

An Algorithmic Framework of Generalized Primal–Dual Hybrid Gradient Methods for Saddle Point Problems

Bingsheng He^{1,2} · Feng Ma³ · Xiaoming Yuan⁴

Received: 2 August 2016 / Accepted: 3 February 2017 / Published online: 21 February 2017
© Springer Science+Business Media New York 2017

Abstract The primal–dual hybrid gradient method (PDHG) originates from the Arrow–Hurwicz method, and it has been widely used to solve saddle point problems, particularly in image processing areas. With the introduction of a combination parameter, Chambolle and Pock proposed a generalized PDHG scheme with both theoretical and numerical advantages. It has been analyzed that except for the special case where the combination parameter is 1, the PDHG cannot be casted to the proximal point algorithm framework due to the lack of symmetry in the matrix associated with the proximal regularization terms. The PDHG scheme is non-symmetric also in the sense that one variable is updated twice while the other is only updated once at each iteration. These nonsymmetry features also explain why more theoretical issues remain challenging for generalized PDHG schemes; for example, the worst-case convergence rate of PDHG measured by the iteration complexity in a noner-

godic sense is still missing. In this paper, we further consider how to generalize the PDHG and propose an algorithmic framework of generalized PDHG schemes for saddle point problems. This algorithmic framework allows the output of the PDHG subroutine to be further updated by correction steps with constant step sizes. We investigate the restriction onto these step sizes and conduct the convergence analysis for the algorithmic framework. The algorithmic framework turns out to include some existing PDHG schemes as special cases, and it immediately yields a class of new generalized PDHG schemes by choosing different step sizes for the correction steps. In particular, a completely symmetric PDHG scheme with the golden-ratio step sizes is included. Theoretically, an advantage of the algorithmic framework is that the worst-case convergence rate measured by the iteration complexity in both the ergodic and nonergodic senses can be established.

Bingsheng He was supported by the NSFC Grants 11471156 and 91530115, Feng Ma was supported by the NSFC Grant 71401176 and Xiaoming Yuan was supported by the General Research Fund from Hong Kong Research Grants Council: HKBU12300515.

✉ Xiaoming Yuan
xmyuan@hkbu.edu.hk
Bingsheng He
hebma@nju.edu.cn
Feng Ma
mafengnju@gmail.com

- ¹ Department of Mathematics, Nanjing University, Nanjing, China
- ² Department of Mathematics, South University of Science and Technology of China, Shenzhen, China
- ³ High-Tech Institute of Xi'an, Xi'an 710025, Shaanxi, China
- ⁴ Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong

Keywords Convex programming · Saddle point problem · Image restoration · Variational models · Primal–dual hybrid gradient method · Convergence rate · Variational inequalities

1 Introduction

We consider the saddle point problem

$$\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} \Phi(x, y) := \theta_1(x) - y^T A x - \theta_2(y), \quad (1.1)$$

where $A \in \mathfrak{R}^{m \times n}$, $\mathcal{X} \subseteq \mathfrak{R}^n$ and $\mathcal{Y} \subseteq \mathfrak{R}^m$ are closed convex sets, $\theta_1 : \mathfrak{R}^n \rightarrow \mathfrak{R}$ and $\theta_2 : \mathfrak{R}^m \rightarrow \mathfrak{R}$ are convex but not necessarily smooth functions. The solution set of (1.1) is assumed to be nonempty throughout our discussion. The model (1.1) captures a variety of applications in different

areas. For examples, finding a saddle point of the Lagrangian function of the canonical convex minimization model with linear equality or inequality constraints is a special case of (1.1). Moreover, a number of variational image restoration problems with the total variation (TV) regularization (see [25]) can be reformulated as special cases of (1.1), see details in, e.g., [5, 27–29].

Since the work [29], the primal–dual hybrid gradient (PDHG) method has attracted much attention and it is still being widely used for many applications, especially in image processing areas. We start the discussion of PDHG with the scheme proposed in [5]:

$$\begin{cases} x^{k+1} = \arg \min \{ \Phi(x, y^k) + \frac{r}{2} \|x - x^k\|^2 \mid x \in \mathcal{X} \}, \\ \tilde{x}^k = x^{k+1} + \tau(x^{k+1} - x^k), \\ y^{k+1} = \arg \max \{ \Phi(\tilde{x}^k, y) - \frac{s}{2} \|y - y^k\|^2 \mid y \in \mathcal{Y} \}, \end{cases} \tag{1.2}$$

where $\tau \in [0, 1]$ is a combination parameter, and $r > 0$ and $s > 0$ are proximal parameters of the regularization terms. The scheme (1.2) splits the coupled term $y^T Ax$ in (1.1) and treats the functions θ_1 and θ_2 individually; the resulting subproblems are usually easier than the original problem (1.1). In [5], it was shown that the PDHG scheme (1.2) is closely related to the extrapolational gradient methods in [20, 24], the Douglas-Rachford splitting method in [9, 21] and the alternating direction method of multipliers (ADMM) in [12]. In particular, we refer to [10, 26] for the equivalence between a special case of (1.2) and a linearized version of the ADMM. As analyzed in [5], the convergence of (1.2) can be guaranteed under the condition

$$rs > \|A^T A\|. \tag{1.3}$$

parameters, see, e.g., [3, 10, 13]. Here, to expose our main idea more clearly and to avoid heavy notation, we only focus on the case where both r and s are constant and they satisfy the condition in (1.3) throughout our discussion.

In [16], we showed that the special case of (1.2) with $\tau = 1$ is an application of the proximal point algorithm (PPA) in [22], and thus the scheme in [14] can be immediately combined with the PDHG scheme (see Algorithm 4 in [16]). Its numerical efficiency has also been verified therein. This PPA revisit has been further studied in [23], in which a preconditioning version of the PDHG scheme (1.2) was proposed. When $\tau \neq 1$,¹ it is shown in [16] [see also (2.7b)] that the matrix associated with the proximal regularization terms in (1.2) is not symmetric and thus the scheme (1.2) cannot be casted to an application of the PPA. The techniques in [14] are thus not applicable to (1.2) with $\tau \neq 1$. But the convergence can be guaranteed if the output of the PDHG subroutine (1.2) is further corrected by some correction steps (see Algorithms 1 and 2 in [16]). The step sizes of these correction steps, however, must be calculated iteratively and they usually require expensive computation (e.g., multiplications of matrices or matrices and vectors) if high dimensional variables are considered. It is thus natural to ask whether or not we can further correct the output of (1.2) when $\tau \neq 1$ while the step sizes of the correction steps are iteration-independent (more precisely, constants)? Moreover, it is easy to notice that the PDHG scheme (1.2) is not symmetric also in the sense that the variable x is updated twice while y is updated only once at each iteration. So one more question is whether or not we can update y also right after the PDHG subroutine and modify (1.2) as a completely symmetric version? To answer these questions, we propose the following unified algorithmic framework that allows the variables to be updated both, either, or neither, after the PDHG subroutine (1.2):

$$\text{(Algorithmic Framework)} \quad \begin{cases} \begin{cases} \tilde{x}^k = \arg \min \{ \Phi(x, y^k) + \frac{r}{2} \|x - x^k\|^2 \mid x \in \mathcal{X} \}, \\ \tilde{x}^k = \tilde{x}^k + \tau(\tilde{x}^k - x^k), \\ \tilde{y}^k = \arg \max \{ \Phi(\tilde{x}^k, y) - \frac{s}{2} \|y - y^k\|^2 \mid y \in \mathcal{Y} \}, \end{cases} & \tag{1.4a} \\ \begin{cases} x^{k+1} = x^k - \alpha(x^k - \tilde{x}^k), \\ y^{k+1} = y^k - \beta(y^k - \tilde{y}^k). \end{cases} & \tag{1.4b} \end{cases}$$

When $\tau = 0$, the scheme (1.2) reduces to the PDHG scheme in [29] which is indeed the Arrow–Hurwicz method in [1]. In addition to the numerical advantages shown in, e.g., [5, 16], the theoretical significance of extending $\tau = 0$ to $\tau \in [0, 1]$ was demonstrated in [15]. That is, if $\tau = 0$, then the scheme (1.2) could be divergent even if r and s are fixed at very large values; thus, the condition (1.3) is not sufficient to ensure the convergence of (1.2). Note that it could be numerically beneficial to tune the parameters r and s as shown in, e.g., [16, 28], and it is still possible to investigate the convergence of the PDHG scheme (1.2) with adaptively adjusted proximal

In (1.4), the output of the PDHG subroutine (1.2) is denoted by $(\tilde{x}^k, \tilde{y}^k)$, $\alpha > 0$ and $\beta > 0$ are step sizes of the correction steps to further update the variables. This new algorithmic framework is thus a combination of the PDHG subroutine (1.2) with a correction step. We also call $(\tilde{x}^k, \tilde{y}^k)$ generated by (1.4a) a predictor and the new iterate (x^{k+1}, y^{k+1}) updated by (1.4b) a corrector. Note that compared with the PDHG (1.2), two more constants α and β are involved in

¹ We analyzed the case $\tau \in [-1, 1]$ in [16]; but for simplicity we only focus on $\tau \in [0, 1]$ in this paper.

the algorithmic framework (1.4). But the framework (1.4) with different choices of α and β corresponds to various specific PDHG-like algorithms, and as we shall show, it is very easy to theoretically determine these constants and thus a series of PDHG-like algorithms can be specified by the framework (1.4). Therefore, the additional constants in (1.4) do not increase the difficulty of implementing PDHG-like algorithms based on (1.4).

