Algorithms based on operator-averaged operators
Miguel Simões

Abstract—A class of algorithms comprised by certain semismooth Newton and active-set methods is able to solve convex minimization problems involving sparsity-inducing regularizers very rapidly; the speed advantage of methods from this class is a consequence of their ability to benefit from the sparsity of the corresponding solutions by solving smaller inner problems than conventional methods. The convergence properties of such conventional methods (e.g., the forward–backward and the proximal–Newton ones) can be studied very elegantly under the framework of iterations of scalar-averaged operators—this is not the case for the aforementioned class. However, we show in this work that by instead considering operator-averaged operators, one can indeed study methods of that class, and also to derive algorithms outside of it that may be more convenient to implement than existing ones. Additionally, we present experiments whose results suggest that methods based on operator-averaged operators achieve substantially faster convergence than conventional ones.

Index Terms—Convex optimization, primal–dual optimization, semismooth Newton method, forward–backward method.

I. INTRODUCTION

ANY large-scale inverse problems in signal and image processing can be formulated as minimization problems whose objective functions are sums of proper lower-semicontinuous convex functions. An example is

\[
\min_{x \in \mathbb{R}^n} f(x) + g(x) + \sum_{j=1}^{N} h_j(L_j x),
\]

where \( f : \mathbb{R}^n \to (-\infty, +\infty] \) is assumed to be smooth—i.e., to be differentiable with a Lipschitz-continuous gradient—, and \( g : \mathbb{R}^n \to (-\infty, +\infty] \) and the \( N \) functions \( h_j : \mathbb{R}^{m_j} \to (-\infty, +\infty] \) with \( L_j \in \mathbb{R}^{m_j \times n} \) for \( j = \{1, \ldots, N\} \), may not be. Typical examples of problems that fit this formulation are problems where \( f \) is a data-fitting term, and \( g \) and \( h_j \) are regularizers. This is the case of the regression method known as least absolute shrinkage and selection operator (LASSO), where \( f = \|H : x - b\|_2^2 \), \( g = \mu \|\cdot\|_1 \), and \( N = 0 \), for given \( H \in \mathbb{R}^m \times n \), \( b \in \mathbb{R}^m \), and \( \mu > 0 \). The functions \( g \) and \( h_j \) can also be indicator functions of convex sets, allowing one to consider constrained problems within this framework. For example, if in the problem just discussed we additionally consider \( h_1 \) (with \( N = 1 \)) to be the indicator function of the set of nonnegative real numbers, i.e., if \( h_1 = \delta_{\mathbb{R}^+} \), we are constraining \( x \geq 0 \).

Methods known as splitting methods convert \( \mathbb{H} \) into a sequence of separable subproblems that are easier to solve than the original one. One way of studying the convergence properties of such methods is to consider that their iterations can be described by sequences of averaged operators (see below for a definition). This study is advantageous because it unifies and simplifies the analysis of a large number of algorithms. However, not all splitting methods can be described as sequences of averaged operators. This is the case, for example, of methods belonging to the class of algorithms comprised by semismooth Newton and active-set methods. Such methods have been applied to develop fast solvers for problems involving sparsity-inducing regularizers, since their iterations involve computing the solution of subproblems whose dimensionality is smaller than in other methods. In this work, we consider an extension of the concept of averaged operators, and use this extension to study some of the algorithms belonging to the class just discussed. Additionally, we develop new methods that do not belong to this class but that are instances of the aforementioned extension, and that have convenient practical applications. In the remainder of this section, we list the notation used throughout this work (Subsection I-A), provide a brief overview of the forward–backward (FB) method and some of its variants as to motivate our contributions (Subsection I-B), and conclude by enumerating those contributions and providing an outline of the paper (Subsection I-C).

A. Notation

Calligraphic uppercase letters denote real Hilbert spaces, as in \( \mathcal{X}, \mathcal{V} \). We denote the scalar product of a Hilbert space by \( \langle \cdot, \cdot \rangle \) and the associated norm by \( \|\cdot\| \). \( 2^\mathcal{V} \) denotes the power set of \( \mathcal{V} \), i.e., the set of all subsets of \( \mathcal{V} \). An operator (or mapping) \( A : \mathcal{X} \to \mathcal{V} \) maps each point in \( \mathcal{X} \) to a point in \( \mathcal{V} \). A set-valued operator \( A : \mathcal{X} \to 2^\mathcal{V} \) maps each element in \( \mathcal{X} \) to a set in \( \mathcal{V} \). I denotes the identity operator. \( B(\mathcal{X}, \mathcal{V}) \) denotes the space of bounded linear operators from \( \mathcal{X} \) to \( \mathcal{V} \); we set \( B(\mathcal{X}) \equiv B(\mathcal{X}, \mathcal{X}) \). Given an operator \( A \in B(\mathcal{X}, \mathcal{V}) \), its adjoint \( A^* \) is the operator \( A^* : \mathcal{V} \to \mathcal{X} \) such that for all \( x \in \mathcal{X} \) and \( u \in \mathcal{V} \), \( \langle Ax, u \rangle = \langle x, A^* u \rangle \). \( S(\mathcal{X}) \) denotes the space of self-adjoint bounded linear operators from \( \mathcal{X} \) to \( \mathcal{X} \); i.e., \( S(\mathcal{X}) \equiv \{ A \in B(\mathcal{X}) \mid A = A^* \} \). Given two operators \( A, B \in S(\mathcal{X}) \), the Loewner partial ordering on \( S(\mathcal{X}) \) is defined by \( A \succeq B \iff \langle Ax, x \rangle \geq \langle Bx, x \rangle, \forall x \in \mathcal{X} \). An operator \( A \) is said to be positive semidefinite if \( A \) is a self-adjoint bounded linear operator and \( A \succeq 0 \). Let \( \alpha \in [0, +\infty[ ; \mathcal{P}_\alpha(\mathcal{X}) \) denotes the space of positive semidefinite operators \( A \) such that \( A \succeq \alpha I \), i.e., \( \mathcal{P}_\alpha(\mathcal{X}) \equiv \{ A \in S(\mathcal{X}) \mid A \succeq \alpha I \} \). Given an operator \( A \in \mathcal{P}_\alpha \), its positive square root \( \sqrt{A} \) is the unique operator \( \sqrt{A} \in \mathcal{P}_\alpha \) such that \( (\sqrt{A})^2 = A \). For all \( A \in \mathcal{P}_\alpha \), we define a semi-product and a semi-norm (a scalar product and a norm if \( \alpha > 0 \)) by \( \langle \cdot, \cdot \rangle_A \equiv \langle \cdot, \cdot \rangle \) and by \( \| A \|_A \equiv \sqrt{\langle A, \cdot \rangle} \), respectively. The domain of a set-valued operator \( A : \mathcal{X} \to 2^{\mathcal{X}} \) is defined by dom \( A \equiv \{ x \in \mathcal{X} \mid A x \neq \emptyset \} \), its graph by gra \( A \equiv \{(x, u) \in \mathcal{X} \times \mathcal{X} \mid u \in A x\} \), the set of zeros by zer \( A \equiv \{ x \in \mathcal{X} \mid 0 \in A x \} \), the range of \( A \) by ran \( A \equiv \{ u \in \mathcal{X} \mid \exists x \in \mathcal{X} \ u = A x \} \), and the

M. Simões is with the Dept. Electr. Eng. (ESAT) – STADIUS, KU Leuven, Leuven, Belgium; e-mail: miguel.simoes@kuleuven.be.
inverse of \( A \) by \( A^{-1} : X \to 2^X : u \to \{ x \in X | u \in Ax \} \). We use the notation \( \{ x^k \} \) as a shorthand for representing the sequence \( \{ x^k \} \). We say that a sequence \( \{ x^k \} \) in \( H \) converges in the norm (or strongly converges) to a point \( x \) in \( H \) if \( \| x^k - x \| \to 0 \) and say that it converges weakly if, for every \( u \in H, \langle x^k, u \rangle \to \langle x, u \rangle \). We denote weak convergence by \( \xrightarrow{w} \). Strong convergence implies weak convergence to the same limit. In finite-dimensional spaces, weak convergence implies strong convergence. The space of absolutely-summable sequences in \( R \), i.e., the space of sequences \( \{ x^k \} \) in \( R \) such that \( \sum_k |x^k| < \infty \) is denoted by \( \ell^1(R) \). The set of summable sequences in \( [0, +\infty) \) is denoted by \( \ell_c^1(R) \). We denote by \( \bigoplus_{j \in \{1, \ldots, N\}} V_j \) the Hilbert direct sum [11 Example 2.1] of the Hilbert spaces \( V_j, j \in \{1, \ldots, N\} \). Given two set-valued operators \( A : X \to 2^Y \) and \( B : X \to 2^Z \), their parallel sum is \( A \square B \triangleq (A^{-1} + B^{-1})^{-1} \). Additionally, we denote by \( \mathbb{R} \) the set of real numbers, \( \mathbb{R}^n \) the set of real column vectors of length \( n \), and by \( \mathbb{R}^{m \times n} \) the set of real matrices with \( m \) rows and \( n \) columns. \( a^T \) denotes the transpose of a vector \( a \) and \( A^T \) denotes the transpose of a matrix \( A \). \( [a]_j \) denotes the \( i \)-th element of a vector \( a \), \([A]_{ij} \) denotes the \( j \)-th column of a matrix \( A \), and \( [A]_{ij} \) denotes the element of the \( i \)-th row and \( j \)-th column of a matrix. \( \| A \|_F \triangleq \sqrt{\text{Tr}(AA^T)} \) denotes the Frobenius norm of a matrix \( A \). Finally, let \( f : X \to ]-\infty, +\infty[ \) be a function. Its domain is denoted by \( \text{dom } f \triangleq \{ x \in X | f(x) < +\infty \} \) and its epigraph by \( \text{epi } f \triangleq \{ (x, s) \in \mathbb{R} \times X | f(x) \leq s \} \). The function \( f \) is lower semi-continuous if \( f \) is closed in \( X \times \mathbb{R} \) and convex if \( f \) is convex in \( X \times \mathbb{R} \). We use \( \Gamma_0(X) \) to denote the class of all lower semi-continuous convex functions \( f \) from \( X \) to \( ]-\infty, +\infty[ \) that are proper, i.e., such that \( \text{dom } f \neq \emptyset \). Given two functions \( f \in \Gamma_0(X) \) and \( g \in \Gamma_0(X) \), their infimal convolution is \( f \ast\ast g : X \to ]-\infty, +\infty[ : x \mapsto \inf_{u \in X} (f(u) + g(x - u)) \), where \( \inf \) denotes the infimum of a function. The indicator function of a set \( C \subseteq X \) is defined as \( \chi_C(x) \triangleq 0 \) if \( x \in C \) and \( \chi_C(x) \triangleq +\infty \) otherwise. When \( C \) is convex, closed and non-empty, \( \chi_C(x) \in \Gamma_0(X) \). The maximum and signum operators are denoted by \( \max(\cdot) \) and \( \text{sgn}(\cdot) \), respectively.

