FIXED POINT METHODS AND OPTIMIZATION
Volume 1 (2024), No. 2, 101–124
https://doi.org/10.69829/fpmo-024-0102-ta02
Tulipa Opera Scholarum

FORWARD-REFLECTED-BACKWARD METHOD WITH TWO-STEP INERTIAL FOR VARIATIONAL INEQUALITIES

CHINEDU IZUCHUKWU[1] AND YEKINI SHEHU[2*]

[1] *School of Mathematics, University of the Witwatersrand, Private Bag 3, Johannesburg, 2050, South Africa*
[2] *School of Mathematical Sciences, Zhejiang Normal University, Jinhua, 321004, People's Republic of China*

Abstract. A forward-reflected-backward splitting method of Malitsky-Tam with two-step inertial extrapolation and self-adaptive step sizes is proposed to solve variational inequalities in quasi-monotone setting. Our method features one projection onto the feasible set and one functional evaluation at each iteration. A two-step inertial extrapolation is added to further improve on the convergence speed of the proposed method and self-adaptive step sizes are used in order to reduce computational complexity of our method. Weak convergence analysis are obtained under some easy to verify conditions on the iterative parameters in Hilbert spaces. Preliminary numerical tests are performed to support the theoretical analysis and show the superiority of our method over recent related methods in the literature.

Keywords. Forward-backward method, variational inequalities, quasi-monotone, two-step inertial, weak convergence, Hilbert spaces.

1. Introduction

Throughout this paper, we assume that H is a real Hilbert space with inner product $\langle .,. \rangle$ and induced norm $\|.\|$. Suppose C is a nonempty, closed and convex subset of H. Given a continuous operator $A : H \to H$, the Variational Inequality Problem ((VIP) for short) is defined as:

$$\text{Find } x^* \in C \text{ such that } \langle Ax^*, z - x^* \rangle \geq 0 \ \forall z \in C. \tag{1.1}$$

It has been shown in [1, 2, 11, 13, 17, 18, 28, 37], for example, that problems arising from economics, engineering mechanics, mathematical programming, transportation, and other applied sciences, can be converted to VIP (1.1). Let S represent the set of solutions to VIP (1.1).

Several projection-type iterative methods have been proposed to solve VIP (1.1) in the literature. One of the popular methods is the extragradient method [19]: $x_1 \in C$, $\gamma_n \in (0, \frac{1}{L})$ and $L > 0$,

$$\begin{cases} y_n = P_C(x_n - \gamma_n A x_n) \\ x_{n+1} = P_C(x_n - \gamma_n A y_n), n \geq 1, \end{cases} \tag{1.2}$$

where A is (pseudo)-monotone and Lipschitz continuous in Hilbert spaces. Several versions of (1.2) have been studied in the literature (see, e.g., [3, 14, 29, 43, 47]). The extragradient method (1.2) is computationally expensive especially in cases where P_C does not have a closed-form solution and A has complex evaluation (like in problems arising from optimal control theory).

*Corresponding author.
E-mail address: chinedu.izuchukwu@wits.ac.za (C. Izuchukwu), yekini.shehu@zjnu.edu.cn (Y. Shehu)
2020 Mathematics Subject Classification: 47H05, 47J20, 47J25, 65K15, 90C25.
Accepted November 21, 2024.

The iterative method: $x_1, y_1 \in C$ and $\gamma_n \in (0, \frac{1}{3L}]$,

$$\begin{cases} x_{n+1} = P_C(x_n - \gamma_n A y_n) \\ y_{n+1} = P_C(x_{n+1} - \gamma_n A y_n), n \geq 1, \end{cases} \tag{1.3}$$

was proposed by Popov [33] and weak convergence results were obtained when A is (pseudo)-monotone and Lipschitz continuous in Hilbert spaces (see, e.g., [12, 27]). The method (1.3) is computationally cheaper than the extragradient method (1.2) because (1.3) requires one (rather than two) functional evaluation of A at each iteration. However, (1.3) involves two P_C per iteration.

The subgradient extragradient method [4]: $x_1 \in H$, $\gamma_n \in (0, \frac{1}{L})$ and $L > 0$,

$$\begin{cases} y_n = P_C(x_n - \gamma_n A x_n) \\ T_n := \{w \in H : \langle x_n - \gamma_n A x_n - y_n, w - y_n \rangle \leq 0\}, \\ x_{n+1} = P_{T_n}(x_n - \gamma_n A y_n), n \geq 1, \end{cases} \tag{1.4}$$

was proposed by Censor et al. [4], which features two functional evaluation of A, one projection onto C and one projection onto half-space at each iteration. Weak convergence results of (1.4) were obtained when A is (pseudo)-monotone and Lipchitz continuous (see, e.g., [5, 6, 20, 36, 50]).

The forward-backward-forward method [45] is given as: $x_1 \in H$, $\gamma_n \in (0, \frac{1}{L})$ and $L > 0$,

$$\begin{cases} y_n = P_C(x_n - \gamma_n A x_n) \\ x_{n+1} = y_n + \gamma_n (A x_n - A y_n), n \geq 1. \end{cases} \tag{1.5}$$

The method converges weakly and it has only one projection P_C and but two functional evaluations of A at each iteration.

Inertial versions of the above-mentioned methods (1.2)-(1.5) with one-step inertial extrapolation $w_n = x_n + \theta(x_n - x_{n-1}), \theta \in [0, 1)$ have also been investigated in the literature and convergence results were obtained under the assumption that A is (pseudo)-monotone and Lipschitz continuous (see, for example, [7, 10, 35, 40–42, 44]).

Related works. In [24], Malitsky introduced the projected reflected gradient method: $x_1, y_1 \in H$, $\gamma_n \in (0, \frac{\sqrt{2}-1}{L})$ and $L > 0$,

$$\begin{cases} x_{n+1} = P_C(x_n - \gamma_n A y_n) \\ y_{n+1} = 2x_{n+1} - x_n, n \geq 1, \end{cases} \tag{1.6}$$

and gave weak convergence results for solving VIP (1.1) in real Hilbert spaces with A being monotone and Lipschitz continuous. It can be seen that method (1.6) involves only one P_C and one functional evaluation of A at each iteration unlike the methods (1.2)-(1.5). Numerical tests in [24] showed that method (1.6) is more efficient and outperforms methods (1.2)-(1.5). Several modifications of the projected reflected gradient method (1.6) with the extrapolation $w_n = x_n + \theta(x_n - x_{n-1}), \theta \geq 0$ have been studied in [23, 26, 39, 51] when A is monotone and Lipschitz continuous.

In [26], a forward-reflected-backward splitting method was studied. We give the VIP setting as (see [26, Algorithm 3.1]): $x_0, x_1 \in H$, $\gamma_0, \gamma_1 > 0$, $\delta \in (0, 1)$, $\sigma \in (0, 1)$ and $\rho \in \{1, \sigma^{-1}\}$;

$$x_{n+1} = P_C(x_n - \gamma_n \mathcal{A} x_n - \gamma_{n-1}(\mathcal{A} x_n - \mathcal{A} x_{n-1})), \ n \geq 1, \tag{1.7}$$

where $\gamma_n = \rho\gamma_{n-1}\sigma^i$ with i being the smallest nonnegative integer satisfying

$$\gamma_n\|\mathcal{A}x_{n+1} - \mathcal{A}x_n\| \leq \frac{\delta}{2}\|x_{n+1} - x_n\|.$$

Weak convergence of (1.7) and its one-step inertial extrapolation $w_n = x_n + \theta(x_n - x_{n-1}), \theta \in [0, 1)$ with constant step sizes (see [26, Section 4]) was obtained in [26] when A is monotone and Lipschitz continuous, and it is noticed that (1.7) requires only one P_C and one functional evaluation of A per iteration.

At this point, we want to point out that very few projection-type methods for solving VIP (1.1) when A is quasi-monotone (which is weaker than the pseudo-monotonicity or monotonicity assumption) are available in the literature. This is partly due to the fact that the arguments for the convergence analysis of these projection-type methods when A is monotone or pseudo-monotone cannot be carried over when A is quasi-monotone. For example, if A is quasi-monotone in VIP (1.1), then weak or Minty formulation of VIP (1.1) is not equivalent to VIP (1.1).

Recently, Liu and Yang [22] proved that the forward-backward-forward method (1.5) converges weakly to a solution of VIP (1.1) when A is quasi-monotone, Lipschitz continuous and sequentially weakly continuous in an infinite dimensional Hilbert space. They also pointed out that their weak convergence result holds for the extragradient method (1.2) (see also [34]) and the subgradient extragradient method (1.4) with the inherent drawbacks of these methods enumerated above in (1.2)-(1.5).

Quite recently, Wang et al. [48] obtained weak convergence results of a projection and contraction method with one-step inertial extrapolation $w_n = x_n + \theta(x_n - x_{n-1}), \theta \in [0, 1)$ and Barzilai-Borwein step size strategy to solve VIP (1.1) when A is quasi-monotone and Lipschitz continuous in Hilbert spaces. However, the proposed method in [48] involves computations of two projections onto the feasible set C and two evaluations of A.

It was shown in [32, Section 4] by example that one-step inertial extrapolation $w_n = x_n + \theta(x_n - x_{n-1}), \theta \in [0, 1)$ may fail to provide acceleration. It was remarked in [21, Chapter 4] that the use of inertia of more than two points x_n, x_{n-1} could provide acceleration. For example, the following two-step inertial extrapolation

$$y_n = x_n + \theta(x_n - x_{n-1}) + \delta(x_{n-1} - x_{n-2}) \tag{1.8}$$

with $\theta > 0$ and $\delta < 0$ can provide acceleration. The failure of one-step inertial acceleration of ADMM was also discussed in [31, Section 3] and adaptive acceleration for ADMM was proposed instead. Polyak [30] also discussed that the multi-step inertial methods can boost the speed of optimization methods even though neither the convergence nor the rate result of such multi-step inertial methods was established in [30]. Some results on multi-step inertial methods have recently been studied in [8,9].

Our contributions. Motivated by [26], our aim in this paper is to propose a forward-reflected-backward splitting method with one evaluation of A, one computation of P_C, two-step inertial extrapolation and self-adaptive step sizes to solve VIP (1.1) when A is quasi-monotone. Our proposed method has the following features:

- our method involves one projection onto feasible set C per iteration;
- one evaluation of A is only needed at each iteration;
- two-step inertial extrapolation is incorporated in our proposed method to speed up the iterations;
- self-adaptive step sizes are adopted in our proposed method.

In comparisons with the methods proposed in [22, 38, 48, 49, 53], our contributions in this paper are:

- we propose a two-step inertial forward-reflected-backward splitting method to solve VIP (1.1) and we give weak convergence results when A is quasi-monotone and Lipschitz continuous in real Hilbert spaces. Our proposed method is simple and does not require neither further evaluations of A nor further projections as compared with the methods in [22, 48]. Also, our method does not involve the projection onto intersection of the feasible set C and $n+1$ half spaces as done in [38, 49, 53];
- our proposed method is applicable in situations where A is pseudo-monotone and Lipchitz continuous. Therefore, our method overcomes the deficiency in the one-step inertial extragradient method studied in [40], subgradient extragradient method in [6, 52] and one-step inertial subgradient extragradient method in [7, 10, 36, 44, 50] for both monotone and pseudo-monotone cases of VIP (1.1);
- we give numerical computations of our proposed method and compare with the methods in [22, 25, 26, 46, 48]. Our preliminary computational results show that our proposed method is efficient and converges faster (in terms of CPU time and number of iterations) than the methods in [22, 25, 26, 46, 48].

Outline. The paper is arranged as follows: In Section 2, we give some basic definitions and results needed in our analysis while in Section 3, we introduce our proposed method. Weak convergence analysis of our method is presented in Section 4 and numerical tests are performed in Section 5. Finally, we give concluding remarks in Section 6.

2. Preliminaries

An operator $A : H \to H$ is

(i) *Lipschitz continuous* with constant L, if there exists $L > 0$ such that
$$\|Ax - Ay\| \le L\|x - y\| \ \forall x, y \in H,$$
(ii) *monotone*, if
$$\big\langle Ax - Ay, x - y\big\rangle \ge 0 \ \forall x, y \in H,$$
(iii) *pseudo-monotone*, if
$$\langle Ay, x - y\rangle \ge 0 \implies \langle Ax, x - y\rangle \ge 0 \ \forall x, y \in H,$$
(iv) quasi-monotone, if
$$\langle Ay, x - y\rangle > 0 \implies \langle Ax, x - y\rangle \ge 0 \ \forall x, y \in H,$$
(v) *sequentially weakly-strongly continuous,* if for every sequence $\{x_n\}$ that converges weakly to a point x, the sequence $\{Ax_n\}$ converges strongly to Ax,

(vi) *sequentially weakly continuous,* if for every sequence $\{x_n\}$ that converges weakly to a point x, the sequence $\{Ax_n\}$ converges weakly to Ax.

