Bregman three-operator splitting methods

Xin Jiang* Lieven Vandenberghe*

February 28, 2022

Abstract

The paper presents primal–dual proximal splitting methods for convex optimization, in which generalized Bregman distances are used to define the primal and dual proximal update steps. The methods extend the primal and dual Condat–Vũ algorithms and the primal–dual three-operator (PD3O) algorithm. The Bregman extensions of the Condat–Vũ algorithms are derived from the Bregman proximal point method applied to a monotone inclusion problem. Based on this interpretation, a unified framework for the convergence analysis of the two methods is presented. We also introduce a line search procedure for stepsize selection in the Bregman dual Condat–Vũ algorithm applied to equality-constrained problems. Finally, we propose a Bregman extension of PD3O and analyze its convergence.

1 Introduction

We discuss proximal splitting methods for optimization problems in the form

$$\text{minimize} \quad f(x) + g(Ax) + h(x), \tag{1}$$

where f, g, and h are convex functions, and h is differentiable. This general problem covers a wide variety of applications in machine learning, signal and image processing, operations research, control, and other fields [18, 40, 30, 11]. In this paper, we consider proximal splitting methods based on Bregman distances for solving (1) and some interesting special cases of (1).

Recently, several primal–dual first-order methods have been proposed for the three-term problem (1): the Condat–Vũ algorithm [19, 49, 52], the primal–dual three-operator (PD3O) algorithm [50], and the primal–dual Davis–Yin (PDDY) algorithm [44]. Algorithms for some special cases of (1) are also of interest. These include the Chambolle–Pock algorithm, also known as the primal–dual hybrid gradient (PDHG) method [10, 12] (when $h = 0$), the Loris–Verhoeven algorithm [33, 15, 22] (when $f = 0$), the proximal gradient algorithm (when $g = 0$), and the Davis–Yin splitting algorithm [20] (when $A = I$). All these methods handle the nonsmooth functions f and g via the standard Euclidean proximal operator.

To further improve the efficiency of proximal algorithms, proximal operators based on generalized Bregman distances have been proposed and incorporated in many methods [14, 23, 26, 2, 47, 3, 6, 34, 45]. Bregman distances offer two potential benefits. First, the Bregman distance can help build a more accurate local optimization model around the current iterate. This is often interpreted as a form of preconditioning. For example, diagonal or quadratic preconditioning [41, 28, 32] has

*Department of Electrical and Computer Engineering, UCLA. Email: jiangxjames@ucla.edu, vandenbe@ucla.edu.

been shown to improve the convergence rate of PDHG, as well as the accuracy of the computed solution [1]. As a second benefit, a Bregman proximal operator of a function may be easier to compute than the standard Euclidean proximal operator, and therefore reduce the complexity per iteration of an optimization algorithm. Recent applications of this kind include optimal transport problems [16], optimization over nonnegative trigonometric polynomials [13], and sparse semidefinite programming [29].

Extending standard proximal methods and their convergence analysis to Bregman distances is not straightforward because some fundamental properties of the Euclidean proximal operator no longer hold for Bregman proximal operators. An example is the Moreau decomposition which relates the (Euclidean) proximal operators of a closed convex function and its conjugate [36]. Another example is the simple relation between the proximal operators of a function g and the composition with a linear function $g(Ax)$ when AA^T is a multiple of the identity; see, *e.g.*, [18, 4]. This composition rule is used in [38] to establish the equivalence between some well-known first-order proximal methods for problem (1) with $A = I$ and with general A.

The purpose of this paper is to present new Bregman extensions and convergence results for the Condat–Vũ and PD3O algorithms. The main contributions are as follows.

- The Condat–Vũ algorithm [19, 49] exists in a primal and a dual variant. We discuss extensions of the two algorithms that use Bregman proximal operators in the primal and dual updates. The Bregman primal Condat–Vũ algorithm first appeared in [12, Algorithm 1], and is also a special case of the algorithm proposed in [51] for a more general convex–concave saddle point problem. We give a new derivation of this method and its dual variant, by applying the Bregman proximal point method to the primal–dual optimality conditions. Based on the interpretation, we provide a unified framework for the convergence analysis of the two variants, and show an $O(1/k)$ ergodic convergence rate, which is consistent with previous results for Euclidean proximal operators in [19, 49] and Bregman proximal operators in [12]. We also give a convergence result for the primal and dual iterates.

- We propose an easily implemented backtracking line search technique for selecting stepsizes in the Bregman dual Condat–Vũ algorithm for problems with equality constraints. The proposed backtracking procedure is similar to the technique in [35] for the special setting of PDHG with Euclidean proximal operators, but has some important differences even in this special case. We give a detailed analysis of the algorithm with line search and recover the $O(1/k)$ ergodic rate of convergence for related algorithms in [35, 29].

- We propose a Bregman extension for PD3O and establish an ergodic convergence result.

The paper is organized as follows. Section 2 gives a precise statement of the problem (1), and reviews the duality theory that will be used in the rest of the paper. In Section 3 we review some well-known first-order proximal methods and establish connections between them. Section 4 provides some necessary background on Bregman distances. In Section 5 we discuss the Bregman primal and dual Condat–Vũ algorithms and analyze their convergence. The line search technique and its convergence are discussed in Section 6. In Section 7 we extend PD3O to a Bregman proximal method and analyze its convergence. Section 8 contains results of a numerical experiment.

2 Duality theory and merit functions

This section summarizes the facts from convex duality theory that underlie the primal–dual methods discussed in the paper. We also describe primal–dual merit functions that will be used in the convergence analysis.

We use the notation $\langle x, y\rangle = x^T y$ for the standard inner product of vectors x and y, and $\|x\| = \langle x, x\rangle^{1/2}$ for the Euclidean norm of a vector x. Other norms will be distinguished by a subscript.

2.1 Problem formulation

In (1) the vector x is an n-vector and A is an $m \times n$ matrix. The functions f, g, h are closed and convex, and h is differentiable, *i.e.*,

$$h(x) \geq h(x') + \langle \nabla h(x'), x - x'\rangle \quad \text{for all } x, x' \in \mathbf{dom}\, h,$$

where $\mathbf{dom}\, h$ is an open convex set. We assume that $f + h$ and g are proper, *i.e.*, have nonempty domains.

An important example of (1) is $g = \delta_C$, the indicator function of a closed convex set C. With $g = \delta_C$, the problem is equivalent to

$$\begin{array}{ll} \text{minimize} & f(x) + h(x) \\ \text{subject to} & Ax \in C. \end{array}$$

For $C = \{b\}$ the constraints are a set of linear equations $Ax = b$. This special case actually covers all applications of the more general problem (1), since (1) can be reformulated as

$$\begin{array}{ll} \text{minimize} & f(x) + g(y) + h(x) \\ \text{subject to} & Ax = y, \end{array}$$

at the expense of increasing the problem size by introducing a splitting variable y.

2.2 Dual problem and optimality conditions

The dual of problem (1) is

$$\text{maximize} \quad -(f + h)^*(-A^T z) - g^*(z), \tag{2}$$

where $(f + h)^*$ and g^* are the conjugates of $f + h$ and g:

$$(f + h)^*(w) = \sup_x \left(\langle w, x\rangle - f(x) - h(x)\right), \qquad g^*(z) = \sup_y \left(\langle z, y\rangle - g(y)\right).$$

The conjugate $(f + h)^*$ is the infimal convolution f^* and h^*, denoted by $f^* \,\square\, h^*$:

$$f^* \,\square\, h^* = \inf_w \left((f^*(w) + h^*(z - w)\right).$$

The primal–dual optimality conditions for (1) and (2) are

$$0 \in \partial f(x) + \nabla h(x) + A^T z, \qquad 0 \in \partial g^*(z) - Ax.$$

Here ∂f and ∂g^* are the subdifferentials of f and g^*. We often write the optimality conditions as

$$0 \in \begin{bmatrix} 0 & A^T \\ -A & 0 \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} + \begin{bmatrix} \partial f(x) + \nabla h(x) \\ \partial g^*(z) \end{bmatrix}. \tag{3}$$

Throughout the paper, we assume that the optimality conditions (3) are solvable.

We will refer to the convex–concave function

$$\mathcal{L}(x, z) = f(x) + h(x) + \langle z, Ax \rangle - g^*(z)$$

as the *Lagrangian* of (1). We follow the convention that $\mathcal{L}(x, z) = +\infty$ if $x \notin \mathbf{dom}(f + h)$ and $\mathcal{L}(x, z) = -\infty$ if $x \in \mathbf{dom}(f + h)$ and $z \notin \mathbf{dom}\, g^*$. The objective functions in (1) and the dual problem (2) can be expressed as

$$\sup_z \mathcal{L}(x, z) = f(x) + h(x) + g(Ax), \qquad \inf_x \mathcal{L}(x, z) = -(f + h)^*(-A^T z) - g^*(z).$$

Solutions $x^\star$, $z^\star$ of the optimality conditions (3) form a saddle-point of $\mathcal{L}$, *i.e.*, satisfy

$$\inf_x \sup_z \mathcal{L}(x, z) = \sup_z \mathcal{L}(x^\star, z) = \mathcal{L}(x^\star, z^\star) = \inf_x \mathcal{L}(x, z^\star) = \sup_z \inf_x \mathcal{L}(x, z). \tag{4}$$

In particular, $\mathcal{L}(x^\star, z^\star)$ is the optimal value of (1) and (2).

2.3 Merit functions

The algorithms discussed in this paper generate primal and dual iterates and approximate solutions x, z with $x = \mathbf{dom}\, f \cap \mathbf{dom}\, h$ and $z \in \mathbf{dom}\, g^*$. The feasibility conditions $Ax \in \mathbf{dom}\, g$ and $-A^T z \in \mathbf{dom}(f + h)^*$ are not necessarily satisfied. Hence the duality gap

$$\sup_{z'} \mathcal{L}(x, z') - \inf_{x'} \mathcal{L}(x', z) = f(x) + h(x) + g(Ax) + (f + h)^*(-A^T z) + g^*(z) \tag{5}$$

may not always be useful as a merit function to measure convergence.

If we add constraints $x' \in X$ and $z' \in Z$ to the optimization problems on the left-hand side of (5), where X and Z are compact sets, we obtain a function

$$\eta(x, z) = \sup_{z' \in Z} \mathcal{L}(x, z') - \inf_{x' \in X} \mathcal{L}(x', z) \tag{6}$$

defined for all $x \in \mathbf{dom}(f + h)$ and $z \in \mathbf{dom}\, g^*$. This follows from the fact that the functions $f + h + \delta_X$ and $g^* + \delta_Z$ are closed and co-finite, so their conjugates have full domain [43, Corollary 13.3.1]. If $\eta(x, z)$ is easily computed, and $\eta(x, z) \geq 0$ for all $x \in \mathbf{dom}(f + h)$ and $z \in \mathbf{dom}\, g^*$ with equality only if x and z are optimal, then the function η can serve as a *merit function* in primal–dual algorithms for problem (1).

If $\mathbf{dom}(f+h)$ and $\mathbf{dom}\, g^*$ are bounded, then X and Z can be chosen to contain $\mathbf{dom}(f+h)$ and $\mathbf{dom}\, g^*$. Then the constraints in (6) are redundant and $\eta(x, z)$ is the duality gap (5). Boundedness of $\mathbf{dom}(f + h)$ and $\mathbf{dom}\, g^*$ is a common assumption in the literature on primal–dual first-order methods.

A weaker assumption is that (1) and (2) have an optimal primal solution $x^\star \in X$ and an optimal dual solution $z^\star \in Z$. If that is the case,

$$\eta(x, z) \geq \mathcal{L}(x, z^\star) - \mathcal{L}(x^\star, z) \geq 0$$

for all $x \in \mathbf{dom}(f+h)$ and $z \in \mathbf{dom}\, g^*$. The second inequality follows from (4) and is an equality only if x and z are optimal. It follows that $\eta(x,z) \geq 0$ with equality only if x, z are optimal.

Whether $\eta(x,z)$ is easy to evaluate depends on the problem and choice of sets X and Z. A general expression for the first term in (6) is

$$\begin{aligned}\sup_{z' \in Z} \mathcal{L}(x,z') &= f(x) + h(x) + (g \,\square\, \sigma_Z)(Ax) \\ &= f(x) + h(x) + \inf_y \left(g(y) + \sigma_Z(Ax - y)\right),\end{aligned}$$

where $\sigma_Z(w) = \sup_{z \in Z} \langle z, w \rangle$ is the support function of Z. The corresponding expression for the second term in (6) are

$$\begin{aligned}\inf_{x' \in X} \mathcal{L}(x',z) &= -g^*(z) - ((f+h)^* \,\square\, \sigma_X)(-A^T z) \\ &= -g^*(z) - \inf_w \left((f+h)^*(w) + \sigma_X(A^T z + w)\right).\end{aligned}$$

Consider for example the primal and dual pair

$$\begin{array}{ll} \text{minimize} & f(x) + h(x) \\ \text{subject to} & Ax = b \end{array} \qquad\qquad \text{maximize} \quad -b^T z - (f+h)^*(-A^T z).$$

Here $g = \delta_{\{b\}}$. If we take $Z = \{z \mid \|z\| \leq \gamma\}$, then $\sigma_Z(y) = \gamma\|y\|$ and $(g \,\square\, \sigma_Z)(Ax) = \gamma\|Ax - b\|$. If in addition $\mathbf{dom}(f+h)$ is bounded and we take $X \supseteq \mathbf{dom}(f+h)$, then

$$\eta(x,z) = f(x) + h(x) + \gamma\|Ax - b\| + b^T z + (f+h)^*(-A^T z)$$

with domain $\mathbf{dom}(f+h) \times \mathbf{R}^m$. The first three terms are the primal objective augmented with an exact penalty for the constraint $Ax = b$.

As another example, consider

$$\begin{array}{ll} \text{minimize} & \|x\|_1 \\ \text{subject to} & Ax \leq b \end{array} \qquad\qquad \begin{array}{ll} \text{maximize} & b^T z \\ \text{subject to} & \|A^T z\|_\infty \leq 1 \\ & z \geq 0. \end{array}$$

This is an example of (1) with $f(x) = \|x\|_1$, $h(x) = 0$, and g the indicator function of $\{y \mid y \leq b\}$. The domains $\mathbf{dom}\, f$ and $\mathbf{dom}\, g^*$ are unbounded. If we choose $X = \{x \mid \|x\|_\infty \leq \kappa\}$ and $Z = \{z \mid 0 \leq z \leq \lambda \mathbf{1}\}$, then

$$\sigma_X(w) = \kappa\|w\|_1, \qquad (f^* \,\square\, \sigma_X)(w) = \kappa \sum_{i=1}^n \max\{0, |w_i| - 1\}$$

and

$$\sigma_Z(y) = \lambda \sum_{i=1}^m \max\{0, y_i\}, \qquad (g \,\square\, \sigma_Z)(y) = \lambda \sum_{i=1}^m \max\{0, y_i - b_i\}.$$

Hence, for this example the merit function (6) is

$$\eta(x,z) = \|x\|_1 + \lambda \sum_{i=1}^m \max\{0, (Ax - b)_i\} - b^T z + \kappa \sum_{i=1}^n \max\{0, |(A^T z)_i| - 1\}$$

with domain $\mathbf{R}^n \times \mathbf{R}^m_+$. The second term is an exact penalty for the primal constraint $Ax \leq b$. The last term is an exact penalty for the dual constraint $\|A^T z\|_\infty \leq 1$.

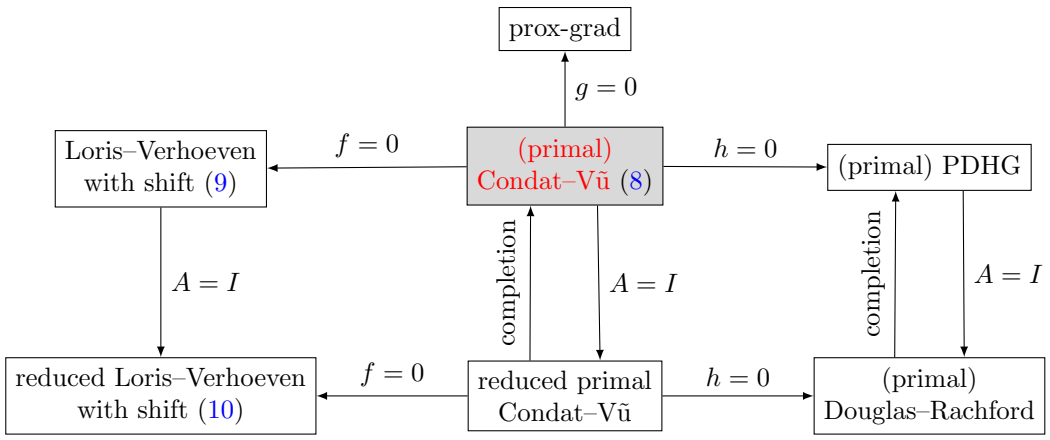

Figure 1: Proximal methods derived from primal Condat–Vũ algorithm.

3 First-order proximal algorithms: survey and connections

In this section, we discuss several first-order proximal algorithms and their connections. We start with four three-operator splitting algorithms for problem (1): the primal and dual variants of the Condat–Vũ algorithm [19, 49], the primal–dual three-operator (PD3O) algorithm [50], and the primal–dual Davis–Yin (PDDY) algorithm [44]. For each of the four algorithms, we make connections with other first-order proximal algorithms, using reduction (*i.e.*, setting some parts in (1) to zero) and the "completion" reformulation (based on extending A to a matrix with orthogonal rows and equal row norms) [38]. We focus on the formal connections between algorithms. The connections do not necessarily provide the best approach for convergence analysis or the best known convergence results.

The *proximal operator* or *proximal mapping* of a closed convex function $f\colon \mathbf{R}^n \to \mathbf{R}$ is defined as

$$\operatorname{prox}_f(y) = \operatorname*{argmin}_x \big(f(x) + \frac{1}{2}\|x - y\|^2\big). \tag{7}$$

If f is closed and convex, the minimizer in the definition exists and is unique for all y [36]. We will call (7) the *standard* or the *Euclidean proximal operator* when we need to distinguish it from Bregman proximal operators defined in Section 4.