**B. Background**

We assume that all the optimization problems under consideration are convex and have at least one minimizer. For the convenience of the reader, we collect some notions on convex analysis and monotone operators, and on semismooth Newton and active-set methods in Appendices [3] and [4] respectively. Consider now that we want to solve (1) with \( N = 0 \), and that we use a splitting method for that effect, such as the FB one. This method is characterized by the iteration \( x^{k+1} \leftarrow \text{prox}_{\tau f} (x^k - \tau \nabla f(x^k)) \), which is performed consecutively for all \( k \in \mathbb{Z} \), and where \( \tau > 0 \). The FB method, also known as proximal–gradient, is a first-order method and can be seen as an extension of the gradient method to nonsmooth optimization [2], [3]. It can be shown that the iterates \( x^k \) of this method converge to a solution of (1) at a sublinear rate, or, under certain assumptions, at a linear rate. The inclusion of relaxation steps in the FB method may improve its convergence speed in practice; the resulting method is given by the iteration, for all \( k \in \mathbb{Z} \), of

\[
x^{k+1} \leftarrow x^k + \lambda^k \left( \text{prox}_{\tau g} (x^k - \tau \nabla f(x^k)) - x^k \right),
\]

where \( \lambda^k > 0 \) is the so-called relaxation parameter [4]. Additionally, it is also possible to consider the use of inertial steps. An example of such use is the method characterized by the iteration \( x^{k+1} \leftarrow (1 - \alpha_k)x^{k-1} + (\alpha_k - \lambda^k)x^k + \lambda^k \text{prox}_{\tau g} (x^k - \tau \nabla f(x^k)) \), \( \forall k \in \mathbb{Z} \), and with \( \alpha_k > 0 \). The convergence of this method was studied in [5] under some conditions on \( f \) and \( g \). The use of inertial steps is not limited to this specific formulation, and there are a number of alternatives [6], most notably the one studied in [7], which was shown to obtain the optimal convergence rate, in function values \( f(x^k) + g(x^k) \), for first-order methods. The variations discussed so far have virtually the same computational cost as the original FB method; this is not the case of the methods that we discuss next. The (relaxed) proximal–Newton method [8]–[12] makes use of second-order information about \( f \). It is characterized by the iteration \( x^{k+1} \leftarrow x^k + \lambda^k \left( \text{prox}_{\tau g} (x^k - [B^k]^{-1} \nabla f(x^k)) - x^k \right) \), where \( B^k \) is the Hessian of \( f \) at \( x^k \). The local convergence rate of the iterates \( x^k \) generated by this method can be shown to be quadratic under some conditions. However, computing \( \text{prox}_{\tau g} \) can be prohibitive in many problems of interest, and \( B^k \) is typically replaced by an approximation of the Hessian of \( f \) at \( x^k \); it can be shown that the iterates of the resulting method converge to a solution at a superlinear rate under some conditions. Finally, one can also consider incorporating a generalized derivative of \( \text{prox}_{\tau g} \). This is the case of the methods known as semismooth, whose relation to (2) we discuss in Section [11].

It is sometimes useful to consider splitting methods as instances of fixed-point methods. A solution \( \hat{x} \) of Problem (1) with \( N = 0 \) satisfies the fixed-point equation \( \hat{x} = \text{prox}_{\tau g} (\hat{x} - \tau \nabla f(\hat{x})) \), for \( \tau > 0 \), and it can be shown that the FB method produces a sequence of points that is Fejér monotone with respect to \( \text{Fix } \text{prox}_{\tau g}[-\tau \nabla f(\cdot)] \) [13]. More generally, if one wishes to find fixed points of a nonexpansive operator \( R \), one can consider the Krasnosel’skiĭ–Mann (KM) method, which is characterized by the iteration \( x^{k+1} \leftarrow T_{\lambda^k} (x^k) \triangleq x^k + \lambda^k (R(x^k) - x^k) \), where \( T_{\lambda^k} \) is termed a \( \lambda^k \)-averaged operator. By making \( R = \text{prox}_{\tau g}[-\tau \nabla f(\cdot)] \), we recover (2).

**C. Contributions and outline**

Consider yet another alternative to (2), where we replace the scalars \( \lambda^k \) by linear operators \( \Lambda^k \) such that, for every \( k, I \succ \Lambda^k \succ 0 \):

\[
x^{k+1} \leftarrow x^k + \Lambda^k \left( \text{prox}_{\tau g} [x^k - \tau \nabla f(x^k)] - x^k \right).
\]

The study of this iteration and, more generally, of the following extension of the KM scheme:

\[
x^{k+1} \leftarrow \sum_{k} (x^k) \triangleq x^k + \Lambda^k (R(x^k) - x^k),
\]

will be the basis of this work. For convenience, we call the operators \( \Lambda^k \), operator-averaged operators. This formulation
is rather general and allows one to consider extensions of current methods in two directions. Firstly, we consider a
generic nonexpansive operator $\mathbf{R}$ instead of just its instance as
$\mathbf{R} = \text{prox}_{\tau g}[-\tau \nabla f(\cdot)]$, as is commonly done in the literature
on semismooth Newton methods; this allows one to tackle
more complex problems, such as ones of the form of (1) and
Problem III.6 [cf. Subsection III-B]. Secondly, we make very
mild assumptions on the operators $\Lambda^k$ [cf. Section III]; this
allows one to consider a broad range of them, which may
incorporate second-order information about the problem to
solve (or not). We provide here what we consider to be an
interesting example of this flexibility. Consider again that we
want to solve (1) with $N = 0$. We can adress this problem
with an “intermediate” method between the proximal–Newton
and the semismooth Newton methods if we make $\Lambda^k$ to be
the inverse of the (possibly regularized) Hessian of $f$. This
removes the difficulty of computing the operator $\text{prox}_{\tau g}$,
and also of computing a generalized derivative of the typically
nonconvex and nonsmooth term $\text{prox}_{\tau g}$. This method is not,
strictly speaking, a second-order method, but it seems to have
fast convergence in practice [cf. Section IV].

The outline of this work is as follows. In Section II we list
work related to ours. In Section III we show that operator-
averaged operators have a contractive property and prove the
convergence of fixed-point iterations of these operators under
certain conditions. We base ourselves on the fact that they pro-
duce a sequence that is Fejér monotone (specifically, variable-
metric Fejér monotone) [14, 15]. In more detail, we show in
Subsection III-A how operator-averaged operators can be used to
extend an existing algorithm from the literature—a variable-
metric FB method, a generalization of the proximal–Newton
discussed above—, and we prove its convergence under certain
conditions. Additionally, we show in Subsection III-B how
operator-averaged operators can be used to solve a primal–dual
problem first studied in [16], which generalizes many convex
problems of interest in signal and image processing [15, 16].
In Section IV we discuss applications of the proposed method
to solve two inverse problems in signal processing, and discuss
a framework that contemplates the incorporation of existing
active-set methods, in a off-the-shelf fashion, to solve $\ell_2$-
regularized minimization problems. We defer all proofs to
Appendix C.

II. RELATED WORK

A similar formulation to (4) has been studied in the context
of numerical analysis. Consider that one wishes to solve the
system $\mathbf{A}x = \mathbf{b}$, where $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $\mathbf{b} \in \mathbb{R}^n$. This
system can be solved for $x$, under certain conditions, by iterating
$x^{k+1} \leftarrow x^k + \lambda^k (\mathbf{A}x^k - \mathbf{b})$. This iteration, whose form is
similar to the KM scheme discussed above, is known as the
Richardson iteration, and is a particular instance of a first-order
linear nonstationary iterative method (examples of others are
the Jacobi and the Gauss–Seidel iterations) [17, 18]. Some-
times, for computational reasons (e.g., if $\mathbf{A}$ is deemed to be too
ill conditioned), it is convenient to consider a preconditioned
version of the linear system: $\mathbf{P}_A \mathbf{x} = \mathbf{P}_b$, where $\mathbf{P}$ is an
invertible matrix. The corresponding preconditioned version
of the Richardson iteration is $x^{k+1} \leftarrow x^k + \lambda^k \mathbf{P} (\mathbf{A}x^k - \mathbf{b})$.
Now consider that one wishes to solve the fixed-point equation
of the operator $\mathbf{R}$. One could equally consider a left-
preconditioned scheme to solve this equation: find $x \in \mathcal{X}$
such that $\mathbf{A}\mathbf{R}(x) = \mathbf{A}x$. By mimicking the preconditioned
version of the Richardson iteration described above, one
obtains the iteration $x^{k+1} \leftarrow x^k + \lambda^k \left( \mathbf{R} (\mathbf{x}^k) - \mathbf{x}^k \right)$, which
corresponds to making $\Lambda^k$ fixed in (4), i.e., making, for all
$k$, $\Lambda^k = \Lambda$, where $\Lambda > 0$. The preconditioner $\Lambda$ is
different from the operator $\mathbf{B}^k$ discussed in Subsection III-A
and both can be used simultaneously: in Subsection III-A
we consider a version of the FB method that makes use of
the two. The core iteration of that version of the method is
$x^{k+1} \leftarrow x^k + \lambda^k \left( \text{prox}_{\tau g}^{\mathbf{B}^k} (x^k - [\mathbf{B}^k]^{-1} \nabla f (x^k)) - x^k \right)$.

Iterations of operator-averaged operators can also be seen as
iterations of a line-search method if one considers
$\lambda^k (\mathbf{R} (\mathbf{x}^k) - \mathbf{x}^k)$ to be a step in the direction of the fixed-
point residual $\mathbf{R} (\mathbf{x}^k) - \mathbf{x}^k$ with step-length parameter $\lambda^k$.
This idea has been explored in [19], where the authors
considered steps with length parameters $\lambda^k \geq 1$. The term
$\Lambda^k (\mathbf{R} (\mathbf{x}^k) - \mathbf{x}^k)$ can also be seen as indicating a search
direction by noting the similarities with second-order line-
search methods [20, Chapter 3]: incorporating second-order
information about the fixed-point residual in $\Lambda^k$ results in
a Newton-like method. However, other directions could be
taken by an appropriate selection of $\Lambda^k$. The idea of exploring
directions different from the one given by $\mathbf{R} (\mathbf{x}^k) - \mathbf{x}^k$ has
also been explored in recent work [21]. The authors studied an
algorithm comprised of two steps: the first step corresponds
to a step in a fast direction—making use of second-order
information about the problem at hand—and the second one
to a projection into a half-space. This projection is made to
guarantee convergence, and enforces Fejér monotonicity. The
extension of the KM scheme discussed in the present work
could also be used to extend the method from that paper, since
the use of operator-averaged operators expands the range of
directions that still satisfy a Fejér-monotonicity condition.

