Fixed Point Theory, 21(2020), No. 2, 819-832
DOI: 10.24193/fpt-ro.2020.2.58
http://www.math.ubbcluj.ro/~nodeacj/sfptcj.html

MODIFIED RELAXED CQ ALGORITHMS FOR SPLIT FEASIBILITY AND SPLIT EQUALITY PROBLEMS IN HILBERT SPACES

HAI YU AND FENGHUI WANG

Department of Mathematics, Luoyang Normal University
Luoyang 471022, China

Abstract. In this paper, we investigate the split feasibility problem (SFP) and the split equality problem (SEP) in Hilbert spaces. Motivated by the technique of relaxed projections, we respectively propose a modified relaxed CQ algorithm for the SFP and a modified relaxed alternating CQ algorithm for the SEP. Under standard assumptions, we show that the proposed algorithms converge weakly to a solution of the SFP and the SEP, respectively. Finally, we conduct some numerical experiments to demonstrate the advantage of our proposed algorithms.
Key Words and Phrases: Split feasibility problem, split equality problem, CQ algorithm, projection.
2010 Mathematics Subject Classification: 47J25, 47J20, 47H10, 49N45, 65J15.

1. Introduction

Let C and Q be two nonempty closed convex subsets of the real Hilbert spaces H_1 and H_2, respectively. The split feasibility problem (SFP) is formulated as finding a point $x^* \in H_1$ with the property:

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

where $A : H_1 \to H_2$ is a bounded linear operator. The SFP was first introduced by Censor and Elfving [5] and has received much attention due to its applications in signal processing and image reconstruction [14]. To solve the SFP, Censor and Elfving [5] proposed an iterative algorithm based on the multidistance idea. But their algorithm requires the calculation of matrix inverses at each iteration. Later, in [3, 4], Byrne introduced a projection method called the CQ algorithm that does not involve matrix inverses. More specifically, the CQ algorithm is defined as follows:

$$x_{n+1} = P_C(x_n - \gamma A^*(I - P_Q)Ax_n),$$

where $\gamma \in (0, \frac{2}{\|A\|^2})$, A^* is the corresponding adjoint operator of A, and P_C and P_Q stand for the projections onto C and Q, respectively.

In the CQ algorithm, Byrne assumed that the projections P_C and P_Q are easily calculated. However, in many cases it is impossible or needs too much work to exactly compute the projection (see [1, 9, 11]). To overcome this difficulty, the relaxed projection method was adopted in some literatures [10, 12, 13, 18, 20, 21, 22, 23, 24, 25]. Yang [26] presented a relaxed CQ algorithm, in which C and Q are level sets of convex functions $c : H_1 \to \mathbb{R}$ and $q : H_2 \to \mathbb{R}$, respectively. The relaxed CQ algorithm is given as follows:

$$x_{n+1} = P_{C_n}(x_n - \gamma A^*(I - P_{Q_n})Ax_n), \tag{1.2}$$

where $\gamma \in (0, \frac{2}{\|A\|^2})$ and

$$C_n = \{x \in H_1 \mid c(x_n) + \langle \xi_n, x - x_n \rangle \leq 0\}, \quad \xi_n \in \partial c(x_n), \tag{1.3}$$

$$Q_n = \{y \in H_2 \mid q(Ax_n) + \langle \eta_n, y - Ax_n \rangle \leq 0\}, \quad \eta_n \in \partial q(Ax_n), \tag{1.4}$$

where $\partial c(x_n)$ is the subdifferential of c at x_n (see Definition 2.3).

In the relaxed CQ algorithm (1.2), since C_n and Q_n are both halfspaces, the projections P_{C_n} and P_{Q_n} have closed-form expressions. Thus they are easily to be computed. Some relaxed CQ algorithms have been considered by many authors, see, e.g., [17, 14, 22, 8, 27]. Among these works, López et al. [14] improved Yang's relaxed CQ algorithm as follows:

$$x_{n+1} = P_{C_n}(x_n - \tau_n A^*(I - P_{Q_n})Ax_n), \tag{1.5}$$

where

$$\tau_n := \frac{\rho_n \|(I - P_{Q_n})Ax_n\|^2}{2\|A^*(I - P_{Q_n})Ax_n\|^2}, \quad 0 < \rho_n < 4. \tag{1.6}$$

It is readily seen that this algorithm has no need of any prior information of the norm $\|A\|$.

Recently, Moudafi [15] introduced the following split equality problem (SEP):

$$\text{Find} \quad x \in C, \; y \in Q \quad \text{such that} \quad Ax = By, \tag{1.7}$$

where H_1, H_2, H_3 are real Hilbert spaces, $C \subseteq H_1, Q \subseteq H_2$ are two nonempty, closed and convex subsets, and $A : H_1 \to H_3, B : H_2 \to H_3$ are two bounded linear operators. Obviously, if $B = I$ and $H_3 = H_2$, then (1.7) reduces to (1.1). In order to solve the SEP (1.7), Moudafi [15] introduced the following alternating CQ algorithm (ACQA):

$$\begin{cases} x_{n+1} = P_C(x_n - \gamma_n A^*(Ax_n - By_n)), \\ y_{n+1} = P_Q(y_n + \gamma_n B^*(Ax_{n+1} - By_n)). \end{cases}$$

Similarly, the ACQA involves two projections P_C and P_Q and might be hard to be implemented. To overcome this difficulty, Moudafi [16] presented the following relaxed alternating CQ algorithm (RACQA):

$$\begin{cases} x_{n+1} = P_{C_n}(x_n - \gamma A^*(Ax_n - By_n)), \\ y_{n+1} = P_{Q_n}(y_n + \gamma B^*(Ax_{n+1} - By_n)), \end{cases} \tag{1.8}$$

where $\gamma > 0$, C_n is given as in (1.3) and Q_n is given by

$$Q_n = \{y \in H_2 \mid q(y_n) + \langle \eta_n, y - y_n \rangle \leq 0\}, \quad \eta_n \in \partial q(y_n). \tag{1.9}$$

Under suitable conditions, Moudafi [16] proved that the sequence $\{(x_n, y_n)\}$ generated by the RACQA converges weakly to a solution of (1.7).

