J. Nonlinear Funct. Anal. 2023 (2023) 20 https://doi.org/10.23952/jnfa.2023.20

Journal of Nonlinear Functional Analysis
Available online at http://jnfa.mathres.org

TWO-STEP INERTIAL ADAPTIVE ITERATIVE ALGORITHM FOR SOLVING THE SPLIT COMMON FIXED POINT PROBLEM OF DIRECTED OPERATORS

SHUYU CHEN, HENG DU, JIAJUN HUANG, ZHENTAO CHEN, JING ZHAO*

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

Abstract. In this paper, we use the dual variable to propose a two-step inertial adaptive iterative algorithm for solving the split common fixed point problem of directed operators in real Hilbert spaces. Under suitable conditions, we obtain the weak convergence of the proposed algorithm and give applications in the split feasibility problem. A numerical experiment is given to illustrate the efficiency of the proposed iterative algorithm.

Keywords. Adaptive iterative algorithm; Inertial acceleration; Numerical experiment; Split feasibility problem; Weak convergence.

1. INTRODUCTION

Over the past two decades, the split common fixed-point problem (SCFP) and the split feasibility problem (SFP) received more and more attention, especially in medical image reconstruction and signal proceeding [2].

Let H_1 and H_2 be two real Hilbert spaces. Let $U : H_1 \to H_1$ and $T : H_2 \to H_2$ be nonlinear mappings, and let $A : H_1 \to H_2$ be a bounded linear operator. In 2009, the SCFP was introduced by Censor and Segal [6], and it can be formulated as the problem of finding

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

where $F(U)$ 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 projection operators, the SCFP is transformed into the SFP [3, 5, 19, 20], which can be formulated as the problem of finding

$$x \in C \;\text{ such that }\; Ax \in Q, \tag{1.2}$$

where C and Q are nonempty closed convex subsets of H_1 and H_2, respectively.

Many algorithms have been introduced to solve the SFP recently. In 2002, Byrne [3] first proposed the following celebrated CQ algorithm for numerically solving the SFP which generates a sequence $\{x_k\}$ by

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

*Corresponding author.
E-mail address: zhaojing200103@163.com (J. Zhao)
Received November 30, 2022; Accepted April 17, 2023.

where $\gamma_k \in (0, \frac{2}{\lambda})$ and λ is the spectral radius of the operator A^*A. A number of scholars focused on extending the CQ algorithm which avoids computing the inverse of the matrix.

In CQ algorithm and its variants, we notice that the stepsize γ_k depends on the largest eigenvalue of matrix or bounded linear operator norm. In order to avoid this difficulty, Lopez et al. [14] proposed and investigated the following stepsize selection method:

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

where $\inf_k \rho_k(4-\rho_k) > 0$ and $f(x) := \frac{1}{2}\|(I-P_Q)Ax\|^2$.

It is known that the SFP can be turned into an optimization problem. Indeed it is also clear. We assume that the SFP (1.2) is consistent, and then the SFP is equivalent to the following minimization: $\min_{x\in C} f(x)$, where $f(x) := \frac{1}{2}\|(I-P_Q)Ax\|^2$. Furthermore, the SFP can be also turned into a separable convex optimization problem:

$$\min_{x\in H_1} \iota_C(x) + f(x),$$

where $\iota_C(x)$ is an indicator function of the set C defined by

$$\iota_C(x) = \begin{cases} 0, & x \in C, \\ +\infty, & x \notin C. \end{cases}$$

In 2013, Chen et al. [7] considered minimizing the sum of two proper lower semi-continuous convex functions, as demonstated below:

$$x^* = \arg\min_{x\in R^n} f_1(x) + f_2(x), \tag{1.4}$$

where f_1 and f_2 are proper lower semi-continuous convex functions from R^n to $(-\infty, +\infty]$, and f_2 is differentiable on R^n with $1/\beta$-Lipschitz continuous gradient for some $\beta \in (0, +\infty)$. They introduced the following iterative sequence to solve the convex separable problem (1.7):

$$\begin{cases} v_{k+1} = (I - prox_{\frac{\gamma}{\lambda} f_1})(x_k - \gamma\nabla f_2(x_k) + (1-\lambda)v_k), \\ x_{k+1} = x_k - \gamma\nabla f_2(x_k) - \lambda v_{k+1}, \end{cases} \tag{1.5}$$

where λ and γ are two positive numbers. Under appropriate conditions [7], the sequence $\{x_k\}$ converges to a solution of the problem (1.4). They obtained that x is the primal variable and v is actually the dual variable of the primal-dual form related to (1.4). Let $f_1(x) = \iota_C(x)$ and $f_2(x) = f(x)$, then the algorithm (1.5) becomes the following primal-dual method for solving the SFP (1.2):

$$\begin{cases} v_{k+1} = (I - P_C)(x_k - \gamma A^*(I-P_Q)Ax_k + (1-\lambda)v_k), \\ x_{k+1} = x_k - \gamma A^*(I-P_Q)Ax_k - \lambda v_{k+1}. \end{cases} \tag{1.6}$$

Censor and Segal [6] proposed the following iterative algorithm for solving the SCFP (1.1) of directed operators:

$$x_{k+1} = U(x_k - \gamma A^*(I-T)Ax_k). \tag{1.7}$$

We note that, when projection operators P_C and P_Q are replaced by directed operators, CQ-algorithm (1.3) becomes iterative scheme (1.7).

