

A primal-dual splitting algorithm for composite monotone inclusions with minimal lifting

Francisco J. Aragón-Artacho* Radu I. Bot† David Torregrosa-Belén‡

August 25, 2022

Abstract

In this work, we study resolvent splitting algorithms for solving composite monotone inclusion problems. The objective of these general problems is finding a zero in the sum of maximally monotone operators composed with linear operators. Our main contribution is establishing the first primal-dual splitting algorithm for composite monotone inclusions with minimal lifting. Specifically, the proposed scheme reduces the dimension of the product space where the underlying fixed point operator is defined, in comparison to other algorithms, without requiring additional evaluations of the resolvent operators. We prove the convergence of this new algorithm and analyze its performance in a problem arising in image deblurring and denoising. This work also contributes to the theory of resolvent splitting algorithms by extending the minimal lifting theorem recently proved by Malitsky and Tam to schemes with resolvent parameters.

Keywords monotone operator · monotone inclusion · splitting algorithm · primal-dual algorithm · minimal lifting

MSC2020 47H05 · 65K10 · 90C30

1 Introduction

In the last decades, monotone inclusion problems have become an attractive topic of research in operator theory and numerical optimization. The wide variety of situations in applied mathematics that can be modeled as finding a zero of the sum of mixtures of maximally monotone operators is one of the reasons for its increasing popularity. Among the methods that are usually employed for tackling these problems, *splitting algorithms* (see, e.g., [2, Chapter 26]) are the ones that have received more attention. Using simple operations, these methods define an iterative sequence which separately handles the operators defining the problem and is convergent to a solution to the inclusion problem. Further, as these methods only use first-order information, they are well suited for large-scale optimization problems.

In this work, we focus on the study of *primal-dual splitting algorithms* for composite monotone inclusion problems in real Hilbert spaces of the following form.

*Department of Mathematics, University of Alicante, Alicante, SPAIN. Email: francisco.aragon@ua.es

†Faculty of Mathematics, University of Vienna, Vienna, AUSTRIA. Email: radu.bot@univie.ac.at

‡Department of Mathematics, University of Alicante, Alicante, SPAIN. Email: david.torregrosa@ua.es

Problem 1.1. Let \mathcal{H} and $(\mathcal{G}_j)_{1 \leq j \leq m}$ be real Hilbert spaces. Let $A_1, \dots, A_n : \mathcal{H} \rightrightarrows \mathcal{H}$ be maximally monotone operators, let $B_j : \mathcal{G}_j \rightrightarrows \mathcal{G}_j$ be maximally monotone and $L_j : \mathcal{H} \rightarrow \mathcal{G}_j$ be a bounded linear operator whose adjoint is denoted by L_j^* , for all $j \in \{1, \dots, m\}$. The problem consists in solving the primal inclusion

$$\text{find } x \in \mathcal{H} \text{ such that } 0 \in \sum_{i=1}^n A_i(x) + \sum_{j=1}^m L_j^* B_j(L_j x), \quad (1)$$

together with its associated dual inclusion

$$\text{find } (u_1, \dots, u_m) \in \mathcal{G}_1 \times \dots \times \mathcal{G}_m \text{ such that } (\exists x \in \mathcal{H}) \begin{cases} -\sum_{j=1}^m L_j^* u_j \in \sum_{i=1}^n A_i(x), \\ u_j \in B_j(L_j x) \quad j = 1, \dots, m. \end{cases} \quad (2)$$

Problem 1.1 encompasses numerous important problems in mathematical optimization and real-world applications, see e.g. [10, 11, 20]. In these settings, it is highly desirable to devise algorithms that simultaneously obtain solutions to both problems (1) and (2) –namely, a *primal-dual solution*– and which only make use of resolvents of the maximally monotone operators, forward evaluations of the linear operators and their adjoints, scalar multiplication and vector addition. Many splitting methods can be found in the literature satisfying these conditions, see e.g. [3, 4, 5, 12, 22]. One of the best-known primal-dual algorithm is the one proposed by Briceño-Arias and Combettes in [7], which was further studied in [6]. To derive this scheme, let us consider first the particular instance of Problem 1.1 in which $n = m = 1$ and let us define the pair of operators M and N given by

$$\begin{cases} M : \mathcal{H} \times \mathcal{G} \rightrightarrows \mathcal{H} \times \mathcal{G} : (x, u) \rightarrow A(x) \times B^{-1}(u), \\ N : \mathcal{H} \times \mathcal{G} \rightarrow \mathcal{H} \times \mathcal{G} : (x, u) \rightarrow (L^* u, -Lx). \end{cases}$$

The operator M is maximally monotone and N is a skew symmetric bounded linear operator. Further, the set of zeros of the sum $M + N$ consists of primal-dual solutions to Problem 1.1. Applying the forward-backward-forward algorithm to the problem of finding the zeros of $M + N$ results in the fixed point iteration given by

$$x^{k+1} = (J_{\gamma M} (\text{Id} - \gamma N) + \gamma N (\text{Id} - J_{\gamma M} (\text{Id} - \gamma N)))(x^k) \quad \forall k \geq 0, \quad (3)$$

where $\gamma > 0$, Id denotes the identity operator and $J_{\gamma A}$ stands for the resolvent of A with parameter γ (see Definition 2.3). Thus, since the resolvent of a cartesian product is the cartesian product of the resolvents, it can be seen that (3) is a *full splitting algorithm*, as it only requires evaluations of the resolvents $J_{\gamma A}$ and $J_{\gamma B^{-1}}$, and of the linear operator and its adjoint.

The general problem involving more than two operators can be addressed by setting

$$\begin{aligned} A &:= A_1, \quad B := A_2 \times \dots \times A_n \times B_1 \times \dots \times B_m \\ \text{and} \quad L &:= \text{Id} \times \overset{(n)}{\dots} \times \text{Id} \times L_1 \times \dots \times L_m. \end{aligned}$$

In this case, according to (3), the resulting algorithm is generated by a fixed point iteration of an operator defined in the ambient space $\mathcal{H}^n \times \mathcal{G}_1 \times \dots \times \mathcal{G}_m$. The dimension of the underlying space is directly related to the memory requirements of the resulting algorithm. In general, a smaller dimension of the space translates into less consumption of computational resources. For this reason, the development of algorithms with reduced dimension for solving monotone inclusion problems has recently become an active topic of research [8, 16, 14, 19].

Lifted splitting algorithms The notion of *lifted splitting*, first introduced in [19], relates a fixed point algorithm with the dimension of its underlying ambient space. Consider the simplest case of the classical monotone inclusion problem obtained by setting $m = 0$ in (1):

Problem 1.2. Let $A_1, \dots, A_n : \mathcal{H} \rightrightarrows \mathcal{H}$ be maximally monotone operators and consider the problem

$$\text{find } x \in \mathcal{H} \text{ such that } 0 \in \sum_{i=1}^n A_i(x).$$

A fixed point algorithm for finding a solution to Problem 1.2 employs a d -fold lifting if its underlying fixed point operator can be defined on the d -fold Cartesian product \mathcal{H}^d . For example, if $n = 2$, the famous Douglas–Rachford algorithm [15] makes use of a 1-fold lifting, since it can be written as the fixed point iteration

$$x^{k+1} = x^k + \lambda (J_{A_2} (2J_{A_1} - \text{Id}) - J_{A_1}) (x^k) \quad \forall k \geq 0,$$

with $\lambda \in]0, 2[$. Until very recently, the only way to tackle the problem when $n > 2$ was using Pierra’s product space reformulation [18], which implies an n -fold lifting. Nowadays, various algorithms have been proposed allowing to solve the problem by only resorting to an $(n - 1)$ -lifting, see e.g. [8, ?]. This reduction from n to $n - 1$ has been proven to be minimal [16] when the algorithms are required to be *frugal resolvent splittings* [19], which means that each of the resolvents J_{A_1}, \dots, J_{A_n} is evaluated only once per iteration.

To the best of the authors’ knowledge, the notion of lifting has not been developed in the setting of primal-dual inclusions given by Problem 1.1. We will say that a primal-dual splitting has (d, f) -lifting if the underlying fixed point operator can be written in the product space

$$\mathcal{H}^d \times \mathcal{G}_1^{f_1} \times \dots \times \mathcal{G}_m^{f_m},$$

with $f = \sum_{j=1}^m f_j$. Thus, the Briceño-Arias–Combettes primal-dual splitting algorithm makes use of an (n, m) -fold lifting. This is also the case for the other primal-dual algorithms existing in the literature. In this work, we propose the first $(n - 1, m)$ -lifted splitting method for solving primal-dual inclusions and demonstrate the minimality of the algorithm. In order to do this, it is important to note the definition of frugal resolvent splitting does not allow the use of parametrized resolvents. The inclusion of these resolvent parameters is of crucial importance for controlling the Lipschitz constants of the linear operators in Problem 1.1, as can be seen in all the existent primal-dual schemes. This motivates the introduction of the concept of *frugal parametrized resolvent splitting* whose definition coincides with the one of frugal resolvent splitting except that it permits the inclusion of resolvent parameters. Our contribution to the theory of minimal lifting splitting methods is double: (i) we extend the results of Malitsky–Tam in [16, Section 3] to frugal parametrized resolvent splitting algorithms, (ii) we prove that for a frugal primal-dual parametrized resolvent splitting (see Section 3 for a precise definition) with (d, m) -fold lifting to solve Problem 1.1, one necessarily has $d \geq n - 1$. Our proposed algorithm is the first algorithm in the literature being minimal according to this relation.

The rest of this work is structured as follows. In Section 2 we recall some preliminary notions and results. In particular, in Section 2.1 we present the extension of the results by Malitsky–Tam [16] to parametrized resolvent splitting algorithms. In Section 3, we introduce the first primal-dual algorithm with reduced lifting for composite monotone inclusion problems and prove its convergence. The concept of parametrized resolvent splitting is adapted to primal-dual schemes in Section 3. We prove a minimality theorem under the

hypothesis of frugality and show that our proposed algorithm verifies it. In Section 4 we include a numerical experiment on image deblurring and compare the performance of the new algorithm with the best performing primal-dual algorithm for this problem. The paper ends with some conclusions and possible future work directions in Section 5. Finally, in Appendix A, a detailed proof of the results in Section 2.1 is presented.

2 Preliminaries

Throughout this paper, \mathcal{H} , \mathcal{G} and $(\mathcal{G}_j)_{1 \leq j \leq m}$ are real Hilbert spaces. Otherwise stated, to simplify the notation we will employ $\langle \cdot, \cdot \rangle$ and $\| \cdot \|$ to denote the inner product and the induced norm, respectively, of any space. We use \rightarrow to denote norm convergence of a sequence. We denote by \mathcal{H}^n the product Hilbert space $\mathcal{H}^n = \mathcal{H} \times \cdots \times \mathcal{H}$ with inner product defined as

$$\langle (x_1, \dots, x_n), (\bar{x}_1, \dots, \bar{x}_n) \rangle := \sum_{i=1}^n \langle x_i, \bar{x}_i \rangle \quad \forall (x_1, \dots, x_n), (\bar{x}_1, \dots, \bar{x}_n) \in \mathcal{H}^n.$$

Sequences and sets in product spaces are marked with bold, e.g., $\mathbf{x} = (x_1, \dots, x_n) \in \mathcal{H}^n$.

For a *set-valued operator*, we write $A : \mathcal{H} \rightrightarrows \mathcal{H}$, in opposite to $A : \mathcal{H} \rightarrow \mathcal{H}$ which denotes a *single-valued operator*. The notation dom , Fix , zer and gra is used for the *domain*, the *set of fixed points*, the *zeros* and the *graph* of A , respectively, i.e.,

$$\begin{aligned} \text{dom } A &:= \{x \in \mathcal{H} : A(x) \neq \emptyset\}, & \text{Fix } A &:= \{x \in \mathcal{H} : x \in A(x)\}, \\ \text{zer } A &:= \{x \in \mathcal{H} : 0 \in A(x)\}, & \text{gra } A &:= \{(x, u) \in \mathcal{H} \times \mathcal{H} : u \in A(x)\}. \end{aligned}$$

The inverse operator of A , denoted by A^{-1} , is the operator whose graph is given by $\text{gra } A^{-1} = \{(u, x) \in \mathcal{H} \times \mathcal{H} : u \in A(x)\}$. The identity operator is denoted by Id . When $L : \mathcal{H} \rightarrow \mathcal{G}$ is a bounded linear operator, we use $L^* : \mathcal{G} \rightarrow \mathcal{H}$ to denote its *adjoint*, which is the unique bounded linear operator such that $\langle Lx, y \rangle = \langle x, L^*y \rangle$, for all $x \in \mathcal{H}$ and $y \in \mathcal{G}$.

