**Noname manuscript No.**
(will be inserted by the editor)

# A new splitting method for monotone inclusions of three operators

**Yunda Dong · Xiaohuan Yu**

Received: date / Accepted: date

**Abstract** In this article, we consider monotone inclusions in real Hilbert spaces and suggest a new splitting method. The associated monotone inclusions consist of the sum of one bounded linear monotone operator and one inverse strongly monotone operator and one maximal monotone operator. The new method, at each iteration, first implements one forward-backward step as usual and next implements a descent step, and it can be viewed as a variant of a proximal-descent algorithm in a sense. Its most important feature is that, at each iteration, it needs evaluating the inverse strongly monotone part once only in the forward-backward step and, in contrast, the original proximal-descent algorithm needs evaluating this part twice both in the forward-backward step and in the descent step. Moreover,

✉ Yunda Dong
School of Mathematics and Statistics, Zhengzhou University, Zhengzhou, 450001, PR China
Tel.: +86-0371-67780033
Fax: +86-0371-67780033
E-mail: ydong@zzu.edu.cn

Xiaohuan Yu
School of Mathematics and Statistics, Zhengzhou University, Zhengzhou, 450001, PR China

unlike a recent work, we no longer require the adjoint operation of this bounded linear monotone operator in the descent step. Under standard assumptions, we analyze weak and strong convergence properties of this new method. Rudimentary experiments indicate the superiority of our suggested method over several recently-proposed ones for our test problems.

**Keywords** Monotone inclusions · Self-adjoint operator · inverse strongly monotone · Splitting method · Weak convergence

**Mathematics Subject Classification (2000)** 58E35 · 65K15

## 1 Introduction

Let $\mathcal{H}$ be a real infinite-dimensional Hilbert space with usual inner product $\langle x,\ y\rangle$ and induced norm $\|x\| = \sqrt{\langle x,\ x\rangle}$ for $x, y \in \mathcal{H}$. We consider the following monotone inclusion of finding an $x \in \mathcal{H}$ such that

$$0 \in F(x) + B(x), \quad \text{with} \quad F := L + C, \tag{1}$$

where $L : \mathcal{H} \to \mathcal{H}$ is nonzero, bounded, linear and monotone, and $C : \mathcal{H} \to \mathcal{H}$ is inverse strongly monotone, and $B : \mathcal{H} \rightrightarrows \mathcal{H}$ is (possibly multi-valued) maximal monotone. For an instance, if $F$ is linear, we may think of $L, C$ as

$$L := 0.5(F - F^*), \quad C := 0.5(F + F^*), \tag{2}$$

respectively, where $F^*$ stands for adjoint operator of $F$. The problem (1) covers convex quadratic programs, linear monotone complementarity problems, and linear monotone variational inequalities and so on.

A first iterative scheme for solving the monotone inclusion above is the forward-backward splitting method [1, 2], whose recursive formula is

$$(I + \alpha_k B)(x^{k+1}) \ni x^k - \alpha_k F(x^k),$$

where $\alpha_k > 0$ is called a steplength. For any chosen starting point $x^0 \in \mathcal{H}$, it is strongly convergent provided that the forward operator is further $\kappa$-Lipschitz continuous and $\mu$-strongly monotone, and $\alpha_k \equiv \alpha \in (0, 2\mu/\kappa^2)$. See [3] for further discussions.

A second iterative scheme is the Peaceman/Douglas-Rachford splitting method of Lions and Mercier [1], whose recursive formulae can be written as

$$\begin{aligned} &(I + \alpha B)(y^k) \ni x^k - \alpha F(x^k), \\ &(I + \alpha F)(x^{k+1}) \ni x^k + \alpha F(x^k) - \gamma_k(x^k - y^k), \end{aligned}$$

where $\alpha > 0$, $\gamma_k \in [0, 2]$ and the series $\sum \gamma_k(2 - \gamma_k)$ diverges. We refer to [4–8] for further discussions.

A third iterative scheme is the Tseng's splitting method [9]. Let $\mathcal{X}$ be some closed convex set intersecting the solution set of the problem (1), choose the s-tarting point $x^0 \in \mathcal{X}$. The recursive formulae of the Tseng's splitting method are

$$\begin{aligned} &(I + \alpha_k B)(y^k) \ni x^k - \alpha_k F(x^k), \\ &x^{k+1} = P_{\mathcal{X}}[y^k - \alpha_k F(y^k) + \alpha_k F(x^k)], \end{aligned}$$

where $\alpha_k > 0$ and $P_{\mathcal{X}}$ is usual projection. Under suitable assumptions, Tseng proved its weak convergence without requiring strong monotonicity of $F$. Yet, the original way of choosing $\alpha_k$ typically results in small steplength phenomenon and turns out to be less efficient as pointed out in [10], in which a relaxed version

$$(I + \alpha_k B)(y^k) \ni x^k - \alpha_k F(x^k), \tag{3}$$

$$x^{k+1} = P_{\mathcal{X}}[x^k - \gamma_k(x^k - y^k - \alpha_k F(x^k) + \alpha_k F(y^k))], \tag{4}$$

where $\gamma_k > 0$, was suggested. In theory, it is an extension of projection type method from monotone variational inequalities [11–13] to monotone inclusions and can be interpreted as a special case of proximal-descent algorithm [14], which is a more flexible variant of the HPE method [15]. Recently, this relaxed version was efficiently implemented for some test problems [16]. Notice that, almost in the same time, [17] independently suggested a conceptual method similar to (3) and (4) but without our self-adaptive choice of $\alpha_k$, and proved the method's convergence in the finite-dimensional space.

A fourth iterative scheme, in the case of $F$ being linear, is a splitting method suggested in [10, 18], which is an extension of projection type methods [19, 12] for monotone variational inequalities. For any given starting point $x^0 \in \mathcal{H}$, its recursive formulae are given by

$$(I + B)(y^k) \ni (I - F)(x^k), \tag{5}$$

$$x^{k+1} = x^k - \gamma_k (I + F^*)(x^k - y^k), \tag{6}$$

where $\gamma_k := \theta\|x^k - y^k\|^2 / \|(I + F^*)(x^k - y^k)\|^2$ and $\theta \in (0, 2)$. For pertinent discussions, we refer to [20–22].

Obviously, the splitting method described by (5)-(6) admits a generalized form

$$(J + B)(y^k) \ni (J - F)(x^k), \tag{7}$$

$$x^{k+1} = x^k - \gamma_k S^{-1}(J + F^*)(x^k - y^k), \tag{8}$$

where $J, S$ are bounded, linear and strongly monotone, and $\gamma_k$ is given by

$$\gamma_k := \theta\|x^k - y^k\|_{J^+}^2 / \|(J + F^*)(x^k - y^k)\|_{S^{-1}}^2,$$

$\theta \in (0, 2)$, and $J^+ := 0.5(J + J^*)$.

Very recently, [23] suggested replacing (8) by

$$x^{k+1} = x^k - \hat{\gamma}_k S^{-1}(J + L^*)(x^k - y^k), \tag{9}$$

where $\hat{\gamma}_k > 0$, and analyzed weak convergence of the resulting method. This method has two remarkable features. (i) One needs evaluating the inverse strongly monotone part only once at each iteration. (ii) When used for solving the convex minimization problem (31) below, it can recover some known methods once $J$ and $S$ are properly chosen, where $J$ can be not self-adjoint and $S$ is self-adjoint.

Inspired by this, we instead suggest replacing (8) by

$$x^{k+1} = P_{\mathcal{X}}^S[x^k - \gamma_k S^{-1}(J - L)(x^k - y^k)], \tag{10}$$

where $\gamma_k > 0$ and $P_{\mathcal{X}}^S$ denotes projector onto the set $\mathcal{X}$ in the sense of (27) below, and $S$ is further self-adjoint.

When we were writing this article, a report [24] appeared. If specialized to the problem (1) above, then the main method stated in [24, Theorem 2] is described by (7) and

$$x^{k+1} = P_{\mathcal{X}}^{J^+}[x^k - \tilde{\gamma}_k (J^+)^{-1}(J^- - L)(x^k - y^k)], \tag{11}$$

where $\tilde{\gamma}_k > 0$ and $J^- := 0.5(J - J^*)$ stands for the skew operation. Its weak convergence was analyzed via the fixed-point approach.