Clearly, $(ii) \implies (iii) \implies (iv)$ but the converses may fail.
Let S_D be the solution set of the following Minty formulation of VIP (1.1):
$$\text{Find } x^* \in C \text{ such that } \langle Az, z - x^*\rangle \ge 0 \ \forall z \in C. \tag{2.1}$$
Then, S_D is a closed and convex subset of C, and since C is convex and A is continuous, we have that $S_D \subset S$.

S_D is nonempty when the following situations arise.

Lemma 2.1. *(see [53]) Suppose either*

(i) A *is pseudo-monotone on* C *and* $S \neq \emptyset$,
(ii) A *is the gradient of* G, *where* G *is a differential quasi-convex function on an open set* $\mathcal{K} \supset C$ *and attains its global minimum on* C,
(iii) A *is quasi-monotone on* C, $A \neq 0$ *on* C *and* C *is bounded,*
(iv) A *is quasi-monotone on* C, $A \neq 0$ *on* C *and there exists a positive number* r *such that, for every* $x \in C$ *with* $\|x\| \geq r$, *there exists* $y \in C$ *such that* $\|y\| \leq r$ *and* $\langle Ax, y - x\rangle \leq 0$,
(v) A *is quasi-monotone on* C, *int* C *is nonempty and there exists* $z \in S$ *such that* $Az \neq 0$.

Then, S_D *is nonempty.*

The metric projection, denoted by P_C, is a mapping defined on H onto C which assigns to each $v \in H$, the unique point in C, denoted by $P_C v$ such that

$$\|v - P_C v\| = \inf\{\|v - y\| : \ y \in C\}.$$

It is well known that P_C is characterized by the inequality

$$\langle v - P_C v, y - P_C v\rangle \leq 0, \ \ \forall y \in C. \tag{2.2}$$

Lemma 2.2. *The following identities hold for all* $u, v \in H$*:*

$$2\langle u, v\rangle = \|u\|^2 + \|v\|^2 - \|u - v\|^2 = \|u + v\|^2 - \|u\|^2 - \|v\|^2.$$

Lemma 2.3. *Let* $x, y, z \in H$ *and* $a, b \in \mathbb{R}$. *Then*

$$\begin{aligned} \|(1+a)x - (a-b)y - bz\|^2 &= (1+a)\|x\|^2 - (a-b)\|y\|^2 - b\|z\|^2 + (1+a)(a-b)\|x-y\|^2 \\ &\quad + b(1+a)\|x-z\|^2 - b(a-b)\|y-z\|^2. \end{aligned}$$

Proof.

$$\begin{aligned} \|(1+a)x - (a-b)y - bz\|^2 &= \langle (1+a)x - (a-b)y - bz, (1+a)x - (a-b)y - bz\rangle \\ &= (1+a)^2\|x\|^2 - 2(1+a)(a-b)\langle x, y\rangle - 2b(1+a)\langle x, z\rangle \\ &\quad + 2b(a-b)\langle y, z\rangle + (a-b)^2\|y\|^2 + b^2\|z\|^2 \\ &= (1+a)^2\|x\|^2 - (1+a)(a-b)(\|x\|^2 + \|y\|^2 - \|x-y\|^2) \\ &\quad - b(1+a)(\|x\|^2 + \|z\|^2 - \|x-z\|^2) \\ &\quad + b(a-b)(\|y\|^2 + \|z\|^2 - \|y-z\|^2) + (a-b)^2\|y\|^2 + b^2\|z\|^2 \\ &= (1+a)\|x\|^2 - (a-b)\|y\|^2 - b\|z\|^2 \\ &\quad + (1+a)(a-b)\|x-y\|^2 + b(1+a)\|x-z\|^2 - b(a-b)\|y-z\|^2. \end{aligned}$$

□

3. Proposed Method

We introduce our proposed method to solve VIP (1.1) below and give some discussions about our method.

Algorithm 1 2-Step Inertial Projection Method with Adaptive Step Size

1: Choose the parameters $\gamma_0, \gamma_1 > 0$, $\mu \in \left(\delta, \frac{1-2\delta}{2}\right)$ with $\delta \in (0, \frac{1}{4})$, $\beta \leq 0$, and $\theta \in [0, 1)$. Choose a nonnegative real sequence $\{a_n\}$ such that $\sum_{n=1}^{\infty} a_n < \infty$. Choose $x_{-1}, x_0, x_1 \in H$ as starting points. Set $n := 1$.

2: Compute

$$\left\{ \begin{array}{l} w_n = x_n + \theta(x_n - x_{n-1}) + \beta(x_{n-1} - x_{n-2}), \\ x_{n+1} = P_C\left(w_n - ((\gamma_n + \gamma_{n-1})Ax_n - \gamma_{n-1}Ax_{n-1})\right), \end{array} \right. \tag{3.1}$$

where

$$\gamma_{n+1} = \begin{cases} \min\left\{\frac{\mu\|x_n - x_{n+1}\|}{\|Ax_n - Ax_{n+1}\|},\ \gamma_n + a_n\right\}, & \text{if } Ax_n \neq Ax_{n+1}, \\ \gamma_n + a_n, & \text{otherwise.} \end{cases} \tag{3.2}$$

3: Set $n \leftarrow n+1$, and **go to 2**.

We assume that the following conditions are satisfied in order to obtain weak convergence of our proposed Algorithm 1.

Assumption 3.1. Suppose the following conditions are fulfilled:

(a) $S_D \neq \emptyset$,
(b) A is Lipschitz continuous on C with constant $L > 0$,
(c) A satisfies the following condition: whenever $\{x_n\} \subset C$ and $x_n \rightharpoonup v^*$, one has $\|Av^*\| \leq \liminf_{n\to\infty} \|Ax_n\|$,
(d) A is quasi-monotone on H.

We further assume the following conditions on the inertial parameters θ and β, and iterative parameters δ and μ in Algorithm 1.

Assumption 3.2. Assume that θ and β satisfy the following conditions

(a) $0 \leq \theta < \min\left\{\frac{1-\mu}{2}, \frac{2\delta}{3}, \frac{\frac{1}{2}-\beta+\delta-\mu}{3}\right\}$,
(b) $\beta \leq 0$.

Remark 3.3. Observe that if $\mu < \frac{1-2\delta}{2}$, then $0 < \frac{1}{2} + \delta - \mu$. We also note that if $\beta \leq 0$, then by Assumption 3.2 (a), we obtain

$$\beta < \min\left\{\frac{2\delta - 3\theta}{2(1+\theta)}, \frac{1}{2} + \delta - \mu\right\}.$$

Remark 3.4.

(i) Clearly, Algorithm 1 requires only one metric projection onto the feasible set C and one evaluation of A per iteration.
(ii) Assumption 3.1(c) is strictly weaker than the sequentially weakly continuous assumption in [22] and other recent papers for solving pseudo-monotone VIPs. An example of an operator satisfying Assumption 3.1(c) but not sequentially weakly continuous is $Av = v\|v\|\ \forall v \in C$ (see [43]).
(iii) No Lipschitz constant of A is needed as input parameter in our proposed Algorithm 1 and Algorithm 1 does not adopt the line search procedure but rather self-adaptive step sizes. Observe from (3.2) that $\lim_{n\to\infty} \gamma_n = \gamma$, where $\gamma \in [\min\{\frac{\mu}{L}, \gamma_1\},\ \gamma_1 + a]$, with $a = \sum_{n=1}^{\infty} a_n$ (see [22]).
(iv) When $\theta = 0 = \beta$ and $a_n = 0$ in Algorithm 1, our proposed method reduces to the method studied in [46, Algorithm 3.1]. Also, Algorithm 1 becomes [16, Algorithm 3.2] when $\theta = 0 = \beta$.

The sequence $\{a_n\}$ introduced in (3.2) allows the step sizes to increase from iteration to iteration and hence, reduces the dependence of Algorithm 1 on the initial step size γ_1. Note also that since $\lim\limits_{n\to\infty} a_n = 0$, the step sizes maybe non-increasing for large n. Hence, $\{a_n\}$ is added in (3.2) to improve on the self-adaptive step sizes used in [46, Algorithm 3.1] (which are non-increasing).

4. Convergence Results

In this section, we give our convergence results below.

Lemma 4.1. *Suppose Assumptions 3.1 (a) and (b) are satisfied and Assumptions 3.2 are fulfilled. Then the sequence $\{x_n\}$ generated by Algorithm 1 is bounded.*

Proof. Pick $z \in S_D$. Then $z \in S \subset C$. Using (2.2) and Lemma 2.2, we obtain

$$\begin{aligned}
0 &\le 2\langle x_{n+1} - w_n + (\gamma_n + \gamma_{n-1})Ax_n - \gamma_{n-1}Ax_{n-1}, z - x_{n+1}\rangle \\
&= 2\langle x_{n+1} - w_n, z - x_{n+1}\rangle + 2\gamma_n\langle Ax_n, z - x_{n+1}\rangle \\
&+ 2\gamma_{n-1}\langle Ax_n - Ax_{n-1}, z - x_{n+1}\rangle \\
&= \|w_n - z\|^2 - \|x_{n+1} - w_n\|^2 - \|x_{n+1} - z\|^2 + 2\gamma_n\langle Ax_n, z - x_{n+1}\rangle \\
&+ 2\gamma_{n-1}\langle Ax_n - Ax_{n-1}, z - x_{n+1}\rangle.
\end{aligned} \tag{4.1}$$

Since $x_{n+1} \in C$ and $z \in S_D$, we get from (2.1) that $\langle Ax_{n+1}, x_{n+1} - z\rangle \ge 0,\ \ \forall n \ge 1$. This implies that $\langle Ax_n, z - x_{n+1}\rangle \le \langle Ax_n - Ax_{n+1}, z - x_{n+1}\rangle,\ \ \forall n \ge 1$. Then (4.1) becomes

$$\begin{aligned}
\|x_{n+1} - z\|^2 \le \|w_n - z\|^2 - \|x_{n+1} - w_n\|^2 + 2\gamma_n\langle Ax_n - Ax_{n+1}, z - x_{n+1}\rangle \\
+ 2\gamma_{n-1}\langle Ax_n - Ax_{n-1}, z - x_n\rangle + 2\gamma_{n-1}\langle Ax_n - Ax_{n-1}, x_n - x_{n+1}\rangle.
\end{aligned} \tag{4.2}$$

By (3.2), we get

$$\begin{aligned}
2\gamma_{n-1}\langle Ax_n - Ax_{n-1}, x_n - x_{n+1}\rangle &\le 2\gamma_{n-1}\|Ax_n - Ax_{n-1}\|\|x_n - x_{n+1}\| \\
&\le 2\frac{\gamma_{n-1}}{\gamma_n}\mu\|x_n - x_{n-1}\|\|x_n - x_{n+1}\| \\
&\le \frac{\gamma_{n-1}}{\gamma_n}\mu\Big(\|x_n - x_{n-1}\|^2 + \|x_{n+1} - x_n\|^2\Big).
\end{aligned} \tag{4.3}$$

Similarly, we obtain

$$2\gamma_n\langle Ax_{n+1} - Ax_n, z - x_{n+1}\rangle \ge -\frac{\gamma_n}{\gamma_{n+1}}\mu\Big(\|x_{n+1} - x_n\|^2 + \|x_{n+1} - z\|^2\Big). \tag{4.4}$$

By Remark 3.4(iii) and the condition $\mu \in \left(\delta,\ \frac{1-2\delta}{2}\right)$, we have that $\lim\limits_{n\to\infty} \frac{\gamma_{n-1}}{\gamma_n}\mu = \mu < \frac{1}{2} - \delta$. Hence, there exists $n_0 \ge 1$ such that $\frac{\gamma_{n-1}}{\gamma_n}\mu < \frac{1}{2} - \delta\ \forall n \ge n_0$. Furthermore, $-\frac{\gamma_n}{\gamma_{n+1}}\mu > -\frac{1}{2} + \delta > -\frac{1}{2}\ \forall n \ge n_0$. Consequently, (4.3) and (4.4) become

$$2\gamma_{n-1}\langle Ax_n - Ax_{n-1}, x_n - x_{n+1}\rangle \le \left(\frac{1}{2} - \delta\right)\Big(\|x_n - x_{n-1}\|^2 + \|x_{n+1} - x_n\|^2\Big), \tag{4.5}$$

$\forall n \ge n_0$ and

$$2\gamma_n\langle Ax_{n+1} - Ax_n, z - x_{n+1}\rangle \ge -\frac{1}{2}\Big(\|x_{n+1} - x_n\|^2 + \|x_{n+1} - z\|^2\Big), \tag{4.6}$$

$\forall n \ge n_0$, respectively.
Using (4.5) in (4.2), we have $\forall n \ge n_0$,

$$\begin{aligned}
\|x_{n+1} - z\|^2 \ \ \le\ \ & \|w_n - z\|^2 - \|x_{n+1} - w_n\|^2 + 2\gamma_n\langle Ax_n - Ax_{n+1}, z - x_{n+1}\rangle \\
& + 2\gamma_{n-1}\langle Ax_n - Ax_{n-1}, z - x_n\rangle + \Big(\frac{1}{2} - \delta\Big)\|x_n - x_{n-1}\|^2
\end{aligned}$$

$$+\Big(\frac{1}{2}-\delta\Big)\|x_{n+1}-x_n\|^2. \tag{4.7}$$

Observe that

$$\begin{aligned} w_n - z &= x_n + \theta(x_n - x_{n-1}) + \beta(x_{n-1} - x_{n-2}) - z \\ &= (1+\theta)(x_n - z) - (\theta - \beta)(x_{n-1} - z) - \beta(x_{n-2} - z). \end{aligned}$$