To simplify the notation, we will use $\llbracket k, l \rrbracket$ to denote the set of integers between $k, l \in \mathbb{N}$, i.e.,

$$\llbracket k, l \rrbracket := \begin{cases} \{k, k+1, \dots, l\} & \text{if } k \leq l, \\ \emptyset & \text{otherwise.} \end{cases}$$

Definition 2.1. An operator $T : \mathcal{H} \rightarrow \mathcal{H}$ is said to be

(i) κ -Lipschitz continuous for $\kappa > 0$ if

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

(ii) nonexpansive if it is 1-Lipschitz continuous, i.e.,

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

(iii) α -averaged nonexpansive for $\alpha \in]0, 1[$ if

$$\|T(x) - T(y)\|^2 + \frac{1-\alpha}{\alpha} \|(\text{Id} - T)(x) - (\text{Id} - T)(y)\|^2 \leq \|x - y\|^2 \quad \forall x, y \in \mathcal{H}.$$

Definition 2.2. A set-valued operator $A : \mathcal{H} \rightrightarrows \mathcal{H}$ is monotone if

$$\langle x - y, u - v \rangle \geq 0 \quad \forall (x, u), (y, v) \in \text{gra } A.$$

Furthermore, A is said to be maximally monotone if there exists no monotone operator $B : \mathcal{H} \rightrightarrows \mathcal{H}$ such that $\text{gra } B$ properly contains $\text{gra } A$.

Definition 2.3. Given an operator $A : \mathcal{H} \rightrightarrows \mathcal{H}$, the resolvent of A with parameter $\gamma > 0$ is the operator $J_{\gamma A} : \mathcal{H} \rightrightarrows \mathcal{H}$ defined by $J_{\gamma A} := (\text{Id} + \gamma A)^{-1}$.

The next result contains Minty's theorem [17].

Proposition 2.4 ([2, Proposition 23.10]). Let $A : \mathcal{H} \rightrightarrows \mathcal{H}$ be monotone and let $\gamma > 0$. Then,

(i) $J_{\gamma A}$ is single-valued,

(ii) $\text{dom } J_{\gamma A} = \mathcal{H}$ if and only if A is maximally monotone.

2.1 Parametrized resolvent splitting

Besides developing lifted splitting algorithms with reduced dimension, different works have been devoted to determine the minimal dimension reduction that can be achieved under some conditions. This is the case of [16, 19], where a minimality result is obtained for the classical monotone inclusion Problem 1.2. In what follows, we employ T for denoting a *fixed point operator* and S for a *solution operator*, both depending on the maximally monotone operators appearing in the problem.

Definition 2.5 (Fixed point encoding [19]). A pair of operators (T, S) is a fixed point encoding for Problem 1.2 if, for all particular instance of the problem,

$$\text{Fix } T \neq \emptyset \iff \text{zer} \left(\sum_{i=1}^n A_i \right) \neq \emptyset \text{ and } \mathbf{z} \in \text{Fix } T \implies S(\mathbf{z}) \in \text{zer} \left(\sum_{i=1}^n A_i \right).$$

Previous works on minimality are based on the concept of *resolvent splitting*, which does not allow employing parametrized resolvents (i.e., it only permits computation of the resolvents J_{A_1}, \dots, J_{A_n}). In this work, we introduce the notion of *parametrized resolvent splitting* and adapt the minimality result in [16, Section 3] to the more general parametrized setting. Since the reasoning is very similar to the one in the mentioned reference, we only present the results here and refer the interested reader to Appendix A for a detailed demonstration.

Definition 2.6 (Parametrized resolvent splitting). A fixed point encoding (T, S) for Problem 1.2 is a parametrized resolvent splitting if, for all particular instances of the problem, there is a finite procedure that evaluates T and S at a given point which only uses vector addition, scalar multiplication, and the parametrized resolvents of A_1, \dots, A_n .

Definition 2.7 (Frugality). A parametrized resolvent splitting (T, S) for Problem 1.2 is frugal if, in addition, each of the parametrized resolvents of A_1, \dots, A_n is used exactly once.

Definition 2.8 (Lifting [19]). Let $d \in \mathbb{N}$. A fixed point encoding (T, S) is a d -fold lifting for Problem 1.2 if $T : \mathcal{H}^d \rightarrow \mathcal{H}^d$ and $S : \mathcal{H}^d \rightarrow \mathcal{H}$.

Example 2.9. In [8], a product space reformulation with reduced dimension is proposed, which applied to Problem 1.2 yields the following lifted splitting. Given any $\gamma > 0$ and $\lambda \in]0, 2]$, the algorithm in [8, Theorem 5.1] can be defined by the operator $R : \mathcal{H}^{n-1} \rightarrow \mathcal{H}^{n-1}$ given by

$$R(\mathbf{z}) := \mathbf{z} + \lambda \begin{pmatrix} x_1 - x_0 \\ x_2 - x_0 \\ \vdots \\ x_{n-1} - x_0 \end{pmatrix},$$

where $\mathbf{z} = (z_0, z_1, \dots, z_{n-1})$ and $\mathbf{x} = (x_0, x_1, \dots, x_{n-1}) \in \mathcal{H}^n$ is the vector defined as

$$\begin{cases} x_0 = J_{\frac{\gamma}{n-1}A_n} \left(\frac{1}{n-1} \sum_{i=1}^{n-1} z_i \right), \\ x_i = J_{\gamma A_i}(2x_0 - z_i) \quad \forall i \in \llbracket 1, n-1 \rrbracket. \end{cases}$$

Moreover, if we let $S : \mathcal{H}^{n-1} \rightarrow \mathcal{H}$ be the operator given by

$$S(\mathbf{z}) := J_{\frac{\gamma}{n-1}A_n} \left(\frac{1}{n-1} \sum_{i=1}^{n-1} z_i \right),$$

then the pair (R, S) is a frugal parametrized resolvent splitting with $(n-1)$ -fold lifting which is not a resolvent splitting, since it makes use of resolvent parameters.

Malitsky and Tam prove in [16, Theorem 3.3] that the minimal lifting that one can achieve for Problem 1.2 with frugal resolvent splittings is $n-1$. From their proof, it cannot be directly determined whether the same result holds when the resolvents are allowed to have different parameters. The next theorem provides an affirmative answer to this question.

Theorem 2.10 (Minimal lifting for frugal parametrized splittings). *Let $n \geq 2$ and let (T, S) be a frugal parametrized resolvent splitting with d -fold lifting for Problem 1.2. Then, $d \geq n-1$.*

Proof. See Theorem A.3 in Appendix A. □

3 A primal-dual splitting with minimal lifting

In this section we devise a primal-dual splitting algorithm for Problem 1.1 with minimal lifting. We base our analysis in the case in which the primal problem involves only one linear composition, i.e. $m = 1$, and later extend to an arbitrary finite number of linearly composed maximally monotone operators by appealing to a product space reformulation. Lastly, we prove minimality of the algorithm by adapting the concept of lifted splitting to primal-dual algorithms.

The case with one linear composition

Let $n \geq 2$. We start by considering the primal-dual problem given by

$$\text{find } x \in \mathcal{H} \text{ such that } 0 \in \sum_{i=1}^n A_i(x) + L^*B(Lx), \quad (4)$$

and

$$\text{find } u \in \mathcal{G} \text{ such that } 0 \in -L \left(\sum_{i=1}^n A_i \right)^{-1} (-L^*u) + B^{-1}(u), \quad (5)$$

where $A_1, \dots, A_n : \mathcal{H} \rightrightarrows \mathcal{H}$ and $B : \mathcal{G} \rightrightarrows \mathcal{G}$ are maximally monotone operators and $L : \mathcal{H} \rightarrow \mathcal{G}$ is a bounded linear operator. Note that in this case (5) corresponds to the Attouch–Théra dual problem of (4), see [1]. In the following, we denote the set of solutions of (4) and (5) by \mathcal{P} and \mathcal{D} , respectively, and consider the set \mathbf{Z} defined as

$$\mathbf{Z} := \left\{ (x, u) \in \mathcal{H} \times \mathcal{G} : -L^*u \in \sum_{i=1}^n A_i(x) \text{ and } u \in B(Lx) \right\},$$

which is useful for tackling primal-dual inclusion problems. It is well-known that \mathbf{Z} is a subset of $\mathcal{P} \times \mathcal{D}$ and that

$$\mathcal{P} \neq \emptyset \iff \mathbf{Z} \neq \emptyset \iff \mathcal{D} \neq \emptyset.$$

Indeed, we have

$$\begin{aligned} \exists x \in \mathcal{P} &\iff (\exists x \in \mathcal{H}) \quad 0 \in \sum_{i=1}^n A_i(x) + L^*B(Lx) \\ &\iff (\exists (x, u) \in \mathcal{H} \times \mathcal{G}) \begin{cases} -L^*(u) \in \sum_{i=1}^n A_i(x), \\ u \in B(Lx), \end{cases} \\ &\iff (\exists (x, u) \in \mathcal{H} \times \mathcal{G}) \begin{cases} x \in \left(\sum_{i=1}^n A_i \right)^{-1} (-L^*u), \\ Lx \in B^{-1}(u), \end{cases} \\ &\iff (\exists u \in \mathcal{G}) \quad 0 \in -L \left(\sum_{i=1}^n A_i \right)^{-1} (-L^*u) + B^{-1}(u) \iff \exists u \in \mathcal{D}. \end{aligned}$$

We refer to an element of \mathbf{Z} as a *primal-dual solution* of (4)-(5).

Now, we introduce a fixed point algorithm for solving the primal-dual problem given by (4)-(5). Let $\lambda, \gamma > 0$ and let $T : \mathcal{H}^{n-1} \times \mathcal{G} \rightarrow \mathcal{H}^{n-1} \times \mathcal{G}$ be the operator given by

$$T \begin{pmatrix} \mathbf{z} \\ v \end{pmatrix} := \begin{pmatrix} \mathbf{z} \\ v \end{pmatrix} + \lambda \begin{pmatrix} x_2 - x_1 \\ x_3 - x_2 \\ \vdots \\ x_n - x_{n-1} \\ \gamma(y - Lx_n) \end{pmatrix}, \quad (6)$$

where $(\mathbf{x}, y) = (x_1, \dots, x_n, y) \in \mathcal{H}^n \times \mathcal{G}$ depends on $(\mathbf{z}, v) = (z_1, \dots, z_{n-1}, v) \in \mathcal{H}^{n-1} \times \mathcal{G}$ in the following way

$$\begin{cases} x_1 = J_{A_1}(z_1), \\ x_i = J_{A_i}(z_i + x_{i-1} - z_{i-1}), \quad \forall i \in \llbracket 2, n-1 \rrbracket, \\ x_n = J_{A_n}(x_1 + x_{n-1} - z_{n-1} - L^*(\gamma Lx_1 - v)), \\ y = J_{B/\gamma} \left(L(x_1 + x_n) - \frac{v}{\gamma} \right). \end{cases} \quad (7)$$

In the next lemma we characterize the set of fixed points of the operator T by means of the set of primal-dual solutions to (4)-(5).

Lemma 3.1. *Let $n \geq 2$ and $\lambda, \gamma > 0$. The following assertions hold.*

- (i) *If $(\bar{x}, \bar{u}) \in \mathbf{Z}$, then there exists $\bar{z} \in \mathcal{H}^{n-1}$ such that $(\bar{z}, \gamma L\bar{x} - \bar{u}) \in \text{Fix } T$.*
- (ii) *If $(\bar{z}_1, \dots, \bar{z}_{n-1}, \bar{v}) \in \text{Fix } T$, then $(J_{A_1}(\bar{z}_1), \gamma L\bar{x} - \bar{v}) \in \mathbf{Z}$.*

As a result,

$$\text{Fix } T \neq \emptyset \iff \mathbf{Z} \neq \emptyset.$$

Proof. (i) Let $(\bar{x}, \bar{u}) \in \mathbf{Z}$. Then, $\bar{u} \in B(L\bar{x})$ and there exists $(a_1, \dots, a_n) \in \mathcal{H}^n$ such that $a_i \in A_i(\bar{x})$ and $-L^*\bar{u} = \sum_{i=1}^n a_i$. Consider the vectors $(\bar{z}_1, \dots, \bar{z}_{n-1}, \bar{v}) \in \mathcal{H}^{n-1} \times \mathcal{G}$ defined as

$$\begin{cases} \bar{z}_1 := \bar{x} + a_1 \in (\text{Id} + A_1)(\bar{x}), \\ \bar{z}_i := a_i + \bar{z}_{i-1} = (\text{Id} + A_i)(\bar{x}) - \bar{x} + \bar{z}_{i-1}, \quad \forall i \in \llbracket 2, n-1 \rrbracket, \\ \bar{v} := \gamma L\bar{x} - \bar{u} \in (\gamma \text{Id} - B)(L\bar{x}). \end{cases}$$

Then, we deduce that $\bar{x} = J_{A_1}(\bar{z}_1)$ and $\bar{x} = J_{A_i}(\bar{z}_i + \bar{x} - \bar{z}_{i-1})$ for all $i \in \llbracket 2, n-1 \rrbracket$. Moreover, we have

$$\begin{aligned} 2\bar{x} - \bar{z}_{n-1} - L^*(\gamma L\bar{x} - \bar{v}) &= 2\bar{x} - \bar{z}_{n-1} - L^*(\bar{u}) \\ &= \bar{x} + a_n + \bar{x} - \bar{z}_{n-1} + \sum_{i=1}^{n-1} a_i \\ &= \bar{x} + a_n + \bar{x} - \bar{z}_{n-1} + \sum_{i=2}^{n-1} (\bar{z}_i - \bar{z}_{i-1}) + \bar{z}_1 - \bar{x} = (\text{Id} + A_n)(\bar{x}). \end{aligned}$$

Altogether, we obtain

$$\begin{cases} \bar{x} = J_{A_1}(\bar{z}_1), \\ \bar{x} = J_{A_i}(\bar{z}_i + \bar{x} - \bar{z}_{i-1}), \quad \forall i \in \llbracket 2, n-1 \rrbracket, \\ \bar{x} = J_{A_n}(2\bar{x} - \bar{z}_{n-1} - L^*(\gamma L\bar{x} - \bar{v})), \\ L\bar{x} = J_{B/\gamma} \left(2L\bar{x} - \frac{\bar{v}}{\gamma} \right), \end{cases}$$

which implies that $(\bar{z}_1, \dots, \bar{z}_{n-1}, \bar{v}) \in \text{Fix } T$.

