

RESEARCH

Open Access



# New generalized variable stepsizes of the CQ algorithm for solving the split feasibility problem

Peiyuan Wang<sup>1,2\*</sup> , Jianjun Zhou<sup>1</sup>, Risheng Wang<sup>1</sup> and Jie Chen<sup>1</sup>

\*Correspondence: wangpy629@163.com  
<sup>1</sup>Postdoctoral workstation, China Marine Development and Research Center (CMDRC), P.O. Box 1301, Beijing, 102249, China  
<sup>2</sup>Naval Aviation University, Yantai, 266041, China

## Abstract

Variable stepsize methods are effective for various modified CQ algorithms to solve the split feasibility problem (SFP). The purpose of this paper is first to introduce two new simpler variable stepsizes of the CQ algorithm. Then two new generalized variable stepsizes which can cover the former ones are also proposed in real Hilbert spaces. And then, two more general KM (Krasnosel'skii-Mann)-CQ algorithms are also presented. Several weak and strong convergence properties are established. Moreover, some numerical experiments have been taken to illustrate the performance of the proposed stepsizes and algorithms.

**MSC:** 46E20; 46H35; 47H14; 47L30

**Keywords:** split feasibility problem; CQ algorithm; variable stepsize; generalized variable stepsize

## 1 Introduction

Since the CQ algorithm for solving the split feasibility problem (SFP) was proposed [1] in order to get better convergence speed, much attention has been paid to improve the variable stepsize of CQ algorithm.

Let  $H_1$  and  $H_2$  be real Hilbert spaces,  $C$  and  $Q$  be nonempty closed convex subsets of  $H_1$  and  $H_2$ , respectively, and  $A : H_1 \rightarrow H_2$  be a bounded linear operator. In this setting, the SFP [2] is formulated as finding a point  $\hat{x}$  with the requirement

$$\hat{x} \in C \quad \text{and} \quad A\hat{x} \in Q. \quad (1.1)$$

Denote the set of solutions for the SFP by  $\Gamma = C \cap A^{-1}(Q)$ , and define  $f : H_1 \rightarrow R$  by

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

we see the function  $f(x)$  is convex and continuously differentiable.

In [1, 3], Byrne proposed his CQ algorithm; it generates a sequence  $\{x_n\}$  via the recursion

$$x_{n+1} = P_C(I - \tau_n A^*(I - P_Q)Ax_n), \quad n \geq 0, \quad (1.2)$$

where initial  $x_0 \in H_1$  and  $\tau_n \in (0, 2/\|A\|^2)$ ,  $P_C$  and  $P_Q$  are the orthogonal projections onto  $C$  and  $Q$ , respectively.

Considering that the projection onto the nonempty closed convex sets  $C$  and  $Q$  might be hard to be implemented, Yang [4] and Xu [5] showed that if  $C$  and  $Q$  are level sets of convex functions, it just needs to project onto half-spaces  $C_n$  and  $Q_n$ , thus the so-called relaxed CQ algorithm is reduced to the following formula:

$$x_{n+1} = P_{C_n}(I - \tau_n A^*(I - P_{Q_n})Ax_n), \quad n \geq 0. \tag{1.3}$$

In (1.3), the projections  $P_{C_n}$  and  $P_{Q_n}$  have closed-form expressions, thus they are easily to be computed. Define  $f_n : H_1 \rightarrow \mathbb{R}$  by

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

then the convex objective  $f_n$  is also differentiable and has a Lipschitz gradient given by

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

Noting that when applying (1.2) and (1.3) to solve the practical problems such as signal processing and image reconstruction, which can be covered by the SFP, it is hard to avoid that a fixed stepsize related to the norm of  $A$  sometimes affects convergence of the algorithms. Therefore, in order not to compute the matrix inverse and the largest eigenvalue of the matrix  $A^T A$  and have a sufficient decrease of the objective function at each iteration, people have invented various algorithms with a variable or self-adaptive stepsize. Since Qu [6] presented a searching method by adopting Armijo-like search, many similar methods have been proposed, such as [7–12] etc. However, through these methods, self-adaptive stepsize at each iteration can be achieved, most formats of them are becoming more complex, it is difficult to apply them to some practical problems, and this needs considerable time complexity, especially for the large-scale setting and sparse problem.

On the other hand, another way to construct the variable stepsize without calculating matrix norm was proposed by Yang in [13], that is,

$$\tau_n := \frac{\rho_n}{\|\nabla f(x_n)\|}, \tag{1.4}$$

where  $\{\rho_n\}$  is a sequence of positive real numbers such that

$$\sum_{n=0}^{\infty} \rho_n = \infty, \quad \sum_{n=0}^{\infty} \rho_n^2 < \infty.$$

at the same time, the additional conditions that  $Q$  is a bounded subset and  $A$  is a full column rank matrix are required. Wang *et al.* [14] applied (1.4) to solve the SFP. Afterwards, in order to remove the two additional conditions, López *et al.* [15] introduced another choice of the stepsize sequence  $\{\tau_n\}$  as follows:

$$\tau_n := \frac{\rho_n f_n(x_n)}{\|\nabla f_n(x_n)\|^2}, \tag{1.5}$$

where  $\{\rho_n\}$  is chosen in the interval  $(0, 4)$ . Furthermore, in paper [16] we were aware that if  $\nabla f_n(x_n) = 0$  in (1.5) for some  $n \geq 1$ , the corresponding algorithms in [15] have to be terminated. In this case,  $x_n$  may not be in  $C$  and is not necessarily a solution of SFP (1.1). With this observation, we have introduced a stepsize sequence  $\{\tau_n\}$  as follows:

$$\tau_n := \frac{\rho_n f_n(x_n)}{(\|\nabla f_n(x_n)\| + \omega_n)^2}, \tag{1.6}$$

where  $\{\rho_n\}$  is also chosen in the interval  $(0, 4)$  and  $\{\omega_n\}$  is a sequence of positive numbers in  $(0, 1)$ . This choice of the stepsize sequence makes the associated algorithms never terminate unless the solution of SFP has been found. However, there exists inconvenience dealing with the choice of the parameter  $\omega_n$ . Only when it is a small number, a similar convergence speed as adopting (1.5) can be guaranteed. After observing many experiments, the order of magnitude for  $\omega_n$  usually about less than  $10^{-5}$  can satisfy that, it closely relates with the bit of computer and the float precision of the calculation software.

In order to improve these and avoid the calculations of  $f_n(x)$  or  $\nabla f_n(x_n)$ , in this paper, we firstly propose two simpler choices of a stepsize as follows:

$$\tau_n := \frac{\rho_n \|x_n - \bar{x}_n\|^2}{\|Ax_n - A\bar{x}_n\|^2}, \tag{1.7}$$

and

$$\tau_n := \frac{\rho_n \|\bar{x}_n\|^2}{\|A\bar{x}_n\|^2}, \tag{1.8}$$

where  $\{\rho_n\}$  is chosen in the interval  $(0, 2)$ ,  $x_n \neq \bar{x}_n$ , and  $\bar{x}_n \neq 0$ . The advantages of our choices (1.7) and (1.8) lie in facts that they not only possess simpler formats easily to be calculated and implemented in practice, but also have significantly faster convergence speed, especially in a large-scale setting and sparse problem. Secondly, we present two generalized variable stepsizes as follows:

$$\tau_n := \frac{\rho_n \|r(x_n)\|^2}{\|Ar(x_n)\|^2} \tag{1.9}$$

or

$$\tau_n := \frac{\rho_n \|p(x_n)\|^2}{\|A^T p(x_n)\|^2}, \tag{1.10}$$

where  $\{\rho_n\}$  is chosen in the interval  $(0, 2)$ , and  $r(x_n) \neq 0$  and  $p(x_n) \neq 0$ . Consequently, stepsizes (1.5)-(1.8) are the special cases of (1.9) or (1.10), and many similar stepsize formats can be obtained from them.

Recently, Yao *et al.* [17] have applied (1.5) to an improved CQ algorithm with a generalized Halpern iteration. In paper [18], we have modified the relative parameters with satisfactory conditions. Then, in this paper, we combine the iterations in [17, 18] with the KM-CQ iterations in [19, 20]. We propose two more general KM-CQ algorithms with the generalized variable stepsize (1.9) or (1.10), and they can be used to approach the minimum norm solution of the SFP that solves special variational inequalities.