As we see from the above, the halfspace C_n is constructed via x_n, and x_{n+1} is thus defined as the projection of $y_n := x_n - \tau_n A^*(I - P_{Q_n})Ax_n$ onto C_n. However, it seems that y_n is much better than x_n since y_n has been updated, so it is natural to directly construct a halfspace via y_n instead of x_n. On the other hand, the denominator of the stepsize τ_n in (1.5) may be zero in a certain iteration. To ensure the stepsize well defined, we also modified τ_n. In this paper, we introduce a modified relaxed CQ algorithm for the SFP (1.1) by constructing a halfspace in y_n rather than in x_n. Moreover, for the SEP (1.7), we also propose a modified relaxed alternating CQ algorithm. To illustrate our algorithm's efficiency, we present a comparison with the existing relaxed CQ algorithms.

2. Preliminaries

Throughout this paper, we denote by H a Hilbert space and by I the identity operator on H. For a differentiable functional f, denote by ∇f the gradient of f. Given a sequence $\{x_n\}$ in H, $\omega_w(x_n)$ (resp., $\omega(x_n)$) stands for the set of cluster points in the weak (resp., strong) topology. '$x_n \rightharpoonup x$' (resp.,'$x_n \to x$') means the weak (resp., strong) convergence of $\{x_n\}$ to x.

Definition 2.1. [2, 4] Let D be a nonempty subset of H and let $T : D \to H$. Then T is

(1) nonexpansive if
$$\|Tx - Ty\| \leq \|x - y\|, \quad \forall x, y \in D;$$
(2) firmly nonexpansive if
$$\|Tx - Ty\|^2 \leq \|x - y\|^2 - \|(I - T)x - (I - T)y\|^2, \quad \forall x, y \in D;$$
(3) ν-inverse strongly monotone (ν-ism) if there is $\nu > 0$ such that
$$\langle Tx - Ty, x - y\rangle \geq \nu\|Tx - Ty\|^2, \quad \forall x, y \in D.$$

For any $x \in H$, the projection onto a nonempty closed convex subset C is defined as
$$P_C x = \operatorname{argmin}\{\|y - x\| \mid y \in C\}.$$
The projection P_C has the following well-known properties.

Lemma 2.2. [2, 4] *Let $C \subseteq H$ be a nonempty closed convex subset. Then for all $x, y \in H$ and $z \in C$,*

(1) $\langle x - P_C x, z - P_C x\rangle \leq 0$;
(2) *P_C and $I - P_C$ are both nonexpansive;*
(3) *P_C and $I - P_C$ are both 1-ism;*
(4) *P_C and $I - P_C$ are both firmly nonexpansive.*

The following will be used in our convergence analysis.

Definition 2.3. Let $\lambda \in (0, 1)$ and $f : H \to (-\infty, +\infty]$ be a proper function.

(i) f is convex if
$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y), \forall x, y \in H.$$

(ii) A vector $u \in H$ is a subgradient of f at a point x if

$$f(y) \geq f(x) + \langle u, y - x \rangle, \quad \forall y \in H.$$

(iii) The set of all subgradients of f at x, denoted by $\partial f(x)$, is called the subdifferential of f.

Lemma 2.4. [2] *Suppose that H is finite-dimensional and let $f : H \to \mathbb{R}$ be a convex function. Then*

(i) *The function f is continuous;*
(ii) *The function f is subdifferentiable everywhere;*
(iii) *The subdifferentials of f are uniformly bounded on any bounded subset.*

Lemma 2.5. [4, 14] *Let $f : H \to (-\infty, +\infty]$ be defined by*

$$f(x) = \frac{1}{2}\|(I - P_Q)Ax\|^2.$$

Then

(i) *f is convex and differential.*
(ii) $\nabla f(x) = A^*(I - P_Q)Ax, \quad x \in H.$
(iii) *f is weakly lower semi-continuous on H.*
(iv) *∇f is $\|A\|^2$-Lipschitz:*

$$\|\nabla f(x) - \nabla f(y)\| \leq \|A\|^2 \|x - y\|, \quad x, y \in H.$$

The convergence analysis of the proposed algorithm is based on Fejér monotonicity.

Definition 2.6. Let C be a nonempty closed convex subset in H. A sequence $\{x_n\}$ in H is said to be Fejér monotone with respect to C if

$$\|x_{n+1} - z\| \leq \|x_n - z\|, \quad \forall n \geq 1, \quad \forall z \in C.$$

Lemma 2.7. [6] *Let C be a nonempty closed convex subset in H. If the sequence $\{x_n\}$ is Fejér monotone with respect to C, then the following hold:*

(i) *$x_n \rightharpoonup x^* \in C$ if and only if $\omega_w(x_n) \subseteq C$;*
(ii) *the sequence $\{P_C x_n\}$ converges strongly;*
(iii) *if $x_n \rightharpoonup x^* \in C$, then $x^* = \lim_{n \to \infty} P_C x_n$.*

3. A modified relaxed CQ algorithm for the split feasibility problem

In this section, we will propose a modified relaxed CQ algorithm for the SFP (1.1), in which the closed convex subsets C and Q satisfy the following assumptions.