(ii) Let $(\bar{z}_1, \dots, \bar{z}_{n-1}, \bar{v}) \in \text{Fix } T$ and set $\bar{x} := J_{A_1}(\bar{z}_1)$. By (6), $y = L\bar{x}$ and $\bar{x}_i = \bar{x}$ for all $i = 1, \dots, n$. Consequently, from (7) we derive

$$\begin{cases} \bar{z}_1 - \bar{x} \in A_1(\bar{x}), \\ \bar{z}_i - \bar{z}_{i-1} \in A_i(\bar{x}), \quad \forall i \in \llbracket 2, n-1 \rrbracket, \\ \bar{x} - \bar{z}_{n-1} - L^*(\gamma L\bar{x} - \bar{v}) \in A_n(\bar{x}), \\ \gamma L\bar{x} - \bar{v} \in B(L\bar{x}). \end{cases}$$

Summing together the first n inclusions above and setting $\bar{u} := \gamma L\bar{x} - \bar{v}$, we deduce

$$\begin{cases} -L^*\bar{u} \in \sum_{i=1}^n A_i(\bar{x}), \\ \bar{u} \in B(L\bar{x}), \end{cases}$$

which implies $(\bar{x}, \bar{u}) \in \mathbf{Z}$, as claimed. \square

The following technical lemma provides nonexpansive properties of the operator T in the Hilbert space $\mathcal{H}^{n-1} \times \mathcal{G}$ with scalar product given by

$$\langle (z_1, \dots, z_{n-1}, v), (\bar{z}_1, \dots, \bar{z}_{n-1}, \bar{v}) \rangle_\gamma := \sum_{i=1}^{n-1} \langle z_i, \bar{z}_i \rangle_{\mathcal{H}} + \frac{1}{\gamma} \langle v, \bar{v} \rangle_{\mathcal{G}}, \quad (8)$$

for $(z_1, \dots, z_{n-1}, v), (\bar{z}_1, \dots, \bar{z}_{n-1}, \bar{v}) \in \mathcal{H}^{n-1} \times \mathcal{G}$ and $\gamma > 0$.

Lemma 3.2. *For all $(\mathbf{z}, v) = (z_1, \dots, z_{n-1}, v) \in \mathcal{H}^{n-1} \times \mathcal{G}$ and $(\bar{\mathbf{z}}, \bar{v}) = (\bar{z}_1, \dots, \bar{z}_{n-1}, \bar{v}) \in \mathcal{H}^{n-1} \times \mathcal{G}$,*

$$\begin{aligned} & \|T(\mathbf{z}, v) - T(\bar{\mathbf{z}}, \bar{v})\|_\gamma^2 + \frac{1-\lambda}{\lambda} \|(\text{Id} - T)(\mathbf{z}, v) - (\text{Id} - T)(\bar{\mathbf{z}}, \bar{v})\|_\gamma^2 \\ & + \frac{1-\gamma\|L\|^2}{\lambda} \left\| \sum_{i=1}^{n-1} (\text{Id} - T)(\mathbf{z}, v)_i - \sum_{i=1}^{n-1} (\text{Id} - T)(\bar{\mathbf{z}}, \bar{v})_i \right\|_\gamma^2 \leq \|(\mathbf{z}, v) - (\bar{\mathbf{z}}, \bar{v})\|_\gamma^2, \end{aligned} \quad (9)$$

where $\|\cdot\|_\gamma$ denotes the norm induced by the scalar product (8). In particular, if $\lambda \in]0, 1[$ and $\gamma \in \left]0, \frac{1}{\|L\|^2}\right]$, the operator T is λ -averaged nonexpansive.

Proof. Let $(x_1, \dots, x_n, y) \in \mathcal{H}^n \times \mathcal{G}$ and $(\bar{x}_1, \dots, \bar{x}_n, \bar{y}) \in \mathcal{H}^n \times \mathcal{G}$ be given by (7) from (\mathbf{z}, v) and $(\bar{\mathbf{z}}, \bar{v})$, respectively. For simplicity, we denote $(\mathbf{z}^+, v^+) = T(\mathbf{z}, v)$ and $(\bar{\mathbf{z}}^+, \bar{v}^+) = T(\bar{\mathbf{z}}, \bar{v})$. Since $z_1 - x_1 \in A_1(x_1)$ and $\bar{z}_1 - \bar{x}_1 \in A_1(\bar{x}_1)$, by monotonicity of A_1

$$\begin{aligned} 0 & \leq \langle (z_1 - x_1) - (\bar{z}_1 - \bar{x}_1), x_1 - \bar{x}_1 \rangle \\ & = \langle (z_1 - x_1) - (\bar{z}_1 - \bar{x}_1), x_1 - x_2 \rangle + \langle (z_1 - x_1) - (\bar{z}_1 - \bar{x}_1), x_2 - \bar{x}_1 \rangle. \end{aligned} \quad (10)$$

For every $i \in \llbracket 2, n-1 \rrbracket$, we have $z_i + x_{i-1} - z_{i-1} - x_i \in A_i(x_i)$ and $\bar{z}_i + \bar{x}_{i-1} - \bar{z}_{i-1} - \bar{x}_i \in A_i(\bar{x}_i)$ and thus, by monotonicity of A_i

$$\begin{aligned} 0 & \leq \langle (z_i + x_{i-1} - z_{i-1} - x_i) - (\bar{z}_i + \bar{x}_{i-1} - \bar{z}_{i-1} - \bar{x}_i), x_i - \bar{x}_i \rangle \\ & = \langle (z_i - x_i) - (\bar{z}_i - \bar{x}_i), x_i - \bar{x}_i \rangle - \langle (z_{i-1} - x_{i-1}) - (\bar{z}_{i-1} - \bar{x}_{i-1}), x_i - \bar{x}_i \rangle \\ & = \langle (z_i - x_i) - (\bar{z}_i - \bar{x}_i), x_i - x_{i+1} \rangle + \langle (z_i - x_i) - (\bar{z}_i - \bar{x}_i), x_{i+1} - \bar{x}_i \rangle \\ & \quad - \langle (z_{i-1} - x_{i-1}) - (\bar{z}_{i-1} - \bar{x}_{i-1}), x_i - \bar{x}_{i-1} \rangle \\ & \quad - \langle (z_{i-1} - x_{i-1}) - (\bar{z}_{i-1} - \bar{x}_{i-1}), \bar{x}_{i-1} - \bar{x}_i \rangle. \end{aligned} \quad (11)$$

Now, since $x_1 + x_{n-1} - z_{n-1} - x_n - L^*(\gamma Lx_1 - v) \in A_n(x_n)$ and $\bar{x}_1 + \bar{x}_{n-1} - \bar{z}_{n-1} - \bar{x}_n - L^*(\gamma L\bar{x}_1 - \bar{v}) \in A_n(\bar{x}_n)$, again monotonicity of A_n results in the inequality

$$\begin{aligned} 0 & \leq \langle x_1 + x_{n-1} - z_{n-1} - x_n - L^*(\gamma Lx_1 - v), x_n - \bar{x}_n \rangle \\ & \quad - \langle \bar{x}_1 + \bar{x}_{n-1} - \bar{z}_{n-1} - \bar{x}_n - L^*(\gamma L\bar{x}_1 - \bar{v}), x_n - \bar{x}_n \rangle \\ & = \langle (x_{n-1} - z_{n-1}) - (\bar{x}_{n-1} - \bar{z}_{n-1}), x_n - \bar{x}_n \rangle + \langle (x_1 - \bar{x}_1) - (x_n - \bar{x}_n), x_n - \bar{x}_n \rangle \\ & \quad - \langle \gamma(Lx_1 - L\bar{x}_1) - (v - \bar{v}), Lx_n - L\bar{x}_n \rangle \\ & = \langle (x_{n-1} - z_{n-1}) - (\bar{x}_{n-1} - \bar{z}_{n-1}), x_n - \bar{x}_{n-1} \rangle + \langle (x_1 - \bar{x}_1) - (x_n - \bar{x}_n), x_n - \bar{x}_n \rangle \\ & \quad + \langle (x_{n-1} - z_{n-1}) - (\bar{x}_{n-1} - \bar{z}_{n-1}), \bar{x}_{n-1} - \bar{x}_n \rangle \\ & \quad - \langle \gamma(Lx_1 - L\bar{x}_1) - (v - \bar{v}), Lx_n - L\bar{x}_n \rangle. \end{aligned} \quad (12)$$

Finally, we have $\gamma L(x_1 + x_n) - v - \gamma y \in B(y)$ and $\gamma L(\bar{x}_1 + \bar{x}_n) - \bar{v} - \gamma \bar{y} \in B(\bar{y})$, so by monotonicity of B we get

$$0 \leq \langle (\gamma L(x_1 + x_n) - v - \gamma y) - (\gamma L(\bar{x}_1 + \bar{x}_n) - \bar{v} - \gamma \bar{y}), y - \bar{y} \rangle. \quad (13)$$

Summing together (10)-(13) and rearranging, yields

$$\begin{aligned}
0 \leq & \sum_{i=1}^{n-1} \langle (x_i - x_{i+1}) - (\bar{x}_i - \bar{x}_{i+1}), z_i - \bar{z}_i \rangle + \sum_{i=1}^{n-1} \langle (x_i - \bar{x}_i) - (x_{i+1} - \bar{x}_{i+1}), \bar{x}_i - x_i \rangle \\
& + \langle (x_1 - \bar{x}_1) - (x_n - \bar{x}_n), x_n - \bar{x}_n \rangle + \langle (Lx_n - L\bar{x}_n) - (y - \bar{y}), v - \bar{v} \rangle \\
& + \gamma \langle (L(x_1 + x_n) - L(\bar{x}_1 + \bar{x}_n)) - (y - \bar{y}), y - \bar{y} \rangle \\
& - \gamma \langle Lx_1 - L\bar{x}_1, Lx_n - L\bar{x}_n \rangle.
\end{aligned} \tag{14}$$

The sums in (14) can be written, respectively, as

$$\begin{aligned}
& \sum_{i=1}^{n-1} \langle (x_i - x_{i+1}) - (\bar{x}_i - \bar{x}_{i+1}), z_i - \bar{z}_i \rangle \\
& = \frac{1}{\lambda} \sum_{i=1}^{n-1} \langle (z_i - z_i^+) - (\bar{z}_i - \bar{z}_i^+), z_i - \bar{z}_i \rangle \\
& = \frac{1}{\lambda} \langle (\mathbf{z} - \mathbf{z}^+) - (\bar{\mathbf{z}} - \bar{\mathbf{z}}^+), \mathbf{z} - \bar{\mathbf{z}} \rangle \\
& = \frac{1}{2\lambda} (\|(\mathbf{z} - \mathbf{z}^+) - (\bar{\mathbf{z}} - \bar{\mathbf{z}}^+)\|^2 - \|\mathbf{z}^+ - \bar{\mathbf{z}}^+\|^2 + \|\mathbf{z} - \bar{\mathbf{z}}\|^2),
\end{aligned} \tag{15}$$

and

$$\begin{aligned}
& \sum_{i=1}^{n-1} \langle (x_i - \bar{x}_i) - (x_{i+1} - \bar{x}_{i+1}), \bar{x}_i - x_i \rangle \\
& = \frac{1}{2} \sum_{i=1}^{n-1} (\|x_{i+1} - \bar{x}_{i+1}\|^2 - \|x_i - \bar{x}_i\|^2 - \|(x_i - x_{i+1}) - (\bar{x}_i - \bar{x}_{i+1})\|^2) \\
& = \frac{1}{2} \left(\|x_n - \bar{x}_n\|^2 - \|x_1 - \bar{x}_1\|^2 - \frac{1}{\lambda^2} \sum_{i=1}^{n-1} \|(z_i - z_i^+) - (\bar{z}_i - \bar{z}_i^+)\|^2 \right) \\
& = \frac{1}{2} \left(\|x_n - \bar{x}_n\|^2 - \|x_1 - \bar{x}_1\|^2 - \frac{1}{\lambda^2} \|(\mathbf{z} - \mathbf{z}^+) - (\bar{\mathbf{z}} - \bar{\mathbf{z}}^+)\|^2 \right).
\end{aligned} \tag{16}$$

