J. Appl. Numer. Optim. 5 (2023), No. 3, pp. 371-390
Available online at http://jano.biemdas.com
https://doi.org/10.23952/jano.5.2023.3.06

SELF-ADAPTIVE STEPSIZE METHOD WITH INERTIAL EFFECTS FOR SOLVING GENERALIZED SPLIT FEASIBILITY PROBLEMS WITH APPLICATIONS

CHINEDU IZUCHUKWU[1,*], MAGGIE APHANE[2], KAZEEM OLALEKAN AREMU[3]

[1]*School of Mathematics, University of the Witwatersrand, Private Bag 3, Johannesburg 2050, South Africa*
[2]*Department of Mathematics and Applied Mathematics, Sefako Makgatho Health Sciences University, Ga-Rankuwa, Pretoria 0204, South Africa*
[3]*Department of Mathematics, Usmanu Danfodiyo University Sokoto, 2346 Sokoto, Nigeria*

Abstract. In this paper, we consider a class of generalized split feasibility problem over the solution set of monotone variational inclusion problems, which includes the split common null point problem, the split variational problems, and other related split type problems. We propose a new self-adaptive stepsize method coupled with inertial extrapolation techniques for solving this problem in real Hilbert spaces. We prove that the sequence generated by the proposed method converges strongly to a solution of the problem under the assumption that the associated singlevalued operator in the monotone variational inclusion problem is not required to be inverse-strongly monotone. Our method uses the stepsizes that are generated at each iteration by some simple calculations, which allows it to be easily implemented without the prior knowledge of the operator norm or the Lipschitz constant of the singlevalued operator. Finally, we apply our results to solve optimal control problems, split linear inverse problems, and least absolute selection and shrinkage operator problems.

Keywords. Generalized split feasibility problems; Inertial effects; Self-adaptive stepsize method.

2020 Mathematics Subject Classification. 65K15, 90C30.

1. INTRODUCTION

Let C and Q be nonempty, closed, and convex subsets of real Hilbert spaces H_1 and H_2, respectively. Let $T : H_1 \to H_2$ be a bounded linear operator. The Split Feasibility Problem (SFP) is formulated as:

$$\text{Find } x^* \in C \text{ such that } y^* = Tx^* \in Q. \tag{1.1}$$

This problem was first investigated by Censor and Elfving [1] (see also [2, 3]) in finite dimensional spaces. The SFP finds numerous real-world applications, such as intensity-modulated radiation therapy treatment planning, phase retrieval, image and signal processing, data compression, computerized tomography, and so on; see, e.g., [3, 4, 5] and the references therein.

In 2012, Byrne *et al.* [6] considered the following problem: Given set-valued mappings $B_i : H_1 \to 2^{H_1}$, $1 \le i \le m$, and $M_j : H_2 \to 2^{H_2}$ $1 \le j \le N$, respectively, and bounded linear operators

*Corresponding author.
E-mail address: chinedu.izuchukwu@wits.ac.za (C. Izuchukwu).
Received June 9, 2023; Accepted August 18, 2023.

$T_j : H_1 \to H_2$, $1 \leq j \leq N$, the Split Common Null Point Problem (SCNPP) that they investigated is to find $x \in H_1$ such that

$$x \in \left(\cap_{i=1}^{m} B_i^{-1}(0)\right) \cap \left(\cap_{j=1}^{N} T_j^{-1}\left(M_j^{-1}(0)\right)\right),$$

where $B_i^{-1}(0)$ and $M_j^{-1}(0)$ are null point sets of B_i and M_j, respectively. When $m = N = 1$, their problem reduces to finding a point $x \in H_1$ such that

$$x \in B^{-1}(0) \ \text{ and } \ Tx \in M^{-1}(0). \tag{1.2}$$

When problem (1.2) is viewed separately, the problem of finding $x \in H$ such that $x \in B^{-1}(0)$ is the classical Null Point Problem (NPP). Byrne *et al.* [6] proposed the following algorithm and established some convergence theorems for solving SCNPP (1.2) when B and M are maximal monotone operators

$$x_{n+1} = (I + \lambda B)^{-1}(x_n + \tau T^*((I + \lambda M)^{-1} - I)Tx_n), n \geq 1,$$

where $\tau \in (0, \frac{1}{L})$ with L being the spectral radius of the operator T^*T.

In [7, Theorems 4.2 and 4.3], Takahashi *et al.* introduced and studied the following Generalized Split Feasibility Problem (GSFP) over the solution set of NPP:

$$\text{Find } x \in H_1 \text{ such that } \ x \in B^{-1}(0) \ \text{ and } \ Tx \in Fix(S), \tag{1.3}$$

where $S : H_2 \to H_2$ is a nonexpansive mapping, $Fix(S)$ is the set of fixed points of S, and $B : H_1 \to 2^{H_1}$ is a maximal monotone operator. To solve the GSFP (1.3), Takahashi *et al.* [7] proposed the following two methods:

$$x_{n+1} = (I + \lambda_n B)^{-1}(x_n + \tau_n T^*(S - I)Tx_n), \ n \geq 1, \tag{1.4}$$

where $0 < \liminf_{n\to\infty} \lambda_n \leq \limsup_{n\to\infty} \lambda_n < \infty$ and $0 < \liminf_{n\to\infty} \tau_n \leq \limsup_{n\to\infty} \tau_n < \frac{1}{||T||^2}$ and

$$x_{n+1} = \beta_n x_n + (1 - \beta_n)(I + \lambda_n B)^{-1}(x_n + \lambda_n T^*(S - I)Tx_n), \ n \geq 1, \tag{1.5}$$

where $\sum_{n=1}^{\infty} \beta_n(1 - \beta_n) = \infty$, $0 < a \leq \lambda_n \leq b < \frac{1}{||T||^2}$, and $\sum_{n=1}^{\infty} |\lambda_n - \lambda_{n+1}| < \infty$. They proved in [7, Theorem 4.2 and Theorem 4.3] that Algorithm (1.4) and Algorithm (1.5) respectively, converge weakly to a solution of the GSFP (1.3). Another form of the SIP which is more general than the SFP (1.1) is the following Split Variational Inequality Problem (SVIP), introduced and studied by Censor *et al.* [3] (see also [2]):

$$\text{Find } x^* \in C \text{ such that } \ \langle A(x^*), x - x^* \rangle \geq 0, \ \forall x \in C, \tag{1.6}$$

and such that $y^* = Tx^* \in Q$ solves

$$\langle g(y^*), y - y^* \rangle \geq 0, \ \forall y \in Q, \tag{1.7}$$

where C and Q are nonempty, closed, and convex subsets of real Hilbert spaces H_1 and H_2, respectively, $T : H_1 \to H_2$ is a bounded linear operator, $A : H_1 \to H_1$ and $g : H_2 \to H_2$ are two given operators. Similar to the case of SCNPP (1.2), when problem (1.6)-(1.7) is viewed separately, (1.6) is the classical Variational Inequality Problem (VIP). To solve the SVIP (1.6)-(1.7), Censor *et al.* [3] proposed the following method:

$$x_{n+1} = P_C(I - \lambda A)(x_n + \tau T^*(P_Q(I - \lambda g) - I)Tx_n), n \geq 1, \tag{1.8}$$

where $\tau \in (0, \frac{1}{L})$ with L being the spectral radius of T^*T. They proved that their algorithm converges weakly to a solution of problem (1.6)-(1.7) under the assumption that $A,\ g$ are α_1, α_2-inverse-strongly monotone and $\lambda \in (0, 2\alpha)$ (where $\alpha := \min\{\alpha_1, \alpha_2\}$).

Motivated by the results of Takahashi *et al.* [7] and Censor *et al.* [3], Tian and Jiang [8] introduced and studied the following GSFP over the solution set of VIP:

$$\text{Find } x \in C \text{ such that } \langle Ax, y - x \rangle \geq 0 \ \ \forall\, y \in C \ \text{ and } Tx \in Fix(S), \tag{1.9}$$

where C is a nonempty, closed, and convex subset H_1, $T : H_1 \to H_2$ is a bounded and linear operator, $A : H_1 \to H_1$ is a given operator, and $S : H_2 \to H_2$ is nonexpansive. They proposed the following algorithm for solving (1.9):

$$\begin{cases} y_n = P_C(x_n - \tau_n T^*(I - S)Tx_n), \\ t_n = P_C(y_n - \lambda_n A(y_n)), \\ w_n = P_C(y_n - \lambda_n A(t_n)), \\ x_{n+1} = \alpha_n f(x_n) + (1 - \alpha_n) w_n,\ n \geq 1; \end{cases} \tag{1.10}$$

where $\{\alpha_n\} \subset (0,1)$ with $\lim\limits_{n\to\infty} \alpha_n = 0$ and $\sum\limits_{n=1}^{\infty} \alpha_n = \infty$, f is a contraction mapping, $\{\tau_n\} \subset [a,b]$ for some $a, b \in (0, \frac{1}{||T||^2})$, $\{\lambda_n\} \subset [c,d]$ for some $c, d \in (0, \frac{1}{L})$, and A is a monotone and L-Lipschitz continuous operator.

Based on the work of Censor *et al.* [3], Moudafi [9] recently introduced and studied a new type of SIP, called the Split Monotone Variational Inclusion Problem (SMVIP), which is to find

$$x \in H_1 \text{ such that } 0 \in A(x) + B(x), \tag{1.11}$$

and such that $y = Tx \in H_2$ solves

$$0 \in g(y) + M(y), \tag{1.12}$$

where $B : H_1 \to 2^{H_1}$ and $M : H_2 \to 2^{H_2}$ are maximal monotone operators, $T : H_1 \to H_2$ is a bounded and linear operator, $A : H_1 \to H_1$ and $g : H_2 \to H_2$ are singlevalued operators. We also note that if (1.11) and (1.12) are considered separately, we have that (1.11) is the classical Monotone Variational Inclusion Problem (MVIP). To solve the SMVIP (1.11)-(1.12), Moudafi [9] proposed the following iterative algorithm and obtained weak convergence result:

$$x_{n+1} = (I + \lambda B)^{-1}(I - \lambda A)(x_n + \tau T^*((I + \lambda M)^{-1}(I - \lambda g) - I)Tx_n),\ n \geq 1, \tag{1.13}$$

where $A,\ g$ are α_1, α_2-inverse-strongly monotone operators with $\lambda \in (0, 2\alpha)$ ($\alpha = \min\{\alpha_1, \alpha_2\}$), $\tau \in (0, \frac{1}{L})$ with L being the spectral radius of the operator T^*T. For the results on SIPs, we refer to [10, 11, 12, 13, 14] and the references therein.

