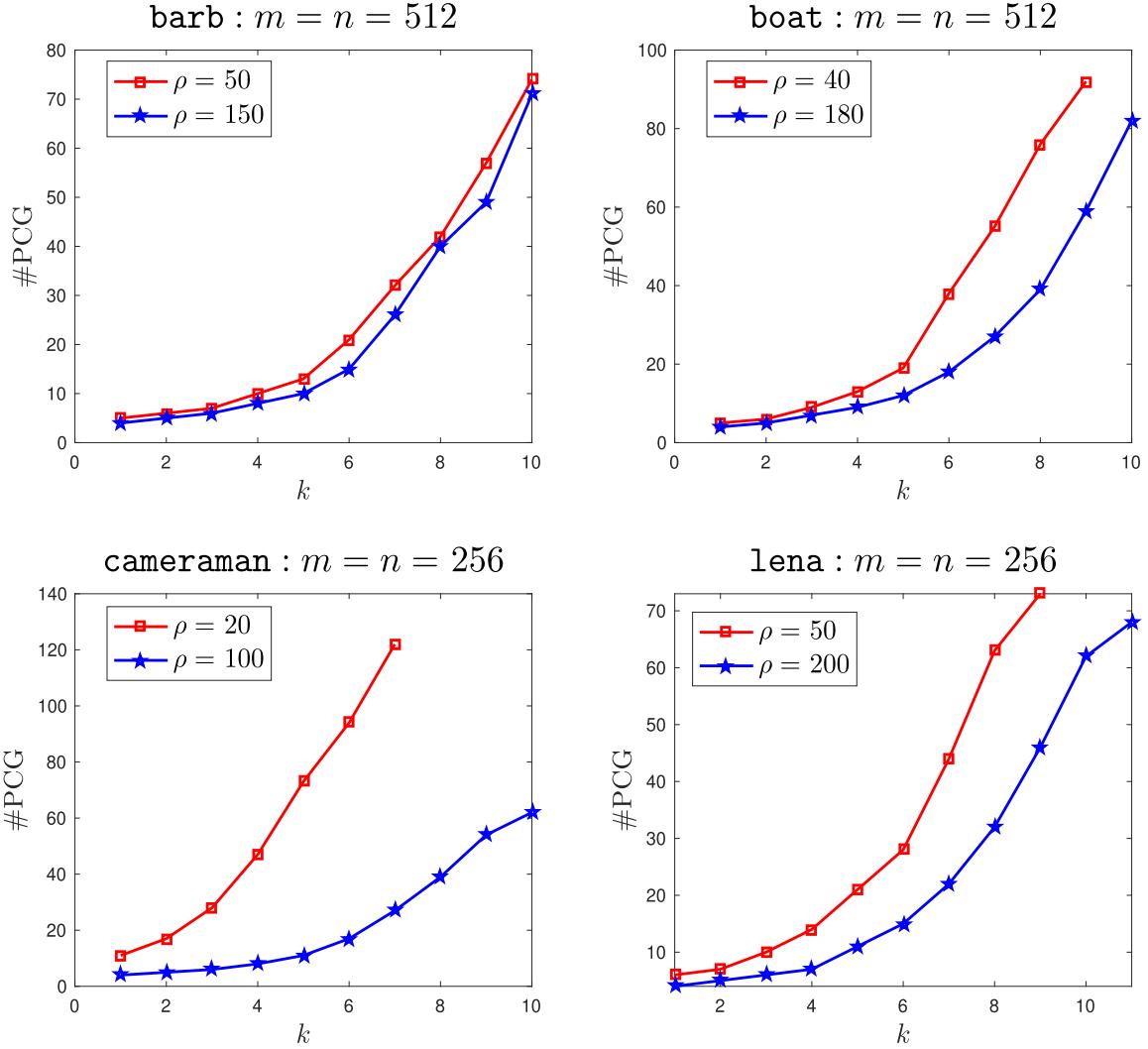


FIGURE 5. Averaged PCG iterations of Algorithm 4 for solving (5.8) with different noised input images and regularization parameters.

