Mathematical Programming (2020) 182:233–273
https://doi.org/10.1007/s10107-019-01403-1

FULL LENGTH PAPER

Series A

Uniqueness of DRS as the 2 operator resolvent-splitting and impossibility of 3 operator resolvent-splitting

Ernest K. Ryu[1]

Received: 21 February 2018 / Accepted: 3 April 2019 / Published online: 22 May 2019

Abstract

Given the success of Douglas–Rachford splitting (DRS), it is natural to ask whether DRS can be generalized. Are there other 2 operator resolvent-splittings sharing the favorable properties of DRS? Can DRS be generalized to 3 operators? This work presents the answers: no and no. In a certain sense, DRS is the unique 2 operator resolvent-splitting, and generalizing DRS to 3 operators is impossible without lifting, where lifting roughly corresponds to enlarging the problem size. The impossibility result further raises a question. How much lifting is necessary to generalize DRS to 3 operators? This work presents the answer by providing a novel 3 operator resolvent-splitting with provably minimal lifting that directly generalizes DRS.

Keywords Douglas–Rachford splitting · Splitting methods · Maximal monotone operators · Lower bounds · First-order methods

Mathematics Subject Classification 47H05 · 47H09 · 65K10 · 90C25

1 Introduction

In 1979, Lions and Mercier presented Douglas–Rachford splitting (DRS) which solves the monotone inclusion problem

$$\underset{x \in \mathbb{R}^d}{\text{find}} \quad 0 \in (A + B)x$$

with

$$z^{k+1} = (1 - \theta/2)z^k + (\theta/2)(2J_{\alpha A} - I)(2J_{\alpha B} - I)z^k$$

✉ Ernest K. Ryu
eryu@math.ucla.edu

1 7324 Mathematical Sciences, UCLA, Los Angeles, CA 90095, USA

for any $\alpha > 0$ and $\theta \in (0, 2)$, where A and B are maximal monotone operators and $J_{\alpha A}$ and $J_{\alpha B}$ are their resolvents [27,38,44]. Since its introduction, DRS has enjoyed great popularity and has provided great value to the field of optimization.

Given the success of DRS, one may ask the following two questions:

1. Are there other 2 operator resolvent-splittings?
2. Can we generalize DRS to 3 operators?

In fact, the second question has been a long-standing open problem posed by Lions and Mercier themselves: "[T]he convergence seems difficult to prove ... in the case of a sum of 3 operators." After all, identifying why a tool works and generalizing it is a common and often fruitful exercise in mathematics.

This work presents the answers to these questions: no and no. In a certain sense, DRS is the unique 2 operator resolvent-splitting. In a certain sense, there is no 3 operator resolvent-splitting without lifting, where lifting roughly corresponds to enlarging the problem size.

This impossibility result further raises the following question:

3. To generalize DRS to 3 operators, how much lifting is necessary?

This work presents the answer by providing a novel 3 operator resolvent-splitting with provably minimal lifting.

Background. To discuss what constitutes a generalization of DRS, we first point out a few key properties of DRS. Perhaps a generalization of DRS should satisfy these as well.

1. DRS is a *resolvent-splitting* in that it is constructed with scalar multiplication, addition, and resolvents.
2. DRS is *frugal* in that it uses $J_{\alpha A}$ and $J_{\alpha B}$ only once per iteration.
3. DRS *converges unconditionally* in that it works for any maximal monotone A and B.
4. DRS uses *no lifting* in that the fixed-point mapping maps from $\mathbb{R}^d$ to $\mathbb{R}^d$, where $x \in \mathbb{R}^d$. In other words, DRS does not enlarge the problem size.

Consider the proximal point method (PPM) [7,41,42,49], which finds an $x \in \mathbb{R}^d$ such that $0 \in Ax$ with

$$x^{k+1} = J_{\alpha A} x^k$$

for any $\alpha > 0$ and maximal monotone A. DRS generalizes PPM, and both methods are frugal, converge unconditionally, use no lifting, and rely on resolvents. Therefore, to require the 4 properties in a generalization of DRS seems reasonable.

Many other splittings have been presented since DRS, and they have certainly provided great value to the field of optimization. These splittings solve a wide range of different problem classes and are designed to be effective under a wide range of different computational considerations. Many of them include DRS as a special case and therefore are generalizations of DRS, in that sense. However, they do not satisfy the 4 stated properties and therefore are not generalizations of DRS, in this sense.

Forward–backward splitting (FBS) [43],

$$x^{k+1} = J_{\alpha B}(I - \alpha A)x^k,$$

which requires A to be cocoercive, is frugal, uses no lifting, but is not a resolvent-splitting. Primal-dual hybrid gradient method (PDHG) [13,30,45,59], also known as Chambolle–Pock,

$$\begin{aligned} x^{k+1} &= J_A(x^k - \alpha u^k) \\ u^{k+1} &= (I - J_B)(u^k + \alpha(2x^{k+1} - x^k)) \end{aligned}$$

is frugal but uses lifting. Davis–Yin splitting (DYS) [25], which finds an $x \in \mathbb{R}^d$ such that $0 \in (A + B + C)x$, where C is cocoercive,

$$z^{k+1} = (I - J_{\alpha B} + J_{\alpha A} \circ (2J_{\alpha B} - I - \alpha C \circ J_{\alpha B}))z^k$$

is frugal, uses no lifting, but is not a resolvent-splitting. Other methods, such as FBFS [54], PPXA [20,21], PDFP^{2}O/PAPC [15,28,39], RFBS [1], Condat–Vũ [24,55], GFBS [47], PD3O [57], PDFP [16], AFBA [36], FBHFS [10], FDRS [8,46], FRB [40], projective splitting [19,29,32,33], and the methods of [5,9,18,22] all fail to satisfy the 4 properties.

Organization of the paper. In Sect. 2, we show that DRS is the only frugal, unconditionally convergent resolvent-splitting without lifting for the 2 operator problem. We do so by characterizing all frugal resolvent-splittings without lifting and showing that DRS is the only one among them that unconditionally converges.

In Sect. 3, we show that there is no resolvent-splitting without lifting for the 3 operator problem, even if the splitting is not frugal and not convergent. In particular, we show such a scheme without lifting cannot be a fixed-point encoding.

In Sect. 4, we define and quantify the notion of lifting for the 3 operator problem. We then provide a novel frugal, unconditionally convergent resolvent-splitting with provably minimal lifting for the 3 operator problem that directly generalizes DRS.

Definitions We briefly review some standard notation and results of operator theory. Interested readers can find in-depth discussion of these concepts in standard references such as [3,50].

Write $\langle \cdot, \cdot \rangle$ for the standard Euclidean inner product in $\mathbb{R}^d$. We say A is an operator on $\mathbb{R}^d$ if A maps points of $\mathbb{R}^d$ to subsets of $\mathbb{R}^d$. Given a matrix $M \in \mathbb{R}^{d \times d}$ also write $M : \mathbb{R}^d \to \mathbb{R}^d$ to denote the linear operator defined by the matrix M. In particular, write I for both the identity operator and the identity matrix. Write $\mathcal{M}(\mathbb{R}^d)$ for the set of all maximal monotone operators on $\mathbb{R}^d$. For any maximal monotone operator A and $\alpha > 0$, write

$$J_{\alpha A} = (I + \alpha A)^{-1}$$

for the resolvent of A. A mapping $T:\mathbb{R}^d\to\mathbb{R}^d$ is nonexpansive if

$$\|Tx-Ty\|^2\le\|x-y\|^2$$

for all $x,y\in\mathbb{R}^d$. A mapping $F:\mathbb{R}^d\to\mathbb{R}^d$ is firmly nonexpansive if

$$\|Fx-Fy\|^2\le\langle x-y,Fx-Fy\rangle$$

for all $x,y\in\mathbb{R}^d$. Resolvents are firmly nonexpansive. Given a mapping $T:\mathbb{R}^d\to\mathbb{R}^d$ and a starting point $z^0\in\mathbb{R}^d$, we call

$$z^{k+1}=Tz^k$$

the fixed-point iteration with respect to T. A fixed-point iteration with respect to a nonexpansive mapping need not converge. A mapping $T:\mathbb{R}^d\to\mathbb{R}^d$ is averaged if it can be expressed as $T=(1-\theta)I+\theta R$, where $R:\mathbb{R}^d\to\mathbb{R}^d$ is nonexpansive and $\theta\in(0,1)$. Note that R and T share the same fixed points. The fixed-point iteration with respect to an averaged mapping $T:\mathbb{R}^d\to\mathbb{R}^d$ converges in that $z^k\to z^\star$ where $Tz^\star=z^\star$, if a fixed point exists.

For any $A\in\mathcal{M}(\mathbb{R}^d)$, write $\mathbf{zer}A=\{x\mid 0\in Ax\}$ for the set of zeros of A. Consider the monotone inclusion problem of finding an element of $\mathbf{zer}(A+B)$, where $A,B\in\mathcal{M}(\mathbb{R}^d)$. Peaceman–Rachford splitting (PRS) [38,44] is the fixed-point iteration

$$z^{k+1}=(2J_{\alpha A}-I)(2J_{\alpha B}-I)z^k$$

with $\alpha>0$. PRS is not guaranteed to converge. Douglas–Rachford splitting (DRS) is the fixed-point iteration

$$z^{k+1}=(1-\theta/2)z^k+(\theta/2)(2J_{\alpha A}-I)(2J_{\alpha B}-I)z^k.$$

with $\alpha>0$ and $\theta\in(0,2)$. (Some may call this "relaxed PRS".) DRS is guaranteed to converge in the sense that $z^k\to z^\star$ for some $z^\star$ where $J_{\alpha B}z^\star\in\mathbf{zer}(A+B)$, if $\mathbf{zer}(A+B)$ is not empty.

2 Uniqueness of DRS as the unique frugal, unconditionally convergent 2 operator resolvent-splitting without lifting

In this section, we define what a frugal, unconditionally convergent 2 operator resolvent-splitting without lifting is and prove DRS is the only such splitting.

2.1 Definitions

When reading the definitions, it is helpful to think of DRS as a specific example. In the terminology and notation we soon establish, DRS is an unconditionally convergent frugal resolvent-splitting without lifting and $d' = d$, $T(A, B, z) = (1 - \theta/2)I + (\theta/2)(2J_{\alpha A} - I)(2J_{\alpha B} - I)$, and $S(A, B, z) = J_{\alpha B}$.

Given a dimension d, define the problem class (2op-$\mathbb{R}^d$) to be the collection of monotone inclusion problems of the form

$$\underset{x \in \mathbb{R}^d}{\text{find}} \quad 0 \in (A + B)x \tag{2op-$\mathbb{R}^d$}$$

with $A, B \in \mathcal{M}(\mathbb{R}^d)$.

Fixed-point encoding. A pair of functions (T, S) is a *fixed-point encoding* for the problem class (2op-$\mathbb{R}^d$) if

$$\exists z^\star \in \mathbb{R}^{d'} \text{ such that } \begin{pmatrix} T(A, B, z^\star) = z^\star \\ S(A, B, z^\star) = x^\star \end{pmatrix} \quad \Leftrightarrow \quad 0 \in (A + B)(x^\star)$$

for all $A, B \in \mathcal{M}(\mathbb{R}^d)$. We call

$$T : \mathcal{M}(\mathbb{R}^d) \times \mathcal{M}(\mathbb{R}^d) \times \mathbb{R}^{d'} \to \mathbb{R}^{d'}$$

the *fixed-point mapping* and

$$S : \mathcal{M}(\mathbb{R}^d) \times \mathcal{M}(\mathbb{R}^d) \times \mathbb{R}^{d'} \to \mathbb{R}^d,$$

the *solution mapping*. To clarify, a fixed-point encoding is defined for the entire problem class (2op-$\mathbb{R}^d$), rather than a single instance of the monotone inclusion problem.

When we fix $A, B \in \mathcal{M}(\mathbb{R}^d)$, fixed points of $T(A, B, \cdot) : \mathbb{R}^{d'} \to \mathbb{R}^{d'}$ corresponds to zeros of $A + B$. We say that points in $\mathbf{zer}(A + B)$ are *encoded* as fixed points of $T(A, B, \cdot)$. For notational simplicity, we often drop the dependency on A and B and write Tz and Sz for $T(A, B, z)$ and $S(A, B, z)$.

In this section, we only consider $d' = d$, as we limit our attention to fixed-point encodings without lifting (formally defined soon). In general, however, the dimension d of problems in (2op-$\mathbb{R}^d$) and the dimension d' of the fixed-point mapping need not be the same. The purpose of allowing $d' \neq d$ will become clearer later in Sect. 4, where an analogously defined d' is larger than d.

Under this definition, DRS is a *collection* of fixed-point encodings. For each choice of d, $\alpha > 0$, $\theta \in (0, 2)$, and $\eta \in \mathbb{R}$ the pair of functions (T, S) defined by

$$\begin{aligned} T(A, B, z) &= (1 - \theta/2)z + (\theta/2)(2J_{\alpha A} - I)(2J_{\alpha B} - I)z, \\ S(A, B, z) &= \eta J_{\alpha A} z + (1 - \eta) J_{\alpha B}(2J_{\alpha A} - I)z \end{aligned}$$

is a instance of DRS and it is a fixed-point encoding for the problem class (2op-$\mathbb{R}^d$).

Frugal resolvent-splitting without lifting. Loosely speaking, (T, S) is a *resolvent-splitting* for the problem class (2op-$\mathbb{R}^d$) if it is a fixed-point encoding constructed with resolvents of A and B, addition, and scalar multiplication. Loosely speaking, (T, S) is *frugal* if it uses $J_{\alpha A}$ and $J_{\beta B}$ once, in that a single evaluation of $J_{\alpha A}$ and a single evaluation of $J_{\beta B}$ is used to evaluate *both* Tz and Sz for any z. (T, S) is *without lifting* if $T(A, B, \cdot) : \mathbb{R}^d \to \mathbb{R}^d$ and $S(A, B, \cdot) : \mathbb{R}^d \to \mathbb{R}^d$ for any $A, B \in \mathcal{M}(\mathbb{R}^d)$, i.e., if $d' = d$.

