Several projection algorithms for solving the split equality problem

QIAO-LI DONG, SONGNIAN HE, HAN-BAO YUAN
Civil Aviation University of China
College of Science
Jinbei Road 2898 , 300300 Tianjin
CHINA
corresponding email: dongql@lsec.cc.ac.cn

Abstract: The split equality problem (SEP) has extraordinary utility and broad applicability in many areas of applied mathematics. Many researchers studied the SEP and proposed algorithms to solve it. However, there are only convergence results of the algorithms in their results and is no estimate on the rate of the convergence. In this paper, we introduce three projection algorithms for solving the split equality problem (SEP), two of which are self-adaptive. The global rate of convergence is firstly investigated. One algorithm is proved to have a global convergence rate $O(1/k)$ and two other algorithms have a global convergence rate $O(1/k^2)$.

Key–Words: Split equality problem; Split feasibility problem; Self-adaptive algorithm; Global rate of convergence; Fast algorithm.

1 Introduction

The split equality problem (SFP) is firstly introduced by Moudafi [1]. Let $C \subset \mathbb{R}^N$, $Q \subset \mathbb{R}^M$ be two nonempty closed convex sets and let A and B be J by N and J by M real matrices, respectively. The split equality problem in [1] is to find

$$x \in C,\ y \in Q,\ \text{such that}\ \ Ax = By, \qquad (1)$$

which allows asymmetric and partial relations between the variables x and y. The interest of the SEP is to cover many situation, for instance in decomposition methods for PDE's, applications in game theory and in intensity-modulated radiation therapy (IMRT). In decision sciences, this allows to consider agents who interplay only via some components of their decision variables (see e.g. [2]). In (IMRT), this amounts to envisage a weak coupling between the vector of doses absorbed in all voxels and that of the radiation intensity (see [3] for further details). Algorithms for solving convex feasibility problems continue to receive great attention; see for instance [4, 5, 6] and also [7, 8, 9, 10, 11, 12]. If $J = M$ and $B = I$, then the convex feasibility problem (1) reduces to the split feasibility problem (originally introduced in Censor and Elfving [13]) which is to find $x \in C$ with $Ax \in Q$.

For solving the SEP (1), Moudafi [1] introduced the following alternating CQ algorithm

$$\begin{cases} x_{k+1} = P_C(x_k - \gamma_k A^*(Ax_k - By_k)), \\ y_{k+1} = P_Q(y_k + \gamma_k B^*(Ax_{k+1} - By_k)), \end{cases} \qquad (2)$$

where $\gamma_k \in (\varepsilon, \min(\frac{1}{\lambda_A}, \frac{1}{\lambda_B}) - \varepsilon)$, λ_A and λ_B are the spectral radius of A^*A and B^*B, respectively. The above alternating CQ algorithm involves two projections P_C and P_Q and hence might be hard to be implemented in the case where one of them fails to have a closed-form expression. So, followed the ideas of Fukushima [14, 15, 16], Moudafi [17] proposed a relaxed alternating CQ-algorithm which only needs projections onto half-spaces. Define the closed convex sets C and Q as level sets:

$$C = \{x \in H_1 : c(x) \le 0\},$$

and

$$Q = \{y \in H_2 : q(y) \le 0\},$$

where $c : H_1 \to \mathbb{R}$ and $q : H_2 \to \mathbb{R}$ are convex functions which are subdifferentiable on C and Q respectively. The relaxed alternating CQ-algorithm is defined by

$$\begin{cases} x_{k+1} = P_{C_k}(x_k - \gamma_k A^*(Ax_k - By_k)), \\ y_{k+1} = P_{Q_k}(y_k + \gamma_k B^*(Ax_{k+1} - By_k)), \end{cases}$$

where (C_k), (Q_k) are two sequences of closed convex sets defined by

$$C_k = \{x \in H_1 : c(x_k) + \langle \xi_k, x - x_k \rangle \le 0\},$$

where $\xi_k \in \partial c(x_k)$, and

$$Q_k = \{y \in H_2 : q(y_k) + \langle \eta_k, y - y_k \rangle \le 0\},$$

where $\eta_k \in \partial q(y_k)$.

By defining product space, Byrne and Moudafi [18] presented a simultaneous iterative algorithm which is also called projected Landweber algorithm):

$$\begin{cases} x_{k+1} = P_C(x_k - \gamma_k A^*(Ax_k - By_k)), \\ y_{k+1} = P_Q(y_k + \gamma_k B^*(Ax_k - By_k)), \end{cases} \tag{3}$$

where $\gamma_k \in (0, \frac{2}{\lambda_A+\lambda_B})$. Compared algorithms (2) and (3), one can find that the alternating CQ algorithm (2) looks like Gauss-Seidel iteration and algorithm (3) looks like Jacobi iteration.

Recently, using Tikhonov regularization (see [19, 20]), Chen et al. [21] made a modification to the algorithm (3) and used the regularization method to establish a single-step iteration:

$$\begin{cases} x_{k+1} = P_C\left((1-\epsilon_k\gamma_k)x_k - \gamma_k A^*(Ax_k - By_k)\right), \\ y_{k+1} = P_Q\left((1-\epsilon_k\gamma_k)y_k + \gamma_k B^*(Ax_k - By_k)\right), \end{cases} \tag{4}$$

for solving the SEP in infinite-dimensional Hilbert spaces. Under the following assumptions on ϵ_k, γ_k:

(i) $0 < \gamma_k \le \epsilon_k/(\|A\|^2 + \|B\|^2 + \epsilon_k)^2$ for all (large enough) k;

(ii) $\epsilon_k \to 0$ and $\gamma_k \to 0$;

(iii) $\sum_{k=1}^{\infty} \epsilon_k\gamma_k = \infty$;

(iv) $(|\gamma_{k+1} - \gamma_k| + \gamma_k|\epsilon_{k+1} - \epsilon_k|)/(\epsilon_{k+1}\gamma_{k+1})^2$;

they showed that the sequence generated by such algorithm strongly converges to the minimum-norm solution of the SEP.