Additionally, we can establish parallels between the proposed
scheme (4) and coordinate-descent methods [22, 23].
Consider the case where $\{\Lambda^k\}$ is a sequence of diagonal
operators whose entries are either 0 or 1. These operators can
then be used to select a particular coordinate—or block of
coordinates—of $x$. Since the operator $\Lambda^k$ is binary, we have
$I \succeq \Lambda^k \succeq 0$ and not I $\succeq \Lambda^k \succ 0$, as we assumed initially.
Algorithms resulting from this choice of binary $\Lambda^k$ have been
studied elsewhere (e.g., [24, 25]) and are not analyzed in the
remainder of this paper.

We now discuss the connection with semismooth Newton
and active-set methods alluded to previously. If we make
$\Lambda^k = [\nabla (h(x^k))]^{-1}$ and $\mathbf{R} = \text{prox}_{\tau g}[-\tau \nabla f(\cdot)]$, where
$f = \|y - \mathbf{H} \cdot \|_2^2$, $g = \tau \|1\|_1$, and $\nabla (h(x^k))$ is defined to be the B-
differential of $\mathbf{R}$, we recover a semismooth Newton method
to address the LASSO problem [cf. Appendix B]. Other active-
set methods [26–28] can also be considered as particular
examples of the proposed scheme: in those cases, the operator
$\Lambda^k$ takes a different form.

Results discussed in this paper were first presented in
the PhD dissertation of the author [29, Subsections 5.2 and
III. AN EXTENSION OF THE KM SCHEME

In what follows we consider exclusively finite-dimensional spaces; this is reasonable since we are mainly concerned with digital signals and images. We also focus on optimization, and we do not discuss a more general approach, often found in the literature, that makes use of monotone operators instead of gradients and subgradients of convex functions. However, in the proofs provided in Section III, we avoided assumptions that prevented a generalization to infinite-dimensional spaces; we also developed the proofs in the setting of monotone operators.

Definition III.1 (Operator-averaged operators). Let $D$ be a nonempty subset of $\mathcal{X}$, let $\varepsilon \in [0, 1]$, and let $\Lambda$ be an operator in $\mathcal{X}$ such that $\mu I \succeq \Lambda \succeq \alpha I$, where $\mu, \alpha \in [\varepsilon, 1 - \varepsilon]$. We say that an operator $T_{\Lambda} : D \to \mathcal{X}$ is an operator-averaged operator, or, more specifically, a $\Lambda$-averaged operator, if there exists a nonexpansive operator $R : D \to \mathcal{X}$ such that $T_{\Lambda} \triangleq (I - \Lambda) + \Lambda R$.

We have proved the following results:

Proposition III.2. Let $D$ be a nonempty subset of $\mathcal{X}$, let $\varepsilon \in [0, 1]$, let $\Lambda$ be an operator in $\mathcal{X}$ satisfying $\mu I \succeq \Lambda \succeq \alpha I$, where $\mu, \alpha \in [\varepsilon, 1 - \varepsilon]$, let $\mathbf{R} : D \to X$ be a nonexpansive operator, and let $T_{\Lambda} : D \to \mathcal{X}$ be a $\Lambda$-averaged operator. Then the operator $T_{\Lambda}$ is $\mu$-averaged in the metric induced by $\Lambda^{-1}$. In other words, the operator $T_{\Lambda}$ verifies

$$\|T_{\Lambda}(x) - T_{\Lambda}(y)\|_{\Lambda^{-1}} \leq \|x - y\|_{\Lambda^{-1}} - \frac{\mu}{\alpha \mu} \frac{1}{\|I - T_{\Lambda}\|_{\Lambda^{-1}}} \|x - y\|_{\Lambda^{-1}}$$

for all $x, y \in D$.

Theorem III.3. Let $D$ be a nonempty closed convex subset of $\mathcal{X}$, let $\varepsilon \in [0, 1]$, let $\eta^k \in \ell^+_1([\varepsilon, 1 - \varepsilon])$, let $\{\Lambda^k\}$ be a sequence of operators in $B(\mathcal{X})$ such that, for all $k \in \mathbb{N}$, $\mu^k \varepsilon \Lambda^k \geq \alpha^k I$, with $\mu^k, \alpha^k \in [\varepsilon, 1 - \varepsilon]$ and $(1 + \eta^k)\Lambda^{k+1} \geq \Lambda^k$, and let $R : D \to D$ be a nonexpansive operator such that $\text{Fix } R \neq \emptyset$. Additionally, let $x^0 \in D$ and let, for all $k$, $\{x^k\}$ be a sequence generated by $\mathbf{A}$. Then $\{x^k\}$ converges to a point in $\text{Fix } R$.

A. An extension of a variable-metric FB method

In this subsection, we show how operator-averaged operators can be used to extend proximal–Newton and variable-metric FB methods of the type introduced in Subsection III-B. Consider Algorithm 1 to solve Problem 1 with $N = 0$. In what follows, $\{U^k\}, \{\Lambda^k\}$ are sequences of bounded linear operators, and $\{a^k\}, \{b^k\}$ are absolutely summable sequences that can be used to model errors. The following theorem establishes some convergence properties of this algorithm.

**Algorithm 1**

1: Choose $x^0 \in \mathcal{X}$
2: $k \leftarrow 1$
3: **while** stopping criterion is not satisfied **do**
4: Choose $\gamma^k > 0$, $U^k > 0$, and $I \succ \Lambda^k > 0$
5: $y^k \leftarrow x^k - \gamma^k U^k (\nabla f(x^k) + b^k)$
6: $x^{k+1} \leftarrow x^k + \Lambda^k \left(\text{prox}_{\gamma^k g} y^k + a^k - x^k\right)$
7: $k \leftarrow k + 1$
8: **end while**

**Theorem III.4.** Let $g \in \Gamma_0(\mathcal{X})$, let $\beta \in ]0, +\infty[$, and let $f : \mathcal{X} \to \mathbb{R}$ be convex and differentiable with a $1/\beta$-Lipschitzian gradient. Let $\{U^k\}$ be a sequence of operators in $\mathcal{X}$ such that, for all $k \in \mathbb{N}$, $\mu_0 I \succeq U^k \succeq \alpha_0 I$ and $\mu_k \leq \alpha_k I$, with $\mu, \alpha \in ]0, +\infty[$, and let $\epsilon \in ]0, \min\{1, 2\beta/(\alpha U + 1)\}$, let $\{\Lambda^k\}$ be a sequence of operators in $\mathcal{X}$ such that, for all $k$, $\Lambda^k U^k = U^k \Lambda^k$, with $\mu \succeq \Lambda^k \succeq \alpha I$ and $\mu, \alpha \in [\varepsilon, 1]$, and let $\epsilon^k \in \ell_1^+(\mathbb{N})$, and let $(1 + \eta^k)\Lambda^{k+1} U^{k+1} \succeq \Lambda^k U^k$. Let $\{\gamma^k\}$ be a sequence in $[\varepsilon, (2\beta - \epsilon)/\mu_k]$ and let $\{a^k\}, \{b^k\} \in \ell_1^+(\mathbb{N})$.

Suppose that $Z = \text{Argmin} (f + g) \neq \emptyset$. Let $\{x^k\}$ be a sequence generated by Algorithm 1. Then the following hold:

1) $\{x^k\}$ is quasi-Fejér monotone with respect to $Z$ relative to $\{(\Lambda^k U^k)^{-1}\}$.
2) $\{x^k\}$ converges weakly to a point in $Z$.

**Remark III.5.** The assumption that, for every $k$, $U^k$ and $\Lambda^k$ commute, i.e., $\Lambda^k U^k - U^k \Lambda^k = 0$, may seem to be severe. However, take into account that existing algorithms consider one of these operators to be the identity operator in semismooth Newton methods, $U^k = I$, $\forall k$, whereas in variable-metric FB methods, $\Lambda^k = I$, $\forall k$.

B. Primal–dual optimization algorithms

Combettes and Pesquet studied a primal–dual problem that generalizes many other problems [10] Problem 4.1. By being able to devise an algorithm to solve this problem, we are effectively tackling a large number of problems simultaneously—problem 1 is one of these.

Let $g \in \Gamma_0(\mathcal{X})$, let $\mu \in ]0, +\infty[$, let $f : \mathcal{X} \to \mathcal{X}$ be convex and differentiable with a $\mu^{-1}$-Lipschitzian gradient, and let $z \in \mathcal{X}$. Let $N$ be a strictly positive integer; for every $j \in \{1, \ldots, N\}$, let $r_j \in v_j$, let $h_j \in \Gamma_0(v_j)$, let $v_j \in ]0, +\infty[$, let $l_j \in \Gamma_0(v_j)$ be $v_j$-strongly convex, let $L_j \in B(\mathcal{X}, v_j)$ such that $L_j \neq 0$, and let $w_j$ be real numbers in $]0, 1]$ such that $\sum_{j=1}^N w_j = 1$. The problem is as follows:

**Problem III.6.** Solve the primal minimization problem,

$$\minimize_{x \in \mathcal{X}} g(x) + \sum_{j=1}^N w_j (h_j(x) l_j(x - r_j) + f(x) - f(x) - z)$$
together with its corresponding dual minimization problem,
\[
\begin{align*}
\text{minimize} & \quad (y^* \ast \text{int } h^*) \left( z - \sum_{j=1}^N \omega_j L_j^* d_j \right) \\
& \quad + \sum_{j=1}^N \omega_j \left( h_j^*(d_j) + l_j^*(d_j) + \langle d_j, r_j \rangle \right).
\end{align*}
\]

The sets of solutions to these primal and dual problems are denoted by \( P \) and \( D \), respectively. We recover (1) by making \( N = 1, r_1 = 0, z = 0 \), and \( l_1 : u \to 0 \) if \( u = 0 \) and \( l_1 : u \to +\infty \) otherwise.

