OPTIMIZATION ERUDITORUM
Volume 1 (2024), No. 1, 56–74
https://doi.org/10.69829/oper-024-0101-ta05
Tulipa Opera Scholarum

SCALED FORWARD-BACKWARD ALGORITHM AND THE MODIFIED SUPERIORIZED VERSION FOR SOLVING THE SPLIT MONOTONE VARIATIONAL INCLUSION PROBLEM

MINGXIA ZHENG[1] AND YANNI GUO[1,*]

[1] *College of Science, Civil Aviation University of China, Tianjin 300300, China*

Abstract. In this paper, we propose an inexact scaled forward-backward algorithm to solve the split monotone variational inclusion problem in a real Hilbert space and prove the strong convergence of it under appropriate conditions. Based on this, we discuss the bounded perturbation resilience of the exact algorithm for introducing the corresponding superiorized version and the superiorization algorithm with restarted perturbations. The numerical experiments illustrate that the proposed algorithms perform well and the superiorization version with restarted perturbations has advantage in decreasing the number of the iterations.

Keywords. Forward-backward algorithm; Split monotone variational inclusion problem; Bounded perturbation resilience; Superiorization.

1. Introduction

Let H_1 and H_2 be real Hilbert spaces. Let $B_1 : H_1 \to 2^{H_1}$, $B_2 : H_1 \to 2^{H_2}$ be multi-valued maximal monotone operators. Let $C_1 : H_1 \to H_1$, $C_2 : H_2 \to H_2$ be ν_1-inverse strongly monotone and ν_2-inverse strongly monotone operators, respectively. Let $A : H_1 \to H_2$ be a bounded linear operator from H_1 to H_2. We consider the following split monotone variational inclusion problem (SMVIP) in this paper: find $x^* \in H_1$ such that

$$0 \in (B_1 + C_1)x^* \text{ and } y^* = Ax^* \in H_2 \text{ such that } 0 \in (B_2 + C_2)y^*. \tag{1.1}$$

If $C_1 = C_2 = 0$, (1.1) reduces to the following split variational inclusion problem: find $x^* \in H_1$ such that

$$0 \in B_1x^* \text{ and } y^* = Ax^* \in H_2 \text{ such that } 0 \in B_2y^*. \tag{1.2}$$

Further, if we have $B_1 = N_C$ and $B_2 = N_Q$, (1.2) simplifies to split feasibility problem, which was originally formulated by Censor and Elfving [1] in 1994 in the following form:

$$\text{find } x^* \in C \text{ such that } y^* = Ax^* \in Q,$$

where N_C and N_Q are the normal cones of the nonempty closed and convex sets $C \subseteq H_1$ and $Q \subseteq H_2$, respectively. In fact, SMVIP provides a unified framework for issues like split feasibility problem, split minimization problem, split variational inequality problem, split variational inclusion problem and monotone inclusion problem and has found wide applications in real-world such as compressed sensing and image recovery([2]), intensity-modulated radiation therapy treatment planning([3]), signal processing and image reconstruction([4]), among others.

Moudafi [5] introduced SMVIP in 2011 and proposed the following iterative algorithm:

$$x_{n+1} = U(x_n + \gamma A^*(T - I)Ax_n),\ x_0 \in H_1, \tag{1.3}$$

*Corresponding author.
E-mail addresses: ynguo@amss.ac.cn (Y. N. Guo), and 2021061032@cauc.edu.cn (M. X. Zheng)
2020 Mathematics Subject Classification: 47H05;47H10;49J45.
Accepted: May 18, 2024.

where $U = J_\lambda^{B_1}(I - \lambda C_1)$, $T = J_\lambda^{B_2}(I - \lambda C_2)$, $J_\lambda^{B_1}$ and $J_\lambda^{B_2}$ are the resolvents of B_1 and B_2, respectively, $\lambda \in (0, 2\min\{v_1, v_2\})$, $\gamma \in (0, \frac{1}{\|A^*A\|})$, $A^* : H_2 \to H_1$ is the adjoint operator of A. The weak convergence of this algorithm was addressed. In 2020, Zhao et al.[6] proposed a proximity algorithm for solving (1.1) and obtained the strong convergence by introducing a contraction and a strongly positive bounded linear operator. The iterative scheme is

$$\begin{cases} y_n = x_n + \tau A^*(T - I)Ax_n,\ x_0 \in H_1, \\ x_{n+1} = U(\gamma\alpha_n f(x_n) + (I - \alpha_n F)y_n), n \geq 0, \end{cases}$$

where $U = J_{\lambda_1}^{B_2}(I - \lambda_1 C_1)$, $T = J_{\lambda_2}^{B_2}(I - \lambda_2 C_2)$, $\lambda_1 \in (0, 2v_1)$ and $\lambda_2 \in (0, 2v_2)$. $f : H_1 \to H_1$ is a μ-contraction, $\mu \in [0, 1)$. $F : H_1 \to H_1$ is a strongly positive bounded linear operator with coefficient μ_1.

On the other hand, to accelerate the convergence speed of algorithm (1.3), Yao et al. [8] incorporated well-established inertial extrapolation techniques and resulted in the following algorithm with inertial extrapolations and self-adaptive step sizes

$$\begin{cases} w_n = x_n + \theta_n(x_n - x_{n-1}),\ x_0, x_1 \in H_1, \\ x_{n+1} = U(w_n + \gamma_n A^*(T - I)Aw_n), n \geq 1. \end{cases} \tag{1.4}$$

They proved that the sequence generated by algorithm (1.4) converges weakly to a solution of (1.1) under some mild assumptions. Very recently, Zhou et al. [7] introduced adaptive hybrid steepest descent algorithm involving an inertial term and proved that the generated sequence converges strongly to a point of the solution set of problem (1.1). Moreover, Izuchukwu et al. [9] proposed an algorithm to address problem (1.1) without the need to assume that C_1 and C_2 are inverse strongly monotone operators. The algorithm is as follows:

$$\begin{cases} w_n = x_n + \theta_n(x_n - x_{n-1}),\ x_0, x_1 \in H_1, \\ y_n = J_{\lambda_2}^{B_2}(I - \lambda_2 C_2)Aw_n, \\ v_n = Aw_n - y_n - \lambda_2(C_2 Aw_n - C_2 y_n), \\ \hat{t}_n = Aw_n - \hat{\alpha}\eta_n v_n, \\ b_n = w_n + \gamma_n A^*(\hat{t}_n - Aw_n), \\ u_n = J_{\lambda_1}^{B_1}(I - \lambda_1 C_1)b_n, \\ r_n = b_n - u_n - \lambda_1(C_1 b_n - C_1 u_n), \\ x_{n+1} = (1 - \rho_n)w_n + \rho_n(b_n - \alpha\beta_n r_n), n \geq 1, \end{cases} \tag{1.5}$$

where

$$\eta_n = \begin{cases} \frac{\langle Aw_n - y_n, v_n\rangle}{\|v_n\|^2}, & v_n \neq 0, \\ 0, & v_n = 0, \end{cases} \quad \text{and} \quad \beta_n = \begin{cases} \frac{\langle b_n - u_n, r_n\rangle}{\|r_n\|^2}, & r_n \neq 0, \\ 0, & r_n = 0. \end{cases}$$

Under certain conditions on the parameters, they proved that the sequence generated by the algorithm possesses weak convergence. More articles on solving (1.1) can be found in references [10]-[13]. Another way of accelerating strategies is scaling techniques for determining the descent direction, which enables algorithms to better adapt to variations in data at different scales when dealing with problems such as image restoration and machine learning [14]-[16].

Besides that, Censor [17] proposed a superiorization method for solving constrained optimization problems with large data, which can improve the convergence speed of an algorithm to some extent. Superiorization method is a heuristic method, which introduces a perturbation in each of its iterations and aims to obtain lower values of the objective function. At the same time, the sequence generated by this superiorization version converges to a feasible solution of the optimization problem by means of that the basic algorithm (that is the algorithm without perturbations) possesses the bounded perturbation resilience (see Definition 3.4 in Section 3.2 below). In addition, superiorization method can get more information than the basic algorithm by defining a cost (or target) function. Up to now, superiorization method has found wild applications in real world, such as computer tomography scanning [18]-[20], radiotherapy inverse treatment planning [21]-[22], medical image restoration ([23]), convex feasibility problems [24]-[25], etc.

The perturbations introduced in the superiorization method usually have the form $\beta_n v_n,\ \ n = 1, 2, \cdots$, where $\{v_n\}$ is a bounded sequence and $\{\beta_n\}$ is a summable sequence of positive numbers. That the superiorization algorithm performs well in practice when β_n is set to ac^n where $a > 0$, $c \in (0, 1)$ are constants. But the perturbations will become negligible due to that $\beta_n = ac^n$ will quickly decrease to zero as the number of iteration steps increases. This situation is undesirable because it may affect the accelaraion of the algorithm. The permission to restart the perturbation and to maintain the summability of $\{\beta_n\}$ in the superiorized algorithm will increase the computational efficiency [22].

Inspired by the above results, we propose an inexact scaled forward-backward algorithm with inertial term for solving problem (1.1) and analyze the strong convergence of the generated sequence. Then we discuss the bounded perturbation resilience of the exact version of it. As a consequence, we show the superiorization version of the exact algorithm and the superiorization algorithm with restarted perturbations.

The rest of this paper is organized as follows. In Section 2, we review some background on monotone operators and convex analysis. In Section 3, we propose an inexact algorithm for solving problem (1.1) and prove the strong convergence of the generated sequence. After discussing the bounded perturbation resilience of the exact algorithm (3.23), we introduce the pseudocode of the superiorized version of algorithm (3.23) and the superiorized version with restarted perturbations. Finally, in Section 4, we illustrate the validity of the proposed algorithms and the performance of them under different values of the parameters by two numerical examples.

2. Preliminaries

In this section, we recall some notations, definitions and lemmas used in this paper. Let H be a real Hilbert space with the inner product $\langle \cdot, \cdot \rangle$ and the induced norm $\|\cdot\|$. Let $\{x_n\} \subset H$ be a sequence. That $\{x_n\}$ converges strongly to x^* is denoted by $\to$ and $\{x_n\}$ converges weakly to x^* is denoted by $\rightharpoonup$. The set of all weak cluster points of $\{x_n\}$ is denoted by $\omega_w(x_n) = \{x \in H | x_{n_k} \rightharpoonup x \text{ and } \{x_{n_k}\} \subset \{x_n\}\}$. Let $T : H \to H$ be a nonlinear operator and Fix(T) be the fixed set of T. That is Fix(T) $:= \{x \in H : Tx = x\}$. Denote by I the identity operator on H.