3.1 Condat–Vũ three-operator splitting algorithm

We start with the (primal) Condat–Vũ three-operator splitting algorithm, which was proposed independently by Condat [19] and Vũ [49],

$$x^{(k+1)} = \operatorname{prox}_{\tau f}\big(x^{(k)} - \tau(A^T z^{(k)} + \nabla h(x^{(k)}))\big) \tag{8a}$$

$$z^{(k+1)} = \operatorname{prox}_{\sigma g^*}\big(z^{(k)} + \sigma A(2x^{(k+1)} - x^{(k)})\big). \tag{8b}$$

The stepsizes σ and τ must satisfy

$$\sigma\tau\|A\|_2^2 + \tau L \leq 1,$$

where $\|A\|_2$ is the spectral norm of A, and L is the Lipschitz constant of ∇h with respect to the Euclidean norm. Many other first-order proximal algorithms can be viewed as special cases of (8),

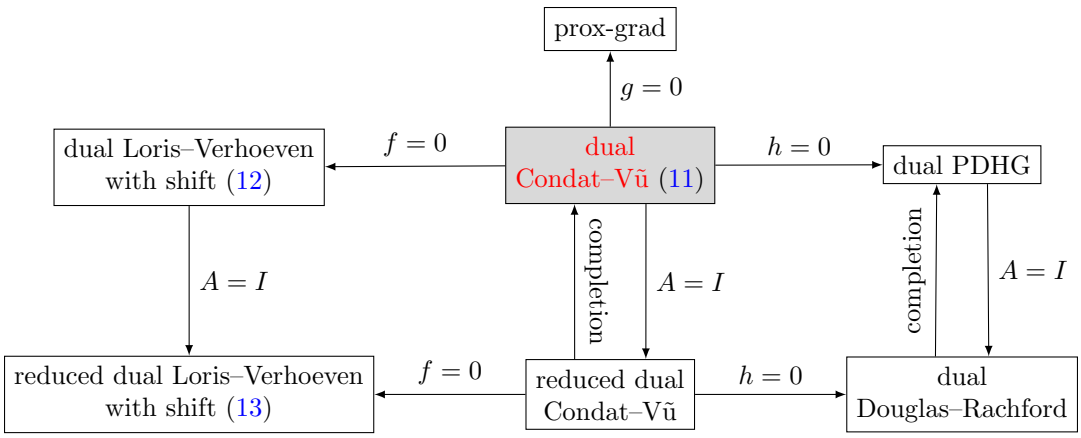

Figure 2: Proximal methods derived from dual Condat–Vũ algorithm.

and their connections are summarized in Figure 1. When $h = 0$, algorithm (8) reduces to the (primal) primal–dual hybrid gradient (PDHG) method [42, 10, 12], or PDHGMu in [25]. When $g = 0$ in (8) (and assuming $z^{(0)} = 0$), we obtain the proximal gradient algorithm. When $f = 0$, we obtain a variant of the Loris–Verhoeven algorithm [33, 15, 22],

$$x^{(k+1)} = x^{(k)} - \tau(A^T z^{(k)} + \nabla h(x^{(k)})) \tag{9a}$$

$$z^{(k+1)} = \operatorname{prox}_{\sigma g^*}\big((I - \sigma\tau AA^T)z^{(k)} + \sigma A(x^{(k+1)} - \tau\nabla h(x^{(k)})\big). \tag{9b}$$

We refer to this as *Loris–Verhoeven with shift*, for reasons that will be clarified later. Furthermore, when $A = I$ in PDHG, we obtain the Douglas–Rachford splitting (DRS) algorithm [31, 24, 17]. Conversely, the "completion" technique in [38] shows that PDHG coincides with DRS applied to a reformulation of the problem. Similarly, when $A = I$ in the primal Condat–Vũ algorithm (8), we obtain a new algorithm and refer to it as the *reduced primal Condat–Vũ algorithm*. Conversely, the reduced primal Condat–Vũ algorithm reverts to (8) via the "completion" trick. We can also set $f = 0$ in the reduced Condat–Vũ algorithm or $A = I$ in (9), and obtain the *reduced Loris–Verhoeven algorithm with shift*:

$$x^{(k+1)} = x^{(k)} - \tau(z^{(k)} + \nabla h(x^{(k)})) \tag{10a}$$

$$z^{(k+1)} = \operatorname{prox}_{\sigma g^*}\big((1 - \sigma\tau)z^{(k)} + \sigma(x^{(k+1)} - \tau\nabla h(x^{(k)})\big). \tag{10b}$$

Finally, due to the absence of f in (10), it is not clear how to apply the "completion" trick to (10) to obtain (9).

Condat [19] also discusses a variant of (8), which we will call the dual Condat–Vũ algorithm:

$$z^{(k+1)} = \operatorname{prox}_{\sigma g^*}(z^{(k)} + \sigma A x^{(k)}) \tag{11a}$$

$$x^{(k+1)} = \operatorname{prox}_{\tau f}\big(x^{(k)} - \tau(A^T(2z^{(k+1)} - z^{(k)}) + \nabla h(x^{(k)}))\big). \tag{11b}$$

Figure 2 summarizes the proximal algorithms derived from (11). When $h = 0$, algorithm (11) reduces to PDHG applied to the dual of (1) (with $h = 0$), which is shown to be equivalent to linearized ADMM [39, 40] (also called Split Inexact Uzawa in [25]). Setting $g = 0$ in (11) yields the proximal gradient algorithm. When $f = 0$, we obtain a new algorithm:

$$z^{(k+1)} = \operatorname{prox}_{\sigma g^*}(z^{(k)} + \sigma A x^{(k)}) \tag{12a}$$

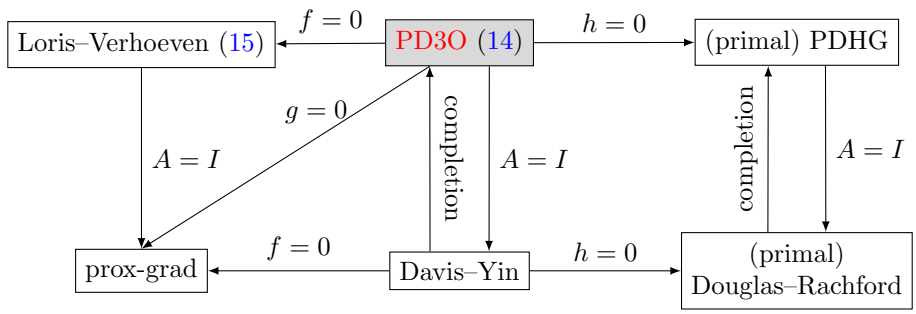

Figure 3: Proximal algorithms derived from PD3O.

$$x^{(k+1)} = x^{(k)} - \tau(A^T(2z^{(k+1)} - z^{(k)}) + \nabla h(x^{(k)})). \tag{12b}$$

Following the previous naming convention, we call it *dual Loris–Verhoeven algorithm with shift*. Furthermore, setting $A = I$ in (11) gives the *reduced dual Condat–Vũ algorithm*. Conversely, applying the "completion" trick to this reduced algorithm recovers (11). Similarly, setting $A = I$ in dual PDHG gives dual DRS, *i.e.*, DRS with f and g switched, and conversely, the "completion" trick recovers dual PDHG from dual DRS. We can also set $A = I$ in (12) or $f = 0$ in the reduced dual Condat–Vũ algorithm, and obtain the *reduced dual Loris–Verhoeven algorithm with shift*:

$$z^{(k+1)} = \operatorname{prox}_{\sigma g^*}(z^{(k)} + \sigma x^{(k)}) \tag{13a}$$
$$x^{(k+1)} = x^{(k)} - \tau(2z^{(k+1)} - z^{(k)} + \nabla h(x^{(k)})). \tag{13b}$$

3.2 Primal–dual three-operator (PD3O) splitting algorithm

The third diagram, Figure 3, starts with the primal–dual three-operator (PD3O) splitting algorithm [50]

$$x^{(k+1)} = \operatorname{prox}_{\tau f}(x^{(k)} - \tau(A^T z^{(k)} + \nabla h(x^{(k)}))) \tag{14a}$$
$$z^{(k+1)} = \operatorname{prox}_{\sigma g^*}(z^{(k)} + \sigma A(2x^{(k+1)} - x^{(k)} + \tau\nabla h(x^{(k)}) - \tau\nabla h(x^{(k+1)}))). \tag{14b}$$

Compared with the Condat–Vũ algorithm (8), PD3O seems to have slightly more complicated updates and larger complexity per iteration, but the requirement for the stepsizes is looser: $\sigma\tau\|A\|_2^2 \leq 1$ and $\tau \leq 1/L$. When $h = 0$, (14) reduces to the (primal) PDHG. The classical proximal gradient algorithm can be obtained by setting $g = 0$. When $f = 0$, it reduces to the iterations

$$x^{(k+1)} = x^{(k)} - \tau(A^T z^{(k)} + \nabla h(x^{(k)})) \tag{15a}$$
$$z^{(k+1)} = \operatorname{prox}_{\sigma g^*}\big((I - \sigma\tau AA^T)z^{(k)} + \sigma A(x^{(k+1)} - \tau\nabla h(x^{(k+1)}))\big). \tag{15b}$$

This algorithm was discovered independently as the Loris–Verhoeven algorithm [33], the primal–dual fixed point algorithm based on proximity operator (PDFP^{2}O) [15], and the proximal alternating predictor corrector (PAPC) [22]. Comparison with (9) reveals a minor difference between these two algorithms: the gradient term in the z-update is taken at the newest primal iterate $x^{(k+1)}$ in Loris–Verhoeven (15) and at the previous point $x^{(k)}$ in the shifted version. This difference is inherited in the proximal gradient algorithm and its shifted version (10).

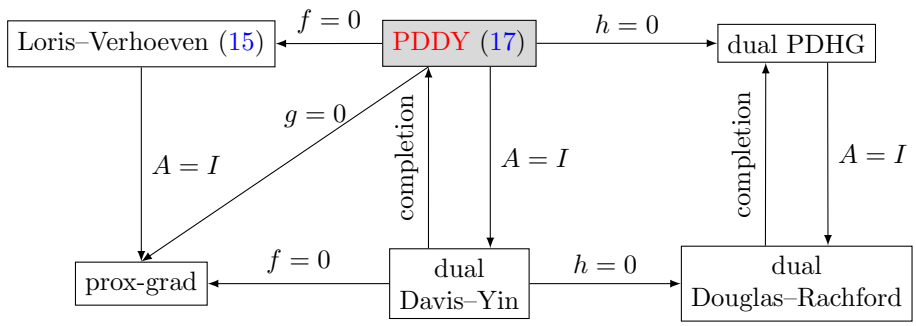

Figure 4: Proximal algorithms derived from PDDY.

Furthermore, when $A = I$ and $\sigma = 1/\tau$ in PD3O, we recover the well-known Davis–Yin splitting (DYS) algorithm [20]. We can also set $A = I$ in (15) and obtain the iterations

$$x^{(k+1)} = x^{(k)} - \tau(z^{(k)} + \nabla h(x^{(k)})) \tag{16a}$$

$$z^{(k+1)} = \operatorname{prox}_{\sigma g^*}\big((1-\sigma\tau)z^{(k)} + \sigma(x^{(k+1)} - \tau\nabla h(x^{(k+1)})\big). \tag{16b}$$

The stepsize conditions require $\sigma\tau \le 1$ and $\tau \le 1/L$. Thus we can set $\sigma = 1/\tau$ and apply Moreau decomposition. The resulting algorithm is exactly the proximal gradient method. The only difference in the z-update between (10) and (16) is the point at which the gradient of h is taken. The second algorithm uses the most up-to-date iterate $x^{(k+1)}$ when evaluating the gradient of h, and this choice allows a larger stepsize τ.

3.3 Primal–dual Davis–Yin (PDDY) splitting algorithm

The core algorithm in Figure 4 is the primal–dual Davis–Yin (PDDY) splitting algorithm [44]

$$z^{(k+1)} = \operatorname{prox}_{\sigma g^*}(z^{(k)} + \sigma A x^{(k)}) \tag{17a}$$

$$x^{(k+1)} = \operatorname{prox}_{\tau f}\big(x^{(k)} - \tau A^T(2z^{(k+1)} - z^{(k)}) - \tau\nabla h(x^{(k)} + \tau A^T(z^{(k)} - z^{(k+1)}))\big). \tag{17b}$$

The requirement for stepsizes is the same as that in PD3O: $\sigma\tau\|A\|_2^2 \le 1$ and $\tau \le 1/L$. Figure 4 is almost identical to Figure 3 with the roles of f and g exchanged. When $h = 0$, PDDY reduces to the dual PDHG. In addition, when $A = I$ and $\sigma = 1/\tau$, PDDY reduces to the Davis–Yin algorithm, but with f and g exchanged. Similarly, when $h = 0$, $A = I$ and $\sigma = 1/\tau$, PDDY reverts to the Douglas–Rachford algorithm with f and g switched.

We have seen that the middle and right parts of Figure 4 are those of Figure 3 with f and g switched. However, when one of the functions f or g is absent, the algorithms reduced from PD3O and PDDY are exactly the same. In particular, when $f = 0$, PDDY reduces to the Loris–Verhoeven algorithm.

4 Bregman distances

In this section we give the definition of Bregman proximal operators and the basic properties that will be used in the paper. We refer the interested reader to [9] for an in-depth discussion of Bregman distances, their history, and applications.

Let ϕ be a convex function with a domain that has nonempty interior, and assume ϕ is continuous on $\mathbf{dom}\,\phi$ and continuously differentiable on $\mathbf{int}(\mathbf{dom}\,\phi)$. The *generalized distance* (or *Bregman distance*) generated by the *kernel function* ϕ is defined as the function

$$d(x,y) = \phi(x) - \phi(y) - \langle \nabla\phi(y), x-y\rangle,$$

with domain $\mathbf{dom}\,d = \mathbf{dom}\,\phi \times \mathbf{int}(\mathbf{dom}\,\phi)$. The corresponding *Bregman proximal operator* of a function f is

$$\begin{aligned}\mathrm{prox}_f^{\phi}(y,a) &= \operatorname*{argmin}_x \left(f(x) + \langle a, x\rangle + d(x,y)\right) &(18)\\ &= \operatorname*{argmin}_x \left(f(x) + \langle a, x\rangle + \phi(x) - \langle \nabla\phi(y), x\rangle\right). &(19)\end{aligned}$$

It is assumed that for every a and every $y \in \mathbf{int}(\mathbf{dom}\,\phi)$ the minimizer $\hat{x} = \mathrm{prox}_f^{\phi}(y,a)$ is unique and in $\mathbf{int}(\mathbf{dom}\,\phi)$.

The distance generated by the kernel $\phi(x) = (1/2)\|x\|^2$ is the squared Euclidean distance $d(x,y) = (1/2)\|x-y\|^2$. The corresponding Bregman proximal operator is the standard proximal operator applied to $y-a$:

$$\mathrm{prox}_f^{\phi}(y,a) = \mathrm{prox}_f(y-a).$$

For this distance, closedness and convexity of f guarantee that the proximal operator is well defined. The questions of existence and uniqueness are more complicated for general Bregman distances. There are no simple general conditions that guarantee that for every a and every $y \in \mathbf{int}(\mathbf{dom}\,\phi)$ the generalized proximal operator (18) is uniquely defined and in $\mathbf{int}(\mathbf{dom}\,\phi)$. Some sufficient conditions are provided (see, for example, [8, Section 4.1], [3, Assumption A]), but they may be quite restrictive or difficult to verify in practice. In applications, however, the Bregman proximal operator is used with specific combinations of f and ϕ, for which the minimization problem in (18) is particularly easy to solve. In those applications, existence and uniqueness of the solution follow directly from the closed-form solution or availability of a fast algorithm to compute it. A typical example will be provided in Section 8.

From the expression (19) we see that $\hat{x} = \mathrm{prox}_f^{\phi}(y,a)$ satisfies

$$\nabla\phi(y) - \nabla\phi(\hat{x}) - a \in \partial f(\hat{x}).$$

Equivalently, by definition of subgradient,

$$\begin{aligned}f(x) + \langle a, x\rangle &\geq f(\hat{x}) + \langle a, \hat{x}\rangle + \langle \nabla\phi(y) - \nabla\phi(\hat{x}), x-\hat{x}\rangle\\ &= f(\hat{x}) + \langle a, \hat{x}\rangle + d(\hat{x},y) + d(x,\hat{x}) - d(x,y) &(20)\end{aligned}$$

for all $x \in \mathbf{dom}\,f \cap \mathbf{dom}\,\phi$.

5 Bregman Condat–Vũ three-operator splitting algorithms

We now discuss two Bregman three-operator splitting algorithms for the problem (1). The algorithms use a generalized distance d_{p} in the primal space, generated by a kernel ϕ_{p}, and a generalized distance d_{d} in the dual space, generated by a kernel ϕ_{d}. The first algorithm is

$$x^{(k+1)} = \mathrm{prox}_{\tau f}^{\phi_{\mathrm{p}}}\big(x^{(k)}, \tau A^T z^{(k)} + \tau\nabla h(x^{(k)})\big) \tag{21a}$$

$$z^{(k+1)} = \operatorname{prox}_{\sigma g^*}^{\phi_{\rm d}}\big(z^{(k)}, -\sigma A(2x^{(k+1)} - x^{(k)})\big) \tag{21b}$$

and will be referred to as the *Bregman primal Condat–Vũ algorithm*. The second algorithm will be called the *Bregman dual Condat–Vũ algorithm*:

$$z^{(k+1)} = \operatorname{prox}_{\sigma g^*}^{\phi_{\rm d}}(z^{(k)}, -\sigma A x^{(k)}) \tag{22a}$$

$$x^{(k+1)} = \operatorname{prox}_{\tau f}^{\phi_{\rm p}}(x^{(k)}, \tau A^T(2z^{(k+1)} - z^{(k)}) + \tau \nabla h(x^{(k)})). \tag{22b}$$

The two algorithms need starting points $x^{(0)} \in \mathbf{int}(\mathbf{dom}\,\phi_{\rm p}) \cap \mathbf{dom}\,h$, and $z^{(0)} \in \mathbf{int}(\mathbf{dom}\,\phi_{\rm d})$. Conditions on stepsizes σ, τ will be specified later. When Euclidean distances are used for the primal and dual proximal operators, the two algorithms reduce to the primal and dual variants of the Condat–Vũ algorithm (8) and (11), respectively. Algorithm (21) has been proposed in [11]. Here we discuss it together with (21) in a unified framework.