6. CONCLUDING REMARKS

In this work, we introduce a novel dynamical system, called primal-dual flow, for solving affine constrained convex optimization. The current model is a modification of the standard saddle-point dynamics. In continuous level, exponential decay of a tailored Lyapunov function is established. Then, in discrete level, primal-dual type algorithms are obtained from proper time discretizations of the presented primal-dual flow and nonergodic convergence rates are established *via* a unified discrete Lyapunov function.

The proposed methods adopt dynamically changing parameters and the subproblem with respect to the multiplier is solved by the semi-smooth Newton iteration. This can be quite efficient provided that the problem has nice properties such as semi-smoothness and sparsity, as showed by numerical results of the l_1 - l_2 problem and the total-variation based denoising model.

To the end, we list several ongoing works. First, well-posedness (existence and uniqueness) of the primal-dual flow system (3.1) is an interesting topic. Also, the exponential decay property (2.9) and weak convergence of the

trajectory under general nonsmooth setting deserves future investigations. Besides, rigorous convergence rate analysis with inexact computation and restart technique requires further attentions.

APPENDIX A. AN OVER-RELAXATION PERSPECTIVE

In Section 2.2, we introduced our primal-dual flow by adding the extra term x' , which is motivated from the disappointing estimate in Lemma 2.1 and leads to the desired exponential decay, and later in Section 2.3, we provided an equilibrium illustration to show further the positive effects of this correction.

To better understand the modification from the saddle-point system (1.4) to our new model (2.8), in this appendix, by using the PPA-like interpretation [44], we give a discrete over-relaxation perspective, which indicates somewhat subtle connection with the hidden symmetrization from the Arrow–Hurwicz algorithm [89] to the PDHG method [16]. We hope this provides a more reasonably intrinsic explanation.

The Arrow–Hurwicz algorithm can be applied to (1.1) and reads as

$$\begin{cases} x_{k+1} = \underset{x \in \mathbb{R}^n}{\operatorname{argmin}} \left\{ \mathcal{L}(x, \lambda_k) + \frac{1}{2r} \|x - x_k\|^2 \right\}, \\ \lambda_{k+1} = \underset{\lambda \in \mathbb{R}^m}{\operatorname{argmax}} \left\{ \mathcal{L}(x_{k+1}, \lambda) - \frac{1}{2\tau} \|\lambda - \lambda_k\|^2 \right\}, \end{cases} \quad (\text{A.1})$$

with step sizes $r, \tau > 0$. It also corresponds to a semi-implicit discretization for (1.4):

$$\begin{cases} \frac{x_{k+1} - x_k}{r} \in -\partial_x \mathcal{L}(x_{k+1}, \lambda_k), \\ \frac{\lambda_{k+1} - \lambda_k}{\tau} = \nabla_\lambda \mathcal{L}(x_{k+1}, \lambda_{k+1}). \end{cases} \quad (\text{A.2})$$

Following [44] and ([13], Chapt. 8), we use the PPA-like interpretation to demonstrate the lack of symmetry of the Arrow–Hurwicz algorithm. Introduce

$$Z = \begin{pmatrix} x \\ \lambda \end{pmatrix}, \quad M(Z) = \begin{pmatrix} \partial f(x) + A^\top \lambda \\ b - Ax \end{pmatrix} \quad \text{and} \quad Q = \begin{pmatrix} I/r & -A^\top \\ O & I/\tau \end{pmatrix},$$

where the maximally monotone operator M has been defined in (2.15). We then have the variational inequality characterization for (A.1) (or (A.2)):

$$\langle Q(Z_{k+1} - Z_k) + M(Z_{k+1}), Z - Z_{k+1} \rangle \geq 0 \quad \forall Z \in \mathbb{R}^{n+m}.$$

