Journal of Computational and Applied Mathematics 484 (2026) 117507

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

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

Two inertial multistep projection-type algorithms for solving mixed split feasibility problems in Hilbert space

Nguyen Song Ha [a], Simeon Reich [b], Truong Minh Tuyen [a,*], Pham Thi Thu [c]

[a] TNU-University of Sciences, Thainguyen, Vietnam
[b] Department of Mathematics, The Technion – Israel Institute of Technology, 32000, Haifa, Israel
[c] Thainguyen University of Technology, Thainguyen, Vietnam

ARTICLE INFO

2010 MSC:
47H09
47J25
65K10
90C25

Keywords:
Hilbert space
Hybrid projection algorithm
Inertial multistep method
Metric projection
Shrinking projection algorithm
Split feasibility problem

ABSTRACT

We study the mixed split feasibility problem in real Hilbert space. In order to find a solution to this problem, we use hybrid and shrinking projection methods to propose two new inertial multistep projection-type algorithms. A distinctive feature of our methods is that the inertial parameters are only required to be bounded, rather than diminishing or constrained to lie within fixed intervals such as $[-1, 1]$ or $[0, a]$, as is commonly imposed in many existing inertial schemes. This relaxation makes the selection of inertial factors more flexible and easier to implement while still ensuring strong convergence. In addition, the other control parameters are selected so that the implementation of our algorithm does not depend on any prior information regarding the norms of the transfer operators.

1. Introduction

Reich et al. [1] have recently considered the following *mixed split feasibility problem* (MSFP, for short). Assume that

(A1) $\mathbb{H}$, $\mathbb{H}_i$ and $\mathcal{H}$ are real Hilbert spaces, where $i = 1, 2, 3, \ldots, N$; C and Q_i are nonempty, closed and convex subsets of $\mathbb{H}$ and $\mathbb{H}_i$, respectively.
(A2) $\mathcal{P}_i : \mathbb{H} \to \mathbb{H}_i$ and $\mathcal{L}_i : \mathbb{H}_i \to \mathcal{H}$ are bounded linear operators.
(A3) y is a given element in $\mathcal{H}$.
(A4) $\Omega = \cap_{i=1}^N \{x \in C : \mathcal{P}_i x \in Q_i\} \cap \{x \in C : \sum_{i=1}^N \mathcal{L}_i(\mathcal{P}_i x) = y\} \neq \emptyset$.

Given these data, the MSFP is to find an element $p_* \in \Omega$.

Remark 1.1. (a) If $\mathcal{L}_i \equiv 0$ for all $i = 1, 2, \ldots, N$ and $y = 0$, then the MSFP becomes the split feasibility problem with multiple output sets (SFP-MOS, for short), which has been studied in [2,3] (see Section 4 for more details). In particular, when $N = 1$, the SFP-MOS reduces exactly to the classical split feasibility problem (SFP, for short), first introduced by Censor et al. in [4].

* Corresponding author.
E-mail addresses: hans@tnus.edu.vn (N.S. Ha), sreich@technion.ac.il (S. Reich), tuyentm@tnus.edu.vn (T.M. Tuyen), phamthithu@tnut.edu.vn (P.T. Thu).

https://doi.org/10.1016/j.cam.2026.117507
Received 2 December 2024; Received in revised form 14 February 2026
Available online 22 February 2026

Note that this generalization is practically meaningful, since in many real world applications, such as signal recovery, image reconstruction, and other inverse problems, one typically works with multiple sensing operators or observation channels rather than a single one. Consequently, the SFP-MOS provides a more realistic modeling framework than the classical SFP, which justifies the need to develop algorithms specifically tailored to its multi-operator structure.

In addition, there are practical problems that can be modeled within the MSFP framework. For instance, in a multi-stage semi-product manufacturing process, an initial input passes through several processing stages, each governed by its own operational constraints, and the final output must satisfy an aggregate system-wide requirement, such as an overall quality metric or a resource balance constraint.

(b) In the case where $N = 2$, $i \in \{1, 2\}$, $\mathbb{H} = \mathbb{H}_1 \times \mathbb{H}_2$ and $C = Q_1 \times Q_2$, for each $x = (x^{(1)}, x^{(2)}) \in \mathbb{H}$, we define the operators $\mathcal{P}_i : \mathbb{H} \to \mathbb{H}_i$ as follows: $\mathcal{P}_i x := \mathcal{P}_i(x^{(1)}, x^{(2)}) = x^{(i)}$. If we now take $y = 0$ and $\mathcal{L}_1 = -\mathcal{L}_2$, then the MSFP reduces to the split equality problem (SEP, for short) which is considered in [5,6] (see Section 4 for more details).

Both the SFP-MOS and the SEP have drawn a lot of interest from mathematicians around the world mainly due to their applications in intensity-modulated radiation therapy [4,7,8], signal processing and image reconstruction [9], game theory, decomposition methods for PDE, decision sciences, and inertial Nash equilibria [10,11]. Furthermore, the MSFP is a general problem which includes both the SFP-MOS and the SEP. Therefore, studying iterative methods and algorithms for solving the MSFP is an important research topic. Moreover, such methods and algorithms can be applied to solving some other split problems.

To find a solution to the MSFP, Reich et al. [1] introduced two new algorithms which are based on the unconstrained optimization approach. Weak and strong convergence results for the sequences generated by the proposed algorithms were also established under some appropriate conditions involving the control parameters.

In recent years, inertial-type algorithms have attracted much interest, mainly because of their good convergence properties. Such algorithms were initially introduced by Polyak [12], who proposed an inertial scheme in order to improve the convergence speed of the previously known one-step methods for solving evolution equations involving potential operators. A common characteristic of these algorithms is that the next iteration, which depends on the previous two iterations, often has the following form:

$$b_n = a_n + \theta_n(a_n - a_{n-1}) \quad \forall n \geq 1,$$

where a_{n-1} and a_n are given previously, while the inertial factors $\{\theta_n\}$ constitute a real number sequence. The idea of using (more than) two-step inertial extrapolation was also suggested by Polyak [13] in order to accelerate the convergence. However, both the convergence and the convergence rate were not provided. Very recently, several authors have replaced one-step inertial extrapolation with two-step inertial extrapolation (see, for example, Iyiola and Shehu [14], Jolaoso et al. [15]), which has the following form:

$$b_n = a_n + \theta_n(a_n - a_{n-1}) + \delta_n(a_{n-1} - a_{n-2}) \quad \forall n \geq 1,$$

where a_{n-2}, a_{n-1} and a_n are given previously, while the inertial factors $\{\theta_n\}$ and $\{\delta_n\}$ constitute real number sequences. They pointed out that two-step inertial extrapolation could be more beneficial numerically and could accelerate convergence when compared with one-step inertial extrapolation. Note also that inertial-type algorithms have been studied and applied to solving many other problems (see, for instance, Iyiola and Shehu [14], Jolaoso et al. [15], Alvarez [16], Attouch and Cabot [17,18,19], Attouch and László [20], Attouch and Peypouquet [21], Dong et al. [22], Ha et al. [23], Hao and Zhao [24], Sahu et al. [25], Shehu et al. [26], Thong et al. [27], Wang and Yu [28]). These seemingly minor changes turn out to dramatically improve the performance of existing algorithms. Some recent studies in [29–33] have used multistep inertial extrapolation in order to solve monotone inclusion, minimization, fixed point, and split feasibility problems. Furthermore, observe that in many existing methods the inertial factors are often required not only to be diminishing but also to lie within fixed intervals such as $[-1, 1]$ or $[0, a]$ for some positive real number a. Therefore, the following natural questions arise:

Question 1: Can we employ multistep inertial extrapolation in order to devise an algorithm for solving the MSFP?

Question 2: Is it possible to devise a new algorithm with relaxation parameters and a relaxed choice of inertial factors?

Our aim in the present paper is to answer the above questions in the affirmative. To this end, we introduce in this paper two new algorithms for solving the MSFP (see Section 3). Our algorithms are constructed via a combination of hybrid and shrinking projection methods which use multistep inertial extrapolation. The convergence of the sequences generated by our proposed algorithms is established when some mild conditions on the parameters are satisfied. Some corollaries of our main theorems concerning the SFP-MOS and the SEP are presented in Section 4. Finally, in Section 5 we present two numerical examples and compare the effectiveness of our algorithms with that of some known methods.

2. Preliminaries

In this section, we use $\langle \cdot, \cdot \rangle_{\mathbb{H}}$ and $\| \cdot \|_{\mathbb{H}}$ to denote the inner product and the induced norm on the real Hilbert space $\mathbb{H}$, respectively. The symbols $\to$ and $\rightharpoonup$ stand for strong and weak convergence, respectively. We also denote by $I^{\mathbb{H}}$ the identity mapping on $\mathbb{H}$.

Let C be a nonempty, closed and convex subset of a real Hilbert space $\mathbb{H}$. It is well known that for each $x \in \mathbb{H}$, there exists a unique point $P_C^{\mathbb{H}}(x) \in C$ which satisfies

$$\|x - P_C^{\mathbb{H}}(x)\|_{\mathbb{H}} = \inf_{z \in C} \|x - z\|_{\mathbb{H}}. \tag{2.1}$$

The mapping $P_C^{\mathbb{H}} : \mathbb{H} \to C$ defined by (2.1) is called the metric projection of $\mathbb{H}$ onto C.

Recall that an operator $\mathcal{T} : C \to C$ is called nonexpansive if

$$\|\mathcal{T}(x) - \mathcal{T}(y)\|_{\mathbb{H}} \le \|x - y\|_{\mathbb{H}}$$

for all $x, y \in C$. The set of fixed points of an operator $\mathcal{T} : C \to C$ is denoted by $\mathrm{Fix}(\mathcal{T})$, that is, $\mathrm{Fix}(\mathcal{T}) = \{x \in C : \mathcal{T}(x) = x\}$. It is known that the metric projection $P_C^{\mathbb{H}}$ is a nonexpansive mapping and that $\mathrm{Fix}(P_C^{\mathbb{H}}) = C$.

The following lemmas are very useful for proving our main results.

Lemma 2.1. (see [34, Theorem 3.4])
For every x and y in $\mathbb{H}$, we have $y = P_C^{\mathbb{H}}(x)$ if and only if $y \in C$ and

$$\langle x - y, z - y\rangle_{\mathbb{H}} \le 0 \quad \forall z \in C.$$

Lemma 2.2. *Let $P_C^{\mathbb{H}}$ be the metric projection of $\mathbb{H}$ onto C. Then the following statements hold true:*

(i) (see [35, Lemma 2.1])

$$\|x - P_C^{\mathbb{H}}(x)\|_{\mathbb{H}}^2 + \|y - P_C^{\mathbb{H}}(x)\|_{\mathbb{H}}^2 \le \|x - y\|_{\mathbb{H}}^2$$

for all $x \in \mathbb{H}$ and $y \in C$.

(ii) (see [34, Proposition 4.2 and Corollary 4.10])

$$\langle x - y, (I^{\mathbb{H}} - P_C^{\mathbb{H}})(x) - (I^{\mathbb{H}} - P_C^{\mathbb{H}})(y)\rangle_{\mathbb{H}} \ge \|(I^{\mathbb{H}} - P_C^{\mathbb{H}})(x) - (I^{\mathbb{H}} - P_C^{\mathbb{H}})(y)\|_{\mathbb{H}}^2$$

for all $x, y \in \mathbb{H}$. It follows that $I^{\mathbb{H}} - P_C^{\mathbb{H}}$ is a nonexpansive mapping.