We now make the definitions precise. Let I be the "identity mapping" defined as $I : \mathcal{M}(\mathbb{R}^d) \times \mathcal{M}(\mathbb{R}^d) \times \mathbb{R}^d \to \mathbb{R}^d$ and $I(A, B, z) = z$ for any $A, B \in \mathcal{M}(\mathbb{R}^d)$ and $z \in \mathbb{R}^d$. Let $J_{\alpha,1}$ be the resolvent with respect to the first operator defined as $J_{\alpha,1} : \mathcal{M}(\mathbb{R}^d) \times \mathcal{M}(\mathbb{R}^d) \times \mathbb{R}^d \to \mathbb{R}^d$ and $J_{\alpha,1}(A, B, z) = J_{\alpha A}(z)$ for any $A, B \in \mathcal{M}(\mathbb{R}^d)$ and $z \in \mathbb{R}^d$. Define $J_{\beta,2}$ likewise with $J_{\beta,2}(A, B, z) = J_{\beta B}(z)$. Define the class of mappings

$$\mathcal{F}_0 = \{I\} \cup \{J_{\alpha,1} \mid \alpha > 0\} \cup \{J_{\beta,2} \mid \beta > 0\}.$$

Recursively define

$$\mathcal{F}_{i+1} = \{F + G \mid F, G \in \mathcal{F}_i\} \cup \{F \circ G \mid F, G \in \mathcal{F}_i\} \cup \{\gamma F \mid F \in \mathcal{F}_i,\ \gamma \in \mathbb{R}\}$$

for $i = 0, 1, 2, \ldots$. The "composition" $F \circ G$ is defined with

$$(F \circ G)(A, B, z) = F(A, B, G(A, B, z))$$

for any $z \in \mathbb{R}^d$ and $A, B \in \mathcal{M}(\mathbb{R}^d)$. Note that $\mathcal{F}_0 \subset \mathcal{F}_1 \subset \mathcal{F}_2 \subset \cdots$. Finally define

$$\mathcal{F} = \bigcup_{i=0}^{\infty} \mathcal{F}_i.$$

To clarify, elements of $\mathcal{F}$ map $\mathcal{M}(\mathbb{R}^d) \times \mathcal{M}(\mathbb{R}^d) \times \mathbb{R}^d$ to $\mathbb{R}^d$. If $R \in \mathcal{F}$ and $A, B \in \mathcal{M}(\mathbb{R}^d)$, then $R(A, B, \cdot) : \mathbb{R}^d \to \mathbb{R}^d$. These mappings are constructed with (finitely many) resolvents of A and B, addition, and scalar multiplication.

As an aside, we could have defined $\mathcal{F}$ as the "near-ring" generated by $J_{\alpha,1}$ and $J_{\beta,2}$ for all $\alpha > 0$ and $\beta > 0$ and γI for all $\gamma \in \mathbb{R}$. The set is not a ring because $T \circ (U + V) \neq T \circ U + T \circ V$ for non-linear functions.

We say (T, S) is a *resolvent-splitting without lifting* for the problem class (2op-$\mathbb{R}^d$), if (T, S) is a fixed-point encoding for the problem class (2op-$\mathbb{R}^d$), and $T, S \in \mathcal{F}$. (Remember, $T \in \mathcal{F}$ implies $T(A, B, \cdot) : \mathbb{R}^d \to \mathbb{R}^d$.)

When $T, S \in \mathcal{F}$, one can evaluate $T(A, B, z)$ and $S(A, B, z)$ for given $z \in \mathbb{R}^d$ and $A, B \in \mathcal{M}(\mathbb{R}^d)$ in finitely many steps, where each step is scalar multiplication, vector addition, or a resolvent evaluation. We say (T, S) is *frugal* if it has a step-by-step (serial) evaluation procedure such that exactly one step computes $J_{\alpha A}$, exactly one step computes $J_{\beta B}$, and both $T(A, B, z)$ and $S(A, B, z)$ are output at the end.

Unconditional convergence. We say (T, S) *converges unconditionally* for the problem class (2op-$\mathbb{R}^d$) if

$$T^k z^0 \to z^\star, \quad Sz^\star \in \mathbf{zer}(A+B)$$

for any $z^0 \in \mathbb{R}^{d'}$ and $A, B \in \mathcal{M}(\mathbb{R}^d)$ as $k \to \infty$, when $\mathbf{zer}(A+B) \neq \emptyset$. To clarify,

$$T^k = \underbrace{T \circ T \circ \cdots \circ T}_{k \text{ times}}.$$

We say the convergence is unconditional because there are no conditions on the operators $A, B \in \mathcal{M}(\mathbb{R}^d)$ or the starting point $z^0 \in \mathbb{R}^d$.

For example, with DRS, the z^k-iterates do not, in general, converge to a solution. Rather, $z^k \to z^\star$, where $J_{\alpha B} z^\star$ is a solution to the monotone inclusion problem, when a solution exists.

The notion of unconditional convergence is unrelated to weak and strong convergence. In infinite dimensional spaces, we would require the convergence $T^k z^0 \to z^\star$ to hold weakly, but weak and strong convergence coincide in finite dimensions. We avoid infinite dimensional spaces because defining the notion of lifting would be awkward when $d = \infty$.

Equivalence. Given a fixed-point iteration, we can scale it with a nonzero scalar to get another one that is essentially the same, i.e.,

$$z^{k+1} = T(z^k) \quad \Leftrightarrow \quad az^{k+1} = aT(a^{-1}az^k)$$

for any $a \in \mathbb{R}$ such that $a \neq 0$. Given resolvent-splitting, we can swap the role of A and B to get another one that is conceptually no different, i.e.,

$$(T(A, B, \cdot), S(A, B, \cdot)) \quad \Leftrightarrow \quad (T(B, A, \cdot), S(B, A, \cdot)).$$

Two resolvent-splittings without lifting are *equivalent* if one can be obtained from the other through scaling with a nonzero scalar and/or swapping the role of A and B.

2.2 Uniqueness result

Theorem 1 *Up to equivalence, (T, S) is a frugal resolvent-splittings without lifting for the problem class (2op-$\mathbb{R}^d$) if and only if it is of the form*

$$\begin{aligned} x_1 &= J_{\alpha A} z \\ x_2 &= J_{\beta B}((1+\beta/\alpha)x_1 - (\beta/\alpha)z) \\ T(z) &= z + \theta(x_2 - x_1) \\ S(z) &= \eta x_1 + (1-\eta)x_2 \end{aligned}$$

for some $\alpha, \beta > 0$, $\theta \neq 0$, and $\eta \in \mathbb{R}$.

Note that Theorem 1 says nothing about convergence. Theorem 2 characterizes the splittings of Theorem 1 that do converge.

Theorem 2 (T, S) *of* Theorem 1 *converges unconditionally if and only if* $\alpha = \beta$ *and* $\theta \in (0, 2)$ *if* $d \geq 2$.

When $d = 1$, the splittings (T, S) of Theorem 1 may converge under more general conditions, but we do not pursue this discussion.

Corollary 1 *Up to equivalence, the class of DRS splittings (the collection parameterized by* $\alpha > 0$, $\theta \in (0, 2)$*, and* $\eta \in \mathbb{R}$*) are the only frugal, unconditionally convergent resolvent-splittings without lifting for the problem class* (2op-$\mathbb{R}^d$) *when* $d \geq 2$.

Proof A frugal resolvent-splittings without lifting must be equivalent to a splitting of the form of Theorem 1. If it is furthermore unconditionally convergent, then, by Theorem 2, we have $\alpha = \beta$ and $\theta \in (0, 2)$, which corresponds to DRS. □

2.3 Proof of Theorem 1

2.3.1 Outline

The main part of proof, which shows that any frugal resolvent-splitting without lifting is of the form of Theorem 1, can be divided into in roughly three steps. In the first step, we represent a given resolvent-splitting (T, S) with a linear system of equations, and simplify the system using Gaussian elimination. In the second step, we show that the system of linear equalities must imply certain equalities one would expect from a fixed-point encoding. This is done by using a Farkas-type lemma to take a certain element from the null space of the linear system and using it to construct the counter example. In the third step, we use the conclusion of the second step to eliminate and characterize the parameters of (T, S).

2.3.2 Proof

Showing that (T, S) of Theorem 1 is indeed a fixed-point encoding is straightforward. Let $x^\star \in \mathbb{R}^d$ satisfy $0 \in (A + B)x^\star$. Let $\tilde{A}x^\star \in Ax^\star$ and $\tilde{B}x^\star \in Bx^\star$ such that $\tilde{A}x^\star + \tilde{B}x^\star = 0$, and let $z_0 = x^\star + \alpha\tilde{A}x^\star$. Then $x_1 = x_2 = x^\star$, $Tz_0 = z_0$, and $Sz_0 = x^\star$. On the other hand, assume $T(A, B, z^\star) = z^\star$. Then $x_1 = x_2$. Write $x^\star = x_1 = x_2$, $\tilde{A}x^\star = (1/\alpha)(z^\star - x^\star)$, and $\tilde{B}x^\star = (1/\alpha)(x^\star - z^\star)$. Then $\tilde{A}x^\star \in Ax^\star$, $\tilde{B}x^\star \in Bx^\star$, and $\tilde{A}x^\star + \tilde{B}x^\star = 0$, which implies $x^\star = S(z^\star)$ is a solution.

We now need to show the other direction, that any frugal resolvent-splitting without lifting for the problem class (2op-$\mathbb{R}^d$) is of the form of Theorem 1, up to equivalence.

First, we discuss the following Farkas-type lemma.

Lemma 1 *Let* $M \in \mathbb{R}^{m\times n}$ *and* $c \in \mathbb{R}^n$ *be fixed coefficients, and let* $v \in \mathbb{R}^n$ *be a variable. If there is a* $w \in \mathbb{R}^m$ *such that* $w^T M = c^T$ *then the linear equalities* $Mv = 0$ *imply the linear equality* $c^T v = 0$*. If there is no such* w*, then there is an instance of* $v \in \mathbb{R}^n$ *such that* $Mv = 0$ *but* $c^T v \neq 0$.

An equivalent way to state Lemma 1 is to say that $Mv = 0$ implies $c^T v = 0$ if and only if we can linearly combine the rows of $Mv = 0$ to obtain $c^T v = 0$. We say Lemma 1 is of Farkas-type as it resembles Farkas' result on systems of linear inequalities [31]. For a systematic study on Farkas-type theorems, see [4,26]. Lemma 1 can be directly and easily proved with standard linear algebra.

We now proceed onto the main proof. Let (T, S) be a frugal resolvent-splitting without lifting.

Consider an evaluation procedure of (T, S) that establishes frugality. In the step-by-step computation, either $J_{\alpha A}$ or $J_{\beta B}$ is evaluated before the other. Without loss of generality, assume $J_{\alpha A}$ is evaluated before $J_{\beta B}$ in this ordering, since we can otherwise consider $(\tilde{T}, \tilde{S})$ defined with

$$\tilde{T}(A, B, z) = T(B, A, z), \qquad \tilde{S}(A, B, Z) = S(B, A, z),$$

the equivalent splitting with the order of A and B swapped.

Consider the evaluation of Tz_0 and Sz_0 for $z_0 \in \mathbb{R}^d$. Write z_1 and z_2 for the inputs and x_1 and x_2 for the outputs of the resolvent evaluations with respect to A and B, i.e., $x_1 = J_{\alpha A} z_1$ and $x_2 = J_{\beta B} z_2$. Define $\tilde{A}x_1$ and $\tilde{B}x_2$ with $x_1 + \alpha \tilde{A} x_1 = z_1$ and $x_2 + \beta \tilde{B} x_2 = z_2$. By definition of resolvents, we have $\tilde{A}x_1 \in Ax_1$ and $\tilde{B}x_2 \in Bx_2$.

All computational steps except the evaluations of $J_{\alpha A}$ and $J_{\beta B}$ amount to forming linear combinations of previous information since scalar multiplication and vector addition are the only other operations allowed in (T, S). Therefore, we can express the evaluation of Tz_0 and Sz_0 as

$$0 = \begin{bmatrix} * & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ * & * & * & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \beta & 0 \\ * & * & * & * & * & 1 & * & * & 0 \\ * & * & * & * & * & * & * & * & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}. \tag{1}$$

Each scalar in the matrix represents a $d \times d$ block. The symbol $*$ denotes a fixed scalar coefficient that we have not yet parameterized. Row 1 defines z_1, the input to $J_{\alpha A}$. Row 2 represents $x_1 = J_{\alpha A} z_1 \Leftrightarrow x_1 + \alpha A x_1 \ni z_1$. Row 3 defines z_2, the input to $J_{\beta B}$. Row 4 represents $x_2 = J_{\beta B} z_2 \Leftrightarrow x_2 + \beta B x_2 \ni z_2$. Row 5 defines Tz_0. Row 6 defines Sz_0. We will simplify the system first and then explicitly parameterize the coefficients to keep the notation tractable.