The third term in (14), becomes

$$\langle (x_1 - \bar{x}_1) - (x_n - \bar{x}_n), x_n - \bar{x}_n \rangle = \frac{1}{2} (\|x_1 - \bar{x}_1\|^2 - \|x_n - \bar{x}_n\|^2 - \|(x_1 - \bar{x}_1) - (x_n - \bar{x}_n)\|^2), \tag{17}$$

while the fourth term yields

$$\begin{aligned}
& \langle (Lx_n - L\bar{x}_n) - (y - \bar{y}), v - \bar{v} \rangle \\
& = \frac{1}{\gamma\lambda} \langle (v - v^+) - (\bar{v} - \bar{v}^+), v - \bar{v} \rangle \\
& = \frac{1}{2\gamma\lambda} (\|(v - v^+) - (\bar{v} - \bar{v}^+)\|^2 - \|v^+ - \bar{v}^+\|^2 + \|v - \bar{v}\|^2).
\end{aligned} \tag{18}$$

Lastly, making use of the Cauchy–Schwarz and Young’s inequalities, the second last term

of (14) gives

$$\begin{aligned}
& \gamma \langle (L(x_1 + x_n) - L(\bar{x}_1 + \bar{x}_n)) - (y - \bar{y}), y - \bar{y} \rangle \\
&= \gamma (\langle Lx_1 - L\bar{x}_1, y - \bar{y} \rangle + \langle (Lx_n - L\bar{x}_n) - (y - \bar{y}), y - \bar{y} \rangle) \\
&= \frac{\gamma}{2} (\|Lx_n - L\bar{x}_n\|^2 - \|(Lx_n - L\bar{x}_n) - (y - \bar{y})\|^2 - \|y - \bar{y}\|^2) \\
&\quad + \gamma \langle Lx_1 - L\bar{x}_1, y - \bar{y} \rangle \\
&\leq \frac{\gamma}{2} \left(\|Lx_n - L\bar{x}_n\|^2 - \frac{1}{\gamma^2 \lambda^2} \|(v - v^+) - (\bar{v} - \bar{v}^+)\|^2 - \|y - \bar{y}\|^2 \right) \\
&\quad + \frac{\gamma}{2} \|Lx_1 - L\bar{x}_1\|^2 + \frac{\gamma}{2} \|y - \bar{y}\|^2 \\
&= \frac{\gamma}{2} \|Lx_1 - L\bar{x}_1\|^2 + \frac{\gamma}{2} \|Lx_n - L\bar{x}_n\|^2 - \frac{1}{2\gamma\lambda^2} \|(v - v^+) - (\bar{v} - \bar{v}^+)\|^2,
\end{aligned} \tag{19}$$

while the last term can be rearranged as follows

$$\begin{aligned}
& -\gamma \langle Lx_1 - L\bar{x}_1, Lx_n - L\bar{x}_n \rangle \\
&= \frac{\gamma}{2} (\|L(x_1 - x_n) - L(\bar{x}_1 - \bar{x}_n)\|^2 - \|Lx_1 - L\bar{x}_1\|^2 - \|Lx_n - L\bar{x}_n\|^2).
\end{aligned} \tag{20}$$

Summing together (19) and (20) and using the Lipschitz continuity of L , we get

$$\begin{aligned}
& \gamma \langle (L(x_1 + x_n) - L(\bar{x}_1 + \bar{x}_n)) - (y - \bar{y}), y - \bar{y} \rangle - \gamma \langle Lx_1 - L\bar{x}_1, Lx_n - L\bar{x}_n \rangle \\
&= \frac{\gamma}{2} \|L(x_1 - x_n) - L(\bar{x}_1 - \bar{x}_n)\|^2 - \frac{1}{2\gamma\lambda^2} \|(v - v^+) - (\bar{v} - \bar{v}^+)\|^2 \\
&\leq \frac{\gamma \|L\|^2}{2} \|(x_1 - x_n) - (\bar{x}_1 - \bar{x}_n)\|^2 - \frac{1}{2\gamma\lambda^2} \|(v - v^+) - (\bar{v} - \bar{v}^+)\|^2.
\end{aligned} \tag{21}$$

Multiplying (14) by 2λ and substituting equations (15)-(21), we obtain the final inequality

$$\begin{aligned}
& \|z^+ - \bar{z}^+\|^2 + \left(\frac{1}{\lambda} - 1 \right) \left(\|(z - z^+) - (\bar{z} - \bar{z}^+)\|^2 + \frac{1}{\gamma} \|(v - v^+) - (\bar{v} - \bar{v}^+)\|^2 \right) \\
&+ \frac{1}{\gamma} \|v^+ - \bar{v}^+\|^2 + \lambda (1 - \gamma \|L\|^2) \|(x_1 - x_n) - (\bar{x}_1 - \bar{x}_n)\|^2 \\
&\leq \|z - \bar{z}\|^2 + \frac{1}{\gamma} \|v - \bar{v}\|^2.
\end{aligned}$$

To complete the proof, just note that

$$\begin{aligned}
\lambda(x_1 - x_n) - \lambda(\bar{x}_1 - \bar{x}_n) &= \lambda \sum_{i=1}^{n-1} (x_i - x_{i+1}) - \lambda \sum_{i=1}^{n-1} (\bar{x}_i - \bar{x}_{i+1}) \\
&= \sum_{i=1}^{n-1} (z_i - z_i^+) - \sum_{i=1}^{n-1} (\bar{z}_i - \bar{z}_i^+),
\end{aligned}$$

from where (9) finally follows. \square

Next we state our main result, which establishes the convergence of the iterative algorithm defined by the operator T in (6)-(7).

Theorem 3.3. Let $n \geq 2$, let $L : \mathcal{H} \rightarrow \mathcal{G}$ be a bounded linear operator and let $A_1, \dots, A_n : \mathcal{H} \rightrightarrows \mathcal{H}$ and $B : \mathcal{G} \rightrightarrows \mathcal{G}$ be maximally monotone operators with $\text{zer}(\sum_{i=1}^n A_i + L^*BL) \neq \emptyset$. Further, let $\lambda \in]0, 1[$ and $\gamma \in]0, \frac{1}{\|L\|^2}]$. Given an initial point $(\mathbf{z}^0, v^0) = (z_1^0, \dots, z_{n-1}^0, v^0) \in \mathcal{H}^{n-1} \times \mathcal{G}$, consider the sequences given by

$$\begin{pmatrix} \mathbf{z}^{k+1} \\ v^{k+1} \end{pmatrix} = \begin{pmatrix} \mathbf{z}^k \\ v^k \end{pmatrix} + \lambda \begin{pmatrix} x_2^k - x_1^k \\ x_3^k - x_2^k \\ \vdots \\ x_n^k - x_{n-1}^k \\ \gamma(y^k - Lx_n^k) \end{pmatrix} \quad \forall k \geq 0, \quad (22)$$

with

$$\begin{cases} x_1^k = J_{A_1}(z_1^k), \\ x_i^k = J_{A_i}(z_i^k + x_{i-1}^k - z_{i-1}^k), \quad \forall i \in \llbracket 2, n-1 \rrbracket, \\ x_n^k = J_{A_n}(x_1^k + x_{n-1}^k - z_{n-1}^k - L^*(\gamma Lx_1^k - v^k)), \\ y^k = J_{B/\gamma} \left(L(x_1^k + x_n^k) - \frac{v^k}{\gamma} \right). \end{cases} \quad (23)$$

Then the following statements hold.

- (i) The sequence $(\mathbf{z}^k, v^k)_{k \in \mathbb{N}}$ converges weakly to a point $(\bar{\mathbf{z}}, \bar{v}) \in \text{Fix } T$.
- (ii) The sequence $(x_1^k, \dots, x_n^k, y^k)_{k \in \mathbb{N}}$ converges weakly to $(\bar{x}, \dots, \bar{x}, L\bar{x})$ with $\bar{x} \in \mathcal{P}$.
- (iii) The sequence $(\gamma Lx_i^k - v^k)_{k \in \mathbb{N}}$ converges weakly to $\gamma L\bar{x} - \bar{v} \in \mathcal{D}$, for all $i \in \llbracket 1, n \rrbracket$.

Proof. (i) The sequence in (22) is the fixed point iteration generated as

$$\begin{pmatrix} \mathbf{z}^{k+1} \\ v^{k+1} \end{pmatrix} = T \begin{pmatrix} \mathbf{z}^k \\ v^k \end{pmatrix} \quad \forall k \geq 0.$$

Since $\lambda \in]0, 1[$ and $\gamma \in]0, \|L\|^{-2}]$, T is averaged nonexpansive by Lemma 3.2 and, moreover, $\text{Fix } T = \emptyset$, due to $\mathbf{Z} \neq \emptyset$ and Lemma 3.1(i). Then, by [2, Theorem 5.15] the sequence $(\mathbf{z}^k, v^k)_{k \in \mathbb{N}}$ converges weakly to a point $(\bar{\mathbf{z}}, \bar{v}) \in \text{Fix } T$ and $\lim_{k \rightarrow \infty} \|(\mathbf{z}^{k+1}, v^{k+1}) - (\mathbf{z}^k, v^k)\|_\gamma = 0$.

(ii) From (i), the sequence $(\mathbf{z}^k, v^k)_{k \in \mathbb{N}}$ is bounded. Then, nonexpansivity of the resolvents and boundedness of the linear operator L imply that the sequence $(\mathbf{x}^k, y^k)_{k \in \mathbb{N}} = (x_1^k, \dots, x_n^k, y^k)_{k \in \mathbb{N}}$ is also bounded. Further, the fact that $(\mathbf{z}^{k+1}, v^{k+1})_{k \in \mathbb{N}} - (\mathbf{z}^k, v^k)_{k \in \mathbb{N}} \rightarrow 0$, as $k \rightarrow \infty$, implies by (22) that

$$y^k - Lx_n^k \rightarrow 0 \text{ and } x_{i+1}^k - x_i^k \rightarrow 0, \text{ for all } i \in \llbracket 1, n-2 \rrbracket. \quad (24)$$

Next, by making use of the definition of resolvents and (23), we can write

$$C \begin{pmatrix} z_1^k - x_1^k \\ (z_2^k - x_2^k) - (z_1^k - x_1^k) \\ \vdots \\ (z_{n-1}^k - x_{n-1}^k) - (z_{n-2}^k - x_{n-2}^k) \\ x_n^k \\ \gamma(L(x_1^k + x_n^k) - y^k) - v^k \end{pmatrix} \ni \begin{pmatrix} x_1^k - x_n^k \\ x_2^k - x_n^k \\ \vdots \\ x_{n-1}^k - x_n^k \\ x_1^k - x_n^k + \gamma L^*(Lx_n^k - y^k) \\ y^k - Lx_n^k \end{pmatrix}, \quad (25)$$

where the operator $C: \mathcal{H}^n \times \mathcal{G} \rightrightarrows \mathcal{H}^n \times \mathcal{G}$ is given by

$$C := \begin{pmatrix} A_1^{-1} \\ A_2^{-1} \\ \vdots \\ A_{n-1}^{-1} \\ A_n \\ B^{-1} \end{pmatrix} + \begin{pmatrix} 0 & 0 & \dots & 0 & -\text{Id} & 0 \\ 0 & 0 & \dots & 0 & -\text{Id} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & -\text{Id} & 0 \\ \text{Id} & \text{Id} & \dots & \text{Id} & 0 & L^* \\ 0 & 0 & \dots & 0 & -L & 0 \end{pmatrix}. \quad (26)$$

The operator C is maximally monotone as the sum of a maximally monotone operator and a skew symmetric linear operator (see, e.g., [2, Corollary 25.5 (i) & Example 20.35]). Thus, the graph of C is sequentially closed in the weak-strong topology, by demiclosedness of maximally monotone operators [2, Corollary 20.38].

