Commun Nonlinear Sci Numer Simulat 140 (2025) 108374

Contents lists available at ScienceDirect

Communications in Nonlinear Science and Numerical Simulation

journal homepage: www.elsevier.com/locate/cnsns

Research paper

Strong convergence of Bregman projection algorithms for solving split feasibility problems

Liya Liu [a], Songxiao Li [b], Bing Tan [a,*]

[a] School of Mathematics and Statistics, Southwest University, Chongqing, 400715, China
[b] Department of Mathematics, Shantou University, Shantou, 515063, China

ARTICLE INFO

MSC:
47J20
47J25
47J30
47H10
68W10

Keywords:
Split feasibility problem
Bregman projection
CQ algorithm
Strong convergence
Armijo-line search

ABSTRACT

Bregman distance methods play a key role in solving problems in nonlinear analysis and optimization theory, since the Bregman distance is a useful substitute for the metric. The main purpose of this paper is to investigate two new iterative algorithms based on the Bregman distance and the Bregman projection for solving split feasibility problems in real Hilbert spaces. The algorithms are constructed around these methods: Byrne's CQ method, Polyak's gradient method, Halpern method, and hybrid projection method. The proposed methods involve inertial extrapolation terms and self-adaptive step sizes. We prove that the proposed iterations converge strongly to the Bregman projection of the initial point onto the solution set. Some numerical examples are provided to illustrate the computational effectiveness of our algorithms. The main results extend and improve the recent results related to the split feasibility problem.

1. Introduction

Let H_1 and H_2 be real Hilbert spaces equipped with inner product $\langle \cdot, \cdot \rangle$ and norm $\|\cdot\|$. Let C and Q be nonempty, closed, and convex subsets of real Hilbert spaces H_1 and H_2, respectively, and let A be a bounded operator from H_1 to H_2. Recall that the split feasibility problem (shortly, SFP) consists of finding an element x^* satisfying

$$x^* \in C \text{ such that } Ax^* \in Q. \tag{1.1}$$

The SFP in Euclidean spaces was first introduced by Censor and Elfying in 1994 [1]. Afterwards, the SFP has been employed for modeling inverse problems which arise from phase retrievals and the intensity-modulated radiation therapy [2]. The SFP has aroused numerous interests among researchers since it has been successfully applied to solve some real-world problems, such as image reconstructions, machine learning, and signal processing; see, e.g., [3–5].

Since the introduction of SFP, several types of iterative schemes have been presented for solving it. Byrne [6] introduced the so-called CQ algorithm, which is the most popular and practical algorithm that solves the SFP. For a given initial point $x_0 \in H_1$, let $\{x_n\}$ be the sequence generated by the following manner

$$x_{n+1} = P_C(x_n - \alpha A^*(I - P_Q)Ax_n), \ \ \forall n \in \mathbb{N}, \tag{1.2}$$

where I denotes the identity operator, P_C and P_Q are the metric projections onto C and Q, respectively, A^* is an adjoint operator of A, and α is a fixed real number in $\left(0, \frac{2}{\rho(A^*A)}\right)$ with $\rho(A^*A)$ being the spectral radius of A^*A. It is proved that the generated sequence

* Corresponding author.
E-mail addresses: liya42@swu.edu.cn (L. Liu), lisx@stu.edu.cn (S. Li), bingtan@swu.edu.cn (B. Tan).

https://doi.org/10.1016/j.cnsns.2024.108374
Received 11 May 2024; Received in revised form 18 August 2024; Accepted 25 September 2024
Available online 28 September 2024

$\{x_n\}$ converges weakly to a solution of the SFP (1.1). The CQ algorithm can be regarded as a special case of the gradient projection method, since the SFP is equivalent to the constrained convex minimization problem $\min_{x\in C} \frac{1}{2}\|(I-P_Q)Ax\|^2$.

The extensions of CQ algorithms have been studied by many authors; see, e.g., [7–10]. It is noted that the selection of the step size α in (1.2) depends on any prior information of $\rho(A^*A)$, which in general is not easy to compute in practice. To overcome this drawback, several self-adaptive step size methods have been established to determine the step size α in (1.2). Among these research works, López et al. [8] studied an ingenious dynamic step size method such that the convergence of (1.2) is guaranteed without calculating the spectral norm of the operator A, that is

$$\alpha_n := \frac{\rho_n \|(I-P_Q)Ax_n\|^2}{\|A^*(I-P_Q)Ax_n\|^2}, \tag{1.3}$$

where $\rho_n \in (0,4)$. Subsequently, Anh et al. [5] proposed an alternative way that is to select the step size α_n by

$$\alpha_n := \frac{\rho_n}{\max\{1, \|A^*(I-P_Q)Ax_n\|\}}, \tag{1.4}$$

where $\{\rho_n\}$ is a positive sequence satisfying $\lim_{n\to\infty} \rho_n = 0$ and $\sum_{n=0}^{\infty} \rho_n = \infty$. It is noted that the implementation of the dynamic step size α_n in both (1.3) and (1.4) does not need any prior information of the operator norm $\|A\|$ or its estimation. An alternative way is to select the step size α_n by using the Armijo line search rule; see [9,10]. Precisely, given $\mu \in (0,1)$ and $\nu \in (0,1)$, let $\alpha_n := \mu\nu^{\tau_n}$ with τ_n being the smallest nonnegative integer τ satisfying

$$\mu\nu^{\tau}\|A^*(I-P_Q)Ax_n - A^*(I-P_Q)Ay_n\| \le \lambda\|x_n - y_n\|, \ \ \forall \lambda \in (0,1), \tag{1.5}$$

where $y_n = P_C(x_n - \alpha_n A^*(I-P_Q)Ax_n)$. The mentioned self-adaptive step size methods (1.3)–(1.5) have been studied extensively and generalized in various ways; see, e.g., [11,12].

It is well known that the gradient method is one of the simplest iterative methods for solving unconstrained minimization problems. The convergence of such method has been studied by a number of authors. Among them, Polyak modified the gradient method by adopting a new adaptive way of determining the step size sequence; see [13]. The algorithm is of the form: For an arbitrary starting point $x_0 \in H_1$, let $\{x_n\}$ be the sequence generated by

$$x_{n+1} = x_n - \alpha_n \nabla f(x_n),$$

where $f : H_1 \to \mathbb{R}$ is a convex and differentiable function and $\alpha_n := \rho_n \frac{f(x_n)-f^*}{\|\nabla f(x_n)\|^2}$ with $0 < \epsilon \le \rho_n \le 2-\epsilon$ and $f^* := \min f(x)$. It is noted that the SFP (1.1) can be reformulated as the unconstraint minimization problem:

$$\min f(x) := \frac{1}{2}\|(I-P_C)x\|^2 + \frac{1}{2}\|(I-P_Q)Ax\|^2.$$

This motivates that Polyak's gradient method in the unconstraint minimization can be applicable for solving the SFP (1.1); see [14]. The corresponding iterative sequence takes the following form: For any fixed initial point $x_0 \in H_1$,

$$x_{n+1} = x_n - \alpha_n[(I-P_C)x_n + A^*(I-P_Q)Ax_n],$$

where

$$\alpha_n := \frac{\rho_n(\|(I-P_C)x_n\|^2 + \|(I-P_Q)Ax_n\|^2)}{2\|(I-P_C)x_n + A^*(I-P_Q)Ax_n\|^2}. \tag{1.6}$$

In this paper, we are concerned with a continuation of study on the CQ algorithm and Polyak's gradient algorithm. We mention here that the selection of the step sizes in (1.3)–(1.6) requires to calculate the metric projections P_C and P_Q. In general, the computation of the metric projection onto a nonempty closed and convex subset is expensive or even impossible because there is no explicit formula. In this sense, the execution efficiency of step size methods (1.3)–(1.6) may be seriously affected, since the evaluations of projections P_C and P_Q are involved therein. In particular, the choice of the step size α_n in (1.5) requires to compute P_C and P_Q several times which may cost much. This naturally motivates the following research direction: *Can we design efficient and simple step sizes, which are independent of the operator norm and different from the step sizes mentioned above?*

There are many real-world problems in management and engineering that arise in infinite-dimensional spaces; see, e.g., [15–17]. It is emphasized that the strong convergence is often much more desirable than the weak convergence in infinite-dimensional spaces. The study of turning the weak convergence into the strong convergence is an interesting research, which has attracted much attention of authors; see [15,17]. However, the CQ algorithm and Polyak's gradient algorithm may fail to converge strongly in the infinite-dimensional case [10,12,18]. For the sake of achieving the strong convergence, we recall some methods that actually enforce the strong convergence property. Among them, a more classical and simple method seems to be the Halpern method, which has been used to guarantee the strong convergence of the algorithm for solving SFPs; see [19,20]. Another method is the hybrid projection method, wherein the strong convergence property is forced by adding the metric projection of a point x_0 onto the intersection of two associated half-spaces at each iteration. It was proved that the generated sequence starting with x_0 converges strongly to the solution set of the SFP, which is closest to x_0; see [21]. For more theoretical results related to the hybrid projection method, see [22]. Motivated by this research trend, we aim to modify the CQ method and Polyak's gradient method so as to establish the strong convergence in infinite-dimensional cases.

The Bregman distance is an elegant and effective distance function introduced by Bregman in 1976 [23] (see Section 2 for a definition). It generalizes a wide range of measures such as the squared Euclidean distance, the Itakura–Saito divergence, and the

Kullback–Leibler divergence. The Bregman distance, which capable of exploring the nonlinear correlations of data features, has found applications in various areas including machine learning, computational geometry, operations research, and information theory; see, e.g., [24,25]. The Bregman distance, which is derived from the various choices of Bregman functions, can be regarded as a useful and flexible substitute for a usual norm distance. In this sense, it opened a growing area of research in which the Bregman distance can be applied in the process of designing and analyzing iterative algorithms; see, e.g., [22,26]. Since the Bregman distance is asymmetric and does not satisfy the triangle inequality, one has that the traditional metric method cannot be applied in the convergence analysis of algorithms in the framework of Bregman distance. In general, different convergence proofs of iterative sequences are developed depending on the choice of Bregman functions. It is emphasized that the lack of research on the study of Bregman distance algorithms for solving SFPs calls for the further investigation.

The inertial technique was first proposed by Polyak in 1964 [27]. This technique can be regarded as a two-step iterative method based on the discrete version of the second-order dissipative dynamical system. It is known for its efficiency in speeding up the convergence properties of iterative processes. The main feature of this technique is that the next iterate is updated by making use of the last two iterates. In recent years, it is desirable to work with algorithms with a high convergence speed. One way to achieve this is by incorporating the inertial technique into some known methods. As can be seen in many earlier research, there has been an increasing interest in studying the influence of the inertial technique on the convergence performance; see, e.g., [18,28–30]. Recently, various inertial type algorithms were proposed and analyzed based on the norm distance, for example, the inertial relaxed CQ algorithm [18], the inertial shrinking projection algorithm [28], the inertial extragradient algorithm [29], and the inertial proximal point algorithm [30]. Our concern in this paper is focused on modifying the inertial technique by using the Bregman distance instead of the norm distance, which is quite different from the earlier ones.

Inspired and motivated by the above results and the ongoing research interest in these directions, the purpose of this paper is to design two iterative algorithms for solving the SFP that employ Bregman distances and Bregman projections. The step sizes are selected adaptively by adopting the Armijo-like line search method. The strong convergence results of the algorithms are established under mild conditions. Some preliminary numerical results are given to illustrate the performance and the efficiency of our proposed algorithms. Results obtained in this paper extend and improve the previously known results in this field. Our main contributions of this paper are fourfold.

(i) By investigating and applying different analytical methods, similar previous results are extended from the case of norm distances to the case of Bregman distances.
(ii) Neither the prior information of the bounded linear operator norm nor any extra projection step is required to update step sizes. This makes our method more practical and cheaper to implement than some related methods.
(iii) Inertial terms are added to accelerate the convergence speed of the proposed algorithms.
(iv) Our iterative algorithms are shown to converge strongly to a solution of the SFP, which is an important factor to consider in an infinite-dimensional space.