Lemma 2.3. (see [34, Corollary 2.42 and Lemma 2.35])
Let $\{x_n\}$ be a sequence in $\mathbb{H}$. Then the following statements hold true:

(i) *If $x_n \rightharpoonup x$ and $\|x_n\|_{\mathbb{H}} \to \|x\|_{\mathbb{H}}$ as $n \to \infty$, then $x_n \to x$ as $n \to \infty$.*
(ii) *If $x_n \rightharpoonup x$ as $n \to \infty$, then $\|x\|_{\mathbb{H}} \le \liminf_{n\to\infty} \|x_n\|_{\mathbb{H}}$.*

Lemma 2.4. (see [34, Theorem 4.17])
Let C be a nonempty, closed, convex and bounded subset of $\mathbb{H}$. Let $\mathcal{T} : C \to \mathbb{H}$ be a nonexpansive mapping. Then the mapping $I^{\mathbb{H}} - \mathcal{T}$ is demiclosed, that is, whenever $\{x_n\}$ is a sequence in C for which $x_n \rightharpoonup x \in C$ and $x_n - \mathcal{T}(x_n) \to y \in \mathbb{H}$, it follows that $x - \mathcal{T}(x) = y$.

3. Main results

From now on, we assume that conditions (A1)–(A4) are fulfilled. For each $x \in \mathbb{H}$, define the function $f : \mathbb{H} \to \mathbb{R}$ by

$$f(x) = \frac{\|(I^{\mathbb{H}} - P_C^{\mathbb{H}})(x)\|_{\mathbb{H}}^2 + \sum_{i=1}^N \|(I^{\mathbb{H}_i} - P_{Q_i}^{\mathbb{H}_i})(\mathcal{P}_i x)\|_{\mathbb{H}_i}^2 + \left\| \sum_{i=1}^N \mathcal{L}_i(\mathcal{P}_i x) - y\right\|_{\mathcal{H}}^2}{2}.$$

It is readily verified that f is convex and Fréchet differentiable on $\mathbb{H}$, and its gradient is given by

$$F(x) = (I^{\mathbb{H}} - P_C^{\mathbb{H}})(x) + \sum_{i=1}^N \mathcal{P}_i^*(I^{\mathbb{H}_i} - P_{Q_i}^{\mathbb{H}_i})(\mathcal{P}_i x) + \sum_{i=1}^N \mathcal{P}_i^*\mathcal{L}_i^*\Big(\sum_{i=1}^N \mathcal{L}_i(\mathcal{P}_i x) - y\Big),$$

where $\mathcal{P}_i^*$ and $\mathcal{L}_i^*$ denote the adjoint operators of $\mathcal{P}_i$ and $\mathcal{L}_i$, respectively (see, for instance, [34, Example 2.48 and Corollary 12.30]). Moreover, it is easy to see that the MSFP is equivalent to the following minimization problem:

$$\min_{x\in\mathbb{H}} f(x).$$

Consequently, a point x_* solves the MSFP if and only if $F(x_*) = 0$. This condition is equivalent to the fixed-point equation

$$x_* = x_* - \lambda F(x_*),$$

for some $\lambda > 0$. Motivated by the iterative structure suggested by this fixed-point formulation, we now introduce the following inertial multistep hybrid projection algorithm.

Suppose that $\{\theta_n^j\}$ is a real bounded sequence for each $j = 1, 2, 3, \dots, k$ and that $\alpha_n \in [\underline{\alpha}, \overline{\alpha}] \subset (0, 2)$ for all $n \in \mathbb{N}$.

Remark 3.1. *In Algorithm 1, we note that $c_n = b_n - \lambda_n F(b_n)$ and $B_n = \|F(b_n)\|^2$. Hence $B_n = 0$ if and only if $F(b_n) = 0$. In this case, (3.3) yields $\lambda_n = 0$, and thus $c_n = b_n$. Conversely, if $B_n > 0$, then $\lambda_n > 0$ and $F(b_n) \ne 0$, so $c_n \ne b_n$. Therefore, within Algorithm 1 the condition $c_n = b_n$ is equivalent to $F(b_n) = 0$, which is equivalent to $b_n \in \Omega$. Consequently, the condition $c_n = b_n$ can be used as a termination criterion, in which case the current iterate b_n is a solution of the MSFP. The same remark applies to Algorithms 2–6.*

The strong convergence of the sequence $\{a_n\}$ generated by Algorithm 1 is established in the following theorem.

Theorem 3.1. *Suppose that conditions (A1)-(A4) hold. For each $j = 1, 2, \dots, k$, let $\{\theta_n^j\}$ be a bounded real sequence, and let $\alpha_n \in [\underline{\alpha}, \overline{\alpha}] \subset (0, 2)$ for all $n \ge 0$. Then the sequence $\{a_n\}$ generated by Algorithm 1 converges strongly to $a_* = P_\Omega^{\mathbb{H}}(a_0)$.*

Algorithm 1

Step 1. Choose $a_{-k}, a_{-k+1}, a_{-k+2}, \dots, a_0 \in \mathbb{H}$ arbitrarily, and set $n := 0$. **Step 2.** Given $a_{n-k}, a_{n-k+1}, \dots, a_n$, compute

$$b_n = a_n + \sum_{j=1}^{k} \theta_n^j (a_{n-k+j} - a_{n-k+j-1}). \tag{3.1}$$

Step 3. Compute

$$c_n = b_n - \lambda_n F(b_n), \tag{3.2}$$

where the parameter λ_n is defined by

$$\lambda_n = \begin{cases} 0, & \text{if } B_n = 0, \\ \alpha_n \dfrac{A_n}{B_n}, & \text{otherwise,} \end{cases} \tag{3.3}$$

where

$$A_n := \|(I^{\mathbb{H}} - P_C^{\mathbb{H}})(b_n)\|_{\mathbb{H}}^2 + \sum_{i=1}^{N} \|(I^{\mathbb{H}_i} - P_{Q_i}^{\mathbb{H}_i})(\mathcal{P}_i b_n)\|_{\mathbb{H}_i}^2 + \|\sum_{i=1}^{N} \mathcal{L}_i(\mathcal{P}_i b_n) - y\|_{\mathcal{H}}^2,$$

$$B_n := \|(I^{\mathbb{H}} - P_C^{\mathbb{H}})(b_n) + \sum_{i=1}^{N} \mathcal{P}_i^* (I^{\mathbb{H}_i} - P_{Q_i}^{\mathbb{H}_i})(\mathcal{P}_i b_n) + \sum_{i=1}^{N} \mathcal{P}_i^* \mathcal{L}_i^* (\sum_{i=1}^{N} \mathcal{L}_i(\mathcal{P}_i b_n) - y)\|_{\mathbb{H}}^2.$$

Step 4. Define two closed half-spaces U_n and V_n as follows:

$$U_n = \{z \in \mathbb{H} : \|c_n - z\|_{\mathbb{H}} \le \|b_n - z\|_{\mathbb{H}}\}, \tag{3.4}$$

$$V_n = \{z \in \mathbb{H} : \langle z - a_n, a_0 - a_n\rangle_{\mathbb{H}} \le 0\}. \tag{3.5}$$

Step 5. Compute

$$a_{n+1} = P_{U_n \cap V_n}^{\mathbb{H}}(a_0). \tag{3.6}$$

Step 6. Set $n \leftarrow n+1$ and go to Step 2.

Proof. We divide the proof into several steps. **Claim 1.** We have $\Omega \subset U_n \cap V_n$ for all $n \ge 0$.

First, it is not difficult to see that the sets U_n and V_n can equivalently be redefined as follows:

$$U_n = \{z \in \mathbb{H} : \langle b_n - c_n, z\rangle_{\mathbb{H}} \le \frac{1}{2}(\|b_n\|_{\mathbb{H}}^2 - \|c_n\|_{\mathbb{H}}^2)\},$$
$$V_n = \{z \in \mathbb{H} : \langle a_0 - a_n, z\rangle_{\mathbb{H}} \le \langle a_0 - a_n, a_n\rangle_{\mathbb{H}}\}.$$

Thus, both of them are two closed half-spaces of $\mathbb{H}$ for each $n \ge 0$.

We now take any $x \in \Omega$, that is, $x \in C$ and

$$\mathcal{P}_i(x) \in Q_i \quad \forall i = 1, 2, 3, \dots, N, \qquad \sum_{i=1}^{N} \mathcal{L}_i(\mathcal{P}_i x) = y.$$

Using (3.2), we have

$$\begin{aligned} \|c_n - x\|_{\mathbb{H}}^2 &= \|b_n - x - \lambda_n F(b_n)\|_{\mathbb{H}}^2 \\ &= \|b_n - x\|_{\mathbb{H}}^2 - 2\lambda_n \langle F(b_n), b_n - x\rangle_{\mathbb{H}} + \lambda_n^2 \|F(b_n)\|_{\mathbb{H}}^2. \end{aligned} \tag{3.7}$$

We also note that

$$\begin{aligned} \langle F(b_n), b_n - x\rangle_{\mathbb{H}} = &\langle (I^{\mathbb{H}} - P_C^{\mathbb{H}})(b_n), b_n - x\rangle_{\mathbb{H}} \\ &+ \sum_{i=1}^{N} \langle \mathcal{P}_i^* (I^{\mathbb{H}_i} - P_{Q_i}^{\mathbb{H}_i})(\mathcal{P}_i b_n), b_n - x\rangle_{\mathbb{H}} \\ &+ \sum_{i=1}^{N} \langle \mathcal{P}_i^* \mathcal{L}_i^* (\sum_{i=1}^{N} \mathcal{L}_i(\mathcal{P}_i b_n) - y), b_n - x\rangle_{\mathbb{H}} \end{aligned}$$

$$\begin{aligned}&= \langle (I^{\mathbb{H}} - P_C^{\mathbb{H}})(b_n), b_n - x\rangle_{\mathbb{H}} \\ &+ \sum_{i=1}^{N} \langle (I^{\mathbb{H}_i} - P_{Q_i}^{\mathbb{H}_i})(\mathcal{P}_i b_n), \mathcal{P}_i b_n - \mathcal{P}_i x\rangle_{\mathbb{H}_i} \\ &+ \langle \sum_{i=1}^{N} \mathcal{L}_i(\mathcal{P}_i b_n) - y, \sum_{i=1}^{N} \mathcal{L}_i(\mathcal{P}_i b_n) - \sum_{i=1}^{N} \mathcal{L}_i(\mathcal{P}_i x)\rangle_{\mathcal{H}} \\ &= \langle (I^{\mathbb{H}} - P_C^{\mathbb{H}})(b_n) - (I^{\mathbb{H}} - P_C^{\mathbb{H}})(x), b_n - x\rangle_{\mathbb{H}} \\ &+ \sum_{i=1}^{N} \langle (I^{\mathbb{H}_i} - P_{Q_i}^{\mathbb{H}_i})(\mathcal{P}_i b_n) - (I^{\mathbb{H}_i} - P_{Q_i}^{\mathbb{H}_i})(\mathcal{P}_i x), \mathcal{P}_i b_n - \mathcal{P}_i x\rangle_{\mathbb{H}_i} \\ &+ \| \sum_{i=1}^{N} \mathcal{L}_i(\mathcal{P}_i b_n) - y\|_{\mathcal{H}}^2. \end{aligned} \tag{3.8}$$