Taking $Z = Z^* \in \Omega^*$ and utilizing the fact: $0 \in M(Z^*)$, we find that

$$\begin{aligned} & \frac{1}{2} \|Z_{k+1} - Z^*\|_Q^2 - \frac{1}{2} \|Z_k - Z^*\|_Q^2 \\ &= \langle Q(Z_{k+1} - Z_k), Z_{k+1} - Z^* \rangle - \frac{1}{2} \|Z_{k+1} - Z_k\|_Q^2 + \frac{1}{2} \langle (Q^\top - Q)(Z_{k+1} - Z^*), Z_k - Z^* \rangle \\ &\leq \underbrace{-\langle M(Z_{k+1}), Z_{k+1} - Z^* \rangle}_{\leq 0} - \frac{1}{2} \|Z_{k+1} - Z_k\|_Q^2 + \frac{1}{2} \langle (Q^\top - Q)(Z_{k+1} - Z^*), Z_k - Z^* \rangle \\ &\leq -\frac{1}{2} \|Z_{k+1} - Z_k\|_Q^2 + \underbrace{\frac{1}{2} \langle (Q^\top - Q)(Z_{k+1} - Z^*), Z_k - Z^* \rangle}_{\neq 0}. \end{aligned} \quad (\text{A.3})$$

As Q is not symmetric, the last term makes it hard to obtain the descent estimate, and what's even worse, the scheme (A.1) is not necessarily convergent [42].

The PDHG method of Chambolle and Pock introduces a parameter $\theta \in [0, 1]$ and becomes

$$\begin{cases} x_{k+1} = \operatorname{argmin}_{x \in \mathbb{R}^n} \left\{ \mathcal{L}(x, \lambda_k) + \frac{1}{2r} \|x - x_k\|^2 \right\}, \\ \lambda_{k+1} = \operatorname{argmax}_{\lambda \in \mathbb{R}^m} \left\{ \mathcal{L}(x_{k+1} + \theta(x_{k+1} - x_k), \lambda) - \frac{1}{2\tau} \|\lambda - \lambda_k\|^2 \right\}, \end{cases} \quad (\text{A.4})$$

which is also equivalent to

$$\begin{cases} \frac{x_{k+1} - x_k}{r} \in -\partial_x \mathcal{L}(x_{k+1}, \lambda_k), \\ \frac{\lambda_{k+1} - \lambda_k}{\tau} = \nabla_\lambda \mathcal{L}(x_{k+1} + \theta(x_{k+1} - x_k), \lambda_{k+1}). \end{cases} \quad (\text{A.5})$$

Comparing this with the previous discretization (A.2), we observe the additional extrapolation term $x_{k+1} - x_k$. For the case $\theta = 1$, we have ergodic convergence rate $\mathcal{O}(1/k)$ under the condition $r\tau \|A\|^2 < 1$. Moreover, applying the above PPA-like framework to the PDHG method (with $\theta = 1$), one observes that the estimate (A.3) is now improved to

$$\frac{1}{2} \|Z_{k+1} - Z^*\|_{\widehat{Q}}^2 - \frac{1}{2} \|Z_k - Z^*\|_{\widehat{Q}}^2 \leq -\frac{1}{2} \|Z_{k+1} - Z_k\|_{\widehat{Q}}^2 \quad \text{with} \quad \widehat{Q} = \begin{pmatrix} I/r & -A^\top \\ -A & I/\tau \end{pmatrix},$$

where \widehat{Q} is a symmetrization of Q , due to the over-relaxation $x_{k+1} - x_k$.

Surprisingly, instead of the original saddle-point system (1.4), the PDHG method (A.4) is more likely a time discretization (cf. (A.5)) for the modified model

$$\begin{cases} x' = -\nabla_x \mathcal{L}(x, \lambda), \\ \lambda' = \nabla_\lambda \mathcal{L}(x + x', \lambda), \end{cases}$$

which differs from our primal-dual flow (2.8) only in the time scaling parameters. In conclusion, the extra derivative x' in $\nabla_\lambda \mathcal{L}(x + x', \lambda)$ corresponds to discrete over-relaxation $x_{k+1} - x_k$ in PDHG, which possibly brings hidden symmetrization.

Acknowledgements. This work was supported by the NSFC project 11625101. The author would like to thank Professor Jun Hu for useful comments and advices. Besides, the author want to thank the two anonymous reviewers, as the manuscript was greatly benefit from their invaluable suggestions.