Therefore, by Lemma 2.3, we obtain

$$\begin{aligned} \|w_n - z\|^2 &= \|(1+\theta)(x_n - z) - (\theta - \beta)(x_{n-1} - z) - \beta(x_{n-2} - z)\|^2 \\ &= (1+\theta)\|x_n - z\|^2 - (\theta - \beta)\|x_{n-1} - z\|^2 - \beta\|x_{n-2} - z\|^2 \\ &\quad +(1+\theta)(\theta - \beta)\|x_n - x_{n-1}\|^2 + \beta(1+\theta)\|x_n - x_{n-2}\|^2 \\ &\quad -\beta(\theta - \beta)\|x_{n-1} - x_{n-2}\|^2. \end{aligned} \tag{4.8}$$

Furthermore,

$$\begin{aligned} \|x_{n+1} - w_n\|^2 &= \|x_{n+1} - (x_n + \theta(x_n - x_{n-1}) + \beta(x_{n-1} - x_{n-2}))\|^2 \\ &= \|x_{n+1} - x_n - \theta(x_n - x_{n-1}) - \beta(x_{n-1} - x_{n-2})\|^2 \\ &= \|x_{n+1} - x_n\|^2 - 2\theta\langle x_{n+1} - x_n, x_n - x_{n-1}\rangle \\ &\quad -2\beta\langle x_{n+1} - x_n, x_{n-1} - x_{n-2}\rangle + \theta^2\|x_n - x_{n-1}\|^2 \\ &\quad +2\beta\theta\langle x_n - x_{n-1}, x_{n-1} - x_{n-2}\rangle + \beta^2\|x_{n-1} - x_{n-2}\|^2 \\ &\geq \|x_{n+1} - x_n\|^2 - 2\theta\|x_{n+1} - x_n\|\|x_n - x_{n-1}\| \\ &\quad -2\beta\|x_{n+1} - x_n\|\|x_{n-1} - x_{n-2}\| + \theta^2\|x_n - x_{n-1}\|^2 \\ &\quad -2\beta\theta\|x_{n-1} - x_n\|\|x_{n-1} - x_{n-2}\| + \beta^2\|x_{n-1} - x_{n-2}\|^2 \\ &\geq \|x_{n+1} - x_n\|^2 - \theta\|x_{n+1} - x_n\|^2 - \theta\|x_n - x_{n-1}\|^2 \\ &\quad -\beta\|x_{n+1} - x_n\|^2 - \beta\|x_{n-1} - x_{n-2}\|^2 + \theta^2\|x_n - x_{n-1}\|^2 \\ &\quad -\beta\theta\|x_{n-1} - x_n\|^2 - \beta\theta\|x_{n-1} - x_{n-2}\|^2 + \beta^2\|x_{n-1} - x_{n-2}\|^2 \\ &= (1-\beta-\theta)\|x_{n+1} - x_n\|^2 + (\theta^2 - \theta - \beta\theta)\|x_n - x_{n-1}\|^2 \\ &\quad +(\beta^2 - \beta - \beta\theta)\|x_{n-1} - x_{n-2}\|^2. \end{aligned} \tag{4.9}$$

Using (4.8) and (4.9) in (4.7), we obtain for all $n \geq n_0$,

$$\begin{aligned} \|x_{n+1} - z\|^2 &\leq (1+\theta)\|x_n - z\|^2 - (\theta - \beta)\|x_{n-1} - z\|^2 - \beta\|x_{n-2} - z\|^2 \\ &\quad +(1+\theta)(\theta - \beta)\|x_n - x_{n-1}\|^2 + \beta(1+\theta)\|x_n - x_{n-2}\|^2 \\ &\quad -\beta(\theta - \beta)\|x_{n-1} - x_{n-2}\|^2 - (1-\beta-\theta)\|x_{n+1} - x_n\|^2 \\ &\quad -(\theta^2 - \theta - \beta\theta)\|x_n - x_{n-1}\|^2 - (\beta^2 - \beta - \beta\theta)\|x_{n-1} - x_{n-2}\|^2 \\ &\quad +2\gamma_{n-1}\langle Ax_n - Ax_{n-1}, z - x_n\rangle + \Big(\frac{1}{2}-\delta\Big)\|x_n - x_{n-1}\|^2 \\ &\quad +\Big(\frac{1}{2}-\delta\Big)\|x_{n+1} - x_n\|^2 + 2\gamma_n\langle Ax_n - Ax_{n+1}, z - x_{n+1}\rangle. \end{aligned} \tag{4.10}$$

Then, we have from (4.10) (noting that $2\beta(1+\theta) \leq 0$) that for all $n \geq n_0$,

$$\begin{aligned} \|x_{n+1} - z\|^2 &\leq (1+\theta)\|x_n - z\|^2 - (\theta - \beta)\|x_{n-1} - z\|^2 - \beta\|x_{n-2} - z\|^2 \\ &\quad +(1+\theta)(\theta - \beta)\|x_n - x_{n-1}\|^2 - \beta(\theta - \beta)\|x_{n-1} - x_{n-2}\|^2 \\ &\quad -(1-\beta-\theta)\|x_{n+1} - x_n\|^2 - (\theta^2 - \theta - \beta\theta)\|x_n - x_{n-1}\|^2 \\ &\quad -(\beta^2 - \beta - \beta\theta)\|x_{n-1} - x_{n-2}\|^2 + 2\gamma_{n-1}\langle Ax_n - Ax_{n-1}, z - x_n\rangle \\ &\quad +\Big(\frac{1}{2}-\delta\Big)\|x_n - x_{n-1}\|^2 + \Big(\frac{1}{2}-\delta\Big)\|x_{n+1} - x_n\|^2 \end{aligned}$$

$$
\begin{aligned}
&+2\gamma_n\langle Ax_n - Ax_{n+1}, z - x_{n+1}\rangle \\
= \quad & (1+\theta)\|x_n - z\|^2 - (\theta-\beta)\|x_{n-1} - z\|^2 - \beta\|x_{n-2} - z\|^2 \\
&+\Big[(1+\theta)(\theta-\beta) - (\theta^2 - \theta - \beta\theta)\Big]\|x_n - x_{n-1}\|^2 - (1-\beta-\theta)\|x_{n+1} - x_n\|^2 \\
&-\Big[\beta(\theta-\beta) + (\beta^2 - \beta - \beta\theta)\Big]\|x_{n-1} - x_{n-2}\|^2 + \Big(\frac{1}{2} - \delta\Big)\|x_n - x_{n-1}\|^2 \\
&+2\gamma_{n-1}\langle Ax_n - Ax_{n-1}, z - x_n\rangle + \Big(\frac{1}{2} - \delta\Big)\|x_{n+1} - x_n\|^2 \\
&+2\gamma_n\langle Ax_n - Ax_{n+1}, z - x_{n+1}\rangle. \qquad (4.11)
\end{aligned}
$$

Therefore, we obtain from (4.11) that

$$
\begin{aligned}
& \|x_{n+1} - z\|^2 - \theta\|x_n - z\|^2 - \beta\|x_{n-1} - z\|^2 + (1-\beta-\theta)\|x_{n+1} - x_n\|^2 \\
&+2\gamma_n\langle Ax_{n+1} - Ax_n, z - x_{n+1}\rangle \\
\leq \quad & \|x_n - z\|^2 - \theta\|x_{n-1} - z\|^2 - \beta\|x_{n-2} - z\|^2 + (1-\beta-\theta)\|x_n - x_{n-1}\|^2 \\
&+(3\theta-1)\|x_n - x_{n-1}\|^2 + \beta\|x_{n-1} - x_{n-2}\|^2 \\
&+2\gamma_{n-1}\langle Ax_n - Ax_{n-1}, z - x_n\rangle + \Big(\frac{1}{2} - \delta\Big)\|x_n - x_{n-1}\|^2 \\
&+\Big(\frac{1}{2} - \delta\Big)\|x_{n+1} - x_n\|^2, \forall n \geq n_0.
\end{aligned}
$$

Thus, $\forall n \geq n_0$,

$$
\begin{aligned}
& \|x_{n+1} - z\|^2 - \theta\|x_n - z\|^2 - \beta\|x_{n-1} - z\|^2 + \Big(\frac{1}{2} - \beta - \theta + \delta\Big)\|x_{n+1} - x_n\|^2 \\
&+2\gamma_n\langle Ax_{n+1} - Ax_n, z - x_{n+1}\rangle \\
\leq \quad & \|x_n - z\|^2 - \theta\|x_{n-1} - z\|^2 - \beta\|x_{n-2} - z\|^2 + \Big(\frac{1}{2} - \beta - \theta + \delta\Big)\|x_n - x_{n-1}\|^2 \\
&+(1-2\delta)\|x_n - x_{n-1}\|^2 + (3\theta-1)\|x_n - x_{n-1}\|^2 + \beta\|x_{n-1} - x_{n-2}\|^2 \\
&+2\gamma_{n-1}\langle Ax_n - Ax_{n-1}, z - x_n\rangle. \qquad (4.12)
\end{aligned}
$$

For each $n \geq 0$, define

$$
\begin{aligned}
\Gamma_n := {} & \|x_n - z\|^2 - \theta\|x_{n-1} - z\|^2 - \beta\|x_{n-2} - z\|^2 + \Big(\frac{1}{2} - \beta - \theta + \delta\Big)\|x_n - x_{n-1}\|^2 \\
&+2\gamma_{n-1}\langle Ax_n - Ax_{n-1}, z - x_n\rangle.
\end{aligned}
$$

We next show that $\Gamma_n \geq 0$. Now,

$$
\begin{aligned}
\Gamma_n \quad = \quad & \|x_n - z\|^2 - \theta\|x_{n-1} - z\|^2 - \beta\|x_{n-2} - z\|^2 \\
&+\Big(\frac{1}{2} - \beta - \theta + \delta\Big)\|x_n - x_{n-1}\|^2 \\
&+2\gamma_{n-1}\langle Ax_n - Ax_{n-1}, z - x_n\rangle \\
\geq \quad & \|x_n - z\|^2 - \theta\|x_{n-1} - z\|^2 - \beta\|x_{n-2} - z\|^2 \\
&+\Big(\frac{1}{2} - \beta - \theta + \delta\Big)\|x_n - x_{n-1}\|^2 \\
&-2\gamma_{n-1}\|Ax_n - Ax_{n-1}\|\|x_n - z\| \\
\geq \quad & \|x_n - z\|^2 - \theta\|x_{n-1} - z\|^2 - \beta\|x_{n-2} - z\|^2 \\
&+\Big(\frac{1}{2} - \beta - \theta + \delta\Big)\|x_n - x_{n-1}\|^2
\end{aligned}
$$

$$
\begin{aligned}
& -2\frac{\gamma_{n-1}\mu}{\gamma_n}\|x_n - x_{n-1}\|\|x_n - z\| \\
\geq \;& \|x_n - z\|^2 - \theta\|x_{n-1} - z\|^2 - \beta\|x_{n-2} - z\|^2 \\
& +\Big(\frac{1}{2} - \beta - \theta + \delta\Big)\|x_n - x_{n-1}\|^2 \\
& -\frac{\gamma_{n-1}\mu}{\gamma_n}\Big[\|x_n - x_{n-1}\|^2 + \|x_n - z\|^2\Big] \\
= \;& \Big(1 - \frac{\gamma_{n-1}\mu}{\gamma_n}\Big)\|x_n - z\|^2 \\
& +\Big(\frac{1}{2} - \beta - \theta + \delta - \frac{\gamma_{n-1}\mu}{\gamma_n}\Big)\|x_n - x_{n-1}\|^2 \\
& -\theta\|x_{n-1} - z\|^2 - \beta\|x_{n-2} - z\|^2. \qquad (4.13)
\end{aligned}
$$

Using Lemma 2.2, we obtain

$$
\begin{aligned}
\|x_{n-1} - z\|^2 &= \|(x_{n-1} - x_n) + (x_n - z)\|^2 \\
&= \|x_n - x_{n-1}\|^2 + \|x_n - z\|^2 + 2\langle x_{n-1} - x_n, x_n - z\rangle \\
&\leq 2\|x_n - x_{n-1}\|^2 + 2\|x_n - z\|^2. \qquad (4.14)
\end{aligned}
$$

If we combine (4.13) and (4.14), we get

$$
\begin{aligned}
\Gamma_n \;&\geq \Big(1 - \frac{\gamma_{n-1}\mu}{\gamma_n}\Big)\|x_n - z\|^2 + \Big(\frac{1}{2} - \beta - \theta + \delta - \frac{\gamma_{n-1}\mu}{\gamma_n}\Big)\|x_n - x_{n-1}\|^2 \\
&\quad -2\theta\|x_n - x_{n-1}\|^2 - 2\theta\|x_n - z\|^2 - \beta\|x_{n-2} - z\|^2 \\
&= \Big(1 - \frac{\gamma_{n-1}\mu}{\gamma_n} - 2\theta\Big)\|x_n - z\|^2 - \beta\|x_{n-2} - z\|^2 \\
&\quad +\Big(\frac{1}{2} - \beta - \theta + \delta - \frac{\gamma_{n-1}\mu}{\gamma_n} - 2\theta\Big)\|x_n - x_{n-1}\|^2. \qquad (4.15)
\end{aligned}
$$

Observe that

$$
\lim_{n\to\infty}\Big(\frac{1}{2} - \beta - \theta + \delta - \frac{\gamma_{n-1}\mu}{\gamma_n} - 2\theta\Big) = \frac{1}{2} - \beta - \theta + \delta - \mu - 2\theta > 0,
$$

by the condition that $\theta < \frac{\frac{1}{2} - \beta + \delta - \mu}{3}$ and

$$
\lim_{n\to\infty}\Big(1 - \frac{\gamma_{n-1}\mu}{\gamma_n} - 2\theta\Big) = 1 - \mu - 2\theta > 0,
$$

by the condition that $\theta < \frac{1-\mu}{2}$. Therefore, there exists $n_1 \in \mathbb{N}, n_1 \geq n_0$ such that

$$
\frac{1}{2} - \beta - \theta + \delta - \frac{\gamma_{n-1}\mu}{\gamma_n} - 2\theta > 0;\; 1 - \frac{\gamma_{n-1}\mu}{\gamma_n} - 2\theta > 0, \forall n \geq n_1. \qquad (4.16)
$$

Hence, by (4.16) and $\beta \leq 0$, we obtain from (4.15) that $\Gamma_n \geq 0, \forall n \geq n_1 \geq n_0$.