The rest of this paper is organized as follows. Section 2 reviews some propositions and known lemmas. Section 3 gives two modified CQ algorithms with simpler variable stepsizes and shows the weak convergence. Section 4 presents two general KM-CQ algorithms with the generalized variable stepsizes and proves the strong convergence. In Section 5, we include numerical experiments to testify the better performance of the proposed stepsizes and algorithms with typical problems of signal processing and image restoration. Finally, Section 6 gives some conclusions and further research aim.

## 2 Preliminaries

Let  $H$  be a real Hilbert space with the inner product  $\langle \cdot, \cdot \rangle$  and the norm  $\| \cdot \|$ , respectively. Let  $C$  be a nonempty closed convex subset of  $H$ . We will use the following notations:

- ‘ $\rightarrow$ ’ stands for strong convergence;
- ‘ $\rightharpoonup$ ’ stands for weak convergence;
- $I$  stands for the identity mapping on  $H$ .

Recall that a mapping  $T : C \rightarrow H$  is nonexpansive iff  $\|Tx - Ty\| \leq \|x - y\|$  for all  $x, y \in C$ .

A mapping  $\psi : C \rightarrow H$  is said to be  $\delta$ -contractive iff there exists a constant  $\delta \in [0, 1)$  such that

$$\| \psi(x) - \psi(y) \| \leq \delta \|x - y\|$$

for all  $x, y \in C$ .

Recall also that the nearest point projection from  $H$  onto  $C$ , denoted by  $P_C$ , assigns to each  $x \in H$  the unique point  $P_Cx \in C$  with the property  $\|x - P_Cx\| \leq \|x - y\|, \forall y \in C$ . We collect the basic properties of  $P_C$  as follows.

**Proposition 2.1** ([6, 21])

- (p<sub>1</sub>)  $\langle x - P_Cx, y - P_Cx \rangle \leq 0$  for all  $x \in H, y \in C$ ;
- (p<sub>2</sub>)  $\|P_Cx - P_Cy\|^2 \leq \langle P_Cx - P_Cy, x - y \rangle$  for every  $x, y \in H$ ;
- (p<sub>3</sub>)  $\|P_Cx - P_Cy\|^2 \leq \|x - y\|^2 - \|(I - P_C)x - (I - P_C)y\|^2$  for all  $x, y \in H$ ;
- (p<sub>4</sub>)  $\langle (I - P_C)x - (I - P_C)y, x - y \rangle \geq \|(I - P_C)x - (I - P_C)y\|^2$  for all  $x, y \in H$ ;
- (p<sub>5</sub>)  $\|P_Cx - z\|^2 \leq \|x - z\|^2 - \|P_Cx - x\|^2$  for all  $x \in H, z \in C$ .

In a Hilbert space  $H$ , the next following facts are well known.

**Proposition 2.2**  $\forall x, y \in H, \forall t \in \mathbb{R}$ ,

- (i)  $\|x \pm y\|^2 = \|x\|^2 \pm 2\langle x, y \rangle + \|y\|^2$ ;
- (ii)  $\|tx + (1 - t)y\|^2 = t\|x\|^2 + (1 - t)\|y\|^2 - t(1 - t)\|x - y\|^2$ .

For the SFP, we assume that the following conditions are satisfied in a Hilbert space [5]:

- (i) The solution set of the SFP is nonempty.
- (ii) The set  $C$  is given by

$$C = \{x \in H_1 | c(x) \leq 0\},$$

where  $c : H_1 \rightarrow \mathbb{R}$  is a convex function and  $C$  is nonempty.

The set  $Q$  is given by

$$Q = \{y \in H_2 | q(y) \leq 0\},$$

where  $q : H_2 \rightarrow \mathbb{R}$  is a convex function and  $Q$  is nonempty.

(iii) Assume that both  $c$  and  $q$  are subdifferentiable on  $H_1$  and  $H_2$ , respectively. The sub-differentials are defined as follows:

$$\partial c(x) = \{\xi \in H_1 | c(z) \geq c(x) + \langle \xi, z - x \rangle \text{ for all } z \in H_1\} \neq \emptyset,$$

for all  $x \in C$ , and

$$\partial q(y) = \{\eta \in H_2 | q(u) \geq q(y) + \langle \eta, u - y \rangle \text{ for all } u \in H_2\} \neq \emptyset,$$

for all  $y \in Q$ .

(iv) We also assume that  $\partial c$  and  $\partial q$  are bounded on bounded sets. The set  $C_n$  is given by

$$C_n = \{x \in H_1 | c(x_n) + \langle \xi_n, x - x_n \rangle \leq 0\},$$

where  $\xi_n \in \partial c(x_n)$ . The set  $Q_n$  is constructed by

$$Q_n = \{y \in H_2 | q(Ax_n) + \langle \eta_n, y - Ax_n \rangle \leq 0\},$$

where  $\eta_n \in \partial q(Ax_n)$ .

It is easily seen that  $C \subseteq C_n$  and  $Q \subseteq Q_n$  for all  $n$ .

The relaxed CQ algorithm [4] can be seen as a special case of the classical gradient projection method (GPM). To see this, we can consider the following convex minimization problem:

$$\min_{x \in C_n} f_n(x). \tag{2.1}$$

It is well known that  $\hat{x} \in C_n$  is a solution of problem (2.1) if and only if

$$\langle \nabla f_n(\hat{x}), x - \hat{x} \rangle \geq 0, \quad \forall x \in C_n. \tag{2.2}$$

Also, we know that (2.2) holds true if and only if

$$\hat{x} = P_{C_n}(I - \tau \nabla f_n)\hat{x}, \quad \forall \tau > 0.$$

Therefore, we use the GPM to solve the SFP, for any  $x_0 \in H_1$ ,

$$x_{n+1} = P_{C_n}[x_n - \tau_n \nabla f_n(x_n)], \quad n \geq 0, \tag{2.3}$$

where  $\tau_n \in (0, 2/L)$ , while  $L$  is the Lipschitz constant of  $\nabla f_n$ . Noting that  $L = \|A\|^2$ , we see that (2.3) is exactly the relaxed CQ algorithm (1.3).

**Proposition 2.3** (see [22]) *Let  $\{\alpha_n\}$  be a sequence of nonnegative real numbers such that*

$$\alpha_{n+1} \leq (1 - t_n)\alpha_n + t_n b_n, \quad n \geq 1,$$

where  $\{t_n\}$  is a sequence in  $(0, 1)$  and  $b_n \in \mathbb{R}$  such that

- (i)  $\sum_{n=1}^\infty t_n = \infty$ ;
- (ii)  $\overline{\lim}_n b_n \leq 0$  or  $\sum_{n=1}^\infty |t_n b_n| < \infty$ .

Then  $\alpha_n \rightarrow 0$ .

**Proposition 2.4** ([23, 24]) *For some countable index set  $\Lambda$ , we denote by  $\ell_p = \ell_p(\Lambda)$ ,  $1 \leq p \leq \infty$ . Soft-thresholding leads to the unique minimizer of a functional combining  $\ell_2$  and  $\ell_1$ -norms,*

$$S_\mu(a) = \arg \min_{x \in \ell_2(\Lambda)} (\|x - a\|^2 + 2\mu \|x\|_1), \tag{2.4}$$

where  $\mu$  is a certain positive number and  $S_\mu$  is the soft-thresholding operation defined by  $S_\mu(a)_i = S_\mu(a_i)$ ,  $i \in \Lambda$ , with

$$S_\mu(z) = \begin{cases} z - \mu, & z > \mu, \\ 0, & |z| \leq \mu, \\ z + \mu, & z < -\mu. \end{cases}$$

For the  $\ell_1$ -ball  $B_R = \{x \in \ell_2 : \|x\|_1 \leq R\}$  with  $R := \|x^*\|_1$ , where  $x^* \in \ell_2$  is the solution of problem (2.4), we replace the thresholding with the projection  $P_{B_R}$ , and with a slight abuse of notation, we denote  $P_{B_R}$  by  $P_R$ ; then we introduce two properties of  $\ell_2$ -projections onto  $\ell_1$ -balls [25].

