RESEARCH   **Open Access**

CrossMark

Simultaneous and semi-alternating projection algorithms for solving split equality problems

Qiao-Li Dong[*] and Dan Jiang

[*]Correspondence:
dongql@lsec.cc.ac.cn
Tianjin Key Laboratory for Advanced
Signal Processing, College of
Science, Civil Aviation University of
China, Tianjin, 300300, China

Abstract

In this article, we first introduce two simultaneous projection algorithms for solving the split equality problem by using a new choice of the stepsize, and then propose two semi-alternating projection algorithms. The weak convergence of the proposed algorithms is analyzed under standard conditions. As applications, we extend the results to solve the split feasibility problem. Finally, a numerical example is presented to illustrate the efficiency and advantage of the proposed algorithms.

MSC: 90C47; 49J35

Keywords: simultaneous projection algorithm; semi-alternating projection algorithm; maximal monotone operator; split equality problem

1 Introduction

Let H_1, H_2 and H_3 be real Hilbert spaces, let $C \subseteq H_1$ and $Q \subseteq H_2$ be two nonempty closed convex sets, and let $A : H_1 \to H_3$ and $B : H_2 \to H_3$ be two bounded linear operators.

In this article, we consider the classical split equality problem (SEP), which was first introduced by Moudafi [1]. The SEP can mathematically be formulated as follows:

$$\text{Find} \quad x \in C, y \in Q \quad \text{such that} \quad Ax = By. \tag{1}$$

Throughout this paper, assume that SEP (1) is consistent and denote by

$$\Gamma = \{x \in C, y \in Q : Ax = By\}$$

the solution of SEP (1). Then Γ is closed, convex and nonempty.

The split equality problem (1) is actually an optimization problem with weak coupling in the constraint (see [1] for details) and its interest covers many situations, for instance, in domain decomposition for PDEs, game theory and intensity-modulated radiation therapy (IMRT). In domain decomposition for PDEs, this equals to the variational form of a PDE in a domain that can be decomposed into two non-overlapping subdomains with a common interface (see, e.g., [2]). In decision sciences, this allows to consider agents who interplay only via some components of their decision variables (see, e.g., [3]). In IMRT,

this amounts to envisaging a weak coupling between the vector of doses absorbed in all voxels and that of the radiation intensity (see [4] for further details). Attouch [5] pointed out more applications of the SEP in optimal control theory, surface energy and potential games, whose variational form can be seen as a SEP.

Next we present an example, in which a separable optimization problem can be rewritten as a split equality problem.

Example 1.1 Consider the separable optimization problem

$$
\begin{aligned}
&\text{minimize} f(x) + g(y) \\
&\quad \text{subject to } Ax = By,
\end{aligned}
\tag{2}
$$

with $x \in R^N$ and $y \in R^M$, where $A \in R^{J \times N}$ and $B \in R^{J \times M}$. Assume that f and g are convex and the solution set of problem (2) is nonempty.

Set $C = \text{argmin}\{f(x) \mid x \in R^N\}$ and $Q = \text{argmin}\{g(y) \mid y \in R^M\}$. Then the optimization problem (2) equals to the following split equality problem:

$$
\text{Find} \quad x \in C, y \in Q \quad \text{such that} \quad Ax = By.
\tag{3}
$$

A great deal of literature on algorithms for solving SEP has been published, most of which are projection methods [1–3, 6–11]. Based on the classical projection gradient algorithm, Byrne and Moudafi [12] introduced the following algorithm, which is also called the simultaneous iterative method [13]:

$$
\begin{cases}
x^{k+1} = P_C(x^k - \beta_k A^*(Ax^k - By^k)), \\
y^{k+1} = P_Q(y^k + \beta_k B^*(Ax^k - By^k)),
\end{cases}
\tag{4}
$$

where $\beta_k \in (\varepsilon, 2/(\lambda_A + \lambda_B) - \varepsilon)$, λ_A and λ_B are the operator (matrix) norms $\|A\|$ and $\|B\|$ (or the largest eigenvalues of A^*A and B^*B), respectively. To determine stepsize β_k, one needs first to calculate (or estimate) the operator norms $\|A\|$ and $\|B\|$. In general, it is difficult or even impossible. On the other hand, even if we know the norm of A and B, the algorithm (4) method with fixed stepsize may be slow.

In order to deal with this, the authors [9] introduced a self-adaptive projection algorithm, in which the stepsize is computed by using an Armijo search.

Define the function $F : H_1 \times H_2 \to H_1$ by

$$
F(x, y) = A^*(Ax - By),
$$

and the function $G : H_1 \times H_2 \to H_2$ by

$$
G(x, y) = B^*(By - Ax).
$$

The self-adaptive projection algorithm in [9] is defined as follows.

Algorithm 1.1 Given constants $\sigma_0 > 0$, $\alpha \in (0, 1)$, $\theta \in (0, 1)$ and $\rho \in (0, 1)$. Let $x^0 \in H_1$ and $y^0 \in H_2$ be taken arbitrarily. For $k = 0, 1, 2, \ldots$, compute

$$\begin{cases} u^k = P_C(x^k - \beta_k F(x^k, y^k)), \\ v^k = P_Q(y^k - \beta_k G(x^k, y^k)), \end{cases} \tag{5}$$

where β_k is chosen to be the largest $\beta \in \{\sigma_k, \sigma_k \alpha, \sigma_k \alpha^2, \ldots\}$ satisfying

$$\beta^2 \big(\big\| F(x^k, y^k) - F(u^k, v^k) \big\|^2 + \big\| G(x^k, y^k) - G(u^k, v^k) \big\|^2 \big)$$
$$\leq \theta^2 \big(\big\| x^k - u^k \big\|^2 + \big\| y^k - v^k \big\|^2 \big). \tag{6}$$

Compute

$$\begin{cases} x^{k+1} = P_C(x^k - \beta_k F(u^k, v^k)), \\ y^{k+1} = P_Q(y^k - \beta_k G(u^k, v^k)). \end{cases} \tag{7}$$

If

$$\beta_k^2 \big(\big\| F(x^k, y^k) - F(x^{k+1}, y^{k+1}) \big\|^2 + \big\| G(x^k, y^k) - G(x^{k+1}, y^{k+1}) \big\|^2 \big)$$
$$\leq \rho^2 \big(\big\| x^k - x^{k+1} \big\|^2 + \big\| y^k - y^{k+1} \big\|^2 \big), \tag{8}$$

then set $\sigma_k = \sigma_0$; otherwise, set $\sigma_k = \beta_k$.

In fact, Algorithm 1.1 can be seen as an extension of the classical extragradient method first proposed by Korpelevich [14]. Notice that, in Algorithm 1.1, the stepsize of the prediction (5) and that of the correction (7) are equal. Thus these two steps seem to be 'symmetric'.

Recently, Chuang and Du [15] proposed the following projection algorithm (which is called the hybrid projected Landweber algorithm).

Algorithm 1.2 Given constants $\sigma > 0$, $\alpha \in (0, 1)$ and $\theta \in (0, 1)$, let $x^0 \in H_1$ and $y^0 \in H_2$ be taken arbitrarily. For $k = 0, 1, 2, \ldots$, compute

$$\begin{cases} u^k = P_C(x^k - \beta_k F(x^k, y^k)), \\ v^k = P_Q(y^k - \beta_k G(x^k, y^k)), \end{cases}$$

where β_k is chosen via (6) and (8). Compute next iterates x^{k+1} and y^{k+1} by

$$\begin{cases} x^{k+1} = P_C(x^k - \rho_k c_k), \\ y^{k+1} = P_Q(y^k - \rho_k d_k), \end{cases} \tag{9}$$

where

$$\begin{cases} c_k := (x^k - u^k) - \beta_k (F(x^k, y^k) - F(u^k, v^k)); \\ d_k := (y^k - v^k) - \beta_k (G(x^k, y^k) - G(u^k, v^k)), \end{cases}$$

and

$$\rho_k := \frac{\langle x^k - u^k, c_k \rangle + \langle y^k - v^k, d_k \rangle}{\|c_k\|^2 + \|d_k\|^2}. \tag{10}$$

Note that Algorithm 1.2 with $\rho_k \equiv 1$ in (9) can be seen as a special case of Tseng's method [8, 16]. The projections in the second step of Tseng's method are made onto two nonempty closed convex sets $X \subseteq H_1$ and $Y \subseteq H_2$, other than C and Q. X and Y can be any sets such that the intersections of X and C (and Y and Q) are nonempty, and they may be taken to have simple structures so that the projections onto them are easy to calculate.

Chuang and Du [15] proved the convergence of Algorithm 1.2 and also presented the convergence property of Algorithm 1.2 as follows:

$$\begin{aligned}
\|x^{k+1} - x^*\|^2 &+ \|y^{k+1} - y^*\|^2 \\
&\le \|x^k - x^*\|^2 + \|y^k - y^*\|^2 - \rho_k^2 (\|c_k\|^2 + \|d_k\|^2),
\end{aligned} \tag{11}$$

where $(x^*, y^*) \in \Gamma$.

The stepsize β_k in Algorithms 1.1 and 1.2 is obtained through the Armijo search (6). In general, the computational cost of a self-adaptive algorithm is large, since one may need to calculate (5) several times to get the stepsize β_k.

To overcome this difficulty, the authors [17] introduced a projection algorithm for which the stepsize does not depend on the operator norms $\|A\|$ and $\|B\|$, and one can directly compute the stepsize instead of using the Armijo search.

Algorithm 1.3 Choose initial guesses $x^0 \in H_1, y^0 \in H_2$ arbitrarily. Assume that the kth iterate $x^k \in C$, $y^k \in Q$ has been constructed and $Ax^k - By^k \ne 0$; then we calculate the $(k+1)$th iterate (x^{k+1}, y^{k+1}) via the formula

$$\begin{cases} x^{k+1} = P_C(x^k - \beta_k A^*(Ax^k - By^k)), \\ y^{k+1} = P_Q(y^k + \beta_k B^*(Ax^k - By^k)), \end{cases} \tag{12}$$

where the stepsize β_k is chosen in such a way that

$$\beta_k = \sigma_k \min \left\{ \frac{\|Ax^k - By^k\|^2}{\|A^*(Ax^k - By^k)\|^2}, \frac{\|Ax^k - By^k\|^2}{\|B^*(Ax^k - By^k)\|^2} \right\}, \tag{13}$$

where $0 < \sigma_k < 1$. If $Ax^k - By^k = 0$, then $(x^{k+1}, y^{k+1}) = (x^k, y^k)$ is a solution of SEP (1) and the iterative process stops; otherwise, we set $k := k + 1$ and go onto (12) to evaluate the next iterate (x^{k+2}, y^{k+2}).

Note that the choice in (13) of the stepsize β_k is independent of the norms $\|A\|$ and $\|B\|$.

Polyak [18, 19] first introduced the inertial extrapolation algorithms, which were widely studied as an acceleration process. The authors [20] made an inertial modification for Algorithm 1.3 and introduced the following inertial projection methods for SEP.

Algorithm 1.4 Choose initial guesses $x^0, x^1 \in H_1$, $y^0, y^1 \in H_2$ arbitrarily. Compute

$$\begin{cases} (\bar{x}^k, \bar{y}^k) = (x^k, y^k) + \alpha_k(x^k - x^{k-1}, y^k - y^{k-1}), \\ x^{k+1} = P_C(\bar{x}^k - \beta_k A^*(A\bar{x}^k - B\bar{y}^k)), \\ y^{k+1} = P_Q(\bar{y}^k + \beta_k B^*(A\bar{x}^k - B\bar{y}^k)), \end{cases} \tag{14}$$

where $\alpha_k \in (0, 1)$ and the stepsize γ_k is chosen in the same way as (13).

They showed the weak convergence of Algorithm 1.4 under some conditions on the inertial parameter α_k.

In fact, Algorithm 1.4 can be seen as a FISTA (fast iterative shrinkage-thresholding algorithm) introduced by Beck and Teboulle [21] to solve the linear inverse problems, if we take the inertial parameter $\alpha_k = \frac{t_k - 1}{t_{k+1}}$, where $t_1 = 1$ and $t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2}$, $k \geq 1$, and choose a constant stepsize β_k or choose β_k via a backtracking stepsize rule. A shortcoming of the method of Beck and Teboulle is that they could not prove the convergence of the iterative sequence (x^k, y^k). Chambolle and Dossal [22] improved the choice of the inertial parameter, took $\alpha_k = \frac{k-1}{k+a}$, where $a > 2$, and presented the convergence of the iterative sequence (x^k, y^k).