Definition 2.1. ([26])
(i) $T : H \to H$ is said to be non-expansive if $\forall x, y \in H$, it has

$$\|Tx - Ty\| \leq \|x - y\|.$$

(ii) $T : H \to H$ is said to be L-Lipschitz continuous with $L \geq 0$ if $\forall x, y \in H$, it has

$$\|Tx - Ty\| \leq L\|x - y\|.$$

T is said to be contractive if $0 \leq L < 1$.
(iii) $T : H \to H$ is said to be firmly non-expansive if $\forall x, y \in H$, it has

$$\langle Tx - Ty\rangle \geq \|Tx - Ty\|^2.$$

In particular, a firmly non-expansive operator is $\frac{1}{2}$-averaged.
(iv) $T : H \to H$ is said to be v-inverse strongly monotone (v-ism) with $v > 0$, if $\forall x, y \in H$, it has

$$\langle x - y, Tx - Ty\rangle \geq v\|Tx - Ty\|^2.$$

Definition 2.2. ([5]) Let $A : H \to 2^H$ be a multi-valued operator.
(i) A is said to be monotone, if $\forall x, y \in H$, $u \in Ax$, $v \in Ay$, it has

$$\langle u - v, x - y \rangle \geq 0.$$

(ii) A is said to be maximal monotone, if it is monotone and the graph

$$\text{Graph}(A) = \{(x, y) \in H \times H : y \in Ax\}$$

is not properly contained in the graph of any other monotone operator. Furthermore, A is maximal monotone if and only if for every $(u, v) \in H \times H$,

$$(u, v) \in \text{Graph}(A) \Leftrightarrow (\forall (x, y) \in \text{Graph}(A)) \;\; \langle x - u, y - v \rangle \geq 0.$$

(iii) the resolvent of A is defined as

$$J_\lambda^A = (I + \lambda A)^{-1}, \; \lambda > 0.$$

Lemma 2.3. ([5]) *Let H_1 and H_2 be real Hilbert spaces. Let $B_i : H_i \to 2^{H_i}$ be maximal monotone operator, $C_i : H_i \to H_i$ be v_i-inverse strongly monotone operator for each $i \in \{1, \; 2\}$. Let $A : H_1 \to H_2$ be a bounded linear operator. Denote the solution set of problem (1.1) by Ω. Then it is easy to check that $\forall \lambda > 0$,*

$$x^* \in \Omega \Leftrightarrow x^* \in \mathit{Fix}(J_\lambda^{B_1}(I - \lambda C_1)) \text{ and } Ax^* \in \mathit{Fix}(J_\lambda^{B_2}(I - \lambda C_2)).$$

Furthermore, if $\lambda \in (0, 2min\{v_1, v_2\})$, $J_\lambda^{B_1}(I - \lambda C_1)$, $J_\lambda^{B_2}(I - \lambda C_2)$ are averaged operators and hence they are non-expansive. If $\gamma \in \left(0, \frac{1}{\|AA^*\|}\right)$, then the operator $V := I + \gamma A^*(J_\lambda^{B_2}(I - \lambda C_2) - I)A$ is averaged, and hence it is also non-expansive, where A^* is the adjoint operator of A.

Lemma 2.4. ([27]) *Let H be a Hilbert space. Let $T : H \to H$ be a non-expansive operator. Then for all $x, y \in H$, we have*

$$\langle (x - Tx) - (y - Ty), Ty - Tx \rangle \leq \tfrac{1}{2} \|(Tx - x) - (Ty - y)\|^2,$$

and consequently if $y \in F(T)$ then

$$\langle x - Tx, Ty - Tx \rangle \leq \tfrac{1}{2} \|Tx - x\|^2.$$

Lemma 2.5. ([28]) *Let $x, y \in H$ and $\alpha \in \mathbb{R}$. Then*
(i) $|\langle x, y \rangle| \leq \|x\| \|y\|$,
(ii) $\|x + y\|^2 \leq \|x\|^2 + 2\langle y, x + y \rangle$,
(iii) $\|\alpha x + (1 - \alpha) y\|^2 = \alpha \|x\|^2 + (1 - \alpha) \|y\|^2 - \alpha (1 - \alpha) \|x - y\|^2$,
(iv) $2\langle x, y \rangle = \|x + y\|^2 - \|x\|^2 - \|y\|^2 = \|x\|^2 + \|y\|^2 - \|x - y\|^2$.

Lemma 2.6. ([29]) *Let H be a real Hilbert space, $C \subset H$ be a nonempty closed convex set. Let $T : C \to C$ is a non-expansive operator with Fix$(T) \neq \emptyset$. The operator $I - T$ is said to be demiclosed at zero if for any sequence $\{x_n\} \subset C$, that $x_n \rightharpoonup \in C$ and $\|x_n - Tx\| \to 0$ implies $x \in$ Fix(T).*

Lemma 2.7. ([30]) *Assume that $\{s_n\}$ is a sequence of nonnegative real numbers satisfying*

$$s_{n+1} \leq (1 - \varpi_n) s_n + \varpi_n \sigma_n + \delta_n, \forall n \geq 1,$$

where $\{\varpi_n\} \subset (0, 1)$, $\{\sigma_n\}$, $\{\delta_n\} \subset \mathbb{R}$ satisfying
(i) $\sum_{n=1}^{\infty} \varpi_n = +\infty$,
(ii) $\lim\limits_{n \to \infty} sup \sigma_n \leq 0$,
(iii) $\sum_{n=1}^{\infty} |\delta_n| < +\infty$.
Then $\lim_{n \to \infty} s_n = 0$.

3. Algorithms and the Convergence Analyses

In this section, we propose the scaled forward-backward algorithm and analyse the strong convergence of the generated sequence in subsection 3.1. Then we verify the bounded perturbation resilience of the exact version of the proposed algorithm in subsection 3.2. Finally, we give the corresponding superiorization algorithm and the version with restarted perturbations in subsection 3.3.

3.1. **The scaled forward-backward algorithm.**

In this subsection, we propose an inexact scaled forward-backward algorithm with inertial terms and perturbations to solve problem (1.1). Then we prove the strong convergence of it.

Given arbitrary $x_0,\ x_1 \in H_1$. We define an iterative sequence $\{x_n\}$ by

$$\begin{cases} y_n = x_n + \theta_n(x_n - x_{n-1}) + e_n, \\ z_n = y_n + \gamma_n A^*(J_\varsigma^{B_2}(Ay_n - \varsigma D_2(Ay_n)C_2Ay_n) - Ay_n), \\ x_{n+1} = \alpha_n f(y_n) + (1-\alpha_n)J_\lambda^{B_1}(z_n - \lambda D_1(z_n)C_1z_n), n \geq 1, \end{cases} \tag{3.1}$$

where $\theta_n \geq 0,\ \gamma_n > 0,\ \varsigma, \lambda \in (0, 2\min\{v_1, v_2\})$, $\alpha_n \in [0,1]$. $f : H_1 \to H_1$ is a μ-contraction operator with $\mu \in [0,1)$. $\{e_n\}$ is a perturbated sequence. $D_i : H_i \to B(H_i, H_i),\ i = 1,2$ are scaled operators.

To obtain the strong convergence result of algorithm (3.1), we begin with some assumptions.

Assumption 3.1. Supposed that the operators and the solution set satisfy the following conditions, respectively:

(i) $B_1 : H_1 \to 2^{H_1}$, $B_2 : H_2 \to 2^{H_2}$ are maximal monotone operators,

(ii) $C_1 : H_1 \to H_1$ is a v_1-inverse strongly monotone operator, and $C_2 : H_2 \to H_2$ is a v_2-inverse strongly monotone operator,

(iii) $A : H_1 \to H_2$ is a bounded linear operator, and

(iv) the solution set Ω of problem (1.1) is nonempty, that is,

$$\Omega = \{p \in H_1 |\ 0 \in (B_1 + C_1)p \text{ and } 0 \in (B_2 + C_2)Ap\} \neq \emptyset.$$

Assumption 3.2. Suppose the following conditions hold:

(i) $\theta_n \in [0,1)$, $\sum_{n=1}^\infty \theta_n\|x_n - x_{n-1}\| < +\infty$,

(ii) $\{\alpha_n\} \subset (0,1)$, $\lim_{n\to\infty} \alpha_n = 0$, $\sum_{n=1}^\infty \alpha_n = +\infty$ and $\sum_{n=1}^\infty |\alpha_n - \alpha_{n-1}| < +\infty$,

(iii) $0 < \inf\gamma_n \leq \gamma_n \leq \sup\gamma_n < \frac{1}{\|AA^*\|}$ and $\sum_{n=1}^\infty |\gamma_n - \gamma_{n-1}| < +\infty$,

(iv) $\sum_{n=1}^\infty \|e_n\| < +\infty$, and

(v) For each $x \in H_1,\ y \in H_2$, the scaled operators $D_1(x),\ \ D_2(y)$ are linear bounded operators and satisfy $\sum_{n=1}^\infty \|\varrho_1(z_n)\| := \sum_{n=1}^\infty \|D_1(z_n)C_1z_n - C_1z_n\| < +\infty$, $\sum_{n=1}^\infty \|\varrho_2(Ay_n)\| := \sum_{n=1}^\infty \|D_2(Ay_n)C_2Ay_n - C_2Ay_n\| < +\infty$.

Theorem 3.3. *Let Assumption 3.1 and Assumption 3.2 hold. Then the sequence $\{x_n\}$ generated by algorithm (3.1) converges strongly to $x^* \in \Omega$, where $x^* = P_\Omega f(x^*)$.*

Proof. We first prove that $\{x_n\}$ is a bounded sequence.

Denote by $U := J_\lambda^{B_1}(I - \lambda C_1),\ T := J_\varsigma^{B_2}(I - \varsigma C_2)$, and $\widetilde{U}z_n := J_\lambda^{B_1}(I - \lambda D_1C_1)z_n := J_\lambda^{B_1}(z_n - \lambda D_1(z_n)C_1z_n)$, $\widetilde{T}Ay_n := J_\varsigma^{B_2}(I - \varsigma D_2C_2)Ay_n := J_\varsigma^{B_2}(Ay_n - \varsigma D_2(Ay_n)C_2Ay_n)$ for simplicity.