To see the generality of (1.4), when $\alpha = \beta = 1$, the algorithmic framework (1.4) reduces to the generalized PDHG (1.2) in [5]; when $\tau = 1$ and $\alpha = \beta \in (0, 2)$, the improved version of PDHG from the PPA revisit (Algorithm 4 in [16]) is recovered. Moreover, if $\alpha = 1$, we obtain a symmetric generalized PDHG [see (4.9)] that updates both the variables twice at each iteration, and in particular, a completely symmetric version with the golden-ratio step sizes [see (4.12)]. Therefore, based on this algorithmic framework, some existing PDHG schemes can be recovered and a class of new generalized PDHG schemes can be easily proposed by choosing different values of α , β and τ . We shall focus on the case where $\tau \in [0, 1]$ and investigate the restriction onto these constants so that the convergence of the algorithmic framework can be ensured. The analysis is conducted in a unified manner. In addition, the mentioned nonsymmetry features of the PDHG scheme (1.2) may also explain the difficulty of proving its worst-case convergence rate measured by the iteration complexity in a nonergodic sense. For the generalized PDHG algorithmic framework (1.4), we shall show that the worst-case convergence rate in both the ergodic and nonergodic senses can be derived when the parameters are appropriately restricted. This is a theoretical advantage of the algorithmic framework (1.4).

Finally, we would like to mention that there is a rich set of literature discussing how to extend the PDHG with more sophisticated analysis to more complicated saddle point models or to more abstract spaces, e.g., [6–8]. But in this paper, we concentrate on the canonical saddle point model (1.1) in a finitely dimensional space and the representative PDHG scheme (1.2) to present our idea of algorithmic design clearly.

The rest of this paper is organized as follows. In Sect. 2, we summarize some understandings of the model (1.1) and the algorithmic framework (1.4) from the variational inequality perspective. Then, we prove the convergence of (1.4) in Sect. 3. In Sect. 4, we elaborate on the conditions that can ensure the convergence of (1.4); some specific generalized PDHG schemes are thus yielded based on the algorithmic framework (1.4). Then, we derive the worst-case convergence rate measured by the iteration complexity for (1.4) in Sect. 5. We test the efficiency of these specific generalized PDHG algorithms by an image restoration model in Sect. 6. Finally, we draw some conclusions in Sect. 7.

2 Variational Inequality Understanding

In this section, we provide the variational inequality (VI) reformulation of the model (1.1) and characterize the algorithmic framework (1.4) via a VI. These VI understandings are the basis of our analysis to be conducted.

2.1 Variational Inequality Reformulation of (1.1)

We first show that the saddle point problem (1.1) can be written as a VI problem. More specifically, if $(x^*, y^*) \in \mathcal{X} \times \mathcal{Y}$ is a solution point of the saddle point problem (1.1), then we have

$$\forall y \in \mathcal{Y}, \forall x \in \mathcal{X}, \Phi(x^*, y) \leq \Phi(x^*, y^*) \leq \Phi(x, y^*). \tag{2.1}$$

Obviously, the second inequality in (2.1) is equivalent to

$$\begin{aligned} x^* \in \mathcal{X}, \quad \theta_1(x) - \theta_1(x^*) + (x - x^*)^T (-A^T y^*) \geq 0, \\ \forall x \in \mathcal{X}; \end{aligned}$$

and the first one in (2.1) can be written as

$$y^* \in \mathcal{Y}, \quad \theta_2(y) - \theta_2(y^*) + (y - y^*)^T (Ax^*) \geq 0, \quad \forall y \in \mathcal{Y}.$$

Therefore, finding a solution point (x^*, y^*) of (1.1) is equivalent to solving the VI problem: Find $u^* = (x^*, y^*)$ such that

$$\begin{aligned} \text{VI}(\Omega, F, \theta) \quad u^* \in \Omega, \quad \theta(u) - \theta(u^*) \\ + (u - u^*)^T F(u^*) \geq 0, \quad \forall u \in \Omega, \end{aligned} \tag{2.2a}$$

where

$$\begin{aligned} u = \begin{pmatrix} x \\ y \end{pmatrix}, \theta(u) = \theta_1(x) + \theta_2(y), \quad F(u) = \begin{pmatrix} -A^T y \\ Ax \end{pmatrix} \\ \text{and } \Omega = \mathcal{X} \times \mathcal{Y}. \end{aligned} \tag{2.2b}$$

We denote by Ω^* the set of all solution points of $\text{VI}(\Omega, F, \theta)$ (2.2). Notice that Ω^* is convex (see Theorem 2.3.5 in [11] or Theorem 2.1 in [17]).

Clearly, for the mapping F given in (2.2b), we have

$$(u - v)^T (F(u) - F(v)) = 0, \quad \forall u, v \in \Omega. \tag{2.3}$$

Moreover, under the condition (1.3), the matrix

$$\begin{pmatrix} rI_n & A^T \\ A & sI_m \end{pmatrix} \tag{2.4}$$

is positive definite. The positive definiteness of this matrix plays a significant role in analyzing the convergence of the PDHG scheme (1.2), see e.g. [5, 10, 16, 28].

2.2 Variational Inequality Characterization of (1.4)

Then, we rewrite the algorithmic framework (1.4) also in the VI form. Note that the PDHG subroutine (1.4a) can be written as

$$\begin{cases} \tilde{x}^k = \arg \min \{ \theta_1(x) - (y^k)^T Ax + \frac{r}{2} \|x - x^k\|^2 \mid x \in \mathcal{X} \}, \\ \bar{x}^k = \tilde{x}^k + \tau(\tilde{x}^k - x^k), \end{cases} \tag{2.5a}$$

$$\tag{2.5b}$$

$$\begin{cases} \tilde{y}^k = \arg \min \{ \theta_2(y) + y^T A\bar{x}^k + \frac{s}{2} \|y - y^k\|^2 \mid y \in \mathcal{Y} \}. \end{cases} \tag{2.5c}$$

Thus, the optimality conditions of (2.5a) and (2.5c) are

$$\begin{aligned} \tilde{x}^k \in \mathcal{X}, \quad \theta_1(x) - \theta_1(\tilde{x}^k) + (x - \tilde{x}^k)^T (-A^T y^k + r(\tilde{x}^k - x^k)) \\ \geq 0, \quad \forall x \in \mathcal{X}, \end{aligned}$$

and

$$\begin{aligned} \tilde{y}^k \in \mathcal{Y}, \quad \theta_2(y) - \theta_2(\tilde{y}^k) + (y - \tilde{y}^k)^T (A\bar{x}^k + s(\tilde{y}^k - y^k)) \\ \geq 0, \quad \forall y \in \mathcal{Y}, \end{aligned}$$

respectively. Using (2.5b), we get the following VI containing only (x^k, y^k) and $(\tilde{x}^k, \tilde{y}^k)$:

$$\begin{aligned} (\tilde{x}^k, \tilde{y}^k) \in \mathcal{X} \times \mathcal{Y}, \quad (\theta_1(x) - \theta_1(\tilde{x}^k)) + (\theta_2(y) - \theta_2(\tilde{y}^k)) \\ + \begin{pmatrix} x - \tilde{x}^k \\ y - \tilde{y}^k \end{pmatrix}^T \left\{ \begin{pmatrix} -A^T \tilde{y}^k \\ A\tilde{x}^k \end{pmatrix} + \begin{pmatrix} r(\tilde{x}^k - x^k) + A^T(\tilde{y}^k - y^k) \\ \tau A(\tilde{x}^k - x^k) + s(\tilde{y}^k - y^k) \end{pmatrix} \right\} \\ \geq 0, \quad \forall (x, y) \in \mathcal{X} \times \mathcal{Y}. \end{aligned} \tag{2.6}$$

Prediction Step (PDHG subroutine step)

$$\begin{aligned} \tilde{u}^k \in \Omega, \quad \theta(u) - \theta(\tilde{u}^k) + (u - \tilde{u}^k)^T \{ F(\tilde{u}^k) \\ + Q(\tilde{u}^k - u^k) \} \geq 0, \quad \forall u \in \Omega, \end{aligned} \tag{2.7a}$$

where

$$Q = \begin{pmatrix} rI_n & A^T \\ \tau A & sI_m \end{pmatrix} \tag{2.7b}$$

Correction Step

$$u^{k+1} = u^k - M(u^k - \tilde{u}^k), \tag{2.8a}$$

where

$$M = \begin{pmatrix} \alpha I_n & 0 \\ 0 & \beta I_m \end{pmatrix}. \tag{2.8b}$$

3 Convergence

In this section, we investigate the convergence of the algorithmic framework (1.4). Recall that (1.4) can be rewritten as the prediction-correction way (2.7)–(2.8). We prove the convergence of (1.4) under the condition

$$\text{(Convergence Condition)} \quad \begin{cases} H := QM^{-1} \succ 0 \text{ and } H = H^T, \\ G := Q^T + Q - M^T H M \succ 0, \end{cases} \tag{3.1a}$$

$$\tag{3.1b}$$

Then, because of the notation in (2.2b), the PDHG subroutine (1.4a) can be written compactly as (2.7). Accordingly, the step (1.4b) can be rewritten as (2.8) and overall the algorithmic framework (1.4) can be explained as the prediction-correction way (2.7)–(2.8).

where the matrices Q and M are given in (2.7b) and (2.8b), respectively. Thus, it suffices to check the positive definiteness of two matrices to ensure the convergence of the algorithmic framework (1.4). The condition (3.1) also

implies some specific restrictions onto the involved parameters in (1.4) to be discussed in Sect. 4.

