Proximal Splitting Algorithms: Overrelax them all!

Laurent Condat*†1, Daichi Kitahara[2], Andrés Contreras‡3, and Akira Hirabayashi[2]

[1]Visual Computing Center, King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia
[2]College of Information Sci. and Eng., Ritsumeikan University, Kusatsu, Japan
[3]University of Chile, Santiago, Chile

Draft, v0.2 December 16, 2019

1 Introduction

Many problems in statistics, machine learning, signal and image processing, or control can be formulated as convex optimization problems [1–7]. In the age of 'big data', with the explosion in size and complexity of the data to process, it is increasingly important to be able to solve convex optimization problems, whose solutions live in very high dimensional spaces [8–12]. There is an extensive literature about *splitting* methods for solving convex optimization problems, with applications in various fields [9, 13–20]. They consist of simple, easy to compute, steps that can deal with the terms in the objective function separately.

In this paper, we first present several splitting methods in the single umbrella of the forward–backward iteration to solve monotone inclusions, applied with preconditioning. In addition, we show that, when the smooth term in the objective function is quadratic, convergence is guaranteed with larger values of the relaxation parameter than previously known. By relaxation, we mean the following: let us consider an iterative algorithm of the form $z^{(i+1)} = T(z^{(i)})$, for some operator T, which converges to some fixed point and solution $z^\star$. Then $z^{(i+1)}$ tends to be closer than $z^{(i)}$ to $z^\star$, on average. So, we may want, starting at $z^{(i)}$, to move further in the direction $z^{(i+1)} - z^{(i)}$, which improves the estimate. This yields the *relaxed* iteration:

$$\left\lfloor \begin{array}{l} z^{(i+\frac{1}{2})} = T(z^{(i)}) \\ z^{(i+1)} = z^{(i)} + \rho^{(i)}(z^{(i+\frac{1}{2})} - z^{(i)}). \end{array} \right. \tag{1}$$

While there is no interest in doing underrelaxation with $\rho^{(i)}$ less than 1, overrelaxation with $\rho^{(i)}$ larger than 1 may be beneficial to the convergence speed. This is what is most often observed in practice. So, in this paper, we emphasize the possibility to overrelax most, if not all, proximal splitting algorithms.

2 The forward–backward algorithm

Let $\mathcal{X}$ be a real Hilbert space. Let $M : \mathcal{X} \to 2^{\mathcal{X}}$ be a set-valued maximally monotone operator. Let $C : \mathcal{X} \to \mathcal{X}$ be a ξ-cocoercive operator, for some real $\xi > 0$; that is, ξC is firmly nonexpansive, which

*Corresponding author. Contact: see his webpage https://lcondat.github.io

†The first author did a part of this work during a stay at Ritsumeikan University in 2018, hosted by the second and fourth authors, thanks to a fellowship from JSPS.

‡The third author contributed to this work during his visit of the first author at GIPSA-lab, Grenoble, France, in Spring 2019.

means that, for every $(z,z') \in \mathcal{Z}^2$, $\xi\|Cz - Cz'\|^2 \leq \langle z - z', Cz - Cz'\rangle$ [21, Proposition 4.4]. For notions of operator theory, we refer the reader to the textbooks [21, 22] or to the papers [17, 23]. Let us consider the monotone inclusion

$$0 \in Mz + Cz, \tag{2}$$

whose solution set is supposed nonempty. Let $z^{(0)} \in \mathcal{Z}$ be some initial estimate of a solution and let $\gamma > 0$ be some real parameter. The classical forward–backward iteration, to find a solution of (2), consists in iterating

$$z^{(i+1)} = J_{\gamma M}(z^{(i)} - \gamma C z^{(i)}), \tag{3}$$

where we denote by $J_N = (N + \mathrm{Id})^{-1}$ the resolvent of an operator N and Id is the identity. This method, proposed by Mercier [24], was further developed by many authors [25–31].

A not-so-well known extension of the forward–backward iteration consists in *relaxing* it. Let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence of relaxation parameters. The iteration becomes:

Forward–backward iteration for (2): for $i = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} z^{(i+\frac{1}{2})} = J_{\gamma M}(z^{(i)} - \gamma C z^{(i)}) \\ z^{(i+1)} = z^{(i)} + \rho^{(i)}(z^{(i+\frac{1}{2})} - z^{(i)}). \end{array}\right. \tag{4}$$

We can mention [32, Theorem 25.8] as the first convergence result, to our knowledge, about the relaxed forward–backward iteration, with a smaller relaxation range than in Theorem 2.1 below, though.

We can remark that the explicit mapping from $z^{(i)}$ to $z^{(i+\frac{1}{2})}$ in (4) can be equivalently written under the implicit form:

$$0 \in Mz^{(i+\frac{1}{2})} + Cz^{(i)} + \frac{1}{\gamma}(z^{(i+\frac{1}{2})} - z^{(i)}). \tag{5}$$

The now standard convergence result for the forward–backward iteration is the following:

Theorem 2.1 (forward–backward algorithm (4)**)** Let $z^{(0)} \in \mathcal{Z}$. Let $\gamma \in (0, 2\xi)$. Set $\delta = 2 - \gamma/(2\xi)$. Let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence in $[0,\delta]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(\delta - \rho^{(i)}) = +\infty$. Then the sequence $(z^{(i)})_{i\in\mathbb{N}}$ defined by the iteration (4) converges weakly to a solution of (2).

This result is [33, Lemma 4.4], see also [21, Theorem 26.14]. The proof relies on the Krasnosel'skiĭ–Mann theorem [21, Proposition 5.16 and Proposition 5.34], and the fact that the operator $T = (\gamma M + \mathrm{Id})^{-1} \circ (\mathrm{Id} - \gamma C)$, which maps $z^{(i)}$ to $z^{(i+\frac{1}{2})}$, is $(1/\delta)$-averaged. Moreover, the fixed points of T are the solutions to (2), as is clearly visible in (5).

We can allow $\gamma = 2\xi$ in Theorem 2.1, but in this case $\delta = 1$ and we cannot set $\rho^{(i)} = 1$. Since we are interested in over-relaxation, we write all theorems such that $\rho^{(i)} = 1$ can be used, if no relaxation is wanted.

We can note that in this algorithm, like in every algorithm of the paper, the sequence $(z^{(i+\frac{1}{2})})_{i\in\mathbb{N}}$ converges to the same solution as $(z^{(i)})_{i\in\mathbb{N}}$. For any i, $z^{(i+\frac{1}{2})}$ may be a better estimate of the solution than $z^{(i)}$, since it is in the domain of M; that is, $Mz^{(i+\frac{1}{2})} \neq \emptyset$.

We can also remark that if γ is close to 2ξ, δ is close to 1, so that overrelaxation cannot be used. This explains why the relaxed forward–backward iteration is not so well known.

Now, let P be a bounded, self-adjoint, strongly positive, linear operator on $\mathcal{Z}$. Clearly, solving (2) is equivalent to solving

$$0 \in P^{-1}Mz + P^{-1}Cz. \tag{6}$$

Let $\mathcal{Z}_P$ be the Hilbert space obtained by endowing $\mathcal{Z}$ with the inner product $(x, x') \mapsto \langle x, x'\rangle_P = \langle x, Px'\rangle$. Then $P^{-1}M$ is maximally monotone in $\mathcal{Z}_P$ [21, Proposition 20.24]. However, the cocoercivity of $P^{-1}C$ in $\mathcal{Z}_P$ has to be checked on a case-by-case basis. The *preconditioned* forward–backward iteration to solve (6) is

Preconditioned forward–backward iteration for (2): for $i = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} z^{(i+\frac{1}{2})} = (P^{-1}M + \mathrm{Id})^{-1}(z^{(i)} - P^{-1}Cz^{(i)}) \\ z^{(i+1)} = z^{(i)} + \rho^{(i)}(z^{(i+\frac{1}{2})} - z^{(i)}). \end{array} \right. \tag{7}$$

The corresponding convergence result follows:

Theorem 2.2 (preconditioned forward–backward algorithm (7)**)** Suppose that $P^{-1}C$ is χ-cocoercive in $\mathcal{Z}_P$, for some $\chi > \frac{1}{2}$. Set $\delta = 2 - 1/(2\chi)$. Let $z^{(0)} \in \mathcal{Z}$ and let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence in $[0, \delta]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(\delta - \rho^{(i)}) = +\infty$. Then the sequence $(z^{(i)})_{i\in\mathbb{N}}$ defined by the iteration (7) converges weakly to a solution of (2).

Proof This is a straightforward application of Theorem 2.1 in $\mathcal{Z}_P$ instead of $\mathcal{Z}$, with $\gamma = 1$. Weak convergence in $\mathcal{Z}_P$ is equivalent to weak convergence in $\mathcal{Z}$, and the solution sets of (2) and (6) are the same. □

We can remark that the explicit mapping from $z^{(i)}$ to $z^{(i+\frac{1}{2})}$ in (7) can be equivalently written under the implicit form:

$$0 \in Mz^{(i+\frac{1}{2})} + Cz^{(i)} + P(z^{(i+\frac{1}{2})} - z^{(i)}). \tag{8}$$

If $C = 0$, the forward–backward iteration reduces to the proximal point algorithm [21,34]. Let us give a formal convergence statement for the preconditioned proximal point algorithm. Let $M : \mathcal{Z} \to 2^{\mathcal{Z}}$ be a maximally monotone operator. The problem is to solve

$$0 \in Mz, \tag{9}$$

whose solution set is supposed nonempty. Let $z^{(0)} \in \mathcal{Z}$ and let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence of relaxation parameters. The relaxed and preconditioned proximal point algorithm consists in the iteration:

Proximal point iteration for (9): for $i = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} z^{(i+\frac{1}{2})} = (P^{-1}M + \mathrm{Id})^{-1}z^{(i)} \\ z^{(i+1)} = z^{(i)} + \rho^{(i)}(z^{(i+\frac{1}{2})} - z^{(i)}). \end{array} \right. \tag{10}$$

The mapping from $z^{(i)}$ to $z^{(i+\frac{1}{2})}$ in (10) can be equivalently written as

$$0 \in Mz^{(i+\frac{1}{2})} + P(z^{(i+\frac{1}{2})} - z^{(i)}). \tag{11}$$

The convergence of the proximal point algorithm can be stated as follows:

Theorem 2.3 (proximal point algorithm (10)**)** Let $z^{(0)} \in \mathcal{Z}$ and let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence in $[0, 2]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(2 - \rho^{(i)}) = +\infty$. Then the sequence $(z^{(i)})_{i\in\mathbb{N}}$ defined by the iteration (10) converges weakly to a solution of (9).

Proof $P^{-1}M$ is maximally monotone in $\mathcal{Z}_P$ [21, Proposition 20.24]. So, its resolvent $(P^{-1}M + \mathrm{Id})^{-1}$ is firmly nonexpansive in $\mathcal{Z}_P$ and by virtue of the Krasnosel'skiĭ–Mann theorem [21, Proposition 5.16 and Proposition 5.34], $(z^{(i)})_{i\in\mathbb{N}}$ converges weakly in $\mathcal{Z}_P$ to some element $z^\star \in \mathcal{Z}$ with $0 \in P^{-1}Mz^\star$, so that $0 \in Mz^\star$. □

2.1 The case where C is affine

Let us continue with our analysis of the forward–backward iteration to solve (2). In this section, we suppose that, in addition to being ξ-cocoercive, C is affine; that is, $C : z \in \mathcal{X} \mapsto Qz + c$, for some bounded, self-adjoint, positive, nonzero, linear operator Q on $\mathcal{X}$ and some element $c \in \mathcal{X}$. We have $\xi = 1/\|Q\|$, where the operator norm of Q is $\|Q\| = \sup\left\{\|Qz\| \;:\; z \in \mathcal{X}, \|z\| \le 1\right\}$. Now, we can write (5) as

$$0 \in (M + C)z^{(i+\frac{1}{2})} + P(z^{(i+\frac{1}{2})} - z^{(i)}), \tag{12}$$

where

$$P = \tfrac{1}{\gamma}\mathrm{Id} - Q. \tag{13}$$

So, the forward–backward iteration (4) can be interpreted as a preconditioned proximal point iteration (10), applied to find a zero of $M + C$! Since P must be strongly positive, we must have $\gamma \in (0, \xi)$, so that the admissible range for γ is halved. But in return, we get the larger range $(0, 2)$ for relaxation. Hence, we have the following new convergence result:

Theorem 2.4 (forward–backward algorithm (4)**, affine case)** Let $z^{(0)} \in \mathcal{X}$, let $\gamma \in (0, \xi)$ and let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence in $[0, 2]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(2 - \rho^{(i)}) = +\infty$. Then the sequence $(z^{(i)})_{i\in\mathbb{N}}$ defined by the iteration (4) converges weakly to a solution of (2).