Clearly, both (10) and (11) share the feature of (9) evaluating the inverse strongly monotone part only once at each iteration. On the other hand, we shall see that, even if $S := I$, the involved three directions

$$d^k = -(J - L)(x^k - y^k),\ \hat{d}^k = -(J + L^*)(x^k - y^k),\ \tilde{d}^k = -(J^- - L)(x^k - y^k)$$

are different in general because $d^k$ needs neither the adjoint operation of $L$ as in $\hat{d}^k$ nor the skew operation of $J$ as in $\tilde{d}^k$. The only exception is that the former directions can coincide when $L$ is skew and this occurs when used for solving the aforementioned convex minimization problem (31). Of course, as discussed below, the resulting parameters $\gamma_k$, $\hat{\gamma}_k$, $\tilde{\gamma}_k$ are also different.

The rest of this paper is organized as follows. In Sect. 2, we specify the notation used and review some useful results. In Sect. 3, we state in details our suggested method. In Sect. 4, we analyze weak and strong convergence of our suggested method under suitable assumptions. In Sect. 5, we compare our suggested method with some existing ones. In Sect. 6, we did numerical experiments to confirm the superiority of our suggested method over several recently-proposed ones [16, 23–26] for our test problems. In Sect. 7, we close this article by some concluding remarks.

## 2 Preliminaries

In this section, we first give some basic definitions and then provide some auxiliary results for later use.

Let $\mathcal{BL}(\mathcal{H})$ be the set of all nonzero, bounded, linear operators in $\mathcal{H}$. If $S \in \mathcal{BL}(\mathcal{H})$ is further self-adjoint and strongly monotone, then we use $\|x\|_S$ to stand for $\sqrt{\langle x, S(x)\rangle}$ for all $x \in \mathcal{H}$. $I$ stands for the identity operator, i.e., $Ix = x$ for all $x \in \mathcal{H}$. $\mathrm{dom}T$ stands for the effective domain of $T$, i.e., $\mathrm{dom}T := \{x \in \mathcal{H} : Tx \neq \emptyset\}$. Throughout this article, for any given $J \in \mathcal{BL}(\mathcal{H})$, one may split it into

$$J = J^+ + J^- \qquad \text{with}\ \ J^+ := 0.5(J + J^*),\ J^- := 0.5(J - J^*), \tag{12}$$

where $J^*$ stands for the adjoint operator of $J$. Notice that such adjoint operator must exist uniquely. For any given $J, J' \in \mathcal{BL}(\mathcal{H})$, we use the notation $J \succeq J'$ ($J \succ J'$) to stand for that $J - J'$ is monotone (strongly monotone). This is the corresponding Löwner partial ordering between two bounded and linear operators.

**Definition 2.1** Let $T : \mathcal{H} \to \mathcal{H}$ be a single-valued operator. If there exists some constant $\kappa > 0$ such that

$$\|T(x) - T(y)\| \le \kappa \|x - y\|, \quad \forall x, y \in \mathcal{H},$$

then $T$ is called Lipschitz continuous.

To concisely give the following definition, we agree on that the notation $(x, w) \in T$ and $x \in \mathcal{H}$, $w \in T(x)$ have the same meaning. Moreover, $w \in Tx$ if and only if $x \in T^{-1}w$, where $T^{-1}$ stands for the inverse of $T$.

**Definition 2.2** Let $T : \mathcal{H} \rightrightarrows \mathcal{H}$ be an operator. It is called monotone if and only if

$$\langle x - x',\ w - w' \rangle \ge 0, \qquad \forall (x, w) \in T,\ \ \forall (x', w') \in T;$$

maximal monotone if and only if it is monotone and for given $\hat{x} \in \mathcal{H}$ and $\hat{w} \in \mathcal{H}$ the following implication relation holds

$$\langle x - \hat{x},\ w - \hat{w} \rangle \ge 0, \quad \forall (x, w) \in T \quad \Rightarrow \quad (\hat{x}, \hat{w}) \in T.$$

**Definition 2.3** Let $T : \mathcal{H} \rightrightarrows \mathcal{H}$ be an operator. $T$ is called uniformly monotone if there exists an increasing function $\phi_T : [0, +\infty) \to [0, +\infty)$ that $\phi_T(t) = 0$ if and only if $t = 0$, and

$$\langle x - x',\ w - w' \rangle \ge \phi_T(\|x - x'\|), \qquad \forall (x, w) \in T,\ \ \forall (x', w') \in T.$$

In the case of $\phi_T(t) = \mu_T t^2$ with $\mu_T > 0$, $T$ is called $\mu_T$-strongly monotone.

**Definition 2.4** Let $f : \mathcal{H} \to (-\infty, +\infty]$ be a closed, proper and convex function. Then for any given $x \in \mathcal{H}$ the sub-differential of $f$ at $x$ is defined by

$$\partial f(x) := \{s \in \mathcal{H} : f(y) - f(x) \geq \langle s, y - x\rangle, \ \forall y \in \mathcal{H}\}.$$

Each $s$ is called a sub-gradient of $f$ at $x$. Moreover, if $f$ is further continuously differentiable, then $\partial f(x) = \{\nabla f(x)\}$, where $\nabla f(x)$ is the gradient of $f$ at $x$.

**Definition 2.5** Let $C : \mathcal{H} \to \mathcal{H}$ be an operator. $C$ is called $c$-inverse strongly monotone if there exists some $c > 0$ such that

$$\langle x - y, C(x) - C(y)\rangle \geq c\|C(x) - C(y)\|^2, \forall x, y \in \mathcal{H}.$$

In particular, if $C(x) = Mx + q$, where $M$ is an $n \times n$ positive semi-definite matrix and $q$ is an $n$-dimensional vector, then

$$\langle x, Cx\rangle \geq \lambda_{\max}^{-1}\|Cx\|^2, \ \ \forall x \in \mathcal{H},$$

where $\lambda_{\max} > 0$ is the largest eigenvalue of $M$.

Notice that Definition 2.5 is an instance of the celebrated Baillon-Haddad theorem (cf. [27, Remark 3.5.2]). Sometimes, we also call that the operator $C$ given in this definition is $c$-cocoercive.

As is well known, the sub-differential of any closed, proper and convex function in an infinite-dimensional Hilbert space is maximal monotone as well. Let $\Omega$ be some nonempty closed convex set in $\mathcal{R}^n$. The associated indicator function defined by

$$\delta_\Omega(x) = \begin{cases} 0 & \text{if } x \in \Omega, \\ \infty & \text{if } x \notin \Omega. \end{cases}$$

is a closed, proper and convex function. Furthermore, for any positive number $\alpha > 0$, we always have $(I + \alpha\partial\delta_\Omega)^{-1} = P_\Omega$, where $P_\Omega$ is usual projection.

Let $\mathcal{X}$ be nonempty, closed and convex subset in $\mathcal{H}$. Let $P_{\mathcal{X}}^{S}[v]$ be projection of $v$ onto the set $\mathcal{X}$ induced by the distance $\|v-u\|_S$, i.e., it is the unique solution to the minimization problem

$$\min\left\{\|v-u\|_S : u \in \mathcal{X}\right\}. \tag{13}$$

Then the relation

$$\left\|P_{\mathcal{X}}^{S}[u]-P_{\mathcal{X}}^{S}[v]\right\|_S \leq \|u-v\|_S$$

holds for all $u, v$ in $\mathcal{H}$. In the case of $S=I$, the projection defined here becomes the usual one. Such a property tells us that the projection operator is non-expansive.

**Lemma 2.1** *Assume that $B : \mathcal{H} \rightrightarrows \mathcal{H}$ is maximal monotone. Let $J, J' \in \mathcal{BL}(\mathcal{H})$. Let $Q \in \mathcal{BL}(\mathcal{H})$, where $Q^*Q = QQ^* = I$. Assume that they satisfy $J = Q^*DQ, J' = Q^*D'Q$, where $D, D'$ are given by*

$$D(x) = (j_1x_1, j_2x_2, ..., j_nx_n, ...), \qquad D'(x) = (j_1'x_1, j_2'x_2, ..., j_n'x_n, ...).$$

*If $+\infty > j_{\max}' \geq j_n' \geq j_n \geq j_{\min} > 0$ for all $n$, then*

$$\|J[x-(J+B)^{-1}(Jx+z)]\| \leq \|J'[x-(J'+B)^{-1}(J'x+z)]\|,$$

*where $z \in \mathcal{H}$. In particular, when $J = \frac{1}{\alpha}I$ and $J' = \frac{1}{\alpha'}I$ and $\alpha \geq \alpha' > 0$, we have*

$$\frac{\|x-(I+\alpha B)^{-1}(x+\alpha z)\|}{\alpha} \leq \frac{\|x-(I+\alpha' B)^{-1}(x+\alpha' z)\|}{\alpha'}.$$

*Proof* We may mimic the proof of [16, Lemma 1] to complete the proof here, and the second conclusion of this lemma has been proven there. □

Notice that, in finite-dimensional case, the assumptions in Lemma 2.1 mean that $J, J'$ are symmetric, positive definite matrices, and $Q$ is orthogonal matrix. This lemma will not be further used. Yet, we include it here for its own interest.