In [25], the following self-adaptive iterative algorithm with the dual variable was proposed based on the idea in [7] for solving the SCFP (1.1) of directed operators:

$$\begin{cases} v_{k+1} = (I-U)(x_k - \gamma_k A^*(I-T)Ax_k + (1-\lambda)v_k), \\ x_{k+1} = x_k - \gamma_k A^*(I-T)Ax_k - \lambda v_{k+1}, \end{cases} \tag{1.8}$$

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 $\gamma > 0$. Observe that algorithm (1.8) generalizes algorithm ((1.7) since algorithm (1.8) becomes algorithm (1.7) as $\lambda = 1$.

Some authors introduced some algorithms to solve the SCFP (1.1); see, e.g., [4, 11, 12, 16, 23, 24]. In optimization theory, the inertial technique is used to speed up the convergence rate. Polyak [18] firstly proposed the heavy ball method and Nesterov [17] introduced a modified heavy ball method for minimizing a smooth convex function. It is remarkable that inertial term makes use of the previous two iterates such that the performance of the algorithm is improved greatly. In [1], by employing the idea of the heavy ball method to the setting of a general maximal monotone operator, Alvarez and Attouch proposed inertial proximal point algorithm:

$$\begin{cases} y_k = x_k + \alpha_k(x_k - x_{k-1}), \\ x_{k+1} = (I + \lambda_k F)^{-1}(y_k), \end{cases} \tag{1.9}$$

where F is a maximal monotone operator. They proved that, if $\{\alpha_k\} \subseteq [0,1)$ satisfies

$$\sum_{k=1}^{\infty} \alpha_k \|x_k - x_{k-1}\|^2 < \infty$$

and $\{\lambda_k\}$ is non-decreasing, then $\{x_k\}$ generated by (1.9) converges weakly to a zero point of F.

In [15], Maing*é* introduced the following inertial Mann iterative algorithm to find the fixed point of nonexpansive mapping T:

$$\begin{cases} y_k = x_k + \alpha_k(x_k - x_{k-1}), \\ x_{k+1} = (1-\beta_k)y_k + \beta_k T(y_k). \end{cases} \tag{1.10}$$

Recently, for solving the SFP in Hilbert spaces, Dang et al. [8] proposed the inertial relaxed CQ algorithm:

$$\begin{cases} y_k = x_k + \alpha_k(x_k - x_{k-1}), \\ x_{k+1} = P_{C_k}(y_k - \lambda_k A^*(I - P_{Q_k})Ay_k). \end{cases}$$

They proved the weak convergence theorem for Picard-type and Mann-type iteration processes, where the stepsize $\gamma \in (0, 2/L)$ and L denotes the spectral radius of A^*A. In order to overcome the difficulty of computing the bounded linear operator norm, Gibali et al. [13] introduced self-adaptive inertial relaxed CQ algorithm for solving the SFP in real Hilbert spaces. Shehu et al. [21] proposed self-adaptive projection method with an inertial technique and proved strong convergence for split feasibility problems in Banach spaces.

In [26], a self-adaptive inertial iterative algorithm with one-step inertial technique was proposed and demonstrated below:

$$\begin{cases} y_k = x_k + \alpha_k (x_k - x_{k-1}), \\ \omega_{k+1} = (I-U)(y_k - \gamma_k A^*(I-T)Ay_k + (1-\lambda)\omega_k), \\ x_{k+1} = y_k - \gamma_k A^*(I-T)Ay_k - \lambda \omega_{k+1}, \end{cases} \tag{1.11}$$

where $0 \leq \alpha_k \leq \bar{\alpha}_k$, and

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

Besides, the stepsize γ_k is chosen in such a way that

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

The numerical experiments demonstrate that this algorithm has faster convergence speed and shorter iteration time, and it can effectively solve the SCFP.

In [10], Dong et al. proposed the following multi-step inertial Krasnosel'skiǐ–Mann algorithm(MiKM) for solving the fixed point problem of nonexpansive operator T:

$$\begin{cases} y_k = x_k + \sum_{n \in \mathscr{S}_k} a_{k,n} (x_{k-n} - x_{k-n-1}), \\ z_k = x_k + \sum_{n \in \mathscr{S}_k} b_{k,n} (x_{k-n} - x_{k-n-1}), \\ x_{k+1} = (1-\lambda_k) y_k + \lambda_k T(z_k), \end{cases}$$

where $a_{k,n}, b_{k,n} \in (-1,2]^{|\mathscr{S}_k|}$ for each $k \geq 2$ and $|\mathscr{S}_k|$ denotes the number of elements of the set $\mathscr{S}_k$.

