SIAM J. IMAGING SCIENCES
Vol. 3, No. 4, pp. 1015–1046

A General Framework for a Class of First Order Primal-Dual Algorithms for Convex Optimization in Imaging Science*

Ernie Esser†, Xiaoqun Zhang‡, and Tony F. Chan§

Abstract. We generalize the primal-dual hybrid gradient (PDHG) algorithm proposed by Zhu and Chan in [*An Efficient Primal-Dual Hybrid Gradient Algorithm for Total Variation Image Restoration*, CAM Report 08-34, UCLA, Los Angeles, CA, 2008] to a broader class of convex optimization problems. In addition, we survey several closely related methods and explain the connections to PDHG. We point out convergence results for a modified version of PDHG that has a similarly good empirical convergence rate for total variation (TV) minimization problems. We also prove a convergence result for PDHG applied to TV denoising with some restrictions on the PDHG step size parameters. We show how to interpret this special case as a projected averaged gradient method applied to the dual functional. We discuss the range of parameters for which these methods can be shown to converge. We also present some numerical comparisons of these algorithms applied to TV denoising, TV deblurring, and constrained l_1 minimization problems.

Key words. convex optimization, total variation minimization, primal-dual methods, operator splitting, l_1 basis pursuit

AMS subject classifications. 90C25, 90C06, 49K35, 49N45, 65K10

DOI. 10.1137/09076934X

1. Introduction.

Total variation (TV) minimization problems arise in many image processing applications for regularizing inverse problems where one expects the recovered image or signal to be piecewise constant or have a sparse gradient. However, a lack of differentiability makes minimizing TV regularized functionals computationally challenging, and so there is considerable interest in efficient algorithms, especially for large scale problems. More generally, there is interest in practical methods for solving nondifferentiable convex optimization problems, TV minimization being an important special case.

The primal-dual hybrid gradient (PDHG) algorithm [58] in a general setting is a method for solving problems of the form

$$\min_{u\in\mathbb{R}^m} J(Au) + H(u),$$

where J and H are closed proper convex functions and $A \in \mathbb{R}^{n\times m}$. Usually, $J(Au)$ will correspond to a regularizing term of the form $\|Au\|$, in which case the PDHG method works

*Received by the editors August 31, 2009; accepted for publication (in revised form) September 17, 2010; published electronically December 14, 2010. This work was supported by ONR grant N00014-03-1-0071, NSF grant DMS-0610079, NSF grant CCF-0528583, and NSF grant DMS-0312222.
http://www.siam.org/journals/siims/3-4/76934.html
†Department of Mathematics, University of California at Irvine, Irvine, CA 92617 (eesser@math.uci.edu).
‡Department of Mathematics, Institute of Natural Science, Shanghai Jiaotong University, Shanghai 200240, China (xqzhang@sjtu.edu.cn).
§Office of the President, Hong Kong University of Science and Technology (HKUST), Clear Water Bay, Kowloon, Hong Kong (tonyfchan@ust.hk).

by using duality to rewrite it as the saddle point problem

$$\min_{u\in\mathbb{R}^m}\max_{\|p\|_*\le 1}\langle p, Au\rangle + H(u)$$

and then alternating dual and primal steps of the form

$$p^{k+1} = \arg\max_{\|p\|_*\le 1}\langle p, Au^k\rangle - \frac{1}{2\delta_k}\|p-p^k\|_2^2,$$
$$u^{k+1} = \arg\min_{u\in\mathbb{R}^m}\langle p^{k+1}, Au\rangle + H(u) + \frac{1}{2\alpha_k}\|u-u^k\|_2^2$$

for appropriate parameters α_k and δ_k. Here, $\|\cdot\|$ denotes an arbitrary norm on $\mathbb{R}^n$ and $\|\cdot\|_*$ denotes its dual norm defined by

$$\|x\|_* = \max_{\|y\|\le 1}\langle x, y\rangle,$$

where $\langle\cdot,\cdot\rangle$ is the standard Euclidean inner product. Formulating the saddle point problem uses the fact that $\|\cdot\|_{**} = \|\cdot\|$ [32], from which it follows that $\|Au\| = \max_{\|p\|_*\le 1}\langle p, Au\rangle$.

PDHG can also be applied to more general convex optimization problems. However, its performance for problems such as TV denoising is of special interest since it compares favorably with other popular methods. An adaptive time stepping scheme for PDHG was proposed in [58] and shown to outperform other popular TV denoising algorithms like Chambolle's method [10], the method of Chan, Golub, and Mulet (CGM) [13], fast total variation deconvolution (FTVd) [51], and split Bregman [29] in many numerical experiments with a wide variety of stopping conditions. Aside from some special cases of the PDHG algorithm such as gradient projection and subgradient descent, the theoretical convergence properties were not known.

PDHG is an example of a first order method, meaning that it requires only functional and gradient evaluations. Other examples of first order methods popular for TV minimization include gradient descent, Chambolle's method, FTVd, and split Bregman. Second order methods, such as CGM and semismooth Newton approaches [30, 31, 17], work by essentially applying Newton's method to an appropriate formulation of the optimality conditions and therefore also require information about the Hessian. This usually requires some smoothing of the objective functional. These methods can be superlinearly convergent and are therefore useful for computing benchmark solutions of high accuracy. However, the cost per iteration is usually higher, so for large scale problems or when high accuracy is not required, these are often less practical than the first order methods that have much lower cost per iteration. Here, we will focus on a class of first order methods related to PDHG that are simple to implement and can also be directly applied to nondifferentiable functionals.

PDHG is also an example of a primal-dual method. Each iteration updates both a primal and a dual variable. It is thus able to avoid some of the difficulties that arise when working only on the primal or the dual side. For example, for TV minimization, gradient descent applied to the primal functional has trouble where the gradient of the solution is zero because the functional is not differentiable there. Chambolle's method is a method on the dual that is very effective for TV denoising, but does not easily extend to applications where the dual problem is more complicated, such as TV deblurring. Primal-dual algorithms can avoid these difficulties

to some extent. Other examples include CGM, the semismooth Newton approaches mentioned above, split Bregman, and, more generally, other Bregman iterative algorithms [56, 55, 54] and Lagrangian-based methods.

In this paper we show that we can make a small modification to the PDHG algorithm which has little effect on its performance but allows the modified algorithm to be interpreted as a special case of a split inexact Uzawa method that is analyzed and shown to converge in [57]. After initial preparation of this paper, it was brought to our attention that the specific modified PDHG algorithm applied here had been previously proposed by Pock et al. [40] for minimizing the Mumford–Shah functional. In [40] the authors also proved convergence for a special class of saddle point problems. In recent preprints [11, 12] that appeared during the current paper's review process, this convergence argument has been generalized and gives a stronger statement of the convergence of the modified PDHG algorithm for the same range of fixed parameters. Chambolle and Pock also provide a convergence rate analysis in [12]. While the modified PDHG method with fixed step sizes is nearly as effective as fixed parameter versions of PDHG, well-chosen adaptive step sizes can improve the rate of convergence. It is proved in [12] that certain adaptive step size schemes accelerate the convergence rate of the modified PDHG method in cases when the objective functional has additional regularity. With more restrictions on the step size parameters, we prove a convergence result for the original PDHG method applied to TV denoising by interpreting it as a projected averaged gradient method on the dual.

We additionally show that the modified PDHG method can be applied in the same ways PDHG was extended in [58] to apply to additional problems such as TV deblurring, l_1 minimization, and constrained minimization problems. For these applications we point out the range of parameters for which the convergence theory is applicable.

Another contribution of this paper is to describe a general algorithm framework from the perspective of PDHG that explains the close connections to modified PDHG, split inexact Uzawa, and more classical methods including proximal forward backward splitting (PFBS) [34, 39, 15], the alternating minimization algorithm (AMA) [50], the alternating direction method of multipliers (ADMM) [25, 27, 6], and Douglas–Rachford splitting [18, 24, 26, 19, 20]. These connections provide some additional insight about where PDHG and modified PDHG fit relative to existing methods.

The organization of this paper is as follows. In section 2, we discuss primal-dual formulations for a general problem. We define a general version of PDHG and discuss in detail the framework in which it can be related to other similar algorithms. These connections are diagrammed in Figure 1. In section 3, we define a discretization of the TV seminorm and review the details about applying PDHG to TV deblurring–type problems. In section 4, we show how to interpret PDHG applied to TV denoising as a projected averaged gradient method on the dual and present a convergence result for a special case. Then in section 5, we discuss the application of the modified PDHG algorithm to constrained TV and l_1 minimization problems. Section 6 presents numerical experiments for TV denoising, constrained TV deblurring, and constrained l_1 minimization, comparing the performance of the modified PDHG algorithm with that of other methods.

2. General algorithm framework. In this section we consider a general class of problems to which PDHG can be applied. We define equivalent primal, dual, and several primal-dual

formulations. We also place PDHG in a general framework that connects it to other related alternating direction methods applied to saddle point problems.

2.1. Primal-dual formulations. PDHG can more generally be applied to what we will refer to as the primal problem,

$$\min_{u\in\mathbb{R}^m} F_P(u), \tag{P}$$

where

$$F_P(u) = J(Au) + H(u), \tag{2.1}$$

$A \in \mathbb{R}^{n\times m}$, and $J : \mathbb{R}^n \to (-\infty, \infty]$ and $H : \mathbb{R}^m \to (-\infty, \infty]$ are closed proper convex functions. Assume that there exists a solution u^* to (P). So that we can use Fenchel duality [44, Corollary 31.2.1] later, we also assume that there exists $u \in \operatorname{ri}(\operatorname{dom} H)$ such that $Au \in \operatorname{ri}(\operatorname{dom} J)$, which is almost always true in practice. When J is a norm, it is shown how to use the dual norm to define a saddle point formulation of (P) as

$$\min_{u\in\mathbb{R}^m} \max_{\|p\|_*\le 1} \langle Au, p\rangle + H(u).$$

This can equivalently be written in terms of the Legendre–Fenchel transform, or convex conjugate, of J, denoted by J^* and defined by

$$J^*(p) = \sup_{w\in\mathbb{R}^n} \langle p, w\rangle - J(w).$$

When J is a closed proper convex function, we have that $J^{**} = J$ [21]. Therefore,

$$J(Au) = \sup_{p\in\mathbb{R}^n} \langle p, Au\rangle - J^*(p).$$

Thus an equivalent saddle point formulation of (P) is

$$\min_{u\in\mathbb{R}^m} \sup_{p\in\mathbb{R}^n} L_{PD}(u, p), \tag{PD}$$

where

$$L_{PD} = \langle p, Au\rangle - J^*(p) + H(u). \tag{2.2}$$

This holds even when J is not a norm, but in the case when $J(w) = \|w\|$, we can then use the dual norm representation of $\|w\|$ to write

$$\begin{aligned} J^*(p) &= \sup_w \langle p, w\rangle - \max_{\|y\|_*\le 1} \langle w, y\rangle \\ &= \begin{cases} 0 & \text{if } \|p\|_* \le 1, \\ \infty & \text{otherwise,} \end{cases} \end{aligned}$$

in which case we can interpret J^* as the indicator function for the unit ball in the dual norm.

Let (u^*, p^*) be a saddle point of L_{PD}. In particular, this means that

$$\max_{p\in\mathbb{R}^n} \langle p, Au^*\rangle - J^*(p) + H(u^*) = L_{PD}(u^*, p^*) = \min_{u\in\mathbb{R}^m} \langle p^*, Au\rangle + H(u) - J^*(p^*),$$

from which we can deduce the equivalent optimality conditions and then use the definitions of the Legendre transform and subdifferential to write these conditions in two ways:

$$-A^T p^* \in \partial H(u^*) \quad \Leftrightarrow \quad u^* \in \partial H^*(-A^T p^*), \tag{2.3}$$

$$Au^* \in \partial J^*(p^*) \quad \Leftrightarrow \quad p^* \in \partial J(Au^*), \tag{2.4}$$

where ∂ denotes the subdifferential. The subdifferential $\partial F(x)$ of a convex function $F : \mathbb{R}^m \to (-\infty, \infty]$ at the point x is defined by the set

$$\partial F(x) = \{q \in \mathbb{R}^m : F(y) \geq F(x) + \langle q, y - x\rangle \ \forall y \in \mathbb{R}^m\}.$$

Another useful saddle point formulation, which we will refer to as the split primal problem, is obtained by introducing the constraint $w = Au$ in (P) and forming the Lagrangian

$$L_P(u, w, p) = J(w) + H(u) + \langle p, Au - w\rangle. \tag{2.5}$$

The corresponding saddle point problem is

$$\max_{p\in\mathbb{R}^n} \inf_{u\in\mathbb{R}^m, w\in\mathbb{R}^n} L_P(u, w, p). \tag{SP$_{\mathrm{P}}$}$$

Although p was introduced in (2.5) as a Lagrange multiplier for the constraint $Au = w$, it has the same interpretation as the dual variable p in (PD). It follows immediately from the optimality conditions that if (u^*, w^*, p^*) is a saddle point for (SP$_{\mathrm{P}}$), then (u^*, p^*) is a saddle point for (PD).

