3-2023

An inertial triple-projection algorithm for solving the split feasibility problems

Yazheng DANG

Marcus ANG
Singapore Management University, marcusang@smu.edu.sg

Jie SUN

Citation

DANG, Yazheng; ANG, Marcus; and SUN, Jie. An inertial triple-projection algorithm for solving the split feasibility problems. (2023). *Journal of Industrial and Management Optimization*. 19, (3), 1813-1826.
Available at: https://ink.library.smu.edu.sg/lkcsb_research/7132

Journal of Industrial and Management Optimization
Vol. 19, No. 3, March 2023, pp. 1813–1826
doi:10.3934/jimo.2022019

AN INERTIAL TRIPLE-PROJECTION ALGORITHM FOR SOLVING THE SPLIT FEASIBILITY PROBLEM

Yazheng Dang*

Department of Management
University of Shanghai for Science and Technology, Shanghai PRC 200093

Marcus Ang

Lee Kong Chian School of Business
Singapore Management University, Singapore 178899

Jie Sun

School of Electric Engineering, Computing, and Mathematical Science
Curtin University, Bentley, Australia 6102 and
School of Business
National University of Singapore, Singapore 119245

(Communicated by Bin Li)

Abstract. This paper proposes a new inertial triple-projection algorithm for solving the split feasibility problem. The process of projections is divided into three parts. Each part adopts a different variable stepsize to obtain its projection point, which is different from the existing extragradient methods. Flexible rules are employed for selecting the stepsizes and the inertial technique is used for improving the convergence. Convergence results are proven. Numerical experiments show that the proposed method converges more quickly than the general CQ algorithm.

1. **Introduction.** The split feasibility problem (SFP) is to find a point x^* satisfying

$$x^* \in C, \ Ax^* \in Q, \tag{1.1}$$

where C and Q are nonempty closed convex sets of $\Re^N$ and $\Re^M$, respectively, and A is an M by N real matrix. It could arise in many fields such as approximation theory [11] and image reconstruction [5, 6, 14] etc. The projection method is a general way to solve the SFP. Let P_C denote the orthogonal projection onto C; that is, $P_C(x) = \arg\min_{y\in C} \|x-y\|$. The CQ algorithm [4] presented by Byrne is a classical method that takes an initial point x^0 arbitrarily, and calculates the iterative step via

$$x^{k+1} = P_C\left((I-\gamma A^T(I-P_Q)A)x^k\right), \tag{1.2}$$

where I is the unit matrix of suitable dimension, $0<\gamma<2/\rho(A^TA)$ and $\rho(A^TA)$ is the spectral radius of A^TA.

2020 *Mathematics Subject Classification.* 37C25, 47H09, 90C25.
Key words and phrases. Split feasibility problem, triple-projection algorithm, Armijo-type line search, inertial technique, convergence analysis.

*Corresponding author: Yazheng Dang.

Inspired by the CQ algorithm, many other projection methods have been developed for solving the SFP, see, for example, [2, 7, 9, 12, 20]. Most of these algorithms use invariable stepsize related to a Lipschitz constant, which is inflexible, and only use the current iterate to obtain the next iterate, which may lead to slow convergence. There are two ways to improve this situation. The first way is developed by He [13], in which a method with self-adaptive stepsize was used for solving a variational problem. The numerical results in [13] have shown that the self-adaptive strategy is valid and robust. Subsequently, many self-adaptive projection methods were proposed for solving the SFP [8, 21, 22]. The second way to improve convergence is due to Polyak [16], who proposed an inertial gradient method to speed up convergence in solving smooth convex minimization problems. The main idea of the inertial method is to make use of two previous iterates in updating the next iterate. Since the presence of inertial term in the algorithm speeds up the convergence, many inertial algorithms have been widely studied (see, for example, [1, 6, 17]).

Motivated by the self-adaptive method presented in He [13] and Zhao [22] for solving multiple-sets split problem and by the inertial strategy proposed of Polyak [16], we propose an inertial triple-projection algorithm for solving the SFP, which uses different variable stepsize at different projection steps instead of the same stepsize as in [1, 6, 19]. In addition, in the first projection step, we combine the Armijo linear-search technology with the inertial strategy to obtain the first projection point; in the second projection step, we adopt a self-adaptive stepsize to obtain the second projection point; and finally, we generate the next iteration x^{k+1} by using the third projection of the initial point on the intersection of three sets. This algorithm is shown to be globally convergent to a solution under certain mild assumptions. Preliminary numerical experiments show that the proposed method may converge more quickly than the CQ algorithm.

The main contributions of the paper are as follows.

(i) The proposed algorithm employs different stepsizes at different projection steps instead of using a fixed stepsize;

(ii) Our algorithm uses the iterate x^{k+1} generated by the projection of the current point on the intersection of two halfspaces and a convex set C, instead of only on the current projection region. This improves the efficiency of convergence without increasing the computational cost.

(iii) To construct a halfspace that contains the solution set, we employ two previous iterative points with self-adaptive stepsizes to accelerate the rate of convergence.

2. Preliminaries.

Let I denote the identity operator, $\mathrm{Fix}(T)$ denote the set of the fixed points of an operator T *i.e.*, $\mathrm{Fix}(T) := \{x \mid x = Tx\}$, let Γ denote the solution set of the SFP, that is,

$$\Gamma = \{y \in C \mid Ay \in Q\}. \tag{2.1}$$

The following definitions and results will be used later on.