We mention here that all these papers share a common computational weakness, which is the fact that the stepsize τ (τ_n) depends on the operator norm $||T||$ of T. For Algorithms (1.10) and (1.13), stepsize $\{\lambda_n\}$ (or λ) depends on the knowledge of the coefficient of the operators A and g. We know that stepsizes play essential roles in the convergence properties of iterative methods, since the efficiency of the methods depends heavily on it. When the stepsize depends on the knowledge of either the operator norm or the coefficient of an operator, it usually slows down the convergence rate of the method. Moreover, in many practical cases, the operator norm or the coefficient of a given operator may not be known or may be difficult to estimate. Therefore, iterative methods that does not depend on any of these, are more applicable in practice. For this reason, López *et al.* [15] proposed the following self-adaptive stepsize method for solving the SFP (1.1):

$$x_{n+1} = P_C(x_n + \tau_n T^*(P_Q - I)Tx_n), n \geq 1,$$

where the stepsize $\{\tau_n\}$ is computed as $\tau_n := \frac{\rho_n f(x_n)}{||\nabla f(x_n)||^2}$, with $f(x) = \frac{1}{2}||(I-P_Q)Tx||^2$, $\nabla f(x) = T^*(I-P_Q)Tx,\ \ 0 < \rho_n < 4$, and $\inf \rho_n(4-\rho_n) > 0$.

Based on the result of López *et al.* [15], Moudafi and Thakur [16] proposed the split proximal method for solving split minimization problem (that is, finding a minimizer x of a convex function f such that Tx minimizes another convex function g):

$$x_{n+1} = \text{prox}_{\lambda \tau_n f}(x_n - \tau_n T^*(I - prox_{\lambda g})Tx_n),\ \ n \geq 1,$$

where $\text{prox}_{\lambda g}(x) = \arg\min_{u \in H_2}\{g(u) + \frac{1}{2\lambda}\|u-x\|^2\}$ and the stepsize $\tau_n := \rho_n \frac{h(x_n)+l(x_n)}{\theta^2(x_n)}$ with $0 \leq \rho_n < 4$, $\theta(x) := \sqrt{\|\nabla h(x)\|^2 + \|\nabla l(x)\|^2}$, $h(x) = \frac{1}{2}\|(I - prox_{\lambda g})Tx\|^2$, and $l(x) = \frac{1}{2}\|(I - prox_{\lambda \mu_n f})x\|^2$.

Motivated by the results of López et al. [15] and Moudafi and Thakur [16], Tang and Gibali [17] introduced the following self-adaptive stepsize method for solving the SCNPP (1.2):

$$\begin{cases} y_n = (1-\beta_n)x_n + \beta_n(I+\lambda B)^{-1}x_n, \\ x_{n+1} = \alpha_n x_n + (1-\alpha_n)(y_n + \tau_n T^*((I+\lambda M)^{-1} - I)Ty_n), n \geq 1, \end{cases}$$

where

$$\beta_n = \begin{cases} \rho_n \frac{h(x_n)}{||F(x_n)||^2 + ||H(x_n)||^2}, & ||F(x_n)||^2 + ||H(x_n)||^2 \neq 0, \\ 0, & \text{else}, \end{cases}$$

and

$$\tau_n = \begin{cases} \rho_n \frac{f(y_n)}{||F(x_n)||^2 + ||H(x_n)||^2}, & ||F(x_n)||^2 + ||H(x_n)||^2 \neq 0, \\ 0, & \text{else}, \end{cases}$$

with $f(x) = \frac{1}{2}||(I-(I+\lambda M)^{-1})Tx||^2$, $h(x) = \frac{1}{2}||(I-(I+\lambda B)^{-1})x||^2$, $F(x) = \nabla f(x)$, and $H(x) = \nabla h(x)$.

Motivated by the recent interest in this direction of research, it is our purpose in this paper to:

- study the following generalization of the SIPs (1.1), (1.2), (1.3), (1.6)-(1.7), (1.9), and (1.11)-(1.12); which we call the GSFP over the solution set of SMVIP.

$$\text{Find } \ x \in H_1 \text{ such that } 0 \in A(x) + B(x) \text{ and } y = Tx \in Fix(S), \tag{1.14}$$

  where $B : H_1 \to 2^{H_1}$ is a maximal monotone operator, $T : H_1 \to H_2$ is a bounded and linear operator, $A : H_1 \to H_1$ is a monotone and Lipschitz continuous operator, and $S : H_2 \to H_2$ is a nonexpansive mapping;
- propose a new iteratively generated self-adaptive stepsize procedure with the stepsizes $\{\tau_n\}$ and $\{\lambda_n\}$ being generated at each iteration by some simple calculations. Thus our method can be easily implemented;
- study some applications of the GSFP (1.14) to optimal control problems, split linear inverse problems, least absolute selection, and shrinkage operator problems via numerical computations.

By combining the new self-adaptive stepsize procedure with a viscosity method, and the inertial extrapolation technique, we propose a new method for solving the GSFP (1.14) in real Hilbert spaces. We prove that the sequence generated by this method converges strongly to a solution of the GSFP. We also derive the methods for solving problems (1.1), (1.2), (1.3), (1.6)-(1.7), (1.9), and (1.11)-(1.12).

We organize the rest of the paper as follows: We first recall some basic results in Section 2. Some discussions about our proposed method are given in Section 3. The convergence analysis of the proposed method is investigated in Section 4. Corollaries of our results are also discussed in Section 4. In Section 5, the established theorem is applied to solve optimal control problems, split linear inverse problems and least absolute selection and shrinkage operator problems, via numerical computations. We then conclude with some final remarks in Section 6, the last section.

2. Preliminaries

Let H be a real Hilbert space, and let A be a nonlinear operator defined on H. A point $x \in H$ is called a *fixed point* of A if $Ax = x$. The operator A is said to be

(i) α-*inverse-strongly monotone* (ism) if there exists $\alpha > 0$ such that $\langle Ax - Ay, x - y\rangle \geq \alpha\|Ax - Ay\|^2$ for all $x, y \in H$;
(ii) *monotone* if $\langle Ax - Ay, x - y\rangle \geq 0$ for all $x, y \in H$;
(iii) *L-Lipschitz continuous* if there exists a constant $L > 0$ such that $\|Ax - Ay\| \leq L\|x - y\|$ for all $x, y \in H$. If $L \in (0,1)$ then A is a *contraction* while A is *nonexpansive* if $L = 1$.

Clearly, α-inverse-strongly monotone operators are monotone. We also know that every α-inverse-strongly monotone operator is $\frac{1}{\alpha}$-Lipschitz continuous.

Let A be a multivalued operator, i.e. $A : H \to 2^H$. Recall that A is called *monotone* if $\langle x - y, u - v\rangle \geq 0$ for all $x, y \in H$, $u \in A(x)$, $v \in A(y)$, and A is *maximal monotone* if the graph $Gra(A)$ of A defined by $Gra(A) := \{(x,y) \in H \times H : y \in A(x)\}$ is not properly contained in the graph of any other monotone operator. It is generally known that A is maximal monotone if and only if, for $(x,u) \in H \times H$, $\langle x - y, u - v\rangle \geq 0$ for all $(y,v) \in Gra(A)$ implies $u \in A(x)$. The resolvent operator J_λ^A associated with a multivalued operator A and λ is the mapping $J_\lambda^A : H \to 2^H$ defined by $J_\lambda^A(x) = (I + \lambda A)^{-1}x$ for all $x \in H$ and $\lambda > 0$, where I is the identity operator on H. It is well-known that if the operator A is monotone, then J_λ^A is singlevalued and nonexpansive. For more details on monotone operators and their resolvents, we refer to [18, 19, 20, 21, 22, 23, 24].

Recall that a mapping $S_1 : H \to H$ is said to be averaged nonexpansive if $S_1 = (1-\beta)I + \beta S_2$ holds for a nonexpansive operator $S_2 : H \to H$ and $\beta \in (0,1)$. Recall that the metric projection, denoted as P_C, is a map defined from H onto C, where C a nonempty, closed, and convex subset H, which assigns each $x \in H$ to the unique point in C, denoted by $P_C x$ such that $||x - P_C x|| = \inf\{||x - y|| : \ y \in C\}$. It is well known that $P_C x$ is characterized by the inequality $\langle x - P_C x, z - P_C x\rangle \leq 0$ for all $z \in C$. We also define the normal cone of C at a point $z \in H$, as $N_C z := \{d \in H : \langle d, y - z\rangle \leq 0, \ \forall\ y \in C\}$ if $z \in C$ and $\emptyset$, otherwise.

The following equalities and inequalities are known in Hilbert spaces.

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

where $x, y \in H$ and $\alpha \in (0,1)$.

The following lemmas are needed for our strong convergence analysis.

Lemma 2.1. [25] *Let H be a real Hilbert space, and let $f : H \to H$ be a nonlinear mapping. Then*

(i) if f is η-ism and $\gamma > 0$, then γf is $\frac{\eta}{\gamma}$-ism;

(ii) f is averaged if and only if the complement $I - f$ is η-ism for some $\eta > \frac{1}{2}$. Indeed, for $\beta \in (0,1)$, f is β-averaged if and only if $I - f$ is $\frac{1}{2\beta}$-ism.