The dual problem is

$$\max_{p\in\mathbb{R}^n} F_D(p), \tag{D}$$

where the dual functional $F_D(p)$ is a concave function defined by

$$F_D(p) = \inf_{u\in\mathbb{R}^m} L_{PD}(u, p) = \inf_{u\in\mathbb{R}^m} \langle p, Au\rangle - J^*(p) + H(u) = -J^*(p) - H^*(-A^T p). \tag{2.6}$$

Note that this is equivalent to defining the dual by

$$F_D(p) = \inf_{u\in\mathbb{R}^m, w\in\mathbb{R}^n} L_P(u, w, p). \tag{2.7}$$

Since we assumed that there exists an optimal solution u^* to the convex problem (P), it follows from Fenchel duality [44, Corollary 31.2.1] that there exists an optimal solution p^* to (D) and $F_P(u^*) = F_D(p^*)$. Moreover, u^* solves (P) and p^* solves (D) if and only if (u^*, p^*) is a saddle point of $L_{PD}(u, p)$ [44, Lemma 36.2].

By introducing the constraint $y = -A^T p$ in (D) and forming the corresponding Lagrangian

$$L_D(p, y, u) = J^*(p) + H^*(y) + \langle u, -A^T p - y\rangle, \tag{2.8}$$

we obtain yet another saddle point problem,

$$\max_{u\in\mathbb{R}^m} \inf_{p\in\mathbb{R}^n, y\in\mathbb{R}^m} L_D(p,y,u), \tag{$\mathrm{SP_D}$}$$

which we will refer to as the split dual problem. Although u was introduced in ($\mathrm{SP_D}$) as a Lagrange multiplier for the constraint $y = -A^T p$, it actually has the same interpretation as the primal variable u in (P). Again, it follows from the optimality conditions that if (p^*, y^*, u^*) is a saddle point for ($\mathrm{SP_D}$), then (u^*, p^*) is a saddle point for (PD). Note also that

$$F_P(u) = -\inf_{p\in\mathbb{R}^n, y\in\mathbb{R}^m} L_D(p,y,u).$$

2.2. Algorithm framework and connections to PDHG. In this section we define a general version of PDHG applied to (PD) and discuss connections to related algorithms that can be interpreted as alternating direction methods applied to ($\mathrm{SP_P}$) and ($\mathrm{SP_D}$). These connections are summarized in Figure 1.

The main tool for drawing connections between the algorithms in this section is the Moreau decomposition [35, 15].

Theorem 2.1 (see [15]). *If J is a closed proper convex function on $\mathbb{R}^m$ and $f \in \mathbb{R}^m$, then*

$$f = \arg\min_u J(u) + \frac{1}{2\alpha}\|u-f\|_2^2 + \alpha \arg\min_p J^*(p) + \frac{\alpha}{2}\left\|p - \frac{f}{\alpha}\right\|_2^2. \tag{2.9}$$

It was shown in [58] that PDHG applied to TV denoising can be interpreted as a primal-dual proximal point method applied to a saddle point formulation of the problem [43]. More generally, applied to (PD) it yields the following algorithm.

Algorithm. PDHG on (PD).

$$p^{k+1} = \arg\max_{p\in\mathbb{R}^n} -J^*(p) + \langle p, Au^k\rangle - \frac{1}{2\delta_k}\|p-p^k\|_2^2, \tag{2.10a}$$

$$u^{k+1} = \arg\min_{u\in\mathbb{R}^m} H(u) + \langle A^T p^{k+1}, u\rangle + \frac{1}{2\alpha_k}\|u-u^k\|_2^2. \tag{2.10b}$$

Here, p^0, u^0 are arbitrary and $\alpha_k, \delta_k > 0$.

2.2.1. Proximal forward backward splitting: Special cases of PDHG. Two notable special cases of PDHG are $\alpha_k = \infty$ and $\delta_k = \infty$. These special cases correspond to the PFBS method [34, 39, 15] applied to (D) and (P), respectively.

PFBS is an iterative splitting method that can be used to find a minimum of a sum of two convex functionals by alternating a (sub-)gradient descent step with a proximal step. Applied to (D) it yields

$$p^{k+1} = \arg\min_{p\in\mathbb{R}^n} J^*(p) + \frac{1}{2\delta_k}\|p-(p^k+\delta_k Au^{k+1})\|_2^2, \tag{2.11}$$

where $u^{k+1} \in \partial H^*(-A^T p^k)$. Since $u^{k+1} \in \partial H^*(-A^T p^k) \Leftrightarrow -A^T p^k \in \partial H(u^{k+1})$, which is equivalent to

$$u^{k+1} = \arg\min_{u\in\mathbb{R}^m} H(u) + \langle A^T p^k, u\rangle,$$

(2.11) can be written as the following algorithm.

ALGORITHM. PFBS on (D).

$$u^{k+1} = \arg\min_{u\in\mathbb{R}^m} H(u) + \langle A^T p^k, u\rangle, \tag{2.12a}$$

$$p^{k+1} = \arg\min_{p\in\mathbb{R}^n} J^*(p) + \langle p, -Au^{k+1}\rangle + \frac{1}{2\delta_k}\|p - p^k\|_2^2. \tag{2.12b}$$

Even though the order of the updates is reversed relative to PDHG, since the initialization is arbitrary, it is still a special case of (2.10), where $\alpha_k = \infty$.

If we assume that $J(\cdot) = \|\cdot\|$, we can interpret the p^{k+1} step as an orthogonal projection onto a convex set,

$$p^{k+1} = \Pi_{\{p:\|p\|_*\le 1\}}\left(p^k + \delta_k A u^{k+1}\right).$$

Then PFBS applied to (D) can be interpreted as a (sub-)gradient projection algorithm.

As a special case of [15, Theorem 3.4], the following convergence result applies to (2.12).

Theorem 2.2. *Fix $p^0 \in \mathbb{R}^n$, $u^0 \in \mathbb{R}^m$ and let (u^k, p^k) be defined by (2.12). If H^* is differentiable, $\nabla(H^*(-A^T p))$ is Lipschitz continuous with Lipschitz constant equal to $\frac{1}{\beta}$, and $0 < \inf \delta_k \le \sup \delta_k < 2\beta$, then $\{p^k\}$ converges to a solution of (D) and $\{u^k\}$ converges to the unique solution of (P).*

Proof. Convergence of $\{p^k\}$ to a solution of (D) follows from [15, Theorem 3.4]. From (2.12a), u^{k+1} satisfies $-A^T p^k \in \partial H(u^{k+1})$, which, from the definitions of the subdifferential and Legendre transform, implies that $u^{k+1} = \nabla H^*(-A^T p^k)$. So by continuity of ∇H^*, $u^k \to u^* = \nabla H^*(-A^T p^*)$. From (2.12b) and the convergence of $\{p^k\}$, $Au^* \in \partial J^*(p^*)$. Therefore (u^*, p^*) satisfies the optimality conditions (2.3), (2.4) for (PD), which means u^* solves (P) [44, Theorem 31.3]. Uniqueness follows from the assumption that H^* is differentiable, which by [44, Theorem 26.3] means that $H(u)$ in the primal functional is strictly convex. ∎

It will be shown in section 2.2.3 how to equate modified versions of the PDHG algorithm with convergent alternating direction methods, namely, split inexact Uzawa methods from [57] applied to the split primal ($\mathrm{SP_P}$) and split dual ($\mathrm{SP_D}$) problems. The connection there is very similar to the equivalence from [50] between PFBS applied to (D) and what Tseng in [50] called the alternating minimization algorithm (AMA) applied to ($\mathrm{SP_P}$). AMA applied to ($\mathrm{SP_P}$) is an alternating direction method that alternately minimizes first the Lagrangian $L_P(u, w, p)$ with respect to u and then the augmented Lagrangian $L_P + \frac{\delta_k}{2}\|Au - w\|_2^2$ with respect to w before updating the Lagrange multiplier p.

ALGORITHM. AMA on ($\mathrm{SP_P}$).

$$u^{k+1} = \arg\min_{u\in\mathbb{R}^m} H(u) + \langle A^T p^k, u\rangle, \tag{2.13a}$$

$$w^{k+1} = \arg\min_{w\in\mathbb{R}^n} J(w) - \langle p^k, w\rangle + \frac{\delta_k}{2}\|Au^{k+1} - w\|_2^2, \tag{2.13b}$$

$$p^{k+1} = p^k + \delta_k(Au^{k+1} - w^{k+1}). \tag{2.13c}$$

To see the equivalence between (2.12) and (2.13), first note that (2.13a) is identical to (2.12a), so it suffices to show that (2.13b) and (2.13c) together are equivalent to (2.12b). Combining (2.13b) and (2.13c) yields

$$p^{k+1} = (p^k + \delta_k A u^{k+1}) - \delta_k \arg\min_w J(w) + \frac{\delta_k}{2}\left\| w - \frac{(p^k + \delta_k A u^{k+1})}{\delta_k}\right\|_2^2.$$

By the Moreau decomposition (Theorem 2.1), this is equivalent to

$$p^{k+1} = \arg\min_p J^*(p) + \frac{1}{2\delta_k}\|p - (p^k + \delta_k A u^{k+1})\|_2^2,$$

which is exactly (2.12b).

In [50], convergence of (u^k, w^k, p^k) satisfying (2.13) to a saddle point of $L_P(u, w, p)$ is directly proved under the assumption that H is strongly convex, an assumption that directly implies the condition on H^* in Theorem 2.2.

The other special case of PDHG, where $\delta_k = \infty$, can be analyzed in a similar manner. The following corresponding algorithm is PFBS applied to (P).

Algorithm. PFBS on (P).

$$p^{k+1} = \arg\min_{p\in\mathbb{R}^n} J^*(p) + \langle -Au^k, p\rangle, \tag{2.14a}$$

$$u^{k+1} = \arg\min_{u\in\mathbb{R}^m} H(u) + \langle u, A^T p^{k+1}\rangle + \frac{1}{2\alpha_k}\|u - u^k\|_2^2. \tag{2.14b}$$

This is analogously equivalent to AMA applied to ($\mathrm{SP_D}$), as follows.

Algorithm. AMA on ($\mathrm{SP_D}$).

$$p^{k+1} = \arg\min_{p\in\mathbb{R}^m} J^*(p) + \langle -Au^k, p\rangle, \tag{2.15a}$$

$$y^{k+1} = \arg\min_{y\in\mathbb{R}^m} H^*(y) - \langle u^k, y\rangle + \frac{\alpha_k}{2}\|y + A^T p^{k+1}\|_2^2, \tag{2.15b}$$

$$u^{k+1} = u^k + \alpha_k(-A^T p^{k+1} - y^{k+1}). \tag{2.15c}$$

The equivalence again follows from the Moreau decomposition (Theorem 2.1), and the analogous version of Theorem 2.2 applies to (2.14).

2.2.2. Reinterpretation of PDHG as relaxed AMA. The general form of PDHG (2.10) can also be interpreted as alternating direction methods applied to ($\mathrm{SP_P}$) or ($\mathrm{SP_D}$). These interpretations turn out to be relaxed forms of AMA. They can be obtained by modifying the objective functional for the Lagrangian minimization step by adding either $\frac{1}{2\alpha_k}\|u - u^k\|_2^2$ to (2.13a) or $\frac{1}{2\delta_k}\|p - p^k\|_2^2$ to (2.15a). The equivalence of these relaxed AMAs to the general form of PDHG (2.10) follows by an argument similar to that in section 2.2.1.

Although equating PDHG to this relaxed AMA does not yield any direct convergence results for PDHG, it does show a close connection to the alternating direction method of multipliers (ADMM) [25, 27, 6], which does have a well-established convergence theory [20]. If, instead of adding proximal terms of the form $\frac{1}{2\alpha_k}\|u-u^k\|_2^2$ and $\frac{1}{2\delta_k}\|p-p^k\|_2^2$ to the first step of AMA applied to ($\mathrm{SP_P}$) and ($\mathrm{SP_D}$), we fix α and δ and add the augmented Lagrangian penalties $\frac{\delta}{2}\|Au-w^k\|_2^2$ and $\frac{\alpha}{2}\|A^Tp+y^k\|_2^2$, then we get exactly ADMM applied to ($\mathrm{SP_P}$) and ($\mathrm{SP_D}$), respectively.

ADMM applied to ($\mathrm{SP_P}$) can be interpreted as Douglas–Rachford splitting [18] applied to (D), and ADMM applied to ($\mathrm{SP_D}$) can be interpreted as Douglas–Rachford splitting applied to (P) [24, 26, 19, 20]. It is also shown in [23, 46, 53] how to interpret these as the split Bregman algorithm of [29]. A general convergence result for ADMM can be found in [20].

2.2.3. Modifications of PDHG. In this section we show that two slightly modified versions of the PDHG algorithm, denoted PDHGMp and PDHGMu, can be interpreted as a split inexact Uzawa method from [57] applied to ($\mathrm{SP_P}$) and ($\mathrm{SP_D}$), respectively. In the constant step size case, PDHGMp replaces p^{k+1} in the u^{k+1} step (2.10b) with $2p^{k+1}-p^k$, whereas PDHGMu replaces u^k in the p^{k+1} step (2.10a) with $2u^k-u^{k-1}$. The variable step size case will also be discussed. For appropriate parameter choices these modified algorithms are nearly as efficient as PDHG numerically, and known convergence results [57, 11, 12] can be applied. Convergence of PDHGMu for a special class of saddle point problems is also proved in [40] based on an argument in [41].