Because of the VI reformulation (2.2) of the model (1.1), we are interested in using the VI characterization (2.7) to discern how accurate an iterate generated by the algorithmic framework (1.4) is to a solution point of VI(Ω, F, θ). This analysis is summarized in the following theorem.

Theorem 3.1 *Let $u^k = (x^k, y^k)$ be given by the algorithmic framework (1.4). Under the condition (3.1), we have*

$$\theta(u) - \theta(\tilde{u}^k) + (u - \tilde{u}^k)^T F(\tilde{u}^k) \geq \frac{1}{2}(\|u - u^{k+1}\|_H^2 - \|u - u^k\|_H^2) + \frac{1}{2}\|u^k - \tilde{u}^k\|_G^2, \quad \forall u \in \Omega. \tag{3.2}$$

Proof It follows from (3.1a) that $Q = HM$. Using (2.8a), we know that (2.7a) can be written as

$$\theta(u) - \theta(\tilde{u}^k) + (u - \tilde{u}^k)^T F(\tilde{u}^k) \geq (u - \tilde{u}^k)^T H(u^k - u^{k+1}), \quad \forall u \in \Omega. \tag{3.3}$$

For H satisfying (3.1a), we apply the identity

$$(a - b)^T H(c - d) = \frac{1}{2} \left\{ \|a - d\|_H^2 - \|a - c\|_H^2 \right\} + \frac{1}{2} \left\{ \|c - b\|_H^2 - \|d - b\|_H^2 \right\}$$

to the right-hand side of (3.3) with

$$a = u, \quad b = \tilde{u}^k, \quad c = u^k, \quad \text{and} \quad d = u^{k+1},$$

and obtain

$$(u - \tilde{u}^k)^T H(u^k - u^{k+1}) = \frac{1}{2}(\|u - u^{k+1}\|_H^2 - \|u - u^k\|_H^2) + \frac{1}{2}(\|u^k - \tilde{u}^k\|_H^2 - \|u^{k+1} - \tilde{u}^k\|_H^2). \tag{3.4}$$

For the last term of the right-hand side of (3.4), we have

$$\begin{aligned} & \|u^k - \tilde{u}^k\|_H^2 - \|u^{k+1} - \tilde{u}^k\|_H^2 \\ &= \|u^k - \tilde{u}^k\|_H^2 - \|(u^k - \tilde{u}^k) - (u^k - u^{k+1})\|_H^2 \\ \stackrel{(3.1a)}{=} & \|u^k - \tilde{u}^k\|_H^2 - \|(u^k - \tilde{u}^k) - M(u^k - \tilde{u}^k)\|_H^2 \\ &= 2(u^k - \tilde{u}^k)^T HM(u^k - \tilde{u}^k) \\ &\quad - (u^k - \tilde{u}^k)^T M^T HM(u^k - \tilde{u}^k) \\ &= (u^k - \tilde{u}^k)^T (Q^T + Q - M^T HM)(u^k - \tilde{u}^k) \\ \stackrel{(3.1b)}{=} & \|u^k - \tilde{u}^k\|_G^2. \end{aligned} \tag{3.5}$$

Substituting (3.4) and (3.5) into (3.3), we obtain the assertion (3.2). \square

With the assertion in Theorem 3.1, we can show that the sequence $\{u^k\}$ with u^k given by (1.4) is strictly contractive with respect to the solution set Ω^* . We prove this property in the following theorem.

Theorem 3.2 *Let $u^k = (x^k, y^k)$ be given by the algorithmic framework (1.4). Under the condition (3.1), we have*

$$\|u^{k+1} - u^*\|_H^2 \leq \|u^k - u^*\|_H^2 - \|u^k - \tilde{u}^k\|_G^2, \quad \forall u^* \in \Omega^*. \tag{3.6}$$

Proof Setting $u = u^*$ in (3.2), we get

$$\|u^k - u^*\|_H^2 - \|u^{k+1} - u^*\|_H^2 \geq \|u^k - \tilde{u}^k\|_G^2 + 2\{\theta(\tilde{u}^k) - \theta(u^*) + (\tilde{u}^k - u^*)^T F(\tilde{u}^k)\}. \tag{3.7}$$

Then, using (2.3) and the optimality of u^* , we have

$$\begin{aligned} & \theta(\tilde{u}^k) - \theta(u^*) + (\tilde{u}^k - u^*)^T F(\tilde{u}^k) \\ &= \theta(\tilde{u}^k) - \theta(u^*) + (\tilde{u}^k - u^*)^T F(u^*) \geq 0 \end{aligned}$$

and thus

$$\|u^k - u^*\|_H^2 - \|u^{k+1} - u^*\|_H^2 \geq \|u^k - \tilde{u}^k\|_G^2. \tag{3.8}$$

The assertion (3.6) follows directly. \square

Theorem 3.2 shows that the sequence $\{u^k\}$ with u^k given by (1.4) is Fèjer monotone and the convergence of $\{u^k\}$ to a $u^* \in \Omega^*$ in H -norm is immediately implied, see, e.g., [2].

4 How to Ensure the Condition (3.1)

In Sect. 3, the convergence of the algorithmic framework (1.4) is derived under the condition (3.1). Now, we discuss how to appropriately choose the parameters τ, α and β to ensure the condition (3.1). With different choices of these parameters in the algorithmic framework (1.4), a series of PDHG schemes are specified for the saddle point problem (1.1).

4.1 General Study

Recall the definitions of Q and M in (2.7b) and 2.8b, respectively. Let us take a closer look at the matrices H and G defined in (3.1).

First, we have

$$H = QM^{-1} = \begin{pmatrix} rI_n & A^T \\ \tau A & sI_m \end{pmatrix} \begin{pmatrix} \frac{1}{\alpha}I_n & 0 \\ 0 & \frac{1}{\beta}I_m \end{pmatrix} = \begin{pmatrix} \frac{r}{\alpha}I_n & \frac{1}{\beta}A^T \\ \frac{\tau}{\alpha}A & \frac{1}{\beta}sI_m \end{pmatrix}.$$

Thus, we set $\beta = \frac{\alpha}{\tau}$ to ensure that H is symmetric. With this restriction, we have

$$H = \begin{pmatrix} \frac{r}{\alpha} I_n & \frac{\tau}{\alpha} A^T \\ \frac{\tau}{\alpha} A & \frac{\tau}{\alpha} s I_m \end{pmatrix} = \frac{1}{\alpha} \begin{pmatrix} r I_n & \tau A^T \\ \tau A & \tau s I_m \end{pmatrix}.$$

Then, it is clear that under the condition (1.3), we can choose any $\tau \in (0, 1]$ to ensure the positive definiteness of H . Overall, to ensure the positive definiteness and symmetry of H , we pose the restriction:

$$\tau \in (0, 1], \quad \beta = \frac{\alpha}{\tau} \quad \text{and} \quad \alpha > 0. \tag{4.1}$$

Second, let us deal with the matrix G in (3.1). Since $HM = Q$, we have

$$\begin{aligned} Q^T + Q - M^T H M &= Q^T + Q - M^T Q \\ &= \begin{pmatrix} 2r I_n & (1 + \tau) A^T \\ (1 + \tau) A & 2s I_m \end{pmatrix} - \begin{pmatrix} \alpha I_n & 0 \\ 0 & \frac{\alpha}{\tau} I_m \end{pmatrix} \begin{pmatrix} r I_n & A^T \\ \tau A & s I_m \end{pmatrix} \\ &= \begin{pmatrix} 2r I_n & (1 + \tau) A^T \\ (1 + \tau) A & 2s I_m \end{pmatrix} - \begin{pmatrix} \alpha r I_n & \alpha A^T \\ \alpha A & \frac{\alpha}{\tau} s I_m \end{pmatrix} \\ &= \begin{pmatrix} (2 - \alpha) r I_n & (1 + \tau - \alpha) A^T \\ (1 + \tau - \alpha) A & (2 - \frac{\alpha}{\tau}) s I_m \end{pmatrix}. \end{aligned} \tag{4.2}$$

Therefore, we need to ensure

$$G = \begin{pmatrix} (2 - \alpha) r I_n & (1 + \tau - \alpha) A^T \\ (1 + \tau - \alpha) A & (2 - \frac{\alpha}{\tau}) s I_m \end{pmatrix} > 0. \tag{4.3}$$

We present a strategy to ensure (4.3) in the following theorem.

Theorem 4.1 For given $\tau \in (0, 1]$, under the condition (1.3), the assertion (4.3) holds when

$$\tau = 1 \quad \text{and} \quad \alpha \in (0, 2), \tag{4.4}$$

or

$$\tau \in (0, 1), \quad \text{and} \quad 0 < \alpha \leq (1 + \tau) - \sqrt{1 - \tau}. \tag{4.5}$$

Proof First, if $\tau = 1$, it follows from (4.2) that

$$G = (2 - \alpha) \begin{pmatrix} r I_n & A^T \\ A & s I_m \end{pmatrix}.$$

Under the condition (1.3), we have $G > 0$ for all $\alpha \in (0, 2)$.