In Section 5.1 we show that the proposed algorithms can be interpreted as the Bregman proximal point method applied to a monotone inclusion problem. In Section 5.2 we analyze their convergence. In Section 5.3 we discuss the connections between the two algorithms and other Bregman proximal splitting methods.

Assumptions Throughout Section 5 we make the following assumptions. The kernel functions $\phi_{\rm p}$ and $\phi_{\rm d}$ are 1-strongly convex with respect to norms $\|\cdot\|_{\rm p}$ and $\|\cdot\|_{\rm d}$, respectively:

$$d_{\rm p}(x, x') \geq \frac{1}{2}\|x - x'\|_{\rm p}^2, \qquad d_{\rm d}(z, z') \geq \frac{1}{2}\|z - z'\|_{\rm d}^2 \tag{23}$$

for all $(x, x') \in \mathbf{dom}\,d_{\rm p}$ and $(z, z') \in \mathbf{dom}\,d_{\rm d}$. The assumption that the strong convexity constants are equal to one can be made without loss of generality, by scaling the norms (or distances) if needed. We also assume that the function $L\phi_{\rm p} - h$ is convex for some $L > 0$. More precisely, $\mathbf{dom}\,\phi_{\rm p} \subseteq \mathbf{dom}\,h$ and

$$h(x) - h(x') - \langle \nabla h(x'), x - x'\rangle \leq L d_{\rm p}(x, x') \quad \text{for all } (x, x') \in \mathbf{dom}\,d_{\rm p}. \tag{24}$$

Note that this assumption is looser than the one in [11, Equation (4)]. We denote by $\|A\|$ the matrix norm

$$\|A\| = \sup_{u\neq 0, v\neq 0} \frac{\langle v, Au\rangle}{\|v\|_{\rm d}\|u\|_{\rm p}} = \sup_{u\neq 0} \frac{\|Au\|_{\rm d,*}}{\|u\|_{\rm p}} = \sup_{v\neq 0} \frac{\|A^T v\|_{\rm p,*}}{\|v\|_{\rm d}}, \tag{25}$$

where $\|\cdot\|_{\rm p,*}$ and $\|\cdot\|_{\rm d,*}$ are the dual norms of $\|\cdot\|_{\rm p}$ and $\|\cdot\|_{\rm d}$.

It is also assumed that the primal–dual optimality conditions (3) have a solution $(x^\star, z^\star)$ with $x^\star \in \mathbf{dom}\,\phi_{\rm p}$ and $z^\star \in \mathbf{dom}\,\phi_{\rm d}$.

5.1 Derivation from Bregman proximal point method

The Bregman Condat–Vũ algorithms (21) and (22) can be viewed as applications of the Bregman proximal point algorithm to the optimality conditions (3). This interpretation extends the derivation of the Bregman PDHG algorithm from the Bregman proximal point algorithm given in [29]. The idea originates with He and Yuan's interpretation of PDHG as a "preconditioned" proximal point algorithm [27].

The Bregman proximal point algorithm [23, 9, 26] is an algorithm for monotone inclusion problems $0 \in F(u)$. The update $u^{(k+1)}$ in one iteration of the algorithm is defined as the solution of the inclusion

$$\nabla\phi(u^{(k)}) - \nabla\phi(u^{(k+1)}) \in F(u^{(k+1)}),$$

where ϕ is a Bregman kernel function. Applied to (3), with a kernel function $\phi_{\rm pd}$, the algorithm generates a sequence $(x^{(k)}, z^{(k)})$ defined by

$$\nabla\phi_{\rm pd}(x^{(k)}, z^{(k)}) - \nabla\phi_{\rm pd}(x^{(k+1)}, z^{(k+1)}) \in \begin{bmatrix} A^T z^{(k+1)} + \partial f(x^{(k+1)}) + \nabla h(x^{(k+1)}) \\ -Ax^{(k+1)} + \partial g^*(z^{(k+1)}) \end{bmatrix}. \tag{26}$$

5.1.1 Primal–dual Bregman distances

We introduce four possible primal–dual kernel functions: the functions

$$\phi_+(x,z) = \frac{1}{\tau}\phi_{\rm p}(x) + \frac{1}{\sigma}\phi_{\rm d}(z) + \langle z, Ax\rangle, \qquad \phi_-(x,z) = \frac{1}{\tau}\phi_{\rm p}(x) + \frac{1}{\sigma}\phi_{\rm d}(z) - \langle z, Ax\rangle,$$

where $\sigma, \tau > 0$, and the functions

$$\phi_{\rm dcv}(x,z) = \phi_+(x,z) - h(x), \qquad \phi_{\rm pcv}(x,z) = \phi_-(x,z) - h(x).$$

The subscripts in ϕ_+ and ϕ_- refer to the sign of the inner product term $\langle z, Ax\rangle$. The subscripts in $\phi_{\rm pcv}$ and $\phi_{\rm dcv}$ indicate the algorithm (Bregman primal or dual Condat-Vũ) for which these distances will be relevant. If these kernel functions are convex, they generate the following Bregman distances. The distances generated by ϕ_+ and ϕ_- are

$$\begin{aligned} d_+(x,z;x',z') &= \frac{1}{\tau}d_{\rm p}(x,x') + \frac{1}{\sigma}d_{\rm d}(z,z') + \langle z-z', A(x-x')\rangle \\ d_-(x,z;x',z') &= \frac{1}{\tau}d_{\rm p}(x,x') + \frac{1}{\sigma}d_{\rm d}(z,z') - \langle z-z', A(x-x')\rangle, \end{aligned}$$

respectively, and the distances generated by $\phi_{\rm dcv}$ and $\phi_{\rm pcv}$ are

$$\begin{aligned} d_{\rm dcv}(x,z;x',z') &= d_+(x,z;x',z') - h(x) + h(x') + \langle \nabla h(x'), x-x'\rangle \\ d_{\rm pcv}(x,z;x',z') &= d_-(x,z;x',z') - h(x) + h(x') + \langle \nabla h(x'), x-x'\rangle. \end{aligned}$$

We now show that ϕ_+ and ϕ_- are convex if

$$\sigma\tau\|A\|^2 \le 1$$

and strongly convex if $\sigma\tau\|A\|^2 < 1$, and that the functions $\phi_{\rm dcv}$ and $\phi_{\rm pcv}$ are convex if

$$\sigma\tau\|A\|^2 + \tau L \le 1 \tag{27}$$

and strongly convex if $\sigma\tau\|A\|^2 + \tau L < 1$.

Proof. To show that the kernel functions ϕ_+ and ϕ_- are convex, we show that d_+ and d_- are nonnegative. Suppose $\sigma\tau\|A\|^2 \le \delta^2$ with $0 < \delta \le 1$. Then (23) and the arithmetic–geometric mean inequality imply that

$$\big|\langle z-z', A(x-x')\rangle\big| \le \|A\|\|z-z'\|_{\rm d}\|x-x'\|_{\rm p}$$

$$
\begin{aligned}
&\le \frac{\delta}{\sqrt{\sigma\tau}}\|z-z'\|_{\mathrm{d}}\|x-x'\|_{\mathrm{p}} \\
&\le \frac{\delta}{2\tau}\|x-x'\|_{\mathrm{p}}^2 + \frac{\delta}{2\sigma}\|z-z'\|_{\mathrm{p}}^2 \\
&\le \frac{\delta}{\tau}d_{\mathrm{p}}(x,x') + \frac{\delta}{\sigma}d_{\mathrm{d}}(z,z'). \qquad (28)
\end{aligned}
$$

Therefore

$$
\begin{aligned}
d_{\pm}(x,z;x',z') &= \frac{1}{\tau}d_{\mathrm{p}}(x,x') + \frac{1}{\sigma}d_{\mathrm{d}}(z,z') \pm \langle z-z', A(x-x')\rangle \\
&\ge \frac{1-\delta}{\tau}d_{\mathrm{p}}(x,x') + \frac{1-\delta}{\sigma}d_{\mathrm{d}}(z,z') \\
&\ge \frac{1-\delta}{2\tau}\|x-x'\|_{\mathrm{p}}^2 + \frac{1-\delta}{2\sigma}\|z-z'\|_{\mathrm{d}}^2.
\end{aligned}
$$

With $\delta = 1$, this shows convexity of ϕ_+ and ϕ_-; with $\delta < 1$, strong convexity.

Similarly, if $\sigma\tau\|A\|^2 \le \delta(\delta - \tau L)$, with $0 < \delta \le 1$, then

$$
\begin{aligned}
\big|\langle z-z', A(x-x')\rangle\big| &\le \frac{\sqrt{\delta(\delta-\tau L)}}{\sqrt{\sigma\tau}}\|z-z'\|_{\mathrm{d}}\|x-x'\|_{\mathrm{p}} \\
&\le \frac{\delta-\tau L}{2\tau}\|x-x'\|_{\mathrm{p}}^2 + \frac{\delta}{2\sigma}\|z-z'\|_{\mathrm{d}}^2 \\
&\le \frac{\delta-\tau L}{\tau}d_{\mathrm{p}}(x,x') + \frac{\delta}{\sigma}d_{\mathrm{d}}(z,z')
\end{aligned}
$$

and

$$
\begin{aligned}
d_{\mathrm{dcv/pcv}}(x,z;x',z') &= \frac{1}{\tau}d_{\mathrm{p}}(x,x') + \frac{1}{\sigma}d_{\mathrm{d}}(z,z') \pm \langle z-z', A(x-x')\rangle \\
&\quad - h(x) + h(x') + \langle \nabla h(x'), x-x'\rangle \\
&\ge (\frac{1-\delta}{\tau} + L)d_{\mathrm{p}}(x,x') + \frac{1-\delta}{\sigma}d_{\mathrm{d}}(z,z') - h(x) + h(x') + \langle \nabla h(x'), x-x'\rangle \\
&\ge \frac{1-\delta}{\tau}d_{\mathrm{p}}(x,x') + \frac{1-\delta}{\sigma}d_{\mathrm{d}}(z,z'). \qquad \square
\end{aligned}
$$

5.1.2 Bregman Condat-Vũ algorithms from proximal point method

The Bregman primal Condat–Vũ algorithm (21) is the Bregman proximal point method with the kernel function $\phi_{\mathrm{pd}} = \phi_{\mathrm{pcv}}$. If we take $\phi_{\mathrm{pd}} = \phi_{\mathrm{pcv}}$ in (26), we obtain two coupled inclusions that determine $x^{(k+1)}$, $z^{(k+1)}$. The first one is

$$
\begin{aligned}
0 \;\in\; & \frac{1}{\tau}(\nabla\phi_{\mathrm{p}}(x^{(k+1)}) - \nabla\phi_{\mathrm{p}}(x^{(k)})) - A^T(z^{(k+1)} - z^{(k)}) - \nabla h(x^{(k+1)}) + \nabla h(x^{(k)}) \\
& + A^T z^{(k+1)} + \partial f(x^{(k+1)}) + \nabla h(x^{(k+1)}) \\
= \; & \frac{1}{\tau}(\nabla\phi_{\mathrm{p}}(x^{(k+1)}) - \nabla\phi_{\mathrm{p}}(x^{(k)})) + A^T z^{(k)} + \nabla h(x^{(k)}) + \partial f(x^{(k+1)}).
\end{aligned}
$$

This shows that $x^{(k+1)}$ solves the optimization problem

$$
\text{minimize} \quad f(x) + \langle A^T z^{(k)} + \nabla h(x^{(k)}), x\rangle + \frac{1}{\tau}d_{\mathrm{p}}(x, x^{(k)}).
$$

The solution is the x-update (21a) in the Bregman primal Condat–Vũ method. The second inclusion is

$$\begin{aligned} 0 &\in \frac{1}{\sigma}(\nabla\phi_{\rm d}(z^{(k+1)}) - \nabla\phi_{\rm d}(z^{(k)})) - A(x^{(k+1)} - x^{(k)}) - Ax^{(k+1)} + \partial g^*(z^{(k+1)}) \\ &= \frac{1}{\sigma}(\nabla\phi_{\rm d}(z^{(k+1)}) - \nabla\phi_{\rm d}(z^{(k)})) - A(2x^{(k+1)} - x^{(k)}) + \partial g^*(z^{(k+1)}). \end{aligned}$$

This shows that $z^{(k+1)}$ solves the optimization problem

$$\text{minimize} \quad g^*(z) - \langle z, A(2x^{(k+1)} - x^{(k)})\rangle + \frac{1}{\sigma}d_{\rm d}(z, z^{(k)}).$$

The solution is the z-update (21b).

Choosing $\phi_{\rm pd} = \phi_{\rm dcv}$ in (26) yields the Bregman dual Condat–Vũ algorithm (22). Substituting $\phi_{\rm pd} = \phi_{\rm dcv}$ in (26) gives the inclusions

$$\begin{aligned} 0 &\in \frac{1}{\tau}(\nabla\phi_{\rm p}(x^{(k+1)}) - \nabla\phi_{\rm p}(x^{(k)})) + A^T(z^{(k+1)} - z^{(k)}) - \nabla h(x^{(k+1)}) + \nabla h(x^{(k)}) \\ &\quad + A^T z^{(k+1)} + \partial f(x^{(k+1)}) + \nabla h(x^{(k+1)}) \\ &= \frac{1}{\tau}(\nabla\phi_{\rm p}(x^{(k+1)}) - \nabla\phi_{\rm p}(x^{(k)})) + A^T(2z^{(k+1)} - z^{(k)}) + \nabla h(x^{(k)}) + \partial f(x^{(k+1)}) \end{aligned}$$

and

$$\begin{aligned} 0 &\in \frac{1}{\sigma}(\nabla\phi_{\rm d}(z^{(k+1)}) - \nabla\phi_{\rm d}(z^{(k)})) + A(x^{(k+1)} - x^{(k)}) - Ax^{(k+1)} + \partial g^*(z^{(k+1)}) \\ &= \frac{1}{\sigma}(\nabla\phi_{\rm d}(z^{(k+1)}) - \nabla\phi_{\rm d}(z^{(k)})) - Ax^{(k)} + \partial g^*(z^{(k+1)}). \end{aligned}$$

The second inclusion shows that $z^{(k+1)}$ is given by the z-update (22a). Given $z^{(k+1)}$, one can solve the first inclusion for $x^{(k+1)}$ and obtains the x-update (22b).

5.2 Convergence analysis

The derivation in Section 5.1 allows us to apply existing convergence theory for the Bregman proximal point method for monotone inclusions to the proposed algorithms (21) and (22). The literature on the Bregman proximal point method for monotone inclusions [23, 26, 9] focuses on the convergence of iterates, and this generally requires additional assumptions on $\phi_{\rm p}$ and $\phi_{\rm d}$ (beyond the assumptions of convexity made in Section 5.1). In this section we present a self-contained convergence analysis and give a direct proof of an $O(1/k)$ rate of ergodic convergence. We also give a self-contained proof of convergence of the iterates $x^{(k)}$ and $z^{(k)}$.

We make the assumptions listed in Section 5.1: the strong convexity assumption (23) for the primal and dual kernels $\phi_{\rm p}$ and $\phi_{\rm d}$, and the *relative smoothness* property (24) of the function h. We assume that the stepsizes σ, τ satisfy (27), and that the primal–dual optimality condition (3) has a solution $(x^\star, z^\star) \in \mathbf{dom}\,\phi_{\rm p} \times \mathbf{dom}\,\phi_{\rm d}$.

For the sake of brevity we combine the analysis of the Bregman primal and the Bregman dual Condat-Vũ algorithms. In the following, d, $\tilde{d}$, $\tilde{\phi}$ are defined as

$$d = d_-, \qquad \tilde{d} = d_{\rm pcv}, \qquad \tilde{\phi} = \phi_{\rm pcv}$$

for the Bregman primal Condat–Vũ algorithm (21) and as

$$d = d_+, \qquad \tilde{d} = d_{\mathrm{dcv}}, \qquad \tilde{\phi} = \phi_{\mathrm{dcv}}$$

for the Bregman dual Condat–Vũ algorithm (22).

5.2.1 One-iteration analysis

We first show that the iterates $x^{(k+1)}$, $z^{(k+1)}$ generated by the Bregman Condat–Vũ algorithms (21) and (22) satisfy

$$\begin{aligned} &\mathcal{L}(x^{(k+1)}, z) - \mathcal{L}(x, z^{(k+1)}) \\ &\quad \leq \quad d(x, z; x^{(k)}, z^{(k)}) - d(x, z; x^{(k+1)}, z^{(k+1)}) - \tilde{d}(x^{(k+1)}, z^{(k+1)}; x^{(k)}, z^{(k)}) \end{aligned} \tag{29}$$

for all $x \in \mathbf{dom}\, f \cap \mathbf{dom}\, \phi_{\mathrm{p}}$ and $z \in \mathbf{dom}\, g^* \cap \mathbf{dom}\, \phi_{\mathrm{d}}$.

Proof. We write (21) and (22) in a unified notation as

$$x^{(k+1)} = \mathrm{prox}_{\tau f}^{\phi_{\mathrm{p}}}(x^{(k)}, \tau(A^T \tilde{z} + \nabla h(x^{(k)}))) \tag{30a}$$

$$z^{(k+1)} = \mathrm{prox}_{\sigma g^*}^{\phi_{\mathrm{d}}}(z^{(k)}, -\sigma A \tilde{x}) \tag{30b}$$

where $\tilde{x}$ and $\tilde{z}$ are defined in the following table:

Bregman primal Condat–Vũ algorithm	$\tilde{x} = 2x^{(k+1)} - x^{(k)}$	$\tilde{z} = z^{(k)}$
Bregman dual Condat–Vũ algorithm	$\tilde{x} = x^{(k)}$	$\tilde{z} = 2z^{(k+1)} - z^{(k)}$.