Inspired and motivated by the above research works, for solving the SCFP (1.1) of directed operators, we construct two-step self-adaptive iterative algorithm by using inertial extrapolation and primal-dual algorithm. The contents of this paper are as follows. First, we give some useful definitions and results for the convergence analysis of the iterative algorithm. Second, we prove a weak convergence theorem of the proposed algorithm. Finally, we provide a numerical experiment for solving the SFP (1.2) to illustrate the convergence behavior and the effectiveness of the proposed algorithm.

2. Preliminaries

In this paper, we denote the inner product by $\langle \cdot, \cdot \rangle$ and the norm by $\|\cdot\|$, we use $\rightharpoonup$ and $\to$ to denote the weak convergence and strong convergence, respectively. We use $\omega_w(x_k) = \{x : \exists x_{k_j} \rightharpoonup x\ as\ j \to \infty\}$ to stand for the weak limit set of $\{x_k\}$, and let H be a real Hilbert space.

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

(i) nonexpansive if

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

for all $x, y \in H$;

(ii) firmly nonexpansive if $2T - I$ is nonexpansive or equivalent to

$$\|Tx - Ty\|^2 \leq \|x - y\|^2 - \|(x-y) - (Tx - Ty)\|^2$$

for all $x, y \in H$;

(iii) firmly quasi-nonexpansive (also called directed operator) if $F(T) \neq \varnothing$ and

$$\|Tx-q\|^2 \leqslant \|x-q\|^2 - \|x-Tx\|^2$$

or equivalent to

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

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

(iv) demiclosed at the origin if, for any sequence $\{x_k\}$ which weakly converges to x, the sequence $\{Tx_k\}$ strongly converges to 0, then $Tx = 0$.

Lemma 2.2. [22] *Let $T : H \to H$ be an operator. Then the following statements are equivalent:*
(i) T is directed;
(ii) there holds the relation:

$$\|x-Tx\|^2 \leq \langle x-q, x-Tx\rangle, \;\; q \in F(T), \;\; x \in H.$$

Lemma 2.3. [10] *For any $a, b \in H$, the following holds:*

$$\|a-b\|^2 \leq (1+\|b\|)\|a\|^2 + \|b\| + \|b\|^2.$$

Lemma 2.4. [9] *Assume that $\{a_k\}$ is a sequence of nonnegative real numbers such that*

$$a_{k+1} \leqslant (1+\gamma_k)a_k + \delta_k,$$

where the sequences $\{\gamma_k\}$ is in $[0,+\infty)$, in addition, both $\{\gamma_k\}$ and $\{\delta_k\}$ satisfy the following conditions:
(i) $\sum_{k=0}^{\infty} \gamma_k < +\infty$;
(ii) $\sum_{k=0}^{\infty} \delta_k < +\infty$ or $\sup \delta_k \leqslant 0$.

Then $\lim_{k\to+\infty} a_k$ exists.

Lemma 2.5. *Let K be a nonempty closed convex subset of real Hilbert space. Let $\{x_k\}$ be a bounded sequence which satisfies the following properties:*
(i) every weak limit point of $\{x_k\}$ lies in K;
(ii) $\lim_{k\to\infty} \|x_k - x\|$ exists for every $x \in K$.

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

3. Weak Convergence of Two-Step Inertial Adaptive Iterative Algorithm

In this paper, we make use of the following assumptions:
(A1) $U : H_1 \to H_1$ and $T : H_2 \to H_2$ are directed operators, and $A : H_1 \to H_2$ 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. (Two-step inertial adaptive iterative algorithm)
Choose two sequences $\{\rho_k\}_{k=1}^{\infty} \subset [0,\infty)$ and $\{\varepsilon_k\}_{k=1}^{\infty} \subset [0,\infty)$ satisfying

$$0 < \rho_k < 2$$

and

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

Select arbitrary starting points x_0, x_1, $v_1 \in H_1$, $\lambda \in [0,1]$, $\gamma > 0$.

Iterative step: For $k \geq 1$, given the iterates x_{k-2}, x_{k-1}, x_k and v_k, choose α_k, β_k such that $0 \leqslant \max\{\alpha_k, \beta_k\} \leqslant \overline{\alpha}_k$, where

$$\overline{\alpha}_k = \begin{cases} \frac{\varepsilon_k}{\|x_k - x_{k-1}\| + \|x_{k-1} - x_{k-2}\|}, & x_k \neq x_{k-1} \text{ } or \text{ } x_{k-1} \neq x_{k-2}, \\ \theta, & otherwise. \end{cases} \tag{3.1}$$