The split inexact Uzawa method from [57] applied to ($\mathrm{SP_D}$) can be thought of as a modification of ADMM. Applying the main idea of the Bregman operator splitting algorithm from [56], it adds $\frac{1}{2}\langle p-p^k,(\frac{1}{\delta_k}I-\alpha_kAA^T)(p-p^k)\rangle$ to the penalty term $\frac{\alpha_k}{2}\|A^Tp+y^k\|_2^2$ in the objective functional for the first minimization step. To ensure $\frac{1}{\delta_k}I-\alpha_kAA^T$ is positive definite, choose $0<\delta_k<\frac{1}{\alpha_k\|A\|^2}$. Adding this extra term, as in the surrogate functional approach of [16], has the effect of linearizing the penalty term and decoupling the variables previously coupled by the matrix A^T. The updates for y^{k+1} and u^{k+1} remain the same as for ADMM. By combining terms for the p^{k+1} update, the resulting algorithm can be written as follows.

Algorithm. Split inexact Uzawa applied to ($\mathrm{SP_D}$).

$$p^{k+1}=\arg\min_{p\in\mathbb{R}^n} J^*(p)+\langle -Au^k,p\rangle+\frac{1}{2\delta_k}\|p-p^k+\alpha_k\delta_kA(A^Tp^k+y^k)\|_2^2, \tag{2.16a}$$

$$y^{k+1}=\arg\min_{y\in\mathbb{R}^m} H^*(y)-\langle u^k,y\rangle+\frac{\alpha_k}{2}\|y+A^Tp^{k+1}\|_2^2, \tag{2.16b}$$

$$u^{k+1}=u^k+\alpha_k(-A^Tp^{k+1}-y^{k+1}). \tag{2.16c}$$

The above algorithm can be shown to converge at least for fixed step sizes α and δ satisfying $0<\delta<\frac{1}{\alpha\|A\|^2}$.

Theorem 2.3 (see [57]). *Let $\alpha_k=\alpha>0$, $\delta_k=\delta>0$, and $0<\delta<\frac{1}{\alpha\|A\|^2}$. Let (p^k,y^k,u^k) satisfy (2.16). Also let p^* be optimal for (D) and $y^*=-A^Tp^*$. Then*

- $\|A^Tp^k+y^k\|_2\to 0$,

- $J^*(p^k) \to J^*(p^*)$,
- $H^*(y^k) \to H^*(y^*)$,

and all convergent subsequences of (p^k, y^k, u^k) *converge to a saddle point of* L_D (2.8).

Moreover, the split inexact Uzawa algorithm can be rewritten in a form that is very similar to PDHG. Since the y^{k+1} (2.16b) and u^{k+1} (2.16c) steps are the same as those for AMA on ($\mathrm{SP_D}$) (2.15), by the same argument they are equivalent to the u^{k+1} update in PDHG (2.10b). From (2.16c), we have that

$$y^k = \frac{u^{k-1}}{\alpha_{k-1}} - \frac{u^k}{\alpha_{k-1}} - A^T p^k. \tag{2.17}$$

Substituting this into (2.16a), we see that (2.16) is equivalent to a modified form of PDHG, where u^k is replaced by $\left((1 + \frac{\alpha_k}{\alpha_{k-1}})u^k - \frac{\alpha_k}{\alpha_{k-1}}u^{k-1}\right)$ in (2.10a). The resulting form of the algorithm, which follows, will be denoted PDHGMu.

Algorithm. PDHGMu.

$$p^{k+1} = \arg\min_{p\in\mathbb{R}^n} J^*(p) + \left\langle p, -A\left(\left(1 + \frac{\alpha_k}{\alpha_{k-1}}\right)u^k - \frac{\alpha_k}{\alpha_{k-1}}u^{k-1}\right)\right\rangle + \frac{1}{2\delta_k}\|p - p^k\|_2^2, \tag{2.18a}$$

$$u^{k+1} = \arg\min_{u\in\mathbb{R}^m} H(u) + \langle A^T p^{k+1}, u\rangle + \frac{1}{2\alpha_k}\|u - u^k\|_2^2. \tag{2.18b}$$

Note that from (2.17) and (2.18b), $y^{k+1} \in \partial H(u^{k+1})$, which we could substitute instead of (2.17) into (2.16a) to get an equivalent version of PDHGMu, whose updates depend only on the previous iteration instead of the previous two.

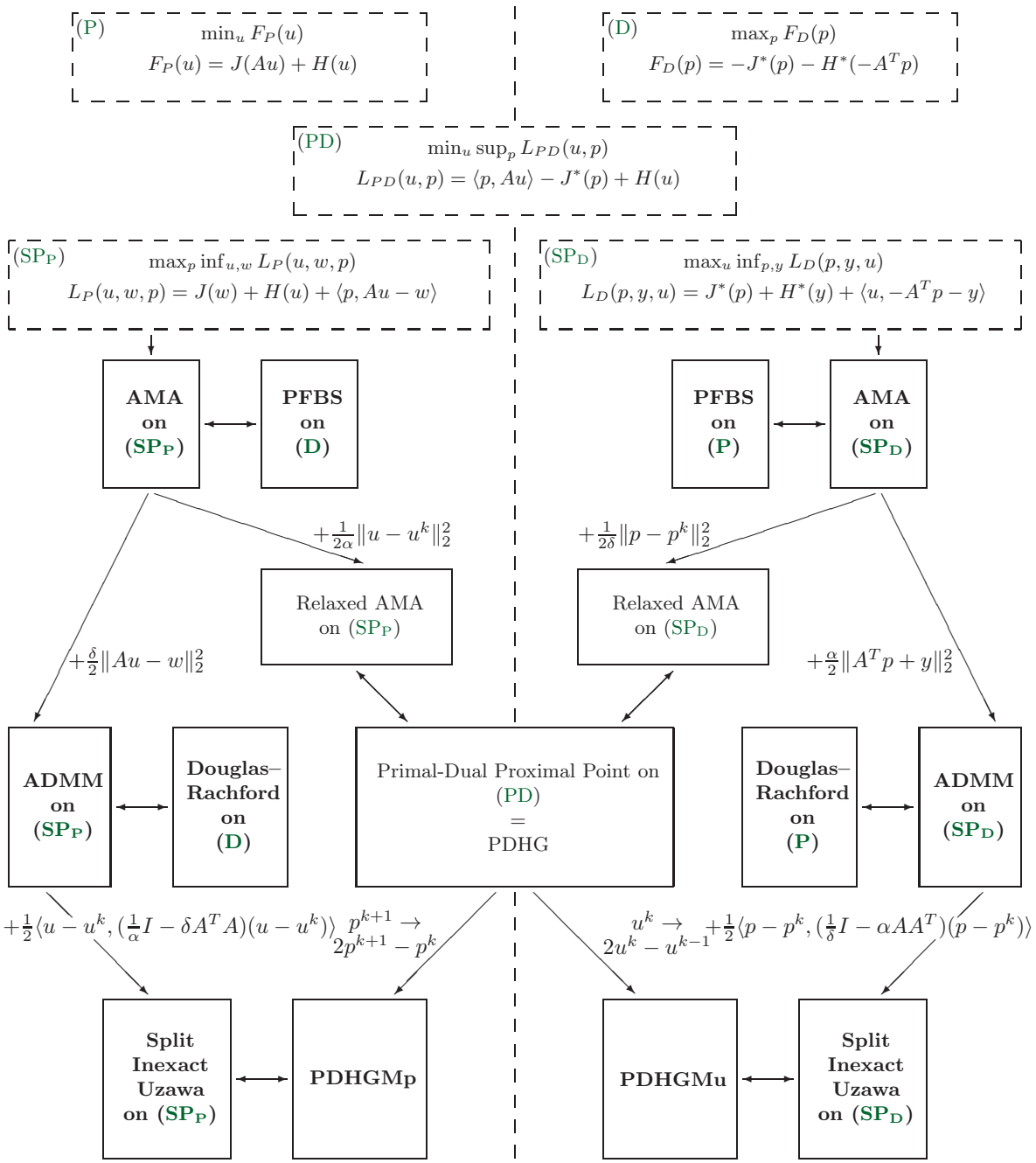
By the equivalence of PDHGMu and split inexact Uzawa on ($\mathrm{SP_D}$), Theorem 2.3 again applies to the PDHGMu iterates with y^k defined by (2.17). However, there is a stronger statement for the convergence of PDHGMu in [11, 12].

Theorem 2.4 (see [11]). *Let* $\alpha_k = \alpha > 0$, $\delta_k = \delta > 0$, *and* $0 < \delta < \frac{1}{\alpha\|A\|^2}$. *Let* (p^k, u^k) *satisfy* (2.18). *Then* (u^k, p^k) *converges to a saddle point of* L_{PD} (2.2).

Similarly, the corresponding split inexact Uzawa method applied to ($\mathrm{SP_P}$) is obtained by adding $\frac{1}{2}\langle u - u^k, (\frac{1}{\alpha_k}I - \delta_k A^T A)(u - u^k)\rangle$ to the u^{k+1} step of ADMM applied to ($\mathrm{SP_P}$). This leads to a similar modification of PDHG denoted as PDHGMp, where p^{k+1} is replaced by $\left((1 + \frac{\delta_{k+1}}{\delta_k})p^{k+1} - \frac{\delta_{k+1}}{\delta_k}p^k\right)$ in (2.10b).

The modifications to u^k and p^k in the split inexact Uzawa methods are reminiscent of the predictor-corrector step in Chen and Teboulle's predictor-corrector proximal method (PCPM) [14, 49]. Despite some close similarities, however, the algorithms are not equivalent. The modified PDHG algorithms are more implicit than PCPM.

The connections between the algorithms discussed so far are diagrammed in Figure 1. For simplicity, constant step sizes are assumed in the diagram. Double arrows indicate equivalences between algorithms, while single arrows show how to modify them to arrive at related methods.



Legend:
(P): Primal
(D): Dual
(PD): Primal-Dual
(SP_P): Split Primal
(SP_D): Split Dual

AMA: Alternating Minimization Algorithm (2.2.1)
PFBS: Proximal Forward Backward Splitting (2.2.1)
ADMM: Alternating Direction Method of Multipliers (2.2.2)
PDHG: Primal Dual Hybrid Gradient (2.2)
PDHGM: Modified PDHG (2.2.3)
Bold: Well Understood Convergence Properties

Figure 1. *PDHG-related algorithm framework.*

3. PDHG for TV deblurring. In this section we review from [58] the application of PDHG to the TV deblurring and denoising problems, but using the present notation. Both problems are of the form

$$\min_{u\in\mathbb{R}^m} \|u\|_{TV} + \frac{\lambda}{2}\|Ku - f\|_2^2, \tag{3.1}$$

where $\|\cdot\|_{TV}$ denotes the discrete TV seminorm to be defined. If K is a linear blurring operator, this corresponds to a TV regularized deblurring model. It also includes the TV denoising case when $K = I$ [45]. These applications are analyzed in [58], which also mentions possible extensions, such as to TV denoising with a constraint on the variance of u and also to l_1 minimization.

3.1. Total variation discretization. We define a discretization of the TV seminorm and in particular define a norm, $\|\cdot\|_E$, and a matrix, D, such that $\|u\|_{TV} = \|Du\|_E$. Thus (3.1) is of the same form as the primal problem (P) with $J(w) = \|w\|_E$, $A = D$, and $H(u) = \frac{\lambda}{2}\|Ku-f\|_2^2$. The details are included for completeness.

Define the discretized version of the TV seminorm by

$$\|u\|_{TV} = \sum_{p=1}^{M_r}\sum_{q=1}^{M_c}\sqrt{(D_1^+u_{p,q})^2 + (D_2^+u_{p,q})^2} \tag{3.2}$$

for $u \in \mathbb{R}^{M_r\times M_c}$. Here, D_k^+ represents a forward difference in the kth index, and we assume Neumann boundary conditions. It will be useful to instead work with vectorized $u \in \mathbb{R}^{M_rM_c}$ and to rewrite $\|u\|_{TV}$. The convention for vectorizing an $M_r \times M_c$ matrix will be to associate the (p, q) element of the matrix with the $(q-1)M_r + p$ element of the vector. Consider a graph $G(\mathcal{E}, \mathcal{V})$ defined by an $M_r \times M_c$ grid with $\mathcal{V} = \{1, \dots, M_rM_c\}$ the set of $m = M_rM_c$ nodes and $\mathcal{E}$ the set of $e = 2M_rM_c - M_r - M_c$ edges. Assume that the nodes are indexed so that the node corresponding to element (p, q) is indexed by $(q-1)M_r + p$. The edges, which will correspond to forward differences, can be indexed arbitrarily. Define $D \in \mathbb{R}^{e\times m}$ to be the edge-node adjacency matrix for this graph. So for a particular edge $\eta \in \mathcal{E}$ with endpoint indices $i, j \in \mathcal{V}$ and $i < j$, we have

$$D_{\eta,\nu} = \begin{cases} -1 & \text{for } \nu = i, \\ 1 & \text{for } \nu = j, \\ 0 & \text{for } \nu \neq i, j. \end{cases} \tag{3.3}$$

The matrix D is a discretization of the gradient, and $-D^T$ is the corresponding discretization of the divergence [22].