REFERENCES

- [1] T. Aspelmeier, C. Charitha and D.R. Luke, Local linear convergence of the ADMM/Douglas–Rachford algorithms without strong convexity and application to statistical imaging. *SIAM J. Imaging Sci.* **9** (2016) 842–868.
- [2] H. Attouch, Z. Chbani, J. Peypouquet and P. Redont, Fast convergence of inertial dynamics and algorithms with asymptotic vanishing viscosity. *Math. Program. Series B* **168** (2018) 123–175.
- [3] H. Attouch, X. Goudou and P. Redont, The heavy ball with friction method, I. The continuous dynamical system: Global exploration of the local minima of a real-valued function by asymptotic analysis of a dissipative dynamical system. *Commun. Contemp. Math.* **2** (2000) 1–34.
- [4] H. Attouch, J. Peypouquet and P. Redont, Fast convex optimization via inertial dynamics with Hessian driven damping. *J. Differ. Equ.* **261** (2016).
- [5] A. Beck, *First-Order Methods in Optimization*, volume 1 of *MOS–SIAM Series on Optimization*. Society for Industrial and Applied Mathematics and the Mathematical Optimization Society (2017).

- [6] A. Beck and M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imag. Sci.* **2** (2009) 183–202.
- [7] A. Beck and M. Teboulle, Gradient-based algorithms with applications to signal-recovery problems, in D. Palomar and Y. Eldar, editors, *Convex Optimization in Signal Processing and Communications*. Cambridge University Press, Cambridge (2009) 42–88.
- [8] D. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*. Academic Press, New York (2014).
- [9] S. Bonettini and V. Ruggiero, On the convergence of primal–dual hybrid gradient algorithms for total variation image restoration. *J. Math. Imaging Vis.* **44** (2012) 236–253.
- [10] S. Boyd, N. Parikh, E. Chu and J. Peleato, B. and Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.* **3** (2010) 1–122.
- [11] H. Brézis, *Operateurs Maximaux Monotones: Et Semi-Groupes De Contractions Dans Les Espaces De Hilbert*, North-Holland Publishing Co., North-Holland Mathematics Studies, No. 5. Notas de Matemática (50) (1973).
- [12] J.-F. Cai, S. Osher and Z. Shen, Linearized Bregman iterations for compressed sensing. *Math. Comput.* **78** (2009) 1515–1536.
- [13] C. Clason and T. Valkonen, Nonsmooth Analysis and Optimization. Preprint <https://arxiv.org/abs/2001.00216> (2020).
- [14] E. Candès and M. Wakin, An introduction to compressive sampling. *IEEE Signal Process. Mag.* (2008) 21–30.
- [15] A. Chambolle, An algorithm for total variation minimization and applications. *J. Math. Imag. Vision* **20** (2004) 89–97.
- [16] A. Chambolle and T. Pock, A first-order primal-dual algorithm for convex problems with applications to imaging. *J. Math. Imag. Vis.* **40** (2011) 120–145.
- [17] A. Chambolle and T. Pock, An introduction to continuous optimization for imaging. *Acta Numer.* **25** (2016) 161–319.
- [18] A. Chambolle and T. Pock, On the ergodic convergence rates of a first-order primal–dual algorithm. *Math. Program.* **159** (2016) 253–287.
- [19] L. Chen and H. Luo, A unified convergence analysis of first order convex optimization methods via strong Lyapunov functions. Preprint arXiv:2108.00132 (2021).