Definition 2.1. Given $T : \Re^N \to \Re^N$,

a) T is said to be monotone if

$$\langle T(x) - T(y), x - y\rangle \geq 0, \forall x, y \in \Re^N;$$

b) T is said to be nonexpansive if

$$\|T(x) - T(y)\| \leq \|x - y\|, \forall x, y \in \Re^N;$$

c) T is said to be co-coercive on $\Re^N$ with modulus $\alpha > 0$, if

$$\langle T(x) - T(y), x - y\rangle \geq \alpha\|T(x) - T(y)\|^2, \forall x, y \in \Re^N;$$

d) T is said to be Lipschitz continuous on $\Re^N$ with constant $L > 0$, if

$$\|T(x) - T(y)\| \leq L\|x - y\|, \forall x, y \in \Re^N.$$

Lemma 2.1 [8]**.** *Let C be a nonempty closed convex subset in $\Re^N$ and Let $P_C(x)$ be the projection of x on C. Then for any $x, y \in \Re^N$ and $z \in C$,*
(1) $\langle P_C(x) - x, z - P_C(x)\rangle \geq 0$;
(2) $\|P_C(x) - P_C(y)\|^2 \leq \langle P_C(x) - P_C(y), x - y\rangle$;
(3) $\|P_C(x) - z\|^2 \leq \|x - z\|^2 - \|P_C(x) - x\|^2$;
(4) $\|P_C(x) - P_C(y)\| \leq \|x - y\| - \|P_C(x) - x + y - P_C(y)\|$.

Remark 2.1. From item (2) of Lemma 2.1, we know that P_C is a monotone, co-coercive with modulus 1 and nonexpansive operator. Moreover, the operator $I - P_C$ is also co-coercive with modulus 1.
Lemma 2.2 [17]. *Let F be a mapping from $\Re^N$ into $\Re^N$. For any $x \in \Re^N$ and $\alpha \geq 0$, define $x(\alpha) = P_C(x - \alpha F(x))$ and $e(x, \alpha) = x - x(\alpha)$. Then, we have*

$$\min\{1, \alpha\}\|e(x, 1)\| \leq \|e(x, \alpha)\| \leq \max\{1, \alpha\}\|e(x, 1)\|.$$

3. **Inertial triple-projection algorithm and its convergence.**

In the following, we define the function $F : \Re^N \to \Re^N$ as

$$F(x) := A^T(I - P_Q)Ax$$

and respectively define

$$x(\beta_k) := P_C(x^k - \beta_k F(x^k)) \text{ and } e(x^k, \beta_k) := x^k - x(\beta_k).$$

By Lemma 8.1 in [5], the operator F_k is $1/\rho(A^TA)$-inverse strongly monotone or co-coercive with modulus $1/\rho(A^TA)$ and Lipschitz continuous with $\rho(A^TA)$, where $\rho(A^TA)$ is the largest eigenvalue of the matrix A^TA.

Next, we describe our inertial triple-projection algorithm.

Algorithm 3.1
Step 0. Select initial points $x^0 = x^1 \in C$ arbitrarily, parameters $\gamma > 0, l \in (0, 1), \lambda > 1, t_k \in \Theta = [t_{\min}, t_{\max}]$ for some fixed $0 < t_{\min} < t_{\max} < 2$, $\{\theta_k\} \subset (0, 1)$. Set $k = 1$.
Step 1. Set

$$w^k = P_C(x^k + \theta_k(x^k - x^{k-1})). \tag{3.1}$$

Compute

$$v^k = P_C(w^k - \beta_k F(w^k)), \tag{3.2}$$

where $\beta_k = \gamma l^{m_k}$ and m_k is the smallest nonnegative integer such that

$$\langle F(w^k), w^k - v^k\rangle \geq \lambda\langle F(w^k) - F(v^k), w^k - v^k\rangle. \tag{3.3}$$

Step 2. Find

$$u^k = P_C\left(w^k - t_k \frac{\langle F(v^k), w^k - v^k\rangle}{\|F(v^k)\|^2} F(v^k)\right). \tag{3.4}$$

Step 3. Find

$$S_k := \{u \in \Re^N | \|u^k - u\| \le \|w^k - u\|\}, \tag{3.5}$$

and

$$H_k = \{u | \langle x^k - u, x^1 - x^k \rangle \ge 0\}. \tag{3.6}$$

Step 4. Compute

$$x^{k+1} = P_{C \bigcap S_k \bigcap H_k}(x^1). \tag{3.7}$$

Set $k = k + 1$ and go to Step 1.

From the following lemma, we can see (3.3) is well defined.

Lemma 3.1. *There exists a nonnegative number m_k satisfying (3.3), for $k \ge 0$.*

Proof. By (3.1) we have $w^k \in C$, so $P_C(w^k) = w^k$. By (2) of Lemma 2.1, we have

$$\begin{aligned}\langle F(w^k), w^k - v^k \rangle &= \frac{1}{\beta_k} \langle \beta_k F(w^k), P_C(w^k) - P_C(w^k - \beta_k F(w^k)) \rangle \\ &\ge \frac{1}{\beta_k} \|w^k - v^k\|^2. \end{aligned} \tag{3.8}$$