In this paper, inspired by the work in [17, 23, 24], we introduce two simultaneous projection algorithms by improving the stepsizes β_k and ρ_k of the second step (7) and (9) in Algorithms 1.1 and 1.2, respectively. We also present two alternating projection algorithms, in which we take an alternating technique in the first step.

The structure of the paper is as follows. In the next section, we present some concepts and lemmas which will be used in the main results. In Section 3, two classes of projection algorithms are provided and their weak convergence is analyzed. In Section 4, we extend the results to the split feasibility problem. In the final section, some numerical results are provided, which show the advantages of the proposed algorithms.

2 Preliminaries

Let H be a real Hilbert space with the inner product $\langle \cdot, \cdot \rangle$ and the induced norm $\| \cdot \|$, and let D be a nonempty, closed and convex subset of H. We write $x^k \rightharpoonup x$ to indicate that the sequence $\{x^k\}_{k=0}^{\infty}$ converges weakly to x and $x^k \to x$ to indicate that the sequence $\{x^k\}_{k=0}^{\infty}$ converges strongly to x. Given a sequence $\{x^k\}_{k=0}^{\infty}$, denote by $\omega_w(x^k)$ its weak ω-limit set, that is, any $x \in \omega_w(x^k)$ such that there exists a subsequence $\{x^{k_j}\}_{j=0}^{\infty}$ of $\{x^k\}_{k=0}^{\infty}$ which converges weakly to x.

In this paper, an important tool of our work is the projection. Let H be a real Hilbert space and C be a closed convex subset of H. Recall that the `projection` from H onto C, denoted by P_C, is defined in such a way that, for each $x \in H$, $P_C(x)$ is the unique point in C such that

$$\|x - P_C(x)\| = \min\{\|x - z\| : z \in C\}.$$

The following two lemmas are useful characterizations of projections.

Lemma 2.1 ([25]) *Given $x \in H$ and $z \in C$. Then $z = P_C(x)$ if and only if*

$$\langle x - z, y - z \rangle \leq 0, \quad \forall y \in C.$$

Lemma 2.2 ([25, 26]) *For any $x, y \in H$ and $z \in C$, it holds*

  (i) $\|P_C(x) - P_C(y)\| \leq \|x - y\|$;

  (ii) $\|P_C(x) - z\|^2 \leq \|x - z\|^2 - \|P_C(x) - x\|^2$.

Definition 2.1 The `normal cone` of C at $v \in C$, denoted by $N_C(v)$, is defined as

$$N_C(v) := \{d \in H \mid \langle d, y - v \rangle \leq 0 \text{ for all } y \in C\}.$$

Definition 2.2 Let $A : H \rightrightarrows 2^H$ be a point-to-set operator defined on a real Hilbert space H. The operator A is called a `maximal monotone operator` if A is `monotone`, i.e.,

$$\langle u - v, x - y \rangle \geq 0 \quad \text{for all } u \in A(x) \text{ and } v \in A(y),$$

and the graph $G(A)$ of A,

$$G(A) := \{(x, u) \in H \times H \mid u \in A(x)\},$$

is not properly contained in the graph of any other monotone operator.

It is clear [27, Theorem 3] that a monotone mapping A is maximal if and only if, for any $(x, u) \in H \times H$, if $\langle u - v, x - y \rangle \geq 0$ for all $(v, y) \in G(A)$, then it follows that $u \in A(x)$.

Lemma 2.3 ([26]) *Let D be a nonempty, closed and convex subset of a Hilbert space H. Let (x^k) be a bounded sequence which satisfies the following properties:*

  (i) *every limit point of $\{x^k\}_{k=0}^{\infty}$ lies in D;*

  (ii) $\lim_{n \to \infty} \|x^k - x\|$ *exists for every $x \in D$.*

Then $\{x^k\}$ converges weakly to a point in D.

3 Main results

In this section, we present two classes of projection algorithms and establish their weak convergence under standard conditions.

3.1 Simultaneous projection algorithms

Let $S = C \times Q \in H := H_1 \times H_2$. Define $K = [A, -B] : H_1 \times H_2 \to H_1 \times H_2$, and let K^* be the adjoint operator of K, then the original problem (1) can be modified as

$$\text{Find} \quad z = (x, y) \in S \quad \text{such that} \quad Kw = 0. \tag{15}$$

Note that if the solution set of (15) is nonempty, it equals the following minimization problem:

$$\min_{z \in S} \frac{1}{2} \|Kz\|^2, \tag{16}$$

which is a standard (and a simple) problem from the convex optimization point of view. There are many methods for solving the minimization problem (16) such as the classical projection gradient method. Algorithm (4) (also Algorithm 1.3) is the exact projection gradient method when applied to (16).

Inspired by Cai [24] and Dong et al. [17], we propose two new simultaneous projection algorithms by improving the stepsizes in the second step of Algorithms 1.1 and 1.2.

Algorithm 3.1 Given constants $\sigma > 0$, $\alpha \in (0, 1)$, $\theta \in (0, 1)$ and $\rho \in (0, 1)$, let $z_0 = (x^0, y^0) \in H = H_1 \times H_2$ be taken arbitrarily.

For $k = 0, 1, 2, \ldots$, compute

$$w^k = P_S\big(z^k - \beta_k K^* K\big(z^k\big)\big), \tag{17}$$

where β_k is chosen to be the largest $\beta \in \{\sigma_k, \sigma_k\alpha, \sigma_k\alpha^2, \ldots\}$ satisfying

$$\beta \big\| K^* K\big(z^k\big) - K^* K\big(w^k\big) \big\| \leq \theta \big\| z^k - w^k \big\|. \tag{18}$$

Compute next iterates z^{k+1} by

$$z_{\mathrm{I}}^{k+1} = z^k - \gamma \rho_k d\big(z^k, w^k\big), \tag{19}$$

or

$$z_{\mathrm{II}}^{k+1} = P_S\big(z^k - \gamma \beta_k \rho_k K^* K\big(w^k\big)\big), \tag{20}$$

where $\gamma \in [0, 2)$,

$$d\big(z^k, w^k\big) := \big(z^k - w^k\big) - \beta_k\big(K^* K\big(z^k\big) - K^* K\big(w^k\big)\big), \tag{21}$$

and

$$\rho_k := \frac{\langle z^k - w^k, d(z^k, w^k)\rangle + \beta_k \|K(w^k)\|^2}{\|d(z^k, w^k)\|^2}. \tag{22}$$

If

$$\beta_k \big\| K^* K\big(z^k\big) - K^* K\big(z^{k+1}\big) \big\| \leq \rho \big\| z^k - z^{k+1} \big\|, \tag{23}$$

then set $\sigma_k = \sigma_0$; otherwise, set $\sigma_k = \beta_k$.

Remark 3.1 Let $z = (x, y)$. Then we have (see Section 4.4.1 in [28])

$$P_S(z) = (P_C x, P_Q y).$$

It is easy to see

$$K^* K z = \begin{pmatrix} A^* A & -A^* B \\ -B^* A & B^* B \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} A^*(Ax - By) \\ B^*(By - Ax) \end{pmatrix}.$$

Define the function $F : H_1 \times H_2 \to H_1$ by

$$F(x, y) = A^*(Ax - By), \tag{24}$$

and the function $G : H_1 \times H_2 \to H_2$ by

$$G(x, y) = B^*(By - Ax). \tag{25}$$

By setting $z^k = (x^k, y^k)$ and $w^k = (u^k, v^k)$, Algorithm 3.1 can be rewritten as follows:
For $k = 0, 1, 2, \ldots$, compute

$$\begin{cases} u^k = P_C(x^k - \beta_k F(x^k, y^k)), \\ v^k = P_Q(y^k - \beta_k G(x^k, y^k)), \end{cases} \tag{26}$$

where β_k is chosen to be the largest $\beta \in \{\sigma_k, \sigma_k \alpha, \sigma_k \alpha^2, \ldots\}$ satisfying

$$\beta^2 \big(\big\| F(x^k, y^k) - F(u^k, v^k) \big\|^2 + \big\| G(x^k, y^k) - G(u^k, v^k) \big\|^2 \big)$$
$$\leq \theta^2 \big(\big\| x^k - u^k \big\|^2 + \big\| y^k - v^k \big\|^2 \big). \tag{27}$$

Compute next iterates x^{k+1} and y^{k+1} by

$$\begin{cases} x_{\mathrm{I}}^{k+1} = x^k - \gamma \rho_k c_k, \\ y_{\mathrm{I}}^{k+1} = y^k - \gamma \rho_k d_k, \end{cases} \tag{28}$$

or

$$\begin{cases} x_{\mathrm{II}}^{k+1} = P_C(x^k - \gamma \beta_k \rho_k F(u^k, v^k)), \\ y_{\mathrm{II}}^{k+1} = P_Q(y^k - \gamma \beta_k \rho_k G(u^k, v^k)), \end{cases} \tag{29}$$

where $\gamma \in [0, 2)$,

$$\begin{cases} c_k := (x^k - u^k) - \beta_k (F(x^k, y^k) - F(u^k, v^k)); \\ d_k := (y^k - v^k) - \beta_k (G(x^k, y^k) - G(u^k, v^k)), \end{cases} \tag{30}$$

and

$$\rho_k := \frac{\langle x^k - u^k, c_k \rangle + \langle y^k - v^k, d_k \rangle + \beta_k \| Au^k - Bv^k \|^2}{\| c_k \|^2 + \| d_k \|^2}. \tag{31}$$

If

$$\beta_k^2 \big(\big\| F(x^k, y^k) - F(x^{k+1}, y^{k+1}) \big\|^2 + \big\| G(x^k, y^k) - G(x^{k+1}, y^{k+1}) \big\|^2 \big)$$
$$\leq \rho^2 \big(\big\| x^k - x^{k+1} \big\|^2 + \big\| y^k - y^{k+1} \big\|^2 \big), \tag{32}$$

then set $\sigma_k = \sigma_0$; otherwise, set $\sigma_k = \beta_k$.

For convenience, we call the projection algorithms which use update forms (19) (or (28)) and (20) (or (29)) Algorithm 3.1(I) and Algorithm 3.1(II), respectively.

Remark 3.2 For Algorithm 3.1, we can get the following conclusions:

(i) The only difference between Algorithm 3.1(II) and Algorithm 1.1 is that they use different stepsizes in the definitions of x^{k+1} and y^{k+1}.

(ii) There are two differences between Algorithm 3.1(I) and Algorithm 1.2. Firstly, the stepsize ρ_k in (10) of Algorithm 3.1(I) is larger than that in (36) of Algorithm 1.2. Secondly, there are no projections on the second step (19).

Remark 3.3 By the definitions of d_k in (21), the projection equation (17) can be written as

$$w^k = P_S\big(w^k - \big(\beta_k K^* K\big(w^k\big) - d\big(z^k, w^k\big)\big)\big).$$

So, from Lemma 2.1 we have

$$\big\langle z - w^k, \beta_k K^* K\big(w^k\big) - d\big(z^k, w^k\big)\big\rangle \geq 0, \quad \forall z \in S. \tag{33}$$

Lemma 3.1 *The search rule* (18) *is well defined. Besides* $\underline{\beta} \leq \beta_k \leq \sigma$, *where*

$$\underline{\beta} = \min\bigg\{\sigma, \frac{\alpha\theta}{\|K\|^2}\bigg\}. \tag{34}$$

Proof Obviously, $\beta_k \leq \sigma_k \leq \sigma_0$. In the latter case, we know that β_k/α must violate inequality (18), that is,

$$\big\|K^* K\big(z^k\big) - K^* K\big(w^k\big)\big\| \geq \theta \frac{\|z^k - w^k\|}{\beta_k/\alpha}. \tag{35}$$

On the other hand, we have

$$\big\|K^* K\big(z^k\big) - K^* K\big(w^k\big)\big\| \leq \|K\|^2 \big\|z^k - w^k\big\|. \tag{36}$$

Consequently, we get (34). □

Lemma 3.2 *Let* (z^k) *and* (w^k) *be generated by Algorithm* 3.1, *and let* d_k *and* ρ_k *be given by* (21) *and* (36), *respectively. Then we have*

$$\rho_k \geq \frac{1 - \theta}{1 + \theta^2}. \tag{37}$$

Proof By the Cauchy-Schwarz inequality, we have

$$
\begin{aligned}
\big\langle z^k - w^k, d\big(z^k, w^k\big)\big\rangle \\
&= \big\|z^k - w^k\big\|^2 - \beta_k\big\langle z^k - w^k, K^* K\big(z^k\big) - K^* K\big(w^k\big)\big\rangle \\
&\geq \big\|z^k - w^k\big\|^2 - \beta_k\big\|z^k - w^k\big\|\big\|K^* K\big(z^k\big) - K^* K\big(w^k\big)\big\| \\
&\geq \big\|z^k - w^k\big\|^2 - \theta\big\|z^k - w^k\big\|^2 \\
&= (1 - \theta)\big\|z^k - w^k\big\|^2.
\end{aligned}
\tag{38}
$$