Proof In view of (11) and (12), this is Theorem 2.3 applied to the problem (9) with $M + C$ as the maximally monotone operator. □

Now, let us assume that $\mathcal{X}$ is of finite dimension. Then we can improve the range $(0, \xi)$ for γ in Theorem 2.4 into $(0, \xi]$. For this, let A be a linear operator from $\mathcal{X}$ to some finite-dimensional real Hilbert space $\mathcal{U}$, such that $A^*A = Q$. For instance, $A = \sqrt{Q}$, the unique positive self-adjoint linear operator on $\mathcal{X}$ such that $\sqrt{Q}\sqrt{Q} = Q$. In many cases, $Q = A^*A$ for some A in the first place, for instance in least-squares problems, see below. We do not need to exhibit A, the fact that it exists, in finite dimension, is sufficient here. Then we can rewrite the problem (2) as $0 \in A^* \,\mathrm{Id}\, A + M'z$, where $M'\cdot = M \cdot +c$. To solve such an inclusion involving two monotone operators and one linear operator, the operator version [33, 35–37] of the Chambolle–Pock algorithm [38], which is detailed in Section 4, can be used. Starting at any pair of elements $z^{(0)} \in \mathcal{X}$ and $u^{(0)} \in \mathcal{U}$, it iterates:

$$\left\lfloor \begin{aligned} u^{(i+\frac{1}{2})} &= J_{\sigma(\mathrm{Id})^{-1}}(u^{(i)} + \sigma A z^{(i)}) \\ &= \tfrac{1}{1+\sigma}(u^{(i)} + \sigma A z^{(i)}) \\ z^{(i+\frac{1}{2})} &= J_{\tau M'}\big(z^{(i)} - \tau A^*(2u^{(i+\frac{1}{2})} - u^{(i)})\big) \\ &= J_{\tau M}\big(z^{(i)} - \tau A^*(2u^{(i+\frac{1}{2})} - u^{(i)}) - c\big) \\ z^{(i+1)} &= z^{(i)} + \rho^{(i)}(z^{(i+\frac{1}{2})} - z^{(i)}) \\ u^{(i+1)} &= u^{(i)} + \rho^{(i)}(u^{(i+\frac{1}{2})} - u^{(i)}). \end{aligned} \right. \tag{14}$$

Then $(z^{(i)})_{i\in\mathbb{N}}$ converges to a solution $z^\star$, with $0 \in Qz^\star + c + Mz^\star$, if the parameters $\tau > 0$ and σ are such that $\tau\sigma/\xi \le 1$ and $(\rho^{(i)})_{i\in\mathbb{N}}$ is a sequence in $[0, 2]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(2 - \rho^{(i)}) = +\infty$; see [33, Theorem 3.3] and [37] for two different proofs, where we emphasize the difficulty of allowing $\tau\sigma = \xi$.

So, set $\sigma = 1$; the condition for convergence becomes $\tau \le \xi$. Then the z-update becomes $z^{(i+\frac{1}{2})} = J_{\tau M}(z^{(i)} - \tau A^*Az^{(i)} - c)$ and we can discard the variable u, which is not used any more. Thus, we recover exactly, once again, the forward–backward iteration (4). It is remarkable that the forward–backward iteration can be viewed as a particular case of the Chambolle–Pock algorithm, which itself can be viewed as a Douglas–Rachford algorithm [37]. This is specific to C being affine, however.

Note that there is no reason for $\sigma = 1$ in the Chambolle–Pock algorithm to be the best value in practice; we may want to use the Chambolle-Pock algorithm with another value; but this requires the knowledge of A and A^*, and not only $Q = A^*A$. We can also note that linear convergence of the forward–backward can be proved if Q is strongly positive, which is not the case for the Chambolle–Pock algorithm.

Hence, we have the following convergence result, which extends Theorem 2.4 with the possibility of setting $\gamma = \xi$:

Theorem 2.5 (forward–backward algorithm (4)**, affine case)** Suppose that $\mathcal{Z}$ is of finite dimension. Let $z^{(0)} \in \mathcal{Z}$, let $\gamma \in (0, \xi]$ and let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence in $[0, 2]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(2 - \rho^{(i)}) = +\infty$. Then the sequence $(z^{(i)})_{i\in\mathbb{N}}$ defined by the iteration (4) converges to a solution of (2).

Thus, if $\gamma = \xi$, we can set $\rho^{(i)} = 1.49$ according to Theorem 2.1; with Theorem 2.4, we can do better and set $\rho^{(i)} = 1.99$.

So, which value of γ should be used in practice? If Q is badly conditioned and we do not use overrelaxation ($\rho^{(i)} = 1$), γ close to 2ξ is probably the best choice. If Q is well conditioned and we still do not use overrelaxation, $\gamma = \xi$ may be a better choice. Indeed, if $Q = (1/\xi)\mathrm{Id}$ and $\gamma = \xi$, then $z^{(1)} = (\gamma M + \mathrm{Id})^{-1}(z^{(0)} - \gamma C z^{(0)}) = (\gamma M + \mathrm{Id})^{-1}(0)$ and $z^{(1)}$ is a solution to (2), so that the algorithm converges in one iteration! In any case, a value of γ less than ξ is not interesting. So, the authors' recommendation for a given practical problem is to try the two settings: $\gamma = (2 - \epsilon)\xi$, for a small $\epsilon > 0$, and $\rho^{(i)} = 1$ on one hand, $\gamma = (1 - \epsilon)\xi$ and $\rho^{(i)} = 2 - \epsilon$ on the other hand.

2.2 Applications to convex optimization

Let $\mathcal{X}$ be a real Hilbert space. In the following, we denote by $\Gamma_0(\mathcal{X})$ the set of convex, proper, lower semicontinuous functions from $\mathcal{X}$ to $\mathbb{R} \cup \{+\infty\}$ [21]. For any function $f \in \Gamma_0(\mathcal{X})$, we define its proximity operator [39] as

$$\mathrm{prox}_f : \mathcal{X} \to \mathcal{X}, x \mapsto \operatorname*{arg\,min}_{x' \in \mathcal{X}} \Big(f(x') + \tfrac{1}{2}\|x - x'\|^2 \Big). \tag{15}$$

It is well known that $\mathrm{prox} = J_{\partial f} = (\partial f + \mathrm{Id})^{-1}$, where ∂f is the subdifferential of f [21]. There are fast and exact methods to compute the proximity operator of a large class of functions [13, 40, 41], see also the website `http://proximity-operator.net`.

Let $f \in \Gamma_0(\mathcal{X})$ and let $h : \mathcal{X} \to \mathbb{R}$ be a convex and Fréchet differentiable function with β-Lipschitz continuous gradient ∇h, for some real $\beta > 0$. We consider the convex optimization problem

$$\operatorname*{minimize}_{x \in \mathcal{X}} \; f(x) + h(x), \tag{16}$$

where the solution set is supposed nonempty. The well known Fermat's rule [21, Theorem 27.2] states that the problem (16) is equivalent to (2) with $M = \partial f$, which is maximally monotone, and $C = \nabla h$, which is ξ-cocoercive, with $\xi = 1/\beta$ [21, Corollary 18.17]. Hence, it is natural to use the forward–backward iteration to solve (16). Let $\gamma > 0$, let $x^{(0)} \in \mathcal{X}$ and let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence of relaxation parameters. The forward–backward iteration is:

Forward–backward iteration for (16): for $i = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} x^{(i+\frac{1}{2})} = \mathrm{prox}_{\gamma f}\big(x^{(i)} - \gamma \nabla h(x^{(i)})\big) \\ x^{(i+1)} = x^{(i)} + \rho^{(i)}\big(x^{(i+\frac{1}{2})} - x^{(i)}\big). \end{array} \right. \tag{17}$$

As a direct consequence of Theorem 2.1, we have:

Theorem 2.6 (forward–backward algorithm (17)**)** Let $x^{(0)} \in \mathcal{X}$ and let $\gamma \in (0, 2/\beta)$. Set $\delta = 2 - \gamma\beta/2$. Let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence in $[0, \delta]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(\delta - \rho^{(i)}) = +\infty$. Then the sequence $(x^{(i)})_{i\in\mathbb{N}}$ defined by the iteration (17) converges weakly to a solution of (16).

Now, let us focus on the case where h is quadratic:

$$h : x \mapsto \tfrac{1}{2}\langle x, Qx\rangle + \langle x, c\rangle, \tag{18}$$

for some bounded, self-adjoint, positive, nonzero, linear operator Q on $\mathcal{X}$ and some element $c \in \mathcal{X}$. A very common example is a least-squares penalty, in particular to solve inverse problems [5]; that is,

$$h : x \mapsto \tfrac{1}{2}\|Ax - y\|^2, \tag{19}$$

for some bounded linear operator A from $\mathcal{X}$ to a real Hilbert space $\mathcal{Y}$ and some element $y \in \mathcal{Y}$. Clearly, (19) is an instance of (18) with $Q = A^*A$, where A^* is the adjoint of A, and $c = A^*y$.

We have, for every $x \in \mathcal{X}$,

$$\nabla h(x) = Qx + c. \tag{20}$$

So, $\beta = \|Q\|$. Set $P = \frac{1}{\gamma}\mathrm{Id} - Q$. We can remark that the update in (17) can be written as:

$$x^{(i+\frac{1}{2})} = \operatorname*{arg\,min}_{x\in\mathcal{X}} f(x) + h(x) + \tfrac{1}{2}\|x - x^{(i)}\|_P^2, \tag{21}$$

where we introduce the norm $\|\cdot\|_P : x \mapsto \sqrt{\langle x, Px\rangle}$. So, $x^{(i+\frac{1}{2})}$ can be viewed as being obtained by applying the proximity operator of $f + h$ with the preconditioned norm $\|\cdot\|_P$. Hence, as a direct consequence of Theorem 2.5, we have:

Theorem 2.7 (forward–backward algorithm (17)**, quadratic case)** Suppose that $\mathcal{X}$ is of finite dimension. Let $x^{(0)} \in \mathcal{X}$, let $\gamma \in (0, 1/\beta]$ and let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence in $[0, 2]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(2 - \rho^{(i)}) = +\infty$. Then the sequence $(x^{(i)})_{i\in\mathbb{N}}$ defined by the iteration (17) converges to a solution of (16).

3 The Loris–Verhoeven algorithm

Let $\mathcal{X}$ and $\mathcal{U}$ be two real Hilbert spaces. Let $g \in \Gamma_0(\mathcal{U})$ and let $h : \mathcal{X} \to \mathbb{R}$ be a convex and Fréchet differentiable function with β-Lipschitz continuous gradient ∇h, for some real $\beta > 0$. Let $L : \mathcal{X} \to \mathcal{U}$ be a bounded linear operator.

