





Article

Half-Space Relaxation Projection Method for Solving Multiple-Set Split Feasibility Problem

Guash Haile Taddele ^{1,2}, Poom Kumam ^{1,3,4,*}, Anteneh Getachew Gebrie ² and Kanokwan Sitthithakerngkiet ⁵

¹ Department of Mathematics, King Mongkut's University of Technology Thonburi, Bangkok 10140, Thailand; guashhaile79@gmail.com

² Department of Mathematics, College of Computational and Natural Science, Debre Berhan University, Debre Berhan P.O. Box 445, Ethiopia; antgetm@gmail.com

³ Center of Excellence in Theoretical and Computational Science (TaCS-CoE), Science Laboratory Building, King Mongkut's University of Technology Thonburi (KMUTT), 126 Pracha-Uthit Road, Bang Mod, Thung Khru, Bangkok 10140, Thailand

⁴ Department of Medical Research, China Medical University Hospital, China Medical University, Taichung 40402, Taiwan

⁵ Intelligent and Nonlinear Dynamic Innovations Research Center, Department of Mathematics, Faculty of Applied Science, King Mongkut's University of Technology North Bangkok (KMUTNB), Wongsawang, Bangsue, Bangkok 10800, Thailand; kanokwan.s@sci.kmutnb.ac.th

* Correspondence: poom.kumam@mail.kmutt.ac.th

Received: 3 June 2020; Accepted: 21 July 2020; Published: 24 July 2020



Abstract: In this paper, we study an iterative method for solving the multiple-set split feasibility problem: find a point in the intersection of a finite family of closed convex sets in one space such that its image under a linear transformation belongs to the intersection of another finite family of closed convex sets in the image space. In our result, we obtain a strongly convergent algorithm by relaxing the closed convex sets to half-spaces, using the projection onto those half-spaces and by introducing the extended form of selecting step sizes used in a relaxed CQ algorithm for solving the split feasibility problem. We also give several numerical examples for illustrating the efficiency and implementation of our algorithm in comparison with existing algorithms in the literature.

Keywords: multiple-set split feasibility problem; relaxed CQ algorithm; subdifferential; strong convergence; Hilbert space

MSC: 47H09; 47J25; 65K10; 49J52

1. Introduction

1.1. Split Inverse Problem

Split Inverse Problem (SIP) is an archetypal model presented in ([1], Section 2), and it is stated as

$$\begin{cases} \text{find } x^* \in X \text{ that solves IP1} \\ \text{such that} \\ y^* = Ax^* \in Y \text{ and solves IP2} \end{cases}$$

where A is a bounded linear operator from a space X to another space Y , and IP1 and IP2 are two inverse problems installed in X and Y , respectively. Real-world inverse problems can be cast into this framework by making different choices of the spaces X and Y (including the case $X = Y$), and by choosing appropriate inverse problems for IP1 and IP2. For example, image restoration, computer tomograph

and intensity-modulated radiation therapy (IMRT) treatment planning generate functions that can be transformed to have an SIP model; see in [2–5]. The split feasibility problem [6] and multiple-set split feasibility problem [7] are the first instances of the SIP, where the two problems IP1 and IP2 are of the Convex Feasibility Problem (CFP) type [8]. In the SIP framework, many authors studied cases for which IP1 and IP2 are convex feasibility problems, minimization problems, equilibrium problems, fixed point problems, null point problems and so on; see, for example [2,3,5,9–20].

1.2. Split Feasibility Problem and Multiple-Set Split Feasibility Problem

Let H be a real Hilbert space and let $T : H \rightarrow H$ be an operator. We say that T is ρ -strongly quasi-nonexpansive, where $\rho \geq 0$, if $\text{Fix}T = \{x \in H : Tx = x\} \neq \emptyset$ and

$$\|Tx - p\|^2 \leq \|x - p\|^2 - \rho\|Tx - p\|^2, \forall (x, p) \in H \times \text{Fix}T. \tag{1}$$

If $\rho = 0$ in (1), then T is called a quasi-nonexpansive operator. If $\rho > 0$ in (1), then we say that T is strongly quasi-nonexpansive. Obviously, a nonexpansive operator having a fixed point is quasi-nonexpansive. If T is quasi-nonexpansive, then $\text{Fix}T$ is closed and convex. For $\nu \geq 0$ denote by $T_\nu := (1 - \nu)I + \nu T$, the ν -relaxation of T , where I is the identity operator and ν is called relaxation parameter. If T is quasi-nonexpansive, then usually one applies a relaxation parameter $\nu \in [0, 1]$.

Let H_1 and H_2 be real Hilbert spaces and let $A : H_1 \rightarrow H_2$ be a bounded linear operator. Given a nonempty closed convex subsets $\{C_1, \dots, C_N\}$ and $\{Q_1, \dots, Q_M\}$ of H_1 and H_2 , respectively. The Multiple-Set Split Feasibility Problem (MSSFP), which was introduced by Censor et al. [7], is formulated as finding a point

$$\bar{x} \in \bigcap_{i=1}^N C_i \text{ such that } A\bar{x} \in \bigcap_{j=1}^M Q_j. \tag{2}$$

Denote by Ω the set of solutions for (2). The MSSFP (2) with $N = M = 1$ is known as the Split Feasibility Problem (SFP), which is formulated as finding a point

$$\bar{x} \in C \text{ such that } A\bar{x} \in Q, \tag{3}$$

where C and Q are nonempty closed convex subsets of real Hilbert spaces H_1 and H_2 , respectively. The SFP was first introduced in 1994 by Censor and Elfving [6] for modeling inverse problems in finite-dimensional Hilbert spaces for modeling inverse problems that arise from phase retrievals and in medical image reconstruction. SFP plays an important role in the study of signal processing, image reconstruction, intensity-modulated radiation, therapy, etc. [2,3,5,11]. Several iterative algorithms were presented to solve the SFP and MSSFP provided the solution exists; see, for example, in [3,6,21–32]. The algorithm proposed by Censor and Elfving [6] for solving the SFP involves the computation of the inverse of A per each iteration assuming the existence of the inverse of A , a fact that makes the algorithm nonapplicable in practice. Most methods employ the Landweber operators, see the definition and its property in [33]. In general, all of these methods produce sequences that converge weakly to a solution. Byrne [3] proposed the following iteration for solving the SFP, and called it the CQ-algorithm or the projected Landweber method:

$$x_{n+1} = P_C V_\nu(x_n), \tag{4}$$

where $x_1 \in H_1$ is arbitrary, V_ν is a ν -relaxation of the Landweber operator V (corresponding to P_Q), i.e.,

$$V = I - \frac{1}{\|A\|^2} A^*(I - P_Q)A,$$

where $\nu \in (0, 2)$, A^* denotes the adjoint of A , and $\|A\|^2$ is the spectral norm of AA^* . It is well known that the CQ algorithm (4) does not necessarily converge strongly to the solution of SFP in the infinite-dimensional Hilbert space. An important advantage of the CQ algorithm by Byrne [3,21] is that computation of the inverse of A (matrix inverses) is not necessary. The Landweber operator is used for a more general type of problem called the split common fixed point problem of quasi-nonexpansive operators (see, for example, [10,34,35]), but the implementation of the Landweber-type algorithm generated requires prior knowledge of the operator norm. However, the operator norm is a global invariant and is often difficult to estimate; see, for example, the Theorem of Hendrickx and Olshevsky in [36]. To overcome this difficulty, Lopez et al. [2] introduced a new way of selecting the step sizes for solving the SFP (3) such that the information of the operator norm is not necessary. To be precise, Lopez et al. [2] proposed

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

where $\gamma_n = \frac{\rho_n f(x_n)}{\|\nabla f(x_n)\|^2}$, $\forall n \geq 1, \rho_n \in (0, 4), f(x_n) = \frac{1}{2}\|(I - P_Q)Ax_n\|^2$ and $\nabla f(x_n) = A^*(I - P_Q)Ax_n$. In addition, the computation of the projection on a closed convex set is not easy. In order to overcome this drawback, Yang [37] considered SFPs in which the involved sets C and Q are given as sub-level sets of convex functions, i.e.,

$$C = \{x \in H_1 : c(x) \leq 0\} \text{ and } Q = \{y \in H_2 : q(y) \leq 0\}, \tag{5}$$

where $c : H_1 \rightarrow \mathbb{R}$ and $q : H_2 \rightarrow \mathbb{R}$ are convex and subdifferentiable functions on H_1 and H_2 , respectively, and that ∂c and ∂q are bounded operators (i.e., bounded on bounded sets). It is known that every convex function defined on a finite-dimensional Hilbert space is subdifferentiable and its subdifferential operator is a bounded operator (see [38]). In this situation, the efficiency of the CQ method is extremely affected because, in general, the computation of projections onto such subsets is still very difficult. Motivated by Fukushima’s relaxed projection method in [39], Yang [37] suggested calculating the projection onto a half-space containing the original subset instead of the latter set itself. More precisely, Yang introduced a relaxed CQ algorithm using a half-space relaxation projection method for solving SFP. The proposed algorithm by Yang is given as follows:

$$x_{n+1} = P_{C_n}(x_n - \gamma f_n(x_n)),$$

where $f_n = A^*(I - P_{Q_n})A$, for each $n \in \mathbb{N}$ the set C_n is given by

$$C_n = \{x \in H_1 : c(x_n) \leq \langle \xi_n, x_n - x \rangle\}, \tag{6}$$

where $\xi_n \in \partial c(x_n)$, and the set Q_n is given by

$$Q_n = \{y \in H_2 : q(Ax_n) \leq \langle \varepsilon_n, Ax_n - y \rangle\}, \tag{7}$$

where $\varepsilon_n \in \partial q(Ax_n)$. Obviously, C_n and Q_n are half-spaces and $C \subset C_n$ and $Q \subset Q_n$ for every $n \geq 1$. More important, since the projections onto C_n and Q_n have the closed form, the relaxed CQ algorithm is now easily implemented. The specific form of the metric projections onto C_n and Q_n can be found in [38,40,41].

For solving the MSSFP (2), many methods have been developed; see, for example, in [7,26,42–49] and references therein. We aim to propose a strongly convergent algorithm with high efficiency that is easy to implement in solving the MSSFP. Motivated by Yang [37], we are interested in solving the MSSFP (2) in which the involved sets C_i ($i \in \{1, \dots, N\}$) and Q_j ($j \in \{1, \dots, M\}$) are given as sub-level sets of convex functions, i.e.,

$$C_i = \{x \in H_1 : c_i(x) \leq 0\} \text{ and } Q_j = \{y \in H_2 : q_j(y) \leq 0\}, \tag{8}$$

where $c_i : H_1 \rightarrow \mathbb{R}$ and $q_j : H_2 \rightarrow \mathbb{R}$ are convex functions for all $i \in \{1, \dots, N\}$, $j \in \{1, \dots, M\}$. We assume that each c_i and q_j are subdifferentiable on H_1 and H_2 , respectively, and that ∂c_i and ∂q_j are bounded operators (i.e., bounded on bounded sets). In what follows, we define $N + M$ half-spaces at point x_n by

$$C_{i,n} = \{x \in H_1 : c_i(x_n) \leq \langle \xi_{i,n}, x_n - x \rangle\}, \tag{9}$$

where $\xi_{i,n} \in \partial c_i(x_n)$, and

$$Q_{j,n} = \{y \in H_2 : q_j(Ax_n) \leq \langle \varepsilon_{j,n}, Ax_n - y \rangle\}, \tag{10}$$

where $\varepsilon_{j,n} \in \partial q_j(Ax_n)$.

The paper contributes to developing the algorithm for the MSSFP in the direction of half-space relaxation (assuming C_i and Q_j are given as a sub-level sets of convex functions (8)) and parallel computation of projection onto half-spaces (9) and (10) without prior knowledge of the operator norm.

This paper is organized in the following way. In Section 2, we recall some basic and useful facts that will be used in the proof of our results. In Section 3, we introduce the extended form of the way of selecting step sizes used in the relaxed CQ algorithm for solving the SFP by [37] and Lopez et al. [2] to work for the MSSFP framework, and we analyze the strong convergence of our proposed algorithm. In Section 4, we give some numerical examples to discuss the performance of the proposed algorithm. Finally, we give some conclusions.

2. Preliminary

In this section, in order to prove our result, we recall some basic notions and useful results in a real Hilbert space H . The symbols " \rightharpoonup " and " \rightarrow " denote weak and strong convergence, respectively.

Let C be a nonempty closed convex subset of H . The metric projection on C is a mapping $P_C : H \rightarrow C$ defined by

$$P_C(x) = \arg \min\{\|y - x\| : y \in C\}, \quad x \in H.$$

Lemma 1. [50] *Let C be a closed convex subset of H . Given $x \in H$ and a point $z \in C$, then $z = P_C(x)$ if and only if*

$$\langle x - z, y - z \rangle \leq 0, \quad \forall y \in C.$$

The mapping $T : H \rightarrow H$ is firmly nonexpansive if

$$\|Tx - Ty\|^2 \leq \|x - y\|^2 - \|(I - T)x - (I - T)y\|^2, \quad \forall x, y \in H,$$

which is equivalent to

$$\|Tx - Ty\|^2 \leq \langle Tx - Ty, x - y \rangle, \quad \forall x, y \in H.$$

If T is firmly nonexpansive, $I - T$ is also firmly nonexpansive. The metric projection P_C on a closed convex subset C of H is firmly nonexpansive.