(A1) The solution set $S = \{x \in C \mid Ax \in Q\}$ is nonempty.

(A2) The sets C and Q are given by

$$C = \{x \in H_1 \mid c(x) \leq 0\}, \tag{3.1}$$

and

$$Q = \{y \in H_2 \mid q(y) \leq 0\}, \tag{3.2}$$

where $c : H_1 \to \mathbb{R}$ and $q : H_2 \to \mathbb{R}$ are two lower semicontinuous convex functions on H_1 and H_2, respectively.

(A3) For any $x \in H_1$ and $y \in H_2$, at least one subgradient $\xi \in \partial c(x)$ and $\eta \in \partial q(y)$ can be calculated, respectively. We assume also that the subdifferential operators ∂c and ∂q are bounded on bounded sets.

It is worth noting that every convex function defined on a finite dimensional Hilbert space satisfies conditions (A2) and (A3) by Lemma 2.4. In what follows, we define

$$f_n(x) = \frac{1}{2}\|(I - P_{Q_n})Ax\|^2, \quad n \geq 0,$$

where Q_n is given as in (1.4). By Lemma 2.5, we have

$$\nabla f_n(x) = A^*(I - P_{Q_n})Ax.$$

We are now in position to introduce the following modified relaxed CQ-algorithm for solving the SFP (1.1), where C and Q are given in (3.1) and (3.2), respectively.

Algorithm 3.1. Let x_0 be arbitrary. Given x_n, construct x_{n+1} via the formula

$$\begin{cases} y_n = x_n - \tau_n \nabla f_n(x_n), \\ x_{n+1} = P_{C_n'}(y_n), \end{cases} \tag{3.3}$$

where C_n' and τ_n are respectively defined as follows:

$$C_n' = \{x \in H_1 \mid c(y_n) + \langle \xi_n, x - y_n \rangle \leq 0\}, \quad \xi_n \in \partial c(y_n),$$

and

$$\tau_n = \frac{\rho_n f_n(x_n)}{(\|\nabla f_n(x_n)\| + \epsilon)^2}, \tag{3.4}$$

where $0 < \rho_n < 4$ and $\epsilon > 0$ is a small enough number. If $x_{n+1} = y_n = x_n$, then stop; otherwise, set $n := n + 1$ and go to (3.3) to compute the next iterate x_{n+2}.

Remark 3.2. (1) It is obvious that $C \subseteq C_n'$. Since C_n' is a halfspace, the projection $P_{C_n'}$ has a closed-form expression, which indicates that our algorithm is also easily implemented.

(2) In our algorithm, we construct the halfspace C_n' via y_n instead of x_n.

(3) Compared with (1.6), in (3.4), we add the item ϵ so that the stepsize is well defined.

Theorem 3.3. *In Algorithm 3.1, if $x_{n+1} = y_n = x_n$ for some $n \geq 0$, then x_n is a solution of the SFP* (1.1).

Proof. From (3.3), if $x_{n+1} = y_n$, then $y_n \in C_n'$. This implies that $c(y_n) \leq 0$. Since $y_n = x_n$, we have $x_n \in C$ and $\tau_n \nabla f_n(x_n) = 0$. This implies that $(I - P_{Q_n})Ax_n = 0$, i.e. $Ax_n \in Q_n$. Thus we have $q(Ax_n) \leq 0$ from (1.4). Therefore $Ax_n \in Q$ and the proof is complete. $\square$

By Theorem 3.3, we see that if Algorithm 3.1 terminates in a finite (say n) step of iterations, then x_n is a solution of the SFP. Thus in the rest of this section, we assume that Algorithm 3.1 does not terminate in a finite number of iterations, and hence generates an infinite sequence $\{x_n\}$. The convergence of Algorithm 3.1 is proved below.

Theorem 3.4. *Assume that* $\inf_n \rho_n(4-\rho_n) > 0$. *Then the sequence* $\{x_n\}$ *generated by Algorithm 3.1 converges weakly to a solution of the SFP* (1.1).

Proof. Let $x^* \in S$. Since $P_{C_n'}$ is firmly nonexpansive, we have

$$\begin{aligned}\|x_{n+1}-x^*\|^2 &= \|P_{C_n'}(x_n - \tau_n \nabla f_n(x_n)) - x^*\|^2 \\ &\le \|(x_n - x^*) - \tau_n \nabla f_n(x_n)\|^2 - \|(I-P_{C_n'})y_n\|^2 \\ &= \|x_n - x^*\|^2 - 2\tau_n\langle \nabla f_n(x_n), x_n - x^*\rangle \\ &\quad + (\tau_n\|\nabla f_n(x_n)\|)^2 - \|(I-P_{C_n'})y_n\|^2.\end{aligned}$$

Note that $I - P_{Q_n}$ is 1-ism from Lemma 2.2. This implies that

$$\begin{aligned}\langle \nabla f_n(x_n), x_n - x^*\rangle &= \langle (I-P_{Q_n})Ax_n, Ax_n - Ax^*\rangle \\ &= \langle (I-P_{Q_n})Ax_n - (I-P_{Q_n})Ax^*, Ax_n - Ax^*\rangle \\ &\ge \|(I-P_{Q_n})Ax_n\|^2 \\ &= 2f_n(x_n).\end{aligned}$$