Using Lemma 2.2 (ii) and (3.8), we find that

$$\begin{aligned}\langle F(b_n), b_n - x\rangle_{\mathbb{H}} &\geq \|(I^{\mathbb{H}} - P_C^{\mathbb{H}})(b_n) - (I^{\mathbb{H}} - P_C^{\mathbb{H}})(x)\|_{\mathbb{H}}^2 \\ &+ \sum_{i=1}^{N} \|(I^{\mathbb{H}_i} - P_{Q_i}^{\mathbb{H}_i})(\mathcal{P}_i b_n) - (I^{\mathbb{H}_i} - P_{Q_i}^{\mathbb{H}_i})(\mathcal{P}_i x)\|_{\mathbb{H}_i}^2 \\ &+ \| \sum_{i=1}^{N} \mathcal{L}_i(\mathcal{P}_i b_n) - y\|_{\mathcal{H}}^2 \\ &= \|(I^{\mathbb{H}} - P_C^{\mathbb{H}})(b_n)\|_{\mathbb{H}}^2 + \sum_{i=1}^{N} \|(I^{\mathbb{H}_i} - P_{Q_i}^{\mathbb{H}_i})(\mathcal{P}_i b_n)\|_{\mathbb{H}_i}^2 \\ &+ \| \sum_{i=1}^{N} \mathcal{L}_i(\mathcal{P}_i b_n) - y\|_{\mathcal{H}}^2 \\ &= A_n. \end{aligned} \tag{3.9}$$

We also observe that

$$B_n = \|F(b_n)\|_{\mathbb{H}}^2. \tag{3.10}$$

Thus, using (3.7), (3.9) and (3.10), we obtain

$$\|c_n - x\|_{\mathbb{H}}^2 \leq \|b_n - x\|_{\mathbb{H}}^2 - 2\lambda_n A_n + \lambda_n^2 B_n. \tag{3.11}$$

It follows from (3.3) and (3.11) that

$$\|c_n - x\|_{\mathbb{H}}^2 \leq \begin{cases} \|b_n - x\|_{\mathbb{H}}^2, & \text{if } \lambda_n = 0, \\ \|b_n - x\|_{\mathbb{H}}^2 - \alpha_n(2-\alpha_n)\dfrac{A_n^2}{B_n}, & \text{if } \lambda_n = \alpha_n \dfrac{A_n}{B_n}. \end{cases} \tag{3.12}$$

Since $0 < \underline{\alpha} \leq \alpha_n \leq \overline{\alpha} < 2$, in both cases we have

$$\|c_n - x\|_{\mathbb{H}}^2 \leq \|b_n - x\|_{\mathbb{H}}^2 \quad \forall n \geq 0. \tag{3.13}$$

Hence, using (3.4) and (3.13), we find that $x \in U_n$ for all $n \geq 0$, that is, $\Omega \subset U_n$ for all $n \geq 0$.

Next, with $n = 0$, it is clear that $\Omega \subset V_0 = \mathbb{H}$. Suppose that $\Omega \subset V_n$ for some $n \geq 0$. Using Lemma 2.1 and the fact that $a_{n+1} = P_{U_n \cap V_n}^{\mathbb{H}}(a_0)$, we have

$$\langle a_0 - a_{n+1}, z - a_{n+1}\rangle_{\mathbb{H}} \leq 0 \quad \forall z \in U_n \cap V_n.$$

Thus, it follows from $x \in \Omega \subset U_n \cap V_n$ that

$$\langle a_0 - a_{n+1}, x - a_{n+1}\rangle_{\mathbb{H}} \leq 0.$$

This implies that $x \in V_{n+1}$. Employing mathematical induction, we now conclude that $x \in V_n$ for all $n \geq 0$, that is, $\Omega \subset V_n$ for all $n \geq 0$.

Therefore, we find that $\Omega \subset U_n \cap V_n$ for all $n \geq 0$, as claimed. In addition, this also shows that the sequence $\{a_n\}$ is well defined.

Claim 2. The sequence $\{a_n\}$ is bounded.

It follows from Lemma 2.1 and (3.5) that $a_n = P_{V_n}^{\mathbb{H}}(a_0)$. Since $a_* = P_{\Omega}^{\mathbb{H}}(a_0) \in \Omega \subset V_n$, we see that

$$\|a_n - a_0\|_{\mathbb{H}} \leq \|a_* - a_0\|_{\mathbb{H}} \quad \forall n \geq 0, \tag{3.14}$$

which implies that the sequence $\{a_n\}$ is bounded. Hence, the sequences $\{b_n\}$ and $\{c_n\}$ are also bounded thanks to (3.1) and (3.13).

Claim 3. The following limits exist:

$$\lim_{n\to\infty} \|a_{n+1} - a_n\|_{\mathbb{H}} = 0, \tag{3.15}$$

$$\lim_{n\to\infty} \|a_{n+1} - b_n\|_{\mathbb{H}} = 0, \tag{3.16}$$

$$\lim_{n\to\infty} \|b_n - c_n\|_{\mathbb{H}} = 0. \tag{3.17}$$

From (3.6), we observe that $a_{n+1} \in U_n \cap V_n \subset V_n$. Applying Lemma 2.2 (i) with $x = a_0$, $C = V_n$, $y = a_{n+1}$, and $P^{\mathbb{H}}_C(x) = P^{\mathbb{H}}_{V_n}(a_0) = a_n$, we obtain

$$\|a_{n+1} - a_n\|^2_{\mathbb{H}} \le \|a_{n+1} - a_0\|^2_{\mathbb{H}} - \|a_n - a_0\|^2_{\mathbb{H}}, \tag{3.18}$$

which implies that the sequence $\{\|a_n - a_0\|_{\mathbb{H}}\}$ is increasing for all $n \ge 0$. Combining this fact with Claim 2, we see that there exists the finite limit

$$\lim_{n\to\infty} \|a_n - a_0\|_{\mathbb{H}}.$$

Thus, we have verified the limit (3.15), as claimed, thanks to (3.18). Besides, we observe that

$$\begin{aligned}
\|a_{n+1} - b_n\|_{\mathbb{H}} &= \|a_{n+1} - (a_n + \sum_{j=1}^{k} \theta_n^j (a_{n-k+j} - a_{n-k+j-1}))\|_{\mathbb{H}} \\
&= \|(a_{n+1} - a_n) - \sum_{j=1}^{k} \theta_n^j (a_{n-k+j} - a_{n-k+j-1})\|_{\mathbb{H}} \\
&\le \|(a_{n+1} - a_n)\|_{\mathbb{H}} + \sum_{j=1}^{k} |\theta_n^j| \|a_{n-k+j} - a_{n-k+j-1}\|_{\mathbb{H}}.
\end{aligned}$$

Using (3.15) and the boundedness of the real number sequences $\{\theta_n^j\}$, we obtain the existence of the limit (3.16), as asserted.

Furthermore, since $a_{n+1} \in U_n$, we have

$$\|a_{n+1} - c_n\|_{\mathbb{H}} \le \|a_{n+1} - b_n\|_{\mathbb{H}}.$$

Combining this with (3.16), we find that $\|a_{n+1} - c_n\|_{\mathbb{H}} \to 0$. Using the estimate

$$\|b_n - c_n\|_{\mathbb{H}} \le \|a_{n+1} - b_n\|_{\mathbb{H}} + \|a_{n+1} - c_n\|_{\mathbb{H}},$$

we obtain the limit (3.17), as claimed.

Claim 4. The sequence $\{a_n\}$ converges strongly to $a_* \in \Omega$.

We consider the following two cases:

Case 1. $\lambda_n = 0$.

In this case, it follows from (3.10) that

$$B_n = \|F(b_n)\|^2_{\mathbb{H}} = 0.$$

This implies that $F(b_n) = 0$. Using (3.9), we find that $A_n = 0$.

Case 2. $\lambda_n = \alpha_n \dfrac{A_n}{B_n}$.

Using (3.12), for any $x \in \Omega$, we have

$$\begin{aligned}
\alpha_n(2-\alpha_n)\frac{A_n^2}{B_n} &\le \|b_n - x\|^2_{\mathbb{H}} - \|c_n - x\|^2_{\mathbb{H}} \\
&\le M_1(\|b_n - x\|_{\mathbb{H}} - \|c_n - x\|_{\mathbb{H}}) \\
&\le M_1(\|b_n - c_n\|_{\mathbb{H}}),
\end{aligned}$$

where $M_1 = \sup_n\{\|b_n - x\|_{\mathbb{H}} + \|c_n - x\|_{\mathbb{H}}\}$. Since $0 < \underline{\alpha} \le \alpha_n \le \overline{\alpha} < 2$ and $\|b_n - c_n\|_{\mathbb{H}} \to 0$, we infer that

$$\frac{A_n^2}{B_n} \to 0. \tag{3.19}$$

By using the Cauchy-Schwarz inequality, we obtain

$$
\begin{aligned}
B_n &= \|(I^{\mathbb{H}} - P_C^{\mathbb{H}})(b_n) + \sum_{i=1}^N \mathcal{P}_i^*(I^{\mathbb{H}_i} - P_{Q_i}^{\mathbb{H}_i})(\mathcal{P}_i b_n) + \sum_{i=1}^N \mathcal{P}_i^* \mathcal{L}_i^*(\sum_{i=1}^N \mathcal{L}_i(\mathcal{P}_i b_n) - y)\|_{\mathbb{H}}^2 \\
&\le \Bigg(\|(I^{\mathbb{H}} - P_C^{\mathbb{H}})(b_n)\|_{\mathbb{H}} + \|\sum_{i=1}^N \mathcal{P}_i^*(I^{\mathbb{H}_i} - P_{Q_i}^{\mathbb{H}_i})(\mathcal{P}_i b_n)\|_{\mathbb{H}} \\
&\qquad + \|\sum_{i=1}^N \mathcal{P}_i^* \mathcal{L}_i^*(\sum_{i=1}^N \mathcal{L}_i(\mathcal{P}_i b_n) - y)\|_{\mathbb{H}} \Bigg)^2 \\
&\le 3\Bigg(\|(I^{\mathbb{H}} - P_C^{\mathbb{H}})(b_n)\|_{\mathbb{H}}^2 + \|\sum_{i=1}^N \mathcal{P}_i^*(I^{\mathbb{H}_i} - P_{Q_i}^{\mathbb{H}_i})(\mathcal{P}_i b_n)\|_{\mathbb{H}}^2 \\
&\qquad + \|\sum_{i=1}^N \mathcal{P}_i^* \mathcal{L}_i^*(\sum_{i=1}^N \mathcal{L}_i(\mathcal{P}_i b_n) - y)\|_{\mathbb{H}}^2 \Bigg) \\
&\le 3\Bigg(\|(I^{\mathbb{H}} - P_C^{\mathbb{H}})(b_n)\|_{\mathbb{H}}^2 + \left(\sum_{i=1}^N \|\mathcal{P}_i^*\| \|(I^{\mathbb{H}_i} - P_{Q_i}^{\mathbb{H}_i})(\mathcal{P}_i b_n)\|_{\mathbb{H}_i} \right)^2 \\
&\qquad + \left(\sum_{i=1}^N \|\mathcal{P}_i^* \mathcal{L}_i^*\| \left\| \sum_{i=1}^N \mathcal{L}_i(\mathcal{P}_i b_n) - y \right\|_{\mathcal{H}} \right)^2 \Bigg) \\
&\le 3 \max \left\{ 1, N \max_{1\le i\le N} \|\mathcal{P}_i\|^2, N^2 \max_{1\le i\le N} \|\mathcal{L}_i \mathcal{P}_i\|^2 \right\} A_n = M_2 A_n,
\end{aligned} \tag{3.20}
$$

where $M_2 = 3\max\left\{1, N\max_{1\le i\le N}\|\mathcal{P}_i\|^2, N^2\max_{1\le i\le N}\|\mathcal{L}_i\mathcal{P}_i\|^2\right\} > 0$. It now follows from (3.19) and (3.20) that $A_n \to 0$.