Also by (4.12), we get

$$
\begin{aligned}
\Gamma_{n+1} &\leq \Gamma_n + \Big[(1 - 2\delta) + (3\theta - 1)\Big]\|x_n - x_{n-1}\|^2 \\
&\quad +\beta\|x_{n-1} - x_{n-2}\|^2 \\
&= \Gamma_n - (2\delta - 3\theta)\|x_n - x_{n-1}\|^2 + \beta\|x_{n-1} - x_{n-2}\|^2. \qquad (4.17)
\end{aligned}
$$

Since $\theta < \frac{2\delta}{3}$, we have $2\delta - 3\theta > 0$ and by $\beta \leq 0$, we obtain from (4.17) that $\lim_{n\to\infty} \Gamma_n$ exists. Consequently,

$$\lim_{n\to\infty} -\Big[-(2\delta - 3\theta)\|x_n - x_{n-1}\|^2 + \beta\|x_{n-1} - x_{n-2}\|^2\Big] = 0. \tag{4.18}$$

Hence,

$$\begin{aligned} 0 &\leq (2\delta - 3\theta)\|x_n - x_{n-1}\|^2 \\ &\leq -\Big[-(2\delta - 3\theta)\|x_n - x_{n-1}\|^2 + \beta\|x_{n-1} - x_{n-2}\|^2\Big]. \end{aligned} \tag{4.19}$$

We obtain from (4.19) that

$$\lim_{n\to\infty} (2\delta - 3\theta)\|x_n - x_{n-1}\|^2 = 0.$$

Hence, we have

$$\lim_{n\to\infty} \|x_{n+1} - x_n\| = 0. \tag{4.20}$$

Also,

$$\begin{aligned} \|x_{n+1} - w_n\| &= \|x_{n+1} - x_n - \theta(x_n - x_{n-1}) - \beta(x_{n-1} - x_{n-2})\| \\ &\leq \|x_{n+1} - x_n\| + \theta\|x_n - x_{n-1}\| + |\beta|\|x_{n-1} - x_{n-2}\| \to 0 \end{aligned} \tag{4.21}$$

as $n \to \infty$. Since $\lim_{n\to\infty} \Gamma_n$ exists and $\lim_{n\to\infty} \|x_n - x_{n-1}\| = 0$, we obtain from (4.15) that the sequence $\{x_n\}$ is bounded. □

Using the ideas in [22], we have the following lemma.

Lemma 4.2. *Let $\{x_n\}$ be generated by Algorithm 1 such that Assumption 3.1(a)-(d) and Assumptions 3.2 (a)-(b) are satisfied. If v^* is one of the weak cluster points of $\{x_n\}$, then we have at least one of the following: $v^* \in S_D$ or $Av^* = 0$.*

Proof. By Lemma 4.1, $\{x_n\}$ is bounded. Hence, let v^* be a weak cluster point of $\{x_n\}$. Then, we can choose a subsequence of $\{x_n\}$, denoted by $\{x_{n_k}\}$ such that $x_{n_k} \rightharpoonup v^* \in C$.
We consider the following two possible cases.
Case I: Suppose that $\limsup_{k\to\infty} \|Ax_{n_k}\| = 0$. Then, $\lim_{k\to\infty} \|Ax_{n_k}\| = \liminf_{k\to\infty} \|Ax_{n_k}\| = 0$. Thus, we obtain from Assumption 3.1(c) that

$$0 < \|Av^*\| \leq \liminf_{k\to\infty} \|Ax_{n_k}\| = 0. \tag{4.22}$$

This means that $Av^* = 0$.
Case II: Suppose that $\limsup_{k\to\infty} \|Ax_{n_k}\| > 0$. Then without loss of generality, we can choose a subsequence of $\{Ax_{n_k}\}$ still denoted by $\{Ax_{n_k}\}$ such that $\lim_{k\to\infty} \|Ax_{n_k}\| = M_1 > 0$. Now, using (2.2), we obtain for all $y \in C$, that

$$\begin{aligned} 0 &\leq \langle x_{n_k+1} - x_{n_k} + ((\gamma_{n_k} + \gamma_{n_k-1})Ax_{n_k} + \gamma_{n_k-1}Ax_{n_k-1}), y - x_{n_k+1}\rangle \\ &= \langle x_{n_k+1} - x_{n_k}, y - x_{n_k+1}\rangle + \gamma_{n_k}\langle Ax_{n_k}, y - x_{n_k}\rangle \\ &+ \gamma_{n_k}\langle Ax_{n_k}, x_{n_k} - x_{n_k+1}\rangle + \gamma_{n_k-1}\langle Ax_{n_k} - Ax_{n_k-1}, y - x_{n_k+1}\rangle. \end{aligned} \tag{4.23}$$

Since A is Lipschitz continuous on C, we have from (4.20) that $\lim_{n\to\infty} \|Ax_n - Ax_{n-1}\| = 0$. Now, using this and (4.20) in (4.23), we get

$$0 \leq \liminf_{k\to\infty} \langle Ax_{n_k}, y - x_{n_k}\rangle \leq \limsup_{k\to\infty} \langle Ax_{n_k}, y - x_{n_k}\rangle < \infty,\ \forall y \in C. \tag{4.24}$$

Based on (4.24), we consider the following two cases under **Case II**:

Case 1: Suppose that $\limsup_{k\to\infty}\langle Ax_{n_k}, y - x_{n_k}\rangle > 0 \ \forall y \in C$. Then we can choose a subsequence of $\{x_{n_k}\}$ denoted by $\{x_{n_{k_j}}\}$ such that $\lim_{j\to\infty}\langle Ax_{n_{k_j}}, y - x_{n_{k_j}}\rangle > 0$. Thus, there exists $j_0 \geq 1$ such that $\langle Ax_{n_{k_j}}, y - x_{n_{k_j}}\rangle > 0 \ \forall j \geq j_0$, which by the quasi-monotonicity of A on C, implies that $\langle Ay, y - x_{n_{k_j}}\rangle \geq 0 \ \forall y \in C, j \geq j_0$. Hence, letting $j \to \infty$, we get that $\langle Ay, y - v^*\rangle \geq 0 \ \forall y \in C$. Therefore, $v^* \in S_D$.

Case 2: Suppose that $\limsup_{k\to\infty}\langle Ax_{n_k}, y - x_{n_k}\rangle = 0 \ \forall y \in C$. Then, by (4.24), we get

$$\lim_{k\to\infty}\langle Ax_{n_k}, y - x_{n_k}\rangle = 0 \ \forall y \in C, \tag{4.25}$$

from which we get that

$$\langle Ax_{n_k}, y - x_{n_k}\rangle + |\langle Ax_{n_k}, y - x_{n_k}\rangle| + \frac{1}{k+1} > 0 \ \forall y \in C. \tag{4.26}$$

Also, since $\lim_{k\to\infty}\|Ax_{n_k}\| = M_1 > 0$, we can find $k_0 \geq 1$ such that $\|Ax_{n_k}\| > \frac{M_1}{2} \ \forall k \geq k_0$. Hence, we can set $b_{n_k} = \frac{Ax_{n_k}}{\|Ax_{n_k}\|^2} \ \forall k \geq k_0$. Thus, $\langle Ax_{n_k}, b_{n_k}\rangle = 1 \ \forall k \geq k_0$. Therefore, by (4.26), we get

$$\left\langle Ax_{n_k}, y + b_{n_k}\left[|\langle Ax_{n_k}, y - x_{n_k}\rangle| + \frac{1}{k+1} - x_{n_k}\right]\right\rangle > 0,$$

and using the quasi-monotonicity of A on H, we obtain

$$\left\langle A\Big(y + b_{n_k}\left[|\langle Ax_{n_k}, y - x_{n_k}\rangle| + \frac{1}{k+1}\right]\Big), y + b_{n_k}\big[|\langle Ax_{n_k}, y - x_{n_k}\rangle| + \frac{1}{k+1}\big] - x_{n_k}\right\rangle \geq 0.$$

This implies that

$$\begin{aligned}
&\langle Ay, y + b_{n_k}\left[|\langle Ax_{n_k}, y - x_{n_k}\rangle| + \frac{1}{k+1}\right] - x_{n_k}\rangle \\
&\geq \langle Ay, A(y + b_{n_k}\big[|\langle Ax_{n_k}, y - x_{n_k}\rangle| + \frac{1}{k+1}\big]), y + b_{n_k}\big[|\langle Ax_{n_k}, y - x_{n_k}\rangle| + \frac{1}{k+1}\big] - x_{n_k}\rangle \\
&\geq -\|Ay - A(y + b_{n_k}\left[|\langle Ax_{n_k}, y - x_{n_k}\rangle| + \frac{1}{k+1}\right])\| \cdot \|y + b_{n_k}\left[|\langle Ax_{n_k}, y - x_{n_k}\rangle| + \frac{1}{k+1}\right] - x_{n_k}\| \\
&\geq -L\|b_{n_k}\left[|\langle Ax_{n_k}, y - x_{n_k}\rangle| + \frac{1}{k+1}\right]\| \cdot \|y + b_{n_k}\left[|\langle Ax_{n_k}, y - x_{n_k}\rangle| + \frac{1}{k+1}\right] - x_{n_k}\| \\
&= \frac{-L}{\|Ax_{n_k}\|}\Big(|\langle Ax_{n_k}, y - x_{n_k}\rangle| + \frac{1}{k+1}\Big) \cdot \|y + b_{n_k}\left[|\langle Ax_{n_k}, y - x_{n_k}\rangle| + \frac{1}{k+1}\right] - x_{n_k}\| \\
&\geq \frac{-2L}{M_1}\Big(|\langle Ax_{n_k}, y - x_{n_k}\rangle| + \frac{1}{k+1}\Big)M_2,
\end{aligned} \tag{4.27}$$

for some $M_2 > 0$, where the existence of M_2 is from the boundedness of $\{y + b_{n_k}\Big[|\langle Ax_{n_k}, y - x_{n_k}\rangle| + \frac{1}{k+1}\Big] - x_{n_k}\}$. Now, observe that (4.25) implies that $\lim_{k\to\infty}\Big(|\langle Ax_{n_k}, y - x_{n_k}\rangle| + \frac{1}{k+1}\Big) = 0$. Thus, as $k \to \infty$ in (4.27), we get that $\langle Ay, y - v^*\rangle \geq 0, \ \ \forall y \in C$. Therefore, $v^* \in S_D$. □

We now give our weak convergence theorem below.

Theorem 4.3. *Suppose Assumptions 3.1(a)-(d), Assumptions 3.2 (a)-(b) are fulfilled and $Ax \neq 0, \forall x \in C$. Then, $\{x_n\}$ generated by Algorithm 1 converges weakly to an element of $S_D \subset S$.*

Proof. Suppose $w_\omega(x_n)$ is the set of weak cluster points of $\{x_n\}$. We show that

$$w_\omega(x_n) \subset S_D.$$

Take $v^* \in w_\omega(x_n)$. Then, there exists a subsequence $\{x_{n_k}\} \subset \{x_n\}$ such that $x_{n_k} \rightharpoonup v^*, k \to \infty$. Since C is weakly closed, we have that $v^* \in C$. Since $Ax \neq 0, \forall x \in C$, we have $Av^* \neq 0$. By Lemma

4.2, we get $v^* \in S_D$. Therefore, $w_\omega(x_n) \subset S_D$.

