J. Nonlinear Var. Anal. 5 (2021), No. 1, pp. 43-57
Available online at http://jnva.biemdas.com
https://doi.org/10.23952/jnva.5.2021.1.04

A NEW INERTIAL SELF-ADAPTIVE ALGORITHM FOR SPLIT COMMON FIXED-POINT PROBLEMS

JING ZHAO*, YUAN LI

College of Science, Civil Aviation University of China, Tianjin 300300, China

Abstract. In this paper, we study the split common fixed-point problem of quasi-nonexpansive operators in Hilbert space. We establish a weak convergence theorem of the proposed iterative algorithm, which combines the primal-dual method and the inertial method. In our algorithm, the step sizes are chosen self-adaptively so that the implementation of the algorithm does not need any prior information about bounded linear operator norms. Finally, numerical results are included to illustrate the efficiency of the proposed algorithm.

Keywords. Split common fixed-point problem; Quasi-nonexpansive operator; Inertial technique; Self-adaptive algorithm; Weak convergence.

1. Introduction

The split common fixed point problem (SCFP), introduced by Censor and Segal [1], is an inverse problem that aims to find an element in a fixed point set such that its image under a linear transformation belongs to another fixed point set. More specifically, given two real Hilbert spaces H_1 and H_2, the SCFP consists of finding $x \in H_1$ such that

$$x \in F(U),\ Ax \in F(T), \tag{1.1}$$

where $A : H_1 \to H_2$ is a bounded linear operator, while $F(U)$ and $F(T)$ stand for the fixed point sets of $U : H_1 \to H_1$ and $T : H_2 \to H_2$, respectively.

In particular, if U and T are both metric projections, then the SCFP is reduced to the well-known split feasibility problem (SFP), which includes many convex optimization problems as special cases, and plays an important role in medical image reconstruction and in signal processing (see, e.g., [2, 3, 4, 5, 6]). The SFP was first introduced by Censor and Elfving [7] and it can be formulated as the problem of finding a point $x \in H_1$ such that

$$x \in C,\ Ax \in Q, \tag{1.2}$$

where $C \subseteq H_1$ and $Q \subseteq H_2$ are the nonempty closed convex sets. As an optimization problem, the SFP can be used to solve airport aircraft taxiway route optimization. Recently, many efficient iterative algorithms for solving the SFP have been proposed and analyzed (see, e.g., [8, 9, 10, 11, 12] and the references therein).

*Corresponding author.
E-mail addresses: zhaojing200103@163.com (J. Zhao), 1754565922@qq.com (Y. Li).
Received September 12, 2020; Accepted November 26, 2020.

One of the known methods for solving the SFP is Byrne's CQ algorithm [3], which generates a sequence $\{x_k\}$ via

$$x_{k+1} = P_C(I - \gamma_k A^*(I - P_Q)A)x_k \tag{1.3}$$

for each $k \in \mathbb{N}$, where P_C and P_Q are the (orthogonal) projection onto C and Q, respectively, $\gamma_k \in (0, \frac{2}{\lambda})$ with λ being the spectral radius of the operator A^*A.

As a generalization of the SFP, the SCFP has been successfully applied in the field of intensity-modulated radiation therapy (IMRT) treatment planning. We note that, when projection operators P_C and P_Q are replaced by directed operators, CQ-algorithm (1.3) is reduced to the following iterative scheme:

$$x_{k+1} = U(x_k - \gamma_k A^*(I - T)Ax_k), \tag{1.4}$$

which was proposed by Censor and Segal [1] for solving the SCFP (1.1) of directed operators. Since then, many authors have introduced various iterative methods for solving the SCFP; see, for example, [13, 14, 15, 16, 17]. We note that many iterative algorithms for solving the SCFP were motivated and proposed by the works of the SFP.

For solving the SCFP (1.1) of quasi-nonexpansive operators, Moudafi [13] introduced the following relaxed algorithm:

$$x_{k+1} = (1 - \alpha_k)u_k + \alpha_k U(u_k), \;\; k \in N, \tag{1.5}$$

where $u_k = x_k + \gamma\beta A^*(T - I)Ax_k$, $\beta \in (0,1)$, $\alpha_k \in (0,1)$ and $\gamma \in (0, \frac{1}{\eta\beta})$ with η being the spectral radius of the operator A^*A. Moudafi proved the weak convergence of algorithm (1.5) in Hilbert spaces.

In [18], based on the idea in [19], the following general iterative algorithm with the dual variable has been proposed for the SFP (1.2)

$$\begin{cases} \omega_{k+1} = (I - P_C)(x_k - \gamma_k A^*(I - P_Q)Ax_k + (1-\lambda)\omega_k), \\ x_{k+1} = x_k - \gamma_k A^*(I - P_Q)Ax_k - \lambda\omega_{k+1}, \end{cases} \tag{1.6}$$

where $\{\gamma_k\} \subset (0, \frac{2}{\|A\|^2})$, and $\lambda \in (0,1]$. If $\lambda = 1$, then algorithm (1.6) is reduced to CQ algorithm (1.3). So, it generalizes CQ algorithm (1.3) for solving the SFP (1.1).

It is observed that, the weak convergence of the CQ method is guaranteed under the assumption that $\{\gamma_k\} \subset (0, \frac{2}{\|A\|^2})$. So, the implementation of algorithms requires a norm estimation of the bounded linear operator A, or the spectral radius of the matrix A^TA in finite-dimensional framework. It is not always easy in practice to compute (or, at least, estimate) the matrix norm of A. To overcome this difficulty, there have been many works to select the stepsize without any prior information of the matrix norm (see, for example, [20, 21, 22]). Lopez et al. [23] introduced a modified CQ method by replacing the step-size $\{\gamma_k\}$ in (1.3) with the following adaptive step:

$$\gamma_k := \frac{\rho_k f(x_k)}{\|\nabla f(x_k)\|^2},$$

where $\rho_k \in (0,4)$, $f(x_k) = \frac{1}{2}\|(I - P_Q)Ax_k\|^2$ and $\nabla f(x_k) = A^*(I - P_Q)Ax_k$ for all $k \geq 1$.

Motivated by algorithm (1.6), the following self-adaptive iterative algorithm [24] with the dual variable was proposed for the SCFP (1.1) of directed operators:

$$\begin{cases} \omega_{k+1} = (I - U)(x_k - \gamma_k A^*(I - T)Ax_k + (1-\lambda)\omega_k), \\ x_{k+1} = x_k - \gamma_k A^*(I - T)Ax_k - \lambda\omega_{k+1}, \end{cases} \tag{1.7}$$

where the stepsize γ_k is chosen by

$$\gamma_k := \begin{cases} \frac{\rho_k \|(I-T)Ax_k\|^2}{\|A^*(I-T)Ax_k\|^2}, & (I-T)Ax_k \neq 0, \\ \gamma, & (I-T)Ax_k = 0 \end{cases}$$

with $0 < \rho_k < 2$ and $\gamma > 0$. We note that algorithm (1.7) generalizes algorithm (1.4).

To speed up the convergence rate, Polyak [25] first introduced the heavy ball method of the two-order time dynamical system, which is a two-step iterative method for minimizing a smooth convex function f. To improve the convergence rate, Nesterov [26] proposed a modified heavy ball method as follows

$$\begin{cases} y_k = x_k + \alpha_k(x_k - x_{k-1}), \\ x_{k+1} = y_k - \lambda_k \nabla f(y_k), \end{cases}$$

where $\alpha_k \in [0,1)$ is an inertial factor and $\{\lambda_k\}$ is a positive sequence. Here, the inertial is represented by the term $\alpha_k(x_k - x_{k-1})$. It is remarkable that inertial methodology makes use of the previous two iterates such that the performance of the algorithm is improved greatly. It was shown that inertial methods, such as, [8, 27, 28, 29], greatly improved the performance of their original (non-inertial) algorithms, that is, $\alpha_k = 0$.