Observe that in the algorithm (2), the determination of the stepsize γ_n depends on the operator (matrix) norms $\|A\|$ and $\|B\|$ (or the largest eigenvalues of A^TA and B^TB). This means that in order to implement the alternating CQ algorithm (2), one has first to compute (or, at least, estimate) operator norms of A and B, which is in general not an easy work in practice.

In [22], inspired by Tseng [23] (also see [24]), the authors proposed a self-adaptive algorithm to solve the SEP.

Algorithm DH: Given constants $\sigma_0 > 0$, $\beta \in (0, 1)$, $\theta \in (0, 1)$, $\rho \in (0, 1)$. Let $x_0 \in H_1$ and $y_0 \in H_2$ be arbitrary. For $k = 0, 1, 2, \ldots$, compute

$$\begin{cases} u_k = P_C(x_k - \tau_k F(x_k, y_k)), \\ v_k = P_Q(y_k - \tau_k G(x_k, y_k)), \end{cases} \tag{5}$$

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

$$\|F(x_k, y_k) - F(u_k, v_k)\|^2 + \|G(x_k, y_k) - G(u_k, v_k)\|^2 \le \theta^2 \frac{\|x_k - u_k\|^2 + \|y_k - v_k\|^2}{\gamma^2}. \tag{6}$$

Construct the half-spaces X_k and Y_k the bounding hyperplane of which support C and Q at u_k and v_k, respectively,

$$X_k := \{u \in H_1 \mid \langle x_k - \tau_k F(x_k, y_k) - u_k, u - u_k\rangle \le 0\},$$
$$Y_k := \{v \in H_2 \mid \langle y_k - \tau_k G(x_k, y_k) - v_k, v - v_k\rangle \le 0\}.$$

Set

$$\begin{cases} x_{k+1} = P_{X_k}(u_k - \gamma_k(F(u_k, v_k) - F(x_k, y_k))), \\ y_{k+1} = P_{Y_k}(v_k - \gamma_k(G(u_k, v_k) - G(x_k, y_k))). \end{cases} \tag{7}$$

If

$$\|F(x_{k+1}, y_{k+1}) - F(x_k, y_k)\|^2 + \|G(x_{k+1}, y_{k+1}) - G(x_k, y_k)\|^2 \le \rho^2 \frac{\|x_{k+1} - x_k\|^2 + \|y_{k+1} - y_k\|^2}{\gamma_k^2},$$

then set $\sigma_k = \sigma_0$; otherwise, set $\sigma_k = \gamma_k$. In [25], the authors introduced a projection algorithm with a way of selecting the stepsizes such that the implementation of the algorithm does not need any priori information about the operator norms for solving the SEP. The stepsize in (3) is taken by:

$$\gamma_k = \sigma_k \min\left\{ \frac{\|Ax_k - By_k\|^2}{\|A^*(Ax_k - By_k)\|^2}, \frac{\|Ax_k - By_k\|^2}{\|B^*(Ax_k - By_k)\|^2} \right\},$$

where $\sigma_k \in (0, 1)$. A relaxed projection algorithm and a viscosity are also discussed and we showed that the sequence generated by the viscosity algorithm converges in norm to the solution of the SEP.

In this paper, we introduce three projection algorithms for solving the SEP, inspired by Beck and Teboulle's iterative shrinkage-thresholding algorithm for linear inverse problem [26]. The first algorithm is self-adaptive, the sequences generated by which are proved to converge to a solution of the SEP. Two other algorithms are fast and proved to have a global convergence rate $O(1/k^2)$. Furthermore, the third algorithm accelerates the first algorithm.

The rest of this paper is organized as follows. In the next section, some useful facts and tools are given. A self-adaptive algorithm is proposed and its global rate of convergence is presented. in section 3. In section 4, we consider two fast algorithms and obtain global rate of convergence. A numerical example is given to illustrate the efficiency of different algorithms in section 5.

2 Preliminaries

Throughout this paper, assume the split equality problem (1) is consistent and denote by Γ the solution of (1), i.e.,

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

then Γ is closed, convex and nonempty.

Let $I = M + N$, $S = C \times Q$ in $\mathbb{R}^N \times \mathbb{R}^M = \mathbb{R}^I$. Define

$$G = [A, -B], \quad u = \begin{bmatrix} x \\ y \end{bmatrix}. \tag{8}$$

The original problem can now be reformulated as finding $u \in S$ with $Gw = 0$, or, more generally, minimizing the function $\|Gu\|$ over $u \in S$ which can be written as the following minimization problem:

$$\min_{u\in\mathbb{R}^I} \{F(u) \equiv f(u) + \iota_S(u)\},$$

where $f(u) = \frac{1}{2}\|Gu\|^2$ and $\iota_S(u)$ is a indicator function of the set S defined by

$$\iota_S(u) = \begin{cases} 0, & u \in S \\ +\infty, & \text{otherwise.} \end{cases}$$

It is easy to verified that $L(f) = \|\nabla f\| = \|A\|^2 + \|B\|^2$.

For any $\tau > 0$, consider the following quadratic approximation of $Q_\tau(u)$ at a given point v:

$$Q_\tau(u, v) := f(v)+\langle u-v, \nabla f(v)\rangle+\frac{\tau}{2}\|u-v\|^2+\iota_S(u),$$

which admits a unique minimizer

$$p_\tau(v) := \arg\min\{Q_\tau(u, v) : u \in \mathbb{R}^I\}. \tag{9}$$

Simple algebra shows that (ignoring constant terms in v)

$$\begin{aligned} p_\tau(v) &= \arg\min_u \left\{ \iota_S(u) + \frac{\tau}{2}\left\| u - \left(v - \frac{1}{\tau}\nabla f(v)\right)\right\|^2 \right\} \\ &= P_S(v - \frac{1}{\tau}\nabla f(v)). \end{aligned} \tag{10}$$

The following lemma is well-known and fundamental property for a smooth function in the class $C^{1,1}$; e.g., [27, 28].