- [20] L. Chen and H. Luo, First order optimization methods based on Hessian-driven Nesterov accelerated gradient flow. Preprint arXiv:1912.09276 (2019).
- [21] S. Chen, D. Donoho and M. Saunders, Atomic decomposition by basis pursuit. *SIAM J. Sci. Comput.* **20** (1999) 33–61.
- [22] A. Cherukuri, B. Gharesifard and J. Cortés, Saddle-point dynamics: conditions for asymptotic stability of saddle points. *SIAM J. Control Optim.* **55** (2017) 486–511.
- [23] A. Cherukuri, E. Mallada and J. Cortés, Asymptotic convergence of constrained primal-dual dynamics. *Syst. Control Lett.* **87** (2016) 10–15.
- [24] F. Clarke, *Optimization and Nonsmooth Analysis*. Number 5 in Classics in Applied Mathematics. Society for Industrial and Applied Mathematics (1987).
- [25] D. Davis and W. Yin, Faster convergence rates of relaxed Peaceman-Rachford and ADMM under regularity assumptions. Preprint arXiv:1407.5210 (2015).
- [26] W. Deng and W. Yin, On the global and linear convergence of the generalized alternating direction method of multipliers. *J. Sci. Comput.* **66** (2016) 889–916.
- [27] J. Dennis and R. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Number 16 in Classics in applied mathematics. Society for Industrial and Applied Mathematics, Philadelphia (1996).
- [28] B. Djafari-Rouhani and H. Khatibzadeh, *Nonlinear Evolution and Difference Equations of Monotone Type in Hilbert Spaces*. CRC Press, Boca Raton (2019), 1st edition.
- [29] J. Douglas and H.H. Rachford, On the numerical solution of heat conduction problems in two and three space variables. *Trans. Am. Math. Soc.* **82** (1956) 421–439.
- [30] J. Eckstein, Augmented Lagrangian and alternating direction methods for convex optimization: a tutorial and some illustrative computational results, Technical report, Rutgers University (2012).
- [31] 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 J. Imag. Sci.* **3** (2010) 1015–1046.
- [32] F. Facchinei and J. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problems*, vol. 2. Springer, New York (2006).
- [33] D. Feijer and F. Paganini, Stability of primal-dual gradient dynamics and applications to network optimization. *Automatica* **46** (2010) 1974–1981.
- [34] G. Franca, D. Robinson and R. Vidal, ADMM and accelerated ADMM as continuous dynamical systems. *35th Int. Conf. Mach. Learn. ICML 2018* **4** (2018) 2528–2536.
- [35] M. Fortin and R. Glowinski, On decomposition-coordination methods using an augmented Lagrangian, In *Studies in Mathematics and Its Applications*, volume 15 of *Augmented Lagrangian Methods: Applications to the Numerical Solution of Boundary-Value Problems*. North-Holland Publishing, Amsterdam (1983).
- [36] D. Gabay and B. Mercier, A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Comput. Math. Appl.* **2** (1976) 17–40.
- [37] P. Giselsson and S. Boyd, Linear convergence and metric selection for Douglas–Rachford splitting and ADMM. *IEEE Trans. Automat. Contr.* **62** (2017) 532–544.
- [38] J. Han and D. Sun, Newton and quasi-Newton methods for normal maps with polyhedral sets. *J. Optim. Theory Appl.* **94** (1997) 659–676.