Lemma 2.2. [26] *Let H be a real Hilbert space, and let $S : H \to H$ be a nonexpansive mapping with $Fix(S) \neq \emptyset$. If $\{x_n\}$ is a sequence in H converging weakly to x^* and if $\{(I-S)x_n\}$ converges strongly to y, then $(I-S)x^* = y$.*

Lemma 2.3. [27] *Let H be a real Hilbert space, $A : H \to H$ be a monotone and Lipschitz continuous operator, and $B : H \to 2^H$ be a maximal monotone operator. Then, the operator $(A+B) : H \to 2^H$ is maximal monotone.*

Lemma 2.4. [28] *Let $\{a_n\}$ be a sequence of non-negative real numbers, $\{\beta_n\}$ be a sequence of real numbers in $(0,1)$ with condition $\sum_{n=1}^{\infty} \beta_n = \infty$, and $\{d_n\}$ be a sequence of real numbers. Assume that $a_{n+1} \leq (1-\beta_n)a_n + \beta_n d_n$ for $n \geq 0$. If $\limsup_{k\to\infty} d_{n_k} \leq 0$ for every subsequence $\{a_{n_k}\}$ of $\{a_n\}$ satisfying the condition: $\liminf_{k\to\infty} \left(a_{n_k+1} - a_{n_k}\right) \geq 0$, then $\lim\limits_{n\to\infty} a_n = 0$.*

Lemma 2.5. [29] *Let $\{a_n\}$ be a sequence of nonnegative real numbers satisfying the following: $a_{n+1} \leq (1-\beta_n)a_n + \sigma_n + \gamma_n$ for $n \geq 1$, where $\{\beta_n\}$ is a sequence in $(0,1)$ and $\{\sigma_n\}$ is a real sequence. Suppose that $\sum_{n=1}^{\infty} \gamma_n < \infty$ and $\sigma_n \leq \beta_n M$ for some $M \geq 0$. Then, $\{a_n\}$ is a bounded sequence.*

3. Proposed Method

In this section, we present our method and discuss some of its features. We begin with the following assumptions under which our strong convergence is obtained.

Assumption 3.1. H_1 and H_2 are two real Hilbert spaces. Furthermore, we assume that the following hold:

(a) $B : H_1 \to 2^{H_1}$ is a maximal monotone operator and $A : H_1 \to H_1$ is a monotone and Lipschitz continuous operator but the Lipschitz constant need not to be known.
(b) $T : H_1 \to H_2$ is a bounded linear operator such that $T \neq 0$.
(c) $S : H_2 \to H_2$ is nonexpansive and $f : H_1 \to H_1$ is a contraction with coefficient $\rho \in (0,1)$.
(c) The solution set $\Gamma := \{z \in (A+B)^{-1}(0) : Tz \in Fix(S)\}$ is nonempty.

Next, we state the conditions under which our parameters are chosen.

Assumption 3.2. Suppose that $\{\alpha_n\}$ and $\{\varepsilon_n\}$ are positive sequences satisfying the following conditions:

(a) $\{\alpha_n\} \subset (0,1)$ with $\lim_{n\to\infty} \alpha_n = 0$ and $\sum_{n=1}^{\infty} \alpha_n = \infty$.
(b) $\varepsilon_n = o(\alpha_n)$, where $\varepsilon_n = o(\alpha_n)$ means $\lim_{n\to\infty} \frac{\varepsilon_n}{\alpha_n} = 0$.

In the sequel, we define $Fx := T^*(I-S)Tx$. We shall see that F is Lipschitz continuous. We now present the proposed method of this paper.

Algorithm 3.3.

Step 0: Choose sequences $\{\alpha_n\}_{n=1}^{\infty}$ and $\{\varepsilon_n\}_{n=1}^{\infty}$ such that the conditions from Assumption 3.2 hold, and let $\tau_1, \lambda_1 > 0,\ \mu, \delta \in (0,1), \theta \geq 3$ and $x_0, x_1 \in H_1$ be given arbitrarily. Set $n := 1$.
Step 1. Given the iterates x_{n-1} and x_n $(n \geq 1)$, choose the sequence $\{\theta_n\}$ such that $0 \leq \theta_n \leq \bar{\theta}_n$, where

$$\bar{\theta}_n := \begin{cases} \min\left\{\frac{n-1}{n+\theta-1}, \frac{\varepsilon_n}{\|x_n - x_{n-1}\|}\right\} & \text{if } x_n \neq x_{n-1}, \\ \frac{n-1}{n+\theta-1} & \text{otherwise.} \end{cases}$$

Step 2. Set $u_n = x_n + \theta_n(x_n - x_{n-1})$. Then, compute

$$w_n = u_n - \tau_n F u_n \text{ and } y_n = (I + \lambda_n B)^{-1}(I - \lambda_n A) w_n. \tag{3.1}$$

Step 3. Compute $x_{n+1} = \alpha_n f(x_n) + (1-\alpha_n) z_n$, where $z_n = y_n - \lambda_n(Ay_n - Aw_n)$. Update:

$$\tau_{n+1} = \begin{cases} \min\left\{\frac{\delta\|w_n - u_n\|}{\|Fw_n - Fu_n\|},\ \tau_n\right\}, & \text{if } Fw_n \neq Fu_n, \\ \tau_n, & \text{otherwise.} \end{cases} \tag{3.2}$$

$$\lambda_{n+1} = \begin{cases} \min\left\{\frac{\mu\|w_n - y_n\|}{\|Aw_n - Ay_n\|},\ \lambda_n\right\}, & \text{if } Aw_n \neq Ay_n, \\ \lambda_n, & \text{otherwise.} \end{cases} \tag{3.3}$$

Stopping criterion: If $y_n = w_n = u_n = x_n$, then stop, otherwise, set $n := n+1$ and go back to **Step 1.**

Remark 3.1. (a) Assumption 3.1 (a) requires that operator A to be monotone and Lipschitz continuous, which is weaker than the inverse-strongly monotonicity assumptions required in [2, 9, 12, 13].

(b) Stepsizes $\{\tau_n\}$ and $\{\lambda_n\}$ given by (3.2) and (3.3) respectively are generated at each iteration by some simple calculations. Thus, $\{\tau_n\}$ and $\{\lambda_n\}$ are easily implemented without the prior knowledge of the operator norm $\|T\|$ and the Lipschitz constant of A respectively.

(c) Step 1 of Algorithm 3.3 is also easily implemented since the value of $\|x_n - x_{n-1}\|$ is a priori known before choosing θ_n. Moreover, we will see in Section 5 that , different choices of the inertial factor $\theta \geq 3$ plays crucial role in the convergence properties of our method.

We next show that the stopping criterion of Algorithm 3.3 is valid.

Lemma 3.1. *If $y_n = w_n = u_n = x_n$ in Algorithm 3.3, then $x_n \in \Gamma$.*

Proof. If $y_n = w_n = u_n = x_n$, then it is clear from (3.1) that $x_n \in (A+B)^{-1}(0)$. Also, we obtain that $x_n = x_n - \tau_n F x_n = x_n - \tau_n T^*(I-S)Tx_n$, which implies that $T^*(I-S)Tx_n = 0$. That is, $STx_n = Tx_n + v$, where $T^*v = 0$. Now, let $z \in \Gamma$, then

$$\begin{aligned} \|Tx_n - Tz\|^2 &= \|Tx_n - Tz\|^2 + 2\langle Tx_n - Tz, v\rangle \\ &= \|STx_n - Tz\|^2 - \|v\|^2 \\ &\leq \|Tx_n - Tz\|^2 - \|v\|^2, \end{aligned}$$

which implies that $\|v\| = 0$. That is, $v = 0$. Hence $STx_n = Tx_n$, which gives that $Tx_n \in Fix(S)$. Thus $x_n \in \Gamma$. □

The following results concerns the stepsizes $\{\tau_n\}$ and $\{\lambda_n\}$ generated by (3.2) and (3.3) respectively.

Lemma 3.2. *The limit of the stepsize* $\{\tau_n\}$ *exists and* $\lim_{n\to\infty} \tau_n > 0$.

Proof. From (3.2), it is obvious that $\tau_{n+1} \leq \tau_n$ for all $n \in \mathbb{N}$. We also know that if S is nonexpansive, then $I-S$ is $\frac{1}{2}$-inverse-strongly monotone. Thus we obtain for all $x, y \in H_1$ that

$$\begin{aligned}||Fx-Fy||^2 &\leq \|T^*\|^2\|(I-S)Tx-(I-S)Ty\|^2\\ &\leq 2\|T\|^2\langle x-y, T^*(I-S)Tx-T^*(I-S)Ty\rangle\\ &\leq 2\|T\|^2\|x-y\|\|T^*(I-S)Tx-T^*(I-S)Ty\|,\end{aligned}$$

which implies that $\|Fx-Fy\| \leq 2\|T\|^2\|x-y\|$. Therefore, F is Lipschitz continuous. Thus, we get in the case of $Fw_n \neq Fu_n$ that

$$\tau_{n+1} = \min\left\{\frac{\delta\|w_n-u_n\|}{\|Fw_n-Fu_n\|}, \tau_n\right\} \geq \min\left\{\frac{\delta}{2\|T\|^2}, \tau_n\right\}.$$

Hence, by induction, we obtain that $\{\tau_n\}$ is bounded below by $\min\left\{\frac{\delta}{2\|T\|^2}, \tau_1\right\}$. Hence, the limit of $\{\tau_n\}$ exists and $\lim_{n\to\infty} \tau_n \geq \min\left\{\tau_1, \frac{\delta}{2\|T\|^2}\right\} > 0$. □

Remark 3.2. Similar to Lemma 3.2, we obtain that the limit of the stepsize $\{\lambda_n\}$ exists and $\lim_{n\to\infty} \lambda_n > 0$.