Lemma 1 *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable function with Lipschitz continuous gradient and Lipschitz constant $L(f)$. Then, for any $L > L(f)$,*

$$f(x) \le f(y) + \langle x - y, \nabla f(y)\rangle + \frac{L}{2}\|x - y\|^2,$$

for every $x, y \in \mathbb{R}^n$.

The following lemma is key for the proof of the main result.

Lemma 2 *Let $y \in \mathbb{R}^n$ and $\tau > 0$ be such that*

$$F(p_\tau(y)) \le Q_\tau(p_\tau(y), y). \tag{11}$$

Then for any $x \in \mathbb{R}^n$,

$$F(x)-F(p_\tau(y)) \ge \frac{\tau}{2}\|p_\tau(y)-y\|^2+\tau\langle y-x, p_\tau(y)-y\rangle.$$

Proof. See the appendix.

Remark 3 *Note that from Lemma 1, it follows that if $\tau \ge L(f)$, then the condition (11) is always satisfied for $p_\tau(v)$.*

3 A self-adaptive algorithm

Although Byrne and Moudafi [18] proved the convergence of the CQ algorithm (3), there was no estimate of the rate of convergence. Here we rewrite the CQ algorithm (3) and consider the rate of the convergence of CQ algorithm together with a self-adaptive algorithm.

Algorithm 4 *Let $L_1 \ge L(f)$ be a fixed constant and given $\tau_k \in (L(f), L_1)$. Let x_1 be arbitrary. For $k = 1, 2, \dots,$ compute*

$$u_{k+1} = P_S(u_k - \frac{1}{\tau_k}\nabla f(u_k)). \tag{12}$$

Using (8) and expressing (12) in terms of x and y, we obtain the following equivalent form of Algorithm 4:

Algorithm 4* *Let $L_1 \ge L(f)$ be a fixed constant and given $\tau_k \in (L(f), L_1)$. Let x_1 be arbitrary. For $k = 1, 2, \dots,$ compute*

$$\begin{cases} x_{k+1} = P_C(x_k - \frac{1}{\tau_k}A^T(Ax_k - By_k)), \\ y_{k+1} = P_Q(y_k + \frac{1}{\tau_k}B^T(Ax_k - By_k)). \end{cases} \tag{13}$$

It should be noted that in the above algorithm (13), we take $\tau_k \ge \frac{1}{\|A\|^2+\|B\|^2}$, instead of $\tau_k \ge \frac{2}{\|A\|^2+\|B\|^2}$ (as in Byrne and Moudafi's algorithm) which is restricted to a smaller range.

Next, we propose a self-adaptive algorithm which solve the SEP (1) without prior knowledge of spectral radius of the matrices A^TA and B^TB. The sequence generated by the algorithm converges to a solution of the SEP and the global rate of convergence is presented.

Algorithm 5 *Given $\gamma > 0$ and $\eta > 1$. Let u_1 be arbitrary. For $k = 1, 2, \ldots$, find the smallest nonnegative integer m_k such that $\tau_k = \gamma\eta^{m_k}$ and*

$$u_{k+1} = P_S(u_k - \frac{1}{\tau_k}\nabla f(u_k)), \tag{14}$$

which satisfies

$$\begin{aligned} f(u_{k+1}) - f(u_k) + \langle \nabla f(u_k), u_k - u_{k+1}\rangle \\ \leq \frac{\tau_k}{2}\|u_k - u_{k+1}\|^2. \end{aligned} \tag{15}$$

Using (8) and expressing (14) and (15) in terms of x and y, we obtain the following equivalent form of Algorithm 5:

Algorithm 5* *Given $\gamma > 0$ and $\eta > 1$. Let (x_1, y_1) be arbitrary. For $k = 1, 2, \ldots$, find the smallest nonnegative integer m_k such that $\tau_k = \gamma\eta^{m_k}$ and*

$$\begin{cases} x_{k+1} = P_C(x_k - \frac{1}{\tau_k}A^T(Ax_k - By_k)), \\ y_{k+1} = P_Q(y_k + \frac{1}{\tau_k}B^T(Ax_k - By_k)), \end{cases} \tag{16}$$

which satisfies

$$\begin{aligned} &\frac{1}{2}\|Ax_{k+1} - By_{k+1}\|^2 - \frac{1}{2}\|Ax_k - By_k\|^2 \\ &+ A^T(Ax_k - By_k)(x_k - x_{k+1}) \\ &+ B^T(By_k - Ax_k)(y_k - y_{k+1}) \\ &\leq \frac{\tau_k}{2}\|x_k - x_{k+1}\|^2 + \frac{\tau_k}{2}\|y_k - y_{k+1}\|^2. \end{aligned} \tag{17}$$

Remark 6 *Note that the sequence of function values $\{f(u_k)\}$ produced by the algorithm 5 is nonincreasing. Indeed, for every $n \geq 1$,*

$$f(u_{k+1}) \leq Q_{\tau_k}(u_{k+1}, u_k) \leq Q_{\tau_k}(u_k, u_k) = f(x_k), \tag{18}$$

where the first inequality comes from (15), and the second inequality follows from (9). τ_n in (18) is chosen by the backtracking rule (15).

Lemma 7

$$\beta L(f) \leq \tau_k \leq \alpha L(f). \tag{19}$$

where $\alpha = \frac{L_1}{L(f)}, \beta = 1$ in Algorithm 4 and $\alpha = \eta, \beta = \frac{\gamma}{L(f)}$ in Algorithm 5.