Often, the template problem (16) of minimizing the sum of two functions is too simple and we would like, instead, to

$$\underset{x\in\mathcal{X}}{\text{minimize}}\; g(Lx) + h(x), \tag{22}$$

where the solution set is supposed nonempty. We assume that there is no simple way to compute the proximity operator of $g \circ L$. The problem (22) is equivalent to the monotone inclusion

$$0 \in L^*\partial g(Lx) + \nabla h(x). \tag{23}$$

To get rid of the annoying operator L, we introduce an auxiliary variable $u \in \partial g(Lx)$, which shall be called the *dual* variable, so that the problem now consists in finding $x \in \mathcal{X}$ and $u \in \mathcal{U}$ such that

$$\begin{cases} u \in \partial g(Lx) \\ 0 = L^*u + \nabla h(x) \end{cases}. \tag{24}$$

The interest in increasing the dimension of the problem is that we obtain a system of two monotone inclusions, which are decoupled: ∂g and ∇h appear separately in the two inclusions. So, equivalently, the problem is to find a pair of objects $z = (x, u)$ in $\mathcal{Z} = \mathcal{X} \times \mathcal{U}$ such that

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \underbrace{\begin{pmatrix} L^*u \\ -Lx + (\partial g)^{-1}(u) \end{pmatrix}}_{Mz} + \underbrace{\begin{pmatrix} \nabla h(x) \\ 0 \end{pmatrix}}_{Cz}. \tag{25}$$

The operator $M : \mathcal{Z} \to 2^{\mathcal{Z}}, (x,u) \mapsto (L^*u, -Lx + (\partial g)^{-1}u)$ is maximally monotone [21, Proposition 26.32 (iii)] and $C : \mathcal{Z} \to \mathcal{Z}, (x,u) \mapsto (\nabla h(x), 0)$ is ξ-cocoercive, with $\xi = 1/\beta$. Thus, it is natural to think of the forward–backward iteration to solve the problem (25). However, to make the resolvent of M computable with the proximity operator of g, we need preconditioning. The solution consists in the iteration, that we first write in implicit form:

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \underbrace{\begin{pmatrix} L^*u^{(i+\frac{1}{2})} \\ -Lx^{(i+\frac{1}{2})} + (\partial g)^{-1}u^{(i+\frac{1}{2})} \end{pmatrix}}_{Mz^{(i+\frac{1}{2})}} + \underbrace{\begin{pmatrix} \nabla h(x^{(i)}) \\ 0 \end{pmatrix}}_{Cz^{(i)}} + \underbrace{\begin{pmatrix} \frac{1}{\tau}\mathrm{Id} & 0 \\ 0 & \frac{1}{\sigma}\mathrm{Id} - \tau LL^* \end{pmatrix}}_{P} \underbrace{\begin{pmatrix} x^{(i+\frac{1}{2})} - x^{(i)} \\ u^{(i+\frac{1}{2})} - u^{(i)} \end{pmatrix}}_{z^{(i+\frac{1}{2})} - z^{(i)}}, \tag{26}$$

where $\tau > 0$ and $\sigma > 0$ are two real parameters, $z^{(i)} = (x^{(i)}, u^{(i)})$ and $z^{(i+\frac{1}{2})} = (x^{(i+\frac{1}{2})}, u^{(i+\frac{1}{2})})$. It is not straightforward to see that this yields an explicit iteration. The key is to remark that we have

$$x^{(i+\frac{1}{2})} = x^{(i)} - \tau\nabla h(x^{(i)}) - \tau L^* u^{(i+\frac{1}{2})}, \tag{27}$$

so that we can update the primal variable $x^{(i+\frac{1}{2})}$, once the dual variable $u^{(i+\frac{1}{2})}$ is available. So, the first step of the algorithm is to construct $u^{(i+\frac{1}{2})}$. It depends on $Lx^{(i+\frac{1}{2})}$, which is not yet available, but using (27), we can express it using $x^{(i)}$ and $LL^*u^{(i+\frac{1}{2})}$. This last term is canceled in the preconditioner P to make the update explicit. That is,

$$0 \in -Lx^{(i+\frac{1}{2})} + (\partial g)^{-1}u^{(i+\frac{1}{2})} + (\tfrac{1}{\sigma}\mathrm{Id} - \tau LL^*)(u^{(i+\frac{1}{2})} - u^{(i)}) \tag{28}$$

$$\Leftrightarrow 0 \in -Lx^{(i)} + \tau L\nabla h(x^{(i)}) + \tau LL^*u^{(i+\frac{1}{2})} + (\partial g)^{-1}u^{(i+\frac{1}{2})} + (\tfrac{1}{\sigma}\mathrm{Id} - \tau LL^*)(u^{(i+\frac{1}{2})} - u^{(i)}) \tag{29}$$

$$\Leftrightarrow 0 \in -Lx^{(i)} + \tau L\nabla h(x^{(i)}) + \tau LL^*u^{(i)} + (\partial g)^{-1}u^{(i+\frac{1}{2})} + \tfrac{1}{\sigma}(u^{(i+\frac{1}{2})} - u^{(i)}) \tag{30}$$

$$\Leftrightarrow \big(\sigma(\partial g)^{-1} + \mathrm{Id}\big)u^{(i+\frac{1}{2})} \ni \sigma Lx^{(i)} - \tau\sigma L\nabla h(x^{(i)}) - \tau\sigma LL^*u^{(i)} + u^{(i)} \tag{31}$$

$$\Leftrightarrow u^{(i+\frac{1}{2})} = \mathrm{prox}_{\sigma g^*}\Big(\sigma L\big(x^{(i)} - \tau\nabla h(x^{(i)})\big) + u^{(i)} - \tau\sigma LL^*u^{(i)}\Big), \tag{32}$$

where we recall that the conjugate function $g^* \in \Gamma_0(\mathcal{U}) : u \mapsto \sup_{u' \in \mathcal{U}} \langle u, u' \rangle - g(u')$ is such that $\partial g^* = (\partial g)^{-1}$ [21, Corollary 16.30]. We also recall the Moreau identity, which allows computing the proximity operator of g^* from the one of g, and conversely [13]:

$$\mathrm{prox}_{\sigma g^*}(u) = u - \sigma\, \mathrm{prox}_{g/\sigma}(u/\sigma). \tag{33}$$

Let us define the dual convex optimization problem associated to the primal problem (22):

$$\underset{u \in \mathcal{U}}{\mathrm{minimize}}\ g^*(u) + h^*(-L^*u). \tag{34}$$

If a pair $(x,u) \in \mathcal{X} \times \mathcal{U}$ is a solution to (25), then x is a solution to (22) and u is a solution to (34).

Thus, let $\tau > 0$ and $\sigma > 0$, let $x^{(0)} \in \mathcal{X}$ and $u^{(0)} \in \mathcal{U}$, and let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence of relaxation parameters. The primal–dual forward–backward iteration, which we call the Loris–Verhoeven iteration is:

Loris–Verhoeven iteration for (22) **and** (34): for $i = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} u^{(i+\frac{1}{2})} = \mathrm{prox}_{\sigma g^*}\Big(u^{(i)} + \sigma L\big(x^{(i)} - \tau\nabla h(x^{(i)}) - \tau L^*u^{(i)}\big)\Big) \\ x^{(i+1)} = x^{(i)} - \rho^{(i)}\tau\big(\nabla h(x^{(i)}) + L^*u^{(i+\frac{1}{2})}\big) \\ u^{(i+1)} = u^{(i)} + \rho^{(i)}(u^{(i+\frac{1}{2})} - u^{(i)}). \end{array}\right. \tag{35}$$

This algorithm was first proposed by Loris and Verhoeven, in the case where h is a least-squares term [42]. It was then rediscovered several times and named *Primal–Dual Fixed-Point algorithm based on the Proximity Operator* (PDFP2O) [43] or *Proximal Alternating Predictor–Corrector* (PAPC) algorithm [44]. The above interpretation of the algorithm as a primal–dual forward–backward iteration has been presented in [45].

We can note that the Loris–Verhoeven iteration can be rewritten by adding auxiliary variables $a^{(i)} = \nabla h(x^{(i)})$ and $b^{(i)} = L^* u^{(i)}$, for $i \in \mathbb{N}$, so that there is only one call to ∇h and L^* per iteration:

Loris–Verhoeven iteration for (22) **and** (34): for $i = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} a^{(i)} = \nabla h(x^{(i)}) \\ u^{(i+\frac{1}{2})} = \mathrm{prox}_{\sigma g^*}\Big(u^{(i)} + \sigma L\big(x^{(i)} - \tau(a^{(i)} + b^{(i)})\big)\Big) \\ b^{(i+\frac{1}{2})} = L^* u^{(i+\frac{1}{2})} \\ x^{(i+1)} = x^{(i)} - \rho^{(i)}\tau(a^{(i)} + b^{(i+\frac{1}{2})}) \\ u^{(i+1)} = u^{(i)} + \rho^{(i)}(u^{(i+\frac{1}{2})} - u^{(i)}) \\ b^{(i+1)} = b^{(i)} + \rho^{(i)}(b^{(i+\frac{1}{2})} - b^{(i)}). \end{array} \right. \tag{36}$$

As an application of Theorem 2.2, we obtain the following convergence result:

Theorem 3.1 (Loris–Verhoeven algorithm (35)**)** Let $x^{(0)} \in \mathcal{X}$ and $u^{(0)} \in \mathcal{U}$. Let $\tau \in (0, 2/\beta)$ and $\sigma > 0$ be such that $\sigma\tau\|L\|^2 < 1$. Set $\delta = 2 - \tau\beta/2$. Let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence in $[0, \delta]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(\delta - \rho^{(i)}) = +\infty$. Then the sequences $(x^{(i)})_{i\in\mathbb{N}}$ and $(u^{(i)})_{i\in\mathbb{N}}$ defined by the iteration (35) converge weakly to a solution of (22) and to a solution of (34), respectively.

Proof In view of (26) and (8), this is Theorem 2.2 applied to the problem (25). For this, P must be strongly positive, which is the case if and only if $\sigma\tau\|L\|^2 < 1$. Moreover, $P^{-1}C$ is $1/(\tau\beta)$-cocoercive in $\mathcal{Z}_P$. □

The following result makes it possible to have $\sigma\tau\|L\|^2 = 1$; it is a consequence of the analysis in [37] of the PD3O algorithm [46], of which the Loris–Verhoeven algorithm is a particular case. See also [43, Theorem 3.4 and Theorem 3.5] for the same result, but without relaxation.

Theorem 3.2 (Loris–Verhoeven algorithm (35)**)** Suppose that $\mathcal{X}$ and $\mathcal{U}$ are of finite dimension. Let $x^{(0)} \in \mathcal{X}$ and $u^{(0)} \in \mathcal{U}$. Let $\tau \in (0, 2/\beta)$ and $\sigma > 0$ be such that $\sigma\tau\|L\|^2 \leq 1$. Set $\delta = 2 - \tau\beta/2$. Let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence in $[0, \delta]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(\delta - \rho^{(i)}) = +\infty$. Then the sequences $(x^{(i)})_{i\in\mathbb{N}}$ and $(u^{(i)})_{i\in\mathbb{N}}$ defined by the iteration (35) converge to a solution of (22) and to a solution of (34), respectively.

According to Theorem 3.2, if $\mathcal{X} = \mathcal{U}$ is of finite dimension and $L = \mathrm{Id}$, we can set $\sigma = 1/\tau$ and the Loris–Verhoeven iteration becomes

$$\left\lfloor \begin{array}{l} u^{(i+\frac{1}{2})} = \mathrm{prox}_{g^*/\tau}\big(x^{(i)}/\tau - \nabla h(x^{(i)})\big) \\ x^{(i+\frac{1}{2})} = x^{(i)} - \tau\nabla h(x^{(i)}) - \tau u^{(i+\frac{1}{2})} \\ \phantom{x^{(i+\frac{1}{2})}} = \mathrm{prox}_{\tau g}\big(x^{(i)} - \tau\nabla h(x^{(i)})\big) \\ x^{(i+1)} = x^{(i)} + \rho^{(i)}(x^{(i+\frac{1}{2})} - x^{(i)}) \\ u^{(i+1)} = u^{(i)} + \rho^{(i)}(u^{(i+\frac{1}{2})} - u^{(i)}). \end{array} \right. \tag{37}$$

So, we can discard the dual variable and we recover the forward–backward iteration (17). It is interesting that in this primal algorithm, there is an implicit dual variable $u^{(i+1)} = -\nabla h(x^{(i)}) + (x^{(i)} - x^{(i+\frac{1}{2})})/\tau$, which converges to a solution of the dual problem; that is, to a minimizer of $h^*(-u) + g^*(u)$.