The optimality condition (20) for the proximal operator evaluation (30a) is that

$$\tau(f(x^{(k+1)}) - f(x)) \leq d_{\mathrm{p}}(x, x^{(k)}) - d_{\mathrm{p}}(x^{(k+1)}, x^{(k)}) - d_{\mathrm{p}}(x, x^{(k+1)}) + \tau\langle A^T \tilde{z} + \nabla h(x^{(k)}), x - x^{(k+1)}\rangle$$

for all $x \in \mathbf{dom}\, f \cap \mathbf{dom}\, \phi_{\mathrm{p}}$. The optimality condition for (30b) is that

$$\sigma(g^*(z^{(k+1)}) - g^*(z)) \leq d_{\mathrm{d}}(z, z^{(k)}) - d_{\mathrm{d}}(z^{(k+1)}, z^{(k)}) - d_{\mathrm{d}}(z, z^{(k+1)}) - \sigma\langle z - z^{(k+1)}, A\tilde{x}\rangle$$

for all $z \in \mathbf{dom}\, g^* \cap \mathbf{dom}\, \phi_{\mathrm{d}}$. Combining the two inequalities gives

$$\begin{aligned} &\mathcal{L}(x^{(k+1)}, z) - \mathcal{L}(x, z^{(k+1)}) \\ &\quad = \quad f(x^{(k+1)}) - f(x) + h(x^{(k+1)}) - h(x) + g^*(z^{(k+1)}) - g^*(z) + \langle A^T z, x^{(k+1)}\rangle - \langle z^{(k+1)}, Ax\rangle \\ &\quad \leq \quad \frac{1}{\tau}\Big(d_{\mathrm{p}}(x, x^{(k)}) - d_{\mathrm{p}}(x, x^{(k+1)}) - d_{\mathrm{p}}(x^{(k+1)}, x^{(k)})\Big) \\ &\qquad + \frac{1}{\sigma}\Big(d_{\mathrm{d}}(z, z^{(k)}) - d_{\mathrm{d}}(z, z^{(k+1)}) - d_{\mathrm{d}}(z^{(k+1)}, z^{(k)})\Big) \\ &\qquad + h(x^{(k+1)}) - h(x) + \langle \nabla h(x^{(k)}), x - x^{(k+1)}\rangle \\ &\qquad + \langle A^T \tilde{z}, x - x^{(k+1)}\rangle - \langle z - z^{(k+1)}, A\tilde{x}\rangle + \langle A^T z, x^{(k+1)}\rangle - \langle z^{(k+1)}, Ax\rangle \\ &\quad \leq \quad \frac{1}{\tau}\Big(d_{\mathrm{p}}(x, x^{(k)}) - d_{\mathrm{p}}(x, x^{(k+1)}) - d_{\mathrm{p}}(x^{(k+1)}, x^{(k)})\Big) \\ &\qquad + \frac{1}{\sigma}\Big(d_{\mathrm{d}}(z, z^{(k)}) - d_{\mathrm{d}}(z, z^{(k+1)}) - d_{\mathrm{d}}(z^{(k+1)}, z^{(k)})\Big) \end{aligned}$$

$$
\begin{aligned}
&+ h(x^{(k+1)}) - h(x^{(k)}) + \langle \nabla h(x^{(k)}), x^{(k+1)} - x^{(k)} \rangle \\
&+ \langle A^T \tilde{z}, x - x^{(k+1)} \rangle - \langle z - z^{(k+1)}, A\tilde{x} \rangle + \langle A^T z, x^{(k+1)} \rangle - \langle z^{(k+1)}, Ax \rangle
\end{aligned} \tag{31}
$$

for all $x \in \mathbf{dom}\, f \cap \mathbf{dom}\, \phi_{\mathrm{p}}$ and all $z \in \mathbf{dom}\, g^* \cap \mathbf{dom}\, \phi_{\mathrm{d}}$. The second inequality follows from convexity of h. Substituting the expressions for $\tilde{x}$ and $\tilde{z}$ in the Bregman primal Condat–Vũ algorithm (21), we obtain on the last line of (31)

$$
\begin{aligned}
&\langle A^T \tilde{z}, x - x^{(k+1)} \rangle - \langle z - z^{(k+1)}, A\tilde{x} \rangle + \langle A^T z, x^{(k+1)} \rangle - \langle z^{(k+1)}, Ax \rangle \\
&\quad = \langle z^{(k)}, A(x - x^{(k+1)}) \rangle - \langle z - z^{(k+1)}, A(2x^{(k+1)} - x^{(k)}) \rangle + \langle A^T z, x^{(k+1)} \rangle - \langle z^{(k+1)}, Ax \rangle \\
&\quad = \langle z^{(k)} - z^{(k+1)}, A(x - x^{(k+1)}) \rangle + \langle z - z^{(k+1)}, A(x^{(k)} - x^{(k+1)}) \rangle \\
&\quad = -\langle z - z^{(k)}, A(x - x^{(k)}) \rangle + \langle z - z^{(k+1)}, A(x - x^{(k+1)}) \rangle + \langle z^{(k+1)} - z^{(k)}, A(x^{(k+1)} - x^{(k)}) \rangle.
\end{aligned}
$$

If we substitute the expressions for $\tilde{x}$ and $\tilde{z}$ in the Bregman dual Condat–Vũ algorithm, the last line of (31) becomes

$$
\begin{aligned}
&\langle A^T \tilde{z}, x - x^{(k+1)} \rangle - \langle z - z^{(k+1)}, A\tilde{x} \rangle + \langle A^T z, x^{(k+1)} \rangle - \langle z^{(k+1)}, Ax \rangle \\
&\quad = \langle A^T(z - z^{(k)}), x - x^{(k)} \rangle - \langle A^T(z - z^{(k+1)}), x - x^{(k+1)} \rangle - \langle A^T(z^{(k+1)} - z^{(k)}), x^{(k+1)} - x^{(k)} \rangle.
\end{aligned}
$$

Therefore, for both algorithms, (31) implies that

$$
\begin{aligned}
&\mathcal{L}(x^{(k+1)}, z) - \mathcal{L}(x, z^{(k+1)}) \\
&\quad \leq \frac{1}{\tau} d_{\mathrm{p}}(x, x^{(k)}) + \frac{1}{\sigma} d_{\mathrm{d}}(z, z^{(k)}) \mp \langle z - z^{(k)}, A(x - x^{(k)}) \rangle \\
&\qquad - \Big(\frac{1}{\tau} d_{\mathrm{p}}(x, x^{(k+1)}) + \frac{1}{\sigma} d_{\mathrm{d}}(z, z^{(k+1)}) \mp \langle z - z^{(k+1)}, A(x - x^{(k+1)}) \rangle \Big) \\
&\qquad - \Big(\frac{1}{\tau} d_{\mathrm{p}}(x^{(k+1)}, x^{(k)}) + \frac{1}{\sigma} d_{\mathrm{d}}(z^{(k+1)}, z^{(k)}) \mp \langle z^{(k+1)} - z^{(k)}, A(x^{(k+1)} - x^{(k)}) \rangle \Big) \\
&\qquad + h(x^{(k+1)}) - h(x^{(k)}) - \langle \nabla h(x^{(k)}), x^{(k+1)} - x^{(k)} \rangle,
\end{aligned}
$$

if we select the minus sign in $\mp$ for the Bregman primal Condat–Vũ algorithm, and the plus sign for the Bregman dual Condat–Vũ algorithm. For the primal method, this shows

$$
\begin{aligned}
&\mathcal{L}(x^{(k+1)}, z) - \mathcal{L}(x, z^{(k+1)}) \\
&\quad \leq d_-(x, z; x^{(k)}, z^{(k)}) - d_-(x, z; x^{(k+1)}, z^{(k+1)}) - d_{\mathrm{pcv}}(x^{(k+1)}, z^{(k+1)}; x^{(k)}, z^{(k)}).
\end{aligned}
$$

For the dual method,

$$
\begin{aligned}
&\mathcal{L}(x^{(k+1)}, z) - \mathcal{L}(x, z^{(k+1)}) \\
&\quad \leq d_+(x, z; x^{(k)}, z^{(k)}) - d_+(x, z; x^{(k+1)}, z^{(k+1)}) - d_{\mathrm{dcv}}(x^{(k+1)}, z^{(k+1)}; x^{(k)}, z^{(k)}).
\end{aligned}
$$

□

5.2.2 Ergodic convergence

We define averaged iterates

$$
x_{\mathrm{avg}}^{(k)} = \frac{1}{k} \sum_{i=1}^{k} x^{(i)}, \qquad z_{\mathrm{avg}}^{(k)} = \frac{1}{k} \sum_{i=1}^{k} z^{(i)} \tag{32}
$$

for $k \geq 1$. We show that

$$\mathcal{L}(x_{\text{avg}}^{(k)}, z) - \mathcal{L}(x, z_{\text{avg}}^{(k)}) \leq \frac{2}{k}\Big(\frac{1}{\tau} d_{\text{p}}(x, x^{(0)}) + \frac{1}{\sigma} d_{\text{d}}(z, z^{(0)})\Big) \tag{33}$$

for all $x \in \mathbf{dom}\, f \cap \mathbf{dom}\, \phi_{\text{p}}$ and $z \in \mathbf{dom}\, g^* \cap \mathbf{dom}\, \phi_{\text{d}}$.

Proof. From (29), since $\mathcal{L}(u, v)$ is convex in u and concave in v,

$$\begin{aligned}
\mathcal{L}(x_{\text{avg}}^{(k)}, z) - \mathcal{L}(x, z_{\text{avg}}^{(k)}) &\leq \frac{1}{k}\sum_{i=1}^{k}\big(\mathcal{L}(x^{(i)}, z) - \mathcal{L}(x, z^{(i)})\big) \\
&\leq \frac{1}{k}\big(d(x, z; x^{(0)}, z^{(0)}) - d(x, z; x^{(k)}, z^{(k)})\big) \\
&\leq \frac{1}{k} d(x, z; x^{(0)}, z^{(0)}) \\
&\leq \frac{2}{k}\Big(\frac{1}{\tau} d_{\text{p}}(x, x^{(0)}) + \frac{1}{\sigma} d_{\text{d}}(z, z^{(0)})\Big)
\end{aligned}$$

for all $x \in \mathbf{dom}\, f \cap \mathbf{dom}\, \phi_{\text{p}}$ and $z \in \mathbf{dom}\, g^* \cap \mathbf{dom}\, \phi_{\text{d}}$. The last step follows from (28). □

Substituting $x = x^\star$, $z = z^\star$ in (33) gives

$$\mathcal{L}(x_{\text{avg}}^{(k)}, z^\star) - \mathcal{L}(x^\star, z_{\text{avg}}^{(k)}) \leq \frac{2}{k}\Big(\frac{1}{\tau} d_{\text{p}}(x^\star, x^{(0)}) + \frac{1}{\sigma} d_{\text{d}}(z^\star, z^{(0)})\Big).$$

More generally, if $X \subseteq \mathbf{dom}\, \phi_{\text{p}}$ and $Z \subseteq \mathbf{dom}\, \phi_{\text{d}}$ are compact sets that contain optimal solutions $x^\star$, $z^\star$, respectively, then

$$\eta(x_{\text{avg}}^{(k)}, z_{\text{avg}}^{(k)}) \leq \frac{2}{k}\Big(\frac{1}{\tau}\sup_{x \in X} d_{\text{p}}(x, x^{(0)}) + \frac{1}{\sigma}\sup_{z \in Z} d_{\text{d}}(z, z^{(0)})\Big),$$

where η is the merit function (6).

5.2.3 Monotonicity properties

For $x = x^\star$, $z = z^\star$, the left-hand side of (29) is nonnegative and therefore

$$d(x^\star, z^\star; x^{(k+1)}, z^{(k+1)}) \leq d(x^\star, z^\star; x^{(k)}, z^{(k)}) - \tilde{d}(x^{(k+1)}, z^{(k+1)}; x^{(k)}, z^{(k)}) \tag{34}$$

for $k \geq 0$. Hence $d(x^\star, z^\star; x^{(k+1)}, z^{(k+1)}) \leq d(x^\star, z^\star; x^{(k)}, z^{(k)})$ and

$$d(x^\star, z^\star; x^{(k)}, z^{(k)}) \leq d(x^\star, z^\star; x^{(0)}, z^{(0)}). \tag{35}$$

The inequality (34) also implies that

$$\sum_{i=0}^{k} \tilde{d}(x^{(i+1)}, z^{(i+1)}; x^{(i)}, z^{(i)}) \leq d(x^\star, z^\star; x^{(0)}, z^{(0)}).$$

Hence $\tilde{d}(x^{(k+1)}, z^{(k+1)}; x^{(k)}, z^{(k)}) \to 0$.

5.2.4 Convergence of iterates

Convergence of iterates can be shown under additional assumptions about the primal and dual distance functions. The following two assumptions are common in the literature on Bregman distances [14, 23, 26, 9].

1. For fixed x and z, the sublevel sets $\{x' \mid d_{\rm p}(x, x') \leq \gamma\}$ and $\{z' \mid d_{\rm d}(z, z') \leq \gamma\}$ are closed. In other words, the distances $d_{\rm p}(x, x')$ and $d_{\rm d}(z, z')$ are closed functions of x' and z', respectively. Since a sum of closed functions is closed, the distance $d(x, z; x', z')$ is a closed function of (x', z'), for fixed (x, z).

2. If $\tilde{x}^{(k)} \in \mathbf{int}(\mathbf{dom}\,\phi_{\rm p})$ converges to $x \in \mathbf{dom}\,\phi_{\rm p}$, then $d_{\rm p}(x, \tilde{x}^{(k)}) \to 0$. Similarly, if $\tilde{z}^{(k)} \in \mathbf{int}(\mathbf{dom}\,\phi_{\rm d})$ converges to $z \in \mathbf{dom}\,\phi_{\rm d}$, then $d_{\rm d}(z, \tilde{z}^{(k)}) \to 0$.

We also assume that $\sigma\tau\|A\|^2 + \tau L < 1$. As shown in Section 5.1.1 this implies that the kernel functions $\phi_{\rm pcv}$ and $\phi_{\rm dcv}$ are strongly convex and that

$$\tilde{d}(x, z; x', z') \geq \frac{\alpha}{2\tau}\|x - x'\|_{\rm p}^2 + \frac{\alpha}{2\sigma}\|z - z'\|_{\rm d}^2 \tag{36}$$

for some $\alpha > 0$. Similarly, $\sigma\tau\|A\|^2 < 1$ implies that

$$d(x, z; x', z') \geq \frac{\beta}{2\tau}\|x - x'\|_{\rm p}^2 + \frac{\beta}{2\sigma}\|z - z'\|_{\rm d}^2 \tag{37}$$

for some $\beta > 0$. Recall that $d = d_-$, $\tilde{d} = d_{\rm pcv}$ for the Bregman primal Condat–Vũ algorithm (21), and $d = d_+$, $\tilde{d} = d_{\rm dcv}$ for the Bregman dual Condat–Vũ algorithm.

Proof. We first note that $\tilde{d}(x^{(k+1)}, z^{(k+1)}; x^{(k)}, z^{(k)}) \to 0$ and (36) imply that $x^{(k+1)} - x^{(k)} \to 0$ and $z^{(k+1)} - z^{(k)} \to 0$.

The inequality (35), together with (37), implies that the sequence $(x^{(k)}, z^{(k)})$ is bounded. Let $(x^{(k_i)}, z^{(k_i)})$ be a convergent subsequence of $(x^{(k)}, z^{(k)})$ with limit $(\hat{x}, \hat{z})$. Since $x^{(k_i+1)} - x^{(k_i)} \to 0$ and $z^{(k_i+1)} - z^{(k_i)} \to 0$, the sequence $(x^{(k_i+1)}, z^{(k_i+1)})$ also converges to $(\hat{x}, \hat{z})$. We show that $(\hat{x}, \hat{z})$ satisfies the optimality condition (3).

From (35), $d(x^\star, z^\star; x^{(k_i)}, z^{(k_i)})$ is bounded. Since the sublevel sets $\{(x', z') \mid d(x^\star, z^\star; x', z') \leq \gamma\}$ are closed subsets of $\mathbf{int}(\mathbf{dom}\,\phi_{\rm p}) \cap \mathbf{int}(\mathbf{dom}\,\phi_{\rm d})$, the limit $(\hat{x}, \hat{z}) \in \mathbf{int}(\mathbf{dom}\,\phi_{\rm p}) \cap \mathbf{int}(\mathbf{dom}\,\phi_{\rm d})$. The iterates in the subsequence satisfy

$$\nabla\phi_{\rm pd}(x^{(k_i)}, z^{(k_i)}) - \nabla\phi_{\rm pd}(x^{(k_i+1)}, z^{(k_i+1)}) + \begin{bmatrix} -A^T z^{(k_i+1)} \\ Ax^{(k_i+1)} \end{bmatrix} \in \begin{bmatrix} \partial f(x^{(k_i+1)}) + \nabla h(x^{(k_i+1)}) \\ \partial g^*(z^{(k_i+1)}) \end{bmatrix}, \tag{38}$$

where $\phi_{\rm pd} = \phi_{\rm pcv}$ in the Bregman primal Condat–Vũ algorithm and $\phi_{\rm pd} = \phi_{\rm dcv}$ in the Bregman dual Condat–Vũ algorithm. The left-hand side of (38) converges to $(-A^T\hat{z}, A\hat{x})$ because $\nabla\phi_{\rm pd}$ is continuous on $\mathbf{int}(\mathbf{dom}\,\phi_{\rm pd})$. Since the operator on right-hand side of (38) is maximal monotone the limit point $(\hat{x}, \hat{z})$ satisfies the optimality condition

$$\begin{bmatrix} -A^T\hat{z} \\ A\hat{x} \end{bmatrix} \in \begin{bmatrix} \partial f(\hat{x}) + \nabla h(\hat{x}) \\ \partial g^*(\hat{z}) \end{bmatrix}$$

(see [7, page 27], [46, Lemma 3.2]).