**Lemma 2.1** *For any  $a \in \ell_2$  and for  $\mu > 0$ ,  $\|S_\mu(a)\|_1$  is a piecewise linear, continuous, decreasing function of  $\mu$ ; moreover, if  $a \in \ell_1$ , then  $\|S_0(a)\|_1 = \|a\|_1$  and  $\|S_\mu(a)\|_1 = 0$  for  $\mu \geq \max_i |a_i|$ .*

**Lemma 2.2** *If  $\|a\|_1 > R$ , then the  $\ell_2$  projection of  $a$  on the  $\ell_1$ -ball with radius  $R$  is given by  $P_R(a) = S_\mu(a)$ , where  $\mu$  (depending on  $a$  and  $R$ ) is chosen such that  $\|S_\mu(a)\|_1 = R$ . If  $\|a\|_1 \leq R$ , then  $P_R(a) = S_0(a) = a$ .*

Next, we discuss a method to compute  $\mu$ .

**Proposition 2.5** ([25]) *For any  $a \in \Omega \subseteq \ell_2$ ,  $\dim(\Omega) = n$ , sort the absolute values of the components of  $a$  by descending order, obtaining the rearranged sequence  $(a_i^*)_{i=1, \dots, n}$ . Then we perform a search to find  $k$  such that*

$$\|S_{a_k^*}(a)\|_1 = \sum_{i=1}^{k-1} (a_i^* - a_k^*) \leq R < \sum_{i=1}^k (a_i^* - a_{k+1}^*) = \|S_{a_{k+1}^*}(a)\|_1$$

or equivalently,

$$\|S_{a_k^*}(a)\|_1 = \sum_{i=1}^{k-1} i(a_i^* - a_{i+1}^*) \leq R < \sum_{i=1}^k i(a_i^* - a_{i+1}^*) = \|S_{a_{k+1}^*}(a)\|_1.$$

Set  $v := k^{-1}(R - \|S_{a_k^*}(a)\|_1)$ , and  $\mu := a_k^* + v$ , then

$$\begin{aligned} \|S_\mu(a)\|_1 &= \sum_{i \in \Omega} \max(|a_i| - \mu, 0) = \sum_{i=1}^k (a_i^* - \mu) \\ &= \sum_{i=1}^{k-1} (a_i^* - a_k^*) + kv = \|S_{a_k^*}(a)\|_1 + kv = R. \end{aligned}$$

### 3 CQ algorithms with two simpler variable stepsizes

In this section, two simpler variable stepsizes are proposed below. The advantages of the two stepsizes, comparing with (1.5) and (1.6), are that neither prior information about the matrix norm  $A$  nor any other conditions on  $Q$  and  $A$  are required.

#### 3.1 A simpler variable stepsize for CQ algorithm

We propose a new and simpler variable stepsize method for solving the feasibility problem. The algorithm is presented as follows.

**Algorithm 3.1** For any initial data  $x_0 \in H_1$ ,  $u \in H_1$  and  $u \neq 0$ , assume that the  $n$ th iterate  $x_n$  has been constructed; then we compute the  $(n + 1)$ th iterate  $x_{n+1}$  via the formula

$$\bar{x}_n = P_{C_n}(t_n u + (1 - t_n)x_n), \tag{3.1}$$

$$x_{n+1} = P_{C_n}(x_n - \tau_n \nabla f_n(x_n)), \tag{3.2}$$

where

$$\tau_n := \frac{\rho_n \|x_n - \bar{x}_n\|^2}{\|Ax_n - A\bar{x}_n\|^2}, \tag{3.3}$$

with  $\{\rho_n\} \subset (0, 2)$  and  $\{t_n\} \subset (0, 1)$ . If  $x_{n+1} = x_n$  or  $Ax_n = A\bar{x}_n$  for some  $n \geq 0$ , then  $x_n$  is a solution of SFP (1.1) and the iteration stops; otherwise, we set  $n := n + 1$  and go to (3.1) to compute the next iterate  $x_{n+2}$ .

**Remark 3.1** We can easily approximate the upper bound  $\lambda$  of the eigenvalue interval to the symmetric matrix  $A^T A$  from [1, 3], thus for any  $x_{n \geq 0}$ , we can obtain  $\tau_n \in (0, 2/\lambda) \subset (0, 2/L)$ , where  $L$  is the largest eigenvalue of  $A^T A$ .

Now we prove the convergence property of Algorithm 3.1.

**Theorem 3.1** If  $\Gamma \neq \emptyset$  and  $\liminf_n \tau_n(2 - \lambda \tau_n) \geq \sigma > 0$ , the sequence  $\{x_n\}$  generated by Algorithm 3.1 converges weakly to a solution of SFP (1.1).

*Proof* Let  $x^*$  be a solution of SFP, since  $C \subseteq C_n$ ,  $Q \subseteq Q_n$ , thus  $x^* = P_C(x^*) = P_{C_n}(x^*)$  and  $Ax^* = P_Q(Ax^*) = P_{Q_n}(Ax^*)$ . It shows that  $x^* \in C_n$  and  $\nabla f_n(x^*) = 0$  for all  $n = 0, 1, \dots$ , using (3.2) and (p<sub>4</sub>), we have

$$\begin{aligned} \|x_{n+1} - x^*\|^2 &= \|P_{C_n}(x_n - \tau_n \nabla f_n(x_n)) - x^*\|^2 \\ &\leq \|(x_n - x^*) - \tau_n \nabla f_n(x_n)\|^2 \end{aligned}$$

$$\begin{aligned}
 &= \|x_n - x^*\|^2 + \tau_n^2 \|\nabla f_n(x_n)\|^2 - 2\tau_n \langle x_n - x^*, \nabla f_n(x_n) \rangle \\
 &= \|x_n - x^*\|^2 + \tau_n^2 \|\nabla f_n(x_n)\|^2 \\
 &\quad - 2\tau_n \langle x_n - x^*, \nabla f_n(x_n) - \nabla f_n(x^*) \rangle \\
 &\leq \|x_n - x^*\|^2 + \tau_n^2 L \|(I - P_{Q_n})Ax_n\|^2 \\
 &\quad - 2\tau_n \langle Ax_n - Ax^*, (I - P_{Q_n})Ax_n - (I - P_{Q_n})Ax^* \rangle \\
 &\leq \|x_n - x^*\|^2 + \tau_n^2 L \|(I - P_{Q_n})Ax_n\|^2 \\
 &\quad - 2\tau_n \|(I - P_{Q_n})Ax_n - (I - P_{Q_n})Ax^*\|^2 \\
 &= \|x_n - x^*\|^2 + (\tau_n^2 L - 2\tau_n) \|(I - P_{Q_n})Ax_n\|^2. \tag{3.4}
 \end{aligned}$$

Combining with Remark 3.1, we know  $\tau_n^2 L - 2\tau_n < 0$ , thus it implies that the sequence  $\{\|x_n - x^*\|^2\}$  is monotonically decreasing and hence  $\{x_n\}$  is bounded. Consequently, from (3.4) we get

$$\lim_{n \rightarrow \infty} \|(I - P_{Q_n})Ax_n\|^2 = 0. \tag{3.5}$$

Assume that  $\bar{x}$  is an accumulation point of  $\{x_n\}$  and  $\{x_{n_i}\} \rightarrow \bar{x}$ ,  $\{x_{n_i}\}$  is a subsequence of  $\{x_n\}$ . Then from (3.5) it follows

$$\lim_{n_i \rightarrow \infty} \|(I - P_{Q_{n_i}})Ax_{n_i}\|^2 = 0. \tag{3.6}$$

Next we show  $\bar{x} \in \Gamma$ .

Firstly, we show  $\bar{x} \in C$ . We prove it from two cases.

*Case 1.*  $\lim_{n_i \rightarrow \infty} x_{n_i+1} \neq \bar{x}$ . Without loss of generality, we may assume  $\lim_{n_i \rightarrow \infty} x_{n_i+1} = \tilde{x} \neq \bar{x}$ . Set  $z_{n_i} = x_{n_i} - \tau_{n_i} \nabla f_{n_i}(x_{n_i})$ , so  $x_{n_i+1} = P_{C_{n_i}}(z_{n_i})$  is the solution of the following programming:

$$\min \frac{1}{2} \|z - z_{n_i}\|^2 \quad \text{s.t. } c(x_{n_i}) + \langle \xi_{n_i}, z - x_{n_i} \rangle \leq 0.$$

By the Kuhn-Tucker condition, there exists a nonnegative number  $v_{n_i}$  such that

$$x_{n_i+1} - z_{n_i} + v_{n_i} \xi_{n_i} = 0, \tag{3.7}$$

$$v_{n_i} (c(x_{n_i}) + \langle \xi_{n_i}, x_{n_i+1} - x_{n_i} \rangle) = 0. \tag{3.8}$$

If there exist infinite  $n_i$  such that  $v_{n_i} = 0$  or  $\xi_{n_i} = 0$ , from (3.7) and (3.8) it leads to  $\tilde{x} = \bar{x}$ , so the contradiction happens. Therefore,  $v_{n_i} > 0$  and  $\xi_{n_i} \neq 0$  for sufficiently large  $n_i$ . We go on to divide the discussion into two cases.