In order to overcome the difficulty of computing bounded linear operator norms, Gibali et al. [30] recently introduced a self-adaptive inertial relaxed CQ algorithm for solving the SFP in real Hilbert spaces. For solving the multiple-sets split feasibility problems, Suantai et al. [31] based on the inertial technique with the self-adaptive method established strong convergence theorems via a relaxed CQ algorithm with Halpern's iteration. For recent other inertial iterative algorithms for SFP and the SCFP, we can refer to [32, 33, 34].

Inspired and motivated by the above results, for solving the SCFP (1.1) of quasi-nonexpansive operators, we construct a new self-adaptive iterative algorithm by combining inertial effects and the primal-dual method. The contents of this paper are organized as follows. First, we give some useful definitions and results for the convergence analysis of the iterative algorithm in Section 2. Second, we prove a weak convergence theorem of the proposed self-adaptive algorithm with the dual variable in Section 3. Some related results for solving the SFP (1.2) are also obtained in this section. Finally, numerical experiment is provided to illustrate the effectiveness of our proposed algorithm in Section 4, the last section.

2. Preliminaries

In this paper, we denote the inner product by $\langle \cdot, \cdot \rangle$ and the norm by $\|\cdot\|$. We use $\to$ and $\rightharpoonup$ to denote the strong convergence and weak convergence, respectively. We use $\omega_w(x_k) = \{x : \exists x_{k_j} \rightharpoonup x\}$ to stand for the weak ω-limit set of $\{x_k\}$.

Definition 2.1. An operator $T : H \to H$ is said to be

(i) *nonexpansive* if, for all $x, y \in H$,

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

(ii) *firmly nonexpansive* if, for all $x, y \in H$, $2T - I$ is nonexpansive or, equivalently,

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

(iii) *quasi-nonexpansive* if $F(T) \neq \emptyset$ and

$$\|Tx - q\| \leq \|x - q\|$$

for all $x \in H$ and $q \in F(T)$;

(iv) *firmly quasi-nonexpansive* (also called a directed operator) if $F(T) \neq \emptyset$ and

$$\langle x-q, Tx-q\rangle \geq \|Tx-q\|^2$$

for all $x \in H$ and $q \in F(T)$;

(v) *demiclosed* at the origin if, for any sequence $\{x_n\}$ which weakly converges to x, the sequence $\{Tx_n\}$ strongly converges to 0, then $Tx = 0$.

Recall that the metric (nearest point) projection from H onto a closed and convex nonempty subset C of H, denoted by P_C, is defined as follows: for each $x \in H$,

$$P_C(x) = arg\min_{y\in C}\{\|x-y\|\}.$$

A useful and simple norm equality is the following:

$$\|\alpha x+(1-\alpha)y\|^2 = \alpha\|x\|^2+(1-\alpha)\|y\|^2-\alpha(1-\alpha)\|x-y\|^2$$

for all x, $y \in H$ and $\alpha \in \mathbb{R}$.

Lemma 2.1. [13] *Let H be a real Hilbert space, and let $T : H \to H$ be a quasi-nonexpansive operator. Set $T_\alpha = (1-\alpha)I+\alpha T$ for $\alpha \in [0,1)$. Then the following properties are reached for $(x,q) \in H \times F(T)$:*

(i) $\langle x-Tx, x-q\rangle \geq \frac{1}{2}\|x-Tx\|^2$ and $\langle x-Tx, q-Tx\rangle \leq \frac{1}{2}\|x-Tx\|^2$;

(ii) $\|T_\alpha x-q\|^2 \leq \|x-q\|^2-\alpha(1-\alpha)\|Tx-x\|^2$;

(iii) $\langle x-T_\alpha x, x-q\rangle \geq \frac{\alpha}{2}\|x-Tx\|^2$.

Remark 2.1. Let $T_\alpha = (1-\alpha)I+\alpha T$, where $T : H \to H$ is a quasi-nonexpansive operator and $\alpha \in [0,1)$. We have $F(T_\alpha) = F(T)$ and $\|T_\alpha x-x\|^2 = \alpha^2\|Tx-x\|^2$. It follows from (ii) of Lemma 2.1 that $\|T_\alpha x-q\|^2 \leq \|x-q\|^2-\frac{1-\alpha}{\alpha}\|T_\alpha x-x\|^2$, which implies that T_α is firmly quasi-nonexpansive when $\alpha = \frac{1}{2}$. On the other hand, if $\widehat{T}$ is a firmly quasi-nonexpansive operator, we can easily obtain $\widehat{T} = \frac{1}{2}I+\frac{1}{2}T$, where T is quasi-nonexpansive.

It follows from Lemma 2.1 (iii) that the following proposition can be obtained easily.

Proposition 2.1. *Let $T : H \to H$ be a quasi-nonexpansive operator and $\alpha \in [0,1)$. If $T_\alpha = (1-\alpha)I+\alpha T$, then*

$$\|(I-T_\alpha)x\|^2 \leq 2\alpha\langle x-q, (I-T_\alpha)x\rangle$$

for all $(x,q) \in H \times F(T)$.

Lemma 2.2. [8] *Let the sequences $\{\phi_k\}_{k=1}^\infty \subset [0,\infty)$ and $\{\delta_k\}_{k=1}^\infty \subset [0,\infty)$ satisfy*

(i) $\phi_{k+1}-\phi_k \leq \theta_k(\phi_k-\phi_{k-1})+\delta_k$,

(ii) $\Sigma_{k=1}^\infty \delta_k < \infty$,

(iii) $\{\theta_k\} \subset [0,\theta]$, where $\theta \in [0,1)$.

Then $\{\phi_k\}$ is a converging sequence and $\Sigma_{k=1}^\infty[\phi_{k+1}-\phi_k]_+ < \infty$, where $[t]_+ = \max\{t,0\}$ for any $t \in \mathbb{R}$.

Lemma 2.3. [10] *Let K be a closed and convex nonempty subset of a real Hilbert space H. Let $\{x_k\}$ be a bounded sequence, which satisfies the following properties:*

(a) *every weak limit point of $\{x_k\}$ lies in K;*

(b) $\lim_{k\to\infty}\|x_k-x\|$ *exists for every $x \in K$.*

Then $\{x_k\}$ converges weakly to a point in K.

Lemma 2.4. [28, 35] *Let K be a closed and convex nonempty subset of real Hilbert space H. Let $T : K \to K$ be a nonexpansive mapping. Then $I - T$ is demiclosed at origin.*

3. The Self-adaptive Inertial Iterative Algorithm

In this section, we use the dual variable combined with the inertial technique to establish a new self-adaptive iterative algorithm for solving the SCFP (1.1) of quasi-nonexpansive operators. We make use of the following assumptions:

(A1) $U : H_1 \to H_1$ and $T : H_2 \to H_2$ are quasi-nonexpansive operators. $U_{\alpha_1} = (1-\alpha_1)I + \alpha_1 U$ and $T_{\alpha_2} = (1-\alpha_2)I + \alpha_2 T$, where $\alpha_1, \alpha_2 \in [0,1)$. $A : H_1 \to H_2$ is a bounded linear operator such that $A \neq 0$;

(A2) Γ denotes the solution set of the SCFP (1.1) and Γ is nonempty.

