Chen and Li *Journal of Inequalities and Applications* (2021) 2021:190
https://doi.org/10.1186/s13660-021-02725-5

Journal of Inequalities and Applications
a SpringerOpen Journal

RESEARCH Open Access

The inertial relaxed algorithm with Armijo-type line search for solving multiple-sets split feasibility problem

Wenting Chen[1] and Meixia Li[2*]

*Correspondence:
limeixia001@163.com
[2]School of Mathematics and Information Science, Weifang University, 261061, Weifang, Shandong, P.R. China
Full list of author information is available at the end of the article

Abstract

The multiple-sets split feasibility problem is the generalization of split feasibility problem, which has been widely used in fuzzy image reconstruction and sparse signal processing systems. In this paper, we present an inertial relaxed algorithm to solve the multiple-sets split feasibility problem by using an alternating inertial step. The advantage of this algorithm is that the choice of stepsize is determined by Armijo-type line search, which avoids calculating the norms of operators. The weak convergence of the sequence obtained by our algorithm is proved under mild conditions. In addition, the numerical experiments are given to verify the convergence and validity of the algorithm.

MSC: 65K05; 65K10; 49J52

Keywords: Multiple-sets split feasibility problem; Inertia relaxed algorithm; Armijo-type line search; Convergence

1 Introduction

Let H_1 and H_2 be real Hilbert spaces, $t \geq 1$ and $r \geq 1$ be integers, $\{C_i\}_{i=1}^t$ and $\{Q_j\}_{j=1}^r$ be the nonempty, closed, and convex subsets of H_1 and H_2.

In this paper, we study the multiple-sets split feasibility problem (MSSFP). This problem is to find a point x^* such that

$$x^* \in C = \bigcap_{i=1}^{t} C_i, \qquad Ax^* \in Q = \bigcap_{j=1}^{r} Q_j, \tag{1.1}$$

where $A : H_1 \to H_2$ is a given bounded linear operator and A^* is the adjoint operator of A. Censor et al. [6] first proposed this problem in finite-dimensional Hilbert spaces, mainly based on the inverse problem modeling in the intensity modulation radiation treatment modeling, the signal processing, and image reconstruction. Because of these implements, there are many algorithms that are proposed to work out the multiple-sets split feasibility problem, such as [24, 25, 28–30]. If $t = r = 1$, the multiple-sets split feasibility problem turns into the split feasibility problem, see [5].

It is well known that the split feasibility problem amounts to the following minimization problem:

$$\min \frac{1}{2}\left\|x - P_C(x)\right\|^2 + \frac{1}{2}\left\|Ax - P_Q(Ax)\right\|^2, \tag{1.2}$$

where P_C is the metric projection on C and P_Q is the metric projection on Q. It is important to note that since a projection on an ordinary closed convex set has not closed form method, it is difficult to calculate its projection. Fukushima [11] proposed a new relaxation projection formula to overcome this difficulty. Specifically, he calculated a projection on a convex functions level set by calculating a series of projections onto half-spaces containing the general level set. Yang [26] proposed a relaxed CQ algorithm for working out the split feasibility problem in the context of a finite-dimensional Hilbert space, in which closed convex subsets C and Q are the level sets of convex functions, which are proposed as follows:

$$C = \left\{x \in H_1 : c(x) \le 0\right\} \quad \text{and} \quad Q = \left\{y \in H_2 : q(y) \le 0\right\}, \tag{1.3}$$

where $c : H_1 \to R$ and $q : H_2 \to R$ are convex functions which are weakly lower semi-continuous. Meanwhile, they assumed that c is subdifferentiable on H_1 and ∂c is a bounded operator in any bounded subset of H_1. Similarly, q is subdifferentiable on H_2 and ∂q is also a bounded operator in any bounded subset of H_2. Then two sets are defined at point x_n as follows:

$$C_n = \left\{x \in H_1 : c(x_n) \le \langle \xi_n, x_n - x\rangle\right\}, \tag{1.4}$$

where $\xi_n \in \partial c(x_n)$, and

$$Q_n = \left\{y \in H_2 : q(Ax_n) \le \langle \zeta_n, Ax_n - y\rangle\right\}, \tag{1.5}$$

where $\zeta_n \in \partial q(Ax_n)$. We can easily see that C_n and Q_n are half-spaces. For all $n \ge 1$, we easily know that $C_n \supset C$ and $Q_n \supset Q$. Under this framework, the projection can be simply computed because of the particular form of the metric projection of the sets C_n and Q_n, for details, please see [18]. Using this framework, Yang [26] built a new relaxed CQ algorithm, which was used to solve the split feasibility problem by using the semi-spaces C_n and Q_n, rather than the sets C and Q. Whereafter, Shehu [19] came up with a relaxed CQ method with alternating inertial extrapolation step, which was used to solve the split feasibility problem by using the half spaces C_n and Q_n. At the same time, they verified their convergence in certain appropriate step size.

In this paper, we consider a class of multiple-sets split feasibility problem (1.1), where the convex sets are defined by

$$C_i = \left\{x \in H_1 : c_i(x) \le 0\right\} \quad \text{and} \quad Q_j = \left\{y \in H_2 : q_j(y) \le 0\right\}, \tag{1.6}$$

where $c_i : H_1 \to R$ $(i = 1, 2, \dots, t)$ and $q_j : H_2 \to R$ $(j = 1, 2, \dots, r)$ are the convex functions which are weakly lower semi-continuous. Meanwhile, it is assumed that c_i $(i = 1, 2, \dots, t)$ are subdifferentiable on H_1 and ∂c_i $(i = 1, 2, \dots, t)$ are the bounded operators in any bounded subsets of H_1. Similarly, q_j ($j = 1, 2, \dots, r$) are subdifferentiable on H_2 and ∂q_j ($j = 1, 2, \dots, r$) are the bounded operators in any bounded subsets of H_2. In the whole study, we represent the solution set of the multiple-sets split feasibility problem (1.1) by S, when it is consistent. Censor et al. [6] invented the following distance function:

$$f(x) = \frac{1}{2}\sum_{i=1}^{t} l_i \left\| x - P_{C_i}(x) \right\|^2 + \frac{1}{2}\sum_{j=1}^{r} \lambda_j \left\| Ax - P_{Q_j}(Ax) \right\|^2, \tag{1.7}$$

where l_i ($i = 1, 2, \dots, t$) and λ_j ($j = 1, 2, \dots, r$) are positive constants such that $\sum_{i=1}^{t} l_i + \sum_{j=1}^{r} \lambda_j = 1$. Then we know that

$$\nabla f(x) = \sum_{i=1}^{t} l_i \bigl(x - P_{C_i}(x)\bigr) + \sum_{j=1}^{r} \lambda_j A^*(I - P_{Q_j})Ax. \tag{1.8}$$

They proposed the following algorithm:

$$x_{n+1} = P_\Omega\bigl(x_n - \rho \nabla f_n(x_n)\bigr), \tag{1.9}$$

where $\Omega \subseteq R^N$ is the auxiliary brief nonempty closed convex set satisfying $\Omega \cap S \neq \emptyset$ and $\rho > 0$. When L was the Lipschitz constant of $\nabla f(x)$ and $\rho \in (0, 2/L)$, they proved that the sequence $\{x_n\}$ produced by (1.9) converged to a solution of the multiple-sets split feasibility problem.

In order to improve the practicability of the method, in allusion to the split convex programming problem, Nesterov [17] proposed the next iterative process.

$$\begin{aligned} &y_n = x_n + \theta_n(x_n - x_{n-1}), \\ &x_{n+1} = y_n - \lambda_n \nabla f(y_n), \quad n \ge 1, \end{aligned} \tag{1.10}$$

where λ_n is a positive array and $\theta_n \in [0, 1)$ is an inertial element. Besides that, there are many other correlative algorithms, for example, the inertial forward-backward splitting method, the inertial Mann method, and the moving asymptotes method, for details, please see [1–4, 10, 12, 14–16].