(1) If  $\inf\{\|\xi_{n_i}\|\} > 0$ , we may assume  $\xi_{n_i} \rightarrow \bar{\xi} \neq 0$ . Then  $\bar{\xi} \in \partial c(\bar{x})$  by the lower semicontinuity of  $\partial c(x)$ . From (3.7) we have  $v_{n_i} \rightarrow \langle \bar{\xi}, \tilde{x} - \bar{x} \rangle / \|\bar{\xi}\|^2 \triangleq \bar{v}$ . Thus we obtain from (3.7) and (3.8)

$$\tilde{x} - \bar{x} + \bar{v} \bar{\xi} = 0, \tag{3.9}$$

$$\bar{v} (c(\bar{x}) + \langle \bar{\xi}, \tilde{x} - \bar{x} \rangle) = 0. \tag{3.10}$$

They mean that  $\tilde{x} = P_{C(\bar{x})}(\bar{x})$ .

Therefore,  $\bar{x} \neq P_{C(\bar{x})}(\bar{x})$  by assumption. From (p<sub>5</sub>) we have

$$\|\tilde{x} - x^*\|^2 \leq \|\bar{x} - x^*\|^2 - \|\bar{x} - \tilde{x}\|^2 < \|\bar{x} - x^*\|^2.$$

Since  $\{\|x_n - x^*\|^2\}$  is decreasing, both  $\bar{x}$  and  $\tilde{x}$  are the accumulation points of  $\{x_n\}$ , then  $\|\tilde{x} - x^*\|^2 = \|\bar{x} - x^*\|^2$ . It is a contradiction, so this case will not occur.

(2) If  $\inf\{\|\xi_{n_i}\|\} = 0$ , then we get the lower semicontinuity of  $\partial c(x)$ . Since  $c(x)$  is convex, then  $\bar{x}$  is a minimizer of  $c(x)$  over  $H_1$ . Since  $c(x^*) \leq 0$ , then  $c(\bar{x}) \leq c(x^*) \leq 0$ . So  $\bar{x} \in C$ .

Case 2.  $\lim_{n_i \rightarrow \infty} x_{n_i+1} = \bar{x}$ . As in Case 1, one has (3.7) and (3.8) by the Kuhn-Tucker condition. If there exist infinite  $n_i$  s.t.  $\nu_{n_i} = 0$  or  $\xi_{n_i} = 0$ , then we have  $P_{C_{n_i}}(z_{n_i}) = z_{n_i}$ , so  $c(x_{n_i}) + \langle \xi_{n_i}, z_{n_i} - x_{n_i} \rangle \leq 0$ . Since

$$\lim_{n_i \rightarrow \infty} (z_{n_i} - x_{n_i}) = \lim_{n_i \rightarrow \infty} \tau_{n_i} A^T (P_{Q_{n_i}} - I) A x_{n_i} = 0,$$

then  $c(\bar{x}) \leq 0$ . Therefore, we have  $\bar{x} \in C$ .

Assume  $\nu_{n_i} > 0$  and  $\xi_{n_i} \neq 0$  for sufficiently large  $n_i$ . If  $\inf\{\|\xi_{n_i}\|\} = 0$ , such as above, it follows  $\bar{x} \in C$ . If  $\inf\{\|\xi_{n_i}\|\} > 0$ , similar to Case 1(1), it leads to  $\bar{x} = P_{C(\bar{x})}(\bar{x})$ , which implies  $c(\bar{x}) + \langle \bar{\xi}, \bar{x} - \bar{x} \rangle \leq 0$ . So  $\bar{x} \in C$ .

In summary, we can conclude  $\bar{x} \in C$ .

Secondly, we need to show  $A\bar{x} \in Q$ . From (3.5) we have

$$\lim_{n_i \rightarrow \infty} \|(I - P_{Q_{n_i}}) A x_{n_i}\|^2 = 0. \tag{3.11}$$

Since  $P_{Q_{n_i}}(A x_{n_i}) \in Q_{n_i}$ , we have

$$q(A x_{n_i}) + \langle \eta_{n_i}, P_{Q_{n_i}}(A x_{n_i}) - A x_{n_i} \rangle \leq 0.$$

Moreover, limiting the inequality and taking account of (3.11), we obtain that

$$q(A\bar{x}) \leq 0,$$

that is,  $A\bar{x} \in Q$ .

Therefore  $\bar{x}$  is a solution of SFP. Thus we may replace  $x^*$  in (3.4) with  $\bar{x}$ , and get  $\{\|x_n - \bar{x}\|\}$  is convergent. Because there exists a subsequence  $\{\|x_{n_i} - \bar{x}\|\}$  convergent to 0, then  $x_n \rightarrow \bar{x}$  as  $n \rightarrow \infty$ . □

### 3.2 The other simpler variable stepsize for CQ algorithm

In this part, we introduce the other simpler choice of the stepsize  $\tau_n$ , which also is a variable stepsize to CQ algorithm. Either combining with the relaxed CQ algorithm [4], we have the next algorithm.

**Algorithm 3.2** Choose the initial data  $\forall x_0 \in H_1$ , for  $u \in H_1$  and  $u \neq 0$ . Assume that the  $n$ th iterate  $x_n$  has been constructed; then we compute the  $(n + 1)$ th iterate  $x_{n+1}$  via the formula

$$\bar{x}_n = P_{C_n}(t_n u + (1 - t_n)x_n), \tag{3.12}$$

$$x_{n+1} = P_{C_n}(x_n - \tau_n \nabla f_n(x_n)), \tag{3.13}$$

where

$$\tau_n := \frac{\rho_n \|\bar{x}_n\|^2}{\|A\bar{x}_n\|^2}, \tag{3.14}$$

with  $\{\rho_n\} \subset (0, 2)$  and  $\{t_n\} \subset (0, 1)$ . If  $x_{n+1} = x_n$ , then stop and  $x_n$  is a solution of SFP (1.1); otherwise, back to (3.12) and continue to compute  $x_{n+2}$ .

Obviously, (3.14) is also consistent with Remark 3.1. Thus, similar to the proof of Theorem 3.1, we can deduce that Algorithm 3.2 converges weakly to a solution of SFP (1.1).

#### 4 Two general KM-CQ algorithms with generalized variable stepsize

In this section, we integrate the variable stepsizes from (1.5) to (1.8) and obtain a variable stepsize that can cover them. After that, we apply it to improve the algorithms presented in [17] and [18] and construct two algorithms for approximating some solution of (1.1).

Let  $\psi : C \rightarrow H_1$  be a  $\delta$ -contraction with  $\delta \in (0, 1)$ , let  $r : H_1 \rightarrow H_1 \setminus \Theta$  and  $q : H_1 \rightarrow H_2 \setminus \Theta$  be nonzero operators, where  $\Theta$  denotes the zero point.

##### 4.1 A generalized variable stepsize for a general KM-CQ algorithm

The next recursion not only possesses a more generalized adaptive descent step, but it also can be implemented easily by the relaxed method.

**Algorithm 4.1** Choose the initial data  $x_0 \in H_1$  arbitrarily. Assume that the  $n$ th iterate  $x_n$  has been constructed; then we compute the  $(n + 1)$ th iterate  $x_{n+1}$  via the formula

$$x_{n+1} = (1 - \beta_n)x_n + \beta_n P_{C_n} [\alpha_n \psi(x_n) + (1 - \alpha_n)U_n x_n], \tag{4.1}$$

where  $U_n x_n = (I - \tau_n A^T (I - P_{Q_n}) A)x_n$ ,  $\tau_n := \frac{\rho_n \|r(x_n)\|^2}{\|Ar(x_n)\|^2}$  or  $\tau_n := \frac{\rho_n \|p(x_n)\|^2}{\|A^T p(x_n)\|^2}$ ,  $\{\rho_n\} \subset (0, 2)$ ,  $\{t_n\} \subset (0, 1)$ ,  $\{\alpha_n\}$  and  $\{\beta_n\}$  are two real sequences in  $[0, 1]$ . If  $x_{n+1} = x_n$  for some  $n \geq 0$ , then  $x_n$  is a solution of SFP (1.1) and the iteration stops; otherwise, continue to compute  $x_{n+2}$ .

