NUMERICAL FUNCTIONAL ANALYSIS AND OPTIMIZATION
2021, VOL. 42, NO. 14, 1627–1644
https://doi.org/10.1080/01630563.2021.2001749

Convergence of an Inertial Shadow Douglas-Rachford Splitting Algorithm for Monotone Inclusions

Jingjing Fan[a], Xiaolong Qin[b], and Bing Tan[a]

[a]Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu, China; [b]Department of Mathematics, Zhejiang Normal University, Zhejiang, China

ABSTRACT

An inertial shadow Douglas-Rachford splitting algorithm for finding zeros of the sum of monotone operators is proposed in Hilbert spaces. Moreover, a three-operator splitting algorithm for solving a class of monotone inclusion problems is also concerned. The weak convergence of the algorithms is investigated under mild assumptions. Some numerical experiments are implemented to illustrate our main convergence results.

ARTICLE HISTORY
Received 1 January 2020
Revised 5 August 2021
Accepted 6 August 2021

KEYWORDS
Inertial algorithm; Monotone inclusion; Shadow Douglas-Rachford splitting algorithm; Three-operator splitting

1. Introduction and preliminaries

Throughout this paper, H is assume to be a real Hilbert space with inner product $\langle \cdot, \cdot \rangle$ and induced norm $|| \cdot ||$. Let $A : H \rightrightarrows H$ be a set-valued operator on H. $D(A) := \{x \in H : Ax \neq \emptyset\}$ stands for the domain of A and $R(A) := \{Az : z \in D(A)\}$ stands for the range of A. Recall that operator $A : H \rightrightarrows H$ is said to be monotone iff $\langle x' - y', x - y \rangle \geq 0$ for all $x' \in Ax$ and $y' \in Ay$. Recall that operator $A : H \rightrightarrows H$ is maximal iff the graph, $G(A)$, of A is not in the graph of other monotone operators properly. For a monotone operator, one also knows that it is maximal iff $\langle x' - y', x - y \rangle \geq 0$, where $x' \in Ax$ and $y, y' \in H$, implies $y' \in Ay$. For each positive real number r, one can define the resolvent operator, $J_{rA} : R(Id + rA) \to D(A)$ of A, by $J_{rA} := (Id + rA)^{-1}$, where Id is the identity operator on H. In this paper, the zero set of A is denoted by $A^{-1}(0)$, that is, $A^{-1}(0) := \{x \in D(A) : 0 \in Ax\}$.

Consider the following inclusion problem, which consists of

$$\text{finding} \quad x \in H \quad \text{such that} \quad 0 \in (A + B)(x), \tag{1.1}$$

where $A : H \rightrightarrows H$ is a set-valued maximally monotone operator and $B : H \to H$ is a single-valued monotone operator, i.e., $\langle B(x) - B(y), x - y \rangle \geq 0$, $(x, y \in H)$ and L-Lipschitz continuous, i.e., $||B(x) - B(y)|| \leq L||x - y||$,

CONTACT Xiaolong Qin qxlxajh@163.com Department of Mathematics, Zhejiang Normal University, Zhejiang, China.

($L > 0$). The solution set of (1.1) is denoted by $(A+B)^{-1}(0)$. This problem includes, as special cases, convex programming problems, split feasibility problems and minimization problems; see, e.g., [1–5]. Several real world problems from the areas of signal recovery, image processing, network communications, location theory, etc, can be formulated as (1.1); see, for instance, [6–8]. Splitting algorithms are efficient and powerful for dealing with problem (1.1). Among them, forward-backward and Douglas-Rachford splitting methods are two fundamental algorithms for solving monotone inclusion problems; see, e.g., [9–13] and the references therein.

In this paper, we focus our attention on the Douglas-Rachford algorithm [14], which was first formulated for solving linear equations and generalized to monotone inclusions in [15] later. The algorithm is of the form in a real Hilbert space H

$$\text{(DR)} \quad x_{n+1} = J_{rA}(2J_{rB} - Id)x_n + (Id - J_{rB})x_n,$$

where Id is the identity on H. Recently, many authors studied this algorithm and its variants; see, e.g., [16–22]. The Douglas-Rachford method generates a fixed point sequence as follows

$$x_{n+1} = \left(\frac{Id + R_{\lambda A}R_{\lambda B}}{2}\right)x_n, \tag{1.2}$$

where $R_{\lambda B} = 2J_{\lambda B} - Id$ denotes the reflected resolvent of monotone operator λB. The iteration (1.2) can be viewed as a discretization of the continuous time dynamical system

$$\dot{x}(t) = J_{\lambda A}(2J_{\lambda B}(x(t)) - x(t)) - J_{\lambda B}(x(t)), \tag{1.3}$$

where the discretizations $\dot{x}(t) \approx x_{n+1} - x_n$ and $x(t) \approx x_n$ are used. Denote $z(t) = J_{\lambda B}(x(t))$,

$$y(t) = x(t) - z(t) \in \lambda B(z(t)),$$

and

$$\dot{x}(t) = \dot{z}(t) + \dot{y}(t).$$

By using these identities in (1.3), we obtain

$$\begin{cases} \dot{z}(t) + z(t) = J_{\lambda A}(z(t) - y(t)) - \dot{y}(t), \\ y(t) \in \lambda B(z(t)). \end{cases} \tag{1.4}$$

Recently, Csetnek, Malitsky and Tam [23] proposed the following shadow Douglas-Rachford splitting algorithm by considering different discretizations of dynamical system (1.4):

$$\text{(SDR)} \quad x_{n+1} = J_{\lambda A}(x_n - \lambda B(x_n)) - \lambda(B(x_n) - B(x_{n-1})). \tag{1.5}$$

This algorithm naturally arises from a nonstandard discretization of a continuous dynamical system associated with the Douglas-Rachford splitting algorithm (1.2), which converges to a solution of (1.1) weakly whenever $\lambda \in \left(0, \frac{1}{3L}\right)$ with L being the Lipschitz constant of B.

In [24], Moudafi and Oliny introduced the following inertial proximal point algorithm for solving the monotone inclusion problem of the sum of two monotone operators:

$$\begin{cases} w_n = x_n + \alpha_n(x_n - x_{n-1}), \\ x_{n+1} = J_{\lambda A}(w_n - \lambda B(x_n)). \end{cases} \tag{1.6}$$

They obtained a weak convergence theorem provided that $0 < \lambda < 2/L$, where L is the Lipschitz constant of B.

Inspired and motivated by the mentioned results above, we introduce inertial shadow Douglas-Rachford splitting algorithms by incorporating the inertial terms (1.6) in the shadow Douglas-Rachford splitting algorithm (1.5) to solve inclusion problem (1.1) in the framework of Hilbert spaces. We obtain the weak convergence of the algorithms and give some numerical experiments to support our main results.

The rest of this paper is organized as follows. In Section 2, we propose an inertial shadow Douglas-Rachford splitting algorithm for solving inclusion problems and show its convergence. In Section 3, we propose a variant which solves three-operator inclusion problems. Finally, in Section 4, numerical experiments are provided to support our algorithms.