Consider Algorithm 2 to solve Problem III.6. In what follows, for all \( j \), \( \{U^k\}, \{A^k\}, \{U^k_j\}, \{A^k_j\} \) are sequences of linear operators, and \( \{a^k\}, \{b^k\}, \{c^k\}, \{e^k\} \) are absolutely-summable sequences that can be used to model errors. Algorithm 2 is an extension of [15] Example 6.4. The following corollary establishes some convergence properties.

**Corollary III.7.** Suppose that \( \mathbf{z} \in \text{ran} \left( (g + \sum_{j=1}^N \omega_j L_j^* (\partial h_j \ast \text{int } \partial l_j) (L_j : -r_j) + \nabla f) \right) \) and set \( \beta \triangleq \min \{\mu, \nu_1, \ldots, \nu_N\} \). Let \( \{U^k\} \) be a sequence of operators in \( X \) such that, for all \( k \in \mathbb{N} \), \( \mu U^k \succeq \alpha U \) with \( \mu U, \alpha U \in [0, +\infty] \), let \( \epsilon \in [0, \min\{1, \beta\}] \), let \( \{A^k\} \) be a sequence of operators in \( X \) such that, for all \( k \), \( A^k U^k = U^k A^k \) with \( \mu U \succeq \alpha_I \) and \( \alpha \in [\epsilon, 1] \). Let \( \{U^k_j\} \) be a sequence of operators in \( V_j \) such that, for all \( k \), \( \mu U^k_j \succeq \alpha U \) with \( \mu U, \alpha U \in [0, +\infty] \), let \( \{A^k_j\} \) be a sequence of operators in \( V_j \) such that, for all \( k \), \( A^k_j U^k_j = U^k_j A^k_j \) with \( \mu \succeq \alpha \) and \( \alpha \in [\epsilon, 1] \). Let \( \{a^k\}, \{b^k\}, \{c^k\}, \{e^k\} \) be absolutely-summable sequences generated by Algorithm 2. Then \( x^k \) converges to a point in \( P \) and \( (d^k_1, \ldots, d^k_N) \) converges to a point in \( D \).

**IV. EXPERIMENTS**

**A. Inverse integration**

We give a practical example of a simple problem that can be solved via Algorithm 2. Consider the constrained problem

\[
\text{minimize } \|b - Hx\|^2 + \mu \|x\|_1,
\]

where \( b \in \mathbb{R}^n \), \( c \in \mathbb{R}^d \), \( \mu > 0 \), \( H = \frac{1}{n} \tilde{H} \), and \( \tilde{H} \in \mathbb{R}^{n \times n} \) is a lower-triangular matrix of ones. Griesse and Lorenz studied a non-constrained, and therefore simpler, version of this problem in the context of inverse integration [32 Section 4.1]. Problem (5) can be solved via Algorithm 2— we discuss implementation details elsewhere [50 Section IV]. We simulate an example similar to the one studied by Griesse and Lorenz [32 Section 4.1] but consider the noise to be Gaussian with a signal-to-noise ratio (SNR) of 30 dB. We have set \( \mu = 3 \times 10^{-5} \), \( c = -80 \), and \( d = 52 \). We compared Algorithm 2 (denoted in what follows as Proposed) with the alternating-direction method of multipliers (ADMM) and with the method by Condat (CM) [33] to solve (5). We manually tuned the different parameters of the three methods in order to achieve the fastest convergence results in practice. We arbitrarily chose the result of ADMM after \( 10^7 \) iterations as representative of the solution given by the three methods. Fig.1 illustrates the behavior of the three methods by showing the root-mean-squared error (RMSE) between the estimates of each method and the representative solution, as a function of time. The three methods were initialized with the zero vector. The experiments were performed using MATLAB on an Intel Core i7 CPU running at 3.20 GHz, with 32 GB of RAM.

In this example, we did not enforce any assumptions on \( A^k \) but verified in practice that they were satisfied. However, in more complex examples, it may be necessary to devise a strategy that generates a sequence \( \{A^k\} \) satisfying the assumptions of Corollary III.7. This is akin to the necessity of devising
globalization strategies in other Newton-like methods [34]. Chapter 8].

B. Plug-and-play methods

Recently, a number of works have explored a technique known as plug-and-play ADMM to solve $\ell_1$-regularized problems with quadratic data-fitting terms. In that technique, the user substitutes one of the steps of the regular ADMM by an off the shelf, state-of-the-art denoising algorithm [35], [36]. Here, we explore a similar idea by considering the use of a semismooth Newton (or other active-set method) to replace a step of ADMM. Whereas the concern of the former approach is to improve the SNR of the estimates produced by it, our goal is instead to improve convergence speed. Consider problems of the form of (1) with $f = \frac{1}{2} \|Hx - b\|^2$, $N = 1$, and where we further assume that $g(x) \in \Gamma_0(\mathbb{R}^n)$ is a sparsity-inducing regularizer. Existing semismooth Newton methods do not consider the existence of a separate term $h$, and, in general, are not able to solve this problem efficiently, since that would require the computation of a (generalized) derivative of $\text{prox}(g+h)$. In what follows, we discuss an algorithm that can be used to solve this problem, that does not require that computation, and that allows the use of an existing semismooth Newton method in a plug-and-play fashion. We propose the following algorithm:

**Algorithm 3**

1: Choose $x^0 \in \mathbb{R}^n$, $v^0 \in \mathbb{R}^n$, $d^0 \in \mathbb{R}^n$, $\tau > 0$

2: $k \leftarrow 1$

3: while stopping criterion is not satisfied do

4: $\mathbf{p}^k = \text{prox}_{\tau g}(x^k - \tau(\nabla f(x^k) + \gamma(x^k - v^k + d^k)))$

5: $x^{k+1} = x^k + \Lambda^k(\mathbf{p}^k - x^k)$

6: $v^{k+1} = \text{prox}_{\gamma}(x^{k+1} + d^k)$

7: $d^{k+1} = d^k + (x^{k+1} - v^{k+1})$

8: $k \leftarrow k + 1$

9: end while

Lines 4 and 5 of Algorithm 3 take a form similar to Algorithm 1 and, in principle, be replaced by any active-set method [26]–[28], [39]–[40]. In the following corollary, we discuss some of the convergence guarantees of this algorithm by showing that it is an instance of Algorithm 2. Note that not all existing active-set methods will strictly obey the convergence conditions given here.

**Corollary IV.1.** Suppose that $0 \in \text{ran } (\partial g + \partial h + \nabla f)$. Set $\beta = \|H^*H\|$ and let $\epsilon \in [0, \min\{1, \beta\}]$. For every $k \in \mathbb{N}$, set $\delta_k = \frac{1}{\beta} - 1$ and suppose that $\delta_k \geq \frac{1}{\beta k^2 \epsilon} \geq \frac{1}{2\beta\epsilon}$ holds. Let $\{\Lambda^k\}$ be sequences of operators satisfying $\mu I \succeq \Lambda^k \succeq \alpha I$, $\mu, \alpha \in [\epsilon, 1]$, and let $\{x^k\}$ be a sequence generated by Algorithm 3. Then $x^k$ converges weakly to a solution of (1) with $f = \frac{1}{2} \|Hx - b\|^2$ and $N = 1$.

C. Spectral unmixing

Hyperspectral images are images with a relatively large number of channels—usually known as spectral bands—corresponding to short frequency ranges along the electromagnetic spectrum. Frequently, their spatial resolution is low, and it is of interest to identify the materials that are present in a given pixel; a pixel typically corresponds to a mixture of different materials—known as endmembers—, each with a corresponding spectral signature. Spectral unmixing techniques produce a vector of abundances, or percentages of occupation, for each endmember, in each pixel [41]. Consider that we want to spectrally unmix a hyperspectral image with $N$ bands. We assume that the set of spectral signatures of the endmembers that may be present in a pixel is known through a database of $P$ materials (i.e., a database of reflectance profiles as a function of wavelength). We formulate the problem for a given pixel $j$ as

$$
\begin{align*}
\text{minimize} & \quad \|\mathbf{y}_{h,j} - \mathbf{u}_j\|_2^2 + \mu \|\mathbf{a}_j\|_1 \\
\text{subject to} & \quad \mathbf{a}_j \succeq 0, \quad i = \{1, \cdots, P\}.
\end{align*}
$$

(6)

where $\mathbf{a}_j \in \mathbb{R}^P$ is the vector of each endmember’s abundances for pixel $j$, to be estimated, $\mathbf{u} \in \mathbb{R}^{N \times P}$ is a matrix corresponding to the spectral database, $\mathbf{y}_{h,j} \in \mathbb{R}^{N \times M}$ corresponds to a matrix representation of a hyperspectral image with $M$ pixels (i.e., corresponds to the lexicographical ordering of a 3-D data cube), and $\beta$ is a regularization parameter. We tested two spectral dictionaries $\mathbf{U}$: a randomly generated dictionary and a real-world one. The former is given by sampling i.i.d. the standard Gaussian distribution, and the latter by a selection of 498 different mineral types from a USGS library, set up as detailed in [42]. The problem was generated as follows: we start by generating a vector of abundances with $P = 224$ and with 5 nonzero elements, where the abundances are drawn from a Dirichlet distribution; we made $N = P$ and $M = 100$, and added Gaussian noise such that it would result in a SNR of 40 dB. For this example, we implemented Algorithm 3, although we did not replace lines 4 and 5 by any off-the-shelf method but by a direct implementation of those lines, which can be done as detailed in [30, Section IV]. Furthermore, we compared two versions of the proposed method, corresponding to different choices for the sequence of operators $\{\Lambda^k\}$. The first, denoted as Proposed - variable, corresponds to the inverse B-differential of $(p^k - x^k)$; this choice is similar to the one made for the example detailed in Subsection IV-A. The second, denoted as Proposed - fixed, corresponds to making $\Lambda^k = \Lambda$ for all $k$, where $\Lambda$ corresponds to the scaled inverse of the regularized Hessian of $\|\mathbf{y}_{h,j} - \mathbf{u}_j\|_2^2$. By regularized, we mean that we added the matrix $\epsilon I$ with $\epsilon = 10^2$ to the Hessian before computing the inverse, and by scaled, that we scale the resulting inverse in order to guarantee that it obeys the condition $\mu I \succeq \Lambda$; in other words, we made $\Lambda^{-1} = \rho (\|\mathbf{y}_{h,j}\|_2^2 \mathbf{y}_{h,j} + \epsilon I)$, for a given $\rho$. We compared the two version of the proposed method as detailed in Subsection IV-A.

V. Conclusions

This work discusses the use of operator-averaged operators to construct algorithms with fast convergence properties. These are particularly suitable to address problems with sparsity-inducing regularizers (and possibly other regularizers and
the choice of an automatic strategy to select the sequence of $\Lambda^k$ operators, as well as possible ways to relax some of the assumptions on them.