Hence

$$\begin{aligned}\|x_{n+1}-x^*\|^2 &\le \|x_n - x^*\|^2 - 4\tau_n f_n(x_n) + \tau_n^2(\|\nabla f_n(x_n)\| + \epsilon)^2 - \|(I-P_{C_n'})y_n\|^2 \\ &= \|x_n - x^*\|^2 - \rho_n(4-\rho_n)\frac{f_n^2(x_n)}{(\|\nabla f_n(x_n)\|+\epsilon)^2} - \|(I-P_{C_n'})y_n\|^2.\end{aligned} \tag{3.5}$$

Since $\inf_n \rho_n(4-\rho_n) > 0$, the sequence $\{x_n\}$ is Fejér monotone with respect to S. This implies that the sequence $\{\|x_n - x^*\|\}$ is convergent and hence $\{x_n\}$ is a bounded sequence. Furthermore, from (3.5) and the assumption on ρ_n, we can immediately get that

$$\sum_{n=0}^{\infty} \|(I-P_{C_n'})y_n\|^2 < \infty,$$

and

$$\sum_{n=0}^{\infty} \frac{f_n^2(x_n)}{(\|\nabla f_n(x_n)\|+\epsilon)^2} < \infty.$$

In particular, we have

$$\lim_{n\to\infty} \|(I-P_{C_n'})y_n\| = 0.$$

and

$$\lim_{n\to\infty} \frac{f_n(x_n)}{\|\nabla f_n(x_n)\|+\epsilon} = 0. \tag{3.6}$$

By the Lipschitz continuity of $\nabla f_n(x)$, we have

$$\|\nabla f_n(x_n)\| + \epsilon = \|\nabla f_n(x_n) - \nabla f_n(x^*)\| + \epsilon \le \|A\|^2\|x_n - x^*\| + \epsilon.$$

This implies that $\{\|\nabla f_n(x_n)\| + \epsilon\}$ is bounded. It then follows from (3.6) that $f_n(x_n) \to 0$, namely $\|(I-P_{Q_n})Ax_n\| \to 0$, as $n \to \infty$.

By Lemma 2.7, to show the weak convergence of $\{x_n\}$ to a solution of the SFP (1.1), it suffices to show that $\omega_w(x_n) \subseteq S$, since $\{x_n\}$ is Fejér monotone with respect to S. Now let $\bar{x} \in \omega_w(x_n)$ and $\{x_{n_k}\}$ be a subsequence of $\{x_n\}$ such that $x_{n_k} \rightharpoonup \bar{x}$. In the following, we show that $\bar{x}$ is a solution of the SFP (1.1). First we show $A\bar{x} \in Q$. Since ∂q is bounded on bounded sets, there is a constant $\delta_1 > 0$ such that $\|\eta_n\| \leq \delta_1$ for all $n \geq 0$. From (1.4) and the fact that $P_{Q_n}(Ax_n) \in Q_n$, it follows that

$$q(Ax_n) \leq \langle \eta_n, Ax_n - P_{Q_n}(Ax_n)\rangle \leq \delta_1 \|(I - P_{Q_n})Ax_n\|. \tag{3.7}$$

The weakly lower semicontinuity of q and (3.7) imply that

$$q(A\bar{x}) \leq \liminf_{k\to\infty} q(Ax_{n_k}) \leq \lim_{k\to\infty} \delta_1 \|(I - P_{Q_{n_k}})Ax_{n_k}\| = 0.$$

It turns out that $A\bar{x} \in Q$.

We next turn to prove $\bar{x} \in C$. It is easily seen that

$$\|y_n - x_n\| = \tau_n \|\nabla f_n(x_n)\| \leq \frac{\rho_n f_n(x_n)}{\|\nabla f_n(x_n)\| + \epsilon} \to 0.$$

This implies that the sequence $\{y_n\}$ is bounded and $y_{n_k} \rightharpoonup \bar{x}$, since $\{x_n\}$ is bounded and $x_{n_k} \rightharpoonup \bar{x}$. Since ∂c is bounded on bounded sets, there is a constant $\delta_2 > 0$ such that $\|\xi_n\| \leq \delta_2$ for all $n \geq 0$. By the definition of C_n' and the fact that $P_{C_n'}(y_n) \in C_n'$, we obtain

$$c(y_n) \leq \langle \xi_n, y_n - P_{C_n'} y_n\rangle \leq \delta_2 \|(I - P_{C_n'})y_n\|. \tag{3.8}$$

Again, the weakly lower semicontinuity of c and (3.8) imply that

$$c(\bar{x}) \leq \liminf_{k\to\infty} c(y_{n_k}) \leq \lim_{k\to\infty} \delta_2 \|(I - P_{C_{n_k}'})y_{n_k}\| = 0.$$

Consequently, $\bar{x} \in C$. Altogether we conclude that $\bar{x}$ is a solution of the SFP. This completes the proof. □

4. A modified relaxed alternating CQ algorithm for the split equality problem

For problem (1.7), we always assume that the following assumption holds:

(**A4**) The solution set $S = \{x \in C, y \in Q \mid Ax = By\}$ is nonempty.

In what follows, we will treat problem (1.7) under the assumptions (**A2**), (**A3**) and (**A4**). Let us now introduce a modified relaxed alternating CQ algorithm for the SEP (1.7).