Definition 1. *The subdifferential of a convex function $f : H \rightarrow \mathbb{R}$ at $x \in H$, denoted by $\partial f(x)$, is defined by*

$$\partial f(x) = \{\xi \in H : f(z) \geq f(x) + \langle \xi, z - x \rangle, \quad \forall z \in H\}.$$

If $\partial f(x) \neq \emptyset$, f is said to be subdifferentiable at x . If the function f is continuously differentiable then $\partial f(x) = \{\nabla f(x)\}$, this is the gradient of f .

Definition 2. *The function $f : H \rightarrow \mathbb{R}$ is called weakly lower semi-continuous at x_0 if the sequence $\{x_n\}$ weakly converges to x_0 implies*

$$\liminf_{n \rightarrow \infty} f(x_n) \geq f(x_0).$$

A function that is weakly lower semi-continuous at each point of H is called weakly lower semi-continuous on H .

Lemma 2. [3,51] Let H_1 and H_2 be real Hilbert spaces and $f : H_1 \rightarrow \mathbb{R}$ is given by $f(x) = \frac{1}{2}\|(I - P_Q)Ax\|^2$ where Q is closed convex subset of H_2 and $A : H_1 \rightarrow H_2$ be a bounded linear operator. Then

- (i) The function f is convex and weakly lower semi-continuous on H_1 ;
- (ii) $\nabla f(x) = A^*(I - P_Q)Ax$ for $x \in H_1$;
- (iii) ∇f is $\|A\|^2$ -Lipschitz, i.e., $\|\nabla f(x) - \nabla f(y)\| \leq \|A\|^2\|x - y\|, \forall x, y \in H_1$.

Lemma 3. [27,52] Let C and Q be closed convex subsets of real Hilbert spaces H_1 and H_2 , respectively, and $f : H_1 \rightarrow \mathbb{R}$ is given by $f(x) = \frac{1}{2}\|(I - P_Q)Ax\|^2$, where $A : H_1 \rightarrow H_2$ be a bounded linear operator. Then for $\lambda > 0$ and $\bar{x} \in H_1$ the following statements are equivalent.

- (i) The point \bar{x} solves the SFP (3), i.e., $\bar{x} \in \{x \in C : Ax \in Q\}$;
- (ii) The point \bar{x} is the fixed point of the mapping $P_C(I - \lambda \nabla f)$, i.e.,

$$P_C(\bar{x} - \lambda \nabla f(\bar{x})) = \bar{x}.$$

Lemma 4. [53] Let H be a real Hilbert space. Then, for all $x, y \in H$ and $\alpha \in [0, 1]$, we have

- (i) $\|\alpha x + (1 - \alpha)y\|^2 = \alpha\|x\|^2 + (1 - \alpha)\|y\|^2 - \alpha(1 - \alpha)\|x - y\|^2$;
- (ii) $\|x + y\|^2 = \|x\|^2 + \|y\|^2 + 2\langle x, y \rangle$;
- (iii) $\|x + y\|^2 \leq \|x\|^2 + 2\langle y, x + y \rangle$.

Lemma 5. [54] Let $\{a_n\}$ be the sequence of nonnegative numbers such that

$$a_{n+1} \leq (1 - \alpha_n)a_n + \alpha_n\delta_n,$$

where $\{\delta_n\}$ is a sequence of real numbers bounded from above and $0 \leq \alpha_n \leq 1$ and $\sum_{n=1}^{\infty} \alpha_n = \infty$. Then it holds that

$$\limsup_{n \rightarrow \infty} a_n \leq \limsup_{n \rightarrow \infty} \delta_n.$$

3. Half-Space Relaxation Projection Algorithm

In this section, we propose an iterative algorithm to solve the MSSFP (2). To make our algorithm more efficient and the implementation of the algorithm more easy, we assume that the convex sets C_i and Q_j are given in the form of (8) and we use projections onto half-spaces $C_{i,n}$ and $Q_{j,n}$ defined in (9) and (10), respectively, instead of onto C_i and Q_j , just as the relaxed or inexact methods in [5,37,39,55]. Moreover, in order to remove the requirement of the estimated value of the operator norm and solve the MSSFP when finding the operator norm is not easy, we now introduce a new way of selecting the step sizes for solving the MSSFP (2) given as follows for $x \in H_1$, and $C_{i,n}$ and $Q_{j,n}$ are half-spaces defined in (9) and (10).

- (i) For each $i \in \{1, \dots, N\}$ and $n \geq 1$, define

$$g_{i,n}(x) = \frac{1}{2}\|(I - P_{C_{i,n}})x\|^2 \text{ and so } \nabla g_{i,n}(x) = (I - P_{C_{i,n}})x.$$

- (ii) $g_n(x)$ and $\nabla g_n(x)$ are defined as $g_n(x) = g_{i_{n_x},n}(x)$ and so $\nabla g_n(x) = \nabla g_{i_{n_x},n}(x)$ where $i_{n_x} \in \{1, \dots, N\}$ such that for each $n \geq 1$,

$$i_{n_x} \in \arg \max\{g_{i,n}(x) : i \in \{1, \dots, N\}\}.$$

(iii) For each $j \in \{1, \dots, M\}$ and $n \geq 1$, define

$$f_{j,n}(x) = \frac{1}{2} \|(I - P_{Q_{j,n}})Ax\|^2 \text{ and } \nabla f_{j,n}(x) = A^*(I - P_{Q_{j,n}})Ax.$$

From Aubin [51], $g_{i,n}$ and $f_{j,n}$ are convex, weakly lower semi-continuous and differentiable for each $i \in \{1, \dots, N\}$ and $j \in \{1, \dots, M\}$. Now, using $\nabla g_{i,n}$, $g_{i,n}$, g_n , ∇g_n , $f_{j,n}$ and $\nabla f_{j,n}$ given in (i)–(iii) above, and assuming that the solution set Ω of the MSSFP (2) is nonempty, we propose and analyze the strong convergence of our algorithm, called the *Half-Space Relaxation Projection Algorithm*.

Note that, the iterative scheme in Algorithm 1 (HSRPA) is established in away that $\nabla g_n(z_n)$ and $\nabla f_{j,n}(z_n)$ are computed, i.e., the projections $P_{C_{i,n}}$ and $P_{Q_{j,n}}$ are computed, in parallel setting under simple assumptions on step sizes.

Algorithm 1: Half-Space Relaxation Projection Algorithm (HSRPA).

Initialization: Choose $u, x_1 \in H_1$. Let the positive real constants λ_1, λ_2 and δ_j ($j = 1, \dots, M$), and the real sequences $\{\alpha_n\}$, $\{\beta_n\}$ and $\{\rho_n\}$ satisfy the following conditions:

- (C1) $\lambda_1, \lambda_2 \in (0, 1)$ and $\lambda_1 + \lambda_2 = 1$.
- (C2) $0 < \delta_j < 1$ for all $j \in \{1, \dots, M\}$, and $\sum_{j=1}^M \delta_j = 1$.
- (C3) $0 < \alpha_n < 1$, $\lim_{n \rightarrow \infty} \alpha_n = 0$ and $\sum_{n=1}^{\infty} \alpha_n = \infty$.
- (C4) $0 < a \leq \beta_n \leq b < 1$ for all $n \in \mathbb{N}$.
- (C5) $0 < \lambda \rho_n < 4\bar{\lambda}$ and $\liminf_{n \rightarrow \infty} \rho_n(4\bar{\lambda} - \lambda \rho_n) > 0$, where $\lambda = \max\{\lambda_1, \lambda_2\}$ and $\bar{\lambda} = \min\{\lambda_1, \lambda_2\}$.

Iterative Step: Proceed with the following computations:

$$\begin{cases} z_n = (1 - \alpha_n)x_n + \alpha_n u, \\ y_n = z_n - \sum_{j=1}^M \delta_j \tau_{j,n} (\lambda_1 \nabla g_n(z_n) + \lambda_2 \nabla f_{j,n}(z_n)), \\ x_{n+1} = (1 - \beta_n)z_n + \beta_n y_n \end{cases}$$

where

$$\tau_{j,n} = \rho_n \frac{f_{j,n}(z_n) + g_n(z_n)}{d_j(z_n)},$$

for

$$d_j(z_n) = \begin{cases} 1, & \text{if } \|\nabla g_n(z_n)\|^2 + \|\nabla f_{j,n}(z_n)\|^2 = 0 \\ \|\nabla g_n(z_n)\|^2 + \|\nabla f_{j,n}(z_n)\|^2, & \text{otherwise.} \end{cases}$$

Lemma 6. If $\{j \in \{1, \dots, M\} : \|\nabla g_n(z_n)\|^2 + \|\nabla f_{j,n}(z_n)\|^2 \neq 0\} = \emptyset$ at some iterate n in HSRPA, then z_n is the solution of MSSFP (2).

Proof. $\{j \in \{1, \dots, M\} : \|\nabla g_n(z_n)\|^2 + \|\nabla f_{j,n}(z_n)\|^2 \neq 0\} = \emptyset$ implies

$$\begin{aligned} & \|\nabla g_n(z_n)\|^2 + \|\nabla f_{j,n}(z_n)\|^2 = 0, \quad \forall j \in \{1, \dots, M\}, \\ & \Leftrightarrow \|\nabla g_n(z_n)\| = 0 = \|\nabla f_{j,n}(z_n)\|, \quad \forall j \in \{1, \dots, M\}, \\ & \Leftrightarrow \|\nabla g_{i,n}(z_n)\| = 0 = \|\nabla f_{j,n}(z_n)\|, \quad \forall i \in \{1, \dots, N\}, \forall j \in \{1, \dots, M\}, \\ & \Leftrightarrow (I - P_{C_{i,n}})(z_n) = 0 = A^*(I - P_{Q_{j,n}})A(z_n), \quad \forall i \in \{1, \dots, N\}, \forall j \in \{1, \dots, M\}. \end{aligned}$$

Thus, one can get $P_{C_{i,n}}(z_n - \lambda \nabla f_{j,n}(z_n)) = z_n$ for all $i \in \{1, \dots, N\}$ and $j \in \{1, \dots, M\}$. Since $C_i \subset C_{i,n}$, we get $z_n \in C_{i,n}$ for all $i \in \{1, \dots, N\}$. Combined with the fixed point relation Lemma 3 (ii),

we also get that $Az_n \in Q_{j,n}$ for all $j \in \{1, \dots, M\}$. Following the representations of the sets $C_{i,n}$ and $Q_{j,n}$ in (9) and (10) we obtain that $c_i(z_n) \leq 0$ for all $i \in \{1, \dots, N\}$ and $q_j(Az_n) \leq 0$ for all $i \in \{1, \dots, N\}$, and this implies that $z_n \in C_i$ for all $i \in \{1, \dots, N\}$ and $Az_n \in Q_j$ for all $j \in \{1, \dots, M\}$, which completes the proof. \square

By Lemma 6, we can conclude that the HSRPA terminates at some iterate n when $\{j \in \{1, \dots, M\} : \|\nabla g_n(z_n)\|^2 + \|\nabla f_{j,n}(z_n)\|^2 \neq 0\} = \emptyset$. Otherwise, if the HSRPA does not stop, then we have the following strong convergence theorem for the approximation of the solution of the problem of MSSFP (2).