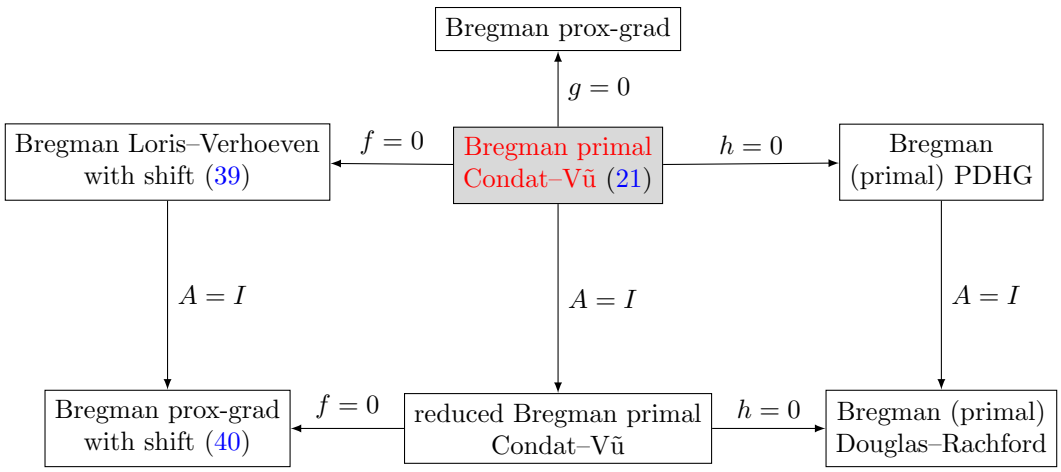

Figure 5: Proximal algorithms derived from Bregman primal Condat–Vũ algorithm (21).

To show convergence of the entire sequence $(x^{(k)}, z^{(k)})$, we substitute $(\hat{x}, \hat{z})$ in (29):

$$\mathcal{L}(x^{(k+1)}, \hat{z}) - \mathcal{L}(\hat{x}, z^{(k+1)}) \leq d(\hat{x}, \hat{z}; x^{(k)}, z^{(k)}) - d(\hat{x}, \hat{z}; x^{(k+1)}, z^{(k+1)}).$$

Since the left-hand side is nonnegative, we have $d(\hat{x}, \hat{z}; x^{(k)}, z^{(k)}) \leq d(\hat{x}, \hat{z}; x^{(k-1)}, z^{(k-1)})$ for all $k \geq 1$. This further implies that

$$d(\hat{x}, \hat{z}; x^{(k)}, z^{(k)}) \leq d(\hat{x}, \hat{z}; x^{(k_i)}, z^{(k_i)})$$

for all $k \geq k_i$. By the second additional assumption mentioned above, the right-hand side converges to zero. Then the left-hand side also converges to zero and, from (37) $x^{(k)} \to \hat{x}$ and $z^{(k)} \to \hat{z}$. $\square$

5.3 Relation to other Bregman proximal splitting algorithms

Following similar steps as in Section 3, we obtain several Bregman proximal splitting methods as special cases of (21) and (22). The connections are summarized in Figure 5 and Figure 6. A comparison of Figures 1 and 5 shows that all the reduction relations ($A = I$) are still valid. However, it is unclear how to apply the "completion" operation to algorithms based on non-Euclidean Bregman distances.

When $h = 0$, (21) reduces to Bregman PDHG [12]. When $g = 0$, $g^* = \delta_{\{0\}}$ (and assuming $z^{(0)} = 0$), we obtain the Bregman proximal gradient algorithm [3]. When $f = 0$ in (21), we obtain the *Bregman Loris–Verhoeven algorithm with shift:*

$$x^{(k+1)} = \operatorname*{argmin}_x \left(\langle \nabla h(x^{(k)}) - A^T z^{(k)}, x \rangle + \frac{1}{\tau} d_{\mathrm{p}}(x, x^{(k)}) \right) \tag{39a}$$

$$z^{(k+1)} = \operatorname{prox}_{\sigma g^*}^{\phi_{\mathrm{d}}} \left(z^{(k)}, -\sigma A(2x^{(k+1)} - x^{(k)}) \right). \tag{39b}$$

Furthermore, when $A = I$ in (21), we recover the reduced Bregman primal Condat–Vũ algorithm. Similarly, setting $A = I$ in Bregman PDHG yields the Bregman Douglas–Rachford algorithm. Last, when we set $A = I$ in (39), we have the *Bregman reduced Loris–Verhoeven algorithm with shift:*

$$x^{(k+1)} = \operatorname*{argmin}_x \left(\langle \nabla h(x^{(k)}) - z^{(k)}, x \rangle + \frac{1}{\tau} d_{\mathrm{p}}(x, x^{(k)}) \right) \tag{40a}$$

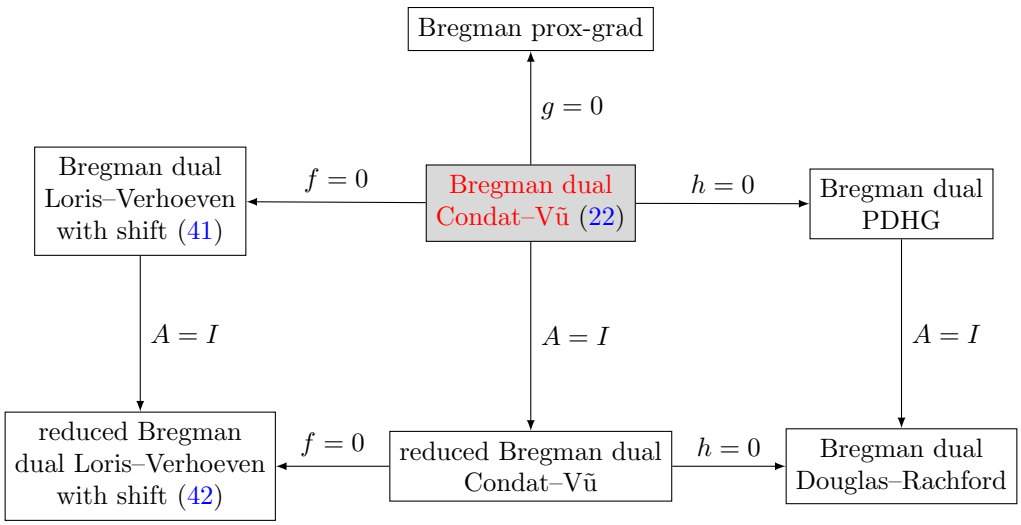

Figure 6: Proximal algorithms derived from Bregman dual Condat–Vũ algorithm (22).

$$z^{(k+1)} = \operatorname{prox}_{\sigma g^*}^{\phi_{\mathrm{d}}}\big(z^{(k)}, -\sigma(2x^{(k+1)} - x^{(k)})\big). \tag{40b}$$

Similarly, the Bregman dual Condat–Vũ algorithm (22) can be reduced to some other Bregman proximal splitting methods, as summarized in Figure 6. In particular, when $f = 0$ in (22), we obtain the *Bregman dual Loris–Verhoeven algorithm with shift*:

$$z^{(k+1)} = \operatorname{prox}_{\sigma g^*}^{\phi_{\mathrm{d}}}(z^{(k)}, -\sigma A x^{(k)}) \tag{41a}$$

$$x^{(k+1)} = \operatorname*{argmin}_x \big(\langle A^T(2z^{(k+1)} - z^{(k)}) + \nabla h(x^{(k)}), x\rangle + \frac{1}{\tau} d_{\mathrm{p}}(x, x^{(k)})\big). \tag{41b}$$

Moreover, setting $A = I$ in (41) yields the *reduced Bregman Loris–Verhoeven algorithm with shift:*

$$z^{(k+1)} = \operatorname{prox}_{\sigma g^*}^{\phi_{\mathrm{d}}}(z^{(k)}, -\sigma x^{(k)}) \tag{42a}$$

$$x^{(k+1)} = \operatorname*{argmin}_x \big(\langle 2z^{(k+1)} - z^{(k)} + \nabla h(x^{(k)}), x\rangle + \frac{1}{\tau} d_{\mathrm{p}}(x, x^{(k)})\big). \tag{42b}$$

6 Bregman dual Condat–Vũ algorithm with line search

The algorithms (21) and (22) use constant parameters σ and τ. The stepsize condition (27) involves the matrix norm $\|A\|$ and the Lipschitz constant L in (24). Estimating or bounding $\|A\|$ for a large matrix can be difficult. As an added complication, the norms $\|\cdot\|_{\mathrm{p}}$ and $\|\cdot\|_{\mathrm{d}}$ in the definition of the matrix norm (25) are assumed to be scaled so that the strong convexity parameters of the primal and dual kernels are equal to one. Close bounds on the strong convexity parameters may also be difficult to obtain. Using conservative bounds for $\|A\|$ and L results in unnecessarily small values of σ and τ, and can dramatically slow down the convergence. Even when the estimates of $\|A\|$ and L are accurate, the requirements for the stepsizes (27) are still too strict in most iterations, as observed in [1]. In view of the above arguments, line search techniques for primal–dual proximal methods have recently become an active area of research. Malitsky and Pock [35] proposed a line search technique for PDHG and the Condat–Vũ algorithm in the Euclidean case. The algorithm with

adaptive parameters in [48] focuses on a special case of (1) (*i.e.*, $f = 0$) and extends the Loris–Verhoeven algorithm (15). A Bregman proximal splitting method with line search is discussed in [29] and considers the problem (1) with $h = 0$ and $g = \delta_{\{b\}}$. In this section, we extend the Bregman dual Condat–Vũ algorithm (22) with a varying parameter option, in which the stepsizes are chosen adaptively without requiring any estimates or bounds for $\|A\|$ or the strong convexity parameter of the kernels. The algorithm is restricted to problems in the equality constrained form

$$\begin{array}{ll} \text{minimize} & f(x) + h(x) \\ \text{subject to} & Ax = b. \end{array} \tag{43}$$

This is a special case of (1) with $g = \delta_{\{b\}}$, the indicator function of the singleton $\{b\}$.

The details of the algorithm are discussed in Section 6.1 and a convergence analysis is presented in Section 6.2. The main conclusion is an $O(1/k)$ rate of ergodic convergence, consistent with previous results for related algorithms [35, 29].

Assumptions We make the same assumptions as in Section 5.1, but define

$$\phi_{\mathrm{d}}(z) = \frac{1}{2}\|z\|^2, \qquad d_{\mathrm{d}}(z, z') = \frac{1}{2}\|z - z'\|^2, \qquad \|z\|_{\mathrm{d}} = \|z\|,$$

where $\|\cdot\|$ is the Euclidean norm, and define $\|A\|$ accordingly as

$$\|A\| = \sup_{u\neq 0, v\neq 0} \frac{\langle v, Au\rangle}{\|v\|\|u\|_{\mathrm{p}}} = \sup_{u\neq 0} \frac{\|Au\|}{\|u\|_{\mathrm{p}}} = \sup_{v\neq 0} \frac{\|A^T v\|_{\mathrm{p},*}}{\|v\|}.$$

6.1 Algorithm

The algorithm uses the following iteration, with starting points $x^{(0)} \in \mathbf{int}(\mathbf{dom}\,\phi_{\mathrm{p}}) \cap \mathbf{dom}\,h$ and $z^{(-1)} = z^{(0)}$:

$$\bar{z}^{(k+1)} = z^{(k)} + \theta_k(z^{(k)} - z^{(k-1)}) \tag{44a}$$

$$x^{(k+1)} = \mathrm{prox}_{\tau_k f}^{\phi_{\mathrm{p}}}\big(x^{(k)}, \tau_k(A^T \bar{z}^{(k+1)} + \nabla h(x^{(k)}))\big) \tag{44b}$$

$$z^{(k+1)} = z^{(k)} + \sigma_k(Ax^{(k+1)} - b). \tag{44c}$$

With constant parameters $\theta_k = 1$, $\sigma_k = \sigma$, $\tau_k = \tau$, the algorithm can be simplified as

$$\begin{aligned} x^{(k+1)} &= \mathrm{prox}_{\tau f}^{\phi_{\mathrm{p}}}\big(x^{(k)}, \tau A^T(2z^{(k)} - z^{(k-1)}) + \tau\nabla h(x^{(k)}))\big) \\ z^{(k+1)} &= z^{(k)} + \sigma(Ax^{(k+1)} - b). \end{aligned}$$

Except for the numbering of the dual iterates, this is the Bregman dual Condat–Vũ algorithm (22) applied to (43).

In the line search algorithm, the parameters θ_k, τ_k, σ_k are determined by a backtracking search. At the start of the algorithm, we set τ_{-1} and σ_{-1} to some positive values. To start the search in iteration k we choose $\bar{\theta}_k \geq 1$. For $i = 0, 1, 2, \ldots$, we set $\theta_k = 2^{-i}\bar{\theta}_k$, $\tau_k = \theta_k\tau_{k-1}$, $\sigma_k = \theta_k\sigma_{k-1}$, and compute $\bar{z}_{k+1}$, x_{k+1}, z_{k+1} using (44). For some $\delta \in (0, 1]$, if

$$\langle z^{(k+1)} - \bar{z}^{(k+1)}, A(x^{(k+1)} - x^{(k)})\rangle + h(x^{(k+1)}) - h(x^{(k)}) - \langle \nabla h(x^{(k)}), x^{(k+1)} - x^{(k)}\rangle$$

$$\leq \quad \frac{\delta^2}{\tau_k} d_{\mathrm{p}}(x^{(k+1)}, x^{(k)}) + \frac{1}{2\sigma_k} \|\bar{z}^{(k+1)} - z^{(k+1)}\|^2, \tag{45}$$

we accept the computed iterates $\bar{z}^{(k+1)}$, $x^{(k+1)}$, $z^{(k+1)}$ and parameters θ_k, σ_k, τ_k, and terminate the backtracking search. If (45) does not hold, we increment i and continue the backtracking search.

The backtracking condition (45) is similar to the condition in the line search algorithm for PDHG with Euclidean proximal operators [35, Algorithm 4], but it is not identical, even in the Euclidean case. The proposed condition is weaker and allows larger stepsizes than the condition in [35, Algorithm 4].

6.2 Convergence analysis

The proof strategy is the same as in [29, Section 3.3], extended to account for the function h. The main conclusion is an $O(1/k)$ rate of ergodic convergence, shown in equation (53).

6.2.1 Lower bound on algorithm parameters

We first show that the stepsizes are bounded below by

$$\tau_k \geq \tau_{\min} \triangleq \min\left\{\tau_{-1}, \frac{-L + \sqrt{L^2 + 4\delta^2\beta^2\|A\|^2}}{2\beta^2\|A\|^2}\right\}, \qquad \sigma_k \geq \sigma_{\min} \triangleq \beta\tau_{\min}, \tag{46}$$

where $\beta = \sigma_{-1}/\tau_{-1}$. The lower bounds imply that the backtracking eventually terminates with positive stepsizes σ_k and τ_k.

Proof. Applying the result in Section 5.1.1, with $\tau = \tau_k/\delta^2$, $\sigma = \sigma_k$, we see that the backtracking condition (45) holds at iteration k if

$$\tau_k\sigma_k\|A\|^2 + \tau_k L \leq \delta^2.$$

Then mathematical induction can be used to prove (46). The two lower bounds (46) hold at $k = 0$ by the definition of $\tau_{\min}$ and $\sigma_{\min}$. Now assume $\tau_{k-1} \geq \tau_{\min}$, $\sigma_{k-1} \geq \sigma_{\min}$, and consider the kth iteration. The first attempt of θ_k is $\theta_k = \bar{\theta}_k \geq 1$. If this value is accepted, then

$$\tau_k = \bar{\theta}_k\tau_{k-1} \geq \tau_{k-1} \geq \tau_{\min}, \qquad \sigma_k = \bar{\theta}_k\sigma_{k-1} \geq \sigma_{k-1} \geq \sigma_{\min}.$$

Otherwise, one or more backtracking steps are needed. Denote by $\tilde{\theta}_k$ the last rejected value. Then $\tilde{\theta}_k^2\tau_{k-1}^2\beta^2\|A\|^2 + \tilde{\theta}_k\tau_{k-1}L > \delta^2$ and the accepted θ_k satisfies

$$\theta_k = \frac{\tilde{\theta}_k}{2} \geq \frac{-L + \sqrt{L^2 + 4\delta^2\beta^2\|A\|^2}}{2\tau_{k-1}\beta^2\|A\|^2}.$$

Therefore,

$$\tau_k = \theta_k\tau_{k-1} > \frac{-L + \sqrt{L^2 + 4\delta^2\beta^2\|A\|^2}}{2\beta^2\|A\|^2}, \qquad \sigma_k = \beta\tau_k \geq \beta\tau_{\min}. \qquad \square$$

6.2.2 One-iteration analysis

The iterates $x^{(k+1)}$, $z^{(k+1)}$, $\bar{z}^{(k+1)}$ generated by the algorithm (44) satisfy

$$\begin{aligned}\mathcal{L}(x^{(k+1)}, z) - \mathcal{L}(x, \bar{z}^{(k+1)}) \leq \frac{1}{\tau_k}\Big(d_{\mathrm{p}}(x, x^{(k)}) - d_{\mathrm{p}}(x, x^{(k+1)}) - (1-\delta^2) d_{\mathrm{p}}(x^{(k+1)}, x^{(k)})\Big) \\ + \frac{1}{2\sigma_k}\Big(\|z - z^{(k)}\|^2 - \|z - z^{(k+1)}\|^2 - \|\bar{z}^{(k+1)} - z^{(k)}\|^2\Big)\end{aligned} \tag{47}$$

for all $x \in \mathbf{dom}\, f \cap \mathbf{dom}\, \phi_{\mathrm{p}}$ and all z. Here $\mathcal{L}(x, z) = f(x) + h(x) + \langle z, Ax - b\rangle$.