2. Two-operator splitting

In this section, we consider the problem of finding a point $x \in H$ such that

$$0 \in (A + B)(x),$$

where $A : H \to H$ is set-valued maximal monotone, and $B : H \to H$ is single-valued monotone and L-Lipschitz (but not necessarily cocoercive, that is, $\gamma||Bx - By||^2 \leq \langle x - y, Bx - By \rangle,\ \forall x, y \in H$, where γ is a positive real constant). We next give the inertial shadow Douglas-Rachford splitting algorithm

$$(\text{ISDR}) \begin{cases} w_n = x_n + \alpha_n(x_n - x_{n-1}), \\ x_{n+1} = J_{\lambda A}(w_n - \lambda B(x_n)) - \lambda(B(x_n) - B(x_{n-1})), \end{cases} \tag{2.1}$$

where (α_n) is non-decreasing with $\alpha_1 = 0$ and $0 \leq \alpha_n \leq \alpha < 1$ for each $n \geq 1$.

The following lemmas will be useful in the proof of the convergence analysis.

Lemma 2.1. *([23]) Suppose that $A : H \rightrightarrows H$ and $B : H \rightrightarrows H$ are maximally monotone operators. Let $\lambda > 0$. Then the set-valued operator on $H \times H$ defined by*

$$\begin{pmatrix} x \\ y \end{pmatrix} \in \left(\begin{bmatrix} \lambda A \\ (\lambda B)^{-1} \end{bmatrix} + \begin{bmatrix} 0 & Id \\ -Id & 0 \end{bmatrix} \right) \begin{pmatrix} x \\ y \end{pmatrix},$$

is demiclosed. That is, its graph is a sequentially closed set in the weak-strong topology.

Lemma 2.2. *([25]) Let C be a nonempty set and $(x_n) \subset H$ be a sequence in H such that the following two conditions hold:*

a. $\lim_{n\to+\infty} ||x_n - x||$, *for any $x \in C$, exists;*
b. *every sequential weak cluster point of $(x_n)_{n\in\mathbb{N}}$ is in C.*

Then $(x_n)_{n\in\mathbb{N}}$ converges to a point in C weakly.

To analyze our algorithm, we also need the following tool.

Lemma 2.3. *Let $A : H \rightrightarrows H$ be a maximally monotone operator. Let (w_n) be given by (2.1) and let $(y_n) \subset H$ be an arbitrary sequence. Let $x_0, x_{-1} \in H$ and (x_n) be a sequence defined by*

$$x_{n+1} = J_A(w_n - y_n) - (y_n - y_{n-1}), \forall n \in \mathbb{N}. \tag{2.2}$$

Then, for all $x \in H$ and $y \in -A(x)$, one has

$$\begin{aligned} &||x_{n+1} - x||^2 + 2\langle y_n - y, x_{n+1} - x\rangle + ||y_n - y||^2 \\ &\le ||x_n - x||^2 + 2\langle y_{n-1} - y, x_n - x\rangle + ||y_{n-1} - y||^2 - ||x_{n+1} - x_n||^2 \\ &\quad - 3||y_n - y_{n-1}||^2 + 4\langle y_n - y_{n-1}, x_n - x_{n+1}\rangle - 2\alpha_n\langle x_n - x, x - x_{n+1}\rangle \\ &\quad - 2\alpha_n\langle x - x_{n-1}, x - x_{n+1}\rangle + 2\alpha_n\langle x_{n-1} - x_n, y_{n-1} - y_n\rangle \\ &\quad - 2\langle y - y_n, x - x_n\rangle. \end{aligned} \tag{2.3}$$

Proof. By utilizing the definition of the resolvent and (2.2), one has

$$x_{n+1} + (y_n - y_{n-1}) - w_n + y_n \in -A(x_{n+1} + (y_n - y_{n-1})). \tag{2.4}$$

Since $x \in H$, $-y \in A(x)$ and A is monotone, one concludes that

$$\langle x_{n+1} + (y_n - y_{n-1}) - w_n + y_n - y, x - x_{n+1} - (y_n - y_{n-1})\rangle \ge 0,$$

which can be equivalently rewritten as

$$\begin{aligned} 0 \le &\langle x_{n+1} - x_n, x - x_{n+1}\rangle + \langle x_{n+1} - x_n, y_{n-1} - y_n\rangle + \langle y_n - y_{n-1}, x - x_{n+1}\rangle \\ &+ \langle y_n - y_{n-1}, y_{n-1} - y_n\rangle + \alpha_n\langle x_{n-1} - x_n, x - x_{n+1}\rangle + \alpha_n\langle x_{n-1} - x_n, y_{n-1} - y_n\rangle \\ &+ \langle y_n - y, x - x_{n+1}\rangle + \langle y_n - y, y_{n-1} - y_n\rangle. \end{aligned} \tag{2.5}$$

To simplify (2.5), one notes that

$$\begin{aligned} &2\langle x_{n+1} - x_n, x - x_{n+1}\rangle = ||x_n - x||^2 - ||x_{n+1} - x_n||^2 - ||x_{n+1} - x||^2, \\ &2\langle y_n - y_{n-1}, y - y_n\rangle = ||y_{n-1} - y||^2 - ||y_n - y_{n-1}||^2 - ||y_n - y||^2, \\ &\langle y_n - y_{n-1}, x - x_{n+1})\rangle = \langle y_n - y_{n-1}, x_n - x_{n+1}\rangle + \langle y_{n-1} - y, x_n - x\rangle + \langle y - y_n, x_n - x\rangle. \end{aligned}$$

Combing these formulas with (2.5) gives the inequality

$$\begin{aligned}&||x_{n+1}-x||^2+2\langle y_n-y,x_{n+1}-x\rangle+||y_n-y||^2\\ &\le ||x_n-x||^2+2\langle y_{n-1}-y,x_n-x\rangle+||y_{n-1}-y||^2-||x_{n+1}-x_n||^2\\ &\quad -3||y_n-y_{n-1}||^2+4\langle y_n-y_{n-1},x_n-x_{n+1}\rangle-2\alpha_n\langle x_n-x,x-x_{n+1}\rangle\\ &\quad -2\alpha_n\langle x-x_{n-1},x-x_{n+1}\rangle+2\alpha_n\langle x_{n-1}-x_n,y_{n-1}-y_n\rangle\\ &\quad -2\langle y-y_n,x-x_n\rangle.\end{aligned}$$

The proof is complete. □

We are now ready to present our main convergence theorem.

Theorem 2.1. *Let $A:H\rightrightarrows H$ be a set-valued maximally monotone operator and let $B:H\to H$ be a single-valued monotone and L-Lipschitz with $(A+B)^{-1}(0)\neq\emptyset$. Let $x_0,x_{-1}\in H$, $\varepsilon>0$, and $\lambda\in[\varepsilon,\frac{1-3(\alpha+1)\varepsilon}{3(\alpha+1)L}]$. Then the sequence (x_n) generated by the Algorithm ISDR (2.1) converges to a point in $(A+B)^{-1}(0)$ weakly.*

Proof. Let $x\in(A+B)^{-1}(0)$ and set

$$y=\lambda B(x)\in-\lambda A(x).$$

Since (2.1) is one of the form specified by (2.2), one applies Lemma 2.3 to the monotone operator λA with $y_n=\lambda B(x_n)$ to deduce that (2.3) holds. Since B is monotone, we have $\langle y_n-y,x_n-x\rangle\ge 0$. Hence

$$\begin{aligned}&||x_{n+1}-x||^2+2\langle y_n-y,x_{n+1}-x\rangle+||y_n-y||^2\\ &\le ||x_n-x||^2+2\langle y_{n-1}-y,x_n-x\rangle+||y_{n-1}-y||^2-||x_{n+1}-x_n||^2\\ &\quad -3||y_n-y_{n-1}||^2+4\langle y_n-y_{n-1},x_n-x_{n+1}\rangle-2\alpha_n\langle x_n-x,x-x_{n+1}\rangle\\ &\quad -2\alpha_n\langle x-x_{n-1},x-x_{n+1}\rangle+2\alpha_n\langle x_{n-1}-x_n,y_{n-1}-y_n\rangle.\end{aligned}\tag{2.6}$$