The rest of the paper is organized as follows. In Section 2, we recall some definitions and lemmas, which will be used in the proof of main results. In Section 3, we propose the algorithms and give their strong convergence analysis. In Section 4, some numerical experiments are provided to illustrate the efficiency and the performance of our proposed methods. Finally, the paper is concluded with a brief summary in Section 5.

2. Preliminaries

In this section, we state some necessary theoretical background material. Let H be a real Hilbert space with its inner product $\langle \cdot, \cdot \rangle$ and the norm $\|\cdot\|$. Let $\mathbb{N}$ and $\mathbb{R}$ denote the set of all positive integers and the set of all real numbers, respectively.

Definition 1. Let H_1 and H_2 be two Hilbert spaces and let $A : H_1 \to H_2$ be a bounded operator. An operator $A^* : H_2 \to H_1$ is called the adjoint operator of A, if

$$\langle Ax, y\rangle = \langle x, A^* y\rangle, \quad \forall x \in H_1,\ y \in H_2.$$

We denote the domain of $f : H \to \mathbb{R}$ by $\mathrm{dom} f = \{x \in H : f(x) < \infty\}$. Let $f : H \to \mathbb{R}$ be a lower semi-continuous, convex, and differentiable function with $\mathrm{dom} f \neq \emptyset$.

Definition 2. Recall that the function $f : H \to \mathbb{R}$ is said to be

(i) Gâteaux differentiable at x if there is a gradient of f at x, denoted by $\nabla f(x)$, such that the limit $\langle \nabla f(x), w\rangle = \lim_{\lambda\to 0} \frac{f(x+\lambda w)-f(x)}{\lambda}$ exists for all $w \in H$.
(ii) Fréchet differentiable at x if the limit in (i) is attained uniformly for $\|w\| = 1$.
(iii) uniformly Fréchet differentiable on a subset C of H, if the limit in (i) is attained uniformly for all $x \in C$ and $\|w\| = 1$.

Similarly, the domain of ∇f is denoted by $\mathrm{dom}\nabla f$. It is emphasized that ∇f is uniformly continuous on bounded subsets of H if f is uniformly Fréchet differentiable and bounded on bounded subsets of H; see [31].

Definition 3 (*[23,32]*)**.** Given a strictly convex and differentiable function f with its gradient ∇f, the Bregman distance D_f : $\mathrm{dom} f \times \mathrm{dom}\nabla f \to [0, \infty)$ with respect to f is defined by $D_f(x, y) = f(x) - f(y) - \langle \nabla f(y), x - y\rangle,\ \forall x \in \mathrm{dom} f,\ y \in \mathrm{dom}\nabla f$.

It is noted that the Bregman distance is in general not a metric, since it does not necessarily satisfy the triangle inequality [33], but the Bregman distance fulfills various geometric properties which make it a substitute for a measure of the norm distance [34] (see below).

The two point identity: For any $x, y \in \mathrm{dom}\nabla f$, $D_f(x, y) + D_f(y, x) = \langle \nabla f(x) - \nabla f(y), x - y\rangle$.

The three point identity: For any $x \in \mathrm{dom} f$ and $y, z \in \mathrm{dom}\nabla f$,

$$D_f(x, y) = D_f(x, z) - D_f(y, z) + \langle \nabla f(z) - \nabla f(y), x - y\rangle. \tag{2.1}$$

Definition 4 (*[23]*). Denote by Π_C^f the Bregman projection with respect to the Bregman function f of a point $x \in \mathrm{dom}\nabla f$ onto a set $C \subset \mathrm{dom} f$, which is defined as the point in C such that $D_f(\Pi_C^f x, x) \le D_f(y, x)$, $\forall y \in C$.

Remark 1. When $C \subset \mathrm{dom} f$ is a nonempty, closed, and convex set, the point $\Pi_C^f x$ in Definition 4 is unique.

Lemma 1. *The properties of the Bregman projection are summarized as follows; see [32,35].*

(i) $\langle \nabla f(\Pi_C^f x) - \nabla f(x), y - \Pi_C^f x\rangle \ge 0$, $\forall x \in \mathrm{dom}\nabla f$, $y \in C$;
(ii) $D_f(y, \Pi_C^f x) + D_f(\Pi_C^f x, x) \le D_f(y, x)$, $\forall x \in \mathrm{dom}\nabla f$, $y \in C$.

Definition 5. We recall that a function $f : H \to \mathbb{R}$ is said to be strongly convex with a constant $\delta > 0$ if

$$f(y) - f(x) \ge \langle \nabla f(x), y - x\rangle + \frac{\delta}{2}\|y - x\|^2, \ \forall x \in \mathrm{dom}\nabla f, \ y \in \mathrm{dom} f.$$

The Bregman distance D_f with respect to the δ-strongly convex function f can be characterized by the inequality of $D_f(y, x) \ge \frac{\delta}{2}\|y - x\|^2$, $\forall x \in \mathrm{dom}\nabla f$, $y \in \mathrm{dom} f$; see [26]. For any $x \in \mathrm{dom}\nabla f$ and $y \in \mathrm{dom} f$, the strong convexity of f implies that $D_f(y, x) = 0$ is equivalent to $x = y$.

Lemma 2 (*[22]*). *Let $f : H \to \mathbb{R}$ be strongly convex, Fréchet differentiable, and bounded on bounded subsets of H. Given two sequences $\{x_n\}$ and $\{y_n\}$ in H, we have that*

$$\lim_{n\to\infty} D_f(x_n, y_n) = 0 \Rightarrow \lim_{n\to\infty} \|x_n - y_n\| = 0 \Rightarrow \lim_{n\to\infty} \|\nabla f(x_n) - \nabla f(y_n)\| = 0.$$

Definition 6. The Fenchel conjugate function of $f : H \to \mathbb{R}$ is the convex function $f^* : H \to (-\infty, +\infty]$ defined by

$$f^*(x^*) := \sup_{x\in H}\{\langle x^*, x\rangle - f(x)\}.$$

Remark 2. Let $f^* : H \to (-\infty, +\infty]$ be a Fenchel conjugate function of $f : H \to \mathbb{R}$. Then

$$\nabla f^*(\nabla f(x)) = x, \forall x \in dom\nabla f \text{ and } \nabla f(\nabla f^*(x^*)) = x^*, \forall x^* \in dom\nabla f^*.$$

Definition 7 (*[36]*). The function $f : H \to \mathbb{R}$ is said to be Legendre if it satisfies

(i) $\mathrm{dom}\nabla f \ne \emptyset$ and ∂f is single-valued on its domain;
(ii) $\mathrm{dom}\nabla f^* \ne \emptyset$ and ∂f^* is single-valued on its domain.

Given a Legendre function $f : H \to \mathbb{R}$, define a function $V_f : \mathrm{dom} f^* \times \mathrm{dom} f \to [0, +\infty)$ associated with f by

$$V_f(\eta, x) := f(x) - \langle \eta, x\rangle + f^*(\eta), \ \forall \eta \in \mathrm{dom} f^*, \ x \in \mathrm{dom} f.$$

Some properties of the function V_f can be summarized as follows; see [37].

(i) $V_f(\nabla f(y), x) = D_f(x, y)$, $\forall y \in \mathrm{dom}\nabla f$, $x \in \mathrm{dom} f$;
(i) $V_f(\eta, x) + \langle \zeta, \nabla f^*(\eta) - x\rangle \le V_f(\eta + \zeta, x)$, $\forall \eta \in \mathrm{dom} f^*, \zeta \in \mathrm{dom} f^*$, $x \in \mathrm{dom} f$;
(iii) V_f is nonnegative and convex in its first variable.

Since V_f is convex in the first variable, then

$$D_f\left(x, \nabla f^*\left(\sum_{i=1}^N \lambda_i \nabla f(y_i)\right)\right) \le \sum_{i=1}^N \lambda_i D_f(x, y_i), \ \forall x \in \mathrm{dom} f, \tag{2.2}$$

where $\{y_i\}_{i=1}^N \subset H$ and $\{\lambda_i\}_{i=1}^N \subset [0, 1]$ satisfies that $\sum_{i=1}^N \lambda_i = 1$.

Lemma 3 (*[38]*). *Let $\{a_n\}$, $\{\lambda_n\}$, $\{\beta_n\}$, and $\{b_n\}$ be nonnegative real sequences such that for each $n \ge n_0$ (where n_0 is a positive integer), $a_{n+1} \le (1 - \lambda_n - \beta_n)a_n + \beta_n a_{n-1} + \lambda_n b_n$, $n \ge 1$, where $\sum_{n=1}^\infty \lambda_n = \infty$, $\{\beta_n\} \subset [0, \frac{1}{2}]$, and $\limsup_{n\to\infty} b_n \le 0$. Then $\lim_{n\to\infty} a_n = 0$.*

Lemma 4 ([39]). *Let $\{a_n\}$ be a nonnegative real sequence that does not decrease at infinity. Then there exists an increasing sequence $\{m_k\} \subset \mathbb{N}$ such that $m_k \to \infty$ as $k \to \infty$ and the following properties are satisfied by all (sufficiently large) numbers $k \in \mathbb{N}$: $a_{m_k} \leq a_{m_k+1}$ and $a_k \leq a_{m_k+1}$. In fact, $m_k = \max\{j \leq k : a_j \leq a_{j+1}\}$.*

3. Algorithms and convergence analysis

In this section, we design two iterative algorithms based on Bregman distances and Bregman projections for solving SFPs and analyze their convergence properties.

3.1. First type of Bregman projection algorithm

In this subsection, we present our first algorithm. We begin with the following standard assumptions under which we establish the strong convergence result.

($A1$) $A : H_1 \to H_2$ is a bounded operator with an adjoint $A^* : H_2 \to H_1$.
($A2$) The function $f : H_1 \to \mathbb{R}$ is δ_1-strongly convex, Legendre, which is bounded and uniformly Fréchet differentiable on bounded subsets of H_1.
($A3$) The function $g : H_2 \to \mathbb{R}$ is δ_2-strongly convex, Legendre, which is bounded and uniformly Fréchet differentiable on bounded subsets of H_2.
($A4$) The solution set $\Gamma := \{x \in C : Ax \in Q\}$ is nonempty.
($A5$) The nonnegative real sequence $\{\eta_n\} \subset (0,1)$ satisfies that $\lim_{n\to\infty} \eta_n = 0$.
($A6$) The real sequence $\{\lambda_n\} \subset (0,1)$ satisfies that $\sum_{n=1}^{\infty} \lambda_n = \infty$ and $\lim_{n\to\infty} \lambda_n = 0$.

Below is the Polyak's gradient-based algorithm by combining Bregman techniques with the Halpern method. The algorithm is described as follows.

Algorithm 1 (Halpern-type Polyak's gradient algorithm with the Bregman distance)

Step 1: Let $x_0, x_1 \in H_1$ be arbitrary. Take $\nu \in (0,1), \tau \in (0,1)$, $\sigma \in (0,\infty)$, and $\mu \in (0,\infty)$. Set $n = 1$.
Step 2: Compute $y_n = \nabla f^*(\nabla f(x_n) + \beta_n(\nabla f(x_{n-1}) - \nabla f(x_n)))$, where β_n is defined by

$$\beta_n = \begin{cases} \min\{\sigma, \frac{\eta_n}{D_f(x_n,x_{n-1})}\}, & \text{if } x_{n-1} \neq x_n, \\ \sigma, & \text{otherwise.} \end{cases} \tag{3.1}$$

Step 3: Compute

$$\begin{aligned} p_n &= \nabla f^*(\nabla f(y_n) - \nabla f(\Pi_C^f y_n)), \\ q_n &= \nabla g^*(A^*(\nabla g(Ay_n) - \nabla g(\Pi_Q^g Ay_n))), \\ z_n &= \nabla f^*(\nabla f(y_n) - \alpha_n(\nabla f(p_n) + \nabla g(q_n))), \end{aligned} \tag{3.2}$$

where $\alpha_n := \mu\nu^{\kappa_n}$ with κ_n being the smallest nonnegative integer κ satisfying

$$\mu\nu^{\kappa}(D_f(z_n, y_n) + D_g(Az_n, Ay_n)) \leq \tau D_f(z_n, y_n). \tag{3.3}$$

If $y_n = z_n$, then stop and y_n is a solution of the SFP. Otherwise, go to ***Step 4***.
Step 4: Compute

$$x_{n+1} = \nabla f^*(\lambda_n \nabla f(x_0) + (1-\lambda_n)\nabla f(z_n)). \tag{3.4}$$

Update $n := n+1$ and go to ***Step 2***.