Theorem 1. *The sequence $\{x_n\}$ generated by HSRPA converges strongly to the solution point \bar{x} of MSSFP (2) ($x_n \rightarrow \bar{x} \in \Omega$) where $\bar{x} = P_{\Omega}u$.*

Proof. Let $\bar{x} \in \Omega$. Since $I - P_{C_{i,n}}$ and $I - P_{Q_{j,n}}$ are firmly nonexpansive, and since \bar{x} verifies (2), we have for all $x \in H_1$

$$\begin{aligned} \langle \nabla g_{i,n}(x), x - \bar{x} \rangle &= \langle (I - P_{C_{i,n}})x, x - \bar{x} \rangle \\ &\geq \|(I - P_{C_{i,n}})x\|^2 = 2g_{i,n}(x), \end{aligned} \tag{11}$$

and

$$\begin{aligned} \langle \nabla f_{j,n}(x), x - \bar{x} \rangle &= \langle A^*(I - P_{Q_{j,n}})Ax, x - \bar{x} \rangle \\ &= \langle (I - P_{Q_{j,n}})Ax, Ax - A\bar{x} \rangle \\ &\geq \|(I - P_{Q_{j,n}})Ax\|^2 = 2f_{j,n}(x). \end{aligned} \tag{12}$$

Using definition of y_n and Lemma 4 (ii), we have

$$\begin{aligned} \|y_n - \bar{x}\|^2 &= \left\| z_n - \sum_{j=1}^M \delta_j \tau_{j,n} (\lambda_1 \nabla g_n(z_n) + \lambda_2 \nabla f_{j,n}(z_n)) - \bar{x} \right\|^2 \\ &\leq \|z_n - \bar{x}\|^2 + \left\| \sum_{j=1}^M \delta_j \tau_{j,n} (\lambda_1 \nabla g_n(z_n) + \lambda_2 \nabla f_{j,n}(z_n)) \right\|^2 \\ &\quad - 2 \left\langle \sum_{j=1}^M \delta_j \tau_{j,n} (\lambda_1 \nabla g_n(z_n) + \lambda_2 \nabla f_{j,n}(z_n)), z_n - \bar{x} \right\rangle. \end{aligned} \tag{13}$$

Using convexity of $\|\cdot\|^2$, we have

$$\begin{aligned} &\left\| \sum_{j=1}^M \delta_j \tau_{j,n} (\lambda_1 \nabla g_n(z_n) + \lambda_2 \nabla f_{j,n}(z_n)) \right\|^2 \\ &\leq \sum_{j=1}^M \delta_j (\tau_{j,n})^2 \left\| \lambda_1 \nabla g_n(z_n) + \lambda_2 \nabla f_{j,n}(z_n) \right\|^2 \\ &\leq \sum_{j=1}^M \delta_j (\tau_{j,n})^2 (\lambda_1 \|\nabla g_n(z_n)\|^2 + \lambda_2 \|\nabla f_{j,n}(z_n)\|^2) \\ &\leq \sum_{j=1}^M \left\{ \lambda \delta_j \left(\rho_n \frac{f_{j,n}(z_n) + g_n(z_n)}{d_j(z_n)} \right)^2 (\|\nabla g_n(z_n)\|^2 + \|\nabla f_{j,n}(z_n)\|^2) \right\} \\ &\leq \sum_{j=1}^M \left\{ \lambda \delta_j \left(\rho_n \frac{f_{j,n}(z_n) + g_n(z_n)}{d_j(z_n)} \right)^2 d_j(z_n) \right\} \\ &= \lambda \rho_n^2 \sum_{j=1}^M \left\{ \delta_j \frac{(f_{j,n}(z_n) + g_n(z_n))^2}{d_j(z_n)} \right\}. \end{aligned} \tag{14}$$

From (11) and (12), we have

$$\begin{aligned}
 & \left\langle \sum_{j=1}^M \delta_j \tau_{j,n} (\lambda_1 \nabla g_n(z_n) + \lambda_2 \nabla f_{j,n}(z_n)), z_n - \hat{x} \right\rangle \\
 &= \sum_{j=1}^M \delta_j \tau_{j,n} \left\langle (\lambda_1 \nabla g_n(z_n) + \lambda_2 \nabla f_{j,n}(z_n)), z_n - \hat{x} \right\rangle \\
 &= \sum_{j=1}^M \delta_j \tau_{j,n} (\lambda_1 \langle \nabla g_n(z_n), z_n - \hat{x} \rangle + \lambda_2 \langle \nabla f_{j,n}(z_n), z_n - \hat{x} \rangle) \\
 &\geq \sum_{j=1}^M \delta_j \tau_{j,n} (2\lambda_1 g_n(z_n) + 2\lambda_2 f_{j,n}(z_n)) \\
 &= \sum_{j=1}^M \left\{ \delta_j \rho_n \frac{f_{j,n}(z_n) + g_n(z_n)}{d_j(z_n)} (2\lambda_1 g_n(z_n) + 2\lambda_2 f_{j,n}(z_n)) \right\} \\
 &\geq 2\bar{\lambda} \rho_n \sum_{j=1}^M \left\{ \delta_j \frac{(f_{j,n}(z_n) + g_n(z_n))^2}{d_j(z_n)} \right\}.
 \end{aligned} \tag{15}$$

In view of (13), (14) and (15), we have

$$\begin{aligned}
 \|y_n - \bar{x}\|^2 &\leq \|z_n - \bar{x}\|^2 + \lambda \rho_n^2 \sum_{j=1}^M \left\{ \delta_j \frac{(f_{j,n}(z_n) + g_n(z_n))^2}{d_j(z_n)} \right\} \\
 &\quad - 4\bar{\lambda} \rho_n \sum_{j=1}^M \left\{ \delta_j \frac{(f_{j,n}(z_n) + g_n(z_n))^2}{d_j(z_n)} \right\} \\
 &= \|z_n - \bar{x}\|^2 + \rho_n (\lambda \rho_n - 4\bar{\lambda}) \sum_{j=1}^M \left\{ \delta_j \frac{(f_{j,n}(z_n) + g_n(z_n))^2}{d_j(z_n)} \right\}.
 \end{aligned} \tag{16}$$

From (16) and (C5), we have

$$\|y_n - \bar{x}\| \leq \|z_n - \bar{x}\|. \tag{17}$$

Using (17), Lemma 4 (i) and the definition of x_{n+1} , we get

$$\begin{aligned}
 \|x_{n+1} - \bar{x}\|^2 &= \|(1 - \beta_n)z_n + \beta_n y_n - \bar{x}\|^2 \\
 &= \|(1 - \beta_n)(z_n - \bar{x}) + \beta_n(y_n - \bar{x})\|^2 \\
 &= (1 - \beta_n)\|z_n - \bar{x}\|^2 + \beta_n\|y_n - \bar{x}\|^2 - \beta_n(1 - \beta_n)\|z_n - y_n\|^2 \\
 &\leq \|z_n - \bar{x}\|^2 - \beta_n(1 - \beta_n)\|z_n - y_n\|^2.
 \end{aligned} \tag{18}$$

From (18) and the definition of z_n , we get

$$\begin{aligned}
 \|x_{n+1} - \bar{x}\| &\leq \|z_n - \bar{x}\| = (1 - \alpha_n)\|x_n - \bar{x}\| + \alpha_n\|u - \bar{x}\| \\
 &\leq \max\{\|x_n - \bar{x}\|, \|u - \bar{x}\|\} \\
 &\quad \vdots \\
 &\leq \max\{\|x_1 - \bar{x}\|, \|u - \bar{x}\|\},
 \end{aligned}$$

which shows that $\{x_n\}$ is bounded. Consequently, $\{z_n\}$, $\{Az_n\}$ and $\{y_n\}$ are all bounded.

Now,

$$\frac{1}{\beta_n} (x_{n+1} - z_n) = \frac{1}{\beta_n} ((1 - \beta_n)z_n + \beta_n y_n - z_n) = y_n - z_n, \tag{19}$$

and

$$\|y_n - z_n\|^2 = \frac{1}{\beta_n^2} \|x_{n+1} - z_n\|^2 = \frac{\alpha_n}{\beta_n} \left(\frac{\|x_{n+1} - z_n\|^2}{\alpha_n \beta_n} \right). \tag{20}$$

Using (18) and (19), we have

$$\|x_{n+1} - \bar{x}\|^2 \leq \|z_n - \bar{x}\|^2 - \frac{1 - \beta_n}{\beta_n} \|x_{n+1} - z_n\|^2. \tag{21}$$

From the definition of z_n , we have

$$\begin{aligned} \|z_n - \bar{x}\|^2 &= \|(1 - \alpha_n)x_n + \alpha_n u - \bar{x}\|^2 \\ &= (1 - \alpha_n)^2 \|x_n - \bar{x}\|^2 + \alpha_n^2 \|u - \bar{x}\|^2 + 2\alpha_n(1 - \alpha_n)\langle x_n - \bar{x}, u - \bar{x} \rangle \\ &\leq (1 - \alpha_n)\|x_n - \bar{x}\|^2 + \alpha_n^2 \|u - \bar{x}\|^2 + 2\alpha_n(1 - \alpha_n)\langle x_n - \bar{x}, u - \bar{x} \rangle. \end{aligned} \tag{22}$$

Thus, (21) and (22) gives

$$\begin{aligned} \|x_{n+1} - \bar{x}\|^2 &\leq (1 - \alpha_n)\|x_n - \bar{x}\|^2 + \alpha_n^2 \|u - \bar{x}\|^2 \\ &\quad + 2\alpha_n(1 - \alpha_n)\langle x_n - \bar{x}, u - \bar{x} \rangle - \frac{1 - \beta_n}{\beta_n} \|x_{n+1} - z_n\|^2. \end{aligned} \tag{23}$$

That is,

$$\|x_{n+1} - \bar{x}\|^2 \leq (1 - \alpha_n)\|x_n - \bar{x}\|^2 - \alpha_n \Gamma_n, \tag{24}$$

where

$$\Gamma_n = -\alpha_n \|u - \bar{x}\|^2 + 2(1 - \alpha_n)\langle \bar{x} - x_n, u - \bar{x} \rangle + \frac{1 - \beta_n}{\alpha_n \beta_n} \|x_{n+1} - z_n\|^2.$$

We know that $\{x_n\}$ is bounded and so it is bounded below. Hence, Γ_n is bounded below. Furthermore, using Lemma 5 and (C3), we have

$$\limsup_{n \rightarrow \infty} \|x_n - \bar{x}\| \leq \limsup_{n \rightarrow \infty} (-\Gamma_n) = -\liminf_{n \rightarrow \infty} \Gamma_n. \tag{25}$$

Therefore, $\liminf_{n \rightarrow \infty} \Gamma_n$ is a finite real number and by (C3), we have

$$\liminf_{n \rightarrow \infty} \Gamma_n = \liminf_{n \rightarrow \infty} (2\langle \bar{x} - x_n, u - \bar{x} \rangle + \frac{1 - \beta_n}{\alpha_n \beta_n} \|x_{n+1} - z_n\|^2).$$

Since $\{x_n\}$ is bounded, there exists a subsequence $\{x_{n_k}\}$ of $\{x_n\}$ such that $x_{n_k} \rightarrow p$ for some $p \in H_1$ and

$$\liminf_{n \rightarrow \infty} \Gamma_n = \liminf_{k \rightarrow \infty} (2\langle \bar{x} - x_{n_k}, u - \bar{x} \rangle + \frac{1 - \beta_{n_k}}{\alpha_{n_k} \beta_{n_k}} \|x_{n_k+1} - z_{n_k}\|^2). \tag{26}$$

Since $\{x_n\}$ is bounded and $\liminf \Gamma_n$ is finite, we have that $\frac{1 - \beta_{n_k}}{\alpha_{n_k} \beta_{n_k}} \|x_{n_k+1} - z_{n_k}\|^2$ is bounded. Also, by (C4), we have $\frac{1 - \beta_n}{\alpha_n \beta_n} \geq \frac{1 - b}{a_n \beta_n} > 0$ and so we have that $\frac{1}{\alpha_{n_k} \beta_{n_k}} \|x_{n_k+1} - z_{n_k}\|^2$ is bounded. Observe from (C3) and (C4), we have

$$0 < \frac{\alpha_{n_k}}{\beta_{n_k}} \leq \frac{\alpha_{n_k}}{a} \rightarrow 0, \quad k \rightarrow \infty.$$

Therefore, we obtain from (20) and $\frac{\alpha_{n_k}}{\beta_{n_k}} \rightarrow 0, k \rightarrow \infty$ that

$$\|y_{n_k} - z_{n_k}\| \rightarrow 0, \quad k \rightarrow \infty. \tag{27}$$

From the definition of x_{n+1} , we have

$$\|x_{n_k+1} - z_{n_k}\| = \beta_{n_k} \|y_{n_k} - z_{n_k}\| \rightarrow 0, \quad k \rightarrow \infty,$$

and

$$\|z_{n_k} - x_{n_k}\| = \alpha_{n_k} \|u - x_{n_k}\| \rightarrow 0, \quad k \rightarrow \infty. \tag{28}$$

Hence,

$$\|x_{n_k+1} - x_{n_k}\| \leq \|x_{n_k+1} - z_{n_k}\| + \|z_{n_k} - x_{n_k}\| \rightarrow 0, \quad k \rightarrow \infty.$$

Now, using (16), we obtain

$$\begin{aligned} \rho_{n_k}(4\bar{\lambda} - \lambda\rho_{n_k}) \sum_{j=1}^M \left\{ \delta_j \frac{(f_{j,n_k}(z_{n_k}) + g_{n_k}(z_{n_k}))^2}{d_j(z_{n_k})} \right\} \\ \leq (\|z_{n_k} - \bar{x}\| - \|y_{n_k} - \bar{x}\|)(\|z_{n_k} - \bar{x}\| + \|y_{n_k} - \bar{x}\|) \\ \leq \|z_{n_k} - y_{n_k}\|(\|z_{n_k} - \bar{x}\| + \|y_{n_k} - \bar{x}\|). \end{aligned} \tag{29}$$

Therefore, (27), (29) and (C5) gives

$$\rho_{n_k}(4\bar{\lambda} - \lambda\rho_{n_k}) \sum_{j=1}^M \left\{ \delta_j \frac{(f_{j,n_k}(z_{n_k}) + g_{n_k}(z_{n_k}))^2}{d_j(z_{n_k})} \right\} \rightarrow 0, \quad k \rightarrow \infty. \tag{30}$$

Again using (C5) together with (30) yields

$$\sum_{j=1}^M \left\{ \delta_j \frac{(f_{j,n_k}(z_{n_k}) + g_{n_k}(z_{n_k}))^2}{d_j(z_{n_k})} \right\} \rightarrow 0, \quad k \rightarrow \infty. \tag{31}$$

Hence, in view of (31) and restriction condition (C2), we have

$$\frac{(f_{j,n_k}(z_{n_k}) + g_{n_k}(z_{n_k}))^2}{d_j(z_{n_k})} \rightarrow 0, \quad k \rightarrow \infty, \tag{32}$$

for all $j \in \{1, \dots, M\}$.