Again, let us focus on the case where h is quadratic; that is, $h : x \mapsto \frac{1}{2}\langle x, Qx\rangle + \langle x, c\rangle$, for some bounded, self-adjoint, positive, nonzero, linear operator Q on $\mathcal{X}$ and some element $c \in \mathcal{X}$. We have

$\beta = \|Q\|$. We can rewrite the primal–dual inclusion (26), which characterizes the Loris–Verhoeven iteration (35), as

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \underbrace{\begin{pmatrix} \nabla h(x^{(i+\frac{1}{2})}) + L^* u^{(i+\frac{1}{2})} \\ -Lx^{(i+\frac{1}{2})} + (\partial g)^{-1} u^{(i+\frac{1}{2})} \end{pmatrix}}_{Mz^{(i+\frac{1}{2})}} + \underbrace{\begin{pmatrix} \frac{1}{\tau}\mathrm{Id} - Q & 0 \\ 0 & \frac{1}{\sigma}\mathrm{Id} - \tau LL^* \end{pmatrix}}_{P} \underbrace{\begin{pmatrix} x^{(i+\frac{1}{2})} - x^{(i)} \\ u^{(i+\frac{1}{2})} - u^{(i)} \end{pmatrix}}_{z^{(i+\frac{1}{2})} - z^{(i)}}. \tag{38}$$

As an application of Theorem 2.4, we have:

Theorem 3.3 (Loris–Verhoeven algorithm (35)**, quadratic case)** Let $x^{(0)} \in \mathcal{X}$ and $u^{(0)} \in \mathcal{U}$. Let $\tau \in (0, 1/\beta)$ and $\sigma > 0$ be such that $\sigma\tau\|L\|^2 < 1$. Suppose that $(\rho^{(i)})_{i\in\mathbb{N}}$ is a sequence in $[0, 2]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(2 - \rho^{(i)}) = +\infty$. Then the sequences $(x^{(i)})_{i\in\mathbb{N}}$ and $(u^{(i)})_{i\in\mathbb{N}}$ defined by the iteration (35) converge weakly to a solution of (22) and to a solution of (34), respectively.

Furthermore, we can apply Theorem 2.5 to allow $\tau = 1/\beta$. Indeed, we want to solve $0 \in P^{-1}Mz + P^{-1}Cz$ in $\mathcal{Z}_P$, with P defined in (26). We have $P^{-1}C : (x, u) \mapsto (\tau Qx + \tau c, 0)$, which is affine and $1/(\tau\beta)$-cocoercive in $\mathcal{Z}_P$. Hence, as a direct application of Theorem 2.5, we have:

Theorem 3.4 (Loris–Verhoeven algorithm (35)**, quadratic case)** Suppose that $\mathcal{X}$ and $\mathcal{U}$ are of finite dimension. Let $x^{(0)} \in \mathcal{X}$ and $u^{(0)} \in \mathcal{U}$. Let $\tau \in (0, 1/\beta]$ and $\sigma > 0$ be such that $\sigma\tau\|L\|^2 < 1$. Suppose that $(\rho^{(i)})_{i\in\mathbb{N}}$ is a sequence in $[0, 2]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(2-\rho^{(i)}) = +\infty$. Then the sequences $(x^{(i)})_{i\in\mathbb{N}}$ and $(u^{(i)})_{i\in\mathbb{N}}$ defined by the iteration (35) converge to a solution of (22) and to a solution of (34), respectively.

4 The Chambolle–Pock algorithm

We now consider a similar problem to (22), but this time we want to make use of the proximity operators of the two functions. So, let $\mathcal{X}$ and $\mathcal{U}$ be two real Hilbert spaces. Let $f \in \Gamma_0(\mathcal{X})$ and $g \in \Gamma_0(\mathcal{U})$. Let $L : \mathcal{X} \to \mathcal{U}$ be a bounded linear operator. We want to

$$\underset{x\in\mathcal{X}}{\text{minimize}}\ f(x) + g(Lx), \tag{39}$$

where the solution set is supposed nonempty. Under mild conditions on f and g called qualification constraints [21, Theorem 27.2], (39) is equivalent to solving

$$0 \in \partial f(x) + L^*\partial g(Lx). \tag{40}$$

Again, to get rid of the annoying operator L, we introduce an auxiliary variable $u \in \partial g(Lx)$, which shall be called the dual variable, so that the problem now consists in finding $x \in \mathcal{X}$ and $u \in \mathcal{U}$ such that

$$\begin{cases} u \in \partial g(Lx) \\ 0 \in L^*u + \partial f(x) \end{cases}. \tag{41}$$

Let us define the dual convex optimization problem associated to the primal problem (39):

$$\underset{u\in\mathcal{U}}{\text{minimize}}\ f^*(-L^*u) + g^*(u). \tag{42}$$

If a pair $(x, u) \in \mathcal{X} \times \mathcal{U}$ is a solution to (41), then x is a solution to (39) and u is a solution to (42). Indeed, we have

$$\begin{cases} Lx \in (\partial g)^{-1}(u) \\ x \in (\partial f)^{-1}(-L^*u) \end{cases}, \tag{43}$$

so that u is solution to $0 \in (\partial g)^{-1}(u) - L(\partial f)^{-1}(-L^*u)$, which is exactly the first-order characterization of a solution to (42).

To solve the primal and dual problems (39) and (42) jointly, Chambolle and Pock proposed the following algorithms [38], see also [47, 48]:

Chambolle–Pock iteration, form I, for (39) **and** (42): for $i = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} x^{(i+\frac{1}{2})} = \mathrm{prox}_{\tau f}\big(x^{(i)} - \tau L^* u^{(i)}\big) \\ u^{(i+\frac{1}{2})} = \mathrm{prox}_{\sigma g^*}\big(u^{(i)} + \sigma L(2x^{(i+\frac{1}{2})} - x^{(i)})\big) \\ x^{(i+1)} = x^{(i)} + \rho^{(i)}(x^{(i+\frac{1}{2})} - x^{(i)}) \\ u^{(i+1)} = u^{(i)} + \rho^{(i)}(u^{(i+\frac{1}{2})} - u^{(i)}). \end{array} \right. \tag{44}$$

Chambolle–Pock iteration, form II, for (39) **and** (42): for $i = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} u^{(i+\frac{1}{2})} = \mathrm{prox}_{\sigma g^*}\big(u^{(i)} + \sigma L x^{(i)}\big) \\ x^{(i+\frac{1}{2})} = \mathrm{prox}_{\tau f}\big(x^{(i)} - \tau L^*(2u^{(i+\frac{1}{2})} - u^{(i)})\big) \\ u^{(i+1)} = u^{(i)} + \rho^{(i)}(u^{(i+\frac{1}{2})} - u^{(i)}) \\ x^{(i+1)} = x^{(i)} + \rho^{(i)}(x^{(i+\frac{1}{2})} - x^{(i)}). \end{array} \right. \tag{45}$$

We can note that the Chambolle–Pock algorithm is self-dual: if we apply the Chambolle–Pock iteration form I to the problem (42), to minimize $\tilde{f} + \tilde{g} \circ \tilde{L}$ with $\tilde{f} = g^*$, $\tilde{g} = f^*$, $\tilde{L} = -L^*$, and the roles of x and u are switched, as well as the roles of τ and σ, we obtain exactly the Chambolle–Pock iteration form II.

Chambolle and Pock proved the convergence in the finite-dimensional case, assuming that $\tau\sigma\|L\|^2 < 1$ and $\rho^{(i)} = 1$ [38]. The convergence was proved in a different way by He and Yuan, with a constant relaxation parameter $\rho^{(i)} = \rho \in (0, 2)$ and the same other hypotheses [49]; indeed, they observed that the Chambolle–Pock algorithm is a primal–dual proximal point algorithm to find a primal–dual pair $z = (x, u)$ in $\mathcal{Z} = \mathcal{X} \times \mathcal{U}$, solution to the monotone inclusion

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \underbrace{\begin{pmatrix} \partial f(x) + L^* u \\ -Lx + (\partial g)^{-1}(u) \end{pmatrix}}_{Mz}. \tag{46}$$

The operator $M : \mathcal{Z} \to 2^{\mathcal{Z}}, (x, u) \mapsto (\partial f(x) + L^* u, -Lx + (\partial g)^{-1} u)$ is maximally monotone [21, Proposition 26.32 (iii)]. Then, one can observe that the Chambolle–Pock iteration form I satisfies

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \underbrace{\begin{pmatrix} \partial f(x^{(i+\frac{1}{2})}) + L^* u^{(i+\frac{1}{2})} \\ -Lx^{(i+\frac{1}{2})} + (\partial g)^{-1} u^{(i+\frac{1}{2})} \end{pmatrix}}_{Mz^{(i+\frac{1}{2})}} + \underbrace{\begin{pmatrix} \frac{1}{\tau}\mathrm{Id} & -L^* \\ -L & \frac{1}{\sigma}\mathrm{Id} \end{pmatrix}}_{P} \underbrace{\begin{pmatrix} x^{(i+\frac{1}{2})} - x^{(i)} \\ u^{(i+\frac{1}{2})} - u^{(i)} \end{pmatrix}}_{z^{(i+\frac{1}{2})} - z^{(i)}}. \tag{47}$$

The Chambolle–Pock iteration form II satisfies the same primal–dual inclusion, but with $-L$ replaced by L in P.

Hence, the Chambolle–Pock iteration is a preconditioned primal–dual proximal point algorithm and, as a consequence of Theorem 2.3, we have [33, Theorem 3.2]:

Theorem 4.1 (Chambolle–Pock algorithm (44) **or** (45)**)** Let $x^{(0)} \in \mathcal{X}$ and $u^{(0)} \in \mathcal{U}$. Let $\tau > 0$ and $\sigma > 0$ be such that $\sigma\tau\|L\|^2 < 1$. Let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence in $[0, 2]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(2 - \rho^{(i)}) = +\infty$. Then the sequences $(x^{(i)})_{i\in\mathbb{N}}$ and $(u^{(i)})_{i\in\mathbb{N}}$ defined either by the iteration (44) or by the iteration (45) converge weakly to a solution of (39) and to a solution of (42), respectively.

In addition, the first author proved that in the finite dimensional setting, one can set $\sigma\tau\|L\|^2 = 1$ [33, Theorem 3.3], see also [37] for another proof. Thus, we have:

Theorem 4.2 (Chambolle–Pock algorithm (44) **or** (45)**)** Suppose that $\mathcal{X}$ and $\mathcal{U}$ are of finite dimension. Let $x^{(0)} \in \mathcal{X}$ and $u^{(0)} \in \mathcal{U}$. Let $\tau > 0$ and $\sigma > 0$ be such that $\sigma\tau\|L\|^2 \leq 1$. Let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence in $[0, 2]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(2 - \rho^{(i)}) = +\infty$. Then the sequences $(x^{(i)})_{i\in\mathbb{N}}$ and $(u^{(i)})_{i\in\mathbb{N}}$ defined either by the iteration (44) or by the iteration (45) converge to a solution of (39) and to a solution of (42), respectively.