Next, one estimates the inner product in (2.6). Using the Lipschitzness of B, one arrives at

$$\begin{aligned}&2\langle y_{n-1}-y,x_n-x\rangle\le\lambda L(||x_{n-1}-x||^2+||x_n-x||^2),\\ &2\alpha_n\langle x_{n-1}-x_n,y_{n-1}-y_n\rangle\le\alpha_n\lambda L(||x_{n-1}-x_n||^2+||x_{n-1}-x_n||^2),\\ &2\langle x_{n+1}-x_n,y_{n-1}-y_n\rangle\le\lambda L(||x_{n+1}-x_n||^2+||x_{n-1}-x_n||^2).\end{aligned}\tag{2.7}$$

Using Young's inequality, one can estimate

$$\begin{aligned}2\alpha_n\langle x_n-x,x-x_{n+1}\rangle&\le||x_n-x||^2+\alpha_n^2||x-x_{n+1}||^2,\\ 2\alpha_n\langle x_{n-1}-x,x-x_{n+1}\rangle&\le||x_{n-1}-x||^2+\alpha_n^2||x-x_{n+1}||^2,\\ 2\langle y_n-y_{n-1},x_n-x_{n+1}\rangle&\le 3||y_n-y_{n-1}||^2+\frac{1}{3}||x_{n+1}-x_n||^2.\end{aligned}\tag{2.8}$$

Combining (2.7), (2.8) and $0 \leq \alpha_n \leq \alpha < 1$, one asserts that

$$(1+\lambda L)||x_{n+1}-x||^2 + (\lambda^2L^2+\lambda L+1)||x_n-x||^2 + \left(\frac{2}{3}-\lambda L\right)||x_{n+1}-x_n||^2 \\ \leq (1+\lambda L)||x_n-x||^2 + (\lambda^2L^2+\lambda L+1)||x_{n-1}-x||^2 + (2\alpha+1)\lambda L||x_{n-1}-x_n||^2. \tag{2.9}$$

Since $\lambda \in [\varepsilon, \frac{1-3(\alpha+1)\varepsilon}{3(\alpha+1)L}]$, one concludes that

$$(1+\lambda L)||x_{n+1}-x||^2 + (\lambda^2L^2+\lambda L+1)||x_n-x||^2 + \left(\frac{2\alpha+1}{3(\alpha+1)}+\varepsilon\right)||x_{n+1}-x_n||^2 \\ \leq (1+\lambda L)||x_n-x||^2 + (\lambda^2L^2+\lambda L+1)||x_{n-1}-x||^2 + \frac{2\alpha+1}{3(\alpha+1)}||x_{n-1}-x_n||^2, \tag{2.10}$$

which further yields

$$(1+\lambda L)||x_{n+1}-x||^2 + (\lambda^2L^2+\lambda L+1)||x_n-x||^2 \\ + \frac{2\alpha+1}{3(\alpha+1)}||x_{n+1}-x_n||^2 + \varepsilon\sum_{i=0}^{n}||x_{i+1}-x_i||^2 \\ \leq (1+\lambda L)||x_0-x||^2 + (\lambda^2L^2+\lambda L+1)||x_{-1}-x||^2 + \frac{2\alpha+1}{3(\alpha+1)}||x_{-1}-x_0||^2. \tag{2.11}$$

From this, it follows that (x_n) is bounded and that $||x_{n+1}-x_n|| \to 0$ as n goes to the infinity. Borrowing the Lipschitz continuity of B, one derives that $||y_{n+1}-y_n|| \to 0$ as n goes to the infinity. Setting $z_n = x_n + y_{n-1}$, one also has that $||z_{n+1}-z_n|| \to 0$. Since $z_n = (Id+\lambda B)(x_n) + (y_{n-1}-y_n)$, one finds that

$$x_n = J_{\lambda B}(z_n - (y_{n-1}-y_n)).$$

Since (x_n) is bounded, $||y_{n+1}-y_n|| \to 0$ and $J_{\lambda B}$ is nonexpansive, it then follows that the sequence (z_n) is also bounded. Let $\bar{x}$ and $\bar{z}$ be sequential weak cluster points of bounded sequences (x_n) and (z_n), respectively. Due to (2.11), one sees that the following limit exists

$$\lim_{n\to\infty}\left((1+\lambda L)||x_{n+1}-x||^2 + (\lambda^2L^2+\lambda L+1)||x_n-x||^2 + \frac{2\alpha+1}{3(\alpha+1)}||x_{n+1}-x_n||^2\right). \tag{2.12}$$

Since (x_n) is bounded, $||x_{n+1}-x_n|| \to 0$, it then follows that limit (2.12) is equal to $\lim_{n\to\infty}||x_n-x||^2$. From (2.4), one has

$$-\begin{pmatrix} z_{n+1}-z_n \\ z_{n+1}-z_n \end{pmatrix} \in \left(\begin{bmatrix} \lambda A \\ (\lambda B)^{-1} \end{bmatrix} + \begin{bmatrix} 0 & Id \\ -Id & 0 \end{bmatrix} \right) \begin{pmatrix} z_{n+1}-z_n+x_n \\ z_{n+1}-x_{n+1}-\alpha_n(x_n-x_{n-1}) \end{pmatrix}. \tag{2.13}$$

Using Lemma 2.1, one finds that its graph is demiclosed. Thus, by taking the limit along a subsequence of (x_n) which converges to $\bar{x}$ in (2.13), one deduces that $0 \in (A+B)(\bar{x})$. Since the cluster point $\bar{x}$ of (x_n) was chosen arbitrarily, sequence (x_n) is weakly convergent by Lemma 2.2 and the proof is complete. □

3. Three-operator splitting

In this section, we consider the following inclusion problem of finding $x \in H$ such that

$$0 \in (A+B+C)(x), \tag{3.1}$$

where operators A, B, C satisfy the following assumption:

Assumption 3.1. Throughout this section the following hold:

i. Operator $A : H \rightrightarrows H$ is set-valued maximal monotone.
ii. Operator $B : H \to H$ is single-valued monotone and L_1-Lipschitz.
iii. Operator $C : H \to H$ is $1/L_2$-cocoercive, i.e., $\langle C(x)-C(y), x-y\rangle \geq \frac{1}{L_2}||C(x)-C(y)||^2$, $(\frac{1}{L_2} > 0)$.

This problem (3.1) could be solved by using the two-operator splitting algorithm in Section 2, that is, we consider the two operators: A and $(B+C)$, where $(B+C)$ is L-Lipschitz continuous with $L = L_1 + L_2$. Then, according to Theorem 2.1, the stepsize λ should satisfy

$$\lambda < \frac{1}{3(\alpha+1)L} = \frac{1}{3(\alpha+1)L_1 + 3(\alpha+1)L_2}.$$

In this section, we give a modification of λ as $\lambda < \frac{2}{3L_2+3(2\alpha+3)L_1}$. Then, our modified ISDR algorithm for solving the monotone inclusion described above is as follows

$$\begin{cases} w_n = x_n + \alpha_n(x_n - x_{n-1}), \\ x_{n+1} = J_{\lambda A}(w_n - \lambda(B+C)(x_n)) - \lambda(B(x_n) - B(x_{n-1})). \end{cases} \tag{3.2}$$

The following lemma, which play an important role in the convergence analysis of iteration 3.2, could be derived from [26, Lemma 5.1]. For the sake of completeness, we still give the proof.