Therefore, in both cases, we have $A_n \to 0$, which leads to

$$\|(I^{\mathbb{H}} - P_C^{\mathbb{H}})(b_n)\|_{\mathbb{H}} \to 0, \tag{3.21}$$

$$\|(I^{\mathbb{H}_i} - P_{Q_i}^{\mathbb{H}_i})(\mathcal{P}_i b_n)\|_{\mathbb{H}_i}^2 \to 0 \quad \forall i = 1, 2, 3, \dots, N, \tag{3.22}$$

$$\|\sum_{i=1}^N \mathcal{L}_i(\mathcal{P}_i b_n) - y\|_{\mathcal{H}}^2 \to 0. \tag{3.23}$$

Next, since $\{a_n\}$ is a bounded sequence, there exists a subsequence $\{a_{n_k}\}$ of $\{a_n\}$ which converges weakly to some $x_* \in \mathbb{H}$, that is,

$$a_{n_k} \rightharpoonup x_*.$$

Let $\{b_{n_k}\}$ be any such weakly convergent subsequence. Using (3.15) and (3.16), we find that $\|a_n - b_n\| \to 0$. Thus, we conclude that

$$b_{n_k} \rightharpoonup x_*.$$

In view of Lemma 2.4 and (3.21), we infer that $x_* \in C$. In addition, since $\mathcal{P}_i$ and $\mathcal{L}_i$ are bounded linear operators, we find that

$$\mathcal{P}_i b_{n_k} \rightharpoonup \mathcal{P}_i x_*, \quad \mathcal{L}_i(\mathcal{P}_i b_{n_k}) \rightharpoonup \mathcal{L}_i(\mathcal{P}_i x_*)$$

for all $i = 1, 2, \dots, N$. Thus, using (3.22) and Lemma 2.4, we infer that $\mathcal{P}_i x_* \in Q_i$ for each $i = 1, 2, \dots, N$. And, from (3.23) it follows that $\sum_{i=1}^N \mathcal{L}_i(\mathcal{P}_i x_*) = y$. Thus, we find that $x_* \in \Omega$.

Finally, since $a_* = P_\Omega^{\mathbb{H}}(a_0)$ and $x_* \in \Omega$, using (3.14) and Lemma 2.3 (i), we have

$$
\begin{aligned}
\|a_* - a_0\|_{\mathbb{H}} &\le \|x_* - a_0\|_{\mathbb{H}} \le \liminf_{k\to\infty} \|a_{n_k} - a_0\|_{\mathbb{H}} \\
&\le \limsup_{k\to\infty} \|a_{n_k} - a_0\|_{\mathbb{H}} \le \|a_* - a_0\|_{\mathbb{H}}.
\end{aligned}
$$

Using the uniqueness of the nearest point a_*, we find that $a_* = x_*$. Moreover, we also have

$$\|a_{n_k} - a_0\|_{\mathbb{H}} \to \|a_* - a_0\|_{\mathbb{H}}.$$

In view of Lemma 2.3 (ii), we obtain $a_{n_k} \to a_*$. Using again the uniqueness of a_*, we conclude that $a_n \to a_*$ as $n \to \infty$. This completes the proof. □

We now propose a modification of Algorithm 1 by combining it with the shrinking projection method. Our second algorithm is formulated as follows:

Theorem 3.2. *Suppose that conditions (A1)-(A4) hold. For each $j = 1, 2, \dots, k$, let $\{\theta_n^j\}$ be a bounded real sequence, and let $\alpha_n \in [\underline{\alpha}, \overline{\alpha}] \subset (0, 2)$ for all $n \ge 0$. Then the sequence $\{a_n\}$ defined by Algorithm 2 converges strongly to $a_* = P_\Omega^{\mathbb{H}}(a_0)$.*

Algorithm 2

Step 1. Choose $a_{-k}, a_{-k+1}, a_{-k+2}, \dots, a_0 \in \mathbb{H}$ arbitrarily, $W_{-1} = \mathbb{H}$ and set $n = 0$.
Step 2. Given $a_{n-k}, a_{n-k+1}, \dots, a_n$, compute b_n as in (3.1).
Step 3. Compute c_n and λ_n as in (3.2) and (3.3).
Step 4. Define a subset W_n as follows:

$$W_n = \{z \in W_{n-1} : \|c_n - z\|_{\mathbb{H}} \le \|b_n - z\|_{\mathbb{H}}\}.$$

Step 5. Compute $a_{n+1} = P^{\mathbb{H}}_{W_n}(a_0)$.
Step 6. Set $n \leftarrow n+1$, and go to Step 2.

Proof. We divide the proof into several steps. **Claim 1.** We have $\Omega \subset W_n$ for all $n \ge 0$.

It is not difficult to see that W_n is a closed and convex subset of $\mathbb{H}$ for all $n \ge 0$. Employing an argument similar to the one used in the proof of Claim 1 of the proof of Theorem 3.1, and using mathematical induction, we also conclude that $\Omega \subset W_n$ for all $n \ge 0$. Hence, the sequence $\{a_n\}$ is well defined.

Claim 2. The sequence $\{a_n\}$ converges strongly to an element $x_* \in \mathbb{H}$.

For each $x \in \Omega \subset W_n$, it follows from $a_{n+1} = P^{\mathbb{H}}_{W_n}(a_0)$ that

$$\|a_{n+1} - a_0\|_{\mathbb{H}} \le \|x - a_0\|_{\mathbb{H}}, \tag{3.24}$$

which implies that the sequence $\{a_n\}$ is bounded.

For any $m \ge n \ge 1$, since $a_m \in W_{m-1} \subset W_{n-1}$ and $a_n = P^{\mathbb{H}}_{W_{n-1}}(a_0)$, in view of Lemma 2.2 (i), we find that

$$\|a_m - a_n\|^2_{\mathbb{H}} \le \|a_m - a_0\|^2_{\mathbb{H}} - \|a_n - a_0\|^2_{\mathbb{H}} \to 0 \tag{3.25}$$

as $n, m \to \infty$. This ensures that $\{a_n\}$ is a Cauchy sequence. Hence, the sequence $\{a_n\}$ converges strongly to an element $x_* \in \mathbb{H}$.

Claim 3. The following limits exist:

$$\lim_{n\to\infty} \|a_{n+1} - a_n\|_{\mathbb{H}} = 0, \tag{3.26}$$

$$\lim_{n\to\infty} \|a_{n+1} - b_n\|_{\mathbb{H}} = 0, \tag{3.27}$$

$$\lim_{n\to\infty} \|b_n - c_n\|_{\mathbb{H}} = 0, \tag{3.28}$$

$$\lim_{n\to\infty} \|a_n - b_n\|_{\mathbb{H}} = 0. \tag{3.29}$$

Using (3.25), we get the limit (3.26), as claimed. Furthermore, by employing an argument similar to the one used in the proof of Claim 3 of the proof of Theorem 3.1, we also obtain the limits (3.27) and (3.28) (because $a_{n+1} \in W_n$). Using (3.26) and (3.27), we also obtain the last limit (3.29), as asserted.

Claim 4. The sequence $\{a_n\}$ converges strongly to $a_* = P^{\mathbb{H}}_{\Omega}(a_0)$.

Since $a_n \to x_*$, we have $b_n \to x_*$ thanks to (3.29). Employing an argument similar to the one used in the proof Claim 4 of the proof of Theorem 3.1, we find that $x_* \in \Omega$.

Finally, in (3.24), letting $n \to \infty$, we infer that $\|x_* - a_0\|_{\mathbb{H}} \le \|a_* - a_0\|_{\mathbb{H}}$. It now follows from the properties of the metric projection that $x_* = a_*$.

This completes the proof. ☐

Remark 3.2. *To compute the projection in (3.6) of Algorithm 1, we can make use of the explicit formula provided in [34, Propositions 28.18 and 28.19]. In addition, we observe that both $P^{\mathbb{H}}_{U_n \cap V_n}(a_0)$ and $P^{\mathbb{H}}_{W_n}(a_0)$ involve projecting onto the intersection of closed half-spaces, which is equivalent to solving a convex optimization problem over a polyhedral feasible region. Consequently, these projections can be computed in an efficient and reliable manner using the built-in MATLAB solver `quadprog`.*

4. Corollaries

Let $\mathbb{H}$, $\mathbb{H}_i$ $(i = 1, 2, 3, \dots, N)$, and $\mathcal{H}$ be real Hilbert spaces. Let $C \subset \mathbb{H}$ and $Q_i \subset \mathbb{H}_i$ be nonempty, closed, and convex sets. For each $i = 1, 2, 3, \dots, N$, let $\mathcal{P}_i : \mathbb{H} \to \mathbb{H}_i$ and $\mathcal{L}_i : \mathbb{H}_i \to \mathcal{H}$ be bounded linear operators.

Remark 4.1. *The split feasibility problem with multiple output sets (SFP-MOS, for short) is stated as follows:*

$$\text{Find an element } x^\dagger \in C \text{ such that } \mathcal{P}_i(x^\dagger) \in Q_i \quad \forall i = 1, 2, 3, \dots, N.$$

We denote the solution set of the SFP-MOS by $\widehat{\Omega}$ and assume that $\widehat{\Omega} \ne \emptyset$. It is not difficult to see that if we choose $\mathcal{L}_i \equiv 0$ for all $i = 1, 2, 3, \dots, N$ and $y = 0$, then the MSFP becomes the SFP-MOS. Therefore, we can use Algorithms 1 and 2 to find a solution to the SFP-MOS.

For each $x \in \mathbb{H}$, we now put

$$\widehat{F}(x) := (I^{\mathbb{H}} - P^{\mathbb{H}}_C)(x) + \sum_{i=1}^{N} \mathcal{P}_i^*(I^{\mathbb{H}_i} - P^{\mathbb{H}_i}_{Q_i})(\mathcal{P}_i x).$$

The following algorithm is obtained from Algorithm 1.