**Theorem 4.1** Suppose that the SFP is consistent, that is,  $\Gamma = C \cap A^{-1}(Q) \neq \emptyset$ ,  $\underline{\lim}_n \tau_n (2 - \lambda \tau_n) \geq \sigma > 0$ . Assume that the sequences  $\{\alpha_n\}$  and  $\{\beta_n\}$  satisfy the following conditions:

- (C<sub>1</sub>)  $\lim_{n \rightarrow \infty} \alpha_n = 0$  and  $\sum_{n=1}^{\infty} \alpha_n = \infty$ ;
- (C<sub>2</sub>)  $0 < \underline{\lim}_n \beta_n$ .

Then  $\{x_n\}$  defined by (4.1) converges strongly to  $x^* = P_{\Gamma} \psi x^*$ , which solves the following variational inequality:

$$\langle (\psi - I)x^*, y - x^* \rangle \leq 0, \quad \forall y \in \Gamma. \tag{4.2}$$

*Proof* Since  $P_{\Gamma} : H_1 \rightarrow \Gamma \subset C$  is nonexpansive and  $\psi : C \rightarrow H_1$  is  $\delta$ -contractive, therefore, we have  $P_{\Gamma} \psi : C \rightarrow C$  is a contraction with  $\delta \in (0, 1)$ . By the Banach contractive mapping principle, there exists a unique  $x^* \in C$  such that  $x^* = P_{\Gamma} \psi x^*$ . By virtue of (p<sub>1</sub>), we see that (4.2) holds true.

By virtue of  $x^*$  being a solution of the SFP,  $x^* \in C \cap A^{-1}(Q)$ , and  $C \subseteq C_n, Q \subseteq Q_n$ , then  $x^* = P_C(x^*) = P_{C_n}(x^*)$  and  $Ax^* = P_Q(Ax^*) = P_{Q_n}(Ax^*)$ . From (p<sub>4</sub>) we have

$$\begin{aligned}
 \|U_n x_n - x^*\|^2 &= \|x_n - \tau_n \nabla f_n(x_n) - x^*\|^2 \\
 &= \|x_n - x^*\|^2 + \|\tau_n \nabla f_n(x_n)\|^2 - 2\tau_n \langle x_n - x^*, \nabla f_n(x_n) \rangle \\
 &= \|x_n - x^*\|^2 + \tau_n^2 \|\nabla f_n(x_n)\|^2 \\
 &\quad - 2\tau_n \langle Ax_n - Ax^*, (I - P_{Q_n})Ax_n - (I - P_{Q_n})Ax^* \rangle \\
 &\leq \|x_n - x^*\|^2 + \tau_n^2 L \|(I - P_{Q_n})Ax_n\|^2 \\
 &\quad - 2\tau_n \|(I - P_{Q_n})Ax_n\|^2 \\
 &= \|x_n - x^*\|^2 + (\tau_n^2 L - 2\tau_n) \|(I - P_{Q_n})Ax_n\|^2.
 \end{aligned}
 \tag{4.3}$$

Obviously,  $\tau_n \in (0, 2/L)$  from Remark 3.1, then  $\tau_n^2 L - 2\tau_n < 0$ , in particular, we obtain

$$\|U_n x_n - x^*\| \leq \|x_n - x^*\|.
 \tag{4.4}$$

At this point, we can establish the boundedness of  $\{x_n\}$ . To see this, using (4.1) we have

$$\begin{aligned}
 \|x_{n+1} - x^*\| &= \|(1 - \beta_n)x_n + \beta_n P_{C_n}[\alpha_n \psi(x_n) + (1 - \alpha_n)U_n x_n] - x^*\| \\
 &\leq (1 - \beta_n)\|x_n - x^*\| + \beta_n \|P_{C_n}[\alpha_n \psi(x_n) + (1 - \alpha_n)U_n x_n] - x^*\| \\
 &\leq (1 - \beta_n)\|x_n - x^*\| + \beta_n \|\alpha_n \psi(x_n) + (1 - \alpha_n)U_n x_n - x^*\| \\
 &= (1 - \beta_n)\|x_n - x^*\| \\
 &\quad + \beta_n \|\alpha_n (\psi(x_n) - \psi(x^*)) + (1 - \alpha_n)(U_n x_n - x^*) + \alpha_n \psi(x^*) - \alpha_n x^*\| \\
 &\leq (1 - \beta_n)\|x_n - x^*\| + \alpha_n \beta_n \|\psi(x_n) - \psi(x^*)\| + (1 - \alpha_n)\beta_n \|U_n x_n - x^*\| \\
 &\quad + \alpha_n \beta_n \|\psi(x^*) - x^*\| \\
 &\leq (1 - (1 - \delta)\alpha_n \beta_n)\|x_n - x^*\| + \alpha_n \beta_n \|\psi(x^*) - x^*\| \\
 &= (1 - (1 - \delta)\alpha_n \beta_n)\|x_n - x^*\| + (1 - \delta)\alpha_n \beta_n \frac{\|\psi(x^*) - x^*\|}{1 - \delta} \\
 &\leq \max \left\{ \|x_0 - x^*\|, \frac{\|\psi(x^*) - x^*\|}{1 - \delta} \right\} = M,
 \end{aligned}$$

for all  $n \geq 0$ , which indicates  $\{x_n\}$  is bounded. Set  $z_n = P_{C_n}[\alpha_n \psi(x_n) + (1 - \alpha_n)U_n x_n]$ , thus  $\{z_n\}$  is also bounded.

Next, we prove  $x_n \rightarrow x^*$  ( $n \rightarrow \infty$ ). By virtue of (4.1), Proposition 2.1(p<sub>2</sub>) and (4.4), we have

$$\begin{aligned}
 \|x_{n+1} - x^*\|^2 &= \|(1 - \beta_n)x_n + \beta_n P_{C_n}[\alpha_n \psi(x_n) + (1 - \alpha_n)U_n x_n] - x^*\|^2 \\
 &\leq (1 - \beta_n)\|x_n - x^*\|^2 + \beta_n \|P_{C_n}[\alpha_n \psi(x_n) + (1 - \alpha_n)U_n x_n] - x^*\|^2,
 \end{aligned}
 \tag{4.5}$$

where we set

$$\begin{aligned} & \|P_{C_n}[\alpha_n \psi(x_n) + (1 - \alpha_n)U_n x_n] - x^*\|^2 \\ &= \|P_{C_n}[w_n] - x^*\|^2 \\ &= \langle P_{C_n}[w_n] - x^*, P_{C_n}[w_n] - x^* \rangle \\ &= \langle P_{C_n}[w_n] - w_n, P_{C_n}[w_n] - x^* \rangle + \langle w_n - x^*, P_{C_n}[w_n] - x^* \rangle, \end{aligned}$$

since  $\langle P_{C_n}[w_n] - w_n, P_{C_n}[w_n] - x^* \rangle \leq 0$ , we have

$$\begin{aligned} & \|P_{C_n}[w_n] - x^*\|^2 \\ & \leq \langle w_n - x^*, P_{C_n}[w_n] - x^* \rangle \\ &= \langle \alpha_n \psi(x_n) + (1 - \alpha_n)U_n x_n - x^*, P_{C_n}[w_n] - x^* \rangle \\ &= \langle \alpha_n(\psi(x_n) - \psi(x^*)) + (1 - \alpha_n)(U_n x_n - x^*) + \alpha_n(\psi(x^*) - x^*), P_{C_n}[w_n] - x^* \rangle \\ & \leq (\alpha_n \|\psi(x_n) - \psi(x^*)\| + (1 - \alpha_n)\|U_n x_n - x^*\|) \|P_{C_n}[w_n] - x^*\| \\ & \quad + \alpha_n \langle \psi(x^*) - x^*, P_{C_n}[w_n] - x^* \rangle \\ & \leq (1 - (1 - \delta)\alpha_n) \|x_n - x^*\| \|P_{C_n}[w_n] - x^*\| + \alpha_n \langle \psi(x^*) - x^*, P_{C_n}[w_n] - x^* \rangle \\ & \leq \frac{1 - (1 - \delta)\alpha_n}{2} \|x_n - x^*\|^2 + \frac{1}{2} \|P_{C_n}[w_n] - x^*\|^2 \\ & \quad + \alpha_n \langle \psi(x^*) - x^*, P_{C_n}[w_n] - x^* \rangle. \end{aligned}$$