Next, we simplify the system (1). Permute the rows of (1) to get the equivalent linear system

$$0 = \begin{bmatrix} * & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & 1 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & * & * & 1 & * & * & * \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \beta & 0 \\ * & * & * & * & * & * & * & * & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}. \tag{2}$$

Since permuting the rows is a reversible process, (1) and (2) are equivalent. In the step-by-step evaluation procedure of (T, S), the evaluation of T or S completes before the other. As the first case, assume the evaluation of S completes first, which means the evaluation of S does not depend on the evaluation of T. Then the linear system is of the form

$$0 = \begin{bmatrix} * & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & 1 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & * & * & 1 & * & * & * \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \beta & 0 \\ * & * & * & * & * & \mathbf{0} & * & * & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}.$$

The **boldface symbols** denote where to pay attention in the linear systems. Perform Gaussian elimination to get

$$0 = \begin{bmatrix} * & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & 1 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & * & * & 1 & * & * & \mathbf{0} \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \beta & 0 \\ * & * & * & * & * & 0 & * & * & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}.$$

This corresponds to left-multiplication by the invertible matrix

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & * \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

As the other case, assume evaluation of T completes first, which means the evaluation of T does not depend on the evaluation of S. Then the linear system is of the form

$$0 = \begin{bmatrix} * & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & 1 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & * & * & 1 & * & * & \mathbf{0} \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \beta & 0 \\ * & * & * & * & * & * & * & * & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}.$$

Perform Gaussian elimination to get

$$0 = \begin{bmatrix} * & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & 1 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & * & * & 1 & * & * & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \beta & 0 \\ * & * & * & * & * & \mathbf{0} & * & * & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}. \tag{3}$$

This corresponds to left-multiplication by the invertible matrix

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & * & 0 & 0 & 1 \end{bmatrix}.$$

Regardless of which of the two cases we start from, we arrive at the same linear system (3). Continue the Gaussian elimination to get

$$0 = \begin{bmatrix} * & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ * & \mathbf{0} & * & 1 & 0 & 0 & 0 & 0 & 0 \\ * & \mathbf{0} & * & \mathbf{0} & * & 1 & \mathbf{0} & \mathbf{0} & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \beta & 0 \\ * & \mathbf{0} & * & \mathbf{0} & * & 0 & \mathbf{0} & \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}.$$

This corresponds to left-multiplying (3) by the invertible matrix

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ * & 1 & 0 & 0 & 0 & 0 \\ * & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ * & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & * & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & * & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & * & * & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & * & * & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

We now explicitly parameterize the unspecified parameters one at a time.

$$0 = \begin{bmatrix} \boldsymbol{-a} & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ * & 0 & * & 1 & 0 & 0 & 0 & 0 & 0 \\ * & 0 & * & 0 & * & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \beta & 0 \\ * & 0 & * & 0 & * & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}.$$

The role of a is to define $z_1 = az_0$. So the evaluation of (T, S) starts with $J_{\alpha A}(az_0)$. If $a = 0$, then $J_{\alpha A}$ ignores the input z_0 and always uses 0 as the input. Since (T, S) accesses A only through the evaluation of $J_{\alpha A}$, how is it possible that (T, S) evaluates $J_{\alpha A}$ only at 0 and still encodes the zeros of $A + B$?

We now show $a \neq 0$. Assume $a = 0$ for contradiction. Let

$$A(x) = c_1 x, \qquad B(x) = c_2,$$

where $c_1 > 0$ is unspecified and $0 \neq c_2 \in \mathbb{R}^d$. Then $J_{\alpha A}0 = 0$, and the mappings T and S are independent of the value of c_1. So the set of fixed points of T and the set of $Sz^\star$, where $z^\star$ is a fixed point of T, do not depend on c_1. However, the solution $\{-c_1^{-1}c_2\} = \mathbf{zer}(A+B)$ does depend on c_1. Since (T, S) is assumed to be a fixed-point encoding, T must have a fixed-point $z^\star$ and it must satisfy $-c_1^{-1}c_2 = Sz^\star$, which is a contradiction.

Knowing $a \neq 0$, we can absorb the top-left a into z_0 and left-multiply by an invertible matrix to get the equivalent system

$$0 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \boldsymbol{a} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{-1} & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ * & 0 & * & 1 & 0 & 0 & 0 & 0 & 0 \\ * & 0 & * & 0 & * & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \beta & 0 \\ * & 0 & * & 0 & * & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{a} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}.$$

This further simplifies to the equivalent system

$$0 = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ * & 0 & * & 1 & 0 & 0 & 0 & 0 & 0 \\ * & 0 & * & 0 & * & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \beta & 0 \\ * & 0 & * & 0 & * & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{a} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ \boldsymbol{aT(a^{-1}az_0)} \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ \boldsymbol{S(a^{-1}az_0)} \end{bmatrix}.$$

By redefining $(T(z_0), S(z_0))$ to be the equivalent scaled splitting $(aT(a^{-1}az_0), S(a^{-1}az_0))$, we get the equivalent system

$$0 = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \boldsymbol{\theta_1} & 0 & \boldsymbol{\theta_2} & 1 & 0 & 0 & 0 & 0 & 0 \\ \boldsymbol{\theta_3} & 0 & \boldsymbol{\theta_4} & 0 & \boldsymbol{\theta_5} & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \beta & 0 \\ \boldsymbol{\theta_6} & 0 & \boldsymbol{\theta_7} & 0 & \boldsymbol{\theta_8} & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{z_0} \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ \boldsymbol{Tz_0} \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ \boldsymbol{Sz_0} \end{bmatrix}, \tag{4}$$

where we have now explicitly parameterized the remaining parameters as $\theta_1, \ldots, \theta_8$.

The system (4) defines (T, S), i.e., it specifies the evaluation of (T, S) at any input z_0. Of course, x_1, x_2, and Sz_0 need not be solutions to the monotone inclusion problem, since the input z_0 is arbitrary. To summarize our progress, we have shown that any frugal resolvent-splitting without lifting is equivalent to a frugal resolvent-splitting of the form (4).

We now take a moment to consider what happens with DRS under this setup. Although this discussion is not part of the proof, it will provide us with a sense of direction. Under this formulation, DRS has the form

$$0 = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -2 & 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & \theta & 0 & -\theta & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \alpha & 0 \\ 0 & 0 & -1+\eta & 0 & -\eta & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}.$$

Row 1 defines $z_1 = z_0$ as the input to $J_{\alpha A}$. Row 4 corresponds to $x_1 = J_{\alpha A} z_1$. Row 2 defines $z_2 = 2x_1 - z_0$ as the input to $J_{\alpha B}$. Row 5 corresponds to $x_2 = J_{\alpha B} z_2$. Row 3 defines $Tz_0 = z_0 + \theta(x_2 - x_1)$. Row 6 defines $Sz_0 = \eta x_2 + (1-\eta)x_1$. This linear system represents the evaluation of (T, S) at any arbitrary input z_0, i.e., the system defines (T, S).

To show that DRS is a fixed-point encoding, one considers evaluations of (T, S) at fixed points and shows $x_1 = x_2 = Sz_0$ and $\tilde{A}x_1 + \tilde{B}x_2 = 0$. To do this, we add a row representing the fixed-point condition $z_0 = Tz_0$

$$0 = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -2 & 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & \theta & 0 & -\theta & 1 & 0 & 0 & 0 \\ \mathbf{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \alpha & 0 \\ 0 & 0 & -1+\eta & 0 & -\eta & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}.$$

Now the system represents evaluations of (T, S) at fixed points. We then left-multiply the system with

$$(1/\theta)\begin{bmatrix} 0 & 0 & 1 & -1 & 0 & 0 & 0 \end{bmatrix}$$

to get $x_1 = x_2$, left-multiply the system with

$$\begin{bmatrix} 0 & 0 & -\eta/\theta & \eta/\theta & 0 & 0 & 1 \end{bmatrix}$$

to get $Sz_0 = x_1$, and left-multiply the system with

$$(1/\alpha)\begin{bmatrix}1 & 1 & 1/\theta & -1/\theta & 1 & 1 & 0\end{bmatrix}$$

to get $0 = \tilde{A}x_1 + \tilde{B}x_2$.

With DRS, it is possible to perform Gaussian elimination with the linear equalities defining (T, S) and the fixed-point condition $Tz_0 = z_0$ to conclude $x_1 = x_2 = Sz_0$ and $\tilde{A}x_1 + \tilde{B}x_2 = 0$. With other fixed-point encodings, should we not be able to do the same? How else could $Tz_0 = z_0$ certify $Sz_0 \in \mathbf{zer}(A + B)$? This turns out to be true: we must be able to establish $x_1 = x_2$, $x_1 = Sz_0$, and $\tilde{A}x_1 + \tilde{B}x_2 = 0$ through a linear combination of the linear equalities as otherwise we can construct counter examples that contradict the assumption that (T, S) is a fixed-point encoding.

We now return to the proof. Consider (4) with the fixed-point condition $T(z_0) = z_0$ added

$$0 = \underbrace{\begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \theta_1 & 0 & \theta_2 & 1 & 0 & 0 & 0 & 0 & 0 \\ \theta_3 & 0 & \theta_4 & 0 & \theta_5 & 1 & 0 & 0 & 0 \\ \mathbf{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \beta & 0 \\ \theta_6 & 0 & \theta_7 & 0 & \theta_8 & 0 & 0 & 0 & 1 \end{bmatrix}}_{=M} \underbrace{\begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}}_{=v}. \tag{5}$$

System (5) represents evaluations of (T, S) at a fixed points.

We claim that the linear equalities (5) must imply $x_1 = x_2$, $Sz_0 = x_1$, and $\tilde{A}x_1 + \tilde{B}x_2 = 0$. We prove these three implications one-by-one by assuming otherwise and constructing counter examples.

Assume for contradiction that (5) does not imply the linear equality $x_1 = x_2$. By Lemma 1, this means there is a specific instance

$$v' = (z_0', z_1', x_1', z_2', x_2', T(z_0'), \tilde{A}x_1', \tilde{B}x_2', S(z_0')) \in \mathbb{R}^{9d}$$

such that $Mv' = 0$ but $x_1' \neq x_2'$. The vector v' represents an evaluation of $(T(A, B, \cdot), S(A, B, \cdot))$ for any $A, B \in \mathcal{M}(\mathbb{R}^d)$ satisfying

$$\tilde{A}x_1' \in Ax_1', \quad \tilde{B}x_2' \in Bx_2'.$$

The evaluation is at a fixed point, i.e., $T(A, B, z_0') = z_0'$, since we enforced $T(z_0) = z_0$ in (5). Define

$$A(x) = x - x_1' + \tilde{A}x_1', \qquad B(x) = x - x_2' + \tilde{B}x_2'.$$

A and B are monotone operators constructed to match the evaluations $A(x_1') = \tilde{A}x_1'$ and $B(x_2') = \tilde{B}x_2'$. Write $x^\star = S(A, B, z_0')$. Since (T, S) is a fixed-point encoding, we have

$$0 = (A + B)x^\star.$$

However, $x_1' \neq x_2'$, so either $x_1' \neq x^\star$ or $x_2' \neq x^\star$ or both. Without loss of generality assume $x_1' \neq x^\star$. Loosely speaking, $x_1 \neq x^\star$ means (T, S) was able to identify that $x^\star$ is a solution without examining the output of A at $x^\star$, the purported solution. Since the evaluation of $(T(A, B, z_0'), S(A, B, z_0'))$ depends on A only through Ax_1', what prevents us from changing the operator value at $x^\star$? Define

$$C(x) = 2(x - x_1') + \tilde{A}x_1'.$$

Since $C(x_1') = \tilde{A}x_1'$, we still have $T(C, B, z_0') = z_0'$ and $S(C, B, z_0') = x^\star$, i.e., changing A to C does not affect the evaluation of T and S at z_0'. However,

$$0 \notin (C + B)x^\star,$$

since $C(x^\star) \neq A(x^\star)$. In other words, $T(C, B, z_0') = z_0'$, but $S(C, B, z_0')$ is not a zero of $C + B$. So (T, S) fails to be a fixed-point encoding for $C, B \in \mathcal{M}(\mathbb{R}^d)$, and we have a contradiction. This proves that the linear system of equalities (5) does imply the linear equality $x_1 = x_2$.

Next, assume for contradiction that (5) does not imply the linear equality $Sz_0 = x_1$. By Lemma 1, this means there is a specific instance

$$v' = (z_0', z_1', x_1', z_2', x_2', T(z_0'), \tilde{A}x_1', \tilde{B}x_2', S(z_0')) \in \mathbb{R}^{9d}$$

such that $Mv' = 0$ but $S(z_0') \neq x_1' = x_2'$. (We now know that $x_1' = x_2'$.) Again, loosely speaking, $S(z_0') \neq x_1'$ means (T, S) was able to identify that $S(z_0')$ is a solution without examining the output of A at $S(z_0')$, the purported solution, so we draw a contradiction by changing the operator value at $S(z_0')$. Using the same definition of A, B, and C, the same arguments carry over and we can establish $T(A, B, z_0') = T(C, B, z_0') = z_0'$ and $S(A, B, z_0') = S(C, B, z_0')$. Define $x^\star = S(A, B, z_0')$. Since we assumed (for contradiction) that $x^\star \neq x_1'$, we have

$$(A + B)(x^\star) \neq (C + B)(x^\star).$$

Remember that A, B, and C are single-valued. So it is not possible for both $0 = (A + B)(x^\star)$ and $0 = (C + B)(x^\star)$ to be true. Therefore (T, S) fails to be a fixed-point encoding for the instance $A, B \in \mathcal{M}(\mathbb{R}^d)$ or $C, B \in \mathcal{M}(\mathbb{R}^d)$, and we have a contradiction. This proves that the linear system of equalities (5) does imply the linear equality $Sz_0 = x_1$.

Finally, assume for contradiction that (5) does not imply the linear equality $\tilde{A}x_1 + \tilde{B}x_2 = 0$. By Lemma 1, this means there is a specific instance

$$v' = (z_0', z_1', x_1', z_2', x_2', T(z_0'), \tilde{A}x_1', \tilde{B}x_2', S(z_0')) \in \mathbb{R}^{9d}$$

such that $Mv' = 0$ but $\tilde{A}x_1' + \tilde{B}x_2' \neq 0$. Loosely speaking, $\tilde{A}x_1' + \tilde{B}x_2' \neq 0$ means (T, S) was able to identify that $S(z_0')$ is a solution without obtaining outputs of A and B that sum to 0, and we draw a contradiction by demonstrating that this is not possible when A and B are single-valued. We now know that $x_1' = x_2' = S(z_0')$. Define

$$A(x) = x - x_1' + \tilde{A}x_1', \qquad B(x) = x - x_2' + \tilde{B}x_2'.$$

Then $T(A, B, z_0) = z_0$, but

$$(A + B)(S(z_0')) = \tilde{A}x_1' + \tilde{B}x_2' \neq 0,$$

i.e., the purported solution $S(z_0')$ is not a solution. So (T, S) fails to be a fixed-point encoding for the instance $A, B \in \mathcal{M}(\mathbb{R}^d)$, and we have a contradiction. This proves that the linear system of equalities (5) does imply the linear equality $\tilde{A}x_1 + \tilde{B}x_2 = 0$.