Algorithm 3.1. (Self-adaptive inertial iterative algorithm)
Initialization: Choose $\alpha_1 \in (0, \frac{1}{2}]$, $\alpha_2 \in (0,1)$, and two positive sequences $\{\rho_k\}_{k=1}^{\infty}$ and $\{\varepsilon_k\}_{k=1}^{\infty} \subset [0,\infty)$ satisfying

$$0 < \rho_k < \frac{1}{\alpha_2}$$

and

$$\sum_{k=1}^{\infty} \varepsilon_k < \infty.$$

Select arbitrary starting points x_0, x_1, $\omega_0 \in H_1$, $\eta \in [0,1)$, $\{\lambda_k\} \subseteq (0,1]$, $\gamma > 0$, and set $k = 1$, $\omega_1 = \omega_0$.
Iterative step: For $k \geq 1$, given the iterates x_{k-1}, x_k and ω_{k-1}, choose β_k such that $0 \leq \beta_k \leq \bar{\beta}_k$, where

$$\bar{\beta}_k := \begin{cases} \min\{\eta, \frac{\varepsilon_k}{\|x_k - x_{k-1}\|^2 + \|\omega_{k-1}\|^2}\}, & if\ x_k \neq x_{k-1}\ or\ \omega_{k-1} \neq 0, \\ \eta, & otherwise. \end{cases} \tag{3.1}$$

Compute

$$\begin{cases} y_k = x_k + \beta_k(x_k - x_{k-1}), \\ \omega_{k+1} = (I - U_{\alpha_1})(y_k - \gamma_k A^*(I - T_{\alpha_2})Ay_k + (1-\lambda_{k+1})\omega_k), \\ x_{k+1} = y_k - \gamma_k A^*(I - T_{\alpha_2})Ay_k - \lambda_{k+1}\omega_{k+1}, \end{cases}$$

where the stepsize γ_k is chosen in such a way that

$$\gamma_k := \begin{cases} \frac{\rho_k \|(I - T_{\alpha_2})Ay_k\|^2}{\|A^*(I - T_{\alpha_2})Ay_k\|^2}, & (I - T_{\alpha_2})Ay_k \neq 0, \\ \gamma, & (I - T_{\alpha_2})Ay_k = 0. \end{cases} \tag{3.2}$$

Remark 3.1. In Algorithm 3.1, the inertial extrapolation factor β_k and the stepsize γ_k are chosen via a self-adaptive way. We give the way of selecting the stepsizes such that the implementation of the algorithm does not need any prior information about the norm of the bounded linear operator.

Remark 3.2. From (3.1), we have that

$$\beta_k(\|x_k - x_{k-1}\|^2 + \|\omega_{k-1}\|^2) \leq \bar{\beta}_k(\|x_k - x_{k-1}\|^2 + \|\omega_{k-1}\|^2) \leq \varepsilon_k$$

and then

$$\Sigma_{k=1}^{\infty} \beta_k(\|x_k - x_{k-1}\|^2 + \|\omega_{k-1}\|^2) < \infty. \tag{3.3}$$

For example, we take $\varepsilon_k = \frac{1}{k^{1.5}}$, i.e.,

$$\bar{\beta}_k := \begin{cases} \min\{\eta, \frac{1}{k^{1.5}(\|x_k-x_{k-1}\|^2+\|\omega_{k-1}\|^2)}\}, & if\ x_k \neq x_{k-1}\ or\ \omega_{k-1} \neq 0, \\ \eta, & otherwise. \end{cases}$$

Hence,

$$\Sigma_{k=1}^{\infty}\beta_k(\|x_k - x_{k-1}\|^2 + \|\omega_{k-1}\|^2) < \infty.$$

From the following lemma, we will see that $\{\gamma_k\}$ is well-defined.

Lemma 3.1. *$\{\gamma_k\}$ defined by (3.2) is well-defined.*

Proof. Taking $z \in \Gamma$, i.e., $z \in F(U)$ and $Az \in F(T)$, we have

$$\begin{aligned} \|A^*(I-T_{\alpha_2})Ay_k\| \cdot \|y_k - z\| &\geq \langle A^*(I-T_{\alpha_2})Ay_k, y_k - z\rangle \\ &= \langle (I-T_{\alpha_2})Ay_k, Ay_k - Az\rangle \\ &\geq \frac{1}{2\alpha_2}\|(I-T_{\alpha_2})Ay_k\|^2. \end{aligned}$$

Consequently, we have $\|A^*(I-T_{\alpha_2})Ay_k\| > 0$ when $\|(I-T_{\alpha_2})Ay_k\| \neq 0$. This leads that $\{\gamma_k\}$ is well-defined. □

Lemma 3.2. *Let $\{(\omega_k, x_k)\}$ be the sequence generated by Algorithm 3.1. Then, for any $z \in \Gamma$, the following inequality holds*

$$\begin{aligned} &\|x_{k+1} - z\|^2 + \lambda_{k+1}\|\omega_{k+1}\|^2 \\ &\leq \|y_k - z\|^2 + \lambda_{k+1}\|\omega_k\|^2 - \lambda_{k+1}^2\|\omega_k\|^2 - \lambda_{k+1}(1-\lambda_{k+1})\|\omega_{k+1} - \omega_k\|^2 \\ &\quad - \lambda_{k+1}(\frac{1}{\alpha_1} - 2)\|\omega_{k+1}\|^2 - \gamma_k\big(\frac{1}{\alpha_2}\|(I-T_{\alpha_2})Ay_k\|^2 - \gamma_k\|A^*(I-T_{\alpha_2})Ay_k\|^2\big). \end{aligned}$$

Proof. Denoting

$$u_k = y_k - \gamma_k A^*(I-T_{\alpha_2})Ay_k,$$

we have $\omega_{k+1} = (I-U_{\alpha_1})(u_k + (1-\lambda_{k+1})\omega_k)$ and $x_{k+1} = u_k - \lambda_{k+1}\omega_{k+1}$. Taking $z \in \Gamma$, we have $z \in F(U)$ and $Az \in F(T)$. So, $z \in F(U_{\alpha_1})$ and $Az \in F(T_{\alpha_2})$. It follows from Algorithm 3.1 and Proposition 2.1 that

$$\|\omega_{k+1}\|^2 \leq 2\alpha_1\langle \omega_{k+1}, u_k - z + (1-\lambda_{k+1})\omega_k\rangle$$

and

$$\|x_{k+1} - z\|^2 = \|u_k - z\|^2 - 2\lambda_{k+1}\langle u_k - z, \omega_{k+1}\rangle + \lambda_{k+1}^2\|\omega_{k+1}\|^2.$$

Thus, we have

$$\begin{aligned} &\|x_{k+1} - z\|^2 + \lambda_{k+1}\|\omega_{k+1}\|^2 \\ &= \|u_k - z\|^2 - 2\lambda_{k+1}\langle u_k - z, \omega_{k+1}\rangle + \frac{1}{\alpha_1}\lambda_{k+1}\|\omega_{k+1}\|^2 - \lambda_{k+1}(\frac{1}{\alpha_1} - 1 - \lambda_{k+1})\|\omega_{k+1}\|^2 \\ &\leq \|u_k - z\|^2 - 2\lambda_{k+1}\langle u_k - z, \omega_{k+1}\rangle + 2\lambda_{k+1}\langle u_k - z + (1-\lambda_{k+1})\omega_k, \omega_{k+1}\rangle \\ &\quad - \lambda_{k+1}(\frac{1}{\alpha_1} - 1 - \lambda_{k+1})\|\omega_{k+1}\|^2 \\ &= \|u_k - z\|^2 + 2\lambda_{k+1}(1-\lambda_{k+1})\langle \omega_k, \omega_{k+1}\rangle - \lambda_{k+1}(\frac{1}{\alpha_1} - 1 - \lambda_{k+1})\|\omega_{k+1}\|^2. \end{aligned}$$

Since $2\langle \omega_{k+1}, \omega_k \rangle = \|\omega_{k+1}\|^2 - \|\omega_{k+1} - \omega_k\|^2 + \|\omega_k\|^2$, we obtain

$$\begin{aligned} &\|x_{k+1} - z\|^2 + \lambda_{k+1}\|\omega_{k+1}\|^2 \\ &\leq \|u_k - z\|^2 + \lambda_{k+1}(1-\lambda_{k+1})\|\omega_k\|^2 + \lambda_{k+1}(1-\lambda_{k+1})\|\omega_{k+1}\|^2 \\ &\quad - \lambda_{k+1}(\frac{1}{\alpha_1} - 1 - \lambda_{k+1})\|\omega_{k+1}\|^2 - \lambda_{k+1}(1-\lambda_{k+1})\|\omega_{k+1} - \omega_k\|^2 \\ &= \|u_k - z\|^2 + \lambda_{k+1}\|\omega_k\|^2 - \lambda_{k+1}^2\|\omega_k\|^2 - \lambda_{k+1}(1-\lambda_{k+1})\|\omega_{k+1} - \omega_k\|^2 \\ &\quad - \lambda_{k+1}(\frac{1}{\alpha_1} - 2)\|\omega_{k+1}\|^2. \end{aligned} \tag{3.4}$$