Proof: It is easy to verify (19) for Algorithm 4. By $\eta > 1$ and the choice of τ_k, we get $\tau_k \geq \gamma$. From Lemma 1, it follows that inequality (17) is satisfied for $\tau_k \geq L(f)$, where $L(f)$ is the Lipschitz constant of ∇f. So, for Algorithm 5 one has $\tau_k \leq \eta L(f)$ for every $k \geq 1$. □

Theorem 8 *Let $\{x_k, y_k\}$ be a sequence generated by Algorithm 4* and 5*. Then the sequence $\{x_k, y_k\}$ converges to a solution of the SEP (1), and furthermore for any $k \geq 1$ it holds that*

$$\begin{aligned} &\|Ax_k - By_k\|^2 \\ &\leq \frac{\eta(\|A\|^2 + \|B\|^2)(\|x_0 - x^*\|^2 + \|y_0 - y^*\|^2)}{k}, \\ &\forall (x^*, y^*) \in \Gamma. \end{aligned} \tag{20}$$

Proof: Let $u_n = (x_n, y_n)$ and $u^* = (x^*, y^*)$. Invoking Lemma 2 with $x = u^*, y = u_n$, and $\tau = \tau_n$, we obtain

$$\begin{aligned} &-\frac{2}{\tau_k}f(u_{n+1}) \\ &\geq \|u_{n+1} - u_n\|^2 + 2\langle u_n - u^*, u_{n+1} - u_n\rangle \\ &= \|u_{n+1} - u^*\|^2 - \|u_n - u^*\|^2, \end{aligned}$$

which combined with (19) and the fact that $f(u^*) = 0$ and $f(u_{n+1}) \geq 0$ yields

$$\frac{2}{\eta L(f)}(f(u^*) - f(u_{n+1})) \geq \|u_{n+1} - u^*\|^2 - \|u_n - u^*\|^2, \tag{21}$$

which implies

$$\|u_{n+1} - u^*\| \leq \|u_n - u^*\|.$$

It follows that $\lim_{k\to\infty}\|u_k - u^*\|$ exists and thus $\lim_{k\to\infty}\|x_k - x^*\|$ and $\lim_{k\to\infty}\|y_k - y^*\|$ exist. Summing the inequality (21) over $n = 0, 1, \ldots, k-1$ gives

$$-\frac{2}{\eta L(f)}\sum_{n=0}^{k-1} f(u_{n+1}) \geq \|u_k - u^*\|^2 - \|u_0 - u^*\|^2. \tag{22}$$

Invoking Lemma 2 one more time with $x = y = u_n$ and $\tau = \tau_n$ yields

$$\frac{2}{\tau_n}(f(u_n) - f(u_{n+1})) \geq \|u_n - u_{n+1}\|^2. \tag{23}$$

Since $\tau_n \geq \gamma$ (see (19)) and $f(u_n) - f(u_{n+1}) \geq 0$ (see (18)), it follows that

$$\frac{2}{\gamma}(f(u_n) - f(u_{n+1})) \geq \|u_n - u_{n+1}\|^2.$$

Multiplying the last inequality by n and summing over $n = 0, \ldots, k-1$, we obtain

$$\begin{aligned} &\frac{2}{\gamma}\sum_{n=0}^{k-1}(nf(u_n) - (n+1)f(u_{n+1}) + f(u_{n+1})) \\ &\geq \sum_{n=0}^{k-1} n\|u_n - u_{n+1}\|^2, \end{aligned}$$

which simplifies to

$$\frac{2}{\gamma}\left(-kf(u_k)+\sum_{n=0}^{k-1}f(u_{n+1})\right)\geq\sum_{n=0}^{k-1}n\|u_n-u_{n+1}\|^2. \tag{24}$$

Adding (22) and (24) times $\frac{\gamma}{\eta L(f)}$, we get

$$\begin{aligned}&-\frac{2k}{\eta L(f)}f(u_k)\\&\geq\|u_k-u^*\|^2+\frac{\gamma}{\eta L(f)}\sum_{n=0}^{k-1}n\|u_n-u_{n+1}\|^2\\&\quad-\|u_0-u^*\|^2,\end{aligned}$$

and hence it follows that

$$f(u_k)\leq\frac{\eta L(f)\|u_0-u^*\|^2}{2k},\quad\forall u^*\in\Gamma, \tag{25}$$

which yields (20) and

$$\lim_{k\to\infty}f(u_k)=0, \tag{26}$$

i.e.,

$$\lim_{k\to\infty}\|Ax_k-By_k\|=0. \tag{27}$$

Using (23) and (26) and expressing them in x_k and y_k, we obtain

$$\lim_{k\to\infty}\|x_k-x_{k+1}\|=0,\quad\lim_{k\to\infty}\|y_k-y_{k+1}\|=0.$$

Let $(\hat{x},\hat{y})\in\omega_w(x_k,y_k)$, then there exist two subsequences of (x_k) and (y_k) (again labeled (x_k) and (y_k)) which converge weakly to $\hat{x}$ and $\hat{y}$. Note that the two equalities in (13) can be rewritten as

$$\begin{cases}\tau_k(x_k-x_{k+1})-A^T(Ax_k-By_k)\in N_C(x_{k+1}),\\\tau_k(y_k-y_{k+1})+B^T(Ax_k-By_k)\in N_Q(x_{k+1}),\end{cases} \tag{28}$$

where N_C and N_Q is the normal cone to the convex sets C and Q, respectively. The graphs of the maximal monotone operators N_C, N_Q are weakly-strongly closed and by passing to the limit in the last inclusions, we obtain that

$$\hat{x}\in C\quad\text{and}\quad\hat{y}\in Q.$$

Furthermore, the weak convergence of (Ax_k-By_k) to $A\hat{x}-B\hat{y}$ and lower semicontinuity of the squared norm imply

$$\|A\hat{x}-B\hat{y}\|\leq\liminf_{k\to\infty}\|Ax_k-By_k\|=0,$$

where (27) is used. Hence $(\hat{x},\hat{y})\in\Gamma$.