By using $\langle z^k - w^k, K^*K(z^k) - K^*K(w^k) \rangle = \langle K(z^k) - K(w^k), K(z^k) - K(w^k) \rangle = \|K(z^k) - K(w^k)\|^2$, we have

$$
\begin{aligned}
\|d(z^k, w^k)\|^2 &= \|z^k - w^k\|^2 + \beta_k^2 \|K^*K(z^k) - K^*K(w^k)\|^2 \\
&\quad - 2\beta_k \langle z^k - w^k, K^*K(z^k) - K^*K(w^k) \rangle \\
&\leq \|z^k - w^k\|^2 + \theta^2 \|z^k - w^k\|^2 - 2\beta_k \|K(z^k) - K(w^k)\|^2 \\
&\leq (1 + \theta^2) \|z^k - w^k\|^2.
\end{aligned}
$$

So, we get (37). □

Lemma 3.3 *Let (z^k) and (w^k) be generated by Algorithm 3.1, and let d_k be given by (21). Then, for all $(z^*) \in \Gamma$, we have*

$$
\langle z^k - z^*, d(z^k, w^k) \rangle \geq \rho_k \|d(z^k, w^k)\|^2. \tag{39}
$$

Proof Take arbitrarily $z^* \in \Gamma$, that is, $z^* \in S$, and $K(z^*) = 0$. By setting $z = z^*$ in (33), we get

$$
\langle z^* - w^k, \beta_k K^*K(w^k) - d(z^k, w^k) \rangle \geq 0,
$$

which implies that

$$
\langle w^k - z^*, d(z^k, w^k) \rangle \geq \beta_k \langle w^k - z^*, K^*K(w^k) \rangle.
$$

It is easy to show that

$$
\langle w^k - z^*, K^*K(w^k) \rangle = \langle K(w^k - z^*), K(w^k) \rangle = \|K(w^k)\|^2. \tag{40}
$$

So we have

$$
\begin{aligned}
\langle z^k - z^*, d(z^k, w^k) \rangle &= \langle z^k - w^k, d(z^k, w^k) \rangle + \langle w^k - z^*, d(z^k, w^k) \rangle \\
&\geq \langle z^k - w^k, d(z^k, w^k) \rangle + \beta_k \|K(w^k)\|^2 \\
&= \rho_k \|d(z^k, w^k)\|^2,
\end{aligned}
$$

which implies (39). □

Theorem 3.1 *Let (z^k) be generated by Algorithm 3.1(I). If Γ is nonempty, then we have*

$$
\|z_{\mathrm{I}}^{k+1} - z^*\|^2 \leq \|z^k - z^*\|^2 - \gamma(2 - \gamma)\rho_k^2 \|d(z^k, w^k)\|^2, \quad \forall z^* \in \Gamma \tag{41}
$$

and (z^k) converges weakly to a solution of SEP (1).

Proof Let $z^* \in \Gamma$, that is, $z^* \in S$, and $K(z^*) = 0$. Then, from (39), we have

$$
\begin{aligned}
\|z_{\mathrm{I}}^{k+1} - z^*\|^2 &= \|z^k - z^*\|^2 + \gamma^2 \rho_k^2 \|d(z^k, w^k)\|^2 - 2\gamma\rho_k \langle z^k - z^*, d(z^k, w^k) \rangle \\
&\leq \|z^k - z^*\|^2 + \gamma^2 \rho_k^2 \|d(z^k, w^k)\|^2 - 2\gamma\rho_k^2 \|d(z^k, w^k)\|^2 \\
&= \|z^k - z^*\|^2 - \gamma(2 - \gamma)\rho_k^2 \|d(z^k, w^k)\|^2, \tag{42}
\end{aligned}
$$

which yields (41). Since $\gamma \in (0, 2)$, (42) implies that the sequence $(\|z^k - z^*\|^2)$ is decreasing and thus converges. Moreover, (z^k) is bounded. This implies that

$$\lim_{k \to \infty} \rho_k^2 \|d(z^k, w^k)\|^2 = 0. \tag{43}$$

From the definition of ρ_k, Lemmas 3.1 and 3.2, we have

$$\begin{aligned}
\rho_k^2 \|d(z^k, w^k)\|^2 &= \rho_k (\langle z^k - w^k, d(z^k, w^k)\rangle + \beta_k \|K(w^k)\|^2) \\
&\geq \rho_k [(1 - \theta) \|z^k - w^k\|^2 + \beta_k \|K(w^k)\|^2] \\
&\geq \frac{(1 - \theta)^2}{1 + \theta^2} \|z^k - w^k\|^2 + \frac{1 - \theta}{1 + \theta^2} \underline{\beta} \|K(w^k)\|^2,
\end{aligned}$$

which implies

$$\|z^k - w^k\|^2 \leq \frac{1 + \theta^2}{(1 - \theta)^2} \rho_k^2 \|d(z^k, w^k)\|^2$$

and

$$\|K(w^k)\|^2 \leq \frac{1 + \theta^2}{(1 - \theta)\underline{\beta}} \rho_k^2 \|d(z^k, w^k)\|^2.$$

Using (43), we get

$$\lim_{k \to \infty} \|z^k - w^k\| = 0 \tag{44}$$

and

$$\lim_{k \to \infty} \|K(w^k)\| = 0.$$

By the boundedness of K, we get

$$\lim_{k \to \infty} \|K(z^k)\| = 0. \tag{45}$$

Let $\hat{z} \in \omega_w(z^k)$, then there exists a subsequence (z^{k_i}) of (z^k) which converges weakly to $\hat{z}$. From (44), it follows that the subsequence (w^{k_i}) also converges weakly to $\hat{z}$. We will show that $\hat{z}$ is a solution of SEP (1). The weak convergence of $(K(z^{k_i}))$ to $K(\hat{z})$ and lower semicontinuity of the squared norm imply that

$$\|K(\hat{z})\| \leq \liminf_{i \to \infty} \|K(z^{k_i})\| = 0,$$

that is, $K(\hat{z}) = 0$. By noting that the equality in (17) can be rewritten as

$$\frac{z^{k_i} - w^{k_i}}{\beta_{k_i}} - K^* K(z^{k_i}) \in N_S(w^{k_i}),$$

and that the graph of the maximal monotone operator N_S is weakly-strongly closed, and by passing to the limit in the last inclusions, we obtain, from (44) and (45), that

$$\hat{z} \in S.$$

Hence $\hat{z} \in \Gamma$. Now we can apply Lemma 2.3 to $D := \Gamma$ to get that the full sequence (z^k) converges weakly to a point in Γ. This completes the proof. □

Remark 3.4 By using Remark 3.1, the contraction inequality (41) can be rewritten as follows:

$$\left\| x_I^{k+1} - x^* \right\|^2 + \left\| y_I^{k+1} - y^* \right\|^2 \le \left\| x^k - x^* \right\|^2 + \left\| y^k - y^* \right\|^2 - \gamma(2-\gamma)\rho_k^2 \left(\|c_k\|^2 + \|d_k\|^2 \right). \quad (46)$$

It is obvious that the ρ_k in (31) is larger than that in (10). Let $\gamma = 1$. Comparing (46) and (11), we claim that Algorithm 3.1(I) has a better contraction property than Algorithm 1.2.

Theorem 3.2 *Let (z^k) be generated by Algorithm 3.1(II). Assume that Γ is nonempty. Then we have*

$$\left\| z_{II}^{k+1} - z^* \right\|^2 \le \left\| z^k - z^* \right\|^2 - \gamma(2-\gamma)\rho_k^2 \left\| d(z^k, w^k) \right\|^2 - \left\| z_I^{k+1} - z_{II}^{k+1} \right\|^2, \quad \forall z^* \in \Gamma. \quad (47)$$

Furthermore, (z^k) converges weakly to a solution of SEP (1).

Proof Let $z^* \in \Gamma$, that is, $z^* \in S$, and $K(z^*) = 0$. Using Lemma 2.2(ii), we have

$$\begin{aligned}
\left\| z_{II}^{k+1} - z^* \right\|^2 &\le \left\| z^k - \gamma\beta_k\rho_k K^* K(w^k) - z^* \right\|^2 - \left\| z^k - \gamma\beta_k\rho_k K^* K(w^k) - z_{II}^{k+1} \right\|^2 \\
&= \left\| z^k - z^* \right\|^2 - \left\| z^k - z_{II}^{k+1} \right\|^2 - 2\gamma\beta_k\rho_k \langle z_{II}^{k+1} - z^*, K^* K(w^k) \rangle.
\end{aligned} \quad (48)$$

By setting $z = z_{II}^{k+1}$ in (33), we get

$$\begin{aligned}
-2\gamma\beta_k\rho_k &\langle z_{II}^{k+1} - w^k, K^* K(w^k) \rangle \\
&\le -2\gamma\rho_k \langle z_{II}^{k+1} - w^k, d(z^k, w^k) \rangle \\
&= -2\gamma\rho_k \langle z^k - w^k, d(z^k, w^k) \rangle - 2\gamma\rho_k \langle z_{II}^{k+1} - z^k, d(z^k, w^k) \rangle.
\end{aligned} \quad (49)$$

It holds

$$\begin{aligned}
-2\gamma\rho_k \langle z_{II}^{k+1} - z^k, d(z^k, w^k) \rangle &= - \left\| z^k - z_{II}^{k+1} - \gamma\rho_k d(z^k, w^k) \right\|^2 \\
&\quad + \left\| z^k - z_{II}^{k+1} \right\|^2 + \gamma^2\rho_k^2 \left\| d(z^k, w^k) \right\|^2.
\end{aligned} \quad (50)$$

Substituting (50) in the right-hand side of (49) and using $z^k - \gamma\rho_k d(z^k, w^k) = z_I^{k+1}$, we obtain

$$\begin{aligned}
-2\gamma\beta_k\rho_k &\langle z_{II}^{k+1} - w^k, K^* K(w^k) \rangle \\
&\le -2\gamma\rho_k \langle z^k - w^k, d(z^k, w^k) \rangle - \left\| z_I^{k+1} - z_{II}^{k+1} \right\|^2 \\
&\quad + \left\| z^k - z_{II}^{k+1} \right\|^2 + \gamma^2\rho_k^2 \left\| d(z^k, w^k) \right\|^2.
\end{aligned} \quad (51)$$

From (40), we get

$$-2\gamma\beta_k\rho_k\langle w^k - w^*, K^*K(w^k)\rangle = -2\gamma\beta_k\rho_k\|K(w^k)\|^2. \tag{52}$$

So, adding (51) and (52) and using the definition of ρ_k, we obtain

$$\begin{aligned}
-2\gamma\beta_k&\rho_k\langle z_{\mathrm{II}}^{k+1} - z^*, K^*K(w^k)\rangle \\
&\leq -2\gamma\rho_k\big(\langle z^k - w^k, d(z^k, w^k)\rangle + \beta_k\|K(w^k)\|^2\big) \\
&\quad - \|z_{\mathrm{I}}^{k+1} - z_{\mathrm{II}}^{k+1}\|^2 + \|z^k - z_{\mathrm{II}}^{k+1}\|^2 + \gamma^2\rho_k^2\|d(z^k, w^k)\|^2 \\
&\leq -2\gamma\rho_k^2\|d(z^k, w^k)\|^2 + \gamma^2\rho_k^2\|d(z^k, w^k)\|^2 - \|z_{\mathrm{I}}^{k+1} - z_{\mathrm{II}}^{k+1}\|^2 + \|z^k - z_{\mathrm{II}}^{k+1}\|^2 \\
&\leq -\gamma(2-\gamma)\rho_k^2\|d(z^k, w^k)\|^2 - \|z_{\mathrm{I}}^{k+1} - z_{\mathrm{II}}^{k+1}\|^2 + \|z^k - z_{\mathrm{II}}^{k+1}\|^2. \tag{53}
\end{aligned}$$

Adding (48) and (53), we obtain (47). Employing arguments which are similar to those used in the proof of Theorem 3.1, we obtain that the whole sequence (z^k) weakly converges to a solution of SEP (1), which completes proof. □

Remark 3.5 Comparing inequalities (41) and (47), we conclude that Algorithm 3.1(II) seems to have a better contraction property than Algorithm 3.1(I) since z_{II}^{k+1} is closer to z^* than z_{I}^{k+1} when z^k is the same.