Under the motivation of the above study, we provide a relaxed CQ algorithm to solve the multiple-sets split feasibility problem by using an alternating inertial step. In this algorithm, the stepsize is determined by line search. Hence, it avoids the calculation of the operators norms. Furthermore, we prove the weak convergence of the algorithm under some mild conditions. In addition, the inertial factor of the controlling parameters β_n can be selected as far as possible to close to the one, such as [7–9, 20–23, 27].

The structure of the paper is as follows. The basic concepts, definitions, and related results are described in Sect. 2. The third section presents the algorithm and its proof, and the fourth section provides the corresponding numerical experiment, which verifies the validity and stability of the algorithm. The final summarization is offered in Sect. 5.

2 Preliminaries

In this section, we give some basic concepts and relevant conclusions. Suppose that H is a Hilbert space.

Look back upon that a mapping $T: H \to H$ is called

(a) *nonexpansive* if $\|Tx - Ty\| \leq \|x - y\|$ for all $x, y \in H$;
(b) *firmly nonexpansive* if $\|Tx - Ty\|^2 \leq \|x - y\|^2 - \|(I - T)x - (I - T)y\|^2$ for all $x, y \in H$. Equivalently, for all $x, y \in H$, $\|Tx - Ty\|^2 \leq \langle x - y, Tx - Ty\rangle$.

As we all know, T is firmly nonexpansive if and only if $I - T$ is firmly nonexpansive.

For a point $u \in H$ and C is a nonempty, closed, and convex set of H, there is a unique point $P_C u \in C$ such that

$$\|u - P_C u\| \leq \|u - y\|, \quad \forall y \in C, \tag{2.1}$$

where P_C is the metric projection of H on C. The following is a list of the significant quality of the metric projection. It is well known that P_C is the firmly nonexpansive mapping on C. Meanwhile, P_C possesses

$$\langle x - y, P_C x - P_C y\rangle \geq \|P_C x - P_C y\|^2, \quad \forall x, y \in H. \tag{2.2}$$

Moreover, the characteristic of the $P_C x$ is

$$P_C x \in C \quad \text{and} \quad \langle x - P_C x, P_C x - y\rangle \geq 0, \quad \forall y \in C. \tag{2.3}$$

This representation means that

$$\|x - y\|^2 \geq \|x - P_C x\|^2 + \|y - P_C x\|^2, \quad \forall x \in H, \forall y \in C. \tag{2.4}$$

Suppose that a function $f : H \to R$, the element $g \in H$ is thought to be the *subgradient* of f on a point x if

$$f(y) \geq f(x) + \langle y - x, g\rangle, \quad \forall y \in H. \tag{2.5}$$

Besides, $\partial f(x)$ is the subdifferential of f at the point x which is described by

$$\partial f(x) = \bigl\{g \in H : f(y) \geq f(x) + \langle y - x, g\rangle, \forall y \in H\bigr\}. \tag{2.6}$$

The function $f : H \to R$ is thought to be weakly lower semi-continuous on a point x if $\{x_n\}$ converges weakly to x. It means that

$$f(x) \leq \liminf_{n\to\infty} f(x_n). \tag{2.7}$$

Lemma 2.1 ([23]) *Suppose that* $\{C_i\}_{i=1}^t$ *and* $\{Q_j\}_{j=1}^r$ *are the closed and convex subsets of* H_1 *and* H_2, *and* $A : H_1 \to H_2$ *is the bounded linear operator. At the same time, suppose that* $f(x)$ *is a function described by* (1.7). *Then* $\nabla f(x)$ *is Lipschitz continuous with* $L = \sum_{i=1}^t l_i + \|A\|^2 \sum_{j=1}^r \lambda_j$ *as a Lipschitz constant.*

Lemma 2.2 ([19]) *Suppose* $x, y \in H$. *Then*
(*i*) $\|x + y\|^2 = \|x\|^2 + 2\langle x, y\rangle + \|y\|^2$;
(*ii*) $\|x + y\|^2 \leq \|x\|^2 + 2\langle y, x + y\rangle$;
(*iii*) $\|\alpha x + \beta y\|^2 = \alpha(\alpha + \beta)\|x\|^2 + \beta(\alpha + \beta)\|y\|^2 - \alpha\beta\|x - y\|^2, \forall \alpha, \beta \in R$.

Lemma 2.3 ([18]) *Suppose that the half-spaces C_k and Q_k are defined as* (1.4) *and* (1.5). *Then the projections onto them from the points x and y are given as follows, respectively*:

$$P_{C_k}(x) = \begin{cases} x - \frac{c(x^k)+\langle \xi^k, x-x^k\rangle}{\|\xi_k\|^2}\xi_k & \text{if } c(x^k) + \langle \xi^k, x-x^k\rangle > 0; \\ x & \text{if } c(x^k) + \langle \xi^k, x-x^k\rangle \leq 0; \end{cases}$$

and

$$P_{Q_k}(y) = \begin{cases} y - \frac{q(y^k)+\langle \zeta^k, y-y^k\rangle}{\|\zeta_k\|^2}\zeta_k & \text{if } q(y^k) + \langle \zeta^k, y-y^k\rangle > 0; \\ y & \text{if } q(y^k) + \langle \zeta^k, y-y^k\rangle \leq 0. \end{cases}$$

3 The algorithm and convergence analysis

For $n \geq 1$, define

$$C_i^n = \{x \in H_1 : c_i(x_n) \leq \langle \xi_i^n, x_n - x\rangle\}, \tag{3.1}$$

where $\xi_i^n \in \partial c_i(x_n)$ for $i = 1, 2, \ldots, t$, and

$$Q_j^n = \{y \in H_2 : q_j(Ax_n) \leq \langle \zeta_j^n, Ax_n - y\rangle\}, \tag{3.2}$$

where $\zeta_j^n \in \partial q_j(Ax_n)$ for $j = 1, 2, \ldots, r$. We can easily see that C_i^n $(i = 1, 2, \ldots, t)$ and Q_j^n $(j = 1, 2, \ldots, r)$ are half-spaces. It is easy to see that, for all $n \geq 1$, $C_i^n \supset C_i$ $(i = 1, 2, \ldots, t)$ and $Q_j^n \supset Q_j$ $(j = 1, 2, \ldots, r)$. We define

$$f_n(x) = \frac{1}{2}\sum_{i=1}^{t} l_i \|x - P_{C_i^n}(x)\|^2 + \frac{1}{2}\sum_{j=1}^{r} \lambda_j \|Ax - P_{Q_j^n}(Ax)\|^2, \tag{3.3}$$

where C_i^n $(i = 1, 2, \ldots, t)$ and Q_j^n $(j = 1, 2, \ldots, r)$ are respectively given by (3.1) and (3.2). Then we know

$$\nabla f_n(x) = \sum_{i=1}^{t} l_i\big(x - P_{C_i^n}(x)\big) + \sum_{j=1}^{r} \lambda_j A^*(I - P_{Q_j^n})Ax, \tag{3.4}$$

where A^* denotes the adjoint operator of A. And l_i $(i = 1, 2, \ldots, t)$ and λ_j $(j = 1, 2, \ldots, r)$ are positive constants such that $\sum_{i=1}^{t} l_i + \sum_{j=1}^{r} \lambda_j = 1$.

Now, we propose an algorithm for solving the multiple-sets split feasibility problem (1.1), where C_i $(i = 1, 2, \ldots, t)$ and Q_j $(j = 1, 2, \ldots, r)$ are as shown in (1.6).

Algorithm 3.1 (The inertial relaxed algorithm with Armijo-type line search) *Step* 1: *Given $\gamma > 0$, $l \in (0,1)$, $\mu \in (0,1)$, select the parameter β_n such that*

$$0 \leq \beta_n < \frac{1-\mu}{1+\mu}. \tag{3.5}$$

Select starting points $x_0, x_1 \in H_1$ and set $n = 1$.