The difference between Theorems 4.1 and 4.2 is that $\sigma\tau\|L\|^2 = 1$ is allowed in the latter. This is a significant improvement: in practice, one can set $\sigma = 1/(\tau\|L\|^2)$ in the algorithms and have only one parameter left, namely τ, to tune. Moreover, if $\mathcal{X} = \mathcal{U}$ and $L = \mathrm{Id}$, the problem is

$$\underset{x\in\mathcal{X}}{\text{minimize}}\ f(x) + g(x), \tag{48}$$

and by setting $\sigma = 1/\tau$ in the Chambolle–Pock algorithm, the Douglas–Rachford method [25, 30, 50, 51] is recovered as a special case:

Douglas–Rachford iteration for (48): for $i = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} x^{(i+\frac{1}{2})} = \mathrm{prox}_{\tau f}\big(x^{(i)} - \tau u^{(i)}\big) \\ u^{(i+\frac{1}{2})} = \mathrm{prox}_{\sigma g^*}\big(u^{(i)} + (2x^{(i+\frac{1}{2})} - x^{(i)})/\tau\big) \\ x^{(i+1)} = x^{(i)} + \rho^{(i)}(x^{(i+\frac{1}{2})} - x^{(i)}) \\ u^{(i+1)} = u^{(i)} + \rho^{(i)}(u^{(i+\frac{1}{2})} - u^{(i)}). \end{array} \right. \tag{49}$$

Note that the Douglas–Rachford algorithm is not a primal–dual proximal point algorithm, since the operator P above is not strongly positive any more. So, its weak convergence is difficult to establish; it was shown only in 2011 [51]. However, if $\mathcal{X}$ is of finite dimension, convergence is easy to show. For this let us define the auxiliary variable

$$s^{(i)} = x^{(i)} - \tau u^{(i)}. \tag{50}$$

The Douglas–Rachford iteration only depends on this concatenated variable, not on the full pair $(x^{(i)}, u^{(i)})$, and we can rewrite it as:

Douglas–Rachford iteration for (48): for $i = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} x^{(i+\frac{1}{2})} = \mathrm{prox}_{\tau f}(s^{(i)}) \\ u^{(i+\frac{1}{2})} = \mathrm{prox}_{g^*/\tau}\big((2x^{(i+\frac{1}{2})} - s^{(i)})/\tau\big) \\ s^{(i+1)} = s^{(i)} + \rho^{(i)}(x^{(i+\frac{1}{2})} - \tau u^{(i+\frac{1}{2})} - s^{(i)}), \end{array} \right. \tag{51}$$

or, equivalently,

Douglas–Rachford iteration for (48): for $i = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} x^{(i+\frac{1}{2})} = \mathrm{prox}_{\tau f}(s^{(i)}) \\ s^{(i+1)} = s^{(i)} + \rho^{(i)}\big(\mathrm{prox}_{\tau g}(2x^{(i+\frac{1}{2})} - s^{(i)}) - x^{(i+\frac{1}{2})}\big). \end{array} \right. \tag{52}$$

One can show that

$$s^{(i+1)} = \Big(\tfrac{\rho^{(i)}}{2}(2\mathrm{prox}_{\tau g} - \mathrm{Id}) \circ (2\mathrm{prox}_{\tau f} - \mathrm{Id}) + \tfrac{2-\rho^{(i)}}{2}\mathrm{Id}\Big)s^{(i)}. \tag{53}$$

Since this operator is averaged, we get weak convergence of $(s^{(i)})_{i\in\mathbb{N}}$ to some element. But in infinite dimension, one cannot invoke continuity arguments, so one cannot deduce that $(x^{(i+\frac{1}{2})})_{i\in\mathbb{N}}$ converges. In a finite-dimensional setting, however, the proximity operator is continuous, so that one can indeed conclude that $(x^{(i+\frac{1}{2})})_{i\in\mathbb{N}}$ converges to some minimizer of $f + g$, for every $\tau > 0$ and sequence $(\rho^{(i)})_{i\in\mathbb{N}}$ in $[0, 2]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(2 - \rho^{(i)}) = +\infty$.

The Douglas–Rachford algorithm is equivalent to the Alternating Direction Method of Multipliers (ADMM). The ADMM goes back to Glowinski and Marocco [52], and Gabay and Mercier [53]. This algorithm has been studied extensively, see e.g. [50, 54–60]. Its equivalence with the Douglas–Rachford algorithm is well known [27, 50, 61]. The ADMM was rediscovered by Osher et al. and called *Split Bregman Algorithm* [62, 63]; this method has received significant attention in image processing [55, 64–67]. The equivalence between the ADMM and the Split Bregman Algorithm is now well established [47, 63, 68]. Meanwhile, the ADMM has been popularized in image processing by a series of papers of Figueiredo, Bioucas-Dias et al. , e.g. [69–71].

Let us consider a slightly modified version of the Douglas–Rachford algorithm, which we will use in the next section. Let $\mathcal{X}$ be a real Hilbert space and let $f \in \Gamma_0(\mathcal{X})$, $g \in \Gamma_0(\mathcal{X})$, $c \in \mathcal{X}$. We consider the problem:

$$\underset{x\in\mathcal{X}}{\text{minimize}}\ f(x) + g(x) + \langle x, c\rangle. \tag{54}$$

The dual problem is

$$\underset{u\in\mathcal{X}}{\text{minimize}}\ f^*(-u-c) + g^*(u). \tag{55}$$

We again suppose that f and g satisfy some qualification constraints [21, Corollary 27.6], so that the solution set of $0 \in \partial f(x) + \partial g(x) + c$ is nonempty. Adding the linear term $\langle x, c\rangle$ does not add any difficulty: one can apply the Douglas–Rachford algorithm above to minimize $\tilde{f} + g$, where $\tilde{f} = f + \langle \cdot, c\rangle$, using the fact that $\text{prox}_{\tau\tilde{f}}(x) = \text{prox}_{\tau f}(x - \tau c)$. So, the Douglas–Rachford algorithm is

Douglas–Rachford iteration for (54): for $i = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} x^{(i+\frac{1}{2})} = \text{prox}_{\tau f}(s^{(i)} - \tau c) \\ s^{(i+1)} = s^{(i)} + \rho^{(i)}\big(\text{prox}_{\tau g}(2x^{(i+\frac{1}{2})} - s^{(i)}) - x^{(i+\frac{1}{2})}\big). \end{array}\right. \tag{56}$$

Using the Moreau identity (33) and starting from the form (49), we can write the algorithm as

Douglas–Rachford iteration for (54): for $i = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} v^{(i+\frac{1}{2})} = \text{prox}_{f^*/\tau}\big(\frac{1}{\tau}x^{(i)} - u^{(i)} - c\big) \\ x^{(i+\frac{1}{2})} = x^{(i)} - \tau(u^{(i)} + v^{(i+\frac{1}{2})} + c) \\ u^{(i+\frac{1}{2})} = \text{prox}_{g^*/\tau}\big(\frac{1}{\tau}x^{(i+\frac{1}{2})} - v^{(i+\frac{1}{2})} - c\big) \\ x^{(i+1)} = x^{(i)} + \rho^{(i)}(x^{(i+\frac{1}{2})} - x^{(i)}) \\ u^{(i+1)} = u^{(i)} + \rho^{(i)}(u^{(i+\frac{1}{2})} - u^{(i)}). \end{array}\right. \tag{57}$$

As an application of [21, Corollary 28.3], we have:

Theorem 4.3 (Douglas–Rachford algorithm (57)**)** Let $x^{(0)} \in \mathcal{X}$ and $u^{(0)} \in \mathcal{X}$. Let $\tau > 0$. Let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence in $[0, 2]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(2 - \rho^{(i)}) = +\infty$. Then the sequences $(x^{(i)})_{i\in\mathbb{N}}$ and $(u^{(i)})_{i\in\mathbb{N}}$ defined by the iteration (57) converge weakly to a solution of (54) and to a solution of (55), respectively. Moreover, $(u^{(i+\frac{1}{2})} + v^{(i+\frac{1}{2})} + c)_{i\in\mathbb{N}}$ converges strongly to 0.

If one wants to implement the Douglas–Rachford algorithm like in (57), which may be interesting if the proximity operators of the conjugate functions are simple, it is better to use the scaled variable $\tilde{x} = x/\tau - c$:

Douglas–Rachford iteration for (54): for $i = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} v^{(i+\frac{1}{2})} = \text{prox}_{f^*/\tau}\big(\tilde{x}^{(i)} - u^{(i)}\big) \\ \tilde{x}^{(i+\frac{1}{2})} = \tilde{x}^{(i)} - u^{(i)} - v^{(i+\frac{1}{2})} - c \\ u^{(i+\frac{1}{2})} = \text{prox}_{g^*/\tau}\big(\tilde{x}^{(i+\frac{1}{2})} - v^{(i+\frac{1}{2})}\big) \\ \tilde{x}^{(i+1)} = \tilde{x}^{(i)} + \rho^{(i)}(\tilde{x}^{(i+\frac{1}{2})} - \tilde{x}^{(i)}) \\ u^{(i+1)} = u^{(i)} + \rho^{(i)}(u^{(i+\frac{1}{2})} - u^{(i)}), \end{array}\right. \tag{58}$$

and, after a certain number of iterations, use $x^{(i+\frac{1}{2})} = \tau(\tilde{x}^{(i+\frac{1}{2})} + c)$, which is in the domain of f, as an estimate of a minimizer of $f + g$.

5 The generalized Chambolle–Pock algorithm

Let $\mathcal{X}$, $\mathcal{U}$, $\mathcal{V}$ be real Hilbert spaces. Let $f \in \Gamma_0(\mathcal{V})$ and $g \in \Gamma_0(\mathcal{U})$. Let $K : \mathcal{X} \to \mathcal{V}$ and $L : \mathcal{X} \to \mathcal{U}$ be nonzero bounded linear operators. We consider the following problem, generalizing (39):

$$\underset{x\in\mathcal{X}}{\text{minimize}}\ f(Kx) + g(Lx) + \langle x, c\rangle, \tag{59}$$

where the solution set is supposed nonempty. We suppose that f and g satisfy some qualification constraints, so that the solution set of $0 \in K^*\partial f(Kx) + L^*\partial g(Lx) + c$ is nonempty. The dual problem is

$$\underset{(u,v)\in\mathcal{U}\times\mathcal{V}}{\text{minimize}}\ f^*(v) + g^*(u) \quad \text{s.t.} \quad K^*v + L^*u + c = 0. \tag{60}$$

Clearly, if $K = \mathrm{Id}$ and $L = \mathrm{Id}$, (59) and (60) revert to (54) and (55), and we can use the Douglas–Rachford algorithm (57) to solve these problems. On the other hand, if $K = \mathrm{Id}$ and $c = 0$, (59) and (60) revert to (39) and (42), and we can use the Chambolle–Pock algorithm (44) or (45).

Let $\tau > 0$, $\sigma > 0$, $\eta > 0$, let $x^{(0)} \in \mathcal{X}$, $u^{(0)} \in \mathcal{U}$, $v^{(0)} \in \mathcal{V}$, and let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence of relaxation parameters. We consider the algorithm:

Generalized Chambolle–Pock iteration (59) **and** (60): for $i = 0, 1, \dots$

$$\left\lfloor \begin{aligned} v^{(i+\frac{1}{2})} &= \mathrm{prox}_{\eta f^*}\Big(v^{(i)} + \eta K\big(x^{(i)} - \tau(L^*u^{(i)} + K^*v^{(i)} + c)\big)\Big)\\ x^{(i+\frac{1}{2})} &= x^{(i)} - \tau(L^*u^{(i)} + K^*v^{(i+\frac{1}{2})} + c)\\ u^{(i+\frac{1}{2})} &= \mathrm{prox}_{\sigma g^*}\big(u^{(i)} + \sigma L(2x^{(i+\frac{1}{2})} - x^{(i)})\big)\\ &= \mathrm{prox}_{\sigma g^*}\Big(u^{(i)} + \sigma L\big(x^{(i+\frac{1}{2})} - \tau(L^*u^{(i)} + K^*v^{(i+\frac{1}{2})} + c)\big)\Big)\\ x^{(i+1)} &= x^{(i)} + \rho^{(i)}(x^{(i+\frac{1}{2})} - x^{(i)})\\ u^{(i+1)} &= u^{(i)} + \rho^{(i)}(u^{(i+\frac{1}{2})} - u^{(i)})\\ v^{(i+1)} &= v^{(i)} + \rho^{(i)}(v^{(i+\frac{1}{2})} - v^{(i)}). \end{aligned}\right. \tag{61}$$

The primal–dual inclusion satisfied at every iteration is

$$\begin{pmatrix} 0\\ 0\\ 0 \end{pmatrix} \in \begin{pmatrix} c + L^*u^{(i+\frac{1}{2})} + K^*v^{(i+\frac{1}{2})}\\ -Lx^{(i+\frac{1}{2})} + \partial g^*(u^{(i+\frac{1}{2})})\\ -Kx^{(i+\frac{1}{2})} + \partial f^*(v^{(i+\frac{1}{2})}) \end{pmatrix} + \underbrace{\begin{pmatrix} \frac{1}{\tau}\mathrm{Id} & -L^* & 0\\ -L & \frac{1}{\sigma}\mathrm{Id} & 0\\ 0 & 0 & \frac{1}{\eta}\mathrm{Id} - \tau KK^* \end{pmatrix}}_{P} \begin{pmatrix} x^{(i+\frac{1}{2})} - x^{(i)}\\ u^{(i+\frac{1}{2})} - u^{(i)}\\ v^{(i+\frac{1}{2})} - v^{(i)} \end{pmatrix}. \tag{62}$$

P is strongly positive if and only if

$$\tau\sigma\|L\|^2 < 1 \quad \text{and} \quad \tau\eta\|K\|^2 < 1. \tag{63}$$

Therefore, since the algorithm is a preconditioned primal–dual proximal point algorithm, we can apply Theorem 2.3 and we get:

Theorem 5.1 (generalized Chambolle–Pock algorithm (61)**)** Let $x^{(0)} \in \mathcal{X}$, $u^{(0)} \in \mathcal{U}$, $v^{(0)} \in \mathcal{V}$. Let $\tau > 0$, $\sigma > 0$, $\eta > 0$ be such that $\tau\sigma\|L\|^2 < 1$ and $\tau\eta\|K\|^2 < 1$. Let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence in $[0, 2]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(2-\rho^{(i)}) = +\infty$. Then the sequences $(x^{(i)})_{i\in\mathbb{N}}$ and $(u^{(i)}, v^{(i)})_{i\in\mathbb{N}}$ defined by the iteration (61) converge weakly to a solution of (59) and to a solution of (60), respectively.