By the inequality $\langle a, b \rangle \le \frac{\|a\|^2}{2} + \frac{\|b\|^2}{2}$ and the nonexpansiveness of F, we get

$$\begin{aligned}\langle F(w^k) - F(v^k), w^k - v^k \rangle &\le \frac{\|F(w^k) - F(v^k)\|^2}{2} + \frac{\|w^k - v^k\|^2}{2} \\ &\le \frac{\rho(A^T A)^2 + 1}{2} \|v^k - w^k\|^2, \end{aligned} \tag{3.9}$$

where $\rho(A^T A)$ is the largest eigenvalue of the matrix $A^T A$. Obviously, there must exist a constant m such that $\frac{1}{\gamma l^m} \ge \frac{\lambda(\rho(A^T A)^2 + 1)}{2}$. Hence

$$\begin{aligned}\langle F(w^k), w^k - v^k \rangle &\ge \frac{1}{\gamma l^m} \|w^k - v^k\|^2 \\ &\ge \frac{\lambda(\rho(A^T A)^2 + 1)}{2} \|w^k - v^k\|^2 \\ &\ge \lambda \langle F(w^k) - F(v^k), w^k - v^k \rangle, \end{aligned}$$

the proof is completed. □

Lemma 3.2. *Suppose $\Gamma \ne \emptyset$ and the sequences $\{w^k\}$ and $\{v^k\}$ are generated by Algorithm 3.1. Then, $-F(v^k)$ is a descent direction of the function $\frac{1}{2}\|w - z\|^2$ at the point w^k, where $z \in \Gamma$.*

Proof. From (3.3) and (3.7), one has

$$\begin{aligned}\langle F(v^k), w^k - v^k \rangle &= \langle F(v^k) - F(w^k), w^k - v^k \rangle + \langle F(w^k), w^k - v^k \rangle \\ &\ge (1 - \frac{1}{\lambda}) \langle F(w^k), w^k - v^k \rangle \\ &\ge (1 - \frac{1}{\lambda}) \frac{1}{\beta_k} \|w^k - v^k\|^2, \end{aligned}$$

that is

$$\langle F(v^k), w^k - v^k \rangle \ge (1 - \frac{1}{\lambda}) \frac{1}{\gamma} \|w^k - v^k\|^2 \ge 0. \tag{3.10}$$

Obviously, for $z \in \Gamma$, $F(z) = 0$. Since F is monotonic and $z \in \Gamma$, we have

$$\begin{aligned}\langle F(v^k), w^k - z\rangle &= \langle F(v^k), w^k - v^k\rangle + \langle F(v^k), v^k - z\rangle \\ &\geq \langle F(v^k), w^k - v^k\rangle + \langle F(z), v^k - z\rangle,\end{aligned}$$

that is,

$$\langle F(v^k), w^k - z\rangle \geq \langle F(v^k), w^k - v^k\rangle. \tag{3.11}$$

Combining (3.11) with (3.10), we obtain the result. □

Remark 3.1. From the monotonicity of F, we know that

$$\langle F(v^k), z - v^k\rangle \leq \langle F(z), z - v^k\rangle = 0,$$

along with (3.11), we obtain that the hyperplane

$$H_k := \{x \in \Re^N | \langle F(v^k), x - v^k\rangle = 0\}$$

separates w^k from the set Γ.

The following lemmas are also important for our convergence analysis.

Lemma 3.3. *Suppose that the solution set Γ is nonempty. Then the sequence $\{x^k\}$ generated by Algorithm 3.1 is well defined.*

Proof. In view of Step 4 of the algorithm, it suffices to show that $C \bigcap S_k \cap H_k \neq \emptyset$ (Obviously, $C \cap S_k \cap H_k$ is a closed and convex set $\forall k \geq 1$). We will in fact show that $\Gamma \subset C \bigcap S_k \cap H_k$. The fact that $\Gamma \subset C$ is obvious by (1.1). To show that $\Gamma \subset S_k$, let $z \in \Gamma, \alpha_k = \frac{\langle F(v^k), w^k - v^k\rangle}{\|F(v^k)\|^2}$. From (3.4), we have

$$\begin{aligned}\|u^k - z\|^2 &= \|P_C[w^k - t_k\alpha_k F(v^k)] - z\|^2 \\ &\leq \|w^k - z - t_k\alpha_k F(v^k)\|^2 \\ &= \|w^k - z\|^2 - 2t_k\alpha_k\langle F(v^k), w^k - z\rangle + t_k^2\alpha_k^2\|F(v^k)\|^2 \\ &\leq \|w^k - z\|^2 - 2t_k\alpha_k\langle F(v^k), w^k - v^k\rangle + t_k^2\alpha_k^2\|F(v^k)\|^2.\end{aligned}$$

Hence,

$$\|u^k - z\|^2 \leq \|w^k - z\|^2 - t_k(2 - t_k)\frac{\langle F(v^k), w^k - v^k\rangle^2}{\|F(v^k)\|^2}. \tag{3.12}$$

From (3.10), we obtain

$$\|u^k - z\|^2 \leq \|w^k - z\|^2 - t_k(2 - t_k)(1 - \frac{1}{\lambda})^2\frac{1}{\gamma^2}\frac{\|w^k - v^k\|^4}{\|F(v^k)\|^2}. \tag{3.13}$$

By (3.13), we know that for all k,

$$\|u^k - z\|^2 \leq \|w^k - z\|^2, \tag{3.14}$$

so $z \in S_k$, which implies $\Gamma \subset S_k$, for all $k \in N$. We next prove that $\Gamma \subset H_k$ for all $k \geq 1$ by induction. Obviously, if $k = 1$, then

$$\Gamma \subseteq H_1 = \Re^N.$$

Suppose that $\Gamma \subset H_k$. Then $\Gamma \subset S_k \cap H_k \cap C$.