Proof. The optimality condition for the primal prox-operator (44b) gives

$$\begin{aligned} & f(x^{(k+1)}) - f(x) \\ \leq\ & \frac{1}{\tau_k}\big(d_{\mathrm{p}}(x, x^{(k)}) - d_{\mathrm{p}}(x, x^{(k+1)}) - d_{\mathrm{p}}(x^{(k+1)}, x^{(k)})\big) + \langle A^T \bar{z}^{(k+1)} + \nabla h(x^{(k)}), x - x^{(k+1)}\rangle \end{aligned}$$

for all $x \in \mathbf{dom}\, f \cap \mathbf{dom}\, \phi_{\mathrm{p}}$. Hence

$$\begin{aligned} & f(x^{(k+1)}) + h(x^{(k+1)}) - f(x) - h(x) \\ \leq\ & \frac{1}{\tau_k}(d_{\mathrm{p}}(x, x^{(k)}) - d_{\mathrm{p}}(x, x^{(k+1)}) - d_{\mathrm{p}}(x^{(k+1)}, x^{(k)})) + \langle A^T \bar{z}_{k+1}, x - x^{(k+1)}\rangle \\ & + h(x^{(k+1)}) - h(x) + \langle \nabla h(x^{(k)}), x - x^{(k+1)}\rangle \\ \leq\ & \frac{1}{\tau_k}(d_{\mathrm{p}}(x, x^{(k)}) - d_{\mathrm{p}}(x, x^{(k+1)}) - d_{\mathrm{p}}(x^{(k+1)}, x^{(k)})) + \langle A^T \bar{z}^{(k+1)}, x - x^{(k+1)}\rangle \\ & + h(x^{(k+1)}) - h(x^{(k)}) - \langle \nabla h(x^{(k)}), x^{(k+1)} - x^{(k)}\rangle. \end{aligned} \tag{48}$$

The second inequality follows from the convexity of h, *i.e.*, $h(x) \geq h(x^{(k)}) + \langle \nabla h(x^{(k)}), x - x^{(k)}\rangle$. The dual update (44c) implies that

$$\langle z - z^{(k+1)}, Ax^{(k+1)} - b\rangle = \frac{1}{\sigma_k}\langle z - z^{(k+1)}, z^{(k+1)} - z^{(k)}\rangle \quad \text{for all } z. \tag{49}$$

This equality at $k = i - 1$ is

$$\begin{aligned} \langle z - z^{(i)}, Ax^{(i)} - b\rangle &= \frac{1}{\sigma_{i-1}}\langle z - z^{(i)}, z^{(i)} - z^{(i-1)}\rangle \\ &= \frac{1}{2\sigma_{i-1}}\left(\|z - z^{(i-1)}\|^2 - \|z - z^{(i)}\|^2 - \|z^{(i)} - z^{(i-1)}\|^2\right). \end{aligned} \tag{50}$$

The equality (49) at $k = i - 2$ is

$$\begin{aligned} \langle z - z^{(i-1)}, Ax^{(i-1)} - b\rangle &= \frac{1}{\sigma_{i-2}}\langle z - z^{(i-1)}, z^{(i-1)} - z^{(i-2)}\rangle \\ &= \frac{\theta_{i-1}}{\sigma_{i-1}}\langle z - z^{(i-1)}, z^{(i-1)} - z^{(i-2)}\rangle \\ &= \frac{1}{\sigma_{i-1}}\langle z - z^{(i-1)}, \bar{z}^{(i)} - z^{(i-1)}\rangle. \end{aligned}$$

We evaluate this at $z = z^{(i)}$ and add it to the equality at $z = z^{(i-2)}$ multiplied by θ_{i-1}:

$$\begin{aligned}\langle z^{(i)} - \bar{z}^{(i)}, Ax^{(i-1)} - b\rangle &= \frac{1}{\sigma_{i-1}}\langle z^{(i)} - \bar{z}^{(i)}, \bar{z}^{(i)} - z^{(i-1)}\rangle \\ &= \frac{1}{2\sigma_{i-1}}\left(\|z^{(i)} - z^{(i-1)}\|^2 - \|z^{(i)} - \bar{z}^{(i)}\|^2 - \|\bar{z}^{(i)} - z^{(i-1)}\|^2\right). \end{aligned} \tag{51}$$

Now we combine (48) for $k = i-1$, with (50) and (51). For $i \geq 1$,

$$\begin{aligned}
&\mathcal{L}(x^{(i)}, z) - \mathcal{L}(x, \bar{z}^{(i)}) \\
&= f(x^{(i)}) + h(x^{(i)}) + \langle z, Ax^{(i)} - b\rangle - f(x) - h(x) - \langle \bar{z}^{(i)}, Ax - b\rangle \\
&\leq \frac{1}{\tau_{i-1}}\Big(d_{\mathrm{p}}(x, x^{(i-1)}) - d_{\mathrm{p}}(x, x^{(i)}) - d_{\mathrm{p}}(x^{(i)}, x^{(i-1)})\Big) + \langle A^T\bar{z}^{(i)}, x - x^{(i)}\rangle + \langle z, Ax^{(i)} - b\rangle \\
&\quad - \langle \bar{z}^{(i)}, Ax - b\rangle + h(x^{(i)}) - h(x^{(i-1)}) - \langle \nabla h(x^{(i-1)}), x^{(i)} - x^{(i-1)}\rangle \\
&= \frac{1}{\tau_{i-1}}\Big(d_{\mathrm{p}}(x, x^{(i-1)}) - d_{\mathrm{p}}(x, x^{(i)}) - d_{\mathrm{p}}(x^{(i)}, x^{(i-1)})\Big) + \langle z - \bar{z}^{(i)}, Ax^{(i)} - b\rangle \\
&\quad + h(x^{(i)}) - h(x^{(i-1)}) - \langle \nabla h(x^{(i-1)}), x^{(i)} - x^{(i-1)}\rangle \\
&= \frac{1}{\tau_{i-1}}\Big(d_{\mathrm{p}}(x, x^{(i-1)}) - d_{\mathrm{p}}(x, x^{(i)}) - d_{\mathrm{p}}(x^{(i)}, x^{(i-1)})\Big) \\
&\quad + \langle z^{(i)} - \bar{z}^{(i)}, A(x^{(i)} - x^{(i-1)})\rangle + \langle z - z^{(i)}, Ax^{(i)} - b\rangle + \langle z^{(i)} - \bar{z}^{(i)}, Ax^{(i-1)} - b\rangle \\
&\quad + h(x^{(i)}) - h(x^{(i-1)}) - \langle \nabla h(x^{(i-1)}), x^{(i)} - x^{(i-1)}\rangle \\
&= \frac{1}{\tau_{i-1}}\Big(d_{\mathrm{p}}(x, x^{(i-1)}) - d_{\mathrm{p}}(x, x^{(i)}) - d_{\mathrm{p}}(x^{(i)}, x^{(i-1)}))\Big) \\
&\quad + \frac{1}{2\sigma_{i-1}}\Big(\|z - z^{(i-1)}\|^2 - \|z - z^{(i)}\|^2 - \|\bar{z}^{(i)} - z^{(i-1)}\|^2 - \|\bar{z}^{(i)} - z^{(i)}\|^2\Big) \\
&\quad + \langle A^T(z^{(i)} - \bar{z}^{(i)}), x^{(i)} - x^{(i-1)}\rangle + h(x^{(i)}) - h(x^{(i-1)}) - \langle \nabla h(x^{(i-1)}), x^{(i)} - x^{(i-1)})\rangle \\
&\leq \frac{1}{\tau_{i-1}}\left(d_{\mathrm{p}}(x, x^{(i-1)}) - d_{\mathrm{p}}(x, x^{(i)}) - (1-\delta^2)d_{\mathrm{p}}(x^{(i)}, x^{(i-1)})\right) \\
&\quad + \frac{1}{2\sigma_{i-1}}\left(\|z - z^{(i-1)}\|^2 - \|z - z^{(i)}\|^2 - \|\bar{z}^{(i)} - z^{(i-1)}\|^2\right),
\end{aligned}$$

which is the desired result (47). The first inequality follows from (48). In the second last step we substitute (50) and (51). The last step uses the line search exit condition (45) at $k = i-1$. □

6.2.3 Ergodic convergence

We define the averaged primal and dual sequences

$$x_{\mathrm{avg}}^{(k)} = \frac{1}{\sum_{i=1}^k \tau_{i-1}}\sum_{i=1}^k \tau_{i-1}x^{(i)}, \qquad \bar{z}_{\mathrm{avg}}^{(k)} = \frac{1}{\sum_{i=1}^k \tau_{i-1}}\sum_{i=1}^k \tau_{i-1}\bar{z}^{(i)}$$

for $k \geq 1$. We show that

$$\mathcal{L}(x_{\mathrm{avg}}^{(k)}, z) - \mathcal{L}(x, \bar{z}_{\mathrm{avg}}^{(k)}) \leq \frac{1}{\sum_{i=1}^k \tau_{i-1}}\big(d(x, x^{(0)}) + \frac{1}{2\beta}\|z - z^{(0)}\|^2\big) \tag{52}$$

$$\leq \frac{1}{k\tau_{\min}}\big(d(x, x^{(0)}) + \frac{1}{2\beta}\|z - z^{(0)}\|^2\big) \tag{53}$$

for all $x \in \mathbf{dom}\, f \cap \mathbf{dom}\, \phi_{\mathrm{p}}$ and all z. This holds for any choice of $\delta \in (0, 1]$ in (45). If we compare (52) and (33), we note that the two left-hand sides involve different dual iterates ($\bar{z}_{\mathrm{avg}}^{(k)}$ as opposed to $z_{\mathrm{avg}}^{(k)}$).

Proof. From (47),

$$\mathcal{L}(x^{(i)}, z) - \mathcal{L}(x, \bar{z}^{(i)}) \leq \frac{1}{\tau_{i-1}}\Big(d_{\mathrm{p}}(x, x^{(i-1)}) - d_{\mathrm{p}}(x, x^{(i)}) + \frac{1}{2\beta}\|z - z^{(i-1)}\|^2 - \frac{1}{2\beta}\|z - z^{(i)}\|^2\Big).$$

Since $\mathcal{L}$ is convex in x and affine in z,

$$\begin{aligned}
&(\sum_{i=1}^{k} \tau_{i-1})\big(\mathcal{L}(x_{\mathrm{avg}}^{(k)}, z) - \mathcal{L}(x, \bar{z}_{\mathrm{avg}}^{(k)})\big) \\
&\quad\leq \sum_{i=1}^{k} \tau_{i-1}(\mathcal{L}(x^{(i)}, z) - \mathcal{L}(x, \bar{z}^{(i)})) \\
&\quad\leq d_{\mathrm{p}}(x, x^{(0)}) - d_{\mathrm{p}}(x, x^{(k)}) + \frac{1}{2\beta}(\|z - z^{(0)}\|^2 - \|z - z^{(k)}\|^2) \\
&\quad\leq d_{\mathrm{p}}(x, x^{(0)}) + \frac{1}{2\beta}\|z - z^{(0)}\|^2. \qquad (54)
\end{aligned}$$

Dividing by $\sum_{i=1}^{k} \tau_{i-1}$ and plugging in $x = x^\star$, $z = z^\star$ gives (52). □

Substituting $x = x^\star$ in (54) yields

$$f(x_{\mathrm{avg}}^{(k)}) + h(x_{\mathrm{avg}}^{(k)}) + z^T(Ax_{\mathrm{avg}}^{(k)} - b) - f(x^\star) \leq \frac{1}{\sum_{i=1}^{k} \tau_{i-1}}\big(d_{\mathrm{p}}(x^\star, x^{(0)}) + \frac{1}{2\beta}\|z - z^{(0)}\|^2\big)$$

for all z, since $Ax^\star = b$. More generally, suppose $X \subseteq \mathbf{dom}\, f \cap \mathbf{dom}\, \phi_{\mathrm{p}}$ is a compact set containing an optimal solution $x^\star$, and $Z = \{z \mid \|z\| \leq \gamma\}$ contains a dual optimal $z^\star$, then the merit function η defined in (6) satisfies

$$\begin{aligned}
\eta(x_{\mathrm{avg}}^{(k)}, \bar{z}_{\mathrm{avg}}^{(k)}) &= f(x_{\mathrm{avg}}^{(k)}) + h(x_{\mathrm{avg}}^{(k)}) + \gamma\|Ax_{\mathrm{avg}}^{(k)} - b\| - \inf_{x \in X}(f(x) + h(x) + z^T(Ax - b)) \\
&\leq \frac{1}{\sum_{i=1}^{k} \tau_{i-1}}\Big(\sup_{x \in X} d_{\mathrm{p}}(x, x^{(0)}) + \frac{1}{2\beta}(\gamma + \|z^{(0)}\|)^2\Big) \\
&\leq \frac{1}{k\tau_{\min}}\Big(\sup_{x \in X} d_{\mathrm{p}}(x, x^{(0)}) + \frac{1}{2\beta}(\gamma + \|z^{(0)}\|)^2\Big).
\end{aligned}$$

6.2.4 Monotonicity properties and convergence of iterates

For $x = x^\star$, $z = z^\star$, the left-hand side of (47) is nonnegative and we obtain

$$\begin{aligned}
&d(x^\star, x^{(k+1)}) + \frac{1}{2\beta}\|z^\star - z^{(k+1)}\|^2 \\
&\quad\leq d(x^\star, x^{(k)}) + \frac{1}{2\beta}\|z^\star - z^{(k)}\|^2 - \big((1 - \delta^2)d(x^\star, x^{(k)}) + \frac{1}{2\beta}\|z^\star - z^{(k)}\|^2\big)
\end{aligned}$$

$$\leq \quad d(x^\star, x^{(k)}) + \frac{1}{2\beta}\|z^\star - z^{(k)}\|^2$$

for $k \geq 0$. Moreover,

$$\sum_{i=0}^{k} \Big((1-\delta^2)(d_{\mathrm{p}}(x^{(i+1)}, x^{(i)}) + \frac{1}{2\beta}\|\bar{z}^{(i+1)} - z^{(i)}\|^2 \Big) \leq d_{\mathrm{p}}(x^\star, x^{(0)}) + \frac{1}{2\beta}\|z^\star - \bar{z}^{(0)}\|^2.$$

These inequalities hold for any value $\delta \in (0, 1]$. In particular, the last inequality implies that $\bar{z}^{(i+1)} - z^{(i)} \to 0$. When $\delta < 1$ it also implies that $d_{\mathrm{p}}(x^{(i+1)}, x^{(i)}) \to 0$ and, by the strong convexity assumption on ϕ_{p}, that $x^{(i+1)} - x^{(i)} \to 0$. With additional assumptions similar to those in Section 5.2.3, one can show the convergence of iterates; see [29, Section 3.3.4].

7 Bregman PD3O algorithm

In this section we propose the *Bregman PD3O algorithm*, another Bregman proximal method for the problem (1). Bregman PD3O also involves two generalized distances, d_{p} and d_{d}, generated by ϕ_{p} and ϕ_{d}, respectively, and it consists of the iterations

$$x^{(k+1)} = \mathrm{prox}_{\tau f}^{\phi_{\mathrm{p}}}(x^{(k)}, \tau A^T z^{(k)} + \tau \nabla h(x^{(k)})) \tag{55a}$$

$$z^{(k+1)} = \mathrm{prox}_{\sigma g^*}^{\phi_{\mathrm{d}}}(z^{(k)}, -\sigma A(2x^{(k+1)} - x^{(k)} + \tau(\nabla h(x^{(k)}) - \nabla h(x^{(k+1)})))). \tag{55b}$$

The only difference between Bregman PD3O and Bregman primal Condat–Vũ algorithm (21) is the additional term $\tau(\nabla h(x^{(k)}) - \nabla h(x^{(k+1)}))$. Thus the two algorithms (21) and (55) reduce to the same method when h is absent from problem (1). The additional term allows PD3O to use larger stepsizes than the Condat–Vũ algorithm. If we use the same matrix norm $\|A\|$ and Lipschitz constant L in the analysis for the two methods, then the conditions are

$$\begin{array}{ll} \text{Condat–Vũ:} & \sigma\tau\|A\|^2 + \tau L \leq 1 \\ \text{PD3O:} & \sigma\tau\|A\|^2 \leq 1, \quad \tau \leq 1/L. \end{array} \tag{56}$$

The range of possible parameters is illustrated in Figure 7.

In Section 7.1 we provide the detailed convergence analysis of the Bregman PD3O method. The connections between Bregman PD3O and several other Bregman proximal methods are discussed in Section 7.2.

Assumptions Throughout Section 7 we make the following assumptions. The kernel functions ϕ_{p} and ϕ_{d} are 1-strongly convex with respect to the Euclidean norm and an arbitrary norm $\|\cdot\|_{\mathrm{d}}$, respectively:

$$d_{\mathrm{p}}(x, x') \geq \frac{1}{2}\|x - x'\|^2, \qquad d_{\mathrm{d}}(z, z') \geq \frac{1}{2}\|z - z'\|_{\mathrm{d}}^2. \tag{57}$$

The assumptions that the strong convexity constants are one can be made without loss of generality, by scaling the distances. The definition of $\|A\|$ follows (25) and reduces to

$$\|A\| = \sup_{v \neq 0} \frac{\|Av\|_{\mathrm{d},*}}{\|v\|}.$$

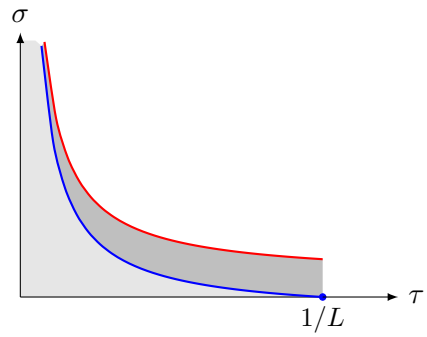

Figure 7: Acceptable stepsizes in Condat–Vũ algorithms and PD3O. We assume the same matrix norm $\|A\|$ and Lipschitz constant L are used in the analysis of the two algorithms. The light gray region under the blue curve is defined by the inequality for the Condat–Vũ algorithms in (56). The region under the red curve shows the values allowed by the stepsized conditions for PD3O.