Since $\lim_{n\to\infty} \Gamma_n$ exists and $\lim_{n\to\infty} \|x_{n+1} - x_n\| = 0$, we have that

$$\lim_{n\to\infty} \Big[\|x_n - z\|^2 - \theta\|x_{n-1} - z\|^2 - \beta\|x_{n-2} - z\|^2\Big] \tag{4.28}$$

exists for all $z \in S_D$.

We now show that $x_n \rightharpoonup x^* \in S_D$. Let us assume that there exist $\{x_{n_k}\} \subset \{x_n\}$ and $\{x_{n_j}\} \subset \{x_n\}$ such that $x_{n_k} \rightharpoonup v^*, k \to \infty$ and $x_{n_j} \rightharpoonup x^*, j \to \infty$. We show that $v^* = x^*$.

Observe that

$$2\langle x_n, x^* - v^*\rangle = \|x_n - v^*\|^2 - \|x_n - x^*\|^2 - \|v^*\|^2 + \|x^*\|^2, \tag{4.29}$$

$$\begin{aligned} 2\langle -\theta x_{n-1}, x^* - v^*\rangle &= -\theta\|x_{n-1} - v^*\|^2 + \theta\|x_{n-1} - x^*\|^2 \\ &\quad +\theta\|v^*\|^2 - \theta\|x^*\|^2 \end{aligned} \tag{4.30}$$

and

$$\begin{aligned} 2\langle -\beta x_{n-2}, x^* - v^*\rangle &= -\beta\|x_{n-2} - v^*\|^2 + \beta\|x_{n-2} - x^*\|^2 \\ &\quad +\beta\|v^*\|^2 - \beta\|x^*\|^2. \end{aligned} \tag{4.31}$$

Addition of (4.29), (4.30) and (4.31) gives

$$\begin{aligned} 2\langle x_n - \theta x_{n-1} - \beta x_{n-2}, x^* - v^*\rangle &= \Big(\|x_n - v^*\|^2 - \theta\|x_{n-1} - v^*\|^2 - \beta\|x_{n-2} - v^*\|^2\Big) \\ &\quad -\Big(\|x_n - x^*\|^2 - \theta\|x_{n-1} - x^*\|^2 - \beta\|x_{n-2} - x^*\|^2\Big) \\ &\quad +(1 - \theta - \beta)(\|x^*\|^2 - \|v^*\|^2). \end{aligned}$$

According to (4.28), we have

$$\lim_{n\to\infty} \Big[\|x_n - x^*\|^2 - \theta\|x_{n-1} - x^*\|^2 - \beta\|x_{n-2} - x^*\|^2\Big]$$

exists and

$$\lim_{n\to\infty} \Big[\|x_n - v^*\|^2 - \theta\|x_{n-1} - v^*\|^2 - \beta\|x_{n-2} - v^*\|^2\Big]$$

exists. This implies that

$$\lim_{n\to\infty} \langle x_n - \theta x_{n-1} - \beta x_{n-2}, x^* - v^*\rangle$$

exists. Now,

$$\begin{aligned} \langle v^* - \theta v^* - \beta v^*, x^* - v^*\rangle &= \lim_{k\to\infty} \langle x_{n_k} - \theta x_{n_k-1} - \beta x_{n_k-2}, x^* - v^*\rangle \\ &= \lim_{n\to\infty} \langle x_n - \theta x_{n-1} - \beta x_{n-2}, x^* - v^*\rangle \\ &= \lim_{j\to\infty} \langle x_{n_j} - \theta x_{n_j-1} - \beta x_{n_j-2}, x^* - v^*\rangle \\ &= \langle x^* - \theta x^* - \beta x^*, x^* - v^*\rangle, \end{aligned}$$

and this yields

$$(1 - \theta - \beta)\|x^* - v^*\|^2 = 0.$$

Since $\beta \leq 0 < 1 - \theta$, we obtain that $x^* = v^*$. Hence, $\{x_n\}$ converges weakly to a point in S_D. This completes the proof.

□

5. Numerical Experiments

In this section, using the following test examples, we compare Algorithm 1 with other known methods in the literature ([46, Algorithm 3.1], [22, Algorithms 3.1, 3.2 and 3.3], [25, Algorithm 1], [26, Algorithm 3.1] and [48, Algorithm 3.1]). We also consider our method with the cases when $\theta = 0 = \beta$ (in this case, Algorithm 1 reduces to [16, Algorithm 3.2]) and $\beta = 0$ (in this case, Algorithm 1 reduces to [15, Algorithm 1]) to show the advantage gained with the introduction of the two-step inertial extrapolation $w_n = x_n + \theta(x_n - x_{n-1}) + \beta(x_{n-1} - x_{n-2})$ in Algorithm 1.

Example 5.1. *This example was also considered in [22]. Let $C = [-1, 1]$ and*

$$Av = \begin{cases} 2v - 1, & v > 1, \\ v^2, & v \in [-1, 1], \\ -2v - 1, & v < -1. \end{cases}$$

Here A is quasi-monotone and Lipschitz continuous with $S_D = \{-1\}$ and $S = \{-1, 0\}$.

Example 5.2. *[22] Let $C = [0, 1]^m$ and $Av = (h_1 v, h_2 v, \cdots, h_m v)$, where $h_i v = v_{i-1}^2 + v_i^2 + v_{i-1}v_i + v_i v_{i+1} - 2v_{i-1} + 4v_i + v_{i+1} - 1,\ i = 1, 2, \cdots, m$, $v_0 = v_{m+1} = 0$. Then A is quasi-monotone.*

The next example shows that our Algorithm 1 still works for the VIP (1.1) when A is not quasi-monotone.

Example 5.3. *[22] Let $C = \{v \in \mathbb{R}^2 : v_1^2 + v_2^2 \leq 1,\ 0 \leq v_1\}$ and $A(v_1, v_2) = (-v_1 e^{v_2}, v_2)$. It can be shown that A is not quasi-monotone with $(1, 0) \in S_D$ and $S = \{(1, 0), (0, 0)\}$.*

We consider the next example in infinite dimensional Hilbert space.

Example 5.4. *Let $H = \{v = (v_1, v_2, \ldots, v_i, \ldots) : \sum_{i=1}^{\infty} |v_i|^2 < +\infty\}$. Let $\alpha, \beta \in \mathbb{R}$ be such that $\beta > \alpha > \frac{\beta}{2} > 0$. Take $C_\alpha = \{v \in H : \|v\| \leq \alpha\}$ and $A_\beta(v) = (\beta - \|v\|)v$. Then A is pseudo-monotone and Lipschitz continuous (but not monotone), and hence quasi-monotone. Furthermore, we have that $S_D = \{0\} = S$.*

We test the above examples by performing the following experiments.

Experiment 1:
In this experiment, we compare Algorithm 1 with other methods that involve only one evaluation of A per iteration (that is, Algorithm 3.1 in [46], Algorithm 1 in [25] and Algorithm 3.1 in [26]), in order to validate the benefits of incorporating the two-step inertial extrapolation.
During the computation for this experiment, we make use of the following:

- Algorithm 1: Take $a_n = \frac{16}{(n+1)^{1.1}}$ and $\delta = 0.24$. Then we can choose $\mu = 0.25$ and $\theta \in \{0, 0.05, 0.1, 0.15\}$. We also take $\beta = \{-1, 0\}$, $\gamma_0 = 0.5$ and $\gamma_1 = 1$.
- Algorithm 3.1 in [46]: Take $\mu = 0.25$, $\lambda_0 = 0.5$ and $\lambda_1 = 1$.
- Algorithm 1 in [25]: Take $\phi = 1.5, \lambda_0 = 0.5$ and $\bar{\lambda} = 15$. According to the author in [25], $\bar{\lambda}$ is given only to ensure the boundedness of the step size. Thus, it makes sense to choose it quite large (see [25, page 389]).
- Algorithm 3.1 in [26]: Take $\delta = 0.5, \sigma = 0.5, \rho = \frac{1}{\sigma}, \lambda_0 = 0.5$ and $\lambda_1 = 1$.

We then take $\text{TOL}_n := \max\left\{\|x_{n+1} - x_n\|^2,\ \|x_n - x_{n-1}\|^2\right\}$ with stopping criterion $\text{TOL}_n < \varepsilon$ for all the algorithms in this experiment, where ε is the predetermined error. In particular, we take $\varepsilon = 10^{-12}$. Note that $\text{TOL}_n = 0$ implies that x_n is a solution of Problem (1.1). We also choose $x_{-1},\ x_0$ and x_1 as follows: $x_{-1} = -0.1$, $x_0 = 0.1$, $x_1 = 0.2$ for Example 5.1; $m = 50$ while $x_{-1},\ x_0$ and x_1 are randomly chosen for Example 5.2; $x_{-1} = (-0.4, -0.4)$, $x_0 = (0.2, 0.1)$, $x_1 = (0.8, 0.5)$ for Example 5.3; $x_{-1} = x_0 = (1, \frac{1}{2}, \frac{1}{4}, \cdots)$, $x_1 = (\frac{4}{5}, \frac{16}{25}, \frac{64}{125}, \cdots)$ for Example 5.4.

Experiment 2:
In this experiment, we compare Algorithm 1 with methods that involve more than one evaluation of A per iteration, in particular, those methods in the literature that were used for solving VIP with quasi-monotone operator. That is, Algorithm 3.1, Algorithm 3.2 and Algorithm 3.3 in [22] (modified Tseng's method, extragradient method and subgradient extragradient method, respectively), and Algorithm 3.1 in [48].

- Algorithm 1: $\text{TOL}_n := \|x_n - x_{n+1}\| + \|x_n - x_{n-1}\|$, $\delta = 0.24$, $\mu = \frac{0.99(1-2\delta)}{2}$, $\theta = 0.99\left(\min\left(\frac{2\delta}{3},\ \frac{1-\mu}{2},\ \frac{0.5-\beta+\delta-\mu}{3}\right)\right)$, $a_n = \frac{1}{(n+1)^{1.1}}$, $\beta = -2$, $\gamma_0 = 0.5$ and $\gamma_1 = 1$.
- Algorithms 3.1, 3.2 and 3.3 in [22]: $\text{TOL}_n := \|y_n - x_n\|/\min\{\gamma_n, 1\}$, $\mu = 0.5$, $a_n = \frac{1}{(n+1)^{1.1}}$ and $\gamma_1 = 1$.
- Algorithm 3.1 in [48]: $\text{TOL}_n := \|x_n - P_C(x_n - Ax_n)\|$, $\alpha = 0.6,\ \gamma = 1.99,\ \lambda_1 = 1,\ \epsilon_n = \frac{10000}{n^{1.001}}$ and $\theta = \frac{10}{n}$.

Unlike in **Experiment 1**, we use different TOL_n for different methods for the comparison in this experiment because these methods have different stopping criteria. However, the choices we have made here have also been used by the authors of these methods (see [22, Section 4] and [48, Section 5]).
Furthermore, we choose x_{-1}, x_0 and x_1 as follows:

- **Example 5.1:** Case 1: $x_{-1} = -0.1$, $x_0 = 0.1$, $x_1 = 0.2$; Case 2: $x_{-1} = -0.2$, $x_0 = 0.2$, $x_1 = 0.6$; Case 3: $x_{-1} = 0.1$, $x_0 = -0.1$, $x_1 = 0.7$ and Case 4: $x_{-1} = -0.9$, $x_0 = 0.9$, $x_1 = 0.1$.
- **Example 5.2:** $m \in \{50, 100, 150, 200\}$ while x_{-1}, x_0 and x_1 are randomly chosen.
- **For Example 5.3:** Case 1: $x_{-1} = (0.11, -0.8)$, $x_0 = (0.2, 0.1)$, $x_1 = (0.8, 0.5)$; Case 2: $x_{-1} = (-0.11, -0.8)$, $x_0 = (0.1, 0.2)$, $x_1 = (0.1, -0.2)$; Case 3: $x_{-1} = (-0.1, -0.2)$, $x_0 = (0.5, -0.5)$, $x_1 = (0.3, 0.6)$ and Case 4: $x_{-1} = (0.1, -0.2)$, $x_0 = (0.3, -0.6)$, $x_1 = (0.5, -0.5)$.
- **Example 5.4:** Case 1: $x_1 = (\frac{2}{3}, \frac{4}{9}, \frac{8}{27}, \cdots)$, $x_{-1} = x_0 = (\frac{2}{3}, \frac{4}{9}, \frac{8}{27}, \cdots)$; Case 2: $x_1 = (\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \cdots)$, $x_{-1} = x_0 = (\frac{2}{3}, \frac{4}{9}, \frac{8}{27}, \cdots)$; Case 3: $x_1 = (\frac{4}{5}, \frac{16}{25}, \frac{64}{125}, \cdots)$, $x_{-1} = x_0 = (1, \frac{1}{2}, \frac{1}{4}, \cdots)$; Case 4: $x_1 = (\frac{3}{4}, \frac{9}{16}, \frac{27}{64}, \cdots)$; $x_{-1} = x_0 = (1, \frac{1}{4}, \frac{1}{9}, \cdots)$.