With the assertions proved, we proceed to complete the proof. Gaussian elimination on (5) gives us the equivalent system

$$0 = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \theta_1 & 0 & \theta_2 & 1 & 0 & 0 & 0 & 0 & 0 \\ \boldsymbol{\theta_3 + 1} & 0 & \theta_4 & 0 & \theta_5 & \mathbf{0} & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \beta & 0 \\ \theta_6 & 0 & \theta_7 & 0 & \theta_8 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}. \tag{6}$$

Because the system of linear equalities must imply $x_1 = x_2$ and because of where the zeros and nonzeros are placed, we have $\theta_3 = -1$ and $\theta_4 = -\theta_5 = \theta$ for some $\theta \neq 0$. Let us further spell out this argument. The linear equality $x_1 = x_2$ can be expressed as

$$0 = \begin{bmatrix} 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}. \tag{7}$$

By Lemma 1, the system of linear equalities (6) implies (7) if and only if we can linearly combine the rows of (6) to get (7). Row 7 of (6) cannot be used in the linear combination, as any nonzero contribution from row 7 will place a nonzero component in the 9th column. Row 6 of (6) also cannot be used in the linear combination, as any nonzero contribution from row 6 will place a nonzero component in the 8th column. Repeating this argument tells us that rows 7, 6, 5, 4, 2, and 1 cannot be used in the linear combination. Therefore, a scalar multiple of row 3 of (6) must equal (7), and this tells us $\theta_3 = -1$ and $\theta_4 = -\theta_5 = \theta$ for some $\theta \neq 0$.

Plugging in the values of θ_3, θ_4 and θ_5, we get

$$0 = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \theta_1 & 0 & \theta_2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \theta & 0 & -\theta & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \beta & 0 \\ \theta_6 & 0 & \theta_7 & 0 & \theta_8 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}. \tag{8}$$

Because the linear equalities must imply $x_1 = Sz_0$ and because of where the zeros and nonzeros are placed, $\theta_6 = 0$, $\theta_7 = -1 + \eta$, and $\theta_8 = -\eta$ for some $\eta \in \mathbb{R}$. Let us further spell out this argument. The linear equality $Sz_0 = x_1$ can be expressed as

$$0 = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix} \tag{9}$$

By Lemma 1, the system of linear equalities (8) implies (9) if and only if we can linearly combine the rows of (8) to get (9). Row 6 cannot be used in the linear combination, as any nonzero contribution will place a nonzero component in the 8th column. Row 5 cannot be used in the linear combination, as any nonzero contribution will place a nonzero component in the 7th column. Repeating this argument tells us that rows 6, 5, 4, 2, and 1 cannot be used in the linear combination. This leaves us with the rows

$$0 = \begin{bmatrix} 0 & 0 & \theta & 0 & -\theta & 0 & 0 & 0 & 0 \\ \theta_6 & 0 & \theta_7 & 0 & \theta_8 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}$$

to imply (9). This is possible only if $\theta_6 = 0$, $\theta_7 = -1 + \eta$, and $\theta_8 = -\eta$ for some $\eta \in \mathbb{R}$.

Plugging in the values of θ_6, θ_7 and θ_8, we get

$$0 = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \theta_1 & 0 & \theta_2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \theta & 0 & -\theta & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \beta & 0 \\ 0 & 0 & -1+\eta & 0 & -\eta & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}. \tag{10}$$

Because the system of linear equalities must imply $0 = \tilde{A}x_1 + \tilde{B}x_2$ and because of where the zeros and nonzeros are placed, we have $\theta_1 = \beta/\alpha$ and $\theta_2 = -1 - \beta/\alpha$. Let us further spell out this argument. The linear equality $0 = \tilde{A}x_1 + \tilde{B}x_2$ can be expressed as

$$0 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}. \tag{11}$$

Left-multiply (10) by the invertible matrix

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ -(1+\theta_2) & 1 & 1/\theta & 0 & -(1+\theta_2) & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

to get

$$0 = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \theta_1 & 0 & \theta_2 & 1 & 0 & 0 & 0 & 0 & 0 \\ \mathbf{1+\theta_1+\theta_2} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{-(1+\theta_2)\alpha} & \boldsymbol{\beta} & \mathbf{0} \\ -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \beta & 0 \\ 0 & 0 & -1+\eta & 0 & -\eta & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}.$$

Left-multiply by the invertible matrix

$$\begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

to permute the rows and get

$$0 = \begin{bmatrix} 1+\theta_1+\theta_2 & 0 & 0 & 0 & 0 & 0 & -(1+\theta_2)\alpha & \beta & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & \alpha & 0 & 0 \\ \theta_1 & 0 & \theta_2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & \beta & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1+\eta & 0 & -\eta & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}. \tag{12}$$

By the lemma and the equivalence of (10) and (12), the system of linear equalities (10) implies (11) if and only if we can linearly combine the rows of (12) to get (11).

Row 7 cannot be used in the linear combination, as any nonzero contribution will place a nonzero component in the 9th column. Row 6 cannot be used in the linear combination, as any nonzero contribution will place a nonzero component in the 6th column. Repeating this argument tells us that rows 7, 6, 5, 4, 3, and 2 cannot be used in the linear combination. This leaves us with

$$0 = \begin{bmatrix} 1+\theta_1+\theta_2 & 0 & 0 & 0 & 0 & 0 & -(1+\theta_2)\alpha & \beta & 0 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \\ \tilde{A}x_1 \\ \tilde{B}x_2 \\ Sz_0 \end{bmatrix}$$

to imply (11) and this requires $\theta_1 = \beta/\alpha$ and $\theta_2 = -1 - \beta/\alpha$.

Finally, plugging in the parameters and expressing the splitting in functional form, we get the splitting of Theorem 1. ◻

2.4 Proof of Theorem 2

When $\alpha = \beta$, the splitting (T, S) of Theorem 1 reduces to the setup of DRS. The fixed-point iteration with respect to the DRS operator converges for all maximal monotone A and B if and only if $\theta \in (0, 2)$. That DRS converges for $\theta \in (0, 2)$ is well known [3, §26.3], and that DRS may diverge for some maximal monotone operators when $\theta \notin (0, 2)$ can be verified by considering the operators $A = 0$ and $B = N_{\{0\}}$, where $N_{\{0\}}$ denotes the normal cone operator with respect to the set $\{0\}$.

Now assume $\alpha \neq \beta$. We provide counter examples, single-valued maximal monotone operators A and B such that $\{0\} = \mathbf{zer}(A+B)$ and $T^k z^0$ diverges for any $z^0 \neq 0$. Note that the parameters α and β are fixed and are provided by the splitting. Our counter examples rely on α and β.

For the moment, consider the case $d = 2$. Consider the problem

$$\underset{x\in\mathbb{R}^2}{\text{find}} \quad 0 = (A+B)x,$$

where

$$A = \begin{bmatrix} 0 & \tan(\omega)/\alpha \\ -\tan(\omega)/\alpha & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 0 & -\tan(\omega)/\beta \\ \tan(\omega)/\beta & 0 \end{bmatrix}$$

and $\alpha, \beta > 0$, and $\omega \in (0, \pi/2)$. We identify A and B as maximal monotone operators from $\mathbb{R}^2 \to \mathbb{R}^2$. Note that $x^\star = 0$ is the unique solution.

With basic algebra, we can show that

$$Tz = \begin{bmatrix} 1 & (\theta/2)(1-\beta/\alpha)\cos(\omega)\sin(\omega) \\ -(\theta/2)(1-\beta/\alpha)\cos(\omega)\sin(\omega) & 1 \end{bmatrix} z.$$

With basic eigenvalue computation, we get

$$|\lambda_1|^2 = |\lambda_2|^2 = 1 + ((\theta/2)(1-\beta/\alpha)\cos(\omega)\sin(\omega))^2 > 1,$$

where λ_1, λ_2 are the eigenvalues of the matrix that defines T. So if $z^0 \neq 0$, the iteration $z^{k+1} = Tz^k$ diverges in that $\|z^k\| \to \infty$ and $\|Sz^k\| \to \infty$.

When $d > 2$, we arrive at the same conclusion with

$$\begin{bmatrix} A & 0 & 0 & \cdots & 0 \\ 0 & 0 & \ddots & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & & 0 \end{bmatrix} \in \mathbb{R}^{d \times d}, \qquad \begin{bmatrix} B & 0 & 0 & \cdots & 0 \\ 0 & 0 & \ddots & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & & 0 \end{bmatrix} \in \mathbb{R}^{d \times d},$$

which is the same counter example embedded into d dimensions. □

3 Impossibility of 3 operator resolvent-splitting without lifting

Define the problem class (3op-$\mathbb{R}^d$) to be the collection of monotone inclusion problems of the form

$$\underset{x \in \mathbb{R}^d}{\text{find}} \quad 0 \in (A + B + C)x \tag{3op-$\mathbb{R}^d$}$$

with $A, B, C \in \mathcal{M}(\mathbb{R}^d)$. A pair of functions (T, S) is a fixed-point encoding for the problem class (3op-$\mathbb{R}^d$) if

$$\exists z^\star \in \mathbb{R}^{d'} \text{ such that } \begin{pmatrix} T(A, B, C, z^\star) = z^\star \\ S(A, B, C, z^\star) \;= x^\star \end{pmatrix} \quad \Leftrightarrow \quad 0 \in (A + B + C)(x^\star).$$

We call

$$T : \mathcal{M}(\mathbb{R}^d) \times \mathcal{M}(\mathbb{R}^d) \times \mathcal{M}(\mathbb{R}^d) \times \mathbb{R}^{d'} \to \mathbb{R}^{d'}$$

the fixed-point mapping and

$$S : \mathcal{M}(\mathbb{R}^d) \times \mathcal{M}(\mathbb{R}^d) \times \mathcal{M}(\mathbb{R}^d) \times \mathbb{R}^{d'} \to \mathbb{R}^d,$$

the solution mapping. The four key terms, resolvent-splitting, frugal, unconditional convergence, and no lifting, are defined analogously.

To define the notion of resolvent-splitting without lifting for the problem class (3op-$\mathbb{R}^d$), we define the class of mappings $\mathcal{G}$ similarly to how we defined $\mathcal{F}$. Let I be the "identity mapping" defined as $I : \mathcal{M}(\mathbb{R}^d) \times \mathcal{M}(\mathbb{R}^d) \times \mathcal{M}(\mathbb{R}^d) \times \mathbb{R}^d \to \mathbb{R}^d$ and $I(A, B, C, z) = z$ for any $A, B, C \in \mathcal{M}(\mathbb{R}^d)$ and $z \in \mathbb{R}^d$. Let $J_{\alpha,1}$ be the resolvent with respect to the first operator defined as $J_{\alpha,1} : \mathcal{M}(\mathbb{R}^d) \times \mathcal{M}(\mathbb{R}^d) \times \mathcal{M}(\mathbb{R}^d) \times \mathbb{R}^d \to \mathbb{R}^d$ and $J_{\alpha,1}(A, B, C, z) = J_{\alpha A}(z)$ for any $A, B, C \in \mathcal{M}(\mathbb{R}^d)$ and $z \in \mathbb{R}^d$. Define $J_{\beta,2}$ likewise with $J_{\beta,2}(A, B, C, z) = J_{\beta B}(z)$ and $J_{\gamma,3}$ likewise as $J_{\gamma,3}(A, B, C, z) = J_{\gamma C}(z)$. Let

$$\mathcal{G}_0 = \{I\} \cup \{J_{\alpha,1} \,|\, \alpha > 0\} \cup \{J_{\beta,2} \,|\, \beta > 0\} \cup \{J_{\gamma,3} \,|\, \gamma > 0\}.$$

Recursively define

$$\mathcal{G}_{i+1} = \{F + G \,|\, F, G \in \mathcal{G}_i\} \cup \{F \circ G \,|\, F, G \in \mathcal{G}_i\} \cup \{\gamma F \,|\, F \in \mathcal{G}_i,\ \gamma \in \mathbb{R}\}$$

for $i = 0, 1, 2, \ldots,$ where "composition" $F \circ G$ is defined analogously. Finally, define

$$\mathcal{G} = \bigcup_{i=0}^{\infty} \mathcal{G}_i.$$

Elements of $\mathcal{G}$ map $\mathcal{M}(\mathbb{R}^d) \times \mathcal{M}(\mathbb{R}^d) \times \mathcal{M}(\mathbb{R}^d) \times \mathbb{R}^d$ to $\mathbb{R}^d$. If $R \in \mathcal{F}$ and $A, B, C \in \mathcal{M}(\mathbb{R}^d)$, then $R(A, B, C, \cdot) : \mathbb{R}^d \to \mathbb{R}^d$. If (T, S) is a fixed-point encoding for the problem class (3op-$\mathbb{R}^d$) and $T, S \in \mathcal{G}$, then (T, S) is a resolvent-splitting without lifting for the problem class (3op-$\mathbb{R}^d$).

Frugality is defined analogously with the notion of evaluation procedures. We only use the notion of frugality informally for the problem class (3op-$\mathbb{R}^d$).

Unconditional convergence is also defined analogously. We say (T, S) converges unconditionally for the problem class (3op-$\mathbb{R}^d$) if

$$T^k z^0 \to z^\star, \quad S z^\star \in \mathbf{zer}(A + B + C)$$

for any $z^0 \in \mathbb{R}^d$ and $A, B, C \in \mathcal{M}(\mathbb{R}^d)$, when $\mathbf{zer}(A + B + C) \neq \emptyset$.

3.1 Impossibility result

If one could find a frugal, unconditionally convergent resolvent-splitting without lifting for (3op-$\mathbb{R}^d$), it would be a satisfying generalization of DRS to 3 operators. However, this is impossible. Even if we drop frugality and convergence as requirements, this is impossible.

Theorem 3 *There is no resolvent-splitting* ***without lifting*** *for* (3op-$\mathbb{R}^d$).

Clarification. Assume $T(A, B, C, \cdot) : \mathbb{R}^d \to \mathbb{R}^d$ and $S(A, B, C, \cdot) : \mathbb{R}^d \to \mathbb{R}^d$ are constructed with finitely many resolvents,

$$
\begin{gathered}
J_{\alpha(1)A}, J_{\alpha(2)A}, \ldots, J_{\alpha(n_A)A} \\
J_{\beta(1)B}, J_{\beta(2)B}, \ldots, J_{\beta(n_B)B} \\
J_{\gamma(1)C}, J_{\gamma(2)C}, \ldots, J_{\gamma(n_C)C}
\end{gathered}
$$

where the parameters $\alpha(i), \beta(j), \gamma(k)$ may be different. Theorem 3 states that (T, S) fails to be a fixed-point encoding.