Also define $E \in \mathbb{R}^{e\times m}$ such that

$$E_{\eta,\nu} = \begin{cases} 1 & \text{if } D_{\eta,\nu} = -1, \\ 0 & \text{otherwise.} \end{cases} \tag{3.4}$$

The matrix E will be used to identify the edges used in each forward difference. Now define a norm on $\mathbb{R}^e$ by

$$\|w\|_E = \sum_{\nu=1}^{m}\left(\sqrt{E^T(w^2)}\right)_\nu. \tag{3.5}$$

Note that in this context, the square root and w^2 denote componentwise operations. Another way to interpret $\|w\|_E$ is as the sum of the l_2 norms of vectors w^ν, where

$$w^\nu = \begin{bmatrix} \vdots \\ w_e \\ \vdots \end{bmatrix} \quad \text{for } e \text{ such that } E_{e,\nu} = 1, \qquad \nu = 1, \dots, m. \tag{3.6}$$

Typically, which is to say away from the boundary, w^ν is of the form $w^\nu = \left[\begin{smallmatrix} w_{e_1^\nu} \\ w_{e_2^\nu} \end{smallmatrix}\right]$, where e_1^ν and e_2^ν are the edges used in the forward difference at node ν. So in terms of w^ν, $\|w\|_E = \sum_{\nu=1}^m \|w^\nu\|_2$, and we take $\|w^\nu\|_2 = 0$ in the case that w^ν is empty for some ν. The discrete TV seminorm defined above (3.2) can be written in terms of $\|\cdot\|_E$ as

$$\|u\|_{TV} = \|Du\|_E.$$

Use of the matrix E is nonstandard but also more general. For example, by redefining D and adding edge weights, this notation can be easily extended to other discretizations and even nonlocal TV.

By definition, the dual norm $\|\cdot\|_{E^*}$ to $\|\cdot\|_E$ is

$$\|x\|_{E^*} = \max_{\|y\|_E \le 1} \langle x, y \rangle. \tag{3.7}$$

This dual norm arises in the saddle point formulation of (3.1) on which the PDHG algorithm for TV deblurring is based. If x^ν is defined analogously to w^ν in (3.6), then the Cauchy–Schwarz inequality can be used to show that

$$\|x\|_{E^*} = \max_\nu \|x^\nu\|_2.$$

Altogether, $\|\cdot\|_E$ and $\|\cdot\|_{E^*}$ are analogous to $\|\cdot\|_1$ and $\|\cdot\|_\infty$, respectively, and can be expressed as

$$\|w\|_E = \left\|\sqrt{E^T(w^2)}\right\|_1 = \sum_{\nu=1}^m \|w^\nu\|_2 \qquad \text{and} \qquad \|x\|_{E^*} = \left\|\sqrt{E^T(x^2)}\right\|_\infty = \max_\nu \|x^\nu\|_2.$$

3.2. Saddle point formulations. The saddle point formulation for PDHG applied to TV minimization problems in [58] is based on the observation that

$$\|u\|_{TV} = \max_{p \in X} \langle p, Du \rangle, \tag{3.8}$$

where

$$X = \{p \in \mathbb{R}^e : \|p\|_{E^*} \le 1\}. \tag{3.9}$$

The set X, which is the unit ball in the dual norm of $\|\cdot\|_E$, can also be interpreted as a Cartesian product of unit balls in the l_2 norm. For example, in order for Du to be in X, the

discretized gradient $\begin{bmatrix} u_{p+1,q}-u_{p,q} \\ u_{p,q+1}-u_{p,q} \end{bmatrix}$ of u at each node (p,q) would have to have Euclidean norm less than or equal to 1. The dual norm interpretation is another way to explain (3.8) since

$$\max_{\{p:\|p\|_{E^*}\leq 1\}} \langle p, Du\rangle = \|Du\|_E,$$

which equals $\|u\|_{TV}$ by definition. Using duality to rewrite $\|u\|_{TV}$ is common to many primal-dual approaches for TV minimization including CGM [13], the second order cone programming formulation used in [28], and the semismooth Newton methods in [30, 31, 17]. Here, analogous to the definition of (PD), it can be used to reformulate problem (3.1) as the min-max problem

$$\min_{u\in\mathbb{R}^m} \max_{p\in X} \Phi(u,p) := \langle p, Du\rangle + \frac{\lambda}{2}\|Ku-f\|_2^2. \tag{3.10}$$

3.3. Existence of saddle point. One way to ensure that there exists a saddle point (u^*,p^*) of the convex-concave function Φ is to restrict u and p to be in bounded sets. Existence then follows from [44, Theorem 37.6]. The dual variable p is already required to lie in the convex set X. Assume that

$$\ker(D)\bigcap \ker(K) = \{0\}.$$

This is equivalent to assuming that $\ker(K)$ does not contain the vector of all ones, which is very reasonable for deblurring problems where K is an averaging operator. With this assumption, it follows that there exists $c\in\mathbb{R}$ such that the set

$$\left\{u : \|Du\|_E + \frac{\lambda}{2}\|Ku-f\|_2^2 \leq c\right\}$$

is nonempty and bounded. Thus we can restrict u to a bounded convex set.

3.4. Optimality conditions. If (u^*,p^*) is a saddle point of Φ, it follows that

$$\max_{p\in X} \langle p, Du^*\rangle + \frac{\lambda}{2}\|Ku^*-f\|_2^2 = \Phi(u^*,p^*) = \min_{u\in\mathbb{R}^m} \langle p^*, Du\rangle + \frac{\lambda}{2}\|Ku-f\|_2^2,$$

from which we can deduce the optimality conditions

$$D^Tp^* + \lambda K^T(Ku^*-f) = 0, \tag{3.11}$$

$$p^*E\sqrt{E^T(Du^*)^2} = Du^*, \tag{3.12}$$

$$p^* \in X. \tag{3.13}$$

The second optimality condition (3.12) with E defined by (3.4) can be understood as a discretization of $p^*|\nabla u^*| = \nabla u^*$.

3.5. PDHG for unconstrained TV deblurring. In [58] it is shown how to interpret the PDHG algorithm applied to (3.1) as a primal-dual proximal point method for solving (3.10) by iterating

$$p^{k+1} = \arg\max_{p\in X} \langle p, Du^k\rangle - \frac{1}{2\lambda\tau_k}\|p-p^k\|_2^2, \tag{3.14a}$$

$$u^{k+1} = \arg\min_{u\in\mathbb{R}^m} \langle p^{k+1}, Du\rangle + \frac{\lambda}{2}\|Ku-f\|_2^2 + \frac{\lambda(1-\theta_k)}{2\theta_k}\|u-u^k\|_2^2. \tag{3.14b}$$

The index k denotes the current iteration. Also, τ_k and θ_k are the dual and primal step sizes, respectively. The parameters in terms of δ_k and α_k from (2.10) are given by

$$\theta_k = \frac{\lambda\alpha_k}{1+\alpha_k\lambda}, \qquad \tau_k = \frac{\delta_k}{\lambda}.$$

The above max and min problems can be explicitly solved, yielding the following algorithm.

ALGORITHM. PDHG for TV deblurring.

$$p^{k+1} = \Pi_X\left(p^k + \tau_k\lambda Du^k\right), \tag{3.15a}$$

$$u^{k+1} = \left((1-\theta_k)I + \theta_k K^TK\right)^{-1}\left((1-\theta_k)u^k + \theta_k\left(K^Tf - \frac{1}{\lambda}D^Tp^{k+1}\right)\right). \tag{3.15b}$$

Here, Π_X is the orthogonal projection onto X defined by

$$\Pi_X(q) = \arg\min_{p\in X}\|p-q\|_2^2 = \frac{q}{E\max\left(\sqrt{E^T(q^2)},1\right)}, \tag{3.16}$$

where the division and max are understood in a componentwise sense. With q^ν defined analogously to w^ν in (3.6), we could alternatively write $(\Pi_X(q))_\eta = \frac{q_\eta}{\max(\|q^\nu\|_2,1)}$, where ν is the node at which edge η is used in a forward difference. For example, $\Pi_X(Du)$ can be thought of as a discretization of

$$\begin{cases}\frac{\nabla u}{|\nabla u|} & \text{if } |\nabla u|>1,\\ \nabla u & \text{otherwise.}\end{cases}$$

In the denoising case where $K=I$, the p^{k+1} update remains the same and the u^{k+1} simplifies to

$$u^{k+1} = (1-\theta_k)u^k + \theta_k\left(f - \frac{1}{\lambda}D^Tp^{k+1}\right).$$

4. Interpretation of PDHG as projected averaged gradient method for TV denoising. Even though we know of convergence results (Theorems 2.3 and 2.4) for the modified PDHG algorithms PDHGMu (2.18) and PDHGMp, it would be nice to show convergence of the original PDHG method (2.10) because PDHG still has some numerical advantages. Empirically, the stability requirements for the step size parameters are less restrictive for PDHG, so there is more freedom to tune the parameters to improve the rate of convergence. In this section, we restrict attention to PDHG applied to TV denoising and prove a convergence result assuming certain conditions on the parameters.

4.1. Projected gradient special case. Recall that in the case of TV denoising, problem (P) becomes

$$\min_{u\in\mathbb{R}^m}\|u\|_{TV} + \frac{\lambda}{2}\|u-f\|_2^2, \tag{4.1}$$

with $J = \|\cdot\|_E$, $A = D$, and $H(u) = \frac{\lambda}{2}\|u - f\|_2^2$, in which case PFBS on (D) simplifies to

$$p^{k+1} = \arg\min_{p\in\mathbb{R}^n} J^*(p) + \frac{1}{2\delta_k}\|p - (p^k + \delta_k D\nabla H^*(-D^T p^k))\|_2^2.$$

Since J^* is the indicator function for the unit ball, denoted as X (3.9), in the dual norm $\|\cdot\|_{E^*}$, this is exactly an orthogonal projection onto the convex set X (3.16). Letting $\tau_k = \frac{\delta_k}{\lambda}$ and also using that

$$H^*(-D^T p) = \frac{1}{2\lambda}\|\lambda f - D^T p\|_2^2 - \frac{\lambda}{2}\|f\|_2^2,$$

the algorithm simplifies to the following.

Algorithm. Gradient projection for TV denoising.

$$p^{k+1} = \Pi_X\left(p^k - \tau_k D(D^T p^k - \lambda f)\right). \tag{4.2}$$

Many variations of gradient projection applied to TV denoising are discussed in [59]. As already noted in [58], algorithm PDGH applied to TV denoising reduces to projected gradient descent when $\theta_k = 1$. Equivalence to (3.15) in the $\theta_k = 1$ case can be seen by plugging $u^k = (f - \frac{1}{\lambda}D^T p^k)$ into the update for p^{k+1}. This can be interpreted as projected gradient descent applied to

$$\min_{p\in X} G(p) := \frac{1}{2}\|D^T p - \lambda f\|_2^2, \tag{4.3}$$

an equivalent form of the dual problem.

Theorem 4.1. *Fix $p^0 \in \mathbb{R}^n$. Let p^k be defined by (4.2) with $0 < \inf \tau_k \le \sup \tau_k < \frac{1}{4}$, and define $u^{k+1} = f - \frac{D^T p^k}{\lambda}$. Then $\{p^k\}$ converges to a solution of (4.3), and $\{u^k\}$ converges to a solution of (4.1).*

Proof. Since ∇G is Lipschitz continuous with Lipschitz constant $\|DD^T\|$ and $u^{k+1} = \nabla H^*(-D^T p^k) = f - \frac{D^T p^k}{\lambda}$, then by Theorem 2.2 the result follows if $0 < \inf \tau_k \le \sup \tau_k < \frac{2}{\|DD^T\|}$. The bound $\|DD^T\| \le 8$ follows from the Gerschgorin circle theorem. ∎

4.1.1. AMA equivalence and soft thresholding interpretation. By the general equivalence between PFBS and AMA, (4.2) is equivalent to the following algorithm.

Algorithm. AMA for TV denoising.

$$u^{k+1} = f - \frac{D^T p^k}{\lambda}, \tag{4.4a}$$

$$w^{k+1} = \tilde{S}_{\frac{1}{\delta_k}}\left(Du^{k+1} + \frac{1}{\delta_k}p^k\right), \tag{4.4b}$$

$$p^{k+1} = p^k + \delta_k(Du^{k+1} - w^{k+1}). \tag{4.4c}$$

Here $\tilde{S}$ denotes the soft thresholding operator for $\|\cdot\|_E$ defined by

$$\tilde{S}_\alpha(f) = \arg\min_z \|z\|_E + \frac{1}{2\alpha}\|z - f\|_2^2.$$

This soft thresholding operator is closely related to the projection Π_X defined by (3.16). A direct application of Moreau's decomposition (Theorem 2.1) shows that $\tilde{S}_\alpha(f)$ can be defined by

$$\tilde{S}_\alpha(f) = f - \alpha\Pi_X\left(\frac{f}{\alpha}\right) = f - \Pi_{\alpha X}(f). \tag{4.5}$$

Similar projections can be derived for other norms.