All the computations are performed using Matlab 2016 (b) which is running on a personal computer with an Intel(R) Core(TM) i5-2600 CPU at 2.30GHz and 8.00 Gb-RAM.
In Tables 1-6, "Iter" and "CPU" mean the CPU time in seconds and the number of iterations, respectively. In Tables 1-2, "Alg." and "Ex." mean Algorithm and Example, respectively.

The numerical results for **Experiment 1** are given in Tables 1-2 and Figures 1-2 while that of **Experiment 2** are given in Tables 3-6 and Figures 3-6.

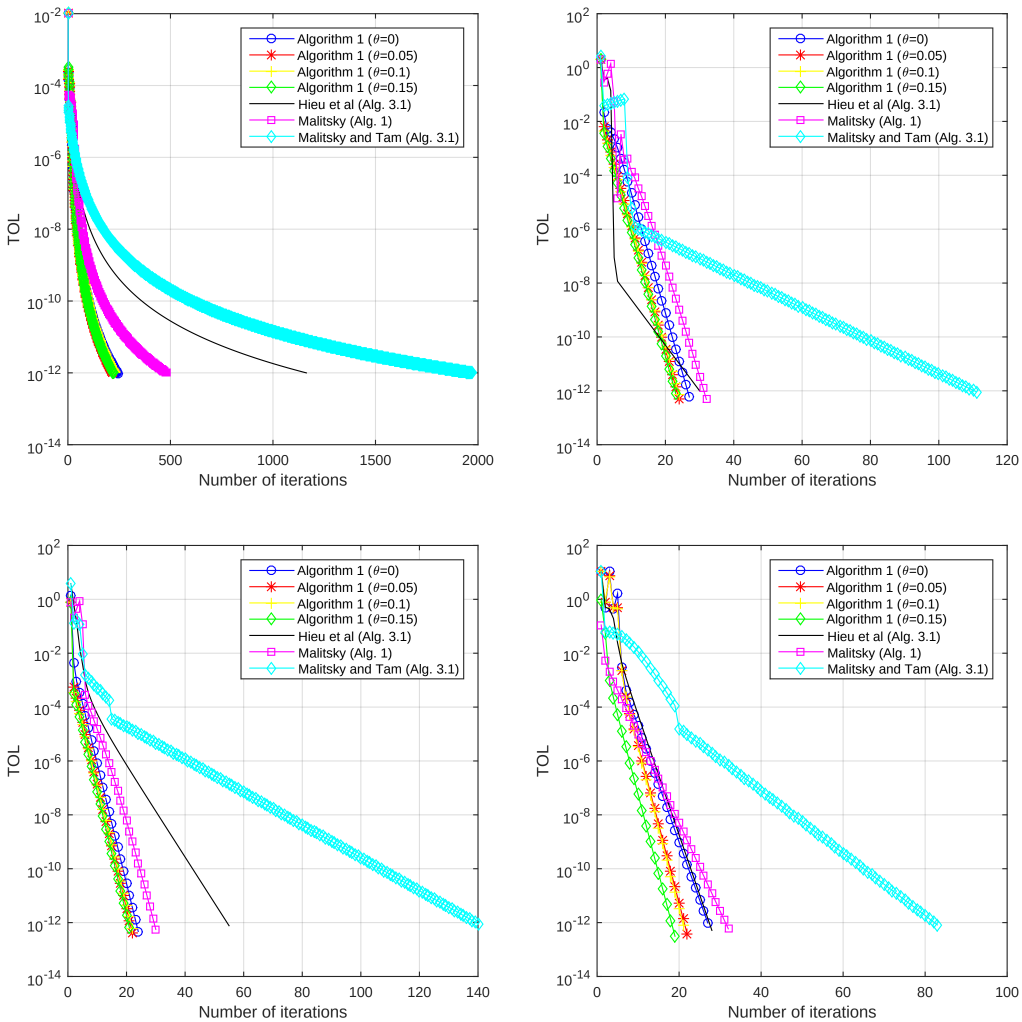

FIGURE 1. The behavior of TOL_n for **Experiment 1 with** $\beta = 0$; $\epsilon = 10^{-12}$:: Top Left: **Example 5.1**; Top Right: **Example 5.2**; Bottom Left: **Example 5.3**; Bottom Right: **Example 5.4**.

Table 1: Comparison of algorithms for **Experiment 1 with** $\beta = 0$; $\epsilon = 10^{-12}$.

		Alg.1 ($\theta = 0$)	Alg.1 ($\theta = 0.05$)	Alg.1 ($\theta = 0.1$)	Alg.1 ($\theta = 0.15$)	Alg.3.1 [46]	Alg.1 [25]	Alg.3.1 [26]
Ex. 5.1	CPU	0.0168	0.0053	0.0050	0.0080	0.3316	0.1199	0.4865
	Iter	242	220	231	221	1163	481	1971
Ex. 5.2	CPU	0.0220	0.0159	0.0017	0.0022	0.1197	0.0842	0.2469
	Iter	27	24	24	23	30	32	111
Ex. 5.3	CPU	0.0143	0.0057	0.0035	0.0044	0.1247	0.0232	0.2423
	Iter	24	22	22	21	55	30	140
Ex. 5.4	CPU	1.2092	0.9814	0.9181	0.8354	1.2002	1.1737	3.6282
	Iter	27	22	21	19	28	32	83

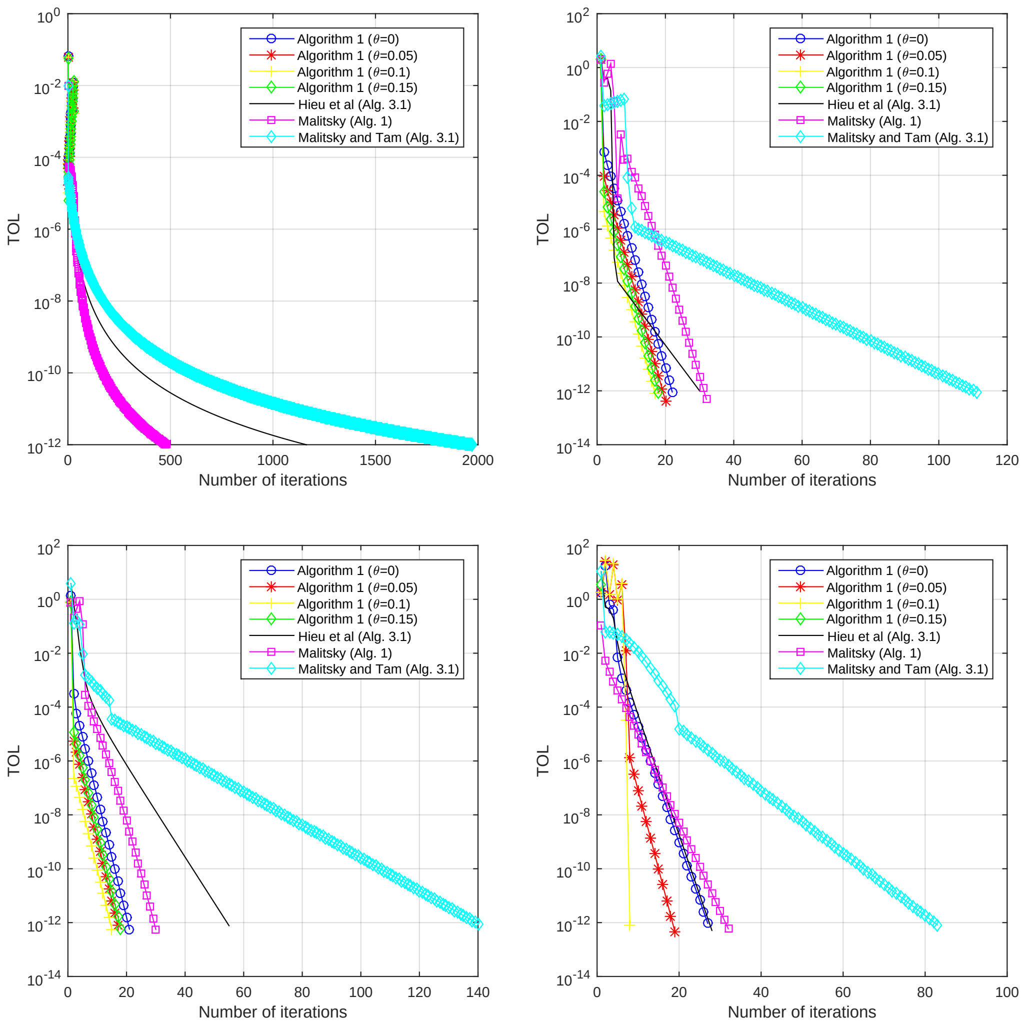

FIGURE 2. The behavior of TOL_n for **Experiment 1 with** $\beta = -1$**;** $\epsilon = 10^{-12}$**:** Top Left: **Example 5.1**; Top Right: **Example 5.2**; Bottom Left: **Example 5.3**; Bottom Right: **Example 5.4**.

Table 2: Comparison of algorithms for **Experiment 1 with** $\beta = -1$**;** $\epsilon = 10^{-12}$**.**

		Alg.1 ($\theta = 0$)	Alg.1 ($\theta = 0.05$)	Alg.1 ($\theta = 0.1$)	Alg.1 ($\theta = 0.15$)	Alg.3.1 [46]	Alg.1 [25]	Alg.3.1 [26]
Ex.	CPU	0.0112	0.0029	0.0017	0.0015	0.3263	0.1110	0.4812
5.1	Iter	26	27	28	29	1163	481	1971
Ex.	CPU	0.0115	0.0021	0.0022	0.0018	0.1070	0.0821	0.2303
5.2	Iter	22	20	17	18	30	32	111
Ex.	CPU	0.0116	0.0107	0.0012	0.0010	0.1141	0.0213	0.2305
5.3	Iter	21	17	15	18	55	30	140
Ex.	CPU	1.0272	0.8911	0.3867	0.1079	1.2260	1.1932	3.6216
5.4	Iter	27	19	8	2	28	32	83

Table 3: Comparison of algorithms for **Experiment 2 (Example 5.1)**; $\epsilon = 10^{-6}$.

Algorithms	Case 1		Case 2		Case 3		Case 4	
	CPU	Iter	CPU	Iter	CPU	Iter	CPU	Iter
Algorithm 1	0.0010	389	0.0081	487	0.0020	655	0.0011	355
Algorithm 3.1 in [22]	0.0289	1002	0.0247	1005	0.0264	1005	0.0270	996
Algorithm 3.2 in [22]	0.0833	1008	0.0778	1019	0.0478	1023	0.0241	1001
Algorithm 3.3 in [22]	0.0400	998	0.0694	1009	0.0785	1013	0.0715	992
Algorithm 3.1 in [48]	0.1331	2177	0.1164	2120	0.1487	1439	0.0965	1529

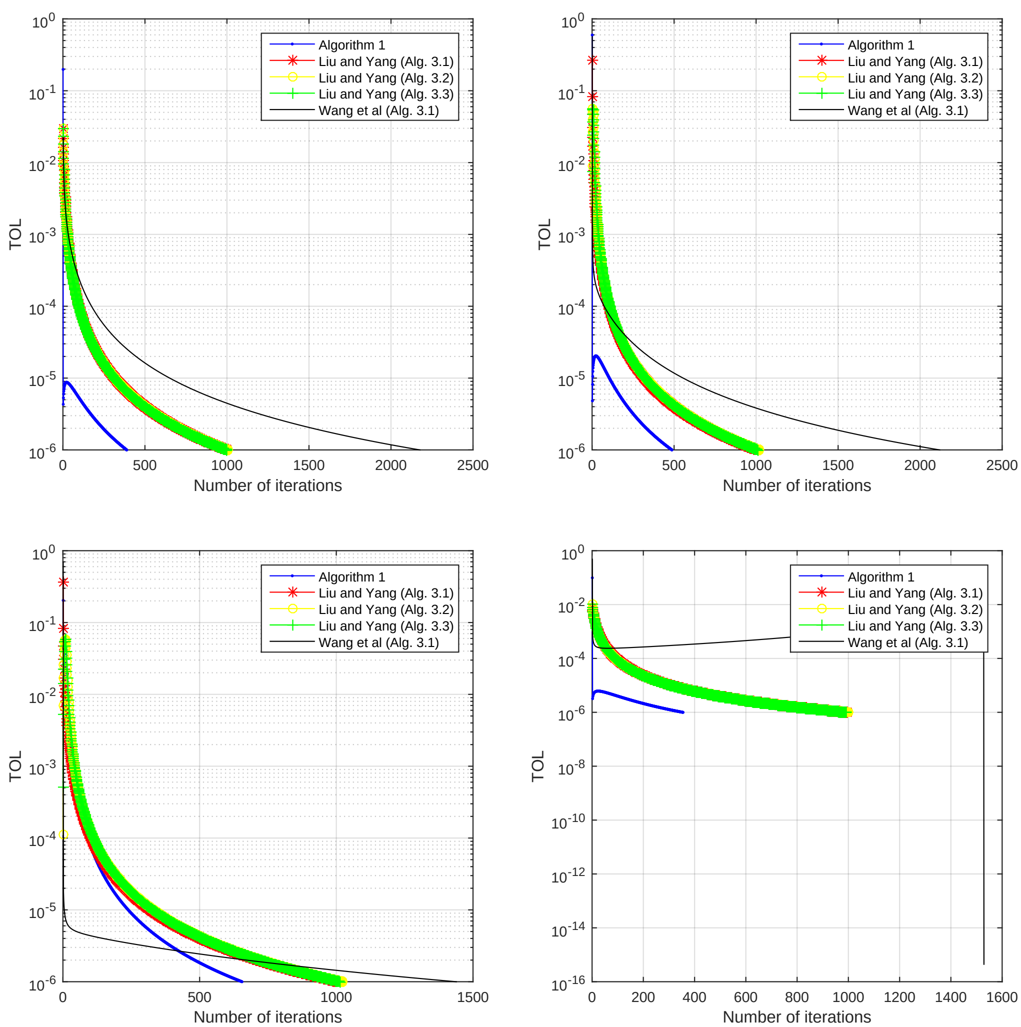

FIGURE 3. The behavior of TOL_n for **Experiment 2 (Example 5.1)**; $\epsilon = 10^{-6}$: Top Left: **Case 1**; Top Right: **Case 2**; Bottom left: **Case 3**; Bottom Right: **Case 4**.