Compute

$$\begin{cases} y_k = x_k + \alpha_k (x_k - x_{k-1}) + \beta_k (x_{k-1} - x_{k-2}), \\ v_{k+1} = (I - U)(y_k - \gamma_k A^*(I-T)Ay_k + (1-\lambda) v_k), \\ x_{k+1} = y_k - \gamma_k A^*(I-T)Ay_k - \lambda v_{k+1}, \end{cases}$$

where the stepsize γ_k is chosen in such a way that

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

Remark 3.1. By (3.1), we have that

$$\max\{\alpha_k, \beta_k\} (\|x_k - x_{k-1}\| + \|x_{k-1} - x_{k-2}\|) \leq \varepsilon_k,$$

and then

$$\sum_{k=1}^{\infty} \max\{\alpha_k, \beta_k\} (\|x_k - x_{k-1}\| + \|x_{k-1} - x_{k-2}\|) < +\infty.$$

There are many choices for sequence $\{\varepsilon_k\}$. For example, we take $\varepsilon_k = \frac{1}{k^2}$, i.e.,

$$\max\{\alpha_k, \beta_k\} \leq \frac{1}{k^2 (\|x_k - x_{k-1}\| + \|x_{k-1} - x_{k-2}\|)}.$$

Then

$$\sum_{k=1}^{\infty} \max\{\alpha_k, \beta_k\} (\|x_k - x_{k-1}\| + \|x_{k-1} - x_{k-2}\|) < +\infty.$$

Lemma 3.2. *The stepsize γ_k defined by (3.2) is well-defined.*

Proof. Taking $x \in \Gamma$, i.e., $x \in F(U)$ and $Ax \in F(T)$, by Lemma 2.2 (ii) we have

$$\begin{aligned} \|A^*(I-T)Ay_k\| \cdot \|y_k - x\| &\geq \langle A^*(I-T)Ay_k, y_k - x \rangle \\ &= \langle (I-T)Ay_k, Ay_k - Ax \rangle \\ &\geq \|(I-T)Ay_k\|^2. \end{aligned}$$

Consequently, we have $\|A^*(I-T)Ay_k\| > 0$ when $\|(I-T)Ay_k\| \neq 0$. □

Lemma 3.3. *Let $\{(v_k, x_k)\}$ be the sequence generated by Algorithm 3.1. Then, for any $z \in \Gamma$, the following inequality holds:*

$$\begin{aligned} \|x_{k+1} - z\|^2 + \lambda \|v_{k+1}\|^2 \leq \|y_k - z\|^2 + \lambda \|v_k\|^2 - \lambda^2 \|v_k\|^2 - \lambda(1-\lambda) \|v_{k+1} - v_k\|^2 \\ - \gamma_k \left(2\|(I-T)Ay_k\|^2 - \gamma_k \|A^*(I-T)Ay_k\|^2\right). \end{aligned}$$

Proof. Denoting $u_k = y_k - \gamma_k A^*(I-T)Ay_k$, we have $v_{k+1} = (I-U)(u_k + (1-\lambda)v_k)$ and $x_{k+1} = u_k - \lambda v_{k+1}$. Taking $z \in \Gamma$, we have $z \in F(U)$ and $Az \in F(T)$. It following from Algorithm 3.1 and Lemma 2.2 (ii) that

$$\|v_{k+1}\|^2 = \|(I-U)(u_k + (1-\lambda)v_k)\|^2 \leq \langle v_{k+1}, u_k - z + (1-\lambda)v_k \rangle$$

and

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

Thus we have

$$\begin{aligned}
&\|x_{k+1}-z\|^2 + \lambda\|v_{k+1}\|^2 \\
=&\|u_k-z\|^2 - 2\lambda\langle u_k-z, v_{k+1}\rangle + \lambda^2\|v_{k+1}\|^2 + \lambda\|v_{k+1}\|^2 \\
=&\|u_k-z\|^2 - 2\lambda\langle u_k-z, v_{k+1}\rangle + 2\lambda\|v_{k+1}\|^2 - \lambda(1-\lambda)\|v_{k+1}\|^2 \\
\le&\|u_k-z\|^2 - 2\lambda\langle u_k-z, v_{k+1}\rangle + 2\lambda\langle u_k-z+(1-\lambda)v_k, v_{k+1}\rangle - \lambda(1-\lambda)\|v_{k+1}\|^2 \\
=&\|u_k-z\|^2 + 2\lambda(1-\lambda)\langle v_k, v_{k+1}\rangle - \lambda(1-\lambda)\|v_{k+1}\|^2.
\end{aligned}$$

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

$$\begin{aligned}
&\|x_{k+1}-z\|^2 + \lambda\|v_{k+1}\|^2 \\
\le&\|u_k-z\|^2 + \lambda(1-\lambda)\|v_k\|^2 - \lambda(1-\lambda)\|v_{k+1}-v_k\|^2 \\
=&\|u_k-z\|^2 + \lambda\|v_k\|^2 - \lambda^2\|v_k\|^2 - \lambda(1-\lambda)\|v_{k+1}-v_k\|^2.
\end{aligned}$$