For each $i \in \{1, \dots, N\}$ and for each $j \in \{1, \dots, M\}$, $\nabla f_{j,n}(\cdot)$ and $\nabla g_{i,n}(\cdot)$ are Lipschitz continuous with constant $\|A\|^2$ and 1, respectively. Since the sequence $\{z_n\}$ is bounded and

$$\|\nabla f_{j,n}(z_n)\| = \|\nabla f_{j,n}(z_n) - \nabla f_{j,n}(\bar{x})\| \leq \|A\|^2 \|z_n - \bar{x}\|, \quad \forall j \in \{1, \dots, M\},$$

$$\|\nabla g_{i,n}(z_n)\| = \|\nabla g_{i,n}(z_n) - \nabla g_{i,n}(\bar{x})\| \leq \|z_n - \bar{x}\|, \quad \forall i \in \{1, \dots, N\},$$

we have the sequences $\{\|\nabla g_{i,n}(z_n)\|\}_{n=1}^\infty$ and $\{\|\nabla f_{j,n}(z_n)\|\}_{n=1}^\infty$, which are bounded. Hence, we have $\{d_j(z_n)\}_{n=1}^\infty$ bounded and hence $\{d_j(z_{n_k})\}_{k=1}^\infty$ is bounded. Consequently, from (32), we have

$$\lim_{k \rightarrow \infty} f_{j,n_k}(z_{n_k}) = \lim_{k \rightarrow \infty} g_{n_k}(z_{n_k}) = 0, \quad \forall j \in \{1, \dots, M\}. \tag{33}$$

From the definition of $g_{n_k}(z_{n_k})$, we can have

$$g_{i,n_k}(z_{n_k}) \leq g_{n_k}(z_{n_k}), \quad \forall i \in \{1, \dots, N\}. \tag{34}$$

Therefore, (33) and (34) gives

$$\lim_{k \rightarrow \infty} f_{j,n_k}(z_{n_k}) = \lim_{k \rightarrow \infty} g_{i,n_k}(z_{n_k}) = 0, \quad \forall i \in \{1, \dots, N\}, \forall j \in \{1, \dots, M\}.$$

That is, for all $i \in \{1, \dots, N\}, j \in \{1, \dots, M\}$, we have

$$\lim_{k \rightarrow \infty} \|(I - P_{Q_{j,n_k}})Az_{n_k}\| = \lim_{k \rightarrow \infty} \|(I - P_{C_{i,n_k}})z_{n_k}\| = 0. \tag{35}$$

Therefore, since $\{z_n\}$ is bounded and from the boundedness assumption of the subdifferential operator ∂q_j , the sequence $\{\varepsilon_{j,n}\}_{n=1}^\infty$ is bounded. In view of this and (35), for all $j \in \{1, \dots, M\}$ we have

$$\begin{aligned} q_j(Az_{n_k}) &\leq \langle \varepsilon_{j,n_k}, Az_{n_k} - P_{Q_{j,n_k}}(Az_{n_k}) \rangle \\ &\leq \|\varepsilon_{j,n_k}\| \|(I - P_{Q_{j,n_k}})Az_{n_k}\| \rightarrow 0, \quad k \rightarrow \infty. \end{aligned} \tag{36}$$

Similarly, from the boundedness of $\{\xi_{i,n}\}_{n=1}^\infty$ and (35), for all $i \in \{1, \dots, N\}$ we obtain

$$\begin{aligned} c_i(z_{n_k}) &\leq \langle \xi_{i,n_k}, z_{n_k} - P_{C_{i,n_k}}(z_{n_k}) \rangle \\ &\leq \|\xi_{i,n_k}\| \|(I - P_{C_{i,n_k}})z_{n_k}\| \rightarrow 0, \quad k \rightarrow \infty. \end{aligned} \tag{37}$$

Since $x_{n_k} \rightharpoonup p$ and using (28), we have $z_{n_k} \rightharpoonup p$ and hence $Az_{n_k} \rightharpoonup Ap$. The weak lower semi-continuity of $q_j(\cdot)$ and (36) implies that

$$q_j(Ap) \leq \liminf_{k \rightarrow \infty} q_j(Az_{n_k}) \leq \limsup_{k \rightarrow \infty} q_j(Az_{n_k}) \leq 0, \quad \forall j \in \{1, \dots, M\}.$$

That is, $Ap \in Q_j$ for all $j \in \{1, \dots, M\}$.

Likewise, the weak lower semi-continuity of $c_i(\cdot)$ and (37) implies that

$$c_i(p) \leq \liminf_{k \rightarrow \infty} c_i(z_{n_k}) \leq 0, \quad \forall i \in \{1, \dots, N\}.$$

That is, $p \in C_i$ for all $i \in \{1, \dots, N\}$. Hence, $p \in \Omega$.

Take $\bar{x} = P_\Omega u$. Then, we obtain from (26) and Lemma 1 that

$$\begin{aligned} \liminf_{n \rightarrow \infty} \Gamma_n &= \liminf_{k \rightarrow \infty} \left(2\langle \bar{x} - x_{n_k}, u - \bar{x} \rangle + \frac{1 - \beta_{n_k}}{\alpha_{n_k} \beta_{n_k}} \|x_{n_{k+1}} - z_{n_k}\|^2 \right) \\ &\geq 2 \liminf_{k \rightarrow \infty} \langle \bar{x} - x_{n_k}, u - \bar{x} \rangle \\ &\geq 2\langle \bar{x} - p, u - \bar{x} \rangle \geq 0. \end{aligned}$$

Then we have from (25) that

$$\limsup_{n \rightarrow \infty} \|x_n - \bar{x}\|^2 \leq \limsup_{n \rightarrow \infty} (-\Gamma_n) = -\liminf_{n \rightarrow \infty} \Gamma_n \leq 0.$$

Therefore, $\|x_n - \bar{x}\| \rightarrow 0$ and this implies that $\{x_n\}$ converges strongly to \bar{x} . This completes the proof. \square

Remark 1.

- i. When the point u in HSRPA is taken to be 0, from Theorem 1, we see that the limit point \bar{x} of the sequence $\{x_n\}$ is the unique minimum norm solution of the MSSFP, i.e., $\|\bar{x}\| = \min_{x \in \Omega} \|x\|$.
- ii. In the algorithm (HSRPA), the stepsize $\tau_{j,n}$ can also be replaced by

$$\tau_{j,n}^* = \rho_n \frac{f_{j,n}(z_n) + g_n(z_n)}{d_j^2(z_n)}, \tag{38}$$

where

$$d_j(z_n) = \begin{cases} 1, & \text{if } \max\{\|\nabla g_n(z_n)\|, \|\nabla f_{j,n}(z_n)\|\} = 0 \\ \max\{\|\nabla g_n(z_n)\|, \|\nabla f_{j,n}(z_n)\|\}, & \text{otherwise.} \end{cases}$$

The proof for the strong convergence of the HSRPA using the stepsize $\tau_{j,n}^*$ defined in (38) is almost the same as the proof of Theorem 1. To be precise, only slight rearrangement in (14) is required in the proof of Theorem 1.

If $M = N = 1$, we have the following algorithm as a consequence of HSRPA concerning SFP (3) assuming that C and Q are given as sub-level sets of convex functions (5) and by constructing half-spaces (6) and (7), and defining $g_n(x) = \frac{1}{2}\|(I - P_{C_n})x\|^2$, $\nabla g_n(x) = (I - P_{C_n})x$, $f_n(x) = \frac{1}{2}\|(I - P_{Q_n})Ax\|^2$ and $\nabla f_n(x) = A^*(I - P_{Q_n})Ax$.

Corollary 1. Assume that $\bar{x} \in C \cap A^{-1}(Q) \neq \emptyset$. Then, the sequence $\{x_n\}$ generated by Algorithm 2 converges strongly to the solution $\bar{x} = P_{C \cap A^{-1}(Q)}(u)$ of SFP (3).

Algorithm 2: Algorithm for solving the SFP.

Initialization: Choose $u, x_1 \in H_1$. Let the positive real constants λ_1 and λ_2 , and the real sequences $\{\alpha_n\}$, $\{\beta_n\}$ and $\{\rho_n\}$ satisfy the following conditions:

- (A1) $\lambda_1, \lambda_2 \in (0, 1)$ and $\lambda_1 + \lambda_2 = 1$.
- (A2) $0 < \alpha_n < 1$, $\lim_{n \rightarrow \infty} \alpha_n = 0$ and $\sum_{n=1}^{\infty} \alpha_n = \infty$.
- (A3) $0 < a \leq \beta_n \leq b < 1$ for all $n \in \mathbb{N}$.
- (A4) $0 < \lambda \rho_n < 2\bar{\lambda}$ and $\liminf_{n \rightarrow \infty} \rho_n(4\bar{\lambda} - \lambda \rho_n) > 0$ where $\lambda = \max\{\lambda_1, \lambda_2\}$ and $\bar{\lambda} = \min\{\lambda_1, \lambda_2\}$.

Iterative Step: Proceed with the following computations:

$$\begin{cases} z_n = (1 - \alpha_n)x_n + \alpha_n u, \\ y_n = z_n - \tau_n(\lambda_1 \nabla g_n(z_n) + \lambda_2 \nabla f_n(z_n)), \\ x_{n+1} = (1 - \beta_n)z_n + \beta_n y_n, \end{cases}$$

where

$$\tau_n = \rho_n \frac{f_n(z_n) + g_n(z_n)}{d(z_n)},$$

for

$$d(z_n) = \begin{cases} 1, & \text{if } \|\nabla g_n(z_n)\|^2 + \|\nabla f_n(z_n)\|^2 = 0 \\ \|\nabla g_n(z_n)\|^2 + \|\nabla f_n(z_n)\|^2, & \text{otherwise.} \end{cases}$$

4. Preliminary Numerical Results and Applications

In this section, we illustrate the numerical performance and applicability of HSRPA by solving some problems. In the first example, we investigate the numerical performance of HSRPA with regards to different choices of the control parameters α_n, β_n and ρ_n . In Example 2, we illustrate the numerical properties of HSRPA in comparison with three other strongly convergent algorithms, namely the gradient projection method (GPM) by Censor et al. ([7], Algorithm 1), the perturbed projection method (PPM) by Censor et al. ([43], Algorithm 5), and the self-adaptive projection method (SAPM) by Zhao and Yang ([46], Algorithm 3.2). As mentioned in Remark 1(ii), the stepsize in HSRPA can be replaced by the stepsize (38). Therefore in Example 3, we analyze the effect of the two stepsizes in HSRPA for different choices of λ_1 and λ_2 . Additionally, we compare HSRPA with et al. ([48], Algorithm 2.1). Also, a comparison of Algorithm 2 with the strong convergence result of SFP proposed by Shehu et al. [56] is given in Example 4. Finally in Section 4.1, we present a sparse signal recovery experiment to illustrate the efficiency of Algorithm 2 by comparing with algorithms proposed by Lopez [2] and Yang [37]. The numerical results are completed on a standard TOSHIBA laptop with Intel(R) Core(TM) i5-2450M CPU@2.5GHz 2.5GHz with memory 4GB. The programme is implemented in MATLAB R2020a.

Example 1. Consider MSSFP (2) for $H_1 = \mathbb{R}^s$, $H_2 = \mathbb{R}^t$, $A : \mathbb{R}^s \rightarrow \mathbb{R}^t$ given by $A(x) = G_{t \times s}(x)$, where $G_{t \times s}$ is $t \times s$ matrix, the closed convex subsets C_i ($i \in \{1, \dots, N\}$) of \mathbb{R}^s are given by

$$C_i = \{x = (x_1, \dots, x_s)^T \in \mathbb{R}^s : c_i(x) \leq 0\},$$

where $c_i(x) = \|x - x_i^0\|^2 - r_i^2$ such that

- $r_i = r$ for all $i \in \{2, \dots, N\}$, where r is a positive real number,

- $x_i^0 = (x_{1,i}, \dots, x_{s,i})^T = (0, \dots, 0, i - 1)^T \in \mathbb{R}^s$ for each $i = 1, \dots, N$,

and the closed convex subsets Q_j ($j \in \{1, \dots, M\}$) of \mathbb{R}^t are given by

$$Q_j = \{y = (y_1, y_2, \dots, y_t)^T \in \mathbb{R}^t : q_j(y) \leq 0\},$$

where

$$q_j(y) = \sum_{k=1}^t a_{k,j}(y_k - y_{k,j}) - b_j,$$

such that for each $j \in \{1, \dots, M\}$;

- $a_{k,j} = 2^j, \forall k = 1, \dots, t$,
- $b_j = j - 1$,
- $y_{t,j} = r\theta - \frac{b_j}{a_{t,j}}$, and $y_{k,j} = 0, \forall k = 1, \dots, t - 1$, where θ is a nonzero real number.