Clarification. Another way to state Theorem 3 is to say that no element of the near-ring $\mathcal{G}$ is a fixed-point encoding for (3op-$\mathbb{R}^d$).

3.2 Proof of Theorem 3

3.2.1 Outline

The proof can be divided into in roughly three steps. In the first step, we set up the notation and express the evaluation of T with a set of linear and non-linear equalities. In the second step, we show that the linear equalities, coupled with the fixed point condition and some additional assumptions, cannot show that the three operators are evaluated at a same single point. In the third step, we use the conclusion of the second step and a Farkas-type lemma to take a certain element from the null space of the linear system and use it to construct a counter example.

3.2.2 Proof

Assume for contradiction that (T, S) is a resolvent-splitting without lifting. Let n be the total number of resolvent evaluations required to compute T and S. The specific value of n depends on how you count, i.e., whether you simplify things and whether some resolvent evaluations are counted redundantly. All that matters is that n is finite.

Since $T, S \in \mathcal{G}$ there is a finite evaluation procedure for (T, S), and we can find a sequential ordering for the resolvent evaluations. Using this ordering, we label the resolvents $J_1, J_2, \ldots, J_n$, where J_i is one of $J_{\alpha A}$, $J_{\beta B}$, or $J_{\gamma C}$ for some $\alpha > 0$, $\beta > 0$, or $\gamma > 0$ for each $i = 1, \ldots, n$. We call z_i the point at which J_i is evaluated and $x_i = J_i(z_i)$ for $i = 1, \ldots, n$. In the process of evaluating Tz_0 and Sz_0, we get $z_0, z_1, x_1, z_2, x_2, \ldots, z_n, x_n$, in this order. Since scalar multiplication and vector addition are the only operations allowed aside from resolvent evaluations, z_i is defined as a linear combination of $z_0, z_1, x_1, z_2, x_2, \ldots, z_{i-1}, x_{i-1}$ for each $i = 1, \ldots, n$, by nature of the ordering. Likewise, Tz_0 can be expressed as a linear combination of $z_0, z_1, x_1, z_2, x_2, \ldots, z_n, x_n$. Without loss of generality, assume $J_{\alpha A}$, $J_{\beta B}$, and $J_{\gamma C}$ are all used least once with some $\alpha > 0$, $\beta > 0$, and $\gamma > 0$. Otherwise, if, for example, $J_{\alpha A}$ is never used, we let $z_{n+1} = 0$ and $J_{n+1} = J_A$ to fix the issue. This is equivalent to evaluating the resolvent at the end and not using the output.

Say $J_{\alpha A}$, $J_{\beta B}$, and $J_{\gamma C}$ are evaluated n_A, n_B, and n_C times, respectively. So $n_A + n_B + n_C = n$. Let $a(1), a(2), \ldots, a(n_A) \in \{1, 2, \ldots, n\}$ be distinct indicies and let $\alpha(1), \alpha(2), \ldots, \alpha(n_A) > 0$ be parameters so that

$$x_{a(\ell)} = J_{a(\ell)}(z_{a(\ell)}) = J_{\alpha(\ell)A}(z_{a(\ell)}).$$

In other words, $x_{a(1)}, x_{a(2)}, \ldots, x_{a(n_A)}$ are the outputs of the resolvents of A. Likewise, let $b(1), b(2), \ldots, b(n_B) \in \{1, 2, \ldots, n\}$ and $c(1), c(2), \ldots, c(n_C) \in \{1, 2, \ldots, n\}$ be distinct indices and let $\beta(1), \beta(2), \ldots, \beta(n_B) > 0$ and $\gamma(1), \gamma(2), \ldots, \gamma(n_C) > 0$ be parameters so that

$$x_{b(\ell)} = J_{b(\ell)}(z_{b(\ell)}) = J_{\beta(\ell)B}(z_{b(\ell)})$$

for $\ell = 1, \ldots, n_B$ and

$$x_{c(\ell)} = J_{c(\ell)}(z_{c(\ell)}) = J_{\gamma(\ell)C}(z_{c(\ell)})$$

for $\ell = 1, \ldots, n_C$.

We express the evaluation of Tz_0 with the following system of linear and non-linear equalities:

$$0 = \begin{bmatrix} * & 1 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ * & * & * & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ * & * & * & * & * & 1 & \cdots & 0 & 0 & 0 \\ & & & \vdots & & & & & & \\ & & & \vdots & & & & & & \\ * & * & * & * & * & * & \cdots & 1 & 0 & 0 \\ * & * & * & * & * & * & \cdots & * & * & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ z_3 \\ \vdots \\ z_n \\ x_n \\ Tz_0 \end{bmatrix}$$

$$x_1 = J_1(z_1),\ x_2 = J_2(z_2),\ \ldots,\ x_n = J_n(z_n),$$

where the $*$ denote unspecified scalar coefficients. (Each scalar in the matrix should be interpreted as a $d \times d$ block. We have seen this notation in the proof of Theorem 1.) Each linear equality except the last one defines z_i for $i = 1, \ldots, n$. The last linear equality defines $T(z_0)$. With Gaussian elimination, we obtain the simpler equivalent system

$$0 = \begin{bmatrix} * & 1 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ * & \mathbf{0} & * & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ * & \mathbf{0} & * & \mathbf{0} & * & 1 & \cdots & 0 & 0 & 0 \\ & & & \vdots & & & & & & \\ & & & \vdots & & & & & & \\ * & \mathbf{0} & * & \mathbf{0} & * & \mathbf{0} & \cdots & 1 & 0 & 0 \\ * & \mathbf{0} & * & \mathbf{0} & * & \mathbf{0} & \cdots & \mathbf{0} & * & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ z_3 \\ \vdots \\ z_n \\ x_n \\ Tz_0 \end{bmatrix}$$

$$x_1 = J_1(z_1),\ x_2 = J_2(z_2),\ \ldots,\ x_n = J_n(z_n).$$

The **boldface symbols** denote where to pay attention in the linear systems. To summarize our progress, we have set up the notation and shown that these linear and non-linear equalities define the evaluation of T at any input z_0.

We now take a moment to consider what happens with DRS under a similar formulation. Although this discussion is not part of the proof, it will provide us with a sense of direction. Under this formulation, DRS has the form

$$0 = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & -2 & 1 & 0 & 0 \\ -1 & 0 & \theta & 0 & -\theta & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \end{bmatrix}$$

$$x_1 = J_{\alpha A}(z_1),\ x_2 = J_{\alpha B}(z_2).$$

With DRS we can combine the linear equalities with the fixed point condition $z_0 = Tz_0$ to show that $x_1 = x_2$ when the input z_0 is a fixed point. More specifically, we add $z_0 = Tz_0$ to the linear system

$$0 = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & -2 & 1 & 0 & 0 \\ \mathbf{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \\ -1 & 0 & \theta & 0 & -\theta & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \\ x_1 \\ z_2 \\ x_2 \\ Tz_0 \end{bmatrix}$$

and left-multiply

$$(1/\theta) \begin{bmatrix} 0 & 0 & -1 & 1 \end{bmatrix}$$

to get

$$0=\begin{bmatrix}0 & 0 & 1 & 0 & -1 & 0\end{bmatrix}\begin{bmatrix}z_0\\ z_1\\ x_1\\ z_2\\ x_2\\ Tz_0\end{bmatrix}.$$

This is equivalent to combining

$$\begin{aligned}Tz_0 &= z_0+\theta(x_2-x_1)\\ z_0 &= Tz_0\end{aligned}$$

to conclude $x_1 = x_2$ when the input z_0 is a fixed point. Remember, x_1 and x_2 are the points where A and B are (indirectly) evaluated. ($J_{\alpha A}$ is directly evaluated at z_1, so $z_1 \in x_1 + \alpha Ax_1$, and we indirectly obtain the output $(1/\alpha)(z_1 - x_1) \in Ax_1$. Likewise, we indirectly obtain the output $(1/\alpha)(z_2 - x_2) \in Bx_2$.)

These arguments show that with DRS, T evaluates A and B at the same point when the input z_0 is a fixed point. (Further arguments would establish that the same point is a solution by showing that the outputs of A and B sum to 0.) In general, given a fixed-point encoding (T, S) and a fixed point z_0, shouldn't the evaluation of T examine the output of all (2 or 3) operators at the solution Sz_0? Otherwise how could $z_0 = Tz_0$ certify that Sz_0 is a solution?

We now return to the setup of (3op-$\mathbb{R}^d$). Can we combine the linear equalities and the fixed-point condition $z_0 = Tz_0$ to show that A, B, and C are evaluated at a same single point? It turns out that we cannot. (This by itself is not a contradiction. Just because we can't show something with one approach doesn't mean it can't be shown.) However, this approach runs into a problem. If we proceed to construct a counter example to draw a contradiction, we run into certain difficulties. We need a modified approach.

Instead, assume the input z_0 furthermore satisfies the additional linear equalities

$$\begin{aligned}x_{a(1)} &= x_{a(2)} = \cdots = x_{a(n_A)}\\ x_{b(1)} &= x_{b(2)} = \cdots = x_{b(n_B)}\\ x_{c(1)} &= x_{c(2)} = \cdots = x_{c(n_C)}\end{aligned} \tag{13}$$

in addition to the fixed-point condition $z_0 = Tz_0$. Since (T, S) is a fixed point encoding, $z_0 = Tz_0$ should certify that Sz_0 is a solution, regardless of the additional assumptions (13). Now can we use the linear equalities defining T, $z_0 = Tz_0$, and (13) to show that $x_{a(1)} = x_{b(1)} = x_{c(1)}$? No we cannot. Let's see why.

We add the fixed-point condition $z_0 = Tz_0$ to the system of linear equalities and perform Gaussian elimination:

$$
0=\underbrace{\begin{bmatrix}
* & 1 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0\\
* & 0 & * & 1 & 0 & 0 & \cdots & 0 & 0 & 0\\
* & 0 & * & 0 & * & 1 & \cdots & 0 & 0 & 0\\
 & & & \vdots & & & & & & \\
 & & & \vdots & & & & & & \\
* & 0 & * & 0 & * & 0 & \cdots & 1 & 0 & 0\\
\mathbf{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{1}\\
\boldsymbol{*} & \mathbf{0} & \boldsymbol{*} & \mathbf{0} & \boldsymbol{*} & \mathbf{0} & \cdots & \mathbf{0} & \boldsymbol{*} & \mathbf{0}
\end{bmatrix}}_{=M}
\underbrace{\begin{bmatrix} z_0\\ z_1\\ x_1\\ z_2\\ x_2\\ z_3\\ \vdots\\ z_n\\ x_n\\ Tz_0\end{bmatrix}}_{=v}
$$

$$
x_1=J_1(z_1),\ x_2=J_2(z_2),\ \ldots,\ x_n=J_n(z_n). \tag{14}
$$

Define the last row of M to be m. Let N be a matrix such that

$$
0=N\begin{bmatrix} z_0\\ z_1\\ x_1\\ \vdots\\ z_n\\ x_n\\ Tz_0\end{bmatrix} \quad \Leftrightarrow \quad (13).
$$

More specifically, let $N\in\mathbb{R}^{(n-3)d\times(2n+2)d}$ contain only 0, 1, and -1 and let the nonzeros only be on the columns corresponding to the x-variables (columns number $3, 5, \ldots, 2n+1$). Note that (13) represents $n-3$ linear equalities, and this is reflected as the number of rows in N. The positions of the nonzeros in N depend on the ordering of the resolvent evaluations, and N is not unique. Let

$$
L=\begin{bmatrix} M\\ N\end{bmatrix}, \qquad \tilde{N}=\begin{bmatrix} m\\ N\end{bmatrix},
$$

where m is the last row of M. So $0=Lv$ means v satisfies the linear equalities (14) and (13). Let's try to combine rows of $0=Lv$ to establish $x_{a(1)}=x_{b(1)}=x_{c(1)}$. By Lemma 1, $0=Lv$ implies $x_{a(1)}=x_{b(1)}$ and $x_{b(1)}=x_{c(1)}$ if and only if we can linearly combine the rows of $0=Lv$ to get $x_{a(1)}=x_{b(1)}$ and $x_{b(1)}=x_{c(1)}$.

Every row of M except the last one cannot be used in the linear combination to prove a linear equality only involving the x-variables, as any nonzero contribution will place a nonzero component on a column corresponding to z_i or Tz_0 (column number $1, 2, 4, 6\ldots, 2n, 2n+2$). This leaves us with the rows of

$$
0=\begin{bmatrix} m\\ N\end{bmatrix} v
$$

to show $x_{a(1)} = x_{b(1)}$ and $x_{b(1)} = x_{c(1)}$. The linear equality $0 = Nv$ enforces

$$\begin{aligned} x_{a(1)} &= x_{a(2)} = \cdots = x_{a(n_A)} \\ x_{b(1)} &= x_{b(2)} = \cdots = x_{b(n_B)} \\ x_{c(1)} &= x_{c(2)} = \cdots = x_{c(n_C)}, \end{aligned}$$

which are $n-3$ equalities. Therefore, $\mathcal{N}(N)$, the nullspace of N, has codimension $(n-3)d$. The linear equality $0 = mv$ can establish $x_{a(1)} = x_{b(1)}$ or $x_{b(1)} = x_{c(1)}$, but not both. If $0 = \tilde{N}v$ implies both $x_{a(1)} = x_{b(1)}$ and $x_{b(1)} = x_{c(1)}$, then in total $0 = \tilde{N}v$ enforces $x_1 = \cdots = x_n$, which are $n-1$ equalities. So $\mathcal{N}(\tilde{N})$, the nullspace of $\tilde{N}$, would have codimension $(n-1)d$ or less, but this reduction in codimension by $2d$ is a contradiction since m is just one row.

Therefore, the linear system $0 = Lv$ does not imply both $x_{a(1)} = x_{b(1)}$ and $x_{b(1)} = x_{c(1)}$. This by itself is not a contradiction. Rather, we use this fact to construct a counter example, $A, B, C \in \mathcal{M}(\mathbb{R}^d)$ such that (T, S) fails to be a fixed-point encoding. The additional assumption (13) will help us in this construction.