It follows from $Az \in F(T)$ and Lemma 2.2 (ii) that

$$\langle y_k-z, A^*(I-T)Ay_k\rangle = \langle Ay_k-Az, (I-T)Ay_k\rangle \ge \|(I-T)Ay_k\|^2,$$

which implies that

$$\begin{aligned}
&\|u_k-z\|^2 \\
=&\|y_k-\gamma_k A^*(I-T)Ay_k-z\|^2 \\
=&\|y_k-z\|^2 - 2\gamma_k\langle y_k-z, A^*(I-T)Ay_k\rangle + \gamma_k^2\|A^*(I-T)Ay_k\|^2 \\
\le&\|y_k-z\|^2 - 2\gamma_k\|(I-T)Ay_k\|^2 + \gamma_k^2\|A^*(I-T)Ay_k\|^2 \\
=&\|y_k-z\|^2 - \gamma_k\left(2\|(I-T)Ay_k\|^2 - \gamma_k\|A^*(I-T)Ay_k\|^2\right).
\end{aligned}$$

Hence, we have

$$\begin{aligned}
&\|x_{k+1}-z\|^2 + \lambda\|v_{k+1}\|^2 \\
\le&\|y_k-z\|^2 + \lambda\|v_k\|^2 - \lambda^2\|v_k\|^2 - \lambda(1-\lambda)\|v_{k+1}-v_k\|^2 \\
&-\gamma_k\left(2\|(I-T)Ay_k\|^2 - \gamma_k\|A^*(I-T)Ay_k\|^2\right).
\end{aligned}$$

□

Theorem 3.4. *Suppose $I-U$ and $I-T$ are demiclosed at the origin, $0<\lambda\le 1$, and*

$$0 < \liminf_{k\to\infty}\rho_k \le \limsup_{k\to\infty}\rho_k < 2.$$

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

Proof. Taking $z \in \Gamma$, from Lemma 3.3, we obtain

$$\begin{aligned} & \|x_{k+1}-z\|^2+\lambda \|v_{k+1}\|^2 \\ \leq & \|y_k-z\|^2+\lambda \|v_k\|^2-\lambda^2 \|v_k\|^2-\lambda(1-\lambda)\|v_{k+1}-v_k\|^2 \\ & -\gamma_k\left(2\|(I-T)Ay_k\|^2-\gamma_k\|A^*(I-T)Ay_k\|^2\right) \\ \leq & \|y_k-z\|^2+\lambda \|v_k\|^2-\lambda^2 \|v_k\|^2-\gamma_k\left(2\|(I-T)Ay_k\|^2-\gamma_k\|A^*(I-T)Ay_k\|^2\right). \end{aligned} \tag{3.3}$$

By Lemma 2.3, we have

$$\begin{aligned} \|y_k-z\|^2 = & \|x_k+\alpha_k(x_k-x_{k-1})+\beta_k(x_{k-1}-x_{k-2})-z\|^2 \\ = & \|x_k-z-(\alpha_k(x_{k-1}-x_k)+\beta_k(x_{k-2}-x_{k-1}))\|^2 \\ \leq & \|x_k-z\|^2(1+\|\alpha_k(x_{k-1}-x_k)+\beta_k(x_{k-2}-x_{k-1})\|) \\ & +\|\alpha_k(x_{k-1}-x_k)+\beta_k(x_{k-2}-x_{k-1})\|+\|\alpha_k(x_{k-1}-x_k)+\beta_k(x_{k-2}-x_{k-1})\|^2. \end{aligned} \tag{3.4}$$

Set $b_k=\alpha_k(x_{k-1}-x_k)+\beta_k(x_{k-2}-x_{k-1})$. It follows from (3.3) and (3.4) that

$$\begin{aligned} & \|x_{k+1}-z\|^2+\lambda\|v_{k+1}\|^2 \\ \leq & \|x_k-z\|^2(1+\|b_k\|)+\|b_k\|+\|b_k\|^2+\lambda\|v_k\|^2 \\ & -\lambda^2\|v_k\|^2-\gamma_k\left(2\|(I-T)Ay_k\|^2-\gamma_k\|A^*(I-T)Ay_k\|^2\right) \\ \leq & \left(\|x_k-z\|^2+\lambda\|v_k\|^2\right)(1+\|b_k\|)+\|b_k\|+\|b_k\|^2 \\ & -\lambda^2\|v_k\|^2-\gamma_k\left(2\|(I-T)Ay_k\|^2-\gamma_k\|A^*(I-T)Ay_k\|^2\right). \end{aligned}$$

Let $d_k=\|x_k-z\|^2+\lambda\|v_k\|^2$. Then it follows that

$$\begin{aligned} d_{k+1} \leq & d_k(1+\|b_k\|)+\|b_k\|+\|b_k\|^2 \\ & -\lambda^2\|v_k\|^2-\gamma_k\left(2\|(I-T)Ay_k\|^2-\gamma_k\|A^*(I-T)Ay_k\|^2\right). \end{aligned} \tag{3.5}$$