Remark 3.

(i) The inertial term improves the computational efficiency of Algorithm 1.
(ii) Our suggested algorithm embeds a new Armijo-type line search criterion (3.3) that allows it to work adaptively and cheaply without any extra Bregman projection step.

Now we will prove some lemmas that will be used in our convergence analysis. The following lemma is regarding the step size property of Algorithm 1.

Lemma 5. *The Armijo-type line search criterion (3.3) is well-defined. Moreover, $\alpha_n \in (0,\mu]$, $\forall n \in \mathbb{N}$.*

Proof. (1) Suppose that $D_f(z_n, y_n) + D_g(Az_n, Ay_n) = 0$. In this case, we obtain that $\kappa_n = 0$ holds.

(2) Suppose that $D_f(z_n, y_n) + D_g(Az_n, Ay_n) \neq 0$. In this case, we additionally suppose that $D_f(z_n, y_n) = 0$. Since the Bregman distance is always nonnegative, one finds that $z_n = y_n$ and so $Az_n = Ay_n$. This gives that $D_g(Az_n, Ay_n) = 0$. Therefore, we conclude that $D_f(z_n, y_n) + D_g(A(z_n), A(y_n)) = 0$, which is a contradiction. Hence, $D_f(z_n, y_n) \neq 0$. We assume that the contrary of (3.3) holds for any integer κ, that is,

$$\mu\nu^{\kappa}(D_f(z_n, y_n) + D_g(Az_n, Ay_n)) > \tau D_f(z_n, y_n). \tag{3.5}$$

Since $\mu \in (0, \infty)$ and $\nu \in (0, 1)$, it follows from (3.5) that

$$0 = \lim_{\kappa\to\infty} \mu\nu^{\kappa}(D_f(z_n, y_n) + D_g(Az_n, Ay_n)) > \tau D_f(z_n, y_n) > 0.$$

This yields a contradiction. Therefore, there exists a finite nonnegative integer $\tilde{\kappa} \in \mathbb{N}$ such that $\mu\nu^{\tilde{\kappa}}(D_f(z_n, y_n) + D_g(Az_n, Ay_n)) \leq \tau D_f(z_n, y_n)$. Then (3.3) holds. This implies that $\alpha_n \in (0, \mu]$, $\forall n \in \mathbb{N}$. The proof is completed. □

Remark 4. It is worth mentioning that the proof of Lemma 5 does not use the conditions that A is linear and $\|A\| \neq 0$, which is an improvement to the previous results in the literatures; see, e.g., [6].

Lemma 6. *If $y_n = z_n$ holds for some integer n, then y_n is a solution of the SFP.*

Proof. We assume that $y_n = z_n$. By using (3.2) and Remark 2, the assumption can be rewritten as

$$\nabla f(y_n) = \nabla f(y_n) - \alpha_n(\nabla f(y_n) - \nabla f(\Pi_C^f y_n) + A^*(\nabla g(Ay_n) - \nabla g(\Pi_Q^g Ay_n))). \tag{3.6}$$

It follows from (3.6) and the obtained result of $\alpha_n \in (0, \mu]$, $\forall n \in \mathbb{N}$ in Lemma 5 that

$$0 = \nabla f(y_n) - \nabla f(\Pi_C^f y_n) + A^*(\nabla g(Ay_n) - \nabla g(\Pi_Q^g Ay_n)). \tag{3.7}$$

Given $\hat{x} \in \Gamma$, it holds that $\hat{x} \in C$ and $A(\hat{x}) \in Q$. By (3.7), we find that

$$\begin{aligned} 0 =& \langle \nabla f(y_n) - \nabla f(\Pi_C^f y_n) + A^*(\nabla g(Ay_n) - \nabla g(\Pi_Q^g Ay_n)), y_n - \hat{x}\rangle \\ =& \langle \nabla f(y_n) - \nabla f(\Pi_C^f y_n), y_n - \hat{x}\rangle + \langle \nabla g(Ay_n) - \nabla g(\Pi_Q^g Ay_n), Ay_n - A\hat{x}\rangle. \end{aligned} \tag{3.8}$$

By using the three point identity (2.1) and Lemma 1(ii), we have

$$\langle \nabla f(y_n) - \nabla f(\Pi_C^f y_n), y_n - \hat{x}\rangle = D_f(\hat{x}, y_n) + D_f(y_n, \Pi_C^f y_n) - D_f(\hat{x}, \Pi_C^f y_n) \geq D_f(y_n, \Pi_C^f y_n) \geq 0 \tag{3.9}$$

and

$$\langle \nabla g(Ay_n) - \nabla g(\Pi_Q^g Ay_n), Ay_n - A\hat{x}\rangle = D_g(A\hat{x}, Ay_n) + D_g(Ay_n, \Pi_Q^g Ay_n) - D_g(A\hat{x}, \Pi_Q^g Ay_n) \geq D_g(Ay_n, \Pi_Q^g Ay_n) \geq 0. \tag{3.10}$$

By considering (3.8), (3.9), and (3.10), we obtain that

$$\langle \nabla f(y_n) - \nabla f(\Pi_C^f y_n), y_n - \hat{x}\rangle = \langle \nabla g(Ay_n) - \nabla g(\Pi_Q^g Ay_n), Ay_n - A\hat{x}\rangle = 0. \tag{3.11}$$

Putting (3.9), (3.10), and (3.11) together, one observes that

$$D_f(y_n, \Pi_C^f y_n) = D_g(Ay_n, \Pi_Q^g Ay_n) = 0. \tag{3.12}$$

Hence, (3.12) asserts that $\|y_n - \Pi_C^f y_n\| = 0$ and $\|Ay_n - \Pi_Q^g Ay_n\| = 0$. This further implies that $y_n \in C$ and $Ay_n \in Q$, that is, $y_n \in \Gamma$. This completes the proof. □

Remark 5. Lemma 6 implies that if the iterative sequence generated by Algorithm 1 terminates within finite steps, then the current iterative point must be a solution of the SFP (1.1). Without loss of generality, we assume that Algorithm 1 generates an infinite iterative sequence in the following convergence analysis.

Lemma 7. *Suppose that conditions (A1)-(A6) hold. Let $\{z_n\}$ and $\{y_n\}$ be the sequences generated by Algorithm 1. Then for any $\hat{x} \in \Gamma$, we obtain that*

$$\begin{aligned} & D_f(\hat{x}, z_n) \\ \leq & D_f(\hat{x}, y_n) - D_f(z_n, y_n) + \tau D_f(z_n, y_n) - \alpha_n[D_f(z_n, \Pi_C^f y_n) + D_f(\Pi_C^f y_n, y_n) + D_g(Az_n, \Pi_Q^g Ay_n) + D_g(\Pi_Q^g Ay_n, Ay_n)]. \end{aligned} \tag{3.13}$$

Proof. By virtue of the three point identity (2.1), we deduce that

$$D_f(\hat{x}, z_n) = D_f(\hat{x}, y_n) - D_f(z_n, y_n) + \langle \nabla f(y_n) - \nabla f(z_n), \hat{x} - z_n\rangle, \tag{3.14}$$

$$\langle \nabla f(y_n) - \nabla f(\Pi_C^f y_n), \Pi_C^f y_n - z_n\rangle = -D_f(z_n, \Pi_C^f y_n) + D_f(z_n, y_n) - D_f(\Pi_C^f y_n, y_n), \tag{3.15}$$

and

$$\langle\nabla g(Ay_n)-\nabla g(\Pi_Q^g Ay_n),\Pi_Q^g Ay_n-Az_n\rangle=-D_g(Az_n,\Pi_Q^g Ay_n)+D_g(Az_n,Ay_n)-D_g(\Pi_Q^g Ay_n,Ay_n). \tag{3.16}$$

On the other hand, Lemma 1(i) asserts that

$$\langle\nabla f(y_n)-\nabla f(\Pi_C^f y_n),\hat{x}-\Pi_C^f y_n\rangle\le 0 \tag{3.17}$$

and

$$\langle\nabla g(Ay_n)-\nabla g(\Pi_Q^g Ay_n),A\hat{x}-\Pi_Q^g Ay_n\rangle\le 0. \tag{3.18}$$

By the definitions of p_n and q_n, it follows that

$$\begin{aligned}\langle\nabla f(p_n),\hat{x}-z_n\rangle=&\langle\nabla f(y_n)-\nabla f(\Pi_C^f y_n),\hat{x}-z_n\rangle\\=&\langle\nabla f(y_n)-\nabla f(\Pi_C^f y_n),\hat{x}-\Pi_C^f y_n\rangle+\langle\nabla f(y_n)-\nabla f(\Pi_C^f y_n),\Pi_C^f y_n-z_n\rangle.\end{aligned} \tag{3.19}$$

and

$$\begin{aligned}\langle\nabla g(q_n),\hat{x}-z_n\rangle=&\langle\nabla g(Ay_n)-\nabla g(\Pi_Q^g Ay_n),A\hat{x}-Az_n\rangle\\=&\langle\nabla g(Ay_n)-\nabla g(\Pi_Q^g Ay_n),A\hat{x}-\Pi_Q^g Ay_n\rangle+\langle\nabla g(Ay_n)-\nabla g(\Pi_Q^g Ay_n),\Pi_Q^g Ay_n-Az_n\rangle.\end{aligned} \tag{3.20}$$

By using the definition of z_n, one sees that

$$\langle\nabla f(y_n)-\nabla f(z_n),\hat{x}-z_n\rangle=\alpha_n\langle\nabla f(p_n),\hat{x}-z_n\rangle+\alpha_n\langle\nabla g(q_n),\hat{x}-z_n\rangle. \tag{3.21}$$

By substituting (3.15)–(3.21) into (3.14), we deduce that

$$\begin{aligned}D_f(\hat{x},z_n)\le&D_f(\hat{x},y_n)-D_f(z_n,y_n)+\alpha_n[D_f(z_n,y_n)+D_g(Az_n,Ay_n)]\\&-\alpha_n[D_f(z_n,\Pi_C^f y_n)+D_f(\Pi_C^f y_n,y_n)+D_g(Az_n,\Pi_Q^g Ay_n)+D_g(\Pi_Q^g Ay_n,Ay_n)].\end{aligned}$$

This completes the proof. □

We are now in a position to prove the strong convergence result of Algorithm 1.