Step 2: *For the iterations* x_n, x_{n-1}, *calculate*

$$y_n = \begin{cases} x_n & n = \text{ even}, \\ x_n + \beta_n(x_n - x_{n-1}) & n = \text{ odd}. \end{cases} \tag{3.6}$$

Step 3: *Calculate*

$$z_n = P_\Omega\bigl(y_n - \tau_n \nabla f_n(y_n)\bigr), \tag{3.7}$$

where $\tau_n = \gamma l^{m_n}$ *and* m_n *is the smallest nonnegative integer such that*

$$\tau_n \bigl\| \nabla f_n(y_n) - \nabla f_n(z_n) \bigr\| \le \mu \|y_n - z_n\|.$$

Step 4: *Compute the new iterate point*

$$x_{n+1} = P_\Omega\bigl(y_n - \tau_n \nabla f_n(z_n)\bigr). \tag{3.8}$$

Step 5: *Set* $n \leftarrow n+1$, *and go to Step* 2.

In the following, we prove the convergence of Algorithm 3.1.

Lemma 3.1 *Suppose that the solution set of MSSFP is nonempty, that is,* $S \neq \emptyset$ *and* $\{x_n\}$ *is any sequence generated by Algorithm* 3.1. *Then* $\{x_{2n}\}$ *is Fejer monotone with respect to* S (*i.e.*, $\|x_{2n+2} - z\| \le \|x_{2n} - z\|, \forall z \in S$).

Proof Choose a point z in S. We have

$$\begin{aligned}
&\|x_{2n+2} - z\|^2 \\
&\quad= \bigl\|P_\Omega\bigl(y_{2n+1} - \tau_{2n+1}\nabla f_{2n+1}(z_{2n+1})\bigr) - z\bigr\|^2 \\
&\quad\le \bigl\|(y_{2n+1} - z) - \tau_{2n+1}\nabla f_{2n+1}(z_{2n+1})\bigr\|^2 \\
&\qquad- \bigl\|x_{2n+2} - y_{2n+1} + \tau_{2n+1}\nabla f_{2n+1}(z_{2n+1})\bigr\|^2 \\
&\quad= \|y_{2n+1} - z\|^2 - 2\tau_{2n+1}\bigl\langle \nabla f_{2n+1}(z_{2n+1}), y_{2n+1} - z\bigr\rangle - \|x_{2n+2} - y_{2n+1}\|^2 \\
&\qquad- 2\tau_{2n+1}\bigl\langle \nabla f_{2n+1}(z_{2n+1}), x_{2n+2} - y_{2n+1}\bigr\rangle \\
&\quad= \|y_{2n+1} - z\|^2 - 2\tau_{2n+1}\bigl\langle \nabla f_{2n+1}(z_{2n+1}), z_{2n+1} - z\bigr\rangle \\
&\qquad- 2\tau_{2n+1}\bigl\langle \nabla f_{2n+1}(z_{2n+1}), y_{2n+1} - z_{2n+1}\bigr\rangle - \|x_{2n+2} - y_{2n+1}\|^2 \\
&\qquad- 2\tau_{2n+1}\bigl\langle \nabla f_{2n+1}(z_{2n+1}), x_{2n+2} - y_{2n+1}\bigr\rangle \\
&\quad= \|y_{2n+1} - z\|^2 - 2\tau_{2n+1}\bigl\langle \nabla f_{2n+1}(z_{2n+1}), z_{2n+1} - z\bigr\rangle \\
&\qquad- 2\tau_{2n+1}\bigl\langle \nabla f_{2n+1}(z_{2n+1}), x_{2n+2} - z_{2n+1}\bigr\rangle \\
&\qquad- \|x_{2n+2} - z_{2n+1} + z_{2n+1} - y_{2n+1}\|^2 \\
&\quad= \|y_{2n+1} - z\|^2 - 2\tau_{2n+1}\bigl\langle \nabla f_{2n+1}(z_{2n+1}), z_{2n+1} - z\bigr\rangle \\
&\qquad- 2\tau_{2n+1}\bigl\langle \nabla f_{2n+1}(z_{2n+1}), x_{2n+2} - z_{2n+1}\bigr\rangle - \|x_{2n+2} - z_{2n+1}\|^2
\end{aligned} \tag{3.9}$$

$$
\begin{aligned}
&\quad - 2\langle x_{2n+2} - z_{2n+1}, z_{2n+1} - y_{2n+1}\rangle - \|z_{2n+1} - y_{2n+1}\|^2 \\
&= \|y_{2n+1} - z\|^2 - \|x_{2n+2} - z_{2n+1}\|^2 - \|z_{2n+1} - y_{2n+1}\|^2 \\
&\quad - 2\big\langle z_{2n+1} - y_{2n+1} + \tau_{2n+1}\nabla f_{2n+1}(z_{2n+1}), x_{2n+2} - z_{2n+1}\big\rangle \\
&\quad - 2\tau_{2n+1}\big\langle \nabla f_{2n+1}(z_{2n+1}), z_{2n+1} - z\big\rangle.
\end{aligned}
$$

Due to the fact that $z_{2n+1} \in \Omega$, we have

$$
\begin{aligned}
&\big\langle z_{2n+1} - y_{2n+1} + \tau_{2n+1}\nabla f_{2n+1}(y_{2n+1}), x_{2n+2} - z_{2n+1}\big\rangle \\
&\quad = \big\langle P_\Omega\big(y_{2n+1} - \tau_{2n+1}\nabla f_{2n+1}(y_{2n+1})\big) - y_{2n+1} + \tau_{2n+1}\nabla f_{2n+1}(y_{2n+1}), \\
&\qquad x_{2n+2} - P_\Omega\big(y_{2n+1} - \tau_{2n+1}\nabla f_{2n+1}(y_{2n+1})\big)\big\rangle \geq 0.
\end{aligned}
\tag{3.10}
$$

As a result,

$$
\begin{aligned}
&-2\big\langle z_{2n+1} - y_{2n+1} + \tau_{2n+1}\nabla f_{2n+1}(z_{2n+1}), x_{2n+2} - z_{2n+1}\big\rangle \\
&\quad \leq 2\big\langle y_{2n+1} - z_{2n+1} - \tau_{2n+1}\nabla f_{2n+1}(z_{2n+1}), x_{2n+2} - z_{2n+1}\big\rangle \\
&\qquad + 2\big\langle z_{2n+1} - y_{2n+1} + \tau_{2n+1}\nabla f_{2n+1}(y_{2n+1}), x_{2n+2} - z_{2n+1}\big\rangle \\
&\quad = 2\big\langle \tau_{2n+1}\nabla f_{2n+1}(y_{2n+1}) - \tau_{2n+1}\nabla f_{2n+1}(z_{2n+1}), x_{2n+2} - z_{2n+1}\big\rangle \\
&\quad \leq 2\tau_{2n+1}\big\|\nabla f_{2n+1}(y_{2n+1}) - \nabla f_{2n+1}(z_{2n+1})\big\| \|x_{2n+2} - z_{2n+1}\| \\
&\quad \leq \tau_{2n+1}^2\big\|\nabla f_{2n+1}(y_{2n+1}) - \nabla f_{2n+1}(z_{2n+1})\big\|^2 + \|x_{2n+2} - z_{2n+1}\|^2 \\
&\quad \leq \mu^2\|y_{2n+1} - z_{2n+1}\|^2 + \|x_{2n+2} - z_{2n+1}\|^2.
\end{aligned}
\tag{3.11}
$$