In fact, it is not necessary to assume that J is a norm to obtain similar projection interpretations. It is enough that J be a convex 1-homogeneous function, as Chambolle points out in [10], when deriving a projection formula for the solution of the TV denoising problem. By letting $z = D^T p$, the dual problem (4.3) is solved by the projection

$$z = \Pi_{\{z:z=D^T p, \|p\|_{E^*}\le 1\}}(\lambda f),$$

and the solution to the TV denoising problem is given by

$$u^* = f - \frac{1}{\lambda}\Pi_{\{z:z=D^T p, \|p\|_{E^*}\le 1\}}(\lambda f).$$

However, the projection is nontrivial to compute.

4.2. Projected averaged gradient. In the $\theta \neq 1$ case, still for TV denoising, the projected gradient descent interpretation of PDHG extends to an interpretation as a projected averaged gradient descent algorithm. For the sake of simplicity, consider parameters τ and θ that are independent of k. Then plugging u^{k+1} into the update for p yields

$$p^{k+1} = \Pi_X\left(p^k - \tau d_\theta^k\right), \tag{4.6}$$

where

$$d_\theta^k = \theta\sum_{i=1}^{k}(1-\theta)^{k-i}\nabla G(p^i) + (1-\theta)^k\nabla G(p^0)$$

is a convex combination of gradients of G at the previous iterates p^i. Note that d_θ^k is not necessarily a descent direction.

This kind of averaging of previous iterates suggests a connection to Nesterov's method [36]. Several recent papers study variants of his method and their applications. Weiss, Aubert, and Blanc-Féraud in [52] apply a variant of Nesterov's method [37] to smoothed TV functionals. Beck and Teboulle in [1] and Becker, Bobin, and Candès in [3] also study variants of Nesterov's method that apply to l_1 and TV minimization problems. Tseng gives a unified treatment of accelerated proximal gradient methods like Nesterov's in [48]. However, despite some tantalizing similarities to PDHG, it appears that none is equivalent.

In the following section, the connection to a projected average gradient method on the dual is made for the more general case when the parameters are allowed to depend on k. Convergence results are presented for some special cases.

4.2.1. Convergence. For a minimizer $\overline{p}$, the optimality condition for the dual problem (4.3) is

$$\overline{p} = \Pi_X(\overline{p} - \tau\nabla G(\overline{p})) \quad \forall \tau \geq 0 \tag{4.7}$$

or, equivalently,

$$\langle \nabla G(\overline{p}), p - \overline{p}\rangle \geq 0 \quad \forall p \in X.$$

In the following, we denote $\overline{G} = \min_{p\in X} G(p)$ and let X^* denote the set of minimizers. As mentioned above, the PDHG algorithm (3.15) for TV denoising is related to a projected gradient method on the dual variable p. When τ and θ are allowed to depend on k, the algorithm can be written as

$$p^{k+1} = \Pi_X\left(p^k - \tau_k d^k\right), \tag{4.8}$$

where

$$d^k = \sum_{i=0}^{k} s_k^i \nabla G(p^i), \quad s_k^i = \theta_{i-1}\prod_{j=i}^{k-1}(1-\theta_j).$$

Note that

$$\sum_{i=0}^{k} s_k^i = 1, \quad s_k^i = (1-\theta_{k-1})s_{k-1}^i \qquad \forall k \geq 0,\ i \leq k, \qquad \text{and} \tag{4.9}$$

$$d^k = (1-\theta_{k-1})d^{k-1} + \theta_{k-1}\nabla G(p^k). \tag{4.10}$$

As above, the direction d^k is a linear (convex) combination of gradients of all previous iterates. We will show that d^k is an ϵ-gradient at p^k. This means that d^k is an element of the ϵ-differential (ϵ-subdifferential for nonsmooth functionals), $\partial_\epsilon G(p)$, of G at p^k defined by

$$G(q) \geq G(p^k) + \langle d^k, q - p^k\rangle - \epsilon \quad \forall q \in X.$$

When $\epsilon = 0$ this is the definition of d^k being a subgradient (in this case, the gradient) of G at p^k.

For p and q, the Bregman distance based on G between p and q is defined as

$$D(p,q) = G(p) - G(q) - \langle \nabla G(q), p - q\rangle \quad \forall p, q \in X. \tag{4.11}$$

From (4.3), the Bregman distance (4.11) reduces to

$$D(p,q) = \frac{1}{2}\|D^T(p-q)\|_2^2 \leq \frac{L}{2}\|p-q\|^2,$$

where L is the Lipschitz constant of ∇G.

Lemma 4.2. *For any $q \in X$, we have*

$$G(q) - G(p^k) - \langle d^k, q - p^k\rangle = \sum_{i=0}^{k} s_k^i (D(q,p^i) - D(p^k,p^i)).$$

Proof. For any $q \in X$,

$$\begin{aligned} G(q) - G(p^k) - \langle d^k, q - p^k \rangle &= G(q) - G(p^k) - \left\langle \sum_{i=0}^{k} s_k^i \nabla G(p^i), q - p^k \right\rangle \\ &= \sum_{i=0}^{k} s_k^i G(q) - \sum_{i=0}^{k} s_k^i G(p^i) - \sum_{i=0}^{k} s_k^i \langle \nabla G(p^i), q - p^i \rangle \\ &\quad + \sum_{i=0}^{k} s_k^i (G(p^i) - G(p^k) - \langle \nabla G(p^i), p^i - p^k \rangle) \\ &= \sum_{i=0}^{k} s_k^i (D(q, p^i) - D(p^k, p^i)). \end{aligned}$$

■

Lemma 4.3. *The direction d^k is an ϵ_k-gradient of p^k, where $\epsilon_k = \sum_{i=0}^{k} s_k^i D(p^k, p^i)$.*

Proof. By Lemma 4.2,

$$G(q) - G(p^k) - \langle d^k, q - p^k \rangle \geq - \sum_{i=0}^{k} s_k^i D(p^k, p^i) \quad \forall q \in X.$$

By the definition of ϵ-gradient, we obtain that d^k is an ϵ_k-gradient of G at p^k, where

$$\epsilon_k = \sum_{i=0}^{k} s_k^i D(p^k, p^i).$$

■

Lemma 4.4. *If $\theta_k \to 1$, then $\epsilon_k \to 0$.*

Proof. Let $h_k = G(p^k) - G(p^{k-1}) - \langle d^{k-1}, p^k - p^{k-1} \rangle$; then using the Lipschitz continuity of ∇G and the boundedness of d^k, we obtain

$$|h_k| = |D(p^k, p^{k-1}) + \langle (\nabla G(p^{k-1}) - d^{k-1}, p^k - p^{k-1}) \rangle| \leq \frac{L}{2} \|p^k - p^{k-1}\|_2^2 + C_1 \|p^k - p^{k-1}\|_2,$$

where L is the Lipschitz constant of ∇G, and C_1 is some positive constant. Since $\epsilon_k = \sum_{i=0}^{k} s_k^i D(p^k, p^i)$, p^k is bounded, and $\sum_{i=0} s_k^i = 1$, it follows that ϵ_k is bounded for any k.

Meanwhile, by replacing q with p^k and p^k with p^{k-1} in Lemma 4.2, we obtain $h_k = \sum_{i=0}^{k-1} s_{k-1}^i (D(p^k, p^i) - D(p^{k-1}, p^i))$. From

$$s_k^i = (1 - \theta_{k-1}) s_{k-1}^i \quad \forall 1 \leq i \leq k-1,$$

we get

$$\begin{aligned} \epsilon_k &= (1 - \theta_{k-1}) \sum_{i=0}^{k-1} s_{k-1}^i D(p^k, p^i) \\ &= (1 - \theta_{k-1}) \epsilon_{k-1} + (1 - \theta_{k-1}) \sum_{i=0}^{k-1} s_{k-1}^i (D(p^k, p^i) - D(p^{k-1}, p^i)) \\ &= (1 - \theta_{k-1})(\epsilon_{k-1} + h_k). \end{aligned}$$

By the boundedness of h_k and ϵ_k, we get immediately that if $\theta_{k-1} \to 1$, then $\epsilon_k \to 0$. ■

Since $\epsilon_k \to 0$, the convergence of p^k follows directly from classical [47, 33] ϵ-gradient methods. Possible choices of the step size τ_k are given in the following theorem.

Theorem 4.5 (see [47, 33]; convergence to the optimal set using divergent series τ_k). *Let $\theta_k \to 1$, and let τ_k satisfy $\tau_k > 0$, $\lim_{k\to\infty} \tau_k = 0$, and $\sum_{k=1}^{\infty} \tau_k = \infty$. Then the sequence p^k generated by (4.8) satisfies $G(p^k) \to \overline{G}$ and $dist\{p^k, X^*\} \to 0$.*

Since we require $\theta_k \to 1$, the algorithm is equivalent to projected gradient descent in the limit. However, it is well known that a divergent step size for τ_k is slow, and we can expect a better convergence rate without letting τ_k go to 0. In the following, we prove a different convergence result that does not require $\tau_k \to 0$, but still requires $\theta_k \to 1$.

Lemma 4.6. *For p^k defined by (4.8), we have $\langle d^k, p^{k+1} - p^k\rangle \le -\frac{1}{\tau_k}\|p^{k+1} - p^k\|_2^2$.*

Proof. Since p^{k+1} is the projection of $p^k - \tau_k d^k$ onto X, it follows that

$$\langle p^k - \tau_k d^k - p^{k+1}, p - p^{k+1}\rangle \le 0 \quad \forall p \in X.$$

Replacing p with p^k, we thus get

$$\langle d^k, p^{k+1} - p^k\rangle \le -\frac{1}{\tau_k}\|p^{k+1} - p^k\|_2^2. \tag{4.12}$$

■

Lemma 4.7. *Let p^k be generated by the method (4.8); then*

$$G(p^{k+1}) - G(p^k) - \frac{\beta_k^2}{\alpha_k}\|p^k - p^{k-1}\|_2^2 \le -\frac{(\alpha_k + \beta_k)^2}{\alpha_k}\left\|p^k - \left(\frac{\alpha_k}{\alpha_k + \beta_k}p^{k+1} + \frac{\beta_k}{\alpha_k + \beta_k}p^{k-1}\right)\right\|_2^2,$$

where

$$\alpha_k = \frac{1}{\tau_k\theta_{k-1}} - \frac{L}{2}, \quad \beta_k = \frac{1 - \theta_{k-1}}{2\theta_{k-1}\tau_{k-1}}. \tag{4.13}$$

Proof. By using the Taylor expansion and the Lipschitz continuity of ∇G (or directly from the fact that G is a quadratic function), we have

$$G(p^{k+1}) - G(p^k) \le \langle \nabla G(p^k), p^{k+1} - p^k\rangle + \frac{L}{2}\|p^{k+1} - p^k\|_2^2.$$

Since by (4.10), $\nabla G(p^k) = \frac{1}{\theta_{k-1}}(d^k - (1 - \theta_{k-1})d^{k-1})$, using (4.12) we have

$$\begin{aligned} G(p^{k+1}) - G(p^k) &\le \frac{1}{\theta_{k-1}}\langle d^k, p^{k+1} - p^k\rangle - \frac{1 - \theta_{k-1}}{\theta_{k-1}}\langle d^{k-1}, p^{k+1} - p^k\rangle + \frac{L}{2}\|p^{k+1} - p^k\|_2^2 \\ &= \left(\frac{L}{2} - \frac{1}{\tau_k\theta_{k-1}}\right)\|p^{k+1} - p^k\|_2^2 - \frac{1 - \theta_{k-1}}{\theta_{k-1}}\langle d^{k-1}, p^{k+1} - p^k\rangle. \end{aligned}$$

On the other hand, since p^k is the projection of $p^{k-1} - \tau_{k-1}d^{k-1}$, we get

$$\langle p^{k-1} - \tau_{k-1}d^{k-1} - p^k, p - p^k\rangle \le 0 \quad \forall p \in X.$$

Replacing p with p^{k+1}, we thus get

$$\langle d^{k-1}, p^{k+1} - p^k\rangle \ge \frac{1}{\tau_{k-1}}\langle p^{k-1} - p^k, p^{k+1} - p^k\rangle.$$

This yields

$$G(p^{k+1}) - G(p^k) \le -\alpha_k \|p^{k+1} - p^k\|^2 - 2\beta_k \langle p^{k-1} - p^k, p^{k+1} - p^k \rangle$$
$$= -\frac{(\alpha_k + \beta_k)^2}{\alpha_k} \left\| p^k - \left(\frac{\alpha_k}{\alpha_k + \beta_k} p^{k+1} + \frac{\beta_k}{\alpha_k + \beta_k} p^{k-1} \right) \right\|^2 + \frac{\beta_k^2}{\alpha_k} \|p^k - p^{k-1}\|^2,$$

where α_k and β_k are defined as in (4.13). ■

Theorem 4.8. *If α_k and β_k defined as in (4.13) are such that $\alpha_k > 0$, $\beta_k \ge 0$ and*

$$\sum_{k=0}^{\infty} \frac{(\alpha_k + \beta_k)^2}{\alpha_k} = \infty, \quad \sum_{k=0}^{\infty} \frac{\beta_k^2}{\alpha_k} < \infty, \quad \lim_{k \to \infty} \frac{\beta_k}{\alpha_k} = 0, \tag{4.14}$$

then every limit point pair (p^∞, d^∞) of a subsequence of (p^k, d^k) is such that p^∞ is a minimizer of (4.3) and $d^\infty = \nabla G(p^\infty)$.