Remark 5.5.

Table 4: Comparison of algorithms for **Experiment 2 (Example 5.2)**; $\epsilon = 10^{-7}$.

Algorithms	$m = 50$		$m = 100$		$m = 150$		$m = 200$	
	CPU	Iter	CPU	Iter	CPU	Iter	CPU	Iter
Algorithm 1	0.0014	12	0.0020	13	0.0012	9	0.0009	10
Algorithm 3.1 in [22]	0.1061	76	0.1096	80	0.1054	66	0.1046	70
Algorithm 3.2 in [22]	0.1084	81	0.1126	87	0.1085	75	0.1079	82
Algorithm 3.3 in [22]	0.1061	66	0.1108	62	0.1091	64	0.1096	53
Algorithm 3.1 in [48]	0.1039	53	0.1078	59	0.1031	44	0.1070	50

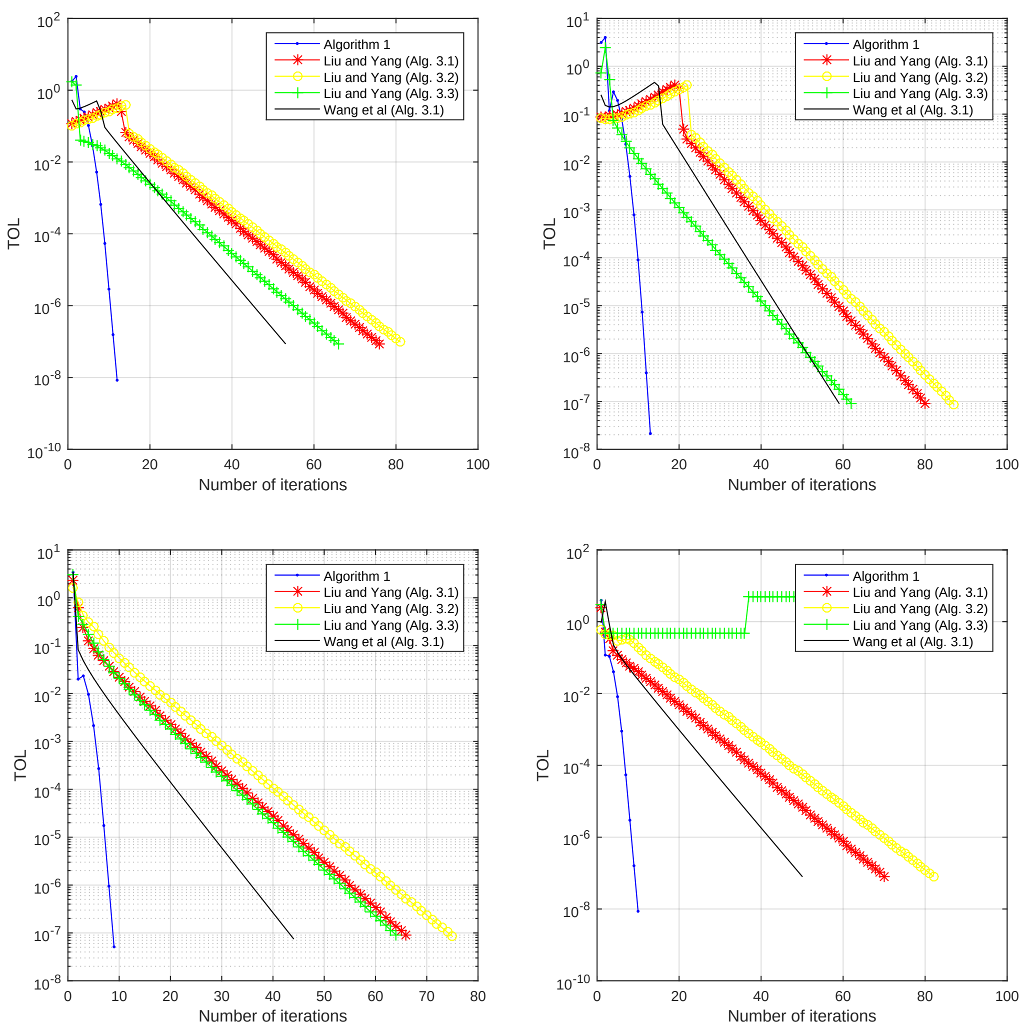

FIGURE 4. The behavior of TOL_n for **Experiment 2 (Example 5.2)**; $\epsilon = 10^{-7}$: Top Left: $m = 50$; Top Right: $m = 100$; Bottom left: $m = 150$; Bottom Right: $m = 200$.

(i) In our convergence analysis, we assume that $\delta \in (0, \frac{1}{4})$, $\mu \in \left(\delta, \frac{1-2\delta}{2}\right)$ and $0 \leq \theta < \min\left\{\frac{1-\mu}{2}, \frac{2\delta}{3}, \frac{\frac{1}{2}-\beta+\delta-\mu}{3}\right\}$. This means that μ and θ both depend on the choice of δ. In particular, the closer δ is to $\frac{1}{4}$, the larger the interval of the inertial parameter θ but the smaller

Table 5: Comparison of algorithms for **Experiment 2 (Example 5.3)**; $\epsilon = 10^{-5}$

Algorithms	Case 1		Case 2		Case 3		Case 4	
	CPU	Iter	CPU	Iter	CPU	Iter	CPU	Iter
Algorithm 1	0.0010	7	0.0023	15	0.0011	10	0.0011	9
Algorithm 3.1 in [22]	0.1027	48	0.1085	73	0.1039	60	0.1067	56
Algorithm 3.2 in [22]	0.1052	52	0.1058	79	0.1062	65	0.1090	61
Algorithm 3.3 in [22]	0.1045	39	0.1042	37	0.1065	41	0.1076	37
Algorithm 3.1 in [48]	0.1374	97	0.1184	82	0.1175	108	0.1167	92

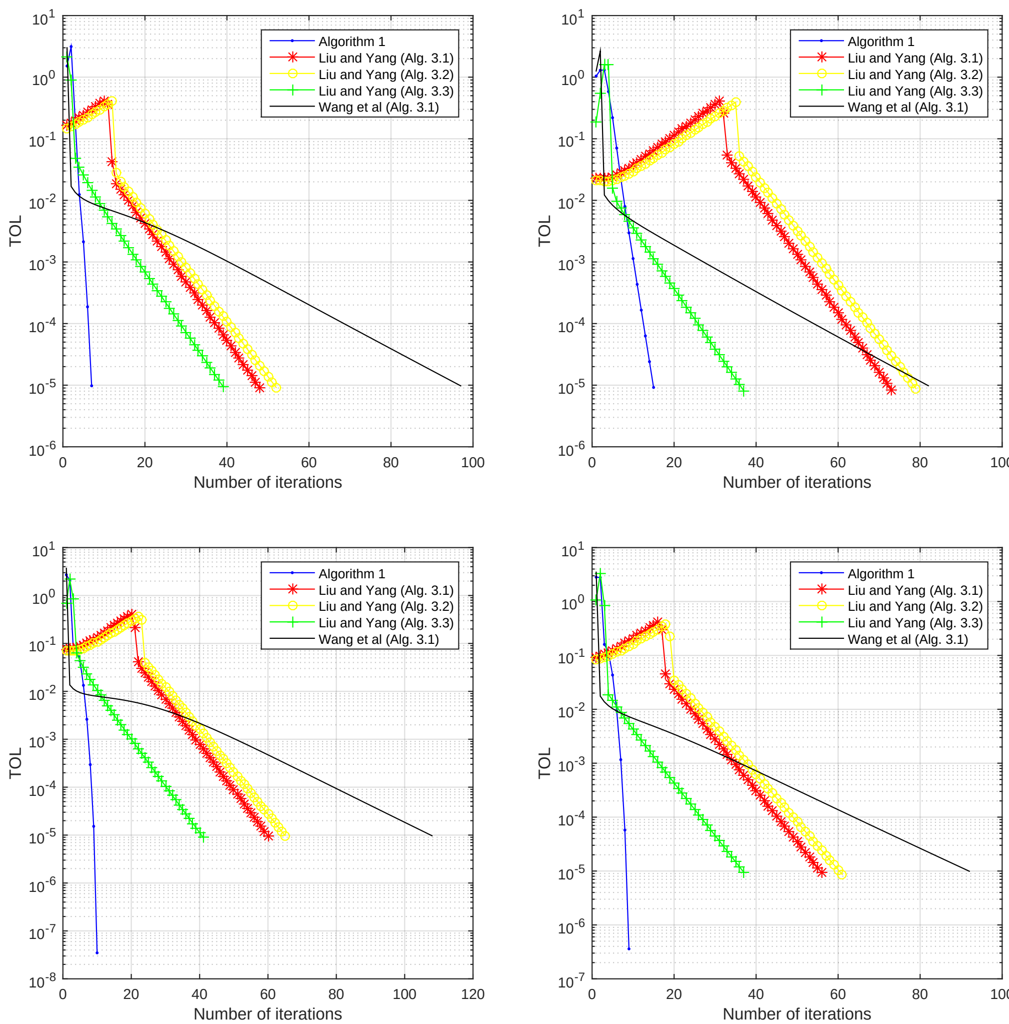

FIGURE 5. The behavior of TOL_n for **Experiment 2 (Example 5.3)**; $\epsilon = 10^{-5}$: Top Left: **Case 1**; Top Right: **Case 2**; Bottom left: **Case 3**; Bottom Right: **Case 4**.

the interval of μ. On the other hand, the closer δ is to zero, the larger the interval of μ but the smaller the interval of θ.

(ii) Our numerical experiments in Section 5 show the benefits gained (in terms of number of iterations and CPU time) over [46, Algorithm 3.1], [22, Algorithms 3.1, 3.2 and 3.3], [25, Algorithm 1], [26, Algorithm 3.1] and [48, Algorithm 3.1], by introducing the inertial parameters θ and β

Table 6: Comparison of algorithms for **Experiment 2 (Example 5.4)**; $\epsilon = 10^{-3}$.