Now, we consider $\tau \in (0, 1)$. According to numerical linear algebra, e.g., Theorem 7.7.7 in [19], for $G > 0$, we need to ensure

$$\begin{aligned} (2 - \alpha) > 0, \quad \left(2 - \frac{\alpha}{\tau}\right) > 0 \quad \text{and} \quad (2 - \alpha) \left(2 - \frac{\alpha}{\tau}\right) r s \\ > (1 + \tau - \alpha)^2 \|A^T A\|. \end{aligned}$$

Thus, under the condition (1.3), it is enough to guarantee

$$(2 - \alpha) > 0, \quad \left(2 - \frac{\alpha}{\tau}\right) > 0, \tag{4.6a}$$

and

$$(2 - \alpha) \left(2 - \frac{\alpha}{\tau}\right) \geq (1 + \tau - \alpha)^2. \tag{4.6b}$$

By a manipulation, the inequality (4.6b) is equivalent to

$$\left(\frac{1}{\tau} - 1\right) \alpha^2 + 2 \left((1 + \tau) - \left(1 + \frac{1}{\tau}\right) \right) \alpha + (4 - (1 + \tau)^2) \geq 0,$$

and thus

$$\left(\frac{1 - \tau}{\tau}\right) \alpha^2 + 2 \left(\frac{\tau^2 - 1}{\tau}\right) \alpha + (3 + \tau)(1 - \tau) \geq 0.$$

Multiplying by the positive factor $\frac{\tau}{1 - \tau}$, we get

$$\alpha^2 - 2(1 + \tau)\alpha + \tau(3 + \tau) \geq 0.$$

Because the smaller root of the equation

$$\alpha^2 - 2(1 + \tau)\alpha + \tau(3 + \tau) = 0$$

is $(1 + \tau) - \sqrt{1 - \tau}$, the condition (4.6b) is satisfied when (4.5) holds.

In the following, we show that (4.6a) is satisfied when (4.5) holds. It follows from (4.5) directly $\alpha < 2$ for all $\tau \in (0, 1)$. In addition, because

$$(1 - \tau) - \sqrt{1 - \tau} < 0, \quad \forall \tau \in (0, 1),$$

we have

$$(1 + \tau) - \sqrt{1 - \tau} < 2\tau, \quad \forall \tau \in (0, 1).$$

Consequently, using (4.5), we get

$$\frac{\alpha}{\tau} \leq \frac{(1 + \tau) - \sqrt{1 - \tau}}{\tau} < 2, \quad \forall \tau \in (0, 1).$$

The proof is complete. □

In the following theorem, we summarize the restriction onto the involved parameters that can ensure the convergence of the algorithmic framework (1.4).

Theorem 4.2 Under the condition (1.3), if the parameters τ, α and β in (1.4) are chosen such that either

$$\tau = 1, \quad \text{and} \quad \alpha = \beta \in (0, 2), \tag{4.7a}$$

or

$$\tau \in (0, 1), \quad 0 < \alpha \leq 1 + \tau - \sqrt{1 - \tau} \quad \text{and} \quad \beta = \frac{\alpha}{\tau}, \tag{4.7b}$$

then the matrices Q and M defined, respectively, in (2.7b) and (2.8b) satisfy the convergence condition (3.1). Thus, the sequence $\{u^k\}$ with u^k given by the algorithmic framework (1.4) converges to a solution point of $VI(\Omega, F, \theta)$.

4.2 Special Cases

In this subsection, we specify the general restriction posed in Theorem 4.2 on the parameters of the algorithmic framework (1.4) and discuss some special cases of this framework.

4.2.1 Case I: $\tau = 1$

According to (4.7a), we can take $\alpha = \beta \in (0, 2)$ and thus the algorithmic framework (1.4) can be written as

$$\begin{cases} \begin{cases} \tilde{x}^k = \arg \min \left\{ \Phi(x, y^k) + \frac{r}{2} \|x - x^k\|^2 \mid x \in \mathcal{X} \right\}, \\ \bar{x}^k = 2\tilde{x}^k - x^k, \end{cases} & (4.8a) \\ \begin{cases} \tilde{y}^k = \arg \max \left\{ \Phi(\bar{x}^k, y) - \frac{s}{2} \|y - y^k\|^2 \mid y \in \mathcal{Y} \right\}. \\ \begin{pmatrix} x^{k+1} \\ y^{k+1} \end{pmatrix} = \begin{pmatrix} x^k \\ y^k \end{pmatrix} - \alpha \begin{pmatrix} x^k - \tilde{x}^k \\ y^k - \tilde{y}^k \end{pmatrix}. \end{cases} & (4.8b) \end{cases}$$

This is exactly Algorithm 4 in [16] which can be regarded as an improved version of (1.2) using the technique in [14]. Clearly, choosing $\alpha = \beta = 1$ in (4.8) recovers the special case of (1.2) with $\tau = 1$

4.2.2 Case II: $\alpha = 1$

Now let $\alpha = 1$ be fixed. In this case, the algorithmic framework (1.4) can be written as

$$\begin{cases} \begin{cases} x^{k+1} = \arg \min \left\{ \Phi(x, y^k) + \frac{r}{2} \|x - x^k\|^2 \mid x \in \mathcal{X} \right\}, \\ \bar{x}^k = x^{k+1} + \tau(x^{k+1} - x^k), \end{cases} \\ \begin{cases} \tilde{y}^k = \arg \max \left\{ \Phi(\bar{x}^k, y) - \frac{s}{2} \|y - y^k\|^2 \mid y \in \mathcal{Y} \right\}, \\ y^{k+1} = y^k + \beta(\tilde{y}^k - y^k), \end{cases} \end{cases} \tag{4.9}$$

which is symmetric in the sense that it updates both the variables x and y twice at each iteration. For this special case, the restriction in Theorem 4.2 can be further specified. We present it in the following theorem.

Theorem 4.3 Under the condition (1.3), if $\alpha = 1$; the parameters β and τ satisfy

$$\tau \in \left[\frac{\sqrt{5} - 1}{2}, 1 \right], \quad \text{and} \quad \beta = \frac{1}{\tau}, \tag{4.10}$$

then the condition (4.7a) or (4.7b) is satisfied and thus the PDHG scheme (4.9) is convergent.

Proof If $\alpha = 1$ and $\tau = 1$, it follows from (4.10) that the condition (4.7a) is satisfied. Now we consider $\alpha = 1$ and $\tau \in (0, 1)$. In this case, the condition (4.7b) becomes

$$\tau \in (0, 1) \quad \text{and} \quad 1 \leq 1 + \tau - \sqrt{1 - \tau},$$

and thus

$$\tau \in (0, 1) \quad \text{and} \quad \tau^2 + \tau - 1 \geq 0.$$

The above condition is satisfied for all $\tau \in \left[\frac{\sqrt{5} - 1}{2}, 1 \right)$. The proof is complete. \square

Therefore, with the restriction in (4.10), the PDHG scheme (4.9) can be written as

$$\begin{cases} \begin{cases} x^{k+1} = \arg \min \left\{ \Phi(x, y^k) + \frac{r}{2} \|x - x^k\|^2 \mid x \in \mathcal{X} \right\}, \\ \bar{x}^k = x^{k+1} + \tau(x^{k+1} - x^k), \\ \tilde{y}^k = \arg \max \left\{ \Phi(\bar{x}^k, y) - \frac{s}{2} \|y - y^k\|^2 \mid y \in \mathcal{Y} \right\}, \\ y^{k+1} = y^k + \frac{1}{\tau}(\tilde{y}^k - y^k), \end{cases} & (4.11) \end{cases}$$

where $\tau \in \left[\frac{\sqrt{5} - 1}{2}, 1 \right]$. Indeed, we can further consider choosing $\tau_0 = \frac{\sqrt{5} - 1}{2}$. Then, we have $\frac{1}{\tau_0} = 1 + \tau_0$ and it follows from (4.11) that

$$\begin{aligned} y^k + \frac{1}{\tau_0}(\tilde{y}^k - y^k) &= y^k + (1 + \tau_0)(\tilde{y}^k - y^k) \\ &= \tilde{y}^k + \tau_0(\tilde{y}^k - y^k). \end{aligned}$$

Thus, the scheme (4.11) can be further specified as

$$\begin{cases} \begin{cases} x^{k+1} = \arg \min \left\{ \Phi(x, y^k) + \frac{r}{2} \|x - x^k\|^2 \mid x \in \mathcal{X} \right\}, \\ \bar{x}^k = x^{k+1} + \tau_0(x^{k+1} - x^k), \\ \tilde{y}^k = \arg \max \left\{ \Phi(\bar{x}^k, y) - \frac{s}{2} \|y - y^k\|^2 \mid y \in \mathcal{Y} \right\}, \\ y^{k+1} = \tilde{y}^k + \tau_0(\tilde{y}^k - y^k), \end{cases} & (4.12) \end{cases}$$

where $\tau_0 = \frac{\sqrt{5} - 1}{2}$. It is a completely symmetric PDHG scheme in the sense that both the variables are updated twice at each iteration and the additional updating steps both use the golden-ratio step sizes.