Proof. The proof is adapted from [4, Propositions 2.3.1 and 2.3.2] and Lemma 4.7. Since p^k and d^k are bounded, the subsequence (p^k, d^k) has a convergent subsequence. Let (p^∞, d^∞) be a limit point of the pair (p^k, d^k), and let (p^{k_m}, d^{k_m}) be a subsequence that converges to (p^∞, d^∞). For $k_m > n_0$, Lemma 4.7 implies that

$$G(p^{k_m}) - G(p^{n_0}) \le -\sum_{k=n_0}^{k_m} \frac{(\alpha_k + \beta_k)^2}{\alpha_k} \left\| p^k - \left(\frac{\alpha_k}{\alpha_k + \beta_k} p^{k+1} + \frac{\beta_k}{\alpha_k + \beta_k} p^{k-1} \right) \right\|_2^2$$
$$+ \sum_{k=n_0}^{k_m} \frac{\beta_k^2}{\alpha_k} \|p^{k-1} - p^k\|_2^2.$$

By the boundedness of the constraint set X, the conditions (4.14) for α_k and β_k, and the fact that $G(p)$ is bounded from below, we conclude that

$$\left\| p^k - \left(\frac{\alpha_k}{\alpha_k + \beta_k} p^{k+1} + \frac{\beta_k}{\alpha_k + \beta_k} p^{k-1} \right) \right\|_2 \to 0.$$

Given $\epsilon > 0$, we can choose m large enough such that $\|p^{k_m} - p^\infty\|_2 \le \frac{\epsilon}{3}$, $\|p^k - (\frac{\alpha_k}{\alpha_k+\beta_k} p^{k+1} + \frac{\beta_k}{\alpha_k+\beta_k} p^{k-1})\|_2 \le \frac{\epsilon}{3}$ for all $k \ge k_m$, and $\frac{\beta_{k_m}}{\alpha_{k_m}+\beta_{k_m}} \|(p^{k_m-1} - p^\infty)\|_2 \le \frac{\epsilon}{3}$. This third requirement is possible because $\lim_{k\to\infty} \frac{\beta_k}{\alpha_k} = 0$. Then

$$\left\| (p^{k_m} - p^\infty) - \frac{\alpha_{k_m}}{\alpha_{k_m} + \beta_{k_m}} (p^{k_m+1} - p^\infty) - \frac{\beta_{k_m}}{\alpha_{k_m} + \beta_{k_m}} (p^{k_m-1} - p^\infty) \right\|_2 \le \frac{\epsilon}{3}$$

implies that

$$\left\| \frac{\alpha_{k_m}}{\alpha_{k_m} + \beta_{k_m}} (p^{k_m+1} - p^\infty) + \frac{\beta_{k_m}}{\alpha_{k_m} + \beta_{k_m}} (p^{k_m-1} - p^\infty) \right\|_2 \le \frac{2}{3}\epsilon.$$

Since $\frac{\beta_{k_m}}{\alpha_{k_m}+\beta_{k_m}} \|(p^{k_m-1} - p^\infty)\|_2 \le \frac{\epsilon}{3}$, we have

$$\|p^{k_m+1} - p^\infty\|_2 \le \frac{\alpha_{k_m} + \beta_{k_m}}{\alpha_{k_m}} \epsilon.$$

Note that k_m+1 is not necessarily an index for the subsequence $\{p^{k_m}\}$. Since $\lim_k \frac{\alpha_k+\beta_k}{\alpha_k}=1$, we have $\|p^{k_m+1}-p^\infty\|_2\to 0$ when $m\to\infty$. According to (4.8), the limit point (p^∞,d^∞) is therefore such that

$$p^\infty=\Pi_X(p^\infty-\tau d^\infty) \tag{4.15}$$

for $\tau>0$.

Now it remains to show that the corresponding subsequence $d^{k_m}=(1-\theta_{k_m-1})d^{k_m-1}+\theta_{k_m-1}\nabla G(p^{k_m})$ converges to $\nabla G(p^\infty)$. By the same technique, and the fact that $\theta_k\to 1$, we can get $\|\nabla G(p^{k_m})-d^\infty\|\le\epsilon$. Thus $\nabla G(p^{k_m})\to d^\infty$. On the other hand, $\nabla G(p^{k_m})\to\nabla G(p^\infty)$. Thus $d^\infty=\nabla G(p^\infty)$. Combining this with (4.15) and the optimal condition (4.7), we conclude that p^∞ is a minimizer. ■

In summary, the overall conditions on θ_k and τ_k are

- $\theta_k\to 1$, $\tau_k>0$,
- $0<\tau_k\theta_k<\frac{2}{L}$,
- $\sum_{k=0}^\infty \frac{(\alpha_k+\beta_k)^2}{\alpha_k}=\infty$,
- $\lim_{k\to\infty}\frac{\beta_k}{\alpha_k}=0$,
- $\sum_{k=0}^\infty \frac{\beta_k^2}{\alpha_k}<\infty$,

where

$$\alpha_k=\frac{1}{\tau_k\theta_{k-1}}-\frac{L}{2},\quad \beta_k=\frac{1-\theta_{k-1}}{2\theta_{k-1}\tau_{k-1}}. \tag{4.16}$$

Finally, we have $\theta_k\to 1$, and for τ_k the classical conditions for the projected gradient descent algorithm ($0<\tau_k<\frac{2}{L}$) and divergent step size ($\lim_k\tau_k\to 0$, $\sum_k\tau_k\to\infty$) are special cases of the above conditions. The algorithm converges empirically for a much wider range of parameters. For example, convergence with $0<\theta_k\le c<1$ and even $\theta_k\to 0$ is numerically demonstrated in [58], but a theoretical proof is still an open problem.

5. Extensions to constrained minimization.

The extension of PDHG to constrained minimization problems is discussed in [58] and applied, for example, to TV denoising with a constraint of the form $\|u-f\|^2\le m\sigma^2$ with σ^2 an estimate of the variance of the Gaussian noise. Such extensions work equally well with the modified PGHD algorithms. In the context of our general primal problem (P), if u is constrained to be in a convex set S, then this still fits in the framework of (P) since the indicator function for S can be incorporated into the definition of $H(u)$.

5.1. General convex constraint.

Consider the case when $H(u)$ is exactly the indicator function $g_S(u)$ for a convex set $S\subset\mathbb{R}^m$, which would mean

$$H(u)=g_S(u):=\begin{cases}0 & \text{if } u\in S,\\ \infty & \text{otherwise.}\end{cases}$$

Applying PDHG or the modified versions results in a primal step that can be interpreted as an orthogonal projection onto S. For example, when applying PDHGMu, the p^{k+1} step (2.18a)

remains the same, and the u^{k+1} step (2.18b) becomes

$$u^{k+1} = \Pi_S \left(u^k - \alpha_k A^T p^{k+1} \right).$$

For this algorithm to be practical, the projection Π_S must be straightforward to compute. Suppose the constraint on u is of the form $\|Ku - f\|_2 \leq \epsilon$ for some matrix K and $\epsilon > 0$. Then

$$\Pi_S(z) = (I - K^\dagger K)z + K^\dagger \begin{cases} Kz & \text{if } \|Kz - f\|_2 \leq \epsilon, \\ f + r\left(\frac{Kz - KK^\dagger f}{\|Kz - KK^\dagger f\|_2}\right) & \text{otherwise,} \end{cases}$$

where

$$r = \sqrt{\epsilon^2 - \|(I - KK^\dagger)f\|_2^2}$$

and $K^\dagger$ denotes the pseudoinverse of K. Note that $(I - K^\dagger K)$ represents the orthogonal projection onto $\ker(K)$. A special case where this projection is easily computed is when $KK^T = I$ and $K^\dagger = K^T$. In this case, the projection onto S simplifies to

$$\Pi_S(z) = (I - K^T K)z + K^T \begin{cases} Kz & \text{if } \|Kz - f\|_2 \leq \epsilon, \\ f + \epsilon\left(\frac{Kz - f}{\|Kz - f\|_2}\right) & \text{otherwise.} \end{cases}$$

5.2. Constrained TV deblurring. In the notation of problem (P), the unconstrained TV deblurring problem (3.1) corresponds to $J = \|\cdot\|_E$, $A = D$, and $H(u) = \frac{\lambda}{2}\|Ku - f\|_2^2$. A constrained version of this problem,

$$\min_{\|Ku - f\|_2 \leq \epsilon} \|u\|_{TV}, \tag{5.1}$$

can be rewritten as

$$\min_u \|Du\|_E + g_T(Ku),$$

where g_T is the indicator function for $T = \{z : \|z - f\|_2 \leq \epsilon\}$ defined by

$$g_T(z) = \begin{cases} 0 & \text{if } \|z - f\|_2 \leq \epsilon, \\ \infty & \text{otherwise.} \end{cases} \tag{5.2}$$

With the aim of eventually ending up with an explicit algorithm for this problem, we use some operator splitting ideas, letting

$$H(u) = 0 \qquad \text{and} \qquad J(Au) = J_1(Du) + J_2(Ku),$$

where $A = \left[\begin{smallmatrix} D \\ K \end{smallmatrix}\right]$, $J_1(w) = \|w\|_E$, and $J_2(z) = g_T(z)$. Letting $p = \left[\begin{smallmatrix} p_1 \\ p_2 \end{smallmatrix}\right]$, it follows that $J^*(p) = J_1^*(p_1) + J_2^*(p_2)$. Applying PDHG (2.10) with the u^{k+1} step written first, we obtain the following algorithm.

ALGORITHM. PDHG for constrained TV deblurring.

$$u^{k+1} = u^k - \alpha_k(D^T p_1^k + K^T p_2^k), \tag{5.3a}$$

$$p_1^{k+1} = \Pi_X\left(p_1^k + \delta_k D u^{k+1}\right), \tag{5.3b}$$

$$p_2^{k+1} = p_2^k + \delta_k K u^{k+1} - \delta_k \Pi_T\left(\frac{p_2^k}{\delta_k} + K u^{k+1}\right). \tag{5.3c}$$

Here, Π_T is defined by

$$\Pi_T(z) = f + \frac{z-f}{\max\left(\frac{\|z-f\|_2}{\epsilon}, 1\right)}. \tag{5.4}$$

In the constant step size case, to get the PDHGMp version of this algorithm, we would replace $D^T p_1^k + K^T p_2^k$ with $D^T(2p_1^k - p_1^{k-1}) + K^T(2p_2^k - p_2^{k-1})$.

5.3. Constrained l_1 minimization. Sparse approximation problems that seek to find a sparse solution satisfying some data constraints sometimes use the type of constraint described in the previous section [9]. A simple example of such a problem is

$$\min_u \|u\|_1 \qquad \text{such that} \qquad \|Ku - f\|_2 \le \epsilon, \tag{5.5}$$

where u is what we expect to be sparse, $K = R\Gamma\Psi^T$, R is a row selector, Γ is orthogonal, and Ψ is a tight frame with $\Psi^T\Psi = I$. $R\Gamma$ can be thought of as selecting some coefficients in an orthonormal basis. We will compare two different applications of PDHGMu, one that stays on the constraint set and one that does not.

Letting $J = \|\cdot\|_1$, $A = I$, $S = \{u : \|Ku - f\|_2 \le \epsilon\}$, and $H(u)$ equal the indicator function $g_S(u)$ for S, application of PDHGMu yields the following method in which u^k satisfies the constraint at each iteration.

ALGORITHM. PDHGMu for constrained l_1 minimization (stays in constraint set).

$$p^{k+1} = \Pi_{\{p:\|p\|_\infty \le 1\}}\left(p^k + \delta_k\left(\left(1 + \frac{\alpha_k}{\alpha_{k-1}}\right)u^k - \frac{\alpha_k}{\alpha_{k-1}}u^{k-1}\right)\right), \tag{5.6a}$$

$$u^{k+1} = \Pi_S\left(u^k - \alpha_k p^{k+1}\right). \tag{5.6b}$$

Here

$$\Pi_{\{p:\|p\|_\infty \le 1\}}(p) = \frac{p}{\max(|p|, 1)},$$

and

$$\Pi_S(u) = (I - K^T K)u + K^T\left(f + \frac{Ku - f}{\max\left(\frac{\|Ku-f\|_2}{\epsilon}, 1\right)}\right).$$

As before, Theorem 2.4 applies when $\alpha_k = \alpha > 0$, $\delta_k = \delta > 0$, and $\delta < \frac{1}{\alpha}$. Also, since $A = I$, the case when $\delta = \frac{1}{\alpha}$ is exactly ADMM applied to ($\mathrm{SP_D}$), which is equivalent to Douglas–Rachford splitting on (P).

In general, Π_S may be difficult to compute. It is possible to apply PDHGMu to (5.5) in a way that simplifies this projection but no longer stays in the constraint set at each iteration. The strategy is essentially to reverse the roles of J and H in the previous example, letting $J(u) = g_T(Ku)$ and $H(u) = \|u\|_1$ with g_T defined by (5.2). The following algorithm results.