Algorithm 4.1. Let (x_0, y_0) be arbitrary. Given (x_n, y_n), construct (x_{n+1}, y_{n+1}) via the formula

$$\begin{cases} u_n = x_n - \gamma A^*(Ax_n - By_n), \\ x_{n+1} = P_{C_n''}(u_n); \\ v_n = y_n + \gamma B^*(Ax_{n+1} - By_n), \\ y_{n+1} = P_{Q_n''}(v_n), \end{cases} \tag{4.1}$$

where $\gamma \in (0, \min(\frac{1}{\|A\|^2}, \frac{1}{\|B\|^2}))$, C_n'' and Q_n'' are respectively defined as follows:

$$C_n'' = \{x \in H_1 \mid c(u_n) + \langle \xi_n, x - u_n\rangle \leq 0\}, \quad \xi_n \in \partial c(u_n), \tag{4.2}$$

and

$$Q_n'' = \{y \in H_2 \mid q(v_n) + \langle \eta_n, y - v_n \rangle \le 0\}, \quad \eta_n \in \partial q(v_n). \tag{4.3}$$

Remark 4.2. It is obvious that $C \subseteq C_n''$ and $Q \subseteq Q_n''$. Since C_n'' and Q_n'' are also halfspaces, the proposed algorithm is easily implemented.

Theorem 4.3. *Let $\{(x_n, y_n)\}$ be the sequence generated by Algorithm 4.1. Then $\{(x_n, y_n)\}$ converges weakly to a solution of the SEP* (1.7).

Proof. Let $(x^*, y^*) \in S$ be arbitrarily chosen. Then $x^* \in C$ (and thus $x^* \in C_n''$); $y^* \in Q$ (and thus $y^* \in Q_n''$), $Ax^* = By^*$. Using the fact that $P_{C_n''}$ is firmly nonexpansive, the first equality of the algorithm 4.1 gives

$$\begin{aligned} \|x_{n+1} - x^*\|^2 &= \|x_n - x^* - \gamma A^*(Ax_n - By_n)\|^2 - \|(I - P_{C_n''})u_n\|^2 \\ &\le \|x_n - x^*\|^2 - 2\gamma\langle Ax_n - By_n, Ax_n - Ax^*\rangle \\ &+\gamma^2\|A\|^2\|Ax_n - By_n\|^2 - \|(I - P_{C_n''})u_n\|^2. \end{aligned} \tag{4.4}$$

Similarly, the second equality of the algorithm 4.1 leads to

$$\begin{aligned} \|y_{n+1} - y^*\|^2 &= \|y_n - y^* + \gamma B^*(Ax_{n+1} - By_n)\|^2 - \|(I - P_{Q_n''})v_n\|^2 \\ &\le \|y_n - y^*\|^2 + 2\gamma\langle Ax_{n+1} - By_n, By_n - By^*\rangle \\ &+\gamma^2\|B\|^2\|Ax_{n+1} - By_n\|^2 - \|(I - P_{Q_n''})v_n\|^2. \end{aligned} \tag{4.5}$$

On the other hand, we have

$$\begin{aligned} 2\langle Ax_n - By_n, Ax_n - Ax^*\rangle &= \|Ax_n - By_n\|^2 + \|Ax_n - Ax^*\|^2 - \|By_n - Ax^*\|^2 \\ &= \|Ax_n - By_n\|^2 + \|Ax_n - Ax^*\|^2 - \|By_n - By^*\|^2 \end{aligned} \tag{4.6}$$

and

$$\begin{aligned} 2\langle Ax_{n+1} - By_n, By_n - By^*\rangle &= -\|Ax_{n+1} - By_n\|^2 - \|By_n - By^*\|^2 \\ &\quad +\|Ax_{n+1} - By^*\|^2 \\ &= -\|Ax_{n+1} - By_n\|^2 - \|By_n - By^*\|^2 \\ &\quad +\|Ax_{n+1} - Ax^*\|^2. \end{aligned} \tag{4.7}$$

It follows from (4.4)-(4.7) that

$$\begin{aligned} \|x_{n+1} - x^*\|^2 \le{}& \|x_n - x^*\|^2 - \gamma(1 - \gamma\|A\|^2)\|Ax_n - By_n\|^2 \\ &- \gamma\|Ax_n - Ax^*\|^2 + \gamma\|By_n - By^*\|^2 - \|(I - P_{C_n''})u_n\|^2. \end{aligned}$$

and

$$\begin{aligned} \|y_{n+1} - y^*\|^2 \le{}& \|y_n - y^*\|^2 - \gamma(1 - \gamma\|B\|^2)\|Ax_{n+1} - By_n\|^2 \\ &- \gamma\|By_n - By^*\|^2 + \gamma\|Ax_{n+1} - Ax^*\|^2 - \|(I - P_{Q_n''})v_n\|^2. \end{aligned}$$

Adding the two last inequalities, we obtain

$$\begin{aligned} &\|x_{n+1} - x^*\|^2 + \|y_{n+1} - y^*\|^2 \\ &\le \|x_n - x^*\|^2 + \|y_n - y^*\|^2 - \gamma\|Ax_n - Ax^*\|^2 + \gamma\|Ax_{n+1} - Ax^*\|^2 \\ &-\gamma(1 - \gamma\|A\|^2)\|Ax_n - By_n\|^2 - \|(I - P_{C_n''})u_n\|^2 \\ &-\gamma(1 - \gamma\|B\|^2)\|Ax_{n+1} - By_n\|^2 - \|(I - P_{Q_n''})v_n\|^2. \end{aligned} \tag{4.8}$$

Let

$$\Gamma_n(x^*, y^*) = \|x_n - x^*\|^2 + \|y_n - y^*\|^2 - \gamma\|Ax_n - Ax^*\|^2. \tag{4.9}$$