4.2.3 Case III: $\beta = 1$

We can also fix $\beta = 1$. In this case, the algorithmic framework (1.4) reduces to

$$\begin{cases} \tilde{x}^k = \arg \min \left\{ \Phi(x, y^k) + \frac{r}{2} \|x - x^k\|^2 \mid x \in \mathcal{X} \right\}, \\ \tilde{x}^k = \tilde{x}^k + \tau(\tilde{x}^k - x^k), \end{cases} \quad (4.13a)$$

$$\begin{cases} y^{k+1} = \arg \max \left\{ \Phi(\tilde{x}^k, y) - \frac{s}{2} \|y - y^k\|^2 \mid y \in \mathcal{Y} \right\}. \\ x^{k+1} = x^k - \alpha(x^k - \tilde{x}^k). \end{cases} \quad (4.13b)$$

For this case, the restriction on the involved parameters in Theorem 4.2 can be further specified as the following theorem.

Theorem 4.4 *Under the condition (1.3), if $\beta = 1$; the parameters α and τ satisfy*

$$\alpha = \tau \in (0, 1], \quad (4.14)$$

then the convergence condition (4.3) is satisfied.

Proof First, in this case, the condition (4.1) is satisfied. The matrix G in (4.3) becomes

$$G = \begin{pmatrix} (2 - \alpha)rI_n & A^T \\ A & sI_m \end{pmatrix},$$

which is positive definite for all $\alpha \in (0, 1]$ under the assumption (1.3). \square

Therefore, considering the restriction in (4.14), we can specify the scheme (4.13) as

$$\begin{cases} \tilde{x}^k = \arg \min \left\{ \Phi(x, y^k) + \frac{r}{2} \|x - x^k\|^2 \mid x \in \mathcal{X} \right\}, \\ \tilde{x}^k = \tilde{x}^k + \tau(\tilde{x}^k - x^k), \end{cases} \quad (4.15a)$$

$$\begin{cases} y^{k+1} = \arg \max \left\{ \Phi(\tilde{x}^k, y) - \frac{s}{2} \|y - y^k\|^2 \mid y \in \mathcal{Y} \right\}. \\ x^{k+1} = x^k - \tau(x^k - \tilde{x}^k). \end{cases} \quad (4.15b)$$

where $\tau \in (0, 1]$.

5 Convergence Rate

In this section, we establish the worst-case convergence rate measured by the iteration complexity for the algorithmic framework (1.4) in both the ergodic and nonergodic senses.

5.1 Convergence Rate in the Ergodic Sense

We first derive the worst-case convergence rate in the ergodic sense for the algorithmic framework (1.4). For this purpose,

we need to define an approximate solution of the problem (1.1).

Let $(x^*, y^*) \in \Omega^*$ be a saddle point of (1.1). According to the definition (2.1), we have

$$\Phi(x, y^*) - \Phi(x^*, y) \geq 0, \quad \forall x \in \mathcal{X}, y \in \mathcal{Y},$$

which, by using the notation defined in (2.2b), can be specified as

$$\theta(u) - \theta(u^*) + (u - u^*)^T F(u^*) \geq 0, \quad \forall u \in \Omega.$$

That is, $\theta(u) - \theta(u^*) + (u - u^*)^T F(u^*)$ can be used to measure the primal–dual gap of the model (1.1) at the point $u \in \Omega$. It is thus reasonable to define an approximate solution of the problem (1.1) in terms of the primal–dual gap, as stated below.

Definition 5.1 For given $\epsilon > 0$ and a solution point u^* of the problem (1.1), $u \in \Omega$ is called an ϵ -approximate solution of the problem (1.1) if it satisfies

$$0 \leq \theta(u) - \theta(u^*) + (u - u^*)^T F(u^*) \leq \epsilon. \quad (5.1)$$

The following theorem essentially implies that we can find an ϵ -approximate solution of the problem (1.1) based on $O(1/\epsilon)$ iterations generated by the algorithmic framework (1.4).

Theorem 5.2 *Let $\{u^k = (x^k, y^k)\}$ be the sequence generated by the algorithmic framework (1.4) under the conditions (1.3) and (3.1); and $u^* \in \Omega^*$ be a saddle point of (1.1). For any integer $t > 0$, let \tilde{u}_t be defined as*

$$\tilde{u}_t = \frac{1}{t + 1} \sum_{k=0}^t \tilde{u}^k. \quad (5.2)$$

Then we have

$$\begin{aligned} \tilde{u}_t \in \Omega, \quad \theta(\tilde{u}_t) - \theta(u^*) + (\tilde{u}_t - u^*)^T F(u^*) \\ \leq \frac{1}{2(t + 1)} \|u^0 - u^*\|_H^2. \end{aligned} \quad (5.3)$$

Proof First, we know $\tilde{u}^k \in \Omega$ for all $k \geq 0$. Because of the convexity of \mathcal{X} and \mathcal{Y} , it follows from (5.2) that $\tilde{u}_t \in \Omega$. Using (2.3) and (3.2), we have

$$\begin{aligned} \theta(u) - \theta(\tilde{u}^k) + (u - \tilde{u}^k)^T F(u) + \frac{1}{2} \|u - u^k\|_H^2 \\ \geq \frac{1}{2} \|u - u^{k+1}\|_H^2, \quad \forall u \in \Omega. \end{aligned} \quad (5.4)$$

Summing the inequality (5.4) over $k = 0, 1, \dots, t$, we obtain

$$(t + 1)\theta(u) - \sum_{k=0}^t \theta(\tilde{u}^k) + \left((t + 1)u - \sum_{k=0}^t \tilde{u}^k \right)^T F(u) + \frac{1}{2} \|u - u^0\|_H^2 \geq 0, \quad \forall u \in \Omega.$$

Since \tilde{u}_t is defined in (5.2), we have

$$\frac{1}{t + 1} \sum_{k=0}^t \theta(\tilde{u}^k) - \theta(u) + (\tilde{u}_t - u)^T F(u) \leq \frac{1}{2(t + 1)} \|u - u^0\|_H^2, \quad \forall u \in \Omega. \tag{5.5}$$

Further, because of the convexity of $\theta(u)$, we get

$$\theta(\tilde{u}_t) \leq \frac{1}{t + 1} \sum_{k=0}^t \theta(\tilde{u}^k).$$

Substituting it into (5.5) and setting $u = u^*$, the assertion of this theorem follows directly. \square

In the following theorem, we state the worst-case $O(1/t)$ convergence rate measured by the iteration complexity for the algorithmic framework (1.4) in the ergodic sense. The proof is clear based on Theorem 5.2; thus omitted.

Theorem 5.3 *Let $\{u^k = (x^k, y^k)\}$ be the sequence generated by the algorithmic framework (1.4) under the conditions (1.3) and (3.1); and $u^* \in \Omega^*$ be a saddle point of (1.1). For given $\epsilon > 0$, based on t iterations of (1.4) with $t = \lceil \frac{\|u^0 - u^*\|_H^2}{2\epsilon} - 1 \rceil = O(1/\epsilon)$, the point \tilde{u}_t defined as the average of all these t iterations in (5.2) is an ϵ -approximate solution of the problem (1.1) in sense of (5.1), i.e., it satisfies*

$$\tilde{u}_t \in \Omega, \quad 0 \leq \theta(\tilde{u}_t) - \theta(u^*) + (\tilde{u}_t - u^*)^T F(u^*) \leq \epsilon.$$

Finally, we would like to remark that the condition of $G > 0$ in (3.1) can be relaxed to $G \geq 0$ for deriving the convergence rate result in this subsection.

5.2 Convergence Rate in a Nonergodic Sense

In this subsection, we derive the worst-case $O(1/\sqrt{t})$ convergence rate in a nonergodic sense for the algorithmic framework (1.4). Note that a nonergodic worst-case convergence rate is generally stronger than its ergodic counterparts. We first prove a lemma.

Lemma 5.4 *Let $u^k = (x^k, y^k)$ be given by the algorithmic framework (1.4). Under the condition (3.1), we have*

$$(u^k - \tilde{u}^k)^T M^T H M \{ (u^k - \tilde{u}^k) - (u^{k+1} - \tilde{u}^{k+1}) \} \geq \frac{1}{2} \| (u^k - \tilde{u}^k) - (u^{k+1} - \tilde{u}^{k+1}) \|_{(Q^T + Q)}^2. \tag{5.6}$$

Proof Setting $u = \tilde{u}^{k+1}$ in (2.7a), we get

$$\theta(\tilde{u}^{k+1}) - \theta(\tilde{u}^k) + (\tilde{u}^{k+1} - \tilde{u}^k)^T F(\tilde{u}^k) \geq (\tilde{u}^{k+1} - \tilde{u}^k)^T Q (u^k - \tilde{u}^k). \tag{5.7}$$

Note that (2.7a) is also true for $k := k + 1$. Thus, it holds that

$$\theta(u) - \theta(\tilde{u}^{k+1}) + (u - \tilde{u}^{k+1})^T F(\tilde{u}^{k+1}) \geq (u - \tilde{u}^{k+1})^T Q (u^{k+1} - \tilde{u}^{k+1}), \quad \forall u \in \Omega.$$

Then, setting $u = \tilde{u}^k$ in the above inequality, we obtain

$$\theta(\tilde{u}^k) - \theta(\tilde{u}^{k+1}) + (\tilde{u}^k - \tilde{u}^{k+1})^T F(\tilde{u}^{k+1}) \geq (\tilde{u}^k - \tilde{u}^{k+1})^T Q (u^{k+1} - \tilde{u}^{k+1}). \tag{5.8}$$

Combining (5.7) and (5.8), and using the monotonicity of F [see (2.3)], we have

$$(\tilde{u}^k - \tilde{u}^{k+1})^T Q \{ (u^k - \tilde{u}^k) - (u^{k+1} - \tilde{u}^{k+1}) \} \geq 0. \tag{5.9}$$

Adding the term

$$\{ (u^k - \tilde{u}^k) - (u^{k+1} - \tilde{u}^{k+1}) \}^T Q \{ (u^k - \tilde{u}^k) - (u^{k+1} - \tilde{u}^{k+1}) \}$$

to both sides of (5.9), and using $v^T Q v = \frac{1}{2} v^T (Q^T + Q) v$, we obtain

$$(u^k - u^{k+1})^T Q \{ (u^k - \tilde{u}^k) - (u^{k+1} - \tilde{u}^{k+1}) \} \geq \frac{1}{2} \| (u^k - \tilde{u}^k) - (u^{k+1} - \tilde{u}^{k+1}) \|_{(Q^T + Q)}^2.$$

Substituting $(u^k - u^{k+1}) = M(u^k - \tilde{u}^k)$ into the left-hand side of the last inequality and using $Q = HM$, we obtain (5.6) and the lemma is proved. \square

Now, we establish the worst-case $O(1/\sqrt{t})$ convergence rate in a nonergodic sense for the algorithmic framework (1.4). Similar techniques can be found in [18].