At the end of this section, we review weak convergence of the method [23].

**Proposition 2.1** *If the operator $C$ further satisfies*

$$\langle x-y, C(x)-C(y)\rangle \geq \hat{c}\|C(x)-C(y)\|^2_{(J^+)^{-1}}, \forall x, y \in \mathcal{H}, \quad \text{with } \hat{c} > 1/4 \tag{14}$$

*and the parameter $\hat{\gamma}_k$ in (9) is given by*

$$\hat{\gamma}_k := \hat{\theta}\|x^k - y^k\|^2_{J^+}/\|(J+L^*)(x^k - y^k)\|^2_{S^{-1}}, \quad 0 < \hat{\theta} < 2 - 1/(2\hat{c}), \tag{15}$$

*then the method described by (7) and (9) is weakly convergent.*

## 3 Method

In this section, we describe step by step our suggested method just mentioned above. Furthermore, to make a comparison, we also give counterpart of the method described by (3) and (4) for monotone inclusions.

Below we state our suggested method in details.

**Algorithm 3.1**

**Step 0**. Choose $J_0$, $\mathcal{X}_0 = \mathcal{H}$, $S \in \mathcal{BL}(\mathcal{H})$, $\theta \in (0, 2)$. Find $c$. Choose $x^0 \in \mathcal{H}$. Set $k := 0$.

**Step 1** (Forward-backward step). Choose $J_k \in \mathcal{BL}(\mathcal{H})$ and compute

$$(J_k + B)(y^k) \ni (J_k - F)(x^k). \tag{16}$$

If $y^k = x^k$, then stop. Otherwise, calculate $D_k := J_k^+ - L^+ - \frac{1}{4c} I$ and

$$\gamma_k = \theta\langle x^k - y^k, D_k(x^k - y^k)\rangle/\|(J_k - L)(x^k - y^k)\|^2_{S^{-1}} \tag{17}$$

and go to Step 2.

**Step 2** (Descent step). Find some closed convex set $\mathcal{X}_k$ containing the solution set. Compute

$$x^{k+1} = P^S_{\mathcal{X}_k}[x^k - \gamma_k S^{-1}(J_k - L)(x^k - y^k)]. \tag{18}$$

Set $k := k+1$, and go to Step 1.

Note that, in the case $\mathcal{X}_k \equiv \mathcal{H}$ being further finite-dimensional and $J_k \equiv J$, the descent step (18) reduces to

$$x^{k+1} = x^k - \gamma_k S^{-1}(J - L)(x^k - y^k),$$

where the involved $S$ and $J$ are positive definite matrices and $S$ is further symmetric. The idea of using them to replace the identity matrix can be found in [12, Algorithm 2.2]. In such case, if we further assume $L^* = -L$, then the descent direction here, given by $-S^{-1}(J - L)(x^k - y^k)$, just coincides with the one (9).

*Remark 3.1* (i) It shall be specially stressed that $\mathcal{X}_k$ containing the solution set can be well chosen self-adaptively. As implied by Lemma 4.1 below, we may make use of the following inequality

$$\langle x^k - x^*, (J_k - L)(x^k - y^k)\rangle \geq \langle x^k - y^k, D_k(x^k - y^k)\rangle$$

(which holds for each solution point) to define the closed half-space

$$\mathcal{X}_k := \{x \in \mathcal{H} : \langle x^k - x, (J_k - L)(x^k - y^k)\rangle - \langle x^k - y^k, D_k(x^k - y^k)\rangle \geq 0\}. \quad (19)$$

Obviously, $x^k$ does not belong to this set and the latter must include all solution points of the problem under consideration.

(ii) When $L$ satisfies $L = -L^*$ and $J_k := \alpha_k^{-1} I$, then the condition

$$D_k := J_k^+ - L^+ - \frac{1}{4c} I \succ 0 \quad (20)$$

corresponds to $\sup_k \alpha_k < 4c$. When $J = \alpha^{-1} I$, $S = I$, $L = 0$, the upper bound of [23, Algorithm 1] is $4c$ as well.

*Remark 3.2* When the involved $C(x) := Mx + q$ is further linear, we need to estimate the largest eigenvalue of $M$, as mentioned in Definition 2.5. If it is known in advance or easy to calculate, then we will use it directly. Otherwise, an alternative is an upper bound of the largest eigenvalue of the matrix $M$ and is given by

$$\max_{i=1,\ldots,n} \sum_{j=1}^{n} |m_{ij}|. \tag{21}$$

*Remark 3.3* Let's compare the parameter $\gamma_k$ in the Algorithm 3.1 with (15). Clearly, their differences are three-fold. (i) The ingredients in the parameter $\gamma_k$ are $\theta$, $\langle x^k - y^k, D_k(x^k - y^k)\rangle$ and $(J_k - L)(x^k - y^k)$ whereas the ingredients in the parameter $\hat{\gamma}_k$ correspond to $\hat{\theta}$, $\|x^k - y^k\|^2_{J^+}$ and $(J + L^*)(x^k - y^k)$, respectively. (ii) Be aware that the $\theta$ is merely in the interval $(0, 2)$ and has nothing to do with the inverse strongly monotone constant. (iii) In practical implementations, generating $L^T$ from $L$ needs transpose operation on the given matrix $L$, and this will lead to a bit increase of CPU time when the number of non-zero entries in $L$ is large.

*Remark 3.4* For the Algorithm 3.1 above, when $F := L + C$ and $C$ vanishes, the parameter $\gamma_k$ and the descent step are

$$\gamma_k = \theta\langle x^k - y^k, (J_k^+ - F^+)(x^k - y^k)\rangle / \|(J_k - F)(x^k - y^k)\|^2_{S^{-1}} \tag{22}$$

and

$$x^{k+1} = P^S_{\mathcal{X}_k}[x^k - \gamma_k S^{-1}(J_k - F)(x^k - y^k)], \tag{23}$$

respectively. The corresponding method can be viewed as the counterpart of the method described by (3) and (4). If we set $F(x) = Wx + q$, where $W$ is an $n \times n$ positive semi-definite matrix and $q$ is an $n$-dimensional vector, and in the case of $J_k \equiv \alpha^{-1} I$, then we may choose

$$\alpha < 1/\|0.5(W + W^T)\| \tag{24}$$

to guarantee convergence. As to the set $\mathcal{X}_k$, we may adopt the following inequality (see (27) below)

$$\langle y^k - x^*, J_k(x^k - y^k) - F(x^k) + F(y^k)\rangle \geq 0$$

(which holds for each solution point) to define the closed half-space

$$\mathcal{X}_k := \{x \in \mathcal{H} : \langle y^k - x, (J_k - F)(x^k - y^k)\rangle \geq 0\}. \tag{25}$$

Obviously, $x^k$ does not belong to this set and the latter must include all solution points of the problem under consideration.

## 4 Convergence Properties

In this section, we analyze convergence behaviours of the Algorithm 3.1. Under standard assumptions, we prove its weak convergence. If either $L$ in the forward operator or the backward operator $B$ is uniformly monotone, then it must be strongly convergent.

To analyze convergence behaviours of the Algorithm 3.1, we first make the following assumptions.

**Assumption 4.1** Assume that (i) $L \in \mathcal{BL}(\mathcal{H})$ and the forward operator $F := L + C$.

(ii) The operator $C$ is $c$-inverse strongly monotone.

(iii) The backward operator $B : \mathcal{H} \rightrightarrows \mathcal{H}$ is maximal monotone.

(iv) $L, B$ is uniformly monotone.

(v) $S \in \mathcal{BL}(\mathcal{H})$ is self-adjoint and strongly monotone.

To further simplify the proof details of convergence analysis of the Algorithm 3.1, we introduce the following lemma.