As $I - P_{C_i^{2n+1}}$ and $I - P_{Q_j^{2n+1}}$ are firmly-nonexpansive and $\nabla f_{2n+1}(z_{2n+1}) = 0$, then

$$
\begin{aligned}
&2\tau_{2n+1}\big\langle \nabla f_{2n+1}(z_{2n+1}), z_{2n+1} - z\big\rangle \\
&\quad = 2\tau_{2n+1}\big\langle \nabla f_{2n+1}(z_{2n+1}) - \nabla f_{2n+1}(z), z_{2n+1} - z\big\rangle \\
&\quad = 2\tau_{2n+1}\Bigg\langle \sum_{i=1}^t l_i(I - P_{C_i^{2n+1}})z_{2n+1} + \sum_{j=1}^r \lambda_j A^*(I - P_{Q_j^{2n+1}})Az_{2n+1} \\
&\qquad - \sum_{i=1}^t l_i(I - P_{C_i^{2n+1}})z - \sum_{j=1}^r \lambda_j A^*(I - P_{Q_j^{2n+1}})Az, z_{2n+1} - z\Bigg\rangle \\
&\quad = 2\tau_{2n+1}\Bigg[\Bigg\langle \sum_{i=1}^t l_i(I - P_{C_i^{2n+1}})z_{2n+1} - \sum_{i=1}^t l_i(I - P_{C_i^{2n+1}})z, z_{2n+1} - z\Bigg\rangle \\
&\qquad + \Bigg\langle \sum_{j=1}^r \lambda_j(I - P_{Q_j^{2n+1}})Az_{2n+1} - \sum_{j=1}^r \lambda_j(I - P_{Q_j^{2n+1}})Az, Az_{2n+1} - Az\Bigg\rangle\Bigg] \\
&\quad \geq 2\tau_{2n+1}\Bigg[\sum_{i=1}^t l_i\big\|(I - P_{C_i^{2n+1}})z_{2n+1} - (I - P_{C_i^{2n+1}})z\big\|^2 \\
&\qquad + \sum_{j=1}^r \lambda_j\big\|(I - P_{Q_j^{2n+1}})Az_{2n+1} - (I - P_{Q_j^{2n+1}})Az\big\|^2\Bigg]
\end{aligned}
\tag{3.12}
$$

$$\geq \frac{2\mu l}{\|A\|^2}\left[\sum_{i=1}^{t} l_i \left\|(I-P_{C_i^{2n+1}})z_{2n+1}\right\|^2 + \sum_{j=1}^{r} \lambda_j \left\|(I-P_{Q_j^{2n+1}})Az_{2n+1}\right\|^2\right].$$

Putting (3.10), (3.11), (3.12) into (3.9), one has

$$\begin{aligned}
&\|x_{2n+2}-z\|^2 \\
&\quad\leq \|y_{2n+1}-z\|^2 - \left(1-\mu^2\right)\|y_{2n+1}-z_{2n+1}\|^2 \\
&\qquad - \frac{2\mu l}{\|A\|^2}\left[\sum_{i=1}^{t} l_i \left\|(I-P_{C_i^{2n+1}})z_{2n+1}\right\|^2 + \sum_{j=1}^{r} \lambda_j \left\|(I-P_{Q_j^{2n+1}})Az_{2n+1}\right\|^2\right].
\end{aligned} \tag{3.13}$$

Similar to the discussion of (3.13), we can know

$$\begin{aligned}
&\|x_{2n+1}-z\|^2 \\
&\quad\leq \|y_{2n}-z\|^2 - \left(1-\mu^2\right)\|y_{2n}-z_{2n}\|^2 \\
&\qquad - \frac{2\mu l}{\|A\|^2}\left[\sum_{i=1}^{t} l_i \left\|(I-P_{C_i^{2n}})z_{2n}\right\|^2 + \sum_{j=1}^{r} \lambda_j \left\|(I-P_{Q_j^{2n}})Az_{2n}\right\|^2\right] \\
&\quad= \|x_{2n}-z\|^2 - \left(1-\mu^2\right)\|y_{2n}-z_{2n}\|^2 \\
&\qquad - \frac{2\mu l}{\|A\|^2}\left[\sum_{i=1}^{t} l_i \left\|(I-P_{C_i^{2n}})z_{2n}\right\|^2 + \sum_{j=1}^{r} \lambda_j \left\|(I-P_{Q_j^{2n}})Az_{2n}\right\|^2\right].
\end{aligned} \tag{3.14}$$

According to (3.6), we obtain

$$\begin{aligned}
&\|y_{2n+1}-z\|^2 \\
&\quad= \left\|x_{2n+1}+\beta_{2n+1}(x_{2n+1}-x_{2n})-z\right\|^2 \\
&\quad= \left\|x_{2n+1}-\beta_{2n+1}z+\beta_{2n+1}z+\beta_{2n+1}(x_{2n+1}-x_{2n})-z\right\|^2 \\
&\quad= \left\|(1+\beta_{2n+1})(x_{2n+1}-z)-\beta_{2n+1}(x_{2n}-z)\right\|^2 \\
&\quad= (1+\beta_{2n+1})\|x_{2n+1}-z\|^2 - \beta_{2n+1}\|x_{2n}-z\|^2 \\
&\qquad + \beta_{2n+1}(1+\beta_{2n+1})\|x_{2n+1}-x_{2n}\|^2.
\end{aligned} \tag{3.15}$$

Substituting (3.14) and (3.15) into (3.13), one has

$$\begin{aligned}
&\|x_{2n+2}-z\|^2 \\
&\quad\leq \|x_{2n}-z\|^2 - (1+\beta_{2n+1})\left(1-\mu^2\right)\|y_{2n}-z_{2n}\|^2 \\
&\qquad - (1+\beta_{2n+1})\frac{2\mu l}{\|A\|^2}\left[\sum_{i=1}^{t} l_i \left\|(I-P_{C_i^{2n}})z_{2n}\right\|^2 + \sum_{j=1}^{r} \lambda_j \left\|(I-P_{Q_j^{2n}})Az_{2n}\right\|^2\right] \\
&\qquad - \left(1-\mu^2\right)\|y_{2n+1}-z_{2n+1}\|^2 + \beta_{2n+1}(1+\beta_{2n+1})\|x_{2n+1}-x_{2n}\|^2 \\
&\qquad - \frac{2\mu l}{\|A\|^2}\left[\sum_{i=1}^{t} l_i \left\|(I-P_{C_i^{2n+1}})z_{2n+1}\right\|^2 + \sum_{j=1}^{r} \lambda_j \left\|(I-P_{Q_j^{2n+1}})Az_{2n+1}\right\|^2\right].
\end{aligned} \tag{3.16}$$

Note that

$$\begin{aligned}
&\|x_{2n+1}-x_{2n}\| \\
&\quad\le \|x_{2n+1}-z_{2n}\|+\|z_{2n}-x_{2n}\| \\
&\quad= \bigl\|P_\Omega\bigl(y_{2n}-\tau_{2n}\nabla f_{2n}(z_{2n})\bigr)-z_{2n}\bigr\|+\|x_{2n}-z_{2n}\| \\
&\quad\le \bigl\|y_{2n}-\tau_{2n}\nabla f_{2n}(z_{2n})-y_{2n}+\tau_{2n}\nabla f_{2n}(y_{2n})\bigr\|+\|x_{2n}-z_{2n}\| \\
&\quad= \tau_{2n}\bigl\|\nabla f_{2n}(y_{2n})-\nabla f_{2n}(z_{2n})\bigr\|+\|x_{2n}-z_{2n}\| \\
&\quad\le (1+\mu)\|x_{2n}-z_{2n}\|.
\end{aligned}\tag{3.17}$$

Combining (3.16) and (3.17), we have

$$\begin{aligned}
&\|x_{2n+2}-z\|^2 \\
&\quad\le \|x_{2n}-z\|^2-\bigl(1-\mu^2\bigr)\|y_{2n+1}-z_{2n+1}\|^2 \\
&\qquad-(1+\beta_{2n+1})\frac{2\mu l}{\|A\|^2}\Biggl[\sum_{i=1}^{t}l_i\bigl\|(I-P_{C_i^{2n}})z_{2n}\bigr\|^2+\sum_{j=1}^{r}\lambda_j\bigl\|(I-P_{Q_j^{2n}})Az_{2n}\bigr\|^2\Biggr] \\
&\qquad-\bigl[(1+\beta_{2n+1})\bigl(1-\mu^2\bigr)-\beta_{2n+1}(1+\beta_{2n+1})(1+\mu)^2\bigr]\|y_{2n}-z_{2n}\|^2 \\
&\qquad-\frac{2\mu l}{\|A\|^2}\Biggl[\sum_{i=1}^{t}l_i\bigl\|(I-P_{C_i^{2n+1}})z_{2n+1}\bigr\|^2+\sum_{j=1}^{r}\lambda_j\bigl\|(I-P_{Q_j^{2n+1}})Az_{2n+1}\bigr\|^2\Biggr] \\
&\quad\le \|x_{2n}-z\|^2,
\end{aligned}\tag{3.18}$$

where $\mu\in(0,1)$, $\beta_{2n+1}\in[0,\frac{1-\mu}{1+\mu}]$, so $(1-\mu^2)>0$, $(1+\beta_{2n+1})(1-\mu^2)-\beta_{2n+1}(1+\beta_{2n+1})(1+\mu)^2>0$, $1+\beta_{2n+1}>0$, $\frac{2\mu l}{\|A\|^2}>0$.