- [39] D. Han, D. Sun and L. Zhang, Linear rate convergence of the alternating direction method of multipliers for convex composite quadratic and semi-definite programming. Preprint arXiv:1508.02134 (2015).
- [40] A. Haraux, vol. 17 of Systèmes dynamiques dissipatifs et applications, Recherches en Mathématiques Appliquées [Research in Applied Mathematics]. Masson, Paris (1991).
- [41] X. He, R. Hu and Y. Fang, Convergence rates of inertial primal-dual dynamical methods for separable convex optimization problems. Preprint arXiv:2007.12428 (2020).
- [42] B. He, F. Ma and X. Yuan, An algorithmic framework of generalized primal-dual hybrid gradient methods for saddle point problems. *J. Math. Imag. Vis.* **58** (2017) 279–293.
- [43] B. He, Y. You and X. Yuan, On the convergence of primal-dual hybrid gradient algorithm. *SIAM J. Imaging Sci.* **7** (2014) 2526–2537.
- [44] B. He and X. Yuan, Convergence analysis of primal-dual algorithms for a saddle-point problem: from contraction perspective. *SIAM J. Imag. Sci.* **5** (2012) 119–149.
- [45] B. Huang, S. Ma and D. Goldfarb, Accelerated linearized Bregman method. *J. Sci. Comput.* **54** (2013) 428–453.
- [46] F. Jiang, X. Cai, Z. Wu and D. Han, Approximate first-order primal-dual algorithms for saddle point problems. *Math. Comput.* **90** (2021) 1227–1262.
- [47] M. Kang, M. Kang and M. Jung, Inexact accelerated augmented Lagrangian methods. *Comput. Optim. Appl.* **62** (2015) 373–404.
- [48] M. Kang, S. Yun, H. Woo and M. Kang, Accelerated Bregman method for linearly constrained ℓ_1 - ℓ_2 minimization. *J. Sci. Comput.* **56** (2013) 515–534.
- [49] G. Lan and R. Monteiro, Iteration-complexity of first-order penalty methods for convex programming. *Math. Program.* **138** (2013) 115–139.
- [50] Y.-J. Lee, J. Wu, J. Xu and L. Zikatanov, Robust subspace correction methods for nearly singular systems. *Math. Models Methods Appl. Sci.* **17** (2007) 1937–1963.
- [51] D. Li, X. Sun and K. Toh, On the efficient computation of a generalized Jacobian of the projector over the Birkhoff polytope. *Math. Program.* **179** (2020) 419–446.
- [52] H. Li, C. Fang and Z. Lin, Convergence rates analysis of the quadratic penalty method and its applications to decentralized distributed optimization. Preprint arXiv:1711.10802 (2017).
- [53] X. Li, D. Sun and K. Toh, An asymptotically superlinearly convergent semismooth Newton augmented Lagrangian method for linear programming. *SIAM J. Optim.* **30** (2020) 2410–2440.
- [54] T. Lin and M.I. Jordan, A control-theoretic perspective on optimal high-order optimization. Preprint arXiv:1912.07168 (2019).
- [55] Y. Liu, X. Yuan, S. Zeng and J. Zhang, Partial error bound conditions and the linear convergence rate of the alternating direction method of multipliers. *SIAM J. Numer. Anal.* **56** (2018) 2095–2123.
- [56] H. Lu, An $O(s^r)$ -resolution ODE framework for discrete-time optimization algorithms and applications to convex-concave saddle-point problems. arXiv:2001.08826 (2020).
- [57] H. Luo, Accelerated differential inclusion for convex optimization. *Optimization* (2021) <https://doi.org/10.1080/02331934.2021.2002327>.
- [58] H. Luo and L. Chen, From differential equation solvers to accelerated first-order methods for convex optimization. *Math. Program.* (2021) <https://doi.org/10.1007/s10107-021-01713-3>.
- [59] Y. Nesterov, Gradient methods for minimizing composite functions. *Math. Program. Ser. B* **140** (2013) 125–161.
- [60] D. Niu, C. Wang, P. Tang, Q. Wang and E. Song, A sparse semismooth Newton based augmented Lagrangian method for large-scale support vector machines. Preprint arXiv:1910.01312 (2019).
- [61] D. O'Connor and L. Vandenberghe, On the equivalence of the primal-dual hybrid gradient method and Douglas–Rachford splitting. *Math. Program.* (2020). <https://doi.org/10.1007/s10107-018-1321-1>
- [62] S. Osher, M. Burger, D. Goldfarb, J. Xu and W. Yin, An iterative regularization method for total variation-based image restoration. *SIAM J. Multiscale Model. Simul.* **4** (2019) 460–489.
- [63] N. Parikh and S. Boyd, Proximal algorithms. *Found. Trends Optim.* **1** (2014).
- [64] T. Pock, D. Cremers, H. Bischof and A. Chambolle, An algorithm for minimizing the Mumford-Shah functional. In *2009 IEEE 12th International Conference on Computer Vision*. IEEE, Kyoto (2009) 1133–1140.
- [65] D.W. Peaceman and H.H. Rachford, The numerical solution of parabolic and elliptic differential equations. *J. Soc. Ind. Appl. Math.* **3** (1955) 28–41.
- [66] L. Qi, Convergence analysis of some algorithms for solving nonsmooth equations. *Math. Oper. Res.* **18** (1993) 227–244.
- [67] L. Qi and J. Sun, A nonsmooth version of Newton's method. *Math. Program.* **58** (1993) 353–367.
- [68] L. Rudin, S. Osher and E. Fatemi, Nonlinear total variation based noise removal algorithms. *Phys. D* **60** (1992) 259–268.
- [69] Y. Saad, Iterative Methods for Sparse Linear Systems, 2nd. Society for Industrial and Applied Mathematics, USA (2003).
- [70] S. Sabach and M. Teboulle, Faster Lagrangian-based methods in convex optimization. arXiv:2010.14314 (2020).
- [71] W. Su, S. Boyd and E. Candès, A differential equation for modeling Nesterov's accelerated gradient method: theory and insights. *J. Mach. Learn. Res.* **17** (2016) 1–43.
- [72] M. Tao and X. Yuan, Accelerated Uzawa methods for convex optimization. *Math. Comput.* **86** (2016) 1821–1845.
- [73] Q. Tran-Dinh, A unified convergence rate analysis of the accelerated smoothed gap reduction algorithm. *Optim. Lett.* (2021) <https://doi.org/10.1007/s11590-021-01775-4>.
- [74] Q. Tran-Dinh, Proximal alternating penalty algorithms for nonsmooth constrained convex optimization. *Comput. Optim. Appl.* **72** (2019) 1–43.