Now, let $(\bar{\mathbf{x}}, \bar{y})$ be a weak sequential cluster point of $(\mathbf{x}^k, y^k)_{k \in \mathbb{N}}$. Due to (24), $\bar{\mathbf{x}}$ is of the form $\bar{\mathbf{x}} = (\bar{x}, \dots, \bar{x}) \in \mathcal{H}^n$ and $\bar{y} = L\bar{x}$. Taking the limit along a subsequence of $(\mathbf{x}^k, y^k)_{k \in \mathbb{N}}$ which converges weakly to $(\bar{\mathbf{x}}, \bar{y})$ and using demiclosedness of C , equations (25) and (26) yield the expression

$$\begin{cases} \bar{z}_1 - \bar{x} \in A_1(\bar{x}), \\ \bar{z}_i - \bar{z}_{i-1} \in A_i(\bar{x}), \quad \forall i \in \llbracket 2, n-1 \rrbracket, \\ \bar{x} - \bar{z}_{n-1} - L^*(\gamma L\bar{x} - \bar{v}) \in A_n(\bar{x}), \\ \gamma L\bar{x} - \bar{v} \in B(L\bar{x}), \end{cases}$$

which, by summing the first n equations, implies that $(\bar{x}, \gamma L\bar{x} - \bar{v}) \in \mathbf{Z}$ with $\bar{x} = J_{A_1}(\bar{z}_1)$. In particular, we have shown that $(\bar{\mathbf{x}}, \bar{y})$ is directly obtained from $\bar{\mathbf{z}}$, implying that it is the unique weak sequential cluster point of the bounded sequence $(\mathbf{x}^k, y^k)_{k \in \mathbb{N}}$. Thus, the full sequence converges weakly to this point.

(iii) From (i)-(ii), for all $i \in \llbracket 1, n \rrbracket$, we deduce that the sequence $(\gamma Lx_i^k - v^k)_{k \in \mathbb{N}}$ weakly converges to $\gamma L\bar{x} - \bar{v}$, which belongs to \mathcal{D} since $(\bar{x}, \gamma L\bar{x} - \bar{v}) \in \mathbf{Z}$. \square

Remark 3.4 (Malitsky–Tam resolvent splitting [16] as a special case). Consider Problem (4)-(5) in the particular case in which $L = \text{Id}$. Then, $B: \mathcal{H} \rightrightarrows \mathcal{H}$ and equation (4) becomes the classical monotone inclusion problem with $(n+1)$ -operators. Furthermore, by setting $\gamma = 1$ in Theorem 3.3, it is straightforward to see that the sequences in (22)-(23) yield the Malitsky–Tam resolvent splitting with minimal lifting for $(n+1)$ -operators.

Remark 3.5 (On the parameter γ in the definition of the norm $\|\cdot\|_\gamma$). In Lemma 3.2, we proved that the operator T is λ -averaged nonexpansive with respect to the norm $\|\cdot\|_\gamma$ induced by the scalar product defined in (8). Although the use of this norm did not require detours from the usual procedure to prove convergence of the fixed point algorithm in Theorem 3.3, it may numerically affect the performance of the algorithm. To give an intuition about this, consider the norm of the *sequence of residuals* $(\|(\mathbf{z}^{k+1}, v^{k+1}) - (\mathbf{z}^k, v^k)\|_\gamma)_{k \in \mathbb{N}}$, which converges to 0 as the algorithm reaches a fixed point, and note that we have

$$\left\| (\mathbf{z}^{k+1}, v^{k+1}) - (\mathbf{z}^k, v^k) \right\|_\gamma^2 = \|\mathbf{z}^{k+1} - \mathbf{z}^k\|^2 + \frac{1}{\gamma} \|v^{k+1} - v^k\|^2 \quad \forall k \geq 0.$$

Lemma 3.2 implies that this sequence is monotone decreasing, but if γ is very small, the weight of the sequence of dual variables $(v^{k+1} - v^k)_{k \in \mathbb{N}}$ in the norm would be much larger than the one of the sequence of primal variables $(\mathbf{z}^{k+1} - \mathbf{z}^k)_{k \in \mathbb{N}}$, so a small decrease in the value of $\|v^{k+1} - v^k\|$ will readily imply a decrease of the norm of the sequence of residuals even if $\|\mathbf{z}^{k+1} - \mathbf{z}^k\|$ does not diminish much. Because of that, a larger number of iterations

might be needed to achieve convergence of the primal sequence, which can slow down the overall convergence of the algorithm. Nonetheless, it is possible to perform some sort of pre-conditioning to prevent from having a large constant in the definition of the norm. We will further comment on this in the numerical experiment in Section 4.

The case with multiple linear compositions

A standard product space reformulation permits to extend our method to the more general inclusion Problem 1.1, which has finitely many linearly composed maximally monotone operators. We detail this in the following corollary, while the resulting scheme is displayed in Algorithm 1.

Algorithm 1 Primal-dual splitting for Problem 1.1 with $(n-1, m)$ -lifting, with $n \geq 2$.

Require: $\lambda \in]0, 1[$ and $\gamma \in \left]0, 1/\sum_{j=1}^m \|L_j\|^2\right]$.

- 1: Choose $\mathbf{z}^0 = (z_1^0, \dots, z_{n-1}^0) \in \mathcal{H}^{n-1}$ and $\mathbf{v}^0 = (v_1^0, \dots, v_m^0) \in \mathcal{G}_1 \times \dots \times \mathcal{G}_m$.
- 2: **for** $k = 0, 1, \dots$ **do**
- 3: Compute

$$\begin{pmatrix} \mathbf{z}^{k+1} \\ \mathbf{v}^{k+1} \end{pmatrix} = \begin{pmatrix} \mathbf{z}^k \\ \mathbf{v}^k \end{pmatrix} + \lambda \begin{pmatrix} x_2^k - x_1^k \\ x_3^k - x_2^k \\ \vdots \\ x_n^k - x_{n-1}^k \\ \gamma(y_1^k - L_1 x_n^k) \\ \vdots \\ \gamma(y_m^k - L_m x_n^k) \end{pmatrix}, \quad (27)$$

with $\mathbf{x}^k = (x_1^k, \dots, x_n^k) \in \mathcal{H}^n$ and $\mathbf{y}^k = (y_1^k, \dots, y_m^k) \in \mathcal{G}_1 \times \dots \times \mathcal{G}_m$ computed as

$$\begin{cases} x_1^k = J_{A_1}(z_1^k), \\ x_i^k = J_{A_i}(z_i^k + x_{i-1}^k - z_{i-1}^k) \quad \forall i \in \llbracket 2, n-1 \rrbracket, \\ x_n^k = J_{A_n} \left(x_1^k + x_{n-1}^k - z_{n-1}^k - \sum_{j=1}^m L_j^*(\gamma L_j x_1^k - v_j^k) \right), \\ y_j^k = J_{B_j/\gamma} \left(L_j(x_1^k + x_n^k) - \frac{v_j^k}{\gamma} \right) \quad \forall j \in \llbracket 1, m \rrbracket. \end{cases} \quad (28)$$

4: **end for**

Corollary 3.6. *Let $n \geq 2$ and assume that Problem 1.1 has a solution. Let $\lambda \in]0, 1[$ and $\gamma \in \left]0, 1/\sum_{j=1}^m \|L_j\|^2\right]$. Given some initial points $\mathbf{z}^0 = (z_1, \dots, z_{n-1}) \in \mathcal{H}^{n-1}$ and $\mathbf{v}^0 = (v_1^0, \dots, v_m^0) \in \mathcal{G}_1 \times \dots \times \mathcal{G}_m$, consider the sequences $(\mathbf{z}^k, \mathbf{v}^k)_{k \in \mathbb{N}}$ and $(\mathbf{x}^k, \mathbf{y}^k)_{k \in \mathbb{N}}$ generated by Algorithm 1. Then, the following assertions hold:*

- (i) *The sequence $(\mathbf{z}^k, \mathbf{v}^k)_{k \in \mathbb{N}}$ converges weakly to a point $(\bar{\mathbf{z}}, \bar{\mathbf{v}}) \in \mathcal{H}^{n-1} \times \mathcal{G}_1 \times \dots \times \mathcal{G}_m$.*
- (ii) *The sequence $(x_1^k, \dots, x_n^k, y_1^k, \dots, y_m^k)_{k \in \mathbb{N}}$ converges weakly to $(\bar{x}, \dots, \bar{x}, L_1 \bar{x}, \dots, L_m \bar{x})$ with $\bar{x} \in \mathcal{H}$ solving the primal inclusion (1).*
- (iii) *For all $i \in \llbracket 1, n \rrbracket$, the sequence $(\gamma L_1 x_i^k - v_1^k, \dots, \gamma L_m x_i^k - v_m^k)_{k \in \mathbb{N}}$ converges weakly to $(\gamma L_1 \bar{x} - \bar{v}_1, \dots, \gamma L_m \bar{x} - \bar{v}_m)$, which solves the dual inclusion (2).*

Proof. Just note that Problem 1.1 can be reformulated as an instance of Problem (4)-(5) by replacing B by the operator $\mathbf{B} : \mathcal{G}_1 \times \cdots \times \mathcal{G}_m \rightrightarrows \mathcal{G}_1 \times \cdots \times \mathcal{G}_m$ defined as the cartesian product $\mathbf{B} := \times_{j=1}^m B_j$ and L by the linear operator $\mathbf{L} := \times_{j=1}^m L_j$. In particular, $\|\mathbf{L}\|^2 = \sum_{j=1}^n \|L_j\|^2$ and its adjoint operator is $\mathbf{L}^* : \mathcal{G}_1 \times \cdots \times \mathcal{G}_m \rightarrow \mathcal{H} : (v_1, \dots, v_m) \rightarrow \sum_{j=1}^m L_j^* v_j$. Hence, the result follows by considering the averaged nonexpansive operator T in (6) for this choice of operators and applying Theorem 3.3. \square

Minimality for primal-dual parametrized resolvent splitting

We begin by extending the definition of fixed point encoding to englobe primal-dual problems. As in Section 2.1, we denote by T a fixed point operator and by S a solution operator, both parametrized by the maximally monotone operators as well as the linear and adjoint operators appearing in Problem 1.1.

Definition 3.7 (Fixed point encoding). *A pair of operators (T, S) is a fixed point encoding for Problem 1.1 if, for all particular instance of the problem,*

$$\text{Fix } T \neq \emptyset \iff \text{zer} \left(\sum_{i=1}^n A_i + \sum_{j=1}^m L_j^* B_j L_j \right) \neq \emptyset \text{ and } \mathbf{w} \in \text{Fix } T \implies S(\mathbf{w}) \in \mathbf{Z},$$

where we recall that \mathbf{Z} denotes the set of primal-dual solutions of the problem.

When talking about lifting for primal-dual problems, the need to distinguish between variables in the space of primal solutions and dual solutions arises. This motivates the following definition.

Definition 3.8. *Primal-dual lifting] Let $d, f \in \mathbb{Z}_+$. A fixed point encoding (T, S) is a (d, f) -fold lifting for Problem 1.1 if*

$$T : \mathcal{H}^d \times \mathcal{G}_1^{f_1} \times \cdots \times \mathcal{G}_m^{f_m} \rightarrow \mathcal{H}^d \times \mathcal{G}_1^{f_1} \times \cdots \times \mathcal{G}_m^{f_m}$$

and

$$S : \mathcal{H}^d \times \mathcal{G}_1^{f_1} \times \cdots \times \mathcal{G}_m^{f_m} \rightarrow \mathcal{H} \times \mathcal{G}_1 \times \cdots \times \mathcal{G}_m,$$

where $f_j \geq 0$ for all $j \in \llbracket 1, m \rrbracket$ and $f = \sum_{j=1}^m f_j$. We adopt the convention that the space \mathcal{G}_j vanishes from the equation when $f_j = 0$.

The need to control the Lipschitz constants of the linear operators requires the introduction of parameters in the resolvents of the maximally monotone operators. This motivates the definition of parametrized resolvent splitting introduced in Section 2.1 and which we now adapt to primal-dual splitting algorithms.

Definition 3.9 (Primal-dual parametrized resolvent splitting). *A fixed point encoding (T, S) for Problem 1.1 is a primal-dual parametrized resolvent splitting if, for all particular instance of the problem, there is a finite procedure that evaluates T and S at a given point which only uses vector addition, scalar multiplication and the parametrized resolvents of A_1, \dots, A_n and B_1, \dots, B_m .*

Definition 3.10 (Frugality). *A primal-dual parametrized resolvent splitting (T, S) for Problem 1.1 is frugal if, in addition, each of the parametrized resolvents of A_1, \dots, A_n and B_1, \dots, B_m is used exactly once.*

Remark 3.11 (On the absence of restrictions on the evaluation of the linear operators). Since in the finite case, a forward evaluation of a linear operator is computationally equivalent to performing vector addition and scalar multiplication, this suggests that for practical applications there is no computational need to control the number of evaluations of the linear operators in the definition of frugality.

Example 3.12. Let $n \geq 2$ and consider Problem 1.1. Let $T : \mathcal{H}^{n-1} \times \mathcal{G}_1 \times \cdots \times \mathcal{G}_m \rightarrow \mathcal{H}^{n-1} \times \mathcal{G}_1 \times \cdots \times \mathcal{G}_m$ be the operator defined in (6) by setting $B := \times_{j=1}^m B_j$ and $L := \times_{j=1}^m L_j$. Let $S : \mathcal{H}^{n-1} \times \mathcal{G}_1 \times \cdots \times \mathcal{G}_m \rightarrow \mathcal{H} \times \mathcal{G}_1 \times \cdots \times \mathcal{G}_m$ be defined as

$$S \begin{pmatrix} \mathbf{z} \\ \mathbf{v} \end{pmatrix} := \begin{pmatrix} J_{A_1}(z_1) \\ \gamma L_1 J_{A_1}(z_1) - v_1 \\ \vdots \\ \gamma L_m J_{A_1}(z_1) - v_m \end{pmatrix}.$$

Then, by Lemma 3.1 and Corollary 3.6, the pair (T, S) is a frugal parametrized resolvent splitting with $(n-1, m)$ -fold lifting.