It follows from $Az \in F(T_{\alpha_2})$ and Proposition 2.1 that

$$\langle y_k - z, A^*(I - T_{\alpha_2})Ay_k \rangle = \langle Ay_k - Az, (I - T_{\alpha_2})Ay_k \rangle \geq \frac{1}{2\alpha_2}\|(I - T_{\alpha_2})Ay_k\|^2,$$

which implies that

$$\begin{aligned} \|u_k - z\|^2 &= \|y_k - z\|^2 - 2\gamma_k\langle y_k - z, A^*(I - T_{\alpha_2})Ay_k\rangle + \gamma_k^2\|A^*(I - T_{\alpha_2})Ay_k\|^2 \\ &\leq \|y_k - z\|^2 - \gamma_k\big(\frac{1}{\alpha_2}\|(I - T_{\alpha_2})Ay_k\|^2 - \gamma_k\|A^*(I - T_{\alpha_2})Ay_k\|^2\big). \end{aligned}$$

This together with (3.4) yields

$$\begin{aligned} &\|x_{k+1} - z\|^2 + \lambda_{k+1}\|\omega_{k+1}\|^2 \\ &\leq \|y_k - z\|^2 + \lambda_{k+1}\|\omega_k\|^2 - \lambda_{k+1}^2\|\omega_k\|^2 - \lambda_{k+1}(1-\lambda_{k+1})\|\omega_{k+1} - \omega_k\|^2 \\ &\quad - \lambda_{k+1}(\frac{1}{\alpha_1} - 2)\|\omega_{k+1}\|^2 - \gamma_k\big(\frac{1}{\alpha_2}\|(I - T_{\alpha_2})Ay_k\|^2 - \gamma_k\|A^*(I - T_{\alpha_2})Ay_k\|^2\big). \end{aligned}$$

The proof is completed. □

Theorem 3.1. *Let $I - U$ and $I - T$ be demiclosed at origin. Let $\{\lambda_k\} \subseteq (0,1]$ be nonincreasing,* $\liminf_{k\to\infty} \lambda_k > 0$*, and*

$$0 < \liminf_{k\to\infty} \rho_k \leq \limsup_{k\to\infty} \rho_k < \frac{1}{\alpha_2}.$$

Let $\{(\omega_k, x_k)\}$ be the sequence generated by Algorithm 3.1, then $\{x_k\}$ converges weakly to a solution $x^* \in \Gamma$ and $\{(\omega_k, x_k)\}$ converges weakly to the point $(0, x^*)$.

Proof. Let $z \in \Gamma$. From Lemma 3.2, we obtain

$$\begin{aligned} &\|x_{k+1} - z\|^2 + \lambda_{k+1}\|\omega_{k+1}\|^2 \\ &\leq \|y_k - z\|^2 + \lambda_{k+1}\|\omega_k\|^2 - \lambda_{k+1}^2\|\omega_k\|^2 - \lambda_{k+1}(1-\lambda_{k+1})\|\omega_{k+1} - \omega_k\|^2 \\ &\quad - \lambda_{k+1}(\frac{1}{\alpha_1} - 2)\|\omega_{k+1}\|^2 - \gamma_k\big(\frac{1}{\alpha_2}\|(I - T_{\alpha_2})Ay_k\|^2 - \gamma_k\|A^*(I - T_{\alpha_2})Ay_k\|^2\big) \\ &\leq \|y_k - z\|^2 + \lambda_k\|\omega_k\|^2 - \lambda_{k+1}^2\|\omega_k\|^2 - \gamma_k\big(\frac{1}{\alpha_2}\|(I - T_{\alpha_2})Ay_k\|^2 - \gamma_k\|A^*(I - T_{\alpha_2})Ay_k\|^2\big). \end{aligned} \tag{3.5}$$

On the other hand, one has

$$\begin{aligned} \|y_k - z\|^2 &= \|(1+\beta_k)(x_k - z) - \beta_k(x_{k-1} - z)\|^2 \\ &= (1+\beta_k)\|x_k - z\|^2 - \beta_k\|x_{k-1} - z\|^2 + \beta_k(1+\beta_k)\|x_k - x_{k-1}\|^2. \end{aligned}$$

This together with (3.5) yields that

$$\begin{aligned}
&\|x_{k+1}-z\|^2+\lambda_{k+1}\|\omega_{k+1}\|^2\\
\le&(1+\beta_k)\|x_k-z\|^2-\beta_k\|x_{k-1}-z\|^2+\beta_k(1+\beta_k)\|x_k-x_{k-1}\|^2+\lambda_k\|\omega_k\|^2\\
&-\lambda_{k+1}^2\|\omega_k\|^2-\gamma_k\big(\frac{1}{\alpha_2}\|(I-T_{\alpha_2})Ay_k\|^2-\gamma_k\|A^*(I-T_{\alpha_2})Ay_k\|^2\big).
\end{aligned}\tag{3.6}$$

Let $d_k=\|x_k-z\|^2+\lambda_k\|\omega_k\|^2$ for $k\ge 1$. It follows from (3.6) that

$$\begin{aligned}
d_{k+1}\le d_k&+\beta_k(\|x_k-z\|^2-\|x_{k-1}-z\|^2)+\beta_k(1+\beta_k)\|x_k-x_{k-1}\|^2\\
&-\lambda_{k+1}^2\|\omega_k\|^2-\gamma_k\big(\frac{1}{\alpha_2}\|(I-T_{\alpha_2})Ay_k\|^2-\gamma_k\|A^*(I-T_{\alpha_2})Ay_k\|^2\big),
\end{aligned}$$

which implies that

$$\begin{aligned}
&d_{k+1}-d_k\\
&\le\beta_k(\|x_k-z\|^2+\lambda_k\|\omega_k\|^2-\|x_{k-1}-z\|^2-\lambda_{k-1}\|\omega_{k-1}\|^2)+\beta_k\lambda_{k-1}\|\omega_{k-1}\|^2-\lambda_{k+1}^2\|\omega_k\|^2\\
&\quad+\beta_k(1+\beta_k)\|x_k-x_{k-1}\|^2-\gamma_k\big(\frac{1}{\alpha_2}\|(I-T_{\alpha_2})Ay_k\|^2-\gamma_k\|A^*(I-T_{\alpha_2})Ay_k\|^2\big)\\
&\le\beta_k(d_k-d_{k-1})+2\beta_k(\|x_k-x_{k-1}\|^2+\|\omega_{k-1}\|^2)\\
&\quad-\lambda_{k+1}^2\|\omega_k\|^2-\gamma_k\big(\frac{1}{\alpha_2}\|(I-T_{\alpha_2})Ay_k\|^2-\gamma_k\|A^*(I-T_{\alpha_2})Ay_k\|^2\big).
\end{aligned}\tag{3.7}$$

For the case $(I-T_{\alpha_2})Ay_k=0$, we have

$$d_{k+1}-d_k\le\beta_k(d_k-d_{k-1})+2\beta_k(\|x_k-x_{k-1}\|^2+\|\omega_{k-1}\|^2)-\lambda_{k+1}^2\|\omega_k\|^2.\tag{3.8}$$