For any $z \in \Gamma$, by Lemma 2.1 and the fact that

$$x^{k+1} = P_{S_k \cap H_k \cap C}\left(x^1\right),$$

we have that

$$\left\langle z - x^{k+1}, x^1 - x^{k+1}\right\rangle \leq 0$$

Thus, $z \in H_{k+1}$. The induction is complete, which concludes that $\Gamma \subset S_k \cap H_k \cap C$ for all $k \geq 1$. □

Lemma 3.4. *Let $\{x^k\}$ be a sequence generated by Algorithm 3.1. Then*
1) $\lim_{k\to\infty} \|x^{k+1} - x^k\| = 0$;
2) $\lim_{k\to\infty} \|x^k - w^k\| = 0,\ \lim_{k\to\infty} \|u^k - w^k\| = 0$;
3) $\lim_{k\to\infty} \|u^k - v^k\| = 0,\ \lim_{k\to\infty} \|x^k - v^k\| = 0$;
4) $\lim_{k\to\infty} \|(I - P_Q)Aw^k\| = 0$.

Proof. For 1). Let $z \in \Gamma$. Since $\Gamma \subset C \bigcap S_k \bigcap H_k, \forall k \geq 1$ and $x^{k+1} = P_{C \bigcap S_k \bigcap H_k}$ (x^1), we have

$$\|x^{k+1} - x^1\| \leq \|z - x^1\|, \quad \forall k \geq 1.$$

Thus,$\{\|x^{k+1} - x^1\|\}$ is bounded.

We observe that $x^{k+1} \in H_k$ and by (3.6) we have

$$\langle x^k - x^1, x^k - x^{k+1}\rangle \leq 0. \tag{3.15}$$

Then

$$\begin{aligned} \|x^{k+1} - x^k\|^2 &= \|x^{k+1} - x^1 - (x^k - x^1)\|^2 \\ &= \|x^{k+1} - x^1\|^2 - 2\langle x^{k+1} - x^1, x^k - x^1\rangle + \|x^k - x^1\|^2 \\ &= \|x^{k+1} - x^1\|^2 - 2\langle x^{k+1} - x^k + x^k - x^1, x^k - x^1\rangle + \|x^k - x^1\|^2 \\ &= \|x^{k+1} - x^1\|^2 - 2\langle x^{k+1} - x^k, x^k - x^1\rangle - \|x^k - x^1\|^2. \end{aligned}$$

From (3.15), we have

$$\|x^{k+1} - x^k\|^2 \leq \|x^{k+1} - x^1\|^2 - \|x^k - x^1\|^2, \tag{3.16}$$

that is, $\|x^k - x^1\|^2 \leq \|x^{k+1} - x^1\|^2 - \|x^{k+1} - x^k\|^2$. Thus, $\|x^k - x^1\|^2 \leq \|x^{k+1} - x^1\|^2$, therefore, $\{\|x^k - x^1\|\}$ is a bounded monotone nondecreasing sequence. Hence, $\lim_{k\to\infty} \|x^k - x^1\|$ exists. From (3.16), we have

$$\lim_{k\to\infty} \|x^{k+1} - x^k\| = 0.$$

For 2). By 1), we have

$$\lim_{k\to\infty} \|x^{k+1} - x^k\| = \lim_{k\to\infty} \|x^k - x^{k-1}\| = 0.$$

Recall (3.1)

$$w^k = P_C(x^k + \theta_k(x^k - x^{k-1})),$$

Since $x^k \in C$, by Lemma 2.1(3), we have

$$\|w^k - x^k\| = \|P_C(x^k + \theta_k(x^k - x^{k-1})) - x^k\| \leq |\theta_k|\|x^k - x^{k-1}\|.$$

From 1), we have

$$\lim_{k\to\infty} \|x^k - w^k\| = 0. \tag{3.17}$$

From 1) and (3.17) , we get

$$\|x^{k+1} - w^k\| = \|x^{k+1} - x^k + x^k - w^k\| \leq \|x^{k+1} - x^k\| + \|x^k - w^k\| \to 0, \text{as } k \to \infty.$$

By the construction of S_k, we have

$$\|x^{k+1} - u^k\| \leq \|x^{k+1} - w^k\| \to 0, as\ k \to \infty,$$

that is, $\lim_{k\to\infty}\|x^{k+1}-u^k\|=0$.
Similarly, $\|x^k-u^k\|\le\|x^k-x^{k+1}\|+\|x^{k+1}-u^k\|$, it then follows that

$$\lim_{k\to\infty}\|x^k-u^k\|=0. \tag{3.18}$$

It follows from (3.17) and (3.18) that

$$\lim_{k\to\infty}\|w^k-u^k\|=0. \tag{3.19}$$

For 3). Clearly,

$$\begin{aligned}\|F(v^k)\| &\le \|F(v^k)-z\|+\|z\|\le\rho(A^TA)\|w^k-\beta_kF(w^k)-z\|+\|z\|\\ &\le\rho(A^TA)\|w^k-z\|+\gamma\|F_k(w_k)\|+\|z\|.\end{aligned} \tag{3.20}$$