Theorem 1. *Let conditions (A1)-(A6) hold. Then the sequence $\{x_n\}$ generated by Algorithm 1 converges strongly to $x^\dagger\in\Gamma$ with $x^\dagger=\Pi_\Gamma^f(x_0)$.*

Proof. Given $\hat{x}\in\Gamma$, by combining the definition of $\{y_n\}$ with (2.2), one deduces that

$$D_f(\hat{x},y_n)=D_f(\hat{x},\nabla f^*((1-\beta_n)\nabla f(x_n)+\beta_n\nabla f(x_{n-1})))\le(1-\beta_n)D_f(\hat{x},x_n)+\beta_n D_f(\hat{x},x_{n-1}). \tag{3.22}$$

On the basis of (3.13) and (3.22), we obtain that

$$D_f(\hat{x},z_n)\le(1-\beta_n)D_f(\hat{x},x_n)+\beta_n D_f(\hat{x},x_{n-1}). \tag{3.23}$$

This implies that

$$D_f(\hat{x},z_n)\le\max\{D_f(\hat{x},x_n),D_f(\hat{x},x_{n-1})\}. \tag{3.24}$$

By virtue of (2.2) and (3.4), we obtain that

$$D_f(\hat{x},x_{n+1})\le\lambda_n D_f(\hat{x},x_0)+(1-\lambda_n)D_f(\hat{x},z_n). \tag{3.25}$$

By applying (3.25) with (3.24), one finds that

$$D_f(\hat{x},x_{n+1})\le\max\{D_f(\hat{x},x_0),D_f(\hat{x},z_n)\}\le\max\{D_f(\hat{x},x_0),D_f(\hat{x},x_n),D_f(\hat{x},x_{n-1})\}\le\cdots\le\max\{D_f(x,x_0),D_f(\hat{x},x_0)\}.$$

This implies that $\{D_f(\hat{x},x_n)\}$ is bounded. By noticing the relation $D_f(x,y)\ge\frac{\delta_1}{2}\|x-y\|^2$ $(\forall x\in\mathrm{dom}f,y\in\mathrm{dom}\nabla f)$, it follows that $\{x_n\}$ is also bounded and so is the sequence $\{Ax_n\}$. By using (3.13), (3.22), and (3.25), we obtain that

$$\begin{aligned}&(1-\lambda_n)\left\{(1-\tau)D_f(z_n,y_n)+\alpha_n[D_f(z_n,\Pi_C^f y_n)+D_f(\Pi_C^f y_n,y_n)+D_g(Az_n,\Pi_Q^g Ay_n)+D_g(\Pi_Q^g Ay_n,Ay_n)]\right\}\\&\le\lambda_n D_f(\hat{x},x_0)+D_f(\hat{x},x_n)-D_f(\hat{x},x_{n+1})-\beta_n[D_f(\hat{x},x_n)-D_f(\hat{x},x_{n-1})].\end{aligned} \tag{3.26}$$

By applying the property of $V_f(\cdot,\cdot)$ with (3.23), one deduces that

$$\begin{aligned} D_f(x^\dagger, x_{n+1}) =& V_f(\lambda_n \nabla f(x_0) + (1-\lambda_n)\nabla f(z_n), x^\dagger) \\ \le& V_f(\lambda_n \nabla f(x_0) + (1-\lambda_n)\nabla f(z_n) - \lambda_n(\nabla f(x_0) - \nabla f(x^\dagger)), x^\dagger) \\ &+ \langle \lambda_n(\nabla f(x_0) - \nabla f(x^\dagger)), \nabla f^*(\lambda_n \nabla f(x_0) + (1-\lambda_n)\nabla f(z_n)) - x^\dagger\rangle \\ =& V_f(\lambda_n \nabla f(x^\dagger) + (1-\lambda_n)\nabla f(z_n), x^\dagger) + \lambda_n\langle \nabla f(x_0) - \nabla f(x^\dagger), x_{n+1} - x^\dagger\rangle \\ =& D_f(x^\dagger, \nabla f^*(\lambda_n \nabla f(x^\dagger) + (1-\lambda_n)\nabla f(z_n))) + \lambda_n\langle \nabla f(x_0) - \nabla f(x^\dagger), x_{n+1} - x^\dagger\rangle \\ \le& (1-\lambda_n) D_f(x^\dagger, z_n) + \lambda_n\langle \nabla f(x_0) - \nabla f(x^\dagger), x_{n+1} - x^\dagger\rangle \\ \le& (1-\lambda_n) D_f(x^\dagger, x_n) - (1-\lambda_n)\beta_n(D_f(x^\dagger, x_n) - D_f(x^\dagger, x_{n-1})) + \lambda_n\langle \nabla f(x_0) - \nabla f(x^\dagger), x_{n+1} - x^\dagger\rangle \\ =& [1-\lambda_n - (1-\lambda_n)\beta_n] D_f(x^\dagger, x_n) + (1-\lambda_n)\beta_n D_f(x^\dagger, x_{n-1}) + \lambda_n\langle \nabla f(x_0) - \nabla f(x^\dagger), x_{n+1} - x^\dagger\rangle. \end{aligned} \tag{3.27}$$

Now we consider two possible cases in proving that every weak cluster point of $\{x_n\}$ belongs to the solution set Γ.

Case 1. Assume that there exists $N \in \mathbb{N}$ such that $D_f(\hat{x}, x_{n+1}) \le D_f(\hat{x}, x_n)$ for any $n \ge N$. In this case, we derive that $\{D_f(\hat{x}, x_n)\}$ is convergent and

$$\lim_{n\to\infty}(D_f(\hat{x}, x_n) - D_f(\hat{x}, x_{n+1})) = 0. \tag{3.28}$$

By using (3.26), (3.28), and Condition (A2), we obtain that

$$\lim_{n\to\infty} D_f(z_n, y_n) = \lim_{n\to\infty} D_f(\Pi_C^f(y_n), y_n) = \lim_{n\to\infty} D_g(\Pi_Q^g(A(y_n)), A(y_n)) = 0. \tag{3.29}$$

By considering the definition of $\{y_n\}$ and combining (2.2) with (3.1), one deduces that

$$\begin{aligned} D_f(x_n, y_n) =& D_f(x_n, \nabla f^*((1-\beta_n)\nabla f(x_n) + \beta_n \nabla f(x_{n-1}))) \\ \le& (1-\beta_n) D_f(x_n, x_n) + \beta_n D_f(x_n, x_{n-1}) = \beta_n D_f(x_n, x_{n-1}) \le \eta_n. \end{aligned} \tag{3.30}$$

By using condition $(A5)$, it follows from (3.30) that

$$\lim_{n\to\infty} D_f(x_n, y_n) = 0. \tag{3.31}$$

By using (2.2) and (3.4), we have that

$$D_f(z_n, x_{n+1}) \le \lambda_n D_f(z_n, x_0) + (1-\lambda_n) D_f(z_n, z_n) \to 0, \text{ as } n \to \infty. \tag{3.32}$$

By using Lemma 2, (3.29), (3.31), and (3.32), we obtain that

$$\lim_{n\to\infty}\|z_n - y_n\| = \lim_{n\to\infty}\|x_n - y_n\| = \lim_{n\to\infty}\|z_n - x_{n+1}\| = 0. \tag{3.33}$$

It follows from (3.33) that

$$\lim_{n\to\infty}\|x_n - x_{n+1}\| \le \lim_{n\to\infty}\left(\|x_n - y_n\| + \|z_n - y_n\| + \|z_n - x_{n+1}\|\right) = 0. \tag{3.34}$$

The boundedness of $\{x_n\}$ implies that there exists a subsequence $\{x_{n_k}\}$ of $\{x_n\}$ such that $\{x_{n_k}\}$ converges weakly to some $x^\ddagger \in H_1$ and

$$\limsup_{n\to\infty}\langle \nabla f(x_0) - \nabla f(x^\dagger), x_n - x^\dagger\rangle = \lim_{k\to\infty}\langle \nabla f(x_0) - \nabla f(x^\dagger), x_{n_k} - x^\dagger\rangle. \tag{3.35}$$

By (3.31), one finds that the subsequence $\{y_{n_k}\}$ of $\{y_n\}$ also converges weakly to $x^\ddagger$. This together with (3.29) implies that $x^\ddagger \in C$ and $A(x^\ddagger) \in Q$, that is, $x^\ddagger \in \Gamma$. Since $x^\dagger = \Pi_\Gamma^f(x_0)$, by applying Lemma 1(i) with (3.35), one finds that

$$\limsup_{n\to\infty}\langle \nabla f(x_0) - \nabla f(x^\dagger), x_n - x^\dagger\rangle = \langle \nabla f(x_0) - \nabla f(x^\dagger), x^\ddagger - x^\dagger\rangle \le 0. \tag{3.36}$$

By using (3.34) and (3.36), we find that

$$\limsup_{n\to\infty}\langle \nabla f(x_0) - \nabla f(x^\dagger), x_{n+1} - x^\dagger\rangle \le \limsup_{n\to\infty}\langle \nabla f(x_0) - \nabla f(x^\dagger), x_n - x^\dagger\rangle + \limsup_{n\to\infty}\langle \nabla f(x_0) - \nabla f(x^\dagger), x_{n+1} - x_n\rangle \le 0. \tag{3.37}$$

By using Lemma 3, $(A6)$, (3.27), and (3.36), one has that $\lim_{n\to\infty} D_f(x^\dagger, x_n) = 0$. This together with the relation $D_f(x,y) \ge \frac{\delta_1}{2}\|x-y\|^2$ $(\forall x \in \operatorname{dom} f, y \in \operatorname{dom}\nabla f)$ gives that $\lim_{n\to\infty}\|x_n - x^\dagger\| = 0$. This ensures that $\lim_{n\to\infty} x_n = x^\dagger$.

Case 2. Assume that there exists a subsequence $\{D_f(\hat{x}, x_{n_m})\}$ of $\{D_f(\hat{x}, x_n)\}$ such that $D_f(\hat{x}, x_{n_m}) \le D_f(\hat{x}, x_{n_m+1})$ for any $m \in \mathbb{N}$. By applying Lemma 4, there exists an increasing sequence $\{\varphi(m)\} \subset \mathbb{N}$ such that $\lim_{m\to\infty}\varphi(m) = \infty$ and for any $m \in \mathbb{N}$

$$D_f(\hat{x}, x_{\varphi(m)}) \le D_f(\hat{x}, x_{\varphi(m)+1}) \text{ and } D_f(\hat{x}, x_m) \le D_f(\hat{x}, x_{\varphi(m)+1}). \tag{3.38}$$

By rearranging and using (3.26), one observes that

$$\begin{aligned} &(1-\lambda_{\varphi(m)})\Big\{(1-\tau)D_f(z_{\varphi(m)}, y_{\varphi(m)}) + \alpha_{\varphi(m)}[D_f(z_{\varphi(m)}, \Pi_C^f y_{\varphi(m)}) + D_f(\Pi_C^f y_{\varphi(m)}, y_{\varphi(m)}) + D_g(Az_{\varphi(m)}, \Pi_Q^g Ay_{\varphi(m)}) \\ &+ D_g(\Pi_Q^g(Ay_{\varphi(m)}), Ay_{\varphi(m)})]\Big\} \\ \le& \lambda_{\varphi(m)} D_f(\hat{x}, x_0) + D_f(\hat{x}, x_{\varphi(m)}) - D_f(\hat{x}, x_{\varphi(m)+1}) - \beta_{\varphi(m)}[D_f(\hat{x}, x_{\varphi(m)}) - D_f(\hat{x}, x_{\varphi(m)-1})]. \end{aligned} \tag{3.39}$$

It follows from (3.39) that

$$\lim_{m\to\infty} D_f(\Pi_C^f y_{\varphi(m)}, y_{\varphi(m)}) = \lim_{m\to\infty} D_g(\Pi_Q^g A y_{\varphi(m)}, A y_{\varphi(m)}) = 0.$$

Using the same similar argument as in ***Case 1***, one has that there exists a subsequence of $\{x_{\varphi(m)}\}$ converges weakly to some $x^{\ddagger} \in \Gamma$ and

$$\limsup_{m\to\infty} \langle \nabla f(x_0) - \nabla f(x^{\dagger}), x_{\varphi(m)+1} - x^{\dagger} \rangle \le 0. \tag{3.40}$$

By applying (3.27) with (3.38), one gets that

$$D_f(x^{\dagger}, x_{\varphi(m)+1}) \le (1-\lambda_{\varphi(m)}) D_f(x^{\dagger}, x_{\varphi(m)}) + \lambda_{\varphi(m)} \langle \nabla f(x_0) - \nabla f(x^{\dagger}), x_{\varphi(m)+1} - x^{\dagger} \rangle. \tag{3.41}$$

By combining (3.38) with (3.41), we find that

$$D_f(x^{\dagger}, x_m) \le D_f(x^{\dagger}, x_{\varphi(m)}) \le \langle \nabla f(x_0) - \nabla f(x^{\dagger}), x_{\varphi(m)+1} - x^{\dagger} \rangle. \tag{3.42}$$

It follows from (3.40) and (3.42) that $\limsup_{m\to\infty} D_f(x^{\dagger}, x_m) = 0$. This together with the relation $D_f(x, y) \ge \frac{\delta_1}{2}\|x-y\|^2$, $\forall x \in \mathrm{dom} f, y \in \mathrm{dom}\nabla f$ implies that $\lim_{m\to\infty} \|x^{\dagger} - x_m\| = 0$. Therefore, $\{x_m\}$ converges strongly to $x^{\dagger}$. This completes the proof. □

3.2. *Second type of Bregman projection algorithm*

In this subsection, we propose a new Bregman projection algorithm which combines Byrne's CQ method with the hybrid projection method. In order to prove its strong convergence, we further make the following assumption:

(A1)* $\|A\| \ne 0$.