The following result shows that the lifting of Algorithm 1 is minimal among frugal primal-dual parametrized resolvent splitting algorithms with m dual variables.

Theorem 3.13 (Minimality theorem for frugal parametrized splitting). *Let (T, S) be a frugal primal-dual parametrized resolvent splitting for Problem 1.1 with (d, m) -fold lifting. Then, if $n \geq 2$, necessarily $d \geq n-1$.*

Proof. By way of contradiction, let (T, S) be a frugal primal-dual parametrized resolvent splitting for Problem 1.1 with (d, m) fold lifting and $d < n-1$. Consider the instance of the problem in which $L_j = \text{Id} : \mathcal{H} \rightarrow \mathcal{H}$ for all $j \in \llbracket 1, m \rrbracket$. Then, Problem 1.1 becomes the classical monotone inclusion problem with $n+m$ operators and (T, S) is a frugal resolvent splitting with $(d+m)$ -fold lifting for such problem with $d+m < n+m-1$, which contradicts Theorem 2.10. \square

Finally, we conclude this section by highlighting that Algorithm 1 can be applied with $n < 2$, by setting $A_i = 0$ if required. However, a reduction in the lifting is not obtained in this case.

Remark 3.14 (Algorithm 1 when $n \leq 1$). Consider Algorithm 1 applied to Problem 1.1 with $n \leq 1$. We distinguish the two cases:

- (i) If $n = 1$, then Algorithm 1 has $(1, m)$ -lifting. Indeed, equations (27) and (28) become

$$\begin{pmatrix} z^{k+1} \\ \mathbf{v}^{k+1} \end{pmatrix} = \begin{pmatrix} z^k \\ \mathbf{v}^k \end{pmatrix} + \lambda \begin{pmatrix} x^k - z^k \\ \gamma(y_1^k - L_1 x^k) \\ \vdots \\ \gamma(y_m^k - L_m x^k) \end{pmatrix} \quad \forall k \geq 0, \quad (29)$$

and

$$\begin{cases} x^k = J_{A_1} \left(z^k - \sum_{j=1}^m L_j^* (\gamma L_j z^k - v_j^k) \right), \\ y_j^k = J_{B_j/\gamma} \left(L_j (z^k + x^k) - \frac{v_j^k}{\gamma} \right), \quad \forall j \in \llbracket 1, m \rrbracket, \end{cases} \quad (30)$$

respectively. This means that, in contrast with what happens when $n \geq 2$, there is no reduction in the lifting with respect to the number of operators involved.

- (ii) If $n = 0$, the scheme also has $(1, m)$ -lifting. In fact, the scheme is the same as in the previous case but substituting J_{A_1} by Id in (30). Note that this is also the lifting obtained by the already known algorithms in the literature applied to this case.

4 Numerical experiments

In this section, we test our algorithm for solving an ill-conditioned linear inverse problem which arises in image deblurring and denoising. Let $b \in \mathbb{R}^n$ be an observed blurred and noisy image of size $M \times N$, with $n = MN$ for grayscale and $n = 3MN$ for color images, and denote by $A \in \mathbb{R}^{n \times n}$ the blur operator. The problem can be tackled by means of the regularized convex non-differentiable problem

$$\inf_{s \in \mathbb{R}^n} \{ \|As - b\|_1 + \alpha_1 \|Ws\|_1 + \alpha_2 TV(s) + \delta_{[0,1]^n}(s) \}, \quad (31)$$

where $\alpha_1, \alpha_2 > 0$ are regularization parameters, $\delta_{[0,1]^n}$ denotes the indicator function of the set $[0, 1]^n$, $TV : \mathbb{R}^n \rightarrow \mathbb{R}$ is the discrete isotropic total variation function and W is the linear operator given by the normalized *nonstandard Haar transform* [21].

Recalling Remark 3.5, it is of interest to consider a mechanism which allows tuning the parameter γ appearing in the definition of the norm given by the inner product in (8) to an appropriate value. To this aim, we perform in (31) a change of variable of the form $s = \mu x$, with $\mu > 0$, and instead handle the problem

$$\inf_{x \in \mathbb{R}^n} \left\{ \mu \left\| Ax - \frac{b}{\mu} \right\|_1 + \alpha_1 \mu \|Wx\|_1 + \alpha_2 TV(\mu x) + \delta_{[0,1/\mu]^n}(x) \right\}. \quad (32)$$

Below we will see the way in which the choice of μ can help setting a suitable parameter γ .

The minimization problem in (32) can be modeled as a composite monotone inclusion problem. For this, define the operator $L : \mathbb{R}^n \rightarrow \mathbb{R}^n \times \mathbb{R}^n : x \rightarrow (L_1x, L_2x)$ where L_1 and L_2 are defined component-wise as

$$(L_1x)_{i,j} = \begin{cases} \frac{x_{i+1,j} - x_{i,j}}{\mu}, & \text{if } i < M, \\ 0, & \text{otherwise,} \end{cases} \quad \text{and } (L_2x)_{i,j} = \begin{cases} \frac{x_{i,j+1} - x_{i,j}}{\mu}, & \text{if } j < N, \\ 0, & \text{otherwise.} \end{cases} \quad (33)$$

Then the parametrized total variation function can be written as $TV(\mu \cdot) = \|L(\cdot)\|_\times$, with $\|(p, q)\|_\times := \sum_{i=1}^m \sum_{j=1}^n \sqrt{p_{i,j}^2 + q_{i,j}^2}$. Furthermore, an upper bound of the Lipschitz constant of L is given by $\|L\|^2 \leq 8\mu^2$ (see [9] for details).

By [2, Proposition 27.5], obtaining a solution to the following problem is equivalent to solving (32)

$$\text{find } x \in \text{zer} (N_{[0,1/\mu]^n} + W^* \circ \partial g_1 \circ W + A^* \circ \partial g_2 \circ A + L^* \circ \partial g_3 \circ L), \quad (34)$$

with $g_1 : \mathbb{R}^n \rightarrow \mathbb{R}$, $g_1(y) = \alpha_1 \mu \|y\|_1$, $g_2 : \mathbb{R}^n \rightarrow \mathbb{R}$, $g_2(y) = \mu \|y - b/\mu\|_1$, $g_3 : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$, $g_3(p, q) = \alpha_2 \|(p, q)\|_\times$, and $N_{[0,1/\mu]^n}$ the normal cone operator to the set $[0, 1/\mu]^n$. In order to implement Algorithm 1 for solving (34), we need the expression of the following resolvents and proximity operators. By [2, Proposition 23.25 (iii)], the second term in (34) is a maximally monotone operator and its resolvent can be expressed as

$$J_{W^* \circ \partial g_1 \circ W} = \text{Id} - W^* \circ (\text{Id} - \text{prox}_{g_1}) \circ W = \text{Id} - W^* \circ \text{prox}_{g_1^*} \circ W,$$

where $\text{prox}_g = J_{\partial g}$ denotes the proximity operator of a function g , and g_1^* is the conjugate function to g_1 , which is equal to the indicator function $\delta_{[-\alpha_1\mu, \alpha_1\mu]^n}$, and thus $\text{prox}_{g_1^*} =$

$P_{[-\alpha_1\mu, \alpha_1\mu]^n}$. Given $\sigma > 0$, the proximity operators of g_2 and g_3 are, respectively,

$$\text{prox}_{\sigma g_2}(x) = \frac{b}{\mu} + \text{prox}_{\sigma\mu\|\cdot\|_1}\left(x - \frac{b}{\mu}\right) = \frac{b}{\mu} + \text{sign}\left(x - \frac{b}{\mu}\right) \odot \left[\left|x - \frac{b}{\mu}\right| - \sigma\mu\right]_+,$$

where \odot denotes element-wise product and $[\cdot]_+$ and $|\cdot|$ are applied element-wise, and

$$\text{prox}_{\sigma g_3} = \text{Id} - \sigma \text{prox}_{\frac{1}{\sigma}g_3^*} \circ \frac{1}{\sigma} \text{Id} = \text{Id} - \sigma P_S \circ \frac{1}{\sigma} \text{Id},$$

since the conjugate function of g_3 is $g_3^* : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^n$, $g_3^* = \delta_S$, with the set S defined as

$$S := \left\{ (p, q) \in \mathbb{R}^n \times \mathbb{R}^n : \max_{1 \leq i \leq M, 1 \leq j \leq N} \sqrt{p_{i,j}^2 + q_{i,j}^2} \leq \alpha_2 \right\},$$

and the projection operator $P_S : \mathbb{R}^n \times \mathbb{R}^n \rightarrow S$ is given component-wise by

$$(p_{i,j}, q_{i,j}) \mapsto \alpha_2 \frac{(p_{i,j}, q_{i,j})}{\max\{\alpha_2, \sqrt{p_{i,j}^2 + q_{i,j}^2}\}}, \quad 1 \leq i \leq M, 1 \leq j \leq N.$$

Hence, when choosing $z^0 \in \mathbb{R}^n$, $v_1^0 \in \mathbb{R}^n$ and $v_2^0 \in \mathbb{R}^n \times \mathbb{R}^n$ as starting values, and letting $\lambda \in]0, 1[$ and $\gamma \in]0, 1/(\|A\|^2 + \|L\|^2)]$, the iterative scheme in Algorithm 1 becomes

$$\left\{ \begin{array}{l} x_1^k = P_{[0,1/\mu]^n}(z^k), \\ x_2^k = (\text{Id} - W^* \circ P_{[-\alpha_1\mu, \alpha_1\mu]^n} \circ W)(2x_1^k - z^k - A^*(\gamma Ax_1^k - v_1^k) - L^*(\gamma Lx_1^k - v_2^k)), \\ y_1^k = \frac{b}{\mu} + \text{prox}_{\frac{\mu}{\gamma}\|\cdot\|_1}\left(A(x_1^k + x_2^k) - \frac{v_1^k}{\gamma} - \frac{b}{\mu}\right), \\ y_2^k = \left(\text{Id} - \frac{1}{\gamma}P_S\right)\left(\gamma L(x_1^k + x_2^k) - v_2^k\right), \\ z^{k+1} = z^k + \lambda(x_2^k - x_1^k), \\ v_1^{k+1} = v_1^k + \lambda\gamma(y_1^k - Ax_2^k), \\ v_2^{k+1} = v_2^k + \lambda\gamma(y_2^k - Lx_2^k). \end{array} \right.$$

In our experiment, we replicate the problem in [5, Section 4.2], where an extensive comparison between different primal-dual algorithms is presented. Since the best performing algorithm is the Douglas–Rachford type primal-dual method in [5, Algorithm 3.1], we limit our comparison to this algorithm, whose detailed implementation is given in the cited work. We ran our experiments in Matlab, making use of the inbuilt functions `fspecial` and `imfilter` to define an operator A which is a Gaussian blur operator of size 9×9 with standard deviation 4 and reflexive boundary conditions. In particular, A verifies $\|A\| = 1$ and $A^* = A$. We employed as observed image b a picture taken at the Schönbrunn Palace Gardens (Vienna) subjected to the already specified blur followed by the addition of a zero-mean Gaussian noise with standard deviation 10^{-3} (see Figure 2). To test the influence on the performance of the picture size, we resized the original picture to different pixel resolutions (see Table 1).

When measuring the quality of the restored images, we use the *improvement in signal-to-noise-ratio* (ISNR), which is given by

$$\text{ISNR}_k = 10 \log_{10} \left(\frac{\|x - b\|^2}{\|x - x^k\|^2} \right),$$

where x and x^k are the original and the reconstructed image at iteration k , respectively. We tuned the regularization parameters in order to guarantee an adequate ISNR value for the restored images, setting $\alpha_1 := 0.005$ and $\alpha_2 := 0.009$.

We recall that the stepsize parameter γ of Algorithm 1 must be taken in the interval $\gamma \in]0, 1/(\|A\|^2 + \|L\|^2)] =]0, 1/(1 + 8\mu^2)]$. When $\mu = 1$ (i.e., we solve (31)), this interval is $]0, 0.111]$. In our numerical experiments we empirically observed that a very small stepsize negatively affects the performance of the algorithm, as mentioned in Remark 3.5. After testing different options, the most convenient one seems to be $\mu = 1/\sqrt{8}$, which implies making the Lipschitz constant of both linear operators in the problem equal to 1.