In fact, by the boundedness of $\{w^k\}$ and the continuity of F, we know that $\{F(w^k)\}$ is also bounded. Thus, there exists a constant $M>0$ such that $\|F(v^k)\|\le M$ for all k. Consequently, we obtain from (3.13) and the definition of t_k that

$$\lim_{k\to\infty}\|w^k-v^k\|=0. \tag{3.21}$$

From (3.19) and (3.21), it is easy to see that

$$\lim_{k\to\infty}\|u^k-v^k\|=0. \tag{3.22}$$

Thus, from (3.18) and (3.22) we obtain

$$\lim_{k\to\infty}\|x^k-v^k\|=0. \tag{3.23}$$

For 4). Define

$$e(x,\mu)=x-P_C(x-\mu F(x))$$

Then from Lemma 2.2, the definition of β_k and equation (3.21), we have

$$\begin{aligned}\lim_{k\to\infty}\|e(w^k,1)\| &\le \lim_{k\to\infty}\frac{\|w^k-v^k\|}{\min\{1,\beta_k\}}\\ &\le\lim_{k\to\infty}\frac{\|w^k-v^k\|}{\min\{1,\underline{\beta}\}}=0,\end{aligned} \tag{3.24}$$

where $\underline{\beta}=\frac{l}{\lambda(\rho(A^TA)^2+1}$. Using part (1) of Lemma 2.1 and note that $x^*\in\Gamma$, we have for all $i=1,2,\cdots$,

$$\langle w^k-F(w^k)-P_C(w^k-F(w^k)),x^*-P_C(w^k-F(w^k))\rangle\le 0,$$

that is,

$$\langle e(w^k,1)-F(w^k),w^k-x^*-e(w^k,1)\rangle\ge 0.$$

From the above inequality and (1) of Lemma 2.1, we know for all $i=1,2,\cdots$,

$$\begin{aligned}\langle w^k-x^*,e(w^k,1)\rangle &\ge \|e(w^k,1)\|^2-\langle F(w^k),e(w^k,1)\rangle+\langle F(w^k),w^k-x^*\rangle\\ &= \|e(w^k,1)\|^2-\langle F(w^k),e(w^k,1)\rangle\\ &\quad+\langle F(w^k)-F(x^*),w^k-x^*\rangle\\ &= \|e(w^k,1)\|^2-\langle F(w^k),e(w^k,1)\rangle\\ &\quad+\langle (I-P_Q)(Aw^k)-A^T(I-P_Q)(Ax^*),Aw^k-Ax^*\rangle\\ &\ge \|e(w^k,1)\|^2-\langle F(w^k),e(w^k,1)\rangle\\ &\quad+\|(I-P_Q)(Aw^k)-A^T(I-P_Q)(Ax^*)\|^2.\end{aligned}$$

Hence,

$$\langle w^k - x^*, e_k(w^k,1)\rangle \le \|e_k(w^k,1)\|^2 - \langle F(w^k), e(w^k,1)\rangle + \|(I-P_Q)(Aw^k)\|^2. \quad (3.25)$$

Since

$$\|F(w^k)\| = \|F_k(w^k) - F_k(x^*)\| \le \rho(A^TA)\|w^k - x^*\|, \forall i = 1,2,\cdots,$$

and $\{w^{k_i}\}$ is bounded, the sequence $\{F(w^k)\}$ is also bounded. Therefore, from (3.24) and (3.25), we obtain

$$\lim_{k\to\infty} \|(I-P_Q)(Aw^k)\| = 0,$$

that is,

$$\lim_{k\to\infty} P_Q(Aw^k) - Aw^k = 0. \quad (3.26)$$

□

Theorem 3.1. *Suppose $\Gamma \neq \emptyset$. Then, the sequence $\{x^k\}$ generated by Algorithm 3.1 converged to $\tilde{x}$, where $\tilde{x} = P_\Gamma(x^1)$.*

Proof. We know that (1) of Lemma 3.4 implies that $\lim_{k\to\infty}\|x^k - x^1\|$ exists. Now, we show that $x^k \to \tilde{x} \in \Gamma$. Let $m, l \in N$, then from (3) of Lemma 2.1, we have

$$\|x^m - x^l\| = \|x^m - P_{C\cap S_{l-1}\bigcap H_{l-1}}x^1\|^2 \le \|x^m - x^1\|^2 - \|x^l - x^1\|^2$$

$$= \|x^m - x^1\|^2 - \|P_{C\cap S_{l-1}\bigcap H_{l-1}}x^1 - x^1\|^2 \to 0.$$

Therefore, $\|x^m - x^l\| \to 0$ as $m, l \to \infty$. Thus $\{x^k\}$ is a Cauchy sequence in C. Since C is closed and convex, it implies that there exists $\tilde{x} \in C$. Since $\|x^k - v^k\| \to 0$, $\|x^k - w^k\| \to 0$, then $w^k \to \tilde{x}$, hence $A\tilde{x} \in Q$. Therefore, $\tilde{x} \in \Gamma$.