3.2 Semi-alternating projection algorithms

Inspired by Algorithm 2.2 in [15] and based on Algorithm 3.1, we present two semi-alternating projection algorithms, whose name comes from an alternating technique taken in the first step.

Algorithm 3.2 Given constants $\sigma_0 > 0$, $\alpha \in (0,1)$, $\theta \in (0,1)$ and $\rho \in (0,1)$, let $x^0 \in H_1$ and $y^0 \in H_2$ be taken arbitrarily.

For $k = 0, 1, 2, \ldots$, compute

$$\begin{cases} u^k = P_C(x^k - \beta_k F(x^k, y^k)), \\ v^k = P_Q(y^k - \beta_k G(u^k, y^k)), \end{cases} \tag{54}$$

where β_k is chosen to be the largest $\beta \in \{\sigma_k, \sigma_k\alpha, \sigma_k\alpha^2, \ldots\}$ satisfying

$$\begin{aligned}
\beta^2\big(&\|F(x^k, y^k) - F(u^k, v^k)\|^2 + \|G(u^k, y^k) - G(u^k, v^k)\|^2\big) \\
&\leq \theta^2\big(\|x^k - u^k\|^2 + \|y^k - v^k\|^2\big). \tag{55}
\end{aligned}$$

Compute next iterates x^{k+1} and y^{k+1} by

$$\begin{cases} x_{\mathrm{I}}^{k+1} = x^k - \gamma\rho_k c_k, \\ y_{\mathrm{I}}^{k+1} = y^k - \gamma\rho_k d_k, \end{cases} \tag{56}$$

or

$$
\begin{cases}
x_{\mathrm{II}}^{k+1} = P_C(x^k - \gamma\beta_k\rho_k F(u^k, v^k)), \\
y_{\mathrm{II}}^{k+1} = P_Q(y^k - \gamma\beta_k\rho_k G(u^k, v^k)),
\end{cases}
\tag{57}
$$

where $\gamma \in [0, 2)$,

$$
\begin{cases}
c_k := (x^k - u^k) - \beta_k(F(x^k, y^k) - F(u^k, v^k)); \\
d_k := (y^k - v^k) - \beta_k(G(u^k, y^k) - G(u^k, v^k)),
\end{cases}
\tag{58}
$$

and

$$
\rho_k := \frac{\langle x^k - u^k, c_k \rangle + \langle y^k - v^k, d_k \rangle + \beta_k \|Au^k - Bv^k\|^2}{\|c_k\|^2 + \|d_k\|^2}.
\tag{59}
$$

If

$$
\beta_k^2 \big(\|F(x^k, y^k) - F(x^{k+1}, y^{k+1})\|^2 + \|G(x^k, y^k) - G(x^{k+1}, y^{k+1})\|^2 \big)
$$
$$
\leq \rho^2 \big(\|x^k - x^{k+1}\|^2 + \|y^k - y^{k+1}\|^2 \big),
\tag{60}
$$

then set $\sigma_k = \sigma_0$; otherwise, set $\sigma_k = \beta_k$.

For convenience, we call the projection algorithms which use update forms (56) and (57) Algorithm 3.2(I) and Algorithm 3.2(II), respectively.

Remark 3.6 By the definitions of c_k and d_k in (58), the projection equation (54) can be written as

$$
\begin{cases}
u^k = P_C(u^k - (\beta_k F(u^k, v^k) - c_k)), \\
v^k = P_Q(v^k - (\beta_k G(u^k, v^k) - d_k)).
\end{cases}
$$

So, from Lemma 2.1 we have

$$
\begin{cases}
\langle x - u^k, \beta_k F(u^k, v^k) - c_k \rangle \geq 0, & \forall x \in C, \\
\langle y - v^k, \beta_k G(u^k, v^k) - d_k \rangle \geq 0, & \forall y \in Q.
\end{cases}
\tag{61}
$$

Lemma 3.4 *The search rule* (55) *is well defined. Besides* $\underline{\beta}^* \leq \beta_k \leq \sigma_0$, *where*

$$
\underline{\beta}^* = \min\left\{ \sigma_0, \frac{\alpha\theta}{\sqrt{2}\|A\|^2}, \frac{\alpha\theta}{\|B\|\sqrt{2(\|A\|^2 + \|B\|^2)}} \right\}.
\tag{62}
$$

Proof Obviously, $\beta_k \leq \sigma_k \leq \sigma_0$. In the latter case, we know that β_k/α must violate inequality (55), that is,

$$
\beta^2/\alpha^2 \big(\|F(x^k, y^k) - F(u^k, v^k)\|^2 + \|G(u^k, y^k) - G(u^k, v^k)\|^2 \big)
$$
$$
\geq \theta^2 \big(\|x^k - u^k\|^2 + \|y^k - v^k\|^2 \big).
$$

On the other hand, we have

$$
\begin{aligned}
\big\| & F\big(x^k, y^k\big) - F\big(u^k, v^k\big)\big\|^2 + \big\|G\big(u^k, y^k\big) - G\big(u^k, v^k\big)\big\|^2 \\
&= \big\|A^*\big(Ax^k - By^k\big) - A^*\big(Au^k - Bv^k\big)\big\|^2 + \big\|B^*\big(By^k - Au^k\big) - B^*\big(Bv^k - Au^k\big)\big\|^2 \\
&\leq \|A\|^2\big(\big\|Ax^k - Au^k\big\| + \big\|By^k - Bv^k\big\|\big)^2 + \|B\|^4\big\|y^k - v^k\big\|^2 \\
&\leq 2\|A\|^2\big(\|A\|^2\big\|x^k - u^k\big\|^2 + \|B\|^2\big\|y^k - v^k\big\|^2\big) + \|B\|^4\big\|y^k - v^k\big\|^2 \\
&\leq 2\|A\|^4\big\|x^k - u^k\big\|^2 + \|B\|^2\big(2\|A\|^2 + \|B\|^2\big)\big\|y^k - v^k\big\|^2 \\
&\leq \max\big\{2\|A\|^4 + \|B\|^2\big(2\|A\|^2 + \|B\|^2\big)\big\}\big(\big\|x^k - u^k\big\|^2 + \big\|y^k - v^k\big\|^2\big).
\end{aligned}
$$

So, we get (62). □

Lemma 3.5 *Let (x^k, y^k) and (u^k, v^k) be generated by Algorithm* 3.2 *, and let c_k, d_k and ρ_k be given by* (58) *and* (59), *respectively. Then we have*

$$
\rho_k \geq \frac{1 - \theta}{1 + \theta^2}.
$$

Proof By the Cauchy-Schwarz inequality, we have

$$
\begin{aligned}
\big\langle & x^k - u^k, c_k\big\rangle + \big\langle y^k - v^k, d_k\big\rangle \\
&= \big\|x^k - u^k\big\|^2 + \big\|y^k - v^k\big\|^2 - \beta_k\big\langle x^k - u^k, F\big(x^k, y^k\big) - F\big(u^k, v^k\big)\big\rangle \\
&\quad - \beta_k\big\langle y^k - v^k, G\big(u^k, y^k\big) - G\big(u^k, v^k\big)\big\rangle \\
&\geq \big\|x^k - u^k\big\|^2 + \big\|y^k - v^k\big\|^2 \\
&\quad - \beta_k\big(\big\|x^k - u^k\big\|\big\|F\big(x^k, y^k\big) - F\big(u^k, v^k\big)\big\| + \big\|y^k - v^k\big\|\big\|G\big(u^k, y^k\big) - G\big(u^k, v^k\big)\big\|\big). \quad (63)
\end{aligned}
$$

It holds

$$
\begin{aligned}
\beta_k^2\big(&\big\|x^k - u^k\big\|\big\|F\big(x^k, y^k\big) - F\big(u^k, v^k\big)\big\| + \big\|y^k - v^k\big\|\big\|G\big(u^k, y^k\big) - G\big(u^k, v^k\big)\big\|\big)^2 \\
&= \beta_k^2\big(\big\|x^k - u^k\big\|^2\big\|F\big(x^k, y^k\big) - F\big(u^k, v^k\big)\big\|^2 + \big\|y^k - v^k\big\|^2\big\|G\big(u^k, y^k\big) - G\big(u^k, v^k\big)\big\|^2 \\
&\quad + 2\big\|x^k - u^k\big\|\big\|F\big(x^k, y^k\big) - F\big(u^k, v^k\big)\big\|\big\|y^k - v^k\big\|\big\|G\big(u^k, y^k\big) - G\big(u^k, v^k\big)\big\|\big) \\
&\leq \beta_k^2\big(\big\|x^k - u^k\big\|^2\big\|F\big(x^k, y^k\big) - F\big(u^k, v^k\big)\big\|^2 + \big\|y^k - v^k\big\|^2\big\|G\big(u^k, y^k\big) - G\big(u^k, v^k\big)\big\|^2 \\
&\quad + \big\|x^k - u^k\big\|^2\big\|G\big(u^k, y^k\big) - G\big(u^k, v^k\big)\big\|^2 + \big\|y^k - v^k\big\|^2\big\|F\big(x^k, y^k\big) - F\big(u^k, v^k\big)\big\|^2\big) \\
&= \beta_k^2\big(\big\|F\big(x^k, y^k\big) - F\big(u^k, v^k\big)\big\|^2 + \big\|G\big(u^k, y^k\big) - G\big(u^k, v^k\big)\big\|^2\big)\big(\big\|x^k - u^k\big\|^2 + \big\|y^k - v^k\big\|^2\big) \\
&\leq \theta^2\big(\big\|x^k - u^k\big\|^2 + \big\|y^k - v^k\big\|^2\big)^2. \quad (64)
\end{aligned}
$$

So, we obtain

$$
\begin{aligned}
\big\langle & x^k - u^k, c_k\big\rangle + \big\langle y^k - v^k, d_k\big\rangle \\
&\geq \big\|x^k - u^k\big\|^2 + \big\|y^k - v^k\big\|^2 - \theta\big(\big\|x^k - u^k\big\|^2 + \big\|y^k - v^k\big\|^2\big) \\
&= (1 - \theta)\big(\big\|x^k - u^k\big\|^2 + \big\|y^k - v^k\big\|^2\big). \quad (65)
\end{aligned}
$$

From the definition of F and G, we have

$$
\begin{aligned}
\|c_k\|^2 &+ \|d_k\|^2 \\
&= \|x^k - u^k\|^2 + \|y^k - v^k\|^2 \\
&\quad + \beta_k^2 \big(\|F(x^k, y^k) - F(u^k, v^k)\|^2 + \|G(u^k, y^k) - G(u^k, v^k)\|^2 \big) \\
&\quad - 2\beta_k \big(\langle x^k - u^k, F(x^k, y^k) - F(u^k, v^k) \rangle + \langle y^k - v^k, G(u^k, y^k) - G(u^k, v^k) \rangle \big) \\
&\le \|x^k - u^k\|^2 + \|y^k - v^k\|^2 + \theta^2 \big(\|x^k - u^k\|^2 + \|y^k - v^k\|^2 \big) \\
&\quad - 2\beta_k \big(\langle A(x^k - u^k), A(x^k - u^k) - B(y^k - v^k) \rangle \\
&\quad - \langle B(y^k - v^k), A(u^k - u^k) - B(y^k - v^k) \rangle \big) \\
&\le \|x^k - u^k\|^2 + \|y^k - v^k\|^2 + \theta^2 \big(\|x^k - u^k\|^2 + \|y^k - v^k\|^2 \big) \\
&\quad - 2\beta_k \big(\|A(x^k - u^k)\|^2 - \langle A(x^k - u^k), B(y^k - v^k) \rangle + \|B(y^k - v^k)\|^2 \big).
\end{aligned}
\tag{66}
$$