**Lemma 4.1** *If Assumption 4.1 holds and* $D_k := J_k^+ - L^+ - \frac{1}{4c}I$*, then the sequences* $\{x^k\}$ *and* $\{y^k\}$ *generated by the Algorithm 3.1 described by (16) satisfy*

$$\langle x^k - x^*, (J_k - L)(x^k - y^k)\rangle \geq \langle x^k - y^k, D_k(x^k - y^k)\rangle + (\phi_L + \phi_B)(\|y^k - x^*\|).$$

*Proof* It follows from (16) that

$$B(y^k) \ni J_k(x^k - y^k) - F(x^k), \tag{26}$$

which, together with $B(x^*) \ni -F(x^*)$ and uniform monotonicity of $B$, implies

$$\begin{aligned}
&\phi_B(\|y^k - x^*\|)\\
&\leq \langle y^k - x^*, J_k(x^k - y^k) - F(x^k) + F(x^*)\rangle\\
&= \langle y^k - x^*, J_k(x^k - y^k) - F(x^k) + F(y^k)\rangle - \langle y^k - x^*, F(y^k) - F(x^*)\rangle \qquad (27)\\
&= \langle y^k - x^*, J_k(x^k - y^k) - L(x^k) + L(y^k)\rangle - \langle y^k - x^*, C(x^k) - C(y^k)\rangle\\
&\quad - \langle y^k - x^*, F(y^k) - F(x^*)\rangle,
\end{aligned}$$

where the last equality follows from $F := L + C$. Thus, we get

$$\begin{aligned}
&\langle x^k - x^*, J_k(x^k - y^k) - L(x^k) + L(y^k)\rangle\\
&\geq \langle x^k - y^k, J_k(x^k - y^k) - L(x^k) + L(y^k)\rangle + \langle y^k - x^*, C(x^k) - C(y^k)\rangle\\
&\quad + \langle y^k - x^*, F(y^k) - F(x^*)\rangle + \phi_B(\|y^k - x^*\|),
\end{aligned}$$

which, together with $F := L + C$ and uniform monotonicity of $L$ indicating

$$\langle y^k - x^*, F(y^k) - F(x^*)\rangle \geq \langle y^k - x^*, C(y^k) - C(x^*)\rangle + \phi_L(\|y^k - x^*\|),$$

implies

$$\begin{aligned}
&\langle x^k - x^*, J_k(x^k - y^k) - L(x^k) + L(y^k)\rangle\\
&\geq \langle x^k - y^k, J_k(x^k - y^k) - L(x^k) + L(y^k)\rangle + \langle y^k - x^*, C(x^k) - C(y^k)\rangle\\
&\quad + \langle y^k - x^*, C(y^k) - C(x^*)\rangle + (\phi_B + \phi_L)(\|y^k - x^*\|).
\end{aligned}$$

Since $J_k, L$ are linear, we further get

$$
\begin{aligned}
&\langle x^k - x^*, (J_k - L)(x^k - y^k)\rangle \\
&\geq \langle x^k - y^k, (J_k - L)(x^k - y^k)\rangle + \langle y^k - x^*, C(x^k) - C(y^k)\rangle + \langle y^k - x^*, C(y^k) - C(x^*)\rangle \\
&\quad + (\phi_B + \phi_L)(\|y^k - x^*\|) \\
&= \langle x^k - y^k, (J_k^+ - L^+)(x^k - y^k)\rangle + \langle y^k - x^*, C(x^k) - C(x^*)\rangle + (\phi_B + \phi_L)(\|y^k - x^*\|)
\end{aligned}
$$

where the notation $J^+$ is given by (12). Combining this with that $C$ is $c$-inverse strongly monotone in Assumption 4.1 yields

$$
\begin{aligned}
&\langle x^k - x^*, (J_k - L)(x^k - y^k)\rangle \\
&\geq \langle x^k - y^k, (J_k^+ - L^+)(x^k - y^k)\rangle + \langle y^k - x^k, C(x^k) - C(x^*)\rangle \\
&\quad + \langle x^k - x^*, C(x^k) - C(x^*)\rangle + (\phi_B + \phi_L)(\|y^k - x^*\|) \\
&\geq \langle x^k - y^k, (J_k^+ - L^+)(x^k - y^k)\rangle - \frac{1}{2}\left(2c\|C(x^k) - C(x^*)\|^2 + \frac{1}{2c}\|x^k - y^k\|^2\right) \\
&\quad + c\|C(x^k) - C(x^*)\|^2 + (\phi_B + \phi_L)(\|y^k - x^*\|) \\
&= \langle x^k - y^k, (J_k^+ - L^+)(x^k - y^k)\rangle - \frac{1}{4c}\|x^k - y^k\|^2 + (\phi_B + \phi_L)(\|y^k - x^*\|) \\
&= \langle x^k - y^k, D_k(x^k - y^k)\rangle + (\phi_B + \phi_L)(\|y^k - x^*\|).
\end{aligned}
$$

The proof is complete. □

**Theorem 4.1** *If Assumption 4.1 holds and there exists some $\rho > 0$ such that*

$$
J_k^+ \succeq L^+ + \frac{1}{4c}I + \rho I, \qquad \sup_k \|J_k\| < +\infty, \quad k = 0, 1, ..., \tag{28}
$$

*then (i) the sequence $\{x^k\}$ generated by the Algorithm 3.1 described by (16)-(18) must converge weakly to an element of the solution set (if nonempty) of the problem (1); (ii) the sequence $\{x^k\}$ is strongly convergent if either $L$ or $B$ is uniformly monotone.*

*Proof* Let $x^*$ be a solution of the problem (1) above. It follows from non-expansive property of the projection operator and (18) that

$$\|x^{k+1} - x^*\|_S^2 = \|P_{\mathcal{X}_k}^S[x^k - \gamma_k S^{-1}(J_k - L)(x^k - y^k)] - P_{\mathcal{X}_k}^S[x^*]\|_S^2$$
$$\leq \|x^k - \gamma_k S^{-1}(J_k - L)(x^k - y^k) - x^*\|_S^2.$$

Thus, we can further get

$$\begin{aligned}
&\|x^{k+1} - x^*\|_S^2 \\
&\leq \|x^k - \gamma_k S^{-1}(J_k - L)(x^k - y^k) - x^*\|_S^2 \\
&= \|x^k - x^*\|_S^2 - 2\gamma_k \langle x^k - x^*, (J_k - L)(x^k - y^k)\rangle + \gamma_k^2 \|(J_k - L)(x^k - y^k)\|_{S^{-1}}^2.
\end{aligned}$$

Combining this with Lemma 4.1 and (28) yields

$$\begin{aligned}
\|x^{k+1} - x^*\|_S^2 &\leq \|x^k - x^*\|_S^2 - 2\gamma_k \langle x^k - y^k, D_k(x^k - y^k)\rangle + \gamma_k^2 \|(J_k - L)(x^k - y^k)\|_{S^{-1}}^2 \\
&\quad - 2\gamma_k(\phi_L + \phi_B)(\|y^k - x^*\|) \\
&\leq \|x^k - x^*\|_S^2 - 2\gamma_k \langle x^k - y^k, D_k(x^k - y^k)\rangle + \gamma_k^2 \|(J_k - L)(x^k - y^k)\|_{S^{-1}}^2 \\
&= \|x^k - x^*\|_S^2 - (2-\theta)\gamma_k \langle x^k - y^k, D_k(x^k - y^k)\rangle \\
&\leq \|x^k - x^*\|_S^2 - (2-\theta)\gamma_k \rho \|x^k - y^k\|^2,
\end{aligned}$$

where the last inequality follows from the notation $D_k := J_k^+ - L^+ - \frac{1}{4c} I$ and (28). It can be easily seen that the sequence $\{\gamma_k\}$ has a positive lower bound, say $\gamma_{\min}$. Thus, we eventually get

$$\|x^{k+1} - x^*\|_S^2 \leq \|x^k - x^*\|_S^2 - (2-\theta)\rho\gamma_{\min} \|x^k - y^k\|^2.$$

Therefore, we can conclude that

$$(i) \ \lim_{k \to +\infty} \|x^k - x^*\|_S \ \text{ exists,} \tag{29}$$

$$(ii) \ \|x^k - y^k\| \to 0. \tag{30}$$

In views of (29) and (v) in Assumption 4.1, the sequence $\{x^k\}$ must be bounded in norm. Thus, there exists some subsequence $\{x^{k_j}\}$ such that it converges weakly to $x^\infty$, so does $\{y^{k_j}\}$ according to (30). On the other hand, it follows form (26) that

$$(F+B)(y^k) \ni J_k(x^k - y^k) - F(x^k) + F(y^k).$$

In views of Assumption 4.1, (28) and (30), we have

$$\|J_k(x^k - y^k) - F(x^k) + F(y^k)\| \to 0,$$

while the subsequence $\{y^{k_j}\}$ converges weakly to $x^\infty$. Thus, we know that

$$0 \in (F+B)(x^\infty),$$

i.e., the weak cluster point $x^\infty$ solves the monotone inclusion (1) above. Here, we need maximality of $B+F$, which follows from [28]. The proof of uniqueness of the weak cluster point is standard and can be found in [9, 10, 7, 16].