We also assume that the gradient of h is L-Lipschitz continuous with respect to the Euclidean norm: $\mathbf{dom}\, h = \mathbf{R}^n$ and

$$h(y) - h(x) - \langle \nabla h(x), y - x \rangle \leq \frac{L}{2}\|y - x\|^2, \quad \text{for any } x, y \in \mathbf{dom}\, h. \tag{58}$$

The parameters τ and σ must satisfy

$$\sigma\tau\|A\|^2 \leq 1, \qquad \tau \leq 1/L. \tag{59}$$

Finally, we assume that the optimality condition (3) has a solution $(x^\star, z^\star) \in \mathbf{dom}\, \phi_{\rm p} \times \mathbf{dom}\, \phi_{\rm d}$.

Note that (58) is a stronger assumption than (24). (Combined with the first inequality in (57), it implies (24).) We will use the following consequence of (58):

$$h(y) - h(x) - \langle \nabla h(x), y - x \rangle \geq \frac{1}{2L}\|\nabla h(y) - \nabla h(x)\|^2 \tag{60}$$

for all x, y [37, Theorem 2.1.5].

7.1 Convergence analysis

7.1.1 A primal–dual Bregman distance

We introduce a primal–dual kernel

$$\phi_{\rm pd3o}(x, y, z) = \frac{1}{\tau}\phi_{\rm p}(x) + \frac{1}{\sigma}\phi_{\rm d}(z) + \frac{\tau}{2}\|y\|^2 - \langle y, x \rangle - \langle z, A(x - \tau y) \rangle,$$

where $\sigma, \tau > 0$. If $\phi_{\rm pd3o}$ is convex, the generated Bregman distance is given by

$$\begin{aligned} d_{\rm pd3o}(x, y, z; x', y', z') = {} & \frac{1}{\tau}d_{\rm p}(x, x') + \frac{1}{\sigma}d_{\rm d}(z, z') + \frac{\tau}{2}\|y - y'\|^2 \\ & - \langle y - y', x - x' \rangle - \langle z - z', A(x - x') \rangle + \tau\langle z - z', A(y - y') \rangle. \end{aligned} \tag{61}$$

We now show that $\phi_{\rm pd3o}$ is convex if $\sigma\tau\|A\|^2 \leq 1$.

Proof. It is sufficient to show that d_{pd3o} is nonnegative:

$$\begin{aligned} d_{\mathrm{pd3o}}(x,y,z;x',y',z') &\geq \frac{1}{2\tau}\|x-x'\|^2 + \frac{\tau}{2}\|A^T(z-z')\|^2 + \frac{\tau}{2}\|y-y'\|^2 \\ &\quad - \langle y-y', x-x'\rangle - \langle z-z', A(x-x')\rangle + \tau\langle z-z', A(y-y')\rangle \\ &= \frac{1}{2}\Big\|\frac{1}{\sqrt{\tau}}(x-x') - \sqrt{\tau}(y-y') - \sqrt{\tau}A^T(z-z')\Big\|^2 \\ &\geq 0. \end{aligned} \tag{62}$$

In step 1 we use the strong convexity assumption (57), the definition of $\|A\|$ (25) with $\|\cdot\|_{\mathrm{p}} = \|\cdot\|$, and the assumption $\sigma\tau\|A\|^2 \leq 1$. The bound on $d_{\mathrm{d}}(z,z')$ follows from

$$\frac{1}{\sigma}d_{\mathrm{d}}(z,z') \geq \frac{1}{2\sigma}\|z-z'\|_{\mathrm{d}}^2 \geq \frac{\|A^T(z-z')\|^2}{2\sigma\|A\|^2} \geq \frac{\tau}{2}\|A^T(z-z')\|^2. \qquad \square$$

Note that the convexity of ϕ_{pd3o} only requires the first inequality in the stepsize condition (59). Although the Bregman PD3O algorithm (55) is not the Bregman proximal point method for the Bregman kernel ϕ_{pd3o}, the distance d_{pd3o} will appear in the key inequality (63) of the convergence analysis.

7.1.2 One-iteration analysis

We first show that the iterates $x^{(k+1)}$, $z^{(k+1)}$ generated by Bregman PD3O (55) satisfy

$$\begin{aligned} &\mathcal{L}(x^{(k+1)},z) - \mathcal{L}(x,z^{(k+1)}) \\ &\quad\leq\ d_{\mathrm{pd3o}}\big(x,\nabla h(x),z;x^{(k)},\nabla h(x^{(k)}),z^{(k)}\big) - d_{\mathrm{pd3o}}\big(x,\nabla h(x),z;x^{(k+1)},\nabla h(x^{(k+1)}),z^{(k+1)}\big) \\ &\qquad - d_{\mathrm{pd3o}}\big(x^{(k+1)},\nabla h(x),z^{(k+1)};x^{(k)},\nabla h(x^{(k)}),z^{(k)}\big) \end{aligned} \tag{63}$$

for all $x \in \mathbf{dom}\, f \cap \mathbf{dom}\, \phi_{\mathrm{p}}$ and $z \in \mathbf{dom}\, g^* \cap \mathbf{dom}\, \phi_{\mathrm{d}}$.

Proof. Recall that Bregman PD3O differs from the Bregman primal Condat–Vũ algorithm (21) only in an additional term in the dual update. The proof in Section 6.2.2 therefore applies up to (31), with

$$\tilde{x} = 2x^{(k+1)} - x^{(k)} + \tau(\nabla h(x^{(k)}) - \nabla h(x^{(k+1)})), \qquad \tilde{z} = z^{(k)}.$$

Hence,

$$\begin{aligned} &\mathcal{L}(x^{(k+1)},z) - \mathcal{L}(x,z^{(k+1)}) \\ &\quad\leq\ d_-(x,z;x^{(k)},z^{(k)}) - d_-(x,z;x^{(k+1)},z^{(k+1)}) - d_-(x^{(k+1)},z^{(k+1)};x^{(k)},z^{(k)}) \\ &\qquad - \tau\langle A^T(z-z^{(k+1)}),\nabla h(x^{(k)}) - \nabla h(x^{(k+1)})\rangle \\ &\qquad - h(x^{(k+1)}) + h(x^{(k)}) + \langle \nabla h(x^{(k)}), x^{(k+1)} - x^{(k)}\rangle \\ &\quad=\ d_-(x,z;x^{(k)},z^{(k)}) + \frac{\tau}{2}\|\nabla h(x) - \nabla h(x^{(k)})\|^2 \\ &\qquad - \langle (x-\tau A^T z) - (x^{(k)} - \tau A^T z^{(k)}), \nabla h(x) - \nabla h(x^{(k)})\rangle \\ &\qquad - \Big(d_-(x,z;x^{(k+1)},z^{(k+1)}) + \frac{\tau}{2}\|\nabla h(x) - \nabla h(x^{(k+1)})\|^2 \\ &\qquad - \big\langle x - \tau A^T z - (x^{(k+1)} - \tau A^T z^{(k+1)}), \nabla h(x) - \nabla h(x^{(k+1)})\big\rangle\Big) \end{aligned}$$

$$
\begin{aligned}
& - \Big(d_-(x^{(k+1)}, z^{(k+1)}; x^{(k)}, z^{(k)}) + \frac{\tau}{2}\|\nabla h(x) - \nabla h(x^{(k)})\|^2 \\
& - \big\langle (x^{(k+1)} - \tau A^T z^{(k+1)}) - (x^{(k)} - \tau A^T z^{(k)}), \nabla h(x) - \nabla h(x^{(k)}) \big\rangle \Big) \\
& - (h(x) - h(x^{(k+1)}) - \langle \nabla h(x^{(k+1)}), x - x^{(k+1)} \rangle - \frac{\tau}{2}\|\nabla h(x) - \nabla h(x^{(k+1)})\|^2) \\
= \;& d_{\mathrm{pd3o}}(x, \nabla h(x), z; x^{(k)}, \nabla h(x^{(k)}), z^{(k)}) - d_{\mathrm{pd3o}}(x, \nabla h(x), z; x^{(k+1)}, \nabla h(x^{(k+1)}), z^{(k+1)}) \\
& - d_{\mathrm{pd3o}}(x^{(k+1)}, \nabla h(x), z^{(k+1)}; x^{(k)}, \nabla h(x^{(k)}), z^{(k)}) \\
& - (h(x) - h(x^{(k+1)}) - \langle \nabla h(x^{(k+1)}), x - x^{(k+1)} \rangle - \frac{\tau}{2}\|\nabla h(x) - \nabla h(x^{(k+1)})\|^2) \\
\leq \;& d_{\mathrm{pd3o}}(x, \nabla h(x), z; x^{(k)}, \nabla h(x^{(k)}), z^{(k)}) - d_{\mathrm{pd3o}}(x, \nabla h(x), z; x^{(k+1)}, \nabla h(x^{(k+1)}), z^{(k+1)}) \\
& - d_{\mathrm{pd3o}}(x^{(k+1)}, \nabla h(x), z^{(k+1)}; x^{(k)}, \nabla h(x^{(k)}), z^{(k)}) \\
\leq \;& d_{\mathrm{pd3o}}(x, \nabla h(x), z; x^{(k)}, \nabla h(x^{(k)}), z^{(k)}) - d_{\mathrm{pd3o}}(x, \nabla h(x), z; x^{(k+1)}, \nabla h(x^{(k+1)}), z^{(k+1)}).
\end{aligned}
$$

Step 3 follows from definition of d_{pd3o} (61). In step 4 we use the Lipschitz condition (60) and the second inequality in the stepsize condition (59). The last step follows from the fact that d_{pd3o} is nonnegative (62). □

7.1.3 Ergodic convergence

The iterates generated by Bregman PD3O (55) satisfy

$$
\mathcal{L}(x_{\mathrm{avg}}^{(k)}, z) - \mathcal{L}(x, z_{\mathrm{avg}}^{(k)}) \leq \frac{2}{k}\Big(\frac{1}{\tau} d_{\mathrm{p}}(x, x^{(0)}) + \frac{1}{\sigma} d_{\mathrm{d}}(z, z^{(0)}) + \frac{\tau}{2}\|\nabla h(x) - \nabla h(x^{(0)})\|^2\Big),
$$

for all $x \in \mathbf{dom}\, f \cap \mathbf{dom}\, \phi_{\mathrm{p}}$ and all $z \in \mathbf{dom}\, g^* \cap \mathbf{dom}\, \phi_{\mathrm{d}}$, where the averaged iterates are defined in (32).

Proof. From (63), since $\mathcal{L}(u, v)$ is convex in u and concave in v,

$$
\begin{aligned}
& \mathcal{L}(x_{\mathrm{avg}}^{(k)}, z) - \mathcal{L}(x, z_{\mathrm{avg}}^{(k)}) \\
& \leq \frac{1}{k}\sum_{i=1}^{k} \big(\mathcal{L}(x^{(i)}, z) - \mathcal{L}(x, z^{(i)})\big) \\
& \leq \frac{1}{k}\Big(d_{\mathrm{pd3o}}(x, \nabla h(x), z; x^{(0)}, \nabla h(x^{(0)}), z^{(0)}) - d_{\mathrm{pd3o}}(x, \nabla h(x), z; x^{(k)}, \nabla h(x^{(k)}), z^{(k)})\Big) \\
& \leq \frac{1}{k} d_{\mathrm{pd3o}}(x, \nabla h(x), z; x^{(0)}, \nabla h(x^{(0)}), z^{(0)}) \\
& \leq \frac{2}{k}\Big(\frac{1}{\tau} d_{\mathrm{p}}(x, x^{(0)}) + \frac{1}{\sigma} d_{\mathrm{d}}(z, z^{(0)}) + \frac{\tau}{2}\|\nabla h(x) - \nabla h(x^{(0)})\|^2\Big)
\end{aligned}
$$

for all $x \in \mathbf{dom}\, f \cap \mathbf{dom}\, \phi_{\mathrm{p}}$ and $z \in \mathbf{dom}\, g^* \cap \mathbf{dom}\, \phi_{\mathrm{d}}$. □

7.2 Relation to other Bregman proximal algorithms

The proposed algorithm (55) can be viewed as an extension to PD3O (14) using generalized distances, and reduces to several Bregman proximal methods by reduction. These algorithms can also be organized into a diagram similar to Figure 3. Figure 8 starts from Bregman PD3O (55), and summarizes its connection to several Bregman proximal methods. When $h = 0$, (55) reduces to

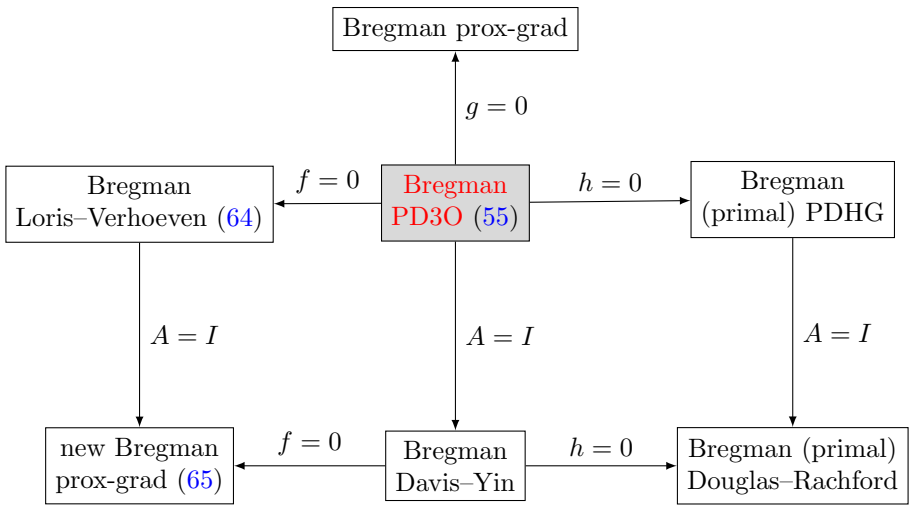

Figure 8: Proximal algorithms derived from Bregman PD3O.

Bregman PDHG, and when $g = 0$, (55) reduces to the Bregman proximal gradient algorithm. More interestingly, when $f = 0$, Bregman PD3O reduces to the Bregman Loris–Verhoeven algorithm

$$x^{(k+1)} = \operatorname*{argmin}_x \big(\langle \nabla h(x^{(k)}) - A^T z^{(k)}, x\rangle + \frac{1}{\tau} d_{\mathrm{p}}(x, x^{(k)})\big) \tag{64a}$$

$$z^{(k+1)} = \operatorname{prox}^{\phi_{\mathrm{d}}}_{\sigma g^*}\big(z^{(k)}, -\sigma A\big(2x^{(k+1)} - x^{(k)} + \tau\big(\nabla h(x^{(k)}) - \nabla h(x^{(k+1)})\big)\big)\big). \tag{64b}$$

Setting $A = I$ (with $\sigma = 1/\tau$), we obtain a new variant of Bregman proximal gradient algorithm:

$$x^{(k+1)} = \operatorname*{argmin}_x \big(\langle \nabla h(x^{(k)}) - z^{(k)}, x\rangle + \frac{1}{\tau} d_{\mathrm{p}}(x, x^{(k)})\big) \tag{65a}$$

$$z^{(k+1)} = \operatorname{prox}^{\phi_{\mathrm{d}}}_{\tau^{-1} g^*}\big(z^{(k)}, -\frac{1}{\tau} A\big(2x^{(k+1)} - x^{(k)}\big) - A(\nabla h(x^{(k)}) - \nabla h(x^{(k+1)}))\big). \tag{65b}$$

The difference between (65) and (40) is the additional term $\tau(\nabla h(x^{(k)}) - \nabla h(x^{(k+1)}))$, the same as the difference between (21) and (55). When the Euclidean proximal operator is used, (65) reduces to the proximal gradient method. Unfortunately, however, the new algorithm (65) does not seem to be equivalent to the Bregman proximal gradient algorithm due to the lack of Moreau decomposition in the generalized case. Nevertheless, the new algorithm (65) may still be interesting on its own, especially when the generalized proximal operator of g^* is easy to compute while the (Euclidean or generalized) proximal operator of g is computationally expensive. Finally, setting $A = I$ (and $\sigma = 1/\tau$) in Bregman PD3O (55) gives a Bregman Davis–Yin algorithm.

8 Numerical experiment

In this section we evaluate the performance of the Bregman primal Condat–Vũ algorithm (21), Bregman dual Condat–Vũ algorithm with line search (44), and Bregman PD3O (55). The main goal of the example is to validate and illustrate the difference in the stepsize conditions (56), and

the usefulness of the line search procedure. We consider the convex optimization problem

$$\begin{array}{ll} \text{minimize} & \psi(x) = \lambda\|Ax\|_1 + \frac{1}{2}\|Cx-b\|^2 \\ \text{subject to} & \mathbf{1}^T x = 1, \quad x \succeq 0, \end{array} \tag{66}$$

where $x \in \mathbf{R}^n$ is the optimization variable, $C \in \mathbf{R}^{m\times n}$, and $A \in \mathbf{R}^{(n-1)\times n}$ is the difference matrix

$$A = \begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & 1 \end{bmatrix}. \tag{67}$$

This problem is of the form of (1) with

$$f(x) = \delta_H(x), \qquad g(y) = \lambda\|y\|_1, \qquad g^*(z) = \begin{cases} 0 & \|z\|_\infty \le \lambda \\ +\infty & \text{otherwise}, \end{cases} \qquad h(x) = \frac{1}{2}\|Cx-b\|^2,$$

and δ_H is the indicator function of the hyperplane $H = \{x \in \mathbf{R}^n \mid \mathbf{1}^T x = 1\}$. We use the relative entropy distance

$$d_{\mathrm{p}}(x,y) = \sum_{i=1}^n (x_i \log(x_i/y_i) - x_i + y_i), \qquad \mathbf{dom}\, d_{\mathrm{p}} = \mathbf{R}^n_+ \times \mathbf{R}^n_{++}.$$

in the primal space. This distance is 1-strongly convex with respect to ℓ_1-norm [5] (and also ℓ_2-norm). With the relative entropy distance, all the primal iterates $x^{(k)}$ remain feasible. In the dual space we use the Euclidean distance. Thus, the matrix norm (25) in the stepsize condition (27) for the Bregman Condat–Vũ algorithms is the (1,2)-operator norm

$$\|A\|_{1,2} = \sup_{v\neq 0} \frac{\|Av\|}{\|v\|_1} = \max_{i=1,\ldots,n} \|a_i\| = \sqrt{2},$$

where a_i is the ith column of A. In the Bregman PD3O algorithm, we use the squared Euclidean distance $d_{\mathrm{p}}(x,y) = \frac{1}{2}\|x-y\|^2$, and the matrix norm in the stepsize condition (59) is the spectral norm $\|A\|_2$. For the difference matrix (67), $\|A\|_2$ is bounded above by 2, and very close to this upper bound for large n.