Now we present our iterative scheme in Algorithm 2 below.

Algorithm 2 (The modified Byrne's CQ algorithm with Armijo-line search)

Step 1: Let $x_0, x_1 \in H_1$ be arbitrary. Take $\varsigma, \tau \in (0,1)$ and $\sigma, \varrho \in (0,\infty)$. Set $n = 1$.
Step 2: Compute $y_n = \nabla f^*(\nabla f(x_n) + \beta_n(\nabla f(x_{n-1}) - \nabla f(x_n)))$ with β_n defined by

$$\beta_n = \begin{cases} \min\{\sigma, \frac{\eta_n}{D_f(x_n, x_{n-1})}\}, & \text{if } x_{n-1} \ne x_n, \\ \sigma, & \text{otherwise.} \end{cases}$$

Step 3: Compute

$$w_n = \nabla f^*(\nabla f(y_n) - \alpha_n A^*(\nabla g(Ay_n) - \nabla g(\Pi_Q^g A y_n))), \tag{3.43}$$

where $\alpha_n := \varrho\varsigma^{\gamma_n}$ with γ_n the smallest nonnegative integer γ satisfying

$$\varrho\varsigma^{\gamma} D_g A(w_n, Ay_n) \le \tau D_f(w_n, y_n). \tag{3.44}$$

If $y_n = \Pi_C^f w_n$, then stop and y_n is a solution of the SFP. Otherwise, go to ***Step 4***.
Step 4: Compute

$$x_{n+1} = \Pi_{C_n \cap Q_n}^f x_0, \tag{3.45}$$

where

$$\begin{aligned} C_n &= \{z \in H_1 : D_f(z, \Pi_C^f w_n) \le D_f(z, y_n)\}, \\ Q_n &= \{z \in H_1 : \langle \nabla f(x_0) - \nabla f(x_n), z - x_n \rangle \le 0\}. \end{aligned} \tag{3.46}$$

Update $n := n + 1$ and go to ***Step 2***.

The following elementary lemmas are quite helpful to analyze the convergence of Algorithm 2.

Lemma 8. *The Armijo-type search rule (3.44) is well-defined. Moreover,* $\alpha_n \in (0, \varrho]$, $\forall n \in \mathbb{N}$.

Proof. The proof is similar to Lemma 5, so we omit the details here for brevity. □

Lemma 9. *If* $y_n = \Pi_C^f(w_n)$ *holds for some integer* n*, then* y_n *is a solution of the SFP.*

Proof. The fact $y_n = \Pi_C^f(w_n)$ implies that $y_n \in C$. Next we will prove that $A(y_n) \in Q$. Let $\hat{x} \in \Gamma$. Thus $\hat{x} \in C$ and $A(\hat{x}) \in Q$. It follows from Lemma 1 that

$$\langle \nabla f(y_n) - \nabla f(w_n), \hat{x} - y_n \rangle \ge 0, \tag{3.47}$$

and

$$D_g(A\hat{x}, \Pi^g_Q Ay_n) \le D_g(A\hat{x}, Ay_n). \tag{3.48}$$

By using Remark 2, the three point identity (2.1), and the fact $\alpha_n \in (0, \varrho]\,(\forall n \in \mathbb{N})$ in Lemma 8, we obtain by (3.43), (3.47), and (3.48) that

$$\begin{aligned} 0 \le& \langle A^*(\nabla g(Ay_n) + \nabla g(\Pi^g_Q Ay_n)), \hat{x} - y_n\rangle = \langle \nabla g(Ay_n) + \nabla g(\Pi^g_Q Ay_n), A\hat{x} - Ay_n\rangle \\ =& D_g(A\hat{x}, \Pi^g_Q Ay_n) - D_g(A\hat{x}, Ay_n) - D_g(Ay_n, \Pi^g_Q Ay_n) \le -D_g(Ay_n, \Pi^g_Q Ay_n) \le 0. \end{aligned} \tag{3.49}$$

By using (3.49), one concludes that $D_g(Ay_n, \Pi^g_Q Ay_n) = 0$. This together with the relation $D_g(x, y) \ge \frac{\delta_2}{2}\|x-y\|^2$ $(\forall x \in \mathrm{dom} g, y \in \mathrm{dom}\nabla g)$ gives that $\|Ay_n - \Pi^g_Q Ay_n\| = 0$ i.e. that $Ay_n \in Q$. This completes the proof. □

Lemma 10. *Under conditions (A1)-(A5) and (A1*), suppose that Algorithm 2 reaches an iteration $n + 1$. Let $\{y_n\}$ and $\{w_n\}$ be the sequences generated by Algorithm 2. Then for any $\hat{x} \in \Gamma$, it holds that*

$$D_f(\hat{x}, \Pi^f_C w_n) \le D_f(\hat{x}, y_n) - (1-\tau)D_f(w_n, y_n) - \alpha_n D_g(Aw_n, \Pi^g_Q Ay_n) - D_f(\Pi^f_C w_n, w_n). \tag{3.50}$$

Proof. By using the three point identity (2.1), one gives that

$$\begin{aligned} D_f(\hat{x}, w_n) =& D_f(\hat{x}, y_n) - D_f(w_n, y_n) + \langle \nabla g(y_n) - \nabla g(w_n), \hat{x} - w_n\rangle \\ =& D_f(\hat{x}, y_n) - D_f(w_n, y_n) + \alpha_n\langle A^*(\nabla g(Ay_n) - \nabla g(\Pi^g_Q Ay_n)), \hat{x} - w_n\rangle. \end{aligned} \tag{3.51}$$

By using the three point identity (2.1), we obtain that

$$\langle \nabla g(Ay_n) - \nabla g(\Pi^g_Q Ay_n), A\hat{x} - Ay_n\rangle = D_g(A\hat{x}, \Pi^g_Q Ay_n) - D_g(Ay_n, \Pi^g_Q Ay_n) - D_g(A\hat{x}, Ay_n) = -D_g(Ay_n, \Pi^g_Q Ay_n), \tag{3.52}$$

and

$$\langle \nabla g(Ay_n) - \nabla g(\Pi^g_Q Ay_n), Ay_n - Aw_n\rangle = D_g(Aw_n, Ay_n) + D_g(Ay_n, \Pi^g_Q Ay_n) - D_g(Aw_n, \Pi^g_Q Ay_n). \tag{3.53}$$

By combining (3.52) with (3.53), one obtains that

$$\begin{aligned} \langle A^*(\nabla g(Ay_n) - \nabla g(\Pi^g_Q Ay_n)), \hat{x} - w_n\rangle =& \langle \nabla g(Ay_n) - \nabla g(\Pi^g_Q Ay_n), A\hat{x} - Ay_n\rangle + \langle \nabla g(Ay_n) - \nabla g(\Pi^g_Q Ay_n), Ay_n - Aw_n\rangle \\ \le& D_g(Aw_n, Ay_n) - D_g(Aw_n, \Pi^g_Q Ay_n). \end{aligned} \tag{3.54}$$

By substituting (3.54) into (3.51) and applying (3.44), one gives that

$$\begin{aligned} D_f(\hat{x}, w_n) =& D_f(\hat{x}, y_n) - D_f(w_n, y_n) + \alpha_n D_g(Aw_n, Ay_n) - \alpha_n D_g(Aw_n, \Pi^g_Q Ay_n) \\ \le& D_f(\hat{x}, y_n) - (1-\tau)D_f(w_n, y_n) - \alpha_n D_g(Aw_n, \Pi^g_Q Ay_n). \end{aligned} \tag{3.55}$$

On the other hand, Lemma 1(ii) yields that

$$D_f(\hat{x}, \Pi^f_C w_n) \le D_f(\hat{x}, w_n) - D_f(\Pi^f_C w_n, w_n). \tag{3.56}$$

By combining (3.55) with (3.56), we conclude that

$$D_f(\hat{x}, \Pi^f_C w_n) \le D_f(\hat{x}, y_n) - (1-\tau)D_f(w_n, y_n) - \alpha_n D_g(Aw_n, \Pi^g_Q Ay_n) - D_f(\Pi^f_C w_n, w_n).$$

This completes the proof. □

We are now in a position to state our strong convergence result of Algorithm 2.

Theorem 2. *Suppose that conditions (A1)–(A5) and (A1*) hold. Then the sequence $\{x_n\}$ generated by Algorithm 2 converges strongly to $x^\dagger \in \Gamma$ with $x^\dagger = \Pi^f_\Gamma x_0$.*

Proof. For simplicity, we divide the proof into three claims as follows.

Claim 1. *$\{x_n\}$ is well-defined. Now we show that C_n and Q_n are nonempty, closed, and convex sets for any $n \in \mathbb{N}$. It follows from the definition of C_n that*

$$D_f(z, \Pi^f_C w_n) \le D_f(z, y_n), \quad \forall z \in C_n. \tag{3.57}$$

By using the definition of $D_f(\cdot,\cdot)$, one obtains that (3.57) is equivalent to

$$f(y_n) - f(\Pi^f_C w_n) - \langle \nabla f(y_n), y_n\rangle + \langle \nabla f(\Pi^f_C w_n), \Pi^f_C w_n\rangle \le \langle \nabla f(\Pi^f_C w_n) - \nabla f(y_n), z\rangle.$$

It is obvious that C_n is a closed and convex set. Based on (3.46), it is also obvious that Q_n is a closed and convex set.

Next we show that $\Gamma \subset C_n \cap Q_n$, for any $n \in \mathbb{N}$. From Lemma 10, we observe that $\Gamma \subset C_n$, for any $n \in \mathbb{N}$. On the other hand, it is clear that $\Gamma \subset Q_1 = H_1$. Therefore, $\Gamma \subset C_1 \cap Q_1$ and hence $\Pi^f_{C_1 \cap Q_1} x_0$ is well defined. Suppose that $\Gamma \subset C_m \cap Q_m$ for some $m \in \mathbb{N}$. This indicates that $C_m \cap Q_m$ is nonempty, closed, and convex. Therefore, there exists a unique point x_{m+1} such that x_{m+1} is the Bregman projection of x_0 onto $C_m \cap Q_m$. This together with Lemma 1(i) further gives that

$$\langle \nabla f(x_{m+1}) - \nabla f(x_0), z - x_{m+1} \rangle \geq 0, \ \ \forall z \in C_m \cap Q_m. \tag{3.58}$$

On the other hand, the fact $\Gamma \subset C_m \cap Q_m$ and (3.58) assert that

$$\langle \nabla f(x_{m+1}) - \nabla f(x_0), z - x_{m+1} \rangle \geq 0, \ \ \forall z \in \Gamma. \tag{3.59}$$

It follows from (3.59) that $\Gamma \subset Q_{m+1}$ and therefore $\Gamma \subset C_{m+1} \cap Q_{m+1}$. By induction, we show that $\Gamma \subset C_n \cap Q_n$, for any $n \in \mathbb{N}$. Hence $C_n \cap Q_n$ is nonempty, closed, and convex. This implies that $x_{n+1} = \Pi^f_{C_n \cap Q_n} x_0$ is well defined.