For any $p \in \Omega$, we have $Up = p$ and $TAp = Ap$ by Lemma 2.3.

$$\begin{aligned} \|x_{n+1} - p\| =& \|\alpha_n f(y_n) + (1-\alpha_n)u_n - p\| \\ =& \|\alpha_n(f(y_n) - f(p)) + \alpha_n(f(p) - p) + (1-\alpha_n)(u_n - p)\| \\ \leq& \alpha_n\|f(y_n) - f(p)\| + \alpha_n\|f(p) - p\| + (1-\alpha_n)\|u_n - p\| \\ \leq& \alpha_n\mu\|y_n - p\| + \alpha_n\|f(p) - p\| + (1-\alpha_n)\|u_n - p\|. \end{aligned} \tag{3.2}$$

Further,

$$\|y_n - p\| = \|x_n - p + \theta_n(x_n - x_{n-1}) + e_n\| \leq \|x_n - p\| + \theta_n\|x_n - x_{n-1}\| + \|e_n\|, \tag{3.3}$$

and it follows from the non-expansiveness of U (see Lemma 2.3), $J_\lambda^{B_1}$ and Assumption 3.2 (v) that

$$\begin{aligned}\|u_n - p\| =& \|\widetilde{U}z_n - p\| \\ \leq& \|\widetilde{U}z_n - Uz_n\| + \|Uz_n - Up\| \\ =& \|J_\lambda^{B_1}(I - \lambda D_1C_1)z_n - J_\lambda^{B_1}(I - \lambda C_1)z_n\| + \|z_n - p\| \\ \leq& \|(I - \lambda D_1C_1)z_n - (I - \lambda C_1)z_n\| + \|z_n - p\| \\ =& \|z_n - p\| + \lambda\|\varrho_1(z_n)\|.\end{aligned} \tag{3.4}$$

To estimate the value of $\|z_n - p\|$ in (3.4), we note from Lemma 2.4 that

$$\begin{aligned}2\gamma_n\langle y_n - p, A^*(T - I)Ay_n\rangle =& 2\gamma_n\langle A(y_n - p) + (T - I)Ay_n - (T - I)Ay_n, (T - I)Ay_n\rangle \\ =& 2\gamma_n\left(\langle TAy_n - Ap, (T - I)Ay_n\rangle - \|(T - I)Ay_n\|^2\right) \\ \leq& 2\gamma_n\left(\frac{1}{2}\|(T - I)Ay_n\|^2 - \|(T - I)Ay_n\|^2\right) \\ =& -\gamma_n\|(T - I)Ay_n\|^2,\end{aligned}$$

which indicates

$$\begin{aligned}&\|y_n + \gamma_nA^*(T - I)Ay_n - p\|^2 \\ =& \|y_n - p\|^2 + 2\gamma_n\langle y_n - p, A^*(T - I)Ay_n\rangle + \gamma_n^2\|A^*(T - I)Ay_n\|^2 \\ \leq& \|y_n - p\|^2 + 2\gamma_n\langle y_n - p, A^*(T - I)Ay_n\rangle + \gamma_n^2\|AA^*\|\|(T - I)Ay_n\|^2 \\ \leq& \|y_n - p\|^2 - \gamma_n\|(T - I)Ay_n\|^2 + \gamma_n^2\|AA^*\|\|(T - I)Ay_n\|^2 \\ =& \|y_n - p\|^2 - \gamma_n(1 - \gamma_n\|AA^*\|)\|(T - I)Ay_n\|^2 \\ \leq& \|y_n - p\|^2\end{aligned}$$

as $0 < \inf\gamma_n \leq \gamma_n \leq \sup\gamma_n < \frac{1}{\|AA^*\|}$. Hence,

$$\begin{aligned}\|z_n - p\| =& \|y_n + \gamma_nA^*(\widetilde{T} - I)Ay_n - p\| \\ =& \|y_n + \gamma_nA^*(\widetilde{T} - I)Ay_n - \gamma_nA^*(T - I)Ay_n + \gamma_nA^*(T - I)Ay_n - p\| \\ \leq& \|\gamma_nA^*(\widetilde{T} - I)Ay_n - \gamma_nA^*(T - I)Ay_n\| + \|y_n + \gamma_nA^*(T - I)Ay_n - p\| \\ \leq& \gamma_n\|A^*\|\|\widetilde{T}Ay_n - TAy_n\| + \|y_n + \gamma_nA^*(T - I)Ay_n - p\| \\ =& \gamma_n\|A^*\|[\|J_\varsigma^{B_2}(I - \varsigma D_2C_2)Ay_n - J_\varsigma^{B_2}(I - \varsigma C_2)Ay_n\|] + \|y_n + \gamma_nA^*(T - I)Ay_n - p\| \\ \leq& \varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\| + \|y_n + \gamma_nA^*(T - I)Ay_n - p\| \qquad (3.5) \\ \leq& \varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\| + \|y_n - p\|. \qquad (3.6)\end{aligned}$$

Substituting (3.6) into (3.4), we get

$$\begin{aligned}\|u_n - p\| \leq& \|z_n - p\| + \lambda\|\varrho_1(z_n)\| \qquad (3.7) \\ \leq& \|y_n - p\| + \varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\| + \lambda\|\varrho_1(z_n)\|. \qquad (3.8)\end{aligned}$$

Now, combining (3.2), (3.8) and (3.3) yields

$$\begin{aligned}
&\|x_{n+1}-p\| \\
\leq&\alpha_n\mu\|y_n-p\|+\alpha_n\|f(p)-p\|+(1-\alpha_n)\|u_n-p\| \\
\leq&\alpha_n\mu\|y_n-p\|+\alpha_n\|f(p)-p\|+(1-\alpha_n)\left(\|y_n-p\|+\varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\|+\lambda\|\varrho_1(z_n)\|\right) \\
\leq&(\alpha_n\mu+1-\alpha_n)\|y_n-p\|+\alpha_n\|f(p)-p\|+\varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\|+\lambda\|\varrho_1(z_n)\| \\
\leq&(1-\alpha_n(1-\mu))\|x_n-p\|+\alpha_n(1-\mu)\frac{\|f(p)-p\|}{1-\mu} \\
&+\varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\|+\lambda\|\varrho_1(z_n)\|+\theta_n\|x_n-x_{n-1}\|+\|e_n\| \\
\leq&(1-\alpha_n(1-\mu))\|x_n-p\|+\alpha_n(1-\mu)\frac{\|f(p)-p\|}{1-\mu}+M_n,
\end{aligned}$$

where $M_n=\varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\|+\lambda\|\varrho_1(z_n)\|+\theta_n\|x_n-x_{n-1}\|+\|e_n\|$ such that $\sum_{n=1}^{\infty}M_n<+\infty$. By mathematical induction, we obtain

$$\|x_{n+1}-p\| \leq\max\left\{\|x_0-p\|,\frac{\|f(p)-p\|}{1-\mu}\right\}+\sum_{n=1}^{\infty}M_n.$$

So $\{x_n\}$ is bounded. Consequently, $\{y_n\}$, $\{u_n\}$ are bounded due to (3.3), (3.8) and Assumption 3.2 (v). Hence, $\{f(y_n)\}$ is also bounded.

Next, we prove that $\omega_w(x_n)\subset\Omega$. First of all, the boundedness of $\{x_n\}$ implies that $\omega_w(x_n)\neq\emptyset$. For any $q\in\omega_w(x_n)$, then there exists a subsequence $\{x_{n_k}\}$ of $\{x_n\}$ such that $x_{n_k}\rightharpoonup q$ as $k\to\infty$. In the following, we prove that $0\in(B_1+C_1)q$ by utilizing the maximum monotonicity of the operator B_1+C_1 and that $y_{n_k}\rightharpoonup q$ as $k\to\infty$, $\lim_{n\to\infty}\|(T-I)Ay_{n_k}\|=0$, which indicates that $Aq=TAq$. Then $0\in(B_2+C_2)Aq$ follows from Lemma 2.3. To show $0\in(B_1+C_1)q$, we need to verify $u_{n_k}\rightharpoonup q$ as $k\to\infty$ and $\lim_{n\to\infty}\|u_n-z_n\|=0$ at first.

From the definitions of $\{x_n\}$ and $\{y_n\}$, we get

$$\begin{aligned}
\|x_{n+1}-x_n\| =&\|\alpha_nf(y_n)+(1-\alpha_n)u_n-\alpha_{n-1}f(y_{n-1})-(1-\alpha_{n-1})u_{n-1}\| \\
=&\|\alpha_n(f(y_n)-f(y_{n-1}))+(\alpha_n-\alpha_{n-1})f(y_{n-1}) \\
&+(1-\alpha_n)(u_n-u_{n-1})-(\alpha_n-\alpha_{n-1})u_{n-1}\| \\
\leq&\alpha_n\mu\|y_n-y_{n-1}\|+(1-\alpha_n)\|u_n-u_{n-1}\|+|\alpha_n-\alpha_{n-1}|\|f(y_{n-1})-u_{n-1}\|,
\end{aligned} \tag{3.9}$$

and

$$\begin{aligned}
\|y_n-y_{n-1}\| =&\|x_n+\theta_n(x_n-x_{n-1})+e_n-x_{n-1}-\theta_{n-1}(x_{n-1}-x_{n-2})-e_{n-1}\| \\
\leq&\|x_n-x_{n-1}\|+\theta_n\|x_n-x_{n-1}\|+\theta_{n-1}\|x_{n-1}-x_{n-2}\|+\|e_n\|+\|e_{n-1}\|.
\end{aligned} \tag{3.10}$$