For the case $(I-T)Ay_k=0$, we have

$$d_{k+1} \leq d_k(1+\|b_k\|)+\|b_k\|+\|b_k\|^2-\lambda^2\|v_k\|^2. \tag{3.6}$$

Otherwise, we deduce from (3.2) and (3.5) that

$$\begin{aligned} d_{k+1} \leq & d_k(1+\|b_k\|)+\|b_k\|+\|b_k\|^2 \\ & -\lambda^2\|v_k\|^2-\rho_k(2-\rho_k)\frac{\|(I-T)Ay_k\|^4}{\|A^*(I-T)Ay_k\|^2}. \end{aligned} \tag{3.7}$$

From the assumptions on ρ_k, λ, (3.6) and (3.7), we see that $d_{k+1} \leq d_k(1+\|b_k\|)+\|b_k\|+\|b_k\|^2$. Apply Lemma 2.4 and assume that

$$a_{k+1}:=d_{k+1}, \gamma_k:=\|b_k\|, \; and \; \delta_k:=\|b_k\|+\|b_k\|^2.$$

It follows from (3.1) that

$$\begin{aligned}\|b_k\| &= \|\alpha_k(x_{k-1}-x_k)+\beta_k(x_{k-2}-x_{k-1})\| \\ &\leq \max\{\alpha_k,\beta_k\}(\|x_k-x_{k-1}\|+\|x_{k-1}-x_{k-2}\|) \\ &< \varepsilon_k.\end{aligned}$$

That means $\sum_{k=1}^{\infty}||b_k||<+\infty$, $\sum_{k=0}^{\infty}\gamma_k<+\infty$, $\sum_{k=0}^{\infty}\delta_k<+\infty$, and then we obtain that $\lim_{k\to\infty}d_k$ exists. Thus it follows that $\{d_k\}$ is bounded and hence $\{x_k\}$ is bounded. From (3.6) and (3.7), we also have

$$\begin{aligned}\lambda^2\|v_k\|^2 &\leq -d_{k+1}+d_k(1+\|b_k\|)+\|b_k\|+\|b_k\|^2 \\ &= d_k-d_{k+1}+d_k\|b_k\|+\|b_k\|+\|b_k\|^2,\end{aligned}$$

which implies that

$$\lim_{k\to\infty}\|v_k\|=0 \tag{3.8}$$

by taking into account that $\lambda>0$, and $\lim_{k\to\infty}\delta_k=0$. Then $\lim_{k\to\infty}\|x_k-z\|^2=\lim_{k\to\infty}(d_k-\lambda\|v_k\|^2)=\lim_{k\to\infty}d_k$ exists. We still denote $u_k=y_k-\gamma_kA^*(I-T)Ay_k$.

Now, we prove that

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

If $(I-T)Ay_k=0$, it is clear that

$$y_k-u_k=\gamma_kA^*(I-T)Ay_k=0. \tag{3.9}$$

Otherwise, it follows from (3.7) that

$$\rho_k(2-\rho_k)\frac{\|(I-T)Ay_k\|^4}{\|A^*(I-T)Ay_k\|^2}\leq d_k-d_{k+1}+d_k\|b_k\|+\|b_k\|+\|b_k\|^2.$$

It is obvious from assumption $0<\liminf_{k\to\infty}\rho_k\leq\limsup_{k\to\infty}\rho_k<2$ that

$$\lim_{k\to\infty}\frac{\|(I-T)Ay_k\|^4}{\|A^*(I-T)Ay_k\|^2}=0.$$

Thus

$$\lim_{k\to\infty}\frac{\|(I-T)Ay_k\|^2}{\|A^*(I-T)Ay_k\|}=0. \tag{3.10}$$

It follows from $A\neq 0$, (3.10) and

$$\frac{\|(I-T)Ay_k\|^2}{\|A^*(I-T)Ay_k\|}\geq\frac{\|(I-T)Ay_k\|^2}{\|A\|\,\|(I-T)Ay_k\|}=\frac{1}{\|A\|}\|(I-T)Ay_k\|,$$

that $\lim_{k\to\infty}\|(I-T)Ay_k\|=0$. And from (3.10), we have

$$\|y_k-u_k\|=\|\gamma_kA^*(I-T)Ay_k\|=\rho_k\frac{\|(I-T)Ay_k\|^2}{\|A^*(I-T)Ay_k\|}\to 0 \tag{3.11}$$

as $k\to\infty$. Combining (3.9) and (3.11), for the whole sequence $\{y_k\}$, we obtain

$$\lim_{k\to\infty}\|(I-T)Ay_k\|=\lim_{k\to\infty}\|y_k-u_k\|=0. \tag{3.12}$$

From Algorithm 3.1, we have $x_{k+1}=u_k-\lambda v_{k+1}$. It follows from (3.8) that

$$\lim_{k\to\infty}\|x_{k+1}-u_k\|=\lim_{k\to\infty}\lambda\|v_{k+1}\|=0. \tag{3.13}$$

On the other hand, it follows from (3.1) that

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

which further yields that

$$\lim_{k\to\infty}\|y_k-x_k\|=\lim_{k\to\infty}\left(\alpha_k\left(\|x_k-x_{k-1}\|\right)+\beta_k\left(\|x_{k-1}-x_{k-2}\|\right)\right)=0. \tag{3.14}$$