Therefore,

$$\|P_{C_n}[w_n] - x^*\|^2 \leq (1 - (1 - \delta)\alpha_n) \|x_n - x^*\|^2 + 2\alpha_n \langle \psi(x^*) - x^*, P_{C_n}[w_n] - x^* \rangle. \tag{4.6}$$

Substituting (4.6) into (4.5) can yield

$$\begin{aligned} & \|x_{n+1} - x^*\|^2 \\ & \leq (1 - \beta_n) \|x_n - x^*\|^2 \\ & \quad + (1 - (1 - \delta)\alpha_n)\beta_n \|x_n - x^*\|^2 + 2\alpha_n \beta_n \langle \psi(x^*) - x^*, P_{C_n}[w_n] - x^* \rangle \\ & \leq (1 - (1 - \delta)\alpha_n \beta_n) \|x_n - x^*\|^2 + (1 - \delta)\alpha_n \beta_n \frac{2}{1 - \delta} \langle \psi(x^*) - x^*, P_{C_n}[w_n] - x^* \rangle. \end{aligned} \tag{4.7}$$

Since  $x^* \in C \subseteq C_n$ ,  $P_{C_n} : H_1 \rightarrow C \subseteq C_n$  and  $\psi : C \subseteq C_n \rightarrow H_1$ , then  $P_{C_n} \psi : C_n \rightarrow C_n$ ,  $x^* = P_{C_n} \psi x^*$ .

Due to the property of the projection ( $p_1$ ) in Proposition 2.1,

$$\begin{aligned} & \limsup_{n \rightarrow \infty} \langle \psi(x^*) - x^*, P_{C_n}[w_n] - x^* \rangle \\ &= \max_{P_{C_n}[w_n] \in C_n} \langle \psi(x^*) - P_{C_n} \psi(x^*), P_{C_n}[w_n] - P_{C_n} \psi(x^*) \rangle \leq 0. \end{aligned} \tag{4.8}$$

Applying (4.8) and Proposition 2.3 to (4.7), we deduce that  $x_n \rightarrow x^*$ .

Assume that  $\hat{x}$  is an accumulation point of  $\{x_n\}$  and  $x_{n_i} \rightarrow \hat{x}$ , where  $\{x_{n_i}\}_{i=1}^\infty$  is a subsequence of  $\{x_n\}$ . Next we will prove that  $\hat{x}$  is a solution of SFP.

As  $x_n \rightarrow x^*$ , we know  $x_n \rightarrow x^*$ , that is,

$$\lim_{n \rightarrow \infty} \|x_{n+1} - x_n\| = 0. \tag{4.9}$$

Therefore, limit (4.4), we can obtain

$$\lim_{n \rightarrow \infty} \|U_n(x_n) - x^*\| = 0.$$

Then, limit (4.3), we get

$$\lim_{n \rightarrow \infty} \|(I - P_{Q_n})Ax_n\| = 0. \tag{4.10}$$

On the one hand, we show  $\hat{x} \in C$ .

Notice that  $x_{n_i} \rightarrow \hat{x}$  and  $x_{n_{i+1}} - x_{n_i} \rightarrow 0$  ( $i \rightarrow \infty$ ). Since  $x_{n_{i+1}} \in C_{n_i}$ , then by virtue of the definition of  $C_{n_i}$ , we have

$$c(x_{n_i}) + \langle \xi_{n_i}, x_{n_{i+1}} - x_{n_i} \rangle \leq 0, \quad \forall i = 1, 2, \dots,$$

taking the limit and using (4.9), we obtain that

$$c(\hat{x}) \leq 0.$$

Hence, we get  $\hat{x} \in C$ .

On the other hand, we need to show  $A\hat{x} \in Q$ .

Since  $P_{Q_{n_i}}(Ax_{n_i}) \in Q_{n_i}$ , we have

$$q(Ax_{n_i}) + \langle \eta_{n_i}, P_{Q_{n_i}}(Ax_{n_i}) - Ax_{n_i} \rangle \leq 0,$$

taking  $n_i \rightarrow \infty$ , by virtue of (4.10), we deduce that

$$q(A\hat{x}) \leq 0,$$

that is,  $A\hat{x} \in Q$ .

Therefore,  $\hat{x}$  is a solution of SFP.

Thus we may replace  $x^*$  in (4.7) with  $\hat{x}$  and get  $\{\|x_n - \hat{x}\|\}$  is convergent. Because there exists a subsequence  $\{\|x_{n_i} - \hat{x}\|\}$  convergent to 0, then  $x_n \rightarrow \hat{x}$  as  $n \rightarrow \infty$ . □

### 4.2 The other extended algorithm

Let  $h : C \rightarrow H_1$  be a  $\kappa$ -contraction. Let  $B : H_1 \rightarrow H_1$  be a self-adjoint strongly positive bounded linear operator with coefficient  $\lambda \in (0, 1)$ , for  $\forall x \in H_1$ , there exists  $\langle Bx, x \rangle \geq \lambda \|x\|^2$ . Take a constant  $\sigma$  such that  $0 < \sigma\kappa < \lambda$ .

As  $B$  is self-adjoint, we have  $\|B\| = \sup_{\|x\|=1} \langle Bx, x \rangle$ .  $I - B$  is also self-adjoint, then

$$\begin{aligned} \|I - B\| &= \sup_{\|x\|=1} \langle (I - B)x, x \rangle = \sup_{\|x\|=1} \{ \|x\|^2 - \langle Bx, x \rangle \} \\ &\leq \sup_{\|x\|=1} \{ (1 - \lambda) \|x\|^2 \} \leq 1 - \lambda. \end{aligned}$$

In (4.1), we set  $\psi(x) = \sigma h(x) + (I - B)U_n x$ , thus

$$\begin{aligned} \|\psi(x) - \psi(y)\| &\leq \sigma \kappa \|x - y\| + \|I - B\| \|x - y\| \\ &\leq (\sigma \kappa + 1 - \lambda) \|x - y\|, \end{aligned}$$

for  $\forall x, y \in H_1$ , we know that  $\sigma \kappa + 1 - \lambda \in (0, 1)$ ,  $\psi : C \rightarrow H_1$  is still a contraction. Accordingly, we have the following extended algorithm that is a special case of Algorithm 4.1.

**Algorithm 4.2** Choose the initial data  $x_0 \in H_1$  arbitrarily. Assume that the  $n$ th iterate  $x_n$  has been constructed; then we compute the  $(n + 1)$ th iterate  $x_{n+1}$  via the formula

$$x_{n+1} = (1 - \beta_n)x_n + \beta_n P_{C_n} [\alpha_n \sigma h(x_n) + (I - \alpha_n B)U_n x_n], \tag{4.11}$$

where  $U_n x_n = (I - \tau_n A^T (I - P_{Q_n}) A)x_n$ ,  $\tau_n := \frac{\rho_n \|r(x_n)\|^2}{\|Ar(x_n)\|^2}$  or  $\tau_n := \frac{\rho_n \|p(x_n)\|^2}{\|A^T p(x_n)\|^2}$ ,  $\tau_n \in (0, 2/L)$ ,  $\{\rho_n\} \subset (0, 2)$ ,  $\{t_n\} \subset (0, 1)$ ,  $\{\alpha_n\}$  and  $\{\beta_n\}$  are two real sequences in  $[0, 1]$ . If  $x_{n+1} = x_n$ , then stop and  $x_n$  is a solution of SFP (1.1); otherwise, continue to compute  $x_{n+2}$ .

**Theorem 4.2** Suppose that the SFP is consistent, that is,  $\Gamma = C \cap A^{-1}(Q) \neq \emptyset$ ,  $\underline{\lim}_n \tau_n (2 - \lambda \tau_n) \geq \sigma > 0$ , assume that the sequences  $\{\alpha_n\}$  and  $\{\beta_n\}$  satisfy the following conditions:

- (C<sub>1</sub>)  $\lim_{n \rightarrow \infty} \alpha_n = 0$  and  $\sum_{n=1}^{\infty} \alpha_n = \infty$ ;
- (C<sub>2</sub>)  $0 < \underline{\lim}_n \beta_n$ .