4. Convergence Analysis

Lemma 4.1. *Let* $\{x_n\}$ *be a sequence generated by Algorithm 3.3 such that Assumption 3.1 and Assumption 3.2 hold. Then* $\{x_n\}$ *is bounded.*

Proof. Let $p \in \Gamma$, then $0 \in (A+B)(p)$. Also, from (3.1), we obtain that

$$Ay_n + \frac{1}{\lambda_n}(w_n - \lambda_n Aw_n - y_n) \in (A+B)y_n.$$

Hence, we obtain from Lemma 2.3 that

$$\langle Ay_n + \frac{1}{\lambda_n}(w_n - \lambda_n Aw_n - y_n), y_n - p\rangle \geq 0,$$

which implies that

$$\langle y_n - w_n - \lambda_n(Ay_n - Aw_n), y_n - p\rangle \leq 0. \tag{4.1}$$

Again, from (3.3), it is clear that

$$||Ay_n - Aw_n|| \leq \frac{\mu}{\lambda_{n+1}}||w_n - y_n|| \ \forall n \in \mathbb{N} \tag{4.2}$$

holds for both $Aw_n = Ay_n$ and $Aw_n \neq Ay_n$. Also, we obtain from Lemma 3.2 that

$$\lim_{n\to\infty}\left(1 - \lambda_n^2\frac{\mu^2}{\lambda_{n+1}^2}\right) = 1 - \mu^2 > 0. \tag{4.3}$$

Thus, there exists $n_0 \in \mathbb{N}$ such that $1-\lambda_n^2 \frac{\mu^2}{\lambda_{n+1}^2} > 0\ \forall n \geq n_0$. Hence, by Step 3, (4.2), and (4.1), we obtain

$$\begin{aligned}
\|z_n - p\|^2 &= \|w_n - p\|^2 + \|y_n - w_n\|^2 + \lambda_n^2 \|Ay_n - Aw_n\|^2 + 2\langle y_n - p, y_n - w_n\rangle \\
&\quad -2\langle y_n - w_n, y_n - w_n\rangle - 2\lambda_n \langle y_n - p, Ay_n - Aw_n\rangle \\
&= \|w_n - p\|^2 - \|y_n - w_n\|^2 + \lambda_n^2 \|Ay_n - Aw_n\|^2 + 2\langle y_n - p, y_n - w_n - \lambda_n(Ay_n - Aw_n)\rangle \\
&\leq \|w_n - p\|^2 - \left(1 - \lambda_n^2 \frac{\mu^2}{\lambda_{n+1}^2}\right) \|y_n - w_n\|^2 + 2\langle y_n - p, y_n - w_n - \lambda_n(Ay_n - Aw_n)\rangle \\
&\leq \|w_n - p\|^2 - \left(1 - \lambda_n^2 \frac{\mu^2}{\lambda_{n+1}^2}\right) \|y_n - w_n\|^2 \qquad (4.4) \\
&\leq \|w_n - p\|^2, \ \forall n \geq n_0. \qquad (4.5)
\end{aligned}$$

Since F is $2||T||$-Lipschitz continuous, it is $\frac{1}{2||T||}$-inverse-strongly monotone. Hence, we obtain from Lemma 2.1 that $I - \tau_n F$ is $\tau_n \|T\|^2$-averaged. That is, $I - \tau_n T^*(I-S)T = (1-\beta_n)I + \beta_n S_n$ for all $n \in \mathbb{N}$, where $\beta_n = \tau_n \|T\|^2$ and S_n is nonexpansive for all $n \in \mathbb{N}$. Therefore, we can rewrite w_n from (3.1) as

$$w_n = (1-\beta_n)u_n + \beta_n S_n u_n, \ n \geq 1. \qquad (4.6)$$

Hence, we obtain that

$$\begin{aligned}
\|w_n - p\|^2 &\leq (1-\beta_n)\|u_n - p\|^2 + \beta_n \|S_n u_n - p\|^2 - \beta_n(1-\beta_n)\|u_n - S_n u_n\|^2 \\
&\leq \|u_n - p\|^2 - \beta_n(1-\beta_n)\|u_n - S_n u_n\|^2 \qquad (4.7) \\
&\leq \|u_n - p\|^2.
\end{aligned}$$

Observe from Step 1 and Assumption 3.2 that $\theta_n ||x_n - x_{n-1}|| \leq \varepsilon_n$ for all $n \in \mathbb{N}$, which implies that

$$\frac{\theta_n}{\alpha_n} ||x_n - x_{n-1}|| \leq \frac{\varepsilon_n}{\alpha_n} \to 0, \text{as } n \to \infty. \qquad (4.8)$$

Hence, there exists $M > 0$ such that $\frac{\theta_n}{\alpha_n} ||x_n - x_{n-1}|| \leq M$ for all $n \in \mathbb{N}$. Thus, we obtain from Step 2 that

$$\begin{aligned}
\|u_n - p\| &\leq ||x_n - p|| + \theta_n ||x_n - x_{n-1}|| \\
&= ||x_n - p|| + \alpha_n \frac{\theta_n}{\alpha_n} ||x_n - x_{n-1}|| \\
&\leq ||x_n - p|| + \alpha_n M, \ \forall n \in \mathbb{N}. \qquad (4.9)
\end{aligned}$$

Combining (4.5), (4.7), and (4.9), we obtain that

$$\begin{aligned}
\|x_{n+1} - p\| &\leq \alpha_n \|f(x_n) - p\| + (1-\alpha_n)\|z_n - p\| \\
&\leq \alpha_n \rho \|x_n - p\| + \alpha_n \|f(p) - p\| + (1-\alpha_n)\|z_n - p\| \\
&\leq (1 - \alpha_n(1-\rho))\|x_n - p\| + \alpha_n \left[M + \|f(p) - p\|\right], \ \forall n \geq n_0,
\end{aligned}$$

which implies by Lemma 2.5 that $\{x_n\}$ is bounded. Consequently, $\{w_n\}$, $\{u_n\}$, $\{y_n\}$, $\{z_n\}$, and $\{f(x_n)\}$ are all bounded. $\square$

Lemma 4.2. *Let $\{x_n\}$ be a sequence generated by Algorithm 3.3 such that Assumption 3.1 and Assumption 3.2 hold. If there exists a subsequence $\{x_{n_k}\}$ of $\{x_n\}$ convergent weakly to a point $z \in H_1$ and $\lim_{n\to\infty} \|w_{n_k} - y_{n_k}\| = 0 = \lim_{n\to\infty} \|w_{n_k} - x_{n_k}\|$, then $z \in \Gamma$.*

Proof. By Step 2 and (4.8), we obtain that

$$||u_n - x_n|| = \alpha_n \frac{\theta_n}{\alpha_n} ||x_n - x_{n-1}|| \to 0, \text{ as } n \to \infty. \tag{4.10}$$

Thus, we obtain from the hypothesis that

$$\lim_{k\to\infty} ||u_{n_k} - w_{n_k}|| = 0. \tag{4.11}$$

Now, let the subsequence $\{x_{n_k}\}$ of $\{x_n\}$ be weakly convergent to a point $z \in H_1$. Then, it follows that the subsequences $\{w_{n_k}\}$, $\{y_{n_k}\}$, and $\{u_{n_k}\}$ are also weakly convergent to $z \in H_1$. Also, by Lemma 3.2, we obtain that $\lim_{n\to\infty} \tau_n = \tau > 0$. Furthermore, since $F \equiv T^*(I-S)T$ is Lipschitz continuous, we have that $\{T^*(I-S)Tu_{n_k}\}$ is bounded. Hence,

$$\|(I - \tau_{n_k} T^*(I-S)T)u_{n_k} - (I - \tau T^*(I-S)T)u_{n_k}\| = |\tau_{n_k} - \tau| \|T^*(I-S)Tu_{n_k}\| \to 0$$

as $k \to \infty$. That is, $\lim_{k\to\infty} \|w_{n_k} - (I - \tau T^*(I-S)T)u_{n_k}\| = 0$, which implies from (4.11) that $\lim_{k\to\infty} \|u_{n_k} - (I - \tau T^*(I-S)T)u_{n_k}\| = 0$. Thus, by Lemma 2.2, we obtain that $z \in Fix(I - \tau T^*(I - S)T)$. Hence, using the same line of argument as in the proof of Lemma 3.1, we obtain that $Tz \in Fix(S)$. Now, let $(v, w) \in G(A+B)$. Then $w - Av \in B(v)$. Also, we obtain from (3.1) that $\frac{1}{\lambda_{n_k}}(w_{n_k} - \lambda_{n_k} A w_{n_k} - y_{n_k}) \in B(y_{n_k})$. Thus, we have from the monotonicity of B that

$$\langle v - y_{n_k}, w - Av - \frac{1}{\lambda_{n_k}}(w_{n_k} - \lambda_{n_k} A w_{n_k} - y_{n_k}) \rangle \geq 0,$$

which together with the monotonicity of A yields

$$\begin{aligned} \langle v - y_{n_k}, w \rangle &\geq \langle v - y_{n_k}, Av - A(y_{n_k}) \rangle + \langle v - y_{n_k}, A(y_{n_k}) - A(w_{n_k}) \rangle \\ &\quad + \langle v - y_{n_k}, \frac{1}{\lambda_{n_k}}(w_{n_k} - y_{n_k}) \rangle \\ &\geq \langle v - y_{n_k}, A(y_{n_k}) - A(w_{n_k}) \rangle + \langle v - y_{n_k}, \frac{1}{\lambda_{n_k}}(w_{n_k} - y_{n_k}) \rangle. \end{aligned} \tag{4.12}$$

Passing limit as $k \to \infty$ in (4.12), we obtain $\langle v - z, w \rangle \geq 0$. Also, by Lemma 2.3, $A + B$ is maximal monotone. Thus we obtain that $0 \in (A+B)z$. This gives that $z \in \Gamma$. □

We now present the main theorem of this paper.

Theorem 4.1. *Let $\{x_n\}$ be a sequence generated by Algorithm 3.3 such that Assumption 3.1 and Assumption 3.2 hold. Then, $\{x_n\}$ converges strongly to $z^* = P_\Gamma f(z^*)$.*