Finally, let us prove the second conclusion. In views of Lemma 4.1, we have

$$\langle x^k - x^*, (J_k - L)(x^k - y^k)\rangle \geq \langle x^k - y^k, D_k(x^k - y^k)\rangle + (\phi_L + \phi_B)(\|y^k - x^*\|).$$

Meanwhile, Assumption 4.1 and the condition (28) tell us that $\|J_k - L\|$ is bounded. Combining these with (29) and (30) yields

$$\lim_{k\to+\infty}(\phi_L + \phi_B)(\|y^k - x^*\|) = 0.$$

This indicates that if either $L$ or $B$ is uniformly monotone, then the sequence $\{y^k\}$ must converge strongly to an element of the solution set (if nonempty) of the problem (1) above, so does $\{x^k\}$ according to (30). □

*Remark 4.1* (i) In the second conclusion of Theorem 4.1, under proper assumptions, we analyze strong convergence of the Algorithm 3.1. Similar convergence properties also hold for the method described by (3) and (4), although this was not apparently pointed out in the first author's Ph.D. dissertation at that time.

(ii) As to a relaxed version [10, Algorithm 4.2.4] described by (3) and (4) of the Tseng's splitting method, its locally linear rate of convergence can be found in [10, Theorem 4.2.4]. Moreover, under standard assumptions, it is not difficult to mimic the corresponding proof techniques to confirm locally linear rate of convergence of the Algorithm 3.1 suggested in this article.

## 5 Relations to other methods

In this section, we discuss relations of the above-mentioned Algorithm 3.1 to other existing ones.

### 5.1 Case 1

When $F$ vanishes, $\mathcal{X}_k \equiv \mathcal{H}$, $S = I$, the corresponding version of the Algorithm 3.1 is closely related to [29].

### 5.2 Case 2

In the case that $C$ inside $F$ vanishes, $J_k \equiv \alpha_k^{-1} I$, $S = I$, the Algorithm 3.1 reduces to an instance of [10, Algorithm 4.2.4] or [16, Algorithm 1].

### 5.3 Case 3

In the case that $C$ inside $F$ vanishes, $J_k \equiv \alpha_k^{-1} I$, $S = I$ and $B$ is taken to be the differential of the indicator function of some nonempty, closed and convex subset in finite-dimensional spaces, the Algorithm 3.1 reduces to projection type method proposed in [11–13].

### 5.4 Case 4

An application of the Algorithm 3.1 is to solve the following convex minimization

$$\min \quad f(x) + g(Ex) + h(x) \tag{31}$$

where $f, h : \mathcal{H} \to \mathcal{R}$ are proper, closed and convex functions, and the gradient of $h$ is $\kappa$-Lipschitz continuous, $g : \mathcal{G} \to \mathcal{R}$ is a proper, closed and convex function, and $E : \mathcal{H} \to \mathcal{G}$ is a nonzero, bounded and linear operator. Under mild assumptions, its optimality conditions are

$$0 \in L(x, u) + C(x, u) + B(x, u),$$

where

$$L(x, u) := (E^* u, -Ex),\ C(x, u) := (\nabla h(x), 0),\ B(x, u) := (\partial f(x), \partial g^{-1}(u)).$$

Thus, if we further follow [23, Sect. 5] to set

$$J := \begin{pmatrix} \frac{1}{\tau} I & 0 \\ -\eta E & \frac{1}{\sigma} I \end{pmatrix}, \quad S^{-1} := \mu S_1^{-1} + (1-\mu) S_2^{-1},$$

where $\tau > 0$, $\sigma > 0$, $\mu \in [0, 1]$, $S_1$ and $S_2$ are defined by

$$S_1(x, u) := (\frac{1}{\tau} x + (1-\eta) E^* u, (1-\eta) Ex + \frac{1}{\sigma} u + \tau(1-\eta)(2-\eta) EE^* u),$$
$$S_2(x, u) := (\frac{1}{\tau} x + \sigma(2-\eta) E^* Ex - E^* u, -Ex + \frac{1}{\sigma} u),$$

the main iterative formulae of the Algorithm 3.1 become

$$\bar{x}^k = (I + \tau\partial f)^{-1}(x^k - \tau E^* u^k - \tau\nabla h(x^k)),$$
$$\bar{u}^k = (I + \sigma\partial g^{-1})^{-1}(u^k + \sigma E((1-\eta)x^k + \eta\bar{x}^k)),$$

and

$$x^{k+1} = x^k + \gamma_k(\tilde{x}^k - \mu\tau(2-\eta)E^*\tilde{u}^k),$$
$$u^{k+1} = u^k + \gamma_k(\sigma(1-\mu)(2-\eta)E\tilde{x}^k + \tilde{u}^k),$$

where $\tilde{x}^k = \bar{x}^k - x^k$, $\tilde{u}^k = \bar{u}^k - u^k$ and $\gamma_k$ is given by

$$\gamma_k = \theta\frac{(\frac{1}{\tau} - \frac{\kappa}{4})\|\tilde{x}^k\|^2 + (\frac{1}{\sigma} - \frac{\kappa}{4})\|\tilde{u}^k\|^2 - \eta\langle\tilde{x}^k, E^*\tilde{u}^k\rangle}{V(\tilde{x}^k, \tilde{u}^k)},$$

where $0 < \theta < 2$ and

$$V(\tilde{x}^k, \tilde{u}^k) = \frac{1}{\tau}\|\tilde{x}^k\|^2 + \frac{1}{\sigma}\|\tilde{u}^k\|^2 + (1-\mu)\sigma(1-\eta)(2-\eta)\|E\tilde{x}^k\|^2$$
$$+\mu\tau(2-\eta)\|E^*\tilde{u}^k\|^2 + 2(\mu\eta - 2\mu - \eta + 1)\langle\tilde{x}^k, E^*\tilde{u}^k\rangle.$$

For this special case of the Algorithm 3.1, we shall clarify how to give conditions on $D := J^+ - L^+ - \frac{1}{4c}I$. In this case, $L^+ = 0$ and $C$ is of the same inverse strongly monotone constant as $\nabla h$. In view of the Baillon-Haddad theorem, the latter's inverse strongly monotone constant is $1/\kappa$. Thus, $c = 1/\kappa$ and, for any given $z := (x, u)$, we have

$$\begin{aligned}\langle z, Dz\rangle &= (\frac{1}{\tau} - \frac{\kappa}{4})\|x\|^2 + (\frac{1}{\sigma} - \frac{\kappa}{4})\|u\|^2 - \eta\langle Ex, u\rangle \\ &\geq (\frac{1}{\tau} - \frac{\kappa}{4} - \frac{\eta\|E\|^2}{2\epsilon})\|x\|^2 + (\frac{1}{\sigma} - \frac{\kappa}{4} - \frac{\eta\epsilon}{2})\|u\|^2,\end{aligned}$$

for all $\epsilon > 0$. Selecting

$$\epsilon := \sqrt{\|E\|^2 + \frac{1}{\eta^2}(\frac{1}{\tau} - \frac{1}{\sigma})^2} - \frac{1}{\eta}(\frac{1}{\tau} - \frac{1}{\sigma})$$

yields $\langle z, Dz\rangle \geq \mu_D \|z\|^2$, where

$$\mu_D := \frac{1}{2}(\frac{1}{\tau} + \frac{1}{\sigma}) - \frac{\kappa}{4} - \frac{1}{2}\sqrt{\eta^2\|E\|^2 + (\frac{1}{\tau} - \frac{1}{\sigma})^2}.$$

When $\kappa$ vanishes, it becomes

$$\mu_{J^+} := \frac{1}{2}(\frac{1}{\tau} + \frac{1}{\sigma}) - \frac{1}{2}\sqrt{\eta^2\|E\|^2 + (\frac{1}{\tau} - \frac{1}{\sigma})^2}. \tag{32}$$

To guarantee $\mu_D > 0$, we require the following conditions

$$\tau < 4/\kappa,\ \sigma < 4/\kappa, \quad \mu_{J^+} > \frac{\kappa}{4}. \tag{33}$$

If $h$ vanishes, then these conditions reduce to

$$\frac{1}{\tau} > \frac{\sigma}{4}\eta^2\|E\|^2. \tag{34}$$

Meanwhile, it is easy to check that the $\hat{c}$ in (14) is $\hat{c} := \mu_{J^+}/\kappa$. So, the condition $\hat{c} > 1/4$ corresponds to $\mu_{J^+} > \kappa/4$.

If we set $\eta = 2$, the method above coincides with [26, 25] used for solving the convex minimization under the consideration. The assumption on $\tau$ and $\sigma$ there is

$$\min\{\frac{1}{\tau}, \frac{1}{\sigma}\}\left(1 - \sqrt{\tau\sigma}\|E\|\right) > \frac{\kappa}{2}. \tag{35}$$

Obviously, when $h$ vanishes, it is the same as (34). As remarked in [23, Remark 5.2], our assumption $\mu_{J^+} > \frac{\kappa}{4}$ in (33) on $\tau$ and $\sigma$ is less conservative.