Finally, we show that $\tilde{x} = P_\Gamma(x^1)$. Suppose there exists $\tilde{y} \in \Gamma$ such that $\tilde{y} = P_\Gamma(x^1)$. Then $\tilde{y} \in C \cap S_k \bigcap H_k$, and by the iterative sequence of Algorithm 3.1 we have

$$\|x^k - x^1\| \le \|\tilde{y} - x^1\|.$$

Thus

$$\|x^k - \tilde{y}\|^2 = \|x^k - x^1 + x^1 - \tilde{y}\|^2$$

$$= \|x^k - x^1\|^2 + \|x^1 - \tilde{y}\|^2 + 2\langle x^k - x^1, x^1 - \tilde{y}\rangle$$

$$\le \|\tilde{y} - x^1\| + \|x^1 - \tilde{y}\|^2 + 2\langle x^k - x^1, x^1 - \tilde{y}\rangle$$

Letting $k \to \infty$, we have

$$\|x^k - \tilde{y}\|^2 \le 2\|\tilde{y} - x^1\| + 2\langle x^k - x^1, x^1 - \tilde{y}\rangle$$

$$= 2\langle \tilde{x} - \tilde{y}, x^1 - \tilde{y}\rangle \le 0,$$

where the last inequality is due to Lemma 2.1 and the fact that $\tilde{y} = P_\Gamma(x^1)$ and $\tilde{x} \in \Gamma$. Hence,

$$\tilde{x} = \tilde{y} = P_\Gamma(x^1).$$

Thus, the sequence $\{x^k\}$ converges to a point $P_\Gamma(x^1)$. This completes the proof. □

4. **Preliminary numerical results.** We use two numerical examples to show that the algorithm converges faster than the CQ algorithm and double project algorithm (DPA)[16]. Throughout the computational experiments, we set $Error_k = \|x^{k+1} - x^k\|^2 \leq \varepsilon = 10^{-4}$ as the stop criterion. In the algorithm, we set $\lambda = 20, \gamma = 10, l = 0.01$ in Algorithm 3.1. We can see the numerical results of Examples 4.1-4.2 from the following Tables 1 and 2 and Figures 1 and 2. In these two tables , "k" ,"s" and "x^*" denote the number of iterates, cpu time in seconds and the solution, respectively. All codes are written in MATLAB7.0.

Example 4.1. Let

$$A = \begin{bmatrix} 2 & -1 & 3 \\ 4 & 2 & 5 \\ 2 & 0 & 2 \end{bmatrix};$$

$C = \{x \in \Re^3 | \ x_1 + x_2^2 + 2x_3 \leq 0\}$; and $Q = \{x \in \Re^3 | \ x_1^2 + x_2 - x_3 \leq 0\}$. Find $x \in C$ with $Ax \in Q$.

From Table 1 and Figure 1, we can see that Algorithm 3.1 converges more quickly than CQ algorithm and DPA. Figure 2 displays the changing trend of errors with different inertial factors, and we can see that the algorithm with bigger inertial factor converges more quickly than the algorithm with smaller inertial factor .

Example 4.2. Let $A = (a_{ij})_{M\times N}$, $a_{ij} \in (0, 1)$ be a random matrix, M, N be two positive integers. $C = \{x \in \Re^N | \ \sum_{l=1}^{N} x_l^2 \leq r^2\}$; $Q = \{x \in \Re^M | \ x \leq b\}$. The vector b is generated by using the following way: Given a random $N-$dimensional negative vector (each component is negative) $z \in C, r = \|z\|$, taking $b = Az$. Find $x \in C$ with $Ax \in Q$. We select $e_0 = (0, 0, \cdots, 0)$ as the initial point in the example.

Table 2 shows the numerical results of Example 4.2 for CQ Algorithm, DPA and Algorithm 3.1 with different t_k, respectively. The results tell us that the inertial technique has good behavior for the high dimensional case and the inertial effect is very obvious for the big inertial factor.

From Table 1 and Table 2, as well as from Figure 1 and Figure 2, we can see that Algorithm 3.1 is effective and promising for solving the SFP.

5. **Concluding remarks.** This paper presents a triple-projection method with different rules of stepsize selection and inertial technique for solving the SFP. The first projection point is obtained by employing an Amijo-line-search rule and inertial strategy; The second projection point is calculated by adopting a self-adaptive selection technique of stepsize, which is different from the self-adaptive projection methods proposed in [21] and [22] that use the co-coercivity and the Lipschitz continuity property of the gradient mappings. The next iterate is obtained from the current iterative point on the intersection of two halfspaces and the convex set C. Preliminary numerical results show that the proposed method is practical and promising for solving the SFP.

Acknowledgments. The first author's work was partially supported by National Natural Science Foundation of China under grants 72071130 and 71901145.

REFERENCES

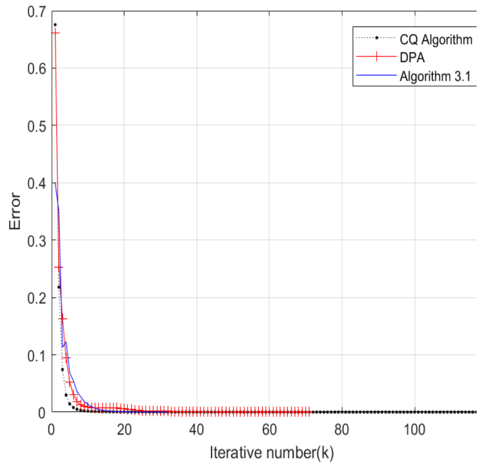
[1] P. K. Anh, D. V. Thong and N. T. Vinh, Improved inertial extragradient methods for solving pseudo-monotone variational inequalities, *Optimization*, **14** (2020), 1157–1175.