Proof. Let $z^* = P_\Gamma f(z^*)$. From (4.7), we obtain

$$\begin{aligned} \|w_n - z^*\|^2 &\leq ||x_n - z^*||^2 + 2\theta_n \langle x_n - z^*, x_n - x_{n-1} \rangle + \theta_n^2 ||x_n - x_{n-1}||^2 \\ &\leq ||x_n - z^*||^2 + \theta_n ||x_n - x_{n-1}|| \left[2||x_n - z^*|| + \theta_n ||x_n - x_{n-1}||\right] \\ &\leq ||x_n - z^*||^2 + 3\theta_n ||x_n - x_{n-1}|| \bar{M}, \end{aligned} \tag{4.13}$$

for some $\bar{M} > 0$. By Step 3, (4.4), (4.13), (4.5), (4.7), and (4.9), we obtain that

$$\begin{aligned}
&\|x_{n+1} - z^*\|^2 \\
&\leq (1-\alpha_n)^2 \|z_n - z^*\|^2 + 2\alpha_n \langle f(x_n) - z^*, x_{n+1} - z^* \rangle \\
&\leq (1-\alpha_n)^2 \left[||x_n - z^*||^2 + 3\theta_n ||x_n - x_{n-1}|| \bar{M} \right] - (1-\alpha_n)^2 \left(1 - \lambda_n^2 \frac{\mu^2}{\lambda_{n+1}^2} \right) ||y_n - w_n||^2 \\
&\quad + 2\alpha_n \langle f(x_n) - z^*, x_{n+1} - z^* \rangle \\
&\leq (1-\alpha_n)^2 \left[||x_n - z^*||^2 + 3\theta_n ||x_n - x_{n-1}|| \bar{M} \right] - (1-\alpha_n)^2 \left(1 - \lambda_n^2 \frac{\mu^2}{\lambda_{n+1}^2} \right) ||y_n - w_n||^2 \\
&\quad + 2\alpha_n \rho \|x_n - z^*\| \|x_{n+1} - z^*\| + 2\alpha_n \langle f(z^*) - z^*, x_{n+1} - z^* \rangle \\
&\leq (1-\alpha_n)^2 \left[||x_n - z^*||^2 + 3\theta_n ||x_n - x_{n-1}|| \bar{M} \right] - (1-\alpha_n)^2 \left(1 - \lambda_n^2 \frac{\mu^2}{\lambda_{n+1}^2} \right) ||y_n - w_n||^2 \\
&\quad + 2\alpha_n \rho \|x_n - z^*\| \left[||x_n - z^*|| + \alpha_n \left(||f(x_n) - z^*|| + M \right) \right] + 2\alpha_n \langle f(z^*) - z^*, x_{n+1} - z^* \rangle \\
&\leq (1-\alpha_n)^2 \left[||x_n - z^*||^2 + 3\theta_n ||x_n - x_{n-1}|| \bar{M} \right] - (1-\alpha_n)^2 \left(1 - \lambda_n^2 \frac{\mu^2}{\lambda_{n+1}^2} \right) ||y_n - w_n||^2 \\
&\quad + 2\alpha_n \rho \|x_n - z^*\|^2 + \alpha_n^2 K + 2\alpha_n \langle f(z^*) - z^*, x_{n+1} - z^* \rangle, \ \forall n \geq n_0,
\end{aligned} \tag{4.14}$$

for some $K > 0$. Hence, we obtain from (4.14) that

$$\begin{aligned}
\|x_{n+1} - z^*\|^2 &\leq (1 - 2\alpha_n(1-\rho)) \|x_n - z^*\|^2 + \alpha_n^2 \|x_n - z^*\|^2 + 3\alpha_n M \bar{M} \\
&\quad - (1-\alpha_n)^2 \left(1 - \lambda_n^2 \frac{\mu^2}{\lambda_{n+1}^2} \right) \|y_n - w_n\|^2 + \alpha_n^2 K + 2\alpha_n \langle f(z^*) - z^*, x_{n+1} - z^* \rangle \\
&\leq (1 - 2\alpha_n(1-\rho)) \|x_n - z^*\|^2 + \alpha_n v_n, \ \forall n \geq n_0,
\end{aligned} \tag{4.15}$$

where $v_n := \alpha_n \left[\|x_n - z^*\|^2 + 3M\bar{M} + K \right] + 2\langle f(z^*) - z^*, x_{n+1} - z^* \rangle$. To show that $\{x_n\}$ converges strongly to z^*, we apply Lemma 2.4. That is, we show that $\limsup_{k\to\infty} v_{n_k} \leq 0$ for every subsequence $\{\|x_{n_k} - z^*\|\}$ of $\{\|x_n - z^*\|\}$ satisfying

$$\liminf_{k\to\infty} \left(\|x_{n_k+1} - z^*\| - \|x_{n_k} - z^*\| \right) \geq 0. \tag{4.16}$$

Now, suppose that $\{\|x_{n_k} - z^*\|\}$ is a subsequence of $\{\|x_n - z^*\|\}$ such that (4.16) holds. Then,

$$\begin{aligned}
& \liminf_{k\to\infty} \left(\|x_{n_k+1} - z^*\|^2 - \|x_{n_k} - z^*\|^2 \right) \\
= \ & \liminf_{k\to\infty} \left[\left(\|x_{n_k+1} - z^*\| - \|x_{n_k} - z^*\| \right) \left(\|x_{n_k+1} - z^*\| + \|x_{n_k} - z^*\| \right) \right] \geq 0.
\end{aligned}$$

Hence, we obtain from (4.15) and Assumption 3.2 that

$$\begin{aligned}
&\limsup_{k\to\infty} \left[(1-\alpha_{n_k})^2 \left(1 - \lambda_{n_k}^2 \frac{\mu^2}{\lambda_{n_k+1}^2} \right) ||y_{n_k} - w_{n_k}||^2 \right] \\
&\leq \limsup_{k\to\infty} \left[||x_{n_k} - z^*||^2 - ||x_{n_k+1} - z^*||^2 \right] = -\liminf_{k\to\infty} \left[||x_{n_k+1} - z^*||^2 - ||x_{n_k} - z^*||^2 \right] \leq 0,
\end{aligned}$$

which from (4.3) gives that

$$\lim_{k\to\infty} ||y_{n_k} - w_{n_k}|| = 0. \tag{4.17}$$

From (4.7) and (4.13), we obtain that

$$\begin{aligned}\beta_{n_k}(1-\beta_{n_k})\|u_{n_k} - S_{n_k}u_{n_k}\| &\le \|u_{n_k} - z^*\|^2 - \|w_{n_k} - z^*\|^2 \\ &\le \|x_{n_k} - z^*\|^2 + 3\alpha_{n_k}M\bar{M} - \|w_{n_k} - z^*\|^2. \end{aligned} \tag{4.18}$$

We also have $-\|w_{n_k} - z^*\|^2 \le -\|x_{n_k+1} - z^*\|^2 + 2\alpha_{n_k}\langle f(x_{n_k}) - z^*, x_{n_k+1} - z^*\rangle$, which implies from (4.18) that

$$\begin{aligned}\limsup_{k\to\infty}\beta_{n_k}(1-\beta_{n_k})\|u_{n_k} - S_{n_k}u_{n_k}\| &\le \limsup_{k\to\infty}\left[||x_{n_k} - z^*||^2 + 3\alpha_{n_k}M\bar{M} - ||x_{n_k+1} - z^*||^2\right] \\ &\quad + \limsup_{k\to\infty} 2\alpha_{n_k}\langle f(x_{n_k}) - z^*, x_{n_k+1} - z^*\rangle \\ &= -\liminf_{k\to\infty}\left[||x_{n_k+1} - z^*||^2 - ||x_{n_k} - z^*||^2\right] \le 0,\end{aligned}$$

which implies that $\lim_{k\to\infty} ||Su_{n_k} - u_{n_k}|| = 0$. Thus, we obtain from (4.6) that $\lim_{k\to\infty} ||w_{n_k} - u_{n_k}|| = 0$. Hence, we obtain from (4.10) that $\lim_{k\to\infty} ||w_{n_k} - x_{n_k}|| = 0$. Observe also from Step 3 and (4.17) that $||z_{n_k} - y_{n_k}|| = ||\lambda_{n_k}(Ay_{n_k} - Aw_{n_k})|| \le \lambda_{n_k}L||w_{n_k} - y_{n_k}|| \to 0$ as $k \to \infty$. Thus, we obtain from (4.17) that $\lim_{k\to\infty} ||z_{n_k} - w_{n_k}|| = 0$. Also, we have that $||x_{n_k+1} - z_{n_k}|| = \alpha_{n_k}||f(x_{n_k}) - z_{n_k}|| \to 0$ as $k \to \infty$. It follows that $\lim_{n\to\infty} \|x_{n_k+1} - x_{n_k}\| = 0$. Again, by Lemma 4.1, we can chose a subsequence $\{x_{n_{k_j}}\}$ of $\{x_{n_k}\}$ such that $\{x_{n_{k_j}}\}$ converges weakly to $z \in H_1$ and

$$\limsup_{k\to\infty}\langle f(z^*) - z^*, x_{n_k} - z^*\rangle = \lim_{j\to\infty}\langle f(z^*) - z^*, x_{n_{k_j}} - z^*\rangle = \langle f(z^*) - z^*, z - z^*\rangle.$$

Also, we obtain from (4.17) and Lemma 4.2 that $z \in \Gamma$. Since $z^* = P_\Gamma f(z^*)$, we obtain from the previous equality that $\limsup_{k\to\infty}\langle f(z^*) - z^*, x_{n_k} - z^*\rangle \le 0$, which implies that $\limsup_{k\to\infty}\langle f(z^*) - z^*, x_{n_k+1} - z^*\rangle \le 0$. Thus, $\limsup_{k\to\infty} v_{n_k} \le 0$. Hence from Lemma 2.4, we obtain that $\{x_n\}$ converges strongly to z^*. □

Remark 4.1. Note that from the characterization of metric projections, we have that

$$z^* = P_\Gamma f(z^*) \Longleftrightarrow \langle f(z^*) - z^*, z^* - z\rangle \ge 0 \ \forall z \in \Gamma. \tag{4.19}$$

Therefore, one advantage of adopting a viscosity-type algorithm is that it also converges strongly to a solution to variational inequality (4.19). Moreover, viscosity-type algorithms have higher rate of convergence than their Halpern-type counterpart (see [13]). More so, it has been established (see [13, Remark 3.7]) that Halpern-type iterations imply viscosity iterations. In fact, setting $f(x) = u$ for all $x \in H_1$ in Algorithm 3.3, we derive a new Halpern-type method for solving the GSFP (1.14) as a corollary of Theorem 4.1.

Remark 4.2. Recall that, if we set $B = N_C$ in (1.14), we recover problem (1.9) as a special case of problem (1.14). Recall also that $y^* \in Q$ is a solution to classical VIP (1.7) if and only if $y^* \in Fix(P_Q(I - \lambda g))$. We know that $P_Q(I - \lambda g)$ (where g is α_2-inverse-strongly monotone and $\lambda \in (0, 2\alpha_2)$) is an averaged-nonexpansive mapping, which is a special type of nonexpansive mappings. Thus, by setting $B = N_C$ and $S = P_Q(I - \lambda g)$, we recover SVIP (1.6)-(1.7) as a special case of problem (1.14). Therefore, we can set $B = N_C$ in Algorithm 3.3 (since N_C is maximal monotone) to obtain that $(I+\lambda N_C)^{-1}(I-\lambda A) = P_C(I-\lambda A)$. Hence, Algorithm 3.3 with $B = N_C$ can be used to solve problem (1.9) while Algorithm 3.3 with $B = N_C$ and $S = P_Q(I-\lambda g)$ can be used to solve SVIP (1.6)-(1.7) as corollaries of Theorem 4.1.