By Assumption 3.2 (v) and Lemma 2.3, we further get

$$\begin{aligned}
\|u_n-u_{n-1}\| =&\|\widetilde{U}z_n-\widetilde{U}z_{n-1}\| \\
\leq&\|\widetilde{U}z_n-Uz_n\|+\|\widetilde{U}z_{n-1}-Uz_{n-1}\|+\|Uz_n-Uz_{n-1}\| \\
\leq&\lambda\|\varrho_1(z_n)\|+\lambda\|\varrho_1(z_{n-1})\|+\|z_n-z_{n-1}\|,
\end{aligned} \tag{3.11}$$

and

$$\begin{aligned}\|z_n - z_{n-1}\| =&\|y_n + \gamma_n A^*(\widetilde{T} - I)Ay_n - (y_{n-1} + \gamma_{n-1}A^*(\widetilde{T} - I)Ay_{n-1})\| \\ \le&\|\gamma_n A^*(\widetilde{T} - I)Ay_n - \gamma_n A^*(T - I)Ay_n\| \\ &+ \|\gamma_{n-1}A^*(\widetilde{T} - I)Ay_{n-1} - \gamma_{n-1}A^*(T - I)Ay_{n-1}\| \\ &+ \|y_n + \gamma_n A^*(T - I)Ay_n - (y_{n-1} + \gamma_{n-1}A^*(T - I)Ay_{n-1})\| \\ \le&\gamma_n\|A^*\|\|\widetilde{T}Ay_n - TAy_n\| + \gamma_{n-1}\|A^*\|\|\widetilde{T}Ay_{n-1} - TAy_{n-1}\| \\ &+ \|(I + \gamma_n A^*(T - I)A)(y_n - y_{n-1}) + (I + \gamma_n A^*(T - I)A)y_{n-1} \\ &- (I + \gamma_{n-1}A^*(T - I)A)y_{n-1}\| \\ \le&\varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\| + \varsigma\gamma_{n-1}\|A^*\|\|\varrho_2(Ay_{n-1})\| + \|y_n - y_{n-1}\| \\ &+ |\gamma_n - \gamma_{n-1}|\|A^*(T - I)Ay_{n-1}\|.\end{aligned} \tag{3.12}$$

So, by (3.9)-(3.12), we have

$$\begin{aligned}&\|x_{n+1} - x_n\| \\ \le&\alpha_n\mu\|y_n - y_{n-1}\| + (1 - \alpha_n)\|u_n - u_{n-1}\| + |\alpha_n - \alpha_{n-1}|\|f(y_{n-1}) - u_{n-1}\| \\ \le&\alpha_n\mu\|y_n - y_{n-1}\| + |\alpha_n - \alpha_{n-1}|\|f(y_{n-1}) - u_{n-1}\| \\ &+ (1 - \alpha_n)\left(\lambda\|\varrho_1(z_n)\| + \lambda\|\varrho_1(z_{n-1})\| + \|z_n - z_{n-1}\|\right) \\ \le&\alpha_n\mu\|y_n - y_{n-1}\| + |\alpha_n - \alpha_{n-1}|\|f(y_{n-1}) - u_{n-1}\| + (1 - \alpha_n)[\lambda\|\varrho_1(z_n)\| + \lambda\|\varrho_1(z_{n-1})\| \\ &+ \varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\| + \varsigma\gamma_{n-1}\|A^*\|\|\varrho_2(Ay_{n-1})\| + \|y_n - y_{n-1}\| + |\gamma_n - \gamma_{n-1}|\|A^*(T - I)Ay_{n-1}\|] \\ \le&(1 - \alpha_n(1 - \mu))\|x_n - x_{n-1}\| + N_n,\end{aligned}$$

where $N_n = \theta_n\|x_n - x_{n-1}\| + \theta_{n-1}\|x_{n-1} - x_{n-2}\| + \|e_n\| + \|e_{n-1}\| + |\alpha_n - \alpha_{n-1}|\|f(y_{n-1}) - u_{n-1}\| + \lambda\|\varrho_1(z_n)\| + \lambda\|\varrho_1(z_{n-1})\| + \varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\| + \varsigma\gamma_{n-1}\|A^*\|\|\varrho_2(Ay_{n-1})\| + |\gamma_n - \gamma_{n-1}|\|A^*(T - I)Ay_{n-1}\|$ satisfying $\sum_{n=1}^{\infty} N_n < +\infty$.

Take $\varpi_n = \alpha_n(1 - \mu),\ \sigma_n = 0,\ \delta_n = N_n,\ n \ge 1$ in Lemma 2.7. We obtain

$$\lim_{n\to\infty} \|x_{n+1} - x_n\| = 0. \tag{3.13}$$

As a result,

$$\begin{aligned}\|u_n - x_n\| \le&\|u_n - x_{n+1}\| + \|x_{n+1} - x_n\| \\ =&\|u_n - \alpha_n f(y_n) - (1 - \alpha_n)u_n\| + \|x_{n+1} - x_n\| \\ \le&\alpha_n\|u_n - f(y_n)\| + \|x_{n+1} - x_n\| \\ \to&0 \ \text{ as } \ n \to \infty.\end{aligned}$$

Thus, we have

$$\lim_{n\to\infty} \|u_n - x_n\| = 0, \tag{3.14}$$

which together with $x_{n_k} \rightharpoonup q$ indicate

$$u_{n_k} \rightharpoonup q \text{ as } k \to \infty.$$

On the other hand, we have

$$\begin{aligned}\|u_n - p\|^2 \le&(\|z_n - p\| + \lambda\|\varrho_1(z_n)\|)^2 \\ =&\|z_n - p\|^2 + 2\lambda\|z_n - p\|\|\varrho_1(z_n)\| + (\lambda\|\varrho_1(z_n)\|)^2,\end{aligned}$$

and

$$\begin{aligned}
\|z_n - p\|^2 \leq &(\|y_n + \gamma_n A^*(T-I)Ay_n - p\| + \gamma_n \varsigma \|A^*\|\|\varrho_2(Ay_n)\|)^2 \\
\leq &\|y_n + \gamma_n A^*(T-I)Ay_n - p\|^2 + (\varsigma\gamma_n \|A^*\|\|\varrho_2(Ay_n)\|)^2 \\
&+ 2\varsigma\gamma_n \|A^*\|\|\varrho_2(Ay_n)\|\|y_n - p\| \\
\leq &\|y_n - p\|^2 - \gamma_n(1-\gamma_n\|AA^*\|)\|(T-I)Ay_n\|^2 + (\varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\|)^2 \\
&+ 2\varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\|\|y_n - p\|
\end{aligned}$$

based on (3.4), (3.5), respectively. Hence, we get

$$\begin{aligned}
\|u_n - p\|^2 \leq &\|y_n - p\|^2 - \gamma_n(1-\gamma_n\|AA^*\|)\|(T-I)Ay_n\|^2 + (\varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\|)^2 \\
&+ 2\varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\|\|y_n - p\| + 2\lambda\|z_n - p\|\|\varrho_1(z_n)\| + (\lambda\|\varrho_1(z_n)\|)^2.
\end{aligned} \tag{3.15}$$

Also, because

$$\begin{aligned}
\|y_n - p\|^2 = &\|x_n + \theta_n(x_n - x_{n-1}) - p + e_n\|^2 \\
\leq &\|x_n + \theta_n(x_n - x_{n-1}) - p\|^2 + 2\langle e_n, y_n - p\rangle \\
\leq &\|x_n - p\|^2 + \theta_n^2\|x_n - x_{n-1}\|^2 + 2\theta_n\langle x_n - p, x_n - x_{n-1}\rangle + 2\langle e_n, y_n - p\rangle \\
\leq &\|x_n - p\|^2 + \theta_n^2\|x_n - x_{n-1}\|^2 + \theta_n(\|x_n - p\|^2 - \|x_{n-1} - p\|^2 \\
&+ \|x_n - x_{n-1}\|^2) + 2\|e_n\|\|y_n - p\| \\
= &\|x_n - p\|^2 + \theta_n(\|x_n - p\|^2 - \|x_{n-1} - p\|^2) \\
&+ (\theta_n + \theta_n^2)\|x_n - x_{n-1}\|^2 + 2\|e_n\|\|y_n - p\| \\
\leq &\|x_n - p\|^2 + \theta_n(\|x_n - p\|^2 - \|x_{n-1} - p\|^2) \\
&+ 2\theta_n\|x_n - x_{n-1}\|^2 + 2\|e_n\|\|y_n - p\|,
\end{aligned} \tag{3.16}$$

we conclude that by applying Lemma 2.5 (ii), (3.15) and (3.16)

$$\begin{aligned}
&\|x_{n+1} - p\|^2 \\
= &\|\alpha_n f(y_n) + (1-\alpha_n)u_n - p\|^2 \\
= &\|\alpha_n(f(y_n) - f(p)) + \alpha_n(f(p) - p) + (1-\alpha_n)(u_n - p)\|^2 \\
\leq &\alpha_n\|f(y_n) - f(p)\|^2 + (1-\alpha_n)\|u_n - p\|^2 + 2\alpha_n\langle f(p) - p, x_{n+1} - p\rangle \\
\leq &\alpha_n\mu\|y_n - p\|^2 + 2\alpha_n\langle f(p) - p, x_{n+1} - p\rangle + (1-\alpha_n)[\|y_n - p\|^2 \\
&- \gamma_n(1-\gamma_n\|AA^*\|)\|(T-I)Ay_n\|^2 + (\varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\|)^2 \\
&+ 2\varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\|\|y_n - p\| + 2\lambda\|z_n - p\|\|\varrho_1(z_n)\| + (\lambda\|\varrho_1(z_n)\|)^2] \\
\leq &(1-\alpha_n(1-\mu))[\|x_n - p\|^2 + \theta_n(\|x_n - p\|^2 - \|x_{n-1} - p\|^2) + 2\theta_n\|x_n - x_{n-1}\|^2 \\
&+ 2\|e_n\|\|y_n - p\|] + 2\alpha_n\langle f(p) - p, x_{n+1} - p\rangle \\
&- (1-\alpha_n)\gamma_n(1-\gamma_n\|AA^*\|)\|(T-I)Ay_n\|^2 + (\varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\|)^2 \\
&+ 2\varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\|\|y_n - p\| + 2\lambda\|z_n - p\|\|\varrho_1(z_n)\| + (\lambda\|\varrho_1(z_n)\|)^2 \\
\leq &(1-\alpha_n(1-\mu))\|x_n - p\|^2 + \theta_n(\|x_n - p\|^2 - \|x_{n-1} - p\|^2) \\
&+ 2\alpha_n\langle f(p) - p, x_{n+1} - p\rangle - (1-\alpha_n)\gamma_n(1-\gamma_n\|AA^*\|)\|(T-I)Ay_n\|^2 + K_n,
\end{aligned} \tag{3.17}$$