Notice that, $\bigcap_{i=1}^N C_i = \emptyset$ for $N > 2r + 1$, $\bigcap_{i=1}^N C_i \neq \emptyset$, $\bigcap_{i=1}^N C_i$ contains infinite points for $0 < N < 2r + 1$, $\bigcap_{i=1}^N C_i = \{(0, \dots, 0, r)^T\}$ for $N = 2r + 1$, and r is a natural number. Moreover, $\bigcap_{j=1}^M Q_j \neq \emptyset$,

and $(0, \dots, 0, r\theta)^T \in \bigcap_{j=1}^M Q_j$.

We consider for $s = t$, $G_{t \times s} = \theta I_{s \times s}$, $N = 2r + 1$, and $M = 4$, where r is natural number, and $I_{s \times s}$ is $s \times s$ identity matrix. Thus, $A((0, \dots, 0, r)^T) = (0, \dots, 0, r\epsilon)^T$, and hence the solution set of the MSSEFP is $\Omega = \{(0, \dots, 0, r)^T\}$.

For each $i \in \{1, \dots, N\}$ and $j \in \{1, \dots, M\}$, the subdifferential is given by

$$\partial c_i(z_n) = \begin{cases} \left\{ \frac{z_n - x_i^0}{\|z_n - x_i^0\|} \right\}, & \text{if } z_n - x_i^0 \neq 0, \\ \{\varepsilon_i \in \mathbb{R}^s : \|\varepsilon_i\| \leq 1\}, & \text{otherwise.} \end{cases}$$

and $\partial q_j(Az_n) = \{(a_{1,j}, \dots, a_{t,j})^T\}$.

Note that the projection

$$P_{C_{i,n}}(z_n) = \arg \min \{ \|x - z_n\| : x \in C_{i,n} \},$$

where $C_{i,n} = \{x \in H_1 : c_i(z_n) \leq \langle \xi_{i,n}, z_n - x \rangle\}$, is solving the following quadratic programming problems with inequality constraint

$$\begin{aligned} & \text{minimize } \frac{1}{2}x^T \bar{H}x + \bar{B}_n^T x + \bar{c} \\ & \text{subject to } \bar{D}_{i,n}x \leq \bar{F}_i, \end{aligned} \tag{39}$$

where $\bar{H} = 2I_{s \times s}$, $\bar{B}_n = -2z_n$, $\bar{c} = \|z_n\|^2$, $\bar{D}_{i,n} = \xi_{i,n} = [\xi_{i,n,1}, \dots, \xi_{i,n,s}]$, $\bar{F}_i = r_i^2 - \|z_n - x_i^0\|^2 + \langle \xi_{i,n}, z_n \rangle$. Moreover, the projection $P_{Q_{j,n}}(Az_n)$ for where $Q_{j,n} = \{y \in H_2 : q_j(Az_n) \leq \langle \varepsilon_{j,n}, Az_n - y \rangle\}$, is solving the following quadratic programming problems with inequality constraint

$$\begin{aligned} & \text{minimize } \frac{1}{2}w^T \hat{H}w + \hat{B}_n^T w + \hat{c} \\ & \text{subject to } \hat{D}_{j,n}(w) \leq \hat{F}_j, \end{aligned} \tag{40}$$

where $\hat{H} = 2I_{t \times t}$, $\hat{B}_n = -2Az_n$, $\hat{c} = \|Az_n\|^2$, $\hat{D}_{j,n} = \varepsilon_{j,n}$, $\hat{F}_j = b_j + \langle \varepsilon_{j,n}, Az_n \rangle + \langle a_j, y_j^0 - Az_n \rangle$ for $a_j = (a_{1,j}, \dots, a_{t,j})$. The problems (39) and (40) can be effectively solved by its appropriate solver in MATLAB. In the following our experiments we took

$$\xi_{i,n} = \begin{cases} \frac{z_n - x_i^0}{\|z_n - x_i^0\|}, & \text{if } z_n - x_i^0 \neq 0, \\ 0, & \text{otherwise,} \end{cases}$$

and $\varepsilon_{j,n} = (a_{1,j}, \dots, a_{t,j})^T$.

We study the numerical behavior of HSRPA for different parameters $\alpha_n, \beta_n, \rho_n$ and for different dimensions $s = t$, where $r = 2$ (i.e., $N = 5$), $\lambda_1 = \lambda_2 = \frac{1}{2}$, $\delta_j = \frac{j}{4}$ for all $j \in \{1, 2, 3, 4\}$ are fixed. Notice that for the choice of $\lambda_1 = \lambda_2 = \frac{1}{2}$ the parameter ρ_n is chosen in such a way that $0 < \rho_n < 4$ and $\liminf_{n \rightarrow \infty} \rho_n(4 - \rho_n) > 0$. The numerical results are shown in Figures 1 and 2.

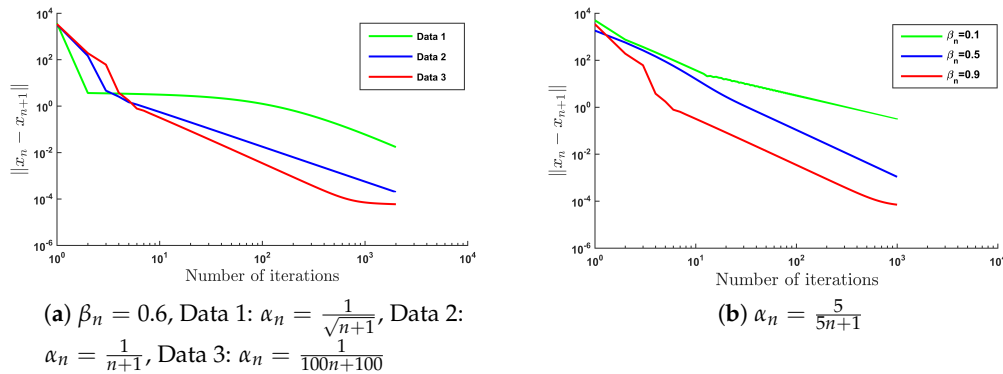


Figure 1. For $t = s = 100, \rho_n = 2, \theta = 10$ and for randomly generated starting points x_1 and u .

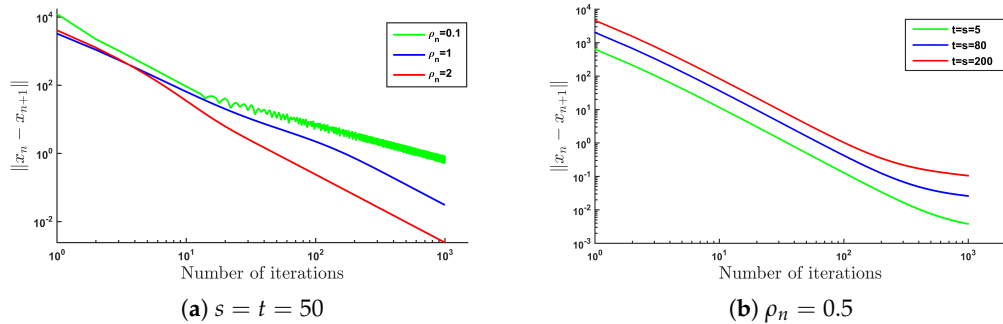


Figure 2. For $\alpha_n = \frac{5}{5n+1}, \beta_n = \frac{1}{2} + \frac{1}{n+4}, \theta = 150$ and for randomly generated starting points x_1 and u .

In view of Figures 1 and 2, we see that our algorithm works better for

- (i) A sequence $\{\alpha_n\}$ in which the terms are very close to zero;
- (ii) A sequence $\{\beta_n\}$ in which terms are very close to 1;
- (iii) A sequence $\{\rho_n\}$ in which terms are larger but not exceeding 4.

Example 2. Comparing HSRPA with GPM, PPM and SAPM.

Consider MSSEF (2) for $H_1 = \mathbb{R}^3, H_2 = \mathbb{R}^3, A : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ given by $A = \theta I_{3 \times 3}$, the closed convex subsets C_i ($i \in \{1, \dots, N\}$) of \mathbb{R}^3 are given by

$$C_i = \{x \in \mathbb{R}^3 : c_i(x) \leq 0\},$$

where θ is a real constant, $I_{3 \times 3}$ is 3×3 identity matrix and

$$c_i(x) = (-1)^i (\langle x, w_i^0 \rangle - \gamma_i)$$

such that $w_i^0 = (i, i + 1, \frac{i+2}{\sqrt{2}})^T$ and $\gamma_i = 5i + 3$, and the closed convex subsets Q_j ($j \in \{1, \dots, M\}$) of \mathbb{R}^3 are given by

$$Q_j = \{y \in \mathbb{R}^3 : q_j(y) \leq 0\},$$

where $q_1(y) = \|y\|^2 - 12\theta^2, q_j(y) = (-1)^j (\langle y, z_j^0 \rangle - 3\theta(j + j^2))$ and $z_j^0 = (j, j^2, \sqrt{2}j^2)^T$ for each $j \in \{2, \dots, M\}$.

For each choice of θ the solution set of the MSSFP is $\{(3, 1, \sqrt{2})\}$, i.e., $\bar{x} = (3, 1, \sqrt{2})$.

We choose GPM, PPM and SAPM because the problem under consideration is the same and the approach has some common features with our approach.

GPM: The proposed iterative algorithm solving MSSFP is reduced to

$$x_{n+1} = x_n + \varrho \left(\sum_{i=1}^N \sigma_i (P_{C_i} - I)x_n + \sum_{j=1}^M v_j A^* (P_{Q_j} - I)Ax_n \right), \tag{41}$$

where $\varrho \in (0, \frac{2}{L})$, $L = \sum_{i=1}^N \sigma_i + \omega \sum_{j=1}^M v_j$ and $\sum_{i=1}^N \sigma_i + \sum_{j=1}^M v_j = 1$ for $\sigma_i > 0, v_j > 0$ and ω is the spectral radius of $A^*A = A^tA$.

PPM: The proposed algorithm is obtained by replacing the projections on the closed convex subset C_i and Q_j in (41) by the half-space $C_{i,n}$ and $C_{j,n}$.

SAPM: The proposed iterative algorithm for solving the MSSFP is reduced to

$$\begin{cases} y_n = x_n - \varrho_n \nabla p_n(x_n), \\ \varrho_n = \gamma \zeta^{ln}, \text{ where } \min\{l : \|\nabla p_n(x_n) - \nabla p_n(y_n)\| \leq \frac{\mu \|x_n - y_n\|}{\gamma \zeta^l}, l = 0, 1, 2, \dots\}, \\ x_{n+1} = x_n - \varrho_n \nabla p_n(y_n), \end{cases}$$

where $\nabla p_n(x_n) = \sum_{i=1}^N \sigma_i (I - P_{C_{i,n}})x_n + \sum_{j=1}^M v_j A^* (I - P_{Q_{j,n}})Ax_n$, $\sum_{i=1}^N \sigma_i + \sum_{j=1}^M v_j = 1$ for $\sigma_i > 0, v_j > 0$, $\gamma > 0, \zeta \in (0, 1), \mu \in (0, 1)$.

In our algorithm we took $\zeta_{i,n} = (-1)^i x_i^0$ for each $i \in \{1, \dots, N\}$ and

$$\varepsilon_{1,n} = \begin{cases} \frac{z_n - z_1^0}{\|z_n - z_1^0\|}, & \text{if } z_n - z_1^0 \neq 0, \\ 0, & \text{otherwise,} \end{cases}$$

and $\varepsilon_{j,n} = (-1)^j z_j^0$ for each $j \in \{2, \dots, M\}$.

For the purpose of comparison we took the following data:

HSRPA: $\lambda_1 = \lambda_2 = \frac{1}{2}, \delta_j = \frac{j}{1+\dots+M}$ for $j \in \{1, \dots, M\}, \alpha_n = \frac{1}{n+1}, \beta_n = \frac{n+2}{2n+6}, \rho_n = 1$.

GPM: $\varrho = \frac{1}{3}, \sigma_i = \frac{i}{2(1+\dots+N)}$ for $i \in \{1, \dots, N\}, v_j = \frac{j}{2(1+\dots+M)}$ for $j \in \{1, \dots, M\}$.