Lemma 3.1. *Let $x \in (A+B+C)^{-1}(0)$, and $x_0, x_{-1} \in H$. Let (x_n) be a sequence defined by (3.2). Suppose $\lambda \in (0, \frac{2}{3L_2+(6\alpha+9)L_1})$ and $\lambda^2 L_1 L_2 < 1$. Then there exists an $\varepsilon > 0$ such that, for all $n \in N$,*

$$(1+\lambda L)||x_{n+1}-x||^2+(\lambda^2L^2+\lambda L+1)||x_n-x||^2+((2+2\alpha)\lambda L_1+\varepsilon)||x_{n+1}-x_n||^2$$
$$\leq (1+\lambda L)||x_n-x||^2+(\lambda^2L^2+\lambda L+1)||x_{n-1}-x||^2+(2+2\alpha)\lambda L_1||x_{n-1}-x_n||^2.$$

Proof. Borrowing the definition of the resolvent and (3.2), one obtains that
$x_{n+1}+\lambda(B(x_n)-B(x_{n-1}))-w_n+\lambda(B(x_n)+C(x_n))\in -A(x_{n+1}+\lambda(B(x_n)-B(x_{n-1})))$.

Since $0\in(A+B+C)(x)$, one has $-(B+C)(x)\in A(x)$. Combining this with the monotonicity of A yields that

$$\langle x_{n+1}-x_n+\lambda(B(x_n)-B(x_{n-1}))-\alpha_n(x_n-x_{n-1})+\lambda(B(x_n)+C(x_n))-\lambda(B+C)(x),$$
$$x-x_{n+1}-\lambda(B(x_n)-B(x_{n-1}))\rangle\geq 0.$$

It follows that

$$\langle x_{n+1}-x_n+\lambda(B(x_n)-B(x_{n-1}))-\alpha_n(x_n-x_{n-1})+\lambda(B(x_n)-B(x))+\lambda(C(x_n)-C(x)),$$
$$x-x_{n+1}-\lambda(B(x_n)-B(x_{n-1}))\rangle\geq 0,$$

which we can rewrite as

$$\begin{aligned}0\leq&\langle x_{n+1}-x_n,x-x_{n+1}\rangle+\langle x_{n+1}-x_n,\lambda(B(x_{n-1})-B(x_n))\rangle\\&+\langle\lambda(B(x_n)-B(x_{n-1})),x-x_{n+1}\rangle+\langle\lambda(B(x_n)-B(x_{n-1})),\lambda(B(x_{n-1})-B(x_n))\rangle\\&+\alpha_n\langle x_{n-1}-x_n,x-x_{n+1}\rangle+\alpha_n\langle x_{n-1}-x_n,\lambda(B(x_{n-1})-B(x_n))\rangle\\&+\langle\lambda(B(x_n)-B(x)),x-x_{n+1}\rangle+\langle\lambda(B(x_n)-B(x)),\lambda(B(x_{n-1})-B(x_n))\rangle\\&+\langle\lambda(C(x_n)-C(x)),x-x_n\rangle+\langle\lambda(C(x_n)-C(x)),x_n-x_{n+1}\rangle\\&+\langle\lambda(C(x_n)-C(x)),\lambda(B(x_{n-1})-B(x_n))\rangle.\end{aligned}\tag{3.3}$$

Using the $1/L_2$-cocoercivity of C, one has

$$\langle C(x_n)-C(x),x-x_n\rangle\leq -\frac{1}{L_2}||C(x_n)-C(x)||^2,$$

and

$$\langle C(x_n)-C(x),x_n-x_{n+1}\rangle\leq\frac{1}{2L_2}||C(x_n)-C(x)||^2+\frac{L_2}{2}||x_{n+1}-x_n||^2.$$

In view of the Lipschitzness of B, one obtains

$$\begin{aligned}2\langle\lambda(C(x_n)-C(x)),\lambda(B(x_{n-1})-B(x_n))\rangle&\leq\lambda L_1||x_{n-1}-x_n||^2+\lambda^3L_1||C(x_n)-C(x)||^2\\&\leq\lambda L_1||x_{n-1}-x_n||^2+\frac{\lambda}{L_2}||C(x_n)-C(x)||^2,\end{aligned}$$

which together with (3.3) deduces that

$$(1+\lambda L)||x_{n+1}-x||^2+(\lambda^2L^2+\lambda L+1)||x_n-x||^2+\left(\frac{2}{3}-\lambda L_1-\lambda L_2\right)||x_{n+1}-x_n||^2$$
$$\leq (1+\lambda L)||x_n-x||^2+(\lambda^2L^2+\lambda L+1)||x_{n-1}-x||^2+(2\alpha+2)\lambda L_1||x_{n-1}-x_n||^2.$$

Put $\varepsilon := \left(\frac{2}{3}-\lambda L_1-\lambda L_2\right)-(2\alpha+2)\lambda L_1=\frac{2}{3}-\lambda(L_2+(2\alpha+3)L_1)>0$. It follows that

$$(1+\lambda L)||x_{n+1}-x||^2+(\lambda^2L^2+\lambda L+1)||x_n-x||^2+((2+2\alpha)\lambda L_1+\varepsilon)||x_{n+1}-x_n||^2$$
$$\leq (1+\lambda L)||x_n-x||^2+(\lambda^2L^2+\lambda L+1)||x_{n-1}-x||^2+(2+2\alpha)\lambda L_1||x_{n-1}-x_n||^2.$$

The proof is complete. □

The following theorem is main result of the three-operator splitting scheme in this section.

Theorem 3.1. *Consider Algorithm (3.2) under Assumption 3.1 and assume* $(A+B+C)^{-1}(0)\neq\emptyset$. *Suppose that* $x_0, x_{-1}\in H$ *and* $\lambda\in(0,\frac{2}{3L_2+(6\alpha+9)L_1})$. *Then the sequence* (x_n) *generated by modified ISDR algorithm (3.2) converges to a point in* $(A+B+C)^{-1}(0)$ *weakly.*

Proof. From (3.1) and Theorem 2.1, one can conclude the proof immediately. So, we omit the details. □

Now, we present a corollary of Theorem 3.2. Let K be a nonempty closed convex subset of H, $B: H\to H$ be a monotone and Lipschitz continuous operator, and $C: H\to H$ be a cocoercive operator. Then, we consider the following variational inequality problem of two-operator sum form:

$$\text{Find} \quad x^*\in K \quad \text{such that} \quad \langle (B+C)(x^*), x-x^*\rangle\geq 0, \quad \forall x\in K. \qquad \text{(VIP1)}$$

Denote by N_K the normal cone of K. The problem (VIP1) is equivalent to the following variational inclusion problem:

$$\text{Find} \quad x^*\in H \quad \text{such that} \quad 0\in (N_K+B+C)(x^*).$$

Note that the operator N_K is maximally monotone. Moreover, $J_{\lambda N_K}(x)=P_K(x)$ for all $x\in H$. Thus, the following corollary follows directly from Theorem 3.1.