Otherwise, we deduce from (3.2) and (3.7) that

$$\begin{aligned}
d_{k+1}-d_k\le&\beta_k(d_k-d_{k-1})+2\beta_k(\|x_k-x_{k-1}\|^2+\|\omega_{k-1}\|^2)\\
&-\lambda_{k+1}^2\|\omega_k\|^2-\rho_k\big(\frac{1}{\alpha_2}-\rho_k\big)\frac{\|(I-T_{\alpha_2})Ay_k\|^4}{\|A^*(I-T_{\alpha_2})Ay_k\|^2}.
\end{aligned}\tag{3.9}$$

From the assumptions on $\{\rho_k\}$ and $\{\lambda_k\}$, (3.8)-(3.9), we get

$$d_{k+1}-d_k\le\beta_k(d_k-d_{k-1})+2\beta_k(\|x_k-x_{k-1}\|^2+\|\omega_{k-1}\|^2).$$

Setting

$$\phi_k:=d_k,\quad\theta_k:=\beta_k,\quad\delta_k:=2\beta_k(\|x_k-x_{k-1}\|^2+\|\omega_{k-1}\|^2),$$

and using Lemma 2.2, we obtain that $\lim_{k\to\infty}d_k$ exists. Thus, it follows that $\{d_k\}$ is bounded. Hence, $\{x_k\}$ is bounded. From (3.8) and (3.9), we also have

$$\lambda_{k+1}^2\|\omega_k\|^2\le d_k-d_{k+1}+\beta_k(d_k-d_{k-1})+2\beta_k(\|x_k-x_{k-1}\|^2+\|\omega_{k-1}\|^2),$$

which implies that

$$\lim_{k\to\infty}\|\omega_k\|=0.\tag{3.10}$$

Taking into account that $\liminf_{k\to\infty}\lambda_k>0$, $\{d_k\}$ converges and $\delta_k\to 0$ as $k\to\infty$. So, $\lim_{k\to\infty}\|x_k-z\|^2=\lim_{k\to\infty}(d_k-\lambda_k\|\omega_k\|^2)=\lim_{k\to\infty}d_k$ exists. We still denote $u_k=y_k-\gamma_kA^*(I-T_{\alpha_2})Ay_k$. Now, we show that

$$\lim_{k\to\infty}\|(I-T_{\alpha_2})Ay_k\|=\lim_{k\to\infty}\|y_k-u_k\|=0.$$

If $(I-T_{\alpha_2})Ay_k=0$, then

$$y_k-u_k=\gamma_k A^*(I-T_{\alpha_2})Ax_k=0. \tag{3.11}$$

Otherwise, it follows from (3.9) that

$$\begin{aligned}&\rho_k\Big(\frac{1}{\alpha_2}-\rho_k\Big)\frac{\|(I-T_{\alpha_2})Ay_k\|^4}{\|A^*(I-T_{\alpha_2})Ay_k\|^2}\\&\le d_k-d_{k+1}+\beta_k(d_k-d_{k-1})+2\beta_k(\|x_k-x_{k-1}\|^2+\|\omega_{k-1}\|^2),\end{aligned}$$

which implies

$$\lim_{k\to\infty}\frac{\|(I-T_{\alpha_2})Ay_k\|^2}{\|A^*(I-T_{\alpha_2})Ay_k\|}=0. \tag{3.12}$$

It follows from (A1) and

$$\begin{aligned}\frac{1}{\|A\|}\|(I-T_{\alpha_2})Ay_k\|&=\|(I-T_{\alpha_2})Ay_k\|\frac{\|(I-T_{\alpha_2})Ay_k\|}{\|A\|\|(I-T_{\alpha_2})Ay_k\|}\\&\le\|(I-T_{\alpha_2})Ay_k\|\frac{\|(I-T_{\alpha_2})Ay_k\|}{\|A^*(I-T_{\alpha_2})Ay_k\|}\\&=\frac{\|(I-T_{\alpha_2})Ay_k\|^2}{\|A^*(I-T_{\alpha_2})Ay_k\|}\end{aligned}$$

that $\lim_{k\to\infty}\|(I-T_{\alpha_2})Ay_k\|=0$. So, for the whole sequence $\{y_k\}$, we have

$$\lim_{k\to\infty}\|(I-T_{\alpha_2})Ay_k\|=0. \tag{3.13}$$

It follows from (3.12) that

$$\|y_k-u_k\|=\|\gamma_k A^*(I-T_{\alpha_2})Ay_k\|=\rho_k\frac{\|(I-T_{\alpha_2})Ay_k\|^2}{\|A^*(I-T_{\alpha_2})Ay_k\|}\to 0$$

as $k\to\infty$. From (3.11), we arrive at

$$\lim_{k\to\infty}\|y_k-u_k\|=0. \tag{3.14}$$

From Algorithm 3.1, we have $x_{k+1}=u_k-\lambda_{k+1}\omega_{k+1}$, which implies from (3.10) that

$$\lim_{k\to\infty}\|x_{k+1}-u_k\|=\lim_{k\to\infty}\lambda_{k+1}\|\omega_{k+1}\|=0. \tag{3.15}$$

On the other hand, it follows from (3.3) that

$$\lim_{k\to\infty}\beta_k(\|x_k-x_{k-1}\|^2+\|\omega_{k-1}\|^2)=0,$$

which implies that $\lim_{k\to\infty}\beta_k\|x_k-x_{k-1}\|^2=0$. Using the fact $\beta_k^2\|x_k-x_{k-1}\|^2\le\beta_k\|x_k-x_{k-1}\|^2$, we conclude that $\lim_{k\to\infty}\beta_k\|x_k-x_{k-1}\|=0$. Hence,

$$\lim_{k\to\infty}\|y_k-x_k\|=\lim_{k\to\infty}\beta_k\|x_k-x_{k-1}\|=0. \tag{3.16}$$

From (3.14), (3.15) and (3.16), we obtain

$$\lim_{k\to\infty}\|x_k-u_k\|=\lim_{k\to\infty}\|x_{k+1}-x_k\|=0. \tag{3.17}$$

In view of (3.10), we have

$$\lim_{k\to\infty}\|u_k+(1-\lambda_{k+1})\omega_k-U_{\alpha_1}(u_k+(1-\lambda_{k+1})\omega_k)\|=\lim_{k\to\infty}\|\omega_{k+1}\|=0. \tag{3.18}$$

Now, we show that $\omega_w(x_k) \subseteq \Gamma$. Since $I-U$ and $I-T$ are demiclosed at 0, we see from $I-U_{\alpha_1} = (1-\alpha_1)(I-U)$ and $I-T_{\alpha_2} = (1-\alpha_2)(I-T)$ that $I-U_{\alpha_1}$ and $I-T_{\alpha_2}$ are also demiclosed at 0. Let $\bar{x} \in \omega_w(x_k)$. That is, there exists a subsequence $\{x_{k_j}\}$ of $\{x_k\}$ such that $x_{k_j} \rightharpoonup \bar{x}$ as $j \to \infty$. Then, we have from (3.16) that $y_{k_j} \rightharpoonup \bar{x}$ and $Ay_{k_j} \rightharpoonup A\bar{x}$. From (3.10) and (3.17), we have $u_{k_j} + (1-\lambda_{k_j+1})\omega_{k_j} \rightharpoonup \bar{x}$ as $j \to \infty$. It follows from (3.13) and (3.18) that $\bar{x} \in F(U_{\alpha_1})$ and $A\bar{x} \in F(T_{\alpha_2})$, which implies that $\bar{x} \in \Gamma$. So $\omega_w(x_k) \subseteq \Gamma$.

Finally, by using Lemma 2.3 we have $x_k \rightharpoonup x^*$, where x^* is a solution of the SCFP (1.1). Thus, it follows from $\omega_k \to 0$ that $(\omega_k, x_k) \rightharpoonup (0, x^*)$. This completes the proof. □

Remark 3.3. Algorithm 3.1 includes the following algorithms for solving the SCFP (1.1) as special cases.