**ACKNOWLEDGMENTS**

This work was supported by the Internal Funds KU Leuven project PDMT1/21/018. I thank Prof.s Bioucas Dias, Borges de Almeida, and Panagiotis Patrinos for comments on how to improve this work. I dedicate this paper to the memory of Prof. Bioucas Dias.

**APPENDIX A**

**ON CONVEX ANALYSIS AND ON MONOTONE OPERATORS**

The notions of subgradient and subdifferential of a convex function (in the sense of Moreau and Rockafellar [43, Chapter 23]) are useful when dealing with nonsmooth functions. A vector $p \in \mathbb{R}^n$ is said to be a subgradient of a function $g \in \Gamma_0(\mathbb{R}^n)$ at a point $x \in \mathbb{R}^n$ if $g(y) \geq g(x) + \langle p, y - x \rangle$, $\forall y \in \mathbb{R}^n$. The set of all subgradients of $g$ at $x$ is called the subdifferential of $g$ at $x$ and is denoted by $\partial g(x)$. The set-valued operator $\partial g : \mathbb{R}^n \to 2^{\mathbb{R}^n} : x \to \partial g(x)$ is called the subdifferential of $g$. For a differentiable function $f \in \Gamma_0(\mathbb{R}^n)$, the subdifferential at $x$ is a singleton, i.e., $\partial f(x) = \{\nabla f(x)\}$. The subdifferential operator is critical to our interests. We recall Fermint’s rule [1, Theorem 16.2]: $x$ is a minimum of a proper convex function $g$ if and only if $0 \in \partial g(x)$. This means that the minimizers of $g$ are the zeros of the operator $\partial g$. A smooth surrogate of a function $g \in \Gamma_0(\mathbb{R}^n)$ is the so-called Moreau envelope. It is defined by $\tau g(x) \triangleq \inf_{u \in \mathbb{R}^n} \{g(u) + \frac{1}{\tau^2}\|x - u\|^2\}$, where $\tau > 0$. The function $\tau g$ is continuously differentiable, even if $g$ is not. Both functions share the same minimizers [1 Proposition 12.9(iii)]. The proximal operator of $g$ is $\text{prox}_{\tau g}(x) \triangleq \arg \min_{u \in \mathbb{R}^n} \{g(u) + \frac{1}{\tau^2}\|x - u\|^2\}$, which is simply the point that achieves the infimum of the Moreau envelope (this point is unique, since $u \to g(u) + \frac{1}{\tau^2}\|x - u\|^2$ is strictly convex). The proximal operator of $g$ relative to the norm $\| \cdot \|_2^2$, where $U \in \mathbb{R}^{n \times n}$ and $U > 0$, is $\text{prox}_U g(x) \triangleq \arg \min_{u \in \mathbb{R}^n} \{g(u) + \frac{1}{2}\|x - u\|^2_2\}$. When $g(x) = \|x\|_1$, $\text{prox}_{\tau g}(x)$ can be evaluated component-wise. The proximal operator for each element reduces to the so-called soft-thresholding operator [44], which is given by $[\text{prox}_{\tau\|\cdot\|_1}(x)]_i = \max(\{\|x_i\|_1 - \tau, 0\}) \text{sign}(x_i)$, $\forall i \in \{1, \ldots, n\}$. The proximal operator of the indicator function when the set $C \in \mathbb{R}^n$ is convex, closed and non-empty is an Euclidean projection onto $C$, which we denote by $P_C(x) \triangleq \arg \min_{u \in C} \|x - u\|^2_2$. For the proximal operators of more functions see, e.g., [43] or the website [http://proximity-operator.net](http://proximity-operator.net). Closely related to these ideas is the concept of Legendre–Fenchel conjugate of a function $f$, which is defined as $f^* : \mathbb{R}^n \to [-\infty, +\infty] : x \to \sup_{u \in \mathbb{R}^n} \{\langle x, u \rangle - f(u)\}$. We recall some of its properties. Consider that $f \in \Gamma_0(\mathbb{R}^n)$. Then $f^* \in \Gamma_0(\mathbb{R}^n)$ and, by the Fenchel–Moreau theorem [1 Theorem 13.32], the biconjugate of $f$ (the conjugate of the conjugate) is equal to $f$, i.e., $f^{**} = f$. Another property is that [1 Proposition 16.9] $u \in \partial f(x) \iff x \in \partial f^*(u)$, $\forall x, u \in \mathbb{R}^n$, or, in other

Figure 2. RMSE, as a function of time, between the estimates of each iteration and the representative solution, for the four methods (CM, ADMM, Proposed - variable, and Proposed - fixed). Top: Gaussian dictionary U; bottom: real dictionary U.
words, $\partial f^* = (\partial f)^{-1}$. The notion of conjugate is also important in
establishing the so-called Moreau's decomposition, 
$\text{prox}_{\tau g}(x) + \tau \text{prox}_{g/\tau}(x/\tau) = x, \forall x \in \mathbb{R}^n$. Finally, we say that $f$ is strongly convex if $f - \frac{\alpha}{2} \left\| x \right\|^2$ is convex, with $\alpha > 0$.

An operator $A : X \to 2^X$ is said to be monotone if 
$(u - v, x - y) \geq 0, \forall (x, u) \in \text{gra } A, \forall (y, v) \in \text{gra } A$. An operator is maximally monotone if there exists no other monotone operator whose graph properly contains gra $A$. As an example, if $g \in \Gamma_0(A'), \text{then } \partial g$ is maximally monotone (Theorem 20.40). Let $\beta \in [0, +\infty[$; we say that an operator is strongly monotone with constant $\beta$ if $A - \beta I$ is monotone. Monotone operators are connected to optimization problems as follows. Take, for example, Problem (I) with $N = 0$. According to Fermat’s rule, its solutions should satisfy the inclusion $0 \in \nabla f(x) + \partial g(x)$. Consequently, solving this problem can be seen as a particular case of the problem of finding a zero of the sum of two monotone operators $A$ and $C$ acting on a Hilbert space $X$, i.e., find $x \in X$, such that $0 \in A(x) + C(x)$, if ones makes $A = \nabla f$ and $C = \partial g$. We now list some properties of operators. We say that an operator $A$ is Lipschitz continuous with constant $L > 0$ if $\left\| u - v \right\| \leq L \left\| x - y \right\|, \forall (x, u) \in \text{gra } A, \forall (y, v) \in \text{gra } A$. When $L = 1$, the operator $A$ is said to be nonexpansive; when $L < 1$, it is said to be contractive. Let $D$ be a nonempty set of $X$ and let $R : D \to X$ be a nonexpansive operator. We say that an operator $A : D \to \mathcal{X}$ is $\lambda$-averaged if there exists $\lambda \in \{0, 1\}$ such that $A = (1- \lambda)I + \lambda R$. An averaged operator $A$ obeys the following contractive property (Proposition 4.25): $\left\| A(x) - A(y) \right\|^2 \leq \left\| x - y \right\|^2 - \frac{1- \lambda}{\lambda} \left\| (I - A)(x) - (I - A)(y) \right\|^2, \forall x \in D, \forall y \in D$. When $\lambda = 1/2$, $A$ is said to be firmly nonexpansive. Proximal operators are examples of firmly nonexpansive operators (Corollary 23.8). Let $\beta \in [0, +\infty[$; we say that a (single-valued) operator $C : D \to \mathcal{X}$ is $\beta$-cocoercive if $(C(x) - C(y), x - y) \geq \beta \left\| C(x) - C(y) \right\|^2, \forall x \in D, \forall y \in D$. An operator $C$ is $\beta$-cocoercive if and only if $\beta C$ is $\frac{1}{\beta}$-averaged (Remark 4.24(iv)). Let $f \in \Gamma_0(X)$ and let $\nabla f$ be $\beta$-Lipschitz continuous. Then, according to the Baillon–Haddad theorem (Corollary 18.16), $\nabla f$ is $\frac{1}{\beta}$-cocoercive.

In order to prove under which conditions iterative algorithms such as the ones that we discuss in this work solve optimization problems, it can be useful to consider fixed-point methods. The set of fixed points of an operator $A : X \to X$ is Fix $A \triangleq \{x \in X \mid x = A(x)\}$. If $A$ is a Lipschitz-continuous operator, Fix $A$ is closed and (Proposition 4.14). If $A$ is nonexpansive, Fix $A$ is closed and convex (Corollary 4.15). Fixed-point methods try to find the fixed points of an operator (if they exist) by producing a sequence of points $\{x^k\}$ that should converge to one of them, given an initial point $x^0 \in X$. A sequence $\{x^k\}$ is said to be Fejér monotone with respect to a nonempty closed and convex set $S$ in $X$ if $\left\| x^{k+1} - x \right\| \leq \left\| x^k - x \right\|, \forall x \in S$. Such a sequence is bounded. Consequently, it possesses a subsequence that converges weakly to a point $x \in X$. Such a point is said to be a weak sequential cluster point of $\{x^k\}$, and we denote the set of weak sequential cluster points of $\{x^k\}$ by $W$. Interestingly, it is also a consequence of Fejér monotonicity that a necessary and sufficient condition for the sequence $\{x^k\}$ to converge weakly to a point in $S$ is that $W \subset S$.[13], [1] Chapters 2 and 5. It is sometimes useful to consider the notions of quasi-Fejér monotonicity and of Fejér monotonicity relative to a variable metric. A sequence $\{x^k\}$ is said to be quasi-Fejér monotone with respect to a nonempty closed and convex set $S$ in $X$ if $\left\| x^{k+1} - x \right\| \leq \left\| x^k - x \right\| + \epsilon^k, \exists \{\epsilon^k\} \subset \ell_1^2(\mathbb{N}), \forall x \in S$, and it said to be Fejér monotone with respect to a nonempty closed and convex set $S$ in $X$ relative to a sequence $\{V^k\}$ if $\left\| x^{k+1} - x \right\| \leq \left(1 + \eta^k\right)\left\| x^k - x \right\|, \forall x \in S$ such that (a) $V^k \in \mathcal{P}_+(X)$ for $\alpha \in [0, +\infty[$ and $\forall k \in \mathbb{N}$, (b) $\sup_k \|V^k\| < \infty$, and (c) $(1 + \eta^k) V^{k+1} \geq V^k$ with $\{\eta^k\} \subset \ell_1^2(\mathbb{N})$ and $\forall k \in \mathbb{N}$. The zeros of a monotone operator can be found by using fixed-point methods on appropriate operators. Let $A : X \to 2^X$ be a maximally monotone operator and assume that zero $A \neq \emptyset$. Associated to this operator is its resolvent $J_{\tau A} \triangleq (I + \tau A)^{-1}$. The set of fixed points of $J_{\tau A}$ coincides with the set of zeros of $A$ (Proposition 23.38). It can be shown that $J_{\tau A}$ is firmly nonexpansive (Proposition 23.7) and that $J_{\tau \partial g} = \text{prox}_{\tau g}$ (Example 23.3).