Theorem 5.5 *Let $u^k = (x^k, y^k)$ be given by the algorithmic framework (1.4). Under the condition (3.1), for any integer $t > 0$, we have*

$$\|M(u^t - \tilde{u}^t)\|_H^2 \leq \frac{1}{(t + 1)c_0} \|u^0 - u^*\|_H^2, \tag{5.10}$$

where $c_0 > 0$ is a constant independent of t .

Proof First, setting $a = M(u^k - \tilde{u}^k)$ and $b = M(u^{k+1} - \tilde{u}^{k+1})$ in the identity

$$\|a\|_H^2 - \|b\|_H^2 = 2a^T H(a - b) - \|a - b\|_H^2,$$

we obtain

$$\begin{aligned} & \|M(u^k - \tilde{u}^k)\|_H^2 - \|M(u^{k+1} - \tilde{u}^{k+1})\|_H^2 \\ &= 2(u^k - \tilde{u}^k)^T M^T H M [(u^k - \tilde{u}^k) - (u^{k+1} - \tilde{u}^{k+1})] \\ & \quad - \|M[(u^k - \tilde{u}^k) - (u^{k+1} - \tilde{u}^{k+1})]\|_H^2. \end{aligned}$$

Inserting (5.6) into the first term of the right-hand side of the last equality, we obtain

$$\begin{aligned} & \|M(u^k - \tilde{u}^k)\|_H^2 - \|M(u^{k+1} - \tilde{u}^{k+1})\|_H^2 \\ & \geq \|(u^k - \tilde{u}^k) - (u^{k+1} - \tilde{u}^{k+1})\|_{(Q^T + Q)}^2 \\ & \quad - \|M[(u^k - \tilde{u}^k) - (u^{k+1} - \tilde{u}^{k+1})]\|_H^2 \\ & = \|(u^k - \tilde{u}^k) - (u^{k+1} - \tilde{u}^{k+1})\|_G^2 \geq 0. \end{aligned}$$

The last inequality holds because the matrix $(Q^T + Q) - M^T H M = G$ and $G \geq 0$. We thus have

$$\|M(u^{k+1} - \tilde{u}^{k+1})\|_H \leq \|M(u^k - \tilde{u}^k)\|_H, \quad \forall k > 0. \tag{5.11}$$

Recall that the matrix M is positive definite. Thus, the sequence $\{\|M(u^k - \tilde{u}^k)\|_H^2\}$ is monotonically non-increasing. Then, it follows from $G > 0$ and Theorem 3.2 there is a constant $c_0 > 0$ such that

$$\begin{aligned} \|u^{k+1} - u^*\|_H^2 & \leq \|u^k - u^*\|_H^2 - c_0 \|M(u^k - \tilde{u}^k)\|_H^2, \\ & \quad \forall u^* \in \Omega^*. \end{aligned} \tag{5.12}$$

Furthermore, it follows from (5.12) that

$$\sum_{k=0}^{\infty} c_0 \|M(u^k - \tilde{u}^k)\|_H^2 \leq \|u^0 - u^*\|_H^2, \quad \forall u^* \in \Omega^*. \tag{5.13}$$

Therefore, for any integer $t > 0$, we have

$$(t + 1) \|M(u^t - \tilde{u}^t)\|_H^2 \leq \sum_{k=0}^t \|M(u^k - \tilde{u}^k)\|_H^2. \tag{5.14}$$

The assertion (5.10) follows from (5.13) and (5.14) immediately. \square

Let $d := \inf\{\|u^0 - u^*\|_H \mid u^* \in \Omega^*\}$. Then, for any given $\epsilon > 0$, Theorem 5.5 shows that the algorithmic framework (1.4) needs at most $\lceil d^2 / (c_0 \epsilon) \rceil$ iterations to ensure that $\|M(u^k - \tilde{u}^k)\|_H^2 \leq \epsilon$. Hence, if $\|M(u^k - \tilde{u}^k)\|_H$ is regarded as an measurement of the accuracy of u^k to a solution point, Theorem 5.5 indicates a worst-case $O(1/\sqrt{t})$ convergence rate

in a nonergodic sense for the algorithmic framework (1.4). It is worthwhile to mention that the specific sequence $\{u^k\}$ with u^k given by the algorithmic framework (1.4) has the special property of that u^k being a solution point of $VI(\Omega, F, \theta)$ if $\|M(u^k - \tilde{u}^k)\|_H^2 = 0$ (see (2.7a) and due to $Q = HM$). Thus, it is reasonable to use $\|M(u^k - \tilde{u}^k)\|_H$, or $\|M(u^k - \tilde{u}^k)\|_H^2$, to measure the accuracy of an iterate u^k to a solution point.

6 Numerical Results

In this section, we report some preliminary numerical results and verify the efficiency of some specific generalized PDHG algorithms stemming from the proposed algorithmic framework (1.4). The proposed algorithms were coded by MATLAB R2015a, and all our experiments were performed on a desktop with Windows 7 system and an Intel(R) Core(TM) i5-4590 CPU processor (3.30 GHz) with a 8 GB memory.

We consider the standard total variational (TV) blurry and noisy imaging restoration model

$$\min_y \int_D |\nabla y| + \frac{\lambda}{2} \|By - z\|^2, \tag{6.1}$$

where D is the image domain with its area being $|D|$, z is the given observed image, ∇ is a discrete gradient operator (see [25]), B is the matrix representation of a space invariant blurring operator, and $\lambda > 0$ is a constant balancing the data-fidelity and TV regularization terms. As shown in, e.g. [5, 10, 29], the model (6.1) can be reformulated as the saddle point problem

$$\min_{y \in \mathfrak{R}^N} \max_{x \in X} y^T Ax + \frac{\lambda}{2} \|By - z\|^2 \tag{6.2}$$

where X is the Cartesian product of some unit balls in \mathfrak{R}^2 (see, e.g., [5, 16, 25, 29], for details), and A is the matrix representation of the negative divergence operator.

Clearly, (6.2) is a special case of (1.1) and thus the algorithmic framework (1.4) is applicable. We now illustrate how to solve the resulting subproblems (1.4a) when (1.4) is applied to (6.2); more details are referred to, e.g., [5, 16, 29]. For the x -subproblem, it reduces to

$$\tilde{x}^k = \arg \min \left\{ (y^k)^T Ax + \frac{r}{2} \|x - x^k\|^2 \mid x \in X \right\}. \tag{6.3}$$

Hence, the solution is given by

$$\tilde{x}^k = P_X \left[x^k + \frac{1}{r} A^T y^k \right], \tag{6.4}$$

where P_X denotes the projection operator onto X . As mentioned, X is the Cartesian product of some unit balls in \mathfrak{R}^2 ,

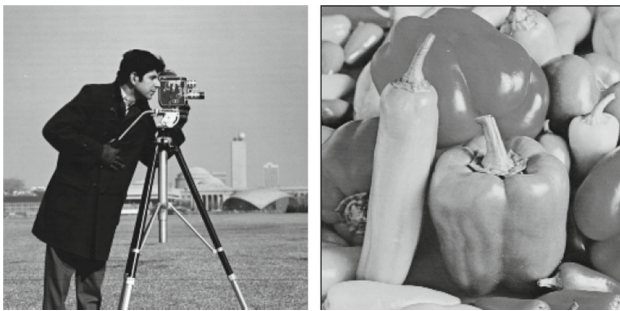


Fig. 1 Original images: cameraman (256 × 256) and pepper (256 × 256)

Table 1 Scenarios to be tested

Gaussian blur	Medium motion blur	Severe motion blur
hsize = 21, sigma = 5	theta = 135, len = 21	theta = 135, len = 91

thus the projection operator P_X can be computed easily. For the y -subproblem, it is

$$\tilde{y}^k = \arg \max \left\{ y^T A \bar{x}^k + \frac{\lambda}{2} \|B y - z\|^2 - \frac{s}{2} \|y - y^k\|^2 \mid y \in \mathfrak{R}^N \right\}. \tag{6.5}$$

Table 2 Three Specific Algorithms Based on (1.4)

Case I	$\tau = 1, \alpha = \beta = 1.8$
Case II	$\tau = \frac{\sqrt{5}-1}{2}, \alpha = 1, \beta = \frac{1}{\tau} = \frac{2}{\sqrt{5}-1}$
Case III	$\tau = \alpha = \beta = 1$

Thus, we need to solve the system of equations

$$(\tilde{y}^k - y^k) + \frac{1}{s} (\lambda B^T (B \tilde{y}^k - z) + A \bar{x}^k) = 0, \tag{6.6}$$

whose solution can be obtained by the Fast Fourier Transform (FFT) or Discrete Cosine Transform (DCT) (see e.g., [29] for details).