(i) If $\lambda_{k+1} \equiv \lambda \in (0,1]$, Algorithm 3.1 is reduced to the following self-adaptive inertial algorithm for solving the SCFP of quasi-nonexpansive operators

$$\begin{cases} y_k = x_k + \beta_k(x_k - x_{k-1}), \\ \omega_{k+1} = (I-U_{\alpha_1})(y_k - \gamma_k A^*(I-T_{\alpha_2})Ay_k + (1-\lambda)\omega_k), \\ x_{k+1} = y_k - \gamma_k A^*(I-T_{\alpha_2})Ay_k - \lambda\omega_{k+1}, \end{cases}$$

where $0 \le \beta_k \le \bar{\beta}_k$, $\bar{\beta}_k$ and γ_k are chosen by (3.1) and (3.2), respectively.

(ii) If $\lambda_{k+1} \equiv 1$, Algorithm 3.1 is reduced to the following self-adaptive inertial CQ algorithm for solving the SCFP of quasi-nonexpansive operators

$$\begin{cases} y_k = x_k + \beta_k(x_k - x_{k-1}), \\ x_{k+1} = U_{\alpha_1}(y_k - \gamma_k A^*(I-T_{\alpha_2})Ay_k), \end{cases} \tag{3.19}$$

where γ_k is chosen by (3.2) and $0 \le \beta_k \le \bar{\beta}_k$ satisfies

$$\bar{\beta}_k := \begin{cases} \min\{\eta, \frac{\varepsilon_k}{\|x_k - x_{k-1}\|^2}\}, & if\ x_k \neq x_{k-1}, \\ \eta, & otherwise \end{cases} \tag{3.20}$$

with $\eta \in [0,1)$, $\{\varepsilon_k\}_{k=1}^{\infty} \subset [0,\infty)$ and $\sum_{k=1}^{\infty} \varepsilon_k < \infty$.

(iii) If $\beta_k \equiv 0$, Algorithm 3.1 is reduced to the following self-adaptive primal-dual algorithm for solving the SCFP of quasi-nonexpansive operators

$$\begin{cases} \omega_{k+1} = (I-U_{\alpha_1})(x_k - \gamma_k A^*(I-T_{\alpha_2})Ax_k + (1-\lambda_{k+1})\omega_k), \\ x_{k+1} = x_k - \gamma_k A^*(I-T_{\alpha_2})Ax_k - \lambda_{k+1}\omega_{k+1}, \end{cases} \tag{3.21}$$

where $\{\gamma_k\}$ is chosen by

$$\gamma_k := \begin{cases} \frac{\rho_k\|(I-T_{\alpha_2})Ax_k\|^2}{\|A^*(I-T_{\alpha_2})Ax_k\|^2}, & (I-T_{\alpha_2})Ax_k \neq 0, \\ \gamma, & (I-T_{\alpha_2})Ax_k = 0 \end{cases} \tag{3.22}$$

with $0 < \liminf_{k\to\infty} \rho_k \le \limsup_{k\to\infty} \rho_k < \frac{1}{\alpha_2}$ and $\gamma > 0$.

(iv) If $\beta_k \equiv 0$ and $\lambda_{k+1} \equiv 1$, Algorithm 3.1 is reduced to relaxed algorithm (1.5), which was proposed by [13] with self-adaptive step sizes for solving the SCFP of quasi-nonexpansive operators

$$x_{k+1} = U_{\alpha_1}(x_k - \gamma_k A^*(I-T_{\alpha_2})Ax_k),$$

where $\{\gamma_k\}$ is chosen by (3.22).

(v) If $\alpha_1 = \alpha_2 = \frac{1}{2}$, Algorithm 3.1 is reduced to the following self-adaptive inertial algorithm for solving the SCFP of directed operators

$$\begin{cases} y_k = x_k + \beta_k(x_k - x_{k-1}), \\ \omega_{k+1} = (I-U)(y_k - \gamma_k A^*(I-T)Ay_k + (1-\lambda_{k+1})\omega_k), \\ x_{k+1} = y_k - \gamma_k A^*(I-T)Ay_k - \lambda_{k+1}\omega_{k+1}, \end{cases}$$

where $0 \le \beta_k \le \bar{\beta}_k$, $\bar{\beta}_k$ is chosen by (3.1) and $\{\gamma_k\}$ is chosen in such a way that

$$\gamma_k := \begin{cases} \frac{\rho_k \|(I-T)Ay_k\|^2}{\|A^*(I-T)Ay_k\|^2}, & (I-T)Ay_k \ne 0, \\ \gamma, & (I-T)Ay_k = 0 \end{cases} \tag{3.23}$$

with $0 < \liminf_{k\to\infty} \rho_k \le \limsup_{k\to\infty} \rho_k < 2$ and $\gamma > 0$.

(vi) If $\lambda_k \equiv \lambda \in (0,1]$, $\beta_k \equiv 0$, and $\alpha_1 = \alpha_2 = \frac{1}{2}$, Algorithm 3.1 is reduced to self-adaptive iterative algorithm (1.7) proposed by [24] for solving the SCFP of directed operators

$$\begin{cases} \omega_{k+1} = (I-U)(x_k - \gamma_k A^*(I-T)Ax_k + (1-\lambda)\omega_k), \\ x_{k+1} = x_k - \gamma_k A^*(I-T)Ax_k - \lambda\omega_{k+1}, \end{cases}$$

where γ_k is chosen in self-adaptive way by (3.23).

(vii) If $\lambda_k \equiv 1$, $\beta_k \equiv 0$, and $\alpha_1 = \alpha_2 = \frac{1}{2}$, Algorithm 3.1 is reduced to the well-known CQ iterative algorithm (1.4) proposed by [1] for solving the SCFP of directed operators

$$x_{k+1} = U(x_k - \gamma_k A^*(I-T)Ax_k),$$

where γ_k is chosen in self-adaptive way by (3.23).

It is well known that the projection operator P_C is firmly nonexpansive. So, $I - P_C$ is demi-closed at the origin by Lemma 2.4. Assume that the solution set of the SFP (1.2) is nonempty. From Theorem 3.1, we can deduce easily the following results.

Corollary 3.1. *Let $\{\lambda_k\} \subseteq (0,1]$ be a nonincreasing sequence with $\liminf_{k\to\infty} \lambda_k > 0$. Let $\{\varepsilon_k\}_{k=1}^\infty \subset [0,\infty)$ and $\{\rho_k\}_{k=1}^\infty \subseteq [0,2]$ be sequences satisfying $\sum_{k=1}^\infty \varepsilon_k < \infty$ and $0 < \liminf_{k\to\infty} \rho_k \le \limsup_{k\to\infty} \rho_k < 2$. Let $\{(\omega_k, x_k)\}$ be a sequence generated by*

$$\begin{cases} x_0, x_1, \omega_0 \in H_1 \text{ chosen arbitrarily}, \\ \eta \in [0,1), \omega_1 = \omega_0, \\ y_k = x_k + \beta_k(x_k - x_{k-1}), \\ \omega_{k+1} = (I-P_C)(y_k - \gamma_k A^*(I-P_Q)Ay_k + (1-\lambda_{k+1})\omega_k), \\ x_{k+1} = y_k - \gamma_k A^*(I-P_Q)Ay_k - \lambda_{k+1}\omega_{k+1}, \end{cases} \tag{3.24}$$

where $0 \le \beta_k \le \bar{\beta}_k$ with $\bar{\beta}_k$ being chosen by (3.1) and the stepsize $\{\gamma_k\}$ is chosen in such a way that

$$\gamma_k := \begin{cases} \frac{\rho_k \|(I-P_Q)Ay_k\|^2}{\|A^*(I-P_Q)Ay_k\|^2}, & (I-P_Q)Ay_k \ne 0, \\ \gamma, & (I-P_Q)Ay_k = 0 \end{cases} \tag{3.25}$$

with $\gamma > 0$. Then, $\{x_k\}$ converges weakly to a point x^*, where x^* is a solution of the SFP (1.2) and the sequence $\{(\omega_k, x_k)\}$ converges weakly to the point $(0, x^*)$.