We note that

$$\gamma\|Ax_n - Ax^*\|^2 \leq \gamma\|A\|^2\|x_n - x^*\|^2.$$

Therefore

$$\Gamma_n(x^*, y^*) \geq (1 - \gamma\|A\|^2)\|x_n - x^*\|^2 + \|y_n - y^*\|^2 \geq 0. \tag{4.10}$$

In view of (4.8), we obtain the following inequality

$$\begin{aligned} \Gamma_{n+1}(x^*, y^*) \quad & \leq \Gamma_n(x^*, y^*) - \gamma(1 - \gamma\|A\|^2)\|Ax_n - By_n\|^2 - \|(I - P_{C''_n})u_n\|^2 \\ & -\gamma(1 - \gamma\|B\|^2)\|Ax_{n+1} - By_n\|^2 - \|(I - P_{Q''_n})v_n\|^2. \end{aligned} \tag{4.11}$$

This together with (4.10) implies that the sequence $\{\Gamma_n(x^*, y^*)\}$ is decreasing and lower bounded by 0. Consequently the sequence $\{\Gamma_n(x^*, y^*)\}$ is bounded and converges to some finite limit $\gamma(x^*, y^*)$. By passing to the limit in (4.11) and by taking into account the assumption on γ, we finally obtain that

$$\lim_{n\to+\infty} \|Ax_n - By_n\| = \lim_{n\to+\infty} \|(I - P_{C''_n})u_n\| = 0,$$

and

$$\lim_{n\to+\infty} \|Ax_{n+1} - By_n\| = \lim_{n\to+\infty} \|(I - P_{Q''_n})v_n\| = 0.$$

Since $\{\Gamma_n(x^*, y^*)\}$ is bounded, in view of (4.10), the sequences $\{x_n\}$ and $\{y_n\}$ are also bounded. Let $\bar{x}$ and $\bar{y}$ be respectively weak cluster points of the sequences $\{x_n\}$ and $\{y_n\}$. Without loss of generality, we can assume that $x_n \rightharpoonup \bar{x}$ and $y_n \rightharpoonup \bar{y}$. By the definitions of u_n and v_n, it follows that

$$\|u_n - x_n\| = \gamma\|A^*(Ax_n - By_n)\| \leq \gamma\|A\|\|Ax_n - By_n\| \to 0,$$

and

$$\|v_n - y_n\| = \gamma\|B^*(Ax_{n+1} - By_n)\| \leq \gamma\|B\|\|Ax_{n+1} - By_n\| \to 0.$$

This implies that $u_n \rightharpoonup \bar{x}$ and $v_n \rightharpoonup \bar{y}$.

Since ∂c is bounded on bounded sets, there is a constant $\delta_1 > 0$ such that $\|\xi_n\| \leq \delta_1$ for all $n \geq 0$. From (4.2) and the fact that $P_{C''_n}(u_n) \in C''_n$, it follows that

$$c(u_n) \leq \langle \xi_n, (I - P_{C''_n})u_n\rangle \leq \delta_1\|(I - P_{C''_n})u_n\|. \tag{4.12}$$

The weakly lower semicontinuity of c and (4.12) imply that

$$c(\bar{x}) \leq \liminf_{n\to\infty} c(u_n) \leq \lim_{n\to\infty} \delta_1\|(I - P_{C''_n})u_n\| = 0.$$

It turns out that $\bar{x} \in C$. Likewise, Since ∂q is bounded on bounded sets, there is a constant $\delta_2 > 0$ such that $\|\eta_n\| \leq \delta_2$ for all $n \geq 0$. From (4.3) and the fact that $P_{Q''_n}(u_n) \in Q''_n$, it follows that

$$q(v_n) \leq \langle \eta_n, (I - P_{Q''_n})v_n\rangle \leq \delta_2\|(I - P_{Q''_n})v_n\|.$$

Again, the weakly lower semicontinuity of q leads to

$$q(\bar{y}) \leq \liminf_{n\to\infty} q(v_n) \leq \lim_{n\to\infty} \delta_2\|(I - P_{Q''_n})v_n\| = 0.$$

Therefore $\bar{y} \in Q$. Furthermore, the weak convergence of $\{Ax_n - By_n\}$ to $A\bar{x} - B\bar{y}$ and the weakly lower semicontinuity of the squared norm imply

$$\|A\bar{x} - B\bar{y}\|^2 \leq \liminf_{n\to+\infty} \|Ax_n - By_n\|^2 = 0.$$

Hence $(\bar{x}, \bar{y}) \in S$.

We next turn to show the uniqueness of the weak cluster point. Let $\hat{x}$ and $\hat{y}$ be other weak cluster points of $\{x_n\}$ and $\{y_n\}$, respectively. By the definition of Γ_n, we have

$$\begin{aligned}\Gamma_n(\bar{x}, \bar{y}) = {} & \Gamma_n(\hat{x}, \hat{y}) + \|\bar{x} - \hat{x}\|^2 + \|\bar{y} - \hat{y}\|^2 - \gamma\|A\bar{x} - A\hat{x}\|^2 \\ & + 2\langle x_n - \hat{x}, \hat{x} - \bar{x}\rangle + 2\langle y_n - \hat{y}, \hat{y} - \bar{y}\rangle - 2\gamma\langle Ax_n - A\hat{x}, A\hat{x} - A\bar{x}\rangle.\end{aligned}$$

By passing to the limit in the above relation, we obtain

$$\gamma(\bar{x}, \bar{y}) = \gamma(\hat{x}, \hat{y}) + \|\bar{x} - \hat{x}\|^2 + \|\bar{y} - \hat{y}\|^2 - \gamma\|A\bar{x} - A\hat{x}\|^2.$$