Since

$$
\begin{aligned}
-2\beta_k \big(&\|A(x^k - u^k)\|^2 - \langle A(x^k - u^k), B(y^k - v^k) \rangle + \|B(y^k - v^k)\|^2 \big) \\
&\le -2\beta_k \big(\|A(x^k - u^k)\|^2 - \|A(x^k - u^k)\| \|B(y^k - v^k)\| + \|B(y^k - v^k)\|^2 \big) \\
&\le -2\beta_k \Big(\|A(x^k - u^k)\|^2 - \frac{1}{2} \big(\|A(x^k - u^k)\|^2 + \|B(y^k - v^k)\|^2 \big) + \|B(y^k - v^k)\|^2 \Big) \\
&\le -\beta_k \big(\|A(x^k - u^k)\|^2 + \|B(y^k - v^k)\|^2 \big),
\end{aligned}
$$

by (66), we get

$$
\begin{aligned}
\|c_k\|^2 &+ \|d_k\|^2 \\
&\le (1 + \theta^2) \big(\|x^k - u^k\|^2 + \|y^k - v^k\|^2 \big) - \beta_k \big(\|A(x^k - u^k)\|^2 + \|B(y^k - v^k)\|^2 \big) \\
&\le (1 + \theta^2) \big(\|x^k - u^k\|^2 + \|y^k - v^k\|^2 \big).
\end{aligned}
\tag{67}
$$

Combining (65) and (67), we complete the proof. $\qquad \square$

Lemma 3.6 *Let (x^k, y^k) and (u^k, v^k) be generated by Algorithm 3.2, and let c_k and d_k be given by* (58). *Then, for all $(x^*, y^*) \in \Gamma$, we have*

$$
\langle x^k - x^*, c_k \rangle + \langle y^k - y^*, d_k \rangle \ge \rho_k \big(\|c_k\|^2 + \|d_k\|^2 \big).
$$

Proof Take arbitrarily $(x^*, y^*) \in \Gamma$, that is, $x^* \in C, y^* \in Q$, and $Ax^* = By^*$. By setting $(x, y) = (x^*, y^*)$ in (61), we get

$$
\begin{cases}
\langle x^* - u^k, \beta_k F(u^k, v^k) - c_k \rangle \ge 0, \\
\langle y^* - v^k, \beta_k G(u^k, v^k) - d_k \rangle \ge 0,
\end{cases}
$$

which implies that

$$\begin{cases} \langle u^k - x^*, c_k \rangle \geq \beta_k \langle u^k - x^*, F(u^k, v^k) \rangle, \\ \langle v^k - y^*, d_k \rangle \geq \beta_k \langle v^k - y^*, G(u^k, v^k) \rangle. \end{cases}$$

By the definition of F and G, we have

$$\begin{aligned} \langle u^k - x^*, F(u^k, v^k) \rangle &+ \langle v^k - y^*, G(u^k, v^k) \rangle \\ &= \langle u^k - x^*, A^*(Au^k - Bv^k) \rangle + \langle v^k - y^*, B^*(Bv^k - Au^k) \rangle \\ &= \langle Au^k - Ax^*, Au^k - Bv^k \rangle + \langle Bv^k - By^*, Bv^k - Au^k \rangle \\ &= \langle Au^k - Bv^k - (Ax^* - By^*), Au^k - Bv^k \rangle \\ &= \|Au^k - Bv^k\|^2, \end{aligned} \tag{68}$$

and

$$\begin{aligned} \langle x^k - x^*, c_k \rangle &+ \langle y^k - y^*, d_k \rangle \\ &= \langle x^k - u^k, c_k \rangle + \langle y^k - v^k, d_k \rangle + \langle u^k - x^*, c_k \rangle + \langle v^k - y^*, d_k \rangle \\ &\geq \langle x^k - u^k, c_k \rangle + \langle y^k - v^k, d_k \rangle + \beta_k \|Au^k - Bv^k\|^2 \\ &= \rho_k (\|c_k\|^2 + \|d_k\|^2). \end{aligned}$$

So, we complete the proof. □

Theorem 3.3 *Let (x^k, y^k) be generated by Algorithm 3.2(I). If Γ is nonempty, then we have*

$$\begin{aligned} \|x_I^{k+1} - x^*\|^2 &+ \|y_I^{k+1} - y^*\|^2 \\ &\leq \|x^k - x^*\|^2 + \|y^k - y^*\|^2 - (2 - \gamma)\gamma\rho_k^2(\|c_k\|^2 + \|d_k\|^2), \quad \forall(x^*, y^*) \in \Gamma, \end{aligned} \tag{69}$$

and (x^k, y^k) converges weakly to a solution of SEP (1).

Proof Let $(x^*, y^*) \in \Gamma$, that is, $x^* \in C$, $y^* \in Q$, and $Ax^* = By^*$. Then we have

$$\begin{aligned} \|x_I^{k+1} - x^*\|^2 &= \|x^k - \gamma\rho_k c_k - x^*\|^2 \\ &= \|x^k - x^*\|^2 + \gamma^2\rho_k^2\|c_k\|^2 - 2\gamma\rho_k\langle x^k - x^*, c_k \rangle. \end{aligned}$$

Similarly, we get

$$\|y_I^{k+1} - y^*\|^2 = \|y^k - y^*\|^2 + \gamma^2\rho_k^2\|d_k\|^2 - 2\gamma\rho_k\langle y^k - y^*, d_k \rangle.$$

Adding the above inequalities and using Lemma 3.6, we have

$$\begin{aligned} \|x_I^{k+1} - x^*\|^2 &+ \|y_I^{k+1} - y^*\|^2 \\ &= \|x^k - x^*\|^2 + \|y^k - y^*\|^2 + \gamma^2\rho_k^2(\|c_k\|^2 + \|d_k\|^2) \end{aligned}$$

$$- 2\gamma\rho_k\langle x^k - x^*, c_k\rangle - 2\gamma\rho_k\langle y^k - y^*, d_k\rangle$$
$$\leq \|x^k - x^*\|^2 + \|y^k - y^*\|^2 - (2 - \gamma)\gamma\rho_k^2(\|c_k\|^2 + \|d_k\|^2), \tag{70}$$

which yields (69). Since $\gamma \in (0, 2)$, (70) implies that the sequence $\|x^k - x^*\|^2 + \|y^k - y^*\|^2$ is decreasing and thus converges. Moreover, (x^k) and (y^k) are bounded. This implies that

$$\lim_{k\to\infty} \rho_k^2(\|c_k\|^2 + \|d_k\|^2) = 0. \tag{71}$$

From the definition of ρ_k, Lemmas 3.4 and 3.5, we have

$$\rho_k^2(\|c_k\|^2 + \|d_k\|^2)$$
$$= \rho_k(\langle x^k - u^k, c_k\rangle + \langle y^k - v^k, d_k\rangle + \beta_k\|Au^k - Bv^k\|^2)$$
$$\geq \rho_k[(1 - \theta)(\|x^k - u^k\|^2 + \|y^k - v^k\|^2) + \beta_k\|Au^k - Bv^k\|^2]$$
$$\geq \frac{(1 - \theta)^2}{1 + \theta^2}(\|x^k - u^k\|^2 + \|y^k - v^k\|^2) + \frac{1 - \theta}{1 + \theta^2}\underline{\beta}\|Au^k - Bv^k\|^2,$$

which implies

$$\|x^k - u^k\|^2 + \|y^k - v^k\|^2 \leq \frac{1 + \theta^2}{(1 - \theta)^2}\rho_k^2(\|c_k\|^2 + \|d_k\|^2)$$

and

$$\|Au^k - Bv^k\|^2 \leq \frac{1 + \theta^2}{(1 - \theta)\underline{\beta}}\rho_k^2(\|c_k\|^2 + \|d_k\|^2).$$

Using (71), we get

$$\lim_{k\to\infty} \|x^k - u^k\| + \|y^k - v^k\| = 0, \tag{72}$$

and

$$\lim_{k\to\infty} \|Au^k - Bv^k\| = 0. \tag{73}$$

Hence, we get

$$\lim_{k\to\infty} \|Ax^k - By^k\| = 0.$$

Let $(\hat{x}, \hat{y}) \in \omega_w(x^k, y^k)$, then there exist two subsequences (x^{k_i}) and (y^{k_i}) of (x^k) and (y^k) which converge weakly to $\hat{x}$ and $\hat{y}$, respectively. From (72), it follows that (u^{k_i}) and (v^{k_i}) also converge weakly to $\hat{x}$ and $\hat{y}$, respectively. We will show that $(\hat{x}, \hat{y})$ is a solution of SEP (1). The weak convergence of $(Ax^{k_i} - By^{k_i})$ to $A\hat{x} - B\hat{y}$ and the lower semicontinuity of the squared norm imply that

$$\|A\hat{x} - B\hat{y}\| \leq \liminf_{i\to\infty} \|Ax^{k_i} - By^{k_i}\| = 0,$$

that is, $A\hat{x} = B\hat{y}$. By noting that the two equalities in (54) can be rewritten as

$$\begin{cases} \frac{x^{k_i} - u^{k_i}}{\beta_{k_i}} - A^*(Au^{k_i} - Bv^{k_i}) \in N_C(u^{k_i}), \\ \frac{y^{k_i} - v^{k_i}}{\beta_{k_i}} - B^*(Bv^{k_i} - Au^{k_i}) \in N_Q(v^{k_i}), \end{cases}$$

and that the graphs of the maximal monotone operators N_C and N_Q are weakly-strongly closed, and by passing to the limit in the last inclusions, we obtain, from (72) and (73), that

$$\hat{x} \in C, \qquad \hat{y} \in Q.$$

Hence $(\hat{x}, \hat{y}) \in \Gamma$.

Now we can apply Lemma 2.3 to $D := \Gamma$ to get that the full sequence (x^k, y^k) converges weakly to a point in Γ. This completes the proof. □

Remark 3.7 Employing arguments which are similar to those used in Remark 3.4, comparing (69) and (2.48) in [15], we conclude that Algorithm 3.2(I) has a better contraction property than the hybrid alternating CQ-algorithm in [15].

Theorem 3.4 *Let (x^k, y^k) be generated by Algorithm* 3.2(II). *If Γ is nonempty, then we have*

$$\left\| x_{\mathrm{II}}^{k+1} - x^* \right\|^2 + \left\| y_{\mathrm{II}}^{k+1} - y^* \right\|^2 \leq \left\| x^k - x^* \right\|^2 + \left\| y^k - y^* \right\|^2 - \gamma(2 - \gamma)\rho_k^2 \left(\|c_k\|^2 + \|d_k\|^2 \right)$$
$$- \left\| x_{\mathrm{I}}^{k+1} - x_{\mathrm{II}}^{k+1} \right\|^2 - \left\| y_{\mathrm{I}}^{k+1} - y_{\mathrm{II}}^{k+1} \right\|^2, \tag{74}$$

and (x^k, y^k) converges weakly to a solution of SEP (1).

Proof Let $(x^*, y^*) \in \Gamma$, that is, $x^* \in C$, $y^* \in Q$, and $Ax^* = By^*$. Using Lemma 2.2(ii), we have

$$\left\| x_{\mathrm{II}}^{k+1} - x^* \right\|^2 \leq \left\| x^k - \gamma\beta_k\rho_k F(u^k, v^k) - x^* \right\|^2 - \left\| x^k - \gamma\beta_k\rho_k F(u^k, v^k) - x_{\mathrm{II}}^{k+1} \right\|^2$$
$$= \left\| x^k - x^* \right\|^2 - \left\| x^k - x_{\mathrm{II}}^{k+1} \right\|^2 - 2\gamma\beta_k\rho_k \langle x_{\mathrm{II}}^{k+1} - x^*, F(u^k, v^k) \rangle.$$

Similarly, we get

$$\left\| y_{\mathrm{II}}^{k+1} - y^* \right\|^2 \leq \left\| y^k - y^* \right\|^2 - \left\| y^k - y_{\mathrm{II}}^{k+1} \right\|^2 - 2\gamma\beta_k\rho_k \langle y_{\mathrm{II}}^{k+1} - y^*, G(u^k, v^k) \rangle.$$

Adding the above inequalities, we obtain

$$\left\| x_{\mathrm{II}}^{k+1} - x^* \right\|^2 + \left\| y_{\mathrm{II}}^{k+1} - y^* \right\|^2$$
$$\leq \left\| x^k - x^* \right\|^2 + \left\| y^k - y^* \right\|^2 - \left\| x^k - x_{\mathrm{II}}^{k+1} \right\|^2 - \left\| y^k - y_{\mathrm{II}}^{k+1} \right\|^2$$
$$- 2\gamma\beta_k\rho_k \langle x_{\mathrm{II}}^{k+1} - x^*, F(u^k, v^k) \rangle - 2\gamma\beta_k\rho_k \langle y_{\mathrm{II}}^{k+1} - y^*, G(u^k, v^k) \rangle. \tag{75}$$