Claim 2. *We show that $w_n(x_n) \subset \Gamma$. By using the definition of Q_n and Lemma 1(i), we get that $x_n = \Pi^f_{Q_n} x_0$. Hence, Lemma 1(ii) and the fact $\Gamma \subset Q_n$ assert that*

$$D_f(x_n, x_0) \leq D_f(z, x_0) - D_f(z, x_n) \leq D_f(z, x_0), \ \ \forall z \in \Gamma. \tag{3.60}$$

Therefore, $\{D_f(x_n, x_0)\}$ is bounded by $D_f(z, x_0)$ for any $z \in \Gamma$. This implies that $\{x_n\}$ is bounded. Then there exists a subsequence $\{x_{n_k}\}$ of $\{x_n\}$ such that $\{x_{n_k}\}$ converges weakly to some $x^\ddagger \in H_1$. Next we show that $x^\ddagger \in \Gamma$. By the definitions of C_n and Q_n, we deduce that

$$D_f(x_{n+1}, \Pi^f_C w_n) \leq D_f(x_{n+1}, y_n) \tag{3.61}$$

and

$$\langle \nabla f(x_0) - \nabla f(x_n), x_{n+1} - x_n \rangle \leq 0. \tag{3.62}$$

By using the three point identity (2.1) and (3.62), one gets that

$$D_f(x_{n+1}, x_n) = D_f(x_{n+1}, x_0) - D_f(x_n, x_0) + \langle \nabla f(x_0) - \nabla f(x_n), x_{n+1} - x_n \rangle \leq D_f(x_{n+1}, x_0) - D_f(x_n, x_0). \tag{3.63}$$

Thus (3.63) asserts that $\{D_f(x_n, x_0)\}$ is increasing. This together with the boundedness of $\{x_n\}$ asserts that $\lim_{n\to\infty} D_f(x_n, x_0)$ exists. By (3.63), one observes that

$$\lim_{n\to\infty} D_f(x_{n+1}, x_n) = 0. \tag{3.64}$$

This implies that

$$\lim_{n\to\infty} \|\nabla f(x_{n+1}) - \nabla f(x_n)\| = 0. \tag{3.65}$$

By applying the three point identity (2.1), we get that

$$D_f(x_{n+1}, x_{n-1}) = D_f(x_{n+1}, x_n) - D_f(x_{n-1}, x_n) + \langle \nabla f(x_n) - \nabla f(x_{n-1}), x_{n+1} - x_{n-1} \rangle. \tag{3.66}$$

By using the boundedness of $\{x_n\}$, (3.64), and (3.65), one observes that

$$\lim_{n\to\infty} D_f(x_{n+1}, x_{n-1}) = 0. \tag{3.67}$$

By using the definition of $\{y_n\}$ and (2.2), one gets that

$$D_f(x_{n+1}, y_n) = D_f(x_{n+1}, \nabla f^*((1-\beta_n)\nabla f(x_n) + \beta_n \nabla f(x_{n-1}))) \leq (1-\beta_n) D_f(x_{n+1}, x_n) + \beta_n D_f(x_{n+1}, x_{n-1}). \tag{3.68}$$

In view of (3.61), (3.64), (3.67), and (3.68), one gets that

$$\lim_{n\to\infty} D_f(x_{n+1}, y_n) = \lim_{n\to\infty} D_f(x_{n+1}, \Pi^f_C(w_n)) = 0. \tag{3.69}$$

By virtue of (3.69), one obtains that

$$\lim_{n\to\infty} \|\nabla f(x_{n+1}) - \nabla f(\Pi^f_C(w_n))\| = \lim_{n\to\infty} \|\nabla f(x_{n+1}) - \nabla f(y_n)\| = 0. \tag{3.70}$$

By the triangle inequality, we derive that

$$\|\nabla f(\Pi^f_C w_n) - \nabla f(y_n)\| \leq \|\nabla f(\Pi^f_C w_n) - \nabla f(x_{n+1})\| + \|\nabla f(x_{n+1}) - \nabla f(y_n)\|. \tag{3.71}$$

By applying (3.70) with (3.71), we conclude that

$$\lim_{n\to\infty} \|\nabla f(\Pi^f_C w_n) - \nabla f(y_n)\| = 0. \tag{3.72}$$

By remembering (3.50) in Lemma 10 and the three point identity (2.1), one gets that

$$\begin{aligned} 0 \leq& (1-\tau) D_f(w_n, y_n) + \alpha_n D_g(A w_n, \Pi^g_Q A y_n) + D_f(\Pi^f_C w_n, w_n) \leq D_f(\hat{x}, y_n) - D_f(\hat{x}, \Pi^f_C w_n) \\ &\leq - \langle \nabla f(y_n) - \nabla f(\Pi^f_C w_n), \hat{x} - \Pi^f_C w_n \rangle. \end{aligned} \tag{3.73}$$

Therefore, it follows from (3.73) that

$$\lim_{n\to\infty} D_f(w_n, y_n) = \lim_{n\to\infty} D_g(Aw_n, \Pi_Q^g Ay_n) = \lim_{n\to\infty} D_f(\Pi_C^f w_n, w_n) = 0. \tag{3.74}$$

Then $\{y_{n_k}\}$ and $\{w_{n_k}\}$ also converge weakly to $x^\ddagger$. We further obtain that $x^\ddagger \in \Gamma$.

Claim 3. *We show that $x_n \to x^\dagger \in \Gamma$ with $x^\dagger = \Pi_\Gamma^f x_0$. Since $x_n = \Pi_{Q_n}^f x_0$ and $\Gamma \subset Q_n$, one has that $D_f(x_n, x_0) \le D_f(x^\dagger, x_0)$. This together with the three point identity (2.1) implies that*

$$D_f(x_n, x^\dagger) = D_f(x_n, x_0) - D_f(x^\dagger, x_0) + \langle \nabla f(x_0) - \nabla f(x^\dagger), x_n - x^\dagger\rangle \le \langle \nabla f(x_0) - \nabla f(x^\dagger), x_n - x^\dagger\rangle. \tag{3.75}$$

Taking into account that the limit point $x^\ddagger$ of $\{x_{n_k}\}$ belongs to Γ and $x^\dagger$ is the Bregman projection of x_0 onto Γ, it follows from Lemma 1(i) that

$$\limsup_{k\to\infty} \langle \nabla f(x_0) - \nabla f(x^\dagger), x_{n_k} - x^\dagger\rangle \le 0. \tag{3.76}$$

By combining (3.75) with (3.76), it follows that $\limsup_{k\to\infty} D_f(x_{n_k}, x^\dagger) = 0$. This implies that $x_{n_k} \to x^\dagger$, as $k \to \infty$. Since $\{x_{n_k}\}$ is an arbitrary weakly convergent subsequence of $\{x_n\}$, we conclude that $\{x_n\}$ converges strongly to $x^\dagger$. The proof is completed. □

Remark 6. The convergence proof of our algorithms cannot rely on the triangle inequality and it solely derived from the geometric properties of Bregman distance.

4. Numerical experiments

In this section, we provide numerical experiments of the proposed algorithms in signal processing and image deblurring to verify the effectiveness of the algorithms and compare them with the schemes in the literature [7,8]. Our code is implemented in MATLAB 2023b on a MacBook Air having a chip Apple M2 and 8 GB of RAM.

Example 1. In this example, our goal is to solve the signal processing problem using the proposed algorithms. The mathematical model for signal processing represented by $b = Ax$, where

- $A \in \mathbb{R}^{m\times n}$ represents a matrix that maps the input signal x to the output signal b.
- $x \in \mathbb{R}^{n\times 1}$ is the input signal vector with k nonzero components that needs to be recovered (i.e., original signal).
- $b \in \mathbb{R}^{m\times 1}$ is the output signal vector (i.e., degraded signal).

This equation signifies that the output signal b is obtained by applying a linear transformation represented by the matrix A to the input signal x. Finding a sparse solution of the linear inverse problem can be seen as solving the constrained least squares problem:

$$\min_x f(x) := \frac{1}{2}\|Ax - b\|_2^2, \text{ such that } \|x\|_1 \le r,$$

where $r > 0$ is a given constant. In this convex optimization problem, the objective function $f(x)$ represents the least squares error between the observed and predicted output signals. To convert this convex optimization problem into a feasibility problem, we can introduce the following problem:

$$\text{find } x \in C, \text{ such that } Ax \in Q,$$

where $C := \{\|x\|_1 \le r\}$ and $Q = \{b\}$.

For our Algorithms 1 and 2, we set $f(x) = g(x) = \frac{1}{2}\|x\|_2^2$. One can check that f and g satisfy conditions (A2) and (A3) with strongly convex coefficient 1. Moreover, we have $\nabla f^*(x) = (\nabla f)^{-1}(x)$. It is not difficult to see that $\nabla f(x) = x$ and $(\nabla f)^{-1}(x) = x$. The corresponding Bregman distance is given by $D_f(x, y) = \frac{1}{2}\|x - y\|_2^2$ which is the squared Euclidean distance (SE). For simplicity, we abbreviate our Algorithms 1 and 2 with the squared Euclidean distance as our Alg. 1-SE and our Alg. 2-SE, respectively.

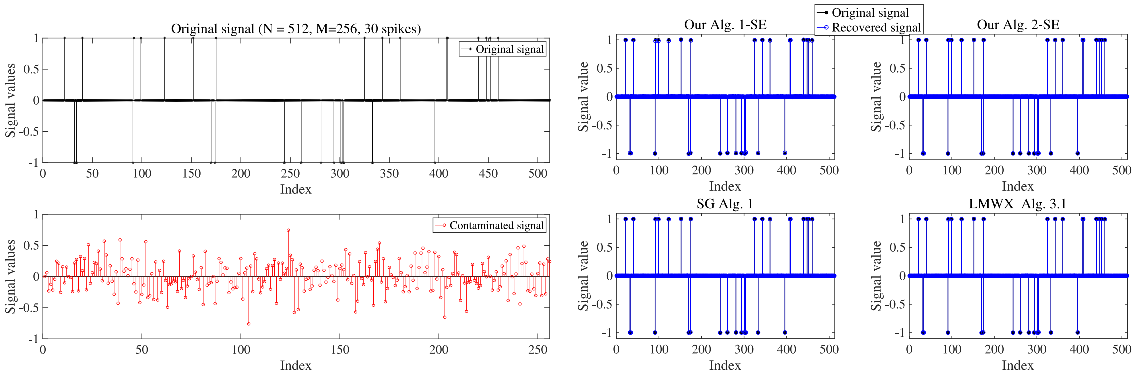
In the following examples, we compare the proposed algorithms with Shehu and Gibali's Algorithm 1 [7] (abbreviated as SG Alg. 1) and López et al.'s Algorithm 3.1 [8] (abbreviated as LMWX Alg. 3.1). The parameters of all the algorithms are set as follows.

- For our Alg. 1-SE, we set $\mu = 2$, $\nu = 0.9$, $\tau = 0.9$, $\sigma = 0.3$, $\eta_n = \frac{1}{10(n+1)^2}$, and $\lambda_n = \frac{1}{100(n+1)}$. Choose $\varrho = 2$, $\varsigma = 0.9$, $\tau = 0.9$, $\sigma = 0.3$, and $\eta_n = \frac{1}{10(n+1)^2}$ for our Alg. 2-SE.
- Select $\theta = 0.3$, $\gamma = 2$, $l = 0.5$, and $\mu = 0.5$ for SG Alg. 1 [7].
- For LMWX Alg. 3.1 [8], we pick $\rho_n = 0.2$.