The generalized Chambolle–Pock algorithm (61) has appeared in the literature under different names, like Alternating Proximal Gradient Method [72], Generalized Alternating Direction Method of Multipliers [73], or Preconditioned ADMM [74]. The convergence results derived in this section generalize previously known results.

If $K = \mathrm{Id}$ and we set $\eta = 1/\tau$, we recover the Chambolle–Pock algorithm form I, to minimize $\tilde{f} + g \circ L$, with $\tilde{f} = f + \langle \cdot, c\rangle$. Indeed, in that case, the first two updates become:

$$\left| \begin{array}{l} v^{(i+\frac{1}{2})} = \mathrm{prox}_{f^*/\tau}\big(x^{(i)}/\tau - L^* u^{(i)} - c\big) \\ x^{(i+\frac{1}{2})} = x^{(i)} - \tau L^* u^{(i)} - \tau v^{(i+\frac{1}{2})} - \tau c \\ \qquad\quad = \mathrm{prox}_{\tau f}\big(x^{(i)} - \tau L^* u^{(i)} - \tau c\big) \\ \qquad\quad = \mathrm{prox}_{\tau \tilde{f}}\big(x^{(i)} - \tau L^* u^{(i)}\big). \end{array} \right. \tag{64}$$

Therefore, the generalized Chambolle–Pock indeed generalizes the Chambolle–Pock algorithm to any linear operator K. But we can remark that setting $\eta = 1/\tau$ is not allowed in Theorem 5.1. So, to extend the range of parameters to $\tau\sigma\|L\|^2 \leq 1$ and $\tau\eta\|K\|^2 \leq 1$, we now analyze the algorithm from another point of view: we show that it is a Douglas–Rachford algorithm in a primal–dual space. This analysis is inspired by the one in [37], showing that the Chambolle–Pock algorithm is a primal–dual Douglas–Rachford algorithm.

Let us assume that $\mathcal{X}$, $\mathcal{U}$, $\mathcal{V}$ are of finite dimension. Let $\tau > 0$, $\sigma > 0$, $\eta > 0$ be such that $\tau\sigma\|L\|^2 \leq 1$ and $\tau\eta\|K\|^2 \leq 1$. Let A be a linear operator from $\mathcal{A}$ to $\mathcal{V}$, for some finite-dimensional real Hilbert space $\mathcal{A}$, such that $KK^* + AA^* = (\tau\eta)^{-1}\mathrm{Id}$; for instance $A = \sqrt{(\tau\eta)^{-1}\mathrm{Id} - KK^*}$, the unique positive self-adjoint linear operator on $\mathcal{A} = \mathcal{V}$, the composition of which with itself is $(\tau\eta)^{-1}\mathrm{Id} - KK^*$. We do not need to exhibit A, the fact that it exists, in finite dimension, is sufficient here. Similarly, Let B be a linear operator from $\mathcal{B}$ to $\mathcal{U}$, for some finite-dimensional real Hilbert space $\mathcal{B}$, such that $LL^* + BB^* = (\tau\sigma)^{-1}\mathrm{Id}$. We introduce the real Hilbert space $\mathcal{Z} = \mathcal{X} \times \mathcal{A} \times \mathcal{B}$ and the functions of $\Gamma_0(\mathcal{Z})$:

$$F : (x, a, b) \in \mathcal{Z} \mapsto f(Kx + Aa) + \imath_0(b), \tag{65}$$

$$G : (x, a, b) \in \mathcal{Z} \mapsto g(Lx + Bb) + \imath_0(a), \tag{66}$$

where $\imath_0(s) = \{0 \text{ if } s = 0,\ +\infty \text{ else}\}$. Then we can rewrite the problem (59) as:

$$\underset{z \in \mathcal{Z}}{\mathrm{minimize}}\ F(z) + G(z) + \langle z, (c, 0, 0)\rangle. \tag{67}$$

We can now apply the Douglas–Rachford algorithm (57) in the augmented space $\mathcal{Z}$. For this, we need to observe that the proximity operators of F^* and G^* are easy to compute: for any $\tau > 0$, we have [37, eq. 15]

$$\mathrm{prox}_{F^*/\tau} : (x, a, b) \in \mathcal{Z} \mapsto (K^* v, A^* v, b) \text{ with } v = \mathrm{prox}_{\eta f^*}(\tau\eta K x + \tau\eta A a) \tag{68}$$

$$\mathrm{prox}_{G^*/\tau} : (x, a, b) \in \mathcal{Z} \mapsto (L^* u, a, B^* u) \text{ with } u = \mathrm{prox}_{\sigma g^*}(\tau\sigma L x + \tau\sigma B b) \tag{69}$$

After some substitutions, notably replacing AA^* by $(\tau\eta)^{-1}\mathrm{Id} - K^*K$ and BB^* by $(\tau\sigma)^{-1}\mathrm{Id} - L^*L$, we recover exactly the algorithm in (61).

Hence, as an application of Theorem 4.3, we get:

Theorem 5.2 (generalized Chambolle–Pock algorithm (61)**)** Suppose that $\mathcal{X}$, $\mathcal{U}$, $\mathcal{V}$ are of finite dimension. Let $x^{(0)} \in \mathcal{X}$, $u^{(0)} \in \mathcal{U}$, $v^{(0)} \in \mathcal{V}$. Let $\tau > 0$, $\sigma > 0$, $\eta > 0$ be such that $\tau\sigma\|L\|^2 \leq 1$ and $\tau\eta\|K\|^2 \leq 1$. Let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence in $[0, 2]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(2 - \rho^{(i)}) = +\infty$. Then the sequences $(x^{(i)})_{i\in\mathbb{N}}$ and $(u^{(i)}, v^{(i)})_{i\in\mathbb{N}}$ defined by the iteration (61) converge to a solution of (59) and to a solution of (60), respectively.

Therefore, in practice, one can keep τ as the single parameter to tune and set $\sigma = 1/(\tau\|L\|^2)$ and $\eta = 1/(\tau\|K\|^2)$.

6 The Condat–Vũ algorithm

Let us consider the primal optimization problem:

$$\underset{x\in\mathcal{X}}{\text{minimize}}\; f(x) + g(Lx) + h(x), \tag{70}$$

where $f \in \Gamma_0(\mathcal{X})$, $g \in \Gamma_0(\mathcal{U})$, $h : \mathcal{X} \to \mathbb{R}$ is a convex and Fréchet differentiable function with β-Lipschitz continuous gradient ∇h, for some real $\beta > 0$, and $L : \mathcal{X} \to \mathcal{U}$ is a bounded linear operator. Under some mild qualification constraints on f, g, h, the problem (70) is equivalent to solving

$$0 \in \partial f(x) + L^*\partial g(Lx) + \nabla h(x). \tag{71}$$

As a matter of fact, without further assumptions, a solution to (71) is a solution to (70), but the converse need not be true. For instance, the solution sets of (70) and (71) are the same if

$$0 \in \operatorname{sri}\big(L(\operatorname{dom}(f)) - \operatorname{dom}(g)\big), \tag{72}$$

where sri denotes the strong relative interior [21] and dom denotes the domain, which is the set of points where the function does not take the value $+\infty$. See [32, Proposition 27.5, Corollary 27.6] for other examples of qualification constraints. So, we assume that the solution set of (71) is nonempty.

Like in the previous sections, let us introduce a dual variable u, so that we can rewrite the problem (71) as the search of a pair of objects $z = (x, u)$ in $\mathcal{Z} = \mathcal{X} \times \mathcal{U}$ such that

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \underbrace{\begin{pmatrix} \partial f(x) + L^*u \\ -Lx + (\partial g)^{-1}(u) \end{pmatrix}}_{Mz} + \underbrace{\begin{pmatrix} \nabla h(x) \\ 0 \end{pmatrix}}_{Cz}. \tag{73}$$

A pair $(x, u) \in \mathcal{X} \times \mathcal{U}$ is a solution to (73) if and only if x is a solution to (71) and $u \in \partial g(Lx)$ is a solution to the dual problem

$$\underset{u\in\mathcal{U}}{\text{minimize}}\,(f + h)^*(-L^*u) + g^*(u). \tag{74}$$

The operator $M : \mathcal{Z} \to 2^{\mathcal{Z}}, (x, u) \mapsto (\partial f(x) + L^*u, -Lx + (\partial g)^{-1}u)$ is maximally monotone [21, Proposition 26.32 (iii)] and $C : \mathcal{Z} \to \mathcal{Z}, (x, u) \mapsto (\nabla h(x), 0)$ is ξ-cocoercive, with $\xi = 1/\beta$. Thus, it is again natural to think of the forward–backward iteration, with preconditioning. The difference with the construction in Section 3 is the presence of the nonlinear operator ∂f, which prevents us to express $x^{(i+\frac{1}{2})}$ in terms of $x^{(i)}$ and $u^{(i+\frac{1}{2})}$. Instead, the iteration is made explicit by canceling the dependence of $x^{(i+\frac{1}{2})}$ from $u^{(i+\frac{1}{2})}$ in P. That is, the iteration, written in implicit form, is:

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \underbrace{\begin{pmatrix} \partial f(x^{(i+\frac{1}{2})}) + L^*u^{(i+\frac{1}{2})} \\ -Lx^{(i+\frac{1}{2})} + (\partial g)^{-1}u^{(i+\frac{1}{2})} \end{pmatrix}}_{Mz^{(i+\frac{1}{2})}} + \underbrace{\begin{pmatrix} \nabla h(x^{(i)}) \\ 0 \end{pmatrix}}_{Cz^{(i)}} + \underbrace{\begin{pmatrix} \frac{1}{\tau}\mathrm{Id} & -L^* \\ -L & \frac{1}{\sigma}\mathrm{Id} \end{pmatrix}}_{P} \underbrace{\begin{pmatrix} x^{(i+\frac{1}{2})} - x^{(i)} \\ u^{(i+\frac{1}{2})} - u^{(i)} \end{pmatrix}}_{z^{(i+\frac{1}{2})} - z^{(i)}}, \tag{75}$$

where $\tau > 0$ and $\sigma > 0$ are two real parameters, $z^{(i)} = (x^{(i)}, u^{(i)})$ and $z^{(i+\frac{1}{2})} = (x^{(i+\frac{1}{2})}, u^{(i+\frac{1}{2})})$. Thus, the primal–dual forward–backward iteration is:

Condat–Vũ iteration form I for (70) **and** (74): for $i = 0, 1, \ldots$

$$\left\lfloor\begin{aligned} x^{(i+\frac{1}{2})} &= \operatorname{prox}_{\tau f}\big(x^{(i)} - \tau\nabla h(x^{(i)}) - \tau L^*u^{(i)}\big) \\ u^{(i+\frac{1}{2})} &= \operatorname{prox}_{\sigma g^*}\big(u^{(i)} + \sigma L(2x^{(i+\frac{1}{2})} - x^{(i)})\big) \\ x^{(i+1)} &= x^{(i)} + \rho^{(i)}(x^{(i+\frac{1}{2})} - x^{(i)}) \\ u^{(i+1)} &= u^{(i)} + \rho^{(i)}(u^{(i+\frac{1}{2})} - u^{(i)}). \end{aligned}\right. \tag{76}$$

This algorithm was proposed independently by the first author [33] and by B. C. Vũ [35].