If one wishes to find a fixed point of a nonexpansive operator $R$, one may use the KM method [cf. Subsection [8]]. Under certain conditions, it can be show that the sequence $\{x^k\}$ is Fejér monotone and that it converges weakly to a point in $\text{Fix } R$, even when $R$ is merely nonexpansive (Proposition 5.15). This method can be seen as a sequence of iterations of averaged operators since the operators $T^\lambda_k$ are $\lambda^k$-averaged operators. In order to prove the zeros of $A$, one may consider a scheme based on an iteration of the form $x^{k+1} = J_{\tau A} x^k$, $x^0 \in X$, which converges since $J_{\tau A}$ is $1/2$-averaged.

APPENDIX B

ON SEMISMooth NEWTON METHODS

We briefly discuss the use of semismooth Newton methods to solve the LASSO problem, which, we recall, consists in the minimization of the cost function $\|Hx - b\|^2_2 + \mu \|x\|_1$. To address problems such as this, it was shown by Hintermüller [26] that the use of certain semismooth Newton methods is equivalent to the use of active-set ones (see below for a definition). The minimizer of the LASSO problem is assumed to be sparse; if we know which of its entries are zero, instead of solving this problem, we can solve an equivalent problem. Let $\hat{x} \in \mathbb{R}^n$ be a solution to the problem, let $\hat{A}$ denote the set comprising the indices of the entries of $\hat{x}$ that are zero, and let $\hat{I}$ denote the set comprising the remaining indices. The alternative problem

$$
\begin{align*}
\text{minimize}_{x \in \mathbb{R}^n} & \quad \|y - Hx\|^2_2 + \mu \|x\|_1 \\
\text{subject to} & \quad \|x\|_i = 0, i \in \hat{A}
\end{align*}
$$

(7)

shares the same set of solutions. By writing the objective function of the latter problem as $\|y - (H^\ast)^j x^j\|^2 + \mu \|x\|_1$ with $j \in \hat{I}$, it is clear that this problem has a much smaller number of non-constrained variables than the original. If operations involving $[H^\ast]^j$ are cheaper to perform than the ones involving the full matrix $H$, an optimization algorithm will typically solve this problem faster than the original one. In practice, we do not know beforehand which entries of $\hat{x}$ are zero. Active-set methods address this issue by finding
estimates of the sets $\hat{A}$ and $\hat{I}$ by following some predefined strategy; the choice of strategy determining how both sets are estimated yields different algorithms [47]–[53]. Define $G : \mathbb{R}^n \to \mathbb{R}^n : u \mapsto u - \text{prox}_{\mu G}^\parallel \parallel u, (u - 2 \tau H^+(Hu - y))$. The solution to the LASSO problem should satisfy the non-linear equation $G(x) = 0$. This equation is nonsmooth, since $\text{prox}_{\mu G}^\parallel \parallel$ is not everywhere differentiable. There are, however, generalizations of the concept of differentiability that are applicable to nonconvex operators such as $G$. One of them is the B(ouligand)-differentiable [54] Definition 4.6.2, which is defined as follows. Suppose that a generic operator $G : D \subset \mathbb{R}^n \to \mathbb{R}^m$ is locally Lipschitz, where $D$ is an open subset. Then by Rademacher’s theorem, $G$ is differentiable almost everywhere in $D$. Let $C$ denote the subset of $\mathbb{R}^n$ consisting of the points where $G$ is differentiable (in the sense of Fréchet [1] Definition 2.45). The B-differential of $G$ at $x$ is $\partial_B G(x) \triangleq \{ \lim_{x_i \to x} \nabla G(x^i) \}$, where $\{ x^i \}$ is a sequence such that $x^i \in C$ for all $j$ and $\nabla G(x^i)$ denotes the Jacobian of $G$ at $x^i$. As an example critical to our interests, consider the operator $\text{prox}_{\mu G}^\parallel \parallel$. Its B-differential $B = \partial_B \text{prox}_{\mu G}^\parallel \parallel (x)$ is given by [32] Proposition 3.1 $[B]_{ii} = 1$ if $\|x_i\|_i > \tau$, $[B]_{ii} = 0$ if $\|x_i\|_i < \tau$, and $[B]_{ii} \in \{0, 1\}$ if $\|x_i\|_i = \tau$. Since the B-differential of an operator at a given point may not be unique, it may be convenient to consider a single $V \in D$, for example, the binary diagonal matrix $[V]_{ii} = 1$ if $|x_i| > \tau$ and $[V]_{ii} = 0$ otherwise.

The generalization of the concept of differentiability just discussed can also be used to formulate the so-called semismooth Newton method based on the B-differential, which is characterized by the iteration $x^{k+1} \leftarrow x^k - [V^k]^\parallel \parallel G(x^k)$, where $V^k \in \partial_B G(x^k)$. It can be shown that this method locally converges superlinearly for operators known as semismooth [55]. Let $x \in D$ and $d \in \mathbb{R}^n$; semismooth operators are operators that are directionally differentiable at a neighborhood of $x$ and that, for any $V \in \partial_B G(x + d)$, satisfy the condition $Vd - G'(x,d) = 0$. $G'(x,d)$ denotes the directional derivative [1] Definition 17.1 of $G$ at $x$ along $d$ and $o(\cdot)$ denotes little-O notation. Examples of semismooth functions are the Euclidean norm and piecewise-differentiable functions [56] Chapter 2, $\text{prox}_{\mu G}^\parallel \parallel$ being an example of the latter. For more details on these methods, see [56]–[59].

Appendix C

Proofs

This section includes the proofs of all the propositions, theorems, and corollaries of this work. It starts with a preliminary result.

Preliminary Result

Lemma C.1. Let $\alpha \in [0, +\infty]$ and let $V \in \mathcal{P}_\alpha(\mathcal{X})$. Then, for all $x,y \in \mathcal{X}$,

$$
\|Vx + (I - V)y\|^2 = \langle V(V-I)(x-y), x-y \rangle + \|x\|_V^2 - \|y\|_V^2 + \|y\|^2.
$$

Proof. Fix $x$ and $y$ in $\mathcal{X}$. Then $\|Vx + (I - V)y\|^2 = \langle Vx, Vx \rangle + 2 \langle Vx, (I - V)y \rangle + \langle (I - V)y, (I - V)y \rangle = \langle Vx, Vx \rangle + 2 \langle Vx, y \rangle - 2 \langle Vx, Vy \rangle = \langle Vx, Vx \rangle - \langle Vx, y \rangle.$

Proof of Proposition III.2

Fix $x$ and $y$ in $\mathcal{X}$. In $D$. By making $V \triangleq \Lambda^{-1}$, we have $\alpha^{-1}I \preceq V \preceq \mu^{-1}I$, and, by noting that $R = (I - V) + VT_A$, we verify that $\|Rx - Ry\|^2 = \|V(x - y) + (VAx - TAy)\|^2 = \|V(x - y)\|^2 + \|VAx - TAy\|^2$. Since $\|V(x - y)\|^2 = \langle V(x - y), y \rangle = \langle (V - I)(x - (T_A - I)y), (T_A - I)x - (T_A - I)y \rangle + \|TAx - TAy\|^2 - \|x - y\|^2 + \|x - y\|^2.$

Proof of Theorem III.3

1) Straightforward.

2) Since $x^{0} \in D$ and $D$ is convex, [4] produces a well-defined sequence in $D$. By making $V^{k} \triangleq (\Lambda^{k})^{-1}I, \forall k$, [14] Lemma 2.1] yield, for all $k$, that $(\alpha^{k})^{-1}I \succeq V^{k+1} \succeq (\mu^{k})^{-1}I$ with $(1 + \eta^{k})V^{k} \succeq V^{k+1}$. [4] implies that, for all $k$, $\|V^{k+1} - x^{*}\|^{2}_{V^{k}} = \|T_Ax^{k} - TA_{x^{*}}\|^{2}_{V^{k}} \leq \|x^{k} - x^{*}\|^{2}_{V^{k}} - \frac{1 - \mu^{k}}{\mu^{k}}\|I - T_{A_{x^{*}}}\|^{2}_{V^{k}} + \frac{1 - \mu^{k}}{\mu^{k}}\|x^{k+1} - x^{k}\|^{2}_{V^{k}}$
where step (i) follows from Proposition 12. Since, for any given \( z \in X \), we verify that 
\[
(1 + \eta^k) \| \bar{x}^k - x^k \|_{V_k}^2 \leq \| z \|_{V_k}^2,
\]
\( \forall k \in \mathbb{N} \), using (10), we get 
\[
\eta^k \| x^{k+1} - x^k \|_{V_k}^2 \leq (1 + \eta^k) \| x^{k+1} - x^k \|_{V_k}^2 \leq (1 + \eta^k) \| x^k - x^* \|_{V_k}^2.
\]
3) Since \( \{ x^k \} \) is Fejér monotone with respect to \( \{ V_k \} \), the sequence \( \{ \| x^k - x^* \|_{V_k}^2 \} \) converges to 0 Proposition 3.1). Define \( \zeta \triangleq \sup_k \| x^k - x^* \|_{V_k} < +\infty \). It follows from (3) that, for all \( k \),
\[
\| x^{k+1} - x^k \|_{V_k}^2 \leq \| x^k - x^* \|_{V_k}^2 + \frac{1 - \mu_k}{\mu_k} \| x^{k+1} - x^k \|_{V_k}^2.
\]
and, in view of (9), we can write
\[
\| x^{k+1} - x^* \|_{V_k}^2 \leq (1 + \eta^k) \left( \| x^k - x^* \|_{V_k}^2 - \frac{1 - \mu_k}{\mu_k} \| x^{k+1} - x^k \|_{V_k}^2 \right).
\]
and
\[
\| x^{k+1} - x^k \|_{V_k}^2 \leq (1 + \eta^k) \| x^k - x^* \|_{V_k}^2 - \alpha^k (1 - \mu_k) \| x^{k+1} - x^k \|_{V_k}^2 \leq \| x^k - x^* \|_{V_k}^2 + \zeta \eta^k - \alpha^k \| x^{k+1} - x^k \|_{V_k}^2 \leq \| x^k - x^* \|_{V_k}^2 + \zeta \eta^k - \alpha^k \| x^{k+1} - x^k \|_{V_k}^2 \leq \| x^k - x^* \|_{V_k}^2 + \zeta \eta^k.
\]
For every \( K \in \mathbb{N} \), by iterating (12) we can write that
\[
\| x^k - x^* \|_{V_k}^2 \leq \| x^k - x^* \|_{V_k}^2 + \zeta \eta^k - \alpha^k \| x^{k+1} - x^k \|_{V_k}^2 \leq \| x^k - x^* \|_{V_k}^2 + \zeta \eta^k.
\]
and
\[
\| x^k - x^* \|_{V_k}^2 \leq \| x^k - x^* \|_{V_k}^2 + \zeta \eta^k.
\]
4) Let \( x \) be a weak sequential clustering point of \( \{ x^k \} \). It follows from (1) Corollary 4.18 that \( x \in \text{Fix R} \). In view of (14) Lemma 2.3] and [14 Theorem 3.3], the proof is complete.