We construct the counter example for the case $d = 1$. When $d > 1$, we can use the same 1 dimensional construction repeated for the d coordinates. More specifically, if $A \in \mathcal{M}(\mathbb{R})$, then $\tilde{A}$ defined with

$$\tilde{A}(x) = (A(x_1), A(x_2), \ldots, A(x_d)),$$

where $x = (x_1, x_2, \ldots, x_d) \in \mathbb{R}^d$, satisfies $\tilde{A} \in \mathcal{M}(\mathbb{R}^d)$. This sort of construction based on the 1 dimensional counter example will provide a d dimensional counter example.

Assume $d = 1$. By Lemma 1, there is a specific instance

$$v' = (z'_0, z'_1, x'_1, z'_2, x'_2, \ldots, z'_n, x'_n, T(z'_0)) \in \mathbb{R}^{2n+2}$$

that satisfies $Mv' = 0$ and the linear equalities of (13), but $x'_{a(1)} \neq x'_{b(1)}$ or $x'_{b(1)} \neq x'_{c(1)}$ or both. Without loss of generality, say $x'_{b(1)} \neq x'_{c(1)}$.

Define A such that

$$J_{\alpha(i)A}(z'_{a(i)}) = x'_{a(1)}$$

for all $i = 1, \ldots, n_A$. In particular, we achieve this by defining

$$A(x'_{a(1)}) = \left[\min_{i=1,\ldots,n_A} \left(z'_{a(i)} - x'_{a(1)} \right) / \alpha(i), \max_{i=1,\ldots,n_A} \left(z'_{a(i)} - x'_{a(1)} \right) / \alpha(i) \right].$$

For the moment, leave $A(x)$ for $x \neq x'_{a(1)}$ unspecified. Define $B(x'_{b(1)})$ and $C(x'_{c(1)})$ likewise. By construction, $z'_0 = T(A, B, C, z'_0)$, even though A, B, and C are not yet fully specified. Write $x' = S(z'_0)$. We have $x' \neq x'_{b(1)}$ or $x' \neq x'_{c(1)}$ since $x'_{b(1)} \neq x'_{c(1)}$. Without loss of generality, let $x' \neq x'_{c(1)}$.

Now we define

$$A(x) = \begin{cases} (x - x'_{a(1)}) + \min\{A(x'_{a(1)})\} & \text{for } x < x'_{a(1)} \\ (x - x'_{a(1)}) + \max\{A(x'_{a(1)})\} & \text{for } x > x'_{a(1)} \end{cases}$$

and

$$B(x) = \begin{cases} (x - x'_{b(1)}) + \min\{A(x'_{b(1)})\} & \text{for } x < x'_{b(1)} \\ (x - x'_{b(1)}) + \max\{A(x'_{b(1)})\} & \text{for } x > x'_{b(1)}. \end{cases}$$

(This makes A and B maximal monotone.) By construction, $(A + B)(x')$ is a bounded subset of $\mathbb{R}$, and $C(x')$ is unspecified. Depending on whether $x' < x'_{c(1)}$ or $x' > x'_{c(1)}$, we can make $C(x')$ an arbitrarily small or large value, respectively (and still have C be monotone). In either case, we make $C(x')$ single-valued and so small or so large that $0 \notin (A+B+C)(x')$. We extend the definition of C to all of $\mathbb{R}$ to make it maximal monotone.

So we have maximal monotone operators A, B, and C, such that $z'_0 = T(A, B, Cz'_0)$ but the $x' = S(z'_0)$ does not satisfy $0 \in (A+B+C)x'$. This contradicts the assumption that (T, S) is a fixed-point encoding. □

4 Attainment of 3 operator resolvent-splitting with minimal lifting

Loosely speaking, we say $(\boldsymbol{T}, S)$ is a a resolvent-splitting with *ℓ-fold lifting* for the problem class (3op-$\mathbb{R}^d$) if $(\boldsymbol{T}, S)$ is a fixed-point encoding and

$$\boldsymbol{T}(A, B, C, \cdot) : \mathbb{R}^{\ell d} \to \mathbb{R}^{\ell d}, \qquad S(A, B, C, \cdot) : \mathbb{R}^{\ell d} \to \mathbb{R}^d$$

is constructed with scalar multiplication, vector addition, and resolvent evaluations. Note that onefold lifting corresponds to no lifting. Frugality is defined analogously. We define these terms informally since they are not used in a rigorous statement. Theorem 3 states a resolvent-splitting for (3op-$\mathbb{R}^d$) requires lifting. Then how much? The answer is twofold lifting.

A standard trick to solve (3op-$\mathbb{R}^d$) is to "copy" variables and form an enlarged problem

$$\underset{x_1, x_2, x_3 \in \mathbb{R}^d}{\text{find}} \quad 0 \in \begin{bmatrix} Ax_1 \\ Bx_2 \\ Cx_3 \end{bmatrix} + N_{\{(x_1, x_2, x_3) \,|\, x_1 = x_2 = x_3\}}(x_1, x_2, x_3),$$

where N_K is the normal cone operator with respect to the set K. By applying DRS in an appropriately scaled space, we get the parallel proximal algorithm (PPXA) [20,21], which generalizes Spingarn's method of partial inverse [51]. The PPXA splitting is given by $(\boldsymbol{T}, S)$

$$
\begin{aligned}
x_A &= J_{(\gamma/\omega_A)A}(z_A)\\
x_B &= J_{(\gamma/\omega_B)B}(z_B)\\
x_C &= J_{(\gamma/\omega_C)C}(z_C)\\
\bar{z} &= \omega_A z_A + \omega_B z_B + \omega_C z_C\\
\bar{x} &= \omega_A x_A + \omega_B x_B + \omega_C x_C\\
T_A(\boldsymbol{z}) &= z_A + \theta(2\bar{x} - \bar{z} - x_A)\\
T_B(\boldsymbol{z}) &= z_B + \theta(2\bar{x} - \bar{z} - x_B)\\
T_C(\boldsymbol{z}) &= z_C + \theta(2\bar{x} - \bar{z} - x_C)\\
S(\boldsymbol{z}) &= J_{\alpha_A A}(z_A),
\end{aligned}
\tag{PPXA}
$$

where $\omega_A, \omega_B, \omega_C > 0$ satisfy $\omega_A+\omega_B+\omega_C = 1$ and $\theta \in (0, 2)$. This frugal, unconditionally convergent resolvent-splitting uses threefold lifting, since $\boldsymbol{T} = (T_A, T_B, T_C) : \mathbb{R}^{3d} \to \mathbb{R}^{3d}$.

So constructing a resolvent-splitting for (3op-$\mathbb{R}^d$) is impossible with onefold lifting, but it is possible with threefold lifting. It turns out that twofold lifting is sufficient, and we therefore call twofold lifting the *minimal lifting* for (3op-$\mathbb{R}^d$).

4.1 Attainment result

Theorem 4 *The pair* $(\boldsymbol{T}, S)$*, where* $\boldsymbol{T} : \mathbb{R}^{2d} \to \mathbb{R}^{2d}$ *and* $S : \mathbb{R}^{2d} \to \mathbb{R}^d$*, defined as*

$$
\begin{aligned}
x_1 &= J_{\alpha A}(z_1)\\
x_2 &= J_{\alpha B}(x_1 + z_2)\\
x_3 &= J_{\alpha C}\,(x_1 - z_1 + x_2 - z_2)\\
T_1(\boldsymbol{z}) &= z_1 + \theta(x_3 - x_1)\\
T_2(\boldsymbol{z}) &= z_2 + \theta(x_3 - x_2)\\
S(\boldsymbol{z}) &= (1/3)(x_1 + x_2 + x_3)
\end{aligned}
$$

with $\boldsymbol{z} = (z_1, z_2)$ *and* $\boldsymbol{T} = (T_1, T_2)$*, is a fixed-point encoding, and* $(\boldsymbol{T}, S)$ *converges unconditionally for* $\theta \in (0, 1)$ *and* $\alpha > 0$.

Therefore, $(\boldsymbol{T}, S)$ for any $\theta \in (0, 1)$ is a frugal, unconditionally convergent, resolvent-splitting with minimal lifting for (3op-$\mathbb{R}^d$). When $B = 0$, the splitting of Theorem 4 reduces to DRS. In this sense, this splitting is a direct generalization of DRS with minimal lifting.

Remark To the best of the author's knowledge, the splitting of Theorem 4 cannot be reduced to an instance of a known splitting method. This is why Theorem 4 is proved from first principles.

4.2 Proof of Theorem 4

Throughout the proof, write $\boldsymbol{y} = (y_1, y_2)$ and $\boldsymbol{z} = (z_1, z_2)$. Without loss of generality, assume $\alpha = 1$. We first show that $(\boldsymbol{T}, S)$ is a fixed-point encoding.

Assume $\boldsymbol{z}$ is a fixed point of $\boldsymbol{T}$. Since $\boldsymbol{z}$ is a fixed point, we have $T_1(\boldsymbol{z}) = z_1$ and $T_2(\boldsymbol{z}) = z_2$, and this implies $x_1 = x_2 = x_3$. Write

$$\begin{aligned} a &= z_1 - x_1 \\ b &= x_1 + z_2 - x_2 \\ c &= x_1 - z_1 + x_2 - z_2 - x_3. \end{aligned}$$

Add the three and use $x_1 = x_3$ to get

$$a + b + c = 0.$$

Since $a \in Ax_1$, $b \in Bx_2$, and $c \in Cx_3$, by the definitions of x_1, x_2, and x_3, this proves $x_1 = x_2 = x_3$ is a solution to (3op-$\mathbb{R}^d$).

Now assume $x^\star$ is a solution to (3op-$\mathbb{R}^d$), and let $a \in Ax^\star$, $b \in Bx^\star$, and $c \in Cx^\star$ so that $a + b + c = 0$. We then define

$$\boldsymbol{z}^\star = (a + x^\star, b)$$

It is straightforward to verify that $\boldsymbol{T}(\boldsymbol{z}^\star) = \boldsymbol{z}^\star$ and $S(\boldsymbol{z}^\star) = x^\star$.

Next we show that $(\boldsymbol{T}, S)$ converges unconditionally for $\theta \in (0, 1)$. We show this by showing $\boldsymbol{T}$ is nonexpansive for $\theta = 1$, and appealing to the KM iteration theorem [3, Proposition 5.16], which states that an averaged nonexpansive iteration converges to a fixed point, if a fixed point exists.

Let $\theta = 1$. Define $M : \mathbb{R}^{2d} \to \mathbb{R}^{2d}$ with

$$M(x_1, x_2) = \begin{bmatrix} x_1 + x_2 \\ x_1 + x_2 \end{bmatrix},$$

which is the linear operator corresponding to the matrix

$$\begin{bmatrix} I & I \\ I & I \end{bmatrix} \in \mathbb{R}^{2d \times 2d}.$$

Define $\boldsymbol{U}$ as

$$\boldsymbol{U}(\boldsymbol{z}) = \begin{bmatrix} J_A(z_1) - z_1 \\ J_B(z_2 + x_1) - z_2 \end{bmatrix} \in \mathbb{R}^{2d},$$

and we can write

$$\boldsymbol{T} = -\boldsymbol{U} + \begin{bmatrix} J_C \\ J_C \end{bmatrix} \circ M \circ \boldsymbol{U}.$$

To clarify, $\circ$ here denotes the composition of operators from $\mathbb{R}^{2d}$ to $\mathbb{R}^{2d}$. Define

$$N = \begin{bmatrix} -I & I \\ I & -I \end{bmatrix} \in \mathbb{R}^{2d\times 2d}.$$

Then

$$\begin{aligned}
&\|\boldsymbol{T}(\boldsymbol{y}) - \boldsymbol{T}(\boldsymbol{z})\|^2 \\
&= \|\boldsymbol{U}(\boldsymbol{y}) - \boldsymbol{U}(\boldsymbol{z})\|^2 + \left\| \begin{bmatrix} J_C \\ J_C \end{bmatrix} \circ M \circ \boldsymbol{U}(\boldsymbol{y}) - \begin{bmatrix} J_C \\ J_C \end{bmatrix} \circ M \circ \boldsymbol{U}(\boldsymbol{z}) \right\|^2 \\
&\quad - 2\left\langle \boldsymbol{U}(\boldsymbol{y}) - \boldsymbol{U}(\boldsymbol{z}), \begin{bmatrix} J_C \\ J_C \end{bmatrix} \circ M \circ \boldsymbol{U}(\boldsymbol{y}) - \begin{bmatrix} J_C \\ J_C \end{bmatrix} \circ M \circ \boldsymbol{U}(\boldsymbol{z}) \right\rangle \\
&\le \|\boldsymbol{U}(\boldsymbol{y}) - \boldsymbol{U}(\boldsymbol{z})\|^2 \\
&\quad + \left\langle M \circ (\boldsymbol{U}(\boldsymbol{y}) - \boldsymbol{U}(\boldsymbol{z})), \begin{bmatrix} J_C \\ J_C \end{bmatrix} \circ M \circ \boldsymbol{U}(\boldsymbol{y}) - \begin{bmatrix} J_C \\ J_C \end{bmatrix} \circ M \circ \boldsymbol{U}(\boldsymbol{z}) \right\rangle \\
&\quad - 2\left\langle \boldsymbol{U}(\boldsymbol{y}) - \boldsymbol{U}(\boldsymbol{z}), \begin{bmatrix} J_C \\ J_C \end{bmatrix} \circ M \circ \boldsymbol{U}(\boldsymbol{y}) - \begin{bmatrix} J_C \\ J_C \end{bmatrix} \circ M \circ \boldsymbol{U}(\boldsymbol{z}) \right\rangle \\
&= \|\boldsymbol{U}(\boldsymbol{y}) - \boldsymbol{U}(\boldsymbol{z})\|^2 \\
&\quad + (\boldsymbol{U}(\boldsymbol{y}) - \boldsymbol{U}(\boldsymbol{z}))^T N \left(\begin{bmatrix} J_C \\ J_C \end{bmatrix} \circ M \circ \boldsymbol{U}(\boldsymbol{y}) - \begin{bmatrix} J_C \\ J_C \end{bmatrix} \circ M \circ \boldsymbol{U}(\boldsymbol{z}) \right) \\
&= \|\boldsymbol{U}(\boldsymbol{y}) - \boldsymbol{U}(\boldsymbol{z})\|^2.
\end{aligned}$$