TABLE 1. The numerical results of Example 4.1

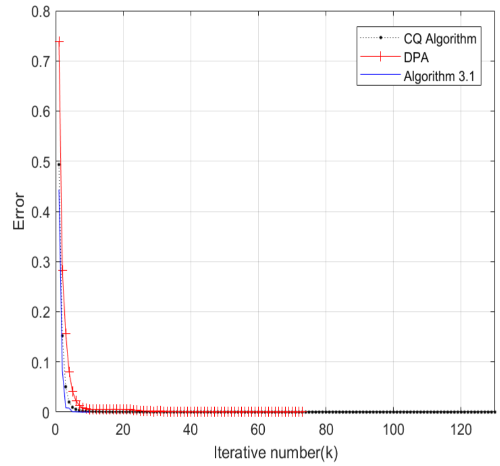
Initial point	CQ Algorithm with stepsize $\frac{1}{\rho(A^T A)}$)	DPA with $t_k = 1$	Algorithm 3.1 with $t_k = 1, \theta_k = 0.8$
$x^0 =$	$k = 121; s = 0.0469$	$k = 71; s = 0.3125$	$k = 32; s = 0.0156$
$(3, 2, 2)^T$	$x^* = (0.0026; -0.0012; -0.0012)^T$	$x^* = (0.4281; -0.2507; 0.8563)^T$	$x^* = (1.8244; -0.1457; -0.6787)^T$
$x^0 =$	$k = 130; s = 0.0625$	$k = 73; s = 0.2031$	$k = 10; s = 0.0469$
$(2, 3, 4)^T$	$x^* = (-0.0035; 0.0001; 0.0036)^T$	$x^* = (0.4281; -0.2507; 0.8563)^T$	$x^* = (-0.80761; 0.3253; 1.4823)^T$
$x^0 =$	$k = 222; s = 0.0781$	$k = 79; s = 0.1875$	$k = 16; s = 0.0625$
$(0, -1, 5)^T$	$x^* = (00031; -0.0054; 0.0085)^T$	$x^* = (0.4281; -0.2507; 0.8563)^T$	$x^* = (0.5849; 0.2902; 0.1249)^T$
$x^0 =$	$k = 219; s = 0.0938$	$k = 73; s = 0.2031$	$k = 24; s = 0.0469$
$(-2, 4, 3)^T$	$x^* = (-0.0086, 0.0055, 0.0031)^T$	$x^* = (0.4281; -0.2507; 0.8563)^T$	$x^* = (-0.0730; 0.1899; 0.7371)^T$

TABLE 2. The numerical results of Example 4.2

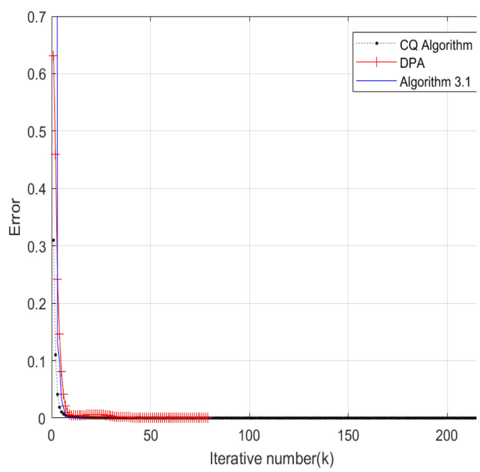
M, N	CQ with stepsize $\frac{1}{\rho(A^TA)}$	t_k	Algorithm3.1 with $\theta_k = 0.2$	Algorithm3.1 with $\theta_k = 0.8$	DPA
$M = 20$,	$k = 374, s = 0.2033$	0.8	$k = 227, s = 0.147$	$k = 168, s = 0.1251$	$k = 385, s = 0.2203$
$N = 10$		1.0	$k = 162, s = 0.1052$	$k = 120, s = 0.0701$	$k = 301, s = 0.1902$
		1.8	$k = 73, s = 0.0700$	$k = 54, s = 0.0512$	$k = 284, s = 0.1597$
$M = 100$,	$k = 2695, s = 5.7346$	0.4	$k = 2431, s = 7.32123$	$k = 2201, s = 6.3075$	$k = 2514, s = 8.4576$
$N = 90$		1	$k = 1071, s = 5.432$	$k = 661, s = 4.1297$	$k = 2303, s = 7.3501$
		1.6	$k = 464, s = 2.6507$	$k = 401, s = 2.1295$	$k = 1998, s = 6.9980$



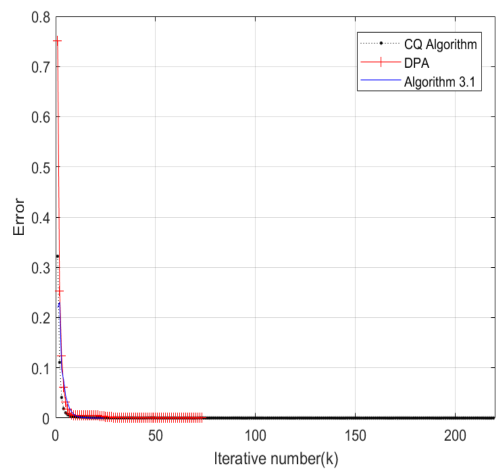
(a) For initial point $x^0 = (3, 2, 2)^T$



(b) For initial point $x^0 = (2, 3, 4)^T$



(c) For initial point $x^0 = (0, -1, 5)^T$



(d) For initial point $x^0 = (-2, 4, 3)^T$

FIGURE 1. The error results of Example of 4.1 for different initial points.

[2] N. Buong, P. Hoai and K. T. Binh, Iterative regularization methods for the multiple-sets split feasibility problem in hilbert spaces, *Acta Appl. Math.*, **165** (2020), 183–197.