Algorithm 3

Step 1. Choose $a_{-k}, a_{-k+1}, a_{-k+2}, \dots, a_0 \in \mathbb{H}$ arbitrarily, and set $n := 0$.
Step 2. Given $a_{n-k}, a_{n-k+1}, \dots, a_n$, compute b_n as in (3.1).
Step 3. Compute

$$c_n = b_n - \widehat{\lambda}_n \widehat{F}(b_n), \tag{4.1}$$

where the parameter $\widehat{\lambda}_n$ is defined by

$$\widehat{\lambda}_n = \begin{cases} 0, & \text{if } \widehat{B}_n = 0, \\ \alpha_n \dfrac{\widehat{A}_n}{\widehat{B}_n}, & \text{otherwise,} \end{cases} \tag{4.2}$$

where

$$\widehat{A}_n := \|(I^{\mathbb{H}} - P^{\mathbb{H}}_C)(b_n)\|^2_{\mathbb{H}} + \sum_{i=1}^{N} \|(I^{\mathbb{H}_i} - P^{\mathbb{H}_i}_{Q_i})(\mathcal{P}_i b_n)\|^2_{\mathbb{H}_i},$$

$$\widehat{B}_n := \|(I^{\mathbb{H}} - P^{\mathbb{H}}_C)(b_n) + \sum_{i=1}^{N} \mathcal{P}_i^*(I^{H_i} - P^{\mathbb{H}_i}_{Q_i})(\mathcal{P}_i b_n)\|^2_{\mathbb{H}}.$$

Step 4. Define two closed half-spaces U_n and V_n as follows:

$$U_n = \{z \in \mathbb{H} : \|c_n - z\|_{\mathbb{H}} \le \|b_n - z\|_{\mathbb{H}}\},$$
$$V_n = \{z \in \mathbb{H} : \langle z - a_n, a_0 - a_n\rangle_{\mathbb{H}} \le 0\}.$$

Step 5. Compute $a_{n+1} = P^{\mathbb{H}}_{U_n \cap V_n}(a_0)$.
Step 6. Set $n \leftarrow n+1$ and go to Step 2.

Corollary 4.1. *For each $j = 1, 2, \dots, k$, let $\{\theta_n^j\}$ be a bounded real sequence, and let $\alpha_n \in [\underline{\alpha}, \overline{\alpha}] \subset (0, 2)$ for all $n \ge 0$. Then the sequence $\{a_n\}$ generated by Algorithm 3 converges strongly to the unique solution to the equation $a_* = P^{\mathbb{H}}_{\widehat{\Omega}}(a_0)$.*

Our next algorithm is obtained from Algorithm 2.

Algorithm 4

Step 1. Choose $a_{-k}, a_{-k+1}, a_{-k+2}, \dots, a_0 \in \mathbb{H}$ arbitrarily, $W_{-1} = \mathbb{H}$ and set $n = 0$.
Step 2. Given $a_{n-k}, a_{n-k+1}, \dots, a_n$, compute b_n as in (3.1).
Step 3. Compute c_n and $\widehat{\lambda}_n$ as in (4.1) and (4.2).
Step 4. Define a subset W_n as follows:

$$W_n = \{z \in W_{n-1} : \|c_n - z\|_{\mathbb{H}} \le \|b_n - z\|_{\mathbb{H}}\}.$$

Step 5. Compute $a_{n+1} = P^{\mathbb{H}}_{W_n}(a_0)$.
Step 6. Set $n \leftarrow n+1$, and go to Step 2.

Corollary 4.2. *For each $j = 1, 2, \dots, k$, let $\{\theta_n^j\}$ be a bounded real sequence, and let $\alpha_n \in [\underline{\alpha}, \overline{\alpha}] \subset (0, 2)$ for all $n \ge 0$. Then the sequence $\{a_n\}$ generated by Algorithm 4 converges strongly to the unique solution to the equation $a_* = P^{\mathbb{H}}_{\widehat{\Omega}}(a_0)$.*

Remark 4.2. We now consider the case where $\mathbb{H} = \mathbb{H}_1 \times \mathbb{H}_2 \times \dots \times \mathbb{H}_N$ and $C = Q_1 \times Q_2 \times \dots \times Q_N$, while the operator $\mathcal{P}_i : \mathbb{H} \to \mathbb{H}_i$ is defined by

$$\mathcal{P}_i(x^{(1)}, \dots, x^{(i)}, \dots, x^{(N)}) = x^{(i)} \tag{4.3}$$

for each $x = (x^{(1)}, \dots, x^{(i)}, \dots, x^{(N)}) \in \mathbb{H}$. Then the MSFP becomes the following problem:

$$\text{Find an element } x \in C \text{ such that } \sum_{i=1}^{N} \mathcal{L}_i(x^{(i)}) = y. \tag{4.4}$$

We denote the solution set of Problem (4.4) by $\tilde{\Omega}$ and assume that $\tilde{\Omega} \neq \emptyset$. It is not difficult to see that if $N = 2$ and $y = 0$, then Problem (4.4) can be reduced to the SEP, that is,

$$\text{Find an element } (u, v) \in Q_1 \times Q_2 \text{ such that } A(u) = B(v), \tag{4.5}$$

where $A = \mathcal{L}_1$ and $B = -\mathcal{L}_2$. Therefore, we can apply Algorithms 1 and 2 to solving Problem (4.4) and Problem (4.5).

We also note that in this case the metric projection can be computed as follows:

$$P_C^{\mathbb{H}}(x) = (P_{Q_1}^{\mathbb{H}}(x^{(1)}), P_{Q_2}^{\mathbb{H}}(x^{(2)}), \dots, P_{Q_N}^{\mathbb{H}}(x^{(N)}))$$

for all $x = (x^{(1)}, x^{(2)}, \dots, x^{(N)}) \in \mathbb{H}$ (see, for example, [34, Proposition 28.3]).

For each $x = (x^{(1)}, \dots, x^{(i)}, \dots, x^{(N)}) \in \mathbb{H}$, we now put

$$\begin{aligned}
&F_1(x) = (x^{(1)} - P_{Q_1}^{\mathbb{H}}(x^{(1)}), \dots, x^{(i)} - P_{Q_i}^{\mathbb{H}}(x^{(i)}), \dots, x^{(N)} - P_{Q_N}^{\mathbb{H}}(x^{(N)})),\\
&F_2(x) = (\mathcal{L}_1^*(\sum_{i=1}^{N} \mathcal{L}_i(x^{(i)}) - y), \dots, \mathcal{L}_i^*(\sum_{i=1}^{N} \mathcal{L}_i(x^{(i)}) - y), \dots, \mathcal{L}_N^*(\sum_{i=1}^{N} \mathcal{L}_i(x^{(i)}) - y)),\\
&\tilde{F}(x) = 2F_1(x) + F_2(x).
\end{aligned}$$

The following algorithm is obtained from Algorithm 1.

Algorithm 5

Step 1. Choose $a_{-k}, a_{-k+1}, a_{-k+2}, \dots, a_0 \in \mathbb{H}$ arbitrarily, and set $n := 0$.
Step 2. Given $a_{n-k}, a_{n-k+1}, \dots, a_n$, compute b_n as in (3.1).
Step 3. Compute

$$c_n = b_n - \tilde{\lambda}_n \tilde{F}(b_n), \tag{4.6}$$

where the parameter $\tilde{\lambda}_n$ is defined by

$$\tilde{\lambda}_n = \begin{cases} 0, & \text{if } \tilde{B}_n = 0, \\ \alpha_n \dfrac{\tilde{A}_n}{\tilde{B}_n}, & \text{otherwise}, \end{cases} \tag{4.7}$$

where

$$\begin{aligned}
\tilde{A}_n &:= 2\|F_1(b_n)\|_{\mathbb{H}}^2 + \|\sum_{i=1}^{N} \mathcal{L}_i(b_n^{(i)}) - y\|_{\mathcal{H}}^2,\\
\tilde{B}_n &:= \|2F_1(b_n) + F_2(b_n)\|_{\mathbb{H}}^2.
\end{aligned}$$

Step 4. Define two closed half-spaces U_n and V_n as follows:

$$\begin{aligned}
U_n &= \{z \in \mathbb{H} : \|c_n - z\|_{\mathbb{H}} \le \|b_n - z\|_{\mathbb{H}}\},\\
V_n &= \{z \in \mathbb{H} : \langle z - a_n, a_0 - a_n \rangle_{\mathbb{H}} \le 0\}.
\end{aligned}$$

Step 5. Compute $a_{n+1} = P_{U_n \cap V_n}^{\mathbb{H}}(a_0)$.
Step 6. Set $n \leftarrow n + 1$ and go to Step 2.

Corollary 4.3. *Let $\mathbb{H}$, C, and $\mathcal{P}_i$ $(i = 1, 2, 3, \dots, N)$ be as in Remark 4.2. For each $j = 1, 2, \dots, k$, let $\{\theta_n^j\}$ be a bounded real sequence, and let $\alpha_n \in [\underline{\alpha}, \overline{\alpha}] \subset (0, 2)$ for all $n \ge 0$. Then the sequence $\{a_n\}$ generated by Algorithm 5 converges strongly to the unique solution to the equation $a_* = P_{\tilde{\Omega}}^{\mathbb{H}}(a_0)$.*

Algorithm 6, our next algorithm, is obtained from Algorithm 2.

Corollary 4.4. *Let $\mathbb{H}$, C, and $\mathcal{P}_i$ $(i = 1, 2, 3, \dots, N)$ be as in Remark 4.2. For each $j = 1, 2, \dots, k$, let $\{\theta_n^j\}$ be a bounded real sequence, and let $\alpha_n \in [\underline{\alpha}, \overline{\alpha}] \subset (0, 2)$ for all $n \ge 0$. Then the sequence $\{a_n\}$ generated by Algorithm 6 converges strongly to the unique solution to the equation $a_* = P_{\tilde{\Omega}}^{\mathbb{H}}(a_0)$.*

Algorithm 6

Step 1. Choose $a_{-k}, a_{-k+1}, a_{-k+2}, \ldots, a_0 \in \mathbb{H}$ arbitrarily, $W_{-1} = \mathbb{H}$ and set $n = 0$.
Step 2. Given $a_{n-k}, a_{n-k+1}, \ldots, a_n$, compute b_n as in (3.1).
Step 3. Compute c_n and $\tilde{\lambda}_n$ as in (4.6) and (4.7).
Step 4. Define a subset W_n as follows:

$$W_n = \{z \in W_{n-1} : \|c_n - z\|_{\mathbb{H}} \le \|b_n - z\|_{\mathbb{H}}\}.$$

Step 5. Compute $a_{n+1} = P^{\mathbb{H}}_{W_n}(a_0)$.
Step 6. Set $n \leftarrow n + 1$, and go to Step 2.

5. Numerical experiments

Our algorithms are implemented in MATLAB 2014a running on the DESKTOP-8LDGIN0, Intel(R) Core(TM) i5-4210U CPU @ 1.70GHz with 2.40 GHz and 4GB RAM.

The first example is presented to illustrate the performance of Algorithms 1 and 2 in solving the MSFP.

Example 5.1.
We consider the MSFP under the following hypotheses:

(O1) $\mathbb{H} = \mathbb{R}^m$, $\mathbb{H}_i = \mathbb{R}^{m_i}$ and $\mathcal{H} = \mathbb{R}^n$ are (finite-dimensional) Euclidean spaces and $i = 1, 2, 3$. The sets C and Q_i are given by

$$C = \{x \in \mathbb{H} : \langle a, x\rangle \le b\},$$
$$Q_i = \{x \in \mathbb{H}_i : \|x - a_i\| \le r_i\},$$

where b and the radii r_i are randomly generated in the intervals $[0, 10]$ and $[m_i, m_i + 1]$, respectively; the coordinates of a and the centers a_i are randomly generated in the intervals $[1, 5]$ and $[-1, 1]$, respectively.

(O2) $\mathcal{P}_i : \mathbb{R}^m \to \mathbb{R}^{m_i}$ and $\mathcal{L}_i : \mathbb{R}^{m_i} \to \mathbb{R}^n$ are bounded linear operators, where the elements of the representing matrices of which are randomly generated in the closed interval $[-5, 5]$ for each $i = 1, 2, 3$.