PPM: $\varrho = \frac{1}{3}, \sigma_i = \frac{i}{2(1+\dots+N)}$ for $i \in \{1, \dots, N\}, v_j = \frac{j}{2(1+\dots+M)}$ for $j \in \{1, \dots, M\}$.

SAPM: $\gamma = 1, \zeta = \frac{1}{2} = \mu, \sigma_i = \frac{i}{2(1+\dots+N)}$ for $i \in \{1, \dots, N\}, v_j = \frac{j}{2(1+\dots+M)}$ for $j \in \{1, \dots, M\}$.

The numerical results are shown in Table 1 and Figure 3. To permit comparisons between the three algorithms about the number of iterations (Iter(n)) and the time of execution in seconds (CPUt(s)), we have used the relative difference $\frac{\|x_n - x_{n+1}\|}{\|x_1 - u\|}$, which is less than ϵ , as the stopping criteria in Table 1.

From the numerical results in Table 1 and Figure 3 of Example 2, we can see that our algorithm (HSRPA) has better performance than GPM, PPM and SAPM. To be specific, HSRPA converges faster and requires less iterations than GPM, PPM and SAPM. In view of CPU time of execution, our algorithm has comparable performance with GPM, PPM and SAPM.

Table 1. For $\theta = 3, u = (4, -6, 9)$ and $x_1 = (-4, 7, -19)$.

		$\epsilon = 10^{-2}, N = 3, M = 2$	$\epsilon = 10^{-3}, N = 4 = M$
HSRPA	Iter(n)	271	359
	CPUt(s)	2.478534	2.7094060
	$\ x_n - \bar{x}\ $	$1.29153127891261 \times 10^{-11}$	$3.95548438843874 \times 10^{-5}$
GPM	Iter(n)	286	421
	CPUt(s)	2.350976	3.0921890
	$\ x_n - \bar{x}\ $	$1.32185889807060 \times 10^{-11}$	$4.00584274497442 \times 10^{-5}$
PPM	Iter(n)	293	368
	CPUt(s)	2.462534	2.6903446
	$\ x_n - \bar{x}\ $	$1.34325049285599 \times 10^{-11}$	$3.97217074806315 \times 10^{-5}$
SAPM	Iter(n)	277	397
	CPUt(s)	2.192005	2.6393334
	$\ x_n - \bar{x}\ $	$1.30707774298130 \times 10^{-11}$	$4.0068355895327 \times 10^{-5}$

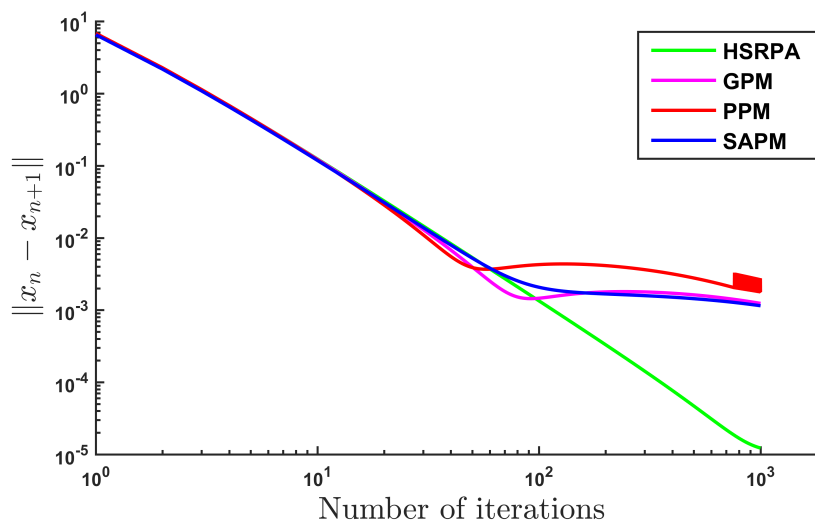


Figure 3. For $N = 50$ and starting points $u = (-1, \dots, -1) \in \mathbb{R}^N$ (in HSRPA) and $x_1 = -2u$.

Example 3. Consider the MSSFP with $H_1 = \mathbb{R}^s, H_2 = \mathbb{R}^t$ and $C_i = \{x \in \mathbb{R}^s : -5ie_1 \leq x \leq (N - i)e_1\}$ ($i \in \{1, \dots, N\}$) and $Q_j = \{y \in \mathbb{R}^t : (j - M)e_1 \leq y \leq je_1\}$, ($j \in \{1, \dots, M\}$), with a different number of feasible sets N and M , and dimensions s and t . We randomly generated the operator $A = (a_{ij})_{t \times s}$, with $a_{ij} \in [0, 10]$. In this example, we see the effect of the different choices of λ_1 and λ_2 for the numerical results of the HSRPA using both stepsizes $\tau_{j,n}$ and $\tau_{j,n}^*$ (given in (38)), and also we compare the HSRPA with the algorithm proposed by Zhao et al. ([48], Algorithm 2.1). For the HSRPA we take $u = (1, 1, 2)^T, x_1 = \text{rand}[-100, 100]$, $N = 3, M = 10, s = t = 3, \rho_n = 3, \alpha_n = \frac{1}{n+1}, \beta_n = \frac{n+5}{2n+6}$ and $\delta_j = \frac{j}{1+\dots+M}, j \in \{1, \dots, M\}$. For ([48], Algorithm 2.1), we take $x^0 = \text{rand}[-100, 100], t = 3, r = 10, N = M = 3, \omega_k = 1, \alpha_i = (\frac{1013}{3000})^i, i = 1, 2, 3,$ and $\beta_j = (\frac{1013}{3000})^j, j = 1, 2, \dots, 10$. We use $\|x_{n+1} - x_n\| < 10^{-4}$ as the stopping criteria. The results are presented in Table 2 below.

Interestingly, it can be observed from Table 2 that, for $\lambda_1 \leq \lambda_2$ HSRPA with the stepsize $\tau_{j,n}$ is faster in terms of less number of iterations and CPU-run time than HSRPA with the stepsize $\tau_{j,n}^*$. On the contrary, HSRPA ($\tau_{j,n}^*$) has better performance for $\lambda_1 > \lambda_2$ and HSRPA with either of the stepsizes converges faster and requires less number of iterations than the compared algorithm ([48], Algorithm 2.1).

Table 2. Comparative result of the Half-Space Relaxation Projection Algorithm (HSRPA) (with two different stepsizes $\tau_{j,n}$ and $\tau_{j,n}^*$) with ([48], Algorithm 2.1) for different choices of λ_1 and λ_2 .

Choices of λ_1, λ_2	HSRPA ($\tau_{j,n}$)		HSRPA ($\tau_{j,n}^*$)		Algorithm 2.1	
	Ite.	CPUt(s)	Ite.	CPUt(s)	Ite.	CPUt(s)
$\lambda_1 = 0.9, \lambda_2 = 0.1$	56	8.9971	51	7.3628	235	28.7590
$\lambda_1 = 0.8, \lambda_2 = 0.2$	55	5.8883	37	3.7528		
$\lambda_1 = 0.7, \lambda_2 = 0.3$	45	4.7402	39	3.9150		
$\lambda_1 = 0.6, \lambda_2 = 0.4$	42	4.6510	38	3.9080		
$\lambda_1 = 0.5, \lambda_2 = 0.5$	38	3.9295	45	4.5573		
$\lambda_1 = 0.4, \lambda_2 = 0.6$	36	3.1803	44	4.4400		
$\lambda_1 = 0.3, \lambda_2 = 0.7$	42	4.3067	50	5.0784		
$\lambda_1 = 0.2, \lambda_2 = 0.8$	29	3.4536	36	3.8640		
$\lambda_1 = 0.1, \lambda_2 = 0.9$	32	3.3220	38	4.1409		

Example 4. Consider the Hilbert space $H_1 = H_2 = L^2([0, 1])$ with norm $\|x\| := \sqrt{\int_0^1 |x(t)|^2 dt}$ and the inner product given by $\langle x, y \rangle = \int_0^1 x(t)y(t)dt$. The two nonempty, closed and convex sets are $C = \{x \in L^2([0, 1]) : \langle x(t), 3t^2 \rangle = 0\}$ and $Q = \{x \in L^2([0, 1]) : \langle x, \frac{t}{3} \rangle \geq -1\}$, and the linear operator is given as $(Ax)(t) = x(t)$, i.e., $\|A\| = 1$ or $A = I$ is the identity. The orthogonal projection onto C and Q have an explicit formula; see, for example, [57]

$$P_C(w(t)) = \begin{cases} w(t) - \frac{\langle w(t), 3t^2 \rangle}{\|3t^2\|_{L^2}^2} 3t^2, & \text{if } \langle w(t), 3t^2 \rangle \neq 0, \\ w(t), & \text{if } \langle w(t), 3t^2 \rangle = 0. \end{cases}$$

$$P_Q(w(t)) = \begin{cases} w(t) - \frac{\langle w(t), \frac{-t}{3} \rangle - 1}{\|\frac{-t}{3}\|_{L^2}^2} (\frac{-t}{3}), & \text{if } \langle w(t), \frac{-t}{3} \rangle < -1, \\ w(t), & \text{if } \langle w(t), \frac{-t}{3} \rangle \geq -1. \end{cases}$$

We consider the following problem

$$\text{find } x^* \in C \text{ such that } Ax^* \in Q. \tag{42}$$

It is clear that Problem (42) has a nonempty solution set Ω since $0 \in \Omega$. In this case, the iterative scheme in Algorithm 2 ($u, x_1 \in C$, with $\lambda_1 = \lambda_2 = \frac{1}{2}$, $\rho_n = \frac{7}{2}$, $\alpha_n = \frac{1}{n+1}$, and $\beta_n = \frac{n+2}{2n+6}$) becomes

$$\begin{cases} z_n = (1 - \frac{1}{n+1})x_n + \frac{u}{n+1}, \\ y_n = z_n - \frac{7}{2}((I - P_{C_n})z_n + A^*(I - P_{Q_n})Az_n), \\ x_{n+1} = (1 - \frac{n+2}{2n+6})z_n + (\frac{n+2}{2n+6})y_n, \end{cases}$$

where

$$\tau_n = \frac{7(\|(I - P_{C_n})z_n\|^2 + \|(I - P_{Q_n})Az_n\|^2)}{d(z_n)},$$

for

$$d(z_n) = \begin{cases} 1, & \text{if } \|(I - P_{C_n})z_n\|^2 + \|A^*(I - P_{Q_n})Az_n\|^2 = 0 \\ \|(I - P_{C_n})z_n\|^2 + \|A^*(I - P_{Q_n})Az_n\|^2, & \text{otherwise.} \end{cases}$$

In this example, we compare Algorithm 2 with the strong convergence result of SFP proposed by Shehu et al. [56]. The iterative scheme (27) in [56] for $u, x_1 \in C$, with $\alpha_n = \frac{1}{n+1}$, $\beta_n = \frac{n}{2(n+1)} = \gamma_n$ and $t_n = \frac{1}{\|A\|^2}$ was reduced into the following form

$$\begin{cases} y_n = [x_n - \frac{1}{\|A\|^2} A^*(Ax_n - P_{Q_n}(Ax_n))] \\ x_{n+1} = P_C(\frac{u}{n+1} + \frac{nx_n}{2(n+1)} + \frac{ny_n}{2(n+1)}), n \geq 1. \end{cases} \tag{43}$$

We see here that our iterative scheme can be implemented to solve the problem (42) considered in this example. We use $\|x_{n+1} - x_n\| < 10^{-3}$ as stopping criteria for both algorithms and the outcome of the numerical experiment is reported in Figure 4. It can be observed from Figure 4 that, for different choices of u and x_1 , Algorithm 2 is faster in terms of less number of iterations and CPU-run time than the algorithm proposed by Shehu et al. [56].