The first line follows from simply plugging in the expression for $\boldsymbol{T}$ and expanding the squares. The second line, the inequality, follows from firm nonexpansiveness of J_C. The third line follows from the reasoning

$$v^T \begin{bmatrix} I & I \\ I & I \end{bmatrix} u - 2v^T u = v^T \begin{bmatrix} -I & I \\ I & -I \end{bmatrix} u = v^T N u$$

for any $v, u \in \mathbb{R}^{2d}$. The last line follows from recognizing that the second term is 0 with the reasoning

$$\begin{bmatrix} a \\ b \end{bmatrix}^T N \begin{bmatrix} c \\ c \end{bmatrix} = \begin{bmatrix} a \\ b \end{bmatrix}^T \begin{bmatrix} -c + c \\ c - c \end{bmatrix} = 0$$

for any $a, b, c \in \mathbb{R}^d$. Next we have

$$\begin{aligned}
&\|\boldsymbol{U}(\boldsymbol{y}) - \boldsymbol{U}(\boldsymbol{z})\|^2 \\
&= \|\boldsymbol{y} - \boldsymbol{z}\|^2 + \left\| \begin{bmatrix} J_A(y_1) - J_A(z_1) \\ J_B(y_2 + J_A(y_1)) - J_B(z_2 + J_A(z_1)) \end{bmatrix} \right\|^2 \\
&\quad - 2\left\langle \boldsymbol{y} - \boldsymbol{z}, \begin{bmatrix} J_A(y_1) - J_A(z_1) \\ J_B(y_2 + J_A(y_1)) - J_B(z_2 + J_A(z_1)) \end{bmatrix} \right\rangle
\end{aligned}$$

$$
\begin{aligned}
&= \|\boldsymbol{y}-\boldsymbol{z}\|^2 - \|J_A(y_1)-J_A(z_1)\|^2 - \|J_B(y_2+J_A(y_1)) - J_B(z_2+J_A(z_1))\|^2 \\
&\quad - 2\left(\langle y_1 - z_1, J_A(y_1)-J_A(z_1)\rangle - \|J_A(y_1)-J_A(z_1)\|^2\right) \\
&\quad - 2\,\langle y_2 - z_2, J_B(y_2+J_A(y_1)) - J_B(z_2+J_A(z_1))\rangle \\
&\quad + 2\|J_B(y_2+J_A(y_1)) - J_B(z_2+J_A(z_1))\|^2 \\
&\leq \|\boldsymbol{y}-\boldsymbol{z}\|^2 - \|J_A(y_1)-J_A(z_1)\|^2 - \|J_B(y_2+J_A(y_1)) - J_B(z_2+J_A(z_1))\|^2 \\
&\quad - 2\,\langle y_2 - z_2, J_B(y_2+J_A(y_1)) - J_B(z_2+J_A(z_1))\rangle \\
&\quad + 2\,\langle y_2 + J_A(y_1) - z_2 - J_A(z_1), J_B(y_2+J_A(y_1)) - J_B(z_2+J_A(z_1))\rangle \\
&= \|\boldsymbol{y}-\boldsymbol{z}\|^2 - \|J_A(y_1)-J_A(z_1)\|^2 - \|J_B(y_2+J_A(y_1)) - J_B(z_2+J_A(z_1))\|^2 \\
&\quad + 2\,\langle J_A(y_1)-J_A(z_1), J_B(y_2+J_A(y_1)) - J_B(z_2+J_A(z_1))\rangle \\
&= \|\boldsymbol{y}-\boldsymbol{z}\|^2 - \|J_A(y_1)-J_A(z_1) + J_B(y_2+J_A(y_1)) - J_B(z_2+J_A(z_1))\|^2 \\
&\leq \|\boldsymbol{y}-\boldsymbol{z}\|^2.
\end{aligned}
$$

The first line follows from plugging in the definition of $\boldsymbol{U}$ and expanding the squares. The second line follows from separating the norm and inner product on $\mathbb{R}^{2d}$ to separate norms and inner products on $\mathbb{R}^d$. The third line, the inequality, follows from applying the firm nonexpansiveness inequality twice, once for J_A and once for J_B. The fourth line follows from combining the two inner products. The fifth line follows from completing the square. The final inequality follows from droipping the negative sum of square. $\square$

4.3 Numerical examples

Whether the splitting of Theorem 4 is fast or efficient is somewhat beside the point. The purpose of Theorem 4 is to establish attainment of minimal lifting, and it says nothing about the rate of convergence.

Nevertheless, we present some experiments with the splitting of Theorem 4 in this section. These experiments are meant to be merely illustrative, and whether the splitting of Theorem 4 has any advantage over existing methods such as PPXA and whether the notion of minimal lifting translates to any practical performance advantage is a question to be addressed in future work.

Signal denoising with outliers. Consider the problem

$$
\begin{array}{ll}
\underset{x\in\mathbb{R}^d}{\text{minimize}} & \|x_S - a\|_1 + \lambda\|Ux - b\|_1 \\
\text{subject to} & x \geq 0,
\end{array}
$$

where $S \subseteq \{1, 2, \ldots, d\}$, $a \in \mathbb{R}^{|S|}$, $b \in \mathbb{R}^d$, and $U \in \mathbb{R}^{d\times d}$ is a unitary matrix representing a wavelet transform. The statistical interpretation is that we noisily observe x on a subset S of its indices, noisily observe x in the wavelet domain, and have a priori knowledge that x is nonnegative. The ℓ^1-norm is used for robustness against outliers. We reformulate this problem as

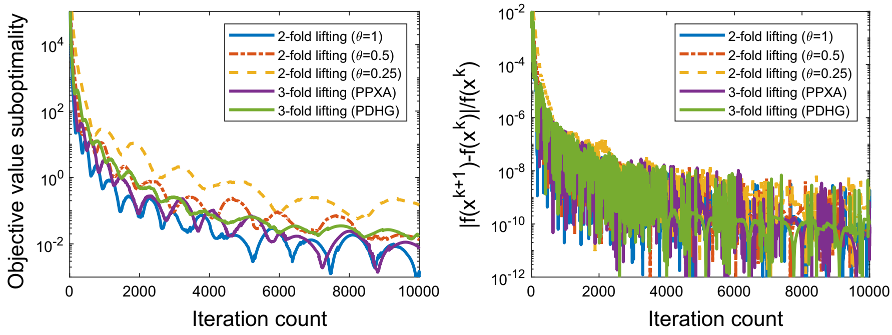

Fig. 1 Objective value and $|f(x^{k+1}) - f(x^k)|/f(x^k)$ versus iterations for the denoising problem

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \underbrace{\|x_S - a\|_1}_{=f(x)} + \underbrace{\lambda\|Ux - b\|_1}_{=g(x)} + \underbrace{\delta_{\mathbb{R}^d_+}(x)}_{=h(x)}$$

and apply the splitting of Theorem 4, PPXA, and PDHG with $A = \partial f$, $B = \partial g$, and $C = \partial h$. Because U is unitary, $J_{\alpha\partial g}$ has a closed-form formula. For the experiments, we used synthetic data with $d = 2^{20}$ and $|S| \approx d/5$. The code for data generation and optimization is provided on the author's website for scientific reproducibility.

Figure 1 shows the results. The splitting of Theorem 4, which uses twofold lifting, is competitive with PPXA and PDHG, which use threefold lifting. For all methods, the parameters were roughly tuned for best performance. We do not plot distance to solution, since solution does not seem to be unique.

Portfolio optimization. Consider the Markowitz portfolio optimization [11] problem

$$\begin{array}{ll}\underset{x\in\mathbb{R}^d}{\text{minimize}} & (1/2)\sum_{i=1}^n (a_i^T x - b)^2 \\ \text{subject to} & x \in \Delta \\ & \mu^T x \geq b,\end{array}$$

where d is the number of assets, $a_1, \ldots, a_n \in \mathbb{R}^d$ are n realizations of the returns on the assets, $\Delta = \{x \in \mathbb{R}^d \mid x_1, \ldots, x_d \geq 0,\ x_1 + \cdots + x_d = 1\}$ is the standard simplex for portfolios with no short positions, $\mu \in \mathbb{R}^d$ is the (estimated) average return of the assets, and $b \in \mathbb{R}$ is the desired expected return. We reformulate this problem as

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \underbrace{\frac{1}{2}\sum_{i=1}^n (a_i^T x - b)^2}_{=f(x)} + \underbrace{\delta_\Delta(x)}_{=g(x)} + \underbrace{\delta_{\{x\,|\,\mu^T x \geq b\}}(x)}_{=h(x)}$$

and apply the splitting of Theorem 4 and PPXA with $A = \nabla f$, $B = \partial g$, and $C = \partial h$. We also run the method of Boţ and Hendrich [5, Algorithm 3.1], which has been used to solve portfolio optimization problems [6]. To evaluate $J_{\alpha\nabla f}$, we compute the Cholesky factorization of $(I + \alpha A^T A)$ once and use the direct formula

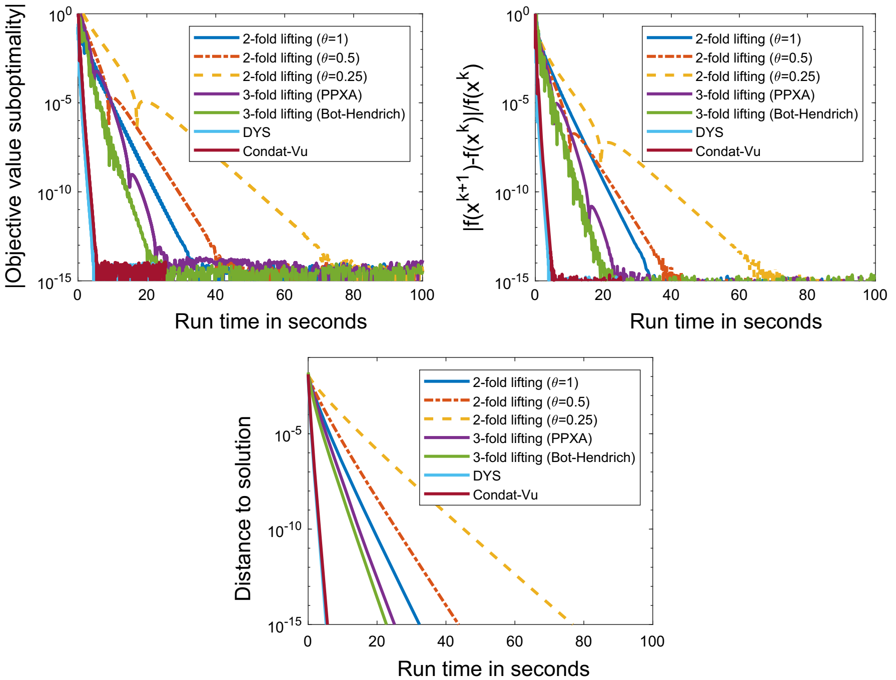

Fig. 2 |Objective value suboptimality|, $|f(x^{k+1}) - f(x^k)|/f(x^k)$, and distance to solution versus iterations for the portfolio optimization problem. We take the absolute value in the first plot, because the slightly infeasible iterates produce objective values lower than the optimal value. The rough cost per iteration is 0.025 s for CV and DYS and 0.15 s for the splitting of Theorem 4, PPXA, and Bot–Hendrich

$$J_{\alpha \nabla f}(z) = (I + \alpha A^T A)^{-1}(z + \alpha A^T b\mathbf{1})$$

where

$$A = \begin{bmatrix} a_1^T \\ \vdots \\ a_n^T \end{bmatrix}, \qquad \mathbf{1} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}.$$

Finally, we also run DYS [25] and Condat–Vũ [24,55], for which direct evaluations of ∇f were used instead of $J_{\alpha \nabla f}$. To compute the projection onto the simplex, we use the algorithm and code of [17]. For the experiments, we used synthetic data with $n = 30{,}000$ and $d = 10000$, which make the data approximately 2 GB in size. The Cholesky factorization of $I + \alpha A^T A$ requires about 30 min to compute for this problem size. The code for data generation and optimization is provided on the author's website for scientific reproducibility.

Figure 2 shows the results. The splitting of Theorem 4, which uses twofold lifting, is competitive with PPXA and Boţ–Hendrich, which use threefold lifting. However, DYS and Condat–Vũ are faster than the splittings that only use resolvents. Run-time measurements exclude the time it took to compute the Cholesky factorization, about

35.9 s. An Intel Core i7-2600 CPU operating at 3.40 GHz was used for the experiments. For all methods, the parameters were roughly tuned for best performance.

Poisson denoising with 1D total variation Consider the problem

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \lambda\sum_{i=1}^{d}\ell(x_i;y_i)+\sum_{i=1}^{d-1}|x_{i+1}-x_i|$$

where $y\in\mathbb{R}^d$, $\lambda>0$, and

$$\ell(x;y)=\begin{cases}-y^T\log(x)+x & \text{for } y>0,\ x>0\\ 0 & \text{for } y=0,\ x\geq 0\\ \infty & \text{otherwise.}\end{cases}$$

The statistical interpretation is that we wish to recover a 1D signal with small total variation corrupted by Poisson noise. The first term is the negative log-likelihood for Poisson random variables [12,14,37,58] and the second term is the 1D total variation penalty, also called fused lasso in the statistics literature [48,52,53]. 1D total variation denoising has been studied in [2,23,34,35,56]. For simplicity, assume d is odd. We reformulate this problem as

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \underbrace{\lambda\sum_{i=1}^{d}\ell(x_i;y_i)}_{=f(x)}+\underbrace{\sum_{i=1,3,\ldots,d-2}|x_{i+1}-x_i|}_{=g(x)}+\underbrace{\sum_{i=2,4,\ldots,d-1}|x_{i+1}-x_i|}_{=h(x)}$$

and apply the splitting of Theorem 4, PPXA, and PDHG with $A=\nabla f$, $B=\partial g$, and $C=\partial h$. We compute $J_{\alpha\nabla f}$ with

$$J_{\alpha\nabla f}(z)=\frac{1}{2}\left(z-\alpha\lambda+\sqrt{(z-\alpha\lambda)^2+4\alpha\lambda y}\right),$$

where the operations are elementwise. Although f is differentiable, its domain is not closed and the gradient is not Lipschitz continuous. Therefore, splittings that use ∇f are not applicable, unless a line-search is implemented. For the experiments, we used synthetic data with $n=3000$, $d=1001$, and $\lambda=1$. The code for data generation and optimization is provided on the author's website for scientific reproducibility.