To show the uniqueness of the weak cluster points, we will use the same strick as in the celebrated Opial Lemma. Now, by setting

$$\Gamma_k(x^*,y^*)=\|x_k-x^*\|^2+\|y_k-y^*\|^2,$$

the existence of $\lim_{k\to\infty}\|x_k-x^*\|$ and $\lim_{k\to\infty}\|y_k-y^*\|$ implies the existence of $\lim_{k\to\infty}\Gamma_k(x^*,y^*)$, which is denoted by $l(\hat{x},\hat{y})$. Indeed, let $(\bar{x},\bar{y})$ be other weak cluster point of (x_k,y_k). By passing to the limit in the relation

$$\begin{aligned}\Gamma_k(\hat{x},\hat{y})&=\Gamma_k(\bar{x},\bar{y})+\|\hat{x}-\bar{x}\|^2+\|\hat{y}-\bar{y}\|^2\\&\quad+2\langle x_k-\bar{x},\bar{x}-\hat{x}\rangle+2\langle y_k-\bar{y},\bar{y}-\hat{y}\rangle,\end{aligned}$$

we obtain

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

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

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

By adding the two last equalities, we obtain

$$\|\hat{x}-\bar{x}\|^2+\|\hat{y}-\bar{y}\|^2=0.$$

Hence $(\hat{x},\hat{y})=(\bar{x},\bar{y})$, this implies that the whole sequence (x_k,y_k) converges to a solution of the SEP (1), which completes the proof. □

4 Two fast algorithms

In this section, we introduce two fast projection algorithms. The global rate of convergence of the two algorithms are investigated and the sequence $\{f(u_k)\}$ has the better complexity rate $O(1/k^2)$.

Firstly, we present an algorithm which accelerates the algorithm 4.

Algorithm 9 *Let $L_1\geq L(f)$ be a fixed constant and given $\tau_k\in(L(f),L_1)$. Let u_0 be arbitrary and set $v_1=u_0,t_1=1$. For $k=1,2,\ldots,$ compute*

$$u_k=P_S(v_k-\frac{1}{\tau_k}\nabla f(v_k)), \tag{29}$$

$$t_{k+1}=\frac{1+\sqrt{1+4t_k^2}}{2}. \tag{30}$$

Set

$$v_{k+1}=u_k+\left(\frac{t_k-1}{t_{k+1}}\right)(u_k-u_{k-1}). \tag{31}$$

Using (8) and expressing (29) and (31) in terms of x and y, we obtain the following equivalent form of Algorithm 9:

Algorithm 9* *Let $L_1 \geq L(p)$ be a fixed constant and given $\tau_k \in (L(p), L_1)$. Let (x_0, y_0) be arbitrary and set $(s_1, w_1) = (x_0, y_0), t_1 = 1$. For $k = 1, 2, \ldots,$ compute*

$$\begin{cases} x_k = P_C(s_k - \frac{1}{\tau_k}A^T(As_k - Bw_k)), \\ y_k = P_Q(w_k + \frac{1}{\tau_k}B^T(As_k - Bw_k)), \end{cases} \tag{32}$$

$$t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2}. \tag{33}$$

Set

$$\begin{cases} s_{k+1} = x_k + \left(\frac{t_k - 1}{t_{k+1}}\right)(x_k - x_{k-1}), \\ w_{k+1} = y_k + \left(\frac{t_k - 1}{t_{k+1}}\right)(y_k - y_{k-1}). \end{cases} \tag{34}$$

The following algorithm is self-adaptive and accelerates Algorithm 5.

Algorithm 10 *Given $\gamma > 0$ and $\eta > 1$. Let u_0 be arbitrary and set $v_1 = u_0, t_1 = 1$. For $k = 1, 2, \ldots,$ find the smallest nonnegative integer m_k such that $\tau_k = \gamma\eta^{m_k}$ and*

$$u_k = P_S(v_k - \frac{1}{\tau_k}\nabla f(v_k)), \tag{35}$$

which satisfies

$$f(u_k) - f(v_k) + \langle \nabla f(v_k), v_k - u_k \rangle \leq \frac{\tau_k}{2}\|v_k - u_k\|^2. \tag{36}$$

Compute

$$t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2}, \tag{37}$$

and

$$v_{k+1} = u_k + \left(\frac{t_k - 1}{t_{k+1}}\right)(u_k - u_{k-1}). \tag{38}$$

Using (8) and expressing (35), (36) and (38) in terms of x and y, we obtain the following equivalent form of Algorithm 10:

Algorithm 10* *Given $\gamma > 0$ and $\eta > 1$. Let (x_0, y_0) be arbitrary and set $(s_1, w_1) = (x_0, y_0), t_1 = 1$. For $k = 1, 2, \ldots,$ find the smallest nonnegative integer m_k such that $\tau_k = \gamma\eta^{m_k}$ and*

$$\begin{cases} x_k = P_C(s_k - \frac{1}{\tau_k}A^T(As_k - Bw_k)), \\ y_k = P_Q(w_k + \frac{1}{\tau_k}B^T(As_k - Bw_k)), \end{cases} \tag{39}$$

which satisfies

$$\begin{aligned} &\frac{1}{2}\|Ax_k - By_k\|^2 - \frac{1}{2}\|As_k - Bw_k\|^2 \\ &+ A^T(As_k - Bw_k)(s_k - x_k) \\ &+ B^T(Bw_k - As_k)(w_k - y_k) \\ &\leq \frac{\tau_k}{2}\|x_k - s_k\|^2 + \frac{\tau_k}{2}\|y_k - w_k\|^2. \end{aligned} \tag{40}$$

Compute

$$t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2}, \tag{41}$$

and

$$\begin{cases} s_{k+1} = x_k + \left(\frac{t_k - 1}{t_{k+1}}\right)(x_k - x_{k-1}), \\ w_{k+1} = y_k + \left(\frac{t_k - 1}{t_{k+1}}\right)(y_k - y_{k-1}). \end{cases} \tag{42}$$

Lemma 11

$$\beta L(f) \leq \tau_k \leq \alpha L(f). \tag{43}$$

where $\alpha = \frac{L_1}{L(f)}, \beta = 1$ in Algorithm 9 and $\alpha = \eta, \beta = \frac{\gamma}{L(f)}$ in Algorithm 10.