We test the grayscale image cameraman (256 × 256) and pepper (256 × 256), as shown in Fig. 1. These images are degraded by convolutions and the zero-mean Gaussian noise with standard deviation $\sigma = 10^{-3}$. The blur operator is generated by the functions **fspecial** and **imfilter** in MATLAB Image Processing Toolbox with the types “motion” and “Gaussian”; zero-mean Gaussian noise is added by the script **imnoise**. The specific scenarios (i.e., the input of the script **fspecial**) under test are listed in Table 1, and the degraded images are shown in Fig. 2. For the parameter λ in (6.2), we take $\lambda = 250$ for the “motion” blur cases and $\lambda = 1000$ for the “Gaussian” blur case.

As we have mentioned, the quantity $\|M(u^k - \tilde{u}^k)\|_H^2$ measures the accuracy of the iterate u^k to a solution point of



Fig. 2 Degraded images: *first row* (cameraman) and *second row* (pepper). From *left to right*: scenarios of Gaussian blur, medium motion blur and severe motion blur as specified in Table 1

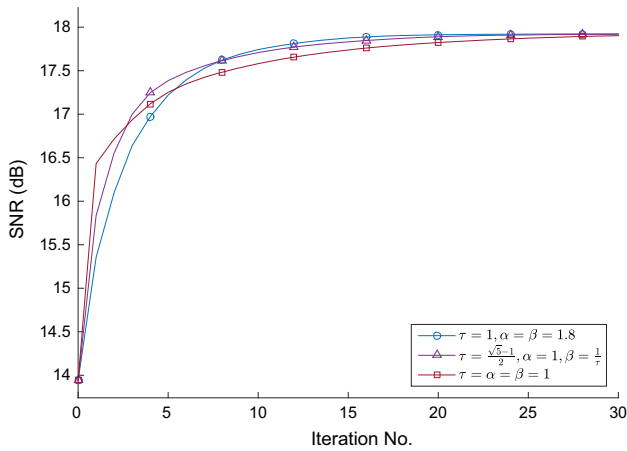


Fig. 3 Evolutions of SNRs with respect to iterations for “Gaussian” blur (cameraman)

$VI(\Omega, F, \theta)$. Therefore, we could use

$$\frac{\|M(u^k - \tilde{u}^k)\|_H^2}{\max\{1, \|Mu^k\|_H^2\}} \leq Tol \tag{6.7}$$

as the stopping criterion, where $Tol > 0$ is an assigned accuracy. The quality of a restored image (denoted by y^k) is measured by the signal-to-noise (SNR) value in decibel (dB):

$$SNR = 20 \log_{10} \frac{\|y^*\|}{\|y^k - y^*\|}, \tag{6.8}$$

where y^* represents a clean image.

Among algorithms based on the algorithmic framework (1.4), we choose three representative cases to test with parameters specified in Table 2. Case I is Algorithm (4.8) in Sect. 4.2.1, and as mentioned, it is Algorithm 4 considered in [16] which is also an improved version of (1.2) using the technique in [14]. As reported in [16], usually $\alpha = 1.8$ is

Table 3 Numerical results for cameraman with $Tol = 10^{-6}$

	Gaussian blur			Medium motion blur		
	It.	CPU	SNR	It.	CPU	SNR
Case I	31	0.26	17.95	21	0.17	27.29
Case II	33	0.26	17.94	17	0.13	27.29
Case III	43	0.33	17.94	21	0.16	27.30

numerically favorable. So we fix it as 1.8 too in our experiments. Case II is the completely symmetric version (4.12) in Sect. 4.2.2; this case is of our interests because of its theoretical beauty. Case III is the generalized PDHG originally considered in [5] which is also a special case of the algorithmic framework (1.4); it usually works very well as widely shown in imaging processing literature, and thus we use it as a benchmark in our experiments. For the step-size parameters r and s in the subproblems of the algorithmic framework (1.4), throughout we fix $1/r = 0.03$ and $s = \frac{10}{9} \cdot 8/r$. Thus, the condition $rs > \|A^T A\|$ is guaranteed to be satisfied, since A^T , the matrix representation of the discrete gradient operator, has $\|A^T A\| \leq 8$ as proved in [4]. Note that our emphasis is to test the possible improvement in speed by combining the step (1.4b) with the original PDHG step (1.4a). So we just fix the values of r and s and observe the difference in α and β in the framework (1.4). We refer to [16] for some empirical study on the step-size parameters r and s . The initial iterates to implement all algorithms are set as $x^0 = 0$ and y^0 be the degraded image. In the following, “It.” and “CPU” represent the iteration numbers and computing time in seconds, respectively.

In Figs. 3 and 4, we plot the evolutions of SNR values with respect to iterations for cameraman. Then, in Table 3, we set $Tol = 10^{-6}$ in the stopping criterion (6.7) and report the respective “It.” and “CPU” for the Gaussian and medium motion blur cases when some well-approximated “maximal”

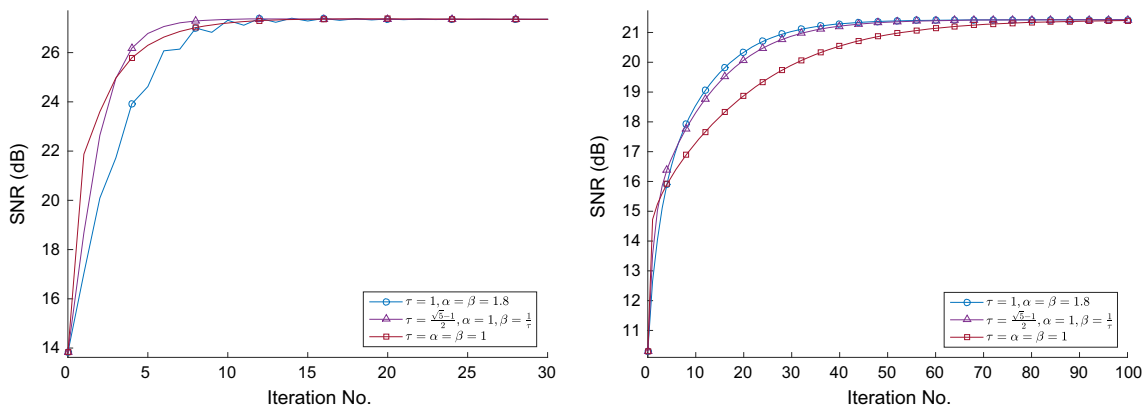


Fig. 4 Evolutions of SNRs with respect to iterations for “motion” blur. *Left*: the scenario of medium motion blur; *right*: the scenario of severe motion blur (cameraman)

Table 4 Numerical results for cameraman with Severe motion blur and different Tol

	$Tol = 10^{-6}$			$Tol = 10^{-7}$			$Tol = 10^{-8}$		
	It.	CPU	SNR	It.	CPU	SNR	It.	CPU	SNR
Case I	33	0.27	21.08	55	0.45	21.39	86	0.71	21.41
Case II	38	0.29	21.12	67	0.50	21.40	111	0.83	21.41
Case III	56	0.42	21.02	99	0.74	21.39	154	1.10	21.41

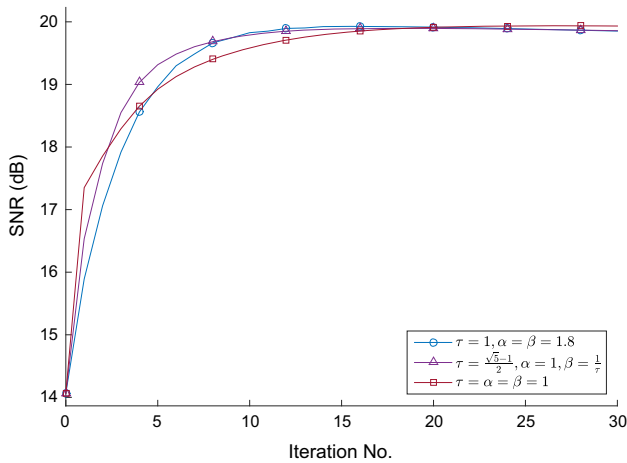


Fig. 5 Evolutions of SNRs with respect to iterations for “Gaussian” blur (pepper)

SNR values (as observed in Figs. 3, 4) are achieved. The reason we set $Tol = 10^{-6}$ is that we found it is sufficiently accurate for these cases to achieve high SNR values. In Table 4, we report the results for the severe motion blur case. Since the model is generally more ill-conditioned with severer blur, $Tol = 10^{-6}$ is not enough to restore images with acceptable quality. So, in this table, we test the case with Tol is as small as 10^{-8} and report all the results when $Tol = 10^{-6}, 10^{-7}$ and 10^{-8} . Figures 5 and 6 and Tables 5 and 6 are the results for the pepper.