Remark 3.4. (i) If $\lambda_{k+1} \equiv 1$, then algorithm (3.24) is reduced to the following inertial self-adaptive CQ algorithm for solving the SFP (1.2)

$$\begin{cases} y_k = x_k + \beta_k(x_k - x_{k-1}), \\ x_{k+1} = P_C(y_k - \gamma_k A^*(I - P_Q)Ay_k), \end{cases}$$

where $0 \leq \beta_k \leq \bar{\beta}_k$, $\bar{\beta}_k$ and γ_k are chosen by (3.20) and (3.25), respectively.

(ii) If $\beta_k \equiv 0$, then algorithm (3.24) is reduced to the following self-adaptive primal-dual algorithm for solving the SFP (1.2)

$$\begin{cases} \omega_{k+1} = (I - P_C)(x_k - \gamma_k A^*(I - P_Q)Ax_k + (1 - \lambda_{k+1}\omega_k)), \\ x_{k+1} = x_k - \gamma_k A^*(I - P_Q)Ax_k - \lambda_{k+1}\omega_{k+1}, \end{cases}$$

where γ_k is chosen by (3.25).

(iii) If $\beta_k \equiv 0$ and $\lambda_{k+1} \equiv \lambda \in (0,1]$, then algorithm (3.24) is reduced to the self-adaptive primal-dual algorithm (1.6) proposed in [18] for solving the SFP (1.2)

$$\begin{cases} \omega_{k+1} = (I - P_C)(x_k - \gamma_k A^*(I - P_Q)Ax_k + (1 - \lambda\omega_k)), \\ x_{k+1} = x_k - \gamma_k A^*(I - P_Q)Ax_k - \omega_{k+1}, \end{cases}$$

where γ_k is chosen by (3.25).

(iv) If $\beta_k \equiv 0$ and $\lambda_{k+1} \equiv 1$, then algorithm (3.24) is reduced to the well-known CQ iterative algorithm (1.3) proposed in [3] for solving the SFP (1.2)

$$x_{k+1} = P_C(x_k - \gamma_k A^*(I - P_Q)Ax_k),$$

where γ_k is chosen in self-adaptive way by (3.25).

4. Numerical experiment

In this section, we provide a numerical experiment and show the performance of our proposed self-adaptive inertial iterative Algorithm for solving the SCFP (1.1). All the codes are written in MATLAB and are performed on a personal Lenovo computer with Intel(R) Core(TM)i5-6200U CPU @2.30GHz 2.40GHz and RAM 4.00GB.

In this part, we take the following experiment parameters $\eta = 0.9$, $\gamma = 0.5$, $\rho_k = 1.95$ and $\varepsilon_k = \frac{1}{k^{1.5}}$ for $k \geq 1$ in Algorithm 3.1. In the implementation, we take $p(x) < \varepsilon = 10^{-4}$ as the stopping criterion, where

$$p(x) = \|x - Ux\| + \|Ax - T(Ax)\|.$$

In the numerical results listed in the following tables, "Iter." denote the number of iterations.

Example 4.1. Define mappings $U : \mathbb{R}^5 \to \mathbb{R}^5$ and $T : \mathbb{R}^6 \to \mathbb{R}^6$ as follows:

$$U(x) = \frac{1}{2}x, \quad T(y) = (z_1, z_2, z_3, z_4, z_5, z_6)^T,$$

where $x = (x_1, x_2, x_3, x_4, x_5)^T \in \mathbb{R}^5$, $y = (y_1, y_2, y_3, y_4, y_5, y_6)^T \in \mathbb{R}^6$ and

$$z_i = \begin{cases} y_i, & y_i < 0, \\ 0, & y_i \geq 0, \end{cases} \quad (1 \leq i \leq 6).$$

Note that U and T are quasi-nonexpansive operators. Obviously, $F(U) = \{(0,0,0,0,0)^T\}$ and

$$F(T) = \{(y_1, y_2, y_3, y_4, y_5, y_6)^T \in \mathbb{R}^6 : y_i \leq 0,\ 1 \leq i \leq 6\}.$$

TABLE 1. Numerical results with different β_k and $\lambda_k \equiv \lambda$, where $\beta_k = \tau\bar{\beta}_k$.

	λ	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
	τ	0.9	0.8	0.7	0.7	0.6	0.6	0.6	0.6	0.5	0.5
Algorithm 3.1	Iter.	306	149	86	77	56	50	45	42	37	29
algorithm (3.21)	Iter.	390	197	133	101	82	70	61	54	49	45 (Algorithm (1.5))

Let

$$A = \begin{pmatrix} 2 & -1 & 30 & 2 & 3 \\ 1 & -2 & 50 & 2 & 1 \\ 2 & 0 & 24 & 10 & -27 \\ 2 & -16 & 6 & -3 & 5 \\ 5 & 87 & -3 & 6 & 7 \\ 12 & -4 & 7 & -9 & 3 \end{pmatrix}.$$

We apply the proposed inertial Algorithm 3.1 to solve the SCFP (1.1). Let $\beta_k = \tau\bar{\beta}_k$ for $k \geq 1$. In Algorithm 3.1, inertial extrapolation factor $\beta_k \in [0, \bar{\beta}_k]$. We can choose different inertial extrapolation factors by adjusting parameter $\tau \in [0,1]$. If $\tau = 0$, i.e., $\beta_k \equiv 0$, then Algorithm 3.1 is reduced to the primal-dual algorithm (3.21) without inertial technique for solving the SCFP (1.1) of quasi-nonexpansive operators. We take $\alpha_1 = \alpha_2 = \frac{1}{2}$, $x_0 = (-30, 50, 8, 5, 0)^T$, $x_1 = (-5, 1, -80, 8, -1)^T$ and $\omega_0 = (-1, -1, -1, 2, 6)^T$. In Table 1, we present our experiments with different $\lambda_k \equiv \lambda \in (0,1]$ and different inertial extrapolation factors. We take $\lambda_k \equiv \lambda = 0.1, 0.2 \cdots, 1$. If $\lambda_k \equiv \lambda = 1$, then Algorithm (3.1) is reduced to the inertial algorithm (3.19) corresponding to the original algorithm (1.5) with self-adaptive step sizes. Moreover, if $\lambda_k \equiv \lambda = 1$ and $\beta_k \equiv 0$, then Algorithm 3.1 is reduced to the algorithm (1.5) with self-adaptive step sizes. Table 1 shows the iterative numbers of Algorithm 3.1, algorithm (3.21) without inertial technique and algorithm (1.5) with self-adaptive step sizes. Further, Figure 1 presents error value versus the iteration numbers. In Figure 1, we report the behavior of Algorithm 3.1 and algorithm (3.21) for $\lambda_k = 0.3$ and $\lambda_k = 0.4$, where $\beta_k = 0.7\bar{\beta}_k$ in Algorithm 3.1. It can be seen that Algorithm 3.1 is significantly faster than algorithm (3.21) and algorithm (1.5). So the inertial term constructed in Algorithm 3.1 improves the speed of convergence.

Remark 4.1. In the numerical experiment, it is revealed that the sequence, generated via the Algorithm 3.1 involving the inertial technique and dual variable, converges faster than the original algorithms when we choose suitable inertial extrapolation factors. This concludes that the inertial term constructed in Algorithm 3.1 improves the speed of convergence for solving the SCFP (1.1) of quasi-nonexpansive operators.

5. THE CONCLUSION

The main aim of this paper is to propose a new iterative algorithm for solving the SCFP (1.1) of quasi-nonexpansive operators. Two highlights are that the determination of the stepsizes does not need any prior information about the operator norms, and our iterative algorithm combines the primal-dual method and the inertial method. The weak convergence result of the proposed algorithm extends and generalizes some existing ones for the SCFP.