For this experiment, we take $y = 0$. It is easy to see that $\Omega \ne \emptyset$ because $0 \in \Omega$. We use Algorithms 1 and 2 with

$$k = 3, \quad m = 5, \quad m_1 = 25, \quad m_2 = 45, \quad m_3 = 65, \quad n = 85,$$

the coordinates of the initial points a_{-3}, a_{-2}, a_{-1} and a_0 are randomly generated in the closed interval $[-5, 5]$ and select the control parameter $\alpha_n = 1.925$ for all $n \ge 0$.

We also compare our Algorithms 1 and 2 with the algorithms in [1]. The control parameters for each algorithm are chosen as follows:

- Algorithm R1 (Algorithm 1 in [1]): $x_0 = a_0$.
- Algorithm R2 (Algorithm 2 in [1]): $x_0 = a_0$ and $h(x) = 0.9x$ for all $x \in \mathbb{R}^m$.

We use the stopping condition

$$\mathrm{err} = \frac{\|(I^{\mathbb{H}} - P^{\mathbb{H}}_C)(a_n)\|^2_{\mathbb{H}} + \sum_{i=1}^3 \|(I^{\mathbb{H}_i} - P^{\mathbb{H}_i}_{Q_i})(\mathcal{P}_i a_n)\|^2_{\mathbb{H}_i} + \|\sum_{i=1}^3 \mathcal{L}_i(\mathcal{P}_i a_n) - y\|^2_{\mathcal{H}}}{5}$$

which is required to be smaller than the given tolerance TOL.

Our numerical results are presented in Tables 1 and 2.

Remark 5.1. From the numerical results reported in Tables 1 and 2 for Example 5.1, it is observed that Algorithms 1 and 2 converge significantly faster than Algorithms R1 and R2 in terms of both iteration numbers and CPU time for all tolerance levels. Moreover, Algorithm 2 exhibits the best overall performance, achieving high accuracy with remarkably fewer iterations and lower computational cost, while Algorithms R1 and R2 require substantially more iterations despite reaching comparable accuracy.

Next, we present an example to illustrate the applicability of the proposed methods to the SFP-MOS, which is a special case of the MSFP. This example focuses on evaluating the performance of Algorithms 3 and 4, and the proposed algorithms are also compared with several existing methods.

Example 5.2. We consider the SFP-MOS with the following hypotheses:

(O1) $i \in \{1, 2, 3\}$ and $\mathbb{H} = \mathbb{H}_i = L^2[0, 1]$ with the inner product

$$\langle x, y\rangle = \int_0^1 x(t)y(t)dt$$

and the induced norm $\|x\|^2 = \langle x, x\rangle$ for all $x = x(t), y = y(t) \in L^2[0, 1]$.

Table 1
Table of numerical results and comparisons among algorithms.

TOL		Parameters	Algorithm 1	Algorithm 2	Parameters	Algorithm R1	Algorithm R2
10^{-4}	n		52	20		379	175
	err		$4.1240e-05$	$2.6258e-06$		$9.8623e-05$	$7.2981e-05$
	CPU-time (s)		0.1582	0.0033		0.0209	0.0091
10^{-5}	n	$\theta_n^1 = 3$	65	20	$\rho_n = 0.5$	480	236
	err	$\theta_n^2 = 2$	$8.4311e-06$	$2.6258e-06$	$\alpha_n = \frac{1}{n}$	$9.5543e-06$	$7.7397e-06$
	CPU-time (s)	$\theta_n^3 = 1$	0.2121	0.0033		0.0243	0.0121
10^{-6}	n		86	29		690	312
	err		$3.8213e-07$	$8.5703e-07$		$8.4449e-07$	$9.6769e-07$
	CPU-time (s)		0.2585	0.0043		0.0336	0.0191
10^{-4}	n		29	21		6782	122
	err		$9.6596e-05$	$8.3817e-05$		$9.9917e-05$	$9.8438e-05$
	CPU-time (s)		0.1037	0.0033		0.3102	0.0066
10^{-5}	n	$\theta_n^1 = -3$	61	29	$\rho_n = 1.25$	9413	208
	err	$\theta_n^2 = -2$	$3.4579e-06$	$6.9764e-06$	$\alpha_n = \frac{1}{n^{0.4}}$	$9.9979e-06$	$9.7605e-06$
	CPU-time (s)	$\theta_n^3 = -1$	0.1800	0.0038		0.4348	0.0108
10^{-6}	n		69	37		12,192	309
	err		$6.2323e-08$	$9.8712e-07$		$9.9982e-07$	$9.8882e-07$
	CPU-time (s)		0.2342	0.0043		0.5634	0.0158

Table 2
Table of numerical results and comparisons among algorithms.

TOL		Parameters	Algorithm 1	Algorithm 2	Parameters	Algorithm R1	Algorithm R2
10^{-4}	n		27	10		7229	649
	err		$4.1987e-05$	$4.7670e-05$		$9.9963e-05$	$9.9965e-05$
	CPU-time (s)		0.0910	0.0031		0.3370	0.0315
10^{-5}	n	$\theta_n^1 = \frac{1}{n}$	501	11	$\rho_n = 1.5$	9888	1172
	err	$\theta_n^2 = \frac{1}{n^2}$	$8.3640e-06$	$2.7808e-06$	$\alpha_n = \frac{1}{n^{0.6}}$	$9.9942e-06$	$9.9985e-06$
	CPU-time (s)	$\theta_n^3 = \frac{1}{n^3}$	1.5778	0.0033		0.4644	0.0554
10^{-6}	n		503	13		12,688	1828
	err		$6.0812e-07$	$7.5039e-07$		$9.9974e-07$	$9.9742e-07$
	CPU-time (s)		1.6060	0.0035		0.5829	0.0933
10^{-4}	n		25	15		9331	4408
	err		$9.2409e-05$	$1.0683e-05$		$9.9935e-05$	$9.9906e-05$
	CPU-time (s)		0.0879	0.0019		0.4392	0.2081
10^{-5}	n	$\theta_n^1 = \frac{1}{n}$	39	20	$\rho_n = 1.925$	12,098	6428
	err	$\theta_n^2 = \frac{2n}{n+1}$	$6.0109e-06$	$1.2654e-06$	$\alpha_n = \frac{1}{n^{0.8}}$	$9.9962e-06$	$9.9938e-06$
	CPU-time (s)	$\theta_n^3 = -\frac{4n}{n+3}$	0.1239	0.0034		0.5535	0.3036
10^{-6}	n		40	25		14,978	8634
	err		$4.1135e-07$	$5.0170e-07$		$9.9985e-07$	$9.9902e-07$
	CPU-time (s)		0.1146	0.0045		0.7551	0.4134

The sets C and Q_i are given by

$$C = \{x \in L^2[0,1] : \langle \mathfrak{a}, x \rangle \leq \zeta\},$$
$$Q_i = \{x \in L^2[0,1] : \langle \mathfrak{a}_i, x \rangle \leq \zeta_i\},$$

where

$$\mathfrak{a} = \mathfrak{a}(t) = t, \quad \mathfrak{a}_i = \mathfrak{a}_i(t) = \sqrt{t^2 + i}, \quad \forall t \in [0,1],$$
$$\zeta = \frac{1}{3}, \quad \zeta_i = \frac{(i+1)\sqrt{i+1} - i\sqrt{i}}{3(i+1)}.$$

(O2) For each $i \in \{1,2,3\}$, the bounded linear operator $\mathcal{P}_i : L^2[0,1] \to L^2[0,1]$ is defined by $\mathcal{P}_i(x)(t) = \frac{x(t)}{i+1}$ for all $t \in [0,1]$.

For this experiment, it is not difficult to see that $\widehat{\Omega} \neq \emptyset$ because $x(t) = t \in \widehat{\Omega}$. We use Algorithms 3 and 4 with $k = 3$ and select the initial points and the control parameters as follows:

$$a_{-3} = a_{-3}(t) = \sqrt{t}, \quad a_{-2} = a_{-2}(t) = e^t,$$
$$a_{-1} = a_{-1}(t) = \sin t, \quad a_0 = a_0(t) = \cos t, \quad \forall t \in [0,1],$$
$$\theta_n^1 = 0.0003, \quad \theta_n^2 = 0.002, \quad \theta_n^3 = -0.01, \quad \alpha_n = 1.75, \quad \forall n \in \mathbb{N}.$$

We use the stopping condition

$$\text{err} = \frac{\|(I^{\mathbb{H}} - P_C^{\mathbb{H}})(a_n)\|_{\mathbb{H}}^2 + \sum_{i=1}^3 \|(I^{\mathbb{H}_i} - P_{Q_i}^{\mathbb{H}_i})(\mathcal{P}_i a_n)\|_{\mathbb{H}_i}^2}{4}$$

which is required to be smaller than the given tolerance TOL.

We also compare our Algorithms 3 and 4 with some previous algorithms (Algorithms 5.1 and 5.2 in [36], Algorithms (1.5) and (1.6) in [3]). The control parameters for each algorithm are chosen as follows:

• Algorithm A (Algorithm 5.1 in [36]):

$$\tau = \frac{0.01}{1 + \sum_{i=1}^3 \|\mathcal{P}_i\|^2}.$$

• Algorithm B (Algorithm 5.2 in [36]):

$$\tau = \frac{0.01}{1 + \sum_{i=1}^3 \|\mathcal{P}_i\|^2}, \quad \alpha_n = \frac{1}{n}.$$

• Algorithm C (Algorithm (1.5) in [3]):

$$\gamma_n = \frac{0.005}{3 \times \max_{1\le i\le 3}\{\|\mathcal{P}_i\|^2\}}.$$

• Algorithm D (Algorithm (1.6) in [3]):

$$\gamma_n = \frac{0.005}{3 \times \max_{1\le i\le 3}\{\|\mathcal{P}_i\|^2\}}, \quad \alpha_n = \frac{1}{n}.$$

For Algorithm B, we choose the fixed element $u = \ln(t+1)$ for all $t \in [0,1]$ while we take $f(x) = 0.75x$ for all $x \in L^2[0,1]$ in Algorithm D.

We also use the stopping rule

$$\text{err} = \frac{\|(I^{\mathbb{H}} - P_C^{\mathbb{H}})(x_n)\|_{\mathbb{H}}^2 + \sum_{i=1}^3 \|(I^{\mathbb{H}_i} - P_{Q_i}^{\mathbb{H}_i})(\mathcal{P}_i x_n)\|_{\mathbb{H}_i}^2}{4} < \text{TOL}$$

for Algorithm A, Algorithm B, Algorithm C and Algorithm D.

The numerical results are presented in Table 3.

Using either one-step or two-step inertial extrapolation, or in the case where the inertial parameters diminish to zero, we obtain the numerical results presented in Table 4.

Remark 5.2. From Table 3, it can be seen that Algorithms 3 and 4 converge faster and require less CPU time than the comparison algorithms (A–D) for all tolerance levels, while delivering comparable or better accuracy. Besides, Algorithm 4 consistently attains the desired accuracy with fewer iterations and lower computational time.

Table 4 further clarifies the role of inertial components in the proposed schemes. As the number of inertial components is gradually reduced, the acceleration effect becomes less pronounced, suggesting that employing more inertial steps generally improves convergence efficiency. Nevertheless, even under reduced inertial factors, our proposed algorithms remain stable and continue to perform competitively.

We now illustrate an application of our algorithms to the signal recovery problem with multiple observations.