In the example here, we first set $n = 512$ and $m = 256$. We randomly generate the original signal $x \in [-1, 1]$ with k spikes (spikes have a value of ± 1). Subsequently, we generate a random matrix A with m rows and n columns, and then orthogonalize its columns. then the degenerate signal is generated by $b = Ax$. The mean square error $\text{MSE} = \frac{1}{n}\|\hat{x} - x\|_2^2$ was used as a measure of the reconstruction error of the recovered signal $\hat{x}$ and the original signal x. The stopping criterion for all algorithms is $\text{MSE} < 10^{-6}$. The numerical results of the proposed algorithms and the comparison algorithms under signals with different sparsities are displayed

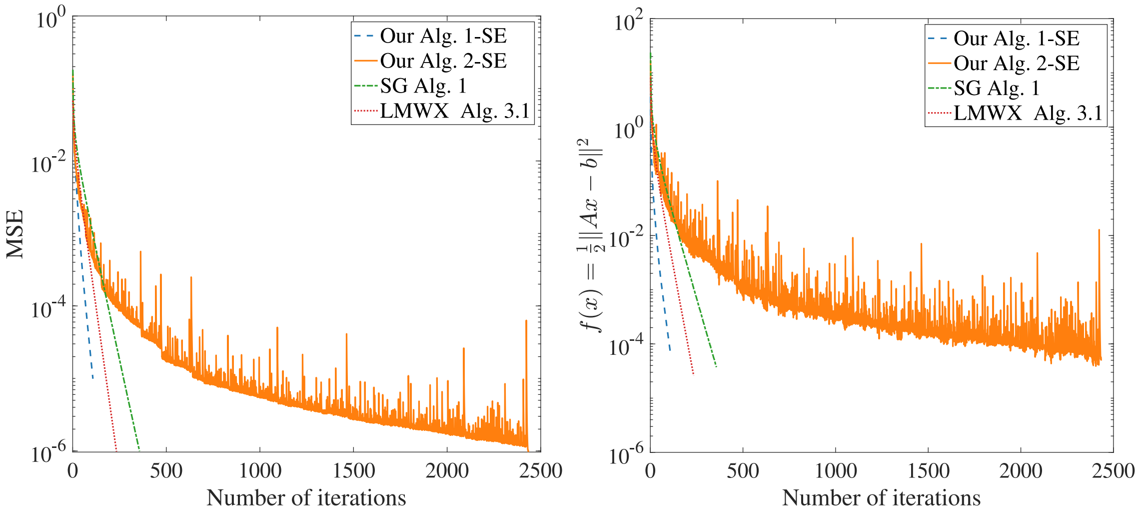
Table 1
Numerical results for all algorithms in Example 1.

Algorithms	$k=10$		$k=20$		$k=30$		$k=40$	
	Iter.	Time (s)	Iter.	Time (s)	Iter.	Time (s)	Iter.	Time (s)
Our Alg. 1-SE	55	0.07	105	0.11	108	0.12	219	0.20
Our Alg. 2-SE	1025	0.48	2274	1.05	2434	0.99	7595	3.30
SG Alg.1	170	0.23	339	0.38	357	0.39	786	0.91
LMWX Alg. 3.1	118	0.03	224	0.04	234	0.04	476	0.11



(a) The original signal and degraded signal　　(b) The recovered signals

Fig. 1. The original signal, the degraded signal, and the recovered signals under $k=30$.



(a) MSE　　(b) $f(x)$

Fig. 2. Trends of all algorithms regarding MSE and $f(x)$, for $k=30$, respectively.

in Table 1, where "Iter". denotes the number of iterations required to reach the stopping condition and "Time (s)" represents the execution time in seconds.

The reconstruction results for all algorithms at $k=30$ are displayed in Figs. 1 and 2.

Remark 7. From Table 1, Figs. 1, and 2, we observe that our proposed algorithms perform well in solving signal processing problems and achieve good recovery results. Moreover, the information in Table 1 indicates that our Alg. 1-SE requires the fewest number of iterations, while our Alg. 2-SE needs the most. It is noteworthy that the time required by our Alg. 2-SE is also the longest, which is not surprising as this algorithm requires a projection calculation at the end of each iteration. However, it should be noted that both our algorithms achieve strong convergence in infinite-dimensional spaces, whereas the methods in [7,8] only obtain weak convergence. On the other hand, it can be from Fig. 2 that our Alg. 1-SE converges the fastest.

Example 2. In this example, we consider applying the proposed algorithms for solving the image deblurring problem. Recall that the model representation of the image deblurring problem is given as:

$$b = Ax + \epsilon,$$

where

- $b \in \mathbb{R}^{m\times m}$ is the observed blurred image.
- $A \in \mathbb{R}^{m\times m}$ is the blurring operator, which represents the effect of the blurring process on the original image x.
- $x \in \mathbb{R}^{m\times m}$ is the original sharp image that we aim to recover.
- $\epsilon \in \mathbb{R}^{m\times m}$ is the noise present in the observed image.

This equation signifies that the observed blurred image b is obtained by convolving the original sharp image x with the blurring operator A, which introduces blurring effects such as motion blur or out-of-focus blur. Additionally, the observed image may contain noise represented by ϵ. The goal of image deblurring is to estimate the original sharp image x from the observed blurred image b, taking into account the blurring effects and noise present in the observed image.

We can convert the image deblurring problem into the following constrained convex optimization problem:

$$\min_{x\in C} \frac{1}{2}\|Ax - (b-\epsilon)\|_2^2,$$

where C represents the domain of the original image x. This problem can fall into the framework of the split feasibility problem by setting $Q := \|y : \|y-(b-\epsilon)\| \le s$, where s is a sufficiently small positive constant.

In our Algorithms 1 and 2, we consider $f(x) = g(x) = \sum_{i=1}^{j} x_i \log\left(x_i\right)$ which satisfies assumptions (A2) and (A3). In addition, one has $\nabla f^*(x) = (\nabla f)^{-1}(x)$, where

$$\nabla f(x) = \left(1+\log\left(x_1\right), \dots, 1+\log\left(x_j\right)\right)^T$$
$$(\nabla f)^{-1}(x) = \left(\exp\left(x_1-1\right), \dots, \exp\left(x_j-1\right)\right)^T.$$

Moreover, the corresponding Bregman distance is given by

$$D_f(x,y) = \sum_{i=1}^{j}\left(x_i \log\left(\frac{x_i}{y_i}\right) + y_i - x_i\right),$$

which is the Kullback–Leibler distance (KL).

Now we denote our Algorithm 1 with the Kullback–Leibler distance as our Alg. 1-KL. In the following examples, we compare the computational efficiency of our Alg. 1-SE, our Alg. 1-KL, our Alg. 2-SE, SG Alg. 1 [7], and LMWX Alg. 3.1 [8]. The parameters of all the algorithms are set as follows.

- For our Alg. 1-SE and our Alg. 1-KL, we set $\mu = 5$, $\nu = 0.8$, $\tau = 0.9$, $\sigma = 0.3$, $\eta_n = \frac{1}{10(n+1)^2}$, and $\lambda_n = \frac{1}{100(n+1)}$. Choose $\varrho = 5$, $\varsigma = 0.8$, $\tau = 0.9$, $\sigma = 0.3$, and $\eta_n = \frac{1}{10(n+1)^2}$ for our Alg. 2-SE.
- Select $\theta = 0.3$, $\gamma = 0.5$, $l = 0.5$, and $\mu = 0.5$ for SG Alg. 1 [7].
- For LMWX Alg. 3.1 [8], we pick $\rho_n = 2$.

In this example, we select three test images with a domain of $[0,1]$. The test images are initially contaminated by a 9×9 Gaussian random blur with a standard deviation of 1.5, and further corrupted by random Gaussian white noise with zero mean and a standard deviation of 10^{-4}. The maximum number of iterations 100 is used as a common stopping criterion for all algorithms. We use Signal-to-Noise Ratio (SNR) and Peak Signal-to-Noise Ratio (PSNR) to measure the reconstruction quality. The definition of SNR is as follows:

$$\mathrm{SNR} = 20\log_{10}\frac{\|x\|_2}{\|\hat{x}-x\|_2}.$$

The definition of PSNR is given by the following equation:

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{\mathrm{MAX}^2}{\mathrm{MSE}}\right),$$

where MAX is the maximum possible pixel value of the image x, and MSE (Mean Squared Error) is the average squared difference between the original and reconstructed image. It is well known that higher values of SNR and PSNR indicate better reconstruction quality.

The numerical results of the proposed algorithms as well as the comparison ones for the three test images are displayed in Table 2. Figs. 3, 4, and 5 show the reconstructed images for all the algorithms in the three images, respectively. Finally, as an example, we present the variation curves of SNR and PSNR for all algorithms under the image “Cameraman” in Fig. 6.

Remark 8. From Table 2 and Figs. 3–6, it can be observed that our algorithms perform well in solving image deblurring problems and outperform the algorithms in [7,8]. Moreover, note that our Alg. 1-KL achieves higher reconstruction quality than our Alg. 1-SE in some cases (see Table 2). This suggests that our Algorithm 1 may achieve better results when using the Kullback–Leibler distance compared to using the squared Euclidean distance.

Table 2
Numerical results of all algorithms for different images.

Algorithms	Cameraman			Pepper			Barbara		
	Time (s)	SNR	PSNR	Time (s)	SNR	PSNR	Time (s)	SNR	PSNR
Our Alg. 1-SE	2.17	21.36	26.95	2.48	23.38	29.12	6.53	19.38	25.27
Our Alg. 1-KL	5.96	21.62	27.20	6.25	23.21	28.95	19.70	19.40	25.28
Our Alg. 2-SE	2.23	21.10	26.69	2.46	23.09	28.83	6.41	19.30	25.18
SG Alg. 1	1.50	21.16	26.74	1.58	23.19	28.93	4.54	19.28	25.17
LMWX Alg. 3.1	0.97	19.94	25.52	1.04	20.81	26.55	3.13	18.21	24.10

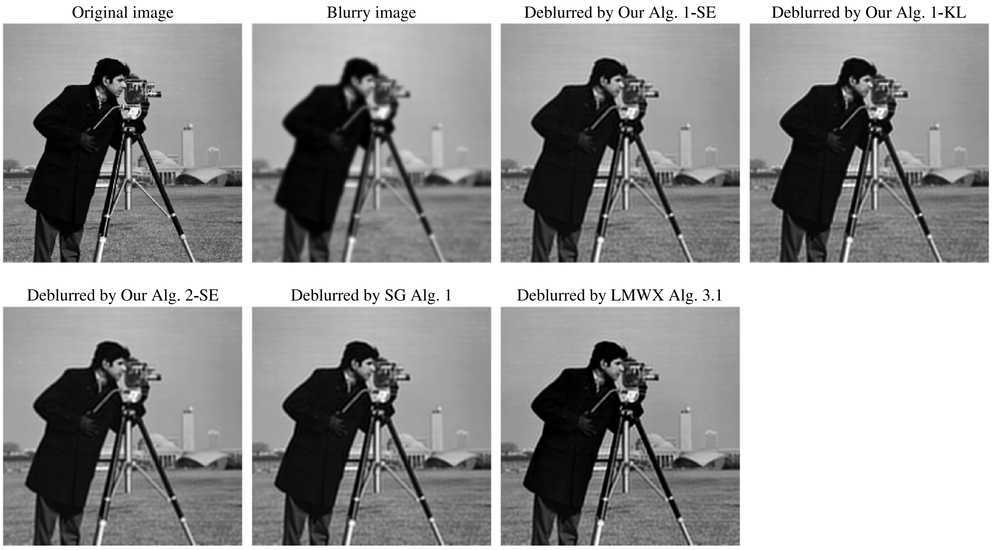


Fig. 3. Reconstructed images for all algorithms under image “Cameraman”.