Then  $\{x_n\}$  defined by (4.11) converges strongly to  $x^* = P_{\Gamma} [\sigma h(x^*) + (I - B)U_n x^*]$ , which solves the following variational inequality:

$$\langle \sigma h(x^*) - B(x^*), y - x^* \rangle \leq 0, \quad \forall y \in \Gamma.$$

### 5 Numerical experiments and results

This section considers two numerical experiments to illustrate the performance of the above proposed variable stepsizes in CQ algorithm. Firstly, we see that a great amount of problems in signal and image processing can be seen as estimating  $x \in \mathbb{R}^N$  from the linear observation model

$$y = Ax + \varepsilon, \tag{5.1}$$

where  $y \in \mathbb{R}^M$  is the observed or measured data with noisy  $\varepsilon$ .  $A : \mathbb{R}^N \rightarrow \mathbb{R}^M$  denotes the bounded linear observation or measurement operator. Sometimes, the range of  $A$  may not be closed in most inverse problems, therefore, if  $A$  is ill-conditioned, the problem will be ill-posed.

If  $x$  is a sparse expansion, finding the solutions of (5.1) can be seen as finding a solution to the least-square problem

$$\min_{x \in \mathbb{R}^N} \frac{1}{2} \|y - Ax\|^2 \quad \text{subject to } \|x\|_1 < t \quad (5.2)$$

for any real number  $t > 0$ .

Problem (5.2) is a particular case of SFP (1.1) where  $C = \{x \in \mathbb{R}^N : \|x\|_1 \leq t\}$  and  $Q = \{y\}$ , i.e., find  $\|x\|_1 \leq t$  such that  $Ax = y$ . Therefore, CQ algorithm can be applied to solve (5.2). From Propositions 2.4 and 2.5 the projection onto  $C$  can be easily computed [15], while Lemmas 2.1 and 2.2 show the special situation of Proposition 2.4.

Next, following the experiments in [15, 26], we choose two particular problems, i.e., the compressed sensing and the image deconvolution, which are covered by (5.1). The experiments compare the performances of the proposed stepsizes of the CQ algorithm in this paper with the stepsizes in [15] and [16].

### 5.1 Compressed sensing

We consider a typical compressed sensing model, where a sparse signal recovery problem with a signal  $x \in \mathbb{R}^N$ , and  $N = 2^{12}$ . This original signal  $x$  contains only  $m = 50$  spikes with amplitude  $\pm 1$ , and the spikes are located at random, see the top of Figure 1.  $x$  is being reconstructed from  $M = 2^{10}$  measurements, thus  $A$  is a  $M \times N$  matrix randomly obtained with independent samples of a orthonormalized standard Gaussian distribution, and the noisy  $\varepsilon$  is with variance  $\sigma_\varepsilon^2 = 10^{-4}$ . To (5.2), we also set  $t=50$ .

For stepsizes (1.5) and (1.6) in [15] and [16], respectively, we still consider the stepsize with constant  $\rho = 2$ ,  $\omega_n = (n + 2)^{-5}$ , while for stepsizes (3.3) and (3.14), we set  $\rho = 1$ . For (3.1) and (3.12), we set  $t = 0.1$  and  $u = \text{rand}(N, 1) \in (0, 1)$ . For (4.1) we use (3.1) and (3.3), and set  $\alpha_n = (n + 2)^{-1}$ ,  $\beta_n = 1 - (n + 2)^{-1}$  and  $\psi \equiv 0$  as its special case. All the processes are started with the initial signal  $x_0 = 0$  and finished with the stop rule

$$\|x_{n+1} - x_n\| / \|x_n\| < 10^{-3}.$$

We also calculated the mean squared error (MSE) for the results

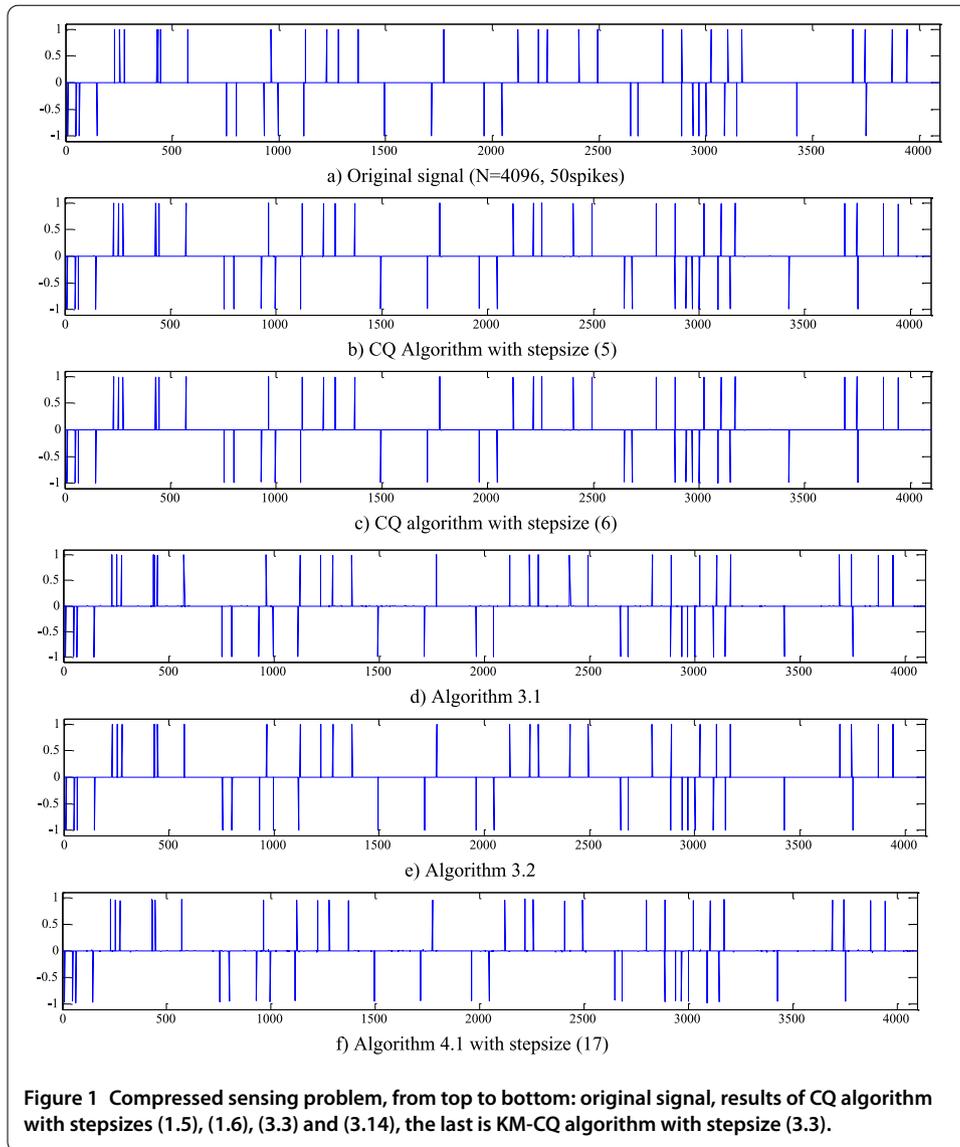
$$MSE = (1/N) \|x^* - x\|,$$

where  $x^*$  is an estimated signal of  $x$ .

The simulated results of different algorithms with different steps can be seen in Figure 1 and Table 1. Algorithms 3.1 and 3.2 have less iteration steps and smaller MSE, especially for Algorithm 3.1. Thus, we see that stepsizes (3.3) and (3.14) not only have simpler formats than before, but also can make CQ algorithms have faster iteration and better restored precision.

### 5.2 Image deconvolution

In this subsection, we apply the CQ algorithms in the paper to image deconvolution. The observation model can also be described as (5.1), we wish to estimate an original image  $x$  from an observation  $y$ , while matrix  $A$  represents the observation operator, and  $\varepsilon$  is a sample of a zero-mean white Gaussian field of variance  $\sigma$ . For the 2D image deconvolution



**Table 1** Results of different stepsizes and algorithms for Figure 1

Algorithms	MSE	$n$	CPU time (s)
CQ with (1.5)	2.8021e-005	66	0.8504
CQ with (1.6)	2.844e-005	66	0.8579
3.1	5.1714e-006	25	0.6858
3.2	1.5253e-006	64	1.5685

problem,  $A$  is a block-circulant matrix with circulant blocks [27]. We stress that the goal of these experiments is not to assess the restored precision of the algorithms, but to apply the algorithms in paper to solve this particular SFP, then compare the iterative speed and restored precision of the proposed stepsizes against the CQ algorithms.