Reversing the role of $(\bar{x}, \bar{y})$ and $(\hat{x}, \hat{y})$, we also have

$$\gamma(\hat{x}, \hat{y}) = \gamma(\bar{x}, \bar{y}) + \|\bar{x} - \hat{x}\|^2 + \|\bar{y} - \hat{y}\|^2 - \gamma\|A\bar{x} - A\hat{x}\|^2.$$

By adding the two last equalities, we obtain

$$(1 - \gamma\|A\|^2)\|\bar{x} - \hat{x}\|^2 + \|\bar{y} - \hat{y}\|^2 \leq 0.$$

Since $1 - \gamma\|A\|^2 > 0$, we obtain $\bar{x} = \hat{x}$ and $\bar{y} = \hat{y}$, which implies that the whole sequence $\{(x_n, y_n)\}$ converges weakly to a solution of problem (1.7). This completes the proof. $\square$

5. Numerical experiments

In this section, we present two numerical experiments, to illustrate the performance of the proposed algorithms. For simplicity, we denote Yang's relaxed CQ algorithm (1.2), López's relaxed CQ algorithm (1.5) and Moudafi's relaxed alternating CQ algorithm (1.8) by Yang's algorithm, López's algorithm and Moudafi's algorithm, respectively. These algorithms are coded in MATLAB 2012b on a 4 GB RAM, 3.30 GHz, Intel(R) Core(TM) i5-4590 personal computer. In what follows, Iter. denotes the numbers of iterations, and CPU denotes the computing time.

Example 1. In this example, we apply Algorithm 3.1 to solve the LASSO problem. Let us first recall the LASSO problem [19] which is given as follows:

$$\begin{aligned}&\min_{x\in\mathbb{R}^n} \tfrac{1}{2}\|Ax - b\|^2, \\ &s.t. \quad \|x\|_1 \leq t,\end{aligned} \tag{5.1}$$

where $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$ and $t > 0$ is a given constant. Let $C = \{x \mid c(x) \leq 0\}$, where $c(x) = \|x\|_1 - t$ and $Q = \{b\}$, then problem (5.1) can be seen as an SFP (1.1). In this example, the vector $x \in \mathbb{R}^n$ is a K-sparse signal that is generated from uniform distribution in the interval $[-2, 2]$ with K non-zero elements. The matrix $A \in \mathbb{R}^{m\times n}$ is generated from a normal distribution with mean zero and one variance. The vector b is taken as equal to Ax, so no noise is assumed. The goal is then to recover the K-sparse signal x by solving the LASSO problem (5.1).

Throughout the experiment, the parameters used in these algorithms are set with $t = K$, $\epsilon = 10^{-6}$, $\gamma = \frac{1}{\|A\|^2}$, $\rho_n = 2$. The stopping criteria is that $\|x_{n+1} - x_n\| \le \epsilon$. The results are reported in Table 1 and Table 2.

TABLE 1. Numerical results for Example 1 when $m = 120$, $n = 512$

K-sparse signal	Yang's algorithm		López's algorithm		Algorithm 3.1	
	iter.	CPU (s)	iter.	CPU (s)	iter.	CPU (s)
$K = 10$	736	0.1937	571	0.1592	396	0.1124
$K = 20$	1706	0.4282	1395	0.3249	835	0.2088
$K = 30$	8368	1.7704	7440	1.6380	4245	0.8920

TABLE 2. Numerical results for Example 1 when $m = 240$, $n = 1024$

K-sparse signal	Yang's algorithm		López's algorithm		Algorithm 3.1	
	iter.	CPU (s)	iter.	CPU (s)	iter.	CPU (s)
$K = 20$	1269	2.6727	1124	2.3923	670	1.4781
$K = 30$	2150	4.5455	1868	3.9673	1135	2.3790
$K = 40$	8821	18.4178	7985	16.4961	4923	10.1224

From Tables 1-2, our algorithm demonstrates better performance compared with Yang's algorithm and López's algorithm, in terms of cpu time and the numbers of iterations.

Example 2. Let

$$C = \left\{ x \in \mathbb{R}^n \mid \sum_{i=1}^{n} 10^{\frac{i-1}{n-1}} x_i^2 \le 1 \right\},$$

$$Q = \left\{ y \in \mathbb{R}^m \mid \sum_{i=1}^{m} 10^{\frac{i-1}{m-1}} y_i^2 \le 1 \right\}.$$

The matrices $A = (a_{ij})_{p\times n}$, $a_{ij} \in [0, 10]$ and $B = (b_{ij})_{p\times m}$, $b_{ij} \in [0, 10]$ are generated randomly. In this example, we apply Algorithm 4.1 to solve the split equality problem:

$$\text{Find} \quad x \in C, \ y \in Q \quad \text{such that} \quad Ax = By.$$

It is obvious that C and Q are both ellipsoids [7]. Let

$$c(x) = \sum_{i=1}^{n} 10^{\frac{i-1}{n-1}} x_i^2 - 1 \text{ and } q(y) = \sum_{i=1}^{m} 10^{\frac{i-1}{m-1}} y_i^2 - 1,$$

then $C = \{x \in \mathbb{R}^n \mid c(x) \le 0\}$ and $Q = \{y \in \mathbb{R}^m \mid q(y) \le 0\}$.
Throughout the experiment, the parameters used in these algorithms are set with

$$n = m = p = 10, \ \epsilon = 10^{-6}, \ \gamma = 0.9 \times \min\left(\frac{1}{\|A\|^2}, \ \frac{1}{\|B\|^2}\right).$$

The stopping criteria is that $\|x_{n+1} - x_n\|^2 + \|y_{n+1} - y_n\|^2 \le \epsilon^2$. The numerical results are reported in Table 3 using different initial points.