Example 5.3. We consider the signal recovery problem with $N = 30$ data observations, which can be formulated as the following constrained optimization model:

$$\min\left\{\frac{1}{2}\|\mathbb{A}_i x - b_i\|^2 : x \in \mathbb{R}^k, \ \|x\|_1 \le p\right\}, \qquad i = 1,2,\dots,30,$$

Table 3
Table of numerical results and comparisons among algorithms.

TOL	Algorithms	n	err	CPU - time (s)
10^{-3}	Algorithm 3	7	$2.592088252769408e-04$	0.0134
	Algorithm 4	5	$6.301511419269003e-04$	0.0156
	Algorithm A	6193	$9.993555064239153e-04$	0.2134
	Algorithm B	2068	$9.998835311812956e-04$	0.0766
	Algorithm C	7561	$9.994936108471009e-04$	0.1708
	Algorithm D	4078	$9.997368699476096e-04$	0.0963
10^{-4}	Algorithm 3	21	$1.571470584330984e-05$	0.0160
	Algorithm 4	7	$2.082606537679928e-05$	0.0158
	Algorithm A	8460	$9.993534383884936e-05$	0.3020
	Algorithm B	4007	$9.999097573477408e-05$	0.1523
	Algorithm C	10498	$9.999920285254293e-05$	0.2399
	Algorithm D	14751	$9.998683764195466e-05$	0.3459
10^{-5}	Algorithm 3	22	$6.957675846735487e-06$	0.0161
	Algorithm 4	8	$5.391978201870193e-06$	0.0159
	Algorithm A	10837	$9.993549652061280e-06$	0.3841
	Algorithm B	6069	$9.994998608800226e-06$	0.2229
	Algorithm C	13577	$9.998215446628838e-06$	0.3116
	Algorithm D	47031	$9.999709923957786e-06$	1.0664

Table 4
Table of numerical results.

	TOL		Algorithm 3	Algorithm 4
$\theta_n^1 = 0.0003, \theta_n^2 = 0.002, \theta_n^3 = 0$	10^{-3}	n	10	5
		err	$4.6245e-04$	$4.6366e-04$
		CPU - time (s)	0.0137	0.0149
	10^{-4}	n	11	7
		err	$3.7428e-05$	$1.9065e-05$
		CPU - time (s)	0.0142	0.0156
	10^{-5}	n	25	9
		err	$8.4936e-06$	$1.3624e-06$
		CPU - time (s)	0.0164	0.0165
$\theta_n^1 = 0.0003, \theta_n^2 = 0, \theta_n^3 = 0$	10^{-3}	n	62	5
		err	$8.7842e-04$	$4.6680e-04$
		CPU - time (s)	0.0219	0.0149
	10^{-4}	n	112	7
		err	$2.5967e-05$	$1.9965e-05$
		CPU - time (s)	0.0311	0.0153
	10^{-5}	n	113	9
		err	$7.2034e-06$	$1.3280e-06$
		CPU - time (s)	0.0312	0.0167
$\theta_n^1 = \frac{1}{n+1}, \theta_n^2 = \frac{n}{n^2+1}, \theta_n^3 = \frac{n^2}{n^3+1}$	10^{-3}	n	14	8
		err	$9.4882e-04$	$1.0450e-04$
		CPU - time (s)	0.0145	0.0155
	10^{-4}	n	32	9
		err	$9.7116e-05$	$9.8050e-05$
		CPU - time (s)	0.0175	0.0169
	10^{-5}	n	43	10
		err	$2.4594e-06$	$3.0315e-06$
		CPU - time (s)	0.0190	0.0171

where each perception matrix $\mathbb{A}_i \in \mathbb{R}^{m\times k}$ and its associated measurement vector $b_i \in \mathbb{R}^m$ are given. Here $\|\cdot\|$ denotes the Euclidean norm and $\|\cdot\|_1$ is the ℓ_1-norm.

For every $i \in \{1, 2, \dots, 30\}$, the matrix $\mathbb{A}_i$ is generated from a standard normal distribution. The true sparse signal $a_{\text{true}} \in \mathbb{R}^k$ is constructed by sampling from the uniform distribution on $[-1, 1]$ with p randomly selected nonzero components.

In the noiseless case, the observations satisfy

$$b_i^* = \mathbb{A}_i a_{\text{true}}, \qquad i = 1, 2, \dots, 30.$$

Table 5
Table of numerical results and comparisons among algorithms.

TOL	Algorithms	n	SNR	MSE	CPU - time (s)
10^{-4}	Algorithm 3	7	31.9352	$5.3891e-05$	0.4455
	Algorithm 4	6	30.5783	$7.3655e-05$	0.3922
	Algorithm R3	29	29.2635	$9.9696e-05$	1.6169
	Algorithm R4	13	29.4147	$9.6286e-05$	0.7337
10^{-5}	Algorithm 3	59	39.3482	$9.7771e-06$	3.6094
	Algorithm 4	10	40.2468	$7.9497e-06$	0.6712
	Algorithm R3	94	39.3326	$9.8123e-06$	5.1890
	Algorithm R4	103	39.2989	$9.8888e-06$	5.6549
10^{-6}	Algorithm 3	233	49.2762	$9.9406e-07$	14.3054
	Algorithm 4	27	60.9328	$6.7882e-08$	2.1348
	Algorithm R3	291	49.2579	$9.9826e-07$	16.2949
	Algorithm R4	951	49.2587	$9.9806e-07$	52.3420

In connection with the SFP-MOS, this signal recovery problem can be reformulated as a feasibility problem of the form

$$\text{Find } x^{\dagger} \in C \text{ such that } \mathcal{P}_i(x^{\dagger}) \in Q_i, \quad i = 1, 2, \dots, 30,$$

where the operators and constraint sets are defined by

$$\mathcal{P}_i(x) := \mathbb{A}_i x, \qquad Q_i := \{b_i^*\} \subset \mathbb{R}^m,$$

and the structural sparsity requirement is represented by the convex set

$$C := \{ x \in \mathbb{R}^k : \|x\|_1 \le p \}.$$

Besides, in order to evaluate $P_C^{\mathbb{R}^k}$, we shall use Duchi et al. method in [37].

In this experiment, we use Algorithms 3 and 4 with the following parameters:

$$\theta_n^1 = \frac{1}{n+1}, \quad \theta_n^2 = \frac{1}{(n+1)^3}, \quad \theta_n^3 = \frac{1}{(n+1)^5}, \quad \alpha_n = 1.9999 - \frac{1}{n+2}, \quad \forall n \ge 0,$$

$$k = 1024, \quad m = 2048, \quad p = 256.$$

The initial points a_{-3}, a_{-2}, and a_{-1} are randomly generated in $\mathbb{R}^k$, while $a_0 = 0$. We further compare Algorithms 3 and 4 with the algorithms reported in [3]. The experimental settings for each algorithm are specified as follows:

• Algorithm R3 (Algorithm 3 in [1]):

$$x_0 = a_0, \quad \rho_n = \alpha_n.$$

• Algorithm R4 (Algorithm 4 in [1]):

$$x_0 = a_0, \quad \rho_n = \alpha_n,$$

and $h(x) = 0.75x$ for all $x \in \mathbb{R}^k$.

To evaluate the recovery accuracy, we use the mean squared error (MSE), defined as

$$\text{MSE} = \frac{1}{k}\|a_n - a_{\text{true}}\|^2,$$

which is required to be smaller than a given tolerance, denoted by TOL. We also evaluate the signal quality using the signal-to-noise ratio (SNR), which compares the power of the desired signal to that of the background error. It is computed as

$$\text{SNR} = 20 \log \frac{\|a_{\text{true}}\|}{\|a_{\text{true}} - a_n\|}.$$

The numerical results are reported in Table 5, while Fig. 1 illustrates both the original and the recovered signals obtained by the proposed algorithms. The behavior of the SNR is depicted in Fig. 2.

Remark 5.3. *Table 5 reports the signal recovery results in terms of SNR, MSE, iteration numbers, and CPU time. Overall, Algorithms 3 and 4 yield better reconstruction quality (higher SNR and lower MSE) with fewer iterations and lower computational time than R3 and R4 across all tolerance levels. Furthermore, Algorithm 4 offers the best balance between accuracy and efficiency, particularly as the tolerance decreases, where it achieves substantially improved SNR/MSE while maintaining low computational time.*

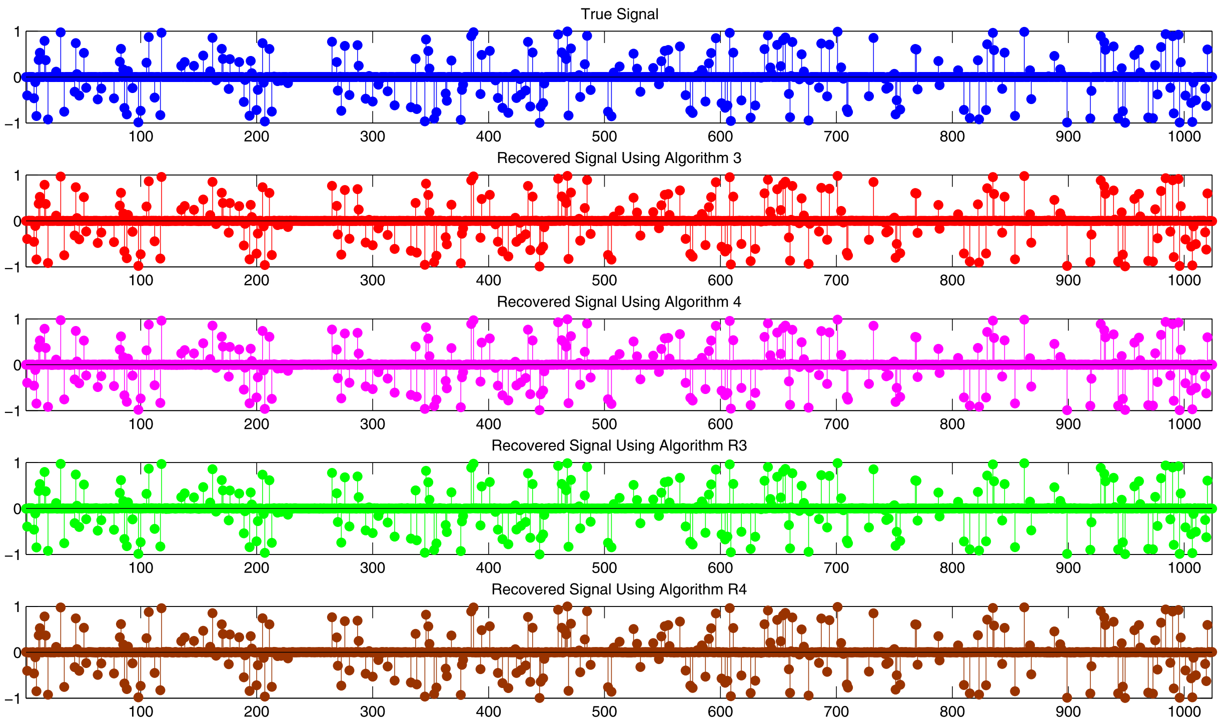

Fig. 1. Original signal and recovered signals with TOL = 10^{-5}.