Corollary 3.1. *Let K be a nonempty closed convex subset of H. Let $B: H\to H$ be a monotone and L_1-Lipschitz continuous operator and $C: H\to H$ be a $1/L_2$-cocoercive operator. Assume that* $(N_K+B+C)^{-1}(0)\neq\emptyset$, $x_0, x_{-1}\in H$ *and* $\lambda\in(0,\frac{2}{3L_2+(6\alpha+9)L_1})$. *Let (x_n) be the sequence generated by the following manner:*

$$x_{n+1}=P_K(w_n-\lambda(B+C)(x_n))-\lambda(B(x_n)-B(x_{n-1})). \qquad (3.4)$$

Then, the sequence (x_n) converges weakly to a solution of problem (VIP1).

4. Numerical experiments

In order to evaluate the performance of the proposed algorithms, this section reports four numerical experiments (convex minimization, convex feasibility, signal processing, and variational inequality problem) to illustrate the convergence of the Algorithm ISDR (2.1) and the Algorithm (3.4). Moreover, we compare the suggested algorithms with the Algorithm SDR (1.5) and the Algorithm (5.2) proposed by Malitsky and Tam [26]. The codes were written in Matlab R2018a and run on a PC Desktop Intel(R) Core(TM) i5-8250M CPU@1.60 GHz 1.8 GHz, RAM 8.00 GB.

4.1. Convex minimization problems

Let f and g be two convex, lower semi-continuous functions such that f is differentiable with L-Lipschitz continuous gradient, and the proximal map of g can be computed. The convex minimization problem consists of finding $x_0 \in H$ such that

$$f(x_0) + g(x_0) \le f(x) + g(x), \forall x \in H.$$

It is known that the convex minimization problem is a special case of the inclusion problem (1.1), which consists of finding $x_0 \in H$ such that

$$0 \in \nabla f(x_0) + \partial g(x_0),$$

where ∇f is a gradient of f and ∂g is a subdifferential of g.

Example 4.1. Consider the following minimization problem:

$$\min_{x \in R^3} ||x||_2^2 + (3, 5, -1)x + 9 + ||x||_1,$$

where $x = (x_1, x_2, x_3) \in R^3$. Set $f(x) = ||x||_2^2 + (3, 5, -1)x$ and $g(x) = ||x||_1$. Thus, $\nabla h(x) = 2x + (3, 5, -1)$ and

$$(I + \partial g)^{-1}x = (\max\{|x_1| - r, 0\}\text{sign}(x_1), \max\{|x_2| - r, 0\}\text{sign}(x_2), \max\{|x_3| - r, 0\}\text{sign}(x_3)).$$

We solve this problem by Algorithm SDR and Algorithm ISDR.

We choose $\lambda = \frac{1}{4}$, $\alpha_n = 0.3$. x_0 and x_1 are generated in (0, 1) randomly. From Figure 1, we see that our algorithm converges more efficiently.

4.2. Convex feasibility problems

Let H_1 and H_2 be two real Hilbert spaces. Let $T : H_1 \to H_2$ be a bounded and linear operator, and let T^* be the adjoint of T. Let $C \subseteq H_1$ and $Q \subseteq H_2$ be nonempty, convex, and closed sets. The split feasibility problem (SFP) is formulated as follows:

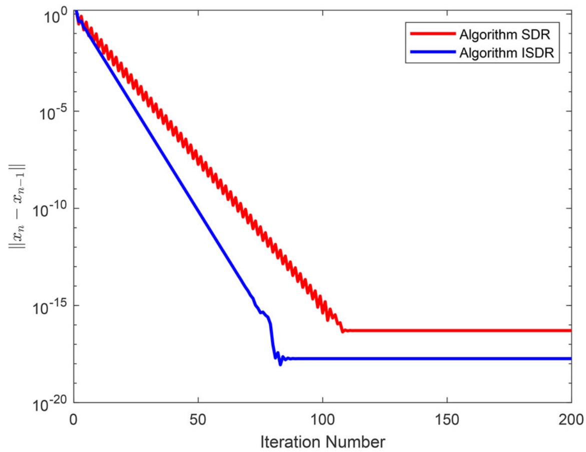


Figure 1. Comparison of Algorithms SDR and Algorithm ISDR in Example 4.1.

$$\text{find a point } x \in C \text{ such that } Tx \in Q.$$

Take $Ax := \nabla\left(\frac{1}{2}\|Tx - P_Q Tx\|^2\right) = T^*(I - P_Q)Tx$ and $B = \partial i_C$ (the indicator function), where P_Q is the metric projection onto Q. Thus, SFP has an inclusion structure. It can be seen that A is Lipschitz continuous with module $L = ||T||^2$ and B is maximally monotone; see, e.g., [27].

Example 4.2. Consider $H = L^2([0, 2\pi])$ with

$$\langle f, g \rangle = \int_0^{2\pi} f(t)g(t)dt,$$

and the associated norm given as

$$||f||_2 := \left(\int_0^{2\pi} |f(t)|^2 dt\right)^{\frac{1}{2}}, \quad \forall f, g \in L^2([0, 2\pi]).$$

Consider the half-space

$$C = \{x \in L^2([0, 2\pi] : \langle x, u \rangle \le 1\},$$

and

$$Q = \{x \in L^2([0, 2\pi]) : ||x - w||_2 \le 4\},$$

where $u : [0, 2\pi] \to \mathbb{R}$, $u(t) = 1$ for all $t \in [0, 2\pi]$, and $f : [0, 2\pi] \to \mathbb{R}$, $w(t) = \sin(t)$ for all $t \in [0, 2\pi]$. The set C and Q are nonempty, convex, and closed sets in $L^2([0, 2\pi])$. Suppose that $T : L^2([0, 2\pi]) \to L^2([0, 2\pi])$ is defined by $(Tx)(t) := x(t)$ with $(T^*x)(t) = x(t)$ and $||T|| = 1$. The problem in this example is to:

$$\text{find } x^* \in C \text{ such that } Tx^* \in Q. \tag{4.1}$$

Table 1. Comparison between proposed Algorithm ISDR, Algorithm SDR and Algorithm YMDR.

		Algorithm ISDR		Algorithm SDR		Algorithm YMDR	
Cases	Initial points	Iter.	Time(s)	Iter.	Time(s)	Iter.	Time(s)
I	$x_0 = \frac{t^2}{10}, x_1 = \frac{t^2}{10}$	14	10.81	20	42.64	20	18.89
II	$x_0 = \frac{t^2}{10}, x_1 = \frac{e^t}{2}$	31	91.19	43	118.69	43	113.46
III	$x_0 = \frac{t^2}{10}, x_1 = \frac{t^4}{24} + \frac{e^t}{10}$	28	126.51	39	163.77	39	176.61
IV	$x_0 = \frac{t^2}{10}, x_1 = \frac{t^4+3t^3+t+4}{13}$	30	123.42	41	191.91	41	233.2

Since $(Tx)(t) = x(t), \forall x \in L^2([0, 2\pi])$, (4.1) is actually a convex feasibility problem of the form:

$$\text{find } x^* \in C \cap Q.$$

Problem (4.1) can be translate to a inclusion formulation of $A + B$, where $Ax := \nabla\left(\frac{1}{2}\|Tx - P_Q Tx\|^2\right) = T^*(I - P_Q)Tx$ and $B = \partial i_C$. It is clear that A is 1-Lipschitz continues and B is maximal monotone.