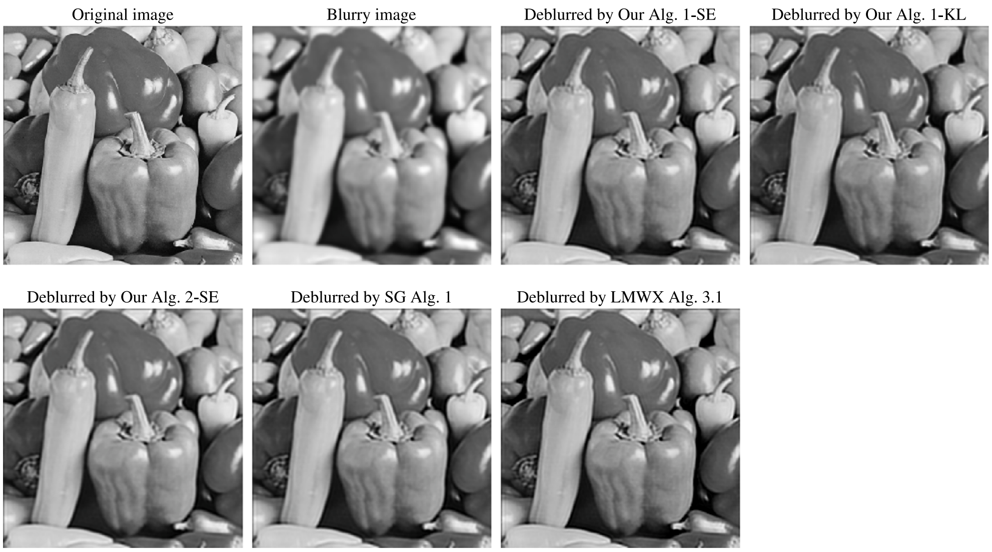
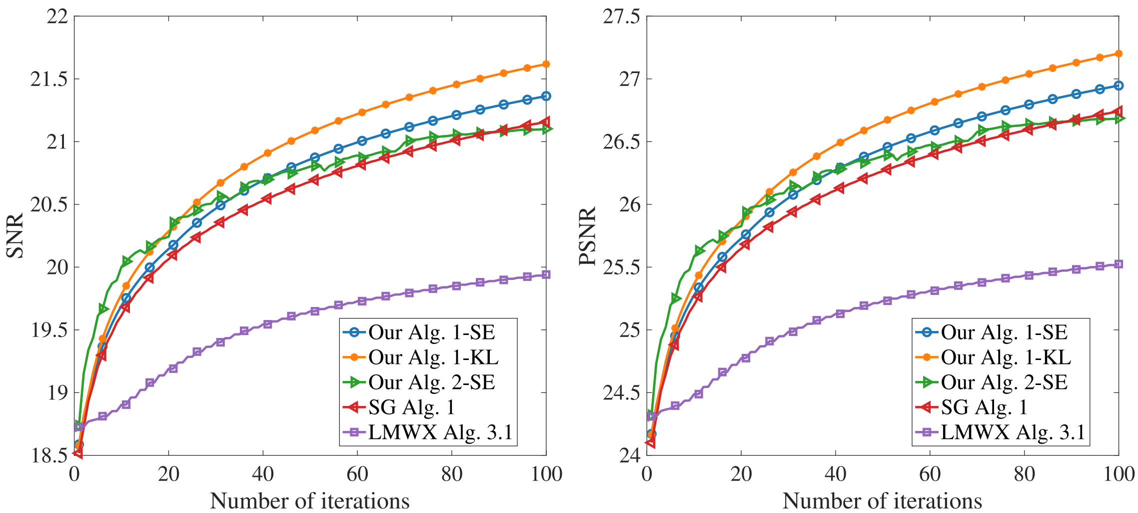


Fig. 4. Reconstructed images for all algorithms under image “Pepper”.

Fig. 5. Reconstructed images for all algorithms under image "Barbara".



(a) SNR (b) PSNR

Fig. 6. The variation curves of SNR and PSNR for all algorithms under the image "Cameraman".

5. Conclusion

In this paper, two novel adaptive inertial Bregman iteration algorithms are proposed to solve split feasibility problems in Hilbert spaces. By leveraging the Halpern method and the hybrid projection method, we establish the strong convergence of the proposed algorithms under appropriate conditions. Furthermore, some numerical experiments in signal processing and image deblurring validate the computational efficiency of our methods. The results of this paper extend and enhance the latest findings on split feasibility problems.

CRediT authorship contribution statement

Liya Liu: Writing – review & editing, Writing – original draft, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. **Songxiao Li:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. **Bing Tan:** Writing – review & editing, Writing – original draft, Visualization, Software, Investigation, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Liya Liu and Songxiao Li report financial support was provided by National Natural Science Foundation of China. Bing Tan reports financial support was provided by Natural Science Foundation Project of Chongqing. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

The first author was supported by National Natural Science Foundation of China (Grant No. 12201517) and the Natural Science Foundation of Chongqing, China (Grant No. CSTB2022NSCQ-MSX0580). The corresponding author thanks the support of the Fundamental Research Funds for the Central Universities, China (Grant No. SWU-KQ24052) and the Natural Science Foundation of Chongqing, China (Grant No. CSTB2024NSCQ-MSX0354). The second author was supported by NNSF of China (No. 12371131), STU Scientific Research Initiation Grant (No. NTF23004), Guangdong Basic and Applied Basic Research Foundation (Nos. 2023A1515010614 and 2024A1515012404).

References

[1] Censor Y, Elfving T. A multiprojection algorithm using Bregman projections in a product space. Numer Algorithms 1994;8(2–4):221–39.
[2] Censor Y, Bortfeld T, Martin B, Trofimov A. A unified approach for inversion problems in intensity-modulated radiation therapy. Phys Med Biol 2006;51:2353–65.
[3] Shehu Y, Dong QL, Liu LL. Global and linear convergence of alternated inertial methods for split feasibility problems. Rev R Acad Cienc Exactas Fís Nat Ser A Mat 2021;115:1–26.
[4] Censor Y, Segal A. Iterative projection methods in biomedical inverse problems. In: Censor Y, Jiang M, Louis AK, editors. Mathematical methods in biomedical imaging and intensity-modulated therapy. vol. 10, 2008, p. 65–96.
[5] Anh PK, Vinh NT, Dung VT. A new self-adaptive CQ algorithm with an application to the LASSO problem. J Fixed Point Theory Appl 2018;20:142.
[6] Byrne C. Iterative oblique projection onto convex sets and the split feasibility problem. Inverse Problems 2002;18(2):441–53.
[7] Shehu Y, Gibali A. New inertial relaxed method for solving split feasibilities. Optim Lett 2021;15(6):2109–26.
[8] López G, Martín-Márquez V, Wang F, et al. Solving the split feasibility problem without prior knowledge of matrix norms. Inverse Problems 2012;28(8):085004.
[9] Qu B, Xiu NH. A note on the CQ algorithm for the split feasibility problem. Inverse Problems 2005;21(5):1655.
[10] Gibali A, Liu LW, Tang YC. Note on the modified relaxation CQ algorithm for the split feasibility problem. Optim Lett 2018;12:817–30.
[11] Taddele GH, Kumam P, Sunthrayuth P, et al. Self-adaptive algorithms for solving split feasibility problem with multiple output sets. Numer Algorithms 2023;92(2):1335–66.
[12] Dong QL, He SN, Rassias MT. General splitting methods with linearization for the split feasibility problem. J Global Optim 2021;79:813–36.
[13] Polyak BT. Minimization of unsmooth functionals. USSR Comput Math Math Phys 1969;9(3):14–29.
[14] Wang F. Polyak's gradient method for split feasibility problem constrained by level sets. Numer Algorithms 2018;77(3):925–38.
[15] Yao Y, Iyiola OS, Shehu Y. Subgradient extragradient method with double inertial steps for variational inequalities. J Sci Comput 2022;90:1–29.
[16] Cheval H, Kohlenbach U, Leuştean L. On modified halpern and tikhonov-mann iterations. J Optim Theory Appl 2023;197(1):233–51.
[17] Oyewole OK, Abass HA, Mebawondu AA, et al. A tseng extragradient method for solving variational inequality problems in Banach spaces. Numer Algorithms 2022;89(2):769–89.
[18] Sahu DR, Cho YJ, Dong QL, et al. Inertial relaxed CQ algorithms for solving a split feasibility problem in Hilbert spaces. Numer Algorithms 2021;87:1075–95.
[19] Ceng LC, Ansari QH, Yao JC. Relaxed extragradient methods for finding minimum-norm solutions of the split feasibility problem. Nonlinear Anal 2012;75(4):2116–25.
[20] Shehu Y, Iyiola OS, Enyi CD. An iterative algorithm for solving split feasibility problems and fixed point problems in Banach spaces. Numer Algorithms 2016;72:835–64.
[21] Solodov MV, Svaiter BF. Forcing strong convergence of proximal point iterations in a Hilbert space. Math Program 2000;87:189–202.
[22] Sunthrayuth P, Jolaoso LO, Cholamjiak P. New bregman projection methods for solving pseudo-monotone variational inequality problem. J Appl Math Comput 2022;68(3):1565–89.
[23] Bregman LM. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. USSR Comput Math Math Phys 1967;7(3):200–17.
[24] Lohit H, Kumar D. Modified total bregman divergence driven picture fuzzy clustering with local information for brain MRI image segmentation. Appl Soft Comput 2023;144:110460.
[25] Hayashi M. Bregman divergence based em algorithm and its application to classical and quantum rate distortion theory. IEEE Trans Inform Theory 2023;69(6):3460–92.
[26] Jolaoso LO, Sunthrayuth P, Cholamjiak P, et al. Analysis of two versions of relaxed inertial algorithms with bregman divergences for solving variational inequalities. J Comput Appl Math 2022;41(7):300.
[27] Polyak BT. Some methods of speeding up the convergence of iteration methods. USSR Comput Math Math Phys 1964;4(5):1–17.
[28] Taiwo A, Jolaoso LO, Mewomo OT. Inertial-type algorithm for solving split common fixed point problems in Banach spaces. J Sci Comput 2021;86:1–30.
[29] Thong DV, Vuong PT. R-linear convergence analysis of inertial extragradient algorithms for strongly pseudo-monotone variational inequalities. J Comput Appl Math 2022;406:114003.
[30] Iyiola OS, Shehu Y. Convergence results of two-step inertial proximal point algorithm. Appl Numer Math 2022;182:57–75.
[31] Reich S, Sabach S. A strong convergence theorem for a proximal-type algorithm in reflexive Banach spaces. J Nonlinear Convex Anal 2009;10(3):471–85.
[32] Bauschke HH, Borwein JM. Legendre functions and the method of random bregman projections. J Convex Anal 1997;4(1):27–67.
[33] Bauschke HH, Wang X, Ye J, et al. Bregman distances and Chebyshev sets. J Approx Theory 2009;159(1):3–25.

[34] Censor Y, Iusem AN, Zenios SA. An interior point method with bregman functions for the variational inequality problem with paramonotone operators. Math Program 1998;81:373–400.
[35] Butnariu D, E. Resmerita. Bregman distances, totally convex functions, and a method for solving operator equations in Banach spaces. Abstr Appl Anal 2006;2006:84919.
[36] Rockafellar RT. Convex analysis. Princeton: Princeton University Press; 1970.
[37] Martín Márquez V, Reich S, Sabach S. Iterative methods for approximating fixed points of Bregman nonexpansive operators. Discrete Contin Dyn Syst 2013;6(4):1043–63.
[38] Chbani Z, Riahi H. Weak and strong convergence of an inertial proximal method for solving Ky Fan minimax inequalities. Optim Lett 2013;7:185–206.
[39] Maingé PE. Strong convergence of projected subgradient methods for nonsmooth and nonstrictly convex minimization. Set-Valued Anal. 2008;16(7–8):899–912.