An alternative is to update u before x, instead of the converse. This yields a different algorithm, characterized by the primal–dual inclusion

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \underbrace{\begin{pmatrix} \partial f(x^{(i+\frac{1}{2})}) + L^* u^{(i+\frac{1}{2})} \\ -Lx^{(i+\frac{1}{2})} + (\partial g)^{-1} u^{(i+\frac{1}{2})} \end{pmatrix}}_{Mz^{(i+\frac{1}{2})}} + \underbrace{\begin{pmatrix} \nabla h(x^{(i)}) \\ 0 \end{pmatrix}}_{Cz^{(i)}} + \underbrace{\begin{pmatrix} \frac{1}{\tau}\mathrm{Id} & L^* \\ L & \frac{1}{\sigma}\mathrm{Id} \end{pmatrix}}_{P} \underbrace{\begin{pmatrix} x^{(i+\frac{1}{2})} - x^{(i)} \\ u^{(i+\frac{1}{2})} - u^{(i)} \end{pmatrix}}_{z^{(i+\frac{1}{2})} - z^{(i)}}. \tag{77}$$

This corresponds to the primal–dual forward–backward iteration:

Condat–Vũ iteration form II for (70) **and** (74): for $i = 0, 1, \ldots$

$$\begin{aligned} & u^{(i+\frac{1}{2})} = \mathrm{prox}_{\sigma g^*}\big(u^{(i)} + \sigma L x^{(i)}\big) \\ & x^{(i+\frac{1}{2})} = \mathrm{prox}_{\tau f}\big(x^{(i)} - \tau \nabla h(x^{(i)}) - \tau L^*(2u^{(i+\frac{1}{2})} - u^{(i)})\big) \\ & u^{(i+1)} = u^{(i)} + \rho^{(i)}(u^{(i+\frac{1}{2})} - u^{(i)}) \\ & x^{(i+1)} = x^{(i)} + \rho^{(i)}(x^{(i+\frac{1}{2})} - x^{(i)}). \end{aligned} \tag{78}$$

As an application of Theorem 2.2, we obtain the following convergence result [33, Theorem 3.1]:

Theorem 6.1 (Condat–Vũ algorithm (76) **or** (78)**)** Let $x^{(0)} \in \mathcal{X}$ and $u^{(0)} \in \mathcal{U}$. Let $\tau > 0$ and $\sigma > 0$ be such that $\tau\big(\sigma\|L\|^2 + \frac{\beta}{2}\big) < 1$. Set $\delta = 2 - \frac{\beta}{2}\big(\frac{1}{\tau} - \sigma\|L\|^2\big)^{-1} > 1$. Let $(\rho^{(i)})_{i\in\mathbb{N}}$ be a sequence in $[0, \delta]$ such that $\sum_{i\in\mathbb{N}} \rho^{(i)}(\delta - \rho^{(i)}) = +\infty$. Then the sequences $(x^{(i)})_{i\in\mathbb{N}}$ and $(u^{(i)})_{i\in\mathbb{N}}$ defined either by the iteration (76) or by the iteration (78) converge weakly to a solution of (70) and to a solution of (74), respectively.

Proof In view of (75) and (77), this is Theorem 2.2 applied to the problem (73). The condition on τ and σ implies that $\sigma\tau\|L\|^2 < 1$, so that P is strongly positive, by virtue of the properties of the Schur complement. Let us establish the cocoercivity of $P^{-1}C$ in $\mathcal{Z}_P$. Set $\chi = \frac{1}{\beta}\big(\frac{1}{\tau} - \sigma\|L\|^2\big)$. In both cases (75) and (77), we have, for every $z = (x, u)$ and $z' = (x', u')$ in $\mathcal{Z}$,

$$\begin{aligned} \|P^{-1}Cz - P^{-1}Cz'\|_P^2 &= \langle P^{-1}Cz - P^{-1}Cz', Cz - Cz' \rangle && (79) \\ &= \Big\langle \tfrac{1}{\sigma}\big(\tfrac{1}{\sigma\tau}\mathrm{Id} - L^*L\big)^{-1}\big(\nabla h(x) - \nabla h(x')\big), \nabla h(x) - \nabla h(x') \Big\rangle && (80) \\ &\leq \big(\tfrac{1}{\tau} - \sigma\|L\|^2\big)^{-1}\big\langle \nabla h(x) - \nabla h(x'), \nabla h(x) - \nabla h(x') \big\rangle && (81) \\ &\leq \beta\big(\tfrac{1}{\tau} - \sigma\|L\|^2\big)^{-1}\big\langle x - x', \nabla h(x) - \nabla h(x') \big\rangle && (82) \\ &= \beta\big(\tfrac{1}{\tau} - \sigma\|L\|^2\big)^{-1}\big\langle z - z', Cz - Cz' \big\rangle && (83) \\ &= \beta\big(\tfrac{1}{\tau} - \sigma\|L\|^2\big)^{-1}\big\langle z - z', P^{-1}Cz - P^{-1}Cz' \big\rangle_P. && (84) \end{aligned}$$

So, $P^{-1}C$ is χ-cocoercive in $\mathcal{Z}_P$. Moreover, $\chi > 1/2$ if and only if $\tau(\sigma\|L\|^2 + \beta/2) < 1$. Finally, $\delta = 2 - 1/(2\chi)$. □

We can observe that if $h = 0$, the Condat–Vũ iteration reverts to the Chambolle–Pock iteration. So, the former can be viewed as a generalization of the latter. Accordingly, if we set $\beta = 0$ in Theorem 6.1, we recover Theorem 4.1.

The Condat–Vũ algorithm and the Loris–Verhoeven algorithms are two different primal–dual forward–backward algorithms. The latter needs to have $f = 0$, but larger values of τ and σ are allowed; this may be beneficial to the convergence speed in practice.

For the Condat–Vũ algorithm too, let us focus on the case where h is quadratic; that is,

$$h : x \mapsto \tfrac{1}{2}\langle x, Qx \rangle + \langle x, c \rangle, \tag{85}$$

for some bounded, self-adjoint, positive, nonzero, linear operator Q on $\mathcal{X}$ and some element $c \in \mathcal{X}$. We have $\beta = \|Q\|$. We can rewrite the primal–dual inclusion (75), which characterizes the Condat–Vũ iteration (76), as

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \underbrace{\begin{pmatrix} (\partial f + \nabla h)(x^{(i+\frac{1}{2})}) + L^* u^{(i+\frac{1}{2})} \\ -Lx^{(i+\frac{1}{2})} + (\partial g)^{-1} u^{(i+\frac{1}{2})} \end{pmatrix}}_{Mz^{(i+\frac{1}{2})}} + \underbrace{\begin{pmatrix} \frac{1}{\tau}\mathrm{Id} - Q & -L^* \\ -L & \frac{1}{\sigma}\mathrm{Id} \end{pmatrix}}_{P} \underbrace{\begin{pmatrix} x^{(i+\frac{1}{2})} - x^{(i)} \\ u^{(i+\frac{1}{2})} - u^{(i)} \end{pmatrix}}_{z^{(i+\frac{1}{2})} - z^{(i)}}. \tag{86}$$

Similarly, we can rewrite the primal–dual inclusion (77), which characterizes the second form of the Condat–Vũ iteration (78), as (86), with $-L$ replaced by L in P.

In both cases, using the properties of the Schur complement, P is strongly positive if and only if

$$\tau \|Q + \sigma L^* L\| < 1 \tag{87}$$

(which implies that $\tau < 1/\beta$). A sufficient condition for this inequality to hold is $\tau(\sigma\|L\|^2 + \beta) < 1$. However, in some applications, $\|Q + \sigma L^* L\|$ may be smaller than $\sigma\|L\|^2 + \beta$, so that larger stepsizes τ and σ may be used when h is quadratic, for the benefit of convergence speed.

Thus, when h is quadratic, the Condat–Vũ iteration can be viewed as a preconditioned Chambolle–Pock iteration. Accordingly, as an application of Theorem 2.4, we have:

Theorem 6.2 (Condat–Vũ algorithm (76) **or** (78)**, quadratic case)** Let $x^{(0)} \in \mathcal{X}$ and $u^{(0)} \in \mathcal{U}$. Let $\tau > 0$ and $\sigma > 0$ be such that $\tau \|Q + \sigma L^* L\| < 1$. Let $(\rho^{(i)})_{i \in \mathbb{N}}$ be a sequence in $[0, 2]$ such that $\sum_{i \in \mathbb{N}} \rho^{(i)}(2 - \rho^{(i)}) = +\infty$. Then the sequences $(x^{(i)})_{i \in \mathbb{N}}$ and $(u^{(i)})_{i \in \mathbb{N}}$ defined either by the iteration (76) or by the iteration (78) converge weakly to a solution of (70) and to a solution of (74), respectively.

Finally, we can apply Theorem 2.5 to allow $\tau\|Q + \sigma L^* L\| = 1$. For this, we first strengthen the analysis in (79)–(84). Let us suppose that $\tau\sigma\|L\|^2 < 1$. Then $P^{-1}C$ is χ-cocoercive in $\mathcal{Z}_P$, with $\chi = \| \left(\frac{1}{\tau}\mathrm{Id} - \sigma L^* L\right)^{-1} Q\|^{-1}$. $\chi \geq 1$ if $\tau\|Q + \sigma L^* L\| \leq 1$. Hence, we have:

Theorem 6.3 (Condat–Vũ algorithm (76) **or** (78)**, quadratic case)** Suppose that $\mathcal{X}$ and $\mathcal{U}$ are of finite dimension. Let $x^{(0)} \in \mathcal{X}$ and $u^{(0)} \in \mathcal{U}$. Let $\tau > 0$ and $\sigma > 0$ be such that $\tau\sigma\|L\|^2 < 1$ and $\tau\|Q + \sigma L^* L\| \leq 1$. Let $(\rho^{(i)})_{i \in \mathbb{N}}$ be a sequence in $[0, 2]$ such that $\sum_{i \in \mathbb{N}} \rho^{(i)}(2 - \rho^{(i)}) = +\infty$. Then the sequences $(x^{(i)})_{i \in \mathbb{N}}$ and $(u^{(i)})_{i \in \mathbb{N}}$ defined either by the iteration (76) or by the iteration (78) converge to a solution of (70) and to a solution of (74), respectively.

We can note that $\tau(\beta + \sigma\|L\|^2) \leq 1$ implies $\tau\sigma\|L\|^2 < 1$ and $\tau\|Q + \sigma L^* L\| \leq 1$.

References

[1] D. P. Palomar and Y. C. Eldar, Eds., *Convex Optimization in Signal Processing and Communications*. Cambridge University Press, 2009.

[2] J.-L. Starck, F. Murtagh, and J. Fadili, *Sparse Image and Signal Processing: Wavelets, Curvelets, Morphological Diversity*. Cambridge University Press, 2010.

[3] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, “Optimization with sparsity-inducing penalties,” *Foundations and Trends in Machine Learning*, vol. 4, no. 1, pp. 1–106, 2012.

[4] N. G. Polson, J. G. Scott, and B. T. Willard, “Proximal algorithms in statistics and machine learning,” *Statist. Sci.*, vol. 30, no. 4, pp. 559–581, 2015.

[5] A. Chambolle and T. Pock, "An introduction to continuous optimization for imaging," *Acta Numerica*, vol. 25, pp. 161–319, 2016.

[6] G. Stathopoulos, H. Shukla, A. Szucs, Y. Pu, and C. N. Jones, "Operator splitting methods in control," *Foundations and Trends in Systems and Control*, vol. 3, no. 3, pp. 249–362, 2016.

[7] L. Condat, "Discrete total variation: New definition and minimization," *SIAM J. Imaging Sciences*, vol. 10, no. 3, pp. 1258–1290, 2017.

[8] M. Elad, *Sparse and redundant representations: From theory to applications in signal and image processing*. Springer, 2010.

[9] S. Sra, S. Nowozin, and S. J. Wright, *Optimization for Machine Learning*. The MIT Press, 2011.

[10] S. Bubeck, "Convex optimization: Algorithms and complexity," *Foundations and Trends in Machine Learning*, vol. 8, no. 3–4, pp. 231–357, 2015.