ALGORITHM. PDHGMu for constrained l_1 minimization (does not stay in constraint set).

$$v^{k+1} = p^k + \delta_k K\left(\left(1 + \frac{\alpha_k}{\alpha_{k-1}}\right)u^k - \frac{\alpha_k}{\alpha_{k-1}}u^{k-1}\right), \tag{5.7a}$$

$$p^{k+1} = v^{k+1} - \delta_k \Pi_T\left(\frac{v^{k+1}}{\delta_k}\right), \tag{5.7b}$$

$$w^{k+1} = u^k - \alpha_k K^T p^{k+1}, \tag{5.7c}$$

$$u^{k+1} = w^{k+1} - \alpha_k \Pi_{\{p:\|p\|_\infty \le 1\}}\left(\frac{w^{k+1}}{\alpha_k}\right). \tag{5.7d}$$

Here, v^{k+1} and w^{k+1} are just place holders, and Π_T is defined by (5.4).

This variant of PDHGMu is still an application of the split inexact Uzawa method (2.16). Also, since $\|K\| \le 1$, the conditions for convergence are the same as for (5.6). Moreover, since $KK^T = I$, if $\delta = \frac{1}{\alpha}$, then this method can again be interpreted as ADMM applied to the split dual problem.

Note that Π_T is much simpler to compute than Π_S. The benefit of simplifying the projection step is important for problems where $K^\dagger$ is not practical to deal with numerically.

6. Numerical experiments.

We perform three numerical experiments to show that the modified and unmodified PDHG algorithms have similar performance and applications. The first is a comparison between PDHG, PDHGMu, and ADMM applied to TV denoising. The second compares the application of PDHG and PDHGMp to a constrained TV deblurring problem. The third experiment applies PDHGMu in two different ways to a constrained l_1 minimization problem.

6.1. PDHGM, PDHG, and ADMM for TV denoising.

Here, we closely follow the numerical example presented in Table 4 of [58], which compared PDHG to Chambolle's method [10] and CGM [13] for TV denoising. We use the same 256×256 cameraman image with intensities in $[0, 255]$. The image is corrupted with zero mean white Gaussian noise having standard deviation 20. We also use the same parameter $\lambda = .053$. Both adaptive and fixed step size strategies are compared. In all examples, we initialize $u^0 = f$ and $p^0 = 0$. Figure 2 shows the clean and noisy images along with a benchmark solution for the denoised image.

Recall that the PDHG algorithm for the TV denoising problem (4.1) is given by (3.15) with $K = I$. The adaptive strategy used for PDHG is the same one proposed in [58], where

$$\tau_k = .2 + .008k, \qquad \theta_k = \frac{.5 - \frac{5}{15+k}}{\tau_k}. \tag{6.1}$$

Figure 2. *Original, noisy, and benchmark denoised cameraman images.*

These can be related to the step sizes δ_k and α_k in (2.10) by

$$\delta_k = \lambda\tau_k, \qquad \alpha_k = \frac{\theta_k}{\lambda(1-\theta_k)}.$$

These time steps do not satisfy the requirements of Theorem 4.8, which requires $\theta_k \to 1$. However, we find that the adaptive PDHG strategy (6.1), for which $\theta_k \to 0$, is much better numerically for TV denoising.

When applying the PDHGMu algorithm to TV denoising, the stability requirement means that using the same adaptive time steps of (6.1) can be unstable. Instead, the adaptive strategy we use for PDHGMu is

$$\alpha_k = \frac{1}{\lambda(1+.5k)}, \qquad \delta_k = \frac{1}{8.01\alpha_k}. \tag{6.2}$$

Unfortunately, no adaptive strategy for PDHGMu can satisfy the requirements of Theorem 2.3, which assumes fixed time steps. However, the rate of convergence of the adaptive PDHGMu strategy for TV denoising is empirically better than the fixed parameter strategies.

We also perform some experiments with fixed α and δ. A comparison is made to gradient projection (4.2). We also compare to FISTA [1] applied to the dual of the TV denoising problem (4.3). As discussed in [2], where this application is referred to as FGP, it can be thought of as an acceleration of gradient projection. Much like the modification to PDHG, it replaces p^k in (4.2) with a combination of the previous iterates, namely,

$$p^k + \frac{t_k - 1}{t_{k+1}}(p^k - p^{k-1}),$$

where

$$t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}.$$

An additional comparison is made to ADMM as applied to ($\mathrm{SP_P}$). This algorithm alternates soft thresholding, solving a Poisson equation, and updating the Lagrange multiplier. This is equivalent to the split Bregman algorithm [29], which was compared to PDHG elsewhere in [58]. However, by working with the ADMM form of the algorithm, it is easier to use

Table 1
Iterations required for TV denoising.

Algorithm	tol $= 10^{-2}$	tol $= 10^{-4}$	tol $= 10^{-6}$
PDHG (adaptive)	14	70	310
PDHGMu (adaptive)	19	92	365
PDHG $\alpha = 5$, $\delta = .025$	31	404	8209
PDHG $\alpha = 1$, $\delta = .125$	51	173	1732
PDHG $\alpha = .2$, $\delta = .624$	167	383	899
PDHGMu $\alpha = 5$, $\delta = .025$	21	394	8041
PDHGMu $\alpha = 1$, $\delta = .125$	38	123	1768
PDHGMu $\alpha = .2$, $\delta = .624$	162	355	627
PDHG $\alpha = 5$, $\delta = .1$	22	108	2121
PDHG $\alpha = 1$, $\delta = .5$	39	123	430
PDHG $\alpha = .2$, $\delta = 2.5$	164	363	742
PDHGMu $\alpha = 5$, $\delta = .1$	unstable		
PDHGMu $\alpha = 1$, $\delta = .5$	unstable		
PDHGMu $\alpha = .2$, $\delta = 2.5$	unstable		
Proj. Grad. $\delta = .0132$	46	721	14996
FGP $\delta = .0066$	24	179	1264
ADMM $\delta = .025$	17	388	7951
ADMM $\delta = .125$	22	100	1804
ADMM $\delta = .624$	97	270	569

the duality gap as a stopping condition since u and p have the same interpretations in both algorithms. As in [58] we use the relative duality gap R for the stopping condition defined by

$$R(u,p) = \frac{F_P(u) - F_D(p)}{F_D(p)} = \frac{\left(\|u\|_{TV} + \frac{\lambda}{2}\|u-f\|_2^2\right) - \left(\frac{\lambda}{2}\|f\|_2^2 - \frac{1}{2\lambda}\|D^T p - \lambda f\|_2^2\right)}{\frac{\lambda}{2}\|f\|_2^2 - \frac{1}{2\lambda}\|D^T p - \lambda f\|_2^2},$$

which is the duality gap divided by the dual functional. The duality gap is defined to be the difference between the primal and dual functionals. This quantity is always nonnegative and is zero if and only if (u,p) is a saddle point of (3.10) with $K = I$. Table 1 shows the number of iterations required for the relative duality gap to fall below tolerances of 10^{-2}, 10^{-4}, and 10^{-6}. Note that the complexity of the PDHG and PDHGMu iterations scale like $O(m)$, whereas the ADMM iterations scale like $O(m \log m)$. Results for PDHGMp were identical to those for PDHGMu and are therefore not included in the table. All of the examples are for the same 256×256 cameraman image. As the problem size increases, more iterations would be required for all of the tabulated methods.

From Table 1, we see that PDHG and PDHGMu both benefit from adaptive step size schemes. The adaptive versions of these algorithms are compared in Figure 4(a), which plots the relative l_2 error to the benchmark solution versus the number of iterations. PDHG with the adaptive step sizes outperforms all of the other numerical experiments, but for identical fixed parameters, PDHGMu performs slightly better than PDHG. However, for fixed α the stability requirement, $\delta < \frac{1}{\alpha\|D\|^2}$, for PDHGMu places an upper bound on δ which is empirically

Figure 3. *Original image, blurry/noisy image, and image recovered from* 300 *PDHGMp iterations.*

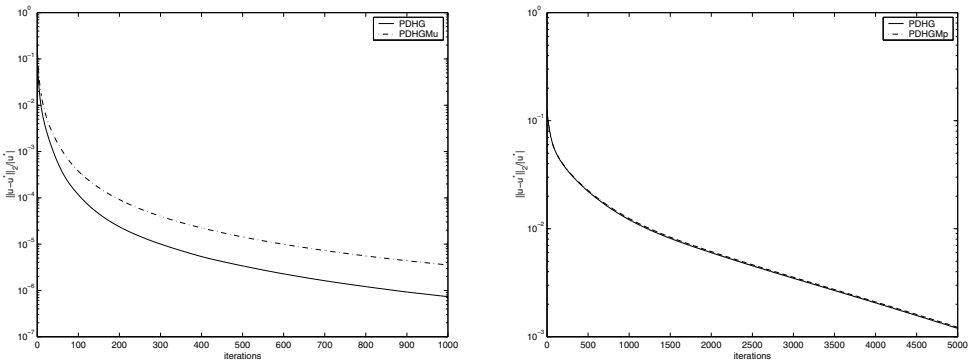
about four times less than for PDHG. Table 1 shows that for fixed α, PDHG with larger δ outperforms PDHGMu. The stability restriction for PDHGMu is also why the same adaptive time stepping scheme used for PDHG cannot be used for PDHGMu. We also note that fixed parameter versions of PDHG and PDHGMu are competitive with FGP.

Table 1 also demonstrates that larger α is more effective when the relative duality gap is large, and smaller α is better when this duality gap is small. Since PDHG for large α is similar to projected gradient descent, roughly speaking this means the adaptive PDHG algorithm starts out closer to PFBS on (D), but gradually becomes more like PFBS on (P).

All of the methods in Table 1 are at best linearly convergent, so superlinearly convergent methods like CGM and semismooth Newton will eventually outperform them when high accuracy is desired.

6.2. PDHGMp for constrained TV deblurring. PDHGMp and PDHG also perform similarly for constrained TV deblurring (5.1). For this example we use the same cameraman image from the previous section and let K be a convolution operator corresponding to a normalized Gaussian blur with a standard deviation of 3 in a 17×17 window. Letting h denote the clean image, the given data f is taken to be $f = Kh + \eta$, where η is zero mean Gaussian noise with standard deviation 1. We thus set $\epsilon = 256$. For the numerical experiments we used the fixed parameter versions of PDHG and PDHGMp with $\alpha = .33$ and $\delta = .33$. The images h and f and the recovered image from 300 iterations of PDHGMp are shown in Figure 3. Figure 4(b) compares the relative l_2 error to the benchmark solution as a function of the number of iterations for PDHG and PDHGMp. Empirically, with the same fixed parameters, the performance of these two algorithms is nearly identical, and the curves are indistinguishable in Figure 4(b). Although many iterations are required for a high accuracy solution, Figure 3 shows the result can be visually satisfactory after just a few hundred iterations.

6.3. PDHGMu for constrained l_1 minimization. Here we compare two applications of PDHGMu, (5.6) and (5.7), applied to (5.5) with $\epsilon = .01$. Let $K = R\Gamma\Psi^T$, where R is a row selector, Γ is an orthogonal two-dimensional (2D) discrete cosine transform (DCT), and Ψ is a redundant translation invariant 2D Haar wavelet transform normalized so that $\Psi^T\Psi = I$. It follows that $KK^T = I$ and $K^\dagger = K^T$. For a simple example, let h be a 32×32 image, shown in Figure 5, that is a linear combination of just four Haar wavelets. Let R select 64 of the lowest frequency DCT measurements and define $f = R\Gamma h$. The constrained l_1 minimization model aims to recover a sparse signal in the wavelet domain that is consistent with these



(a) Denoising (adaptive time steps). (b) Deblurring ($\alpha = .33$, $\delta = .33$).

Figure 4. *l_2 error versus iterations for PDHG and PDHGMp.*

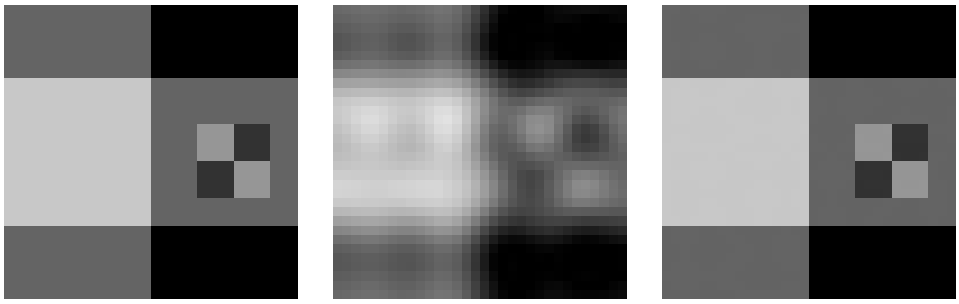


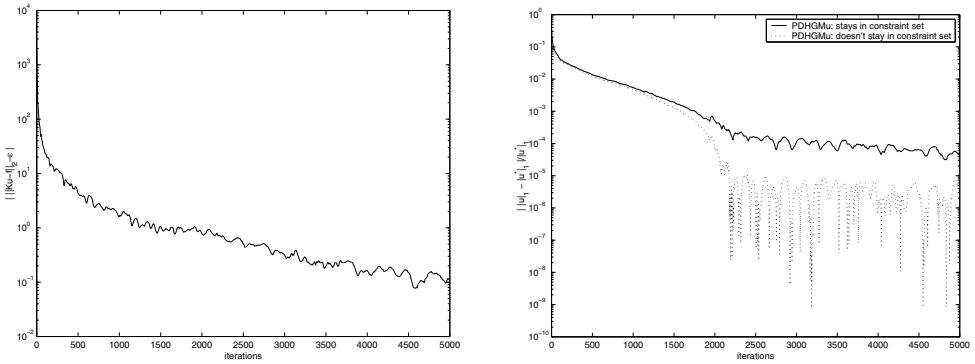
Figure 5. *Original, damaged, and benchmark recovered images.*

partial DCT measurements [8]. We have kept the example simple so as to focus on the two possible ways to handle the constraint using PDHGMu.