where $K_n = 2\theta_n\|x_n - x_{n-1}\|^2 + 2\|e_n\|\|y_n - p\| + (\varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\|)^2 + 2\varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\|\|y_n - p\| + 2\lambda\|z_n - p\|\|\varrho_1(z_n)\| + (\lambda\|\varrho_1(z_n)\|)^2$ such that $\sum_{n=1}^{\infty} K_n < +\infty$. Thus, the second term on the right of (3.17) can be estimated as follows

$$\begin{aligned}
&(1-\alpha_n)(\gamma_n-\gamma_n^2\|AA^*\|)\|(T-I)Ay_n\|^2\\
\leq&\|x_n-p\|^2-\|x_{n+1}-p\|^2+\theta_n(\|x_n-p\|^2-\|x_{n-1}-p\|^2)+2\alpha_n\|f(p)-p\|\|x_{n+1}-p\|+K_n\\
=&(\|x_n-p\|-\|x_{n+1}-p\|)(\|x_n-p\|+\|x_{n+1}-p\|)\\
&+\theta_n(\|x_n-p\|-\|x_{n-1}-p\|)(\|x_n-p\|+\|x_{n-1}-p\|)+2\alpha_n\|f(p)-p\|\|x_{n+1}-p\|+K_n\\
\leq&\|x_n-x_{n+1}\|(\|x_n-p\|+\|x_{n+1}-p\|)+\theta_n\|x_n-x_{n-1}\|(\|x_n-p\|+\|x_{n-1}-p\|)\\
&+2\alpha_n\|f(p)-p\|\|x_{n+1}-p\|+K_n\\
\to&0 \text{ as } n\to\infty,
\end{aligned}$$

which implies

$$\lim_{n\to\infty}\|(T-I)Ay_n\|=0 \tag{3.18}$$

in view of Assumption 3.2 (iii). Therefore,

$$\begin{aligned}
\|x_n-z_n\|\leq&\|x_n-y_n\|+\|y_n-z_n\|\\
\leq&\|x_n-y_n\|+\gamma_n\|A^*\|\|(\widetilde{T}-I)Ay_n\|\\
\leq&\|x_n-y_n\|+\gamma_n\|A^*\|(\|(\widetilde{T}-I)Ay_n-(T-I)Ay_n\|+\|(T-I)Ay_n\|)\\
\leq&\|x_n-y_n\|+\varsigma\gamma_n\|A^*\|\|\varrho_2(Ay_n)\|+\gamma_n\|A^*\|\|(T-I)Ay_n\|\\
\to&0 \text{ as } n\to\infty
\end{aligned} \tag{3.19}$$

since

$$\|x_n-y_n\|\leq\theta_n\|x_n-x_{n-1}\|+\|e_n\|\to 0 \text{ as } n\to\infty. \tag{3.20}$$

Finally, by using (3.14) and (3.19), we get

$$\|u_n-z_n\|\leq\|u_n-x_n\|+\|x_n-z_n\|\to 0 \text{ as } n\to\infty.$$

It yields

$$\lim_{n\to\infty}\|u_n-z_n\|=0. \tag{3.21}$$

Now, we prove $0\in(B_1+C_1)q$. Observe that $u_{n_k}=J_\lambda^{B_1}(I-\lambda D_1C_1)z_{n_k}$. We get

$$(I-\lambda D_1C_1)z_{n_k}\in(I+\lambda B_1)u_{n_k}, \text{ or } \tfrac{1}{\lambda}(z_{n_k}-\lambda D_1C_1z_{n_k}-u_{n_k})\in B_1u_{n_k}.$$

Siminarly, for any $(v,z)\in\text{Graph}(B_1+C_1)$, we have $z-C_1v\in B_1v$. Then by the monotonicity of B_1, it has

$$\left\langle v-u_{n_k}, z-C_1v-\tfrac{1}{\lambda}(z_{n_k}-\lambda D_1C_1z_{n_k}-u_{n_k})\right\rangle\geq 0.$$

Consequently,

$$\begin{aligned}
&\langle v-u_{n_k},z\rangle\\
\geq&\left\langle v-u_{n_k},C_1v+\tfrac{1}{\lambda}(z_{n_k}-\lambda D_1C_1z_{n_k}-u_{n_k})\right\rangle\\
=&\left\langle v-u_{n_k},C_1v-D_1C_1z_{n_k}+C_1u_{n_k}-C_1u_{n_k}+\tfrac{1}{\lambda}(z_{n_k}-u_{n_k})\right\rangle\\
=&\langle v-u_{n_k},C_1v-C_1u_{n_k}\rangle+\langle v-u_{n_k},C_1u_{n_k}-D_1C_1z_{n_k}\rangle+\left\langle v-u_{n_k},\tfrac{1}{\lambda}(z_{n_k}-u_{n_k})\right\rangle\\
\geq&v_1\|C_1v-C_1u_{n_k}\|^2+\langle v-u_{n_k},C_1u_{n_k}-D_1C_1z_{n_k}\rangle+\left\langle v-u_{n_k},\tfrac{1}{\lambda}(z_{n_k}-u_{n_k})\right\rangle\\
\geq&\langle v-u_{n_k},C_1u_{n_k}-D_1C_1z_{n_k}\rangle+\left\langle v-u_{n_k},\tfrac{1}{\lambda}(z_{n_k}-u_{n_k})\right\rangle.
\end{aligned} \tag{3.22}$$

We claim that the two terms on the right side of (3.22) tend to zero as n goes to infinity. In fact, from Assumption 3.2 (v), (3.21) and the fact that C_1 is a $\frac{1}{v_1}$-Lipschitz continuous operator, we derive

$$\begin{aligned}
&|\langle v-u_{n_k}, C_1u_{n_k}-D_1C_1z_{n_k}\rangle| \\
=&|\langle v-u_{n_k}, C_1u_{n_k}-D_1C_1z_{n_k}+C_1z_{n_k}-C_1z_{n_k}\rangle| \\
\le&|\langle v-u_{n_k}, C_1u_{n_k}-C_1z_{n_k}\rangle| + |\langle v-u_{n_k}, C_1z_{n_k}-D_1C_1z_{n_k}\rangle| \\
\le&\frac{1}{v_1}\|v-u_{n_k}\|\|u_{n_k}-z_{n_k}\| + \|v-u_{n_k}\|\|\varrho_1(z_{n_k})\| \\
\to&0 \text{ as } k\to\infty.
\end{aligned}$$

This means

$$\lim_{k\to\infty}\langle v-u_{n_k}, C_1u_{n_k}-D_1C_1z_{n_k}\rangle = 0.$$

Moreover, that

$$\lim_{k\to\infty}\frac{1}{\lambda}\langle v-u_{n_k}, z_{n_k}-u_{n_k}\rangle = 0$$

follows from

$$0\le\frac{1}{\lambda}|\langle v-u_{n_k}, z_{n_k}-u_{n_k}\rangle| \le \frac{1}{\lambda}\|v-u_{n_k}\|\|u_{n_k}-z_{n_k}\| \to 0 \text{ as } k\to\infty.$$

Taking the limit with respect to k on both sides of (3.22), we get

$$\lim_{k\to\infty}\langle v-u_{n_k}, z\rangle = \langle v-q, z\rangle \ge 0.$$

Thus we have $0\in(B_1+C_1)q$ as B_1+C_1 is a maximal monotone operator.

Next, we prove $0\in(B_2+C_2)Aq$. By (3.20) and that $x_{n_k}\rightharpoonup q$ as $k\to\infty$, we get $y_{n_k}\rightharpoonup q$ as $k\to\infty$, which implies $Ay_{n_k}\rightharpoonup Aq$ as $k\to\infty$ since A is a bounded linear operator. We conclude that $Aq=TAq$ by (3.18) and Lemma 2.6. Moreover, $0\in(B_2+C_2)Aq$ follows from Lemma 2.3. Hence, $q\in\Omega$. So we obtain $\omega_w(x_n)\subset\Omega$.

Finally, we prove $\{x_n\}$ converges strongly to $x^*=P_\Omega f(x^*)$. From (3.17), we know that

$$\begin{aligned}
&\|x_{n+1}-x^*\|^2 \\
\le&(1-\alpha_n(1-\mu))\|x_n-x^*\|^2+\theta_n(\|x_n-x^*\|^2-\|x_{n-1}-x^*\|^2) \\
&+2\alpha_n\langle f(x^*)-x^*, x_{n+1}-x^*\rangle-(1-\alpha_n)\gamma_n(1-\gamma_n\|AA^*\|)\|(T-I)Ay_n\|^2+K_n \\
\le&(1-\alpha_n(1-\mu))\|x_n-x^*\|^2+\theta_n\|x_n-x_{n-1}\|(\|x_n-x^*\|+\|x_{n-1}-x^*\|) \\
&+2\alpha_n\langle f(x^*)-x^*, x_{n+1}-x^*\rangle+K_n \\
=&(1-\varpi_n)\|x_n-x^*\|^2+\varpi_n\sigma_n+\delta_n.
\end{aligned}$$

Set $\varpi_n=\alpha_n(1-\mu)$, $\sigma_n=\frac{2}{1-\mu}\langle f(x^*)-x^*, x_{n+1}-x^*\rangle$ and $\delta_n=\theta_n\|x_n-x_{n-1}\|(\|x_n-x^*\|+\|x_{n-1}-x^*\|)+K_n$ in Lemma 2.7. Then it is clear that $\sum_{n=1}^\infty\varpi_n=\infty$ and $\sum_{n=1}^\infty\delta_n<\infty$. We verify that $\{\sigma_n\}$ satisfies the condition (ii) of Lemma 2.7 now. Obviously, we have

$$|\sigma_n|\le\frac{2}{1-\mu}\|f(x^*)-x^*\|\|x_{n+1}-x^*\|<+\infty,$$

which implies that $\limsup_{n\to\infty}\sigma_n$ is a finite number. Therefore, there exists a subsequence $\{\sigma_{n_j}\}\subset\{\sigma_n\}$ such that $\limsup_{n\to\infty}\sigma_n=\lim_{j\to\infty}\sigma_{n_j}$. Without loss of generality, assume $x_{n_j}\rightharpoonup\widetilde{q}$ as $j\to\infty$.