**Proof of Theorem 3.4**

We start by proving a more general version of Algorithm 1. Consider that Lines 5 and 6 were replaced by
\[
y^k \leftarrow x^k - \frac{\eta^k}{\lambda^k} U^k (C^k x^k + b^k) \quad \text{and} \quad x^{k+1} \leftarrow x^k + \Delta^k (J_{A^k} U^k A^k x^k + a^k - x^k),
\]
respectively. Additionally, let \( A : X \rightarrow 2^X \) be a maximally monotone operator, let \( \beta \in (0, +\infty) \), let \( C \) be a \( \beta \)-cocoercive operator, and suppose instead that \( Z = \text{zer} (A + C) \neq \emptyset \). This more general version of Algorithm 1 allows one to address the problem of finding \( x \in X \) such that \( 0 \in A(x) + C(x) \).

Define, for all \( k \),
\[
A^k \triangleq \frac{\eta^k}{\lambda^k} U^k A, \quad C^k \triangleq \frac{\eta^k}{\lambda^k} U^k C, \quad \Phi^k \triangleq U^k \Delta^k, \quad p^k \triangleq J_{A^k} U^k A, \quad q^k \triangleq J_{A^k} (x^k - C^k x^k), \quad s^k \triangleq x^k + \Phi^k (q^k - x^k).
\]
We have from (15) Eq. (4.8)] that
\[
\| x^k - q^k \|_{(U^k)^{-1}} \leq \frac{2 \delta}{\sqrt{\nu^k}} \| b^k \|.
\]
Additionally, for any \( x^* \in Z \), from (15) Eq. (4.12)] we can write that
\[
\| q^k - x^* \|_{(U^k)^{-1}} \leq \| x^k - x^* \|_{(U^k)^{-1}} - \frac{2 \delta}{\sqrt{\nu^k}} \| b^k \|.
\]
We now establish some identities. For all \( k \), we have that
\[
\| x^{k+1} - s^k \|_{(\Phi^k)^{-1}} = \| x^{k+1} + \Phi^k (p^k + a^k - x^k) - (x^k + \Phi^k (q^k - x^k)) \|_{(\Phi^k)^{-1}} \leq \| \Phi^k (p^k + a^k - q^k) \|_{(\Phi^k)^{-1}} \leq \| \Phi^k a^k \|_{(\Phi^k)^{-1}} + \| \Phi^k (p^k - q^k) \|_{(\Phi^k)^{-1}} \leq \| \sqrt{\| U^k \|^{-1}} \sqrt{\Phi^k} a^k \| + \| \sqrt{\| U^k \|^{-1}} \sqrt{\Phi^k} (p^k - q^k) \|_{(U^k)^{-1}} \leq \left( \| \sqrt{U^k} \|^{-1} \right) \| \sqrt{\Phi^k} a^k \| + \left( \| \sqrt{U^k} \|^{-1} \right) \| \sqrt{\Phi^k} (p^k - q^k) \|_{(U^k)^{-1}} \leq \sqrt{\frac{1}{\| U^k \|}} \| \sqrt{\Phi^k} a^k \| + \frac{2 \beta - \epsilon}{\sqrt{\| U^k \|}} \| b^k \| \quad \text{and that} \quad \| s^k - x^* \|_{(\Phi^k)^{-1}} \leq \frac{2 \delta}{\sqrt{\nu^k}} \| b^k \|.
\]
We now establish some identities. For all \( k \), we have that
\[
\| x^{k+1} - s^k \|_{(\Phi^k)^{-1}} = \| x^{k+1} + \Phi^k (p^k + a^k - x^k) - (x^k + \Phi^k (q^k - x^k)) \|_{(\Phi^k)^{-1}} \leq \| \Phi^k (p^k + a^k - q^k) \|_{(\Phi^k)^{-1}} \leq \| \Phi^k a^k \|_{(\Phi^k)^{-1}} + \| \Phi^k (p^k - q^k) \|_{(\Phi^k)^{-1}} \leq \| \sqrt{\| U^k \|^{-1}} \sqrt{\Phi^k} a^k \| + \| \sqrt{\| U^k \|^{-1}} \sqrt{\Phi^k} (p^k - q^k) \|_{(U^k)^{-1}} \leq \left( \| \sqrt{U^k} \|^{-1} \right) \| \sqrt{\Phi^k} a^k \| + \left( \| \sqrt{U^k} \|^{-1} \right) \| \sqrt{\Phi^k} (p^k - q^k) \|_{(U^k)^{-1}} \leq \sqrt{\frac{1}{\| U^k \|}} \| \sqrt{\Phi^k} a^k \| + \frac{2 \beta - \epsilon}{\sqrt{\| U^k \|}} \| b^k \| \quad \text{and that} \quad \| s^k - x^* \|_{(\Phi^k)^{-1}} \leq \frac{2 \delta}{\sqrt{\nu^k}} \| b^k \|.
\]
Finally, these identities yield $\|x^{k+1} - q^k\|_{(p_{k+1})^{-1}}^2 \leq (1 + \eta^k)^2 \|x^{k+1} - s^k\|_{(p_{k+1})^{-1}}^2 \leq (\epsilon^k)^2$.

We are now able to prove quasi-Fejér monotonicity of \{x^k\}: $\|x^{k+1} - x^*\|_{(p_{k+1})^{-1}} \leq \leq (1 + \eta^k)^2 \|x^k - x^*\|_{(p_k)^{-1}}^2 + \epsilon^k$. (19)

Since \{a^k\} and \{b^k\} are absolutely summable, $\sum_k \epsilon^k < +\infty$. From the assumptions, in view of [14 Proposition 4.1(i)], we conclude that \{x^k\} is quasi-Fejér monotone with respect to $Z$ relative to \{(p_k)^{-1}\}.

As a consequence of [7] and [14 Proposition 4.1(ii)], $\|x^k - x^*\|_{(p_k)^{-1}}$ converges. We define $\zeta \triangleq \sup_k \|x^k - x^*\|_{(p_k)^{-1}} < +\infty$.

Moreover, $\|x^{k+1} - x^*\|_{(p_{k+1})^{-1}} =$

$\leq \|s^k - x^*\|_{(p_{k+1})^{-1}} + \|x^{k+1} - s^k\|_{(p_{k+1})^{-1}} + 2 \|s^k - x^*\|_{(p_{k+1})^{-1}} - \|x^{k+1} - s^k\|_{(p_{k+1})^{-1}}$.

$\|x^{k} - x^*\|_{(p_k)^{-1}}^2 - \|x^{k+1} - x^*\|_{(p_{k+1})^{-1}}^2$.

For every $k \in \mathbb{N}$, by iterating (21), we can write that $\epsilon^2 \sum_{k=0}^{K} \|Ck - Cx^k\|^2 \leq$

$\leq \|x^0 - x^*\|_{(p_0)^{-1}}^2 + \|x^{f+1} - x^*\|_{(p_{f+1})^{-1}}^2$.

$\sum_{k=0}^{K} (\zeta^2 \eta^k + 2 \delta \epsilon \eta^k + (\epsilon^k)^2)$.

$\leq \zeta^2 + \sum_{k=0}^{K} (\zeta^2 \eta^k + 2 \delta \epsilon \eta^k + (\epsilon^k)^2)$.

Taking the limit from this inequality as $K \to +\infty$ yields $\sum_{k=0}^{\infty} \|Ck - Cx^k\|^2 \leq \frac{1}{\zeta^2} \sum_{k=0}^{\infty} (\zeta^2 \eta^k + 2 \delta \epsilon \eta^k + (\epsilon^k)^2) < +\infty$.

Following a similar reasoning, we can show that $\sum_{k=0}^{\infty} \|x^k - q^k\| - (C^k x^k - C^k x^k)^2 \|_{(U_{k})^{-1}} < +\infty$.

Let $x \in Z$ be a weak sequential limit point of $\{x^k\}$. In view of [14 Theorem 3.3], it remains to show that $x \in Z$. Since the sequences \{\|Ck - Cx^k\|^2\} and \{\|x^k - q^k\| - (C^k x^k - C^k x^k)^2 \|_{(U_{k})^{-1}} < +\infty$.

Finally, by making $A = \partial g$ and $C = \nabla f$, we recover the original Algorithm 1. By [11 Theorem 20.40], $\partial g$ is maximally monotone and, by [11 Corollary 18.16], $\nabla f$ is $\beta$-cocoercive. Additionally, Argmin $(f + g) = \text{zer} (\partial f + \partial g)$, by [11 Corollary 26.3].

**Proof of Corollary III.2**

The proof provided here follows the structure of similar proofs [13, 16, 60]. As in the proof of Theorem III.2, we start by proving a more general version of Algorithm 2 consider that Lines 6 and 11 were replaced by $\mathbf{q}_j \leftarrow \mathbf{U}_j \mathbf{B}_j^{-1} (\mathbf{d}_j + \mathbf{U}_j^k (\mathbf{L}_j x^k - \mathbf{E}_j \mathbf{d}_j - \mathbf{e}_j^k - \mathbf{r}_j)) + \mathbf{b}_j$ and $\mathbf{p}_j \leftarrow \mathbf{U}_j \mathbf{A} (x^j - \mathbf{U}_j \left(\sum_{j=1}^{N} \omega^j \mathbf{L}_j^* \mathbf{y}_j^k + \mathbf{C} \mathbf{x}^k + \mathbf{c}^k - \mathbf{z}\right)) + \mathbf{a}^k$, respectively. Additionally, let $A : \mathcal{X} \to 2^{\mathcal{X}}$ be a maximally monotone operator, let $\mu \in [0, +\infty]$, and let $C : \mathcal{X} \to \mathcal{X}$ be $\mu$-cocoercive; for every $j \in \{1, \ldots, N\}$, let $\mathbf{B}_j : \mathcal{V}_j \to 2^{\mathcal{V}_j}$ be maximally monotone, let $\mathbf{E}_j : \mathcal{V}_j \to 2^{\mathcal{V}_j}$ be maximally monotone and $\mu_j$-strongly monotone, and suppose instead that $z \in \text{ran} (A + \sum_{j=1}^{N} \omega^j \mathbf{L}_j^* (\mathbf{B}_j \mathbf{E}_j (\mathbf{L}_j x - \mathbf{r}_j)) + \mathbf{C})$.