For the numerical experiments, we let $\alpha = .99$ and $\delta = .99$. We also scale $\|u\|_1$ by $\mu = 10$ to accelerate the rate of convergence. For the initialization, let $p^0 = 0$ and let $u^0 = \Psi z^0$, where $z^0 = \Gamma^T R^T R \Gamma h$ is the backprojection obtained by taking the inverse DCT of f with the missing measurements replaced by 0. Let u^* denote the benchmark solution. The recovered $z^* = \Psi^T u^*$ is nearly equal to h, but due to the nonuniqueness of minimizers, u^* has more nonzero wavelet coefficients than the originally selected four. Figure 5 shows h, z^0, and z^*.

Both versions of PDHGMu applied to this problem have simple iterations that scale like $O(m)$, but they behave somewhat differently. The first version (5.6) by definition satisfies the constraint at each iteration. However, these projections onto the constraint set destroy the sparsity of the approximate solution so it can be a little slower to recover a sparse solution. The other version (5.7), on the other hand, more quickly finds a sparse approximate solution but can take a long time to satisfy the constraint to a high precision.

To compare the two approaches, we compare plots of how the constraint and l_1 norm vary with iterations. Figure 6(a) plots $|\|Ku^k - f\|_2 - \epsilon|$ against the iterations k for (5.7). Note that this is always zero for (5.6), which stays on the constraint set. Figure 6(b) compares the differences $\frac{|\|u^k\|_1 - \|u^*\|_1|}{\|u^*\|_1}$ for both algorithms on a semilog plot. The empirical rate of



(a) Constraint versus iterations for the PDHGMu version (5.6) ($\alpha = .99$, $\delta = .99$).

(b) l_1 norm comparison ($\alpha = .99$, $\delta = .99$).

Figure 6. *Comparison of two applications of PDHGMu to constrained l_1 minimization.*

convergence to $\|u^*\|_1$ was similar for both algorithms despite the many oscillations. The second version of PDHGMu (5.7) was a little faster to recover a sparse solution, but (5.6) had the advantage of staying on the constraint set. For different applications with more complicated K, the simpler projection step in (5.7) would be an advantage of that approach.

Acknowledgments. We thank Paul Tseng for pointing out some key references and for helpful discussions about PDHG and the Moreau decomposition. Thanks also go to the anonymous referees, whose comments have significantly improved the quality of this paper.

REFERENCES

[1] A. Beck and M. Teboulle, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM J. Imaging Sci., 2 (2009), pp. 183–202.

[2] A. Beck and M. Teboulle, *Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems*, IEEE Trans. Image Process., 18 (2009), pp. 2419–2434.

[3] S. Becker, J. Bobin, and E. J. Candès, *NESTA: A Fast and Accurate First-Order Method for Sparse Recovery*, preprint, 2009; available online from http://www.acm.caltech.edu/~emmanuel/papers/NESTA.pdf.

[4] D. Bertsekas, *Nonlinear Programming*, 2nd ed., Athena Scientific, Belmont, MA, 1999.

[5] D. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Athena Scientific, Belmont, MA, 1996.

[6] D. Bertsekas and J. Tsitsiklis, *Parallel and Distributed Computation*, Prentice–Hall, Englewood Cliffs, NJ, 1989.

[7] S. Boyd and L. Vandenberghe, *Convex Analysis*, Cambridge University Press, Cambridge, UK, 2006.

[8] E. Candès and J. Romberg, *Signal recovery from random projections*, in SPIE International Symposium on Electronic Imaging: Computational Imaging III, Vol. 5674, San Jose, CA, 2005, pp. 76–86.

[9] E. Candès, J. Romberg, and T. Tao, *Stable signal recovery from incomplete and inaccurate measurements*, Comm. Pure Appl. Math., 59 (2006), pp. 1207–1223.

[10] A. Chambolle, *An algorithm for total variation minimization and applications*, J. Math. Imaging Vision, 20 (2004), pp. 89–97.

[11] A. Chambolle, V. Caselles, M. Novaga, D. Cremers, and T. Pock, *An Introduction to Total Variation for Image Analysis*, preprint, 2009; available online from http://hal.archives-ouvertes.fr/docs/00/43/75/81/PDF/preprint.pdf.

[12] A. Chambolle and T. Pock, *A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging*, preprint, 2010; available online from http://hal.archives-ouvertes.fr/docs/00/49/08/26/PDF/pd_alg_final.pdf.
[13] T. F. Chan, G. H. Golub, and P. Mulet, *A nonlinear primal-dual method for total variation-based image restoration*, SIAM J. Sci. Comput., 20 (1999), pp. 1964–1977.
[14] G. Chen and M. Teboulle, *A proximal-based decomposition method for convex minimization problems*, Math. Programming, 64 (1994), pp. 81–101.
[15] P. L. Combettes and V. R. Wajs, *Signal recovery by proximal forward-backward splitting*, Multiscale Model. Simul., 4 (2005), pp. 1168–1200.
[16] I. Daubechies, M. Defrise, and C. De Mol, *An iterative thresholding algorithm for linear inverse problems with a sparsity constraint*, Comm. Pure Appl. Math., 57 (2004), pp. 1413–1457.
[17] Y. Dong, M. Hintermüller, and M. Neri, *An efficient primal-dual method for L^1 TV image restoration*, SIAM J. Imaging Sci., 2 (2009), pp. 1168–1189.
[18] J. Douglas and H. H. Rachford, *On the numerical solution of heat conduction problems in two and three space variables*, Trans. Amer. Math. Soc., 82 (1956), pp. 421–439.
[19] J. Eckstein, *Splitting Methods for Monotone Operators with Applications to Parallel Optimization*, Ph.D. Thesis, Department of Civil Engineering, Massachusetts Institute of Technology, Cambridge, MA, 1989; available online from http://hdl.handle.net/1721.1/14356.
[20] J. Eckstein and D. Bertsekas, *On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators*, Math. Programming, 55 (1992), pp. 293–318.
[21] I. Ekeland and R. Témam, *Convex Analysis and Variational Problems*, Classics Appl. Math. 28, SIAM, Philadelphia, 1999.
[22] A. Elmoataz, O. Lezoray, and S. Bougleux, *Nonlocal discrete regularization on weighted graphs: A framework for image and manifold processing*, IEEE Trans. Image Process., 17 (2008), pp. 1047–1060.
[23] E. Esser, *Applications of Lagrangian-Based Alternating Direction Methods and Connections to Split Bregman*, CAM Report 09-31, UCLA, Los Angeles, CA, 2009.
[24] D. Gabay, *Methodes numeriques pour l'optimisation non-lineaire*, These de Doctorat d'Etat et Sciences Mathematiques, Universite Pierre et Marie Curie, Paris, 1979.
[25] D. Gabay and B. Mercier, *A dual algorithm for the solution of nonlinear variational problems via finite-element approximations*, Comput. Math. Appl., 2 (1976), pp. 17–40.
[26] R. Glowinski and P. Le Tallec, *Augmented Lagrangian and Operator-Splitting Methods in Nonlinear Mechanics*, SIAM, Philadelphia, 1989.
[27] R. Glowinski and A. Marrocco, *Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de Dirichlet non linéaires*, Rev. Française Automat. Informat. Recherche Opérationnelle, no. R-2 (1975), pp. 41–76.
[28] D. Goldfarb and W. Yin, *Second-order cone programming methods for total variation-based image restoration*, SIAM J. Sci. Comput., 27 (2005), pp. 622–645.
[29] T. Goldstein and S. Osher, *The split Bregman algorithm for L1-regularized problems*, SIAM J. Imaging Sci., 2 (2009), pp. 323–343.
[30] M. Hintermüller and K. Kunisch, *Total bounded variation regularization as bilaterally constrained optimization problem*, SIAM J. Appl. Math., 64 (2004), pp. 1311–1333.
[31] M. Hintermüller and G. Stadler, *An infeasible primal-dual algorithm for total bounded variation–based inf-convolution-type image restoration*, SIAM J. Sci. Comput., 28 (2006), pp. 1–23.
[32] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, UK, 1985.
[33] F. Larsson, M. Patriksson, and A.-B. Stromberg, *On the convergence of conditional ϵ-subgradient methods for convex programs and convex-concave saddle-point problems*, European J. Oper. Res., 151 (2003), pp. 461–473.
[34] P. L. Lions and B. Mercier, *Splitting algorithms for the sum of two nonlinear operators*, SIAM J. Numer. Anal., 16 (1979), pp. 964–979.
[35] J. J. Moreau, *Proximité et dualité dans un espace hilbertien*, Bull. Soc. Math. France, 93 (1965), pp. 273–299.
[36] Y. Nesterov, *Dual extrapolation and its applications to solving variational inequalities and related problems*, Math. Program. Ser. B, 109 (2007), pp. 319–344.
[37] Y. Nesterov, *Smooth minimization of non-smooth functions*, Math. Program. Ser. A, 103 (2005), pp. 127–152.

[38] J. Nocedal and S. Wright, *Numerical Optimization*, Springer-Verlag, New York, 1999.
[39] G. B. Passty, *Ergodic convergence to a zero of the sum of monotone operators in Hilbert space*, J. Math. Anal. Appl., 72 (1979), pp. 383–390.
[40] T. Pock, D. Cremers, H. Bischof, and A. Chambolle, *An algorithm for minimizing the Mumford-Shah functional*, in Proceedings of the 12th IEEE International Conference on Computer Vision (ICCV), 2009, pp. 1133–1140.
[41] L. Popov, *A modification of the Arrow-Hurwicz method for search of saddle points*, Math. Notes, 28 (1980), pp. 845–848.
[42] R. T. Rockafellar, *Augmented Lagrangians and applications of the proximal point algorithm in convex programming*, Math. Oper. Res., 1 (1976), pp. 97–116.
[43] R. T. Rockafellar, *Monotone operators and the proximal point algorithm*, SIAM J. Control Optim., 14 (1976), pp. 877–898.
[44] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.
[45] L. Rudin, S. Osher, and E. Fatemi, *Nonlinear total variation based noise removal algorithms*, Phys. D, 60 (1992), pp. 259–268.
[46] S. Setzer, *Split Bregman algorithm, Douglas-Rachford splitting and frame shrinkage*, in Scale Space and Variational Methods in Computer Vision, Lecture Notes in Comput. Sci. 5567, Springer-Verlag, Berlin, 2009, pp. 464–476.
[47] N. Z. Shor, K. C. Kiwiel, and A. Ruszcyǹski, *Minimization Methods for Nondifferentiable Functions*, Springer-Verlag, Berlin, 1985.
[48] P. Tseng, *On Accelerated Proximal Gradient Methods for Convex-Concave Optimization*, preprint, 2008.
[49] P. Tseng, *Alternating projection-proximal methods for convex programming and variational inequalities*, SIAM J. Optim., 7 (1997), pp. 951–965.
[50] P. Tseng, *Applications of a splitting algorithm to decomposition in convex programming and variational inequalities*, SIAM J. Control Optim., 29 (1991), pp. 119–138.
[51] Y. Wang, J. Yang, W. Yin, and Y. Zhang, *A new alternating minimization algorithm for total variation image reconstruction*, SIAM J. Imaging Sci., 1 (2008), pp. 248–272.
[52] P. Weiss, L. Blanc-Féraud, and G. Aubert, *Efficient schemes for total variation minimization under constraints in image processing*, SIAM J. Sci. Comput., 31 (2009), pp. 2047–2080.
[53] C. Wu and X.-C. Tai, *Augmented Lagrangian Method, Dual Methods, and Split Bregman Iteration for ROF, Vectorial TV, and High Order Models*, CAM Report 09-76, UCLA, Los Angeles, CA, 2009.
[54] W. Yin, *Analysis and Generalizations of the Linearized Bregman Method*, CAM Report 09-42, UCLA, Los Angeles, CA, 2009.
[55] W. Yin, S. Osher, D. Goldfarb, and J. Darbon, *Bregman iterative algorithms for ℓ_1-minimization with applications to compressed sensing*, SIAM J. Imaging Sci., 1 (2008), pp. 143–168.
[56] X. Zhang, M. Burger, X. Bresson, and S. Osher, *Bregmanized nonlocal regularization for deconvolution and sparse reconstruction*, SIAM J. Imaging Sci., 3 (2010), pp. 253–276.
[57] X. Zhang, M. Burger, and S. Osher, *A unified primal-dual algorithm framework based on Bregman iteration*, J. Sci. Comput., to appear; available online from http://www.springerlink.com/content/5549227304608363/.
[58] M. Zhu and T. F. Chan, *An Efficient Primal-Dual Hybrid Gradient Algorithm for Total Variation Image Restoration*, CAM Report 08-34, UCLA, Los Angeles, CA, 2008.
[59] M. Zhu, S. J. Wright, and T. F. Chan, *Duality-based algorithms for total-variation-regularized image restoration*, Comput. Optim. Appl., 47 (2010), pp. 377–400.