By setting $(x, y) = (x_{\mathrm{II}}^{k+1}, y_{\mathrm{II}}^{k+1})$ in (61), we get

$$-2\gamma\beta_k\rho_k \langle x_{\mathrm{II}}^{k+1} - u^k, F(u^k, v^k) \rangle - 2\gamma\beta_k\rho_k \langle y_{\mathrm{II}}^{k+1} - v^k, G(u^k, v^k) \rangle$$
$$\leq -2\gamma\rho_k \langle x_{\mathrm{II}}^{k+1} - u^k, c_k \rangle - 2\gamma\rho_k \langle y_{\mathrm{II}}^{k+1} - v^k, d_k \rangle$$
$$= -2\gamma\rho_k \left(\langle x^k - u^k, c_k \rangle + \langle y^k - v^k, d_k \rangle \right) - 2\gamma\rho_k \left(\langle x_{\mathrm{II}}^{k+1} - x^k, c_k \rangle + \langle y_{\mathrm{II}}^{k+1} - y^k, d_k \rangle \right). \tag{76}$$

It holds

$$-2\gamma\rho_k\langle x_{\mathrm{II}}^{k+1} - x^k, c_k\rangle = -\left\|x^k - x_{\mathrm{II}}^{k+1} - \gamma\rho_k c_k\right\|^2 + \left\|x^k - x_{\mathrm{II}}^{k+1}\right\|^2 + \gamma^2\rho_k^2\|c_k\|^2. \tag{77}$$

Similarly, we get

$$-2\gamma\rho_k\langle y_{\mathrm{II}}^{k+1} - y^k, d_k\rangle = -\left\|y^k - y_{\mathrm{II}}^{k+1} - \gamma\rho_k d_k\right\|^2 + \left\|y^k - y_{\mathrm{II}}^{k+1}\right\|^2 + \gamma^2\rho_k^2\|d_k\|^2. \tag{78}$$

Substituting (77) and (78) in the right-hand side of (76) and using $x^k - \gamma\rho_k c_k = x_{\mathrm{I}}^{k+1}$ and $y^k - \gamma\rho_k d_k = y_{\mathrm{I}}^{k+1}$, we obtain

$$\begin{aligned}
&-2\gamma\beta_k\rho_k\langle x_{\mathrm{II}}^{k+1} - u^k, F(u^k, v^k)\rangle - 2\gamma\beta_k\rho_k\langle y_{\mathrm{II}}^{k+1} - v^k, G(u^k, v^k)\rangle \\
&\quad \leq -2\gamma\rho_k\left(\langle x^k - u^k, c_k\rangle + \langle y^k - v^k, d_k\rangle\right) \\
&\qquad - \left\|x_{\mathrm{I}}^{k+1} - x_{\mathrm{II}}^{k+1}\right\|^2 + \left\|x^k - x_{\mathrm{II}}^{k+1}\right\|^2 + \gamma^2\rho_k^2\|c_k\|^2 \\
&\qquad - \left\|y_{\mathrm{I}}^{k+1} - y_{\mathrm{II}}^{k+1}\right\|^2 + \left\|y^k - y_{\mathrm{II}}^{k+1}\right\|^2 + \gamma^2\rho_k^2\|d_k\|^2.
\end{aligned} \tag{79}$$

From (68), we have

$$\begin{aligned}
&-2\gamma\beta_k\rho_k\langle u^k - x^*, F(u^k, v^k)\rangle - 2\gamma\beta_k\rho_k\langle v^k - y^*, G(u^k, v^k)\rangle \\
&\quad = -2\gamma\beta_k\rho_k\left\|Au^k - Bv^k\right\|^2.
\end{aligned} \tag{80}$$

So, adding (79) and (80) and using the definition of ρ_k, we obtain

$$\begin{aligned}
&-2\gamma\beta_k\rho_k\langle x_{\mathrm{II}}^{k+1} - x^*, F(u^k, v^k)\rangle - 2\gamma\beta_k\rho_k\langle y_{\mathrm{II}}^{k+1} - y^*, G(u^k, v^k)\rangle \\
&\quad \leq -2\gamma\rho_k\left(\langle x^k - u^k, c_k\rangle + \langle y^k - v^k, d_k\rangle + \beta_k\left\|Au^k - Bv^k\right\|^2\right) \\
&\qquad - \left\|x_{\mathrm{I}}^{k+1} - x_{\mathrm{II}}^{k+1}\right\|^2 + \left\|x^k - x_{\mathrm{II}}^{k+1}\right\|^2 + \gamma^2\rho_k^2\|c_k\|^2 \\
&\qquad - \left\|y_{\mathrm{I}}^{k+1} - y_{\mathrm{II}}^{k+1}\right\|^2 + \left\|y^k - y_{\mathrm{II}}^{k+1}\right\|^2 + \gamma^2\rho_k^2\|d_k\|^2 \\
&\quad \leq -2\gamma\rho_k^2\left(\|c_k\|^2 + \|d_k\|^2\right) + \gamma^2\rho_k^2\left(\|c_k\|^2 + \|d_k\|^2\right) \\
&\qquad - \left\|x_{\mathrm{I}}^{k+1} - x_{\mathrm{II}}^{k+1}\right\|^2 - \left\|y_{\mathrm{I}}^{k+1} - y_{\mathrm{II}}^{k+1}\right\|^2 + \left\|x^k - x_{\mathrm{II}}^{k+1}\right\|^2 + \left\|y^k - y_{\mathrm{II}}^{k+1}\right\|^2 \\
&\quad \leq -\gamma(2 - \gamma)\rho_k^2\left(\|c_k\|^2 + \|d_k\|^2\right) \\
&\qquad - \left\|x_{\mathrm{I}}^{k+1} - x_{\mathrm{II}}^{k+1}\right\|^2 - \left\|y_{\mathrm{I}}^{k+1} - y_{\mathrm{II}}^{k+1}\right\|^2 + \left\|x^k - x_{\mathrm{II}}^{k+1}\right\|^2 + \left\|y^k - y_{\mathrm{II}}^{k+1}\right\|^2.
\end{aligned} \tag{81}$$

Adding (75) and (81), we obtain (74). Employing arguments which are similar to those used in the proof of Theorem 3.3, we obtain that the whole sequence (x^k, y^k) weakly converges to a solution of SEP (1), which completes proof. □

4 Applications

The split feasibility problem (SFP) formulated as follows:

$$\text{Find} \quad x \in C \quad \text{such that} \quad Ax \in Q, \tag{82}$$

was originally introduced in Censor and Elfving [29]. The SFP can be a model for many inverse problems where constraints are imposed on the solutions in the domain of a linear operator as well as in the operator's range. It has a variety of specific applications in real world, such as medical care, image reconstruction and signal processing (see [30–33] for details).

In fact, the SEP is equivalent to the SFP. Firstly, observe that the equality $Ax = By$ in (1) equals to

$$(B^*B)^{-1}B^*BAx = (B^*B)^{-1}(B^*B)y = y.$$

So, if we define the linear and bounded operator $L = (B^*B)^{-1}B^*BA : H_1 \to H_2$, then the SEP becomes a special case of the SFP with the operator L (e.g., see [34, 35]).

On the other hand, if $H_2 = H_3$ and $B = I$, then the split equality problem (1) reduces to the split feasibility problem.

Based on this equivalence, we can construct iterative algorithms for the SEP by using the algorithms for the SFP if the operator $(B^*B)^{-1}$ is easily computed. We also can extend the algorithms for the SEP to the SFP.

Next, we present an algorithm for the SFP based on Algorithm 3.2.

Define the function $F : H_1 \times H_2 \to H_1$ by

$$F(x, y) = A^*(Ax - y),$$

and the function $G : H_1 \times H_2 \to H_2$ by

$$G(x, y) = y - Ax.$$

Algorithm 4.1 Given constants $\sigma_0 > 0$, $\alpha \in (0, 1)$, $\theta \in (0, 1)$ and $\rho \in (0, 1)$, let $x^0 \in H_1$ and $y^0 \in H_2$ be taken arbitrarily.

For $k = 0, 1, 2, \ldots$, compute

$$\begin{cases} u^k = P_C(x^k - \beta_k F(x^k, y^k)), \\ v^k = P_Q(y^k - \beta_k G(u^k, y^k)), \end{cases}$$

where β_k is chosen to be the largest $\beta \in \{\sigma_k, \sigma_k\alpha, \sigma_k\alpha^2, \ldots\}$ satisfying

$$\beta^2 \left(\left\| F(x^k, y^k) - F(u^k, v^k) \right\|^2 + \left\| y^k - v^k \right\|^2 \right) \le \theta^2 \left(\left\| x^k - u^k \right\|^2 + \left\| y^k - v^k \right\|^2 \right).$$

Compute next iterates x^{k+1} and y^{k+1} by

$$\begin{cases} x_{\mathrm{I}}^{k+1} = x^k - \gamma\rho_k c_k, \\ y_{\mathrm{I}}^{k+1} = y^k - \gamma\rho_k d_k, \end{cases}$$

or

$$\begin{cases} x_{\mathrm{II}}^{k+1} = P_C(x^k - \gamma\beta_k\rho_k F(u^k, v^k)), \\ y_{\mathrm{II}}^{k+1} = P_Q(y^k - \gamma\beta_k\rho_k G(u^k, v^k)), \end{cases}$$

where $\gamma \in [0, 2)$,

$$
\begin{cases}
c_k := (x^k - u^k) - \beta_k(F(x^k, y^k) - F(u^k, v^k)); \\
d_k := (y^k - v^k) - \beta_k(y^k - v^k),
\end{cases}
$$

and

$$
\rho_k := \frac{\langle x^k - u^k, c_k \rangle + \langle y^k - v^k, d_k \rangle + \beta_k \|Au^k - v^k\|^2}{\|c_k\|^2 + \|d_k\|^2}.
$$

If

$$
\beta_k^2 \left(\left\| F(x^k, y^k) - F(x^{k+1}, y^{k+1}) \right\|^2 + \left\| G(x^k, y^k) - G(x^{k+1}, y^{k+1}) \right\|^2 \right) \\
\leq \rho^2 \left(\left\| x^k - x^{k+1} \right\|^2 + \left\| y^k - y^{k+1} \right\|^2 \right),
\tag{83}
$$

then set $\sigma_k = \sigma_0$; otherwise, set $\sigma_k = \beta_k$.

Using Theorem 3.3, we get the convergence of Algorithm 4.1.

Theorem 4.1 *Let (x^k, y^k) be generated by Algorithm* 4.1. *If the set of solutions of the SFP is nonempty, then (x^k, y^k) converges weakly to a solution of SFP* (82).

Remark 4.1 Similarly, it is easy to extend Algorithm 3.1 and Theorems 3.1 and 3.2 to the SFP. Here we omit it.

5 Numerical examples

In this section, we use the numerical example in [8] to demonstrate the efficiency and advantage of Algorithms 3.1 and 3.2 by comparing them with Algorithms 1.1, 1.2 and 1.3.

We denote the vector with all elements 0 by e_0, and the vector with all elements 1 by e_1 in what follows. In the numerical results listed in the following table, 'Iter.' and 'Sec.' denote the number of iterations and the cpu time in seconds, respectively. For Algorithms 1.1, 1.2, 3.1 and 3.2, 'InIt.' denotes the number of total iterations of finding suitable β_k.

Example 5.1 The SEP with $A = (a_{ij})_{J \times N}$, $B = (b_{ij})_{J \times M}$, $C = \{x \in R^N \mid \|x\| \leq 0.25\}$, $Q = \{y \in R^M \mid e_0 \leq y \leq u\}$, where $a_{ij} \in [0, 1]$, $b_{ij} \in [0, 1]$ and $u \in [e_1, 2e_1]$ are all generated uniformly randomly.

In the implementations, we take $\|Ax - By\| < \varepsilon = 10^{-4}$ as the stopping criterion. Take the initial value $x_0 = 10e_1$, $y_0 = -10e_1$.