Table 5 Numerical results for pepper with $Tol = 10^{-6}$

	Gaussian blur			Medium motion blur		
	It.	CPU	SNR	It.	CPU	SNR
Case I	31	0.25	19.81	21	0.17	27.98
Case II	33	0.25	19.81	16	0.12	27.97
Case III	43	0.32	19.87	20	0.15	27.96

The reported experiment results clearly show the efficiency of some specific generalized PDHG algorithms derived from the algorithmic framework (1.4). In particular, the completely symmetric version (4.12) is competitive with the overrelaxed version (4.8) and they both improve the generalized PDHG in (1.2) favorably. These preliminary numerical results verify the possibility of specifying more efficient algorithms based on the algorithmic framework (1.4).

7 Conclusions

We propose an algorithmic framework of generalized primal-dual hybrid gradient (PDHG) methods for saddle point problems. We theoretically study its convergence and numerically verify the efficiency of some specific algorithms derived from the algorithmic framework. This algorithmic framework includes some existing PDHG schemes as spe-

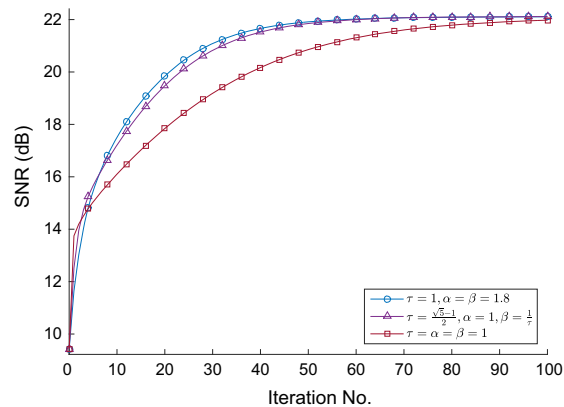
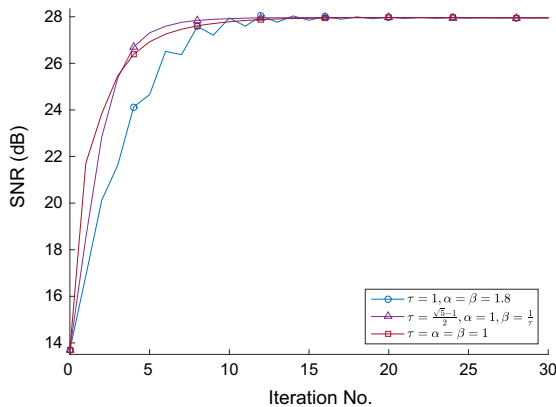


Fig. 6 Evolutions of SNRs with respect to iterations for “motion” blur. *Left*: the scenario of medium motion blur; *right*: the scenario of severe motion blur (pepper)

Table 6 Numerical results for pepper with severe motion blur and different Tol

	Tol = 10^{-6}			Tol = 10^{-7}			Tol = 10^{-8}		
	It.	CPU	SNR	It.	CPU	SNR	It.	CPU	SNR
Case I	34	0.30	21.39	58	0.52	22.03	87	0.78	22.10
Case II	40	0.35	21.55	69	0.60	22.08	109	0.94	22.12
Case III	60	0.49	21.35	104	0.84	22.02	156	1.30	22.10

cial cases, and it yields a class of new generalized PDHG schemes. It also possesses some theoretical advantages such as the worst-case convergence rate measured by the iteration complexity in a nonergodic sense. Our analysis provides a unified perspective to the study of some PDHG schemes for saddle point problems. It is interesting to know if our analysis could be extended to some special nonconvex cases with strong application backgrounds in image processing such as some semiconvex variational image restoration models. We leave it as a research topic in the future.

References

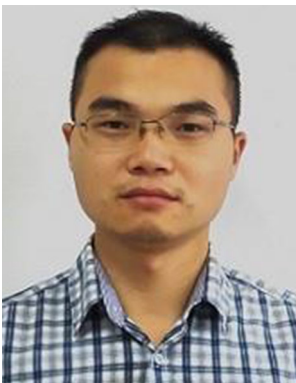
- Arrow, K.J., Hurwicz, L., Uzawa, H.: Studies in linear and non-linear programming. In: Chenery, H.B., Johnson, S.M., Karlin, S., Marschak, T., Solow, R.M. (eds.) *Stanford Mathematical Studies in the Social Science*, vol. II. Stanford University Press, Stanford (1958)
- Blum, E., Oettli, W.: *Mathematische Optimierung Grundlagen und Verfahren*. Springer, Berlin (1975)
- Bonettini, S., Ruggiero, V.: On the convergence of primal–dual hybrid gradient algorithms for total variation image restoration. *J. Math. Imaging Vis.* **44**, 236–253 (2012)
- Chambolle, A.: An algorithm for total variation minimization and applications. *J. Math. Imaging Vis.* **20**, 89–97 (2004)
- Chambolle, A., Pock, T.: A first-order primal–dual algorithms for convex problem with applications to imaging. *J. Math. Imaging Vis.* **40**, 120–145 (2011)
- Chambolle, A., Pock, T.: On the ergodic convergence rates of a first-order primal–dual algorithm. *Math. Program.* **159**, 253–287 (2016)
- Chambolle, A., Pock, T.: A remark on accelerated block coordinate descent for computing the proximity operators of a sum of convex functions. *SMAI J. Comput. Math.* **1**, 29–54 (2015)
- Condat, L.: A primal–dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms. *J. Optim. Theory Appl.* **158**, 460–479 (2013)
- Douglas, J., Rachford, H.H.: On the numerical solution of the heat conduction problem in 2 and 3 space variables. *Trans. Am. Math. Soc.* **82**, 421–439 (1956)
- Esser, E., Zhang, X., Chan, T.F.: A general framework for a class of first order primal–dual algorithms for TV minimization. *SIAM J. Imaging Sci.* **3**, 1015–1046 (2010)
- Facchinei, F., Pang, J.S.: *Finite-Dimensional Variational Inequalities and Complementarity Problems*, Springer Series in Operations Research, vol. I. Springer, New York (2003)
- Glowinski, R., Marocco, A.: Approximation par éléments finis d'ordre un et résolution par pénalisation-dualité d'une classe de problèmes non linéaires. *RAIRO* **R2**, 41–76 (1975)
- Goldstein, T., Li, M., Yuan, X.M.: Adaptive primal–dual splitting methods for statistical learning and image processing. In: *Advances in Neural Information Processing Systems*, pp. 2089–2097 (2015)
- Gol'shtein, E.G., Tret'yakov, N.V.: Modified Lagrangian in convex programming and their generalizations. *Math. Program. Study* **10**, 86–97 (1979)
- He, B.S., You, Y.F., Yuan, X.M.: On the convergence of primal–dual hybrid gradient algorithm. *SIAM J. Imaging Sci.* **7**, 2526–2537 (2014)
- He, B.S., Yuan, X.M.: Convergence analysis of primal–dual algorithms for a saddle-point problem: from contraction perspective. *SIAM J. Imaging Sci.* **5**, 119–149 (2012)
- He, B.S., Yuan, X.M.: On the $O(1/n)$ convergence rate of Douglas–Rachford alternating direction method. *SIAM J. Numer. Anal.* **50**, 700–709 (2012)
- He, B.S., Yuan, X.M.: On non-ergodic convergence rate of Douglas–Rachford alternating directions method of multipliers. *Numer. Math.* **130**, 567–577 (2015)
- Horn, R.A., Johnson, C.R.: *Matrix Analysis*. Cambridge University Press, Cambridge (2012)
- Korpelevich, G.M.: The extragradient method for finding saddle points and other problems. *Ekon. Mat. Metod.* **12**, 747–756 (1976)
- Lions, P.L., Mercier, B.: Splitting algorithms for the sum of two nonlinear operators. *SIAM J. Numer. Anal.* **16**, 964–979 (1979)
- Martinet, B.: Regularization d'inequations variationnelles par approximations successives. *Rev. Fr. Inform. Rech. Oper.* **4**, 154–159 (1970)
- Pock, T., Chambolle, A.: Diagonal preconditioning for first order primal–dual algorithms in convex optimization. In: *IEEE International Conference on Computer Vision*, pp. 1762–1769 (2011)
- Popov, L.D.: A modification of the Arrow–Hurwitz method of search for saddle points. *Mat. Zamet.* **28**(5), 777–784 (1980)
- Rudin, L., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. *Phys. D* **60**, 227–238 (1992)
- Shefi, R.: Rate of convergence analysis for convex non-smooth optimization algorithms. PhD Thesis, Tel Aviv University, Israel (2015)
- Weiss, P., Blanc-Feraud, L., Aubert, G.: Efficient schemes for total variation minimization under constraints in image processing. *SIAM J. Sci. Comput.* **31**(3), 2047–2080 (2009)
- Zhang, X., Burger, M., Osher, S.: A unified primal–dual algorithm framework based on Bregman iteration. *J. Sci. Comput.* **46**(1), 20–46 (2010)
- Zhu, M., Chan, T.F.: An efficient primal–dual hybrid gradient algorithm for total variation image restoration. CAM Report 08–34, UCLA, Los Angeles, CA (2008)



Bingsheng He is Professor at Department of Mathematics, Southern University of Science and Technology, and Nanjing University, China. His main research interests are numerical optimization, especially splitting methods for monotone variational inequalities and convex optimization.



Xiaoming Yuan is Professor at Department of Mathematics, Hong Kong Baptist University, Hong Kong. His main research interests are numerical optimization and its applications in image processing, statistical learning, and scientific computing.



Feng Ma is currently lecturer at the High-Tech Institute of Xi'an, China. His work focuses on numerical optimization.