The initialization of each of the methods was the following:

- DR1([5, Algorithm 3.1]): starting points $x_0 = b$ and $(v_{1,0}, v_{2,0}, v_{3,0}) = (0, 0, 0)$, $\sigma_1 = 1$, $\sigma_2 = 0.05$, $\sigma_3 = 0.05$, $\tau = 1(\sigma_1 + \sigma_2 + 8\sigma_3)^{-1} - 0.01$, $\lambda_n = 1.5$ for all $n \in \mathbb{N}$.
- Algorithm 1 with $\mu = 1$: starting points $z_0 = b$ and $(v_1^0, v_2^0) = (0, 0)$, $\lambda = 0.99$ and $\gamma = 1/9$;
- Algorithm 1 with $\mu = 1/\sqrt{8}$: starting points $z_0 = b/\mu$ and $(v_1^0, v_2^0) = (0, 0)$, $\lambda = 0.99$ and $\gamma = 1/2$.

We performed 400 iterations of each of the algorithms and compared the values of the objective function in (32) and the ISNR with respect to the CPU time, which provides a more realistic comparison than iteration count, since DR1 has a higher computational cost per iteration than Algorithm 1. The tests were ran on a desktop of Intel Core i7-4770 CPU 3.40GHz with 32GB RAM, under Windows 10 (64-bit). The algorithms were ran 3 times, once for each of the RGB components of the picture. The evolution in CPU time of adding these 3 values of the objective function and those of the ISNR for the 640×768 -sized picture are represented in Figure 1, where we observe that Algorithm 1 with $\mu = 1/\sqrt{8}$ obtains slightly better values than those returned by DR1, but in significantly less time.

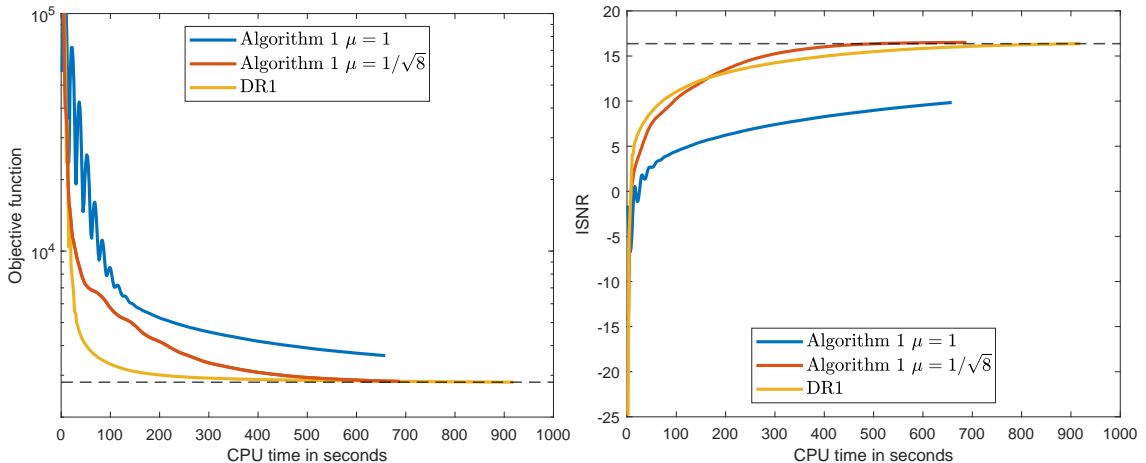


Figure 1: The evolution of the values of the objective function and of the ISNR in CPU time for 400 iterations of Algorithm 1 with $\mu = 1$ and $\mu = 1/\sqrt{8}$ and DR1, using the 640×768 pixels image displayed in Figure 2.

The restored images are presented in Figure 2. There is no much difference between the ones corresponding to Algorithm 1 with $\mu = 1/\sqrt{8}$ (bottom-middle) and DR1 (bottom-right), but a close look at the image obtained with Algorithm 1 with $\mu = 1$ permits to observe its worse quality. To show that this trend in the performance of the algorithms is

not affected by the image size, we present in Table 1 the results from running the algorithms on the same picture for five different pixel resolutions. Overall, we notice that the CPU time required for computing the 400 iterations is significantly lower for Algorithm 1, as expected. On average, DR1 required 45% more time than Algorithm 1 to compute the 400 iterations, independently of the size of the image. Regarding the parameter μ , Algorithm 1 with $\mu = 1$ is notably outperformed by the other two methods, making thus clear the influence that this parameter has on it. The function values obtained were slightly lower for DR1, while the ISNR was slightly lower for Algorithm 1 with $\mu = 1/\sqrt{8}$, which implies that both algorithms performed similarly with respect to the restored image quality.

Interpretation of the results of the experiments

The experimental results show that, after performing the same number of iterations, Algorithm 1 with $\mu = 1/\sqrt{8}$ obtains similar results in the function values and the measurement in the quality of the image recovery than those obtained by DR1, but in considerably less time. This decrease in the running time can be attributed to the reduction in the lifting of the operator. Although in the first iterations DR1 achieves a larger reduction of the objective function, the quality of the restored image is not sufficient, as assessed by the low ISNR values. On the other hand, Algorithm 1 with $\mu = 1$ can be discarded, as it obtains higher objective and lower ISNR values. Consequently, Algorithm 1 with $\mu = 1/\sqrt{8}$ is the preferable choice to address problem (31).

Resolution	Function values			ISNR			CPU time		
	$\mu = 1$	$\mu = 1/\sqrt{8}$	DR1	$\mu = 1$	$\mu = 1/\sqrt{8}$	DR1	$\mu = 1$	$\mu = 1/\sqrt{8}$	DR1
80×96	55.0	43.2	42.8	9.7	15.8	15.8	5.9	5.8	8.7
160×192	225.5	174.3	173.4	8.4	14.3	14.2	16.0	16.2	21.1
320×384	920.3	711.2	706.0	8.7	14.9	14.8	54.7	51.5	74.0
640×768	3630.3	2825.2	2804.5	9.8	16.5	16.4	294.4	293.1	465.4
1280×1536	13084.0	10360.0	10327.0	12.8	21.0	21.0	1654.2	1638.5	2349.6

Table 1: Results from running on the picture displayed in Figure 2 (for various pixel resolutions) 400 iterations of Algorithm 1 with $\mu = 1$ and $\mu = 1/\sqrt{8}$, and DR1.

5 Conclusions and open questions

In this work, we have considered the composite monotone inclusion problem together with its dual counterpart given by Problem 1.1. We have extended the definition of resolvent splitting given in [19] to encompass primal-dual algorithms and the inclusion of parameters in the resolvent and presented a definition of minimal lifting for frugal schemes of this form. We have proposed the first primal-dual algorithm which presents minimal lifting in this sense, and show its good performance with a numerical example.

To conclude, we outline possible directions for further research.

Establishing an optimal criterion for tuning the stepsize γ : We pointed out in Remark 3.5 the influence that the parameter γ can have in the performance of the algorithm. In Section 4 we presented a possibility for controlling this parameter, by making use of a change of variable which modifies the Lipschitz constants of the linear operators, and we empirically showed that it significantly affects the speed of performance of the algorithm. However, there is no guarantee that this strategy is optimal. It would be interesting to further investigate which is the best way for tuning the value of γ .

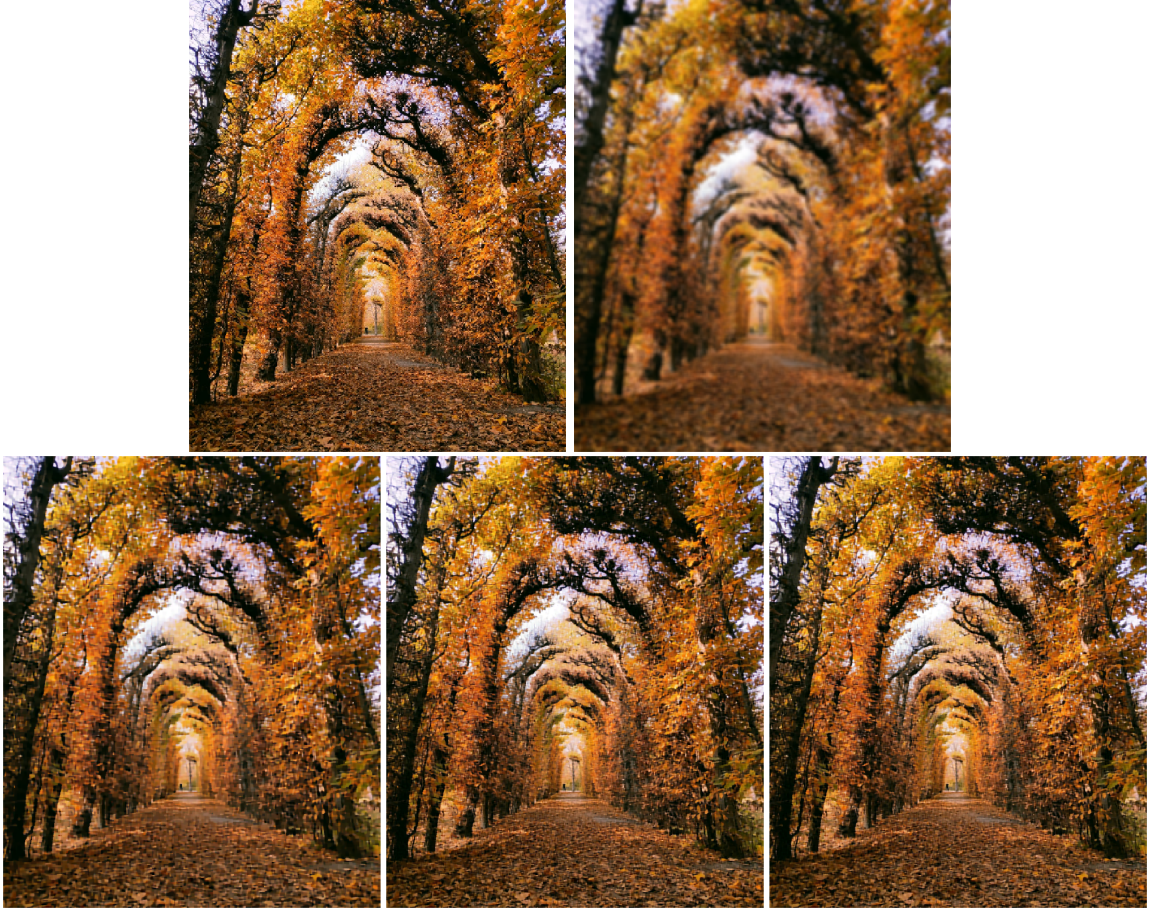


Figure 2: On the top, the original 640×768 pixels image and the blurred and noisy image. On the bottom the images restored after computing 400 iterations of Algorithm 1 with $\mu = 1$ (left) and $\mu = 1/\sqrt{8}$ (middle), and DR1 (right).

Achieving lifting reduction in the dual variables: The reduction in the lifting with respect to the number of operators achieved in the algorithm here presented only affects the primal variables. It remains open the question of whether it is possible to reduce the dimension of the underlying space associated to the linearly composed operators. More precisely, if we consider the problem given by

$$\text{find } x \in \mathcal{H} \text{ such that } 0 \in \sum_{j=1}^m L_j^* B_j(L_j x),$$

is it possible to obtain an algorithm for solving this problem with $(0, m - 1)$ -fold lifting (according to Definition 3.8)? Or even with $(1, m - 1)$ or $(0, m)$ -fold lifting? All these questions remain open.

Acknowledgements FJAA and DTB were partially supported by the Ministry of Science, Innovation and Universities of Spain and the European Regional Development Fund (ERDF) of the European Commission, Grant PGC2018-097960-B-C22. FJAA was partially supported by the Generalitat Valenciana (AICO/2021/165). RIB was partially supported by FWF (Austrian Science Fund), project P 34922-N. DTB was supported by MINECO and European Social Fund (PRE2019-090751) under the program “Ayudas para contratos predoctorales para la formación de doctores” 2019.