We make comparison of Algorithms 1.1, 1.2, 1.3, 3.1, 3.2 and FISTA with different J, N, M, and report the results in Tables 1, 2, 3 and Figure 1. We take the stepsize β_k via a backtracking stepsize rule. For comparison, we tried to choose best parameters through numerical experiments. We take $\gamma = 0.8$, $\theta = 0.99$, $\sigma = 50$, $\rho = 0.1$ and $\alpha = 0.1$ in Algorithms 1.1, 1.2, 3.1 and 3.2. And we take $\sigma_k = 0.65$ in Algorithm 1.3. So the stepsize β_k is chosen in such a way that

$$
\beta_k = 0.65 \times \min \left\{ \frac{\|Ax^k - By^k\|^2}{\|A^*(Ax^k - By^k)\|^2}, \frac{\|Ax^k - By^k\|^2}{\|B^*(Ax^k - By^k)\|^2} \right\}.
$$

**Table 1 Computational results for Example 5.1 with (N, M) = (100, 50)**

J		50	100	150	200	250
Algorithm 1.1	Iter.	3263	90,378	297,135	65,795	31,655
	Inlt.	13,748	377,483	864,321	172,925	82,979
	Sec.	2.188	36.672	110.563	25.500	13.781
Algorithm 1.2	Iter.	8732	194,940	434,539	82,993	43,689
	Inlt.	46,182	1,069,758	1,327,537	225,913	141,813
	Sec.	6.797	106.234	189.078	38.094	23.422
Algorithm 1.3	Iter.	336	2012	4302	1327	676
	Sec.	0.063	0.406	1.125	0.406	0.219
FISTA	Iter.	1389	2580	3787	3260	2491
	Inlt.	1397	2589	3796	3270	2501
	Sec.	0.391	0.453	0.734	0.656	0.563
Algorithm 3.1(I)	Iter.	199	793	4342	482	718
	Inlt.	254	913	5206	1145	1148
	Sec.	0.094	0.156	0.953	0.156	0.188
Algorithm 3.1(II)	Iter.	152	764	2176	580	697
	Inlt.	184	860	2232	604	769
	Sec.	0.078	0.219	0.406	0.125	0.125
Algorithm 3.2(I)	Iter.	81	2208	5704	1022	362
	Inlt.	178	4680	7562	2261	846
	Sec.	0.063	0.469	1.891	0.250	0.094
Algorithm 3.2(II)	Iter.	65	952	2311	629	263
	Inlt.	80	976	2375	653	288
	Sec.	0.031	0.188	0.516	0.156	0.063

Table 2 Computational results for Example 5.1 with (N, M) = (150, 150)

J		50	100	150	200	250
Algorithm 1.1	Iter.	5639	14,609	36,702	895,632	364,304
	Inlt.	29,524	43,313	107,748	2,740,752	1,179,620
	Sec.	2.813	6.297	19.125	551.250	259.656
Algorithm 1.2	Iter.	9175	33,306	109,165	2,481,066	566,203
	Inlt.	28,810	139,247	389,839	10,985,694	3,467,821
	Sec.	3.906	19.922	69.797	2079.344	649.234
Algorithm 1.3	Iter.	2559	9995	40,713	1,535,172	353,573
	Sec.	0.531	3.063	16.563	793.875	219.938
FISTA	Iter.	2158	3078	3092	17,010	9264
	Inlt.	2167	3088	3102	17,021	9275
	Sec.	0.375	0.656	0.797	4.922	2.984
Algorithm 3.1(I)	Iter.	123	186	1069	26,742	8307
	Inlt.	131	690	1779	32,790	15,519
	Sec.	0.031	0.063	0.359	7.750	5.141
Algorithm 3.1(II)	Iter.	136	171	726	17,575	3765
	Inlt.	160	187	807	18,007	3813
	Sec.	0.063	0.125	0.188	5.047	1.922
Algorithm 3.2(I)	Iter.	83	808	477	27,199	10,584
	Inlt.	182	1713	1140	31,301	12,084
	Sec.	0.063	0.125	0.147	13.078	5.563
Algorithm 3.2(II)	Iter.	43	235	297	15,515	7331
	Inlt.	66	251	322	15,839	7520
	Sec.	0.006	0.063	0.094	7.094	3.750

Table 3 Computational results for Example 5.1 with $(N, M) = (200, 250)$

J		50	100	150	200	250
Algorithm 1.1	Iter.	3477	6828	15,749	48,555	255,440
	Inlt.	11,943	21,084	60,677	336,456	1,326,061
	Sec.	1.688	3.844	12.672	67.453	334.891
Algorithm 1.2	Iter.	10,464	21,319	32,055	122,743	483,468
	Inlt.	30,876	86,575	162,387	420,341	2,185,371
	Sec.	5.281	16.063	32.156	111.891	600.625
Algorithm 1.3	Iter.	648	6647	16,810	44,817	132,873
	Sec.	0.188	2.781	9.500	32.734	118.203
FISTA	Iter.	2346	2931	4040	3804	6977
	Inlt.	2355	2941	4051	3815	6989
	Sec.	0.500	0.750	1.250	1.344	3.141
Algorithm 3.1(I)	Iter.	109	236	343	814	2077
	Inlt.	151	278	415	975	2518
	Sec.	0.031	0.094	0.188	0.344	7.078
Algorithm 3.1(II)	Iter.	168	188	262	756	1106
	Inlt.	180	200	268	792	1142
	Sec.	0.063	0.109	0.125	0.218	0.438
Algorithm 3.2(I)	Iter.	117	168	818	1718	1725
	Inlt.	128	222	986	2000	2408
	Sec.	0.063	0.063	0.438	1.063	1.281
Algorithm 3.2(II)	Iter.	82	83	373	582	1240
	Inlt.	98	108	400	644	1285
	Sec.	0.031	0.078	0.199	0.406	0.563


Figure 1 Numbers of projections with $(N, M) = (100, 50)$.
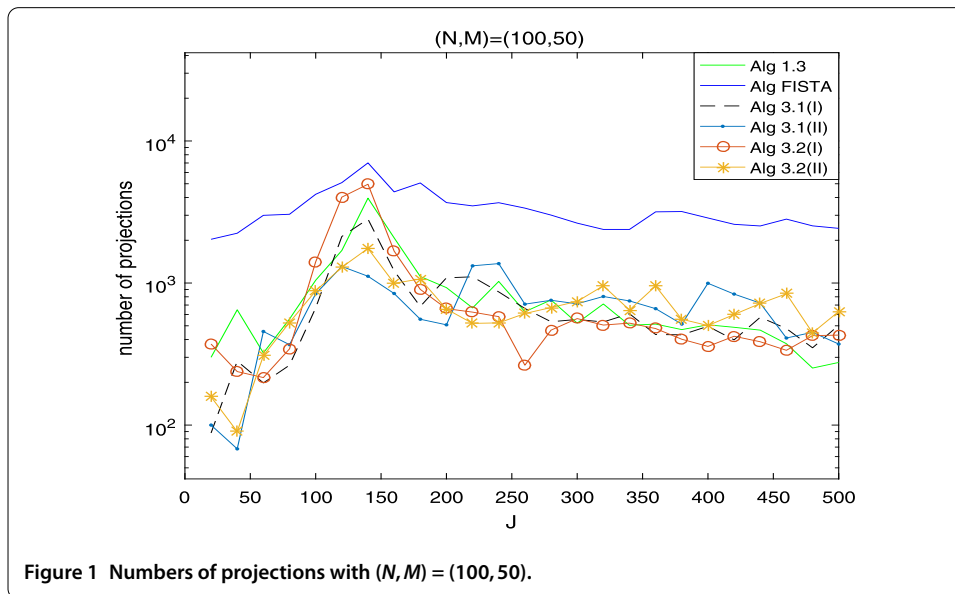
We take $L_0 = 13$, $\eta = 2$ and $a = 7$ for FISTA with backtracking (see [21]). For comparison, the same random values are taken in each test for different algorithms.

The numerical results are listed in Tables 1, 2, 3 and Figures 1-6, from which we can get some conclusions:

(1) Algorithm 1.2 behaves worst, and Algorithm 1.1 is superior to it, while inferior to Algorithms 1.3, 3.1 and 3.2.

Figure 2 Numbers of matrix-vector evaluations with $(N, M) = (100, 50)$.

Figure 3 Numbers of projections with $(N, M) = (150, 150)$.
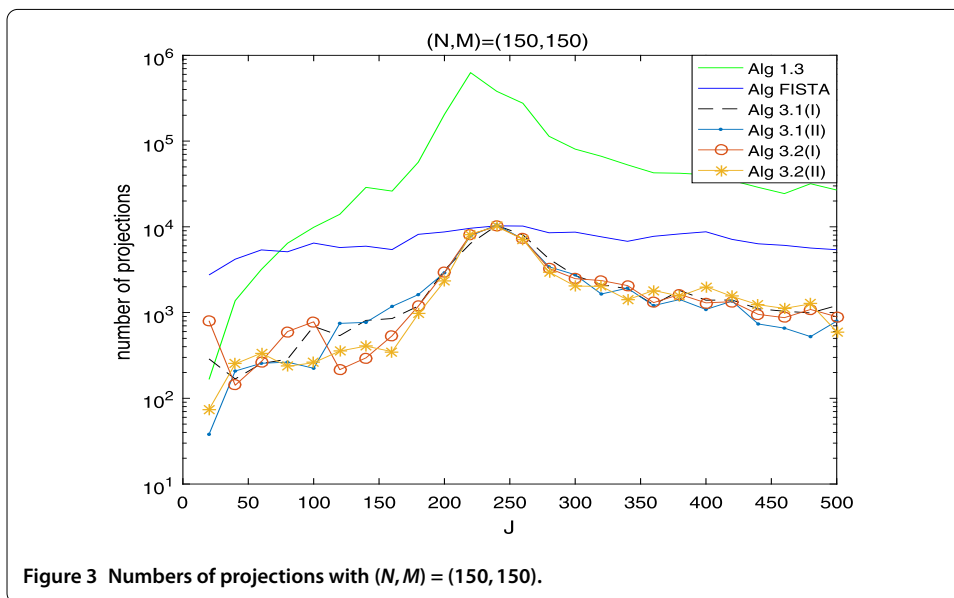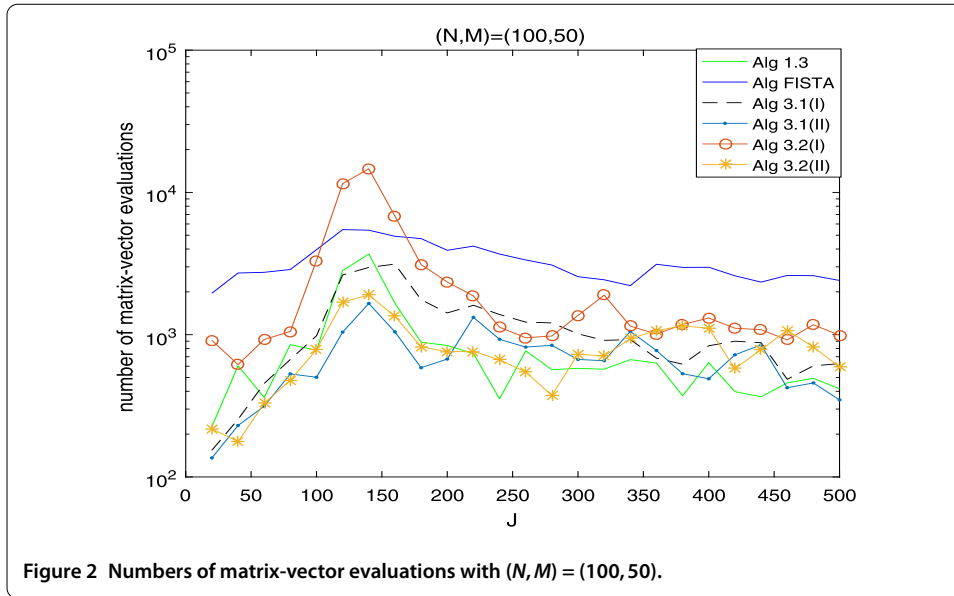
(2) The numbers of projections and matrix-vector evaluations that Algorithms 1.3, 3.1 and 3.2 need are close when M, N are small. However, the numbers of projections and matrix-vector evaluations that Algorithms 3.1 and 3.2 need are less than those of Algorithm 1.3 as M, N become bigger.

(3) In Figures 1, 3 and 5, the number of projections of Algorithm 3.1(I) and (II) (or Algorithm 3.2) is close although the iteration number of Algorithm 3.1(II) is less than that of Algorithm 3.1(I). The reason is that two projections are needed in Algorithm 3.1(II) while one projection is needed in Algorithm 3.1(I) per each iteration.

(4) In Tables 1, 2 and 3, Algorithm 3.1(II) (or Algorithm 3.2(II)) has better performance than Algorithm 3.1(I) (or Algorithm 3.2(I)), maybe because the projections onto C and Q are very simple.

Figure 4 Numbers of matrix-vector evaluations with (N, M) = (150, 150).