Then $\widetilde{q} \in \Omega$ and $x_{n_j+1} \rightharpoonup \widetilde{q}$ as $j \to \infty$ according to (3.13). So,

$$\begin{aligned}\limsup_{n\to\infty} \sigma_n &= \frac{2}{1-\mu} \limsup_{n\to\infty} \langle f(x^*) - x^*, x_{n+1} - x^* \rangle \\ &= \frac{2}{1-\mu} \lim_{j\to\infty} \langle f(x^*) - x^*, x_{n_j+1} - x^* \rangle \\ &= \frac{2}{1-\mu} \langle f(x^*) - x^*, \widetilde{q} - x^* \rangle \\ &\leq 0.\end{aligned}$$

As a result of Lemma 2.7, we have

$$\lim_{n\to\infty} \|x_n - x^*\| = 0.$$

That is $\{x_n\}$ converges strongly to $x^* = P_\Omega f(x^*)$. □

3.2. **Bounded perturbation algorithm and convergence results.** In this subsection, we start with the definition of bounded perturbation resilience of an algorithmic operator, then we introduce the bounded perturbation algorithm of the exact version of algorithm (3.1). The strong convergence of it follows from Theorem 3.3.

Definition 3.4. (Bounded perturbation resilience [20]). Given a problem ψ. An algorithmic operator $T : H \to H$ is said to be bounded perturbation resilient if the following condition holds: if the sequence $\{x_n\}$, generated by $x_{n+1} = Tx_n$ with $x_0 \in H$, converges to a solution of ψ, then any sequence $\{y_n\}$, generated by $y_{n+1} = T(y_n + \beta_n v_n)$ with $y_0 \in H$, also converges to a solution of ψ, where $\{v_n\}$ is a bounded sequence in H and the scalars $\beta_n (n = 0, 1, 2, \cdots)$ satisfy $\beta_n > 0$ and $\sum_{n=0}^{\infty} \beta_n < \infty$.

We consider the exact version of algorithm (3.1) as the algorithmic operator T. Let $e_n \equiv 0, n \geq 1$ in algorithm (3.1), we get the following exact version: given arbitrary $x_0,\ x_1 \in H_1$, define

$$\begin{cases} y_n = x_n + \theta_n(x_n - x_{n-1}), \\ z_n = y_n + \gamma_n A^*(J_\varsigma^{B_2}(Ay_n - \varsigma D_2(Ay_n)C_2Ay_n) - Ay_n), \\ x_{n+1} = \alpha_n f(y_n) + (1-\alpha_n) J_\lambda^{B_1}(z_n - \lambda D_1(z_n)C_1 z_n), n \geq 1. \end{cases} \tag{3.23}$$

Corollary 3.5. *Let Assumption 3.1 and Assumption 3.2 hold. Then the sequence $\{x_n\}$ generated by algorithm (3.23) converges strongly to $x^* \in \Omega$, where $x^* = P_\Omega f(x^*)$.*

The bounded perturbation algorithm of algorithm (3.23) is : given arbitrary $x_0,\ x_1 \in H_1$, iterate

$$\begin{cases} w_n = x_n + \beta_n v_n, \\ y_n^{'} = w_n + \theta_n(w_n - w_{n-1}), \\ z_n^{'} = y_n^{'} + \gamma_n A^*(J_\varsigma^{B_2}(Ay_n^{'} - \varsigma D_2(Ay_n^{'})C_2Ay_n^{'}) - Ay_n^{'}), \\ x_{n+1} = \alpha_n f(y_n^{'}) + (1-\alpha_n) J_\lambda^{B_1}(z_n^{'} - \lambda D_1(z_n^{'})C_1 z_n^{'}), n \geq 1. \end{cases} \tag{3.24}$$

Theorem 3.6. *Suppose that Assumption 3.1 and Assumption 3.2 hold. Let $\{\beta_n\}$ and $\{v_n\}$ fulfill the conditions stated in Definition 3.4. Then the sequence $\{x_n\}$ generated by (3.24) converges strongly to $x^* \in \Omega$. Hence, algorithm (3.23) is bounded perturbation resilient.*

Proof. Plug $w_n = x_n + \beta_n v_n$ into the second formula of (3.24). We get

$$\begin{aligned} y_n^{'} &= x_n + \beta_n v_n + \theta_n(x_n + \beta_n v_n - (x_{n-1} + \beta_{n-1} v_{n-1})) \\ &= x_n + \theta_n(x_n - x_{n-1}) + \beta_n v_n + \theta_n(\beta_n v_n - \beta_{n-1} v_{n-1}) \\ &= x_n + \theta_n(x_n - x_{n-1}) + e_n^{'}, \end{aligned}$$

where $e_n^{'} = \beta_n v_n + \theta_n(\beta_n v_n - \beta_{n-1}v_{n-1})$ such that

$$\begin{aligned}\sum_{n=1}^{\infty}\|e_n^{'}\| &= \sum_{n=1}^{\infty}\|\beta_n v_n + \theta_n(\beta_n v_n - \beta_{n-1}v_{n-1})\| \\ &\le \sum_{n=1}^{\infty}[\|\beta_n v_n\| + \theta_n\|\beta_n v_n\| + \theta_n\|\beta_{n-1}v_{n-1}\|] \\ &\le \sum_{n=1}^{\infty}\beta_n[\|v_n\| + \theta_n\|v_n\|] + \sum_{n=1}^{\infty}\theta_n\beta_{n-1}\|v_{n-1}\| \\ &< +\infty.\end{aligned}$$

So algorithm (3.24) can be rewritten as

$$\begin{cases} y_n^{'} = w_n + \theta_n(w_n - w_{n-1}) + e_n^{'}, \\ z_n^{'} = y_n^{'} + \gamma_n A^*(J_{\varsigma}^{B_2}(Ay_n^{'} - \varsigma D_2(Ay_n^{'})C_2Ay_n^{'}) - Ay_n^{'}), \\ x_{n+1} = \alpha_n f(y_n^{'}) + (1-\alpha_n)J_{\lambda}^{B_1}(z_n^{'} - \lambda D_1(z_n^{'})C_1 z_n^{'}), n \ge 1, \end{cases}$$

which is algorithm (3.1). So we get that the sequence $\{x_n\}$ generated by (3.24) converges strongly to $x^* \in \Omega$ according to Theorem 3.3. It also means that algorithm (3.23) is bounded perturbation resilient. □

3.3. **Superiorization algorithm and superiorization algorithm with restarted perturbations.** Superiorization algorithm works by using the bounded perturbation resilience of a basic algorithm and a nonascending direction of the target function at each iteration point. Then this algorithm provides us an automatic way for finding a solution of problem (1.1) and getting a reduced target function value. In this subsection, we introduce the concept of nonascending direction at first. Then we give the superiorized version (AlgS) of algorithm (3.23) and the superiorized version with restarted perturbations (AlgSR).

Let $\Phi : H \to \mathbb{R}$ be a target function. A vector $v \in H$ is said to be a nonascending direction of the function Φ at point x, if $\|v\| \leqslant 1$ and there exists a positive number ε such that for any $\delta \in [0, \varepsilon)$, it has $\Phi(x + \delta v) \leqslant \Phi(x)$. Such nonascending direction always exists. For example, the zero vector v is a nonascending direction of Φ at x.

The pseudocode of the superiorized version (AlgS) of the algorithm (3.23) is given in the next page.

Note that the summable sequence $\{\beta_n\} = \{ac^n\}$ employed in AlgS decreases to zero quite fast. This will make the perturbations $\beta_n v_n (n = 0, 1, 2, \cdots)$ insignificant. That restarting the perturbations to a previous value while maintaining the summability of the perturbation sequence is a useful method as it may be improve the performance of an algorithm. In the following pseudocode of superiorized version of algorithm (3.23) with restarted perturbations (AlgSR), $\{W_r\}$ is a positive integer sequence used to control when the perturbations return to a previous value.

4. Numerical experiments

In this section, we present some numerical experiments to illustrate the performance of the proposed algorithms. All numerical results are written in MATLAB R2023(b) on an Intel(R) Core(TM) i5-8265U CPU, 1.80 GHz computer with a 8.00GB RAM. Denote the number of iterations by “Iter.”, the CPU time by “Sec.” in seconds and the stopping criterion $\|x_{n+1} - x_n\| < \varepsilon$ by “ε”.

Example 4.1. Let $H_1 = H_2 = \mathbb{R}$ equipped with the inner product $\langle x, y\rangle = xy$ ($\forall\, x, y \in \mathbb{R}$) and the induced norm $\|x\| = |x|$ ($\forall x \in \mathbb{R}$). Define $B_1 : H_1 \to 2^{H_1}$ by $B_1 x = 3x$, $B_2 : H_2 \to 2^{H_2}$ by $B_2 x = 2x$, $C_1 : H_1 \to H_1$ by $C_1 x = \sin x$ and $C_2 : H_2 \to H_2$ by $C_2 x = 3x$ ($\forall x \in \mathbb{R}$). Obviously, both B_1 and B_2 are maximal monotone operators. C_1 is a 1-inverse strongly monotone operator and C_2 is a

Superiorized version of algorithm (3.23)

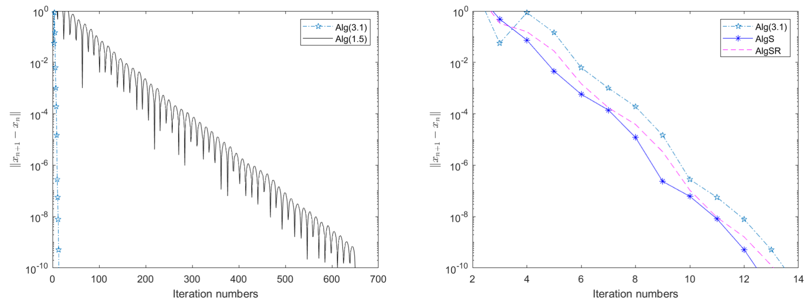
Given $x_0, x_1 \in H_1$,$c \in (0,1)$, $a > 0$, $N \in \mathbb{N}_+$, $\{\alpha_n\} \subset (0,1)$, $\theta_n \in [0,1)$, $\gamma_n \in (0, \frac{1}{\|AA^*\|})$;
set $n = 1, l = -1$;
repeat
set $x_n^0 = x_n$, $z = x_{n-1}$;
  for $k = 0 : N-1$;
    set v_n^k to be a nonascending vector for Φ at x_n^k;
    set $l = l+1$;
    when $\Phi(x_n^k + ac^l v_n^k) \geq \Phi(x_n^k)$
      $l = l+1$;
    end
    set $x_n^{k+1} = x_n^k + ac^l v_n^k$;
  end
set $x_n = x_n^N$;
compute $y_n = x_n + \theta_n(x_n - z)$;
set $z_n^{'} = y_n^{'} + \gamma_n A^*(J_\varsigma^{B_2}(Ay_n^{'} - \varsigma D_2(Ay_n^{'})C_2 Ay_n^{'}) - Ay_n^{'})$;
set $x_{n+1} = \alpha_n f(y_n^{'}) + (1-\alpha_n) J_\lambda^{B_1}(z_n^{'} - \lambda D_1(z_n^{'})C_1 z_n^{'})$;
set $n = n+1$.