Algorithms	Case 1		Case 2		Case 3		Case 4	
	CPU	Iter	CPU	Iter	CPU	Iter	CPU	Iter
Algorithm 1	0.4823	11	0.3030	7	0.4608	12	0.4986	12
Algorithm 3.1 in [22]	1.1965	29	1.2443	30	1.1143	27	1.1004	27
Algorithm 3.2 in [22]	1.7106	34	1.5688	31	1.8092	35	1.8236	35
Algorithm 3.3 in [22]	13.5817	25	1.7173	25	1.9735	25	14.3086	25
Algorithm 3.1 in [48]	25.4557	25	1.4581	25	1.5841	25	25.3428	25

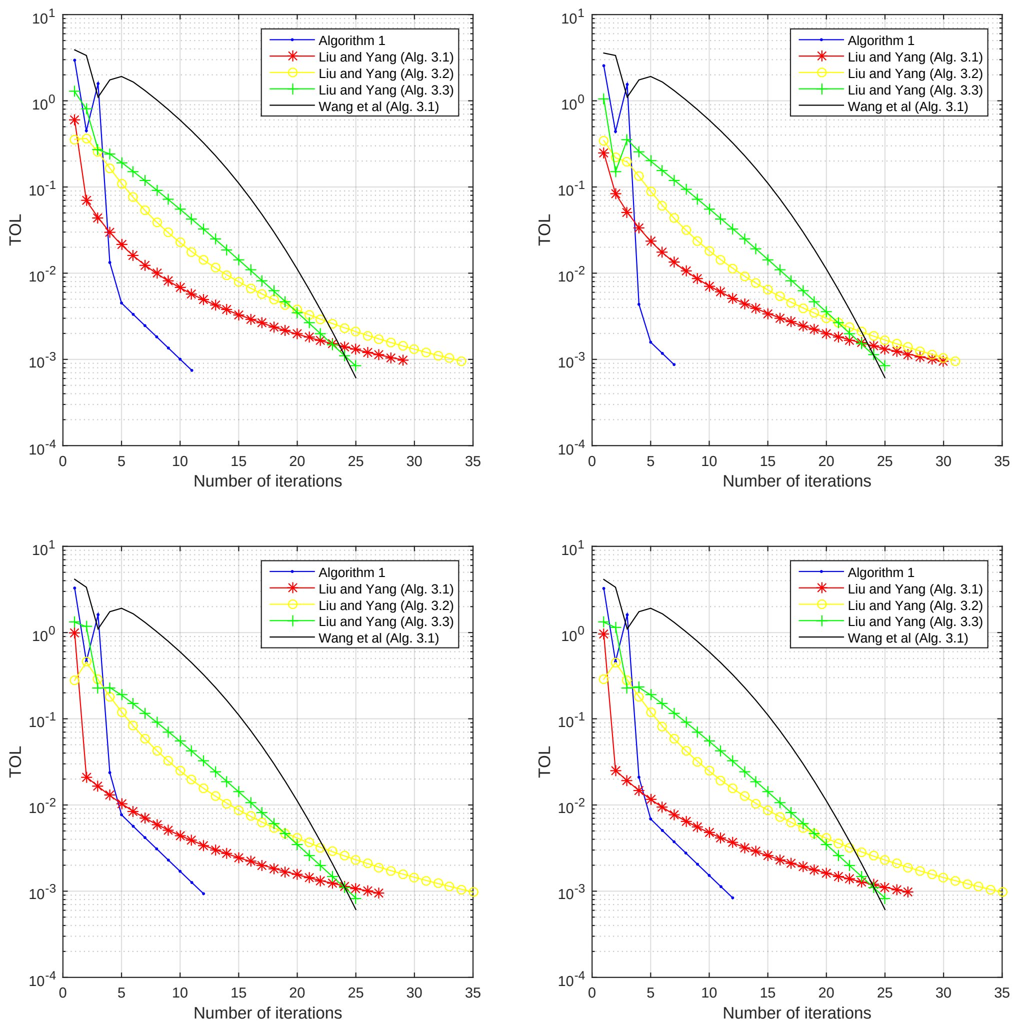

FIGURE 6. The behavior of TOL_n for **Experiment 2 (Example 5.4)**; $\epsilon = 10^{-3}$: Top Left: **Case 1**; Top Right: **Case 2**; Bottom left: **Case 3**; Bottom Right: **Case 4**.

in our proposed Algorithm 1.

For instance, in **Experiment 1**, we considered different choices of θ and β including the choices $\theta = 0 = \beta$ and $\beta = 0$. The numerical results from Tables 1-2 and Figures 1-2 validate the advantage brought by the two-step inertial extrapolation.

6. Final Remarks

In this paper, we have presented a forward-reflected-backward splitting method with two-step inertial extrapolation and self-adaptive step sizes to solve variational inequalities with quasi-monotone operators in real Hilbert spaces. We give weak convergence analysis of our method under some standard conditions. Numerical illustration showed that our method is computationally cheaper than other related methods in the literature. As part of our future projects, we shall consider a relaxed version of our proposed method with correction term to solve variational inequalities in the setting of quasi-monotone.

Data Availability

Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

Statements and Declarations

The authors declare that they have no competing interests.

References

[1] J.-P. Aubin and I. Ekeland. *Applied Nonlinear Analysis*. Wiley, New York, 1984.

[2] C. Baiocchi and A. Capelo. *Variational and quasivariational inequalities. Applications to free-boundary problems*. Wiley, New York, 1984.

[3] L.-C. Ceng, N. Hadjisavvas, and N.-C. Wong. Strong convergence theorem by a hybrid extragradient-like approximation method for variational inequalities and fixed point problems. *Journal of Global Optimization*, 46:635–646, 2010.

[4] Y. Censor, A. Gibali, and S. Reich. Extensions of korpelevich's extragradient method for the variational inequality problem in euclidean space. *Optimization*, 61:1119–1132, 2011.

[5] Y. Censor, A. Gibali, and S. Reich. Strong convergence of subgradient extragradient methods for the variational inequality problem in hilbert space. *Optimization Method and Software*, 26:827–845, 2011.

[6] Y. Censor, A. Gibali, and S. Reich. The subgradient extragradient method for solving variational inequalities in hilbert space. *Journal of Optimization Theory and Applications*, 148:318–335, 2011.

[7] X. Chang, S. Liu, Z. Deng, and S. Li. An inertial subgradient extragradient algorithm with adaptive stepsizes for variational inequality problems. *Optimization Method and Software*, 4:1507–1526, 2022.

[8] P. L. Combettes and L. E. Glaudin. Quasi-nonexpansive iterations on the affine hull of orbits: from mann's mean value algorithm to inertial methods. *SIAM Journal on Optimization*, 4:2356–2380, 2017.

[9] Q.-L. Dong, J. Z. Huang, X. H. Li, Y. J. Cho, and T. M. Rassias. Mikm: multi-step inertial krasnosel'skii–mann algorithm and its applications. *Journal of Global Optimization*, 73:801–824, 2019.

[10] J. Fan, L. Liu, and X. Qin. A subgradient extragradient algorithm with inertial effects for solving strongly pseudomonotone variational inequalities. *Optimization*, 69:2199–2215, 2020.

[11] G. Fichera. Sul pproblem elastostatico di signorini con ambigue condizioni al contorno. *Atti della Accademia Nazionale dei Lincei, Classe di Scienze Fisiche, Matematiche e Naturali, Rendiconti Lincei Matematica E Applicazioni*, 34:138–142, 1963.

[12] A. Gibali and D. V. Thong. A new low-cost double projection method for solving variational inequalities. *Optimization and Engineering*, 21:1613–1634, 2020.

[13] R. Glowinski, J.-L. Lions, and R. Trémoliéres. *Numerical Analysis of Variational Inequalities*. Springer, Amsterdam, 1981.

[14] B.-S. He, Z.-H. Yang, and X.-M. Yuan. An approximate proximal-extragradient type method for monotone variational inequalities. *Journal of Mathematical Analysis and Applications*, 300:362–374, 2004.

[15] C. Izuchukwu, Y. Shehu, and J.-C. Yao. New inertial forward-backward type for variational inequalities with quasi-monotonicity. *Journal of Global Optimization*, 84:441–464, 2022.

[16] C. Izuchukwu, Y. Shehu, and J.-C. Yao. A simple projection method for solving quasimonotone variational inequality problems. *Optimization and Engineering*, 24:915–938, 2023.

[17] E. N. Khobotov. Modification of the extragradient method for solving variational inequalities and certain optimization problems. *USSR Computational Mathematics and Mathematical Physics*, 27:120–127, 1987.

[18] D. Kinderlehrer and G. Stampacchia. *An Introduction to Variational Inequalities and Their Applications*. Academic Press, New York, 1980.

[19] G. M. Korpelevich. An extragradient method for finding saddle points and for other problems. *Ekonomika i matematicheskie metody*, 12:747–756, 1976.

[20] R. Kraikaew and S. Saejung. Strong convergence of the halpern subgradient extragradient method for solving variational inequalities in hilbert spaces. *Journal of Optimization Theory and Applications*, 163:399–412, 2014.

[21] J. Liang. *Convergence rates of first-order operator splitting methods*. PhD thesis, Normandie Université; GREYC CNRS UMR 6072, 2016.

[22] H. Liu and J. Yang. Weak convergence of iterative methods for solving quasimonotone variational inequalities. *Computational Optimization and Applications*, 77:491–508, 2020.

[23] P.-E. Maingé and M.-L. Gobinddass. Convergence of one-step projected gradient methods for variational inequalities. *Journal of Optimization Theory and Application*, 171:146–168, 2016.

[24] Y. Malitsky. Projected reflected gradient methods for monotone variational inequalities. *SIAM Journal on Optimization*, 25:502–520, 2015.

[25] Y. Malitsky. Golden ratio algorithms for variational inequalities. *Mathematical Programming*, 184:383–410, 2020.

[26] Y. Malitsky and M. K. Tam. A forward-backward splitting method for monotone inclusions without cocoercivity. *SIAM Journal on Optimization*, 30:1451–1472, 2020.

[27] Y. V. Malitsky and V. Semenov. An extragradient algorithm for monotone variational inequalities. *Cybernetics and Systems Analysis*, 50:271–277, 2014.

[28] P. Marcotte. Applications of khobotov's algorithm to variational and network equlibrium problems. *Information Systems and Operational Research*, 29:258–270, 1991.

[29] M. A. Noor. Extragradient methods for pseudomonotone variational inequalities. *Journal of Optimization Theory and Applications*, 117:475–488, 2003.

[30] B. T. Polyak. *Introduction to optimization*. Optimization Software, Publications Division, New York, 1987.

[31] C. Poon and J. Liang. *In Advances In Neural Information Processing Systems*. Trajectory of Alternating Direction Method of Multipliers and Adaptive Acceleration, 2019.

[32] C. Poon and J. Liang. *Geometry of First-Order Methods and Adaptive Acceleration*. arXiv:2003.03910, 2020.

[33] L. Popov. A modification of the arrow-hurwicz method for finding saddle points. *Mathematical Notes*, 28:845–848, 1980.

[34] Salahuddin. The extragradient method for quasi-monotone variational inequalities. *Optimization*, 71:2519–2528, 2022.

[35] Y. Shehu and O. S. Iyiola. Projection methods with alternating inertial steps for variational inequalities: Weak and linear convergence. *Applied Numerical Mathematics*, 157:315–337, 2020.

[36] Y. Shehu, O. S. Iyiola, and S. Reich. A modified inertial subgradient extragradient method for solving variational inequalities. *Optimization and Engineering*, 1:1–29, 2022.

[37] G. Stampacchia. *"variational Inequalities", in: Theory and Applications of Monotone Operators*. Proceedings of the NATO Advanced Study Institute, Italy, 1968.

[38] G.-j. Tang, Z. Wan, and N.-J. Huang. Strong convergence of a projection-type method for mixed variational inequalities in hilbert spaces. *Numerical Functional Analysis and Optimization*, 39:1103–1119, 2018.

[39] D. V. Thong, A. Gibali, and P. T. Vuong. An explicit algorithm for solving monotone variational inequalities. *Applied Numerical Mathematics*, 171:408–425, 2022.

[40] D. V. Thong and D. V. Hieu. Inertial extragradient algorithms for strongly pseudomonotone variational inequalities. *Journal of Computational and Applied Mathematics*, 341:80–98, 2018.

[41] D. V. Thong, D. V. Hieu, and T. M. Rassias. Self adaptive inertial subgradient extragradient algorithms for solving pseudomonotone variational inequality problems. *Optimization Letters*, 14:115–144, 2020.

[42] D. V. Thong, X.-H. Li, Q.-L. Dong, Y. J. Cho, and T. M. Rassias. An inertial popov's method for solving pseudomonotone variational inequalities. *Optimization Letters*, 15:757–777, 2021.

[43] D. V. Thong, J. Yang, Y. J. Cho, and T. M. Rassias. Explicit extragradient-like method with adaptive stepsizes for pseudomonotone variational inequalities. *Optimization Letters*, 15:2181–2199, 2021.

[44] M. Tian and M. Tong. Self-adaptive subgradient extragradient method with inertial modification for solving monotone variational inequality problems and quasi-nonexpansive fixed point problems. *Journal of Inequalities and Applications*, 7:1–19, 2019.

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

[46] D. Van Hieu, P. K. Anh, and L. D. Muu. Modified forward–backward splitting method for variational inclusions. *4OR-A Quarterly Journal of Operations Research*, 19:127–151, 2021.

[47] P. T. Vuong. On the weak convergence of the extragradient method for solving pseudomonotone variational inequalities. *Journal of Optimization Theory and Application*, 176:399–409, 2018.

[48] Z.-b. Wang, X. Chen, J. Yi, and Z. y. Chen. Inertial projection and contraction algorithms with larger step sizes for solving quasimonotone variational inequalities. *Journal of Global Optimization*, pages 1–24, 2022.

[49] Z.-b. Wang, Z. y. Chen, Y. bin Xiao, and C. Zhang. A new projection-type method for solving multi-valued mixed variational inequalities without monotonicity. *Applicable Analysis*, 99:1453–1466, 2020.

[50] J. Yang. Self-adaptive inertial subgradient extragradient algorithm for solving pseudomonotone variational inequalities. *Applicable Analysis*, 100:1067–1078, 2021.

[51] J. Yang and H. Liu. A modified projected gradient method for monotone variational inequalities. *Journal of Optimization Theory and Application*, 179:197–211, 2018.

[52] J. Yang, H. Liu, and Z. Liu. Modified subgradient extragradient algorithms for solving monotone variational inequalities. *Optimization*, 67:2247–2258, 2018.

[53] M. Ye and Y. He. A double projection method for solving variational inequalities without monotonicity. *Computational Optimization and Applications*, 60:141–150, 2015.