Hence,

$$\|x_{2n+2}-z\|\le\|x_{2n}-z\|. \qquad \square$$

Theorem 3.1 *Suppose that $S\neq\emptyset$ and $\{x_n\}$ is any sequence generated by Algorithm 3.1. Then $\{x_n\}$ converges weakly to a point in S.*

Proof According to Lemma 3.1, it is easy to know that $\lim_{n\to\infty}\|x_{2n+2}-z\|$ exists. This means that $\{x_{2n}\}$ is bounded. In addition, from (3.18), we conclude that

$$\lim_{n\to\infty}\Biggl[\sum_{i=1}^{t}l_i\bigl\|(I-P_{C_i^{2n}})z_{2n}\bigr\|+\sum_{j=1}^{r}\lambda_j\bigl\|(I-P_{Q_j^{2n}})Az_{2n}\bigr\|\Biggr]=0.$$

That is to say,

$$\lim_{n\to\infty}\bigl\|(I-P_{C_i^{2n}})z_{2n}\bigr\|=0\quad(i=1,2,\ldots,t)\tag{3.19}$$

and

$$\lim_{n\to\infty}\bigl\|(I-P_{Q_j^{2n}})Az_{2n}\bigr\|=0\quad(j=1,2,\ldots,r).\tag{3.20}$$

Besides,

$$\lim_{n\to\infty}\|x_{2n}-z_{2n}\|=0. \tag{3.21}$$

Owing to $I-P_{C_i^{2n}}$ and $I-P_{Q_j^{2n}}$ are nonexpansive, then

$$\begin{aligned}
&\bigl\|(I-P_{C_i^{2n}})x_{2n}\bigr\| \\
&\quad\le \bigl\|(I-P_{C_i^{2n}})x_{2n}-(I-P_{C_i^{2n}})z_{2n}\bigr\|+\bigl\|(I-P_{C_i^{2n}})z_{2n}\bigr\| \\
&\quad\le \|x_{2n}-z_{2n}\|+\bigl\|(I-P_{C_i^{2n}})z_{2n}\bigr\| \quad (i=1,2,\dots,t)
\end{aligned} \tag{3.22}$$

and

$$\begin{aligned}
&\bigl\|(I-P_{Q_j^{2n}})Ax_{2n}\bigr\| \\
&\quad\le \bigl\|(I-P_{Q_j^{2n}})Ax_{2n}-(I-P_{Q_j^{2n}})Az_{2n}\bigr\|+\bigl\|(I-P_{Q_j^{2n}})Az_{2n}\bigr\| \\
&\quad\le \|Ax_{2n}-Az_{2n}\|+\bigl\|(I-P_{Q_j^{2n}})Az_{2n}\bigr\| \\
&\quad\le \|A\|\|x_{2n}-z_{2n}\|+\bigl\|(I-P_{Q_j^{2n}})Az_{2n}\bigr\| \quad (j=1,2,\dots,r).
\end{aligned} \tag{3.23}$$

According to (3.19) and (3.21), from (3.22), we conclude that

$$\lim_{n\to\infty}\bigl\|(I-P_{C_i^{2n}})x_{2n}\bigr\|=0 \quad (i=1,2,\dots,t). \tag{3.24}$$

According to (3.20) and (3.21), from (3.23), we conclude that

$$\lim_{n\to\infty}\bigl\|(I-P_{Q_j^{2n}})Ax_{2n}\bigr\|=0 \quad (j=1,2,\dots,r). \tag{3.25}$$

Similar to the discussion in (3.24) and (3.25), we know

$$\begin{aligned}
&\lim_{n\to\infty}\bigl\|(I-P_{C_i^{2n+1}})x_{2n+1}\bigr\|=0 \quad (i=1,2,\dots,t), \\
&\lim_{n\to\infty}\bigl\|(I-P_{Q_j^{2n+1}})Ax_{2n+1}\bigr\|=0 \quad (j=1,2,\dots,r).
\end{aligned} \tag{3.26}$$

As ∂c_i for $i=1,2,\dots,t$ are bounded on bounded sets, we have a constant $\xi>0$ such that $\|\xi_i^{2n}\|\le\xi$ $(i=1,2,\dots,t)$. On account of $P_{C_i^{2n}}x_{2n}\in C_i^{2n}$, we obtain from the algorithm and (3.24) that

$$\begin{aligned}
c_i(x_{2n}) &\le \bigl\langle \xi_i^{2n}, y_{2n}-P_{C_i^{2n}}x_{2n}\bigr\rangle=\bigl\langle \xi_i^{2n},(I-P_{C_i^{2n}})x_{2n}\bigr\rangle \\
&\le \xi\bigl\|(I-P_{C_i^{2n}})x_{2n}\bigr\|\longrightarrow 0, \quad n\longrightarrow+\infty.
\end{aligned} \tag{3.27}$$

As $\{x_{2n}\}$ is bounded, there exists a weakly convergent subsequence $\{x_{2n_k}\}\subset\{x_{2n}\}, k\in N$, such that $x_{2n_k}\rightharpoonup x^*, x^*\in H_1$. According to c_i $(i=1,2,\dots,t)$ being continuous and (3.27), we have

$$c_i\bigl(x^*\bigr)\le\liminf_{k\to\infty}c_i(x_{2n_k})\le 0, \quad i=1,2,\dots,t. \tag{3.28}$$

So $x^*\in C_i$ for $i=1,2,\dots,t$.

As ∂q_j for $j = 1, 2, \ldots, r$ are bounded on bounded sets, we have a constant $\zeta > 0$ such that $\|\zeta_j^{2n}\| \leq \zeta$ $(j = 1, 2, \ldots, r)$. On account of $P_{Q_j^{2n}} x_{2n} \in Q_j^{2n}$, we obtain from the algorithm and (3.25) that

$$\begin{aligned} q_j(Ax_{2n}) &\leq \langle \zeta_j^{2n}, Ay_{2n} - P_{Q_j^{2n}} Ax_{2n} \rangle = \langle \zeta_j^{2n}, (I - P_{Q_j^{2n}}) Ax_{2n} \rangle \\ &\leq \zeta \| (I - P_{Q_j^{2n}}) Ax_{2n} \| \longrightarrow 0, \quad n \longrightarrow +\infty. \end{aligned} \tag{3.29}$$

According to q_j for $j = 1, 2, \ldots, r$ are continuous and (3.29), we have

$$q_j(Ax^*) \leq \liminf_{k \to \infty} q_j(Ax_{2n_k}) \leq 0, \quad j = 1, 2, \ldots, r. \tag{3.30}$$

So $Ax^* \in Q_j$ for $j = 1, 2, \ldots, r$.

Thus, $x^* \in S$.

Now, we are going to prove that $\{x_{2n+1}\}$ converges to x^*. Just for the sake of convenience, we are still going to use $\{x_{2n+1}\}$ for the proof. According to $\lim_{n\to\infty} \|x_{2n} - x^*\|$ exists and $\lim_{n\to\infty} \|x_{2n_k} - x^*\| = 0$, these mean that $\lim_{n\to\infty} \|x_{2n} - x^*\| = 0$. Thus, x^* is sole.