If we further set $\eta = 2$ and $h \equiv 0$, the method above coincides with [30] used for solving the convex minimization under the consideration, and the involved parameters $\tau$ and $\sigma$ satisfy (34).

For the convex minimization (31) above, we always have $-L = L^*$. This indicates that the direction $d^k$ in the Algorithm 3.1 coincides with the one $\hat{d}^k$ [23]. So, the Algorithm 3.1 can recover all known methods that can be viewed as special cases of [23]. We refer to [23, Sect. 5] for more details and will not list them one by one.

5.5 Case 5

The problem considered in the report [24] is to find an $x \in \mathcal{H}$ such that

$$0 \in F(x) + B(x), \quad \text{with} \quad F := N + C,$$

where $N : \mathcal{H} \to \mathcal{H}$ is monotone and $\kappa$-Lipschitz continuous, $C : \mathcal{H} \to \mathcal{H}$ is $c$-inverse strongly monotone, and $B : \mathcal{H} \rightrightarrows \mathcal{H}$ is maximal monotone. The main method there is stated in [24, Theorem 2] and, in our notation, reads

$$(J + B)(y^k) \ni (J - F)(x^k)$$
$$x^{k+1} = P_{\mathcal{X}}^{J^+}[x^k - \tilde{\gamma}_k (J^+)^{-1}(J^-(x^k - y^k) - N(x^k) + N(y^k))].$$

In the case of $N := L$ being further linear, the method above becomes

$$(J + B)(y^k) \ni (J - F)(x^k) \tag{36}$$
$$x^{k+1} = P_{\mathcal{X}}^{J^+}[x^k - \tilde{\gamma}_k (J^+)^{-1}(J^- - L)(x^k - y^k)]. \tag{37}$$

For weak convergence, one of the assumptions made in [24, Theorem 2] is

$$\|N - J^-\|^2 < \mu_{J^+}\left(\mu_{J^+} - \frac{1}{2c}\right).$$

where $\|N - J^-\|$ stands for Lipschitz constant of $N - J^-$. In the $N := L$ case, this assumption becomes

$$\|L - J^-\|^2 < \mu_{J^+}\left(\mu_{J^+} - \frac{1}{2c}\right) \quad \Rightarrow \quad \mu_{J^+} > \frac{1}{4c} + \sqrt{\|L - J^-\|^2 + \frac{1}{16c^2}}.$$

When $J$ is further self-adjoint, it implies

$$\mu_{J^+} > \frac{1}{4c} + \sqrt{\|L\|^2 + \frac{1}{16c^2}}, \tag{38}$$

where we have assumed that $N := L$ is $\kappa$-Lipschitz continuous and $\kappa = \|L\|$. In contrast, in the case of $J_k \equiv J$, our suggested condition (28) is

$$J^+ \succ L^+ + \frac{1}{4c} I.$$

When we have a look at the optimality condition of the convex minimization problem (31) above, in which $L$ is skew and thus $L^+ = 0$, we will find that our assumptions on $J$ can be much weaker.

Based on these observations, as far as the monotone inclusion (1) above is concerned, our suggested the Algorithm 3.1 is widely different from the method stated in [24, Theorem 2] in general.

## 6 Rudimentary experiments

In this section, our primary goal is to confirm that the Algorithm 3.1 outperformed the method [23] for our two test problems. Furthermore, we showed the numerical superiority of the Algorithm 3.1 over several recently-proposed ones for the first test problem.

All numerical experiments were run in MATLAB R2014a (8.3.0.532) with 64-bit (win64) on a desktop computer with an Intel(R) Core(TM) i5-7400 CPU 3.00 GHz and 8.00 GB of RAM. The operating system is Windows 10.

Our first test problem is to solve the following linear monotone complementarity problem [16]

$$0 \in F(x) + \partial\delta_\Omega(x), \quad \text{with} \quad F := L + C, \tag{39}$$

where $\delta_\Omega$ is the indicator function of the first orthant $\Omega = \{x : x_i \geq 0, i = 1, \cdots, n\}$ and $C(x) := (1-t)Mx + q$, where $t \in [0, 1)$, and the associated matrices are $n \times n$ block tridiagonal

$$M = \begin{pmatrix} \bar{Q} & -I & & & & \\ -I & \bar{Q} & -I & & & \\ & \cdot & \cdot & \cdot & & \\ & & \cdot & \cdot & \cdot & \\ & & & -I & \bar{Q} & -I \\ & & & & -I & \bar{Q} \end{pmatrix}, \quad L = tM + \frac{h\bar{c}}{2} \begin{pmatrix} O & I & & & & \\ -I & O & I & & & \\ & \cdot & \cdot & \cdot & & \\ & & \cdot & \cdot & \cdot & \\ & & & -I & O & I \\ & & & & -I & O \end{pmatrix},$$

and $n = m^2$, where $h = 1/(m+1)$, $\bar{c} = 100$, $I$ is an $m \times m$ identity matrix and $\bar{Q}$ is an $m \times m$ matrix of the form $4I + Q + Q^T$ with $Q_{ij} = -1$ whenever $j = i+1,\ i = 1, \cdots, m-1$, otherwise $Q_{ij} = 0$. To know the solution of this problem in advance, we set $q = -((1-t)M + L)e_1$, where $e_1$ is the first column of the corresponding identity matrix. Thus, $x^* = e_1$ is the unique solution of this complementarity problem. Obviously, the problem corresponds to the monotone inclusion (1) with

$$F(x) = L(x) + C(x), \quad B(x) = \partial \delta_{\Omega}(x).$$

Clearly, it follows from (21) that the largest eigenvalue of $M$ is less than 6. Thus, in practical implementations, we adopted

$$c = 1/6, \quad J_k \equiv \alpha^{-1} I, \quad \theta \in (0, 2), \quad S := I. \tag{40}$$

Set $n = 50^2$ and $x^0 = (1, ..., 1)^T$. Recall that $\|x\|_\infty = \max_{i=1,\cdots,n}\{|x_i|\}$. We made use of the following stopping criterion

$$\|x^k - x^*\|_\infty \le \epsilon \|x^0 - x^*\|_\infty.$$

Next, we focus on how to choose the associated steplength $\alpha$ of the Algorithm 3.1 when $J_k \equiv \alpha^{-1} I$ and $F$ is bounded and linear. According to our accumulated experience in the last ten years, the constant steplength $\alpha$ shall be around $1/\kappa(F)$, where $\kappa(F)$ is given by

$$\kappa(F) := \sup_{x \neq y} \frac{\langle x - y, F(x) - F(y) \rangle}{\|x - y\|^2}.$$

For the complementarity problem (39) above, we can make use of (21) to get $\kappa(F) \approx 6$. Thus, in the case of $\alpha = 1/6$, the requirement (28) is satisfied because

$$\alpha = 1/6, \quad L^+ = 0.5M, \quad c = 1/6.$$

Yet, the steplength in some methods can be required to be less than $1/\kappa(F)$, see (24) for an instance. In such case, we may take $\alpha = 0.75/\kappa(F)$ or something like this. Although there has been no general theory of this issue, our doing is usually useful in splitting methods without introducing dual variables and thus is recommended.

Below, we compared the Algorithm 3.1 with other recently-proposed ones, which are labeled as follows.

NEW0: It corresponds to the method stated in Remark 3.4, with $\mathcal{X}_k$ replaced by $\mathcal{H}$, and it can be viewed as an instance of the Algorithm 3.1 as well. We set $\theta = 1.9$. According to the condition on steplength (24) and the discussions in the fifth paragraph of this section, we took $\alpha = 0.75/6$.

NEW: It corresponds to the Algorithm 3.1 and $\mathcal{X}_k \equiv \mathcal{H}$ and the relations (40) hold and $c = 1/(6(1-t))$, and we took

$$\alpha \in \frac{1}{6}\{0.5, 0.6, 0.7, ..., 1.5\}, \quad \theta \in \{1.5, 1.7, 1.9\}.$$

For better numerical performance, we set $\alpha = 1.4/6$ and $\theta = 1.9$.