In the specific calculation process, we use the following formula for the projections onto set C and set Q, respectively (see [27]),

$$P_C(z) = \begin{cases} \dfrac{1 - \int_0^{2\pi} z(t)dt}{4\pi^2} + z, & \displaystyle\int_0^{2\pi} z(t)dt > 1, \\ z, & \displaystyle\int_0^{2\pi} z(t)dt \le 1. \end{cases}$$

For $w \in L_2([0, 2\pi]$, one also has

$$P_Q(w) = \begin{cases} \sin + \dfrac{4}{\sqrt{\int_0^{2\pi} |w(t) - \sin(t)|^2 dt}}(w - \sin), & \displaystyle\int_0^{2\pi} |w(t) - \sin(t)|^2 dt > 16, \\ w, & \displaystyle\int_0^{2\pi} |w(t) - \sin(t)|^2 dt \le 16. \end{cases}$$

We compare our proposed Algorithm ISDR with Algorithm SDR and Algorithm YMDR (1.11) proposed in [26] via different initial points x_0 and x_1. We use the stopping criterion

$$E_n = \frac{1}{2}\|P_C(x_n) - x_n\|_2^2 + \frac{1}{2}\|P_Q(T(x_n)) - T(x_n)\|_2^2 < \epsilon,$$

where $\varepsilon = 10^{-4}$. Other parameters are chosen as $\lambda = 0.25$ and $\alpha_n = 0.3$. The results are presented in Table 1, Figures 2 and 3.

Remark 4.1.

- The numerical results of Example 4.2 illustrate that Algorithm ISDR is efficient, easy to implement, and, most importantly, very fast.

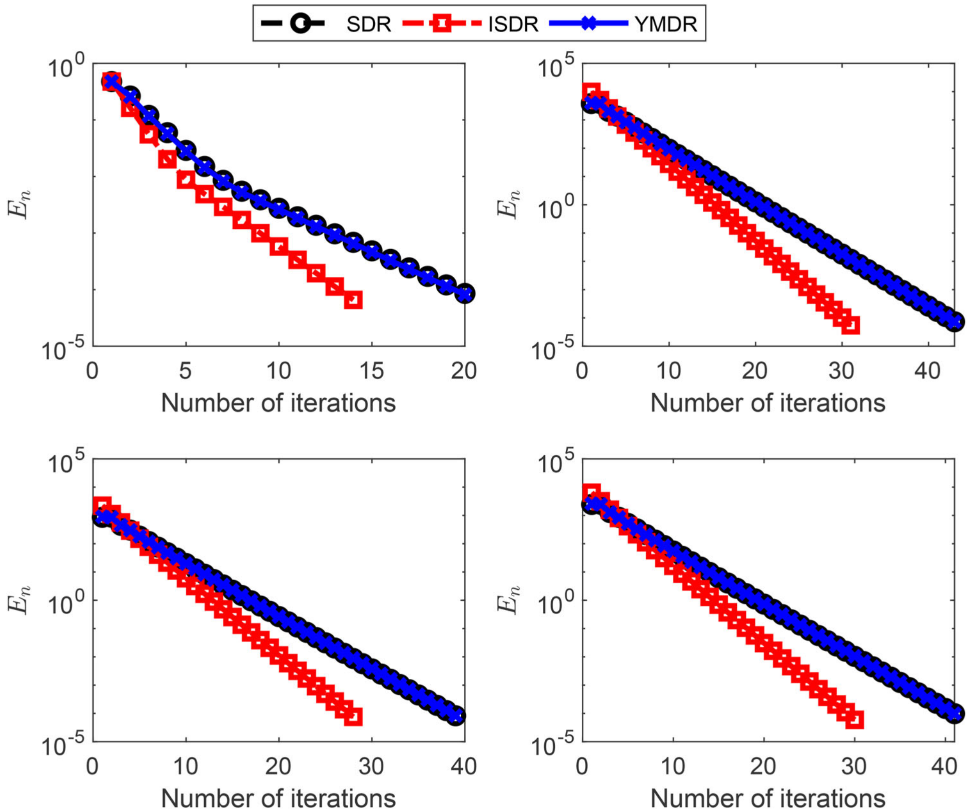


Figure 2. Comparison Algorithm ISDR, Algorithm SDR and Algorithm YMDR.

- Our proposed algorithm ISDR is consistent in the sense that the choice of initial points does not affect the required number of iterations needed to achieve desired results.
- By comparing our Algorithm ISDR with Algorithm SDR and Algorithm YMDR, we see from the same expected outcome that our algorithm is better.

4.3. Signal processing problems

Digital signal reconstruction is one of the earliest problems in the file restoration, the astronomical imaging, the medical and some other applications. Many problems in signal processing and image recovery can be formulated as a linear inverse problem, which is modeled as

$$b = Az + v, \tag{4.2}$$

where $b \in R^j$ is the noisy measurement, $A \in R^{j\times i}$ models the acquisition device, $z \in R^i$ is the original signal to be reconstructed and $v \in R^j$ is the additive noise. In this example, we restrict our attention to recover an

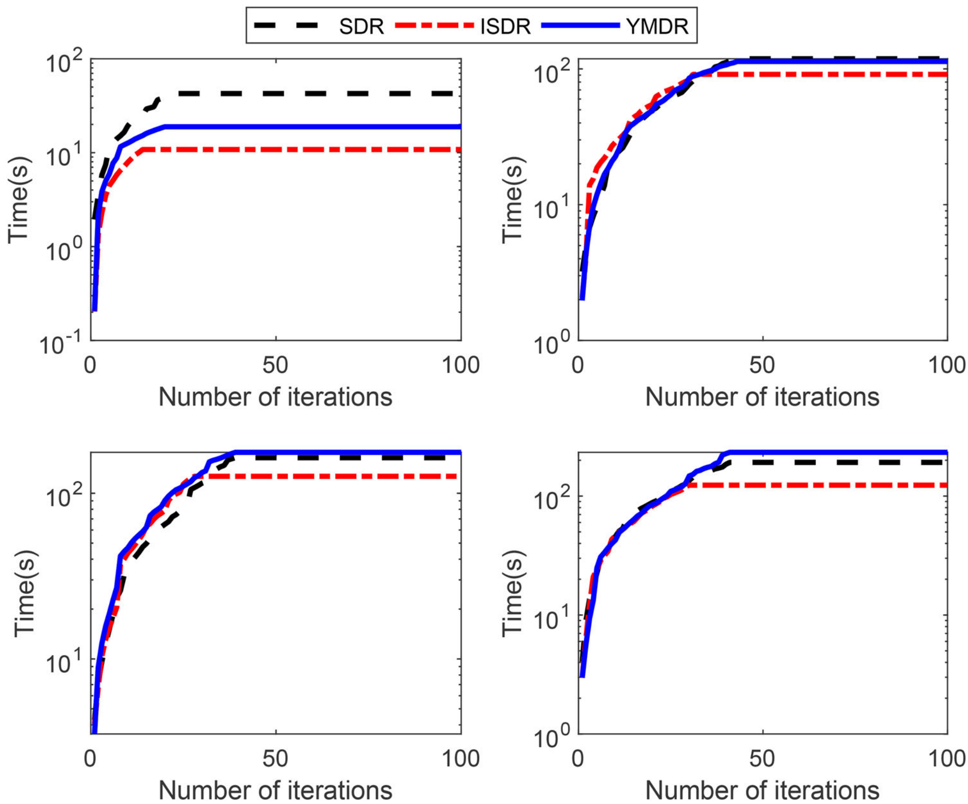


Figure 3. Comparison Algorithm ISDR, Algorithm SDR and Algorithm YMDR.