Using the same discussion in (3.9)–(3.13), we can know that

$$\begin{aligned} &\|x_{2n+1} - x^*\|^2 \\ &\quad= \|P_\Omega(y_{2n} - \tau_{2n} \nabla f_{2n}(z_{2n})) - x^*\|^2 \\ &\quad\leq \|y_{2n} - x^*\|^2 - \|x_{2n+1} - z_{2n}\|^2 - \|z_{2n} - y_{2n}\|^2 \\ &\qquad - 2\langle z_{2n} - y_{2n} + \tau_{2n} \nabla f_{2n}(z_{2n}), x_{2n+1} - z_{2n} \rangle \\ &\qquad - 2\tau_{2n} \langle \nabla f_{2n}(z_{2n}), z_{2n} - x^* \rangle \\ &\quad\leq \|y_{2n} - x^*\|^2 - (1 - \mu^2) \|y_{2n} - z_{2n}\|^2 \\ &\qquad - \frac{2\mu l}{\|A\|^2} \Biggl[\sum_{i=1}^t l_i \| (I - P_{C_i^{2n}}) z_{2n} \|^2 + \sum_{j=1}^r \lambda_j \| (I - P_{Q_j^{2n}}) A z_{2n} \|^2 \Biggr] \\ &\quad\leq \|y_{2n} - x^*\|^2. \end{aligned} \tag{3.31}$$

Thus,

$$\|x_{2n+1} - x^*\|^2 \leq \|y_{2n} - x^*\|^2 = \|x_{2n} - x^*\|^2. \tag{3.32}$$

Consequently,

$$\lim_{n\to\infty} \|x_{2n+1} - x^*\| = 0. \tag{3.33}$$

To sum up,

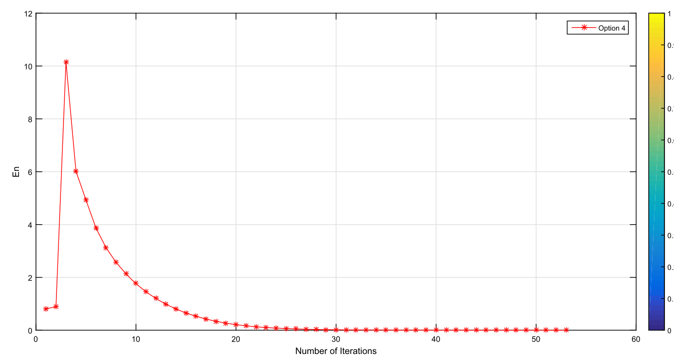
$$\lim_{n\to\infty} x_n = x^*. \qquad \square$$

4 Numerical examples

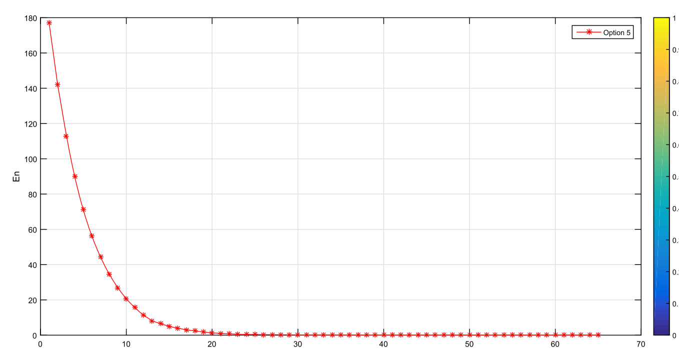
As in Example 4.1, we will provide the results in this section. The whole codes are written in Matlab R2012a. All the numerical results are carried out on a personal Lenovo Thinkpad

Table 1 Algorithm 3.1 in this paper under diverse options of x_0 and x_1

	No. of Iter.	Time		No. of Iter.	Time(s)
Option 1	2	0.005017	Option 5	65	0.12809
Option 2	62	0.162404	Option 6	2	0.003902
Option 3	196	0.427017	Option 7	41	0.074836
Option 4	53	0.095574	Option 8	40	0.052645



(a) Option 4 of Error history in Example 4.1



(b) Option 5 of Error history in Example 4.1

Figure 1 Error history in Example 4.1

computer with Intel(R) Core(TM) i7-3517U CPU 2.40 GHz and RAM 8.00GB. Firstly, we are going to come up with some different x_0, x_1 in our Algorithm 3.1. These results are provided in Table 1 and Fig. 1. Secondly, we contrast Algorithm 3.1 in this paper and Algorithm 3.1 in [23]. From the numerical results of Example 1 in [23], it is better than the results in [13]. So our algorithm is compared to Algorithm 3.1 in [23]. These results are provided in Table 2. Lastly, we check the stability of the iteration number for Algorithm 3.1 in this paper comparing with Algorithm 3.1 in [23]. These results are provided in Figs. 2–4.

Table 2 Comparison of Algorithm 3.1 in this paper and Algorithm 3.1 in [23]

			Alg 3.1 in this paper	Alg 3.1 in [23]
Option 1	$x_0 = (-4,-2,3)^T$	No. of Iter.	2	21
	$x_1 = (-1,2,0)^T$	cpu(Time)	0.004028	0.004364
Option 4	$x_0 = (1,-6,-4)^T$	No. of Iter.	53	248
	$x_1 = (-1,2,0)^T$	cpu(Time)	0.030029	0.049955
Option 5	$x_0 = (-4,-2,-3)^T$	No. of Iter.	65	122
	$x_1 = (-5,-2,-3)^T$	cpu(Time)	0.053327	0.023334
Option 6	$x_0 = (-5.34,-7.36,-3.21)^T$	No. of Iter.	2	29
	$x_1 = (-1.23,-2.13,-3.56)^T$	cpu(Time)	0.003902	0.005509

Example 4.1 ([13]) Suppose that $H_1 = H_2 = R^3$, $r = t = 2$, and $l_1 = l_2 = \lambda_1 = \lambda_2 = \frac{1}{4}$. We give that

$$\begin{aligned} C_1 &= \bigl\{x = (a,b,c)^T \in R^3 : a + b^2 + 2c \leq 0\bigr\}, \\ C_2 &= \left\{x = (a,b,c)^T \in R^3 : \frac{a^2}{16} + \frac{b^2}{9} + \frac{c^2}{4} - 1 \leq 0\right\}, \\ Q_1 &= \bigl\{x = (a,b,c)^T \in R^3 : a^2 + b - c \leq 0\bigr\}, \\ Q_2 &= \left\{x = (a,b,c)^T \in R^3 : \frac{a^2}{4} + \frac{b^2}{4} + \frac{c^2}{9} - 1 \leq 0\right\}, \end{aligned} \tag{4.1}$$

and

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

To find $x^* \in C_1 \cap C_2$ such that $Ax^* \in Q_1 \cap Q_2$.