According to papers [26, 27], we also take the well-known Cameraman image. In the experiments, we employ Haar wavelets, and the blur point spread functions are uniform blur with size  $9 \times 9$ ,  $h_{ij} = (1 + i^2 + j^2)^{-1}$ , for  $i, j = -4, \dots, 4$  and for  $i, j = -7, \dots, 7$ . The noise variance is  $\sigma^2 = 0.308, 2$  and  $8$ , respectively. We have  $N = M = 256^2$ , then the block-circulant

**Table 2 Results for the restorations of different stepsizes and algorithms**

Blur kernel	$\sigma^2$	Algorithms	SNR (dB)	$n$	CPU time (s)
$9 \times 9$ uniform	0.308	CQ with (1.5)	16.1802	52	3.3883
		CQ with (1.6)	16.1722	48	3.1636
		3.1	14.5266	19	1.3847
		3.2	14.5265	19	1.3199
		4.1 with (3.3)	14.2464	34	2.4280
$h_{ij} = (1 + i^2 + j^2)^{-1}$ for $i, j = -4, \dots, 4$	2	CQ with (1.6)	22.6184	19	1.3007
		CQ with (1.5)	22.4329	17	1.1718
		3.1	19.8401	14	1.0746
		3.2	19.8401	14	1.0266
		4.1 with (3.3)	19.4912	33	2.3306
$h_{ij} = (1 + i^2 + j^2)^{-1}$ for $i, j = -7, \dots, 7$	8	CQ with (1.6)	12.7305	39	2.6739
		CQ with (1.5)	14.6363	42	2.8604
		3.1	19.3561	18	1.3390
		3.2	19.3560	18	1.3057
		4.1 with (3.3)	18.6140	33	2.3628

matrix  $A$  can be constructed by the blur point spread functions, and  $A$  may be very ill-conditioned. Set all the threshold values  $\mu = 0.25$ ,  $t$  is the sum of all the pixel values in the original image. Moreover, we use  $y = IFFT(FFT(A) * FFT(x)) + \varepsilon$  to obtain the observation, where  $FFT$  is the fast Fourier transform,  $IFFT$  is the inverse fast Fourier transform. Other settings in the above stepsizes and algorithms are the same as in 4.1. We set the initial image  $x_0 = 0$  and also follow the stop rule  $\|x_{n+1} - x_n\| / \|x_n\| < 10^{-3}$ .

The results of iteration steps, CPU time and the SNR improvements are presented in Table 2. It also testifies that the proposed stepsizes and algorithms in this paper can give better performance.

### 6 Conclusions and discussion

In this paper, we have proposed two simpler variable stepsizes for the CQ algorithm. Compared with the other related variable stepsizes, they also need not to compute the largest eigenvalue of  $A$  and can be calculated easily. Furthermore, we also presented a more general KM-CQ algorithm with generalized variable stepsizes. As a special case, we deduced another general format. Obviously, both the general algorithms with the generalized variable stepsizes can solve the SFP and some special variational inequality problem better. The corresponding weak and strong convergence properties have also been established. In the experiments, through the compressed sensing and image deconvolution models, we compare the proposed stepsizes with the former ones, the results obtained from the proposed stepsizes and algorithms appear to be significantly better.

We should notice that the values of parameter  $\rho_n$  are fixed in the above experiments. Actually, a different value of  $\rho_n$  can also affect the convergence speed of the algorithms. Therefore, our future work is to find the method to choose a self-adaptive sequence  $\{\rho_n\}$ .

#### Acknowledgements

The authors would like to thank the associate editor and the referees for their comments and suggestions. The research was supported by Professor Haiyun Zhou.

#### Competing interests

The authors declare that they have no competing interests.

### Authors' contributions

All authors took equal roles in deriving results and writing of this paper. All authors have read and approved the final manuscript.

### Publisher's Note

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

Received: 16 March 2017 Accepted: 26 May 2017 Published online: 12 June 2017

### References

1. Byrne, C: Iterative oblique projection onto convex sets and the split feasibility problem. *Inverse Probl.* **18**, 441-453 (2002)
2. Censor, Y, Elfving, T: A multiprojection algorithm using Bregman projections in a product space. *Numer. Algorithms* **8**, 221-239 (1994)
3. Byrne, CL: A unified treatment of some iterative algorithms in signal processing and image reconstruction. *Inverse Probl.* **20**, 103-120 (2004)
4. Yang, QZ: The relaxed CQ algorithm solving the split feasibility problem. *Inverse Probl.* **20**, 1261-1266 (2004)
5. Xu, HK: Iterative methods for the split feasibility problem in infinite-dimensional Hilbert spaces. *Inverse Probl.* **26**, 1-17 (2010)
6. Qu, B, Xiu, N: A note on the CQ algorithm for the split feasibility problem. *Inverse Probl.* **21**, 1655-1665 (2005)
7. Qu, B, Xiu, N: A new halfspace-relaxation projection method for the split feasibility problem. *Linear Algebra Appl.* **428**, 1218-1229 (2008)
8. Wang, W, Gao, Y: A modified algorithm for solving the split feasibility problem. *Int. Math. Forum* **4**, 1389-1396 (2009)
9. Wang, Z, Yang, Q: The relaxed inexact projection methods for the split feasibility problem. *Appl. Math. Comput.* **217**, 5347-5359 (2011)
10. Li, M: Improved relaxed cq methods for solving the split feasibility problem, advanced modeling and optimization. *Appl. Math. Comput.* **13**, 305-317 (2011)
11. Abdellah, B, Muhammad, AN: On descent-projection method for solving the split feasibility problems. *J. Glob. Optim.* **54**, 627-639 (2012)
12. Dong, Q, Yao, Y, He, S: Weak convergence theorems of the modified relaxed projection algorithms for the split feasibility problem in Hilbert spaces. *Optim. Lett.* **8**, 1031-1046 (2014). doi:10.1007/s11590-013-0619-4
13. Yang, Q: On variable-step relaxed projection algorithm for variational inequalities. *J. Math. Anal. Appl.* **302**, 166-179 (2005)
14. Wang, F, Xu, H, Su, M: Choices of variable of the steps of the cq algorithm for the split feasibility problem. *Fixed Point Theory* **12**, 489-496 (2011)
15. López, G, Martín-Máquez, V, Wang, F: Solving the split feasibility problem without prior knowledge of matrix norms. *Inverse Probl.* **28**, 1-18 (2012)
16. Zhou, HY, Wang, PY: Adaptively relaxed algorithms for solving the split feasibility problem with a new stepsize. *J. Inequal. Appl.* **2014**, 448 (2014)
17. Yao, Y, Postolache, M, Liou, Y: Strong convergence of a self adaptive method for the split feasibility problem. *Fixed Point Theory Appl.* **2013**, 201 (2013)
18. Zhou, HY, Wang, PY: Some remarks on the paper "strong convergence of a self-adaptive method for the split feasibility problem". *Numer. Algorithms* **70**, 333-339 (2015)
19. Dang, Y, Gao, Y: The strong convergence of a km-cq-like algorithm for a split feasibility problem. *Inverse Probl.* **27**, 015007 (2011)
20. Yu, X, Shahzad, N, Yao, Y: Implicit and explicit algorithms for solving the split feasibility problem. *Optim. Lett.* **6**, 1447-1462 (2012)
21. Byrne, C: A unified treatment of some iterative algorithms in signal processing and image reconstruction. *Inverse Probl.* **20**, 103-120 (2004)
22. Xu, H: Iterative algorithms for nonlinear operators. *J. Lond. Math. Soc.* **66**, 240-256 (2002)
23. Deubachies, I, Defrise, M, DeMol, C: An iterative thresholding algorithm for linear inverse problems. *Commun. Pure Appl. Math.* **57**, 1413-1457 (2004)
24. Chambolle, A, DeVore, R, Lee, N: Nonlinear wavelet image processing: variational problems, compression, and noise removal through wavelet shrinkage. *IEEE Trans. Image Process.* **7**, 319-335 (1998)
25. Daubechies, I, Fornasier, M, Loris, I: Accelerated projected gradient method for linear inverse problems with sparsity constraints. *J. Fourier Anal. Appl.* **14**, 764-792 (2008)
26. Figueiredo, M, Nowak, R, Wright, S: A gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems. *IEEE J. Sel. Top. Signal Process.* **1**, 586-598 (2007)
27. Figueiredo, M, Nowak, R: A bound optimization approach to wavelet-based image deconvolution. In: *IEEE International Conference on Image Processing - ICIP'2005*, pp. 1-4 (2005)