From (3.12), (3.13), and (3.14), we have

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

By Algorithm 3.1 and (3.8), we have

$$\lim_{k\to\infty}\|u_k+(1-\lambda)v_k-U\left(u_k+(1-\lambda)\,v_k\right)\|=\lim_{k\to\infty}\|v_{k+1}\|=0. \tag{3.16}$$

Now, we show that $\omega_w(x_k)\subseteq\Gamma$. Let $\bar{x}\in\omega_w(x_k)$, i.e., there exist a subsequence $\{x_{k_j}\}$ of $\{x_k\}$ such that $x_{k_j}\rightharpoonup\bar{x}$ as $j\to\infty$. Then, from (3.14), we have $Ay_{k_j}\rightharpoonup A\bar{x}$ as $j\to\infty$. By (3.8) and (3.15), we have $u_{k_j}+(1-\lambda)v_{k_j}\rightharpoonup\bar{x}$ as $j\to\infty$. It follows from (3.12), (3.16), and the demiclosedness of U and T that $\bar{x}\in F(U)$ and $A\bar{x}\in F(T)$, which imply that $\bar{x}\in\Gamma$, so $\omega_w(x_k)\subseteq\Gamma$.

Finally, by Lemma 2.5, we have $x_k\rightharpoonup x^*$ as $k\to\infty$, where $x^*\in\Gamma$. Thus, it follows from $v_k\to 0$ that $(v_k,x_k)\rightharpoonup(0,x^*)$ as $k\to\infty$. □

Remark 3.5. (i) When $\alpha_k=\beta_k=0$, Algorithm 3.1 becomes the self-adaptive primal-dual algorithm (1.8) which was proposed in [25] for solving the SCFP of directed operators. (ii) When $\beta_k=0$, Algorithm 3.1 becomes the one-step inertial adaptive iterative algorithm (1.11) for solving the SCFP of directed operators.

4. Numerical Experiments

In this section, we carry out a numerical experiment and demonstrate the performance of the proposed Algorithm 3.1 for solving the SFP (1.2) by comparing Algorithm 3.1, the original algorithm (1.8), and the self-adaptive one-step inertial iterative algorithm (1.11). All the codes are written by MATLAB and are performed on a personal ASUS computer with AMD Ryzen™ 7 5800H CPU @3.2GHz 4.4GHz and RAM 16.00GB. We denote $e_0=(0,0,\cdots,0)^T$ and $e_1=(1,1,\cdots,1)^T$, and in the table, we use 'Iter.' to denote the number of iteration.

Example 4.1. Let $A=\left(a_{ij}\right)_{N\times M}$ be a random matrix, where $a_{ij}\in[-40,-20]$ and N,M are two positive integers. Choose a M-dimensional negative vector z. Let $r=\|z\|$ and $b=Az$. Take

$$C=\left\{x=(x_i)\in R^M \mid \sum_{i=1}^{M}x_i^2\leqslant r\right\},$$

and

$$Q=\left\{y\in R^N \mid y\leqslant b\right\}.$$

Now we find $x\in C$ and $Ax\in Q$. It is easy to see that $\Gamma\neq\varnothing$. In Algorithm 3.1, the directed operators U and T become projection operators P_C and P_Q, respectively, and we take $\theta=0.1$, $\gamma=0.5$, $\rho_k=1.95$, $\varepsilon_k=\frac{1}{k^{1.01}}$. We define the function $p(x)$ by

$$p(x)=\|x-P_Cx\|+\|Ax-P_QAx\|$$

and we take $p(x)<\varepsilon=10^{-20}$ as the stopping criterion.

In order to solve our example, we can take inertial extrapolation factor α_k, $\beta_k\in[0,\overline{\alpha}_k]$. In our example, if $\beta_k=0$, Algorithm 3.1 becomes the self-adaptive inertial iterative algorithm (1.11) with one-step inertial technique, and if $\alpha_k = \beta_k \equiv 0$, Algorithm 3.1 becomes the primal-dual algorithm (1.11) without inertial technique and we can choose different inertial extrapolation factors by adjusting parameter $\tau \in [0,1]$.

In Table 1-Table 3, we present out numerical experiments with different dimension spaces and inertial extrapolation factors. Let $\alpha_k = \beta_k = \tau\overline{\alpha}_k$. We show iteration numbers with dimesions $(N,M) = (20,40),(30,50),(40,60)$ and initial points x_0, x_1, x_2, ω_0 are generated randomly. In addition, we adjust parameters $\tau = 0, 0.1, 0.2, ..., 0.9, 1.0$.

We can find that Algorithm 3.1 is more effective for solving our example with different dimension spaces and inertial extrapolation factors.

TABLE 1. Numerical result with different α_k, β_k, where the two cases are $\alpha_k = \tau\overline{\alpha}_k$, $\beta_k = 0$ and $\alpha_k = \beta_k = \tau\overline{\alpha}_k$.