Superiorized version of algorithm (3.23) with restarted perturbations

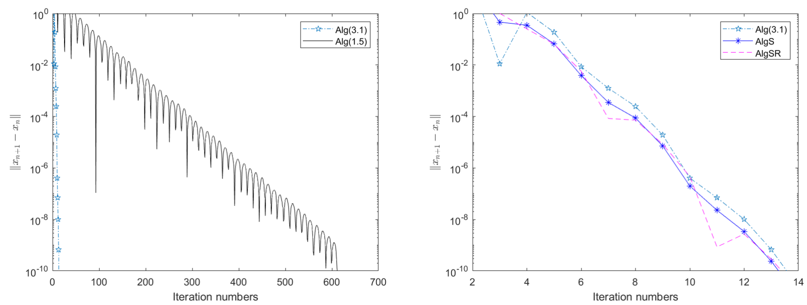
Given x_0, x_1, $c \in (0,1)$, $a > 0$, $\{\alpha_n\} \subset (0,1)$, $\theta_n \in [0,1)$, $\gamma_n \in (0, \frac{1}{\|AA^*\|})$, $N \in N_+$, and a sequence of positive intergers $\{W_r\}_{r=0}^{\infty}$.
set $n = 1, l = -1, w = 0, r = 0$;
repeat
set $x_n^0 = x_n$, $z = x_{n-1}$;
  for $k = 0 : N-1$;
    set v_n^k to be a nonascending vector for Φ at x_n^k;
    set $l = l+1$;
    While $\Phi(x_n^k + ac^l v_n^k) \geq \Phi(x_n^k)$
      set $l = l+1$;
    end
    set $x_n^{k+1} = x_n^k + ac^l v_n^k$;
  end
  set $w = w+1$;
  If $w = W_r$;
    set $r = r+1, l = r, w = 0$;
  end
set $x_n = x_n^N$;
compute $y_n = x_n + \theta_n(x_n - z)$;
set $x_{n+1} = \alpha_n f(y_n) + (1-\alpha_n)\widetilde{U}(y_n + \gamma A^*(\widetilde{T} - I)Ay_n)$;
set $n = n+1$.

3-inverse strongly monotone operator. Let $Ax = \frac{x}{5}$ $(\forall\, x \in \mathbb{R})$. Then $A : H_1 \to H_2$ is a bounded linear operator on $\mathbb{R}$ and the adjoint A^* of A is defined by $A^*y = \frac{y}{5}$ $(\forall y \in \mathbb{R})$.

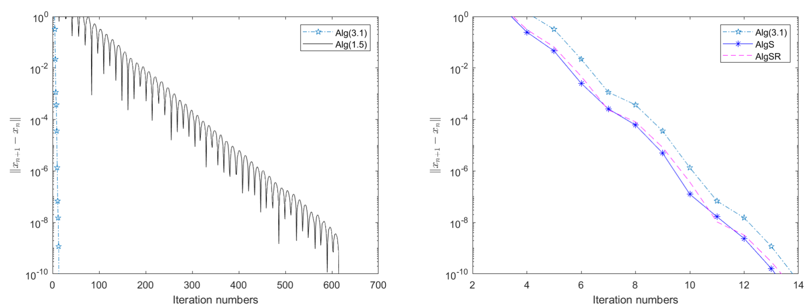
In the numerical experiments, we choose $D_1(x) = D_2(Ax) = [1 + \frac{t}{n^2}]I$ ($t \in (-1,1)$ is a constant), $\theta_n = \frac{1}{n^{1.2}+1}$, $\gamma_n = \frac{1}{25\sqrt{n}} + 0.01$, $e_n = 0$, $\lambda \in (0, 2\min\{1,3\}) = (0,2)$, $\alpha_n = \frac{1}{\sqrt{n+1}}$ and $f(x) = \frac{3x}{10}$ in algorithm (3.1) (Alg (3.1)), AlgS and AlgSR. Choose $N = 10$, $c = 0.8$, the target function $\Phi(x) = \frac{1}{2}|x|^2$, $\forall x \in \mathbb{R}$ in AlgS, AlgSR and choose $W_r = W = 30$ in AlgSR. For algorithm (1.5) (Alg (1.5)), we use the parameters provided in [9].



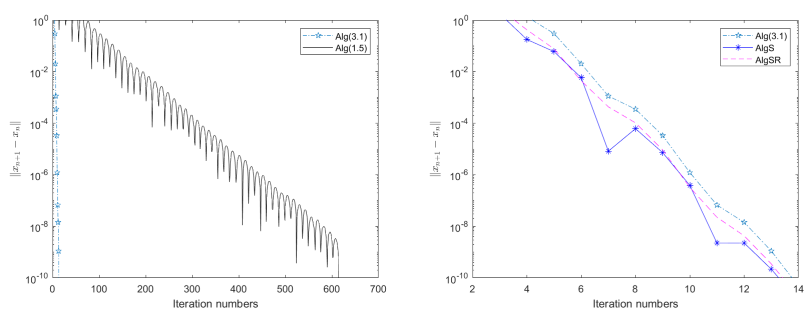
(a) Alg (1.5), Alg (3.1), AlgS and AlgSR ($x_0 = 37$, $x_1 = 68$)



(b) Alg (1.5), Alg (3.1), AlgS and AlgSR ($x_0 = -45$, $x_1 = -82$)



(c) Alg (1.5), Alg (3.1), AlgS and AlgSR ($x_0 = 105$, $x_1 = -127$)



(d) Alg (1.5), Alg (3.1), AlgS and AlgSR ($x_0 = -93$, $x_1 = 118$)

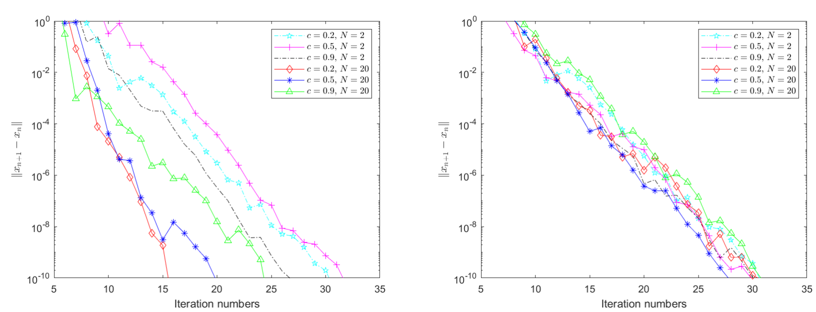
FIGURE 1. The number of iterations with the different initial values

Figure 1 (a)-(d) illustrate the numerical results of Alg (1.5), Alg (3.1), AlgS and AlgSR with the different initial values and the different stopping criterions. From the left figures of (a)-(d), we can see that Alg (3.1), AlgS and AlgSR have much fewer iterations than Alg (1.5) under the same stopping criterions. We compare the numerical performance of the three algorithms Alg (3.1), AlgS and AlgSR in the right figures of (a)-(d). It is reported that the three algorithms have the same iteration numbers when the stopping criterion is 10^{-10}. AlgSR or AlgS has better numerical performance than the other two algorithms under some stopping criterions.

Example 4.2. Let $H_1 = H_2 = l_2(\mathbb{R})$, where $l_2(\mathbb{R}) = \{x = (x_1, x_2, \cdots, x_i, \cdots) | x_i \in \mathbb{R}, \sum_{i=1}^{\infty} |x_i|^2 < \infty\}$ with the norm $\|x\| = \sqrt{\sum_{i=1}^{\infty} |x_i|^2}$ $(\forall x \in l_2(\mathbb{R}))$. Let $B_1 x = 2x,\ B_2 x = 5x$ $(\forall x \in l_2(\mathbb{R}))$. Clearly, B_1 and B_2 are maximal monotone operators from $l_2(\mathbb{R})$ to $l_2(\mathbb{R})$. Let $C_1 x = (\frac{x_1+|x_1|}{2}, \frac{x_2+|x_2|}{2}, \frac{x_3+|x_3|}{2}, \cdots)$, $C_2 x = 3x$ $(\forall x \in l_2(\mathbb{R}))$. It is easy to verify that C_1 is 1-inverse strongly monotone and C_2 is 3-inverse strongly monotone. Define a bounded linear operaor A from H_1 to H_2 as $Ax = (0, x_1, \frac{x_2}{2}, \frac{x_3}{3}, \cdots)$ $(\forall x \in l_2(\mathbb{R}))$. Then the adjoint operator A^* of A is $A^* y = (y_2, \frac{y_3}{2}, \frac{y_4}{3}, \cdots)$ $(\forall y \in l_2(\mathbb{R}))$.

In the following numerical experiments, we select $D_1(x) = D_2(Ax) = [1 - \frac{t}{n^2}]I$ ($t \in (-1, 1)$ is a constant), $\theta_n = \frac{1}{n^2}$, $\gamma_n = 0.001 + \frac{1}{\|AA^*\| n^{\frac{1}{2}}}$, $\lambda, \varsigma \in (0, 2\min\{1, 3\}) = (0, 2)$ and $\alpha_n = \frac{1}{2n+1}$ in Alg (3.1), AlgS and AlgSR. We define a target function $\Phi(x) = \frac{1}{2}\|x\|^2$ and the nonascending direction v of the function Φ at point x as $v = -\frac{x}{\|x\|}$ if $\|x\| \neq 0$ or $v = 0$, otherwise in AlgS and AlgSR. We use the parameters given in [9] for Alg (1.5). Choose the initial values $x_0 = (\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \cdots)$, $x_1 = (1, 2, 3, \cdots)$.

We first consider how the kernel c of the summable perturbations and the inner loop number N affect the iterations of AlgS and AlgSR. Then we report the numerical results of AlgSR with the different W. Finally, we compare the numerical performance of Alg (1.5), Alg (3.1), AlgS and AlgSR.