Proof: Following the line of Lemma 7, one can easily show (43) for Algorithm 9 and 10. □

The next result provides the key recursive relation for the sequence $\{f(u_k)\}$.

Lemma 12 *The sequence $\{u_k\}$ generated via Algorithm 9 or Algorithm 10 satisfies, for every $k \geq 1$*

$$\frac{2}{\tau_k}t_k^2 r_k - \frac{2}{\tau_{k+1}}t_{k+1}^2 r_{k+1} \geq \|q_{k+1}\|^2 - \|q_k\|^2,$$

where $r_k := f(u_k)$, $q_k := t_k u_k - (t_k - 1)u_{k-1} - u^*$ with $u^* = (x^*, y^*) \in \Gamma$.

Proof. See Appendix. □

We also need the following trivial facts.

Lemma 13 *Let $\{a_k, b_k\}$ be positive sequences of reals satisfying*

$$a_k - a_{k+1} \geq b_{k+1} - b_k, \quad \forall k \geq 1, \text{ with } a_1 + b_1 \leq c,\ c > 0.$$

Then $a_k \leq c$ for every $k \geq 1$.

Lemma 14 *The positive sequence $\{t_k\}$ generated in (33) with $t_1 = 1$ satisfies $t_k \geq (k+1)/2$ for all $k \geq 1$.*

Theorem 15 *Let $\{u_n\}$ be generated by Algorithm 9 or Algorithm 10. Then for any $k \geq 1$*

$$f(u_k) \leq \frac{2\alpha L(f)\|u_0 - u^*\|^2}{(k+1)^2}, \quad \forall u^* \in \Gamma. \tag{44}$$

Proof: Let us define the quantities

$$a_k := \frac{2}{\tau_k} t_k^2 r_k, \quad b_k := \|q_k\|^2,$$
$$c := \|v_1 - u^*\|^2 = \|u_0 - u^*\|^2,$$

and recall (cf. Lemma 12) that $r_k := f(u_k)$. Then, by Lemma 12 we have for every $n \geq 1$

$$a_k - a_{k+1} \geq b_{k+1} - b_k,$$

and hence assuming that $a_1 + b_1 \leq c$ holds true, invoking Lemma 13, we obtain that

$$\frac{2}{\tau_k} t_k^2 r_k \leq \|u_0 - u^*\|^2,$$

which combined with $t_k \geq (k+1)/2$ (by Lemma 14) yields

$$r_k \leq \frac{2\tau_k\|u_0 - u^*\|^2}{(k+1)^2}.$$

Utilizing the upper bound on τ_k given in (43), the desired result (44) follows. Thus, all that remains is to prove the validity of the relation $a_1 + b_1 \leq c$. Since $t_1 = 1$, and using the definition of q_k given in Lemma 12, we have here

$$a_1 = \frac{2}{\tau_1} t_1^2 r_1 = \frac{2}{\tau_1} r_1, \quad b_1 = \|q_1\|^2 = \|u_1 - u^*\|^2.$$

Applying Lemma 2 to the points $x := u^*, y := v_1$ with $\tau = \tau_1$, we get

$$f(u^*) - f(x_1) \geq \frac{\tau_1}{2}\|u_1 - v_1\|^2 + \tau_1\langle v_1 - u^*, u_1 - v_1\rangle. \tag{45}$$

Thus, using $f(u^*) = 0$, we obtain

$$\begin{aligned} -f(u_1) &\geq \frac{\tau_1}{2}\|u_1 - v_1\|^2 + \tau_1\langle v_1 - u^*, u_1 - v_1\rangle \\ &= \frac{\tau_1}{2}\{\|u_1 - u^*\|^2 - \|v_1 - u^*\|^2\}. \end{aligned}$$

Consequently,

$$\frac{2}{\tau_1} r_1 \leq \|v_1 - u^*\|^2 - \|u_1 - u^*\|^2,$$

that is, $a_1 + b_1 \leq c$ holds true. □

Remark 16 *From Theorem 15, we have, for (x_k, y_k) generated by algorithms 9* and 10*,*

$$\begin{aligned} &\|Ax_k - By_k\|^2 \\ &\leq \frac{4\alpha(\|A\|^2 + \|B\|^2)(\|x_0 - x^*\|^2 + \|y_0 - y^*\|^2)}{(k+1)^2}, \\ &\forall (x^*, y^*) \in \Gamma. \end{aligned}$$

Remark 17 *Different from Theorem 8, there is not convergence of the sequence $\{u_n\}$ in Theorem 15 for algorithms 9 and 10. Combettes and Pesquet [29] concluded that the convergence of the sequence $\{u_n\}$ generated by Algorithm 9 or Algorithm 10 is no longer guaranteed in general.*

5 Preliminary computational results

In this section, we present some preliminary numerical results. We apply four algorithms to solve an example, and compare the numerical results.

For convenience, 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 tables, 'Iter.' and 'Sec.' denoted the number of iterations and the cpu time in seconds, respectively.

Example 18 *The SEP with $A = (a_{ij})_{J\times N}$ $B = (b_{ij})_{J\times M}$, $C = \{x \in \mathbb{R}^N | \|x\| \leq 0.25\}$, $Q = \{y \in \mathbb{R}^M | \, e_0 \leq y \leq U\}$, where $a_{ij} \in [0,1]$, $b_{ij} \in [0,1]$ and $U \in [e_1, 2e_1]$ are all generated randomly. In the implementation, we took $\|Ax - By\| < \varepsilon = 10^{-4}$ as the stopping criterion. Take the initial value $x = (0,0,\ldots,0)^T \in \mathbb{R}^N$, $y = (1,1,\ldots,1)^T \in \mathbb{R}^M$.*

We tested the algorithms 4, 5, 9 and 10 with different M, N and J. In algorithms 4 and 9, since a smaller τ_n was more efficient than a larger one, we chose $\tau_n = L(f)$ in the experiment. We took $\gamma = 9$, $\eta = 4$ for algorithms 5 and 10. For comparison, the same random values were taken in each test for four algorithms. The numerical results were listed in table 1, from which we could observe the efficiency of the algorithms 4,5, 9 and 10, both from the points of view of number of iterations and cpu time. We found that the algorithm 9, in fact, accelerated the algorithm 4 and the algorithm 10 accelerated the algorithm 5 at most cases except several special cases, which deserves further research.