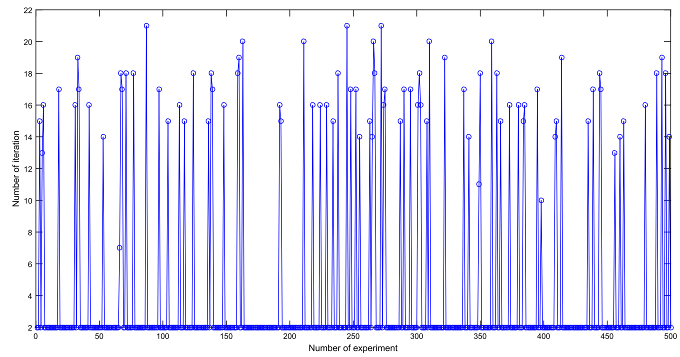
In the first place, let $\gamma = 2$, $l = 0.5$, $\mu = 0.95$, and $\beta_n = \frac{1}{n+1}$. Next, we study the iteration number required for the convergence of the sequence under different initial values. The condition for stopping the iteration is

$$E_n = \frac{1}{2}\sum_{i=1}^{2}\bigl\|x_n - P_{C_i^n}(x_n)\bigr\|^2 + \frac{1}{2}\sum_{j=1}^{2}\bigl\|Ax_n - P_{Q_j^n}(Ax_n)\bigr\|^2 < 10^{-4}. \tag{4.2}$$

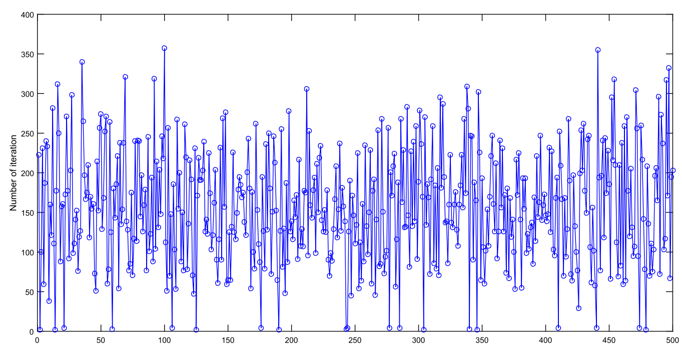
We select diverse options of x_0 and x_1 as follows.

Option 1: $x_0 = (1,1,5)^T$ and $x_1 = (5,-3,2)^T$;
Option 2: $x_0 = (-4,3,-2)^T$ and $x_1 = (-5,2,1)^T$;
Option 3: $x_0 = (7,5,1)^T$ and $x_1 = (7,-3,-1)^T$;
Option 4: $x_0 = (1,-6,-4)^T$ and $x_1 = (-4,1,6)^T$;
Option 5: $x_0 = (-4,-2,-3)^T$ and $x_1 = (-5,-2,-3)^T$;
Option 6: $x_0 = (-5.34,-7.36,-3.21)^T$ and $x_1 = (0.23,-2.13,3.56)^T$;
Option 7: $x_0 = (-2.345,2.431,1.573)^T$ and $x_1 = (1.235,-1.756,-4.234)^T$;
Option 8: $x_0 = (5.32,2.33,7.75)^T$ and $x_1 = (3.23,3.75,-3.86)^T$.

From Table 1, we can see the iteration number and running time of Algorithm 3.1 in this paper for diverse options of x_0 and x_1.



(a) The iteration number of Algorithm 3.1 in this paper



(b) The iteration number of Algorithm 3.1 in [22]

Figure 2 The iteration number of Algorithm 3.1 in this paper and Algorithm 3.1 in [23]

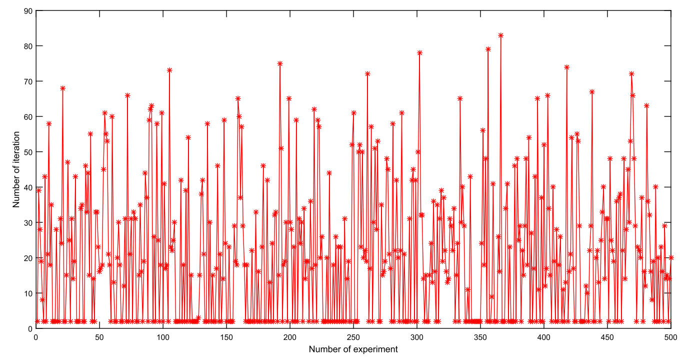
In Algorithm 3.1, if we choose x_0 and x_1 as Option 4 and Option 5, the compartment of the error E_n is gradually converging, for details, please see Fig. 1. For other options, they are also gradually converging, which we do not show here.

Now, we compare Algorithm 3.1 in this paper and Algorithm 3.1 in [23]. The results are as shown in Table 2. Furthermore, in order to test the stability of the iteration number, 500 diverse initial value points are randomly selected for the experiment in the context of Algorithm 3.1 in this paper, for instance,

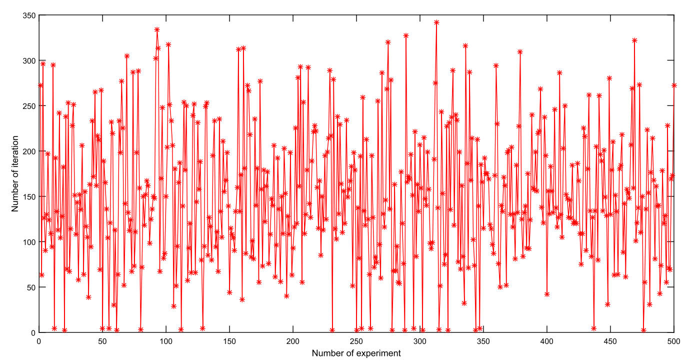
$$x_0 = \mathrm{rand}(3,1), \qquad x_1 = \mathrm{rand}(3,1) * 10,$$
$$x_0 = \mathrm{rand}(3,1), \qquad x_1 = \mathrm{rand}(3,1) * 50,$$
$$x_0 = \mathrm{rand}(3,1), \qquad x_1 = \mathrm{rand}(3,1) * 100,$$

the consequences are separately shown in Fig. 2(a), Fig. 3(a), and Fig. 4(a).

In the same way, we also offer 500 experiments for diverse initial value points which are randomly selected in the context of Algorithm 3.1 in [23]. For $\forall n \in N$, let $\alpha_n = \frac{1}{n+1}$,



(a) The iteration number of Algorithm 3.1 in this paper



(b) The iteration number of Algorithm 3.1 in [22]

Figure 3 The iteration number of Algorithm 3.1 in this paper and Algorithm 3.1 in [23]

$\rho_n = 3.95$, and $\omega_n = \frac{1}{1+n^{1.2}}$. Suppose $\beta = 0.5$ and $\beta_n = \beta$, for instance,

$$u = \text{rand}(3,1), \qquad x_0 = \text{rand}(3,1), \qquad x_1 = \text{rand}(3,1) * 10,$$

$$u = \text{rand}(3,1), \qquad x_0 = \text{rand}(3,1), \qquad x_1 = \text{rand}(3,1) * 50,$$

$$u = \text{rand}(3,1), \qquad x_0 = \text{rand}(3,1), \qquad x_1 = \text{rand}(3,1) * 100,$$

the consequences are separately shown in Fig. 2(b), Fig. 3(b), and Fig. 4(b).

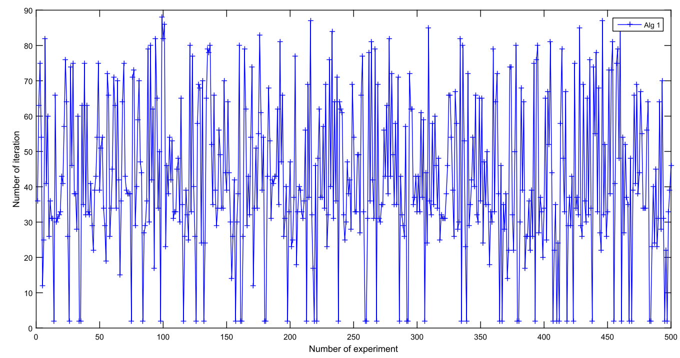
From Tables 1–2 and Figs. 1–4, we can obtain the following conclusions.

1. Algorithm 3.1 in this paper is efficient for some different options and has a nice convergence speed and lower iteration number.

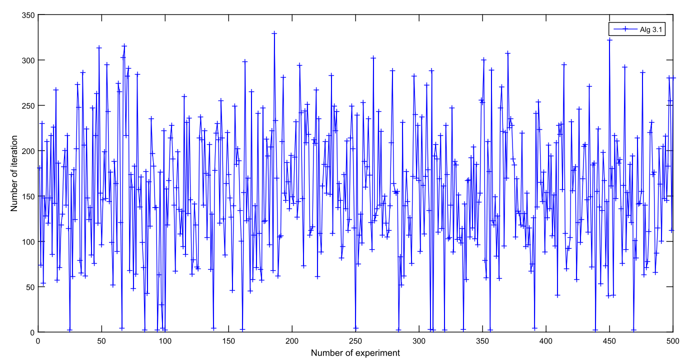
2. As you can see, for every option of x_0 and x_1, there is no important difference in CPU running times or iteration number. Therefore, our preliminary speculation is that the different options of x_0 and x_1 have negligible influence on the convergence of this algorithm.