(N, M)=(20, 40)												
τ		0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
$\alpha_k = \tau\overline{\alpha}_k$, $\beta_k = 0$	*Iter.*	346	55	44	40	37	34	32	31	30	29	28
$\alpha_k = \beta_k = \tau\overline{\alpha}_k$	*Iter.*	346	42	33	30	27	26	25	24	23	23	22

TABLE 2. Numerical result with different α_k, β_k, where the two cases are $\alpha_k = \tau\overline{\alpha}_k$, $\beta_k = 0$ and $\alpha_k = \beta_k = \tau\overline{\alpha}_k$.

(N, M)=(30, 50)												
τ		0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
$\alpha_k = \tau\overline{\alpha}_k$, $\beta_k = 0$	*Iter.*	316	52	44	39	36	34	31	30	29	28	27
$\alpha_k = \beta_k = \tau\overline{\alpha}_k$	*Iter.*	316	41	34	30	28	27	25	25	24	23	23

TABLE 3. Numerical result with different α_k, β_k, where the two cases are $\alpha_k = \tau\overline{\alpha}_k$, $\beta_k = 0$ and $\alpha_k = \beta_k = \tau\overline{\alpha}_k$.

(N, M)=(40, 60)												
τ		0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
$\alpha_k = \tau\overline{\alpha}_k$, $\beta_k = 0$	*Iter.*	404	54	46	41	37	34	32	30	28	27	26
$\alpha_k = \beta_k = \tau\overline{\alpha}_k$	*Iter.*	404	41	32	29	27	25	25	24	23	23	22

Funding

This paper was supported by the Innovation and Entrepreneurship Training Program for College Students of Civil Aviation University of China (Grant No. 202210059071).

REFERENCES

[1] 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.

[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] A. Cegielski, General method for solving the split common fixed point problem, J. Optim. Theory Appl. 165 (2015) 385-404.

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

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

[7] 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.

[8] Y. Dang, J. Sun, H. Xu, Inertial accelerated algorithms for solving a split feasibility problem, J. Ind. Manag. Optim. 13 (2017) 1383-1394.

[9] Q.L. Dong, A. Gibali, D. Jiang, S.H. Ke, Convergence of projection and contraction algorithms with outer perturbations and their applications to sparse signals recovery, J. Fixed Point Theory Appl. 20 (2018) 16.

[10] Q.L. Dong, J.Z. Huang, X.H. Li, Y.J. Cho, Th.M. Rassias, MiKM: multi-step inertial Krasnosel'skiĭ–Mann algorithm and its applications, J. Global. Optim. 73 (2019) 801-824.
Q.L. Dong, An alternated inertial general splitting method with linearization for the split feasibility problem, Optimmization, 10.1080/02331934.2022.2069567.

[11] M. Eslamian, G.Z. Eskandani, M. Raeisi, Split common null point and common fixed point problems between Banach spaces and Hilbert spaces, Mediterr. J. Math. 14 (2017) 119.

[12] D.M. Giang, J.J. Strodiot, V.H. Nguyen, Strong convergence of an iterative method for solving the multiple-set split equality fixed point problem in a real Hilbert space, RACSAM, 111 (2017) 983-998.

[13] 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. Ind. Manage. Optim. 15 (2019) 963-984.

[14] 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.

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

[16] A. Moudafi, A note on the split common fixed point problem for quasi-nonexpansive operators, Nonlinear Anal. 74 (2011) 4083-4087.

[17] Y. Nesterov, A method for solving the convex programming problem with convergence rate O(1/k2), Dokl. Akad. Nauk Sssr, 269 (1983) 543-547.

[18] 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.

[19] X. Qin, J.C. Yao, A viscosity iterative method for a split feasibility problem, J. Nonlinear Convex Anal. 20 (2019) 1497-1506.

[20] 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.

[21] Y. Shehu, P.T. Vuong, P. Cholamjiak, A self-adaptive projection method with an inertial technique for split feasibility problems in Banach spaces with applications to image restoration problems, J. Fixed Point Theory Appl. 21 (2019) 50.

[22] F. Wang, H.K. Xu, Cyclic algorithms for split feasibility problems in Hilbert spaces, Nonlinear Anal. 74 (2011) 4105-4111.

[23] L. Yang, F. Zhao, J.K. Kim, The split common fixed point problem for demicontractive mappings in Banach spaces, J. Computational Anal. Appl. 22 (2017) 858-863.

[24] J. Zhao, S. He, Viscosity approximation methods for split common fixed-point problem of directed operators, Numerical Functional Anal. Optim. 36 (2015) 528-547.

[25] J. Zhao, D. Hou, A self-adaptive iterative algorithm for the split common fixed point problems, Numer. Algor 82 (2019) 1047-1063.

[26] J. Zhao, N.N. Zhao, D. Hou, Inertial accelerated algorithms for the split common fixed-point problem of directed operators, Optimization 70 (2021) 1-33.