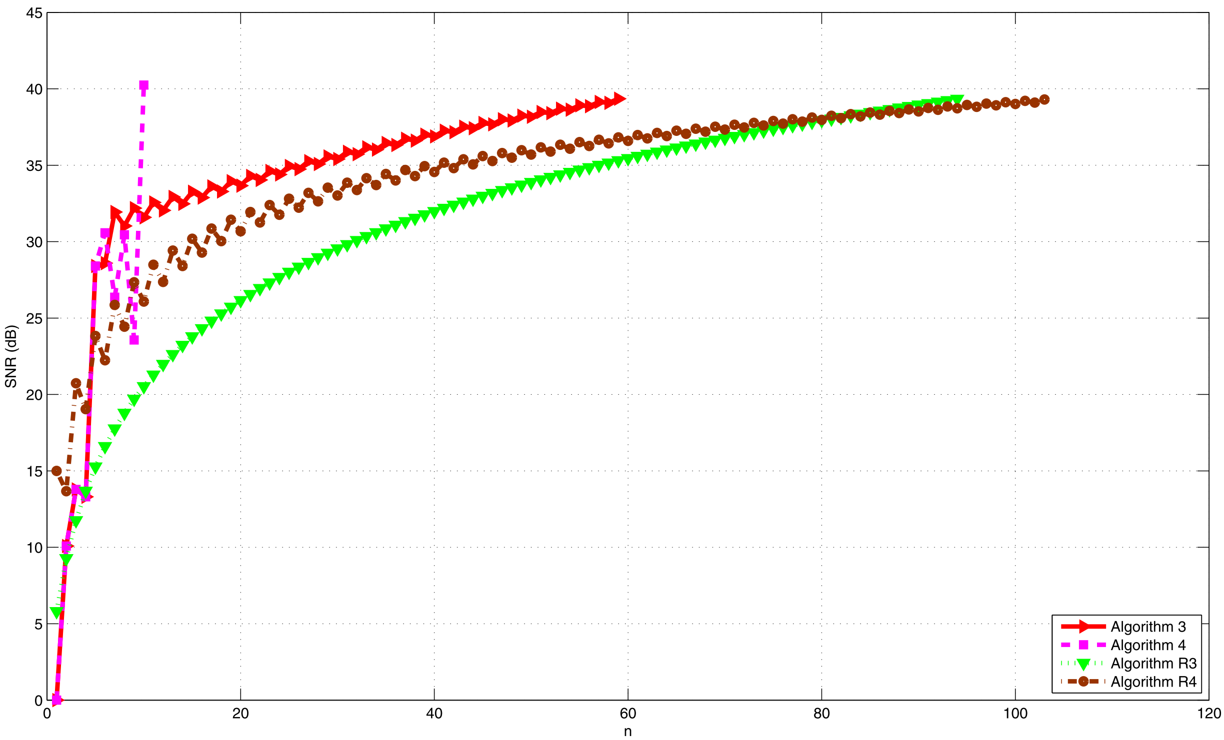

Fig. 2. The behavior of SNR with TOL = 10^{-5}.

6. Conclusions

In this paper, we study the mixed split feasibility problem in real Hilbert spaces, which simultaneously generalizes the SFP-MOS (and, in particular, the classical SFP) and the SEP. To solve the MSFP, we developed two inertial multistep projection-type algorithms constructed via the hybrid projection method and the shrinking projection method. Our convergence analysis shows that the sequences generated by the proposed algorithms converge strongly under suitable assumptions on the control parameters.

A distinctive feature of our algorithms is that the inertial factors are only required to be bounded, whereas most existing inertial schemes impose stronger conditions, such as requiring diminishing sequences or restricting the inertial parameters to prescribed intervals like $[-1, 1]$ or $[0, a]$ for some positive number a. This relaxation provides greater flexibility in choosing the inertial parameters, simplifies implementation, and enhances the practical applicability of the algorithms while still ensuring strong convergence. Several corollaries were also derived, covering important special cases, including the SFP-MOS and the SEP.

Furthermore, we provided three numerical examples demonstrating the efficiency and applicability of the proposed algorithms. The MSFP framework opens new directions for further research, including the development of faster inertial or adaptive methods and additional applications in split feasibility, signal processing, image reconstruction, and related areas.

Ethical Approval

Not applicable

Competing interests

The authors declare that they have no conflict of interest.

Authors' contributions

All authors wrote the main manuscript text and reviewed the manuscript.

Availability of data and materials

Not applicable

Funding

This research was funded by the TNU-University of Sciences for research group NCM.DHKH.2025.03.

Data availability

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

Declaration of competing interest

The authors declare that they have no conflict of interest.

Acknowledgements

The authors are very grateful to the seven anonymous referees for their close reading of the manuscript, their pertinent comments, and their helpful suggestions.

References

[1] S. Reich, T.M. Tuyen, N.S. Ha, N.V. Ninh, A new mixed split feasibility problem model in Hilbert spaces, Optimization 75 (2) (2026) 487–513.
[2] S. Reich, T.M. Tuyen, Iterative methods for solving the generalized split common null point problem in Hilbert spaces, Optimization 69 (5) (2019) 1013–1038.
[3] S. Reich, T.M. Tuyen, M.T.N. Ha, The split feasibility problem with multiple output sets in Hilbert spaces, Optim. Lett. 14 (2020) 2335–2353.
[4] Y. Censor, T. Elfving, A multiprojection algorithm using Bregman projections in a product space, Numer. Algorithms 8 (1994) 221–239.
[5] A. Moudafi, E. Al-Shemas, Simultaneous iterative methods for split equality problems, Trans. Math. Program. Appl. 1 (2) (2013) 1–11.
[6] A. Moudafi, Alternating CQ-algorithms for convex feasibility and split fixed point problems, J. Nonlinear Convex Anal. 15 (2014) 809–818.
[7] Y. Censor, T. Bortfeld, B. Martin, A. Trofimov, A unified approach for inversion problems in intensity-modulated radiation therapy, Phys. Med. Biol. 51 (2006) 2353–2365.
[8] Y. Censor, T. Elfving, N. Kopf, T. Bortfeld, The multiple-sets split feasibility problem and its application, Inverse Probl. 21 (2005) 2071–2084.

[9] C. Byrne, A unified treatment of some iterative algorithms in signal processing and image reconstruction, Inverse Probl. 20 (2004) 103–120.
[10] H. Attouch, J. Bolte, P. Redont, A. Soubeyran, A new class of alternating proximal minimization algorithms with costs-to-move, SIAM J. Optim. 18 (2007) 1061–1081.
[11] H. Attouch, J. Bolte, P. Redont, A. Soubeyran, Alternating proximal algorithms for weakly coupled minimization problems. applications to dynamical games and PDEs, J. Convex Anal. 15 (2008) 485–506.
[12] B.T. Polyak, Some methods of speeding up the convergence of iteration methods, USSR Comput. Math. Math. Phys. 4 (5) (1964) 1–17.
[13] B.T. Polyak, Introduction to Optimization, New York, Optimization Software Inc., 1987.
[14] O.S. Iyiola, Y. Shehu, Convergence results of two-step inertial proximal point algorithm, Appl. Numer. Math. 182 (2022) 57–75.
[15] L.O. Jolaoso, Y. Shehu, J.C. Yao, Strongly convergent inertial proximal point algorithm without on-line rule, J. Optim. Theory Appl. 200 (2) (2024) 555–584.
[16] F. Alvarez, Weak convergence of a relaxed and inertial hybrid projection-proximal point algorithm for maximal monotone operators in Hilbert space, SIAM J. Optim. 14 (3) (2004) 773–782.
[17] H. Attouch, A. Cabot, Convergence rates of inertial forward-backward algorithms, SIAM J. Optim. 28 (2018) 849–874.
[18] H. Attouch, A. Cabot, Convergence of damped inertial dynamics governed by regularized maximally monotone operators, J. Differ. Eq. 264 (2018) 7138–7182.
[19] H. Attouch, A. Cabot, Convergence of a relaxed inertial proximal algorithm for maximally monotone operators, Math. Program. 184 (2020) 243–287.
[20] H. Attouch, S.C. László, Continuous Newton-like inertial dynamics for monotone inclusions, Set Valued Var. Anal. 29 (2021) 555–581.
[21] H. Attouch, J. Peypouquet, Convergence of inertial dynamics and proximal algorithms governed by maximal monotone operators, Math. Program. Ser. B 174 (2019) 391–432.
[22] Q.L. Dong, Y.J. Cho, L.L. Zhong, T.M. Rassias, Inertial projection and contraction algorithms for variational inequalities, J. Global Optim. 70 (2018) 687–704.
[23] N.S. Ha, T.M. Tuyen, P.T.V. Huyen, Inertial proximal point algorithm for the split common solution problem of monotone operator equations, Comp. Appl. Math. 42 (303) (2023).
[24] Y. Hao, J. Zhao, Two-step inertial Bregman projection iterative algorithm for solving the split feasibility problem, Appl. Nonlinear Anal. 1 (1) (2024) 64–78.
[25] D.R. Sahu, Y.J. Cho, Q.-L. Dong, M. Kashyap, X. Li, Inertial relaxed CQ algorithms for solving a split feasibility problem in Hilbert spaces, Numer. Algorithms 87 (2021) 1075–1095.
[26] Y. Shehu, L. Liu, X. Mu, Q.L. Dong, Analysis of versions of relaxed inertial projection and contraction method, Appl. Numer. Math. 165 (2021) 1–21.
[27] D.V. Thong, N.T. Vinh, Y.J. Cho, New strong convergence theorem of the inertial projection and contraction method for variational inequality problems, Numer. Algorithms 84 (2020) 285–305.
[28] F. Wang, H. Yu, An adaptive and expanded framework for multiple-operator split common fixed point problems, J. Optim. Theory Appl. 207 (2025), Paper No. 8, 25 pp.
[29] A. Ahmad, P. Kumam, Y.J. Cho, K. Sitthithakerngkiet, Halpern-type relaxed algorithms with alternated and multi-step inertia for split feasibility problems with applications in classification problems, Constr. Math. Anal. 8 (2) (2025) 50–80.
[30] A. Ahmad, P. Kumam, T. Seangwattana, Strong convergence multi-step inertial golden ratio-based algorithms for split feasibility problems with applications, Math. Model. Numer. Simul. Appl. 5 (2) (2025) 307–347.
[31] A. Ahmad, P. Kumam, M.M. Yahaya, K. Sitthithakerngkiet, Alternated and multi-step inertial algorithm with three-term conjugate gradient-like direction for split feasibilities with applications in classification problems and elastic net, Comp. Appl. Math. 44 (2025), Paper No. 325, 32 pp.
[32] P.L. Combettes, L.E. Glaudin, Quasinonexpansive iterations on the affine hull of orbits: from Mann's mean value algorithm to inertial methods, SIAM J. Optim. 27 (2017) 2356–2380.
[33] Q.L. Dong, J.Z. Huang, X.H. Li, Y.J. Cho, T.M. Rassias, MiKM: multi-step inertial Krasnosel'skii-Mann algorithm and its applications, J. Global Optim. 73 (2019) 801–824.
[34] H.H. Bauschke, P.L. Combettes, Convex Analysis and Monotone Operator Theory in Hilbert Spaces, New York, Springer, 2011.
[35] E. Kopecká, S. Reich, A note on alternating projections in Hilbert space, J. Fixed Point Theory Appl. 12 (2012) 41–47.
[36] H. Cui, F. Wang, The split common fixed point problem with multiple output sets for demicontractive mappings, Optimization 73 (6) (2024) 1933–1947.
[37] J. Duchi, S. Shalev-Shwartz, Y. Singer, T. Chandra, Efficient projections onto the ℓ_1-ball for learning in high dimensions, in: Proceedings of the 25th International Conference on Machine Learning, Helsinki, 2008.