- [75] Q. Tran-Dinh and V. Cevher, Constrained convex minimization via model-based excessive gap. In *In Proc. the Neural Information Processing Systems (NIPS)*, Vol. 27. Montreal, Canada (2014) 721–729.
- [76] Q. Tran-Dinh, O. Fercoq and V. Cevher, A smooth primal-dual optimization framework for nonsmooth composite convex minimization. *SIAM J. Optim.* **28** (2018) 96–134.
- [77] Q. Tran-Dinh and Y. Zhu, Augmented Lagrangian-based decomposition methods with non-ergodic optimal rates. arXiv:[1806.05280](https://arxiv.org/abs/1806.05280) (2018).
- [78] Q. Tran-Dinh and Y. Zhu, Non-stationary first-order primal-dual algorithms with faster convergence rates. *SIAM J. Optim.* **30** (2020) 2866–2896.
- [79] T. Valkonen, Inertial, corrected, primal-dual proximal splitting. *SIAM J. Optim.* **30** (2020) 1391–1420.
- [80] A. Wibisono, A. Wilson and M. Jordan, A variational perspective on accelerated methods in optimization. *Proc. Natl. Acad. Sci.* **113** (2016) E7351–E7358.
- [81] A.C. Wilson, B. Recht and M.I. Jordan, A Lyapunov analysis of accelerated methods in optimization. *J. Mach. Learn. Res.* **22** (2021) 1–34.
- [82] Y. Xu, Accelerated first-order primal-dual proximal methods for linearly constrained composite convex programming. *SIAM J. Optim.* **27** (2017) 1459–1484.
- [83] J. Xu and L. Zikatanov, Algebraic multigrid methods. *Acta Numer.* **26** (2017) 591–721.
- [84] W.H. Yang and D. Han, Linear convergence of the alternating direction method of multipliers for a class of convex optimization problems. *SIAM J. Numer. Anal.* **54** (2016) 625–640.
- [85] M. Yan and W. Yin, Self equivalence of the alternating direction method of multipliers. arXiv:[1407.7400](https://arxiv.org/abs/1407.7400) (2015).
- [86] W. Yin, Analysis and generalizations of the linearized Bregman method. *SIAM J. Imag. Sci.* **3** (2010) 856–877.
- [87] X. Yuan, S. Zeng and J. Zhang, Discerning the linear convergence of ADMM for structured convex optimization through the lens of variational analysis. *J. Mach. Learn. Res.* **21** (2020) 1–74.
- [88] X. Zeng, J. Lei and J. Chen, Dynamical primal-dual accelerated method with applications to network optimization. Print arXiv:[1912.03690](https://arxiv.org/abs/1912.03690) (2019), pp. 1–22.
- [89] M. Zhu and T. Chan, An efficient primal-dual hybrid gradient algorithm for total variation image restoration. Technical report CAM Report 08-34, UCLA, Los Angeles, CA, USA (2008).

Subscribe to Open (S2O)

A fair and sustainable open access model



This journal is currently published in open access under a Subscribe-to-Open model (S2O). S2O is a transformative model that aims to move subscription journals to open access. Open access is the free, immediate, online availability of research articles combined with the rights to use these articles fully in the digital environment. We are thankful to our subscribers and sponsors for making it possible to publish this journal in open access, free of charge for authors.

Please help to maintain this journal in open access!

Check that your library subscribes to the journal, or make a personal donation to the S2O programme, by contacting subscribers@edpsciences.org

More information, including a list of sponsors and a financial transparency report, available at: <https://www.edpsciences.org/en/math-s2o-programme>