3. With regard to Table 2, for some different options of x_0 and x_1, our Algorithm 3.1 clearly outperforms Algorithm 3.1 in [23].

4. According to Figs. 2–4, we conclude that the iteration number of Algorithm 3.1 in this paper is stable. Moreover, we can see the iteration number of Algorithm 3.1 in this paper is lower than that of Algorithm 3.1 in [23]. For example, in Fig. 4, the iteration number of



(a) The iteration number of Algorithm 3.1 in this paper



(b) The iteration number of Algorithm 3.1 in [22]

Figure 4 The iteration number of Algorithm 3.1 in this paper and Algorithm 3.1 in [23]

Algorithm 3.1 in this paper is basically stable at about 50. However, Algorithm 3.1 in [23] is basically stable at about 150.

5 Conclusions

In this paper, we propose the inertial relaxed CQ algorithm for solving the convex multiple-sets split feasibility problem. And the global convergence conclusions are obtained. Our consequences generalize and produce some existing associated outcomes. Moreover, the preliminary numerical conclusions reveal that our presented algorithm is superior to some existing relaxed CQ algorithms in some cases about solving the convex multiple-sets split feasibility problem.

Acknowledgements
We thank the anonymous referees and the editor for their constructive comments and suggestions, which have improved this manuscript.

Funding
This manuscript is supported by the Natural Science Foundation of China (Grant No. 11401438) and Shandong Provincial Natural Science Foundation, China (Grant No. ZR2020MA027, ZR2019MA022).

Availability of data and materials
All data generated or analysed during this study are included in this manuscript.

Declarations

Competing interests
The authors declare that they have no competing interests regarding the present manuscript.

Authors' contributions
All authors contributed equally to the writing of this paper. All authors read and approved the final manuscript.

Author details
[1]College of Mathematics and Systems Science, Shandong University of Science and Technology, 266590, Qingdao, Shandong, P.R. China. [2]School of Mathematics and Information Science, Weifang University, 261061, Weifang, Shandong, P.R. China.

Publisher's Note
Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 8 July 2021 Accepted: 23 November 2021 Published online: 07 December 2021

References
1. Abass, H.A., Aremu, K.O., Jolaoso, L.O., Mewomo, O.T.: An inertial forward-backward splitting method for approximating solutions of certain optimization problems. J. Nonlinear Funct. Anal. **2020**, Article ID 6 (2020)
2. Bot, R.I., Csetnek, E.R.: An inertial alternating direction method of multipliers. Minimax Theory Appl. **1**, 29–49 (2016)
3. Bot, R.I., Csetnek, E.R.: An inertial forward-backward-forward primal-dual splitting algorithm for solving monotone inclusion problems. Numer. Algorithms **71**, 519–540 (2016)
4. Bot, R.I., Csetnek, E.R., Hendrich, C.: Inertial Douglas–Rachford splitting for monotone inclusion. Appl. Math. Comput. **256**, 472–487 (2015)
5. Censor, Y., Elfving, T.: A multiprojection algorithm using Bregman projection in product space. Numer. Algorithms **8**, 221–239 (1994)
6. Censor, Y., Elfving, T., Kopf, N., Bortfeld, T.: The multiple-sets split feasibility problem and its applications for inverse problem. Inverse Probl. **21**, 2071–2084 (2005)
7. Chuang, C.S.: Hybrid inertial proximal algorithm for the split variational inclusion problem in Hilbert spaces with applications. Optimization **66**, 777–792 (2017)
8. Dang, Y., Gao, Y.: The strong convergence of a KM-CQ-like algorithm for a split feasibility problem. Inverse Probl. **27**, 015007 (2011)
9. Dang, Y.Z., Sun, J., Xu, H.K.: Inertial accelerated algorithms for solving a split feasibility problem. J. Ind. Manag. Optim. **13**, 1383–1394 (2017)
10. Dong, Q.L., Yuan, H.B., Cho, Y.J., Rassias, T.: Modified inertial Mann algorithm and inertial CQ algorithm for nonexpansive mappings. Optim. Lett. **12**, 87–102 (2018)
11. Fukushima, M.: A relaxed projection method for variational inequalities. Math. Program. **35**, 58–70 (1986)
12. Guessab, A., Driouch, A., Nouisser, O.: A globally convergent modified version of the method of moving asymptotes. Appl. Anal. Discrete Math. **13**, 905–917 (2019)
13. He, S., Zhao, Z., Luo, B.: A relaxed self-adaptive CQ algorithm for the multiple-sets split feasibility problem. Optimization **64**, 1907–1918 (2015)
14. Lorenz, D.A., Pock, T.: An inertial forward-backward algorithm for monotone inclusions. J. Math. Imaging Vis. **51**, 311–325 (2015)
15. Maingé, P.E.: Inertial iterative process for fixed points of certain quasi-nonexpansive mappings. Set-Valued Anal. **15**, 67–79 (2007)
16. Mostafa, B., Thierry, E., Guessab, A.: A moving asymptotes algorithm using new local convex approximation methods with explicit solutions. Electron. Trans. Numer. Anal. **43**, 21–44 (2014)
17. Nesterov, Y.: A method for solving the convex programming problem with convergence rate O(1/k2). Dokl. Akad. Nauk SSSR **269**, 543–547 (1983)
18. Polyak, B.T.: Minimization of unsmooth functionals. USSR Comput. Math. Math. Phys. **9**, 14–29 (1969)
19. Shehu, Y., Gibali, A.: New inertial relaxed method for solving split feasibilities. Optim. Lett. **5**, 1–18 (2020)
20. Shehu, Y., Iyiola, O.S.: Convergence analysis for the proximal split feasibility problem using an inertial extrapolation term method. J. Fixed Point Theory Appl. **19**, 2483–2510 (2017)
21. Shehu, Y., Vuong, P.T., Cholamjiak, P.: 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**, 1–24 (2019)
22. Suantai, S., Pholasa, N., Cholamjiak, P.: The modified inertial relaxed CQ algorithm for solving the split feasibility problems. J. Ind. Manag. Optim. **14**, 1595–1615 (2018)
23. Suantai, S., Pholasa, N., Cholamjiak, P.: Relaxed CQ algorithms involving the inertial technique for multiple-sets split feasibility problems. Rev. R. Acad. Cienc. Exactas Fís. Nat., Ser. A Mat. **113**, 1081–1099 (2019)
24. Wang, F.: A new algorithm for solving the multiple-sets split feasibility problem in Banach spaces. Numer. Funct. Anal. Optim. **35**, 99–110 (2014)
25. Xu, H.K.: A variable Krasonosel'skii–Mann algorithm and the multiple-set split feasibility problem. Inverse Probl. **22**, 2021–2034 (2006)
26. Yang, Q.: The relaxed CQ algorithm for solving the split feasibility problem. Inverse Probl. **20**, 1261–1266 (2004)
27. Yao, Y., Qin, X., Yao, J.C.: Convergence analysis of an inertial iterate for the proximal split feasibility problem. J. Nonlinear Convex Anal. **20**, 489–498 (2019)
28. Zhang, W., Han, D., Li, Z.: A self-adaptive projection method for solving the multiple-sets split feasibility problem. Inverse Probl. **25**, 115001 (2009)

29. Zhao, J., Yang, Q.: Self-adaptive projection methods for the multiple-sets split feasibility problem. Inverse Probl. **27**, 035009 (2011)
30. Zhao, J., Zhang, Y., Yang, Q.: Modified projection methods for the split feasibility problem and multiple sets feasibility problem. Appl. Math. Comput. **219**, 1644–1653 (2012)