LP: It corresponds to the method proposed by Latafat and Patrinos [23]. Since

$$J^+ = \alpha^{-1} I, \quad C(x) := (1-t)Mx + q, \quad 0 \le t < 1$$

in our test problem, the involved condition (14) indicates

$$(1-t)\lambda_i \geq \hat{c}\alpha(1-t)^2\lambda_i^2,\ i=1,...,n, \quad \hat{c} > 1/4,$$

where $\lambda_1 \leq ... \leq \lambda_n$ are the eigenvalues of $M$. Consequently, we set

$$\hat{c} = \frac{1}{\alpha(1-t)\lambda_n}.$$

We took 6 as a rough evaluation of the largest eigenvalue $\lambda_n$ in practice and

$$\alpha \in \frac{1}{6}\{0.5, 0.6, 0.7, ..., 1.5\}, \quad \hat{\theta} \in 2 - 1/(2\hat{c}) - \{0.1, 0.3, 0.5\}.$$

For better numerical performance, we set $\alpha = 0.6/6$ and $\hat{\theta} = 2-1.5*\alpha-0.1 = 1.750$.

BD: It corresponds to the method, described by (36) and (37), proposed by Briceno and Davis in [26, Theorem 2]. In the case of $J = \alpha^{-1}I$, we took $\mu_{J^+} = \alpha^{-1}$,

$$\|L\| = (1-t)6 + \frac{h\bar{c}}{2} \approx 5, \quad c = \frac{1}{6(1-t)}.$$

Thus, we made use of (38) to get $\alpha < 0.1733$ in the case of $t = 0.5$. In practical implementations, we chose

$$\alpha \in \{0.01, 0.05, 0.1, 1/6, 0.17\}, \quad \tilde{\gamma}_k \in \{1, 0.5, 0.01, 0.05, 0.001\}.$$

VC: It corresponds to the method proposed by Vu [25] and Condat [26]. For our test problem (39) above, its main iterative formulae can be stated as follows

$$\tilde{x}^k = (I+\tau C)^{-1}(x^k - \tau L x^k - \tau v^k),$$

$$\tilde{v}^k = (I+\sigma\partial\delta_{\Omega}^{-1})^{-1}(v^k + \sigma(2\tilde{x}^k - x^k)),$$

$$x^{k+1} = \rho\tilde{x}^k + (1-\rho)x^k,$$

$$v^{k+1} = \rho\tilde{v}^k + (1-\rho)v^k,$$

where $\tau$, $\sigma$ and $\rho$ are positive parameters. Notice that $v$ here corresponds the dual variable. In practical implementations, we had to resort to the following Moreau identity

$$(I+\sigma B^{-1})^{-1}(u) \equiv u - \sigma(I+\frac{1}{\sigma}B)^{-1}(\frac{u}{\sigma}), \quad \forall \sigma > 0, \ \forall u \in \mathcal{H}$$

for any given maximal monotone operator $B$. For this method, there are two proximal parameters. One is $\tau$ with respect to $C$ and the other is $\frac{1}{\sigma}$ with respect to $\partial\delta_{\Omega}$ (set $B := \partial\delta_{\Omega}$). How to choose them is critical for better numerical performance. Unfortunately, this issue remains open. Here we suggested a practically useful idea of choosing $\tau$ and $\sigma$ for our test problem in the case of $t = 0.5$, i.e., neither of these two proximal parameters $\tau$ and $\sigma^{-1}$ should be large and small. To this end, we set

$$\tau, \ \sigma^{-1} \in \{0.01, 0.05, 0.1, 0.2, 0.3, ..., 1\}$$

Thus, we got 144 pairs of $(\tau, \sigma)$. According to the conditions in [26, Theorem 3.2], each shall satisfy

$$\frac{1}{\tau} - \sigma \geq \frac{\beta}{2},$$

where $\beta$ is the Lipschitz constant of $L$ and, for our test problem,

$$\beta \approx (1-t)6 + \frac{h\bar{c}}{2}2 \approx 5.$$

So, the filtered 30 pairs $(\tau, \sigma)$ are

$$(0.2, 1), (0.2, 2), \qquad (0.1, j), \ j = 1, ..., 7,$$

$$(0.05, j), \ j = 1, ..., 10, \qquad (0.01, j), \ j = 1, ..., 10, 20.$$

Then, for each given pair, we made use of one condition in [26, Theorem 3.2] to determine the corresponding upper bound $\delta$ of $\rho$ and the computed value of $\delta$ is

$$1.30, 1.10, 1.56, 1.52, 1.50, 1.50, 1.50, 1.30, 1.10, 1.76, 1.75, 1.74, 1.72, 1.71, 1.69,$$

**Table 1** Numerical results for all the algorithms above when $t = 0.5$

| $\epsilon$ | NEW0 | | NEW | | LP | | BD | | VC | | OLD | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $k$ | TIME | $k$ | TIME | $k$ | TIME | $k$ | TIME | $k$ | TIME | $k$ | TIME |
| $10^{-6}$ | 192 | 0.0196 | 140 | 0.010 | 290 | 0.020 | - | - | 166 | 2.863 | 193 | 0.027 |
| $10^{-9}$ | 219 | 0.017 | 166 | 0.012 | 344 | 0.023 | - | - | 184 | 2.886 | 218 | 0.032 |

$$1.67, 1.65, 1.62, 1.60, 1.95, 1.95, 1.94, 1.94, 1.94, 1.94, 1.94, 1.94, 1.94, 1.94, 1.93,$$

respectively. Next, we chose $\rho \in \{0.5\delta, 0.7\delta, 0.9\delta\}$. Finally, we found out that $\tau = 0.2$, $\sigma = 1$ and $\rho = 0.9\delta = 0.9 * 1.30 = 1.17$ are good choices for this test problem. Notice that, since $m = 50$ and thus the number of variables is not large, we ran VC using the eig function in MATLAB to factor and store the involved co-efficient matrix of this system of linear equations before the iteration process.

OLD: It corresponds to [16, Algorithm 1] with the only difference: the parameter $\gamma_k$ there has been updated as

$$\theta \frac{\langle x^k - x^k(\alpha_k), x^k - x^k(\alpha_k) - \alpha_k F(x^k) + \alpha_k F(x^k(\alpha_k))\rangle}{\|x^k - x^k(\alpha_k) - \alpha_k F(x^k) + \alpha_k F(x^k(\alpha_k))\|^2}$$

for a bit better numerical performance (we have already made use of the same notation as [16]). Other parameters used are as follows:

$$\alpha_{-1} = 1, \quad \beta \in \{0.5, 0.7, 0.9\}, \quad \rho = \{0.1, 0.2, 0.3, ..., 0.9\}, \quad \theta \in \{1.5, 1.7, 1.9\}$$

and we found out that $\beta = 0.9$, $\rho = 0.4$ and $\theta = 1.9$ are good choices for this test problem.

We reported the corresponding numerical results in Table 1, where "TIME" means the elapsed time using tic and toc (in seconds) and the notation " $-$ " indicates that the algorithm did not return results within ten seconds.

From Table 1, we can see that NEW outperformed LP clearly. In fact, it is by far the fastest among all these algorithms above. Note that VC has to solve one system of linear equations at each iteration and thus becomes time-consuming.

By the way, for the test problem above, in the $t = 0$ case, the directions

$$d^k = -(J - L)(x^k - y^k),\ \hat{d}^k = -(J + L^T)(x^k - y^k)$$

coincide because $-L = L^T$. So, in order to compare the Algorithm 3.1 with [23], we set $t = 0.5$ to avoid this. If our goal is to solve the test problem above without any intention to compare these two methods, then we shall set $C(x) := Mx + q$ directly and either works well.

Our second test problem is to solve the same type of complementarity problem as (39)

$$0 \in F(x) + \partial\delta_\Omega(x), \quad \text{with} \quad F := L + C,$$

where $\Omega$ is the first orthant, and $L$ is given by

$$L = \begin{pmatrix} 2 & -0.5 & -0.4 & 0 \\ -0.5 & 2 & 0 & -0.3 \\ -0.6 & 0 & 2 & -0.5 \\ 0 & -0.7 & -0.5 & 2 \end{pmatrix},$$

and $C(x) := Mx + q$,

$$M = \begin{pmatrix} 2 & -0.5 & -0.5 & 0 \\ -0.5 & 2 & 0 & -0.5 \\ -0.5 & 0 & 2 & -0.5 \\ 0 & -0.5 & -0.5 & 2 \end{pmatrix}, \quad q = -(M + L)e_1,$$

where $e_1 = (1,0,0,0)^T$. Thus, $x^* = e_1$ is the unique solution of this complementarity problem.

Note that the largest eigenvalue of $M$ is 3 and $\kappa(F) \approx 6$. Thus, in practical implementations, we adopted

$$c = 1/3, \quad J_k \equiv \alpha^{-1} I, \quad \theta \in (0,2), \quad S := I. \tag{41}$$

Set $x^0 = (1,1,1,1)^T$. In addition, we made use of the same type of stopping criterion used in the first test problem.