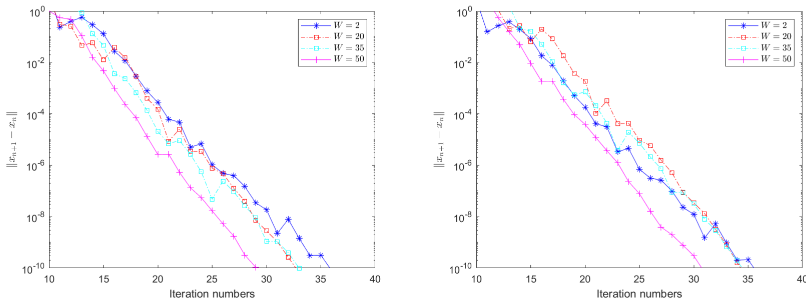


(a) AlgS (b) AlgSR

FIGURE 2. Comparison of iteration steps of AlgS and AlgSR with different c and N

In Figure 2, we choose the contraction operator $f = 0.7$, the number of consecutive step-sizes of the restart $W = 50$, the stopping criterion $\varepsilon = 10^0, 10^{-2}, \cdots, 10^{-10}$ and set $c = 0.2, 0.5, 0.9$, $N = 2, 20$, respectively. It can be seen that the iteration numbers of AlgS is greatly affected by the different values of $c,\ N$ and it has the least number of iterations when $c = 0.2,\ N = 20$. In contrast to this, the iteration numbers of AlgSR is less affected by the values of $c,\ N$ and it has the least number of iterations when $c = 0.5,\ N = 20$.

In Figure 3, we choose the contraction operator $f = 0.7$, the stopping criterion $\varepsilon = 10^0, 10^{-2}, \cdots, 10^{-10}$. Set $W = 2, 20, 35, 50$, respectively. We compare the number of iterations of AlgSR with the different values of W in two cases: $c = 0.5, N = 20$ and $c = 0.9, N = 2$. It can be seen that AlgSR has better convergence performance with $W = 50$ in the both cases.



(a) AlgSR ($c = 0.5, N = 20$) (b) AlgSR ($c = 0.9, N = 2$)

FIGURE 3. Comparison of iteration numbers of AlgSR under different c, N and W

TABLE 1. Numerical results for different contraction operator $f(x)$

$f(x)$	Alg (1.5)		Alg (3.1)		AlgS			AlgSR		
	Iter.	Sec.	Iter.	Sec.	Iter.	Sec.	Φ.	Iter.	Sec.	Φ.
$0.1x$	6113	0.3186	39	0.0129	35	0.0332	0	29	0.0257	0
$0.5x$	6113	0.3129	41	0.0120	36	0.0333	0	34	0.0261	0
$0.9x$	6113	0.3232	43	0.0133	36	0.0299	0	36	0.0277	0

In Table 1, we choose $c = 0.9,\ N = 2$, $W = 50$, the stopping criterion $\varepsilon = 10^{-12}$ and compare the numerical performance of Alg (1.5), Alg (3.1), AlgS and AlgSR when $f = 0.1,\ 0.5,\ 0.9$, respectively. The experiments show that the proposed algorithms Alg (3.1), AlgS and AlgSR have obviously advantage than Alg (1.5) in decreasing the number of iterations and the running time whatever the value f takes. Especially, the superiorized algorithm with restarted perturbation has the minimum number of iteration while running relatively little compute time.

5. CONCLUSION

In this paper, we proposed an inexact scaled forward-backward algorithm for solving the split monotone variational inclusion problem and proved the strong convergence of the generated sequence under some appropriate conditions. We also discussed the bounded perturbation resilience of the exact version of it and introduced the superiorization algorithm as well as the superiorization algorithm with restarted perturbations. In numerical experiments, we illustrated the effectiveness of the proposed algorithms and showed the effect of the kernel c, inner loop number N and the number of restarted steps W on the number of iterations. The numerical results show that choosing of appropriate W may effectively reduce the number of iterations.

STATEMENTS AND DECLARATIONS

The authors declare that they have no conflict of interest, and the manuscript has no associated data.

ACKNOWLEDGMENTS

The authors would like to acknowledge the reviewers for the very valuable comments which helped to improve the presentation of this paper.

References

[1] Y. Censor and T. Elfving. A multiprojection algorithm using Bregman projections in a product space. *Numerical Algorithms*, 8: 221-239, 1994.

[2] N. Pakkaranang, P. Kumam, Y. I. Suleiman, and B. Ali. Bounded perturbation resilience of viscosity proximal algorithm for solving split variational inclusion problems with applications to compressed sensing and image recovery. *Mathematical Methods in the Applied Sciences*, 45(8): 4085-4107, 2022.

[3] Y. Censor, T. Bortfeld, B. Martin, and A. Trofimov. A unified approach for inversion problems in intensity-modulated radiation therapy. *Physics in Medicine and Biology*, 51(10): 2353-2365, 2006.

[4] C. Byrne. A unified treatment of some iterative algorithms in signal processing and image reconstruction. *Inverse Problems*, 20(1): 103-120, 2004.

[5] A. Moudafi. Split monotone variational inclusions. *Journal Of Optimization Theory and Applications*, 150(3): 275-283, 2011.

[6] X. Zhao, J.-C. Yao, and Y. Yao. A proximal algorithm for solving split monotone variational inclusions. *University Politehnica of Bucharest Scientific Bulletin - Series A - Applied Mathematics and Physics*, 82(3): 43-52, 2020.

[7] Z. Zhou, B. Tan, and S. Li. Adaptive hybrid steepest descent algorithms involving an inertial extrapolation term for split monotone variational inclusion problems. *Mathematical Methods in the Applied Sciences*, 45(15): 8835-8853, 2022.

[8] Y. Yao, Y. Shehu, X. H. Li, and Q. L. Dong. A method with inertial extrapolation step for split monotone inclusion problems. *Optimization*, 70(4): 741-761, 2021.

[9] C. Izuchukwu, S. Reich, and Y. Shehu. Relaxed inertial methods for solving the split monotone variational inclusion problem beyond co-coerciveness. *Optimization*, 72(2): 607-646, 2023.

[10] Y. Shehu and F. U. Ogbuisi. An iterative method for solving split monotone variational inclusion and fixed point problems. *Revista De La Real Academia De Ciencias Exactas Fisicas Y Naturales Serie A-matematicas*, 110(2): 503-518, 2016.

[11] K. R. Kazmi, R. Ali, and M. Furkan. Hybrid iterative method for split monotone variational inclusion problem and hierarchical fixed point problem for a finite family of non-expansive mappings. *Numerical Algorithms*, 79(2): 499-527, 2018.

[12] J. L. Guan, L. C. Ceng, and B. Hu. Strong convergence theorem for split monotone variational inclusion with constraints of variational inequalities and fixed point problems. *Journal of Inequalities and Applications*, 2018: 1-29, 2018.

[13] Z. Wang, X. Long, Z. Lei, and Z. Chen. New self-adaptive methods with double inertial steps for solving splitting monotone variational inclusion problems with applications. *Communications in Nonlinear Science and Numerical Simulation*, 114, 2022.

[14] S. Bonettini, R. Zanella, and L. Zanni. A scaled gradient projection method for constrained image deblurring. *Inverse Problems*, 25(1): 015002, 2009.

[15] S. Bonettini and M. Prato. New convergence results for the scaled gradient projection method. *Inverse Problems*, 31(9): 095008, 2015.

[16] D. Li, M. Lamoureux, and W. Liao. Incorporating multiple a priori information for inverse problem by inexact scaled gradient projection. *Journal of Computational and Applied Mathematics*, 437: 115460, 2024,

[17] Y. Censor, R. Davidi, and T. G. Herman. Perturbation resilience and superiorization of iterative algorithms. *Inverse Problems*, 26(6): 065008, 2010.

[18] M. Guenter, S. Collins, A. Ogilvy, W. Hare, and A. Jirasek. Superiorization versus regularization: A comparison of algorithms for solving image reconstruction problems with applications in computed tomography. *Medical Physics*, 49(2): 1065-1082, 2022.

[19] T. Nikazad, M. Abbasi, L. Afzalipour, and T. Elfving. A new step size rule for the superiorization method and its application in computerized tomography. *Numerical Algorithms*, 90(3): 1253-1277, 2022.

[20] Y. Censor, R. Davidi, G. T. Herman, R.W. Schulte, and L. Tutreashvili. Projected subgradient minimization versus superiorization. *Journal of Optimization Theory and Applications*, 160(3): 730-747, 2014.

[21] R. Davidi, Y. Censor, R. W. Schulte, S. Geneser, and L. Xing. Feasibility-seeking and superiorization algorithms applied to inverse treatment planning in radiation therapy. *Communications In Contemporary Mathematics*, 636: 83-92, 2015.

[22] F. J. Aragón-Artacho, Y. Censor, A. Gibali, and D. T.-B. The superiorization method with restarted perturbations for split minimization problems with an application to radiotherapy treatment planning. *Applied Mathematics and Computation*, 440: 127627, 2023.

[23] T. Nikazad, R. Davidi, and G. T. Herman. Accelerated perturbation-resilient block-iterative projection methods with application to image reconstruction. *Inverse Problems*, 28(3): 035005, 2012.

[24] Y. Censor. Can linear superiorization be useful for linear optimization problems? *Inverse Problems*, 33(4): 044006, 2017.

[25] M. Ertürk and A. Salkım. Superiorization and bounded perturbation resilience of a gradient projection algorithm solving the convex minimization problem. *Optimization Letters*, 17(8): 1957-1978, 2023.

[26] H. K. Xu. Iterative methods for the split feasibility problem in infinite-dimensional Hilbert spaces. *Inverse Problems*, 26(10): 105018, 2010.
[27] G. Crombez. A hierarchical presentation of operators with fixed points on Hilbert spaces. *Numerical Functional Analysis and Optimization*, 27(3-4): 259-277, 2006.
[28] H. H. Bauschke and P. L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces.* CMS Books in Mathematics, 2011.
[29] K. Goebel and S. Reich. *Uniform convexity, hyperbolic geometry, and non-expansive mappings.* Marcel Dekker Inc., New York, 1984.
[30] H. K. X. Iterative algorithms for nonlinear operators. *Journal Of The London Mathematical Society*, 66(1): 240-256, 2002.