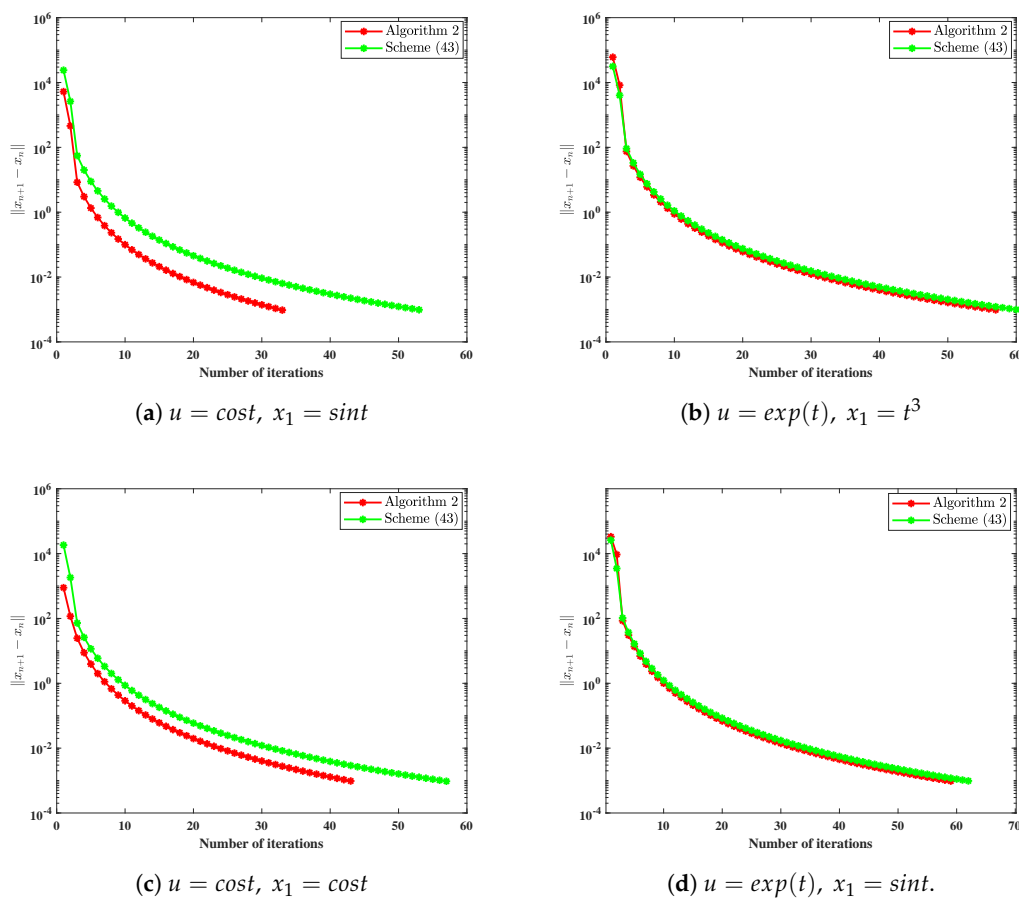


Figure 4. Comparison of Algorithm 2 and algorithm by Shehu [56] for different choices of u and x_1 .

4.1. Application to Signal Recovery

In this part, we consider the problem of recovering a noisy sparse signal. Compressed sensing can be modeled as the following linear equation:

$$b = Ax + \epsilon, \tag{44}$$

where $x \in \mathbb{R}^N$ is a vector with L nonzero components to be recovered, $b \in \mathbb{R}^M$ is the measured data with noisy ϵ (when $\epsilon = 0$, it means that there is no noise to the observed data) and A is an $M \times N$ bounded linear observation operator with ($N > M$). The problem in Equation (44) can be seen as the LASSO problem, which has wide application in signal processing theory [58].

$$\begin{aligned} \min_{x \in \mathbb{R}^N} \frac{1}{2} \|Ax - b\|_2^2 \\ \text{subject to } \|x\|_1 \leq t, \end{aligned} \tag{45}$$

where $t > 0$ is a given constant. It can be observed that (45) indicates the potential of finding a sparse solution of the SFP (3) due to the ℓ_1 constraint. Thus, we apply Algorithm 2 to solve problem (45).

Let $C := \{x : \|x\|_1 \leq t\}$ and $Q = \{b\}$, then the minimization problem (45) can be seen as an SFP (3). Denote the level set C_n by,

$$C_n = \{x \in \mathbb{R}^N : \omega(x_n) + \langle \zeta_n, x - x_n \rangle \leq 0\}, \tag{46}$$

where $\zeta_n \in \partial\omega(x_n)$ with the convex function $\omega(x) = \|x\|_1 - t$.

For a special case where $Q = Q_n = \{b\}$, Algorithm 2 converges to the solution of (45) and moreover, since the projection onto the level set has an explicit formula, Algorithm 2 can be easily implemented. In the sequel, we present a sparse signal recovery experiment to illustrate the efficiency of Algorithm 2 by comparing with algorithms proposed by Lopez [2] and Yang [37].

The vector x is an L -sparse signal with L non-zero elements that is generated from uniform distribution in the interval $[-2, 2]$. The matrix A is generated from a normal distribution with mean zero and one variance. The observation b is generated by white Gaussian noise with signal-to-noise ratio $SNR = 40$. The process is started with $t = L$, x_0 and u are randomly generated $N \times 1$ vectors. The goal is then to recover the L -sparse signal x by solving (45). The restoration accuracy is measured by the mean squared error as follows:

$$MSE = \frac{1}{N} \|x_{n+1} - x_n\| \leq \epsilon, \tag{47}$$

where x_n is an estimated signal of x , and $\epsilon > 0$ is a given small constant. We take the stopping criteria $\epsilon = 10^{-6}$ and we choose the corresponding parameters $\rho_n = 3.5, \lambda_1 = \lambda_2 = 0.5$ for Algorithm 2. We also choose $\gamma = \frac{1}{\|A\|^2}$ and $\rho_n = 2$ for the algorithm by Yang [37] and algorithm by Lopez [2], respectively.

It can be observed from Figures 5–8 that the recovered signal by the proposed algorithm has less number of iterations and MSE. Therefore, the quality of the signal recovered by the proposed algorithm is better than the compared algorithms.

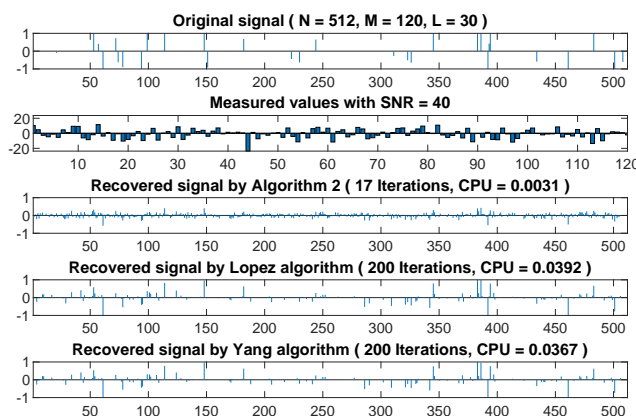


Figure 5. The original L -sparse signal versus the recovered sparse signal by Algorithm 2, algorithms by Lopez [2] and Yang [37] when $N = 512, M = 120$ and $L = 30$.

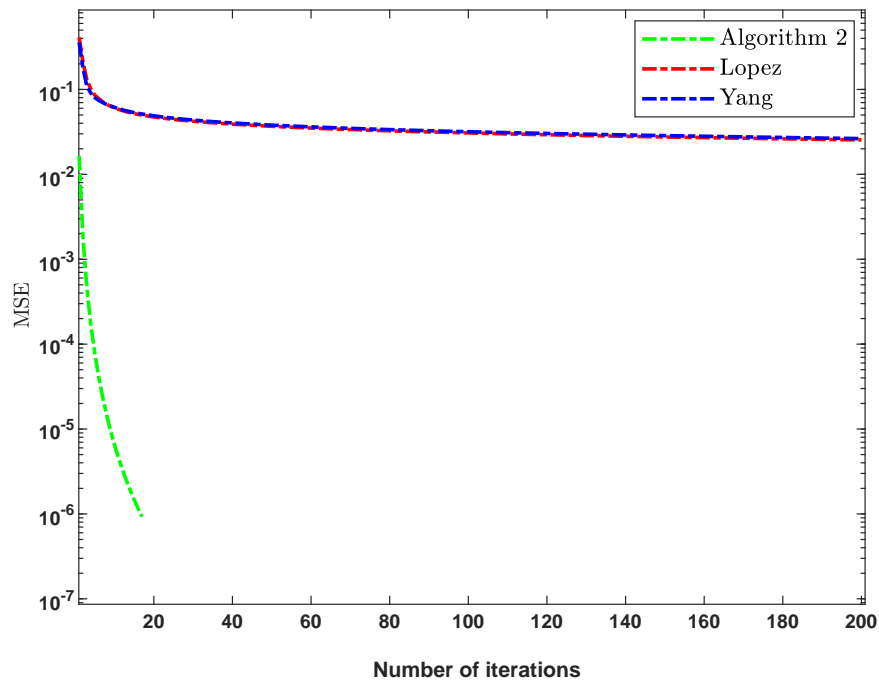


Figure 6. MSE versus the number of iterations, and the comparison of Algorithm 2 with that of algorithms by Lopez [2] and Yang [37] when $N = 512$, $M = 120$ and $L = 30$.

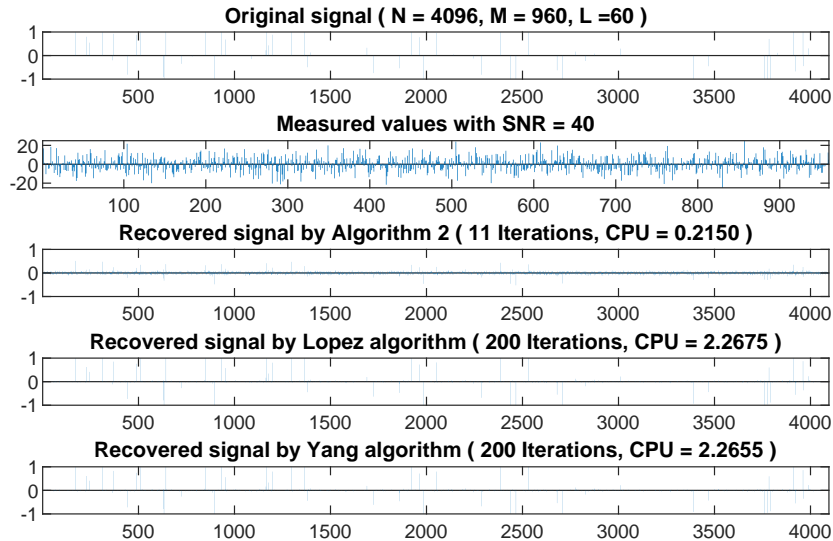


Figure 7. The original L -sparse signal versus the recovered sparse signal by Algorithm 2, algorithms by Lopez [2] and Yang [37] when $N = 4096$, $M = 960$ and $L = 60$.

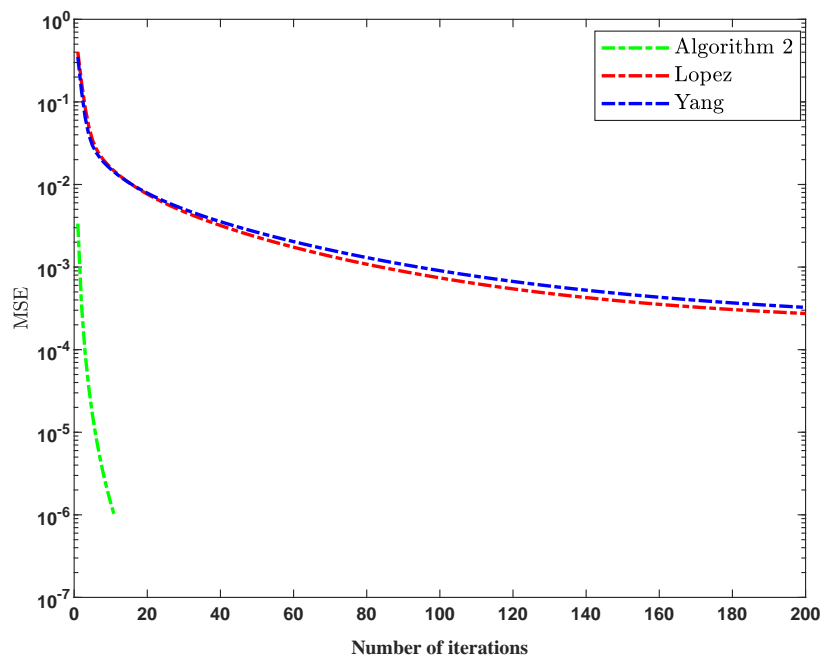


Figure 8. MSE versus the number of iterations, and the comparison of Algorithm 2 with that of algorithms by Lopez [2] and Yang [37] when $N = 4096$, $M = 960$ and $L = 60$.

5. Conclusions

In this paper, we present a strong convergence iterative algorithm solving MSSFP with a way of selecting the stepsizes such that the implementation of the algorithm does not need any prior information as regards the operator norms. Preliminary numerical results are reported to illustrate the numerical behavior of our algorithm (HSRPA), and to compare it with those well known in the literature, including, Censor et al. ([7], Algorithm 1), Censor et al. ([43], Algorithm 5), Zhao and Yang ([46], Algorithm 3.2) and Zhao et al. ([48], Algorithm 2.1). The numerical results show that our proposed Algorithm is practical and promising for solving MSSFP. Algorithm 2 is applied in signal recovery. The experiment results show that Algorithm 2 has fewer iterations than that of algorithms proposed by Yang [37], Lopez [2] and Shehu [56].