approximation of the signal z. The LASSO problem is particular case of the linear problems of type (4.2) as

$$\min_{x \in R^i} \frac{1}{2} ||Ax - b||^2 + \lambda ||x||_1, \tag{4.3}$$

where $||x||_1$ is a regularizer, the ℓ_1-norm is defined as $||x||_1 = \sum_n |x_n|$ and the parameter λ is related to the level of noise $||v||$. By substituting $A(x) = \frac{1}{2}||Ax - b||^2$ and $B(x) = \lambda ||x||_1$, we can see that problem (4.3) is reduced to

$$\text{find} \quad x \in R^i \quad \text{such that} \quad 0 \in (\partial B + \nabla A)(x).$$

We see that A is a smooth function satisfying $\nabla A(x) = A^*(Ax - b)$ and ∇A is L-Lipschitz continuous with $L = ||A^* A||$. The proximal operator of $B(x) = \lambda ||x||_1$ is given as

$$\text{prox}_{\gamma B}(x)_k = \max\left\{0, 1 - \frac{\lambda \gamma}{|x_k|}\right\} x_k.$$

Example 4.3. In this example, our aim is to recover a sparse signal $z \in \mathbb{R}^{400}$ with 16 non zero elements. The purpose of our model is to solve $b =$

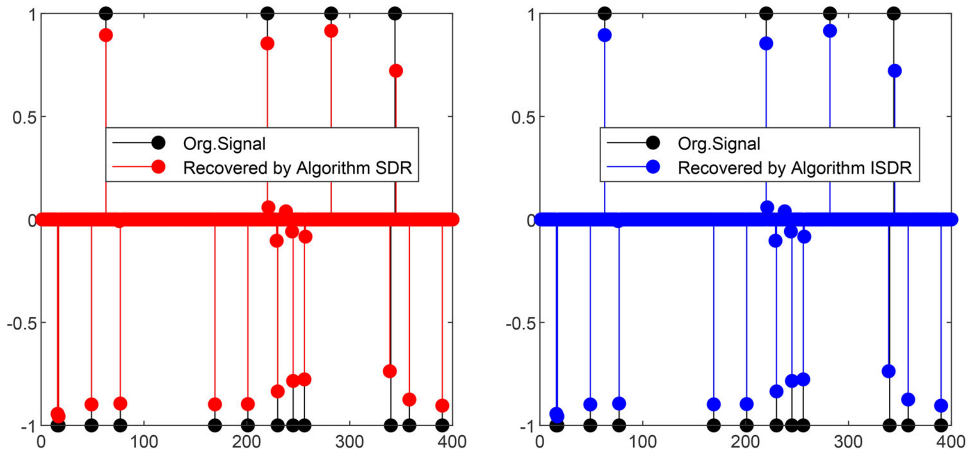


Figure 4. Comparison of Algorithm SDR and Algorithm ISDR for recovery of a sparse $k = 16$ signal.

$Az + v$, where v is a realization of Gaussian white noise with the variance is 10^{-2}. The problem can be rewritten as

$$\min_{x \in \mathbb{R}^{400}} \frac{1}{2} ||Ax - b||_2^2 + \lambda ||x||_1.$$

We take the regularization parameter $\lambda = 0.5$, the step size $\gamma = 0.2/L$ and the maximum number of iterations is 5×10^4. The corresponding parameters in SDR and ISDR are the same as in Example 4.1. Figures 4 and 5 illustrate the recovery results.

The recovery sparse signal (with 16 nonzero elements) z from noise observation vector b by Algorithm SDR and Algorithm ISDR are given in Figure 4. In Figure 5, we present the discrepancy of the term $\frac{1}{2}||Ax - b||^2$. In science and engineering, signal to noise ratio (SNR) is the ratio of signal power to the noise power, which is a measure that compares the level of a desired signal to the level of background noise. SNR often expressed in decibels such that it is defined as follows:

$$\text{SNR(dB)} = 10 \log_{10} \left(\frac{P_{\text{signal}}}{P_{\text{noise}}} \right),$$

where P_{signal} is Power of signal, P_{noise} is Power of noise. The numerical results show that $SNR = 6.77818281384 52(\text{dB})$ in Algorithm SDR and $SNR = 6.77818281384 54(\text{dB})$ in Algorithm ISDR. These images and data illustrate that both methods are effective in solving the problem, but the Algorithm ISDR is more efficient than Algorithm SDR.

4.4. Variational inclusion problems

Next, we present a numerical example involving three operators to demonstrate the Algorithm (3.4) proposed in Section 3.

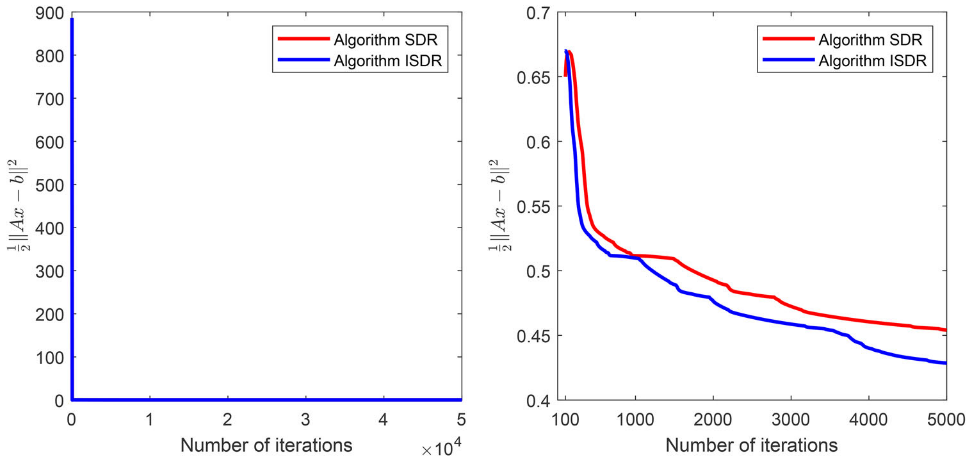
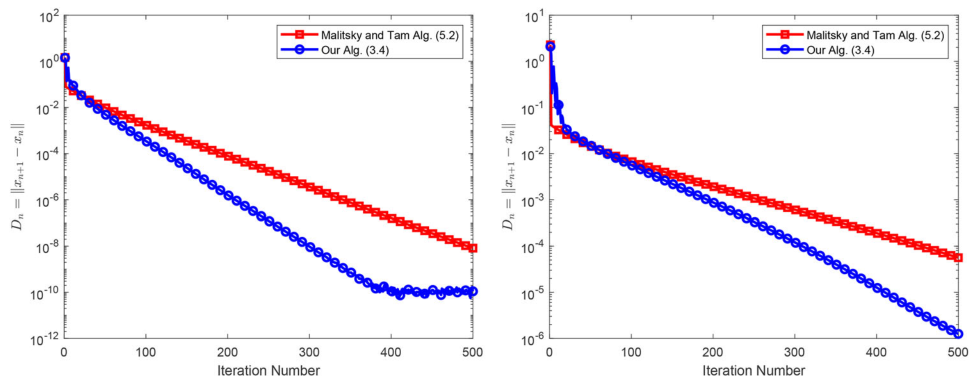


Figure 5. Comparison of Algorithm SDR and Algorithm ISDR for $\frac{1}{2}||Ax-b||^2$.