Figure 3 shows the results. The splitting of Theorem 4, which uses twofold lifting, is competitive with PPXA and PDHG, which use threefold lifting. For all methods, the parameters were roughly tuned for best performance.

5 Conclusion

This work establishes that DRS is the unique frugal, unconditionally convergent resolvent-splitting without lifting for the 2 operator problem and that there is no

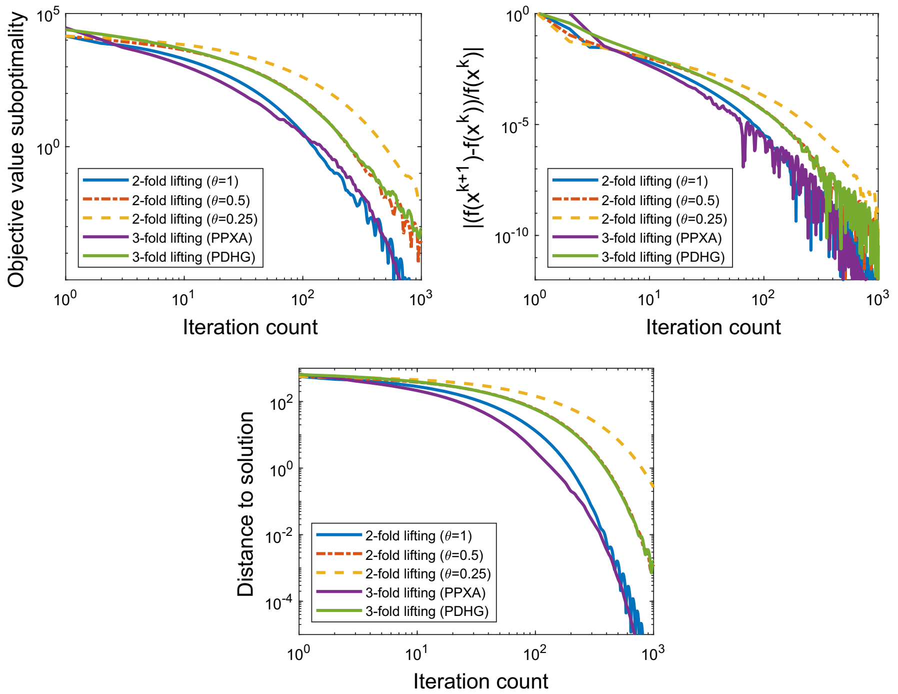

Fig. 3 Objective value, $|(f(x^{k+1}) - f(x^k))/f(x^k)|$, and distance to solution versus iterations for the Poisson denoising with 1D total variation problem

resolvent-splitting without lifting for the 3 operator problem. Furthermore, this work presents a novel, frugal, unconditionally convergent resolvent-splitting for the 3 operator problem that directly generalizes DRS. This splitting proves that twofold lifting is the minimal lifting necessary for the 3 operator problem. In other words, the presented splitting is optimal in terms of frugality and lifting.

The potential for future work based on the ideas presented in this work is large. Analyzing and establishing uniqueness or optimality of other splittings is one direction of future work. Characterizing all splittings of a given setup is another. In particular, there is no reason to believe the splitting of Theorem 4 is unique, so characterizing all frugal, unconditionally convergent resolvent-splittings for the 3 operator problem would be interesting.

Acknowledgements I would like to thank Wotao Yin for helpful comments and suggestions. I would also like to thank the anonymous associate editor and referees whose comments improved the paper significantly. In particular, the signal denoising numerical example was suggested by one of the anonymous reviewers. This work is supported in part by NSF Grant DMS-1720237 and ONR Grant N000141712162.

References

1. Banert, S.: A relaxed forward–backward splitting algorithm for inclusions of sums of monotone operators. Master's thesis, Technische Universität Chemnitz (2012)

2. Barbero, Á., Sra, S.: Fast Newton-type methods for total variation regularization. In: Proceedings of the 28th International Conference on International Conference on Machine Learning (ICML), pp. 313–320 (2011)
3. Bauschke, H.H., Combettes, P.L.: Convex Analysis and Monotone Operator Theory in Hilbert Spaces, 2nd edn. Springer, New York (2017)
4. Boţ, R., Wanka, G.: Farkas-type results with conjugate functions. SIAM J. Optim. **15**(2), 540–554 (2005)
5. Boţ, R.I., Hendrich, C.: A Douglas–Rachford type primal-dual method for solving inclusions with mixtures of composite and parallel-sum type monotone operators. SIAM J. Optim. **23**(4), 2541–2565 (2013)
6. Boţ, R.I., Hendrich, C.: Convex risk minimization via proximal splitting methods. Optim. Lett. **9**(5), 867–885 (2015)
7. Brezis, H., Lions, P.L.: Produits infinis de resolvantes. Isr. J. Math. **29**(4), 329–345 (1978)
8. Briceño-Arias, L.M.: Forward-Douglas–Rachford splitting and forward-partial inverse method for solving monotone inclusions. Optimization **64**(5), 1239–1261 (2015)
9. Briceño-Arias, L.M., Combettes, P.L.: A monotone + skew splitting model for composite monotone inclusions in duality. SIAM J. Optim. **21**(4), 1230–1250 (2011)
10. Briceño-Arias, L.M., Davis, D.: Forward–backward–half forward algorithm with non self-adjoint linear operators for solving monotone inclusions. SIAM J. Optim. **28**(4), 2839–2871 (2018)
11. Brodie, J., Daubechies, I., De Mol, C., Giannone, D., Loris, I.: Sparse and stable Markowitz portfolios. Proc. Natl. Acad. Sci. U. S. A. **106**(30), 12267–12272 (2009)
12. Byrne, C.L.: Iterative image reconstruction algorithms based on cross-entropy minimization. IEEE Trans. Image Process. **2**(1), 96–103 (1993)
13. 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)
14. Chaux, C., Pesquet, J., Pustelnik, N.: Nested iterative algorithms for convex constrained image recovery problems. SIAM J. Imaging Sci. **2**(2), 730–762 (2009)
15. Chen, P., Huang, J., Zhang, X.: A primal-dual fixed point algorithm for convex separable minimization with applications to image restoration. Inverse Probl. **29**(2), 025011 (2013)
16. Chen, P., Huang, J., Zhang, X.: A primal-dual fixed point algorithm for minimization of the sum of three convex separable functions. Fixed Point Theory Appl. **2016**, 54 (2016)
17. Chen, Y., Ye, X.: Projection onto a simplex. arXiv preprint arXiv:1101.6081 (2011)
18. Combettes, P.L., Condat, L., Pesquet, J.C., Vũ, B.C.: A forward–backward view of some primal-dual optimization methods in image recovery. In: IEEE International Conference on Image Processing (2014)
19. Combettes, P.L., Eckstein, J.: Asynchronous block-iterative primal-dual decomposition methods for monotone inclusions. Math. Program. **168**(1), 645–672 (2018)
20. Combettes, P.L., Pesquet, J.C.: A proximal decomposition method for solving convex variational inverse problems. Inverse Probl. **24**(6), 065014 (2008)
21. Combettes, P.L., Pesquet, J.C.: Proximal splitting methods in signal processing. In: Bauschke, H., Burachik, R., Combettes, P., Elser, V., Luke, D., Wolkowicz, H. (eds.) Fixed-Point Algorithms for Inverse Problems in Science and Engineering, pp. 185–212. Springer, Berlin (2011)
22. Combettes, P.L., Pesquet, J.C.: Primal-dual splitting algorithm for solving inclusions with mixtures of composite, Lipschitzian, and parallel-sum type monotone operators. Set Valued Var. Anal. **20**(2), 307–330 (2012)
23. Condat, L.: A direct algorithm for 1-D total variation denoising. IEEE Signal Process. Lett. **20**(11), 1054–1057 (2013)
24. 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)
25. Davis, D., Yin, W.: A three-operator splitting scheme and its optimization applications. Set Valued Var. Anal. **25**(4), 829–858 (2017)
26. Dinh, N., Goberna, M.A., López, M.A., Son, T.Q.: New Farkas-type constraint qualifications in convex infinite programming. ESAIM Control Optim. Calc. Var. **13**(3), 580–597 (2007)
27. Douglas, J., Rachford, H.H.: On the numerical solution of heat conduction problems in two and three space variables. Trans. Am. Math. Soc. **82**, 421–439 (1956)
28. Drori, Y., Sabach, S., Teboulle, M.: A simple algorithm for a class of nonsmooth convex–concave saddle-point problems. Oper. Res. Lett. **43**(2), 209–214 (2015)

29. Eckstein, J.: A simplified form of block-iterative operator splitting and an asynchronous algorithm resembling the multi-block alternating direction method of multipliers. J. Optim. Theory Appl. **173**(1), 155–182 (2017)
30. Esser, E., Zhang, X., Chan, T.F.: A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science. SIAM J. Imaging Sci. **3**(4), 1015–1046 (2010)
31. Farkas, J.: Theorie der einfachen ungleichungen. Journal für die reine und angewandte Mathematik **124**, 1–27 (1902)
32. Johnstone, P.R., Eckstein, J.: Projective splitting with forward steps: asynchronous and block-iterative operator splitting. arXiv preprint arXiv:1803.07043 (2018)
33. Johnstone, P.R., Eckstein, J.: Convergence rates for projective splitting. SIAM J. Optim. (2019)
34. Kamilov, U., Bostan, E., Unser, M.: Generalized total variation denoising via augmented Lagrangian cycle spinning with Haar wavelets. In: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 909–912 (2012)
35. Karahanoglu, F.I., Bayram, İ., Ville, D.V.D.: A signal processing approach to generalized 1-D total variation. IEEE Trans. Signal Process. **59**(11), 5265–5274 (2011)
36. Latafat, P., Patrinos, P.: Asymmetric forward–backward–adjoint splitting for solving monotone inclusions involving three operators. Comput. Optim. Appl. **68**(1), 57–93 (2017)
37. Le, T., Chartrand, R., Asaki, T.J.: A variational approach to reconstructing images corrupted by Poisson noise. J. Math. Imaging Vis. **27**(3), 257–263 (2007)
38. Lions, P.L., Mercier, B.: Splitting algorithms for the sum of two nonlinear operators. SIAM J. Numer. Anal. **16**(6), 964–979 (1979)
39. Loris, I., Verhoeven, C.: On a generalization of the iterative soft-thresholding algorithm for the case of non-separable penalty. Inverse Probl. **27**(12), 125007 (2011)
40. Malitsky, Y., Tam, M.K.: A forward–backward splitting method for monotone inclusions without cocoercivity. arXiv preprint arXiv:1808.04162 (2018)
41. Martinet, B.: Régularisation d'inéquations variationnelles par approximations successives. Rev. Fr. d'Inform. Rech. Oper. Sér. Rouge **4**(3), 154–158 (1970)
42. Martinet, B.: Determination approchée d'un point fixe d'une application pseudo-contractante. C. R. l'Acad. Sci. Sér. A **274**, 163–165 (1972)
43. Passty, G.B.: Ergodic convergence to a zero of the sum of monotone operators in Hilbert space. J. Math. Anal. Appl. **72**(2), 383–390 (1979)
44. Peaceman, D.W., Rachford, H.H.: The numerical solution of parabolic and elliptic differential equations. J. Soc. Ind. Appl. Math. **3**(1), 28–41 (1955)
45. Pock, T., Cremers, D., Bischof, H., Chambolle, A.: An algorithm for minimizing the Mumford-Shah functional. In: IEEE International Conference on Computer Vision (2009)
46. Raguet, H.: A note on the forward-Douglas-Rachford splitting for monotone inclusion and convex optimization. Optim. Lett. **13**(4), 717–740 (2019). https://doi.org/10.1007/s11590-018-1272-8
47. Raguet, H., Fadili, J., Peyré, G.: A generalized forward–backward splitting. SIAM J. Imaging Sci. **6**(3), 1199–1226 (2013)
48. Rapaport, F., Barillot, E., Vert, J.P.: Classification of arrayCGH data using fused SVM. Bioinformatics **24**(13), i375–i382 (2008)
49. Rockafellar, R.T.: Monotone operators and the proximal point algorithm. SIAM J. Control Optim. **14**(5), 877–898 (1976)
50. Ryu, E.K., Boyd, S.: Primer on monotone operator methods. Appl. Comput. Math. **15**, 3–43 (2016)
51. Spingarn, J.E.: Applications of the method of partial inverses to convex programming: decomposition. Math. Program. **32**(2), 199–223 (1985)
52. Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., Knight, K.: Sparsity and smoothness via the fused lasso. J. R. Stat. Soc. Ser. B. Stat. Methodol. **67**(1), 91–108 (2005)
53. Tibshirani, R., Wang, P.: Spatial smoothing and hot spot detection for CGH data using the fused lasso. Biostatistics **9**(1), 18–29 (2008)
54. Tseng, P.: A modified forward–backward splitting method for maximal monotone mappings. SIAM J. Control Optim. **38**(2), 431–446 (2000)
55. Vũ, B.C.: A splitting algorithm for dual monotone inclusions involving cocoercive operators. Adv. Comput. Math. **38**(3), 667–681 (2013)
56. Wahlberg, B., Boyd, S., Annergren, M., Wang, Y.: An ADMM algorithm for a class of total variation regularized estimation problems. IFAC Proc. Vol. **45**(16), 83–88 (2012)

57. Yan, M.: A new primal-dual algorithm for minimizing the sum of three functions with a linear operator. J. Sci. Comput. **76**(3), 1698–1717 (2018)
58. Zanella, R., Boccacci, P., Zanni, L., Bertero, M.: Efficient gradient projection methods for edge-preserving removal of Poisson noise. Inverse Probl. **25**(4), 045010 (2009)
59. Zhu, M., Chan, T.: An efficient primal-dual hybrid gradient algorithm for total variation image restoration. UCLA CAM Report 08–34 (2008)