Remark 4.3. Similar to Remark 4.2, we know that $(I+\lambda M)^{-1}(I-\lambda g)$ (where M is maximal monotone and g is α_2-inverse strongly monotone with $\lambda \in (0,2\alpha_2)$) is an averaged-nonexpansive mapping, which is a special type of nonexpansive mappings. Thus, by setting $S = (I+\lambda M)^{-1}(I-\lambda g)$, we recover the SMVIP (1.11)-(1.12) as a special case of problem (1.14). Therefore, Algorithm 3.3 with $S = (I+\lambda M)^{-1}(I-\lambda g)$ can be used to solve the SMVIP (1.11)-(1.12) as a corollary of Theorem 4.1.

Remark 4.4. By setting $A \equiv 0$ in (1.14), we recover problem (1.3) as a special case of problem (1.14). Therefore, Algorithm 3.3 with $A \equiv 0$ can be used to solve the GSFP (1.3) as a corollary of Theorem 4.1. Thus Algorithm 3.3 can be reduced to new self-adaptive stepsize methods for solving SCNPP (1.2) and SFP (1.1).

Remark 4.5. If $H_1 = H_2 = H$ and $T = I = S$ in the setting of Remarks 4.2-4.4, we recover methods for solving the classical variational inequality problem (see [30]), monotone variational inclusion problem (see [12, 31]), and null point problem (see [28]).

5. Applications and Numerical Analysis

In this section, we study some applications of our convergence results to optimal control problems, split linear inverse problems, and least absolute selection and shrinkage operator problems via some numerical computations and analysis.

5.1. **Optimal control problem.** Let $H_1 = H_2 = L_2([0,T],\mathbb{R}^m)$ be the Hilbert space of square integrable and measurable vector functions u from $[0,T]$ into $\mathbb{R}^m$, where $m \in \mathbb{N}$ and $0 < T \in \mathbb{R}$, with inner product $\langle u,v\rangle = \sum_{i=1}^m \int_0^T u_i(t)v_i(t)dt$ and norm $||u|| = \sqrt{\langle u,u\rangle}$.

Let U be the set of admissible controls in the form of an m-dimensional box which consists of piecewise continuous functions, defined as: $U = \{u(t) \in L_2([0,T],\mathbb{R}^m) : u_i(t) \in [u_i^-, u_i^+],\ i = 1,2\dots,m\}$. Consider the following optimal control problem:

$$u^*(t) = \arg\min\{J(u) : u \in U\}, \tag{5.1}$$

where the terminal objective function J is defined by $J(u) = \Phi(x(T))$ with Φ being a convex and differentiable function defined on the attainability set. If such a control exists. Then, for each control $u(t) \in U$, we suppose that $x(t) \in L_2([0,T],\mathbb{R}^m)$ (also known as the state variable) satisfies the constraints in the form of a system of linear differential equation:

$$\dot{x}(t) = \mathbb{A}(t)x(t) + \mathbb{B}(t)u(t),\ \ x(0) = x_0,\ t \in [0,T],$$

where $\mathbb{A}(t) \in \mathbb{R}^{n\times n}$, $\mathbb{B}(t) \in \mathbb{R}^{n\times m}$ are given continuous matrices for every $t \in [0,T]$. From the Pontryagin's maximum principle (see [32]), we see that, for any optimal pair (x^*,u^*), there exists a function (the adjoint) $p^* : [0,T] \to \mathbb{R}^m$ such that (x^*,p^*,u^*) solves the following systems:

$$\begin{cases} \dot{x}^*(t) = \mathbb{A}(t)x^*(t) + \mathbb{B}(t)u^*(t) \\ x^*(0) = x_0, \end{cases} \tag{5.2}$$

$$\begin{cases} \dot{p}^*(t) = -\mathbb{A}(t)^{\mathbb{T}} p^*(t) \\ p^*(T) = \nabla\Phi(x(T)), \end{cases} \tag{5.3}$$

$$0 \in \mathbb{B}(t)^{\mathbb{T}} p^*(t) + N_U u^*(t), \tag{5.4}$$

where $N_U(u)$ is the normal cone of U at u, and it is maximal monotone. Putting $Au(t) := \mathbb{B}(t)^{\mathbb{T}} p(t)$, we see that Au is the gradient of the objective function J, which is monotone and Lipschitz continuous (see [33]). Thus (5.4) can be rewritten as $0 \in Au^* + N_U u^*$. Also, we can define the bounded linear mapping $T : L_2([0,1]) \to L_2([0,1])$ by $Tx(s) = \int_0^1 \tilde{K}(s,t)x(t)dt \ \forall x \in L_2([0,1])$, where $\tilde{K}$ is a continuous real-valued function defined on $[0,1]\times[0,1]$. Then, the adjoint of T is

$$T^*x(s) = \int_0^1 \tilde{K}(t,s)x(t)dt \ \forall x \in L_2([0,1]).$$

We can also define the nonexpansive mapping $S : L_2([0,1]) \to L_2([0,1])$ by $Sx(t) = \int_0^1 tx(s)ds$ forall $t \in [0,1]$. Indeed, S is nonexpansive. To see this, by defining $x,y \in L_2([0,1])$, one has

$$|Sx(t) - Sy(t)|^2 \le \left(\int_0^1 t|x(s)-y(s)|ds\right)^2 \le \int_0^1 |x(s)-y(s)|^2 ds = ||x-y||^2.$$

Thus $||Sx - Sy|| \le ||x-y||^2$. Hence, we can apply Algorithm 3.3 with $B = N_U$ to solve the problem (1.14) governed by optimal control problem (5.1).

In order to ensure the implementation of the algorithm, we discretize the continuous functions. We define the mesh size $h := \frac{\mathbb{T}}{N}$, where N is a natural number. We identity any discretized control $u^N = (u_0, u_1, \cdots, u_N)$ with its piece-wise constant extension $u^N(t) = u_i$ for all $t \in [t_i, t_{i+1})$, $i = 0,1,\cdots,N$, where $t_i = ih$, $i = 0,1,\cdots,N$. Furthermore, we identity any discretized state variable $x^N := (x_0, x_1, \cdots, x_N)$ with its piece-wise linear interpolation $x^N(t) = x_i + \frac{t-t_i}{h}(x_{i+1} - x_i)$, for $t \in [t_i, t_{i+1})$, $i = 0,1,\cdots,N-1$. Similarly, for the adjoint variable, $p^N := (p_0, p_1, \cdots, p_N)$ and thus, $Au^N = \left(\mathbb{B}^{\mathbb{T}}(t_0)p_0, \mathbb{B}^{\mathbb{T}}(t_1)p_1, \cdots, \mathbb{B}^{\mathbb{T}}(t_N)p_N\right)^{\mathbb{T}}$. We now consider the Euler discretization technique (see [34]) to discretize the systems of ODEs (5.2) and (5.3). That is, at each iteration, the system of ODEs (5.2) and (5.3) is solved by the Euler method:

$$\begin{cases} x_{i+1} = x_i + h[\mathbb{A}(t_i)x_i + \mathbb{B}(t_i)u_i] \\ x(0) = x_0, \end{cases} \quad \textit{and} \quad \begin{cases} p_i = p_{i+1} + h\mathbb{A}(t_i)^{\mathbb{T}} p_{i+1} \\ p_N = \nabla\Phi(x_N), \end{cases}$$

respectively. It is known that, for the Euler discretization, the difference between the discretized solution $u^N(t)$ and the original solution $u^*(t)$ is proportional to the mesh size h. That is, there exists a constant $C > 0$ such that $\|u^N - u^*\| \le Ch$. We now consider the following typical example taken from [35, Example 1.2]:

$$\begin{array}{ll} \text{minimize} & x_1(1) \\ \text{subject to} & \dot{x}_j(t) = s_j x_{j+1} + u(t), \ \ s_j = -2(m-j+1), \ \ j = 1,2,\cdots,m, \\ & \dot{x}_{m+1}(t) = u(t), \ \ t \in [0,1], \\ & x(0) = 0, \\ & u(t) \in [-1,1], \end{array}$$

where m is a natural number. Note that $\mathbb{A}(t)$ and $\mathbb{B}(t)$ are of the form

$$\mathbb{A}(t)=\begin{pmatrix} 0 & s_1 & 0 & \cdots & 0 & 0\\ 0 & 0 & s_2 & \cdots & 0 & 0\\ 0 & 0 & 0 & \cdots & 0 & 0\\ - & - & - & \cdots & - & -\\ 0 & 0 & 0 & \cdots & 0 & s_m\\ 0 & 0 & 0 & \cdots & 0 & 0 \end{pmatrix}_{m+1,m+1} \quad \text{and} \quad \mathbb{B}(t)=\begin{pmatrix} 1\\ 1\\ 1\\ \vdots\\ 1\\ 1 \end{pmatrix}_{m+1,1}.$$

We choose $N=100$ and the starting point $u_{-1}(t)=u_0(t)$ randomly in U. We also randomly choose $\lambda_1,\tau_1>0$ and $\mu,\delta\in(0,1)$. We choose $\alpha_n=\frac{1}{5n+2}$ and $\theta_n=\bar{\theta}_n$ with different choices of $\theta=3,6,10$ and $\varepsilon_n=\frac{\alpha_n}{n^{0.01}}$. Note that, in our method, the stepsizes $\{\lambda_n\}$ and $\{\tau_n\}$ are generated in each iteration.

For the numerical analysis in this subsection, we compared our method with Algorithm (1.10). For this algorithm, we take $\lambda_n=\frac{1}{2L}$, where L is the Lipschitz constant of the operator A. The numerical results are displayed in Figure 1 and Table 1, with $m=1,2,3,4$.