Author Contributions: Conceptualization, G.H.T. and K.S.; data curation, A.G.G.; formal analysis, G.H.T.; funding acquisition, P.K.; investigation, K.S.; methodology, K.S.; project administration, P.K.; resources, P.K.; software, A.G.G.; supervision, P.K.; validation, G.H.T. and P.K.; visualization, A.G.G.; writing—original draft, G.H.T.; writing—review and editing, A.G.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Center of Excellence in Theoretical and Computational Science (TaCS-CoE), Faculty of Science, KMUTT. The first author was supported by the “Petchra Pra Jom Klao Ph.D. Research Scholarship” from King Mongkut’s University of Technology Thonburi with Grant No. 37/2561.

Acknowledgments: The authors acknowledge the financial support provided by the Center of Excellence in Theoretical and Computational Science (TaCS-CoE), KMUTT. Guash Haile Taddele is supported by the Petchra Pra Jom Klao Ph.D. Research Scholarship from King Mongkut’s University of Technology Thonburi (Grant No.37/2561). Moreover, Kanokwan Sitthithakerngkiet was supported by the Faculty of Applied Science, King Mongkut’s University of Technology, North Bangkok. Contract no. 6342101.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Censor, Y.; Gibali, A.; Reich, S. Algorithms for the split variational inequality problem. *Numer. Algorithms* **2012**, *59*, 301–323. [[CrossRef](#)]
2. López, G.; Martín-Márquez, V.; Wang, F.; Xu, H.K. Solving the split feasibility problem without prior knowledge of matrix norms. *Inverse Probl.* **2012**, *28*, 085004. [[CrossRef](#)]
3. Byrne, C. A unified treatment of some iterative algorithms in signal processing and image reconstruction. *Inverse Probl.* **2003**, *20*, 103. [[CrossRef](#)]
4. Combettes, P. The convex feasibility problem in image recovery. In *Advances in Imaging and Electron Physics*; Elsevier: Amsterdam, The Netherlands, 1996; Volume 95, pp. 155–270.
5. Qu, B.; Xiu, N. A note on the CQ algorithm for the split feasibility problem. *Inverse Probl.* **2005**, *21*, 1655. [[CrossRef](#)]
6. Censor, Y.; Elfving, T. A multiprojection algorithm using Bregman projections in a product space. *Numer. Algorithms* **1994**, *8*, 221–239. [[CrossRef](#)]
7. Censor, Y.; Elfving, T.; Kopf, N.; Bortfeld, T. The multiple-sets split feasibility problem and its applications for inverse problems. *Inverse Probl.* **2005**, *21*, 2071. [[CrossRef](#)]
8. Censor, Y.; Lent, A. Cyclic subgradient projections. *Math. Program.* **1982**, *24*, 233–235. [[CrossRef](#)]
9. Byrne, C.; Censor, Y.; Gibali, A.; Reich, S. The split common null point problem. *arXiv* **2011**, arXiv:1108.5953. [[CrossRef](#)]
10. Censor, Y.; Segal, A. The split common fixed point problem for directed operators. *J. Convex Anal.* **2009**, *16*, 587–600. [[CrossRef](#)]
11. Dang, Y.; Gao, Y.; Li, L. Inertial projection algorithms for convex feasibility problem. *J. Syst. Eng. Electron.* **2012**, *23*, 734–740.
12. He, Z. The split equilibrium problem and its convergence algorithms. *J. Inequalities Appl.* **2012**, *2012*, 162. [[CrossRef](#)]
13. Taiwo, A.; Jolaoso, L.O.; Mewomo, O.T. Viscosity approximation method for solving the multiple-set split equality common fixed-point problems for quasi-pseudocontractive mappings in Hilbert spaces. *J. Ind. Manag. Optim.* **2017**, *13*. [[CrossRef](#)]
14. Aremu, K.O.; Izuchukwu, C.; Ogwo, G.N.; Mewomo, O.T. Multi-step iterative algorithm for minimization and fixed point problems in p-uniformly convex metric spaces. *J. Manag. Optim.* **2017**, *13*. [[CrossRef](#)]
15. Oyewole, O.K.; Abass, H.A.; Mewomo, O.T. A strong convergence algorithm for a fixed point constrained split null point problem. *Rend. Circ. Mat. Palermo Ser. 2* **2020**, *13*, 1–20. [[CrossRef](#)]
16. Izuchukwu, C.; Ugwunnadi, G.C.; Mewomo, O.T.; Khan, A.R.; Abbas, M. Proximal-type algorithms for split minimization problem in P-uniformly convex metric spaces. *Numer. Algorithms* **2019**, *82*, 909–935. [[CrossRef](#)]
17. Jolaoso, L.O.; Alakoya, T.O.; Taiwo, A.; Mewomo, O.T. Inertial extragradient method via viscosity approximation approach for solving equilibrium problem in Hilbert space. *Optimization* **2020**, 1–20. [[CrossRef](#)]
18. He, S.; Wu, T.; Cho, Y.J.; Rassias, T.M. Optimal parameter selections for a general Halpern iteration. *Numer. Algorithms* **2019**, *82*, 1171–1188. [[CrossRef](#)]
19. Dadashi, V.; Postolache, M. Forward–backward splitting algorithm for fixed point problems and zeros of the sum of monotone operators. *Arab. J. Math.* **2020**, *9*, 89–99. [[CrossRef](#)]
20. Yao, Y.; Leng, L.; Postolache, M.; Zheng, X. Mann-type iteration method for solving the split common fixed point problem. *J. Nonlinear Convex Anal.* **2017**, *18*, 875–882. [[CrossRef](#)]
21. Byrne, C. Iterative oblique projection onto convex sets and the split feasibility problem. *Inverse Probl.* **2002**, *18*, 441. [[CrossRef](#)]
22. Dang, Y.; Gao, Y. The strong convergence of a KM–CQ-like algorithm for a split feasibility problem. *Inverse Probl.* **2010**, *27*, 015007.
23. Jung, J.S. Iterative algorithms based on the hybrid steepest descent method for the split feasibility problem. *J. Nonlinear Sci. Appl.* **2016**, *9*, 4214–4225. [[CrossRef](#)]
24. Wang, F.; Xu, H.K. Cyclic algorithms for split feasibility problems in Hilbert spaces. *Nonlinear Anal. Theory Methods Appl.* **2011**, *74*, 4105–4111. [[CrossRef](#)]
25. Xu, H.K. An iterative approach to quadratic optimization. *J. Optim. Theory Appl.* **2003**, *116*, 659–678. [[CrossRef](#)]

26. Xu, H.K. A variable Krasnosel'skii–Mann algorithm and the multiple-set split feasibility problem. *Inverse Probl.* **2006**, *22*, 2021. [[CrossRef](#)]
27. Xu, H.K. Iterative methods for the split feasibility problem in infinite-dimensional Hilbert spaces. *Inverse Probl.* **2010**, *26*, 105018. [[CrossRef](#)]
28. Yu, X.; Shahzad, N.; Yao, Y. Implicit and explicit algorithms for solving the split feasibility problem. *Optim. Lett.* **2012**, *6*, 1447–1462. [[CrossRef](#)]
29. Shehu, Y.; Mewomo, O.T.; Ogbuisi, F.U. Further investigation into approximation of a common solution of fixed point problems and split feasibility problems. *Acta Math. Sci.* **2016**, *36*, 913–930. [[CrossRef](#)]
30. Shehu, Y.; Mewomo, O.T. Further investigation into split common fixed point problem for demicontractive operators. *Acta Math. Sin. Engl. Ser.* **2016**, *32*, 1357–1376. [[CrossRef](#)]
31. Mewomo, O.T.; Ogbuisi, F.U. Convergence analysis of an iterative method for solving multiple-set split feasibility problems in certain Banach spaces. *Quaest. Math.* **2018**, *41*, 129–148. [[CrossRef](#)]
32. Dong, Q.L.; Tang, Y.C.; Cho, Y.J.; Rassias, T.M. "Optimal" choice of the step length of the projection and contraction methods for solving the split feasibility problem. *J. Glob. Optim.* **2018**, *71*, 341–360. [[CrossRef](#)]
33. Cegielski, A. Landweber-type operator and its properties. *Contemp. Math.* **2016**, *658*, 139–148. [[CrossRef](#)]
34. Cegielski, A. General method for solving the split common fixed point problem. *J. Optim. Theory Appl.* **2015**, *165*, 385–404. [[CrossRef](#)]
35. Cegielski, A.; Reich, S.; Zalas, R. Weak, strong and linear convergence of the CQ-method via the regularity of Landweber operators. *Optimization* **2020**, *69*, 605–636.
36. Hendrickx, J.M.; Olshevsky, A. Matrix p-norms are NP-hard to approximate if $p \neq 1, 2, \infty$. *SIAM J. Matrix Anal. Appl.* **2010**, *31*, 2802–2812. [[CrossRef](#)]
37. Yang, Q. The relaxed CQ algorithm solving the split feasibility problem. *Inverse Probl.* **2004**, *20*, 1261. [[CrossRef](#)]
38. Bauschke, H.H.; Borwein, J.M. On projection algorithms for solving convex feasibility problems. *SIAM Rev.* **1996**, *38*, 367–426. [[CrossRef](#)]
39. Fukushima, M. A relaxed projection method for variational inequalities. *Math. Program.* **1986**, *35*, 58–70. [[CrossRef](#)]
40. Alakoya, T.O.; Jolaoso, L.O.; Mewomo, O.T. Modified inertial subgradient extragradient method with self adaptive stepsize for solving monotone variational inequality and fixed point problems. *Optimization* **2020**, 1–30. [[CrossRef](#)]
41. Jolaoso, L.O.; Taiwo, A.; Alakoya, T.O.; Mewomo, O.T. A unified algorithm for solving variational inequality and fixed point problems with application to the split equality problem. *Comput. Appl. Math.* **2020**, *39*, 38. [[CrossRef](#)]
42. Buong, N. Iterative algorithms for the multiple-sets split feasibility problem in Hilbert spaces. *Numer. Algorithms* **2017**, *76*, 783–798. [[CrossRef](#)]
43. Censor, Y.; Motova, A.; Segal, A. Perturbed projections and subgradient projections for the multiple-sets split feasibility problem. *J. Math. Anal. Appl.* **2007**, *327*, 1244–1256. [[CrossRef](#)]
44. Latif, A.; Vahidi, J.; Eslamian, M. Strong convergence for generalized multiple-set split feasibility problem. *Filomat* **2016**, *30*, 459–467. [[CrossRef](#)]
45. Masad, E.; Reich, S. A note on the multiple-set split convex feasibility problem in Hilbert space. *J. Nonlinear Convex Anal.* **2007**, *8*, 367. [[CrossRef](#)]
46. Zhao, J.; Yang, Q. Self-adaptive projection methods for the multiple-sets split feasibility problem. *Inverse Probl.* **2011**, *27*, 035009. [[CrossRef](#)]
47. Zhao, J.; Yang, Q. Several acceleration schemes for solving the multiple-sets split feasibility problem. *Linear Algebra Appl.* **2012**, *437*, 1648–1657.
48. Zhao, J.; Yang, Q. A simple projection method for solving the multiple-sets split feasibility problem. *Inverse Probl. Sci. Eng.* **2013**, *21*, 3537–3546. [[CrossRef](#)]
49. Yao, Y.; Postolache, M.; Zhu, Z. Gradient methods with selection technique for the multiple-sets split feasibility problem. *Optimization* **2019**, *69*, 269–281. [[CrossRef](#)]
50. Osilike, M.O.; Isiogugu, F.O. Weak and strong convergence theorems for nonspreading-type mappings in Hilbert spaces. *Nonlinear Anal. Theory Methods Appl.* **2011**, *74*, 1814–1822. [[CrossRef](#)]
51. Aubin, J.P. *Optima and Equilibria: An Introduction to Nonlinear Analysis*; Springer Science & Business Media: Berlin, Germany, 2013; Volume 140. [[CrossRef](#)]

52. Ceng, L.C.; Ansari, Q.H.; Yao, J.C. An extragradient method for solving split feasibility and fixed point problems. *Comput. Math. Appl.* **2012**, *64*, 633–642. [[CrossRef](#)]
53. Bauschke, H.H.; Combettes, P.L. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*; Springer: Berlin/Heidelberg, Germany, 2011; Volume 408.
54. Maingé, P.E.; Mărușter, Ș. Convergence in norm of modified Krasnoselski–Mann iterations for fixed points of demicontractive mappings. *Appl. Math. Comput.* **2011**, *217*, 9864–9874. [[CrossRef](#)]
55. He, B. Inexact implicit methods for monotone general variational inequalities. *Math. Program.* **1999**, *86*, 199–217.
56. Shehu, Y. Strong convergence result of split feasibility problems in Banach spaces. *Filomat* **2017**, *31*, 1559–1571. [[CrossRef](#)]
57. Cegielski, A. *Iterative Methods for Fixed Point Problems in Hilbert Spaces*; Springer: Berlin/Heidelberg, Germany, 2012; Volume 2057. [[CrossRef](#)]
58. Gibali, A.; Liu, L.W.; Tang, Y.C. Note on the modified relaxation CQ algorithm for the split feasibility problem. *Optim. Lett.* **2018**, *12*, 817–830. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).