Acknowledgements: The research was supported by NSFC (No. 11201476) and Fundamental Research Funds for the Central Universities (No. 3122013D017).

Table 1: Computational results for the example with different J, M, N.

		J	10	30	50
$N = 10$	Alg 4	Iter.	2394	12030	3653
		Sec.	0.125	1.030	0.530
	Alg 5	InIt.	946	2884	3273
		Iter.	941	2201	1507
		Sec.	0.140	0.499	0.936
$M = 20$	Alg 8	Iter.	277	591	387
		Sec.	0.016	0.078	0.078
	Alg 9	InIt.	287	792	563
		Iter.	193	333	227
		Sec.	0.094	0.140	0.187
$N = 30$	Alg 4	Iter.	737	5788	83945
		Sec.	0.062	0.827	19.017
	Alg 5	InIt.	158	4683	18785
		Iter.	153	2015	9308
		Sec.	0.047	1.108	7.207
$M = 30$	Alg 8	Iter.	216	864	1414
		Sec.	0.031	0.140	0.374
	Alg 9	InIt.	503	703	6688
		Iter.	211	304	1958
		Sec.	0.109	0.187	2.512
$N = 100$	Alg 4	Iter.	846	2553	7746
		Sec.	0.125	0.406	1.030
	Alg 5	InIt.	183	791	7363
		Iter.	111	296	2278
		Sec.	0.078	0.281	2.777
$M = 50$	Alg 8	Iter.	221	473	904
		Sec.	0.062	0.094	0.125
	Alg 9	InIt.	223	2895	1580
		Iter.	88	830	451
		Sec.	0.062	1.014	0.655

Appendix

The proof of Lemma 2. From (11), we have

$$p(x) - p(F_\tau(y)) \geq p(x) - R_\tau(F_\tau(y), y). \quad (46)$$

Now, from the fact that p is convex, it follows

$$p(x) \geq p(y) + \langle x - y, \nabla p(y) \rangle. \quad (47)$$

On the other hand, by the definition of $R_\tau(x, y)$, one has

$$\begin{aligned} R_\tau(F_\tau(y), y) = p(y) &+ \langle F_\tau(y) - y, \nabla p(y) \rangle \\ &+ \frac{\tau}{2}\|F_\tau(y) - y\|^2. \end{aligned} \quad (48)$$

Therefore, using (46)-(48), it follows that

$$\begin{aligned} &p(x) - p(F_\tau(y)) \\ &\geq -\frac{\tau}{2}\|F_\tau(y) - y\|^2 + \langle x - F_\tau(y), \nabla p(y) \rangle \\ &= -\frac{\tau}{2}\|F_\tau(y) - y\|^2 + \tau\langle x - F_\tau(y), y - F_\tau(y) \rangle \\ &= \frac{\tau}{2}\|F_\tau(y) - y\|^2 + \tau\langle y - x, F_\tau(y) - y \rangle, \end{aligned}$$

where in the first equality above we used (10). □

The proof of Lemma 12. First we apply Lemma 2 at the points $(x := x_n, y := y_{n+1})$ with $\tau = \tau_{n+1}$, and likewise at the points $(x := x^*, y := y_{n+1})$, to get

$$\begin{aligned} &2\tau_{n+1}^{-1}(v_n - v_{n+1}) \\ &\geq \|x_{n+1} - y_{n+1}\|^2 + 2\langle x_{n+1} - y_{n+1}, y_{n+1} - x_n \rangle, \\ &- 2\tau_{n+1}^{-1}v_{n+1} \\ &\geq \|x_{n+1} - y_{n+1}\|^2 + 2\langle x_{n+1} - y_{n+1}, y_{n+1} - x^* \rangle, \end{aligned}$$

where we used the fact that $p(x^*) = 0$ and $x_{n+1} = F_{\tau_{n+1}}(y_{n+1})$. To get a relation between v_n and v_{n+1}, we multiply the first inequality above by $(t_{n+1} - 1)$ and add it to the second inequality:

$$\begin{aligned} &\frac{2}{\tau_{n+1}}((t_{n+1} - 1)v_n - t_{n+1}v_{n+1}) \\ &\geq t_{n+1}\|x_{n+1} - y_{n+1}\|^2 \\ &+ 2\langle x_{n+1} - y_{n+1}, t_{n+1}y_{n+1} - (t_{n+1} - 1)x_n - x^* \rangle. \end{aligned}$$

Multiplying the last inequality by t_{n+1} and using the relation $t_n^2 = t_{n+1}^2 - t_{n+1}$ which holds thanks to (33), we obtain

$$\begin{aligned} &\frac{2}{\tau_{n+1}}(t_n^2 v_n - t_{n+1}^2 v_{n+1}) \\ \geq\ &\|t_{n+1}(x_{n+1} - y_{n+1})\|^2 \\ &+2t_{n+1}\big\langle x_{n+1} - y_{n+1}, \\ &\qquad t_{n+1}y_{n+1} - (t_{n+1} - 1)x_n - x^* \big\rangle. \end{aligned}$$

Applying the usual Pythagoras relation

$$\|b - a\|^2 + 2\langle b - a, a - c \rangle = \|b - c\|^2 - \|a - c\|^2,$$

to the right-hand side of the last inequality with

$$\begin{aligned} &a := t_{n+1}y_{n+1}, \quad b := t_{n+1}x_{n+1}, \\ &c := (t_{n+1} - 1)x_n + x^*, \end{aligned}$$

we thus get

$$\begin{aligned} &\frac{2}{\tau_{n+1}}(t_n^2 v_n - t_{n+1}^2 v_{n+1}) \\ &\geq \|t_{n+1}x_{n+1} - (t_{n+1} - 1)x_n - x^*\|^2 \\ &\quad - \|t_{n+1}y_{n+1} - (t_{n+1} - 1)x_n - x^*\|^2. \end{aligned}$$