Table 1. Numerical results for the optimal control problem.

m		Alg. 3.3 ($\theta=3$)	Alg. 3.3 ($\theta=6$)	Alg. 3.3 ($\theta=10$)	Alg. (1.10)
1	CPU time (sec)	1.337	0.486	0.913	44.091
	Iterations	18	12	16	64
2	CPU time (sec)	1.304	0.221	0.829	33.371
	Iterations	18	11	12	52
3	CPU time (sec)	2.835	0.517	1.329	63.112
	Iterations	26	12	19	72
4	CPU time (sec)	1.765	0.413	0.501	15.501
	Iterations	18	11	11	46

5.2. **Split linear inverse problem.** As assumed in [4], let $G:H_1\to\mathbb{R}$ be a convex and continuous function, which possibly is nonsmooth and $P:H_1\to\mathbb{R}$ be a Fréchet differentiable function with a Lipschitz continuous gradient ∇P of P. We now consider the following class of Split Linear Inverse Problem (SLIP) in real Hilbert spaces (see, for example [4], for the case of inverse linear problems in $\mathbb{R}^N$):

$$\text{Find } x^*\in H_1 \text{ such that } P(x^*)+G(x^*)=\min_{x\in H_1}[P(x)+G(x)], \text{ and } Tx^*\in Fix(S), \tag{5.5}$$

where $S:H_2\to H_2$ is any nonlinear mapping. Let the solution set of problem (5.5) be denoted by Υ, and assume that it is nonempty, that is, consistent. According to [20], the subdifferential ∂G of G is maximally monotone. Moreover,

$$P(x^*)+G(x^*)=\min_{x\in H_1}[P(x)+G(x)]\Leftrightarrow 0\in\nabla P(x^*)+\partial G(x^*).$$

Thus, we can apply Algorithm 3.3 with $A=\nabla P$ and $B=\partial G$ to solve the SLIP (5.5).

5.3. **Least absolute selection and shrinkage operator (LASSO) problem.** It is important to note that SLIP (5.5) contains the LASSO problem as a special case. In this case, $P(x)=\frac{1}{2}||Dx-b||_2^2$

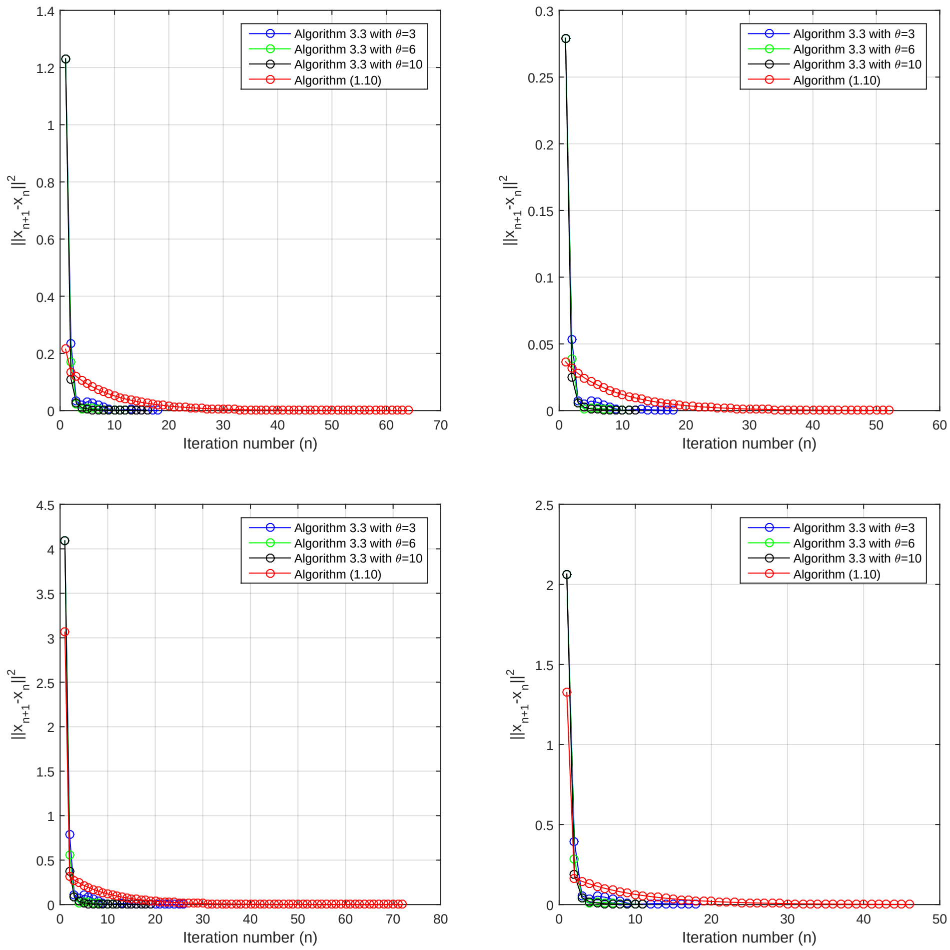

FIGURE 1. Error ($||x_{n-1}-x_n||^2$) vs iteration numbers (n) for the optimal control problem: Top Left: $m=1$; Top Right: $m=2$; Bottom Left: $m=3$; Bottom Right: $m=4$.

and $G(x)=\lambda||x||_1$. That is, the LASSO problem can be related to:

$$\min_{x\in\mathbb{R}^N}\left\{\frac{1}{2}||Dx-b||_2^2+\lambda||x||_1\right\}, \tag{5.6}$$

where $\lambda>0$, $b\in\mathbb{R}^M$, and $D:\mathbb{R}^N\to\mathbb{R}^M$ is an operator. Note that, solving LASSO problem (5.6) is equivalent to solving the following underdetermined linear equation systems (see [36]):

$$Dx=b \ \text{ (linear equations, where } N\gg M). \tag{5.7}$$

Compressed sensing is then applied for finding solutions of problem (5.7) (and hence, solutions of problem (5.6)). In this case that the number of unknowns is greater than the number of equations, system (5.7) generates solutions or no solution. In such situation, the least square method (the method of finding the minimum l_2-norm solution) is then applied to the linear system. In most real-world applications, problem (5.6) can be computed to recover x when x is sparse. Moreover, problem (5.6) can be reformulated as a second-order cone programming problem and can be seen as the pivot for proposing two very important methods; namely, the Iteration Shrinkage Thresholding Algorithm (ISTA) and Fast Iteration Shrinkage Thresholding Algorithm (FISTA) (see [4]), which are very efficient for solving SLIPs.

To apply Algorithm 3.3 to solve LASSO problem (5.6), we set $A = \nabla P$, the gradient of P, where $P(x) = \frac{1}{2}||Dx - b||_2^2$ and $B = \partial G$, the subdifferential of G, where $G(x) = \lambda ||x||_1$. We know that ∇P is $||D||^2$-Lipschitz continuous and monotone, while ∂G is maximally monotone (see [20]). Therefore, Algorithm 3.3 reduces to the following.

Algorithm 5.1
Step 0: Choose sequences $\{\alpha_n\}_{n=1}^{\infty}$ and $\{\varepsilon_n\}_{n=1}^{\infty}$ such that the conditions from Assumption 3.2 hold and let $\tau_1, \lambda_1 > 0,\ \mu, \delta \in (0,1), \theta \geq 3$, and $x_0, x_1 \in H_1$ be given arbitrarily. Set $n := 1$.
Step 1. Given the iterates x_{n-1} and x_n $(n \geq 1)$, choose the sequence $\{\theta_n\}$ such that $0 \leq \theta_n \leq \bar{\theta}_n$, where

$$\bar{\theta}_n := \begin{cases} \min\left\{\frac{n-1}{n+\theta-1}, \frac{\varepsilon_n}{||x_n - x_{n-1}||}\right\} & \text{if } x_n \neq x_{n-1}, \\ \frac{n-1}{n+\theta-1} & \text{otherwise.} \end{cases}$$

Step 2. Set $u_n = x_n + \theta_n(x_n - x_{n-1})$. Compute $w_n = u_n - \tau_n F u_n$ and $y_n = (I + \lambda_n \partial G)^{-1}(I - \lambda_n \nabla P) w_n$.
Step 3. Compute $x_{n+1} = \alpha_n f(x_n) + (1 - \alpha_n) z_n$, where $z_n = y_n - \lambda_n (\nabla P y_n - \nabla P w_n)$.
Update:

$$\tau_{n+1} = \begin{cases} \min\left\{\frac{\delta ||w_n - u_n||}{||F w_n - F u_n||},\ \tau_n\right\}, & \text{if } F w_n \neq F u_n, \\ \tau_n, & \text{otherwise.} \end{cases}$$

and

$$\lambda_{n+1} = \begin{cases} \min\left\{\frac{\mu ||w_n - y_n||}{||\nabla P w_n - \nabla P y_n||},\ \lambda_n\right\}, & \text{if } \nabla P w_n \neq \nabla P y_n, \\ \lambda_n, & \text{otherwise.} \end{cases}$$

Stopping criterion: If $y_n = w_n = u_n = x_n$, then stop, otherwise, set $n := n+1$ and go back to **Step 1.**

In this case, we have that $(I + \lambda \partial G)^{-1}(x) = \text{prox}_G(x) = \arg\min_u \lambda ||x||_1 + \frac{1}{2}||u - x||_2^2$, which is separable in indices. Thus, for $x \in \mathbb{R}^N$, we obtain

$$\begin{aligned} (I + \lambda \partial G)^{-1}(x) &= \text{prox}_{\lambda ||.||_1}(x) \\ &= \left(\text{prox}_{\lambda |.|_1}(x_1), \cdots, \text{prox}_{\lambda |.|_1}(x_N)\right) \\ &= \left(\text{sgn}(x_1) \max\{|x_1| - \lambda, 0\}, \cdots, \text{sgn}(x_N) \max\{|x_N| - \lambda, 0\}\right). \end{aligned}$$

For the numerical analysis of this subsection, we compare Algorithm 5.1 with Algorithm 6.1 of Shehu *et al.* [36]. Note that, in [36], Algorithm 6.1 of [36] was compared with the Proximal-Gradient Method (PGM) of [37] and Algorithms 3.3 and 3.5 of Wang and Xu [38] for solving the LASSO problem (5.6). It was shown there that Algorithm 6.1 of [36] outperforms the PGM of [37] and Algorithms 3.3 and 3.5 of Wang and Xu [38] in both CPU time and number of iterations. Therefore, our Algorithm 5.1 do not only outperforms Algorithm 6.1 of Shehu *et al.* [36] but also performs better than the PGM of [37] and the Algorithms 3.3 and 3.5 of Wang and Xu [38] in both CPU time and number of iterations as reported in Table 2 below. For the comparison, we consider the same choices as in [36]. That is, we take $x_0 = \mathbf{0}$. Furthermore, we denote D_i, $i = 1,2,\dots,M$ by the row of D. Thus D_i represents the ith observation of the independent variable with b_i been the response variable while $x \in \mathbb{R}^N$ is the regression coefficient to be recovered. As in [36], we consider different values of M and N such that $N \gg M$. Also, we randomly generate the data b as $Dx + \alpha e$, where $\alpha = 0.01$, x is a generated sparse vector while D and e are random matrices whose entries are normally distributed with zero mean and variance 1. Set the stopping criterion as $||x_n - x^*||_2 \le \varepsilon$, where $\varepsilon = 10^{-3}$ and x^* is obtained using SPGL1 (see [36]).