[3] C. Byrne, An unified treatment of some iterative algorithm in signal processing and image reconstruction, *Inverse Problems*, **20** (2004), 103–120.

[4] C. Byrne, Iterative oblique projection onto convex sets and the split feasibility problem, *Inverse Problems*, **18** (2002), 441–453.

[5] Y. Censor, Parallel application of block iterative methods in medical imaging and radiation therapy, *Math. Program.*, **42** (1998), 307–325.

[6] Y. Censor and T. Elfving, A multiprojection algorithm using Bregman projections in a product space, *Numer. Algorithms*, **8** (1994), 221–239.

[7] P. Cholamjiak, S. Suantai and Suantai S., A new CQ algorithm for solving split feasibility problems in Hilbert spaces, *Bull. Malay. Math. Sci. Soc.*, **42** (2019), 2517–2534.

[8] Y. Z. Dang and J. Sun, Double projection algorithms for solving the split feasibility problems, *J. Indus. Manag. Optim.*, **15** (2019), 2023–2034.

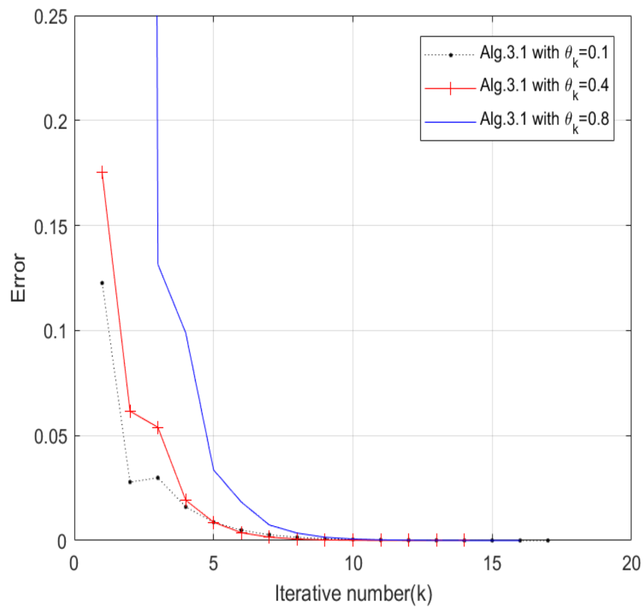


FIGURE 2. The errors of Example 4.1 with initial point $x^0 = (0, -1, 5)^T, t_k = 1$ for different inertial factor θ_k.

[9] Y. Dang, J. Yao and Y. Gao, Relaxed two points projection method for solving the multiple-sets split equality problem, *Numer. Algorithms*, **78** (2018), 263–275.

[10] Y. Z. Dang, J. Sun and H. Xu, Inertial accelerated algorithms for solving a split feasibility problem, *J. Indus. Manag. Optim.*, **13** (2017), 1383–1394.

[11] F. Deutsch, The method of alternating orthogonal projections, *Approximation Theory, Spline Functions and Applications*, Kluwer Academic Publishers, Dordrecht, **356** (1992), 105–121.

[12] A. Gibali, D. T. Mai and N. T. Vinh, A new relaxed CQ algorithm for solving split feasibility problems in Hilbert spaces and its applications, *J. Indus. Manag. Optim.*, **15** (2019), 963–984.

[13] B. He, Inexact implicit methods for monotone general variational inequalities, *Math. Program.*, **86** (1999), 199–217.

[14] G. T. Herman, *Image Reconstruction From Projections: The Fundamentals of Computerized Tomography*, Academic Press, New York, 1980.

[15] N. Nadezhkina and W. Takahashi, Weak convergence theorem by an extragradient method for nonexpansive mappings and monotone mappings, *J. Optim. Theory Appl.*, **128** (2006), 191–201.

[16] B. T. Polyak, Some methods of speeding up the convergence of iterative methods, *J. Math. Anal. Appl.*, **110** (1985), 463–468.

[17] B. Qu and N. Xiu, A note on the CQ algorithm for the split feasibility problem, *Inverse Problems*, **21** (2005), 1655–1665.

[18] D. R. Sahu, Y. J. Cho, Q. L. Dong, M. R. Kashyap and X. H. Li, Inertial relaxed CQ algorithms for solving a split feasibility problem in Hilbert spaces, *Numer. Algorithms*, **87** (2021), 1075–1095.

[19] S. Suantai, N. Pholasa and P. Cholamjiak, Relaxed CQ algorithms involving the inertial technique for multiple-sets split feasibility problems, *Rev. R. Acad. Cienc. Exactas F¨ıs. Nat. Ser. A Mat. RACSAM*, **113** (2019), 1081–1099.

[20] G. H. Taddele, P. Kumam, A. G. Gebrie and K. Sitthithakerngkiet, Half-space relaxation projection method for solving multiple-set split feasibility problem, *Math. Comput. Appl.*, **25** (2020), Paper No. 47, 24 pp.

[21] W. X. Zhang, D. Han and Z. B. Li, A self-adaptive projection method for solving the multiple-sets split feasibility problem, *Inverse Problems*, **25** (2009), 115011, 16 pp.

[22] J. L. Zhao and Q. Yang, Self-adaptive projection methods for the multiple-sets split feasibility problem, *Inverse Problems*, **27** (2011), 035009, 13 pp.

Received August 2021; 1st revision November 2021; final revision December 2021; early access February 2022.

E-mail address: jgdyz@163.com
E-mail address: marcusang@smu.edu.sg
E-mail address: jie.sun@curtin.edu.au