References

- [1] Attouch, H., Théra, M.: A general duality principle for the sum of two operators. *J. Convex Anal.* 3, 1–24 (1996)
- [2] Bauschke, H.H., Combettes, P.L.: *Convex analysis and monotone operator theory in Hilbert spaces*, 2nd edn. Springer, Berlin (2017)
- [3] Boş, R.I., Csetnek, E.R., Heinrich, A.: A primal-dual splitting algorithm for finding zeros of sums of maximally monotone operators. *SIAM J. Optim.* 23(4), 2011–2036 (2013)
- [4] Boş, R.I., Csetnek, E.R., Heinrich, A., Hendrich, C.: On the convergence rate improvement of a primal-dual splitting algorithm for solving monotone inclusion problems. *Math. Program.* 150(2), 251–279 (2015)
- [5] Boş, 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] Boş R.I., Hendrich, C.: Solving monotone inclusions involving parallel sums of linearly composed maximally monotone operators. *Inverse Probl. Imaging* 10(3), 617–640 (2016)
- [7] Briceño-Arias, L., Combettes, P.L.: A monotone + skew splitting model for composite monotone inclusions in duality. *SIAM J. Optim.* 21(4), 1230–1250 (2011)
- [8] Campoy, R.: A product space reformulation with reduced dimension for splitting algorithms. To be published in *Comput. Optim. Appl.* (2022)
- [9] Chambolle, A.: An algorithm for total variation minimization and applications. *J. Math. Imaging Vis.* 20(1–2), 89–97 (2004)
- [10] Chambolle, A., Lions, P. L.: Image recovery via total variation minimization and related problems. *Numer. Math.* 76(2), 167–188 (1997)
- [11] 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)
- [12] Combettes, P.L., Pesquet, J.-C.: Primal-dual splitting algorithm for solving inclusions with mixture of composite, Lipschitzian, and parallel-sum type monotone operators. *Set-Valued Var. Anal.* 20(2), 307–330 (2012)
- [13] Condat, L., Kitahara, D., Contreras, A., Hirabayashi, A.: *Proximal Splitting Algorithms for Convex Optimization: A Tour of Recent Advances, with New Twists*. To be published in *SIAM Review* (2022)
- [14] Dao, M.N., Dizon, N., Hogan, J.A., Tam, M.K.: Constraint reduction reformulations for projection algorithms with applications to wavelet construction. *J. Optim. Theory Appl.* 190, 201–233 (2021)
- [15] Lions, P.L., Mercier, B.: Splitting algorithms for the sum of two nonlinear operators. *SIAM J. Numer. Anal.* 16(6), 964–979 (1979)
- [16] Malitsky, Y., Tam, M.K.: Resolvent splitting for sums of monotone operators with minimal lifting. Preprint (2021). <https://arxiv.org/abs/2108.02897>
- [17] Minty, G.J.: Monotone (nonlinear) operators in Hilbert space. *Duke Math. J.* 29, 341–346 (1962)
- [18] Pierra, G.: Decomposition through formalization in a product space. *Math. Program.* 28, 96–115 (1984)
- [19] Ryu, E.K.: Uniqueness of DRS as the 2-operator resolvent-splitting and impossibility of 3-operator resolvent-splitting. *Math. Program.* 182(1), 233–273 (2020)
- [20] Setzer, S., Steidl, G., Teuber, T.: Infimal convolution regularizations with discrete ℓ_1 -type functionals. *Commun. Math. Sci.* 9(3), 797–827 (2011)

- [21] Stollnitz, E.J., DeRose, T.D., Salesin, H.D.: Wavelets for Computer Graphics: A Primer, Part 1. *IEEE Comput. Graph. Appl.* 15(3), 76–84 (1995)
- [22] Vũ, B.C.: A splitting algorithm for dual monotone inclusions involving cocoercive operators. *Adv. Comput. Math.* 38, 667–681 (2013)

A Proof of the minimality theorem for parametrized resolvent splitting

Throughout this section, we assume that $n \geq 2$ and we denote by \mathcal{A}_n the set of all n -tuples of maximally monotone operators on \mathcal{H} . Hence, an element $A \in \mathcal{A}_n$ is of the form $A = (A_1, \dots, A_n)$, where $A_i : \mathcal{H} \rightrightarrows \mathcal{H}$ are maximally monotone operators for all $i \in \llbracket 1, n \rrbracket$. Every instance of Problem 1.2 is determined by the choice of $A \in \mathcal{A}_n$. In particular, when considering a fixed point encoding for this problem, the fixed point operator and the solution operator are both parametrized in terms of $A \in \mathcal{A}_n$. To emphasize this idea and to facilitate the exposition, we denote these operators by T_A and S_A in the following.

Let (T_A, S_A) be a d -fold lifted frugal parametrized resolvent splitting for Problem 1.2. By definition, there exists a finite procedure for evaluating T_A and S_A using only vector addition, scalar multiplication and the resolvents $J_{\delta_1 A_1}, \dots, J_{\delta_n A_n}$ precisely once, where $\delta = (\delta_1, \dots, \delta_n)^T$ is a vector of positive parameters. Following the same reasoning than in [16, Section 3], we can completely describe the evaluation of a point $\mathbf{z} = (z_1, \dots, z_d) \in \mathcal{H}^d$ by T_A with a series of equations. We directly present them here.

- (i) There exists $\mathbf{x} = (x_1, \dots, x_n) \in \mathcal{H}^n$ and $\mathbf{y} = (y_1, \dots, y_n) \in \mathcal{H}^n$ such that

$$\mathbf{x} = J_{\delta A}(\mathbf{y}) \iff 0 \in \mathbf{x} - \mathbf{y} + \delta A(\mathbf{x}), \quad (35)$$

where $\delta A := (\delta_1 A_1, \dots, \delta_n A_n) \in \mathcal{A}_n$.

- (ii) There exists $Y_z \in \mathbb{R}^{n \times d}$ and a lower-triangular matrix $Y_x \in \mathbb{R}^{n \times n}$ with zeros in the diagonal such that¹

$$\mathbf{y} = Y_z \mathbf{z} + Y_x \mathbf{x}. \quad (36)$$

- (iii) By frugality, there exists $T_z \in \mathbb{R}^{d \times d}$ and $T_x \in \mathbb{R}^{d \times n}$ such that

$$T_A(\mathbf{z}) = T_z \mathbf{z} + T_x \mathbf{x}. \quad (37)$$

Similarly, also by frugality, the evaluation of \mathbf{z} by the solution operator S can be expressed as

$$S_A(\mathbf{z}) = S_z \mathbf{z} + S_x \mathbf{x}, \quad (38)$$

where $S_z \in \mathbb{R}^{1 \times d}$ and $S_x \in \mathbb{R}^{1 \times n}$.

The proof of the next technical lemma can be obtained by following the same steps than in [16, Lemma 3.1], so we do not replicate it here.

Lemma A.1. *Let (T_A, S_A) be a frugal parametrized resolvent splitting for Problem 1.2. Let M denote the block matrix given by*

$$M := \begin{bmatrix} 0 & \text{Id} & -\text{Id} & \delta^T \text{Id} \\ Y_z & Y_x & -\text{Id} & 0 \\ T_z - \text{Id} & T_x & 0 & 0 \end{bmatrix}.$$

If $\mathbf{z} \in \text{Fix } T_A$, then there exists $\mathbf{v} = [\mathbf{z}, \mathbf{x}, \mathbf{y}, \mathbf{a}]^T \in \ker M$ with $\mathbf{a} \in A(\mathbf{x})$. Conversely, if $\mathbf{v} = [\mathbf{z}, \mathbf{x}, \mathbf{y}, \mathbf{a}]^T \in \ker M$ and $\mathbf{a} \in A(\mathbf{x})$, then $\mathbf{z} \in \text{Fix } T_A$, $\mathbf{x} = J_{\delta A}(\mathbf{y})$ and $S_A(\mathbf{z}) = S_z \mathbf{z} + S_x \mathbf{x}$.

Proposition A.2 (Solution operator). *Let (T_A, S_A) be a frugal parametrized resolvent splitting for Problem 1.2. Then, for all $\bar{\mathbf{z}} \in \text{Fix } T_A$ and $\bar{\mathbf{x}} = J_{\delta A}(\bar{\mathbf{y}})$, we have*

$$S_A(\bar{\mathbf{z}}) = \frac{1}{n} \sum_{i=1}^n (\bar{y}_i - \delta_i \bar{a}_i) = \bar{x}_1 = \dots = \bar{x}_n, \quad (39)$$

where $\bar{\mathbf{a}} = A(\bar{\mathbf{x}})$.

¹Here we make use of an abuse of notation. Indeed (36), should be written as $\mathbf{y} = (Y_z \otimes \text{Id})\mathbf{z} + (Y_x \otimes \text{Id})\mathbf{x}$, where \otimes denotes the Kronecker product.

Proof. Consider a particular instance of Problem 1.2 given by some operators $A \in \mathcal{A}_n$. Let T_A and S_A be the fixed point and solution operators of this particular instance, respectively. Let $\bar{\mathbf{z}} \in \text{Fix } T_A$ and $x^* = S_A(\bar{\mathbf{z}})$. By Lemma A.1, there exists $\mathbf{v} := [\mathbf{z}, \mathbf{x}, \mathbf{y}, \mathbf{a}]^T \in \ker M$ with $\bar{\mathbf{a}} \in A(\bar{\mathbf{x}})$ and $x^* = S_A(\bar{\mathbf{z}}) = S_z \bar{\mathbf{z}} + S_x \bar{\mathbf{x}}$.

Consider now the $n+1$ instances of Problem 1.2 given by the n -tuples of maximally monotone operators $A^{(0)}, A^{(1)}, \dots, A^{(n)} \in \mathcal{A}_n$ defined as

$$A^{(0)}(\mathbf{x}) := \bar{\mathbf{a}} \text{ and } A^{(j)}(\mathbf{x}) := \bar{\mathbf{a}} + \begin{bmatrix} 0 \\ \vdots \\ x_j - \bar{x}_j \\ \vdots \\ 0 \end{bmatrix} \quad \forall j \in \llbracket 1, n \rrbracket.$$

Since $\mathbf{v} \in \ker M$ and $\bar{\mathbf{a}} = A^{(j)}(\bar{\mathbf{x}})$, for all $j \in \llbracket 0, n \rrbracket$, Lemma A.1 implies that $\bar{\mathbf{z}} \in \text{Fix } T_{A^{(j)}}$, $\bar{\mathbf{x}} = J_{\delta A^{(j)}}(\bar{\mathbf{y}})$ and thus, $S_{A^{(j)}}(\bar{\mathbf{z}}) = S_z \bar{\mathbf{z}} + S_x \bar{\mathbf{x}} = x^*$ is a solution to every instance. Therefore, we have $0 = \sum_{i=1}^n A_i^{(0)}(x^*) = \sum_{i=1}^n \bar{a}_i$ and hence

$$0 = \sum_{i=1}^n A_i^{(j)}(x^*) = \sum_{i=1}^n \bar{a}_i + x^* - \bar{x}_j = x^* - \bar{x}_j \quad \forall j \in \llbracket 1, n \rrbracket,$$

from where it follows that $x^* = \bar{x}_1 = \dots = \bar{x}_n$. Finally, since $\bar{\mathbf{x}} = J_{\delta A^{(0)}}(\bar{\mathbf{y}})$, we have that $\bar{\mathbf{y}} - \bar{\mathbf{x}} = \delta A^{(0)}(\bar{\mathbf{x}}) = (\delta_1 \bar{a}_1, \dots, \delta_n \bar{a}_n)$. Consequently, $\sum_{i=1}^n \bar{y}_i - n x^* = \sum_{i=1}^n \delta_i \bar{a}_i$, which completes the proof. \square

Note that, although the expression for the solution operator given by (39) differs from the one obtained in [16, Proposition 3.2], it still holds that the vector $\bar{\mathbf{x}}$ belongs to the diagonal subspace of dimension n , which we denote by Δ_n . This is what we employ to prove the following theorem.

Theorem A.3. *Let (T_A, S_A) be a frugal parametrized resolvent splitting with d -fold lifting for Problem 1.2. Then $d \geq n - 1$.*

Proof. Suppose, by contradiction, that (T_A, S_A) is a frugal parametrized resolvent splitting for Problem 1.2 with d -fold lifting such that $d \leq n - 2$. Consider a particular instance of the problem given by $A \in \mathcal{A}_n$ such that $\text{zer}(\sum_{i=1}^n A_i) \neq \emptyset$ and take $\mathbf{z} \in \text{Fix } T_A$. By Lemma A.1, there exists $\mathbf{v} := [\mathbf{z}, \mathbf{x}, \mathbf{y}, \mathbf{a}]^T \in \ker M$ with $\mathbf{a} \in A(\mathbf{x})$. The last row of M implies that $0 = (T_z - \text{Id})\mathbf{z} + T_x \mathbf{x}$. Since $T_x \in \mathbb{R}^{d \times n}$ and $d \leq n - 2$, by the rank-nullity theorem, $\dim \ker T_x = n - \dim \text{rank } T_x \geq n - d \geq 2$. Since Δ_n is a subspace of dimension 1, there exists $\bar{\mathbf{x}} \notin \Delta_n$ such that $T_x \mathbf{x} = T_x \bar{\mathbf{x}}$.

Now, set $\bar{\mathbf{z}} := \mathbf{z}$, $\bar{\mathbf{y}} := Y_z \bar{\mathbf{z}} + Y_x \bar{\mathbf{x}}$ and $\bar{\mathbf{a}} := ((\bar{y}_1 - \bar{x}_1)/\delta_1, \dots, (\bar{y}_n - \bar{x}_n)/\delta_n)$ and consider the instance of the problem given by $\bar{A} \in \mathcal{A}_n$ defined as $\bar{A}(\mathbf{s}) := \bar{\mathbf{a}}$ for all $\mathbf{s} \in \mathcal{H}^n$. Then, $\bar{\mathbf{v}} := [\bar{\mathbf{z}}, \bar{\mathbf{x}}, \bar{\mathbf{y}}, \bar{\mathbf{a}}]^T \in \ker M$ with $\bar{\mathbf{a}} = \bar{A}(\bar{\mathbf{x}})$. By Lemma A.1 and Proposition A.2, this implies that $\bar{\mathbf{x}} \in \Delta_n$, obtaining thus a contradiction which completes the proof. \square