Table 2. Numerical Results for the LASSO Problem.

(N, M)		Alg. 5.1 ($\theta = 3$)	Alg. 5.1 ($\theta = 6$)	Alg. 5.1 ($\theta = 10$)	Alg. 6.1 of [36]
$(12, 0.20)$	CPU time (sec)	7.001	5.102	6.851	14.137
	Iterations	6	3	5	9
$(12, 0.25)$	CPU time (sec)	8.321	5.701	7.013	14.727
	Iterations	6	3	5	9
$(12, 0.30)$	CPU time (sec)	8.407	6.213	7.351	15.573
	Iterations	6	3	5	9
$(12, 0.35)$	CPU time (sec)	8.011	6.832	6.201	15.311
	Iterations	6	3	4	9

In LASSO problem (5.6), the gradient ∇P, where $P(x) = ||Dx - b||_2^2$, may generally not be inverse-strongly monotone. In fact, even in the case that ∇P is $\frac{1}{||D||^2}$-inverse-strongly monotone, the iterative methods with the stepsizes that depends on the operator norm or the knowledge of the coefficient of the underlying operator (in this case $||D||$) may still not be applicable to problem (5.6) since the computation of the constant $\frac{1}{||D||^2}$ is generally a very difficult task to accomplish or impossible to calculate (see [39, Theorem 2.3]). Therefore, iterative methods without these limitations (e.g., Algorithm 3.3 and Algorithm 5.1) seems to have more real-world applications.

6. Conclusion

A class of generalized split feasibility problems over the solution set of monotone variational inclusion problems was considered in real Hilbert spaces. To solve this problem, a new self-adaptive stepsize method was proposed by combing a viscosity method with inertial extrapolation techniques. This method was proved to converge strongly to a solution of the generalized split feasibility problem in the framework of two real Hilbert spaces. The strong convergent result was obtained under some relaxed assumptions, that is, the associated single-valued operator A involved in the monotone variational inclusion problem is assumed to be monotone and Lipschitz continuous and the proposed method uses the stepsizes that are generated at each iteration by some simple calculations, which allows it to be easily implemented without the prior knowledge of the operator norm or the Lipschitz constant of the single-valued operator. The method was also demonstrated to have the capacity of solving other types of split inverse problems. Some applications of the method were discussed and analyzed numerically.

References

[1] Y. Censor, T. Elfving, A multiprojection algorithm using Bregman projections in product space, Numer. Algor. 8 (1994), 221–239.

[2] Y. Censor, A. Gibali, S. Reich, Algorithms for the split variational inequality problem, Numer. Algor. 59 (2012), 301-323.

[3] Y. Censor, A. Gibali, S. Reich, The split variational inequality problem. The Technion-Israel Institute of Technology, Haifa, 2010. arXiv:1009.3780.

[4] A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, SIAM J. Imaging Sci. 2 (2009), 183-202.

[5] T. Humphries, M. Loreto, B. Halter, W. O'Keeffe, L. Ramirez, Comparison of regularized and superiorized methods for tomographic image reconstruction, J. Appl. Numer. Optim. 2 (2020), 77-99.

[6] C. Byrne, Y. Censor, A. Gibali, S. Reich, Weak and strong convergence of algorithms for the split common null point problem, J. Nonlinear Convex Anal. 13 (2012), 759-775.

[7] W. Takahashi, H. K. Xu, J. C. Yao, Iterative methods for generalized split feasibility problems in Hilbert spaces, Set-valued Var. Anal. 23 (2015), 205-221.

[8] M. Tian, B-N. Jiang, Viscosity approximation Methods for a Class of generalized split feasibility problems with variational inequalities in Hilbert space, Numer. Funct. Anal. Optim. 40 (2019), 902-923.

[9] A. Moudafi, Split monotone variational inclusions, J. Optim. Theory Appl. 150 (2011), 275-283.

[10] S. Takahashi, W. Takahashi, The split common null point problem and the shrinking projection method in Banach spaces, Optimization 65 (2016), 281-287.

[11] J.S. Jung, A general iterative algorithm for split variational inclusion problems and fixed point problems of a pseudocontractive mapping, J. Nonlinear Funct. Anal. 2022 (2022), 13.

[12] J.N. Ezeora, C. Izuchukwu, Iterative approximation of solution of split variational inclusion problem, Filomat, 32 (2019), 2921-2932.

[13] C. Izuchukwu, J.N. Ezeora, J. Martinez-Moreno, A modified forward-backward splitting method for a certain class of split monotone variational inclusion problem with application to split linear inverse problems, Comput. Appl. Math. 39 (2020), 188.

[14] L.C. Ceng, On a viscosity iterative algorithm for variational inclusion problems and the fixed point problem of countably many nonexpansive mappings, Appl. Set-Valued Anal. Optim. 3 (2021), 203-214.

[15] G. Lopez, V. Martin-Marquez, F. Wang, H. K. Xu, Solving the split feasibility problem without prior knowledge of matrix norms, Inverse Probl.28 (2012), 085004.

[16] A. Moudafi, B. S. Thakur, Solving proximal split feasibility problems without prior knowledge of operator norms, Optim. Lett. 8 (2014), 2099-2110.

[17] Y. Tang, A. Gibali, New self-adaptive step size algorithms for solving split variational inclusion problems and its applications, Numer. Algor, 83 (2020), 305-331.

[18] C.C. Okeke, C. Izuchukwu, A strong convergence theorem for monotone inclusion and minimization problems in complete CAT(0) spaces, Optim. Methods Softw. 34 (2019), 1168-1183.

[19] F. U. Ogbuisi, C. Izuchukwu, Approximating a zero of sum of two monotone operators which solves a fixed point problem in reflexive Banach spaces, Numer. Funct. Anal. Optim. 42 (2020), 322-343.

[20] R. T. Rockafellar, On the maximality of sums of nonlinear monotone operators, Trans. Amer. Math. Soc. 149 (1970), 75-288.

[21] L. Liu, S.Y. Cho, J.C. Yao, Convergence analysis of an inertial Tseng's extragradient algorithm for solving pseudomonotone variational inequalities and applications, J. Nonlinear Var. Anal. 5 (2021), 627-644.

[22] X. Qin, J.C. Yao, Projection splitting algorithms for nonself operators, J. Nonlinear Convex Anal. 18 (2017), 925–935.

[23] L.C. Ceng, et al., A modified inertial subgradient extragradient method for solving pseudomonotone variational inequalities and common fixed point problems, Fixed Point Theory 21 (2020), 93-108.

[24] L. Liu, X. Qin, Strong convergence theorems for solving pseudo-monotone variational inequality problems and applications, Optimization, 71 (2022), 3603-3626.

[25] H. K. Xu, Iterative methods for split feasibility problem in infinite-dimensional Hilbert space, Inverse Probl. 26 (2010), 105018.

[26] H. Zhou, X. Qin, Fixed Points of Nonlinear Operators. Iterative Method, De Gruyter 2020.

[27] B. Lemaire, Which Fixed Point Does the Iteration Method Select?, In: Gritzmann, P., Horst, R., Sachs, E., Tichatschke, R. (eds) Recent Advances in Optimization. Lecture Notes in Economics and Mathematical Systems, vol 452. Springer, Berlin, Heidelberg, 1997.
[28] S. Saejung, P. Yotkaew, Approximation of zeros of inverse strongly monotone operators in Banach spaces, Nonlinear Anal. 75 (2012), 742-750.
[29] P. E. Mainge, Approximation methods for common fixed points of nonexpansive mappings in Hilbert spaces, J. Math. Anal. Appl. 325 (2007), 469-479.
[30] J. Yang, H. Liu, Strong convergence result for solving monotone variational inequalities in Hilbert space, Numer. Algor. 80 (2019), 741-752.
[31] S.A. Khan, S. Suantai, W. Cholamjiak, Shrinking projection methods involving inertial forward-backward splitting methods for inclusion problems, RACSAM 113 (2019), 645-656.
[32] L.S. Pontryagin, V.G. Boltyanskii, R.V. Gamkrelidze, E.F. Mishchenko, The Mathematical Theory of Optimal Processes, Wiley, New York, 1962.
[33] E. V. Khoroshilova, Extragradient-type method for optimal control problem with linear constraints and convex objective function, Optim. Lett. 7 (2013), 1193-1214.
[34] J.C. Butcher, Numerical Methods for Ordinary Differential Equations, Wiley, New York, 2003.
[35] M. Seydenschwanz, Convergence results for the discrete regularization of linear-quadratic control problems with bang–bang solutions, Comput. Optim. Appl. 629 (2015), 731-760.
[36] Y. Shehu, O.S. Iyiola, F.U. Ogbuisi, Iterative method with inertial terms for nonexpansive mappings: applications to compressed sensing, Numer. Algor. 83 (2020), 1321-1347.
[37] P.L. Combettes, V. Wajs, Signal recovery by proximal forward-backward splitting, SIAM J. Multiscale Model Simul. 4 (2005), 1168-1200.
[38] Y. Wang, H. K. Xu, Strong convergence for the proximal-gradient method, J. Nonlinear Convex Anal. 15 (2014), 581-593.
[39] J. M. Hendrickx, A. Olshevsky, Matrix P-norms are NP-hard to approximate if $P \neq 1, 2, \infty$, SIAM J. Matrix Anal. Appl. 31 (2010), 2802-2812.
[40] C. Izuchukwu, C. C. Okeke, F. O. Isiogugu, A viscosity iterative technique for split variational inclusion and fixed point problems between a Hilbert space and a Banach space, J. Fixed Point Theory Appl. 20 (2018), 1-25.