The Lipschitz constant for h with respect to the ℓ_1-norm is the largest absolute value of the elements in C^TC, *i.e.*, $L_1 = \max_{i,j} |(C^TC)_{ij}|$. This value is used in the stepsize condition (27) for the Bregman Condat–Vũ algorithms. The Lipschitz constant with respect to the ℓ_2-norm is $L_2 = \|C\|_2$, which is used in the stepsize condition (59) for Bregman PD3O.

The matrix norms and Lipschitz constants are summarized as follows:

	matrix norm	Lipschitz constant
Bregman Condat–Vũ	$\|A\|_{1,2} = \sqrt{2}$	$L_1 = \max_{i,j} \|(C^TC)_{ij}\|$
Bregman PD3O	$\|A\|_2 \le 2$	$L_2 = \|C\|_2$.

In the example we use the exact values of L_1 and L_2,

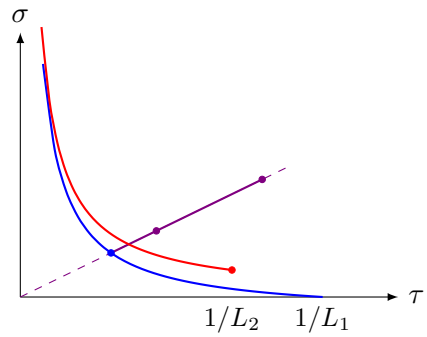

Figure 9: The blue and red curves show the boundaries of the stepsize regions for Bregman Condat–Vũ algorithms and Bregman PD3O, respectively. The blue and red points indicate the chosen parameters in (68) (red for for PD3O, blue for Condat–Vũ). In the Bregman dual Condat–Vũ algorithm with line search, the stepsizes are selected on the dashed straight line. The solid line segment shows the range of stepsizes that were selected, with dots indicating the largest, median, and smallest stepsizes.

The Bregman proximal operator of f has a closed-form solution:

$$\operatorname{prox}_f^{\phi}(y, a) = \frac{1}{\sum_{i=1}^n y_i e^{-a_i}} \begin{bmatrix} y_1 e^{-a_1} \\ \vdots \\ y_n e^{-a_n} \end{bmatrix},$$

and the (Euclidean) proximal operator of g^* is the projection onto the infinity norm ball:

$$\operatorname{prox}_{g^*}(z)_i = \begin{cases} \lambda & z_i > \lambda \\ z_i & |z_i| \leq \lambda \\ -\lambda & z_i < -\lambda. \end{cases}$$

The experiment is carried out in Python 3.6 on a desktop with an Intel Core i5 2.4GHz CPU and 8GB RAM. We set $m = 500$ and $n = 10,000$. The elements in the matrix $C \in \mathbf{R}^{m \times n}$ and $b \in \mathbf{R}^m$ are randomly generated from independent standard Gaussian distributions. For the constant stepsize option, we choose

$$\begin{array}{lll} \text{Condat-Vũ} & \sigma = L_1/2 & \tau = 1/(2L_1) \\ \text{PD3O} & \sigma = L_2/4 & \tau = 1/L_2. \end{array} \tag{68}$$

These two choices, as well as the range of possible parameters, are illustrated in Figure 9. The two choices are on the blue and red curve, respectively, and satisfy the requirement (56) with equality. For the line search algorithm, we set $\bar{\theta}_k = 1.2$ to encourage more aggressive updates, and $\beta = \sigma_{-1}/\tau_{-1} = L_1^2$, which is consistent with the choice in (68).

Numerical results We solve the problem (66) using the Bregman primal Condat–Vũ algorithm (21), the Bregman dual Condat–Vũ algorithm with line search (44), and Bregman PD3O (14). Figure 10 reports the relative distance between the function values to the optimal value $\psi^\star$, which

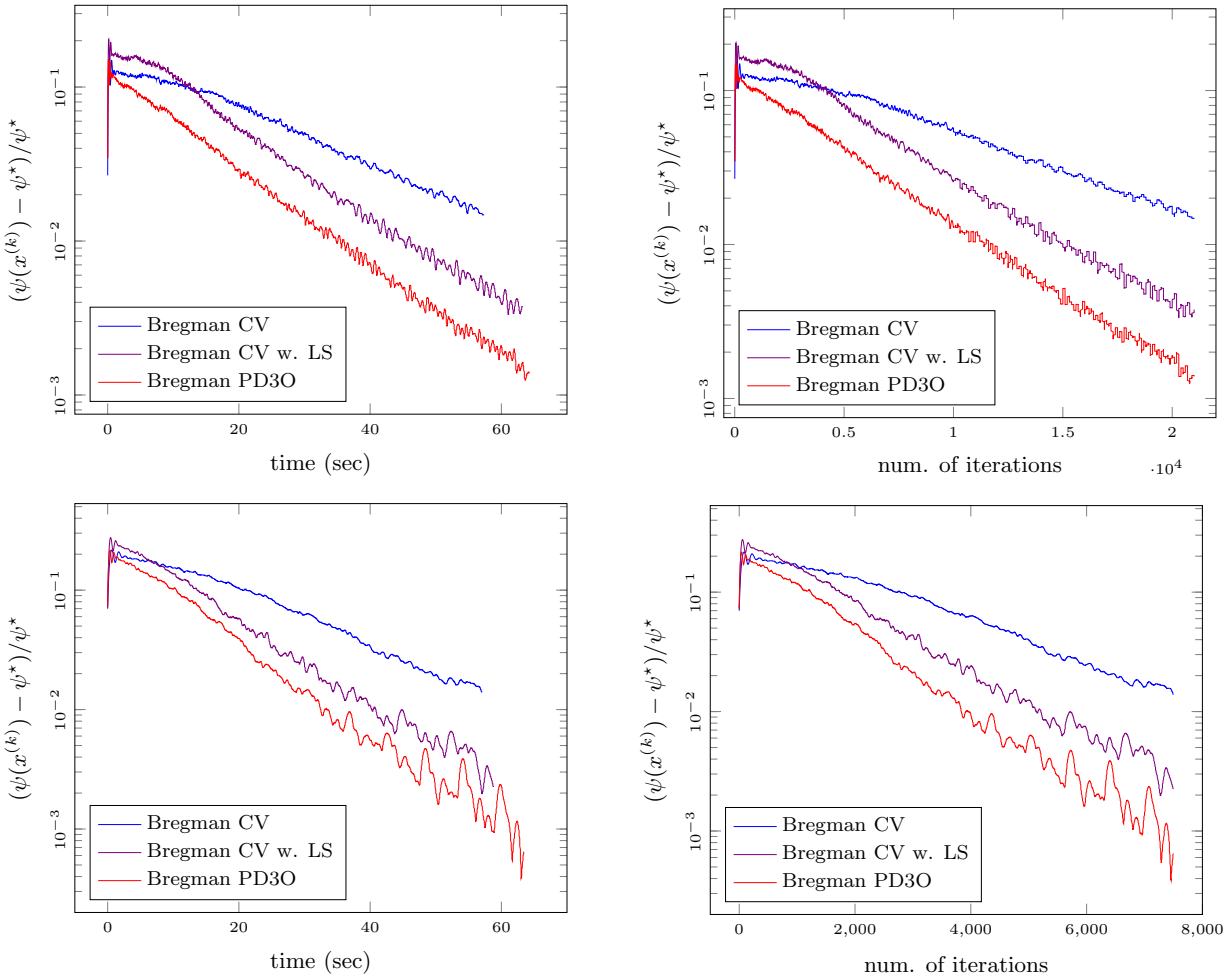

Figure 10: Comparison of three algorithms (Bregman primal Condat–Vũ, Bregman dual Condat–Vũ with line search, and Bregman PD3O) in terms of objective values. The top two figures plot the relative error of the function value versus CPU time and number of iterations for one problem instance (66), respectively. The bottom two figures correspond to another problem instance.

is computed via CVXPY [21]. Comparison between the Bregman primal Condat–Vũ algorithm and Bregman PD3O shows that Bregman PD3O converges faster. Figure 10 also compares the performance between the Bregman primal Condat–Vũ algorithm with constant stepsizes and Bregman dual algorithm with line search. One can see clearly that the line search significantly improves the convergence. On the other hand, the line search does not add much computation overhead, as the plots of the CPU time and the number of iterations are roughly identical. In these experiments Bregman PD3O and the Bregman dual Condat–Vũ algorithm with line search have a similar performance, without one algorithm being conclusively better than the other.

9 Conclusions

We presented two variants of Bregman Condat–Vũ algorithms, introduced a line search technique for the Bregman dual Condat–Vũ algorithm for equality-constrained problems, and proposed a Bregman extension to PD3O. Many open questions remain. It is unclear how to use Bregman distances in PDDY, and how to extend the line search technique to Bregman PD3O, the Bregman primal Condat–Vũ algorithm, and the more general problem (1). Moreover, in the current backtracking technique the ratio of the primal and dual stepsizes is fixed. A further improvement would be to relax this constraint [35, 1].

References

[1] D. Applegate, M. Dóaz, O. Hinder, H. Lu, M. Lubin, B. O'Donoghue, and W. Schudy. Practical large-scale linear programming using primal–dual hybrid gradient. In *Advances in Neural Information Processing Systems*, 2021.

[2] A. Auslender and M. Teboulle. Interior gradient and proximal methods for convex and conic optimization. *SIAM Journal on Optimization*, 16(3):697–725, 2006.

[3] H. H. Bauschke, J. Bolte, and M. Teboulle. A descent lemma beyond Lipschitz gradient continuity: first-order methods revisited and applications. *Mathematics of Operations Research*, 42(2):330–348, 2017.

[4] A. Beck. *First-Order Methods in Optimization*. Society for Industrial and Applied Mathematics, 2017.

[5] A. Beck and M. Teboulle. Gradient-based algorithms with applications to signal recovery. In Y. Eldar and D. Palomar, editors, *Convex Optimization in Signal Processing and Communications*. Cambridge University Press, 2009.

[6] J. Bolte, S. Sabach, S. Teboulle, and Y. Vaisbourd. First order methods beyond convexity and Lipschitz gradient continuity with applications to quadratic inverse problems. *SIAM in Optimization*, 2018.

[7] H. Brézis. *Opérateurs maximaux monotones et semi-groupes de contractions dans les espaces de Hilbert*, volume 5 of *North-Holland Mathematical Studies*. North-Holland, 1973.

[8] S. Bubeck. Convex optimization: algorithms and complexity. *Foundations and Trends Machine Learning*, 8(3–4):231–357, 2015.

[9] Y. Censor and S. A. Zenios. *Parallel Optimization: Theory, Algorithms, and Applications*. Numerical Mathematics and Scientific Computation. Oxford University Press, New York, 1997.

[10] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40:120–145, 2011.

[11] A. Chambolle and T. Pock. An introduction to continuous optimization for imaging. *Acta Numerica*, pages 161–319, 2016.

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

[13] H.-H. Chao and L. Vandenberghe. Entropic proximal operators for nonnegative trigonometric polynomials. *IEEE Transactions on Signal Processing*, 66(18):4826–4838, 2018.

[14] G. Chen and M. Teboulle. Convergence analysis of a proximal-like minimization algorithm using Bregman functions. *SIAM Journal on Optimization*, 3:538–543, 1993.

[15] P. Chen, J. Huang, and X. Zhang. A primal–dual fixed point algorithm for convex separable minimization with applications to image restoration. *Inverse Problems*, 29(2), 2013.

[16] C. Clason, D. A. Lorenz, H. Mahler, and B. Wirth. Entropic regularization of continuous optimal transport problems. *arXiv e-prints*, arXiv:1906.01333, 2019.

[17] P. L. Combettes and J.-C. Pesquet. A Douglas-Rachford splitting approach to nonsmooth convex variational signal recovery. *IEEE Journal of Selected Topics in Signal Processing*, 1(4):564–574, 2007.

[18] P. L. Combettes and J.-C. Pesquet. *Proximal splitting methods in signal processing*, pages 185–212. Springer Optimization and Its Applications. Springer New York, 2011.

[19] L. Condat. A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms. *Journal of Optimization Theory and Applications*, 158(2):460–479, 2013.

[20] D. Davis and W. Yin. A three-operator splitting scheme and its optimization applications. *arXiv e-prints*, arXiv:1504.01032, 2015.

[21] S. Diamond, E. Chu, and S. Boyd. CVXPY: a Python-embedded modeling language for convex optimization, version 0.2. `cvxpy.org`, 2014.

[22] Y. Drori, S. Sabach, and M. Teboulle. A simple algorithm for a class of nonsmooth convex-concave saddle-point problems. *Operations Research Letters*, 43(2):209–214, 2015.

[23] J. Eckstein. Nonlinear proximal point algorithms using Bregman functions, with applications to convex programming. *Mathematics of Operations Research*, 18(1):202–226, 1993.

[24] J. Eckstein and D. Bertsekas. On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. *Mathematical Programming*, 55:293–318, 1992.

[25] 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 Journal on Imaging Sciences*, 3(4):1015–1046, 2010.

[26] O. Güler. Ergodic convergence in proximal point algorithms with Bregman functions. In D.-Z. Du and J. Sun, editors, *Advances in Optimization and Approximation*, pages 155–165. Springer, 1994.

[27] B. He and X. Yuan. Convergence analysis of primal-dual algorithms for a saddle-point problem: from contraction perspective. *SIAM Journal on Imaging Sciences*, 5(1):119–149, 2012.

[28] M. Jacobs, F. Leger, W. Li, and S. Osher. Solving large-scale optimization problems with a convergence rate independent of grid size. *SIAM Journal on Numerical Analysis*, 57(3):1100–1123, 2019.

[29] X. Jiang and L. Vandenberghe. Bregman primal–dual first-order method and applications to sparse semidefinite programming. *Computational Optimization and Applications*, 81(1):127–159, 2022.

[30] N. Komodakis and J. Pesquet. Playing with duality: an overview of recent primal–dual approaches for solving large-scale optimization problems. *IEEE Signal Processing Magazine*, 32(6):31–54, 2015.

[31] P. L. Lions and B. Mercier. Splitting algorithms for the sum of two nonlinear operators. *SIAM Journal on Numerical Analysis*, 16(6):964–979, 1979.

[32] Y. Liu, Y. Xu, and W. Yin. Acceleration of primal–dual methods by preconditioning and simple subproblem procedures. *Journal of Scientific Computing*, 86(2):21, 2021.

[33] I. Loris and C. Verhoeven. On a generalization of the iterative soft-thresholding algorithm for the case of non-separable penalty. *Inverse Problems*, 27(12), 2011.

[34] H. Lu, R. M. Freund, and Y. Nesterov. Relatively smooth convex optimization by first-order methods, and applications. *SIAM Journal on Optimization*, 28(1):333–354, 2018.

[35] Y. Malitsky and T. Pock. A first-order primal-dual algorithm with linesearch. *SIAM Journal on Optimization*, 28(1):411–432, 2018.

[36] J. J. Moreau. Proximité et dualité dans un espace hilbertien. *Bull. Math. Soc. France*, 93:273–299, 1965.

[37] Y. Nesterov. *Lectures on Convex Optimization*. Springer Publishing Company, Incorporated, 2018.

[38] D. O'Conner and L. Vandenberghe. On the equivalence of the primal–dual hybrid gradient method and Douglas–Rachford splitting. *Mathematical Programming*, 2018.

[39] H. Ouyang, N. He, and A. Gray. Stochastic ADMM for nonsmooth optimization. *arXiv e-prints*, 1211.0632, 2012.

[40] N. Parikh and S. Boyd. Proximal algorithms. *Foundations and Trends in Optimization*, 1(3):123–231, 2013.

[41] T. Pock and A. Chambolle. Diagonal preconditioning for first order primal–dual algorithms in convex optimization. In *International Conference on Computer Vision*, pages 1762–1769, 2011.

[42] T. Pock, D. Cremers, H. Bischof, and A. Chambolle. An algorithm for minimizing the Mumford-Shah functional. In *International Conference on Computer Vision*, pages 1133–1140, 2009.

[43] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, 1970.

[44] A. Salim, L. Condat, K. Mishchenko, and P. Richtárik. Dualize, split, randomize: Fast nonsmooth optimization algorithms. *arXiv preprint*, arXiv:2004.0263, 2020.

[45] M. Teboulle. A simplified view of first order methods for optimization. *Mathematical Programming*, 170(1):67–96, 2018.

[46] P. Tseng. A modified forward-backward splitting method for maximal monotone mappings. *SIAM Journal on Control and Optimization*, 38(2):431–446, 2000.

[47] P. Tseng. On accelerated proximal gradient methods for convex-concave optimization. 2008.

[48] M.-L. Vladarean, Y. Malitsky, and V. Cevher. A first-order primal-dual method with adaptivity to local smoothness. In *Advances in Neural Information Processing Systems*, 2021.

[49] B. C. Vũ. A splitting algorithm for dual monotone inclusions involving cocoercive operators. *Advances in Computational Mathematics*, 38:667–681, 2013.

[50] M. Yan. A new primal-dual algorithm for minimizing the sum of three functions with a linear operator. *Journal of Scientific Computing*, 2018.

[51] E. Yazdandoost Hamedani and N. S. Aybat. A primal–dual algorithm with line search for general convex-concave saddle point problems. *SIAM Journal on Optimization*, 31(2):1299–1329, 2021.

[52] Y. Yu, P. Elango, U. Topcu, and B. Açıkmeşe. Proportional-integral projected gradient method for conic optimization. *arXiv e-prints*, arXiv:2108.10260, 2021.