[11] A. Emrouznejad, Ed., *Big Data Optimization: Recent Developments and Challenges*. Springer, 2016.

[12] K. Polisano, L. Condat, M. Clausel, and V. Perrier, "A convex approach to super-resolution and regularization of lines in images," *SIAM Journal on Imaging Sciences*, vol. 12, no. 1, pp. 211–258, 2019.

[13] P. L. Combettes and J.-C. Pesquet, "Proximal splitting methods in signal processing," in *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, H. H. Bauschke, R. Burachik, P. L. Combettes, V. Elser, D. R. Luke, and H. Wolkowicz, Eds. New York: Springer-Verlag, 2010.

[14] R. I. Boţ, E. R. Csetnek, and C. Hendrich, *Recent Developments on Primal–Dual Splitting Methods with Applications to Convex Minimization*. Springer New York, 2014, pp. 57–99.

[15] N. Parikh and S. Boyd, "Proximal algorithms," *Foundations and Trends in Optimization*, vol. 3, no. 1, pp. 127–239, 2014.

[16] N. Komodakis and J.-C. Pesquet, "Playing with duality: An overview of recent primal–dual approaches for solving large-scale optimization problems," *IEEE Signal Process. Mag.*, vol. 32, no. 6, pp. 31–54, Nov. 2015.

[17] E. K. Ryu and S. Boyd, "A primer on monotone operator methods," *Appl. Comput. Math.*, vol. 1, no. 15, pp. 3–43, 2016.

[18] R. Glowinski, S. J. Osher, and W. Yin, Eds., *Splitting Methods in Communication, Imaging, Science, and Engineering*. Springer International Publishing, 2016.

[19] E. R. Csetnek, "Recent advances in splitting methods for monotone inclusions and nonsmooth optimization problems," Habilitation thesis, University of Vienna, Vienna, Austria, 2016.

[20] A. Beck, *First-Order Methods in Optimization*, ser. MOS-SIAM Series on Optimization. SIAM, 2017.

[21] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, 2nd ed. New York: Springer, 2017.

[22] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[23] P. L. Combettes, "Monotone operator theory in convex optimization," *Mathematical Programming*, vol. 170, no. 1, pp. 177–206, Jul. 2018.

[24] B. Mercier, "Topics in finite element solution of elliptic problems. No. 63 in lectures on mathematics," Tata Institute of Fundamental Research, Bombay, 1979.

[25] P.-L. Lions and B. Mercier, “Splitting algorithms for the sum of two nonlinear operators,” *SIAM J. Numer. Anal.*, vol. 16, no. 6, pp. 964–979, 1979.

[26] G. B. Passty, “Ergodic convergence to a zero of the sum of monotone operators in Hilbert space,” *Journal of Mathematical Analysis and Applications*, vol. 72, pp. 383–390, 1979.

[27] D. Gabay, “Applications of the method of multipliers to variational inequalities,” in *Augmented Lagrangian Methods: Applications to the Solution of Boundary-Value Problems*, M. Fortin and R. Glowinski, Eds., Amsterdam, Netherlands, 1983.

[28] P. Tseng, “Applications of splitting algorithm to decomposition in convex programming and variational inequalities,” *SIAM J. Control Optim.*, vol. 29, pp. 119–138, 1991.

[29] H.-G. Chen, “Forward–backward splitting techniques: Theory and applications,” Ph.D. dissertation, University of Washington, Seattle, WA, 1994.

[30] P. Combettes, “Solving monotone inclusions via compositions of nonexpansive averaged operators,” *Optimization*, vol. 53, no. 5–6, pp. 475–504, 2004.

[31] P. L. Combettes and V. R. Wajs, “Signal recovery by proximal forward–backward splitting,” *Multiscale Model. Simul.*, vol. 4, pp. 1168–1200, 2005.

[32] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. New York: Springer, 2011.

[33] L. Condat, “A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms,” *J. Optim. Theory Appl.*, vol. 158, no. 2, pp. 460–479, 2013.

[34] R. T. Rockafellar, “Monotone operators and the proximal point algorithm,” *SIAM J. Control Optim.*, vol. 14, no. 5, pp. 877–898, 1976.

[35] B. C. Vũ, “A splitting algorithm for dual monotone inclusions involving cocoercive operators,” *Adv. Comput. Math.*, vol. 38, no. 3, pp. 667–681, Apr. 2013.

[36] R. I. BoŢ, E. R. Csetnek, A. Heinrich, and C. Hendrich, “On the convergence rate improvement of a primal–dual splitting algorithm for solving monotone inclusion problems,” *Mathematical Programming*, vol. 150, no. 2, pp. 251–279, May 2015.

[37] D. O’Connor and L. Vandenberghe, “On the equivalence of the primal-dual hybrid gradient method and Douglas–Rachford splitting,” *Mathematical Programming*, 2018, to be published.

[38] A. Chambolle and T. Pock, “A first-order primal-dual algorithm for convex problems with applications to imaging,” *J. Math. Imaging Vision*, vol. 40, no. 1, pp. 120–145, May 2011.

[39] J. J. Moreau, “Fonctions convexes duales et points proximaux dans un espace hilbertien,” *C. R. Acad. Sci. Paris Sér. A Math.*, vol. 255, pp. 2897–2899, 1962.

[40] C. Chaux, P. L. Combettes, J.-C. Pesquet, and V. R. Wajs, “A variational formulation for frame based inverse problems,” *Inverse Problems*, vol. 23, pp. 1495–1518, Jun. 2007.

[41] M. El Gheche, G. Chierchia, and J.-C. Pesquet, “Proximity operators of discrete information divergences,” *IEEE Trans. Inf. Theory*, vol. 64, no. 2, pp. 1092–1104, Feb. 2018.

[42] I. Loris and C. Verhoeven, “On a generalization of the iterative soft-thresholding algorithm for the case of non-separable penalty,” *Inverse Problems*, vol. 27, no. 12, 2011.

[43] P. Chen, J. Huang, and X. Zhang, “A primal–dual fixed point algorithm for convex separable minimization with applications to image restoration,” *Inverse Problems*, vol. 29, no. 2, 2013.

[44] Y. Drori, S. Sabach, and M. Teboulle, "A simple algorithm for a class of nonsmooth convex concave saddle-point problems," *Oper. Res. Lett.*, vol. 43, no. 2, pp. 209–214, 2015.

[45] P. L. Combettes, L. Condat, J.-C. Pesquet, and B. C. Vũ, "A forward–backward view of some primal–dual optimization methods in image recovery," in *Proc. of IEEE ICIP*, Paris, France, Oct. 2014.

[46] M. Yan, "A new Primal–Dual algorithm for minimizing the sum of three functions with a linear operator," *Journal of Scientific Computing*, vol. 76, no. 3, pp. 1698–1717, Sep. 2018.

[47] E. Esser, X. Zhang, and T. Chan, "A general framework for a class of first order primal–dual algorithms for convex optimization in imaging science," *SIAM J. Imaging Sci.*, vol. 3, no. 4, pp. 1015–1046, 2010.

[48] X. Zhang, M. Burger, and S. Osher, "A unified primal–dual algorithm framework based on Bregman iteration," *J. Sci. Comput.*, vol. 46, no. 1, pp. 20–46, 2010.

[49] B. S. He and X. M. Yuan, "Convergence analysis of primal–dual algorithms for a saddle-point problem: from contraction perspective," *SIAM J. Imaging Sci.*, vol. 5, pp. 119–149, 2012.

[50] J. Eckstein and D. P. Bertsekas, "On the Douglas–Rachford splitting method and the proximal point algorithm for maximal monotone operators," *Math. Programming*, vol. 55, pp. 293–318, 1992.

[51] B. F. Svaiter, "On weak convergence of the Douglas–Rachford method," *SIAM J. Control Optim.*, vol. 49, no. 1, pp. 280–287, 2011.

[52] 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," *Revue Française d'Automatique, Informatique et Recherche Opérationnelle*, vol. 9, pp. 41–76, 1975.

[53] D. Gabay and B. Mercier, "A dual algorithm for the solution of nonlinear variational problems via finite element approximation," *Computers & Mathematics with Applications*, vol. 2, no. 1, pp. 17–40, 1976.

[54] J. Eckstein, "Splitting methods for monotone operators with applications to parallel optimization," Ph.D. dissertation, MIT, Cambridge, MA, 1989.

[55] J.-F. Cai, S. Osher, and Z. Shen, "Split Bregman methods and frame based image restoration," *SIAM J. Multiscale Model. Simul.*, vol. 8, no. 2, pp. 337–369, 2010.

[56] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[57] J. Eckstein and W. T. Yao, "Augmented Lagrangian and alternating direction methods for convex optimization: A tutorial and some illustrative computational results," 2012, rUTCOR Research Report.

[58] J. Eckstein and W. Yao, "Understanding the convergence of the alternating direction method of multipliers: Theoretical and computational perspectives," *Pacific Journal of Optimization*, vol. 11, pp. 619–644, 2015.

[59] R. Glowinski, *On Alternating Direction Methods of Multipliers: A Historical Perspective*. Springer Netherlands, 2014, pp. 59–82.

[60] R. Glowinsk, T.-W. Pan, and X.-C. Tai, *Some Facts About Operator-Splitting and Alternating Direction Methods*. Cham: Springer International Publishing, 2016, pp. 19–94.

[61] R. Glowinski and P. L. Tallec, "Augmented Lagrangian and operator-splitting methods in nonlinear mechanics," *SIAM Studies in Applied and Numerical Mathematics*, 1989.

[62] T. Goldstein and S. Osher, "The split Bregman method for L1-regularized problems," *SIAM J. Imaging Sci.*, vol. 2, no. 2, pp. 323–343, 2009.

[63] S. Setzer, "Operator splittings, Bregman methods and frame shrinkage in image processing," *International journal of Computer Vision*, vol. 92, no. 3, pp. 265–280, 2011.

[64] Y. Chen, W. W. Hager, M. Yashtini, X. Ye, and H. Zhang, "Bregman operator splitting with variable stepsize for total variation image reconstruction," *Computational Optimization and Applications*, vol. 54, pp. 317–342, 2013.

[65] Q. Wu, X. He, and T. Hintz, "Augmented Lagrangian method, dual methods, and split Bregman iteration for ROF, vectorial TV, and high order models," *SIAM Journal on Imaging Sciences*, vol. 3, no. 3, pp. 300–339, 2010.

[66] T. Goldstein, X. Bresson, and S. Osher, "Geometric applications of the split Bregman method: Segmentation and surface reconstruction," *J. Sci. Comput.*, vol. 45, pp. 272–293, Oct. 2010.

[67] J.-F. Cai, S. Osher, and Z. Shen, "Linearized Bregman iterations for frame-based image deblurring," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 226–252, 2009.

[68] S. Setzer, "Split Bregman algorithm, Douglas–Rachford splitting and frame shrinkage," in *Proc. of International Conference on Scale Space and Variational Methods in Computer Vision*, 2009, pp. 464–476.

[69] M. A. T. Figueiredo and J. M. Bioucas-Dias, "Restoration of Poissonian images using alternating direction optimization," *IEEE Trans. Image Process.*, vol. 19, no. 12, pp. 3133–3145, Dec. 2010.

[70] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "Fast image recovery using variable splitting and constrained optimization," *IEEE Trans. Signal Process.*, vol. 19, no. 9, pp. 2345–2356, 2010.

[71] ——, "An augmented Lagrangian approach to the constrained optimization formulation of imaging inverse problems," *IEEE Trans. Image Process.*, vol. 20, no. 3, pp. 681–695, Mar. 2011.

[72] S. Ma, "Alternating proximal gradient method for convex minimization," *J. Sci. Comput.*, vol. 68, pp. 546–572, Aug. 2016.

[73] W. Deng and W. Yin, "On the global and linear convergence of the generalized alternating direction method of multipliers," *J. Sci. Comput.*, vol. 66, no. 3, pp. 889–916, 2015.

[74] K. Bredies and H. Sun, "A proximal point analysis of the preconditioned alternating direction method of multipliers," *J. Optim. Theory Appl.*, vol. 173, pp. 878–907, 2017.