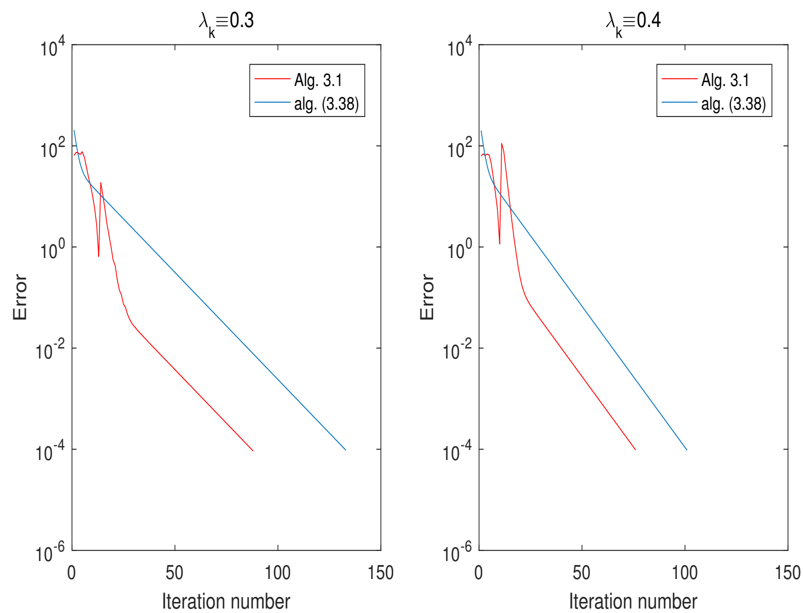

FIGURE 1. Comparison of the number of iterations of Algorithm 3.1 with algorithm(3.21).

Acknowledgments

This work was supported by the National Key Research and Development Program of China (No. 2020YFB1600101).

REFERENCES

[1] Y. Censor, A. Segal, The split common fixed point problem for directed operators, J. Convex Anal. 16 (2009), 587-600.

[2] C. Byrne, A unified treatment of some iterative algorithms in signal processing and image reconstruction, Inverse Probl. 20 (2004), 103-120.

[3] C. Byrne, Iterative oblique projection onto convex subsets and the split feasibility problem, Inverse Probl. 18 (2002), 441-453.

[4] D.R. Sahu, J.C. Yao, M. Verma, K.K. Shuka, Convergence rate analysis of proximal gradient methods with applications to composite minimization problems, Optimization, 70 (2021), 75-100.

[5] S.Y. Cho, B.A. Bin Dehaish, X. Qin, Weak convergence of a splitting algorithm in Hilbert spaces, J. Appl. Anal. Comput. 7 (2017), 427-438.

[6] T.H. Cuong, J.C. Yao, N.D. Yen, Qualitative properties of the minimum sum-of-squares clustering problem, Optimization, 69 (2020), 2131-2154.

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

[8] P.E. Maingé, Convergence theorems for inertial KM-type algorithms, J. Comput. Appl. Math. 219 (2008), 223-236.

[9] X. Qin, L. Wang, J.C. Yao, Inertial splitting method for maximal monotone mappings, J. Nonlinear Convex Anal. 21 (2020), 2325-2333.

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

[11] X. Qin, A. Petrusel, J.C. Yao, CQ iterative algorithms for fixed points of nonexpansive mappings and split feasibility problems in Hilbert spaces, J. Nonlinear Convex Anal. 19 (2018), 157-165.

[12] L. Liu, A hybrid steepest descent method for solving split feasibility problems involving nonexpansive mappings, J. Nonlinear Convex Anal. 20 (2019), 471-488.
[13] A. Moudafi, A note on the split common fixed point problem for quasi-nonexpansive operators, Nonlinear Anal. 74 (2011), 4083-4087.
[14] P. Chuasuk, A. Kaewcharoen, Generalized extragradient iterative methods for solving split feasibility and fixed point problems in Hilbert spaces, Rev. R. Acad. Cienc. Exactas Fis. Nat. Ser. A Math. RACSAM 114 (2020), 34.
[15] S.Y. Cho, A monotone Bregan projection algorithm for fixed point and equilibrium problems in a reflexive Banach space, Filomat, 34 (2020), 1487–1497.
[16] S. Reich, T.M. Tuyen, Two projection algorithms for solving the split common fixed point problem, J. Optim. Theory Appl. 186 (2020), 148-168.
[17] J. Zhao, S. He, Viscosity approximation methods for split common fixed-point problem of directed operators, Numer. Funct. Anal. Optim. 36 (2015), 528-547.
[18] J. Zhao, H. Zong, Iterative algorithms for solving the split feasibility problem in Hilbert spaces, J. Fixed Point Theory Appl. 20 (2018), 11.
[19] P. Chen, J. Huang, X. Zhang, A primal-dual fixed point algorithm for convex separable minimization with applications to image restoration, Inverse Probl. 29 (2013), 025011.
[20] F. Wang, A new method for split common fixed-point problem without priori knowledge of operator norms, J. Fixed Point Theory Appl. 19 (2017), 2427-2436.
[21] Q. Yang, On variable-step relaxed projection algorithm for variational inequalities, J. Math. Anal. Appl. 302 (2005), 166-179.
[22] H. Zhou, P. Wang, Adaptively relaxed algorithms for solving the split feasibility problem with a new step size, J. Inequal. Appl. 2014 (2014), 448.
[23] G. Lopez, V. Martin, F. Wang, H.K. Xu, Solving the split feasibility problem without prior knowledge of matrix norms, Inverse Probl. 28 (2012), 085004.
[24] J. Zhao, D. Hou, A self-adaptive iterative algorithm for the split common fixed point problems, Numer. Algor. 82 (2019), 1047-1063.
[25] B.T. Polyak, Some methods of speeding up the convergence of iteration methods, U.S.S.R. Comput. Math. Math. Phys. 4 (1964), 1-17.
[26] Y. Nesterov, A method for solving the convex programming problem with convergence rate O(1/k2), Dokl. Akad. Nauk SSSR 269 (1983), 543-547.
[27] D.V. Thong, D.V. Hieu, An inertial method for solving split common fixed point problems, J. Fixed Point Theory Appl. 19 (2017), 3029-3051.
[28] F. Alvarez, H. Attouch, An inertial proximal method for maximal monotone operators via discretization of a nonlinear oscillator with damping, Set Valued Anal. 9 (2001), 3-11.
[29] Y. Shehu, J.C. Yao, Rate of convergence for inertial iterative method for countable family of certain quasi-nonexpansive mappings, J. Nonlinear Convex Anal. 21 (2020), 533-541.
[30] A. Gibali, D.T. Mai, N.T. Vinh, A new relaxed CQ algorithm for solving Split Feasibility Problems in Hilbert spaces and its applications, J. Indus. Manage. Optim. 15 (2019), 963-984.
[31] S. Suantai, N. Pholasa, P. Cholamjiak, Relaxed CQ algorithms involving the inertial technique for multiple-sets split feasibility problems, Rev. R. Acad. Cienc. Exactas Fis. Nat. Ser. A Math. 113 (2019), 1081-1099.
[32] A. Hanjing, S. Suantai, Hybrid inertial accelerated algorithms for split fixed point problems of demicontractive mappings and equilibrium problems, Numer. Algor. 85 (2020), 1051-1073.
[33] Y. Shehu, A. Gibali, New inertial relaxed method for solving split feasibilities, Optim. Lett. (2020), https://doi.org/10.1007/s11590-020-01603-1.
[34] R. Suparatulatorn, P. Cholamjiak, S. Suantai, Self-adaptive algorithms with inertial effects for solving the split problem of the demicontractive operators, Rev. R. Acad. Cienc. Exactas Fis. Nat. Ser. A Math. RACSAM 114 (2020), 40.
[35] J. Gornicki, Weak convergence theorems for asymptotically nonexpansive mappings in uniformly convex Banach spaces, Comment. Math. Univ. Carolin. 301 (1998), 249-252.