This more general version of Algorithm 2 allows one to address the following primal–dual problem: solve the primal inclusion of finding $x \in \mathcal{X}$ such that $z \in \mathcal{X} + \sum_{j=1}^{N} \omega^j \mathbf{L}_j^* (\mathbf{B}_j \mathbf{E}_j (\mathbf{L}_j x - \mathbf{r}_j)) + \mathbf{C}$ together with the dual inclusion of finding $d_1 \in \mathcal{V}_1, \ldots, d_N \in \mathcal{V}_N$ such that \exists $x \in \mathcal{X}$ and $\mathbf{z} \in \sum_{j=1}^{N} \omega^j \mathbf{L}_j^* d_j \in \mathcal{X} + \mathcal{X}$ and $d_j \in (\mathbf{B}_j \mathbf{E}_j (\mathbf{L}_j x - \mathbf{r}_j))$, for all $j \in \{1, \ldots, N\}$.

We start by introducing some notation. We denote by $\mathcal{V}$ the Hilbert direct sum of the real Hilbert spaces $\mathcal{V}_j \in \{1, \ldots, N\}$. i.e., $\mathcal{V} = \bigoplus_{j \in \{1, \ldots, N\}} \mathcal{V}_j$. We endow this space with the following scalar product and norm, respectively: $\langle \cdot, \cdot \rangle_{\mathcal{V}} : (a, b) \mapsto \sum_{j=1}^{N} \omega^j \langle a_j, b_j \rangle$ and $\|\cdot\|_{\mathcal{V}} : a \mapsto \sqrt{\sum_{j=1}^{N} \omega^j \|a_j\|^2}$, where $a = (a_1, \ldots, a_j), b = (b_1, \ldots, b_j) \in \mathcal{V}$. Additionally, we denote by $\mathcal{K}$ the Hilbert direct sum $\mathcal{K} = \mathcal{X} \oplus \mathcal{V}$ and endow the resulting space with the following scalar product and norm, respectively: $\langle \cdot, \cdot \rangle_{\mathcal{K}} : ((x, a), (y, b)) \mapsto \langle x, y \rangle + \langle a, b \rangle_{\mathcal{V}}$ and $\|\cdot\|_{\mathcal{K}} : (x, a) \mapsto \sqrt{\|x\|^2 + \|a\|_{\mathcal{V}}^2}$, where $x, y \in \mathcal{X}$.

We define, for all $k \in \mathbb{N}$, $d^k \in \mathcal{V} \triangleq (d_{1, k}, \ldots, d_{N, k})$, $x^k \in \mathcal{K} \triangleq (x^k, d^k)$, $y^k \in \mathcal{K} \triangleq (a^k, q^k, \ldots, q^k)$, $a^k \in \mathcal{K} \triangleq (a^k, b_1^k, \ldots, b_N^k)$, $c^k \in \mathcal{K} \triangleq (c^k, e_1^k, \ldots, e_N^k)$, $f^k \in \mathcal{K} \triangleq (f^k, b_1^k, \ldots, b_N^k)$, $\mathbf{U}_k : \mathcal{K} \to \mathcal{K} : (x, a) \mapsto (\sum_{j=1}^{N} \omega^j L^*_j a_j - z + Ax)$, $\mathbf{V}_k : \mathcal{K} \to \mathcal{K} : (x, a) \mapsto (U^k x, U^k a, U^k N a_N)$, $\Lambda^k : \mathcal{K} \to \mathcal{K} : (x, a) \mapsto (A^k x, A^k_1 a_1, \ldots, A_N^k a_N)$.
\( \mathcal{K} \rightarrow \mathcal{K} : (x, a) \rightarrow (\Lambda^k U^k x, \Lambda^k U^k a_1, \ldots, \Lambda^k U^k a_N) \), where we note that the definition of the operator \( V^k \) is not the same as the equivalent operator in [15 Eq. (6.10)].

We can further rewrite Lines 6 and 7 of Algorithm 2 as 

\[
(U_j^*)^T (d_j^e - q_j^e) + L_j x^k - E_j^T d_j^e \in r_j + B_j^{-1} (q_j^e - b_j^e) + e_j - (U_j^k)^T b_j^e
\]

where Line 11 as \( (x^k - p_k) + \sum_{j=1}^N \omega_j L_j^* (d_j^e - q_j^e) - C x^k \in -z + A (p_k - a_k) + \sum_{j=1}^N \omega_j L_j^* q_j^e + \epsilon_k - (U^k)^T a_k \). In turn, these two modifications can be collapsed into the inclusion \( (x^k - y^k) - C x^k \in A (y^k - a_k) + S a_k + \epsilon_k - f_k \), whereas Lines 8 and 12 can be rewritten as

\[
\alpha + \sqrt{\sum_{j=1}^N \|L_j\|^2} \leq \rho \quad \text{and that, for every } x \in X
\]

\[
\frac{1}{\alpha_0} + \sqrt{\sum_{j=1}^N \|L_j\|^2} \leq \rho \quad \text{and that, for every } x \in \mathcal{X}
\]

\[
\mathcal{K} \rightarrow \mathcal{K} : (x, a) \rightarrow \left( (\Lambda^k U^k x, \Lambda^k U^k a_1, \ldots, \Lambda^k U^k a_N) \right)
\]

where step (i) follows from the identity \( 2 (a, b) \geq -\|a\|^2 - \|b\|^2 \) and step (ii) follows from the fact that \( (1 + \delta^k) \beta^k = 1 \). Following the arguments made in [15 Eqs. (6.16)-(6.18)], this last inequation implies that \( \sup_k \left\| V^{k-1} \right\| \leq 1 - 2 \beta - \epsilon \) and \( (V^{k+1})^{-1} \geq (V^k)^{-1} \in P_{\Lambda/\rho} (\mathcal{K}) \). It follows from the assumptions of the present corollary that \( \sup_k \left\| A^k \right\| \leq 1 \), \( \alpha_k \geq A^k \in P_{\Lambda/\rho} (\mathcal{K}) \), and \( \Phi \geq \Phi^k \). Moreover, it follows from [15 Lemma 3.1] that \( \sum_k \|a_k\|_{\mathcal{K}} \leq +\infty \), \( \sum_k \|e_k\|_{\mathcal{K}} \leq +\infty \), and \( \sum_k \|f_k\|_{\mathcal{K}} \leq +\infty \). It is shown in [16 Eq. (13.3)] that \( z (A + C) \neq 0 \). Additionally, following the arguments made in [16 Eqs. (3.21) and (3.22)], if \( (x^*, d^*) \in \text{zer} (A + C) \), then \( x^* \in P \) and \( d^* \in D \). Consequently, we have a \( x^* \rightleftharpoons (x^*, d_1^*, \ldots, d_J^*) \) such that \( x^* \in \text{zer} (A + C) \) and \( x^k \rightharpoondown x^* \).

By making, for every \( j, A = \delta g, C = \nabla \mu, B_j = \partial \delta h_j \), and \( E_j = \partial l_j \), we recover the original Algorithm 2. The current corollary is proven by using the same arguments as in [16 Theorem 4.2].

**Proof of Corollary 11/17**

Define, for every \( k \in \mathbb{N} \), \( y^k \rightarrow 2d^{k+1} - \bar{d}^k, \bar{d}^k = \gamma d^k \), and note that for every \( k \in \mathbb{N} \) and \( \alpha \) of Algorithm 5 can be rewritten as \( d^{k+1} = d^k + x^{k+1} - \text{prox}_{\gamma} (x^{k+1} + d^k) \). We can rewrite Lines 6 of Algorithm 5 by unfolding this algorithm:

\[
\text{prox}_{\gamma} (x^{k+1} + d^k), \quad d^{k+1} = \frac{1}{\alpha_0} + \frac{1}{\alpha_0} \text{ prox}_{\gamma} (x^k + d^k)
\]

and \( x^{k+1} = x^k + \alpha (p_k - x^k) \), respectively. Note that \( x^k - v^{k+1} + d^{k+1} = x^k - \text{prox}_{\gamma} (x^k + d^k) \) and \( \text{prox}_{\gamma} (x^k + d^k) = \frac{1}{2} \text{ prox}_{\gamma} (x^k + d^k) - \frac{1}{2} \text{ prox}_{\gamma} (x^k + d^k) = \frac{1}{2} \text{ prox}_{\gamma} (x^k + d^k) - \frac{1}{2} \text{ prox}_{\gamma} (x^k + d^k) = \frac{1}{2} \text{ prox}_{\gamma} (x^k + d^k)
\]

The current corollary is proven by invoking Corollary 11/7.

**REFERENCES**

[1] H. Bauschke and P. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. New York, NY, USA: Springer, 2011.

[2] M. Figueiredo and R. Nowak, “An EM algorithm for wavelet-based image restoration,” IEEE Trans. Image Process., vol. 12, no. 8, pp. 906–916, Aug 2003.

[3] I. Daubechies, M. Defrise, and C. De Mol, “An iterative thresholding algorithm for linear inverse problems with a sparsity constraint,” Comm. Pure Appl. Math., vol. 57, no. 11, pp. 1413–1457, 2004.

[4] P. Combettes and V. Wajs, “Signal recovery by proximal forward–backward splitting,” SIAM J. Multiscale Model. Simul., vol. 4, no. 4, pp. 1168–1200, 2005.

[5] J. M. Bioucas-Dias and M. A. T. Figueiredo, “A new TwIST: Two-step iterative shrinkage/thresholding algorithms for image restoration,” IEEE Trans. Image Process., vol. 16, no. 12, pp. 2992–3004, Dec 2007.

[6] D. A. d’Aspremont, D. Scieur, and A. Taylor, “Acceleration Methods,” in *Proc. 25th Int. Conf. Neural Informat. Process. Systems*, Lake Tahoe, Nevada, 2012, pp. 2618–2626.