(a) $m=10$ (b) $m=20$

Figure 6. Numerical results for Example 4.4.

Example 4.4. Consider our problem in $R^m(m=10,20)$ with $A(x)=N_K(x)$ (multi-valued part), $B(x)=F(x)$ (nonlinear component), and $C(x)=Mx+q$ (linear component), where the polyhedral convex set K is defined as $K=\{x\in R^m : Gx\leq f\}$, $F(x)$ is the proximal mapping of the function $g(x)=\frac{1}{4}||x||^4$, that is

$$F(x)=\text{argmin}_{y\in R^m}\left\{\frac{||y||^4}{4}+\frac{1}{2}||y-x||^2\right\},$$

and $M\in R^{m\times m}$ (a symmetric semidefinite matrix) with its entries created randomly in $(-2,2)$, $q\in R^m$ with its entries in $(-2,2)$. It is easy to verify that operator B is monotone and Lipschitz continuous with constant

$L_1 = 1$, and operator C is monotone and Lipschitz continuous with constant $L_2 = ||M||$. We use the proposed Algorithm (3.4) to solve this problem and compare it with the Algorithm (5.2) proposed by Malitsky and Tam [26]. Our parameter settings are as follows. In our Algorithm (3.4), select the inertia parameter $\alpha = 0.6$ and the stepsize $\lambda = \frac{1.9}{3L_2+(6\alpha+9)L_1}$, and set the stepsize $\lambda = \frac{1}{4L_1+L_2}$ in the Algorithm (5.2) proposed by Malitsky and Tam. Since we do not know the exact solution of the problem, we use $D_n = ||x_{n+1} - x_n||$ to measure the iterative error of n-th step. Set the stop criterion to the maximum number of iterations 500 and the initial values $x_{-1} = x_0$. Figure 6(a) and (b) show the numerical behavior of all the algorithms in different dimensions, respectively. From the results obtained, we can see that our proposed Algorithm (3.4) is efficient and robust.

Acknowledgment

The authors would like to thank the editor and anonymous referees for their useful comments that significantly improved the quality of the paper.

ORCID

Bing Tan http://orcid.org/0000-0003-1509-1809

References

[1] Khan, A. A., Tammer, C. (2012). Generalized Dubovitskii-Milyutin approach in set-valued optimization. *Vietnam J. Math.* 40:285–304.

[2] Liu, L., Cho, S.Y., Yao, J.C. (2021). Convergence analysis of an inertial Tsengs extragradient algorithm for solving pseudomonotone variational inequalities and applications, *J. Nonlinear Var. Anals.* 5(4):627–644.

[3] Liu, L. (2019). A hybrid steepest descent method for solving split feasibility problems involving nonexpansive mappings. *J. Nonlinear Convex Anal.* 20:471–488.

[4] Shehu, Y., Iyiola, O.S. (2017). Accelerated hybrid viscosity and steepest-descent method for proximal split feasibility problems. *Optimization.* 67(4):475–492.

[5] Cho, S.Y. (2020). A convergence theorem for generalized mixed equilibrium problems and multivalued asymptotically nonexpansive mappings. *J. Nonlinear Convex Anal.* 21:1017–1026.

[6] Raguet, H., Landrieu, L. (2015). Preconditioning of a generalized forward-backward splitting and application to optimization on graphs. *SIAM J. Imaging Sci.* 8(4): 2706–2739.

[7] Liu, T., Pong, T. K. (2017). Further properties of the forward-backward envelope with applications to difference of convex programming. *Comput. Optim. Appl.* 67(3): 489–520.

[8] Lorenz, D.A., Pock, T. (2015). An inertial forward-backward algorithm for monotone inclusions. *J. Math. Imaging Vis.* 51:311–325. DOI: 10.1007/s10851-014-0523-2.

[9] Eckstein, J., Bertsekas, D.P. (1992). On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. *Math. Program*. 55: 293–318. DOI: 10.1007/BF01581204.

[10] Gibali, A., Thong, D.V. (2018). Tseng type methods for solving inclusion problems and its applications. *Calcolo*. 55(49).

[11] Cui, F., Tang, Y., Zhu, C. (2019). Convergence analysis of a variable metric forward-backward splitting algorithm with applications. *J. Inequal. Appl*. 2019(1):141.

[12] Tseng, P. (2000). A modified forward-backward splitting method for maximal monotone mappings. *SIAM J. Control Optim*. 38(2):431–446. DOI: 10.1137/S0363012998338806.

[13] Dey, S., Vetrivl, V., Xu, H.K. (2019). A neural network method for monotone variational inclusions. *J. Nonlinear Convex Anal*. 20:2387–2395.

[14] Douglas, J., Rachford, H.H. (1956). On the numerical solution of heat conduction problems in two and three space variables. *Trans. Am. Math. Soc*. 82(2):421–439.

[15] Lions, P.L., Mercier, B. (1979). Splitting algorithms for the sum of two nonlinear operators. *SIAM J. Numer. Anal*. 16(6):964–979.

[16] Svaiter, B.F. (2011). On weak convergence of the Douglas-Rachford method. *SIAM J. Control Optim*. 49(1):280–287. DOI: 10.1137/100788100.

[17] Guo, K., Han, D., Yuan, X. (2017). Convergence analysis of Douglas-Rachford splitting method for "strongly + weakly" convex programming. *SIAM J. Numer. Anal*. 55(4):1549–1577.

[18] Artacho, F.J., Borwein, J.M., Tam, M.K. (2016). Global behavior of the Douglas-Rachford method for a nonconvex feasibility problem. *J. Glob. Optim*. 65(2): 309–327.

[19] Bauschke, H.H., Dao, M.N. (2017). On the finite convergence of the Douglas-Rachford algorithm for solving (not necessarily convex) feasibility problems in Euclidean spaces. *SIAM J. Optim*. 27(1):507–537.

[20] Bot, R.I., Csetnek, E.R., Heinrich, A. (2013). A primal-dual splitting algorithm for finding zeros of sums of maximally monotone operators. *SIAM J. Optim*. 23(4): 2011–2036.

[21] Zhao, J., Zong, H. (2018). Iterative algorithms for solving the split feasibility problem in Hilbert spaces. *J. Fixed Point Theory Appl*. 20:11. DOI: 10.1007/s11784-017-0480-7.

[22] Dao, M.N., Tam, M.K. (2019). A Lyapunov-type approach to convergence of the Douglas-Rachford algorithm. *J. Glob. Optim*. 73(1):83–112.

[23] Csetnek, E.R., Malitsky, Y., Tam, M.K. (2019). Shadow Douglas-Rachford splitting for monotone inclusions. *Applied Mathematics Optimization*. 80:665–678. DOI: 10.1007/s00245-019-09597-8.

[24] Moudafi, A., Oliny, M. (2003). Convergence of a splitting inertial proximal method for monotone operators. *J. Comput. Appl. Math*. 155:447–454.

[25] Bo¸T, R.I., Csetnek, E.R. (2017). A dynamical system associated with the fixed points set of a nonexpansive operator. *J. Dyn. Differ. Equ*. 29(2017):155–168.

[26] Malitsky, Y., Tam, M.K. (2020). A forward-backward splitting method for monotone inclusions without cocoercivity. *SIAM J. Optim*. 30(2):1451–1472.

[27] Bauschke, H.H., Combettes, P.L. (2011). *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. New York: Springer.