Figure 5 Numbers of projections with (N, M) = (200, 250).
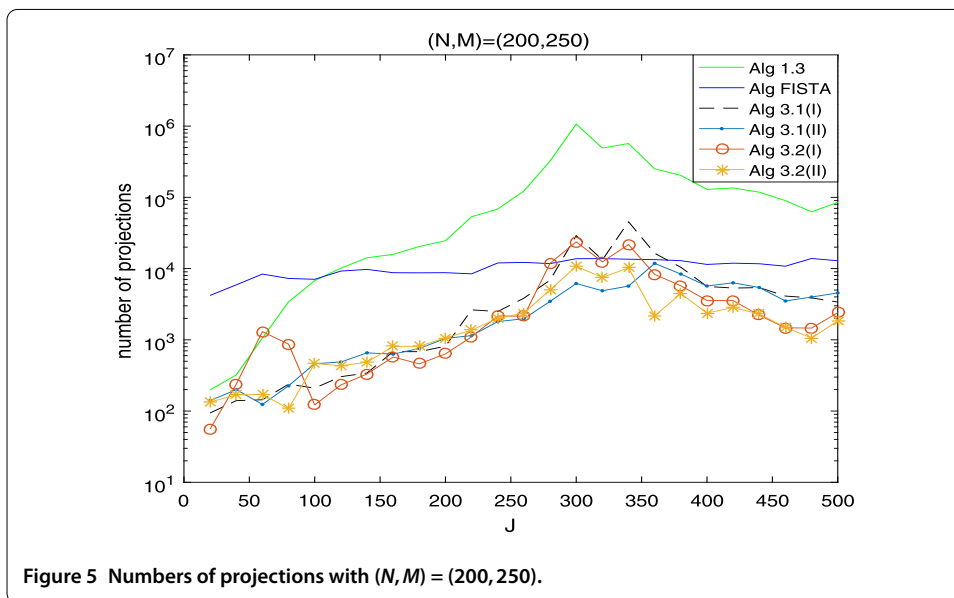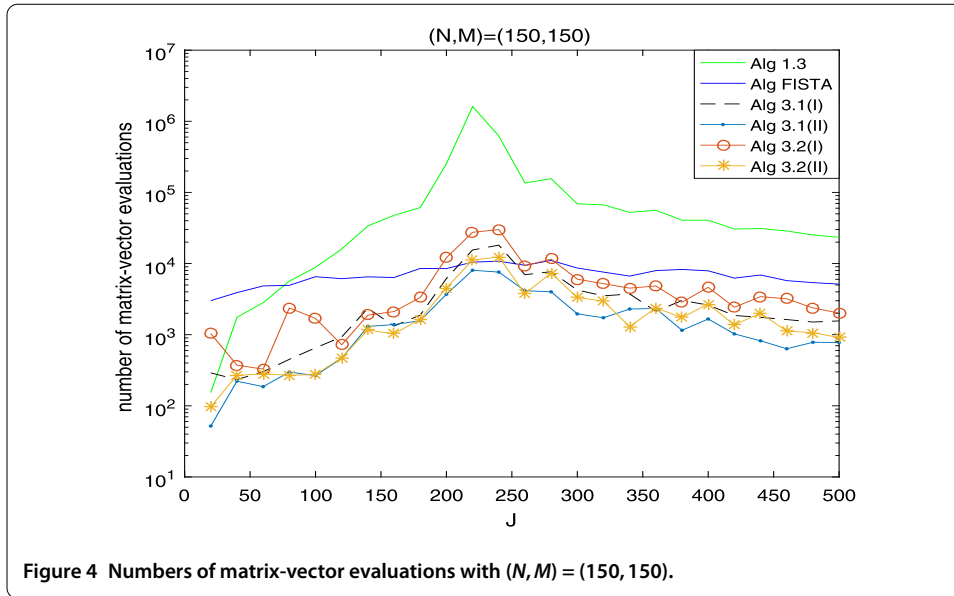
(5) From Figures 1-5, it is observed that there exist peak values for Algorithms 1.3, 3.1 and 3.2, while FISTA has no peak values and is better than Algorithms 1.3, 3.1 and 3.2 near the peak values for some cases. However, for the other values of M, N, J, Algorithms 3.1 and 3.2 behave better than FISTA.

6 Conclusion

In this article we introduce two simultaneous projection algorithms and two semi-alternating projection algorithms to solve the SEP. We present larger stepsizes in (31) and (59) than those in Algorithms 2.1 and 2.2 in [15], which leads to a better contraction property and faster convergence speed of Algorithms 3.1 and 3.2. The weak convergence for the proposed methods is proved under standard conditions.

Figure 6 Numbers of matrix-vector evaluations with (N, M) = (200, 250).
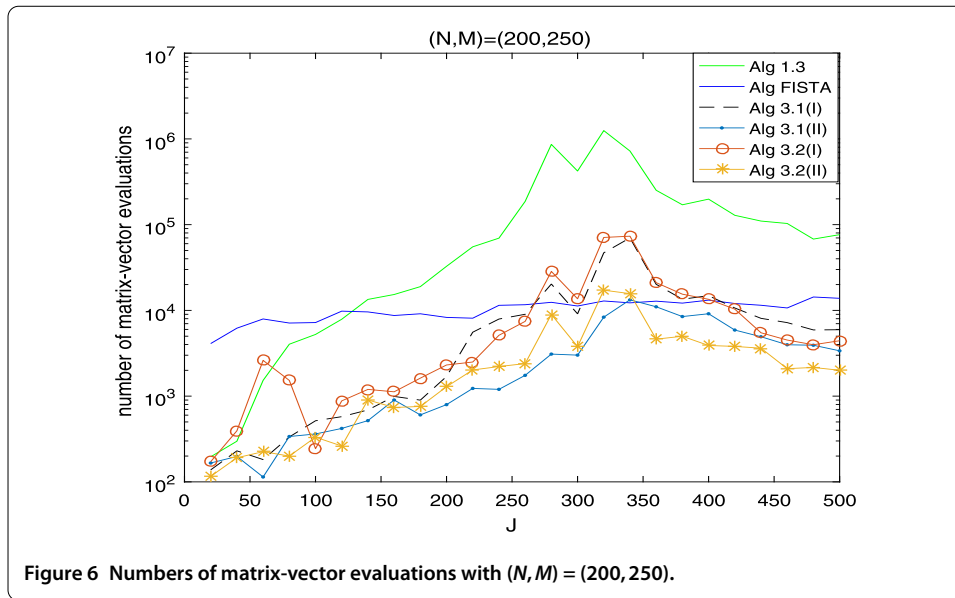
A numerical experiment is provided to illustrate that, except for FISTA, Algorithms 1.3, 3.1 and 3.2 have peak values. It is thus natural to combine our methods with inertial effects. This is one of our future research topics.

Competing interests
The authors declare that they have no competing interests.

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

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

References
 1. Moudafi, A: Alternating CQ-algorithm for convex feasibility and split fixed-point problem. J. Nonlinear Convex Anal. **15**(4), 809-818 (2014)
 2. Attouch, H, Cabot, H, Frankel, F, Peypouquet, J: Alternating proximal algorithms for constrained variational inequalities: application to domain decomposition for PDE's. Nonlinear Anal. **74**(18), 7455-7473 (2011)
 3. Attouch, H, Bolte, J, Redont, P, Soubeyran, A: Alternating proximal algorithms for weakly coupled minimization problems: applications to dynamical games and PDEs. J. Convex Anal. **15**, 485-506 (2008)
 4. Censor, Y, Bortfeld, T, Martin, B, Trofimov, A: A unified approach for inversion problems in intensity-modulated radiation therapy. Phys. Med. Biol. **51**, 2353-2365 (2006)
 5. Attouch, H: Alternating minimization and projection algorithms. From convexity to nonconvexity. Communication in Instituto Nazionale di Alta Matematica Citta Universitaria - Roma, Italy, 8-12 June (2009)
 6. Moudafi, A: A relaxed alternating CQ-algorithm for convex feasibility problems. Nonlinear Anal. **79**, 117-121 (2013)
 7. Tian, D, Shi, L, Chen, R: Iterative algorithm for solving the multiple-sets split equality problem with split self-adaptive step size in Hilbert spaces. J. Inequal. Appl. **2016**, 34 (2016)
 8. Dong, QL, He, SN: Modified projection algorithms for solving the split equality problems. Sci. World J. **2014**, 328787 (2014)
 9. Dong, QL, He, SN: Self-adaptive projection algorithms for solving the split equality problems. Fixed Point Theory **18**(1), 191-202 (2017)
10. Masad, E, Reich, S: A note on the multiple-set split convex feasibility problem in Hilbert space. J. Nonlinear Convex Anal. **8**, 367-371 (2007)

11. Byrne, C, Censor, Y, Gibali, A, Reich, S: The split common null point problem. J. Nonlinear Convex Anal. **13**, 759-775 (2012)
12. Byrne, C, Moudafi, A: Extensions of the CQ algorithm for the split feasibility and split equality problems. Working paper UAG (2013)
13. Moudafi, A, Al-Shemas, E: Simultaneous iterative methods for split equality problems and applications. Trans. Math. Program. Appl. **1**, 1-11 (2013)
14. Korpelevich, GM: The extragradient method for finding saddle points and other problems. Èkon. Mat. Metody **12**, 747-756 (1976)
15. Chuang, CS, Du, WS: Hybrid simultaneous algorithms for the split equality problem with applications. J. Inequal. Appl. **2016**, 198 (2016)
16. Tseng, P: A modified forward–backward splitting method for maximal monotone mappings. SIAM J. Control Optim. **38**, 431-446 (2000)
17. Dong, QL, He, SN, Zhao, J: Solving the split equality problem without prior knowledge of operator norms. Optimization **64**(9), 1887-1906 (2015)
18. Polyak, BT: Some methods of speeding up the convergence of iteration methods. USSR Comput. Math. Math. Phys. **4**, 1-17 (1964)
19. Polyak, BT: Introduction to Optimization. Optimization Software Inc., Publications Division, New York (1987)
20. Dong, QL, Jiang, D: Solve the split equality problem by a projection algorithm with inertial effects. J. Nonlinear Sci. Appl. **10**, 1244-1251 (2017)
21. Beck, A, Teboulle, M: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Imaging Sci. **2**(1), 183-202 (2009)
22. Chambolle, A, Dossal, C: On the convergence of the iterates of the "Fast iterative shrinkage/thresholding algorithm". J. Optim. Theory Appl. **166**, 968-982 (2015)
23. Zhao, J: Solving split equality fixed-point problem of quasi-nonexpansive mappings without prior knowledge of operators norms. Optimization **64**(12), 2619-2630 (2015)
24. Cai, X, Gu, G, He, B: On the $O(1/t)$ convergence rate of the projection and contraction methods for variational inequalities with Lipschitz continuous monotone operators. Comput. Optim. Appl. **57**, 339-363 (2014)
25. Goebel, K, Reich, S: Uniform Convexity, Hyperbolic Geometry, and Nonexpansive Mappings. Dekker, New York (1984)
26. Bauschke, HH, Combettes, PL: Convex Analysis and Monotone Operator Theory in Hilbert Spaces. Springer, Berlin (2011)
27. Rockafellar, RT: Monotone operators and the proximal point algorithm. SIAM J. Control Optim. **14**(5), 877-898 (1976)
28. Cegielski, A: Iterative Methods for Fixed Point Problems in Hilbert Spaces. Lecture Notes in Mathematics. Springer, Berlin (2013)
29. Censor, Y, Elfving, T: A multiprojection algorithm using Bregman projections in a product space. Numer. Algorithms **8**, 221-239 (1994)
30. Censor, Y, Bortfeld, T, Martin, B, Trofimov, A: A unified approach for inversion problems in intensity modulated radiation therapy. Phys. Med. Biol. **51**, 2353-2365 (2006)
31. Cegielski, A, Gibali, A, Reich, S: An algorithm for solving the variational inequality problem over the fixed point set of a quasi-nonexpansive operator in Euclidean space. Numer. Funct. Anal. Optim. **34**, 1067-1096 (2013)
32. Reich, S, Zalas, R: A modular string averaging procedure for solving the common fixed point problem for quasi-nonexpansive mappings in Hilbert space. Numer. Algorithms **72**(2), 297-323 (2016)
33. Aleyner, A, Reich, S: Block-iterative algorithms for solving convex feasibility problems in Hilbert and in Banach spaces. J. Math. Anal. Appl. **343**(1), 427-435 (2008)
34. Gibali, A: A new split inverse problem and application to least intensity feasible solutions. Pure Appl. Funct. Anal. **2**(2), 243-258 (2017)
35. Censor, Y, Gibali, A, Reich, S: Algorithms for the split variational inequality problem. Numer. Algorithms **59**, 301-323 (2012)