Below, we labelled the Algorithm 3.1 and the method [23].

NEW: It corresponds to the Algorithm 3.1 and $\mathcal{X}_k \equiv \mathcal{H}$ and the relations (41) hold and $c = 1/3$. In practical implementations, we made use of

$$\alpha \in \frac{1}{6}\{0.5, 0.6, 0.7, ..., 1.5\}, \quad \theta \in \{1.5, 1.7, 1.9\}$$

as trial values and we found out that $\alpha = 1.5/6$ and $\theta = 1.9$ are good choices for this test problem.

LP: It corresponds to the method proposed by Latafat and Patrinos [23]. Since

$$J^+ = \alpha^{-1} I, \quad C(x) := Mx + q,$$

in our second test problem, the involved condition (14) indicates

$$\lambda_i \geq \hat{c}\alpha\lambda_i^2, \ i = 1, ..., 4, \quad \hat{c} > 1/4,$$

where $\lambda_1 \leq ... \leq \lambda_4 = 3$ are the eigenvalues of $M$. So, we have

$$\hat{c} = 1/(\alpha\lambda_4) = 1/(3\alpha).$$

In practical implementations, we took

$$\alpha \in \frac{1}{6}\{0.5, 0.6, 0.7, ..., 1.5\}, \quad \hat{\theta} \in 2 - 1/(2\hat{c}) - \{0.1, 0.3, 0.5\}.$$

**Table 2** Numerical results for NEW and LP

| $\epsilon$ | NEW | | | | | | LP | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\theta = 1.5$ | | $\theta = 1.7$ | | $\theta = 1.9$ | | $\hat{\theta} = 1.2750$ | | $\hat{\theta} = 1.4750$ | | $\hat{\theta} = 1.6750$ | |
| | $k$ | TIME | $k$ | TIME | $k$ | TIME | $k$ | TIME | $k$ | TIME | $k$ | TIME |
| $10^{-6}$ | 17 | 1.79e-04 | 14 | 1.65e-04 | 12 | 1.52e-04 | 33 | 2.14e-04 | 28 | 2.06e-04 | 25 | 1.82e-04 |
| $10^{-9}$ | 26 | 2.58e-04 | 21 | 2.14e-04 | 17 | 1.72e-04 | 50 | 3.25e-04 | 42 | 3.06e-04 | 38 | 2.75e-04 |

As a result, we found out that $\alpha = 0.9/6$, $\hat{\theta} = 2 - 1.5 * \alpha - 0.1 = 1.6750$ are good choices for this test problem.

We reported the corresponding numerical results in Table 2, where "TIME" means the elapsed time using tic and toc (in seconds).

From Table 2, we can see that NEW (with $\theta = 1.9$) outperformed LP (with $\hat{\theta} = 1.6750$) clearly for the second test problem.

## 7 Conclusions

In this article, we have suggested a new splitting method for monotone inclusions of three operators in real Hilbert spaces. The new method, at each iteration, first implements one forward-backward step as usual and next implements a descent step. The method shares an appealing property of [23] evaluating the associated inverse strongly monotone part only once at each iteration. Under standard assumptions, we have analyzed its weak and strong convergence. Furthermore, we have already done numerical experiments to confirm the superiority of our suggested method over several recently-proposed ones for our test problems. Impressively, we have suggested a practically useful idea of choosing different proximal parameters involved in some splitting methods introducing dual variables.

Finally, we would like to point out some issues to be solved. For example, like [23], how to choose $J$ and $S$ in the Algorithm 3.1 so as to speed up it remains open. We conjecture that it heavily depends on individual problem thus developing a general scheme becomes challenging. Another issue is to ask what other known methods except those listed in Sect. 5 can be covered for appropriate choices of the operators $J$ and $S$. These issues deserve further investigation.

**Acknowledgments** We are very grateful to Xixian Bai at Shandong University for his help in numerical experiments.

## References

1. Lions, P.L., Mercier, B.: Splitting algorithms for the sum of two nonlinear operators. SIAM J. Numer. Anal. 16, 964-979 (1979)
2. Passty, G.B.: Ergodic convergence to a zero of the sum of monotone operators in Hilbert space. J. Math. Anal. Appl. 72, 383-390 (1979)
3. Chen, H.G., Rockafellar, R.T.: Convergence rates in forward-backward splitting. SIAM J. Optim. 7, 421-444 (1997)
4. Lawrence, J., Spingarn, J.E.: On fixed points of non-expansive piecewise isometric mappings. P. Lond. Math. Soc. 55, 605-624 (1987)
5. Eckstein, J., Bertsekas, D.P.: On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. Math. Program. 55(3), 293-318 (1992)
6. Combettes, P.L.: Solving monotone inclusions via compositions of nonexpansive averaged operators. Optim. 53, 475-504 (2004)
7. Dong, Y.D., Fischer, A.: A family of operator splitting methods revisited. Nonlinear Anal. 72, 4307-4315 (2010)

8. Dong, Y.D.: Douglas-Rachford splitting method for semi-definite programming. J. Appl. Math. Comput. 51, 569-591 (2016)
9. Tseng, P.: A modified forward-backward splitting method for maximal monotone mappings. SIAM J. Control Optim. 38(2), 431-446 (2000)
10. Dong, Y.D.: Splitting Methods for Monotone Inclusions. PhD dissertation, Nanjing University (2003)
11. Sun, D.F.: A class of iterative methods for solving nonlinear projection equations. J. Optim. Theory Appl. 91(1), 123-140 (1996)
12. Solodov, M.V., Tseng, P.: Modified projection-type methods for monotone variational inequalities. SIAM J. Control Optim. 34, 1814-1830 (1996)
13. He, B.S.: A class of projection and contraction methods for monotone variational inequalities. Appl. Math. Optim. 35, 69-76 (1997)
14. Dong, Y.D.: A variable metric proximal-descent algorithm for monotone operators. J. Appl. Math. Comput. 60, 563-571 (2012)
15. Solodov, M.V., Svaiter, B.F.: A hybrid approximate extragradient-proximal point algorithm using the enlargement of a maximal monotone operator. Set-Valued Anal. 7, 323-345 (1999)
16. Huang, Y.Y., Dong, Y.D.: New properties of forward-backward splitting and a practical proximal-descent algorithm. Appl. Math. Comput. 237, 60-68 (2014)
17. Noor, M.A.: Mixed quasi-variational inequalities. Appl. Math. Comput. 146, 553-578 (2003)
18. Dong, Y.D.: An LS-free splitting method for composite mappings. Appl. Math. Letters. 18(8), 843-848 (2005)
19. He, B.S.: Solving a class of linear projection equations. Numer. Math. 68, 71-80 (1994)
20. Irschara, A., Zach, C., Klopschitz, M., Bischof, H.: Large-scale, dense city reconstruction from user-contributed photos. Comput. Vis. Image Und. 116, 2-15 (2012)
21. Alotaibi, A., Combettes, P.L., Shahzad, N.: Solving coupled composite monotone inclusions by successive Fejér approximations of their Kuhn-Tucker set. SIAM J. Optim. 24(4): 2076-2095 (2014)

22. Combettes, P.L., Eckstein, J.: Asynchronous block-iterative primal-dual decomposition methods for monotone inclusions. Math. Program. 168(1-2): 645-672 (2018)

23. Latafat, P., Patrinos, P.: Asymmetric forward-backward-adjoint splitting for solving monotone inclusions involving three operators. Comput. Optim. Appl. 68(1), 57-93 (2017)

24. Briceno-Arias, L.M., Davis, D.: Forward-Backward-Half Forward algorithm with non self-adjoint linear operators for solving monotone inclusions. https://arxiv.org/abs/1703.03436 (2017)

25. Vũ, B.C.: A splitting algorithm for dual monotone inclusions involving cocoercive operators. Adv. Comput. Math. 38(3), 667-681 (2013)

26. Condat L.: A primal-dual splitting method for convex optimization involving Lipschitzian,proximable and linear composite terms. J. Optim. Theory Appl. 158(2), 460-479 (2013)

27. Zalinescu, C.: Convex Analysis in General Vector Spaces. World Scientific Publishing, River Edge, NJ (2002)

28. Brezis, H.,Crandall, M.G., Pazy, A.: Perturbation of nonlinear maximal monotone sets in Banach space. Comm. Pure Appl. Math. 23, 123-144 (1970)

29. Agarwal, R.P., Verma, R.U.: Inexact A-proximal point algorithm and applications to nonlinear variational inclusion problems. J. Optim. Theory Appl. 144, 431-444 (2010)

30. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. J. Math. Imaging Vis. 40(1), 120-145 (2011)