Therefore, with y_{n+1} (cf. (31)) and u_n defined by

$$t_{n+1}y_{n+1} = t_{n+1}x_n + (t_n - 1)(x_n - x_{n-1}),$$

and

$$u_n = t_n x_n - (t_n - 1)x_{n-1} - x^*,$$

it follows that

$$\frac{2}{\tau_{n+1}}(t_n^2 v_n - t_{n+1}^2 v_{n+1}) \geq \|u_{n+1}\|^2 - \|u_n\|^2,$$

which combined with the inequality $\tau_{n+1} \geq \tau_n$ yields

$$\frac{2}{\tau_n} t_n^2 v_n - \frac{2}{\tau_{n+1}} t_{n+1}^2 v_{n+1} \geq \|u_{n+1}\|^2 - \|u_n\|^2.$$

The proof is completed. □

References:

[1] A. Moudafi, Alternating CQ-algorithm for convex feasibility and split fixed-point problems, *J. Nonlinear and Convex A.* (submitted for publication).

[2] H. Attouch, J. Bolte, P. Redont, A. Soubeyran, Alternating proximal algorithms for weakly coupled minimization problems. Applications to dynamical games and PDEs, *J. Convex Anal.*, 15, 2008, pp. 485–506.

[3] Y. Censor, T. Bortfeld, B. Martin, A. Trofimov, A unified approach for inversion problems in intensity-modulated radiation therapy, *Phys. Med. Biol.*, 51, 2006, pp. 2353–2365.

[4] A. Aleyner, S. Reich, Block-iterative algorithms for solving convex feasibility problems in Hilbert and in Banach, *J. Math. Anal. Appl.*, 343(1), 2008, pp. 427–435.

[5] H. H. Bauschke, J. M. Borwein, On projection algorithms for solving convex feasibility problems, *SIAM Rev.*, 38, 1996, pp. 367–426.

[6] C. Byrne, *A unified treatment of some iterative algorithms in signal processing and image reconstruction,* Marcel Dekker, New York, 1984.

[7] Q. Yang, The relaxed CQ algorithm for solving the split feasibility problem, *Inverse Probl.*, 20, 2004, pp.1261-1266.

[8] G. López, V. Martín-Márquez, F. Wang, H. K. Xu, Solving the split feasibility problem without prior knowledge of matrix norms, *Inverse Probl.* 27, 2012, 085004.

[9] J. Zhao, J. Zhang Q. Yang, A simple projection method for solving the multiple-sets split feasibility problem, *Inverse Probl. Sci. Eng.*, 21(3), 2013, pp.537-546.

[10] Q. L. Dong, Y. Yao, S. He, Weak convergence theorems of the modified relaxed projection algorithms for the split feasibility problem in Hilbert spaces, *Optim. Lett.*, accepted, 2013.

[11] S. He, W. Zhu, A note on approximating curve with 1-norm regularization method for the split feasibility problem, *J. Appl. Math.*, Volume 2012, Article ID 683890, 10 pages.

[12] Z. Li, D. Han, W. Zhang, A self-adaptive projection-type method for nonlinear multiple-sets split feasibility problem, *Inverse Probl. Sci. Eng.*, 21, (2013), pp.155-170.

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

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

[15] Q. Yang, The relaxed CQ algorithm for solving the split feasibility problem, *Inverse Probl.*, 20, (2004),pp.1261-1266.

[16] B. Qu, N. Xiu, A note on the CQ algorithm for the split feasibility problem, *Inverse Probl.*, 21, (2005), pp.1655-1665.

[17] A. Moudafi, A relaxed alternating CQ-algorithm for convex feasibility problems, *Nonlinear Anal.*, 79, (2013), pp.117-121.

[18] C. Byrne, A. Moudafi, Extensions of the CQ algorithm for the split feasibility and split equality problems, *hal-00776640-version 1,* 2013.

[19] H. K. Xu, Iterative methods for the split feasibility problem in infinite-dimensional Hilbert spaces, *Inverse Probl.*, vol. 26, 2010, No. 10, Article ID 105018, p. 17.

[20] A. Moudafi, Coupling proximal algorithm and Tikhonov method, *Nonlinear Times Digest.*, vol. 1, 1994, no. 2, pp.203-209.

[21] R. Chen, J. Li, Y. Ren, Regularization method for the approximate split equality problem in infinite-dimensional Hilbert spaces, *Abstr. Appl. Anal.*, Volume 2013, Article ID 813635, 5 pages.

[22] Q. L. Dong, S. N. He, Modified projection algorithms for solving the split equality problems, *The Scientific World Journal*, accepted, 2013.

[23] P. Tseng, A modified forwardCbackward splitting method for maximal monotone mappings, *SIAM J. Control Optim.*, 38, (2000), pp.431-446.

[24] J. Zhao, Q. Yang, Several acceleration schemes for solving the multiple-sets split feasibility problem, *Linear Algebra Appl.*, 437, (2012), pp.1648-1657.

[25] Q. L. Dong, S. N. He, Solving the split equality problem without prior knowledge of operator norms, *Optimization*, revised, 2013.

[26] A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, *SIAM J. Imaging Sciences*, 2, 2009, pp.183-202.

[27] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed., Athena Scientific, Belmont, MA. 1999.

[28] J. M. Ortega, W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Classics Appl. Math. , 30, SIAM, Philadelphia, 2000.

[29] P. L. Combettes, J. C. Pesquet, Proximal splitting methods in signal processing, in: *Fixed-Point Algorithms for Inverse Problems in Science and Engineering,* (H. H. Bauschke, R. S. Burachik, P. L. Combettes, V. Elser, D. R. Luke, and H. Wolkowicz, Editors), pp. 185-212. Springer, New York, 2011.