Table 3. Numerical results for Example 2

Initial points	Moudafi's algorithm		Algorithm 4.1	
	iter.	CPU (s)	iter.	CPU (s)
$x_0 = (1, 1, \cdots, 1)^T$ $y_0 = (1, 1, \cdots, 1)^T$	125	0.0194	116	0.0149
$x_0 = (10, 10, \cdots, 10)^T$ $y_0 = (10, 10, \cdots, 10)^T$	176	0.0263	157	0.0152
$x_0 = (100, 100, \cdots, 100)^T$ $y_0 = (100, 100, \cdots, 100)^T$	188	0.0234	163	0.0153
$x_0 = (1, 2, \cdots, n)^T$ $y_0 = (1, 2, \cdots, m)^T$	164	0.0183	148	0.0139

From Tables 3, our algorithm demonstrates better performance than Moudafi's algorithm, in terms of cpu time and the numbers of iterations.

Acknowledgments. This work is supported by the National Natural Science Foundation of China (No. 11971216) and Foundation of He'nan Educational Committee (No. 20A110029, 16A520064, 15A520087).

References

[1] H.H. Bauschke, J.M. Borwein, *On projection algorithms for solving convex feasibility problems*, SIAM Rev., **38**(1996), 367-426.

[2] H.H. Bauschke, P.L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Space*, Springer-Verlag, 2011.

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

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

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

[6] P.L. Combettes, *Quasi-Fejérian analysis of some optimization algorithms*, In: D. Butnariu, Y. Censor, S. Reich (eds.), Inherently Parallel Algorithms in Feasibility and Optimization and Their Applications, Elsevier, New York, 2001, 115-152.

[7] Y.H. Dai, *Fast algorithms for projection on an ellipsoid*, SIAM J. Optim., **16**(2006), 986-1006.

[8] Q. Dong, Y. Yao, S. He, *Weak convergence theorems of the modified relaxed projection algorithms for the split feasibility problem in Hilbert spaces*, Optimization Lett., **8**(3)(2014), 1031-1046.

[9] M. Fukushima, *A relaxed projection method for variational inequalities*, Math. Program., **35**(1986), 58-70.

[10] A. Gibali, L. Liu, Y. Tang, *Note on the modified relaxation CQ algorithm for the split feasibility problem*, Optim. Lett., **12**(4)(2018), 813-830.

[11] B.S. He, *Inexact implicit methods for monotone general variational inequalities*, Math. Program., **A86**(1999), 199-217.

[12] S. He, Z. Zhao, *Strong convergence of a relaxed CQ algorithm for the split feasibility problem*, J. Ineq. Appl., **2013**, 2013:197.

[13] S. He, Z. Zhao, B. Luo, *A relaxed self-adaptive CQ algorithm for the multiple-setes split feasibility problem*, Optimization, **64**(2015), 1907-1918.

[14] G. López, V. Martín, F. Wang, H.K. Xu, *Solving the split feasibility problem without prior knowledge of matrix norms*, Inverse Probl., **28**(2012), 085004.

[15] A. Moudafi, *Alternating CQ-algorithm for convex feasibility and split fixed-point problems*, J. Nonlinear Convex Anal., **15**(2014), 809-818.
[16] A. Moudafi, *A relaxed alternating CQ-algorithm for convex feasibility problems*, Nonlinear Anal., **79**(2013), 117-121.
[17] B. Qu, N.H. Xiu, *A note on the CQ algorithm for the split feasibility problem*, Inverse Probl., **21**(2005), 1655-1665.
[18] B. Qu, N.H. Xiu, *A new halfspace-relaxation projection method for the split feasibility problem*, Linear Algebra Appl., **428**(2008), 1218-1229.
[19] R. Tibshirani, *Regression shrinkage and selection via the LASSO*, J.R. Stat. Soc. B, **58**(1996), 267-288.
[20] F. Wang, *A splitting-relaxed projection method for solving the split feasibility problem*, Fixed Point Theory, **14**(2013), 211-218.
[21] F. Wang, *Polyak's gradient method for split feasibility problem constrained by level sets*, Numerical Algorithms, **77**(2018), 925-938.
[22] Z. Wang, Q. Yang, Y. Yang, *The relaxed inexact projection methods for the split feasibility problem*, Applied Mathematics and Computation, **217**(12)(2011), 5347-5359.
[23] H.K. Xu, *A variable Krasnosel'skii-Mann algorithm and the multiple-set split feasibility problem*, Inverse Probl., **22**(6)(2006), 2021.
[24] H.K. Xu, *Iterative methods for the split feasibility problem in infinite-dimensional Hilbert spaces*, Inverse Probl., **26**(2010), 105018.
[25] Q. Yang, *On variable-step relaxed projection algorithm for variational inequalities*, J. Math. Anal. Appl., **302**(2005), 166-179.
[26] Q. Yang, *The relaxed CQ algorithm solving the split feasibility problem*, Inverse Probl., **20**(2004), 1261-1266.
[27] H. Yu, W. Zhan, F. Wang, *The ball-relaxed CQ algorithms for the split feasibility problem*, Optimization, **67**(2018), 1687-1699.

Received: November 1st, 2019; Accepted: Januray 10, 2020.